Computer Science 3202/6915
Assignment 4 – Linear models for regression – Do your best!

Goal
Apply linear methods for regression and select the best model.

Due date
Sunday March 14th by 11:30pm.

Specifications
Your program should be called A4_Reg.py and it should run in Linux. You program should take two command-line arguments: A filename specifying a tab-delimited plain-text file containing the training data, and a filename specifying a tab-delimited plain-text file containing the test data.

For example, we should be able to execute your program as follows:

```
$python3 A4_Reg.py Traindata.txt Testdata.txt
```

where the $ indicates the terminal prompt.

The data you should use is provided in Brigthspace. For generating your model, you will train with the data given in the file A3_TrainData_noDup.tsv. This is a tab-delimited text file containing 2698 observations. The file has a header but not instance IDs. There are 11 numerical attributes, and the last column is the value to predict.

Once you have your final model, you will predict the values for the observations given in A4_TestData_noDup.tsv. This is a tab-delimited text file containing 196 instances.

This data was taken from the article: P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences by data mining from physicochemical properties. In *Decision Support Systems*, Elsevier, 47(4):547-553, 2009. If you are interested to take a look, this manuscript is available in Brightspace.

Functionality
Your program should do the following:

1. Read the input data. You can assume the input files are in the working directory. **DONE IN ASSIGNMENT 3**
2. Use k-fold CV to assess the performance of a linear model using least squares linear regression with the scikit-learn function LinearRegression (https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html). This is your baseline model. **DONE IN ASSIGNMENT 3**
3. Perform grid-search CV to find the optimal hyper-parameter setting for at least one of these alternatives:
    1. Ridge regression (https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html#sklearn.linear_model.Ridge). Optimize at least alpha.
    2. Lasso

(https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html#sklearn.linear_model.Lasso). Optimize at least alpha.

3. ElasticNet (https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.ElasticNet.html#sklearn.linear_model.ElasticNet). Optimize at least alpha.
4. Least squares linear regression with different attributes. In this case you will select the optimal feature set.

4. Select your best model and create your final model using all of the training data.
5. Predict the quality values for the instances in the test data.
6. Write in a file called A4_predictions_groupXX.txt (where XX is your group number) the predicted values for the instances in the test data. This file must be a text file with a single column containing your predictions for the test instances in exactly the same order as given in the file A4_TestData_noDup.tsv

Submission

Submit through Brightspace the following (one submission per team):

a) Your python code in a single file called A4_Reg.py
b) A PDF file containing:
   1. a brief description of the assessed models (data transformation, regression method, grid considered, etc),
   2. the best hyper-parameter setting found,
   3. the cross-validation performance (R^2 and RSS) of your baseline model and your best model,
   4. an acknowledgement section listing your collaborations and online sources, and
   5. a program specification section listing the Python version and libraries you used.
c) A text file called A4_predictions_groupXX.txt (where XX is your group number) with your predicted values for the instances in the test data. This file must be a text file with a single column containing your predictions for the test instances in exactly the same order as given in the file A4_TestData_noDup.tsv

Performance assessment:

25% of the evaluation for this assignment will be graded based on predictive performance. The team(s) with the best performing model (referred below as 1st ranked model) in terms of RSS (rounded to 2 decimal places) on the test dataset will receive full marks for the performance of their model (2.5). All other teams will receive a mark for the performance of their model proportional to their decrease in performance with respect to the 1st ranked model. For example, if the RSS of the best model of a given team is 10% higher than the RSS of the 1st ranked model then their mark for performance of their model will be 2.25.

Common pitfalls to avoid (i.e., DO NOT do the following or points will be deducted):

1. Submit files in a compressed file.
2. Fail to include some of the sections in the PDF file.
3. Fail to have the predicted values **exactly in the same order** as given in the file A4_TestData_noDup.tsv.

4. Forget to acknowledge a collaboration or source.
5. Miss some functionality.
6. Use different data sets.

Online examples/tutorials:

- https://www.kaggle.com/jnikhilsai/cross-validation-with-linear-regression
- https://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html
- https://machinelearningmastery.com/hyperparameter-optimization-with-random-search-and-grid-search/