Learning Acceptable Lotteries via Queries: Minimizing Aggregated Violation Distances

Nicholas Teh
University of Oxford
nicholas.teh@cs.ox.ac.uk

Paul W. Goldberg
University of Oxford
paul.goldberg@cs.ox.ac.uk

Abstract

When deploying an AI system, a developer can randomize over several candidate versions, sending each request to a version chosen at random, to balance the requirements of multiple stakeholders. We study a setting in which each of n stakeholders reveals only whether a proposed lottery over m candidates is acceptable, and each stakeholder's acceptable lotteries lie on one side of an unknown hyperplane. While previous work sought a lottery that every stakeholder accepts, such a lottery need not exist when, for instance, safety and helpfulness requirements are in conflict. We therefore minimize a power mean of the stakeholders' violations; since binary answers reveal neither the size of a violation nor how to compare violations across stakeholders, we measure each violation by a normalized distance to the stakeholder's acceptance boundary. We characterize exactly when accept/reject answers determine these distances. When they do, $\mathcal{O}(nm \log(1/\varepsilon))$ queries suffice to learn them, where ε is the precision of the utilities, and this bound is tight; when they do not, a natural worst-case approach is optimal up to a constant factor, even compared with methods that make unlimited queries. For limited query budgets, we give a sampling algorithm whose number of queries for a given accuracy does not depend on n, and show that a deterministic algorithm cannot guarantee a small loss without querying almost all stakeholders. Synthetic experiments illustrate the resulting tradeoff between querying many stakeholders and learning each of them accurately.

1 Introduction

Making a language model less likely to give harmful replies tends to make it refuse more harmless requests [Bai et al., 2022, Röttger et al., 2024], and different parties care about these two kinds of error to different degrees. Suppose that a developer has a *cautious* version of a chatbot, which produces a harmful reply on 0.1% of risky prompts but refuses 15% of harmless requests, and a *permissive* version, whose rates are 0.9% and 3%, respectively. Rather than committing to one version, the developer can route each conversation to a version chosen at random, as is routine in A/B testing [Kohavi et al., 2020]. By linearity of expectation, routing 55% of the conversations to the cautious version yields rates of 0.46% and 9.6%, which meet both a safety team's limit of 0.5% on harmful replies and a customer's limit of 10% on refusals, even though neither version meets both limits on its own. Now suppose that a regulator also caps harmful replies at 0.3%. Then at least 75% of the conversations must go to the cautious version, whereas the customer's limit allows at most $7/12$ of them to go there, so no mixture satisfies everyone. Yet the developer still has to deploy something. How should it choose a mixture when every mixture leaves some requirement unmet?

We formalize such decisions using *lotteries* over a finite set of alternatives (here, the versions of the system). Following the standard expected utility model [von Neumann and Morgenstern, 1944, Fishburn, 1982], each stakeholder, or *agent*, has an unknown score for each alternative and an unknown threshold, and accepts a lottery exactly when its expected score meets the threshold;

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

equivalently, its acceptable lotteries lie on one side of a hyperplane. For requirements on average rates, such as that of harmful replies, this linear model is exact, since the rate of a mixture is the corresponding mixture of rates.

In practice, however, the developer observes neither the scores nor the thresholds. Customers may be unwilling to share their test data, a regulator's requirements may exist only in legal text [European Union, 2024], and a safety team's thresholds may be confidential or hard to articulate. What a stakeholder can readily do is evaluate a concrete proposal and say whether it passes; indeed, evaluations of this kind are central to current AI safety practice, where they determine whether a system may be released [Shevlane et al., 2023, Weidinger et al., 2023]. Following Angluin [1988], we call a question of the form "Would agent i accept lottery $\mathbf{x}$?" a *membership query*. Choo et al. [2026b] showed how to use such queries to decide whether some lottery is acceptable to every agent and, if so, to find one. As our example illustrates, however, requirements may conflict, in which case no lottery is unanimously acceptable, and keeping the current system does not avoid the choice (it is itself one of the alternatives). Which lottery should be chosen when unanimity is impossible, and what can be learned about this choice from accept/reject answers alone?

A natural first idea is to minimize the number of agents who reject the chosen lottery; Teh and Goldberg [2026] studied this objective in the same query model. However, the number of rejections does not distinguish between lotteries that violate the same requirement by different amounts. In our example, deploying only the cautious version and routing 75% of the conversations to it are each rejected by the customer alone, but the former misses the customer's limit by five percentage points and the latter by only two. When some requirements concern safety, how severely a requirement is violated matters as much as whether it is violated. Measuring severity is subtle, since the raw shortfall from a threshold has no meaning on its own. Indeed, multiplying an agent's scores and threshold by the same positive constant, or adding the same constant to all of them, leaves every accept/reject answer unchanged but changes the shortfall. Hence raw shortfalls are neither determined by the answers nor comparable across agents (a shortfall of 0.1 percentage points in harmful replies and one of 0.1 points in refusals need not be equally serious).

Instead, we measure violation in the space of lotteries. After normalizing the acceptance boundary of each agent, we take the Euclidean distance, within the affine hull of the simplex, from a rejected lottery to this boundary; the distance is zero for accepted lotteries and does not change under the transformations above. We then aggregate the distances of all agents by a power mean with parameter $p \geq 1$: the case $p = 1$ measures the average violation, while larger values of p put more weight on the largest violations, which is appropriate when a large violation of a single requirement is unacceptable. Once the acceptance boundaries are known, minimizing this objective is a convex problem. The central question is therefore whether membership queries reveal enough about the boundaries. Classifying lotteries correctly does not suffice, since the objective depends on how far a rejected lottery lies from each normalized boundary.

1.1 Our Results

In this paper, we consider n agents and m alternatives. Each agent accepts the lotteries on one side of an unknown hyperplane, and the learner can only ask membership queries. As is standard in query learning, we assume that answers are exact. This assumption is most appropriate when acceptance is decided by an automated evaluation, such as a benchmark with a fixed passing threshold, and is an idealized baseline for human reviewers, who may be uncertain near their boundaries; Section 4.3 discusses noisy answers. We describe the model formally in Section 2.

In Section 3, we begin by asking what an unlimited number of queries can reveal. We show that the answer depends only on how an agent treats the pure alternatives (Theorem 3.1). If she strictly accepts some pure alternative or rejects exactly one, her normalized boundary is determined by the answers, and $\mathcal{O}(m \log(1/\varepsilon))$ queries suffice to find it, where ε is the precision of the utilities; this bound is tight in general. Otherwise, the agent is *distance-underdetermined*: the answers determine her acceptable set exactly, but not how fast her violation grows away from it (Figure 1). For such an agent, the largest violation consistent with all answers is the distance to her acceptable set (Lemma 3.2). On the positive side, minimizing the resulting worst-case objective incurs additive loss at most $\sqrt{2}(|B|/n)^{1/p}$, where B is the set of distance-underdetermined agents (Theorem 3.3). On the negative side, no algorithm can guarantee a loss below $(|B|/n)^{1/p}/(12\sqrt{2})$, even with unlimited queries and randomization (Theorem 3.4). We complement these results with two consequences of convex duality that hold once

the boundaries are known: an optimal lottery gives each violated agent a weight that increases with her violation and uses only alternatives that maximize the weighted score (Theorem 3.6), and some optimal lottery uses at most $n + 1$ alternatives (Theorem 3.7).

Next, in Section 4, we turn to limited query budgets. Our algorithm samples k agents, learns their boundaries, and minimizes the objective of the sample. We show that its loss splits into a boundary-learning term and a sampling term that depends on neither n nor m (Theorem 4.2). When every agent strictly accepts some pure alternative, the algorithm achieves additive error α with probability at least $1 - \delta$ using $\mathcal{O}(m \log(1/\varepsilon) \log(2/\delta)\, \alpha^{-2p})$ queries for each fixed p, independently of n (Corollary 4.3). On the other hand, a deterministic algorithm that queries only $K < n$ agents can suffer additive loss $\Omega(((n - K)/n)^{1/p})$ (Theorem 4.4), so it must query almost all agents in order to guarantee a small loss. We also extend our main guarantees to answers that are flipped independently at random (Proposition 4.5) and to agents with different weights.

Finally, in Section 5, we use synthetic experiments as controlled checks of the theory. They illustrate how a fixed budget should be split between querying many agents and learning each agent accurately, and how larger values of p move the optimum toward reducing the largest violations.

1.2 Related Work

The paper most closely related to ours is that of Choo et al. [2026b], who introduced this query model for lotteries and showed how to decide whether some lottery is acceptable to all agents and, if so, how to find one. For this purpose, it suffices to learn the acceptable set of each agent. A major difference between our work and theirs is that we also compare lotteries that some agents reject, which requires the normalized distance to each boundary; accept/reject answers do not always determine this distance, even when they determine the acceptable set. For learning acceptable sets, we use their single-agent procedure (Lemma 4.1). In a companion paper, Teh and Goldberg [2026] studied the same query model with the objective of minimizing the number of rejecting agents, which depends only on the acceptable sets but is APX-hard to optimize even when all boundaries are known; by contrast, our objective is convex once the boundaries are known. The two objectives are complementary: ours distinguishes slight violations from severe ones, whereas counting rejections is appropriate for requirements that are simply met or not. Our model combines the expected utility view of lotteries [von Neumann and Morgenstern, 1944] with a threshold view of acceptability [Simon, 1955], but our results depend only on the induced accept/reject answers; Appendix A.1 discusses this assumption.

Our motivation is related to recent work on aligning AI systems with the diverse preferences of the people they affect. Sorensen et al. [2024] argued that AI systems should represent a plurality of human values rather than a single aggregate, and Conitzer et al. [2024] argued that social choice theory should guide how diverse human feedback is combined. In a similar vein, Neoh et al. [2026b] selected a small ensemble of AI systems using binary or pairwise feedback while saving on costly evaluations. Our model is narrower: agents only reveal whether a randomized deployment is acceptable to them, and the developer must still choose a deployment when none is acceptable to all.

Randomization over outcomes is central in probabilistic social choice and randomized mechanism design [Gibbard, 1977, Brandt, 2017, Brandt et al., 2016]. Closer in spirit are fair mixing and models with dichotomous preferences, where agents approve or disapprove deterministic outcomes and the decision is a lottery over them [Bogomolnaia et al., 2005, Aziz et al., 2020], as well as portioning and budget aggregation, where a divisible resource is split among projects based on voters' ordinal or cardinal reports [Freeman et al., 2021, Airiau et al., 2023, Freeman and Schmidt-Kraepelin, 2024, Elkind et al., 2026]; we refer to the survey by Suksompong and Teh [2026] for a recent overview of voting over divisible resources. These works typically assume richer preference reports and study axiomatic or strategic properties of aggregation rules, whereas our learner only receives accept/reject answers and must recover enough geometry to compare violations.

Lotteries also appear in constrained machine learning, where randomized classifiers are used to satisfy fairness, error-rate, or resource constraints in expectation [Agarwal et al., 2018, Cotter et al., 2019, Zafar et al., 2019, Agarwal and Deshpande, 2022]. There, the constraints are usually specified by the designer or estimated from labeled data, whereas each of our constraints belongs to an agent and can only be accessed through accept/reject answers.

Methodologically, our work lies between preference elicitation and query learning [Blum et al., 2004, Braziunas and Boutilier, 2008]. Learning with membership queries goes back to Angluin [1988] and is a standard active learning setting [Settles, 2009]; recent work has studied membership query algorithms for halfspaces [Alabdulmohsin et al., 2015, Kelner, 2020, Diakonikolas et al., 2024]. These works aim for classification accuracy, whereas optimizing distances to acceptance boundaries requires the normalized hyperplanes themselves, which binary labels may not reveal.

Finally, once unanimity is impossible, our objective is related to soft constraints, least-violation optimization, and inequality-aware aggregation [Boyd and Vandenberghe, 2004, Dai and Zhang, 2021, Chen and Hooker, 2023, Atkinson, 1970]. Power means interpolate between average-case and worst-case criteria and have recently been studied as learnable social welfare functions [Pardeshi et al., 2024]; we apply them to normalized distances, so that p interpolates between the average and the maximum violation. Appendix A.2 discusses further connections.

2 Preliminaries

Let $N = [n]$ be the set of agents and $S = \{s_1, \dots, s_m\}$ the set of alternatives, where $m \geq 2$ and we write $[z] := \{1, \dots, z\}$ for each positive integer z. A lottery is a vector $\mathbf{x} \in \Delta(S) := \{\mathbf{x} \in \mathbb{R}^m_{\geq 0} : \sum_{j=1}^m x_j = 1\}$, where x_j is the probability of alternative s_j, and $\mathbf{e}_j$ denotes the pure lottery on s_j. We write $\langle \cdot, \cdot \rangle$ and $\| \cdot \|$ for the Euclidean inner product and norm, and $\operatorname{dist}(\mathbf{x}, Y) := \inf_{\mathbf{y} \in Y} \|\mathbf{x} - \mathbf{y}\|$ for a set $Y \subseteq \mathbb{R}^m$. Note that $\|\mathbf{x}\| \leq 1$ for every lottery $\mathbf{x}$, and any two lotteries are at distance at most $\sqrt{2}$.

Each agent $i \in N$ has a utility vector $u_i \in [0,1]^m$ and a threshold $\tau_i \in (0,1]$, and accepts a lottery $\mathbf{x}$ if and only if $U_i(\mathbf{x}) := \langle u_i, \mathbf{x} \rangle \geq \tau_i$. Her *acceptable set* is $A_i := \{\mathbf{x} \in \Delta(S) : U_i(\mathbf{x}) \geq \tau_i\}$, and we let $M_i := \{j \in [m] : U_i(\mathbf{e}_j) \geq \tau_i\}$ be the set of indices of the pure alternatives that she accepts. We assume that each agent accepts some lottery and rejects some lottery; since a linear function on $\Delta(S)$ attains its maximum and minimum at pure lotteries, this means that $\varnothing \neq M_i \neq [m]$. Appendix A.1 shows that every such linear threshold rule over lotteries can be written in this form.

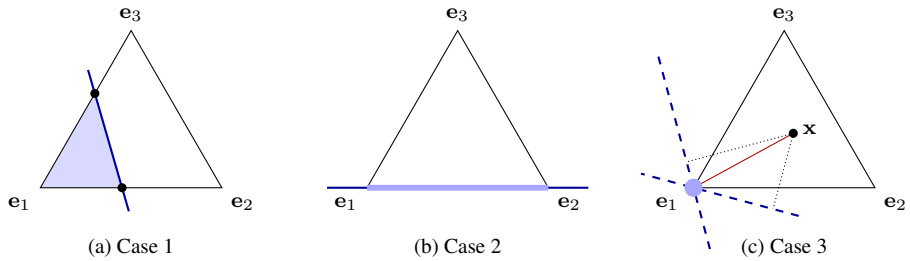
Membership queries. A membership query asks whether agent i accepts lottery $\mathbf{x}$; its answer is $\texttt{Query}(i, \mathbf{x}) = 1$ if $U_i(\mathbf{x}) \geq \tau_i$ and $\texttt{Query}(i, \mathbf{x}) = 0$ otherwise. Queries may be adaptive. Let T_i be the number of queries to agent i, $Q := \sum_{i \in N} T_i$ the total number of queries, and $K := |\{i \in N : T_i > 0\}|$ the number of agents who receive at least one query. This is the standard membership query model [Angluin, 1988, Kearns and Vazirani, 1994]. As shown by Choo et al. [2026b], exact recovery with finitely many queries requires the parameters to have finite precision. Following them, we call an instance *ε-quantized*, where $\varepsilon \in (0, 1/2]$ and $1/\varepsilon$ is an integer, if $u_i(s_j) \in \{0, \varepsilon, 2\varepsilon, \dots, 1\}$ and $\tau_i \in \{\varepsilon, 2\varepsilon, \dots, 1\}$ for all $i \in N$ and $j \in [m]$. We use this assumption only in our query bounds.

Normalized violations. Accept/reject answers do not change if we add the same constant to the utilities and the threshold of an agent (since the probabilities of a lottery sum to one), or if we multiply them by the same positive number. We remove these two degrees of freedom before comparing agents. Let $\bar{u}_i := \frac{1}{m} \sum_{j=1}^m u_i(s_j)$ and $w_i := u_i - \bar{u}_i \mathbf{1}$, where $\mathbf{1}$ is the all-ones vector; $w_i \neq 0$ because $\varnothing \neq M_i \neq [m]$. The *normalized acceptance boundary* of agent i is the pair $(\hat{u}_i, \hat{\tau}_i) := (w_i/\|w_i\|, (\tau_i - \bar{u}_i)/\|w_i\|)$. It satisfies $\|\hat{u}_i\| = 1$ and $\sum_{j=1}^m \hat{u}_i(s_j) = 0$, and agent i accepts $\mathbf{x}$ if and only if $\langle \hat{u}_i, \mathbf{x} \rangle \geq \hat{\tau}_i$. The *violation* of agent i at $\mathbf{x}$ is

$$v_i(\mathbf{x}) := \max\{0, \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x} \rangle\} = \frac{\max\{0, \tau_i - U_i(\mathbf{x})\}}{\|w_i\|},$$

so $v_i(\mathbf{x}) = 0$ exactly when agent i accepts $\mathbf{x}$. Geometrically, let $H_i := \{\mathbf{y} \in \mathbb{R}^m : \sum_{j=1}^m y_j = 1 \text{ and } \langle \hat{u}_i, \mathbf{y} \rangle = \hat{\tau}_i\}$ be the acceptance boundary of agent i within the affine hull of $\Delta(S)$. Since $\hat{u}_i$ is a unit vector whose coordinates sum to zero, $v_i(\mathbf{x}) = \operatorname{dist}(\mathbf{x}, H_i)$ for every lottery $\mathbf{x}$ that agent i rejects. This is the distance to a hyperplane, which can be smaller than $\operatorname{dist}(\mathbf{x}, A_i)$. Moreover, since H_i contains a lottery (on the segment between a lottery that agent i accepts and one that she rejects), we have $v_i(\mathbf{x}) \leq \sqrt{2}$ for every lottery $\mathbf{x}$. Appendix A.3 gives the details and an example.

We aggregate violations by the power mean $V_p(\mathbf{x}) := \left(\frac{1}{n} \sum_{i \in N} v_i(\mathbf{x})^p\right)^{1/p}$ for $p \in [1, \infty)$, and let $V_\infty(\mathbf{x}) := \max_{i \in N} v_i(\mathbf{x})$. For finite p, it is often convenient to minimize $F(\mathbf{x}) := V_p(\mathbf{x})^p$ instead,



(a) Case 1 (b) Case 2 (c) Case 3

Figure 1: The three cases of Section 3.1 for $m = 3$, drawn in the plane containing the simplex. Acceptable sets are light blue and boundaries dark blue. In (a), searching along the edges from $\mathbf{e}_1$ finds two points on the boundary. In (c), the agent accepts only $\mathbf{e}_1$, and each dashed line gives the same answer to every query; the distance from $\mathbf{x}$ to the boundary (dotted) depends on the line, and its largest value is the distance from $\mathbf{x}$ to $\mathbf{e}_1$ (red), as shown in Lemma 3.2.

which has the same minimizers. Since each v_i is the maximum of two affine functions, it is convex, and so are F and V_p. Hence, once the normalized boundaries are known, minimizing V_p over $\Delta(S)$ is a convex optimization problem.

3 Learning from Unlimited Queries

We begin by assuming that every agent can be queried as often as needed.

3.1 What Membership Queries Reveal

Consider a single agent i. Querying the m pure lotteries reveals the set M_i. Since U_i is linear, agent i accepts some lottery with a positive margin if and only if $U_i(\mathbf{e}_j) > \tau_i$ for some $j \in [m]$. This leads to three cases, illustrated in Figure 1 for $m = 3$.

1. Agent i strictly accepts some pure alternative. Then searching along edges of the simplex between accepted and rejected pure lotteries yields $m - 1$ affinely independent points on her boundary, which determine her normalized boundary.
2. Every pure alternative that agent i accepts lies on her boundary, and she rejects exactly one pure alternative. Then the $m - 1$ accepted pure lotteries span her boundary, which is again determined.
3. Every pure alternative that agent i accepts lies on her boundary, and she rejects at least two pure alternatives. Then her acceptable set A_i is the face of the simplex spanned by the accepted pure lotteries. Many different normalized boundaries give this acceptable set, and hence the same answer to every query, but different violations outside A_i.

We call the agents in case 3 *distance-underdetermined*, and let $B := \{i \in N : U_i(\mathbf{e}_j) = \tau_i$ for all $j \in M_i$, and $|M_i| \leq m - 2\}$ be the set of such agents. In the ambiguity statements below, consistency ranges over all profiles in our model, without the quantization restriction used for finite-query bounds.

Theorem 3.1. *There is a deterministic algorithm that, on every ε-quantized instance, makes $\mathcal{O}(nm \log(1/\varepsilon))$ membership queries and finds the normalized boundary $(\hat{u}_i, \hat{\tau}_i)$ of every agent $i \notin B$ and the acceptable set A_i of every agent $i \in B$. For each agent $i \in B$, infinitely many normalized boundaries give the same answer as agent i to every membership query. Moreover, if $m \geq 3$, every deterministic algorithm that finds the normalized boundaries of all agents makes $\Omega(nm \log(1/\varepsilon))$ queries on some ε-quantized instance with $B = \varnothing$.*

In case 1, our algorithm follows the single-agent procedure of Choo et al. [2026b], who used it to learn acceptable sets. Our contribution in Theorem 3.1 is to determine exactly when the answers also fix the normalized boundary (a question that does not arise when one only seeks a unanimously acceptable lottery). The lower bound follows by counting: there are $(1/\varepsilon)^{\Omega(m)}$ ε-quantized agents with different normalized boundaries, and each query has only two possible answers. The lower bound concerns recovering the boundaries; whether minimizing V_p exactly requires as many queries is open.

3.2 Agents with Undetermined Distances

Since queries cannot reveal how fast the violation of an agent in B grows outside her acceptable set, a natural approach is to plan for the largest violation that is consistent with her answers. Our next lemma shows that this largest violation has a simple form. Intuitively, a consistent boundary can be rotated around the face A_i, and the violation at $\mathbf{x}$ is largest when the boundary is perpendicular to the segment from $\mathbf{x}$ to its closest point in A_i, as in Figure 1(c).

Lemma 3.2. *Let $i \in B$. The violations of the normalized boundaries that give the same answer as agent i to every membership query are exactly the functions $\mathbf{x} \mapsto \sum_{j \notin M_i} \delta_j x_j$ on $\Delta(S)$, where $(\delta_j)_{j \notin M_i}$ ranges over all positive numbers with $\sum_{j \notin M_i} \delta_j^2 - \frac{1}{m}\big(\sum_{j \notin M_i} \delta_j\big)^2 = 1$. Moreover, for every $\mathbf{x} \in \Delta(S)$, the largest violation at $\mathbf{x}$ among these boundaries is $\operatorname{dist}(\mathbf{x}, A_i)$, and $\max_{\mathbf{x} \in \Delta(S)} \operatorname{dist}(\mathbf{x}, A_i) = \sqrt{1 + 1/|M_i|}$.*

Replacing the unknown violation of each agent in B by its largest possible value leads to the objective

$$\bar{F}(\mathbf{x}) := \frac{1}{n} \sum_{i \notin B} v_i(\mathbf{x})^p + \frac{1}{n} \sum_{i \in B} \operatorname{dist}(\mathbf{x}, A_i)^p.$$

By Lemma 3.2, $\bar{F}(\mathbf{x})$ is exactly the largest value of $V_p(\mathbf{x})^p$ over all profiles that give the same answers as the true profile to every query. Moreover, $\bar{F}$ is convex, and it is known after the queries of Theorem 3.1. The next theorem bounds the loss from minimizing $\bar{F}$ instead of V_p. Since $\bar{F} \geq F$ and the two functions only differ in the terms of the agents in B, this loss is controlled by these terms at an optimal lottery.

Theorem 3.3. *Fix $p \in [1, \infty)$, let $\bar{\mathbf{x}}$ be a minimizer of $\bar{F}$ over $\Delta(S)$, and let $\mathbf{x}^*$ be a minimizer of V_p over $\Delta(S)$. Then*

$$V_p(\bar{\mathbf{x}})^p - V_p(\mathbf{x}^*)^p \leq \frac{1}{n} \sum_{i \in B} \left(\operatorname{dist}(\mathbf{x}^*, A_i)^p - v_i(\mathbf{x}^*)^p \right).$$

In particular, $V_p(\bar{\mathbf{x}}) \leq V_p(\mathbf{x}^*) + \sqrt{2}\,(|B|/n)^{1/p}$.

The first bound depends on the instance. It shows that $\bar{\mathbf{x}}$ is optimal whenever some minimizer of V_p is accepted by every agent in B, and that the loss is small whenever such a minimizer is close to their acceptable sets. The second bound holds for every instance; by Lemma 3.2, its factor $\sqrt{2}$ can be replaced by $\max_{i \in B} \sqrt{1 + 1/|M_i|}$, which is close to 1 when every agent in B accepts many pure alternatives. Our next result shows that the dependence on $(|B|/n)^{1/p}$ cannot be avoided by any method that only uses membership queries.

Theorem 3.4. *Fix $p \in [1, \infty)$, and let n and b be integers such that b is even and $2 \leq b \leq n$. There exist two profiles with $m = 3$ and $|B| = b$, in which all utilities lie in $\{0, 1/2, 1\}$ and all thresholds are 1, such that the two profiles give the same answer to every membership query and, for every lottery $\mathbf{x}$, the additive losses $V_p(\mathbf{x}) - \min_{\mathbf{y} \in \Delta(S)} V_p(\mathbf{y})$ in the two profiles sum to at least $\frac{1}{6\sqrt{2}}(b/n)^{1/p}$. Consequently, every algorithm that only uses membership queries, even with unlimited queries and randomization, has expected additive loss at least $\frac{1}{12\sqrt{2}}(b/n)^{1/p}$ in one of the two profiles.*

Together with Theorem 3.3, this determines the worst-case loss caused by distance-underdetermined agents up to a factor of 24 when $|B|$ is even. The profiles in the proof are of the kind shown in Figure 1(c): in both profiles, one agent accepts only s_1 and another accepts only s_2, but the rates at which their violations grow toward the other alternatives are swapped, so the optimal lottery leans toward s_1 in one profile and toward s_2 in the other. Since the two profiles are mirror images under swapping s_1 and s_2, a lottery that is nearly optimal in one profile is far from optimal in the other.

If $B = \varnothing$, then $\bar{F} = F$, and Theorems 3.1 and 3.3 yield exact optimization.

Corollary 3.5. *Fix $p \in [1, \infty)$. There is a deterministic algorithm that, on every ε-quantized instance with $B = \varnothing$ (for example, one in which every agent strictly accepts some pure alternative) makes $\mathcal{O}(nm \log(1/\varepsilon))$ queries and outputs a minimizer of V_p over $\Delta(S)$.*

3.3 Structure of Optimal Lotteries

We now assume that the normalized boundaries are known and describe optimal lotteries. While both results in this subsection follow from standard convex analysis, we include them because they explain

Algorithm 1 Budgeted query strategy for fixed p

Input: Total query budget $Q \in \mathbb{Z}_{>0}$, confidence parameter $\delta \in (0,1)$, and $p \in [1,\infty)$.
Output: A lottery $\hat{\mathbf{x}} \in \Delta(S)$.

1: Choose $k \in \mathbb{Z}_{>0}$ and $\eta \in [0, \frac{1}{2})$, possibly depending on Q, δ, and p, such that $kT(\eta) \le Q$.
2: Sample $i_1, \dots, i_k$ independently and uniformly at random from N.
3: **for** $t = 1$ to k **do**
4: $\quad(\tilde{u}_t, \tilde{\tau}_t) \leftarrow$ `LearnHalfspace`(i_t, η), and let $\tilde{v}_t(\mathbf{x}) := \max\{0, \tilde{\tau}_t - \langle \tilde{u}_t, \mathbf{x}\rangle\}$ for $\mathbf{x} \in \Delta(S)$.
5: **end for**
6: **return** any $\hat{\mathbf{x}} \in \arg\min_{\mathbf{x}\in\Delta(S)} \frac{1}{k}\sum_{t=1}^{k} \tilde{v}_t(\mathbf{x})^p$.

how p determines which agents drive the optimum and show that optimal lotteries can be simple. For $1 < p < \infty$, the objective F is differentiable, and its optimality conditions take the following form.

Theorem 3.6. *Fix $p \in (1,\infty)$. A lottery $\mathbf{x}^*$ minimizes V_p over $\Delta(S)$ if and only if every alternative s_j with $x^*_j > 0$ maximizes the weighted score $\sum_{i\in N} v_i(\mathbf{x}^*)^{p-1}\,\hat{u}_i(s_j)$ over all alternatives.*

In other words, at an optimum, each agent receives a weight $v_i(\mathbf{x}^*)^{p-1}$ that increases with her violation, agents who accept $\mathbf{x}^*$ receive weight zero, and the lottery only uses alternatives that are best for the resulting weighted sum of normalized utilities. Larger values of p give relatively more weight to agents with larger violations. Appendix B.6 shows that these weights, multiplied by p, form the unique optimal solution of a dual problem. An analogous description holds for $p = 1$ and $p = \infty$, with weights given by an optimal dual solution: for $p = 1$, every violated agent has weight 1, and for $p = \infty$, only the most violated agents can have positive weight (Appendix B.7).

Theorem 3.6 identifies which alternatives an optimal lottery may use, but not how many it needs. We say that a lottery $\mathbf{x}$ *uses* alternative s_j if $x_j > 0$. Since $V_p(\mathbf{x})$ depends on $\mathbf{x}$ only through the n numbers $\langle \hat{u}_i, \mathbf{x}\rangle$, Carathéodory's theorem yields the following result.

Theorem 3.7. *Fix $p \in [1,\infty)$. For every minimizer $\mathbf{x}^*$ of V_p over $\Delta(S)$, there is a lottery $\bar{\mathbf{x}}$ that uses at most $n+1$ alternatives, uses only alternatives that $\mathbf{x}^*$ uses, and satisfies $v_i(\bar{\mathbf{x}}) = v_i(\mathbf{x}^*)$ for every $i \in N$. In particular, $\bar{\mathbf{x}}$ also minimizes V_p.*

Allowing a small error reduces the number of alternatives further: Appendix B.9 shows that some lottery that uses at most $\lceil \alpha^{-2} \rceil$ alternatives is within additive α of the optimum, and that for $p \ge 2$, the Frank–Wolfe method approaches the optimum while its T-th iterate uses at most $T+1$ alternatives.

4 Limited Query Budgets

We now ask what can be guaranteed with a total budget of Q queries. We focus on instances in which no agent is distance-underdetermined, so that the difficulty lies in how to spend the queries rather than in the ambiguity of Section 3.2. A limited budget creates a breadth-depth tradeoff: querying more agents gives a better picture of the population, but leaves fewer queries for each of them. Relative to the optimum, three kinds of error arise: *boundary-learning error*, since each queried agent's boundary is learned only approximately; *sampling error*, since we optimize over a sample of agents; and the error from agents who receive no queries. The results in this section separate these effects.

4.1 A Sampling Algorithm

Algorithm 1 samples k agents uniformly at random with replacement, learns the normalized boundary of each sampled agent up to accuracy η, and minimizes the objective of the sample. It may use any single-agent routine `LearnHalfspace`(i,η) that, for every agent i, makes at most $T(\eta)$ queries to agent i and returns a pair $(\tilde{u},\tilde{\tau})$ with $\|\tilde{u}\| = 1$, $\sum_{j=1}^{m} \tilde{u}(s_j) = 0$, $\|\tilde{u} - \hat{u}_i\| \le \eta$, and $|\tilde{\tau} - \hat{\tau}_i| \le \eta$. The parameters k and η are chosen so that $kT(\eta) \le Q$: a larger k gives a more representative sample but forces coarser estimates, while a smaller k gives more accurate estimates of a less representative sample. The last step of the algorithm is a convex optimization problem.

When an agent strictly accepts some pure alternative, her boundary can be learned exactly.

Lemma 4.1 (Choo et al., 2026b)**.** *Consider an ε-quantized instance, and let $i \in N$ be an agent with $U_i(\mathbf{e}_j) > \tau_i$ for some $j \in [m]$. There is a deterministic algorithm that makes $\mathcal{O}(m \log(1/\varepsilon))$ queries to agent i and returns $(\hat{u}_i, \hat{\tau}_i)$.*

The procedure of Choo et al. [2026b] returns a linear inequality that defines A_i on $\Delta(S)$. Since agent i is in case 1 of Section 3.1, normalizing this inequality as in Section 2 gives $(\hat{u}_i, \hat{\tau}_i)$; the procedure is also case 1 in the proof of Theorem 3.1. With this routine, we can take $\eta = 0$.

Our main result in this section bounds the loss of Algorithm 1 for any routine with the guarantee above. Its proof compares the true objective F, the objective of the sampled agents with their true boundaries, and the objective that the algorithm minimizes, and bounds their differences uniformly over $\Delta(S)$.

Theorem 4.2. *Algorithm 1 makes at most Q membership queries, and with probability at least $1 - \delta$ over the draw of $i_1, \ldots, i_k$, its output $\hat{\mathbf{x}}$ satisfies*

$$V_p(\hat{\mathbf{x}}) \leq \min_{\mathbf{x} \in \Delta(S)} V_p(\mathbf{x}) + \min\left\{ \left(4p\eta(\sqrt{2} + 2\eta)^{p-1} + 2\rho_k(\delta)\right)^{1/p},\ 4\eta + 2\rho_k(\delta)^{1/p} \right\},$$

where $\rho_k(\delta) := 2p(\sqrt{2})^{p-1}/\sqrt{k} + (\sqrt{2})^p \sqrt{\log(2/\delta)/(2k)}$.

The term $\rho_k(\delta)$ is the sampling error. It depends on neither n nor m; the number of alternatives only enters through the cost $T(\eta)$ of learning one agent. The two expressions in the minimum come from comparing the learned and true objectives at the level of $F = V_p^p$ and of V_p, respectively; the second is smaller when the boundary-learning error dominates. With the exact routine of Lemma 4.1, the boundary-learning error vanishes, and a suitable choice of k gives the following guarantee.

Corollary 4.3. *Fix $p \in [1, \infty)$. There is a constant $C_p > 0$, depending only on p, such that for all $\alpha > 0$ and $\delta \in (0, 1)$, there is a randomized algorithm that, on every ε-quantized instance in which every agent strictly accepts some pure alternative, makes at most $C_p\, m \log(1/\varepsilon)\,(1 + \log(2/\delta))\,\alpha^{-2p}$ queries and returns a lottery $\hat{\mathbf{x}}$ such that $V_p(\hat{\mathbf{x}}) \leq \min_{\mathbf{x} \in \Delta(S)} V_p(\mathbf{x}) + \alpha$ with probability at least $1 - \delta$.*

The number of queries does not depend on n. The exponent $2p$ arises because the sampling error in F decreases at rate $k^{-1/2}$, while an error of α in $V_p = F^{1/p}$ requires an error of order α^p in F.

4.2 Error from Unqueried Agents

Algorithm 1 chooses the queried agents at random, and Theorem 4.2 shows that the loss due to the agents it does not query is captured by the sampling error $\rho_k(\delta)$. Our next result shows that a deterministic algorithm cannot leave agents unqueried in this way, even if it learns the queried agents exactly. The idea is that the preferences of an agent who is never queried can be chosen after the algorithm has fixed its output, so that this output is far from her acceptable set.

Theorem 4.4. *Fix $p \in [1, \infty)$ and $K \in \{0, 1, \ldots, n-1\}$, and consider any deterministic, possibly adaptive, algorithm that queries at most K distinct agents on every profile. Then there is a profile with $m = 3$, $u_i \in \{0, 1\}^3$, and $\tau_i = 1$ for all $i \in N$, on which the algorithm outputs a lottery $\hat{\mathbf{x}}$ with $V_p(\hat{\mathbf{x}}) \geq \min_{\mathbf{x} \in \Delta(S)} V_p(\mathbf{x}) + \sqrt{3/8}\,((n-K)/n)^{1/p}$. The bound also holds with constant $\sqrt{6}/8$ on a profile with $m = 3$, $u_i \in \{0, 1\}^3$, and $\tau_i \in \{\frac{1}{2}, \frac{3}{4}\}$ for all i, in which every agent strictly accepts some pure alternative. In particular, every deterministic algorithm that always outputs a minimizer of V_p must query all n agents on some profile.*

This bound is tight up to a constant factor: if the exact boundaries of K agents are known, minimizing their contribution to F yields additive loss at most $\sqrt{2}((n-K)/n)^{1/p}$ (Proposition C.2). An analogous phenomenon arises for the number of rejecting agents: for $m \geq 3$, Teh and Goldberg [2026] showed that a deterministic algorithm querying at most K distinct agents can be forced to incur $n - K$ more rejections than necessary. Theorem 4.4 does not apply to randomized algorithms such as Algorithm 1, which choose the queried agents at random. For randomized algorithms, we only obtain a weaker statement: an algorithm that always outputs a minimizer of V_p must, for each agent i, query agent i with probability one on some profile (Proposition C.1). Proving a lower bound for randomized algorithms that matches the guarantee of Algorithm 1 remains open.

4.3 Noisy Answers and Weighted Agents

When each answer is wrong independently with probability below $1/2$, repeating each query and taking the majority answer reduces the problem to the case of exact answers.

Proposition 4.5. *Let $\nu \in [0, 1/2)$ and $\delta \in (0, 1)$, and suppose that each answer to a membership query, including each repeated answer to the same query, is flipped independently with probability at most ν. Consider any algorithm that makes at most L queries when answers are exact. Modify it so that it asks each query r times and uses the majority answer, where r is the smallest odd integer that is at least $2\ln(L/\delta)/(1-2\nu)^2$, and so that it stops after L queries, returning an arbitrary output if it has not finished by then. Then the modified algorithm makes at most rL queries, and with probability at least $1-\delta$, every majority answer is correct, so that it behaves exactly as the original algorithm does with exact answers.*

For example, with the routine of Lemma 4.1, Algorithm 1 makes $L = \mathcal{O}(km\log(1/\varepsilon))$ queries. Applying Proposition 4.5 and Theorem 4.2 with $\delta/2$ in place of δ, we obtain the guarantee of Theorem 4.2 with $\rho_k(\delta/2)$ in place of $\rho_k(\delta)$ with probability at least $1-\delta$, using at most rL queries, where $r = \mathcal{O}((1-2\nu)^{-2}\log(2L/\delta))$. Proposition 4.5 does not cover answers that are arbitrary or correlated near the boundary, nor strategic answers. In such cases, exact recovery is impossible without further assumptions, such as a margin around the boundary, but Theorem 4.2 still applies to any routine that learns each boundary up to accuracy η.

Our model gives every agent the same weight, which avoids deciding whose acceptability should count more. In some applications, however, agents have different entitlements, as in weighted fair division [Suksompong and Teh, 2022, 2023, Montanari et al., 2025, Teh, 2026, Suksompong, 2025]. For finite p, Theorems 3.3, 3.6, and 4.2 extend to the weighted objective $\left(\sum_{i\in N} a_i v_i(\mathbf{x})^p\right)^{1/p}$, where $a_i \geq 0$ and $\sum_{i\in N} a_i = 1$. In Theorem 3.3, the term $(|B|/n)^{1/p}$ becomes $(\sum_{i\in B} a_i)^{1/p}$; in Theorem 3.6, the weight of agent i becomes $a_i v_i(\mathbf{x}^*)^{p-1}$; and if Algorithm 1 samples each agent i with probability a_i, Theorem 4.2 holds unchanged (Appendix C.7).

5 Experiments

We use synthetic experiments to illustrate two effects predicted by the theory; they are not meant as evidence of performance in deployment. First, a fixed query budget creates a tradeoff between querying many agents and learning each of them accurately (Theorem 4.2). Second, the parameter p trades off the average violation against the largest violations (Theorem 3.6). Our code is available at `https://github.com/nicteh/Learning-Acceptable-Lotteries-Minimizing-Violation-Distances`.

For the first effect, we run Algorithm 1 with $p = 2$ on two populations, each with $n = 1000$ agents and $m = 10$ alternatives. In the *shared-block* population, agents are noisy variations of a common preference pattern. In the *conflicting-blocks* population, five clusters of agents favor disjoint blocks of alternatives, and the smallest cluster has a stricter threshold. Both populations are ε-quantized with $\varepsilon = 1/32$, and every agent strictly accepts some pure alternative. The algorithm learns each sampled agent with the procedure of Lemma 4.1, which runs a binary search on each of $m-1$ edges of the simplex. With 109 queries, these searches recover the boundary exactly; with fewer queries, we stop each search early and fit the boundary through the midpoints of the final search intervals. Since this learner then has no accuracy guarantee, the experiments illustrate the tradeoff described by Theorem 4.2 rather than test it. Under a total budget of $Q = 3200$ queries, each of the k sampled agents receives $\lfloor Q/k \rfloor$ queries. Appendix D gives the details.

To separate the two sources of error, let $\mathbf{x}_{\text{sample}}$ be a lottery that minimizes the objective of the sampled agents with their true boundaries. We split the total excess $G_{\text{tot}} = V_2(\hat{\mathbf{x}}) - \min_{\mathbf{x}} V_2(\mathbf{x})$ into the *coverage gap* $G_{\text{cov}} = V_2(\mathbf{x}_{\text{sample}}) - \min_{\mathbf{x}} V_2(\mathbf{x})$ and the *boundary-learning gap* $G_{\text{bdry}} = V_2(\hat{\mathbf{x}}) - V_2(\mathbf{x}_{\text{sample}})$, which play the roles of the sampling and boundary-learning terms in Theorem 4.2. Figure 2 shows that the budget must be split carefully. For $k \leq 20$, every sampled agent receives at least 109 queries and is learned exactly, so the total excess consists only of the coverage gap; for the conflicting-blocks population, it is 3.80×10^{-2} at $k = 5$. As k grows, the coverage gap shrinks, but each sampled agent is learned less accurately. At $k = 320$, each sampled agent receives only 10 queries, just enough to query the pure lotteries, and the boundary-learning gap is 3.57×10^{-2},

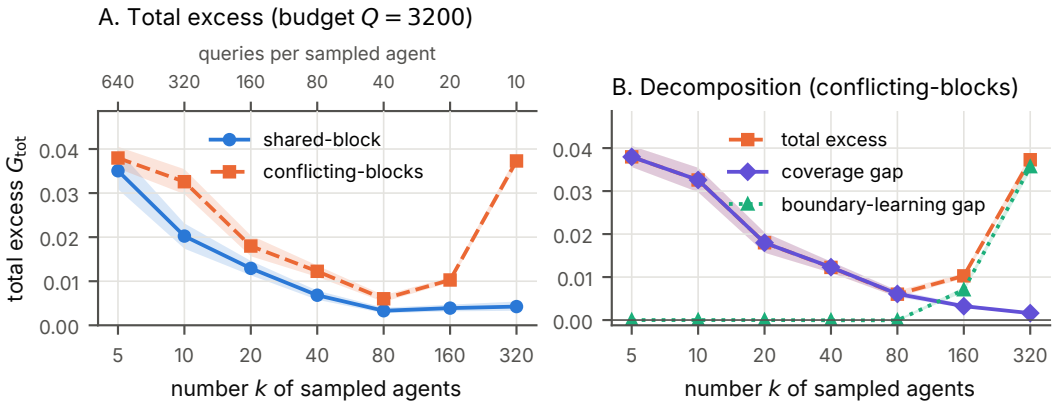


Figure 2: **A fixed query budget creates a tradeoff between coverage and accuracy.** Panel A shows the total excess G_{tot} as the number k of sampled agents varies under a fixed budget of $Q = 3200$ queries; the top axis shows the number $\lfloor Q/k \rfloor$ of queries per sampled agent, of which at most 109 are used. Panel B splits the total excess for the conflicting-blocks population into the coverage gap and the boundary-learning gap. Lines show means over 100 populations, and shaded bands are 95% bootstrap confidence intervals for the means.

about 96% of the total excess. Among the tested values of k, the smallest mean total excess for the conflicting-blocks population is 6.02×10^{-3}, attained at $k = 80$ with 40 queries per sampled agent and significantly smaller than for every other k. For the shared-block population, the smallest mean total excess, 3.30×10^{-3}, is also attained at $k = 80$, but the differences from $k = 160$ and $k = 320$ are within the noise. The best split depends on the population, however: if the thresholds in the conflicting-blocks population lie closer to the middle of each agent's utility range, the coarse boundaries learned from 10 queries hurt much less, and $k = 320$ does as well as $k = 80$ (Appendix D).

Appendix D also varies p on a structured instance with three clusters. As predicted by Theorem 3.6, larger values of p favor agents with larger violations: moving from $p = 1$ to $p = \infty$ reduces the largest violation from 0.391 to 0.225, but increases the average violation from 0.047 to 0.066.

6 Conclusion

In this paper, we have studied how to choose a lottery that minimizes aggregate violation when agents only reveal whether a proposed lottery is acceptable and no lottery need be acceptable to all of them. We showed that accept/reject answers determine the normalized violation of every agent except the distance-underdetermined ones, determined the unavoidable loss caused by these agents up to a constant factor, and analyzed a sampling algorithm whose number of queries does not depend on the number of agents. Overall, our results indicate that binary feedback suffices to optimize the severity of violations, and not only their number (studied by Teh and Goldberg [2026]), as long as no agent is distance-underdetermined. However, minimizing aggregate violation is no substitute for safety requirements that must hold without exception; such requirements should be imposed as constraints on the candidate lotteries rather than traded off against other requirements.

A technical question left by our work is to close the constant-factor gap between Theorems 3.3 and 3.4, and to prove a lower bound for randomized algorithms that query few agents, complementing Theorem 4.4. It would also be interesting to extend the limited-budget analysis of Section 4 to distance-underdetermined agents, and to determine whether exact optimization requires as many queries as exact recovery of the boundaries (Theorem 3.1). When additional information is available, such as a prior over the boundaries consistent with the answers, one could replace the worst-case objective $\bar{F}$ by a less pessimistic one, in the spirit of learning-augmented algorithms [Mitzenmacher and Vassilvitskii, 2021] (see also Appendix A.2). Another direction is to replace the simplex by a general polytope of feasible outcomes; Appendix A.4 explains which parts of our analysis extend and why our query algorithm does not. Finally, a study of elicitation from real stakeholders, whose answers may be uncertain or strategic, would test how far the theory of learning from acceptability feedback carries over to practice.

Acknowledgments and Disclosure of Funding

This work was funded by the Advanced Research + Invention Agency (ARIA) under the ASPAI project. PWG is also supported by EPSRC Grants EP/X040461/1 and EP/X038548/1.

References

Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 60–69, 2018.

Sushant Agarwal and Amit Deshpande. On the power of randomization in fair classification and representation. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency (FAccT)*, pages 1542–1551, 2022.

Priyank Agrawal, Eric Balkanski, Vasilis Gkatzelis, Tingting Ou, and Xizhi Tan. Learning-augmented mechanism design: Leveraging predictions for facility location. *Mathematics of Operations Research*, 49(4):2626–2651, 2024.

Stéphane Airiau, Haris Aziz, Ioannis Caragiannis, Justin Kruger, Jérôme Lang, and Dominik Peters. Portioning using ordinal preferences: Fairness and efficiency. *Artificial Intelligence*, 314:103809, 2023.

Ibrahim M. Alabdulmohsin, Xin Gao, and Xiangliang Zhang. Efficient active learning of halfspaces via query synthesis. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI)*, pages 2483–2489, 2015.

Shiri Alouf-Heffetz, Laurent Bulteau, Edith Elkind, Nimrod Talmon, and Nicholas Teh. Better collective decisions via uncertainty reduction. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 24–30, 2022.

Dana Angluin. Queries and concept learning. *Machine Learning*, 2(4):319–342, 1988.

Anthony B. Atkinson. On the measurement of inequality. *Journal of Economic Theory*, 2(3):244–263, 1970.

Haris Aziz, Anna Bogomolnaia, and Hervé Moulin. Fair mixing: the case of dichotomous preferences. *ACM Transactions on Economics and Computation*, 8(4):18:1–18:27, 2020.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022.

Eric Balkanski, Vasilis Gkatzelis, and Xizhi Tan. Strategyproof scheduling with predictions. In *Proceedings of the 14th Innovations in Theoretical Computer Science Conference (ITCS)*, pages 11:1–11:22, 2023.

Siddhartha Banerjee, Vasilis Gkatzelis, Artur Gorokh, and Billy Jin. Online Nash social welfare maximization with predictions. In *Proceedings of the 33rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1–19, 2022.

Siddhartha Banerjee, Vasilis Gkatzelis, Safwan Hossain, Billy Jin, Evi Micha, and Nisarg Shah. Proportionally fair online allocation of public goods with predictions. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 20–28, 2023.

Ben Berger, Michal Feldman, Vasilis Gkatzelis, and Xizhi Tan. Learning-augmented metric distortion via (p, q)-veto core. In *Proceedings of the 25th ACM Conference on Economics and Computation (EC)*, page 984, 2024.

Avrim Blum, Jeffrey C. Jackson, Tuomas Sandholm, and Martin Zinkevich. Preference elicitation and query learning. *Journal of Machine Learning Research*, 5:649–667, 2004.

Anna Bogomolnaia, Hervé Moulin, and Richard Stong. Collective choice under dichotomous preferences. *Journal of Economic Theory*, 122(2):165–184, 2005.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

Felix Brandt. Rolling the dice: Recent results in probabilistic social choice. In Ulle Endriss, editor, *Trends in Computational Social Choice*, chapter 1, pages 3–26. AI Access, 2017.

Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia, editors. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

Darius Braziunas and Craig Boutilier. Elicitation of factored utilities. *AI Magazine*, 29(4):79–92, 2008.

Violet Xinying Chen and J. N. Hooker. A guide to formulating fairness in an optimization model. *Annals of Operations Research*, 326:581–619, 2023.

Davin Choo, Winston Fu, Tzeh Yuan Neoh, Tze-Yang Poon, and Nicholas Teh. Approximate proportionality in online fair division. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, pages 20417–20450, 2026a.

Davin Choo, Paul W. Goldberg, and Nicholas Teh. Learning unanimously acceptable lotteries via queries. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, pages 20451–20493, 2026b. Extended version available as arXiv:2604.17505.

Saar Cohen, Nicholas Teh, Paul W. Goldberg, and Michael J. Wooldridge. Online fair division with budget constraints. arXiv:2607.23310, 2026a.

Saar Cohen, Nicholas Teh, and Michael Wooldridge. Online fair division under an oblivious adversary. arXiv:2609.28333, 2026b.

Vincent Conitzer, Rachel Freedman, Jobst Heitzig, Wesley H. Holliday, Bob M. Jacobs, Nathan Lambert, Milan Mosse, Eric Pacuit, Stuart Russell, Hailey Schoelkopf, Emanuel Tewolde, and William S. Zwicker. Position: Social choice should guide AI alignment in dealing with diverse human feedback. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pages 9346–9360, 2024.

Andrew Cotter, Maya Gupta, Heinrich Jiang, Nathan Srebro, Karthik Sridharan, Serena Wang, Blake Woodworth, and Seungil You. Training well-generalizing classifiers for fairness metrics and other data-dependent constraints. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pages 1397–1405, 2019.

Yu-Hong Dai and Liwei Zhang. Optimization with least constraint violation. *CSIAM Transactions on Applied Mathematics*, 2(3):551–584, 2021.

Ilias Diakonikolas, Daniel M. Kane, and Mingchen Ma. Active learning of general halfspaces: Label queries vs membership queries. In *Proceedings of the 38th International Conference on Neural Information Processing Systems (NeurIPS)*, pages 49180–49213, 2024.

Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Edith Elkind, Matthias Greger, Patrick Lederer, Warut Suksompong, and Nicholas Teh. Settling the score: Portioning with cardinal preferences. *Artificial Intelligence*, 352:104487, 2026.

European Union. Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official Journal of the European Union, L series, 12 July 2024, 2024.

Peter C. Fishburn. *The Foundations of Expected Utility*. Springer, 1982.

Rupert Freeman and Ulrike Schmidt-Kraepelin. Project-fair and truthful mechanisms for budget aggregation. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 9704–9712, 2024.

Rupert Freeman, David Pennock, Dominik Peters, and Jennifer Wortman Vaughan. Truthful aggregation of budget proposals. *Journal of Economic Theory*, 193:105234, 2021.

Allan Gibbard. Manipulation of schemes that mix voting with chance. *Econometrica*, 45(3):665–681, 1977.

Paul J. Goulart and Yuwen Chen. Clarabel: An interior-point solver for conic programs with quadratic objectives. *arXiv preprint arXiv:2405.12762*, 2024.

Yichen Huang and Nicholas Teh. Online fair division of chores with upgrade recourse. arXiv:2611.00432, 2026a.

Yichen Huang and Nicholas Teh. Online fair division with upgrade recourse. arXiv:2610.16943, 2026b.

Michael J. Kearns and Umesh V. Vazirani. *An Introduction to Computational Learning Theory*. MIT Press, 1994.

Ori Kelner. Learning halfspaces with membership queries. *arXiv preprint arXiv:2012.10985*, 2020.

Ron Kohavi, Diane Tang, and Ya Xu. *Trustworthy Online Controlled Experiments: A Practical Guide to A/B Testing*. Cambridge University Press, 2020.

Thodoris Lykouris and Sergei Vassilvitskii. Competitive caching with machine learned advice. *Journal of the ACM*, 68(4):24:1–24:25, 2021.

Michael Mitzenmacher and Sergei Vassilvitskii. Algorithms with predictions. In Tim Roughgarden, editor, *Beyond the Worst-Case Analysis of Algorithms*, pages 646–662. Cambridge University Press, 2021.

Luisa Montanari, Ulrike Schmidt-Kraepelin, Warut Suksompong, and Nicholas Teh. Weighted envy-freeness for submodular valuations. *Social Choice and Welfare*, 65:983–1016, 2025.

Tzeh Yuan Neoh and Nicholas Teh. Closing gaps in online fair division. arXiv:2609.05310, 2026a.

Tzeh Yuan Neoh and Nicholas Teh. Online fair division with frequency predictions. arXiv:2610.34625, 2026b.

Tzeh Yuan Neoh, Jannik Peters, and Nicholas Teh. Online fair division with additional information. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, pages 92207–92241, 2026a.

Tzeh Yuan Neoh, Nicholas Teh, Je Qin Chooi, Paul W. Goldberg, and Milind Tambe. Efficient ensemble selection from binary and pairwise feedback. arXiv:2605.09588, 2026b.

Kanad Shrikar Pardeshi, Itai Shapira, Ariel D. Procaccia, and Aarti Singh. Learning social welfare functions. In *Proceedings of the 38th Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 41733–41766, 2024.

Manish Purohit, Zoya Svitkina, and Ravi Kumar. Improving online algorithms via ML predictions. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, pages 9661–9670, 2018.

Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. XSTest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 5377–5400, 2024.

Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

Toby Shevlane, Sebastian Farquhar, Ben Garfinkel, Mary Phuong, Jess Whittlestone, Jade Leung, Daniel Kokotajlo, Nahema Marchal, Markus Anderljung, Noam Kolt, Lewis Ho, Divya Siddarth, Shahar Avin, Will Hawkins, Been Kim, Iason Gabriel, Vijay Bolina, Jack Clark, Yoshua Bengio, Paul Christiano, and Allan Dafoe. Model evaluation for extreme risks. *arXiv preprint arXiv:2305.15324*, 2023.

Herbert A. Simon. A behavioral model of rational choice. *The Quarterly Journal of Economics*, 69 (1):99–118, 1955.

Taylor Sorensen, Jared Moore, Jillian Fisher, Mitchell L. Gordon, Niloofar Mireshghallah, Christopher Michael Rytting, Andre Ye, Liwei Jiang, Ximing Lu, Nouha Dziri, Tim Althoff, and Yejin Choi. Position: A roadmap to pluralistic alignment. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pages 46280–46302, 2024.

Warut Suksompong. Weighted fair division of indivisible items: A review. *Information Processing Letters*, 187:106519, 2025.

Warut Suksompong and Nicholas Teh. On maximum weighted Nash welfare for binary valuations. *Mathematical Social Sciences*, 117:101–108, 2022.

Warut Suksompong and Nicholas Teh. Weighted fair division with matroid-rank valuations: Monotonicity and strategyproofness. *Mathematical Social Sciences*, 126:48–59, 2023.

Warut Suksompong and Nicholas Teh. Voting in divisible settings: A survey. In *Proceedings of the 40th AAAI Conference on Artificial Intelligence (AAAI)*, pages 39789–39796, 2026.

Nicholas Teh. Weighted fair division of indivisible mixed manna. arXiv:2609.01580, 2026.

Nicholas Teh and Paul W. Goldberg. Learning lotteries with minimal stakeholder violations from membership queries. In *Proceedings of the 40th International Conference on Neural Information Processing Systems (NeurIPS)*, 2026. To appear. Full version available as arXiv:2610.37828.

Nicholas Teh, Saar Cohen, Karen Frilya Celine, Benjamin Yu, and Michael J. Wooldridge. Online house allocation with subsidy. arXiv:2609.29691, 2026.

John von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Princeton University Press, 1944.

Laura Weidinger, Maribeth Rauh, Nahema Marchal, Arianna Manzini, Lisa Anne Hendricks, Juan Mateos-Garcia, Stevie Bergman, Jackie Kay, Conor Griffin, Ben Bariach, Iason Gabriel, Verena Rieser, and William Isaac. Sociotechnical safety evaluation of generative AI systems. *arXiv preprint arXiv:2310.11986*, 2023.

Chenyang Xu and Pinyan Lu. Mechanism design with predictions. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 571–577, 2022.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gómez-Rodríguez, and Krishna P. Gummadi. Fairness constraints: A flexible approach for fair classification. *Journal of Machine Learning Research*, 20 (75):1–42, 2019.

A Modeling Details and Omitted Discussion

A.1 Halfspace Acceptability and Expected Utility

In this appendix, we clarify the role of the expected utility assumption. Our results use expected utility only through a geometric consequence: the acceptable set of each agent is the intersection of the simplex with a halfspace. Our model should therefore be understood primarily as the assumption that acceptable sets are halfspaces, with expected utility as the standard interpretation for lotteries with known probabilities.

Linear threshold rules. Consider any acceptability rule of the form $A_i = \{\mathbf{x} \in \Delta(S) : \langle \mathbf{c}, \mathbf{x} \rangle \geq d\}$ with $\mathbf{c} \in \mathbb{R}^m$ and $d \in \mathbb{R}$, and assume that $A_i \neq \varnothing$ and $A_i \neq \Delta(S)$. Since a linear function on the simplex attains its minimum and maximum at pure lotteries, we have $c_{\min} < d \leq c_{\max}$, where $c_{\min}$ and $c_{\max}$ are the smallest and largest coordinates of $\mathbf{c}$. Let

$$u_i := \frac{\mathbf{c} - c_{\min}\mathbf{1}}{c_{\max} - c_{\min}}, \quad \tau_i := \frac{d - c_{\min}}{c_{\max} - c_{\min}}.$$

Then $u_i \in [0,1]^m$ and $\tau_i \in (0,1]$, and since $\sum_{j=1}^m x_j = 1$ for every $\mathbf{x} \in \Delta(S)$, we have $\langle \mathbf{c}, \mathbf{x} \rangle \geq d \iff \langle u_i, \mathbf{x} \rangle \geq \tau_i$. Hence the notation of Section 2 is without loss of generality for any nontrivial linear threshold rule; conversely, every pair (u_i, τ_i) in Section 2 induces such a rule. In particular, our results rely neither on interpersonal comparisons of utilities nor on the learner observing utilities; they only rely on the assumption that acceptable sets are halfspaces.

Interpretation for randomized deployments. The alternatives $s_1, \ldots, s_m$ can be candidate classifiers, versions of a foundation model, ranking rules, or allocation policies, and a lottery $\mathbf{x} \in \Delta(S)$ can represent a randomized deployment, a randomized classifier, or a repeated deployment in which alternative s_j is used with frequency x_j. If stakeholder i assigns a score $u_i(s_j)$ to each alternative and accepts the deployments whose expected score is at least τ_i, then acceptability is exactly $\sum_{j=1}^m x_j u_i(s_j) \geq \tau_i$. The score need not be a utility in the psychological sense: it can be, for example, the negative of a rate of harmful replies, the accuracy on a subgroup of users, or any other statistic that is averaged over requests. This interpretation is particularly natural in constrained and fair machine learning, where randomized classifiers and constraints in expectation are standard tools [Agarwal et al., 2018, Cotter et al., 2019]. The difference in our setting is that the constraints are not known to the learner and must be learned from queries.

Scope of the assumption. As is common in learning theory, we fix a hypothesis class (here, the halfspaces of the simplex) and ask what can be identified and optimized from limited labels [Shalev-Shwartz and Ben-David, 2014]. Our feedback model is weaker than one in which cardinal utilities, welfare values, or rankings are observed: the learner never sees u_i or τ_i, and the normalization in Section 2 removes offsets and scales before violations are compared. Some structural restriction is also necessary for query-efficient guarantees. Indeed, with an arbitrary membership oracle on the continuum of lotteries, finitely many queries leave the answers at all other lotteries unconstrained, so no nontrivial guarantee for exact optimization is possible. The halfspace assumption is therefore best read as a choice of model class, one that makes identification, normalization, convex optimization, and finite-query analysis possible.

Beyond expected utility. We do not claim that all human or institutional preferences over lotteries obey expected utility. If acceptability is genuinely nonlinear in the probabilities, exact recovery of halfspaces and the violation geometry of this paper need not apply, and our results should then be viewed as a baseline, analogous to linear models in supervised learning. Two extensions are natural. First, if answers are flipped independently at random, Proposition 4.5 reduces the problem to the case of exact answers. If answers are noisy in other ways, or only approximately described by halfspaces, Theorem 4.2 still shows where the additional error enters: any routine that learns each boundary up to accuracy η contributes the corresponding boundary-learning term to the final loss, so an active learning method for noisy labels could replace exact recovery. Second, richer acceptability models, such as intersections of halfspaces, piecewise-linear acceptable regions, or more general convex sets, can be handled by enlarging the model class. This would change the analysis of learnability and query complexity, and the optimality conditions and the bound on the number of alternatives in Section 3.3 need not hold unchanged, but the same template remains natural: sample agents, learn their acceptability rules, and optimize the resulting objective.

A.2 Additional Related Work

The main text discusses the work most closely related to ours. Here, we elaborate on our companion paper and discuss several further connections, including the use of predictions to circumvent worst-case lower bounds.

Minimizing the number of rejecting agents. Teh and Goldberg [2026] considered the same query model but a different objective: instead of measuring how severely each agent is violated, they counted the agents who reject the chosen lottery. They showed that the problem is APX-hard already when all boundaries are known and all utilities are 0 or 1, gave exact algorithms under additional structure, and proved guarantees, with a sample size independent of n, for optimizing over a uniform sample of agents. They also showed that, for $m \geq 3$, a deterministic algorithm querying at most K distinct agents can be forced to incur $n - K$ more rejections than necessary, which parallels Theorem 4.4. The two objectives differ in two respects. First, the number of rejections depends only on the acceptable sets, which membership queries always determine, so the ambiguity of Section 3.2 does not arise for that objective. Second, our objective is convex once the boundaries are known, so the difficulty in our setting is informational rather than computational. Conceptually, a distance-based objective distinguishes slight violations from severe ones, whereas counting rejections is appropriate for requirements that are simply met or not; indeed, a small aggregate distance can hide many agents who each fall slightly short of their thresholds, which is one reason why requirements that must hold without exception should be imposed as constraints (Section 6).

Identifiability. As in much of learning theory, a structural assumption on the hypothesis class makes finite-query guarantees possible [Kearns and Vazirani, 1994, Shalev-Shwartz and Ben-David, 2014]. In our setting, the hypothesis attached to each agent is a halfspace over the continuum of lotteries, and this assumption leaves an identifiability issue that is specific to acceptability queries: several pairs of utilities and thresholds induce exactly the same membership oracle. The normalized acceptance boundary is therefore the object that matters for the decision, and the results in Section 3 characterize when this object is identifiable, when only the acceptable set is identifiable, and how the remaining ambiguity affects the optimization problem.

Learning over a population of agents. Our algorithm for limited budgets resembles empirical risk minimization over a population of agents: each queried agent contributes a learned violation function, and the returned lottery minimizes the empirical aggregate. The difference from a standard supervised learning pipeline is that each sampled data point is itself an unknown constraint that must first be elicited through membership queries. This is why our analysis separates boundary-learning error, sampling error, and the effect of unqueried agents. The uniform convergence step in the proof of Theorem 4.2 uses standard symmetrization arguments from statistical learning theory [Kearns and Vazirani, 1994, Shalev-Shwartz and Ben-David, 2014], applied to a loss class induced by normalized violations of learned acceptance boundaries.

Randomizing over few alternatives. Theorem 3.7 gives a convex-analytic reason why an optimal lottery need not randomize over all alternatives: some optimal lottery uses at most $n + 1$ alternatives, even though a lottery may use the entire menu. Unlike the axiomatic guarantees usually studied for voting or allocation rules, this property is a structural feature of the optimization problem that arises once the acceptance boundaries have been learned. Besides its usual roles as a fairness or strategic device, randomization here reaches compromises that no single alternative achieves.

Predictions in collective decision-making. Two of our lower bounds are worst-case statements that additional information could circumvent: Theorem 3.4 holds because accept/reject answers cannot distinguish certain boundaries, and Theorem 4.4 holds because a deterministic algorithm learns nothing about the agents it never queries. Learning-augmented algorithms address limits of this kind by using predictions that may be erroneous, with guarantees that improve as the predictions become more accurate yet remain meaningful when they are poor [Lykouris and Vassilvitskii, 2021, Purohit et al., 2018, Mitzenmacher and Vassilvitskii, 2021]. This framework has been applied to mechanism design [Xu and Lu, 2022, Balkanski et al., 2023, Agrawal et al., 2024], voting [Berger et al., 2024], and online fair division, where predictions, together with other relaxations such as recourse and weaker adversaries, help circumvent strong impossibility results [Banerjee et al., 2022, 2023, Choo et al., 2026a, Neoh et al., 2026a, Neoh and Teh, 2026a,b, Cohen et al., 2026a,b, Huang and Teh, 2026b,a, Teh et al., 2026]. Closer to our setting, Choo et al. [2026b] showed that advice (for example, a candidate lottery that is likely to be acceptable, or a guess as to which requirements are tight) reduces the number of queries needed to reach unanimity. Algorithm 1 instead relies on

randomization: by choosing at random whom to query, it avoids the lower bound of Theorem 4.4, which applies only to deterministic algorithms. Combining random sampling with predictions about the agents' boundaries, for instance to replace the worst-case objective $\bar{F}$ of Section 3.2 by a less pessimistic one, is an interesting direction for future work.

Uncertainty in collective decisions. Our learner improves a collective decision by reducing its own uncertainty about the agents' requirements. In a different model, Alouf-Heffetz et al. [2022] studied how a benevolent external party can improve the outcome of issue-by-issue majority voting when some agents are confused about which option is better for them, by educating some of the agents, appointing a subset of competent agents to decide on behalf of the group, or guiding the delegation of votes. There, the uncertainty lies with the agents rather than with the decision maker, and the main questions concern the computational complexity of choosing such an intervention rather than the number of queries.

A.3 Normalization Details and an Example

This subsection gives the details behind the normalized violation in Section 2. Recall that $\bar{u}_i = \frac{1}{m}\sum_{j=1}^{m} u_i(s_j)$ and $w_i = u_i - \bar{u}_i\mathbf{1}$. Then $\sum_{j=1}^{m} w_i(s_j) = 0$, and $\langle w_i, \mathbf{x}\rangle = \langle u_i, \mathbf{x}\rangle - \bar{u}_i$ for every $\mathbf{x} \in \Delta(S)$. Hence

$$\langle u_i, \mathbf{x}\rangle \geq \tau_i \iff \langle w_i, \mathbf{x}\rangle \geq \tau_i - \bar{u}_i.$$

Since agent i accepts some pure alternative and rejects another, u_i is not constant, so $w_i \neq 0$. Dividing by $\|w_i\|$ gives $(\hat{u}_i, \hat{\tau}_i)$, which satisfies

$$\|\hat{u}_i\| = 1, \quad \sum_{j=1}^{m} \hat{u}_i(s_j) = 0, \quad \langle u_i, \mathbf{x}\rangle \geq \tau_i \iff \langle \hat{u}_i, \mathbf{x}\rangle \geq \hat{\tau}_i.$$

For a rejected lottery $\mathbf{x}$, the violation $v_i(\mathbf{x}) = \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}\rangle$ is exactly the Euclidean distance from $\mathbf{x}$ to H_i. Indeed, since the coordinates of $\hat{u}_i$ sum to zero, $\hat{u}_i$ is parallel to the affine hull of $\Delta(S)$, and it is orthogonal to every direction within H_i. Hence the closest point of H_i to $\mathbf{x}$ is $\mathbf{x} + (\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}\rangle)\hat{u}_i$, and since $\|\hat{u}_i\| = 1$, its distance from $\mathbf{x}$ is $\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}\rangle$.

A simple example shows why this normalization matters. Let $m = 2$ and write $\mathbf{x} = (x_1, 1 - x_1)$. Consider two agents $(u_a, \tau_a) = ((1, 0), 1/2)$ and $(u_b, \tau_b) = ((1, 1/2), 3/4)$. Both accept exactly the same lotteries, namely those with $x_1 \geq 1/2$. However, their shortfalls are on different scales: $\max\{0, \tau_a - \langle u_a, \mathbf{x}\rangle\} = \max\{0, \frac{1}{2} - x_1\}$, whereas

$$\max\{0, \tau_b - \langle u_b, \mathbf{x}\rangle\} = \max\left\{0, \frac{3}{4} - x_1 - \frac{1}{2}(1 - x_1)\right\} = \frac{1}{2}\max\left\{0, \frac{1}{2} - x_1\right\}.$$

Thus shortfalls would treat two identical acceptability rules differently. After centering and normalizing, both agents have $\hat{u} = (1/\sqrt{2}, -1/\sqrt{2})$ and $\hat{\tau} = 0$, and hence the same violation $\sqrt{2}\max\{0, \frac{1}{2} - x_1\}$. For rejected $\mathbf{x}$, this is precisely the Euclidean distance within the line $x_1 + x_2 = 1$ from $\mathbf{x}$ to the boundary point $(1/2, 1/2)$.

A.4 Beyond the Simplex

In some applications, the feasible outcomes form a known bounded polytope $P \subseteq \mathbb{R}^m$ other than the simplex (for example, budget allocations with additional upper bounds on individual projects). If each agent accepts the points of P on one side of a hyperplane, violations can be defined as in Section 2, by normalizing the hyperplane within the affine hull of P and measuring Euclidean distances within this affine hull. Several parts of our analysis only use convexity and the diameter of the feasible set, and extend with minor changes. The objective remains convex. The optimality condition of Theorem 3.6 becomes the condition that the optimal point maximizes the weighted score $\sum_{i \in N} v_i(\mathbf{x}^*)^{p-1}\langle \hat{u}_i, \mathbf{x}\rangle$ over $\mathbf{x} \in P$. The analysis of Algorithm 1 goes through with $\sqrt{2}$ replaced by the diameter of P and with the bound $\|\mathbf{x}\| \leq 1$ replaced by a bound on the norms of points in P. Moreover, if the boundary of some agent is not determined by her answers, the distance from a rejected point $\mathbf{x}$ to any boundary consistent with the answers is still at most $\mathrm{dist}(\mathbf{x}, A_i)$, because such a boundary separates $\mathbf{x}$ from A_i; hence the argument behind Theorem 3.3 gives the same bound with $\sqrt{2}$ replaced by the diameter of P. By contrast, our query algorithm in Theorem 3.1 relies on the vertices and edges of the simplex, and characterizing what membership queries reveal for a general polytope requires further work: the

vertices of P need not be affinely independent, so the three cases of Section 3.1 no longer cover all possibilities. It would also be interesting to understand how the combinatorial structure of P, for instance, when P is a polymatroid, affects these questions.

B Omitted Proofs from Section 3

B.1 Proof of Theorem 3.1

We first describe a procedure for a single agent i that makes $\mathcal{O}(m \log(1/\varepsilon))$ queries; running it for every agent gives the claimed algorithm. The high-level idea is to find enough points on the boundary of the agent by searching along edges of the simplex, as illustrated in Figure 1.

Finding the boundary point on an edge. Let $\mathbf{y}$ and $\mathbf{z}$ be pure lotteries such that agent i rejects $\mathbf{y}$ and accepts $\mathbf{z}$. Since U_i is linear, agent i accepts $(1-\lambda)\mathbf{y} + \lambda\mathbf{z}$ if and only if $\lambda \geq \lambda^*$, where

$$\lambda^* := \frac{\tau_i - U_i(\mathbf{y})}{U_i(\mathbf{z}) - U_i(\mathbf{y})} \in (0, 1].$$

Since the instance is ε-quantized, λ^* is a fraction whose denominator is at most $1/\varepsilon$, and any two distinct such fractions differ by at least ε^2. Hence binary search on λ finds λ^*, and thus the boundary point $(1-\lambda^*)\mathbf{y} + \lambda^*\mathbf{z}$, with $\mathcal{O}(\log(1/\varepsilon))$ queries.

The procedure. First, query all m pure lotteries; this reveals M_i, which is nonempty and different from $[m]$. Fix any $b \notin M_i$. For each $j \in M_i$, find the value λ_j^* for the edge from $\mathbf{e}_b$ to $\mathbf{e}_j$. Note that $\lambda_j^* < 1$ if $U_i(\mathbf{e}_j) > \tau_i$, and $\lambda_j^* = 1$ if $U_i(\mathbf{e}_j) = \tau_i$. Hence, with $\mathcal{O}(m \log(1/\varepsilon))$ queries, we can tell which of the following three cases applies.

Case 1: $\lambda_a^* < 1$ for some $a \in M_i$. Then $U_i(\mathbf{e}_a) > \tau_i$. For each $j \in [m] \setminus \{a\}$, let $\mathbf{y}_j$ be the boundary point on the edge from $\mathbf{e}_b$ to $\mathbf{e}_j$ (found above) if $j \in M_i$, and on the edge from $\mathbf{e}_j$ to $\mathbf{e}_a$ if $j \notin M_i$. Each $\mathbf{y}_j$ lies on H_i.

We claim that the $m-1$ points $\mathbf{y}_j$ are affinely independent. For each $j \in [m] \setminus \{a, b\}$, the point $\mathbf{y}_j$ is a convex combination of $\mathbf{e}_j$ and one of $\mathbf{e}_a, \mathbf{e}_b$ with a positive coefficient on $\mathbf{e}_j$: if $j \in M_i$, this coefficient is positive because $\mathbf{e}_b$ is rejected and therefore $\mathbf{y}_j \neq \mathbf{e}_b$, and if $j \notin M_i$, it is positive because $\mathbf{e}_a$ is strictly accepted and therefore $\mathbf{y}_j \neq \mathbf{e}_a$. Moreover, $\mathbf{y}_b$ is a combination of $\mathbf{e}_b$ and $\mathbf{e}_a$. Hence, for $j \in [m] \setminus \{a, b\}$, the j-th coordinate of $\mathbf{y}_j - \mathbf{y}_b$ is positive, while for every $k \in [m] \setminus \{a, b, j\}$, the j-th coordinate of $\mathbf{y}_k - \mathbf{y}_b$ is zero. Thus the $m-2$ vectors $\mathbf{y}_j - \mathbf{y}_b$ with $j \in [m] \setminus \{a, b\}$ are linearly independent, which proves the claim.

Since H_i has dimension $m-2$, it is the affine hull of these points. Hence $\hat{u}_i$ is the unique vector satisfying

$$\|\hat{u}_i\| = 1, \quad \sum_{\ell=1}^{m} \hat{u}_i(s_\ell) = 0, \quad \langle \hat{u}_i, \mathbf{y}_j - \mathbf{y}_b \rangle = 0 \text{ for all } j \in [m] \setminus \{a, b\}, \quad \langle \hat{u}_i, \mathbf{e}_a - \mathbf{y}_b \rangle > 0,$$

where the equalities determine $\hat{u}_i$ up to sign, and the last inequality fixes the sign because $\mathbf{e}_a$ is strictly accepted while $\mathbf{y}_b$ lies on the boundary. Finally, $\hat{\tau}_i = \langle \hat{u}_i, \mathbf{y}_b \rangle$.

Case 2: $\lambda_j^* = 1$ for every $j \in M_i$, and $|M_i| = m-1$. Then $U_i(\mathbf{e}_j) = \tau_i$ for every $j \in M_i$, and $U_i(\mathbf{e}_b) < \tau_i$ for the unique $b \notin M_i$. For any $\mathbf{x} \in \Delta(S)$,

$$U_i(\mathbf{x}) = \sum_{j \in M_i} x_j U_i(\mathbf{e}_j) + x_b U_i(\mathbf{e}_b) = \tau_i(1 - x_b) + x_b U_i(\mathbf{e}_b) \leq \tau_i,$$

with equality if and only if $x_b = 0$. Therefore, $A_i = \{\mathbf{x} \in \Delta(S) : x_b = 0\} = \operatorname{conv}\{\mathbf{e}_j : j \in M_i\}$. The $m-1$ affinely independent points $\mathbf{e}_j$ with $j \in M_i$ lie on H_i, so H_i is their affine hull, and $\hat{u}_i$ is determined up to sign by $\|\hat{u}_i\| = 1$, $\sum_{\ell=1}^{m} \hat{u}_i(s_\ell) = 0$, and $\langle \hat{u}_i, \mathbf{e}_j - \mathbf{e}_{j'} \rangle = 0$ for all $j, j' \in M_i$. We choose the sign so that $\hat{u}_i(s_b) < \hat{u}_i(s_j)$ for $j \in M_i$, and set $\hat{\tau}_i := \hat{u}_i(s_j)$ for any $j \in M_i$.

Case 3: $\lambda_j^* = 1$ for every $j \in M_i$, and $|M_i| \leq m-2$. Then $i \in B$. As in Case 2, $U_i(\mathbf{e}_j) = \tau_i$ for every $j \in M_i$ and $U_i(\mathbf{e}_k) < \tau_i$ for every $k \notin M_i$, so for any $\mathbf{x} \in \Delta(S)$,

$$U_i(\mathbf{x}) = \tau_i \sum_{j \in M_i} x_j + \sum_{k \notin M_i} x_k U_i(\mathbf{e}_k) \leq \tau_i,$$

with equality if and only if $x_k = 0$ for all $k \notin M_i$. Hence

$$A_i = \{\mathbf{x} \in \Delta(S) : x_k = 0 \text{ for all } k \notin M_i\} = \operatorname{conv}\{\mathbf{e}_j : j \in M_i\}, \tag{1}$$

which is known once M_i is known.

In each case, the procedure makes m queries to pure lotteries and $\mathcal{O}(m)$ binary searches with $\mathcal{O}(\log(1/\varepsilon))$ queries each, for a total of $\mathcal{O}(m\log(1/\varepsilon))$ queries. Running it for all agents gives $\mathcal{O}(nm\log(1/\varepsilon))$ queries.

Agents in B. Let $i \in B$. By Lemma 3.2, the normalized boundaries that give the same answers as agent i correspond to the positive vectors $(\delta_j)_{j\notin M_i}$ with $\sum_{j\notin M_i}\delta_j^2 - \frac{1}{m}(\sum_{j\notin M_i}\delta_j)^2 = 1$, and different vectors give different violations. By (3) below, the left-hand side of this equation is positive for every nonzero vector, so every positive vector becomes such a vector after scaling. Since agent i rejects at least two pure alternatives, there are infinitely many positive vectors, no two of which are proportional, and they give infinitely many different normalized boundaries.

Lower bound. Let $m \geq 3$, and consider an agent with $u(s_1) = 0$, $u(s_2) = 1$, $u(s_j) \in \{0, \varepsilon, \dots, 1\}$ for $j \in \{3, \dots, m\}$, and $\tau \in \{\varepsilon, 2\varepsilon, \dots, 1-\varepsilon\}$. Such an agent rejects $\mathbf{e}_1$ and strictly accepts $\mathbf{e}_2$, so she is not in B. Different choices of (u, τ) give different normalized boundaries. Indeed, if (u, τ) and (u', τ') have the same normalized boundary, then their centered vectors are positive multiples of each other, which means that $u' = cu + d\mathbf{1}$ and $\tau' = c\tau + d$ for some $c > 0$ and $d \in \mathbb{R}$; the conditions $u(s_1) = u'(s_1) = 0$ and $u(s_2) = u'(s_2) = 1$ then force $d = 0$ and $c = 1$. Hence there are $(1/\varepsilon - 1)(1/\varepsilon + 1)^{m-2} \geq (1/\varepsilon)^{m-2}$ such agents with pairwise different normalized boundaries, and at least $(1/\varepsilon)^{n(m-2)}$ profiles of n such agents, all ε-quantized and with $B = \varnothing$. Consider a deterministic algorithm that finds the normalized boundaries of all agents, and suppose that it makes at most L queries on each of these profiles. Since each query has two possible answers, the algorithm observes at most 2^L different sequences of answers, and its output is determined by this sequence. As it must give different outputs on different profiles, we have $2^L \geq (1/\varepsilon)^{n(m-2)}$, so $L \geq n(m-2)\log_2(1/\varepsilon) \geq \frac{nm}{3}\log_2(1/\varepsilon)$, where the last inequality uses $m \geq 3$.

B.2 Proof of Lemma 3.2

Let $i \in B$ and $C := [m] \setminus M_i$, so that $|C| \geq 2$. By (1), $A_i = \{\mathbf{x} \in \Delta(S) : x_j = 0 \text{ for all } j \in C\}$. A normalized boundary gives the same answer as agent i to every membership query if and only if its acceptable set within $\Delta(S)$ is A_i. We first describe all such boundaries, and then maximize the violation over them.

Consistent boundaries. Let $(\hat{u}, \hat{\tau})$ be a pair with $\|\hat{u}\| = 1$ and $\sum_{j=1}^m \hat{u}(s_j) = 0$ such that $\{\mathbf{y} \in \Delta(S) : \langle \hat{u}, \mathbf{y}\rangle \geq \hat{\tau}\} = A_i$. Since $\mathbf{e}_j \in A_i$ for $j \in M_i$, we have $\hat{u}(s_j) \geq \hat{\tau}$ for all $j \in M_i$. We claim that, in fact, $\hat{u}(s_j) = \hat{\tau}$ for all $j \in M_i$. Indeed, if $\hat{u}(s_j) > \hat{\tau}$ for some $j \in M_i$, then for any $k \in C$, we have $(1-\lambda)\hat{u}(s_j) + \lambda\hat{u}(s_k) > \hat{\tau}$ for all sufficiently small $\lambda > 0$, so the lottery $(1-\lambda)\mathbf{e}_j + \lambda\mathbf{e}_k \notin A_i$ would be accepted, a contradiction. Also, $\mathbf{e}_j \notin A_i$ for $j \in C$, so $\hat{u}(s_j) < \hat{\tau}$. Let $\delta_j := \hat{\tau} - \hat{u}(s_j) > 0$ for $j \in C$. Since $\sum_{j=1}^m \hat{u}(s_j) = 0$, we have $m\hat{\tau} = \sum_{j\in C}\delta_j$, and since $\|\hat{u}\| = 1$,

$$1 = \sum_{j=1}^m \hat{u}(s_j)^2 = \sum_{j=1}^m (\hat{\tau} - \hat{u}(s_j))^2 - m\hat{\tau}^2 = \sum_{j\in C}\delta_j^2 - \frac{1}{m}\Big(\sum_{j\in C}\delta_j\Big)^2. \tag{2}$$

Moreover, for every $\mathbf{x} \in \Delta(S)$, using $\sum_{j=1}^m x_j = 1$,

$$\hat{\tau} - \langle \hat{u}, \mathbf{x}\rangle = \sum_{j=1}^m x_j(\hat{\tau} - \hat{u}(s_j)) = \sum_{j\in C}\delta_j x_j \geq 0,$$

so the violation of $(\hat{u}, \hat{\tau})$ at $\mathbf{x}$ is $\sum_{j\in C}\delta_j x_j$.

Conversely, let $\delta \in \mathbb{R}^C$ be a positive vector satisfying (2), and set $\hat{\tau} := \frac{1}{m}\sum_{j\in C}\delta_j$, $\hat{u}(s_j) := \hat{\tau}$ for $j \in M_i$, and $\hat{u}(s_j) := \hat{\tau} - \delta_j$ for $j \in C$. Then $\sum_{j=1}^m \hat{u}(s_j) = m\hat{\tau} - \sum_{j\in C}\delta_j = 0$, and the calculation in (2) gives $\|\hat{u}\| = 1$. As above, the violation at $\mathbf{x} \in \Delta(S)$ is $\sum_{j\in C}\delta_j x_j$, which is zero if and only if $x_j = 0$ for all $j \in C$, that is, if and only if $\mathbf{x} \in A_i$. This proves the first part of the lemma. By Appendix A.1, each of these normalized boundaries is induced by some utility vector in $[0,1]^m$ and threshold in $(0,1]$.

The largest violation. Fix $\mathbf{x} \in \Delta(S)$ and let $\mathbf{z} := (x_j)_{j \in C} \in \mathbb{R}^C_{\geq 0}$. By the first part, we need to maximize $\langle \mathbf{z}, \delta \rangle$ over positive vectors $\delta \in \mathbb{R}^C$ with $\delta^\top G \delta = 1$, where $G := I - \frac{1}{m}\mathbf{1}\mathbf{1}^\top$ is a $|C| \times |C|$ matrix and $\mathbf{1}$ is the all-ones vector in $\mathbb{R}^C$. For every nonzero $\delta \in \mathbb{R}^C$, the Cauchy–Schwarz inequality gives $(\mathbf{1}^\top \delta)^2 \leq |C|\,\|\delta\|^2$, so

$$\delta^\top G \delta = \|\delta\|^2 - \frac{1}{m}(\mathbf{1}^\top \delta)^2 \geq \Big(1 - \frac{|C|}{m}\Big)\|\delta\|^2 = \frac{|M_i|}{m}\|\delta\|^2 > 0. \tag{3}$$

Thus G is positive definite, and it is easy to verify that $G^{-1} = I + \frac{1}{|M_i|}\mathbf{1}\mathbf{1}^\top$. By the Cauchy–Schwarz inequality for the inner product induced by G,

$$\langle \mathbf{z}, \delta \rangle = (G^{-1}\mathbf{z})^\top G \delta \leq \sqrt{\mathbf{z}^\top G^{-1}\mathbf{z}}\,\sqrt{\delta^\top G \delta} = \sqrt{\mathbf{z}^\top G^{-1}\mathbf{z}}.$$

If $\mathbf{z} = 0$, then every consistent boundary has violation 0 at $\mathbf{x}$. Otherwise, equality holds for $\delta = G^{-1}\mathbf{z}/\sqrt{\mathbf{z}^\top G^{-1}\mathbf{z}}$, which is positive because $G^{-1}\mathbf{z} = \mathbf{z} + \frac{1}{|M_i|}(\mathbf{1}^\top \mathbf{z})\mathbf{1}$ and $\mathbf{1}^\top \mathbf{z} > 0$. In both cases, the largest violation at $\mathbf{x}$ is

$$\sqrt{\mathbf{z}^\top G^{-1}\mathbf{z}} = \sqrt{\sum_{j \in C} x_j^2 + \frac{1}{|M_i|}\Big(\sum_{j \in C} x_j\Big)^2}. \tag{4}$$

On the other hand, define $\mathbf{y} \in \mathbb{R}^m$ by $y_j := 0$ for $j \in C$ and $y_j := x_j + \frac{1}{|M_i|}\sum_{\ell \in C} x_\ell$ for $j \in M_i$. Then $\mathbf{y} \geq 0$, $\sum_{j=1}^m y_j = 1$, and $y_j = 0$ for $j \in C$, so $\mathbf{y} \in A_i$. Moreover, $\mathbf{x} - \mathbf{y}$ is orthogonal to every vector $\mathbf{h} \in \mathbb{R}^m$ with $h_j = 0$ for $j \in C$ and $\sum_{j \in M_i} h_j = 0$, since $\mathbf{x} - \mathbf{y}$ is constant on the coordinates in M_i. These vectors $\mathbf{h}$ are exactly the directions of the affine hull of A_i, so $\mathbf{y}$ is the orthogonal projection of $\mathbf{x}$ onto this affine hull. Since $\mathbf{y} \in A_i$, it is also the closest point of A_i to $\mathbf{x}$, and therefore

$$\operatorname{dist}(\mathbf{x}, A_i)^2 = \|\mathbf{x} - \mathbf{y}\|^2 = \sum_{j \in C} x_j^2 + \frac{1}{|M_i|}\Big(\sum_{j \in C} x_j\Big)^2.$$

Comparing with (4) shows that the largest violation at $\mathbf{x}$ equals $\operatorname{dist}(\mathbf{x}, A_i)$.

Finally, $\mathbf{x} \mapsto \operatorname{dist}(\mathbf{x}, A_i)^2$ is convex, so its maximum over $\Delta(S)$ is attained at a pure lottery. The formula above gives the value $1 + 1/|M_i|$ at $\mathbf{e}_j$ for $j \in C$, and 0 at $\mathbf{e}_j$ for $j \in M_i$. Hence $\max_{\mathbf{x} \in \Delta(S)} \operatorname{dist}(\mathbf{x}, A_i) = \sqrt{1 + 1/|M_i|}$.

B.3 Proof of Theorem 3.3

We first justify the two claims about $\bar{F}$ made before the theorem. For each $i \notin B$, the function v_i is convex, and since $t \mapsto t^p$ is convex and nondecreasing on $\mathbb{R}_{\geq 0}$, so is v_i^p. For each $i \in B$, the function $\mathbf{x} \mapsto \operatorname{dist}(\mathbf{x}, A_i)$ is convex because A_i is a nonempty closed convex set, and so is its p-th power. Hence $\bar{F}$ is convex and continuous on the compact set $\Delta(S)$, so a minimizer $\bar{\mathbf{x}}$ exists. Moreover, the normalized boundaries of the agents not in B are determined by the answers (Theorem 3.1), the boundaries of different agents can be chosen independently of each other, and, for each $i \in B$, the largest violation at $\mathbf{x}$ consistent with the answers is $\operatorname{dist}(\mathbf{x}, A_i)$ (Lemma 3.2). Therefore, $\bar{F}(\mathbf{x})$ is the largest value of $V_p(\mathbf{x})^p$ over all profiles that give the same answers as the true profile.

In particular, $V_p(\mathbf{x})^p \leq \bar{F}(\mathbf{x})$ for every $\mathbf{x} \in \Delta(S)$, with $\bar{F}(\mathbf{x}) - V_p(\mathbf{x})^p = \frac{1}{n}\sum_{i \in B}(\operatorname{dist}(\mathbf{x}, A_i)^p - v_i(\mathbf{x})^p)$. A minimizer $\mathbf{x}^*$ of V_p exists since V_p is continuous on $\Delta(S)$. Hence

$$V_p(\bar{\mathbf{x}})^p \leq \bar{F}(\bar{\mathbf{x}}) \leq \bar{F}(\mathbf{x}^*) = V_p(\mathbf{x}^*)^p + \frac{1}{n}\sum_{i \in B}\left(\operatorname{dist}(\mathbf{x}^*, A_i)^p - v_i(\mathbf{x}^*)^p\right),$$

which is the first bound. By Lemma 3.2, $\operatorname{dist}(\mathbf{x}^*, A_i) \leq \sqrt{1 + 1/|M_i|} \leq \sqrt{2}$ for every $i \in B$, so

$$V_p(\bar{\mathbf{x}})^p \leq V_p(\mathbf{x}^*)^p + \frac{|B|}{n}(\sqrt{2})^p.$$

Taking p-th roots and using $(a+b)^{1/p} \leq a^{1/p} + b^{1/p}$ for $a, b \geq 0$ gives $V_p(\bar{\mathbf{x}}) \leq V_p(\mathbf{x}^*) + \sqrt{2}(|B|/n)^{1/p}$. Replacing $\sqrt{2}$ by $\max_{i \in B}\sqrt{1 + 1/|M_i|}$ in the last two steps gives the refinement mentioned after the theorem.

B.4 Proof of Theorem 3.4

The high-level idea is to construct two profiles that give the same answer to every query and are mirror images of each other under swapping s_1 and s_2. Each profile has a good lottery that leans toward one of these two alternatives, while, by convexity, every lottery is on average at least as bad as a symmetric one, which is far from optimal in both profiles. We first consider two agents and $m = 3$, and then add more agents. All thresholds are 1. In the first profile, the utility vectors of agents 1 and 2 are $(1, 0, \frac{1}{2})$ and $(\frac{1}{2}, 1, 0)$; in the second profile, they are $(1, \frac{1}{2}, 0)$ and $(0, 1, \frac{1}{2})$. In both profiles, agent 1 accepts only $\mathbf{e}_1$ and agent 2 accepts only $\mathbf{e}_2$, so the two profiles give the same answer to every membership query, and both agents belong to B. Centering and normalizing as in Section 2 gives the following violations at $\mathbf{x} \in \Delta(S)$:

$$\begin{aligned} \text{first profile:} \quad & v_1(\mathbf{x}) = \sqrt{2}\, x_2 + \tfrac{1}{\sqrt{2}}\, x_3, \quad v_2(\mathbf{x}) = \tfrac{1}{\sqrt{2}}\, x_1 + \sqrt{2}\, x_3; \\ \text{second profile:} \quad & v_1(\mathbf{x}) = \tfrac{1}{\sqrt{2}}\, x_2 + \sqrt{2}\, x_3, \quad v_2(\mathbf{x}) = \sqrt{2}\, x_1 + \tfrac{1}{\sqrt{2}}\, x_3. \end{aligned}$$

For instance, in the first profile, agent 1 has shortfall $1 - x_1 - \frac{1}{2}x_3 = x_2 + \frac{1}{2}x_3$ and centered vector $(\frac{1}{2}, -\frac{1}{2}, 0)$ of norm $1/\sqrt{2}$. Let $W_1(\mathbf{x})$ and $W_2(\mathbf{x})$ be the value of $V_p(\mathbf{x})$ in the first and second profile, respectively.

The optimal values are small. In the first profile, both violations at $(\frac{2}{3}, \frac{1}{3}, 0)$ equal $\sqrt{2}/3$, so $W_1(\frac{2}{3}, \frac{1}{3}, 0) = \sqrt{2}/3$. Similarly, $W_2(\frac{1}{3}, \frac{2}{3}, 0) = \sqrt{2}/3$. Hence the minimum of each of W_1 and W_2 over $\Delta(S)$ is at most $\sqrt{2}/3$.

Every lottery is far from optimal in one profile. Let $\sigma(\mathbf{x}) := (x_2, x_1, x_3)$. The violations of the first profile at $\sigma(\mathbf{x})$ are those of the second profile at $\mathbf{x}$ in the opposite order, so $W_1(\sigma(\mathbf{x})) = W_2(\mathbf{x})$ and, similarly, $W_2(\sigma(\mathbf{x})) = W_1(\mathbf{x})$. Hence $W_1 + W_2$ is invariant under σ. Since $W_1 + W_2$ is also convex, for every $\mathbf{x} \in \Delta(S)$,

$$W_1(\mathbf{x}) + W_2(\mathbf{x}) = \frac{1}{2}\Big((W_1 + W_2)(\mathbf{x}) + (W_1 + W_2)(\sigma(\mathbf{x}))\Big) \geq W_1(\mathbf{y}) + W_2(\mathbf{y}),$$

where $\mathbf{y} := \frac{1}{2}(\mathbf{x} + \sigma(\mathbf{x})) = \big(\frac{1-s}{2}, \frac{1-s}{2}, s\big)$ with $s := x_3$. At $\mathbf{y}$, the two violations in each profile are $\frac{1}{\sqrt{2}}$ and $\frac{1+3s}{2\sqrt{2}} \geq \frac{1}{2\sqrt{2}}$. Since the power mean is nondecreasing in each argument and at least the arithmetic mean, $W_1(\mathbf{y}) = W_2(\mathbf{y}) \geq \frac{1}{2}\big(\frac{1}{\sqrt{2}} + \frac{1}{2\sqrt{2}}\big) = \frac{3}{4\sqrt{2}}$. Therefore,

$$W_1(\mathbf{x}) + W_2(\mathbf{x}) \geq \frac{3}{2\sqrt{2}} \quad \text{for every } \mathbf{x} \in \Delta(S).$$

More agents. Let b be even with $2 \leq b \leq n$. The first (resp., second) profile with n agents consists of $b/2$ copies of the two agents of the first (resp., second) profile above, together with $n - b$ agents with utility vector $(1, 1, 0)$ and threshold 1. Each of the added agents accepts exactly the lotteries with $x_3 = 0$; she accepts $\mathbf{e}_1$ and $\mathbf{e}_2$ on her boundary and rejects only $\mathbf{e}_3$, so she is not in B. Hence $|B| = b$ in both profiles, and the two profiles give the same answer to every query. Let $V^{(1)}$ and $V^{(2)}$ denote the objective V_p in the two profiles. Since the added agents have zero violation when $x_3 = 0$, we have

$$\min_{\mathbf{x} \in \Delta(S)} V^{(1)}(\mathbf{x}) \leq V^{(1)}(\tfrac{2}{3}, \tfrac{1}{3}, 0) = \Big(\frac{b}{n}\Big)^{1/p} W_1(\tfrac{2}{3}, \tfrac{1}{3}, 0) = \Big(\frac{b}{n}\Big)^{1/p} \frac{\sqrt{2}}{3},$$

and similarly for $V^{(2)}$. Moreover, $V^{(1)}(\mathbf{x}) \geq (b/n)^{1/p} W_1(\mathbf{x})$ and $V^{(2)}(\mathbf{x}) \geq (b/n)^{1/p} W_2(\mathbf{x})$ for every $\mathbf{x}$, since the added agents contribute nonnegative terms. Combining these bounds, the additive losses of any lottery $\mathbf{x}$ in the two profiles sum to at least

$$\Big(\frac{b}{n}\Big)^{1/p} \left(W_1(\mathbf{x}) + W_2(\mathbf{x}) - \frac{2\sqrt{2}}{3}\right) \geq \Big(\frac{b}{n}\Big)^{1/p} \left(\frac{3}{2\sqrt{2}} - \frac{2\sqrt{2}}{3}\right) = \frac{1}{6\sqrt{2}} \Big(\frac{b}{n}\Big)^{1/p}.$$

Finally, an algorithm that only uses membership queries receives the same answers in the two profiles. Hence, for every realization of its internal randomness, it returns the same lottery in both profiles, and the distribution of its output $\hat{\mathbf{x}}$ is the same. Taking expectations over $\hat{\mathbf{x}}$, the expected additive losses in the two profiles sum to at least $\frac{1}{6\sqrt{2}}(b/n)^{1/p}$, so one of them is at least $\frac{1}{12\sqrt{2}}(b/n)^{1/p}$.

We chose the constant $\frac{1}{12\sqrt{2}}$ for simplicity, as it holds for all p. For specific values of p, the same construction gives larger constants: a short calculation shows that, for $b = n = 2$, every lottery has additive loss at least $\frac{1}{4\sqrt{2}}$ for $p = 1$ and at least $\frac{1}{4\sqrt{5}}$ for $p = 2$ in one of the two profiles.

B.5 Proof of Corollary 3.5

Run the algorithm of Theorem 3.1. Since $B = \varnothing$, it finds all normalized boundaries with $\mathcal{O}(nm\log(1/\varepsilon))$ queries, so every v_i is known on $\Delta(S)$. It remains to minimize V_p, or equivalently F, over $\Delta(S)$. This is equivalent to the convex program

$$\begin{array}{ll} \text{minimize} & \frac{1}{n}\sum_{i\in N} t_i^p \\ \text{subject to} & t_i \geq \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}\rangle \text{ and } t_i \geq 0 \quad \text{for all } i \in N, \\ & \mathbf{x} \in \Delta(S). \end{array} \tag{CP_p}$$

Indeed, for any $\mathbf{x} \in \Delta(S)$, choosing $t_i := v_i(\mathbf{x})$ gives a feasible solution with objective value $F(\mathbf{x})$; conversely, every feasible $(\mathbf{x}, t)$ satisfies $t_i \geq v_i(\mathbf{x})$ for all i, so its objective value is at least $F(\mathbf{x})$. Thus the optimal value of (CP_p) is $\min_{\mathbf{x}\in\Delta(S)} F(\mathbf{x})$, which is attained because F is continuous on the compact set $\Delta(S)$, and for every optimal solution $(\hat{\mathbf{x}}, \hat{t})$ of (CP_p), the lottery $\hat{\mathbf{x}}$ minimizes V_p. The algorithm outputs such a lottery $\hat{\mathbf{x}}$.

B.6 Duality for $1 < p < \infty$ and Proof of Theorem 3.6

Theorem 3.6 follows from the following duality result for the convex program (CP_p), in which the factor $1/n$ is absorbed into the dual variables.

Theorem B.1. *Fix $p \in (1, \infty)$ and let $q := p/(p-1)$. Then*

$$\min_{\mathbf{x}\in\Delta(S)} F(\mathbf{x}) = \max_{\lambda\in\mathbb{R}^n_{\geq 0}} \frac{1}{n}\left(\sum_{i\in N}\left(\lambda_i\hat{\tau}_i - \frac{p-1}{p^q}\lambda_i^q\right) - \max_{j\in[m]}\sum_{i\in N}\lambda_i\hat{u}_i(s_j)\right).$$

Moreover, $\mathbf{x}^* \in \Delta(S)$ minimizes F over $\Delta(S)$ and $\lambda^* \in \mathbb{R}^n_{\geq 0}$ is optimal for the problem on the right-hand side if and only if $\lambda_i^* = p\, v_i(\mathbf{x}^*)^{p-1}$ for all $i \in N$ and every $j \in [m]$ with $x_j^* > 0$ satisfies $\sum_{i\in N}\lambda_i^*\hat{u}_i(s_j) = \max_{\ell\in[m]}\sum_{i\in N}\lambda_i^*\hat{u}_i(s_\ell)$.

Proof. We work with the Lagrangian

$$L(\mathbf{x}, t, \lambda) = \frac{1}{n}\sum_{i\in N}\left(t_i^p + \lambda_i(\hat{\tau}_i - \langle\hat{u}_i, \mathbf{x}\rangle - t_i)\right),$$

where $\mathbf{x} \in \Delta(S)$, $t \in \mathbb{R}^n_{\geq 0}$, and $\lambda \in \mathbb{R}^n_{\geq 0}$. Since (CP_p) is a convex program and Slater's condition holds (choose any $\mathbf{x}$ in the relative interior of $\Delta(S)$ and each t_i larger than $\max\{0, \hat{\tau}_i - \langle\hat{u}_i, \mathbf{x}\rangle\}$), strong duality holds, and the dual optimum is attained.

For fixed λ, minimizing over $\mathbf{x}$ and t gives

$$\inf_{\mathbf{x}\in\Delta(S),\, t\in\mathbb{R}^n_{\geq 0}} L(\mathbf{x}, t, \lambda) = \frac{1}{n}\sum_{i\in N}\lambda_i\hat{\tau}_i - \max_{j\in[m]}\frac{1}{n}\sum_{i\in N}\lambda_i\hat{u}_i(s_j) + \frac{1}{n}\sum_{i\in N}\inf_{r\geq 0}(r^p - \lambda_i r),$$

because a linear function on $\Delta(S)$ attains its minimum at a pure lottery. For $\lambda_i \geq 0$, the function $r \mapsto r^p - \lambda_i r$ is strictly convex on $\mathbb{R}_{\geq 0}$, its minimizer is $(\lambda_i/p)^{1/(p-1)}$, and its minimum value is $-\frac{p-1}{p^q}\lambda_i^q$. This gives the dual problem in the statement.

Now let $\mathbf{x}^*$ minimize F, let $t_i^* := v_i(\mathbf{x}^*)$ for all $i \in N$, and let λ^* be dual-optimal. By the proof of Corollary 3.5, $(\mathbf{x}^*, t^*)$ is optimal for (CP_p). By strong duality, $(\mathbf{x}^*, t^*, \lambda^*)$ is a saddle point of L. Hence each t_i^* minimizes $r \mapsto r^p - \lambda_i^* r$ over $\mathbb{R}_{\geq 0}$, so $t_i^* = (\lambda_i^*/p)^{1/(p-1)}$, that is, $\lambda_i^* = p\, v_i(\mathbf{x}^*)^{p-1}$. Also, $\mathbf{x}^*$ maximizes $\mathbf{x} \mapsto \sum_{i\in N}\lambda_i^*\langle\hat{u}_i, \mathbf{x}\rangle$ over $\Delta(S)$. A linear function on the simplex is maximized exactly by the lotteries that only use alternatives with the largest coefficient, so every $j \in [m]$ with $x_j^* > 0$ satisfies $\sum_{i\in N}\lambda_i^*\hat{u}_i(s_j) = \max_{\ell\in[m]}\sum_{i\in N}\lambda_i^*\hat{u}_i(s_\ell)$.

Conversely, suppose that $\mathbf{x}^* \in \Delta(S)$ and $\lambda^* \in \mathbb{R}^n_{\geq 0}$ satisfy the two conditions, and let $t_i^* := v_i(\mathbf{x}^*)$. Then $(\mathbf{x}^*, t^*)$ is feasible for (CP_p). For each $i \in N$, since $\lambda_i^* = p\,(t_i^*)^{p-1}$, we have

$(t_i^*)^p = \lambda_i^* t_i^* - \frac{p-1}{p^q}(\lambda_i^*)^q$. Moreover, $\lambda_i^*(\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^* \rangle) = \lambda_i^* t_i^*$: if $t_i^* > 0$, then $t_i^* = \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^* \rangle$, and if $t_i^* = 0$, then $\lambda_i^* = 0$. Finally, because every j with $x_j^* > 0$ attains the largest score, $\sum_{i \in N} \lambda_i^* \langle \hat{u}_i, \mathbf{x}^* \rangle = \max_{\ell \in [m]} \sum_{i \in N} \lambda_i^* \hat{u}_i(s_\ell)$. Combining these equalities, we obtain

$$F(\mathbf{x}^*) = \frac{1}{n} \sum_{i \in N} (t_i^*)^p = \frac{1}{n} \sum_{i \in N} \left(\lambda_i^* \hat{\tau}_i - \frac{p-1}{p^q} (\lambda_i^*)^q \right) - \max_{j \in [m]} \frac{1}{n} \sum_{i \in N} \lambda_i^* \hat{u}_i(s_j),$$

that is, the primal objective value at $(\mathbf{x}^*, t^*)$ equals the dual objective value at λ^*. By weak duality, $\mathbf{x}^*$ minimizes F and λ^* is dual-optimal. □

Proof of Theorem 3.6. Minimizing V_p is equivalent to minimizing F. If $\mathbf{x}^*$ minimizes F, let λ^* be dual-optimal; by Theorem B.1, $\lambda_i^* = p\, v_i(\mathbf{x}^*)^{p-1}$, and every alternative that $\mathbf{x}^*$ uses maximizes $\sum_{i \in N} \lambda_i^* \hat{u}_i(s_j) = p \sum_{i \in N} v_i(\mathbf{x}^*)^{p-1} \hat{u}_i(s_j)$. Conversely, if every alternative that $\mathbf{x}^*$ uses maximizes $\sum_{i \in N} v_i(\mathbf{x}^*)^{p-1} \hat{u}_i(s_j)$, then $\mathbf{x}^*$ and $\lambda_i^* := p\, v_i(\mathbf{x}^*)^{p-1}$ satisfy both conditions of Theorem B.1, so $\mathbf{x}^*$ minimizes F. Finally, fix any minimizer $\mathbf{x}^*$; by Theorem B.1, every dual-optimal solution equals $(p\, v_i(\mathbf{x}^*)^{p-1})_{i \in N}$, so the dual optimum is unique. □

B.7 The Cases $p = 1$ and $p = \infty$

This subsection shows that the description after Theorem 3.6 also holds for $p = 1$ and $p = \infty$, with different rules for the weights. In both cases, every optimal lottery only uses alternatives that maximize a weighted score, where the weights form an optimal solution of a dual problem. For $p = 1$, the weights lie in $[0, 1]$, and every violated agent has weight 1. For $p = \infty$, the weights sum to at most 1, and if the optimal value is positive, only the most violated agents can have positive weight.

Theorem B.2 (Duality for $p = 1$)**.** *Let $p = 1$, so that $F = V_1$. Then $\min_{\mathbf{x} \in \Delta(S)} F(\mathbf{x}) = \max_{\lambda \in [0,1]^n} \left(\frac{1}{n} \sum_{i \in N} \lambda_i \hat{\tau}_i - \max_{j \in [m]} \frac{1}{n} \sum_{i \in N} \lambda_i \hat{u}_i(s_j) \right)$.*

Moreover, let $\mathbf{x}^* \in \Delta(S)$ and $\lambda^* \in [0, 1]^n$, and set $t_i^* := v_i(\mathbf{x}^*)$ for all $i \in N$. Then $\mathbf{x}^*$ minimizes F over $\Delta(S)$ and λ^* is dual-optimal if and only if

(i) $\lambda_i^*(\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^* \rangle - t_i^*) = 0$ *for all $i \in N$;*

(ii) $(1 - \lambda_i^*)\, t_i^* = 0$ *for all $i \in N$; and*

(iii) every $j \in [m]$ with $x_j^* > 0$ satisfies $\sum_{i \in N} \lambda_i^* \hat{u}_i(s_j) = \max_{\ell \in [m]} \sum_{i \in N} \lambda_i^* \hat{u}_i(s_\ell)$.

In particular,

$$v_i(\mathbf{x}^*) > 0 \implies \lambda_i^* = 1, \quad \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^* \rangle < 0 \implies \lambda_i^* = 0.$$

In words, every violated agent has weight 1, every agent who strictly accepts $\mathbf{x}^*$ has weight 0, and every agent whose boundary contains $\mathbf{x}^*$ may have any weight in $[0, 1]$ that is consistent with conditions (i)–(iii).

Proof. Consider the epigraph reformulation

$$\begin{array}{ll} \text{minimize} & \frac{1}{n} \sum_{i \in N} t_i \\ \text{subject to} & t_i \geq \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x} \rangle \quad \text{for each } i \in N, \\ & t_i \geq 0 \quad \text{for each } i \in N, \\ & \mathbf{x} \in \Delta(S). \end{array}$$

Its optimal value is $\min_{\mathbf{x} \in \Delta(S)} F(\mathbf{x})$, because for fixed $\mathbf{x}$ the smallest feasible choice is $t_i = v_i(\mathbf{x})$.

For $\lambda \in \mathbb{R}^n_{\geq 0}$, consider the Lagrangian

$$L(\mathbf{x}, t, \lambda) = \frac{1}{n} \sum_{i \in N} (t_i + \lambda_i(\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x} \rangle - t_i)),$$

with domain $\mathbf{x} \in \Delta(S)$ and $t \in \mathbb{R}^n_{\geq 0}$. As in the proof of Theorem B.1, Slater's condition holds, so strong duality applies. For fixed λ, we minimize over $\mathbf{x}$ as in the proof of Theorem B.1, and minimizing over each $t_i \geq 0$ gives

$$\inf_{r \geq 0} (1 - \lambda_i) r = \begin{cases} 0, & \lambda_i \in [0, 1], \\ -\infty, & \lambda_i > 1. \end{cases}$$

Therefore,

$$\min_{\mathbf{x}\in\Delta(S)} F(\mathbf{x}) = \max_{\lambda\in[0,1]^n} \left(\frac{1}{n}\sum_{i\in N} \lambda_i \hat{\tau}_i - \max_{j\in[m]} \frac{1}{n}\sum_{i\in N} \lambda_i \hat{u}_i(s_j) \right).$$

The dual feasible set $[0,1]^n$ is compact, so a dual optimum is attained.

Now let $\mathbf{x}^*$ be primal-optimal and λ^* dual-optimal, and set $t_i^* := v_i(\mathbf{x}^*)$. By strong duality, $(\mathbf{x}^*, t^*, \lambda^*)$ is a saddle point. Thus, each t_i^* minimizes $r \mapsto (1-\lambda_i^*)r$ over $r \geq 0$, which is equivalent to

$$(1-\lambda_i^*)\, t_i^* = 0 \quad \text{for each } i \in N.$$

Complementary slackness for the constraints $t_i \geq \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}\rangle$ gives

$$\lambda_i^* (\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^*\rangle - t_i^*) = 0 \quad \text{for each } i \in N.$$

Finally, $\mathbf{x}^*$ minimizes $\mathbf{x} \mapsto -\sum_{i\in N} \lambda_i^* \langle \hat{u}_i, \mathbf{x}\rangle$ over $\Delta(S)$, or equivalently maximizes $\mathbf{x} \mapsto \sum_{i\in N} \lambda_i^* \langle \hat{u}_i, \mathbf{x}\rangle$. A linear function on the simplex is maximized exactly by the lotteries that only use alternatives with the largest coefficient, so every j with $x_j^* > 0$ satisfies

$$\sum_{i\in N} \lambda_i^* \hat{u}_i(s_j) = \max_{\ell\in[m]} \sum_{i\in N} \lambda_i^* \hat{u}_i(s_\ell).$$

Conversely, suppose $\mathbf{x}^* \in \Delta(S)$ and $\lambda^* \in [0,1]^n$ satisfy the displayed conditions, and set $t_i^* := v_i(\mathbf{x}^*)$. Then $(\mathbf{x}^*, t^*)$ is primal-feasible and λ^* is dual-feasible. Since $(1-\lambda_i^*)t_i^* = 0$, we have $t_i^* = \lambda_i^* t_i^*$ for every i. Using also $\lambda_i^*(\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^*\rangle - t_i^*) = 0$, we obtain

$$t_i^* = \lambda_i^* t_i^* = \lambda_i^* (\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^*\rangle) \quad \text{for each } i \in N.$$

Summing over i and using condition (iii) gives

$$\frac{1}{n}\sum_{i\in N} t_i^* = \frac{1}{n}\sum_{i\in N} \lambda_i^* \hat{\tau}_i - \max_{j\in[m]} \frac{1}{n}\sum_{i\in N} \lambda_i^* \hat{u}_i(s_j).$$

Thus, the primal objective value at $(\mathbf{x}^*, t^*)$ equals the dual objective value at λ^*. By weak duality, both are optimal.

The final implications are immediate: if $v_i(\mathbf{x}^*) = t_i^* > 0$, then $(1-\lambda_i^*)t_i^* = 0$ forces $\lambda_i^* = 1$; and if $\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^*\rangle < 0$, then complementary slackness forces $\lambda_i^* = 0$. □

Theorem B.3 (Duality for $p = \infty$)**.** *We have*

$$\min_{\mathbf{x}\in\Delta(S)} V_\infty(\mathbf{x}) = \max_{\lambda\in\mathbb{R}^n_{\geq 0}:\, \sum_{i\in N}\lambda_i \leq 1} \left(\sum_{i\in N} \lambda_i \hat{\tau}_i - \max_{j\in[m]} \sum_{i\in N} \lambda_i \hat{u}_i(s_j) \right).$$

Moreover, let $\mathbf{x}^* \in \Delta(S)$ and $\lambda^* \in \mathbb{R}^n_{\geq 0}$, and set $t^* := V_\infty(\mathbf{x}^*)$. Then $\mathbf{x}^*$ minimizes V_∞ over $\Delta(S)$ and λ^* is optimal for the displayed dual problem if and only if

- *(i)* $\sum_{i\in N} \lambda_i^* \leq 1$*;*
- *(ii)* $\lambda_i^*(\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^*\rangle - t^*) = 0$ *for all* $i \in N$*;*
- *(iii)* $t^* \left(1 - \sum_{i\in N} \lambda_i^*\right) = 0$*; and*
- *(iv) every* $j \in [m]$ *with* $x_j^* > 0$ *satisfies* $\sum_{i\in N} \lambda_i^* \hat{u}_i(s_j) = \max_{\ell\in[m]} \sum_{i\in N} \lambda_i^* \hat{u}_i(s_\ell)$*.*

In particular, if $V_\infty(\mathbf{x}^*) > 0$, then $\sum_{i\in N} \lambda_i^* = 1$ and

$$\lambda_i^* > 0 \implies v_i(\mathbf{x}^*) = V_\infty(\mathbf{x}^*) \quad \textit{for every } i \in N.$$

Proof. Consider the linear program

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & t \geq \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}\rangle \quad (i \in N), \\ & t \geq 0, \\ & \mathbf{x} \in \Delta(S). \end{array} \tag{CP$_\infty$}$$

For any fixed $\mathbf{x} \in \Delta(S)$, the smallest t for which $(\mathbf{x}, t)$ is feasible is

$$\max\{0, \max_{i \in N}(\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x} \rangle)\} = \max_{i \in N} \max\{0, \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x} \rangle\} = V_\infty(\mathbf{x}).$$

Thus, the optimal value of (CP_∞) is exactly $\min_{\mathbf{x} \in \Delta(S)} V_\infty(\mathbf{x})$, and for every optimal solution $(\mathbf{x}^*, t^*)$ of (CP_∞) we have $t^* = V_\infty(\mathbf{x}^*)$.

For $\lambda \in \mathbb{R}^n_{\geq 0}$, the Lagrangian of (CP_∞) is

$$L(\mathbf{x}, t, \lambda) = t + \sum_{i \in N} \lambda_i (\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x} \rangle - t),$$

with domain $\mathbf{x} \in \Delta(S)$ and $t \in \mathbb{R}_{\geq 0}$. Equivalently,

$$L(\mathbf{x}, t, \lambda) = \sum_{i \in N} \lambda_i \hat{\tau}_i - \langle \sum_{i \in N} \lambda_i \hat{u}_i, \mathbf{x} \rangle + t \left(1 - \sum_{i \in N} \lambda_i\right).$$

Minimizing over $\mathbf{x} \in \Delta(S)$ gives

$$\inf_{\mathbf{x} \in \Delta(S)} L(\mathbf{x}, t, \lambda) = \sum_{i \in N} \lambda_i \hat{\tau}_i - \max_{j \in [m]} \sum_{i \in N} \lambda_i \hat{u}_i(s_j) + t \left(1 - \sum_{i \in N} \lambda_i\right),$$

because a linear function on the simplex attains its minimum at a vertex. Therefore

$$\inf_{\mathbf{x} \in \Delta(S),\, t \in \mathbb{R}_{\geq 0}} L(\mathbf{x}, t, \lambda) = \begin{cases} \sum_{i \in N} \lambda_i \hat{\tau}_i - \max_{j \in [m]} \sum_{i \in N} \lambda_i \hat{u}_i(s_j), & \text{if } \sum_{i \in N} \lambda_i \leq 1, \\ -\infty, & \text{if } \sum_{i \in N} \lambda_i > 1. \end{cases}$$

Thus the dual problem is exactly

$$\max_{\lambda \in \mathbb{R}^n_{\geq 0}:\, \sum_{i \in N} \lambda_i \leq 1} \left(\sum_{i \in N} \lambda_i \hat{\tau}_i - \max_{j \in [m]} \sum_{i \in N} \lambda_i \hat{u}_i(s_j) \right).$$

The primal optimum is attained because V_∞ is continuous on the compact set $\Delta(S)$. The dual feasible set $\{\lambda \in \mathbb{R}^n_{\geq 0} : \sum_{i \in N} \lambda_i \leq 1\}$ is compact, and the dual objective is continuous, so the dual optimum is attained. Since (CP_∞) is a convex program and Slater's condition holds (for example, take $\mathbf{x} = (1/m, \ldots, 1/m)$ and choose t strictly larger than $\max\{0, \max_i(\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x} \rangle)\}$), strong duality applies.

We now prove the KKT characterization. Let $(\mathbf{x}^*, t^*)$ be primal-optimal for (CP_∞) and let λ^* be dual-optimal. By strong duality, $(\mathbf{x}^*, t^*, \lambda^*)$ is a saddle point of the Lagrangian. Thus, complementary slackness gives

$$\lambda_i^*(\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^* \rangle - t^*) = 0 \quad \text{for each } i \in N.$$

Also, t^* minimizes the function $t \mapsto t(1 - \sum_{i \in N} \lambda_i^*)$ over $\mathbb{R}_{\geq 0}$. This is equivalent to

$$\sum_{i \in N} \lambda_i^* \leq 1 \quad \text{and} \quad t^* \left(1 - \sum_{i \in N} \lambda_i^*\right) = 0.$$

Finally, $\mathbf{x}^*$ minimizes $\mathbf{x} \mapsto -\sum_{i \in N} \lambda_i^* \langle \hat{u}_i, \mathbf{x} \rangle$ over $\Delta(S)$, or equivalently maximizes $\mathbf{x} \mapsto \sum_{i \in N} \lambda_i^* \langle \hat{u}_i, \mathbf{x} \rangle$. A linear function on the simplex is maximized exactly by the lotteries that only use alternatives with the largest coefficient, so every $j \in [m]$ with $x_j^* > 0$ satisfies

$$\sum_{i \in N} \lambda_i^* \hat{u}_i(s_j) = \max_{\ell \in [m]} \sum_{i \in N} \lambda_i^* \hat{u}_i(s_\ell).$$

Conversely, suppose $\mathbf{x}^* \in \Delta(S)$ and $\lambda^* \in \mathbb{R}^n_{\geq 0}$ satisfy the displayed conditions, and set $t^* := V_\infty(\mathbf{x}^*)$. Then $(\mathbf{x}^*, t^*)$ is primal-feasible for (CP_∞), and λ^* is dual-feasible. By condition (iv),

$$\sum_{i \in N} \lambda_i^* \langle \hat{u}_i, \mathbf{x}^* \rangle = \sum_{j=1}^m x_j^* \sum_{i \in N} \lambda_i^* \hat{u}_i(s_j) = \max_{\ell \in [m]} \sum_{i \in N} \lambda_i^* \hat{u}_i(s_\ell).$$

Summing condition (ii) over $i \in N$ and using the previous display, we obtain

$$\sum_{i \in N} \lambda_i^* \hat{\tau}_i - \max_{\ell \in [m]} \sum_{i \in N} \lambda_i^* \hat{u}_i(s_\ell) = t^* \sum_{i \in N} \lambda_i^* = t^*,$$

where the last equality follows from condition (iii). Thus, the dual objective value at λ^* equals the primal objective value at $(\mathbf{x}^*, t^*)$. By weak duality, both are optimal.

For the final claim, if $V_\infty(\mathbf{x}^*) = t^* > 0$, then $t^* \left(1 - \sum_{i \in N} \lambda_i^*\right) = 0$ forces $\sum_{i \in N} \lambda_i^* = 1$. And if $\lambda_i^* > 0$, then complementary slackness gives $\hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^* \rangle = t^* > 0$ and thus

$$v_i(\mathbf{x}^*) = \max\{0, \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x}^* \rangle\} = t^* = V_\infty(\mathbf{x}^*).$$

This proves the theorem. □

B.8 Proof of Theorem 3.7

For each alternative s_j, let $\mathbf{g}_j := (\hat{u}_i(s_j))_{i \in N} \in \mathbb{R}^n$, so that $\sum_{j=1}^m x_j \mathbf{g}_j = (\langle \hat{u}_i, \mathbf{x} \rangle)_{i \in N}$ for every lottery $\mathbf{x}$. Since $v_i(\mathbf{x}) = \max\{0, \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x} \rangle\}$, any two lotteries $\mathbf{x}$ and $\mathbf{y}$ with $\sum_{j=1}^m x_j \mathbf{g}_j = \sum_{j=1}^m y_j \mathbf{g}_j$ satisfy $v_i(\mathbf{x}) = v_i(\mathbf{y})$ for every $i \in N$.

Let J be the set of indices of the alternatives that $\mathbf{x}^*$ uses. The point $\sum_{j \in J} x_j^* \mathbf{g}_j$ lies in the convex hull of $\{\mathbf{g}_j : j \in J\} \subseteq \mathbb{R}^n$. By Carathéodory's theorem, there exist a set $J' \subseteq J$ with $|J'| \leq n + 1$ and nonnegative numbers $(\mu_j)_{j \in J'}$ with $\sum_{j \in J'} \mu_j = 1$ such that $\sum_{j \in J'} \mu_j \mathbf{g}_j = \sum_{j \in J} x_j^* \mathbf{g}_j$. Let $\bar{\mathbf{x}}$ be the lottery with $\bar{x}_j := \mu_j$ for $j \in J'$ and $\bar{x}_j := 0$ for $j \notin J'$. Then $\bar{\mathbf{x}}$ uses at most $n+1$ alternatives, all of which are used by $\mathbf{x}^*$, and $v_i(\bar{\mathbf{x}}) = v_i(\mathbf{x}^*)$ for every $i \in N$. Hence $V_p(\bar{\mathbf{x}}) = V_p(\mathbf{x}^*)$, so $\bar{\mathbf{x}}$ also minimizes V_p.

B.9 Approximately Optimal Lotteries with Few Alternatives

We continue to assume that the normalized boundaries are known. Theorem 3.7 shows that some optimal lottery uses at most $n + 1$ alternatives. We now give two results in which the number of alternatives does not depend on n. First, since V_p is 1-Lipschitz on $\Delta(S)$, sampling alternatives from an optimal lottery gives a lottery that uses at most $\lceil \alpha^{-2} \rceil$ alternatives and is within additive α of the optimum (Theorem B.4). Second, for $p \geq 2$, the gradient of F is Lipschitz on $\Delta(S)$, so the Frank–Wolfe method reduces F at the usual rate $\mathcal{O}(1/T)$, while its T-th iterate uses at most $T + 1$ alternatives (Theorem B.5). Each step of this method adds an alternative with the largest weighted score $\sum_{i \in N} v_i(\mathbf{x})^{p-1} \hat{u}_i(s_j)$ at the current lottery $\mathbf{x}$, which matches the weights in Theorem 3.6.

Theorem B.4. *Fix $p \in [1, \infty)$, and let $\mathbf{x}^*$ be a minimizer of V_p over $\Delta(S)$. For every positive integer s, there is a lottery $\mathbf{x}'$ that uses at most s alternatives, all of which are used by $\mathbf{x}^*$, such that $V_p(\mathbf{x}') \leq V_p(\mathbf{x}^*) + 1/\sqrt{s}$. In particular, for every $\alpha > 0$, some lottery $\mathbf{x}'$ that uses at most $\lceil \alpha^{-2} \rceil$ alternatives satisfies $V_p(\mathbf{x}') \leq \min_{\mathbf{x} \in \Delta(S)} V_p(\mathbf{x}) + \alpha$.*

Proof. We first show that V_p is 1-Lipschitz on $\Delta(S)$. Fix $\mathbf{x}, \mathbf{y} \in \Delta(S)$ and $i \in N$. Since $z \mapsto \max\{0, z\}$ is 1-Lipschitz and $\|\hat{u}_i\| = 1$, we have $|v_i(\mathbf{x}) - v_i(\mathbf{y})| \leq |\langle \hat{u}_i, \mathbf{x} - \mathbf{y} \rangle| \leq \|\mathbf{x} - \mathbf{y}\|$. Hence, by the reverse triangle inequality for the ℓ_p-norm,

$$|V_p(\mathbf{x}) - V_p(\mathbf{y})| \leq \left(\frac{1}{n} \sum_{i \in N} |v_i(\mathbf{x}) - v_i(\mathbf{y})|^p \right)^{1/p} \leq \|\mathbf{x} - \mathbf{y}\|.$$

Now draw $J_1, \ldots, J_s$ independently from the distribution $\mathbf{x}^*$ on $[m]$, and let $\bar{\mathbf{x}} := \frac{1}{s} \sum_{t=1}^s \mathbf{e}_{J_t}$. Then $\bar{\mathbf{x}}$ is a lottery that uses at most s alternatives, all of which are used by $\mathbf{x}^*$. Since $\mathbb{E}[\mathbf{e}_{J_t}] = \mathbf{x}^*$, the vectors $\mathbf{e}_{J_t} - \mathbf{x}^*$ are independent with mean zero, so the cross terms vanish and

$$\mathbb{E}\|\bar{\mathbf{x}} - \mathbf{x}^*\|^2 = \frac{1}{s^2} \sum_{t=1}^s \mathbb{E}\|\mathbf{e}_{J_t} - \mathbf{x}^*\|^2 = \frac{1 - \|\mathbf{x}^*\|^2}{s} \leq \frac{1}{s},$$

where we used $\mathbb{E}\|\mathbf{e}_{J_t} - \mathbf{x}^*\|^2 = \mathbb{E}\|\mathbf{e}_{J_t}\|^2 - 2\langle \mathbb{E}[\mathbf{e}_{J_t}], \mathbf{x}^* \rangle + \|\mathbf{x}^*\|^2 = 1 - \|\mathbf{x}^*\|^2$. By the Lipschitz bound and Jensen's inequality,

$$\mathbb{E}[V_p(\bar{\mathbf{x}})] \leq V_p(\mathbf{x}^*) + \mathbb{E}\|\bar{\mathbf{x}} - \mathbf{x}^*\| \leq V_p(\mathbf{x}^*) + \sqrt{\mathbb{E}\|\bar{\mathbf{x}} - \mathbf{x}^*\|^2} \leq V_p(\mathbf{x}^*) + \frac{1}{\sqrt{s}}.$$

Therefore some realization $\mathbf{x}'$ of $\bar{\mathbf{x}}$ satisfies $V_p(\mathbf{x}') \le V_p(\mathbf{x}^*) + 1/\sqrt{s}$. The last sentence of the theorem follows by taking $s = \lceil \alpha^{-2} \rceil$. □

Theorem B.5. *Fix $p \in [2, \infty)$ and let $L_p := p(p-1)2^{(p-2)/2}$. Consider the Frank–Wolfe method for minimizing F over $\Delta(S)$: start from any pure lottery $\mathbf{x}^{(0)}$, and for each $t \ge 0$, choose $j_t \in \arg\max_{j \in [m]} \sum_{i \in N} v_i(\mathbf{x}^{(t)})^{p-1} \hat{u}_i(s_j)$ and let $\mathbf{x}^{(t+1)} := (1-\gamma_t)\mathbf{x}^{(t)} + \gamma_t \mathbf{e}_{j_t}$, where $\gamma_t := \frac{2}{t+2}$. Let $\mathbf{x}^*$ be a minimizer of F over $\Delta(S)$. Then, for every integer $T \ge 0$:*

(i) $\mathbf{x}^{(T)}$ uses at most $T+1$ alternatives;

(ii) $F(\mathbf{x}^{(T)}) - F(\mathbf{x}^*) \le \frac{4L_p}{T+2}$;

(iii) $V_p(\mathbf{x}^{(T)}) \le \left(V_p(\mathbf{x}^*)^p + \frac{4L_p}{T+2} \right)^{1/p} \le V_p(\mathbf{x}^*) + \left(\frac{4L_p}{T+2} \right)^{1/p}$. If, moreover, $V_p(\mathbf{x}^*) > 0$, then $V_p(\mathbf{x}^{(T)}) - V_p(\mathbf{x}^*) \le \frac{4L_p}{p(T+2)V_p(\mathbf{x}^*)^{p-1}} = \frac{4(p-1)2^{(p-2)/2}}{(T+2)V_p(\mathbf{x}^*)^{p-1}}$.

Proof. Extend F to $\mathbb{R}^m$ by the same formula,

$$F(\mathbf{x}) := \frac{1}{n} \sum_{i \in N} (\max\{0, \hat{\tau}_i - \langle \hat{u}_i, \mathbf{x} \rangle\})^p, \quad \mathbf{x} \in \mathbb{R}^m.$$

For $p > 1$, the scalar function $z \mapsto (\max\{0, z\})^p$ is differentiable on $\mathbb{R}$, with derivative $p(\max\{0, z\})^{p-1}$. Thus, this ambient extension is differentiable on $\mathbb{R}^m$, and for every $\mathbf{x} \in \Delta(S)$,

$$\nabla F(\mathbf{x}) = -\frac{p}{n} \sum_{i \in N} v_i(\mathbf{x})^{p-1} \hat{u}_i.$$

Therefore, each step of the Frank–Wolfe method chooses a pure lottery $\mathbf{e}_j$ minimizing $\langle \nabla F(\mathbf{x}), \mathbf{e}_j \rangle$, or equivalently maximizing $\sum_{i \in N} v_i(\mathbf{x})^{p-1} \hat{u}_i(s_j)$.

We next prove that ∇F is L_p-Lipschitz on $\Delta(S)$. Fix $\mathbf{x}, \mathbf{y} \in \Delta(S)$ and $i \in N$. Since $v_i \le \sqrt{2}$ on $\Delta(S)$, the derivative of $z \mapsto z^{p-1}$ is at most $(p-1)(\sqrt{2})^{p-2}$ on $[0, \sqrt{2}]$, and $|v_i(\mathbf{x}) - v_i(\mathbf{y})| \le |\langle \hat{u}_i, \mathbf{x} - \mathbf{y} \rangle|$, we have

$$|v_i(\mathbf{x})^{p-1} - v_i(\mathbf{y})^{p-1}| \le (p-1)(\sqrt{2})^{p-2} |\langle \hat{u}_i, \mathbf{x} - \mathbf{y} \rangle|.$$

Therefore,

$$\|\nabla F(\mathbf{x}) - \nabla F(\mathbf{y})\| = \left\| \frac{p}{n} \sum_{i \in N} (v_i(\mathbf{x})^{p-1} - v_i(\mathbf{y})^{p-1}) \hat{u}_i \right\| \le \frac{p}{n} \sum_{i \in N} |v_i(\mathbf{x})^{p-1} - v_i(\mathbf{y})^{p-1}| \, \|\hat{u}_i\|$$

$$\le p(p-1)(\sqrt{2})^{p-2} \frac{1}{n} \sum_{i \in N} |\langle \hat{u}_i, \mathbf{x} - \mathbf{y} \rangle| \le p(p-1)(\sqrt{2})^{p-2} \|\mathbf{x} - \mathbf{y}\| = L_p \|\mathbf{x} - \mathbf{y}\|.$$

Let $h_t := F(\mathbf{x}^{(t)}) - F(\mathbf{x}^*)$. Since ∇F is L_p-Lipschitz on the convex set $\Delta(S)$, the standard smoothness inequality gives

$$F((1-\gamma)\mathbf{x} + \gamma \mathbf{z}) \le F(\mathbf{x}) + \gamma \langle \nabla F(\mathbf{x}), \mathbf{z} - \mathbf{x} \rangle + \frac{L_p}{2} \gamma^2 \|\mathbf{z} - \mathbf{x}\|^2$$

for all $\mathbf{x}, \mathbf{z} \in \Delta(S)$ and all $\gamma \in [0, 1]$. Applying this with $\mathbf{x} = \mathbf{x}^{(t)}$, $\mathbf{z} = \mathbf{e}_{j_t}$, and $\gamma = \gamma_t$, and using $\|\mathbf{e}_{j_t} - \mathbf{x}^{(t)}\| \le \sqrt{2}$, gives

$$F(\mathbf{x}^{(t+1)}) \le F(\mathbf{x}^{(t)}) + \gamma_t \langle \nabla F(\mathbf{x}^{(t)}), \mathbf{e}_{j_t} - \mathbf{x}^{(t)} \rangle + L_p \gamma_t^2.$$

By the choice of $\mathbf{e}_{j_t}$,

$$\langle \nabla F(\mathbf{x}^{(t)}), \mathbf{e}_{j_t} - \mathbf{x}^{(t)} \rangle \le \langle \nabla F(\mathbf{x}^{(t)}), \mathbf{x}^* - \mathbf{x}^{(t)} \rangle.$$

Since F is convex,

$$\langle \nabla F(\mathbf{x}^{(t)}), \mathbf{x}^* - \mathbf{x}^{(t)} \rangle \le F(\mathbf{x}^*) - F(\mathbf{x}^{(t)}) = -h_t.$$

Thus, $h_{t+1} \le (1-\gamma_t)h_t + L_p\gamma_t^2$. With $\gamma_t = 2/(t+2)$, we prove by induction that

$$h_t \le \frac{4L_p}{t+2} \quad \text{for all } t \ge 0.$$

For $t = 0$, since every $v_i(\mathbf{x}^{(0)}) \le \sqrt{2}$,

$$h_0 \le F(\mathbf{x}^{(0)}) \le (\sqrt{2})^p \le 2L_p = \frac{4L_p}{2},$$

because $p(p-1) \ge 1$ for $p \ge 2$. Now assume $h_t \le 4L_p/(t+2)$. Then

$$h_{t+1} \le \left(1 - \frac{2}{t+2}\right)\frac{4L_p}{t+2} + L_p\frac{4}{(t+2)^2} = \frac{4L_p(t+1)}{(t+2)^2} \le \frac{4L_p}{t+3},$$

which closes the induction. This proves (ii).

Part (i) holds because $\mathbf{x}^{(0)}$ is a pure lottery and each update writes $\mathbf{x}^{(t+1)}$ as a convex combination of $\mathbf{x}^{(t)}$ and one pure lottery $\mathbf{e}_{j_t}$, so each step adds at most one alternative.

For (iii), use $F(\mathbf{x}) = V_p(\mathbf{x})^p$:

$$V_p(\mathbf{x}^{(T)})^p - V_p(\mathbf{x}^*)^p = F(\mathbf{x}^{(T)}) - F(\mathbf{x}^*) \le \frac{4L_p}{T+2}.$$

Consequently,

$$V_p(\mathbf{x}^{(T)}) \le \left(V_p(\mathbf{x}^*)^p + \frac{4L_p}{T+2}\right)^{1/p}.$$

The displayed additive bound by $(4L_p/(T+2))^{1/p}$ follows from $(a+b)^{1/p} \le a^{1/p} + b^{1/p}$ for $a, b \ge 0$. If $V_p(\mathbf{x}^*) > 0$, then the function $r \mapsto r^{1/p}$ is concave on $\mathbb{R}_{\ge 0}$, so

$$(a+b)^{1/p} - a^{1/p} \le \frac{1}{p}a^{1/p-1}b \quad \text{for all } a > 0 \text{ and } b \ge 0.$$

Applying this with $a = V_p(\mathbf{x}^*)^p$ and $b = 4L_p/(T+2)$ gives

$$V_p(\mathbf{x}^{(T)}) - V_p(\mathbf{x}^*) \le \frac{4L_p}{p(T+2)V_p(\mathbf{x}^*)^{p-1}} = \frac{4(p-1)2^{(p-2)/2}}{(T+2)V_p(\mathbf{x}^*)^{p-1}},$$

as claimed. □

C Omitted Proofs and Details from Section 4

C.1 Proof of Theorem 4.2

By Step 1 of Algorithm 1, $kT(\eta) \le Q$, and each of the k invocations of the subroutine uses at most $T(\eta)$ membership queries. Thus, Algorithm 1 uses at most Q membership queries. Let $\hat{\mathbf{x}}$ denote the lottery returned by Algorithm 1, and let $\mathbf{x}^*$ be a minimizer of F over $\Delta(S)$. We compare F with the objective of the sampled agents and with the objective that the algorithm minimizes, namely,

$$F_k(\mathbf{x}) := \frac{1}{k}\sum_{t=1}^{k} v_{i_t}(\mathbf{x})^p \quad \text{and} \quad \widetilde{F}_k(\mathbf{x}) := \frac{1}{k}\sum_{t=1}^{k} \tilde{v}_t(\mathbf{x})^p,$$

where $\tilde{v}_t$ is defined in Algorithm 1. The difference between $\widetilde{F}_k$ and F_k comes from boundary learning, and the difference between F_k and F comes from sampling. By the guarantee of `LearnHalfspace`, for each $t \in [k]$,

$$\|\tilde{u}_t - \hat{u}_{i_t}\| \le \eta, \quad |\tilde{\tau}_t - \hat{\tau}_{i_t}| \le \eta.$$

Since $z \mapsto \max\{0, z\}$ is 1-Lipschitz and $\|\mathbf{x}\| \le 1$ for all $\mathbf{x} \in \Delta(S)$, for every $\mathbf{x} \in \Delta(S)$,

$$|\tilde{v}_t(\mathbf{x}) - v_{i_t}(\mathbf{x})| \le |(\tilde{\tau}_t - \langle \tilde{u}_t, \mathbf{x}\rangle) - (\hat{\tau}_{i_t} - \langle \hat{u}_{i_t}, \mathbf{x}\rangle)| \le |\tilde{\tau}_t - \hat{\tau}_{i_t}| + \|\tilde{u}_t - \hat{u}_{i_t}\|\,\|\mathbf{x}\| \le 2\eta.$$

Since $v_i \le \sqrt{2}$ on $\Delta(S)$ for every agent i (Section 2), it follows that $\tilde{v}_t(\mathbf{x}) \le v_{i_t}(\mathbf{x}) + 2\eta \le \sqrt{2} + 2\eta$ for every $t \in [k]$ and $\mathbf{x} \in \Delta(S)$.

Since $\tilde{v}_t(\mathbf{x}), v_{i_t}(\mathbf{x}) \in [0, \sqrt{2}+2\eta]$ with $p \geq 1$, the mean value theorem gives, uniformly over $\mathbf{x} \in \Delta(S)$,

$$|\tilde{v}_t(\mathbf{x})^p - v_{i_t}(\mathbf{x})^p| \leq p(\sqrt{2}+2\eta)^{p-1} \, |\tilde{v}_t(\mathbf{x}) - v_{i_t}(\mathbf{x})| \leq 2p\eta(\sqrt{2}+2\eta)^{p-1}.$$

Averaging over t gives

$$\sup_{\mathbf{x}\in\Delta(S)} |\widetilde{F}_k(\mathbf{x}) - F_k(\mathbf{x})| \leq 2p\eta(\sqrt{2}+2\eta)^{p-1}. \tag{5}$$

Then,

$$\begin{aligned} F(\hat{\mathbf{x}}) - F(\mathbf{x}^*) &= (F(\hat{\mathbf{x}}) - F_k(\hat{\mathbf{x}})) + (F_k(\hat{\mathbf{x}}) - \widetilde{F}_k(\hat{\mathbf{x}})) + (\widetilde{F}_k(\hat{\mathbf{x}}) - \widetilde{F}_k(\mathbf{x}^*)) \\ &\quad + (\widetilde{F}_k(\mathbf{x}^*) - F_k(\mathbf{x}^*)) + (F_k(\mathbf{x}^*) - F(\mathbf{x}^*)) \\ &\leq |F(\hat{\mathbf{x}}) - F_k(\hat{\mathbf{x}})| + |F_k(\hat{\mathbf{x}}) - \widetilde{F}_k(\hat{\mathbf{x}})| + |\widetilde{F}_k(\mathbf{x}^*) - F_k(\mathbf{x}^*)| + |F_k(\mathbf{x}^*) - F(\mathbf{x}^*)| \\ &\leq 2 \sup_{\mathbf{x}\in\Delta(S)} |F(\mathbf{x}) - F_k(\mathbf{x})| + 2 \sup_{\mathbf{x}\in\Delta(S)} |\widetilde{F}_k(\mathbf{x}) - F_k(\mathbf{x})|. \end{aligned}$$

Combining this with (5), we obtain

$$F(\hat{\mathbf{x}}) \leq F(\mathbf{x}^*) + 2 \sup_{\mathbf{x}\in\Delta(S)} |F(\mathbf{x}) - F_k(\mathbf{x})| + 4p\eta(\sqrt{2}+2\eta)^{p-1}. \tag{6}$$

Let $\sigma_1, \ldots, \sigma_k$ be i.i.d. Rademacher signs (uniform on $\{\pm 1\}$), independent of $i_1, \ldots, i_k$. Standard symmetrization gives

$$\mathbb{E}\left[\sup_{\mathbf{x}\in\Delta(S)} (F(\mathbf{x}) - F_k(\mathbf{x}))\right] \leq \frac{2}{k} \, \mathbb{E}\left[\sup_{\mathbf{x}\in\Delta(S)} \sum_{t=1}^{k} \sigma_t v_{i_t}(\mathbf{x})^p\right]. \tag{7}$$

Since $v_{i_t}(\mathbf{x}) = \max\{0, \hat{\tau}_{i_t} - \langle \hat{u}_{i_t}, \mathbf{x}\rangle\} \leq \sqrt{2}$ for all $\mathbf{x} \in \Delta(S)$, the function $z \mapsto \max\{0, z\}^p$ is $p(\sqrt{2})^{p-1}$-Lipschitz on the set $\{\hat{\tau}_{i_t} - \langle \hat{u}_{i_t}, \mathbf{x}\rangle : \mathbf{x} \in \Delta(S)\} \subseteq (-\infty, \sqrt{2}]$. By the contraction inequality for Rademacher averages,

$$\begin{aligned} \mathbb{E}\left[\sup_{\mathbf{x}\in\Delta(S)} \sum_{t=1}^{k} \sigma_t \, v_{i_t}(\mathbf{x})^p\right] &\leq p(\sqrt{2})^{p-1} \, \mathbb{E}\left[\sup_{\mathbf{x}\in\Delta(S)} \sum_{t=1}^{k} \sigma_t \, (\hat{\tau}_{i_t} - \langle \hat{u}_{i_t}, \mathbf{x}\rangle)\right] \\ &= p(\sqrt{2})^{p-1} \, \mathbb{E}\left[\sup_{\mathbf{x}\in\Delta(S)} \left\langle -\sum_{t=1}^{k} \sigma_t \hat{u}_{i_t}, \mathbf{x} \right\rangle\right], \end{aligned}$$

where the equality holds because $\sum_{t=1}^{k} \sigma_t \hat{\tau}_{i_t}$ does not depend on $\mathbf{x}$ and has zero expectation over (σ_t). By the Cauchy–Schwarz inequality and since $\|\mathbf{x}\| \leq 1$ for all $\mathbf{x} \in \Delta(S)$,

$$\sup_{\mathbf{x}\in\Delta(S)} \left\langle -\sum_{t=1}^{k} \sigma_t \hat{u}_{i_t}, \mathbf{x} \right\rangle \leq \left\| \sum_{t=1}^{k} \sigma_t \hat{u}_{i_t} \right\|.$$

Finally, since $\|\hat{u}_i\| = 1$ for all i,

$$\mathbb{E}\left[\left\| \sum_{t=1}^{k} \sigma_t \hat{u}_{i_t} \right\|\right] \leq \sqrt{\mathbb{E}\left[\left\| \sum_{t=1}^{k} \sigma_t \hat{u}_{i_t} \right\|^2\right]} = \sqrt{\sum_{t=1}^{k} \mathbb{E}[\|\hat{u}_{i_t}\|^2]} = \sqrt{k}.$$

Combining this with (7) gives

$$\mathbb{E}\left[\sup_{\mathbf{x}\in\Delta(S)} (F(\mathbf{x}) - F_k(\mathbf{x}))\right] \leq \frac{2p(\sqrt{2})^{p-1}}{\sqrt{k}}. \tag{8}$$

By the same argument, the same bound holds for $\mathbb{E}[\sup_{\mathbf{x}\in\Delta(S)}(F_k(\mathbf{x}) - F(\mathbf{x}))]$.

Next, we apply McDiarmid's inequality. For any fixed $\mathbf{x} \in \Delta(S)$, $F_k(\mathbf{x})$ is an average of k numbers in $[0, (\sqrt{2})^p]$, so changing one sampled index i_t can change $F_k(\mathbf{x})$ by at most $(\sqrt{2})^p/k$. Thus,

changing one i_t can change $\sup_{\mathbf{x}\in\Delta(S)}(F(\mathbf{x})-F_k(\mathbf{x}))$ by at most $(\sqrt{2})^p/k$ as well. Therefore, with probability at least $1-\delta/2$,

$$\sup_{\mathbf{x}\in\Delta(S)}(F(\mathbf{x})-F_k(\mathbf{x}))\le \mathbb{E}\left[\sup_{\mathbf{x}\in\Delta(S)}(F(\mathbf{x})-F_k(\mathbf{x}))\right]+(\sqrt{2})^p\sqrt{\frac{\log(2/\delta)}{2k}}.$$

Applying the same argument to $\sup_{\mathbf{x}\in\Delta(S)}(F_k(\mathbf{x})-F(\mathbf{x}))$ and taking a union bound over the two events, with probability at least $1-\delta$ we have

$$\sup_{\mathbf{x}\in\Delta(S)}|F(\mathbf{x})-F_k(\mathbf{x})|\le \frac{2p(\sqrt{2})^{p-1}}{\sqrt{k}}+(\sqrt{2})^p\sqrt{\frac{\log(2/\delta)}{2k}}. \tag{9}$$

By the definition of $\rho_k(\delta)$, inequality (9) states that $\sup_{\mathbf{x}\in\Delta(S)}|F(\mathbf{x})-F_k(\mathbf{x})|\le\rho_k(\delta)$ with probability at least $1-\delta$. Substituting this into (6), we obtain

$$F(\hat{\mathbf{x}})\le F(\mathbf{x}^*)+4p\eta(\sqrt{2}+2\eta)^{p-1}+2\rho_k(\delta).$$

Since $V_p(\mathbf{x})=F(\mathbf{x})^{1/p}$, we obtain

$$V_p(\hat{\mathbf{x}})=F(\hat{\mathbf{x}})^{1/p}\le(F(\mathbf{x}^*)+4p\eta(\sqrt{2}+2\eta)^{p-1}+2\rho_k(\delta))^{1/p}.$$

Therefore, using $(a+b)^{1/p}\le a^{1/p}+b^{1/p}$ for $a,b\ge0$,

$$\begin{aligned}V_p(\hat{\mathbf{x}})&\le V_p(\mathbf{x}^*)+(4p\eta(\sqrt{2}+2\eta)^{p-1}+2\rho_k(\delta))^{1/p}\\&=\min_{\mathbf{x}\in\Delta(S)}V_p(\mathbf{x})+(4p\eta(\sqrt{2}+2\eta)^{p-1}+2\rho_k(\delta))^{1/p}.\end{aligned}$$

This proves the first expression in the $\min$ term.

For the second expression in the $\min$ term, define

$$G_k(\mathbf{x}):=F_k(\mathbf{x})^{1/p}=\left(\frac{1}{k}\sum_{t=1}^k v_{i_t}(\mathbf{x})^p\right)^{1/p},\quad \widetilde{G}_k(\mathbf{x}):=\widetilde{F}_k(\mathbf{x})^{1/p}=\left(\frac{1}{k}\sum_{t=1}^k \widetilde{v}_t(\mathbf{x})^p\right)^{1/p}.$$

Since $r\mapsto r^{1/p}$ is strictly increasing on $\mathbb{R}_{\ge0}$ and $\hat{\mathbf{x}}\in\arg\min_{\mathbf{x}\in\Delta(S)}\widetilde{F}_k(\mathbf{x})$, we also have $\hat{\mathbf{x}}\in\arg\min_{\mathbf{x}\in\Delta(S)}\widetilde{G}_k(\mathbf{x})$. Since $|\widetilde{v}_t(\mathbf{x})-v_{i_t}(\mathbf{x})|\le2\eta$ for every $t\in[k]$ and every $\mathbf{x}\in\Delta(S)$, the reverse triangle inequality for the ℓ_p-norm gives, uniformly over $\mathbf{x}\in\Delta(S)$,

$$|\widetilde{G}_k(\mathbf{x})-G_k(\mathbf{x})|\le\left(\frac{1}{k}\sum_{t=1}^k|\widetilde{v}_t(\mathbf{x})-v_{i_t}(\mathbf{x})|^p\right)^{1/p}\le2\eta.$$

Moreover, on the same event as above, for every $\mathbf{x}\in\Delta(S)$,

$$|V_p(\mathbf{x})-G_k(\mathbf{x})|=|F(\mathbf{x})^{1/p}-F_k(\mathbf{x})^{1/p}|\le|F(\mathbf{x})-F_k(\mathbf{x})|^{1/p}\le\rho_k(\delta)^{1/p},$$

where we used that $z\mapsto z^{1/p}$ is $1/p$-Hölder on $\mathbb{R}_{\ge0}$.

Therefore, on this event,

$$\begin{aligned}V_p(\hat{\mathbf{x}})\le G_k(\hat{\mathbf{x}})+\rho_k(\delta)^{1/p}&\le\widetilde{G}_k(\hat{\mathbf{x}})+2\eta+\rho_k(\delta)^{1/p}\\&\le\widetilde{G}_k(\mathbf{x}^*)+2\eta+\rho_k(\delta)^{1/p}\\&\le G_k(\mathbf{x}^*)+4\eta+\rho_k(\delta)^{1/p}\\&\le V_p(\mathbf{x}^*)+4\eta+2\rho_k(\delta)^{1/p}=\min_{\mathbf{x}\in\Delta(S)}V_p(\mathbf{x})+4\eta+2\rho_k(\delta)^{1/p}.\end{aligned}$$

This proves the second expression. Taking the minimum of the expressions completes the proof.

C.2 Proof of Corollary 4.3

If $\alpha \geq \sqrt{2}$, output any lottery $\mathbf{x}$ without making queries. Since $v_i \leq \sqrt{2}$ on $\Delta(S)$ for every agent i, we have $V_p(\mathbf{x}) \leq \sqrt{2} \leq \min_{\mathbf{y}\in\Delta(S)} V_p(\mathbf{y}) + \alpha$.

Now assume that $\alpha < \sqrt{2}$, and run Algorithm 1 with $\eta = 0$, using the routine of Lemma 4.1, which makes $\mathcal{O}(m \log(1/\varepsilon))$ queries per agent. By Theorem 4.2, with probability at least $1 - \delta$,

$$V_p(\hat{\mathbf{x}}) \leq \min_{\mathbf{y}\in\Delta(S)} V_p(\mathbf{y}) + (2\rho_k(\delta))^{1/p}.$$

Since $1 + \sqrt{z} \leq \sqrt{2(1+z)}$ for $z \geq 0$, there is a constant $c_p > 0$, depending only on p, such that $\rho_k(\delta) \leq c_p\sqrt{(1 + \log(2/\delta))/k}$. Hence choosing $k = \lceil 4c_p^2(1 + \log(2/\delta))\alpha^{-2p} \rceil$ ensures that $(2\rho_k(\delta))^{1/p} \leq \alpha$. Since $\alpha < \sqrt{2}$ and $\delta < 1$, the quantity $(1 + \log(2/\delta))\alpha^{-2p}$ is bounded below by a positive constant depending only on p, so the rounding up only affects the constant. Therefore the total number of queries is $k \cdot \mathcal{O}(m \log(1/\varepsilon)) \leq C_p\, m \log(1/\varepsilon)\, (1 + \log(2/\delta))\, \alpha^{-2p}$ for a constant $C_p > 0$ depending only on p.

C.3 Proof of Theorem 4.4

Let $m = 3$, and first consider the profile in which every agent has $u_i = (1, 1, 0)$ and $\tau_i = 1$. For every $\mathbf{x} \in \Delta(S)$, we have $U_i(\mathbf{x}) = x_1 + x_2 = 1 - x_3$, so agent i accepts $\mathbf{x}$ if and only if $x_3 = 0$. Let N' be the set of agents that the algorithm queries on this profile, so that $|N'| \leq K$, and let $\hat{\mathbf{x}}$ be its output. Choose $r \in \arg\min_{j\in\{1,2\}} \hat{x}_j$. Since $\hat{x}_1 + \hat{x}_2 \leq 1$, we have $\hat{x}_r \leq \frac{1}{2}$.

Now change only the agents who are not queried: each agent $i \in N \setminus N'$ now has $u_i = \mathbf{e}_r$ and $\tau_i = 1$, while each agent in N' keeps $u_i = (1, 1, 0)$ and $\tau_i = 1$. In the new profile, every agent accepts some lottery and rejects another: agents in N' accept $\mathbf{e}_1$ and $\mathbf{e}_2$ and reject $\mathbf{e}_3$, and agents in $N \setminus N'$ accept $\mathbf{e}_r$ and reject $\mathbf{e}_3$. Every query that the algorithm makes on the original profile is to an agent in N', whose answers do not change. Since the algorithm is deterministic, an induction on the number of queries shows that it makes the same queries, receives the same answers, and returns the same lottery $\hat{\mathbf{x}}$ on the new profile.

In the new profile, every agent accepts $\mathbf{e}_r$: the agents in N' because its third coordinate is 0, and the agents in $N \setminus N'$ because $U_i(\mathbf{e}_r) = 1 = \tau_i$. Hence $\min_{\mathbf{x}\in\Delta(S)} V_p(\mathbf{x}) = 0$. On the other hand, fix any $i \in N \setminus N'$. Then $\bar{u}_i = \frac{1}{3}$, and w_i has entry $\frac{2}{3}$ at coordinate r and $-\frac{1}{3}$ at the other two coordinates, so $\|w_i\| = \sqrt{6}/3$. Since $U_i(\hat{\mathbf{x}}) = \hat{x}_r \leq \frac{1}{2}$, we have

$$v_i(\hat{\mathbf{x}}) = \frac{1 - \hat{x}_r}{\sqrt{6}/3} \geq \frac{1/2}{\sqrt{6}/3} = \sqrt{\frac{3}{8}}.$$

Consequently,

$$V_p(\hat{\mathbf{x}})^p \geq \frac{|N \setminus N'|}{n} \left(\frac{3}{8}\right)^{p/2} \geq \frac{n - K}{n} \left(\frac{3}{8}\right)^{p/2},$$

and taking p-th roots gives $V_p(\hat{\mathbf{x}}) \geq \sqrt{3/8}\,((n - K)/n)^{1/p}$, which proves the bound. In particular, a deterministic algorithm that always outputs a minimizer of V_p cannot query at most $K \leq n - 1$ agents on every profile.

For the second claim, repeat the argument with threshold $\frac{1}{2}$ instead of 1 for all agents in the initial profile, so that each of them accepts $\mathbf{x}$ if and only if $x_3 \leq \frac{1}{2}$, and with threshold $\frac{3}{4}$ instead of 1 for the changed agents. Then every agent strictly accepts some pure alternative and rejects $\mathbf{e}_3$, every agent still accepts $\mathbf{e}_r$, and each changed agent has violation $(\frac{3}{4} - \hat{x}_r)/(\sqrt{6}/3) \geq \frac{3}{4\sqrt{6}} = \frac{\sqrt{6}}{8}$ at $\hat{\mathbf{x}}$. The rest of the argument is unchanged.

C.4 Randomized Algorithms that Always Output a Minimizer

Proposition C.1. *Let $m \geq 3$ and $p \in [1, \infty)$, and consider a randomized, possibly adaptive, algorithm that outputs a minimizer of V_p with probability one on every profile. Then, for every agent i, there is a profile in which every agent strictly accepts some pure alternative and on which the algorithm queries agent i with probability one.*

Proof. Fix $i \in N$. Consider two profiles in which every agent other than i has utility vector $\mathbf{e}_1 + \mathbf{e}_2$ and threshold $\frac{1}{2}$, and therefore accepts exactly the lotteries with $x_1 + x_2 \geq \frac{1}{2}$. In the first profile, agent i has $u_i = \mathbf{e}_1$ and $\tau_i = \frac{3}{4}$; in the second profile, she has $u_i = \mathbf{e}_2$ and $\tau_i = \frac{3}{4}$. In both profiles, every agent strictly accepts some pure alternative and rejects $\mathbf{e}_3$. In the first profile, the lotteries accepted by all agents are those with $x_1 \geq \frac{3}{4}$; these lotteries have $V_p = 0$, so they are exactly the minimizers of V_p. Similarly, in the second profile, the minimizers of V_p are the lotteries with $x_2 \geq \frac{3}{4}$. No lottery belongs to both sets.

Fix a realization of the internal randomness of the algorithm for which the algorithm does not query agent i on the first profile. Then all of its queries on the first profile are to agents other than i, whose answers are the same in both profiles, so the algorithm makes the same queries and returns the same lottery on both profiles. Since no lottery minimizes V_p in both profiles, this lottery is not a minimizer on one of the two profiles. As the algorithm outputs a minimizer with probability one on each profile, such realizations have probability zero. Hence the algorithm queries agent i with probability one on the first profile. □

C.5 Upper Bound for Unqueried Agents

This subsection shows that the dependence on $(n - K)/n$ in Theorem 4.4 cannot be improved. Fix a set $N' \subseteq N$ of K agents whose normalized boundaries are known, and minimize their contribution $\frac{1}{n}\sum_{i \in N'} v_i(\mathbf{x})^p$ to F. This removes boundary-learning error and sampling error, so the only remaining loss comes from the agents in $N \setminus N'$.

Proposition C.2. *Fix $p \in [1, \infty)$, let $N' \subseteq N$ be a set of K agents whose normalized boundaries are known, and let $\hat{\mathbf{x}}$ minimize $\frac{1}{n}\sum_{i \in N'} v_i(\mathbf{x})^p$ over $\Delta(S)$. Then*

$$V_p(\hat{\mathbf{x}}) \leq \min_{\mathbf{x} \in \Delta(S)} V_p(\mathbf{x}) + \sqrt{2}\left(\frac{n-K}{n}\right)^{1/p}.$$

In particular, if every agent in N' strictly accepts some pure alternative, there is a deterministic algorithm that only queries the agents in N', makes $\mathcal{O}(Km\log(1/\varepsilon))$ queries on every ε-quantized instance, and satisfies the same guarantee.

Proof. A minimizer $\hat{\mathbf{x}}$ exists by continuity and compactness; we keep the factor $1/n$ so that the restricted objective is on the same scale as F. Let $\mathbf{x}^*$ minimize V_p. By the choice of $\hat{\mathbf{x}}$,

$$V_p(\hat{\mathbf{x}})^p = \frac{1}{n}\sum_{i \in N'} v_i(\hat{\mathbf{x}})^p + \frac{1}{n}\sum_{i \in N \setminus N'} v_i(\hat{\mathbf{x}})^p \leq V_p(\mathbf{x}^*)^p + \frac{1}{n}\sum_{i \in N \setminus N'} v_i(\hat{\mathbf{x}})^p.$$

Since $v_i \leq \sqrt{2}$ on $\Delta(S)$ for every agent i, the last sum is at most $\frac{n-K}{n}(\sqrt{2})^p$. Taking p-th roots and using $(a+b)^{1/p} \leq a^{1/p} + b^{1/p}$ for $a, b \geq 0$ gives the bound. For the last claim, Lemma 4.1 finds the normalized boundary of each agent in N' with $\mathcal{O}(m\log(1/\varepsilon))$ queries. □

C.6 Proof of Proposition 4.5

Since the modified algorithm stops after L queries and asks each of them r times, it makes at most rL queries. Fix the internal randomness of the original algorithm, if any. As long as all majority answers are correct, the modified algorithm behaves exactly as the original algorithm with exact answers, and in particular, it finishes within L queries. For $t \in [L]$, let E_t be the event that the modified algorithm asks a t-th query, the majority answers to its first $t-1$ queries are correct, and the majority answer to its t-th query is wrong. If none of these events occurs, then all majority answers are correct.

Fix $t \in [L]$. Conditioned on everything that happens before the t-th query, the r answers to this query are flipped independently, each with probability at most ν. Let X be the number of flipped answers, so that $\mathbb{E}[X] \leq \nu r$. Since r is odd, the majority answer is wrong only if $X \geq r/2$, and Hoeffding's inequality gives

$$\begin{aligned}\Pr\left[X \geq \frac{r}{2}\right] &\leq \Pr\left[X - \mathbb{E}[X] \geq \left(\frac{1}{2} - \nu\right) r\right] \\ &\leq \exp\left(-2r\left(\frac{1}{2} - \nu\right)^2\right) = \exp\left(-\frac{r(1-2\nu)^2}{2}\right) \leq \frac{\delta}{L},\end{aligned}$$

where the last inequality follows from the choice of r. Hence $\Pr[E_t] \le \delta/L$, and by the union bound, the probability that some majority answer is wrong is at most $\sum_{t=1}^{L} \Pr[E_t] \le \delta$.

C.7 Weighted Agents

Let $a_i \ge 0$ for each $i \in N$ with $\sum_{i \in N} a_i = 1$, fix $p \in [1, \infty)$, and consider the weighted objective $V_{p,a}(\mathbf{x}) := \left(\sum_{i \in N} a_i v_i(\mathbf{x})^p \right)^{1/p}$, which equals V_p when $a_i = 1/n$ for all i. We explain how our proofs change.

Theorem 3.3. Replace $\bar{F}$ by $\mathbf{x} \mapsto \sum_{i \notin B} a_i v_i(\mathbf{x})^p + \sum_{i \in B} a_i \operatorname{dist}(\mathbf{x}, A_i)^p$. The proof goes through with $\frac{1}{n}$ replaced by a_i, and gives $V_{p,a}(\bar{\mathbf{x}})^p \le V_{p,a}(\mathbf{x}^*)^p + (\sqrt{2})^p \sum_{i \in B} a_i$, where $\mathbf{x}^*$ minimizes $V_{p,a}$. Hence $V_{p,a}(\bar{\mathbf{x}}) \le V_{p,a}(\mathbf{x}^*) + \sqrt{2}\,(\sum_{i \in B} a_i)^{1/p}$.

Theorem 3.6. For $1 < p < \infty$, the function $\mathbf{x} \mapsto \sum_{i \in N} a_i v_i(\mathbf{x})^p$ is convex and differentiable (see the proof of Theorem B.5), with gradient $-p \sum_{i \in N} a_i v_i(\mathbf{x})^{p-1} \hat{u}_i$. A lottery minimizes a convex differentiable function over $\Delta(S)$ if and only if every alternative that it uses minimizes the corresponding coordinate of the gradient. Hence $\mathbf{x}^*$ minimizes $V_{p,a}$ if and only if every alternative that $\mathbf{x}^*$ uses maximizes $\sum_{i \in N} a_i v_i(\mathbf{x}^*)^{p-1} \hat{u}_i(s_j)$.

Theorem 4.2. Suppose that Algorithm 1 samples $i_1, \dots, i_k$ independently from the distribution $(a_i)_{i \in N}$. Then, for every $\mathbf{x}$, the expectation of $F_k(\mathbf{x}) = \frac{1}{k} \sum_{t=1}^{k} v_{i_t}(\mathbf{x})^p$ is $\sum_{i \in N} a_i v_i(\mathbf{x})^p$. The proof of Theorem 4.2 only uses that the sampled agents are independent and identically distributed and that the expectation of F_k equals the objective, so the theorem holds unchanged with V_p replaced by $V_{p,a}$.

D Experimental Details

This appendix describes the populations, the learner, and the solver used in Section 5, together with additional experiments. Our code, together with the saved results and instructions for reproducing the experiments with a single command, is available at `https://github.com/nicteh/Learning-Acceptable-Lotteries-Minimizing-Violation-Distances`. Throughout, the algorithm only observes membership query answers of the sampled agents; the true boundaries of all agents are used only afterwards, to evaluate the computed lotteries.

Populations. Every agent is generated in the same way. We draw a preference direction by adding independent Gaussian noise to each coordinate of the center of the agent's cluster, and rescale it affinely to $[0, 1]$, so that the agent's best alternative has utility 1 and her worst alternative has utility 0. Her threshold is $\tau_i = \alpha_i$, where α_i is drawn from a Gaussian distribution whose mean is the threshold offset of her cluster and whose standard deviation is 0.035, and then clipped to $[0.2, 0.9]$. Thus the threshold lies a fraction α_i of the way from the lowest to the highest utility of the agent, and larger offsets mean stricter agents. To make the instance ε-quantized with $\varepsilon = 1/32$, we round each utility to the nearest multiple of $1/32$, keeping the best and worst utilities at 1 and 0, and move the threshold to the nearest multiple of $1/32$ strictly between 0 and 1 that is different from all utilities of the agent. Hence every agent strictly accepts her best pure alternative and rejects her worst one, so no agent is distance-underdetermined. Table 1 lists the parameters of the two populations.

Learner. The learner follows case 1 in the proof of Theorem 3.1 (Appendix B.1), which is also the procedure of Lemma 4.1. It queries the m pure lotteries and fixes the rejected alternative b with the smallest index. It then searches for the boundary point on the edge from $\mathbf{e}_b$ to $\mathbf{e}_j$ for each accepted alternative j, lets a be an accepted alternative whose boundary point is closest to $\mathbf{e}_b$, and searches for the boundary point on the edge from $\mathbf{e}_j$ to $\mathbf{e}_a$ for each rejected alternative $j \ne b$. Finally, it computes $(\hat{u}_i, \hat{\tau}_i)$ from these $m - 1$ boundary points as at the end of the proof of Theorem 3.1. Each search is a binary search for the parameter λ^* of the boundary point. Since λ^* is a fraction with denominator at most 32, and any two such fractions differ by at least $1/32^2 = 2^{-10}$, after 11 steps the search interval contains exactly one such fraction, which is λ^*. Hence $10 + 9 \cdot 11 = 109$ queries recover the normalized boundary exactly. If an agent receives $d < 109$ queries, the learner splits the $d - 10$ queries left after the pure lotteries as evenly as possible among the 9 searches, and each search with fewer than 11 steps uses the midpoint of its final interval in place of λ^*. In particular, with $d = 10$,

	Shared-block population	Conflicting-blocks population
(n, m)	$(1000, 10)$	$(1000, 10)$
Clusters	3 clusters with probabilities $0.36/0.34/0.30$.	5 clusters with probabilities $0.30/0.24/0.21/0.17/0.08$.
Cluster centers	The common pattern $(1.2, 1.0, 0.8, 0, \ldots, 0)$ plus independent Gaussian noise with standard deviation 0.22 in each coordinate, drawn once for each cluster of each population.	Cluster $b \in \{1, \ldots, 5\}$ has value 1.25 on alternatives $2b - 1$ and $2b$, and value -0.42 on the other alternatives.
Agent noise	Standard deviation 0.28.	Standard deviation 0.18.
Threshold offsets	$0.70/0.75/0.72$.	$0.32/0.38/0.35/0.42/0.70$; the smallest cluster is the strictest.
Query budget	$Q = 3200$; $k \in \{5, 10, 20, 40, 80, 160, 320\}$; $\lfloor Q/k \rfloor$ queries per sampled agent.	Same.
Populations	100 populations, each with its own samples of agents.	Same.

Table 1: Setup of the finite-query experiments in Figures 2 and 3.

the learner places every boundary point at the middle of its edge, and with $d = 20$, one search takes two steps and the others take one step. If an agent receives more than 109 queries, the learner only uses 109 of them. Every query is answered by comparing the agent's utility for the queried lottery with her threshold in exact rational arithmetic, and is counted; the code checks that every run respects its budget, and that exact recovery returns the true normalized boundary up to floating-point error.

Objective and decomposition. Algorithm 1 samples k agents uniformly at random with replacement. We compare three lotteries: the lottery $\mathbf{x}_{\mathrm{full}}$ that minimizes V_2 for the full population, the lottery $\mathbf{x}_{\mathrm{sample}}$ that minimizes the objective of the sampled agents with their true boundaries, and the output $\hat{\mathbf{x}}$ of the algorithm, which minimizes the objective of the sampled agents with their learned boundaries. All three are evaluated with V_2 on the full population, which gives $G_{\mathrm{tot}} = V_2(\hat{\mathbf{x}}) - V_2(\mathbf{x}_{\mathrm{full}})$, $G_{\mathrm{cov}} = V_2(\mathbf{x}_{\mathrm{sample}}) - V_2(\mathbf{x}_{\mathrm{full}})$, and $G_{\mathrm{bdry}} = V_2(\hat{\mathbf{x}}) - V_2(\mathbf{x}_{\mathrm{sample}})$, so that $G_{\mathrm{tot}} = G_{\mathrm{cov}} + G_{\mathrm{bdry}}$. The coverage gap is nonnegative, but the boundary-learning gap can be negative: a lottery computed from learned boundaries can happen to be better for the full population than one computed from the true boundaries of the same sampled agents. We minimize $F_2 = V_2^2$, which has the same minimizers as V_2, and report all quantities in the scale of V_2. Algorithm 1 may return any minimizer, and the minimizer is not always unique; for example, when some lottery is acceptable to all sampled agents, every such lottery is a minimizer, which often happens for small k. Whenever we minimize V_p, both here and in the p-sweep below, we return the minimizer closest to the uniform lottery, that is, the minimizer with the smallest Euclidean norm. This choice affects the results for small k: another minimizer can give a smaller or a much larger coverage gap.

Solver. We minimize V_p over the simplex with CVXPY [Diamond and Boyd, 2016] and the interior-point solver Clarabel [Goulart and Chen, 2024]: this is a quadratic program for $p = 2$, a linear program for $p \in \{1, \infty\}$, and a conic program for other values of p. To find the minimizer with the smallest norm, we first compute some minimizer $\mathbf{x}'$, and then minimize $\|\mathbf{x}\|^2$ over the lotteries $\mathbf{x}$ with $v_i(\mathbf{x}) \le v_i(\mathbf{x}')$ for all agents i. For $1 < p < \infty$, these are exactly the minimizers, since every minimizer gives each agent the same violation by the strict convexity of $t \mapsto t^p$ on $t > 0$; for $p = 1$ and $p = \infty$, we instead use the constraints $\sum_i v_i(\mathbf{x}) \le \sum_i v_i(\mathbf{x}')$ and $\max_i v_i(\mathbf{x}) \le \max_i v_i(\mathbf{x}')$, respectively. We set the duality gap and feasibility tolerances of Clarabel to 10^{-12}, with at most 1000 iterations. For some conic programs, Clarabel cannot reach this accuracy; we then retry with tolerances 10^{-10}, 10^{-9}, and 10^{-8}, and only accept solutions that Clarabel reports as optimal.

Results. Table 2 lists the means shown in Figure 2. For $k \le 20$, every sampled agent is learned exactly, and the boundary-learning gap is below 10^{-12} in every run. For $k \ge 40$, the budget of $Q = 3200$ queries is used in full. To compare values of k, we pair the runs by population. For the conflicting-blocks population, the total excess at $k = 80$ is smaller than at every other k, with 95% bootstrap confidence intervals for the mean difference that exclude zero. For the shared-block population, the total excess at $k = 80$ is significantly smaller than for $k \le 40$, but the confidence intervals for the differences from $k = 160$ and $k = 320$ contain zero.

Thresholds closer to the middle. The shape of the curves in Figure 2 depends on the population and on the learner. To illustrate this, we repeated the fixed-budget experiment for the conflicting-blocks population with threshold offsets $0.47/0.53/0.50/0.57/0.78$, so that the thresholds lie closer to the

		Shared-block		Conflicting-blocks		Middle thresholds	
k	$\lfloor Q/k \rfloor$	G_{tot}	G_{bdry}	G_{tot}	G_{bdry}	G_{tot}	G_{bdry}
5	640	35.06	0.00	37.96	0.00	35.42	0.00
10	320	20.24	0.00	32.58	0.00	24.01	0.00
20	160	12.92	0.00	17.98	0.00	17.49	0.00
40	80	6.82	0.01	12.25	−0.05	11.78	0.02
80	40	3.30	0.17	6.02	−0.06	6.85	−0.54
160	20	3.90	2.13	10.28	7.05	12.00	7.98
320	10	4.24	3.25	37.29	35.68	6.31	4.05

Table 2: Mean total excess and boundary-learning gap over 100 populations under a fixed budget of $Q = 3200$ queries, in units of 10^{-3}. The last two columns are for the conflicting-blocks population with thresholds closer to the middle of each agent's utility range.

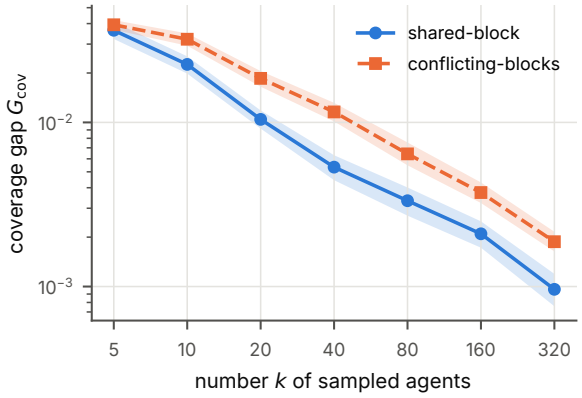


Figure 3: **Coverage gap with exact recovery.** Every sampled agent receives 109 queries and is learned exactly, so the total excess equals the coverage gap G_{cov}. Lines show means over 100 populations, and shaded bands are 95% bootstrap confidence intervals for the means; both axes are logarithmic.

middle of each agent's utility range (last two columns of Table 2). The total excess at $k = 320$ is then 6.31×10^{-3}, comparable to 6.85×10^{-3} at $k = 80$ (the paired difference is not significant), instead of about six times larger. To see why, recall that with 10 queries, the learner places every boundary point at the middle of its edge. Measured by the error $\max\{\|\tilde{u} - \hat{u}_i\|, |\tilde{\tau} - \hat{\tau}_i|\}$, the boundaries learned from 10 queries are about as accurate in both versions of the population (mean error 0.26 with the offsets of Table 1 and 0.24 with the offsets above), and in both cases more accurate than those learned from 20 queries (mean error 0.33 and 0.50), since a single step of binary search can move a boundary point further from the truth than the middle of its edge. What differs is the direction of the errors. With the offsets of Table 1, most thresholds in the conflicting-blocks population lie well below the middle of the utility range, so the boundaries learned from 10 queries make the sampled agents look stricter: at the full-information optimum, their learned violations exceed their true violations by 0.135 on average, compared with 0.014 with the offsets above. Errors in the same direction shift the computed lottery, while errors in different directions partly cancel. Since the guarantee of Theorem 4.2 only depends on the size of the errors, it does not capture this difference.

Exact recovery for every sampled agent. To show the coverage gap on its own, we also ran the algorithm with 109 queries for every sampled agent, for the same values of k and the same populations. Every sampled agent is then learned exactly, and the boundary-learning gap is below 10^{-12} in every run. Figure 3 shows that the coverage gap decreases steadily with k: least-squares fits to the mean curves on a log-log scale have slopes -0.87 for the shared-block population and -0.75 for the conflicting-blocks population.

Power-mean sweep. The p-sweep experiment examines the role of the parameter p. Unlike the finite-query experiments, it uses one structured instance, with $n = 300$ agents and $m = 6$ alternatives,

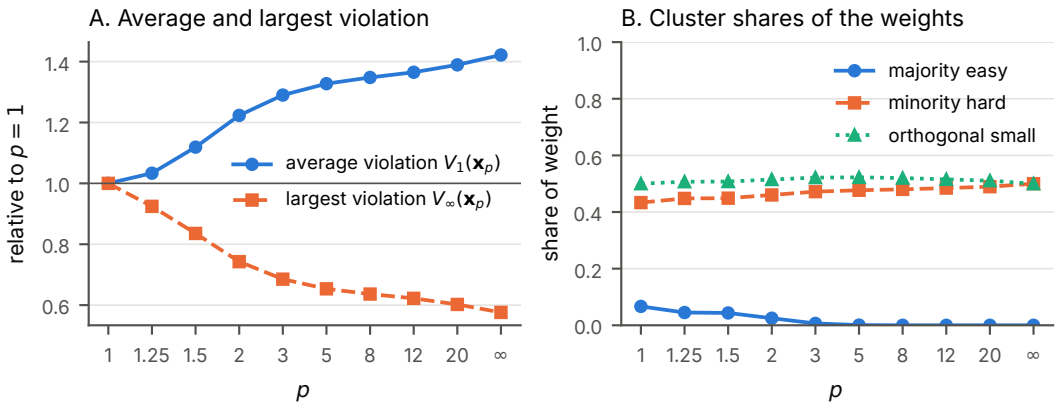


Figure 4: **Changing p changes which agents drive the optimum.** Panel A plots the average and the largest violation of the minimizers $\mathbf{x}_p$, relative to their values for $p = 1$: larger p lowers the largest violation $V_\infty(\mathbf{x}_p)$ but raises the average violation $V_1(\mathbf{x}_p)$. Panel B shows the share of each cluster in the weights $v_i(\mathbf{x}_p)^{p-1}$ of Theorem 3.6 for $1 < p < \infty$. The endpoint shares at $p = 1$ and $p = \infty$ use the descriptive conventions below and need not be dual-optimal weights.

and makes no queries. The agents form three clusters of sizes 180, 75, and 45, which we call majority easy, minority hard, and orthogonal small. The agents are generated as in the finite-query experiments, without quantization, from the cluster centers $(2.0, 0.85, 0.10, -0.95, -0.95, -1.05)$, $(-0.95, 2.45, -0.65, -0.35, -0.25, -0.25)$, and $(-0.50, -0.60, 1.80, 0.75, -0.70, -0.75)$, with agent noise of standard deviation 0.08, 0.07, and 0.08, threshold offsets 0.52, 0.84, and 0.50, threshold noise of standard deviation 0.025, and thresholds clipped to $[0.35, 0.88]$. For each $p \in \{1, 1.25, 1.5, 2, 3, 5, 8, 12, 20, \infty\}$, we compute a minimizer $\mathbf{x}_p$ of V_p and report the average violation $V_1(\mathbf{x}_p)$ and the largest violation $V_\infty(\mathbf{x}_p)$.

Small p emphasizes the average violation, and large p emphasizes the largest violations. In this instance, $V_1(\mathbf{x}_p)$ increases and $V_\infty(\mathbf{x}_p)$ decreases as p grows. Moving from $p = 1$ to $p = \infty$ lowers the largest violation by 42%, from 0.391 to 0.225, while raising the average violation by 42%, from 0.047 to 0.066.

For $1 < p < \infty$, Theorem 3.6 gives agent i the weight $v_i(\mathbf{x}_p)^{p-1}$. We normalize these weights to sum to 1, and define the share of a cluster as the sum of the normalized weights of its agents. As p increases, the share of the majority easy cluster, whose agents have the smallest violations, drops from 0.07 to 0, and its weight moves to the two clusters with larger violations. For $p = 1$, we give equal weight to all agents with violation above 10^{-8}, and for $p = \infty$, we give equal weight to all agents whose violation is at least $\max_j v_j(\mathbf{x}_\infty) - \max\{10^{-8}, 10^{-5} \max_j v_j(\mathbf{x}_\infty)\}$. For $1 < p < \infty$, we also checked the condition of Theorem 3.6 by comparing the weighted scores of the alternatives used by $\mathbf{x}_p$ (those with probability above 10^{-6}) with the largest weighted score; the largest relative difference is 8.5×10^{-7}. Figure 5 shows the violations within each cluster: larger p protects the agents with the largest violations, in the orthogonal small cluster, but shifts some of the violation to the minority hard cluster.

Seeds and confidence intervals. Population $s \in \{0, 1, \ldots, 99\}$ of the shared-block and conflicting-blocks families is generated with the NumPy random generator seeded by $(1, s)$ and $(2, s)$, respectively. The sample of k agents from population s of family $f \in \{1, 2\}$ is drawn with seed (f, s, k, e), where $e = 1$ for the fixed-budget experiment, $e = 2$ for the experiment with exact recovery, and $e = 3$ for the experiment with thresholds closer to the middle. The instance of the p-sweep uses seed 3301. Confidence intervals are 95% percentile bootstrap intervals for the mean over the 100 populations, with 2000 resamples drawn with seed 2026.

Compute resources. All experiments were run on a Linux machine with 2 CPU cores (Intel Xeon, 2.8 GHz) and 8 GB of memory, without a GPU. Running all experiments, including the tests of

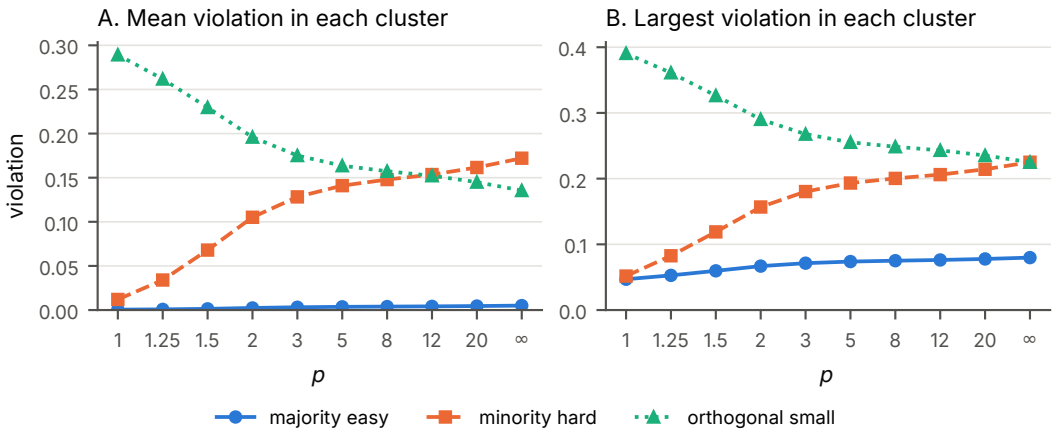


Figure 5: **Cluster-level effects of changing p.** Panel A shows the mean violation within each cluster, and Panel B shows the largest violation within each cluster.

the code, takes less than five minutes. Pilot runs, including those used to choose the population parameters, used a comparable amount of compute.

Limitations. The populations are synthetic, and their parameters were chosen so that the effects studied in the theory are visible; as the experiment with thresholds closer to the middle shows, the shape of the curves depends on these choices. With few queries per agent, the learner is the procedure of Lemma 4.1 stopped early, and other learners may behave differently. The simulations do not model strategic or inconsistent answers. Finally, the full population is used to evaluate the computed lotteries, which lets us compare them with the full-information optimum but is not possible in deployment. The experiments therefore make a narrow claim: on transparent instances, the effects of a limited budget and of the parameter p are those identified by the theory.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the model, the assumptions (in particular, exact answers and acceptable sets that are halfspaces), and the contributions, with pointers to the corresponding theorems. These claims match our results.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 1 states that exact answers are an idealized baseline; Section 4.3 discusses which kinds of noisy answers our results cover and which they do not; Sections 4.2 and 6 state the open questions left by our work; Section 5 and Appendix D state that the experiments are synthetic controlled checks rather than evidence of deployment performance; and Appendices A.1 and A.4 discuss the halfspace assumption and other sets of feasible outcomes.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper states the main modeling assumptions in Section 2 and includes additional assumptions in each theorem statement. The main text explains the idea behind each result, and complete proofs are provided in Appendices B and C.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 5 and Appendix D specify the synthetic population generators, membership query learner, query budgets, solver tolerances, and seeds. The supplementary code archive includes the dependencies and instructions for running all experiments and regenerating the experimental figures, tables, and numerical summaries with a single command.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide the experimental code and saved results under the MIT license as supplementary material on OpenReview. The archive includes a README with installation and reproduction instructions, pinned package versions, and scripts for all experimental figures, tables, and numerical summaries. All data are synthetic and generated by the code.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5 and Appendix D give all settings, including n, m, the cluster, noise, and threshold parameters, the query budgets, the learner, the grid of values of p, and the solver and its tolerances. Appendix D also reports how the results change with other threshold parameters. There are no train/test splits because the experiments are synthetic optimization simulations.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Figures 2 and 3 show 95% confidence intervals for the mean over 100 randomly generated populations, each with its own random sample of agents. Appendix D states that these are percentile bootstrap intervals with 2000 resamples. The p-sweep uses one fixed instance and involves no randomness after the instance is generated, so it has no error bars.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix D states that all experiments run in less than five minutes on a Linux machine with 2 CPU cores and 8 GB of memory, without a GPU, and that pilot runs used a comparable amount of compute.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work is theoretical with synthetic experiments, does not involve human subjects, private data, or high-risk released models, and is intended for responsible use in stakeholder-aware randomized decision settings. To the best of the authors' knowledge, the research conforms to the NeurIPS Code of Ethics.

   Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Section 1 discusses the intended positive impact: when deploying AI systems, choosing a randomization over candidate systems that violates the requirements of several stakeholders as little as possible, using only accept/reject feedback. Section 6 discusses a possible negative impact: minimizing aggregate violation trades the requirements of different stakeholders against one another, so safety requirements that must hold without exception should be imposed as constraints rather than included in the objective. The parameter p also determines how much weight the most violated stakeholders receive (Section 5 and Appendix D).

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The experiments use the open-source software CVXPY and Clarabel (both under the Apache 2.0 license), which are cited in Appendix D, as well as NumPy, SciPy, pandas (BSD licenses), and Matplotlib (Matplotlib license). The supplementary archive documents the software versions in its README and `requirements.txt`. No existing datasets or models are used.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The supplementary code archive on OpenReview includes our experimental code, saved results, and figures, together with an MIT `LICENSE` file. Its README and accompanying documentation describe the methods, experiment settings, seeds, software dependencies, and commands for reproducing the experimental results.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.