Assignment 4: Working with Scikit Learn

Submission deadline: 23:59, Friday, Oct. 25, 2019

Assignment 4

In this assignment, you will use two machine learning algorithms from **Scikit-Learn** to perform a classification task. The dataset that you will be working with is extracted from Census database in USA in 1994. It contains personal information of people relating to their income level. You task is to determine whether a person makes over 50K a year or not, given individuals' personal information.

The assignment contains three parts. In Part 1, you will do preprocessing. In Part 2 you will use 'KNN' Classifier to predict the income level and in Part 3, you will predict the income level using a Decision Tree Classifier.

About the dataset:

The dataset file contains 16 columns; 15 attributes and one target variable which is the 'income': The income is divided into two classes: <= 50 K and >= 50K. Complete information about the attributes can be found in this link: https://archive.ics.uci.edu/ml/datasets/Adult

Note: The dataset for this assignment is available in the dataset folder.

Part 1: Pre-processing

The procedure for preprocessing is as follows:

1. Read the dataset and store it in a Pandas data frames. You need to do the following for both datasets.
2. Convert the 'income' values from **text** to **0 and 1**:(0 for <= 50 k and 1 for > 50k
3. Drop these columns which are redundant or do not contribute much to the prediction:'fnlwgt','education','capital-gain', 'capital-loss','native-country'.
4. Some columns have missing values. You can replace the missing values in each column with the most frequent occurring value of that column.
5. Transfer the categorical values to numerical values through **one-hot encoding**. You can use 'get_dummies' function of Pandas data frame and pass two arguments to it: the dataframe and a list of columns with categorical variables. You should save the results in the same dataframe.

6. Divide the dataset to train set and test set. (70 percent train and 30 percent test)
1. Define the variables for storing information on the features and target for both train and test set (Train_X, Train_y, test_X , Test_y)

Part 2. Classification with K-Nearest Neighbour Classifier (KNN)

In this part, you will use KNN Classifier from Sckit-Learn library of Python to build the classifier.

1. From sklearn.neighbors import KNeighborsClassifier.
2. Build KNN classifier using default parameters.
3. Call 'fit' function of the created model .
4. Create predictions by calling 'predict' function of the fitted model.
5. Use accuracy_score from sklearn to find the accuracy of the model.
6. Compute Confusion Matrix using SKlearn library.

Part 3. Classification with Decision Tree Classifier

To provide the predictions using the DT, you should follow the following steps:

2. From sklearn.tree import DecisionTreeClassifier.
3. Build the classifier with default parameters.
4. Call 'fit' function of the created model.
5. Create predictions by calling 'predict' function of the fitted model.
6. Use accuracy_score from sklearn to find the accuracy of the model.
7. Compute Confusion Matrix using SKlearn library.
8. Finally, compare the accuracy results with the one you gained from KNN classifier.

Grading:

• The result of grading will be either 'Passed' or 'Failed'. Your grade will be assessed on the performance and result of your code. The accepted performance means that you should at least provide 80 per cent of the solutions in each part.

Submission guideline:

• Deadline: 23:59, Friday, Oct. 25, 2019 (submit your assignment through Canvas).

• Source code submitted for the assignments should be your own code. If you use the codes from the internet or if you use someone's code without referencing, that will be treated as plagiarism and you will fail the assignment.
• Source code should be written in one Jupyter notebook files. Filenames could be the student's first name, last name and name of the assignment (firstname.lastname.timeseries).
• The assignment is individual and can NOT be solved in groups.
• The deadline cannot be extended. (No exception)
•If you have any questions, please send an email to this address: aida.mehdipourpirbazari@uis.no.