CS 229 Autumn 2018 Problem Set #4

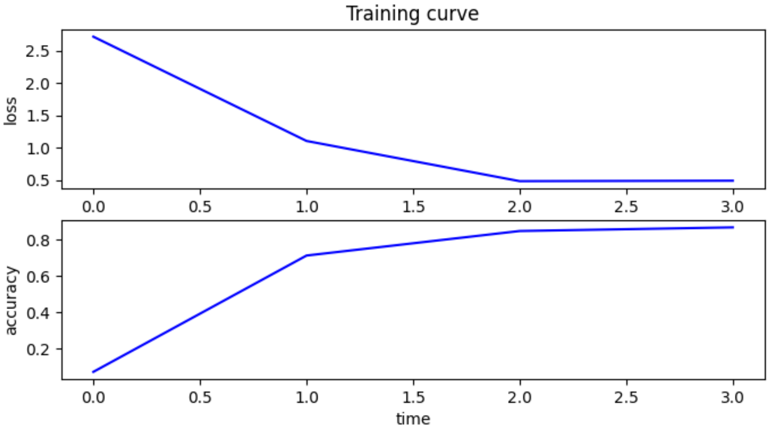
Ruining Li
University of Oxford

Problem 1

(a) Please see the code for a detailed implementation.

(b) Please see the code for a detailed implementation.

The network achieves a 85%+ accuracy on the dev set.



Problem 2

(a) When $\hat{\pi}_0 = \pi_0$, we have:

$$\begin{aligned}\mathop{\mathbb{E}}_{\substack{s\sim p(s)\\ a\sim\pi_0(s,a)}} \frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} R(s,a) &= \sum_{(s,a)} \frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} R(s,a)p(s)\pi_0(s,a) \\ &= \sum_{(s,a)} \pi_1(s,a)R(s,a)p(s) \\ &= \mathop{\mathbb{E}}_{\substack{s\sim p(s)\\ a\sim\pi_1(s,a)}} R(s,a)\end{aligned}$$

as required.

(b) With a similar derivation to part (a), we obtain:

$$\frac{\mathop{\mathbb{E}}_{\substack{s\sim p(s)\\ a\sim\pi_0(s,a)}} \frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} R(s,a)}{\mathop{\mathbb{E}}_{\substack{s\sim p(s)\\ a\sim\pi_0(s,a)}} \frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)}} = \frac{\sum_{(s,a)} \frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} R(s,a)p(s)\pi_0(s,a)}{\sum_{(s,a)} \frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} p(s)\pi_0(s,a)}$$

Here, because $\hat{\pi}_0 = \pi_0$, we obtain that the denominator

$$\sum_{(s,a)} \frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} p(s)\pi_0(s,a) = \sum_{(s,a)} \pi_1(s,a)p(s) = \sum_{(s,a)} p(s,a) = 1.$$

It follows that the weighted importance sampling estimator is equal to the $\mathbb{E}_{\substack{s\sim p(s)\\ a\sim\pi_1(s,a)}} R(s,a)$ as required.

(c) When there is only a single data element in the observational dataset, and we replace the expected value with a sum over the seen values in the dataset, our weighted importance sampling estimator becomes

$$\frac{\frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} R(s,a)p(s)\pi_0(s,a)}{\frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} p(s)\pi_0(s,a)} = R(s,a) \neq R(s,a)p(s)\pi_1(s,a)$$

where $(s,a,R(s,a))$ is the only tuple in the observational dataset.

This concludes our proof that the weighted importance sampling estimator is biased.

(d) (i) When $\hat{\pi}_0 = \pi_0$, the doubly robust estimator is equal to

$$\begin{aligned}
&\sum_{(s,a)} \left\{ (\mathbb{E}_{a\sim\pi_1(s,a)} \hat{R}(s,a)) + \frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} (R(s,a) - \hat{R}(s,a)) \right\} p(s)\pi_0(s,a) \\
=& \sum_{(s,a)} \hat{R}(s,a)p(s)\pi_1(s,a) + \sum_{(s,a)} \frac{\pi_1(s,a)}{\hat{\pi}_0(s,a)} (R(s,a) - \hat{R}(s,a))p(s)\pi_0(s,a) \\
=& \sum_{(s,a)} \hat{R}(s,a)p(s)\pi_1(s,a) + \sum_{(s,a)} \pi_1(s,a)(R(s,a) - \hat{R}(s,a))p(s) \\
=& \mathbb{E}_{\substack{s\sim p(s)\\ a\sim\pi_1(s,a)}} R(s,a)
\end{aligned}$$

as required.

(ii) When $\hat{R}(s,a) = R(s,a)$, the second term in the expectation vanishes and the doubly robust estimator therefore becomes $\mathbb{E}_{\substack{s\sim p(s)\\ a\sim\pi_1(s,a)}} \hat{R}(s,a)$, which again is the same as the $\mathbb{E}_{\substack{s\sim p(s)\\ a\sim\pi_1(s,a)}} R(s,a)$.

(e) (i) The importance sampling estimator would be better because it tends to be more accurate when $\hat{\pi}_0 = \pi_0$. We can model π_0 nearly perfectly as the probability mass is distributed uniformly among actions.

(ii) The regression estimator would be better because it tends to be more accurate when $\hat{R}(s,a) = R(s,a)$. We can model $R(s,a)$ nearly perfectly as the interaction between the drug, patient and lifespan is very simple.

Problem 3

When $u^T u = 1$, $f_u(x) = (u^T x)u$. Therefore, the squared error ℓ which we want to minimize becomes

$$\begin{aligned}
\ell &= \sum_{i=1}^{m} \left\| x^{(i)} - (u^T x^{(i)})u \right\|_2^2 \\
&= \sum_{i=1}^{m} \left\{ (I - uu^T)x^{(i)} \right\}^T \left\{ (I - uu^T)x^{(i)} \right\} \\
&= \sum_{i=1}^{m} \left(x^{(i)} \right)^T (I - uu^T)(I - uu^T)x^{(i)} \\
&= \sum_{i=1}^{m} \left(x^{(i)} \right)^T x^{(i)} + \sum_{i=1}^{m} \left(x^{(i)} \right)^T uu^T uu^T x^{(i)} - 2\sum_{i=1}^{m} \left(x^{(i)} \right)^T uu^T x^{(i)} \\
&= -\sum_{i=1}^{m} \left(x^{(i)} \right)^T uu^T x^{(i)} + \text{const}
\end{aligned}$$

where in the third equality we used the fact that $I - uu^T$ is symmetric and in the last equality we used the fact that $u^T u = 1$.

Therefore, minimizing ℓ is equivalent to maximizing $\sum_{i=1}^{m} \left(x^{(i)} \right)^T uu^T x^{(i)}$, which corresponds to the first principal component (we obtained by finding the "variance maximizing" direction).

Problem 4

(a) When g is the standard normal CDF, our log-likelihood can be simplified as

$$\ell(W) = n \log |W| - \frac{1}{2} nd \log(2\pi) - \sum_{i=1}^{n} \sum_{j=1}^{d} \frac{1}{2} (w_j^T x^{(i)})^2$$

which we want to maximize.

Compute the derivative of ℓ w.r.t W and set it to zero, we obtain

$$n(W^{-1})^T = W X^T X$$

and therefore $W^T W = n X^T X$.

Ambiguity: Note that if we exchange two rows of W, $W^T W$ doesn't change and therefore there are many solutions of W that satisfy the above equation.

(b) If we assume sources are distributed according to a standard Laplace distribution, the log-likelihood can be simplied as

$$\ell(W) = n\log|W| + nd\log\frac{1}{2} - \sum_{i=1}^{n}\sum_{j=1}^{d}|w_j^T x^{(i)}|$$

which we want to maximize.

Compute the derivative of ℓ w.r.t W, we obtain

$$\frac{d\ell}{dW} = n(W^{-1})^T - V$$

where $V \in \mathbb{R}^{d\times d}$ whose j-th row is given by $\operatorname{sign}(w_j^T x^{(i)})x^{(i)}$ and $\operatorname{sign}(x) = \frac{x}{|x|}$.

If we choose to use gradient ascent method to maximize the log-likelihood, the update rule is given by $W := W + \alpha\frac{d\ell}{dW}$ where the derivative is given above.

(c) Please see the code for a detailed implementation.

Problem 5

(a) By the definition of the operator B, we obtain

$$\begin{aligned}
\|B(V_1) - B(V_2)\|_\infty &= \gamma \max_{s\in S}\left|\max_{a\in A}\sum_{s'\in S}P_{sa}(s')V_1(s') - \max_{a\in A}\sum_{s'\in S}P_{sa}(s')V_2(s')\right| \\
&\le \gamma \max_{s\in S}\max_{a\in A}\left|\sum_{s'\in S}P_{sa}(s')(V_1(s') - V_2(s'))\right| \\
&\le \gamma \max_{s\in S, a\in A}\sum_{s'\in S}P_{sa}(s')\,|V_1(s') - V_2(s')| \\
&\le \gamma \max_{s'\in S}|V_1(s') - V_2(s')| \\
&= \gamma\,\|V_1 - V_2\|_\infty
\end{aligned}$$

where in the last inequality we used the fact that $\sum_{s'\in S}P_{sa}(s')$.

(b) Let's suppose, for contradiction, that B has two distinct fixed point, V_1 and V_2. I.e., $B(V_1) = B(V_2)$. Then, by part (a),

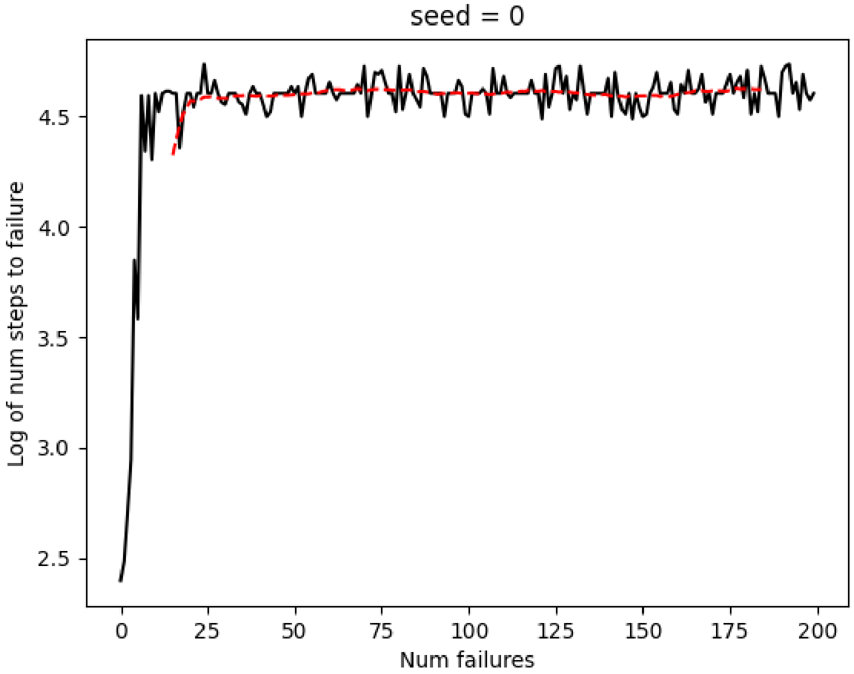
$$\|B(V_1) - B(V_2)\|_\infty = \|V_1 - V_2\|_\infty \le \gamma\,\|V_1 - V_2\|_\infty$$

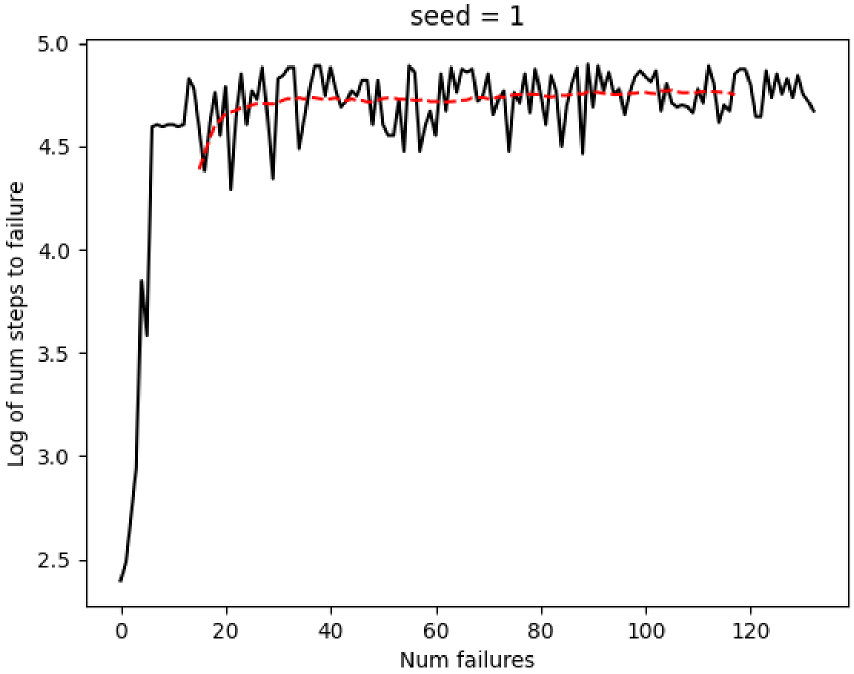
which contradicts with the fact that $\|V_1 - V_2\|_\infty > 0$ and $\gamma < 1$.

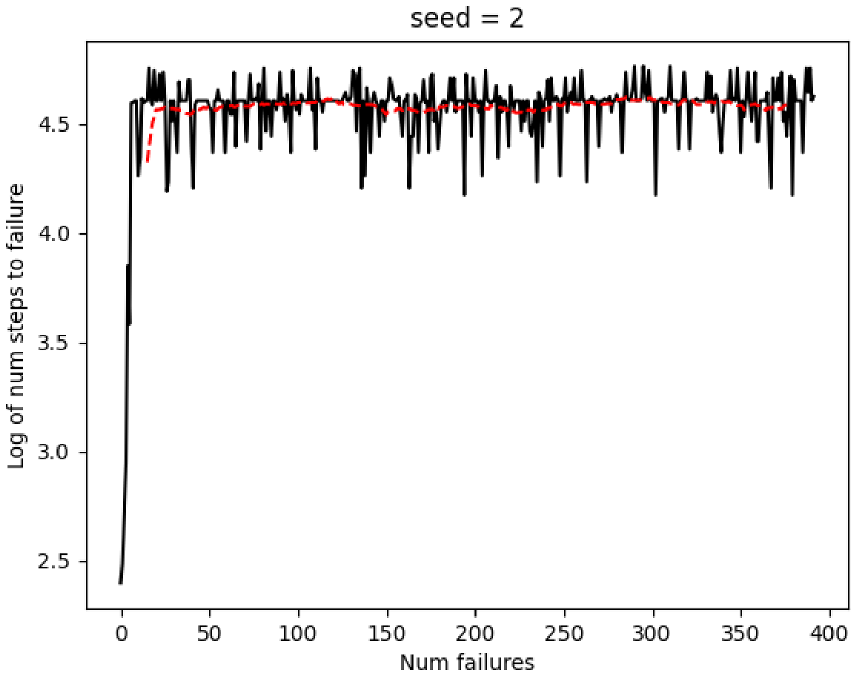
This concludes our proof that B has at most one fixed point.

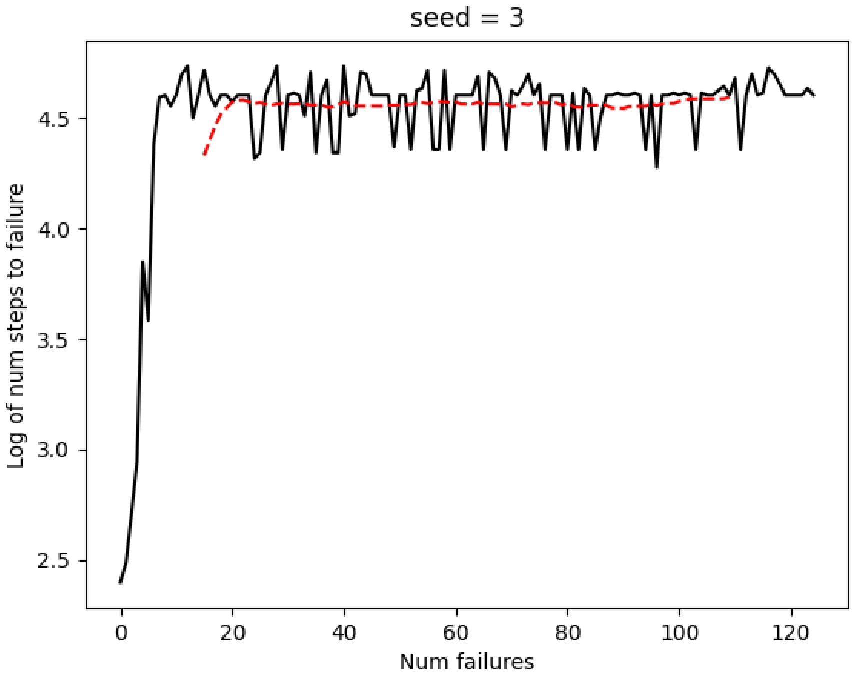
Problem 6

Please see the code for a detailed implementation. The learning curves of four different random initialization seed are shown below:









It took about 30 to 40 trials before the algorithm converged. For different random seeds, we notice that after the algorithm converged, the number of time-steps for which the pole was balanced on each trail is about the same (the log is approximately 4.5). This suggests that the performance of the algorithm is not very sensitive to the random initialization.