KTH ROYAL INSTITUTE
OF TECHNOLOGY

Robust Deep RL with Adversarial Attacks

Authors : Anay Pattanaik, Zhenyi Tang, Shuijing Liu, Gautham Bommannan and Girish Chowdhary

Presenter: Ezgi Korkmaz

Background: FGSM

- Linear approximation of the network and engineer an attack
- Assume a linear model $f(x) = w^T x$ and the change will be $f(x) = w^T x + w^T \eta$
- Adversarial attack for nonlinear network is $\eta_{min} = \epsilon sign(\nabla_x J(\theta, x, y))$
- Loss function in trainning or testing $J(\theta, x, y)$
- In case of images, cross entropy loss between true image label and predicted distribution over labels

Adversarial Attack

$Definition$: An adversarial attack is any possible perturbation that leads the agent into increased probability of taking the worst possible action in that state.

- Only valid for value function based algorithms
- A3C, DDPG are value function based
- Bounded by l_2 norm

Naive Adversarial Attack

- Generate random noise and add it to the current state

Algorithm 1 Naive Attack: DDQN

$NAIVE(Q^{target}, Q, s, \epsilon, n, \alpha, \beta)$
$a^* = argmax_a Q(s, a)$, $Q^* = max_a Q^{target}(s, a)$
for i = 1:n **do**
    $n_i \sim beta(\alpha, \beta) - 0.5$
    $s_i = s + \epsilon * n_i$
    $a_{adv} = argmax_a Q(s_i, a)$
    $Q_{adv}^{target} = Q^{target}(s, a_{adv})$
    if $Q_{adv}^{target} < Q^*$ **then**
        $Q^* = Q_{adv}^{target}$
        $s_{adv} = s_i$
    else
        do nothing
    end if
end for
return s_{adv}

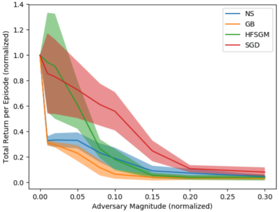
Gradient based Adversarial Attack

Algorithm 2 Gradient based Attack: DDQN

$GRAD(Q^{target}, Q, s, \epsilon, n, \alpha, \beta)$
$a^* = argmax_a Q(s, a), Q^* = max_a Q^{target}(s, a)$
$\pi^{target} = softmax(Q^{target})$
$grad = \nabla_s J(s, \pi^{target})$
$grad\_dir = \frac{\nabla_s J(s, \pi^{target})}{||\nabla_s J(s, \pi^{target})||}$
for i = 1:n **do**
  $n_i \sim beta(\alpha, \beta) - 0.5$
  $s_i = s - n_i * grad\_dir$
  $a_{adv} = argmax_a Q(s_i, a)$
  $Q^{target}_{adv} = Q^{target}(s, a_{adv})$
  if $Q^{target}_{adv} < Q^*$ **then**
    $Q^* = Q^{target}_{adv}$
    $s_{adv} = s_i$
  else
    do nothing
  end if
end for
return s_{adv}

Results

- NS refers to Naive Sampling attack, GB refers to gradient based attack, HFGSM refers to Huang attack



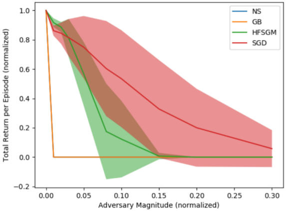


Figure: Left: DDQN Cartpole. Right: DDQN Mountain Car

Comments

- **"Attack is considered successful if a given image is classified as any other image"** not if it is a targeted attack.
- **"Papernot et al. [2016] showed that distillation is secure under l_∞ norm"** which is not Carlini Wagner showed that distillation is not secure if we use CW attack. CW has $\%100$ success rate both in l_∞ and l_0