Homework 6 Template

Use this template to record your answers for Homework 6. Add your answers using LaTeXand then save your document as a PDF to upload to Gradescope. You are required to use this template to submit your answers. **You should not alter this template in any way** other than to insert your solutions. You must submit all 8 pages of this template to Gradescope. Do not remove the instructions page(s). Altering this template or including your solutions outside of the provided boxes can result in your assignment being graded incorrectly. You may lose points if you do not follow these instructions.

You should also save your code as a .py or .zip file and upload it to the **separate** Gradescope coding assignment. Remember to mark all teammates on **both** assignment uploads through Gradescope.

Instructions for Specific Problem Types

On this homework, you must fill in (a) blank(s) for each problem; please make sure your final answer is fully included in the given space. **Do not change the size of the box provided.** For short answer questions you should **not** include your work in your solution. Only provide an explanation or proof if specifically asked. Otherwise, your assignment may not be graded correctly, and points may be deducted from your assignment.

Problem 0: Collaborators

Enter your team's names and Andrew IDs in the boxes below. If you do not do this, you may lose points on your assignment.

Name 1:	Akshay Sharma	Andrew ID 1:	akshaysh
Name 2:	Keitaro Nishimura	Andrew ID 2:	knishimu
Name 3:		Andrew ID 3:	

Problem 1: Crowd-Sourcing Quiz Questions

Please submit your four questions and answers to each using the following form (once per question):

https://forms.gle/xKqhqaxNaXHYRS2c7

Problem 2: Types of Uncertainty

2.1 Combined Variance

eFrom the definition of variance we know that:

$$\text{Var}(y) = \mathbb{E}[y^2] - \mathbb{E}[y]^2$$

By then applying the law of total expectation we can expand it into:

$$\text{Var}(y) = \mathbb{E}[\mathbb{E}[\, y^2 \,|\, x\,]] - \mathbb{E}[\mathbb{E}[\, y \,|\, x\,]]^2$$

Since $\mathbb{E}[\, y^2 \,|\, x\,] = \text{Var}([\, y \,|\, x\,]) - \mathbb{E}[\, y \,|\, x\,]^2$ we can rewrite the equation to:

$$Var(y) = \mathbb{E}[\text{Var}[\, y \,|\, x\,]] + (\mathbb{E}[\mathbb{E}[\, y \,|\, x\,]^2] - \mathbb{E}[\mathbb{E}[\, y \,|\, x\,]]^2)$$

Which we simplify to get:

$$\text{Var}(y) = \text{Var}(\mathbb{E}[\, y \,|\, x\,]) + \mathbb{E}[\text{Var}(\, y \,|\, x\,)]$$

2.2.1 Uncertainty in a Dynamics Model

Let the parameters θ be sampled from a probability distribution $q(\theta)$, s.t. $\theta \sim q(\theta)$

$$Var[s_{t+1}|s_t, a_t] = Var(\mathbb{E}_{(s_{t+1}\sim p)}[(s_{t+1}|s_t, a_t, \theta)]) + \mathbb{E}_{\theta\sim q(\theta)}[Var(s_{t+1}|s_t, a_t, \theta)]$$

where p is the state distribution under the policy parameterized by θ Here the first term $Var(\mathbb{E}_{(s_{t+1}\sim p)}[(s_{t+1}|s_t, a_t, \theta)])$ represents the variance in the expected value of the predicted transition $(s_{t+1}|s_t, a_t)$ given the parameters θ due to the stochasticity in θ which is a measure of disagreement between the models, thus it represents epistemic uncertainty.
The second term $\mathbb{E}_{\theta\sim q(\theta)}[Var(s_{t+1}|s_t, a_t, \theta)]$ represents the expected uncertainty in predicting the transition $(s_{t+1}|s_t, a_t)$ over all the models. This uncertainty of individual model represents that model's prediction of the stochastic nature of the dynamics of the environment, and the average of this uncertainty gives a way to calculate the overall stochasticity or the aleatoric uncertainty in the dynamics.

2.2.2 Estimating the Terms

We assume that the BNN outputs the parameters of a probability distribution instead of the actual state output. Using this we can estimate the first term as the variance in the mean value of the probability distribution over all θ parameters.
The second term can be estimated by calculating the variance of each θ and then calculating the average over all θ.

2.2.3 Connection with Model-Based RL

In Model-Based RL we can use an ensemble of networks instead of BNN like we did in the last assignment. In that case instead of sampling various θ we already have all out networks, and we can sample the output of these networks to estimate the two uncertainties. In this case the degree of agreement between the different networks in the ensemble gives us an estimate of the epistemic uncertainty, whereas the average variance or average uncertainty in the prediction of these models give an estimate of the aleatoric uncertainty.

2.2.4 Connection with Exploration Bonuses

Exploration bonuses help in making the agent explore more of the environment and not just stick to the first good enough policy it comes across. The epistemic uncertainty kinds of give a measure of the non visited states in the environment, so it seems like adding exploration bonuses during the training of a RL agent can help in reducing epistemic uncertainty. On the other hand aleatoric uncertainty is due to the stochasticity of the environment and it can not be reduced by adding exploration bonuses. We can try to better estimate this so we know when is out model uncertain about its prediction.

Problem 3: Bayesian Neural Networks

3.1 Uncertainty in BNNs

Let the network weights θ be sampled from the distribution $q_\phi(\theta)$.

$$\mathbb{E}_{\theta\sim q_\phi(\theta)}[y|x] = \mathbb{E}_{\theta\sim q_\phi(\theta)}[\mathbb{E}_{y\sim q(y|x,\theta)}[y|x,\theta]]$$

The variance of the prediction can be written as,

$$Var(y|x) = Var(\mathbb{E}_{\theta\sim q_\phi}[y|x,\theta]) - \mathbb{E}[Var(y|x,\theta)]$$

3.2 An Objective for BNNs

KL divergence between $p(y|x)$ and $q_\phi(y|x)$ can be written as

$$\begin{aligned} D_{KL}(p||q) &= \mathbb{E}_{(x,y)\sim p}\left[\log\left(\frac{p(y|x)}{q_\phi(y|x)}\right)\right] \\ &= \mathbb{E}_{(x,y)\sim p}[\log(p(y|x))] - \mathbb{E}_{(x,y)\sim p}[\log(q_\phi(y|x))] \end{aligned}$$

The BNN objective can be written as minimizing the above KL divergence wrt ϕ. As the first term on the RHS is independent of ϕ we just need to minimize the second term in the RHS.

$$\begin{aligned} \min_\phi D_{KL}(p||q) &= \min_\phi \mathbb{E}_{(x,y)\sim p}[\log(p(y|x))] - \min_\phi \mathbb{E}_{(x,y)\sim p}[\log(q_\phi(y|x))] \\ \min_\phi D_{KL}(p||q) &= -\min_\phi \mathbb{E}_{(x,y)\sim p}[\log(q_\phi(y|x))] \\ \min_\phi D_{KL}(p||q) &= \max_\phi \mathbb{E}_{(x,y)\sim p}[\log(q_\phi(y|x))] \end{aligned}$$

3.3 REINFORCE for BNNs

Objective function:

$$J(\phi) = \mathbb{E}_{(x,y)\sim p}[\log(q_\phi(y|x))]$$
$$\nabla_\phi J(\phi) = \mathbb{E}_{(x,y)\sim p}[\nabla_\phi \log(q_\phi(y|x))] = \mathbb{E}_{(x,y)\sim p}\left[\frac{\nabla_\phi q_\phi(y|x)}{q_\phi(y|x))}\right]$$

where,

$$\nabla_\phi q_\phi(y|x) = \nabla_\phi \mathbb{E}_{\theta\sim q_\phi}[q(y|x,\theta)] = \int_\theta \nabla_\phi q_\phi(\theta) q(y|x,\theta) d\theta$$
$$= \int_\theta q_\phi(\theta) \nabla_\phi \log q_\phi(\theta) q(y|x,\theta) d\theta$$
$$\nabla_\phi q_\phi(y|x) = \mathbb{E}_{\theta\sim q_\phi}[\nabla_\phi \log q_\phi(\theta) q(y|x,\theta)]$$

Substituting it back in the original equation:

$$\nabla_\phi J(\phi) = \mathbb{E}_{(x,y)\sim p}\left[\frac{\mathbb{E}_{\theta\sim q_\phi}[\nabla_\phi \log q_\phi(\theta) q(y|x,\theta)]}{\mathbb{E}_{\theta\sim q_\phi}[q(y|x,\theta)]}\right]$$

We can do sampling to estimate the above expectations:

$$\nabla_\phi J(\phi) = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{\frac{1}{N_\theta}\sum_{j=1}^{N_\theta}[\nabla_\phi \log q_\phi(\theta_j) q(y_i|x_i,\theta_j)]}{\frac{1}{N_\theta}\sum_{k=1}^{N_\theta}[q(y_i|x_i,\theta_k)]}\right]$$

To maximize the objective we update our parameters ϕ:

$$\phi = \phi + \nabla_\phi J(\phi)$$

3.4 Variational Inference for BNNs

Jensen inequality for a concave function f(x): $\mathbb{E}_{x\sim p}[f(x)] \leq f(\mathbb{E}_{x\sim p}[x])$

$$J(\phi) = \mathbb{E}_{(x,y)\sim p}[\log(q_\phi(y|x))] = \mathbb{E}_{(x,y)\sim p}[\log \mathbb{E}_{\theta\sim q_\phi}[q(y|x,\theta)]]$$
$$J(\phi) \geq \mathbb{E}_{(x,y)\sim p}[\mathbb{E}_{\theta\sim q_\phi}[\log q(y|x,\theta)]]$$
$$J(\phi) \geq \hat{J}(\phi)$$

where, $\hat{J}(\phi) = \mathbb{E}_{(x,y)\sim p}[\mathbb{E}_{\theta\sim q_\phi}[\log q(y|x,\theta)]]$
Now we will try to optimize $\hat{J}(\phi)$.

$$\begin{aligned}
\nabla_\phi \hat{J}(\phi) &= \mathbb{E}_{(x,y)\sim p}[\nabla_\phi \mathbb{E}_{\theta\sim q_\phi}[\log q(y|x,\theta)]] \\
&= \mathbb{E}_{(x,y)\sim p}\left[\nabla_\phi \int_\theta q_\phi(\theta)\log q(y|x,\theta)d\theta\right] \\
&= \mathbb{E}_{(x,y)\sim p}\left[\int_\theta \nabla_\phi q_\phi(\theta)\log q(y|x,\theta)d\theta\right] \\
&= \mathbb{E}_{(x,y)\sim p}\left[\int_\theta q_\phi(\theta)\nabla_\phi \log q_\phi(\theta)\log q(y|x,\theta)d\theta\right] \\
&= \mathbb{E}_{(x,y)\sim p}[\mathbb{E}_{\theta\sim q_\phi}[\nabla_\phi \log q_\phi(\theta)\log q(y|x,\theta)]]
\end{aligned}$$

We can do sampling to estimate the above expectations:

$$\nabla_\phi \hat{J}(\phi) = \frac{1}{N}\sum_{i=1}^{N}\left[\sum_{j=1}^{N_\theta}[\nabla_\phi \log q_\phi(\theta_j)\log q(y_i|x_i,\theta_j)]\right]$$

To maximize this objective we update our parameters ϕ:

$$\phi = \phi + \nabla_\phi \hat{J}(\phi)$$

Problem 4: LQR and iLQR

4.1 LQR

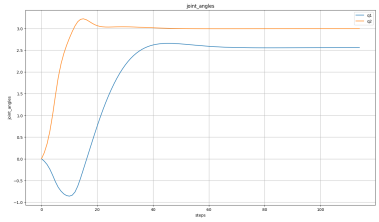

Figure 1: Joint angles vs steps

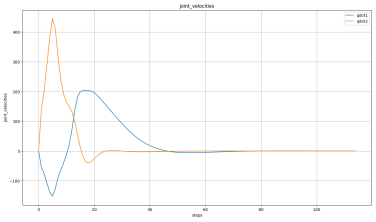

Figure 2: Joint velocities vs steps

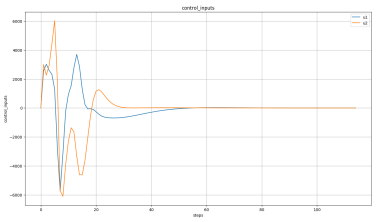

Figure 3: Control inputs vs steps

The total reward obtained was -587.12 and the total number of steps taken were 115

4.2 iLQR

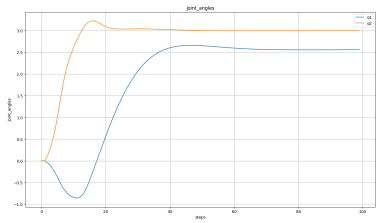

Figure 4: Joint angles vs steps

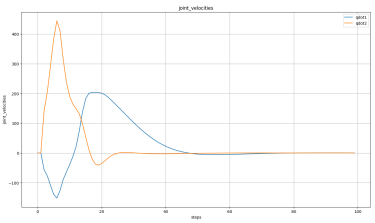

Figure 5: Joint velocities vs steps

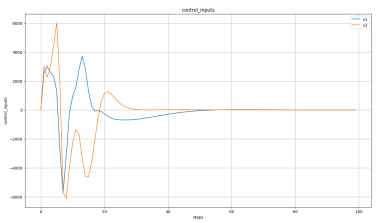

Figure 6: Control inputs vs steps

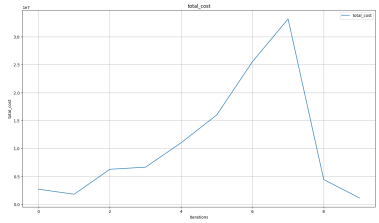

Figure 7: Total cost

The above figures were obtained using LQR output action sequence as the initial input sequence for iLQR. The total reward obtained for this case was -602.67 and the total number of steps taken were 100

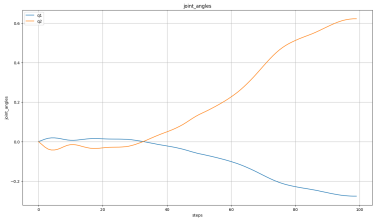

Figure 8: Joint angles vs steps

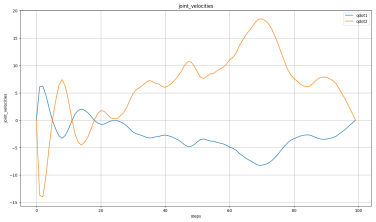

Figure 9: Joint velocities vs steps

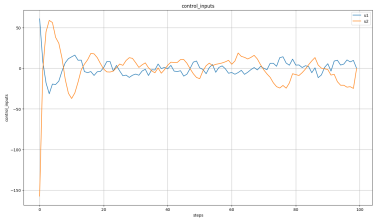

Figure 10: Control inputs vs steps

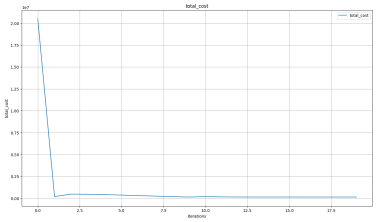

Figure 11: Total cost

The above figures were obtained using a random action sequence as the initial input sequence for iLQR. The total reward obtained was -1484.18 and the total number of steps taken were 100

4.3 Comparing LQR and iLQR

We expected iLQR to achieve better rewards than LQR, but there seems to be some bug in our code which is not allowing that to happen. If we start with LQR output as the initial input to iLQR we end up with very similar results to LQR as the iLQR does not seem to go for many iterations, but for random initial input the performance is bad.
But apart from this we expected iLQR to perform better as we have non linear dynamics which we are trying to linearize and in this case an iterative solver like iLQR can lead better and low error performance.