The Power of Experimental Approaches to Social Choice

by

Ben Armstrong

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2025

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner: Zinovi Rabinovich
Assistant Professor, School of Computer Science,
Carleton University

Supervisor: Kate Larson
Professor, David R. Cheriton School of Computer Science,
University of Waterloo

Internal Member: Robin Cohen
Professor, David R. Cheriton School of Computer Science,
University of Waterloo

Internal Member: Jesse Hoey
Professor, David R. Cheriton School of Computer Science,
University of Waterloo

Internal-External Member: Mark Crowley
Associate Professor, Dept. of Electrical and
Computer Engineering, University of Waterloo

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

With increasing connectivity between humans and the rise of autonomous agents, group decision-making scenarios are becoming ever more commonplace. Simultaneously, the requirements placed upon decision-making procedures grow increasingly nuanced as social choices are made in more niche settings. To support these demands, a deeper understanding of the behaviour of social choice procedures is needed.

The standard theoretical approach to analyze social choice procedures is limited in the type of question it can answer. Theoretical analyses can be rigid: It may speak to the incompatibility of different properties without also providing a deeper understanding of the properties themselves, or might stop at proving the worst-case outcome of a voting rule without communicating the rule's typical behaviour.

In this dissertation, we address these limitations by demonstrating that experimental analysis of social choice domains can provide an understanding of social choice which is both complementary and additional to theoretical findings. In particular, experimental approaches can form a middle ground between theory and practice: more practical than theoretical approaches in a setting more controlled than real-world application. We apply this approach to a new form of delegative voting and to a task of learning existing and novel voting rules. In each area we find results of a type and scale which are infeasible to traditional analysis.

We first examine an abstract model of delegative voting – agents use *liquid democracy* to transitively delegate their vote – in a setting where the voters collectively agree on a correct outcome. Through extensive simulations we show the dynamic effects on group accuracy from varying a wide range of parameters that collectively encompass many types of human behaviour. We identify two features of this paradigm which result in improvements to group accuracy and highlight a possible explanation for their effectiveness. Subsequently, we apply this liquid democracy framework to the process of training an ensemble of classifiers. We show that the experimental findings from our simulations are largely maintained on a task involving real-world data and result in further improvements when considering a novel metric of the training cost of ensembles.

Additionally, we demonstrate the creation of a robust framework for axiomatic comparison of arbitrary voting rules. Rather than proving whether individual rules satisfy particular axioms, we establish a framework for showing experimentally the degree to which rules general satisfy sets of axioms. This enables a new type of question – degrees of axiom satisfaction – and provides a clear example of how to compare a wide range of single and multi-winner voting rules. Using this framework, we develop a procedure for training a model to act as a novel voting rule. This results in a trained model which realizes a far

lower axiomatic violation rate than most existing rules and demonstrates the possibility for new rules which provide superior axiomatic properties.

Acknowledgements

I would foremost like to thank my supervisor Kate Larson for her wisdom and guidance over many years. Without her support this thesis would not exist. Kate provides a superb role model of how to be a positive influence in academia.

I am blessed to have a mother that is eternally supportive. Thank you for your loving assistance through thick and thin.

I also thank many colleagues for their helpful advice on this and other work. Thank you to my committee members – Zinovi Rabinovich, Robin Cohen, Jesse Hoey, and Mark Crowley. For fruitful discussions and feedback I thank my peers at the university: Carter Blair, Alan Tsang, Nanda Kishore Sreenivas, Valerie Platsko, Marvin Pafla, Dave Radke, and many others. Thank you as well to Shiri Alouf-Heffetz and Nimrod Talmon for the productive collaborations. I look forward to working more with all of you.

I am immensely grateful to all those that made Waterloo a friendly and welcoming city. I particularly thank Nancy and David for opening their home to myself and many others over the past decades. To Sebastian, Matthew, Vinodh, Christian, Kayden, Eugene, Alison, Irene, and others I am grateful for your friendship and support.

Finally—*obviously*—I cannot end without thanking Rodney, Samantha, Jack, Jamie, Nadia, Eliza, Hank, Orson, Cory, and Xander. I would not be who I am today without you.

Dedication

To all those without a dedication page.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Group decision-making procedures are increasingly prevalent in many settings, both among humans and among virtual agents. From political elections and jury decisions [68] to tournament design [207] or resource allocation problems [102], diverse individuals with competing interests must frequently arrive at a single aggregate decision. Every setting has a unique combination of factors which affect the quality of a decision. When humans are making a collective decision, their preferences may form many structures: from a controversial, highly polarizing topic, on which different groups may have opposing opinions, to less charged environments, where voters may exhibit little structure and collectively hold a wide range of perspectives. In agent-based settings, agents have specific goals they for which they optimize, such as identifying some globally best outcome, which they do noisily; a decision procedure must optimally aggregate noisy agent votes to best estimate the correct outcome [191].

Theoretical analysis has been used extensively to study problems such as these. Often analysis will prove that certain intuitively bad outcomes are possible – a voting setting may be prone to manipulation, or voters unlikely to identify some ground truth. Or, analysis may show that good outcomes are impossible – the inability of any voting rule to guarantee desirable properties or computation of an election winner in a reasonable amount of time. Theoretical analysis has been well-suited to answer questions of worst-case behaviours like these but is best suited to answering highly precise questions. These answers often lie with pathological edge cases which do not inform about the average case. As social choice is applied to settings with increasingly specific requirements it becomes increasingly important both to understand the expected empirical behaviour of voting procedures, and to optimize procedures for unique domains.

For many important types of question, experimental analysis allows us to better understand the actual performance of a voting procedure. Experiments give both a general understanding of a system and enable highly specific questions by allowing a focus on specific experimental variables. We demonstrate these benefits and possibilities of experimental analysis across multiple unique applications of a variety of social choice domains. By combining social choice with machine learning and optimization techniques we show the synergy between the two fields.

In particular, this dissertation highlights the ability of experimental analysis to enhance the state-of-the-art in both social choice and machine learning. Within that context, our specific contributions range from demonstrating the benefits of delegative voting for epistemic settings to showing that concepts of delegation from social choice can be applied during training classifier ensembles to developing a framework for measuring axiom violations of voting rules and showing that novel learned rules are able to outperform a wide range of existing rules.

1.1 Thesis Statement

This dissertation studies the following hypothesis:

> *Experimental analysis using tools from the fields of social choice and machine learning reveals novel understanding of these fields which are distinct from and complementary to the findings of theoretical approaches.*

We examine this hypothesis within the bounds of the following research questions, which are explored throughout the remainder of this thesis:

1. **Which population-level factors affect the ability of voters to identify ground truth in a delegative voting setting?**
   (a) For voters with the same ability to identify ground truth, to what degree can changes in voter behaviour affect the aggregate accuracy of a population of voters?
   (b) What structural aspects of a setting affect a group's accuracy?
2. **How can delegative voting be used to improve performance on machine learning classification tasks?**

(a) Can ensembles of classifiers be viewed as voters in a way that improves accuracy on classification tasks?

(b) Does a social choice perspective of machine learning benefit ensemble learning in any dimension beyond accuracy?

3. **Can we learn about voting rules through experimental comparison of their elected alternatives?**

   (a) Do existing voting rules exhibit previously unrecognized similarities?

   (b) Do common classifications of voting rules meaningfully capture patterns in rule behaviour?

   (c) How can we experimentally evaluate the axiomatic properties of voting rules?

   (d) How can we learn novel voting rules with desirable axiomatic properties?

1.2 Contributions

In this thesis, our research contributions include:

1. We conduct a thorough experimental analysis of liquid democracy in an epistemic setting. Our experiments are the first to consider the many factors – preference distribution, network type, delegation strategy, amount of delegation – that can affect the truth-finding ability of voters in a delegative framework. We have shown that the most important factor in improving a group's ability to collectively identify truth is avoid dictatorships. We show two effective novel strategies for achieving this: (1) When voters are less likely to delegate to a voter with significant vote power, voter power remains more uniform and accuracy is increased. (2) Incorporating a decay factor which reduces the weight of each delegation at every hop that it travels. this reduces the chance of developing long delegation chains which contain a large fraction of all existing delegations.

2. We show that the epistemic benefits of liquid democracy frequently remain when the framework is applied to an ensemble learning task. By viewing classifiers as voters we are able to translate many existing results from social choice, including liquid democracy, into a machine learning context. We show that the aforementioned method of using power-sensitive delegation mechanisms to avoid dictatorships continues to perform very well.

3. We define a metric of measuring the cost to train a classifier and identify a novel benefit of applying liquid democracy to an ensemble pruning task: dramatic reduction in training cost as compared to naive ensembles.

4. We establish a robust framework for experimental, data-driven comparison of voting rules which allows us to compare the axiomatic violations of single or multi-winner rules for a large class of axioms. We use this framework to identify unexpected similarities between individual voting rules and to measure the cohesiveness of standard classes of voting rules.

5. We identify best practices for training a model to act as a voting rule and use novel learned voting rules to demonstrate that existing rules can be significantly outperformed by new rules on our measure of axiomatic violations.

1.3 Thesis Overview

This thesis is presented across three primary research chapters. In addition to this introduction, this document contains five additional chapters. We briefly describe the contents of each chapter, and note which publication(s) the chapter corresponds with.

- In Chapter 2, we introduce the background necessary for all subsequent chapters. We focus primarily on topics which have some bearing on multiple subsequent chapters. Each subsequent research chapter relies on several topics discussed here, which are briefly reviewed when relevant.

- In Chapter 3, we describe our work experimentally exploring the use of liquid democracy for epistemic voting. We evaluate a number of delegation mechanisms, and identify which factors are most influential to the change in group accuracy under a liquid democracy setting.

- In Chapter 4, we extend our work on liquid democracy to a novel application in ensemble pruning. We demonstrate the benefit of delegation for ensemble pruning and determine in which settings liquid democracy has the most positive impact on ensemble accuracy and training cost.

- In Chapter 5, we train models to replicate existing voting rules and to be novel voting rules which provide desirable axiomatic properties. We identify new connections between classes of voting rule and show that many voting rules perform poorly when evaluated across a range of desirable axioms.

- In Chapter 6, we summarize the main findings of our dissertation and outline a number of avenues to consider in future work.

Much of the work contained in this thesis has appeared through some parts of the following papers:

- (Chapter 3) Shiri Alouf-Heffetz, Ben Armstrong, Kate Larson, and Nimrod Talmon. "How Should We Vote? A Comparison of Voting Systems within Social Networks." International Joint Conference on Artificial Intelligence. 2022.[6]
- (Chapter 3) Ben Armstrong, Shiri Alouf-Heffetz, and Nimrod Talmon. "Optimizing Viscous Democracy." International Joint Conference on Artificial Intelligence. 2024.[9]
- (Chapter 4) Ben Armstrong, and Kate Larson. "On the Limited Applicability of Liquid Democracy." Appears at the 3rd Games, Agents, and Incentives Workshop (GAIW 2021). Held as part of AAMAS 2021. 2021. [12]
- (Chapter 4) Armstrong, Ben, and Kate Larson. "Liquid Democracy for Low-Cost Ensemble Pruning." (Extended Abstract) In Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems. 2024. [10]
- (Chapter 5) Ben Armstrong, and Kate Larson. "Machine Learning to Strengthen Democracy." NeurIPS Joint Workshop on AI for Social Good. Held as part of the 33rd Conference on Neural Information Processing Systems. 2019. [11]

1.4 Application of Thesis

At the outset we wish to clarify how the work we present in this thesis should be interpreted. The field of social choice research uses many politically-loaded terms such as "elections" or "democracy." Many aspects of social choice research *can* relate to ideas of political philosophy and might form the basis of our current and future democratic systems. In this thesis, we *do not* explore such applications. We emphasize that our work is focused on exploring connections between social choice and empirical methods. Our work teaches us about several existing forms of voting but we do not suggest that our results be directly applied to human democracies.

Chapter 2

Background and Related Work

Our work operates at the intersection of social choice and machine learning. In this chapter we introduce the concepts from each of these fields that are most important for understanding and contextualizing our research.

2.1 Social Choice

Social choice is a field that broadly studies situations in which multiple agents must make joint decisions. Our focus is on electoral settings where agents vote to select some winner from a field of alternatives. However, social choice also studies topics such as fair division and matching [39].

Within the voting branch of social choice there are a number of distinct avenues of research, including topics such as:

- **Voting to identify truth, or "epistemic" voting.** When there is an objectively correct outcome, groups of voters are often more effective at guessing the outcome than any individual within the group. Condorcet began the ongoing study of *jury theorems* showing under what conditions on individual voter accuracy and independence of voters do groups of voters improve accuracy [161]. Condorcet's original work also addresses the task of maximum likelihood estimation; estimating the most likely true state for voters sampled from a known distribution.

- **Impossibility theorems and axiomatic analysis of voting rules.** Axioms describe basic properties a rule might satify; Arrow's famed impossibility theorem [13] started a line of research into impossibility theorems: each showing that a specific sets of axioms cannot all be guaranteed by *any* voting rule. In other cases, a voting rule may be shown to be the unique rule satisfying all axioms in a given set [153]. More recently, calls for data-driven axiomatic analysis suggest that experimental frameworks may enable answering analytical questions which are different from traditional theoretical approaches [67].

- **Structure of voter preferences and information given to voting rules.** Voting rules may use ordinal or cardinal preferences. When voter preferences fit certain structural requirements (e.g. being "single-peaked") theoretical results, such as axiomatic satisfaction, can change [75]. Many rules can be categorized by the data that is required to fully determine the winner of the rule.

- **Computational requirements of voting rules.** For some methods of voting such as the Kemeny rule, simply determining a winner is NP-hard [24]. Other research studies the complexity of a voter manipulating the outcome by misrepresenting their preferences or abstaining from an election [217].

Social choice uses the *preferences* of individuals as input to some function which forms a group decision. These preferences can be in an absolute score-based form, i.e. a voter assigns some intensity of preference to each alternative independent of other alternatives. They may also be relative, i.e. a voter ranks each alternative only on whether the voter prefers them more or less than other alternatives. In this work we focus primarily on ordinal preferences, equivalently called *ranked ballots*. We say that each voter has some *preference order* over alternatives which describes their relative preference for each alternative to win some election.

More formally, given voters $V = \{v_1, v_2, ..., v_n\}$ and m alternatives, denoted A, each voter v_i has a preference order P_i which is a total ordering over A. We write this as,

$$P_i : a \succ b \succ c \succ ...$$

This can be read as "v_i prefers a over b, which is preferred over c, which is preferred over ..." To indicate the specific rank in which an alternative is placed, we index P_i: $P_i[1] = a, P_i[2] = b, ...$ or, generally, $P_i[j] = c$ indicates that v_i ranks c in position j. A collection $\mathcal{P}$ of preference orders of all n voters is referred to as a *preference profile*. Figure 2.1 shows an example with the preferences of 4 voters. Let $\#^{\mathcal{P}}_{i>j}$ denote the number of voters in $\mathcal{P}$ that prefer a_i to a_j and $\#^{\mathcal{P}}_{i=r}$ denote the number of voters that rank a_i as their r-th favourite alternative.

$$v_1 : a \succ b \succ c \succ d \qquad v_3 : b \succ a \succ c \succ d$$
$$v_2 : c \succ d \succ a \succ b \qquad v_4 : b \succ a \succ c \succ d$$

Figure 2.1: Preference profile from four voters over four alternatives.

We consider only the setting where voters have strict preferences; however, another common setting also allows for weak preferences, writing $a \succcurlyeq b$ to indicate that "a is weakly preferred to b"; either a is preferred to b or a and b are ranked equally. Using these concepts, we can define a social choice function which is a function mapping a preference profile to a non-empty subset of alternatives.

Definition 2.1.1. *A Social Choice Function (SCF) $\mathcal{F} : \mathcal{P} \to 2^A \setminus \emptyset$ is a function mapping a preference profile $\mathcal{P}$ from n voters to a non-empty subset of alternatives A.*

In common parlance, elections often result in a single winner yet our definition of a social choice function allows for multiple winners. This is due, in part, to tie-breaking methods. A function may consider multiple alternatives as equally valid winners. As a single winner is typically required, tie-breaking becomes necessary. Two common approaches to tie-breaking are referred to as *lexicographic* and *random*: lexicographic tie-breaking assumes there is some underlying ordering of alternatives and selects as winner the tied alternative which comes first in that order. Random tie-breaking selects a tied alternative uniformly at random.

A Social Choice Function is *resolute* if it always returns exactly one winning alternative. Any non-resolute SCF can be made resolute by combining it with a tie-breaking function. We define a Voting Rule as this combination.

Definition 2.1.2. *A Voting Rule is a resolute Social Choice Function $\mathcal{F} : \mathcal{P} \to A$ mapping a set of preference orders $\mathcal{P}$ to a single winning alternative drawn from alternatives A.*

Voting rules are often categorized, based on similarities in their structure and the information needed to compute their winners. We now introduce popular categorizations of voting rules and describe specific examples of rules falling into each class of interest.

2.1.1 Positional Scoring Rules

Many voting rules fall into existing categories such as positional scoring rules, first described by Young [233], which include several common rules.

Definition 2.1.3. *A **positional scoring rule** is defined by a vector* $\mathbf{s}$ *of length* m*. Each voter awards their* i^{th} *favourite alternative a number of points equal to* s_i*. That is, each alternative* $a \in A$ *receives a score of*

$$sc(a, \mathcal{P}) = \sum_{1 \leq i \leq m} \#^{\mathcal{P}}_{a=i} \cdot s_i$$

The alternative receiving the most points is the winner of the election. Some well-known scoring rules and their corresponding vectors are,

***Plurality**:*	$\boldsymbol{s} = (1, 0, 0, \ldots, 0)$
***Anti-Plurality**:*	$\boldsymbol{s} = (0, 0, \ldots, 0, -1)$
***Borda**:*	$\boldsymbol{s} = (m-1, m-2, \ldots, 0)$

In these rules, each voter gives one point to their favourite alternative (plurality), a point against the least favourite alternative (anti-plurality), and points decreasing linearly with each voters ranking (Borda).

While an infinite number of positional scoring rules can be easily imagined, only a small number are ever specifically discussed. However, those that are named, and the class itself, receive a significant amount of attention [39, 8, 145]. Each positional scoring rule can be computed using only the score vector and the $m \times m$ **Rank Matrix** indicating the number of voters that rank each alternative at each position.

$$\mathcal{M}^{\text{ranked}}_{ij} = \#^{\mathcal{P}}_{i=j} \;\forall\; 1 \leq i, j \leq m$$

2.1.2 Fishburn's Classification

Not all rules are positional scoring rules. Fishburn defined three classes of voting rule which are distinct from positional scoring rules [89]. Each class can be loosely thought of as requiring a different amount of information to determine winners.

C1 Rules C1 rules are also referred to as *tournaments*; they compute winners solely based on whether a majority of voters prefer one alternative over another, for all pairs of alternatives. The necessary information to compute a C1 function is encoded in the **Majority Matrix**, created by transforming preference profile $\mathcal{P}$ into an $m \times m$ matrix $\mathcal{M}^{\text{majority}}$ indicating which alternatives win a majority of pairwise comparisons. That is,

$$\mathcal{M}_{ij}^{\text{majority}} = \begin{cases} 1 & \#_{i>j}^{\mathcal{P}} \geq \lceil \frac{n}{2} \rceil \\ 0 & \text{otherwise} \end{cases}$$

Copeland's rule is a C1 function; the score of each alternative under Copeland can be determined purely be considering the number of pairwise comparisons the alternative wins.

> **Definition 2.1.4.** ***Copeland*** *The score of each alternative a_i is equal to the number of alternatives it is preferred to by a majority of voters, subtract the number of alternatives preferred by a majority over a_i. That is,*
>
> $$Copeland(a_i) = |\{a_j \mid \#_{i>j}^{\mathcal{P}} > \frac{n}{2}\}| - |\{a_j \mid \#_{j>i}^{\mathcal{P}} > \frac{n}{2}\}|$$
>
> *The highest scoring alternative(s) win.*

C2 Rules These are also referred to as *weighted tournaments*. Winners depend not only on the number of pairwise competitions each alternative wins but also the margin of victory. The necessary information to compute a C2 function is encoded in the **Weighted Majority Matrix**, an $m \times m$ matrix $\mathcal{M}^{\text{weighted}}$ indicating the number of pairwise comparisons each alternative wins. That is,

$$\mathcal{M}_{ij}^{\text{weighted}} = \#_{i>j}^{\mathcal{P}} \; \forall \; 1 \leq i, j \leq m.$$

Borda's rule, while also a positional scoring rule, is a C2 rule. Analysis reveals that the score of each alternative a depends purely on the number of voters preferring a_i over other alternatives [39]. Brandt et al. show that

$$sc_{\text{Borda}}(a_i, \mathcal{P}) = \sum_{a_j \in A} \frac{\#_{a_i > a_j}^{\mathcal{P}} - \#_{a_j > a_i}^{\mathcal{P}} + n}{2}$$

The score of each alternative under Borda's rule can be computed using their margin of victory over other alternatives, making Borda a C2 rule.

C3 Rules Fishburn describes C3 rules as simply the rules which do not fit into either of C1 or C2. In some cases, C3 rules require more information than C1 or C2 rules, just as C2 rules require strictly more information than C1 rules. Plurality and Anti-Plurality are C3 rules and (in some sense) seems to use strictly less information than Borda's rule. Both the majority and weighted majority matrices are constructed using all pairwise comparisons whereas Plurality can be thought of as relying on pairwise comparisons between each voter's first preference only, information that is not preserved in either matrix.

We now step through an example showing, for each rule we have discussed, how winner determination occurs. Despite all sounding reasonable, each of these rules can return different results.

Example 2.1.1. *Consider four voters choosing between alternatives $A = \{a, b, c, d\}$ with preferences as shown in Figure 2.1. We show the calculation of winners for each of the social choice functions we have defined above.*

Plurality*: Each voter gives one point to their favourite alternative; v_1 gives a point to a, v_2 gives a point to c, v_3 and v_4 both give a point to b. Written as a vector of points given to (a, b, c, d), plurality results in scores of $(1, 2, 1, 0)$ with b as the unique winner.*

Anti-Plurality: *Each voter gives a negative point to their least favourite alternative. This results in a results score vector of $(0, -1, 0, -3)$ with $\{a, c\}$ as tied winners.*

Borda: *Voters give $m - 1$ points to their favourite alternative, decreasing linearly to give 0 points to their least favourite. v_1 gives 3 points to a, 2 points to b, 1 point to c, and 0 points to d. Considering all voters gives a results score vector of $(8, 8, 6, 2)$ with $\{a, b\}$ as tied winners.*

Copeland: *We count the number of pairwise contests each alternative wins and loses. a is preferred to b by 2 voters (no majority winning or losing); a is preferred to c by 3 voters (a majority prefer a over b); a is preferred to d by 4 voters (a majority prefer a over d) resulting in $Copeland(a) = 2 - 0 = 2$. The Copeland score of each alternative is $(2, 2, 1, 0)$ and $\{a, b\}$ are tied winners.*

The winners of Plurality, Borda, Anti-Plurality, and Copeland given the preferences in Figure 2.1 are below.

$$
\begin{array}{lcl}
\mathcal{F}^{\text{Plurality}}(\mathcal{P}) & = & \{b\} \\
\mathcal{F}^{\text{Borda}}(\mathcal{P}) & = & \{a,b\} \\
\mathcal{F}^{\text{Copeland}}(\mathcal{P}) & = & \{a,b\} \\
\mathcal{F}^{\text{Anti-Plurality}}(\mathcal{P}) & = & \{a,c\}
\end{array}
$$

If lexicographic tie-breaking based on the alphabetical ordering of alternatives were to be applied to turn each SCF into a Voting Rule, $\mathcal{F}^{\text{Plurality}}$ would elect b while all other rules would elect a.

2.1.3 Multi-winner Voting

Thus far we have discussed voting when only one alternative is the winner. Many discussions around voting carry with them this implicit assumption of a single winner. This is not always the case. In many settings – electing a parliament, finding multiple recommendations for a group, or identifying several "best" search results [138] – voters are tasked with electing multiple alternatives where alternatives may be political candidates, menu choices for a group meal, or websites as possible search results. Multi-winner voter rules largely fall into three informal categories based on their intended goal(s) [86]:

- **Individual Excellence**: electing alternatives which are popular as individuals without considering which other alternatives are elected.
- **Diversity**: Electing alternatives which are preferred by a wide range of voters. i.e. ensuring that, for as many voters as possible, there is some winning alternative that they consider desirable.
- **Proportional**: Electing alternatives which, as a set, represent blocs of similar voters in proportion to the size of the blocs.

A multi-winner voting rule $\mathcal{F}(\mathcal{P}, k) \in \{C | C \subset A,\ |C| = k\}$ elects a set of exactly k alternatives with $1 \leq k < m$. Multi-winner voting therefore serves as a generalization of the standard single winner setting where $k = 1$. In the literature, multi-winner rules are often synonymously called committee rules [138]. The term *committee* highlights the focus a rule may have on electing a set of winners which are collectively desirable rather than a focus on electing individuals that are each popular.

Here we discuss several different types of multi-winner voting rule. We include some illustrative examples but defer definitions of most of the multi-winner rules that we use to Chapter 5.

The first type of multi-winner rule we discuss is based on voter preference rankings, as are all single winner rules we discuss. Within this class, are extensions of positional scoring rules. Any positional scoring rule can trivially be converted to a multi-winner rule by simply selecting the k highest scoring alternatives.

k-Borda: An extension of Borda to multiple winners. Each voter assigns $m-1$ points to their top ranked alternative, $m-2$ points to their second ranked alternative etc. $\mathcal{F}^{\text{Borda}}$ returns the k alternatives with highest scores.

We also consider several rules that use a restricted view of voter preferences, the class of approval-based rules. Approval-based multi-winner voting rules make use of a voters' *approval ballot* $App(v_i)$ which contains their k highest ranked alternatives[1]. The set of all approval ballots is $\mathcal{P}_{App} = (App(v_1), \ldots, App(v_n))$.

Definition 2.1.5. *Approval-Based Committee (ABC) voting rules are those rules using as input only the approval ballot of each voter. An ABC rule is a function $\mathcal{F}(\mathcal{P}_{App}, k)$ which returns a set of k alternatives.*

Monroe: Considers all ways of assigning each voter to one alternative in committee W, such that every $a \in W$ is assigned to between $\lfloor \frac{n}{k} \rfloor$ and $\lceil \frac{n}{k} \rceil$ voters. The score of W is the number of voters assigned to an alternative that they approve. $\mathcal{F}^{\text{Monroe}}$ selects the committee with the highest score.

Approval-based rules can be further subdivided by whether or not they are Thiele rules. First described in 1895 by Torvald Thiele [212, 125], Thiele rules are characterized by some satisfaction function w which they aim to maximize. Thiele rules can consider ranked ballots but are more frequently used in the approval domain (to which we restrict our attention).

Definition 2.1.6. *A Thiele rule $\mathcal{F}$ is characterized by a satisfaction function $w : \mathbb{N} \to \mathbb{R}$ which scores sets of alternatives based on the number of alternatives in the set which each voter approves. w is weakly monotonically increasing and has $w(0) = 0$. $\mathcal{F}$ elects a set of winners which maximizes the score function [138]:*

$$score_w(\mathcal{P}_{App}, C) = \sum_{v_i \in V} w(|App(v_i) \cap C|)$$

[1]We restrict our attention to the setting where voters approve of exactly their k favourite alternatives as a simplifying assumption when generating preference data; however, our model naturally accommodates the more general setting where each voter can approve differing numbers of alternatives.

Proportional Approval Voting (PAV): Given committee C, define the PAV score of the committee as

$$sc_{\text{PAV}}(C) = \sum_{v_i \in V} \sum_{j=1}^{|C \cap App(v_i)|} \frac{1}{j}.$$

$\mathcal{F}^{\text{PAV}}$ returns $C^* = \arg\max sc_{\text{PAV}}(C)$. This rule aims to select alternatives such that each voter approves of a similar number of elected alternatives. PAV is the original rule described by Thiele [138]; the increased satisfaction each voter receives for an additional winning alternative of which they approve follows the sequence of harmonic numbers: $1, \frac{1}{2}, \frac{1}{3}, ..., \frac{1}{x},$. PAV is the Thiele method with $w(x) = \sum_{j=1}^{x} \frac{1}{j}$.

We use multi-winner voting rules exclusively in Chapter 5. Except where explicitly discussing the multi-winner setting we assume that voting rules elect a single winner.

2.1.4 Axiomatic Social Choice

The axiomatic approach to social choice focuses on identifying which voting rules satisfy – or are uniquely described by – certain sets of properties known as axioms. In contrast to the epistemic approach, which focuses on which voting rule is most likely to identify the correct alternative, the axiomatic approach might recommend one rule over another based on the properties that it satisfies. We provide only a brief overview of this topic as our work only lightly touches upon axiomatic issues.

As an example, a well-known axiom, the *Condorcet criteria*, relies on the concept of a Condorcet winner – a concept around which many voting rules are designed.

Definition 2.1.7. *A* ***Condorcet winner*** *is an alternative which is preferred by a majority of voters to all other alternatives when compared in pairwise competitions.*

A voting rule $\mathcal{F}$ is "Condorcet consistent" (i.e. it satisfies the Condorcet criteria) if, whenever the input preference profile $\mathcal{P}$ contains a Condorcet winner c, then $\mathcal{F}(\mathcal{P}) = c$ Some rules, such as Copeland or the Top Cycle, are known to satisfy the Condorcet criteria while other rules such as Plurality and Borda do not satisfy it[39]. Certain axioms, such as the Condorcet criteria, are often touted as intuitively "good." Proponents of the axiomatic approach argue that a voting rule should satisfy their preferred axioms in order to be suitable for use in an election.

Research in this area typically focuses on impossibilities or necessities. Consider the following three axioms:

1. **Unanimity** (also called Pareto Efficiency): If all voters prefer alternative a over alternative b then the output must also rank a over b.

2. **Non-dictatorship**: There is no voter v such that, no matter the preferences of other voters, the output ranking always matches exactly the ranking of v.

3. **Independence of Irrelevant Alternatives** (IIA): Across two elections over the same voters (with possibly different preferences) and alternatives, if each voter maintains the same ranking of a and b across both elections then both output rankings must rank a and b in the same relative order.

Arrow's Impossibility Theorem – a major contributor to Kenneth Arrow's receipt of the 1972 Nobel Prize – connects these axioms and began a line of inquiry which continues to this day. Note here that a social welfare function is a function mapping a preference profile to a complete ranking over alternatives.

Theorem 2.1.1 (Arrow's Impossibility Theorem)**.** *No social welfare function can exist which satisfies Unanimity, Non-dictatorship, and Independence of Irrelevant Alternatives.*

Arrow's theorem applies to only a specific setting with ranked preferences. Gibbard's proof of what is now typically called the Gibbard-Satterthwaite theorem develops a similar impossibility theorem which applies to cardinal preferences [101]. Stated informally, Gibbard's theorem states that there is no deterministic function mapping voter preferences to a winning alternative which meets all of the following criteria:

1. There are at least 3 alternatives.

2. The function is non-dictatorial (as defined above).

3. The function is strategy-proof; no voter will be happier with the outcome if they misrepresent their preferences.

Beyond impossibility theorems the other common application of axiomatic analysis is to *characterize* a rule. We say that a rule is characterized by a set of axioms if the axioms are mutually satisfied by only that rule. Many, though not all, rules are characterized by some known set of axioms [154, 93].

2.1.5 Epistemic Voting

Epistemic voting refers to the act of voting in order to uncover some underlying, objectively-correct truth. This truth can come in two forms: knowable and unknowable. Voters may answer questions such as "Which policy decision will maximize GDP growth over the next 6 months?" or "Is the defendant in this trial guilty?" In these cases, there is a single correct option but it cannot be truly known. On the other hand, voters may also face questions that have a verifiably correct answer: "Is this a picture of a dog?" or "How many jelly beans are in this jar?"

Available records suggest that epistemic voting was first described by Ramon Llull in the late 13^{th} century with the concept experiencing a "modern" revival via Marie Jean Antoine Nicolas de Caritat, Marquis of Condorcet (typically referred to simply by his surname, *Condorcet*) in 1785 [87, 161]. Llull and Condorcet independently invented a method of determining the winner of an election based on pairwise comparisons. The motivation of the method is to identify some underlying *truth*. Llull describes this as electing "God's candidate" – based on the idea that human preferences weakly reflect the preferences of God and that the best aggregation method should maximize the probability of reconstructing God's preferences [87].

Much work on epistemic voting stems from Condorcet's "Jury Theorem" in which jurors collectively decide whether a defendant is innocent or guilty. Condorcet imaged jurors as voters voting on the defendants guilt and showed that larger juries are more accurate.

Theorem 2.1.2 (Condorcet Jury Theorem)**.** *If n voters make decisions independently and all have identical competence $q > \frac{1}{2}$ which is their probability of selecting the correct outcome, then the probability that the group selects the correct outcome increases monotonically with n. As n approaches ∞, the probability that the group is correct approaches* 1*.*

A vast body of literature has expanded upon this simple theorem in several directions, considering aspects such as the group size, independence of voters, and the uniformity of voter competences [113]. Two important pieces of research inspired by Condorcet relax the uniformity requirement and describe changing the weight given to each voter's vote: Kazmann showed that the jury theorem continues to hold if voters each have a different competence, so long as every voter's competency is greater than half [128]. Several independent research groups have shown that group accuracy is maximized if each voter has a weight proportional to a function of their competence. That is, the input of voter v_i should have weight proportional to $\log(\frac{q_i}{1-q_i})$ [199, 164, 151, 72].

Epistemic voting can be seen more broadly as a question of maximum likelihood estimation, a branch of machine learning which we discuss further in Section 2.4.2. When voter preferences noisily reflect an epistemically correct ranking, and the noise model is known, then maximum likelihood estimation identifies which voting rule is *maximally likely* to return the ground truth outcome.

In our work we use ordinal notation to encompass the epistemic setting. When voters must choose between two alternatives $A = \{a^+, a^-\}$ with a^+ representing the correct outcome and a^- a bad outcome, we can say that the preference order of $a^+ \succ a^-$ is the epistemically correct order while $a^- \succ a^+$ is the incorrect order. If each v_i has some competence level q_i then v_i has probability q_i of having preference order $P_i : a^+ \succ a^-$.

2.2 Voter Preference Distributions

When using a voting rule to aggregate preferences, it is often convenient to consider particular structures of preferences that voters may have, which we refer to as preference *distributions*. Preference distributions typically aim to capture some aspect of human behaviour or provide a convenient framework for demonstrating mathematical results [186].

As a motivating example, Elkind provides two settings in which a group decision must be made [75]: In the first example, a family of three must decide between three activities and have preferences as follows:

$$\begin{aligned}
v_1 &: a^{\text{bike}} \succ a^{\text{shop}} \succ a^{\text{swim}} \\
v_2 &: a^{\text{shop}} \succ a^{\text{swim}} \succ a^{\text{bike}} \\
v_3 &: a^{\text{swim}} \succ a^{\text{bike}} \succ a^{\text{shop}}
\end{aligned}$$

No matter the activity chosen, two of the three family members prefer one of the other options. This situation might be thought of as corresponding to a uniform distribution of preferences over the alternatives. On the other hand, we can consider the same family deciding at what temperature to set their air conditioner. If each family member prefers a different temperature, we can reasonably assume that everybody prefers a temperature strictly closer to their ideal temperature more than a temperature further from their ideal. This might lead to picking the mean or median of all preferences.

This assumption describes a type of preference referred to as *single-peaked* [29]. Informally, preferences are single-peaked if there is some shared ordering of alternatives and

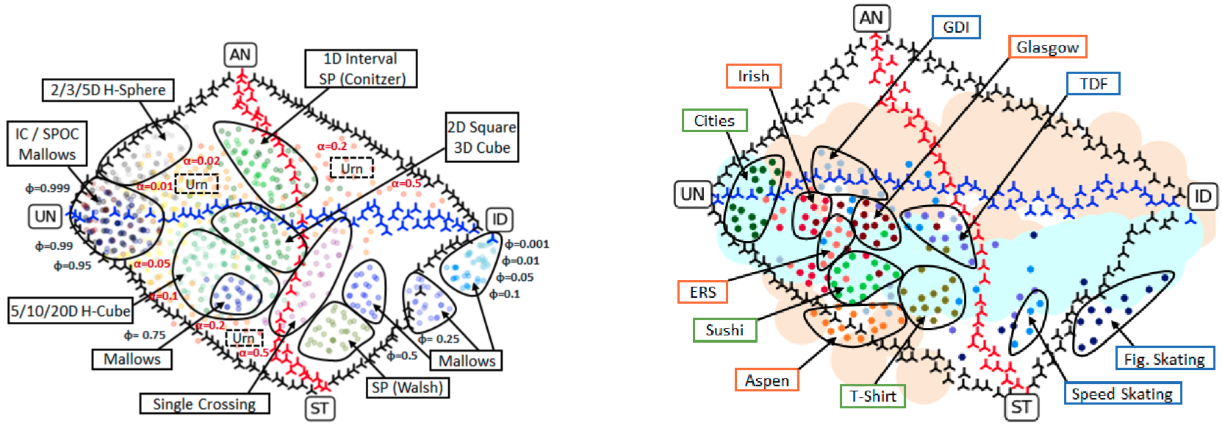


Figure 2.2: A map of elections generated by artificial preference distributions (left) and empirical election data (right). Each dot represents the preferences from a single election. Relative positioning of each dot shows similarity between each the preferences in each election. See Boehmer et al. for the source of the image and a full explanation [35].

voters always prefer alternatives that are closer in the ordering to their favourite more than alternatives further from their favourite. Single-peaked preferences are a very strong restriction on general preference orders and are not suitable for modelling many domains. When alternatives can be thought of as existing in a single dimension, such as temperature, preferences are naturally described as single-peaked.

Different distributions are useful to social choice research in two ways: First, theoretical results and properties change when considering a restricted preference domain. For example, a Condorcet winner does not exist generally; however, there is always a Condorcet winner (or tied "weak" Condorcet winners) under single-peaked preferences.

Second, from an empirical perspective, different preference distributions can be used to model different applied settings where real-world data is limited. Boehmer et al. have developed a "map" of elections which provides a visual intuition as to the similarities between several types of real-world election data and many preference distributions which we reprint in Figure 2.2. The left side shows the similarity between artificial preference distributions, developed in earlier work by Szufa et al. [208], while the right side shows the location of real-world elections mapped in the same underlying space [35].

In our work we explore the following distributions:

Impartial Culture (IC) Each voter is equally likely to have any one of the $m!$ preference orders, regardless of which orders any other voters have selected. IC is a statistical distribution which does not reflect human preferences but upon which mathematical analysis is tractable [114].

Impartial Anonymous Culture (IAC) Preferences profiles are generated collectively, rather than as individual preference orders. Each multi-set of preference orders (i.e. a preference profile) is equally likely to be generated, making voter identities irrelevant. This is in contrast to IC where each individual preference order is equally likely to occur. For example, the profile $\{v_1 : a_1 \succ a_2, v_2 : a_2 \succ a_1, v_3 : a_1 \succ a_2,\}$ is considered identical to the profile $\{v_1 : a_1 \succ a_2, v_2 : a_1 \succ a_2, v_3 : a_2 \succ a_1,\}$ as the same preference orders occur, simply assigned to different voters. IAC provides mathematical convenience rather than aiming to model human preferences [98].

Identity There is some reference ranking r and all voters have exactly r as their preference order.

Mallows Preference orders are noisy copies of a reference ranking r, with noise based on a parameter $\phi \in (0, 1)$. The probability of sampling a preference order r' is equal to

$$\frac{1}{Z(\phi, m)}\phi^{\kappa(r,r')}$$

Where $\kappa(r, r')$ is the number of pairwise swaps between r and r', a.k.a. the Kendall tau distance, and $Z(\phi, m) = (1 + \phi) \cdot (1 + \phi + \phi^2)...(1 + \phi + ... + \phi^{m-1})$ is a normalization constant. $\phi = 1$ results in Impartial Culture while $\phi = 0$ gives the Identity distribution – preferences identical to the reference ranking. Mallow's models aim to roughly capture human preferences [147].

In our experiments we sample ϕ uniformly at random and use the re-parameterized version of the Mallow's model as described by Boehmer et al. which normalizes ϕ such that the distance from the reference ranking increases more directly with ϕ [35].

Urn Parameterized by α. All $m!$ preference orders exist in an "urn." Voters select orders consecutively. After a preference order is selected, it is returned to the urn along with α copies of the ranking. Urn models provide a convenient framework for modelling many tasks but were not originally designed around human preferences [74, 159].

In our experiments we use sample α from a Gamma distribution with shape parameter $k = 0.8$ and scale parameter $\theta = 1$ as described by Boehmer et al. [35].

Single-Peaked Some global ordering of alternatives exists, i.e. alternatives can be placed on some one-dimensional axis. Each voter has a favourite alternative and prefers alternatives closer to their favourite over alternatives further from their favourite. This model naturally captures empirical preferences in very limited settings. We explore two versions of single-peaked preferences, named for their designers: Conitzer [61] and Walsh [220].

The Conitzer Single Peaked distribution generates preference orders by selecting a random position and adding alternatives to the left and right of that position, selecting each direction with uniform probability.

The Walsh Singled Peaked distribution instead uses a recursive procedure, selecting an alternative from either end of the global ordering with equal probability and appending it to the single peaked distribution generated from the remaining alternatives.

Euclidean Voters and alternatives are placed randomly within a topological space and preferences correspond to the distance from each voter to each alternative. Euclidean models aim to produce preferences similar to empirically observed preferences [79]. There are several parameters defining the distribution:

- Shape: The topology of the space in which voters and alternatives are placed. We use only Ball or Cube spaces. Each space has a width of 1.
- Dimension: The dimension of the space being considered. e.g. A Ball topology with dimension 2 is a circle, dimension 3 is a sphere, etc.
- Randomness: Uniform or Gaussian. A voter or alternative is placed at a position in the chosen space according to the chosen random distribution. Gaussian distributions are centered in the middle of the space and have standard deviation of 0.33.

Stratification Alternatives are split into two classes with the size of the first class proportional to a weight parameter $w \in (0, 1)$. All voters rank all alternatives in the first class above those in the second class and rank alternatives uniformly at random within a class. Stratified preferences have been designed to provide a convenient "extreme" distribution to compare with other distributions [35].

In our experiments we use $w = \lfloor \frac{m}{2} \rfloor$ in all cases.

Single-Crossing A single-crossing preference profile is one in which voters can be arranged one-dimensionally on a line and for every pair of alternatives a, b there is some point on the line where all voters left of the point have $a \succ b$ and all voters to the right have $b \succ a$. Single-crossing preferences were originally created by economists to model human preferences about income taxation [156, 76].

Group-Separable A preference profile is group-separable if, for every set of alternatives A' there is some partition of $A' = A'_1 \cup A'_2$ such that every voter either prefers all alternatives in A'_1 over all alternatives in A'_2, or vice-versa. Group-separable preferences were originally developed as a distribution under which a ranking produced by the majority rule satisfied transitivity, not to reflect human preferences [124].

2.3 Liquid Democracy

Liquid democracy (LD) is a framework for delegative voting which has received much attention from social choice researchers over the past 15 years [173]. Liquid democracy operates independently from and is used in conjunction with an existing voting rule. Under a traditional delegative voting scheme, a voter might delegate their vote to another voter who is then required to vote on behalf of themselves and the voter who delegated [58]. Liquid democracy extends this to a transitive setting: If a voter receives a delegation they may also delegate their vote and all of their received delegations. As a result, a voter's delegation may ultimately be transferred several times before being cast as a vote.

In the basic LD setting, each $v_i \in V$ begins with a *weight* $w_i = 1$ and takes exactly one of two actions:

1. v_i delegates to some $v_j \in V$ and w_j is increased by w_i.
2. v_i participates directly in the voting system with an impact equal to w_i (intuitively, a weight of w_i can usually be thought of as equivalent to w_i unweighted clones of v_i).

A delegation function $d : V \rightarrow V$ captures this choice for each voter by mapping voters to their delegates. $d(v_i) = v_j$ indicates that v_i delegates to v_j. If a voter participates directly, we say that they delegate to themselves. We refer to these voters as *representatives* (much literature equivalently uses the term "guru" for these voters). The key distinction between liquid democracy and other forms of delegative voting is transitive delegation. That is, $d(v_i)$ may not refer to the voter that represents v_i but may, instead, delegate their vote and v_i's delegation. We identify the representative of v_i through repeated application of the delegation function until a self-delegation is reached, denoted $d^*(v_i) = d(d(d(...))) = v_j$.

Delegations can be thought of as "flowing" from one voter to another (hence the term *liquid*). To do this, we say that voters exist as nodes on some underlying undirected graph $G = (V, E)$. If G is not otherwise defined, we say that it is the complete graph. We model delegations as flows on a directed graph $D = (V, \{(v_i, v_j) \in E \mid d(v_i) = v_j\})$ induced by the delegation function; each voter is a node and an edge from v_i to v_j exists if, and only if, $d(v_i) = v_j$. The length of the shortest path from v_i to v_j is $\texttt{dist}_D(v_i, v_j)$. If there is no path from v_i to v_j then $\texttt{dist}_D(v_i, v_j) = \infty$ and $\texttt{dist}_D(v_i, v_i) = 0$. Throughout this work we disallow any delegation that would cause a cycle in D.

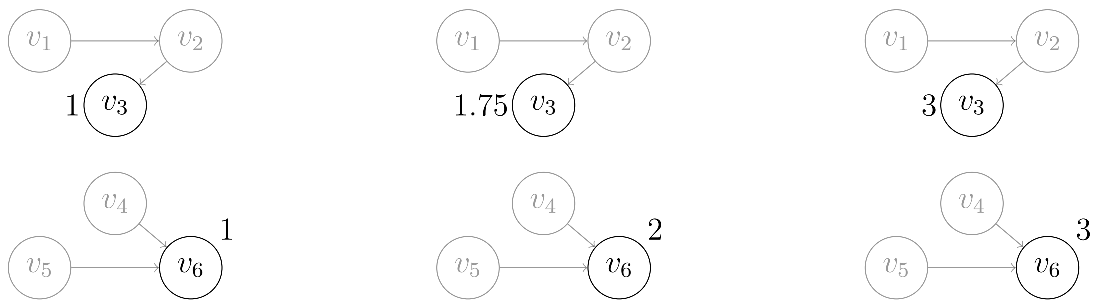
Our work highlights a particular extension of liquid democracy which we introduce here. **Viscous democracy** (VD) is a generalization of LD that reduces the amount of weight transferred through delegation based on the distance the delegation travels. This was originally introduced by Boldi et al. as a method of accounting for the fact that delegations in LD are typically to trusted friends and, as a delegation travels further from its source, that trust diminishes [37]. VD uses a parameter $0 \le \alpha \le 1$ called viscosity. Each time that a delegation travels across an edge in the D its weight is scaled by α. As a result, the weight of v_i is:

$$w_i = \begin{cases} 0 & \text{if } d^*(v_i) \neq v_i \\ \sum_{\{v_j \in V \mid d^*(v_j) = v_i\}} \alpha^{\texttt{dist}_D(v_j, v_i)} & \text{otherwise} \end{cases}$$

There are two special cases of viscosity that merit special attention: When $\alpha = 1$, the model reduces to the standard LD model; viscosity does not degrade the weight of a delegated vote as it travels. When $\alpha = 0$, delegations no longer transfer any weight and each representative votes with a weight of 1. Figure 2.3 shows how representative weight changes under three levels of viscosity.

While originally conceived of as a model for the decrease in trust across distances, viscous democracy is also useful for managing weight. When many voters delegate, it is common for a few representatives to receive a disproportionate number of delegations. This can result in one or a few representatives effectively controlling the outcome of the election. With a low α these representatives receive a significantly lower increase to their weight when receiving delegations from afar; however, representatives receiving only a few delegations are less affected. As a result, viscosity can increase the number of representatives whose vote is able to change the outcome of the election and reduce these near-dictatorships.

Many additional variations of liquid democracy exist which explore alternative methods of structuring delegations, aggregating ballots, and control over which delegations are possible. The remainder of this section introduces recent work across the topics within LD relevant to our work in Chapter 3 and Chapter 4.



(a) Delegation graph and voter weights with viscous democracy using $\alpha = 0$.

(b) Delegation graph and voter weights with viscous democracy using $\alpha = 0.5$.

(c) Delegation graph and voter weights with viscous democracy using $\alpha = 1$.

Figure 2.3: A delegation graph and representative weights under three values of viscosity. v_3 and v_6 vote directly while other voters select a neighbour to receive their delegation. (a) When $\alpha = 0$, delegations have no effect. (b) When $\alpha \in (0, 1)$ representatives receive less weight from their delegators, decreasing the further a delegation has to travel. v_3 receives 0.25 additional weight from v_1 and 0.5 additional weight from v_2, whereas both delegators of v_6 are on hop away and contribute 0.5 weight. (c) When $\alpha = 1$ delegation weight does not decrease as it travels and the model is equivalent to standard liquid democracy.

2.3.1 Epistemic Voting within Liquid Democracy

A great deal of work within LD focuses on an epistemic setting where there is a single correct outcome. We specifically discuss this work separately from our broader, general introduction to epistemic voting in order to maintain the cohesiveness of each section.

Definition 2.3.1. *The group accuracy of a set of voters with competencies q_i and delegation function d is*

$$Q(V, d) = \sum_{l=\lceil \frac{n}{2} \rceil}^{n} \sum_{S \in \mathfrak{R}_l} (\prod_{i \in S} q_i \prod_{j \notin S} 1 - q_j)$$

Where $\mathfrak{R}_l = \{\{v_i \in V_l \mid d^*(i) = i\} \mid V_l \subseteq V$ and $\sum_{v_i \in V_l} w_i = l\}$ contains all sets of representatives with weight summing to l.

Kahng et al. were among the first to consider whether using liquid democracy in an epistemic voting context could improve group accuracy [126]. Their main question is whether there are situations in which liquid democracy will *always* be more accurate than direct voting. In identifying their answer they focus separately on local vs centralized delegation mechanisms. Local mechanisms, where each voter decides for themselves where to vote based only on a limited view of the entire population, are shown to not yield any guaranteed improvement over direct democracy (the setting where every voter represents themselves) under some minor assumptions, such as having voters never choose to delegate to a voter less accurate than themselves. On the other hand, the paper presents an algorithm for a centralized delegation mechanism that will always lead to a strict improvement in accuracy.

The paper was directly responded to by Caragiannis and Micha who suggest that the assumptions of Kahng et al. are too strong [52]. Specifically, they show that there are situations in which achieving the accuracy-maximizing set of delegations requires voters to delegate to less accurate neighbours. In fact, the specific set of voters and network upon which they exist may be such that reasonable local delegation mechanisms such as delegating to your most accurate neighbour could significantly reduce accuracy (e.g. this would occur in a star network with the most accurate voter in the center and a large number of slightly less accurate voters). This is the first paper to consider what they call the *Optimal Delegation Problem* – that of finding the set of delegations which maximize group accuracy. They show that, unfortunately, even approximating the solution to this problem with low error is NP-hard.

More recently, Becker et al. [25] have considered this Optimal Delegation Problem and continued the line of research from Kahng et al. [126] and Caragiannis and Micha [52]. They strengthen the hardness results shown previously by considering particular social network structures. They show that when the network is sparsely connected and voters may suffer "misinformation," i.e. may have accuracy lower than 0.5, the ODP approximation bound becomes much looser. However, they also show that having voters delegate to their most competent neighbour is optimal when the social network is strongly connected. In our recent paper on viscous democracy, we extend the Optimal Delegation Problem to include viscosity – where some weight is lost through delegation [9].

Outside the above series of papers ground truth voting with liquid democracy has also been studied by Bloembergen et al. from a game-theoretic perspective [31]. They consider a setting where voters get an amount of utility equal to the accuracy of their representative if delegating, or equal to their own accuracy minus some small constant representing the "effort" of voting directly. Voters play a game wherein their actions are to delegate to some neighbour or to vote directly. The paper shows that in most situations this game has a Nash equilibrium which will be reached through a series of best responses from voters. Further, simulations show that best response updates typically reach equilibrium with only a very small number of updates per voter. This model is extended by Zhang and Grossi (discussed below) with a focus on how to measure the power of each voter [236].

2.3.2 Delegation Mechanisms

Delegation mechanisms are functions that determine to whom a voter delegates. These mechanisms can be either *local* (i.e. voters decide where to delegate based only upon knowledge of their neighbours) or *centralized* (i.e. a central authority decides the delegation of each voter). Centralized mechanisms have been criticized for potentially removing agency from individual voters; however, their benefits have been made clear through several papers. These were first formally discussed in Kahng et al.'s paper showing that local mechanisms cannot guarantee an improvement in accuracy over direct voting while centralized mechanisms can provide such a guarantee [126].

In our work, we consider both local and centralized delegation mechanisms, focusing more heavily on local mechanisms, and divide the definition of a delegation mechanism into two components. A *delegator selection function* identifies which voters will delegate, while voters use a *delegation probability function* to determine where they will delegate. We often define delegation probability functions in terms of the neighbours of a voter in G: $N_G(v_i) = \{v_j \mid (v_i, v_j) \in E\}$, and the more competent neighbours of a voter: $N_G^+(v_i) = \{v_j \mid (v_i, v_j) \in E \text{ and } q_j > q_i\}$.

Definition 2.3.2 (Delegation Mechanism)**.** *A delegation mechanisms is a tuple* $\mathcal{DM} = (g, \rho)$ *consisting of two functions:*

1. *A* ***delegator selection function*** $g : 2^V \to 2^V$ *selects* n^{final} *representatives who will delegate.*

2. *A* ***delegation probability function*** $\rho : V \times V \to [0, 1]$ *which accepts as input two representatives* v_i *and* v_j *and determines the probability that* v_i *will delegate to* v_j*.*

For example, the mechanism we call *Random Better* which selects the least competent voters to delegate to random neighbours more competent than they are, is defined as,

Definition 2.3.3. *The* ***Random Better*** *delegation mechanism is* $(g^{\text{worst}}, \rho^{\text{random}})$*:*

$$g^{\text{worst}} = K \subseteq V \text{ s.t. } |K| = n^{\text{final}} \text{ and } K = \arg\min \sum_{v_i \in K} q_i$$

$$\rho^{\text{rand\_better}}(v_i, v_j) = \begin{cases} \frac{1}{|N^+(v_i)|} & j \in N^+(v_i) \\ 0 & \text{otherwise} \end{cases}$$

In Chapter 3 and Chapter 4 we define several further delegation mechanisms which explore the possible dynamics of delegation and address concerns about delegation, such as those raised by Kling et al. [130] and Blum and Zuber [32], about the possibility that excessive delegation may lead to harmful concentration of voter weight – effectively creating dictators. Gölz et al. confirm this concern experimentally, but show that centralized delegation mechanism can largely respect voter delegation preferences while also avoiding dictatorships [105]. In their model voters delegate with some fixed probability and delegators approve k possible delegations of which a centralized delegation mechanism selects one with the goal of minimizing the weight of the most powerful voter after all delegations are made. While approximating the optimal delegation mechanism for this problem turns out to be NP-hard, it is possible in practice to develop a mechanism that dramatically reduces the expected maximum weight when voters are required to approve of even just $k = 2$ possible delegates.

A different approach to delegation is taken by Kotsialou and Riley who study two delegation mechanisms from an axiomatic perspective [131]. In their model voters are able to specify a ranking over all possible delegates and delegations are better viewed as a means of educating voters for whom they should vote, rather than allowing someone to vote on their behalf. This leads to an unusual model where a voter and their delegate may end up supporting different alternatives. They first focus on a delegation mechanism similar to the typical model where delegations flow from one voter to another, following each voter's highest preference which will not lead to a cycle, until reaching a representative. They argue that a problem exists if a delegation chain exists containing 3 voters, p delegating to q delegating to s, where s is the second choice delegate for q and q is first choice for p. Specifically, they show that this might lead to a situation where voters are incentivised to *not* receive more delegations and, instead, to abstain from voting. They propose a new delegation mechanism where, in the above situation, p would instead choose to delegate to their second choice rather than to q when q (and subsequent downstream voters) are unable to delegate to their first choice. This is shown to incentivise voters to participate in the election as they will always benefit from acquiring more delegations.

2.3.3 Delegation Structure

In "typical" liquid democracy a delegation represents the complete transfer of one voters weight to one other voter other possibilities exist. For example, the typical idea of delegation is most naturally applied to elections using the plurality rule; a vote is conceptually easier to transfer when it does not include rankings or other preference information. However, a small number of papers have considered alternatives to this type of delegation.

These alternatives have generally come in the form of approvals or rankings over neighbours as possible delegates which are then turned into a single delegation by some centralized mechanisms.

One exception to the above is given by Brill et al. who explored what happens when voters make multiple delegations, not to vote but to inform their own preferences [42]. Voters begin with a partial preference order over alternatives and, for each incomplete pairwise ranking of alternatives, delegate that preference decision to another voter. The paper then explore the complexities this introduces by possibly resulting in non-transitive preferences (e.g. through several delegations one voter may be told to prefer alternative a over b, b over c, and c over a) and various methods to maintain preference transitivity.

Brill et al. also explored a non-standard delegation structure more recently in a model that requires any delegator to submit a preference order over all neighbours as potential delegates [43]. The rankings are then converted to single delegations according to a variety of novel delegation functions which are analyzed axiomatically. This paper concludes with a novel simulation analysis of several measures of quality (average/max length of delegations, average/worst delegation preference in use, etc).

Gölz et al. studied a similar setting but with approvals, rather than rankings, over possible delegates [105]. They attempted to use this information to minimize the maximum weight of any representative in order to avoid dictatorships and maximize the power any individual voter has. They connect the problem to literature on congestion minimization in graphs and design an algorithm to approximate the optimal set of delegations given approval information. In both theoretical and simulation results they find the ability to dramatically reduce maximum voter power even when voters must approve of only 2 possible delegates.

A number of other papers have also discussed the idea of approving of neighbours as possible delegates. However, in practice this tends to reduce to a simple delegation structure as it is possible to decide locally among the approved options rather than submitting them to a central mechanism. For example, Kahng et al. discuss a delegation mechanism where delegators approve of all neighbours more competent than themselves and pick one to delegate to uniformly at random [126]. A similar use of approvals is also discussed by Becker et al. [25].

Paper	Theoretical	Simulation	Empirical	New Delegation Mechanism	Alternative Delegation Structure	Keywords about contribution
Boldi et al. (2011) [37]	✓	✓	×	×	✓	delegation weight decay
Kahng et al. (2018) [126]	✓	×	×	✓	×	ground truth, optimization
Kling et al. (2015) [130]	×	×	✓	×	×	power, real-world
Brill and Talmon (2018) [42]	✓	×	×	×	✓	pairwise delegation, preferences
Christoff and Grossi (2017) [57]	✓	×	×	×	×	cycles, binary aggregation
Gölz et al. (2018) [105]	✓	✓	×	✓	✓	approval, minimizing dictatorships, graph flow
Bloembergen et al. (2019) [31]	✓	✓	×	✓	×	game theory, ground truth
Caragiannis and Micha (2019) [52]	✓	×	×	×	×	ground truth, optimization
Escoffier et al. (2018) [81]	✓	×	×	×	×	preferences over delegations, game theory
Kotsialou and Riley (2018) [131]	✓	×	×	✓	×	axioms, incentivizing participation
Colley et al. (2021) [60]	✓	×	×	✓	✓	generalizing LD, axioms, preferences over delegations
Zhang and Grossi (2020) [236]	✓	✓	×	✓	✓	power, game theory, simulation, ground truth
Becker et al. (2021) [25]	✓	✓	×	✓	×	optimal delegations, simulation of many delegation functions
Brill et al. (2021) [43]	✓	✓	✓	✓	✓	preferences over delegations, axioms, simulations
Halpern et al. (2021) [116]	✓	×	×	✓	×	analysis of delegation rules, ground truth
Markakis and Papasotiropoulos (2021) [150]	✓	×	×	✓	×	approval delegation, minimizing dissatisfaction

Table 2.1: A brief overview of the main liquid democracy papers relevant to our work.

2.3.4 Representative Power

An emerging branch of research deals with measuring the distribution of power among representatives and determining how much power a voter receiving some number of delegations really has; for example, a voter receiving delegations from 10 distinct sources could be considered more powerful than a voter receiving the same weight from a single source as the second voter's weight is more easily removed.

Kling et al. wrote the earliest paper with an explicit focus on power in liquid democracy [130]. They analyzed real-world votes from the German Pirate Party using the LiquidFeedback software and found the existence of "super-voters" who they claimed had an outsized amount of power. In their work they fit a number of existing power indices to the data that they saw, including the Shapley, Banzhaf, Beta indices, and propose a new measure which leads to lower error on their test data.

As discussed above, Viscous Democracy is able to reduce the weight that powerful representatives have by not fully transferring all weight from a delegation [37]. The idea was originally applied as a method of identifying influence in citation networks. In follow-up work Boldi et al. use this principle of viscosity for music recommendation Boldi et al. Since then no prior work has done a theoretical analysis questioning whether viscosity might have any provable benefits in epistemic voting. Our work in Chapter 3 demonstrates the benefits of viscosity for epistemic voting [9].

More recently, a new measure of power has been developed by Zhang and Grossi [236]. They adapt the existing Banzhaf index to measure the influence of both representatives and delegators then use their index to extend the game-theoretic model from Bloembergen et al. [31]. They consider that when voters try to maximize their power in this model a Nash Equilibrium only exists if voters are on a complete social network. They provide simulations showing that the social network structure and degree to which voters are motivated by gaining power has a significant effect on how many voters delegate and the structure of delegations.

Many theoretical results about the concentration of voting power in a ground truth setting are provided by Halpern et al. [116]. They consider a delegation function in which the chance that a voter delegates to any neighbour is related to the competencies of the voter and their neighbour and extend the *positive gain* and *do-no-harm* metrics considered by Kahng et al. [126] to this probabilistic setting. Their primary result is to prove that three broad classes of delegation function are likely to avoid significant concentration of power under a wide range of settings.

Finally, one other paper has considered power in empirical settings. Fritsch et al. study power distribution in three different decentralized autonomous organizations (DAOs) which use liquid democracy for their governance [95]. Voters in these DAOs hold special *voting tokens* which correspond to their voting power. The paper shows a high inequality in the distribution of these tokens (making the comparison that "the inequality in the general distribution of wealth in the world is a lot lower") and that between 8 and 18 voters hold over 50% of the voting tokens in these systems. Thus far it has been very rare for powerful minorities to vote against the less powerful majority.

2.4 Machine Learning and Optimization

The focus of our work is on showing that social choice can benefit from machine learning (ML) and optimization techniques, and that these techniques can provide value to social choice. Each project that we discuss uses different ML paradigms, each with their own purpose and application domain. Here we introduce the most important ML and optimization concepts that we use.

2.4.1 Equivalence of Classification and Voting

A vital component of our work is the equivalence of epistemic voting and classification. As discussed previously, a voter in an epistemic setting does their best to vote for the correct outcome and some aggregation function combines input from multiple voters. On the other hand, ML classification involves a model observing some input features and making a prediction about the class that corresponds with those features. There is (usually) exactly one correct class for any given set of features. While social choice research more often focuses on how to aggregate votes, machine learning research is more likely to focus on how predictions are made.

Our work highlights the equivalence of these two settings. Classifiers making predictions can be seen as a specific instantiation of an epistemic social choice model. When considering classification of an entire dataset, this turns into multi-issue epistemic social choice. This observation allows us to apply results and techniques from one area of research to the other, as we do when using liquid democracy to generate ensembles in Chapter 4.

2.4.2 Voting as Maximum Likelihood Estimation

The area in which prior work has overlapped the most between social choice and machine learning connects epistemic social choice with maximum likelihood estimation. This occurs when there is some correct ranking (epistemic voting) and voter preferences are some noisy reflection of this true ranking. When the noise model of voter preferences is known, we can compute the voting rule which is most likely to identify the true ranking – the maximum likelihood estimate of the noise model [62].

Condorcet initiated this topic of study (without direct reference to statistics or machine learning) by considering a noise model in which each voter ranks two candidates correctly with probability greater than 0.5 [161]. Young subsequently identified the rule most likely to identify the correct ranking under Condorcet's noise model [234]. More recently, Conitzer and Sandholm have found noise models for which many common voting rules are maximum likelihood estimators [62]. Our work does not directly utilize the connection between maximum likelihood estimation and voting; we discuss the topic as one which highlights common ground between machine learning and social choice.

2.4.3 Machine Learning Models

We use three different machine learning models through this work, which we describe below. Each of these models performs *classification* tasks – predicting a class label given a specific set of input features (as opposed to regression, in which the model predicts the value of a continuous function). Here we briefly introduce each model and describe some relative advantages and disadvantages of the model. Of particular interest to our work, and our decisions around when to use each model are two factors: the speed of training a model, and whether a single model can undergo many rounds of iterative training building upon previous rounds.

Decision Trees A decision tree consists of internal nodes which correspond to binary conditions about an input feature, edges connecting each internal node to child nodes based on the condition value, and leaf nodes which represent a class value. An input instance is classified by evaluating the condition of the root node and recursively following the path corresponding to the value of the node's condition until a leaf node is reached. We use decision trees for ensembles of classifiers in Chapter 4.

The example shown in Figure 2.4 depicts a simple classifier which separates input features on two features: x and y. Input examples with both x and y greater than 2 are put in class 1, while other examples are put in class 2.

The typical approach for constructing (i.e. training) decision trees centers on the concept making nodes which maximally reduce "entropy" [184]. The entropy of a dataset loosely measures how evenly split the data is between classes – if only one class is present

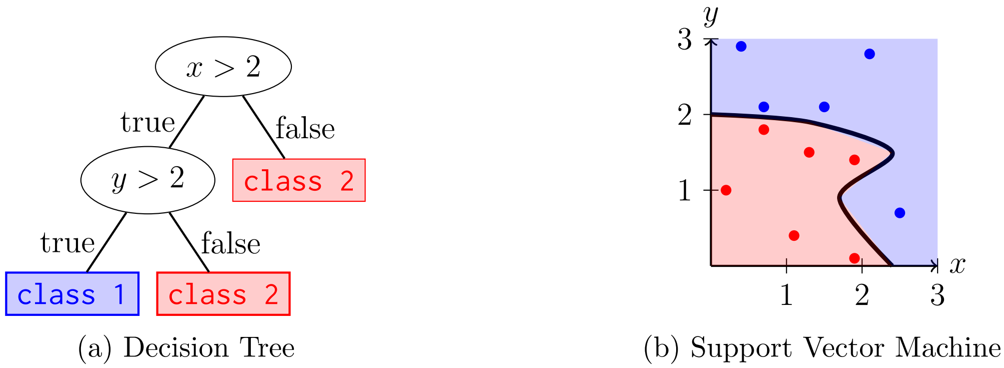


Figure 2.4: Example of a simple classification task using (a) a decision tree, and (b) a support vector machine. In the support vector machine, a decision boundary (black line) separates red and blue points by dividing space into two classes, similar to the decision boundaries shown in the decision tree.

there is zero entropy and entropy is maximized when data is evenly split between classes. For example, if 90% of examples in a training set can be classified by creating a node with the value $x > 2$ while any other decision value would lead to a smaller reduction in unclassified data, an algorithm is likely to create a node with the decision point of $x > 2$. Decision trees are extremely fast to train; however, all common approaches require they not have multiple rounds of training; if more data arrives or the underlying distribution shifts and more training is required then the tree must be retrained from scratch.

Support Vector Machines In Chapter 4 we use Support Vector Machine (SVM) classifiers. A Support Vector Machine aims to find some hyper-plane that perfectly separates training examples into their respective classes. The hyper-plane is constructed to have maximal distance from each of the points closest to it. In two dimensions, this corresponds to constructing a line which has all points of one class on one side and all points of another class on another side.

A linear SVM attempts to do this by drawing a straight line (or hyper-plane in higher dimensions). In many cases, including the example in Figure 2.4, this is not possible. Two ideas make SVMs more practically useful: (1) allowing some misclassifications and aiming to minimize misclassification error, and (2) using a kernel function which serves to transform the underlying space in such a way that a straight line is more likely able to separate the data into classes (equivalently, this can be thought of as constructing a separator that is not linear) [65].

The example in Figure 2.4 shows a setting where some kernel function must have been used to transform the decision boundary so that it can perfectly separate the data. SVMs can be trained quickly using stochastic gradient descent. They do not train as quickly as decision trees; however, an existing SVM can pause and continue training. This is useful for settings where there may be many rounds of training as new data arrives or the target distribution shifts.

Neural Networks Neural networks are at the heart of the rise in visibility of artificial intelligence that begin around 2012 [132]. Perceptrons, precursors to neural networks were first built by Rosenblatt in 1957 [190]. Simple neural networks compose a large number of perceptrons as nodes in inter-connected layers. Each node has some activation condition that triggers an output based on the collective value of the inputs it receives. The theoretical capability of neural networks to act as a general function approximator has been known for several decades; however, not until recently has enough compute power been available to train networks large enough to perform useful tasks (beginning with image recognition in 2012 [132]).

Neural networks are trained by "backpropagation" which works by measuring the error that a network has on some training examples, measuring how much the value of each parameter inside the network contributed to that error, and adjusting the weights in a way that should reduce error. A form of gradient descent is used to adjust the weights [222]. This procedure requires a large amount of time and data, but is highly effective at making use of hidden structure in the underlying data. The process also lends itself to incremental training; it is possible to update a trained network with new training data. We use neural networks in Chapter 5 to learn functions similar to existing voting rules and to act as novel voting rules.

2.4.4 Ensemble Learning

Ensembles in machine learning are an extremely thoroughly studied topic with many research areas and subtopics. Here we aim to describe the fundamental nature of ensembles as well as the specific subtopics relevant to our research.

On the surface, an ensemble can be viewed as a machine learning model with the same interface as decision trees, SVMs, and neural networks: A user interacts with an ensemble by training it and by asking it to make predictions on data. Structurally, an ensemble is an aggregation of many individual machine learning models, all of which are individually trained and themselves make predictions.

The aspects of an ensemble which are of most interest to us are: (1) How ensembles aggregate the output of individual classifiers into a single decision, (2) How ensembles interact with the training of their models, and (3) Ensemble diversity. Polikar refers to these topics as "the three pillars of ensemble systems" [178].

Combining Ensemble Members The most straightforward method of combining the predictions of many classifiers is by a simple plurality vote. If ensemble members are all considered equally important the output is the class which was predicted by most classifiers. If some classifiers are given a higher weight than others, the class which receives the most weight of predictions becomes the output.

While the plurality vote (usually called "majority vote" in ensemble literature [178]) is effective, much work aims to improve upon it. This work focuses on two directions: using information from a classifier beyond their prediction, and using different voting methods.

Many machine learning models are able to output not just a prediction but also a confidence score in that prediction [2]. Schapire and Singer demonstrated that using the confidence of each classifier can improve accuracy in boosting algorithms [194].

As discussed previously, classification can be naturally phrased as a social choice task. As a result, it becomes natural to use voting rules to aggregate ensemble output. Leon et al. compared four voting rules (Plurality, STV, Borda, and Copeland) across ensemble types; their results suggest that Borda and Copeland outperform other methods at aggregating classifier outputs [143]. Campagner et al. found a less marked distinction between voting methods in a similar experimental analysis which compared Plurality, Approval, Borda, Copeland, and several existing ensemble methods [50]. A third paper along similar lines uses Plurality, Borda, Copeland, and Kemeny voting rules. Through both experimental and theoretical analysis they argue that Plurality and Copeland lead to the highest overall accuracy [64].

Training Ensembles Members of an ensemble are often trained with the same underlying algorithm that would be used to train the model if it were not in an ensemble (e.g. stochastic gradient descent for an SVM). However, the ensemble process frequently modifies this algorithm slightly. Two common ensemble approaches are bagging and boosting, which exemplify this idea of slightly modifying the standard training of a classifier:

[2] For an intuition around what confidence means consider the right-hand side of Figure 2.4. A classifier might have low confidence on points near the decision boundary, since they have feature values similar to points in the other class. The classifier might also have high confidence on points far from the decision boundary since they are dissimilar from anything in the other class.

Boosting - The first classifier is trained using an unmodified training algorithm. Subsequent classifiers are trained with additional weight placed on examples that previous classifiers misclassify. This makes the new classifiers more likely to classify those examples correctly and often works well with models that have low accuracy. We compare our results in Chapter 4 against Adaboost, perhaps the most well-known boosting algorithm [94].

Bagging - Each classifier is trained on a different random subset of the entire training set. This works effectively as a method of avoiding overfitting when using classifiers that have high accuracy [40].

Pruning is an ensemble technique typically applied after training individual models. The core idea of pruning is simply to remove ensemble members that are not useful. Zhou et al. proved that pruning weak members can improve the accuracy of ensembles [238]. Since then, many pruning strategies have been explored. One survey divides most approaches into *ranking* or *search-based* methods [192].

Ranking methods score each ensemble member based on some metric, often some combination of diversity and accuracy, then greedily remove the lowest scoring members. A ranking method might use factors such as uncertainty of individual classifiers [170] or the similarity between classifiers [148, 22]to score ensemble members.

Search-based methods for pruning work by searching for a set of classifiers that should remain in the ensemble. This search can take many forms: a heuristic estimation of the strength of each set of classifiers [238], an evolutionary optimization task [182], or phrased as an integer programming problem [235].

We are aware of only one paper which claims to apply pruning during the training of individual classifiers in an incremental learning setting. Zhao et al. develop a method of creating a new ensemble for each increment of data which is subsequently pruned to reach a single final ensemble [237]. Their method achieves high accuracy but appears quite resource-intensive; we focus on reducing the resources needed for an ensemble.

Diversity within Ensembles The drive for diversity in ensembles stems from the observation that if an ensemble is composed solely of several identical copies of the exact same classifier, it would perform no better than the original classifier. On the other hand, in order for classifiers to be perfectly accurate they must always make the same decision. Thus, intuition suggests that classifiers in an ensemble should make decisions which are dissimilar from one another but should become more similar as the classifiers become more accurate.

Diversity can be realized in several ways, such as: using different underlying models, using different parameters of the same model type (e.g. neural networks of different topologies), or training models on different data. Bagging and boosting approaches can be seen as ways of increasing diversity in targeted ways: e.g. boosting methods create new ensembles with the goal that they produce a different output on the specific examples that previous ensemble members misclassify. While diversity is useful for an ensemble, the extent and type of diversity between classifiers that is useful is not well understood.

A precise measure of "useful" diversity remains unclear. In a survey article, Kuncheva and Whitaker explore 10 different measures of diversity, based primarily on classifier output. They find only mild evidence that any of their definitions of diversity are associated with ensemble accuracy [134]. A similar review considers additional measures of diversity and endeavours to categorize diversity measures in terms of what data is used to calculate the measure (either a focus on bias, variance, and covariance, or on the "ambiguity decomposition") [44].

While hard to pin down concretely, the benefit of diversity can be demonstrated empirically by approaches which generate ensembles by adding classifiers that are designed to be highly dissimilar from the existing classifiers [144, 54]. This approach shows clear improvements to the generalization of ensembles on unseen data.

2.4.5 Incremental Learning

Incremental learning is a mode of learning in which a model's training data is divided into multiple pieces which are learned on consecutively, rather than in a single training phase. Here we introduce three motivations for learning incrementally:

1. When the complete set of training data is too large to fit in memory the data can be loaded in batches and each batch learned separately.
2. When the distribution from which data originates changes over time a model may need to continually train in order to maintain high accuracy. This setting is also called *continual learning* [216].
3. When the training algorithm benefits from a continual learning process.

Our work concentrates primarily on the third motivation but is equally applicable to all three points. The major focus of incremental learning algorithms is typically on how the training algorithm must change; either to allow the underlying model to train incrementally, or to optimally adapt training to a changing distribution in a way that learns new information without forgetting previously-learned knowledge.

Some ML training algorithms adapt naturally to many rounds of learning – such as stochastic gradient descent, an inherently iterative algorithm [189]. On the other hand, many decision tree algorithms such ID3 [184] and C4.5 [183] directly construct a tree based on the entire dataset and are not well suited to incremental learning. Incremental variants of decision tree algorithms exist but are not yet the most common or practical algorithms for general use [55].

For this reason, our work using ensembles for incremental learning in Chapter 4 makes use of decision tree ensembles in the non-incremental setting only and uses support vector machines trained with stochastic gradient descent for incremental training. We also briefly discuss the application of our work to a continual learning setting, where the distribution of data changes over time. For that work we use ensembles of neural networks, also trained with stochastic gradient descent [30].

2.4.6 Simulated Annealing

Simulated annealing is a general purpose optimization algorithm inspired by the metallurgical annealing process [129]. Simulated annealing can operate in any domain in which the solution to the optimization problem can be represented as some location in space. The algorithm begins in a random location and makes pseudo-random movements. New locations are discarded in favour of keeping the old location based on a function of their quality and a *temperature* value that decreases over time. Early in the process, new locations which are kept might reduce the solution quality. As temperature decreases, newer locations are more likely to be kept only if they are an improvement. Simulated annealing was originally proposed by Kirkpatrick et al. to find solutions to the NP-hard travelling salesman problem. Similarly, we use simulated annealing in Chapter 3 for the NP-hard task of optimizing delegations in liquid democracy.

We use simulated annealing, rather than any other common optimization technique such as stochastic gradient descent, as the delegation optimization function is not easily analyzable: computing both the exact value and the gradient of the group accuracy function $Q(V, d)$ is expensive, and we do not know whether the function is convex. Using a more easily applicable optimization technique proved sufficient for our purposes.

Chapter 3

Liquid Democracy for Ground Truth Voting

Liquid democracy is often discussed in epistemic domains where voters collectively prefer to elect some "correct" alternative. Prior research has developed theoretical results on the impossibility of guaranteeing an improvement to accuracy from delegation; however, little work has explored the dynamics of group accuracy under a range of experimental settings. More directly: conceptual models with theoretical results abound, yet little work highlights the actual efficacy of liquid democracy for epistemic voting. This chapter demonstrates experimentally the performance of several forms of liquid democracy across a broad spectrum of voter competencies and social network structures.

3.1 Introduction

Liquid democracy is often touted as a method of increasing voter engagement by reducing the effort needed to participate in an election. If some fraction of voters prefer to delegate rather than vote directly, liquid democracy aims to maintain democratic ideals while respecting this preference. Many theoretical models explore novel ideas around a voter using delegation to determine their individual preferences [42], allowing voters to express delegations conditionally [60], or measuring the effective power that a voter has [236].

While the task of identifying delegations which optimize group accuracy in an epistemic setting has been thoroughly analyzed theoretically, and found NP-hard even to approximate [126, 52], it has received minimal experimental analysis. In this chapter, we demonstrate experimentally the performance of several distinct methods of delegating. Each delegation method can be thought of as approximating some behaviour that a human voter might exhibit while delegating, such as "delegate to the smartest person I know." Alternatively, delegation mechanisms might be viewed as behaviours prescribed to agents in a liquid democracy setting and can thus impose arbitrary conditions, such as reducing the chance of delegating to voters with high weight.

We show that even very basic delegation methods lead to significantly improved group accuracy when compared with direct democracy. However, Gölz et al. demonstrated that weight concentration from such methods is significant [105]. Excessive concentration of weight is often considered anti-democratic, as it results in a single (or a few) voters single-handedly deciding the outcome of the election. If that single voter is always correct this can benefit epistemic settings; however, if that voter may vote for an incorrect alternative this can have very harmful effects [130]. Put differently: delegation improves accuracy by finding a balance between the number of competent voters participating directly, and the number of less competent voters who are removed from direct participation via delegation. When too many voters are removed, the benefit of many voters (as shown by Condorcet's Jury Theorem [161]) disappears.

In order to mitigate the concentration of weight during delegation, we introduce a new delegation mechanism which reduces the chance of creating a delegation that increases weight centralization. As well, we extend Boldi et al.'s concept of "viscosity" to epistemic voting. This causes weight to decay as it is transferred through delegation. Through experiments across a wide range of social networks and voter competence distributions we show that, while finding optimal delegations is impractical, we can very effectively improve epistemic performance through the introduction of delegation. The novel contributions of this chapter are:

- We demonstrate the behaviour of liquid democracy for epistemic voting across a wide range of experimental settings (Section 3.4).
- We show that simple delegation mechanisms are able to significantly improve group accuracy over the non-delegative setting of direct democracy (Section 3.4).
- We extend the basic model of liquid democracy to include "viscosity" and show that viscosity is able to further improve group accuracy over the liquid democracy setting by avoiding weight centralization (Section 3.5).
- We identify a delegation mechanism which improves group accuracy by avoiding weight centralization (Section 3.5).

Symbol	Meaning
V	Set of voters.
n	Total number of voters.
$A = \{a^+, a^-\}$	Set of alternatives: a^+ is the sole epistemically correct alternative.
q_i	Competence of voter v_i – chance that v_i will vote for a^+ over a^-.
Q	Group accuracy; the chance that an election will result in a^+ winning.
α	Viscosity; the factor by which weight of a delegation decreases each hop that it travels.
w_i	Weight of voter i.
$W^+(V)$	The set of voters with non-zero weight; the representatives.
$G = (V, E)$	The social network upon which voters exist.
$\texttt{dist}_G(v_i, v_j)$	The length of the shortest path in G between v_i and v_j.
$N_G(v_i)$	The neighbours of v_i on G.
$N_G^+(v_i)$	The neighbours of v_i on G with strictly higher competence than v_i.
d	Delegation function. $d : V \to V$ indicates to which voter i delegates their vote.
D	The subgraph of G induced by d; $D = (V, \{(v_i, v_j) \in E \mid d(v_i) = v_j\})$.
g	Delegator selection function. $g : 2^V \to 2^V$ selects which voters will begin to delegate in the current increment.
ρ	Delegation probability function. $\rho : V \times V \to [0, 1]$ gives the probability that the given representative will delegate to the voter given as the second argument.

Table 3.1: Overview of notation most important throughout this chapter.

In total, we show that liquid democracy is a highly effective system for finding ground truth. Viscosity is a broadly applicable extension of the liquid framework that significantly mitigates the negative impact of excessive weight centralization.

3.2 Model

In this section, we review the concepts introduced in Chapter 2 which are applicable to this chapter. Table 3.1 provides a reference of the notation used through this chapter. We operate here in a setting of epistemic, ordinal social choice. Specifically, we consider a setting with two alternatives $A = \{a^+, a^-\}$ with a^+ being the alternative which is universally correct for all voters. We keep our definition of alternatives abstract; however, this setting of two alternatives can be thought of as either two alternatives or a more general abstraction where a^+ represents *one of several* good outcomes, and a^- represents *one of several* worse outcomes. Across n voters V, each v_i has some competence level q_i representing their ability make an optimal choice. In line with our ordinal setting, v_i has the preference order $a^+ \succ a^-$ with probability q_i, and has the preference order $a^- \succ a^+$ with probability $1 - q_i$. We commonly say that v_i "votes for" a^+ with probability q_i. Voters exist on some underlying social network which is represented by an undirected graph $G = (V, E)$. Each node is represented by a voter and edges between voters represent some relationship between voters. As the graph is undirected, note that an edge between v_i and v_j is equivalently represented as either (v_i, v_j) or (v_i, v_j). The neighbours of v_i in G are denoted $N_G(v_i) = \{v_j \mid (v_i, v_j) \in E\}$. We also consider the subset of neighbours strictly more competent than v_i: $N_G^+(v_i) = \{v_j \mid (v_i, v_j) \in E \ , \ q_j > q_i\}$. As well, $\mathtt{dist}_G(v_i, v_j)$ refers to the distance of the shortest path between v_i and v_j in G, with $\mathtt{dist}_G(v_i, v_i) = 0$. We assume that we work with fully connected graphs so a path between any two voters always exists.

May's Theorem shows that any reasonable aggregation of preferences over two alternatives reduces to weighted plurality voting, where only a voter's first preference matters [153]. As our model always has two alternatives, we use weighted plurality voting to decide on a winner. Correspondingly, each voter has some weight w_i which represents their voting power. In subsequent sections, we will find it convenient to refer to all voters with non-zero weight; we denote these voters as $W^+(V) = \{v_i \mid w_i > 0\}$. Weighted plurality is the voting rule which selects as winner the alternative which receives the most votes, as measured by the sum of weight on ballots in which they are ranked first. That is,

$$\mathcal{F}^{\text{WP}}(\mathcal{P}) = \arg\max_c \sum_{\{P_i \in \mathcal{P} \mid P_i[1] = c\}} w_i$$

3.2.1 Voting Systems

This chapter explores how different conceptual ideas of setting voter weight and choosing which voters should participate in an election affect group accuracy. Here we describe each of the specific methods that we compare.

Direct Democracy

Direct democracy is the setting where all voters vote directly and equally. Each voter has a weight of 1 and votes for a^+ with probability equal to their competence. This is equivalent to the special case of liquid democracy where no voter delegates.

Liquid and Viscous Democracy

Viscous Democracy is a generalization of Liquid Democracy. We first describe the elements common to both settings and subsequently elaborate on their distinction. In both cases, each voter has two options:

1. A voter can **delegate** their vote to some neighbour in G. In most respects, this removes the voter from the election (the exception being that they still receive and transitively delegate the delegations of other voters).
2. A voter can **participate directly** as a *representative*. We model direct participation as a self-delegation. Each representative votes for a^+ with probability equal to their competence.

A delegation function $d : V \rightarrow V$ maps each voter to their delegate. $d(v_i) = v_j$ indicates that v_i delegates their vote to v_j, who may delegate both votes further or may vote directly. We use $d^*(v_i) = d(d(d(...))) = v_j$ to describe the repeated application of d until a self-delegation is reached; $d^*(v_i)$ is the representative of v_i. Making delegations involves a voter selecting one of the edges they are connected to in G which can be seen as inducing a subgraph of G. We refer to this induced subgraph as a *delegation graph* $D = (V, \{(v_i, v_j) \in E | d(v_i) = v_j\})$. D is a directed subgraph of G induced by the delegation function d.

The distinction between Liquid and Viscous Democracy arises in weight calculation. Viscous Democracy is defined by a viscosity parameter $\alpha \in [0, 1]$ which determines how weight flows through the network. The weight of a voter v_i is defined as,

$$w_i = \begin{cases} 0 & \text{if } d^*(v_i) \neq v_i \\ \sum_{\{v_j \in V | d^*(v_j) = v_i\}} \alpha^{\mathtt{dist}_D(v_j, v_i)} & \text{otherwise} \end{cases}$$

That is, voters who delegate have a weight of zero while voters who participate directly gain weight from each delegation they receive, including indirect delegations and their own delegation. The weight a representative receives from a delegator some distance h hops away on the delegation graph D is equal to α^h. Liquid Democracy is the special case of VD where α is always set to 1, so the weight of a representative is simply the number of delegations it receives.

Note that group accuracy is a function of sets of voters that sum to at least half of total weight. As these sets may not change as voter weight changes, group accuracy is a discontinuous function as α varies. As a result, $Q(V)$ may have the same value for many consecutive values of α. We refer to any value of α which maximizes group accuracy as α^*.

3.3 Experimental Setup

The experimental results of this chapter are presented in Section 3.4 and Section 3.5. In this section we describe the implementation of our experimental framework in detail beyond what is described in our model. We present algorithms for estimating group accuracy and for optimizing delegations, and outline the specific preference distributions and delegation mechanisms used by voters. Each of the components described in this section is used in to generate some of the results in the succeeding sections.

3.3.1 Estimating Group Accuracy

Exactly computing the group accuracy Q as stated in Definition 2.3.1 requires iterating over all sets of voters with more than half of total weight and calculating the probability of each particular set of votes occurring. We found this feasible on small scales but the equation required a prohibitively large amount of time for large scale experiments. Instead, we employ Monte Carlo simulation to estimate group accuracy [158] as detailed in Algorithm 1. We do this by running many elections – each election involves sampling a vote from each voter based on their competence – and reporting the fraction of elections in which voters elect a^+. In all reported group accuracy results, we are reporting the proportion of 1000 elections which elected a^+. Comparison with exact computation of group accuracy on small elections showed us that this method typically finds a value within 0.02 of the true group accuracy.

3.3.2 Competence Distributions

In our experiments we select some distribution $\mathcal{D}$ from which we sample voter competencies, so that for each $v_i \in V$ we independently sample a value $q_i \sim \mathcal{D}$. We use the following

Algorithm 1 Monte Carlo Group Accuracy Estimation

Input: V, A, K iterations
Output: Group accuracy Q
Initialize `count_correct` $\leftarrow 0$
for $k = 1$ to K **do**
  Sample a vote a_i from each voter $v_i \in V$
  Compute the weight voting for a^+: $W^+ = \sum_{v_i \in V \mid a_i = a^+} w_i$
  if $W^+ \geq \frac{n}{2}$ **then**
    `count_correct` $\leftarrow$ `count_correct` $+1$
  end if
end for
Return $\frac{\texttt{count\_correct}}{n}$

three families of distributions as implemented in SciPy [218]:

- **Uniform distributions**; $U(a, b)$ refers to a uniform distribution with a minimum value of a and maximum value of b.
- **Truncated Gaussian distributions**; $\mathcal{N}(\mu, \sigma)$ is a truncated Gaussian distribution with mean of μ and standard deviation of σ. In all cases this distribution is truncated to be in $[0, 1]$.
- **Truncated Exponential distributions**; $Exp(\lambda)$ is an exponential distribution with rate parameter λ. As the exponential distribution does not provide an upper bound on sampled values, whenever we sample a value greater than 1 we map the value to 1. This leads to a mean value slightly lower than in the standard exponential distribution, i.e $\mu' = \frac{1}{\lambda} - \frac{e^{-\lambda}}{\lambda}$.

3.3.3 Social Networks

In our experiments we consider two families of artificial network, both of which are chosen for their similarity to existing real-world social networks.

- **Erdős–Rényi (ER)** networks use two parameters: the number of nodes n, and the connection probability p. They are generated by initializing an empty network of n nodes and independently inserting each possible edge with the fixed connection

Delegation Mechanism	Selection Function	Probability Function
Direct	g^{direct}	ρ^{direct}
Max	g^{worst}	ρ^{max}
Random Better	g^{worst}	$\rho^{\text{rand\_better}}$
Proportional Better	g^{worst}	$\rho^{\text{prop\_better}}$
Proportional Weighted	g^{worst}	$\rho^{\text{prop\_weighted}}$

Table 3.2: The delegation selection and probability functions that define each delegation mechanism used through this chapter.

probability p [80]. ER networks are *small-world*; they have a low average distance between any two nodes.

- **Barabási–Albert (BA)** networks use two parameters: the number of nodes n, and attachment parameter m. Nodes are generated via a preferential attachment process beginning with m fully connected nodes. The remaining $n - m$ nodes are added one at a time and connected to m existing node. The probability of a new node being connected to an existing node is proportional to the degree of the existing node [5]. BA networks are *scale-free* – their node degree distribution follows a power law, resulting in many nodes with very low degree and few nodes with very high degree – but are not small-world.

3.3.4 Delegation Mechanisms

Through this chapter we use each of the delegation mechanisms listed in Table 3.2. We refer to Chapter 2 for a formal definition of, and more discussion surrounding, delegation mechanisms.

Delegation Selection Functions

Definition 3.3.1. *Delegator Selection Function slightly abuses notation and does not return n^{final} voters. Rather, $g^{\text{direct}}(V, d)$ selects no representatives and returns $\emptyset$.*

Definition 3.3.2. *Delegator Selection Function $g^{\text{worst}}(V, d)$ selects the n_t^{final} representatives with the lowest competence:*

$$g^{\text{worst}} = K \subseteq V \text{ s.t. } |K| = n^{\text{final}} \text{ and } K = \arg\min \sum_{v_i \in K} q_i$$

Delegation Probability Functions

Definition 3.3.3. *The* ***Direct*** *Delegation Probability Function selects no delegates and has all voters represent themselves. This is equivalent to the situation where each voter delegates to themselves. Note that, when used with g^{direct} this function is redundant as g^{direct} does not select any new delegators.*

$$\rho^{\text{direct}}(v_i, v_j) = \begin{cases} 1 & v_i = v_j \\ 0 & otherwise \end{cases}$$

Definition 3.3.4. *The* ***Random*** *Delegation Probability Function selects delegates uniformly at random for new delegators.*

$$\rho^{\text{random}}(v_i, v_j) = \begin{cases} \frac{1}{|N(v_i)|} & v_j \in N(v_i) \\ 0 & otherwise \end{cases}$$

Definition 3.3.5. *The* ***Random Better*** *Delegation Probability Function selects a delegate for each delegator uniformly at random from neighbouring voters with strictly higher accuracy [126].*

$$\rho^{\text{rand\_better}}(v_i, v_j) = \begin{cases} \frac{1}{|N^+(v_i)|} & v_j \in N^+(v_i) \\ 0 & otherwise \end{cases}$$

Definition 3.3.6. *The* ***Max*** *Delegation Probability Function has each voter delegate to their most competent neighbour. In the case of multiple maximally competent neighbours (not represented in the equation below), ties are broken lexicographically.*

$$\rho^{\max}(v_i, v_j) = \begin{cases} 1 & q_j = \max\{q_k \mid v_k \in N(v_i)\} \\ 0 & otherwise \end{cases}$$

Definition 3.3.7. *The* ***Proportional Better*** *Delegation Probability Function has delegators select a more competent neighbour; however, the chance of delegating to a given voter is directly correlated with the difference between their accuracy and that of the delegator. Shown here are delegation probabilities before normalization.*

$$\rho^{\text{prop\_better}}(v_i, v_j) \propto \begin{cases} \frac{q_j - q_i}{\sum_{v_k \in N^+(v_i)} q_k - q_i} & v_j \in N^+(v_i) \\ 0 & otherwise \end{cases}$$

Definition 3.3.8. *The **Proportional Weighted** Delegation Probability Function returns delegation probabilities based on both the accuracy difference between delegator and delegatee, as well as the weight of the representative ultimately receiving the delegation. A lower weight leads to a higher delegation probability. Shown here are delegation probabilities before normalization.*

$$\rho^{\text{prop\_weighted}}(v_i, v_j) \propto \begin{cases} \frac{1}{w_{d^*(v_j)}} \frac{q_j - q_i}{\sum_{v_k \in N^+(v_i)} q_k - q_i} & v_j \in N^+(v_i) \\ 0 & \textit{otherwise} \end{cases}$$

3.3.5 Simulated Annealing to Optimize Delegations

To find near-optimal sets of delegations we use simulated annealing to construct a delegation graph. We do this using the procedure outlined in Algorithm 2, using the Python `simanneal` library[1]. For each iteration, simulated annealing selects a random voter to make a random delegation (replacing their old delegation, if it exists, or turning a delegator into a representative through self-delegation). This delegation is kept if (1) it improves group accuracy, based on a 100 iteration Monte Carlo accuracy estimation, or (2) the delegation passes the probabilistic threshold based on difference from previous group accuracy and temperature. Over the course of the algorithm temperature decreases, leading to a lower probability of accepting delegations which reduce group accuracy. We refer to the `simanneal` library for the fine details of temperature reduction.

3.4 Experiments: Accuracy within Liquid Democracy

In this section we study the question: "What factors affect group accuracy when using liquid democracy?" We specifically explore the impact of voter competence distributions, the chance that voters delegate and topology of the underlying social network. We show here both our initial results highlighting that LD *is* frequently very beneficial to group accuracy, and our exploratory experiments aiming to identify the factors that lead to this benefit. The work in this corresponds to work we have published in IJCAI 2022 [6].

Algorithm 2 Simulated Annealing to Optimize Delegations

Input: Set of voters V, social network G, number of iterations K
Output: Delegation graph d
for $k = 1$ to K **do**
  {`Attempt to make a delegation.`}
  for $\_ = 1$ to 50 **do** {}
    $v_i \leftarrow$ random voter from V
    $v_j \leftarrow$ random neighbour of v_i in $G \cup \{v_i\}$
    if $\mathit{delegation\_possible}(v_i, v_j)$ **then**
      $\mathit{delegate}(v_i, v_j)$
      break
    end if
  end for
  {`Decide whether to keep delegation.`}
  $E^{\text{new}} \leftarrow \mathit{monte\_carlo\_group\_accuracy}(V, A, 100)$
  if $E^{\text{new}} > E^{\text{old}}$ **and** $\texttt{exp}(\frac{E^{\text{old}} - E^{\text{new}}}{T}) < \mathit{random}(0, 1)$ **then**
    undo delegation
  else
    keep delegation
  end if
  update temperature T
end for
return d

3.4.1 Fraction of Representatives

We first explore how the level of voter participation in elections affects accuracy. This addresses whether there can be "too much" delegation. Figure 3.1 shows the average over 10 trials of simulations increasing the fraction of voters participating directly. The left side begins with 90% of voters delegating, decreasing to 10% delegating on the right side of the figure. In each trial a new random network with average degree of 20 is generated. For BA networks, we set $m = 10$ and in ER networks $p = 0.20202$ ($p = 0.02002$) for networks of 100 (1000) voters. Voter competencies are sampled as follows for each distribution:

Uniform - $\forall v_i : q_i \sim U\,(0.3, 0.7)$

Gaussian - $\forall v_i : q_i \sim \mathcal{N}\,(0.5, 0.1)$

[1] Simanneal can be found at https://github.com/perrygeo/simanneal

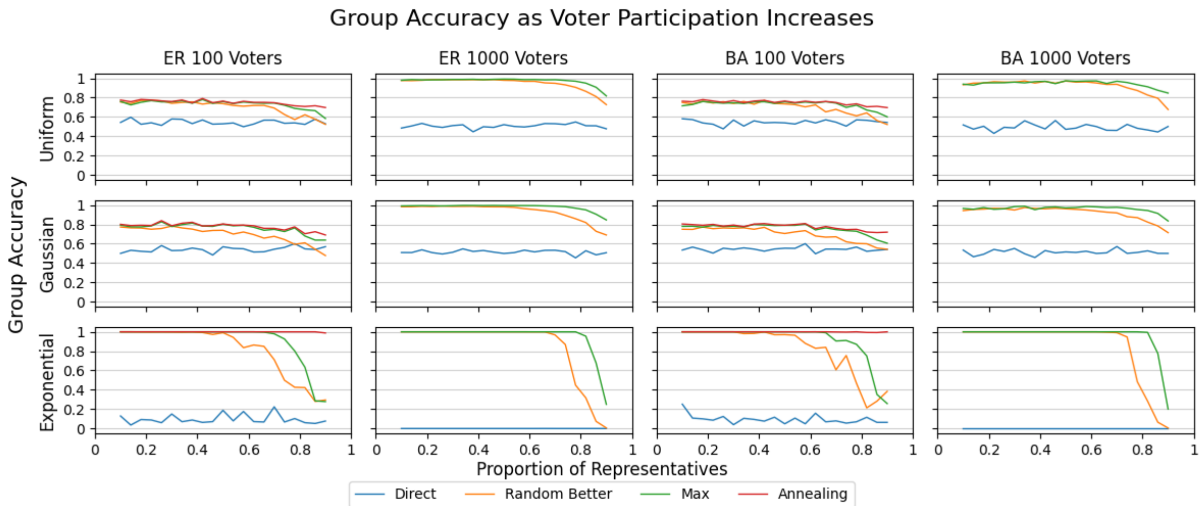


Figure 3.1: Group accuracy under each delegation mechanism as an increasing number of voters participate directly. Direct accuracy is unaffected by the x-axis. Note that annealing results are shown only for networks with 100 voters.

Exponential - $\forall v_i : q_i \sim Exp(2)$

Discussion

Figure 3.1 shows clearly that whenever the fraction of active voters is low (corresponding to the left side of the plots), the alternative voting systems are able to effectively improve group accuracy. This effect hold in all cases but is amplified in settings with more voters.

Interestingly, whenever more than half of the voters delegate (i.e., the left half of the figure) the accuracy achieved by Random Better is very close to that of Max delegations. This means that even though Random Better only requires nonactive voters find a delegate who is more competent than themselves, it still is able to elicit the "experts" of the network. Observe also that both Random Better and Max delegations achieve group accuracy very close to that of the simulated annealing solution. Moreover, all methods are significantly better than direct democracy. Lastly, note that due to the asymmetry of Exponential distribution the differences described above are magnified. With Uniform and Gaussian competence distributions, roughly an equal number of voters have competence above and below 0.5. Under Exponentially distributed competencies, many more voters have accuracy below 0.5 than above. In accordance with Condorcet's Jury Theorem, this leads to a group accuracy under direct democracy that approaches 0 as the number of vot-

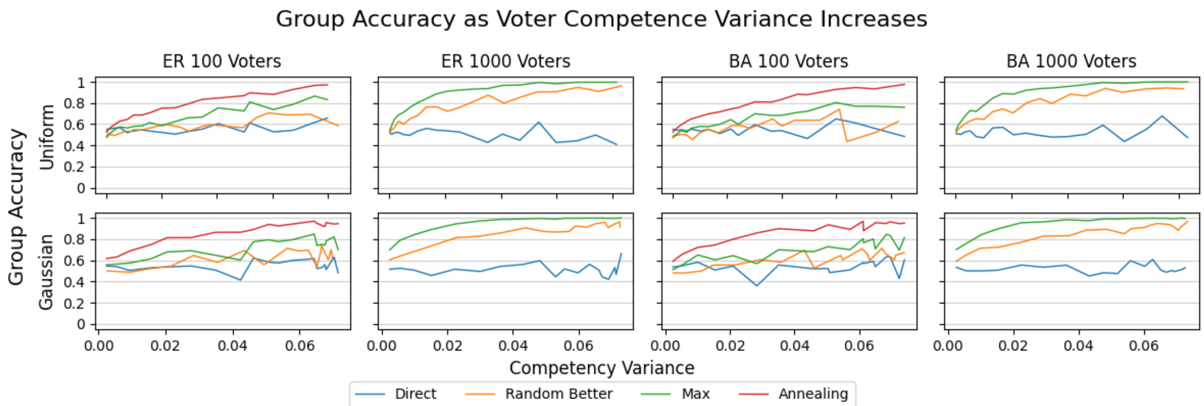


Figure 3.2: Group accuracy as the variance within voter competence increases while maintaining a mean of 0.5. As variance increases, so too does the benefit to group accuracy from LD. Note that annealing results are shown only for networks with 100 voters.

ers increases. However, this also means that some voters have extremely high competence. Liquid democracy is able to effectively transfer weight to these highly competent voters.

Main takeaway from this experiment: Delegation can have a large positive effect on group accuracy across network types and competency distributions.

3.4.2 Increased Variance in Voter Competence

We consider voters with high confidence to be analogous to "experts." We now ask whether the presence of many experts affects the benefits of delegation. We answer this question by changing the variance of voter competence distributions. When there is a higher variance for the same mean value, there are both more experts and more "non-experts." Figure 3.2 shows the accuracy of each voting method averaged over 10 trials. As in Figure 3.1, nodes in each network have an average of 20 neighbours and an average competence of 0.5. However, here we hold constant the chance of being a representative at 90% and increase the variance in each voter's competence. We consider 20 Uniform distributions with (upper, lower) bounds evenly-spaced between $U(0.475, 0.525)$ to $U(0, 1)$, and 20 Gaussian distributions with σ evenly spaced between 0.05 and 0.5. As the Exponential distribution accepts only a single parameter – which affects both the mean and variance – in this experiment we consider only the Uniform and the Gaussian distributions.

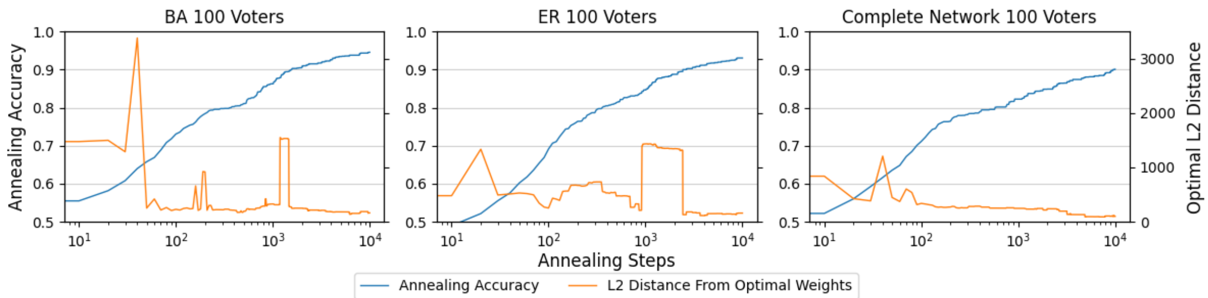


Figure 3.3: Group Accuracy (blue) during the simulated annealing process. As simulated annealing proceeds, the weight distribution among representative approaches the optimal weight distribution as shown by Grofman et al. [113]. The L2 distance between these two distributions is shown in orange.

Discussion

While somewhat noisy, the benefit of "experts" arising from high competence variance is clear in Figure 3.2. We see that in each method, other than Direct voting, group accuracy increases as variance increases. This effect is noticeable despite only a small fraction of voters not participating directly. This ability to utilize "expertise" appears key to the ability of delegative methods to increase group accuracy over Direct voting.

Main takeaway from this experiment: Delegation allows "expert" voters to more than make up for (in terms of group accuracy) voters with very low competence.

3.4.3 Optimizing Liquid Democracy with Simulated Annealing

We now explore the relationship between voter weights during the optimization of delegations and the optimal weight distribution for independent voters in an epistemic setting. Grofman et al. discuss the finding that, in a weighted majority voting system where voters have independent competence values and vote independently, each v_i should have a weight proportional to $w_i \propto \log\frac{q_i}{1-q_i}$[113]. To determine whether simulated annealing approaches weights similar to optimal weights, we use the following process:

1. Calculate the normalized *optimal* weight for each active voter, $\frac{1}{n}\log(\frac{q_i}{1-q_i})\ \forall v_i \in W^+(V)$. Sort the results in descending order and label the resulting vector as w^{opt}.
2. Calculate the normalized *actual* weight for each active voter, $\frac{1}{n}w_i\ \forall v_i \in W^+(V)$. Sort the results in descending order and label the resulting vector as w^*.

3. Finally, calculate the L2 distance between each of the two weight vectors calculated above, $L2 = \sqrt{\sum_{v_i \in W^+(V)} (w_i^{opt} - w_i^*)^2}$.

We calculate the L2 distance between actual and optimal weights using the above procedure after every 10 steps of simulated annealing. The average group accuracy and L2 distance over 10000 steps of simulated annealing is shown in Figure 3.3 which reports the average of 10 trials of this procedure. The figure shows BA and ER networks with an average degree of 20 as well as a complete network, all networks have 100 voters and are regenerated in each trial. Voters have competencies sampled from $N(0.5, 0.2)$.

Discussion

Each network type shows broadly similar results. As time proceeds, simulated annealing tends to approach the optimal delegations. Interestingly, annealing does not monotonically reduce L2 distance. While we hypothesize that high accuracy delegations with low L2 distance from optimal weight distributions do always exist, this experiment shows that there are also delegations which have high accuracy and *do not* have a low L2 distance (e.g. the region in the center plot of Figure 3.3 where the orange line increases beginning at roughly 1000 steps into annealing).

Main takeaway from this experiment: The optimal weights discussed by Grofman et al. naturally arise during the optimization of delegation. Future delegation methods could attempt to establish weights similar to these directly in order to improve accuracy.

3.5 Experiments: The Benefit of Viscosity

In Section 3.4, simulated annealing was able to identify delegation graphs with much higher group accuracy than either delegation method. While simulated annealing is useful, it does not provide a delegation mechanism under which we can explore the dynamics of group accuracy. It does; however, suggest the potential for new delegation mechanisms which improve upon Max and Random Better delegations. To that end, we have developed two new delegation mechanisms – called Proportional Better and Proportional Weighted – which we explore here in conjunction with the addition of viscosity. In this section we show the benefit of these additional delegation methods and argue that viscosity should be an integral component of liquid democracy. This section contains work published at IJCAI 2024 [9].

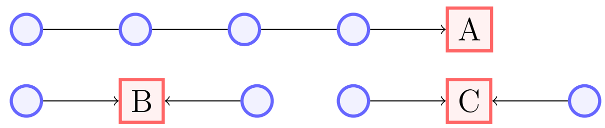


Figure 3.4: A Stars and Chains delegation graph with $s = 2$ star components, with $n_s = 3$, and $c = 1$ chain component with a size of $n_c = 5$. Each node represents a single voters; blue nodes delegate while red nodes vote directly.

Except where otherwise stated (i.e. Figure 3.6) experiments in this section use a more limited variation of parameters, focused on settings where voters are likely to delegate. We set each voter's chance of delegating at 80%. We use 100 voters, ER networks have an connection probability of $p = 0.1$ while BA networks use $m = 10$. Note that this leads to a different average degree between network types, however results remain broadly the same across network types. Each experiment samples competence values from each distribution at 9 different mean values spaced evenly between 0.1 and 0.9:

- Uniform distribution: $U(a,\ a + 0.2)$ for $a \in \{0, 0.1, ..., 0.9\}$.
- Gaussian distribution: $\mathcal{N}(\mu,\ 0.05)$ for $\mu \in \{0.1, 0.2, ..., 0.9\}$.
- Exponential distribution with $\lambda \in \{0.1, 0.2, ...0.9\}$.

3.5.1 Stars and Chains Delegation Graphs

In order to demonstrate the properties of viscosity, we develop a family of delegation graphs.

> **Definition 3.5.1.** *A* ***Stars and Chains*** *(SC) delegation graph is composed of s small star components each with $n_s - 1$ delegators and 1 representative, and c large chains with $n_c - 1$ delegators. Representative in an SC graph are located at the center of the stars and at one end of the chains. Representative in star and chain components have competence c_{comp} and s_{comp}, respectively.*

For our purposes, we typically consider only graphs where each chain component has more voters than are in all star components combined, that is $n_c > sn_s$ (typically we have considered settings where $n_c = sn_s + 1$ but this is not required). Figure 3.4 visualizes a simple SC delegation graph.

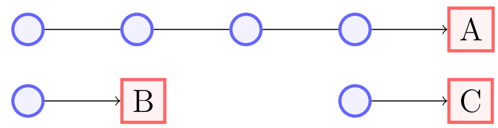


Figure 3.5: A Stars and Chains delegation graph with $s = 2$ star components, with $n_s = 2$, and $c = 1$ chain component with a size of $n_c = 5$. Each node represents a single voters; blue nodes delegate while red nodes vote directly.

3.5.2 Calculating and Reporting α^*

We denote the value of α which maximizes group accuracy under viscous democracy as α^*. Group accuracy is defined in terms of sets of voters able to form a majority of weight. Thus, accuracy is a piece-wise function in relation to α and there may be many possible values of α^*.

As no previous work has considered the effect of viscosity in epistemic settings we must first demonstrate whether it is *possible* for viscosity to change (and, ideally, improve!) group accuracy. As a result, we are primarily interested in differentiating settings where $\alpha^* = 1$ and those where $\alpha^* < 1$. To this end, we do not report a range of α^* values but rather the maximum value of α^* as determined experimentally. When searching for α^* we run 30 elections (each estimating accuracy from 1000 iterations of Monte Carlo sampling as in Algorithm 1) for each value of $\alpha \in \mathcal{A} = \{0, 0.1, 0.2, ..., 1\}$ and record the mean and standard deviation of group accuracy, $(\mu_\alpha, \sigma_\alpha)$.

If there exists some α such that $\mu_\alpha - \sigma_\alpha > \mu_\beta + \sigma_\beta \ \forall \ \beta \in \mathcal{A} \setminus \{\alpha\}$ then this becomes α^*. If such a value does not exist, we say that $\alpha^* = 1$. In this way we quite conservatively report α^* as being below 1 only when group accuracy is strictly improved by the lower viscosity.

3.5.3 Potential Improvement from Viscosity

The initial question we must answer is simply: Can $\alpha < 1$ improve group accuracy? In fact, there exist settings in which α^* is 0, 1, or in the range $(0, 1)$. We now show an example for each of these values of α^*.

Liquid Democracy: $\alpha^* = 1$

We first show that there exist settings in which liquid democracy (i.e., where $\alpha^* = 1$) leads to optimal group accuracy. Below we present an example where increasing α always weakly increases accuracy.

Example 3.5.1. *Consider the delegation graph in Figure 3.5 with one chain and two stars*[2]*. Let* $q_A = 0.9$ *and* $q_B = q_C = 0.4$*. When* $\alpha = 1$*,* $w_A = 5$ *and the outcome depends entirely on* A*'s vote and, thus, group accuracy is* 0.9*. When* $\alpha < 0.848$*,* $w_A < w_B + w_C$ *and accuracy drops to approximately* 0.772*.*

Direct Democracy: $\alpha^* = 0$

We now show, for the topology of the delegation graph, a setting of competencies where lower values of α are weakly superior to higher values.

Example 3.5.2. *Consider again the SC delegation graph of Figure 3.5. Let* $q_A = 0.9$ *and* $q_B = q_C = 0.8$*. As before, when* $\alpha = 1$*, group accuracy depends solely on* A *and is* 0.9*. In this case; however, decreasing* α *to the pivotal* $\alpha = 0.848$ *causes accuracy to increase due to the relative strength of voters* B *and* C*. When* $\alpha < 0.848$*, accuracy increases to approximately* 0.928*.*

Viscous Democracy: $0 < \alpha^* < 1$

In contrast to the previous examples we now present a simple example where accuracy is maximized when $0 < \alpha < 1$.

Example 3.5.3. *Consider a SC delegation graph with* $s = 6$*,* $s_{comp} = 0.8$*,* $n_s = 5$*,* $c = 3$*,* $c_{comp} = 0.5$*,* $n_c = 30$*. That is,* 6 *star components each with size* 5 *and* 3 *chain components each with size* 30*.*

When $0 \leq \alpha < 0.25$*, the group accuracy is roughly* 0.91*. However, the weight of star components increases more quickly than that of chain components as* α *increases. For* $0.25 \leq \alpha \leq 0.5$*, accuracy becomes approximately* 0.94*. For higher values of* α*, the larger chain components begin to dominate and accuracy decreases. Here,* $\alpha^* \in [0.25, 0.5]$*.*

Figure 3.6 showcases a family in which non-extreme alpha (i.e., $0 < \alpha^* < 1$*) is optimal.*

3.5.4 Optimal Amount of Viscosity

While we can easily demonstrate that there exist situations in which $\alpha^* < 1$, a much more interesting result arises when considering how often a non-extreme value of α is optimal in more plausibly realistic settings. We now show that placing voters with randomly chosen competencies on larger networks frequently results in an optimal value of α below 1.

[2] We do note that, in Figure 3.5, the stars are equivalent to chains. In future examples we use the SC family more generally.

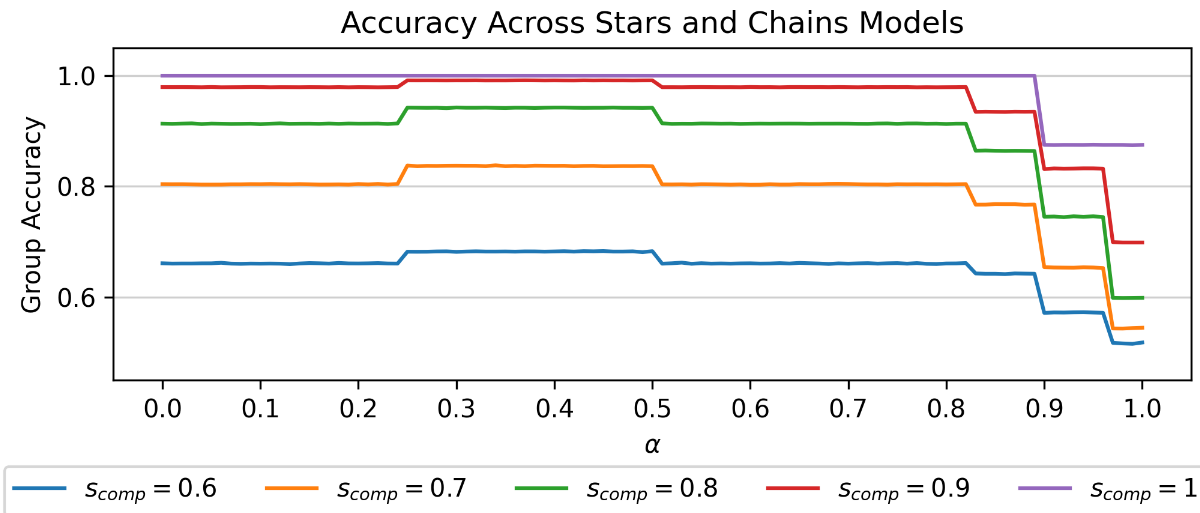


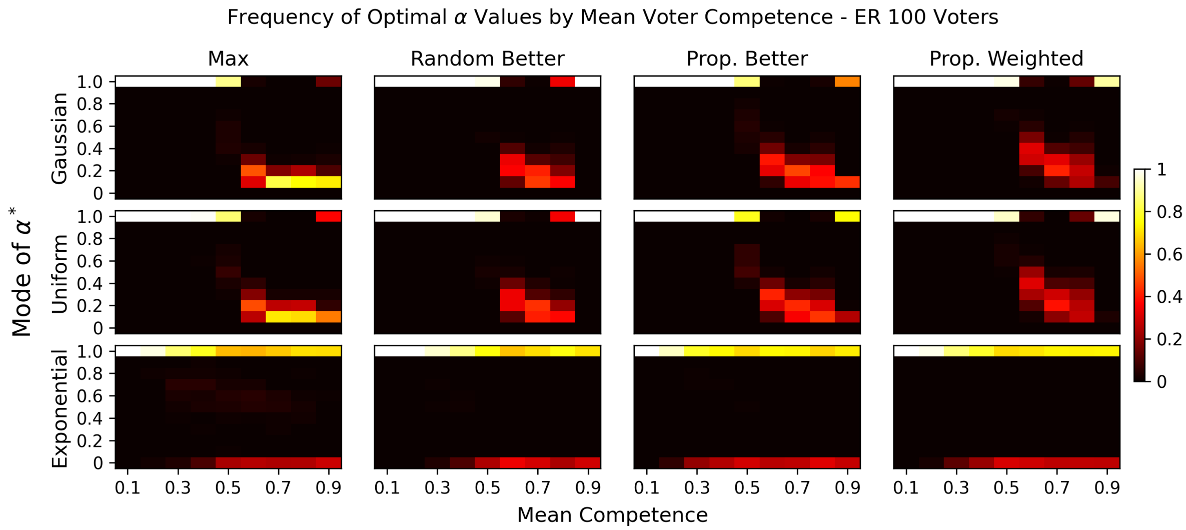
Figure 3.6: Accuracy in Stars and Chains delegation graphs as α varies from 0 to 1. Each series varies s_{comp} and sets $s = 6$ $n_s = 5$, $c = 3$, $c_{\text{comp}} = 0.5$, $n_c = 30$ As α changes, the sets of representatives able to form a majority of weight shifts in a piecewise manner. Optimal α occurs in $[0.25, 0.5]$

We show in Figure 3.7 an experiment in which we identify the mean value of α^* for some set of parameter values. For each combination of parameter values we perform 300 trials and report the fraction of trials in which each value of α maximizes group accuracy.

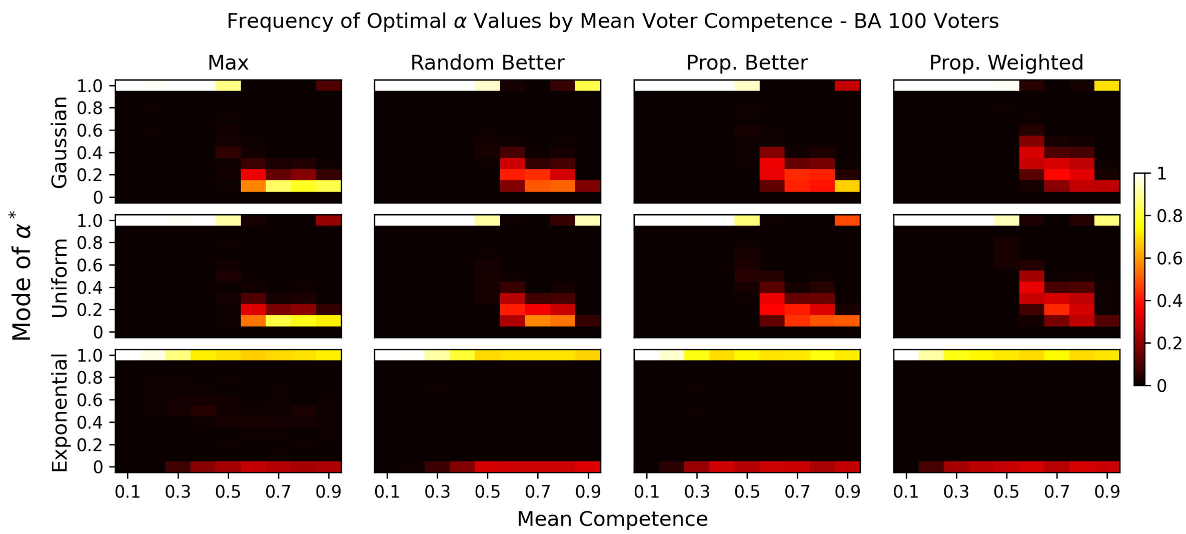
Discussion

Once again, we see that results are extremely similar across network types. In both ER and BA networks, clear trends emerge. When mean voter competence is below 0.5 Uniform and Gaussian competence distributions see no benefit from viscosity, and exponential distributions are very infrequently better off with $\alpha < 1$. However, when mean voter competence is above 0.5 but below 1, we see that α^* is almost always quite low.

A low value of α works to reduce massive weight centralization. Intuitively, from Condorcet's Jury Theorem, having more voters involved in decision-making is better (if those voters have better-than-random competence). When delegation centralizes a majority of weight among only a few voters, a low value of α works to undo that centralization by disproportionately reducing the weight of voters receiving many delegations. Low values of α^* are most prevalent under the Max delegation mechanism, which has the most centralization. Similarly, α^* under Proportional Weighted delegations is frequently below 1



(a) Erdős–Rényi networks with $p = 0.1$.



(b) Barabási–Albert networks with $m = 10$.

Figure 3.7: Distribution of α^* across competence distributions and delegation mechanisms. We performed 300 trials for each mean voter competence value. Each cell shows the fraction of trials for that mean competence value in which the corresponding α value maximizes group accuracy.

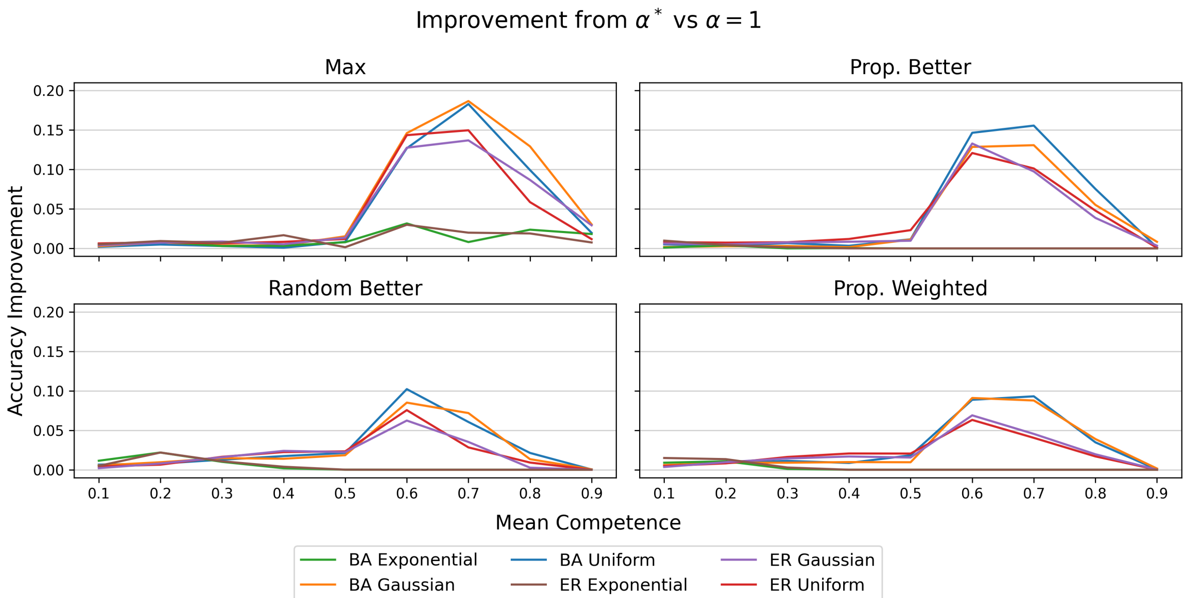


Figure 3.8: Accuracy improvement from using optimal viscosity vs no viscosity ($\alpha = 1$) for voters on BA and ER networks. Each subplot shows the amount of improvement for a single delegation mechanism. Note that values are absolute; an improvement of 0.1 indicates that if group accuracy is 0.6 with $\alpha = 1$, then it becomes 0.7 with $\alpha = \alpha^*$. Results are averaged over 30 trials with 100 voters per network.

but not does not tend to be as low as other methods; Proportional Weighted delegations are specifically chosen in order to decrease the chance of excessive weight centralization.

Finally, when voters reach a very high level of group competence we see that α^* returns to 1. For an intuitive explanation of this, imagine a group of voters that all have competence of $q = 1$. The group accuracy remains identical regardless of weight centralization.

Main takeaway from this experiment: Viscosity improves group accuracy when voters suffer from excessive weight centralization.

3.5.5 Magnitude of Benefit from Viscosity

We have demonstrated that viscosity very often leads to an improvement in group accuracy. However, we have not yet shown how much of an improvement is common. In Figure 3.8 we show that viscosity is not only beneficial but significantly beneficial. This experiment shows the difference in group accuracy between liquid democracy ($\alpha = 1$) and optimized viscous democracy ($\alpha = \alpha^*$). Each plot shows results for all network types and competence distributions using a single delegation mechanism.

Discussion

Just as previously discussed, the delegation mechanism which leads to highest weight centralization (Max) benefits much more from optimal viscosity than Proportional Weighted, which inherently reduces weight centralization. However, for some competence values, viscosity provide a minimum increase in accuracy of nearly 0.1 across all delegation mechanisms. As in most settings, voters with exponential competencies do not significantly benefit from delegation while other competence distributions and network types experience similar benefits.

Main takeaway from this experiment: Very significant improvements to group accuracy are possible with optimal viscosity.

3.6 Conclusions

This chapter shows that liquid and viscous democracy are able to very significantly improve the ability of voters to correctly identify ground truth. Across a wide range of settings, we have shown that basic liquid democracy outperforms direct democracy even when voters have minimal information about the competence of their voters, as with Random Better delegations.

Subsequently, we generalized our model of liquid democracy to the viscous setting. Here we first demonstrated the potential benefit from viscosity, then showed it empirically. Our experiments demonstrate both the prevalence and significance of viscosity's potential positive effect on group accuracy.

Throughout this chapter our results highlight several key observations:

- Liquid and viscous democracy improve accuracy over direct democracy across a wide range of settings.
- When voters have highly variable competence – some voters are experts, while many have low competence – as in the exponential distribution, delegation has an even more pronounced improvement to group accuracy.

- Simple delegation methods such as Random Better can be adapted to avoid weight centralization that reduces accuracy.

- Low viscosity disproportionately reduces the weight of the heaviest voters thereby reducing weight centralization and increasing accuracy.

Our key findings are purely experimental. Theoretical questions about the efficacy of liquid democracy are quite difficult to approach exactly due to the wide range of behaviours voters might exhibit – that is, in a general model of LD like ours, any delegation can reach any voter; this makes reasoning about possibilities and impossibilities difficult. Future theoretical work faces the challenge of identifying which limitations must be placed on the model to support compelling results. In this chapter we demonstrated that an experimental approach is able to provide clear conclusions useful both for guiding future research and for practitioners using liquid democracy to improve performance on practical tasks.

Chapter 4

Ensemble Learning with Liquid Democracy

This chapter extends many of the ideas presented in Chapter 3 to a machine learning task, where classifiers take on the role of voters. Previously we showed that liquid democracy improves the ability of voters to identify a ground truth in a controlled setting meant to resemble human voting structures. We now apply the same ideas to ensemble learning. This setting provides a real-world task on which LD may realize a concrete improvement over the performance of existing ensemble methods in terms of both accuracy and training cost.

4.1 Introduction

Training machine learning systems consumes increasingly large amounts of data and compute. In pursuit of ever-growing performance, model sizes are constantly increasing and more data is collected while the financial and environmental cost of machine learning rises [172, 200]. In this chapter, we demonstrate the use of liquid democracy as a means of reducing training costs while improving or maintaining accuracy.

Specifically, we explore the use of the liquid democracy paradigm as a means to enhance ensemble learning. Rather than fully training all classifiers in an ensemble, we aim to reduce training costs by having weak classifiers delegate to stronger classifiers during the course of training. This effectively "removes" the weak classifiers while redistributing their weight to the stronger classifiers. The goals of this chapter are as follows,

- Demonstrate that delegative principles can be applied to ensemble training by extending the framework of liquid democracy to an incremental learning.
- Use delegation to reduce the cost, in terms of data or compute, required to train an ensemble by cutting short the training of classifiers that delegate.
- Identify delegation mechanisms and parameters under which the accuracy of ensembles is improved.

This chapter develops a system for delegative ensemble pruning in an incremental learning framework which operates over many repeated rounds of refining estimates of classifier accuracies and removing weak classifiers by having them delegate to more accurate classifiers. In Section 4.2 we describe the connection between ensemble learning and epistemic voting upon which this application is built by extending the model of Chapter 3 to a setting in which voters consider multiple issues. We then show analytically in Section 4.3 that delegation in this machine learning setting is almost always able to improve the accuracy of a given ensemble. Our experiments in Section 4.4 show that – while a single round of delegation does not dramatically alter ensemble performance – repeated rounds of delegation throughout training significantly reduce training cost and can improve accuracy. Ultimately we are able to identify parameters that reliably reduce training cost of an ensemble by up to a factor of 30 without degrading accuracy when compared with a naively trained ensemble.

4.2 Model

Here we refresh the background concepts and notation discussed in Chapter 2 that are most relevant in this chapter. The primary additions to our previously described model are an extension to our notation that encompasses a multi-issue epistemic setting, and the inclusion of a time dimension – voters may delegate over many rounds (or *increments*).

As in Chapter 3, we apply liquid democracy to a ground truth setting with alternatives $A = \{a^+, a^-\}$, where a^+ is, objectively, the correct outcome There are n voters $V = \{v_1, ..., v_n\}$ and each $v_i \in V$ receives a noisy, independent signal as to which is the correct alternative. We refer to this as their *competency* (or, interchangeably, the voter's *accuracy*), or their probability of viewing a^+ as the best alternative, and denote a voter's competency as $q_i \in [0, 1]$.

Voters may act as a representative and directly cast a vote by revealing to the voting mechanism their favoured alternative. Here we introduce a cost, $c_i \geq 0$, incurred when v_i

Symbol	Meaning
V	Set of voters (instantiated as classifiers).
n	Total number of voters (a.k.a. classifiers) in V.
n_t^{final}	The number of voters that are not delegating after the t^{th} increment of learning.
$q_{i,t}$	Competence/training accuracy of voter i on the t^{th} increment of learning.
w_i	Weight of voter i.
c	Size of the dataset being trained upon.
d	Delegation function. $d : V \to V$ indicates to which voter i delegates their vote.
g	Delegator selection function. $g : 2^V \to 2^V$ selects which voters will begin to delegate in the current increment.
ρ	Delegation probability function. $\rho : V \times V \to [0, 1]$ gives the probability that the given representative will delegate to the voter given as the second argument.
$N^+(v_i)$	The set of voters with competence higher than v_i.

Table 4.1: Overview of notation most important throughout this chapter.

participates directly which is meant to capture the effort required on behalf of the voter. Note that this cost depends upon the setting and may be zero. Alternatively, a voter may *delegate* their vote to another voter and incur no cost.

If voter v_i is chosen to delegate, they use some delegation function (described in Section 4.2.3) to select a $v_j \in V$ to whom v_i delegates. We denote by $d : V \to V$ the delegation function; $d(v_i) = v_j$ indicates v_i delegates to v_j. $d^*(v_i)$ is the repeated application of $d(v_i)$ until a fixed point is reached and self-delegation is used to indicate that a voter is a representative. The weight of each representative is equal to the number of delegations they receive, including their own:

$$w_i = \begin{cases} 0 & \text{if } d^*(v_i) \neq v_i \\ |\{v_j \mid d^*(v_j) = v_i \ \forall \ v_j \in V\}| & \text{otherwise} \end{cases}$$

After delegation each representative v_i is assigned the preference order $a^+ \succ a^-$ with probability q_i, and has the preference order $a^- \succ a^+$ with probability $1 - q_i$. We commonly say that v_i "votes for" a^+ with probability q_i. We use weighted plurality voting to select a winner:

$$\mathcal{F}^{\text{WP}}(\mathcal{P}) = \arg\max_c \sum_{\{P_i \in \mathcal{P} | P_i[1] = c\}} w_i$$

4.2.1 Extension to Multi-Issue Voting and Machine Learning

We now introduce aspects of our model unique to this chapter. The model we described thus far applies to voters choosing between a single pair of alternatives. In Section 2.4 we draw a parallel between epistemic voting and classification tasks in machine learning: A model making a classification prediction can be viewed as the instantiation of the more abstract concept of a voter making a prediction; many voters are then equivalent to an ensemble of classifiers. In machine learning, prediction tasks are rarely performed on a single example but, rather, predictions are made for an entire dataset. In social choice terms, this corresponds to voting on many issues where each issue is a single election. For the rest of this chapter, we consider the setting where voters *are* classifiers and we use the terms interchangeably. Here we extend our model to encompass the machine learning setting, in the next section we describe the algorithm we use for training classifiers within this setting.

Rather than a single pair of alternatives, we now consider c pairs of alternatives $A_c = \{(a_1^+, a_1^-), ...(a_c^+, a_c^-)\}$, corresponding to the c *examples* within a two-class dataset. Unless stated annotated otherwise, we specifically assume that A_c is a *test* set; voters (i.e. classifiers) are making predictions on this set after having already gone through a training phase. This training phase is used to inform the voter's competence. Each v_i continues to be associated with a competence level q_i; this is the voters *training accuracy*. That is, q_i is the fraction of examples in some training set that v_i classified correctly. This can be thought of as a weak estimate of each voters true accuracy on the entire dataset.

We extend liquid democracy to this multi-issue machine learning setting in the most straightforward way possible. Each voter continues to take one of two actions: (1) They vote as a representative by making a classification predictions on *all* c examples within the test set, or (2) the voter *delegates* and makes no classification predictions.

4.2.2 Training with Incremental Delegation

In addition to the extension from single elections to elections across entire datasets we also add a dimension of time. Rather than classifiers going through a single round of training and delegation, we consider several such rounds, called *increments*. Each increment of training provides a refinement of the weak estimate of each voter's competence while each the delegation in each increment is an opportunity to have additional classifiers with poor performance delegate to more competent classifiers.

The number of voter that do not delegate in increment t is referred to as n_t^{final}, or simply n^{final} when referring to the number remaining after all delegation is complete. We calculate

n_t^{final} based on a parameter r, the proportion of representatives that *do not* delegate at each increment. That is, a $1 - r$ fraction of remaining representatives begin to delegate at each increment, until a minimum of n^{final} representatives remain or all data has been trained on. Typically, the representatives with the lowest competence are chosen to delegate at each increment (the exception to this is when using the *diverse* delegation mechanism).

Algorithm 3 outlines our specific delegation and training procedure in detail. The algorithm is given classifiers V, details of how much delegation to perform (r, n^{final} and T), and training data split into an evenly sized slice for each increment. The algorithm is based on *incremental learning* – training in discrete steps that each build upon the previous training. We denote as $\mathbf{q}_{t,i}$ the mean competence of v_i over all increments up to and including t and $q_{t,i}$ refers to the competence of v_i on the training data in increment t. $\mathbf{w}$ refers to the vector containing the weight of each individual voter.

During increment t of training, each voter trains on a new slice of the training data and their accuracy estimate $\mathbf{q}_{t,i}$ is updated (lines 5-10). The function `select_pruned_clfs` uses information about voter competence and weight to choose which voters should delegate (line 11). Each new delegator then makes their delegation by transferring their weight to another voter (lines 15-20). Once all delegation is complete, the remaining voters are fully fit on the data (lines 23-27). The functions `select_pruned_clfs` and `transfer_weight` correspond to the two components of delegation: selecting classifiers to delegate, and determining which delegations occur; the functions are described in Section 4.2.3. Note that delegation is permanent in this algorithm: Once a voter has delegated they are not trained any further, they make no predictions, and they never change their delegation.

4.2.3 Delegation Mechanisms

Through this chapter we explore each of the delegation mechanisms listed in Table 4.2. We refer to Chapter 2 for a formal definition of, and more discussion surrounding, delegation mechanisms. **Delegator selection functions** implement the `select_pruned_clfs()` method in Algorithm 3 while **delegation probability functions** provides the functionality needed by the `transfer_weight()`. We do not consider an underlying social network in this chapter; which we treat as equivalent to voters being placed on a complete network. Thus, our standard neighbour functions return *all* other voters, and *all more competent* voters, respectively: $N(v_i) = \{v_j \mid v_j \in V \setminus \{v_i\}\}$, and the more competent neighbours of a voter: $N^+(v_i) = \{v_j \mid v_j \in V \text{ and } q_{j,t} > q_{i,t}\}$.

We highlight the Random-R mechanism – developed before we formalized our current framework for delegation mechanisms [12] – removes random voters from the ensemble, but creates no delegations; the weight of those voters is simply removed from the election.

Algorithm 3 Training and Pruning Algorithm. `fit` and `partial_fit` methods refer to methods in the sklearn library for each model type [175]. `select_pruned_clfs` and `transfer_weight` are defined in Section 4.2.3.

Input: $V, r, n^{\text{final}}, T, X^{\text{train}}, Y^{\text{train}}$

$\mathbf{w} \leftarrow [1]^n$
$X^{\text{train}}, Y^{\text{train}} \leftarrow \{X_1^{\text{train}}, Y_1^{\text{train}}, ..., X_T^{\text{train}}, Y_T^{\text{train}}\}$
for $t \in T$ **do**
    {`Train each representative on the current increment.`}
    for $v_i \in V$ **do**
        if $w_i \neq 0$ **then**
            `partial_fit`(v_i, X_t^{train}, Y_t^{train})
            $\mathbf{q}_{t,i} \leftarrow$ `mean`($\{q_{s,i}\ \forall s \leq t\}$)
        end if
    end for
    {`Select new delegators.`}
    $V_t^{\text{remove}} \leftarrow$ `select_pruned_clfs`($V, \mathbf{q}, \mathbf{w}, r$)
    {`End partial training if delegation is complete.`}
    if $V_t^{\text{remove}} = \emptyset$ **then**
        `break`
    end if
    {`Perform delegations.`}
    for $v_i \in V_t^{\text{remove}}$ **do**
        if $|\{w \mid w \neq 0 \ \forall\ w \in \mathbf{w}\}| \leq n^{\text{final}}$ **then**
            `break`
        end if
        `transfer_weight`($w_i, V, \mathbf{w}, \mathbf{q}$)
        $w_i \leftarrow 0$
    end for
end for
for $v_i \in V$ **do**
    if $w_i \neq 0$ **then**
        `fit`(v_i, X^{train}, Y^{train})
    end if
end for

Delegator Selection Functions

We explore three delegator selection functions; two baselines g^{random} and g^{direct} as well as two more intelligent functions g^{worst} and g^{diverse}. We define g^{direct} and g^{worst} in Section 3.3 while g^{random} and g^{diverse} are defined below. The Random delegation mechanism uses g^{random} and the Direct mechanism uses g^{direct}; all other delegation mechanism use g^{worst}.

Definition 4.2.1. *Delegator Selection Function $g^{\text{random}}(V)$ selects n_t^{final} representa-*

Delegation Mechanism	Selection Function	Probability Function
Direct	g^{direct}	ρ^{direct}
Random-R	g^{random}	–
Random	g^{random}	ρ^{random}
Uniform	g^{worst}	ρ^{uniform}
Random Better	g^{worst}	$\rho^{\text{rand\_better}}$
Proportional Better	g^{worst}	$\rho^{\text{prop\_better}}$
Proportional Weighted	g^{worst}	$\rho^{\text{prop\_weighted}}$
Diverse	g^{diverse}	ρ^{diverse}

Table 4.2: The delegation selection and probability functions that define each delegation mechanism.

tives uniformly at random.

Definition 4.2.2. *Delegator Selection Function $g^{\text{diverse}}(V, d)$ identifies the n_t^{final} pairs of voters which are most similar according to one of the diversity metrics discussed in Section 4.4.3 and where the least accurate member is a representative. $g^{\text{diverse}}(V, d)$ selects the least accurate member of each such pair.*

Delegation Probability Functions

Several of the delegation probability functions listed in Table 4.2 are defined in Section 3.3; the functions new in this chapter are defined below.

Definition 4.2.3. *The **Random** Delegation Probability Function selects delegates uniformly at random for new delegators.*

$$\rho^{random}(v_i, v_j) = \frac{1}{n}$$

Definition 4.2.4. *The **Uniform** Delegation Probability Function distributes delegated weight as evenly as possible among voters, breaking ties in favour of the most competent voter. Let $h(v_i) = \{q_{j,t} \mid w_{j,t} \leq w_{k,t} \ \forall \ v_j, v_k \in N^+(v_i)\}$ denote the competencies of voters in $N^+(v_i)$ with minimal weight. The least accurate voters delegate their weight to the most accurate voter in $h(v_i)$, with ties broken randomly.*

$$\rho^{\text{uniform}}(v_i, v_j) = \begin{cases} 1 & q_{j,t} = \max\, q_t \in h(v_i) \\ 0 & \textit{otherwise} \end{cases}$$

Definition 4.2.5. *The* ***Diverse*** *Delegation Probability Function returns delegation probabilities aimed at maintaining high diversity within an ensemble. Let* $S = \{v_a, v_b \in G(V) \times V | q_a \leq q_b\}$ *be the set of pairs of voters where the first voter in the pair is a representative and has lower accuracy than the second voter in the pair. Note that, as a representative, the first voter of each pair in* S *is always eligible to delegate to the second voter in the pair.* ρ^{diverse} *selects the pair of voters in* S *which is most similar to each other on some chosen diversity function* `div` *and has the less accurate voter delegate to the more accurate.*

$$\rho^{\text{diverse}}(v_i, v_j) = \begin{cases} 1 & v_i, v_j = \arg\min_{v_a, v_b \in S} \texttt{div}(v_a, v_b) \\ 0 & \textit{otherwise} \end{cases}$$

Other Delegation Approaches

In points through our analysis and experiments we also consider two other approaches to delegation. When we refer to the **Best** delegations, this means that a brute force search of all possible delegation structures is performed in order to find the delegations which maximize group accuracy. n^{final} is ignored when finding the best delegations.

4.3 Delegation Analysis

Before presenting our experimental results, we analyze two aspects of our model that affect how we interpret our results. First, we ask how likely delegation is to improve accuracy by finding an upper bound on the probability that the initial delegation in an ensemble will reduce accuracy. Additionally, we define a measure of the cost of training an ensemble in terms of the number of times each piece of data is used during training.

4.3.1 Group Accuracy Improvement from Delegation

To give some idea of the extent to which delegations are "safe" and unlikely to reduce group accuracy we consider the frequency with which a single delegation in our setting might improve group accuracy. Specifically, we focus on one delegation in a setting where there are not yet any existing delegations, i.e. the first delegation being made. We do this by analyzing the number of possible initial ways that voters might classify a given dataset and show an upper bound on the number of these states where every delegation reduces group accuracy. We show that these "harmful" states exist but they are vanishingly small in frequency and that delegation improves or does not change accuracy in the vast majority of states.

First we must re-emphasize the parallel between sets of voters and ensembles: Voters voting over multiple issues is equivalent to an ensemble with classifiers making predictions on a dataset. Consider the votes of a set of voters with no delegations. These can be treated as an $n \times c$ binary matrix P where p_{ij} is 1 if voter i voted correctly on the j^{th} example and 0 otherwise, respectively shown with a green check mark and a red x in Figure 4.1. Any column that sums to $\lceil \frac{n}{2} \rceil$ or more indicates the corresponding issue is voted on correctly in aggregate. Alternatively, we can see Figure 4.1 as an ensemble of n voters on a dataset of size c.

An example j is called *pivotal* if $\sum_{i=1}^{n} p_{ij} = \lceil \frac{n}{2} \rceil$. That is, if it is correctly classified by the minimum number of voters required to classify it correctly. Any voter that is correct on a pivotal example is said to be a *pivotal voter* on that example, and may be pivotal on several examples. Similarly, if a voter is incorrect on example j where $\sum_{i=1}^{n} p_{ij} = \lceil \frac{n}{2} \rceil - 1$ they are considered an *incorrect pivotal voter*. We can now begin to establish an upper bound on the number of states in which any individual delegation would result in a decrease in accuracy (and thus, well-informed and well-meaning voters would make no delegations).

Lemma 4.3.1. *If every single delegation reduces group accuracy then each voter must be pivotal on at least one example.*

Proof. This can be proven very straightforwardly by contradiction: If there exists a single voter who is not pivotal on any example, they can delegate to any other voter without causing any examples to change from being classified correctly to being classified incorrectly. □

Note that this lemma does not hold in reverse. If every voter is pivotal on at least one example, it is not the case that delegation must reduce group accuracy. For an example of this, observe that v_1 may delegate to v_2 in both parts of Figure 4.1 without changing group accuracy.

When considering the harmful effects of a single delegation we need only focus on pivotal examples as they are the only correctly classified examples that may become incorrectly classified. Thus the classification decisions on non-pivotal examples are irrelevant

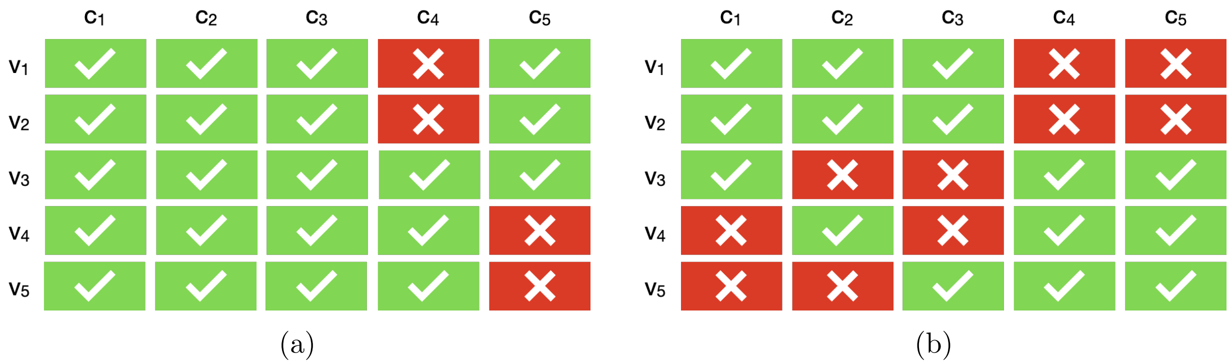


Figure 4.1: Two possible states when an ensemble composed of 5 classifiers predicts the classes of 5 examples. A cell shows whether a particular voter (in rows) is correct or incorrect in classifying each example (in columns). **(a)** All voters are pivotal and only 2 examples (c_4 and c_5) are pivotal. If any non-pivotal examples c_1, c_2, or c_3 were removed all voters in **(a)** would remain pivotal. **(b)** All voters are pivotal and all examples are pivotal.

to this problem and our theorem considers only the states that may occurs within pivotal examples.

Theorem 4.3.1. *In an ensemble with n voters and a dataset of c examples, the total number of ways in which classification decisions can be made on pivotal examples such that every voter is pivotal is $s_{n,c}^{pivotal} = \sum_{c_p=2}^{c} \binom{n}{\lceil \frac{n}{2} \rceil}^{c_p}$. Without any restrictions the same examples could have $s_{n,c}^{total} = \sum_{c_p=2}^{c} 2^{nc_p}$ possible states.*

Proof. During this proof we consider each possible classification outcome as a separate state, see Figure 4.1 for a visualization of 2 states. This can be thought of as a matrix with n rows and c columns where each cell can take only binary values. The cell at position (i, j) refers to whether or not voter v_i classified example j correctly. We refer to voter predictions, and the predictions made on an example as rows and columns respectively.

Say there are c_p pivotal examples in some initial state. In order for each voter to be pivotal at least once, $2 \leq c_p \leq c$. We can obtain an upper-bound estimate on the fraction of states in which delegation is harmful by counting the number of possible states where every voter is pivotal (Lemma 4.3.1 shows that every harmful state meets this condition) and comparing it to the total number of possible outcomes. We count, for some c_p, the number of ways in which c_p columns on n rows can be arranged such that all voters are pivotal. Denote this $s_{n,c_p}^{\text{pivotal}}$ and compare it with the total number of ways to arrange those c_p columns, denoted s_{n,c_p}^{total}.

In practice, we calculate only the ratio of $s^{\text{pivotal}}_{n,c_p}$ and s^{total}_{n,c_p}. The $c - c_p$ columns that are not pivotal have the same number of states in each case so we exclude them from our calculation.

The number of ways to construct a single pivotal column of n rows is $\binom{n}{\lceil \frac{n}{2} \rceil}$. Extended to c_p columns we get $\binom{n}{\lceil \frac{n}{2} \rceil}^{c_p}$. Summing over all possible values of c_p we arrive at a loose upper bound on the total number of possible states where every voter is pivotal on c examples and n voters:

$$s^{\text{pivotal}}_{n,c} = \sum_{c_p=2}^{c} \binom{n}{\lceil \frac{n}{2} \rceil}^{c_p}$$

Whereas, the number of ways to fill in the same columns with no regard for whether or not they are pivotal is simply the number of possible states of an $n \times c_p$ binary matrix, or 2^{nc_p}. Which, summed over all values of c_p becomes,

$$s^{\text{total}}_{n,c} = \sum_{c_p=2}^{c} 2^{nc_p}$$

□

Counting Harmful Delegations

Note that when calculating $s^{\text{pivotal}}_{n,c}$ and $s^{\text{total}}_{n,c}$ we are counting only the (pivotal or total) number of states on c_p columns/examples. Here c_p may be between 2 and c; the remaining $c - c_p$ columns are unrestricted and can be completed in any way for both $s^{\text{pivotal}}_{n,c}$ and $s^{\text{total}}_{n,c}$. This enables us to compare the two values without evaluating the total number of possible states in either case.

Table 4.3 shows this loose upper bound on the proportion of states in which a single delegation is necessarily harmful as n and c grow. With even moderately-sized datasets and ensembles it becomes nearly impossible for delegation to inherently reduce accuracy. Thus, with almost any reasonable delegation mechanism there is strong reason to expect that delegations may improve or, at least, maintain accuracy.

	c				
n	11	21	31	41	51
11	7.71e-08	2.63e-14	8.98e-21	3.07e-27	1.05e-33
21	3.05e-09	5.52e-17	9.99e-25	1.81e-32	3.28e-40
31	4.03e-10	1.16e-18	3.35e-27	9.66e-36	2.78e-44
41	9.23e-11	6.96e-20	5.24e-29	3.95e-38	2.98e-47
51	2.89e-11	7.57e-21	1.98e-30	5.20e-40	1.36e-49

Table 4.3: A loose upper bound on the fraction of states with n voters and a dataset of size c where any single delegation reduces group accuracy.

4.3.2 Data Requirements for Incremental Training

We can calculate the amount of data that must be considered when training an ensemble. This serves as an analogue for compute requirements but is also a means of guiding parameter values (i.e. the answer to "which parameters minimize compute requirements?"). Measuring the exact compute used to train a given ensemble is infeasible[1], rather we measure the amount of data that is seen during training of a model. If a particular example is learned from many times, each time counts toward the cost. This serves as an approximation of compute used during training which allows us to compare between training algorithms and parameter values.

Specifically, we define the "data cost, " δ, of an ensemble as the number of examples that must be used for incremental training during delegation plus the size of the training data multiplied by the number of iterations each classifier takes to be fully fit to the data. We can examine each component of this cost separately.

$$\delta = \delta^{\text{delegation}} + \delta^{\text{training}}$$

Delegation Cost: We define delegation cost as the total number of training examples seen by classifiers across all incremental training that occurs during delegation. When calculating delegation cost we assume that the dataset is of sufficient size to fully delegate (note that this is not always the case with our smallest datasets). The cost depends upon increment size (denoted s; the number of examples trained upon in each increment),

[1]Compute could, for example, be measured in terms of CPU operations (which we simply cannot track on the remote server used to do our training) or energy consumed (even more difficult to track). However, these measures would depend on the processor being used, the dataset, what else the computer is doing, etc. and do not appear meaningful as a basis for comparison.

delegation rate ($1-r$; r denotes the fraction of voters that *do not* delegate in an increment), number of increments (l), initial number of voters (n), and final number of representatives (n^{final}); it is the geometric series,

$$\begin{aligned}\delta^{\text{delegation}} &= \sum_{i=0}^{l} snr^i \\ &= sn\frac{1-r^{l+1}}{1-r}\end{aligned}$$

In fact, the number of increments of delegation can be approximated as a function of n, n^{final}, and r by noting that delegation continues until the number of active voters is equal to n^{final}.

$$nr^l = n^{final}$$

Which gives,

$$l = \frac{log(n^{final}/n)}{log(r)}$$

Combining the formulas above gives a function for calculating delegation cost based on n, n^{final}, and r. We leave out increment size, s, from our calculation as it is a constant factor. Figure 4.2 shows this bound on delegation cost for a range of these parameters. Somewhat surprisingly, there is only mild difference between differing numbers of final representatives at the same delegation rate. This is due to the fact that while there are more delegation increments with a lower value of n^{final}, these additional increments have successively lower delegation cost so do not greatly affect the overall delegation cost.

Training Cost: The cost for training a fully delegated ensemble varies based upon training data size c, number of remaining representatives n^{final}, and ι_i, the number of iterations of training must be performed on each representative v_i. We have,

$$\delta^{\text{training}} = nc\sum \iota_i$$

This is measured experimentally by recording the value during training. The value of ι_i depends upon both the specific data in question and the training algorithm being used and is difficult to estimate analytically.

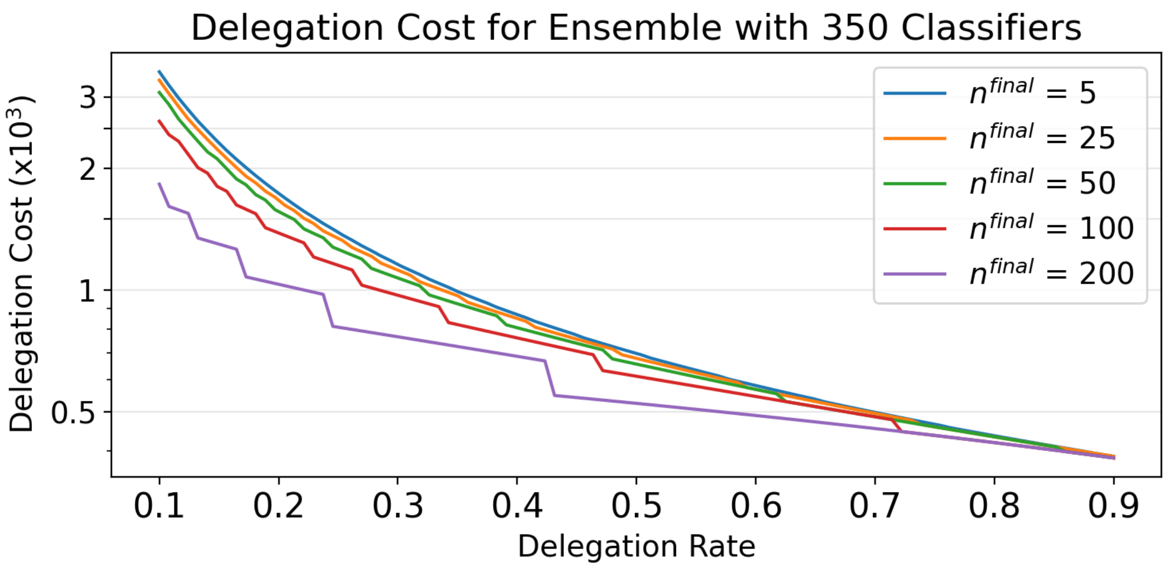


Figure 4.2: The lower bound on delegation cost as n^{final} and delegation rate are varied.

Our experiments hold constant the proportional size of the training data (at half of the entire dataset), and while we use pre-existing tools to train our classifiers we have little control over the number of iterations required to fully train classifiers. Thus, the number of representatives in the ensemble is the primary factor in training cost and the difference in training cost between a full ensemble and a delegating ensemble is the ratio of their sizes. To minimize cost we would always use exactly 1 classifier in our ensemble. Of course, this would reject many of the benefits that come with using ensemble methods so we instead select a minimal number of representatives that provide maximal accuracy. We consider the trade-off between accuracy and cost experimentally in Section 4.4.3.

4.4 Experimental Results

We now present experimental results which highlight two conclusions: First, that a single round of delegation between classifiers typically leads to no clear improvement over a full ensemble with no delegation. Second, that many repeated rounds of delegation during training significantly lowers the data cost of training while often also improving the accuracy of the resulting ensemble.

Dataset	Short Name	Examples	Categorical	Numerical
breast-cancer-w	bcw	699	0	9
credit-approval	cra	690	9	6
heart	hrt	270	0	13
ionosphere	ion	351	34	0
kr-vs-kp	kvk	3196	0	36
spambase	spb	4601	0	57
online-shoppers	onl	12330	10	7
occupancy-detection	occ	20560	0	5

Table 4.4: Datasets used through our experiments. The smaller 6 datasets are used in experiments from the GAIW paper "On the Limited Applicability of Liquid Democracy" [12] while all datasets are used in the subsequent work published at AAMAS [10]. All datasets listed have two classes and appear in the UCI Machine Learning Repository [71].

4.4.1 Data

Through our experiments we have made use of 8 different datasets. Each dataset that we use has exactly two classes and a variable number/type of features. They range in size from only 270 examples to 20560 examples. Table 4.4 lists each dataset, all of which were taken from the UCI Machine Learning Repository. Categorical features are those where a feature value has a finite number of options (e.g. whether an email contains any links), and numerical features are those that may take on any number of values (e.g. the number of links in an email).

4.4.2 One-shot Delegation

We first provide results for experiments in which voters take part in only one round of delegation, or when voters perform delegate myopically with no consideration of other voter behaviour. These experiments show that delegation is typically beneficial but not to a large degree. We first presented these results in the Games, Agents, and Incentives Workshop held as part of AAMAS 2021 [12].

Ensemble Setup

Our ensemble is composed of decision trees with a maximum depth chosen uniformly at random between 1 and 4. Each tree randomly uses Gini or Entropy criteria with equal

Dataset	1	10	20	30	40	50
breast-cancer-w	0.936	0.947	0.944	0.944	0.944	0.947
credit-approval	0.856	0.863	0.863	0.863	0.863	0.863
heart	0.763	0.750	0.766	0.764	0.770	0.767
ionosphere	0.864	0.890	0.892	0.890	0.893	0.89
kr-vs-kp	0.824	0.923	0.926	0.929	0.923	0.923
spambase	0.849	0.868	0.872	0.875	0.875	0.877

Table 4.5: Performance of decision tree ensembles of increasing size for each dataset considered in this subsection.

likelihood. Our experiments were implemented in Python using the scikit-learn library for all machine learning functions [175]. Initially we also experimented with ensembles that included SVM's and Neural Networks with equal probability as Decision Trees but found greatly increased training time and no significant difference in performance. Thus, in this section, we report results only for ensembles composed of Decision Trees. A new ensemble is randomly generated for each experiment configuration. When comparing experiments using a given delegation method against experiments using another delegation method on the same data the ensembles are generated using the same parameters but have a different specific instantiation of classifiers. As noted later, this can lead to noisy comparisons between different experiments.

Ensemble Performance Without Delegation

First, we show in Table 4.5 the test accuracy of ensembles of varying sizes before any delegation is applied. Classifiers are trained on a randomly selected 90% of each dataset and tested on the remainder; we show the average test accuracy over 300 trials. We see clearly that while there is typically a slim improvement from adding ensembles, only on the kr-vs-kp dataset is the difference quite significant.

Accuracy of Delegating Ensembles

We first present results from small-scale ensembles including the best possible accuracy found from an exhaustive search of all possible delegations. Table 4.6 shows our primary delegation mechanisms compared against each other over a variety of ensemble sizes averaged over 50 trials. As expected, the **Best** delegation mechanism generally outperforms

Dataset	Best	Direct	Diverse	Uniform	Random-R
breast-cancer-w	0.951	0.943	0.944	0.951	0.94
credit-approval	0.87	0.862	0.863	0.868	0.861
heart	0.80	0.762	0.758	0.802	0.771
ionosphere	0.902	0.884	0.889	0.903	0.878
kr-vs-kp	0.921	0.905	0.909	0.909	0.881
spambase	0.883	0.869	0.866	0.884	0.862

Table 4.6: Accuracy of several delegation mechanisms on each dataset averaged over 5, 7, and 9 voters with 3 and 5 representatives. Best delegations are from an exhaustive search over all possible valid delegations.

other mechanisms[2]. However, in this small setting the **Uniform** mechanism also performs quite well. Notably, in all cases the **Direct** ensemble with no delegation is outperformed by some form of delegation. This indicates there is generally room for improvement from delegation.

We next scale up our experiments to larger numbers of voters and find a similar result as before. Due to computational intractability we no longer consider the best possible delegations. Table 4.7, Table 4.8, and Table 4.9 show results from experiments considering ensembles with up to 49 voters, averaged over 50 trials. This shows how ensemble size and number of representatives affect the accuracy of the final ensemble. While accuracy does increase slightly as the ensemble grows, it is only a small improvement. We see that the **Uniform** delegation method generally leads to the best-performing ensemble the difference between **Uniform** delegations and the full ensemble is not significant on a t-test with $p < 0.05$.

Finally, we consider what outcomes arise when voters delegate if, and only if, there is a delegation available to them that will strictly improve group accuracy. Voters are selected in random order and, when selected, the group accuracy is calculated for each possible delegation the voter could make. Voters select the delegation option (including no delegation) that most improves group accuracy. This process repeats until no voter is able to improve accuracy by changing their delegation. Table 4.10 shows that, in this setting, there are typically very few pivotal voters. This varies somewhat based upon the dataset being considered. As the number of voters grows it becomes increasingly likely that the

[2] As noted when discussing the ensemble setup, classifiers are instantiated with randomly chosen parameters. In some cases in Table 4.6, the classifiers created when testing the **Uniform** mechanism are slightly more accurate than those created when using the **Best** mechanism, resulting in higher performance for **Uniform**.

Dataset	Direct	Diverse	Uniform	Random-R
breast-cancer-w	0.943	0.939	0.951	0.945
credit-approval	0.862	0.864	0.865	0.861
heart	0.772	0.768	0.785	0.766
ionosphere	0.883	0.874	0.91	0.883
kr-vs-kp	0.906	0.917	0.92	0.883
spambase	0.868	0.871	0.89	0.86

Table 4.7: Accuracy of delegation methods averaged over 9 voters and 5 representatives. Accuracy is highest in **Uniform** column but not by a significant margin.

Dataset	Direct	Diverse	Uniform	Random-R
breast-cancer-w	0.942	0.947	0.955	0.946
credit-approval	0.862	0.863	0.87	0.862
heart	0.76	0.767	0.805	0.767
ionosphere	0.892	0.894	0.908	0.881
kr-vs-kp	0.921	0.924	0.924	0.915
spambase	0.874	0.874	0.89	0.871

Table 4.8: Accuracy of delegation methods averaged over 29 voters and 5, 15, and 25 representatives. Accuracy is highest in **Uniform** column but not by a significant margin.

initial ensemble will not strictly increase its accuracy from any single delegation (though note that this does not mean the ensemble could not benefit from a series of delegations). Ultimately this is unsurprising – as ensemble size grows the chance that a single delegation can improve accuracy should, intuitively, decrease. The average boost in accuracy over all experiments in Table 4.10 is approximately 0.007, showing that while delegation is usually slightly useful, finding beneficial delegations is not likely when taking a greedy perspective towards accuracy improvement.

4.4.3 Repeated Delegation using Incremental Training

The previous section shows that while we may be able to achieve very slim improvements to accuracy through a single round of delegation, there is actually very little significant difference between delegation and the full ensemble. We now consider whether several rounds of delegation, during the training process are able to improve either accuracy or

Dataset	Direct	Diverse	Uniform	Random
breast-cancer-w	0.946	0.949	0.96	0.943
credit-approval	0.863	0.863	0.872	0.861
heart	0.767	0.774	0.82	0.762
ionosphere	0.888	0.894	0.916	0.885
kr-vs-kp	0.923	0.921	0.934	0.91
spambase	0.874	0.877	0.899	0.868

Table 4.9: Accuracy of delegation methods averaged over 49 voters and 5, 15, and 25 representatives. Accuracy is highest in **Uniform** column but not by a significant margin.

dataset	9	19	29	39	49
breast-cancer-w	0.44	0.34	0.26	0.06	0.18
credit-approval	0.12	0	0.02	0	0.02
heart	0.52	0.24	0.24	0.2	0.1
ionosphere	0.34	0.3	0.28	0.1	0.14
kr-vs-kp	0.32	0.16	0.16	0.12	0.14
spambase	1.24	0.82	0.46	0.54	0.36

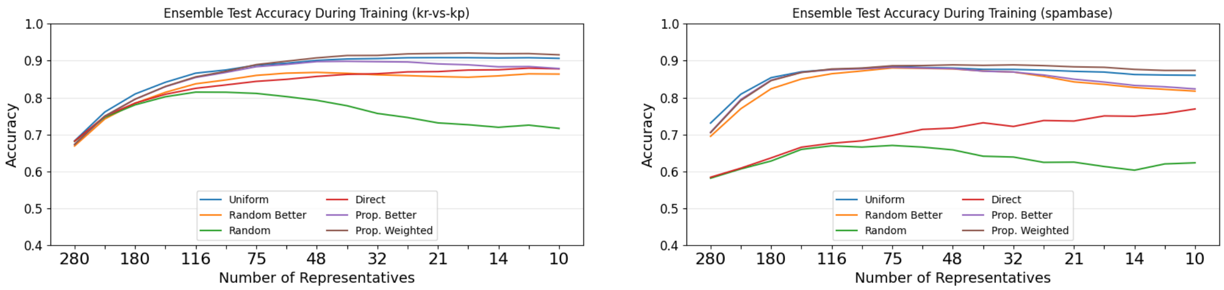
Table 4.10: Mean number of delegations required to reach a state where no single delegation strictly improves accuracy as the initial ensemble size increases from 9 to 49.

the amount of data that must be trained upon.

Here we scale up our previous experiments to consider larger ensembles of 350 classifiers. To support incremental training we now use Support Vector Machines rather than Decision Trees as the classifier making up our ensembles.

Basic Comparison of Delegation Mechanisms

We first compare the general behaviour of each delegation mechanism as well as the full ensemble of “direct” voters. Figure 4.3 shows accuracy during training for each delegation method on the two datasets which benefit most from delegation. Other datasets are shown in Appendix A. The x-axis shows the number of non-delegating classifiers at each training step, corresponding with time. These figures highlight that even relatively simple delegation mechanisms are able to outperform a larger ensemble. The relatively worse results from Random delegations show that removing untrained classifiers from an ensemble may



(a) (b)

Figure 4.3: Accuracy of each basic delegation method during incremental training with `partial_fit` method on kr-vs-kp **(a)**; and spambase **(b)** datasets beginning with 350 classifiers. At each increment 20% (rounded down) of active voters are chosen to become delegators, continuing until 10 representatives remain or the entire dataset has been trained upon. See Appendix A for other datasets.

not always cause harm (when compared with the Direct ensemble). However, as classifiers get more training and the ensemble shrinks, random delegations eventually become detrimental to ensemble accuracy.

These results are strengthened by considering the accuracy of fully trained ensembles on each dataset. In Table 4.11 we compare each delegation mechanism with direct delegation and individual classifier performance. This shows that while delegation is more beneficial on some datasets than others, it is never significantly harmful. In some cases we see that delegation does not provide any advantage over the performance of the median classifier in the ensemble.

Parameter Tuning

Three parameters most directly affect the accuracy and training cost of ensembles: increment size, delegation rate, and initial number of voters. We explore a range of values for each of these parameters and present the results in Figure 4.4. Each hexagon in these ternary graphs represents ensemble test accuracy averaged over 50 trials with one set of parameters. In each trial, classifiers are created with new random seeds, the dataset shuffled, and new train/test data sampled.

Ensemble accuracy corresponds to the colour bar underneath the figure, with lighter values indicating higher accuracy. The parameter values aligned with each cell can be found

Dataset	Uniform	Random Better	P. Better	P. Weighted	Random	Direct	Median Single
bcw	0.898	0.889	0.878	0.907	0.864	0.91	0.966
cra	0.608	0.613	0.609	0.621	0.599	0.618	0.633
hrt	0.589	0.567	0.57	0.577	0.586	0.597	0.625
ion	0.837	0.839	0.841	0.844	0.836	0.845	0.832
kvk	0.925	0.887	0.908	0.934	0.736	0.901	0.784
spb	0.862	0.818	0.836	0.88	0.676	0.851	0.796
onl	0.801	0.748	0.75	0.784	0.716	0.82	0.877
occ	0.92	0.92	0.92	0.927	0.918	0.94	0.988

Table 4.11: A numerical comparison of test accuracy on fully trained and delegated ensembles, direct ensembles, and the median accuracy of a classifier from each ensemble. Each ensemble began with 350 classifiers and, when applicable, 20% of classifiers began to delegate after each round of incremental learning.

by the direction of tick marks outside the graph. The bottom axis, delegation rate, goes "up and to the right". Accordingly, the highest accuracy (lightest colour) with the *uniform* delegation mechanism (top left in Figure 4.4) is found when increment size, delegation rate, and initial number of voters are respectively (65, 0.05, 350) or (85, 0.05, 200).

Across delegation methods and datasets this parameter search generally finds that large increment size, a relatively small ensemble, and a low delegation rate lead to the highest accuracy. This strongly indicates a set of parameters to use in subsequent experiments in order to *maximize accuracy*. However, in order to *minimize cost* different parameters are needed. Calculations in Section 4.3.2 show that a low delegation rate greatly increases training cost, thus presenting a trade-off between ensemble performance and training cost. In order to balance training cost and accuracy our experiments default to parameter values of: a delegation rate of 0.2 (20% of representatives delegate at each increment), an increment size of 25 examples, and an initial ensemble size of 350 classifiers.

Is Diversity Useful?

As we discuss in Section 2.4.4, high diversity among members of an ensemble has often been considered a beneficial trait or even used as a basis for creating ensembles. However, there is still no clear consensus on how to harness diversity in a way that improves ensemble performance. To explore whether diversity is useful in our setting we implemented a number of existing diversity measurements and integrated them into a delegation mechanism.

We evaluate two pairwise diversity measures: *Euclidean distance* and the *q-statistic*

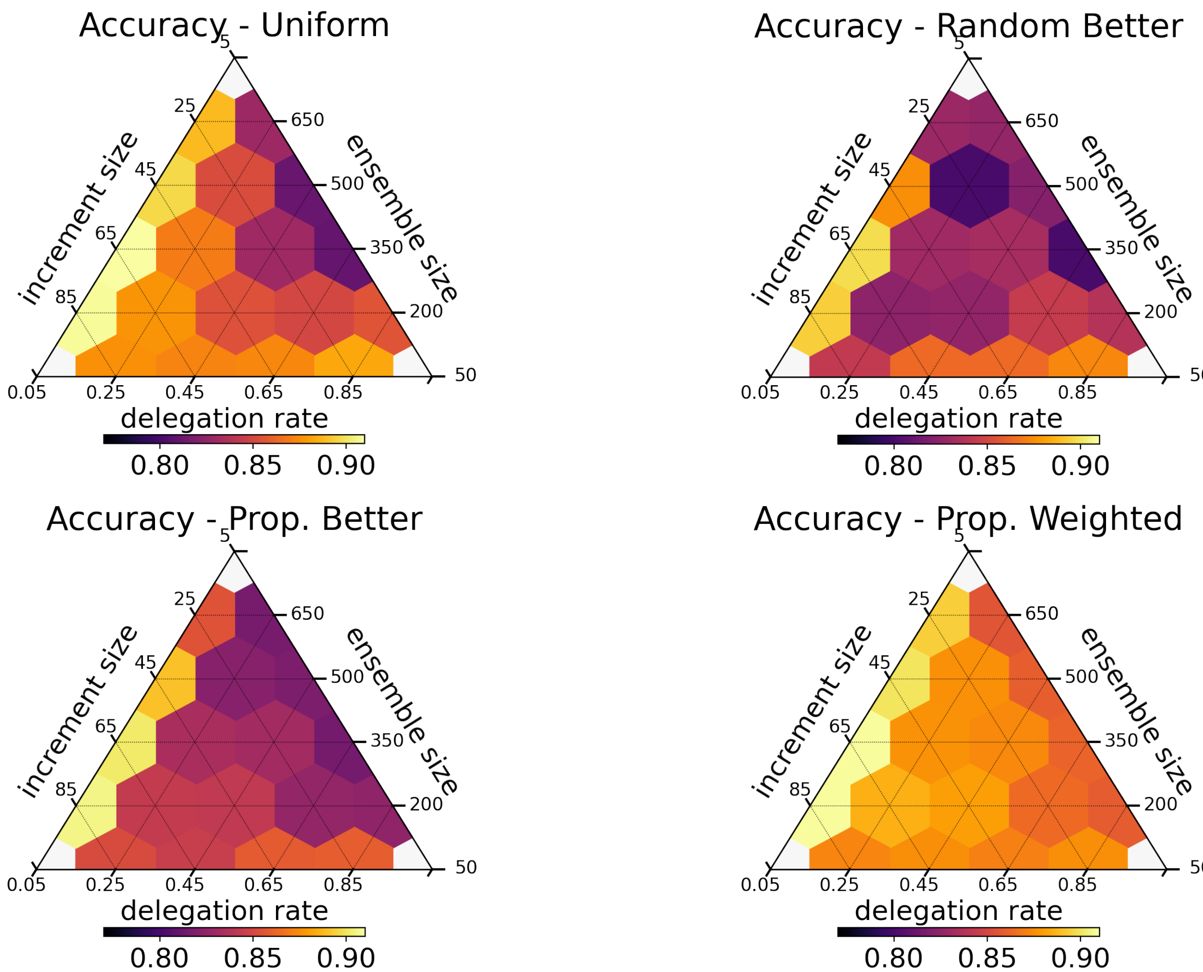
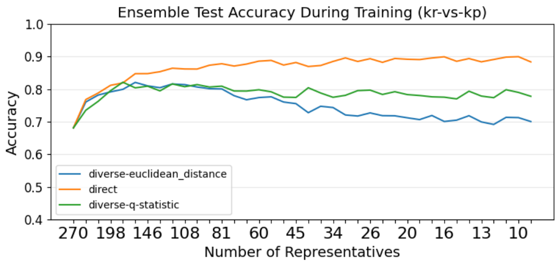
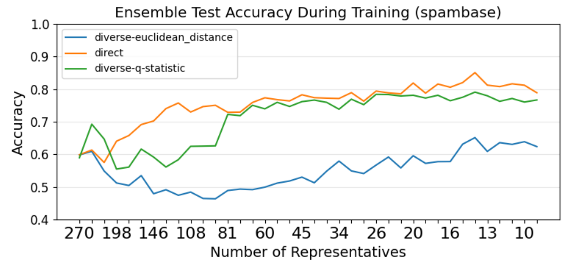


Figure 4.4: Test accuracy of fully trained ensemble across delegation methods as parameters affecting accuracy are varied. Results displayed are from the spambase dataset. Random delegations are omitted as they perform significantly worse than the displayed delegation mechanisms; Direct delegations are omitted as increment size and delegate rate do not affect Direct ensemble performance.

[134]. We calculate the diversity between two classifiers v_i and v_j on training increment i by considering the confusion matrix of each classifier's predictions on data in that increment, D_i and D_j. Kuncheva et al. discuss several metrics based on the confusion matrix we include as Table 4.12. We compared several of these and found the *q-statistic* typically best. For classifiers v_i and v_j, the statistic is defined as,



(a)



(b)

Figure 4.5: Accuracy on test data over incremental training using diversity metrics to guide delegation.

	$\mathbf{D}_j$ correct	$\mathbf{D}_j$ incorrect
$\mathbf{D}_i$ correct	N^{11}	N^{10}
$\mathbf{D}_i$ incorrect	N^{01}	N^{00}

Table 4.12: Confusion matrix defining N based on the number of instances in which each classifier's predictions agree/disagree and are correct/incorrect. e.g. N^{10} is the count of examples in a dataset $\mathbf{D}$ on which v_i made a correct prediction and v_j made an incorrect prediction.

$$\mathcal{Q}_{i,j} = \frac{N^{11}N^{00} - N^{01}N^{10}}{N^{11}N^{00} + N^{01}N^{10}}$$

We also consider the Euclidean distance between the list of predictions made by each pair of classifiers in each increment of training. As discussed in Section 4.2.3, $\rho^{\text{diverse}}(v_i, v_j)$ selects the pairs v_i and v_j which are most similar and chooses the less accurate of each pair to delegate to the more accurate.

Figure 4.5 shows accuracy during training for both measures, compared with direct delegations, for spambase and kr-vs-kp. See Appendix A for similar graphs on the remaining datasets. These experiments were performed before finalizing the parameter values used in other sections and show the average over 30 trials using an initial size of 300 classifiers, a delegation rate of 10%, and an increment size of 25. We find there is no clear evidence that optimizing for diversity provides a benefit to delegation in our setting. This is not to say that diversity cannot be beneficial – a principled method of introducing diversity during classifier creation may improve performance – but using well-known diversity measures to guide delegation does not immediately prove improve performance. Due both to these results and to the high computational requirements of computing diversity for every pair of voters we have not considered diversity in other experiments.

Weight Distribution

Each delegation mechanism that we use leads to a distinct distribution of weights, both in the fully trained ensemble and during training. This distribution is also affected by the size of the dataset. Classifier weights are interesting primarily through their connection with ensemble performance. Intuitively, ensembles with high weight centralization may produce less stable results and their accuracy is much more variable. For example, if one classifier has a majority of weight the ensemble output is always the same as that of the single classifier and beneficial effects of ensembles are removed.

Thus, we consider the minimum number of classifiers required to form a majority of ensemble weight and the maximum number of classifiers able to form a minority of weight. These measures give insight into the tendency of each delegation mechanism towards centralization. The minimum number of classifiers required to form a majority of weight is low if there is a high degree of weight centralization (e.g. if there is a single dictator this has a value of 1). The maximum number of classifiers able to form a minority of weight is similar; if weight is distributed evenly this value is half of the total number of non-delegating classifiers and gets higher as weight becomes more centralized.

Figure 4.6 shows the minimum number of classifiers required to form a majority of weight for the spambase and kr-vs-kp datasets during training. The same figures for all datasets can be found in Appendix A. All results are averaged over 500 trials on standard parameters. Note that the y-axis shows the minimum majority size on a logarithmic scale.

Table 4.13 shows the minimal number of voters required to form a majority after training is complete while Table 4.14 shows the maximum number of voters able to form a minority of weight after training is complete. (Note that this second measure, maximum number of voters able to form a minority, was considered only after running experiments and our recorded data cannot reproduce this measure over the course of training.)

In all cases results are similar and demonstrate clearly that the Proportional Weighted delegation mechanism leads to much less centralization of weight both during and at the end of training. This is unsurprising as the mechanism was designed after noting the problematic issue of weight centralization in other delegation mechanisms. The Proportional Weighted mechanism specifically decreases the chance that a voter will receive a delegation in proportion to how much weight the voter already has.

Dataset	Uniform	Random Better	P. Better	P. Weighted	Random	Representatives
bcw	1.78	1.48	1.68	4.39	1.35	17.2
cra	1.49	1.50	1.66	4.48	1.27	17.4
hrt	4.11	12.94	12.85	22.99	10.97	93
ion	4.53	8.84	9.05	17.87	7.56	75
kvk	2.12	1.13	1.39	3.00	1.07	10
spb	2.12	1.27	1.41	3.03	1.10	10
onl	1.26	1.14	1.14	2.92	1.08	10
occ	1.31	1.12	1.19	2.86	1.09	10

Table 4.13: Minimum number of voters required to form a majority of weight after completing a delegation process beginning with 350 voters. Proportional Weighted delegations consistently exhibit less weight centralization. Values are averaged over 500 trials.

Comparison with other Ensemble Methods

We now compare the training cost and final accuracy of a variety of ensemble methods including our delegative methods as well as Adaboost, a well-known ensemble boosting algorithm [94]. As Proportional Weighted typically outperforms other delegation methods, we first present comparison results showing only this delegation method and compare with other delegation methods in Table A.1 and Table A.2. Specifically, we compare the following ensembles:

- **Direct**: A full ensemble of 350 classifiers, all equally weighted.
- **Prop W Acc**: 350 classifiers using Proportional Weighted delegations with an increment size of 65 and delegation rate of 0.05 (chosen to maximize accuracy).
- **Prop W Cost**: 350 classifiers using Proportional Weighted delegations with an increment size of 25 and delegation rate of 0.85 (chosen to minimize cost).
- **Ada DT Full**: Adaboost with a decision tree as the underlying classifier with up to 350 classifiers (the size of the initial delegating ensemble).
- **Ada SGD Full**: Adaboost with an SVM as the underlying classifier with up to 350 classifiers.
- **Ada DT Small**: Adaboost with a decision tree as the underlying classifier with up to 10 classifiers (the size of the fully delegated ensemble).

Dataset	Uniform	Random Better	P. Better	P. Weighted	Random	Representatives
bcw	15.21	16.29	15.32	12.57	15.71	17.2
cra	15.49	17.63	15.33	12.49	15.78	17.4
hrt	88.85	80.13	80.01	69.78	82.10	93
ion	70.43	66.11	65.86	56.99	67.55	75
kvk	7.87	8.86	8.60	6.96	8.93	10
spb	7.83	8.73	8.57	6.96	8.90	10
onl	8.7	8.86	8.86	7.08	8.92	10
occ	8.64	8.89	8.80	7.11	8.91	10

Table 4.14: Maximum number of voters able to form a minority of weight after completing a delegation process beginning with 350 voters. Proportional Weighted delegations consistently spread weight among a larger number of direct voters. Values are averaged over 500 trials.

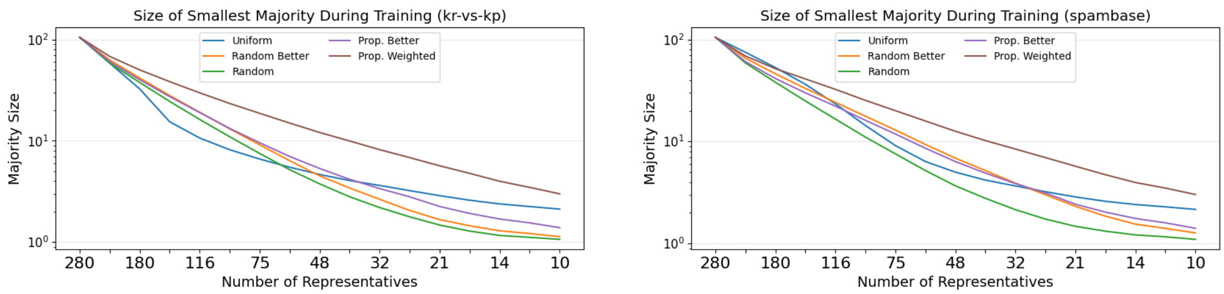
- **Ada SGD Small**: Adaboost with an SVM as the underlying classifier with up to 10 classifiers.

We use the default `sci-kit` learn parameters for Adaboost and each underlying classifier [175]. Results are averaged over 50 trials for each set of parameters.

Table 4.15 and Table 4.16 show results of comparing between ensemble types. Training cost is shown as a proportion of the cost of training a full Direct ensemble. Comparing the two parameterizations of delegation highlights the trade-off between accuracy and cost allowed by delegation. In many cases, delegative ensembles have a comparable cost to Adaboost methods and in some cases accuracy and F1 score from delegation are higher than with Adaboost.

4.5 Discussion

In this chapter we have developed a method for pruning ensembles using liquid democracy, a framework for delegative voting. Applying delegation to remove and re-weight classifiers over the course of incremental training of an ensemble proves highly effective at reducing the training cost of an ensemble while maintaining or improving accuracy over a naive ensemble. Our method has several parameters which we demonstrate can be tuned to effectively manage a trade-off between accuracy and training cost.



(a) (b)

Figure 4.6: Size of the smallest set of classifiers comprising a majority of ensemble weight during training on kr-vs-kp **(a)**; and spambase **(b)** datasets beginning with 350 classifiers. At each increment 20% (rounded down) of active voters are chosen to become delegators, continuing until 10 representatives remain or the entire dataset has been trained on. See Appendix A for other datasets.

We examined several delegation methods and used two novel methods: Proportional Better and Proportional Weighted. As in Chapter 3, our experiments reveal that our Proportional Weighted delegations achieve higher accuracy than other delegation methods. In our analysis of the weight distribution of voters over many rounds of delegation we see that, as hypothesized in the previous chapter, this mechanism is significantly better at avoiding weight centralization than other mechanisms. These results strengthen the argument for a causal link between weight centralization from delegation and reduction in accuracy.

We find that our method is often comparable in either training cost or accuracy to the well-known Adaboost ensemble method. We note also that our method operates along a different dimension than most ensemble methods. Delegation does not affect the learning process itself, while many ensemble methods primarily focus on adjusting how specific parts of training data are learned. Conceptually, there may be potential to use our method in conjunction with other ensemble pruning methods.

4.6 Conclusions

This chapter provides an exploration of a novel application of liquid democracy: ensemble pruning. We demonstrate the significant benefits of this method as compared to naive ensembles. While state-of-the-art ensemble algorithms, such as Adaboost often outperform

Ensemble	breast-cancer-w			credit-approval		
	Acc	F1	Cost	Acc	F1	Cost
Direct	0.907	0.926	1	0.628	0.585	1
Prop W Acc	0.907	0.927	0.782	0.631	0.586	0.817
Prop W Cost	0.9	0.92	**0.033**	0.609	0.584	0.036
Ada DT Full	0.953	0.932	0.039	0.818	0.833	0.07
Ada DT Small	0.957	0.938	0.001	0.852	0.862	0.002
Ada SGD Full	0.965	0.95	0.014	0.653	0.681	0.007
Ada SGD Small	0.965	0.95	0.013	0.65	0.674	0.007
Ensemble	heart			ionosphere		
	Acc	F1	Cost	Acc	F1	Cost
Direct	0.584	0.59	1	0.854	0.765	1
Prop W Acc	0.573	0.574	0.9	0.853	0.762	0.894
Prop W Cost	0.565	0.547	0.035	0.802	0.716	0.034
Ada DT Full	0.758	0.783	0.027	0.916	0.937	0.031
Ada DT Small	0.803	0.824	0.001	0.896	0.921	0.001
Ada SGD Full	0.683	0.718	0.01	0.861	0.898	0.025
Ada SGD Small	0.678	0.718	0.009	0.861	0.898	0.015

Table 4.15: Accuracy, F1 score, and Training Cost (relative to Direct ensembles) when comparing a variety of Adaboost methods against two parameterizations of a delegating ensemble. Bold values indicate when a delegating ensemble outperforms *at least one* Adaboost method.

our method we highlight that ensemble pruning with liquid democracy has strong potential for application with very large datasets or where there are resources to train classifiers in parallel.

We end this chapter by highlighting two possible extensions to this work. As we have discussed, our delegative pruning method is highly generic and does not consider the underlying data itself. Highly successful methods, such as Adaboost, adjust their training based on how well existing classifiers perform on individual examples. Adding an element of this data-level adjustment to either a new aspect of our procedure or as a novel delegation mechanism has potential to enhance performance. Additionally, we note the similarities between incremental learning and continual learning. Continual learning is an incremental setting where the underlying distribution of data being classified is expected to change over time. In joint work with Carter Blair, we have applied ideas of delegation to develop

Ensemble	kr-vs-kp			occupancy-det		
	Acc	F1	Cost	Acc	F1	Cost
Direct	0.91	0.903	1	0.946	0.964	1
Prop W Acc	**0.947**	**0.943**	0.269	0.94	0.96	0.055
Prop W Cost	0.908	0.902	0.026	0.916	0.936	0.029
Ada DT Full	0.966	0.968	0.02	0.99	0.978	0.009
Ada DT Small	0.946	0.948	0.001	0.989	0.977	0
Ada SGD Full	0.941	0.944	0.06	0.984	0.966	0.005
Ada SGD Small	0.91	0.915	0.01	0.984	0.966	0.005
Ensemble	online-shoppers			spambase		
	Acc	F1	Cost	Acc	F1	Cost
Direct	0.869	0.927	1	0.86	0.88	1
Prop W Acc	0.843	**0.906**	0.058	**0.909**	**0.927**	0.198
Prop W Cost	0.768	**0.817**	0.029	**0.869**	**0.89**	0.029
Ada DT Full	0.888	0.605	0.019	0.934	0.916	0.026
Ada DT Small	0.89	0.62	0.001	0.916	0.891	0.001
Ada SGD Full	0.878	0.447	0.012	0.786	0.719	0.013
Ada SGD Small	0.879	0.444	0.011	0.791	0.742	0.012

Table 4.16: Accuracy, F1 score, and Training Cost (relative to Direct ensembles) when comparing a variety of Adaboost methods against two parameterizations of a delegating ensemble. Bold values indicate when a delegating ensemble outperforms *at least one* Adaboost method.

ensembles for the continual learning setting [30]. Additional work can extend this to larger ensembles and more involved forms of delegation.

Chapter 5

Learning Voting Rules

In Chapter 4 we developed a framework for applying social choice (specifically, liquid democracy) and showed that it improves outcomes from machine learning – increasing ensemble accuracy and reducing the cost of training as compared to a full ensemble. In this chapter we take the reverse approach, using machine learning to benefit social choice. Here we explore the idea of using neural networks as voting rules in order to both learn about existing voting rules, and to develop entirely novel voting rules. We expand existing concepts of data-driven axiomatic analysis to create a data-driven approach to training new voting rules optimized for axiom satisfaction. Our framework enables us to experimentally analyze questions of a type and scale that are not feasible using purely theoretical approaches.

5.1 Introduction

Voting rules are frequently analyzed in terms of their axiomatic properties, and are even designed with axiom satisfaction in mind [121]. Despite this, several difficulties prevent new voting rules from satisfying every (or, even, many) desirable axioms. First, impossibility theorems – describing sets of axioms which are impossible to mutually satisfy – abound in social choice [13, 101, 99]. As well, the concept of a "desirable" axiom is not universally agreed upon [198]; some axioms may be vital in one setting while harmful in another. Finally, even with a set of axioms which are shown to be mutually satisfiable and are considered desirable, there is no general formula for constructing a reasonable voting rule[1]

[1]It is conceptually possible to develop a voting rule which iterates over all possible outcomes and elects winners not violating any desirable properties; however, we would consider such a rule *unreasonable*.

which satisfies the properties.

We develop a methodology for automated creation of novel voting rules based on data-driven axiom satisfaction that addresses each of these challenges. We do this by creating training data of elections wherein the election winner is the alternative violating the fewest of some arbitrary set of desirable axioms. Key to this approach is encoding social choice knowledge in a way that generic machine learning techniques are able to utilize. Evaluation of existing voting rules shows they violate desirable axioms at a very high rate while learned rules are able to violate the same axioms far less often.

To approach learning new rules in a principled manner, we first consider the best way to learn voting rules by learning existing voting rules *then* developing new voting rules. Prior work has used neural networks to replicate existing voting rules [46, 133] or to generate new voting rules [7, 88], but only very little of this work has considered best practices for how to learn voting rules [152] (e.g. how big a neural network should be, what types of features are most useful, which types of rules are easiest to learn). We develop insights into existing rules by exploring the most effective ways to learn them. We use these insights in learning novel rules with strong axiomatic properties according to the framework we develop for experimentally measuring axiom violations of arbitrary rules on arbitrary preference distributions. Our results provide a first step towards practically circumventing impossibility theorems by showing the possibility of rules which avoid impossibilities as frequently as possible. In this chapter we pursue three goals:

1. We identify best practices for learning voting rules. We learn existing voting rules which provide a ground truth comparison against which we measure the efficacy of our learning process.

2. We learn about existing voting rules using experimental tools to compare both existing rules and novel rules.

3. We use data about axiom violations to learn new rules which minimize violations across wide sets of axioms.

Ultimately, by showing the existence of voting rules whose winners violate axioms at a far lower rate than those of existing rules we highlight the potential for development of new rules. Such novel voting rules may provide superior axiomatic properties while being more interpretable than a machine learning approach. In domains with specific axiomatic requirements where interpretability is unimportant, our machine learning approach to learning novel rules can provide direct benefit. We structure this chapter as follows:

- In Section 5.2 we briefly review the background material relevant to this chapter and introduce the specific axioms and voting rules we consider.
- In Section 5.3 we describe the data-driven approach we take to measuring the axiom violations of rules; both existing and novel. We also develop a method of comparing the winners of voting rules across a large number of elections in order to measure the similarity between two existing rules or groups of rules.
- In Section 5.4 we describe the procedure we take for learning functions which act as novel voting rules and learning to approximate existing rules.
- In Section 5.5 we describe the process and results of learning to approximate existing rules. We highlight ways in which our learning process could be improved and identify interesting patterns when comparing between the learned rule approximations and actual voting rules.
- In Section 5.6 we learn novel voting rules using our axiom-based approach. We show that axiom violation data suffices to learn new voting rules which significantly outperform many existing voting rules on the axiom violation framework we develop in Section 5.3.

5.2 Setting

We first review the elements of Chapter 2 that are most relevant to this chapter and briefly introduce each of the voting rules and axioms that we use throughout this chapter. We summarize notation used in this chapter in Table 5.1. The most significant difference between this chapter and Chapter 3 and Chapter 4 is that we now move away from the two-alternative epistemic domain and consider the more common setting of elections with different numbers of alternatives and no ground truth agreed upon by voters. Rather than studying the accuracy of a voting process, we are now primarily interested in the behaviour of existing voting rules, and in developing new rules with desirable axiomatic properties.

We consider a set of voters V, each with a complete preference order P over alternatives A. Each voter's preference order P_i is sampled from a distribution $\mathcal{D}$. We now also consider multi-winner elections; a voting rule $\mathcal{F}$ accepts a profile of n preference orders, as well as a number of winners k, with $1 \leq k < m$, and returns a set of k winning alternatives. We omit k when only electing a single alternative. That is, $\mathcal{F}(\mathcal{P}) = \mathcal{F}(\mathcal{P}, 1)$.

Many of the multi-winner voting rules that we consider are approval-based. These rules assume that each voter considers some set of alternatives "acceptable" and submits an

Symbol	Meaning
V	The set of n voters ranking alternatives.
A	The set of m alternatives being ranked by voters.
k	The number of election winners, also called the committee size.
$\mathbb{A}$	The set of axioms under consideration.
P	The preference order of a single voter; a complete ranking of all alternatives.
$\mathcal{P}$	A single preference profile consisting of n preference orders.
$\mathcal{P}_{App}$	The alternatives approved by each voter; equal to their top k preferences.
$\mathbb{P}$	A set of preference profiles being used as testing or training data.
$\mathcal{D}$	The preference distribution from which each voter samples their preference order.
$\mathcal{F}(\mathcal{P}, k)$	A voting rule. k is omitted when single winner rules are clear from context.
$\mathrm{AVR}(\mathcal{F}, \mathbb{P}, \mathbb{A})$	Axiom Violation Rate of $\mathcal{F}$ on preference profiles $\mathbb{P}$ over axioms $\mathbb{A}$.
$\mathtt{diff}(\mathbb{F}^1, \mathbb{F}^2, \mathbb{P})$	Normalized difference between rules $\mathbb{F}^1$ and $\mathbb{F}^2$ on preference profiles $\mathbb{P}$.

Table 5.1: Overview of notation most important throughout this chapter.

approval set which consists of the most preferred alternatives in their preference order. This approval set, denoted $App(v_i)$, contains the k alternatives which v_i most prefers; without information about the relative preference of v_i over the alternatives. In our setting, $App(v_i)$ always contains v_i's k most preferred alternatives, and $\mathcal{P}_{App} = (App(v_1), ...App(v_n))$. When clear from context (e.g. when describing the input to an approval-based voting rule) we omit the subscript and write $\mathcal{P}$ in place of $\mathcal{P}_{App}$.

5.2.1 Voting Rules

Here we list and categorize each voting rule we use. We show single winner and multi-winner rules separately; multi-winner rules are further divided into score-based and approval-based rules.

Observe that multi-winner rules take as a parameter the number of winners, which can be 1. Methodologically, single winner rules can be seen as a restriction of multi-winner rules. Through this chapter we occasionally use the term "committee" to describe the output of a voting rule. We use this term generally for both single and multi-winner rules.

Single Winner Rules

We learn functions approximating each of the single winner rules below. Many of these rules require differing amounts of information to compute their winner, based on Fishburn's classification and whether or not they are a positional scoring rule. Some rules may by differentiated by only minute changes (which can, upon occasion, lead to large differences in final output). We include a range of rules from very common (Plurality, Borda, Instant Runoff) to the less well known (Raynaud) or recently developed (Stable Voting). All the rules that we use are implemented in the Preferential Voting Tools library[2]. Note that we do not include single winner approval voting. Despite being a common rule, we avoided use of any rule which required further parameterization, such as how many alternatives each voter should approve.

Below we list the name of each single winner rule that we consider. A full definition of each rule is included in Appendix B. We list rules in groups according to their Fishburn class with C1 rules coloured red, C2 rules coloured blue, and C3 rules coloured green. Condorcet-consistent rules are marked with an asterisk, and positional scoring rules are underlined.

C1 Rules:

- Banks*
- Bipartisan Set*
- Condorcet*
- Copeland*
- GOCHA*
- Llull*
- Slater*
- Top Cycle*
- Uncovered Set*

C2 Rules:

- Beat Path*
- Blacks*
- <u>Borda</u>
- Borda-Minimax Faceoff*
- Copeland-Global-Borda*
- Loss-Trimmer Voting*
- Minimax*
- Raynaud*
- Simple Stable Voting*
- Split Cycle*
- Stable Voting*

[2]The Preferential Voting Tools library can be found at https://pref-voting.readthedocs.io

C3 Rules:

- Anti-Plurality
- Baldwin*
- Benham*
- Bracket Voting
- Bucklin
- Condorcet Plurality*
- Coombs
- Copeland-Local-Borda*
- Daunou*
- Instant Runoff
- Knockout Voting
- Plurality
- Simplified Bucklin
- Strict Nanson*
- Superior Voting*
- Tideman Alternative GOCHA*
- Tideman Alternative Top Cycle*
- Weak Nanson*
- Weighted Bucklin

Multi-winner Rules

We consider several multi-winner rules as well. Most of these rules can be seen as fitting into the categorization of multi-winner rule goals we described in Chapter 2; rules can focus on individual excellence, diversity, or proportionality. We selected these rules for several goals (1) the inclusion of well-known rules, (2) representing a variety of goals and algorithmic approaches, (3) ability to compare with pre-existing theoretical results. Along with the definition of each rule we list the axioms it is known to satisfy; axiom definitions are found in the following section. Note that we do not include negative results – rules may satisfy axioms that we do not list.

We consider the following rules which are based on the full rankings that each voter provides. Note that the rules k-Borda and Single Non-Transferable Vote are direct extensions of positional scoring rules (Borda and Plurality, respectively) to the multi-winner setting. Any positional scoring rule can trivially be converted to a multi-winner rule by simply selecting the k highest scoring alternatives.

k-Borda: An extension of Borda to multiple winners. Each voter assigns $m-1$ points to their top ranked alternative, $m-2$ points to their second ranked alternative etc. $\mathcal{F}^{\text{Borda}}$ returns the k alternatives with highest scores. k-Borda satisfies the Unanimity axiom [77].

Single Non-Transferable Vote (SNTV): An extension of Plurality to multiple winners. Each voter assigns one point to their most preferred alternative. $\mathcal{F}^{\text{SNTV}}$ returns the k alternatives with the most points. SNTV satisfies the Majority Winner and Solid Coalitions axioms [77].

Single Transferable Vote (STV): $\mathcal{F}_k^{\text{STV}}$ returns all alternatives that are top-ranked by more than a quota of $\frac{n}{k+1}$ voters. If that is less than k alternatives, votes for alternatives that are beyond the quota are transferred to their next most highly-ranked alternative. If no voter receives votes above the quota, the alternative with the fewest votes is eliminated and their votes transferred. This continues until k winners are found. We use *fractional* weight transfer; each voter top-ranking an alternative beyond the quota has an equal portion of their weight transferred to their next favourite alternative such that all excess weight beyond the quota is transferred [213]. STV satisfies the Majority Winner [213] and Solid Coalitions axioms [77].

We also consider several rules that use a restricted view of voter preferences, the class of approval-based rules. Approval-based multi-winner voting rules make use of a voters' *approval ballot* $App(v_i)$ which contains their k highest ranked alternatives[3]. The set of all approval ballots is $\mathcal{P}_{App} = (App(v_1), \ldots, App(v_n))$. Each of the approval-based rules we consider is implemented in the ABCVoting library [137].

Definition 5.2.1. *Approval-based multi-winner voting rules, also referred to as Approval-Based Committee (ABC) rules, are those rules using as input only the approval ballot of each voter. An ABC rule is a function $\mathcal{F}(\mathcal{P}_{App}, k)$ which returns a set of k alternatives.*

Approval-based rules can be further subdivided by whether or not they are in the class of Thiele rules. First described in 1895 by Torvald Thiele [212, 125], Thiele rules are defined by some satisfaction function w which they aim to maximize. Thiele rules can consider ranked ballots but are more frequently used in the approval domain (we only consider Thiele rules in the context of approval-based rules).

Definition 5.2.2. *A Thiele rule $\mathcal{F}$ is characterized by a satisfaction function $w : \mathbb{N} \to \mathbb{R}$ which scores sets of alternatives based on the number of alternatives in the set which each voter approves. w is weakly monotonically increasing and has $w(0) = 0$. $\mathcal{F}$ elects a set of winners which maximizes the score function [138]:*

$$score_w(\mathcal{P}_{App}, C) = \sum_{v_i \in V} w(|App(v_i) \cap C|)$$

[3]We restrict our attention to the setting where voters approve of exactly their k favourite alternatives as a simplifying assumption when generating preference data; however, our model naturally accommodates the more general setting where each voter can approve differing numbers of alternatives.

The approval-based rules which we consider are the following:

Bloc: Define $sc(a)$ as the number of voters approving of alternative a. $sc(a) = \sum_{v \in V} \mathbf{1}_{\{a \in App(v)\}}$. $\mathcal{F}^{\text{Bloc}}$ returns the committee C^* containing the k alternatives receiving the most approvals:

$$C^* = \arg\max_{\substack{C \subseteq A, \\ |C|=k}} \sum_{a \in C} sc(a).$$

In our setting the Bloc rule can be seen as a special case of Approval Voting where all voters approve k alternatives. Bloc satisfies the Strong Pareto Efficiency [138], Unanimity, and Fixed Majority axioms [77].

Proportional Approval Voting (PAV): Given committee C, define the PAV score of the committee as

$$sc_{\text{PAV}}(C) = \sum_{v_i \in V} \sum_{j=1}^{|C \cap App(v_i)|} \frac{1}{j}.$$

$\mathcal{F}^{\text{PAV}}$ returns $C^* = \arg\max sc_{\text{PAV}}(C)$. This rule aims to select alternatives such that each voter approves of a similar number of elected alternatives. PAV is the original rule described by Thiele [138]; the increased satisfaction each voter receives for an additional winning alternative of which they approve follows the sequence of harmonic numbers: $1, \frac{1}{2}, \frac{1}{3}, ..., \frac{1}{x},$. PAV is the Thiele method with $w(x) = \sum_{j=1}^{x} \frac{1}{j}$. PAV satisfies the Pareto, JR, and EJR axioms [138].

Chamberlin-Courant (CC): We consider three variations of the Chamberlin-Courant rule. For committee $C \subset M, |C| = k$, define

$$sc_{CC}(C) = |\{v \in V \mid App(v) \cap C \neq \emptyset\}|$$

$sc_{CC}(C)$ is the number of voters who approve at least one alternative in C. This base version of CC is also a Thiele method, with $w(x) = min(1, x)$; i.e. each voter with at least one approved alternative adds to the score of the committee. Chamberlin-Courant has two variants, an approximation of CC and an alternative tie-breaking method. Chamberlin-Courant satisfies the JR axiom [138].

Sequential CC: $\mathcal{F}^{\text{seq-CC}}$ is an approximation of CC that constructs a winning committee from the empty set by iteratively adding $a \in A$ that increases the sc_{CC} the most at each step. Sequential CC satisfies the JR axiom [138].

Lexicographic CC: $\mathcal{F}^{\text{lex-CC}}$, maximizes sc_{CC}, breaking ties by selecting the committee maximizing the number of voters approving 2 alternatives (then, if ties remain, 3 alternatives, etc.).

Monroe: Considers all ways of assigning each voter to one alternative in committee W, such that every $a \in W$ is assigned to between $\lfloor \frac{n}{k} \rfloor$ and $\lceil \frac{n}{k} \rceil$ voters. The score of W is the number of voters assigned to an alternative that they approve. $\mathcal{F}^{\text{Monroe}}$ selects the committee with the highest score. Monroe satisfies the Unanimity and JR axioms [138].

Greedy Monroe: $\mathcal{F}^{\text{Greedy M.}}$ constructs a winning committee iteratively beginning with an empty committee W. At each step, the alternative that is approved by the most unassigned voters is assigned to either $\lfloor \frac{n}{k} \rfloor$ or $\lceil \frac{n}{k} \rceil$ voters and added to W. Greedy Monroe satisfies the Unanimity and JR axioms [138].

Minimax Approval (MAV): $\mathcal{F}^{\text{MAV}}$ selects the committee that minimizes the maximum Hamming distance between any voter's approved alternatives and the winning committee.

Method of Equal Shares: $\mathcal{F}^{\text{MES}}$ uses two phases. In the first phase, each voter has a budget of $\frac{k}{n}$. Proceed for up to k rounds: Adding a voter to the committee has a cost of 1, which can be split between many voters. In each round, consider alternatives A_r which are not in the committee and are approved of by voters that have a remaining budget summing to at least 1. If A_r is empty, go to phase 2. Otherwise, select $a \in A_r$ such that each voter approving of a must spend at most ρ to add them to the committee. Add a to the committee, adjust the remaining budget of each voter, and proceed to the next round. In the second phase, many possible rules can be used to fill any remaining spots on the committee. We use the sequential Phragmen rule. MES satisfies the JR and PJR axioms (but not the Core). See Lackner and Skowron for further details [138].

E Pluribus Hugo: Also called "Single Divisible Vote with Least-Popular Elimination"; $\mathcal{F}^{\text{EPH}}$ operates in rounds. In each round, each voter divides a single point evenly between all remaining alternatives of which they approve. Alternatives are ranked in order of total summed points from all voters. The two alternatives with the lowest number of points are compared: the one receiving the fewest approvals overall is eliminated. Rounds of elimination continue until k alternatives remain. EPH satisfies the Strong Pareto Efficiency axiom [185]

Random Serial Dictator: $\mathcal{F}^{\text{RSD}}$ selects a single voter to serve as dictator. The winning committee is exactly the set of winners which that voter approves of. The method is named on the expectation that it will be used over many rounds (e.g. different dictators are chosen serially).

Random: $\mathcal{F}^{\text{Random}}$ selects a set of k alternatives uniformly at random and declares them winner with no regard for voter preferences.

Min: $\mathcal{F}^{\text{Min}}$ is a baseline which always selects the committee which violates the fewest possible axioms, breaking ties lexicographically.

Max: $\mathcal{F}^{\text{Max}}$ is a baseline which always selects the committee which violates the largest possible number of axioms, breaking ties lexicographically.

Based on the definitions of these rules, we can loosely categorize them by which of the goals outlined by Faliszewski et al. they aim to satisfy [86, 83]. This categorization is not concrete but only roughly indicates similarities in purpose of the rules we study. We include variations of a rule in the same grouping as the original rule.

Individual Excellence: $\mathcal{F}^{Borda}$, $\mathcal{F}^{SNTV}$, $\mathcal{F}^{Bloc}$, $\mathcal{F}^{EPH}$.

Diversity: $\mathcal{F}^{SNTV}$, $\mathcal{F}^{CC}$.

Proportionality: $\mathcal{F}^{STV}$, $\mathcal{F}^{PAV}$, $\mathcal{F}^{Monroe}$, $\mathcal{F}^{CC}$, $\mathcal{F}^{MES}$, $\mathcal{F}^{EPH}$.

These categories are not mutually exclusive; for example, the Chamberlin-Courant score of a committee is larger when the committee represents a wide range of voters (diversity) and is also high when the committee represents voters proportionally (as many voters then have some non-zero representation). We note that the Minimax Approval rule and the two randomized rules do not neatly fit into any category: MAV considers alternatives as sets, rather than individuals, but focuses only how satisfactory the winning committee is to a single voter rather than measures more obviously related to diversity or proportionality. Random Serial Dictatorship and Random committees are included, not on their own merit, but as a baseline against which we compare other rules.

5.2.2 Axioms

We will use each of the following axioms in our experiments. Note that we use definitions of the standard single-winner Condorcet Winner and Loser axioms which have been extended to a multi-winner setting.

Due to the nature of our experimental framework (described in Section 5.3), we consider only *intraprofile* axioms – axioms for which a violating rule can be identified using only a preference profile and the output of the rule on that profile. As with voting rules, we can loosely categorize axioms by whether they focus on individual excellence, or whether they aim to describe diversity or proportionality constraints. In categorizing axioms, we find the boundary between diversity and proportionality is often blurred and merge the two categories.

Condorcet Winner A Condorcet committee C is one in which all $x \in C$ and for all $y \in A \setminus C$, the majority of voters prefer x to y. A voting rule satisfies the Condorcet Winner axiom if, whenever one exists, it returns a Condorcet committee [97].

Condorcet Loser A Condorcet Losing committee L is one in which all $x \in L$ and for all $y \in A \setminus L$, the majority of voters prefer y to x. A voting rule satisfies the Condorcet Loser axiom if it never returns L [97].

Dummett's Condition If there is a group of $\frac{\ell \cdot n}{k}$ voters that all rank the same ℓ alternatives on top, these ℓ alternatives are in the winning committee [73].

Fixed Majority If there exists a set of alternatives C, $|C| = k$ and a set of voters $X \subseteq V$ with $|X| > \frac{n}{2}$ that all rank each alternative in C above each alternative not in C then the winning committee is C [66, 77].

Local Stability For $q = \lceil \frac{n}{k} \rceil$, a committee C violates local stability if there exists a subset of voters V^* with $|V^*| \geq q$ and an alternative $x \notin C$ such that every voter in V^* prefers x to all members of C [19].

Majority Winner If $\lceil \frac{n}{2} \rceil$ or more voters rank alternative x first in their ballot, then x is in the winning committee [89].

Majority Loser If $\lceil \frac{n}{2} \rceil$ or more voters rank alternative x last in their ballot, then x is not in the winning committee.

Solid Coalitions If at least $\frac{n}{k}$ voters rank some alternative x first, then x should be in the winning committee [77].

Strong Pareto Efficiency A committee C dominates C' if every voter approves at least as many alternatives in C as they approve in C', and at least one voter approves strictly more alternatives in C than in C'. A voting rule satisfies Strong Pareto Efficiency if it never returns a dominated committee [138].

Strong Unanimity If every voter ranks the same k alternatives on top, then those alternatives form the winning committee [77].

Core A committee C is "in the core" if for each non-empty subset of $\mathcal{V} \subseteq V$ of voters, and each non-empty subset of alternatives $T \subseteq A$ with

$$\frac{|T|}{k} \leq \frac{|V|}{n}$$

there is a $v_i \in V$ such that $|App(v_i) \cap T| \leq |App(v_i) \cap W|$. That is, v_i approves of at least as many alternatives in W as they approve of in T. A voting rule satisfies the Core if it always returns a committee that is in the core [138].

For our final two axioms we use the concept of ℓ-cohesiveness [138]:

Definition 5.2.3. *For $\ell \geq 1$, a group of voters is ℓ-cohesive if,*

1. $|V| \geq \ell \cdot \frac{n}{k}$
2. $|\bigcap_{i \in V} App(v_i)| \geq \ell$

That is, a group V is ℓ-cohesive if it (1) contains at least an $\frac{\ell}{k}$ portion of voters, and (2) every voter in the group approves of some set of at least ℓ alternatives. Lackner and Skowron argue that such a group should, intuitively, be represented by at least ℓ seats in the final committee. They show this condition may be impossible to satisfy and instead describe easier extensions of the idea.

Justified Representation (JR) For a winning committee C, it is the case that every 1-cohesive group of voters $\mathcal{V}$ contains a v_i who approves of at least one member of C; i.e. $|W \cap App(v_i)| \geq 1$ [138].

Extended Justified Representation (EJR) For a winning committee C, it is the case that *every* ℓ-cohesive group of voters $\mathcal{V}$ contains a v_i that approves of at least ℓ winners, for $1 \leq \ell \leq k$ [138].

There are many known relationships between these axioms. For example, in the single winner setting, a Majority winner is also a Condorcet winner. In the multi-winner setting, this means that a Majority winner must always be in a Condorcet winning set (though it is possible that either or both of a Majority winner and Condorcet winning set do not exist). Another relationship is between the Fixed Majority set and the Condorcet winning set: If the Fixed Majority set exists, it is also the Condorcet winning set. As well, if a committee satisfies the Core axiom, it also satisfies EJR. And if it satisfies EJR, it also satisfies the more basic JR axiom[138].

Many relationships between axioms can be subtle and only relevant under certain conditions. Learning strong axioms as well as axioms which are strictly easier to satisfy can be beneficial to the learning process and we therefore make no effort to avoid redundancy in axioms. Just as multiple redundant features can improve learning outcomes (as we show in Section 5.5), learning simpler axioms such as Justified Representation may be a useful "stepping stone" for a function on its way to learning the stronger Core axiom.

Each of the axioms we consider can also be informally categorized based on the priority it describes.

Individual Excellence: Majority Winner/Loser, Condorcet Winner/Loser, Pareto, Fixed Majority, Strong Unanimity

Diversity/Proportionality: Solid Coalitions, Dummett's Condition, Local Stability, (Extended) Justified Representation, Core

5.3 Evaluating Voting Rules

We take a data-driven approach to evaluating voting rules in a method similar to that advocated by d'Eon and Larson [67]. In two ways we use the empirical behaviour of voting rules, i.e. the winners each rules chooses, to compare rules against each other. We first define a measure of how frequently axioms are violated. This provides both some idea of "quality" which we can use to compare rules across a range of individual preference profiles and to explore violation rates of an individual or set of axioms.

Subsequently we define a difference metric which explicitly measures the overlap in winners/committees elected by two different rules or groups of rules. This difference metric enables us to identify whether two rules with the same axiom violation rate are actually electing similar winners. Put differently, if two rules have a very low axiom violation rate we want to know whether they are identifying the same committee using different approaches, or whether they are identifying highly distinct committees which both happen to result in good axiomatic properties.

5.3.1 Axiom Violation Rate

While axioms are binary properties that either are, or, are not violated we can also consider the frequency with which they are violated to establish a measure of *degrees of violation*. Here we develop a metric of measuring how often a specific rule violates a set of axioms on some set of preference profiles. This metric allows us to understand both how often a specific voting rule violates certain axioms, as well as determining whether violations are more common on specific preference distributions.

As we focus exclusively on *intraprofile axioms* we can determine whether an axiom is violated by a preference profile using only voting rule $\mathcal{F}$ and the profile itself. The procedure for doing this depends upon the axiom itself; for example, determining whether the Majority axiom is violated requires (1) identifying if a Majority winner exists, and (2) if so, checking if the Majority winner is the winner selected by the $\mathcal{F}$.

If $\mathcal{F}$ violates some axiom A we say $\mathbb{I}(\mathsf{A}, \mathcal{F}) = 1$. Otherwise $\mathbb{I}(\mathsf{A}, \mathcal{F}) = 0$. If A is violated by a specific preference profile $\mathcal{P}$ and committee c, we say $\mathsf{A}(\mathcal{P}, c) = 1$. If A is not violated, $\mathsf{A}(\mathcal{P}, c) = 0$. One of the ways in which we study voting rules through our experiments is the frequency with which a particular social choice function is violated given some set of preference profiles. For a set of axioms $\mathbb{A}$ and set of preference profiles $\mathbb{P}$, the axiom violation rate (AVR) of $\mathcal{F}$ is:

$$\mathrm{AVR}(\mathcal{F}, \mathbb{P}, \mathbb{A}) = \frac{1}{|\mathbb{A}||\mathbb{P}|} \sum_{\mathsf{A} \in \mathbb{A}} \sum_{\mathcal{P} \in \mathbb{P}} \mathsf{A}(\mathcal{P}, \mathcal{F}(\mathcal{P}))$$

It is important to remember that an axiom violation rate of 0 does not prove that $\mathcal{F}$ *satisfies* every axiom; it only indicates that the axioms are not violated on the profiles in $\mathbb{P}$.

5.3.2 Difference Between Rules

We also measure the similarity between winners selected by each rule. This tells us whether two rules with low AVR tend to elect the same winners, or whether the rules find distinct approaches to satisfying the axioms. Further, we can use this to gauge the difference between rules with low AVR and rules with high AVR – are their chosen winners highly disjoint or do they tend to elect only slightly different sets of alternatives?

We define the similarity between rules on a single profile as the fraction of alternatives that both rules elect plus the fraction of alternatives that neither rule elects and take the complement to find a difference between rules. Let

$$\cap_P^+(\mathcal{F}^1, \mathcal{F}^2) = \mathcal{F}^1(P) \cap \mathcal{F}^2(P),$$

and

$$\cap_P^-(\mathcal{F}^1, \mathcal{F}^2) = (M \setminus \mathcal{F}^1(P)) \cap (M \setminus \mathcal{F}^2(P)).$$

The lowest number of winners which $\mathcal{F}^1$ and $\mathcal{F}^2$ must have in common is equal to the maximum number of winners which are distinct; $min(| \cap_P^+ (\mathcal{F}^1, \mathcal{F}^2)|) = max(2k - m, 0)$. See this by observing that when $k < m-k$ it is possible to have two completely disjoint sets of winners. When the number of winners is greater than half the number of alternatives, we can image $\mathcal{F}^1$ selecting k winners and $\mathcal{F}^2$ selecting the remaining $m - k$ alternatives as winners before needing to overlap with $\mathcal{F}^1$ for the remaining $k - (m - k)$ winners, for a total overlap of at least $2k - m$ winners. Similarly, for unelected alternatives, $min(| \cap_P^- (\mathcal{F}^1, \mathcal{F}^2)|) = max((m - k) - k, 0)$.

However, as the minimum number of shared winners grows, the minimum number of shared losers shrinks (i.e. It is always possible that either the sets of winners or the sets of losers do not have any overlap). The minimum simultaneous size of these sets sums to $|m - 2k|$. Equivalently, the maximum number of differences between $\cap_P^+(\mathcal{F}^1, \mathcal{F}^2)$ and $\cap_P^-(\mathcal{F}^1, \mathcal{F}^2)$ is $m - |m - 2k|$. We use this as a scaling factor to ensure that the maximum value of the difference between two rules is always 1.

$$\texttt{diff}(\mathcal{F}^1, \mathcal{F}^2, P) = \frac{m}{m - |m - 2k|}(1 - (\frac{| \cap_P^+ (\mathcal{F}^1, \mathcal{F}^2)|}{m} + \frac{| \cap_P^- (\mathcal{F}^1, \mathcal{F}^2)|}{m}))$$

We extend this notation to sets of profiles:

$$\texttt{diff}(\mathcal{F}^1, \mathcal{F}^2, \mathbb{P}) = 1 - \frac{1}{m|\mathbb{P}|} \sum_{P \in \mathbb{P}} | \cap_P^+ (\mathcal{F}^1, \mathcal{F}^2)| + | \cap_P^- (\mathcal{F}^1, \mathcal{F}^2)|$$

Finally, we extend this to sets of voting rules. We compare sets of voting rules to allow us to measure both the internal cohesiveness of groups such as positional scoring rules, as well as to measure the similarity between different groups of rules.

$$\texttt{diff}(\mathbb{F}^1, \mathbb{F}^2, \mathbb{P}) = \frac{1}{|\mathbb{F}^1||\mathbb{F}^2|} \sum_{\mathcal{F}^1 \in \mathbb{F}^1} \sum_{\mathcal{F}^2 \in \mathbb{F}^2 \setminus \{\mathcal{F}^1\}} \texttt{diff}(\mathcal{F}^1, \mathcal{F}^2, \mathbb{P})$$

Thus, $\texttt{diff}(\mathbb{F}^1, \mathbb{F}^2, \mathbb{P})$ is the difference between two sets of voting rule outputs on some set of preference profiles. A value of 0 indicates that two sets of rules always elect the exact same sets of alternatives while a value of 1 indicates that the sets of rules are maximally different. We can use this to establish the similarity of rules in general settings but can also ask whether rules are more similar to each other on specific preference distributions.

5.4 Procedure for Learning Voting Rules

Here we describe how training data is generated and how our models use the data to learn and elect winners. Our approach to learning voting rules exclusively relies on generating training data which informs the model being trained about what a “good” voting rule looks like – whether that means learning to replicate an existing rule or to elect winners which avoid violating axioms. We provide specific details on training/testing parameter values used for each experiment in Section 5.5 and Section 5.6 and describe the high-level process of learning here.

5.4.1 Generating Data

When a model is trained or tested, it requires some dataset $\mathbb{D} = (\mathbb{P}, C)$ of preference profiles and target winners. The profiles $\mathbb{P}$ are sampled from a preference distribution D which is set as an experimental parameter; each experiment considers several of the preference distributions described in Chapter 2. Target winners are determined based on whether the goal is to approximate an existing voting rule, or to create a model which is itself a novel rule. In both cases, the approach is the same regardless of whether the winning committee includes 1 alternative or as many as $m-1$.

Consider the following preference profile with 4 voters and 4 alternatives which we use as a running example through this section:

$$\mathcal{P}' = \begin{array}{ll} v_1 : a_1 \succ a_2 \succ a_3 \succ a_4 & \quad v_2 : a_2 \succ a_1 \succ a_3 \succ a_4 \\ v_3 : a_3 \succ a_4 \succ a_1 \succ a_2 & \quad v_4 : a_1 \succ a_2 \succ a_4 \succ a_3 \end{array}$$

When the learning target is an existing voting rule $\mathcal{F}$, the target output for each preference profile in the dataset is simply the output of $\mathcal{F}$ on that profile. That is,

$$\mathbb{D} = \{(\mathcal{P}, \mathcal{F}(\mathcal{P}))\ \forall\ \mathcal{P} \in \mathbb{P}\}$$

If our learning target is the Borda rule with $k = 2$ winners, the alternatives in our example profile $\mathcal{P}'$ receive – in lexicographic order – (9, 7, 5, 3) points. We would then add the pair $(\mathcal{P}', \{a_1, a_2\})$ to $\mathbb{D}$.

When the learning target is not an existing voting rule we instead aim to minimize the axiom violation rate on some set of axioms $\mathbb{A}$. Given a set of preference profiles $\mathbb{P}$, we find the committee c which violates the fewest axioms.

$$\mathbb{D} = \{(\mathcal{P}, \arg\min_{c} \sum_{\mathsf{A} \in \mathbb{A}} \mathsf{A}(\mathcal{P}, c)) \ \forall \ \mathcal{P} \in \mathbb{P}\}$$

Say we aim to learn the axioms $\mathbb{A} = \{\texttt{Condorcet}, \texttt{Majority Winner}\}$, again with $k = 2$. We perform a brute force search over all committees of size 2. There is no Majority winner in $\mathcal{P}'$ so we can never satisfy the Majority axiom. However, there is a Condorcet winning set, $\{a_1, a_2\}$, which is the unique committee which satisfies the fewest of our target axioms. Thus, in this case we would also add the pair $(\mathcal{P}', \{a_1, a_2\})$ to $\mathbb{D}$.

When there are multiple committees with minimal axiom violations, we break ties lexicographically. For committees of size larger than 1, we do this by comparing the lowest numbered alternative in each committee and selecting the committee with the lowest low-numbered alternative. If this is tied, we compare the second lowest alternatives, etc. Note that we treat these two possible learning types as mutually exclusive: We are either learning to approximate an existing rule, or we are learning to minimize axiom violations, but never both simultanesouly.

5.4.2 Encoding Data

Input Data

A network could be trained directly from the raw dataset, taking as input a complete preference profile. For n voters and m alternatives, this would require $n \cdot m$ inputs to the network and would only allow the network to work with elections of a fixed number of voters. We are also interested in what information is required to learn specific rules – e.g. information about how alternatives are ranked, or pairwise comparisons between alternatives. We cannot study this while using a complete preference profile as input. Thus, we transform each profile $\mathcal{P}$ into three distinct types of features whose size is a function only of the number of alternatives.

$\mathcal{M}^{\text{majority}}$: An $m \times m$ matrix where the entry at (i, j) is 1 if a_i is preferred by a majority of voters in $\mathcal{P}$ to a_j.

$\mathcal{M}^{\text{weighted}}$: An $m \times m$ matrix where the entry at (i, j) contains the number of voters in $\mathcal{P}$ ranking a_i over a_j.

$\mathcal{M}^{\textbf{ranked}}$: An $m \times m$ matrix where the entry at (i, j) is the number of voters in $\mathcal{P}$ ranking a_i in position j.

The preference profile of our running example is transformed into the three matrices below. We are interested in which data is necessary to learn each existing single winner rule so we do not always use every input matrix. Rather, we explore the efficacy and necessity of all sets of input matrices. Each of these matrices is discussed further in Chapter 2.

-	1	1	1
0	-	1	1
0	0	-	1
0	0	0	-

$\mathcal{M}^{\text{majority}}$: The value at (i, j) is 1 if, and only if, a majority prefer a_i over a_j.

-	3	3	3
1	-	3	3
1	1	-	3
1	1	1	-

$\mathcal{M}^{\text{weighted}}$: The value at (i, j) is the number of voters preferring a_i over a_j.

2	1	1	0
1	2	0	1
1	0	2	1
0	1	1	2

$\mathcal{M}^{\text{ranked}}$: The value at (i, j) is the number of voters ranking a_i in rank j.

Before being given to the network each of the feature matrices is flattened into a one dimensional list and normalized such that the minimum and maximum values in each matrix are 0 and 1 respectively. $\mathcal{M}^{\text{majority}}$ and $\mathcal{M}^{\text{weighted}}$ have their main diagonals removed, as they contain no information. As well, alternatives are randomly renamed to minimize bias towards particular alternatives[4].

Target Data

The k target winners are encoded as k-hot vectors $\hat{y}$. A set of alternatives is encoded as a list of m elements. All values are 0, except for those corresponding to the indices of the k winning alternatives which are set to 1.

In our running example, under each learning paradigm (learning to approximate the Borda rule, or learning to minimize axiom violations) we added $\{a_1, a_2\}$ to the data. We represent this target committee by the 2-hot vector $y =$ `[1, 1, 0, 0]`.

[4]For example, our identity distribution (where all voters have identical preferences) always generates preferences in lexicographic order. Learning on such data would result in networks that elect alternatives a_1 through a_k rather than actually learning on the underlying distribution.

5.4.3 Decoding Network Output

Each network that we train has m output nodes, one corresponding to each alternative. When electing k alternatives, we create a k-hot vector y by setting the value of the k highest output nodes to 1, and the remaining nodes to 0. Each alternative with an index corresponding to a value of 1 is deemed a winner.

As example, a network may output $\hat{y} =$ `[0.87, 0.18, 0.02, 0.53]` (note that these values are chosen arbitrarily). These raw values are used during training to compute losses. During evaluation, if $k = 2$ this becomes the 2-hot vector `[1, 0, 0, 1]` which is interpreted as a winning committee of $\{a_1, a_4\}$.

5.4.4 Training Networks

Networks receive a dataset composed of pairs of flattened matrices and k-hot target vectors. These are used as input and target data. After informal comparisons of several standard loss functions[5], we train networks using L_2 loss in the setting of Section 5.5 and L_1 loss in the setting of Section 5.6. Each loss function is defined below.

$$L_1(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{m|\mathbb{D}|} \sum_{i=1}^{|\mathbb{D}|} \sum_{j=1}^{m} |y_{ij} - \hat{y}_{ij}|$$

$$L_2(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{m|\mathbb{D}|} \sum_{i=1}^{|\mathbb{D}|} \sum_{j=1}^{m} (y_{ij} - \hat{y}_{ij})^2$$

The L_1 loss of our running example with $y =$ `[1, 1, 0, 0]` and $\hat{y} =$ `[0.87, 0.18, 0.02, 0.53]` as above is then,

$$\begin{aligned} L_1(\{y\}, \{[0.87, 0.18, 0.02, 0.53]\}) &= \frac{1}{4}(|y_0 - \hat{y}_0| + |y_1 - \hat{y}_1| + |y_2 - \hat{y}_2| + |y_3 - \hat{y}_3|) \\ &= \frac{1}{4}(|1 - 0.87| + |1 - 0.18| + |0 - 0.02| + |0 - 0.53|) \\ &= 0.375 \end{aligned}$$

[5] We compared the following loss functions in each of our training settings: L_1, L_2, Poisson, Hinge, Cosine Similarity, Cross-entropy.

The gradient of the loss at each output node is used to update the weights of each edge in the network, moving backwards layer by layer. Adjustments in each layer reduce the loss on subsequent predictions. We refer to Goodfellow et al. for further discussion of gradient calculation and a thorough explanation of the back-propagation algorithm [107]. Networks are trained in this manner on larger datasets in order to act as approximators of existing rules and to be entirely new rules.

5.5 Learning Existing Rules

We initially focus our attention on training networks to replicate (or, more precisely, *approximate*) existing voting rules. This serves two purposes: First, we gain some understanding of *how* to learn existing rules, e.g. what features are useful and what loss function is best. Second, this grants insight into similarities between existing voting rules; some rules may be learnable with fewer features than expected and certain sets of rules may be more cohesive than previously known.

In this section we begin by describing the training pipeline used to learn the existing rules we study. We then demonstrate how accurate our learning procedure is, based on which features are used in training. We show a comparison between rules which highlights that certain groups of rules (e.g. Fishburn's C1 rules, or positional scoring rules) are much more cohesive than other groups and end the section with some remarks on how best to use neural networks to learn voting rules.

5.5.1 Training

We train networks following the process described in Section 5.4. In this section we focus exclusively on single winner rules and train 50 networks each on all combinations of the following parameters:

- Training on each of the seven non-empty subsets of feature combinations from $\{\mathcal{M}^{\text{majority}}, \mathcal{M}^{\text{weighted}}, \mathcal{M}^{\text{ranked}}\}$.
- Targeting the output of each single winner voting rule listed in Section 5.2.
- Each preference distribution $\mathcal{D}$ from:
  - Urn

– Impartial Culture
– Impartial Anonymous Culture
– Mallows
– Single Peaked (Conitzer)
– Single Peaked (Walsh)
– Euclidean (3 dimensions, cube topology, uniform random placement); abbreviated as E(3, C, U).
– Euclidean (3 dimensions, cube topology, gaussian placement); abbreviated as E(3, C, G).
– Euclidean (3 dimensions, ball topology, uniform random placement); abbreviated as E(3, B, U).
– Euclidean (3 dimensions, ball topology, gaussian placement); abbreviated as E(3, B, G).

In all cases, we use $n = 50$ voters and $m = 5$ alternatives and generate a training set of 10000 examples from each preference distribution. Networks have 4 fully connected hidden layers, each with 20 nodes using a rectified linear unit activation function. Each network is trained for 200 epochs or until 10 epochs pass without loss decreasing by more than 0.001. We use the Adam optimizer with L_2 loss.

We generate three test sets on which we evaluate trained rules. During evaluation, the alternative output most commonly across each of the 50 trained networks is treated as the predicted value of the networks. Each test set has several profiles sampled from a range of preference distributions. Two of the test sets are generated using an identical process while the third is sampled from the same distributions but excludes any profiles having a Condorcet winner[6]. We refer to these as the primary, secondary, and No Condorcet test sets respectively. Each test set samples profiles from the following distributions for a total size of 440 profiles:

- 20 profiles from Impartial Culture
- 20 profiles from Single Peaked Conitzer
- 20 profiles from Single Peaked Walsh

[6]For single-peaked profiles, each profile has at least two alternatives which are preferred by a majority to all other alternatives and are themselves each preferred to the other by an equal number of voters.

- 20 profiles from Single-Crossing
- 20 profiles from Group-Separable
- 20 profiles from Euclidean (1D, cube topology, uniform random placement)
- 20 profiles from Euclidean (2D, cube topology, uniform random placement)
- 20 profiles from Euclidean (3D, cube topology, uniform random placement)
- 20 profiles from Euclidean (5D, cube topology, uniform random placement)
- 20 profiles from Euclidean (10D, cube topology, uniform random placement)
- 20 profiles from Euclidean (20D, cube topology, uniform random placement)
- 20 profiles from Euclidean (2D, sphere topology, uniform random placement)
- 20 profiles from Euclidean (3D, sphere topology, uniform random placement)
- 20 profiles from Euclidean (5D, sphere topology, uniform random placement)
- 80 profiles from Urn
- 80 profiles from Mallows

5.5.2 Learned Rule Accuracy

In Table 5.2, and Table 5.3 we show the test accuracy of training networks averaged over all training distributions (In order to fit into the margins with a legible font size we have listed Fishburn's C1 and C2 rules in the first table and Fishburn's C3 rules in the second table). While using all features is typically most accurate, or nearly most accurate, we readily observe that rules are generally learned according to the class they are in. That is, Fishburn's C1 rules have highest accuracy with access to $\mathcal{M}^{\text{majority}}$ features, C2 rules have highest accuracy when using $\mathcal{M}^{\text{weighted}}$ features, and positional scoring rules have high accuracy when using $\mathcal{M}^{\text{ranked}}$ features.

$\mathcal{F}^{\text{Borda}}$ is a noteworthy example, being the only scoring rule that is not in C3. While it has highest accuracy using only $\mathcal{M}^{\text{weighted}}$ features, it also tends to have higher accuracy when using $\mathcal{M}^{\text{ranked}}$ features.

Interestingly, while $\mathcal{M}^{\text{weighted}}$ contains strictly more information than $\mathcal{M}^{\text{majority}}$ *every* C1 rule that we consider is learned much more accurately when using $\mathcal{M}^{\text{majority}}$ features.

Target Rule	M	W	R	MW	MR	WR	MWR
Banks*	0.87	0.74	0.71	0.87	0.87	0.72	**0.87**
Bipartisan Set*	0.86	0.74	0.7	0.86	0.86	0.7	**0.86**
Condorcet*	0.89	0.73	0.7	0.89	**0.89**	0.71	0.89
Copeland*	0.88	0.75	0.72	0.88	**0.88**	0.72	0.88
GOCHA*	0.88	0.74	0.7	0.88	0.88	0.71	**0.88**
Llull*	0.88	0.75	0.71	0.88	0.88	0.72	**0.89**
Slater*	0.87	0.74	0.71	0.88	0.88	0.72	**0.88**
Top Cycle*	0.88	0.73	0.7	0.88	0.88	0.71	**0.88**
Uncovered Set*	0.88	0.74	0.71	0.88	0.88	0.71	**0.88**
Beat Path*	0.85	0.75	0.71	0.86	0.86	0.71	**0.86**
Blacks*	0.84	0.76	0.74	**0.84**	0.84	0.74	0.84
<u>Borda</u>	0.75	**0.82**	0.79	0.78	0.76	0.8	0.78
B-M Faceoff*	0.86	0.75	0.71	0.86	0.86	0.72	**0.86**
C-Global-Borda*	0.86	0.76	0.73	0.86	0.86	0.73	**0.86**
L-T Voting*	0.86	0.75	0.71	0.86	0.86	0.71	**0.86**
Minimax*	0.85	0.75	0.71	0.86	0.86	0.71	**0.86**
Raynaud*	0.85	0.74	0.7	0.85	**0.85**	0.71	0.85
S. Stable Voting*	0.85	0.75	0.71	0.85	0.85	0.72	**0.85**
Split Cycle*	0.86	0.75	0.71	0.86	0.86	0.71	**0.86**
Stable Voting*	0.85	0.75	0.72	0.85	0.85	0.72	**0.85**

Table 5.2: Accuracy on the primary test set from learning each voting rule in Fishburn's C1 (red) and C2 (blue) classes, averaged over all training distributions for each combination of features. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold. Features are labelled by as M, W, R respectively for $\mathcal{M}^{\text{majority}}$, $\mathcal{M}^{\text{weighted}}$, and $\mathcal{M}^{\text{ranked}}$ features.

This highlights two practical notes: (1) Networks may not be fully utilizing the training data they are being given, suggesting that larger networks and/or more data may improve performance. (2) Using simple but informative features is useful and can make learning easier than using only more complex features.

Table 5.2, and Table 5.3 show results on our primary test set. In Appendix C we show tables with the same structure containing results for our other two test sets. As expected, both identically sampled test sets have similar results. The test set containing no Condorcet winners exhibits the same quantitative patterns but has generally lower accuracy.

We can also break results down by training distribution. In Table 5.4 and Table 5.5 we

Target Rule	M	W	R	MW	MR	WR	MWR
Anti-Plurality	0.42	0.51	0.72	0.47	0.67	**0.73**	0.68
Baldwin*	0.86	0.74	0.71	0.86	0.86	0.71	**0.86**
Benham*	0.83	0.73	0.71	0.84	**0.84**	0.72	0.84
Bracket Voting	0.82	0.72	0.71	0.82	0.82	0.72	**0.83**
Bucklin	**0.7**	0.67	0.69	0.7	0.68	0.7	0.69
Cond. Plurality*	0.82	0.72	0.73	0.82	0.84	0.73	**0.84**
Coombs	0.82	0.73	0.69	**0.82**	0.79	0.69	0.8
C-Local-Borda*	0.87	0.75	0.72	**0.87**	0.87	0.72	0.87
Daunou*	0.82	0.73	0.72	0.82	0.82	0.73	**0.83**
Instant Runoff	0.75	0.68	0.72	0.75	0.76	0.73	**0.77**
Knockout Voting	0.85	0.75	0.72	0.85	0.85	0.73	**0.85**
Plurality	0.64	0.61	0.88	0.64	0.81	**0.89**	0.83
S. Bucklin	0.65	0.66	0.72	0.69	0.73	0.73	**0.73**
Strict Nanson*	0.86	0.75	0.71	0.86	0.86	0.71	**0.86**
Superior Voting*	0.84	0.73	0.71	0.85	0.85	0.72	**0.85**
T. Alt. GOCHA*	0.85	0.74	0.71	0.85	0.85	0.72	**0.86**
T. Alt. Top Cycle*	0.84	0.73	0.71	0.84	0.84	0.72	**0.84**
Weak Nanson*	0.86	0.76	0.72	0.86	0.86	0.72	**0.86**
Weighted Bucklin	**0.75**	0.72	0.74	0.75	0.74	0.74	0.75

Table 5.3: Accuracy on the primary test set from learning each voting rule in Fishburn's C3 class, averaged over all training distributions for each combination of features. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold. Features are labelled by as M, W, R respectively for $\mathcal{M}^{\text{majority}}$, $\mathcal{M}^{\text{weighted}}$, and $\mathcal{M}^{\text{ranked}}$ features.

show the accuracy of each rule averaged over all feature sets for each training distribution. Results shown are for the primary test set and other test sets are shown in Appendix C. The most striking observation is that with most target rules *most* training distributions generalize to the test profiles with a similar accuracy, except Single Peaked profiles – e.g. the Banks rule is approximated with 87% to 93% accuracy by networks trained individually on each single distribution excluding Single Peaked Conitzer and Walsh. Networks trained on these distributions resulted in 43% and 33% accuracy respectively on the primary test set.

This reflects the highly structured nature of Single Peaked preferences. While other distributions appear to generate preferences similar to each other Single Peaked preferences

Target Rule	IC	Mallows	Urn	IAC	SP Conitzer	SP Walsh	E(3CU)	E(3BU)	E(3CG)	E(3BG)
Banks*	0.91	0.93	0.87	0.92	0.43	0.33	0.93	**0.93**	0.92	0.92
Bipartisan Set*	0.89	0.91	0.86	0.9	0.45	0.36	0.91	**0.91**	0.91	0.9
Condorcet*	0.9	0.93	0.87	0.91	0.43	0.34	**0.94**	0.94	0.94	0.92
Copeland*	0.92	0.93	0.87	0.92	0.45	0.35	**0.94**	0.94	0.93	0.93
GOCHA*	0.91	0.93	0.87	0.92	0.43	0.34	**0.93**	0.93	0.92	0.92
Llull*	0.92	0.93	0.87	0.93	0.44	0.34	**0.94**	0.94	0.93	0.93
Slater*	0.91	0.92	0.87	0.92	0.44	0.35	**0.93**	0.93	0.92	0.92
Top Cycle*	0.9	0.93	0.87	0.91	0.42	0.33	**0.94**	0.94	0.93	0.92
Uncovered Set*	0.91	0.93	0.87	0.92	0.43	0.34	**0.93**	0.93	0.93	0.92
Beat Path*	0.89	0.9	0.85	0.9	0.44	0.35	**0.92**	0.92	0.91	0.9
Blacks*	0.89	0.91	0.85	0.9	0.45	0.35	0.91	**0.91**	0.9	0.91
Borda	0.91	0.89	0.85	**0.92**	0.41	0.22	0.91	0.92	0.9	0.9
B-M Faceoff*	0.89	0.91	0.86	0.9	0.44	0.35	0.92	**0.92**	0.92	0.9
C-Global-Borda*	0.91	0.92	0.86	0.91	0.46	0.35	0.92	**0.92**	0.91	0.91
L-T Voting*	0.9	0.91	0.86	0.9	0.45	0.35	**0.92**	0.92	0.91	0.9
Minimax*	0.89	0.91	0.85	0.9	0.45	0.35	**0.92**	0.92	0.91	0.9
Raynaud*	0.89	0.9	0.85	0.89	0.44	0.35	0.91	**0.91**	0.91	0.89
S. Stable Voting*	0.89	0.91	0.86	0.89	0.44	0.35	**0.91**	0.91	0.91	0.9
Split Cycle*	0.89	0.91	0.85	0.9	0.45	0.35	**0.92**	0.92	0.91	0.9
Stable Voting*	0.89	0.91	0.86	0.89	0.45	0.35	**0.91**	0.91	0.91	0.9

Table 5.4: Accuracy on the primary test set from learning each voting rule in Fishburn's C1 (red) and C2 (blue) classes, averaged over all feature sets for each training distribution. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold.

provide very little information about how a voting rule should behave on non-Single Peaked preferences.

Target Rule	IC	Mallows	Urn	IAC	SP Conitzer	SP Walsh	E(3CU)	E(3BU)	E(3CG)	E(3BG)
Anti-Plurality	0.65	**0.76**	0.55	0.65	0.33	0.33	0.72	0.73	0.66	0.61
Baldwin*	0.89	0.91	0.85	0.9	0.44	0.35	**0.91**	0.91	0.91	0.9
Benham*	0.88	0.89	0.85	0.88	0.45	0.35	0.9	**0.9**	0.89	0.88
Bracket Voting	0.86	0.87	0.84	0.87	0.45	0.36	0.88	**0.89**	0.88	0.87
Bucklin	0.69	0.81	0.76	0.74	0.36	0.22	0.83	0.83	0.82	**0.83**
Cond. Plurality*	0.87	0.88	0.85	0.88	0.47	0.36	**0.89**	0.89	0.89	0.88
Coombs	0.8	0.88	0.83	0.82	0.44	0.35	**0.88**	0.88	0.88	0.86
C-Local-Borda*	0.91	0.92	0.86	0.92	0.46	0.36	**0.93**	0.93	0.92	0.92
Daunou*	0.87	0.88	0.85	0.88	0.43	0.34	**0.89**	0.89	0.89	0.88
Instant Runoff	0.82	0.82	0.8	0.82	0.45	0.37	**0.83**	0.83	0.83	0.81
Knockout Voting	0.9	0.91	0.86	0.9	0.46	0.36	0.91	**0.91**	0.9	0.91
Plurality	0.84	0.83	0.75	**0.85**	0.54	0.39	0.85	0.85	0.84	0.84
S. Bucklin	0.65	**0.84**	0.73	0.73	0.42	0.31	0.84	0.84	0.84	0.82
Strict Nanson*	0.9	0.91	0.86	0.9	0.44	0.35	0.92	**0.92**	0.91	0.9
Superior Voting*	0.89	0.9	0.85	0.89	0.46	0.35	0.9	**0.9**	0.9	0.89
T. Alt. GOCHA*	0.89	0.9	0.86	0.9	0.45	0.35	**0.92**	0.91	0.91	0.89
T. Alt. Top Cycle*	0.88	0.89	0.85	0.89	0.45	0.35	0.9	**0.9**	0.9	0.88
Weak Nanson*	0.9	0.91	0.86	0.91	0.45	0.35	0.92	**0.92**	0.91	0.91
Weighted Bucklin	0.83	0.85	0.8	0.83	0.36	0.32	**0.86**	0.86	0.86	0.84

Table 5.5: Accuracy on the primary test set from learning each voting rule in Fishburn's C3 class, averaged over all feature sets for each training distribution. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold.

5.5.3 Comparison between Rules

Using the difference metric defined in Section 5.3, we can compare the difference between each individual rule and each major class of rule – C1, C2, C3, Condorcet methods, and positional scoring rules. This metric allows us several insights. First, to the differences between existing groups of rules, e.g. whether the Fishburn classes have some internal consistency in terms of which winners they elect. Second, we can compare the distances between existing rule classes with our learned approximations of rules to identify patterns

	C1	C2	C3	Condorcet	Scoring
C1	0.05	-	-	-	-
C2	0.06	0.05	-	-	-
C3	0.12	0.11	0.16	-	-
Condorcet	0.05	0.05	0.11	0.05	-
Scoring	0.30	0.30	0.31	0.29	0.46

Table 5.6: Distance between each group of **learned networks** approximating rules in the corresponding group on the primary test set.

	C1	C2	C3	Condorcet	Scoring
C1	0.04	-	-	-	-
C2	0.07	0.06	-	-	-
C3	0.13	0.13	0.18	-	-
Condorcet	0.06	0.06	0.13	0.06	-
Scoring	0.34	0.33	0.35	0.33	0.48

Table 5.7: Distance between each group of **existing rules** targeting rules in the corresponding group on the primary test set.

in which type of rule is most learnable. Note that C1, C2, and C3 methods are mutually exclusive but Condorcet methods overlap with each of Fishburn's classes (nearly all C1 and C2 rules are Condorcet-consistent, only some C3 rules) and we use positional scoring rules which fall into the C2 and C3 classes.

In Table 5.6 and Table 5.7 we show the difference between each major rule class on learned and real rules respectively. For both learned and existing rules the trends are the same. C1 rules are able to be computed using only $\mathcal{M}^{\text{majority}}$ which relies on pairwise comparison, much like the definition of the Condorcet criteria. Unsurprisingly, C1, C2, and Condorcet rules all tend have a very low distance from other C1/C2/Condorcet rules (as there is a large overlap in membership of the two groups).

When considering the data in Table 5.7 before rounding, each set of C1, C2, and Condorcet rules are closer to their own class of rules than any other class; this demonstrates a strong level of consistency within the classes. Put differently, this can be seen as evidence that each class captures something meaningful about the winners chosen by a rule that is common to other rules in the same class. As nearly all rules in these classes are Condorcet-consistent, the commonality captured by the classes may simply be that they elect the

Condorcet winners. However, a similar trend holds in Table C.18 where we consider rule outputs on the test set with no Condorcet winners.

What we find most surprising is the large difference between positional scoring rules and all sets of rules (including the difference between each positional scoring rule and other positional scoring rules). The set is quite small, containing only Plurality, Borda, and Anti-Plurality, so the potential for it to be an outlier is high.

That said, across all test sets the two existing rules most different from each other are Plurality and Anti-Plurality. Given that Plurality and Anti-Plurality are very antonymous names for voting rules (not to mention the differences in their definitions: one provides information only about favourite alternatives and the other only about least favourites), it is not altogether surprising that they produce very different outcomes. However, each of the two rules is also quite different from Borda. This highlights that the class of positional scoring rules can exhibit a very wide range of behaviours despite having quite a simple and concise definition.

We also observe that, in general, learned rules are slightly less different from one another than existing rules. This difference is especially pronounced on the test data with no Condorcet winners, shown in Appendix C. This suggests to us that neural networks have a tendency towards similarity with one another; networks are not learning an actual voting rule but only approximating one and tend to do so in a way that produces output similar to networks approximating other rules. This echoes the findings of Burka et al. who show that neural networks trained to learn existing rules typically elect similar alternatives regardless of which rule they target[46].

Finally, we briefly consider differences between individual rules. In Appendix C we include tables showing the pairwise differences between all 39 individual rules. We include 6 tables, one for each of learned and actual rules on each test set. As we have observed, Anti-Plurality and Plurality are each typically quite different from every rule. However, we can also observe other trends. In particular, the family of Bucklin rules: Bucklin, Weighted Bucklin and Simplified Bucklin all tend to be different from other rules as well.

These trends hold for the primary and secondary test sets with learned approximations as well as actual rules. On the test set with no Condorcet winners behaviour changes. The above-stated rules all maintain high differences from other rules; however, the difference between nearly any pair of *real* rules grows quite high. This does not hold with the learned rule approximations. While differences between each pair of models grows, many models remain relatively similar to each other. The required lack of structure on elections with no Condorcet winner greatly magnifies the algorithmic differences between each different rule. The models approximating the rules do not fully learn the specific edge cases of each rule and thus have fewer differences to magnify in this setting.

5.5.4 Takeaways

Our results provide an interesting picture of similarities between rules and the difficulty of learning some rules over others but they also suggest useful adjustments that can improve the learning process. Here we identify some insights from our results and list the motivation for changes we make to our network training procedure in the following section.

- Positional scoring rules exhibit high dissimilarity from each other; as a whole, the group has potential to produce a diverse range of behaviour.
- Learned rule accuracy is reasonably high but not perfect. Since models learning C1 rules are not as accurate when training with only $\mathcal{M}^{\text{weighted}}$ features as they are when training with only $\mathcal{M}^{\text{majority}}$, the models may not be fully utilizing the training data they are being given. Using larger networks and more data may improve performance.
- Using simple but informative features (i.e. $\mathcal{M}^{\text{majority}}$) is useful and can make learning easier than using only more complex features ($\mathcal{M}^{\text{weighted}}$). Thus we keep all of the features that we use, despite some information they provide being redundant.

5.6 Learning Novel Rules

Using what we have learned about learning to adjust our training setup we now move away from the task of learning existing rules and begin to learn novel rules. In this section we use our learning framework to develop novel rules that learn to select winners minimizing axiom violations. While we evaluate our multi-winner rules in the single winner setting, our focus is on the aggregate performance of the rules over varying numbers of winners. Many interesting axioms are defined with the multi-winner setting in mind, leading to committee selection appearing as a more interesting and challenging task. We find that common multi-winner rules frequently violate axioms considered desirable, despite a much lower violation rate being possible. In this section we include only a small subset of aggregate results from our larger set of experiments. Additional results, considering each preference distribution and number of alternatives, are found in Appendix D.

5.6.1 Training Pipeline

We train networks following the process described in Section 5.4 with the goal of learning novel rules that minimize Axiom Violation Rate. In this section we turn our focus to multi-winner rules and train 20 networks each on all combinations of the following parameters:

- Training exclusively on all features; the concatenation of $\mathcal{M}^{\text{majority}}$, $\mathcal{M}^{\text{weighted}}$, and $\mathcal{M}^{\text{ranked}}$.
- Number of alternatives $m \in \{5, 6, 7\}$
- Number of winners $1 \leq k < m$.
- Targeting the output of each multi-winner voting rule listed in Section 5.2.
- Each preference distribution $\mathcal{D}$ from:
  - Impartial Culture
  - Impartial Anonymous Culture
  - Stratification with $w = 0.5$; half of alternatives (rounded down) placed on top.
  - Identity
  - Mallows (with ϕ sampled uniformly at random as described by Boehmer et al., [35]).
  - Urn (with α sampled from a Gamma distribution with shape parameter $k = 0.8$ and scale parameter $\theta = 1$ as described by Boehmer et al. [35].).
  - Single Peaked (Conitzer)
  - Single Peaked (Walsh)
  - 8 Euclidean distributions with each combination of: 3 or 10 dimensions, an underlying Ball (B) or Cube (C) topology, and Uniform (U) or Gaussian (G) placement of voters. Abbreviated as E(Dimension, Topology, Placement).
  - Mixed - a uniformly random mixture of all 16 other distributions.

In all cases, we use $n = 50$ voters generate a training set and a test set of 25000 examples from each preference distribution. Networks have 5 fully connected hidden layers, each with 256 nodes using a rectified linear unit activation function. Each network is trained for 50 epochs or until 20 epochs pass without loss decreasing by more than 0.0001. We use the Adam optimizer with L_1 loss. We reduced the number of training epochs after observing few networks requiring more than 40-50 epochs to finish training and change the loss function after noticing a mild improvement to axiom violation rate. The most significant differences to our networks from the previous section are:

- Larger networks (5 hidden layers of 256 nodes, rather than 4 layers of 20 nodes)

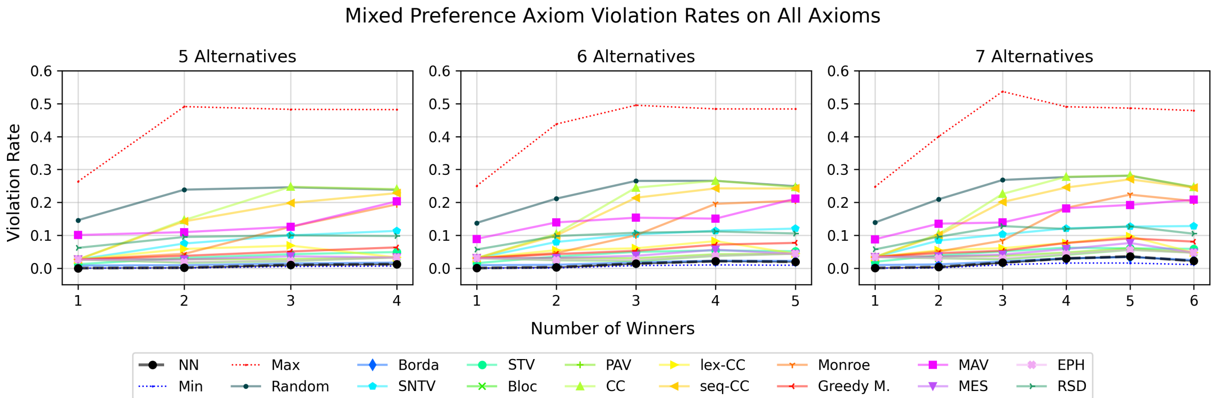


Figure 5.1: Axiom violation rate for each voting rule averaged over all axioms for each number of alternatives on the mixed preference profile sampled uniformly from all individual preference distributions.

- More training data (25000 examples rather than 10000)
- Always using all features.
- Target winners are committees that minimize axiom violations, rather than the output of an existing voting rule.

5.6.2 Minimizing Axiom Violation Rate

Our focus in this section is on training models that act as novel voting rules which minimize axiom violations. We have targeted all axioms listed previously and trained/evaluated over many preference profiles. The first result we show in Figure 5.1 is a very broad view: we show one plot of axiom violation rate for each number of alternatives, showing within each plot the rate for each number of winners. The violation rate is averaged over all axioms we consider and shows results for the preference distribution composed of an even sample of all preference distributions. This is a very high-level perspective; however, the figure does provide some useful initial insights.

First, we see that no rule, even random committees, is anywhere near the maximum possible violation rate (dotted red line). However, some rules (CC, seq-CC) often have nearly as high a violation rate as random committees do, particularly when the number of winners is high. Second, we can see that our trained models (black line) appear to perform extremely well! They consistently have nearly the optimal violation rate (dotted blue line) and appear at least as good as any other rule.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.017**	*.000*	*.000*	.004	**.017**	**.015**	*.000*	*.000*	*.000*	.061	.001	.001	.001	.046	.092
Borda	.021	.001	.004	.004	.021	.125	**0**	.011	**0**	.044	*.000*	*.000*	*.000*	.031	.056
EPH	.040	*.000*	.001	*.000*	.040	.270	.002	.001	**0**	.082	*.000*	*.000*	*.000*	.063	.096
SNTV	.099	**0**	.098	.227	.099	.619	.007	.106	.049	.062	.001	.054	.058	**0**	.012
STV	.048	**0**	.037	.118	.048	.442	.002	.029	**0**	**0**	*.000*	*.000*	.001	**0**	**.001**
Bloc	.039	*.000*	.001	**0**	.039	.254	.002	**0**	**0**	.080	*.000*	*.000*	*.000*	.061	.106
CC	.195	.036	.146	.344	.195	.756	.031	.141	.062	.308	**0**	.084	.091	.232	.301
lex-CC	.061	.005	.007	**0**	.061	.440	.002	.024	**0**	.117	**0**	*.000*	*.000*	.091	.112
seq-CC	.183	.032	.139	.297	.183	.740	.025	.140	.061	.292	**0**	.078	.081	.216	.278
Monroe	.130	.007	.078	.234	.130	.649	.026	.060	**0**	.214	**0**	.002	.006	.180	.231
Greedy M.	.063	.002	.019	.012	.063	.448	.003	.023	**0**	.112	**0**	**0**	**0**	.089	.118
PAV	.043	.001	.001	**0**	.043	.308	.002	.004	**0**	.088	**0**	**0**	**0**	.068	.091
MES	.049	.001	.002	.001	.049	.351	.002	.008	**0**	.096	**0**	**0**	**0**	.075	.095
MAV	.157	.022	.110	.279	.157	.750	.044	.084	**0**	.219	.015	.022	.022	.179	.300
RSD	.105	.008	.056	**0**	.105	.594	.016	.036	**0**	.148	.030	.032	.033	.120	.299
Random	.237	.063	.171	.406	.237	.845	.057	.160	.071	.326	.049	.125	.134	.252	.419

Table 5.8: Axiom violation rates over all test sets of profiles for each voting rule averaged across all axioms for 7 alternatives, all $1 \leq k \leq 6$, and all preference distributions. Bold values indicate the best result of a column, italic values indicate a value rounded down that has a true value strictly below 0.0005 (i.e. at most 12 violations in a test set of 25,000 profiles). Shaded green appears where previous work has shown that the rule satisfies this axiom.

In Table 5.8 we show the raw axiom violation rate for each multi-winner rule that we consider. The table shows the violation rate for 7 alternatives averaged across all preference distributions and number of winners. Green cells indicate that a rule has been shown theoretically to satisfy an axiom (i.e. it should have a violation rate of 0; minor exceptions may occur due to differences in tie-breaking). Values in italics have been rounded to 0. In all cases, we report the mean violation rate across all 20 trained networks. Networks trained on the same parameters are all highly similar to each other: the mean standard deviation over all networks trained on each parameter combination is 0.0002.

In Table 5.8, we see that the strong performance of a trained model does not disappear upon closer investigation. In nearly all axioms, the NN rule has a very similar or a lower violation rate than every other rule. On the axioms that are more focused on electing individually excellent alternatives (Majority Winner through Unanimity in the table), this rings the most true. On some proportional axioms – Solid Coliations and Stability – the NN rule performs much better than the worst rules but is outperformed by a number of rules. While $\mathcal{F}^{\text{Random}}$ typically performs much worse than many rules as we would intuitively expect, the other randomized rule $\mathcal{F}^{\mathrm{RSD}}$ performs surprisingly well by simply following the approvals of a random voter. In fact, compared with several rules (CC, seq-CC, MAV), $\mathcal{F}^{\mathrm{RSD}}$ has a lower violation on all individual excellence axioms and many proportionality axioms.

We now move to an even more detailed views of our results. We visualize first the axiom violation rate for each axiom on each individual preference distribution. Subsequently, we show the violation rates for each rule on the mixed preference distribution.

Differences Between Preference Distributions

In Figure 5.2, we include a subplot for each individual preference distribution with each rule as a series showing the axiom violation rate averaged over all axioms. Echoing our observation from Figure 5.1, it is the case for most preference distributions that all rules have an average AVR much lower than the worst case (dotted red line). We do find a number of additional observations from this view of the data:

- In general, the two distributions which consistently have a low violation rate are IC and IAC, and also have a low worst-case violation rate. These are the two distributions with minimal underlying structure, as opposed to Identity or Single-Peaked preferences which have a very specific structure. That they have low violation rates is unsurprising upon reflection: The chance that some blocs of voters will prefer some shared subset of alternatives (as loosely required by many axioms) is much lower under IC/IAC preferences than under more highly structured preferences.

- While both Single-Peaked distributions show approximately similar trends in axiom violation for rules, Walsh's distribution shows a much higher worst-case violation rate. In light of our previous observation, this can be seen as mildly surprising. Unlike Conitzer's, the Walsh Single-Peaked preferences are sampled uniformly from the entire Single-Peaked domain and referred to as the "Impartial Culture" of Single-Peakedness [220]. However, this more general structure does not translate to a lower violation rate.

- We have used Stratified preferences parameterized such that every voter ranks the same $\lfloor \frac{m}{2} \rfloor$ alternatives above all others. For $m = 7$, this means that with 3 winners all voters approve of the exact same 3 alternatives. This results in the edge case seen in the plot with a high worst-case AVR and strange behaviour from the seq-CC rule.

- Excluding IC and IAC, it appears to be universally the case that the worst-case violation rate occurs with $k = 3$ winners. However, on Single-Peaked and all Euclidean

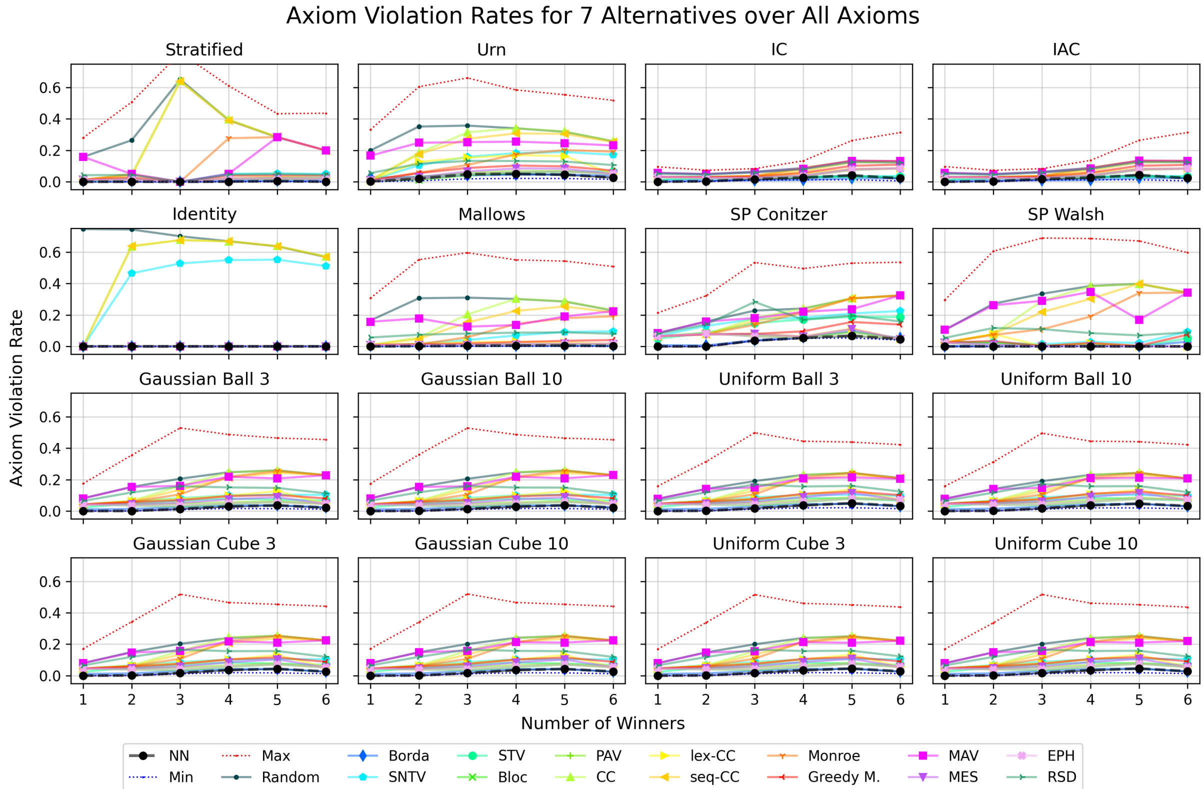


Figure 5.2: Axiom violation rates for each rule under each individual preference distribution for $m = 7$. In all cases our trained model, $\mathcal{F}^{\text{NN}}$, has AVR lower than, or similar to, other rules.

distributions there seems to be a trend of rules having increasing violation rates with a higher number of winners.

Axiom Violation Rate for each Rule

In Figure 5.3, we violation rates for preferences drawn uniformly from all distributions using 7 alternatives. Each axiom is shown as a series with a subplot for each rule. This figure shows a view with more insight into the "behaviour" of each axiom under each rule as the number of winners changes. There are a number of observations that become apparent from the figure:

Figure 5.3: Axiom violation rate with 7 alternatives for each rule on preferences drawn uniformly from all individual preference distributions as the number of winners varies.

- Universally, all existing rules typically have the highest violation rate for the Condorcet Winner axiom. Even on rules such as Borda and EPH, this tends to hold true. However our trained NN rule has a very low Condorcet violation rate, demonstrating the possibiblity of a low violation rate.

- Borda has an AVR similar to our trained NN. Borda is a widely studied rule and that it performs so well is not entirely surprising. Similarly, the Method of Equal Shares was designed specifically with proportionality axioms in mind [176]. Quite interestingly, EPH also exhibits very low axiom violations. Contrary to the other rules, EPH was designed to solve a real-world problem (avoiding strategic manipulation in voting for the Hugo Awards) rather than very specific axiomatic criteria. The performance of EPH and Borda suggests that well thought-out, intuitive rules can exhibit very strong performance.

- Strong Pareto Efficiency and Local Stability are also typically among the most highly violated axioms. These axioms are respectively focused on individual excellence and proportionality. There is no obvious link between violation rates of these axioms and whether a rule is more focused on individual excellence or proportionality. i.e. SNTV and STV frequently violate Pareto efficiency despite being oriented towards individual excellence, while proportional rules such as MES and variants of CC violate Local Stability more than most axioms.

5.6.3 Measuring Rule Differences

We have now seen ample evidence that several rules, particularly our trained NN rule, violate axioms at much lower rates than some other rules. The important follow-up question then becomes – Is our machine learning approach replicating some existing rule or is it finding non-violating committees that are distinct from any existing rule we explore? Given the strong performance of our ML approach, it is useful for designers of new voting rules to know if novel deterministic rules can identify new solutions or if any rule that avoids axiom violations will be similar to an existing rule. We can also ask this question of existing rules: two known rules may follow different procedures and end up with nearly identical results. Knowing this can be useful, e.g. if one of the two rules has much higher computational complexity.

In Table 5.9 we show the difference between all multi-winner rules, as defined in Section 5.3. The table shows results averaged over all number of winners and all preference distributions for $m = 7$ alternatives. Note that the difference values have been normalized so that a value of 0 indicates two rules always elect the same committee and a value of

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.723	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.149	.723	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.487	.723	.466	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.336	.723	.317	.301	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.235	.723	.251	.418	.362	–	–	–	–	–	–	–	–	–	–	–
PAV	.260	.722	.258	.430	.367	.065	–	–	–	–	–	–	–	–	–	–
CC	.572	.723	.602	.589	.574	.487	.464	–	–	–	–	–	–	–	–	–
lex-CC	.318	.722	.313	.452	.398	.144	.089	.440	–	–	–	–	–	–	–	–
seq-CC	.595	.723	.570	.464	.569	.493	.468	.673	.462	–	–	–	–	–	–	–
Monroe	.487	.723	.516	.533	.494	.400	.376	.117	.366	.619	–	–	–	–	–	–
Greedy M.	.351	.723	.336	.428	.406	.233	.204	.513	.219	.402	.432	–	–	–	–	–
MAV	.571	.723	.620	.703	.625	.607	.606	.350	.595	.821	.352	.644	–	–	–	–
MES	.285	.723	.273	.412	.376	.131	.087	.497	.118	.431	.410	.170	.620	–	–	–
EPH	.242	.723	.250	.422	.363	.021	.051	.483	.133	.485	.396	.225	.607	.119	–	–
RSD	.490	.723	.492	.592	.532	.469	.473	.652	.487	.635	.583	.490	.635	.475	.470	–
Min	.051	.723	.159	.479	.326	.256	.280	.568	.334	.601	.484	.364	.569	.302	.263	.496
Max	.967	.723	.950	.875	.922	.951	.940	.823	.923	.758	.876	.914	.842	.932	.946	.866

Table 5.9: Difference between rules for 7 alternatives with $1 \leq k < 7$ averaged over all preference distributions.

1 means the two rules always elect committees with minimal overlap. This table reveals interesting answers to both of the questions we posed above.

First, is our trained network replicating any existing rule? In the first column of the table we see that the NN rule is closest to the Min Violations committee, and second closest Borda's rule. This result is not altogether surprising, as previous work by Burka et al. has also suggested neural networks are prone to approximating Borda's rule [46]. Actually, the difference between the NN rule and each existing rule takes a similar relative order to each rule's axiom violation rate in Table 5.8. Borda, Bloc, EPH, and PAV are, in that order, the existing rules most similar to the NN and also have the lowest average violation rate over all axioms. Contrary to seeing that different rules find committees which are very distinct yet similarly desirable from an axiomatic perspective this suggests that many rules are all electing similar committees using distinct algorithms. Ultimately, these results provide no evidence to support a hypothesis that there exist multiple distinct committees which all provide similarly good axiomatic properties. If such a space exists, it is not what is learned by our model.

The second point of interest in Table 5.9 is in the similarity (or lack thereof) between existing rules. We consider first our baseline "rules." Between all of our baselines we find no surprises: rules are all closer to the minimally violating committee than the maximally violating committee, rules are all quite far from random committees, and Random Serial

Dictatorship is relatively far from most rules but closer than a complete random committee.

In looking at pairs of existing rules, there are two interesting trends. Some rules are extremely similar while others are surprisingly distinct. Each pair from Bloc, PAV, and EPH has a very low distance from each other. Upon examination, this is unsurprising. Each of these rules has voters spread some number of "points" between all alternatives of which they approve. When all voters approve exactly k alternatives (as in our experiments), EPH and Bloc become extremely similar rules algorithmically.

Much more interesting are the rules which have a very high difference from each other. Recall that Minimax Approval Voting (MAV) did not neatly fit into any category of voting rule: individual excellence, diversity, or proportionality. This categorical distinction is reflected in the difference between MAV and other rules. In nearly all cases, MAV selects a committee which is quite distinct from other rules. Most notably, MAV and seq-CC have a difference which is *greater* than the difference between MAV and random committees – the two rules follow a procedure which happens to select for minimally overlapping committees. It is also quite interesting that variations of rules can, despite sounding similar, result in very different outcomes. This can be seen with both {seq-CC, lex-CC, CC} and {Monroe and Greedy Monroe}.

5.7 Discussion

In this chapter, we have applied machine learning to teach us about voting rules. Most existing methods of analysis in social choice rely on strict theoretical proofs and exact categorizations. Here we have undertaken an experimental approach to both of these methods and identified insights unavailable to a theoretical approach.

In our first experiments in Section 5.5 we have developed an understanding of how to use machine learning to approximate existing voting rules. By training with the type of preference information known to be necessary to compute the output of some existing voting rule we can more accurately train a model to replicate that rule – i.e. using $\mathcal{M}^{\text{majority}}$ features is useful for computing Fishburn's C1 rules, such as Copeland. However, we have observed that, in many cases, models can train with features insufficient to fully compute the output of a rule and still learn to approximate a rule with reasonably high accuracy – i.e. training with $\mathcal{M}^{\text{majority}}$ features allows learning many C2 rules with high accuracy.

This observation, while somewhat dependent on the specific learning process, shows that the hard boundaries between different categorizations of rule do not fully reflect the reality that many rules can *nearly* be computed using information required for different categories. Our approach gives both a deeper understanding of how different rules use certain types of information (such as pairwise comparisons, or ordinal rankings), and can suggest answers to open questions (such as whether a rule like Instant Runoff Voting can be computed using only $\mathcal{M}^{\text{majority}}$ or $\mathcal{M}^{\text{weighted}}$ features).

Subsequently, we have taken a data-driven approach to study existing and novel voting rules using an axiom violation metric we have developed. By measuring the degree to which an axiom is violated by a rule on a particular preference distribution we are able to compare rules efficiently and at a scale infeasible to a theoretical approach. This approach holds great promise for simultaneously teaching us about (1) the rules being evaluated, (2) the axioms being evaluated, and (3) the preference distribution(s) in use.

In our experiments we have shown that for a set of desirable axioms describing a wide range of properties (a variety of priorities around individual excellence, diversity, and proportionality) most common multi-winner rules violate the axioms at a rate much greater than theoretically necessary. We also see that we can, in a straightforward manner, train models which act as novel voting rules that violate the axioms at a far lower rate than most existing rules.

We have evaluated rules on specific preference distributions, which reveals the very dramatic differences in axiom violation rates between distributions. When voters have very little structure to their preferences, most axioms have very little room for violations. However, as structure is added and groups of voters with similar preferences appear there is much more room for axiomatic violation.

By defining a distance metric to compare the winners elected by rules we are able to identify the many rules which ultimately elect very similar winners despite following different algorithms. This also shows us that our trained models are not behaving fundamentally different than the existing rules that have lower axiom violation rates – they are similar, just somewhat better.

Our methodological approach of data-driven axiomatic analysis opens several unexplored avenues of future research. We have seen here that most axioms can be easily categorized based on the axiom's prescriptive goals, typically electing individually good alternatives or proportional committees which represent all voters/groups of voters equally well. Many existing rules also tend to have lower axiom violation rate on one or the other of these ideals. There is room to develop both new axioms and new voting rules which provide a more intentional balance between these two goals or which select for alternative measures of quality (such as MAV, which aims to maximize the satisfaction of the least worst-off voter).

While we take a highly experimental approach,Table 5.8 highlights a number of potentially incomplete theoretical results. Geen cells in the table show that a rule has been proven to satisfy the corresponding axiom. However, there are many cells where a rule does not violate an axiom on any of our test instances. Proving (or disproving) that the lack of violations we have seen corresponds to a theoretical guarantee would usefully add to the body of knowledge about each rule. In some cases (i.e. the Unanimity axiom) this may be trivial; however, proving other results we have seen may be quite involved.

Finally, we suggest that future work may develop novel voting rules using our data-driven approach which provide a middle ground between human interpretability and desirable axiomatic properties in the domain of positional scoring rules. This chapter has shown that (1) the class of positional scoring rules contains highly diverse rules, and (2) some positional scoring rules – k-Borda, in particular – already often avoid violating axioms. On novel sets of axioms, or to out-perform k-Borda, we believe it is possible to optimize a positional scoring vector which minimizes axiom violations.

5.8 Conclusions

This chapter develops a novel framework for experimentally measuring the axiomatic properties of existing voting rules, and uses it to guide the development of new learned rules. This framework has revealed both that many existing voting rules violate axioms at a rate far greater than the experimentally demonstrated minimal violation rate, and that a new learned rule is able avoid mutual violations of axioms with very high frequency. We then defined a metric of the difference between voting rules to understand whether our learned rules are determining winners using a fundamentally unique behaviour or if they are selecting winners that are similar to those of other rules with low axiom violation rates. The fact that our learned rules are similar to the existing positional scoring rule k-Borda suggests that future work may benefit from optimizing a positional scoring vector to result in an interpretable rule with excellent axiomatic properties.

Chapter 6

Conclusions

This dissertation has shown the effectiveness of experimental analysis. We demonstrate, across several domains and methods of analysis, that using experimental techniques provides novel understanding of social choice procedures and improves performance of machine learning systems. In this chapter we will briefly review how our contributions relate to the thesis statement made in Chapter 1 and will outline several directions for future work.

6.1 Thesis Contributions

Our original thesis statement was,

> *Experimental analysis using tools from the fields of social choice and machine learning reveals novel understanding of these fields which are distinct from and complementary to the findings of theoretical approaches.*

Through this thesis we have demonstrated many times the ability of experimental analysis to teach us about both social choice and machine learning across varying decision-making domains (epistemic and preference-based), levels of application (abstract models, and classifiers trained on real-world data), and methods of analysis.

We have studied two common paradigms of social choice. In the epistemic setting, where voters have a shared goal of identifying a universally correct outcome (Chapter 3, Chapter 4), experimental comparison of elections held using liquid democracy across a wide range of voter abilities, social networks, and delegation strategies we identified two novel features which improve group accuracy: In a general model the inclusion of viscosity, a weight decay factor applied during delegation, is consistently effective at improving accuracy. In our general model, as well as with classifiers trained on real data, a delegation mechanism which delegates based on the weight of potential representatives is also highly effective at improving performance.

Our experiments further allowed us to identify evidence supporting the hypothesis that the reason for these improvements to group accuracy is that both features significantly reduce the potential for a single voter to capture large swaths of the total voting power, thus increasing the number of relevant decision-makers at any given time.

In the preference-based paradigm where voters have individual preferences and we evaluate based on the properties provided by a voting rule in Chapter 5 we have developed a robust framework for using election data to analyze questions of axiomatic satisfaction which have traditionally been stuck in the realm of theoretical research. Rather than answering questions of binary axiom satisfaction, our framework teaches us about existing rules by allowing us to easily measure the degree of axiom satisfaction provided by rules. We can trivially apply our framework to any voting rule or preference domain – both restrictions on the general setting which are quite imposing to theoretical analysis.

The synergy between machine learning and social choice has been long established for theoretical settings such as maximum likelihood estimation [62]. We have demonstrated that this extends to experimental settings as well. As demand for social choice frameworks with requirements tailored to increasingly specific settings grows [82] we have shown that classical machine learning techniques are able both to act as voting rules themselves – optimized for unique axiomatic requirements – but also to demonstrate bounds on the possible performance of other voting rules (Chapter 5).

We have also highlighted that direct application of social choice frameworks (liquid democracy) to the machine learning task of ensemble training reveals the possibility of identifying new approaches for common machine learning tasks. The delegative pruning technique we introduce in Chapter 4 operates in a way that may work in concert with existing ensemble techniques.

6.2 Directions for Future Work

Our work opens many directions for future work through extending our current results or applying similar techniques to novel domains. We suggest some interesting directions for each of the topics that we have discussed throughout this document.

6.2.1 Liquid Democracy

We have evaluated the effect of many delegation mechanisms on group accuracy. Our results show that the simple Proportional Weighted mechanism outperforms the others we have tested, but we have not proven the mechanism is optimal. Can we learn an optimized mechanism for delegation and the optimal amount of viscosity? Due to the relatively few assumptions in our standard model of liquid democracy, theoretical analysis is quite complex. However, identifying useful restrictions to our setting may allow for establishing bounds on the best/worst-case performance of any delegation mechanism, which would aid in evaluating any new mechanisms.

Our assumption of a singular correct objective allows us to make assumptions about voters loosely following a particular delegation mechanism if it performs well. Often voters may disagree on what the best outcome is; for example, if a voter has accuracy below half they are more likely to vote for the incorrect outcome. One way of interpreting that is to view the voter as preferring that outcome. If voters are of multiple types (i.e. with one optimal outcome per type), do different delegation mechanisms become more beneficial?

6.2.2 Ensemble Learning

Our method of ensemble pruning using liquid democracy operates at the level of classifiers; it does not take into account the values of the specific underlying data. Many ensemble methods are quite the opposite: they consider each classifier's performance on each underlying data sample. In principle, these methods are not mutually exclusive. A very valuable direction for future work is to develop an algorithm which incorporates the benefits of both type of algorithm: a process that benefits from the re-weighting of classifiers via liquid democracy and from the knowledge of classifier accuracy brought about from considering individual data.

A possible domain for this would be the continuous learning task explored in our related work by Blair et al. [30]. In that study we demonstrated the potential for ensembles using liquid democracy to dynamically re-weight classifiers to address the catastrophic forgetting problem inherent to continual learning. The algorithms we compared against primarily considered data-level information that could be incorporated into the delegation procedure we use for continual learning.

6.2.3 Data-driven Analysis of Voting Rules

We have established a robust framework which allows straightforward analysis of arbitrary voting rules under many conditions. A particularly useful aspect of our framework is that it works with data from any preference distribution and a wide class of axioms. This allows asking highly specific questions about the relative compatibility of a particular set of axioms on one preference distribution.

These types of questions lead to two potential benefits. First, theoretical questions can easily guide their analyses with an experimental analysis to identify potential impossibility results. More practically, understanding the behaviour of voting rules in relation to specific sets of axioms will guide the selection of voting rules for settings which prioritize certain properties, and expect certain types of voters. For example, when having "voters" rank output from language models an axiom describing clone-proofness may be important as language models might easily provide multiple responses which are functionally equivalent [63].

In these cases, we are also able to use our framework for generating data used to train novel rules with desirable axiomatic properties. While these learned rules are useful as proof of concept, they are less predictable and have no guarantee of avoiding bad outcomes. As such, learned rules should not be used in safety-critical settings such as political elections but may be useful when restricted to multi-agent domains (e.g. ensemble learning, multi-agent reinforcement learning). Agents do not complain about uninterpretable rules and may not suffer from an occasional bad outcome. On the other hand, in more safety-critical settings, learned rules may guide the discovery of similar, non-learned rules. For example, a learned rule which exhibits high similarity to both the Borda and Plurality rules may be an indication that a positional scoring rule between Borda and Plurality will provide similar results.

References

[1] 2024. URL: https://en.wikipedia.org/wiki/Instant-runoff_voting#Comparison_to_other_voting_systems.

[2] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.

[3] Ben Abramowitz and Nicholas Mattei. "Flexible representative democracy: an introduction with binary issues". In: *arXiv preprint arXiv:1811.02921* (2018).

[4] Stéphane Airiau, Umberto Grandi, and Filipo Studzinski Perotto. "Learning agents for iterative voting". In: *International Conference on Algorithmic Decision Theory*. Springer. 2017, pp. 139–152.

[5] Réka Albert and Albert-László Barabási. "Statistical mechanics of complex networks". In: *Reviews of modern physics* 74.1 (2002), p. 47.

[6] Shiri Alouf-Heffetz et al. "How Should We Vote? A Comparison of Voting Systems within Social Networks". In: *Thirty-First International Joint Conference on Artificial Intelligence (IJCAI 2022)*. 2022, pp. 1734–1740.

[7] Cem Anil and Xuchan Bao. "Learning to Elect". In: *Advances in Neural Information Processing Systems* 34 (2021).

[8] Elliot Anshelevich et al. "Approximating optimal social choice under metric preferences". In: *Artificial Intelligence* 264 (2018), pp. 27–51.

[9] Ben Armstrong, Shiri Alouf-Heffetz, and Nimrod Talmon. "Optimizing Viscous Democracy". In: *Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI 2024)* (2024).

[10] Ben Armstrong and Kate Larson. "Liquid Democracy for Low-Cost Ensemble Pruning". In: *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*. 2024, pp. 2135–2137.

[11] Ben Armstrong and Kate Larson. "Machine learning to strengthen democracy". In: *NeurIPS Joint Workshop on AI for Social Good*. 2019.

[12] Ben Armstrong and Kate Larson. "On the Limited Applicability of Liquid Democracy". In: *3rd Games, Agents, and Incentives Workshop at the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)* (2021).

[13] Kenneth J Arrow. "A difficulty in the concept of social welfare". In: *Journal of political economy* 58.4 (1950), pp. 328–346.

[14] Kenneth J Arrow. *Social choice and individual values*. Vol. 12. Yale university press, 2012.

[15] Muesluem Atas et al. "Liquid Democracy in Group-based Configuration." In: *ConfWS*. 2018, pp. 93–98.

[16] Emmanuelle Auriol and Robert J Gary-Bobo. "On the optimal number of representatives". In: *Public Choice* 153.3 (2012), pp. 419–445.

[17] Hossein Azari Soufiani, David C Parkes, and Lirong Xia. "A statistical decision-theoretic framework for social choice". In: *Advances in Neural Information Processing Systems* 27 (2014).

[18] Haris Aziz et al. "Justified Representation in Approval-Based Committee Voting". In: *Social Choice and Welfare* 48.2 (2017), pp. 461–485.

[19] Haris Aziz et al. "The Condorcet Principle for Multiwinner Elections: From Shortlisting to Proportionality". In: *arXiv preprint arXiv:1701.08023* (2017).

[20] Jimmy Ba and Brendan Frey. "Adaptive dropout for training deep neural networks". In: *Advances in neural information processing systems* 26 (2013).

[21] Philip Bachman, Ouais Alsharif, and Doina Precup. "Learning with pseudo-ensembles". In: *Advances in neural information processing systems* 27 (2014).

[22] Robert E Banfield et al. "Ensemble Diversity Measures and Their Application to Thinning". In: *Information Fusion* 6.1 (2005), pp. 49–62.

[23] Albert-László Barabási and Eric Bonabeau. "Scale-free networks". In: *Scientific american* 288.5 (2003), pp. 50–9.

[24] John Bartholdi, Craig A Tovey, and Michael A Trick. "Voting schemes for which it can be difficult to tell who won the election". In: *Social Choice and welfare* 6 (1989), pp. 157–165.

[25] Ruben Becker et al. "When Can Liquid Democracy Unveil the Truth?" In: *arXiv preprint arXiv:2104.01828* (2021).

[26] Jan Behrens et al. *The principles of LiquidFeedback*. Interaktive Demokratie e. V. Berlin, 2014.

[27] Rachel K. E. Bellamy et al. *AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias*. Oct. 2018. URL: https://arxiv.org/abs/1810.01943.

[28] Gerdus Benade, Paul Gölz, and Ariel D Procaccia. "No stratification without representation". In: *Proceedings of the 2019 ACM Conference on Economics and Computation*. 2019, pp. 281–314.

[29] Duncan Black et al. "The theory of committees and elections". In: (1958).

[30] Carter Blair, Ben Armstrong, and Kate Larson. "Liquid Ensemble Selection for Continual Learning". In: *Proceedings of the 37th Canadian Conference on Artificial Intelligence* (2024).

[31] Daan Bloembergen, Davide Grossi, and Martin Lackner. "On rational delegations in liquid democracy". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 2019, pp. 1796–1803.

[32] Christian Blum and Christina Isabel Zuber. "Liquid democracy: Potentials, problems, and perspectives". In: *Journal of Political Philosophy* 24.2 (2016), pp. 162–182.

[33] Niclas Boehmer, Piotr Faliszewski, and Sonja Kraiczy. "Properties of the Mallows Model Depending on the Number of Alternatives: A Warning for an Experimentalist". In: *International Conference on Machine Learning*. PMLR. 2023, pp. 2689–2711.

[34] Niclas Boehmer et al. "Guide to Numerical Experiments on Elections in Computational Social Choice". In: *arXiv preprint arXiv:2402.11765* (2024).

[35] Niclas Boehmer et al. "Putting a compass on the map of elections". In: *arXiv preprint arXiv:2105.07815* (2021).

[36] Paolo Boldi et al. "Liquid FM: recommending music through viscous democracy". In: *arXiv preprint arXiv:1503.08604* (2015).

[37] Paolo Boldi et al. "Viscous democracy for social networks". In: *Communications of the ACM* 54.6 (2011), pp. 129–137.

[38] Jean-Charles de Borda. "Mémoire sur les élections au scrutin". In: *Histoire de l'Academie Royale des Sciences pour 1781 (Paris, 1784)* (1784).

[39] Felix Brandt et al. *Handbook of computational social choice*. Cambridge University Press, 2016.

[40] Leo Breiman. "Bagging predictors". In: *Machine learning* 24 (1996), pp. 123–140.

[41] Markus Brill. "Interactive democracy". In: *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. 2018, pp. 1183–1187.

[42] Markus Brill and Nimrod Talmon. "Pairwise Liquid Democracy." In: *IJCAI*. Vol. 18. 2018, pp. 137–143.

[43] Markus Brill et al. "Liquid Democracy with Ranked Delegations". In: *arXiv preprint arXiv:2112.07509* (2021).

[44] Gavin Brown et al. "Diversity creation methods: a survey and categorisation". In: *Information Fusion* 6.1 (2005), pp. 5–20.

[45] Peter Bühlmann. "Bagging, boosting and ensemble methods". In: *Handbook of computational statistics: Concepts and methods* (2012), pp. 985–1022.

[46] David Burka et al. *Neural networks would'vote'according to Borda's Rule*. Tech. rep. KIT Working Paper Series in Economics, 2016.

[47] Dávid Burka et al. "Voting: a machine learning approach". In: *European Journal of Operational Research* (2021).

[48] Dávid Burka et al. "Voting: A Machine Learning Approach". In: *European Journal of Operational Research* 299.3 (2022), pp. 1003–1017.

[49] John Burkardt. "The truncated normal distribution". In: *Department of Scientific Computing Website, Florida State University* (2014), pp. 1–35.

[50] Andrea Campagner, Davide Ciucci, and Federico Cabitza. "Aggregation Models in Ensemble Learning: A Large-Scale Comparison". In: *Information Fusion* 90 (2023), pp. 241–252.

[51] Ioannis Caragiannis and Karl Fehrs. "The Complexity of Learning Approval-Based Multiwinner Voting Rules". In: *arXiv preprint arXiv:2110.00254* (2021).

[52] Ioannis Caragiannis and Evi Micha. "A Contribution to the Critique of Liquid Democracy." In: *IJCAI*. 2019, pp. 116–122.

[53] George DC Cavalcanti et al. "Combining diversity measures for ensemble pruning". In: *Pattern Recognition Letters* 74 (2016), pp. 38–45.

[54] Arjun Chandra and Xin Yao. "Evolving Hybrid Ensembles of Learning Machines for Better Generalisation". In: *Neurocomputing* 69.7-9 (2006), pp. 686–700.

[55] Aida Chefrour. "Incremental supervised learning: algorithms and applications in pattern recognition". In: *Evolutionary Intelligence* 12 (2019), pp. 97–112.

[56] François Chollet et al. *Keras*. https://keras.io. 2015.

[57] Zoé Christoff and Davide Grossi. "Binary voting with delegable proxy: An analysis of liquid democracy". In: *arXiv preprint arXiv:1707.08741* (2017).

[58] Gal Cohensius et al. "Proxy Voting for Better Outcomes". In: *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*. 2017, pp. 858–866.

[59] Rachael Colley and Umberto Grandi. "Preserving consistency in multi-issue liquid democracy". In: *31st International Joint Conference on Artificial Intelligence (IJCAI 2022)*. 2022, pp. 201–207.

[60] Rachael Colley, Umberto Grandi, and Arianna Novaro. "Smart Voting". In: *Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI 2020)*. International Joint Conferences on Artifical Intelligence (IJCAI). 2021, pp. 1734–1740.

[61] Vincent Conitzer. "Eliciting single-peaked preferences using comparison queries". In: *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*. 2007, pp. 1–8.

[62] Vincent Conitzer and Tuomas Sandholm. "Common Voting rules as Maximum Likelihood Estimators". In: *arXiv preprint arXiv:1207.1368* (2012).

[63] Vincent Conitzer et al. "Social Choice for AI Alignment: Dealing with Diverse Human Feedback". In: *arXiv preprint arXiv:2404.10271* (2024).

[64] Cristina Cornelio et al. "Voting with Random Classifiers (VORACE)". In: *arXiv preprint arXiv:1909.08996* (2019).

[65] Corinna Cortes. "Support-Vector Networks". In: *Machine Learning* (1995).

[66] Bernard Debord. "Prudent K-Choice Functions: Properties and Algorithms". In: *Mathematical Social Sciences* 26.1 (1993), pp. 63–77.

[67] Greg d'Eon and Kate Larson. "Testing Axioms Against Human Reward Divisions in Cooperative Games". In: *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*. 2020, pp. 312–320.

[68] Franz Dietrich and Kai Spiekermann. "Jury theorems". In: *The Routledge Handbook of Social Epistemology*. Routledge, 2019, pp. 386–396.

[69] Tom Dietterich. "Overfitting and undercomputing in machine learning". In: *ACM computing surveys (CSUR)* 27.3 (1995), pp. 326–327.

[70] Charkes L. Dodgson. *The principles of parliamentary representation*. Harrison and Sons, 1884.

[71] Dheeru Dua and Casey Graff. *UCI Machine Learning Repository*. 2017. URL: http://archive.ics.uci.edu/ml.

[72] Richard O Duda, Peter E Hart, and David G Stork. *Pattern classification and scene analysis*. Vol. 3.

[73] M Dummett. *Voting Procedures*. Oxford University Press, 1984.

[74] Florian Eggenberger and George Pólya. "Über die statistik verketteter vorgänge". In: *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* 3.4 (1923), pp. 279–289.

[75] Edith Elkind. "Restricted preference domains in social choice: Two perspectives". In: *Algorithmic Game Theory: 11th International Symposium, SAGT 2018, Beijing, China, September 11-14, 2018, Proceedings 11*. Springer. 2018, pp. 12–18.

[76] Edith Elkind, Martin Lackner, and Dominik Peters. "Preference restrictions in computational social choice: A survey". In: *arXiv preprint arXiv:2205.09092* (2022).

[77] Edith Elkind et al. "Properties of Multiwinner Voting Rules". In: *Social Choice and Welfare* 48 (2017), pp. 599–632.

[78] Ulle Endriss. "Logic and Social Choice Theory". In: (2012).

[79] James M Enelow and Melvin J Hinich. *The spatial theory of voting: An introduction*. CUP Archive, 1984.

[80] Paul Erdos, Alfréd Rényi, et al. "On the evolution of random graphs". In: *Publ. Math. Inst. Hung. Acad. Sci* 5.1 (1960), pp. 17–60.

[81] Bruno Escoffier, Hugo Gilbert, and Adèle Pass-Lanneau. "The Convergence of Iterative Delegations in Liquid Democracy". In: *CoRR abs/1809.04362* (2018).

[82] Piotr Faliszewski. "Social Choice 2.0 and Customized Multiwinner Voting". In: *The Future of Economic Design: The Continuing Development of a Field as Envisioned by Its Researchers* (2019), pp. 75–81.

[83] Piotr Faliszewski et al. "An Experimental Comparison of Multiwinner Voting Rules on Approval Elections". In: *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*. 2023, pp. 2675–2683.

[84] Piotr Faliszewski et al. "Committee Scoring Rules: Axiomatic Classification and Hierarchy". In: *IJCAI*. 2016, pp. 250–256.

[85] Piotr Faliszewski et al. "Multiwinner Analogues of the Plurality Rule: Axiomatic and Algorithmic Perspectives". In: *Social Choice and Welfare* 51.3 (2018), pp. 513–550.

[86] Piotr Faliszewski et al. "Multiwinner Voting: A New Challenge for Social Choice Theory". In: *Trends in Computational Social Choice* 74.2017 (2017), pp. 27–47.

[87] Alexander Fidora and Carles Sierra. *Ramon Llull, from the Ars Magna to artificial intelligence*. Artificial Intelligence Research Institute Barcelona, 2011.

[88] Daniel Firebanks-Quevedo. "Machine Learning? In MY Election? It's More Likely Than You Think: Voting Rules via Neural Networks". In: (), p. 19.

[89] Peter C Fishburn. "Condorcet Social Choice Functions". In: *SIAM Journal on applied Mathematics* 33.3 (1977), pp. 469–489.

[90] Bailey Flanigan, Daniel Halpern, and Alexandros Psomas. "Smoothed Analysis of Social Choice Revisited". In: *International Conference on Web and Internet Economics*. Springer. 2023, pp. 290–309.

[91] Bailey Flanigan et al. "Fair algorithms for selecting citizens' assemblies". In: *Nature* 596.7873 (2021), pp. 548–552.

[92] Brian Ford. *Delegative Democracy. May 2002*.

[93] Rupert Freeman, Markus Brill, and Vincent Conitzer. "On the axiomatic characterization of runoff voting rules". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 28. 1. 2014.

[94] Yoav Freund and Robert E Schapire. "A decision-theoretic generalization of on-line learning and an application to boosting". In: *European conference on computational learning theory*. Springer. 1995, pp. 23–37.

[95] Robin Fritsch, Marino Müller, and Roger Wattenhofer. "Analyzing Voting Power in Decentralized Governance: Who controls DAOs?" In: *arXiv preprint arXiv:2204.01176* (2022).

[96] Kun Gao et al. "Learning First-Order Rules With Differentiable Logic Program Semantics". In: *arXiv preprint arXiv:2204.13570* (2022).

[97] William V Gehrlein. "The Condorcet Criterion and Committee Selection". In: *Mathematical Social Sciences* 10.3 (1985), pp. 199–209.

[98] William V Gehrlein and Peter C Fishburn. "Condorcet's paradox and anonymous preference profiles". In: *Public Choice* (1976), pp. 1–18.

[99] Christian Geist and Ulrich Endriss. "Automated search for impossibility theorems in social choice theory: Ranking sets of objects". In: *Journal of Artificial Intelligence Research* 40 (2011), pp. 143–174.

[100] Christian Geist and Dominik Peters. "Computer-aided methods for social choice theory". In: *Trends in Computational Social Choice* (2017), pp. 249–267.

[101] Allan Gibbard. "Manipulation of voting schemes: a general result". In: *Econometrica: journal of the Econometric Society* (1973), pp. 587–601.

[102] Renan U Goetz, Yolanda Martinez, and Jofre Rodrigo. "Water allocation by social choice rules: the case of sequential rules". In: *Ecological Economics* 65.2 (2008), pp. 304–314.

[103] Judy Goldsmith et al. "Voting with rank dependent scoring rules". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 28. 1. 2014.

[104] Noah Golowich, Harikrishna Narasimhan, and David C Parkes. "Deep Learning for Multi-Facility Location Mechanism Design." In: *IJCAI*. 2018, pp. 261–267.

[105] Paul Gölz et al. "The fluid mechanics of liquid democracy". In: *International Conference on Web and Internet Economics*. Springer. 2018, pp. 188–202.

[106] Aidan N Gomez et al. "Targeted dropout". In: (2018).

[107] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[108] Naveen Sundar Govindarajulu, RealityEngines AI, and Colin White. "Differentiable Functions for Combining First-order Constraints with Deep Learning via Weighted Proof Tracing". In: ().

[109] James Green-Armytage. "Direct voting and proxy voting". In: *Constitutional Political Economy* 26.2 (2015), pp. 190–220.

[110] Bernard Grofman. "A comment on 'democratic theory: A preliminary mathematical model.'" In: *Public Choice* 21.1 (1975), pp. 99–103.

[111] Bernard Grofman. "Judgmental competence of individuals and groups in a dichotomous choice situation: Is a majority of heads better than one?" In: *Journal of Mathematical Sociology* 6.1 (1978), pp. 47–60.

[112] Bernard Grofman, Guillermo Owen, and Scott L Feld. "Average competence, variability in individual competence, and accuracy of statistically pooled group decisions". In: *Psychological Reports* 50.3 (1982), pp. 683–688.

[113] Bernard Grofman, Guillermo Owen, and Scott L Feld. "Thirteen theorems in search of the truth". In: *Theory and decision* 15.3 (1983), pp. 261–278.

[114] Georges Théodule Guilbaud. "Les théories de l'intérêt général et le problème logique de l'agrégation". In: *Economie appliquée* 5.4 (1952), pp. 501–584.

[115] Nika Haghtalab, Ritesh Noothigattu, and Ariel D Procaccia. "Weighted voting via no-regret learning". In: *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

[116] Daniel Halpern et al. "In Defense of Fluid Democracy". In: *arXiv preprint arXiv:2107.11868* (2021).

[117] Kazuyuki Hara, Daisuke Saitoh, and Hayaru Shouno. “Analysis of dropout learning regarded as ensemble learning”. In: *Artificial Neural Networks and Machine Learning–ICANN 2016: 25th International Conference on Artificial Neural Networks, Barcelona, Spain, September 6-9, 2016, Proceedings, Part II 25*. Springer. 2016, pp. 72–79.

[118] Steve Hardt and Lia CR Lopes. “Google votes: A liquid democracy experiment on a corporate social network”. In: (2015).

[119] James Wycliffe Headlam. *Election by lot at Athens*. 4. At the University Press, 1891.

[120] Dominique Henriet. “The Copeland choice function an axiomatic characterization”. In: *Social Choice and Welfare* 2.1 (1985), pp. 49–63.

[121] Wesley H Holliday and Eric Pacuit. “Split Cycle: a new Condorcet-consistent voting method independent of clones and immune to spoilers”. In: *Public Choice* 197.1 (2023), pp. 1–62.

[122] Yili Hong. “On computing the distribution function for the Poisson binomial distribution”. In: *Computational Statistics & Data Analysis* 59 (2013), pp. 41–51.

[123] J. D. Hunter. “Matplotlib: A 2D graphics environment”. In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.

[124] Ken-ichi Inada. “The Simple Majority Decision Rule”. In: *Econometrica: Journal of the Econometric Society* (1969), pp. 490–506.

[125] Svante Janson. *Phragmén’s and Thiele’s election methods*. Tech. rep. Technical report, 2016.

[126] Anson Kahng, Simon Mackenzie, and Ariel D Procaccia. “Liquid democracy: An algorithmic perspective”. In: *Thirty-Second AAAI Conference on Artificial Intelligence*. 2018.

[127] Drora Karotkin and Jacob Paroush. “Optimum committee size: Quality-versus-quantity dilemma”. In: *Social Choice and Welfare* 20.3 (2003), pp. 429–441.

[128] Raphael G Kazmann. “Democratic organization: A preliminary mathematical model”. In: *Public Choice* 16.1 (1973), pp. 17–26.

[129] Scott Kirkpatrick, C Daniel Gelatt Jr, and Mario P Vecchi. “Optimization by simulated annealing”. In: *science* 220.4598 (1983), pp. 671–680.

[130] Christoph Carl Kling et al. “Voting behaviour and power in online democracy: A study of LiquidFeedback in Germany’s Pirate Party”. In: *arXiv preprint arXiv:1503.07723* (2015).

[131] Grammateia Kotsialou and Luke Riley. "Incentivising Participation in Liquid Democracy with Breadth-First Delegation". In: *arXiv preprint arXiv:1811.03710* (2018).

[132] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet Classification with Deep Convolutional Neural Networks". In: *Advances in Neural Information Processing Systems* 25 (2012).

[133] Hanna Kujawska, Marija Slavkovik, and Jan-Joachim Rückmann. "Predicting the winners of Borda, Kemeny and Dodgson elections with supervised machine learning". In: *Multi-Agent Systems and Agreement Technologies*. Springer, 2020, pp. 440–458.

[134] Ludmila I Kuncheva and Christopher J Whitaker. "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy". In: *Machine learning* 51.2 (2003), pp. 181–207.

[135] Ludmila I Kuncheva et al. "Limits on the majority vote accuracy in classifier fusion". In: *Pattern Analysis & Applications* 6.1 (2003), pp. 22–31.

[136] Alex Labach, Hojjat Salehinejad, and Shahrokh Valaee. "Survey of dropout methods for deep neural networks". In: *arXiv preprint arXiv:1904.13310* (2019).

[137] Martin Lackner, Peter Regner, and Benjamin Krenn. "ABCvoting: A Python Package for Approval-Based Multi-Winner Voting Rules". In: *Journal of Open Source Software* 8.81 (2023), p. 4880. DOI: 10.21105/joss.04880.

[138] Martin Lackner and Piotr Skowron. *Multi-Winner Voting with Approval Preferences*. Springer International Publishing, 2023. ISBN: 9783031090165. DOI: 10.1007/978-3-031-09016-5. URL: http://dx.doi.org/10.1007/978-3-031-09016-5.

[139] Louisa Lam and SY Suen. "Application of majority voting to pattern recognition: an analysis of its behavior and performance". In: *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans* 27.5 (1997), pp. 553–568.

[140] Marc Lanctot et al. "Evaluating Agents using Social Choice Theory". In: *arXiv preprint arXiv:2312.03121* (2023).

[141] Pierre-Simon Laplace. *Essai philosophique sur les probabilités*. H. Remy, 1829.

[142] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *nature* 521.7553 (2015), pp. 436–444.

[143] Florin Leon, Sabina-Adriana Floria, and Costin Bădică. "Evaluating the Effect of Voting Methods on Ensemble-Based Classification". In: *2017 IEEE international conference on INnovations in intelligent Systems and applications (INISTA)*. IEEE. 2017, pp. 1–6.

[144] Yong Liu and Xin Yao. “Ensemble Learning Via Negative Correlation”. In: *Neural networks* 12.10 (1999), pp. 1399–1404.

[145] Bonifacio Llamazares and Teresa Peña. “Scoring rules and social choice properties: some characterizations”. In: *Theory and Decision* 78.3 (2015), pp. 429–450.

[146] Malik Magdon-Ismail and Lirong Xia. “A mathematical model for optimal decisions in a representative democracy”. In: *arXiv preprint arXiv:1807.06157* (2018).

[147] Colin L Mallows. “Non-null ranking models. I”. In: *Biometrika* 44.1/2 (1957), pp. 114–130.

[148] Dragos D Margineantu and Thomas G Dietterich. “Pruning Adaptive Boosting”. In: *ICML*. Vol. 97. Citeseer. 1997, pp. 211–218.

[149] Howard Margolis. “A note on incompetence”. In: *Public Choice* (1976), pp. 119–127.

[150] Evangelos Markakis and Georgios Papasotiropoulos. “An Approval-Based Model for Single-Step Liquid Democracy”. In: *International Symposium on Algorithmic Game Theory*. Springer. 2021, pp. 360–375.

[151] Minsky Marvin and A Papert Seymour. *Perceptrons*. 1969.

[152] Leonardo Matone et al. “DeepVoting: Learning Voting Rules with Tailored Embeddings”. In: *arXiv preprint arXiv:2408.13630* (2024).

[153] Kenneth O May. “A set of independent necessary and sufficient conditions for simple majority decision”. In: *Econometrica: Journal of the Econometric Society* (1952), pp. 680–684.

[154] Vincent Merlin. “The axiomatic characterizations of majority voting and scoring rules”. In: *Mathématiques et sciences humaines. Mathematics and social sciences* 163 (2003).

[155] James C Miller. “A program for direct and proxy voting in the legislative process”. In: *Public choice* 7.1 (1969), pp. 107–113.

[156] James A Mirrlees. “An Exploration in the Theory of Optimum Income Taxation”. In: *The review of economic studies* 38.2 (1971), pp. 175–208.

[157] Farhad Mohsin et al. “Learning to Design Fair and Private Voting Rules”. In: *Cooperative AI workshop at NeurIPS 2020* (2020).

[158] Christopher Z Mooney. *Monte carlo simulation*. Sage, 1997.

[159] Hervé Moulin and Richard Stong. “Filling a Multicolor Urn: An Axiomatic Analysis”. In: *Games and Economic Behavior* 45.1 (2003), pp. 242–269.

[160] Andreas Møller Mulvad and Benjamin Ask Popp-Madsen. "Sortition-infused democracy: Empowering citizens in the age of climate emergency". In: *Thesis Eleven* 167.1 (2021), pp. 77–98.

[161] Jean Antoine Nicolas et al. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. de l'Imprimerie Royale, 1785.

[162] Shmuel Nitzan and Jacob Paroush. "Collective decision making and jury theorems". In: *The Oxford Handbook of Law and Economics* 1 (2017).

[163] Shmuel Nitzan and Jacob Paroush. "Investment in human capital and social self protection under uncertainty". In: *International Economic Review* (1980), pp. 547–557.

[164] Shmuel Nitzan and Jacob Paroush. "Optimal decision rules in uncertain dichotomous choice situations". In: *International Economic Review* (1982), pp. 289–297.

[165] Shmuel Nitzan and Jacob Paroush. "The significance of independent decisions in uncertain dichotomous choice situations". In: *Theory and Decision* 17.1 (1984), pp. 47–60.

[166] Ritesh Noothigattu, Dominik Peters, and Ariel D Procaccia. "Axioms for Learning from Pairwise Comparisons". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 17745–17754.

[167] Guillermo Owen, Bernard Grofman, and Scott L Feld. "Proving a distribution-free generalization of the Condorcet jury theorem". In: *Mathematical Social Sciences* 17.1 (1989), pp. 1–16.

[168] David Parkes and Ariel Procaccia. "Dynamic social choice with evolving preferences". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 27. 1. 2013.

[169] Jacob Paroush. "Stay away from fair coins: A Condorcet jury theorem". In: *Social Choice and Welfare* (1998), pp. 15–20.

[170] Ioannis Partalas, Grigorios Tsoumakas, and Ioannis Vlahavas. "An Ensemble Uncertainty Aware Measure for Directed Hill Climbing Ensemble Pruning". In: *Machine Learning* 81 (2010), pp. 257–282.

[171] Adam Paszke et al. "Pytorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in neural information processing systems* 32 (2019).

[172] David Patterson et al. "Carbon emissions and large neural network training". In: *arXiv preprint arXiv:2104.10350* (2021).

[173] Alois Paulin. "An overview of ten years of liquid democracy research". In: *The 21st Annual International Conference on Digital Government Research*. 2020, pp. 116–121.

[174] Alois Paulin. "Through liquid democracy to sustainable non-bureaucratic government". In: *Proc. Int. Conf. for E-democracy and open government*. 2014, pp. 205–217.

[175] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[176] Dominik Peters and Piotr Skowron. "Proportionality and the limits of welfarism". In: *Proceedings of the 21st ACM Conference on Economics and Computation*. 2020, pp. 793–794.

[177] Jean-Benoit Pilet et al. "Public support for Citizens' Assemblies selected through sortition: Survey and experimental evidence from 15 countries". In: (2020).

[178] Robi Polikar. "Ensemble learning". In: *Ensemble machine learning: Methods and applications* (2012), pp. 1–34.

[179] Ariel D Procaccia. "Axioms Should Explain Solutions". In: *The Future of Economic Design: The Continuing Development of a Field as Envisioned by Its Researchers* (2019), pp. 195–199.

[180] Ariel D Procaccia, Sashank J Reddi, and Nisarg Shah. "A maximum likelihood approach for selecting sets of alternatives". In: *arXiv preprint arXiv:1210.4882* (2012).

[181] Ariel D Procaccia et al. "The learnability of voting rules". In: *Artificial Intelligence* 173.12-13 (2009), pp. 1133–1149.

[182] Chao Qian, Yang Yu, and Zhi-Hua Zhou. "Pareto Ensemble Pruning". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 29. 1. 2015.

[183] J Ross Quinlan. *C4.5: programs for machine learning*. Elsevier, 1993.

[184] J. Ross Quinlan. "Induction of decision trees". In: *Machine learning* 1 (1986), pp. 81–106.

[185] Jameson Quinn and Bruce Schneier. "A proportional voting system for awards nominations resistant to voting blocs". In: *Preprint per https://www. schneier. com/academic/archives/2016/05/a_proportional_votin. html. Accessed* 14 (2019).

[186] Michel Regenwetter. *Behavioral social choice: probabilistic models, statistical inference, and applications*. Cambridge University Press, 2006.

[187] Manon Revel, Tao Lin, and Daniel Halpern. "The Optimal Size of an Epistemic Congress". In: *arXiv preprint arXiv:2107.01042* (2021).

[188] Whitman Richards, H Sebastian Seung, and Galen Pickard. “Neural voting machines”. In: *Neural Networks* 19.8 (2006), pp. 1161–1167.

[189] Herbert Robbins and Sutton Monro. “A stochastic approximation method”. In: *The annals of mathematical statistics* (1951), pp. 400–407.

[190] Frank Rosenblatt. *The Perceptron, a Perceiving and Recognizing Automaton (Project Para)*. Cornell Aeronautical Laboratory, 1957.

[191] Francesca Rossi, Kristen Brent Venable, and Toby Walsh. *A Short Introduction to Preferences: Between AI and Social Choice*. Morgan & Claypool Publishers, 2011.

[192] Omer Sagi and Lior Rokach. “Ensemble Learning: A Survey”. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.4 (2018), e1249.

[193] Howard E Sattler. “Effect of group variability on pooled group decisions”. In: *Psychological Reports* 18.3 (1966), pp. 676–678.

[194] Robert E Schapire and Yoram Singer. “Improved Boosting Algorithms Using Confidence-rated Predictions”. In: *Proceedings of the eleventh annual conference on Computational learning theory*. 1998, pp. 80–91.

[195] Marie Christin Schmidtlein. “Voting by Axioms”. In: *Master’s thesis. ILLC, University of Amsterdam* (2022).

[196] Marie Christin Schmidtlein and Ulle Endriss. “Voting by Axioms”. In: *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*. 2023, pp. 2067–2075.

[197] Yohei Sekiguchi. “A characterization of the plurality rule”. In: *Economics Letters* 116.3 (2012), pp. 330–332.

[198] Amartya Sen. “Internal consistency of choice”. In: *Econometrica: Journal of the Econometric Society* (1993), pp. 495–521.

[199] Lloyd Shapley and Bernard Grofman. “Optimizing group judgmental accuracy in the presence of interdependencies”. In: *Public Choice* 43.3 (1984), pp. 329–343.

[200] Or Sharir, Barak Peleg, and Yoav Shoham. “The cost of training nlp models: A concise overview”. In: *arXiv preprint arXiv:2004.08900* (2020).

[201] Weiran Shen, Pingzhong Tang, and Song Zuo. “Automated mechanism design via neural networks”. In: *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems*. 2019, pp. 215–223.

[202] Nitish Srivastava et al. “Dropout: a simple way to prevent neural networks from overfitting”. In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[203] Ivan D Steiner and Nageswari Rajaratnam. “A model for the comparison of individual and group performance scores”. In: *Behavioral Science* 6.2 (1961), pp. 142–147.

[204] Peter Stone. “Sortition, voting, and democratic equality”. In: *Equality and Representation*. Routledge, 2019, pp. 47–64.

[205] Peter Stone. “The logic of random selection”. In: *Political theory* 37.3 (2009), pp. 375–397.

[206] Jane Suiter et al. “Measuring epistemic deliberation on polarized issues: The case of abortion provision in Ireland”. In: *Political Studies Review* 20.4 (2022), pp. 630–647.

[207] Warut Suksompong. “Tournaments in Computational Social Choice: Recent Developments.” In: *IJCAI*. 2021, pp. 4611–4618.

[208] Stanisław Szufa et al. “Drawing a map of elections in the space of statistical cultures”. In: *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems*. 2020, pp. 1341–1349.

[209] Rein Taagepera. “The size of national assemblies”. In: *Social science research* 1.4 (1972), pp. 385–401.

[210] Tibor Tajti. “New voting functions for neural network algorithms”. In: *Annales Mathematicae et Informaticae*. Vol. 52. Eszterházy Károly Egyetem Líceum Kiadó. 2020, pp. 229–242.

[211] Christino Tamon and Jie Xiang. “On the Boosting Pruning Problem”. In: *Machine Learning: ECML 2000: 11th European Conference on Machine Learning Barcelona, Catalonia, Spain, May 31–June 2, 2000*. Springer. 2000, pp. 404–412.

[212] Thorvald N Thiele. “Om flerfoldsvalg”. In: *Oversigt over det Kongelige Danske Videnskabernes Selskabs Forhandlinger* 1895 (1895), pp. 415–441.

[213] Nicolaus Tideman. “The Single Transferable Vote”. In: *Journal of Economic Perspectives* 9.1 (1995), pp. 27–38.

[214] Isaac Todhunter. *A history of the mathematical theory of probability*. Cambridge University Press, 1865.

[215] Leslie G Valiant. “A Theory of the Learnable”. In: *Communications of the ACM* 27.11 (1984), pp. 1134–1142.

[216] Gido M Van de Ven, Tinne Tuytelaars, and Andreas S Tolias. “Three types of incremental learning”. In: *Nature Machine Intelligence* 4.12 (2022), pp. 1185–1197.

[217] Yu A Veselova. “Computational complexity of manipulation: a survey”. In: *Automation and Remote Control* 77 (2016), pp. 369–388.

[218] Pauli Virtanen et al. “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python”. In: *Nature Methods* 17 (2020), pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

[219] John Von Neumann and Oskar Morgenstern. “Theory of games and economic behavior, 2nd rev”. In: (1947).

[220] Toby Walsh. “Generating single peaked votes”. In: *arXiv preprint arXiv:1503.02766* (2015).

[221] Toby Walsh and Lirong Xia. “Lot-based voting rules.” In: *AAMAS*. Vol. 12. Citeseer. 2012, pp. 603–610.

[222] Paul J Werbos. “Backpropagation Through Time: What It Does and How To Do It”. In: *Proceedings of the IEEE* 78.10 (1990), pp. 1550–1560.

[223] Nic Wilson. “Generating Voting Rules from Random Relations”. In: *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*. 2019, pp. 2267–2269.

[224] Sara Wolk, Jameson Quinn, and Marcus Ogren. “STAR Voting, equality of voice, and voter satisfaction: considerations for voting method reform”. In: *Constitutional Political Economy* 34.3 (2023), pp. 310–334.

[225] Lirong Xia. “Bayesian estimators as voting rules”. In: *Uncertainty in artificial intelligence*. 2018.

[226] Lirong Xia. “Designing social choice mechanisms using machine learning”. In: *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-Agent Systems*. 2013, pp. 471–474.

[227] Lirong Xia and Vincent Conitzer. “A maximum likelihood approach towards aggregating partial orders”. In: *Twenty-Second International Joint Conference on Artificial Intelligence*. 2011.

[228] Tong Xiao et al. “Learning deep feature representations with domain guided dropout for person re-identification”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 1249–1258.

[229] Hiroshi Yamakawa, Michiko Yoshida, and Motohiro Tsuchiya. “Toward delegated democracy: Vote by yourself, or trust your network”. In: *International Journal of Human and Social Sciences* 1.2 (2007), pp. 146–150.

[230] Fan Yang, Zhilin Yang, and William W Cohen. "Differentiable Learning of Logical Rules for Knowledge Base Reasoning". In: *Advances in neural information processing systems* 30 (2017).

[231] H Peyton Young. "An axiomatization of Borda's rule". In: *Journal of economic theory* 9.1 (1974), pp. 43–52.

[232] H Peyton Young. "Condorcet's theory of voting". In: *American Political science review* 82.4 (1988), pp. 1231–1244.

[233] H Peyton Young. "Social Choice Scoring Functions". In: *SIAM Journal on Applied Mathematics* 28.4 (1975), pp. 824–838.

[234] Peyton Young. "Optimal voting rules". In: *Journal of Economic Perspectives* 9.1 (1995), pp. 51–64.

[235] Yi Zhang et al. "Ensemble Pruning Via Semi-definite Programming." In: *Journal of machine learning research* 7.7 (2006).

[236] Yuzhe Zhang and Davide Grossi. "Power in Liquid Democracy". In: *arXiv preprint arXiv:2010.07070* (2020).

[237] Qiang-Li Zhao, Yan-Huang Jiang, and Ming Xu. "Incremental Learning Based on Ensemble Pruning". In: *Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*. Vol. 1. IEEE. 2011, pp. 377–381.

[238] Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. "Ensembling Neural Networks: Many Could be Better Than All". In: *Artificial intelligence* 137.1-2 (2002), pp. 239–263.

APPENDICES

Appendix A

Full Results of Delegative Ensembles

For readability, we highlight a limited number of experiment results in the main text. Here we provide all experiment results from Chapter 4 for each dataset.

A.1 breast-cancer-wisconsin

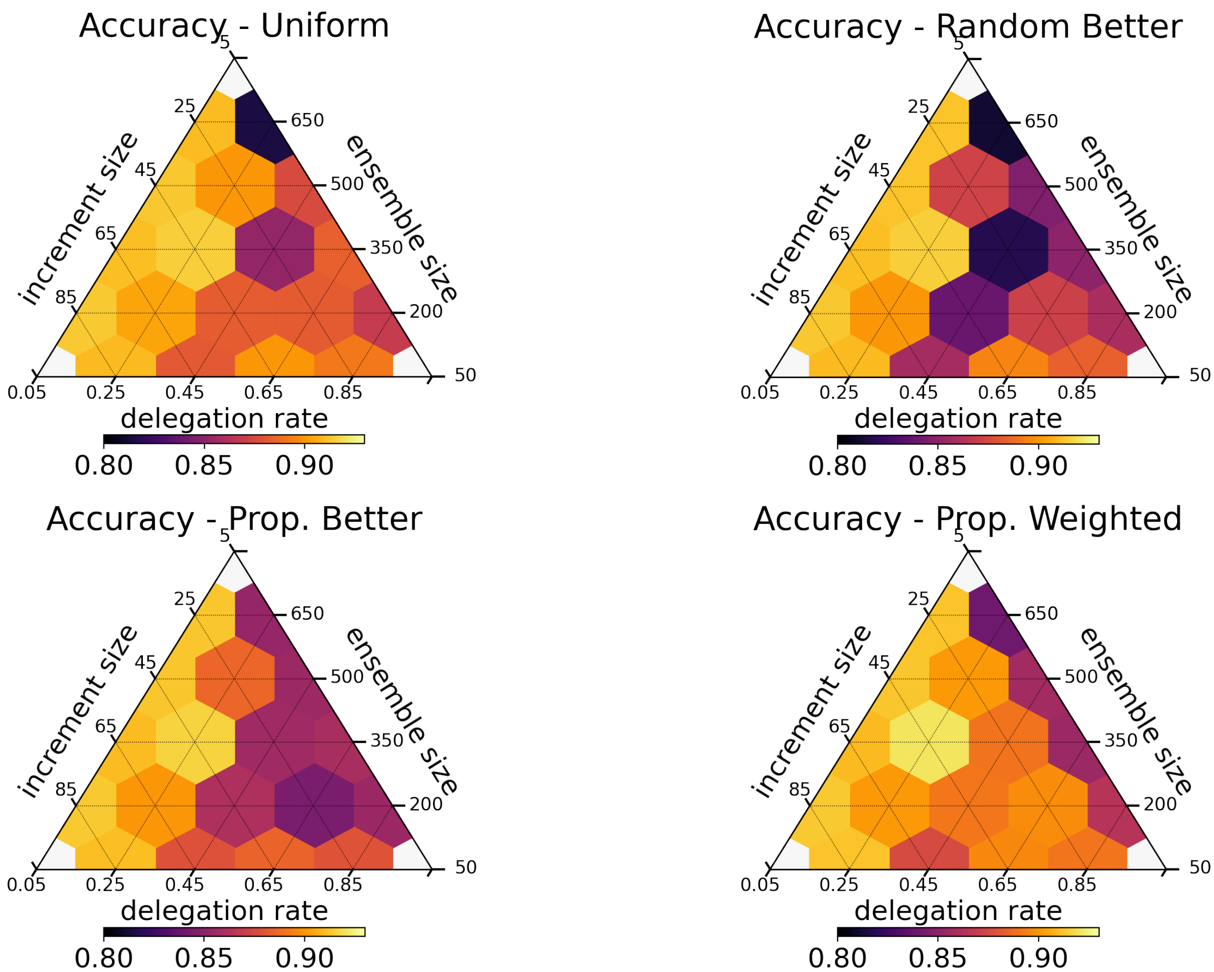


Figure A.1: Test accuracy of fully trained ensembles as parameters are varied. Results from breast-cancer-wisconsin dataset.

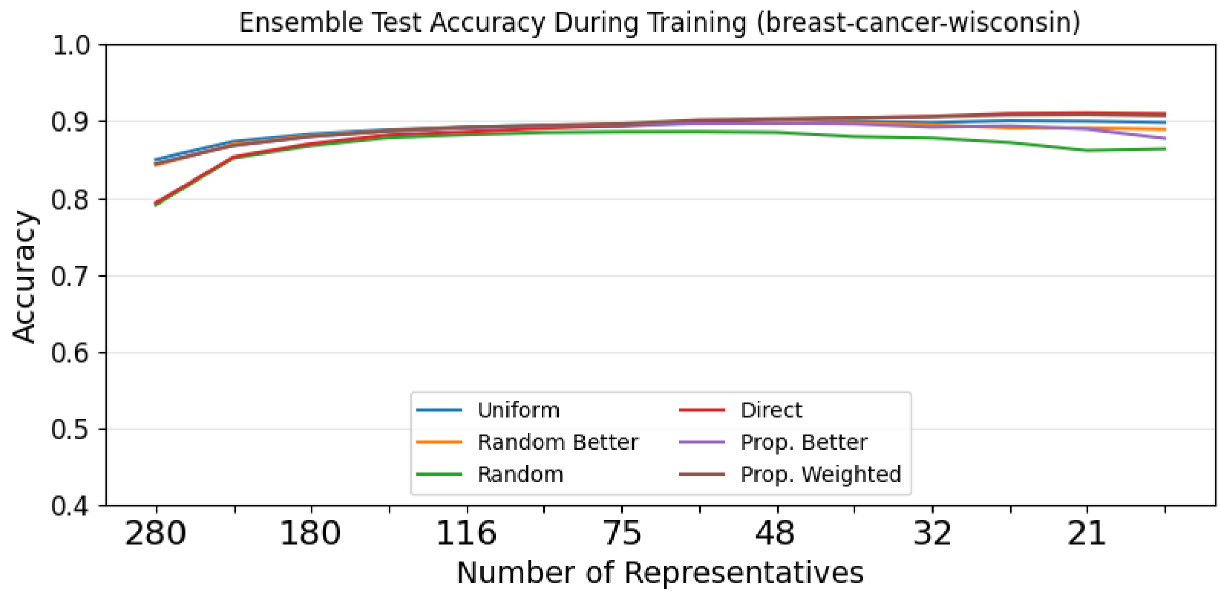


Figure A.2: Test accuracy during training on breast-cancer-wisconsin dataset, averaged over 500 trials.

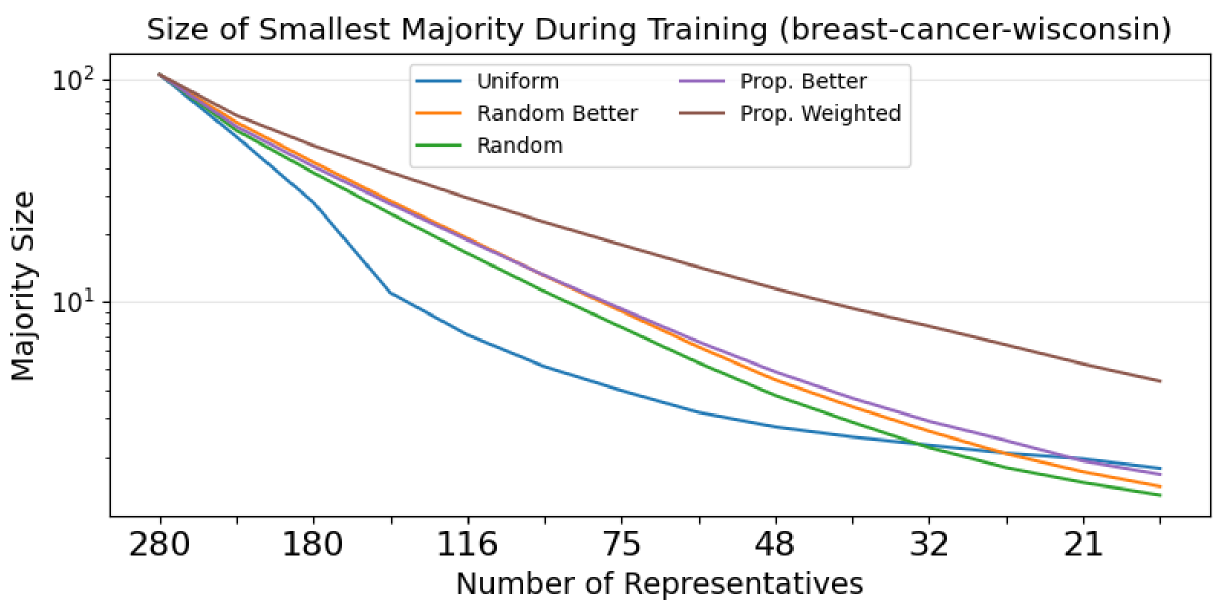


Figure A.3: Minimum majority size during training on the breast-cancer-wisconsin dataset.

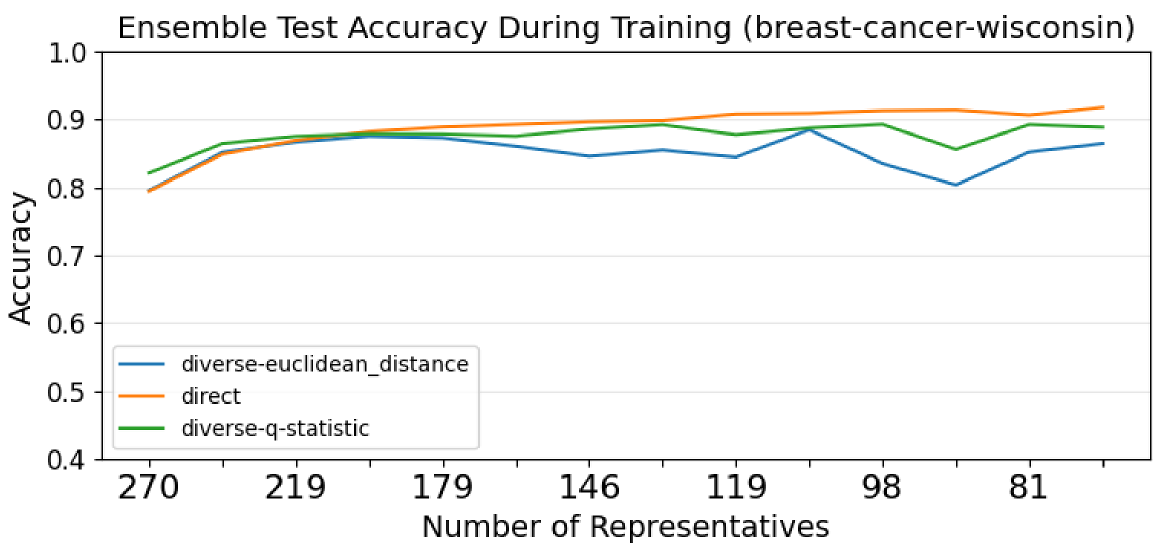


Figure A.4: Test accuracy during training on breast-cancer-wisconsin dataset, averaged over 30 trials, using diversity metrics to guide delegation.

A.2 credit-approval

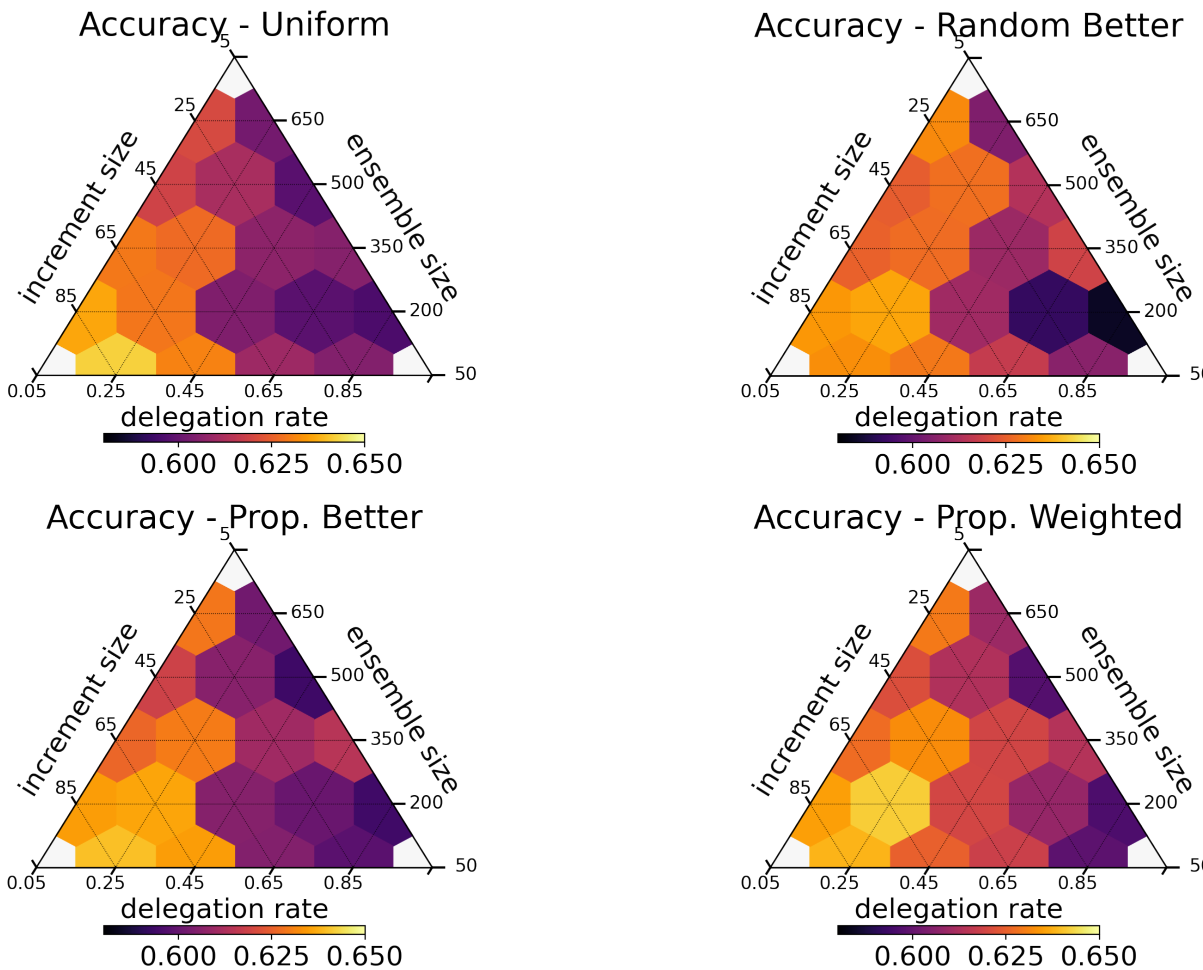


Figure A.5: Test accuracy of fully trained ensembles as parameters are varied. Results from credit-approval dataset.

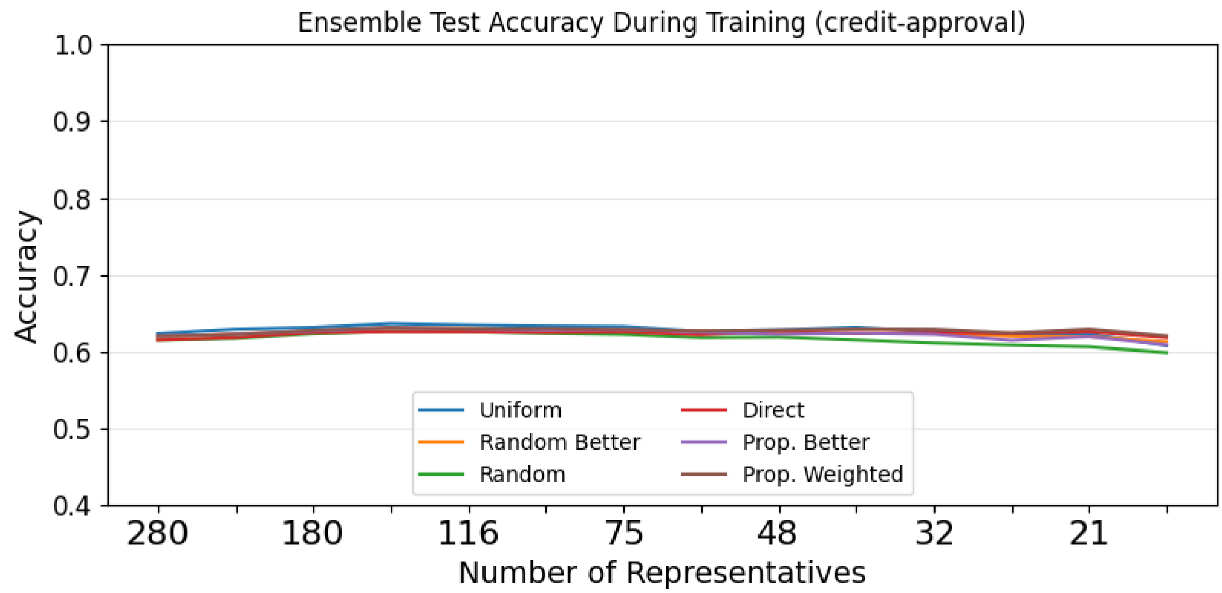


Figure A.6: Test accuracy during training on credit-approval dataset, averaged over 500 trials.

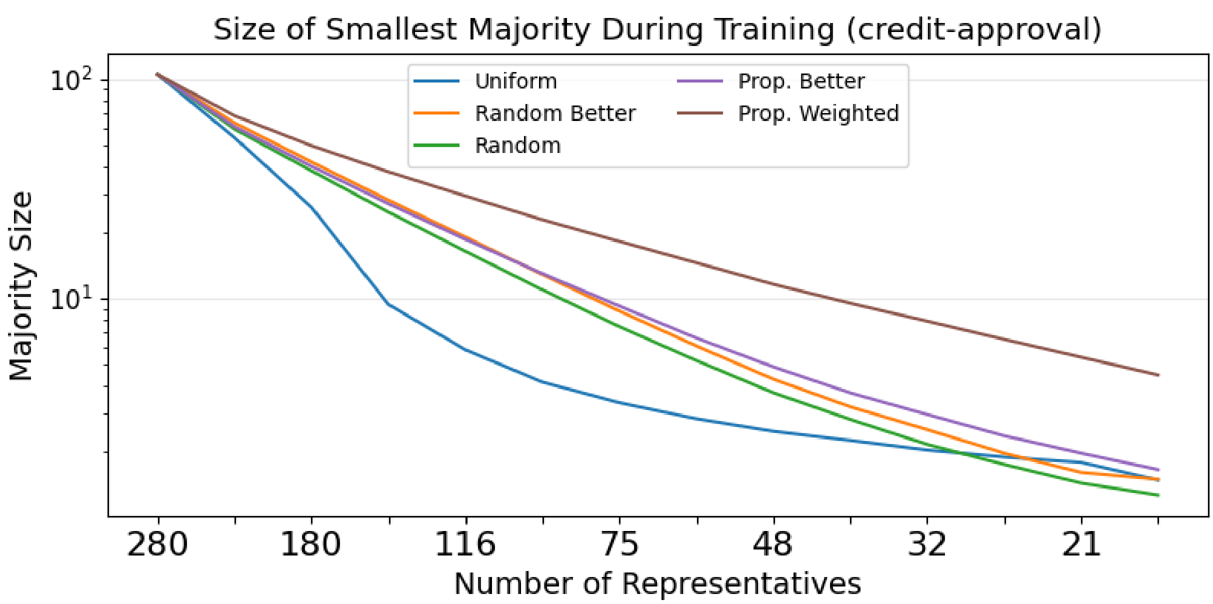


Figure A.7: Minimum majority size during training on the credit-approval dataset.

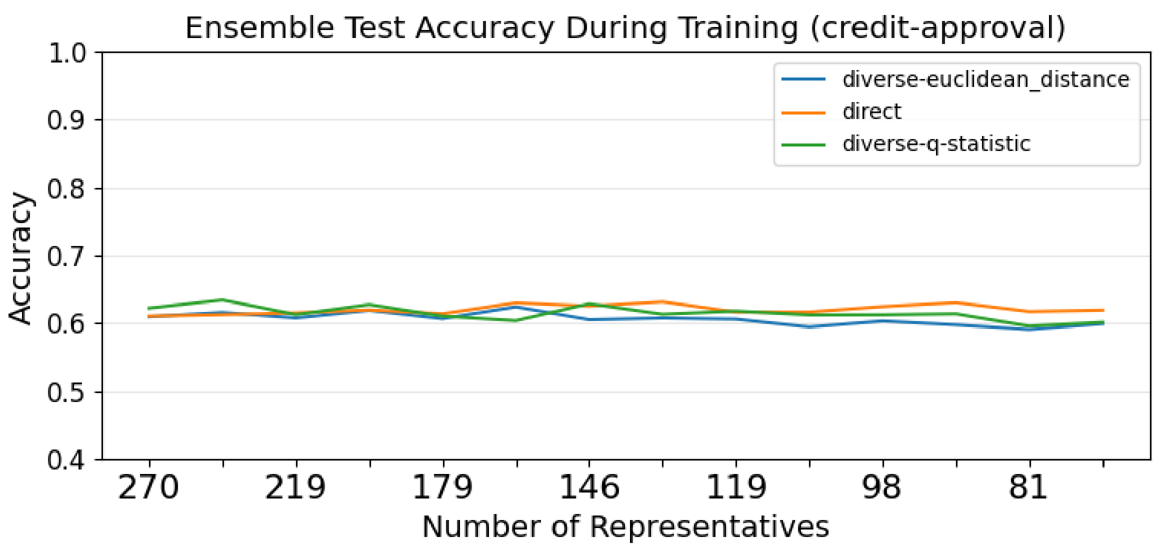


Figure A.8: Test accuracy during training on credit-approval dataset, averaged over 30 trials, using diversity metrics to guide delegation.

A.3 heart

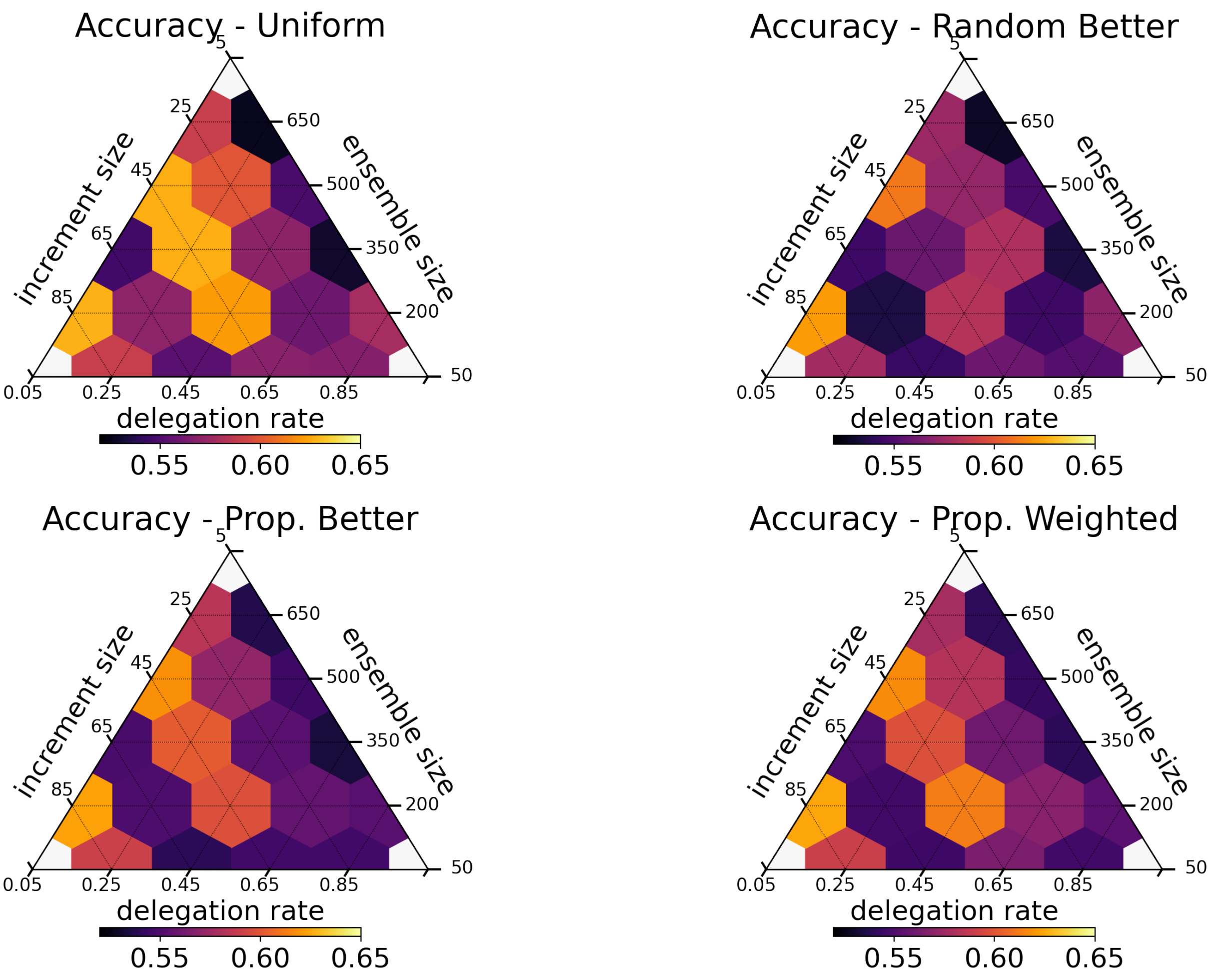


Figure A.9: Test accuracy of fully trained ensembles as parameters are varied. Results from heart dataset.

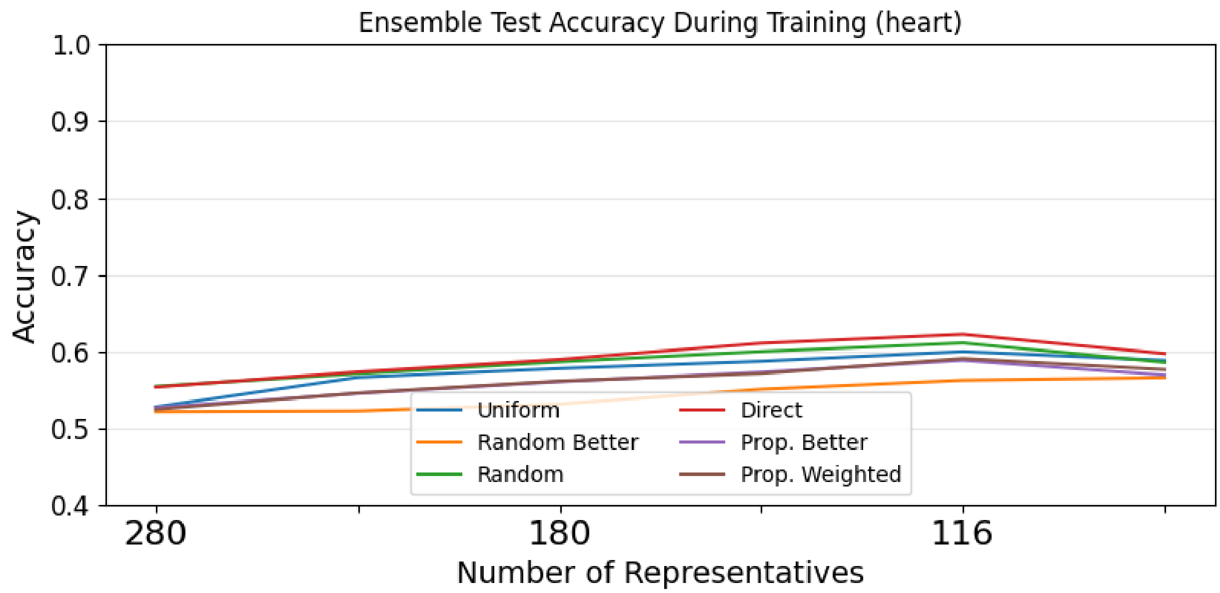


Figure A.10: Test accuracy during training on heart dataset, averaged over 500 trials.

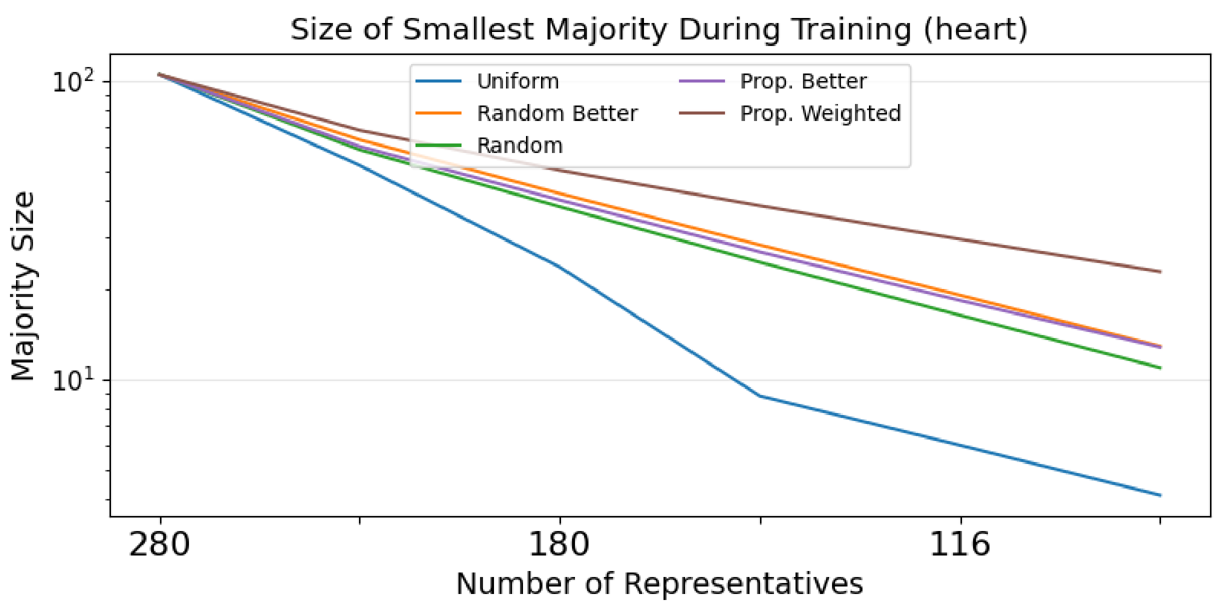


Figure A.11: Minimum majority size during training on the heart dataset.

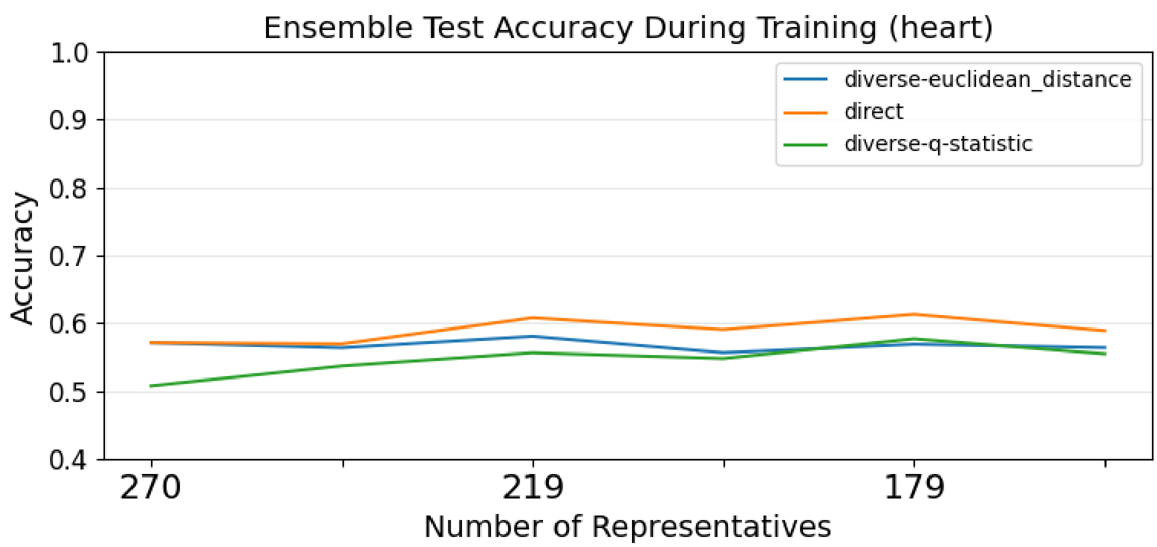


Figure A.12: Test accuracy during training on heart dataset, averaged over 30 trials, using diversity metrics to guide delegation.

A.4 ionosphere

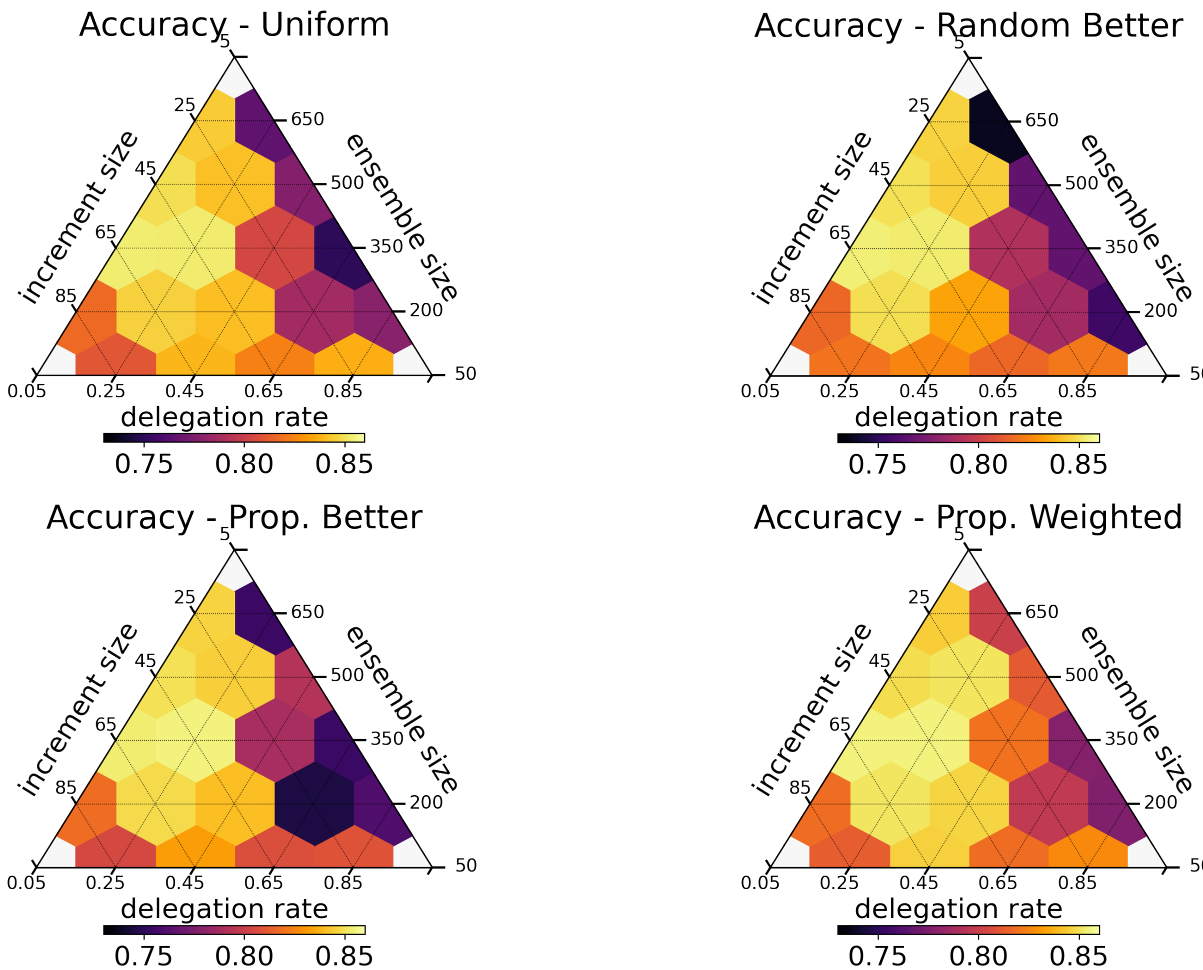


Figure A.13: Test accuracy of fully trained ensembles as parameters are varied. Results from ionosphere dataset.

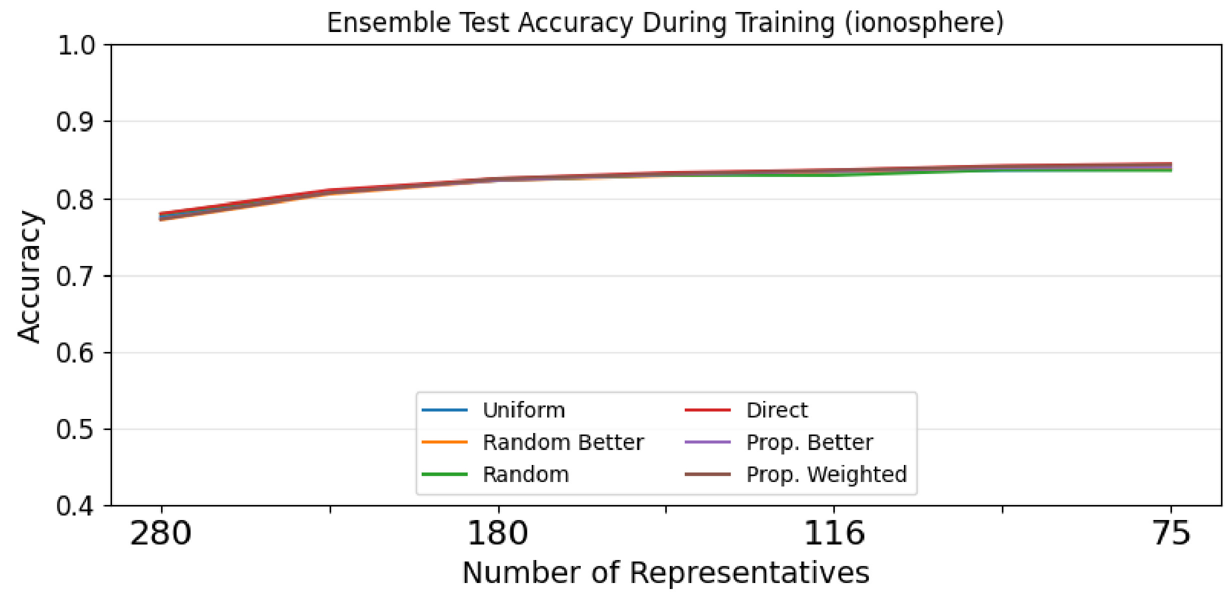


Figure A.14: Test accuracy during training on ionosphere dataset, averaged over 500 trials.

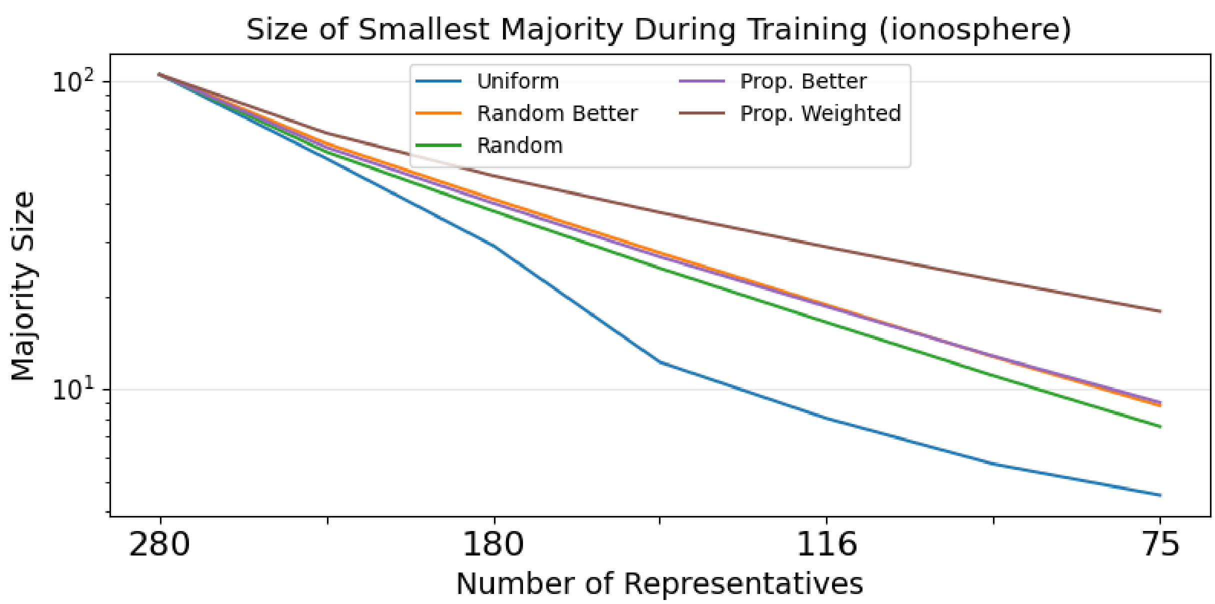


Figure A.15: Minimum majority size during training on the ionosphere dataset.

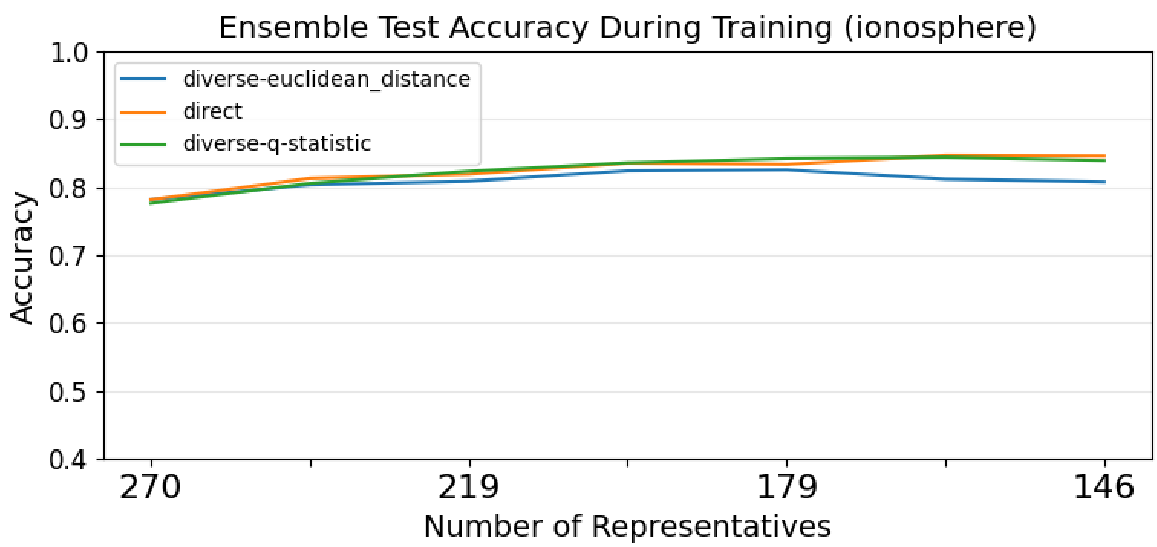


Figure A.16: Test accuracy during training ionosphere dataset, averaged over 30 trials, using diversity metrics to guide delegation.

A.5 kr-vs-kp

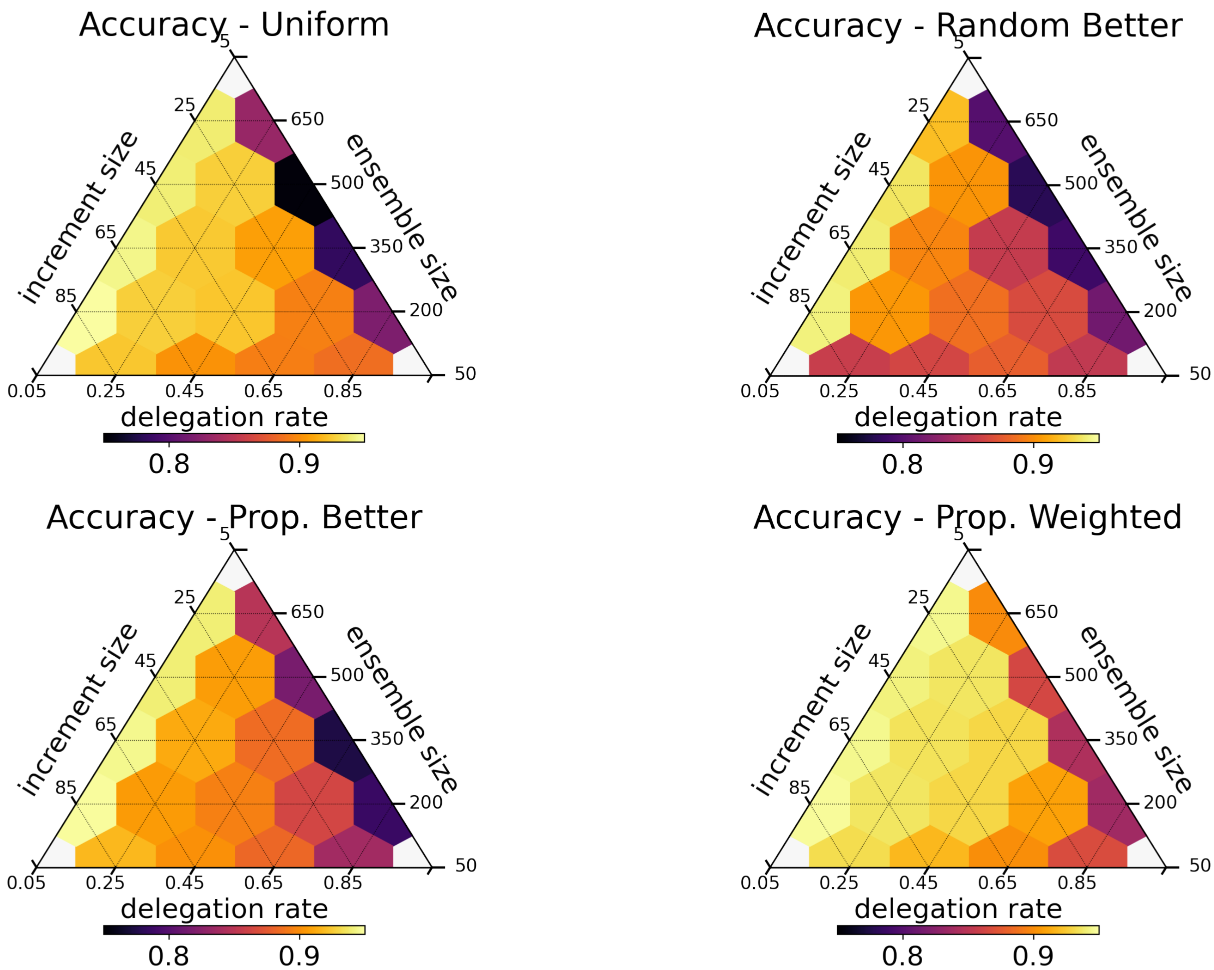


Figure A.17: Test accuracy of fully trained ensembles as parameters are varied. Results from kr-vs-kp dataset.

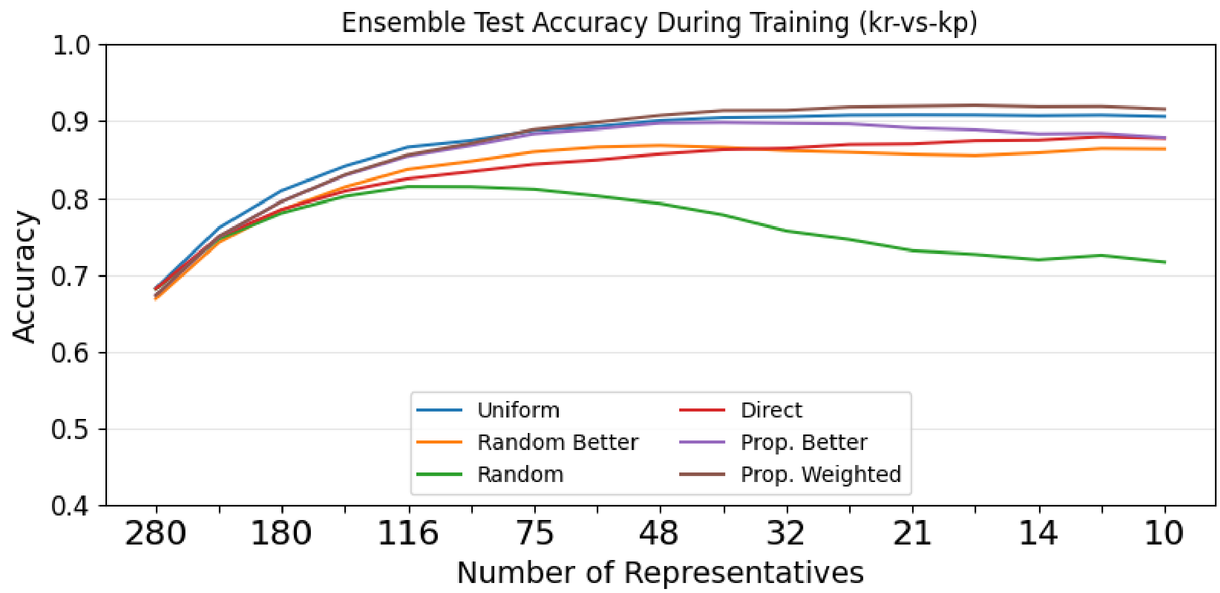


Figure A.18: Test accuracy during training on kr-vs-kp dataset, averaged over 500 trials.

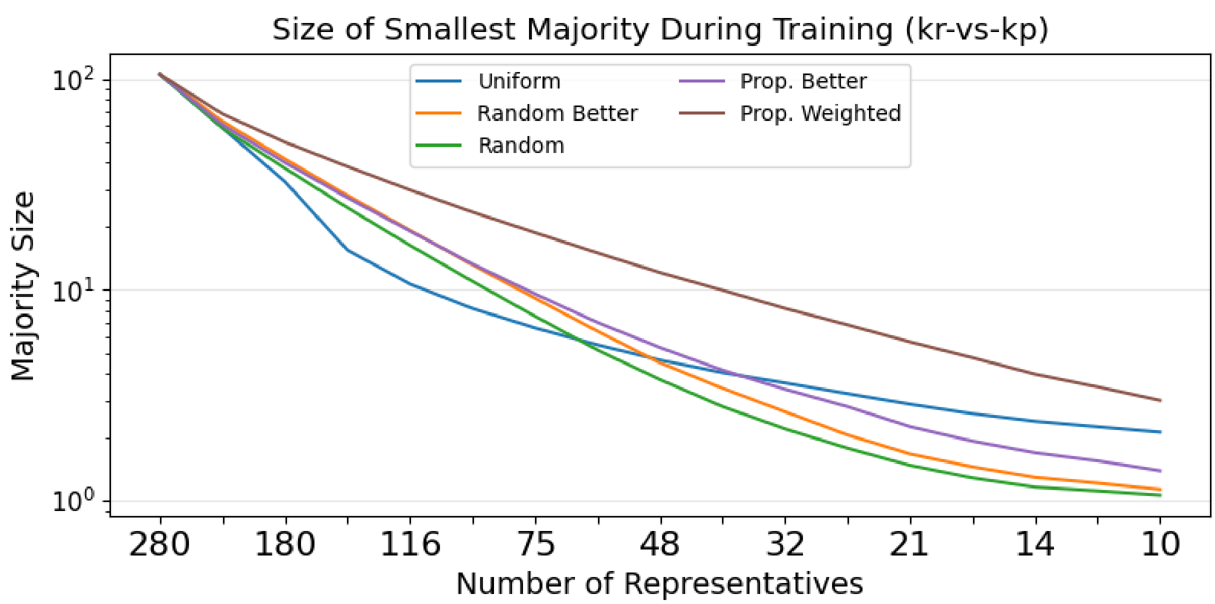


Figure A.19: Minimum majority size during training on the kr-vs-kp dataset.

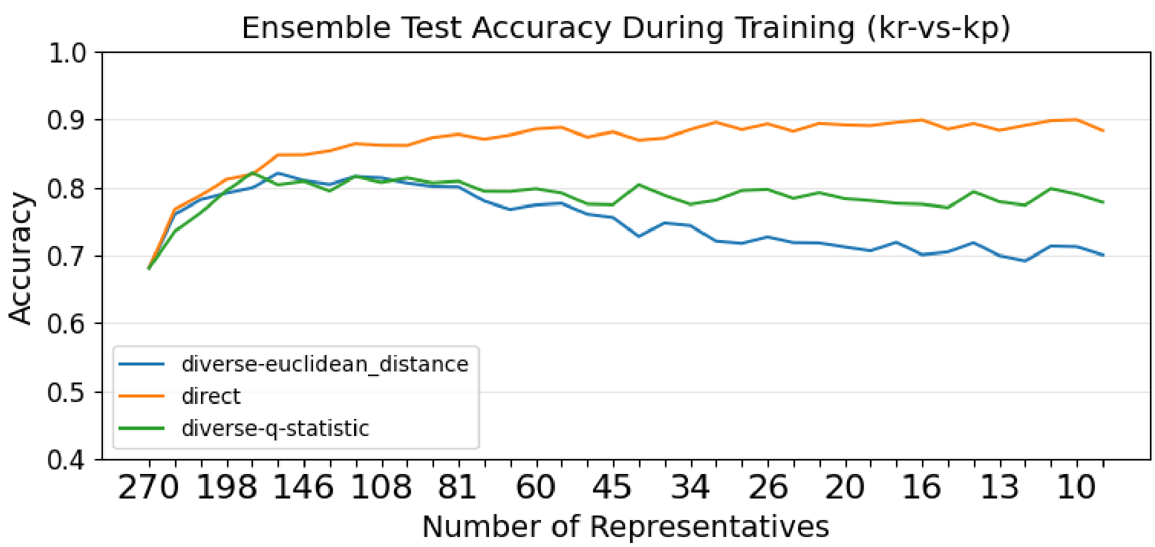


Figure A.20: Test accuracy during training on kr-vs-kp dataset, averaged over 30 trials, using diversity metrics to guide delegation.

A.6 occupancy-detection

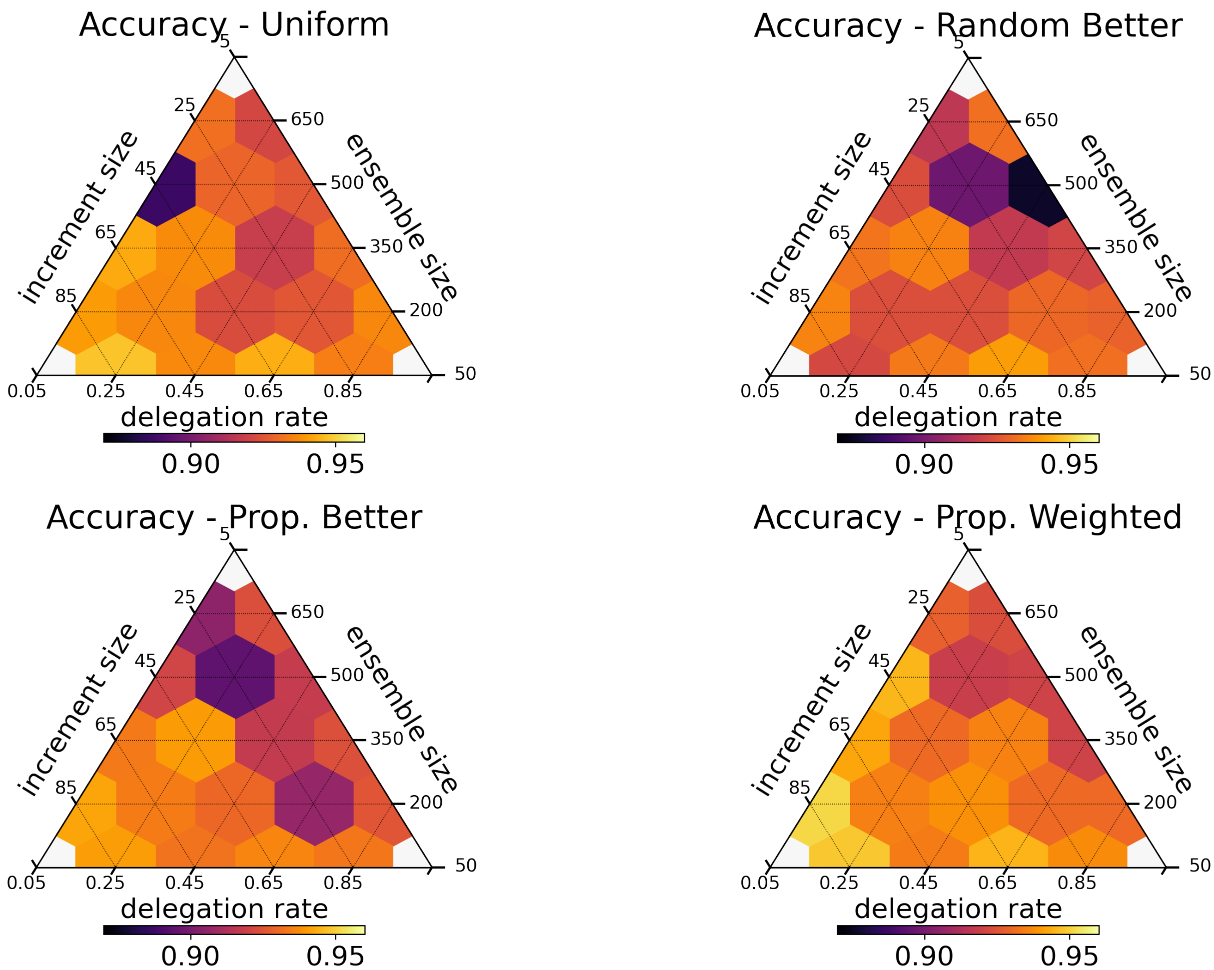


Figure A.21: Test accuracy of fully trained ensembles as parameters are varied. Results from occupancy-detection dataset.

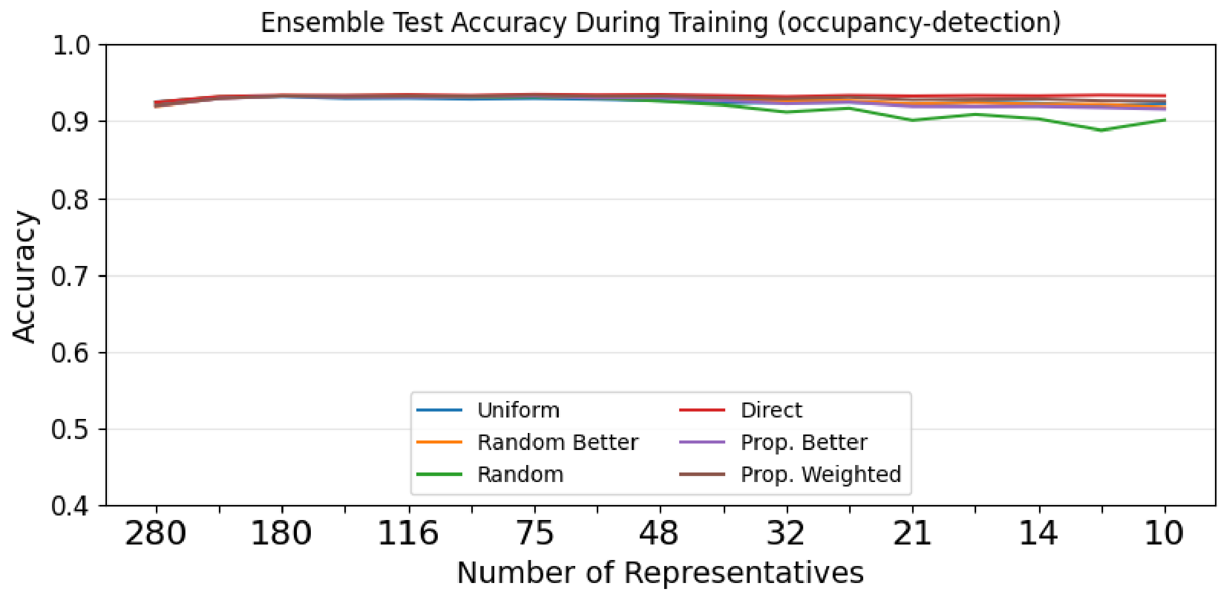


Figure A.22: Test accuracy during training on occupancy-detection dataset, averaged over 500 trials.

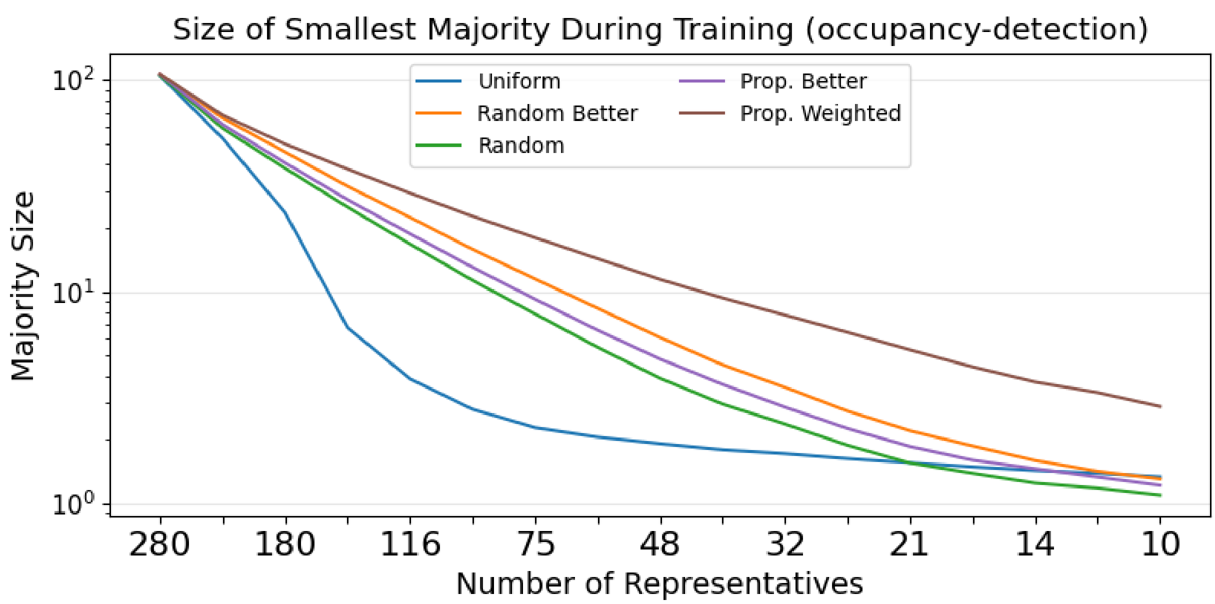


Figure A.23: Minimum majority size during training on the occupancy-detection dataset.

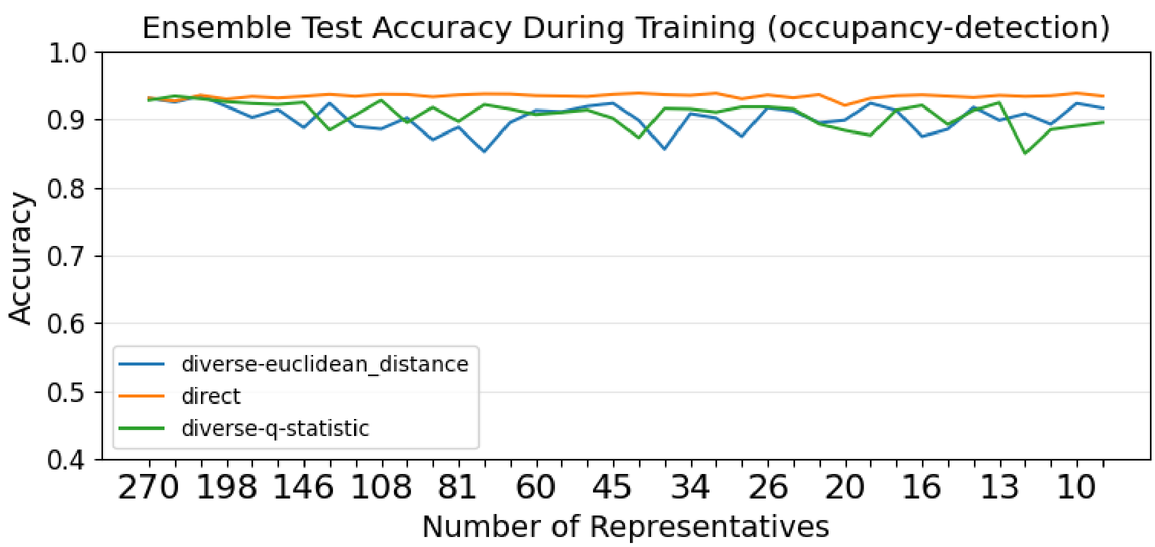


Figure A.24: Test accuracy during training on occupancy-detection dataset, averaged over 30 trials, using diversity metrics to guide delegation.

A.7 online-shoppers

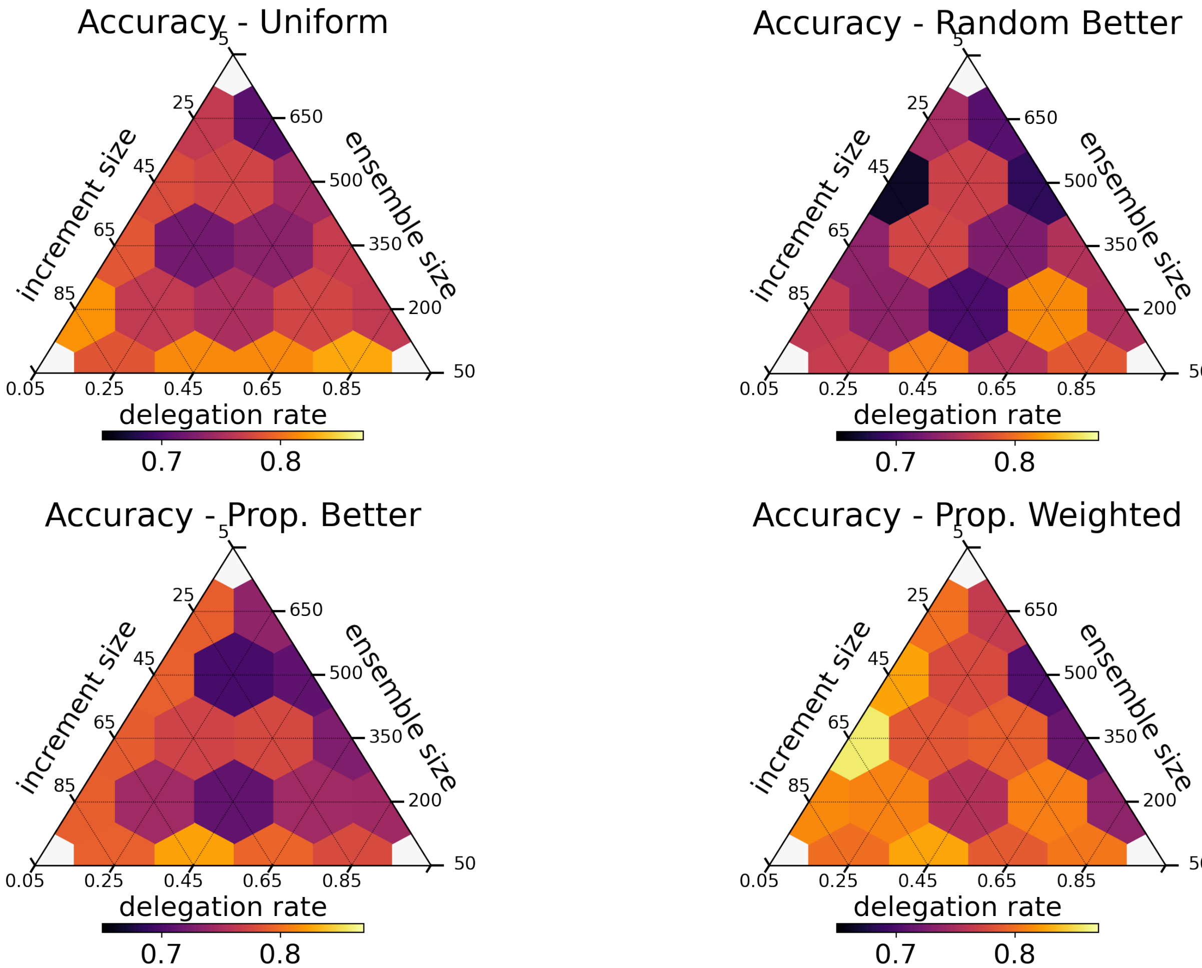


Figure A.25: Test accuracy of fully trained ensembles as parameters are varied. Results from online-shoppers dataset.

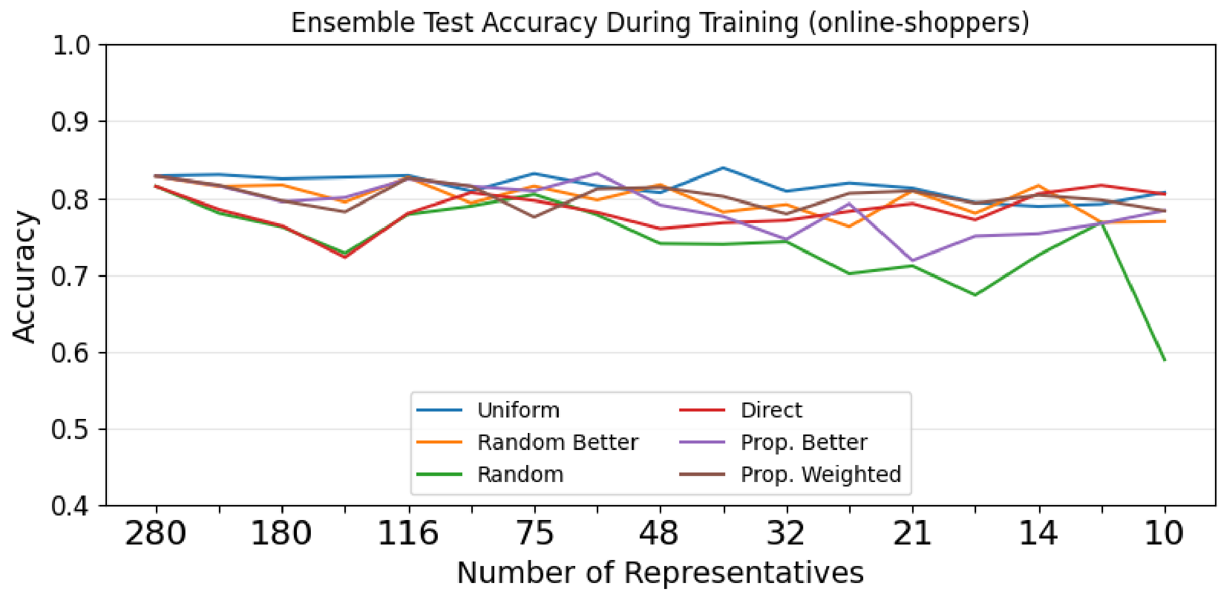


Figure A.26: Test accuracy during training on online-shoppers dataset, averaged over 500 trials.

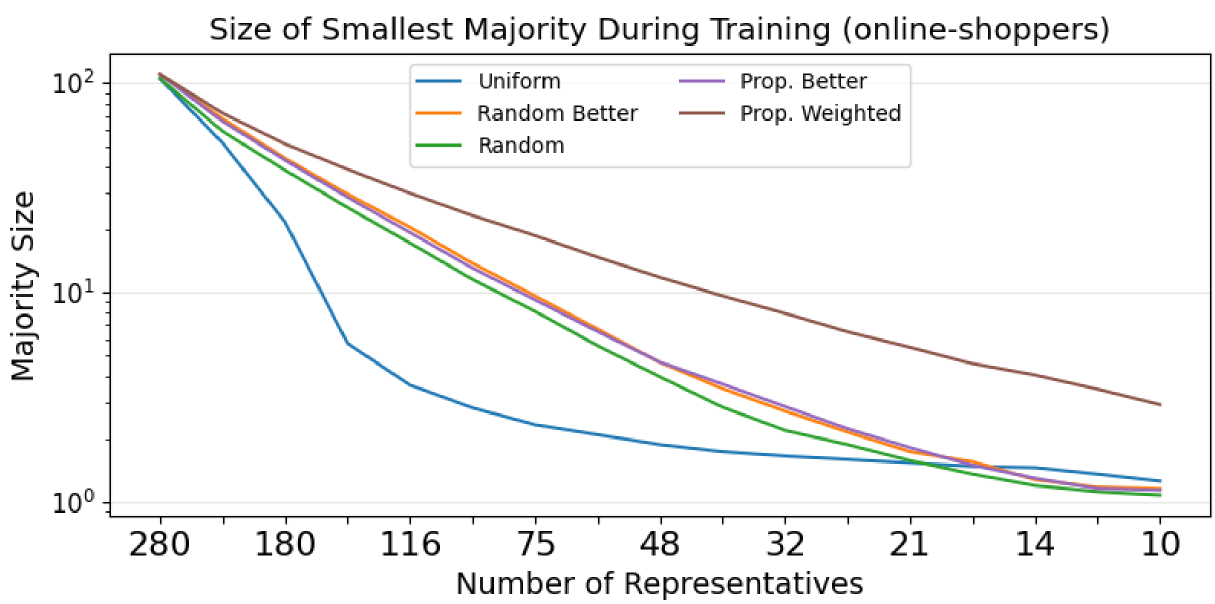


Figure A.27: Minimum majority size during training on the online-shoppers dataset.

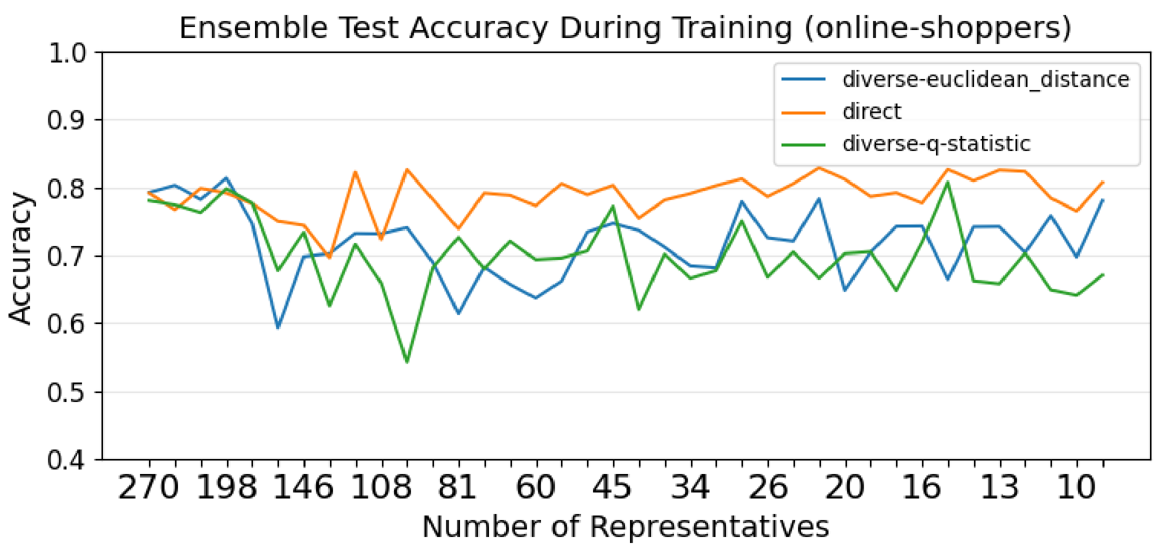


Figure A.28: Test accuracy during training on online-shoppers dataset, averaged over 30 trials, using diversity metrics to guide delegation.

A.8 spambase

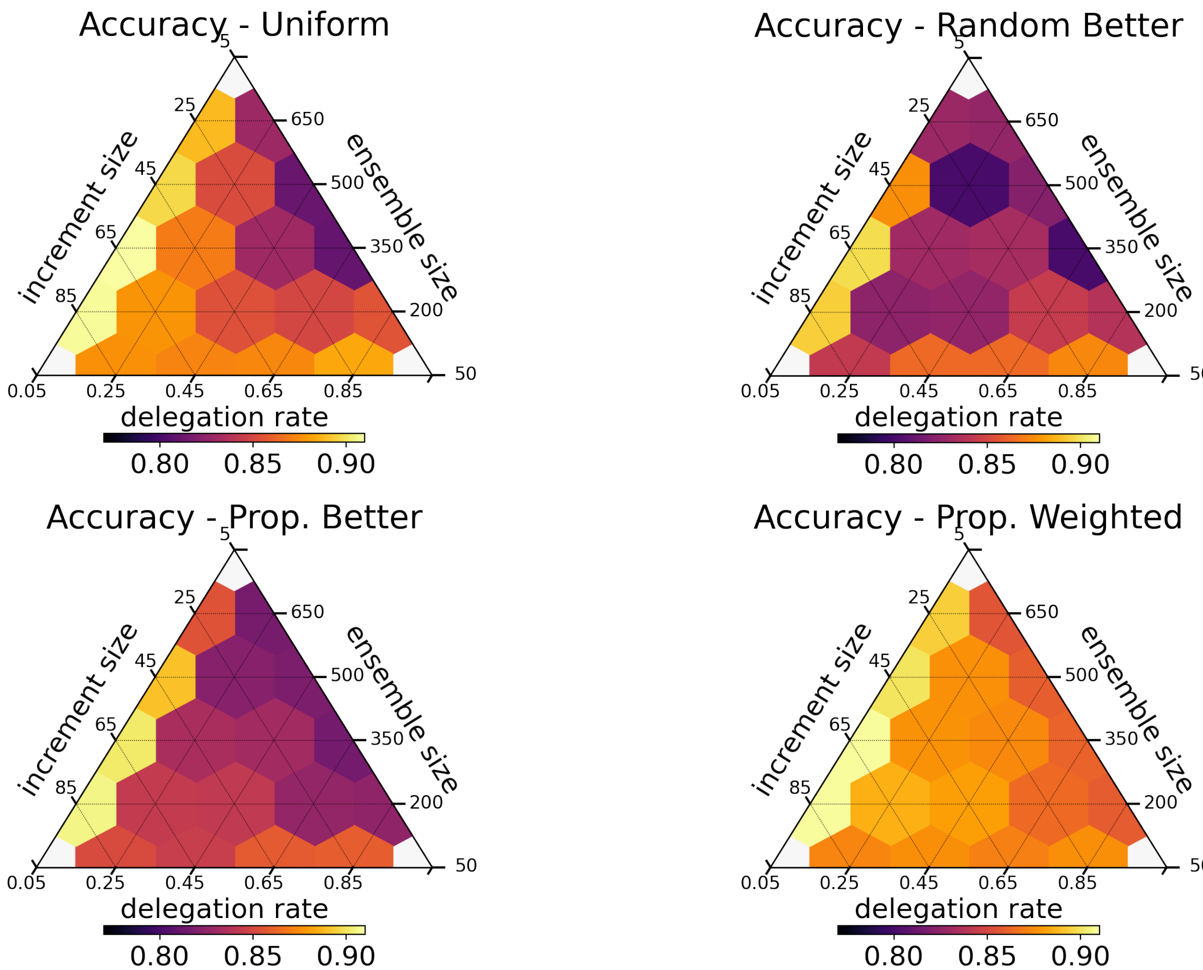


Figure A.29: Test accuracy of fully trained ensembles as parameters are varied. Results from spambase dataset.

A.9 Full Cost and Accuracy Results

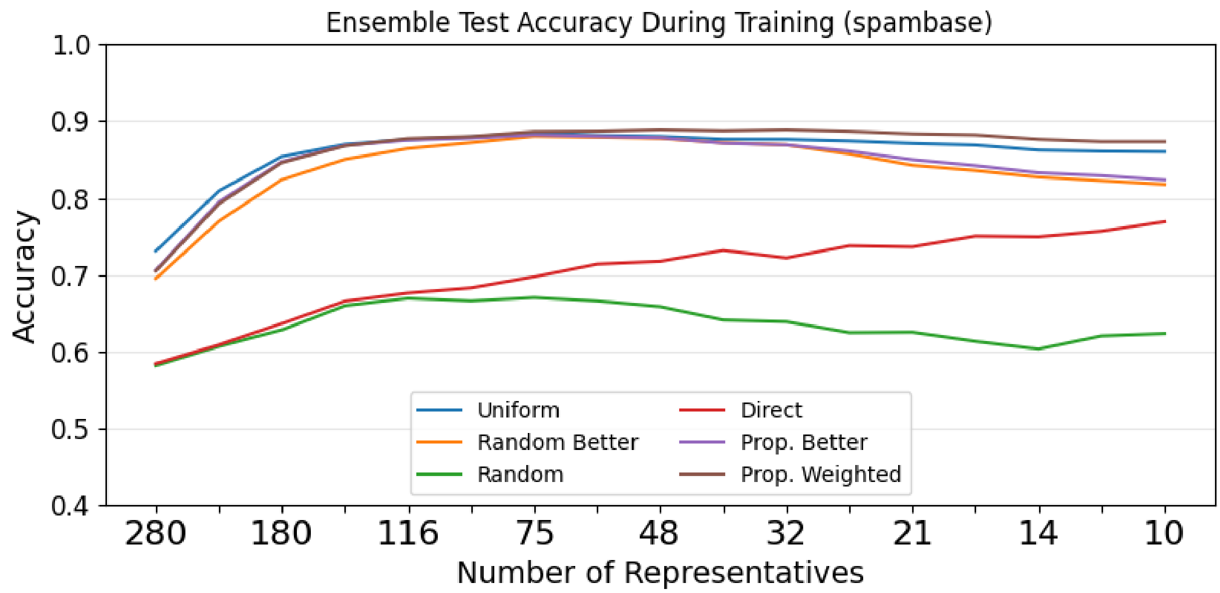


Figure A.30: Test accuracy during training on spambase dataset, averaged over 500 trials.

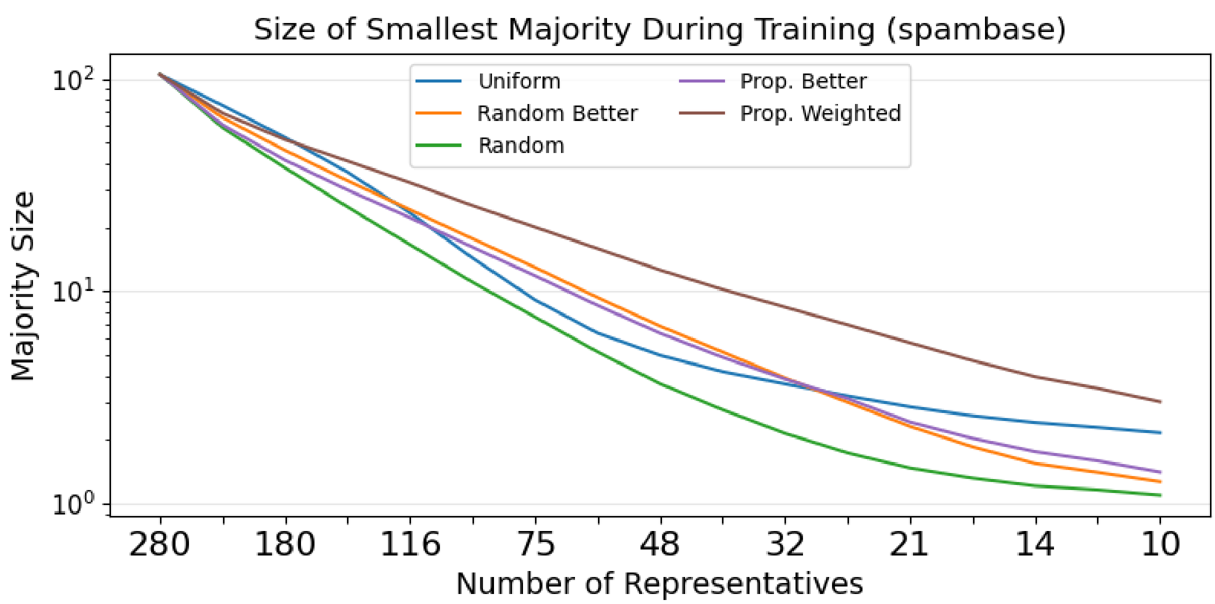


Figure A.31: Minimum majority size during training on the spambase dataset.

Ensemble	breast-cancer-w			credit-approval		
	Acc	F1	Cost	Acc	F1	Cost
Direct	0.907	0.926	1	0.628	0.585	1
Max Acc	0.908	0.928	0.782	0.632	0.588	0.818
Rand B Acc	0.907	0.927	0.782	0.628	0.584	0.83
Prop B Acc	0.908	0.928	0.782	0.63	0.586	0.817
Prop W Acc	0.907	0.927	0.782	0.631	0.586	0.817
Random Acc	0.907	0.926	0.783	0.627	0.584	0.816
Max Cost	0.881	0.897	**0.033**	0.612	0.575	**0.036**
Rand B Cost	0.87	0.886	**0.033**	0.595	0.563	**0.036**
Prop B Cost	0.867	0.886	**0.033**	0.593	0.541	**0.036**
Prop W Cost	0.9	0.92	**0.033**	0.609	0.584	**0.036**
Random Cost	0.863	0.881	**0.032**	0.595	0.521	**0.037**
Ada DT Full	0.953	0.932	0.039	0.818	0.833	0.07
Ada DT Small	0.957	0.938	0.001	0.852	0.862	0.002
Ada SGD Full	0.965	0.95	0.014	0.653	0.681	0.007
Ada SGD Small	0.965	0.95	0.013	0.65	0.674	0.007
Ensemble	**heart**			**ionosphere**		
	Acc	F1	Cost	Acc	F1	Cost
Direct	0.584	0.59	1	**0.854**	0.765	1
Max Acc	0.578	0.574	0.901	**0.852**	0.761	0.895
Rand B Acc	0.569	0.571	0.904	**0.852**	0.76	0.895
Prop B Acc	0.573	0.571	0.9	**0.853**	0.763	0.894
Prop W Acc	0.573	0.574	0.9	**0.853**	0.762	0.894
Random Acc	0.586	0.597	0.9	**0.854**	0.767	0.895
Max Cost	0.586	0.62	0.035	0.799	0.71	0.034
Rand B Cost	0.539	0.503	0.035	0.766	0.68	0.034
Prop B Cost	0.555	0.517	0.035	0.754	0.673	0.034
Prop W Cost	0.565	0.547	0.035	0.802	0.716	0.034
Random Cost	0.531	0.484	0.035	0.723	0.631	0.033
Ada DT Full	0.758	0.783	0.027	0.916	0.937	0.031
Ada DT Small	0.803	0.824	0.001	0.896	0.921	0.001
Ada SGD Full	0.683	0.718	0.01	0.861	0.898	0.025
Ada SGD Small	0.678	0.718	0.009	0.861	0.898	0.015

Table A.1: Accuracy, F1 Score, and Training Cost for all delegation mechanisms using cost minimizing parameters and accuracy maximizing parameters, compared with each variety of Adaboost used. Bold values indicate that delegation outperforms at least one Adaboost method. Results shown for four datasets with other datasets shown in Table A.2.

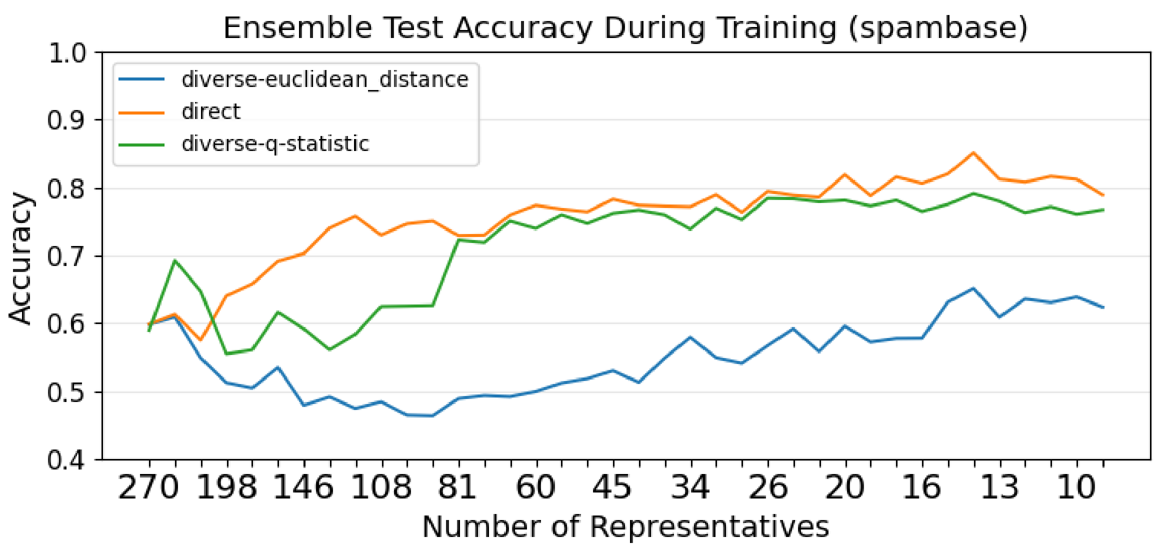


Figure A.32: Test accuracy during training on spambase dataset, averaged over 30 trials, using diversity metrics to guide delegation.

Ensemble	kr-vs-kp			occupancy-det		
	Acc	F1	Cost	Acc	F1	Cost
Direct	**0.91**	0.903	1	0.946	0.964	1
Max Acc	**0.946**	**0.942**	0.272	0.925	0.95	0.056
Rand B Acc	**0.943**	**0.94**	0.272	0.929	0.954	0.056
Prop B Acc	**0.946**	**0.942**	0.269	0.914	0.94	0.055
Prop W Acc	**0.947**	**0.943**	0.269	0.94	0.96	0.055
Random Acc	0.897	0.89	0.309	0.906	0.923	0.056
Max Cost	0.885	0.877	**0.026**	0.924	0.951	0.029
Rand B Cost	0.845	0.842	**0.026**	0.918	0.945	0.029
Prop B Cost	0.856	0.832	**0.026**	0.896	0.913	0.029
Prop W Cost	0.908	0.902	**0.026**	0.916	0.936	0.029
Ada DT Full	0.966	0.968	0.02	0.99	0.978	0.009
Ada DT Small	0.946	0.948	0.001	0.989	0.977	0
Ada SGD Full	0.941	0.944	0.06	0.984	0.966	0.005
Ada SGD Small	0.91	0.915	0.01	0.984	0.966	0.005

Ensemble	online-shoppers			spambase		
	Acc	F1	Cost	Acc	F1	Cost
Direct	0.869	**0.927**	1	**0.86**	**0.88**	1
Max Acc	0.78	**0.828**	0.059	**0.909**	**0.927**	0.197
Rand B Acc	0.719	**0.764**	0.059	**0.897**	**0.918**	0.197
Prop B Acc	0.784	**0.84**	0.058	**0.905**	**0.924**	0.198
Prop W Acc	0.843	**0.906**	0.058	**0.909**	**0.927**	0.198
Random Acc	0.737	**0.777**	0.058	**0.795**	**0.807**	0.198
Max Cost	0.757	**0.81**	0.029	**0.859**	**0.883**	0.03
Rand B Cost	0.735	**0.781**	0.029	**0.843**	**0.871**	0.03
Prop B Cost	0.711	**0.751**	0.029	**0.835**	**0.864**	0.029
Prop W Cost	0.768	**0.817**	0.029	**0.869**	**0.89**	0.029
Random Cost	0.78	**0.835**	0.028	0.675	0.641	0.03
Ada DT Full	0.888	0.605	0.019	0.934	0.916	0.026
Ada DT Small	0.89	0.62	0.001	0.916	0.891	0.001
Ada SGD Full	0.878	0.447	0.012	0.786	0.719	0.013
Ada SGD Small	0.879	0.444	0.011	0.791	0.742	0.012

Table A.2: Accuracy, F1 Score, and Training Cost for all delegation mechanisms using cost minimizing parameters and accuracy maximizing parameters, compared with each variety of Adaboost used. Bold values indicate that delegation outperforms at least one Adaboost method. Results shown for four datasets with other datasets shown in Table A.1.

Appendix B

Voting Rule Definitions

We now provide a brief description of each single winner voting rule that we refer to throughout the thesis. The large majority of rules listed in this appendix are used exclusively in Chapter 5. Recall that our model has n voters selecting $k = 1$ winners from a total of m alternatives. Note that in all cases we use lexicographic tie-breaking.

In many cases we define a rule in a way that allows for multiple winners. In all of these cases we use lexicographic tie-breaking to identify a single winner. For additional details or the exact implementations we used for our rules, we refer to the Preferential Voting Tools library, which provided an implementation for all single winner rules [1]. Many of our definitions are also adapted from the documentation for the Preferential Voting Tools library.

We will rely on two concepts in some rule definitions which we define preemptively to avoid redundancy: The **margin loss** of an alternative a_i is the maximum over all other alternatives a_j of the number of voters ranking $a_j \succ a_i$ subtract the number ranking $a_i \succ a_j$. The **majority graph** of a preference profile contains one node per alternative and has an edge from alternative a_i to a_j if a majority of voters prefer a_i to a_j.

Additionally, we highlight that we place rules in Fishburn's C3 class by default. That is, rules in C1 and C2 can be shown to be computable using tournament and weighted tournament information respectively. C3 rules are those for which no more restricted class *is known*. We do not include a proof that each rule is in the Fishburn class we have assigned, but we have relied upon the following facts to categorize many rules:

- Plurality scores are in C3.

[1] https://pref-voting.readthedocs.io/en/latest/index.html

- More generally, positional information about preferences (alternatives ranked first, last, second, etc.) is typically in C3.
- Global Borda scores (scores for each alternative in the complete preference profile) are in C2. This is an exception to the above generality.
- Borda scores for a subset of alternatives are no longer in C2 and are in C3.
- Comparing two alternatives across all voters is in C1 if identifying which alternative is preferred by a majority, or C2 if identifying how many prefer one over the other.

Anti-Plurality: This is a *C3* rule in Fishburn's classification. A positional scoring rule with the score vector $\mathbf{s} = (0, 0, ..., -1)$. Each voter awards one negative point to their least favourite alternative and the alternative(s) with the highest number of points wins.

Baldwin: This is a *C3* rule in Fishburn's classification. The alternative with the lowest Borda score (among remaining alternatives) is iteratively removed until all remaining alternatives have equal Borda score. The remaining alternatives are the winners.

Banks: This is a *C1* rule in Fishburn's classification. Say that a chain of alternatives is a sequence of alternatives where each alternative in the sequence is preferred by a majority of voters to the next alternative. The winners are the alternatives at the beginning of any chain with maximal length.

Beat Path: This is a *C2* rule in Fishburn's classification. If a_i is preferred by a majority to a_j, the margin of a_i over a_j is the number of voters with $a_i \succ a_j$ subtract the number of voters with $a_j \succ a_i$. A path from a_i to a_j is a list of alternatives where the preceding alternative is preferred by a majority to the succeeding alternative for each pair of adjacent alternatives. The strength of a path is the smallest margin in the path. a_i "defeats" a_j if the strongest path from a_i to a_j is stronger than the strongest path from a_j to a_i. The undefeated alternatives are the winners.

Benham: This is a *C3* rule in Fishburn's classification. If there is a Condorcet winner, that alternative wins. Otherwise, iteratively remove all alternatives with the lowest Plurality score until there is a Condorcet winner.

Bipartisan Set: Also referred to as C1 Maximal Lotteries. Find a probability distribution over alternatives such that sampling alternatives according to their probabilities will, in expectation, elect an alternative that is preferred in a maximal number of pairwise majority contests. Note: This method is randomized and thus does not precisely fit into the C1 definition (exactly computable using only the majority graph). However, as the method does not use any information additional to what is used by C1 methods we consider C1 the most appropriate label.

Blacks: This is a *C2* rule in Fishburn's classification. If there is a Condorcet winner, that alternative wins. Otherwise, return the Borda winner(s).

Borda-Minimax Faceoff: This is a *C2* rule in Fishburn's classification. If the Borda winners are the same as the Minimax winners, those alternative(s) win. Otherwise, do a pairwise comparison between each Borda winner with each Minimax winner. Add to the set of winning alternatives each alternative preferred by a majority in a pairwise comparison (or both in case of a tie).

Borda: This is a *C2* rule in Fishburn's classification. A positional scoring rule with the score vector $\mathbf{s} = (m-1, m-2, ..., 0)$. The Borda score of an alternative is the sum of scores given by each voter. Each voter awards a score to alternatives equal to the voter's rank of that alternative in the preference order. e.g. A voter awards $m-1$ points to their favourite, $m-2$ to their second favourite, etc. The highest scoring alternative(s) are the winner(s).

Bracket Voting: This is a *C3* rule in Fishburn's classification. Place the four alternatives with the highest Plurality scores into a bracket. The alternatives placed 1$^{\text{st}}$ and 4$^{\text{th}}$ face each other in a pairwise contest. The alternatives placed 2$^{\text{nd}}$ and 3$^{\text{rd}}$ face each other in a pairwise contest. The alternative from each contest preferred by more voters face each other. The winner of that pairwise contest is the overall winner.

Bucklin: This is a *C3* rule in Fishburn's classification. Proceed for up to m rounds. In round r, say that the number of votes an alternative receives is the number of voters for whom that alternative appears in the voter's top r preferences (e.g. in the first round, consider each voter's first preference). A round is the final round if any alternative(s) receive a number of votes more than half the number of voters. The winner(s) are all voters who receive a maximal number of votes.

Condorcet Plurality: This is a *C3* rule in Fishburn's classification. If there is a Condorcet winner, that alternative wins. Otherwise, return the Plurality winner(s).

Condorcet: This is a *C1* rule in Fishburn's classification. If a Condorcet winner exists, it is the winner. Otherwise, all alternatives are tied winners.

Coombs: This is a *C3* rule in Fishburn's classification. If there is a majority winner, that alternative wins. Otherwise, iteratively remove all alternatives that are ranked last by the largest number of voters until there is an alternative ranked first by a majority.

Copeland-Global-Borda: This is a *C2* rule in Fishburn's classification. Identify the Copeland winners. From those, return the alternative(s) with the highest Borda score over the complete preferences of each voter.

Copeland-Local-Borda: This is a *C3* rule in Fishburn's classification. Identify the Copeland winners. From those, return the alternative(s) with the highest Borda score, considering the subset of each voter's preferences that includes only the Copeland winners. NOTE: While Copeland is C1 and Borda is C2, this rule does not appear to trivially be in C2. Borda's rule does not satisfy the Independence of Irrelevant Alternative's axiom and the winners of the Borda rule may change when restricted to a subset of preferences in a way that may not be predictable using only C2 information.

Copeland: This is a *C1* rule in Fishburn's classification. The Copeland score of an alternative a_i is the number of alternatives that a_i is ranked above by a majority of voters subtract the number of alternatives ranked above a_i by a majority of voters. The winners are all alternatives with maximal Copeland score.

Daunou: This is a *C3* rule in Fishburn's classification. If there is a Condorcet winner, then that candidate is the winner. Otherwise, iteratively remove Condorcet losers until one does not exist then elect the Plurality winner(s).

GOCHA: This is a *C1* rule in Fishburn's classification. The set of winning alternatives is the smallest set such that every alternative inside the set is preferred by a *strict* majority to every alternative outside the set. Also referred to as the Schwartz set.

Instant Runoff: This is a *C3* rule in Fishburn's classification. If there is a majority winner, that alternative wins. Otherwise, iteratively remove all alternatives that are ranked first by the fewest voters until there is an alternative ranked first by a majority.

Knockout Voting: This is a *C2* rule in Fishburn's classification. Continue until one alternative remains. Find the two alternatives with the lowest global Borda scores. Remove the alternative from these two which is preferred by fewer voters in a pairwise comparison.

Llull: This is a *C1* rule in Fishburn's classification. Each alternative with a maximum Llull score is a winner. Each alternative a_i receives one point for every pairwise comparison in which a_i is preferred by a strict majority of voters, and $\frac{1}{2}$ points for each pairwise comparison in which a_i is preferred by exactly half of the voters.

Loss-Trimmer Voting: This is a *C2* rule in Fishburn's classification. Proceed until a Condorcet winner exists, which becomes the winner. In each round calculate the sum of all margin losses for each alternative. Remove the alternative with the largest sum of losses.

Minimax: This is a *C2* rule in Fishburn's classification. Elect the alternative(s) with the smallest margin loss.

Plurality: This is a *C3* rule in Fishburn's classification. A positional scoring rule with the score vector $\mathbf{s} = (1, 0, 0, ..., 0)$. Each voter awards one point to their favourite alternative . The highest scoring alternative(s) are the winner(s).

Raynaud: This is a *C2* rule in Fishburn's classification. Iteratively remove all alternatives with the largest margin loss, until a single alternative remains/all remaining alternatives have the same loss.

Simple Stable Voting: This is a *C2* rule in Fishburn's classification. Defined recursively. If there is only one alternative, they are the winner. Otherwise, order all pairs of alternatives (a_i, a_j) in descending order of the margin of a_i over a_j. Select as winner a_i from the first pair (a_i, a_j) such that a_i is the Simple Stable Voting winner on the election without a_j.

Simplified Bucklin: This is a *C3* rule in Fishburn's classification. Proceed for up to m rounds. In round r, say that the number of votes an alternative receives is the number of voters for whom that alternative appears in the voter's top r preferences (e.g. in the first round, consider each voter's first preference). In a given round, if any alternatives receive a number of votes more than half the number of voters they are the winners and no further rounds occur. This differs from Bucklin by returning all alternatives with a majority of votes, rather than only the subset of alternatives with a maximal number of votes.

Slater: This is a *C1* rule in Fishburn's classification. Consider the majority graph of preferences. Convert the graph to a ranking by ordering alternatives in descending order of their out-degree. The Slater ranking is the ranking derived from G' where G' is the graph which reverses a minimal number of edges from G to result in a strict ranking.

Split Cycle: This is a *C2* rule in Fishburn's classification. Define a majority cycle as a sequence of alternatives beginning and ending with the same alternative, such that each alternative in the sequence is preferred by a majority of voters to the next alternative in the sequence. Say that a_i defeats a_j in a majority cycle if (1) more voters prefer a_i over a_j than prefer a_j over a_i, and (2) the margin of victory of a_i over a_j is not the smallest margin in the cycle. All undefeated alternatives are the winners.

Stable Voting: This is a *C2* rule in Fishburn's classification. Defined recursively. If there is only one alternative, they are the winner. Otherwise, order pairs of alternatives (a_i, a_j) in descending order of the margin of a_i over a_j, including only the pairs where a_i is undefeated in the Split Cycle rule. Select as winner a_i from the first pair (a_i, a_j) such that a_i is the Simple Stable Voting winner on the election without a_j.

Strict Nanson: This is a *C3* rule in Fishburn's classification. Iteratively remove all alternatives with a Borda score (calculated from only the remaining alternatives) below the mean Borda score until all remaining alternatives have equal Borda scores.

Superior Voting: This is a *C3* rule in Fishburn's classification. Say that a_i is superior to a_j if a majority prefer a_i to a_j. Each voter awards a point to their first preference if no voter is superior to their first preference. Otherwise, the voter awards a point to the alternative superior to their first preference. The alternative with the most points is the winner.

Tideman Alternative GOCHA: This is a *C3* rule in Fishburn's classification. Identify the GOCHA winner(s). If there is only one, that is the winner. Otherwise, beginning from the GOCHA winners, iteratively remove all alternatives ranked first by the fewest voters until all remaining alternatives are ranked first by the same number of voters.

Tideman Alternative Top Cycle: This is a *C3* rule in Fishburn's classification. Identify the Top Cycle winner(s). If there is only one, that is the winner. Otherwise, beginning from the Top Cycle winners, iteratively remove all alternatives ranked first by the fewest voters until all remaining alternatives are ranked first by the same number of voters.

Top Cycle: This is a *C1* rule in Fishburn's classification. The set of winning alternatives is the smallest set such that every alternative inside the set is preferred by a *weak* majority to every alternative outside the set. Also referred to as the Smith set.

Uncovered Set: This is a *C1* rule in Fishburn's classification. Say that a_i defeats a_j if (1) a_i is preferred by a majority to a_j, and (2) for all alternatives a_k, if a_k is preferred by a majority to a_i then a_k is also preferred by a majority to a_j. The winners are all alternatives that are undefeated.

Weak Nanson: This is a *C3* rule in Fishburn's classification. Iteratively remove all alternatives with a Borda score (calculated from only the remaining alternatives) below or equal to the mean Borda score until all remaining alternatives have equal Borda scores.

Weighted Bucklin: This is a *C3* rule in Fishburn's classification. Proceed for up to m rounds. In round r, say that the score an alternative a_i receives is the sum for $1 \leq c \leq r$ of the product of the normalized Borda score that would be given to an alternative in rank c and the number of alternatives ranking a_i in position c. A round is the final round if any alternative(s) receive a number of votes equal to more than half the number of voters. The winner(s) are all voters who receive a maximal score.

Appendix C

Additional Results of Learning Single Winner Voting Rules

In this appendix we include the results of learning to approximate single winner voting rules for each feature set and preference distribution we trained upon. We show the test accuracy for each rule on each of the three datasets we used in Chapter 5: two identically generated sets and one generated from the same preference distributions, where we excluded any preference profiles with a Condorcet winner.

C.1 Results by Training Features Used

Target Rule	M	W	R	MW	MR	WR	MWR
Banks*	0.87	0.74	0.71	0.87	0.87	0.72	**0.87**
Bipartisan Set*	0.86	0.74	0.7	0.86	0.86	0.7	**0.86**
Condorcet*	0.89	0.73	0.7	0.89	**0.89**	0.71	0.89
Copeland*	0.88	0.75	0.72	0.88	**0.88**	0.72	0.88
GOCHA*	0.88	0.74	0.7	0.88	0.88	0.71	**0.88**
Llull*	0.88	0.75	0.71	0.88	0.88	0.72	**0.89**
Slater*	0.87	0.74	0.71	0.88	0.88	0.72	**0.88**
Top Cycle*	0.88	0.73	0.7	0.88	0.88	0.71	**0.88**
Uncovered Set*	0.88	0.74	0.71	0.88	0.88	0.71	**0.88**
Beat Path*	0.85	0.75	0.71	0.86	0.86	0.71	**0.86**
Blacks*	0.84	0.76	0.74	**0.84**	0.84	0.74	0.84
<u>Borda</u>	0.75	**0.82**	0.79	0.78	0.76	0.8	0.78
B-M Faceoff*	0.86	0.75	0.71	0.86	0.86	0.72	**0.86**
C-Global-Borda*	0.86	0.76	0.73	0.86	0.86	0.73	**0.86**
L-T Voting*	0.86	0.75	0.71	0.86	0.86	0.71	**0.86**
Minimax*	0.85	0.75	0.71	0.86	0.86	0.71	**0.86**
Raynaud*	0.85	0.74	0.7	0.85	**0.85**	0.71	0.85
S. Stable Voting*	0.85	0.75	0.71	0.85	0.85	0.72	**0.85**
Split Cycle*	0.86	0.75	0.71	0.86	0.86	0.71	**0.86**
Stable Voting*	0.85	0.75	0.72	0.85	0.85	0.72	**0.85**

Table C.1: Accuracy on the primary test set from learning each voting rule in Fishburn's C1 (red) and C2 (blue) classes, averaged over all training distributions for each combination of features. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold. Features are labelled by as M, W, R respectively for $\mathcal{M}^{\text{majority}}$, $\mathcal{M}^{\text{weighted}}$, and $\mathcal{M}^{\text{ranked}}$ features.

Target Rule	M	W	R	MW	MR	WR	MWR
Anti-Plurality	0.42	0.51	0.72	0.47	0.67	**0.73**	0.68
Baldwin*	0.86	0.74	0.71	0.86	0.86	0.71	**0.86**
Benham*	0.83	0.73	0.71	0.84	**0.84**	0.72	0.84
Bracket Voting	0.82	0.72	0.71	0.82	0.82	0.72	**0.83**
Bucklin	**0.7**	0.67	0.69	0.7	0.68	0.7	0.69
Cond. Plurality*	0.82	0.72	0.73	0.82	0.84	0.73	**0.84**
Coombs	0.82	0.73	0.69	**0.82**	0.79	0.69	0.8
C-Local-Borda*	0.87	0.75	0.72	**0.87**	0.87	0.72	0.87
Daunou*	0.82	0.73	0.72	0.82	0.82	0.73	**0.83**
Instant Runoff	0.75	0.68	0.72	0.75	0.76	0.73	**0.77**
Knockout Voting	0.85	0.75	0.72	0.85	0.85	0.73	**0.85**
Plurality	0.64	0.61	0.88	0.64	0.81	**0.89**	0.83
S. Bucklin	0.65	0.66	0.72	0.69	0.73	0.73	**0.73**
Strict Nanson*	0.86	0.75	0.71	0.86	0.86	0.71	**0.86**
Superior Voting*	0.84	0.73	0.71	0.85	0.85	0.72	**0.85**
T. Alt. GOCHA*	0.85	0.74	0.71	0.85	0.85	0.72	**0.86**
T. Alt. Top Cycle*	0.84	0.73	0.71	0.84	0.84	0.72	**0.84**
Weak Nanson*	0.86	0.76	0.72	0.86	0.86	0.72	**0.86**
Weighted Bucklin	**0.75**	0.72	0.74	0.75	0.74	0.74	0.75

Table C.2: Accuracy on the primary test set from learning each voting rule in Fishburn's C3 class, averaged over all training distributions for each combination of features. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold. Features are labelled by as M, W, R respectively for $\mathcal{M}^{\text{majority}}$, $\mathcal{M}^{\text{weighted}}$, and $\mathcal{M}^{\text{ranked}}$ features.

Target Rule	M	W	R	MW	MR	WR	MWR
Banks*	0.89	0.79	0.76	0.89	**0.9**	0.76	0.9
Bipartisan Set*	0.87	0.78	0.74	0.88	0.88	0.75	**0.88**
Condorcet*	0.9	0.77	0.73	0.9	0.9	0.74	**0.91**
Copeland*	0.9	0.8	0.76	0.9	0.91	0.77	**0.91**
GOCHA*	0.9	0.79	0.76	0.9	0.91	0.76	**0.91**
Llull*	0.9	0.8	0.76	0.91	0.91	0.77	**0.91**
Slater*	0.9	0.8	0.76	0.9	0.9	0.77	**0.91**
Top Cycle*	0.89	0.77	0.74	0.89	0.9	0.74	**0.9**
Uncovered Set*	0.9	0.78	0.75	0.9	0.9	0.75	**0.9**
Beat Path*	0.9	0.8	0.76	0.9	0.9	0.77	**0.9**
Blacks*	0.88	0.81	0.77	0.88	**0.88**	0.78	0.88
Borda	0.81	**0.84**	0.82	0.82	0.81	0.83	0.82
B-M Faceoff*	0.9	0.8	0.76	0.9	0.9	0.77	**0.9**
C-Global-Borda*	0.89	0.81	0.76	0.9	0.9	0.77	**0.9**
L-T Voting*	0.9	0.8	0.76	0.9	0.9	0.77	**0.9**
Minimax*	0.9	0.8	0.76	0.9	0.9	0.77	**0.9**
Raynaud*	0.89	0.79	0.75	0.89	0.89	0.76	**0.89**
S. Stable Voting*	0.89	0.8	0.76	0.89	0.89	0.76	**0.9**
Split Cycle*	0.9	0.8	0.76	0.9	0.9	0.77	**0.9**
Stable Voting*	0.89	0.8	0.76	0.89	0.89	0.77	**0.89**

Table C.3: Accuracy on the secondary test set from learning each voting rule in Fishburn's C1 (red) and C2 (blue) classes, averaged over all training distributions for each combination of features. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold. Features are labelled by as M, W, R respectively for $\mathcal{M}^{\text{majority}}$, $\mathcal{M}^{\text{weighted}}$, and $\mathcal{M}^{\text{ranked}}$ features.

Target Rule	M	W	R	MW	MR	WR	MWR
Anti-Plurality	0.39	0.47	**0.7**	0.44	0.66	0.7	0.67
Baldwin*	0.89	0.79	0.76	0.9	0.9	0.76	**0.9**
Benham*	0.86	0.77	0.75	**0.87**	0.86	0.75	0.86
Bracket Voting	0.85	0.76	0.76	0.85	0.86	0.77	**0.86**
Bucklin	0.7	0.68	0.67	**0.7**	0.64	0.67	0.66
Cond. Plurality*	0.84	0.76	0.75	0.84	0.86	0.76	**0.86**
Coombs	0.86	0.78	0.72	**0.86**	0.81	0.73	0.82
C-Local-Borda*	0.9	0.8	0.76	**0.9**	0.9	0.77	0.9
Daunou*	0.85	0.76	0.75	0.85	0.84	0.75	**0.85**
Instant Runoff	0.72	0.66	0.74	0.72	0.75	0.75	**0.76**
Knockout Voting	0.89	0.81	0.76	0.89	0.89	0.77	**0.89**
Plurality	0.61	0.58	0.89	0.61	0.82	**0.91**	0.84
S. Bucklin	0.59	0.63	0.69	0.64	0.67	**0.69**	0.68
Strict Nanson*	0.89	0.8	0.76	0.9	0.9	0.77	**0.9**
Superior Voting*	0.87	0.78	0.76	0.87	0.88	0.77	**0.88**
T. Alt. GOCHA*	0.9	0.79	0.76	0.9	0.9	0.76	**0.9**
T. Alt. Top Cycle*	0.87	0.77	0.75	0.87	0.87	0.75	**0.87**
Weak Nanson*	0.89	0.8	0.76	0.89	0.89	0.76	**0.9**
Weighted Bucklin	**0.8**	0.76	0.76	0.79	0.78	0.76	0.78

Table C.4: Accuracy on the secondary test set from learning each voting rule in Fishburn's C3 class, averaged over all training distributions for each combination of features. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold. Features are labelled by as M, W, R respectively for $\mathcal{M}^{\text{majority}}$, $\mathcal{M}^{\text{weighted}}$, and $\mathcal{M}^{\text{ranked}}$ features.

Target Rule	M	W	R	MW	MR	WR	MWR
Banks*	0.81	0.55	0.51	0.81	0.82	0.51	**0.82**
Bipartisan Set*	0.57	0.44	0.42	**0.57**	0.57	0.42	0.57
Condorcet*	**0.84**	0.47	0.41	0.84	0.84	0.42	0.84
Copeland*	0.84	0.56	0.52	0.85	0.85	0.53	**0.85**
GOCHA*	0.84	0.54	0.5	0.85	0.85	0.51	**0.85**
Llull*	0.85	0.56	0.51	0.86	0.86	0.52	**0.87**
Slater*	0.83	0.55	0.51	0.83	0.83	0.51	**0.83**
Top Cycle*	0.8	0.53	0.49	0.8	0.8	0.49	**0.81**
Uncovered Set*	0.83	0.52	0.47	0.84	0.84	0.48	**0.84**
Beat Path*	0.8	0.56	0.51	0.8	0.8	0.52	**0.81**
Blacks*	0.54	**0.69**	0.65	0.56	0.58	0.65	0.61
<u>Borda</u>	0.53	**0.76**	0.72	0.62	0.61	0.73	0.65
B-M Faceoff*	0.79	0.57	0.52	0.79	0.79	0.53	**0.79**
C-Global-Borda*	0.64	0.63	0.58	0.66	0.67	0.59	**0.68**
L-T Voting*	0.8	0.53	0.49	0.81	0.8	0.49	**0.81**
Minimax*	0.8	0.56	0.51	0.8	0.8	0.52	**0.8**
Raynaud*	0.75	0.51	0.47	0.75	0.75	0.47	**0.75**
S. Stable Voting*	0.63	0.59	0.52	0.64	0.65	0.53	**0.66**
Split Cycle*	0.79	0.56	0.51	0.79	0.79	0.52	**0.8**
Stable Voting*	0.63	0.59	0.52	0.64	0.66	0.53	**0.66**

Table C.5: Accuracy on the test set with no Condorcet winner from learning each voting rule in Fishburn's C1 (red) and C2 (blue) classes, averaged over all training distributions for each combination of features. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold. Features are labelled by as M, W, R respectively for $\mathcal{M}^{\text{majority}}$, $\mathcal{M}^{\text{weighted}}$, and $\mathcal{M}^{\text{ranked}}$ features.

Target Rule	M	W	R	MW	MR	WR	MWR
Anti-Plurality	0.37	0.43	**0.7**	0.38	0.64	0.7	0.65
Baldwin*	0.76	0.52	0.47	0.77	0.76	0.48	**0.77**
Benham*	0.57	0.43	0.51	0.57	0.61	0.52	**0.61**
Bracket Voting	0.53	0.42	0.54	0.53	0.65	0.54	**0.65**
Bucklin	0.45	0.48	0.6	0.46	0.54	**0.6**	0.56
Cond. Plurality*	0.42	0.36	0.58	0.42	0.58	0.58	**0.6**
Coombs	0.63	0.49	0.43	**0.63**	0.62	0.44	0.61
C-Local-Borda*	0.81	0.56	0.52	**0.82**	0.81	0.52	0.81
Daunou*	0.56	0.45	0.54	0.57	0.58	0.54	**0.59**
Instant Runoff	0.55	0.43	0.56	0.55	0.61	0.56	**0.62**
Knockout Voting	0.6	**0.63**	0.58	0.61	0.62	0.59	0.63
Plurality	0.41	0.37	0.85	0.39	0.74	**0.86**	0.75
S. Bucklin	0.55	0.59	0.72	0.58	0.66	**0.73**	0.68
Strict Nanson*	0.79	0.54	0.49	**0.79**	0.79	0.5	0.79
Superior Voting*	0.57	0.45	0.52	0.57	0.68	0.52	**0.68**
T. Alt. GOCHA*	0.79	0.53	0.5	0.79	0.79	0.5	**0.79**
T. Alt. Top Cycle*	0.66	0.48	0.49	0.66	0.67	0.49	**0.67**
Weak Nanson*	0.77	0.55	0.5	0.77	0.77	0.51	**0.78**
Weighted Bucklin	0.54	0.59	0.61	0.55	0.56	**0.61**	0.58

Table C.6: Accuracy on the test set with no Condorcet winner from learning each voting rule in Fishburn's C3 class, averaged over all training distributions for each combination of features. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold. Features are labelled by as M, W, R respectively for $\mathcal{M}^{\text{majority}}$, $\mathcal{M}^{\text{weighted}}$, and $\mathcal{M}^{\text{ranked}}$ features.

C.2 Results by Training Distribution

Target Rule	IC	Mallows	Urn	IAC	SP Conitzer	SP Walsh	E(3CU)	E(3BU)	E(3CG)	E(3BG)
Banks*	0.91	0.93	0.87	0.92	0.43	0.33	0.93	**0.93**	0.92	0.92
Bipartisan Set*	0.89	0.91	0.86	0.9	0.45	0.36	0.91	**0.91**	0.91	0.9
Condorcet*	0.9	0.93	0.87	0.91	0.43	0.34	**0.94**	0.94	0.94	0.92
Copeland*	0.92	0.93	0.87	0.92	0.45	0.35	**0.94**	0.94	0.93	0.93
GOCHA*	0.91	0.93	0.87	0.92	0.43	0.34	**0.93**	0.93	0.92	0.92
Llull*	0.92	0.93	0.87	0.93	0.44	0.34	**0.94**	0.94	0.93	0.93
Slater*	0.91	0.92	0.87	0.92	0.44	0.35	**0.93**	0.93	0.92	0.92
Top Cycle*	0.9	0.93	0.87	0.91	0.42	0.33	**0.94**	0.94	0.93	0.92
Uncovered Set*	0.91	0.93	0.87	0.92	0.43	0.34	**0.93**	0.93	0.93	0.92
Beat Path*	0.89	0.9	0.85	0.9	0.44	0.35	**0.92**	0.92	0.91	0.9
Blacks*	0.89	0.91	0.85	0.9	0.45	0.35	0.91	**0.91**	0.9	0.91
Borda	0.91	0.89	0.85	**0.92**	0.41	0.22	0.91	0.92	0.9	0.9
B-M Faceoff*	0.89	0.91	0.86	0.9	0.44	0.35	0.92	**0.92**	0.92	0.9
C-Global-Borda*	0.91	0.92	0.86	0.91	0.46	0.35	0.92	**0.92**	0.91	0.91
L-T Voting*	0.9	0.91	0.86	0.9	0.45	0.35	**0.92**	0.92	0.91	0.9
Minimax*	0.89	0.91	0.85	0.9	0.45	0.35	**0.92**	0.92	0.91	0.9
Raynaud*	0.89	0.9	0.85	0.89	0.44	0.35	0.91	**0.91**	0.91	0.89
S. Stable Voting*	0.89	0.91	0.86	0.89	0.44	0.35	**0.91**	0.91	0.91	0.9
Split Cycle*	0.89	0.91	0.85	0.9	0.45	0.35	**0.92**	0.92	0.91	0.9
Stable Voting*	0.89	0.91	0.86	0.89	0.45	0.35	**0.91**	0.91	0.91	0.9

Table C.7: Accuracy on the primary test set from learning each voting rule in Fishburn's C1 (red) and C2 (blue) classes, averaged over all feature sets for each training distribution. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold.

Target Rule	IC	Mallows	Urn	IAC	SP Conitzer	SP Walsh	E(3CU)	E(3BU)	E(3CG)	E(3BG)
Anti-Plurality	0.65	**0.76**	0.55	0.65	0.33	0.33	0.72	0.73	0.66	0.61
Baldwin*	0.89	0.91	0.85	0.9	0.44	0.35	**0.91**	0.91	0.91	0.9
Benham*	0.88	0.89	0.85	0.88	0.45	0.35	0.9	**0.9**	0.89	0.88
Bracket Voting	0.86	0.87	0.84	0.87	0.45	0.36	0.88	**0.89**	0.88	0.87
Bucklin	0.69	0.81	0.76	0.74	0.36	0.22	0.83	0.83	0.82	**0.83**
Cond. Plurality*	0.87	0.88	0.85	0.88	0.47	0.36	**0.89**	0.89	0.89	0.88
Coombs	0.8	0.88	0.83	0.82	0.44	0.35	**0.88**	0.88	0.88	0.86
C-Local-Borda*	0.91	0.92	0.86	0.92	0.46	0.36	**0.93**	0.93	0.92	0.92
Daunou*	0.87	0.88	0.85	0.88	0.43	0.34	**0.89**	0.89	0.89	0.88
Instant Runoff	0.82	0.82	0.8	0.82	0.45	0.37	**0.83**	0.83	0.83	0.81
Knockout Voting	0.9	0.91	0.86	0.9	0.46	0.36	0.91	**0.91**	0.9	0.91
Plurality	0.84	0.83	0.75	**0.85**	0.54	0.39	0.85	0.85	0.84	0.84
S. Bucklin	0.65	**0.84**	0.73	0.73	0.42	0.31	0.84	0.84	0.84	0.82
Strict Nanson*	0.9	0.91	0.86	0.9	0.44	0.35	0.92	**0.92**	0.91	0.9
Superior Voting*	0.89	0.9	0.85	0.89	0.46	0.35	0.9	**0.9**	0.9	0.89
T. Alt. GOCHA*	0.89	0.9	0.86	0.9	0.45	0.35	**0.92**	0.91	0.91	0.89
T. Alt. Top Cycle*	0.88	0.89	0.85	0.89	0.45	0.35	0.9	**0.9**	0.9	0.88
Weak Nanson*	0.9	0.91	0.86	0.91	0.45	0.35	0.92	**0.92**	0.91	0.91
Weighted Bucklin	0.83	0.85	0.8	0.83	0.36	0.32	**0.86**	0.86	0.86	0.84

Table C.8: Accuracy on the primary test set from learning each voting rule in Fishburn's C3 class, averaged over all feature sets for each training distribution. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold.

Target Rule	IC	Mallows	Urn	IAC	SP Conitzer	SP Walsh	E(3CU)	E(3BU)	E(3CG)	E(3BG)
Banks*	0.92	0.94	0.87	0.93	0.53	0.44	**0.95**	0.95	0.94	0.94
Bipartisan Set*	0.89	0.92	0.86	0.9	0.55	0.46	**0.92**	0.92	0.92	0.91
Condorcet*	0.91	0.94	0.86	0.92	0.52	0.43	**0.95**	0.95	0.95	0.93
Copeland*	0.93	0.95	0.88	0.94	0.54	0.45	**0.96**	0.95	0.95	0.95
GOCHA*	0.92	0.95	0.88	0.93	0.53	0.45	**0.96**	0.95	0.95	0.95
Llull*	0.93	0.95	0.88	0.94	0.54	0.45	**0.96**	0.95	0.95	0.94
Slater*	0.93	0.95	0.88	0.94	0.54	0.45	**0.95**	0.95	0.95	0.94
Top Cycle*	0.9	0.94	0.86	0.92	0.52	0.43	**0.95**	0.95	0.94	0.94
Uncovered Set*	0.92	0.94	0.87	0.93	0.54	0.45	**0.95**	0.95	0.94	0.93
Beat Path*	0.93	0.94	0.88	0.94	0.54	0.45	**0.95**	0.95	0.95	0.94
Blacks*	0.92	0.94	0.87	0.93	0.55	0.45	**0.94**	0.94	0.93	0.94
Borda	0.93	0.92	0.88	**0.94**	0.5	0.34	0.93	0.94	0.92	0.93
B-M Faceoff*	0.93	0.95	0.88	0.94	0.54	0.45	**0.95**	0.95	0.95	0.94
C-Global-Borda*	0.92	0.94	0.87	0.93	0.55	0.46	**0.95**	0.95	0.94	0.94
L-T Voting*	0.92	0.94	0.88	0.94	0.54	0.46	**0.95**	0.95	0.95	0.94
Minimax*	0.93	0.94	0.88	0.94	0.54	0.45	0.95	**0.95**	0.95	0.94
Raynaud*	0.91	0.93	0.87	0.92	0.53	0.45	**0.94**	0.94	0.94	0.93
S. Stable Voting*	0.92	0.94	0.87	0.93	0.54	0.46	**0.94**	0.94	0.94	0.94
Split Cycle*	0.93	0.94	0.88	0.94	0.54	0.45	**0.95**	0.95	0.95	0.94
Stable Voting*	0.92	0.94	0.87	0.93	0.54	0.46	**0.95**	0.94	0.94	0.94

Table C.9: Accuracy on the secondary test set from learning each voting rule in Fishburn's C1 (red) and C2 (blue) classes, averaged over all feature sets for each training distribution. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold.

Target Rule	IC	Mallows	Urn	IAC	SP Conitzer	SP Walsh	E(3CU)	E(3BU)	E(3CG)	E(3BG)
Anti-Plurality	0.54	0.72	0.6	0.54	0.32	0.32	**0.72**	0.72	0.7	0.57
Baldwin*	0.92	0.94	0.87	0.93	0.54	0.45	**0.95**	0.94	0.94	0.94
Benham*	0.88	0.91	0.86	0.89	0.54	0.45	**0.92**	0.92	0.91	0.9
Bracket Voting	0.88	0.9	0.85	0.89	0.52	0.45	**0.92**	0.92	0.91	0.9
Bucklin	0.65	**0.8**	0.74	0.69	0.37	0.31	0.8	0.79	0.8	0.78
Cond. Plurality*	0.87	0.89	0.85	0.88	0.55	0.45	**0.91**	0.91	0.9	0.9
Coombs	0.79	0.91	0.84	0.82	0.54	0.45	**0.91**	0.91	0.91	0.9
C-Local-Borda*	0.93	0.95	0.88	0.94	0.54	0.46	**0.95**	0.95	0.95	0.95
Daunou*	0.86	0.9	0.85	0.88	0.53	0.45	**0.91**	0.91	0.91	0.9
Instant Runoff	0.79	0.79	0.77	0.8	0.48	0.4	**0.81**	0.81	0.81	0.8
Knockout Voting	0.92	0.94	0.87	0.93	0.55	0.45	**0.94**	0.94	0.94	0.94
Plurality	0.83	0.82	0.73	0.83	0.56	0.42	0.84	**0.84**	0.84	0.82
S. Bucklin	0.62	**0.82**	0.7	0.68	0.38	0.34	0.76	0.76	0.78	0.72
Strict Nanson*	0.93	0.94	0.88	0.93	0.53	0.45	**0.95**	0.95	0.95	0.94
Superior Voting*	0.91	0.92	0.86	0.91	0.54	0.45	**0.93**	0.93	0.93	0.91
T. Alt. GOCHA*	0.92	0.94	0.87	0.93	0.54	0.45	**0.95**	0.95	0.95	0.94
T. Alt. Top Cycle*	0.89	0.91	0.86	0.91	0.53	0.44	**0.93**	0.92	0.92	0.91
Weak Nanson*	0.92	0.94	0.87	0.93	0.54	0.46	**0.94**	0.94	0.94	0.94
Weighted Bucklin	0.86	0.88	0.82	0.86	0.39	0.41	**0.89**	0.89	0.88	0.87

Table C.10: Accuracy on the secondary test set from learning each voting rule in Fishburn's C3 class, averaged over all feature sets for each training distribution. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold.

Target Rule	IC	Mallows	Urn	IAC	SP Conitzer	SP Walsh	E(3CU)	E(3BU)	E(3CG)	E(3BG)
Banks*	0.82	0.79	0.67	**0.83**	0.38	0.3	0.77	0.78	0.75	0.8
Bipartisan Set*	0.57	0.56	0.52	**0.57**	0.32	0.28	0.56	0.56	0.55	0.57
Condorcet*	0.79	0.78	0.74	**0.8**	0.26	0.17	0.79	0.78	0.76	0.76
Copeland*	**0.85**	0.81	0.66	0.84	0.39	0.32	0.82	0.82	0.8	0.84
GOCHA*	**0.83**	0.8	0.68	0.82	0.39	0.31	0.81	0.81	0.79	0.82
Llull*	**0.84**	0.81	0.7	0.84	0.41	0.33	0.82	0.82	0.8	0.83
Slater*	**0.81**	0.78	0.68	0.81	0.41	0.33	0.8	0.79	0.78	0.8
Top Cycle*	0.83	0.8	0.68	**0.84**	0.27	0.21	0.79	0.79	0.75	0.8
Uncovered Set*	0.8	0.78	0.69	**0.81**	0.38	0.3	0.79	0.78	0.76	0.79
Beat Path*	**0.79**	0.77	0.66	0.79	0.41	0.33	0.78	0.78	0.77	0.78
Blacks*	0.74	0.68	0.52	**0.76**	0.37	0.31	0.68	0.68	0.66	0.73
Borda	0.81	0.7	0.62	**0.83**	0.36	0.26	0.76	0.77	0.72	0.77
B-M Faceoff*	**0.79**	0.76	0.66	0.79	0.41	0.33	0.77	0.77	0.77	0.78
C-Global-Borda*	0.74	0.72	0.58	**0.75**	0.38	0.31	0.72	0.72	0.7	0.74
L-T Voting*	**0.78**	0.75	0.65	0.77	0.42	0.33	0.77	0.77	0.76	0.76
Minimax*	0.79	0.76	0.65	**0.79**	0.41	0.33	0.78	0.78	0.77	0.78
Raynaud*	**0.72**	0.7	0.62	0.72	0.4	0.32	0.72	0.72	0.71	0.71
S. Stable Voting*	0.69	0.66	0.55	0.68	0.38	0.33	0.69	0.68	0.67	**0.69**
Split Cycle*	**0.78**	0.76	0.65	0.78	0.41	0.33	0.77	0.77	0.76	0.77
Stable Voting*	0.69	0.66	0.55	0.69	0.38	0.33	0.69	0.68	0.67	**0.69**

Table C.11: Accuracy on the test set with no Condorcet winner from learning each voting rule in Fishburn's C1 (red) and C2 (blue) classes, averaged over all feature sets for each training distribution. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold.

Target Rule	IC	Mallows	Urn	IAC	SP Conitzer	SP Walsh	E(3CU)	E(3BU)	E(3CG)	E(3BG)
Anti-Plurality	0.64	**0.71**	0.47	0.64	0.27	0.27	0.65	0.66	0.59	0.63
Baldwin*	0.74	0.71	0.63	**0.74**	0.4	0.32	0.73	0.73	0.72	0.74
Benham*	0.63	0.6	0.54	**0.63**	0.37	0.29	0.61	0.6	0.6	0.6
Bracket Voting	0.62	0.57	0.55	**0.63**	0.37	0.3	0.62	0.62	0.61	0.62
Bucklin	0.57	0.57	0.43	0.59	0.34	0.22	0.63	0.63	0.59	**0.7**
Cond. Plurality*	0.63	0.52	0.52	**0.64**	0.36	0.26	0.52	0.51	0.51	0.57
Coombs	0.61	0.61	0.5	0.61	0.39	0.32	**0.62**	0.62	0.62	0.61
C-Local-Borda*	**0.82**	0.78	0.64	0.81	0.39	0.32	0.79	0.79	0.78	0.82
Daunou*	**0.62**	0.61	0.54	0.62	0.37	0.29	0.6	0.61	0.6	0.61
Instant Runoff	0.63	0.59	0.54	**0.63**	0.37	0.33	0.62	0.62	0.61	0.6
Knockout Voting	0.71	0.68	0.55	**0.71**	0.38	0.31	0.68	0.69	0.67	0.71
Plurality	0.71	0.67	0.54	**0.72**	0.44	0.35	0.71	0.71	0.7	0.7
S. Bucklin	0.7	0.75	0.56	0.75	0.33	0.26	0.78	0.79	0.73	**0.8**
Strict Nanson*	**0.77**	0.74	0.65	0.77	0.41	0.33	0.76	0.76	0.75	0.76
Superior Voting*	0.64	0.61	0.57	0.64	0.39	0.3	0.64	0.63	0.63	**0.65**
T. Alt. GOCHA*	**0.77**	0.75	0.64	0.77	0.42	0.33	0.76	0.76	0.75	0.76
T. Alt. Top Cycle*	**0.66**	0.64	0.57	0.66	0.4	0.32	0.66	0.65	0.65	0.65
Weak Nanson*	0.76	0.74	0.63	**0.77**	0.4	0.33	0.76	0.75	0.75	0.76
Weighted Bucklin	0.7	0.64	0.53	**0.71**	0.29	0.28	0.66	0.66	0.63	0.68

Table C.12: Accuracy on the test set with no Condorcet winner from learning each voting rule in Fishburn's C3 class, averaged over all feature sets for each training distribution. Positional scoring rules are underlined and Condorcet methods are marked by an asterisk. Highest accuracy values before rounding are bold.

C.3 Distance Between Groups of Rules

C.3.1 Primary Test Set

	C1	C2	C3	Condorcet	Scoring
C1	0.05	-	-	-	-
C2	0.06	0.05	-	-	-
C3	0.12	0.11	0.16	-	-
Condorcet	0.05	0.05	0.11	0.05	-
Scoring	0.30	0.30	0.31	0.29	0.46

Table C.13: Distance between each group of **learned networks** approximating rules in the corresponding group on the primary test set.

	C1	C2	C3	Condorcet	Scoring
C1	0.04	-	-	-	-
C2	0.07	0.06	-	-	-
C3	0.13	0.13	0.18	-	-
Condorcet	0.06	0.06	0.13	0.06	-
Scoring	0.34	0.33	0.35	0.33	0.48

Table C.14: Distance between each group of **existing rules** targeting rules in the corresponding group on the primary test set.

C.3.2 Secondary Test Set

	C1	C2	C3	Condorcet	Scoring
C1	0.05	-	-	-	-
C2	0.05	0.04	-	-	-
C3	0.12	0.12	0.18	-	-
Condorcet	0.05	0.04	0.12	0.04	-
Scoring	0.33	0.33	0.35	0.32	0.52

Table C.15: Distance between each group of **learned networks** approximating rules in the corresponding group on the secondary test set.

	C1	C2	C3	Condorcet	Scoring
C1	0.04	-	-	-	-
C2	0.05	0.04	-	-	-
C3	0.13	0.13	0.20	-	-
Condorcet	0.04	0.04	0.13	0.04	-
Scoring	0.37	0.37	0.39	0.36	0.58

Table C.16: Distance between each group of **existing rules** targeting rules in the corresponding group on the secondary test set.

C.3.3 Test Set with No Condorcet Winners

	C1	C2	C3	Condorcet	Scoring
C1	0.21	-	-	-	-
C2	0.22	0.16	-	-	-
C3	0.28	0.24	0.29	-	-
Condorcet	0.21	0.18	0.25	0.18	-
Scoring	0.46	0.43	0.45	0.43	0.60

Table C.17: Distance between each group of **learned networks** approximating rules in the corresponding group on the test set with no Condorcet winners.

	C1	C2	C3	Condorcet	Scoring
C1	0.32	-	-	-	-
C2	0.36	0.25	-	-	-
C3	0.43	0.38	0.43	-	-
Condorcet	0.34	0.31	0.39	0.32	-
Scoring	0.57	0.51	0.54	0.53	0.67

Table C.18: Distance between each group of **existing rules** targeting rules in the corresponding group on the test set with no Condorcet winners.

C.4 Distance Between Individual Rules

Below we include tables showing the distance between every pair of individual rules. A pair of tables showing distances for learned models and actual rules is included for each dataset. Recall that Fishburn's C1 rules are red, C2 rules are blue, C3 rules are green. Condorcet-consistent rules are marked with an asterisk and positional scoring rules are underlined. Due to the large number of rules we have examined, each table has been made quite dense to fit on the page. To that end, we exclude leading zeros and decimal points. That is, a distance of 0.23 is reported in the following tables as simply 23.

C.4.1 Primary Test Set

	Banks*	Bipartisan Set*	Condorcet*	Copeland*	GOCHA*	Llull*	Slater*	Top Cycle*	Uncovered Set*	Beat Path*	Blacks*	Borda	B-M Faceoff*	C-Global-Borda*	L-T Voting*	Minimax*	Raynaud*	S. Stable Voting*	Split Cycle*	Stable Voting*	Anti-Plurality	Baldwin*	Benham*	Bracket Voting	Bucklin	Cond. Plurality*	Coombs	C-Local-Borda*	Daunou*	Instant Runoff	Knockout Voting	Plurality	S. Bucklin	Strict Nanson*	Superior Voting*	T. Alt. GOCHA*	T. Alt. Top Cycle*	Weak Nanson*	Weighted Bucklin
Banks*	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bipartisan Set*	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Condorcet*	6	7	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Copeland*	4	4	9	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
GOCHA*	3	3	7	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Llull*	3	3	8	2	2	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Slater*	3	4	8	2	2	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Top Cycle*	4	6	4	7	4	5	6	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Uncovered Set*	2	4	6	4	2	3	3	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Beat Path*	4	4	8	2	3	1	2	6	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Blacks*	7	5	10	4	6	5	5	9	7	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Borda	15	14	18	12	14	13	13	17	15	13	10	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
B-M Faceoff*	4	4	9	2	3	2	2	6	4	1	4	13	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
C-Global-Borda*	7	5	10	4	6	5	5	9	7	4	2	11	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
L-T Voting*	4	4	9	2	3	2	2	6	4	2	5	13	1	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Minimax*	4	4	8	2	3	1	2	6	4	1	4	13	1	4	2	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Raynaud*	4	4	9	3	3	2	2	6	4	2	5	13	2	4	1	2	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
S. Stable Voting*	7	4	10	4	6	5	5	9	7	4	3	12	4	2	4	4	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Split Cycle*	4	4	8	2	2	1	1	6	3	1	4	13	1	4	2	1	2	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Stable Voting*	7	5	10	4	6	5	5	9	7	4	3	12	4	3	4	4	4	2	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Anti-Plurality	51	51	52	50	51	50	51	51	51	50	51	53	50	51	50	50	51	51	50	51	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Baldwin*	5	5	9	3	3	2	2	7	4	2	5	13	2	4	1	2	1	4	2	4	51	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Benham*	6	5	10	3	5	4	3	8	6	3	5	14	3	5	3	3	3	4	3	5	52	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bracket Voting	8	6	10	5	7	6	6	9	8	5	6	14	5	5	5	5	5	5	5	5	52	5	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bucklin	20	19	23	18	20	19	19	22	20	19	18	15	19	18	19	19	19	19	19	19	50	19	19	20	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cond. Plurality*	7	6	10	5	6	5	5	9	7	5	6	15	5	5	5	5	5	5	5	5	52	5	4	4	20	0	-	-	-	-	-	-	-	-	-	-	-	-	-
Coombs	7	6	10	4	6	5	5	9	7	4	5	13	4	4	4	4	4	5	4	5	50	4	5	7	18	7	0	-	-	-	-	-	-	-	-	-	-	-	-
C-Local-Borda*	5	5	9	2	4	3	3	7	5	3	4	12	2	3	2	2	3	3	3	3	51	2	3	5	18	5	4	0	-	-	-	-	-	-	-	-	-	-	-
Daunou*	6	5	9	4	5	4	4	8	6	3	5	14	3	5	3	4	4	5	3	5	51	4	3	4	20	3	6	4	0	-	-	-	-	-	-	-	-	-	-
Instant Runoff	14	14	18	13	14	13	13	16	14	12	14	21	12	14	12	12	12	14	12	14	50	12	11	12	23	12	15	13	11	0	-	-	-	-	-	-	-	-	-
Knockout Voting	6	4	10	3	6	5	4	9	7	4	2	11	4	2	4	4	4	2	4	2	51	4	4	5	18	6	5	3	5	14	0	-	-	-	-	-	-	-	-
Plurality	24	23	26	22	23	22	22	25	24	22	24	29	22	23	22	22	22	23	22	23	55	22	21	21	31	20	24	22	21	16	23	0	-	-	-	-	-	-	-
S. Bucklin	29	31	27	31	29	30	30	27	29	31	32	35	31	32	31	31	31	32	31	32	48	31	32	33	30	32	32	32	32	31	32	36	0	-	-	-	-	-	-
Strict Nanson*	5	5	9	3	3	2	2	7	5	2	5	13	2	4	1	2	1	4	2	4	51	1	3	5	19	5	4	2	4	12	4	22	31	0	-	-	-	-	-
Superior Voting*	6	4	9	3	5	4	4	8	6	4	4	13	4	4	3	4	4	3	4	4	51	4	3	3	19	3	5	3	4	12	4	21	32	4	0	-	-	-	-
T. Alt. GOCHA*	5	5	10	3	4	3	2	7	5	2	5	13	2	4	2	2	2	4	2	4	51	2	3	5	19	5	4	2	4	12	4	22	32	2	4	0	-	-	-
T. Alt. Top Cycle*	5	5	9	3	4	3	2	7	5	2	5	14	2	5	2	2	2	4	2	5	51	2	2	4	19	4	5	3	3	11	4	22	31	2	3	2	0	-	-
Weak Nanson*	5	4	9	2	4	3	2	7	5	2	4	13	2	4	2	2	2	3	2	4	51	2	3	5	18	5	4	2	3	13	3	22	31	2	3	2	3	0	-
Weighted Bucklin	13	12	17	11	13	12	12	15	14	12	10	12	11	10	12	12	12	10	12	11	50	12	12	12	14	12	12	11	12	16	10	25	32	12	11	12	12	11	0

Table C.19: Distance between each individual **learned model** evaluated on the primary test set.

	Banks*	Bipartisan Set*	Condorcet*	Copeland*	GOCHA*	Llull*	Slater*	Top Cycle*	Uncovered Set*	Beat Path*	Blacks*	Borda	B-M Faceoff*	C-Global-Borda*	L-T Voting*	Minimax*	Raynaud*	S. Stable Voting*	Split Cycle*	Stable Voting*	Anti-Plurality	Baldwin*	Benham*	Bracket Voting	Bucklin	Cond. Plurality*	Coombs	C-Local-Borda*	Daunou*	Instant Runoff	Knockout Voting	Plurality	S. Bucklin	Strict Nanson*	Superior Voting*	T. Alt. GOCHA*	T. Alt. Top Cycle*	Weak Nanson*	Weighted Bucklin
Banks*	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bipartisan Set*	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Condorcet*	4	7	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Copeland*	5	3	8	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
GOCHA*	3	3	4	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Llull*	3	3	6	1	2	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Slater*	3	4	7	2	3	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Top Cycle*	2	5	2	6	2	4	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Uncovered Set*	1	3	5	4	2	3	3	2	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Beat Path*	5	5	8	4	4	3	4	6	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Blacks*	7	7	10	5	7	6	7	8	7	6	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Borda	16	15	18	14	16	14	15	17	16	15	8	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
B-M Faceoff*	5	5	8	4	4	3	4	6	5	1	5	14	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
C-Global-Borda*	8	6	10	3	7	5	6	9	7	6	2	11	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
L-T Voting*	6	5	8	4	4	3	3	7	5	3	6	15	2	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Minimax*	5	5	8	4	4	3	4	6	5	0	6	15	1	6	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Raynaud*	5	6	8	5	5	4	5	6	4	3	7	15	2	6	1	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
S. Stable Voting*	7	6	10	4	6	5	5	8	7	3	4	13	4	3	5	3	6	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Split Cycle*	5	5	8	4	4	3	4	6	5	0	6	15	1	6	3	0	3	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Stable Voting*	7	6	10	4	6	5	5	8	7	3	4	13	4	3	5	3	6	0	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Anti-Plurality	55	54	55	54	55	54	55	54	55	54	52	50	55	53	55	54	55	52	54	52	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Baldwin*	5	5	8	4	4	3	4	6	5	2	6	15	1	5	1	2	1	4	2	4	55	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Benham*	6	6	9	5	6	5	5	7	6	5	8	16	5	8	5	5	5	6	5	6	56	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bracket Voting	9	8	11	8	9	8	7	9	9	9	8	17	9	8	9	9	9	8	9	8	56	9	6	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bucklin	18	19	20	17	19	18	18	19	18	19	18	21	19	17	19	19	19	19	19	19	53	19	21	21	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cond. Plurality*	9	8	10	8	9	8	8	9	9	8	8	16	8	8	8	8	9	7	8	7	56	8	6	6	21	0	-	-	-	-	-	-	-	-	-	-	-	-	-
Coombs	9	8	11	7	9	7	8	9	8	7	9	15	7	8	8	7	8	7	7	7	51	8	11	12	19	12	0	-	-	-	-	-	-	-	-	-	-	-	-
C-Local-Borda*	6	5	9	1	5	3	3	7	6	5	5	13	5	3	4	5	5	4	5	4	53	5	6	7	17	8	7	0	-	-	-	-	-	-	-	-	-	-	-
Daunou*	8	8	10	7	9	7	7	9	9	8	6	15	8	7	8	8	8	7	8	7	55	8	6	6	20	2	12	7	0	-	-	-	-	-	-	-	-	-	-
Instant Runoff	15	16	18	15	15	14	15	16	16	14	17	24	14	17	14	14	14	15	14	15	59	14	9	15	26	13	19	16	14	0	-	-	-	-	-	-	-	-	-
Knockout Voting	7	6	9	4	6	5	4	8	6	5	2	11	4	2	5	5	6	3	5	3	53	5	7	7	17	8	8	3	7	17	0	-	-	-	-	-	-	-	-
Plurality	29	28	30	28	29	28	28	30	29	28	28	30	28	28	28	28	29	27	28	27	62	28	26	25	32	20	31	28	22	22	28	0	-	-	-	-	-	-	-
S. Bucklin	23	25	24	25	24	25	25	23	24	27	28	31	27	27	28	27	27	28	27	28	55	28	28	30	20	30	27	27	29	31	27	36	0	-	-	-	-	-	-
Strict Nanson*	5	5	8	4	4	3	3	6	5	1	6	15	2	6	2	1	3	3	1	3	54	2	5	8	18	8	8	4	7	14	4	28	27	0	-	-	-	-	-
Superior Voting*	6	6	9	4	6	5	5	7	6	6	6	15	6	5	6	6	7	5	6	5	55	6	4	6	19	6	10	5	6	14	6	27	28	6	0	-	-	-	-
T. Alt. GOCHA*	5	5	8	3	4	3	3	6	5	3	7	15	3	6	3	3	3	4	3	4	55	2	2	7	19	6	9	4	6	11	5	26	27	2	5	0	-	-	-
T. Alt. Top Cycle*	6	6	9	5	6	5	5	7	6	5	7	16	5	7	5	5	5	6	5	6	56	4	0	6	20	5	11	6	6	9	7	25	28	4	5	2	0	-	-
Weak Nanson*	6	6	9	4	5	4	4	7	6	3	4	13	3	4	4	3	5	2	3	2	53	3	6	8	18	8	8	3	7	15	2	28	27	2	6	4	6	0	-
Weighted Bucklin	15	15	17	14	15	14	15	16	15	14	11	13	13	12	14	14	15	13	14	13	55	14	16	15	15	14	16	14	13	21	12	25	27	14	13	15	15	13	0

Table C.20: Distance between each individual **existing rule** evaluated on the primary test set.

C.4.2 Secondary Test Set

	Banks*	Bipartisan Set*	Condorcet*	Copeland*	GOCHA*	Llull*	Slater*	Top Cycle*	Uncovered Set*	Beat Path*	Blacks*	Borda	B-M Faceoff*	C-Global-Borda*	L-T Voting*	Minimax*	Raynaud*	S. Stable Voting*	Split Cycle*	Stable Voting*	Anti-Plurality	Baldwin*	Benham*	Bracket Voting	Bucklin	Cond. Plurality*	Coombs	C-Local-Borda*	Daunou*	Instant Runoff	Knockout Voting	Plurality	S. Bucklin	Strict Nanson*	Superior Voting*	T. Alt. GOCHA*	T. Alt. Top Cycle*	Weak Nanson*	Weighted Bucklin
Banks*	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bipartisan Set*	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Condorcet*	7	8	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Copeland*	3	3	9	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
GOCHA*	2	3	8	2	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Llull*	3	3	8	2	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Slater*	3	3	8	2	1	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Top Cycle*	3	6	4	6	4	5	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Uncovered Set*	2	4	7	4	2	2	2	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Beat Path*	3	3	8	2	1	1	1	5	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Blacks*	5	4	10	3	4	4	4	8	6	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Borda	12	11	16	10	11	11	11	14	12	11	8	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
B-M Faceoff*	3	3	8	2	1	1	1	5	3	1	4	11	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
C-Global-Borda*	5	3	9	3	4	4	4	7	6	4	1	9	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
L-T Voting*	3	3	8	2	2	1	1	5	3	1	4	11	1	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Minimax*	3	3	8	2	1	1	1	5	3	1	4	11	1	3	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Raynaud*	3	3	8	2	2	1	1	5	3	1	4	11	1	4	1	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
S. Stable Voting*	5	3	10	3	4	4	4	8	6	4	2	9	4	2	4	4	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Split Cycle*	3	3	8	2	1	1	1	5	3	1	4	11	1	4	1	1	1	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Stable Voting*	5	3	10	3	4	4	4	8	6	3	2	9	3	2	4	4	4	1	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Anti-Plurality	60	61	62	60	60	60	61	61	61	61	61	61	60	61	60	60	61	61	60	60	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Baldwin*	3	3	8	2	2	1	1	5	3	1	4	11	1	4	1	1	1	4	1	4	61	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Benham*	4	4	9	3	3	3	3	6	4	3	4	11	3	4	3	3	3	4	3	4	61	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bracket Voting	6	4	10	5	5	5	5	8	6	5	5	12	5	5	5	5	5	5	5	5	61	5	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bucklin	21	20	24	19	20	20	20	22	21	20	19	17	20	19	20	20	20	19	20	19	59	20	21	21	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cond. Plurality*	6	5	9	5	5	5	5	8	6	5	6	13	5	5	5	5	5	5	5	5	62	5	4	3	22	0	-	-	-	-	-	-	-	-	-	-	-	-	-
Coombs	6	6	11	4	5	5	5	8	6	4	5	11	4	5	4	4	5	5	4	5	59	5	6	7	18	7	0	-	-	-	-	-	-	-	-	-	-	-	-
C-Local-Borda*	4	3	9	1	3	2	3	6	4	2	3	10	2	2	2	2	2	3	2	3	60	2	3	4	19	5	4	0	-	-	-	-	-	-	-	-	-	-	-
Daunou*	5	5	9	4	4	4	4	6	5	3	5	12	3	5	3	3	4	5	3	5	61	4	3	4	21	3	6	4	0	-	-	-	-	-	-	-	-	-	-
Instant Runoff	14	14	17	14	13	13	13	15	14	13	15	19	13	14	13	13	13	15	13	14	60	13	12	12	25	12	16	14	12	0	-	-	-	-	-	-	-	-	-
Knockout Voting	5	3	10	3	4	4	4	7	5	3	1	9	3	1	4	3	4	2	4	2	61	4	4	5	19	6	5	2	5	14	0	-	-	-	-	-	-	-	-
Plurality	24	24	27	24	24	24	24	25	24	24	25	30	24	25	23	23	23	24	24	24	65	23	22	22	34	20	26	24	22	18	25	0	-	-	-	-	-	-	-
S. Bucklin	33	35	32	35	34	34	35	32	34	35	36	37	35	36	35	34	35	36	35	36	55	35	36	36	31	36	36	35	36	36	36	42	0	-	-	-	-	-	-
Strict Nanson*	3	3	8	2	2	1	1	5	3	1	4	11	1	4	1	1	1	4	1	3	60	1	3	5	20	5	4	2	3	13	4	23	35	0	-	-	-	-	-
Superior Voting*	5	3	9	3	4	4	4	7	5	3	4	11	3	3	3	3	3	3	3	3	61	3	3	3	20	3	5	3	3	13	4	23	36	3	0	-	-	-	-
T. Alt. GOCHA*	4	4	9	2	2	2	1	5	3	1	4	11	1	4	1	1	1	4	1	4	61	1	2	5	20	5	4	2	3	13	4	23	35	1	3	0	-	-	-
T. Alt. Top Cycle*	3	4	8	3	2	2	2	5	3	2	5	12	2	4	2	2	2	4	2	4	61	2	2	4	21	4	5	3	3	12	4	23	35	2	3	2	0	-	-
Weak Nanson*	4	3	9	1	2	2	2	6	4	2	3	10	2	3	2	2	2	3	2	3	60	2	3	4	19	5	4	1	3	13	3	24	35	2	2	2	2	0	-
Weighted Bucklin	12	11	16	10	11	11	11	14	12	11	9	11	11	9	11	11	11	10	11	10	60	11	11	12	16	12	12	10	12	17	9	26	35	11	10	11	11	10	0

Table C.21: Distance between each individual **learned model** evaluated on the secondary test set.

	Banks*	Bipartisan Set*	Condorcet*	Copeland*	GOCHA*	Llull*	Slater*	Top Cycle*	Uncovered Set*	Beat Path*	Blacks*	Borda	B-M Faceoff*	C-Global-Borda*	L-T Voting*	Minimax*	Raynaud*	S. Stable Voting*	Split Cycle*	Stable Voting*	Anti-Plurality	Baldwin*	Benham*	Bracket Voting	Bucklin	Cond. Plurality*	Coombs	C-Local-Borda*	Daunou*	Instant Runoff	Knockout Voting	Plurality	S. Bucklin	Strict Nanson*	Superior Voting*	T. Alt. GOCHA*	T. Alt. Top Cycle*	Weak Nanson*	Weighted Bucklin
Banks*	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bipartisan Set*	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Condorcet*	6	8	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Copeland*	3	4	8	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
GOCHA*	2	4	7	1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Llull*	2	4	7	1	0	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Slater*	2	4	7	1	0	0	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Top Cycle*	3	5	3	5	4	4	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Uncovered Set*	3	3	6	4	2	2	2	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Beat Path*	2	4	7	2	0	0	0	4	2	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Blacks*	5	5	9	3	4	4	4	7	6	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Borda	11	11	14	8	10	10	10	12	11	10	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
B-M Faceoff*	2	4	7	2	0	0	0	5	2	0	4	9	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
C-Global-Borda*	5	5	9	1	3	3	3	7	5	3	1	7	2	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
L-T Voting*	2	4	7	2	0	0	0	5	2	0	4	10	0	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Minimax*	2	4	7	2	0	0	0	4	2	0	4	10	0	3	0	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Raynaud*	3	4	7	3	2	2	1	4	1	2	6	11	2	5	1	2	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
S. Stable Voting*	5	5	9	1	3	3	3	7	5	3	2	8	2	1	3	3	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Split Cycle*	2	4	7	2	0	0	0	4	2	0	4	10	0	3	0	0	2	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Stable Voting*	5	5	9	1	3	3	3	7	5	3	2	8	2	1	3	3	5	0	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Anti-Plurality	65	64	66	65	65	65	65	65	65	65	64	62	65	64	65	65	66	64	65	64	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Baldwin*	2	4	7	2	1	1	0	4	1	1	5	10	1	4	1	1	1	4	1	4	65	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Benham*	5	6	8	5	5	4	4	6	5	5	7	13	5	7	5	5	5	7	5	7	67	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bracket Voting	6	7	10	6	5	5	5	8	7	5	9	14	5	7	5	5	6	7	5	7	67	6	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bucklin	21	20	23	20	20	21	21	22	22	20	19	20	20	19	21	20	22	19	20	19	65	21	22	23	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cond. Plurality*	7	8	8	8	7	7	7	8	7	7	9	14	7	8	7	7	6	9	7	9	68	7	6	5	24	0	-	-	-	-	-	-	-	-	-	-	-	-	-
Coombs	8	8	11	5	6	6	6	9	8	6	5	10	5	5	6	6	7	5	6	5	63	7	9	10	20	11	0	-	-	-	-	-	-	-	-	-	-	-	-
C-Local-Borda*	4	5	8	0	2	2	2	6	4	2	2	8	2	1	2	2	3	1	2	1	65	2	5	6	20	8	5	0	-	-	-	-	-	-	-	-	-	-	-
Daunou*	5	7	8	7	6	5	5	6	5	6	7	13	5	7	5	6	5	8	6	8	66	5	5	6	23	3	10	6	0	-	-	-	-	-	-	-	-	-	-
Instant Runoff	20	21	23	21	20	20	20	21	20	20	22	25	20	22	20	20	20	23	20	23	69	20	15	19	29	21	25	21	20	0	-	-	-	-	-	-	-	-	-
Knockout Voting	5	5	9	2	3	3	3	7	5	3	1	6	2	0	3	3	5	1	3	1	64	4	7	7	19	9	5	1	7	22	0	-	-	-	-	-	-	-	-
Plurality	31	33	33	33	32	32	32	32	32	32	34	37	32	33	31	32	31	33	32	33	74	32	31	28	38	24	35	32	27	23	34	0	-	-	-	-	-	-	-
S. Bucklin	27	27	26	27	28	28	28	25	27	28	28	30	28	28	28	28	28	27	28	27	62	28	29	30	20	31	29	27	30	33	28	41	0	-	-	-	-	-	-
Strict Nanson*	2	4	7	2	0	0	0	4	2	0	4	10	0	3	0	0	1	3	0	3	65	0	5	5	21	7	6	2	5	20	3	32	28	0	-	-	-	-	-
Superior Voting*	4	6	7	4	4	4	4	6	6	4	7	12	4	5	4	4	5	5	4	5	66	5	3	3	23	5	9	5	6	19	6	29	29	4	0	-	-	-	-
T. Alt. GOCHA*	2	5	7	2	1	0	1	5	3	1	5	10	1	3	1	1	2	4	1	4	65	2	3	5	21	7	6	2	5	19	3	32	29	1	4	0	-	-	-
T. Alt. Top Cycle*	3	5	6	5	4	4	3	4	3	4	6	12	4	6	4	4	3	7	4	7	66	4	2	7	21	7	9	5	5	17	6	32	27	4	5	3	0	-	-
Weak Nanson*	3	4	8	1	2	2	1	5	4	2	4	9	2	2	2	2	3	2	2	2	65	2	6	6	20	7	5	1	6	22	2	32	27	2	5	2	5	0	-
Weighted Bucklin	11	12	15	11	10	10	10	13	12	10	9	11	10	9	10	10	12	10	10	10	65	11	13	13	17	14	13	10	13	24	10	32	28	10	11	11	12	11	0

Table C.22: Distance between each individual **existing rule** evaluated on the secondary test set.

C.4.3 Test Set with no Condorcet Winners

	Banks*	Bipartisan Set*	Condorcet*	Copeland*	GOCHA*	Llull*	Slater*	Top Cycle*	Uncovered Set*	Beat Path*	Blacks*	Borda	B-M Faceoff*	C-Global-Borda*	L-T Voting*	Minimax*	Raynaud*	S. Stable Voting*	Split Cycle*	Stable Voting*	Anti-Plurality	Baldwin*	Benham*	Bracket Voting	Bucklin	Cond. Plurality*	Coombs	C-Local-Borda*	Daunou*	Instant Runoff	Knockout Voting	Plurality	S. Bucklin	Strict Nanson*	Superior Voting*	T. Alt. GOCHA*	T. Alt. Top Cycle*	Weak Nanson*	Weighted Bucklin
Banks*	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bipartisan Set*	22	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Condorcet*	39	45	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Copeland*	15	17	46	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
GOCHA*	11	16	42	9	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Llull*	12	16	44	7	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Slater*	12	17	45	7	5	4	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Top Cycle*	15	30	28	23	18	21	21	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Uncovered Set*	10	22	39	16	11	12	12	14	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Beat Path*	13	18	46	7	6	4	5	22	13	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Blacks*	27	20	50	15	21	19	20	34	28	18	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Borda	35	31	57	26	31	30	31	42	37	29	18	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
B-M Faceoff*	14	18	46	7	6	4	5	22	14	4	18	29	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
C-Global-Borda*	28	20	50	15	21	19	20	35	28	18	8	21	18	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
L-T Voting*	15	19	47	8	8	6	6	23	15	5	19	30	5	19	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Minimax*	13	18	46	7	6	4	4	22	13	3	19	29	4	18	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Raynaud*	14	20	46	10	8	7	6	23	14	6	20	31	6	20	6	6	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
S. Stable Voting*	29	20	51	17	22	21	21	37	30	19	12	24	19	11	20	20	21	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Split Cycle*	13	17	45	7	6	4	4	22	13	4	19	30	4	19	5	3	6	20	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Stable Voting*	29	20	51	17	22	21	21	37	30	19	12	23	19	10	19	19	21	8	20	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Anti-Plurality	55	61	68	54	55	55	55	57	57	55	58	59	55	59	56	55	56	61	55	61	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Baldwin*	15	20	47	9	8	6	6	23	15	6	20	31	5	19	5	6	5	20	5	20	56	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Benham*	20	20	48	12	14	12	12	27	20	11	21	32	11	21	11	11	12	21	11	21	58	11	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bracket Voting	34	25	52	23	26	25	25	40	34	24	28	37	24	26	23	24	25	25	24	25	64	24	19	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bucklin	39	37	60	33	37	36	36	45	41	35	31	28	35	32	36	35	37	33	35	33	58	37	37	41	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cond. Plurality*	29	25	49	20	24	22	22	35	30	22	27	37	22	27	22	22	23	27	22	27	62	22	17	16	41	0	-	-	-	-	-	-	-	-	-	-	-	-	-
Coombs	19	22	49	10	14	12	12	27	20	11	17	27	11	17	11	11	12	19	11	19	54	11	14	26	32	24	0	-	-	-	-	-	-	-	-	-	-	-	-
C-Local-Borda*	18	19	48	6	11	10	9	26	19	8	15	26	8	14	8	8	10	15	9	15	56	9	12	22	33	20	9	0	-	-	-	-	-	-	-	-	-	-	-
Daunou*	19	22	48	13	14	12	12	26	20	12	22	33	12	21	12	12	12	22	12	22	58	12	10	21	38	14	16	12	0	-	-	-	-	-	-	-	-	-	-
Instant Runoff	30	32	55	26	25	24	24	37	30	24	33	41	24	33	23	24	23	34	24	34	57	23	19	28	43	25	27	25	21	0	-	-	-	-	-	-	-	-	-
Knockout Voting	27	19	50	15	20	19	19	34	28	17	8	21	17	8	18	17	19	10	18	10	59	18	20	26	32	26	16	13	21	32	0	-	-	-	-	-	-	-	-
Plurality	43	43	61	38	40	39	38	48	44	38	46	52	39	46	39	38	39	46	39	46	66	39	35	35	53	27	42	39	33	29	46	0	-	-	-	-	-	-	-
S. Bucklin	38	49	43	44	41	43	43	32	39	44	51	52	44	52	45	43	44	54	43	53	57	45	47	54	45	52	46	46	46	47	52	55	0	-	-	-	-	-	-
Strict Nanson*	15	20	47	9	8	7	6	24	15	5	20	30	5	19	5	5	5	20	5	19	56	4	11	24	36	22	11	8	12	23	18	39	45	0	-	-	-	-	-
Superior Voting*	27	22	49	16	21	19	20	34	28	18	22	33	18	21	18	18	20	22	19	22	60	19	15	14	37	14	20	15	17	26	22	34	51	18	0	-	-	-	-
T. Alt. GOCHA*	17	21	48	10	9	8	7	25	17	7	20	30	7	19	5	7	7	20	7	20	57	6	11	23	37	22	11	9	12	23	18	39	46	6	18	0	-	-	-
T. Alt. Top Cycle*	16	21	46	11	11	9	8	24	15	9	22	32	8	21	8	9	8	22	8	22	56	8	9	23	37	21	13	11	10	21	20	38	45	8	19	8	0	-	-
Weak Nanson*	17	19	48	7	10	8	7	25	17	7	17	28	7	17	6	7	7	17	7	17	56	7	11	23	34	21	9	6	12	24	16	39	45	6	17	7	9	0	-
Weighted Bucklin	36	30	57	26	31	30	30	43	38	29	21	22	29	22	29	29	31	23	29	23	60	30	30	34	24	33	27	26	31	36	23	47	51	30	29	30	31	28	0

Table C.23: Distance between each individual **learned model** evaluated on the test set with no Condorcet winners.

	Banks*	Bipartisan Set*	Condorcet*	Copeland*	GOCHA*	Llull*	Slater*	Top Cycle*	Uncovered Set*	Beat Path*	Blacks*	Borda	B-M Faceoff*	C-Global-Borda*	L-T Voting*	Minimax*	Raynaud*	S. Stable Voting*	Split Cycle*	Stable Voting*	Anti-Plurality	Baldwin*	Benham*	Bracket Voting	Bucklin	Cond. Plurality*	Coombs	C-Local-Borda*	Daunou*	Instant Runoff	Knockout Voting	Plurality	S. Bucklin	Strict Nanson*	Superior Voting*	T. Alt. GOCHA*	T. Alt. Top Cycle*	Weak Nanson*	Weighted Bucklin
Banks*	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bipartisan Set*	43	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Condorcet*	55	66	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Copeland*	24	38	66	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
GOCHA*	17	36	61	10	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Llull*	18	37	64	8	3	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Slater*	19	37	65	11	7	5	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Top Cycle*	21	48	36	35	29	32	32	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Uncovered Set*	18	35	54	24	17	17	17	20	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Beat Path*	24	42	67	13	8	5	9	36	23	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Blacks*	49	53	77	37	43	42	44	59	53	40	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Borda	50	54	78	37	44	43	45	59	53	41	0	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
B-M Faceoff*	25	43	68	14	10	7	10	36	23	3	38	38	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
C-Global-Borda*	47	46	76	24	34	32	34	58	48	32	13	14	32	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
L-T Voting*	25	42	68	14	10	7	9	37	24	7	45	46	9	35	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Minimax*	24	42	67	13	8	6	9	36	23	0	41	41	3	32	7	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Raynaud*	26	44	65	21	16	14	13	33	18	12	49	50	12	41	8	12	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
S. Stable Voting*	49	44	76	27	34	33	34	59	48	28	25	26	29	14	33	29	37	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Split Cycle*	23	42	66	15	9	7	7	35	21	2	42	42	4	33	9	2	11	29	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Stable Voting*	49	44	76	27	34	33	34	59	48	28	25	26	29	14	33	29	37	0	29	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Anti-Plurality	59	69	73	56	59	58	59	58	61	60	55	55	59	60	61	60	61	63	60	63	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Baldwin*	25	45	65	18	14	12	11	34	20	9	47	48	10	38	7	10	4	34	9	34	60	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Benham*	40	49	70	36	34	33	33	45	35	33	50	51	32	47	32	33	31	50	33	50	69	32	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bracket Voting	55	51	79	43	44	42	44	62	51	42	54	55	43	47	42	42	45	50	44	50	76	45	36	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bucklin	52	57	76	46	49	48	49	57	55	47	41	42	47	42	47	47	49	44	47	44	62	49	60	63	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Cond. Plurality*	55	60	72	52	52	52	52	59	55	50	63	64	51	59	53	50	54	60	50	60	81	54	43	32	69	0	-	-	-	-	-	-	-	-	-	-	-	-	-
Coombs	37	49	72	30	32	30	30	44	37	29	46	47	28	44	30	29	32	40	28	40	57	31	47	52	50	62	0	-	-	-	-	-	-	-	-	-	-	-	-
C-Local-Borda*	29	42	70	5	14	12	14	39	29	11	36	37	12	24	13	11	20	25	13	25	57	16	36	41	44	52	28	0	-	-	-	-	-	-	-	-	-	-	-
Daunou*	38	56	69	35	36	36	36	40	38	33	50	50	33	48	36	34	37	49	33	49	69	37	35	44	58	27	47	34	0	-	-	-	-	-	-	-	-	-	-
Instant Runoff	41	50	70	38	36	36	35	45	37	35	52	53	35	50	35	35	33	52	35	52	70	35	4	40	61	43	50	39	35	0	-	-	-	-	-	-	-	-	-
Knockout Voting	50	49	78	30	37	35	36	59	50	33	10	11	31	8	36	34	41	19	35	19	59	38	47	50	42	61	43	28	50	49	0	-	-	-	-	-	-	-	-
Plurality	56	61	73	53	53	53	53	60	56	52	64	65	52	60	54	51	55	62	51	62	82	55	45	33	70	1	63	53	28	43	62	0	-	-	-	-	-	-	-
S. Bucklin	34	57	46	42	41	42	43	27	37	45	55	55	45	55	45	45	42	58	44	58	65	43	50	59	41	60	49	44	47	50	55	61	0	-	-	-	-	-	-
Strict Nanson*	24	43	67	16	12	10	8	36	22	6	44	45	7	36	6	6	8	31	4	31	61	5	33	43	47	52	29	14	34	35	35	53	45	0	-	-	-	-	-
Superior Voting*	48	52	75	34	40	38	40	58	49	39	49	49	39	42	40	39	44	46	40	46	70	42	29	20	61	33	52	36	42	33	45	35	58	41	0	-	-	-	-
T. Alt. GOCHA*	25	42	69	14	10	9	11	38	24	9	43	43	10	33	8	9	14	34	10	34	62	12	27	42	50	51	33	14	32	29	35	52	45	11	37	0	-	-	-
T. Alt. Top Cycle*	31	49	66	27	25	24	24	35	25	24	49	50	23	45	23	24	20	46	23	46	65	21	14	45	56	49	41	27	30	17	45	50	45	23	39	17	0	-	-
Weak Nanson*	28	44	69	13	15	14	12	39	27	10	40	41	11	31	11	10	14	27	8	27	59	10	35	44	45	52	27	9	35	38	30	53	44	7	40	14	25	0	-
Weighted Bucklin	49	52	76	36	41	40	42	58	51	37	23	24	35	30	41	37	46	32	38	32	64	44	49	49	42	52	47	35	47	51	28	53	55	41	41	40	47	38	0

Table C.24: Distance between each individual **existing rule** evaluated on the test set with no Condorcet winners.

Appendix D

Additional Results of Learning Multi-Winner Voting Rules

This appendix includes performance of trained networks and existing multi-winner voting rules on all individual preference distributions and numbers of alternatives. These results provide a more detailed view of the data discussed in Chapter 5.

- 5 Alternatives, Uniform Ball 3
- 5 Alternatives, Uniform Ball 10
- 5 Alternatives, Gaussian Cube 3
- 5 Alternatives, Gaussian Cube 10
- 5 Alternatives, Uniform Cube 3
- 5 Alternatives, Uniform Cube 10
- 5 Alternatives, Mixed
- 6 Alternatives – All preferences
- 6 Alternatives, Stratified
- 6 Alternatives, Urn
- 6 Alternatives, IC
- 6 Alternatives, IAC
- 6 Alternatives, Identity
- 6 Alternatives, Mallows
- 6 Alternatives, SP Conitzer
- 6 Alternatives, SP Walsh
- 6 Alternatives, Gaussian Ball 3
- 6 Alternatives, Gaussian Ball 10
- 6 Alternatives, Uniform Ball 3
- 6 Alternatives, Uniform Ball 10
- 6 Alternatives, Gaussian Cube 3
- 6 Alternatives, Gaussian Cube 10
- 6 Alternatives, Uniform Cube 3

- 6 Alternatives, Uniform Cube 10
- 6 Alternatives, Mixed
- 7 Alternatives – All preferences
- 7 Alternatives, Stratified
- 7 Alternatives, Urn
- 7 Alternatives, IC
- 7 Alternatives, IAC
- 7 Alternatives, Identity
- 7 Alternatives, Mallows
- 7 Alternatives, SP Conitzer
- 7 Alternatives, SP Walsh
- 7 Alternatives, Gaussian Ball 3
- 7 Alternatives, Gaussian Ball 10
- 7 Alternatives, Uniform Ball 3
- 7 Alternatives, Uniform Ball 10
- 7 Alternatives, Gaussian Cube 3
- 7 Alternatives, Gaussian Cube 10
- 7 Alternatives, Uniform Cube 3
- 7 Alternatives, Uniform Cube 10
- 7 Alternatives, Mixed

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.006**	*.000*	*.000*	.002	**.006**	**.001**	*.000*	*.000*	**0**	.027	*.000*	*.000*	*.000*	.017	.025
Borda	.012	.002	.007	.002	.012	.096	**0**	.016	**0**	.015	*.000*	*.000*	*.000*	.008	.011
EPH	.025	*.000*	.003	**0**	.025	.205	.005	.001	**0**	.044	*.000*	*.000*	*.000*	.030	.038
SNTV	.078	**0**	.109	.132	.078	.484	.016	.111	.038	.044	*.000*	.041	.041	**0**	.003
STV	.034	**0**	.042	.053	.034	.297	.005	.039	**0**	**0**	*.000*	*.000*	*.000*	**0**	*.000*
Bloc	.024	*.000*	.003	**0**	.024	.197	.005	**0**	**0**	.043	*.000*	*.000*	*.000*	.030	.040
CC	.164	.054	.163	.217	.164	.660	.069	.179	.057	.230	**0**	.071	.071	.166	.196
lex-CC	.047	.010	.013	**0**	.047	.362	.005	.033	**0**	.071	**0**	**0**	**0**	.055	.057
seq-CC	.148	.046	.156	.182	.148	.628	.044	.175	.057	.208	**0**	.059	.060	.145	.167
Monroe	.098	.016	.090	.125	.098	.525	.049	.086	**0**	.136	**0**	.001	.001	.111	.130
Greedy M.	.045	.002	.026	.006	.045	.348	.006	.033	**0**	.063	**0**	**0**	**0**	.046	.051
PAV	.029	.001	.003	**0**	.029	.242	.005	.006	**0**	.046	**0**	**0**	**0**	.033	.036
MES	.031	.001	.004	.001	.031	.256	.005	.009	**0**	.049	**0**	**0**	**0**	.035	.037
MAV	.134	.054	.135	.128	.134	.700	.096	.150	**0**	.147	.013	.015	.015	.118	.176
RSD	.089	.023	.085	**0**	.089	.536	.040	.070	**0**	.095	.023	.024	.024	.073	.161
Random	.218	.109	.214	.285	.218	.804	.127	.225	.071	.250	.050	.111	.112	.191	.287

Table D.1: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across all preferences.

D.1 5 Alternatives – All preferences

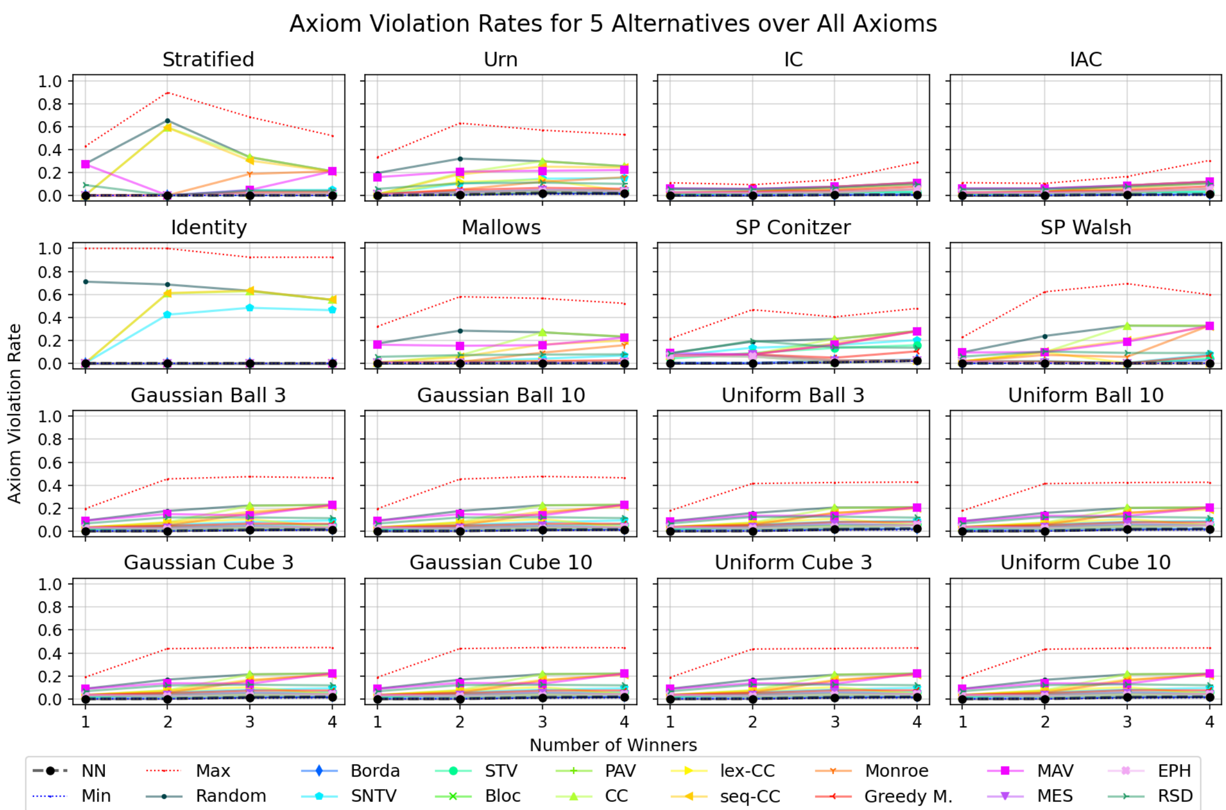


Figure D.1: Axiom violation rates for each rule under each preference distribution for 5 alternatives

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.712	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.127	.712	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.414	.712	.394	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.256	.712	.247	.263	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.197	.712	.230	.355	.303	–	–	–	–	–	–	–	–	–	–	–
PAV	.221	.712	.236	.367	.309	.053	–	–	–	–	–	–	–	–	–	–
CC	.522	.711	.551	.526	.521	.432	.405	–	–	–	–	–	–	–	–	–
lex-CC	.281	.712	.295	.398	.350	.125	.079	.367	–	–	–	–	–	–	–	–
seq-CC	.525	.712	.507	.409	.507	.431	.406	.588	.398	–	–	–	–	–	–	–
Monroe	.425	.711	.454	.455	.427	.332	.310	.124	.318	.542	–	–	–	–	–	–
Greedy M.	.307	.711	.301	.363	.350	.203	.173	.445	.196	.343	.367	–	–	–	–	–
MAV	.574	.712	.606	.679	.606	.589	.574	.387	.539	.755	.363	.606	–	–	–	–
MES	.240	.712	.239	.344	.316	.105	.067	.440	.118	.367	.346	.140	.594	–	–	–
EPH	.201	.712	.229	.356	.303	.011	.045	.429	.119	.426	.329	.197	.588	.097	–	–
RSD	.474	.711	.478	.559	.505	.456	.459	.621	.477	.600	.549	.474	.621	.461	.457	–
Min	.018	.712	.127	.409	.249	.207	.230	.521	.289	.527	.424	.312	.573	.248	.211	.476
Max	.989	.712	.978	.912	.951	.966	.950	.823	.925	.799	.881	.925	.833	.944	.963	.866

Table D.2: Difference between rules for 5 alternatives with $1 \leq k < 5$ averaged over all preference distributions.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Borda	.002	**0**	**0**	**0**	.002	.025	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
EPH	.006	**0**	*.000*	**0**	.006	.082	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
SNTV	.023	**0**	.004	**0**	.023	.295	**0**	.005	**0**	**0**	**0**	**0**	**0**	**0**	**0**
STV	.006	**0**	*.000*	**0**	.006	.083	**0**	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Bloc	.006	**0**	*.000*	**0**	.006	.082	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
CC	.284	.234	.007	.502	.284	.622	.071	.231	.225	.502	**0**	.251	.251	.401	.401
lex-CC	.006	**0**	*.000*	**0**	.006	.082	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
seq-CC	.276	.220	.006	.498	.276	.620	.044	.231	.224	.498	**0**	.224	.224	.396	.396
Monroe	.099	.091	.003	.205	.099	.327	.046	.003	**0**	.205	**0**	.001	.001	.204	.204
Greedy M.	.016	**0**	.001	**0**	.016	.202	**0**	.001	**0**	**0**	**0**	**0**	**0**	**0**	**0**
PAV	.006	**0**	*.000*	**0**	.006	.082	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MES	.006	**0**	*.000*	**0**	.006	.082	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MAV	.132	.222	.005	.249	.132	.527	.082	.183	**0**	.101	**0**	**0**	**0**	.101	.249
RSD	.043	.097	.003	**0**	.043	.362	**0**	.101	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Random	.368	.445	.008	.652	.368	.817	.127	.409	.225	.501	.076	.250	.250	.435	.585

Table D.3: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Stratified preferences.

D.1.1 5 Alternatives, Stratified

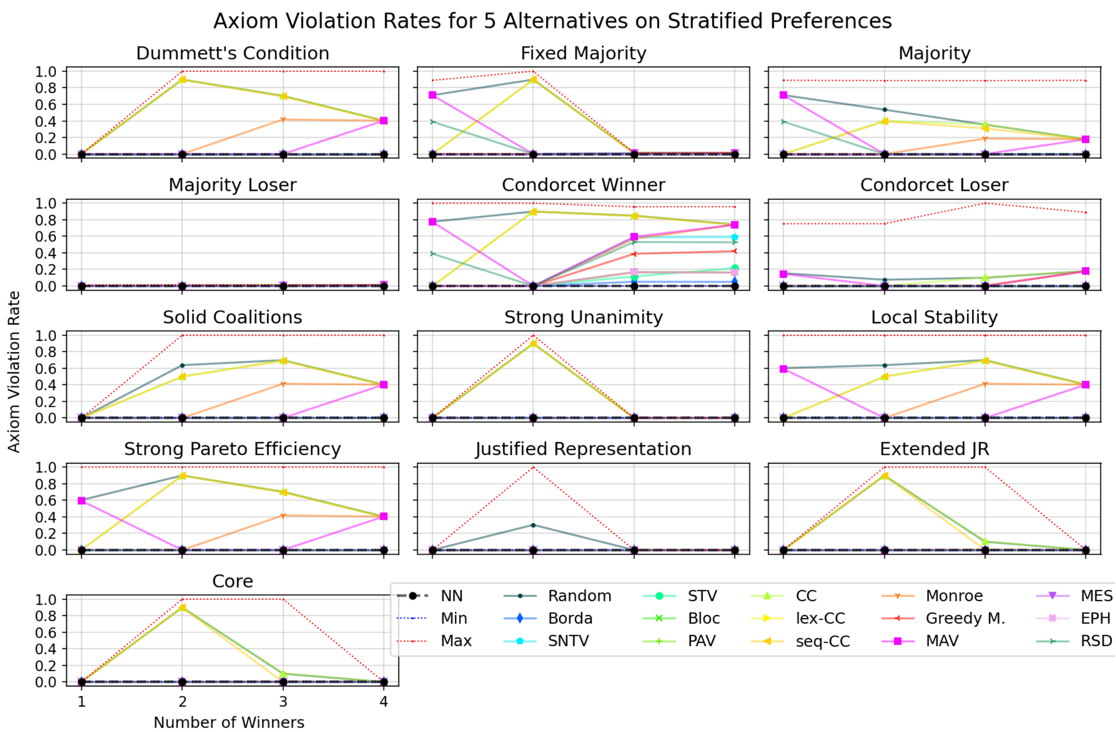


Figure D.2: Axiom violation rate for each axiom on Stratified preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.902	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.114	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.326	.900	.242	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.102	.902	.150	.306	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.091	.901	.144	.319	.149	–	–	–	–	–	–	–	–	–	–	–
PAV	.091	.901	.144	.319	.149	.000	–	–	–	–	–	–	–	–	–	–
CC	.446	.899	.524	.630	.463	.454	.454	–	–	–	–	–	–	–	–	–
lex-CC	.091	.901	.144	.319	.149	.000	.000	.454	–	–	–	–	–	–	–	–
seq-CC	.526	.898	.448	.298	.509	.519	.519	.780	.519	–	–	–	–	–	–	–
Monroe	.269	.901	.343	.449	.282	.270	.270	.198	.270	.606	–	–	–	–	–	–
Greedy M.	.244	.898	.181	.227	.258	.230	.230	.534	.230	.437	.363	–	–	–	–	–
MAV	.651	.900	.698	.785	.670	.661	.661	.601	.661	.984	.486	.709	–	–	–	–
MES	.155	.900	.086	.236	.204	.096	.096	.528	.096	.444	.344	.170	.704	–	–	–
EPH	.091	.901	.144	.319	.149	.000	.000	.454	.000	.519	.270	.230	.661	.096	–	–
RSD	.450	.899	.449	.472	.457	.447	.447	.686	.447	.675	.534	.460	.682	.446	.447	–
Min	.000	.902	.114	.326	.102	.091	.091	.446	.091	.526	.269	.244	.651	.155	.091	.450
Max	1.250	.898	1.250	1.250	1.250	1.250	1.250	1.047	1.250	1.024	1.152	1.250	1.132	1.250	1.250	1.250

Table D.4: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Stratified preferences.

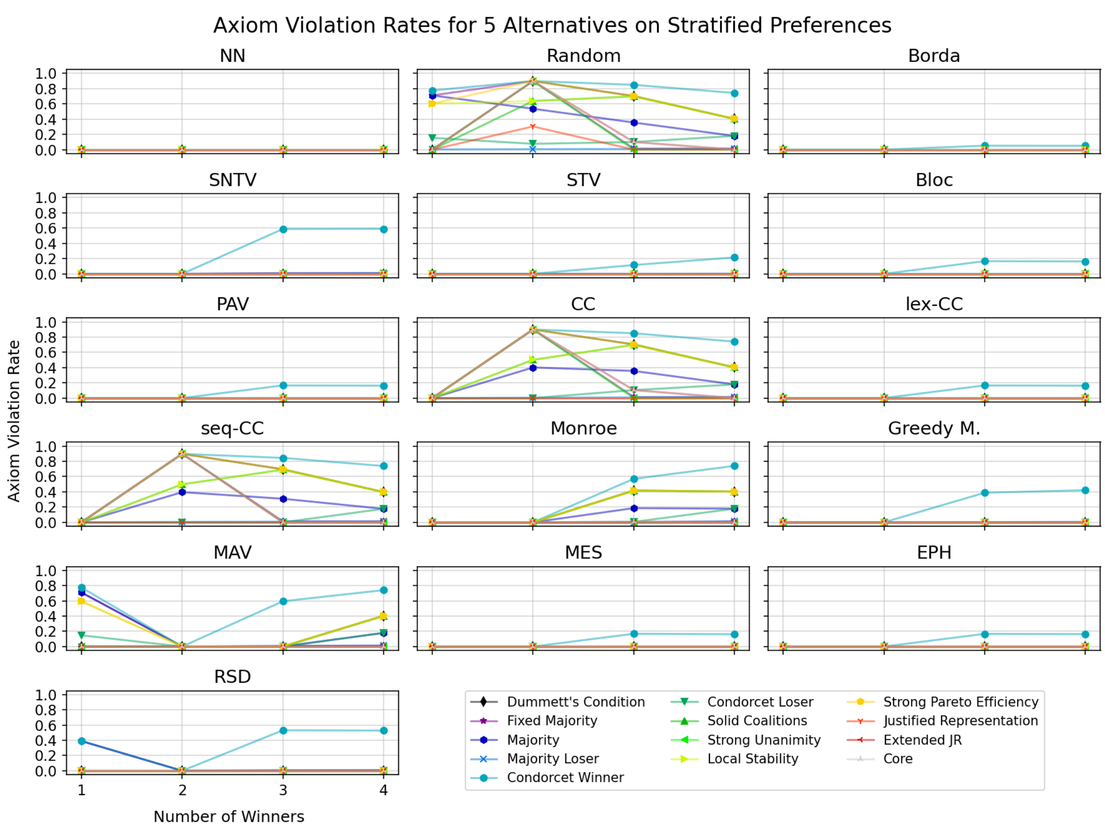


Figure D.3: Axiom violation rate for each rule on Stratified preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.012**	.001	.002	.014	**.012**	**.002**	*.000*	*.000*	**0**	.044	.002	.002	.002	.035	.049
Borda	.022	.016	.016	.018	.022	.120	**0**	.054	**0**	.022	*.000*	*.000*	*.000*	.016	.020
EPH	.025	.004	.004	**0**	.025	.115	.003	.005	**0**	.070	*.000*	*.000*	*.000*	.058	.065
SNTV	.102	**0**	.201	.155	.102	.483	.003	.285	.026	.093	*.000*	.038	.040	**0**	*.000*
STV	.030	**0**	.058	.046	.030	.212	.001	.077	**0**	**0**	*.000*	*.000*	*.000*	**0**	*.000*
Bloc	.023	.002	**.002**	**0**	.023	.105	.003	**0**	**0**	.068	*.000*	*.000*	*.000*	.055	.063
CC	.189	.140	.233	.186	.189	.575	.061	.325	.032	.311	**0**	.059	.063	.229	.245
lex-CC	.075	.063	.061	**0**	.075	.316	.003	.138	**0**	.148	**0**	**0**	**0**	.124	.129
seq-CC	.172	.126	.218	.146	.172	.548	.049	.312	.031	.286	**0**	.043	.047	.207	.219
Monroe	.084	.047	.091	.057	.084	.341	.033	.112	**0**	.151	**0**	*.000*	*.000*	.124	.138
Greedy M.	.045	.021	.028	.004	.045	.212	.007	.047	**0**	.095	**0**	**0**	**0**	.080	.088
PAV	.029	.008	.008	**0**	.029	.139	.003	.014	**0**	.075	**0**	**0**	**0**	.063	.069
MES	.031	.010	.010	*.000*	.031	.149	.003	.018	**0**	.077	**0**	**0**	**0**	.064	.070
MAV	.201	.226	.223	.174	.201	.695	.092	.395	**0**	.273	.014	.018	.018	.211	.269
RSD	.093	.087	.086	**0**	.093	.373	.042	.127	**0**	.150	.018	.024	.024	.125	.159
Random	.267	.281	.280	.297	.267	.750	.122	.438	.041	.348	.062	.117	.125	.269	.339

Table D.5: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Urn preferences.

D.1.2 5 Alternatives, Urn

Figure D.4: Axiom violation rate for each axiom on Urn preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.900	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.217	.900	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.472	.901	.467	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.254	.901	.260	.334	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.203	.899	.263	.413	.292	–	–	–	–	–	–	–	–	–	–	–
PAV	.221	.899	.270	.414	.295	.037	–	–	–	–	–	–	–	–	–	–
CC	.519	.900	.580	.522	.528	.442	.425	–	–	–	–	–	–	–	–	–
lex-CC	.311	.899	.351	.429	.359	.147	.115	.375	–	–	–	–	–	–	–	–
seq-CC	.541	.900	.544	.395	.511	.432	.416	.590	.397	–	–	–	–	–	–	–
Monroe	.354	.900	.415	.458	.371	.268	.253	.196	.301	.522	–	–	–	–	–	–
Greedy M.	.277	.900	.309	.401	.317	.150	.132	.459	.194	.366	.300	–	–	–	–	–
MAV	.739	.900	.813	.878	.810	.802	.794	.520	.745	.957	.591	.822	–	–	–	–
MES	.232	.901	.274	.396	.299	.075	.051	.453	.141	.385	.282	.105	.811	–	–	–
EPH	.209	.899	.264	.413	.292	.009	.030	.439	.141	.428	.265	.145	.800	.068	–	–
RSD	.489	.901	.502	.637	.519	.457	.458	.670	.504	.647	.538	.473	.816	.460	.457	–
Min	.033	.900	.216	.465	.247	.209	.225	.518	.313	.539	.352	.279	.741	.236	.214	.489
Max	1.223	.900	1.210	1.129	1.194	1.204	1.194	1.056	1.142	1.035	1.144	1.168	1.005	1.188	1.201	1.094

Table D.6: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Urn preferences.

Figure D.5: Axiom violation rate for each rule on Urn preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.003**	**0**	**0**	**0**	**.003**	*.000*	*.000*	**0**	**0**	.016	**0**	**0**	**0**	.006	.015
Borda	.009	**0**	**0**	**0**	.009	.106	**0**	**0**	**0**	.007	**0**	**0**	**0**	.002	.003
EPH	.032	**0**	**0**	**0**	.032	.304	.004	**0**	**0**	.046	**0**	**0**	**0**	.025	.033
SNTV	.030	**0**	**0**	**0**	.030	.386	.004	**0**	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
STV	.022	**0**	**0**	**0**	.022	.285	.002	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Bloc	.032	**0**	**0**	**0**	.032	.302	.004	**0**	**0**	.046	**0**	**0**	**0**	.025	.033
CC	.061	**0**	*.000*	**0**	.061	.523	.039	**0**	**0**	.087	**0**	**0**	**0**	.052	.090
lex-CC	.036	**0**	**0**	**0**	.036	.360	.004	**0**	**0**	.047	**0**	**0**	**0**	.026	.033
seq-CC	.053	**0**	**0**	**0**	.053	.489	.020	**0**	**0**	.074	**0**	**0**	**0**	.044	.063
Monroe	.061	**0**	*.000*	**0**	.061	.523	.039	**0**	**0**	.087	**0**	**0**	**0**	.052	.090
Greedy M.	.044	**0**	**0**	**0**	.044	.426	.010	**0**	**0**	.060	**0**	**0**	**0**	.034	.048
PAV	.033	**0**	**0**	**0**	.033	.317	.004	**0**	**0**	.046	**0**	**0**	**0**	.025	.032
MES	.034	**0**	**0**	**0**	.034	.336	.004	**0**	**0**	.046	**0**	**0**	**0**	.025	.032
MAV	.076	**0**	*.000*	**0**	.076	.673	.073	**0**	**0**	.087	**0**	**0**	**0**	.052	.099
RSD	.068	**0**	*.000*	**0**	.068	.622	.052	**0**	**0**	.077	**0**	**0**	**0**	.046	.085
Random	.076	**0**	*.000*	*.000*	.076	.672	.077	**0**	**0**	.088	*.000*	*.000*	*.000*	.054	.100

Table D.7: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across IC preferences.

D.1.3 5 Alternatives, IC

Figure D.6: Axiom violation rate for each axiom on IC preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.902	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.316	.902	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.579	.900	.501	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.410	.901	.404	.344	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.494	.900	.445	.466	.554	–	–	–	–	–	–	–	–	–	–	–
PAV	.505	.901	.454	.467	.555	.054	–	–	–	–	–	–	–	–	–	–
CC	.606	.900	.657	.554	.633	.394	.371	–	–	–	–	–	–	–	–	–
lex-CC	.529	.901	.481	.478	.566	.106	.057	.352	–	–	–	–	–	–	–	–
seq-CC	.689	.902	.599	.415	.626	.460	.430	.590	.415	–	–	–	–	–	–	–
Monroe	.606	.900	.657	.554	.633	.394	.371	.000	.352	.590	–	–	–	–	–	–
Greedy M.	.610	.901	.544	.410	.600	.364	.340	.496	.335	.280	.496	–	–	–	–	–
MAV	.735	.899	.914	.945	.871	.854	.863	.523	.869	1.046	.523	.950	–	–	–	–
MES	.535	.902	.462	.388	.568	.208	.165	.473	.165	.306	.473	.239	.925	–	–	–
EPH	.496	.900	.446	.465	.553	.008	.050	.393	.103	.457	.393	.362	.858	.204	–	–
RSD	.835	.900	.830	.838	.842	.810	.811	.848	.813	.837	.848	.823	.901	.811	.810	–
Min	.015	.902	.317	.575	.405	.499	.511	.608	.534	.689	.608	.612	.737	.540	.502	.836
Max	1.178	.899	1.118	1.063	1.115	1.095	1.087	1.082	1.074	.970	1.082	1.029	1.077	1.062	1.093	.944

Table D.8: Difference between rules for 5 alternatives with $1 \leq k < 5$ on IC preferences.

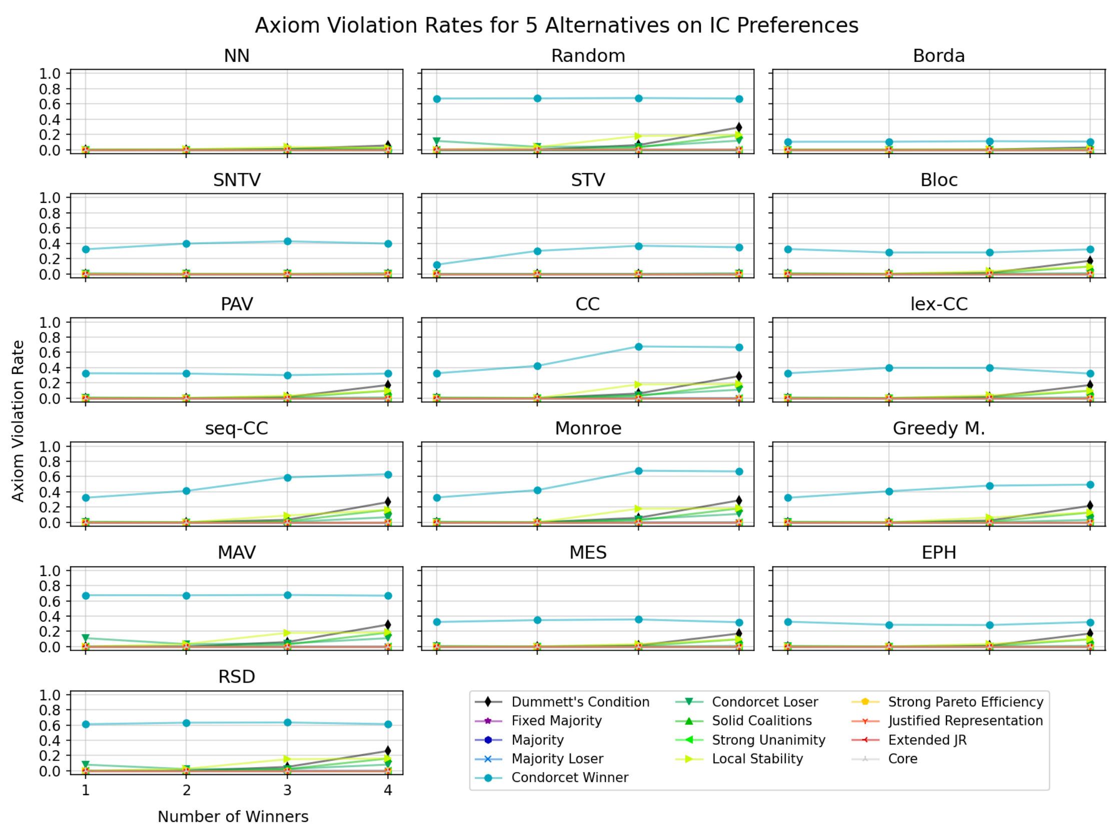


Figure D.7: Axiom violation rate for each rule on IC preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.004**	**0**	**0**	**0**	**.004**	*.000*	*.000*	**0**	**0**	.021	**0**	**0**	**0**	.010	.023
Borda	.010	**0**	**0**	**0**	.010	.112	**0**	**0**	**0**	.009	**0**	**0**	**0**	.003	.004
EPH	.032	**0**	**0**	**0**	.032	.288	.004	**0**	**0**	.054	**0**	**0**	**0**	.032	.043
SNTV	.030	**0**	*.000*	**0**	.030	.380	.005	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
STV	.022	**0**	*.000*	*.000*	.022	.284	.002	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
Bloc	.032	**0**	**0**	**0**	.032	.287	.004	**0**	**0**	.054	**0**	**0**	**0**	.032	.043
CC	.066	*.000*	*.000*	*.000*	.066	.515	.042	*.000*	**0**	.107	**0**	**0**	**0**	.074	.118
lex-CC	.038	**0**	**0**	**0**	.038	.355	.004	**0**	**0**	.057	**0**	**0**	**0**	.034	.045
seq-CC	.057	*.000*	*.000*	**0**	.057	.483	.021	*.000*	**0**	.090	**0**	**0**	**0**	.060	.085
Monroe	.066	*.000*	*.000*	*.000*	.066	.514	.042	*.000*	**0**	.107	**0**	**0**	**0**	.073	.118
Greedy M.	.046	**0**	**0**	**0**	.046	.410	.010	**0**	**0**	.069	**0**	**0**	**0**	.043	.061
PAV	.034	**0**	**0**	**0**	.034	.305	.004	**0**	**0**	.054	**0**	**0**	**0**	.032	.042
MES	.035	**0**	**0**	**0**	.035	.324	.004	**0**	**0**	.055	**0**	**0**	**0**	.033	.044
MAV	.082	*.000*	*.000*	*.000*	.082	.660	.083	*.000*	**0**	.107	*.000*	*.000*	*.000*	.074	.139
RSD	.071	*.000*	*.000*	**0**	.071	.605	.053	*.000*	**0**	.093	*.000*	*.000*	*.000*	.062	.114
Random	.082	*.000*	*.000*	*.000*	.082	.662	.082	*.000*	**0**	.108	*.000*	*.000*	*.000*	.073	.136

Table D.9: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across IAC preferences.

D.1.4 5 Alternatives, IAC

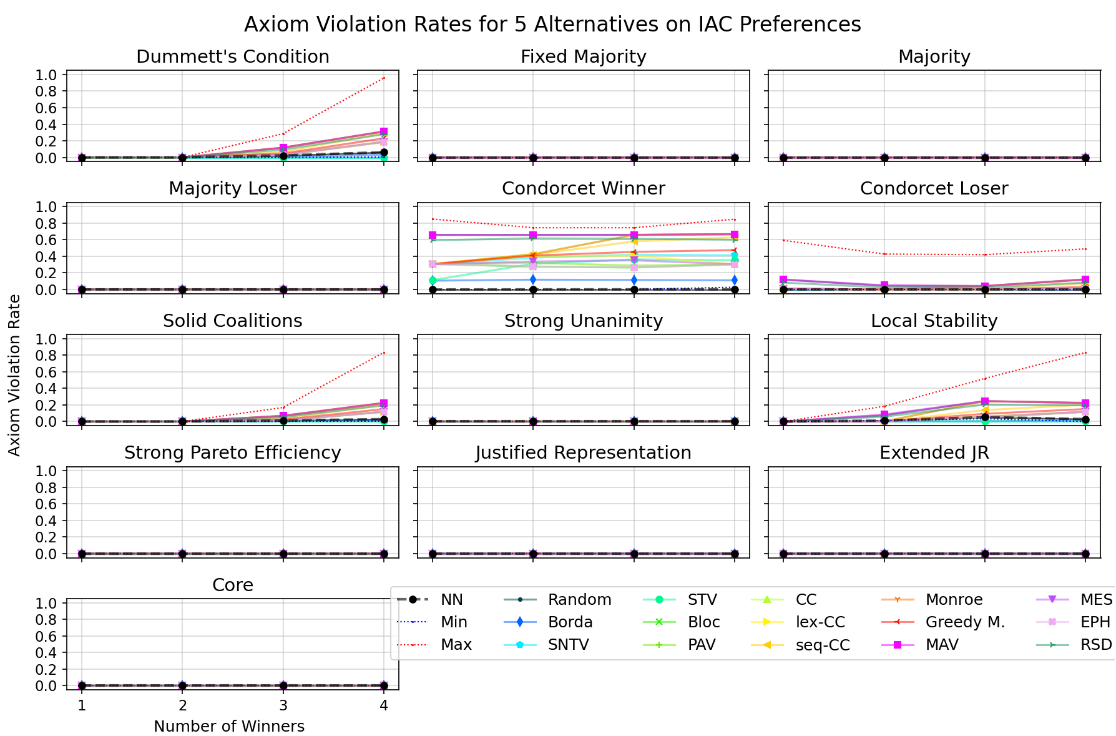


Figure D.8: Axiom violation rate for each axiom on IAC preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.900	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.323	.900	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.572	.902	.499	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.415	.898	.406	.332	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.485	.902	.439	.452	.546	–	–	–	–	–	–	–	–	–	–	–
PAV	.495	.903	.448	.453	.545	.055	–	–	–	–	–	–	–	–	–	–
CC	.601	.902	.655	.539	.629	.402	.379	–	–	–	–	–	–	–	–	–
lex-CC	.523	.903	.480	.466	.559	.116	.067	.354	–	–	–	–	–	–	–	–
seq-CC	.683	.903	.598	.420	.619	.448	.417	.577	.400	–	–	–	–	–	–	–
Monroe	.600	.902	.655	.539	.629	.401	.378	.002	.355	.578	–	–	–	–	–	–
Greedy M.	.595	.902	.532	.409	.587	.339	.313	.482	.310	.284	.482	–	–	–	–	–
MAV	.727	.902	.911	.938	.875	.863	.870	.527	.874	1.044	.527	.948	–	–	–	–
MES	.523	.902	.457	.389	.557	.189	.145	.464	.153	.312	.463	.227	.923	–	–	–
EPH	.486	.903	.439	.451	.545	.008	.051	.401	.113	.445	.400	.336	.865	.185	–	–
RSD	.825	.901	.816	.829	.830	.795	.796	.840	.801	.825	.840	.808	.899	.797	.795	–
Min	.019	.900	.324	.566	.409	.492	.502	.604	.529	.683	.603	.598	.730	.529	.493	.826
Max	1.184	.900	1.125	1.076	1.121	1.099	1.091	1.076	1.077	.977	1.076	1.038	1.058	1.069	1.097	.951

Table D.10: Difference between rules for 5 alternatives with $1 \leq k < 5$ on IAC preferences.

Figure D.9: Axiom violation rate for each rule on IAC preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Borda	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
EPH	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
SNTV	.342	**0**	.375	.582	.342	.582	**0**	.582	.582	.582	**0**	.582	.582	**0**	**0**
STV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Bloc	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
CC	.447	.259	.425	.648	.447	.648	.074	.648	.648	.648	**0**	.648	.648	.259	.259
lex-CC	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
seq-CC	.449	.266	.427	.649	.449	.649	.076	.649	.649	.649	**0**	.649	.649	.266	.266
Monroe	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Greedy M.	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
PAV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MES	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MAV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
RSD	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Random	.645	.500	.502	.851	.645	.851	.150	.851	.851	.851	.275	.851	.851	.500	.500

Table D.11: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Identity preferences.

D.1.5 5 Alternatives, Identity

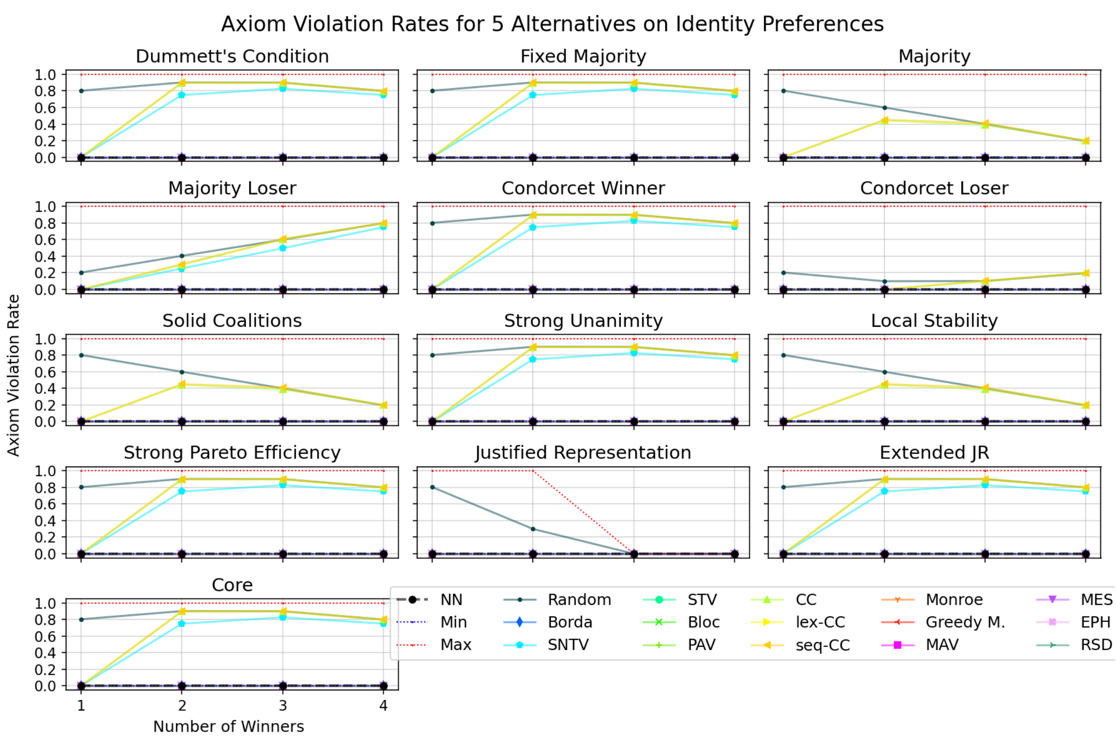


Figure D.10: Axiom violation rate for each axiom on Identity preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.000	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.406	.902	.406	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.000	.901	.000	.406	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.000	.901	.000	.406	.000	–	–	–	–	–	–	–	–	–	–	–
PAV	.000	.901	.000	.406	.000	.000	–	–	–	–	–	–	–	–	–	–
CC	.461	.902	.461	.674	.461	.461	.461	–	–	–	–	–	–	–	–	–
lex-CC	.000	.901	.000	.406	.000	.000	.000	.461	–	–	–	–	–	–	–	–
seq-CC	.463	.902	.463	.158	.463	.463	.463	.724	.463	–	–	–	–	–	–	–
Monroe	.000	.901	.000	.406	.000	.000	.000	.461	.000	.463	–	–	–	–	–	–
Greedy M.	.000	.901	.000	.406	.000	.000	.000	.461	.000	.463	.000	–	–	–	–	–
MAV	.000	.901	.000	.406	.000	.000	.000	.461	.000	.463	.000	.000	–	–	–	–
MES	.000	.901	.000	.406	.000	.000	.000	.461	.000	.463	.000	.000	.000	–	–	–
EPH	.000	.901	.000	.406	.000	.000	.000	.461	.000	.463	.000	.000	.000	.000	–	–
RSD	.000	.901	.000	.406	.000	.000	.000	.461	.000	.463	.000	.000	.000	.000	.000	–
Min	.000	.901	.000	.406	.000	.000	.000	.461	.000	.463	.000	.000	.000	.000	.000	.000
Max	1.250	.899	1.250	1.124	1.250	1.250	1.250	1.026	1.250	1.023	1.250	1.250	1.250	1.250	1.250	1.250

Table D.12: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Identity preferences.

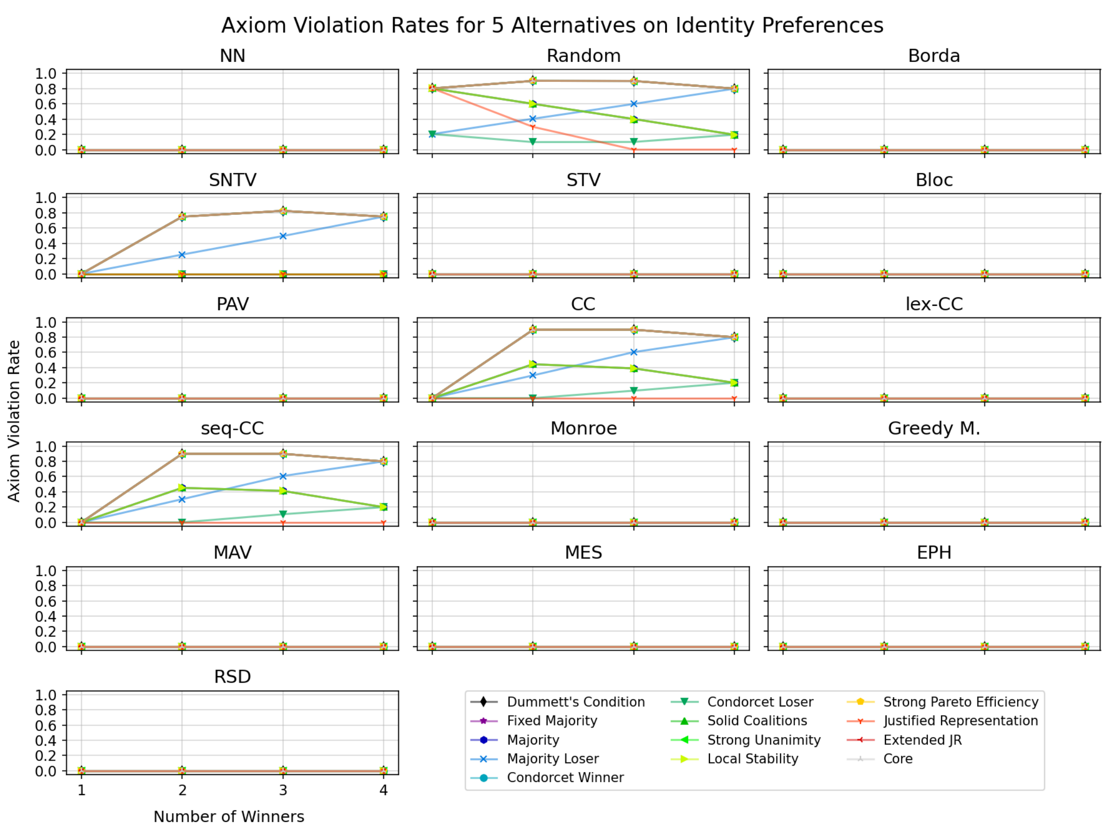


Figure D.11: Axiom violation rate for each rule on Identity preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.001**	**0**	**0**	**0**	**.001**	**.001**	*.000*	**0**	**0**	.002	**0**	**0**	**0**	.001	.003
Borda	.003	*.000*	*.000*	**0**	.003	.032	**0**	*.000*	**0**	.001	**0**	**0**	**0**	*.000*	*.000*
EPH	.007	**0**	**0**	**0**	.007	.074	*.000*	**0**	**0**	.006	**0**	**0**	**0**	.004	.005
SNTV	.032	**0**	.063	.020	.032	.212	.001	.083	.006	.016	**0**	.007	.007	**0**	*.000*
STV	.005	**0**	*.000*	**0**	.005	.068	*.000*	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
Bloc	.007	**0**	**0**	**0**	.007	.074	*.000*	**0**	**0**	.006	**0**	**0**	**0**	.004	.005
CC	.144	.089	.184	.112	.144	.527	.070	.240	.009	.225	**0**	.038	.042	.159	.177
lex-CC	.013	.001	*.000*	**0**	.013	.133	*.000*	.015	**0**	.008	**0**	**0**	**0**	.004	.006
seq-CC	.105	.033	.169	.082	.105	.489	.027	.230	.009	.140	**0**	.022	.025	.068	.075
Monroe	.067	.027	.060	.014	.067	.337	.043	.072	**0**	.115	**0**	*.000*	*.000*	.091	.107
Greedy M.	.015	**0**	.005	**0**	.015	.156	.002	.006	**0**	.012	**0**	**0**	**0**	.008	.010
PAV	.007	**0**	**0**	**0**	.007	.078	*.000*	**0**	**0**	.006	**0**	**0**	**0**	.004	.004
MES	.008	**0**	**0**	**0**	.008	.084	*.000*	**0**	**0**	.006	**0**	**0**	**0**	.004	.005
MAV	.174	.178	.175	.106	.174	.739	.119	.309	**0**	.198	.015	.021	.021	.151	.225
RSD	.070	.049	.050	**0**	.070	.466	.027	.115	**0**	.066	.002	.002	.002	.051	.080
Random	.239	.255	.255	.192	.239	.824	.138	.382	.012	.296	.045	.082	.089	.229	.306

Table D.13: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Mallows preferences.

D.1.6 5 Alternatives, Mallows

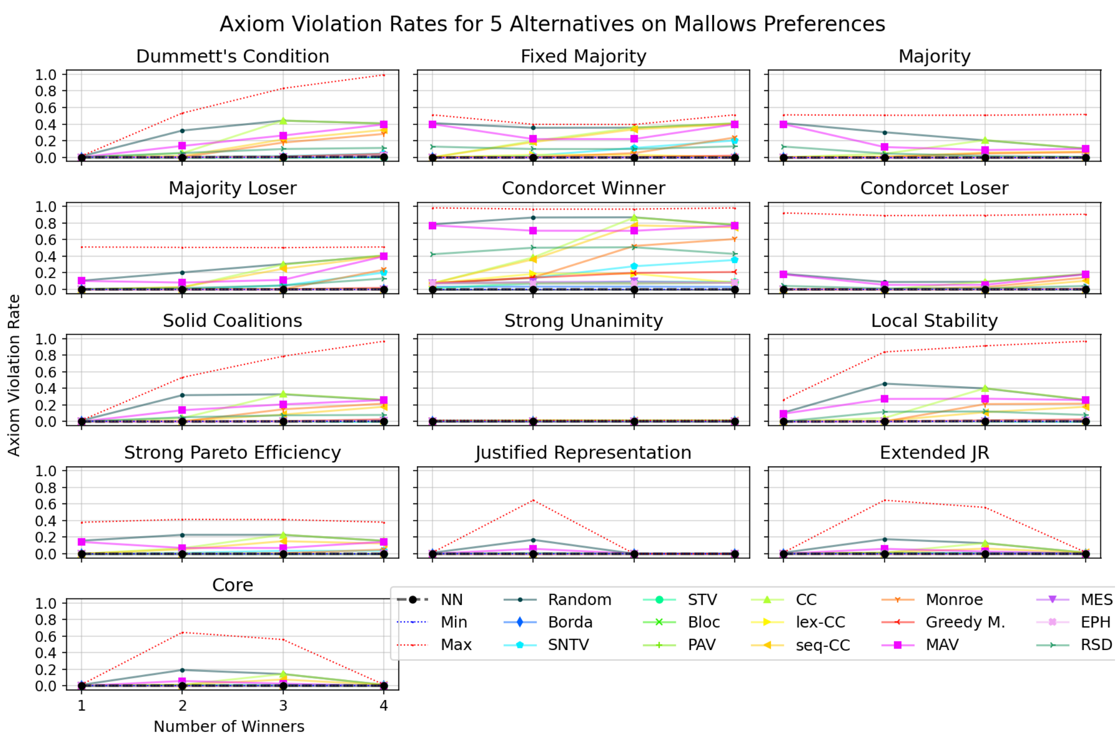


Figure D.12: Axiom violation rate for each axiom on Mallows preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.903	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.068	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.215	.903	.200	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.078	.903	.088	.167	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.106	.901	.100	.196	.128	–	–	–	–	–	–	–	–	–	–	–
PAV	.109	.901	.101	.197	.128	.012	–	–	–	–	–	–	–	–	–	–
CC	.444	.902	.453	.458	.449	.396	.390	–	–	–	–	–	–	–	–	–
lex-CC	.136	.900	.130	.214	.150	.047	.036	.375	–	–	–	–	–	–	–	–
seq-CC	.420	.899	.404	.329	.409	.377	.370	.552	.356	–	–	–	–	–	–	–
Monroe	.304	.901	.313	.346	.309	.255	.250	.146	.252	.473	–	–	–	–	–	–
Greedy M.	.168	.900	.158	.204	.172	.120	.114	.422	.125	.332	.284	–	–	–	–	–
MAV	.801	.901	.829	.865	.820	.816	.817	.529	.813	.983	.592	.842	–	–	–	–
MES	.115	.900	.103	.178	.131	.048	.039	.414	.060	.342	.274	.091	.831	–	–	–
EPH	.106	.901	.100	.196	.128	.001	.011	.395	.046	.377	.255	.119	.816	.048	–	–
RSD	.491	.899	.489	.543	.494	.485	.485	.668	.492	.641	.560	.496	.837	.485	.485	–
Min	.003	.903	.068	.214	.078	.106	.109	.445	.137	.420	.305	.168	.801	.116	.107	.492
Max	1.237	.898	1.230	1.218	1.228	1.226	1.225	1.088	1.220	1.125	1.152	1.213	.991	1.222	1.226	1.128

Table D.14: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Mallows preferences.

Figure D.13: Axiom violation rate for each rule on Mallows preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.008**	**0**	**0**	*.000*	**.008**	*.000*	**0**	**0**	**0**	.044	**0**	**0**	**0**	.021	.035
Borda	.015	**0**	.019	**0**	.015	.063	**0**	.043	**0**	.035	**0**	**0**	**0**	.015	.015
EPH	.039	**0**	.016	**0**	.039	.365	.016	.004	**0**	.045	**0**	**0**	**0**	.022	.034
SNTV	.140	**0**	.312	.449	.140	.762	.078	.169	**0**	*.000*	.002	.003	.003	**0**	.041
STV	.105	**0**	.206	.360	.105	.625	.034	.140	**0**	**0**	**0**	**0**	**0**	**0**	**.001**
Bloc	.038	**0**	.017	**0**	.038	.359	.016	**0**	**0**	.044	**0**	**0**	**0**	.021	.043
CC	.159	**0**	.327	.324	.159	.845	.085	.204	**0**	.088	**0**	.008	.008	.055	.117
lex-CC	.048	**0**	.016	**0**	.048	.457	.016	.030	**0**	.052	**0**	**0**	**0**	.026	.026
seq-CC	.149	**0**	.305	.278	.149	.824	.062	.203	**0**	.088	**0**	*.000*	*.000*	.053	.121
Monroe	.145	**0**	.279	.325	.145	.814	.063	.202	**0**	.085	**0**	*.000*	*.000*	.053	.065
Greedy M.	.073	**0**	.106	.029	.073	.547	.017	.099	**0**	.070	**0**	**0**	**0**	.040	.040
PAV	.047	**0**	.016	**0**	.047	.451	.016	.026	**0**	.050	**0**	**0**	**0**	.025	.025
MES	.047	**0**	.017	**0**	.047	.448	.017	.028	**0**	.051	**0**	**0**	**0**	.025	.025
MAV	.151	**0**	.304	.336	.151	.834	.090	.207	**0**	.088	**0**	**0**	**0**	.055	.055
RSD	.138	**0**	.273	**0**	.138	.653	.046	.109	**0**	.079	.078	.078	.078	.047	.353
Random	.192	**0**	.444	.451	.192	.836	.135	.204	**0**	.088	.030	.038	.038	.054	.178

Table D.15: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across SP Conitzer preferences.

D.1.7 5 Alternatives, SP Conitzer

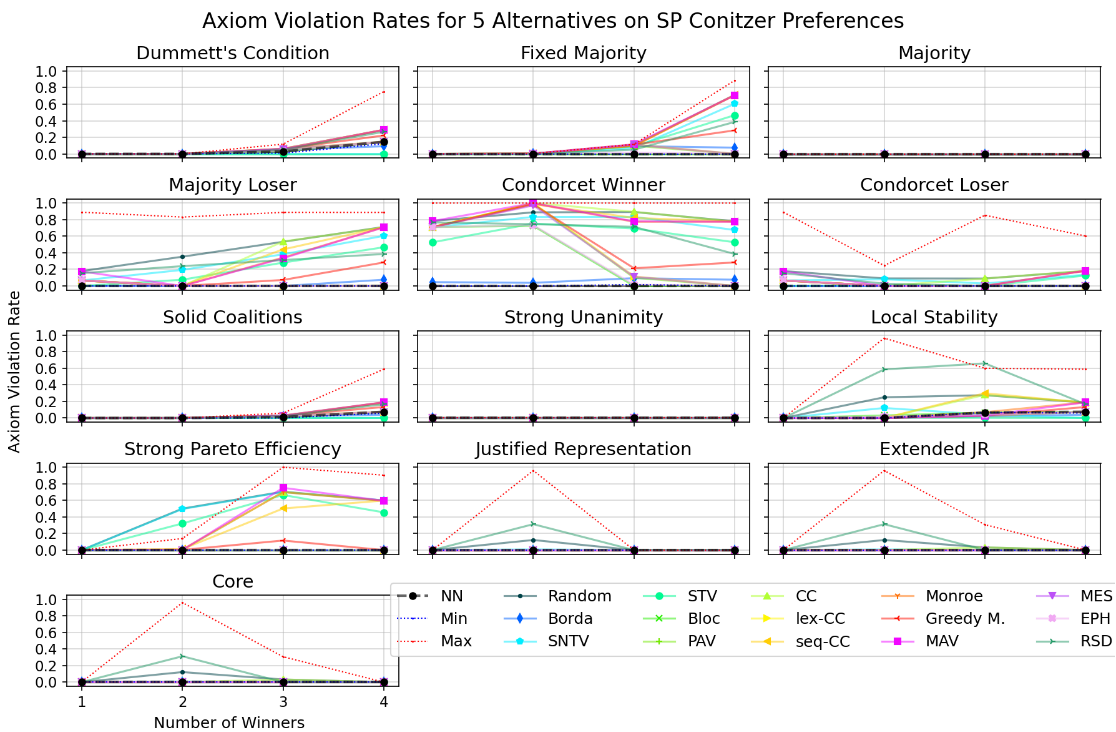


Figure D.14: Axiom violation rate for each axiom on SP Conitzer preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.109	.900	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.809	.898	.802	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.621	.901	.608	.492	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.456	.900	.510	.520	.602	–	–	–	–	–	–	–	–	–	–	–
PAV	.507	.900	.526	.552	.613	.061	–	–	–	–	–	–	–	–	–	–
CC	.831	.899	.842	.600	.715	.385	.345	–	–	–	–	–	–	–	–	–
lex-CC	.509	.900	.528	.554	.615	.064	.003	.344	–	–	–	–	–	–	–	–
seq-CC	.822	.897	.813	.479	.725	.459	.418	.586	.417	–	–	–	–	–	–	–
Monroe	.804	.898	.815	.588	.702	.358	.320	.067	.320	.549	–	–	–	–	–	–
Greedy M.	.607	.899	.593	.482	.646	.244	.191	.454	.189	.305	.435	–	–	–	–	–
MAV	.808	.900	.821	.931	.816	.718	.671	.470	.670	.860	.437	.740	–	–	–	–
MES	.533	.899	.522	.470	.618	.170	.115	.441	.114	.322	.416	.099	.721	–	–	–
EPH	.462	.900	.506	.525	.602	.009	.052	.383	.055	.453	.355	.236	.713	.161	–	–
RSD	.732	.898	.736	.843	.814	.695	.698	.837	.699	.813	.815	.707	.841	.698	.695	–
Min	.009	.901	.106	.806	.618	.465	.515	.833	.518	.820	.806	.607	.811	.530	.471	.732
Max	1.236	.899	1.232	1.047	1.101	1.125	1.093	.976	1.092	.956	1.005	1.075	.973	1.088	1.120	1.015

Table D.16: Difference between rules for 5 alternatives with $1 \leq k < 5$ on SP Conitzer preferences.

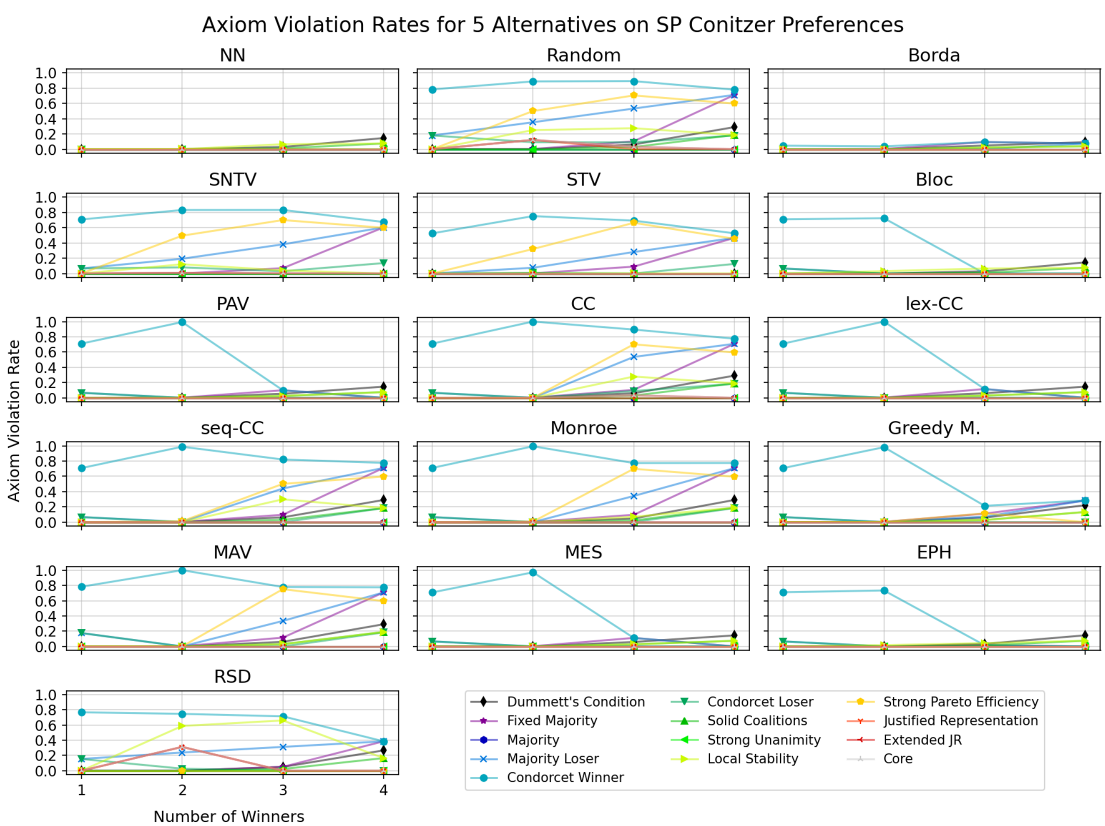


Figure D.15: Axiom violation rate for each rule on SP Conitzer preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Borda	.008	**0**	.028	**0**	.008	.043	**0**	.028	**0**	**0**	**0**	**0**	**0**	**0**	**0**
EPH	.004	**0**	**0**	**0**	.004	.051	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
SNTV	.028	**0**	.084	.003	.028	.194	**0**	.085	**0**	**0**	**0**	**0**	**0**	**0**	**0**
STV	.017	**0**	.045	*.000*	.017	.126	**0**	.045	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Bloc	.004	**0**	**0**	**0**	.004	.050	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
CC	.192	.012	.310	.326	.192	.721	.072	.293	**0**	.237	**0**	.044	.044	.205	.229
lex-CC	.028	.008	**0**	**0**	.028	.299	**0**	.014	**0**	.013	**0**	**0**	**0**	.013	.013
seq-CC	.160	.001	.326	.398	.160	.612	.049	.279	**0**	.151	**0**	*.000*	*.000*	.121	.145
Monroe	.118	.007	.184	.193	.118	.505	.049	.188	**0**	.146	**0**	*.000*	*.000*	.126	.129
Greedy M.	.024	**0**	.072	*.000*	.024	.173	**0**	.073	**0**	*.000*	**0**	**0**	**0**	**0**	*.000*
PAV	.005	**0**	**0**	**0**	.005	.068	**0**	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MES	.005	**0**	**0**	**0**	.005	.060	**0**	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MAV	.176	.015	.305	.342	.176	.831	.094	.281	**0**	.157	**0**	**0**	**0**	.131	.135
RSD	.085	.005	.181	**0**	.085	.512	.013	.154	**0**	.021	.032	.032	.032	.013	.107
Random	.246	.014	.443	.453	.246	.840	.140	.295	**0**	.229	.054	.099	.099	.196	.335

Table D.17: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across SP Walsh preferences.

D.1.8 5 Alternatives, SP Walsh

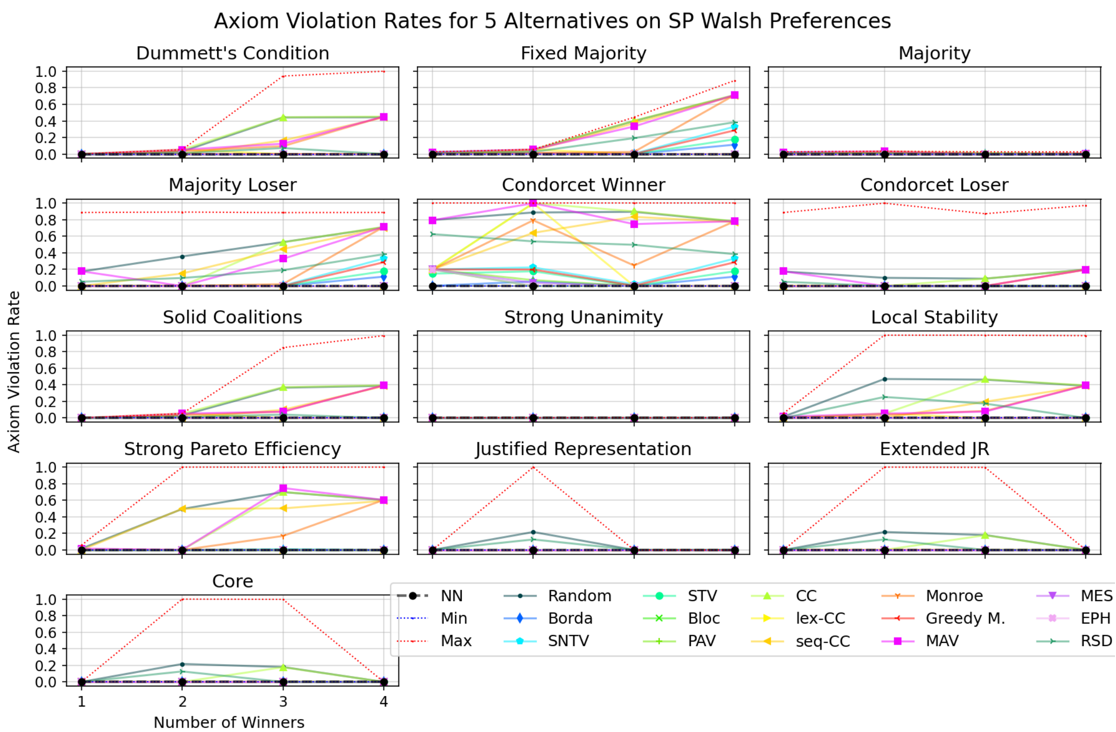


Figure D.16: Axiom violation rate for each axiom on SP Walsh preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.897	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.059	.898	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.239	.895	.201	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.160	.897	.136	.171	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.101	.896	.159	.166	.163	–	–	–	–	–	–	–	–	–	–	–
PAV	.114	.896	.162	.169	.163	.013	–	–	–	–	–	–	–	–	–	–
CC	.571	.897	.580	.507	.543	.470	.467	–	–	–	–	–	–	–	–	–
lex-CC	.226	.896	.271	.250	.251	.125	.122	.345	–	–	–	–	–	–	–	–
seq-CC	.516	.897	.508	.405	.494	.443	.439	.715	.544	–	–	–	–	–	–	–
Monroe	.427	.897	.435	.365	.399	.326	.323	.153	.255	.585	–	–	–	–	–	–
Greedy M.	.223	.894	.223	.154	.213	.150	.146	.515	.243	.391	.371	–	–	–	–	–
MAV	.811	.898	.818	.826	.802	.806	.803	.479	.681	1.028	.510	.834	–	–	–	–
MES	.141	.895	.158	.132	.165	.068	.063	.510	.181	.396	.365	.108	.828	–	–	–
EPH	.102	.896	.159	.166	.163	.001	.013	.470	.125	.442	.326	.149	.806	.067	–	–
RSD	.572	.902	.575	.579	.578	.565	.565	.781	.604	.744	.698	.577	.834	.565	.565	–
Min	.000	.897	.059	.239	.160	.101	.114	.571	.226	.516	.427	.223	.811	.141	.102	.572
Max	1.250	.904	1.250	1.247	1.248	1.248	1.248	1.103	1.248	1.067	1.179	1.247	1.069	1.247	1.248	1.158

Table D.18: Difference between rules for 5 alternatives with $1 \leq k < 5$ on SP Walsh preferences.

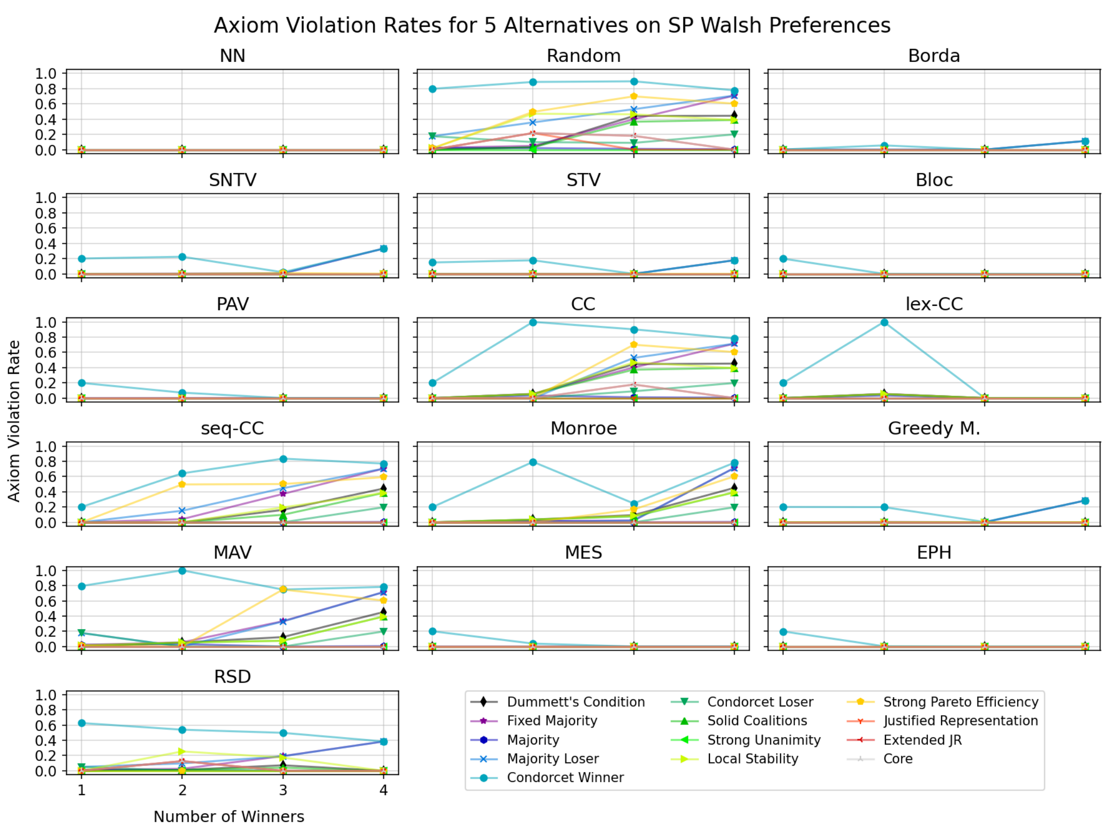


Figure D.17: Axiom violation rate for each rule on SP Walsh preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.006**	*.000*	*.000*	.001	**.006**	**.001**	*.000*	**0**	**0**	.030	*.000*	*.000*	*.000*	.019	.026
Borda	.014	.003	.007	.001	.014	.117	**0**	.017	**0**	.018	*.000*	*.000*	*.000*	.010	.012
EPH	.027	*.000*	.002	**0**	.027	.218	.004	.002	**0**	.046	**0**	**0**	**0**	.033	.040
SNTV	.064	**0**	.097	.104	.064	.530	.016	.081	**0**	.001	*.000*	.003	.003	**0**	.001
STV	.039	**0**	.048	.047	.039	.355	.004	.048	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.026	*.000*	.002	**0**	.026	.207	.004	**0**	**0**	.046	**0**	**0**	**0**	.032	.042
CC	.140	.018	.165	.175	.140	.692	.072	.138	**0**	.186	**0**	.013	.013	.156	.189
lex-CC	.060	.011	.020	**0**	.060	.453	.004	.051	**0**	.092	**0**	**0**	**0**	.073	.074
seq-CC	.121	.014	.151	.111	.121	.657	.043	.133	**0**	.167	**0**	.001	.001	.138	.160
Monroe	.115	.011	.115	.150	.115	.608	.056	.114	**0**	.157	**0**	.001	.001	.130	.150
Greedy M.	.051	.002	.028	.007	.051	.395	.005	.041	**0**	.072	**0**	**0**	**0**	.054	.058
PAV	.031	.001	.003	**0**	.031	.267	.004	.007	**0**	.050	**0**	**0**	**0**	.036	.039
MES	.034	*.000*	.004	.002	.034	.287	.004	.011	**0**	.055	**0**	**0**	**0**	.039	.041
MAV	.151	.032	.172	.110	.151	.781	.115	.152	**0**	.168	.026	.028	.028	.140	.209
RSD	.104	.018	.110	**0**	.104	.609	.047	.075	**0**	.116	.028	.030	.030	.094	.194
Random	.179	.035	.220	.210	.179	.829	.135	.150	**0**	.191	.034	.047	.047	.160	.271

Table D.19: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Gaussian Ball 3 preferences.

D.1.9 5 Alternatives, Gaussian Ball 3

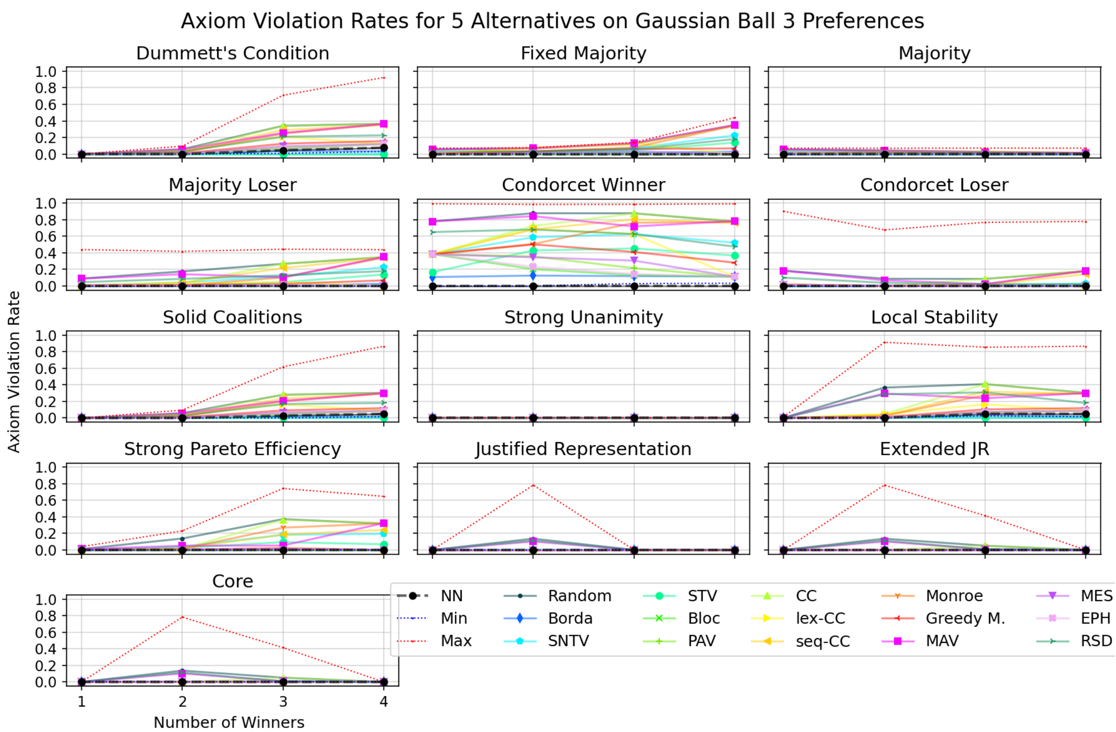


Figure D.18: Axiom violation rate for each axiom on Gaussian Ball 3 preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.898	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.155	.898	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.524	.899	.541	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.319	.899	.337	.335	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.296	.897	.372	.357	.409	–	–	–	–	–	–	–	–	–	–	–
PAV	.327	.896	.376	.374	.416	.067	–	–	–	–	–	–	–	–	–	–
CC	.646	.897	.690	.519	.615	.439	.405	–	–	–	–	–	–	–	–	–
lex-CC	.419	.896	.465	.424	.480	.172	.115	.353	–	–	–	–	–	–	–	–
seq-CC	.628	.897	.650	.449	.594	.444	.415	.570	.390	–	–	–	–	–	–	–
Monroe	.582	.898	.627	.467	.553	.371	.346	.105	.375	.570	–	–	–	–	–	–
Greedy M.	.417	.897	.448	.368	.456	.224	.185	.448	.219	.335	.417	–	–	–	–	–
MAV	.824	.902	.838	.890	.842	.832	.811	.606	.759	.947	.616	.840	–	–	–	–
MES	.342	.896	.385	.343	.421	.123	.077	.450	.159	.368	.392	.140	.830	–	–	–
EPH	.302	.896	.370	.359	.407	.014	.058	.435	.165	.438	.367	.216	.831	.114	–	–
RSD	.670	.898	.683	.727	.704	.639	.643	.786	.670	.758	.755	.659	.866	.643	.639	–
Min	.020	.899	.156	.517	.311	.309	.340	.643	.430	.630	.579	.423	.822	.353	.315	.673
Max	1.228	.899	1.223	1.147	1.190	1.213	1.192	1.051	1.149	1.043	1.100	1.165	.991	1.187	1.209	1.066

Table D.20: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Gaussian Ball 3 preferences.

Figure D.19: Axiom violation rate for each rule on Gaussian Ball 3 preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.006**	*.000*	*.000*	.001	**.006**	**.001**	*.000*	**0**	**0**	.030	*.000*	*.000*	*.000*	.019	.026
Borda	.014	.003	.008	.001	.014	.118	**0**	.017	**0**	.018	*.000*	*.000*	*.000*	.010	.013
EPH	.027	*.000*	.002	**0**	.027	.221	.004	.001	**0**	.047	**0**	**0**	**0**	.033	.041
SNTV	.064	**0**	.096	.102	.064	.529	.016	.081	**0**	.001	*.000*	.002	.002	**0**	.001
STV	.038	**0**	.048	.045	.038	.350	.004	.049	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.026	*.000*	.002	**0**	.026	.210	.004	**0**	**0**	.046	**0**	**0**	**0**	.032	.043
CC	.141	.019	.166	.178	.141	.695	.073	.139	**0**	.189	**0**	.013	.013	.158	.192
lex-CC	.060	.012	.019	**0**	.060	.458	.005	.051	**0**	.093	**0**	**0**	**0**	.074	.075
seq-CC	.122	.014	.153	.112	.122	.658	.043	.134	**0**	.167	**0**	.001	.001	.138	.161
Monroe	.116	.012	.117	.153	.116	.610	.057	.114	**0**	.159	**0**	.001	.001	.131	.152
Greedy M.	.051	.002	.029	.007	.051	.398	.006	.040	**0**	.072	**0**	**0**	**0**	.054	.058
PAV	.031	.001	.003	**0**	.031	.269	.004	.007	**0**	.050	**0**	**0**	**0**	.036	.039
MES	.034	.001	.004	.002	.034	.288	.005	.010	**0**	.055	**0**	**0**	**0**	.040	.041
MAV	.151	.033	.172	.111	.151	.784	.115	.154	**0**	.169	.025	.027	.027	.141	.209
RSD	.104	.018	.109	**0**	.104	.610	.047	.076	**0**	.118	.028	.030	.030	.096	.195
Random	.179	.036	.220	.211	.179	.829	.135	.151	**0**	.191	.033	.048	.048	.160	.271

Table D.21: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Gaussian Ball 10 preferences.

D.1.10 5 Alternatives, Gaussian Ball 10

Figure D.20: Axiom violation rate for each axiom on Gaussian Ball 10 preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.899	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.156	.899	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.524	.897	.540	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.315	.899	.333	.336	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.299	.897	.372	.357	.411	–	–	–	–	–	–	–	–	–	–	–
PAV	.329	.897	.377	.373	.417	.068	–	–	–	–	–	–	–	–	–	–
CC	.649	.897	.693	.522	.620	.440	.405	–	–	–	–	–	–	–	–	–
lex-CC	.423	.898	.467	.425	.483	.174	.116	.354	–	–	–	–	–	–	–	–
seq-CC	.628	.899	.649	.449	.597	.445	.415	.573	.390	–	–	–	–	–	–	–
Monroe	.585	.896	.629	.469	.557	.372	.346	.105	.374	.571	–	–	–	–	–	–
Greedy M.	.420	.898	.450	.367	.459	.228	.188	.452	.223	.334	.419	–	–	–	–	–
MAV	.827	.901	.840	.891	.844	.832	.811	.607	.759	.949	.617	.842	–	–	–	–
MES	.344	.898	.385	.340	.422	.125	.079	.452	.162	.366	.393	.142	.832	–	–	–
EPH	.304	.897	.371	.358	.409	.014	.058	.435	.167	.439	.368	.220	.831	.116	–	–
RSD	.670	.899	.684	.729	.703	.639	.644	.785	.670	.758	.755	.661	.865	.645	.640	–
Min	.019	.899	.157	.517	.308	.311	.341	.646	.433	.631	.582	.427	.826	.355	.317	.672
Max	1.228	.903	1.223	1.147	1.191	1.213	1.192	1.051	1.149	1.043	1.099	1.164	.992	1.187	1.209	1.066

Table D.22: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Gaussian Ball 10 preferences.

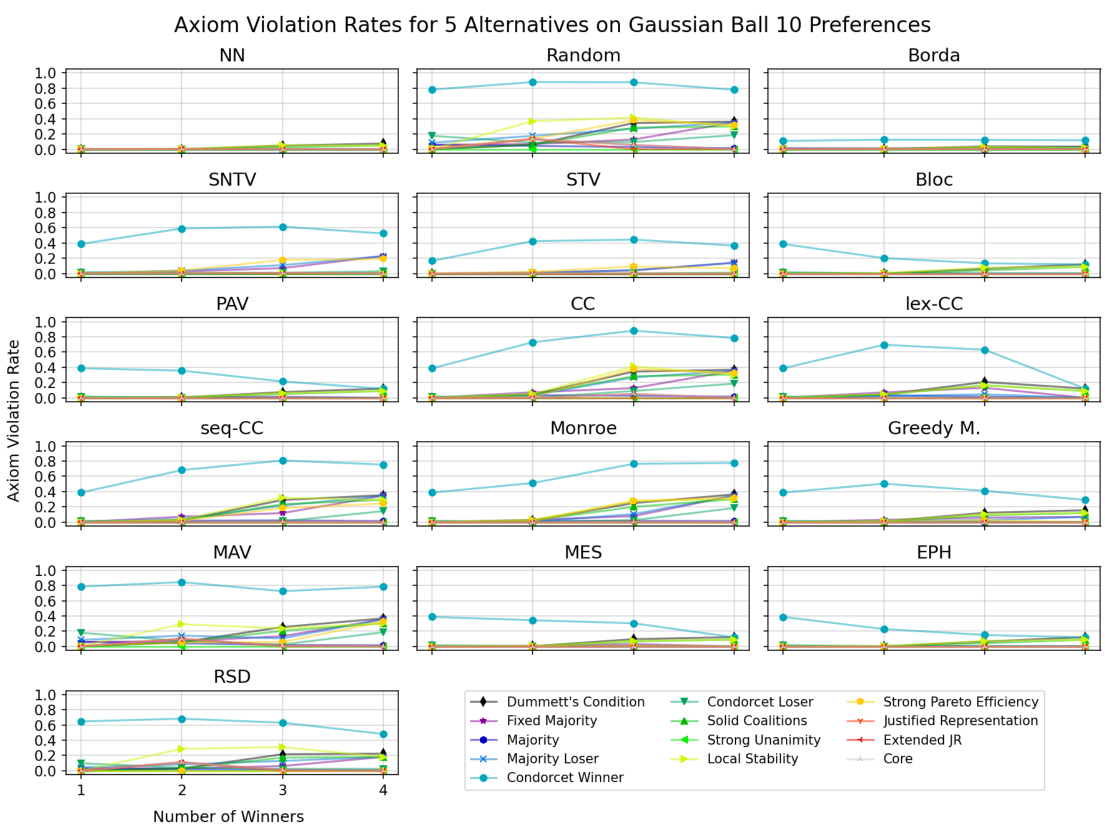


Figure D.21: Axiom violation rate for each rule on Gaussian Ball 10 preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.009**	*.000*	*.000*	.002	**.009**	**.001**	*.000*	**0**	**0**	.045	*.000*	*.000*	*.000*	.030	.041
Borda	.017	.003	.006	.001	.017	.136	**0**	.014	**0**	.025	*.000*	*.000*	*.000*	.015	.018
EPH	.036	*.000*	.003	**0**	.036	.270	.007	.002	**0**	.071	**0**	**0**	**0**	.052	.062
SNTV	.066	**0**	.080	.115	.066	.570	.023	.063	**0**	.001	*.000*	.002	.002	**0**	.002
STV	.043	**0**	.043	.058	.043	.405	.005	.042	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.035	*.000*	.003	**0**	.035	.255	.007	**0**	**0**	.070	**0**	**0**	**0**	.051	.066
CC	.130	.014	.112	.160	.130	.699	.072	.093	**0**	.182	**0**	.009	.009	.151	.184
lex-CC	.066	.009	.015	**0**	.066	.491	.007	.036	**0**	.115	**0**	**0**	**0**	.092	.093
seq-CC	.114	.010	.105	.099	.114	.667	.045	.091	**0**	.167	**0**	*.000*	*.000*	.137	.161
Monroe	.113	.010	.086	.148	.113	.640	.057	.082	**0**	.161	**0**	.001	.001	.132	.155
Greedy M.	.061	.002	.024	.007	.061	.454	.009	.033	**0**	.100	**0**	**0**	**0**	.078	.083
PAV	.042	.001	.003	**0**	.042	.329	.007	.008	**0**	.076	**0**	**0**	**0**	.056	.060
MES	.045	.001	.005	.002	.045	.354	.007	.012	**0**	.083	**0**	**0**	**0**	.062	.064
MAV	.140	.025	.116	.098	.140	.780	.110	.105	**0**	.170	.022	.024	.024	.141	.206
RSD	.108	.015	.083	**0**	.108	.634	.055	.052	**0**	.137	.032	.033	.033	.110	.221
Random	.164	.026	.149	.195	.164	.822	.131	.103	**0**	.185	.030	.038	.038	.154	.260

Table D.23: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Uniform Ball 3 preferences.

D.1.11 5 Alternatives, Uniform Ball 3

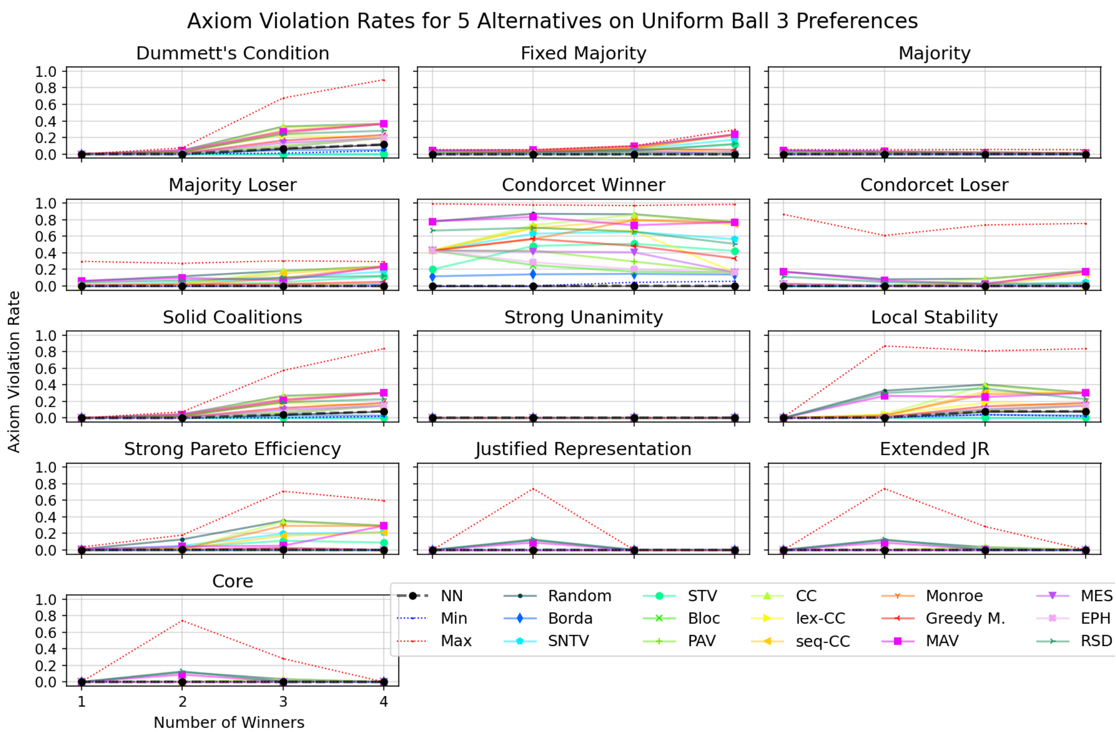


Figure D.22: Axiom violation rate for each axiom on Uniform Ball 3 preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.181	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.576	.901	.592	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.373	.900	.385	.355	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.350	.900	.428	.395	.468	–	–	–	–	–	–	–	–	–	–	–
PAV	.390	.900	.437	.416	.479	.083	–	–	–	–	–	–	–	–	–	–
CC	.669	.899	.718	.532	.637	.438	.395	–	–	–	–	–	–	–	–	–
lex-CC	.470	.900	.517	.459	.535	.177	.105	.350	–	–	–	–	–	–	–	–
seq-CC	.658	.900	.677	.459	.619	.450	.415	.571	.395	–	–	–	–	–	–	–
Monroe	.622	.898	.673	.493	.592	.387	.356	.088	.376	.579	–	–	–	–	–	–
Greedy M.	.478	.900	.511	.406	.522	.255	.208	.448	.230	.326	.436	–	–	–	–	–
MAV	.825	.901	.843	.897	.848	.832	.807	.602	.762	.952	.610	.844	–	–	–	–
MES	.406	.901	.448	.381	.486	.149	.093	.448	.158	.360	.409	.158	.830	–	–	–
EPH	.359	.900	.426	.397	.467	.019	.070	.433	.167	.441	.382	.245	.830	.137	–	–
RSD	.706	.900	.718	.763	.745	.668	.674	.795	.694	.770	.775	.692	.870	.676	.668	–
Min	.031	.901	.181	.565	.360	.370	.408	.664	.487	.662	.618	.489	.822	.423	.379	.710
Max	1.221	.899	1.214	1.126	1.175	1.194	1.166	1.042	1.126	1.022	1.082	1.133	.995	1.159	1.188	1.039

Table D.24: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Uniform Ball 3 preferences.

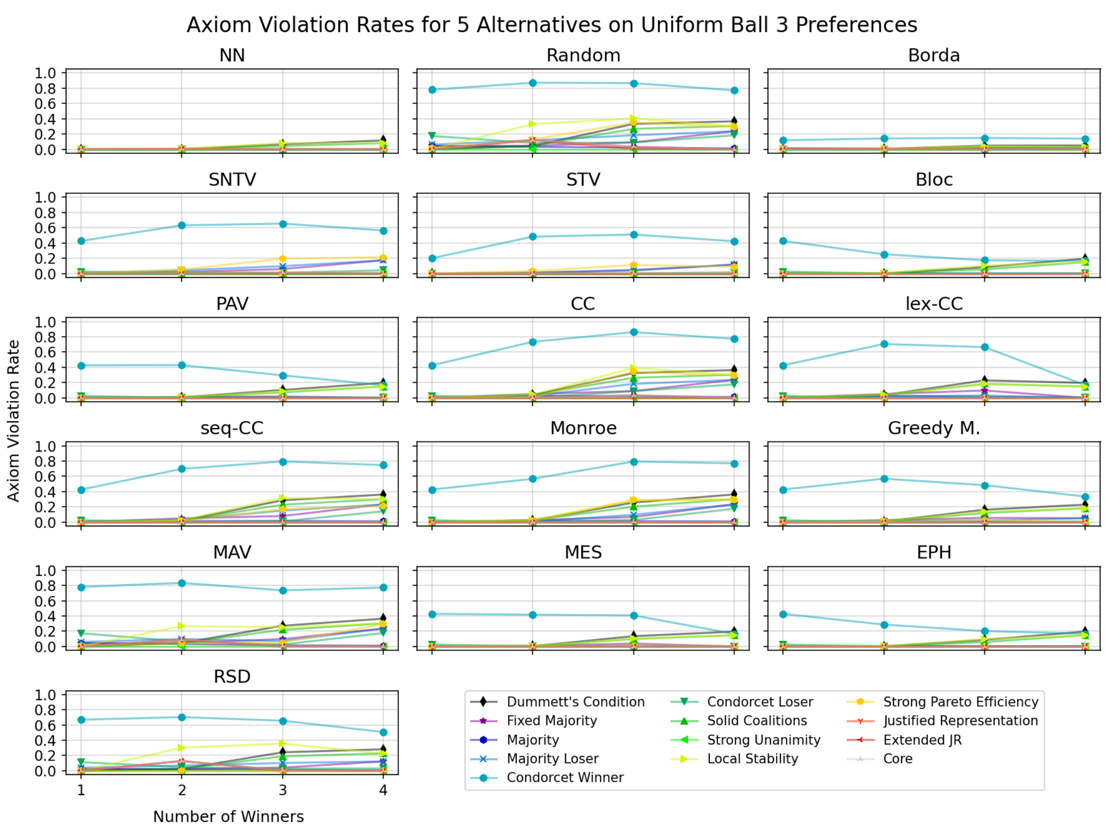


Figure D.23: Axiom violation rate for each rule on Uniform Ball 3 preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.009**	*.000*	*.000*	.002	**.009**	**.001**	*.000*	**0**	**0**	.044	*.000*	*.000*	*.000*	.028	.039
Borda	.017	.003	.006	.001	.017	.137	**0**	.014	**0**	.024	*.000*	*.000*	*.000*	.013	.017
EPH	.036	*.000*	.003	**0**	.036	.274	.007	.002	**0**	.069	**0**	**0**	**0**	.050	.062
SNTV	.066	**0**	.078	.114	.066	.573	.022	.060	**0**	.001	*.000*	.003	.003	**0**	.002
STV	.043	**0**	.043	.057	.043	.408	.005	.040	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.035	*.000*	.003	**0**	.035	.259	.007	**0**	**0**	.069	**0**	**0**	**0**	.050	.065
CC	.130	.015	.108	.162	.130	.700	.074	.090	**0**	.181	**0**	.009	.009	.151	.186
lex-CC	.065	.009	.015	**0**	.065	.495	.007	.035	**0**	.110	**0**	**0**	**0**	.087	.088
seq-CC	.113	.011	.100	.097	.113	.668	.043	.086	**0**	.165	**0**	*.000*	*.000*	.136	.159
Monroe	.113	.011	.082	.148	.113	.642	.060	.078	**0**	.161	**0**	.001	.001	.133	.157
Greedy M.	.060	.002	.023	.009	.060	.457	.008	.032	**0**	.096	**0**	**0**	**0**	.073	.078
PAV	.041	.001	.003	**0**	.041	.333	.007	.007	**0**	.073	**0**	**0**	**0**	.054	.058
MES	.045	.001	.005	.003	.045	.360	.007	.011	**0**	.079	**0**	**0**	**0**	.058	.060
MAV	.139	.026	.112	.097	.139	.779	.113	.102	**0**	.168	.022	.023	.023	.140	.205
RSD	.107	.015	.079	**0**	.107	.634	.055	.051	**0**	.135	.030	.032	.032	.110	.220
Random	.163	.027	.145	.194	.163	.825	.131	.100	**0**	.182	.030	.038	.038	.151	.257

Table D.25: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Uniform Ball 10 preferences.

D.1.12 5 Alternatives, Uniform Ball 10

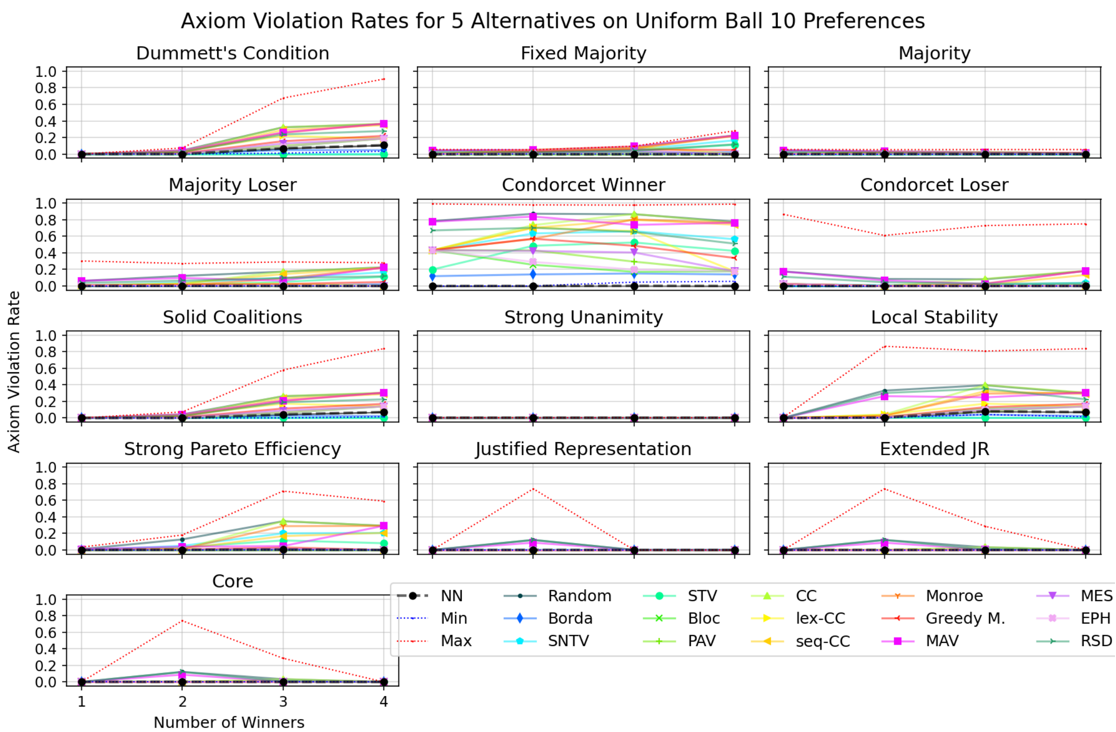


Figure D.24: Axiom violation rate for each axiom on Uniform Ball 10 preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.180	.902	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.578	.901	.593	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.373	.903	.387	.355	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.354	.901	.432	.393	.470	–	–	–	–	–	–	–	–	–	–	–
PAV	.394	.900	.440	.414	.480	.083	–	–	–	–	–	–	–	–	–	–
CC	.669	.899	.718	.530	.636	.435	.394	–	–	–	–	–	–	–	–	–
lex-CC	.474	.901	.519	.458	.535	.176	.105	.349	–	–	–	–	–	–	–	–
seq-CC	.660	.900	.676	.460	.621	.448	.412	.571	.390	–	–	–	–	–	–	–
Monroe	.622	.899	.672	.492	.592	.384	.354	.088	.374	.579	–	–	–	–	–	–
Greedy M.	.481	.899	.512	.408	.524	.258	.210	.448	.231	.325	.436	–	–	–	–	–
MAV	.822	.899	.840	.895	.847	.827	.802	.600	.756	.949	.607	.840	–	–	–	–
MES	.413	.900	.452	.381	.488	.151	.095	.445	.158	.356	.407	.158	.826	–	–	–
EPH	.363	.900	.429	.396	.469	.020	.070	.429	.166	.439	.379	.247	.825	.138	–	–
RSD	.705	.899	.717	.762	.743	.668	.674	.797	.695	.773	.776	.694	.867	.676	.669	–
Min	.030	.901	.180	.568	.360	.374	.412	.665	.490	.663	.619	.492	.819	.429	.382	.710
Max	1.220	.898	1.212	1.125	1.173	1.193	1.165	1.038	1.125	1.023	1.078	1.132	.993	1.157	1.187	1.039

Table D.26: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Uniform Ball 10 preferences.

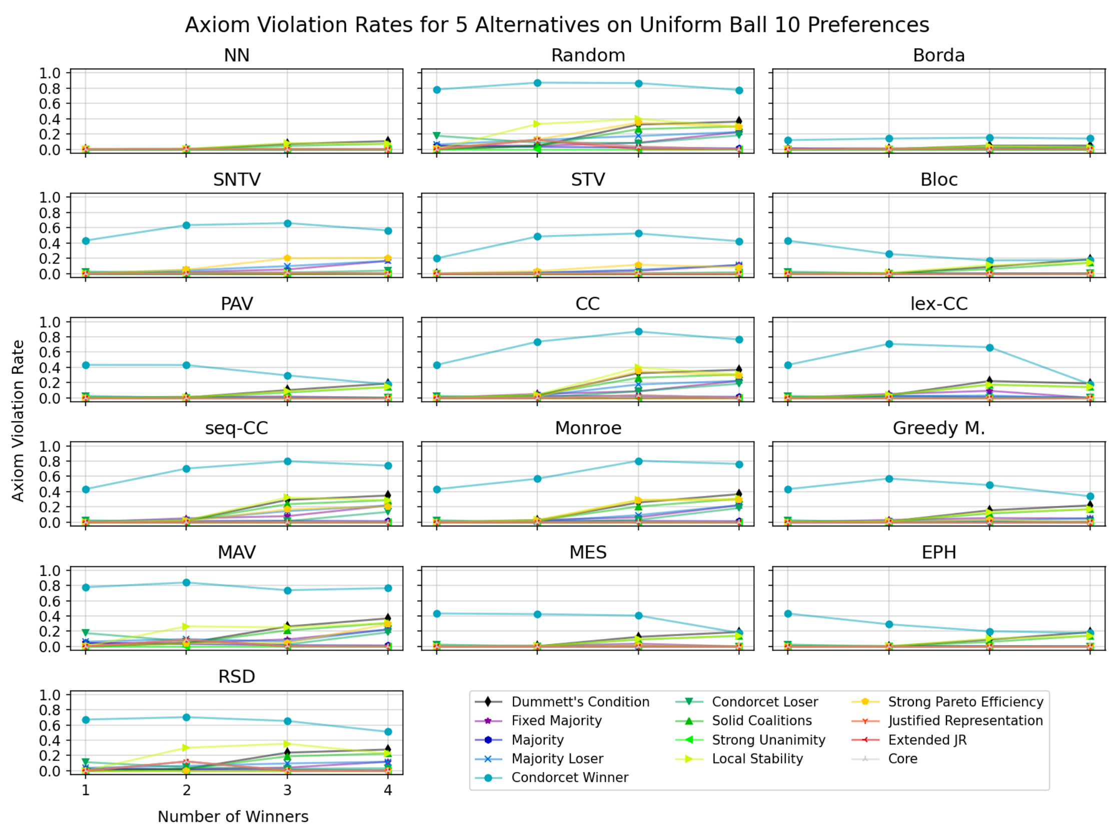


Figure D.25: Axiom violation rate for each rule on Uniform Ball 10 preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.008**	**0**	*.000*	.002	**.008**	**.001**	*.000*	**0**	**0**	.038	*.000*	*.000*	*.000*	.025	.036
Borda	.016	.003	.006	.001	.016	.128	**0**	.016	**0**	.021	**0**	**0**	**0**	.013	.016
EPH	.032	*.000*	.003	**0**	.032	.249	.006	.002	**0**	.059	**0**	**0**	**0**	.043	.053
SNTV	.066	**0**	.089	.116	.066	.557	.020	.072	**0**	.001	*.000*	.002	.002	**0**	.002
STV	.041	**0**	.045	.057	.041	.382	.005	.045	**0**	**0**	*.000*	*.000*	*.000*	**0**	*.000*
Bloc	.031	*.000*	.003	**0**	.031	.236	.006	**0**	**0**	.058	**0**	**0**	**0**	.042	.056
CC	.136	.017	.145	.172	.136	.699	.073	.119	**0**	.184	**0**	.011	.011	.153	.187
lex-CC	.062	.010	.016	**0**	.062	.471	.006	.042	**0**	.100	**0**	**0**	**0**	.079	.081
seq-CC	.120	.012	.135	.110	.120	.668	.045	.116	**0**	.170	**0**	.001	.001	.140	.163
Monroe	.116	.011	.104	.152	.116	.630	.058	.103	**0**	.160	**0**	.001	.001	.131	.153
Greedy M.	.057	.003	.028	.008	.057	.432	.007	.040	**0**	.087	**0**	**0**	**0**	.067	.072
PAV	.037	.001	.003	**0**	.037	.304	.006	.008	**0**	.063	**0**	**0**	**0**	.046	.051
MES	.040	.001	.004	.003	.040	.327	.006	.011	**0**	.068	**0**	**0**	**0**	.050	.053
MAV	.145	.029	.150	.108	.145	.777	.113	.132	**0**	.167	.022	.023	.023	.138	.202
RSD	.106	.016	.101	**0**	.106	.621	.050	.066	**0**	.128	.029	.030	.030	.103	.209
Random	.173	.031	.194	.214	.173	.826	.133	.130	**0**	.187	.031	.041	.041	.156	.265

Table D.27: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Gaussian Cube 3 preferences.

D.1.13 5 Alternatives, Gaussian Cube 3

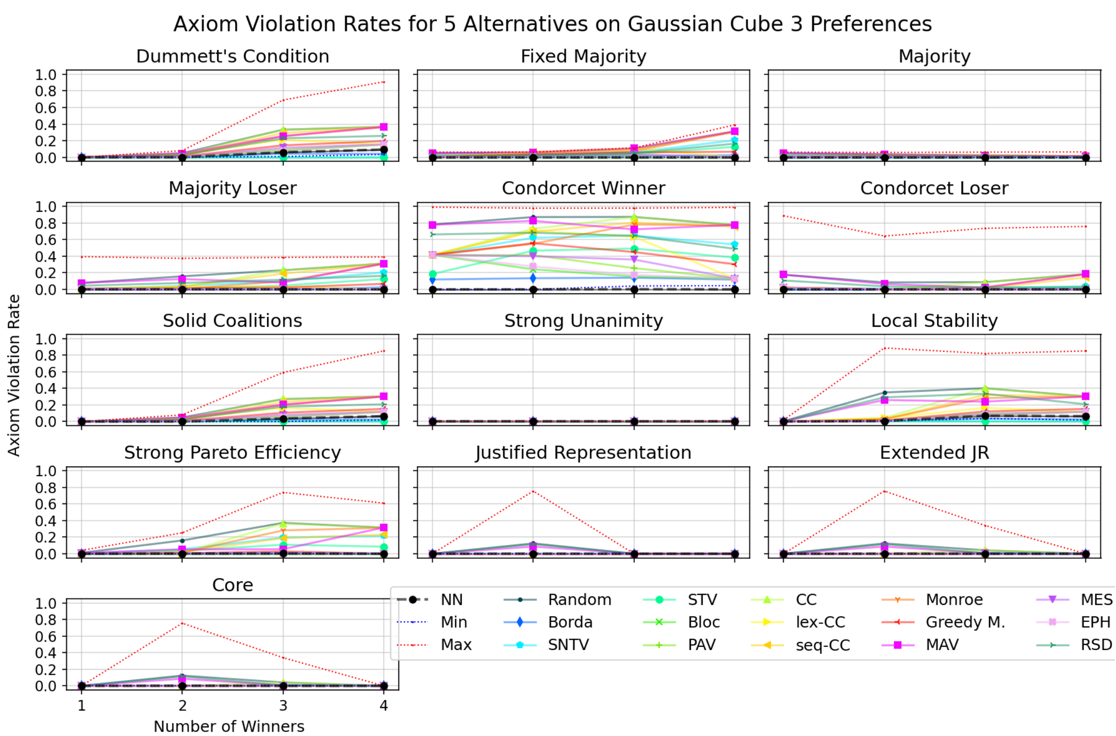


Figure D.26: Axiom violation rate for each axiom on Gaussian Cube 3 preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.171	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.558	.901	.577	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.348	.900	.367	.351	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.329	.901	.405	.378	.442	–	–	–	–	–	–	–	–	–	–	–
PAV	.365	.901	.411	.397	.450	.076	–	–	–	–	–	–	–	–	–	–
CC	.664	.900	.710	.528	.631	.438	.398	–	–	–	–	–	–	–	–	–
lex-CC	.448	.901	.492	.441	.507	.173	.108	.351	–	–	–	–	–	–	–	–
seq-CC	.651	.900	.673	.458	.615	.447	.415	.572	.394	–	–	–	–	–	–	–
Monroe	.610	.900	.658	.483	.578	.378	.350	.096	.372	.575	–	–	–	–	–	–
Greedy M.	.456	.901	.487	.391	.495	.241	.197	.448	.223	.330	.428	–	–	–	–	–
MAV	.821	.897	.836	.897	.844	.830	.805	.602	.756	.948	.611	.839	–	–	–	–
MES	.382	.901	.422	.364	.458	.136	.086	.448	.157	.363	.401	.149	.827	–	–	–
EPH	.336	.901	.402	.381	.441	.018	.065	.432	.164	.439	.373	.231	.828	.125	–	–
RSD	.688	.900	.701	.748	.722	.654	.659	.791	.682	.767	.765	.677	.866	.662	.654	–
Min	.026	.901	.171	.549	.337	.345	.380	.660	.461	.654	.607	.465	.819	.395	.352	.692
Max	1.223	.901	1.216	1.132	1.180	1.201	1.176	1.046	1.136	1.028	1.089	1.145	.996	1.170	1.196	1.054

Table D.28: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Gaussian Cube 3 preferences.

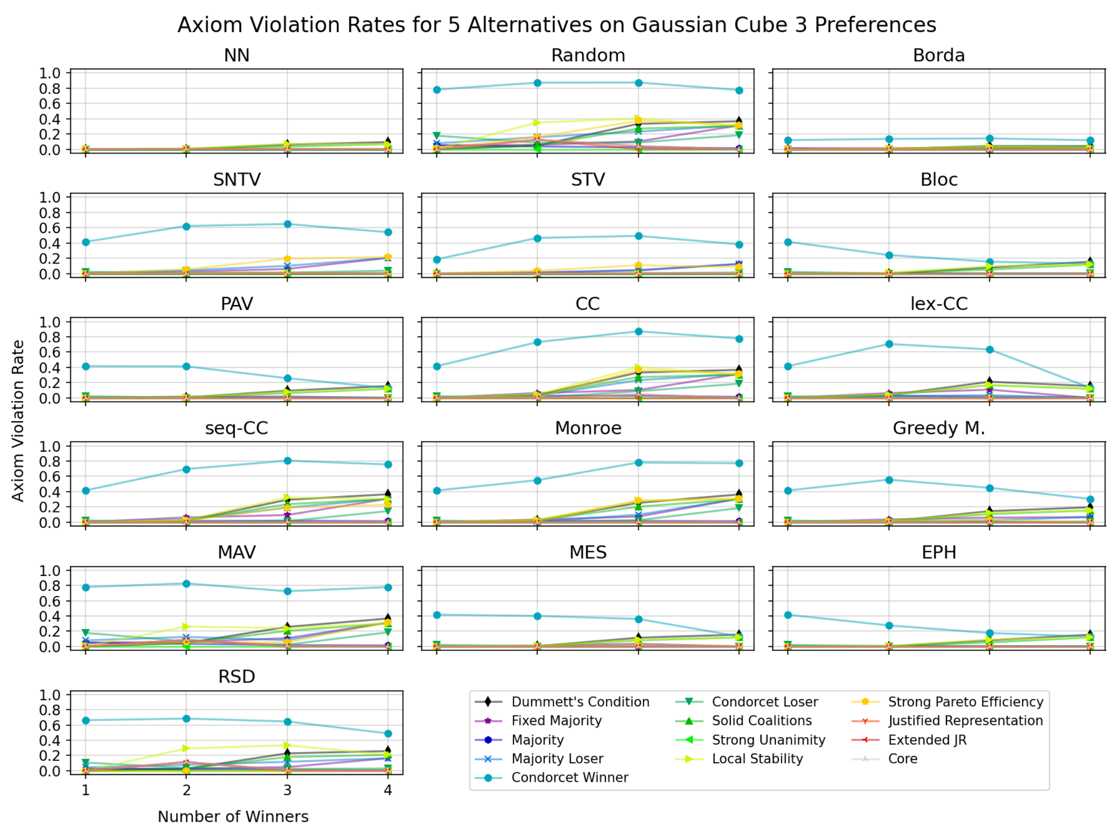


Figure D.27: Axiom violation rate for each rule on Gaussian Cube 3 preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.008**	*.000*	*.000*	.001	**.008**	**.001**	*.000*	**0**	**0**	.039	*.000*	*.000*	*.000*	.025	.035
Borda	.016	.003	.006	.001	.016	.130	**0**	.016	**0**	.022	*.000*	*.000*	*.000*	.013	.016
EPH	.032	*.000*	.003	**0**	.032	.249	.006	.002	**0**	.060	**0**	**0**	**0**	.043	.053
SNTV	.066	**0**	.089	.115	.066	.557	.021	.071	**0**	.001	*.000*	.003	.003	**0**	.002
STV	.041	**0**	.046	.057	.041	.386	.004	.044	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.031	*.000*	.003	**0**	.031	.236	.006	**0**	**0**	.059	**0**	**0**	**0**	.042	.056
CC	.137	.017	.145	.171	.137	.701	.072	.117	**0**	.187	**0**	.011	.011	.156	.190
lex-CC	.063	.010	.017	**0**	.063	.472	.006	.042	**0**	.103	**0**	**0**	**0**	.081	.082
seq-CC	.120	.012	.135	.109	.120	.668	.044	.115	**0**	.169	**0**	*.000*	*.000*	.140	.163
Monroe	.116	.011	.103	.150	.116	.631	.056	.100	**0**	.162	**0**	.001	.001	.133	.155
Greedy M.	.057	.002	.027	.008	.057	.430	.007	.038	**0**	.088	**0**	**0**	**0**	.067	.072
PAV	.037	.001	.003	**0**	.037	.304	.006	.008	**0**	.064	**0**	**0**	**0**	.047	.051
MES	.041	.001	.005	.003	.041	.328	.006	.012	**0**	.070	**0**	**0**	**0**	.051	.053
MAV	.145	.029	.147	.105	.145	.780	.111	.130	**0**	.169	.023	.024	.024	.141	.205
RSD	.107	.016	.099	**0**	.107	.621	.050	.066	**0**	.129	.030	.032	.032	.104	.210
Random	.173	.031	.194	.212	.173	.827	.132	.128	**0**	.188	.031	.041	.041	.157	.265

Table D.29: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Gaussian Cube 10 preferences.

D.1.14 5 Alternatives, Gaussian Cube 10

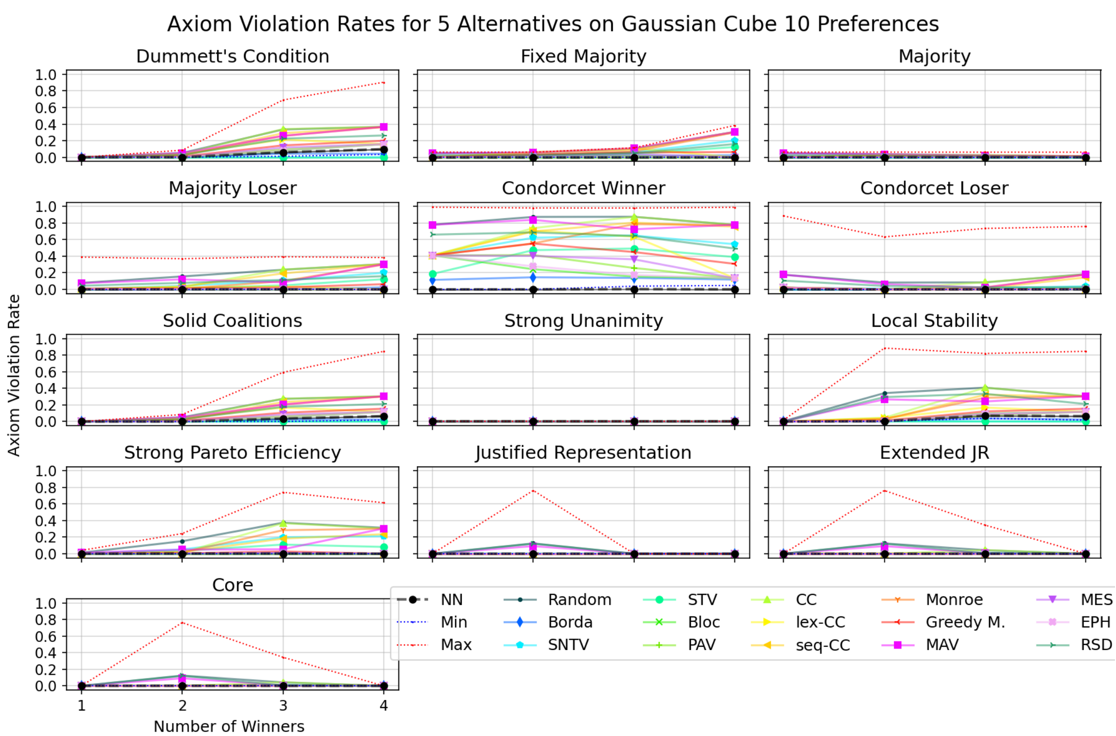


Figure D.28: Axiom violation rate for each axiom on Gaussian Cube 10 preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.900	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.169	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.558	.900	.577	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.352	.899	.368	.350	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.328	.899	.402	.377	.441	–	–	–	–	–	–	–	–	–	–	–
PAV	.364	.899	.409	.397	.450	.077	–	–	–	–	–	–	–	–	–	–
CC	.664	.897	.711	.527	.631	.438	.399	–	–	–	–	–	–	–	–	–
lex-CC	.447	.899	.490	.442	.508	.173	.107	.353	–	–	–	–	–	–	–	–
seq-CC	.649	.900	.669	.458	.612	.449	.415	.572	.393	–	–	–	–	–	–	–
Monroe	.609	.897	.657	.483	.579	.379	.350	.096	.373	.574	–	–	–	–	–	–
Greedy M.	.453	.900	.484	.390	.493	.242	.198	.449	.223	.330	.427	–	–	–	–	–
MAV	.822	.898	.837	.894	.844	.828	.804	.601	.754	.944	.610	.836	–	–	–	–
MES	.380	.900	.420	.363	.457	.138	.086	.449	.156	.363	.401	.149	.824	–	–	–
EPH	.335	.899	.400	.380	.440	.018	.065	.433	.164	.440	.374	.232	.827	.126	–	–
RSD	.688	.900	.702	.746	.723	.652	.657	.789	.680	.764	.764	.676	.864	.660	.652	–
Min	.026	.900	.169	.549	.341	.344	.379	.660	.461	.652	.606	.462	.820	.394	.351	.692
Max	1.223	.901	1.216	1.132	1.179	1.201	1.177	1.046	1.137	1.030	1.090	1.146	.996	1.170	1.196	1.053

Table D.30: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Gaussian Cube 10 preferences.

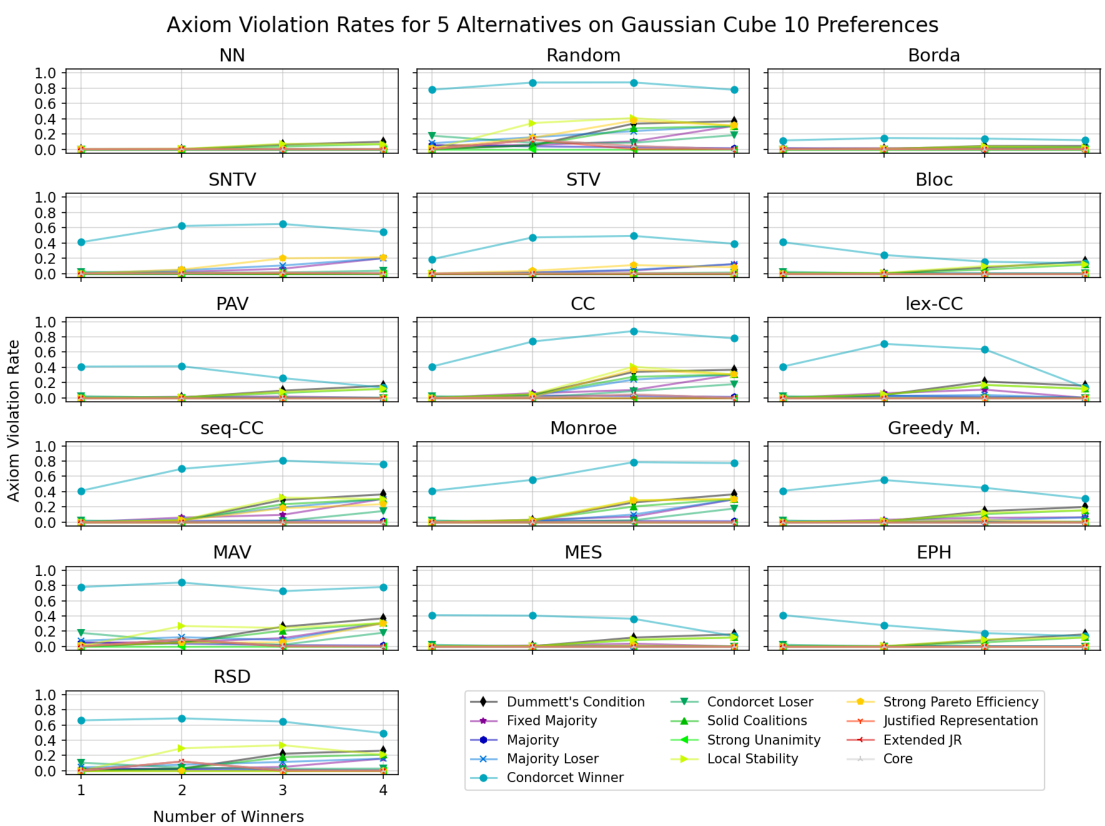


Figure D.29: Axiom violation rate for each rule on Gaussian Cube 10 preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.008**	*.000*	*.000*	.002	**.008**	**.001**	*.000*	**0**	**0**	.039	*.000*	*.000*	*.000*	.025	.036
Borda	.016	.003	.006	.001	.016	.133	**0**	.017	**0**	.023	*.000*	*.000*	*.000*	.013	.017
EPH	.033	*.000*	.003	**0**	.033	.256	.006	.002	**0**	.063	**0**	**0**	**0**	.046	.057
SNTV	.068	**0**	.089	.123	.068	.569	.022	.073	**0**	.001	*.000*	.003	.003	**0**	.002
STV	.042	**0**	.046	.059	.042	.394	.006	.046	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.032	*.000*	.003	**0**	.032	.242	.006	**0**	**0**	.062	**0**	**0**	**0**	.045	.060
CC	.136	.016	.141	.172	.136	.701	.073	.116	**0**	.185	**0**	.010	.010	.153	.188
lex-CC	.063	.010	.015	**0**	.063	.474	.006	.040	**0**	.104	**0**	**0**	**0**	.082	.084
seq-CC	.120	.012	.132	.109	.120	.669	.045	.113	**0**	.170	**0**	.001	.001	.140	.164
Monroe	.116	.012	.103	.152	.116	.636	.058	.101	**0**	.163	**0**	.001	.001	.133	.156
Greedy M.	.059	.003	.027	.008	.059	.439	.008	.039	**0**	.092	**0**	**0**	**0**	.070	.075
PAV	.039	.001	.003	**0**	.039	.313	.006	.008	**0**	.067	**0**	**0**	**0**	.049	.053
MES	.042	.001	.004	.003	.042	.337	.006	.012	**0**	.072	**0**	**0**	**0**	.053	.055
MAV	.144	.028	.144	.105	.144	.779	.112	.128	**0**	.169	.021	.022	.022	.140	.200
RSD	.108	.016	.097	**0**	.108	.627	.050	.065	**0**	.131	.033	.034	.034	.106	.216
Random	.172	.030	.189	.211	.172	.826	.132	.127	**0**	.186	.031	.041	.041	.155	.262

Table D.31: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Uniform Cube 3 preferences.

D.1.15 5 Alternatives, Uniform Cube 3

Figure D.30: Axiom violation rate for each axiom on Uniform Cube 3 preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.901	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.173	.900	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.570	.900	.586	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.362	.901	.376	.353	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.336	.900	.410	.385	.448	–	–	–	–	–	–	–	–	–	–	–
PAV	.373	.901	.418	.405	.459	.080	–	–	–	–	–	–	–	–	–	–
CC	.667	.901	.713	.528	.634	.439	.396	–	–	–	–	–	–	–	–	–
lex-CC	.453	.901	.495	.447	.512	.173	.103	.352	–	–	–	–	–	–	–	–
seq-CC	.654	.900	.674	.458	.616	.449	.412	.570	.391	–	–	–	–	–	–	–
Monroe	.616	.900	.662	.486	.584	.382	.351	.093	.372	.572	–	–	–	–	–	–
Greedy M.	.462	.901	.491	.396	.500	.244	.196	.445	.220	.328	.426	–	–	–	–	–
MAV	.822	.900	.835	.892	.843	.824	.799	.598	.752	.942	.605	.832	–	–	–	–
MES	.389	.901	.429	.372	.465	.142	.086	.445	.151	.361	.401	.148	.821	–	–	–
EPH	.343	.900	.408	.387	.447	.019	.067	.432	.163	.440	.376	.235	.822	.130	–	–
RSD	.694	.897	.710	.751	.729	.653	.659	.789	.681	.767	.765	.678	.866	.663	.654	–
Min	.026	.901	.174	.562	.352	.353	.388	.664	.466	.657	.613	.470	.820	.403	.360	.698
Max	1.221	.899	1.213	1.128	1.176	1.198	1.171	1.044	1.133	1.026	1.085	1.140	.998	1.165	1.192	1.049

Table D.32: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Uniform Cube 3 preferences.

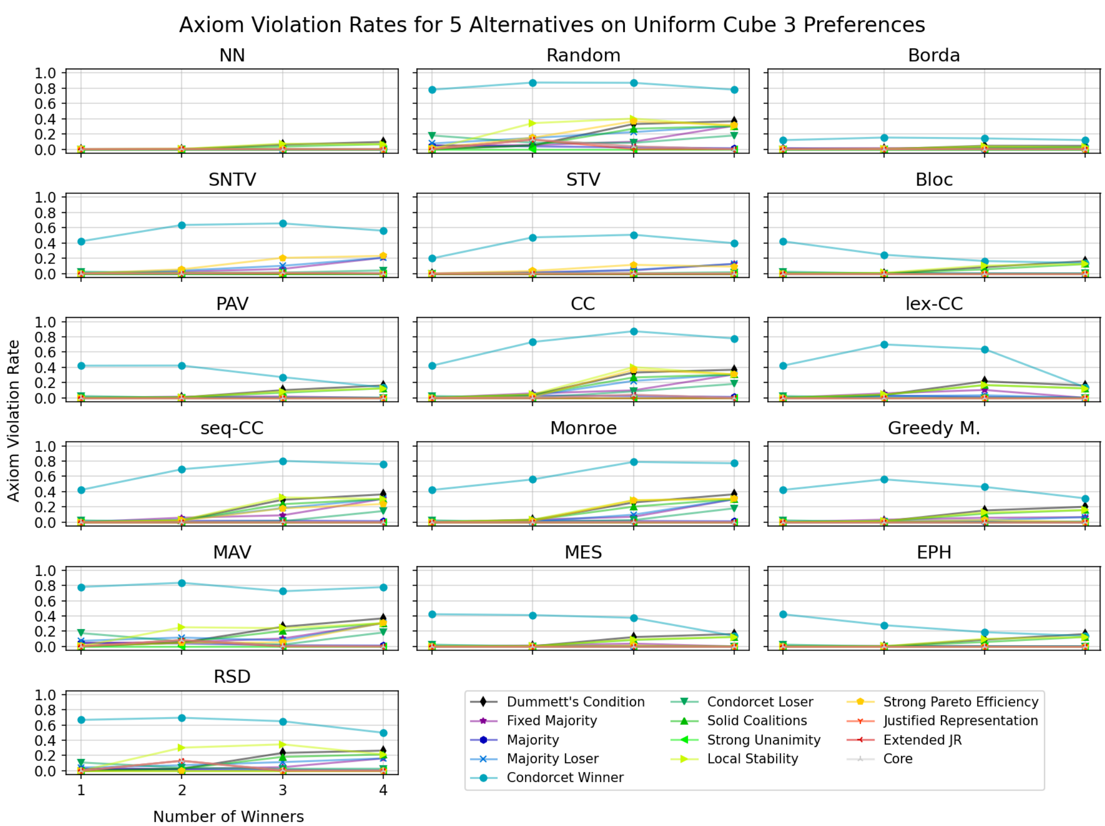


Figure D.31: Axiom violation rate for each rule on Uniform Cube 3 preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.009**	*.000*	*.000*	.002	**.009**	**.001**	*.000*	**0**	**0**	.042	*.000*	*.000*	*.000*	.028	.037
Borda	.016	.003	.007	.001	.016	.131	**0**	.017	**0**	.024	*.000*	*.000*	*.000*	.013	.017
EPH	.033	*.000*	.003	**0**	.033	.258	.006	.002	**0**	.063	**0**	**0**	**0**	.045	.055
SNTV	.067	**0**	.091	.119	.067	.566	.020	.073	**0**	.001	*.000*	.003	.003	**0**	.001
STV	.043	**0**	.048	.059	.043	.397	.005	.047	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.032	*.000*	.003	**0**	.032	.244	.006	**0**	**0**	.063	**0**	**0**	**0**	.045	.059
CC	.136	.016	.142	.175	.136	.703	.072	.115	**0**	.185	**0**	.011	.011	.154	.187
lex-CC	.063	.009	.015	**0**	.063	.474	.006	.041	**0**	.104	**0**	**0**	**0**	.082	.083
seq-CC	.120	.012	.130	.108	.120	.669	.044	.112	**0**	.172	**0**	*.000*	*.000*	.142	.165
Monroe	.116	.011	.105	.155	.116	.636	.058	.100	**0**	.161	**0**	.001	.001	.132	.153
Greedy M.	.058	.002	.027	.008	.058	.437	.008	.038	**0**	.093	**0**	**0**	**0**	.071	.075
PAV	.038	.001	.003	**0**	.038	.313	.006	.008	**0**	.067	**0**	**0**	**0**	.049	.053
MES	.042	.001	.005	.002	.042	.335	.006	.012	**0**	.072	**0**	**0**	**0**	.053	.055
MAV	.144	.027	.143	.107	.144	.777	.111	.128	**0**	.168	.022	.023	.023	.139	.203
RSD	.108	.015	.096	**0**	.108	.626	.052	.065	**0**	.134	.030	.031	.031	.109	.216
Random	.172	.029	.189	.211	.172	.826	.133	.126	**0**	.188	.031	.041	.041	.156	.263

Table D.33: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Uniform Cube 10 preferences.

D.1.16 5 Alternatives, Uniform Cube 10

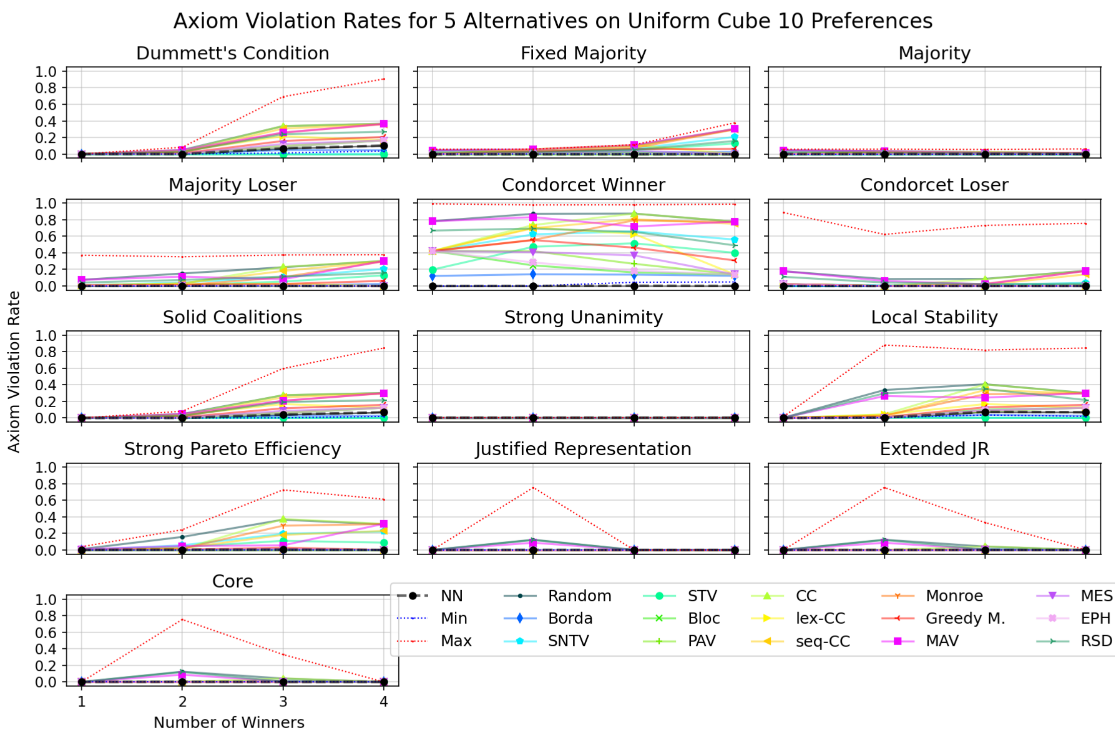


Figure D.32: Axiom violation rate for each axiom on Uniform Cube 10 preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.902	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.174	.902	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.567	.902	.586	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.362	.901	.379	.354	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.336	.901	.413	.385	.449	–	–	–	–	–	–	–	–	–	–	–
PAV	.373	.901	.421	.406	.459	.078	–	–	–	–	–	–	–	–	–	–
CC	.668	.903	.716	.530	.634	.439	.398	–	–	–	–	–	–	–	–	–
lex-CC	.453	.901	.498	.448	.513	.172	.104	.353	–	–	–	–	–	–	–	–
seq-CC	.653	.900	.673	.459	.615	.449	.415	.573	.393	–	–	–	–	–	–	–
Monroe	.615	.903	.664	.488	.584	.381	.352	.093	.373	.575	–	–	–	–	–	–
Greedy M.	.461	.901	.493	.397	.501	.245	.199	.449	.223	.330	.429	–	–	–	–	–
MAV	.822	.902	.837	.893	.845	.826	.800	.596	.753	.943	.606	.834	–	–	–	–
MES	.389	.901	.430	.371	.465	.141	.088	.448	.154	.361	.403	.149	.822	–	–	–
EPH	.344	.901	.410	.388	.448	.018	.065	.433	.162	.441	.376	.234	.824	.129	–	–
RSD	.693	.901	.707	.754	.733	.655	.660	.791	.682	.767	.768	.678	.868	.664	.656	–
Min	.027	.902	.174	.557	.350	.354	.389	.663	.467	.657	.610	.470	.820	.403	.361	.697
Max	1.222	.899	1.214	1.128	1.177	1.197	1.171	1.043	1.133	1.027	1.085	1.141	.996	1.165	1.192	1.046

Table D.34: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Uniform Cube 10 preferences.

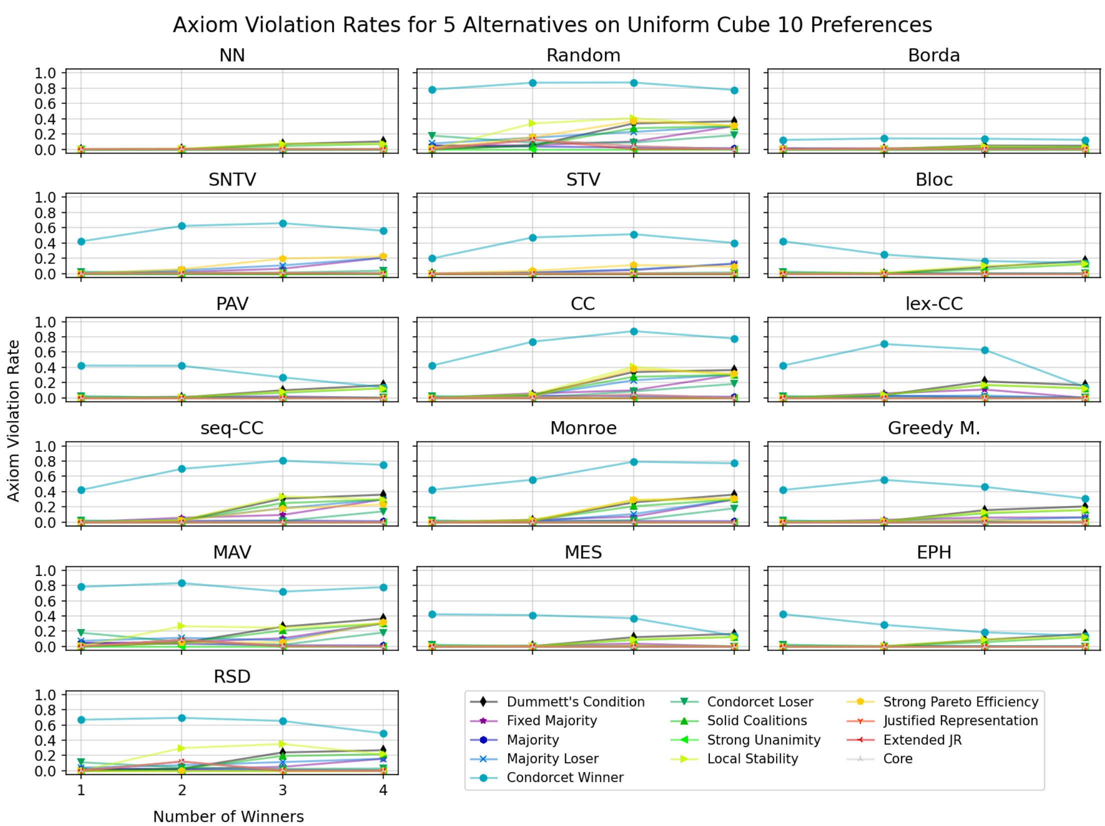


Figure D.33: Axiom violation rate for each rule on Uniform Cube 10 preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.006**	*.000*	**.001**	.002	**.006**	**.001**	*.000*	**0**	**0**	.027	*.000*	*.000*	*.000*	.018	.026
Borda	.012	.002	.007	.002	.012	.096	**0**	.016	**0**	.015	*.000*	*.000*	*.000*	.008	.010
EPH	.025	*.000*	.003	**0**	.025	.206	.005	.001	**0**	.043	**0**	**0**	**0**	.030	.037
SNTV	.079	**0**	.109	.133	.079	.483	.015	.112	.039	.044	*.000*	.041	.041	**0**	.003
STV	.034	**0**	.043	.053	.034	.297	.004	.039	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.024	*.000*	.002	**0**	.024	.198	.005	**0**	**0**	.042	**0**	**0**	**0**	.030	.039
CC	.165	.054	.164	.217	.165	.661	.069	.180	.058	.232	**0**	.072	.073	.167	.197
lex-CC	.046	.009	.013	**0**	.046	.362	.005	.033	**0**	.070	**0**	**0**	**0**	.054	.056
seq-CC	.149	.047	.156	.184	.149	.629	.044	.177	.059	.206	**0**	.061	.061	.144	.165
Monroe	.097	.016	.089	.125	.097	.524	.049	.085	**0**	.136	**0**	.001	.001	.111	.130
Greedy M.	.044	.002	.026	.006	.044	.347	.006	.033	**0**	.061	**0**	**0**	**0**	.045	.050
PAV	.028	.001	.003	**0**	.028	.243	.005	.006	**0**	.045	**0**	**0**	**0**	.032	.035
MES	.030	.001	.004	.001	.030	.257	.005	.009	**0**	.048	**0**	**0**	**0**	.034	.037
MAV	.135	.054	.136	.128	.135	.700	.097	.150	**0**	.147	.013	.015	.015	.118	.178
RSD	.088	.022	.085	**0**	.088	.537	.040	.070	**0**	.093	.022	.024	.024	.073	.161
Random	.217	.108	.213	.284	.217	.804	.126	.225	.071	.248	.049	.111	.111	.188	.285

Table D.35: Average Axiom Violation Rate for 5 alternatives and $1 \leq k < 5$ winners across Mixed preferences.

D.1.17 5 Alternatives, Mixed

Figure D.34: Axiom violation rate for each axiom on Mixed preferences with 5 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.900	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.162	.899	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.505	.900	.495	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.303	.901	.311	.337	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.288	.900	.333	.373	.378	–	–	–	–	–	–	–	–	–	–	–
PAV	.312	.900	.339	.385	.383	.053	–	–	–	–	–	–	–	–	–	–
CC	.612	.900	.653	.546	.596	.431	.405	–	–	–	–	–	–	–	–	–
lex-CC	.371	.900	.398	.417	.424	.125	.079	.367	–	–	–	–	–	–	–	–
seq-CC	.619	.900	.610	.410	.583	.451	.425	.607	.417	–	–	–	–	–	–	–
Monroe	.514	.900	.556	.474	.500	.331	.309	.125	.317	.561	–	–	–	–	–	–
Greedy M.	.398	.900	.402	.362	.425	.220	.190	.464	.213	.343	.384	–	–	–	–	–
MAV	.742	.900	.781	.859	.777	.759	.745	.559	.711	.935	.535	.785	–	–	–	–
MES	.333	.901	.342	.343	.392	.124	.086	.458	.136	.367	.363	.139	.772	–	–	–
EPH	.292	.900	.332	.374	.377	.011	.046	.428	.119	.446	.328	.214	.758	.116	–	–
RSD	.622	.899	.629	.695	.648	.592	.595	.757	.612	.737	.684	.610	.796	.597	.593	–
Min	.018	.901	.160	.499	.295	.297	.321	.611	.379	.619	.514	.402	.742	.340	.301	.624
Max	1.225	.900	1.213	1.139	1.185	1.194	1.178	1.050	1.153	1.027	1.110	1.154	1.030	1.172	1.191	1.075

Table D.36: Difference between rules for 5 alternatives with $1 \leq k < 5$ on Mixed preferences.

Figure D.35: Axiom violation rate for each rule on Mixed preferences with 5 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.011**	*.000*	*.000*	.003	**.011**	**.006**	*.000*	*.000*	**0**	.042	*.000*	*.000*	*.000*	.032	.055
Borda	.016	.002	.006	.003	.016	.113	**0**	.013	**0**	.027	*.000*	*.000*	*.000*	.019	.029
EPH	.032	*.000*	.002	*.000*	.032	.234	.003	.001	**0**	.060	*.000*	*.000*	*.000*	.049	.067
SNTV	.090	**0**	.103	.183	.090	.559	.011	.108	.045	.054	*.000*	.048	.049	**0**	.007
STV	.041	**0**	.039	.085	.041	.373	.002	.032	**0**	**0**	*.000*	*.000*	*.000*	**0**	*.000*
Bloc	.031	*.000*	.001	**0**	.031	.223	.003	**0**	**0**	.059	*.000*	*.000*	*.000*	.048	.073
CC	.179	.037	.154	.291	.179	.719	.041	.156	.060	.264	**0**	.077	.080	.199	.246
lex-CC	.054	.006	.009	**0**	.054	.402	.003	.027	**0**	.093	**0**	**0**	**0**	.076	.085
seq-CC	.167	.032	.147	.241	.167	.698	.032	.155	.060	.248	**0**	.072	.072	.184	.224
Monroe	.115	.007	.084	.188	.115	.599	.035	.072	**0**	.174	**0**	.003	.004	.149	.181
Greedy M.	.054	.002	.021	.010	.054	.400	.004	.026	**0**	.086	**0**	**0**	**0**	.070	.086
PAV	.035	.001	.002	**0**	.035	.268	.003	.005	**0**	.064	**0**	**0**	**0**	.052	.064
MES	.039	.001	.003	.002	.039	.301	.003	.008	**0**	.071	**0**	**0**	**0**	.057	.066
MAV	.148	.028	.128	.223	.148	.732	.063	.104	**0**	.178	.020	.026	.026	.149	.250
RSD	.096	.010	.068	**0**	.096	.571	.024	.046	**0**	.119	.023	.024	.025	.100	.234
Random	.226	.068	.189	.354	.226	.833	.084	.180	.071	.285	.048	.119	.123	.222	.359

Table D.37: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across all preferences.

D.2 6 Alternatives – All preferences

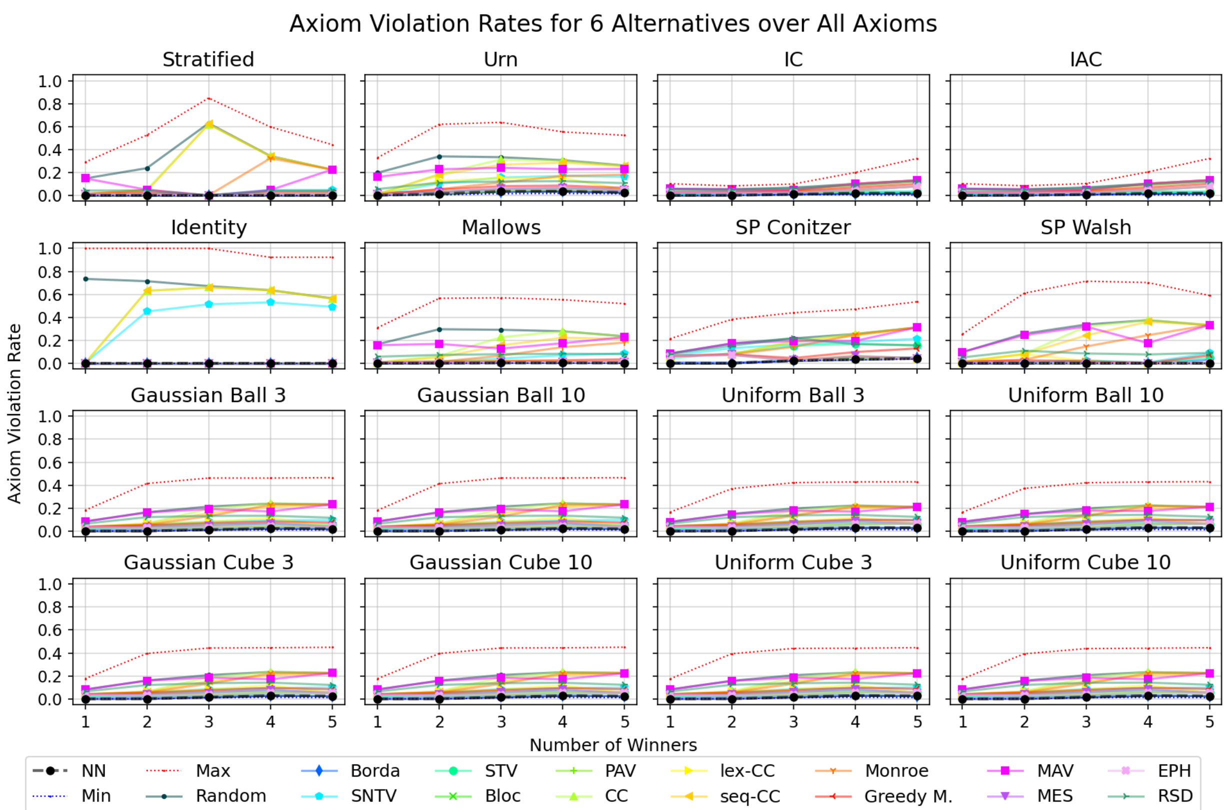


Figure D.36: Axiom violation rates for each rule under each preference distribution for 6 alternatives

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.710	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.139	.710	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.449	.710	.430	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.292	.710	.280	.283	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.218	.710	.241	.384	.332	–	–	–	–	–	–	–	–	–	–	–
PAV	.241	.709	.246	.395	.337	.058	–	–	–	–	–	–	–	–	–	–
CC	.547	.709	.572	.549	.545	.453	.429	–	–	–	–	–	–	–	–	–
lex-CC	.301	.710	.303	.422	.374	.135	.085	.397	–	–	–	–	–	–	–	–
seq-CC	.560	.710	.541	.437	.536	.461	.437	.609	.425	–	–	–	–	–	–	–
Monroe	.462	.709	.487	.491	.463	.366	.343	.111	.344	.569	–	–	–	–	–	–
Greedy M.	.330	.709	.321	.394	.377	.218	.188	.471	.203	.372	.398	–	–	–	–	–
MAV	.574	.709	.612	.683	.614	.598	.594	.361	.576	.791	.360	.629	–	–	–	–
MES	.263	.710	.258	.376	.344	.118	.077	.461	.116	.399	.377	.155	.610	–	–	–
EPH	.224	.710	.241	.386	.332	.016	.047	.449	.126	.455	.363	.211	.598	.109	–	–
RSD	.479	.710	.483	.572	.515	.460	.463	.631	.479	.615	.564	.479	.624	.465	.461	–
Min	.032	.710	.144	.442	.284	.233	.255	.545	.313	.563	.460	.338	.573	.275	.239	.483
Max	.970	.710	.953	.882	.926	.948	.936	.814	.912	.773	.867	.907	.821	.927	.945	.857

Table D.38: Difference between rules for 6 alternatives with $1 \leq k < 6$ averaged over all preference distributions.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	*.000*	**0**	**0**	**0**	*.000*	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Borda	.003	**0**	*.000*	**0**	.003	.041	**0**	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	**0**
EPH	.010	*.000*	**0**	**0**	.010	.131	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
SNTV	.025	**0**	.003	**0**	.025	.314	**0**	.004	**0**	**0**	**0**	**0**	**0**	**0**	**0**
STV	.010	**0**	*.000*	**0**	.010	.128	**0**	.001	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Bloc	.010	*.000*	**0**	**0**	.010	.131	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
CC	.250	.004	.006	.450	.250	.674	.036	.197	.190	.453	**0**	.230	.230	.380	.396
lex-CC	.010	*.000*	**0**	**0**	.010	.131	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
seq-CC	.247	.003	.006	.450	.247	.647	.035	.197	.190	.450	**0**	.231	.231	.377	.393
Monroe	.122	.003	.003	.254	.122	.477	.034	.005	**0**	.256	**0**	.033	.033	.243	.243
Greedy M.	.021	*.000*	.001	**0**	.021	.269	**0**	.001	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
PAV	.010	*.000*	**0**	**0**	.010	.131	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
MES	.010	*.000*	*.000*	**0**	.010	.132	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
MAV	.094	.004	.004	.200	.094	.548	.050	.009	**0**	.103	**0**	**0**	**0**	.102	.201
RSD	.034	.002	.002	**0**	.034	.426	**0**	.006	**0**	.002	**0**	**0**	**0**	.002	.002
Random	.316	.007	.007	.709	.316	.855	.080	.200	.190	.455	.052	.273	.273	.383	.622

Table D.39: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Stratified preferences.

D.2.1 6 Alternatives, Stratified

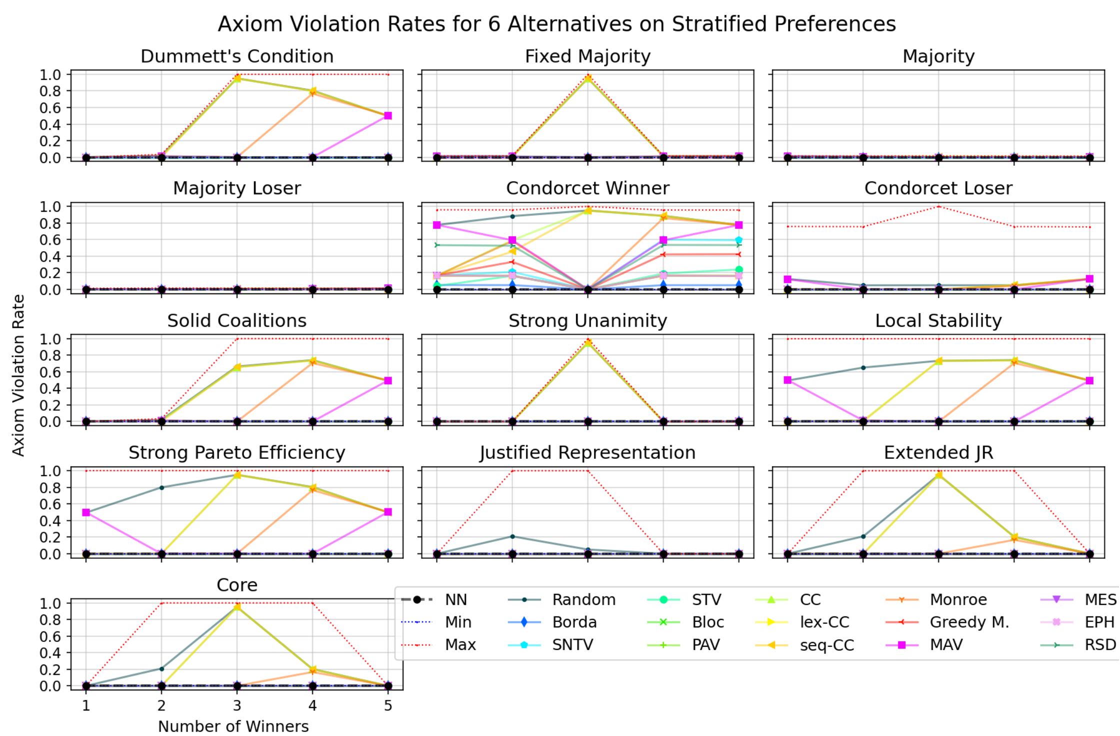


Figure D.37: Axiom violation rate for each axiom on Stratified preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.125	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.355	.867	.309	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.150	.866	.183	.304	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.195	.867	.187	.302	.255	–	–	–	–	–	–	–	–	–	–	–
PAV	.195	.867	.187	.302	.255	.000	–	–	–	–	–	–	–	–	–	–
CC	.541	.867	.561	.590	.550	.452	.452	–	–	–	–	–	–	–	–	–
lex-CC	.195	.867	.187	.302	.255	.000	.000	.452	–	–	–	–	–	–	–	–
seq-CC	.576	.867	.534	.353	.554	.504	.504	.737	.504	–	–	–	–	–	–	–
Monroe	.435	.867	.454	.483	.444	.346	.346	.109	.346	.634	–	–	–	–	–	–
Greedy M.	.305	.866	.279	.228	.315	.223	.223	.525	.223	.428	.418	–	–	–	–	–
MAV	.597	.869	.643	.710	.622	.614	.614	.522	.614	.917	.426	.657	–	–	–	–
MES	.212	.867	.180	.236	.265	.086	.086	.516	.086	.437	.409	.175	.650	–	–	–
EPH	.195	.867	.187	.302	.255	.000	.000	.452	.000	.504	.346	.223	.614	.086	–	–
RSD	.487	.866	.487	.502	.493	.480	.480	.703	.480	.699	.600	.490	.633	.478	.480	–
Min	.000	.866	.125	.355	.150	.195	.195	.541	.195	.576	.435	.305	.597	.212	.195	.487
Max	1.200	.867	1.200	1.200	1.200	1.200	1.200	1.022	1.200	.979	1.123	1.200	1.125	1.200	1.200	1.200

Table D.40: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Stratified preferences.

Figure D.38: Axiom violation rate for each rule on Stratified preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.022**	.001	.004	.028	**.022**	**.010**	*.000*	*.000*	**0**	.078	.004	.004	.004	.065	.092
Borda	.030	.017	.016	.035	.030	.144	**0**	.064	**0**	.040	*.000*	*.000*	*.000*	.032	.041
EPH	.034	.004	.004	*.000*	.034	.127	.002	.008	**0**	.105	*.000*	*.000*	*.000*	.091	.107
SNTV	.120	**0**	.204	.215	.120	.541	.002	.310	.027	.139	*.000*	.059	.063	**0**	*.000*
STV	.038	**0**	.063	.075	.038	.260	*.000*	.094	**0**	**0**	*.000*	*.000*	*.000*	**0**	*.000*
Bloc	.032	.001	**.001**	**0**	.032	.113	.002	**0**	**0**	.103	.001	.001	.001	.088	.106
CC	.214	.146	.229	.254	.214	.608	.036	.336	.030	.382	**0**	.083	.091	.278	.302
lex-CC	.097	.071	.069	**0**	.097	.385	.002	.180	**0**	.210	**0**	**0**	**0**	.166	.178
seq-CC	.199	.135	.220	.207	.199	.593	.033	.330	.029	.361	**0**	.069	.074	.259	.279
Monroe	.105	.052	.099	.101	.105	.388	.023	.136	**0**	.209	**0**	.001	.001	.168	.188
Greedy M.	.058	.024	.031	.008	.058	.247	.004	.061	**0**	.137	**0**	**0**	**0**	.116	.131
PAV	.039	.009	.008	**0**	.039	.155	.002	.018	**0**	.111	**0**	**0**	**0**	.096	.110
MES	.043	.012	.013	*.000*	.043	.181	.002	.027	**0**	.116	**0**	**0**	**0**	.100	.111
MAV	.218	.211	.208	.232	.218	.701	.059	.386	**0**	.344	.022	.039	.039	.259	.330
RSD	.104	.079	.079	**0**	.104	.374	.027	.121	**0**	.198	.023	.033	.033	.164	.215
Random	.287	.268	.264	.369	.287	.749	.081	.427	.038	.425	.071	.150	.161	.323	.409

Table D.41: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Urn preferences.

D.2.2 6 Alternatives, Urn

Figure D.39: Axiom violation rate for each axiom on Urn preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.869	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.244	.868	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.509	.869	.505	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.289	.868	.283	.359	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.220	.869	.277	.446	.312	–	–	–	–	–	–	–	–	–	–	–
PAV	.240	.869	.285	.445	.315	.045	–	–	–	–	–	–	–	–	–	–
CC	.532	.867	.601	.556	.544	.468	.449	–	–	–	–	–	–	–	–	–
lex-CC	.345	.869	.378	.458	.386	.176	.138	.402	–	–	–	–	–	–	–	–
seq-CC	.579	.867	.575	.410	.533	.463	.445	.619	.431	–	–	–	–	–	–	–
Monroe	.379	.868	.446	.493	.395	.303	.286	.190	.331	.553	–	–	–	–	–	–
Greedy M.	.303	.869	.328	.433	.339	.165	.145	.483	.216	.393	.332	–	–	–	–	–
MAV	.692	.866	.779	.852	.777	.769	.761	.467	.713	.938	.533	.791	–	–	–	–
MES	.258	.869	.294	.429	.320	.089	.060	.477	.159	.415	.315	.118	.778	–	–	–
EPH	.227	.869	.279	.445	.312	.014	.034	.464	.167	.457	.299	.159	.767	.080	–	–
RSD	.483	.868	.499	.639	.518	.446	.449	.668	.501	.648	.541	.465	.782	.452	.447	–
Min	.063	.869	.246	.498	.278	.232	.249	.534	.349	.577	.379	.307	.698	.265	.238	.485
Max	1.160	.865	1.139	1.042	1.121	1.143	1.129	.996	1.070	.947	1.084	1.099	.973	1.118	1.138	1.046

Table D.42: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Urn preferences.

Figure D.40: Axiom violation rate for each rule on Urn preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.009**	**0**	**0**	**0**	**.009**	**.014**	.001	**0**	**0**	.031	**0**	**0**	**0**	.022	.055
Borda	.012	**0**	**0**	**0**	.012	.119	**0**	**0**	**0**	.012	**0**	**0**	**0**	.009	.014
EPH	.040	**0**	**0**	**0**	.040	.324	.002	**0**	**0**	.064	**0**	**0**	**0**	.053	.073
SNTV	.033	**0**	**0**	**0**	.033	.420	.003	**0**	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
STV	.025	**0**	**0**	**0**	.025	.326	.001	**0**	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
Bloc	.040	**0**	**0**	**0**	.040	.323	.002	**0**	**0**	.064	**0**	**0**	**0**	.053	.074
CC	.068	**0**	**0**	**0**	.068	.519	.022	**0**	**0**	.111	**0**	**0**	**0**	.088	.141
lex-CC	.046	**0**	**0**	**0**	.046	.393	.002	**0**	**0**	.068	**0**	**0**	**0**	.056	.080
seq-CC	.062	**0**	**0**	**0**	.062	.499	.013	**0**	**0**	.096	**0**	**0**	**0**	.078	.120
Monroe	.068	**0**	**0**	**0**	.068	.519	.022	**0**	**0**	.111	**0**	**0**	**0**	.088	.141
Greedy M.	.053	**0**	**0**	**0**	.053	.445	.006	**0**	**0**	.079	**0**	**0**	**0**	.064	.099
PAV	.041	**0**	**0**	**0**	.041	.339	.002	**0**	**0**	.064	**0**	**0**	**0**	.053	.073
MES	.042	**0**	**0**	**0**	.042	.354	.002	**0**	**0**	.065	**0**	**0**	**0**	.053	.075
MAV	.078	**0**	**0**	**0**	.078	.623	.042	**0**	**0**	.111	**0**	**0**	**0**	.088	.154
RSD	.075	**0**	**0**	**0**	.075	.598	.028	**0**	**0**	.102	**0**	**0**	**0**	.083	.166
Random	.082	**0**	**0**	*.000*	.082	.635	.043	**0**	**0**	.113	**0**	**0**	**0**	.089	.187

Table D.43: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across IC preferences.

D.2.3 6 Alternatives, IC

Figure D.41: Axiom violation rate for each axiom on IC preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.867	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.358	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.622	.867	.530	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.465	.867	.430	.356	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.528	.866	.459	.495	.571	–	–	–	–	–	–	–	–	–	–	–
PAV	.541	.866	.467	.497	.572	.063	–	–	–	–	–	–	–	–	–	–
CC	.608	.866	.658	.570	.637	.405	.381	–	–	–	–	–	–	–	–	–
lex-CC	.570	.866	.506	.510	.589	.138	.085	.343	–	–	–	–	–	–	–	–
seq-CC	.721	.867	.615	.446	.640	.479	.443	.577	.426	–	–	–	–	–	–	–
Monroe	.608	.866	.658	.570	.637	.405	.381	.000	.342	.577	–	–	–	–	–	–
Greedy M.	.644	.867	.556	.440	.613	.379	.353	.506	.360	.309	.506	–	–	–	–	–
MAV	.662	.865	.853	.893	.824	.802	.805	.453	.783	.960	.453	.888	–	–	–	–
MES	.568	.866	.474	.422	.582	.212	.165	.474	.188	.332	.474	.258	.855	–	–	–
EPH	.531	.866	.459	.494	.570	.013	.057	.403	.133	.474	.403	.375	.805	.206	–	–
RSD	.815	.867	.803	.814	.818	.780	.780	.818	.786	.810	.817	.797	.862	.782	.781	–
Min	.038	.867	.360	.617	.459	.535	.548	.614	.576	.722	.613	.646	.668	.574	.538	.815
Max	1.107	.866	1.024	.976	1.028	1.010	1.001	1.010	.988	.870	1.010	.942	1.024	.978	1.007	.897

Table D.44: Difference between rules for 6 alternatives with $1 \leq k < 6$ on IC preferences.

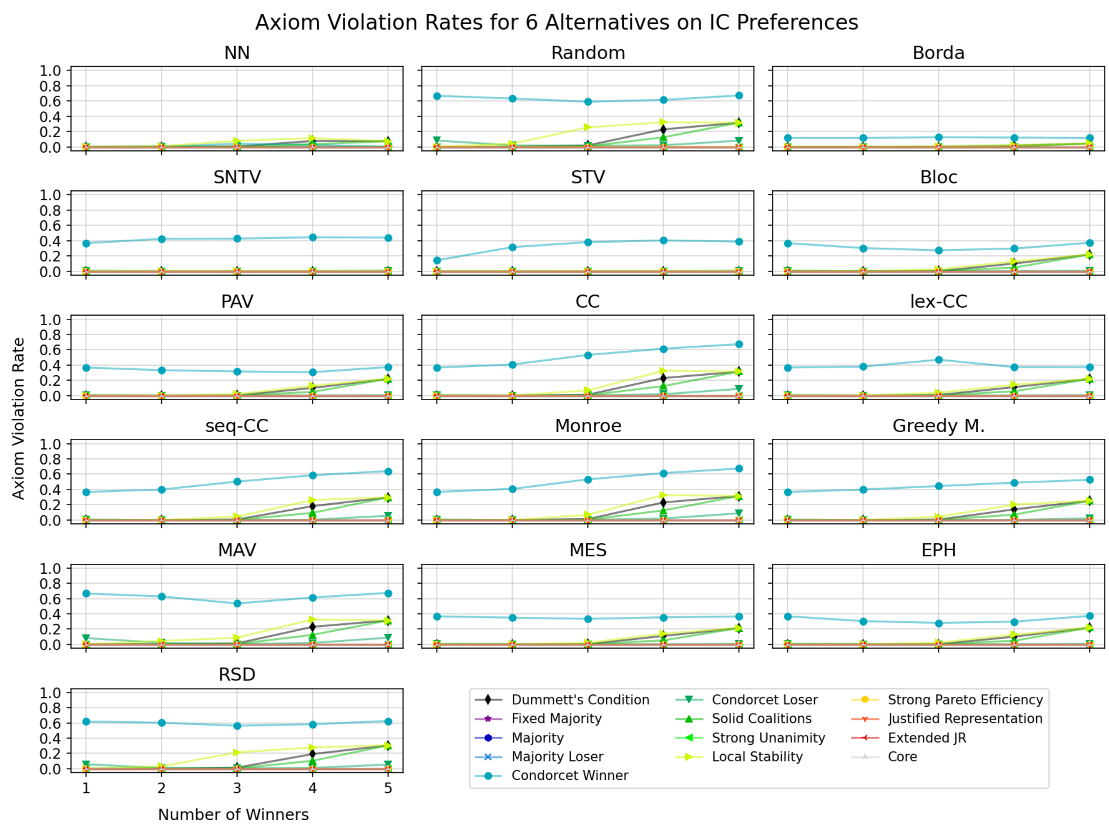


Figure D.42: Axiom violation rate for each rule on IC preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.009**	0	0	0	**.009**	**.009**	.001	0	0	.031	0	0	0	.023	.055
Borda	.012	0	0	0	.012	.116	0	0	0	.013	0	0	0	.010	.015
EPH	.040	0	0	0	.040	.316	.002	0	0	.066	0	0	0	.054	.077
SNTV	.032	0	0	*.000*	.032	.415	.003	0	0	0	0	0	0	0	*.000*
STV	.025	0	0	0	.025	.324	.001	0	0	0	0	0	0	0	*.000*
Bloc	.039	0	0	0	.039	.314	.002	0	0	.065	0	0	0	.053	.077
CC	.069	0	0	*.000*	.069	.514	.023	0	0	.114	0	0	0	.093	.147
lex-CC	.046	0	0	0	.046	.385	.002	0	0	.069	0	0	0	.055	.082
seq-CC	.063	0	0	0	.063	.494	.014	0	0	.102	0	0	0	.082	.124
Monroe	.069	0	0	*.000*	.069	.514	.023	0	0	.114	0	0	0	.093	.147
Greedy M.	.053	0	0	0	.053	.436	.006	0	0	.083	0	0	0	.067	.100
PAV	.040	0	0	0	.040	.329	.002	0	0	.065	0	0	0	.053	.075
MES	.042	0	0	0	.042	.346	.002	0	0	.066	0	0	0	.054	.075
MAV	.079	0	0	*.000*	.079	.621	.044	0	0	.114	*.000*	*.000*	*.000*	.093	.160
RSD	.076	0	0	0	.076	.594	.030	0	0	.106	0	0	0	.084	.172
Random	.083	0	0	*.000*	.083	.629	.044	0	0	.117	*.000*	*.000*	*.000*	.094	.194

Table D.45: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across IAC preferences.

D.2.4 6 Alternatives, IAC

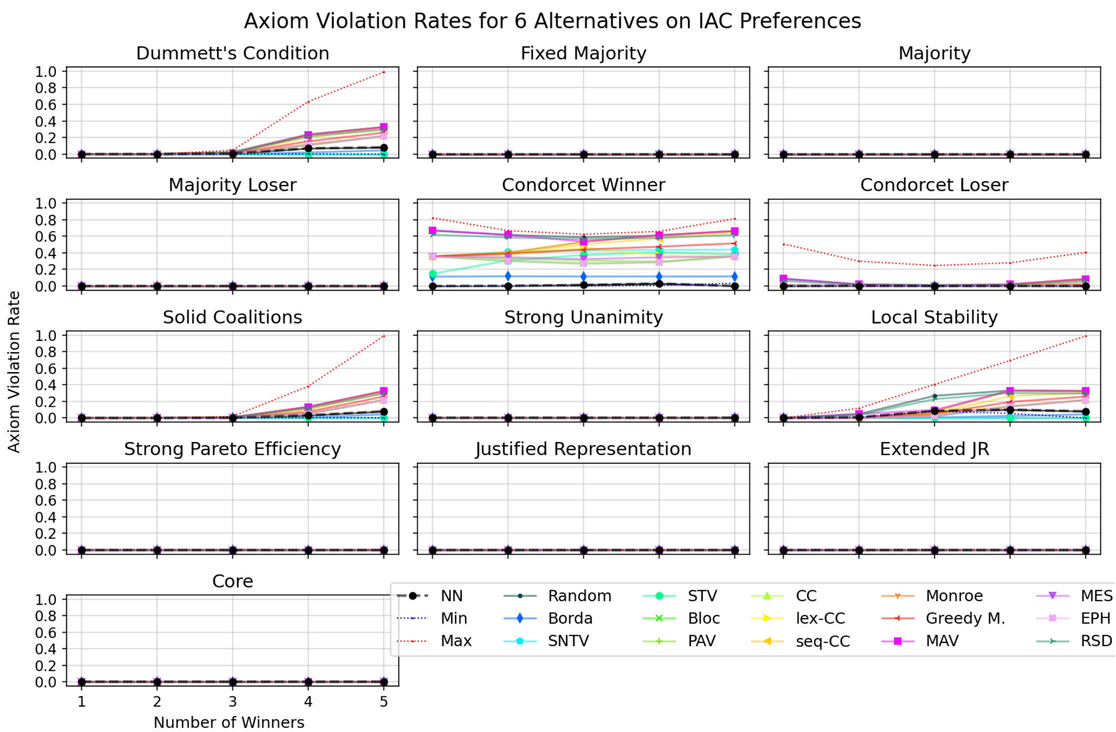


Figure D.43: Axiom violation rate for each axiom on IAC preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.867	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.361	.867	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.617	.868	.531	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.467	.867	.436	.352	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.523	.866	.456	.488	.564	–	–	–	–	–	–	–	–	–	–	–
PAV	.535	.866	.464	.491	.565	.063	–	–	–	–	–	–	–	–	–	–
CC	.606	.865	.659	.566	.634	.409	.386	–	–	–	–	–	–	–	–	–
lex-CC	.564	.867	.503	.503	.581	.139	.087	.346	–	–	–	–	–	–	–	–
seq-CC	.718	.867	.618	.446	.636	.477	.441	.575	.423	–	–	–	–	–	–	–
Monroe	.605	.865	.658	.566	.634	.409	.386	.000	.346	.575	–	–	–	–	–	–
Greedy M.	.638	.867	.557	.441	.608	.372	.347	.505	.355	.309	.504	–	–	–	–	–
MAV	.668	.867	.858	.892	.827	.806	.809	.452	.787	.960	.453	.889	–	–	–	–
MES	.563	.867	.474	.421	.578	.206	.160	.474	.181	.336	.474	.258	.857	–	–	–
EPH	.526	.866	.457	.489	.565	.013	.057	.407	.134	.472	.407	.368	.808	.200	–	–
RSD	.817	.868	.804	.818	.820	.784	.784	.821	.789	.812	.821	.797	.864	.785	.784	–
Min	.037	.867	.364	.612	.461	.530	.543	.609	.572	.719	.609	.640	.672	.570	.533	.818
Max	1.108	.865	1.025	.976	1.028	1.011	1.001	1.009	.988	.869	1.009	.943	1.019	.978	1.008	.895

Table D.46: Difference between rules for 6 alternatives with $1 \leq k < 6$ on IAC preferences.

Figure D.44: Axiom violation rate for each rule on IAC preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Borda	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
EPH	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
SNTV	.397	**0**	.400	.681	.397	.681	**0**	.681	.681	.681	**0**	.681	.681	**0**	**0**
STV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Bloc	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
CC	.499	.293	.445	.730	.499	.730	.047	.730	.730	.730	**0**	.730	.730	.293	.293
lex-CC	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
seq-CC	.497	.289	.443	.730	.497	.730	.046	.730	.730	.730	**0**	.730	.730	.289	.289
Monroe	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Greedy M.	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
PAV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MES	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MAV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
RSD	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Random	.665	.499	.499	.897	.665	.897	.104	.897	.897	.897	.256	.897	.897	.499	.499

Table D.47: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Identity preferences.

D.2.5 6 Alternatives, Identity

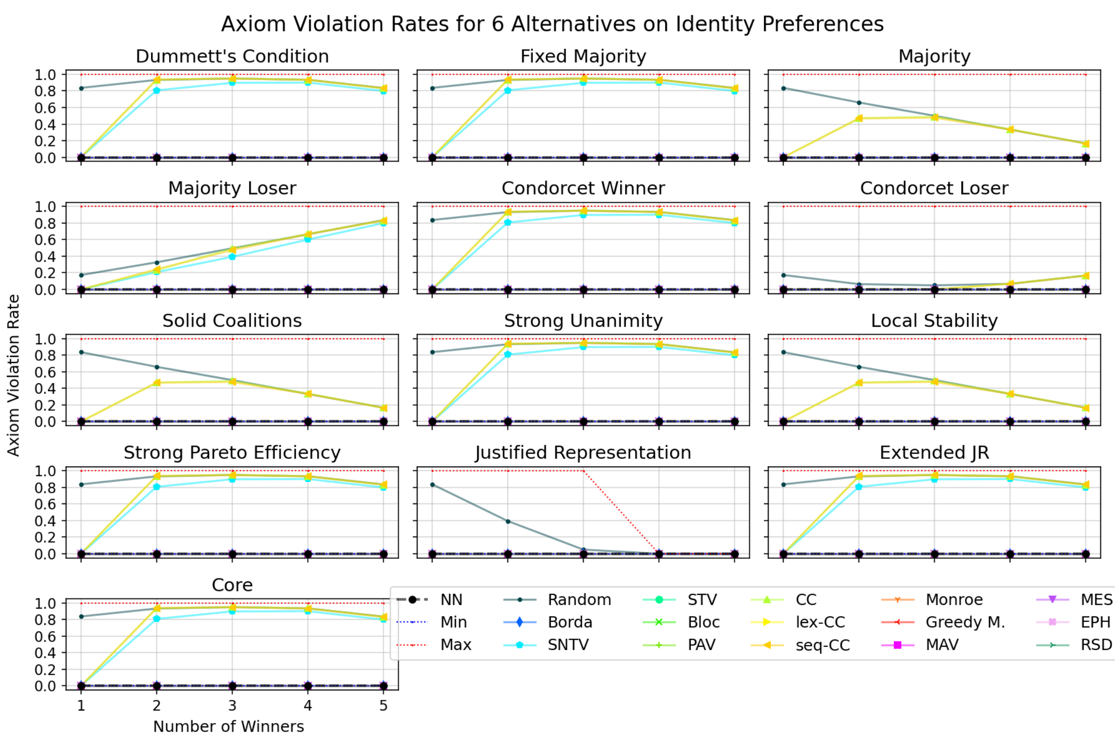


Figure D.45: Axiom violation rate for each axiom on Identity preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.867	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.000	.867	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.440	.868	.440	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.000	.867	.000	.440	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.000	.867	.000	.440	.000	–	–	–	–	–	–	–	–	–	–	–
PAV	.000	.867	.000	.440	.000	.000	–	–	–	–	–	–	–	–	–	–
CC	.491	.867	.491	.708	.491	.491	.491	–	–	–	–	–	–	–	–	–
lex-CC	.000	.867	.000	.440	.000	.000	.000	.491	–	–	–	–	–	–	–	–
seq-CC	.490	.868	.490	.145	.490	.490	.490	.780	.490	–	–	–	–	–	–	–
Monroe	.000	.867	.000	.440	.000	.000	.000	.491	.000	.490	–	–	–	–	–	–
Greedy M.	.000	.867	.000	.440	.000	.000	.000	.491	.000	.490	.000	–	–	–	–	–
MAV	.000	.867	.000	.440	.000	.000	.000	.491	.000	.490	.000	.000	–	–	–	–
MES	.000	.867	.000	.440	.000	.000	.000	.491	.000	.490	.000	.000	.000	–	–	–
EPH	.000	.867	.000	.440	.000	.000	.000	.491	.000	.490	.000	.000	.000	.000	–	–
RSD	.000	.867	.000	.440	.000	.000	.000	.491	.000	.490	.000	.000	.000	.000	.000	–
Min	.000	.867	.000	.440	.000	.000	.000	.491	.000	.490	.000	.000	.000	.000	.000	.000
Max	1.200	.865	1.200	1.040	1.200	1.200	1.200	.955	1.200	.957	1.200	1.200	1.200	1.200	1.200	1.200

Table D.48: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Identity preferences.

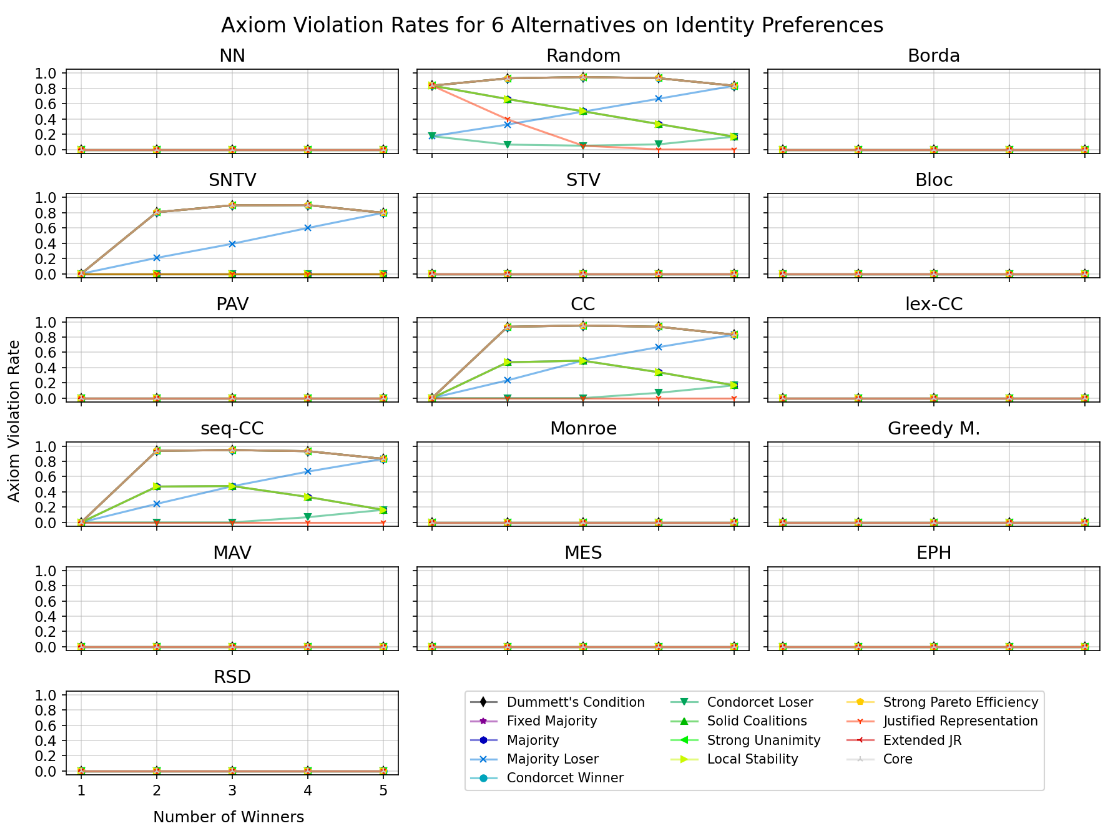


Figure D.46: Axiom violation rate for each rule on Identity preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.002**	**0**	*.000*	**0**	**.002**	**.008**	*.000*	**0**	**0**	.004	*.000*	*.000*	*.000*	.003	.008
Borda	.003	*.000*	*.000*	**0**	.003	.039	**0**	*.000*	**0**	.001	**0**	**0**	**0**	.001	.002
EPH	.009	**0**	**0**	**0**	.009	.090	*.000*	**0**	**0**	.009	**0**	**0**	**0**	.007	.010
SNTV	.044	**0**	.074	.034	.044	.292	*.000*	.111	.007	.028	**0**	.010	.012	**0**	*.000*
STV	.007	**0**	*.000*	**0**	.007	.095	*.000*	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Bloc	.009	**0**	**0**	**0**	.009	.090	*.000*	**0**	**0**	.009	**0**	**0**	**0**	.007	.010
CC	.162	.089	.173	.157	.162	.605	.042	.242	.008	.267	**0**	.056	.063	.190	.213
lex-CC	.016	*.000*	*.000*	**0**	.016	.166	*.000*	.015	**0**	.010	**0**	**0**	**0**	.008	.012
seq-CC	.130	.040	.170	.125	.130	.575	.023	.233	.008	.190	**0**	.043	.048	.107	.122
Monroe	.081	.027	.057	.032	.081	.418	.031	.082	**0**	.146	**0**	*.000*	*.000*	.122	.142
Greedy M.	.020	*.000*	.005	**0**	.020	.196	.001	.006	**0**	.016	**0**	**0**	**0**	.012	.018
PAV	.009	**0**	**0**	**0**	.009	.096	*.000*	**0**	**0**	.008	**0**	**0**	**0**	.007	.009
MES	.010	**0**	**0**	**0**	.010	.103	*.000*	**0**	**0**	.009	**0**	**0**	**0**	.007	.010
MAV	.171	.141	.137	.131	.171	.761	.075	.267	**0**	.216	.020	.026	.026	.169	.254
RSD	.074	.037	.039	**0**	.074	.509	.014	.101	**0**	.078	.002	.003	.003	.065	.112
Random	.253	.226	.228	.246	.253	.859	.092	.346	.010	.349	.051	.113	.124	.273	.375

Table D.49: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Mallows preferences.

D.2.6 6 Alternatives, Mallows

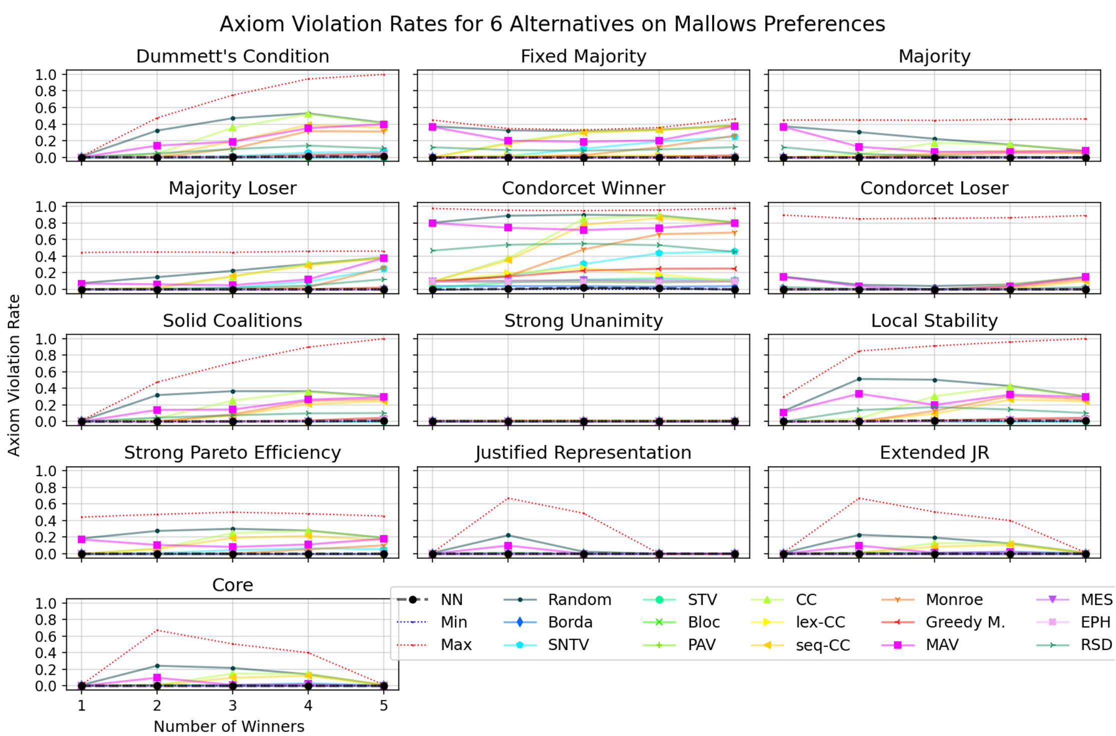


Figure D.47: Axiom violation rate for each axiom on Mallows preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.865	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.078	.865	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.258	.866	.242	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.098	.865	.102	.200	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.118	.865	.109	.233	.142	–	–	–	–	–	–	–	–	–	–	–
PAV	.122	.865	.111	.233	.142	.016	–	–	–	–	–	–	–	–	–	–
CC	.464	.866	.472	.482	.466	.412	.406	–	–	–	–	–	–	–	–	–
lex-CC	.152	.865	.144	.250	.165	.058	.044	.390	–	–	–	–	–	–	–	–
seq-CC	.456	.865	.438	.350	.441	.406	.397	.569	.384	–	–	–	–	–	–	–
Monroe	.339	.866	.347	.384	.341	.287	.281	.136	.276	.500	–	–	–	–	–	–
Greedy M.	.190	.865	.177	.238	.190	.135	.129	.439	.142	.360	.316	–	–	–	–	–
MAV	.741	.866	.770	.817	.764	.759	.760	.474	.751	.929	.515	.785	–	–	–	–
MES	.129	.866	.114	.215	.145	.053	.041	.429	.068	.370	.303	.107	.772	–	–	–
EPH	.119	.865	.109	.233	.142	.002	.015	.412	.057	.405	.287	.135	.760	.052	–	–
RSD	.484	.866	.482	.550	.487	.478	.478	.667	.485	.649	.567	.491	.787	.478	.478	–
Min	.011	.865	.078	.258	.097	.120	.123	.465	.153	.456	.340	.191	.742	.130	.120	.484
Max	1.181	.865	1.169	1.146	1.167	1.166	1.164	1.018	1.153	1.028	1.084	1.148	.969	1.161	1.166	1.077

Table D.50: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Mallows preferences.

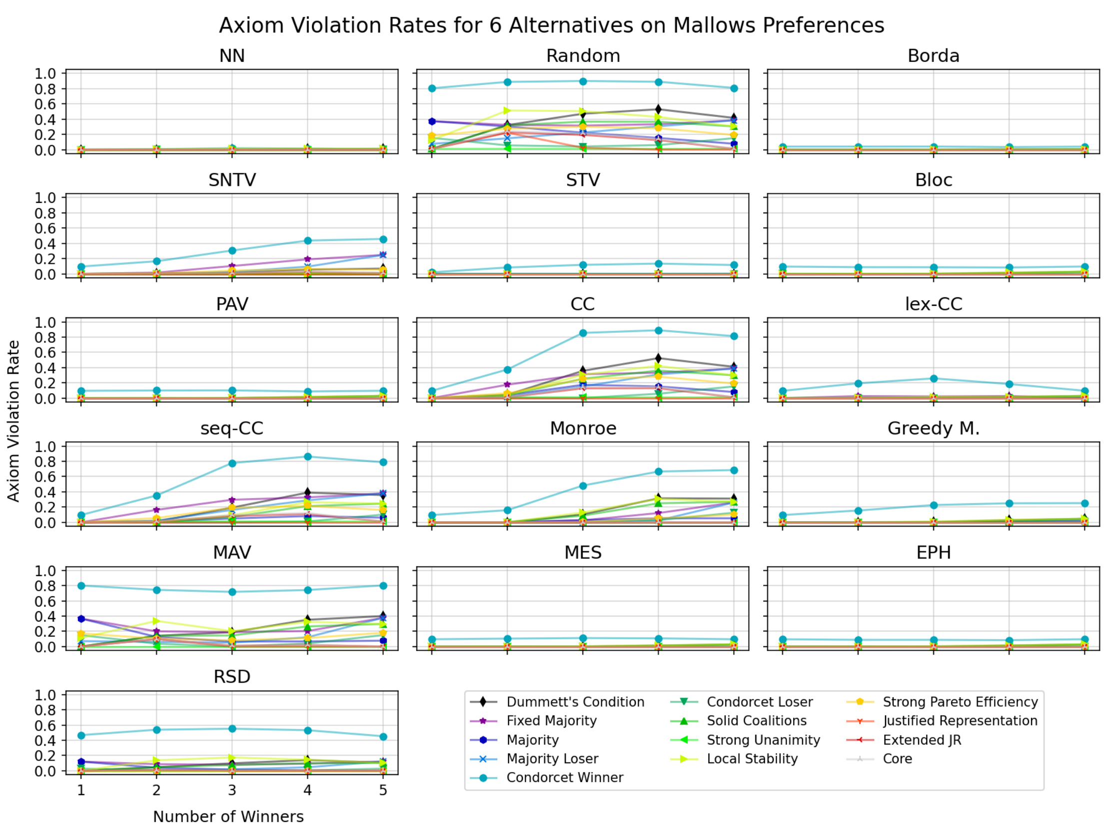


Figure D.48: Axiom violation rate for each rule on Mallows preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.020**	**0**	*.000*	*.000*	**.020**	*.000*	**0**	**0**	**0**	.068	*.000*	*.000*	*.000*	.052	.135
Borda	.028	**0**	.017	*.000*	.028	.086	**0**	.047	**0**	.068	**0**	**0**	**0**	.047	.094
EPH	.047	**0**	.011	**0**	.047	.320	.011	.005	**0**	.070	**0**	**0**	**0**	.053	.136
SNTV	.152	**0**	.328	.546	.152	.821	.044	.159	**0**	*.000*	.001	.001	.014	**0**	.055
STV	.122	**0**	.221	.476	.122	.746	.002	.132	**0**	**0**	**0**	*.000*	.003	**0**	**.002**
Bloc	.047	**0**	.011	**0**	.047	.308	.011	**0**	**0**	.068	*.000*	*.000*	*.000*	.052	.155
CC	.180	**0**	.360	.470	.180	.876	.052	.183	**0**	.113	**0**	.002	.014	.090	.180
lex-CC	.055	**0**	.011	**0**	.055	.401	.011	.038	**0**	.085	**0**	**0**	**0**	.063	.109
seq-CC	.175	**0**	.349	.450	.175	.864	.043	.183	**0**	.112	**0**	.001	.001	.089	.185
Monroe	.169	**0**	.299	.446	.169	.850	.051	.183	**0**	.111	**0**	*.000*	.012	.089	.150
Greedy M.	.082	**0**	.096	.056	.082	.509	.011	.096	**0**	.099	**0**	**0**	**0**	.077	.126
PAV	.051	**0**	.011	**0**	.051	.365	.011	.027	**0**	.079	**0**	**0**	**0**	.059	.111
MES	.054	**0**	.011	**0**	.054	.390	.011	.037	**0**	.085	**0**	**0**	**0**	.062	.108
MAV	.192	**0**	.384	.553	.192	.858	.069	.186	**0**	.113	.012	.013	.013	.090	.201
RSD	.155	**0**	.256	**0**	.155	.691	.032	.094	**0**	.101	.084	.084	.084	.081	.503
Random	.208	**0**	.444	.544	.208	.883	.092	.183	**0**	.112	.023	.024	.038	.089	.269

Table D.51: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across SP Conitzer preferences.

D.2.7 6 Alternatives, SP Conitzer

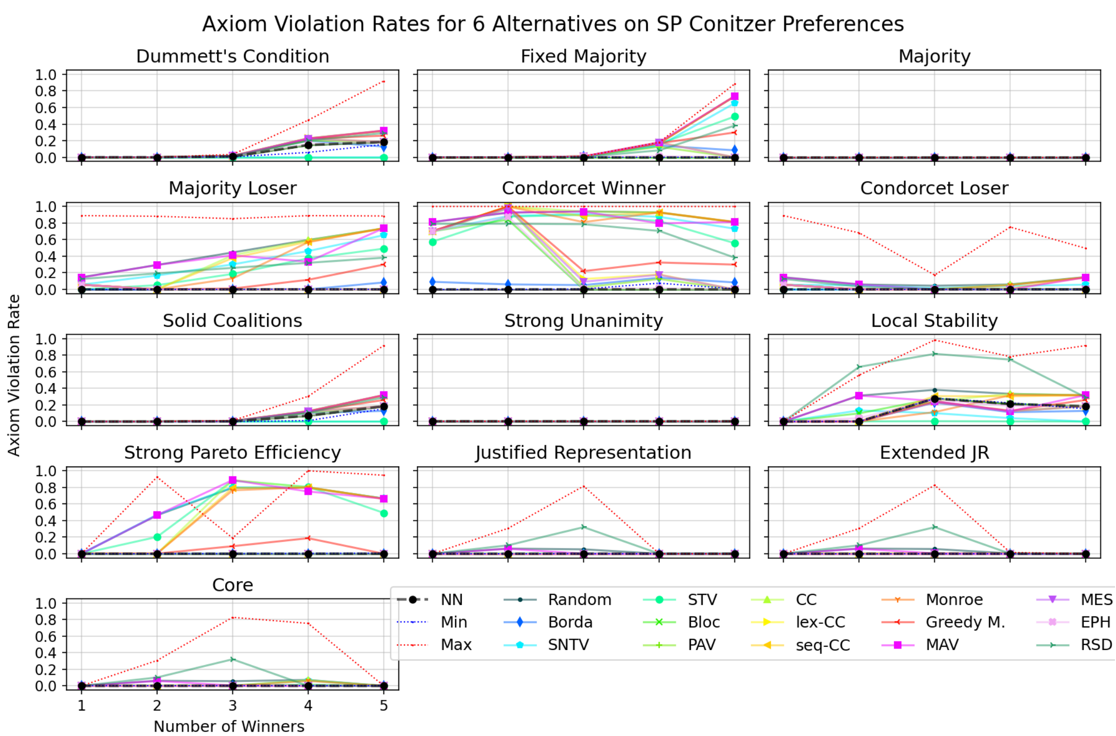


Figure D.49: Axiom violation rate for each axiom on SP Conitzer preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.131	.865	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.773	.866	.769	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.645	.867	.627	.469	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.415	.865	.478	.539	.602	–	–	–	–	–	–	–	–	–	–	–
PAV	.460	.865	.479	.564	.612	.064	–	–	–	–	–	–	–	–	–	–
CC	.826	.866	.841	.619	.703	.431	.394	–	–	–	–	–	–	–	–	–
lex-CC	.475	.865	.486	.571	.617	.081	.016	.391	–	–	–	–	–	–	–	–
seq-CC	.828	.865	.820	.500	.726	.507	.472	.649	.470	–	–	–	–	–	–	–
Monroe	.798	.866	.815	.606	.690	.404	.368	.052	.367	.620	–	–	–	–	–	–
Greedy M.	.573	.864	.556	.502	.648	.251	.204	.508	.194	.353	.486	–	–	–	–	–
MAV	.791	.866	.809	.903	.827	.782	.782	.535	.780	1.011	.579	.850	–	–	–	–
MES	.497	.864	.481	.494	.623	.174	.121	.488	.116	.377	.463	.103	.821	–	–	–
EPH	.431	.865	.485	.549	.606	.023	.049	.421	.065	.494	.394	.237	.784	.159	–	–
RSD	.696	.866	.697	.813	.789	.659	.662	.816	.664	.805	.803	.674	.828	.662	.660	–
Min	.016	.866	.137	.769	.639	.431	.475	.825	.490	.827	.797	.579	.793	.501	.447	.697
Max	1.121	.866	1.108	.986	1.050	1.096	1.091	.984	1.088	.951	1.010	1.076	.951	1.085	1.095	.989

Table D.52: Difference between rules for 6 alternatives with $1 \leq k < 6$ on SP Conitzer preferences.

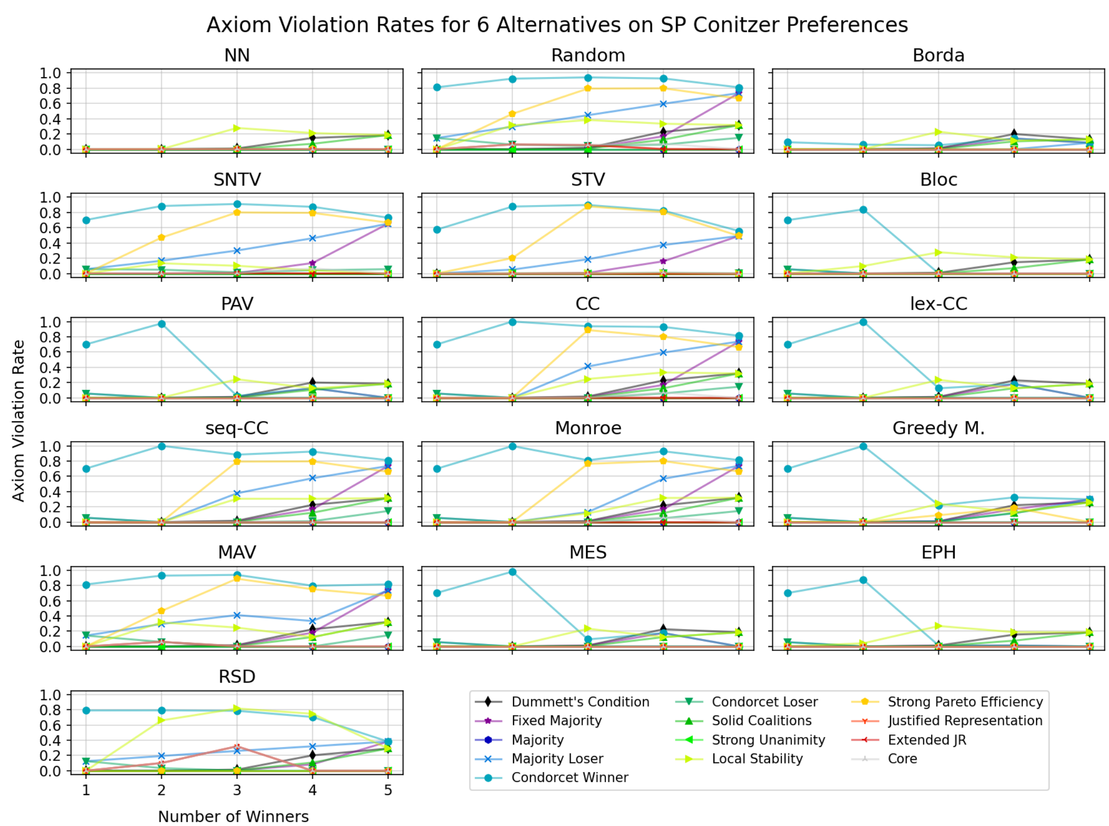


Figure D.50: Axiom violation rate for each rule on SP Conitzer preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	*.000*	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	**0**	*.000*
Borda	.008	**0**	.025	**0**	.008	.057	**0**	.025	**0**	*.000*	**0**	**0**	**0**	**0**	**0**
EPH	.003	**0**	**0**	**0**	.003	.035	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	**0**	*.000*
SNTV	.029	**0**	.078	.012	.029	.204	**0**	.081	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
STV	.014	**0**	.040	.005	.014	.103	**0**	.041	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
Bloc	.003	**0**	**0**	**0**	.003	.036	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	**0**	*.000*
CC	.224	.001	.349	.471	.224	.772	.043	.246	**0**	.289	**0**	.091	.094	.257	.304
lex-CC	.022	*.000*	**0**	**0**	.022	.280	**0**	.009	**0**	.001	**0**	**0**	**0**	.001	.001
seq-CC	.206	.001	.337	.457	.206	.756	.030	.246	**0**	.254	**0**	.053	.053	.227	.262
Monroe	.153	.001	.199	.354	.153	.571	.034	.211	**0**	.216	**0**	.006	.009	.194	.198
Greedy M.	.025	**0**	.060	*.000*	.025	.207	**0**	.062	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
PAV	.003	**0**	**0**	**0**	.003	.042	**0**	*.000*	**0**	*.000*	**0**	**0**	**0**	**0**	*.000*
MES	.004	**0**	**0**	**0**	.004	.050	**0**	*.000*	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
MAV	.234	.002	.381	.560	.234	.853	.072	.230	**0**	.183	.062	.115	.115	.157	.315
RSD	.082	.001	.149	**0**	.082	.531	.004	.124	**0**	.031	.024	.024	.024	.021	.136
Random	.279	.002	.442	.563	.279	.886	.095	.245	**0**	.294	.080	.159	.161	.263	.440

Table D.53: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across SP Walsh preferences.

D.2.8 6 Alternatives, SP Walsh

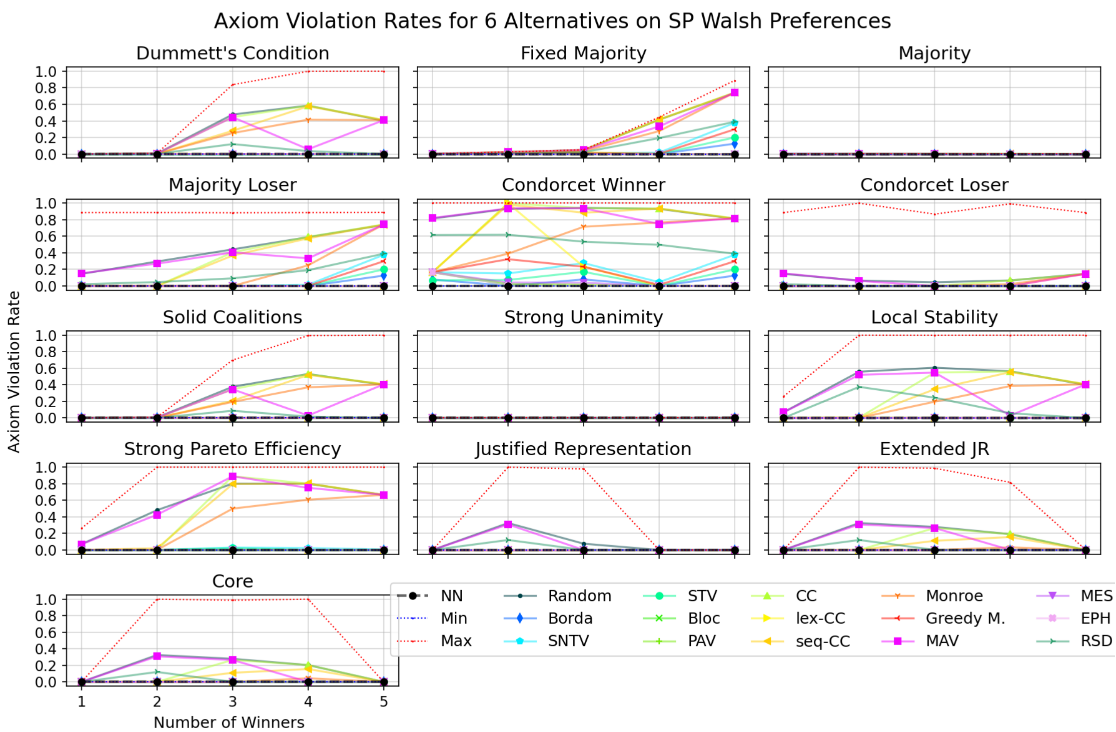


Figure D.51: Axiom violation rate for each axiom on SP Walsh preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.102	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.222	.866	.188	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.105	.865	.129	.173	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.087	.866	.136	.163	.145	–	–	–	–	–	–	–	–	–	–	–
PAV	.093	.866	.135	.163	.144	.008	–	–	–	–	–	–	–	–	–	–
CC	.576	.865	.585	.535	.564	.490	.489	–	–	–	–	–	–	–	–	–
lex-CC	.204	.867	.240	.247	.238	.119	.112	.392	–	–	–	–	–	–	–	–
seq-CC	.572	.867	.560	.466	.551	.512	.511	.690	.511	–	–	–	–	–	–	–
Monroe	.426	.866	.436	.391	.417	.340	.340	.196	.386	.647	–	–	–	–	–	–
Greedy M.	.218	.866	.211	.158	.209	.159	.155	.525	.207	.433	.410	–	–	–	–	–
MAV	.791	.865	.807	.821	.795	.792	.793	.536	.782	1.002	.601	.821	–	–	–	–
MES	.124	.866	.132	.127	.150	.066	.060	.528	.164	.468	.380	.113	.811	–	–	–
EPH	.087	.866	.136	.163	.145	.001	.008	.490	.119	.512	.340	.159	.792	.065	–	–
RSD	.519	.867	.521	.533	.525	.517	.517	.759	.548	.743	.673	.533	.815	.517	.517	–
Min	.000	.866	.102	.222	.105	.087	.093	.576	.204	.572	.426	.218	.791	.124	.087	.519
Max	1.198	.867	1.188	1.175	1.183	1.197	1.192	1.010	1.178	1.017	1.077	1.177	.952	1.189	1.197	1.129

Table D.54: Difference between rules for 6 alternatives with $1 \leq k < 6$ on SP Walsh preferences.

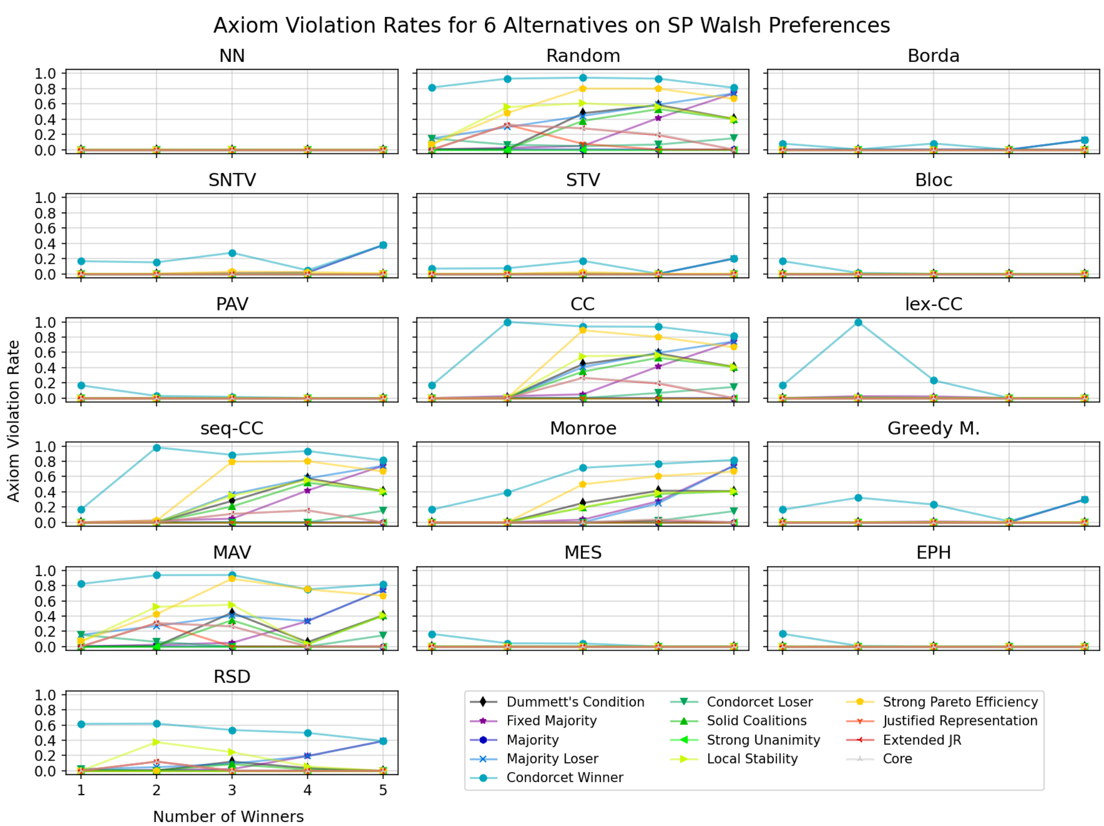


Figure D.52: Axiom violation rate for each rule on SP Walsh preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.011**	*.000*	*.000*	.001	**.011**	**.006**	*.000*	**0**	**0**	.047	*.000*	*.000*	*.000*	.035	.053
Borda	.018	.001	.005	.001	.018	.138	**0**	.010	**0**	.032	*.000*	*.000*	*.000*	.022	.031
EPH	.035	*.000*	.001	*.000*	.035	.265	.003	.001	**0**	.064	**0**	**0**	**0**	.051	.067
SNTV	.073	**0**	.078	.165	.073	.632	.012	.056	**0**	.001	*.000*	.002	.003	**0**	.005
STV	.049	**0**	.040	.086	.049	.467	.002	.035	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.034	*.000*	.001	**0**	.034	.249	.003	**0**	**0**	.062	*.000*	*.000*	*.000*	.049	.073
CC	.154	.007	.135	.272	.154	.772	.044	.086	**0**	.225	**0**	.008	.009	.194	.249
lex-CC	.067	.004	.010	**0**	.067	.517	.003	.029	**0**	.117	**0**	**0**	**0**	.095	.099
seq-CC	.139	.006	.127	.188	.139	.746	.034	.085	**0**	.209	**0**	.004	.004	.179	.225
Monroe	.136	.005	.102	.230	.136	.717	.043	.081	**0**	.203	**0**	.002	.003	.175	.212
Greedy M.	.063	.001	.021	.010	.063	.478	.003	.028	**0**	.101	**0**	**0**	**0**	.083	.094
PAV	.040	*.000*	.001	**0**	.040	.316	.003	.004	**0**	.070	**0**	**0**	**0**	.056	.064
MES	.046	*.000*	.002	.003	.046	.364	.003	.009	**0**	.080	**0**	**0**	**0**	.063	.069
MAV	.168	.012	.143	.237	.168	.844	.076	.091	**0**	.209	.027	.033	.033	.179	.301
RSD	.114	.006	.082	**0**	.114	.665	.029	.044	**0**	.149	.029	.030	.031	.126	.285
Random	.186	.014	.172	.295	.186	.872	.090	.090	**0**	.227	.032	.042	.043	.196	.347

Table D.55: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Gaussian Ball 3 preferences.

D.2.9 6 Alternatives, Gaussian Ball 3

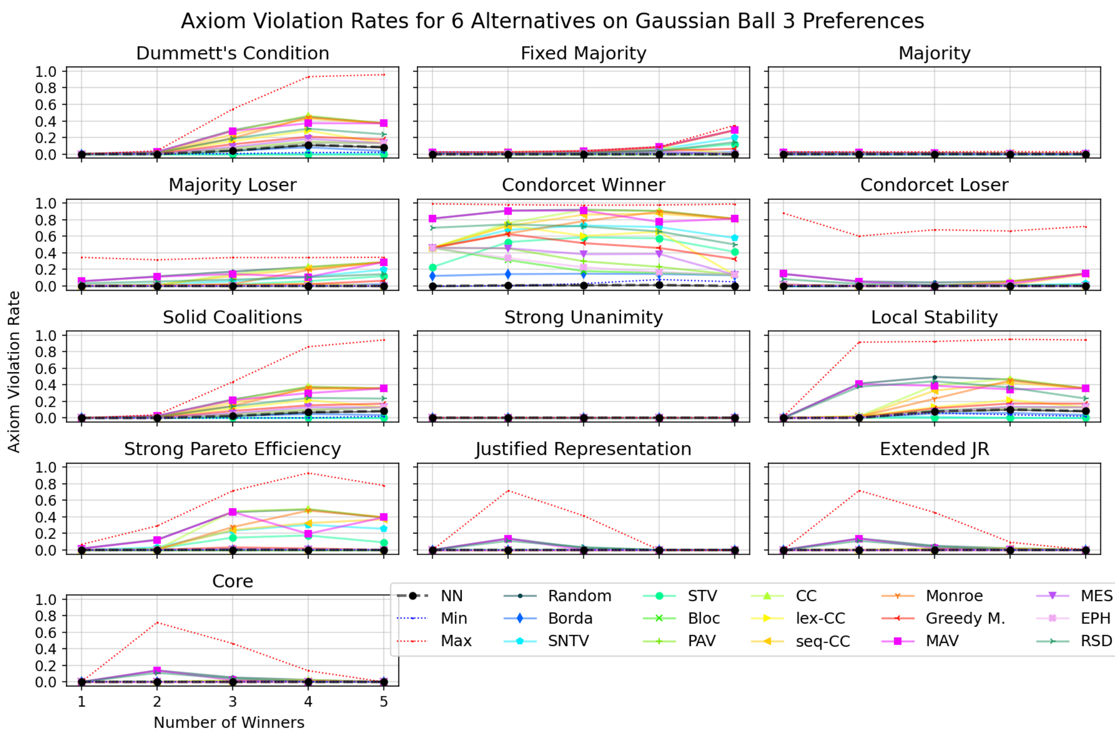


Figure D.53: Axiom violation rate for each axiom on Gaussian Ball 3 preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.865	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.154	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.555	.868	.571	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.367	.865	.376	.353	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.305	.867	.371	.391	.433	–	–	–	–	–	–	–	–	–	–	–
PAV	.336	.867	.376	.408	.439	.075	–	–	–	–	–	–	–	–	–	–
CC	.662	.867	.704	.550	.632	.462	.430	–	–	–	–	–	–	–	–	–
lex-CC	.426	.867	.461	.450	.491	.181	.117	.393	–	–	–	–	–	–	–	–
seq-CC	.651	.868	.669	.477	.617	.476	.447	.604	.431	–	–	–	–	–	–	–
Monroe	.617	.866	.659	.515	.590	.416	.385	.083	.388	.598	–	–	–	–	–	–
Greedy M.	.434	.868	.459	.406	.486	.247	.207	.480	.225	.363	.455	–	–	–	–	–
MAV	.806	.866	.824	.871	.830	.814	.807	.521	.782	.975	.568	.842	–	–	–	–
MES	.360	.867	.394	.380	.450	.147	.094	.474	.152	.399	.430	.160	.820	–	–	–
EPH	.312	.867	.369	.395	.432	.020	.062	.457	.171	.468	.411	.238	.814	.135	–	–
RSD	.651	.866	.665	.723	.695	.622	.626	.773	.649	.753	.751	.648	.840	.631	.623	–
Min	.034	.865	.160	.547	.358	.325	.354	.658	.440	.656	.614	.446	.804	.377	.332	.656
Max	1.164	.868	1.155	1.072	1.114	1.147	1.129	.996	1.092	.968	1.033	1.096	.942	1.120	1.142	1.016

Table D.56: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Gaussian Ball 3 preferences.

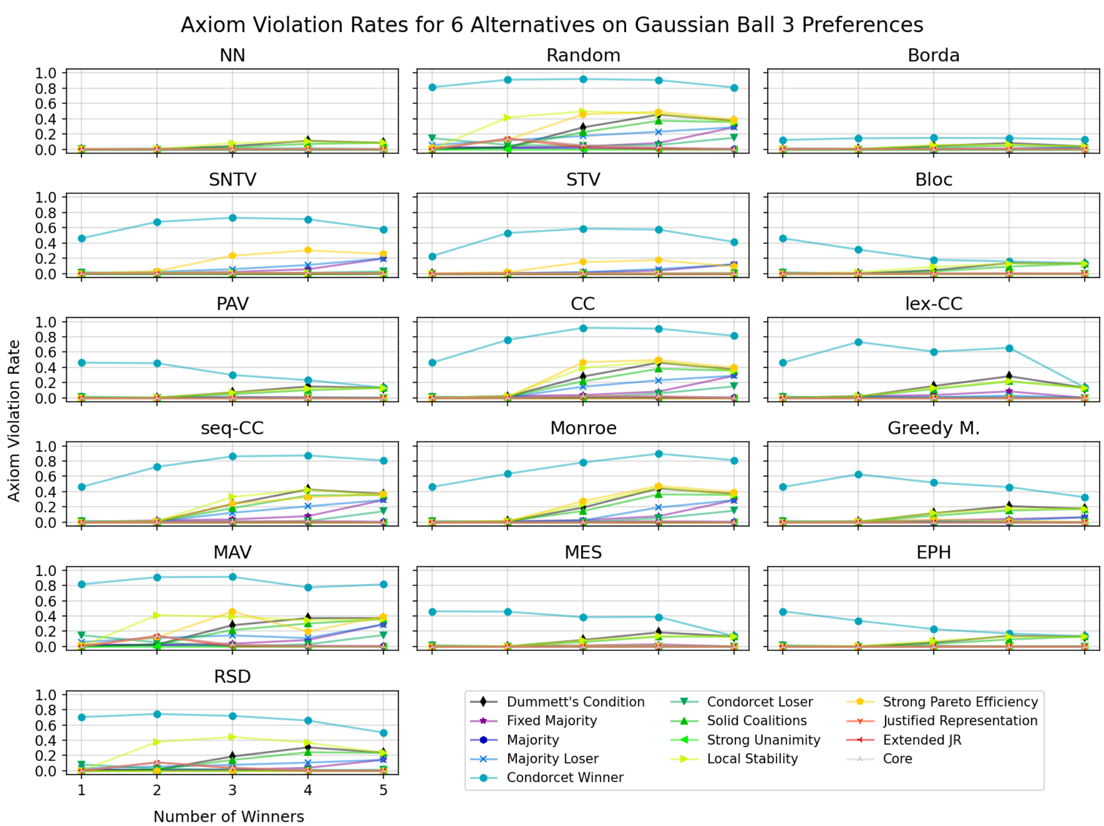


Figure D.54: Axiom violation rate for each rule on Gaussian Ball 3 preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.011**	*.000*	*.000*	.001	**.011**	**.006**	*.000*	**0**	**0**	.047	*.000*	*.000*	*.000*	.035	.052
Borda	.018	.001	.004	.001	.018	.137	**0**	.009	**0**	.033	*.000*	*.000*	*.000*	.022	.032
EPH	.035	*.000*	.001	*.000*	.035	.267	.003	.001	**0**	.065	**0**	**0**	**0**	.051	.067
SNTV	.074	**0**	.079	.165	.074	.635	.012	.057	**0**	.001	*.000*	.002	.002	**0**	.005
STV	.049	**0**	.040	.087	.049	.470	.002	.035	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.034	*.000*	.001	**0**	.034	.251	.003	**0**	**0**	.063	*.000*	*.000*	*.000*	.050	.073
CC	.154	.007	.138	.270	.154	.775	.044	.086	**0**	.223	**0**	.007	.008	.192	.247
lex-CC	.068	.004	.010	**0**	.068	.521	.003	.029	**0**	.119	**0**	**0**	**0**	.097	.101
seq-CC	.139	.005	.126	.189	.139	.749	.034	.085	**0**	.209	**0**	.004	.004	.179	.226
Monroe	.136	.005	.102	.225	.136	.721	.042	.080	**0**	.201	**0**	.002	.003	.173	.209
Greedy M.	.063	.001	.021	.010	.063	.481	.003	.027	**0**	.102	**0**	**0**	**0**	.083	.095
PAV	.040	*.000*	.001	**0**	.040	.317	.003	.003	**0**	.071	**0**	**0**	**0**	.056	.065
MES	.046	*.000*	.002	.003	.046	.368	.003	.008	**0**	.081	**0**	**0**	**0**	.064	.069
MAV	.168	.013	.145	.235	.168	.845	.076	.091	**0**	.209	.028	.032	.032	.179	.302
RSD	.113	.006	.085	**0**	.113	.667	.029	.044	**0**	.148	.029	.030	.030	.124	.283
Random	.186	.013	.174	.296	.186	.872	.090	.090	**0**	.228	.032	.041	.043	.196	.350

Table D.57: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Gaussian Ball 10 preferences.

D.2.10 6 Alternatives, Gaussian Ball 10

Figure D.55: Axiom violation rate for each axiom on Gaussian Ball 10 preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.864	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.155	.865	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.559	.867	.575	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.370	.867	.380	.355	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.310	.866	.375	.391	.438	–	–	–	–	–	–	–	–	–	–	–
PAV	.341	.867	.379	.408	.444	.074	–	–	–	–	–	–	–	–	–	–
CC	.665	.869	.705	.546	.633	.462	.429	–	–	–	–	–	–	–	–	–
lex-CC	.431	.867	.464	.449	.495	.181	.117	.394	–	–	–	–	–	–	–	–
seq-CC	.656	.867	.672	.479	.621	.474	.444	.601	.427	–	–	–	–	–	–	–
Monroe	.620	.868	.660	.511	.592	.415	.385	.082	.388	.594	–	–	–	–	–	–
Greedy M.	.438	.867	.463	.406	.489	.245	.205	.476	.222	.363	.451	–	–	–	–	–
MAV	.805	.866	.823	.869	.827	.815	.809	.521	.784	.976	.568	.842	–	–	–	–
MES	.366	.867	.398	.381	.453	.146	.093	.473	.150	.397	.429	.159	.823	–	–	–
EPH	.317	.866	.373	.394	.438	.020	.061	.456	.170	.466	.410	.236	.815	.133	–	–
RSD	.653	.867	.664	.722	.695	.621	.626	.773	.649	.753	.752	.647	.842	.630	.622	–
Min	.034	.864	.162	.551	.360	.330	.360	.661	.446	.661	.617	.451	.802	.383	.336	.659
Max	1.164	.867	1.155	1.074	1.113	1.147	1.129	.996	1.092	.969	1.034	1.096	.941	1.119	1.142	1.016

Table D.58: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Gaussian Ball 10 preferences.

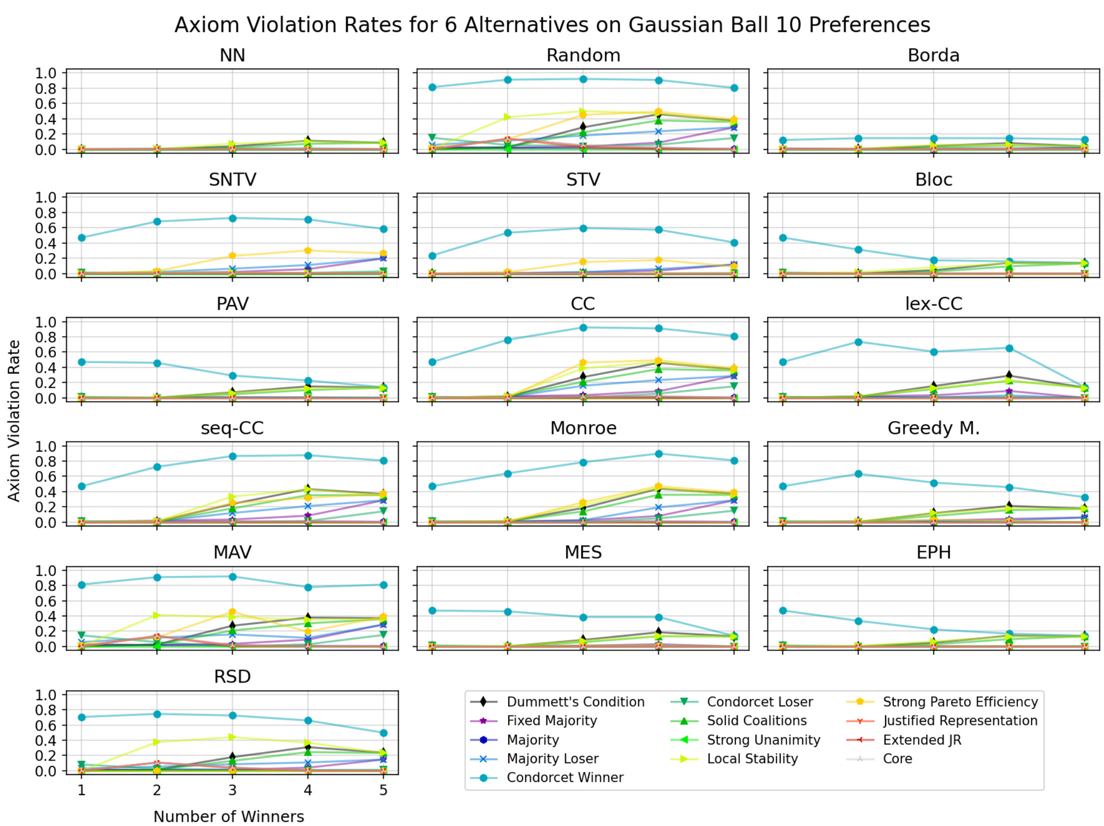


Figure D.56: Axiom violation rate for each rule on Gaussian Ball 10 preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.015**	*.000*	*.000*	.003	**.015**	**.008**	*.000*	**0**	**0**	.063	*.000*	*.000*	*.000*	.049	.075
Borda	.022	.001	.003	.002	.022	.160	**0**	.007	**0**	.042	*.000*	*.000*	*.000*	.029	.041
EPH	.046	*.000*	.002	*.000*	.046	.326	.004	.001	**0**	.094	**0**	**0**	**0**	.077	.098
SNTV	.075	**0**	.058	.182	.075	.671	.016	.037	**0**	.001	*.000*	.001	.002	**0**	.006
STV	.053	**0**	.033	.106	.053	.519	.003	.026	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.045	*.000*	.002	**0**	.045	.306	.005	**0**	**0**	.092	*.000*	*.000*	*.000*	.076	.108
CC	.143	.005	.079	.256	.143	.776	.045	.051	**0**	.216	**0**	.004	.004	.185	.239
lex-CC	.074	.003	.006	**0**	.074	.552	.004	.019	**0**	.142	**0**	**0**	**0**	.117	.121
seq-CC	.131	.004	.076	.169	.131	.753	.034	.051	**0**	.210	**0**	.002	.002	.180	.226
Monroe	.131	.004	.064	.223	.131	.738	.044	.049	**0**	.200	**0**	.001	.002	.172	.209
Greedy M.	.074	.001	.015	.014	.074	.538	.005	.020	**0**	.133	**0**	**0**	**0**	.110	.125
PAV	.052	*.000*	.001	**0**	.052	.383	.004	.004	**0**	.101	**0**	**0**	**0**	.083	.093
MES	.059	*.000*	.002	.005	.059	.439	.004	.008	**0**	.115	**0**	**0**	**0**	.093	.099
MAV	.157	.009	.082	.228	.157	.841	.074	.055	**0**	.207	.022	.025	.025	.176	.294
RSD	.117	.005	.053	**0**	.117	.687	.032	.028	**0**	.168	.030	.031	.031	.143	.317
Random	.172	.009	.101	.277	.172	.863	.087	.054	**0**	.220	.028	.033	.034	.189	.337

Table D.59: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Uniform Ball 3 preferences.

D.2.11 6 Alternatives, Uniform Ball 3

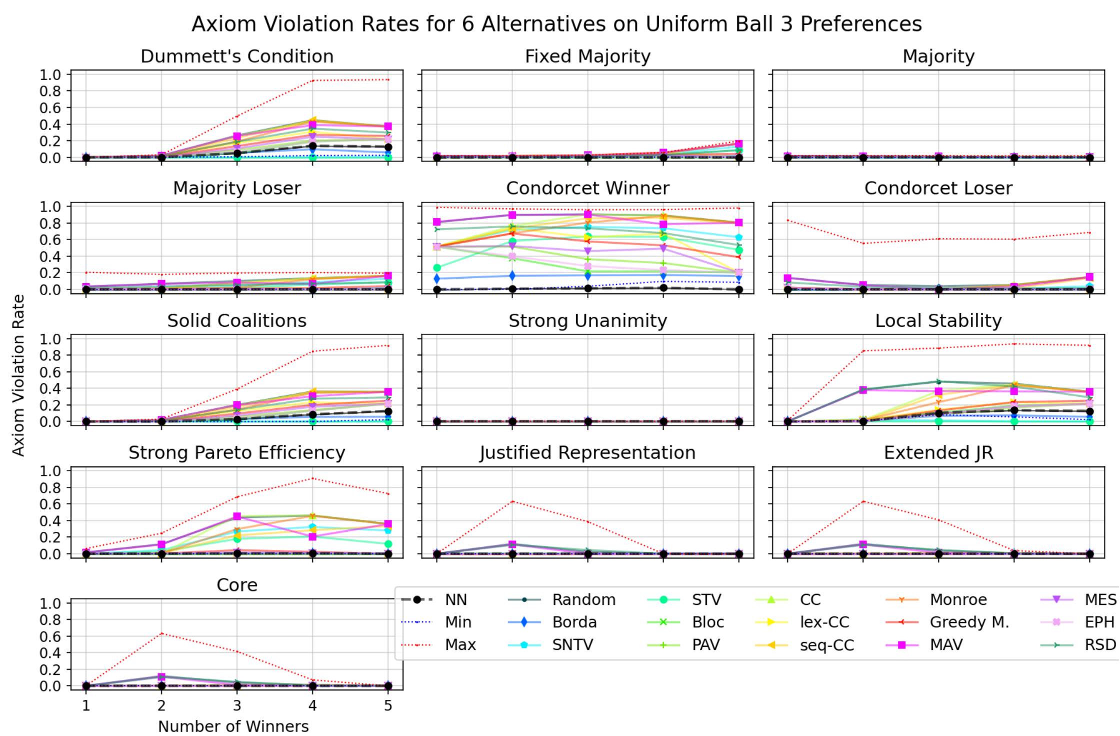


Figure D.57: Axiom violation rate for each axiom on Uniform Ball 3 preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.867	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.180	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.609	.865	.620	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.419	.865	.425	.375	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.364	.866	.430	.430	.498	–	–	–	–	–	–	–	–	–	–	–
PAV	.400	.866	.436	.450	.507	.089	–	–	–	–	–	–	–	–	–	–
CC	.684	.865	.728	.559	.653	.461	.424	–	–	–	–	–	–	–	–	–
lex-CC	.476	.866	.508	.483	.549	.182	.104	.393	–	–	–	–	–	–	–	–
seq-CC	.681	.864	.694	.487	.642	.481	.447	.604	.433	–	–	–	–	–	–	–
Monroe	.651	.865	.695	.533	.623	.426	.392	.069	.391	.601	–	–	–	–	–	–
Greedy M.	.496	.864	.518	.444	.550	.275	.230	.481	.238	.358	.466	–	–	–	–	–
MAV	.809	.866	.832	.876	.835	.816	.809	.510	.790	.981	.553	.851	–	–	–	–
MES	.429	.865	.460	.420	.518	.172	.111	.473	.147	.394	.441	.179	.827	–	–	–
EPH	.373	.866	.427	.434	.498	.028	.072	.454	.168	.472	.420	.264	.817	.157	–	–
RSD	.686	.866	.697	.750	.730	.647	.652	.779	.669	.760	.764	.673	.847	.657	.648	–
Min	.050	.867	.188	.597	.405	.393	.426	.678	.497	.688	.647	.512	.804	.452	.401	.694
Max	1.154	.867	1.143	1.052	1.097	1.128	1.107	.992	1.074	.951	1.021	1.070	.949	1.095	1.122	.996

Table D.60: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Uniform Ball 3 preferences.

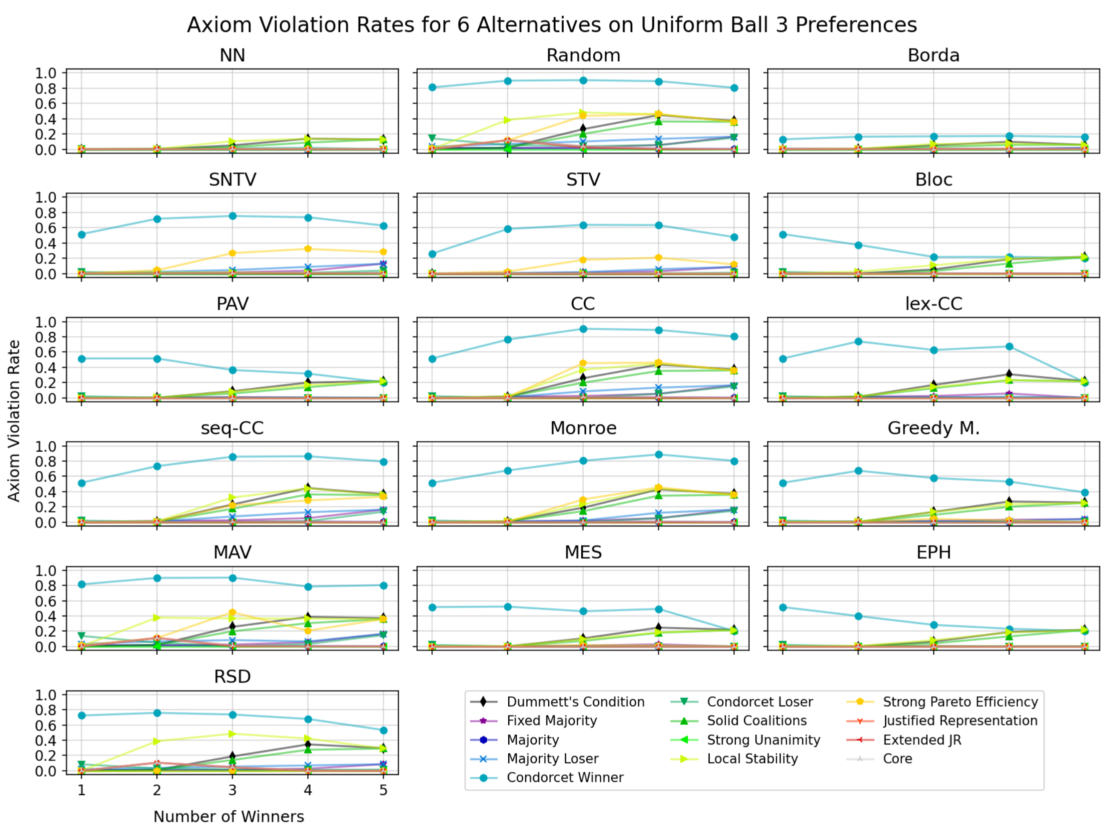


Figure D.58: Axiom violation rate for each rule on Uniform Ball 3 preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.015**	**0**	*.000*	.002	**.015**	**.007**	*.000*	*.000*	**0**	.063	*.000*	*.000*	*.000*	.049	.074
Borda	.022	.001	.004	.001	.022	.159	**0**	.007	**0**	.041	*.000*	*.000*	*.000*	.028	.041
EPH	.045	*.000*	.001	*.000*	.045	.322	.004	.001	**0**	.092	**0**	**0**	**0**	.075	.096
SNTV	.075	**0**	.059	.183	.075	.671	.017	.038	**0**	.001	*.000*	.002	.002	**0**	.006
STV	.053	**0**	.033	.106	.053	.518	.003	.025	**0**	**0**	**0**	*.000*	.001	**0**	**.001**
Bloc	.045	*.000*	.001	**0**	.045	.303	.004	**0**	**0**	.091	*.000*	*.000*	*.000*	.074	.106
CC	.145	.006	.082	.255	.145	.776	.045	.051	**0**	.222	**0**	.004	.005	.189	.245
lex-CC	.074	.003	.007	**0**	.074	.554	.004	.017	**0**	.140	**0**	**0**	**0**	.115	.119
seq-CC	.130	.004	.073	.166	.130	.753	.034	.050	**0**	.206	**0**	.002	.002	.176	.224
Monroe	.132	.004	.066	.221	.132	.736	.044	.048	**0**	.204	**0**	.001	.002	.175	.214
Greedy M.	.073	.001	.015	.013	.073	.538	.005	.019	**0**	.131	**0**	**0**	**0**	.108	.123
PAV	.051	*.000*	.002	**0**	.051	.380	.004	.003	**0**	.099	**0**	**0**	**0**	.080	.089
MES	.058	*.000*	.003	.005	.058	.440	.004	.007	**0**	.112	**0**	**0**	**0**	.091	.096
MAV	.159	.010	.086	.229	.159	.844	.074	.054	**0**	.211	.023	.026	.026	.180	.298
RSD	.118	.005	.054	**0**	.118	.689	.033	.027	**0**	.169	.032	.032	.032	.143	.318
Random	.172	.010	.102	.278	.172	.864	.088	.054	**0**	.221	.027	.032	.034	.189	.334

Table D.61: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Uniform Ball 10 preferences.

D.2.12 6 Alternatives, Uniform Ball 10

Figure D.59: Axiom violation rate for each axiom on Uniform Ball 10 preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.869	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.178	.869	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.607	.866	.619	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.418	.867	.425	.374	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.361	.866	.426	.429	.495	–	–	–	–	–	–	–	–	–	–	–
PAV	.396	.866	.433	.449	.504	.090	–	–	–	–	–	–	–	–	–	–
CC	.683	.866	.727	.559	.653	.461	.424	–	–	–	–	–	–	–	–	–
lex-CC	.474	.867	.505	.484	.547	.184	.105	.393	–	–	–	–	–	–	–	–
seq-CC	.680	.866	.691	.488	.640	.481	.447	.604	.433	–	–	–	–	–	–	–
Monroe	.650	.866	.694	.534	.623	.426	.391	.070	.391	.602	–	–	–	–	–	–
Greedy M.	.493	.867	.516	.443	.547	.276	.230	.482	.239	.357	.466	–	–	–	–	–
MAV	.808	.865	.833	.878	.837	.816	.810	.508	.792	.982	.551	.852	–	–	–	–
MES	.427	.866	.457	.420	.516	.173	.111	.474	.147	.393	.442	.179	.829	–	–	–
EPH	.369	.866	.423	.432	.494	.027	.073	.456	.170	.471	.421	.264	.818	.157	–	–
RSD	.685	.867	.699	.749	.730	.647	.652	.780	.669	.759	.764	.673	.846	.656	.648	–
Min	.049	.869	.186	.596	.404	.389	.423	.677	.496	.687	.645	.509	.803	.450	.397	.693
Max	1.154	.865	1.143	1.049	1.096	1.128	1.106	.988	1.073	.953	1.018	1.069	.943	1.094	1.121	.993

Table D.62: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Uniform Ball 10 preferences.

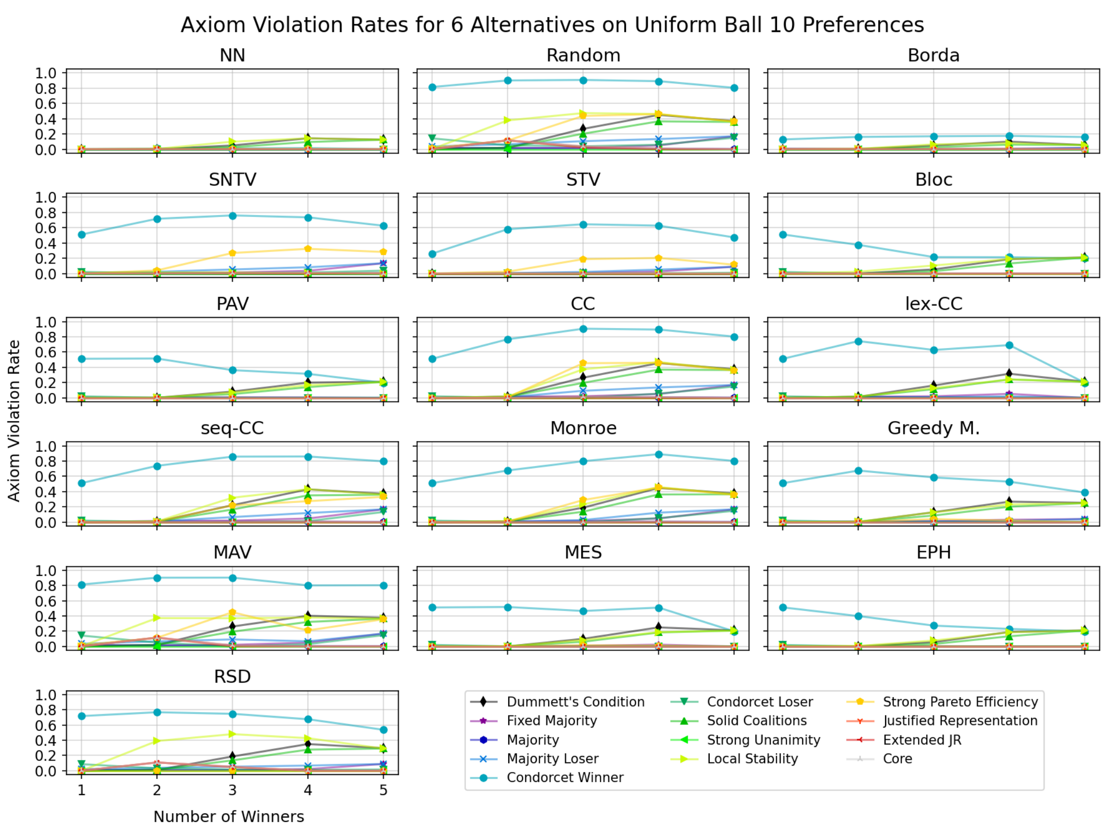


Figure D.60: Axiom violation rate for each rule on Uniform Ball 10 preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.014**	*.000*	*.000*	.002	**.014**	**.009**	*.000*	*.000*	**0**	.058	*.000*	*.000*	*.000*	.044	.067
Borda	.021	.001	.004	.001	.021	.152	**0**	.009	**0**	.039	*.000*	*.000*	*.000*	.027	.038
EPH	.042	*.000*	.001	*.000*	.042	.303	.004	.001	**0**	.082	**0**	**0**	**0**	.066	.086
SNTV	.076	**0**	.072	.184	.076	.662	.014	.049	**0**	.001	*.000*	.002	.003	**0**	.006
STV	.052	**0**	.038	.104	.052	.503	.003	.031	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.041	*.000*	.001	**0**	.041	.285	.004	**0**	**0**	.080	*.000*	*.000*	*.000*	.063	.094
CC	.151	.006	.119	.269	.151	.778	.045	.072	**0**	.221	**0**	.006	.007	.191	.246
lex-CC	.071	.003	.008	**0**	.071	.535	.004	.023	**0**	.131	**0**	**0**	**0**	.108	.112
seq-CC	.137	.005	.111	.184	.137	.752	.033	.072	**0**	.209	**0**	.003	.003	.179	.226
Monroe	.135	.004	.089	.228	.135	.735	.043	.068	**0**	.202	**0**	.001	.002	.174	.211
Greedy M.	.069	.001	.019	.013	.069	.511	.004	.024	**0**	.120	**0**	**0**	**0**	.098	.112
PAV	.047	*.000*	.002	**0**	.047	.356	.004	.005	**0**	.088	**0**	**0**	**0**	.070	.081
MES	.053	*.000*	.003	.004	.053	.408	.004	.008	**0**	.099	**0**	**0**	**0**	.079	.085
MAV	.165	.010	.123	.238	.165	.844	.076	.075	**0**	.208	.026	.029	.029	.178	.302
RSD	.117	.005	.074	**0**	.117	.677	.032	.037	**0**	.163	.029	.030	.030	.139	.307
Random	.182	.011	.150	.298	.182	.867	.088	.075	**0**	.225	.031	.037	.039	.194	.345

Table D.63: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Gaussian Cube 3 preferences.

D.2.13 6 Alternatives, Gaussian Cube 3

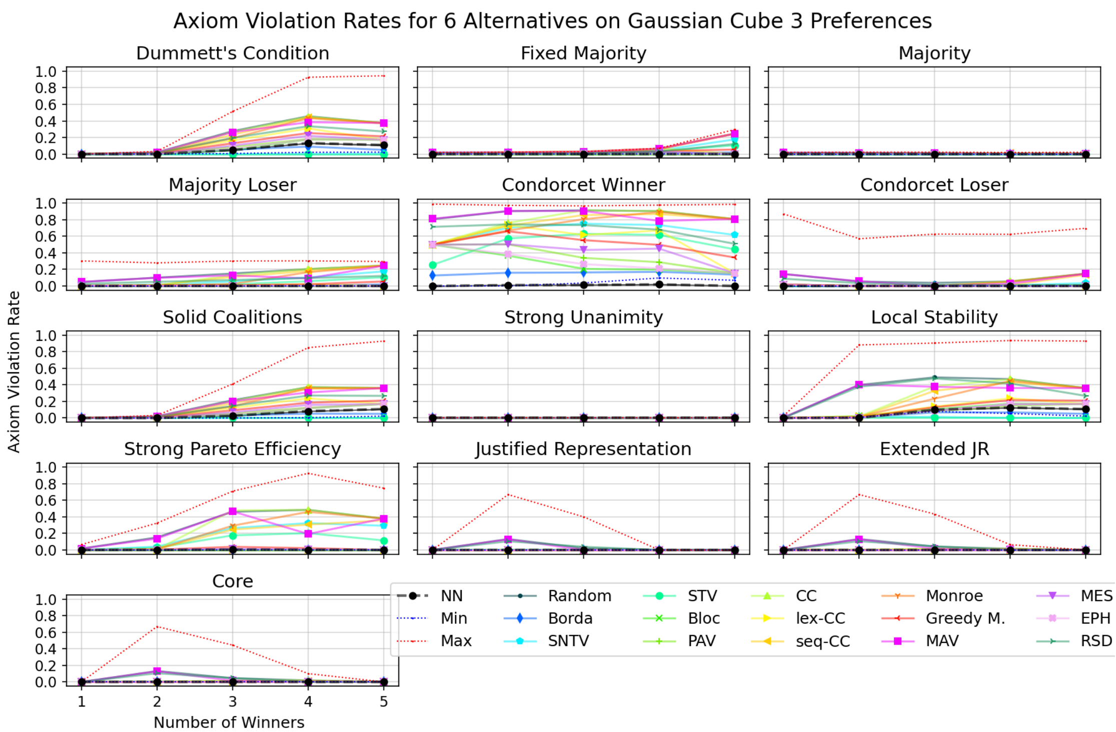


Figure D.61: Axiom violation rate for each axiom on Gaussian Cube 3 preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.864	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.168	.865	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.593	.867	.608	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.402	.866	.411	.369	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.340	.867	.403	.414	.470	–	–	–	–	–	–	–	–	–	–	–
PAV	.373	.867	.407	.433	.477	.083	–	–	–	–	–	–	–	–	–	–
CC	.680	.867	.720	.553	.643	.462	.426	–	–	–	–	–	–	–	–	–
lex-CC	.452	.867	.482	.468	.521	.180	.107	.393	–	–	–	–	–	–	–	–
seq-CC	.671	.867	.684	.484	.635	.475	.442	.604	.428	–	–	–	–	–	–	–
Monroe	.642	.867	.681	.523	.608	.421	.389	.076	.387	.598	–	–	–	–	–	–
Greedy M.	.467	.867	.489	.428	.521	.256	.212	.478	.225	.359	.457	–	–	–	–	–
MAV	.806	.864	.827	.873	.830	.815	.808	.519	.784	.977	.562	.843	–	–	–	–
MES	.399	.867	.428	.406	.488	.158	.099	.471	.143	.393	.434	.164	.822	–	–	–
EPH	.348	.867	.400	.418	.470	.024	.066	.455	.166	.465	.415	.245	.815	.144	–	–
RSD	.668	.867	.681	.738	.717	.633	.637	.775	.655	.756	.757	.656	.841	.641	.634	–
Min	.045	.864	.177	.584	.390	.366	.397	.674	.471	.677	.637	.483	.802	.420	.374	.675
Max	1.156	.868	1.146	1.058	1.101	1.134	1.116	.993	1.083	.959	1.025	1.081	.944	1.105	1.128	1.003

Table D.64: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Gaussian Cube 3 preferences.

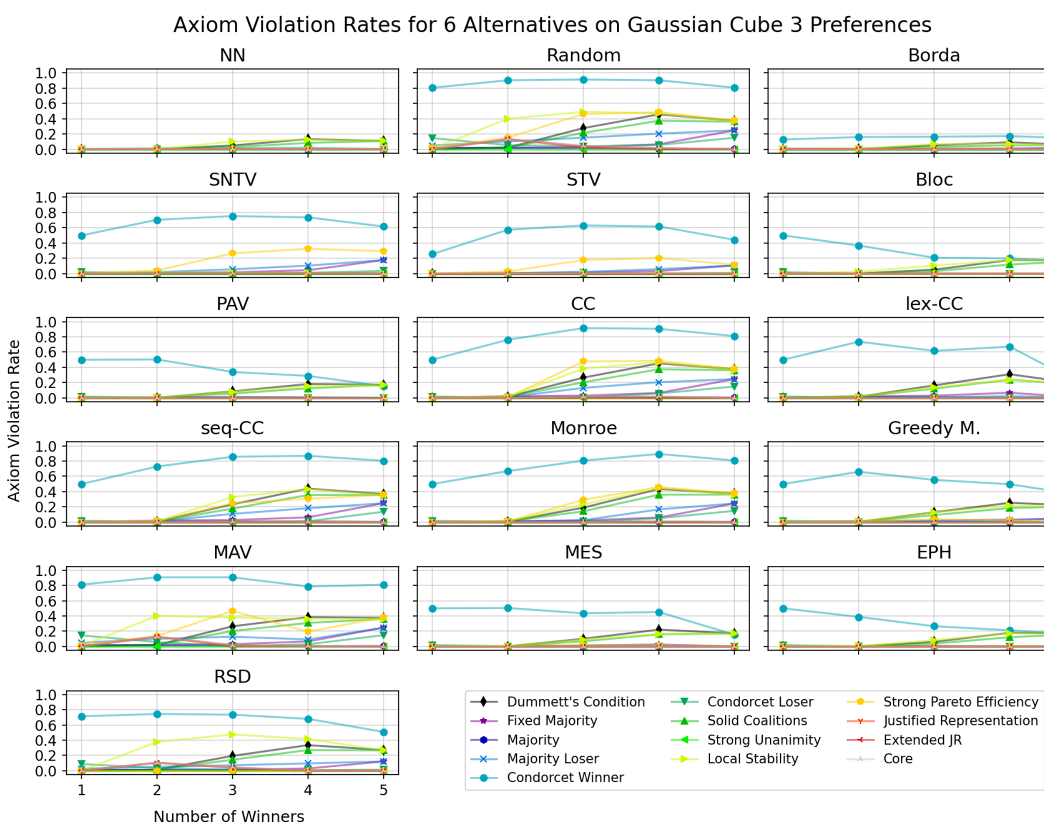


Figure D.62: Axiom violation rate for each rule on Gaussian Cube 3 preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.014**	*.000*	*.000*	.002	**.014**	**.007**	*.000*	**0**	**0**	.058	*.000*	*.000*	*.000*	.044	.067
Borda	.021	.001	.004	.002	.021	.150	**0**	.009	**0**	.039	*.000*	*.000*	*.000*	.028	.039
EPH	.042	*.000*	.001	*.000*	.042	.302	.004	.001	**0**	.082	**0**	**0**	**0**	.066	.085
SNTV	.076	**0**	.071	.185	.076	.660	.015	.048	**0**	.001	*.000*	.002	.003	**0**	.007
STV	.052	**0**	.036	.106	.052	.500	.003	.031	**0**	**0**	**0**	*.000*	*.000*	**0**	**.001**
Bloc	.041	*.000*	.001	**0**	.041	.284	.004	**0**	**0**	.079	*.000*	*.000*	*.000*	.064	.094
CC	.150	.006	.118	.268	.150	.775	.044	.072	**0**	.220	**0**	.006	.007	.188	.244
lex-CC	.071	.003	.008	**0**	.071	.534	.004	.023	**0**	.130	**0**	**0**	**0**	.107	.111
seq-CC	.137	.005	.110	.183	.137	.753	.035	.072	**0**	.212	**0**	.003	.003	.182	.228
Monroe	.135	.005	.089	.228	.135	.731	.043	.068	**0**	.200	**0**	.001	.002	.172	.210
Greedy M.	.070	.001	.019	.013	.070	.512	.004	.025	**0**	.120	**0**	**0**	**0**	.099	.112
PAV	.046	*.000*	.001	**0**	.046	.355	.004	.004	**0**	.088	**0**	**0**	**0**	.071	.081
MES	.054	*.000*	.003	.004	.054	.410	.004	.009	**0**	.100	**0**	**0**	**0**	.080	.086
MAV	.164	.011	.123	.239	.164	.843	.075	.076	**0**	.207	.025	.028	.028	.177	.297
RSD	.117	.006	.073	**0**	.117	.677	.031	.038	**0**	.162	.030	.031	.032	.138	.306
Random	.181	.011	.150	.296	.181	.868	.087	.075	**0**	.225	.031	.037	.038	.193	.344

Table D.65: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Gaussian Cube 10 preferences.

D.2.14 6 Alternatives, Gaussian Cube 10

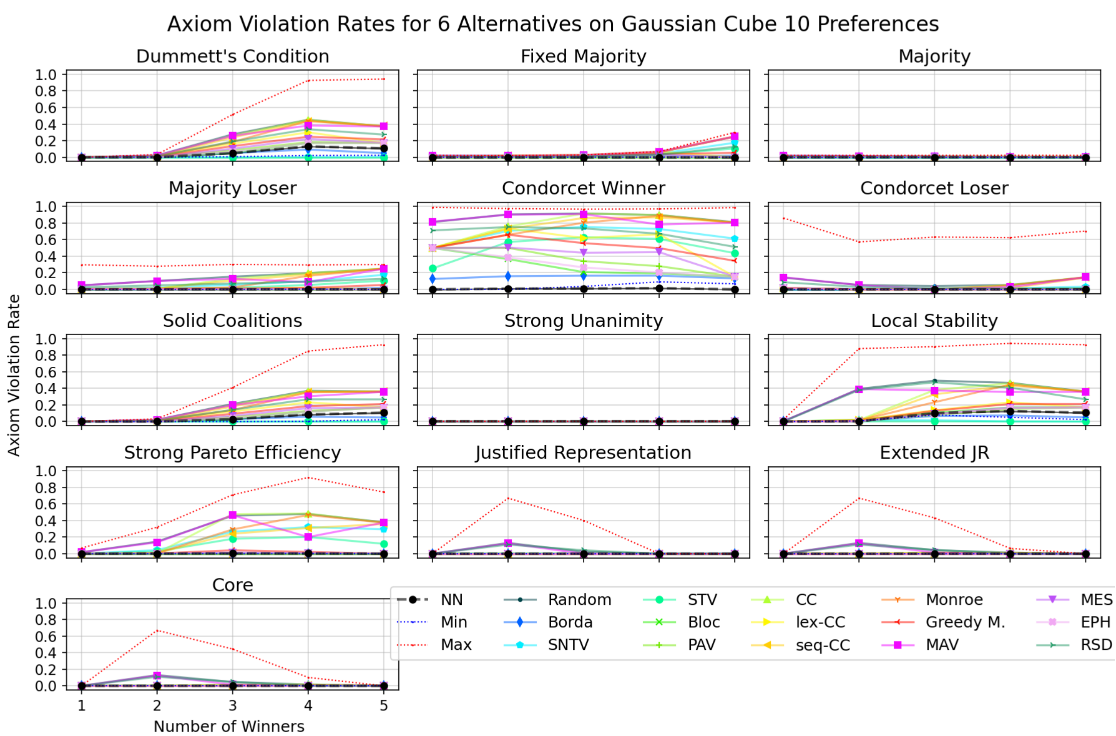


Figure D.63: Axiom violation rate for each axiom on Gaussian Cube 10 preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.867	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.167	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.592	.865	.608	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.400	.866	.410	.368	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.340	.866	.401	.414	.471	–	–	–	–	–	–	–	–	–	–	–
PAV	.373	.866	.407	.433	.479	.083	–	–	–	–	–	–	–	–	–	–
CC	.678	.866	.718	.553	.648	.460	.425	–	–	–	–	–	–	–	–	–
lex-CC	.453	.866	.482	.468	.523	.180	.107	.393	–	–	–	–	–	–	–	–
seq-CC	.673	.866	.685	.488	.637	.477	.445	.603	.430	–	–	–	–	–	–	–
Monroe	.640	.865	.680	.524	.614	.420	.387	.075	.387	.597	–	–	–	–	–	–
Greedy M.	.469	.865	.491	.429	.524	.258	.215	.478	.227	.359	.457	–	–	–	–	–
MAV	.807	.866	.826	.871	.832	.815	.807	.518	.785	.978	.561	.845	–	–	–	–
MES	.401	.866	.429	.405	.491	.159	.101	.471	.144	.395	.434	.166	.823	–	–	–
EPH	.347	.866	.398	.418	.471	.024	.067	.454	.167	.467	.414	.247	.815	.145	–	–
RSD	.670	.866	.681	.740	.717	.633	.639	.776	.657	.755	.758	.659	.844	.643	.635	–
Min	.043	.867	.175	.582	.388	.365	.396	.673	.471	.679	.636	.483	.803	.421	.372	.677
Max	1.157	.869	1.147	1.059	1.102	1.134	1.115	.994	1.081	.956	1.026	1.080	.944	1.104	1.129	1.002

Table D.66: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Gaussian Cube 10 preferences.

Figure D.64: Axiom violation rate for each rule on Gaussian Cube 10 preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.015**	**0**	*.000*	.002	**.015**	**.007**	*.000*	**0**	**0**	.061	*.000*	*.000*	*.000*	.047	.071
Borda	.022	.001	.004	.001	.022	.155	**0**	.009	**0**	.041	*.000*	*.000*	*.000*	.029	.041
EPH	.043	*.000*	.001	*.000*	.043	.308	.004	.002	**0**	.086	*.000*	*.000*	*.000*	.069	.090
SNTV	.077	**0**	.070	.188	.077	.665	.016	.048	**0**	.001	*.000*	.002	.003	**0**	.007
STV	.053	**0**	.037	.108	.053	.507	.003	.031	**0**	**0**	*.000*	*.000*	.001	**0**	**.001**
Bloc	.042	*.000*	.001	**0**	.042	.289	.004	**0**	**0**	.084	*.000*	*.000*	*.000*	.068	.100
CC	.150	.006	.113	.270	.150	.776	.045	.070	**0**	.221	**0**	.006	.007	.190	.246
lex-CC	.072	.003	.008	**0**	.072	.537	.004	.022	**0**	.135	**0**	**0**	**0**	.111	.115
seq-CC	.137	.005	.107	.184	.137	.752	.034	.070	**0**	.211	**0**	.003	.003	.181	.228
Monroe	.135	.004	.087	.230	.135	.734	.043	.067	**0**	.202	**0**	.001	.002	.174	.212
Greedy M.	.071	.001	.019	.013	.071	.515	.004	.025	**0**	.125	**0**	**0**	**0**	.103	.117
PAV	.048	*.000*	.001	**0**	.048	.363	.004	.004	**0**	.093	**0**	**0**	**0**	.075	.085
MES	.055	*.000*	.002	.004	.055	.415	.004	.009	**0**	.104	**0**	**0**	**0**	.084	.090
MAV	.164	.010	.119	.242	.164	.842	.075	.074	**0**	.211	.025	.028	.028	.180	.301
RSD	.118	.005	.071	**0**	.118	.677	.032	.036	**0**	.165	.030	.031	.032	.140	.311
Random	.180	.011	.146	.295	.180	.865	.088	.073	**0**	.224	.030	.036	.037	.192	.341

Table D.67: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Uniform Cube 3 preferences.

D.2.15 6 Alternatives, Uniform Cube 3

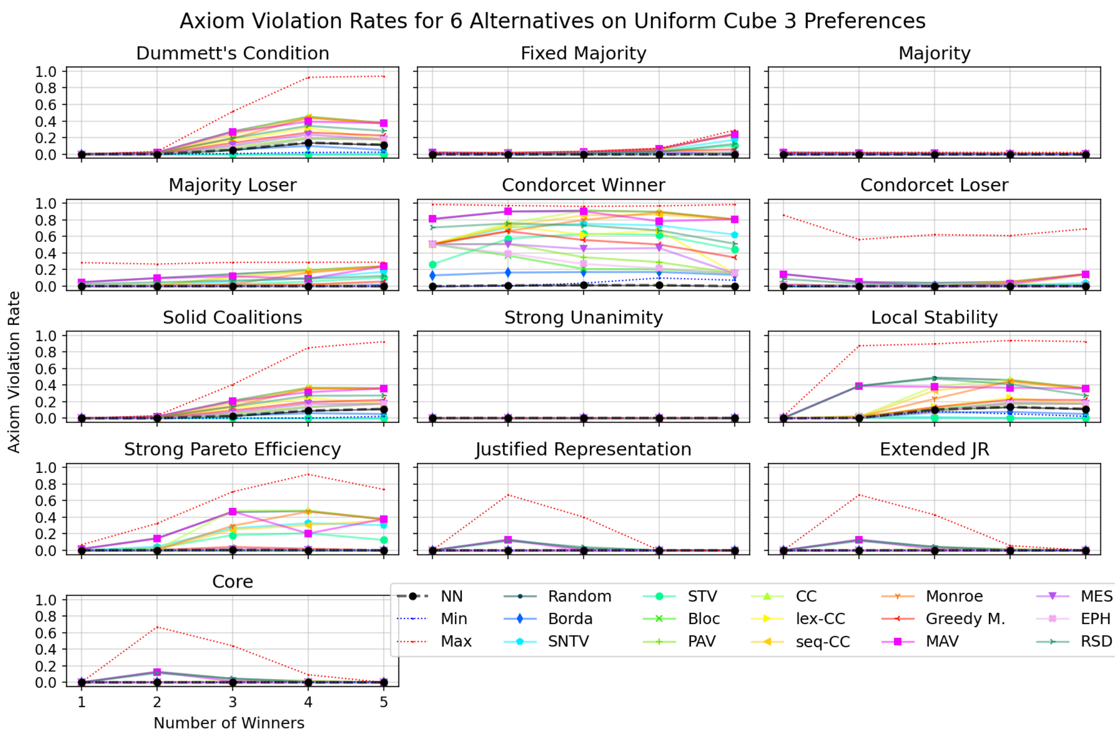


Figure D.65: Axiom violation rate for each axiom on Uniform Cube 3 preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.173	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.601	.865	.619	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.410	.865	.422	.372	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.348	.865	.412	.420	.477	–	–	–	–	–	–	–	–	–	–	–
PAV	.382	.865	.418	.440	.485	.085	–	–	–	–	–	–	–	–	–	–
CC	.685	.864	.727	.556	.648	.461	.425	–	–	–	–	–	–	–	–	–
lex-CC	.460	.865	.490	.473	.528	.180	.106	.393	–	–	–	–	–	–	–	–
seq-CC	.677	.864	.692	.487	.638	.478	.444	.605	.431	–	–	–	–	–	–	–
Monroe	.648	.865	.690	.527	.615	.421	.388	.074	.388	.599	–	–	–	–	–	–
Greedy M.	.475	.864	.499	.435	.529	.261	.216	.479	.228	.358	.460	–	–	–	–	–
MAV	.808	.865	.830	.874	.832	.815	.807	.517	.786	.979	.559	.846	–	–	–	–
MES	.408	.865	.438	.411	.495	.161	.102	.472	.144	.395	.436	.167	.823	–	–	–
EPH	.356	.865	.409	.424	.477	.025	.068	.455	.167	.468	.416	.249	.815	.147	–	–
RSD	.671	.865	.684	.740	.717	.635	.640	.776	.658	.755	.758	.660	.843	.643	.636	–
Min	.046	.866	.183	.590	.397	.375	.407	.678	.480	.684	.642	.491	.803	.430	.383	.679
Max	1.155	.867	1.145	1.056	1.100	1.132	1.111	.993	1.079	.955	1.025	1.077	.944	1.101	1.126	1.001

Table D.68: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Uniform Cube 3 preferences.

Figure D.66: Axiom violation rate for each rule on Uniform Cube 3 preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.014**	**0**	*.000*	.002	**.014**	**.007**	*.000*	**0**	**0**	.059	*.000*	*.000*	*.000*	.046	.069
Borda	.021	.001	.004	.001	.021	.154	**0**	.009	**0**	.040	*.000*	*.000*	*.000*	.028	.040
EPH	.043	*.000*	.001	*.000*	.043	.310	.004	.002	**0**	.086	**0**	**0**	**0**	.069	.089
SNTV	.077	**0**	.069	.186	.077	.664	.016	.048	**0**	.001	*.000*	.002	.003	**0**	.007
STV	.053	**0**	.037	.106	.053	.507	.003	.031	**0**	**0**	**0**	*.000*	*.000*	**0**	**.001**
Bloc	.042	*.000*	.001	**0**	.042	.291	.004	**0**	**0**	.083	*.000*	*.000*	*.000*	.067	.098
CC	.150	.006	.114	.269	.150	.777	.045	.071	**0**	.221	**0**	.006	.007	.190	.245
lex-CC	.072	.003	.008	**0**	.072	.538	.004	.023	**0**	.134	**0**	**0**	**0**	.110	.114
seq-CC	.136	.005	.105	.183	.136	.751	.034	.070	**0**	.211	**0**	.002	.002	.180	.227
Monroe	.135	.004	.087	.230	.135	.736	.043	.067	**0**	.203	**0**	.001	.002	.174	.212
Greedy M.	.071	.001	.019	.013	.071	.517	.005	.024	**0**	.125	**0**	**0**	**0**	.102	.116
PAV	.048	*.000*	.002	**0**	.048	.364	.004	.004	**0**	.093	**0**	**0**	**0**	.075	.085
MES	.055	*.000*	.003	.004	.055	.419	.004	.009	**0**	.104	**0**	**0**	**0**	.084	.090
MAV	.164	.010	.118	.241	.164	.842	.074	.075	**0**	.209	.025	.028	.028	.179	.297
RSD	.118	.006	.070	**0**	.118	.677	.031	.037	**0**	.166	.031	.032	.032	.142	.313
Random	.181	.011	.144	.297	.181	.866	.089	.073	**0**	.226	.030	.036	.038	.195	.345

Table D.69: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Uniform Cube 10 preferences.

D.2.16 6 Alternatives, Uniform Cube 10

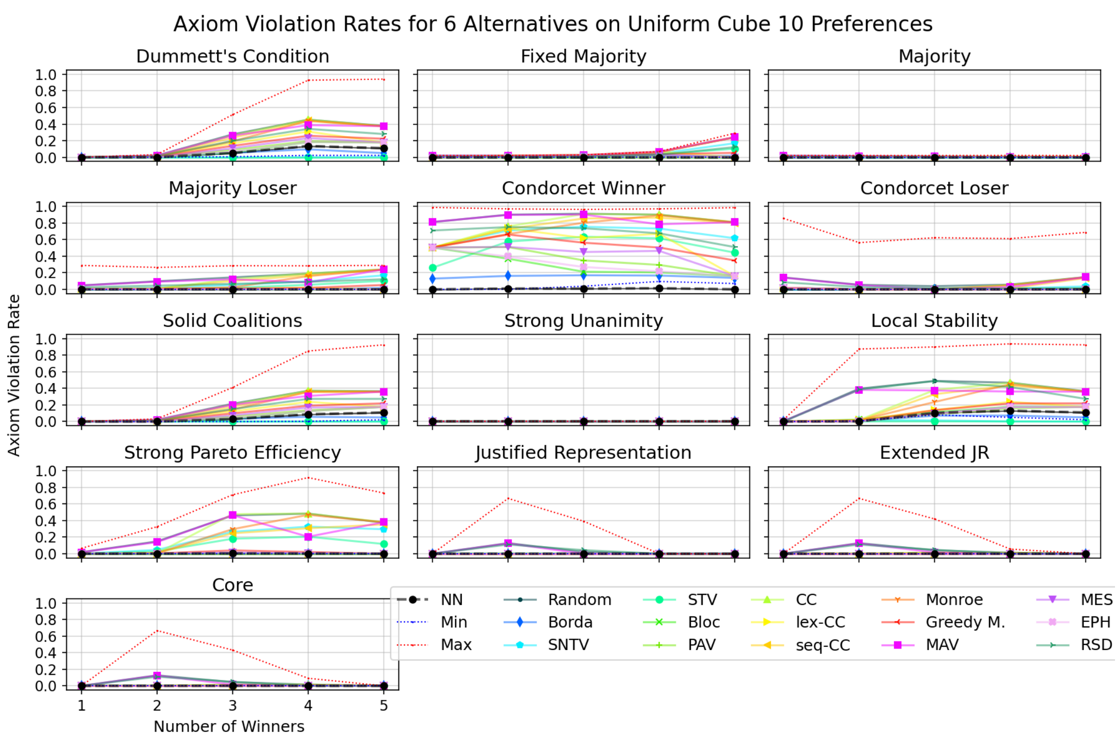


Figure D.67: Axiom violation rate for each axiom on Uniform Cube 10 preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.867	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.172	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.598	.866	.616	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.408	.866	.418	.371	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.346	.866	.410	.419	.477	–	–	–	–	–	–	–	–	–	–	–
PAV	.379	.866	.416	.439	.485	.085	–	–	–	–	–	–	–	–	–	–
CC	.683	.865	.724	.557	.650	.461	.426	–	–	–	–	–	–	–	–	–
lex-CC	.458	.866	.489	.474	.528	.180	.106	.394	–	–	–	–	–	–	–	–
seq-CC	.673	.865	.688	.487	.638	.476	.443	.604	.429	–	–	–	–	–	–	–
Monroe	.646	.865	.688	.528	.617	.422	.389	.074	.390	.599	–	–	–	–	–	–
Greedy M.	.474	.866	.498	.433	.528	.260	.216	.480	.228	.357	.460	–	–	–	–	–
MAV	.807	.867	.827	.873	.832	.813	.806	.516	.786	.978	.557	.845	–	–	–	–
MES	.408	.866	.437	.410	.496	.161	.102	.473	.142	.392	.437	.167	.822	–	–	–
EPH	.354	.866	.407	.423	.477	.025	.069	.455	.167	.467	.417	.249	.814	.147	–	–
RSD	.671	.868	.685	.738	.716	.632	.637	.774	.655	.752	.756	.656	.843	.640	.633	–
Min	.045	.866	.182	.588	.395	.372	.404	.677	.476	.680	.641	.489	.802	.429	.380	.678
Max	1.156	.866	1.145	1.056	1.100	1.131	1.112	.993	1.078	.955	1.024	1.077	.947	1.100	1.126	1.000

Table D.70: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Uniform Cube 10 preferences.

Figure D.68: Axiom violation rate for each rule on Uniform Cube 10 preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.012**	*.000*	**.001**	.005	**.012**	**.009**	*.000*	**0**	**0**	.043	.001	.001	.001	.034	.057
Borda	.016	.002	.006	.003	.016	.114	**0**	.013	**0**	.027	*.000*	*.000*	*.000*	.019	.029
EPH	.032	*.000*	.002	*.000*	.032	.235	.003	.001	**0**	.061	*.000*	*.000*	*.000*	.050	.068
SNTV	.089	**0**	.102	.182	.089	.561	.010	.107	.044	.053	*.000*	.047	.048	**0**	.007
STV	.041	**0**	.039	.085	.041	.376	.002	.032	**0**	**0**	**0**	*.000*	*.000*	**0**	*.000*
Bloc	.032	*.000*	.001	**0**	.032	.224	.003	**0**	**0**	.059	*.000*	*.000*	*.000*	.048	.073
CC	.178	.037	.152	.288	.178	.720	.040	.154	.059	.265	**0**	.077	.079	.200	.246
lex-CC	.054	.006	.009	**0**	.054	.405	.003	.027	**0**	.094	**0**	**0**	**0**	.076	.085
seq-CC	.166	.031	.147	.240	.166	.698	.032	.153	.059	.247	**0**	.071	.071	.183	.223
Monroe	.115	.008	.084	.186	.115	.602	.035	.071	**0**	.174	**0**	.003	.004	.150	.182
Greedy M.	.055	.002	.021	.010	.055	.404	.004	.026	**0**	.087	**0**	**0**	**0**	.071	.087
PAV	.035	.001	.002	**0**	.035	.270	.003	.005	**0**	.065	**0**	**0**	**0**	.052	.064
MES	.040	.001	.003	.002	.040	.303	.003	.008	**0**	.071	**0**	**0**	**0**	.058	.067
MAV	.149	.028	.127	.220	.149	.734	.063	.104	**0**	.179	.020	.027	.027	.150	.252
RSD	.096	.010	.068	**0**	.096	.571	.024	.046	**0**	.120	.023	.024	.024	.100	.235
Random	.226	.068	.188	.352	.226	.833	.084	.179	.070	.285	.049	.119	.122	.223	.360

Table D.71: Average Axiom Violation Rate for 6 alternatives and $1 \leq k < 6$ winners across Mixed preferences.

D.2.17 6 Alternatives, Mixed

Figure D.69: Axiom violation rate for each axiom on Mixed preferences with 6 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.866	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.174	.867	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.535	.866	.523	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.342	.866	.343	.352	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.303	.866	.335	.402	.400	–	–	–	–	–	–	–	–	–	–	–
PAV	.326	.866	.339	.413	.405	.057	–	–	–	–	–	–	–	–	–	–
CC	.630	.866	.664	.566	.611	.453	.428	–	–	–	–	–	–	–	–	–
lex-CC	.388	.865	.397	.440	.442	.136	.085	.397	–	–	–	–	–	–	–	–
seq-CC	.646	.866	.632	.438	.604	.478	.453	.626	.442	–	–	–	–	–	–	–
Monroe	.547	.866	.581	.508	.531	.368	.344	.110	.345	.587	–	–	–	–	–	–
Greedy M.	.418	.866	.414	.395	.447	.235	.206	.488	.222	.372	.416	–	–	–	–	–
MAV	.714	.870	.762	.832	.759	.742	.739	.504	.721	.940	.504	.779	–	–	–	–
MES	.350	.866	.350	.377	.413	.134	.093	.478	.134	.399	.395	.156	.760	–	–	–
EPH	.309	.866	.334	.404	.400	.016	.047	.449	.127	.471	.364	.228	.743	.125	–	–
RSD	.603	.868	.608	.687	.633	.576	.578	.746	.595	.731	.680	.596	.773	.581	.577	–
Min	.035	.866	.176	.528	.333	.317	.339	.629	.398	.647	.546	.425	.715	.360	.322	.606
Max	1.158	.866	1.140	1.064	1.112	1.132	1.119	.997	1.094	.954	1.050	1.089	.986	1.109	1.128	1.028

Table D.72: Difference between rules for 6 alternatives with $1 \leq k < 6$ on Mixed preferences.

Figure D.70: Axiom violation rate for each rule on Mixed preferences with 6 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.017**	*.000*	*.000*	.004	**.017**	**.015**	*.000*	*.000*	*.000*	.061	.001	.001	.001	.046	.092
Borda	.021	.001	.004	.004	.021	.125	**0**	.011	**0**	.044	*.000*	*.000*	*.000*	.031	.056
EPH	.040	*.000*	.001	*.000*	.040	.270	.002	.001	**0**	.082	*.000*	*.000*	*.000*	.063	.096
SNTV	.099	**0**	.098	.227	.099	.619	.007	.106	.049	.062	.001	.054	.058	**0**	.012
STV	.048	**0**	.037	.118	.048	.442	.002	.029	**0**	**0**	*.000*	*.000*	.001	**0**	**.001**
Bloc	.039	*.000*	.001	**0**	.039	.254	.002	**0**	**0**	.080	*.000*	*.000*	*.000*	.061	.106
CC	.195	.036	.146	.344	.195	.756	.031	.141	.062	.308	**0**	.084	.091	.232	.301
lex-CC	.061	.005	.007	**0**	.061	.440	.002	.024	**0**	.117	**0**	*.000*	*.000*	.091	.112
seq-CC	.183	.032	.139	.297	.183	.740	.025	.140	.061	.292	**0**	.078	.081	.216	.278
Monroe	.130	.007	.078	.234	.130	.649	.026	.060	**0**	.214	**0**	.002	.006	.180	.231
Greedy M.	.063	.002	.019	.012	.063	.448	.003	.023	**0**	.112	**0**	**0**	**0**	.089	.118
PAV	.043	.001	.001	**0**	.043	.308	.002	.004	**0**	.088	**0**	**0**	**0**	.068	.091
MES	.049	.001	.002	.001	.049	.351	.002	.008	**0**	.096	**0**	**0**	**0**	.075	.095
MAV	.157	.022	.110	.279	.157	.750	.044	.084	**0**	.219	.015	.022	.022	.179	.300
RSD	.105	.008	.056	**0**	.105	.594	.016	.036	**0**	.148	.030	.032	.033	.120	.299
Random	.237	.063	.171	.406	.237	.845	.057	.160	.071	.326	.049	.125	.134	.252	.419

Table D.73: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across all preferences.

D.3 7 Alternatives – All preferences

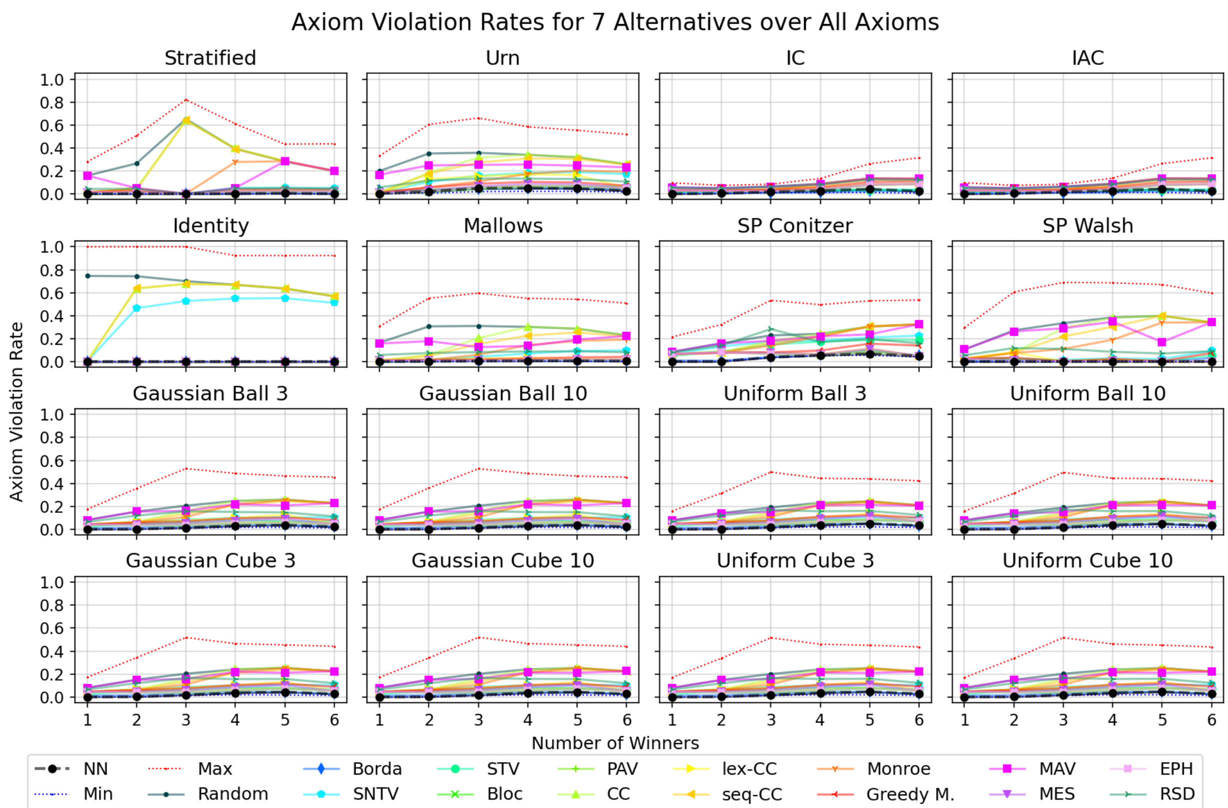


Figure D.71: Axiom violation rates for each rule under each preference distribution for 7 alternatives

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.723	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.149	.723	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.487	.723	.466	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.336	.723	.317	.301	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.235	.723	.251	.418	.362	–	–	–	–	–	–	–	–	–	–	–
PAV	.260	.722	.258	.430	.367	.065	–	–	–	–	–	–	–	–	–	–
CC	.572	.723	.602	.589	.574	.487	.464	–	–	–	–	–	–	–	–	–
lex-CC	.318	.722	.313	.452	.398	.144	.089	.440	–	–	–	–	–	–	–	–
seq-CC	.595	.723	.570	.464	.569	.493	.468	.673	.462	–	–	–	–	–	–	–
Monroe	.487	.723	.516	.533	.494	.400	.376	.117	.366	.619	–	–	–	–	–	–
Greedy M.	.351	.723	.336	.428	.406	.233	.204	.513	.219	.402	.432	–	–	–	–	–
MAV	.571	.723	.620	.703	.625	.607	.606	.350	.595	.821	.352	.644	–	–	–	–
MES	.285	.723	.273	.412	.376	.131	.087	.497	.118	.431	.410	.170	.620	–	–	–
EPH	.242	.723	.250	.422	.363	.021	.051	.483	.133	.485	.396	.225	.607	.119	–	–
RSD	.490	.723	.492	.592	.532	.469	.473	.652	.487	.635	.583	.490	.635	.475	.470	–
Min	.051	.723	.159	.479	.326	.256	.280	.568	.334	.601	.484	.364	.569	.302	.263	.496
Max	.967	.723	.950	.875	.922	.951	.940	.823	.923	.758	.876	.914	.842	.932	.946	.866

Table D.74: Difference between rules for 7 alternatives with $1 \leq k < 7$ averaged over all preference distributions.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.001**	*.000*	**0**	.001	**.001**	**.011**	*.000*	**0**	**0**	.001	**0**	**0**	**0**	.001	.001
Borda	.005	**0**	**0**	**0**	.005	.061	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
EPH	.014	*.000*	**0**	**0**	.014	.183	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
SNTV	.030	**0**	*.000*	*.000*	.030	.396	**0**	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	**0**
STV	.015	**0**	*.000*	**0**	.015	.189	**0**	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Bloc	.014	*.000*	**0**	**0**	.014	.183	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
CC	.264	.004	*.000*	.502	.264	.697	.028	.164	.162	.504	**0**	.224	.224	.449	.466
lex-CC	.016	*.000*	**0**	**0**	.016	.210	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
seq-CC	.260	.004	*.000*	.500	.260	.675	.023	.163	.162	.501	**0**	.222	.222	.446	.463
Monroe	.137	.003	*.000*	.305	.137	.512	.024	*.000*	**0**	.307	**0**	.018	.018	.295	.295
Greedy M.	.026	*.000*	**0**	*.000*	.026	.340	**0**	*.000*	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
PAV	.014	*.000*	**0**	**0**	.014	.184	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
MES	.015	*.000*	**0**	**0**	.015	.195	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
MAV	.124	.004	*.000*	.286	.124	.604	.040	.004	**0**	.192	**0**	*.000*	*.000*	.191	.286
RSD	.038	.002	*.000*	**0**	.038	.481	**0**	.002	**0**	.001	**0**	**0**	**0**	.001	.001
Random	.324	.007	*.000*	.736	.324	.854	.050	.166	.162	.503	.067	.272	.272	.452	.675

Table D.75: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Stratified preferences.

D.3.1 7 Alternatives, Stratified

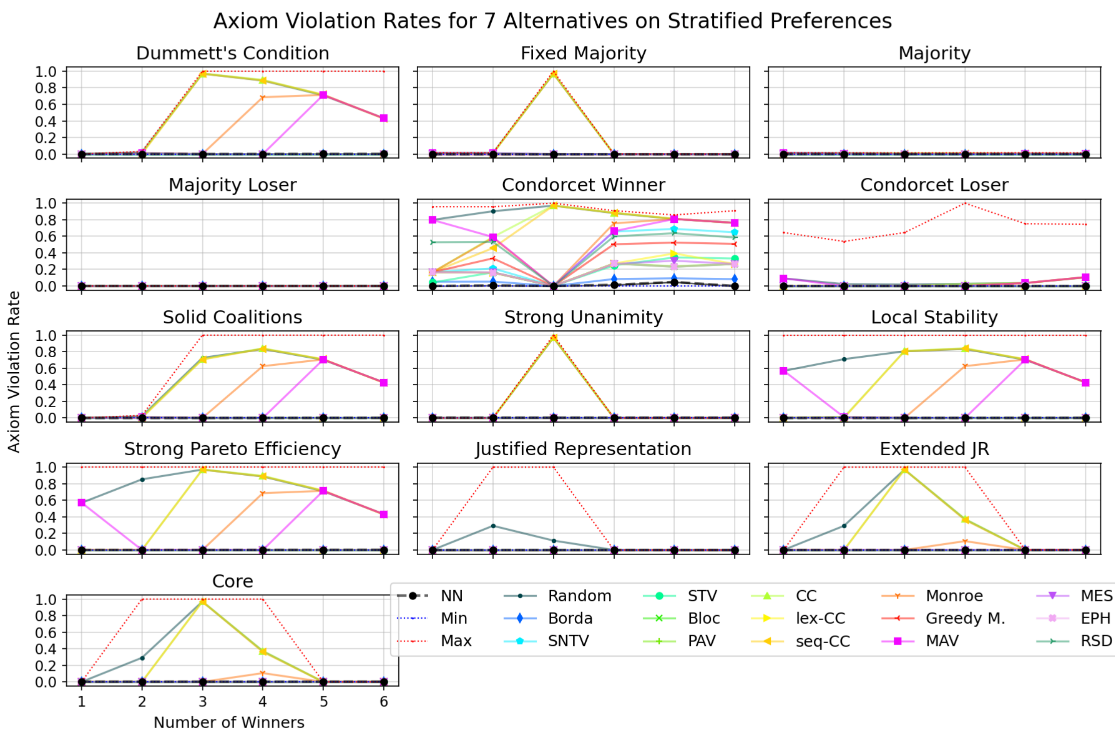


Figure D.72: Axiom violation rate for each axiom on Stratified preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.856	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.142	.856	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.394	.856	.344	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.190	.855	.207	.337	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.220	.856	.206	.338	.285	–	–	–	–	–	–	–	–	–	–	–
PAV	.222	.856	.206	.334	.284	.008	–	–	–	–	–	–	–	–	–	–
CC	.552	.856	.585	.644	.574	.490	.492	–	–	–	–	–	–	–	–	–
lex-CC	.234	.856	.221	.335	.288	.034	.028	.492	–	–	–	–	–	–	–	–
seq-CC	.606	.856	.562	.348	.575	.538	.535	.784	.536	–	–	–	–	–	–	–
Monroe	.430	.856	.460	.519	.450	.363	.365	.138	.365	.672	–	–	–	–	–	–
Greedy M.	.330	.856	.304	.266	.340	.251	.249	.561	.251	.464	.439	–	–	–	–	–
MAV	.601	.858	.651	.737	.635	.627	.629	.441	.629	.921	.346	.671	–	–	–	–
MES	.243	.856	.210	.278	.293	.101	.095	.549	.094	.476	.421	.207	.659	–	–	–
EPH	.220	.856	.206	.338	.285	.000	.007	.490	.034	.538	.363	.251	.627	.101	–	–
RSD	.484	.859	.480	.503	.491	.473	.474	.700	.475	.697	.586	.487	.646	.474	.473	–
Min	.009	.856	.141	.393	.190	.220	.222	.554	.235	.604	.432	.330	.603	.244	.220	.484
Max	1.160	.859	1.157	1.149	1.158	1.159	1.158	.985	1.158	.916	1.069	1.157	1.082	1.158	1.159	1.158

Table D.76: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Stratified preferences.

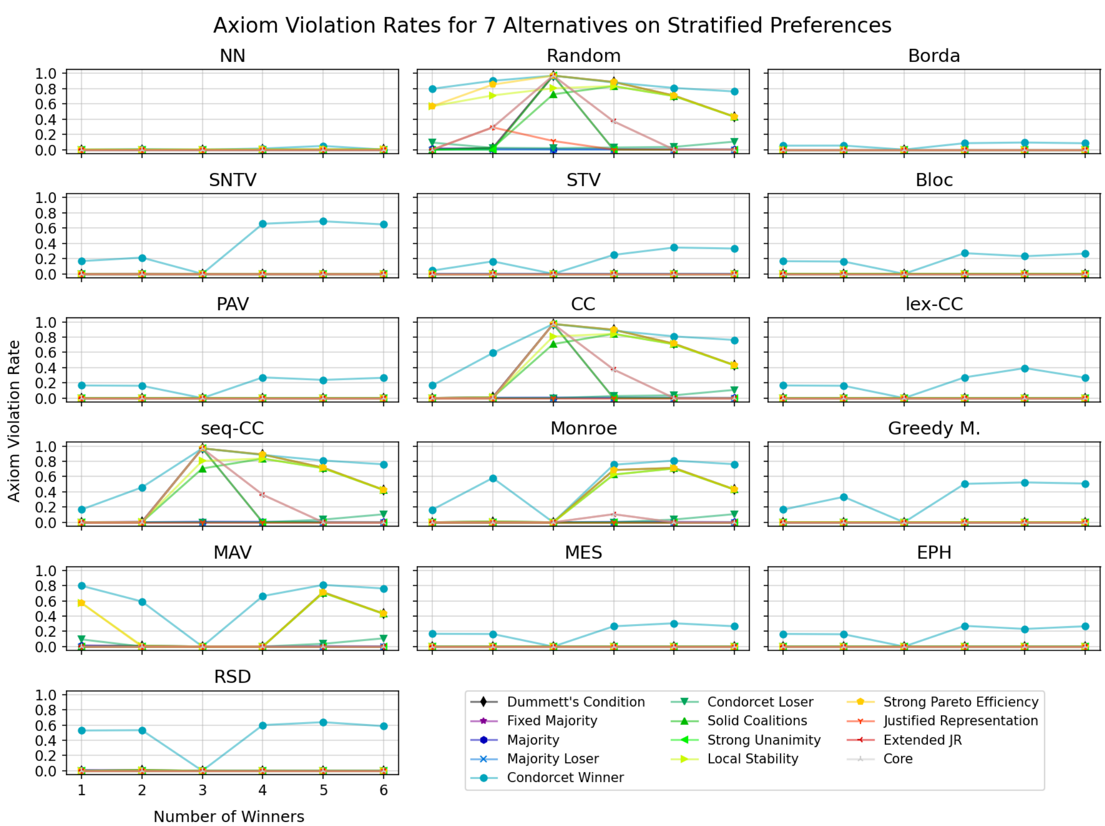


Figure D.73: Axiom violation rate for each rule on Stratified preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.032**	.001	.004	.041	**.032**	**.019**	.001	*.000*	*.000*	.110	.006	.007	.007	.089	.130
Borda	.038	.017	.016	.049	.038	.158	**0**	.074	**0**	.062	.001	.001	.001	.049	.065
EPH	.042	.005	.004	*.000*	.042	.128	.001	.009	**0**	.141	*.000*	*.000*	*.000*	.119	.141
SNTV	.136	**0**	.208	.275	.136	.568	.001	.330	.029	.188	*.000*	.080	.088	**0**	.001
STV	.045	**0**	.070	.110	.045	.295	*.000*	.115	**0**	**0**	*.000*	*.000*	.001	**0**	**.001**
Bloc	.039	.001	**.001**	**0**	.039	.110	.001	**0**	**0**	.138	.001	.001	.001	.115	.142
CC	.237	.154	.231	.318	.237	.616	.028	.348	.031	.452	**0**	.111	.124	.321	.354
lex-CC	.112	.074	.073	**0**	.112	.423	.001	.213	**0**	.266	**0**	*.000*	*.000*	.195	.214
seq-CC	.221	.139	.220	.268	.221	.605	.024	.343	.031	.426	**0**	.092	.100	.296	.325
Monroe	.124	.056	.107	.149	.124	.411	.018	.159	**0**	.268	**0**	.001	.001	.205	.234
Greedy M.	.071	.028	.034	.013	.071	.269	.003	.079	**0**	.180	**0**	**0**	**0**	.145	.166
PAV	.047	.010	.008	**0**	.047	.160	.001	.022	**0**	.148	**0**	**0**	**0**	.124	.143
MES	.054	.015	.016	*.000*	.054	.200	.001	.040	**0**	.155	**0**	**0**	**0**	.128	.145
MAV	.233	.202	.200	.294	.233	.686	.041	.385	**0**	.406	.026	.059	.059	.294	.375
RSD	.111	.076	.075	**0**	.111	.358	.017	.116	**0**	.244	.027	.040	.042	.194	.261
Random	.304	.258	.255	.435	.304	.731	.053	.423	.037	.488	.077	.179	.196	.359	.457

Table D.77: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Urn preferences.

D.3.2 7 Alternatives, Urn

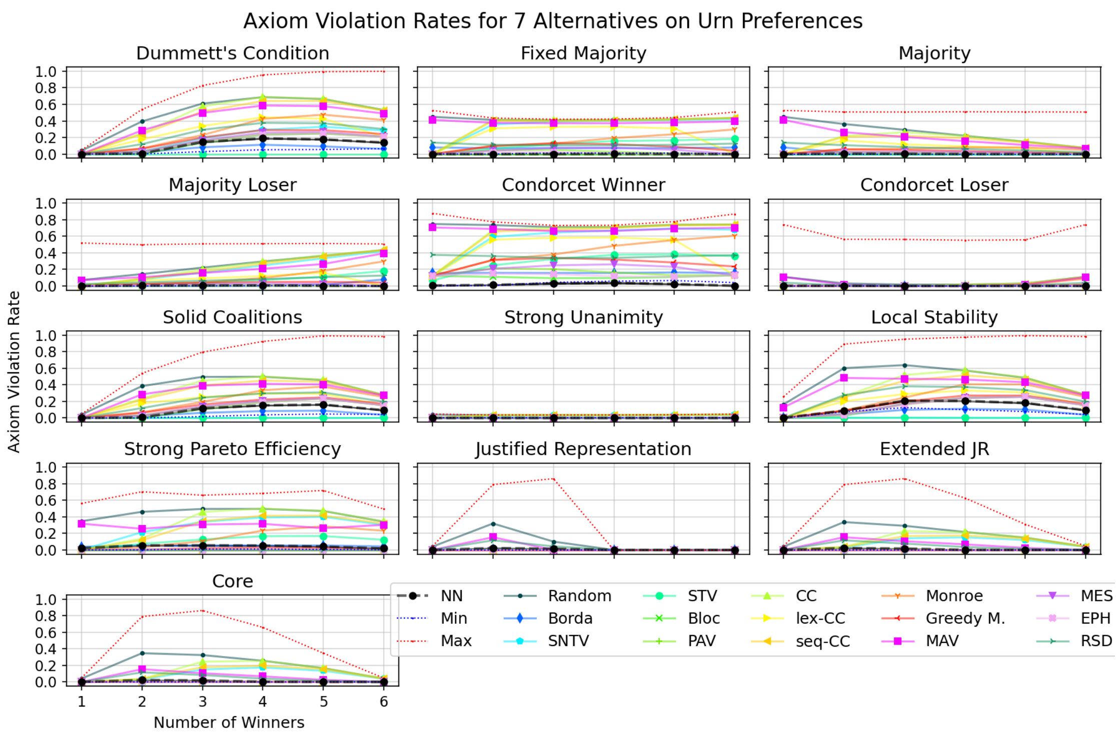


Figure D.74: Axiom violation rate for each axiom on Urn preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.266	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.544	.857	.533	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.325	.858	.305	.382	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.232	.856	.284	.481	.336	–	–	–	–	–	–	–	–	–	–	–
PAV	.254	.856	.292	.481	.341	.053	–	–	–	–	–	–	–	–	–	–
CC	.549	.857	.628	.606	.578	.504	.484	–	–	–	–	–	–	–	–	–
lex-CC	.369	.855	.394	.495	.416	.202	.160	.433	–	–	–	–	–	–	–	–
seq-CC	.615	.855	.595	.425	.560	.495	.478	.674	.470	–	–	–	–	–	–	–
Monroe	.396	.857	.471	.539	.427	.335	.317	.199	.355	.598	–	–	–	–	–	–
Greedy M.	.325	.857	.339	.466	.365	.184	.163	.523	.242	.420	.369	–	–	–	–	–
MAV	.666	.857	.761	.852	.767	.752	.743	.456	.688	.935	.517	.777	–	–	–	–
MES	.276	.856	.303	.464	.346	.106	.073	.512	.180	.448	.347	.133	.760	–	–	–
EPH	.240	.856	.286	.480	.338	.017	.040	.499	.191	.489	.331	.177	.750	.095	–	–
RSD	.483	.857	.495	.651	.520	.440	.444	.681	.503	.659	.549	.464	.771	.449	.441	–
Min	.094	.857	.273	.531	.313	.251	.269	.552	.377	.614	.396	.333	.674	.288	.257	.485
Max	1.126	.857	1.106	1.000	1.087	1.111	1.098	.974	1.044	.903	1.057	1.065	.969	1.086	1.106	1.023

Table D.78: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Urn preferences.

Figure D.75: Axiom violation rate for each rule on Urn preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	.018	**0**	**0**	*.000*	.018	**.039**	.001	**0**	**0**	.055	**0**	**0**	**0**	.036	.107
Borda	**.014**	**0**	**0**	**0**	**.014**	.113	**0**	**0**	**0**	.025	**0**	**0**	**0**	.013	.030
EPH	.046	**0**	**0**	**0**	.046	.326	.001	**0**	**0**	.091	**0**	**0**	**0**	.065	.111
SNTV	.033	**0**	**0**	*.000*	.033	.422	.002	**0**	**0**	**0**	**0**	**0**	**0**	**0**	.002
STV	.026	**0**	**0**	*.000*	.026	.336	.001	**0**	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
Bloc	.046	**0**	**0**	**0**	.046	.324	.001	**0**	**0**	.091	**0**	**0**	**0**	.065	.112
CC	.075	**0**	**0**	*.000*	.075	.493	.014	**0**	**0**	.147	**0**	**0**	**0**	.111	.209
lex-CC	.052	**0**	**0**	**0**	.052	.391	.001	**0**	**0**	.097	**0**	**0**	**0**	.068	.119
seq-CC	.069	**0**	**0**	**0**	.069	.481	.009	**0**	**0**	.132	**0**	**0**	**0**	.099	.177
Monroe	.075	**0**	**0**	*.000*	.075	.493	.014	**0**	**0**	.147	**0**	**0**	**0**	.111	.209
Greedy M.	.060	**0**	**0**	**0**	.060	.436	.004	**0**	**0**	.112	**0**	**0**	**0**	.082	.146
PAV	.046	**0**	**0**	**0**	.046	.337	.001	**0**	**0**	.091	**0**	**0**	**0**	.065	.109
MES	.048	**0**	**0**	**0**	.048	.354	.001	**0**	**0**	.094	**0**	**0**	**0**	.067	.111
MAV	.086	**0**	**0**	*.000*	.086	.575	.027	**0**	**0**	.146	**0**	**0**	**0**	.110	.254
RSD	.081	**0**	**0**	**0**	.081	.553	.017	**0**	**0**	.136	**0**	**0**	**0**	.101	.242
Random	.086	**0**	**0**	*.000*	.086	.578	.026	**0**	**0**	.147	**0**	**0**	**0**	.109	.263

Table D.79: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across IC preferences.

D.3.3 7 Alternatives, IC

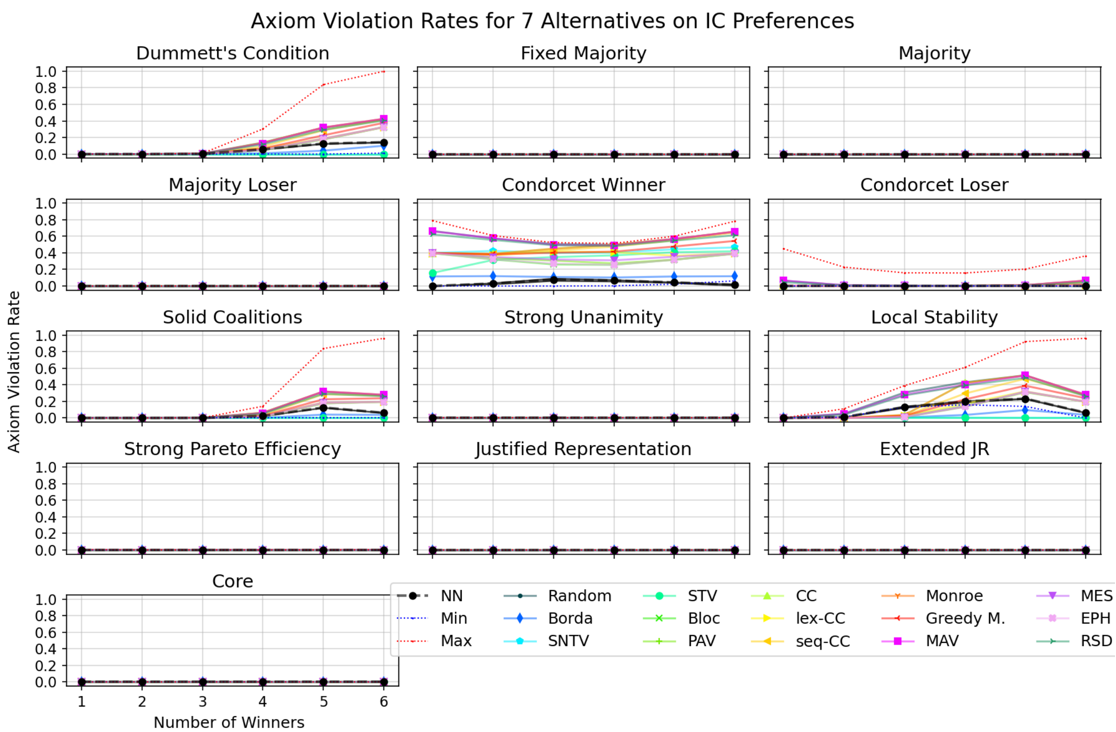


Figure D.76: Axiom violation rate for each axiom on IC preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.398	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.662	.858	.555	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.515	.856	.454	.361	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.561	.857	.464	.524	.585	–	–	–	–	–	–	–	–	–	–	–
PAV	.576	.857	.473	.525	.586	.073	–	–	–	–	–	–	–	–	–	–
CC	.594	.859	.680	.608	.656	.440	.419	–	–	–	–	–	–	–	–	–
lex-CC	.606	.857	.522	.541	.606	.167	.111	.384	–	–	–	–	–	–	–	–
seq-CC	.768	.859	.637	.478	.656	.507	.469	.628	.455	–	–	–	–	–	–	–
Monroe	.594	.859	.680	.608	.656	.439	.419	.001	.384	.628	–	–	–	–	–	–
Greedy M.	.679	.859	.571	.474	.629	.398	.374	.543	.389	.339	.543	–	–	–	–	–
MAV	.586	.857	.858	.898	.827	.809	.822	.470	.821	.993	.470	.901	–	–	–	–
MES	.604	.857	.485	.456	.598	.223	.172	.508	.205	.363	.507	.287	.866	–	–	–
EPH	.566	.857	.466	.523	.585	.019	.064	.439	.161	.500	.438	.393	.815	.216	–	–
RSD	.813	.856	.798	.811	.814	.774	.775	.815	.782	.806	.815	.792	.857	.777	.774	–
Min	.081	.857	.404	.655	.506	.571	.586	.602	.616	.770	.602	.682	.596	.613	.576	.815
Max	1.070	.859	.972	.928	.984	.970	.957	1.014	.946	.799	1.014	.895	1.066	.933	.964	.878

Table D.80: Difference between rules for 7 alternatives with $1 \leq k < 7$ on IC preferences.

Figure D.77: Axiom violation rate for each rule on IC preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	.019	0	0	*.000*	.019	**.040**	.001	0	0	.056	0	0	0	.037	.110
Borda	**.014**	0	0	0	**.014**	.115	0	0	0	.025	0	0	0	.014	.031
EPH	.046	0	0	0	.046	.324	.001	0	0	.091	0	0	0	.064	.111
SNTV	.033	0	0	*.000*	.033	.422	.002	0	0	0	0	0	0	0	.002
STV	.026	0	0	*.000*	.026	.337	.001	0	0	0	0	0	0	0	*.000*
Bloc	.045	0	0	0	.045	.322	.001	0	0	.091	0	0	0	.064	.113
CC	.075	0	0	*.000*	.075	.494	.013	0	0	.149	0	0	0	.111	.211
lex-CC	.052	0	0	0	.052	.391	.001	0	0	.097	0	0	0	.069	.120
seq-CC	.069	0	0	0	.069	.481	.009	0	0	.133	0	0	0	.100	.179
Monroe	.075	0	0	*.000*	.075	.494	.013	0	0	.149	0	0	0	.111	.211
Greedy M.	.060	0	0	0	.060	.436	.004	0	0	.110	0	0	0	.081	.145
PAV	.046	0	0	0	.046	.336	.001	0	0	.091	0	0	0	.064	.109
MES	.048	0	0	0	.048	.353	.001	0	0	.093	0	0	0	.066	.111
MAV	.086	0	0	*.000*	.086	.578	.026	0	0	.148	*.000*	*.000*	*.000*	.110	.256
RSD	.081	0	0	0	.081	.552	.017	0	0	.138	0	0	0	.104	.243
Random	.087	0	0	*.000*	.087	.579	.026	0	0	.147	0	0	0	.111	.265

Table D.81: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across IAC preferences.

D.3.4 7 Alternatives, IAC

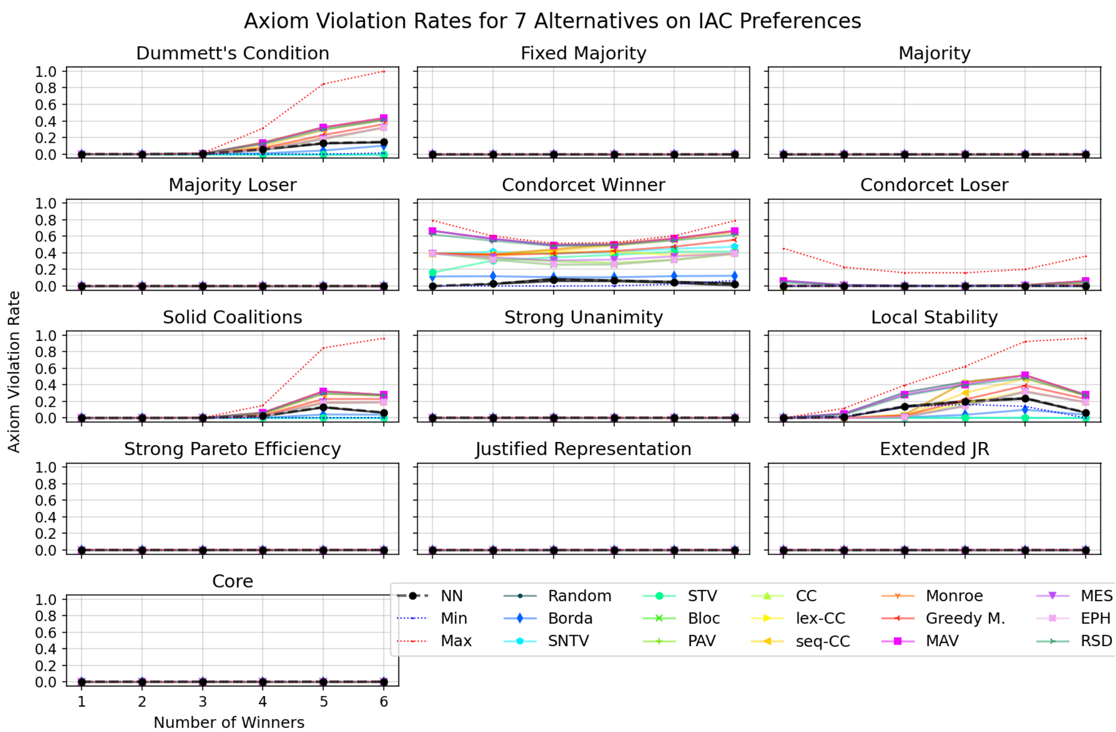


Figure D.78: Axiom violation rate for each axiom on IAC preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.399	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.662	.858	.554	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.517	.860	.454	.361	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.560	.857	.462	.523	.583	–	–	–	–	–	–	–	–	–	–	–
PAV	.574	.857	.471	.525	.584	.073	–	–	–	–	–	–	–	–	–	–
CC	.596	.856	.679	.607	.655	.439	.418	–	–	–	–	–	–	–	–	–
lex-CC	.604	.857	.520	.541	.604	.167	.112	.383	–	–	–	–	–	–	–	–
seq-CC	.768	.856	.636	.478	.656	.506	.468	.629	.454	–	–	–	–	–	–	–
Monroe	.596	.856	.679	.607	.655	.439	.418	.001	.383	.629	–	–	–	–	–	–
Greedy M.	.679	.857	.572	.475	.630	.400	.375	.543	.391	.339	.543	–	–	–	–	–
MAV	.588	.858	.859	.899	.829	.810	.822	.472	.821	.995	.472	.902	–	–	–	–
MES	.602	.857	.484	.455	.596	.223	.172	.505	.205	.363	.505	.287	.865	–	–	–
EPH	.565	.857	.464	.523	.583	.019	.064	.437	.161	.499	.437	.395	.815	.215	–	–
RSD	.812	.857	.795	.811	.813	.775	.777	.815	.784	.805	.815	.793	.854	.778	.776	–
Min	.083	.858	.404	.654	.507	.570	.584	.605	.614	.770	.605	.683	.600	.610	.574	.814
Max	1.071	.859	.973	.930	.984	.968	.957	1.013	.946	.799	1.013	.896	1.067	.935	.964	.879

Table D.82: Difference between rules for 7 alternatives with $1 \leq k < 7$ on IAC preferences.

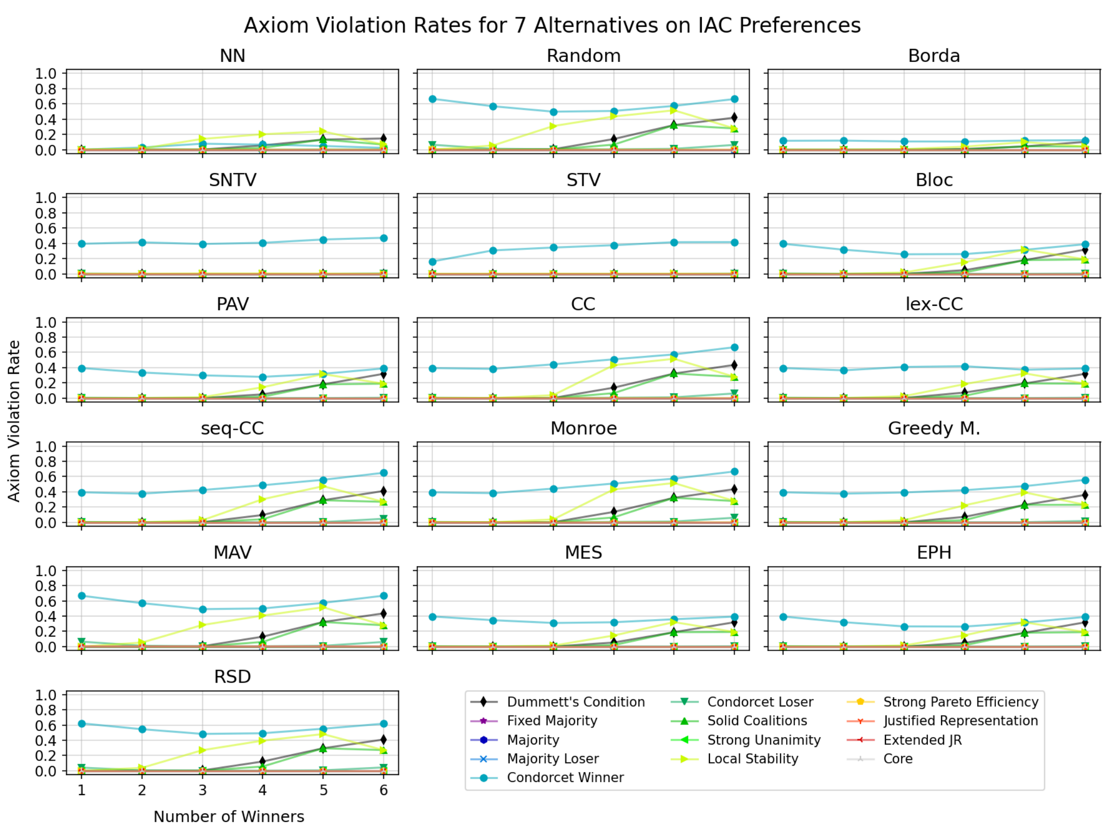


Figure D.79: Axiom violation rate for each rule on IAC preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Borda	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
EPH	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
SNTV	.434	**0**	.416	.747	.434	.747	**0**	.747	.747	.747	**0**	.747	.747	**0**	**0**
STV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Bloc	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
CC	.531	.310	.455	.784	.531	.784	.037	.784	.784	.784	**0**	.784	.784	.310	.310
lex-CC	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
seq-CC	.532	.312	.454	.784	.532	.784	.036	.784	.784	.784	**0**	.784	.784	.312	.312
Monroe	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Greedy M.	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
PAV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MES	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
MAV	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
RSD	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Random	.677	.499	.501	.927	.677	.927	.073	.927	.927	.927	.241	.927	.927	.499	.499

Table D.83: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Identity preferences.

D.3.5 7 Alternatives, Identity

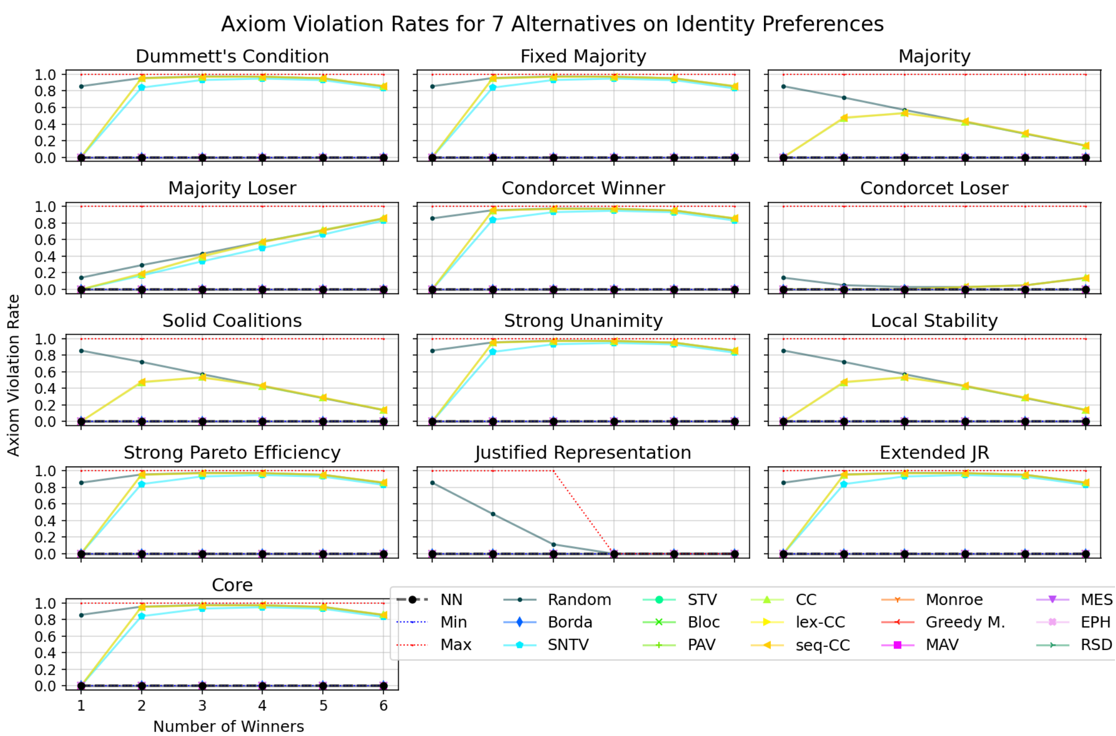


Figure D.80: Axiom violation rate for each axiom on Identity preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.000	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.476	.857	.476	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.000	.858	.000	.476	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.000	.858	.000	.476	.000	–	–	–	–	–	–	–	–	–	–	–
PAV	.000	.858	.000	.476	.000	.000	–	–	–	–	–	–	–	–	–	–
CC	.526	.856	.526	.760	.526	.526	.526	–	–	–	–	–	–	–	–	–
lex-CC	.000	.858	.000	.476	.000	.000	.000	.526	–	–	–	–	–	–	–	–
seq-CC	.524	.857	.524	.141	.524	.524	.524	.822	.524	–	–	–	–	–	–	–
Monroe	.000	.858	.000	.476	.000	.000	.000	.526	.000	.524	–	–	–	–	–	–
Greedy M.	.000	.858	.000	.476	.000	.000	.000	.526	.000	.524	.000	–	–	–	–	–
MAV	.000	.858	.000	.476	.000	.000	.000	.526	.000	.524	.000	.000	–	–	–	–
MES	.000	.858	.000	.476	.000	.000	.000	.526	.000	.524	.000	.000	.000	–	–	–
EPH	.000	.858	.000	.476	.000	.000	.000	.526	.000	.524	.000	.000	.000	.000	–	–
RSD	.000	.858	.000	.476	.000	.000	.000	.526	.000	.524	.000	.000	.000	.000	.000	–
Min	.000	.858	.000	.476	.000	.000	.000	.526	.000	.524	.000	.000	.000	.000	.000	.000
Max	1.167	.858	1.167	1.001	1.167	1.167	1.167	.924	1.167	.927	1.167	1.167	1.167	1.167	1.167	1.167

Table D.84: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Identity preferences.

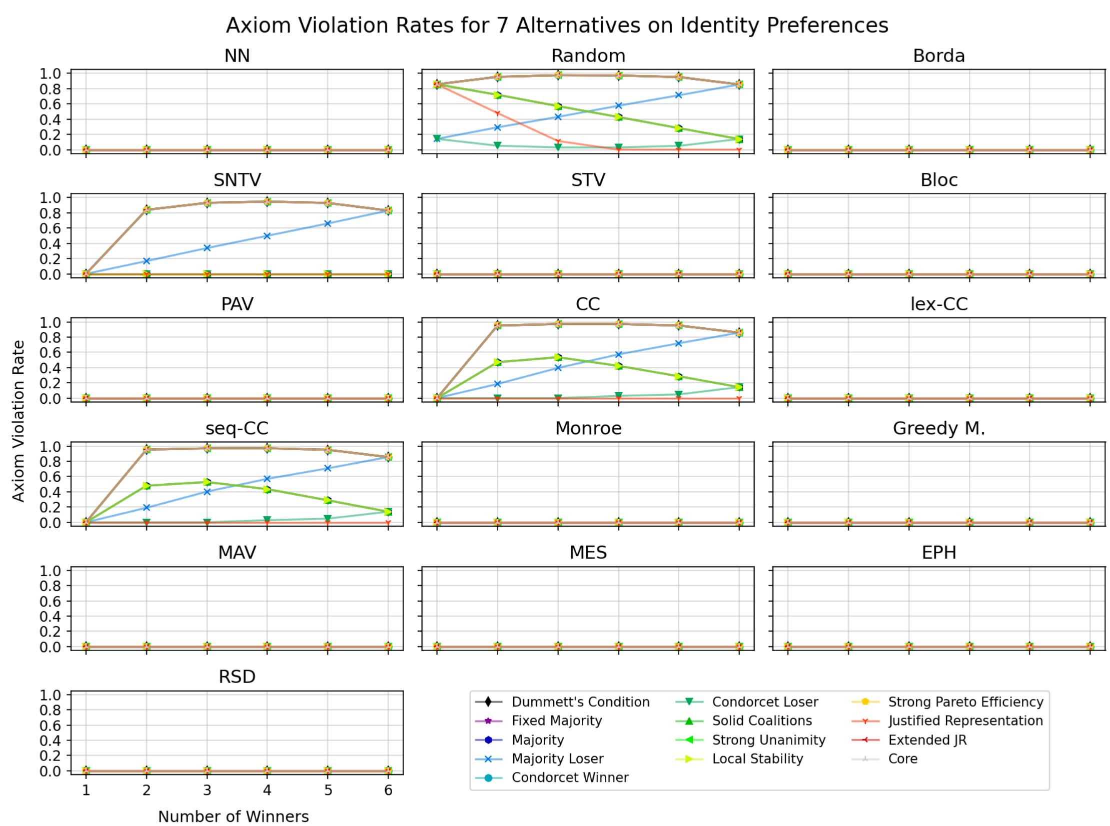


Figure D.81: Axiom violation rate for each rule on Identity preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.003**	*.000*	*.000*	*.000*	**.003**	**.013**	*.000*	**0**	**0**	.008	*.000*	*.000*	*.000*	.005	.016
Borda	.004	*.000*	*.000*	**0**	.004	.046	**0**	*.000*	**0**	.003	**0**	**0**	**0**	.001	.003
EPH	.011	**0**	**0**	**0**	.011	.104	*.000*	**0**	**0**	.012	**0**	**0**	**0**	.008	.014
SNTV	.054	**0**	.082	.052	.054	.359	*.000*	.129	.007	.039	**0**	.015	.019	**0**	*.000*
STV	.009	**0**	*.000*	*.000*	.009	.120	*.000*	*.000*	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
Bloc	.011	**0**	**0**	**0**	.011	.103	*.000*	**0**	**0**	.012	**0**	**0**	**0**	.008	.014
CC	.181	.090	.167	.199	.181	.655	.033	.226	.008	.319	**0**	.074	.085	.232	.269
lex-CC	.021	*.000*	*.000*	**0**	.021	.204	*.000*	.015	**0**	.018	**0**	**0**	**0**	.012	.022
seq-CC	.151	.048	.163	.169	.151	.636	.019	.223	.008	.240	**0**	.064	.072	.146	.174
Monroe	.099	.030	.052	.055	.099	.485	.025	.087	**0**	.197	**0**	.001	.001	.161	.194
Greedy M.	.024	**0**	.005	**0**	.024	.231	.001	.007	**0**	.023	**0**	**0**	**0**	.017	.027
PAV	.011	**0**	**0**	**0**	.011	.109	*.000*	**0**	**0**	.012	**0**	**0**	**0**	.008	.014
MES	.012	**0**	**0**	**0**	.012	.118	*.000*	**0**	**0**	.013	**0**	**0**	**0**	.009	.014
MAV	.168	.115	.109	.152	.168	.768	.054	.229	**0**	.229	.021	.027	.027	.177	.281
RSD	.078	.030	.030	**0**	.078	.543	.009	.088	**0**	.094	.002	.003	.003	.077	.140
Random	.266	.206	.207	.295	.266	.878	.066	.309	.009	.395	.057	.140	.155	.310	.435

Table D.85: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Mallows preferences.

D.3.6 7 Alternatives, Mallows

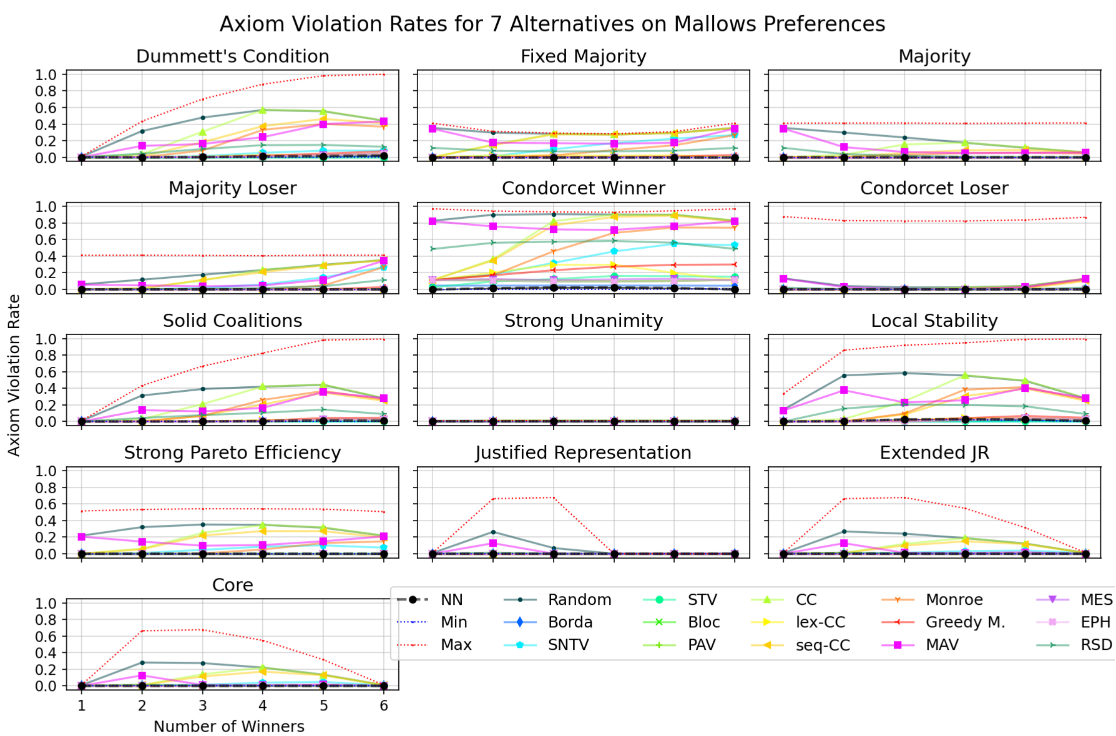


Figure D.82: Axiom violation rate for each axiom on Mallows preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.088	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.292	.857	.272	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.116	.858	.117	.225	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.129	.857	.118	.264	.157	–	–	–	–	–	–	–	–	–	–	–
PAV	.133	.857	.120	.265	.157	.018	–	–	–	–	–	–	–	–	–	–
CC	.490	.857	.505	.524	.499	.445	.439	–	–	–	–	–	–	–	–	–
lex-CC	.172	.857	.161	.285	.188	.073	.058	.421	–	–	–	–	–	–	–	–
seq-CC	.494	.858	.473	.375	.475	.440	.430	.623	.417	–	–	–	–	–	–	–
Monroe	.370	.857	.385	.428	.379	.324	.318	.136	.311	.550	–	–	–	–	–	–
Greedy M.	.210	.857	.195	.272	.211	.152	.145	.475	.164	.390	.355	–	–	–	–	–
MAV	.693	.859	.734	.792	.728	.722	.724	.477	.712	.914	.490	.750	–	–	–	–
MES	.141	.857	.125	.247	.161	.058	.045	.461	.080	.405	.340	.124	.734	–	–	–
EPH	.130	.857	.118	.264	.157	.003	.016	.445	.072	.439	.324	.152	.723	.057	–	–
RSD	.484	.858	.481	.555	.487	.476	.476	.679	.485	.659	.579	.490	.761	.476	.476	–
Min	.017	.857	.088	.291	.116	.131	.135	.492	.174	.493	.372	.211	.696	.143	.132	.484
Max	1.144	.856	1.133	1.113	1.132	1.132	1.130	.991	1.121	.979	1.058	1.115	.989	1.127	1.132	1.057

Table D.86: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Mallows preferences.

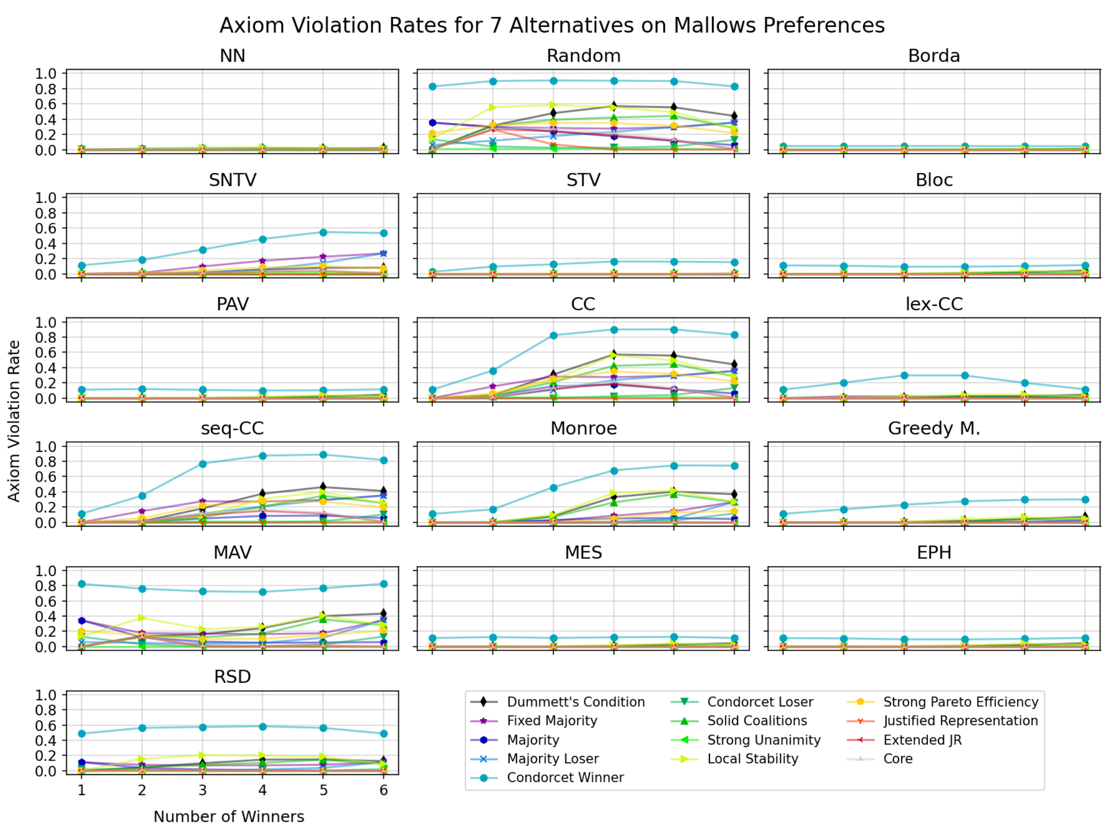


Figure D.83: Axiom violation rate for each rule on Mallows preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.034**	**0**	*.000*	*.000*	**.034**	**.001**	*.000*	*.000*	**0**	.099	.002	.002	.002	.069	.271
Borda	.043	**0**	.014	*.000*	.043	.100	**0**	.047	**0**	.101	.001	.001	.001	.072	.226
EPH	.064	**0**	.008	**0**	.064	.406	.008	.005	**0**	.102	*.000*	*.000*	*.000*	.071	.224
SNTV	.164	**0**	.344	.616	.164	.872	.029	.154	**0**	*.000*	.005	.007	.030	**0**	.079
STV	.138	**0**	.242	.558	.138	.835	.011	.136	**0**	**0**	**0**	**0**	.011	**0**	**.004**
Bloc	.064	**0**	.008	**0**	.064	.383	.008	**0**	**0**	.099	.001	.001	.001	.069	.256
CC	.199	**0**	.379	.557	.199	.912	.039	.170	**0**	.149	**0**	.006	.033	.111	.230
lex-CC	.074	**0**	.008	**0**	.074	.519	.008	.044	**0**	.120	**0**	**0**	**0**	.089	.179
seq-CC	.192	**0**	.348	.511	.192	.896	.035	.169	**0**	.151	**0**	.001	.024	.112	.245
Monroe	.188	**0**	.317	.542	.188	.893	.038	.170	**0**	.147	**0**	*.000*	.025	.110	.199
Greedy M.	.102	**0**	.093	.076	.102	.624	.008	.095	**0**	.132	**0**	**0**	**0**	.100	.197
PAV	.068	**0**	.008	**0**	.068	.464	.008	.027	**0**	.111	**0**	**0**	**0**	.081	.186
MES	.073	**0**	.008	**0**	.073	.509	.008	.043	**0**	.119	**0**	**0**	**0**	.088	.179
MAV	.201	**0**	.370	.615	.201	.894	.048	.172	**0**	.149	.003	.003	.003	.111	.241
RSD	.171	**0**	.247	**0**	.171	.721	.027	.084	**0**	.137	.100	.102	.102	.102	.602
Random	.223	**0**	.443	.614	.223	.918	.064	.170	**0**	.150	.021	.027	.053	.111	.331

Table D.87: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across SP Conitzer preferences.

D.3.7 7 Alternatives, SP Conitzer

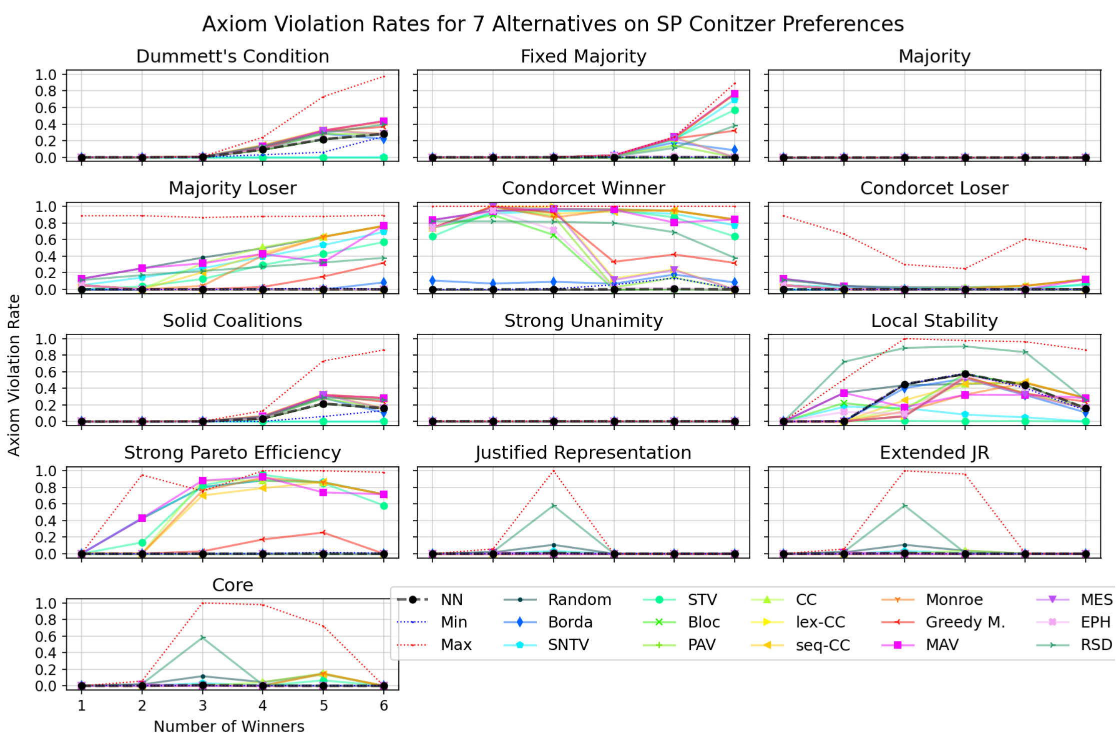


Figure D.84: Axiom violation rate for each axiom on SP Conitzer preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.132	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.780	.856	.773	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.684	.856	.669	.450	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.408	.856	.453	.568	.628	–	–	–	–	–	–	–	–	–	–	–
PAV	.457	.856	.460	.590	.637	.070	–	–	–	–	–	–	–	–	–	–
CC	.821	.855	.825	.650	.710	.474	.445	–	–	–	–	–	–	–	–	–
lex-CC	.479	.856	.470	.597	.641	.092	.023	.444	–	–	–	–	–	–	–	–
seq-CC	.812	.857	.795	.526	.736	.533	.504	.719	.502	–	–	–	–	–	–	–
Monroe	.790	.856	.794	.639	.698	.452	.425	.067	.426	.694	–	–	–	–	–	–
Greedy M.	.565	.857	.543	.530	.666	.257	.210	.559	.202	.387	.540	–	–	–	–	–
MAV	.775	.856	.787	.896	.818	.767	.768	.494	.766	1.018	.536	.836	–	–	–	–
MES	.488	.857	.465	.525	.647	.180	.122	.535	.109	.413	.516	.112	.800	–	–	–
EPH	.425	.856	.461	.578	.633	.026	.053	.466	.075	.521	.444	.242	.770	.163	–	–
RSD	.681	.856	.682	.810	.792	.646	.648	.815	.650	.797	.804	.660	.812	.648	.647	–
Min	.026	.857	.144	.774	.677	.427	.475	.819	.496	.812	.789	.575	.778	.501	.444	.683
Max	1.107	.857	1.095	.957	1.016	1.098	1.092	.963	1.084	.901	.989	1.065	.940	1.081	1.096	.972

Table D.88: Difference between rules for 7 alternatives with $1 \leq k < 7$ on SP Conitzer preferences.

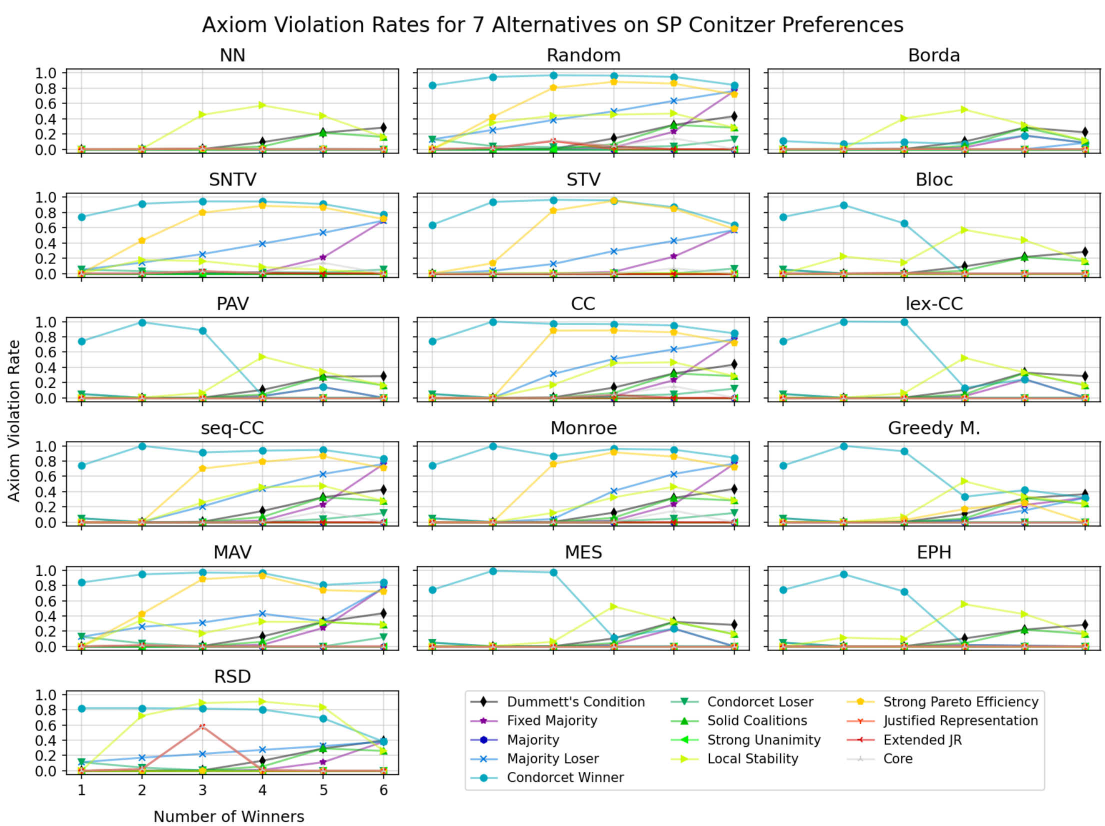


Figure D.85: Axiom violation rate for each rule on SP Conitzer preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	*.000*	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
Borda	.007	**0**	.022	**0**	.007	.051	**0**	.022	**0**	*.000*	**0**	**0**	**0**	**0**	**0**
EPH	.005	**0**	**0**	**0**	.005	.062	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
SNTV	.034	**0**	.074	.029	.034	.264	**0**	.078	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
STV	.021	**0**	.037	.004	.021	.193	**0**	.038	**0**	**0**	**0**	**0**	**0**	**0**	*.000*
Bloc	.005	**0**	**0**	**0**	.005	.059	**0**	**0**	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
CC	.252	.001	.368	.558	.252	.839	.036	.211	**0**	.349	**0**	.101	.124	.310	.384
lex-CC	.021	*.000*	**0**	**0**	.021	.259	**0**	.005	**0**	.001	**0**	**0**	**0**	.001	.001
seq-CC	.227	*.000*	.356	.553	.227	.817	.030	.209	**0**	.282	**0**	.069	.081	.246	.314
Monroe	.180	*.000*	.203	.409	.180	.730	.028	.199	**0**	.262	**0**	.008	.021	.234	.240
Greedy M.	.027	**0**	.055	.003	.027	.231	**0**	.057	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
PAV	.008	*.000*	**0**	**0**	.008	.101	**0**	*.000*	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
MES	.006	**0**	**0**	**0**	.006	.084	**0**	*.000*	**0**	*.000*	**0**	**0**	**0**	*.000*	*.000*
MAV	.253	.001	.367	.648	.253	.886	.052	.196	**0**	.237	.061	.129	.129	.198	.382
RSD	.087	*.000*	.127	**0**	.087	.555	.002	.103	**0**	.040	.035	.037	.040	.028	.169
Random	.306	.001	.443	.648	.306	.921	.068	.210	**0**	.357	.087	.181	.215	.317	.526

Table D.89: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across SP Walsh preferences.

D.3.8 7 Alternatives, SP Walsh

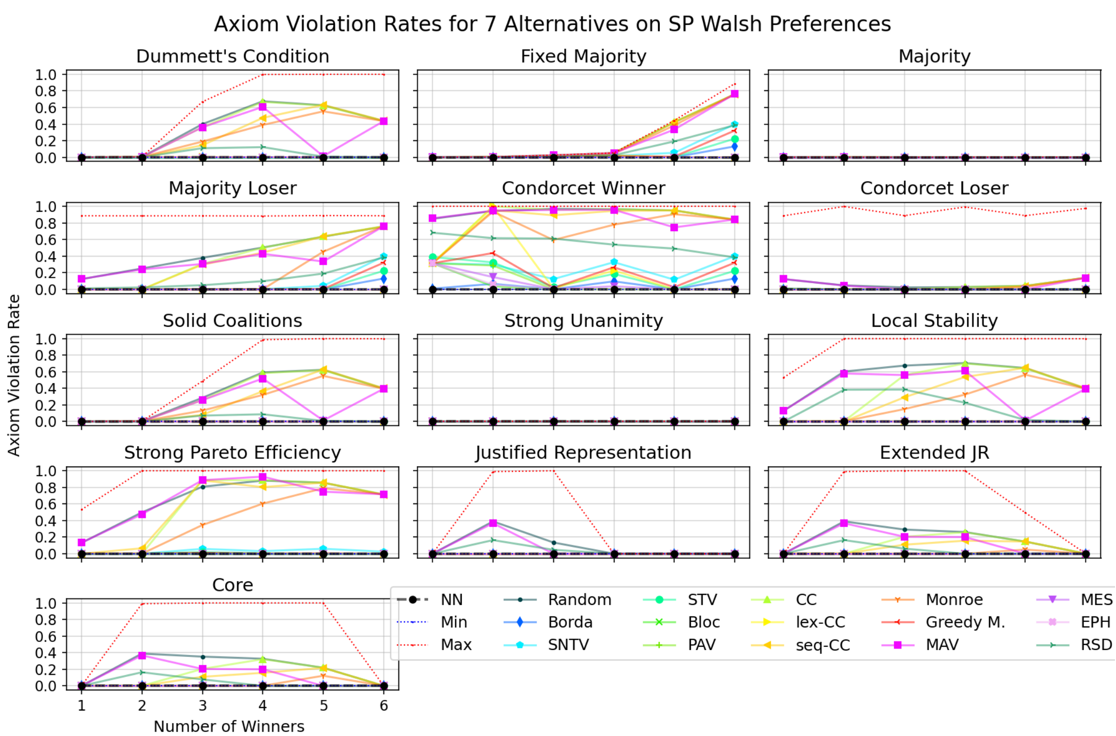


Figure D.86: Axiom violation rate for each axiom on SP Walsh preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.856	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.057	.856	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.255	.857	.232	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.223	.856	.211	.210	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.114	.856	.164	.176	.184	–	–	–	–	–	–	–	–	–	–	–
PAV	.137	.856	.177	.182	.182	.028	–	–	–	–	–	–	–	–	–	–
CC	.629	.858	.639	.563	.598	.522	.504	–	–	–	–	–	–	–	–	–
lex-CC	.207	.856	.239	.222	.219	.099	.076	.440	–	–	–	–	–	–	–	–
seq-CC	.612	.856	.610	.490	.609	.533	.542	.808	.593	–	–	–	–	–	–	–
Monroe	.509	.858	.518	.443	.477	.401	.383	.155	.334	.710	–	–	–	–	–	–
Greedy M.	.236	.856	.236	.163	.239	.156	.160	.573	.205	.464	.455	–	–	–	–	–
MAV	.774	.858	.782	.803	.774	.771	.770	.499	.769	1.019	.593	.798	–	–	–	–
MES	.149	.856	.166	.142	.187	.069	.069	.553	.138	.490	.432	.109	.787	–	–	–
EPH	.116	.856	.163	.176	.184	.004	.027	.522	.099	.533	.401	.154	.771	.065	–	–
RSD	.495	.856	.497	.511	.503	.488	.490	.757	.507	.740	.671	.503	.796	.490	.488	–
Min	.000	.856	.058	.255	.223	.114	.137	.629	.207	.612	.509	.236	.774	.150	.116	.495
Max	1.166	.859	1.166	1.162	1.165	1.166	1.166	.969	1.166	.976	1.063	1.165	.932	1.166	1.166	1.119

Table D.90: Difference between rules for 7 alternatives with $1 \leq k < 7$ on SP Walsh preferences.

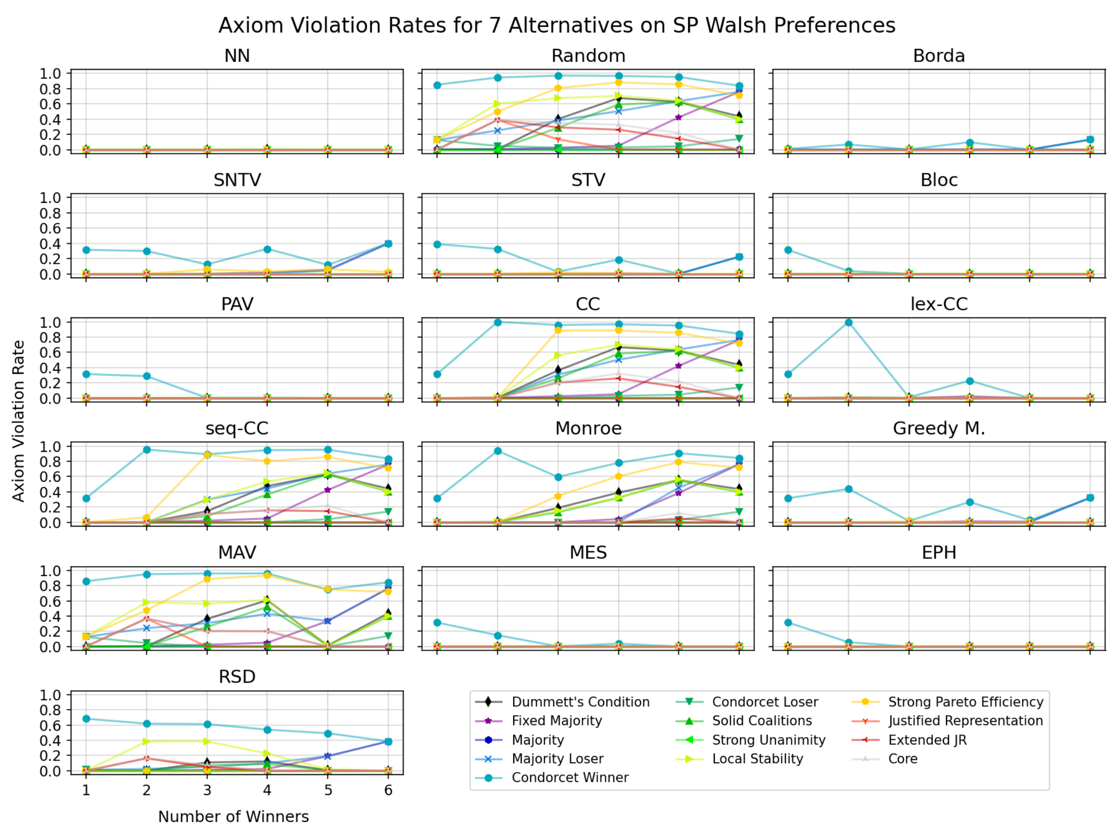


Figure D.87: Axiom violation rate for each rule on SP Walsh preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.017**	*.000*	*.000*	.002	**.017**	**.011**	*.000*	**0**	**0**	.069	*.000*	*.000*	*.000*	.051	.087
Borda	.024	*.000*	.003	.001	.024	.156	**0**	.006	**0**	.051	*.000*	*.000*	*.000*	.036	.058
EPH	.044	*.000*	.001	*.000*	.044	.309	.002	.001	**0**	.090	*.000*	*.000*	*.000*	.069	.097
SNTV	.082	**0**	.065	.222	.082	.711	.009	.040	**0**	.002	*.000*	.001	.005	**0**	.012
STV	.059	**0**	.036	.133	.059	.568	.001	.027	**0**	**0**	**0**	*.000*	.001	**0**	**.001**
Bloc	.042	**0**	.001	**0**	.042	.287	.002	**0**	**0**	.086	*.000*	*.000*	*.000*	.066	.108
CC	.166	.003	.114	.329	.166	.824	.033	.056	**0**	.263	**0**	.006	.012	.222	.301
lex-CC	.075	.001	.006	**0**	.075	.564	.002	.017	**0**	.144	**0**	**0**	**0**	.116	.129
seq-CC	.154	.002	.105	.252	.154	.806	.026	.056	**0**	.252	**0**	.003	.003	.212	.282
Monroe	.151	.002	.088	.287	.151	.787	.030	.054	**0**	.243	**0**	.001	.005	.205	.263
Greedy M.	.074	*.000*	.017	.011	.074	.545	.002	.019	**0**	.133	**0**	**0**	**0**	.105	.129
PAV	.049	*.000*	.001	**0**	.049	.364	.002	.002	**0**	.098	**0**	**0**	**0**	.075	.093
MES	.057	*.000*	.002	.002	.057	.433	.002	.007	**0**	.112	**0**	**0**	**0**	.088	.101
MAV	.175	.005	.111	.307	.175	.879	.052	.058	**0**	.249	.018	.020	.020	.208	.342
RSD	.125	.002	.063	**0**	.125	.708	.018	.027	**0**	.181	.035	.037	.038	.150	.363
Random	.195	.005	.138	.359	.195	.898	.063	.057	**0**	.266	.031	.038	.046	.225	.414

Table D.91: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Gaussian Ball 3 preferences.

D.3.9 7 Alternatives, Gaussian Ball 3

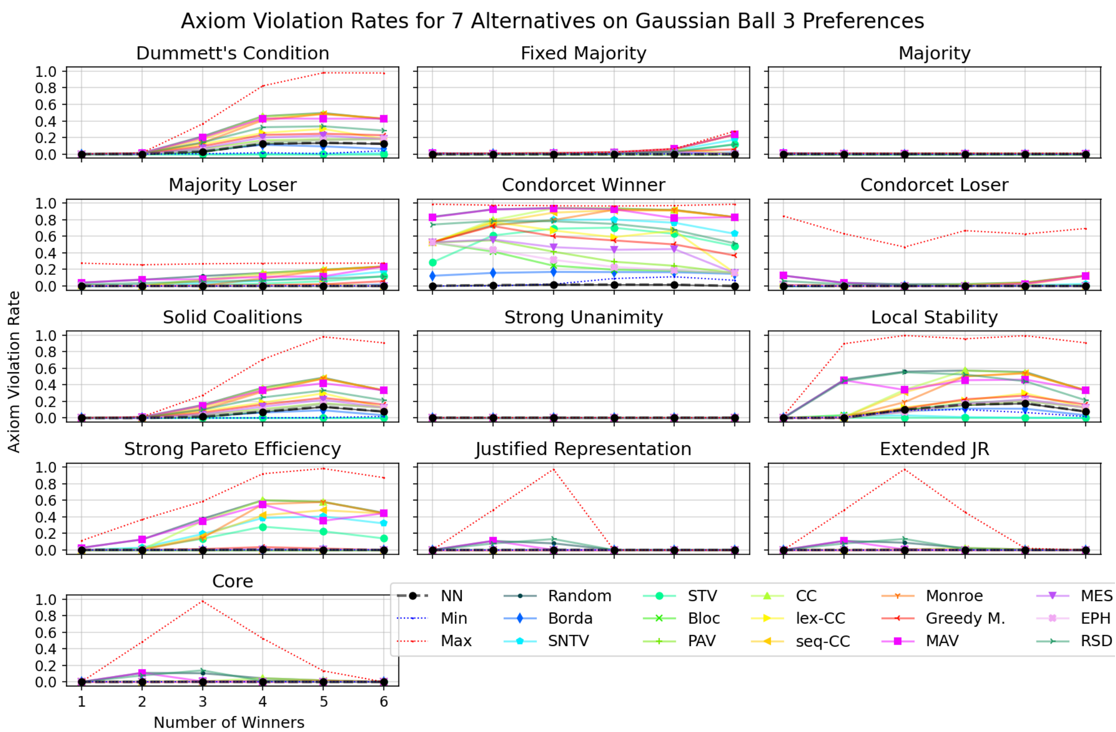


Figure D.88: Axiom violation rate for each axiom on Gaussian Ball 3 preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.154	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.591	.857	.604	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.418	.858	.423	.363	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.316	.856	.374	.429	.468	–	–	–	–	–	–	–	–	–	–	–
PAV	.349	.857	.381	.448	.475	.081	–	–	–	–	–	–	–	–	–	–
CC	.691	.857	.728	.586	.658	.495	.466	–	–	–	–	–	–	–	–	–
lex-CC	.434	.857	.459	.483	.519	.184	.116	.439	–	–	–	–	–	–	–	–
seq-CC	.682	.858	.693	.513	.648	.507	.476	.668	.461	–	–	–	–	–	–	–
Monroe	.650	.857	.686	.556	.623	.452	.420	.086	.410	.643	–	–	–	–	–	–
Greedy M.	.452	.858	.471	.445	.519	.263	.223	.520	.236	.391	.486	–	–	–	–	–
MAV	.789	.856	.807	.864	.817	.799	.794	.470	.780	.980	.528	.833	–	–	–	–
MES	.381	.858	.407	.422	.487	.163	.106	.508	.146	.429	.463	.175	.807	–	–	–
EPH	.325	.857	.372	.434	.469	.026	.064	.490	.170	.497	.446	.252	.798	.148	–	–
RSD	.648	.856	.660	.725	.702	.613	.617	.775	.637	.754	.753	.641	.829	.623	.614	–
Min	.052	.857	.169	.582	.407	.345	.376	.683	.454	.691	.643	.470	.782	.405	.353	.657
Max	1.129	.855	1.125	1.037	1.080	1.122	1.109	.974	1.084	.930	1.010	1.081	.938	1.101	1.117	1.003

Table D.92: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Gaussian Ball 3 preferences.

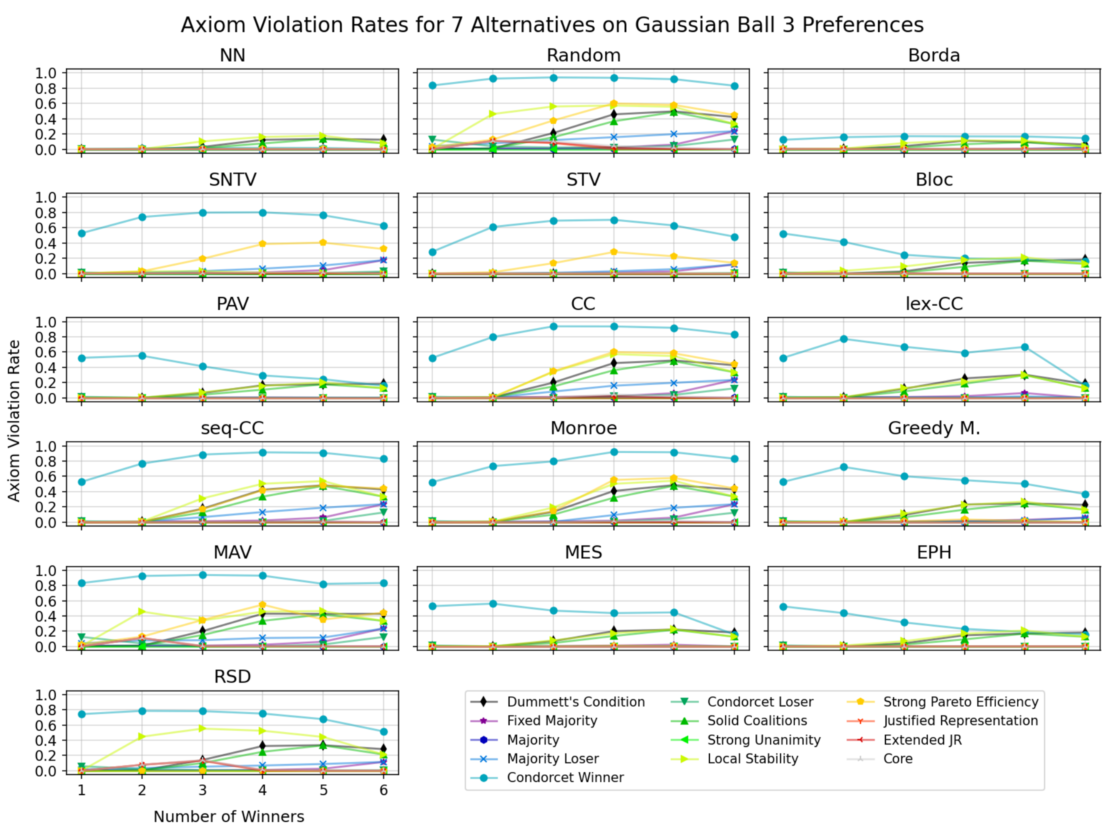


Figure D.89: Axiom violation rate for each rule on Gaussian Ball 3 preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.017**	**0**	*.000*	.002	**.017**	**.010**	*.000*	**0**	**0**	.069	*.000*	*.000*	*.000*	.051	.088
Borda	.024	.001	.003	.001	.024	.154	**0**	.006	**0**	.051	*.000*	*.000*	*.000*	.036	.059
EPH	.044	*.000*	.001	*.000*	.044	.311	.002	.001	**0**	.090	*.000*	*.000*	*.000*	.069	.099
SNTV	.082	**0**	.065	.222	.082	.714	.009	.040	**0**	.002	*.000*	.001	.005	**0**	.012
STV	.059	**0**	.035	.133	.059	.571	.002	.027	**0**	**0**	**0**	*.000*	.001	**0**	**.001**
Bloc	.043	**0**	.001	**0**	.043	.290	.002	**0**	**0**	.087	*.000*	*.000*	*.000*	.066	.108
CC	.166	.003	.113	.329	.166	.828	.034	.056	**0**	.262	**0**	.006	.012	.221	.299
lex-CC	.075	.001	.005	**0**	.075	.567	.002	.017	**0**	.143	**0**	**0**	**0**	.115	.127
seq-CC	.154	.002	.105	.255	.154	.810	.026	.055	**0**	.250	**0**	.002	.003	.210	.280
Monroe	.152	.002	.088	.287	.152	.791	.031	.054	**0**	.244	**0**	.001	.004	.205	.262
Greedy M.	.074	*.000*	.016	.011	.074	.549	.002	.019	**0**	.133	**0**	**0**	**0**	.106	.130
PAV	.049	*.000*	.001	**0**	.049	.367	.002	.002	**0**	.098	**0**	**0**	**0**	.075	.094
MES	.057	*.000*	.001	.002	.057	.434	.002	.006	**0**	.111	**0**	**0**	**0**	.087	.100
MAV	.174	.005	.109	.307	.174	.879	.053	.057	**0**	.248	.019	.020	.020	.207	.341
RSD	.126	.002	.064	**0**	.126	.709	.020	.027	**0**	.182	.036	.038	.040	.150	.366
Random	.195	.005	.137	.358	.195	.898	.063	.057	**0**	.267	.032	.038	.046	.224	.413

Table D.93: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Gaussian Ball 10 preferences.

D.3.10 7 Alternatives, Gaussian Ball 10

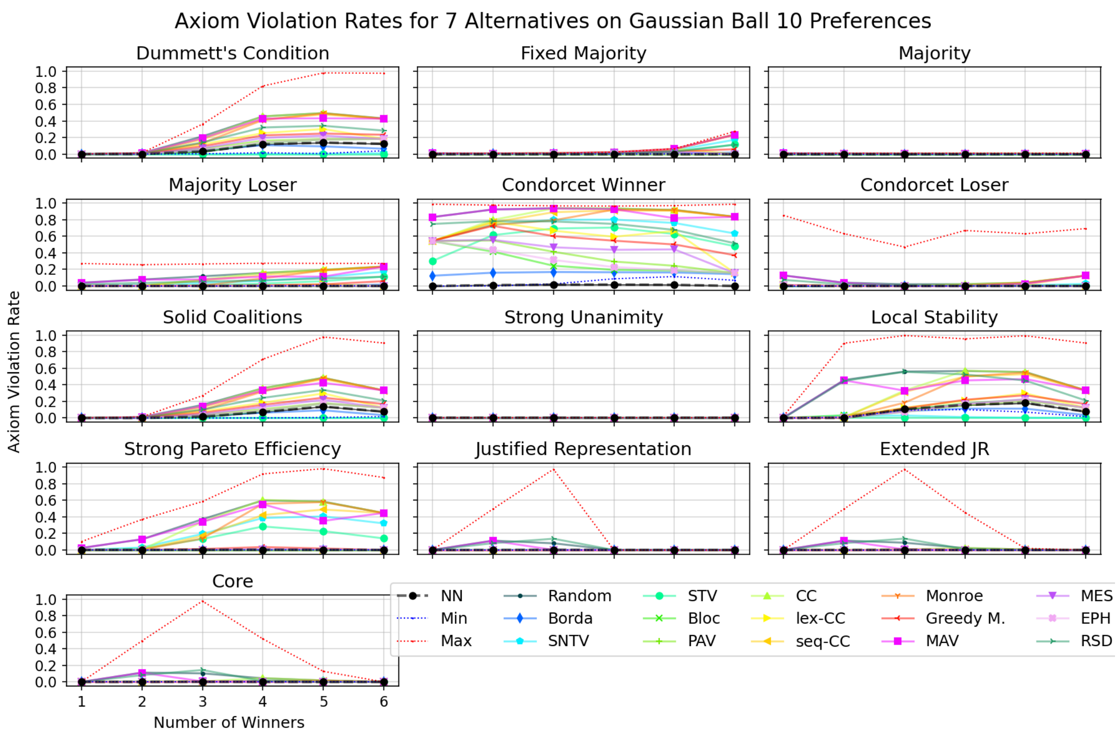


Figure D.90: Axiom violation rate for each axiom on Gaussian Ball 10 preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.154	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.597	.858	.608	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.424	.856	.427	.363	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.319	.858	.375	.431	.468	–	–	–	–	–	–	–	–	–	–	–
PAV	.352	.858	.382	.450	.475	.082	–	–	–	–	–	–	–	–	–	–
CC	.696	.858	.731	.588	.659	.495	.465	–	–	–	–	–	–	–	–	–
lex-CC	.437	.858	.461	.484	.518	.185	.116	.439	–	–	–	–	–	–	–	–
seq-CC	.688	.857	.698	.512	.650	.510	.478	.669	.463	–	–	–	–	–	–	–
Monroe	.655	.858	.689	.558	.625	.452	.420	.085	.409	.645	–	–	–	–	–	–
Greedy M.	.457	.857	.475	.446	.521	.265	.225	.520	.237	.393	.487	–	–	–	–	–
MAV	.789	.858	.806	.863	.814	.796	.792	.469	.778	.980	.526	.830	–	–	–	–
MES	.385	.858	.409	.423	.488	.164	.106	.509	.147	.430	.464	.176	.805	–	–	–
EPH	.328	.858	.373	.436	.469	.026	.064	.490	.171	.500	.446	.254	.796	.149	–	–
RSD	.652	.858	.660	.732	.707	.620	.625	.780	.643	.760	.759	.646	.829	.629	.621	–
Min	.051	.858	.170	.588	.413	.349	.380	.687	.458	.697	.648	.476	.783	.409	.357	.660
Max	1.129	.856	1.125	1.037	1.080	1.122	1.109	.973	1.083	.930	1.008	1.081	.937	1.102	1.117	1.000

Table D.94: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Gaussian Ball 10 preferences.

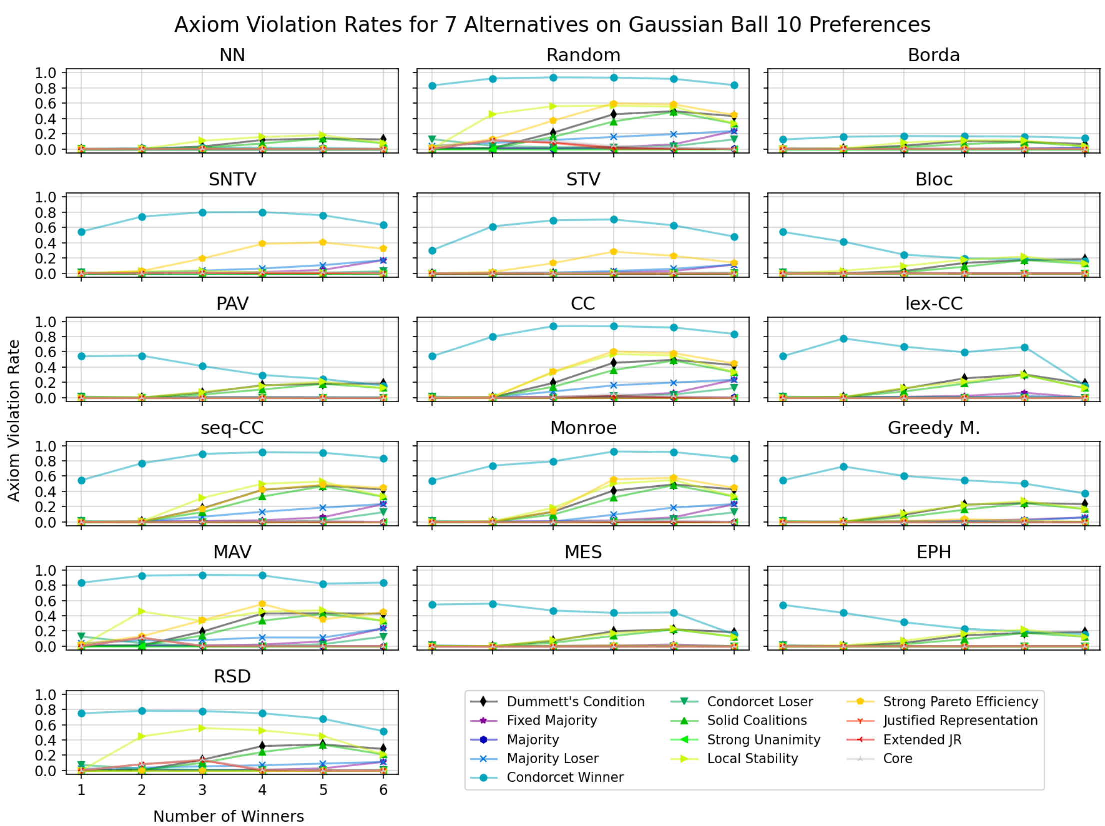


Figure D.91: Axiom violation rate for each rule on Gaussian Ball 10 preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.023**	**0**	*.000*	.003	**.023**	**.016**	*.000*	*.000*	**0**	.092	.001	.001	.001	.070	.117
Borda	.029	*.000*	.002	.002	.029	.179	**0**	.004	**0**	.066	*.000*	*.000*	*.000*	.048	.075
EPH	.056	**0**	.001	*.000*	.056	.372	.003	.001	**0**	.125	*.000*	*.000*	*.000*	.099	.133
SNTV	.084	**0**	.044	.241	.084	.741	.012	.024	**0**	.002	*.000*	.001	.005	**0**	.014
STV	.063	**0**	.026	.156	.063	.615	.003	.017	**0**	**0**	**0**	*.000*	.001	**0**	**.001**
Bloc	.055	**0**	.001	**0**	.055	.345	.003	**0**	**0**	.120	.001	.001	.001	.095	.149
CC	.157	.002	.061	.311	.157	.825	.032	.029	**0**	.256	**0**	.004	.007	.215	.296
lex-CC	.083	.001	.004	**0**	.083	.598	.003	.009	**0**	.173	**0**	**0**	**0**	.141	.154
seq-CC	.147	.002	.056	.229	.147	.810	.027	.030	**0**	.253	**0**	.001	.002	.212	.284
Monroe	.146	.002	.050	.278	.146	.798	.030	.029	**0**	.242	**0**	*.000*	.003	.203	.265
Greedy M.	.086	*.000*	.011	.013	.086	.600	.003	.012	**0**	.169	**0**	**0**	**0**	.138	.167
PAV	.062	*.000*	.001	**0**	.062	.433	.003	.002	**0**	.135	**0**	**0**	**0**	.107	.127
MES	.071	*.000*	.001	.003	.071	.505	.003	.005	**0**	.152	**0**	**0**	**0**	.121	.135
MAV	.166	.003	.059	.298	.166	.875	.049	.030	**0**	.249	.015	.016	.016	.208	.339
RSD	.132	.002	.039	**0**	.132	.729	.021	.016	**0**	.205	.040	.042	.043	.172	.408
Random	.182	.003	.072	.335	.182	.887	.060	.030	**0**	.260	.028	.031	.036	.218	.402

Table D.95: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Uniform Ball 3 preferences.

D.3.11 7 Alternatives, Uniform Ball 3

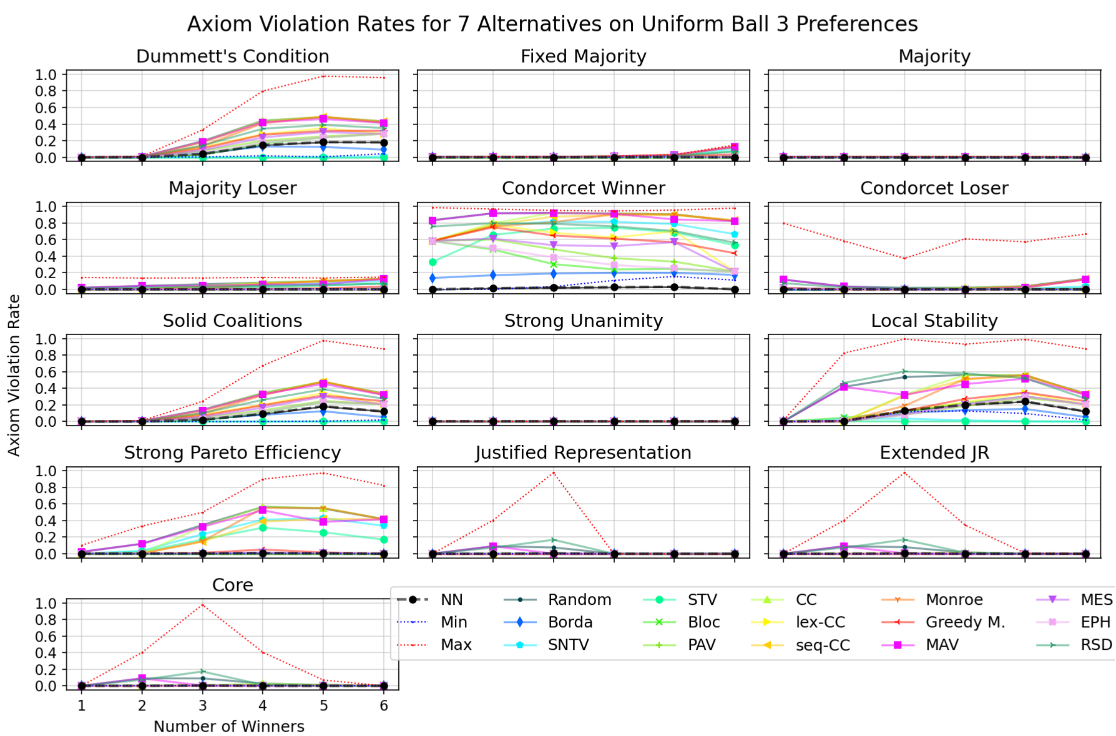


Figure D.92: Axiom violation rate for each axiom on Uniform Ball 3 preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.180	.856	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.638	.855	.648	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.470	.857	.471	.382	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.369	.856	.426	.463	.524	–	–	–	–	–	–	–	–	–	–	–
PAV	.406	.856	.435	.486	.534	.096	–	–	–	–	–	–	–	–	–	–
CC	.708	.856	.748	.594	.675	.493	.460	–	–	–	–	–	–	–	–	–
lex-CC	.477	.857	.500	.513	.569	.184	.103	.439	–	–	–	–	–	–	–	–
seq-CC	.708	.856	.715	.524	.672	.513	.477	.669	.463	–	–	–	–	–	–	–
Monroe	.676	.856	.715	.572	.650	.459	.424	.077	.414	.647	–	–	–	–	–	–
Greedy M.	.506	.856	.523	.479	.574	.289	.245	.523	.253	.388	.496	–	–	–	–	–
MAV	.793	.856	.816	.871	.824	.803	.799	.460	.790	.984	.516	.843	–	–	–	–
MES	.440	.855	.463	.458	.546	.185	.120	.508	.142	.427	.473	.195	.817	–	–	–
EPH	.380	.856	.424	.470	.525	.035	.073	.486	.166	.500	.452	.275	.803	.167	–	–
RSD	.684	.857	.693	.754	.736	.646	.652	.783	.665	.764	.768	.671	.835	.656	.648	–
Min	.074	.858	.200	.626	.454	.410	.446	.695	.508	.720	.666	.530	.783	.474	.421	.693
Max	1.114	.856	1.109	1.015	1.060	1.103	1.086	.971	1.063	.913	1.000	1.054	.941	1.076	1.095	.979

Table D.96: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Uniform Ball 3 preferences.

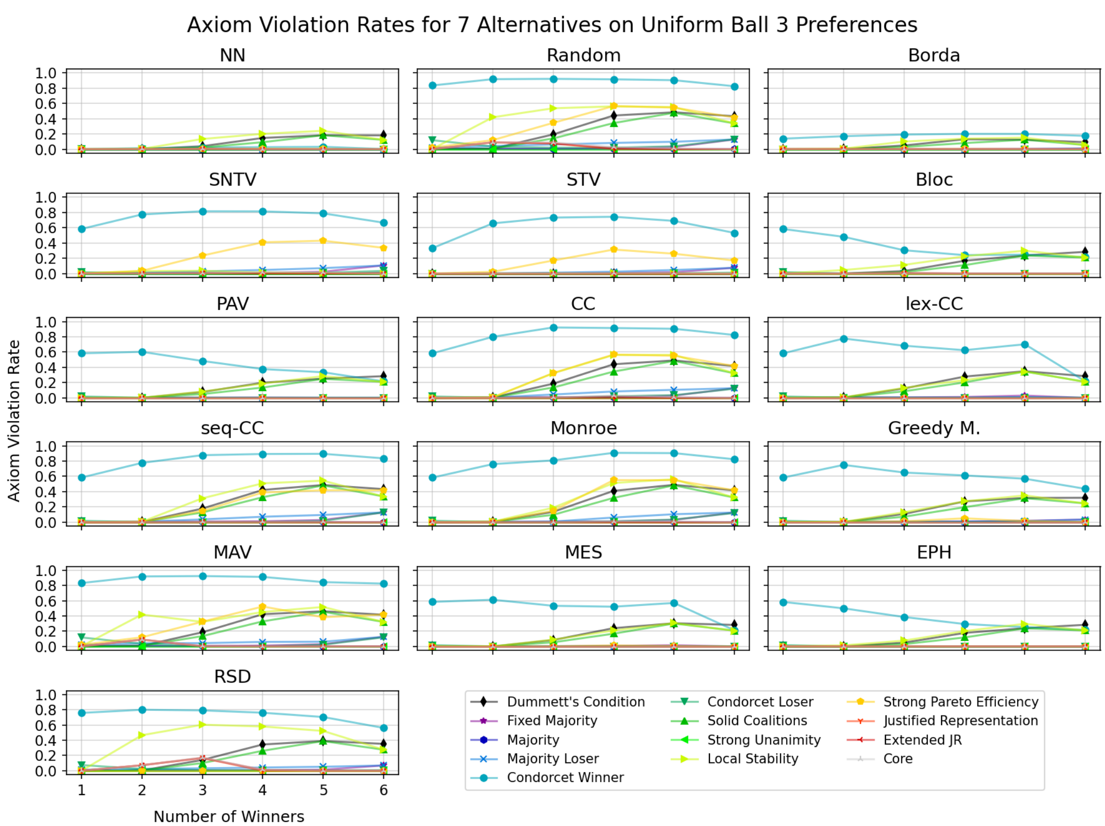


Figure D.93: Axiom violation rate for each rule on Uniform Ball 3 preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.022**	*.000*	*.000*	.004	**.022**	**.015**	*.000*	*.000*	**0**	.089	.001	.001	.001	.068	.114
Borda	.029	*.000*	.002	.002	.029	.180	**0**	.004	**0**	.064	*.000*	*.000*	*.000*	.046	.073
EPH	.056	*.000*	.001	*.000*	.056	.376	.003	.001	**0**	.123	**0**	**0**	**0**	.097	.131
SNTV	.083	**0**	.042	.240	.083	.740	.013	.023	**0**	.002	*.000*	.001	.005	**0**	.015
STV	.063	**0**	.025	.156	.063	.613	.002	.017	**0**	**0**	**0**	*.000*	.001	**0**	**.001**
Bloc	.055	*.000*	.001	**0**	.055	.349	.003	**0**	**0**	.119	.001	.001	.001	.094	.148
CC	.156	.002	.058	.309	.156	.824	.033	.029	**0**	.257	**0**	.003	.007	.216	.294
lex-CC	.083	.001	.003	**0**	.083	.600	.003	.010	**0**	.171	**0**	**0**	**0**	.139	.151
seq-CC	.146	.002	.055	.230	.146	.807	.026	.030	**0**	.252	**0**	.001	.002	.212	.283
Monroe	.146	.002	.049	.277	.146	.799	.032	.029	**0**	.243	**0**	*.000*	.003	.205	.263
Greedy M.	.085	*.000*	.010	.013	.085	.601	.003	.012	**0**	.168	**0**	**0**	**0**	.136	.165
PAV	.062	*.000*	.001	**0**	.062	.436	.003	.002	**0**	.133	**0**	**0**	**0**	.105	.125
MES	.071	*.000*	.002	.003	.071	.510	.003	.005	**0**	.150	**0**	**0**	**0**	.120	.134
MAV	.166	.004	.056	.295	.166	.876	.051	.031	**0**	.250	.015	.015	.015	.208	.337
RSD	.132	.002	.037	**0**	.132	.725	.021	.015	**0**	.205	.042	.044	.044	.171	.405
Random	.181	.004	.070	.335	.181	.886	.060	.031	**0**	.257	.027	.031	.036	.217	.400

Table D.97: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Uniform Ball 10 preferences.

D.3.12 7 Alternatives, Uniform Ball 10

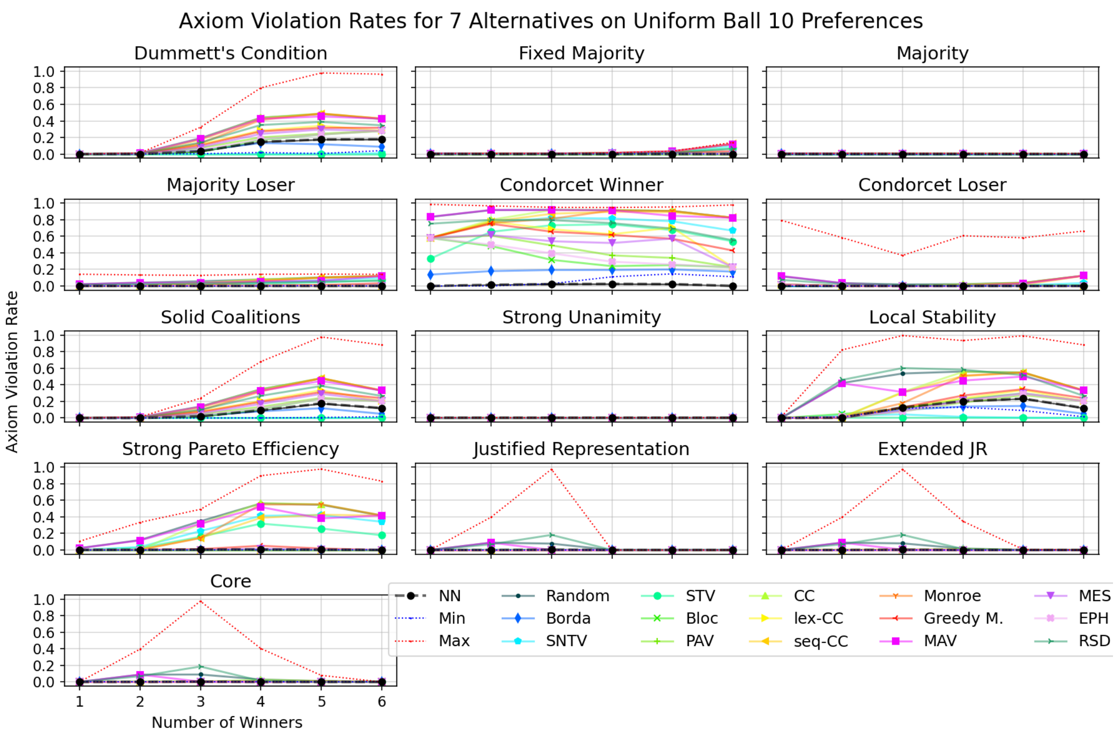


Figure D.94: Axiom violation rate for each axiom on Uniform Ball 10 preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.179	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.639	.857	.649	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.470	.856	.471	.381	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.371	.857	.429	.465	.523	–	–	–	–	–	–	–	–	–	–	–
PAV	.408	.856	.437	.487	.533	.096	–	–	–	–	–	–	–	–	–	–
CC	.709	.856	.748	.595	.673	.494	.461	–	–	–	–	–	–	–	–	–
lex-CC	.479	.856	.502	.515	.568	.185	.103	.439	–	–	–	–	–	–	–	–
seq-CC	.706	.857	.714	.523	.669	.512	.477	.670	.465	–	–	–	–	–	–	–
Monroe	.678	.857	.715	.575	.650	.461	.425	.077	.413	.648	–	–	–	–	–	–
Greedy M.	.506	.856	.523	.480	.574	.289	.244	.524	.251	.387	.497	–	–	–	–	–
MAV	.796	.856	.818	.871	.826	.804	.800	.460	.791	.986	.517	.845	–	–	–	–
MES	.444	.857	.467	.460	.548	.187	.121	.509	.143	.425	.474	.193	.818	–	–	–
EPH	.381	.857	.426	.472	.524	.036	.073	.487	.166	.499	.453	.274	.805	.168	–	–
RSD	.680	.856	.692	.756	.736	.646	.651	.786	.666	.766	.771	.672	.837	.655	.647	–
Min	.073	.857	.199	.627	.454	.411	.445	.697	.508	.719	.668	.530	.786	.476	.420	.690
Max	1.115	.859	1.109	1.014	1.060	1.103	1.086	.968	1.064	.914	.997	1.054	.939	1.076	1.095	.979

Table D.98: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Uniform Ball 10 preferences.

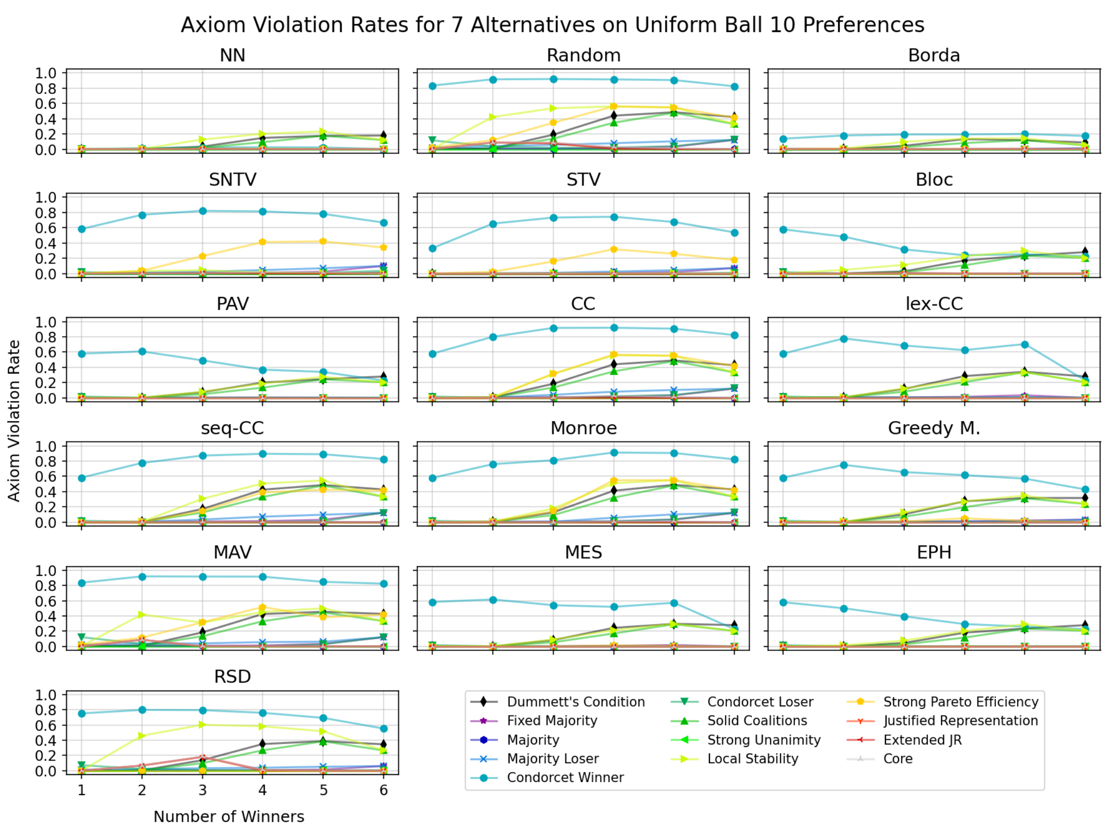


Figure D.95: Axiom violation rate for each rule on Uniform Ball 10 preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.021**	*.000*	*.000*	.004	**.021**	**.015**	*.000*	*.000*	**0**	.082	.001	.001	.001	.062	.106
Borda	.027	.001	.002	.002	.027	.170	**0**	.005	**0**	.061	*.000*	*.000*	*.000*	.044	.070
EPH	.051	*.000*	.001	*.000*	.051	.347	.002	.001	**0**	.110	*.000*	*.000*	*.000*	.086	.118
SNTV	.085	**0**	.058	.242	.085	.734	.010	.033	**0**	.002	.001	.002	.005	**0**	.015
STV	.062	**0**	.032	.156	.062	.595	.002	.023	**0**	**0**	**0**	*.000*	.001	**0**	**.001**
Bloc	.050	*.000*	.001	**0**	.050	.321	.002	**0**	**0**	.105	.001	.001	.001	.083	.132
CC	.165	.003	.098	.332	.165	.826	.033	.046	**0**	.264	**0**	.005	.010	.224	.302
lex-CC	.079	.001	.004	**0**	.079	.578	.002	.012	**0**	.157	**0**	**0**	**0**	.128	.141
seq-CC	.152	.002	.092	.249	.152	.807	.026	.047	**0**	.252	**0**	.002	.003	.212	.283
Monroe	.152	.002	.078	.289	.152	.795	.031	.046	**0**	.248	**0**	.001	.004	.211	.269
Greedy M.	.081	*.000*	.014	.012	.081	.572	.003	.016	**0**	.156	**0**	**0**	**0**	.126	.152
PAV	.056	*.000*	.001	**0**	.056	.404	.002	.002	**0**	.118	**0**	**0**	**0**	.093	.112
MES	.065	*.000*	.001	.003	.065	.475	.002	.006	**0**	.133	**0**	**0**	**0**	.106	.120
MAV	.173	.004	.097	.318	.173	.874	.052	.048	**0**	.252	.016	.018	.018	.212	.341
RSD	.130	.002	.056	**0**	.130	.716	.021	.024	**0**	.197	.039	.041	.042	.165	.393
Random	.192	.004	.121	.364	.192	.890	.062	.048	**0**	.265	.030	.035	.042	.224	.410

Table D.99: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Gaussian Cube 3 preferences.

D.3.13 7 Alternatives, Gaussian Cube 3

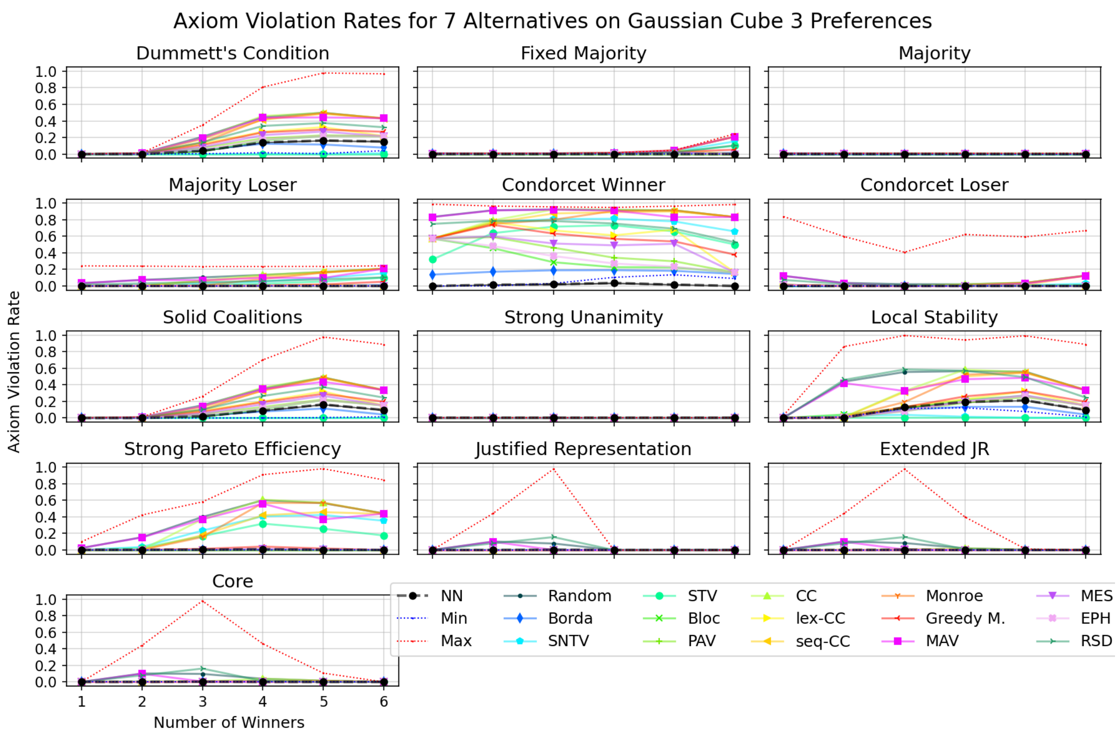


Figure D.96: Axiom violation rate for each axiom on Gaussian Cube 3 preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.168	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.627	.859	.640	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.451	.859	.455	.378	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.346	.858	.403	.451	.501	–	–	–	–	–	–	–	–	–	–	–
PAV	.382	.858	.412	.473	.511	.088	–	–	–	–	–	–	–	–	–	–
CC	.707	.857	.746	.594	.674	.495	.464	–	–	–	–	–	–	–	–	–
lex-CC	.456	.858	.479	.503	.547	.182	.107	.441	–	–	–	–	–	–	–	–
seq-CC	.700	.857	.708	.521	.664	.509	.475	.669	.462	–	–	–	–	–	–	–
Monroe	.672	.857	.709	.569	.644	.458	.424	.079	.413	.645	–	–	–	–	–	–
Greedy M.	.482	.859	.498	.468	.552	.272	.228	.522	.237	.388	.491	–	–	–	–	–
MAV	.791	.858	.810	.868	.820	.801	.797	.465	.784	.982	.520	.838	–	–	–	–
MES	.416	.858	.437	.447	.522	.173	.111	.509	.140	.427	.469	.181	.812	–	–	–
EPH	.357	.858	.401	.459	.503	.031	.067	.489	.165	.497	.450	.258	.800	.155	–	–
RSD	.665	.858	.675	.747	.722	.632	.637	.785	.652	.762	.767	.659	.834	.643	.633	–
Min	.065	.858	.187	.615	.436	.381	.415	.698	.481	.710	.664	.504	.783	.444	.392	.675
Max	1.120	.856	1.114	1.024	1.067	1.111	1.095	.972	1.073	.921	1.003	1.066	.939	1.087	1.104	.988

Table D.100: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Gaussian Cube 3 preferences.

Figure D.97: Axiom violation rate for each rule on Gaussian Cube 3 preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.021**	**0**	*.000*	.003	**.021**	**.016**	*.000*	*.000*	**0**	.081	.001	.001	.001	.061	.104
Borda	.027	*.000*	.002	.002	.027	.168	**0**	.005	**0**	.061	*.000*	*.000*	*.000*	.043	.069
EPH	.051	*.000*	.001	*.000*	.051	.349	.003	.001	**0**	.110	*.000*	*.000*	*.000*	.085	.117
SNTV	.085	**0**	.058	.245	.085	.731	.010	.033	**0**	.002	*.000*	.001	.005	**0**	.014
STV	.062	**0**	.032	.156	.062	.596	.002	.023	**0**	**0**	**0**	*.000*	.001	**0**	**.001**
Bloc	.050	*.000*	.001	**0**	.050	.324	.003	**0**	**0**	.105	*.000*	*.000*	*.000*	.082	.131
CC	.164	.003	.100	.328	.164	.828	.035	.047	**0**	.262	**0**	.005	.010	.220	.298
lex-CC	.079	.001	.004	**0**	.079	.578	.003	.012	**0**	.159	**0**	**0**	**0**	.128	.140
seq-CC	.152	.002	.093	.250	.152	.807	.026	.047	**0**	.253	**0**	.002	.002	.211	.282
Monroe	.151	.002	.077	.286	.151	.797	.032	.046	**0**	.246	**0**	.001	.003	.206	.263
Greedy M.	.080	*.000*	.014	.013	.080	.571	.003	.016	**0**	.155	**0**	**0**	**0**	.124	.150
PAV	.056	*.000*	.001	**0**	.056	.406	.003	.002	**0**	.119	**0**	**0**	**0**	.092	.112
MES	.065	*.000*	.001	.003	.065	.473	.002	.006	**0**	.134	**0**	**0**	**0**	.105	.119
MAV	.173	.004	.097	.313	.173	.876	.053	.048	**0**	.251	.016	.017	.017	.209	.341
RSD	.130	.002	.057	**0**	.130	.717	.020	.023	**0**	.200	.039	.041	.042	.165	.392
Random	.191	.004	.122	.363	.191	.890	.062	.047	**0**	.262	.029	.034	.041	.221	.406

Table D.101: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Gaussian Cube 10 preferences.

D.3.14 7 Alternatives, Gaussian Cube 10

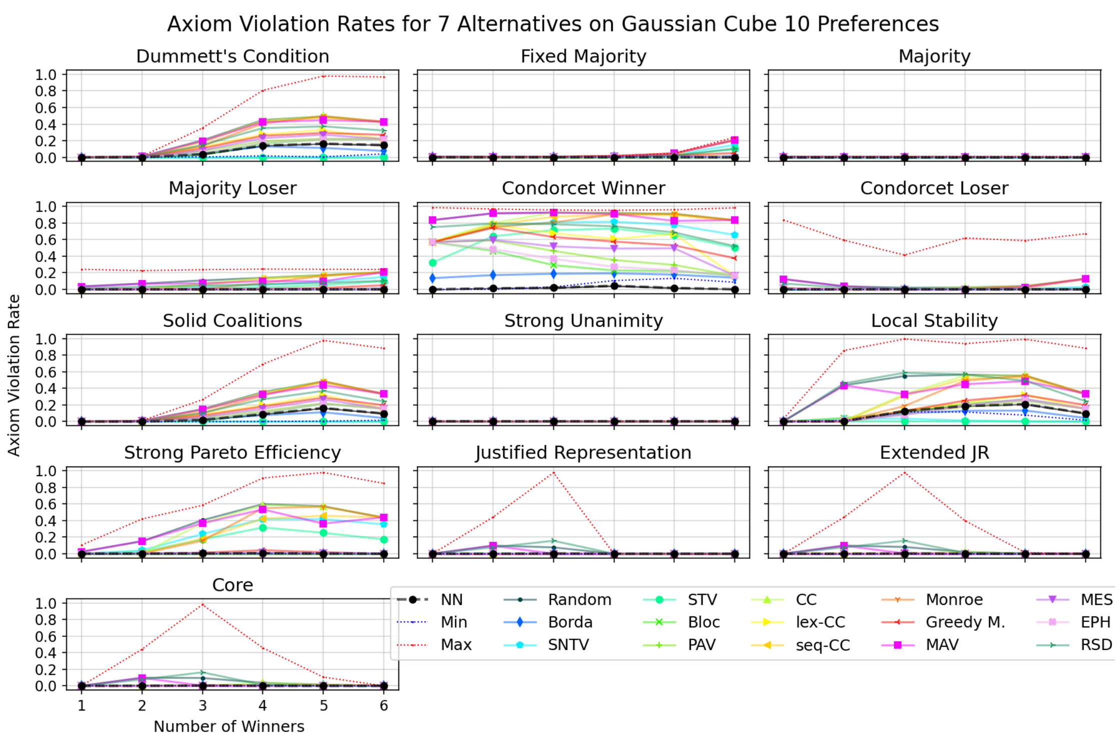


Figure D.98: Axiom violation rate for each axiom on Gaussian Cube 10 preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.856	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.168	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.623	.856	.639	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.452	.857	.459	.376	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.347	.857	.402	.451	.499	–	–	–	–	–	–	–	–	–	–	–
PAV	.382	.857	.410	.473	.509	.089	–	–	–	–	–	–	–	–	–	–
CC	.706	.857	.744	.588	.672	.495	.463	–	–	–	–	–	–	–	–	–
lex-CC	.455	.857	.478	.501	.545	.182	.106	.439	–	–	–	–	–	–	–	–
seq-CC	.698	.858	.708	.524	.662	.510	.477	.670	.463	–	–	–	–	–	–	–
Monroe	.669	.857	.706	.562	.641	.457	.422	.082	.411	.645	–	–	–	–	–	–
Greedy M.	.479	.857	.497	.466	.551	.274	.231	.523	.240	.390	.491	–	–	–	–	–
MAV	.793	.856	.812	.865	.822	.803	.798	.468	.785	.984	.523	.839	–	–	–	–
MES	.413	.858	.435	.445	.521	.175	.113	.509	.142	.426	.469	.181	.813	–	–	–
EPH	.357	.857	.400	.457	.501	.032	.069	.490	.165	.498	.450	.261	.803	.157	–	–
RSD	.664	.858	.676	.744	.720	.630	.634	.782	.650	.760	.764	.656	.834	.639	.631	–
Min	.063	.856	.186	.611	.437	.381	.414	.696	.480	.708	.661	.501	.785	.441	.391	.675
Max	1.120	.859	1.115	1.024	1.067	1.109	1.094	.971	1.071	.920	1.003	1.065	.939	1.085	1.103	.991

Table D.102: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Gaussian Cube 10 preferences.

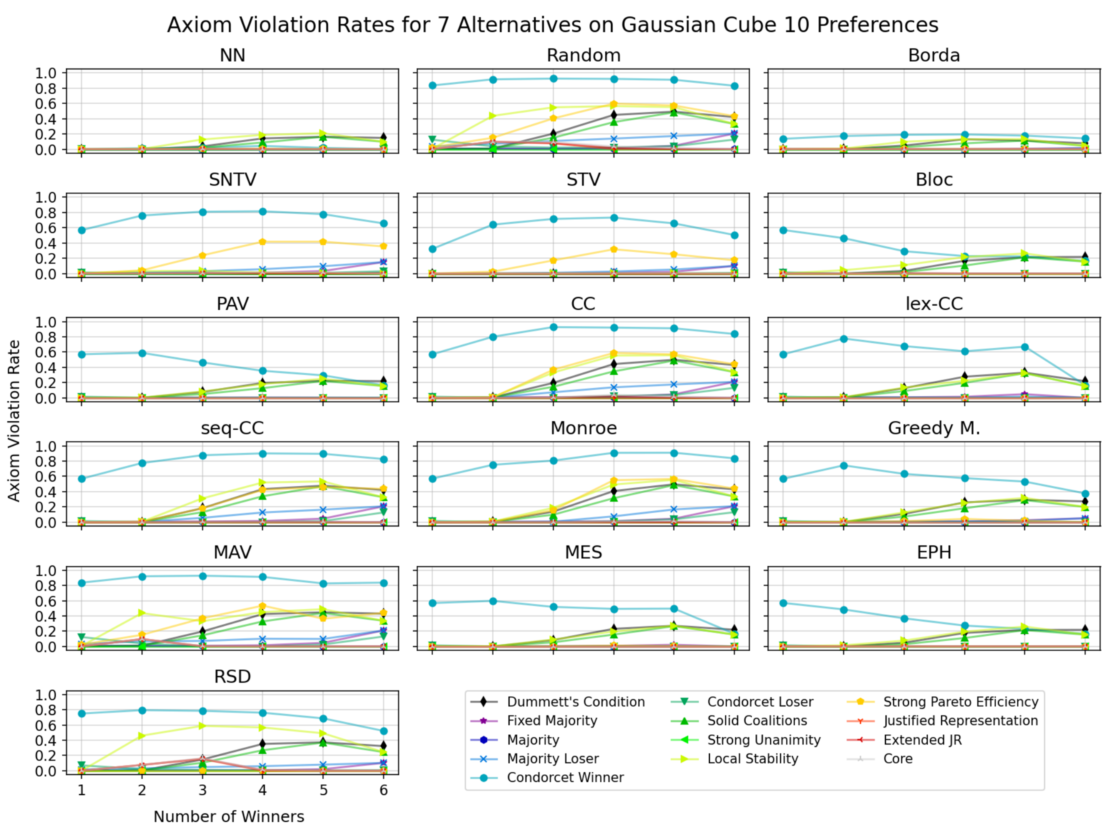


Figure D.99: Axiom violation rate for each rule on Gaussian Cube 10 preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.021**	*.000*	*.000*	.003	**.021**	**.013**	*.000*	*.000*	**0**	.084	.001	.001	.001	.064	.109
Borda	.028	*.000*	.002	.002	.028	.173	**0**	.005	**0**	.062	*.000*	*.000*	*.000*	.045	.071
EPH	.053	*.000*	.001	*.000*	.053	.357	.002	.001	**0**	.114	*.000*	*.000*	*.000*	.090	.123
SNTV	.085	**0**	.056	.248	.085	.736	.011	.032	**0**	.002	*.000*	.001	.005	**0**	.015
STV	.063	**0**	.031	.160	.063	.600	.002	.022	**0**	**0**	*.000*	*.000*	.001	**0**	**.001**
Bloc	.051	*.000*	.001	**0**	.051	.331	.002	**0**	**0**	.109	.001	.001	.001	.086	.138
CC	.163	.003	.092	.329	.163	.826	.033	.044	**0**	.259	**0**	.004	.009	.219	.297
lex-CC	.080	.001	.004	**0**	.080	.583	.002	.012	**0**	.162	**0**	**0**	**0**	.132	.145
seq-CC	.152	.002	.087	.247	.152	.808	.026	.044	**0**	.255	**0**	.002	.002	.215	.285
Monroe	.150	.002	.072	.287	.150	.797	.031	.043	**0**	.245	**0**	*.000*	.003	.206	.264
Greedy M.	.082	*.000*	.014	.014	.082	.577	.003	.016	**0**	.159	**0**	**0**	**0**	.129	.156
PAV	.058	*.000*	.001	**0**	.058	.414	.002	.002	**0**	.122	**0**	**0**	**0**	.096	.117
MES	.067	*.000*	.001	.003	.067	.484	.003	.006	**0**	.138	**0**	**0**	**0**	.110	.124
MAV	.171	.003	.090	.316	.171	.874	.053	.045	**0**	.249	.016	.017	.017	.208	.337
RSD	.131	.002	.053	**0**	.131	.719	.021	.022	**0**	.203	.038	.041	.042	.169	.397
Random	.190	.004	.114	.362	.190	.888	.061	.045	**0**	.262	.029	.034	.040	.221	.406

Table D.103: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Uniform Cube 3 preferences.

D.3.15 7 Alternatives, Uniform Cube 3

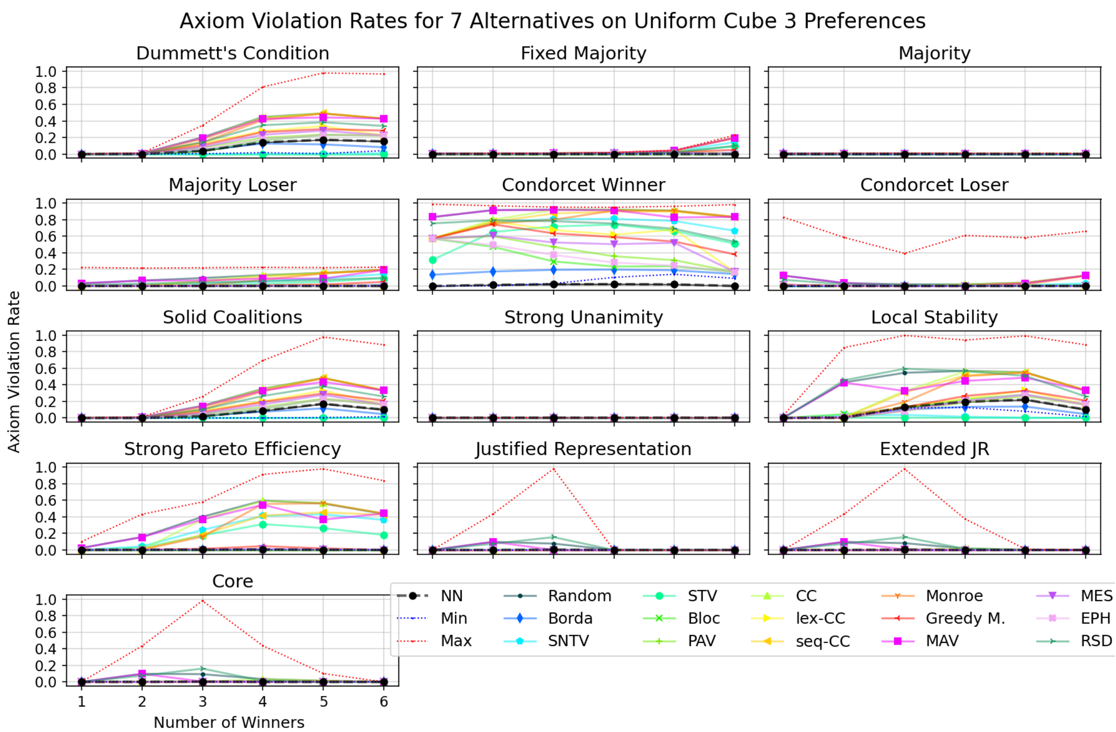


Figure D.100: Axiom violation rate for each axiom on Uniform Cube 3 preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.856	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.171	.856	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.632	.857	.646	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.455	.856	.460	.383	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.352	.857	.407	.458	.506	–	–	–	–	–	–	–	–	–	–	–
PAV	.387	.857	.414	.480	.517	.091	–	–	–	–	–	–	–	–	–	–
CC	.707	.857	.744	.590	.673	.495	.462	–	–	–	–	–	–	–	–	–
lex-CC	.459	.857	.480	.509	.552	.182	.104	.440	–	–	–	–	–	–	–	–
seq-CC	.702	.857	.710	.527	.669	.511	.477	.670	.464	–	–	–	–	–	–	–
Monroe	.672	.858	.708	.566	.645	.458	.423	.079	.412	.647	–	–	–	–	–	–
Greedy M.	.486	.857	.501	.472	.560	.276	.232	.523	.242	.388	.493	–	–	–	–	–
MAV	.793	.856	.813	.862	.823	.800	.795	.464	.784	.982	.518	.836	–	–	–	–
MES	.421	.857	.442	.452	.532	.178	.115	.509	.142	.426	.470	.182	.810	–	–	–
EPH	.362	.857	.405	.464	.508	.033	.070	.489	.165	.499	.451	.262	.800	.160	–	–
RSD	.670	.857	.681	.749	.725	.633	.638	.783	.653	.761	.766	.659	.832	.642	.635	–
Min	.065	.856	.191	.620	.440	.389	.421	.696	.486	.714	.663	.509	.783	.451	.398	.680
Max	1.118	.855	1.112	1.022	1.065	1.107	1.092	.974	1.070	.917	1.005	1.062	.940	1.083	1.101	.989

Table D.104: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Uniform Cube 3 preferences.

Figure D.101: Axiom violation rate for each rule on Uniform Cube 3 preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.021**	**0**	*.000*	.003	**.021**	**.013**	*.000*	**0**	**0**	.085	*.000*	*.000*	*.000*	.064	.109
Borda	.028	*.000*	.002	.002	.028	.174	**0**	.005	**0**	.064	*.000*	*.000*	*.000*	.046	.072
EPH	.053	*.000*	.001	*.000*	.053	.356	.003	.001	**0**	.116	*.000*	*.000*	*.000*	.090	.123
SNTV	.086	**0**	.057	.249	.086	.740	.012	.033	**0**	.003	*.000*	.002	.006	**0**	.016
STV	.063	**0**	.031	.160	.063	.605	.002	.023	**0**	**0**	**0**	*.000*	.001	**0**	**.001**
Bloc	.052	*.000*	.001	**0**	.052	.330	.003	**0**	**0**	.111	.001	.001	.001	.087	.138
CC	.163	.002	.093	.326	.163	.826	.033	.045	**0**	.261	**0**	.005	.010	.219	.298
lex-CC	.080	.001	.004	**0**	.080	.581	.003	.013	**0**	.163	**0**	**0**	**0**	.132	.145
seq-CC	.152	.002	.088	.249	.152	.811	.026	.045	**0**	.254	**0**	.001	.002	.213	.284
Monroe	.150	.002	.072	.285	.150	.797	.031	.044	**0**	.245	**0**	.001	.004	.205	.264
Greedy M.	.083	*.000*	.014	.013	.083	.580	.003	.016	**0**	.161	**0**	**0**	**0**	.129	.157
PAV	.058	*.000*	.001	**0**	.058	.413	.003	.002	**0**	.124	**0**	**0**	**0**	.097	.117
MES	.067	*.000*	.001	.003	.067	.482	.003	.006	**0**	.139	**0**	**0**	**0**	.110	.125
MAV	.171	.004	.090	.314	.171	.874	.051	.046	**0**	.251	.016	.018	.018	.209	.339
RSD	.131	.002	.054	**0**	.131	.718	.021	.023	**0**	.203	.039	.041	.042	.169	.396
Random	.190	.004	.114	.362	.190	.889	.061	.046	**0**	.265	.029	.034	.041	.223	.406

Table D.105: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Uniform Cube 10 preferences.

D.3.16 7 Alternatives, Uniform Cube 10

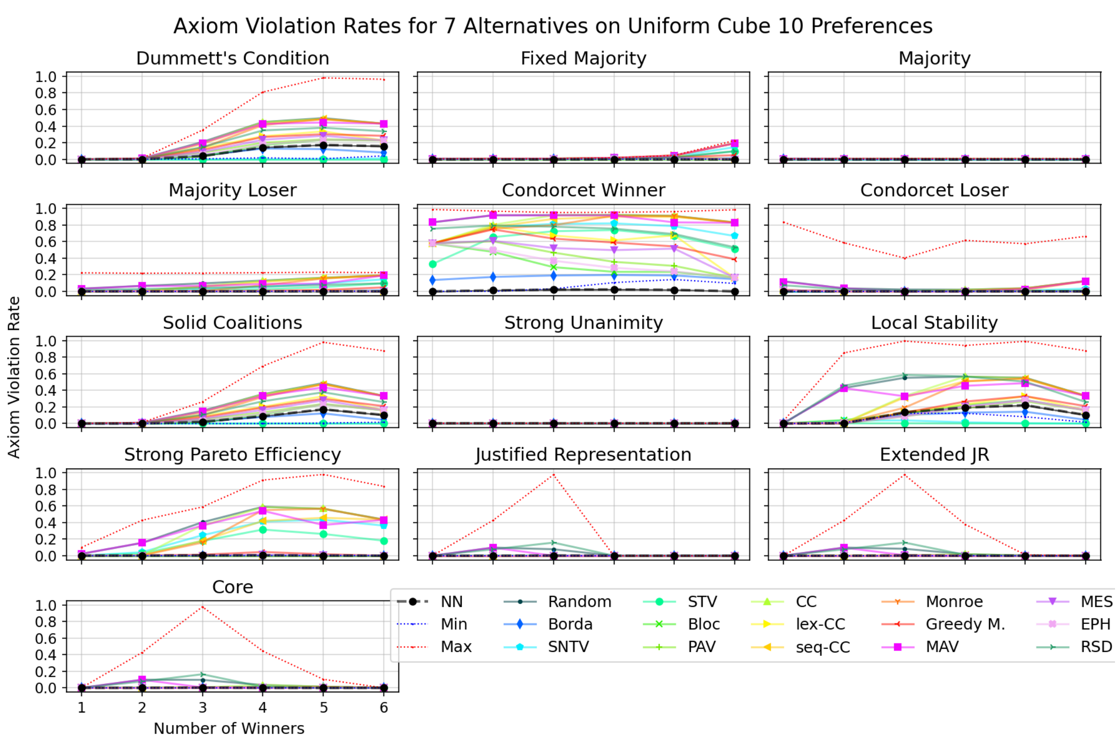


Figure D.102: Axiom violation rate for each axiom on Uniform Cube 10 preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.856	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.171	.857	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.634	.858	.648	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.460	.857	.463	.383	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.353	.858	.408	.456	.507	–	–	–	–	–	–	–	–	–	–	–
PAV	.388	.858	.416	.479	.517	.090	–	–	–	–	–	–	–	–	–	–
CC	.708	.858	.746	.593	.674	.494	.461	–	–	–	–	–	–	–	–	–
lex-CC	.460	.858	.482	.507	.552	.182	.105	.438	–	–	–	–	–	–	–	–
seq-CC	.704	.858	.712	.522	.667	.509	.475	.668	.463	–	–	–	–	–	–	–
Monroe	.673	.857	.710	.570	.646	.457	.422	.079	.411	.645	–	–	–	–	–	–
Greedy M.	.489	.858	.505	.473	.559	.275	.232	.520	.240	.389	.491	–	–	–	–	–
MAV	.792	.855	.812	.867	.822	.799	.795	.464	.783	.982	.518	.836	–	–	–	–
MES	.421	.858	.443	.452	.529	.176	.113	.506	.140	.426	.467	.182	.809	–	–	–
EPH	.363	.858	.406	.463	.508	.033	.069	.488	.165	.497	.450	.262	.799	.158	–	–
RSD	.669	.856	.680	.748	.725	.632	.637	.781	.652	.762	.764	.659	.832	.642	.634	–
Min	.066	.855	.191	.623	.445	.390	.423	.697	.487	.715	.664	.511	.783	.451	.400	.679
Max	1.118	.857	1.113	1.020	1.065	1.108	1.093	.974	1.071	.918	1.005	1.063	.942	1.084	1.101	.987

Table D.106: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Uniform Cube 10 preferences.

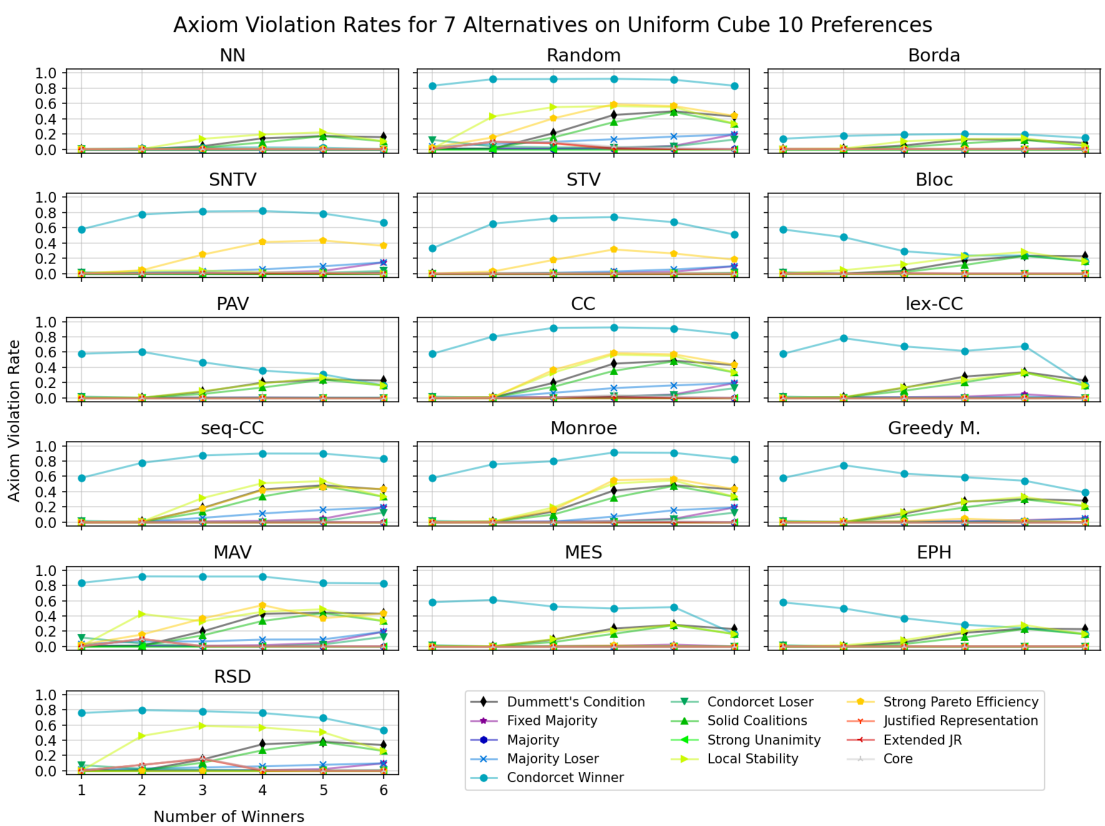


Figure D.103: Axiom violation rate for each rule on Uniform Cube 10 preferences with 7 alternatives.

Method	Mean	Maj W	Maj L	Pareto	Mean.1	Cond W	Cond L	F Maj	Unanimity	Dummett's	JR	EJR	Core	S. Coalitions	Stability
NN	**.018**	*.000*	**.001**	.007	**.018**	**.018**	*.000*	*.000*	*.000*	.063	.001	.001	.001	.047	.094
Borda	.022	.001	.004	.004	.022	.127	**0**	.011	**0**	.044	*.000*	*.000*	*.000*	.031	.056
EPH	.040	*.000*	.001	*.000*	.040	.272	.002	.001	**0**	.082	*.000*	*.000*	*.000*	.063	.097
SNTV	.099	**0**	.097	.227	.099	.620	.007	.106	.049	.062	*.000*	.054	.057	**0**	.012
STV	.048	**0**	.036	.117	.048	.443	.002	.029	**0**	**0**	**0**	*.000*	.001	**0**	**.001**
Bloc	.039	*.000*	.001	**0**	.039	.256	.002	**0**	**0**	.080	*.000*	*.000*	*.000*	.061	.106
CC	.195	.036	.145	.346	.195	.758	.032	.140	.061	.308	**0**	.084	.091	.232	.301
lex-CC	.062	.005	.007	**0**	.062	.443	.002	.024	**0**	.117	**0**	**0**	**0**	.091	.111
seq-CC	.182	.032	.138	.296	.182	.742	.024	.140	.061	.290	**0**	.078	.081	.214	.276
Monroe	.130	.006	.078	.234	.130	.650	.026	.060	**0**	.214	**0**	.002	.006	.179	.231
Greedy M.	.063	.002	.018	.012	.063	.450	.003	.023	**0**	.111	**0**	**0**	**0**	.088	.117
PAV	.044	.001	.001	**0**	.044	.311	.002	.004	**0**	.088	**0**	**0**	**0**	.068	.091
MES	.049	.001	.002	.001	.049	.353	.002	.008	**0**	.096	**0**	**0**	**0**	.075	.094
MAV	.157	.022	.108	.279	.157	.751	.044	.084	**0**	.219	.015	.023	.023	.179	.299
RSD	.105	.008	.055	**0**	.105	.595	.016	.035	**0**	.147	.029	.031	.032	.119	.299
Random	.237	.062	.170	.405	.237	.846	.058	.160	.071	.326	.049	.125	.134	.252	.419

Table D.107: Average Axiom Violation Rate for 7 alternatives and $1 \leq k < 7$ winners across Mixed preferences.

D.3.17 7 Alternatives, Mixed

Figure D.104: Axiom violation rate for each axiom on Mixed preferences with 7 alternatives.

	NN	Random	Borda	SNTV	STV	Bloc	PAV	CC	lex-CC	seq-CC	Monroe	Greedy M.	MAV	MES	EPH	RSD
Random	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Borda	.178	.858	–	–	–	–	–	–	–	–	–	–	–	–	–	–
SNTV	.566	.858	.549	–	–	–	–	–	–	–	–	–	–	–	–	–
STV	.386	.857	.376	.364	–	–	–	–	–	–	–	–	–	–	–	–
Bloc	.314	.859	.335	.433	.423	–	–	–	–	–	–	–	–	–	–	–
PAV	.339	.858	.342	.445	.428	.065	–	–	–	–	–	–	–	–	–	–
CC	.648	.858	.687	.606	.636	.488	.465	–	–	–	–	–	–	–	–	–
lex-CC	.397	.857	.397	.468	.458	.144	.089	.440	–	–	–	–	–	–	–	–
seq-CC	.676	.858	.655	.462	.630	.509	.485	.690	.479	–	–	–	–	–	–	–
Monroe	.563	.857	.600	.549	.555	.400	.376	.117	.365	.635	–	–	–	–	–	–
Greedy M.	.431	.858	.420	.427	.468	.249	.220	.530	.235	.402	.448	–	–	–	–	–
MAV	.685	.858	.744	.829	.746	.728	.727	.471	.716	.948	.474	.770	–	–	–	–
MES	.366	.857	.357	.411	.437	.147	.103	.514	.135	.432	.426	.169	.747	–	–	–
EPH	.321	.858	.334	.436	.424	.021	.052	.484	.133	.501	.396	.240	.729	.135	–	–
RSD	.600	.858	.602	.691	.636	.569	.572	.752	.586	.733	.683	.589	.760	.574	.570	–
Min	.054	.858	.187	.557	.375	.334	.359	.646	.412	.679	.561	.442	.687	.381	.341	.604
Max	1.123	.856	1.106	1.027	1.078	1.103	1.093	.975	1.075	.911	1.028	1.066	.982	1.084	1.099	1.010

Table D.108: Difference between rules for 7 alternatives with $1 \leq k < 7$ on Mixed preferences.

Figure D.105: Axiom violation rate for each rule on Mixed preferences with 7 alternatives.