Automated Mechanism Design without Money via Machine Learning

Harikrishna Narasimhan
Harvard University
hnarasimhan@seas.harvard.edu

Shivani Agarwal
Radcliffe Institute for Advanced
Study, Harvard University
Indian Institute of Science
shivani@csa.iisc.ernet.in

David C. Parkes
Harvard University
parkes@seas.harvard.edu

Abstract

We use statistical machine learning to develop methods for automatically designing mechanisms in domains without money. Our goal is to find a mechanism that best approximates a given target function subject to a design constraint such as strategy-proofness or stability. The proposed approach involves identifying a rich parametrized class of mechanisms that resemble discriminant-based multiclass classifiers, and relaxing the resulting search problem into an SVM-style surrogate optimization problem. We use this methodology to design *strategy-proof* mechanisms for social choice problems with single-peaked preferences, and *stable* mechanisms for two-sided matching problems. To the best of our knowledge, ours is the first automated approach for designing stable matching rules. Experiments on synthetic and real-world data confirm the usefulness of our methods.

1 Introduction

Mechanism design studies situations where a set of self-interested agents each hold private information regarding their preferences over different outcomes. In mechanism design without money, agents make reports about their preferences, perhaps untruthfully, and the mechanism selects an outcome based on these reports. Canonical problems include those of social choice (locating a student center), matching (students to high schools), and assignment (faculty to offices).

The Gibbard-Satterthwaite theorem [Gibbard, 1973; Satterthwaite, 1975], states that *strategy-proofness* (truthful reporting as a dominant strategy) is unattainable in general domains without money. In this light, a large theoretical literature provides results, both positive and negative, for particular domains. Designs are typically justified axiomatically and without optimizing a quantitative objective.

In comparison, the method of *automated mechanism design* (AMD) [Conitzer and Sandholm, 2002] seeks to use computation to automatically find mechanisms that are tailored to the needs of an application. The typical approach is to formulate a search problem over a space of mechanisms and solve using conventional search heuristics [Conitzer and Sandholm, 2004; Guo and Conitzer, 2010; Sui *et al.*, 2013].

Most prior work assumes that the design objective is provided in closed-form. But a designer's requirements may be more complex than maximizing a simple objective such as welfare. In this paper, we adopt a more flexible approach that allows the designer to encode requirements in the form of a *target outcome rule*, with this rule described through labels on a set of inputs drawn from an underlying preference distribution. Consider for example a school choice program that seeks to change an existing matching mechanism so as to increase the diversity of students enrolled in each school. Rather than an explicit objective, we want a designer (perhaps a school board) to be able to present requirements via desired choices on past preferences of students and schools.

There are key challenges in solving this problem. Often, we do not have a complete characterization of the space of desired mechanisms. Even when a characterization is available, identifying the mechanism that best approximates a target rule can be computationally hard. We provide a novel framework to resolve these issues by using tools from machine learning. The methodology involves identifying a rich subset of desired mechanisms that can be parametrized by continuous weights. The distance of a mechanism in this class from the target rule is modeled as a loss function. We use tools from machine learning to relax this distance measure into a continuous surrogate objective, and solve the resulting continuous optimization problem using standard solvers. Specifically, we choose a class of rules that can be modeled as discriminant-based multiclass classifiers, and relax the problem of finding the optimal rule into a support vector machine (SVM) style surrogate optimization problem.

We apply this new framework to two canonical settings: (1) social choice with single-peaked preferences, where the design constraint is *strategy-proofness* (no agent can usefully misreport his preferences); and (2) two-sided matching problems, where the design constraint is *stability* (no two agents are better off being matched with each other than to their assigned match). In each case, we introduce a new class of outcome rules that satisfy the desired property and are amenable to optimization through machine learning techniques.

For the problem of social choice with single-peaked preferences, we introduce a class of parametrized strategy-proof rules called the *weighted generalized median (WGM) rules*, which includes the median and percentile rules considered in previous AMD works [Procaccia and Tennenholtz, 2009;

Sui *et al.*, 2013]. These rules can be modeled (with some relaxation) as discriminant-based classifiers, and we frame a SVM based surrogate optimization problem to find the optimal rule. Experiments on synthetic preference data show that the proposed approach is quite robust, and performs better than the best rules from baseline strategy-proof classes.

For the problem of two-sided matching, we introduce a class of parametrized stable rules called the *weighted polytope rules*, which includes the classical deferred acceptance (DA) mechanisms. These rules resemble discriminant-based structured classifiers, allowing us to frame a structural SVM problem to optimize over the class. To the best of our knowledge, this is the first automated approach for designing stable matching rules. Experiments on synthetic preference data and those motivated by a real-world school choice problem confirm that our method performs better than DA rules.

Related work. Procaccia et al. [2009] considered a setting similar to ours, but focused on showing learnability of particular classes of voting rules, with no incentive considerations. In comparison, we provide a general methodology for AMD without money, and apply it to two settings with specific design constraints. Another related work by Sui et al. [2013] focused on designing strategy-proof percentile rules for a multidimensional variant of our social choice setting. We handle a strictly larger class of strategy-proof (WGM) rules; also, for specific distance measures, our approach can be applied to their setting, by decomposing the optimization problem into a separate problem for each dimension. The use of machine learning for AMD with money was pioneered by Dütting et al. [2015], who use SVMs to design payment rules for auction mechanisms. This is the closest prior work to the present paper, but differs in that it learns payment rules rather than outcome rules, and achieves approximate strategy-proofness. Sample-complexity results are also available for the design of revenue-optimal auctions [Cole and Roughgarden, 2014].

2 Problem Setup

Let $N = \{1, \ldots, n\} = [n]$ denote a set of agents, and Ω be a set of outcomes. Each agent has a total preference order on outcomes (perhaps allowing for ties). We will use $y \succ_i y'$ to denote that agent i strictly prefers outcome $y \in \Omega$ to $y' \in \Omega$, and $y \succeq_i y'$ to denote that agent i strictly prefers y to y' or is indifferent. Let $\mathcal{P}$ denote the set of all allowable preference profiles $\succ = (\succ_i)_{i \in N}$ of the agents. In MD without money, each agent makes a report (perhaps untruthfully) about his preference order, and the mechanism is defined by an *outcome rule* $f : \mathcal{P} \to \Omega$ that maps the agent reports $\hat{\succ} \in \mathcal{P}$ to an outcome $f(\hat{\succ}) \in \Omega$. We assume that the preference profile is distributed according to an unknown distribution $\succ \sim \mathcal{D}$.

We are interested in outcome rules that satisfy one or more design constraints. One such constraint in social choice problems is *strategy-proofness*. Another desirable property in two-sided matching problems is *stability*.

Given a *target outcome rule* $g : \mathcal{P} \to \Omega$ that need not satisfy the design constraint, our goal is to find an outcome rule that closely approximates g, subject to the given constraint. The closeness to g is measured in terms of a *distance function* $\mathbb{D} : \Omega \times \Omega \to \mathbb{R}_+$, where $\mathbb{D}(y, y')$ gives the penalty for outputting outcome y' when the target outcome is y. Since the set of all rules that satisfy the design constraint is often not well understood or otherwise hard to work with directly, as a first step, we identify a sufficiently rich, parametric family of desired rules $\mathcal{F}$, and re-formulate our goal as minimizing the expected distance over this class:

$$\min_{f \in \mathcal{F}} \mathbf{E}_{\succ \sim \mathcal{D}} \big[\mathbb{D}\big(g(\succ), f(\succ)\big) \big]. \quad (1)$$

We assume access to the distribution $\mathcal{D}$ and target rule g through a sample of agent preferences profiles drawn i.i.d. from $\mathcal{D}$, along with labels from the target rule: $S = \big((\succ^1, y^1 = g(\succ^1)), \ldots, (\succ^L, y^L = g(\succ^L)) \big) \in (\mathcal{P} \times \Omega)^L$. We then solve an empirical version of the problem in Eq. (1):

$$\min_{f \in \mathcal{F}} \sum_{r=1}^{L} \mathbb{D}\big(y^r, f(\succ^r)\big).$$

Overview of approach. Our approach essentially involves relaxing the above problem into a continuous optimization problem that can be solved using standard solvers. We begin by identifying a rich class of rules $\mathcal{F}$ that are parametrized by continuous weights $\mathbf{w} \in \mathbb{R}^K$. In the context of learning, the distance $\mathbb{D}$ of a rule in this class from the target can be viewed as a *loss* function, and one can use tools from machine learning to relax $\mathbb{D}$ into a continuous surrogate function. We use the popular multiclass SVM framework of [Crammer and Singer, 2002] to construct this relaxation.

In particular, we are interested in rules that can be modeled (perhaps with some relaxation) as a discriminant based multiclass classifier, i.e. in terms of a (continuous) discriminant function $H_{\mathbf{w}} : \mathcal{P} \times \Omega \to \mathbb{R}$, with the output for a preference profile given by the outcome that maximizes this function: $f_{\mathbf{w}}(\succ) = \operatorname{argmax}_{y \in \Omega} H_{\mathbf{w}}(\succ, y)$. This structure of the outcome rule then allows us to invoke the SVM framework to construct a surrogate objective for $\mathbb{D}$. Specifically, for profile $\succ$ and target y, we replace $\mathbb{D}(y, f_{\mathbf{w}}(\succ))$ by a continuous function: $\max_{y' \in \Omega} \big\{ \mathbb{D}(y, y') + H_{\mathbf{w}}(\succ, y') - H_{\mathbf{w}}(\succ, y) \big\}$. The parameters obtained by solving the resulting continuous optimization problem are used to construct a rule in $\mathcal{F}$.

3 Strategy-Proof Social Choice

We consider a setting with m outcomes $\Omega = \{y_1, \ldots, y_m\}$, and the outcomes are ordered, say as $y_1 < \ldots < y_m$. We assume that preference orders are strict and *single-peaked*. This means that each agent prefers one outcome the most, referred to as his peak, and prefers the other outcomes lesser as they are farther away from his peak.

More formally, let $p_i \in \Omega$ denote agent i's peak. For all $a, b \in \Omega$ such that $b < a < p_i$ or $p_i < a < b$, we have $a \succ_i b$. This preference structure occurs naturally in the one-dimensional facility location problem, where the outcomes are say locations where a fire station can be set up, and each agent prefers one location the most, preferring the other locations relative to his/her most-preferred location. For simplicity, we assume that the number of agents n is odd.

Strategy-proofness. A desirable property of an outcome rule $f : \mathcal{P} \to \Omega$ in this setting is strategy-proofness, where it is in the best interest of every agent to report their peaks

truthfully, whatever the others reports. More formally, f is strategy-proof if for all i: $f(\succ) \succeq_i f((\succ'_i, \succ_{-i}))$, $\forall \succ'_i$, where $(\succ'_i, \succ_{-i})$ is a preference profile where agent i reports $\succ'_i$ while all other agents retain their previous report in $\succ$.

A well-known strategy-proof rule is the *median rule* which returns the median of the reported peaks (under the given outcome ordering) [Black, 1948]. The argument for strategy-proofness of this rule is easy to see. If an agent's peak is at the median, he does not gain by misreporting his peak. If his peak is to the left of the median, the only way he can change the output is by reporting an outcome to the right of the median; but this would result in the rule outputting an outcome that is farther to the right, and less preferred. A symmetric argument holds for a peak to the right of the median. In fact, an outcome rule that outputs any k^{th}-*percentile* of the reported peaks, for a fixed $k \in [n]$, is strategy-proof [Sui *et al.*, 2013].

3.1 Weighted Generalized Median Rules

We introduce a richer class of strategy-proof rules parametrized by continuous weights, called the *weighted generalized median* (WGM) rules (see Figure 1). We will then develop a SVM approach to optimize over this class. We begin with a well-known generalization of the median rule:

Definition 1 (**Generalized Median (GM) Rule** [Moulin, 1980]). *For a vector of pseudo-peaks* $\mathbf{u} = (u_1, \ldots, u_{n-1}) \in \Omega^{n-1}$ *fixed a priori, a GM rule is defined as:*

$$f_{\text{GM}}(\succ; \mathbf{u}) = \text{median}(p_1, \ldots, p_n, u_1, \ldots, u_{n-1}),$$

where p_i is the peak corresponding to preference ordering $\succ_i$.

Example 1. *Let $n = 3$ and $m = 5$. Consider a GM rule with pseudo-peaks $\mathbf{u} = (y_4, y_5)$. If the reported agent peaks are y_3, y_1, y_4 respectively, this rule will output y_4; on the other hand, the median rule will output y_3 for the same report.*

Since the pseudo-peaks $u_1, \ldots, u_{n-1}$ are fixed a priori, and do not change with the reports of the agents, a GM rule is strategy-proof. In fact, for appropriate choices of pseudo-peaks, one can recover the median and percentile rules from the GM rule. E.g. the kth-percentile is obtained by setting $n - k + 1$ peaks to y_1 and the remaining to y_m. Moreover, the class of all GM rules is precisely the set of outcome rules that are strategy-proof, *onto* and *anonymous* [Moulin, 1980].

For any $\mathbf{u} \in \Omega^{n-1}$, define a function $rank_{\mathbf{u}} : \mathcal{P} \times \Omega \to \mathbb{N}$, that for a reported profile $\succ \in \mathcal{P}$ and outcome $y \in \Omega$, returns the number of peaks (both reported and in $\mathbf{u}$) that lie to the left of y: $rank_{\mathbf{u}}(\succ, y) = \sum_{i=1}^{n} \mathbf{1}(p_i \leq y) + \sum_{i=1}^{n-1} \mathbf{1}(u_i \leq y)$. It can then be verified that the GM rule outputs the smallest outcome whose rank is greater or equal to n:

$$f_{\text{GM}}(\succ; \mathbf{u}) = \min\{y \in \Omega \mid rank_{\mathbf{u}}(\succ, y) \geq n\}.$$

Assigning non-negative weights $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n_+$ to each agent and non-negative weights $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_m) \in \mathbb{R}^m_+$ to each outcome in Ω, we define the *weighted rank* as $rank_{\boldsymbol{\alpha}, \boldsymbol{\beta}}(\succ, y) = \sum_{i=1}^{n} \alpha_i \mathbf{1}(p_i \leq y) + \sum_{i=1}^{m} \beta_i \mathbf{1}(y_i \leq y)$. The following is then a continuous extension to the GM rule:

Definition 2 (**Weighted Generalized Median (WGM) Rule**). *Let $\boldsymbol{\alpha} \in \mathbb{R}^n_+$ and $\boldsymbol{\beta} \in \mathbb{R}^m_+$ be non-negative weights on agents and items respectively, and $t \in \mathbb{R}_+$ be a threshold.*

$$\boxed{\mathcal{F}_{\text{SP}} \supset \mathcal{F}_{\text{WGM}} \supset \mathcal{F}_{\text{GM}} \supset \mathcal{F}_{\text{percentile}} \ni f_{\text{median}}}$$

Figure 1: Hierarchy of strategy-proof rules for social choice.

A WGM rule parametrized by $\mathbf{w} = [\boldsymbol{\alpha}, \boldsymbol{\beta}, t]$ outputs for a given preference profile $\succ$, the smallest outcome in Ω whose weighted rank w.r.t. $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ is greater or equal to t:

$$f_{\text{WGM}}(\succ; \mathbf{w}) = \min\{y \in \Omega \mid rank_{\boldsymbol{\alpha}, \boldsymbol{\beta}}(\succ, y) \geq t\}.$$

When $\boldsymbol{\alpha} = 1^n$, $\boldsymbol{\beta}$ is a vector in $\mathbb{N}^m$ with entries adding up to $n - 1$, and $t = n$, the WGM rule reduces to a GM rule with the pseudo-peaks given by the non-zero entries of $\boldsymbol{\beta}$.

Example 2. *Let $n = 3$ and $m = 5$. Consider a WGM rule with $\boldsymbol{\alpha} = (2, 1, 3)$, $\boldsymbol{\beta} = (1.5, 0.5, 0, 0, 1.5)$ and $t = 3$. For agent peaks y_3, y_1, y_4, $rank_{\boldsymbol{\alpha}, \boldsymbol{\beta}}(\succ, y_1) = 2.5$; $rank_{\boldsymbol{\alpha}, \boldsymbol{\beta}}(\succ, y_2) = 3$; $rank_{\boldsymbol{\alpha}, \boldsymbol{\beta}}(\succ, y_3) = 5$; $rank_{\boldsymbol{\alpha}, \boldsymbol{\beta}}(\succ, y_4) = 8$; $rank_{\boldsymbol{\alpha}, \boldsymbol{\beta}}(\succ, y_5) = 9.5$; and the output is y_2.*

Theorem 1. The WGM rules are strategy-proof and contain the median, percentile and GM rules as special cases.

We defer the proof of strategy-proofness (which is similar in spirit to that for GM rules) for the full version of this paper.

3.2 Multiclass SVM for Designing WGM Rules

We next devise a multiclass SVM style approach for solving our original problem in Eq. (1) over WGM rules.

Given a sample $S = ((\succ^1, y^1), \ldots, (\succ^L, y^L))$ labeled by a target rule that need not be strategy-proof, we formulate an empirical version of Eq. (1) over parameters $\mathbf{w} \in \mathbb{R}^{n+m+1}_+$:

$$\min_{\mathbf{w} \in \mathbb{R}^{n+m+1}_+} \sum_{r=1}^{L} \mathbb{D}(y^r, f_{\text{WGM}}(\succ^r; \mathbf{w})).$$

A natural distance measure here is the absolute distance between the output and target outcomes: $\mathbb{D}_{\text{abs}}(y, y') = |y - y'|$. Since a WGM rule is itself a discontinuous function of parameters $\mathbf{w}$, the resulting optimization objective is not continuous in $\mathbf{w}$, and can be difficult to directly optimize. To address this issue, we resort to the multiclass SVM framework and replace $\mathbb{D}_{\text{abs}}$ with a continuous surrogate objective. For this, we adopt a slight variant of the WGM rule that resembles a discriminant-based classifier (but need not be strategy-proof), and eventually construct a WGM rule from the parameters obtained by optimizing a surrogate based on the proxy rule.

Discriminant-based proxy: The proxy rule that we use outputs for any preference profile $\succ$, an outcome whose weighted rank is closest to threshold t in the least-square sense (this is slightly different from returning the smallest outcome in Ω that has weighted rank greater or equal to t):

$$f_{\text{proxy}}(\succ; \mathbf{w}) \in \underset{y \in \Omega}{\operatorname{argmin}} (rank_{\boldsymbol{\alpha}, \boldsymbol{\beta}}(\succ, y) - t)^2 \qquad (2)$$

This resembles a discriminant-based multiclass classifier with discriminant function $H_{\mathbf{w}}(\succ, y) = -(rank_{\boldsymbol{\alpha}, \boldsymbol{\beta}}(\succ, y) - t)^2$, and where the output is the outcome with the highest discriminant score: $f_{\text{proxy}}(\succ; \mathbf{w}) = \operatorname{argmax}_{y \in \Omega} H_{\mathbf{w}}(\succ, y)$.

Surrogate objective: With the proxy rule, we can use the SVM framework to construct a continuous surrogate for $\mathbb{D}_{\text{abs}}$:

$$\sum_{r=1}^{L} \max_{y' \in \Omega} \left\{ \mathbb{D}_{\text{abs}}(y^r, y') + H_{\mathbf{w}}(\succ^r, y') - H_{\mathbf{w}}(\succ^r, y^r) \right\}.$$

The parameters $\mathbf{w}^* = [\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*, t^*]$ obtained by optimizing (a possibly regularized form of) the surrogate objective can then be used to construct a rule from the original WGM class: $f^*_{\text{WGM}}(\succ; \mathbf{w}^*) = \min \left\{ y \in \Omega \mid rank_{\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*}(\succ, y) \geq t^* \right\}$.

Unlike a standard SVM, where the discriminant function is linear in the parameters, discriminant $H_{\mathbf{w}}$ is quadratic in $\mathbf{w}$. As a result, the optimization objective, containing a difference of two quadratic functions, need not be convex. We apply a standard gradient-based method (with multiple random restarts) to solve the optimization problem.

4 Stable Two-sided Matching

The second setting that we consider is two-sided matching [Manlove, 2013]. There are two types of agents, say hospitals H and doctors D, with $N = H \cup D$. Given preferences from agents in one group over agents on the other side, the goal is to design a mechanism that returns a matching of hospitals to doctors. We will allow agents to be unmatched, in which case we will say that an agent is matched with ϕ. The outcome space Ω contains matchings of the form $y : H \cup D \to H \cup D \cup \{\phi\}$. This setting arises in several practical applications such as school choice programs where students need to be matched to schools. For ease of exposition, we will only consider strict agent preferences and one-to-one matchings. Later we will explain how the proposed method extends to weak preferences and many-to-one matchings. It is w.l.o.g. to assume $|H| = |D| = n$.

Stability. A desirable property here is that of stability, which states that a matching is *stable* if no pair of hospital and doctor prefer being matched to each other rather than being matched according to y. Also, no hospital or doctor prefers being unmatched rather than being matched according to y. More formally, y is stable for preference profile $\succ$ if $\nexists (h, d) \in H \times D$ s.t. $d \succ_h y(h)$ and $h \succ_d y(d)$, and moreover, $\nexists h \in H$ s.t. $\phi \succ_h y(h)$, and $\nexists d \in D$ s.t. $\phi \succ_d y(d)$. It is known that a stable matching always exists for a preference profile [Gale and Shapley, 1962].

Example 3. *Let $H = \{h_1, h_2, h_3\}$ and $D = \{d_1, d_2, d_3\}$. Consider the following agent preferences:*

$\succ_{h_1}$: d_2 d_1 d_3 $\succ_{h_2}$: d_1 d_2 d_3 $\succ_{h_3}$: d_2 d_3 d_1
$\succ_{d_1}$: h_1 h_2 h_3 $\succ_{d_2}$: h_3 h_2 h_1 $\succ_{d_3}$: h_1 h_3 h_2

The matching $((h_1, d_1), (h_2, d_3), (h_3, d_2))$ is stable, while the matching $((h_1, d_1), (h_2, d_2), (h_3, d_3))$ is unstable as h_3 and d_2 are better off being matched with each other.

We will say an outcome rule $f : \mathcal{P} \to \Omega$ is then *stable* if it outputs a stable matching for all preference profiles. Unlike strategy-proofness, stability is a local property, in that, it can be checked instance-by-instance.

A stable rule that is widely used is the deferred acceptance (DA) rule [Gale and Shapley, 1962]. This rule works in multiple rounds, with one side of agents making proposals in each round and the other side holding on to the best proposal received so far. Below, we introduce a richer class of parametrized stable rules that span an entire spectrum of intermediate rules between the doctor- and hospital-proposing DA rules. The proposed rules resemble discriminant-based classifiers, and can be optimized using structural SVMs.

4.1 Weighted Polytope (WP) Rules

We begin with a well-known characterization of stable matchings as extreme points of a polytope [Roth *et al.*, 1993]. From this, we construct stable outcome rules by setting up a linear optimization problem over the matching polytope.

Let us slightly overload notation and represent a matching between hospitals and doctors as a Boolean matrix $y \in \{0, 1\}^{n \times n}$ with $\sum_{d=1}^{n} y_{hd} \leq 1$, $\forall h$ and $\sum_{h=1}^{n} y_{hd} \leq 1$, $\forall d$, where $y_{hd} = 1$ iff $y(h) = d$ or equivalently $y(d) = h$. Also, for a preference profile $\succ \in \mathcal{P}$, let $\mathcal{A}(\succ) = \{(h, d) \in H \times D : d \succ_h \phi \text{ and } h \succ_d \phi\}$ denote the set of all pairs of hospitals and doctors who prefer being matched to each other rather than being unmatched. Then the set of all stable matchings for $\succ$ are those that satisfy the following constraints:

$$y_{hd} = 0 \ \ \forall (h, d) \notin \mathcal{A}(\succ);$$

$$y_{hd} + \sum_{d' \succ_h d} y_{hd'} + \sum_{h' \succ_d h} y_{h'd} \geq 1 \ \ \forall (h, d) \in \mathcal{A}(\succ).$$

The first constraint disallows a hospital and doctor from being matched when one of them is better off remaining unmatched. The second constraint ensures that for all other (h, d), either h is matched with d, h is matched to a doctor that it prefers more than d, or d is matched to a hospital he/she prefers more than h.

Let $\Omega(\succ)$ denote the set of all Boolean matchings that satisfy the above constraints for profile $\succ$. Now, suppose we relax the above constraints and allow for fractional matrices in $[0, 1]^{n \times n}$, denoting the resulting polytope as $\widetilde{\Omega}(\succ)$:

Theorem 2. ([Roth *et al.*, 1993]) A matching is stable for $\succ$ if and only if it is an extreme point of $\widetilde{\Omega}(\succ)$.

The above result allows us to construct a stable outcome rule by framing a linear optimization problem over $\widetilde{\Omega}(\succ)$. This will always yield a stable matching.

Definition 3 (**Weighted polytope (WP) rule**). *Define $\boldsymbol{\lambda} : \mathcal{P} \to \mathbb{R}^{n \times n}$ that assigns a weight $\lambda_{hd}(\succ) \in \mathbb{R}$ to each pair of hospital and doctor (h, d). A WP rule is then defined as:*

$$f_{\text{WP}}(\succ; \boldsymbol{\lambda}) \in \underset{y \in \widetilde{\Omega}(\succ)}{\operatorname{argmax}} \sum_{h=1}^{n} \sum_{d=1}^{n} \lambda_{hd}(\succ) y_{hd}.$$

For specific choices of $\boldsymbol{\lambda}$, the proposed rule reduces to the DA rules. To see this, define once again a rank function $rank : \mathcal{P} \times N \to [n]$ that for any (h, d), computes the number of doctors preferred by h less than d or the number of hospitals preferred by d less than h, i.e. $rank(\succ_h, d) = |\{d' \in D : d \succ_h d'\}|$, and $rank(\succ_d, h) = |\{h' \in H : h \succ_d h'\}|$. When $\lambda_{hd}(\succ) = rank(\succ_h, d)$, it is easy to show that the WP rule computes a matching that is optimal for the hospitals (i.e. where each hospital is matched to its most-preferred

achievable doctor), which is precisely what is output by the hospital-proposing DA rule [Gale and Shapley, 1962]. When $\lambda_{hd}(\succ) = rank(\succ_d, h)$, we get the doctor-proposing rule.

Theorem 3. The WP rules are stable and contain the DA rules as special cases.

We prescribe for $\boldsymbol{\lambda}$ the following parametrized form, which yields an intermediate rule between the hospital-proposing and doctor-proposing DA rules:

$$\lambda_{hd}(\succ; \mathbf{W}) = a_{hd}\, rank(\succ_h, d) + b_{hd}\, rank(\succ_d, h) + c_{hd},$$

where $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^{n \times n}$ are parameters and $\mathbf{W} = [\mathbf{a}, \mathbf{b}, \mathbf{c}]$.

4.2 Structural SVM for Designing WP Rules

We next use the framework of (structural) SVMs to solve the original problem in Eq. (1) over WP rules. Given sample $S = \left((\succ^1, y^1), \ldots, (\succ^L, y^L)\right)$ labeled by a target rule that need not be stable, we solve an empirical version of this problem:

$$\min_{\mathbf{W} \in \mathbb{R}^{3(n \times n)}} \sum_{r=1}^{L} \mathbb{D}\big(y^r, f_{\mathrm{WP}}(\succ^r; \mathbf{W})\big). \tag{3}$$

A natural distance measure here is the Hamming distance between the output and target matchings: $\mathbb{D}_{\mathrm{Ham}}(y, y') = \frac{1}{2}\left[\sum_{h=1}^{n} \mathbf{1}(y(h) \neq y'(h)) + \sum_{d=1}^{n} \mathbf{1}(y(d) \neq y'(d))\right]$. Again since a WP rule is discontinuous in parameters $\mathbf{W}$, the above objective is not continuous in $\mathbf{W}$. We use the SVM framework to formulate a discriminant-based continuous relaxation to this problem. We first note from Theorem 2 that a WP rule essentially solves an optimization problem over the set of *Boolean* stable matchings $\Omega(\succ)$. Thus a WP rule can be seen as a discriminant-based *structured* classifier, where the output space is the set of stable matchings and the discriminant function is $H_{\mathbf{W}}(\succ, y) = \sum_{h=1}^{n} \sum_{d=1}^{n} \lambda_{hd}(\succ) y_{hd}$:

$$f_{\mathrm{WP}}(\succ; \mathbf{W}) \in \operatorname*{argmax}_{y \in \Omega(\succ)} H_{\mathbf{W}}(\succ, y).$$

We can now replace the original distance measure $\mathbb{D}_{\mathrm{Ham}}$ in Eq. (3) with the following continuous surrogate objective:

$$\sum_{r=1}^{L} \max_{y' \in \Omega(\succ^r)} \left\{ \mathbb{D}_{\mathrm{Ham}}(y^r, y') + H_{\mathbf{W}}(\succ^r, y') - H_{\mathbf{W}}(\succ^r, y^r) \right\}.$$

The resulting optimization problem (possibly with an additional regularization term on $\mathbf{W}$) resembles a structural SVM problem [Tsochantaridis *et al.*, 2005]. The problem is convex in $\mathbf{W}$, and can be solved efficiently using the standard cutting-plane solver, provided the above 'max' over the set of stable matchings $\Omega(\succ^r)$ can be computed in polynomial time. For the distance function used $\mathbb{D}_{\mathrm{Ham}}$, this maximization problem can be relaxed into an equivalent linear program (by appealing to Theorem 2) and solved efficiently.

Extensions. The proposed method can also be applied to weak preference orders. In this case, the result of [Roth *et al.*, 1993] does not apply, and a linear optimization over the matching polytope is not guaranteed to yield an integral matching. Whenever this happens, we break ties arbitrarily, and compute a WP rule on the resulting strict preferences. Our method also extends to many-to-one matching problems

using the polytope characterization of [Baïou and Balinski, 2000]. An alternate approach that we take in our experiments is to solve a relaxed form of the above SVM problem where the stability constraints are ignored during optimization. We then use the obtained parameters to construct a stable WP rule by setting up a suitable integer program.

5 Experimental Results

5.1 Strategy-Proof Social choice

We generate synthetic preference data for a one-dimensional facility location problem. The goal is to set up a facility in one of m equally-spaced locations (outcomes) on $[0, 1]$ based on agent preferences. The agent preferences are single-peaked and the peaks are drawn uniformly from the space of locations. For this experiment, we adopt a (non-strategy-proof) target rule that associates a weight $w : [0, 1] \to \mathbb{R}_+$ with each location, and outputs the location that has the smallest weighted squared distance from the reported peaks: $g(\succ) \in \operatorname{argmin}_{y \in \Omega} \sum_{i=1}^{n} w(p_i)(y - p_i)^2$ (e.g. this weight could indicate the ease of setting up a facility at a location). The weight function that we consider is $w(z) = e^{-\lambda z}$, where $\lambda \geq 0$ determines the skewness of the weights (higher values of λ leads to higher weights on the left side locations).

We generate 1000 examples, split equally into train and test sets, with the train set used to design a mechanism, and the test set used to estimate its distance from the target rule. We compare the WGM rule obtained using the multiclass SVM style method with three baselines: the GM rule that best approximates the target labels in the train set, obtained by an exhaustive search over pseudo-peaks; the best percentile rule [Sui *et al.*, 2013]; and the best dictatorial rule (a rule that always outputs the peak reported by an a priori fixed agent).

Figure 2(a) plots the average absolute error of the given rule on the test set, as a function of weight parameter λ for 5 agents and 25 locations. Our approach is quite robust to changes in target weights. Moreover, it often performs as good as the best GM rule, and better than the other methods. We also report results for 100 agents and 250 locations. Here we do not include the best GM rule, because an exhaustive search was intractable. We additionally report the accuracy of the non-SP rule in Eq. (2) that is used as a proxy for a WGM rule in the surrogate optimization problem. Clearly, the proxy rule closely mimics the corresponding WGM rule.

We next adopt a target rule that assigns weights to agents rather than outcomes, and as before outputs the outcome with minimum weighted squared distance from the reported peaks. The weight function we use is $w'(s) = e^{-\kappa s/n}$, where $s \in [n]$ is a randomly assigned priority to an agent, and $\kappa \geq 0$ controls the skewness of weights ($\kappa = 0$ implies equal weights on all agents, while a high value leads to a near-dictatorial rule). As seen in Figure 2(b), here our approach performs better than the best GM rule for higher κ values (as the GM class does not include dictatorial rules), and converges to the performance of the best dictatorial rule. The parameters obtained by our method closely match with those in the target rule. For $\kappa = 0$, the parameters have near-equal weights on all agents and large weights on the extreme-left and extreme-right outcomes 0 and m, thus mimicking the median rule. For
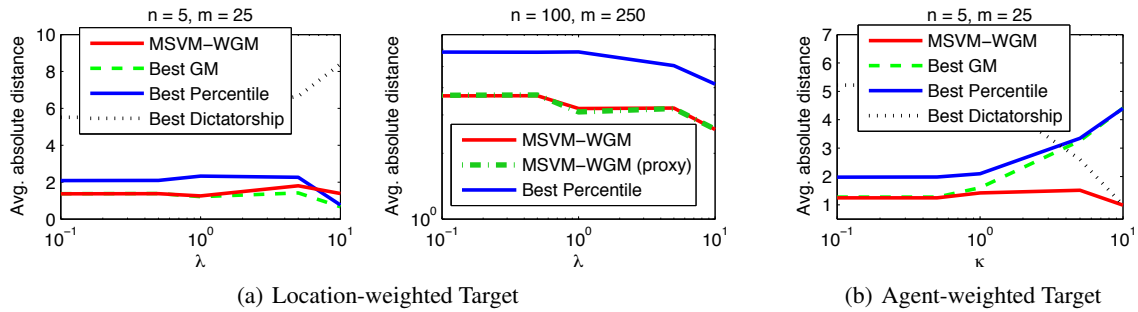
(a) Location-weighted Target	(b) Agent-weighted Target

Figure 2: Experiments on social choice: Plot of test error vs. (a) location weight parameter λ, (b) agent weight parameter κ.

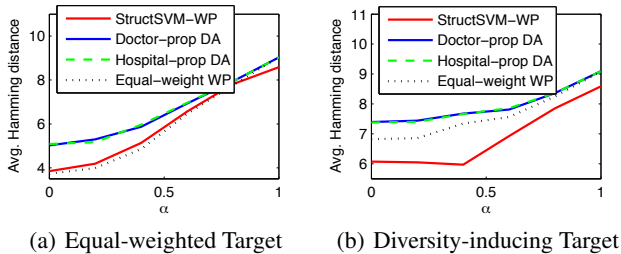|	
(a) Equal-weighted Target	(b) Diversity-inducing Target

Figure 3: Experiments on synthetic data for two-sided matching: Plot of test error vs. preference correlation parameter α.

a large κ, we obtain a large weight on the dictator agent.

5.2 Stable Two-sided Matching

Synthetic data. We consider a one-to-one matching problem between 10 hospitals and 10 doctors, and allow agent preferences to have ties. We give a brief description of the generators and target, and defer additional details to a longer version of the paper. The preferences of a hospital over doctors are generated using a combination of a base score and a hospital-specific score, with a parameter $\alpha \in [0,1]$ determining the weight on the base score, and thus the extent of correlation among agent preferences (higher values imply more concentration). The scores are discretized into equal-length intervals, inducing a weak ordering over the doctors. The preferences of doctors over hospitals are generated similarly.

We consider two (non-stable) target rules. The first is an *equal-weighted* reward-maximizing Hungarian assignment rule that assigns equal weights to all agents; here the reward for matching a hospital and doctor is the sum of the ranks (see definition in Section 4.1) they assign to each other. The second is a *diversity-inducing* Hungarian assignment, where the reward is the sum of the agent ranks, with an additional diversity bonus given to matching of particular groups of hospitals with particular groups of doctors. We generate 1000 examples, divided into equal train-validation-test sets, with the validation set used for parameter tuning in structural SVM.

We compare the structural SVM approach with three stable mechanisms: a hospital-proposing DA rule, a doctor-proposing DA rule, and a WP rule with equal weights on all hospital-doctor pairs ($\mathbf{W} = \mathbf{1}^{3(n \times n)}$). Figure 3 contains plots of test Hamming error vs. correlation parameter α for both

target rules. As the correlation among preferences increases, the error rate increases for all methods; this is because the set of stable matchings for a preference profile becomes smaller as correlation increases (and there is less flexibility to fit the target rule). For the equal-weighted target, our method recovers a WP rule with equal weights on all agents, while performing better than the DA rules for the most part. For the diversity-inducing target, the obtained WP rule performs better than the baselines even in the high correlation regime.

Real-world school-choice data. We next consider a many-to-one matching problem inspired from a real-world school choice data set. The data was obtained from the Wake County, NC Public School System, and contained preferences from 37 schools and 5504 students. Using the data, we estimate a Plackett-Luce preference model for schools and students. We use this model to generate strict preferences for 5 schools and 100 students, and introduce ties to mimic real-world preferences (for students, we truncate their preferences at the top 1 or 2 choices, and for schools, we bin the preference ranks into 20 groups). Each school is assumed to have a capacity of 25. We consider the following (unstable) target rules: the student-proposing Boston mechanism (BM), an equal-weighted Hungarian rule (EH), a Hungarian rule with higher weights to a randomly chosen $1/3^{\text{rd}}$ of the students, assumed to have minority status (MH), and the diversity-inducing Hungarian (DH) rule mentioned above. As seen in Table 1, the proposed method performs better than the baselines in most cases.

Table 1: Experiments motivated by real-world school choice on 5 schools, 100 students, with a quota of 25/school. We use train-test sets of size 200 and report the test (Hamming) error.

	BM	EH	MH	DH
StructSVM-WP	**27.2**	36.0	**22.1**	**47.7**
Student-proposing DA	28.4	46.5	33.4	57.3
Equal-weight WP	29.7	**34.3**	23.0	53.2

References

[Baïou and Balinski, 2000] M. Baïou and M. Balinski. The stable admissions polytope. *Mathematical programming*, 87(3):427–439, 2000.

[Black, 1948] D. Black. On the rationale of group decision-making. *The Journal of Political Economy*, pages 23–34, 1948.

[Cole and Roughgarden, 2014] R. Cole and T. Roughgarden. The sample complexity of revenue maximization. In *STOC*, pages 243–252, 2014.

[Conitzer and Sandholm, 2002] V. Conitzer and T. Sandholm. Complexity of mechanism design. In *UAI*, 2002.

[Conitzer and Sandholm, 2004] V. Conitzer and T. Sandholm. An algorithm for automatically designing deterministic mechanisms without payments. In *AAMAS*, 2004.

[Crammer and Singer, 2002] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:265–292, 2002.

[Dur et al., 2015] U.M. Dur, R.G. Hammond, and T. Morrill. Identifying the harm of manipulable school-choice mechanisms. Technical report, 2015.

[Dütting et al., 2015] P. Dütting, F.A. Fischer, P. Jirapinyo, J.K. Lai, B. Lubin, and D.C. Parkes. Payment rules through discriminant-based classifiers. *ACM Trans. Economics and Comput.*, 3(1):5, 2015.

[Gale and Shapley, 1962] D. Gale and L.S. Shapley. College admissions and the stability of marriage. *American mathematical monthly*, pages 9–15, 1962.

[Gibbard, 1973] A. Gibbard. Manipulation of voting schemes: A general result. *Econometrica: journal of the Econometric Society*, pages 587–601, 1973.

[Guo and Conitzer, 2010] M. Guo and V. Conitzer. Computationally feasible automated mechanism design: General approach and case studies. In *AAAI*, 2010.

[Manlove, 2013] D.F. Manlove. *Algorithmics of matching under preferences*, volume 2. World Scientific, 2013.

[Moulin, 1980] H. Moulin. On strategy-proofness and single peakedness. *Public Choice*, 35(4):437–455, 1980.

[Procaccia and Tennenholtz, 2009] A.D. Procaccia and M. Tennenholtz. Approximate mechanism design without money. In *EC*, 2009.

[Procaccia et al., 2009] A.D. Procaccia, A. Zohar, Y. Peleg, and J.S. Rosenschein. The learnability of voting rules. *Artificial Intelligence*, 173:1133–1149, 2009.

[Roth et al., 1993] A.E. Roth, U.G. Rothblum, and J.H. Vande Vate. Stable matchings, optimal assignments, and linear programming. *Mathematics of Operations Research*, 18(4):803–828, 1993.

[Satterthwaite, 1975] M.A. Satterthwaite. Strategy-proofness and arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of economic theory*, 10(2):187–217, 1975.

[Sui et al., 2013] X. Sui, C. Boutilier, and T. Sandholm. Analysis and optimization of multi-dimensional percentile mechanisms. In *IJCAI*, 2013.

[Tsochantaridis et al., 2005] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6:1453–1484, 2005.