Dr. Eick

COSC 3337 *"Data Science I"* Fall 2022
Problem Set1

Last Updated: September 21, 10p

Task1: Creating a Pipeline Encapsulating Data Sampling, Data Splitting, Feature Selection, Feature Creation and Classification Steps.

Task2 Due: Friday, Sept. 30, 11:59p
Responsible TA: Navid

For this assignment, you must use dataset "PS1-Task2.xlsx" uploaded in MS Teams under "Datasets and Code" channel. This dataset has been created synthetically by the TA for learning purposes! This dataset has 8400 samples, and each sample has four features (feature1, feature2, feature 3 and feature 4) presented in first four columns. The fifth column of the dataset indicates the label of each sample. There are two classes indicated by label 0 and 1. Write your code in **Python** or other language you prefer to answer the tasks listed below.

Note: *The colors you choose for your plots must be based on your **student ID**. As the dataset includes two labels, you will need two colors. Suppose your student ID is "1234567". You will use the six right-most digits to define the first color in hex format. So, the first color is "#234567" in this example. For second color, you must subtract first color number from "FFFFFF". For your convenience, you can use following function code written in Python to generate your colors:*

```python
def plot_colors(student_id):
    color1 = "#"+student_id[1:]
    color2 = "#"+str(hex( int("FFFFFF" ,16) - int(student_id[1:],16)))[2:]
    return color1 , color2
```

Usage example:

```python
psid = "1234567"
color1,color2 = plot_colors(psid)
```

Learning objectives:

- ✓ Creating a pipeline
- ✓ Sampling technique
- ✓ Splitting data
- ✓ Data visualization
- ✓ Feature selection
- ✓ Feature creation
- ✓ Classification

Tasks:

2.1. Find the proportion of two class samples. Report number of class-0 and class-1 samples and the ratio $\frac{\#Class\ 0}{\#Class\ 1}$. **5 Points**

2.2. **Sampling (regular):** Write a function that randomly samples "q" number of samples from dataset (without replacement) and return the new created dataset. Set $q = 1000$ and report the number of class-0 and class-1 samples and the ratio $\frac{\#Class\ 0}{\#Class\ 1}$ for new created dataset. (Call this dataset dataset2). **10 Points**

2.3. **Sampling (Stratified):** Write a function that randomly samples "q" number of samples from dataset (without replacement) and preserves proportion of the number of different class samples. This sampling is called stratified sampling. Set $q = 1000$ and report the number of class-0 and class-1 samples and the ratio $\frac{\#Class\ 0}{\#Class\ 1}$ for new created dataset. (Call this dataset dataset3). **10 Points**

2.4. **Feature Selection:** Compute the covariance matrix for dataset3. Report this covariance matrix. Select two features that you think they may provide better discrimination between two classes. Report selected features (Feature #1, #2, #3 or #4) and explain your reasons. Create a new dataset including only these two features. Call this dataset dataset4. Write a function for this step. **8 Points**

2.5. **Visualization:** Obtain the supervised scatter plot for dataset4. Remember to use your personalized colors for two classes! Do not forget to adjust alpha value (transparency) to see the overlapping areas. Interpret the scatter plot. **5 Points**

2.6. **Visualization:** Obtain four histograms, one for each selected feature in dataset4 and each class instances (first selected feature for class 0 instances, first selected feature for class 1 instances, second selected feature for class 0 instances and second selected feature for class 1 instances). Remember to use your personalized colors for two classes! Discuss the difficulty of separating two classes based on the selected features. **5 Points**

2.7. **Splitting dataset**: Use the function you wrote for subtask 2.3 and select 700 samples from dataset4. Call this new dataset training_set. Call remaining 300 samples as testing_set. Note you need to modify the function in task 2.3 as it needs to return the remained samples as another dataset! **10 Points**

2.8. **Classification:** Train a decision tree with depth=3 using your training_set. Report its classification accuracy using testing_set. Submit the decision tree. **5 Points**

2.9. **Feature creation:** Write a function that accepts a dataset with two features (f_1, f_2) as its input and builds a new dataset with a new feature computed as follows,

$$f_{new} = \sqrt{f_1^2 + f_2^2}$$

Create a new training_set and testing_set by passing training_set and testing_set through this function and call them c_training_set and c_testing_set. **5 Points**

2.10. **Visualization:** Obtain two histograms for c_training_set for the new feature f_{new}, one for the instances of class 0 and one for instances of class 1. Remember to use your personalized colors for two classes! Compare your obtained results with part 2.6 and explain the reasons. **6 Points**

2.11. **Classification:** Train a decision tree with depth=3 using your c_training_set. Report its classification accuracy using c_testing_set. Compare your result with subtask 2.8 and explain the reason. **7 Points**

2.12. **Building a pipeline:** Write a function that accepts

- A dataset
- A variable specifying the sampling method (this variable can be set as "rgl" or "stf" by the user)
- A variable for number of samples in sampled dataset.
- A variable specifying the number of samples in training set

as its input and outputs the classification accuracy. (Call the functions written in previous subtasks in this function. The output of one function must be fed the next one as its input.)

(Take the dataset and all required variables → Sampling based on the selected method → Feature selection → Feature creation → Splitting new dataset to train and test → Train a decision tree with depth=3 → classification accuracy)

Report the classification accuracy for following settings:

1) Main dataset, "stf", 500, 300
2) Main dataset, "rgl", 100,70
3) Main dataset, "stf", 1500,1000

Also, discuss the advantages of using a pipeline. **12 Points**

2.13. **Write a conclusion** (at most 13 sentences!) about what you learned in this task and the problems you encountered during writing your code! **12 Points**

Remark: *Select your features carefully as the next steps depend on your selection! Most of the points of the tasks will be given to explaining the reasons behind the results. Therefore, by showing only graphs without any discussions you will get a few points of that task!*

How to submit:

*Please upload a **ZIP** file including your answers in **PDF** format and your code files in Blackboard. Your PDF report must contain your explanations and any graphs you plotted.*