Actor Critic Agent

Marco Marini

May 23, 2020

Abstract

1 TD Error

The TD error is defined as

$$\delta_t = r_t - r_\pi + v_\pi(s_{t+1}) - v_\pi(s_t) \tag{1}$$

2 Policy actor

The policy actor estimate the probabilities $\pi_a(s)$ of choose action a at status s. The function is the softmax of the actions preferences $h_a(s)$

$$\pi(a, s) = \frac{e^{h_a(s)}}{\sum_k e^{h_k(s)}} \tag{2}$$

simplifying the notation with

$$\begin{aligned} \pi(a, s) &= \pi_a \\ h_a(s) &= h_a \end{aligned}$$

The update of policy gradient is

$$\begin{aligned} \nabla \ln \pi_a &= \frac{1}{\pi_a} \frac{\partial}{\partial h_a} \pi_a \\ &= \frac{1}{\pi_a (\sum_k e^{h_k})^2} \left[e^{h_a} \nabla h_a - e^{h_a} \nabla \sum_k e^{h_k} \right] \\ &= \frac{1}{\sum_k e^{h_k}} \left[\nabla h_a - \sum_k \nabla e^{h_k} \right] \\ &= \frac{1}{\sum_k e^{h_k}} \left[\nabla h_a - \sum_k e^{h_k} \nabla h_k \right] \end{aligned}$$

Let be

$$
\begin{aligned}
A_i(a) = 1, \Rightarrow i = a \\
A_i(a) = 0 \Rightarrow i \neq a
\end{aligned}
$$

then

$$
\begin{aligned}
\nabla \ln \pi_a &= \frac{1}{\sum_k e^{h_k}} \sum_i \left[A_i(a) - e^{h_i} \right] \nabla h_i \\
&= \sum_i \left[\frac{A_i(a)}{\sum_k e^{h_k}} - \pi_i \right] \nabla h_i
\end{aligned}
$$

The backwork propagated TD error to the output neural network is

$$
\begin{aligned}
\delta_{h_a}(t) &= \delta(t) \nabla \ln \pi_a \\
&= \delta(s_t) \sum_i \left[\frac{A_i(a_t)}{\sum_k e^{h_k}} - \pi_i \right]
\end{aligned} \tag{3}
$$

The updated actor preferences are

$$
h_a^*(s_t) = h_a(s_t) + \alpha_h \delta_{h_a}(t) \tag{4}
$$

3 Gaussian policy actor

The Gaussian policy actor estimate the probabilities $\pi(a, s)$ of choose a continuous action a at status s as a normal distributed function of two parameters $\mu(s)$ and $\sigma(s)$.

We change the notation to avoid ambiguity between the constant $\pi = 3.14\ldots$ and the policy $\pi(a, s)$:

$$
\begin{aligned}
\pi(a, s) &= p(a, s) \\
&= \frac{1}{\sigma(s)\sqrt{2\pi}} e^{-\frac{(a-\mu(s))^2}{\sigma(s)^2}}
\end{aligned} \tag{5}
$$

$$
\sigma(s) = e^{h_\sigma(s)} \tag{6}
$$

Futhermore we change the notation to simplifying the equation:

$$
\begin{aligned}
p(a, s) &= p \\
\mu(s) &= \mu \\
\sigma(s) &= \sigma \\
h_\sigma(s) &= h_\sigma
\end{aligned}
$$

to infere the gradient of logarithm of p

$$\nabla \ln p = \left(\frac{\partial}{\partial \mu} + \frac{\partial}{\partial h_\sigma} \right) \ln p$$

the partial derivative by μ is

$$\begin{aligned}
\frac{\partial}{\partial \mu} \ln p &= \frac{1}{p} \frac{\partial p}{\partial \mu} \\
&= \frac{1}{p\sigma\sqrt{2\pi}} e^{\frac{-(a-\mu)^2}{\sigma^2}} \frac{\partial}{\partial \mu} \left[- \frac{(a-\mu)^2}{\sigma^2} \right] \\
&= -\frac{1}{\sigma^2} [2(a-\mu)(-1))] \\
&= \frac{2}{\sigma^2}(a-\mu) \\
\frac{\partial}{\partial h_\sigma} \ln p &= \frac{1}{p} \frac{\partial p}{\partial \sigma} \frac{\partial \sigma}{\partial h_\sigma} \\
\frac{\partial p}{\partial \sigma} &= \frac{1}{\sigma^2 \sqrt{2\pi}} \left\{ \sigma \frac{\partial}{\partial \sigma} \left[e^{-\frac{(a-\mu)^2}{\sigma^2}} \right] - e^{-\frac{(a-\mu)^2}{\sigma^2}} \right\} \\
&= \frac{p}{\sigma} \left\{ \sigma \frac{\partial}{\partial \sigma} \left[-(a-\mu)^2 \sigma^{-2} \right] - 1 \right\} \\
&= \frac{p}{\sigma} \left[-\sigma(a-\mu)^2(-2\sigma^{-3}) - 1) \right] \\
&= \frac{p}{\sigma} \left[2(a-\mu)^2 \sigma^{-2} - 1 \right] \\
\frac{\partial \sigma}{\partial h_\sigma} &= \sigma \frac{\partial}{\partial h_\sigma} \ln p \\
&= \frac{1}{p} \frac{p}{\sigma} \left[2(a-\mu)^2 \sigma^{-2} - 1 \right] \sigma \\
&= 2(a-\mu)^2 \sigma^{-2} - 1
\end{aligned}$$

The backward propagated TD errors to the output network layer are

$$\delta_\mu(t) = \frac{2}{\sigma^2}(a-\mu)\delta(t) \tag{7}$$

$$\delta_{h_\sigma}(t) = \left[2\frac{(a-\mu)^2}{\sigma^2} - 1 \right] \delta(t) \tag{8}$$

The updated actor parameters are:

$$\mu^*(s_t) = \mu(s_t) + \alpha_\mu \delta_\mu(t) \tag{9}$$

$$h^*_\sigma(s_t) = h_\sigma(s_t) + \alpha_{h_\sigma} \delta_{h_\sigma}(t) \tag{10}$$

4 Performance Indicators

The agent has a lot of iper parameters to tune and optimize the learning rate.

In this section we define performance indicators to tune such parameters.

4.1 Neural Network Indicator

Both the critic and actors aproximate the value function and policy function with neural network. To monitor the learning activity of neural network we compute the MSE of extimated functions.

The critic computes the updated value of current state by appling the the bootstrap equation:

$$v^*(s_t) = v(s_{t+1}) + r_t - r_\pi$$

The square error of critic output is

$$\begin{aligned} J_v(t) &= [v^*(s_t) - v(s_{t+1})]^2 \\ &= \delta^2(t) \end{aligned}$$

The square error of policy actors outputs are

$$J_{h,i} = \sum_a (h^*_{a,i}(s_t) - h_{a,i}(s_t))^2$$

where i is index of the actor dimension and a is the action value preference index

The square error of gaussian actors outputs are

$$\begin{aligned} J_{\mu,i} &= (\mu^*_i(s_t) - \mu_i(s_t))^2 \\ J_{h_\sigma,i} &= (h^*_{\sigma,i}(s_t) - h_{\sigma,i}(s_t))^2 \end{aligned}$$

where i is index of the actor dimension.

In the same way we define the square error of outputs for fitted actor:

$$\begin{aligned} J_{v'}(t) &= [v^*(s_t) - v'(s_{t+1})]^2 \\ J_{h',i}(t) &= \sum_a (h^*_{a,i}(s_t) - h'_{a,i}(s_t))^2 \\ J_{\mu',i}(t) &= (\mu^*_i(s_t) - \mu'_i(s_t))^2 \\ J_{h'_\sigma,i}(t) &= (h^*_{\sigma,i}(s_t) - h'_{\sigma,i}(s_t))^2 \end{aligned}$$

The ratio between the total MSE after and before the learning activity indicates the quality of such activity.

$$\begin{aligned} J(t) &= J_v(t) + \sum_i [J_{h,i}(t) + J_{\mu,i}(t) + J_{h_\sigma,i}(t)] \\ J'(t) &= J_{v'}(t) + \sum_i [J_{h',i}(t) + J_{\mu',i}(t) + J_{h'_\sigma,i}(t)] \\ K(t) &= \frac{J'(t)}{J(t)} \end{aligned} \tag{11}$$

A ratio $K(t) \geq 1$ means the error after learning gets worst due a step-size parameter α too high. A ratio $K(t) = 1$ means no change on error and therfore no improvement,. This can be affacted by a local minimum reached or a step-size parameter too low with very poor capacity of learning. A ratio $K(t) < 1$ means an improvement of neural network due to correct step-size parameter. A ratio $K(t) = 0$ means a perfect fit of neural network.

We can classify the steps in three class:

C_0 The steps that created a bad approximation with an increased of MSE ($J > \varepsilon \cup K > 1$)

C_1 The steps that create a trivial approximation with a small reduction of MSE ($J > \varepsilon \cup K_0 \leq K \leq 1$ with $K_0 = 0.9$)

C_2 The remaining steps that have a small MSE or that have reduced significantly the MSE

If the steps in C_0 is greater that a threshold (10% of total steps) the step parameter should be reduced, if the steps in C_1 is greater than a threshold, (50%) of steos, the step parameter may be increased. The figure (1) represents the classification of steps.

It is difficult to determine the effetcs of step parameter changes on the result MSE so an empirical way to reduce or increment the parameter is applied, for example mulipling by exponential factors $(\times 3, \times 10, \times 30, \times 100, \ldots)$ or $(\times 0.3, \times 0.1, \times 0.03, \times 0.01, \ldots)$ until the steps in C_0 and C_1 are below the thresholds.

4.2 Policy Actor Indicators

The actor computes the updated preferernces of current state by adding a step-size parameter to gradient and TD error

$$h^*_{a,i}(s_t) = h_{a,i}(s_t) + \alpha_{h,i}\delta_{h_a,i}(t)$$

To avoid computation overflow the preferences are constratints to a limited range e.g. $(-7, +7)$. The changes of preferences should also be limited to a fraction of the range $(-\varepsilon_h, \varepsilon_h)$, so we can measure the squared distance of changes of preferences:

$$J_{h,i}(t) = \alpha^2_{h,i} \sum_a \delta^2_{h_a,i}(t) \tag{12}$$

A $J_{h,i}(t) \geq \varepsilon^2_h$ means a $\alpha_{h,i}$ parameter value too high and $J_{h,i}(t) \ll \varepsilon^2_h$ means a α_{h_i} parameter value too small, producing an uneffective changes on preferences.

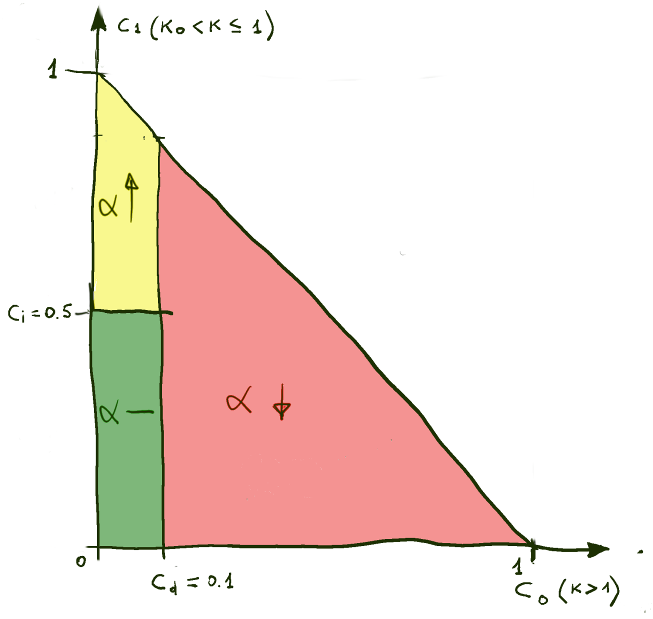

Figure 1: Step classification

Asserting we want to have a p fraction of samples with a $J_{h,i}(t) < \varepsilon_h^2$, we calculate $J_{h,i,p}$ the p centile of $J_{h,i}(t)$ and comupute the $\gamma_{h,i}$ scale parameter

$$\begin{aligned} \varepsilon_h^2 &= \gamma_{h,i}^2 J_{h,i,p} \\ \gamma_{h,i} &= \frac{\varepsilon_h}{\sqrt{J_{h,i}(t)}} \end{aligned} \tag{13}$$

4.3 Gaussian Policy Actor Indicators

The actor computes the updated parameters of current state by adding a step-size parameter to gradient and TD error

$$\begin{aligned} \delta_\mu &= \frac{2}{\sigma^2}(a-\mu)\delta \\ \delta_{h_\sigma} &= \left[2\frac{(a-\mu)^2}{\sigma^2} - 1\right]\delta \end{aligned}$$

The updated gaussian parameters are

$$\begin{aligned} \mu^*(s_t) &= \mu(s_t) + \alpha_\mu \delta_\mu(s_t) \\ h_\sigma^*(s_t) &= h_\sigma(s_t) + \alpha_{h_\sigma} \delta_{h_\sigma}(s_t) \end{aligned}$$

We may consider the changes to the gaussian policy parameter limited to a defined range

$$\begin{aligned} J_\mu(s_t) &< \varepsilon_\mu^2 \\ |\mu^*(s_t) - \mu(s_t)| &< \varepsilon_\mu \\ |\delta_\mu(s_t)| &< \varepsilon_\mu \\ J_{h_\sigma}(s_t) &< \varepsilon_{h_\sigma}^2 \\ [h_\sigma^*(s_t) - h_\sigma(s_t)]^2 &< \varepsilon_{h_\sigma}^2 \\ |\delta_{h_\sigma}(s_t)| &< \varepsilon_{h_\sigma} \end{aligned}$$

An indicator $J_\mu(s_t) \geq \varepsilon_\mu^2$ means an α_μ parameter value too high, on the other hand an indicator$J_\mu(s_t) \ll \varepsilon_\mu^2$ means an α_μ parameter value too small. An indicator $J_{h_\sigma}(s_t) \geq \varepsilon_{h_\sigma}^2$ means an α_σ parameter value too high and an indicator $J_{h_\sigma}(s_t) \ll \varepsilon_{h_\sigma}^2$ means an α_{h_σ} parameter value too small.

Asserting we want to have a p fraction of samples with a $J_\mu(s_t) < \varepsilon_\mu^2$, we calculate $J_{\mu,p}$ the p centile of $J_\mu(s_t)$ and comupute the γ_μ

$$\begin{aligned} \varepsilon_\mu^2 &= \gamma_\mu^2 \sigma_\mu^2 \\ &= \gamma_\mu^2 J_\mu(s_t) \end{aligned}$$

$$\gamma_\mu = \frac{\varepsilon_\mu}{\sqrt{J_{\mu,p}}} \tag{14}$$

In the same way we have

$$\gamma_{h_\sigma} = \frac{\varepsilon_{h\sigma}}{\sqrt{J_{h_\sigma,p}}} \tag{15}$$