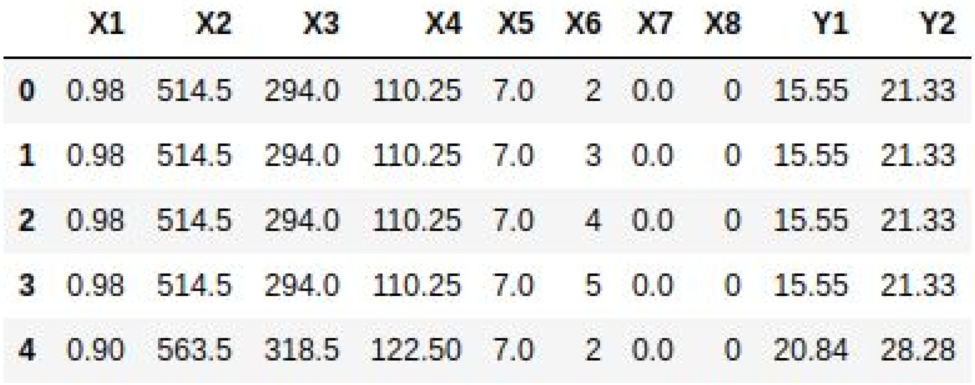
Datascience Home Wo Report

By Beltus Nkwawir _704181021

Problem 1: Energy Estimate

1. Dataset Exploration.

The dataset was read as pandas dataframe using the *pandas.read_excel()* method and the first five columns can be seen in the table below:

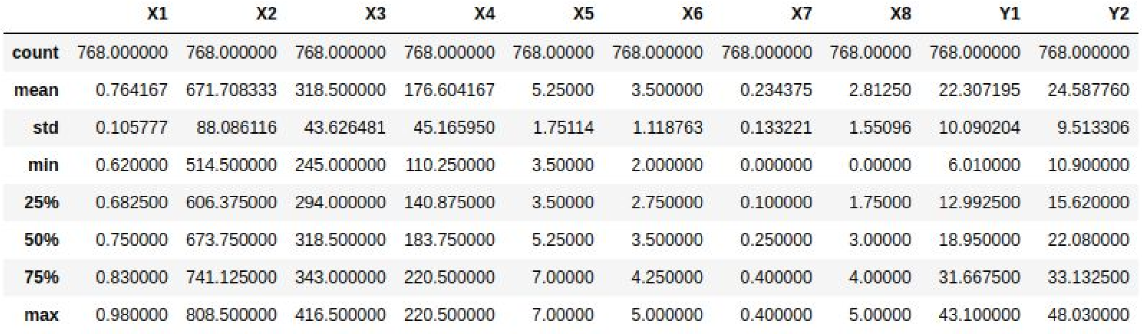


	X1	X2	X3	X4	X5	X6	X7	X8	Y1	Y2
0	0.98	514.5	294.0	110.25	7.0	2	0.0	0	15.55	21.33
1	0.98	514.5	294.0	110.25	7.0	3	0.0	0	15.55	21.33
2	0.98	514.5	294.0	110.25	7.0	4	0.0	0	15.55	21.33
3	0.98	514.5	294.0	110.25	7.0	5	0.0	0	15.55	21.33
4	0.90	563.5	318.5	122.50	7.0	2	0.0	0	20.84	28.28

Figure: First five rows of the dataset

2. Statistical Analysis

Using the *pandas.describe()* method, some intuitive insights could be made about our dataset as seen in the figure below.



	X1	X2	X3	X4	X5	X6	X7	X8	Y1	Y2
count	768.000000	768.000000	768.000000	768.000000	768.00000	768.000000	768.000000	768.00000	768.000000	768.000000
mean	0.764167	671.708333	318.500000	176.604167	5.25000	3.500000	0.234375	2.81250	22.307195	24.587760
std	0.105777	88.086116	43.626481	45.165950	1.75114	1.118763	0.133221	1.55096	10.090204	9.513306
min	0.620000	514.500000	245.000000	110.250000	3.50000	2.000000	0.000000	0.00000	6.010000	10.900000
25%	0.682500	606.375000	294.000000	140.875000	3.50000	2.750000	0.100000	1.75000	12.992500	15.620000
50%	0.750000	673.750000	318.500000	183.750000	5.25000	3.500000	0.250000	3.00000	18.950000	22.080000
75%	0.830000	741.125000	343.000000	220.500000	7.00000	4.250000	0.400000	4.00000	31.667500	33.132500
max	0.980000	808.500000	416.500000	220.500000	7.00000	5.000000	0.400000	5.00000	43.100000	48.030000

Figure: Statistical analysis of the dataset

Task 1: Building and Testing a Ridge Predictive Model.

A ridge regression model was built and by using gridsearch, the different parameters of alpha within the :[0.001,0.01,0.1, 1.0, 10.0] were tested.

For both Y1 and Y1 output labels, the optimal value of alpha obtained was
alpha = 0.1

Using the optimal computed alpha parameter, the Mean Absolute Error and Mean Squared Errors using **10-fold 10 repetitions** with randomly chosen data cross-validation strategy were then evaluated for each output label and the results are shown in the table below.

	Mean of mean Absolute Error Score(MAE)	**Standard Deviation of MAE**	**Mean Square Error Score(MSE)**	**Standard Deviation of MSE**
Y1 - Label	0.911248	0.019398	0.911085	0.018569
Y2- Label	0.880110	0.024485	0.880428	0.030050

Table: Mean and Standard deviation of MAE and MSE for Ridge Regression using optimal C

Task 2: Building a Random Forest Regressor Model

A Random Forest Regressor model was built and using gridsearch on a combination of parameters, the optimal values for the parameters were found to be:

Best Hyperparameter Set For GridSearch RandomForest		
	Y1	**Y2**
Number of estimators	500	250
Max_depth	250	250
min_samples_split	2	2
min_samples_leaf:	1	1

Table: Best hyperparameters using Grid search with Random Forest Classifier

After computing the mean and standard deviations for both Y1 and Y2 using Ridge regression and RandomForest Regressor models and scoring with Mean squared Error and Mean Absolute error, the eight(8) results obtained are presented in the table below

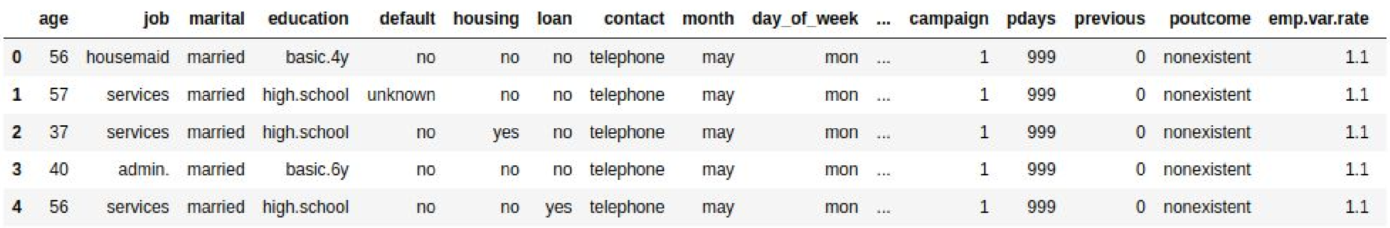
	Mean Absolute Error		**Mean Squared Error**	
Output	RandomForest	Ridge Regression	RandomForest	Ridge Regression
Y1	0.34 ± 0.06	2.12 ± 0.27	0.28 ± 0.11	8.87 ± 1.94
Y2	1.06 ± 0.17	2.30 ± 0.31	3.01 ± 0.92	10.67 ± 3.02

Table: MAE and MSE for Random Forest and Ridge Regression

Problem 2: Bank Telemarketting

1. Dataset Exploration.

The dataset was read as pandas dataframe using the *pandas.read_csv()* method and the first five columns can be seen in the table below:

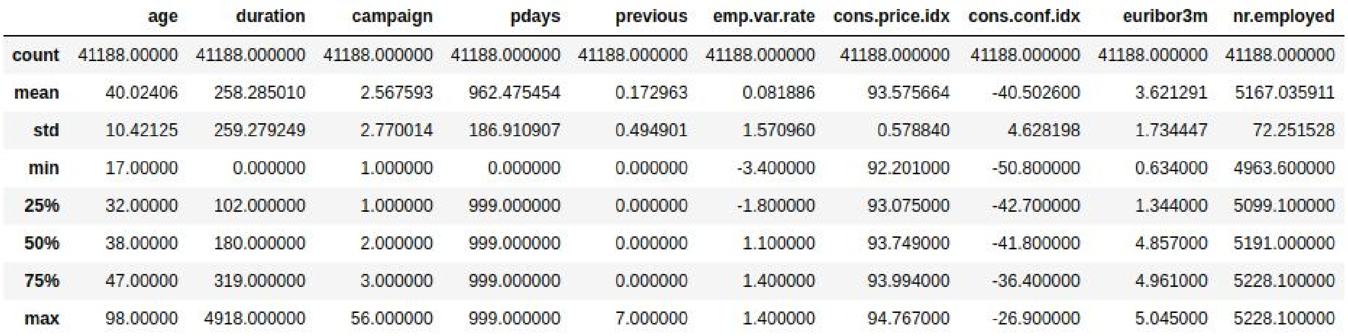


	age	job	marital	education	default	housing	loan	contact	month	day_of_week	...	campaign	pdays	previous	poutcome	emp.var.rate
0	56	housemaid	married	basic.4y	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1
1	57	services	married	high.school	unknown	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1
2	37	services	married	high.school	no	yes	no	telephone	may	mon	...	1	999	0	nonexistent	1.1
3	40	admin.	married	basic.6y	no	no	no	telephone	may	mon	...	1	999	0	nonexistent	1.1
4	56	services	married	high.school	no	no	yes	telephone	may	mon	...	1	999	0	nonexistent	1.1

Figure: First five rows of the dataset

2. Statistical Analysis of the dataset

Using the *pandas.describe()* method, some intuitive insights could be made about our dataset as seen in the figure below.

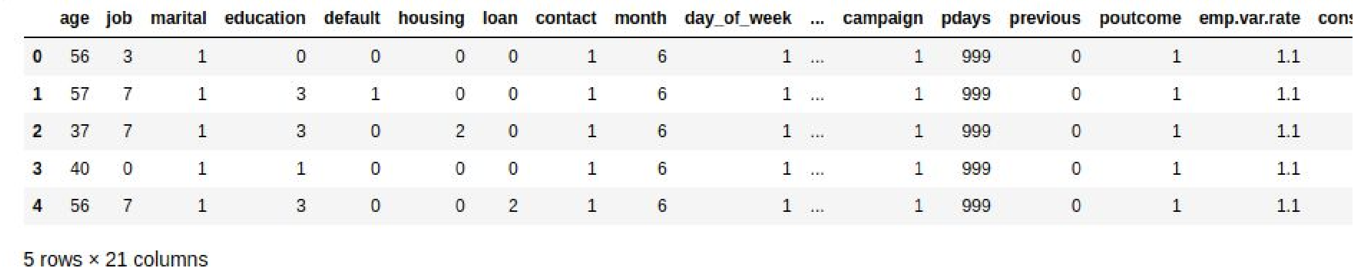


	age	duration	campaign	pdays	previous	emp.var.rate	cons.price.idx	cons.conf.idx	euribor3m	nr.employed
count	41188.00000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000	41188.000000
mean	40.02406	258.285010	2.567593	962.475454	0.172963	0.081886	93.575664	-40.502600	3.621291	5167.035911
std	10.42125	259.279249	2.770014	186.910907	0.494901	1.570960	0.578840	4.628198	1.734447	72.251528
min	17.00000	0.000000	1.000000	0.000000	0.000000	-3.400000	92.201000	-50.800000	0.634000	4963.600000
25%	32.00000	102.000000	1.000000	999.000000	0.000000	-1.800000	93.075000	-42.700000	1.344000	5099.100000
50%	38.00000	180.000000	2.000000	999.000000	0.000000	1.100000	93.749000	-41.800000	4.857000	5191.000000
75%	47.00000	319.000000	3.000000	999.000000	0.000000	1.400000	93.994000	-36.400000	4.961000	5228.100000
max	98.00000	4918.000000	56.000000	999.000000	7.000000	1.400000	94.767000	-26.900000	5.045000	5228.100000

Figure: Statistical analysis of the dataset

3. Conversion of Categorical to Numeric Columns

The dataset had several categorical columns. Unfortunately, machine learning models cannot be trained with such data. Therefore, the columns were transformed to numeric before training the model as can be seen in the image below



	age	job	marital	education	default	housing	loan	contact	month	day_of_week	...	campaign	pdays	previous	poutcome	emp.var.rate	cons
0	56	3	1	0	0	0	0	1	6	1	...	1	999	0	1	1.1	
1	57	7	1	3	1	0	0	1	6	1	...	1	999	0	1	1.1	
2	37	7	1	3	0	2	0	1	6	1	...	1	999	0	1	1.1	
3	40	0	1	1	0	0	0	1	6	1	...	1	999	0	1	1.1	
4	56	7	1	3	0	0	2	1	6	1	...	1	999	0	1	1.1	

5 rows × 21 columns

Figure: categorical to numeric columns

Task 1: Build a Logistic Regression Model

After building a Logistic Regression model with cross-validation use 5-fold cross-validation with 5 repetitions and varying C-hyperparameter with the range of 10^-4 and 10^4, a plot mean AUC score vs C parameter was evaluated and presented below

C-values used for training the model = [0.0001,0.0005,0.001,0.005, 0.01, 0.05 ,0.1, 0.5, 1.0, 5, 10.0, 50, 100.0, 250, 500, 1000.0, 2500, 5000, 7500, 10000.0]

The C hyperparameter with the Highest AUC score is C = 1000.0

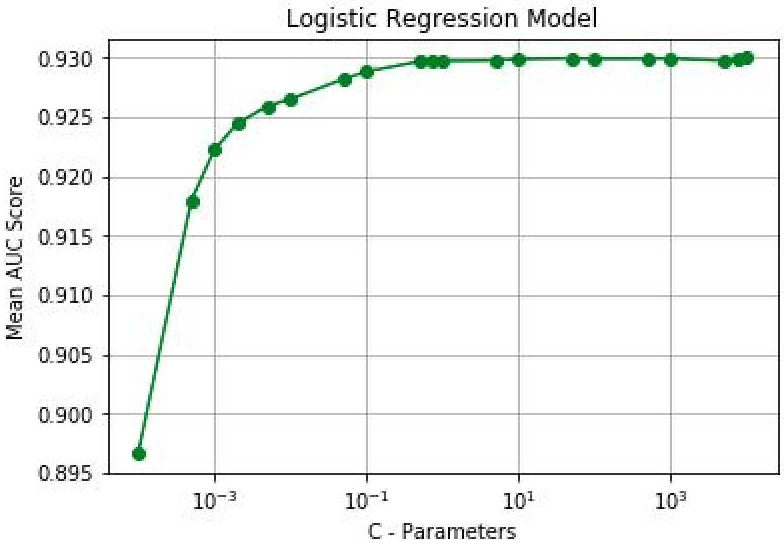


Figure: Graph of Mean AUC Score with Difference values of C-parameters with Logistic Regression Model

Task 2: Build a Random Forest Model

In order to obtain the best combination of parameters that yields the best AUC score, for a random forest classifier, I implemented a Gridsearch with these different hyperparameter values.

The cross-validation in grid search used a cross-validation strategy of **3-fold cross-validation with 3 repetitions.**

Hyperparameters For GridSearch	
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

Best Hyperparameter Set with Best Score

After running the Gridsearch with the above hyperparameter values, the hyperparameter set with the best AUC score was:

Best Hyperparameter Set For GridSearch	
Number of estimators	1000
Max_depth	150
min_samples_split	3
min_samples_leaf:	3
Best score	**0.9467143264126682**

With the best score of

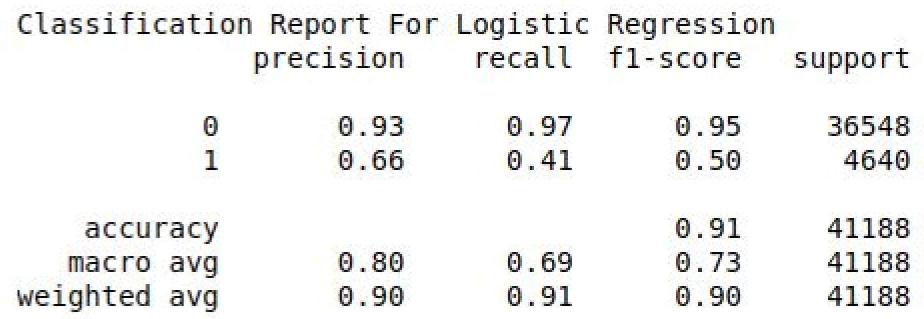
Task 3: Build a Neural Network Model

As specified in the homework document, gridsearch was implemeted 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

Best Hyperparameter Set	
hidden_layer_sizes	(10,10,10)
alpha	0.1
Best score	0.94

Task 4: Classification Reports

1) Classification Report for Logistic Regression with C = 1

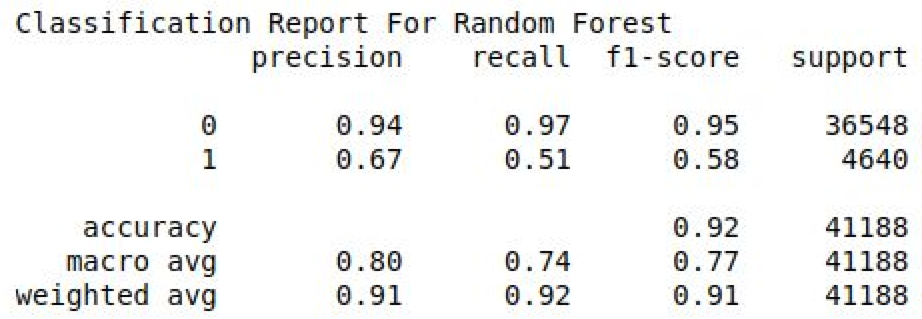


```
Classification Report For Logistic Regression
              precision    recall  f1-score   support

           0       0.93      0.97      0.95     36548
           1       0.66      0.41      0.50      4640

    accuracy                           0.91     41188
   macro avg       0.80      0.69      0.73     41188
weighted avg       0.90      0.91      0.90     41188
```

2) Classification Report for Random Forest Model with optimal Hyperparameters

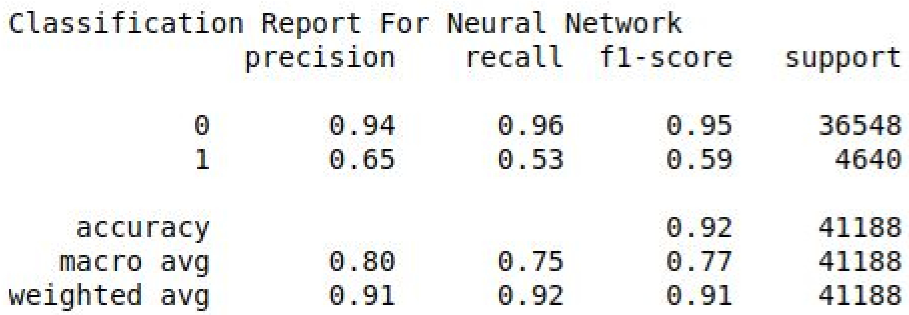


```
Classification Report For Random Forest
              precision    recall  f1-score   support

           0       0.94      0.97      0.95     36548
           1       0.67      0.51      0.58      4640

    accuracy                           0.92     41188
   macro avg       0.80      0.74      0.77     41188
weighted avg       0.91      0.92      0.91     41188
```

3) Classification Report for Neural Network with optimal hyperparameters



```
Classification Report For Neural Network
              precision    recall  f1-score   support

           0       0.94      0.96      0.95     36548
           1       0.65      0.53      0.59      4640

    accuracy                           0.92     41188
   macro avg       0.80      0.75      0.77     41188
weighted avg       0.91      0.92      0.91     41188
```