Spring 2021 BBL536E Homework 1

Remarks:

Write the code yourself. ***Cheating is strictly forbidden***.

For each problem write your code in the function format and give the names of the functions as problem numbers, for example for the solution of problem1:

```
def problem1(input):
  return something
```

Put the codes for all problems into one file (jupter notebook file) and name that file using your student username in the following format: badays_bbl536e_homework1.ipynb. The notebook file should definitely contain the outputs of the functions, if applicable. Sample solution file (sample_solution.ipynb) is given to you to show how to organize your solutions.

Give as much as documentation for your script using comments.

Create a report ("homework1_report.pdf") for the results you are asked in the problems.

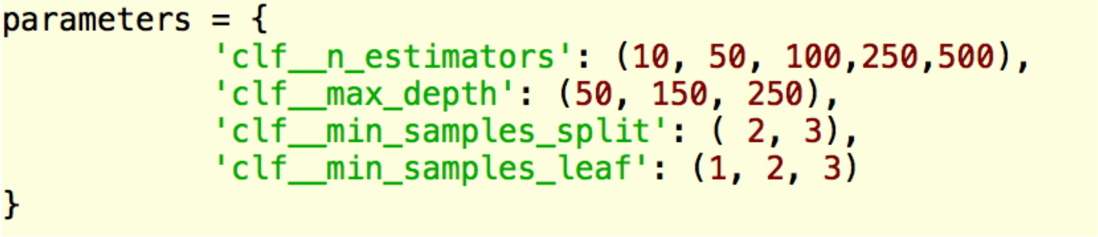
Problem.1 (40 Points)

In this problem, you are going to build predictive models on the estimation of energy performance of residential buildings. You are given **ENB2012_data.xlsx** file. You can also get the dataset from the following link: http://archive.ics.uci.edu/ml/datasets/energy+efficiency. The dataset comprises 768 samples and 8 features. The features are relative compactness, surface area, wall area, roof area, overall height, orientation, glazing area, glazing area distribution. The two output variables are heating load (HL) and cooling load (CL), of residential buildings. Machine learing model can be used to predict heading and cooling loads for the aferomentioned features of a building. There is a scientific article which provides some analysis on this data set: A. Tsanas, A. Xifara: 'Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools', Energy and Buildings, Vol. 49, pp. 560-567, 2012.

For this problem, do the following tasks:

Task1: First build a predictive model using Ridge Regression. Tests the ridge model with the following alpha parameters and find the optimal one: 0.001,0.01,0.1, 1.0, 10.0. Then using the optimal alpha parameter calculate the Mean Absolute Error and Mean Squared Errors using 10-fold 10 repetition with randomly chosen data cross validation stragety. Calculate the mean score and the standart deviation for these cross validations.

Task2: In this task you are going to build a predictive model using RandomForestRegressor. First using gridsearch find the optimal values for the parameters given below:



```
parameters = {
            'clf__n_estimators': (10, 50, 100,250,500),
            'clf__max_depth': (50, 150, 250),
            'clf__min_samples_split': ( 2, 3),
            'clf__min_samples_leaf': (1, 2, 3)
}
```

Report the optimal values you found.

After finding the optimal parameters, using these parameters calculate the Mean Absolute Error and Mean Squared Errors scores for 10-fold 10 repetition with randomly chosen data cross validation stragety. At final step calculate the mean and standart deviation of the scores for cross validations. Do these steps separately for Y1 and Y2 outputs.
In your report, provide a table like given below for the scores. The values in the tables are mean and standart deviations.

	Mean Absolute Error		Mean Squared Error	
Output	RandomForest	RidgeRegression	RandomForest	RidgeRegression
Y1	0.31±0.03	2.091±0.23	0.21±0.05	8.71±1.74
Y2	1.08± 0.17	2.26±0.26	2.61±0.66	10.35±2.53

Problem.2 (60 Points)

In this problem, you are going to build predictive models for bank telemarketing problem. The data is related with direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, in order to access if the product (bank term deposit) would be ('yes') or not ('no') subscribed. You are given **bank-additional-full.csv** file containing the data set.

The dataset can also be obtained from https://archive.ics.uci.edu/ml/datasets/bank+marketing .The following article describes some analysis on this dataset: S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014.

Here is the information on the data attributes:

Input variables:

# bank client data:

1 - age (numeric)

2 - job : type of job (categorical: 'admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown')

3 - marital : marital status (categorical: 'divorced','married','single','unknown'; note: 'divorced' means divorced or widowed)

4 - education (categorical: 'basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown')

5 - default: has credit in default? (categorical: 'no','yes','unknown')

6 - housing: has housing loan? (categorical: 'no','yes','unknown')

7 - loan: has personal loan? (categorical: 'no','yes','unknown')

# related with the last contact of the current campaign:

8 - contact: contact communication type (categorical: 'cellular','telephone')

9 - month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')

10 - day_of_week: last contact day of the week (categorical: 'mon','tue','wed','thu','fri')

11 - duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no'). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.

# other attributes:

12 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)

13 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)

14 - previous: number of contacts performed before this campaign and for this client (numeric)

15 - poutcome: outcome of the previous marketing campaign (categorical: 'failure','nonexistent','success')

# social and economic context attributes

16 - emp.var.rate: employment variation rate - quarterly indicator (numeric)

17 - cons.price.idx: consumer price index - monthly indicator (numeric)

18 - cons.conf.idx: consumer confidence index - monthly indicator (numeric)

19 - euribor3m: euribor 3 month rate - daily indicator (numeric)

20 - nr.employed: number of employees - quarterly indicator (numeric)

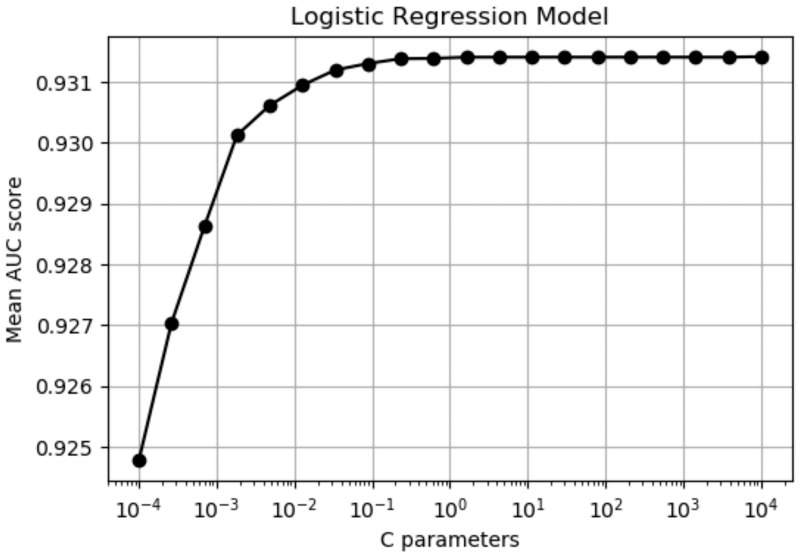
Output variable (desired target):

21 - y - has the client subscribed a term deposit? (binary: 'yes','no')

You are going to build predictive models for the prediction of the ouput whether a given client will subscrive a term deposit or not. You will use data in "*bank-additional-full.csv"* file. Some attributes have "unknown" or "nonexistent" categories. Don't bother to clean this data. You can consider them as a category in that attribute. Note that you may not get the exact results with results given in the assignment. Slightly different results are fine.

You are asked to perform the following tasks:

Task1 (15 points): Build a logistic regression model.
Follow the instructions given when building the model.
For scoring use AUC metric. For cross-validation use 5-fold cross-validation with 5 repetitions. In cross-validation select data randomly. Find the C parameter which gives the highest AUC score. Note that C is a hyperparameter in Logistic Regression which means Inverse of regularization strength. Try model with 20 different C parameters. C parameters should range from 10^{-4} to 10^{4} . Plot mean AUC score vs C parameter. Your figure should look like figure given below. Note that the calculation takes some time since you do model training and test for 500 times.



Task2 (15 points): Build a Random Forest model.
Follow the instructions given when building the model.

Using gridsearch try to find the best score and combination of the following hyperparameters:
Number of estimators: 10,50,100,250,500,1000
max_depth: 50,150,250
min_samples_split: 2,3
min_samples_leaf: 1,2,3

For cross-validation in grid search use a cross validation strategy as 3-fold cross-validation with 3 repetitions.
Report the hyperparameter set yielding the best score. For scoring use AUC score.

Task3 (15 points): Build a neural network model:
Follow the instructions given when building the model.
First, scale your input data so that it has zero mean and one standart deviation. This is important because neural network models are sensitive to input scaling.

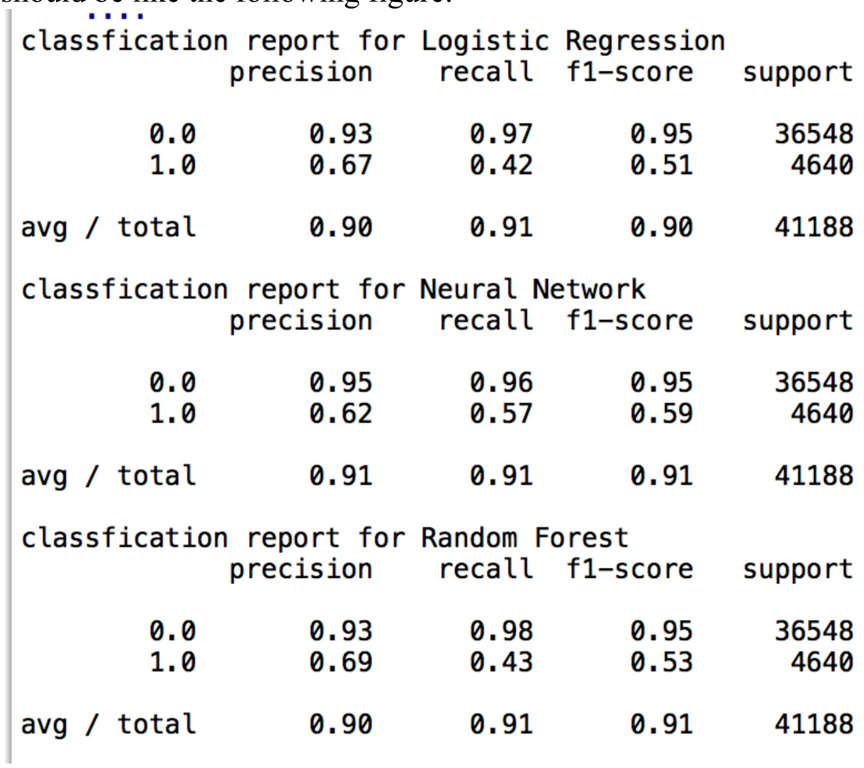
Then using gridsearch try to find the best score and combination of the following hyperparameters:
hidden_layer_sizes: (10,10,10), (10,10,10,10), (10,10,10,10,10), (10,10,10,10,10,10)
alpha: 0.00001, 0.0001, 0.001, 0.01, 0.1
In grid search fit the use AUC score as scoring. For cross-validation in grid search use a cross validation strategy as 3-fold cross-validation with 3 repetitions.
Report the hyperparameter set yielding the best score.

Task4 (15 points): Prepare a classification report for three models.

In this task you are going to perform 5-fold cross-validation with randomly splitting data. For each model print the average classification report for this 5-fold cross validation.

For logistic regression use C=1 and, for Neural Network and Random Forest models use the optimal hyper parameters found in task2 and task3.

The output should be like the following figure:



```
classfication report for Logistic Regression
             precision    recall  f1-score   support

        0.0       0.93      0.97      0.95     36548
        1.0       0.67      0.42      0.51      4640

avg / total       0.90      0.91      0.90     41188

classfication report for Neural Network
             precision    recall  f1-score   support

        0.0       0.95      0.96      0.95     36548
        1.0       0.62      0.57      0.59      4640

avg / total       0.91      0.91      0.91     41188

classfication report for Random Forest
             precision    recall  f1-score   support

        0.0       0.93      0.98      0.95     36548
        1.0       0.69      0.43      0.53      4640

avg / total       0.90      0.91      0.91     41188
```