Strategic Decision Focused Learning

Tinashe Handina[*]
California Institute of Technology

Yuehan Diao
University of Chicago

Adam Wierman
California Institute of Technology

Eric Mazumdar
California Institute of Technology

Abstract

Machine learning (ML) predictions are increasingly being used to guide decision-making, giving rise to the problem of decision-focused learning (DFL) where predictors are optimized for downstream decision quality rather than accuracy alone. However, most existing work assumes a single decision-maker optimizing in isolation. This paper formalizes *strategic* decision-focused learning, where an ML system predicts an exogenous state that some agents observe before playing a game. For example, a park ranger may predict wildlife locations to allocate anti-poaching patrols against strategic poachers. While the exogenous state is unaffected by agent actions, predictions influence agents' strategies and the resulting equilibrium. We find that strategic considerations fundamentally change the learning problem. In particular, we show the prediction accuracy–equilibrium payoff landscape can be non-monotonic—i.e., better predictions can degrade performance. We propose algorithmic approaches to address these challenges and validate them across benchmarks in wildlife conservation and infrastructure protection. Our theory and experiments highlight the importance of accounting for strategic interactions when designing predictors.

1 Introduction

Machine Learning (ML) systems are playing an increasingly vital role in shaping decisions across a wide range of domains. From commerce [4], to security [13] [28], to government [20] [34], decision makers are increasingly relying on the outputs of ML systems as the basis upon which they optimize their actions. Driving the proliferation of ML is its demonstrable ability to understand and predict the evolution of complex systems reasonably well. As ML systems become an ever more consequential tool used for decision making, understanding when standard training objectives are *structurally inadequate* — not merely suboptimal — is of vital importance. This paper identifies one such setting.

Learning with the view that the output of a learning algorithm is going to be used in a downstream optimization problem is an active area of research often called decision-focused learning[15][31][27]. Research in this area, to date, has often focused on a single decision maker leveraging ML outputs to optimize a utility or cost function in isolation [27]. In many consequential applications, however, decision makers find that they are interacting with other actors who often act in a *strategic* manner. This introduces complexities to the optimization problem as the agent now not only has to consider the impact of the decision on the cost function, but also has to take into account how other actors may respond to any particular decision they make.

In this work, we formalize this problem area which we call **strategic decision-focused learning** (SDFL). In this setting, an agent is trying to leverage ML systems to aid with the prediction of some

[*]Corresponding author: thandina@caltech.edu.

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

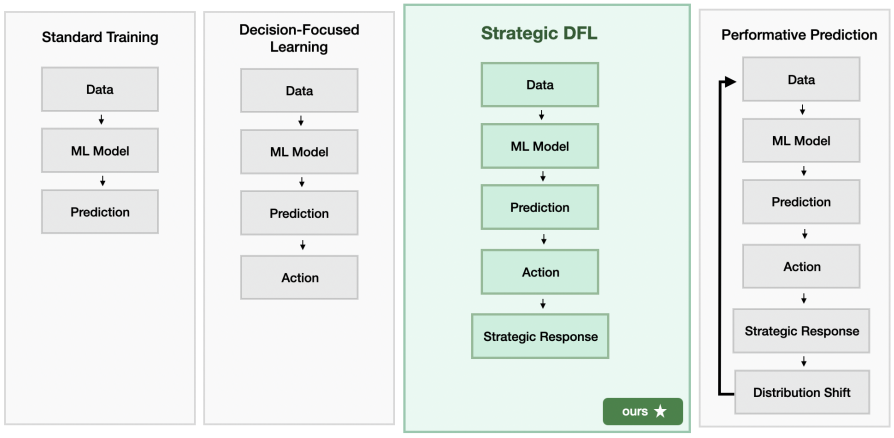


Figure 1: Comparison of different learning paradigms. In contrast to classical and decision-focused learning, SDFL operates in a strategic context, significantly complicating the learning task. Unlike performative prediction or strategic classification, the data distribution is exogenous—free from strategic feedback loops.

exogenous state that affects a game that they play. Such settings are prevalent across a wide range of domains, including:

- **Electricity markets:** ML predictors can be used by producers competing in grid markets to forecast exogenous renewable generation conditions (e.g., wind and solar availability), which are then used to inform pricing strategies that interact with those of competing producers.
- **Motorsport strategy:** In Formula One, teams leverage predictive models to forecast exogenous variables such as tire degradation rates and safety car probabilities, which shape pit-stop and fuel strategies in direct competition with opposing teams.
- **Security and conservation:** Predicting population distributions over a geographic region — such as wildlife density or civilian movement — informs patrol strategies in adversarial security games (we evaluate this concretely in conservation and infrastructure protection domains in this work).

We find that the explicit characterization of the downstream strategic interaction leads to fundamental — not merely practical — considerations for the design and use of machine learning systems. To that end, this work seeks to address the following question:

How should agents develop and make use of Machine Learning for exogenous state predictions in strategic interactions?

Traditional approaches to this problem in the strategic domain have often relied on first predicting exogenous states such as demand [29] or population distributions [26], then optimizing for a decision. We show that this approach is not merely suboptimal but also suffers from a fundamental issue of misalignment. As we show, in strategic settings no monotone transformation of a supervised accuracy loss can serve as a reliable proxy for strategic utility. This failure is structural, arising from the interaction between prediction error and equilibrium response, and persists regardless of model capacity or data quantity.

Our contributions: This paper makes three conceptual contributions and one algorithmic contribution.

(C1) Problem formalization. We formalize strategic decision-focused learning as a distinct problem within the decision-focused learning paradigm. The SDFL problem is defined for general games and solution concepts: an agent predicts an exogenous state and commits to an equilibrium strategy based on that prediction, while opponents respond under the *true* state. This split between the predicted world the agent plans for and the true world opponents respond to is the source of all the difficulty we characterize. The Stackelberg equilibrium is one tractable instantiation we study in depth; the formalization and hardness results hold for general solution concepts.

(C2) Structural hardness. We prove that the mapping from prediction accuracy to equilibrium payoff is *provably non-monotone*: a less accurate prediction can strictly outperform a more accurate one (Proposition 3.2), while the true state in, some instances, may be suboptimal (Proposition 3.3). Together, these results characterize the structure of the accuracy–utility landscape. This is not a worst-case pathology: it is a feature of the landscape, implying that standard predict-then-optimize pipelines are not well motivated for SDFL.

(C3) Geometric characterization of hardness. We prove that, for a class of games, as game dimensionality grows, an increasing proportion of strategies lie near best-response boundaries (Theorem 3.7), explaining geometrically *why* small prediction errors may induce large utility swings. This result characterizes why the hardness can intensify as the action space grows.

(C4) Initial algorithmic framework. Motivated by C1–C3, we propose end-to-end prediction approaches as an initial algorithmic response to SDFL. We present these as proof-of-concept rather than definitive solutions, in the same spirit as early task-loss surrogates in single-agent DFL that preceded formal analysis [3, 31]. Experiments across two real-world security game domains validate that end-to-end approaches outperform predict-then-optimize despite higher supervised prediction error, corroborating our theoretical findings that accuracy and strategic utility are misaligned.

1.1 Related work

Learning how to make use of machine learning for the purposes of downstream decision making has emerged as an important area of research [15, 5]. The key idea powering much of this work is that embedding the optimization problem of interest into the training loop and backpropagating decision-relevant gradients can lead to better performance than first optimizing a prediction and then optimizing for a decision [3, 33]. While standard approaches to decision-focused learning often focus on a single agent optimizing in isolation [27], we focus on extending and formalizing this paradigm for the strategic domain. In particular, we consider what happens when the decision is observed by strategic agents whose choices are affected by the optimization decision itself. We call this setting *strategic decision-focused learning*. Crucially, the strategic setting is not merely a harder instance of standard DFL but a qualitatively different problem: the accuracy–utility misalignment we identify has no analogue in the single-agent case. Closest to our work is game-focused learning in security games [24, 30]: there the payoffs are known, and a defender learns how an attacker reacts and then best-responds to what is learned. In our framework, the state that sets the payoffs, which our opponents already observe, is what is predicted. The predictor then plays an equilibrium of the game it thinks it is in. This introduces new forms of complexity: for example, predicting the true state in some instances may be suboptimal (Proposition 3.3).

Learning in strategic environments: Recent years have seen a surge of research on deploying learning algorithms within strategic environments [17, 16]. This focus on the intersection of learning and strategic interactions has given rise to, among others, the domains of strategic classification [11], adversarial machine learning [14], and multi-agent reinforcement learning [35]. Closely related is performative prediction [23], wherein the act of prediction itself causes a distributional shift in the data the model is trained on. Our work takes an orthogonal track: rather than the prediction changing the input distribution, we ask how to properly account for exogenous state information within a strategic decision-making task where the prediction shapes the equilibrium that emerges. The exogeneity of the data distribution is what separates SDFL from performative prediction; the presence of strategic opponents is what separates it from standard DFL. A key theme arising from this body of work is that learning in strategic environments is often non-trivial, with phenomena such as instability of algorithms resulting in convergence to cycles and chaos [18], and strategic manipulations leading to improved causal discovery [1]. Of particular interest is the insight that small changes in learning parameters may lead to different equilibria with different utilities [36]. Previous work has also found that model selection within the broader context of learning in strategic environments is complicated, with the model expressivity–payoff landscape in general being non-monotonic [10]. Our work contributes a precise characterization of this non-monotonicity in the SDFL setting, grounding it in the geometry of best-response regions rather than model expressivity.

2 Problem Formulation

We formalize strategic decision-focused learning as a distinct problem within the decision-focused learning paradigm. We begin with a general formulation applicable to any game and solution concept, then instantiate it for the Stackelberg setting we study in depth.

In the standard decision-focused learning (DFL) setting, a single agent uses a learned model $f_\theta : \mathcal{X} \to \mathcal{S}$ to predict an uncertain parameter $s \in \mathcal{S}$ and solves a downstream optimization problem whose objective depends on s. The central insight of DFL is that training f_θ to minimize a supervised loss can be suboptimal when downstream decision quality is what matters [5, 31]. Critically, most existing DFL assumes the agent optimizes *in isolation* — no other agent observes or responds to the decision. As we show, this isolation assumption is not merely a simplification but a load-bearing one: its violation changes the learning problem qualitatively.

2.1 Strategic Decision-Focused Learning

We consider a game $G = (\mathcal{N}, \{\mathcal{A}_i\}_{i\in\mathcal{N}}, \{U_i\}_{i\in\mathcal{N}}, \mathcal{S})$ where $\mathcal{N}$ is a finite set of agents, $\mathcal{A}_i$ is the action set of agent i, $\mathcal{S} \subseteq \mathbb{R}^d$ is an exogenous state space, and $U_i : \prod_j \mathcal{A}_j \times \mathcal{S} \to \mathbb{R}$ is agent i's utility. Each agent selects a mixed strategy $\alpha_i \in \Delta(\mathcal{A}_i)$.

The state $s \in \mathcal{S}$ is *exogenous*: drawn from a fixed distribution $P_\mathcal{S}$ independently of any agent's actions. A primary agent i observes only a dataset $\mathcal{D} = \{(x^{(t)}, s^{(t)})\}_{t=1}^T$ and learns a predictor $f_\theta : \mathcal{X} \to \mathcal{S}$. All other agents observe the true realized state s_r.

Let $\sigma : \mathcal{S} \to \prod_j \Delta(\mathcal{A}_j)$ denote a *solution mapping* returning an equilibrium profile for a given state. Agent i commits to $\sigma_i(\hat{s})$ based on their prediction $\hat{s} = f_\theta(x)$, while opponents respond under the true state s_r. The realized joint profile is: $\sigma(\hat{s}, s_r) = \big(\sigma_i(\hat{s}),\ \mathrm{Eq}_{-i}(\sigma_i(\hat{s}),\ s_r)\big)$ where $\mathrm{Eq}_{-i}(\alpha_i, s)$ is the equilibrium response of agents $-i$ to a fixed α_i under state s. The *strategic decision-focused learning problem* is then:

$$\theta^* = \arg\max_\theta\ \mathbb{E}_{(x,s_r)\sim\mathcal{D}}\left[U_i(\sigma(f_\theta(x),\ s_r),\ s_r)\right] \tag{1}$$

The split between the predicted world agent i plans for and the true world opponents respond to is the source of all the difficulty we characterize in Section 3. We note that this formulation is distinct from performative prediction [23], wherein predictions shift $P_\mathcal{S}$ through a feedback loop — here the data distribution is fixed and predictions affect only the equilibrium. Our framework is clearly distinguished from existing learning problems in Figure 1.

2.2 Stackelberg Instantiation

The SDFL objective equation 1 is defined for general games and solution concepts; the hardness results of Section 3 hold in this generality. For the algorithmic contributions and experiments of Sections 4, we instantiate SDFL under the *Stackelberg solution concept*, chosen for tractability and its natural fit to security applications.

We focus on two-player interactions in which agent i leads and agent $-i$ follows. Given predicted state $\hat{s}$, the leader computes: $\bar{\alpha}_i(\hat{s}) = \arg\max_{\alpha_i\in\Delta(\mathcal{A}_i)} U_i(\alpha_i,\ BR_{-i}(\alpha_i, \hat{s}),\ \hat{s})$ where $BR_{-i}(\alpha_i, s) = \arg\max_{\alpha_{-i}} U_{-i}(\alpha_i, \alpha_{-i}, s)$. Since the follower responds under the true state s_r, the leader's realized utility is: $U_i^{\mathrm{realized}}(\hat{s}, s_r) = U_i(\bar{\alpha}_i(\hat{s}),\ BR_{-i}(\bar{\alpha}_i(\hat{s}),\ s_r),\ s_r)$. Stackelberg equilibria are the natural solution concept for security settings, where attackers are assumed to observe both the state and the defender's committed strategy [32].

Illustrative example. A game ranger predicts elephant locations to allocate patrols against strategic poachers. The exogenous state — the true elephant distribution — is unknown to the ranger but (in the worst case) observed by the poachers. The ranger commits to a patrol strategy based on their prediction; the poachers best-respond knowing the truth. The ranger's realized utility depends not on how accurately they predicted elephant locations, but on how well their patrol strategy performs against opponents who know the true state. This gap between prediction quality and strategic performance is precisely what equation 1 captures.

3 Limitations of Accuracy based Predictions in Games

We establish that standard accuracy-based approaches to SDFL suffer from a problem of misalignment. We prove two independent results that together characterize why: first, the mapping from prediction accuracy to equilibrium payoff is provably non-monotone, implying no monotone transformation of a supervised loss can serve as a reliable proxy for strategic utility; second, for a broad class of games, the fraction of strategies near best-response boundaries grows with game dimensionality, explaining geometrically why small prediction errors induce large utility swings. While we instantiate these results in the Stackelberg setting for concreteness, the non-monotonicity result holds for the general SDFL objective equation 1 under other solution mappings, σ, satisfying the conditions of Section 2.

Given a dataset $\mathcal{D}$ relating to the exogenous state s, standard approaches to learning in strategic environments first find some s_{pred} by minimizing a statistical loss function $\mathcal{L}(\phi, \mathcal{D})$ which is proportional to some accuracy metric ϕ for the game instance. After having learned a prediction of the state, the agent then looks to optimize their decision-making given their state prediction, which was yielded from the previous estimation step.

Definition 3.1 (Accuracy Metric)**.** Let $\mathcal{S} \subseteq \mathbb{R}^n$ for some $n \in \mathbb{N}$, and let $s_r \in \mathcal{S}$ be the true realization of the state. An accuracy metric for the game instance is a function $\phi : \mathcal{S} \to \mathbb{R}_{\geq 0}$ defined by $\phi(s) = \|s - s_r\|$ for some symmetric norm $\|\cdot\|$ on $\mathbb{R}^n$.

To extend the previous example of elephant conservation, we assume that the ranger is given some dataset $\mathcal{D}$ on the location of elephants within the reserve. Standard approaches to the strategic decision-focused learning problem would have the ranger find s_{pred} (a prediction of where the elephants would be), and then from there, they would optimize to find a patrol strategy that maximizes their utility. We call this approach to learning **"Predict Then Optimize"** (PTO).

We begin by showing the limitations of this particular learning paradigm. We find that it is possible for a state prediction to be less accurate yet lead the decision maker into selecting an action that yields better payoff when compared to some other, more accurate state prediction. This shows that utility within the context of strategic environments can be non-monotonic with respect to prediction accuracy.

Proposition 3.2. *There exists a game instance with true state s_r, and predictions $s_{p,1}$ and $s_{p,2}$, wherein $s_{p,1}$ is strictly more accurate than $s_{p,2}$ under every accuracy metric ϕ, yet player i's utility at the equilibrium induced by $s_{p,1}$ is strictly lower than that induced by $s_{p,2}$:*

$$U_i(\alpha^*(s_{p,1}),\, s_r) \;<\; U_i(\alpha^*(s_{p,2}),\, s_r)\,,$$

where $\alpha^*(s) = (\alpha_i^*(s),\, BR_{-i}(\alpha_i^*(s), s_r))$ denotes the equilibrium joint strategy profile when the primary plays assuming state s, the rest of the agents see the true state s_r, and utility is evaluated at s_r.

In addition, we find that there exist game instances in which the true state s_r is not the utility-optimal prediction for a player: a strategically calibrated misprediction can induce an equilibrium that, when played against the true state, yields strictly higher utility than the equilibrium induced by s_r itself.

Proposition 3.3. *There exists a game instance with true state s_r and player i such that there exists a prediction $s_p \in \mathcal{S}$ satisfying:*

$$U_i(\alpha^*(s_p),\, s_r) \;>\; U_i(\alpha^*(s_r),\, s_r)\,,$$

where $\alpha^*(s) = (\alpha_i^*(s),\, BR_{-i}(\alpha_i^*(s), s_r))$ denotes the equilibrium joint strategy profile when the primary plays assuming state s, the rest of the agents see the true state s_r, and utility is evaluated at s_r.

The case for utility-aware approaches in games is further strengthened by the general volatility of the strategy-utility landscape. In particular, we see that for a wide characterization of games, the probability that a uniformly sampled strategy is near another strategy with a *different* best response increases with the dimensionality of the game. This phenomenon points to why, in some instances, small perturbations in state predictions result in completely different utilities.

Definition 3.4. Let A, B be two players in a bi-matrix game $G = (\mathbf{A}, \mathbf{B})$. For player A, we define the best response region for each action b of player B to be $R_b^A := \{\alpha \in \Delta(\mathcal{A}) | b \in BR(\alpha)\}$.

Definition 3.5. We say that a particular strategy $\alpha \in R_b^A$ is $\delta-$**regional** if $\exists \alpha' \in R_{b'}^A$ with $b \neq b'$ such that $\|\alpha - \alpha'\|_2 \leq \delta$.

In the above definitions, we formally describe best response regions (i.e., collections of strategies that elicit the same best response from a follower) and δ-regional strategies (strategies that, despite belonging to a particular best response region, are very close to another best response region). We seek to understand the prevalence of $\delta-$regional strategies.

Assumption 3.6. Let $\{G_N\}_{N=2}^{\infty}$ be a sequence of two-player bi-matrix games in which both players have N actions, indexed by $[N] = \{1, \ldots, N\}$. For each action $b \in [N]$ for player B, the best-response region for player A, $R_b^{A,(N)}$, is such that (1) $R_b^{A,(N)} \neq \emptyset$ and (2) for all $\alpha \in R_b^{A,(N)}$, we have $\alpha_b \geq 2\delta_N$, where α_b denotes the probability that player A's mixed strategy α places on its b-th action and $\{\delta_N\}$ is a sequence of positive numbers.

We briefly remark on the strength of these conditions. Condition (1) simply requires that no player B action is strictly dominated—that is, every action in the player B's action set is a best response to some player A mixed strategy. This is a mild non-degeneracy requirement.

Condition (2) is more substantive but natural in a broad class of games. It asks that whenever player B's best response is action b, player A places at least probability $2\delta_N$ on their own action b. Since the numbering of actions is arbitrary, the condition need only hold under *some* permutation. In security game formulations, for instance, the permutation pairs each attacker action b with a defender action $\sigma(b)$, and the condition reduces to requiring that each attacked cell is driven by the defender's commitment to patrolling some corresponding location—a mild structural requirement.

Theorem 3.7. *Suppose the sequence of games $\{G_N\}_{N=2}^{\infty}$ satisfies Assumption 3.6 with $\delta_N = c/N$ for some $c \in (0, \frac{1}{2}]$. For each N, let $\alpha \sim \mathrm{Unif}(\Delta(\mathcal{A}))$ in game G_N, then:*

$$\mathbb{P}\big(\alpha \text{ is } \delta_N\text{-regional}\big) \;\geq\; 1 - \left(1 - \frac{c\sqrt{2}}{N}\right)^{N-1}.$$

Theorem 3.7 illustrates how volatility naturally arises in games. For games satisfying Assumption 3.6, a uniformly sampled leader strategy is δ_N-regional with probability at least $1 - \left(1 - \frac{c\sqrt{2}}{N}\right)^{N-1}$. This bound increases with N and has the non-vanishing limit $1 - e^{-c\sqrt{2}}$, which is approximately 0.507 at $c = \frac{1}{2}$. As the dimensionality of the game grows, small perturbations to the leader's strategy may therefore suffice to change the follower's response. This susceptibility to downstream disruption is precisely what motivates a decision-focused approach.

Remark 3.8. Theorem 3.7 bounds the prevalence of near-boundary strategies in strategy space; it makes no statement about utilities. Two steps connect it to utilities: a small change in the prediction can move the induced strategy across a best-response boundary, and where adjacent regions carry different leader utilities the realized utility then can change sharply. Games with such utility differences across adjacent regions are possible.

4 End to End Predictions in Strategic Settings

The results of Section 3 establish that no accuracy-based training objective can serve as a reliable proxy for strategic utility. This motivates a fundamentally different approach to learning in the SDFL setting, which we call **End-To-End** (ETE) prediction selection: rather than optimizing for predictive accuracy in isolation, ETE directly targets the downstream strategic objective. We present this framework as an initial algorithmic response to SDFL — a proof of concept demonstrating tractability rather than a definitive solution, in the same spirit as early end-to-end approaches in single-agent DFL that preceded formal convergence analysis [3, 31].

Formally, let $f_\theta : \mathcal{X} \to \mathcal{S}$ be a parameterized model mapping observable features $x \in \mathcal{X}$ (derived from dataset $\mathcal{D}$) to a predicted state $f_\theta(x) \in \mathcal{S}$. The ETE problem is to find parameters θ^* that maximize the leader's realized utility at equilibrium as per equation 1. This objective is challenging to optimize directly: the true state, s_r, is unobserved at training time, and the mapping $\theta \mapsto U_i(\cdot)$ passes through an equilibrium computation that is in general non-differentiable in θ.

In the Stackelberg setting, this non-differentiability is concrete: $\bar{\alpha}_i^{f_\theta(x)}$ is the solution to a linear program (LP) whose payoff matrices are determined by $f_\theta(x)$, and the LP argmax is non-smooth. Moreover, by Proposition 3.2 and Theorem 3.7, the objective is neither monotone in prediction accuracy nor smooth in θ, so optimizing a supervised proxy is unreliable. These challenges motivate a two-stage development. We first consider the case where $\mathcal{S}$ is discrete and finite: here, utility can be evaluated exactly for any candidate prediction, which isolates the algorithmic question of how to search over predictions by strategic utility. We then turn to the general continuous setting, where both sources of difficulty are present.

4.1 Warm-up: End to End State Prediction with Discrete, Finite State Spaces

A core principle of ETE is that candidate predictions should be evaluated by the strategic utility they induce, not solely by statistical accuracy. A good ETE algorithm must therefore explore candidate predictions and filter them by downstream performance. We begin by showing this explore-then-filter principle is sound. In the discrete, finite-state setting, bandit-style approaches such as successive elimination can enumerate candidate predictions and evaluate each by its induced utility. We show that the cumulative cost of exploring suboptimal predictions is sublinear in T, even when the equilibrium induced by each candidate must itself be learned (provided per-state equilibrium computation is also sublinear in T).

Theorem 4.1. *Consider a game G with a finite set of exogenous states, and let s_r denote the true realized state. Let $\alpha_{i,t}^s$ denote the strategy selected by player i at time step t given state prediction s produced by algorithm $\mathcal{ALG}$, and let $(\bar{\alpha}_i^s, \bar{\alpha}_{-i}^s)$ be the unique equilibrium joint strategy profile given prediction state s. Suppose utilities are bounded, i.e. $U_i(\cdot, \cdot, s) \in [0, 1]$ for all s. If*

$$\mathop{\mathbb{E}}_{\mathcal{ALG}(s)}\left[\sum_{t=1}^{T}\left|U_i(\alpha_{i,t}^s, \alpha_{-i,t}^s, s') - U_i(\bar{\alpha}_i^s, \bar{\alpha}_{-i}^s, s')\right|\right] \in o(T) \quad \forall \textit{fixed } (s, s'),$$

then $\mathcal{ALG}'$, which combines $\mathcal{ALG}$ with successive elimination to determine an evaluation state, yields:

$$\mathop{\mathbb{E}}_{\mathcal{ALG}'}\left[\sum_{t=1}^{T}\left|U_i(\alpha_{i,t}^{s_t}, \alpha_{-i,t}^{s_t}, s_r) - \max_{s\in\mathcal{S}} U_i(\bar{\alpha}_i^s, \bar{\alpha}_{-i}^s, s_r)\right|\right] \in o(T).$$

4.2 End-to-End Prediction in General Settings

The warm-up setting assumes a finite, discrete state space and allows repeated evaluations of the same state. These luxuries rarely hold in practice. In general applications, the state space is continuous or combinatorially large, and we are given a dataset from which we are to return a single model for state prediction. In this paradigm, we are therefore unable to directly apply bandit-style enumeration. We thus seek an approach that retains the core principle of evaluating candidate predictions via their downstream strategic utility within a single-pass learning framework.

The theory above presents two concrete challenges for optimizing in the SDFL domain. Firstly, Propositions 3.2 and 3.3 establish that the mapping $\theta \mapsto U_i^{\text{realized}}$ is fundamentally misaligned with accuracy-based surrogates. This renders approaches such as gradient descent on prediction error alone insufficient. Secondly, Theorem 3.7 establishes that strategies close to a best-response boundary grow increasingly prevalent as game dimensionality increases; where adjacent regions carry different leader utilities, the utility landscape is discontinuous across the boundary. We therefore derive a heuristic tailored to the discontinuous optimization landscape induced by strategic responses in SDFL problems.

Algorithm 1 addresses these challenges by combining three ideas. First, we perform *intra-region optimization* within regions where the strategic response is fixed, using KKT-directed gradients that exploit differentiability within a fixed best-response zone to directly optimize through the follower's best response. Within these strategic epochs, the Huber regularisation weight λ_e is annealed from $\lambda_{\max}$ to $\lambda_{\min}$, keeping prediction quality grounded in early KKT steps when the strategic gradient would otherwise dominate and destabilize training. Second, we perform *inter-region exploration* to optimize across best-response regions via supervised updates, which shift the induced leader strategy across zone boundaries. A strategic checkpoint filter, motivated by the explore-then-filter principle

Algorithm 1 Hybrid ETE: Alternating Predictive/Strategic Regimes

Input: $\mathcal{D}$, f_θ, T, R, ρ, σ, $\lambda_{\max}$, $\lambda_{\min}$, E_{anneal}
$\mathcal{D}_{\text{grad}}, \mathcal{D}_{\text{eval}} \leftarrow \text{Split}(\mathcal{D}, \rho)$; $\theta^* \leftarrow \theta$; $u^* \leftarrow -\infty$; $e_{\text{kkt}} \leftarrow 0$
for $r = 1$ **to** $\lceil T/R \rceil$ **do**
  $\theta \leftarrow \theta^*$
  for $e = 1$ **to** R **do**
    if r odd: $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{pred}}(f_\theta(X_\mathcal{B} + \epsilon), Y_\mathcal{B})$, $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$
    else: $\lambda_e \leftarrow \lambda_{\max} - (\lambda_{\max} - \lambda_{\min}) \cdot \min\Big(\frac{e_{\text{kkt}}}{E_{\text{anneal}}}, 1\Big)$; $e_{\text{kkt}} \mathrel{+}= 1$;
      $\theta \leftarrow \theta - \eta \nabla_\theta \big[-\mathcal{U}_{\text{KKT}}(f_\theta, \mathcal{B}) + \lambda_e \, \mathcal{L}_{\text{pred}}(f_\theta(X_\mathcal{B}), Y_\mathcal{B})\big]$
  end for
  if $\mathcal{U}(f_\theta, \mathcal{D}_{\text{eval}}) > u^*$: $u^* \leftarrow \mathcal{U}(f_\theta, \mathcal{D}_{\text{eval}})$, $\theta^* \leftarrow \theta$
end for
return θ^*

of Theorem 4.1, selects across candidates produced by both mechanisms to navigate the globally non-convex and discontinuous landscape.

KKT-directed gradients. Given a predicted state s_{pred}, the leader's optimal strategy $\bar{\alpha}_i$ solves a linear program whose KKT conditions form a differentiable system in $\bar{\alpha}_i$ and the payoff parameters. For a fixed follower pure action a_{-i}, implicit differentiation yields $\partial \bar{\alpha}_i / \partial s_{pred}$, enabling backpropagation of strategic utility to θ. However, as Theorem 3.7 predicts, near best-response boundaries the follower's action switches discontinuously, and the KKT gradient—derived with a_{-i} fixed—becomes unreliable. This motivates alternation: odd regimes apply supervised updates with Gaussian input perturbations to traverse $\Delta(\mathcal{A}_i)$ and escape boundary regions, while even regimes apply $\mathcal{U}_{\text{KKT}}$ where the gradient is well-defined.

Strategic checkpoint filter. At the close of every regime of R epochs—whether a supervised or KKT regime—the current parameters θ are evaluated on a held-out set $\mathcal{D}_{\text{eval}}$ via the strategic utility function $\mathcal{U}$, and accepted only if they improve upon the best recorded utility u^*; otherwise training reverts to θ^*. This mechanism is motivated in part by Theorem 4.1: we identify model parameters by evaluating them for their downstream strategic utility.

Remark 4.2. The checkpoint filter selects a model whose expected strategic utility is within ε of the best candidate: $\mathcal{U}(\theta^*) \geq \max_{1 \leq k \leq K} \mathcal{U}(\theta_k) - \varepsilon$ with probability at least $1 - \delta$, provided $|\mathcal{D}_{\text{eval}}| \geq 2 \ln(2K/\delta)/\varepsilon^2$. This follows from standard arguments using Hoeffding's inequality and a union bound over the $K = \lceil T/R \rceil$ checkpoints.

4.3 Experimental Evaluation: Security Games

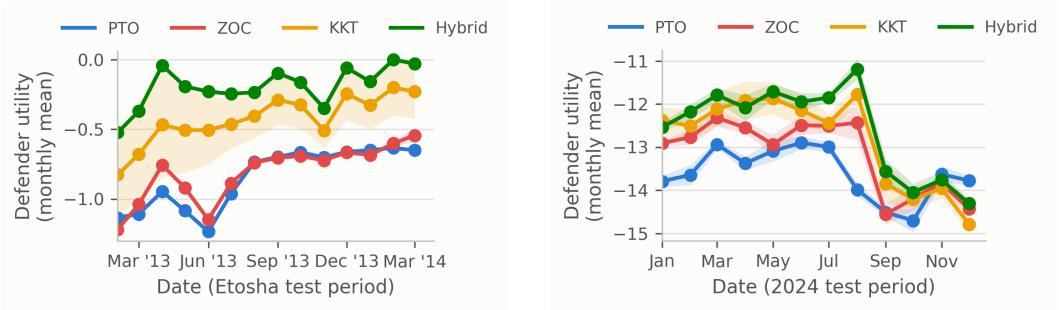
Building on game-theoretic approaches to security [25, 8, 13], we instantiate two Stackelberg security games on an $N \times N$ grid G: a conservation game using GPS tracks of 15 African elephants in Etosha National Park [9], and an infrastructure game using ridership data from 123 Manhattan subway stations (Jan 2022–Dec 2024) [19]. In both, a defender commits to a mixed strategy $\alpha_i \in \Delta(G)$ over patrol allocations, and an attacker who observes this commitment chooses $\alpha_{-i} \in \Delta(G)$ [32]. The exogenous state $s \in \mathbb{Z}_+^N$ captures the value at risk per cell: uncontested attacks cost the defender utility proportional to the cell value n (and reward the attacker proportionally), successful interdiction rewards the defender proportionally and penalizes the attacker by a fixed amount, and unvisited cells contribute zero. Full details are in Appendix B.

4.3.1 Methodology

We frame the problem as supervised regression: given temporal and historical features for each cell in G, predict the scalar value (elephant count or ridership) at a future period. All methods share a common base neural network architecture; comprehensive details on implementation, feature engineering, and training are in Appendix B.

Results. We compare four methods. **PTO** minimizes predictive loss and feeds the resulting prediction directly to the Stackelberg LP. The three ETE variants differ in how strategic information

enters training. **ZOC** (Zeroth-Order Checkpoint) trains on supervised loss but applies a strategic checkpoint filter after each regime, retaining only parameters that improve utility on $\mathcal{D}_{\text{eval}}$. **KKT** differentiates through the Stackelberg LP via KKT conditions, backpropagating strategic utility directly to model parameters. **Hybrid** (Algorithm 1) alternates supervised and KKT regimes with the checkpoint filter applied after each. We evaluate on two axes: Mean Absolute Error $\text{MAE} = \frac{1}{N|\mathcal{T}|}\sum_{t\in\mathcal{T}}\sum_{i\in[N]}|\hat{y}_t^i - y_t^i|$ and downstream strategic utility across all evaluation instances. Table 1 reports both metrics at 500 epochs—the budget at which all methods converged—across both game environments. Utility is averaged over all evaluation instances (396 daily instances for conservation, 1098 for infrastructure) and 5 independent training runs; MAE is computed analogously. Figure 2 shows the same utilities month by month over each test period.



(a) Conservation Security Game (Etosha), test period February 2013 to March 2014.

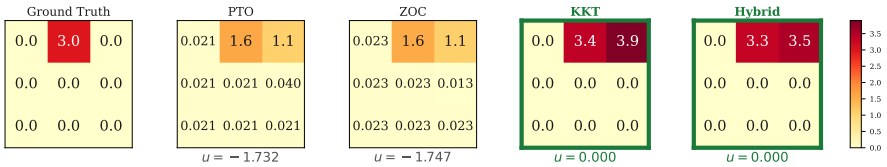
(b) Infrastructure Security Game (Manhattan subway), test period January to December 2024.

Figure 2: Defender utility over the test period for the four methods. Each point is the mean defender utility over the evaluation instances in one calendar month, averaged over the five training runs; the shaded band is one standard error across runs. Higher is better. Averaging each curve over all evaluation instances gives the utilities in Table 1.

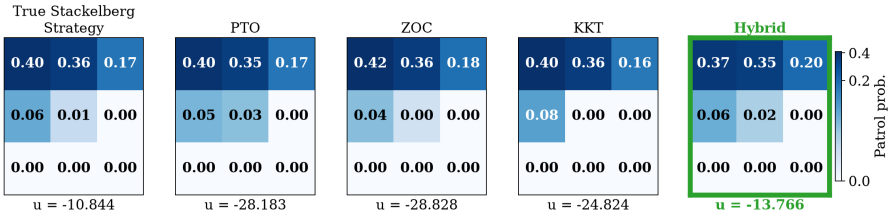
Table 1: ETE vs. PTO at 500 training epochs (mean ± std across 5 runs).

		Conservation Security Game		**Infrastructure Security Game**	
	Method	**Avg. Utility** (↑)	**MAE** (↓)	**Avg. Utility** (↑)	**MAE** (↓)
ETE	ZOC	-0.785 ± 0.009	0.118 ± 0.006	-13.160 ± 0.355	0.559 ± 0.062
	KKT	-0.404 ± 0.483	0.433 ± 0.159	-12.829 ± 0.465	1.104 ± 0.467
	Hybrid	$\mathbf{-0.173 \pm 0.013}$	0.458 ± 0.107	$\mathbf{-12.578 \pm 0.383}$	0.736 ± 0.028
	PTO	-0.831 ± 0.005	$\mathbf{0.110 \pm 0.005}$	-13.611 ± 0.210	$\mathbf{0.409 \pm 0.009}$

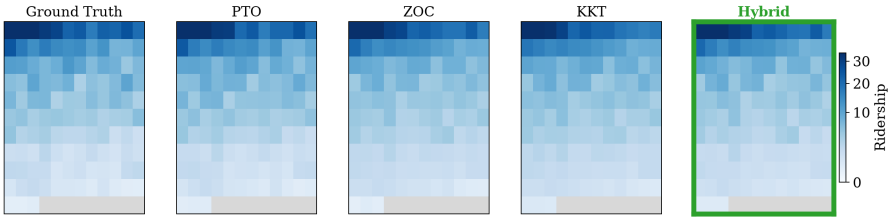
PTO, on average, achieves lower MAE than all ETE approaches, indicating superior predictive accuracy in the conventional supervised learning sense. However, this advantage does not translate into better strategic outcomes: all three ETE methods achieve higher defender utility than PTO in both domains. Critically, Hybrid achieves the best strategic utility across both game environments — outperforming not only PTO but also ZOC and KKT in isolation — demonstrating that combining supervised exploration, KKT-directed gradients, and the strategic checkpoint filter yields consistent gains that neither component delivers on its own.



(a) Conservation Security Game: Elephant Distribution Predictions



(b) Infrastructure Security Game: Patrol Strategies across top 9 stations



(c) Infrastructure Security Game: Ridership Predictions across 123 stations

Figure 3: Instance-level state predictions for select evaluation periods. Winning method(s) shown in **bold green**; utility u reported below each prediction grid.

Figure 3 illustrates the instance-level dynamics underlying these aggregate results. The utility landscape is complex and volatile—consistent with Theorem 3.7—and no single method universally dominates at the individual instance level. In instances where ETE methods outperform PTO, the ETE predictions better capture strategically relevant structure in the state, even at the cost of raw predictive accuracy. We illustrate such instances in the above figure. To probe the general landscape and gain an understanding of how in some cases we may see variation with PTO approaches, we include a 100 action synthetic setting on a grid with multiple modes in Appendix B.1: here we illustrate how different state representations produce distinct downstream strategic consequences.

5 Conclusion

In this work, we formalized *strategic decision-focused learning* as a distinct problem within the decision-focused learning paradigm, targeting settings where ML predictions are consumed by strategically interacting agents. We showed that prediction accuracy and strategic payoff can be non-monotonic, motivating a departure from predict-then-optimize pipelines, and proposed an end-to-end framework that incorporates downstream strategic utility directly into model selection via KKT-directed gradients and strategic evaluation on a held-out partition.

Several important directions remain open. Our formulation assumes followers observe the true state; modeling settings in which followers must also learn the state from data introduces a second layer of strategic complexity worth exploring. More broadly, precisely characterizing the conditions under which equilibrium-aware end-to-end approaches outperform predict-then-optimize, and developing scalable algorithms for this regime, are promising avenues for future work.

References

[1] Bechavod, Y., Ligett, K., Wu, S., and Ziani, J. Gaming helps! learning from strategic interactions in natural dynamics. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, pp. 1234–1242, 2021.

[2] Bhatia, R. *Matrix Analysis*, volume 169 of *Graduate Texts in Mathematics*. Springer, 1997.

[3] Donti, P. L., Amos, B., and Kolter, J. Z. Task-based end-to-end model learning in stochastic optimization. In *Advances in Neural Information Processing Systems*, 2017.

[4] Dritsas, E. and Trigka, M. Machine learning in e-commerce: Trends, applications, and future challenges. *IEEE Access*, 13:99048–99067, 2025. doi: 10.1109/ACCESS.2025.3572865.

[5] Elmachtoub, A. N. and Grigas, P. Smart "predict, then optimize". *Management Science*, 68: 9–26, 2021.

[6] Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AeroGRID, and IGN. World Imagery [basemap]. Esri, `https://www.arcgis.com/home/item.html?id=10df2279f9684e4a9f6a7f08febac2a9`, 2024. Accessed 2024.

[7] Even-Dar, E., Mannor, S., and Mansour, Y. PAC bounds for multi-armed bandit and markov decision processes. In Kivinen, J. and Sloan, R. H. (eds.), *Computational Learning Theory*, volume 2375 of *Lecture Notes in Computer Science*, pp. 255–270. Springer, 2002.

[8] Fang, F., Stone, P., and Tambe, M. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence*, volume 15, 2015.

[9] Getz, W. M., Kilian, W., Zidon, R., and Tsalyuk, M. Data from: Temporal variation in resource selection of African elephants follows long-term variability in resource availability, 2018.

[10] Handina, T. and Mazumdar, E. Understanding model selection for learning in strategic environments. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL `https://proceedings.neurips.cc/paper_files/paper/2024/file/a921f335253add9996d5175ad30896ec-Paper-Conference.pdf`.

[11] Hardt, M., Megiddo, N., Papadimitriou, C., and Wootters, M. Strategic classification. In *Innovations in Theoretical Computer Science*, pp. 111–122, 2016.

[12] Knaster, B., Kuratowski, C., and Mazurkiewicz, S. Ein Beweis des Fixpunktsatzes für n-dimensionale Simplexe. *Fundamenta Mathematicae*, 14(1):132–137, 1929.

[13] Krever, N., Cerny, J., Blanchard, M., and Kroer, C. Guard: Constructing realistic two-player matrix and security games for benchmarking game-theoretic algorithms. In *NeurIPS Datasets and Benchmarks Track (Spotlight)*, 2025. URL `https://arxiv.org/abs/2505.14547`.

[14] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR*, 2018.

[15] Mandi, J., Kotary, J., Berden, S., Mulamba, M., Bucarey, V., Guns, T., and Fioretto, F. Decision-focused learning: Foundations, state of the art, benchmark and future opportunities. 80, 2024. URL `https://doi.org/10.1613/jair.1.15320`.

[16] Mazumdar, E., Ratliff, L. J., and Sastry, S. S. On gradient-based learning in continuous games. *SIAM Journal on Mathematics of Data Science*, 2(1):103–131, 2020.

[17] Mertikopoulos, P. and Zhou, Z. Learning in games with continuous action sets and unknown payoff functions. *Mathematical Programming*, 173(1):465–507, 2019.

[18] Mertikopoulos, P., Papadimitriou, C., and Piliouras, G. Cycles in adversarial regularized learning. In *Proceedings of the 2018 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2018.

[19] Metropolitan Transportation Authority. MTA Subway Hourly Ridership: 2020–2024. https://data.ny.gov/Transportation/MTA-Subway-Hourly-Ridership-2020-2024/wujg-7c2s, 2025. Accessed: 2026-03-23.

[20] Mhasawade, V., Zhao, Y., and Chunara, R. Machine learning and algorithmic fairness in public and population health. *Nature Machine Intelligence*, 3(8):659–666, 2021.

[21] Mirzakhani, M. and Vondrák, J. Sperner's colorings and optimal partitioning of the simplex. In Loebl, M., Nešetřil, J., and Thomas, R. (eds.), *A Journey through Discrete Mathematics: A Tribute to Jiří Matoušek*. Springer, 2017.

[22] OpenStreetMap contributors. OpenStreetMap. https://www.openstreetmap.org, 2024. Map data licensed under the Open Database Licence (ODbL). Retrieved 2024.

[23] Perdomo, J., Zrnic, T., Mendler-Dünner, C., and Hardt, M. Performative prediction. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 7599–7609, 2020.

[24] Perrault, A., Wilder, B., Ewing, E., Mate, A., Dilkina, B., and Tambe, M. End-to-end game-focused learning of adversary behavior in security games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[25] Pita, J., Jain, M., Marecki, J., Ordóñez, F., Portway, C., Tambe, M., Western, C., Paruchuri, P., and Kraus, S. Deployed ARMOR protection: The application of a game theoretic model for security at the Los Angeles International Airport. In *Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Industry Track*, pp. 125–132, 2008.

[26] Robson, A. S., Trimble, M. J., Purdon, A., Young-Overton, K. D., Pimm, S. L., and van Aarde, R. J. Savanna elephant numbers are only a quarter of their expected values. *PLOS ONE*, 12(4), 2017.

[27] Shah, S., Wang, K., Wilder, B., Perrault, A., and Tambe, M. Decision-focused learning without decision-making: Learning locally optimized decision losses. *Advances in Neural Information Processing Systems*, 35:1320–1332, 2022.

[28] Tang, Y., Sun, J., Wang, H., Deng, J., Tong, L., and Xu, W. A method of network attack-defense game and collaborative defense decision-making based on hierarchical multi-agent reinforcement learning. *Computers & Security*, 142:103871, 2024.

[29] Tarallo, E., Akabane, G. K., Shimabukuro, C. I., Mello, J., and Amancio, D. Machine learning in predicting demand for fast-moving consumer goods: An exploratory research. *IFAC-PapersOnLine*, 52(13):737–742, 2019.

[30] Wang, K., Perrault, A., Mate, A., and Tambe, M. Scalable game-focused learning of adversary models: Data-to-decisions in network security games. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, 2020.

[31] Wilder, B., Dilkina, B., and Tambe, M. Melding the data-decisions pipeline: Decision-focused learning for combinatorial optimization. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 1658–1665, 2019.

[32] Yang, R., Ford, B., Tambe, M., and Lemieux, A. Adaptive resource allocation for wildlife protection against illegal poachers. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, 2014.

[33] Yeh, C., Christianson, N., Wu, A., Wierman, A., and Yue, Y. End-to-end conformal calibration for optimization under uncertainty. *ArXiv*, abs/2409.20534, 2024. URL https://api.semanticscholar.org/CorpusID:272987523.

[34] Zhang, E. et al. Position: Social environment design should be further developed for ai-based policy-making. In *Proceedings of the Forty-First International Conference on Machine Learning*, 2024.

[35] Zhang, K., Yang, Z., and Başar, T. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control*, pp. 321–384, 2021.

[36] Zrnic, T., Mazumdar, E., Sastry, S., and Jordan, M. Who leads and who follows in strategic classification? In *Advances in Neural Information Processing Systems*, 2021.

Appendix

Limitations and Broader Impact

Limitations. Our formulation assumes that followers observe the true state of the world; in practice, followers may also need to learn the state from data, introducing a second layer of strategic complexity that our framework does not capture. We provide a local non-monotonicity result motivating end-to-end training, but a precise characterization of when equilibrium-aware approaches outperform predict-then-optimize remains open.

Broader impact. Strategic decision-focused learning is motivated by settings where a decision-maker deploys ML predictions against agents who respond strategically — a structure common in security, infrastructure allocation, and market design. Improved patrol allocation in conservation and transit safety are positive applications, but the same framework could in principle be used in ways that raise fairness and civil-liberties concerns.

A Proofs of Theoretical Results

Restatement of Proposition 3.2. *There exists a game instance with true state s_r, and predictions $s_{p,1}$ and $s_{p,2}$, wherein $s_{p,1}$ is strictly more accurate than $s_{p,2}$ under every accuracy metric ϕ, yet player i's utility at the equilibrium induced by $s_{p,1}$ is strictly lower than that induced by $s_{p,2}$:*

$$U_i(\alpha^*(s_{p,1}),\, s_r) \;<\; U_i(\alpha^*(s_{p,2}),\, s_r)\,,$$

where $\alpha^*(s) = (\alpha_i^*(s),\, BR_{-i}(\alpha_i^*(s), s_r))$ denotes the equilibrium joint strategy profile when the primary plays assuming state s, the rest of the agents see the true state s_r, and utility is evaluated at s_r.

Proof of Proposition 3.2. We consider a Stackelberg game between two players, A and B, in which player A leads. Let $\mathcal{S} :=$

$$\left\{ s_0 = \begin{bmatrix}1\\0\\1\\1\\3\\1\\1\\0\\1\end{bmatrix}, \quad s_1 = \begin{bmatrix}1\\1\\1\\1\\1\\1\\1\\1\\1\end{bmatrix}, \quad s_2 = \begin{bmatrix}0\\0\\0\\0\\9\\0\\0\\0\\0\end{bmatrix} \right\}$$

Let $\mathcal{A} = \mathcal{B} = \{x \in \mathbb{R}^9 : \sum_{i=1}^{9} x_i = 1, x_i \geq 0\ \forall i\}$. We define $\alpha \in \mathcal{A}$ and $\beta \in \mathcal{B}$ to be strategies by players A and B respectively. We denote the utility functions by the following:

$$U_A(\alpha, \beta, s) = \sum_{i=1}^{9} s_i \beta_i (2\alpha_i - 1); \quad U_B(\alpha, \beta, s) = \sum_{i=1}^{9} s_i \beta_i (1 - 2\alpha_i)$$

Given a state $s \in \mathcal{S}$, player B's payoff is linear in β, so their best response is always a vertex of $\mathcal{B}$, i.e., a pure strategy e_k for some $k \in \{1, \ldots, 9\}$.

Consider the following set of linear programs defined for each possible best response action of player B:

$$\begin{aligned} \alpha^{(s,k)} = \arg\max_{\alpha \in \mathcal{A}} \quad & s_k(2\alpha_k - 1) \\ \text{s.t.} \quad & s_k(1 - 2\alpha_k) \geq s_j(1 - 2\alpha_j) \quad \forall\, j \neq k \end{aligned}$$

The constraint ensures that action k is indeed player B's best response under α. Note that player A's Stackelberg strategy given state s is obtained by solving the linear programs and selecting the linear program with the highest value:

$$\alpha^s = \arg\max_k \Big\{ U_A(\alpha^{(s,k)}, e_k, s) : k \in \{1, \ldots, 9\} \Big\}$$

with Stackelberg value:

$$V^*(s) = \max_k \Big\{ U_A(\alpha^{(s,k)}, e_k, s) : k \in \{1, \ldots, 9\} \Big\}$$

Solving the above optimization problems for Stackelberg strategies given predictions of s_1 and s_2 yields: $\alpha^{s_1} = \begin{bmatrix} \frac{1}{9} \\ \frac{1}{9} \\ \frac{1}{9} \\ \frac{1}{9} \\ \frac{1}{9} \\ \frac{1}{9} \\ \frac{1}{9} \\ \frac{1}{9} \\ \frac{1}{9} \end{bmatrix}$ and for the state s_2 we have a set of possible strategies, i.e., $\alpha^{s_2} \in A^{s_2} = \{\alpha \in \mathcal{A} : \alpha_5 > \frac{1}{2}\}$.

Let s_0 be the real state and s_1 and s_2 be predictions. We calculate the utility of the Stackelberg strategies under the state s_0 and find that:

For Strategy α^{s_1}: Recall $\alpha^{s_1} = \frac{1}{9}\mathbf{1}$. Under s_0, player B's payoff from playing action i is $s_{0,i}(1 - 2\alpha_i^{s_1}) = \frac{7}{9}s_{0,i}$, which is maximized at $i = 5$ (where $s_{0,5} = 3$). Hence the best response is e_5, giving player A the utility:

$$U_A(\alpha^{s_1}, e_5, s_0) = 3\left(\frac{2}{9} - 1\right) = -\frac{7}{3}.$$

Worst-case strategy in A^{s_2}. Recall $A^{s_2} = \{\alpha \in \Delta(\mathcal{A}) : \alpha_5 > \frac{1}{2}\}$. For any $\alpha \in A^{s_2}$, we assess player B's best response under s_0. Actions $i \in \{2, 8\}$ satisfy $s_{0,i} = 0$, so $U_B(e_i, \alpha, s_0) = 0$ regardless of α. Action $i = 5$ yields $U_B(e_5, \alpha, s_0) = 3(1 - 2\alpha_5) < 0$ since $\alpha_5 > \frac{1}{2}$. For the remaining actions $i \in \{1, 3, 4, 6, 7, 9\}$, note that $\sum_{i \neq 5} \alpha_i < \frac{1}{2}$, so at least one such action corresponds to $\alpha_i < \frac{1}{2}$, giving $U_B(\alpha, e_i, s_0) = 1 - 2\alpha_i > 0$. Since these actions offer strictly positive utility to player B, player B's best response lies in $\{e_k : k \in \{1, 3, 4, 6, 7, 9\}\}$ for any $\alpha \in A^{s_2}$ with Player A's resulting utility as $U_A = 2\alpha_k - 1$. With this we note that

$$\inf_{\alpha \in A^{s_2}} U_A(\alpha, \beta^*(\alpha), s_0) = -1.$$

Comparison. Even in the worst case over A^{s_2}, player A's utility is strictly greater than under α^{s_1}:

$$\inf_{\alpha \in A^{s_2}} U_A(\alpha, \beta^*(\alpha), s_0) = -1 \; > \; -\frac{7}{3} = U_A(\alpha^{s_1}, e_5, s_0).$$

The sorted error vectors, i.e. the absolute errors $|s_1 - s_0|$ and $|s_2 - s_0|$ arranged in decreasing order, are $(2, 1, 1, 0, 0, 0, 0, 0, 0)$ and $(6, 1, 1, 1, 1, 1, 1, 0, 0)$, so every top-$k$ partial sum of the former is at most half the latter's, and thus by [2, Ch. IV]:

$$\|s_1 - s_0\| \; \leq \; \tfrac{1}{2} \|s_2 - s_0\| \; < \; \|s_2 - s_0\|$$

for every symmetric norm simultaneously. Therefore, despite s_2 being a less accurate prediction of s_0 under every accuracy metric of Definition 3.1 (for instance $\|s_2 - s_0\|_2 = \sqrt{42}$ versus $\|s_1 - s_0\|_2 = \sqrt{6}$), the Stackelberg strategy consistent with the prediction s_2 outperforms the Stackelberg strategy induced by the more accurate prediction s_1. Strict preference extends to any separable loss $\sum_i g(|e_i|)$, where e_i denotes the i-th coordinate of the error $s - s_0$, with g strictly increasing (e.g., Huber, MSE).

□

Restatement of Proposition 3.3. *There exists a game instance with true state s_r and player i such that there exists a prediction $s_p \in \mathcal{S}$ satisfying:*

$$U_i(\alpha^*(s_p),\, s_r) \;>\; U_i(\alpha^*(s_r),\, s_r)\,,$$

where $\alpha^*(s) = (\alpha_i^*(s),\, BR_{-i}(\alpha_i^*(s), s_r))$ denotes the equilibrium joint strategy profile when the primary plays assuming state s, the rest of the agents see the true state s_r, and utility is evaluated at s_r.

Proof of Proposition 3.3. We follow the proof structure and argument of Theorem 3.4 in [10]. We work in the following setting: we have a two-player continuous game in which the primary agent (player i) has strategy space from a compact set $\mathcal{A}_i \subseteq \mathbb{R}$ and the opposing agent (player $-i$) has strategy space from a compact set $\mathcal{A}_{-i} \subseteq \mathbb{R}$, with utilities $U_i(\alpha_i, \alpha_{-i}, s)$ and $U_{-i}(\alpha_i, \alpha_{-i}, s)$ respectively. The exogenous state space $\mathcal{S} = \{0, 1\}$. We assume the following.

Assumption A.1. Assume the game defined on U_i and U_{-i} is strongly monotone on $\mathcal{A}_i \times \mathcal{A}_{-i}$. Further assume that

1. U_i and U_{-i} are jointly concave in α_i and α_{-i}.
2. The gradient mappings ∇U_i and ∇U_{-i} exist and are well defined for all (α_i, α_{-i}). Furthermore, the gradient mappings are L-Lipschitz continuous in the joint action space.
3. The Nash equilibrium $(\alpha_i^\star, \alpha_{-i}^\star) \in \mathcal{A}_i \times \mathcal{A}_{-i}$ is in the interior with $\nabla_{\alpha_i} BR_{-i}(\alpha_i^\star, s_r) \neq 0$.
4. The unique Nash equilibrium $(\alpha_i^\star, \alpha_{-i}^\star)$ is not Pareto-optimal: there exists $(\alpha_i^\dagger, \alpha_{-i}^\dagger) \in \mathcal{A}_i \times \mathcal{A}_{-i}$ such that $U_i(\alpha_i^\dagger, \alpha_{-i}^\dagger, s_r) > U_i(\alpha_i^\star, \alpha_{-i}^\star, s_r)$.

Note the following: because we can achieve strictly higher utility for the jointly concave utility function U_i, we know that $\nabla U_i(\alpha_i^\star, \alpha_{-i}^\star, s_r) \neq 0$.

We define $\bar{U}_i(\alpha_i) := U_i(\alpha_i, BR_{-i}(\alpha_i, s_r), s_r)$ as the primary agent's utility when they play α_i and the opposing agent best-responds under the true state s_r. [10] establishes that there exists a $v \in \mathbb{R}$ with $v \cdot \nabla \bar{U}_i(\alpha_i^\star) < 0$ and a value $\delta > 0$ such that the following construction yields a Nash equilibrium with strictly higher utility for the primary agent. Specifically, setting $\alpha_i' = \alpha_i^\star - \delta v$, $\alpha_{-i}' = BR_{-i}(\alpha_i', s_r)$, and defining

$$\tilde{\Theta} = \{\alpha_i \in \mathcal{A}_i : (\alpha_i - \alpha_i') \cdot v \leq 0\}, \quad \Theta' = \{\alpha_i \in \tilde{\Theta} : \nabla_{\alpha_i} U_i(\alpha_i', \alpha_{-i}', s_r) \cdot (\alpha_i' - \alpha_i) \geq 0\},$$

the restricted game on $\Theta' \times \mathcal{A}_{-i}$ admits $(\alpha_i', \alpha_{-i}')$ as a Nash equilibrium. By a Taylor expansion argument using the L-Lipschitz gradient assumption, one can choose δ small enough so that $\bar{U}_i(\alpha_i') > \bar{U}_i(\alpha_i^\star)$, meaning the primary agent achieves strictly higher utility at this new equilibrium than at $(\alpha_i^\star, \alpha_{-i}^\star)$. To complete the argument, for player i let $s = 0$ correspond to $U_i(\alpha_i, \alpha_{-i}, 0) > -M \quad \forall (\alpha_i, \alpha_{-i})$ with $M \in \mathbb{R}_+$ and then for $s = 1$ let $U_i(\alpha_i, \alpha_{-i}, 1) = U_i(\alpha_i, \alpha_{-i}, 0) \quad \forall (\alpha_i, \alpha_{-i}) \in \Theta' \times \mathcal{A}_{-i}$ and $U_i(\alpha_i, \alpha_{-i}, 1) = -M$ otherwise. For player $-i$ we have $U_{-i}(\alpha_i, \alpha_{-i}, 1) = U_{-i}(\alpha_i, \alpha_{-i}, 0)$. In this simple setup, a prediction (s_p) of state 1 results in better utility for player i even if the real state (s_r) is 0.

Such games exist; for instance, on $\mathcal{A}_i = \mathcal{A}_{-i} = [0, 1]$ the game $U_i(\alpha_i, \alpha_{-i}, 0) = -\frac{1}{2}(\alpha_i - \alpha_{-i})^2 + \alpha_{-i}$, $U_{-i}(\alpha_i, \alpha_{-i}, 0) = -\frac{1}{2}\big(\alpha_{-i} - \frac{\alpha_i}{2} - \frac{1}{4}\big)^2$ satisfies the assumption: it is strongly monotone with $\mu = \frac{1}{4}$, both utilities are concave with affine gradients, $BR_{-i}(\alpha_i, 0) = \frac{\alpha_i}{2} + \frac{1}{4}$ has derivative $\frac{1}{2} \neq 0$, the unique Nash equilibrium $(\frac{1}{2}, \frac{1}{2})$ is interior, and $U_i(1, \frac{3}{4}, 0) = \frac{23}{32} > \frac{1}{2} = U_i(\frac{1}{2}, \frac{1}{2}, 0)$ and $U_{-i}(1, \frac{3}{4}, 0) = 0 = U_{-i}(\frac{1}{2}, \frac{1}{2}, 0)$. Consider the game restricted for α_i to $[\frac{3}{4}, 1]$, we get $\alpha_i' = \frac{3}{4}$, $\alpha_{-i}' = \frac{5}{8}$ and $\bar{U}_i(\frac{3}{4}) = \frac{79}{128} > \frac{1}{2} = \bar{U}_i(\frac{1}{2})$.

As a note, this proof, together with the establishment of v from [10], can be extended to more dimensions by adding the assumption that $\nabla_{\alpha_i} BR_{-i}(\alpha_i^\star, s_r)$ has rank equal to the dimension of $\mathcal{A}_{-i}$.

□

Restatement of Theorem 3.7. *Suppose the sequence of games $\{G_N\}_{N=2}^{\infty}$ satisfies Assumption 3.6 with $\delta_N = c/N$ for some $c \in (0, \frac{1}{2}]$. For each N, let $\alpha \sim \mathrm{Unif}(\Delta(\mathcal{A}))$ in game G_N, then:*

$$\mathbb{P}\big(\alpha \text{ is } \delta_N\text{-regional}\big) \;\geq\; 1 - \left(1 - \frac{c\sqrt{2}}{N}\right)^{N-1}.$$

Proof of Theorem 3.7. We instantiate the geometric arguments of [21] to our context. The conditions of Assumption 3.6 ensure that the best-response regions $\{R_b^{A,(N)}\}_{b\in[N]}$ form a Knaster–Kuratowski–Mazurkiewicz (KKM) covering [12] of the simplex $\Delta(\mathcal{A})$. Let μ_{N-1} denote the $(N-1)$-dimensional Lebesgue measure on $\Delta(\mathcal{A})$.

Define $S = \bigcup_{b\neq b'}(R_b^{A,(N)} \cap R_{b'}^{A,(N)})$ and its δ_N-neighborhood $S_{\delta_N} = \{\alpha \in \Delta(\mathcal{A}) : \exists\, \alpha' \in S,\ \|\alpha - \alpha'\|_2 \leq \delta_N\}$. By definition, α is δ_N-regional if and only if $\alpha \in S_{\delta_N}$. Set $\overline{R}_b^{A,(N)} = R_b^{A,(N)} \setminus S_{\delta_N}$, so that $\bigcup_{b\in[N]} \overline{R}_b^{A,(N)} = \Delta(\mathcal{A}) \setminus S_{\delta_N}$. Let $\delta'_N = \delta_N\sqrt{2}$ and define the shifted regions

$$\widetilde{R}_b^{A,(N)} \;=\; \overline{R}_b^{A,(N)} - \delta'_N e_b \;=\; \big\{\alpha - \delta'_N e_b : \alpha \in \overline{R}_b^{A,(N)}\big\}.$$

The following claim adapts the shrink-and-shift technique of [21, Lemma 3] to our setting.

Claim A.2. *The sets $\widetilde{R}_1^{A,(N)}, \ldots, \widetilde{R}_N^{A,(N)}$ are pairwise disjoint subsets of $(1 - \delta'_N) \cdot \Delta(\mathcal{A})$.*

Proof of Claim A.2. Disjointness. Suppose for contradiction that $\widetilde{R}_b^{A,(N)} \cap \widetilde{R}_{b'}^{A,(N)} \neq \emptyset$ for some $b \neq b'$. Then there exist $\alpha \in \overline{R}_b^{A,(N)}$ and $\beta \in \overline{R}_{b'}^{A,(N)}$ with $\alpha - \delta'_N e_b = \beta - \delta'_N e_{b'}$, giving $\|\alpha - \beta\| = \delta'_N \|e_b - e_{b'}\| = \delta'_N\sqrt{2} = 2\delta_N$. The midpoint $m = \frac{1}{2}(\alpha + \beta)$ lies in $\Delta(\mathcal{A})$ by convexity, so $m \in R_\ell^{A,(N)}$ for some ℓ with $\ell \neq b$ or $\ell \neq b'$ (possibly both). Assuming without loss of generality that $\ell \neq b$, closedness of $R_b^{A,(N)}$ and $R_\ell^{A,(N)}$ yields a point $\alpha' \in R_b^{A,(N)} \cap R_\ell^{A,(N)} \subseteq S$ on the segment from α to m, with $\|\alpha' - \alpha\| \leq \|m - \alpha\| = \delta_N$. This places $\alpha \in S_{\delta_N}$, contradicting $\alpha \in \overline{R}_b^{A,(N)}$.

Containment. For any $\alpha \in \overline{R}_b^{A,(N)}$, Assumption 3.6 gives $\alpha_b \geq 2\delta_N > \delta'_N = \delta_N\sqrt{2}$, so the shifted point $\alpha - \delta'_N e_b$ has all coordinates nonnegative and summing to $1 - \delta'_N$. Hence $\widetilde{R}_b^{A,(N)} \subseteq (1 - \delta'_N) \cdot \Delta(\mathcal{A})$. □

Since each $\widetilde{R}_b^{A,(N)}$ is a translate of $\overline{R}_b^{A,(N)}$, Claim A.2 gives

$$\begin{aligned}\mu_{N-1}\big(\Delta(\mathcal{A}) \setminus S_{\delta_N}\big) &= \sum_{b=1}^{N} \mu_{N-1}\big(\overline{R}_b^{A,(N)}\big) = \sum_{b=1}^{N} \mu_{N-1}\big(\widetilde{R}_b^{A,(N)}\big) \\ &\leq \mu_{N-1}\big((1 - \delta'_N) \cdot \Delta(\mathcal{A})\big) = (1 - \delta_N\sqrt{2})^{N-1}\, \mu_{N-1}\big(\Delta(\mathcal{A})\big),\end{aligned}$$

and therefore

$$\mathbb{P}\big(\alpha \text{ is } \delta_N\text{-regional}\big) = \frac{\mu_{N-1}(S_{\delta_N})}{\mu_{N-1}(\Delta(\mathcal{A}))} \geq 1 - (1 - \delta_N\sqrt{2})^{N-1},$$

where $\alpha \sim \mathrm{Unif}(\Delta(\mathcal{A}))$. Setting $\delta_N = c/N$ for $c \in (0, \frac{1}{2}]$, this becomes

$$\mathbb{P}\big(\alpha \text{ is } \delta_N\text{-regional}\big) \geq 1 - \left(1 - \frac{c\sqrt{2}}{N}\right)^{N-1}.$$

□

Corollary A.3. *Under the hypotheses of Theorem 3.7,*

$$\liminf_{N\to\infty}\ \mathbb{P}\big(\alpha \text{ is } \delta_N\text{-regional}\big) \;\geq\; 1 - e^{-c\sqrt{2}}.$$

Proof. Since $\ln(1-x) \le -x$ for all $x \in (0,1)$,

$$\left(1 - \frac{c\sqrt{2}}{N}\right)^{N-1} \le e^{-c\sqrt{2}(1-1/N)},$$

and taking $N \to \infty$ yields the result. □

Restatement of Theorem 4.1. *Consider a game G with a finite set of exogenous states, and let s_r denote the true realized state. Let $\alpha^s_{i,t}$ denote the strategy selected by player i at time step t given state prediction s produced by algorithm $\mathcal{ALG}$, and let $(\bar{\alpha}^s_i, \bar{\alpha}^s_{-i})$ be the unique equilibrium joint strategy profile given prediction state s. Suppose utilities are bounded, i.e. $U_i(\cdot,\cdot,s) \in [0,1]$ for all s. If*

$$\mathop{\mathbb{E}}_{\mathcal{ALG}(s)}\left[\sum_{t=1}^{T}\left|U_i(\alpha^s_{i,t}, \alpha^s_{-i,t}, s') - U_i(\bar{\alpha}^s_i, \bar{\alpha}^s_{-i}, s')\right|\right] \in o(T) \quad \forall \textit{fixed } (s, s'),$$

then $\mathcal{ALG}'$, which combines $\mathcal{ALG}$ with successive elimination to determine an evaluation state, yields strategies satisfying

$$\mathop{\mathbb{E}}_{\mathcal{ALG}'}\left[\sum_{t=1}^{T}\left|U_i(\alpha^{s_t}_{i,t}, \alpha^{s_t}_{-i,t}, s_r) - \max_{s\in\mathcal{S}} U_i(\bar{\alpha}^s_i, \bar{\alpha}^s_{-i}, s_r)\right|\right] \in o(T).$$

Proof of Theorem 4.1. **Step 1: Bandit Reduction.** Since $\mathcal{S}$ is finite, we treat each state $s \in \mathcal{S}$ as an arm in a multi-armed bandit problem. The goal of successive elimination is to identify a best evaluation state, i.e. a state s under which $\mathcal{ALG}$ achieves the highest utility when evaluated at the true state s_r; by Proposition 3.3 this state need not be s_r itself. The key difficulty is that within a single run of $\mathcal{ALG}(s)$, the rewards

$$X^s_t := U_i(\alpha^s_{i,t}, \alpha^s_{-i,t}, s_r)$$

are correlated across time steps t, since $\mathcal{ALG}$ is adaptive. We resolve this by redefining what constitutes a single arm pull via an epoch structure.

Notation. For each state $s \in \mathcal{S}$ write

$$R^s := U_i(\bar{\alpha}^s_i, \bar{\alpha}^s_{-i}, s_r)$$

for the arm's true value, the equilibrium utility of prediction s evaluated at the true state; R^s is free of any epoch index. Let $s^* \in \arg\max_{s\in\mathcal{S}} R^s$ be a best evaluation state; it need not equal s_r. Write $\Delta_s := R^{s^*} - R^s \ge 0$ for the gaps at these values and $\Delta_{\min}$ for the smallest positive gap. (If no gap is positive, every state is optimal and only the exploration cost below is incurred.)

Step 2: Epoch Structure. Define epochs $k = 1, 2, \ldots$ with lengths $T_k = k$. Let $\mathcal{S}_k \subseteq \mathcal{S}$ denote the set of surviving arms entering epoch k, initialized to $\mathcal{S}_1 = \mathcal{S}$. In each epoch k, for every surviving arm $s \in \mathcal{S}_k$, we run a fresh independent instantiation of $\mathcal{ALG}(s)$ over T_k rounds and compute the empirical reward

$$\hat{R}^s_k := \frac{1}{T_k}\sum_{t=1}^{T_k} U_i(\alpha^s_{i,t}, \alpha^s_{-i,t}, s_r).$$

Since each epoch uses a fresh independent instantiation, $\hat{R}^s_k$ and $\hat{R}^s_{k'}$ are independent across epochs $k \ne k'$, recovering the independence across pulls that the bandit framework requires. Within-epoch correlations are absorbed into the single observation $\hat{R}^s_k$. Since $U_i(\cdot,\cdot,s) \in [0,1]$ by assumption, $\hat{R}^s_k \in [0,1]$, and any bounded random variable is sub-Gaussian.

The epoch rewards are not identically distributed. Applied over an epoch of length T_k, the hypothesis of the theorem (with $s' = s_r$) gives

$$\left|\mathbb{E}[\hat{R}^s_k] - R^s\right| \;\le\; \frac{1}{T_k}\mathop{\mathbb{E}}_{\mathcal{ALG}(s)}\left[\sum_{t=1}^{T_k}\left|U_i(\alpha^s_{i,t}, \alpha^s_{-i,t}, s_r) - R^s\right|\right] \;=\; \frac{o(T_k)}{T_k},$$

which vanishes as the epoch index k grows. This is a property of $\mathcal{ALG}$ on the instance, independent of the horizon T; hence there is a constant k_0, entering the analysis only, with per-epoch bias at most $\Delta_{\min}/8$ for all $k \ge k_0$ and all $s \in \mathcal{S}$.

Step 3: Successive Elimination over States. We now apply the Successive Elimination algorithm of [7] with arms $\mathcal{S}$. Recall that at each epoch k, arm s produces the independent observation $\hat{R}^s_k \in [0,1]$. The empirical mean of arm s after t epochs is

$$\hat{p}^s_t := \frac{1}{t}\sum_{k=1}^{t} \hat{R}^s_k, \qquad \text{centered at } \mathbb{E}[\hat{p}^s_t] = \frac{1}{t}\sum_{k=1}^{t} \mathbb{E}[\hat{R}^s_k].$$

The algorithm maintains a surviving set $\mathcal{S}_t \subseteq \mathcal{S}$ and at each epoch t eliminates any arm s satisfying

$$\hat{p}^{\max}_t - \hat{p}^s_t \;\geq\; 2\alpha_t, \qquad \alpha_t = \sqrt{\frac{\ln(c|\mathcal{S}|t^2/\delta)}{t}},$$

where $\hat{p}^{\max}_t = \max_{s\in\mathcal{S}_t} \hat{p}^s_t$, $c > 4$ is an absolute constant, and $\delta = 1/T$.

The probabilistic engine of Theorem 3 of [7] is the tail bound $\Pr\big[|\hat{p}^s_t - \mathbb{E}[\hat{p}^s_t]| \geq \alpha_t\big] \leq e^{-\alpha_t^2 t} \leq \delta/(c|\mathcal{S}|t^2)$ with a union bound over epochs and arms. This is Hoeffding's inequality, whose requirements are independence and sub-Gaussian summands, both established in Step 2, and which centers the empirical mean at the average of the summands' means; identical distribution is not among its hypotheses. Hence with probability at least $1-\delta$,

$$|\hat{p}^s_t - \mathbb{E}[\hat{p}^s_t]| \leq \alpha_t \qquad \text{for all } s \in \mathcal{S} \text{ and all } t \geq 1;$$

call this event E.

What must be supplied is the mean control: that $\mathbb{E}[\hat{p}^s_t]$ is close enough to R^s. By Step 2,

$$\big|\mathbb{E}[\hat{p}^s_t] - R^s\big| \;\leq\; \frac{k_0}{t} + \frac{\Delta_{\min}}{8} \;\leq\; \frac{\Delta_{\min}}{4} \qquad \text{once } t \geq 8k_0/\Delta_{\min}.$$

With this in hand, on E the confidence intervals cover R^s with radius $\alpha_t + \Delta_{\min}/4$: for $t \geq 8k_0/\Delta_{\min}$ the best arm is never eliminated by a suboptimal arm s, its empirical deficit being at most

$$\hat{p}^s_t - \hat{p}^{s^*}_t \;\leq\; -\Delta_s + \frac{\Delta_{\min}}{2} + 2\alpha_t \;<\; 2\alpha_t;$$

and a suboptimal arm s is eliminated once $4\alpha_t \leq \Delta_s - \Delta_{\min}/2$, for which $\alpha_t \leq \Delta_s/8$ suffices, i.e. within $K^* = O(\ln T/\Delta^2_{\min})$ epochs. Before $t \geq 8k_0/\Delta_{\min}$ no elimination can occur for $T \geq e^{2k_0/\Delta_{\min}}$: an elimination requires $2\alpha_t \leq 1$, hence $t \geq 4\ln T \geq 8k_0/\Delta_{\min}$. An optimal arm may eliminate another optimal arm; this is harmless, and every surviving set contains an optimal arm.

The accounting holds at any finite T: the initial epochs before the bias settles cost $O(1)$ rounds; the elimination phase costs

$$N_{\text{elim}} \;\leq\; |\mathcal{S}| \sum_{k \leq K^*} T_k \;\leq\; |\mathcal{S}|\, K^{*2} \;=\; o(T)$$

rounds; and the failure event E^C contributes at most $\delta T = 1$.

Step 4: Bounding the Cumulative Suboptimality. Split the horizon at the end of epoch K^* and decompose

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathcal{ALG}'}\left[\sum_{t=1}^{T}\left|U_i(\alpha^{s_t}_{i,t}, \alpha^{s_t}_{-i,t}, s_r) - \max_{s\in\mathcal{S}} U_i(\bar{\alpha}^s_i, \bar{\alpha}^s_{-i}, s_r)\right|\right] \\
&\leq \mathop{\mathbb{E}}_{\mathcal{ALG}'}\left[\sum_{t=1}^{T} |\cdots| \;\middle|\; E\right] + \mathop{\mathbb{E}}_{\mathcal{ALG}'}\left[\sum_{t=1}^{T} |\cdots| \;\middle|\; E^C\right] \cdot \mathbb{P}(E^C).
\end{aligned}$$

Contribution from E^C. Since $U_i \in [0,1]$, each summand is bounded by 1, so this term is at most $T \cdot \mathbb{P}(E^C) \leq T\delta = 1$.

Rounds up to epoch K^*, on E. These number at most $N_{\text{elim}} = o(T)$, each contributing at most 1.

Rounds after epoch K^*, on E. By Step 3 every surviving arm s satisfies $R^s = \max_{s'\in\mathcal{S}} R^{s'}$. If a single arm survives, the algorithm commits to it and, by the hypothesis of the theorem applied at that s,

$$\mathop{\mathbb{E}}_{\mathcal{ALG}(s)}\left[\sum_{t}\left|U_i(\alpha^s_{i,t}, \alpha^s_{-i,t}, s_r) - R^s\right|\right] \in o(T),$$

so the remaining rounds contribute $o(T)$. If several optimal arms survive, the epochs continue on them; each epoch on an optimal arm contributes $o(T_k)$ by the same hypothesis, uniformly over the finitely many arms, so these epochs contribute $o(T)$ in total.

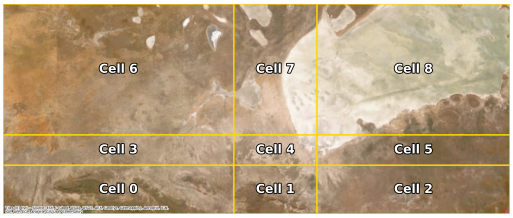
Combining the three contributions, the total cumulative suboptimality of $\mathcal{ALG}'$ satisfies

$$\mathop{\mathbb{E}}_{\mathcal{ALG}'}\left[\sum_{t=1}^{T}\left|U_i(\alpha_{i,t}^{s_t},\alpha_{-i,t}^{s_t},s_r)-\max_{s\in\mathcal{S}}U_i(\bar{\alpha}_i^{s},\bar{\alpha}_{-i}^{s},s_r)\right|\right]\ \in\ o(T).$$

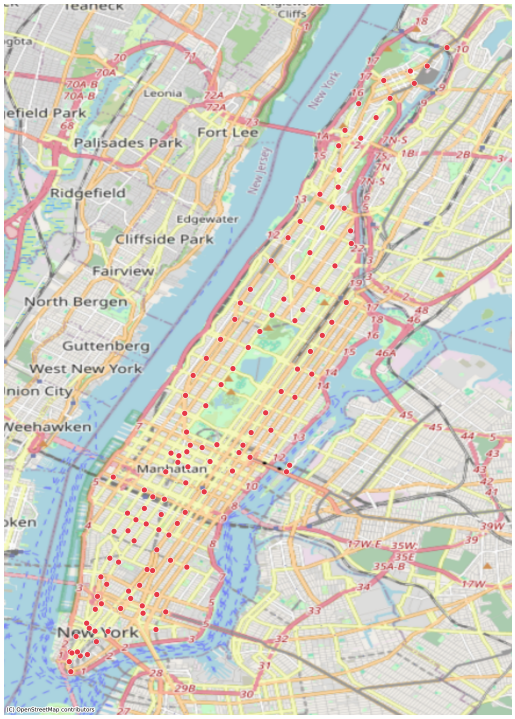
□

B Experimental Details

Code. The code for all experiments, together with the per-seed results behind Table 1 and Figure 2, is available at `https://github.com/thandina/strategic-decision-focused-learning`. The datasets are not redistributed; the repository documents how to obtain them.



(a) Etosha National Park with adaptive patrol-grid overlay [9, 6].



(b) 123 Manhattan subway stations used as targets [19, 22].

Figure 4: Geographic visualizations of the two game environments. (a) Etosha National Park with patrol-grid overlay (Conservation Security Game). (b) 123 Manhattan subway stations used as targets (Infrastructure Security Game).

Conservation Security Game. The cell values record elephant counts, drawn from a study tracking 15 African elephants within Etosha National Park published on the Movebank repository [9], covering movements from 2008 to 2014. The patrol-grid overlay partitions the reserve into nine cells whose boundaries are calibrated to elephant population density and its temporal variation: regions of high and variable activity receive finer-grained cells, while persistently low-activity regions are consolidated into coarser ones. This was to ensure that the experimental setup reflects where the distribution of patrol resources would be most meaningful. We construct 396 game instances spanning February 2013 to March 2014, one per evaluation day. We average the utilities across all instances for both ETE and PTO approaches and compare across training runs for models allowed to train up until a different number of epochs.

Infrastructure Security Game. The cell values record subway ridership, drawn from New York Metropolitan Transit Authority records [19], covering 123 Manhattan stations from 1 January 2022 to 31 December 2024. We construct 1098 game instances by averaging observations over 8-hour windows (approximately 3 instances per day). As with the conservation game, we average utilities across all instances for both approaches and compare across training epoch budgets.

Feature Engineering. For each observation, we extract temporal features capturing details about the period in which the observation occurs, e.g., the month, day of week. We additionally construct W autoregressive lag features per cell, where the i-th lag corresponds to the value observed at that cell i time periods prior. The full feature vector for each observation is thus $\mathbf{x} = (\text{temporal features}, v_{t-1}, v_{t-2}, \ldots, v_{t-W}) \in \mathbb{R}^d$, where v_{t-i} denotes the i-th lagged value and d is the total feature dimension.

Train-Validation-Test Split. We employ a strict temporal split to prevent data leakage. For the infrastructure game, all observations from 2024 are held out as the test set, while the pre-2024 data is partitioned into training and validation sets. For the conservation game, we analogously reserve data from February 24 2013 up until March 28 2014 for final temporal segment for testing. In both cases, we identify the set of unique time periods in the training pool and randomly assign 15% of these time periods—rather than individual rows—to the validation set. This ensures that all cells observed at a given time period fall entirely within either the training or validation fold, preventing the model from exploiting contemporaneous cross-cell information during validation.

Architecture. The Base MLP consists of two fully connected hidden layers. The input is projected to a hidden dimension of $h = 256$, followed by Layer Normalization, a SiLU activation, and dropout with probability $p = 0.2$. The second hidden layer reduces the dimension to $h/2 = 128$ with a SiLU activation, and a final linear layer projects to a single scalar output. A ReLU activation is applied at the output to enforce non-negativity, reflecting the physical constraint that state values cannot be negative. This architecture is shared across both domains. For the Conservation Security Game, the input encodes spatial cell features; for the Infrastructure Security Game, the input consists of 132 features per station — six lagged 8-hour ridership windows and a 123-dimensional station one-hot encoding — and the model is applied independently per station to predict the next ridership value.

Training Procedure. The supervised loss is the Huber loss, which is robust to outlier values compared with mean squared error. Optimization uses AdamW with weight decay 0.01 and a ReduceLROnPlateau schedule that halves the learning rate after 5 epochs without improvement in validation Huber loss. The initial learning rate is $\eta = 10^{-3}$ (Conservation) or $\eta = 0.01$ (Infrastructure) for PTO and ZOC, and $\eta = 5 \times 10^{-4}$ for KKT and Hybrid in both domains. Gaussian input noise $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$ with $\sigma = 0.01$ is applied during the supervised updates of PTO, ZOC and Hybrid; KKT uses no input noise, and KKT and Hybrid clip gradient norms at 1.0. ZOC and Hybrid use regimes of $R = 50$ epochs. All models are trained for 500 epochs over 5 independent seeds. For the Infrastructure Security Game, the KKT and Hybrid models anneal the Huber regularization weight from $\lambda_{\max} = 5.0$ to $\lambda_{\min} = 3.0$ linearly over 300 epochs (for Hybrid, counted from the end of its first supervised regime), with strategic gradient steps restricted to a top-$k = 20$ station subset and up to 32 time windows sampled per epoch. In the Conservation Security Game we do not use Huber regularization in the strategic loss ($\lambda_{\max} = \lambda_{\min} = 0$), and up to 128 time windows are sampled per epoch.

Evaluation. We evaluate along both statistical and strategic axes. On the statistical side, we report *Mean Absolute Error* (MAE) between predicted and actual values across all cell-time pairs in the test set. On the strategic side, the predicted state vector $\hat{s}$ is used to construct the payoff matrices of the security game, from which a Strong Stackelberg Equilibrium (SSE) defender strategy is computed via linear programming and evaluated against the *actual* state s to obtain the realized defender utility. This applies to both domains. For the Infrastructure Security Game, the test set comprises 1,098 8-hour windows across 123 stations; we report *per-window utility* (total utility $/1098$) as the primary metric.

Compute. All experiments were run on a single GPU; full runs (5 seeds, 500 epochs) take approximately 1–2 hours per domain for PTO and ZOC. For the Infrastructure Security Game, KKT and

Hybrid runs take approximately 3–4 hours due to per-epoch LP gradient calls via `cvxpylayers`. CPU-only execution is supported but substantially slower for the differentiable LP methods.

Controlled Experiments

B.1 Synthetic Conservation Security Game

Setup. We design a synthetic security game ($N = 100$ cells, catch penalty $P = 10$) to explore *when* each training objective succeeds or fails, rather than to produce aggregate rankings. The exogenous state is an animal density map (non-negative, peak cell normalised to 10). The defender receives $n_{\text{obs}} = 5$ independent sparse noisy sweeps, each visiting 20 of 100 cells uniformly at random with Poisson-corrupted counts. All four methods share the same four-layer CNN (convolutional with instance normalisation, SiLU activations, Softplus output) trained with AdamW on 1,000 synthetic scenarios for 200 epochs.

Training objectives. PTO minimises Huber loss on predicted density and retains the final epoch. **ZOC** (Zero-Order Checkpointing) trains identically but evaluates downstream strategic utility on a held-out 15% of training data every 20 epochs, retaining the checkpoint that maximises that score. **KKT** directly maximises defender utility by backpropagating through the Stackelberg LP via `cvxpylayers`: the attacker's best-response target t^* is found by binary search (detached from the graph), the LP is re-solved for t^* with gradients on, and exact KKT sensitivities flow back to model parameters. **Hybrid** alternates between 20-epoch Huber regimes and 20-epoch KKT regimes, reverting to the running-best checkpoint after each regime if strategic utility does not improve.

Density states are drawn from three families: (i) *Gaussian blobs* — 2–4 randomly placed modes ($\sigma \in [0.05, 0.25]$); (ii) *sparse spikes* — 1–6 isolated cells with non-zero value; and (iii) *concentric rings*. Crucially, the three families induce structurally different attacker best-response landscapes, letting us study regime-dependent behaviour rather than a single aggregate score. We present the qualitative observations below:

When does Hybrid help? Hybrid is most effective on *spatially extended multi-modal* densities (Figure 5, top row), where it outperforms both ZOC and KKT alone by up to 0.8 utility units. The revert-to-best filter consistently accepts the KKT phase on these instances.

When does KKT alone win? On *sparse spike* densities (Figure 5, second row), KKT alone achieves the highest utility. Hybrid's revert-to-best filter which leads to rejection of some KKT updates, results in sub-optimality for this particular setting.

When does PTO or ZOC win? On densities with *two well-separated modes of unequal weight* (Figure 5, third row), accuracy ends up being a very important part of determining the most dense mode. This plays an outsized role in shaping utility.

Summary. Hybrid is the natural default for multi-modal, spatially extended densities. KKT alone dominates when the density is sparse and the best-response target is unambiguous; it can fail when the landscape is smooth or multi-modal. PTO and ZOC remain competitive when predictive accuracy and strategic utility are well-aligned, most commonly for single-dominant-mode distributions. Designing a switching criterion that conditions Hybrid's acceptance step on the *stability of* t^* rather than a scalar utility threshold is a key direction for future work.

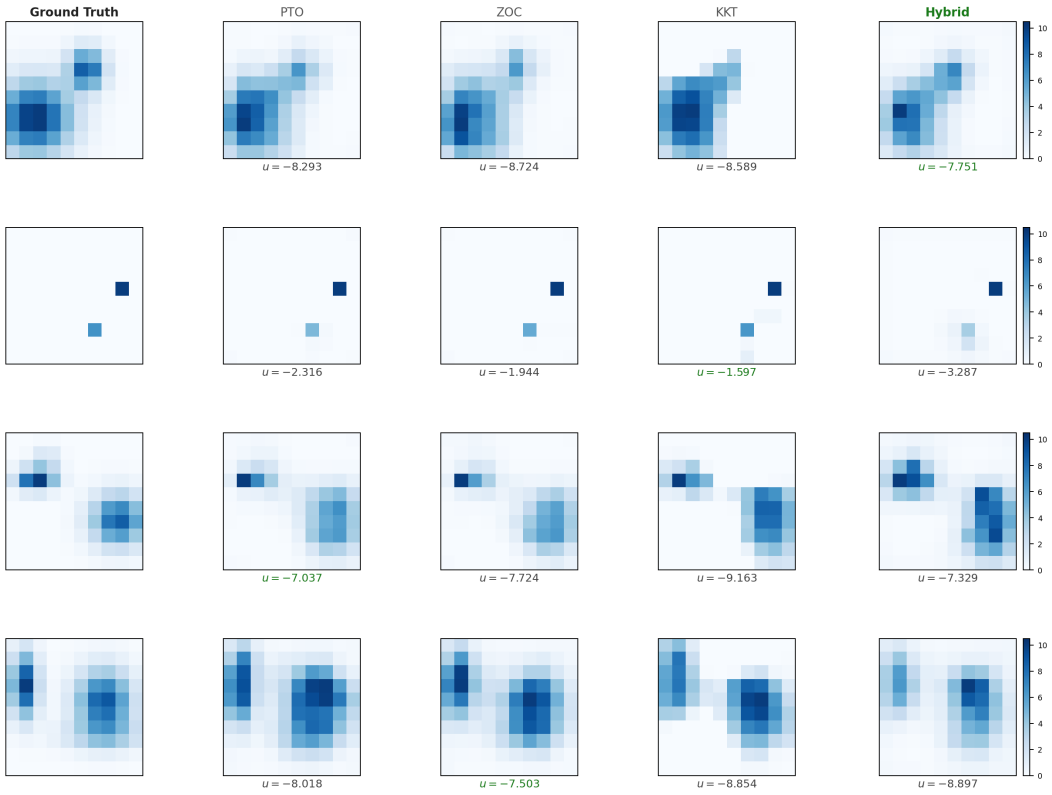
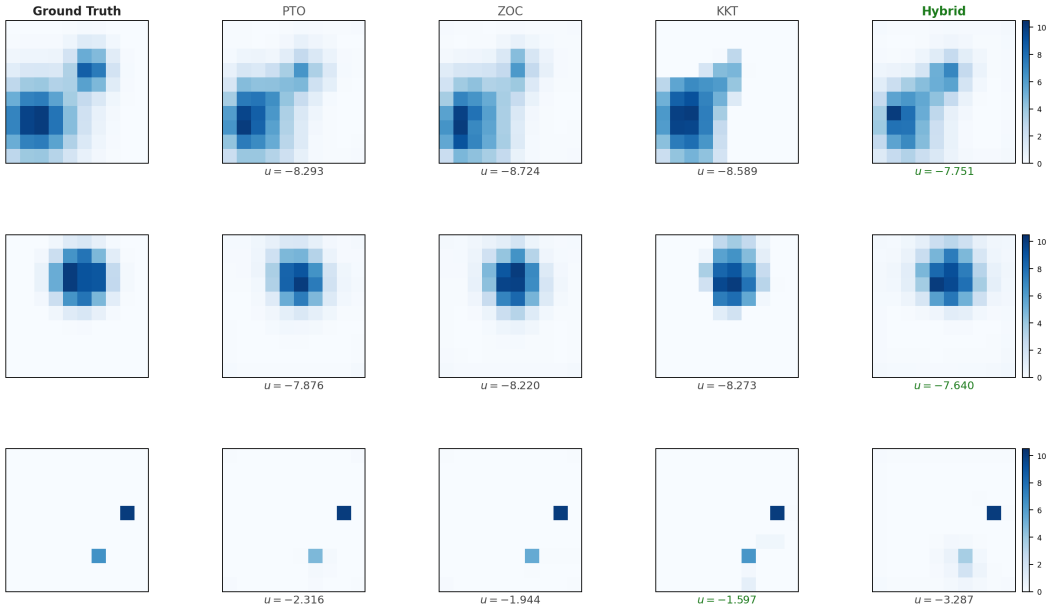
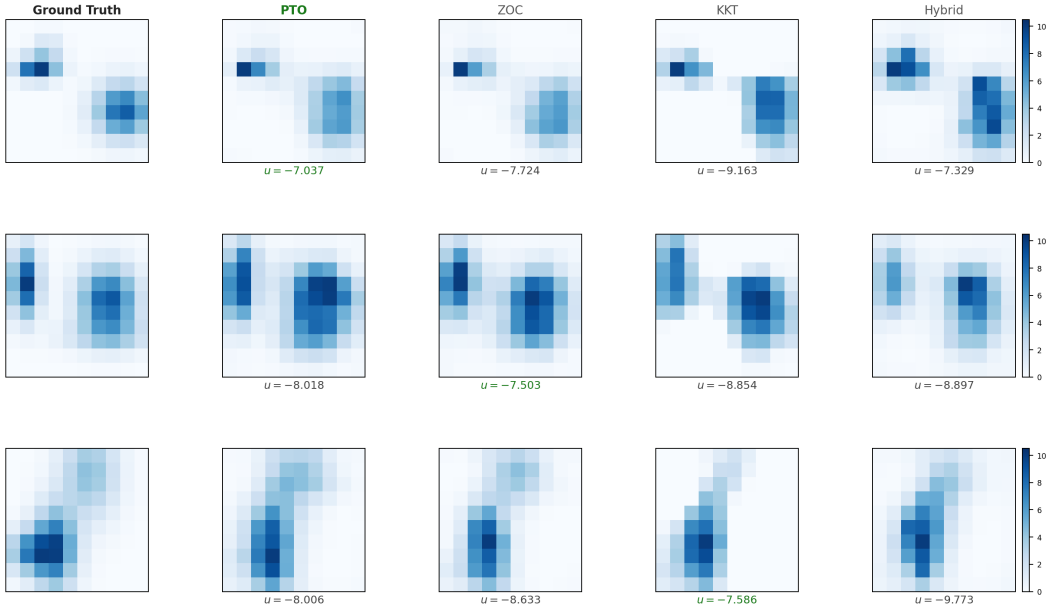


Figure 5: Four qualitative regimes on the synthetic 10×10 security game. Each row is one test scenario; columns show the ground-truth density and the prediction of each method. The winning method (highest defender utility against the true density) is labelled in **bold green**; utility u is reported below each prediction. **Row 1** (*multi-modal blob*): **Hybrid** wins. **Row 2** (*sparse spike*): **KKT** wins . **Row 3** (*two separated modes*): **PTO** wins. **Row 4** (*single blob with secondary cluster*): **ZOC** wins.



(a) Two scenarios where **Hybrid** wins (rows 1–2) alongside a scenario where **KKT** wins (row 3).



(b) **Row 1**: PTO wins. **Row 2**: ZOC wins. **Row 3**: KKT wins.

Figure 6: Scenario-level breakdown illustrating the conditions under which Hybrid, KKT, PTO, and ZOC each succeed or fail. Aggregate metrics obscure these structural differences; the regime panels above make them explicit.

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 paper does reflect the claims made in the abstract and introduction

   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: Paper discusses limitations of the work as well as future directions.

   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: For all theoretical results assumptions are state and proofs are provided.

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: We include our code and information needed to reproduce the results that we obtain.

   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 a copy of the data and code to reproduce the experimental results.

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: We include information on the experimental settings to understand the results.

   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: We include the statistical significance measures.

   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: Yes we include information on computer resources needed to reproduce experiments.

   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: This work conforms with the 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: We include a statement on the societal impacts of the work performed.

    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: Our models do not fit this category.

    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: Data and models used are credited in the paper

    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: [N/A]

    Justification: Paper does not release new assets.

    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: Human subjects or crowdsourcing was not used in this paper.

    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: Paper does not involve crowdsourcing or 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: LLMs are not an important original, non standard component of this research.

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.