PROJECT REPORT

IE7374: Machine Learning in Engineering Fall 2021

Group 2:
Anish Kumar Musunuru, Karan Paresh, Rashmi Nambiar Pappinisseri Puthenveettil

TOPIC: Predicting term deposit subscription by classification

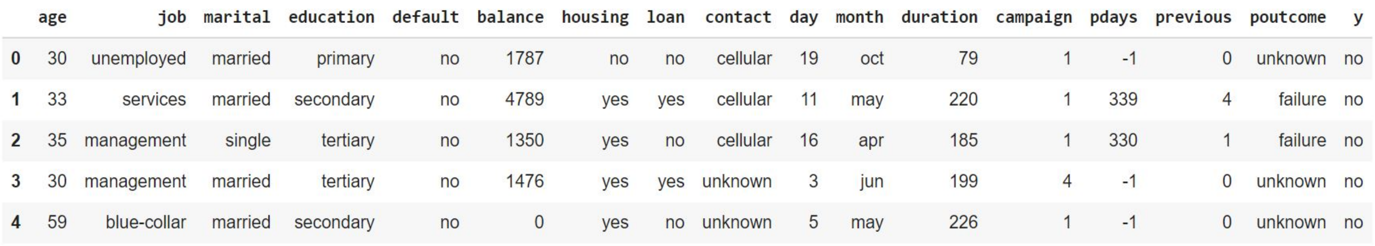
ABSTRACT: -

Banks offer term deposits to its customers on a fixed interest rate over a fixed period of time. This amount is usually used by the banks to loan the amount to other customers. The loan is given with a higher interest rate so that banks can make profit which is called net interest margin. If the term deposit is withdrawn before the fixed period, a penalty fee is to be paid to the bank.

The dataset used in this project deals with direct marketing campaigns for term deposit of a Portuguese banking institution. The campaigns are done on phone calls. The dataset consists of 4521 records with 16 independent features like age, job, marital status, education, default credit, housing loan, personal loan, contact communication type, last contact day, last contact duration, etc.

The goal is to predict whether the product (the bank term deposit) would be subscribed to (y=Yes) or not (y=No). This is a classification problem. The dataset would initially be checked for missing values and data clean-up would be performed. Further, some of the feature values would have to be converted to dummy variables as there are multiple discrete values. For example, 'education' can be primary, secondary or tertiary. Further, for prediction, supervised machine learning methods for classification, like logistic regression, Naive Bayes and Support vector machine (SVM) can be used.

DATA DESCRIPTION:



	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	y
0	30	unemployed	married	primary	no	1787	no	no	cellular	19	oct	79	1	-1	0	unknown	no
1	33	services	married	secondary	no	4789	yes	yes	cellular	11	may	220	1	339	4	failure	no
2	35	management	single	tertiary	no	1350	yes	no	cellular	16	apr	185	1	330	1	failure	no
3	30	management	married	tertiary	no	1476	yes	yes	unknown	3	jun	199	4	-1	0	unknown	no
4	59	blue-collar	married	secondary	no	0	yes	no	unknown	5	may	226	1	-1	0	unknown	no

Input variables:

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')

4 - education (categorical: 'primary','secondary','tertiary','unknown')

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

6-Balance: (numeric)

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

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

# related with the last contact of the current campaign:

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

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

11 - day_of_month: last contact day of the month (numeric)

12 - 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, th duration is not known before a call is performed. Also, after the end of the call y is obviously know. Thus, this input should only be included for benchmark purposes and should be discarded if the inten tion is to have a realistic predictive model.

13 - campaign: number of contacts performed during this campaign and for this client (numeric, incl udes last contact)

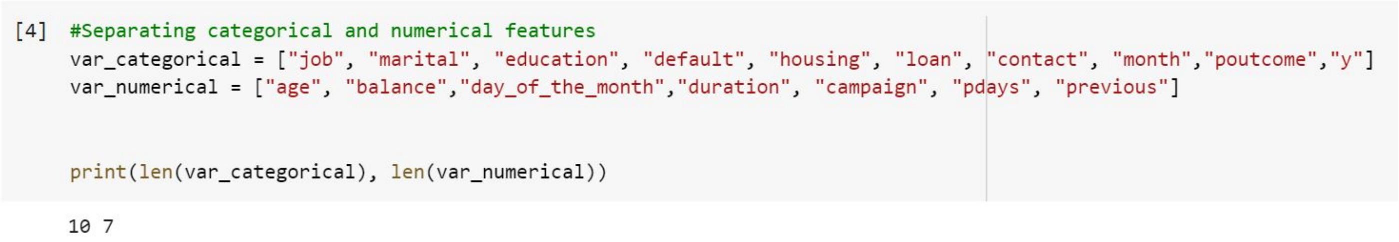
14 - pdays: number of days that passed by after the client was last contacted from a previous campaig n (numeric; -1 means client was not previously contacted)

15 - previous: number of contacts performed before this campaign and for this client (numeric, if the value of pdays=-1, then previous=0)

16 - poutcome: outcome of the previous marketing campaign (categorical: 'failure','unknown','succes s','other', we are converting other to unknown)

Output variable (desired target):
17 - y - has the client subscribed a term deposit? (binary: 'yes','no')

Categorical and numerical variables split: -



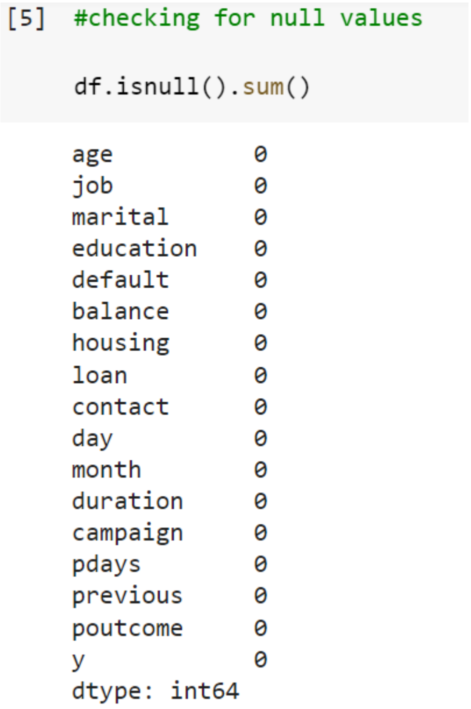
```
[4]  #Separating categorical and numerical features
     var_categorical = ["job", "marital", "education", "default", "housing", "loan", "contact", "month","poutcome","y"]
     var_numerical = ["age", "balance","day_of_the_month","duration", "campaign", "pdays", "previous"]


     print(len(var_categorical), len(var_numerical))

     10 7
```

There are 10 categorical variables and seven numeric variables in the dataset.

Null values check



```
[5]  #checking for null values

     df.isnull().sum()

     age          0
     job          0
     marital      0
     education    0
     default      0
     balance      0
     housing      0
     loan         0
     contact      0
     day          0
     month        0
     duration     0
     campaign     0
     pdays        0
     previous     0
     poutcome     0
     y            0
     dtype: int64
```

There are no values in the dataset, so data imputation is not required.

METHODS:

This is a binary classification problem, which has an output target variable with value 1 and 0 which indicates whether the customer will subscribe to the term deposit or not. The machine learning classification models used are:

(i) Logistic Regression
(ii) SVM
(iii) Feed-Forward Neural Network

Logistic Regression:

The logistic regression we used is our baseline model. The model is built for three different learning rates .

1. Learning rate = 0.0001 the attained accuracy 79.14
2. Learning rate = 0.01 the attained accuracy 80.10
1. Learning rate = 1 the attained accuracy 75.53

SVM:

Support vector machine with Radial Basis Function as a kernel and the optimizer method used is "SLSQP" which helps us to classify non-linear data. The model fails to achieve the optimum results as the accuracy and F1-score are 54.54 and 0.47 respectively. This model performs the worst on the data points.

Neural Network:

The Feed forward neural network is used in this case to predict the classification output value.

The parameters used here are,

(a) epoch =100
(b) batch_size = 10
(c) verbose = 0

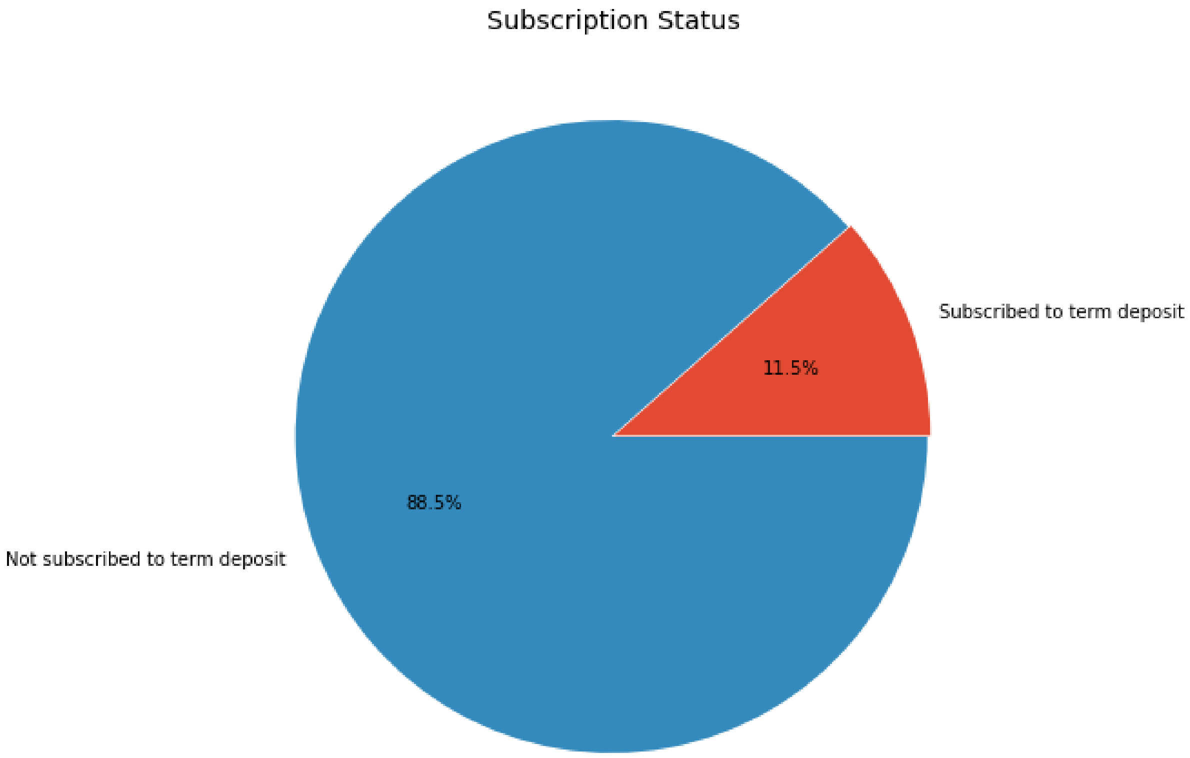
1. One hidden layer Neural network: -

The epochs is 100 and the and the batch size is 10 the test accuracy attained is 85.6%.This is the best model.

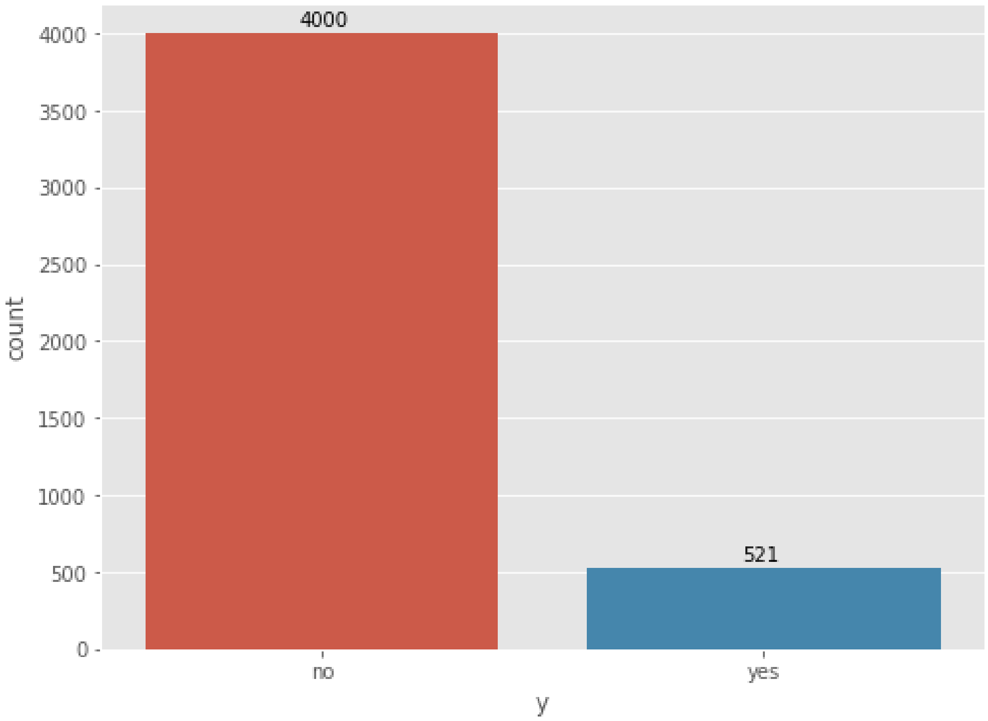
2. Two hidden layer Neural network:

The test accuracy attained is 85.3 %, which is lesser than that of the one hidden layer model and hence it shows that the one hidden layer model is sufficient for this data.

Exploratory Data Analysis

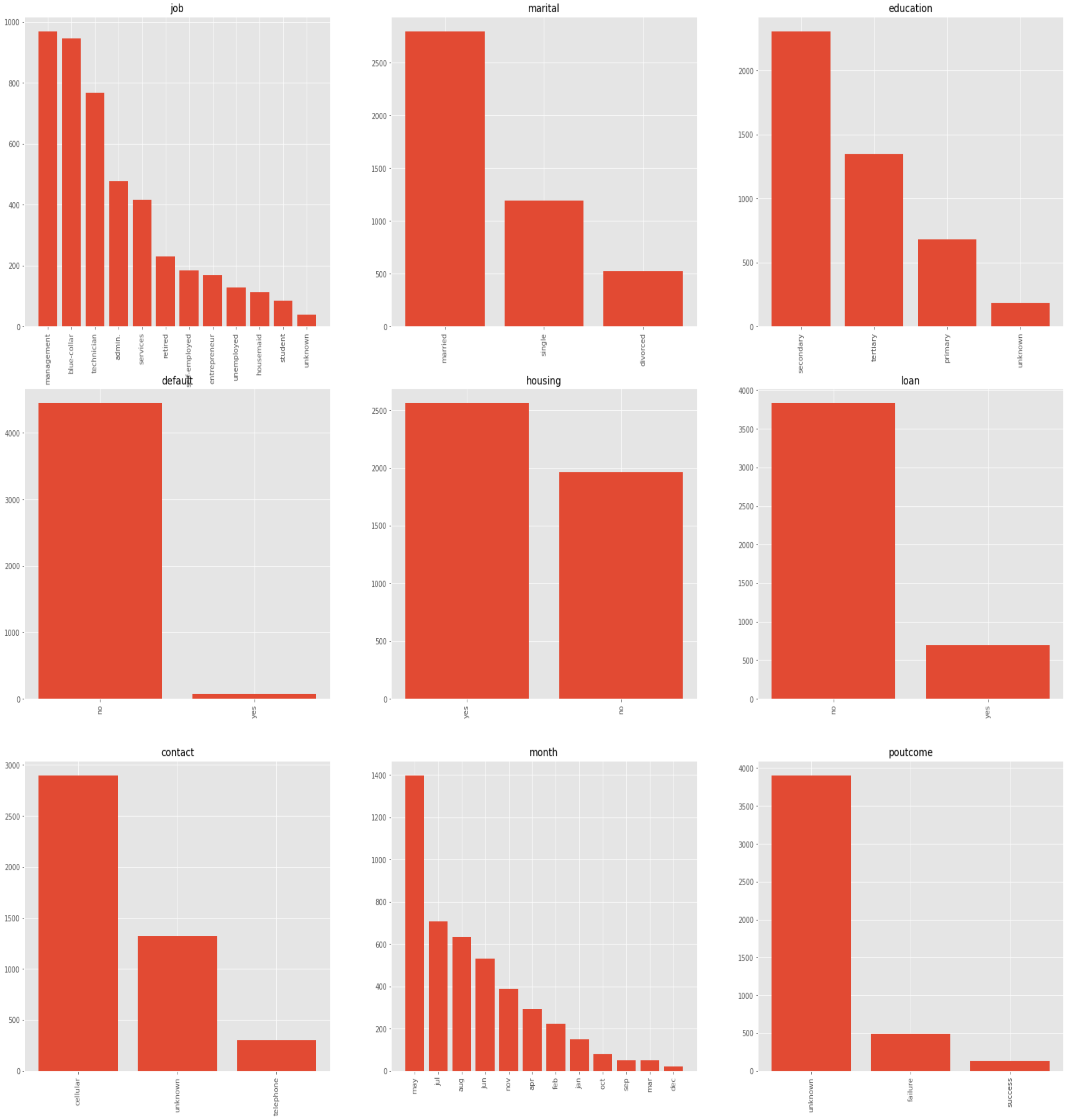


There are totally 4521 number of customers out of which 521 customers has subscribed to the term deposit which is 11.5% of the total number of customers.



The number of customers who did not subscribe to the term deposit are 4000 which is 88.5% of the total number of customers.

Plotting the count of unique values of all the categorical variables of the dataset,
Categorical variable - job", "marital", "education", "default", "housing", "loan", "contact", "month", "poutcome","y”



The above plots shows the distribution of each category of each categorical variable and the results shows that,

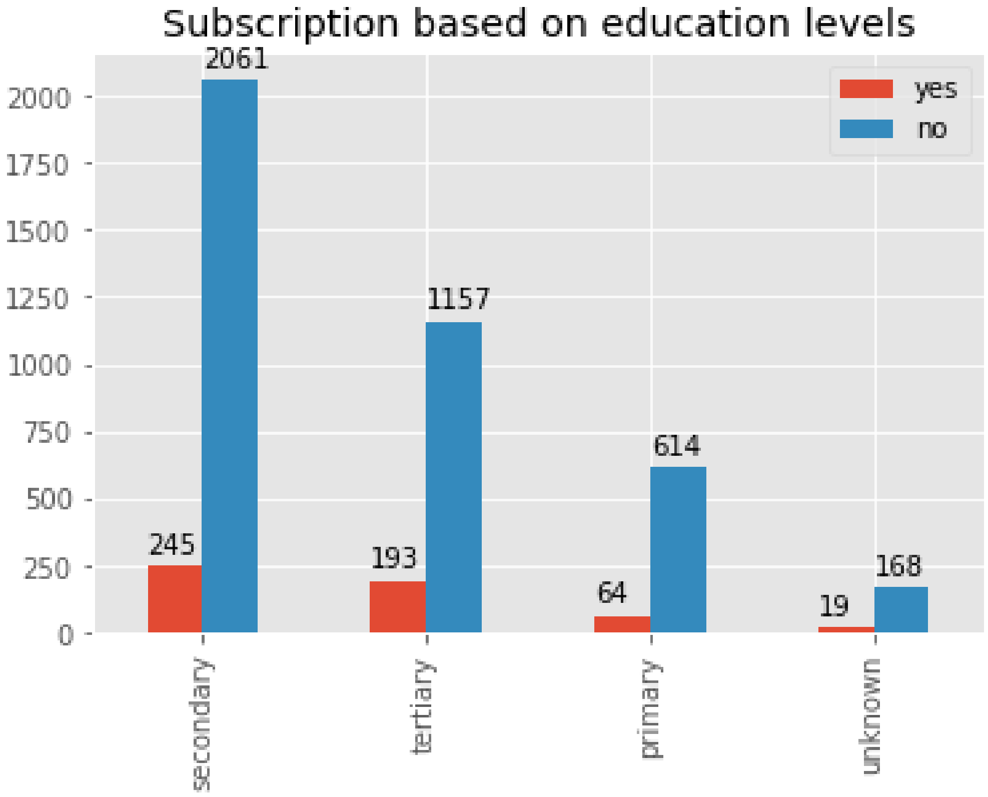
1. The number of clients subscribing to the term deposit is shown in the above graph and it shows that the highest subscribers may be enrolled in the month of may as it has the highest count amongst all other months.

2.The highest count in the education category is of the secondary education that means maximum people who are being contacted holds a secondary education certificate.

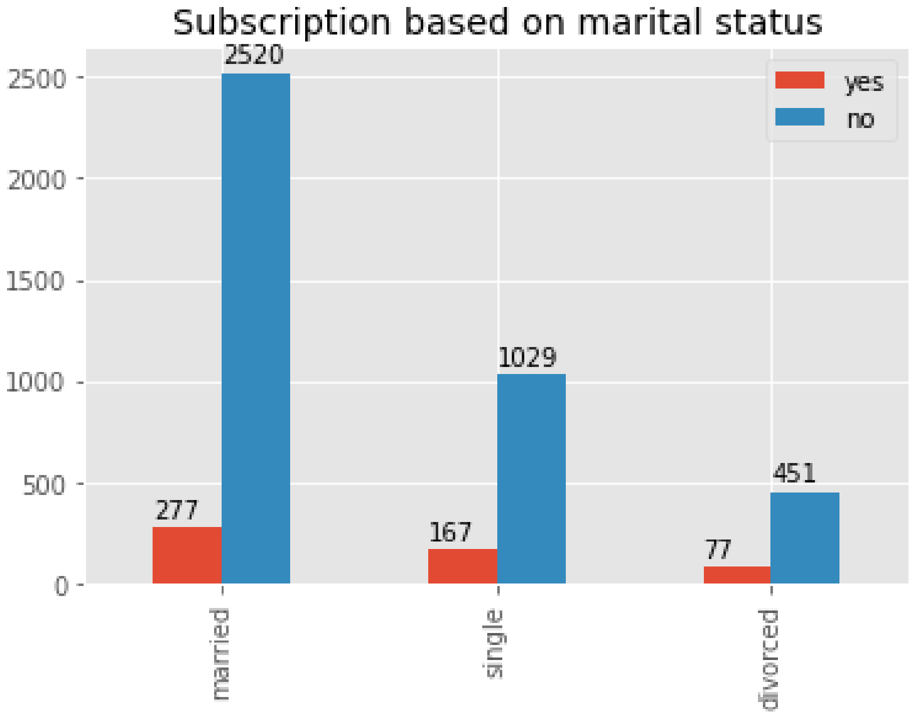
3. The method of communication through a cellular device seems to be the popular one for the bank.

4.The number of customers who has taken the housing loan is much greater than that of those who has take loan for the other purpose.

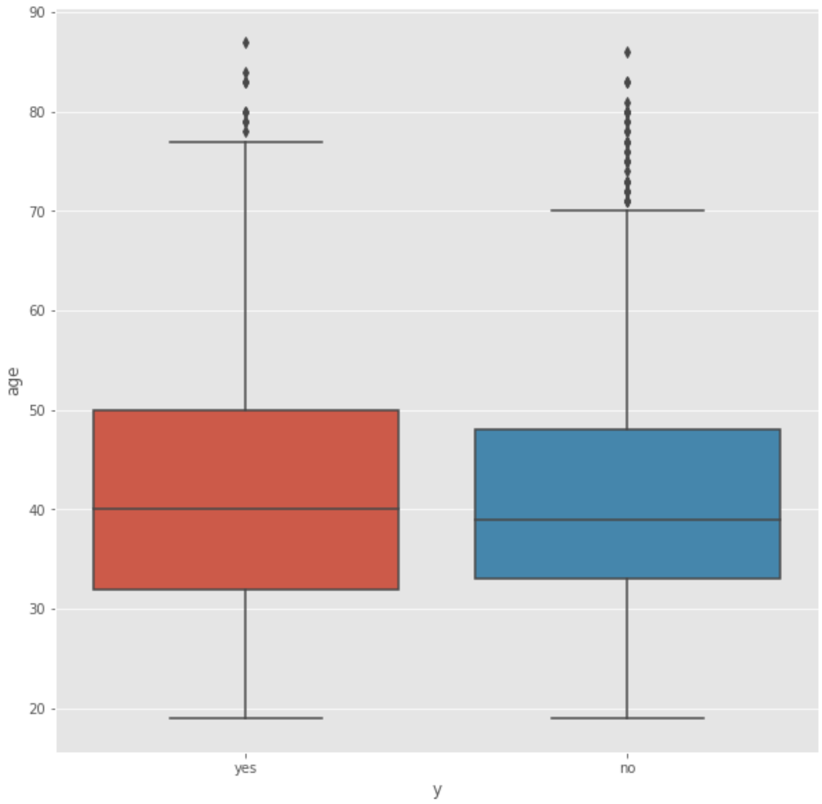
5.Amongst the customers who had been contacted married seems to be most popular category in the marital status variable.



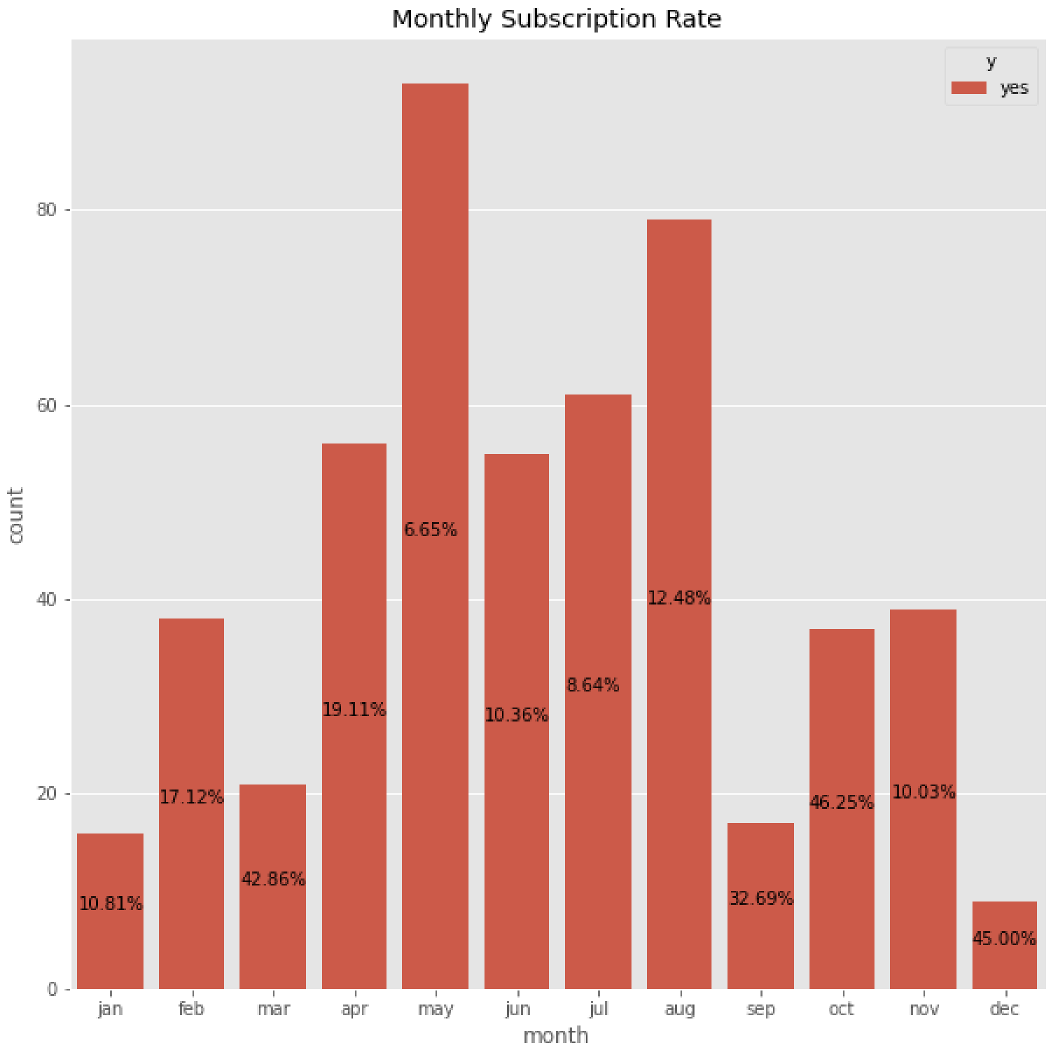
The above Bar graph plots the subscription count based on education levels and the highest subscribers are from the secondary education level followed by tertiary and then primary and least is unknown level category. The unsubscribed customers follow the same order of decrement as it is in that of subscribed customers.



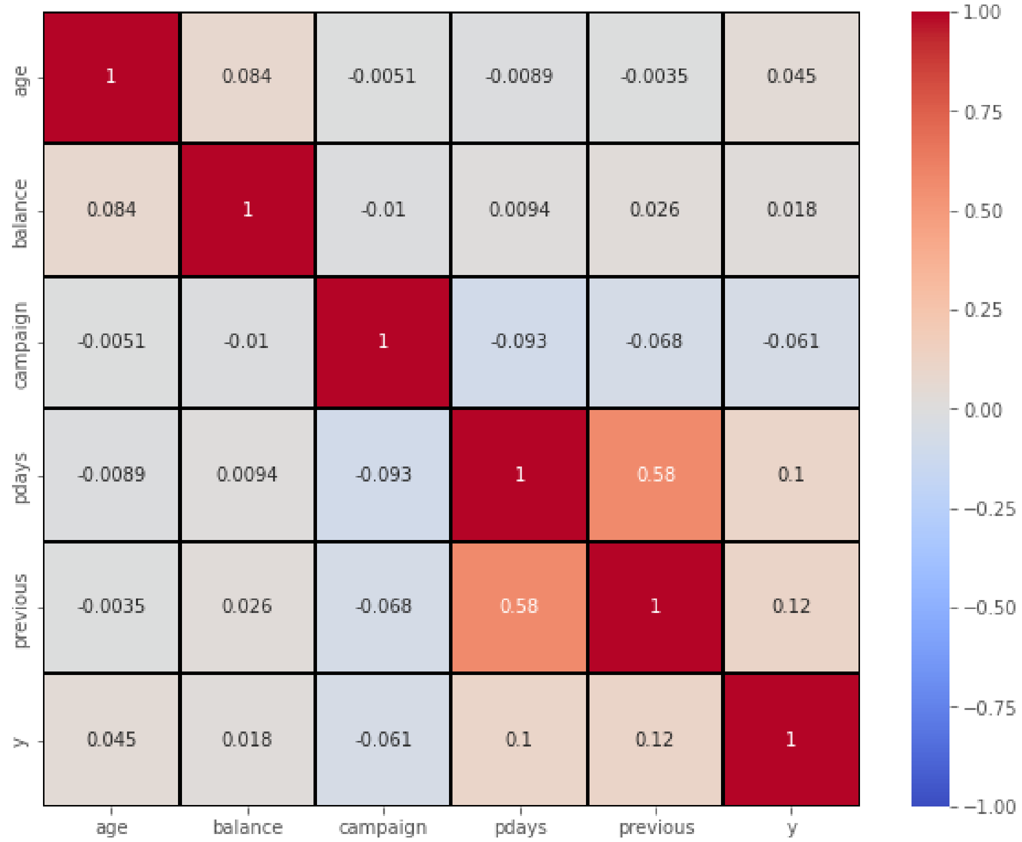
The above Bar graph plots the subscription count based on marital status and the highest subscribers are married followed by single and then divorced is least in the category. The unsubscribed customers follow the same order of decrement as it is in that of subscribed customers.



The minimum values and the means of the ages of the people for people who subscribed to a term deposit, and the ones who didn't is approximately the same. The maximum value for both differs around nine years and we see there are many outliers in both the categories.



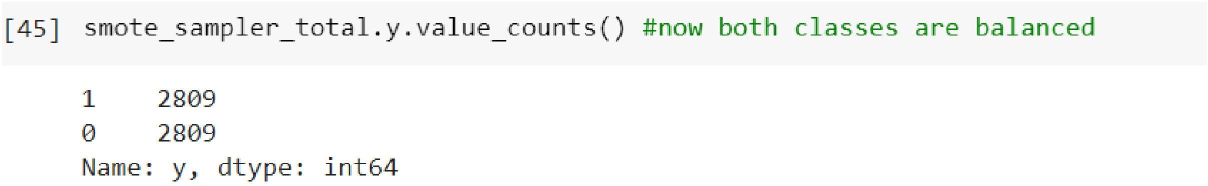
The above countplot patches the percentile rate of subscribed customers and it shows that the month of October and December months have comparitively higher positive subscription rates over contact made for the respective months, although May has a higher count of positive subscription.



The correlation matrix shows that the highest positively correlated variables are "previous" and "pdays" and all other variables are very weakly correlated to each other.

SMOTE sampling: -

Synthetic minority oversampling technique used to balance the dataset in terms of target variable. This technique is implemented so that the model which we use to classify the data point should not be biased towards one of the classes.



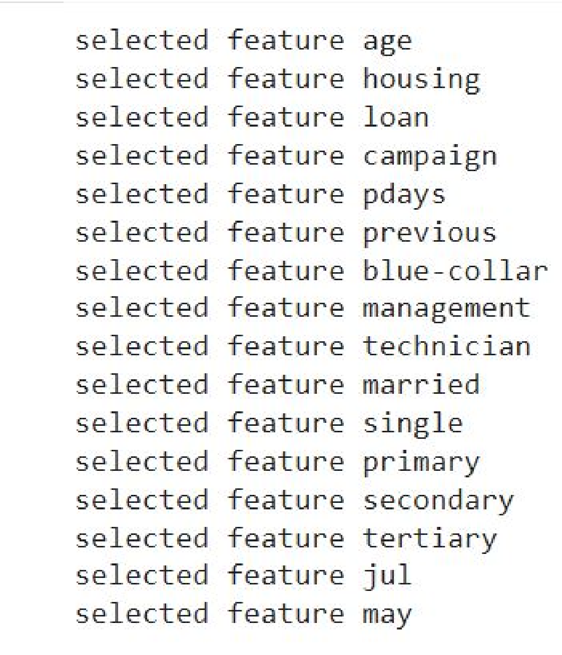
```
[45] smote_sampler_total.y.value_counts() #now both classes are balanced

     1    2809
     0    2809
     Name: y, dtype: int64
```

We can see that once the smote technique is carried out both the classes are now balanced and we can use this data for modeling.

Feature selection: -

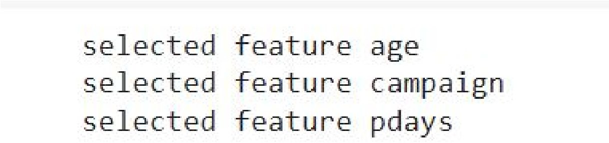
1. L2 (Ridge) penalty feature selection: -



```
selected feature age
selected feature housing
selected feature loan
selected feature campaign
selected feature pdays
selected feature previous
selected feature blue-collar
selected feature management
selected feature technician
selected feature married
selected feature single
selected feature primary
selected feature secondary
selected feature tertiary
selected feature jul
selected feature may
```

L2 penalty feature selection is selected as the best method parameter by doing grid search over the data and we see that the L2 penalty selects 16 features amongst all the variables and we use these features as our predictors to perform modeling.

2. Elastic Net feature selection: -



```
selected feature age
selected feature campaign
selected feature pdays
```

The second technique we use for feature selection is the elastic net method which is a combination of both L1(lasso) and L2(Ridge) it is done to verify the results of L2 penalty method but unfortunately it doesn’t match and it selects only three features as the best predictors. Using less features might cause underfitting problem where train score would be much lesser than that of test score and hence we go with the features selected by the L2 penalty feature selection method.

MODELING

RESULTS & OBSERVATIONS:

1. Logistic Regression:

Model Description:

Logistic regression classifies records using sigmoid as an activation function. It converts the input to a value between 0 and 1. The shape of the function for all possible inputs is an S-shape from 0 to 1 through 0.5.

$$y = \frac{1}{1+e\hat{}(-x)}$$

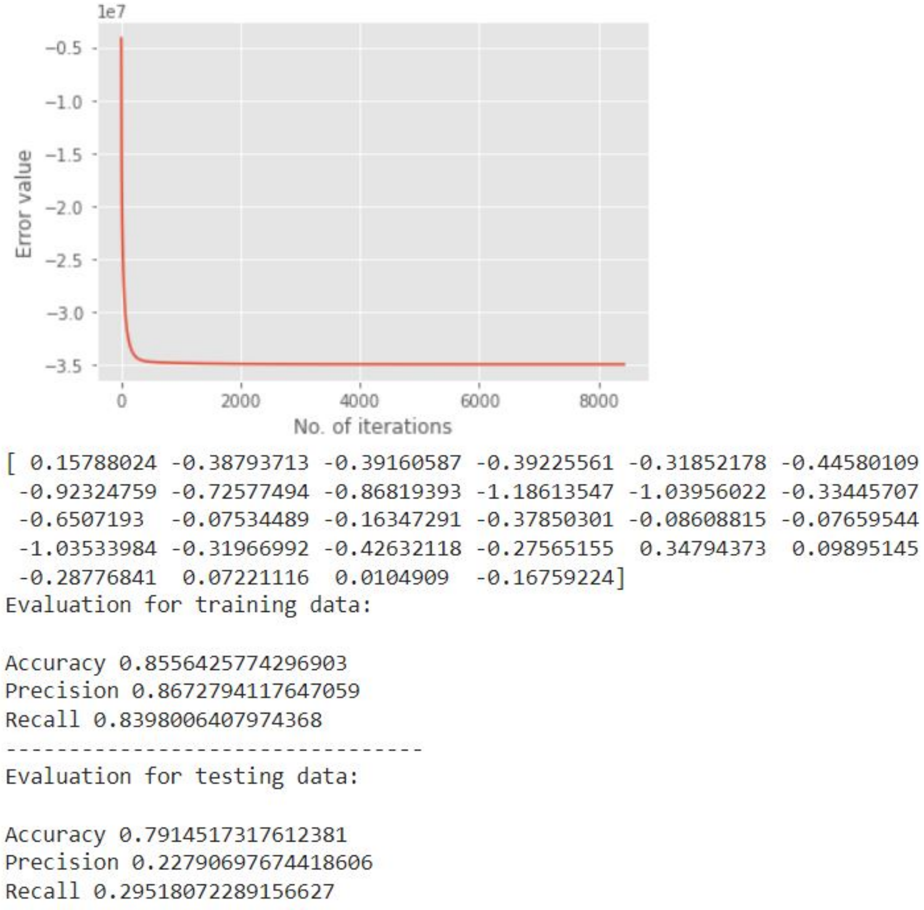
We do logistic regression with smote sampling data. The parameters which we use for this classification method are,

(a) Learning rate = 0.0001

(b) tolerance = 0

(c) maxiteration = 10000

Results: -



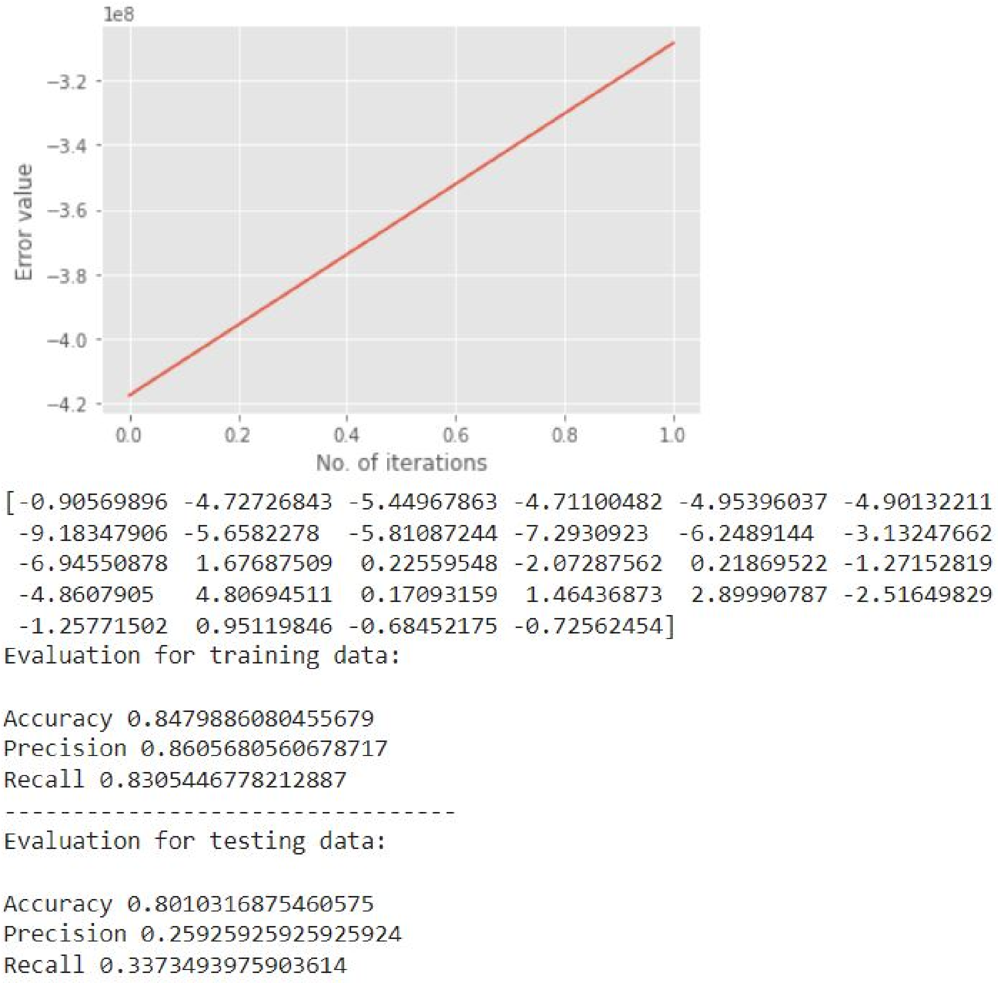
```
[ 0.15788024 -0.38793713 -0.39160587 -0.39225561 -0.31852178 -0.44580109
 -0.92324759 -0.72577494 -0.86819393 -1.18613547 -1.03956022 -0.33445707
 -0.6507193  -0.07534489 -0.16347291 -0.37850301 -0.08608815 -0.07659544
 -1.03533984 -0.31966992 -0.42632118 -0.27565155  0.34794373  0.09895145
 -0.28776841  0.07221116  0.0104909  -0.16759224]
Evaluation for training data:

Accuracy 0.8556425774296903
Precision 0.8672794117647059
Recall 0.8398006407974368
---------------------------------
Evaluation for testing data:

Accuracy 0.7914517317612381
Precision 0.22790697674418606
Recall 0.29518072289156627
```

The logistic regression model with learning rate = 0.001 gives a decent accuracy of 79.1% on test data, this is a good model to perform classification.

When the Learning rate = 0.01



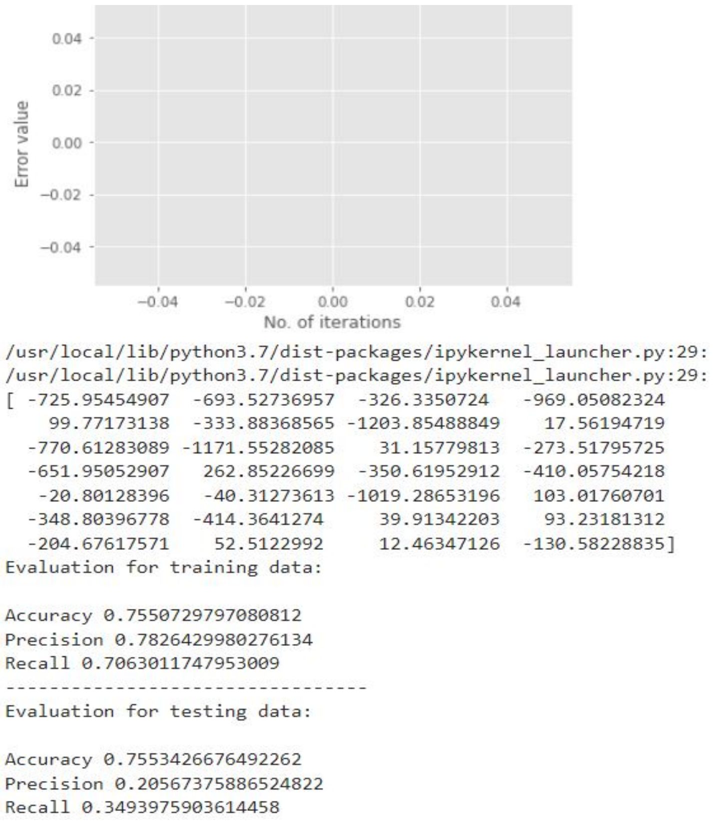
```
[-0.90569896 -4.72726843 -5.44967863 -4.71100482 -4.95396037 -4.90132211
 -9.18347906 -5.6582278  -5.81087244 -7.2930923  -6.2489144  -3.13247662
 -6.94550878  1.67687509  0.22559548 -2.07287562  0.21869522 -1.27152819
 -4.8607905   4.80694511  0.17093159  1.46436873  2.89990787 -2.51649829
 -1.25771502  0.95119846 -0.68452175 -0.72562454]
Evaluation for training data:

Accuracy 0.8479886080455679
Precision 0.8605680560678717
Recall 0.8305446778212887
---------------------------------
Evaluation for testing data:

Accuracy 0.8010316875460575
Precision 0.25925925925925924
Recall 0.3373493975903614
```

When the learning rate decreased to 0.01 the testing accuracy got little better to 80.1% and this is a better model than the previous one.

When the Learning rate = 1



```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:29:
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:29:
[ -725.95454907  -693.52736957  -326.3350724   -969.05082324
    99.77173138  -333.88368565 -1203.85488849    17.56194719
  -770.61283089 -1171.55282085    31.15779813  -273.51795725
  -651.95052907   262.85226699  -350.61952912  -410.05754218
   -20.80128396   -40.31273613 -1019.28653196   103.01760701
  -348.80396778  -414.3641274     39.91342203    93.23181312
  -204.67617571    52.5122992     12.46347126  -130.58228835]
Evaluation for training data:

Accuracy 0.7550729797080812
Precision 0.7826429980276134
Recall 0.7063011747953009
---------------------------------
Evaluation for testing data:

Accuracy 0.7553426676492262
Precision 0.20567375886524822
Recall 0.3493975903614458
```

When the learning rate is 1 both training accuracy and testing accuracy decreases so we can say that higher learning rate might tend to underfit the model.

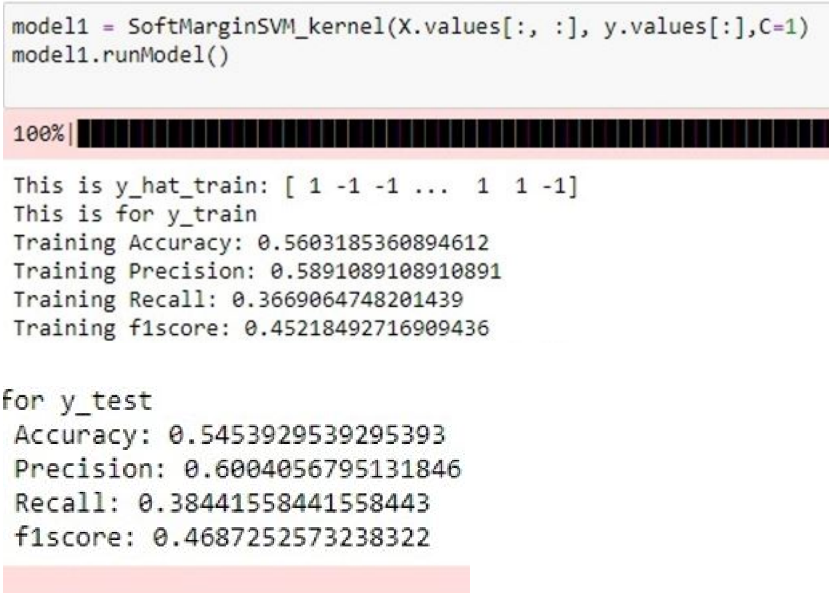
2. Support Vector Machine: (SVM)

Model Description: -

SVM is a classification model which helps to segregate non-linear data by taking the data points to the higher dimensional space and this method is known as kerneling, there are different types of kernels such as “RBF”, “LINEAR”, “PLOYNOMIAL” and etc.

Results: -

Kernel used in this case is “RBF”



```
model1 = SoftMarginSVM_kernel(X.values[:, :], y.values[:],C=1)
model1.runModel()
```

```
100%|
```

```
This is y_hat_train: [ 1 -1 -1 ...  1  1 -1]
This is for y_train
Training Accuracy: 0.5603185360894612
Training Precision: 0.5891089108910891
Training Recall: 0.3669064748201439
Training f1score: 0.45218492716909436
```

```
for y_test
 Accuracy: 0.5453929539295393
 Precision: 0.6004056795131846
 Recall: 0.38441558441558443
 f1score: 0.4687252573238322
```

The SVM classifier gives training accuracy = 56% and testing accuracy = 54% which is way lesser than that of logistic regression and hence we don't want this model to be used for the data points.

3.Neural Network: -

Model Description: -

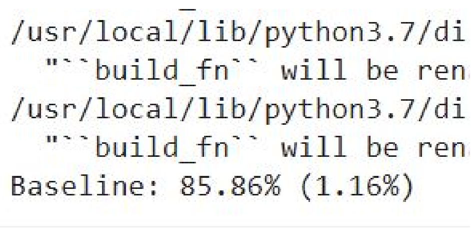
A Neural network is a deep learning technique which has input layer connected to the hidden layer for which we have several activation functions such as "sigmoid", "Tanh", "Relu" and etc, the hidden layers are further connected to the output layer which yields the predicted output value.

The Feed forward neural network is used in this case to predict the classification output value.

The parameters used here are,

(a) epoch =100
(b) batch_size = 10
(c) verbose = 0

One hidden layer Neural network: -

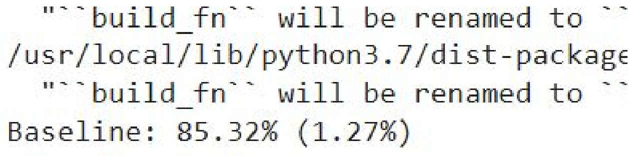


```
/usr/local/lib/python3.7/di
  "``build_fn`` will be ren
/usr/local/lib/python3.7/di
  "``build_fn`` will be ren
Baseline: 85.86% (1.16%)
```

The epochs is 100 and the and the batch size is 10 the test accuracy attained is 85.6%.This is a good model.

Two hidden layer Neural network: -

epoch= 100
batch_size = 10
verbose = 0



```
  "``build_fn`` will be renamed to ``
/usr/local/lib/python3.7/dist-package
  "``build_fn`` will be renamed to ``
Baseline: 85.32% (1.27%)
```

The test accuracy attained is 85.3 %, which is lesser than that of the one hidden layer model and hence it shows that the one hidden layer model is sufficient for this data.

Discussion: -

Throughout the course of the project, we incorporated techniques taught in class. For our baseline model, we used a logistic regression model. This model is a linear classifier. As the data we used was linearly separable, we saw that the baseline model performed well on the dataset.

The second classification Machine Learning model we implemented was SVM using RBF Kernel. As this model is a non-linear classifier, it performed worst as compared to the logistic regression model.

The final model we implemented is a feed-forward neural network using the Tensorflow and Keras packages. Through the implementation process, by playing around with the number of hidden layers, we found out that one hidden layer neural network model performs the best.

REFERENCES

[Moro et al., 2014] 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