UNIVERSITY OF PENNSYLVANIA

ESE 546: PRINCIPLES OF DEEP LEARNING

FALL 2020

[09/02] HOMEWORK 0

DUE: 09/09 WED, 1.30P ET

Changelog:

- **09/01 10p: This place will list changes to questions in the homework along with a timestamp. If you are stuck on a question, do check Canvas to see if there a homework PDF there with a more current changelog.**
- **09/08 1p: Minor change to Problem 5. Modified equation defining the term $H(Y|X)$, where y was changed to $f(y)$.**

Instructions

Read the following instructions carefully before beginning to work on the homework.

- You will submit solutions typeset in LaTeX on Gradescope (strongly encouraged). You can use hw_template.tex on Canvas in the "Homeworks" folder to do so. If your handwriting is *unambiguously legible*, you can submit PDF scans/tablet-created PDFs.
- Please start a new problem on a fresh page and mark all the pages corresponding to each problem. Failure to do so may result in your work not graded completely.
- Clearly indicate the name and Penn email ID of all your collaborators on your submitted solutions.
- For each problem in the homework, you should mention the total amount of time you spent on it.
- You can be informal while typesetting the solutions, e.g., if you want to draw a picture feel free to draw it on paper clearly, click a picture and include it in your solution. Do not spend undue time on typesetting solutions.
- For each homework, you will see an entry of the form "HW 0 PDF". You will also see entries like "HW 0 Problem 7 Code" or "HW 0 Problem 3c Code". There will be multiple "code" entries if there are multiple problems involving programming in the homework. You will upload your solutions in a PDF format to "HW 0 PDF". **For each programming problem/sub-problem,**

you should create a fresh Google Colab notebook. This notebook should contain all the code to reproduce the results of the problem/sub-problem. You will upload the .ipynb file obtained from Colab as your solution for "HW 0 Problem 7 Code" or "HW 0 Problem 3c Code". Name your notebook to be pennkey_hw0_problem7.ipynb, e.g., I will name my code for Problem 7 as pratikac_hw0_problem7.ipynb.

  - To be safe, make sure that the option "Omit code cell output when saving this notebook" is unchecked in the "Notebook Settings" tab in Colab. This way, the instructors may be able to give you partial credit for incorrect output/plots.
  - **This is very important**. Note that the instructors will download your notebook and execute it on Colab themselves, so your notebook should be such that it can be executed independently without any errors to create all output/plots required in the problem.

Notice that even if Problem 7 contains multiple sub-parts, you are required to submit only one Colab notebook. In these cases, you will demarcate clearly within your notebook what is the output cell of each sub-problem.

Credit

The points for the problems may add up to more than 100. If so, you only need to solve for 100 points to get full credit, i.e., your final score will be $\min(\text{your total points}, 100)$.

Problem 1 (10 points). Consider a function $f : A \times B \to \mathbb{R}$ where A and B are non-empty sets.

(a) (5 points) Assuming that sets A and B are finite show that

$$\max_{x \in A} \min_{y \in B} f(x, y) \leq \min_{y \in B} \max_{x \in A} f(x, y).$$

(b) (5 points) For general non-empty (not necessarily finite), show that

$$\sup_{x \in A} \inf_{y \in B} f(x, y) \leq \inf_{y \in B} \sup_{x \in A} f(x, y).$$

Problem 2 (15 points). For the function

$$f(x) = 2x_1^2 - 1.05x_1^4 + \frac{1}{6}x_1^6 - x_1x_2 + x_2^2$$

(a) (5 points) find the global minima in the region $-3 \leq x_1 \leq 3$ and $-3 \leq x_2 \leq 3$.

(b) (5 points) are there any other stationary points? If yes, what are they?

(c) (5 points) plot the contour plot of $f(x)$ and verify your answers to the previous two questions.

Problem 3 (15 points). For the loss function $f(x, y) = x^2 + y^2 - 6xy - 4x - 5y$

(a) (5 points) show analytically using Lagrange multipliers how to minimize the loss subject to constraints

$$\begin{aligned} &y \leq -(x-2)^2 + 4, \text{ and} \\ &y \geq -x + 1. \end{aligned}$$

(b) (5 points) how is the optimal loss affected if the first constraint is changed to

$$y \leq -(x-2)^2 + 4.1.$$

Estimate the difference and explain your answer. Remember that the gradient of the loss at a stationary point is a linear combination of the constraints with the Lagrange multipliers acting as the co-efficients.

(c) (5 points) Write a Python script to confirm your results to parts (a) and (b). You can use the Scipy function scipy.optimize.minimize (https://docs.scipy.org/doc/scipy/reference/optimize.html) to perform this constrained optimization. List the code and your results in the solution PDF.

Problem 4 (15 points). Let X and Y be independent random variables taking values in $\mathcal{X} = \{-1, 1\}$. We have $\mathbb{P}(X = 1) = q$ while Y is uniformly distributed on $\{-1, 1\}$. Let $Z = XY$.

(a) (5 points) Find the conditional distribution $\mathbb{P}(Y|Z)$. Note that conditional probabilities are a function of $Z = z$.

(b) (5 points) The conditional mean of Y given $Z = z$ is

$$\mathbb{E}[Y|Z = z] = \sum_{y \in \mathcal{X}} y \, \mathbb{P}(y|z).$$

Find $\mathbb{E}[Y|Z = z]$ as a function of z.

(c) (5 points) The conditional mean of Y given Z (where the latter is viewed as a random variable) is

$$\mu_{Y|Z} = \mathbb{E}[Y|Z] = \sum_{y \in \mathcal{X}} y\, \mathbb{P}(y|Z).$$

Since $\mu_{Y|Z}$ is a function of the random variable Z, it too is a random variable. Compute the probability distribution of $\mu_{Y|Z}$.

Problem 5 (10 points). Consider the zero-mean jointly Gaussian random variables X and Y with covariance matrix

$$\begin{bmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{bmatrix},$$

i.e.,

$$(X, Y) \sim f(x, y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2} - \frac{2\rho xy}{\sigma_x\sigma_y}\right)\right).$$

(a) (5 points) Find the conditional density $f(x|y)$.

(b) (5 points) Using (a), find the differential entropy of the conditional distribution $X|Y$, i.e., $H(X|Y)$. Note that for a univariate Gaussian random variable X with variance σ_x^2, the differential entropy is given by

$$H(X) = \frac{1}{2}\log\left(2\pi e\sigma_x^2\right).$$

Once you have $H(X|Y = y)$, you can compute the entropy $H(X|Y)$ using

$$H(X|Y) = \int f(y)\, H(X|Y = y)\, \mathrm{d}y$$

Problem 6 (10 points). Consider three independent random variables X_1, X_2, X_3 distributed uniformly on $[0, 1]$. What is the probability that sticks of length X_1, X_2, X_3 form a triangle? That is, $X_1 \leq X_2 + X_3$, etc.

Problem 7 (25 points). In this problem, we will run linear regression using the "scikit-learn" library. You can install this library using

```
[local] pip install scikit-learn
```

Google Colab already has scikit-learn installed.

Load the Boston housing dataset (https://www.cs.toronto.edu/d̃elve/data/boston/bostonDetail.html) using

```
from sklearn.datasets import load_boston
ds = load_boston()

# explore the dataset
print(ds.keys())
print(ds.DESCR)
```

The variable ds.data contains the features with names ds.feature_names and the variable ds.target contains the value of the house. Our goal is to predict the value given the features using a linear model.

(a) (15 points) If the i^{th} datum is denoted by $x^i \in \mathbb{R}^{13}$ and target by $y^i \in \mathbb{R}$, we intend to solve for weights $w \in \mathbb{R}^{13}$ and $b \in \mathbb{R}$ such that

$$y^i \approx w^\top x^i + b$$

for all pairs (x^i, y^i) dataset. A natural way to find w, b is to minimize the average of the *residuals*

$$\ell(w, b) := \frac{1}{2n} \sum_{i=1}^{n} \left(y^i - w^\top x^i - b \right)^2$$

where $n = 506$ is the number of samples in the dataset. Write the *data matrix* as $X \in \mathbb{R}^{506 \times 13}$ and the target matrix as $Y \in \mathbb{R}^{506}$; each row of X is therefore a datum x^i. We can now rewrite $\ell(w, b)$ as

$$\ell(w, b) = \frac{1}{2n} \| Y - Xw - b\mathbb{1} \|_2^2$$

where $\mathbb{1}$ is a vector of all ones. Compute an analytical expression for

$$w^*, b^* = \operatorname*{argmin}_{w,b} \ell(w, b).$$

(b) (10 points) Code up your expression for w^*, b^*. In order to check whether the linear model is accurate for the Boston housing dataset, we should evaluate the model on some held-out data. If the model predicts the values on this held-out data accurately, i.e., has a small residual, we can confidently use the linear model in practice. Split the 506 samples into 80% (= 405) samples that are used for computing w^*, b^* as above. Compute the average of the residuals on the remaining 20% of the data (=101 samples); this is known as the validation error because the model w^*, b^* was not fitted on this data. Similarly the average of the residuals, namely $\ell(w, b)$, on the 80% data used to fit the weights is called the training error. Perform this experiment 2-3 times, each time sampling a different training set of size 80% from the original dataset. Report the mean and standard-deviation of the training and validation error.