The Hashemite University
Prince Al-Hussein bin Abdullah II Faculty for IT
Department of Information Technology
Advance Programming for DS and AI (2010042351)

Assignment 2: Numpy Array Operations
Due Date: Saturday 6-1-2024 11:59 PM
Max score: 20 points

Instructions:

1- Submit **one** python script file ({filename}.ipynb.
2- File naming: name your python script file with your [*your university ID]*. For example, if your team has two members the file's name must be their ID's separated by _.
3- send your code file to your instructor through the designated assignment.
4- Please keep in mind that **late submissions** will result in a **ZERO** score.
5- You must be able to discuss the details of your solution with your instructor.

The Dataset

In-hospital mortality in ICU patients with heart failure

The predictors of in-hospital mortality for intensive care units (ICU)-admitted HF patients

Format
1177 rows x 81 columns:

- **Row 1:** features names,
- **Column 1:** patient ID

Data Info can be found in https://www.kaggle.com/datasets/saurabhshahane/in-hospital-mortality-prediction

Requirements:

The goal of this assignment is to ensure a thorough understanding of Numpy array operations. Throughout this task, you will:

1. Read the "heartFailureDataset.txt" dataset into a Numpy array. You can use the appropriate Numpy function for reading tabular data.

   Hint: Create a list or Numpy array to store the feature names, and use another Numpy array to store the data.

2. Calculate the mean and standard deviation of the **"heart rate"** values for each **"gender"** across all patients. Then, identify the "gender: with the lowest mean **heart rate**.

3. Calculate the median and interquartile range of all features. Then, identify the feature with the smallest and largest interquartile range.

4. Using the np.sum function, calculate the total **“blood pressure”** (“Systolic blood pressure” and “Diastolic blood pressure”) value for each patient. Then, print the top 10 “patient IDs” with the highest **blood pressure** values.
5. Sort the patients in descending order based on their **“glucose”** values. Print the patients IDs the patient IDs of the top 10 patients based on their **"glucose"** values.
6. Using fancy indexing, substitute the **"Creatinine"** values of the top 5 patients, with the corresponding median value of that feature.
7. Using np.argpartition, identify the top 100 values of each feature. Subsequently, compute the mean of each feature, considering only the 100 highest values.
8. Calculate the mean and standard deviation of “**Respiratory rate**” values. Then, identify the patient IDs with 2 standard deviation away from the mean of “**Respiratory** rate” feature.
9. Compute the distance matrix among patients and determine the k-NN (k-nearest neighbors), where k = 3, for each patient using the np.argpartition function.
10. Compute the Pearson correlation coefficient between each pair of patients. Subsequently, identify the pair of patients with the highest correlation coefficient. Provide the correlation matrix, which should be a square matrix.

* You will be asked to provide clear explanations and interpretations of your results.

---------- Good Luck ----------