DOUGLAS COLLEGE – Summer 2023

CSIS 3290 – 001 – Lab 4

- **Create a folder and name it according to naming convention stated below**
- **All the files you are required to submit for the assignment should be placed inside this folder.**
- **If cheating is determined (i.e., you shared your work with another student in the class), your work will a ZERO mark and you will face further consequences.**
- **Make sure to include all the necessary files to make sure that the code can run properly without producing any error**

In this lab, we practice using pipeline for different classifications and regressions models. You need study the textbook, demo code and do your own research to make sure that you can perform all the tasks describe below.

Part1 (classification)

1. Create a python notebook named as **Lab4_part1_ABcXXXXX** with A signifies the first letter of your **first name**, Bc signifies the first two letters of your **last name** and XXXXX denotes the last five digits of your **student ID**.
2. Create a markdown cell at the top of the Jupyter notebook to state the lab, **your name and student ID** with the correct heading.
3. **For each of the following section**, you need to create a **markdown heading cell** followed by a few code cells to complete the tasks. Please also put some comments in each code cell.
   a. **Load the python library**. Please load all the required python libraries in this section
   b. **Read the data**. Use the *Lab3_user_behavior.csv* file. . The target is the **classification** column.
   c. **Drop the unneeded data.** Check if there is any null and drop them.
   d. **Create dummy values for the dataframe**, do not forget to use the drop_first parameter.
   e. **Prepare the features and target variable**, split the dataframe into training and test such that the test size is 25% and specify the random state value.
   f. **Create a pipeline** consisting of (use hyperparameter for each classifier as needed):
      - Any scaler you want to use
      - SelectFromModel feature selection with RandomForest **Classifier** as the estimator. You can use the number of estimators 100 or less.
      - The following classifier model (you need to use a loop):
        - Logistic Regression
        - KNN
        - Linear SVC
        - SVM RBF
        - Decision Tree
        - Naïve Bayes
        - Random forest Classifier
        - AdaBoost Classifier
        - XGBoost Classifier
        - CatBoost Classifier
   g. **Select the best pipe**. Make prediction and analyze its performance by creating the confusion matrix and classification report.

Part2 (regression)

1. Create a python notebook named as **Lab4_part2_ABcXXXXX** with A signifies the first letter of your **first name**, Bc signifies the first two letters of your **last name** and XXXXX denotes the last five digits of your **student ID**
2. Create a markdown cell at the top of the Jupyter notebook to state the lab, **your name and student ID** with the correct heading.
3. **For each of the following section**, you need to create a **markdown heading cell**.
   a. **Load the python library**. Please load all the required python libraries in this section
   b. **Read the data**. Use the *Lab4_car_emission.csv* file. The target is the **emissions** column.
   c. **Drop the unneeded data.** Check if there is any null and drop them.
   d. **Create dummy values for the dataframe**, do not forget to use the drop_first parameter.
   e. **Prepare the features and target variable**, split the dataframe into training and test such that the test size is 25% and specify the random state value.
   f. **Create a pipeline** consisting of (use hyperparameter for each regressor as needed):
      - Any scaler you want to use
      - SelectFromModel feature selection with RandomForest **Regressor** as the estimator. You can use the number of estimators 100 or less. Note: if you use RandomForest Classifier, you may get different results.
      - The following regressor model (you need to use a loop):
        - Linear Regression
        - Decision Tree Regressor
        - Random Forest Regressor
        - GradientBoosting Regressor
        - AdaBoost Regressor
        - XGBoost Regressor
        - CatBoost Regressor
        - LightGBM Regressor (https://pypi.org/project/lightgbm/)
        - SGD Regressor (https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDRegressor.html)
          Note: SGD Regressor might fail to converge. You may want to play around with the max_iter, tol or eta0 hyperparameters

   g. **Select the best pipe**. Make prediction and analyze its performance by plotting the actual vs predicted values.

Note on submission:

- Create a folder named as Lab4_ABcXXXXX following the naming convention.
- Put your Jupyter notebook and the original and cleaned dataset in this folder.
- Zip the file and submit it through the blackboard

LAB/ASSIGNMENT PRE-SUBMISSION CHECKLIST

- Did you follow the naming convention for your files and folder?!
- Does your submission work on another computer?!
- Double check **before** submitting