Nesterov Momentum Adversarial Perturbations in the Deep Reinforcement Learning Domain

Ezgi Korkmaz

Abstract
Deep reinforcement learning algorithms achieved significant success in the last five years and gave birth to a new research area. Currently, deep reinforcement learning algorithms have been deployed in many different fields from resource management to deep neural architecture design. Exploring the generalization capabilities of deep reinforcement learning algorithms still remains an active area of research. One way to assess the generalization of deep reinforcement learning agents lies in investigating their reactions to adversarial perturbations. In this paper, we propose a new Nesterov momentum based optimization to find adversarial perturbations for the deep reinforcement learning domain. We show in our experiments our Nesterov momentum based approach achieves state-of-the-art results in various games from the Atari environment. We believe our proposed approach can be an important initial step in the robustification of deep reinforcement learning agents.

1. Introduction
Deep Neural Networks (DNN)s have gained significant momentum both in research and applications, and are currently utilized in many different domains such as natural language processing (Sutskever et al. (2014), speech recognition Hannun et al. (2014), image recognition Krizhevsky et al. (2012), and self learning systems (Mnih et al., 2015, Schulman et al. (2017), Lillicrap et al. (2015)). Reinforcement learning algorithms, in particular, have seen dramatic recent improvements with the introduction of DNNs as function approximators (Mnih et al., 2015). Following this initial work, deep reinforcement learning has been applied in a variety of areas including: robotics (Gu et al. (2017), Kalashnikov et al. (2018)), autonomous driving Dosovitsky et al. (2017), auction bidding Wang et al. (2017), deep neural network architecture design Baker et al. (2016), network system control (Jay et al. (2019),Chinchali et al. (2018)), grid operation and security (Duan et al. (2019), Huang et al. (2019)), financial trading Noonan (2017), blockchain protocol security Hou et al. (2019), natural language processing (He et al. (2016), Jaques et al. (2017), Wang et al. (2018)), medical treatment and diagnosis (Yauney & Pratik (2018), Suchi (2018), Popova et al. (2018), Thananjeyan et al. (2017), Daochang & Jiang (2018), Ghesu et al. (2017)).

Due to the wide applicability of deep reinforcement learning, understanding the robustness and generalization properties of these algorithms is crucial. In particular, a key question is how these algorithms react in the presence of an adversary. Initially, Szegedy et al. (2014) showed that DNNs used for image classification can be made to fail by adding visually imperceptible perturbations to the input image. Further research by Goodfellow et al. (2015) explained the presence of these imperceptible adversarial perturbations by arguing that DNNs learn approximately linear functions. The authors also introduced a new efficient method to compute adversarial perturbations, and demonstrated that including these perturbations in the cost function for DNN training improves robustness. Later Madry et al. (2017) showed that computing optimal adversarial perturbations is a key step in their robust optimization approach to training DNNs resistant to adversarial perturbations. In particular, they demonstrated that better algorithms for computing adversarial examples used in training leads to more robust networks.

We believe adversarial formulations are a way to assess generalization capabilities of deep reinforcement learning agents and are an initial step towards building robust and reliable agents. For these reasons, in this work we focus on adversarial formulations in deep reinforcement learning and make the following contributions:

• We propose Nesterov momentum based optimization to compute adversarial perturbations in the deep reinforcement learning domain.
• We run multiple experiments in the Atari environment.
in various games to compare state-of-the-art adversarial formulations and our proposed momentum based approach.

- We demonstrate the momentum based approach can reach higher impact levels with a lower bound on the adversarial perturbation.

2. Related Work and Background

2.1. Crafting Adversarial Perturbations

Szegedy et al. (2014) proposed to minimize the distance between the original image and adversarially produced image to create adversarial perturbations. The authors used box-constrained L-BFGS to solve this optimization problem.

$$\begin{equation}
\arg\min_{x_{adv}} = c \cdot \|x_{adv} - x\| - J(x_{adv}, y)
\end{equation}$$

Here $x$ is the input, $y$ is the output label, and $J(x, y)$ is the cost function for image classification. Goodfellow et al. (2015) introduced the fast gradient method (FGM)

$$x_{adv} = x + \epsilon \cdot \frac{\nabla_x J(x, y)}{||\nabla_x J(x, y)||_p},$$

for crafting adversarial examples in image classification by taking the gradient of the cost function $J(x, y)$ used to train the neural network in the direction of the input. Kurakin et al. (2016) proposed an iterative fast gradient method (FGM) by applying FGM multiple times with small step size and clipping the pixel values of intermediate updates after each step to guarantee they are in the $\epsilon$-ball.

Dong et al. (2018) suggested an iterative fast gradient method employing a momentum approach called MI-FGSM to create adversarial perturbations in the image classification domain. Furthermore, the authors increase the transferability of the adversarial examples by utilizing MI-FGSM, and an ensemble of models to produce adversarial perturbations.

Finally, Carlini & Wagner (2017) introduced targeted attacks in the image classification domain based on distance minimization between the adversarial image and the original image while targeting a particular label. In the deep reinforcement learning domain the Carlini & Wagner (2017) formulation is

$$\begin{equation}
\min_{s_{adv} \in D_{x,y}(s)} \|s_{adv} - s\|_p
\end{equation}$$

subject to $a^\star(s) \neq a^\star(s_{adv}),$

where $s$ is the unperturbed input, $s_{adv}$ is the adversarially perturbed input, $a^\star(s)$ is the action taken in the unperturbed state, and $a^\star(s_{adv})$ is the action taken in the adversarial state. This formulation is basically the minimization of the distance to the adversarial state constrained to states leading to sub-optimal actions as determined by the $Q$-network.

2.2. Adversarial Deep Reinforcement Learning

Huang et al. (2017) and Kos & Song (2017) concurrently proposed the first adversarial attacks on deep reinforcement learning agents by using adversarial examples crafted by FGSM. Mandlekar et al. (2017) proposed a physically plausible threat model and used FGSM perturbations to make the agent’s policy more robust. Pattanaik et al. (2018) proposed a projected gradient based method to maximize the probability of the worst possible action in the given state. Lin et al. (2017) considered the timing perspective of the adversarial attacks by using the Carlini & Wagner (2017) formulation. Pinto et al. (2017) use a two player zero-sum discounted Markov game to model the interaction of the adversary and the agent. The authors train the agent in this game with the adversary to increase robustness of the agent. Gleave et al. (2020) modeled the relationship between the agent and the adversary as a two player Markov game and solved it via Proximal Policy Optimization proposed by Schulman et al. (2017). In this work the authors focus on letting the adversary take natural actions in the environment instead of injecting $\ell_p$-norm bounded perturbations.

3. Nesterov Momentum- FGM

In this work we propose a Nesterov momentum-based method (Nesterov 1983) to find the adversarial perturbation.

**Algorithm 1** Nesterov Momentum-FGM

**Input:** Loss function $J$, the bound on perturbation $\epsilon$, actions $a$, states $s$, iterations $T$ and decay factor $\mu$.

**Output:** Adversarially perturbed state $s_{adv}$ with $\|s - s_{adv}\|_2 \leq \epsilon$

$$\begin{equation}
\alpha = \epsilon / T; v_0 = 0; s^0_{adv} = 0
\end{equation}$$

for $t = 1$ to $T$ do

- Calculate $\nabla_{s_{adv}} J(s_{adv}^t + \mu \cdot v_t, a)$
- $v_{t+1} = \mu \cdot v_t + \frac{\nabla_{s_{adv}} J(s_{adv}^t + \mu \cdot v_t, a)}{||v_{t+1}||_1}$
- $s_{adv}^{t+1} = s_{adv}^t + \alpha \cdot \frac{v_{t+1}}{||v_{t+1}||_2}$$

end for

**Return:** $s_{adv} = s_{adv}^T$

In Nesterov momentum the accumulated gradients,

$$\begin{equation}
\nabla_{s_{adv}} J(s_{adv}^t + \mu \cdot v_t, a)
\end{equation}$$
4. Experiments

4.1. Experimental Setup

In our experiments we averaged over 10 episodes for 3 Atari games Bellemare et al. (2013) from the Open AI gym environment Brockman et al. (2016). Our agents are trained with Double-DQN Wang et al. (2016). We normalize the average return of the agents when we calculate the impact of the adversary as follows. Let $R_{\text{max}}$ be the average return for the agent who always chooses the best action in a given state, let $R_{\text{min}}$ be the average return for the agent who always chooses the worst possible action in a given state, and let $R_a$ be the average return of the agent under the influence of the adversary. We define the impact,

$$I = \frac{R_{\text{max}} - R_a}{(R_{\text{max}} - R_{\text{min}})}$$  \hfill (6)

Intuitively this normalization measures how much the adversary degrades the performance of the agent when compared with a worst-case agent which always chooses the worst possible action. It is important to take this worst-case agent into account, because the agent still collects non-zero stochastic rewards from the environment even when always choosing the worst possible action.

The comparison of MI-FGM and our proposed optimization algorithm Nesterov-MFGM for adversarial formulation in deep reinforcement learning is shown in Table 1. It can be seen that our proposed algorithm has a higher mean and lower standard deviation impact on the agent compared to the state-of-the-art.

### Table 1. Attack impacts for MI-FGM and Nesterov-MFGM with $\ell_2$ norm bound, $\epsilon = 10^{-4}$.

<table>
<thead>
<tr>
<th>Games</th>
<th>MI-FGM</th>
<th>Nesterov-MFGM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alien</td>
<td>0.8240±0.23</td>
<td>0.9423±0.03</td>
</tr>
<tr>
<td>Bankheist</td>
<td>0.9030±0.05</td>
<td>0.9468±0.02</td>
</tr>
<tr>
<td>Boxing</td>
<td>0.5904±0.29</td>
<td>0.6798±0.26</td>
</tr>
</tbody>
</table>
4.2. $\ell_2$-norm Bound Comparison

In this section we will compare the state-of-the-art targeted attack Carlini & Wagner (2017) and our proposed Nesterov-MFGM adversary. The adversarial perturbation in $\ell_2$-norm, $\|s - s_{adv}\|_2$, over the states computed by Carlini & Wagner (2017) formulation is shown in Figure 2. Table 2 and Table 3 show the corresponding impact value for the given perturbation profile, and the mean of the $\ell_2$-norm adversarial perturbation over the states. Note that in Table 2 and 3 Carlini & Wagner (2017) achieves less impact yet requires orders of magnitude larger adversarial perturbations.

Table 2. Attack impacts and $\|s - s_{adv}\|_2$ for Carlini & Wagner (2017) and Nesterov-MFGM in Alien.

<table>
<thead>
<tr>
<th>Alien</th>
<th>Carlini &amp; Wagner (2017)</th>
<th>Nesterov-MFGM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>0.8240</td>
<td>0.9329</td>
</tr>
<tr>
<td>$|s - s_{adv}|_2$</td>
<td>5.043</td>
<td>0.9705·10⁻⁵</td>
</tr>
</tbody>
</table>

Table 3. Attack impacts and $\|s - s_{adv}\|_2$ for Carlini & Wagner (2017) and Nesterov-MFGM in Boxing.

<table>
<thead>
<tr>
<th>Boxing</th>
<th>Carlini &amp; Wagner (2017)</th>
<th>Nesterov-MFGM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Impact</td>
<td>0.8564</td>
<td>0.9981</td>
</tr>
<tr>
<td>$|s - s_{adv}|_2$</td>
<td>3.0158</td>
<td>7.0037·10⁻⁵</td>
</tr>
</tbody>
</table>

5. Conclusion

In this paper we proposed a new optimization method based on Nesterov momentum for computing adversarial examples for deep reinforcement learning. We show in various games from the Atari environment that our proposed approach achieves higher impact compared to the state-of-the-art. Investigating the computation of adversarial perturbations is an important first step in designing robust deep reinforcement learning algorithms. Furthermore, we believe our optimization method can be instrumental in adversarial training algorithms for deep reinforcement learning.

References


