Distortion in Real World Settings

Kin Hei Wong

Master of Science

Computer Science

School of Informatics

University of Edinburgh

2023

Abstract

Prior studies on Distortion have focused on theoretical results. In this study, we look at 11 datasets based on real-world datasets to find the distortions of voting mechanisms. We then look at the distribution of scores, the distribution of ranks, and run experiments to see the behavior of high-alternative datasets and the spoiler effect. We find that distortions are all around low and plurality is the clear winner in most cases. This is due to how the score and rank distributions allow high-rank weighting mechanisms to often pick the right choice. Through experimenting with varying voters against alternatives and the spoiler effect, we also find that bad-case distortions are mainly in contrived settings that can be easily mitigated. While prior theoretical work illustrates the limitations of plurality against other voting mechanisms, our results thus show that in real-world situations plurality is very effective.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Kin Hei Wong*)

Acknowledgements

Acknowledgements to the supervisor, Aris Filos-Ratsikas for providing guidance and help throughout this study.

Table of Contents

Chapter 1

Introduction

In a standard election, voters usually express their preferences as ordinal rankings of their preferred candidates. Much of the work in Social Choice Theory in theorems such as Arrow's Impossibility Paradox [3] [4] show that it is impossible to build voting systems that satisfy all the preferable axioms we might want out of a voting system. If voters expressed their preferences as a profile of cardinal utilities, one could easily compute the utility-maximizing candidate. However, for most real elections we see that such preferences are not expressed or may not be even known by the voters themselves.

Distortion measures the ratio of welfare or utility between a given alternative and the best alternative. As such it provides a way to quantitatively evaluate and compare different voting mechanisms. Much of the previous literature has been focused on the theoretical analysis with distortion on voting mechanisms, such as finding the worst-case bounds for voting mechanisms [2]. However, there have yet to be studies that have looked at how these voting mechanisms behave in the context of real-world data.

In this dissertation, we perform an exploratory study of distortion on real-world preferences. We will look at how voting mechanisms behave with different assumptions of the distributions of cardinal scores, different datasets, synthetic datasets, and more.

Chapter 2

Background

2.0.1 Distortion

Distortion measures the worse-case ratio of utility between the best alternative and a given alternative. The formal setting is such that we have alternatives *j* of set *A* of length *m*, and voters *i* of set *N* of length *n*, and each voter has a strict preference ranking $\prec_i$ that contains all alternatives from *A*. The utility function *SW* maps the welfare of each alternative *j* from *A* to every voter *i* in *N*, such that the utilities for each alternative for each voter sums to 1 and adheres to the rankings. If we consider a social choice function *f*, then the distortion is formally defined as the worst-case ratio between the utility-maximizing best alternative and the utility of the alternative picked by the social choice function [5] [2].

$$\text{distortion}(f) = \sup_{(N,A,\mathbf{v})} \frac{\max_{j \in A} \text{SW}(j|\mathbf{v})}{\text{SW}(f(\succ_{\mathbf{v}})|\mathbf{v})}.$$

Figure 2.1: Anshelevich, Elliot, et Al. *“Distortion in Social Choice Problems: The First 15 Years and Beyond.”.* International Joint Conference on Artificial Intelligence (2021). Distortion Equation [1][2]

2.0.2 Voting Mechanisms

7 voting mechanisms are used throughout this study; Plurality, Borda, Veto, Harmonic, Combined, Copeland, and Uniform. While there are many more and often more complex voting rules out there, we decided to focus on basic rules as there already exists

much literature on their theoretical behavior and they are easy to implement and study the behavior of. Voting mechanisms are divided into deterministic and randomized versions, whereby deterministic picks the alternative with the best score assigned, while randomized picks at random with a probability distribution defined by the scores given by each alternative. This idea of using scores in different ways was introduced by Caragiannis et Al. [2] as utility *embeddings*, and for our purposes can provide a more informative way of comparing the performance of mechanisms. We use asymptotic notation to describe the worst-case distortion bounds to compare voting rules.

2.0.2.1 Plurality

Plurality assigns a score of 1 to the top ranked alternative for each preference profile. The upper bound of distortion of plurality is $O(m^2)$, which is also the best achievable for all deterministic voting mechanisms [2]. Randomized plurality has a tight bound of $\Theta(m\sqrt{m})$ [7].

2.0.2.2 Borda

Borda assigns a score of m-1 to the top ranked alternative, m-2 to the next best, and so on until the lowest ranked alternative has a score of 0. The upper bound of distortion for deterministic Borda is unbounded. The tight bound for randomized borda is $\Theta(m^{5/4})$ [7].

2.0.2.3 Uniform

Uniform is random voting. It assigns a score of 1 to every alternative. Randomized uniform has a tight bound of $\Theta(m)$ [7].

2.0.2.4 Veto

Veto assigns a score of 1 to every alternative except the lowest ranking alternative for each preference profile. The upper bound of distortion for deterministic Veto is unbounded. The tight bound for randomized veto is $\Theta(m)$ [7].

2.0.2.5 Harmonic

Harmonic assigns a score of 1 to the top ranked alternative, 1/2 to the next best, and so on until $1/m$ for the lowest ranked alternative. The upper bound for randomized harmonic is $O(\sqrt{m}\log(m))$ [3].

2.0.2.6 Combined

Combined is a randomized voting rule introduced by Boutiller [3], that picks at 50% from uniform, and 50% from randomized harmonic whereby harmonic scores are probabilities at which each alternative is picked. The upper bound for Combined is $O(\sqrt{m\log(m)})$ [3].

2.0.2.7 Copeland Winner

The Condercet Winner is the alternative that has the pairwise majority to all other alternatives. That is to say, in any one-off election against the other alternatives, the Condercet Winner will win and thus is the majority winner. As such, this makes it one of the preferable property in constituting a ”best” alternative when not using distortion. Copeland’s Method assigns each alternative a score of 1 for number of alternatives it has a pairwise majority to, and 0.5 for each alternative is has a pairwise tie to. If there is a Condercet Winner, then Copeland’s Method will pick the Condercet Winner. Some preference profiles do not necessarily have a Condorcet Winner.

Chapter 3

Methods

This study was primarily conducted in Python with a Jupyter notebook. Libraries such as numpy, pandas, matplotlib & seaborn were used for the experiments and for producing necessary graphs or visualizations. All code can be viewed through the GitHub link provided in the appendix.

This study is split into two parts, the primary analysis and the case studies. Our primary analysis will look into the general performance of mechanisms from all datasets. We then perform a case study based on the observations of the primary analysis for specific scenarios we found interesting.

3.0.1 Datasets and Preprocessing

All datasets were sourced from Preflib.org [9]. The descriptions of each dataset were sourced also from Preflib and can be found in the Appendix.

Preflib provides its own libraries to parse the dataset files. Preflib datasets come in 4 data formats, Strict Complete Orders (SOC), Strict Incomplete Orders (SOI), Orders with Ties - Complete (TOC), Orders with Ties – Incomplete. Strict Orders have no ties between alternatives and vice versa for Ties. Complete Orders include all alternatives for each preference profile, while Incomplete Orders may be missing some for each profile.

For tied orders, we simply flattened the tied alternatives with uniform distribution within their preference rank. However, for generating cardinal scores, we assigned the same weight for these tied alternatives in the same “rank” pre-flattening. We believe this method should more accurately reflect the utility scores rather than blindly applying the weight functions continuously. For incomplete orders, we added the missing alternatives below the least ranked alternatives in a uniform distribution, applying cardinal scores to them in the same manner as tied orders.

Datasets often come in multiple files corresponding to different instances of the event, for example, elections across different years, or across different counties. The distortions for each instance correspond to each row for the table of distortions generated for the dataset.

3.0.2 Cardinal Score Generation

As Preflib provides only preference rankings, we needed to generate cardinal scores. The Dirichlet Distribution accepts a vector of weights of length n and outputs a corresponding vector of length n that are unit-sum normalized to 1. While the direct usage of the Dirichlet distribution is to sample a distribution of unknown probabilities from a vector of events, we appropriated the distribution as a straightforward way to generate scores that fitted within our requirements. The ratio of a weight element to another weight corresponds to the size of it is output score relative to other scores, while the magnitude of the weight reduces the variance.

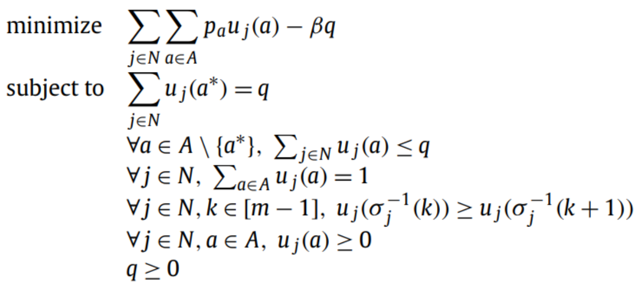
To match with the ordinal rankings, we implemented Linear, Logarithmic, and Exponential weight functions that corresponded to the distance between each score in a preference. For logarithmic scores, most alternatives received only slightly less scores than the previous rank, for exponential, they received much less, etc.

Since the scores are picked from a distribution with some variance, there may be situations in which the scores for some preferences may be larger than the previous rank for logarithmic and linear functions. However, as we are often dealing with many voters and the occurrences and impact of these “breaks” are small, on aggregate the total utility scores for each alternative do reflect the ordinal rankings.

Obviously, it is a naïve assumption that the real utility distribution of scores may be similar to these distributions. Not all voters be using the same distribution overall and there may be different groups using different distributions. Using distributions to generate scores can also be voting mechanism in their right, as linear weights would be very similar to Borda scores, but the introduction of randomization within the ties and incomplete orders may still produce different distributions of scores anyways. As deciding how to assign these distributions and which would best reflect reality would be an entire paper in itself, for the scope of this dissertation we decided to stick with mostly exponential weights as we found that they provided the worst and the most interesting distortions.

3.0.3 Linear Programming

Due to limitations with using distributions to model cardinal utilities, we also employed linear programming to find the worst-case distortions. Boutiller et Al. describe such a set of constraints for randomized voting functions [3]. The scores of the voting function p are used to generate the weights for the expected utility. Because the formula for distortion is not linear, we use a beta value β to approximate the distortion possible under the objective function. For brevity we will not explain all the constraints save for that they formalize the definition of distortion earlier into LP constraints.



$$
\begin{aligned}
\text{minimize} \quad & \sum_{j \in N} \sum_{a \in A} p_a u_j(a) - \beta q \\
\text{subject to} \quad & \sum_{j \in N} u_j(a^*) = q \\
& \forall a \in A \setminus \{a^*\},\ \textstyle\sum_{j \in N} u_j(a) \leq q \\
& \forall j \in N,\ \textstyle\sum_{a \in A} u_j(a) = 1 \\
& \forall j \in N, k \in [m-1],\ u_j(\sigma_j^{-1}(k)) \geq u_j(\sigma_j^{-1}(k+1)) \\
& \forall j \in N, a \in A,\ u_j(a) \geq 0 \\
& q \geq 0
\end{aligned}
$$

Figure 3.1: Boutiller et Al. *"Optimal social choice functions: a utilitarian view."* Artif. Intell. 227 (2012): 190-213: Lemma 3.5. (LP Constraints)

Whereby the LP tries to minimize the expected utility of the randomized mechanism minus the utility of the best alternative q scaled by the constant β. As such, if the objective value is more than or equal to zero, we know that a distortion of at least $\frac{1}{\beta}$ is possible. If it is less than zero, then such a distortion is not possible, since the expected utility must always be less than the utility of the best alternative. The deterministic version of the LP replaces the expected utility $\sum_{j \in N} \sum_{a \in A} p_a u_j(a)$ with $\sum_{j \in N} u_j(a^-)$ whereby a^- is the winner picked by the social choice function.

Since the beta value is an external constant and we do not know the best alternative beforehand, for each alternative, we used SciPy's *fsolve* optimization function to find which value of β returned in an objective value at or close to 0, thus giving us a close approximation of the worse-case distortion, then picked the worst distortions of the alternatives. This process was very computationally intensive, scaling badly with many alternatives, and thus we limited its usage to interesting datasets.

3.0.4 Distortion Pipeline

We found that computational efficiency was a major factor to be considered throughout the experiments. The massive size of some of the data means that the running times for some of our code, especially Linear Programming could take hours. As such, we did not run experiments on all possibilities and focused on interesting cases.

For each file in a dataset, the orders and scores are parsed and then generated. The best alternative is then found. Then a specified number of voting rules is run to obtain both the picked winner and the scores. The distortions are then calculated for each voting rule. For computational efficiency, randomized rules are not picked randomly, and instead, the normalized scores are used to calculate the expected utility that is then used in the distortion calculation. In order to account for the randomness of the distributions, especially with non-complete datasets, we reran this procedure ten times and averaged the results.

Due to computational limitations with the Linear Programming, we only ran randomized Plurality, Veto, Borda, and Harmonic along with deterministic Plurality, Harmonic, and Combined on each file. Our inclusion of only these three for deterministic is due to our later distribution findings with plurality, and we wanted if there were variations in the performance for the best performing rules that matched theoretical bounds. As the distortions for Borda and Veto are unbounded, they were omitted as they would be predictably high in the worse-case. For the sake of computational efficiency, we thus could check over those three much faster.

3.0.5 Relative Standard Distribution

Measuring the dispersion of utility from the best alternatives to other alternatives in a preference profile is used frequently in our analysis. As the scales of scores between data sets are different, we used Relative Standard Deviation to compare results.

$$RSD = \frac{\sigma}{\mu}$$

RSD is defined as the standard deviation σ divided by the mean μ. Because we want to measure the spread from the best alternative, we do not use the actual mean and instead treat the utility of the best alternative as the mean.

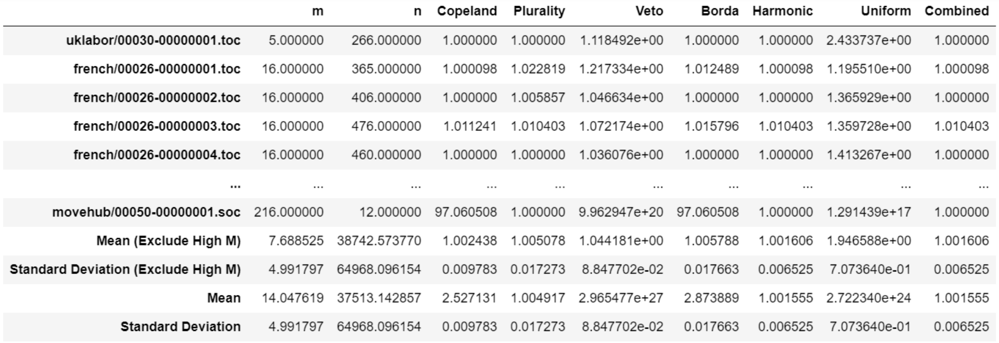
3.0.6 Synthetic Datasets

For our case studies, we often use Mallow's Model Mix sampling to create synthetic datasets. A Mallow's Model Mix accepts several reference orders or elements, each with a dispersion parameter from 0-1. Orders are then sampled from references according to this dispersion, whereby a low dispersion results in many orders similar or the same as the reference order, and a high dispersion of 1 being uniformly random orders [6]. Preflib provides an implementation of Mallow's Mix Sampler that we use in this study.

Chapter 4

Results

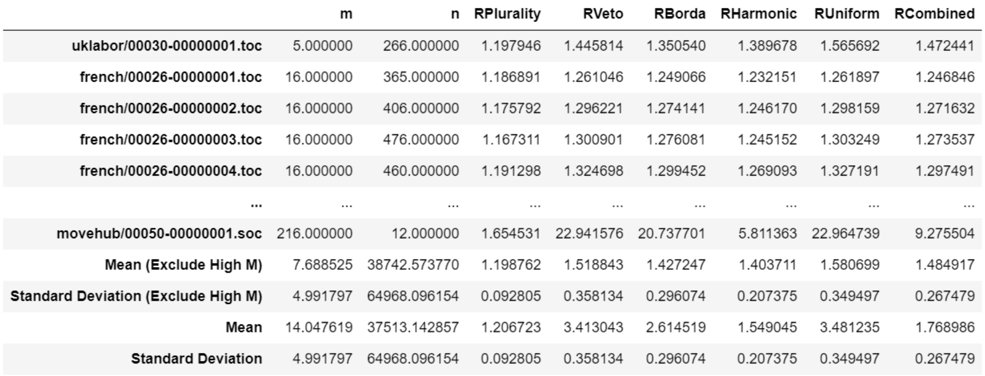
11 datasets with a total of 66 preference profiles were analyzed. The full results for distribution experiments and LP experiments can be viewed in Normal_results.csv and LP_results.csv respectively. Figure 4.1 and Figure 4.2 shows a preview of the results. Due to the large outliers for the two high *m* datasets of ”movehub” and ”university”, we excluded them to better show the variance of the majority of datasets.



	m	n	Copeland	Plurality	Veto	Borda	Harmonic	Uniform	Combined
uklabor/00030-00000001.toc	5.000000	266.000000	1.000000	1.000000	1.118492e+00	1.000000	1.000000	2.433737e+00	1.000000
french/00026-00000001.toc	16.000000	365.000000	1.000098	1.022819	1.217334e+00	1.012489	1.000098	1.195510e+00	1.000098
french/00026-00000002.toc	16.000000	406.000000	1.000000	1.005857	1.046634e+00	1.000000	1.000000	1.365929e+00	1.000000
french/00026-00000003.toc	16.000000	476.000000	1.011241	1.010403	1.072174e+00	1.015796	1.010403	1.359728e+00	1.010403
french/00026-00000004.toc	16.000000	460.000000	1.000000	1.000000	1.036076e+00	1.000000	1.000000	1.413267e+00	1.000000
...	...	...	...	...	...	...	...	...	...
movehub/00050-00000001.soc	216.000000	12.000000	97.060508	1.000000	9.962947e+20	97.060508	1.000000	1.291439e+17	1.000000
Mean (Exclude High M)	7.688525	38742.573770	1.002438	1.005078	1.044181e+00	1.005788	1.001606	1.946588e+00	1.001606
Standard Deviation (Exclude High M)	4.991797	64968.096154	0.009783	0.017273	8.847702e-02	0.017663	0.006525	7.073640e-01	0.006525
Mean	14.047619	37513.142857	2.527131	1.004917	2.965477e+27	2.873889	1.001555	2.722340e+24	1.001555
Standard Deviation	4.991797	64968.096154	0.009783	0.017273	8.847702e-02	0.017663	0.006525	7.073640e-01	0.006525

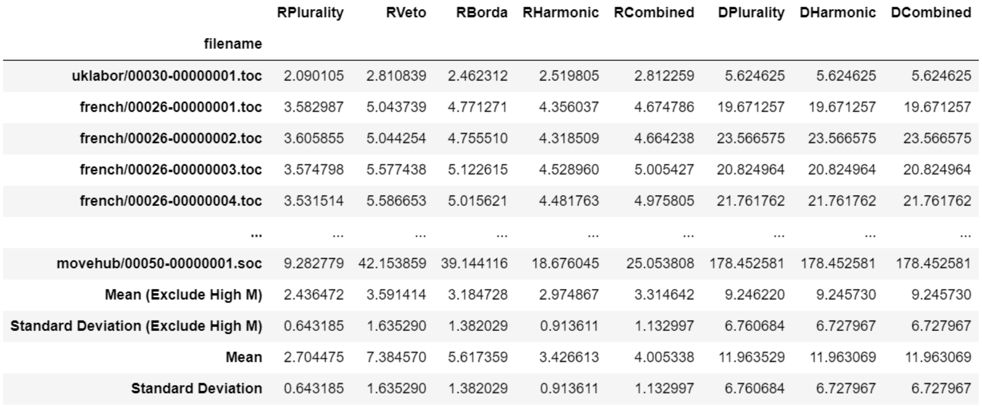
Figure 4.1: Full Deterministic Results (Preview)

Distortions are all around very low with low variance for most of the datasets with low *m*. The low variance is surprising as despite the large amount of variance for m and n, the distortions themselves remain fairly homogeneous across all the datasets. Deterministic performance appears to show that most datasets managed to pick the best or close to the best alternative for most datasets, and randomized also displays generally quite low distortions. Plurality clearly has the best performance of the voting mechanisms, with its low scores and very low variance, while Harmonic and Combined are better than Borda and Veto when looking at the full mean. This matches the theoretical findings of the bounds of these mechanisms. Copeland also performs very well for most datasets with corresponding Condorcet winners but fails disastrously in



	m	n	RPlurality	RVeto	RBorda	RHarmonic	RUniform	RCombined
uklabor/00030-00000001.toc	5.000000	266.000000	1.197946	1.445814	1.350540	1.389678	1.565692	1.472441
french/00026-00000001.toc	16.000000	365.000000	1.186891	1.261046	1.249066	1.232151	1.261897	1.246846
french/00026-00000002.toc	16.000000	406.000000	1.175792	1.296221	1.274141	1.246170	1.298159	1.271632
french/00026-00000003.toc	16.000000	476.000000	1.167311	1.300901	1.276081	1.245152	1.303249	1.273537
french/00026-00000004.toc	16.000000	460.000000	1.191298	1.324698	1.299452	1.269093	1.327191	1.297491
...	...	...	...	...	...	...	...	...
movehub/00050-00000001.soc	216.000000	12.000000	1.654531	22.941576	20.737701	5.811363	22.964739	9.275504
Mean (Exclude High M)	7.688525	38742.573770	1.198762	1.518843	1.427247	1.403711	1.580699	1.484917
Standard Deviation (Exclude High M)	4.991797	64968.096154	0.092805	0.358134	0.296074	0.207375	0.349497	0.267479
Mean	14.047619	37513.142857	1.206723	3.413043	2.614519	1.549045	3.481235	1.768986
Standard Deviation	4.991797	64968.096154	0.092805	0.358134	0.296074	0.207375	0.349497	0.267479

Figure 4.2: Full Randomized Results (Preview)

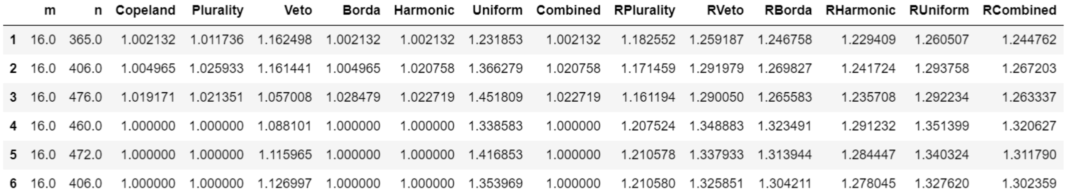


	RPlurality	RVeto	RBorda	RHarmonic	RCombined	DPlurality	DHarmonic	DCombined
filename								
uklabor/00030-00000001.toc	2.090105	2.810839	2.462312	2.519805	2.812259	5.624625	5.624625	5.624625
french/00026-00000001.toc	3.582987	5.043739	4.771271	4.356037	4.674786	19.671257	19.671257	19.671257
french/00026-00000002.toc	3.605855	5.044254	4.755510	4.318509	4.664238	23.566575	23.566575	23.566575
french/00026-00000003.toc	3.574798	5.577438	5.122615	4.528960	5.005427	20.824964	20.824964	20.824964
french/00026-00000004.toc	3.531514	5.586653	5.015621	4.481763	4.975805	21.761762	21.761762	21.761762
...	...	...	...	...	...	...	...	...
movehub/00050-00000001.soc	9.282779	42.153859	39.144116	18.676045	25.053808	178.452581	178.452581	178.452581
Mean (Exclude High M)	2.436472	3.591414	3.184728	2.974867	3.314642	9.246220	9.245730	9.245730
Standard Deviation (Exclude High M)	0.643185	1.635290	1.382029	0.913611	1.132997	6.760684	6.727967	6.727967
Mean	2.704475	7.384570	5.617359	3.426613	4.005338	11.963529	11.963069	11.963069
Standard Deviation	0.643185	1.635290	1.382029	0.913611	1.132997	6.760684	6.727967	6.727967

Figure 4.3: Full LP Results (Preview)

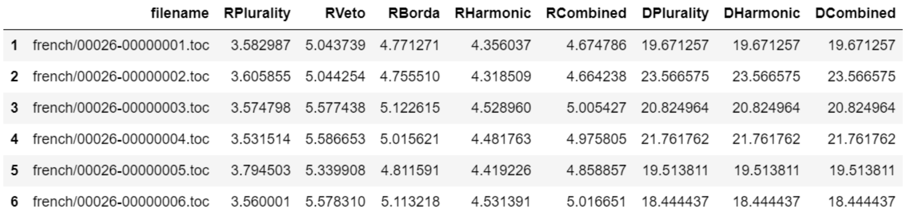
movehub whereby a Condorcet winner does not exist.

All voting mechanisms have better performance than uniform voting which is good since then our voting mechanisms are worth using in reality. In Figure 4.3, the LP worst-case results show a similar pattern, in which randomized plurality leads with harmonic, then combined, then borda and veto trailing. As expected, the LP has worse distortions for every case and a higher standard deviation for every rule. What is also interesting is the deterministic LP results all have the same distortion, implying that each voting rule picked the same or similar alternatives.



	m	n	Copeland	Plurality	Veto	Borda	Harmonic	Uniform	Combined	RPlurality	RVeto	RBorda	RHarmonic	RUniform	RCombined
1	16.0	365.0	1.002132	1.011736	1.162498	1.002132	1.002132	1.231853	1.002132	1.182552	1.259187	1.246758	1.229409	1.260507	1.244762
2	16.0	406.0	1.004965	1.025933	1.161441	1.004965	1.020758	1.366279	1.020758	1.171459	1.291979	1.269827	1.241724	1.293758	1.267203
3	16.0	476.0	1.019171	1.021351	1.057008	1.028479	1.022719	1.451809	1.022719	1.161194	1.290050	1.265583	1.235708	1.292234	1.263337
4	16.0	460.0	1.000000	1.000000	1.088101	1.000000	1.000000	1.338583	1.000000	1.207524	1.348883	1.323491	1.291232	1.351399	1.320627
5	16.0	472.0	1.000000	1.000000	1.115965	1.000000	1.000000	1.416853	1.000000	1.210578	1.337933	1.313944	1.284447	1.340324	1.311790
6	16.0	406.0	1.000000	1.000000	1.126997	1.000000	1.000000	1.353969	1.000000	1.210580	1.325851	1.304211	1.278045	1.327620	1.302359

Figure 4.4: French Results for Distributions



	filename	RPlurality	RVeto	RBorda	RHarmonic	RCombined	DPlurality	DHarmonic	DCombined
1	french/00026-00000001.toc	3.582987	5.043739	4.771271	4.356037	4.674786	19.671257	19.671257	19.671257
2	french/00026-00000002.toc	3.605855	5.044254	4.755510	4.318509	4.664238	23.566575	23.566575	23.566575
3	french/00026-00000003.toc	3.574798	5.577438	5.122615	4.528960	5.005427	20.824964	20.824964	20.824964
4	french/00026-00000004.toc	3.531514	5.586653	5.015621	4.481763	4.975805	21.761762	21.761762	21.761762
5	french/00026-00000005.toc	3.794503	5.339908	4.811591	4.419226	4.858857	19.513811	19.513811	19.513811
6	french/00026-00000006.toc	3.560001	5.578310	5.113218	4.531391	5.016651	18.444437	18.444437	18.444437

Figure 4.5: French Results for LP

Figure 4.4 displays the distribution distortions for the French dataset as a ”representative” dataset. Each index corresponds to an instance of the dataset for brevity, and distortions across instances are quite similar throughout with plurality taking the lead. The LP results in 4.5 have higher distortions as expected, but are similar to the distribution results in having very homogeneous distortions.

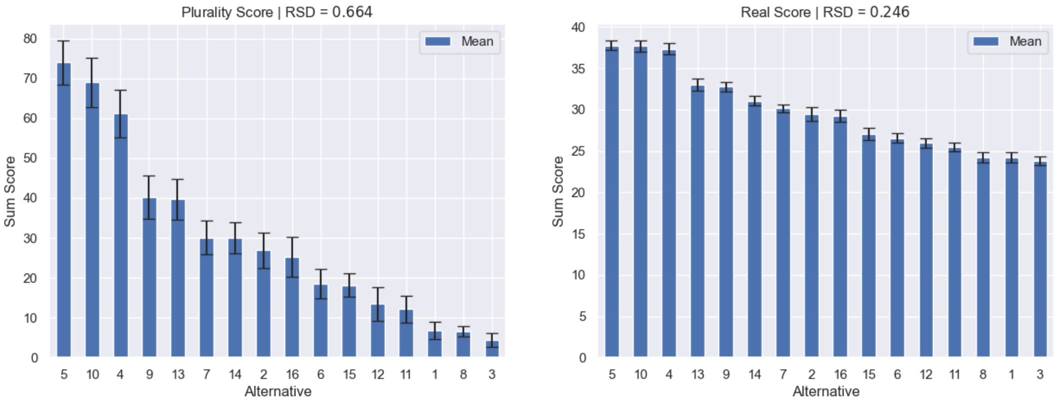


Figure 4.6: French Plurality Scores vs Real Utility Scores for Index 3

The total score of each alternative provides more context for these distortions in Figure 4.6. Due to ranking restrictions, the winners picked by Plurality will always have high utilities that are within the best performing alternatives. This means that in the situation where the Plurality winner is not the best alternative, the distance between its utility and the utility of the best alternative is small enough that the resulting ratio provides a very small distortion. Such events are quite rare, as looking at the plurality scores, much of the scoring weight that is ascribed to the next best alternative is from being the top ranked alternative for many profiles. Rather it is a very small portion of near “ties” that likely decide which alternative is the real winner.

As some of our datasets are not strict complete orderings, the use of randomization to decide ties produces non-deterministic results. Due to how we assign scores based on the initial ties, randomized results are generally the same, but for deterministic results, it will affect the voting mechanisms. Figure 4.5 shows Plurality picking the right winner, but its average distortion is not 1. The reason for this is due to the large variance between the scores of the top ranked alternatives 5, 10, and 4 that plurality is ”alternating” across these choices. Considering the significant difference in error bars between the real scores and plurality, their differences illustrate the loss of information when moving from strict ties to strict-complete orders. That being said, the smaller error bars with Borda and Harmonic in Figure 4.7 may show how less ”extreme” voting rules that assign a score to every alternative are more resilient to such random processes. Nonetheless, their error bars are still large enough to cause that same ”alternating” phenomenon as we see in Borda.

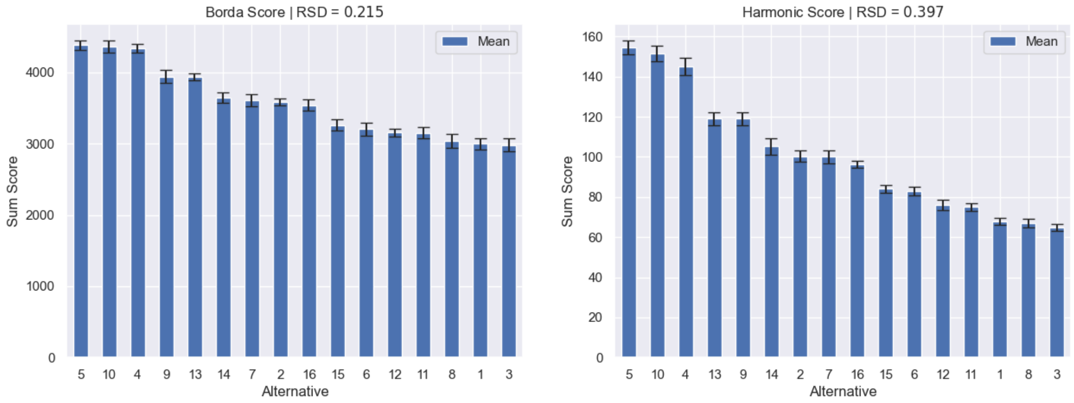


Figure 4.7: French Borda Scores vs Harmonic Scores for Index 3

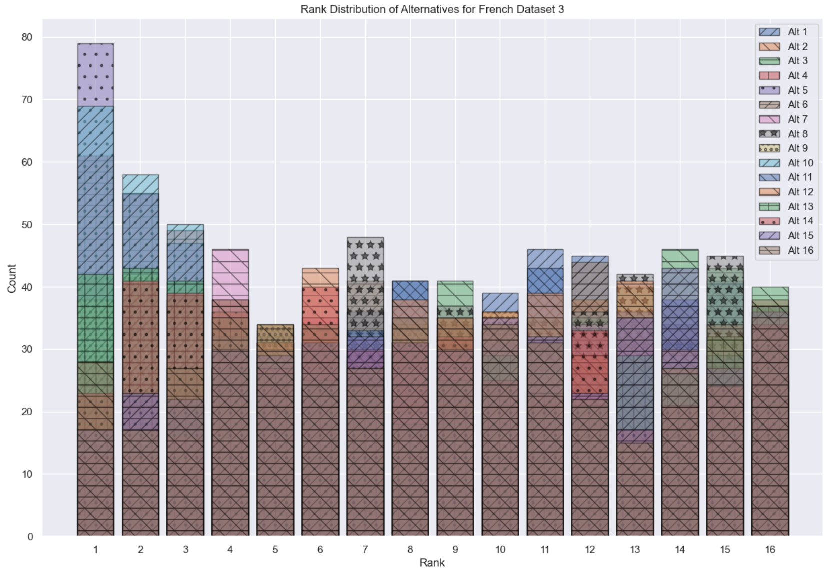


Figure 4.8: Alternative Rank Counts for French Index 3

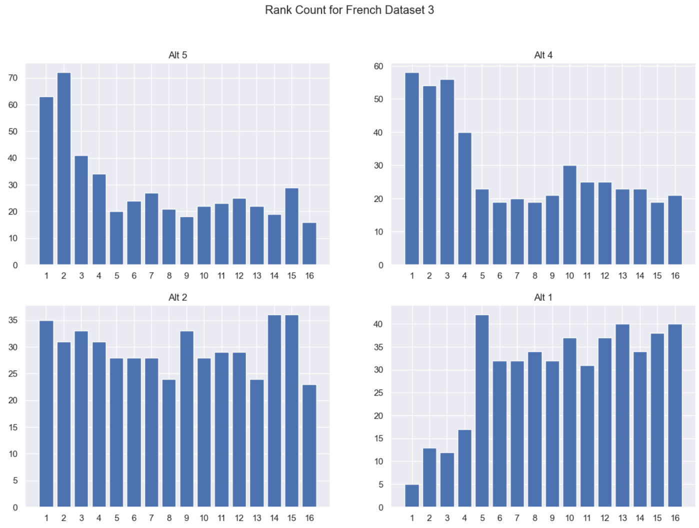


Figure 4.9: Alternative Rank Counts for 5,4,2,1 for French Index 3

For randomized voting rules, because distortion measures against the best alternative, voting rules which assign more weight to lesser scoring alternatives are going to perform considerably worse than rules that assign more relative weight to the top ranks. This is especially noticeable for Borda which is virtually identical to the real score distributions, yet performs worse than Plurality. The harsher penalties that Plurality and Harmonic give to weaker performing alternatives result in higher scores. For this reason, the RSD of the score distributions for each voting rule is highly relevant, as the higher RSDs correlate to placing more weight on the better performing alternatives. This is especially apparent when looking at the rank counts for each alternative in Figure 4.8, whereby the top ranks contribute the most to the total counts for the best performers. Figure 4.9 displays the rank counts for alternatives 5, 4, 2, and 1 for better clarity. We can see the trend of how better-performing alternatives have frequencies in a semi-descending monotonic order, middling alternatives have a semi-uniform frequency, and the worse performing alternatives have a semi-ascending monotonic order.

For a given alternative, the frequencies of ranks were calculated. Then the difference between the frequency of each adjacent rank was calculated with np.diff. This value was then averaged to give a rough estimate of the monotonicity. This was done for the best performing alternative, the middle performing, and the worse for each dataset.

Across all datasets, the findings here were similar to the observations with the *"french"* dataset. The full data for each file can be found in mono.csv

4.0.1 Different Distributions

The probability distribution used to assign scores has a large effect on distortions. Figure 4.10 shows how the distribution of scores changes between logarithmic and exponential for high m. While it was expected that the interaction between the distribution of rank frequencies would affect the distortions, we found that overall exponential distributions provided the worst distortions.

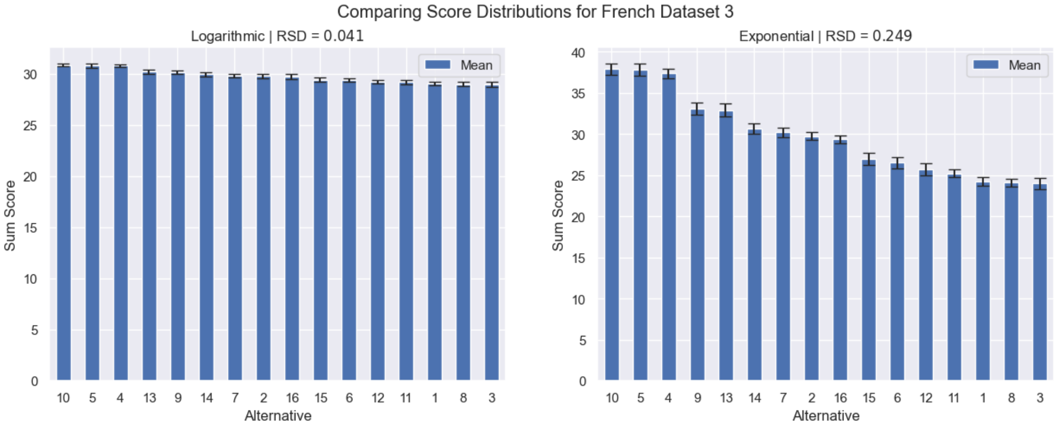


Figure 4.10: Logarithmic vs Exponential Distributions for French Dataset 3

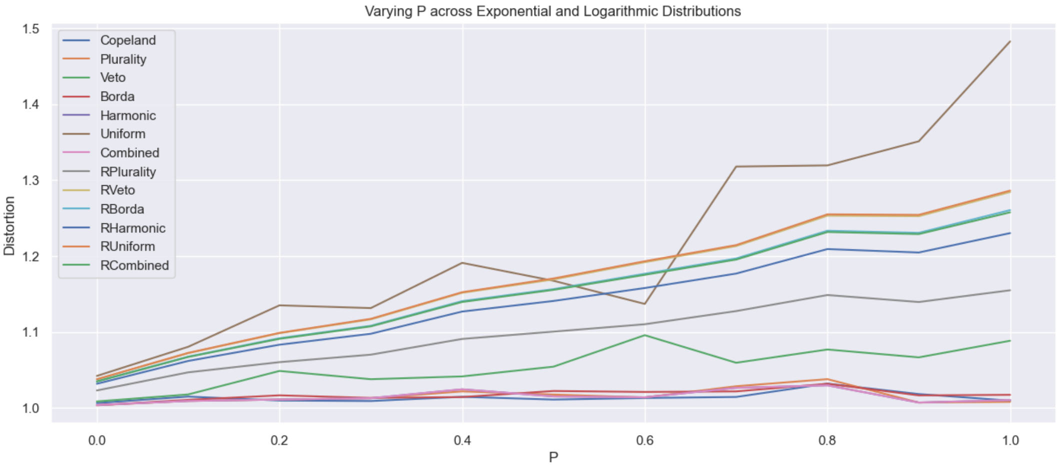
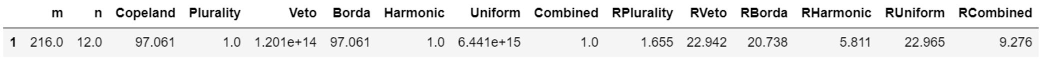


Figure 4.11: Distortion vs Varying Mixed Distributions

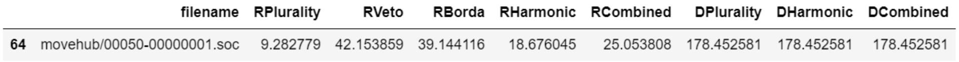
We designed a mixed distribution function with a parameter p between 0-1 that accepts two distribution functions. For each order, distribution 1 is used with a probability p, or distribution 2 is used with a probability $1-p$. Figure 4.11 shows how a mixed distribution affected the distortion as we varied the p from 0 to 1 between exponential and logarithmic distribution functions. We can clearly see that $p = 1$, when the exponential distribution is fully used that we achieve the highest distortions. This experiment was repeated across a few representative data sets and showed similar patterns, thus we used only exponential distributions for computational efficiency for all the distortions in the experiment. Intuitively this makes sense since the difference in scores, the low RSD for logarithmic will result in very low penalties and thus distortions.

4.0.2 High M Datasets



	m	n	Copeland	Plurality	Veto	Borda	Harmonic	Uniform	Combined	RPlurality	RVeto	RBorda	RHarmonic	RUniform	RCombined
1	216.0	12.0	97.061	1.0	1.201e+14	97.061	1.0	6.441e+15	1.0	1.655	22.942	20.738	5.811	22.965	9.276

Figure 4.12: MoveHub Distribution Distortions



	filename	RPlurality	RVeto	RBorda	RHarmonic	RCombined	DPlurality	DHarmonic	DCombined
64	movehub/00050-00000001.soc	9.282779	42.153859	39.144116	18.676045	25.053808	178.452581	178.452581	178.452581

Figure 4.13: MoveHub LP Distortions

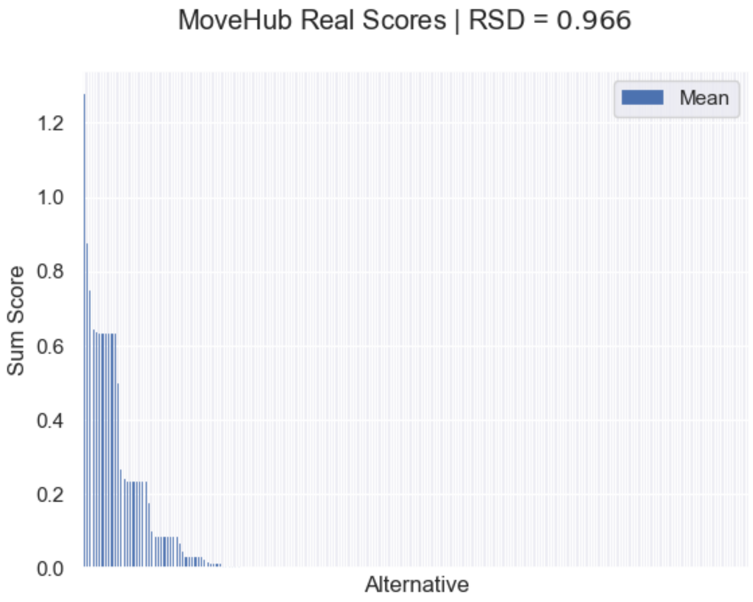


Figure 4.14: MoveHub Real Score Distribution

For datasets with high *m*, distortions become more pronounced and the differences between voting rules become more distinct. The *"MoveHub"* dataset as shown in Figure 4.12 and Figure 4.11 is an unusual dataset, with an m = 216 and an n of 12. With an RSD of 0.966, the real score distribution is extremely skewed, showing similar behavior to Zipf's law. With n greatly less than m, Veto degenerates into uniform, and their extremely high distortions are expected with the penalty of picking against the best alternative being so high. *"MoveHub"* does not have a Condorcet Winner, and Copeland's does not perform well here. The LP Distortions reveal similar relative performance as expected, with Plurality, Harmonic, Combined, Borda then Veto in order of performance. While high, the deterministic distortions are far short of the theoretical upper bounds. A distortion less than the m of 216 is quite good and should highlight that the profiles required for the worst bounds may not be that common.

Figure 4.15 shows the top 5 scoring alternatives given by Borda, of which alternative "121" is the 4th best. Given Borda's standard deviation of 519.41, it was "close" for Borda to achieve the best alternative, however, the massive distortions for picking the wrong choice should highlight the limitations of deterministic mechanisms. In contrast, Plurality's standard deviation of 0.054 means that the mean score of 2 for "121" would clearly make it the best alternative. This is the same for harmonic and combined scores, where we can see most alternatives being well out of the cutoff. The exception is '150' for Combined, but '121' is still far above. The cutoff of 1 standard deviation should thus illustrate the extent of "certainty" each voting mechanism has with its best choice.

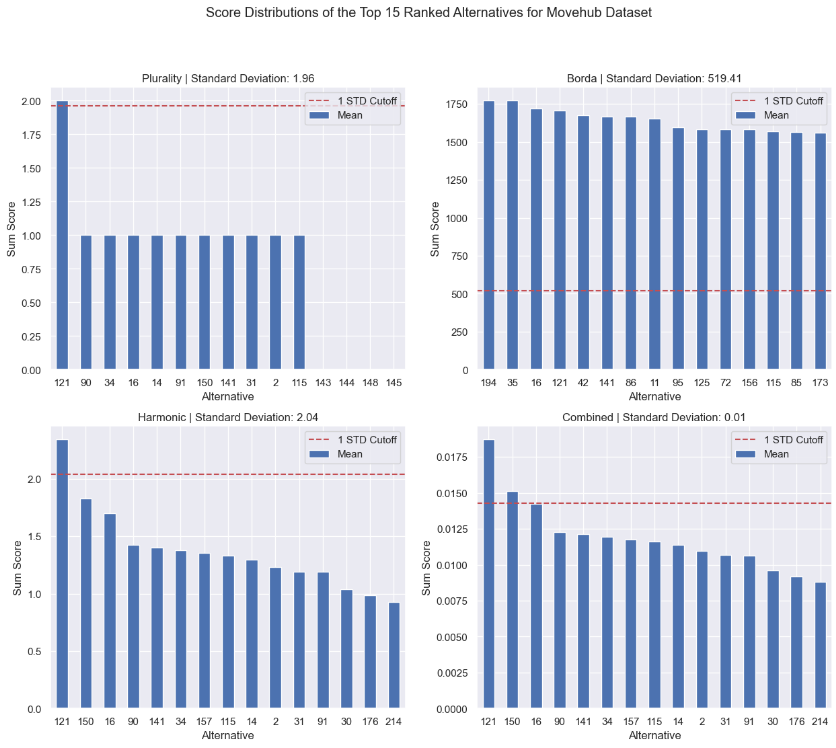


Figure 4.15: Score Distributions for the Top 15 Ranked Alts for Movehub

The *"University"* dataset with $m = 200$ and $n = 19$ confirms similar results in Figure

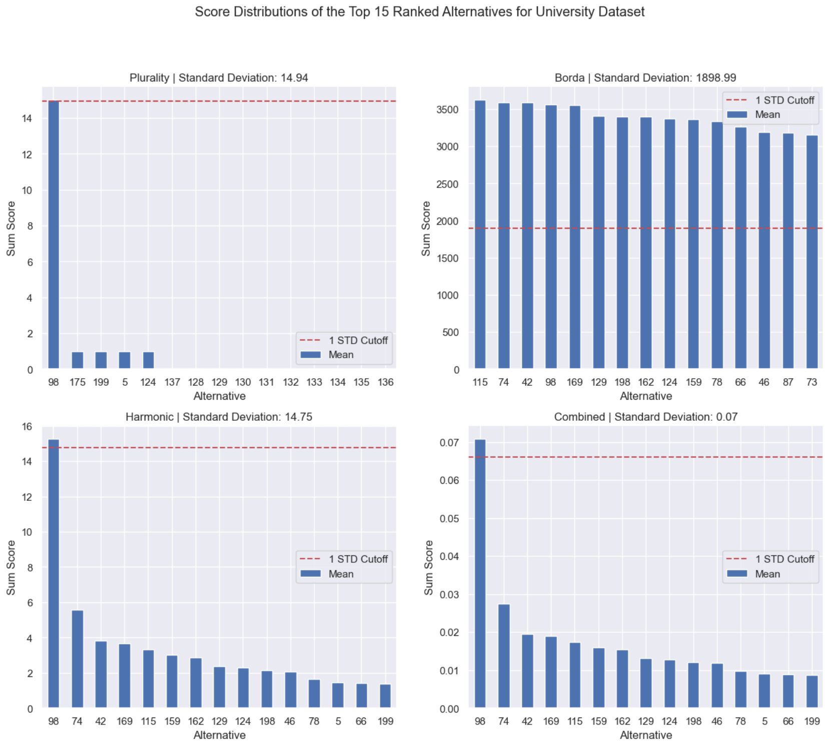


Figure 4.16: Score Distributions for the Top 15 Ranked Alts for University

4.16. In fact for Plurality, Harmonic and Combined the ”certainty” is even more extreme here, where the cutoff is far higher than every other alternative save for plurality.

4.0.3 Synthetic Datasets: Varying *m* against *n*

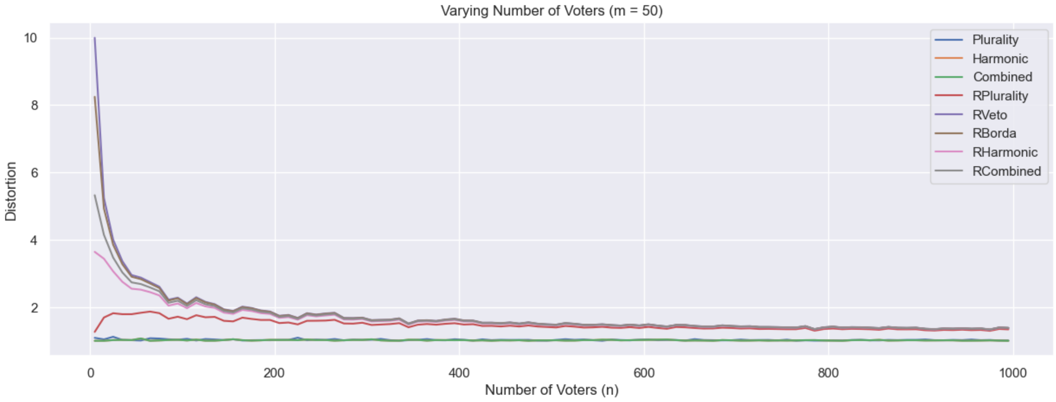


Figure 4.17: Varying N against M

The high m datasets are characterized by having a much higher m than n, leading to real score distributions of very high RSD. However, we may want to see how distributions behave when the ratio between the two is not so high. Prior studies

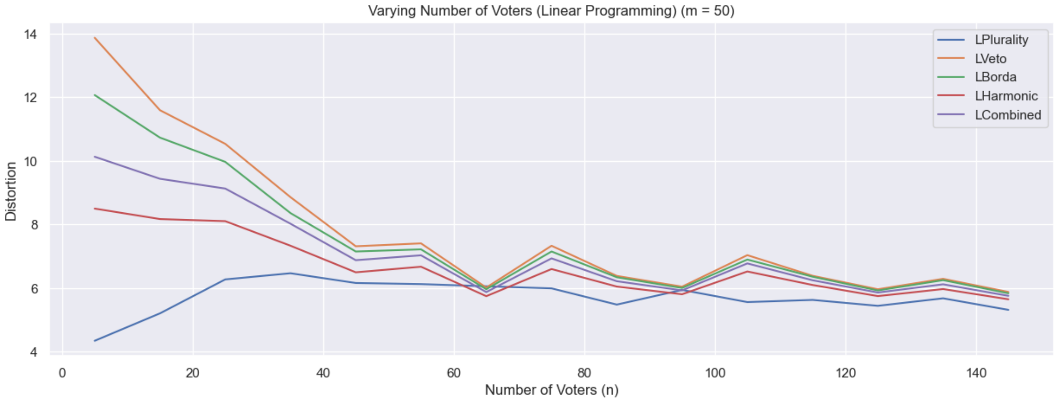


Figure 4.18: Varying N against M (LP Worse Case)

by Caragiannis et Al. [2] have found that when $m \approx n$, distortions can be go down drastically for randomized voting functions. If the $m > n$ by a great amount, then distortion may even reach close to 1, similar to the results we observed with the low m data. As such, we want to see if such a behaviour would be common for a uniform dataset sampled from a Mallow's mix model.

With Mallow's Model Mix sampling, we decided to vary n across a fixed m to see how the distortions and RSD would change. A single random reference was used to build a uniform (i.e) totally random preference profile with a dispersion of 1.

Figure 4.17 shows the distortions for the uniform sampling. Veto and Borda's distortions are not shown for clarity due to their enormous distortions early on. As we can see, the distortion declines as the number of voters increases after the inflection point of 50. This matches Caragiannis et Al.'s findings. A high m alone will not guarantee a high distortion, the ratio of n to m also matters. However, since the distortions plateau at around 100-200, increasing m will still force a minimum distortion. The randomized voting functions do not actually reach 1 and do not appear to be decreasing. The distortions from the LP in Figure 4.18 show that this still holds even in the worst case score profiles. We did not test for large m for the LP due to computational limits. It is also interesting to note how the scores are increasingly clustered after the inflection point, such that the voting rules become highly homogeneous. This may be the behavior we saw with the low m datasets whereby n was greatly higher than m.

4.0.4 Spoiler Effect Testing

The Spoiler Effect is commonly well known phenomenon with the Plurality voting system. It is when voters with similar preferences that are split across different candidates may lose against a single plurality winner that is not the best candidate. Most of the deterministic distortions over 1 for plurality can be seen as examples of the spoiler effect. However, the margins for these events to occur are very small, as the utility of the best alternative and the plurality winner is often very small, as shown by the previous results. We thus decided to build preference profiles that varied these voting margins to see how the voting mechanisms perform.

Our basic example of a Plurality-Loosing profile is with $m = 3$ with candidates A, B, C. 30% of voters have the profile 1: $A > C > B$, 30% have profile 2: $C > A > B$ and 40% have profile 3: $B > C > A$. B is the Plurality winner, but it is clearly not the majority winner as it lacks the pairwise majority for 60% of voters. For Mallow's Model Mix, these three orders are the reference elements. Each has a dispersion of 0.1 for a bit of randomness. Profile 3 is given a *largeWeight* weight that is varied from $(0-1)$ at steps of 0.1. Profile 1 and 2 are given a *smallWeight* weight that is defined by the equation. We sampled from $m = 100$ for each sampling and we repeated 50 times for each iteration and averaged to reduce the effect of randomness.

$$smallWeight = \frac{1 - largeWeight}{2}$$

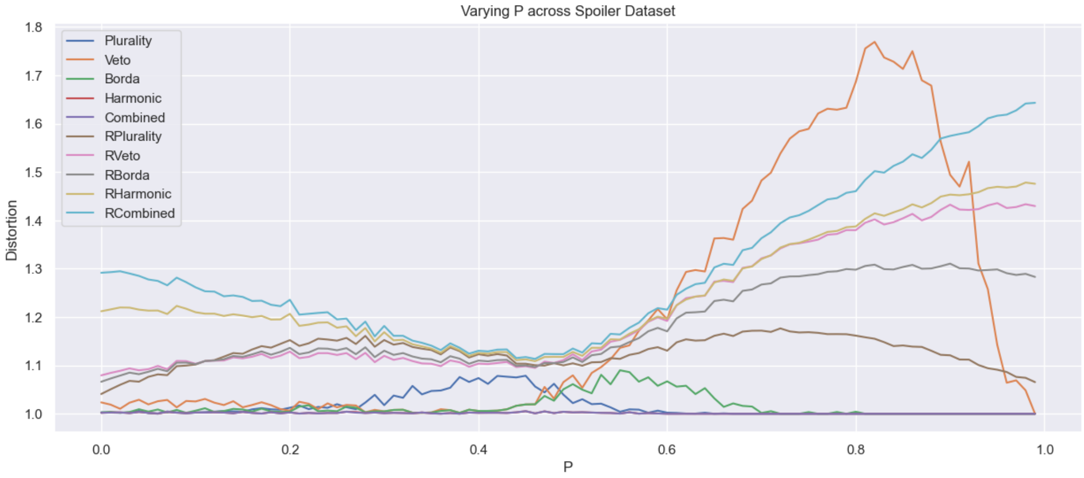


Figure 4.19: Varying Spoiler Effect

In figure 4.19, we can see that from around 0.2 – 0.6, Plurality performs at its worse, with Borda and Veto actually beating it until around 0.5. Randomized Veto and Borda beat other randomized rules including Plurality until around 0.4 until which they become worse. Note that as p increases, the preference profile become more homogeneous with order $(3 > 1 > 2)$. This is likely why Borda, Veto and the randomized voting mechanisms get worse as P increases, as more utility is assigned to alternative 3 such that the weights given to the rest of the alternative ”pull down” the score, similar to what was discussed earlier.

Chapter 5

Evaluation

5.0.1 General Evaluations

The low distortions and homogeneity across different voting mechanisms show how for the most part, most preference profiles encountered in the real world are simple enough for basic voting mechanisms to achieve high social welfare. Even with the worst-case LP distribution of scores, the distortions never reached the theoretical bounds and all stayed nestled within the linear range of m. Due to the thin margins between voters, the spoiler effect's actual social cost is overstated, as even the non-Condorcet-winner will be bringing in large social welfare anyways.

What is more surprising is the dominance of plurality over all other voting rules. While it is expected for deterministic plurality that its worst-case scenario is the best we can do, randomized plurality beats all others despite having a higher theoretical bound $\Theta(m\sqrt{m})$ than all other voting mechanisms. The relative performance of the randomized mechanisms is inverted compared to their theoretical bounds, in which Veto and Uniform should perform the best, yet they perform the worst. In our datasets, this is due to how the high RSD of the plurality allowed it to place less weight on poorly performing alternatives than the other mechanisms. And since we found that most high utility alternatives had a negative monotonicity, aka they derived most of their votes from the top ranks, then plurality would always be successful. While our distributions are experimental assumptions, the rank distributions are not. And the LP results confirm similar results. This thus would imply to reach the theoretical bounds would require highly contrived preference profiles.

The extra operations with varying m and n and the spoiler effect should also illustrate how the bad-case situations can be mitigated. High m situations can be reduced to very low and very homogeneous distortions with sufficiently high n, which is not unrealistic in real world scenarios. The presence of extra voters reducing distortion is similar to the "wisdom of the crowds" effect in how adding increasing information can often help converge to optimal choices. The 0.2-0.6 margin does show that the conditions required for the Spoiler effect to occur is not insignificant, but even in the situation that it does occur the distortions will be small anyways due to the previous properties shown with the monoticity of most data sets. These results should thus show how the bad-case situations would likely be uncommon in reality or how can be mitigated regardless.

Much of the motivation behind social choice theory and the notion of distortion has been to understand the limitations of real-world voting systems. As the most commonly well-known and popular system, plurality is a system we would want to nonintuively "show" that is not the best choice in all scenarios and has limitations. And such limitations have been studied and showed in prior works. But in this study, given the assumptions we've made, it would appear that the intuitive "trust" in plurality for the common voter would be generally well-founded.

5.0.2 Future Directions

The largest limitation of this study was the choice of how to distribute the scores. While we experimented with a random choice between logarithmic and exponential and found that exponential provided worse scores, realistically score distribution would not be random. Score distribution would likely be clustered into groups with similar preference profiles and may use more extreme distributions like some form of k-ranks. While the worst-case situations can be found through running linear programming, more work could be done to try to model how real-world utilities are actually distributed. Another direction would also obviously be to test out more complex voting methods that mirror real-world elections such as the use of IRV in the Australian elections [8].

Chapter 6

Conclusions

Through running distortion experiments with real preference profiles, we found that most voting mechanisms performed quite well, and plurality performed the best despite having worse theoretical worst-case bounds than other competing mechanisms. Through looking at the rank distribution and the score distribution, we found that because most high welfare alternatives also derived most of their utility from high ranks, hence the mechanisms that gave more weight to these high ranks would perform much better and with more certainty. We also found that the spoiler effect only really occurred during a small margin between the frequency of voting profiles with low distortion, and how the sufficiently large n over small m could push distortion scores very low, matching prior theoretical studies. In conclusion, our findings thus find that in most real cases, distortions can be kept very small, and simple rules such as plurality suffice in providing the best or very high utility alternatives. The worst-case situations appeared to rely on more contrived and unique preference profiles that may not be very common in reality.

Bibliography

[1] Anshelevich, Elliot., Aris Filos-Ratsikas, Nisarg Shah and Alexandros A. Voudouris. *“Distortion in Social Choice Problems: The First 15 Years and Beyond.”*. International Joint Conference on Artificial Intelligence (2021).

[2] Caragiannis, Ioannis. and Ariel D. Procaccia. *“Voting almost maximizes social welfare despite limited communication.”* Artif. Intell. 175 (2010): 1655-1671.

[3] Boutilier, Craig, Ioannis Caragiannis, Simi Haber, Tyler Lu, Ariel D. Procaccia and Or Sheffet. *“Optimal social choice functions: a utilitarian view.”* Artif. Intell. 227 (2012): 190-213.

[4] Arrow, Kenneth J. *“Social Choice and Individual Values”* (1951).

[5] Wong, Kin Hei *”Informatics Project Proposal: Distortion with Real Data”* (2023).

[6] Boutiller, Craig. and Tyler Lu. *”Effective sampling and learning for mallows models with pairwise-preference data”*. J. Mach. Learn. Res. 15 (2014): 3783-3829.

[7] Ebadian, Soroush., Aris Filos-Ratsikas, Mohamad Latifian, Nisarg Shah. *”Explainable and Efficient Randomized Voting Rules”*. (2023)

[8] Australian Government. *”Australian Election Commission”*. Last Accessed 22 August 2023. `https://www.aec.gov.au/learn/preferential-voting.html`

[9] Mattei, Nicholas. and Toby Walsh *”PrefLib: A Library of Preference Data”*. Proceedings of Third International Conference on Algorithmic Decision Theory (ADT 2013). `https://www.preflib.org/`

[10] Riley, Jim., Ivan Ryan, Warren D Smith.*”The Election, by Instant Runoff Voting, of UK Labour Party Leader in late-September 2010”* (2010). `http://rangevoting.org/LabourUK2010.html`

[11] Laslier, Jean-Francois. and Karine Van der Straeten. *”A Live Experiment on Approval Voting”*. Experimental Economics 11: 97-105 (2008).

[12] O'Neill, Jeffrey. *"Open STV"*. (2013). `http://www.openstv.org`

[13] Bennett, James. and Stan Lanning. *"The Netflix Prize"*. Proceedings of The KDD Cup and Workshops (2007)

[14] Regenwetter, Michel, Aeri Kim, Arthur Kantor and Moon-Ho R Ho. "The Unexpected Empirical Consensus Among Consensus Methods." Psychological Science 18 (2007): 629 - 635.

[15] Boehmer, Niclas and Nathan Schaar. "Collecting, Classifying, Analyzing, and Using Real-World Elections." ArXiv abs/2204.03589 (2022)

Appendix A

First appendix

A.1 Datasets

- (Files 00026). 2002 French Presidential Elections. 6 Instances corresponding to 6 regions across France [9]. Collected by Jean-Francois Laslier and Karine Van der Straeten [11].
- (Files 00030). 2010 UK Labor Party Leadership Vote [9]. Collected by Jim Riley et Al [10].
- (Files 00019). 2010 Oakland, CA City Council and Mayoral Elections [9]. Collected by Jeffrey O'Neill [12].
- (Files 00005). 2009 Burlington, Vermont Mayoral Election [9]. Collected by Jeffrey O'Neill [12].
- (Files 00016). 2009 Aspen, CO City Council and Mayoral Elections [9]. Collected by Jeffrey O'Neill [12].
- (Files 00020). 2008 Pierce, WA County Elections [9]. Collected by Jeffrey O'Neill [12].
- (Files 00004). Netflix Prize Data that contains the ratings of movies by users, converted into orders [9]. Collected and Converted by James Bennet and Stan Lanning [13].
- (Files 00028). American Psychological Association Elections between 1998-2009 [9]. Collected by Michel Regenwetter et Al. [14].

- (Files 00001). Dublin, Ireland North, West and Meath elections in 2007 [9]. Collected by Jeffrey O'Neill [12].
- (Files 00021). 2008-2012 San Francisco, CA Elections, including Board of Supervisors, District Attorney and Mayoral Elections [9]. Collected by Jeffrey O'Neill [12].
- (Files 00046). 2012-2015 University Ranking preferences by students generated from indicator-based rankings [9]. Collected by Niclas Boehmer and Nathan Schaar [15].
- (Files 00050). MoveHub City Ranking, generated by indicator-based rankings of cities to move to [9]. Collected by Niclas Boehmer and Nathan Schaar [15].

A.2 GitHub Repository Link

`https://github.com/Knhwong/Dissertation/settings`