INTERNATIONAL SUMMER UNIVERSITY PROGRAMME 2020

CCMVI2085U

Machine Learning for Predictive Analytics in Business

Ordinary Examination

Date: 29 July 2020

4-hour-written-exam (9 am -1 pm)

Special instructions: You will need to create a Jupyter notebook using Python 3 on your computer to answer the exam questions. The submission is your Jupyter notebook file and you need to ensure your Python codes are clearly presented and can be executed!

Please use the following structure to create your Jupyter notebook and answer the questions.

Section 1 (10%)

Download dataset bike_sharing_rental.csv from Canvas, and show the following operations in your Jupyter notebook:

1. Load the dataset into your Jupyter notebook and name it df (2%)
2. Report the dimension of df (2%)
3. Show the last 12 observations/rows of df (2%)
4. Show the variable date types of df (2%)
5. Summarize the numeric variables of df (e.g., number of observations, mean, standard deviations, min value, max value) (2%)

Section 2 (10%)

Use the seaborn.PairGrid to draw a set of pairwise plots of all variables in df, where

1. Scatter plots in the upper triangle (in blue color) (2%)
2. Histograms on the diagonal (in gray color) (3%)
3. Correlation coefficients in the lower triangle (5%)

Section 3 (65%)

Develop machine learning models to predict casual users. Therefore, the response varaiable is casual. The following are the operations which you need to show in your Jupyter notebook:

1. Create a dictionary and name it cc. The dictionary has four keys: data, target, feature_name, target_name. The corresponding key values are NumPy arrays. (12%)
   - The key value of target is an array of values from column casual of df.
   - The key value of data is an array whose sub arrays of all other numeric columns of df.
   - The key value of feature_name is an array of numeric feature names.
   - The key value of target_name is casual
2. Randomly split the data into training and test sets. The split ratio is 80:20 (i.e., 80% for training and 20% for testing). Use your allocated random seed if applicable. Name the training input feature data to x_train; name the test input feature data to x_test; name the training target data to y_train; name the test target data to y_test. (8%)
3. Train linear regression, decision tree, random forest, neural network models using the training data and make predictions using the test data. You can either write your own functions or use third-party libraries such as sklearn to train and test your models. You can set the model hyperparameters with the values you think are appropriate and use your allocated random seed if applicable. (20%)
4. Write your own Python function to calculate the mean squared error between the model predictions and the ground truth target values. The function name is mse_new. Use your own function mse_new to report the model's prediction performance for the test data. In the meantime, show the mean squared errors calculated from your mse_new function are equal to the sklearn library's mean_squared_error function. Compare and discuss the performance of these four models. (25%)

Section 4 (15%)

You need to perform feature engineering on variable dteday of df and check if the performance of models can be improved. The following operations should be shown in your Jupyter notebook:

1. Decompose variable dteday of df into three new variables (i.e., day, month, year) into df and remove dteday from df. (3%)
2. Convert the data types of variables day, month, year into int64. (3%)
3. Train and test the linear regression, decision tree, random forest, and neural network models in Section 3 using the same data split settings and model hyperparameter settings. Show and discuss if the model prediction performance can be improved and why. (9%)