
A Social Choice Application

Matt Dickenson

Department of Political Science
Duke University
Durham, NC 27708
mcd31@duke.edu

Abstract

This project implements a social choice web application, with an example use case of a group deciding on which movie to see. Using a formal voting process reduces the complexity of preference elicitation, prevents cascade effects arising from sequential preference revelation, and ensures privacy for voters, reducing the incentive for preference falsification. The Borda rule exhibits a number of desirable characteristics including anonymity, neutrality, unanimity, universal domain, and reinforcement. The major downside of Borda count is its manipulability; however, Monte Carlo simulations show that such manipulation should be rare in practice (especially with many voters) and that the decreases in average utility are relatively modest. Employing this application could greatly improve the coordination process for informal social gatherings.

1 Introduction

1.1 Motivation

This project consists of three parts, centered around the implementation of a social choice web application. The application uses a formal voting process to reduce the complexity of preference elicitation. The Borda rule exhibits a number of desirable characteristics including anonymity, neutrality, unanimity, universal domain, and reinforcement. The major downside of Borda count is its manipulability. Monte Carlo simulations are used to assess the frequency and impact of such manipulations compared to the plurality and veto rules. These results suggest that manipulation should be rare in practice (especially with many voters) and have a modest impact on average utility. This suggests that employing the application's voting procedure could improve outcomes for coordinating group outings.

1.2 Problem Definition

The software application is intended to be used by a group coordinating on a single outcome by aggregating their preferences, e.g. selecting a movie for a night out with friends. A user creates a movie poll by supplying candidate options (movies) and voters (a list of email addresses to be contacted). A unique link is then sent to each voter, who supplies her ranking of the choices. When all votes have been recorded, an email is sent to all voters with the outcome (and possibly more info, such as anonymized rankings), and a link where they can indicate their satisfaction with the outcome. Using this application reduces the complexity of preference elicitation and avoids other undesirable issues that arise in informal group decision-making.

2 A Social Choice Application

2.1 Voting Methods

Desirable properties of the voting method for this application arise from both normative and pragmatic concerns. Anonymity and neutrality ensure that the names of the voters and candidates, respectively, do not influence the outcome. Candidates can be presented to voters in a randomized order to ensure that the format of the list does not bias results. Unanimity requires that if candidate A is preferred over B by all voters, B should not win. The universal domain of potential ballots will be considered in the Monte Carlo simulations, to ensure that the application works with any opinion a voter might entertain about the candidates. Finally, we desire a rule with reinforcement, so that if disjoint sets of voters select the same winner, that candidate should win when the voters are pooled. Practically, this ensures that two small groups of friends who use the app and arrive at the same movie should also have seen that movie if they voted as a single group.

If there are two candidates, the only social decision method that satisfies neutrality, anonymity, and positive responsiveness (if a candidate is tied for the win and moves up in rankings, it will become the winner) is majority rule [?]. Less strict conditions, such as Pareto optimality, also motivate the use of majority rule even if positive responsiveness is thought to be too strict [? ?]. If there are more than two candidates, only a scoring rule will satisfy anonymity, neutrality, reinforcement, and continuity (if two subgroups of voters select different winners, then there is some number k such that k copies of the first subgroup plus a single copy of the second subgroup will elect the first group's winner) [?]. Scoring rules have the additional advantage of being computationally easy.

What voting rule should we choose? Borda count chooses the majority alternative (whenever it exists), and satisfies anonymity, neutrality, unanimity, and reinforcement. A major downside of the Borda rule is that it is easily manipulable [?]. However, for a particular type of manipulation (Constructive Coalitional Weighted-Manipulation), manipulating Borda is NP-complete for three candidates, and it is NP-hard for another manipulation (Constructive Individual Unweighted-Manipulation) under uncertainty about others' votes [?]. More than three candidates should be present for most use cases of the application proposed here. Borda is not unique in this respect: with three or more candidates, there is no reasonable voting rule that is non-manipulable [? ? ?]. To assess the relative merits of using Borda in this application, Monte Carlo simulations will be used to estimate its vulnerability to manipulation compared to the plurality and veto rules.

2.2 User Experience

The application described above has been implemented using the Rails framework in the Ruby language. Although the application has not been deployed publicly, the screenshots in Figure 1 give a sense of the user experience. The basic usage process is as follows:

1. Upon arriving at the homepage, a user is invited to create a poll by supplying candidate options (movie-showtime-location tuples drawn from the Fandango API—see the left pane of Figure 1)
2. The user then invites voters by supplying their names and email addresses
3. An email is sent to each voter inviting her to participate in the poll
4. If the voter chooses to cast a vote, she clicks the unique link in the email (to avoid the need for remembering the password for a user account) and is taken to the voting page (right pane of Figure 1)
5. When all votes are recorded or a deadline has been reached (minimum of one hour before the first candidate showtime), an email is sent to all voters with the outcome

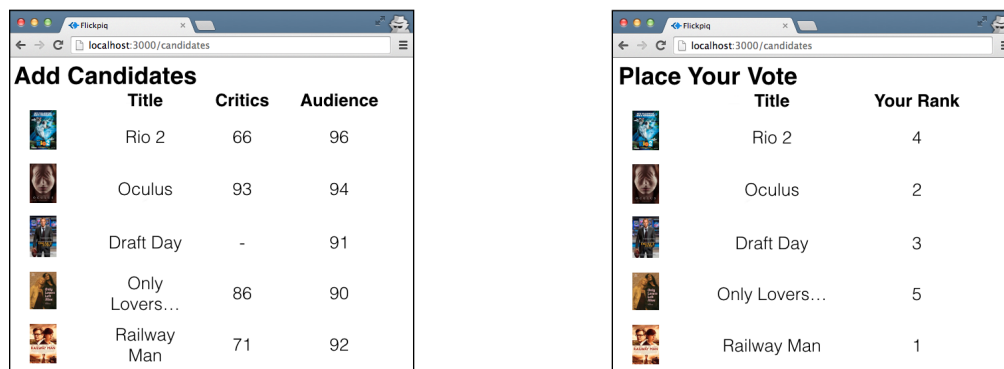


Figure 1: Example screens for poll creator to add candidate movies (left) and for voter to rank movies (right).

2.3 Complexity of Elicitation

How does using this application improve the process of selecting a movie for a group outing? One advantage of this method is that it reduces the average complexity of preference elicitation. When a group coordinates directly over email, there are several disadvantages. First, the pool of voters is not pre-defined: anyone could be added to or omitted from the email conversation at any time. Second, the domain of candidates is not fixed: a new candidate could be added at any time (even when the group is nearing a consensus), extending the time to elicitation exponentially. Third, preferences are expressed in sequence, leading to potential cascading effects resulting in suboptimal outcomes [? ?]. Fourth, because preferences are not expressed privately voters may feel pressure to engage in preference falsification.

The use of a voting application overcomes each of these disadvantages. The number of messages needed in a traditional email exchange is combinatorial in the number of voters, n , and candidates, m . the number of email messages sent by the voting application is $O(n)$: one message for each voter to request her vote, and one message to each voter to reveal the outcome.

3 Simulation and Analysis

This application reduces the complexity of preference elicitation (as described above), but how vulnerable is it to manipulation? Manipulability is a known weakness of the Borda voting method. To assess the risk of manipulation in this application, simulations were conducted to compare Borda count to the plurality and veto rules.

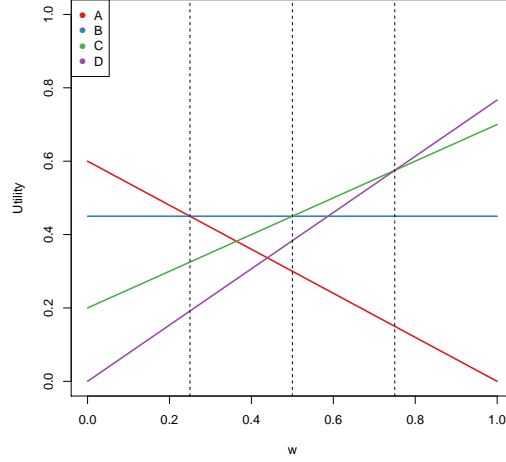


Figure 2: A voter’s utility for four example movies as a function of w . Dashed lines indicate changes in the first-ranked candidate.

3.1 Manipulation Method

The manipulation method explored is a utility-maximizing variant of the Find-Two-Winners (FTW) manipulation [?]. The version employed in this paper is unweighted (all votes have weight $w = 1$) and there is a single potential manipulator. The manipulator M calculates a strategic vote as follows:

1. Let v_M denote M ’s vote, initialized to be her true ranked preferences.
2. Run the voting rule with v_M and all other voter’s true votes (or an estimate of their votes, if simulating without perfect information). Let the winner of this vote be denoted c_1 .
3. For every other candidate $c_2 \neq c_1$:
 - Let \hat{v}_{M,c_2} be a vote in which M ranks c_2 first and c_1 last (leaving all other rankings unchanged)
 - Run the voting rule with \hat{v}_{M,c_2} replacing M ’s true preferences, leaving all other votes unchanged. Denote the winner of this vote c .
 - If $U_M(c) > U(c_1)$, let $v_M = \hat{v}_{M,c_2}$.
4. Return v_M

3.2 Simulation Parameters

Voting was simulated with three different rules: Borda count, plurality, and veto. In addition to the manipulation method, a number of additional simulation parameters were used to model the voting process. Movies, m , were treated as having two distinct attributes that influence utility: a critic rating, r_c , and an audience rating, r_a (both expressed as a percentage). Voters, v , differ in the weights, w , that they assign to critics’ ratings, so that $U_{v_i}(m_j) = w \cdot r_c(m_j) + (1 - w) \cdot r_a(m_j)$. The four movies used for the simulations had audience and critic ratings of (0.00, 0.60), (0.45, 0.45), (0.70, 0.20), and (0.77, 0.20). Although somewhat unrealistic as ratings for actual movies, these values were chosen so that each movie would be ranked first over one-fourth of the range of w (see Figure 2).

Voters’ weights were sampled from the Beta distribution with parameters $(\alpha, \beta) = \{(9, 1), (7, 3), (5, 5), (3, 7), (1, 9)\}$. This allows the mean of w to vary from 0.1 to 0.9 by increments of 0.1. The manipulator’s weight values varied along the same range, but were drawn independently of the voters’. In half of simulations the manipulator knew the other voters’ weights exactly, and in the other half she had to estimate them by maximum likelihood from the common prior (i.e. α and β were known to the

manipulator). The number of non-manipulating voters was varied from one to 10. 100 simulations were run for each combination of these parameters.

3.3 Frequency of Manipulation

How frequently was manipulation observed in the simulations? Overall, manipulation occurred in 18.7 percent of simulations using the Borda rule, 2.1 percent of simulations using plurality, and 80 percent of simulations with the veto rule. For both Borda and plurality, manipulation was observed less often under imperfect information (see Table 1). For these two rules the rate of manipulation decreases in the number of non-manipulating voters, and the difference between them is negligible with eight or more non-manipulators voting (see Figure 3). In reality the manipulator will have imperfect information about others' votes, suggesting that actual manipulations with this application will be uncommon.¹

Table 1: Frequency of Manipulation by Voting Rule

| | Perfect Information | Imperfect Information |
|-----------|---------------------|-----------------------|
| Borda | .20 | .17 |
| Plurality | .03 | .01 |
| Veto | .80 | .80 |

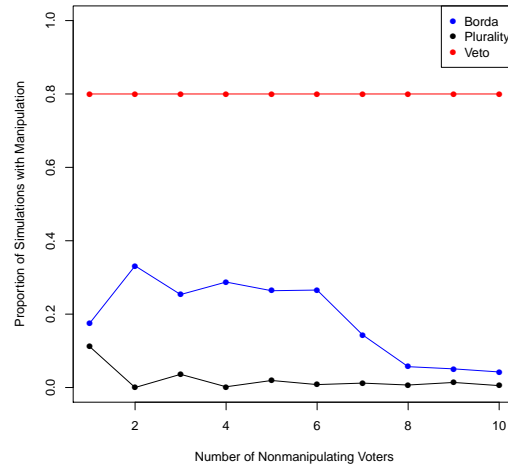


Figure 3: Observed frequency of manipulation by voting rule and number of non-manipulating voters.

3.4 Utility With and Without Manipulation

How do voters fare under each of the three voting rules?² Unsurprisingly, non-manipulators have lower average utility when manipulation occurs (Table 2). Manipulators do better on average when their manipulation is successful (unsurprisingly, since the manipulation attempt is utility-maximizing) except under the veto rule. Although this is odd, further inspection revealed that the average under manipulation with the veto rule was being pulled down by a few outliers, while the maximum with manipulation was much higher (again, for the manipulator) than without. Histograms for the observed utility distributions in 50,000 simulations with the

¹A pre-round of pairwise competition between alternatives could further reduce the vulnerability of the voting process to manipulation [?].

²Another example assessing the impact of manipulation on social welfare is found in [?].

Borda count rule are shown in Figure 4. The modest decrease in average utility observed under manipulation, accompanied by the relative infrequency of manipulation, suggest that the Borda rule is acceptable for this application.

Table 2: Average utilities by scenario

| | Manipulator | | Nonmanipulator | |
|-----------|-------------------|----------------------|-------------------|----------------------|
| | With Manipulation | Without Manipulaiton | With Manipulation | Without Manipulaiton |
| Borda | .58 | .43 | .40 | .52 |
| Plurality | .55 | .44 | .43 | .52 |
| Veto | .51 | .55 | .43 | .45 |

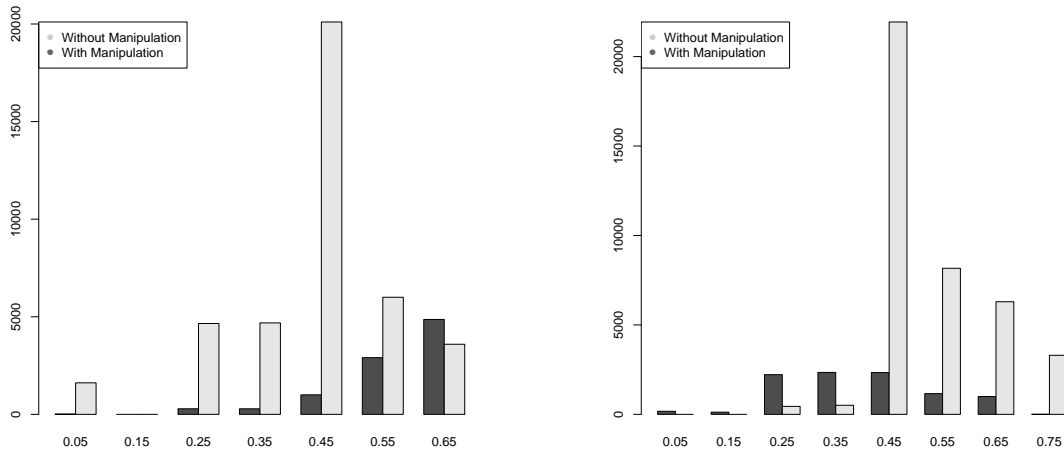


Figure 4: Manipulator's utility distribution (left) and nonmanipulators' average utility distribution (right) with and without manipulation for Borda count simulations.

4 Discussion

There are several promising avenues for further research on this project. A more sophisticated simulation model could consider a prior stage in which the poll organizer selects candidates from a larger pool of options, and assess the effect that this has on the outcome. The voting rule could also be left up to the participants by asking which criteria are desired for a particular situation. Instead of voting an auction could be used so that group members split the cost of the outing, with lower costs for those who dislike the outcome (setting a minimum threshold ensures that the outing occurs only if enough revenue is raised to pay for it). A more sophisticated model of movie-going would account for the combinatorial nature of movies—as combinations of showtimes, films, and locations—and allow voters to express preferences over each of these features separately. This would allow, for example, voters who feel strongly about the time of day but care less about what film they view to have their preferences taken into account; however, finding an outcome to satisfy all participants is computationally hard [?].

Implementing a web application in which users can employ Borda voting to coordinate a group decision extends the use of social choice to a popular setting. Using a formal voting process reduces the complexity of preference elicitation, prevents cascade effects arising from sequential preference revelation, and ensures privacy for voters, reducing the incentive for preference falsification. The Borda rule exhibits a number of desirable characteristics

including anonymity, neutrality, unanimity, universal domain, and reinforcement. The major downside of Borda count is its manipulability; however, Monte Carlo simulations showed that we should expect such manipulation to be rare in practice (especially with many voters) and that the decreases in average utility are relatively modest. Employing this application could greatly improve the coordination process for informal social gatherings.

Acknowledgments

Thanks to Vincent Conitzer and course participants for feedback on earlier versions of this project. Any conclusions or errors are the sole responsibility of the author.

References