1 Motivation

1. The paper aims to experimentally study 3 issues in Q-learning: function approximation error, sampling error and nonstationarity.
2. They aim to study this in a "unit-testing" framework, where a component in an exact algorithm is replaced by another component that introduces a specific source of error.

2 Function Approximation Error and convergence

1. They aim to measure 2 quantities: trend between function approximation and performance, and a measure for the bias in the learning procedure introduced by function approximation.
2. They note a few trends:
3. Smaller architectures produce lower returns and converge to worse solutions.
4. Smaller architectures introduce significant bias in the learning process. As the architecture becomes smaller, the gap between the best result in this model class and the optimal result grows.
5. Interesting sentence: "When the target is bootsrapped, we must represent all Q-functions along the path to the solution, and not only the final result."

3 Sampling Error and Overfitting

1. In RL, we can only compute the empirical Bellman error over a finite set of samples.
2. This leads to error between empirical and expected loss.

3.1 Quantifying Overfitting

1. They observe that training by sampling samples from the replay buffer results in the lowest on-policy validation loss, despite bias due to distribution mis-match from sampling off-policy data.
2. They also show that despite overfitting being an issue, larger architecture still perform better because the bias introduced by smaller architecture dominates.

3.2 Compensate for overfitting

1. They argue against regularization because they believe that weaker architectures introduce bias.
2. They instead use early stopping as a strategy to mitigate overfitting without reducing model size.

4 Non-stationarity

1. They argue that there are two sources of non-stationarity: the changing target values and the changing weighting distribution of the samples.
2. They argue that instabilities due to these 2 sources are not major issues.
3. Other issues such as sampling and function approximation error caused bigger issues.

5 Sampling Distribution

1. They argue that this is little guidance on which weighting distributions should be used to select samples used for training.
2. They argues for the uniformity hypothesis: the best distributions spread weight across larger support of the state-action space.
3. For example, a replay buffer contains state-action tuples from many policies, and therefore would be expected to have wider support than the state-action distribution of a single policy.

5.1 Adversarial Feature Matching

1. They argue for 3 insights in designing the weighing distribution:
2. The function approximator should be incentivized to maximize its ability to distinguish states to minimize function approximation bias.
3. The weighing distribution should emphasize areas where the Q-function incurs high Bellman error
4. High-entropy weighting distributions tend to be more performant.
5. They thus propose the model their problem as a minimax game, where the weighting distribution is a parameterized adversary $p_\phi(s,a)$ which tries to maximize the Bellman error, while the Q-function $(Q_\theta(s,a))$ tries to minimize it.
6. Their feature matching constraint enforces that the expected feature vectors $\mathbb{E}[\Phi(s)]$, under $p_\phi(s,a)$ to roughly match the expected feature vector under uniform sampling from the replay buffer.
7. They also express the output of their Q-function as $Q_\theta(s,a) = w^T\Phi_\theta(s,a)$. where $\Phi_\theta(s,a)$ represent the output of all but the final layer. They argue that intuitively, this constraint restricts the adversary to distributing probability mass among states that are perceptually similar to the Q-function.
8. Their objective is:

$$\begin{array}{c} \min_{\theta,w} \max_\phi \mathbb{E}_{p_\phi(s,a)} \left[(Q_{w,\theta}(s,a) - y(s,a))^2 \right] \\ \text{s.t.} \ \left\| \mathbb{E}_{p_\phi(s,a)}[\Phi(s)] - \frac{\sum_i \Phi(s_i)}{N} \right\| \leq \varepsilon \end{array}$$

9. $\frac{\sum_i \Phi(s_i)}{N}$ denotes an estimator for the true expectation under some sampling distribution.
10. In tabular domain with exact Fitted Q Iteration, they found that AFM performs at par with the top performing weighing distributions, such as Unif and better than Prioritized Sampling.

6 Questions

1. How did they implement and set the hyper-parameters for PER?
2. The performance of SAC + AFM is similar to SAC? Is the sampling distribution of AFM in this case very different from uniform sampling from the replay buffer?
3. The argument that sampling distribution with high entropy does not seem general. Recall the toy experiment in Sutton book where reward is super sparse in a grid world environment.
4. In section 4.1, in "the project of the optimal solution", how is this solution obtained?
5. PAC framework is mentioned at the beginning of section 5. Is this used anywhere?
6. Stochastic prioritization in PER can also be seen as a way to reduce overfitting. Did they implement stochastic prioritization correctly?

7. In section 7.2, the sentence "The function approximator should be incentivized to maximize its ability to distinguish states to minimize function approximation bias.". How is this related to section 4?
8. I don't get the explanation in the second paragraph in section 7.2.
9. "the expected feature vectors $\mathbb{E}[\Phi(s)]$, under $p_\phi(s, a)$ to roughly match the expected feature vector under uniform sampling from the replay buffer". How does this try to maximize the Bellman error?
10. How is $p_\phi(s, a)$ parameterized exactly? Is it a distribution which assigns a probability to each entries in the replay buffer?
11. In the argument about the adversary (They argue that intuitively, this constraint restricts the adversary to distributing probability mass among states that are perceptually similar to the Q-function.). I am not sure this is a valid argument. The states are perceptually similar might be because they are actually states with similar state-action values.
12. In the optimization problem in section 7.2, $\Phi(s)$ should be $\Phi(s, a)$ in the case of continuous action case right?