Homework 10 - Random Forest (RF)

Washington DC Bikeshare data (150 points)

Instruction:

- This is a group-work assignment!
- You are expected to submit the **.ipynb** file and the exported **.html**.
- Only one member in each group needs to submit the assignment. It will be automatically submitted for the rest of group members.

  Good luck and enjoy machine learning!

Question 1 RF regression (60 points)

In this exercise I want you to apply RF regression model to the bikeshare data set which is available on the GitHub folder for HW10. From HW-7 KNN (Washington DC Bikeshare data) we are already familiar with this data set so there is no need to do an EDA. The data are 17379 observations of hourly counts from 2011 to 2012 for bike rides (rentals) from the Capital Bikeshare system in Washington DC. It was originally compiled by Fanaee and Gama in 'Event labeling combining ensemble detectors and background knowledge' (2013). Import the bikeshare.csv as a data frame and name it df.

1. Define your feature space and target variables. Split the data into test (30%) and train set (70%) **(5 points)**
2. From sklearn.ensemble import the relevant function for RF regression. Do the followings: (25 points) **(25 points)**
    1. Train the model with the default features. However use random_state=1000. (5 points)
    2. Make predictions on the test set and save them as y_hat (5 points)
    3. Use the built-in classification report function from sklearn. Report the Accuracy, precision,Construct a data frame named df_predictions with 2 columns. y_test, and y_hat from previous part (5 points)
    4. Visualize actual vs predicted counts in the test set using an scatterplot. Are you visually satisfied with the regression model? (5 points)
    5. Report the R-squared and RMSE_test for the RFF regression model. (5 points)
3. **Tuning hyperparameters**: I specifically want you to use the following:
   param_grid. my_param_grid = n_estimators:[100,200], max_features:['log2', 'auto'], max_depth:[10,None].
   Where you able to improve the model performance? Is random forest using a small tree or bushy tree? How do yo know? **(10 points)**
4. **Cross validation and OOB observations**: estimate the R^2 of the test set using both CV and OOB methods and compare the results. **(15 points)**
5. Feature importance: Plot the feature importance graph and compare your top 5 important features with the top 5 most significant variables derived from a linear regression model. Can you name a variable which was important according to RF but not significant according to linear regression model? What's going on here? **(15 points)**

Question 2 RF classification (90 points)

The managers of Capital Bikeshare have found that the system works smoothly until more than 500 bikes are rented in any one hour. At that point, it becomes necessary to insert extra bikes into the system and move them across stations to balance loads. Do the followings:

1. Define a binary target variable overload. $Overload = 1$ if $cnt > 500$ and 0 otherwise. What are the proportions of overload vs non-overload in your data set? Is the target variable balanced or imbalanced?. **(5 points)**
2. Along with the target variable, define your feature space (X) and split the data into test (30%) and train set (70%). **(5 points)**
3. From sklearn.ensemble import the relevant function for RF classification. Do the followings: **(25 points)**
    1. Train the RF classification model using its default parameters. However use random_state=1000. (5 points)
    2. Generate the predicted probabilities and predicted classifications and save them as y_hat_probs, y_hat respectively. (5 points)
    3. Plot the histogram of y_hat_probs? Explain what you see? Is there a probability threshold at which the model always predict negative or positive? (5 points)
    4. Use the built-in classification report function from sklearn. Report the Accuracy, precision, recall and f1 score along with the confusion matrix. Interpret all of these statistics. Do you trust the accuracy of the model? why? (10 points)
4. **Tuning hyperparameters**: I specifically want you to use the following:
   param_grid. my_param_grid = n_estimators:[100,200], criterion:['gini','entropy'], max_features:['log2', 'auto'], max_depth:[10,None].
   Where you able to improve the model performance? Is random forest using a small tree or bushy tree? How do yo know? **(10 points)**
5. **Cross validation and OOB observations**: estimate the accuracy of the test set using both CV and OOB methods and compare the results. **(15 points)**
6. **Dealing with imbalanced data**: Use the balanced version of RF classifier and save the predictions as y_hat_balanced. Report the precision, recall and f1 score for the balanced RF versus what you found in part 3.4. What happened to recall? does that make sense? **(10 points)**
7. Plot the ROC curve for the balanced RF and report the AUC. Can you trust this number now? why? **(10 points)**
8. **Feature importance**: Plot the feature importance graph and report the top 5 important features. Is your finding consistent with common sense? **(10 points)**