Adversarial Sequential Decision Making

Goran Radanović, Adish Singla, Wen Sun, Xiaojin Zhu

International Joint Conference on AI (IJCAI) 2022
Adversarial Attacks on AI

$\mathbf{x}$

“panda”
57.7% confidence

+ 0.007 $\times$

sign($\nabla_{\mathbf{x}} J(\theta, \mathbf{x}, y)$)

“nematode”
8.2% confidence

= $\mathbf{x} + \varepsilon \text{sign}(\nabla_{\mathbf{x}} J(\theta, \mathbf{x}, y))$

“gibbon”
99.3% confidence

[Goodfellow et al., 2015]

[Sharif et al., 2016]
Adversarial Attacks on ML

Learning to predict

Dataset → Algorithm → Prediction

Influencing prediction

Dataset → Attacker → Algorithm → Prediction
Accounting for Decisions

Attacks on Driving Systems
Accounting for Decisions

Attacks on Driving Systems

Attacks on Conversational AI

Microsoft Created a Twitter Bot to Learn From Users. It Quickly Became a Racist Jerk.
From Prediction to Decisions

Learning to decide

Dataset -> Algorithm -> Prediction -> Decision

Influencing decision

Dataset -> Attacker -> Algorithm -> Prediction -> Decision

Credit to Dylan Foster for ML vs. Decision Making distinction. See link.
Trustworthy Decision Making

Reinforcement Learning

Algorithmic foundations

Scalable algorithms

Next grand challenge!

Early works

2010s-2020s

[Resnick, 1964]

[Williams, 1992]

[Tesauro, 1995]

[Vynials et al., 2019]

[Baker et al., 2020]

DeepMind.com

[Author: Dllu]

Trustworthy decision making
Sequential Decision Making

Agent

Follow policy \( \pi \)

Observe state \( s_t \)

Take action \( a_t \)

Receive reward \( r_t \)

Environment

Update state \( s_{t+1} \)

Model \( \mathcal{M} \)

Maximize performance
Adversarial Sequential Decision Making

Agent

Observe state

Take action

Receive reward

Environment

Model $\mathcal{M}$

Update state

Manipulate Interaction

Maximize performance
Attack Modalities: Test-Time Attacks

Follow a learned policy

Observe state

Take action

Receive reward

Environment

Update state

E.g. manipulate observations to minimize performance

Example
Attack Modalities: Training-Time Attacks

Learn a policy

Observe state

Take action

Environment

Update state

Receive reward

Example

E.g., manipulate rewards & transitions to force a target policy
Defenses Against Adversarial Attacks

Learn a robust policy

Environment

Model $\mathcal{M}$

Take action

Receive reward

Update state

Observe state

Maximize performance

Manipulate Interaction
Outline

• Preliminaries
• Test-time Attacks and Defenses in RL
• Training-time Attacks in RL
• Training-time Defenses in RL
• Adversarial Attacks in Multi-agent RL
• Concluding Remarks
Outline

• Preliminaries
  • Test-time Attacks and Defenses in RL
  • Training-time Attacks in RL
  • Training-time Defenses in RL
  • Adversarial Attacks in Multi-agent RL
• Concluding Remarks
Markov Decision Processes

MDP $\mathcal{M} = (S, A, P, R, \gamma, \mu)$

- $P: S \times A \rightarrow \Delta(S)$
- $R: S \times A \rightarrow \mathbb{R}$
- $\gamma \in [0, 1)$
- $\mu \in \Delta(S)$

- Stochastic stationary policy $\pi: S \rightarrow \Delta(A)$

$$\max_\pi \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) \mid s_0 \sim \mu, a_t \sim \pi(\cdot \mid s_t), s_{t+1} \sim P(\cdot \mid s_t, a_t) \right]$$
Value function $V^\pi : S \to \mathbb{R}$

\[ V^\pi(s) = \mathbb{E}\left[ \sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) \mid s_0 = s, \pi \right] \]
Value Functions

Value function $V^\pi : S \rightarrow \mathbb{R}$

$$V^\pi(s) = \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) \mid s_0 = s, \pi \right]$$

Note that the optimization problem is ...

$$\max_\pi V^\pi(\mu) = \sum_s \mu(s) \cdot V^\pi(s)$$
Value Functions

Value function $V^\pi : S \to \mathbb{R}$

$$V^\pi(s) = \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) \mid s_0 = s, \pi \right]$$

How do we find $V^\pi$?

$$V^\pi(s) = R(s, \pi) + \gamma \cdot \sum_{s'} P(s' \mid s, \pi) \cdot V^\pi(s')$$

Bellman equation
Value Functions

Value function $V^\pi : S \to \mathbb{R}$

$$V^\pi(s) = \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) \mid s_0 = s, \pi \right]$$

State-action value function $Q^\pi : S \times A \to \mathbb{R}$

$$Q^\pi(s, a) = \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) \mid s_0 = s, a_0 = a, \pi \right]$$
Value Functions

Value function $V^\pi : S \to \mathbb{R}$

\[ V^\pi(s) = \mathbb{E}\left[ \sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) \mid s_0 = s, \pi \right] \]

State-action value function $Q^\pi : S \times A \to \mathbb{R}$

\[ Q^\pi(s, a) = \mathbb{E}\left[ \sum_{t=0}^{\infty} \gamma^{t-1} \cdot R(s_t, a_t) \mid s_0 = s, a_0 = a, \pi \right] \]

Advantage function:

\[ A^\pi(s, a) = Q^\pi(s, a) - V^\pi(s) \]
Bellman optimality operator

\[(\mathcal{T}Q)(s, a) = R(s, a) + \gamma \sum_{s', a'} P(s'|s, a) \max_{a'} Q(s', a')\]

If \(Q^*\) satisfies \(\mathcal{T}Q^* = Q^*\), then

\[\pi^*(a|s) = 1.0 \text{ s.t. } a \in \arg\max_{a'} Q^*(s, a')\]

is optimal.
Finding Optimal Policy

• Planning in MDPs: $P$ and $R$ are given
  – Policy iteration: policy evaluation + policy improvement
  – Q-value iteration: calculate $Q^*$

• Reinforcement learning
  – Policy gradient
  – Q-learning: learn $Q^*$
Policy Gradient

• Parametric policy $\pi_\theta(a|s)$

• Gradient update rule: $\theta_{k+1} = \theta_k + \eta \cdot \nabla_\theta V^{\pi_\theta}(\mu)|_{\theta=\theta_k}$
Policy Gradient

- Parametric policy $\pi_\theta(a|s)$

- Gradient update rule:  
  $$\theta_{k+1} = \theta_k + \eta \cdot \nabla_\theta V^{\pi_\theta}(\mu) |_{\theta = \theta_k}$$

- Policy gradient theorem:
  \[
  \nabla_\theta V^{\pi_\theta}(\mu) = \frac{1}{1 - \gamma} \cdot \mathbb{E}_{s,a \sim \pi_\theta} [A^{\pi_\theta}(s,a) \cdot \nabla_\theta \log \pi_\theta(a|s)]
  \]
  \[
  a^{\pi_\theta}(s,a) = (1 - \gamma) \cdot \mathbb{E} \left[ \sum_{t=0}^{\infty} \gamma^{t-1} \cdot \mathbb{I}[s_t = s,a_t = a] | \mu, \pi \right]
  \]
• Preliminaries
  • Test-time Attacks and Defenses in RL
  • Training-time Attacks in RL
  • Training-time Defenses in RL
  • Adversarial Attacks in Multi-agent RL
• Concluding Remarks
• Goodfellow et al., Explaining and Harnessing Adversarial Examples, 2015.
• Sharif et al., Accessorize to a Crime: Real and Stealthy Attacks on State-of-the-Art Face Recognition, 2016.