Robust Spatial-Temporal Incident Prediction

Ayan Mukhopadhyay
Stanford University
Palo Alto, USA

Kai Wang
CRCS*, Harvard University
Boston, USA

Andrew Perrault
CRCS, Harvard University
Boston, USA

Mykel Kochenderfer
Stanford University
Palo Alto, USA

Milind Tambe
CRCS, Harvard University
Boston, USA

Yevgeniy Vorobeychik
Washington University in St. Louis
St. Louis, USA

Abstract

Spatio-temporal criminal incident prediction is among the central issues in law enforcement, with applications ranging from predicting assaults and terrorist acts to predicting poaching. However, state of the art approaches fail to account for evasion in response to predictive models, a common form of which is spatial shift in incident occurrence. We present a novel and general approach for incident forecasting that is robust to such spatial shifts. We propose two techniques for solving the resulting robust optimization problem: first, a constraint generation method guaranteed to yield an optimal solution, and second, a more scalable gradient-based approach. We then apply these techniques for both discrete-time and continuous-time robust incident forecasting. We evaluate our algorithms on three different crime datasets, demonstrating that our approach is significantly more robust than conventional methods.

1 Introduction

The increase in availability of data and algorithmic progress has created new ways of fighting illegal activities like poaching, violent crimes, and terrorism. Predictive analytics and data-driven methods have been developed to understand where crimes could potentially happen, with crime prediction a major part of this literature [1, 2, 3, 4, 5, 6]. Mukhopadhyay et. al. [7] provide a comprehensive review of prior work in this domain. However, a significant limitation of existing crime prediction methods is that they do not account for changes in criminal behavior *in response* to the predictive models. Indeed, people with the malicious intent of committing crimes can potentially alter their behavior in response to proactive policing based on such static predictive models, effectively resulting in spatial shifts in crime incidents [8, 9, 10].

Our goal is to create an algorithmic framework for forecasting crimes that is robust to manipulations in criminal behavior. We seek to identify the vulnerabilities in crime prediction models, create an approach for capturing adversarial actions and finally create a robust prediction model against such actions. Instead of focusing on a specific model of incident arrival, we create a general approach that is flexible to accommodate both continuous-time and discrete-time prediction models. We only assume a convex likelihood function over incident arrival, and model the interaction between the learner and the attacker as a Stackelberg game. In our model, the learner chooses a patrol strategy and the attacker chooses to manipulate its behavior in response to the chosen strategy. Such a problem falls under the paradigm of adversarial learning, which studies the effect of adversarial influence on machine learning models [11, 12, 13]. Adversarial learning has been successfully used in many domains, and it has been used recently to combat implicit biases and imperfections in crime prediction methodologies [14]. However, to the best our knowledge, models specifically aimed to tackle manipulation in criminal behavior to respond to forecasting models have not been explored, and we aim to systematically bridge this gap.

The proposed game-theoretic model involving a learner and an attacker poses two major challenges: a) the resulting optimization problem is intrinsically difficult to solve due to the nested hierarchy of the attacker's and defender's optimization problems, and b) the set of adversarial strategies is combinatorial. We explain how these specific challenges manifest themselves, and develop techniques for addressing them.

Contributions: 1) A general Stackelberg game model

*The abbreviation CRCS stands for the Center for Research on Computation and Society

Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI), PMLR volume 124, 2020.

for robust incident prediction accounting for adversarial spatial crime shifts; 2) an approach based on dynamic constraint generation that computes an optimal leader strategy in the Stackelberg game; 3) a gradient-based algorithm that trades-off optimality and scalability; 4) application of the proposed approach to both discrete-time (Poisson regression and logistic regression) as well as continuous-time (survival analysis) incident prediction models; and finally, 5) evaluation of the proposed approach, demonstrating that it is significantly more robust to adversarial manipulation than conventional methods.

Broader Impact: There are several caveats to predictive policing. It has faced numerous ethical issues in the recent years [7]. Predictive policing can be used to measure possible environmental factors that are correlated with higher crime rates and also gauge the probability of a specific individual being an offender. This paper focuses only on the former. Even when environmental conditions are modeled, any approach that aims to diversify the geographic spread of police patrols increases contact of citizens with law-enforcement authorities. While this effect is a consequence of proactive policing in general [15], it is particularly true for approaches that seek to model robustness against possible spatial shifts. Proactive policing is a double-edged sword in this regard, and we only lay the algorithmic framework for robustness in incident prediction. We direct practitioners to research in criminology [15, 16] to carefully assess perception and expectations from such policies, as well as its effects. In this paper, we focus solely on the technical aspects of robustness in incident prediction methods.

2 Model

We consider a set of equally sized cells G, that spans the entire spatial area under consideration. Let $g_i \in G$ denote the ith cell. Suppose that a sequence of incidents occurs over this area, generated according to some unknown distribution, which is captured into a dataset of incidents, $D_{seq} = \{(t_1, \ell_1, w_1), (t_2, \ell_2, w_2), \dots, (t_n, \ell_n, w_n)\}$, where each incident d_i is identified by its time t_i, location l_i (mapping to a cell in G), and a vector of spatio-temporal features $w_i \in \mathbb{R}^m$, capturing, for example, weather, proximity to liquor stores, and any other potential determinants of crime. Henceforth, we use this dataset as a proxy for a future *baseline* distribution of incidents (i.e., distribution if we follow the specific policy that was implemented at the time the data was collected).

We assume that crime incidents are stochastic, and therefore associate incidence of these with a random variable X, the nature of which depends on the particular model of crime prediction. For example, in discrete-time models, X will capture the number of incidents over a fixed time interval, whereas in continuous-time models X will be the inter-arrival time between incidents. We use X to transform our dataset into an *input* dataset $D = \{(w_i, x_i)\}$ of incident features w_i and associated observation x_i (e.g., crime count).

Suppose that a crime prediction model entails a likelihood function $F(x;\theta,w)$ representing the likelihood of observation x given features w, with θ being the model parameters. We assume that F is convex in θ. A conventional approach to crime prediction is to learn parameters θ that maximize the likelihood of observed data D:

$$\theta^* \in \arg\max_\theta \prod_i F(x_i;\theta,w).$$

Typically, for computational convenience this is transformed into maximizing log-likelihood:

$$\max_\theta \sum_i \log F(x_i;\theta,w) \equiv \sum_i f(x_i;\theta,w), \qquad (1)$$

where $f(x_i;\theta,w) = \log F(x_i;\theta,w)$. Subsequently, we refer to f as both the (log)-likelihood function and the incident prediction model to streamline exposition; later, as we tackle the specific applications these will be distinct.

Attacker Model: Crime prediction is commonly a part of a broader strategy of crime response or prevention by law enforcement, with the typical net result that spatio-temporal patrols are more concentrated in the areas of higher criminal activity. This, in turn, incentivizes potential perpetrators of crimes, such as poachers, to move to other, less actively patrolled areas, to reduce the likelihood of being caught. Our specific model of spatial crime shift is based on two fundamental theories that govern crime occurrence. The first is the opportunity theory of crime [17, 18], which posits that crime locations are deliberate choices by criminals driven by their attractiveness based on a specific utility function. This motivates our consideration of spatial crime shift. The second is the crime clustering theory [19, 20], which posits that opportunities for similar crimes are often clustered and occur close to each other. This motivates another feature of our model: even if there is an incentive for attackers to deviate from their ideal locations, they are also averse to moving too far away, and may indeed be deterred in committing a crime if nearby opportunities do not present themselves. The net result of such *evasion* behavior, that is effectively in response to the incident prediction model f, is a spatial shift in crime incidents.

To formalize, consider a dataset D of potential incidents. We capture the attacker's behavior by *simulating* how the attacker would transform the dataset D in response to a

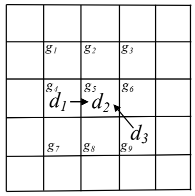

Figure 1: An illustration of spatial shifts by attackers to evade a learned model.

learned function f. To capture the idea that spatial shifts remain in close proximity to the baseline crime location, consider a particular incident d_i such that $l_i = g_k$ (say). Let N_k be the set of *neighboring* cells of g_k. This neighborhood structure can be exogenously specified for the spatial area of interest G; we consider some examples of this in our experiments later. Our model of the attacker is that they are able to shift the incident in space from g_k to any of its neighboring locations N_k. Thus, supposing that N_k includes g_k (since the attacker could potentially choose to not shift the incident), the attacker can choose a new location of crime incident d_i to be anywhere in N_k, as illustrated in Figure 1. We capture this shift by the variable $s_i^j \in \{0, 1\}$, which denotes the attacker's decision to shift incident d_i to cell g_j. Let s be the collection of spatial shifts chosen by the attacker for all the incidents in our dataset. The attacker's objective is to minimize the *predicted* likelihood, i.e., likelihood of incidents in the chosen location s given a fixed model f. Formally, the attacker aims to solve the following optimization problem:

$$\min_{s \in S} \sum_i f(x_i(s); \theta, w), \tag{2}$$

where we make explicit that the attacker's spatial shifts alter the predicted likelihood f by modifying the observed incident characteristics $x_i(s)$, and where S is the set of all possible spatial shifts over all incidents. Note the combinatorial structure of the attacker's optimization problem in having to consider all possible joint incident shifts S. This arises because the impacts of shifts on the likelihood are not necessarily independent (e.g., features w may depend on prior incidents; inter-arrival times between incidents in a given location are changed when a single incident is moved to a different location). We will address it below.

Defender Model: The learner's (defender's) goal is to learn a model f that is robust to adversarial shifts in incidents according to the model above. We formalize it as a robust likelihood maximization problem:

$$\max_\theta \min_{s \in S} \sum_i f(x_i(s), \theta; w). \tag{3}$$

We can equivalently view the model in Equation 3 as a Stackelberg game in which the leader is the learner who commits to a model θ, and the follower is the adversary who first observes the model θ and then shifts incident locations in response.

3 Approach

The optimization problem (3) is difficult to solve for a few reasons. First, the attacker optimization problem involves discrete decisions, making it difficult to deal with the bi-level nature of the problem. Second, the attacker problem is combinatorial. We now propose two general approaches to solve the proposed problem.

3.1 RSALA : Robust Spatial-Temporal Predictions with Linear Attacks

In the first approach, we frame the attacker's problem as a linear optimization problem with an exponential number of variables. Note that each attacker action (a specific choice of s) results in a different modification to the concerned set of incidents, resulting in a new collection of incidents with altered spatial locations. We refer to each such resulting *unique* set of incidents as a *chain*. In essence, a specific collection of shifts $s \in S$ results in a particular chain. We can then think of the attacker's full (combinatorial) action space as the set of all such chains — that is, the set of all possible manipulations to the original dataset. Let there be a total of c such chains. The attacker's objective thus reduces to choosing the chain that results in the lowest likelihood given the model f, which can be represented as

$$\min_\lambda \sum_{i=1}^{c} \lambda_i f_i(x; \theta, w) \tag{4a}$$

$$\text{s.t.} \quad \sum_{i=1}^{c} \lambda_i = 1 \tag{4b}$$

$$\lambda_i \in \{0, 1\} \ \forall i \in \{1, .., c\} \tag{4c}$$

where $f_i(x; \theta, w)$ denotes the likelihood of incidents in the ith chain, and λ is a binary variable which is set to 1 only for the chain that the attacker chooses. The obvious issue with this formulation is that c could be extremely large, making the problem intractable. We can address this issue by looking at the dual of problem 4. First, we point out that the attacker can only choose multiple chains as part of an optimal solution if they contribute the same utility to the attacker's objective. This crucial insight lets us relax the integrality constraint over λ without sacrificing the utility of the attacker, converting problem (4) into a linear program. Then, due to strong duality, we can directly replace the attacker's objective function in problem (3) with its dual, and represent the overall robust likelihood maximization problem as

$$\max_{\theta,\delta} \delta \tag{5a}$$

$$\text{s.t.} \quad \delta - f_i(x, \theta; w) \leq 0 \quad \forall i \in \{1, \ldots, c\} \tag{5b}$$

$$\delta \in \mathbb{R}, \ \theta \in \mathbb{R}^m \tag{5c}$$

where $\delta \in \mathbb{R}$ represents the dual variable.

This formulation has two important advantages: first, it converts the *max-min* hierarchy of problem (3) into a single convex maximization problem, and secondly, it puts the potentially large number of possible attacker actions into a collection of constraints. This, in turn, allows us to solve the problem using constraint generation. A constraint generation approach starts with a subset of the attacker actions, and iteratively updates the model by dynamically generating constraints according to actions taken by the attacker in response to the defender's strategy.

While such an approach makes our algorithm comparatively tractable, the attacker problem is still combinatorial. We use a crucial insight to tackle this. Consider the primary consequence of the attacker's actions: choosing a location of crime effectively changes the overall likelihood of the learned model. However, making such a decision *optimally* is unrealistic from an attacker's perspective, since it would require the attacker to be effectively clairvoyant (they have to account for time of incidents that have not occurred). Consider for example, the attacker choosing to evade detection by minimizing the likelihood of a discrete-time regression model f based on Poisson regression. While it tries to shift incident d_i from cell g_k to g_j (say), it must account for other incidents that could potentially happen in both the cells in the same time-step. This is clearly unreasonable, since in practice attackers cannot account for incidents that have not yet happened. We therefore simplify the attack model: at any point in time t, we restrict the attacker to minimize the likelihood of the model for all incidents $d_k \in D$ such that $t_k \leq t$, since at this time, the attacker can only have information about incidents that have happened before t. This assumption dramatically reduces the complexity of the inner problem, since now the attacker's objective is reduced to an optimization problem over a finite set of cells — the attacker can shift each incident to the cell that results in lowest likelihood, without considering how such a decision can potentially affect future incidents. We use these insights, and present our approach based on constraint generation in Algorithm 1.

We explain some added notation before explaining the algorithm. We use A as a short-hand to denote the attacker's objective function from formulation 2. At any iteration k of the algorithm, we refer to the current set of constraints by ϕ^k, the defender's parameters by θ^k, and the dataset used in iteration k by D^k (which gets updated according to the actions taken by the attacker). Further, we use *Solve*(ϕ^k) to denote solving problem 5 under constraints ϕ^k, and use *Attack*(θ^i) to denote the generation of the attacker's best response against θ^i. Also, we use *Update*(D^k, y) to denote a function that updates the existing dataset with manipulations generated as a response to a specific choice of θ made by the defender (thereby arriving at a new dataset D^{k+1}), and use $f^{D^k}(\theta)$ as a shorthand for $\sum_{d_i \in D^k} f(x_i; \theta, w)$.

Algorithm 1 RSALA

```
1: INPUT: Dataset D, Likelihood Model f, Adversarial Utility function A
2: OUTPUT: Robust model parameters θ*
3: Set θ^0 ← argmax_θ f(x; θ, w); k ← 0; Constraint set φ^0 ← Attack(θ^0); gap ← ∞
4: while gap > ε do
5:     θ^{k+1} ← Solve (φ^k)
6:     φ^{k+1} ← φ^k ∪ Attack(θ^{k+1})
7:     D^{k+1} ← Update(D^k, Attack(θ^{k+1}))
8:     gap ← f^{D^{k+1}}(θ^{k+1}) − f^{D^{k+1}}(θ^k)
9:     k ← k + 1
10: return θ^{k+1}
```

Now, at iteration k in the algorithm, we first compute the defender's optimal parameters θ^{k+1} by solving problem 5 under constraints ϕ^k (refer to step 5). We then update the constraint set by computing the attacker's best response to θ^{k+1} (refer to step 6). Such a response is straight-forward to compute, due to the relaxed version of the attacker's problem mentioned above. The attacker's response is then used to update the dataset (refer to step 7), which is then used in the subsequent iteration. This process is continued until the attacker's gain between successive iterations is within an exogenously specified parameter ϵ.

While *RSALA* is guaranteed to converge in finite time to the optimal solution, the strategy-space of the attacker could be extremely large, and solving the optimization problem 5 at every iteration of *RSALA* is computationally slow. This motivates us to create a heuristic approach, that can balance the trade-off between the quality of solutions and computation time of the algorithm. We call this approach **Ad**versary based **Grad**ient Descent (**AdGrad**).

3.2 AdGrad : Adversary Based Gradient Descent

Convex likelihood functions that do not have closed-form solutions can be maximized using gradient-based approaches. Our problem is not as straightforward: attacker actions s affect the model parameters θ, but these actions are a function of the model parameters as well.

We modify the standard gradient-based approach to enable the defender to take gradient steps that are based on the attacker's adversarial actions. We present this approach in Algorithm 2. We use the same notation as in Algorithm 1, and denote the attacker's decisions at iteration k by $s(k)$. At each iteration of gradient descent, we first calculate the best response of the attacker using the current parameters θ chosen by the defender (refer to step 5). This provides us with an updated set of data with adversarial manipulations, which is then used by the defender to update its parameters using a standard gradient step (refer to step 7). This process is repeated until convergence, and we use the same notion of convergence as in *RSALA*.

Algorithm 2 AdGrad

1: **INPUT** Dataset D, Likelihood Model f, Adversarial Utility function A
2: **OUTPUT** Robust model parameters θ^*
3: Set $\theta^0 \leftarrow \text{argmax}_\theta\ f(x;\theta,w)$; $k \leftarrow 0$; $gap \leftarrow \infty$
4: **while** $gap > \epsilon$ **do**
5: $\quad s(k+1) \leftarrow \text{Attack}(\theta^k)$
6: $\quad D^{k+1} \leftarrow \text{Update}(D^k, s(k+1))$
7: $\quad \theta^{k+1} \leftarrow \theta^k + \alpha\nabla f^{D^{k+1}}(\theta^k; x, w)$
8: $\quad gap \leftarrow f^{D^{k+1}}(\theta^{k+1}) - f^{D^{k+1}}(\theta^k)$
9: $\quad k \leftarrow k+1$
10: **return** θ^{k+1}

So far, we have described the overall idea behind robustness against spatial shifts in the context of incident prediction. Now, we dive into specific models, and apply this idea of robustness. To this end, we first show how robust incident prediction optimization problems can be framed for both continuous-time and discrete-time predictive models. Specifically, we present robustness in the context of a Poisson regression model (count-based and discrete-time), logistic regression (binary response-based model and discrete-time) and spatial-temporal survival analysis (continuous-time).

4 Robustness in Discrete-Time Incident Prediction

4.1 Count-based Model (Poisson Regression)

Consider that the total time in consideration in dataset D is divided into T time-steps. Let x_i^t be a random variable that denotes the number of incidents occurring at any time-step t in cell g_i. The likelihood model $f(x;\theta,w)$ therefore denotes the likelihood of x incidents occurring in a cell at a given time-step, where θ denotes the regression coefficients. In Poisson regression, the random variable of interest follows a Poisson distribution with mean μ, and $\mu = e^{\theta^\top w}$. Thus, the likelihood function for all incidents in dataset D can be represented as F(x;$\theta, w) = \prod_{t=1}^{T}\prod_{g_i\in G}\{\mu_{it}^{x_i^t}(e^{-\mu_{it}})/(x_i^t!)\}$, where w_{it} denotes the features associated with cell g_i at time-step t, and $\mu_{it} = \theta^\top w_{it}$.

Attacker Model: To capture this effect of spatial shifts by the attacker, we first define some notation. Recall that incident $d_i \in D$ and ℓ_i represent the ith incident in our dataset and its location respectively. Further, N_i denotes the *neighbors* of cell g_i. We assume that the attacker could move to any of the neighboring cells to commit the crime, in order to evade detection. The spatial parameter s_j^i is a binary variable that denotes the attacker's choice to shift incident d_j to cell g_i. In our problem, the attacker's objective is to minimize the likelihood of the forecasting model by optimizing over the spatial decisions s. The attacker problem, using the likelihood for Poisson regression, can be represented as

$$\min_s A(s;\theta) \equiv \sum_{t=1}^{T}\sum_{g_i\in G}\Big\{(\sum_{j=1}^{n}\mathbb{1}(d_j,t)s_j^i)\theta^\top w_{it} - e^{\theta^\top w_{it}} - \log(\sum_{j=1}^{n}\mathbb{1}(d_j,t)s_j^i)!\Big\} \tag{6a}$$

$$\text{s.t.} \sum_{g_i\in N_{\ell_j}} s_j^i = 1 \;\; \forall j \in \{1,\ldots,n\} \tag{6b}$$

$$s_j^i \in \{0,1\} \;\; \forall g_i \in G\ \forall d_j \in D \tag{6c}$$

where $\mathbb{1}(d_j,t)$ is an indicator function set to 1 if incident d_j occurred in time-step t, and is 0 otherwise. Constraint 6b enforces the natural bound that the attacker can shift one incident to only location.

Robust Poisson Regression Given the adversarial manipulations, the defender tries to maximize the likelihood of the learned model. Thus, the overall problem of robust incident prediction can be defined as

$$\max_\theta \min_s A(s,\theta) \tag{7a}$$

$$\text{s.t.} \sum_{g_i\in N_{\ell_j}} s_j^i = 1 \;\; \forall j \in \{1,\ldots,n\} \tag{7b}$$

$$s_j^i \in \{0,1\} \;\; \forall g_i \in G\ \forall d_j \in D\ \theta \in \mathbb{R}^m \tag{7c}$$

We can directly use our algorithmic approaches to solve problem (7).

4.2 Binary Prediction Model (Logistic Regression)

Having looked at a count-based regression model, we now explain how our idea of robustness can be applied to binary models of spatial-temporal incident prediction. We choose logistic regression as our approach of interest, that models a binary output variable x through a logit transformation, such that $P(x = 1; \theta, w) = \frac{1}{1+e^{-\theta^\top w}}$, where θ is the set of regression coefficients, and w represents an associated set of features.

We make some assumptions to use logistic regression in the context of incident prediction, as it models a binary response. As in Poisson regression, we assume that the total time under consideration is divided into T discrete time-steps. We further assume a granularity of temporal discretization that ensures that in each time step, only one incident occurs in a cell (essentially, we assume the response variable that measures the presence of crimes in a cell at a specific time step is binary). Such an assumption is actually reasonable for certain kinds of crime incidents (poaching in forests, for example), which are sufficiently displaced temporally. In the absence of adversarial manipulations, the likelihood of incident occurrence across all cells over the entire temporal horizon can be represented as $f(\theta; x, w) = \sum_{g_i \in G} \sum_{t=1}^{T} \log(\frac{1}{1+e^{-\theta^\top w}}) x_i^t + \log(1 - (\frac{1}{1+e^{-\theta^\top w}}))(1 - x_i^t)$ The defender's objective without adversarial manipulations is simply finding $\theta^* = \text{argmax}_\theta f(x; \theta, w)$

Attacker Model: As before, we denote the attacker's action space by the binary spatial parameter $s_j^i \in \{0, 1\}$, which is 1 if the attacker chooses to shift incident d_j to cell g_i, and 0 otherwise. The attacker's objective is then straight-forward, and can be represented as

$$\min_s A(s; \theta) \equiv \sum_{g_i \in G} \sum_{t=1}^{T} \mathbb{1}(d_j, t) \Bigg\{ \log(\frac{1}{1+e^{-\theta^\top w}})(s_j^i) + \log(1 - (\frac{1}{1+e^{-\theta^\top w}}))(1 - s_i^t) \Bigg\} \tag{8a}$$

$$\text{s.t.} \quad \sum_{g_i \in N_{\ell_j}} s_j^i = 1 \;\; \forall j \in \{1, \ldots, n\} \tag{8b}$$

$$s_j^i \in \{0, 1\} \;\; \forall g_i \in G \; \forall d_j \in D \; \theta \in \mathbb{R}^m \tag{8c}$$

where constraint 8b ensures that the attacker's shifts each incident only once.

Robust Logistic Regression Given adversarial intervention, the defender tries to maximize the likelihood of model after taking into account the potential spatial shifts by the attacker. The robust optimization problem can be represented as

$$\max_\theta \min_s A(s, \theta) \tag{9a}$$

$$\text{s.t.} \quad \sum_{g_i \in N_{\ell_j}} s_j^i = 1 \;\; \forall j \in \{1, \ldots, n\} \; \forall g_k \in G \tag{9b}$$

$$s_j^i \in \{0, 1\} \;\; \forall g_i \in G \; \forall d_j \in D \; \theta \in \mathbb{R}^m \tag{9c}$$

5 Robustness in Continuous-Time Incident Prediction

Now, we shift our attention to models of incident prediction that operate in a continuous-time domain. In this case, for each cell $g_i \in G$, we use the random variable X to denote the time between successive incidents, such that $x_i = t_i - t_{i-1}$ represents the time to arrival of the ith incident in the dataset. The goal of a continuous-time predictive model is to learn a distribution $f(x; \theta, w)$ over inter-arrival time between incidents, where θ represents the regression coefficients. Recently, survival analysis has been shown to have state-of-the-art performance for such problems, for example, in crime and traffic accident prediction settings [6, 21, 22]. A parametric survival model for a specific data point $\{x_i, w_i\}$ can be defined as $\log(x_i) = \sum_{i=1}^{m} \theta_j w_{ij} + z$, where w_{ij} denotes the realization of feature j associated with data point i, and z is the error term, distributed according to distribution h. The particular choice of the distribution f depends on how we model the error term z. We adopt a common exponential distribution model for X, used previously in the context of incident prediction [6, 21, 22]. The log-likelihood of the observed data can be represented as $f(x; \theta, w) = \sum_{i=1}^{n} \log h(\log(x_i) - w_i^\top \theta)$. Given such a model for likelihood, the defender tries to find the parameters θ^*, such that $\theta^* = \text{argmax}_\theta f(x; \theta, w)$.

Attacker Model: We assume that the attacker first observes the survival model f, and may shift to a different cell so as to commit a crime in an area with a smaller predicted crime frequency according to f.

To formalize the model, we introduce some notation. For each cell $g_i \in G$, let P_i to define possible successive (in time) pairs of data-points $\{d_k, d_l\}$ that could occur in g_i due to adversarial manipulation. We are specifically interested in successive incidents since the random variable we want to model is the incident inter-arrival time. We illustrate the idea behind such pairs in Figure 1. Consider the set of cells $\{g_1, \ldots, g_9\}$ and incidents $\{d_1, d_2, d_3\}$, which we assume are ordered by their times of occurrences. Now, to look at adversarial perturbations in the dataset, we look at cell g_5 as an example. Incidents in its neighborhood that could move to it form the set $\{d_1, d_2, d_3\}$ (note that this includes d_2 since the attacker could chose to not deviate from the original location of the incident). This gives us three pairs of successive incidents, namely the set $\{(d_1, d_2), (d_2, d_3), (d_1, d_3)\}$. Observe that (d_1, d_3) is also a *potential* pair of successive incidents in g_5 since the attacker could chose to move d_2 to a different cell, which would result in d_1 and d_3 occurring successively in g_5. Moreover, the pair (d_1, d_3) could exist as a pair of (possible) successive incidents if and only if d_2 moves to a different cell. In order to capture this, for any cell g_m and pair of incidents $(d_i, d_j) \in P_m$, we use B_{ij}^m to denote the set of all incidents $d_k \in D$ that could potentially move to g_m such that $t_i < t_k < t_j$. This lets us take into account the fact that d_i and d_j could occur consecutively in g_m if and only if they both move to g_m and none of the incidents from the set B_{ij}^m move to g_m. Finally, we capture the decision of the attacker to move an incident to a cell by the variable s_j^i, which is a binary variable that captures the attacker's decision of shifting d_j shifting to cell g_i.

We are now ready to present the attacker's problem formally as the following optimization problem:

$$\min_s A(s;\theta) \equiv \sum_{g \in G} \sum_{i,j \in P_g} s_i^g s_j^g \Big\{ \prod_{d_k \in B_{ij}^g} (1 - s_k^g) \Big\} f(x_{ij}; w_i, \theta) \tag{10a}$$

$$\text{s.t.} \sum_{g_i \in N_{\ell_j}} s_j^i = 1 \quad \forall j \in \{1, .., n\} \tag{10b}$$

$$s_j^i \in \{0, 1\} \quad \forall g_i \in G, \forall d_j \in D \tag{10c}$$

where x_{ij} denotes the time between potential pair of successive incidents i and j. The objective is the log-likelihood of incident arrivals *after* the shifts introduced by the attacker.

In this optimization problem, the attacker considers all potential pairs of incidents that could occur in each cell due to possible manipulation, and chooses incidents such that the overall likelihood of the model is minimized. The product term in the objective function ensures that likelihood is captured only for decisions made by the attacker, and not for all the possible options that the attacker has. The attacker's action space in terms of which cells to move the crimes to is thus represented by the binary decision variables y, and constraint 10b ensures that the attacker can shift each incident to only one cell (since the same crime cannot be committed at the same time at two different locations).

Robust Survival Analysis The defender's goal is to maximize the likelihood objective A over parameters θ. The resulting formulation is identical to that in Equation (7) for robust Poisson regression, with the likelihood objective $A(s, \theta)$ for the survival model the sole difference.

6 Experimental Evaluation

6.1 Data

We used three real-world crime data from two sources -

- Assault and burglary data from a major metropolitan area of the US — we considered 7057 assault incidents and 2184 burglaries from 7 months in 2014. To further verify our approach, we created two overlapping datasets from 7 months of data, each with 3 months of data for training and 1 month's data for testing. For spatial discretization, we used square grids of side 1 mile throughout. We used risk-terrain, spatial-temporal, and weather data as features.

- Poaching data from Uganda - We use data from a national Park in Uganda, that covers 3893 sq. km. As the park covers an extremely large area, each cell is patrolled only a few times a year. As a result, we do not split the data. We use 3 years of data for training, and 1 year of data for testing. For spatial discretization, we used square grids of side 1 km throughout. We use geographic and terrain data, as well as past animal sightings as features. Patrol reporting in the park is done as a binary response, and we use a total of 18255 reports, with 2602 of the reports being positive (evidence of poaching) and the rest being negative.

6.2 Setup

We use urban data (marked with counts and exact time of events) to evaluate Poisson regression and survival analysis, and poaching data (marked only as binary response) to evaluate logistic regression. We define neighbors for a cell based on its adjacent cells. We also investigate the effect of varying the extent of spatial shifts that the attacker has access to, but present the results later.

We ran experiments for Poisson regression and survival analysis on a 2.4GHz 32-core Ubuntu Linux machine with 32 GB RAM. The experiments for logistic regression (due to confidentiality agreements on data) had to be performed on centers approved by the national park, and were run on 3.2Ghz 6-core Ubuntu Linux machine with 16GB RAM. While evaulating the algorithm *RSALA*, we solved the optimization problem (5) using CVXPY [23]. The only hyper-parameters in our model are the learning rate α for *AdGrad* and the attacker's definition of "neighboring grids" to which the criminals can move. We performed experiments using multiple learning rates, and found that a learning rate of 0.001 as the best parameter value.

6.3 Data Description

We provide a detailed description of the features we use in our predictive models here.

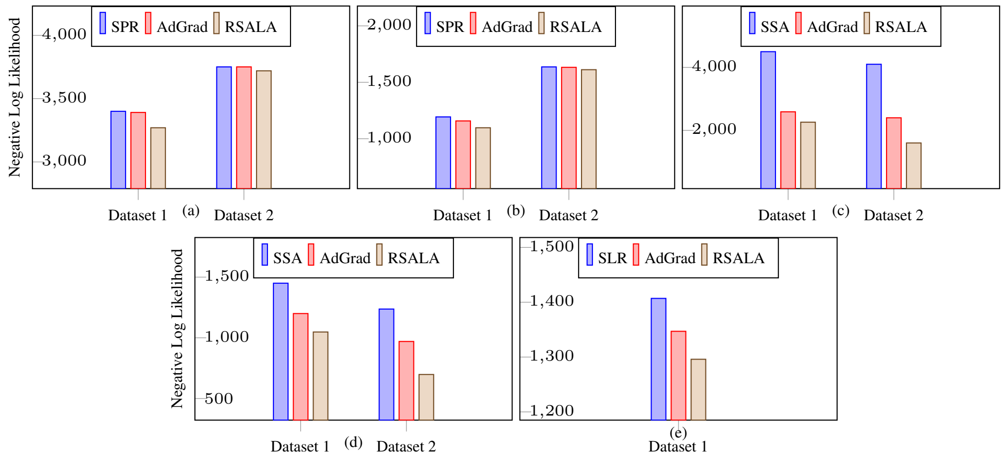

Figure 2: Robustness measured by negative-log-likelihood on test set (lower is better) : (a) Poisson Regression (Assault data) (b) Poisson Regression (Burglary data) (c) Survival Analysis (Assault data) (d) Survival Analysis (Burglary data) (e) Logistic Regression (Poaching data)

6.3.1 Feature set for urban crime data

We used the following covariates for predictive models using urban crime data.

- **Temporal cycles** We use a binary feature for weekdays and weekends. In order to look at the effect of time of day on incidents, we split each day into six zones of four hours each, and captured these by binary features.
- **Temporal and spatial incident correlation** For each cell, we looked at the past incident counts in the last week and month in it as well as neighboring cells as features to capture the effect of temporal and spatial correlation among incidents. We also treated the number of past incidents in each severity category as a feature while predicting incident severity, and considered the long-term effect of temporal correlation by looking at the average number of incidents in the past year.
- **Weather** It is known that weather affects crime occurrence [24]. We included a collection of features, such as rainfall, snowfall, and mean temperature to capture this effect. Weather data was collected from a weather station located in roughly the center of the county (we suppress an exact citation for blind review).
- **Risk-terrain features** We used several risk-terrain features to aid to our prediction model, including population density, housing density, and mean household income at a census tracts level. We also used data with 624 retail shops that sell liquor, 2494 liquor outlets, 41 homeless shelters, and 52 pawn shops.

6.3.2 Feature set for poaching data

We use the following set of features in predictive models for poaching data.

- **Spatial features** We used features that modeled distances to the closest national park boundary, water, road, town, patrol post, mineral lick location, and the cost to reach to the closest village.
- **Geographic terrain** We used data to model habitat type, elevation, slope, topographic wetness index and NPP (net primary productivity) of cells.
- **Animal density** We used historical data to account for animal presence in each of the cells. This included animals like buffalo, elephant, hippopotamus, giraffe, kob, oribi, warthog, and waterbuck.

We use models without accounting for adversarial interventions as our primary baseline. This has two advantages. First, it allows us to evaluate the efficacy of our algorithms on models that are not explicitly trained to be robust, and secondly, it lets us compare our approach with a baseline that has shown better performance than other state-of-the-art alternatives [6, 25]. We refer to the baseline models as standard survival analysis (SSA), standard Poisson regression (SPR), and standard logistic regression (SLR).

6.4 Results

Robustness — To compare the efficacies of the two algorithmic approaches, we begin by looking at how the algorithms *AdGrad* and *RSALA* perform on unseen data, and directly compare their performance to models that do not account for adversarial intervention. To do this, we introduce adversarial manipulations on our test data based on the attacker model described in section 2. We show the results on robustness for all the approaches in Figure 5. In all the cases, we observe that both *RSALA* and *AdGrad* ensure higher robustness against adversarial manipulations than *SPR* and *SSA*. Also, as expected, *RSALA* outperforms *AdGrad*, since it is guaranteed to converge to the optimal solution.

Computational Time — Next, we present training times for robust predictive models. Training times for crime prediction algorithms can be a crucial factor in their deployment, as intervention strategies are often calculated periodically after each shift undertaken by patrols. We show our cumulative training times in Figure 6.4. As expected, we see that *AdGrad* takes considerably less time than *RSALA* to train.

Evaluating attacker's budget — We also seek to understand the effect of the attacker's geographic constraint (budget) on the robustness of the models. In order to do so, we vary the definition of "neighbors" that the attacker can shift to. We increase the attacker's budget gradually; consider a crime in cell $g_i \in G$ and a budget of γ (say). Such a budget would enable the attacker to move to any cell $g_k \in G$, such that g_k and g_i have at-most γ other cells between them (a budget of $\gamma = 0$ reverts to standard models with no spatial shifts). With this notion of attacker's geographic budget, we repeat the entire set of experiments. Our findings for the performance of varying attacker budget are consistent with our findings with using the immediate neighborhood of a cell as potential locations for shifts. Instead, we seek to visualize the predictions made by the forecasting models as we increase the attacker's ability (i.e., as the criminals move farther away to commit crimes).

Specifically, we plot the spatial-temporal survival density learned by varying the attacker's budget as heatmaps over the actual metropolitan area under consideration. We generate the heat maps by predicting assault crimes across all cells for 3 days, and we repeat this procedure 50 times to reduce variance in the predictions. We show the resulting images in Fig. 4, that are generated by attacker budgets from 0 to 3. We see that as the attacker's budget increases, the forecasting models become increasingly cognizant of potential crimes throughout the city, resulting in a spatial distribution of incidents that is spread out. An important insight revealed by this experiment is that a very high attacker budget can create models which essentially predict a high likelihood of crime occurring throughout the area under consideration, which is not necessarily useful in policing. Therefore, we point out that the attacker's budget is a crucial hyperparameter in our models and recommend that system designers choose it carefully based on actual capabilities of the attacker, as well as opportunities that the spatial area under consideration provides for committing crimes.

Performance on non-adversarial data — While adversarial robustness of a model is considered solely in the presence of an attacker, it is important to evaluate the performance of the model on non-adversarial data. The very nature of robustness in our context dictates that we sacrifice performance on non-adversarial data to gain robustness against possible manipulation in attackers' behavior. However, it is crucial to investigate the nuances of this trade-off. To understand this, we sort the total set of cells G according to frequency of incident arrival, and divide the sorted set into 10 bins — the first bin consists of grids with the lowest frequency of incidents, and so on. Then, we assessed the difference between predicted rates by standard models (SSA, SPR and SLR) and robust models (RSALA and AdGrad). To reduce variance in our approach, we evaluated the rate of incident arrival for each bin on 10 randomly chosen points in time from the test set. We present the results for Poisson regression in Figure 5 (our results are consistent across crime types and regression models).

We see that in order to gain robustness, our algorithmic approach underestimates incidents in grids with highest frequency. This reduction is a result of potential spatial shifts to grids with lower frequencies, on which our approach slightly over-estimates arrival rates than a baseline model. This is expected; in fact, this is precisely the behavioral change in attackers that we seek to capture. However, if our approach severely under-estimates the chances of potential crimes in grids with high frequencies, it could be potentially detrimental to policing strategies. We see that RSALA only underestimates frequency at high-crime grids by about 0.04 incidents per hour to gain robustness; AdGrad naturally underestimates less, since it achieves lesser robustness.

6.5 Discussion

Before we conclude, we identify limitations and provide a few words of caution. First, it is important to note that while our optimization framework only accounts for spatial shifts, temporal shifts are also possible in response to learned models. It is natural to assume that attackers can shift their actions in time, by waiting to commit a crime at a particular location (or by committing it early).

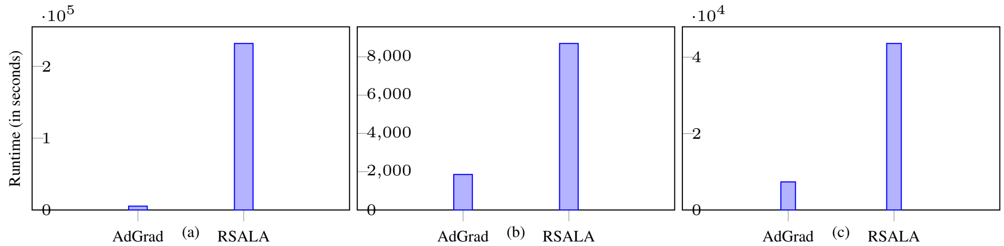

Figure 3: Training time — (a) Poisson Regression (b) Survival Analysis (c) Logistic Regression

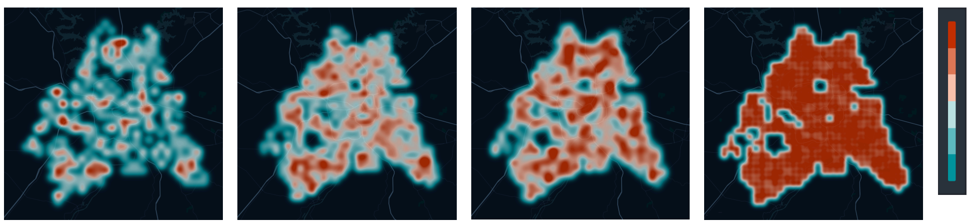

Figure 4: Predicted incident density for assault crimes plotted according to a varying attacker budget. Images from left to right are plotted with an attacker budget of 0, 1, 2 and 3 respectively.

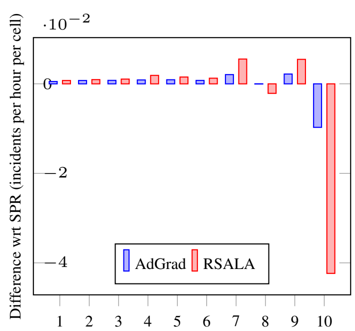

Figure 5: Difference in predicted rates between standard and robust models

Through this paper, we establish the foundation of robustness in case of spatial-temporal predictions, and will address the issue of temporal shifts in future work. Next, we point out some insights about the experimental findings on non-adversarial data. A slight underestimation of frequency in high-density cells is not a drawback in itself; after all, robustness is gained so that the defender can account for adversarial manipulations. However, this raises an important question. First, how does one choose the extent of adversarial manipulation? This issue is important to address, but outside the current scope of our research. This area of crime forecasting and criminology needs more research and empirical trials. In adversarial models, the extent of spatial shifts should be chosen based on the specific problem and the type of crime at hand.

7 Conclusion

Crime prediction models have traditionally been agnostic to adversarial manipulations in criminal behavior in response to learned models. We systematically bridge this gap by creating a principled nested optimization-based framework for predicting crimes that is robust to such manipulations (we present a discussion about the implications of such an approach and some insights in the supplementary material). We frame the interaction between the defender (law-enforcement authorities) and the attacker (people who commit crimes) as a Stackelberg game, and propose two algorithmic approaches to solve the our problem. We show how our approach can accommodate both continuous-time and discrete-time (count-based as well as binary response-based) predictive models. To this end, we form optimization problems for accounting for spatial shifts in Poisson regression, logistic regression, and survival analysis. Finally, we use three real-world crime datasets to evaluate our approaches. Experimental results demonstrate that our approach is significantly more robust to adversarial manipulations than standard predictive models.

References

[1] A. T. Murray, I. McGuffog, J. S. Western, and P. Mullins. Exploratory spatial data analysis techniques for examining urban crime implications for evaluating treatment. *British Journal of Criminology*, 41(2):309–329, 2001.

[2] Leslie W Kennedy, Joel M Caplan, and Eric Piza.

Risk clusters, hotspots, and spatial intelligence: risk terrain modeling as an algorithm for police resource allocation strategies. *Journal of Quantitative Criminology*, 27(3):339–362, 2011.

[3] M. B. Short, M. R. D'Orsogna, V. B. Pasour, G. E. Tita, P. J. Brantingham, A. L Bertozzi, and L. B. Chayes. A statistical model of criminal behavior. *Mathematical Models and Methods in Applied Sciences*, 18(supp01):1249–1267, 2008.

[4] C. Zhang, A. Sinha, and M. Tambe. Keeping pace with criminals: Designing patrol allocation against adaptive opportunistic criminals. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 1351–1359, 2015.

[5] C. Zhang, V. Bucarey, A. Mukhopadhyay, A. Sinha, Y. Qian, Y. Vorobeychik, and M. Tambe. Using abstractions to solve opportunistic crime security games at scale. In *International Conference on Autonomous Agents and Multiagent Systems*, 2016.

[6] A. Mukhopadhyay et al. Optimal allocation of police patrol resources using a continuous-time crime model. In *International Conference on Decision and Game Theory for Security*. Springer, 2016.

[7] A. Mukhopadhyay, G. Pettet, S. Vazirizade, Y. Vorobeychik, M. Kochenderfer, and A. Dubey. A review of emergency incident prediction, resource allocation and dispatch models. *arXiv (preprint 2006.04200)*, 2020.

[8] T. A Reppetto. Crime prevention and the displacement phenomenon. *Crime & Delinquency*, 22(2):166–177, 1976.

[9] M. B. Short, P. J. Brantingham, A. L. Bertozzi, and G. E. Tita. Dissipation and displacement of hotspots in reaction-diffusion models of crime. *Proceedings of the National Academy of Sciences*, 107(9):3961–3965, 2010.

[10] M. A. Andresen and N. Malleson. Police foot patrol and crime displacement: a local analysis. *Journal of Contemporary Criminal Justice*, 30(2):186–199, 2014.

[11] M. Barreno, B. Nelson, R. Sears, A. D. Joseph, and J. D. Tygar. Can machine learning be secure? In *2006 ACM Symposium on Information, Computer and Communications Security*, 2006.

[12] L. Huang, A. D. Joseph, B. Nelson, B. Rubinstein, and J. D Tygar. Adversarial machine learning. In *4th ACM workshop on Security and artificial intelligence*, pages 43–58. ACM, 2011.

[13] V. Yevgeniy and M. Kantarcioglu. Adversarial machine learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 12(3):1–169, 2018.

[14] S. Gholami et al. Adversary models account for imperfect crime data: Forecasting and planning against real-world poachers. In *17th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

[15] Engineering National Academies of Sciences, Medicine, et al. *Proactive policing: Effects on crime and communities*. National Academies Press, 2018.

[16] R. R Johnson. Citizen expectations of police traffic stop behavior. *Policing: An International Journal of Police Strategies & Management*, 27(4):487–497, 2004.

[17] R. V. Clarke. Affect and the reasoning criminal: Past and future. In *Affect and cognition in criminal decision making*, pages 38–59. Routledge, 2013.

[18] M. J. Hindelang, M. R. Gottfredson, and J. Garofalo. *Victims of personal crime: An empirical foundation for a theory of personal victimization*. Ballinger Cambridge, MA, 1978.

[19] S. Chakravorty. Identifying crime clusters: The spatial principles. *Middle States Geographer*, 28:53–58, 1995.

[20] G. Farrell and K. Pease. *Prediction and Crime Clusters*, pages 3862–3871. Springer New York, New York, NY, 2014.

[21] G. Pettet et al. Incident analysis and prediction using clustering and bayesian network. In *IEEE International Conference on Smart City Innovations*, 2017.

[22] A. Mukhopadhyay, Y. Vorobeychik, A. Dubey, and G. Biswas. Prioritized allocation of emergency responders based on a continuous-time incident prediction model. In *International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2017.

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

[24] E. G. Cohn. Weather and crime. *British Journal of Criminology*, 30(1):51–64, 1990.

[25] D. W. Osgood. Poisson-based regression analysis of aggregate crime rates. *Journal of Quantitative Criminology*, 16(1):21–43, 2000.