Data-Driven Techniques: 3$^{\text{rd}}$ Homework

Problem 3.1: Consider the pair of random variables $(\mathcal{Y}, \mathcal{X})$ related through the data model $\mathcal{Y} = 0.8\,\mathcal{X} + \mathcal{W}$ where $\mathcal{W}$ is $\mathcal{N}(0,1)$ (standard Normal). We are interested in computing the following conditional expectations $\mathsf{E}_{\mathcal{Y}}[\mathcal{Y}|\mathcal{X} = X]$ (corresponding to $\mathsf{G}(Y) = Y$) and $\mathsf{E}_{\mathcal{Y}}[\min\{1, \max\{-1, \mathcal{Y}\}\}|\mathcal{X} = X]$, (corresponding to $\mathsf{G}(Y) = \min\{1, \max\{-1, Y\}\}$). a) Use the numerical method introduced in Lecture 11 to compute numerical estimates for the two functions. b) Generate independent realizations $\{(Y_1, X_1), \ldots, (Y_N, X_N)\}$ where $\{X_i\}$ are i.i.d. standard Normal and $\{Y_i\}$ the corresponding Y-samples generated by the data model. Estimate the two conditional expectations using the neural network approach we proposed in Lecture 11. Note that the first conditional expectation is unbounded therefore provide a data-driven estimate using [A1],[A2]. For the second conditional expectation first prove that $\mathsf{E}_{\mathcal{Y}}[\mathsf{G}(\mathcal{Y})|\mathcal{X} = X]$ is bounded in $[-1, 1]$, so use use [A1] and [C1]. For [C1] consider the general version where the range is of the form $[a, b]$ which is presented in the 2024 paper contained in lecture11.zip. Limit yourselves to simple shallow networks with single hidden layer of size m. Experiment with $N = 500$ realizations and model size $m = 50$. Compare with the numerical solution by plotting your estimates. Plot your functions within the interval defined by the largest and smallest sample X_i (rounded to the nearest largest integer). Outside this interval the approximation is bad.

Common to Problems 3.2 & 3.3: Consider a Markov decision process that has two possible dynamics (or transition densities) depending on the action value α: For $\alpha = 1$ we have $\mathcal{S}_{t+1} = 0.8\,\mathcal{S}_t + 1.0 + \mathcal{W}_t$; and for $\alpha = 2$ we have $\mathcal{S}_{t+1} = -2.0 + \mathcal{W}_t$, where $\{\mathcal{W}_t\}$ in both cases is i.i.d. standard Gaussian. If at time t we are at state $\mathcal{S}_t$, depending on the selected action $\alpha_t = 1$ or 2 we generate the next state $\mathcal{S}_{t+1}$ using the dynamics corresponding to action α_t. Additionally, every time we generate a new state $\mathcal{S}_{t+1} = S$ we receive a reward $\mathcal{R}(S) = \min\{2, S^2\}$. We would like to solve the optimal average reward problem we discussed in Lecture 12 for $K = 2$. In other words we would like to find the optimal action policy that leads to optimal reward. We will consider the numerical and the data-driven methods. For the numerical method we follow the presentation in Lecture 11 or Lecture 12. For the data-driven approach we generate $N = 1000$ ***random*** realizations for the action sequence $\{\alpha_t\}$. Each action α_t prompts us to use the corresponding dynamics to go from $\mathcal{S}_t$ to $\mathcal{S}_{t+1}$ thus generating a total of 1001 realizations, starting with $\mathcal{S}_1$ being standard normal. Next we form a set by collecting all pairs $(\mathcal{S}_t, \mathcal{S}_{t+1})$ that were generated using action $\alpha_t = 1$, then a second set by collecting all pairs generated by action $\alpha_t = 2$. The data of these sets will be used for training in the next two problems. Regarding the data-driven approach which will be applied in the next two problems use only shallow networks of size $m = 100$ and if you perform training by optimizing with your own code, for simplicity use the Stochastic Gradient method.

Problem 3.2: Consider the Short-sighted version of the reinforcement learning problem. Apply the numerical method to compute the optimal functions $\mathsf{v}_1(S), \mathsf{v}_2(S)$ and then the data-driven method using the previous two sets of pairs to obtain approximations $\omega\big(\mathsf{u}(S, \theta_{\mathsf{o}}^1)\big)$, $\omega\big(\mathsf{u}(S, \theta_{\mathsf{o}}^2)\big)$ (consult Lecture 12). For the data-driven approach apply [A1] and [C1]. For [C1] show first that the conditional expectations we are interested in, take values in the interval [0,2] and use the general version of [C1] mentioned in the 2024 paper for an interval $[a, b]$. Compare graphically the numerical and the data-driven estimates of the optimal functions. Discuss your results and state explicitly the optimal and approximately optimal action policy.

Problem 3.3: Consider now the case of Infinite future reward with exponential discount factor $\gamma = 0.8$. We end up with a system of $K = 2$ equations in $\mathsf{v}_1(S), \mathsf{v}_2(S)$ (Slide 16, Lecture 12) which we must solve first numerically and then using the data-driven approach. For the numeric solution we have the matrix/vector equation (Slide 17, Lecture 12) which can be solved by iterating over the vectors $\mathsf{V}_1, \mathsf{V}_2$ until they converge. We start with zero vectors. For the data-driven approach we apply the method presented in Slides 18,19, Lecture 12 and identify the approximations $\omega\big(\mathsf{u}(S, \theta_o^1)\big), \omega\big(\mathsf{u}(S, \theta_o^2)\big)$ of $\mathsf{v}_1(S), \mathsf{v}_2(S)$. Again we employ [A1],[C1]. For [C1] we use Slide 21 to first identify proper bounds for $\mathsf{v}_1(S), \mathsf{v}_2(S)$ and use them in the general version of [C1] mentioned in the 2024 paper for an interval $[a, b]$. Compare graphically the numerical and the data-driven estimates of the optimal functions. Discuss your results and state explicitly the optimal and approximately optimal action policy.

Attention!

- Please submit your report **ONLY if you are taking the class for credit and your name appears in Progress**. Reports with names not appearing in Progress will not be evaluated.
- You must upload your report to E-class by Saturday 11 January (2025), **BEFORE MIDNIGHT**. **There will be absolutely no extension to this deadline.**
- The ONLY acceptable file type is PDF. Name your file as follows: firstname-lastname-3.pdf
- In the first page of your report write your full name and university code (Arithmos Mitroou) legibly.
- DO NOT send Python or Matlab code in separate files. Include your code in the pdf file as text **AFTER the presentation of your results of ALL the problems**. Code only, is not considered presentation of results and if your report contains only code then your grade will be 0.
- Do not simply report final outcomes. Since you will be using gradient descent algorithms, present the learning curves where we can see how the algorithms converge.
- Please make sure your files are small to avoid communication problems because of excessive size.
- **DO NOT SEND YOUR REPORT OR PART OF YOUR REPORT DIRECTLY TO MY PERSONAL EMAIL OR TO PROF. PSARAKIS**, it will not be accepted. It is part of your training to learn how to comply with hard deadlines!!!