Designing Social Choice Mechanisms Using Machine Learning

Lirong Xia
School of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138, USA
lxia@seas.harvard.edu

ABSTRACT

Social choice studies ordinal preference and information aggregation with applications in high-stakes political elections as well as low-stakes movie rating websites. Recently, computational aspects of classical social choice mechanisms have been extensively investigated, yet not much has been done in designing new mechanisms with the help of computational techniques.

In this paper, we outline a workflow to formalize a principled approach towards designing novel social choice mechanisms using machine learning. In the workflow, we clearly separate the following two goals of social choice (1) reaching a compromise among agents' subjective preferences, and (2) revealing the ground truth. For each of the two goals, we discuss criteria for evaluation, main challenges, possible solutions, and future directions.

Categories and Subject Descriptors

J.4 [**Computer Applications**]: Social and Behavioral Sciences–Economics; I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems

General Terms

Algorithms, Economics, Theory

Keywords

Social choice, machine learning

1. INTRODUCTION

Social choice studies preference aggregation problems where agents have ordinal preferences over a set of alternatives and want to make a joint decision. Historically, social choice has been mainly focusing on applications in high-stakes decision-making, e.g., political elections, referendum. Recently, researchers found the idea of making social choice appealing for many low-stakes applications as well, e.g., multiagent systems [10], recommender systems [14], meta-search engines [9], belief merging [11], crowdsourcing [22], and many other e-commerce applications.

In many of these new applications, we need to design novel mechanisms. However, this is not a simple task. We see the following challenges.

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013), Ito, Jonker, Gini, and Shehory (eds.), May, 6–10, 2013, Saint Paul, Minnesota, USA.*

1. The multi-objective nature of social choice. Since agents only give ordinal preferences rather than *transferable utilities* of the alternatives, it is hard to view social choice mechanism as a standard single-objective optimization that maximizes the total utility of the agents (called *social welfare*). In the literature, researchers have proposed some desired *axiomatic properties* (*axioms* for short) to evaluate social choice mechanisms. Unfortunately, usually tradeoffs among these axioms have to be made, which was firstly shown by Arrow's famous impossibility theorem [3]. Taking this view point, designing a good social choice mechanism then amounts to solving a multi-objective optimization problem to explore the tradeoff.

2. Incentives of agents. One complication that distinguishes social choice mechanisms from traditional optimization problems is that agents may pursue their own objectives in the aggregation process, which means that sometimes they may lie about their preferences. Unfortunately, usually such strategic behavior is inevitable, as illustrated in the Gibbard-Satterthwaite theorem [15, 29] for the case of *manipulation*.

3. Computational considerations. For some classical social choice mechanisms, determining the joint decision (winner) is computationally hard. The most prominent example is the *Kemeny rule*, for which it is NP-hard to compute the winner [5]. This may cause problems when the number of alternatives is large, for example in combinatorial domains [17]. Another important aspect is to use computational complexity to protect elections. See [12, 13, 28] for recent surveys. Ideally we want to have a social choice mechanism where the outcome can be computed as fast as possible, whereas strategic behavior is hard to compute, if not impossible. Investigating such computational considerations for social choice mechanisms gave rise to the burgeoning field *computational social choice*.

While most previous research in computational social choice focused on studying computational aspects of classical social choice mechanisms motivated by applications in elections, little has been done in applying state-of-the-art computational techniques to design new social choice mechanisms for new applications. In this paper, we hope to make the criteria and principles for designing new social choice mechanisms clear. Technically, we focus on discussing how to use machine learning in the design. We outline a workflow in Figure 1, which is roughly divided into the following three stages.

Stage 1: First, we need to understand the goal of the mechanism. In general, social choice mechanisms aim at achieving the following two goals.

Goal 1: Reach a compromise among agents' subjective preferences.

Goal 2: Reveal the ground truth.

Most classical social choice mechanisms were designed to achieve

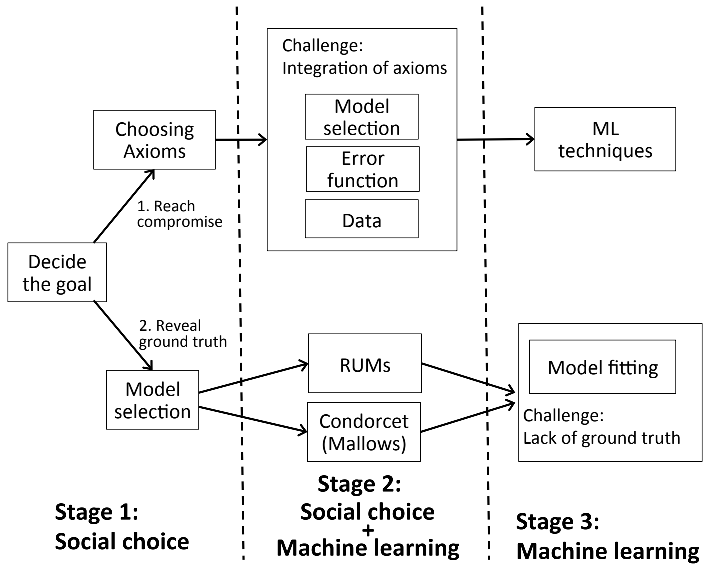

Figure 1: Workflow for designing social choice mechanisms.

the first goal. Therefore, directly applying them to many new applications is inappropriate, since in such cases we want to reveal the ground truth. [1]

To achieve the first goal, we propose to choose a combination of desired axioms for the new mechanism to satisfy. To achieve the second goal, a natural idea is to use *maximium-likelihood estimators (MLEs)*, a well-developed concept in statistics and machine learning. Therefore, for the second goal we need to choose suitable probabilistic models. Stage 1 mainly focus on consideration in social choice.

Stage 2: After having chosen the axioms or probabilistic models, we then propose to anayze the design problem from a machine learning viewpoint, for the two goals respectively. For the first goal, the major challenge is *how to incorporate axioms to the machine learning framework?* We propose the following three ways: model selection, error function formulation, and data augmentation. For the second goal we focus on comparing two classes of models: Condorcet's model (whose MLE is the Kemeny rule) and the random utility models (RUMs).

Stage 3: Finally, we are ready to apply machine learning techniques to design a new mechanism. Specifically, for the second goal, we are facing a model fitting challenge due to the lack of ground truth, for which model fitting methods developed in unsupervised learning is helpful.

One thing worth noting is that sometimes we want to design a new mechanism to achieve both goals. In such cases we need a mechanism that has an MLE interpretation and also satisfies some desired axiomatic properties, see [34] for a recent approach. In this paper we focus on treating the two goals separately to approach the design problem in a clearer and more principled way. Designing a mechanism that achieves both goals is a fascinating and challenging topic for future research.

2. GOAL 1: REACHING COMPROMISE

In this section, we focus on designing *voting rules*, which are social choice mechanisms that select a single winner, and the agents report *linear orders*. Mathematically, a *profile* is a collection of linear orders reported by the agents, and a voting rule r is a mapping from profiles to alternatives.

2.1 Choosing Axioms

Below we list a few representative axiomatic properties. Comprehensive comparisons of commonly studied axioms can be found in [24]. We say a voting rule r satisfies

• **anonymity**, if the winner is invariant to the names of the agents;

• **neutrality**, if whenever we apply a permutation over the names of alternatives to all agents' preferences, the winner is permuted in the same way;

• **consistency**, for each pair of profiles (P_1, P_2), if $r(P_1) = r(P_2)$ then $r(P_1) = r(P_1 \cup P_2)$;

• **Condorcet consistency**, if there exists a *Condorcet winner*,[2] then it must be the winner.

Most axioms can be categorized into the following three classes. Such classification leads to natural numerical measurements for partial satisfiability of axioms, which will be useful in the next section. In the following, each axiom is characterized by a set of inference rules modeled as logical formulas (*Horn clauses*), where each (P, c) is a binary variable that takes 1 if $c = r(P)$.

• A *pointwise axiom* consists of logical formulas of the form $\{\bot \to (P, c)\}$, which reads "the winner for P must be c". For example, Condorcet consistency is a pointwise axiom.

• A *pairwise axiom* consists of logical formulas of the form $\{(P_1, c_1) \to (P_2, c_2)\}$, which reads "if the winner for P_1 is c_1, then the winner for P_2 is c_2". For example, anonymity and neutrality are pairwise axioms.

• A *triple-wise axiom* consists of logical formulas of the form $\{(P_1, c_1) \wedge (P_2, c_2) \to (P_3, c_3)\}$, which reads "if the winner for P_1 is c_1 and the winner for P_2 is c_2, then the winner for P_3 is c_3". For example, consistency is a triple-wise axiom.

Given a voting rule r and an axiom (as a set of formulas for a fixed number of alternatives and agents), let K denote the number of formulas whose premises are true but the conclusion is false. Let L denote the number of formulas whose premises and the conclusion are all true. We define r's satisfiability of the axiom by $L/(K + L)$. A special case is $K = 0$, when r satisfies the axiom in the usual sense.

2.2 Challenge: Incorporating Axiomatic Properties to Machine Learning Framework

Procaccia et al. [27] proposed a natural learning framework for designing voting rules via examples, called *automated design of voting rules*. They assumed that multiple (profile, winner) pairs can be obtained from experts as training data, and aimed at learning a voting rule that minimizes the expected error rate for randomly generated profiles in a PAC learning setting.

If we want to apply the learning techniques in [27] to our framework, and suppose that the data comes from experts, then one fundamental question is, *how can we guarantee that a learned voting rule satisfies the axioms chosen in the first stage?*

Incorporating axioms into model selection. Suppose we can choose a hypothesis space that is exactly the class of voting rules that satisfy the desired axioms, then it is guaranteed that the learned rule will satisfy these axioms. In fact, the approach taken by Procaccia et al. [27] can be seen as an example of this approach, since they designed an efficient learning algorithm for *positional scoring rules*, which are characterized by anonymity, neutrality, con-

[1] To the best of our knowledge, only the *Kemeny* rule was shown to clearly correspond to a process that was designed to achieve the second goal. However, as we will see later in Section 3.1, the Kemeny rule suffers from some critical flaws.

[2] A Condorcet winner is an alternative that beats all other alternatives in pairwise comparisons.

sistency, and *continuity* [35].[3] However, only a few combinations of natural axioms is known to characterizr voting rules that have nice mathematic structures for learning. Identifying mathematical structures of voting rules that are characterized by natural combinations of axioms is an interesting yet challenging direction. One potentially useful class is *generalized scoring rules* [32], which are characterized by anonymity, *homogeneity*, and *finite local consistency* [33]. Moreover, generalized scoring rules are equivalent to combinations of linear binary classifiers via decision trees, where rich machine learning methods can be applied. (The proof is omitted due to the space constraint.)

Incorporating axioms into the error function. Recall from Section 2.1 that the partial satisfiability of axiom can be computed by counting the number of "satisfied" formulas and "unsatisfied" formulas. An interesting idea is to to directly model the (un)satisfiability of axioms as part of the error function. However, this is extremely hard since most, if not all, such satisfiability cannot be easily represented as functions of parameters of the model.

Incorporating axioms into data (and the error function). Since directly modeling the satisfiability into the error function is hard, we may use the following approximation by augmenting the dataset.

• For pointwise axioms, for each $\bot \to (P, c)$, we add (P, c) as a positive example to both training data and testing data, and tweak the error function by adding a term corresponding to misclassification of each such data. For example, for Condorcet consistency we add all pairs (P, c) as positive examples, where c is the Cordorcet winner in P, and each misclassification of such an example will contribute (for example) 0.1 to the error function. In practice this may result in too many new examples, so we may use randomization to choose a subset to add.

• For pairwise axioms, we augment the training data as follows: for each $(P_1, c_1) \to (P_2, c_2)$ used to characterize the axiom, if (P_1, c_1) is in the training data, then we also add (P_2, c_2) to the training data. We note that sometimes a newly added example may lead to another new example by applying a different formula. The error function is tweaked similarly to the pointwise case. For the testing data, we apply similar changes. Again, we may use randomization if there are too many new examples.

• Triple-wise axioms are augmented in a similar way.

Finally, if we are able to successfully incorporate the axioms into the learning framework, we can apply standard machine learning techniques to learn a voting rule. In this paper, we focused on adopting the supervised learning framework developed by Procaccia et al. [27]. In the future we also plan to study unsupervised learning and semi-supervised learning techniques.

3. GOAL 2: REVEALING GROUND TRUTH

To achieve the second goal, it is natural to take the epistemic view of social choice illustrated in Figure 2. In this view, given a "ground truth" outcome o, agents' preferences are generated conditionally independently.

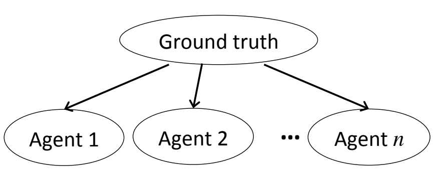

Figure 2: The epistemic view of social choice.

[3]More precisely, these axiomatic properties are variants for social choice mechanisms that may select multiple winners.

Now, designing a mechanism amounts to selecting a probability model, and then use its *maximum likelihood estimator (MLE)* to select the outcome. We will first briefly discuss pros and cons of two popular families of probabilistic models, then point out challenges and potential solutions. In this section, the outcome (ground truth) of a social choice mechanism is a ranking over the alternatives.

3.1 Condorcet's probabilistic model

In 1785, Condorcet [7] proposed the following probabilistic model. Given a ground truth ranking W over the alternatives and a fixed number $p > 1/2$, each agent generates comparisons between pairs of alternatives independently, such that for each pair (a, b), with probability p the agent's preference is the same as the pairwise comparison between a and b in W, and with probability $1 - p$ her preference is different from the pairwise comparison in W. This leads to a distribution over all (possibly cyclic) orders over the alternatives. Restricted to linear orders, Condorcet's model is mathematically equivalent to the Mallows model [21]. It has been shown that the MLE of Condorcet's model is exactly the Kemeny rule [36].

Condorcet's model has been criticized mainly for the following two reasons: first, it assumes too much independence among pairwise comparisons, leading to possibly cyclic preferences; second, computing the winning ranking is an NP-hard problem [5]. The first criticism can be defended by arguing that we only need to focus on the restriction of the model on linear orders, giving us exactly the Mallows model. To address the high computational complexity, various techniques have been developed to solve the problem in practice, including an ILP formulation [8], approximation algorithms [1], and fixed-parameter analyses [6]. More recently, Lu and Boutilier [19] developed an efficient algorithm to learn Mallows model from data.

3.2 The Random Utility Models

In *random utility models* (RUMs) [31], each alternative c_i is parameterized by a ground truth "intrinsic" utility θ_i. Given these intrinsic utilities (parameters), each agent independently samples a random utility U_i for alternative c_i from a distribution μ_i, and rank the alternatives according to the sampled utilities from high to low. This model is illustrated in Figure 3.

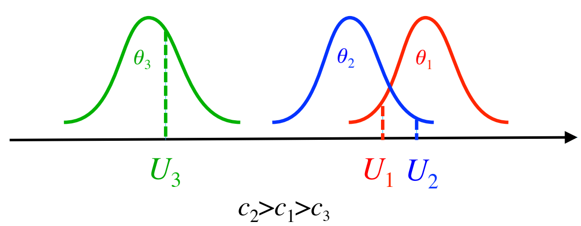

Figure 3: Generating an agent's preferences under RUMs.

Compared to Condorcet's model, RUMs naturally handle the transitivity issue. The main criticism on RUMs has been the computational intractability of MLE inference—no closed-form formula was known for many types of distributions μ_i. The only exception is the *Plackett-Luce* model [20, 26], which has found many applications in economics [23] as well as *learning to rank* [18], and has been applied to study elections [16]. Recently, Azari et al. [4] designed an MC-EM algorithm for MLE inference under general RUMs where each μ_i belongs to the *exponential family*, and showed its efficiency for both synthetic and real-world data.

3.3 Challenge: Model Fitting

Of course building new models and developing faster inference algorithms, especially for RUMs, are exciting research direction. Equipped with the MLE inference algorithms mentioned in the last subsection, we are now able to set out to evaluate existing models for different application domains.

However, evaluating how well an MLE mechanism achieves the second goal (reveal the ground truth) is not easy for social choice applications. This is mainly due to the lack of ground truth in most social choice datasets. If we know the ground truth, for example in some crowdsourcing settings [25], then the criterion for model fitting is quite straightforward: choose the model whose MLE ranking is on average closest to the ground truth according to some distance metric, for example the *Kendall Tau* distance.

Suppose we do not have access to ground truth, which means that the dataset only contains a collection of profiles. In this case we may still test the fitness of the model by applying some generic unsupervised model fitting techniques, including minimizing the Akaike information criterion (AIC) [2] or Bayesian information criterion (BIC) [30]. This may not work as well as the case where we can access to ground truth, but it is probably the best that can be done as general criteria. Collecting data and setting model-fitting criteria require expertise in the application domain, which we believe to be a very promising and practical direction for future research.

4. CONCLUSION

In this paper, we discussed some principles, challenges, and potential solutions for designing social choice mechanism, especially for new applications. Depending on the goal of design, we see completely different combinations of principles and techniques, where machine learning turns out to be helpful for both of them. Therefore, we believe that designing novel social choice applications with the help of computational techniques especially machine learning will constitute an important direction for future research, especially for computational social choice.

5. ACKNOWLEDGMENTS

This work is supported by NSF under Grant #1136996 to the Computing Research Association for the CIFellows Project. The author thanks Hossein Azari, David Parkes, Ariel Procaccia, Georgios Zervas, and anonymous AAMAS-13 reviewers for helpful discussions and suggestions.

6. REFERENCES

[1] N. Ailon, M. Charikar, and A. Newman. Aggregating inconsistent information: Ranking and clustering. In *STOC*, pages 684–693, 2005.

[2] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on AC*, 19(6):716–723, 1976.

[3] K. Arrow. *Social choice and individual values*. New Haven: Cowles Foundation, 2nd edition, 1963. 1st edition 1951.

[4] H. Azari Soufiani, D. C. Parkes, and L. Xia. Random utility theory for social choice. In *NIPS*, 2012.

[5] J. Bartholdi, III, C. Tovey, and M. Trick. Voting schemes for which it can be difficult to tell who won the election. *Social Choice and Welfare*, 6:157–165, 1989.

[6] N. Betzler, M. R. Fellows, J. Guo, R. Niedermeier, and F. A. Rosamond. Fixed-parameter algorithms for kemeny rankings. *TCS*, 410:4554–4570, 2009.

[7] M. d. Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. Paris: L'Imprimerie Royale, 1785.

[8] V. Conitzer, A. Davenport, and J. Kalagnanam. Improved bounds for computing Kemeny rankings. In *AAAI*, pages 620–626, 2006.

[9] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar. Rank aggregation methods for the web. In *WWW*, pages 613–622, 2001.

[10] E. Ephrati and J. S. Rosenschein. The Clarke tax as a consensus mechanism among automated agents. In *AAAI*, pages 173–178, 1991.

[11] P. Everaere, S. Konieczny, and P. Marquis. The strategy-proofness landscape of merging. *JAIR*, 28:49–105, 2007.

[12] P. Faliszewski, E. Hemaspaandra, and L. A. Hemaspaandra. Using complexity to protect elections. *CACM*, 53:74–82, 2010.

[13] P. Faliszewski and A. D. Procaccia. AI's war on manipulation: Are we winning? *AI Magazine*, 31(4):53–64, 2010.

[14] S. Ghosh, M. Mundhe, K. Hernandez, and S. Sen. Voting for movies: the anatomy of a recommender system. In *ACAA*, pages 434–435, 1999.

[15] A. Gibbard. Manipulation of voting schemes: A general result. *Econometrica*, 41:587–601, 1973.

[16] I. C. Gormley and T. B. Murphy. Analysis of Irish third-level college applications data. *JRSSSA*, 169(2):361–379, 2006.

[17] J. Lang and L. Xia. Sequential composition of voting rules in multi-issue domains. *MSS*, 57(3):304–324, 2009.

[18] T.-Y. Liu. *Learning to Rank for Information Retrieval*. Springer, 2011.

[19] T. Lu and C. Boutilier. Learning mallows models with pairwise preferences. In *ICML*, pages 145–152, 2011.

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

[21] C. L. Mallows. Non-null ranking model. *Biometrika*, 44(1/2):114–130, 1957.

[22] A. Mao, A. D. Procaccia, and Y. Chen. Social choice for human computation. In *HCOMP*, 2012.

[23] D. McFadden. Conditional logit analysis of qualitative choice behavior. In *Frontiers of Econometrics*, pages 105–142, New York, NY, 1974. Academic Press.

[24] H. Nurmi. *Comparing voting systems*. Springer, 1987.

[25] T. Pfeiffer, X. A. Gao, A. Mao, Y. Chen, and D. G. Rand. Adaptive Polling and Information Aggregation. In *AAAI*, pages 122–128, 2012.

[26] R. L. Plackett. The analysis of permutations. *JRSSSC*, 24(2):193–202, 1975.

[27] A. D. Procaccia, A. Zohar, Y. Peleg, and J. S. Rosenschein. The learnability of voting rules. *AIJ*, 173:1133—1149, 2009.

[28] J. Rothe and L. Schend. Typical-Case Challenges to Complexity Shields That Are Supposed to Protect Elections Against Manipulation and Control: A Survey. In *ISAIM*, 2012.

[29] M. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10:187–217, 1975.

[30] G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464, 1978.

[31] L. L. Thurstone. A law of comparative judgement. *Psychological Review*, 34(4):273–286, 1927.

[32] L. Xia and V. Conitzer. Generalized scoring rules and the frequency of coalitional manipulability. In *EC*, pages 109–118, 2008.

[33] L. Xia and V. Conitzer. Finite local consistency characterizes generalized scoring rules. In *IJCAI*, pages 336–341, 2009.

[34] L. Xia and V. Conitzer. A maximum likelihood approach towards aggregating partial orders. In *IJCAI*, pages 446–451, 2011.

[35] H. P. Young. Social choice scoring functions. *SIAM Journal on Applied Mathematics*, 28(4):824–838, 1975.

[36] H. P. Young. Optimal voting rules. *JEP*, 9(1):51–64, 1995.