Decision-Focused Learning of Adversary Behavior in Security Games

Submission #20

Abstract

Stackelberg security games are a critical tool for maximizing the utility of limited defense resources to protect important targets from an intelligent adversary. Motivated by green security, where the defender may only observe an adversary's response to defense on a limited set of targets, we study the problem of defending against the same adversary on a larger set of targets from the same distribution. We give a theoretical justification for why standard two-stage learning approaches, where a model of the adversary is trained for predictive accuracy and then optimized against, may fail to maximize the defender's expected utility in this setting. We develop a decision-focused learning approach, where the adversary behavior model is optimized for *decision* quality.

1 Introduction

Many real-world settings call for allocating limited defender resources against a strategic adversary, such as protecting public infrastructure [Tambe, 2011], transportation networks [Okamoto *et al.*, 2012], large public events [Yin *et al.*, 2014], urban crime [Zhang *et al.*, 2015], and green security [Fang *et al.*, 2015]. *Stackelberg security games (SSGs)* are a critical framework for computing defender strategies that maximize expected defender utility to protect important targets from an intelligent adversary [Tambe, 2011].

In many SSG settings, the adversary's utility function is not known a priori. In domains where there are many interactions with the adversary, the history of interactions can be leveraged to construct an *adversary behavior model*: a mapping from target features to values [Kar *et al.*, 2016]. An example of such a domain is protecting wildlife from poaching [Fang *et al.*, 2015]. The adversary's behavior is observable because snares are left behind, which rangers aim to remove (Fig. 1). Various features such as animal counts, distance to the edge of the park, weather and time of day may affect how attractive a particular target is to the adversary.

We focus on the problem of learning adversary models that generalize well: the training data consists of adversary behavior in the context of particular sets of targets, and we wish to achieve a high defender utility in the situation where we are playing against the same adversary and new sets of targets. In problem of poaching prevention, rangers patrol a small portion of the park each day and aim to predict poacher behavior across a large park consisting of targets with novel feature values [Gholami *et al.*, 2018].

Figure 1: Snares removed by rangers in Srepok National Park, Cambodia.

The standard approach to this problem [Nguyen *et al.*, 2013; Yang *et al.*, 2011; Kar *et al.*, 2016] breaks the problem into two stages. In the first, the adversary model is fit to the historical data using a standard machine learning loss function, such as mean squared error. In the second, the defender optimizes her allocation of defense resources against the model of adversary behavior learned in the first stage. Extensive research has focused on the first, predictive stage: developing better models of human behavior [Cui and John, 2014; Abbasi *et al.*, 2016]. We show that models that provide better predictions may not improve the defender's true objective: higher expected utility. This was observed previously by Ford et al. [2015] in the context of network security games, motivating our approach.

We propose a decision-focused approach to adversary modeling in SSGs which directly trains the predictive model to maximize defender expected utility on the historical data. Our approach builds on a recently proposed framework (outside of security games) called decision-focused learning, which aims to optimize the quality of the decisions induced by the predictive model, instead of focusing solely on predictive accuracy [Wilder *et al.*, 2019]; Fig. 2 illustrates our approach vs. a standard two-stage method. The main idea is to integrate a solver for the defender's equilibrium strategy into the loop of machine learning training and update the model to improve the decisions output by the solver.

While decision-focused learning has recently been explored in other domains (see related work), we overcome two main challenges to extend it to SSGs. First, the defender optimization problem is typically nonconvex, whereas previous work has focused on convex problems. Second, decision-focused learning requires counterfactual data—we need to know what our decision outcome quality would have been,

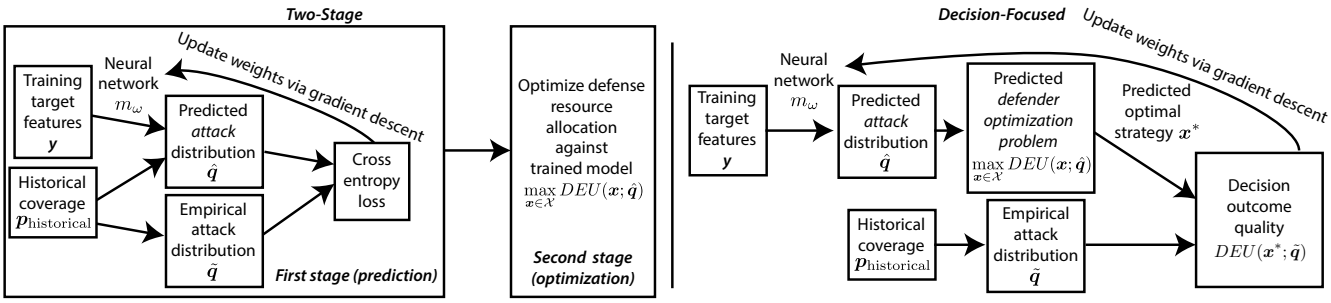

Figure 2: Comparison between a standard two-stage approach to training an adversary model and our decision-focused approach.

had we taken a different action than the one observed in training. By contrast, in SSGs we typically only observe the attacker's response to a fixed historical mixed strategy.

Our contributions are: *First*, we provide a theoretical justification for why decision-focused approaches can outperform two-stage approaches in SSGs. *Second*, we develop a decision-focused learning approach to adversary modeling in SSGs, showing both how to differentiate through general nonconvex problems as well as estimate counterfactual utilities for subjective utility quantal response [Nguyen *et al.*, 2013] and related adversary models.[1]

Related Work. Sinha et al. [2016] and Haghtalab et al. [2016] study the sample complexity (i.e., the number of attacks required) of learning an adversary model. Our setting differs from theirs because their defender observes attacks on the same target set that their defense performance is evaluated on. Ling et al. [2018; 2019] use a differentiable QR equilibrium solver to reconstruct the payoffs of both players from play. This differs from our objective of maximizing the defender's expected utility.

Outside of SSGs, Hartford et al. [2016] and Wright and Leyton-Brown [2017] study the problem of predicting play in unseen games assuming that all payoffs are fully observable; in our case, the defender seeks to maximize expected utility and does not observe the attacker's payoffs. Hartford et al. [2016] is the only other work to apply deep learning to modeling boundedly rational players in games.

Wilder et al. [2019] and Donti et al. [2017] study decision-focused learning for discrete and convex optimization, respectively. Donti et al. use sequential quadratic programming to solve a convex non-quadratic objective and use the last program to calculate derivatives. Here we propose an approach that works for the broader family of nonconvex functions.

2 Setting

We consider the problem of learning to play against an attacker with an unknown attack function $\boldsymbol{q}$. We observe attacks made by the adversary against sets of targets with differing features, and our goal is to generalize to new sets of targets with unseen feature values.

[1]We perform an experimental validation in a longer version of the paper which can be found at https://www.dropbox.com/s/cxvedgwx2dqdk9s/endtoendssg.pdf

Formally, let $\langle \boldsymbol{q}, C_d, D_{\text{train}}, D_{\text{test}} \rangle$ be an instance of a *Stackelberg security game with latent attack function (SSG-LA)*. $\boldsymbol{q}$, which is not observed by the defender, is the true mapping from the features and coverage of each target to the probability that the attacker will attack that target. C_d is the set of constraints that a mixed strategy defense must satisfy for the defender. D_{train} are *training games* of the form $\langle \mathcal{T}, \boldsymbol{y}, \mathcal{A}, \boldsymbol{u}_d, \boldsymbol{p}_{\text{historical}} \rangle$, where $\mathcal{T}$ is the set of targets, and $\boldsymbol{y}$, $\mathcal{A}$, $\boldsymbol{u}_d$ and $\boldsymbol{p}_{\text{historical}}$ are the features, observed attacks, defender's utility function, and historical coverage probabilities, respectively, for each target $i \in \mathcal{T}$. D_{test} are *test games* $\langle \mathcal{T}, \boldsymbol{y}, \boldsymbol{u}_d \rangle$, each containing a set of targets and the associated features and defender values for each target. We assume that all games are drawn i.i.d. from the same distribution. In a green security setting, the training games represent the results of patrols on limited areas of the park and the test games represent the entire park.

The defender's goal is to select a coverage function $\boldsymbol{x}$ that takes the parameters of each test game as input and maximizes her expected utility across the test games against the attacker's true $\boldsymbol{q}$:

$$\max_{\boldsymbol{x} \text{ satisfying } C_d} \mathop{\mathbb{E}}_{\langle \mathcal{T}, \boldsymbol{y}, \boldsymbol{u}_d \rangle \sim D_{\text{test}}} \left[DEU(\boldsymbol{x}(\mathcal{T}, \boldsymbol{y}, \boldsymbol{u}_d); \boldsymbol{q}) \right]. \quad (1)$$

To achieve this, she can observe the attacker's behavior in the training data and learn how he values different combinations of features. We now explore two approaches to the learning problem: the standard two-stage approach taken by previous work and our proposed decision-focused approach.

Two-Stage Approach. A standard two-stage approach to the defender's problem is to estimate the attacker's $\boldsymbol{q}$ function from the training data and optimize against the estimate during testing. This process resembles multiclass classification where the targets are the classes: the inputs are the target features and historical coverages, and the output is a distribution over the predicted attack. Specifically, the defender fits a function $\hat{\boldsymbol{q}}$ to the training data that minimizes a loss function. Using the cross entropy, the loss for a particular training example is

$$\mathcal{L}(\hat{q}(\boldsymbol{y}, \boldsymbol{p}_{\text{historical}}), \mathcal{A}) = -\sum_{i \in T} \tilde{\boldsymbol{q}} \log(\hat{\boldsymbol{q}}_i(\boldsymbol{y}, \boldsymbol{p}_{\text{historical}})), \quad (2)$$

where $\tilde{\boldsymbol{q}} = \frac{\mathcal{A}_i}{|\mathcal{A}|}$ is the *empirical attack distribution* and $\mathcal{A}_i$ is the number of historical attacks that were observed on target i. Note that we use hats to indicate model outputs and tildes

to indicate the ground truth. For each test game $\langle \mathcal{T}, \boldsymbol{y}, \boldsymbol{u}_d \rangle$, coverage is selected by maximizing the defender's expected utility assuming the attack function is $\hat{\boldsymbol{q}}$:

$$\max_{\boldsymbol{x} \text{ satisfying } C_d} DEU(\boldsymbol{x}(\mathcal{T}, \boldsymbol{y}, \boldsymbol{u}_d); \hat{\boldsymbol{q}}). \tag{3}$$

Decision-Focused Learning. The standard approach may fall short when the loss function (e.g., cross entropy) does not align with the true goal of maximizing expected utility. Ultimately, the defender just wants $\hat{\boldsymbol{q}}$ to induce the correct mixed strategy, regardless of how accurate it is in a general sense. The idea behind our decision-focused learning approach is to directly train $\hat{\boldsymbol{q}}$ to maximize defender utility. Define

$$\boldsymbol{x}^*(\hat{\boldsymbol{q}}) = \operatorname*{arg\,max}_{\boldsymbol{x} \text{ satisfying } C_d} DEU(\boldsymbol{x}; \hat{\boldsymbol{q}}) \tag{4}$$

to be the optimal defender coverage function given attack function $\hat{\boldsymbol{q}}$. Ideally, we would find a $\hat{\boldsymbol{q}}$ which maximizes

$$DEU(\hat{\boldsymbol{q}}) = \mathop{\mathbb{E}}_{\langle \mathcal{T}, \boldsymbol{y}, \boldsymbol{u}_d \rangle \sim D_{\text{test}}} \left[DEU(\boldsymbol{x}^*(\hat{\boldsymbol{q}}); \boldsymbol{q}) \right]. \tag{5}$$

This is just the defender's expected utility on the test games when she plans her mixed strategy defense based on attack function $\hat{\boldsymbol{q}}$ but the true function is $\boldsymbol{q}$. While we do not have access to D_{test}, we can estimate Eq. 5 using samples from D_{train} (taking the usual precaution of controlling model complexity to avoid overfitting). The idea behind decision-focused learning is to directly optimize Eq. 5 on the training data instead of using an intermediate loss function such as cross entropy. Minimizing Eq. 5 on the training set via gradient descent requires the gradient, which we can derive using the chain rule:

$$\frac{\partial DEU(\hat{\boldsymbol{q}})}{\partial \hat{\boldsymbol{q}}} = \mathop{\mathbb{E}}_{\langle \mathcal{T}, \boldsymbol{y}, \boldsymbol{u}_d \rangle \sim D_{\text{train}}} \left[\frac{\partial DEU(\boldsymbol{x}^*(\hat{\boldsymbol{q}}); \boldsymbol{q})}{\partial \boldsymbol{x}^*(\hat{\boldsymbol{q}})} \frac{\partial \boldsymbol{x}^*(\hat{\boldsymbol{q}})}{\partial \hat{\boldsymbol{q}}} \right].$$

Here, $\frac{\partial DEU(\boldsymbol{x}^*(\hat{\boldsymbol{q}}); \boldsymbol{q})}{\partial \boldsymbol{x}^*(\hat{\boldsymbol{q}})}$ describes how the defender's true utility with respect to $\boldsymbol{q}$ changes as a function of her strategy $\boldsymbol{x}^*$. $\frac{\partial \boldsymbol{x}^*(\hat{\boldsymbol{q}})}{\partial \hat{\boldsymbol{q}}}$ describes how $\boldsymbol{x}^*$ depends on the estimated attack function $\hat{\boldsymbol{q}}$, which requires differentiating through the optimization problem in Eq. 4. Suppose that we have a means to calculate both terms. Then we can estimate $\frac{\partial DEU(\hat{\boldsymbol{q}})}{\partial \hat{\boldsymbol{q}}}$ by sampling example games from D_{train} and computing gradients on the samples. If $\hat{\boldsymbol{q}}$ is itself implemented in a differentiable manner (e.g., a neural network), this allows us to train the entire system end-to-end via gradient descent. Previous work has explored decision-focused learning in other contexts [Donti *et al.*, 2017; Wilder *et al.*, 2019], but SSGs pose unique challenges that complicate the process of computing both of the required terms above. In Sec. 4, we explore these challenges and propose solutions.

3 Impact of Two-Stage Learning on DEU

We demonstrate that, for natural two-stage training loss functions, decreasing the loss may not lead to increasing the DEU. This indicates that we may be able to improve decision quality by making use of decision-focused learning because a decision-focused approach uses the decision objective as the loss. Thus, reducing the loss function increases the DEU in decision-focused learning.

We begin with a simple case: two-target games with a rational attacker and zero-sum utilities.[2]

Theorem 1. *Consider a two-target SSG with a rational attacker, zero-sum utilities, and a single defense resource to allocate, which is not subject to scheduling constraints (i.e., any nonnegative marginal coverage that sums to one is feasible). Let $z_0 \geq z_1$ be the attacker's values for the targets, which are observed by the attacker, but not the defender, and we assume w.l.o.g. are non-negative and sum to 1.*

The defender has an estimate of the attacker's values $(\hat{z}_0, \hat{z}_1)$ with mean squared error (MSE) ϵ^2. *Suppose the defender optimizes coverage against this estimate. If $\epsilon^2 \leq (1 - z_0)^2$, the ratio between the highest DEU under the estimate of $(\hat{z}_0, \hat{z}_1)$ with MSE ϵ^2 and the lowest DEU is:*

$$\frac{(1 - (z_0 - \epsilon))z_0}{(1 - (z_1 - \epsilon))z_1}. \tag{6}$$

The reason for the gap in defender expected utilities is that the attacker attacks the target with value that is underestimated by $(\hat{z}_0, \hat{z}_1)$. This target has less coverage than it would have if the defender knew the attacker's utilities precisely, allowing the attacker to benefit. When the defender reduces the coverage on the larger value target, the attacker benefits more, causing the gap in expected defender utilities.

Note that because (6) is at least one (since DEU are negative), decreasing the MSE does not necessarily lead to higher DEU. For $\epsilon > \epsilon'$, the learned model at MSE=ϵ^2 will have higher DEU than the model at MSE=$(\epsilon')^2$ if the former underestimates the value of z_1, the latter underestimates the value of z_0 and ϵ, and ϵ' are sufficiently close. In decision-focused learning, the DEU is used as the loss directly—thus, a model with lower loss must have higher DEU.

In the case of Thm. 1, the defender can lose value $z_0\epsilon$, or ϵ as $z_0 \to 1$, compared to the optimum because of an unfavorable distribution of estimation error. We show that this carries over to a boundedly rational QR attacker, with the degree of loss converging towards the rational case as λ increases.

Theorem 2. *Consider the setting of Thm. 1, but in the case of a QR attacker. For any $0 \leq \alpha \leq 1$, if $\lambda \geq \frac{2}{(1-\alpha)\epsilon} \log \frac{1}{(1-\alpha)\epsilon}$, the defender's loss compared to the optimum may be as much as $\alpha(1 - \epsilon)\epsilon$ under a target value estimate with MSE ϵ^2.*

4 Decision-Focused Learning in SSGs with an SUQR Adversary

We now present our technical approach to decision-focused learning in SSGs. As discussed above, we use $DEU(\hat{\boldsymbol{q}})$, the expected utility induced by an estimate $\hat{\boldsymbol{q}}$, as the objective for training. The key idea is to embed the defender optimization problem into training and compute gradients of DEU with respect to the model's predictions. In order to do so, we need two quantities, each of which poses a unique challenge in the context of SSGs.

First, we need $\frac{\partial \boldsymbol{x}^*(\hat{\boldsymbol{q}})}{\partial \hat{\boldsymbol{q}}}$, which describes how the defender's strategy $\boldsymbol{x}^*$ depends on $\hat{\boldsymbol{q}}$. Computing this requires differentiating through the defender's optimization problem. Previous

[2] All proofs are in the supplemental material: https://www.dropbox.com/s/ts87mt46qdqibvn/endtoendSSG_supplement.pdf

work on differentiable optimization considers convex problems [Amos and Kolter, 2017]. However, typical bounded rationality models for $\hat{\boldsymbol{q}}$ (e.g., QR, SUQR, and SHARP) all induce *nonconvex* defender problems. We resolve this challenge by showing how to differentiate through the local optimum output by a black-box nonconvex solver.

Second, we need $\frac{\partial DEU(\boldsymbol{x}^*(\hat{\boldsymbol{q}}),\boldsymbol{q})}{\partial \boldsymbol{x}^*(\hat{\boldsymbol{q}})}$, which describes how the defender's *true* utility with respect to $\boldsymbol{q}$ depends on her strategy $\boldsymbol{x}^*$. Computing this term requires a *counterfactual* estimate of how the attacker would react to a different coverage vector than the historical one. Unfortunately, typical datasets only contain a set of sampled attacker responses to a particular historical defender mixed strategy. Previous work on decision-focused learning in other domains [Donti *et al.*, 2017; Wilder *et al.*, 2019] assumes that the historical data specifies the utility of *any* possible decision, but this assumption breaks down under the limited data available in SSGs. We show that common models like SUQR exhibit a crucial decomposition property that enables unbiased counterfactual estimates. We now explain both steps in more detail.

4.1 Decision-Focused Learning for Nonconvex Optimization

Under nonconvexity, all that we can (in general) hope for is a local optimum. Since there may be many local optima, it is unclear what it means to differentiate through the solution to the problem. We assume that we have black-box access to a nonconvex solver which outputs a fixed local optimum. We show that we can obtain derivatives of that particular optimum by differentiating through a convex quadratic approximation around the solver's output (since existing techniques apply to the quadratic approximation).

We prove that this procedure works for a wide range of nonconvex problems. Specifically, we consider the generic problem $\min_{\boldsymbol{x}\in\mathcal{X}} f(\boldsymbol{x},\theta)$ where f is a (potentially nonconvex) objective which depends on a learned parameter θ. $\mathcal{X}$ is a feasible set that is representable as $\{x : g_1(\boldsymbol{x}),\dots,g_m(\boldsymbol{x}) \leq 0, h_1(\boldsymbol{x}),\dots,h_\ell(\boldsymbol{x}) = 0\}$ for some convex functions $g_1,\dots,g_m$ and affine functions $h_1,\dots,h_\ell$. We assume there exists some $\boldsymbol{x} \in \mathcal{X}$ with $\boldsymbol{g}(\boldsymbol{x}) < 0$, where $\boldsymbol{g}$ is the vector of constraints. In SSGs, f is the defender objective DEU, θ is the attack function $\hat{\boldsymbol{q}}$, and $\mathcal{X}$ is the set of $\boldsymbol{x}$ satisfying C_d. We assume that f is twice continuously differentiable. These two assumptions capture smooth nonconvex problems over a nondegenerate convex feasible set.

Suppose that we can obtain a local optimum of f. Formally, we say that $\boldsymbol{x}$ is a *strict local minimizer* of f if (1) there exist $\boldsymbol{\mu} \in R^m_+$ and $\boldsymbol{\nu} \in R^\ell$ such that $\nabla_{\boldsymbol{x}} f(\boldsymbol{x},\theta) + \boldsymbol{\mu}^\top \nabla \boldsymbol{g}(\boldsymbol{x}) + \boldsymbol{\nu}^\top \nabla \boldsymbol{h}(\boldsymbol{x}) = 0$ and $\boldsymbol{\mu} \odot \boldsymbol{g}(\boldsymbol{x}) = 0$ and (2) $\nabla^2 f(\boldsymbol{x},\theta) \prec 0$. Intuitively, the first condition is first-order stationarity, where $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ are dual multipliers for the constraints, while the second condition says that the objective is strictly convex at $\boldsymbol{x}$ (i.e., we have a strict local minimum, not a plateau or saddle point). We prove the following:

Theorem 3. *Let $\boldsymbol{x}$ be a strict local minimizer of f over $\mathcal{X}$. Then, except on a measure zero set, there exists a convex set $\mathcal{I}$ around $\boldsymbol{x}$ such that $\boldsymbol{x}^*_{\mathcal{I}}(\theta) = \arg\min_{x\in\mathcal{I}\cap\mathcal{X}} f(\boldsymbol{x},\theta)$ is differentiable. The gradients of $\boldsymbol{x}^*_{\mathcal{I}}(\theta)$ with respect to θ are given by the gradients of solutions to the local quadratic approximation $\min_{\boldsymbol{x}\in\mathcal{X}} \frac{1}{2}\boldsymbol{x}^\top \nabla^2 f(\boldsymbol{x},\theta)\boldsymbol{x} + \boldsymbol{x}^\top \nabla f(\boldsymbol{x},\theta)$.*

This states that the local minimizer within the region output by the nonconvex solver varies smoothly with θ, and we can obtain gradients of it by applying existing techniques [Amos and Kolter, 2017] to the local quadratic approximation. It is easy to verify that the defender utility maximization problem for an SUQR attacker satisfies the assumptions of Theorem 3 since the objective is smooth and typical constraint sets for SSGs are polytopes with nonempty interior (see [Xu, 2016] for a list of examples). In fact, our approach is quite general and applies to a range of behavioral models such as QR, SUQR, and SHARP since the defender optimization problem remains smooth in all.

4.2 Counterfactual Adversary Estimates.

We now turn to the second challenge, that of estimating how well a different strategy would perform on the historical games. We focus here on the SUQR attacker, but the main idea extends more widely (as we discuss below). For SUQR, if the historical attractiveness values were known, then $\frac{\partial DEU}{\partial \boldsymbol{x}^*}$ could be easily computed in closed form. The difficulty is that we typically only observe samples from the attack distribution $\boldsymbol{q}$, where for SUQR, $\boldsymbol{q}_i \propto \exp(w\boldsymbol{p}_i + \phi(\boldsymbol{y}_i))$. $\phi(y_i)$ itself is not observed directly.

The crucial property enabling counterfactual estimates is that the attacker's behavior can be decomposed into his reaction to the defender's coverage ($w\boldsymbol{p}_i$) and the impact of target values ($\phi(y_i)$). Suppose that we know w and observe sampled attacks for a particular historical game. Because we can estimate $\boldsymbol{q}_i$ and the term $w\boldsymbol{p}_i$ is known, we can invert the $\exp$ function to obtain an estimate of $\phi(\boldsymbol{y}_i)$ (formally, this corresponds to the maximum likelihood estimator under the empirical attack distribution). Note that we do not know the *entire function* ϕ, only its value at $\boldsymbol{y}_i$, and that the inversion yields $\phi(\boldsymbol{y}_i)$ that is unique up to a constant additive factor. Having recovered $\phi(\boldsymbol{y}_i)$, we can then perform complete counterfactual reasoning for the defender on the historical games.

5 Conclusion

We present a decision-focused approach to adversary modeling in security games. We provide a theoretical justification as to why training an attacker model to maximize DEU can provide higher DEU than training the model to maximize predictive accuracy. We extend past work in decision-focused learning to smooth nonconvex objectives, accounting for the defender's optimization in SSGs against many attacker types, including SUQR.

We conclude that improving predictive accuracy does not guarantee increased DEU in SSGs. We believe this conclusion has important consequences for future research and that our decision-focused approach can be extended to a variety of SSG models where smooth nonconvex objectives and polytope feasible regions are common.

References

[Abbasi *et al.*, 2016] Yasaman Dehghani Abbasi, Noam Ben-Asher, Cleotilde Gonzalez, Don Morrison, Nicole

Sintov, and Milind Tambe. Adversaries wising up: Modeling heterogeneity and dynamics of behavior. In *Proc. of International Conference on Cognitive Modeling*, 2016.

[Amos and Kolter, 2017] Brandon Amos and J. Zico Kolter. Optnet: Differentiable optimization as a layer in neural networks. In *Proc. of ML-17*, 2017.

[Cui and John, 2014] Jinshu Cui and Richard S John. Empirical comparisons of descriptive multi-objective adversary models in stackelberg security games. In *Proc. of GameSec-14*, pages 309–318, 2014.

[Donti *et al.*, 2017] Priya Donti, Brandon Amos, and J. Zico Kolter. Task-based end-to-end model learning in stochastic optimization. In *Proc. of NIPS-17*, pages 5484–5494, 2017.

[Fang *et al.*, 2015] Fei Fang, Peter Stone, and Milind Tambe. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *Proc. of IJCAI-15*, pages 2589–2595, Buenos Aires, 2015.

[Ford *et al.*, 2015] Benjamin Ford, Thanh Nguyen, Milind Tambe, Nicole Sintov, and Francesco Delle Fave. Beware the soothsayer: From attack prediction accuracy to predictive reliability in security games. In *Proc. of GameSec-15*, pages 35–56, 2015.

[Gholami *et al.*, 2018] Shahrzad Gholami, Sara McCarthy, Bistra Dilkina, Andrew Plumptre, Milind Tambe, Margaret Driciru, Fred Wanyama, Aggrey Rwetsiba, Mustapha Nsubaga, Joshua Mabonga, et al. Adversary models account for imperfect crime data: Forecasting and planning against real-world poachers. In *Proc. of AAMAS-18*, pages 823–831, Stockholm, 2018.

[Haghtalab *et al.*, 2016] Nika Haghtalab, Fei Fang, Thanh Hong Nguyen, Arunesh Sinha, Ariel D Procaccia, and Milind Tambe. Three strategies to success: Learning adversary models in security games. In *Proc. of IJCAI-16*, pages 308–314, New York, 2016.

[Hartford *et al.*, 2016] Jason S. Hartford, James R. Wright, and Kevin Leyton-Brown. Deep learning for predicting human strategic behavior. In *Proc. of NIPS-16*, pages 2424–2432, 2016.

[Kar *et al.*, 2016] Debarun Kar, Fei Fang, Francesco M. Delle Fave, Nicole Sintov, Milind Tambe, and Arnaud Lyet. Comparing human behavior models in repeated stackelberg security games: An extended study. *Artificial Intelligence*, 240:65–103, 2016.

[Ling *et al.*, 2018] Chun Kai Ling, Fei Fang, and J. Zico Kolter. What game are we playing? end-to-end learning in normal and extensive form games. In *Proc. of IJCAI-18*, pages 396–402, Stockholm, 2018.

[Ling *et al.*, 2019] Chun Kai Ling, Fei Fang, and J. Zico Kolter. Large scale learning of agent rationality in two-player zero-sum games. In *Proc. of AAAI-19*, Honolulu, 2019.

[Nguyen *et al.*, 2013] Thanh Hong Nguyen, Rong Yang, Amos Azaria, Sarit Kraus, and Milind Tambe. Analyzing the effectiveness of adversary modeling in security games. In *Proc. of AAAI-13*, Bellevue, Washington, 2013.

[Okamoto *et al.*, 2012] Steven Okamoto, Noam Hazon, and Katia Sycara. Solving non-zero sum multiagent network flow security games with attack costs. In *Proc. of AAMAS-12*, pages 879–888. Valencia, 2012.

[Sinha *et al.*, 2016] Arunesh Sinha, Debarun Kar, and Milind Tambe. Learning adversary behavior in security games: A PAC model perspective. In *Proc. of AAMAS-16*, pages 214–222, Singapore, 2016.

[Tambe, 2011] Milind Tambe. *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge University Press, 2011.

[Wilder *et al.*, 2019] Bryan Wilder, Bistra Dilkina, and Milind Tambe. Melding the data-decisions pipeline: Decision-focused learning for combinatorial optimization. In *Proc. of AAAI-19*, Honolulu, 2019.

[Wright and Leyton-Brown, 2017] James R. Wright and Kevin Leyton-Brown. Predicting human behavior in unrepeated, simultaneous-move games. *Games and Economic Behavior*, 106:16–37, 2017.

[Xu, 2016] Haifeng Xu. The mysteries of security games: Equilibrium computation becomes combinatorial algorithm design. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 497–514. ACM, 2016.

[Yang *et al.*, 2011] Rong Yang, Christopher Kiekintveld, Fernando Ordonez, Milind Tambe, and Richard John. Improving resource allocation strategy against human adversaries in security games. In *Proc. of IJCAI-11*, 2011.

[Yin *et al.*, 2014] Yue Yin, Bo An, and Manish Jain. Game-theoretic resource allocation for protecting large public events. In *Proc. of AAAI-14*, 2014.

[Zhang *et al.*, 2015] Chao Zhang, Arunesh Sinha, and Milind Tambe. Keeping pace with criminals: Designing patrol allocation against adaptive opportunistic criminals. In *Proc. of AAMAS-15*, pages 1351–1359, Istanbul, 2015.