Optimal Attack against Autoregressive Models by Manipulating the Environment

Yiding Chen, Xiaojin Zhu
Department of Computer Sciences, University of Wisconsin-Madison
{yiding, jerryzhu}@cs.wisc.edu

Abstract

We describe an optimal adversarial attack formulation against autoregressive time series forecast using Linear Quadratic Regulator (LQR). In this threat model, the environment evolves according to a dynamical system; an autoregressive model observes the current environment state and predicts its future values; an attacker has the ability to modify the environment state in order to manipulate future autoregressive forecasts. The attacker's goal is to force autoregressive forecasts into tracking a target trajectory while minimizing its attack expenditure. In the white-box setting where the attacker knows the environment and forecast models, we present the optimal attack using LQR for linear models, and Model Predictive Control (MPC) for nonlinear models. In the black-box setting, we combine system identification and MPC. Experiments demonstrate the effectiveness of our attacks.

Introduction

Adversarial learning studies vulnerability in machine learning, see e.g. (Vorobeychik and Kantarcioglu 2018; Joseph et al. 2018; Liu et al. 2017; Biggio and Roli 2017; Lowd and Meek 2005). Understanding optimal attacks that might be carried out by an adversary is important, as it prepares us to manage the damage and helps us develop defenses. Time series forecast, specifically autoregressive model, is widely deployed in practice (Hamilton 1994; Box et al. 2015; Fan and Yao 2008) but has not received the attention it deserves from adversarial learning researchers. Adversarial attack in this context means an adversary can subtly perturb a dynamical system at the current time, hence influencing the forecasts about a future time. Prior work (Alfeld, Zhu, and Barford 2016; 2017) did point out vulnerabilities in autoregressive models under very specific attack assumptions. However, it was not clear how to formulate general attacks against autoregressive models.

There are extensive studies on **batch** adversarial attacks against machine learning algorithms. But there is much less work on **sequential** attacks. We say an attack is batch if the attacker performs one attack action at training or test time (the attacker is allowed to change multiple data points); an attack is sequential if the attacker take actions over time. There are batch attacks against support vector machine (Biggio, Nelson, and Laskov 2012; Biggio et al. 2014), deep neural networks (Goodfellow, Shlens, and Szegedy 2014; Nguyen, Yosinski, and Clune 2015), differentially-private learners (Ma, Zhu, and Hsu 2019), contextual bandits (Ma et al. 2018), recurrent neural networks (Papernot et al. 2016), online learning (Wang and Chaudhuri 2018) and reinforcement learning (Ma et al. 2019). Some of these victims are sequential during deployment, but they can be trained from batch offline data; hence they can be prone to batch attacks. In contrast, (Jun et al. 2018) and (Zhang and Zhu 2019) study sequential attacks against stochastic bandits and sequential prediction. Our work studies sequential attack against autoregressive model, which is closer to these two papers. Meanwhile, control theory is receiving increasing attention from the adversarial learning community (Recht 2018; Zhu 2018; Lessard, Zhang, and Zhu 2018). Our work strengthens this connection.

This paper makes three main contributions: (1) We present an attack setting where the adversary must determine the attack sequentially. This generalizes the setting of (Alfeld, Zhu, and Barford 2016; 2017), where the adversary can decide the attack after observing all environmental state values used for forecast. (2) We formulate the attacks as an optimal control problem. (3) When the attacker knows the environmental dynamics and forecast model (white-box setting), we solve the optimal attacks with Linear Quadratic Regulator (LQR) for the linear case, or Model Predictive Control (MPC) and iterative LQR (iLQR) for the nonlinear case; when the attacker does not know the environmental or forecaster(black-box setting), we additionally perform system identification.

The Attack Setting

Autoregressive Review

To fix notation, we briefly review time series forecasting using autoregressive models. There are two separate entities:

1. The **environment** is a fixed dynamical system with scalar-valued states $x_t \in \mathbb{R}$ at time t. The environment has a (potentially non-linear) q-th order dynamics f and is subject to zero-mean noise $w_t \in \mathbb{R}$ with $\mathbb{V}(w_t) = \sigma^2$. Without manipulations from the adversary, the environmental state

evolves as

$$x_{t+1} = f(x_t, \ldots, x_{t-q+1}, w_t) \tag{1}$$

for $t = 0, 1, \ldots$. We take the convention that $x_i = 0$ if $i < 0$. We allow the dynamics f to be either linear or nonlinear.

2. The **forecaster** makes predictions of future environmental states, and will be the victim of the adversary attack. In this paper we mainly focus on a fixed linear $AR(p)$ autoregressive forecaster, regardless of whether the environment dynamics f is linear or not. Even though we allow the forecast model to be nonlinear in black-box setting, we use linear function to approximate the nonlinear autoregressive model. We also allow the possibility $p \neq q$. At time t, the forecaster observes x_t and uses the p most recent observations $x_t, \ldots, x_{t-p+1}$ to forecast the future values of the environmental state. A forecast is made at time t about a future time $t' > t$, we use the notation $y_{t'|t}$ to denote it.

Specifically, at time t the forecaster uses a standard $AR(p)$ model to predict. It initializes by setting $y_{t+1-i|t} = x_{t+1-i}$ for $i = 1 \ldots p$. It then predicts the state at time $t+1$ by

$$y_{t+1|t} = \hat{\alpha}_0 + \sum_{i=1}^{p} \hat{\alpha}_i y_{t+1-i|t}, \tag{2}$$

where $\hat{\alpha}_0, \hat{\alpha}_1, \ldots, \hat{\alpha}_p$ are coefficients of the $AR(p)$ model. We allow the $AR(p)$ model to be a nonlinear function in the black-box setting. The $AR(p)$ model may differ from the true environment dynamics f even when f is linear: for example, the forecaster may have only obtained an approximate model from a previous learning phase. Once the forecaster predicts $y_{t+1|t}$, it can plug the predictive value in (2), shift time by one, and predict $y_{t+2|t}$, and so on. Note all these predictions are made at time t. In the next iteration when the true environment state evolves to x_{t+1} and is observed by the forecaster, the forecaster will make predictions $y_{t+2|t+1}, y_{t+3|t+1}$, and so on.

The Attacker

We next introduce a third entity – an **adversary** (a.k.a. attacker) – who wishes to control the forecaster's predictions for nefarious purposes. The threat model is characterized by three aspects of the adversary:

(i) Knowledge: In the white-box setting, the attacker knows everything above; in the black-box setting, neither environmental dynamics nor forecaster model are known to the attacker.

(ii) Goal: The adversary wants to force the forecaster's predictions $y_{t'|t}$ to be close to some given *adversarial reference target* $y^{\dagger}_{t'|t}$ (the dagger is a mnemonic for attack), for selected pairs of (t, t') of interest to the adversary. Furthermore, the adversary wants to achieve this with "small attacks". These will be made precise below.

(iii) Action: At time t the adversary can add $u_t \in \mathbb{R}$ (the "control input") to the noise w_t. Together u_t and w_t enter the environment dynamics via:

$$x_{t+1} = f(x_t, \ldots, x_{t-q+1}, u_t + w_t). \tag{3}$$

We call this the *state attack* because it changes the underlying environmental states, see Figure 1.

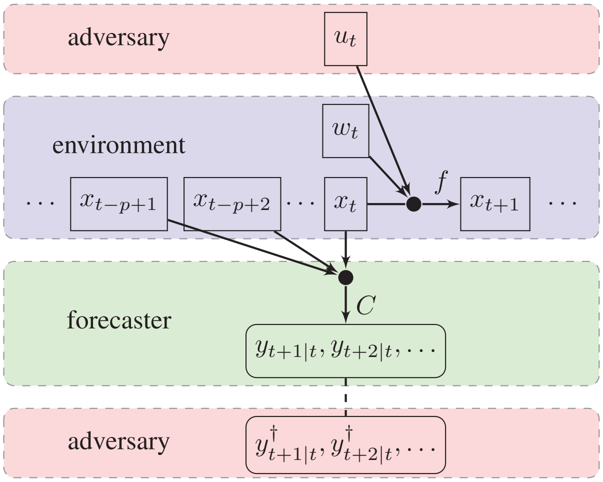
adversary
u_t
environment
w_t
$\cdots$ x_{t-p+1} x_{t-p+2} $\cdots$ x_t
f
x_{t+1} $\cdots$
C
forecaster
$y_{t+1|t}, y_{t+2|t}, \cdots$
adversary
$y^{\dagger}_{t+1|t}, y^{\dagger}_{t+2|t}, \cdots$

Figure 1: The state attack. The lowest layer depicts the attack target, and the adversary compares it against the forecaster's predictions.

White-Box Attack as Optimal Control

We now present an optimal control formulation for the white-box attack. Following control convention (Lee and Markus 1967; Kwakernaak and Sivan 1972), we rewrite several quantities from the previous section in matrix form, so that we can define the adversary's state attack problem by the tuple $(F, C, \{\mathbf{y}^{\dagger}_{t'|t}\}, \{Q_{t'|t}\}, R, T)$. These quantities are defined below.

We introduce a vector-valued environment state representation (denoted by boldface) $\mathbf{x}_t := (1, x_t, \ldots, x_{t-p+1})^{\top} \in \mathbb{R}^{p+1}$. The first entry 1 serves as an offset for the constant term $\hat{\alpha}_0$ in (2). We let $\mathbf{x}_0$ be the known initial state. It is straightforward to generalize to an initial distribution over $\mathbf{x}_0$, which adds an expectation in (9). We rewrite the environment dynamics under adversarial control (3) as $\mathbf{x}_{t+1} = F(\mathbf{x}_t, u_t, w_t) := (1, f(x_t, \ldots, x_{t-p+1}, u_t + w_t), x_t, \ldots, x_{t-p+2})^{\top}$. If f is nonlinear, so is F.

In control language, the forecasts are essentially measurements of the current state $\mathbf{x}_t$. In particular, we introduce a vector of p predictions made at time t about time $t'-p+1, \ldots, t'$ as $\mathbf{y}_{t'|t} := (1, y_{t'|t}, y_{t'-1|t}, \ldots y_{t'-p+1|t})^{\top} \in \mathbb{R}^{p+1}$. (For completeness, we let $y_{t'|t} = x_t$ when $t' \leq t$.) The forecaster's $AR(p)$ forecast model specifies the (linear) measurements as follows. We introduce the $(p+1) \times (p+1)$ measurement (i.e. forecast) matrix [1] ,

[1] The forecaster usually only has an estimate of f. It is likely that the forecaster's $AR(p)$ model has a different order p than the environment dynamics' order q. For simplicity, we will assume $p = q$ below, but explain how to handle $p \neq q$ in Appendix A.

$$C = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ \hat{\alpha}_0 & \hat{\alpha}_1 & \hat{\alpha}_2 & \cdots & \hat{\alpha}_{p-1} & \hat{\alpha}_p \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix}. \tag{4}$$

Then the measurements / forecasts are:

$$\mathbf{y}_{t'|t} = C^{t'-t}\mathbf{x}_t, \quad \forall t' > t. \tag{5}$$

There are cases that $y_{t'|t}$ does not depend on x_t but has been decided at time $\tilde{t} < t$. In such cases, we can simply redefine $y_{t'|t}$ to be $y_{t'|\tilde{t}}$ and rewrite the prediction matrix. For simplicity, we assume $y_{t'|t}$ always depends on x_t in the rest of the paper.

We also vectorize adversarial reference target: $\mathbf{y}^\dagger_{t'|t} := (1, y^\dagger_{t'|t}, y^\dagger_{t'-1|t}, \dots y^\dagger_{t'-p+1|t})^\top \in \mathbb{R}^{p+1}$. We simply let $y^\dagger_{t'|t} = 0$ when $t' \le t$: this is non-essential. In fact, for (t, t') pairs that are uninteresting to the adversary, the target value $y^\dagger_{t'|t}$ can be undefined as they do not appear in the control cost later.

The cost of the adversary consists of two parts: (1) how closely the adversary can force the forecaster's predictions to match the adversarial reference targets; (2) how much control u_t the adversary has to exert. The $(p+1) \times (p+1)$ matrices $Q_{t'|t}$ define the first part, namely the cost of the attacker failing to achieve reference targets. In its simplest form, $Q_{t'|t}$ is a $(p+1) \times (p+1)$ matrix with all zero entries except for a scalar weight $\beta_{t'|t} \in [0, 1]$ at (2,2). In this case, $Q_{t'|t}$ simply picks out the $(t' \mid t)$ element:

$$(\mathbf{y}_{t'|t} - \mathbf{y}^\dagger_{t'|t})^\top Q_{t'|t}(\mathbf{y}_{t'|t} - \mathbf{y}^\dagger_{t'|t}) = \beta_{t'|t}(y_{t'|t} - y^\dagger_{t'|t})^2. \tag{6}$$

For simplicity, we use $\|\mathbf{y}_{t'|t} - \mathbf{y}^\dagger_{t'|t}\|^2_{Q_{t'|t}}$ to denote (6). Critically, by setting the weights $\beta_{t'|t}$ the attacker can express different patterns of attack. For example:

- If $\beta_{T|t} = 1$ for all $t = 0 \dots T-1$ and 0 otherwise, the adversary cares about the forecasts made at all times about the final time horizon T. In this case, it is plausible that the adversarial target $y^\dagger_{T|t} := y^\dagger_T$ is a constant w.r.t. t.
- If $\beta_{t+1|t} = 1$ for all t and 0 otherwise, the adversary cares about all the forecasts about "tomorrow."
- if $\beta_{t'|t} = 1, \forall t, t'$, the adversary cares about all predictions made at all times.

Obviously, the adversary can express more complex temporal attack patterns. The adversary can also choose $\beta_{t'|t}$ value in between 0 and 1 to indicate weaker importance of certain predictions.

The R matrix defines the second part of the adversary cost, namely how much control expenditure u_t the adversary has to exert. In the simplest case, we let R be a scalar $\lambda > 0$:

$$\|u_t\|^2_R := \lambda u_t^2. \tag{7}$$

We use T to denote the *prediction time horizon*: T is the last time index (expressed by t') to be predicted by the forecaster. We define the adversary's *expected quadratic cost* $J(u_{0:(T-1)})$ for action sequence $u_0, \dots, u_{T-1}$ by

$$\mathbb{E}_{w_{0:(T-1)}}\left[\sum_{t=1}^{T-1}\sum_{t'=t+1}^{T}\|\mathbf{y}_{t'|t} - \mathbf{y}^\dagger_{t'|t}\|^2_{Q_{t'|t}} + \sum_{t=0}^{T-1}\|u_t\|^2_R \,\middle|\, \mathbf{x}_0\right].$$

Since the environment dynamics can be stochastic, the adversary must seek *attack policies* $\phi_t : \mathbb{R}^{p+1} \mapsto \mathbb{R}$ to map the observed state $\mathbf{x}_t$ to an attack action:

$$u_t = \phi_t(\mathbf{x}_t) \quad \forall t. \tag{8}$$

Given an adversarial state attack problem $(F, C, \{\mathbf{y}^\dagger_{t'|t}\}, \{Q_{t'|t}\}, R, T)$, we formulate the optimal state attack as the following optimal control problem:

$$\min_{\phi_0, \cdots, \phi_{T-1}} J(u_{0:(T-1)}) \tag{9}$$

$$\text{s.t.}\mathbf{x}_0 \text{ given} \tag{10}$$

$$u_t = \phi_t(\mathbf{x}_t), \quad t = 0, 1, \cdots, T-1 \tag{11}$$

$$\mathbf{x}_{t+1} = F(\mathbf{x}_t, u_t, w_t), \quad t = 0, 1, \cdots, T-1 \tag{12}$$

$$\mathbf{y}_{t'|t} = C^{t'-t}\mathbf{x}_t, \quad \forall t' > t. \tag{13}$$

We next propose solutions to this control problem for linear F and nonlinear F, respectively. For illustrative purpose, we focus on solving the problem when the attack target is to change predictions for "tomorrows". This implies $\beta_{t'|t} = 0$ when $t' \ge t+2$. Under this assumption, $J(u_{0:(T-1)})$ has the following form:

$$\mathbb{E}_{w_{0:(T-1)}}\left[\sum_{t=1}^{T-1}\|C\mathbf{x}_t - \mathbf{y}^\dagger_{t+1|t}\|^2_{Q_{t+1|t}} + \sum_{t=0}^{T-1}\|u_t\|^2_R \,\middle|\, \mathbf{x}_0\right].$$

More attack targets are studied in the experiment section.

Solving Attacks Under Linear F

When the environment dynamics f is linear, the scalar environment state evolves as $x_{t+1} = \alpha_0 + \sum_{i=1}^{p}\alpha_i x_{t+1-i} + u_t + w_t$,where the coefficients $\alpha_0, \dots, \alpha_p$ in general can be different from the forecaster's $AR(p)$ model (2). We introduce the corresponding vector operation

$$\mathbf{x}_{t+1} = F(\mathbf{x}_t, u_t, w_t) := A\mathbf{x}_t + B(u_t + w_t), \tag{14}$$

where A has the same structure as C in (4) except each $\hat{\alpha}$ is replaced by α, and $B = (0, 1, 0, \dots, 0)^\top$. The adversary's attack problem (9) reduces to stochastic Linear Quadratic Regulator (LQR) with tracking, which is a fundamental problem in control theory (Kwakernaak and Sivan 1972). It is well known that such problems have a closed-form solution, though the specific solution for stochastic tracking is often omitted from the literature. In addition, the presence of a forecaster in our case alters the form of the solution. Therefore, for completeness we provide the solution in Algorithm 1.

The derivation is in Appendix. Once the adversarial control policies are computed, the optimal attack sequence is given by: $u_t = \phi_t(\mathbf{x}_t), \quad t = 0, 1, \cdots, T-1$. The astute reader will notice that, $u_{T-1} = \phi_{T-1}(\mathbf{x}_{T-1}) = 0$. This is to be expected: u_{T-1} affects $\mathbf{x}_T$, but $\mathbf{x}_T$ would only affect forecasts after the prediction time horizon T, which the adversary does not care. To minimize the control expenditure, the adversary's rational behavior is to set $u_{T-1} = 0$.

Algorithm 1: $LQR(F, C, \{\mathbf{y}^\dagger_{t'|t}\}, \{Q_{t'|t}\}, R, T)$

```
Input   : $(F, C, \{\mathbf{y}^\dagger_{t'|t}\}, \{Q_{t'|t}\}, R, T)$
$P_T = 0$;
$\mathbf{q}_T = 0$;
for $t = T-1, T-2, \cdots, 1$ do
    $P_t = CA^\top Q_{t+1|t} C + A^\top (I + \frac{1}{\lambda} P_{t+1} BB^\top)^{-1} P_{t+1} A$;
    $\mathbf{q}_t = -2C^\top Q_{t+1|t} \mathbf{y}^\dagger_{t+1|t} + A^\top \mathbf{q}_{t+1} - \frac{1}{\lambda + B^\top P_{t+1} B} A^\top P^\top_{t+1} BB^\top \mathbf{q}_{t+1}$;
end
for $t = 0, 1, \ldots, T-1$ do
    $\phi_t(\mathbf{z}) = -\frac{B^\top \mathbf{q}_{t+1} + 2B^\top P_{t+1} A\mathbf{z}}{2(\lambda + B^\top P_{t+1} B)}$
end
Output : $\phi_0(\cdot), \ldots, \phi_{T-1}(\cdot)$
```

Solving Attacks Under Non-Linear F

When f is nonlinear the optimal control problem (9) in general does not have a closed-form solution. Instead, we introduce an algorithm that combines Model Predictive Control (MPC) (Garcia, Prett, and Morari 1989; Kouvaritakis and Cannon 2015) as the outer loop and Iterative Linear Quadratic Regulator (ILQR) (Li and Todorov 2004) as the inner loop to find an approximately optimal attack. While these techniques are standard in the control community, to our knowledge our algorithm is a novel application of the techniques to adversarial learning.

The outer loop performs MPC, a common heuristic in nonlinear control. At each time $\tau = 0, 1, \ldots$, MPC performs planning by starting at $\mathbf{x}_\tau$, looking ahead l steps and finding a good control sequence $\{\phi_t\}_{t=\tau}^{\tau+l-1}$. However, MPC then carries out only the first control action u^*_τ. This action, together with the actual noise instantiation w_τ, drives the environment state to $\mathbf{x}_{\tau+1} = F(\mathbf{x}_\tau, u^*_\tau, w_\tau)$. Then, MPC performs the l-step planning again but starting at $\mathbf{x}_{\tau+1}$, and again carries out the first control action $u^*_{\tau+1}$. This process repeats. Formally, MPC iterates two steps: at time τ

1. Solve

$$\min_{\phi_{\tau:L(\tau)}} \mathbb{E} \sum_{t=\tau+1}^{L(\tau)+1} \|C\mathbf{x}_t - \mathbf{y}^\dagger_{t+1|t}\|^2_{Q_{t+1|t}} + \sum_{t=\tau}^{L(\tau)} \|\phi_t(x_t)\|^2_R$$

$$\text{s.t.} \mathbf{x}_\tau \text{ given}$$

$$\mathbf{x}_{t+1} = F(\mathbf{x}_t, \phi_t(u_t), w_t), t = \tau, \cdots, L(\tau), \quad (15)$$

The expectation is over $w_\tau, \cdots, w_{L(\tau)}$. Denote the solution by $\{\phi_t\}_{t=\tau}^{L(\tau)}$.

2. Apply $u^*_\tau = \phi_\tau(x_\tau)$ to the F system.

$L(\tau) = \min(\tau + l - 1, T - 2)$, which indicates that the size of the optimization in step 1 will decrease as τ approaches T. The repeated re-planning allows MPC to adjust to new inputs, and provides some leeway if $\{\phi_t\}_{t=\tau}^{L(\tau)}$ cannot be exactly solved, which is the case for our nonlinear F.

We now turn to the inner loop to approximately solve (15). There are two issues that make the problem hard: the expectation over noises $\mathbb{E}_{w_\tau, \cdots, w_{L(\tau)}}$, and the nonlinear F. To address the first issue, we adopt an approximation technique known as "nominal cost" in (Kouvaritakis and Cannon 2015). For planning we simply replace the random variables w with their mean, which is zero in our case. This heuristic removes the expectation, and we are left with the following deterministic system as an approximation to (15):

$$\min_{u_\tau, \cdots, u_{L(\tau)}} \sum_{t=\tau+1}^{L(\tau)+1} \|C\mathbf{x}_t - \mathbf{y}^\dagger_{t+1|t}\|^2_{Q_{t+1|t}} + \sum_{t=\tau}^{L(\tau)} \|u_t\|^2_R$$

$$\text{s.t.} \mathbf{x}_\tau \text{ given}$$

$$\mathbf{x}_{t+1} = F(\mathbf{x}_t, u_t, 0), \quad t = \tau, \cdots, L(\tau). \quad (16)$$

To address the second issue, we adopt ILQR (Li and Todorov 2004) in order to solve (16). The idea of ILQR is to linearize the system around a trajectory, and compute an improvement to the control sequence using LQR iteratively. We show the details in Appendix. We summarize the MPC+ILQR attack in Algorithm 2 and 3.

Algorithm 2: MPC

```
Input   : $F, C, \{\mathbf{y}^\dagger_{t'|t}\}, \{Q_{t'|t}\}, R, T, l, maxiter, tol$
for $t = 0, 1, \ldots, T-2$ do
    Input   : $x_t$
    $u_{t:\min(t+l-1, T-2)} \leftarrow ILQR(x_t, F, C, \{\mathbf{y}^\dagger_{t'|t}\}, \{Q_{t'|t}\}, R, \min(t+l+1, T), maxiter, tol)$;
    Output : $u_t$
end
```

A Greedy Control Policy as the Baseline State Attack Strategy

The optimal state attack objective (9) can be rewritten as a running sum of instantaneous costs. At time $t = 0, 1, \ldots$ the instantaneous cost involves the adversary's control expenditure u_t^2, the attack's immediate effect on the environment state $\mathbf{x}_{t+1}$ (see Figure 1), and consequently on all the forecaster's predictions made at time $t+1$ about time $t+2, \ldots, T$. Specifically, the expected instantaneous cost $g_t(\mathbf{x}_t, u_t)$ at time t is defined as:

$$\mathbb{E}_{w_t} \|CF(\mathbf{x}_t, u_t, w_t) - \mathbf{y}^\dagger_{t+2|t+1}\|^2_{Q_{t+2|t+1}} + \|u_t\|^2_R. \quad (17)$$

This allows us to define a *greedy control policy* ϕ^G, which is easy to compute and will serve as a baseline for state attacks. In particular, the greedy control policy at time t minimizes the instantaneous cost:

$$\phi^G_t(\mathbf{x}_t) \in \text{argmin}_u g_t(\mathbf{x}_t, u).$$

When F is linear, $\phi^G_t(\cdot)$ can be obtained in closed-form. We show the solution in Appendix D.

When f is nonlinear, we let noise $w_t = 0$ and solve the following nonlinear problem using numerical solvers:

$$\min_{u_t} \|CF(\mathbf{x}_t, u_t, 0) - \mathbf{y}^\dagger_{t+2|t+1}\|^2_{Q_{t+2|t+1}} + \|u_t\|^2_R. \quad (18)$$

Algorithm 3: ILQR

```
Input   : $x_0, F, C, \{\mathbf{y}^\dagger_{t'|t}\}, \{Q_{t'|t}\}, R, T, maxiter, tol$
Initialize $u_{0:T-2}$;
for $i = 0, 1, \ldots, maxiter$ do
    for $t = 0 : T-2$ do
        $\mathbf{x}_{t+1} = F(\mathbf{x}_t, u_t, 0)$;
        $D_uF_t = D_uF(\mathbf{x}_t, u_t, 0)$;
        $D_xF_t = D_xF(\mathbf{x}_t, u_t, 0)$
    end
    $P_{T-1} = C^\top Q_{T|T-1} C$;
    $\mathbf{q}_{T-1} = 2C^\top Q_{T|T-1}(C\mathbf{x}_{T-1} - \mathbf{y}^\dagger_{T|T-1})$;
    for $s = T-2, \ldots, 1$ do
        $P_s = C^\top Q_{s+1|s} C + D_\mathbf{x}F_s^\top (I + \frac{1}{\lambda} P_{s+1} D_uF_s D_uF_s^\top)^{-1} P_{s+1} D_\mathbf{x}F_s^\top$;
        $\mathbf{q}_s = 2C^\top Q_{s+1|s}(C\mathbf{x}_s - \mathbf{y}^\dagger_{s+1|s}) + D_\mathbf{x}F_s^\top \mathbf{q}_{s+1} - \frac{(D_uF_s^\top P_{s+1} D_\mathbf{x}F_s)^\top (D_uF_s^\top \mathbf{q}_{s+1} + 2\lambda u_s)}{\lambda + D_uF_s^\top P_{s+1} D_uF_s}$
    end
    for $s = 0, T-2$ do
        $\delta u_s = -\frac{2D_uF_s^\top P_{s+1} D_\mathbf{x}F_s \delta\mathbf{x}_s + D_uF_s^\top \mathbf{q}_{s+1} + 2\lambda u_s}{2(\lambda + D_uF_s^\top P_{s+1} D_uF_s)}$;
        $\delta\mathbf{x}_{s+1} = D_\mathbf{x}F_s \delta\mathbf{x}_s + D_uF_s \delta u_s$
    end
    if $\|\delta u_{0:T-2}\|^2/(T-1) < tol$ then
        Break;
    end
    $u_{0:T-2} \leftarrow u_{0:T-2} + \delta u_{0:T-2}$;
end
Output : $u_{0:T-2}$
```

Black-Box Attack via System Identification

We now consider black-box attack setting where the environment dynamics f and forecaster's model C are no longer known to the attacker. Both the environment forecast models are allowed to be nonlinear. The attacker will perform system identification (Dean et al. 2017), and solve LQR as an inner loop and MPC as an outer loop.

The attacks picks an estimation model order p for both the environment and forecaster. It also picks a buffer length b. In the first $b+p-1$ iterations, the attacker does no attack but collects observations on the free-evolving environment and the forecasts. Then, in each subsequent iteration the attacker estimates a linear environmental model $\hat{\boldsymbol{a}}$ and linear forecast model $\hat{\boldsymbol{c}}$ using a rolling buffer of previous $b+p-1$ iterations. The buffer produces b data points for the attacker to use MLE to solve $p+1$ unknowns in environment model.The attacker then uses MPC and LQR to design an attack action.

The attacker use linear models to estimate both the environmental dynamics and forecast model. At time t, he environmental dynamics is estimated over the environmental state value $x_{t-b-p+1}, \ldots, x_t$ and action sequences $u_{t-b}, \ldots, u_{t-1}$:

$$\min_{a_{0:p}} \sum_{t=t-b}^{t-1} (a_0 + \sum_{i=1}^{p} a_i x_{t+1-i} + u_t - x_{t+1})^2, \tag{19}$$

we use $\hat{\boldsymbol{a}}(t,b)$, a $p+1$-dimensional column vector to denote the minimum point of (19). Due to the nature of attacking autoregressive models, $B = (0, 1, 0, \ldots, 0)^\top \in \mathbb{R}^{(p+1)\times 1}$ is always known to the attacker.

We use $\hat{c}_0, \hat{c}_1, \ldots, \hat{c}_p$ to denote the estimation of forecast model and let

$$C(\hat{c}_{0:p}) = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ \hat{c}_0 & \hat{c}_1 & \hat{c}_2 & \cdots & \hat{c}_{p-1} & \hat{c}_p \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix}. \tag{20}$$

to denote the corresponding forecast matrix. Let $\Lambda(t_1, t_2)$ denote the set of prediction indices which are visible to the attacker: $\Lambda(t_1, t_2) = \{(t, t') \mid t_1 \le t \le t_2, y_{t'|t} \text{ is visible to the attacker}\}$. At time t, the forecast model is estimated over the visible forecasts: $y_{t'|t}, (t, t') \in \Lambda(t-b, t)$:

$$\min_{c_{0:p}} \sum_{(t,t') \in \Lambda(t-b,t)} (B^\top C^{t'-t} \mathbf{x}_t - y_{t'|t})^2, \tag{21}$$

we use $\hat{\boldsymbol{c}}(t,b)$, a $p+1$-dimensional column vector to denote the minimum point of (21). If the attacker only observes sufficient predictions for "tomorrows", i.e. $\Lambda(t_1, t_2) = \{(t, t') \mid t_1 \le t \le t_2, t' = t+1\}$, then $\hat{c}_0, \hat{c}_1, \ldots, \hat{c}_p$ is the OLS solution to (21). However, for more complex prediction pattern, (21) might involve polynomials of $c_0, c_1, \ldots, c_p$.

We can summarize the proposed black-box attack method in Algorithm 4.

Algorithm 4: System identification attack

```
Input   : model order $p$, buffer size $b$, time step of MPC $l$
for $t = 0, 1, \ldots, b+p-2$ do
    Input   : $x_t$, $y_{t'|t}$
    Output : $u_t = 0$
end
for $t = b+p-1, \ldots, T-1$ do
    update $\hat{\boldsymbol{a}}(t,b)$ by (19);
    update $\hat{\boldsymbol{c}}(t,b)$ by (21);
    $\phi_{0:\min(t+l,T)-1} \leftarrow LQR(\{\hat{\boldsymbol{a}}(t,b), B\}, C(\hat{\boldsymbol{c}}(t,b)), \{\mathbf{y}^\dagger_{t'|t}\}, \{Q_{t'|t}\}, R, \min(t+l+1,T)-t)$;
    Output : $u_t \leftarrow \phi_0(x_t)$
    Input   : $x_{t+1}$, $y_{t'|t}$
end
```

Experiments

We now demonstrate the effectiveness of control-based white-box attacks on time series forecast problems. We compare the optimal attacks computed by LQR (for linear f), MPC+iLQR (for nonlinear f), black-box attack, greedy attacks, and the no-attack baseline. While the attack actions were optimized under an expectation over random noise w (c.f. (9)), in the experiments we report the *actual realized cost* based on the noise instantiation that the algorithm experienced:

$$\sum_{t=1}^{T-1}\sum_{t'=t+1}^{T} \|C^{t'-t}\mathbf{x}_t - \mathbf{y}^\dagger_{t'|t}\|^2_{Q_{t'|t}} + \sum_{t=0}^{T-1}\|u_t\|^2_R \qquad (22)$$

where the noise sequence $\{w_t\}_{t=0}^{T-2}$ is incorporated implicitly in $\mathbf{x}_t$, together with the actual attack sequence $\{u_t\}_{t=0}^{T-1}$. To make the balance between attack effect (the quadratic terms involving Q) and control expenditure (the term involving R) more interpretable, we let $R := \lambda = \tilde{\lambda}\sum_{t=1}^{T-1}\sum_{t'=t+1}^{T}\beta_{t'|t}/T$

The Effect of $Q_{t'|t}$ on Attack

In our first synthetic example we demonstrate **the adversary's ability to target different parts of the forecasts via** Q, the quadratic coefficient in cost function. Figure 2 illustrates three choices of attack targets Q: attack "tomorrow", "last day" and "all predictions".

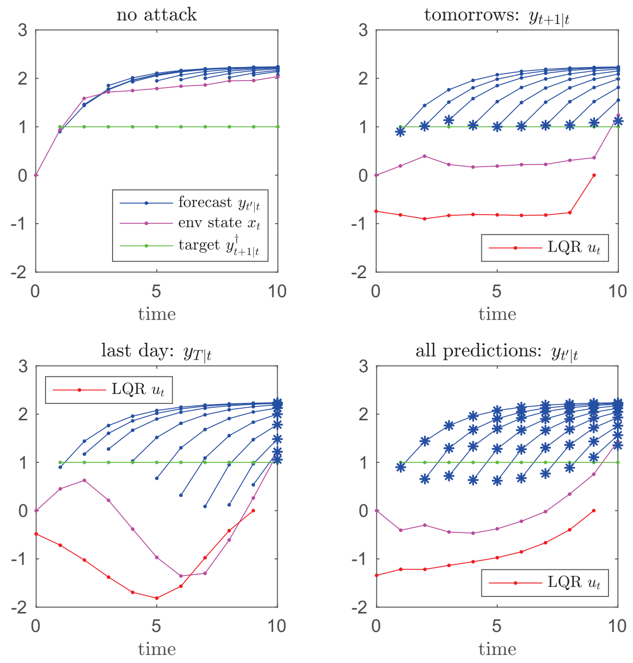

Figure 2: LQR solution on three attack patterns Q. y-axis shows the value of x and u. Each blue forecast curve beginning at time t shows the sequence $\{y_{t'|t-1}\}_{t'=t}^{T}$. The pattern of attack defined by the corresponding Q is highlighted with $*$ on the forecast curves.

For simplicity, we let all adversarial reference targets $y^\dagger_{t'|t}$ be the constant 1. We let the environment evolve according to an $AR(1)$ model: $x_{t+1} = 1 + 0.5x_t + w_t$. We let the noise $w_t \sim N(0, 0.1^2)$ and the initial state $x_0 = 0$. We simulate the case where the forecaster has only an approximation of the environment dynamics, and let the forecaster's model be $x_{t+1} = 0.9 + 0.6x_t$ which is close to, but different from, the environment dynamics. For illustrative purpose, we set the prediction time horizon $T = 10$. Recall that the attacker can change the environment state by adding perturbation u_t: $x_{t+1} = 1 + 0.5x_t + u_t + w_t$. We set $\tilde{\lambda} = 0.1$.

We run LQR and compute the optimal attack sequences u under each Q scenario. They are visualized in Figure 2. Each attack is effective: the blue *'s are closer to the green target line on average, compared to where they would be in the upper-left no-attack panel. Different target selection Q will affect the optimal attack sequence.

Comparing LQR vs. Greedy Attack Policies

We now show **the LQR attack policy is better than the greedy attack policy**. We let the environment evolves by an $AR(3)$ model: $x_{t+1} = f(x_t, x_{t-1}, x_{t-2}, w_t) = 0.4x_t - 0.3x_{t-1} - 0.7x_{t-2} + w_t$, $w_t \sim N(0, 0.1^2)$. The initial values are $x_0 = 10$, $x_{-1} = x_{-2} = 0$, prediction horizon $T = 15$. This environment dynamic is oscillating around 0.

We let the forecaster's model be: $x_{t+1} = 0.41x_t - 0.29x_{t-1} - 0.68x_{t-2}$. Q is "tomorrows". The attacker wants the forecaster to predict a sequence oscillating with smaller amplitude. $y^\dagger_{t+1|t}$ is set as following: we simulate $x_{t+1} = f(x_t, x_{t-1}, x_{t-2}, 0)$, then, let the attack reference target be $y^\dagger_{t|t-1} = 0.5x_t, t = 2, \cdots, T$. We set $\tilde{\lambda} = 0.1$.

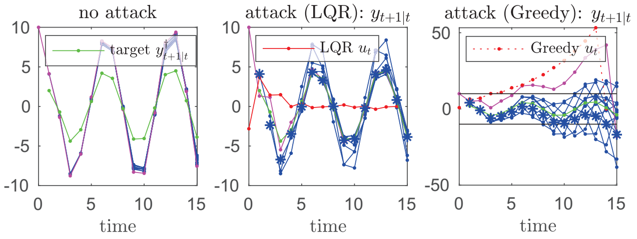

Figure 3: LQR vs. Greedy attacks. The black horizontal lines in the right plot mark the vertical axis range of the middle plot.

We run LQR and Greedy, respectively, to solve this attacking problem. We generate 50 trials with different noise sequences, see Figure 3. Interestingly, LQR drives the predictions close to the attack target, while Greedy *diverges*. The mean actual realized cost of no attack, LQR attack and Greedy attack are $133.9, 11.5, 1492$, respectively. The standard errors are $1.20, 0.02, 12.99$. We perform a paired t-test on LQR vs. Greedy. The null hypothesis of equal mean is rejected with $p = 4 \times 10^{-61}$. This clearly demonstrate the myopic failure of the greedy policy.

MPC+ILQR attack on US Real GNP

This real world data (Tiao and Tsay 1994) models the growth rate of quarterly US real GNP from the first quarter of 1947 to the first quarter of 1991. We use the GNP data to evaluate MPC+iLQR and Greedy, which attack the "last day." The environment's nonlinear threshold model dynamics is:

$$x_{t+1} = \begin{cases} -0.015 - 1.076x_t + w_{1,t} & (x_t, x_{t-1}) \in X_1 \\ -0.006 + 0.630x_t - 0.756x_{t-1} + w_{2,t} & (x_t, x_{t-1}) \in X_2 \\ 0.006 + 0.438x_t + w_{3,t} & (x_t, x_{t-1}) \in X_3 \\ 0.004 + 0.443x_t + w_{4,t} & (x_t, x_{t-1}) \in X_4, \end{cases}$$

where $X_1 = \{(x_t, x_{t-1}) \mid x_t \le x_{t-1} \le 0\}, X_2 = \{(x_t, x_{t-1}) \mid x_t > x_{t-1}, x_{t-1} \le 0\}, X_3 = \{(x_t, x_{t-1}) \mid x_t \le x_{t-1}, x_{t-1} > 0\}, X_4 = \{(x_t, x_{t-1}) \mid x_t > x_{t-1} > 0\}$, $w_{1,t} \sim N(0, 0.0062^2), w_{2,t} \sim N(0, 0.0132^2), w_{3,t} \sim N(0, 0.0094^2), w_{4,t} \sim N(0, 0.0082^2)$. We let $T = 10$, $x_0 = 0.0065$ (according to (Tiao and Tsay 1994)), $x_{-1} = 0$. The forecaster's model is $x_{t+1} = 0.0041 + 0.33x_t + 0.13x_{t-1}$ (according to (Tiao and Tsay 1994)). The attacker can change state value by adding perturbation. The attack target is to drive forecaster's predictions $y_{T|t}, t = 1, \cdots, T-1$ to be close to 0.01. We let $\tilde{\lambda} = 0.001$. MPC+iLQR and Greedy are used to solve this problem. The time step of MPC is set to be $l = 5$. Inside the MPC loop, the stopping condition of iLQR is $tol = 10^{-4}$. The maximum iteration of iLQR is set to be 1000. For Greedy, we use the default setting for the *lsqnonlin* solver in Matlab (Coleman, Branch, and Grace 1999) except that we do provide the gradients

We again run 50 trials, the last one is shown in Figure 4. The mean actual realized cost of no attack, MPC+iLQR attack and Greedy attack are $(6.87, 3.03, 3.23) \times 10^{-4}$ respectively. The standard errors are $(1.40, 0.18, 0.19) \times 10^{-5}$ respectively. The null hypothesis of equal mean is rejected with $p = 6 \times 10^{-65}$ by a paired t-test. As an interesting observation, in the beginning MPC+iLQR adopts a larger attack than Greedy; at time $t = 4$, MPC+iLQR adopts a smaller attack than Greedy, but drives $y_{T|5}$ closer to 0.01. This shows the advantage of looking into future. Since Greedy only focus on current time, it ignores how the attack will affect the future.

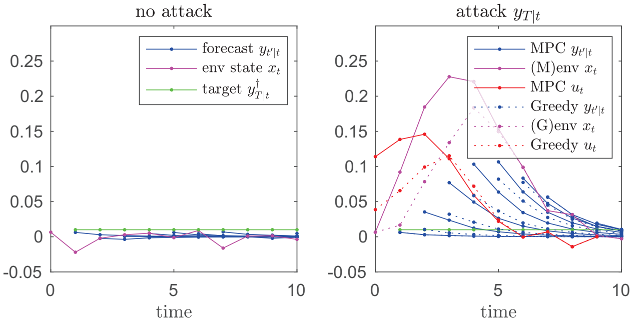

Figure 4: MPC+iLQR and Greedy attack on GNP data. MPC+iLQR: $u_4 = 0.0590, y_{T|5} = 0.0084$; Greedy: $u_4 = 0.0719, y_{T|5} = 0.0079$. $x^\dagger_{T|5} = 0.01$.

Using System Identification in Black-box Attack

We now show **system identification can perform black-box attack effectively**. We compare the system identification attack to an oracle, who has full information of both the environmental dynamics and forecast model. The oracle use MPC+ILQR to attack the forecast.

We use the same dynamic in (Fan and Yao 2008) but we change the noise to be $w_t \sim N(0, 0.1^2)$. The dynamic is $x_{t+1} = 2x_t/(1 + 0.8x_t^2) + w_t$. We let $x_0 = 3, T = 50$. We simulate this dynamical system to $t = 50$ and get a sample sequence from this dynamical system. The forecaster's $AR(1)$ model C is estimated from this sequence. We introduce an attacker who can add perturbation u_t to change the state value: $x_{t+1} = 2x_t/(1 + 0.8x_t^2) + u_t + w_t$. Q is "tomorrow". The attack target is set to be $y^\dagger_{t+1|t} = 2$. We let $\tilde{\lambda} = 0.01$.

Both the attacker and the oracle do nothing but observe the x_t and $y_{t'|t}$ at time $t = 0, \ldots, b + p - 2$, and attack the forecast at time $t = b + p - 1, \ldots, T - 2$. For system identification, we let $b = 15, l = 5, p = 3$. For the oracle, the time step of MPC is set to be $l = 10$. Inside the MPC loop, the stopping condition of ILQR is $tol = 10^{-4}$. The maximum iteration of iLQR is set to be 1000.

We run 100 trials, the last one is shown in Figure 5. The mean actual realized cost of system identification attack and oracle MPC+ILQR attack are $4.20, 1.43$ respectively. Even though the cost of System identification attack is larger than that of the oracle, it is evident from Figure 5 that the attacker can quickly force the forecasts (blue) to the attack target (green) after t=25; the chaotic period between t=20 and t=25 is the price to pay for system identification.

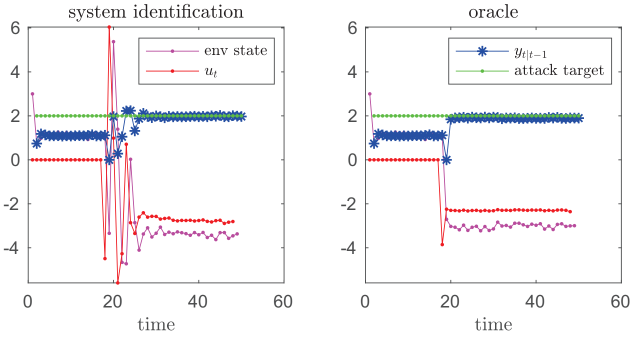

Figure 5: System identification and MPC+ILQR oracle attack

Conclusion

In this paper we formulated adversarial attacks on autoregressive model as optimal control. This sequential attack problem differs significantly from most of the batch attack work in adversarial machine learning. In the white-box setting, we obtained closed-form LQR solutions when the environment is linear, and good MPC approximations when the environment is nonlinear. In the black-box setting, we propose a method via system identification then perform MPC. We demonstrated their effectiveness on synthetic and real data.

Acknowledgment

We thank Laurent Lessard, Yuzhe Ma, and Xuezhou Zhang for helpful discussions. This work is supported in part by NSF 1836978, 1545481, 1704117, 1623605, 1561512, the

MADLab AF Center of Excellence FA9550-18-1-0166, and the University of Wisconsin.

References

Alfeld, S.; Zhu, X.; and Barford, P. 2016. Data poisoning attacks against autoregressive models. In *The Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*.

Alfeld, S.; Zhu, X.; and Barford, P. 2017. Explicit defense actions against test-set attacks. In *The Thirty-First AAAI Conference on Artificial Intelligence (AAAI)*.

Biggio, B., and Roli, F. 2017. Wild patterns: Ten years after the rise of adversarial machine learning. *CoRR* abs/1712.03141.

Biggio, B.; Corona, I.; Nelson, B.; Rubinstein, B. I.; Maiorca, D.; Fumera, G.; Giacinto, G.; and Roli, F. 2014. Security evaluation of support vector machines in adversarial environments. In *Support Vector Machines Applications*. Springer. 105–153.

Biggio, B.; Nelson, B.; and Laskov, P. 2012. Poisoning attacks against support vector machines. *arXiv preprint arXiv:1206.6389*.

Box, G. E.; Jenkins, G. M.; Reinsel, G. C.; and Ljung, G. M. 2015. *Time series analysis: forecasting and control*. John Wiley & Sons.

Coleman, T.; Branch, M. A.; and Grace, A. 1999. Optimization toolbox. *For Use with MATLAB. User's Guide for MATLAB 5, Version 2, Relaese II*.

Dean, S.; Mania, H.; Matni, N.; Recht, B.; and Tu, S. 2017. On the sample complexity of the linear quadratic regulator. *arXiv preprint arXiv:1710.01688*.

Fan, J., and Yao, Q. 2008. *Nonlinear time series: nonparametric and parametric methods*. Springer Science & Business Media.

Garcia, C. E.; Prett, D. M.; and Morari, M. 1989. Model predictive control: theory and practice—a survey. *Automatica* 25(3):335–348.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Hamilton, J. D. 1994. *Time series analysis*, volume 2. Princeton university press Princeton, NJ.

Joseph, A. D.; Nelson, B.; Rubinstein, B. I. P.; and Tygar, J. D. 2018. *Adversarial Machine Learning*. Cambridge University Press. in press.

Jun, K.-S.; Li, L.; Ma, Y.; and Zhu, X. 2018. Adversarial attacks on stochastic bandits. In *Advances in Neural Information Processing Systems (NIPS)*.

Kouvaritakis, B., and Cannon, M. 2015. Stochastic model predictive control. *Encyclopedia of Systems and Control* 1350–1357.

Kwakernaak, H., and Sivan, R. 1972. *Linear optimal control systems*, volume 1. Wiley-Interscience New York.

Lee, E. B., and Markus, L. 1967. Foundations of optimal control theory. Technical report, Minnesota Univ Minneapolis Center For Control Sciences.

Lessard, L.; Zhang, X.; and Zhu, X. 2018. An optimal control approach to sequential machine teaching. *arXiv preprint arXiv:1810.06175*.

Li, W., and Todorov, E. 2004. Iterative linear quadratic regulator design for nonlinear biological movement systems. In *ICINCO (1)*, 222–229.

Liu, C.; Li, B.; Vorobeychik, Y.; and Oprea, A. 2017. Robust linear regression against training data poisoning. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, 91–102. ACM.

Lowd, D., and Meek, C. 2005. Adversarial learning. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, 641–647. ACM.

Ma, Y.; Jun, K.-S.; Li, L.; and Zhu, X. 2018. Data poisoning attacks in contextual bandits. In *Conference on Decision and Game Theory for Security (GameSec)*.

Ma, Y.; Zhang, X.; Sun, W.; and Zhu, J. 2019. Policy poisoning in batch reinforcement learning and control. In *Advances in Neural Information Processing Systems*, 14543–14553.

Ma, Y.; Zhu, X.; and Hsu, J. 2019. Data poisoning against differentially-private learners: Attacks and defenses. *arXiv preprint arXiv:1903.09860*.

Nguyen, A.; Yosinski, J.; and Clune, J. 2015. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 427–436.

Papernot, N.; McDaniel, P.; Swami, A.; and Harang, R. 2016. Crafting adversarial input sequences for recurrent neural networks. In *MILCOM 2016-2016 IEEE Military Communications Conference*, 49–54. IEEE.

Recht, B. 2018. A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*.

Tiao, G. C., and Tsay, R. S. 1994. Some advances in nonlinear and adaptive modelling in time-series. *Journal of forecasting* 13(2):109–131.

Vorobeychik, Y., and Kantarcioglu, M. 2018. Adversarial machine learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning* 12(3):1–169.

Wang, Y., and Chaudhuri, K. 2018. Data poisoning attacks against online learning. *arXiv preprint arXiv:1808.08994*.

Zhang, X., and Zhu, X. 2019. Online data poisoning attack. *arXiv preprint arXiv:1903.01666*.

Zhu, X. 2018. An optimal control view of adversarial machine learning. *arXiv preprint arXiv:1811.04422*.