Making Ethical Decisions by Learning and Aggregating Permutation Processes

Ritesh Noothigattu
Carnegie Mellon University
Pittsburgh, PA 15213
riteshn@cmu.edu

Ariel D. Procaccia
Carnegie Mellon University
Pittsburgh, PA 15213
arielpro@cs.cmu.edu

Iyad Rahwan
Massachusetts Institute of Technology
Cambridge, MA 02139
irahwan@mit.edu

Pradeep Ravikumar
Carnegie Mellon University
Pittsburgh, PA 15213
pradeepr@cs.cmu.edu

Abstract

We present a general approach to automating ethical decisions, drawing on machine learning and computational social choice. In a nutshell, we propose to *learn* a model of societal preferences, and, when faced with a specific ethical dilemma at runtime, efficiently *aggregate* those preferences to identify a desirable choice. We also provide a concrete algorithm that instantiates our approach; some of its crucial steps are informed by a new theory of social choice under *permutation processes*, which extend random utility models to arbitrary subsets of alternatives. Finally, we implement and evaluate our algorithm using synthetic data.

1 Introduction

The problem of ethical decision making, which has long been a grand challenge for AI [21], has recently caught the public imagination. Perhaps its best-known manifestation is a modern variant of the classic *trolley problem* [10]: An autonomous vehicle has a brake failure, leading to an accident with inevitably tragic consequences; due to the vehicle's superior perception and computation capabilities, it can make an informed decision. Should it stay its course and hit a wall, killing its three passengers, one of whom is a young girl? Or swerve and kill a male athlete and his dog, who are crossing the street on a red light? A notable paper by Bonnefon et al. [3] has shed some light on how people address such questions, and even former US President Barack Obama has weighed in.[1]

Arguably the main obstacle to automating ethical decisions is the lack of a formal specification of ground-truth *ethical principles*, which have been the subject of debate for centuries among ethicists and moral philosophers [18, 22]. In their work on fairness in machine learning, Dwork et al. [7] concede that, when ground-truth ethical principles are not available, we must use an "approximation as agreed upon by society." But how can society agree on the ground truth — or an approximation thereof — when even ethicists cannot?

We submit that decision making can, in fact, be automated, even in the absence of such ground-truth principles, by aggregating people's opinions on ethical dilemmas. This view is foreshadowed by a recent position paper by Greene et al. [8], who argue that the field of *computational social choice* [4], which deals with algorithms for aggregating individual preferences towards collective decisions, may provide tools for ethical decision making. But their specific suggestions — incorporating ground-truth

[1] https://www.wired.com/2016/10/president-obama-mit-joi-ito-interview/

ethical principles as the preferences of one of the agents, or capturing them through constraints on the aggregation process — again rely on the specification of ground-truth ethical principles.

By contrast, we propose a framework for ethical decision making based on computational social choice, which, we believe, is quite practical. In addition to serving as a foundation for incorporating future ground-truth ethical and legal principles, it could even provide crucial preliminary guidance on some of the questions faced by ethicists. Our approach consists of four steps:

I *Data collection:* Ask human voters to compare pairs of alternatives (say a few dozen per voter). In the autonomous vehicle domain, an alternative is determined by a vector of features such as the number of victims and their gender, age, health — even species!

II *Learning:* Use the pairwise comparisons to learn a model of the preferences of each voter over all possible alternatives.

III *Summarization:* Combine the individual models into a single model, which approximately captures the collective preferences of all voters over all possible alternatives.

IV *Aggregation:* At runtime, when encountering an ethical dilemma involving a specific subset of alternatives, use the summary model to deduce the preferences of all voters over this particular subset, and apply a voting rule to aggregate these preferences into a collective decision. In the autonomous vehicle domain, the selected alternative is the outcome that society (as represented by the voters whose preferences were elicited in Step I) views as the least catastrophic among the grim options the vehicle currently faces. Note that this step is only applied when all other options have been exhausted, i.e., all technical ways of avoiding the dilemma in the first place have failed, and all legal constraints that may dictate what to do have also failed.

Step I is out of the scope of the paper, but we note that it is possible to collect an adequate dataset through, say, Amazon Mechanical Turk. In fact, in order to perform a descriptive study, a team at the MIT Media Lab has elicited more than 20 million pairwise comparisons between alternatives in the autonomous vehicle domain, from more than 3 million voters, through the website Moral Machine.[2] This demonstrates the feasibility of implementing Step I on a large scale.

Let us, therefore, turn to Steps II, III, and IV. These steps rely on having a *model* for preferences. There is a considerable line of work on distributions over rankings over a *finite* set of alternatives. A popular class of such models is the class of *random utility models* [1, 2, 9, 12, 17], which use random utilities for alternatives to generate rankings over the alternatives. However, we are interested in situations where the set of alternatives is infinite, and any finite subset of alternatives might be encountered; for example, there are uncountably many scenarios an autonomous vehicle might face, as some features are continuous, but at runtime it will face a finite number of options. While there has been some work extending specific models for finite alternatives to the infinite alternative case [6], there is limited work on a general treatment of such infinite alternative models. Therefore, in Section 2, we introduce the notion of a *permutation process*, which is the general mathematical object characterizing distributions over rankings of an infinite set of alternatives.

In Section 3, we focus on developing a theory of aggregation of permutation processes, which is crucial for Step IV. Specifically, we assume that societal preferences are represented as a single permutation process. Given a (finite) subset of alternatives, the permutation process induces a distribution over rankings of these alternatives. In the spirit of *distributional rank aggregation* [16], we view this distribution over rankings as an *anonymous preference profile*, where the probability of a ranking is the fraction of voters whose preferences are represented by that ranking. This means we can apply a voting rule in order to aggregate the preferences — but *which* voting rule should we apply? And how can we compute the outcome *efficiently*? These are some of the central questions in computational social choice, but we show that in our context, under rather weak assumptions on the voting rule and permutation process, they are both moot, in the sense that it is easy to identify alternatives chosen by any "reasonable" voting rule. In slightly more detail, we define the notion of *swap dominance* between alternatives in a preference profile, and show that if the permutation process satisfies a natural property with respect to swap dominance (standard permutation processes do), and the voting rule is *swap-dominance efficient* (all common voting rules are), then any alternative that swap dominates all other alternatives is an acceptable outcome.

[2] http://moralmachine.mit.edu

Armed with these theoretical developments, the task of ethical decision making can be reduced to: learning a permutation process for each voter (Step II); summarizing these individual processes into a single permutation process that satisfies the required swap-dominance property (Step III); and using any swap-dominance efficient voting rule, which is computationally efficient given the swap-dominance property (Step IV).

In Section 4, we present an algorithmic instantiation of our approach, for a specific permutation process, namely the Thurstone-Mosteller (TM) Process [13, 20], and with a specific linear parametrization of its underlying utility process in terms of the alternative features. While these simple choices have been made to illustrate the conceptual framework, we note that, in principle, the framework can be instantiated with more general and complex permutation processes. We also present simulation results that show our approach, at the very least its instantiation for TM processes, provides a computationally and statistically attractive method for ethical decision making.

2 Permutation Processes

Let $\mathcal{X}$ denote a potentially infinite set of alternatives. Given a finite subset $A \subseteq \mathcal{X}$, we are interested in the set $\mathcal{S}_A$ of *rankings* over A. Such a ranking $\sigma \in \mathcal{S}_A$ can be interpreted as mapping alternatives to their positions, i.e., $\sigma(a)$ is the position of $a \in A$ (smaller is more preferred). Let $a \succ_\sigma b$ denote that a is preferred to b in σ, that is, $\sigma(a) < \sigma(b)$.

Turning to our first (conceptual) contribution, let $(\Omega, \mathcal{F}, P)$ denote a probability space, with sample space Ω, a σ-algebra $\mathcal{F}$ on Ω, and and a probability measure P.

Definition 2.1. A *permutation process* $\{\Pi(A) : A \subseteq \mathcal{X}, |A| \in \mathbb{N}\}$ is a collection of distributions defined on a common probability space $(\Omega, \mathcal{F}, P)$. A *sample path* of the permutation process is given by the mapping $\Pi : (\omega, A) \mapsto \sigma$, where $\omega \in \Omega$, A is a finite subset of alternatives, and $\sigma \in \mathcal{S}_A$ is a ranking over A.

Intuitively, given a finite subset of alternatives A, a permutation process defines a distribution over the rankings of these alternatives.

Definition 2.2. A permutation process Π is said to be *consistent* when its sample paths satisfy $\Pi(\omega, A)|_B = \Pi(\omega, B)$, for any finite subsets of alternatives $B \subseteq A \subseteq \mathcal{X}$, and any $\omega \in \Omega$, where $\sigma|_B$ denotes the ranking σ restricted to B.

In other words, for a consistent permutation process Π, the distribution induced by Π over rankings of the alternatives in B is nothing but the distribution obtained by marginalizing out the extra alternatives $A \setminus B$ from the distribution induced by Π over rankings of the alternatives in A.

As we show below, any permutation process that is consistent has a very natural interpretation.

Proposition 2.3. *Given any consistent permutation process Π over a set of alternatives $\mathcal{X}$ such that $|\mathcal{X}| \leq \aleph_1$, there exists a stochastic process U (indexed by $\mathcal{X}$) defined on the same probability space $(\Omega, \mathcal{F}, P)$ as the permutation process Π, such that for any $A = \{x_1, \ldots, x_m\} \subseteq \mathcal{X}$ and $\omega \in \Omega$, $\Pi(\omega, A) = \mathit{sort}(U_{x_1}(\omega), U_{x_2}(\omega), \cdots, U_{x_m}(\omega))$.*

The proposition asserts that the sample paths of the permutation process can be written as the sorted ordering of the corresponding sample paths of the stochastic process U.

Proof (sketch) of Proposition 2.3. Any sample path of a consistent permutation process can be seen to be a total ordering over the set of alternatives $\mathcal{X}$. Because $|\mathcal{X}| \leq \aleph_1$, any total ordering over $\mathcal{X}$ in turn can be expressed as the sorted order of some real values associated with each alternative in $\mathcal{X}$. For the sample path $\Pi(\omega, x_1, \ldots, x_m)$, denote this real-valued mapping by $(U_{x_1}(\omega), U_{x_2}(\omega), \cdots, U_{x_m}(\omega))$. Ranging over the outcomes $\omega \in \Omega$, the result follows. $\square$

We refer to the stochastic process corresponding to a consistent permutation process as its *utility process*, since it is semantically meaningful to obtain a permutation via sorting by utility. We note that we could allow the utility process to give ties, so long as the function sort$(\cdot)$ is endowed with some tie-breaking scheme, e.g., ties are broken lexicographically, which we will assume in the sequel.

As examples of natural permutation processes, we adapt the definitions of two well-known *random utility models*. The difference is that random utility models define a distribution over rankings over a fixed, finite subset of alternatives, whereas permutation processes define a distribution for each finite subset of alternatives, given a potentially infinite space of alternatives.

- **Thurstone-Mosteller (TM) Process** [13, 20]. A Thurstone-Mosteller Process (adaptation of Thurstone's Case V model) is a consistent permutation process, whose utility process U is a Gaussian process with independent utilities and identical variances. In more detail, given a finite set of alternatives $\{x_1, x_2, \cdots, x_m\}$, the utilities $(U_{x_1}, U_{x_2}, \cdots, U_{x_m})$ are independent, and $U_{x_i} \sim \mathcal{N}(\mu_{x_i}, \frac{1}{2})$, where μ_{x_i} denotes the mode utility of alternative x_i.
- **Plackett-Luce (PL) Process** [11, 15]. A Plackett-Luce Process is a consistent permutation process with the following utility process U: Given a finite set of alternatives $\{x_1, x_2, \cdots, x_m\}$, the utilities $(U_{x_1}, U_{x_2}, \cdots, U_{x_m})$ are independent, and each U_{x_i} has a Gumbel distribution with identical scale, i.e. $U_{x_i} \sim \mathcal{G}(\mu_{x_i}, \gamma)$, where $\mathcal{G}$ denotes the Gumbel distribution, and μ_{x_i} denotes the mode utility of alternative x_i. We note that Caron and Teh [6] consider a further Bayesian extension of the above PL process, with a Gamma process prior over the mode utility parameters.

3 Aggregation of Permutation Processes

In social choice theory, a *preference profile* is typically defined as a collection $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_N)$ of N rankings over a finite set of alternatives A, where σ_i represents the preferences of voter i. However, when the identity of voters does not play a role, we can instead talk about an *anonymous preference profile* $\pi \in [0,1]^{|A|!}$, where, for each $\sigma \in \mathcal{S}_A$, $\pi(\sigma) \in [0,1]$ is the *fraction* of voters whose preferences are represented by the ranking σ. Equivalently, it is the probability that a voter drawn uniformly at random from the population has the ranking σ.

How is this related to permutation processes? Given a permutation process Π and a finite subset $A \subseteq \mathcal{X}$, the distribution $\Pi(A)$ over rankings of A can be seen as an anonymous preference profile π, where for $\sigma \in \mathcal{S}_A$, $\pi(\sigma)$ is the probability of σ in $\Pi(A)$. As we shall see in Section 4, Step II (learning) gives us a permutation process for each voter, where $\pi(\sigma)$ represents our *confidence* that the preferences of the voter over A coincide with σ; and after Step III (summarization), we obtain a single permutation process that represents societal preferences.

Our focus in this section is the aggregation of anonymous preference profiles induced by a permutation process (Step IV), that is, the task of choosing the winning alternative(s). To this end, let us define an *anonymous social choice correspondence (SCC)* as a function f that maps any anonymous preference profile π over any finite and nonempty subset $A \subseteq \mathcal{X}$ to a nonempty subset of A. For example, under the ubiquitous *plurality* correspondence, the set of selected alternatives consists of alternatives with maximum first-place votes, i.e., $\arg\max_{a \in A} \sum_{\sigma \in \mathcal{S}_A :\, \sigma(a)=1} \pi(\sigma)$; and under the *Borda count* correspondence, denoting $|A| = m$, each vote awards $m - j$ points to the alternative ranked in position j, that is, the set of selected alternatives is $\arg\max_{a \in A} \sum_{j=1}^{m} (m-j) \sum_{\sigma \in \mathcal{S}_A :\, \sigma(a)=j} \pi(\sigma)$. We work with social choice *correspondences* instead of social choice *functions*, which return a single alternative in A, in order to smoothly handle ties.

3.1 Efficient Aggregation

Our main goal in this section is to address two related challenges. First, which (anonymous) social choice correspondence should we apply? There are many well-studied options, which satisfy different social choice axioms, and, in many cases, lead to completely different outcomes on the same preference profile. Second, how can we apply it in a computationally efficient way? This is not an easy task because, in general, we would need to explicitly construct the whole anonymous preference profile $\Pi(A)$, and then apply the SCC to it. The profile $\Pi(A)$ is of size $|A|!$, and hence this approach is intractable for a large $|A|$. Moreover, in some cases (such as the TM process), even computing the probability of a single ranking may be hard. The machinery we develop below allows us to completely circumvent these obstacles.

Since stating our general main result requires some setup, we first state a simpler instantiation of the result for the specific TM and PL permutation processes (we will directly use this instantiation in Section 4). Before doing so, we recall a few classic social choice axioms. We say that an anonymous SCC f is *monotonic* if the following conditions hold:

1. If $a \in f(\pi)$, and π' is obtained by pushing a upwards in the rankings, then $a \in f(\pi')$.
2. If $a \in f(\pi)$ and $b \notin f(\pi)$, and π' is obtained by pushing a upwards in the rankings, then $b \notin f(\pi')$.

In addition, an anonymous SCC is *neutral* if $f(\tau(\pi)) = \tau(f(\pi))$ for any anonymous preference profile π, and any permutation τ on the alternatives; that is, the SCC is symmetric with respect to the alternatives (in the same way that anonymity can be interpreted as symmetry with respect to voters).

Theorem 3.1. *Let Π be the TM or PL process, let $A \subseteq \mathcal{X}$ be a nonempty, finite subset of alternatives, and let $a \in \arg\max_{x \in A} \mu_x$. Moreover, let f be an anonymous SCC that is monotonic and neutral. Then $a \in f(\Pi(A))$.*

To understand the implications of the theorem, we first note that many of the common voting rules, including plurality, Borda count (and, in fact, all positional scoring rules), Copeland, maximin, and Bucklin [4], are associated with anonymous, neutral, and monotonic SCCs. Specifically, all of these rules have a notion of score, and the SCC simply selects all the alternatives tied for the top score (typically there is only one).[3] The theorem then implies that all of these rules would agree that, given a subset of alternatives A, an alternative $a \in A$ with maximum mode utility is an acceptable winner, i.e., it is at least tied for the highest score, if it is not the unique winner. As we will see in Section 4, such an alternative is very easy to identify, which is why, in our view, Theorem 3.1 gives a satisfying solution to the challenges posed at the beginning of this subsection. We emphasize that this is merely an instantiation of Theorem 3.7, which provides our result for general permutation processes.

The rest of this subsection is devoted to building the conceptual framework, and stating the lemmas, required for the proof of Theorem 3.1, as well as the statement and proof of Theorem 3.7. We relegate all proofs to Appendix A.

Starting off, let π denote an anonymous preference profile (or distribution over rankings) over alternatives A. We define the ranking σ^{ab} as the ranking σ with alternatives a and b swapped, i.e. $\sigma^{ab}(x) = \sigma(x)$ if $x \in A \setminus \{a, b\}$, $\sigma^{ab}(b) = \sigma(a)$, and $\sigma^{ab}(a) = \sigma(b)$.

Definition 3.2. We say that alternative $a \in A$ *swap-dominates* alternative $b \in A$ in anonymous preference profile π over A — denoted by $a \rhd_\pi b$ — if for every ranking $\sigma \in \mathcal{S}_A$ with $a \succ_\sigma b$ it holds that $\pi(\sigma) \geq \pi(\sigma^{ab})$.

In words, a swap-dominates b if every ranking that places a above b has at least as much weight as the ranking obtained by swapping the positions of a and b, and keeping everything else fixed. This is a very strong dominance relation, and, in particular, implies existing dominance notions such as *position dominance* [5]. Next we define a property of social choice correspondences, which intuitively requires that the correspondence adhere to swap dominance relations, if they exist in a given anonymous preference profile.

Definition 3.3. An anonymous SCC f is said to be *swap-dominance-efficient (SwD-efficient)* if for every anonymous preference profile π and any two alternatives a and b, if a swap-dominates b in π, then $b \in f(\pi)$ implies $a \in f(\pi)$.

Because swap-dominance is such a strong dominance relation, SwD-efficiency is a very weak requirement, which is intuitively satisfied by almost any "reasonable" voting rule. This intuition is formalized in the following lemma.

Lemma 3.4. *Any anonymous SCC that satisfies monotonicity and neutrality is SwD-efficient.*

So far, we have defined a property, SwD-efficiency, that any SCC might potentially satisfy. But why is this useful in the context of aggregating permutation processes? We answer this question in Theorem 3.7, but before stating it, we need to introduce the definition of a property that a *permutation process* might satisfy.

Definition 3.5. Alternative $a \in \mathcal{X}$ swap-dominates alternative $b \in \mathcal{X}$ in the permutation process Π — denoted by $a \rhd_\Pi b$ — if for every finite set of alternatives $A \subseteq \mathcal{X}$ such that $\{a, b\} \subseteq A$, a swap-dominates b in the anonymous preference profile $\Pi(A)$.

[3] Readers who are experts in social choice have probably noted that there are no social choice *functions* that are both anonymous and neutral [14], intuitively because it is impossible to break ties in a neutral way. This is precisely why we work with social choice *correspondences*.

We recall that a *total preorder* is a binary relation that is transitive and total (and therefore reflexive).

Definition 3.6. A permutation process Π over $\mathcal{X}$ is said to be *SwD-compatible* if the binary relation $\triangleright_\Pi$ is a total preorder on $\mathcal{X}$.

We are now ready to state our main theorem.

Theorem 3.7. *Let f be an SwD-efficient anonymous SCC, and let Π be an SwD-compatible permutation process. Then for any finite subset of alternatives A, there exists $a \in A$ such that $a \triangleright_\Pi b$ for all $b \in A$. Moreover, $a \in f(\Pi(A))$.*

This theorem asserts that for any SwD-compatible permutation process, any SwD-efficient SCC (which, as noted above, include most natural SCCs, namely those that are monotonic and neutral), given any finite set of alternatives, will always select a very natural winner that swap-dominates other alternatives. A practical use of this theorem requires two things: to show that the permutation process is SwD-compatible, and that it is computationally tractable to select an alternative that swap-dominates other alternatives in a finite subset. The next few lemmas provide some general recipes for establishing these properties for general permutation processes, and, in particular, we show that they indeed hold under the TM and PL processes. First, we have the following definition.

Definition 3.8. Alternative $a \in \mathcal{X}$ *dominates* alternative $b \in \mathcal{X}$ in utility process U if for every finite subset of alternatives containing a and b, $\{a, b, x_3, \ldots x_m\} \subseteq \mathcal{X}$, and every vector of utilities $(u_1, u_2, u_3 \ldots u_m) \in \mathbb{R}^m$ with $u_1 \geq u_2$, it holds that

$$p_{(U_a, U_b, U_{x_3}, \ldots U_{x_m})}(u_1, u_2, u_3 \ldots u_m) \geq p_{(U_a, U_b, U_{x_3}, \ldots U_{x_m})}(u_2, u_1, u_3 \ldots u_m), \tag{1}$$

where $p_{(U_a, U_b, U_{x_3}, \ldots U_{x_m})}$ is the density function of the random vector $(U_a, U_b, U_{x_3}, \ldots U_{x_m})$.

Building on this definition, Lemmas 3.9 and 3.10 directly imply that the TM and PL processes are SwD-compatible, and complete the proof of Theorem 3.1 (see Appendix A).

Lemma 3.9. *Let Π be a consistent permutation process, and let U be its corresponding utility process. If alternative a dominates alternative b in U, then a swap-dominates b in Π.*

Lemma 3.10. *Under the TM and PL processes, alternative a dominates alternative b in the corresponding utility process if and only if $\mu_a \geq \mu_b$.*

3.2 Stability

It turns out that the machinery developed for the proof of Theorem 3.1 can be leveraged to establish an additional desirable property.

Definition 3.11. Given an anonymous SCC f, and a permutation process Π over $\mathcal{X}$, we say that the pair (Π, f) is *stable* if for any nonempty and finite subset of alternatives $A \subseteq \mathcal{X}$, and any nonempty subset $B \subseteq A$, $f(\Pi(A)) \cap B = f(\Pi(B))$ whenever $f(\Pi(A)) \cap B \neq \phi$.

Intuitively, stability means that applying f under the assumption that the set of alternatives is A, and then reducing to its subset B, is the same as directly reducing to B and then applying f. This notion is related to classic axioms studied by Sen [19], specifically his *expansion* and *contraction* properties. In our setting, stability seems especially desirable, as our algorithm would potentially face decisions over many different subsets of alternatives, and the absence of stability may lead to glaringly inconsistent choices.

Theorem 3.12. *Let Π be the TM or PL process, and let f be the Borda count or Copeland SCC. Then the pair (Π, f) is stable.*

The definition of the Copeland SCC, and the proof of the theorem, are relegated to Appendix B. Among other things, the proof requires a stronger notion of SwD-efficiency, which, as we show, is satisfied by Borda count and Copeland, and potentially by other appealing SCCs.

4 Instantiation and Evaluation of Our Approach

In this section, we instantiate and evaluate our approach for ethical decision making, as outlined in Section 1. As proof of concept, we consider a specific permutation process, namely the TM process with a linear parameterization of the utility process parameters as a function of the alternative features.

4.1 Instantiation

Let the set of alternatives be given by $\mathcal{X} \subseteq \mathbb{R}^d$, i.e. each alternative is represented by a vector of d features. Furthermore, let N denote the total number of voters. Assume that the data-collection step (Step I) is complete, i.e., we have some pairwise comparisons for each voter.

Step II: Learning. For each voter, we learn a TM process using his pairwise comparisons to represent his preferences. We assume that the mode utility of an alternative x depends linearly on its features, i.e., $\mu_x = \boldsymbol{\beta}^\intercal x$. Note that we do not need an intercept term, since we care only about the relative ordering of utilities. Also note that the parameter $\boldsymbol{\beta} \in \mathbb{R}^d$ completely describes the TM process, and hence the parameters $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \cdots \boldsymbol{\beta}_N$ completely describe the models of all voters.

Next we provide a computationally efficient method for learning the parameter $\boldsymbol{\beta}$ for a particular voter. Let $(X_1, Z_1), (X_2, Z_2), \cdots, (X_n, Z_n)$ denote the pairwise comparison data of the voter. Specifically, the ordered pair (X_j, Z_j) denotes the j^{th} pair of alternatives compared by the voter, and the fact that the voter chose X_j over Z_j. We use maximum likelihood estimation to estimate $\boldsymbol{\beta}$. The log-likelihood function is

$$\mathcal{L}(\boldsymbol{\beta}) = \log\left[\prod_{j=1}^{n} P(X_j \succ Z_j; \boldsymbol{\beta})\right] = \sum_{j=1}^{n} \log P(U_{X_j} > U_{Z_j}; \boldsymbol{\beta}) = \sum_{j=1}^{n} \log \Phi\left(\boldsymbol{\beta}^\intercal (X_j - Z_j)\right),$$

where Φ is the cumulative distribution function of the standard normal distribution, and the last transition holds because $U_x \sim \mathcal{N}(\boldsymbol{\beta}^\intercal x, \frac{1}{2})$. Note that the standard normal CDF Φ is a log-concave function. This makes the log-likelihood concave in $\boldsymbol{\beta}$, hence we can maximize it efficiently.

Step III: Summarization. After completing Step II, we have N TM processes represented by the parameters $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \cdots \boldsymbol{\beta}_N$. In Step III, we bundle these individual models into a single permutation process $\hat{\Pi}$, which, in the current instantiation, is also a TM process with parameter $\hat{\boldsymbol{\beta}}$ (see Appendix E for a discussion of this point). We perform this step because we must be able to make decisions *fast*, in Step IV. For example, in the autonomous vehicle domain, the AI would only have a split second to make a decision in case of emergency; aggregating information from millions of voters *in real time* will not do. By contrast, Step III is performed offline, and provides the basis for fast aggregation.

Let $\Pi^{\boldsymbol{\beta}}$ denote the TM process with parameter $\boldsymbol{\beta}$. Given a finite subset of alternatives $A \subseteq \mathcal{X}$, the anonymous preference profile generated by the model of voter i is given by $\Pi^{\boldsymbol{\beta}_i}(A)$. Ideally, we would like the summary model to be such that the profile generated by it, $\hat{\Pi}(A)$, is as close as possible to $\Pi^*(A) = \frac{1}{N}\sum_{i=1}^{N} \Pi^{\boldsymbol{\beta}_i}(A)$, the mean profile obtained by giving equal importance to each voter. However, there does not appear to be a straightforward method to compute the "best" $\hat{\boldsymbol{\beta}}$, since the profiles generated by the TM processes do not have an explicit form. Hence, we use utilities as a proxy for the quality of $\hat{\boldsymbol{\beta}}$. Specifically, we find $\hat{\boldsymbol{\beta}}$ such that the summary model induces utilities that are as close as possible to the mean of the utilities induced by the per-voter models, i.e., we want $U_x^{\hat{\boldsymbol{\beta}}}$ to be as close as possible (in terms of KL divergence) to $\frac{1}{N}\sum_{i=1}^{N} U_x^{\boldsymbol{\beta}_\mathbf{i}}$ for each $x \in \mathcal{X}$, where $U_x^{\boldsymbol{\beta}}$ denotes the utility of x under TM process with parameter $\boldsymbol{\beta}$. This is achieved by taking $\hat{\boldsymbol{\beta}} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\beta}_i$, as shown by the following proposition (whose proof appears in Appendix C).

Proposition 4.1. *$\boldsymbol{\beta} = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{\beta}_i$ minimizes $KL\left(\frac{1}{N}\sum_{i=1}^{N} U_x^{\boldsymbol{\beta}_i} \middle\| U_x^{\boldsymbol{\beta}}\right)$ for any $x \in \mathcal{X}$.*

Step IV: Aggregation. As a result of Step III, we have exactly one (summary) TM process $\hat{\Pi}$ (with parameter $\hat{\boldsymbol{\beta}} = \bar{\boldsymbol{\beta}}$) to work with at runtime. Given a finite set of alternatives $A = \{x_1, x_2, \cdots, x_m\}$, we must aggregate the preferences represented by the anonymous preference profile $\hat{\Pi}(A)$. This is where the machinery of Section 3 comes in: We simply need to select an alternative that has maximum mode utility among $\hat{\boldsymbol{\beta}}^\intercal x_1, \hat{\boldsymbol{\beta}}^\intercal x_2, \cdots, \hat{\boldsymbol{\beta}}^\intercal x_m$. Such an alternative would be selected by any anonymous SCC that is monotonic and neutral, when applied to $\hat{\Pi}(A)$, as shown by Theorem 3.1. Moreover, this aggregation method is equivalent to applying the Borda count or Copeland SCCs (due to Lemmas B.5, B.7). Hence, we also have the desired stability property, as shown by Theorem 3.12.

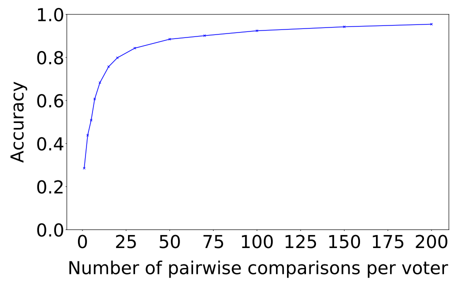


Figure 1: Accuracy of Step II

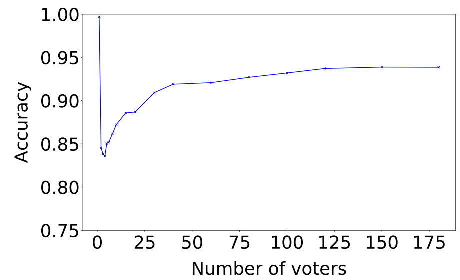


Figure 2: Accuracy of Step III

4.2 Evaluation

In the remainder of the section, we empirically evaluate Steps II and III of our approach, as described in Section 4.1, using synthetic data.

Setup. We represent the preferences of each voter using a TM process. Let $\boldsymbol{\beta}_i$ denote the true parameter corresponding to the model of voter i. We sample $\boldsymbol{\beta}_i$ from $\mathcal{N}(\mathbf{m}, I_d)$ (independently for each voter i), where each mean m_j is sampled independently from the uniform distribution $\mathcal{U}(-1, 1)$, and the number of features is $d = 10$.

In each instance (defined by a subset of alternatives A with $|A| = 5$), the desired winner is given by the application of Borda count to the mean of the profiles of the voters. In more detail, we compute the anonymous preference profile of each voter $\Pi^{\boldsymbol{\beta}_i}(A)$, and then take a mean across all the voters to obtain the desired profile $\frac{1}{N}\sum_{i=1}^{N}\Pi^{\boldsymbol{\beta}_i}(A)$. We then apply Borda count to this profile to obtain the winner. Note that, since we are dealing with TM processes, we cannot explicitly construct $\Pi^{\boldsymbol{\beta}_i}(A)$; we therefore estimate it by sampling rankings according to the TM process of voter i.

Evaluation of Step II (Learning). In practice, the algorithm would not have access to the true parameter $\boldsymbol{\beta}_i$ of voter i, but only to pairwise comparisons, from which we learn the parameters. Thus we compare the computation of the the winner (following the approach described above) using the true parameters, and using the learned parameters as in Step II. We report the accuracy as the fraction of instances, out of 100 test instances, in which the two outcomes match.

To generate each pairwise comparison of voter i, for each of $N = 20$ voters, we first sample two alternatives x_1 and x_2 independently from $\mathcal{N}(\mathbf{0}, I_d)$. Then, we sample their utilities U_{x_1} and U_{x_2} from $\mathcal{N}(\boldsymbol{\beta}_i^\intercal x_1, \frac{1}{2})$ and $\mathcal{N}(\boldsymbol{\beta}_i^\intercal x_2, \frac{1}{2})$, respectively. Of course, the voter prefers the alternative with higher sampled utility. Once we have the comparisons, we learn the parameter $\boldsymbol{\beta}_i$ by computing the MLE (as explained in Step II of Section 4.1). In our results, we vary the number of pairwise comparisons per voter and compute the accuracy to obtain the learning curve shown in Figure 1. Each datapoint in the graph is averaged over 50 runs. Observe that the accuracy quickly increases as the number of pairwise comparisons increases, and with just 30 pairwise comparisons we achieve an accuracy of 84.3%. With 100 pairwise comparisons, the accuracy is 92.4%.

Evaluation of Step III (Summarization). To evaluate Step III, we assume that we have access to the true parameters $\boldsymbol{\beta}_i$, and wish to determine the accuracy loss incurred in the summarization step, where we summarize the individual TM models into a single TM model. As described in Section 4, we compute $\bar{\boldsymbol{\beta}} = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{\beta}_i$, and, given a subset A (which again has cardinality 5), we aggregate using Step IV, since we now have just one TM process. For each instance, we contrast our computed winner with the desired winner as computed previously. We vary the number of voters and compute the accuracy to obtain Figure 2. The accuracies are averaged over 50 runs. Observe that the accuracy increases to 93.9% as the number of voters increases. In practice we expect to have access to thousands, even millions, of votes. We conclude that, surprisingly, the expected loss in accuracy due to summarization is quite small.

Robustness. Our results are robust to the choice of parameters, as we demonstrate in Appendix D.

5 Discussion

The design of intelligent machines that can make ethical decisions is, arguably, one of the hardest challenges in AI. We do believe that our approach takes a significant step towards the implementation of an algorithm that can make *credible* decisions on ethical dilemmas, especially in the autonomous vehicle domain (when all other options have failed). But this paper is clearly not the end-all solution.

Most important is the (primarily conceptual) challenge of extending our framework to incorporate ethical or legal principles — at least for simpler settings where they might be easier to specify. The significant advantage of having our approach in place is that these principles do not need to always lead to a decision, as we can fall back on the societal choice. This allows for a modular design where principles are incorporated over time, without compromising the ability to make a decision in every situation. We discuss additional technical challenges in Appendix E.

References

[1] H. Azari Soufiani, D. C. Parkes, and L. Xia. Random utility theory for social choice. In *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS)*, pages 126–134, 2012.

[2] H. Azari Soufiani, D. C. Parkes, and L. Xia. Computing parametric ranking models via rank-breaking. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, pages 360–368, 2014.

[3] J.-F. Bonnefon, A. Shariff, and I. Rahwan. The social dilemma of autonomous vehicles. *Science*, 352 (6293):1573–1576, 2016.

[4] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

[5] I. Caragiannis, A. D. Procaccia, and N. Shah. When do noisy votes reveal the truth? *ACM Transactions on Economics and Computation*, 4(3): article 15, 2016.

[6] F. Caron and Y. W. Teh. Bayesian nonparametric models for ranked data. In *Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS)*, pages 1529–1537, 2012.

[7] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. S. Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS)*, pages 214–226, 2012.

[8] J. Greene, F. Rossi, J. Tasioulas, K. B. Venable, and B. Williams. Embedding ethical principles in collective decision support systems. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pages 4147–4151, 2016.

[9] J. Guiver and E. Snelson. Bayesian inference for Plackett-Luce ranking models. In *Proceedings of the 26th International Conference on Machine Learning (ICML)*, pages 377–384, 2009.

[10] J. Jarvis Thomson. The trolley problem. *The Yale Law Journal*, 94(6):1395–1415, 1985.

[11] R. D. Luce. *Individual Choice Behavior: A Theoretical Analysis*. Wiley, 1959.

[12] L. Maystre and M. Grossglauser. Fast and accurate inference of Plackett-Luce models. In *Proceedings of the 29th Annual Conference on Neural Information Processing Systems (NIPS)*, pages 172–180, 2015.

[13] F. Mosteller. Remarks on the method of paired comparisons: I. the least squares solution assuming equal standard deviations and equal correlations. *Psychometrika*, 16(1):3–9, 1951.

[14] H. Moulin. *The Strategy of Social Choice*, volume 18 of *Advanced Textbooks in Economics*. North-Holland, 1983.

[15] R. Plackett. The analysis of permutations. *Applied Statistics*, 24:193–202, 1975.

[16] A. Prasad, H. H. Pareek, and P. Ravikumar. Distributional rank aggregation, and an axiomatic analysis. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, pages 2104–2112, 2015.

[17] A. Rajkumar and S. Agarwal. A statistical convergence perspective of algorithms for rank aggregation from pairwise data. In *Proceedings of the 31st International Conference on Machine Learning (ICML)*, pages 118–126, 2014.

[18] J. Rawls. *A Theory of Justice*. Harvard University Press, 1971.

[19] A. K. Sen. Choice functions and revealed preference. *Review of Economic Studies*, 38(3):307–317, 1971.

[20] L. L. Thurstone. A law of comparative judgement. *Psychological Review*, 34:273–286, 1927.

[21] W. Wallach and C. Allen. *Moral Machines: Teaching Robots Right from Wrong*. Oxford University Press, 2008.

[22] B. Williams. *Ethics and the Limits of Philosophy*. Harvard University Press, 1986.

A Proof of Theorem 3.1 and Omitted Lemmas

A.1 Proof of Lemma 3.4

Let f be an anonymous SCC that satisfies monotonicity and neutrality. Let π be an arbitrary anonymous preference profile, and let a, b be two arbitrary alternatives such that $a \rhd_\pi b$. Now, suppose for the sake of contradiction that $b \in f(\pi)$ but $a \notin f(\pi)$.

Consider an arbitrary ranking σ with $a \succ_\sigma b$. Since $a \rhd_\pi b$, $\pi(\sigma) \geq \pi(\sigma^{ab})$. In other words, we have an excess weight of $\pi(\sigma) - \pi(\sigma^{ab})$ on σ. For this excess weight of σ, move b upwards and place it just below a. By monotonicity, b still wins and a still loses in this modified profile. We repeat this procedure for every such σ (i.e. for its excess weight, move b upwards, until it is placed below a). In the resulting profile, a still loses. Now, for each of the modified rankings, move a down to where b originally was. By monotonicity, a still loses in the resulting profile π', i.e., $a \notin f(\pi')$.

On the other hand, this procedure is equivalent to shifting the excess weight $\pi(\sigma) - \pi(\sigma^{ab})$ from σ to σ^{ab} (for each σ with $a \succ_\sigma b$). Hence, the profile π' we end up with is such that $\pi'(\sigma) = \pi(\sigma^{ab})$ and $\pi'(\sigma^{ab}) = \pi(\sigma)$, i.e. the new profile is the original profile with a and b swapped. Therefore, by neutrality, it must be the case that $a \in f(\pi')$. This contradicts our conclusion that $a \notin f(\pi')$, thus completing the proof. $\square$

A.2 Proof of Theorem 3.7

Let f, Π, and A as in the theorem statement. Since Π is SwD-compatible, $\rhd_\Pi$ is a total preorder on $\mathcal{X}$. In turn, the relation $\rhd_\Pi$ restricted to A is a total preorder on A. Therefore, there is $a \in A$ such that $a \rhd_\Pi b$ for all $b \in A$.

Suppose for the sake of contradiction that $a \notin f(\Pi(A))$, and let $b \in A \setminus \{a\}$. Then it holds that $a \rhd_\Pi b$. In particular, $a \rhd_{\Pi(A)} b$. But, because f is SwD-efficient and $a \notin f(\Pi(A))$, we have that $b \notin f(\Pi(A))$. This is true for every $b \in A$, leading to $f(\Pi(A)) = \phi$, which contradicts the definition of an SCC. $\square$

A.3 Proof of Lemma 3.9

Let a and b be two alternatives such that a dominates b in U. In addition, let A be a finite set of alternatives containing a and b, let π denote the anonymous preference profile $\Pi(A)$, and let $m = |A|$. Consider an arbitrary ranking σ such that $a \succ_\sigma b$. Now, let $x_\ell = \sigma^{-1}(\ell)$ denote the alternative in position ℓ of σ, and let $i = \sigma(a)$, $j = \sigma(b)$, i.e.,

$$x_1 \succ_\sigma x_2 \cdots \succ_\sigma x_i (= a) \succ_\sigma \cdots \succ_\sigma x_j (= b) \succ_\sigma \cdots \succ_\sigma x_m.$$

Then,

$$\begin{aligned}\pi(\sigma) &= P(U_{x_1} > U_{x_2} > \cdots > U_{x_i} > \cdots > U_{x_j} > \cdots > U_{x_m})\\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{u_1} \cdots \int_{-\infty}^{u_{i-1}} \cdots \int_{-\infty}^{u_{j-1}} \cdots \int_{-\infty}^{u_{m-1}} p(u_1, u_2, \cdots, u_i, \cdots u_j, \cdots, u_m) du_m \cdots du_1.\end{aligned}$$

In this integral, because of the limits, we always have $u_i \geq u_j$. Moreover, since $x_i = a$ dominates $x_j = b$ in U, we have

$$\pi(\sigma) \geq \int_{-\infty}^{\infty} \int_{-\infty}^{u_1} \cdots \int_{-\infty}^{u_{i-1}} \cdots \int_{-\infty}^{u_{j-1}} \cdots \int_{-\infty}^{u_{m-1}} p(u_1, u_2, \cdots, u_j, \cdots u_i, \cdots, u_m) du_m \cdots du_1.$$

The right-hand side of this equation is exactly $\pi(\sigma^{ab})$. Hence, we have $\pi(\sigma) \geq \pi(\sigma^{ab})$. It follows that $a \rhd_\pi b$, i.e., $a \rhd_{\Pi(A)} b$. Also, this is true for any finite A containing a and b. We conclude that $a \rhd_\Pi b$. $\square$

A.4 Proof of Lemma 3.10

We establish the property separately for the TM and PL processes.

TM process. Let a and b be two alternatives such that $\mu_a \geq \mu_b$. Since we are dealing with a TM process, $U_a \sim \mathcal{N}(\mu_a, \frac{1}{2})$ and $U_b \sim \mathcal{N}(\mu_b, \frac{1}{2})$. Let A be any finite set of alternatives containing a

and b. Since utilities are sampled independently in a TM process, the difference between the two sides of Equation (1) is that the left-hand side has $p_{U_a}(u_1)p_{U_b}(u_2)$, while the right-hand side has $p_{U_a}(u_2)p_{U_b}(u_1)$. It holds that

$$\begin{aligned} p_{U_a}(u_1)p_{U_b}(u_2) &= \frac{1}{\sqrt{\pi}} \exp\left(-(u_1-\mu_a)^2\right) \frac{1}{\sqrt{\pi}} \exp\left(-(u_2-\mu_b)^2\right). \\ &= \frac{1}{\pi} \exp\left(-u_1^2-\mu_a^2-u_2^2-\mu_b^2+2u_1\mu_a+2u_2\mu_b\right). \end{aligned} \tag{2}$$

We have $u_1 \geq u_2$ and $\mu_a \geq \mu_b$. Therefore,

$$\begin{aligned} u_1\mu_a + u_2\mu_b &= u_1\mu_b + u_1(\mu_a-\mu_b) + u_2\mu_b \\ &\geq u_1\mu_b + u_2(\mu_a-\mu_b) + u_2\mu_b \\ &= u_1\mu_b + u_2\mu_a \end{aligned}$$

Substituting this into Equation (2), we obtain

$$\begin{aligned} p_{U_a}(u_1)p_{U_b}(u_2) &\geq \frac{1}{\pi} \exp\left(-u_1^2-\mu_a^2-u_2^2-\mu_b^2+2u_1\mu_b+2u_2\mu_a\right) \\ &= \frac{1}{\pi} \exp\left(-(u_2-\mu_a)^2-(u_1-\mu_b)^2\right) \\ &= p_{U_a}(u_2)p_{U_b}(u_1) \end{aligned}$$

It follows that Equation (1) holds true. Hence, a dominates b in the corresponding utility process.

To show the other direction, let a and b be such that $\mu_a < \mu_b$. If we choose u_1, u_2 such that $u_1 > u_2$, using a very similar approach as above, we get $p_{U_a}(u_1)p_{U_b}(u_2) < p_{U_a}(u_2)p_{U_b}(u_1)$. And so, a does not dominate b in the corresponding utility process. $\square$

PL process. Let a and b be two alternatives such that $\mu_a \geq \mu_b$. Since we are dealing with a PL process, $U_a \sim \mathcal{G}(\mu_a, \gamma)$ and $U_b \sim \mathcal{G}(\mu_b, \gamma)$. Let A be any finite set of alternatives containing a and b. Since utilities are sampled independently in a PL process, the difference between the two sides of Equation (1) is that the left-hand side has $p_{U_a}(u_1)p_{U_b}(u_2)$, while the right-hand side has $p_{U_a}(u_2)p_{U_b}(u_1)$. It holds that

$$\begin{aligned} p_{U_a}(u_1)p_{U_b}(u_2) &= \frac{1}{\gamma} \exp\left(-\frac{u_1-\mu_a}{\gamma} - e^{-\frac{u_1-\mu_a}{\gamma}}\right) \frac{1}{\gamma} \exp\left(-\frac{u_2-\mu_b}{\gamma} - e^{-\frac{u_2-\mu_b}{\gamma}}\right) \\ &= \frac{1}{\gamma^2} \exp\left(-\frac{u_1-\mu_a}{\gamma} - e^{-\frac{u_1-\mu_a}{\gamma}} - \frac{u_2-\mu_b}{\gamma} - e^{-\frac{u_2-\mu_b}{\gamma}}\right) \\ &= \frac{1}{\gamma^2} \exp\left(-\frac{u_1-\mu_a+u_2-\mu_b}{\gamma} - \left(e^{-\frac{u_1}{\gamma}} e^{\frac{\mu_a}{\gamma}} + e^{-\frac{u_2}{\gamma}} e^{\frac{\mu_b}{\gamma}}\right)\right). \end{aligned} \tag{3}$$

We also know that $e^{-\frac{u_2}{\gamma}} \geq e^{-\frac{u_1}{\gamma}}$ and $e^{\frac{\mu_a}{\gamma}} \geq e^{\frac{\mu_b}{\gamma}}$. Similar to the proof for the TM process, we have

$$e^{-\frac{u_2}{\gamma}} e^{\frac{\mu_a}{\gamma}} + e^{-\frac{u_1}{\gamma}} e^{\frac{\mu_b}{\gamma}} \geq e^{-\frac{u_1}{\gamma}} e^{\frac{\mu_a}{\gamma}} + e^{-\frac{u_2}{\gamma}} e^{\frac{\mu_b}{\gamma}}.$$

Substituting this into Equation (3), we obtain

$$\begin{aligned} p_{U_a}(u_1)p_{U_b}(u_2) &\geq \frac{1}{\gamma^2} \exp\left(-\frac{u_1-\mu_a+u_2-\mu_b}{\gamma} - \left(e^{-\frac{u_2}{\gamma}} e^{\frac{\mu_a}{\gamma}} + e^{-\frac{u_1}{\gamma}} e^{\frac{\mu_b}{\gamma}}\right)\right) \\ &= \frac{1}{\gamma} \exp\left(-\frac{u_2-\mu_a}{\gamma} - e^{-\frac{u_2-\mu_a}{\gamma}}\right) \frac{1}{\gamma} \exp\left\{-\frac{u_1-\mu_b}{\gamma} - e^{-\frac{u_1-\mu_b}{\gamma}}\right\} \\ &= p_{U_a}(u_2)p_{U_b}(u_1) \end{aligned}$$

It follows that Equation (1) holds true. Hence, a dominates b in the corresponding utility process.

To show the other direction, let a and b be such that $\mu_a < \mu_b$. If we choose u_1, u_2 such that $u_1 > u_2$, using a very similar approach as above, we get $p_{U_a}(u_1)p_{U_b}(u_2) < p_{U_a}(u_2)p_{U_b}(u_1)$. And so, a does not dominate b in the corresponding utility process. $\square$

A.5 Proof of Theorem 3.1

By Lemma 3.4, the anonymous SCC f is SwD-efficient. Lemmas 3.9 and 3.10 directly imply that when Π is the TM or PL process, $\rhd_\Pi$ is indeed a total preorder. In particular, $a \rhd_\Pi b$ if $\mu_a \geq \mu_b$. So, an alternative a in A with maximum mode utility satisfies $a \rhd_\Pi b$ for all $b \in A$. By Theorem 3.7, if $a \in A$ is such that $a \rhd_\Pi b$ for all $b \in A$, then $a \in f(\Pi(A))$; the statement of the theorem follows. $\square$

B More on Stability and Proof of Theorem 3.12

Before proving Theorem 3.12, we examine some examples that illustrate stability (or the lack thereof).

Example B.1. Let f be the Borda count SCC, and let the set of alternatives be $\mathcal{X} = \{u, v, w, x, y\}$. Also, let Π be a consistent permutation process, which, given all the alternatives, gives a uniform distribution on the two rankings $(x \succ u \succ v \succ y \succ w)$ and $(y \succ w \succ x \succ u \succ v)$. The outcome of applying f on this profile is $\{x\}$ (since x has the strictly highest Borda score). But, the outcome of applying f on the profile $\Pi(\{w, x, y\})$ is $\{y\}$ (since y now has the strictly highest Borda score). Hence, $f(\Pi(\{u, v, w, x, y\})) \cap \{w, x, y\} \neq f(\Pi(w, x, y))$, even though the left-hand side is nonempty. We conclude that the tuple (Π, f) does not satisfy stability.

For the next example (and the statement of Theorem 3.12), we need to define the *Copeland* SCC. For an anonymous preference profile π over A, we say that $a \in A$ beats $b \in A$ in a pairwise election if

$$\sum_{\sigma \in \mathcal{S}_A :\, a \succ_\sigma b} \pi(\sigma) > \frac{1}{2}.$$

The *Copeland score* of an alternative is the number of other alternatives it beats in pairwise elections; the Copeland SCC selects all alternatives that maximize the Copeland score.

Example B.2. Consider the permutation process of Example B.1, and let f be the Copeland SCC. Once again, it holds that $f(\Pi(u, v, w, x, y)) = \{x\}$ and $f(\Pi(w, x, y)) = \{y\}$. Hence the pair (Π, f) is not stable.

Now, in the spirit of Theorem 3.7, let us see whether the pair (Π, f) satisfies stability when f is an SwD-efficient anonymous SCC, and Π is an SwD-compatible permutation process. Example B.3 constructs such a Π that is not stable with respect to the plurality SCC (even though plurality is SwD-efficient).

Example B.3. Let f be the plurality SCC and the set of alternatives be $\mathcal{X} = \{a, b, c\}$. Also, let Π be the consistent permutation process, which given all alternatives, gives the following profile: 0.35 weight on $(a \succ b \succ c)$, 0.35 weight on $(b \succ a \succ c)$, 0.1 weight on $(c \succ a \succ b)$, 0.1 weight on $(a \succ c \succ b)$ and 0.1 weight on $(b \succ c \succ a)$. All the swap-dominance relations in this permutation process are: $a \rhd_\Pi b$, $b \rhd_\Pi c$ and $a \rhd_\Pi c$. Hence, $\rhd_\Pi$ is a total preorder on $\mathcal{X}$, and Π is SwD-compatible. Now, for this permutation process Π and the plurality SCC f, we have: $f(\Pi(\{a, b, c\})) = \{a, b\}$ and $f(\Pi(\{a, b\})) = \{a\}$. Therefore, (Π, f) is not stable.

This happens because Plurality is not *strongly* SwD-efficient, as defined below (Example B.3 even shows why plurality violates this property).

Definition B.4. An anonymous SCC f is said to be *strongly SwD-efficient* if for every anonymous preference profile π over A, and any two alternatives $a, b \in A$ such that $a \rhd_\pi b$,

1. If $b \not\rhd_\pi a$, then $b \notin f(\pi)$.
2. If $b \rhd_\pi a$, then $b \in f(\pi) \Leftrightarrow a \in f(\pi)$.

It is clear that any strongly SwD-efficient SCC is also SwD-efficient.

Lemma B.5. *The Borda count and Copeland SCCs are strongly SwD-efficient.*

Proof. Let π be an arbitrary anonymous preference profile over alternatives A, and let $a, b \in A$ such that $a \rhd_\pi b$. This means that for all $\sigma \in \mathcal{S}_A$ with $a \succ_\sigma b$, we have $\pi(\sigma) \geq \pi(\sigma^{ab})$. We will examine the two conditions (of Definition B.4) separately.

Case 1: $b \not\triangleright_\pi a$
This means that there exists a ranking $\sigma_* \in \mathcal{S}_A$ with $b \succ_{\sigma_*} a$ such that $\pi(\sigma_*) < \pi(\sigma_*^{ab})$. Below we analyze each of the SCCs mentioned in the theorem.

Borda count. $\mathcal{S}_A$ can be partitioned into pairs of the form (σ, σ^{ab}), where σ is such that $a \succ_\sigma b$. We reason about how each pair contributes to the Borda scores of a and b. Consider an arbitrary pair (σ, σ^{ab}) with $a \succ_\sigma b$. The score contributed by σ to a is $(m - \sigma(a))\pi(\sigma)$, and the score contributed to b is $(m - \sigma(b))\pi(\sigma)$. That is, it gives an excess score of $(\sigma(b) - \sigma(a))\pi(\sigma)$ to a. Similarly, the score of a contributed by σ^{ab} is $(m - \sigma^{ab}(a))\pi(\sigma^{ab}) = (m - \sigma(b))\pi(\sigma^{ab})$, and the score contributed to b is $(m - \sigma(a))\pi(\sigma^{ab})$. So, b gets an excess score of $(\sigma(b) - \sigma(a))\pi(\sigma^{ab})$ from σ^{ab}. Combining these observations, the pair (σ, σ^{ab}) gives a an excess score of $(\sigma(b) - \sigma(a))(\pi(\sigma) - \pi(\sigma^{ab}))$, which is at least 0. Since this is true for every pair (σ, σ^{ab}), a has Borda score that is at least as high as that of b. Furthermore, the pair $(\sigma_*^{ab}, \sigma_*)$ is such that $\pi(\sigma_*^{ab}) - \pi(\sigma_*) > 0$, so, this pair gives a an excess score that is strictly positive. We conclude that a has strictly higher Borda score than b, hence b is not selected by Borda count.

Copeland. Let $c \in A \setminus \{a, b\}$. In a pairwise election between b and c, the total weight of rankings that place b over c is

$$\sum_{\sigma \in \mathcal{S}_A :\, b \succ_\sigma c} \pi(\sigma) = \sum_{\sigma \in \mathcal{S}_A :\, (b \succ_\sigma c) \wedge (a \succ_\sigma c)} \pi(\sigma) + \sum_{\sigma \in \mathcal{S}_A :\, (b \succ_\sigma c) \wedge (c \succ_\sigma a)} \pi(\sigma).$$

For the rankings in the second summation (on the right-hand side), we have $b \succ_\sigma a$ by transitivity. Hence, $\pi(\sigma) \le \pi(\sigma^{ab})$ for such rankings. Therefore,

$$\begin{aligned}
\sum_{\sigma \in \mathcal{S}_A :\, b \succ_\sigma c} \pi(\sigma) &\le \sum_{\sigma \in \mathcal{S}_A :\, (b \succ_\sigma c) \wedge (a \succ_\sigma c)} \pi(\sigma) + \sum_{\sigma \in \mathcal{S}_A :\, (b \succ_\sigma c) \wedge (c \succ_\sigma a)} \pi(\sigma^{ab}) \\
&= \sum_{\sigma \in \mathcal{S}_A :\, (b \succ_\sigma c) \wedge (a \succ_\sigma c)} \pi(\sigma) + \sum_{\sigma' \in \mathcal{S}_A :\, (a \succ_{\sigma'} c) \wedge (c \succ_{\sigma'} b)} \pi(\sigma') = \sum_{\sigma \in \mathcal{S}_A :\, a \succ_\sigma c} \pi(\sigma).
\end{aligned}$$

In summary, we have

$$\sum_{\sigma \in \mathcal{S}_A :\, b \succ_\sigma c} \pi(\sigma) \le \sum_{\sigma \in \mathcal{S}_A :\, a \succ_\sigma c} \pi(\sigma).$$

Hence, if b beats c in a pairwise competition, then so does a. Therefore, the Copeland score of a (due to all alternatives other than a and b) is at least as high as that of b. Further, in a pairwise competition between a and b, the weight of rankings that position a above b is $\sum_{\sigma \in \mathcal{S}_A :\, a \succ_\sigma b} \pi(\sigma)$ and the weight of those that prefer b over a is $\sum_{\sigma \in \mathcal{S}_A :\, b \succ_\sigma a} \pi(\sigma)$. But, because $\pi(\sigma) \ge \pi(\sigma^{ab})$ for any σ with $a \succ_\sigma b$, and $\pi(\sigma_*^{ab}) > \pi(\sigma_*)$, a beats b. Therefore, a has a strictly higher Copeland score than b, and b is not selected by Copeland.

Case 2: $b \triangleright_\pi a$
In this case, $a \triangleright_\pi b$ and $b \triangleright_\pi a$. This means that for all $\sigma \in \mathcal{S}_A$, we have $\pi(\sigma) = \pi(\sigma^{ab})$. In other words, $\tau(\pi) = \pi$, where τ is the permutation that swaps a and b. Both Borda count and Copeland are neutral SCCs. So, we have $\tau(f(\pi)) = f(\tau(\pi))$, which is in turn equal to $f(\pi)$. Hence, a is selected if and only if b is selected.

We conclude that both conditions of Definition B.4 are satisfied by Borda count and Copeland. $\square$

Lemma B.6. *Let Π be a consistent permutation process that is SwD-compatible. Then, for any finite subset of alternatives $A \subseteq \mathcal{X}$, $\left(\triangleright_{\Pi(A)}\right) = (\triangleright_\Pi|_A)$.*

In words, as long as Π is consistent and SwD-compatible, marginalizing out some alternatives from a profile does not remove or add any swap-dominance relations.

Proof of Lemma B.6. We first show that for any $B \subseteq A \subseteq \mathcal{X}$, $\left(\triangleright_{\Pi(A)}|_B\right) = \left(\triangleright_{\Pi(B)}\right)$.

Let $a, b \in B$ such that $a \triangleright_{\Pi(A)} b$. Now, let $\sigma \in \mathcal{S}_B$ be an arbitrary ranking such that $a \succ_\sigma b$. Also, let π_B denote $\Pi(B)$ and π_A denote $\Pi(A)$. Then, since Π is consistent,

$$\pi_B(\sigma) = \sum_{\sigma_2 \in \mathcal{S}_A :\, \sigma_2|_B = \sigma} \pi_A(\sigma_2).$$

Now, for $\sigma_2 \in \mathcal{S}_A$ such that $\sigma_2|_B = \sigma$, we have $a \succ_{\sigma_2} b$ and therefore $\pi_A(\sigma_2) \geq \pi_A(\sigma_2^{ab})$ (because $a \rhd_{\Pi(A)} b$). It follows that

$$\pi_B(\sigma) = \sum_{\sigma_2 \in \mathcal{S}_A : \sigma_2|_B = \sigma} \pi_A(\sigma_2) \geq \sum_{\sigma_2 \in \mathcal{S}_A : \sigma_2|_B = \sigma} \pi_A(\sigma_2^{ab}) = \sum_{\sigma_2' \in \mathcal{S}_A : \sigma_2'|_B = \sigma^{ab}} \pi_A(\sigma_2') = \pi_B(\sigma^{ab}).$$

Therefore, $a \rhd_{\Pi(B)} b$, that is, $\left(\rhd_{\Pi(A)}|_B\right) \subseteq \left(\rhd_{\Pi(B)}\right)$.

Next we show that $\left(\rhd_{\Pi(B)}\right) \subseteq \left(\rhd_{\Pi(A)}|_B\right)$. Let $a, b \in B$ such that $a \rhd_{\Pi(B)} b$. Suppose for the sake of contradiction that $a \not\rhd_{\Pi(A)} b$. This implies that $a \not\rhd_{\Pi} b$. However, $\rhd_{\Pi}$ is a total preorder because Π is SwD-compatible (by definition). It follows that $b \rhd_{\Pi} a$, and, in particular, $b \rhd_{\Pi(A)} a$ and $b \rhd_{\Pi(B)} a$.

As before, let π_A denote $\Pi(A)$ and π_B denote $\Pi(B)$. Because $a \not\rhd_{\Pi(A)} b$, there exists $\sigma_* \in \mathcal{S}_A$ with $a \succ_{\sigma_*} b$ such that $\pi_A(\sigma_*) < \pi_A(\sigma_*^{ab})$. Moreover, because $a \rhd_{\Pi(B)} b$ and $b \rhd_{\Pi(B)} a$, it holds that $\pi_B(\sigma_*|_B) = \pi_B\left((\sigma_*|_B)^{ab}\right)$. The consistency of Π then implies that

$$\sum_{\sigma_1 \in \mathcal{S}_A :\, \sigma_1|_B = \sigma_*|_B} \pi_A(\sigma_1) = \sum_{\sigma_2 \in \mathcal{S}_A :\, \sigma_2|_B = (\sigma_*|_B)^{ab}} \pi_A(\sigma_2). \tag{4}$$

Note $\sigma_1 = \sigma_*$ is a ranking that appears on the left-hand side of Equation (4), and $\sigma_2 = \sigma_*^{ab}$ is a ranking that appears on the right-hand side. Furthermore, we know that $\pi_A(\sigma_*) < \pi_A(\sigma_*^{ab})$. It follows that there exists $\sigma' \in \mathcal{S}_A$ with $\sigma'|_B = \sigma_*|_B$ such that $\pi_A(\sigma') > \pi_A\left((\sigma')^{ab}\right)$. Also, since $\sigma'|_B = \sigma_*|_B$, it holds that $a \succ_{\sigma'} b$. We conclude that it cannot be the case that $b \rhd_{\Pi(A)} a$, leading to a contradiction. Therefore, if $a \rhd_{\Pi(B)} b$, then $a \rhd_{\Pi(A)} b$, i.e., $\left(\rhd_{\Pi(B)}\right) \subseteq \left(\rhd_{\Pi(A)}|_B\right)$.

We next prove the lemma itself, i.e., that $\left(\rhd_{\Pi(A)}\right) = \left(\rhd_{\Pi}|_A\right)$. Firstly, for $a, b \in A$, if $a \rhd_{\Pi} b$, then $a \rhd_{\Pi(A)} b$ by definition. So, we easily get $\left(\rhd_{\Pi}|_A\right) \subseteq \left(\rhd_{\Pi(A)}\right)$.

In the other direction, let $a, b \in A$ such that $a \rhd_{\Pi(A)} b$. Let C be an arbitrary set of alternatives containing a and b. From what we have shown above, we have $\left(\rhd_{\Pi(A)}|_{\{a,b\}}\right) = \left(\rhd_{\Pi(\{a,b\})}\right)$. Also, $\left(\rhd_{\Pi(C)}|_{\{a,b\}}\right) = \left(\rhd_{\Pi(\{a,b\})}\right)$. This gives us $\left(\rhd_{\Pi(A)}|_{\{a,b\}}\right) = \left(\rhd_{\Pi(C)}|_{\{a,b\}}\right)$. Hence, $a \rhd_{\Pi(C)} b$, and this is true for every such subset C. We conclude that $a \rhd_{\Pi} b$, that is, $\left(\rhd_{\Pi(A)}\right) \subseteq \left(\rhd_{\Pi}|_A\right)$. ☐

Lemma B.7. *Let f be a strongly SwD-efficient anonymous SCC, and let Π be a consistent permutation process that is SwD-compatible. Then for any finite subset of alternatives A, $f(\Pi(A)) = \{a \in A : a \rhd_{\Pi} b \text{ for all } b \in A\}$.*

Proof. Let A be an arbitrary finite subset of alternatives. Since strong SwD-efficiency implies SwD-efficiency, Theorem 3.7 gives us

$$f(\Pi(A)) \supseteq \{a \in A : a \rhd_{\Pi} b \text{ for all } b \in A\}.$$

In the other direction, let $a \in f(\Pi(A))$. Suppose for the sake of contradiction that there exists $b \in A$ such that $a \not\rhd_{\Pi} b$. Since $\rhd_{\Pi}$ is a total preorder, it follows that $b \rhd_{\Pi} a$. By Lemma B.6, it holds that $\left(\rhd_{\Pi(A)}\right) = \left(\rhd_{\Pi}|_A\right)$, and therefore $a \not\rhd_{\Pi(A)} b$ and $b \rhd_{\Pi(A)} a$. But, since f is strongly SwD-efficient, it follows that $a \notin f(\Pi(A))$, which contradicts our assumption. Hence,

$$f(\Pi(A)) \subseteq \{a \in A : a \rhd_{\Pi} b \text{ for all } b \in A\},$$

and we have the desired result. ☐

Theorem B.8. *Let Π be a consistent permutation process that is SwD-compatible, and let f be a strongly SwD-efficient anonymous SCC. Then the pair (Π, f) is stable.*

Proof. Consider an arbitrary subset of alternatives A, and let $B \subseteq A$. By Lemma B.7, $f(\Pi(A)) = \{a \in A : a \rhd_{\Pi} b \text{ for all } b \in A\}$, and similarly for B. Suppose $f(\Pi(A)) \cap B \neq \phi$, and let $a \in f(\Pi(A)) \cap B$, i.e. $a \in f(\Pi(A))$ and $a \in B$. This means that $a \rhd_{\Pi} b$ for all $b \in A$, and, therefore $a \rhd_{\Pi} b$ for all $b \in B$. We conclude that $a \in f(\Pi(B))$, and hence $f(\Pi(A)) \cap B \subseteq f(\Pi(B))$.

In the other direction, let $a \in f(\Pi(B))$. This means that $a \rhd_{\Pi} b$ for all $b \in B$. Suppose for the sake of contradiction that $a \notin f(\Pi(A))$. This means that there exists $c \in A$ such that $a \not\rhd_{\Pi} c$. We assumed

$f(\Pi(A)) \cap B \neq \phi$, so let $d \in f(\Pi(A)) \cap B$. Then, $d \rhd_\Pi c$. In summary, we have $d \rhd_\Pi c$ and $a \not\rhd_\Pi c$, which together imply that $a \not\rhd_\Pi d$ (otherwise, it would violate transitivity). But $d \in B$, leading to $a \notin f(\Pi(B))$, which contradicts the assumption. Therefore, indeed $a \in f(\Pi(A))$, and it holds that $f(\Pi(B)) \subseteq f(\Pi(A)) \cap B$, as long as $f(\Pi(A)) \cap B \neq \phi$. □

We are now ready to prove Theorem 3.12.

Proof of Theorem 3.12. From Lemma B.5, Borda count and Copeland are strongly SwD-efficient. Lemmas 3.9 and 3.10 imply that when Π is the TM or PL process, $\rhd_\Pi$ is a total preorder. In particular, $a \rhd_\Pi b$ if $\mu_a \geq \mu_b$. Hence, Π is SwD-compatible. Therefore, by Theorem B.8, the pair (Π, f) is stable. □

C Proof of Proposition 4.1

Let $\bar{\boldsymbol{\beta}} = \frac{1}{N}\sum_{i=1}^N \boldsymbol{\beta}_i$. We know that $U_x^{\boldsymbol{\beta}}$ denotes the utility of x under the TM process with parameter $\boldsymbol{\beta}$. So, $U_x^{\boldsymbol{\beta}} \sim \mathcal{N}(\boldsymbol{\beta}^\intercal x, \frac{1}{2})$. Let its density be given by $q_{x,\boldsymbol{\beta}}(\cdot)$. Also, $U_x^{\boldsymbol{\beta}_i} \sim \mathcal{N}(\boldsymbol{\beta}_i^\intercal x, \frac{1}{2})$. Hence, $\frac{1}{N}\sum_{i=1}^N U_x^{\boldsymbol{\beta}_i} \sim \mathcal{N}(\bar{\boldsymbol{\beta}}^\intercal x, \frac{1}{2N})$. Let its density function be denoted by $p_x(\cdot)$. Then

$$KL(p_x \| q_{x,\boldsymbol{\beta}}) = \int p_x(t) \log p_x(t) dt - \int p_x(t) \log q_{x,\boldsymbol{\beta}}(t) dt.$$

Since the first term does not depend on $\boldsymbol{\beta}$, let us examine the second term:

$$\begin{aligned}
-\int p_x(t) \log q_{x,\boldsymbol{\beta}}(t) dt &= -\int p_x(t) \log\left(\frac{1}{\sqrt{\pi}} \exp\left(-(t-\boldsymbol{\beta}^\intercal x)^2\right)\right) dt \\
&= -\int p_x(t) \left[-\frac{1}{2}\log(\pi) - (t-\boldsymbol{\beta}^\intercal x)^2\right] dt \\
&= \frac{1}{2}\log(\pi)\left(\int p_x(t)dt\right) + \int p_x(t)\left(t^2 + (\boldsymbol{\beta}^\intercal x)^2 - 2t\boldsymbol{\beta}^\intercal x\right) dt \\
&= \frac{1}{2}\log(\pi) + \left(\int t^2 p_x(t)dt + (\boldsymbol{\beta}^\intercal x)^2 \int p_x(t)dt - 2\boldsymbol{\beta}^\intercal x \int t p_x(t)dt\right) \\
&= \frac{1}{2}\log(\pi) + \left(\left(\frac{1}{2N} + (\bar{\boldsymbol{\beta}}^\intercal x)^2\right) + (\boldsymbol{\beta}^\intercal x)^2 - 2\boldsymbol{\beta}^\intercal x(\bar{\boldsymbol{\beta}}^\intercal x)\right) \\
&= \frac{1}{2}\log(\pi) + \frac{1}{2N} + \left(\bar{\boldsymbol{\beta}}^\intercal x - \boldsymbol{\beta}^\intercal x\right)^2.
\end{aligned}$$

This term is minimized at $\boldsymbol{\beta} = \bar{\boldsymbol{\beta}}$ for any x, and therefore $KL(\frac{1}{N}\sum_{i=1}^N U_x^{\boldsymbol{\beta}_i} \| U_x^{\boldsymbol{\beta}})$ is minimized at that value as well. □

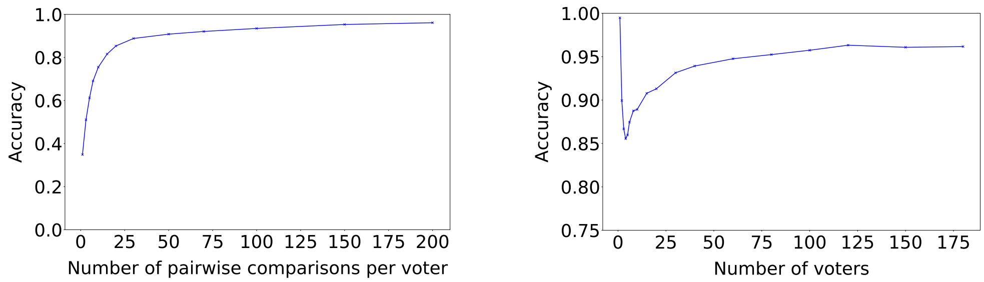
D Robustness of the Empirical Results

In Section 4.2, we presented experiments with the following parameters: each instance has 5 alternatives, the number of features is $d = 10$, and, in Step II, we let number of voters be $N = 20$. In this appendix, to demonstrate the robustness of both steps, we show experimental results for different values of these parameters (keeping everything else fixed).

D.1 Number of Alternatives

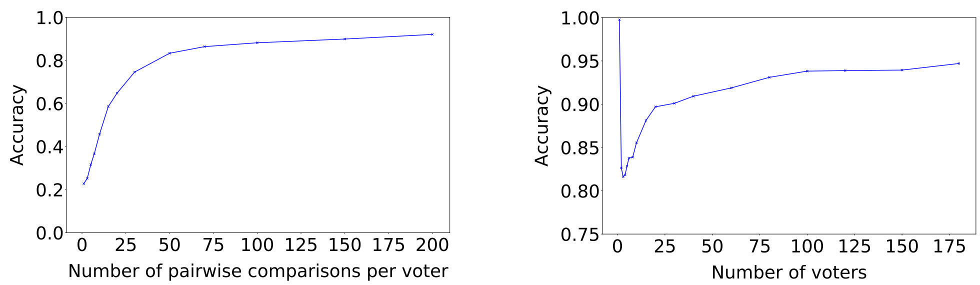
To show robustness with respect to the number of alternatives, we run experiments with $|A| = 3$ (instead of $|A| = 5$). The results are shown in Figure 3.

Similarly to Section 4.2, for Step II, we observe that the accuracy quickly increases as the number of pairwise comparisons increases, and with just 30 pairwise comparisons we achieve an accuracy of 88.8%. With 100 pairwise comparisons, the accuracy is 93.5%. For Step III, we observe that the accuracy increases to 96.2% as the number of voters increases.



(a) Accuracy of Step II (b) Accuracy of Step III

Figure 3: Results with 3 alternatives per instance



(a) Accuracy of Step II (b) Accuracy of Step III

Figure 4: Results with number of features $d = 20$

D.2 Number of Features

To show robustness with respect to the number of features d, we run experiments with $d = 20$ (instead of $d = 10$). The results are shown in Figure 4.

Again, for Step II, we observe that the accuracy quickly increases (though slower than in Section 4.2, because of higher dimension) as the number of pairwise comparisons increases. With just 30 pairwise comparisons we achieve an accuracy of 74.6%, and with 100 pairwise comparisons, the accuracy is 88.2%. For Step III, we observe that the accuracy increases to 94.7% as the number of voters increases.

D.3 Number of Voters in Step II

To show robustness with respect to the number of voters N in Step II, we run the Step II experiments with 40 (instead of $N = 20$). The results are shown in Figure 5.

As before, we observe that the accuracy quickly increases as the number of pairwise comparisons increases, and with just 30 pairwise comparisons we achieve an accuracy of 89.3%. With 100 pairwise comparisons, the accuracy is 94.9%.

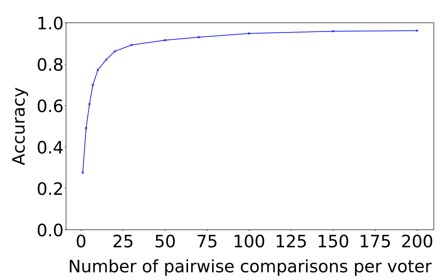


Figure 5: Accuracy of Step II with number of voters $N = 40$

E Additional Discussion

As mentioned in Section 4, we have made some specific choices to instantiate our approach. We discuss two of the most consequential choices.

First, we assume that the mode utilities have a linear structure. This means that, under the TM model, the estimation of the maximum likelihood parameters is a convex program (see Section 4.1), hence we can learn the preferences of a large number of voters — potentially millions of voters, as in the Moral Machine dataset. Moreover, a straightforward summarization method works well. However, dealing with a richer representation for utilities would require new methods for both learning and summarization (Steps II and III).

Second, the instantiation given in Section 4 summarizes the N individual TM models as a single TM model. While the empirical results of Section 4.2 suggest that this method is quite accurate, even higher accuracy can potentially be achieved by summarizing the N models as a *mixture* of K models, for a relatively small K. This leads to two technical challenges: What is a good algorithm for generating this mixture of, say, TM models? And, since the framework of Section 3 would not apply, how should such a mixture be aggregated — does the (apparently mild) increase in accuracy come at great cost to computational efficiency?