CS 37300	— Homework 1—	Fall 2025
Due date on Gradescope		

General/Submission Instructions

Your answers must be submitted as a **PDF** where **each page contains only one question**. A typed PDF using the provided tex template is required, you can compile the tex file here. For full credits, make sure to include complete answers, with all the necessary derivation steps.

This homework contains two parts: a set of theoretical questions and a set of programming questions. Both parts are to be submitted via Gradescope. Some of the programming questions may require you to include plots and/or analysis on your PDF report, while the other questions will be autograded in Gradescope.

For this programming assignment, download the provided `hw1.zip` file on **Brightspace**. You can **only** use `numpy`, `pandas` and `matplotlib`. You may see other libraries imported, but those are for testing/debugging purposes, and therefore you should not use any of them to "simplify" your solution.

Q1 (25 pts): Theoretical Questions

1. (8 pts) True or False *For full credits, include a justification or example in your answer.*

(a) A continuous distribution's density can take values exceeding 1 at some points.

(b) For a real-valued random variable X with $F(t) = P(X \leq t)$, it is possible for the cumulative distribution function to exceed 1.

(c) Suppose a pair of real random variables has joint density

$$f(X, Y) = 7XY; \quad 0 \leq X \leq 1, \quad 0 \leq Y \leq 2.$$

Then X and Y are independent.

(d) If A and B are conditionally independent given Z, meaning $P(A, B \mid Z) = P(A \mid Z)P(B \mid Z)$, then conditioning further on B does not change A's conditional probability: $P(A \mid B, Z) = P(A \mid Z)$.

2. (4 pts) A reliability study tracks the lifetimes (in months) of 60 identical batteries: 5 fail in $[60, 69]$, 25 in $[70, 79]$, 20 in $[80, 89]$, and 10 in $[90, 99]$. Using this sample, estimate the following quantities

(a) What is the estimated probability $P(A)$ that a randomly chosen battery lasts more than 69 months?

(b) What is the estimated probability $P(B)$ that a randomly chosen battery lasts more than 79 months?

3. (4 pts) An box contains an equal number of red and blue balls. Draw with replacement repeatedly until the first red ball appears.

(a) What is the sample space for this experiment? What is the probability that the first red appears on the i-th draw?

(b) Let E be the event that the first red appears after an even number of draws. Which outcomes constitute E? What is $P(E)$?

4. (4 pts) Determine the unique matrix A^{-1} satisfying $AA^{-1} = I$ for

$$A = \begin{bmatrix} 5 & 4 & -7 \\ 4 & 3 & -5 \\ 2 & 4 & 1 \end{bmatrix}$$

5. (5 pts) Independence

A small game is played with three fair, independent "switches." Each switch $i \in \{1, 2, 3\}$ outputs a bit $T_i \in \{0, 1\}$ with $P(T_i = 1) = P(T_i = 0) = \frac{1}{2}$, independently across i (i.e., $T_i \sim \text{Bernoulli}(\frac{1}{2})$). The game then lights up three indicators defined by

$$X = T_1 \oplus T_2, \qquad Y = T_2 \oplus T_3, \qquad Z = T_3 \oplus T_1,$$

where $\oplus$ denotes the XOR (exclusive OR) operator. Show that the random variables X, Y, Z are pairwise independent but not mutually independent.

Q2 (20 pts): Numpy Basics

1. (20 pts) Implementation

In the file `numpy_basics.py`, use `numpy` to implement the following functions. You may refer to the `numpy` documentation at: https://numpy.org/doc/stable/reference/routines.html.

(a) **(4 pt)** Implement `create_zero_matrix()` function, which takes in two values for the number of rows and the number of columns to create. Please write clearly and concisely - clarity and brevity will be rewarded. Refer to known facts as necessary. a matrix (2-D array) filled with 0's. You may assume both input numbers are non-negative.

(b) **(4 pt)** Implement `create_vector()` function, which takes in an integer $n \geq 0$ and returns a vector (1-D array) of length `size` with random integers between 0 (inclusive) and n^2 (exclusive).

(c) **(4 pt)** Implement `calculate_matrix_inverse()` function, which takes in a matrix and returns the inverse matrix. You can assume the matrix is full rank.

(d) **(4 pt)** Implement `calculate_dot_product()` function, which takes in two vectors and returns the dot product of them.

(e) **(4 pt)** Implement `solve_linear_system()` function, which takes in an array and a vector and returns the solution of the linear system $\mathbf{Ax} = \mathbf{b}$. You can assume the linear system has a unique solution.

Q3 (25 pts): Yelp Dataset

You will work with the Yelp dataset in the question. This dataset is part of the Yelp academic dataset and consists of data about restaurants. Please review the dataset called `yelp.csv` under the dataset folder.

Dataset description: It contains 28 attributes: 6 numeric and 22 discrete. The first row of the file is a header row with the names of the attributes where names are separated by a comma (,).

1. (20 pts) Implementation

In `yelp.py`, implement the following functions. You should use the `pandas` library. You may refer to `pandas` documentation at: https://pandas.pydata.org/docs/reference/index.html.

(a) **(4 pts)** Implement `read_data()` function, which reads data from the given file path and return a pandas data frame.

(b) **(4 pts)** Implement `average_rating()` function, which calculates the average star ratings of restaurants in each state, and then save the ratings in a lists in alphabetical order of the state abbreviation, starting with `AZ` in ascending order. Return a `numpy` array containing the average star ratings.

(c) **(4 pts)** Implement `rating_stats_given_review_count()` function, which calculates the mean and the standard deviation of the star ratings of restaurants that have at least that many ratings. Return a tuple of two floats representing the mean and the standard deviation.

(d) **(4 pt)** Implement `plot_cdf()` function, which generates a plot of the Cumulative Distribution Function of the review count for restaurants in Nevada (state code `NV`).

This is a curve that describes, for a value x, the proportion of restaurants in Nevada that have at most x reviews. As the count values can vary a lot, it is a good idea to use a logarithm scale for the x-axis, that is, instead of having the equally spaced ticks in the axis indicate sequential numbers (e.g. $1, 2, 3, \cdots$), they indicate values whose logarithm has that progression (e.g. $10^0, 10^1, 10^2, \cdots$). In `matplotlib`, check the `ax.set_xscale` function. Use a logarithm scale for the x-axis and make sure to include proper labels in both axis and a descriptive title to receive full points.

(e) **(4 pt)** Implement `make_boxplots` function, which aims at generating a plot containing boxplots that describe the distribution of number of checkins for each possible star rating. There should be 9 boxplot, side by side in the figure, one for each possible star value. Use logarithm scale for the y-axis. Make sure to include proper labels for the axis and a descriptive title to receive full points.

2. (5 pts) Evaluation

You are given a function called `evaluate_yelp()` in `yelp.py` to test your code. **If your code runs without errors, you will be able to generate the cdf and box plots**.

Please make sure these plots are in the submitted report (Submitted through Gradescope).