[Mustafa Suman] Assignment 2: Policy Gradients

Due September 25, 11:59 pm

4 Policy Gradients

- Create two graphs:
  - In the first graph, compare the learning curves (average return vs. number of environment steps) for the experiments prefixed with `cartpole`. (The small batch experiments.)
  - In the second graph, compare the learning curves for the experiments prefixed with `cartpole_lb`. (The large batch experiments.)

  For all plots in this assignment, the x-axis should be number of environment steps, logged as `Train_EnvstepsSoFar` (*not* number of policy gradient iterations).
- Answer the following questions briefly:
  - Which value estimator has better performance without advantage normalization: the trajectory-centric one, or the one using reward-to-go?
  - Did advantage normalization help?
  - Did the batch size make an impact?
- Provide the exact command line configurations (or `#@params` settings in Colab) you used to run your experiments, including any parameters changed from their defaults.

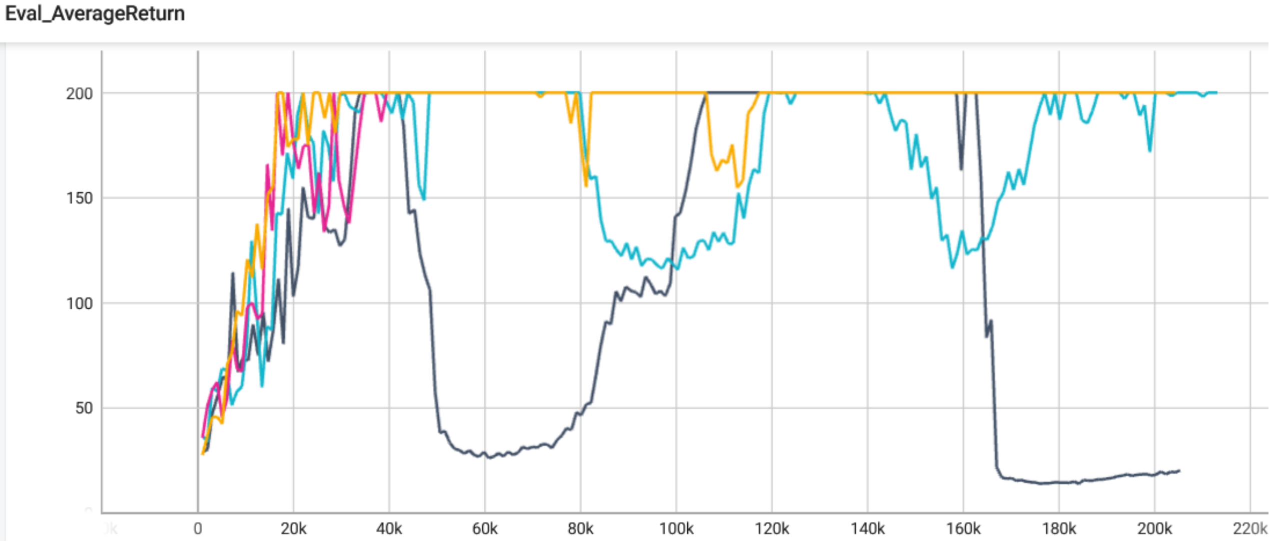


Figure 1: Small batch size. Black = plain, Blue = rtg, Pink = na, Yellow = rtg and na

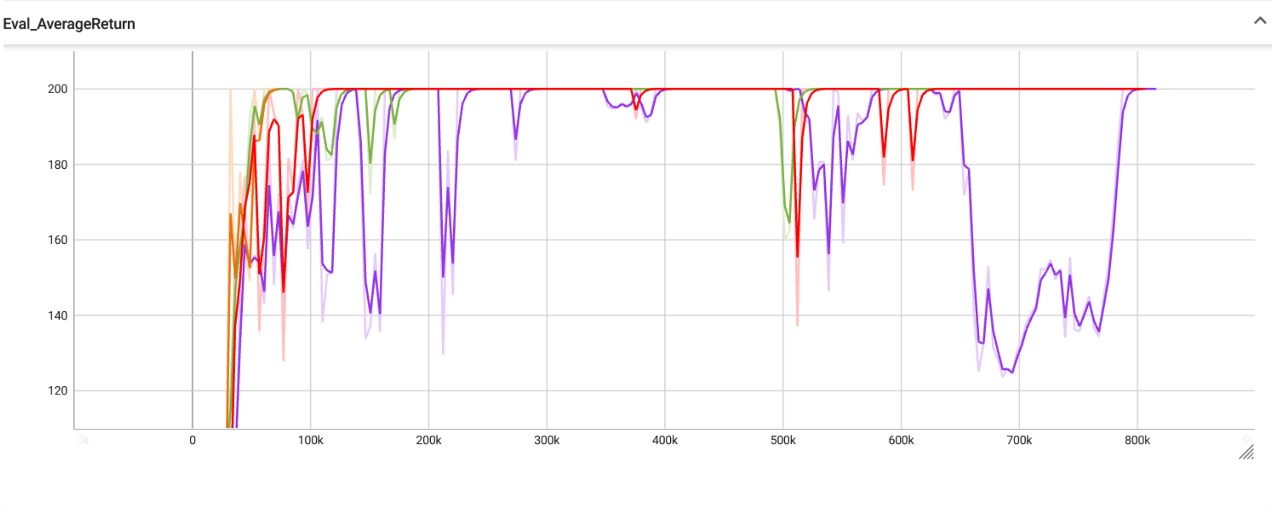


Figure 2: Large batch size. Lila = plain, Green = rtg, Orange = na, red = rtg and na

Comments: Which value estimator has better performance without advantage normalization: the trajectory-centric one, or the one using reward-to-go? small: Reward to Go is way more stable and doesn't show a dip in performance while also reaching a higher return earlier. Big: Same effect – Did advantage normalization help? without the reward to go it helped but crashed unfortunately since the policy returned nan logits. Considering the reward to go, it helped stabilize the training process and preventing some deep ditches. (although in the big batch case the effect vanished/remains unclear) – Did the batch size make an impact? Comparing the plain (and rtg) methods, it helped stabilizing the process and had a positive impact, for the other cases it is not as clear.

The commands were the once provided without any changes.

5 Neural Network Baseline

- Plot a learning curve for the baseline loss.
- Plot a learning curve for the eval return. You should expect to achieve an average return over 300 for the baselined version.
- Run another experiment with a decreased number of baseline gradient steps (`-bgs`) and/or baseline learning rate (`-blr`). How does this affect (a) the baseline learning curve and (b) the performance of the policy?
- **Optional:** Add `-na` back to see how much it improves things. Also, set `video_log_freq 10`, then open TensorBoard and go to the "Images" tab to see some videos of your HalfCheetah walking along!

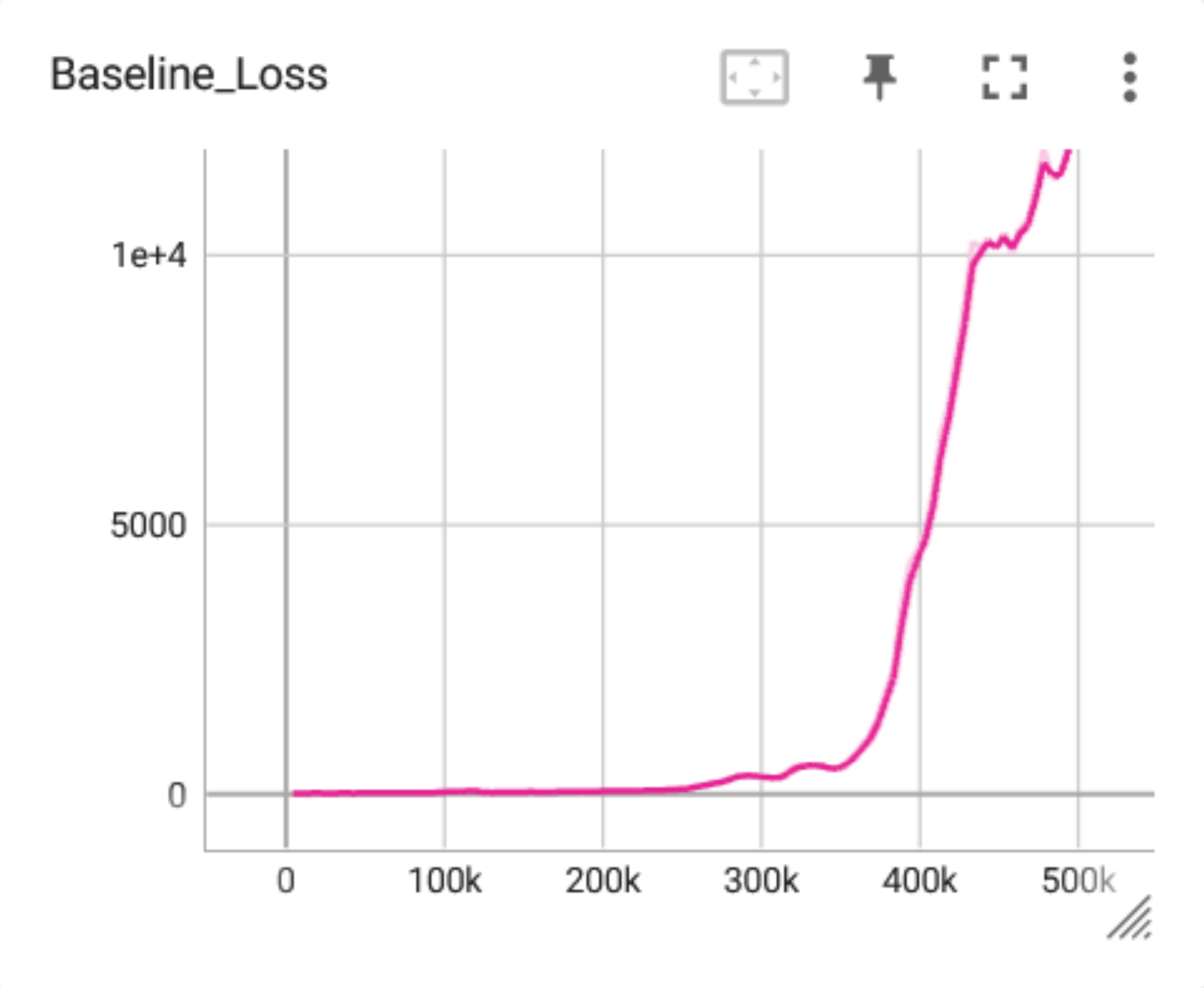


Figure 3: Baseline Loss

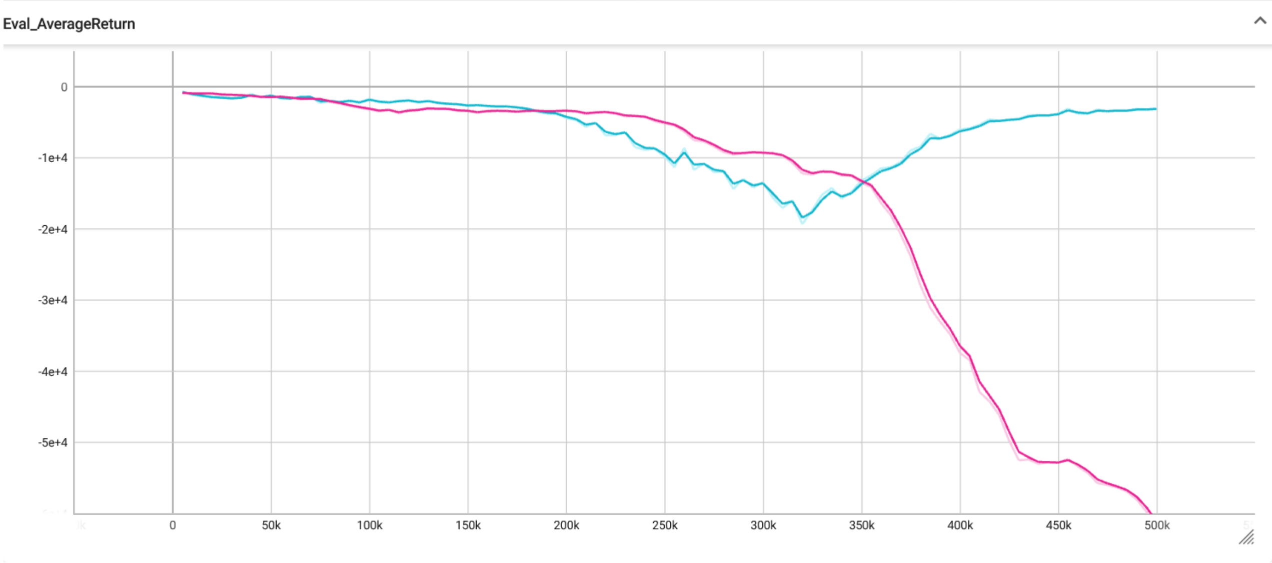


Figure 4: Comparison of Return. Blue = without Baseline, Purple = with Baseline.

It seems like I have some sort of hardware issue. The further parameter investigations do not make any sense. The commands were the once provided without any changes.

6 Generalized Advantage Estimation

- Provide a single plot with the learning curves for the `LunarLander-v2` experiments that you tried. Describe in words how λ affected task performance. The run with the best performance should achieve an average score close to 200 (180+).
- Consider the parameter λ. What does $\lambda = 0$ correspond to? What about $\lambda = 1$? Relate this to the task performance in `LunarLander-v2` in one or two sentences.

This did not work. $\lambda = 0$ and $\lambda = 1$ should equal to regular Monte-Carlo Q-learning emphasizing higher values of n which should be better in the LunarLander case, since the landing part is the most crucial and happens rather at a foreseeable future than in the next immediate steps.

7 Hyperparameter Tuning

1. Provide a set of hyperparameters that achieve high return on `InvertedPendulum-v4` in as few environment steps as possible.
2. Show learning curves for the average returns with your hyperparameters and with the default settings, with environment steps on the x-axis. Returns should be averaged over 5 seeds.

Same as before.

8 (Extra Credit) Humanoid

1. Plot a learning curve for the Humanoid-v4 environment. You should expect to achieve an average return of at least 600 by the end of training. Discuss what changes, if any, you made to complete this problem (for example: optimizations to the original code, hyperparameter changes, algorithmic changes).

1

a) $$\mathbb{E}_{\underline{\tau \sim p_\theta(\tau)}}\left(\underline{\nabla_\theta \log \pi_\theta(\tau)} \cdot \underline{r(\tau)}\right)$$

$$= \underline{\sum_{k=1}^{\infty} \theta^k \cdot (1-\theta)} \cdot \underline{\nabla_\theta \log\left(\theta^k \cdot (1-\theta)\right)} \cdot \underline{k}$$

$$= \sum_{k=1}^{\infty} \theta^k (1-\theta) \cdot \left(\nabla_\theta \log(\theta^k) + \nabla_\theta \log(1-\theta)\right) \cdot k$$

$$= \sum_{k=1}^{\infty} \theta^k \cdot (1-\theta) \cdot \left(k \cdot \frac{1}{\theta} - \frac{1}{1-\theta}\right) \cdot k$$

$$= \sum_{k=1}^{\infty} \theta^k (1-\theta) \cdot \left(\frac{k(1-\theta) - \theta}{\theta(1-\theta)}\right) \cdot k$$

$$= \sum_{k=1}^{\infty} \theta^{k-1} \left(k(1-\theta) - \theta\right) \cdot k$$

$$= \sum_{k=1}^{\infty} k^2 \cdot \theta^{k-1} (1-\theta) - k \cdot \theta^k$$

k=0 or k=1 makes no differena

$$\overset{\text{prepare for hint}}{=} \theta(1-\theta) \sum_{\underline{k=0}}^{\infty} k^2 \cdot \theta^{k-2} - \theta \sum_{\underline{k=0}}^{\infty} k \cdot \theta^{k-1}$$

$$\overset{\text{Hint + geom. Series}}{=} \theta(1-\theta)\left(\frac{d^2}{d\theta^2}\left(\frac{1}{1-\theta}\right) + \frac{1}{\theta}\frac{d}{d\theta}\left(\frac{1}{1-\theta}\right)\right) - \theta \frac{d}{d\theta} \frac{1}{1-\theta}$$

$$= \theta(1-\theta)\left(\frac{2}{(1-\theta)^3} + \frac{1}{\theta} \frac{1}{(1-\theta)^2}\right) - \frac{\theta}{(1-\theta)^2}$$

$$= \frac{1}{(1-\theta)^2}$$

b) $$\mathbb{E}_{\tau \sim \pi_\theta}\left(R(\tau)\right) = \sum_{k=1}^{\infty} \theta^k (1-\theta) \cdot k$$

$$= \sum_{k=1}^{\infty} k \cdot \left(\theta^k - \theta^{k+1}\right) \overset{\text{telescope sum}}{=} \sum_{k=1}^{\infty} \theta^k \overset{\text{geom. Series}}{=} \frac{1}{1-\theta} - \frac{1-\theta}{1-\theta} = \frac{\theta}{1-\theta}$$

Hence:

$$\nabla_\theta \mathbb{E}_{\tau \sim \pi_\theta}\left(R(\tau)\right) = \frac{1}{(1-\theta)^2}$$

$$\mathrm{Var}_\theta\left(\nabla_\theta \log \pi_\theta(\tau) \cdot r(\tau)\right) = \mathbb{E}_{\tau \sim \pi_\theta(\tau)}\left[\left(\nabla_\theta \log \pi_\theta(\tau) \cdot R(\tau)\right)^2\right] - \mathbb{E}_{\tau \sim \pi_\theta(\tau)}\left[\nabla_\theta \log \pi_\theta(\tau) \cdot R(\tau)\right]^2$$

$$= \mathbb{E}_{\tau \sim \pi_\theta(\tau)}\left[\left(\nabla_\theta \log \pi_\theta(\tau) \cdot R(\tau)\right)^2\right] - \frac{1}{(1-\theta)^4}$$

$$= \sum_{k=0}^{\infty} \theta^k \cdot (1-\theta)\left(\nabla_\theta \log \pi_\theta(\tau) \cdot k\right)^2 - \frac{1}{(1-\theta)^4}$$

$$= \sum_{k=0}^{\infty} \theta^k (1-\theta)\left(\frac{k}{\theta} - \frac{1}{1-\theta}\right)^2 \cdot k^2 - \frac{1}{(1-\theta)^4}$$

$$\overset{\text{W. Alpha}}{=} \frac{4\theta^2 + 9\theta + 1}{(\theta-1)^4\,\theta} - \frac{1}{(1-\theta)^4}$$

$$\overset{\text{W. Alpha}}{=} \frac{4\theta^2 + 8\theta + 1}{(\theta-1)^4\,\theta}$$

Eyeballing:

Argmin ≈ 0.1

Argmax = towards 0 and 1

a) Before:

$$\nabla_\theta J(\theta) = \sum_{k=0}^{\infty} \theta^k (1-\theta) \left(\sum_{t=0}^{k} \nabla_\theta \log(\theta)\, k + \nabla_\theta \log(1-\theta) \cdot 0 \right)$$

Now with reward to go:

$$\nabla_\theta J(\theta) = \sum_{k=0}^{\infty} \theta^k (1-\theta) \left(\sum_{t=0}^{k} \nabla_\theta \log(\theta)(k-t) + \nabla_\theta \log(1-\theta) \cdot 0 \right)$$

$$= \sum_{k=0}^{\infty} \theta^k (1-\theta) \sum_{t=0}^{k} \frac{1}{\theta} (k-t)$$

$$\overset{\text{Gauß}}{=} \sum_{k=0}^{\infty} \theta^{k-1} (1-\theta) \frac{1}{2} k (k+1)$$

$$= \frac{(1-\theta)}{2} \sum_{k=0}^{\infty} \theta^{k-1} (k^2 + k)$$

$$= \frac{(1-\theta)}{2} \left(\theta \sum_{k=0}^{\infty} \theta^{k-2} k^2 + \sum_{k=0}^{\infty} \theta^k k \right)$$

$$\overset{\text{see 1}}{=} \frac{(1-\theta)\theta}{2} \left(\frac{2}{(1-\theta)^3} + \frac{1}{\theta} \frac{1}{(1-\theta)^2} \right) + \frac{1}{2(1-\theta)}$$

$$= \frac{1}{(1-\theta)^2}$$

b)

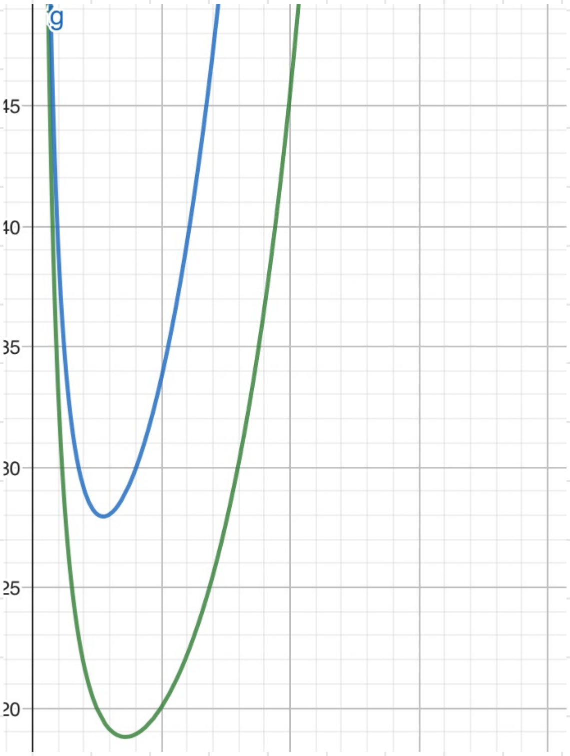
$$\mathrm{Var}_\theta\left(\nabla_\theta J(\theta)\right) = \mathbb{E}_{\tau\sim\pi_\theta(\tau)}\left(\left(\sum_{t=0}^{k}\nabla_\theta \log(\theta)(k-t)\right)^2\right) - \mathbb{E}_{\tau\sim\pi_\theta(\tau)}\left(\sum_{t=0}^{k}\nabla_\theta \log(\theta)(k-t)\right)^2$$

$$\overset{a)}{=} \mathbb{E}_{\tau\sim\pi_\theta(\tau)}\left(\left(\frac{1}{2\theta}k(k+1)\right)^2\right) - \frac{1}{(1-\theta)^4}$$

$$= \sum_{k=0}^{\infty}\theta^k(1-\theta)\frac{1}{4\theta^2}k^2(k^2+2k+1) - \frac{1}{(1-\theta)^4}$$

$$\overset{\text{W. Alph}}{=} \dots$$

$$= \frac{\theta+3\theta^2+\theta^3}{(1-\theta)^4\cdot\theta^2}$$



blue = original

green = reward to go