Saheli Saha
8123698628
sahelisaha06@gmail.com

Deliverables:
Provide the following:
 The source code you used to build the model and make predictions. (You are free to use any language and any open-source package/library)
 Briefly answer the following questions:
- o Describe your model and why did you choose this model over other types of models?
- o Describe any other models you have tried and why do you think this model preforms better?
- o How did you handle missing data?
- o How did you handle categorical (string) data?
- o How did you handle unbalanced data?
- o How did you test your model?

I have considered six approaches to select my model those are described below. At first I have given a synopsis of all the approaches and at the end I have described my model. There I have briefly explained how I have handled missing data, categorical data and unbalanced data after processing the data, mentioned the application of feature selection technique in the data and finally application of prediction algorithms with cross validation and parameter tuning technique to build my model more accurately.
In the last section, I have mentioned the reason for choosing my model by giving a brief comparison of my model with other approaches with accuracy level.
I have attached the source code named **'Customer_Response_Prediction.py'**.

1. **Missing data replacement and application of Classification Algorithms:**
   - ➢ At first I deleted the observations where 'profession' is unknown. As I noticed for 61 unknown profession observations there are 6 blank 'days of week' values, 36 unknown/blank 'schooling' values, 2 unknowns 'marital' values, 19 values of 'default' are unknown, 54 values of 'default' are non-existent. Hence, I concluded that deleting 61 observations out of 7414 observation will not affect my prediction model. So, after dropping 61 observations there are 7353 observations to work with.
   - ➢ For other categorical features, I have replaced missing data with most frequently occurred value in the respective columns, then I have ranked them in a meaningful manner, such as for month I have used the corresponding numerical values of it. Likewise, for other categorical values as per their verity I have assigned numerical values to them.
   - ➢ For 'pdays' and 'pmonths' I have replaced 999 to 0 as 999 can confuse my machine learning model as a variable with very high value.
   - ➢ Divided it test and train data with different ratio (10:90, 20:80, 25:75) to check for unbalanced data and did feature scaling.
   - ➢ Applied Support Vector Machine, Naive Bayes classifier and Logistic Regression on the data.

Saheli Saha
8123698628
sahelisaha06@gmail.com

- At the end, I have used confusion matrix, classification report and classification score to understand the efficiency of the model.

2. **Missing data replacement and application of Factor Analysis and Classification Algorithms:**

    I used exactly same approach as discussed in the first approach. The only difference is I have applied Factor Analysis for feature selection as my data is a mixture of categorical and quantitative variable. After that, applied the classification algorithms on it. Used confusion matrix, classification report and classification score for the model evaluation.

3. **Application of KNN for missing value imputation and Classification Algorithms:**
    - For handling, missing data, I have used KNN algorithm I did some experiment with K parameter value and finally choose k = 15.
    - After the missing value imputed properly for handling categorical data I have ranked the data in a meaningful manner, such as for month I have used the numerical value. Likewise, for other categorical values as per their verity I have assigned numerical values to them.
    - For 'pdays' and 'pmonths' I have replaced 999 to 0 as 999 can confuse my machine learning model as a variable with very high value.
    - Then I have separated the target variable and independent variable and experimented with test size and train size in 25:75, 20:80 and 10:90 ratio to measure the unbalanced data and applied feature scaling for normalization.
    - Applied Support Vector Machine, Naive Bayes classifier and Logistic Regression on the processed data.
    - To measure my model used confusion matrix, classification report and classification score to understand the efficiency of the model.

4. **Application of KNN, Dummy Variables, Factor Analysis and Classification Algorithms:**
    - Similar to 4th approach I applied KNN to handle missing values.
    - For replacing categorical variables to numerical variables applied dummy variables.
    - For feature selection, I used Factor Analysis.
    - Finally, for prediction used Support Vector Machine, Naive Bayes classifier and Logistic Regression on the data.
    - I handled unbalanced data by feature scaling and taking three different test train (25:75, 20:80 and 10:90) ratios.
    - I have used confusion matrix, classification report and classification score to understand the efficiency of the model.

5. I tried to process my data by deleting all missing values. There I got only 3552 observations out of 7414. Deleting almost 50% data is not a wise decision to go with so I did not move forward with this approach.

Saheli Saha
8123698628
sahelisaha06@gmail.com

6. **Missing data replacement and Dummy Variables for Categorical Data, Application of Factor Analysis and Logistic Regression (Chosen Model):**

➢ Description of the model: -

Missing Data Handling: -

- There are 61 missing value for attribute **profession**. I deleted those observations. I noticed for 61 unknowns 'profession' observations there are 6 blank 'days of week' values, 36 unknown/blank 'schooling' values, 2 unknowns for 'marital' values, 19 values of 'default' are unknown, 54 values of 'default' is non-existent. Hence, I concluded that deleting 61 observations out of 7414 observation will not affect my prediction model. After dropping 61 observations there 7353 observations to work with, which is 99.17% of the original data.
- For unknown values in 'schooling', 'marital', 'default', 'housing, 'loan', 'day_of_week' I have replaced them with the most frequent one in that column, as that value is already has maximum occurrence so adding more of it will not effect my model.
- For one quantitative feature 'custAge' with missing values I have replaced those missing values with the median of the column. I experimented my model with both mean and median. As both are giving same level of accuracy I have decided to replace missing values of 'custAge' with median of the column.
- Like other approaches I have replaced 'pdays' and 'pmonths' 999 value replaced with 0. As 999 can confuse my machine learning model as a variable with very high value.

Categorical Data Handling: -

- Features with two kind categorical values such as - 'default', 'housing', 'loan', 'contact', 'responded' (Target Variable) replaced it to dummy variables, 0 and 1 while replacing the unknown values with the most frequent values.
- For rest of the categorical features – 'profession', 'schooling', 'marital', 'month', 'day_of_week', 'poutcome' I applied dummy variables method directly.

Saheli Saha
8123698628
sahelisaha06@gmail.com

Unbalanced Data Handling: -

- Processed the whole data and divided it for target variable and independent variables.
- I used different test and train data ratio i.e. 10:20, 20:80, 25:75 to check the balance of the data. My target variable contains only two kind of values 0 or 1 (yes:1, no:0) I directly looked for number of 0 and 1 present in both my test target variable (y_test) and train target variable(y_train)
    - For 10:20 :- y_train- 0: 6353, 1: 264 ---3.9% values are 1.
      y_test- 0: 693, 1: 43 ----- 5.8% values are 1.
    - For 20:80:-y_train- 0: 5643, 1: 239 --- 4.1 % values are 1.
      y_test- 0: 1403, 1: 68 ----- 4.6 % values are 1.
    - For 25:75 :- y_train- 0: 5292, 1: 222- 4.02 % values are 1.
      y_test- 0: 1754, 1: 85 --- 4.62 % values are 1.

According to the above statistics 20:80 test and train data ratio is the best to apply prediction model on.

- I used confusion matrix to understand the error rate. If the error rate is high, then my data is unbalanced.
    - For 10:90 test train data ratio the error rate is 5.8%
    - For 20:80 test train data ratio the error rate is 4.6%
    - For 25:75 test train data ratio the error rate is 4.6%

Although error rate is same for both 20:80 and 25:75 test train data ratio, but considering both (0,1) statistics and confusion matrix I have considered 20:80 observation is the best one to go with.

Feature Selection: -

- After applying Dummy Variables, I got 57 features so for feature selection applied **Factor Analysis**. Factor Analysis is a powerful feature selection algorithm which works on both quantitative and categorical data.
- At first I applied Factor Analysis with n_components = None. This will give me the features with most variance in a rank wise manner which can be seen by calling components_ attribute.
- After looking at variances I selected 16 features. Hence, I choose- n_components= 16.
- Using n_components value as 16 applied Factor Analysis.

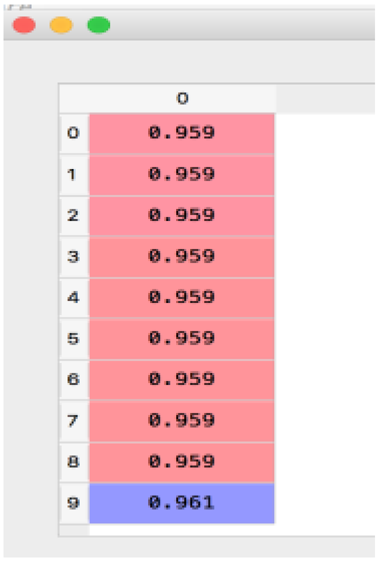
Saheli Saha
8123698628
sahelisaha06@gmail.com

Application of Logistic Regression for Prediction: -

- I choose three algorithms for prediction - **Support Vector Machine (SVM)**, **Naive Bayes classifier** and **Logistic Regression** on the data. Based on confusion matrix, classification report and classification score Logistic Regression performed best for 20:80 test train data ratio.

Logistic Regression for Prediction: -

- I used Logistic Regression for prediction. Logistic Regression is one of the most powerful algorithm used for classification problems.
- To understand efficiency of my model I used K- Fold cross validation method, for 10 iteration (cv = 10) and as I have a large dataset I have considered n_job = -1.
- K- Fold cross validation will run my model to nine different combination of test and train data and finally on the assigned test set. Based on that it will give the accuracy score. Expected output must have high accuracy with low variance. Following is the result of accuracy score for ten different iterations.

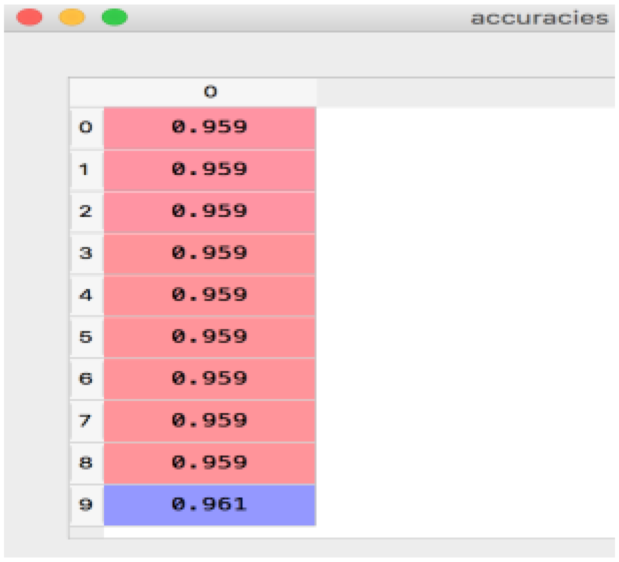


	0
0	0.959
1	0.959
2	0.959
3	0.959
4	0.959
5	0.959
6	0.959
7	0.959
8	0.959
9	0.961

- Mean of the accuracy = 0.9593678671≈ 0.96 Variance of the accuracy = 0.000484275. Accuracy level is 96% and variance is quite low.
- To improve my model, I did parameter tuning using GridSearch. I got C = 1 to be the best parameter.
- Best score = 0.959 ≈ 0.96.
- I got the same accuracy level so my model is performing good and I added C = 1 as parameter.

Saheli Saha
8123698628
sahelisaha06@gmail.com

- Using these parameters, I executed my model again which gave me the following accuracy score for ten iteration.



	0
0	0.959
1	0.959
2	0.959
3	0.959
4	0.959
5	0.959
6	0.959
7	0.959
8	0.959
9	0.961

- These values are same as previous one. Mean and Variance is also same
- As the accuracy score is 96% I choose this model.
- Finally, I have used confusion matrix, classification report and classification score to understand the efficiency of the model.

Parameters to select this model: -

- Confusion Matrix: - A breakdown of predictions into a table showing correct and incorrect predictions. Diagonal values (0,1) and (1,0) indicates the error values.

- Classification Report: - This builds a detailed text report showing the main classification metrics. We will look at the f1-score which is derived jointly from precision and recall values. f1-score varies from 0 to 1. 0 is considered to be worst where 1 is the best.

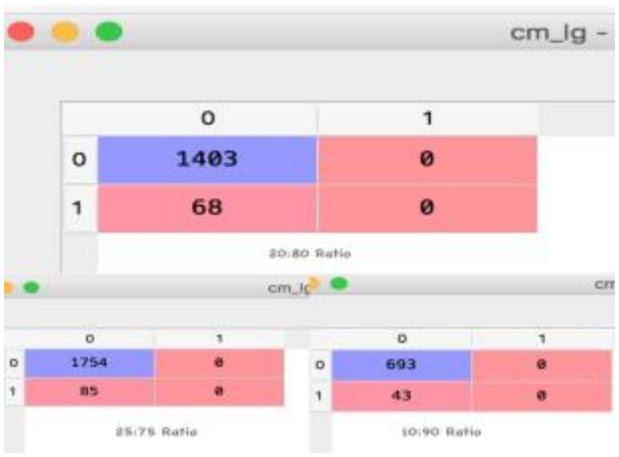
- Classification Score: - This Returns the mean accuracy on the given test data and labels.

Saheli Saha
8123698628
sahelisaha06@gmail.com

Model Test For chosen approach: -

I have used above mentioned three metrics to analyze the accuracy of my model. For different test train dataset ratio- 10:90, 20:80, 25:75.

<u>Confusion Matrix for Logistic Regression: -</u>



20:80 Ratio

	0	1
0	1403	0
1	68	0

25:75 Ratio

	0	1
0	1754	0
1	85	0

10:90 Ratio

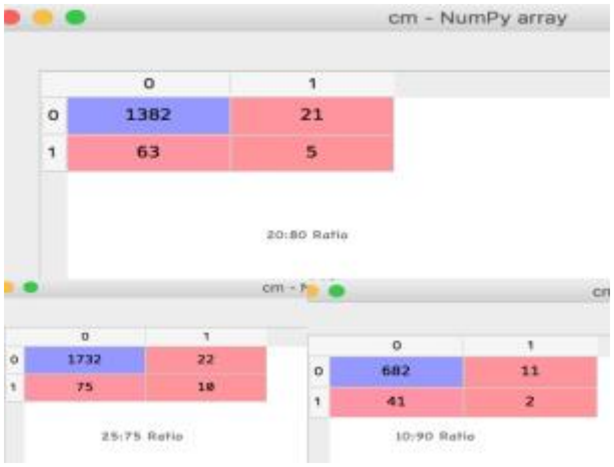
	0	1
0	693	0
1	43	0

Error rate for Test-Train Data Ratio is 10:90 = $\frac{43}{43+693} * 100 = \frac{43}{736} * 100$ = 5.8%

Error rate for Test-Train Data Ratio is 20:80 = $\frac{68}{68+1403} * 100 = \frac{68}{1471} * 100$ = 4.6% (chosen one)

Error rate for Test-Train Data Ratio is 25:75 = $\frac{85}{85+1754} * 100 = \frac{85}{1839} * 100$ = 4.6%

<u>Confusion Matrix for Support Vector Machine: -</u>



20:80 Ratio

	0	1
0	1382	21
1	63	5

25:75 Ratio

	0	1
0	1732	22
1	75	10

10:90 Ratio

	0	1
0	682	11
1	41	2

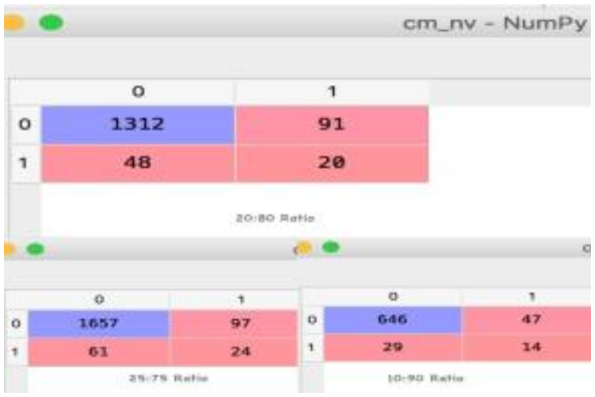
Saheli Saha
8123698628
sahelisaha06@gmail.com

Error rate for Test-Train Data Ratio is 10:90 = $\frac{52}{736} * 100 = 7.07\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{84}{1471} * 100 = \frac{68}{1471} * 100 = 5.7\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{85}{1839} * 100 = \frac{85}{1839} * 100 = 5.3\%$

Confusion Matrix for Naïve Bayes: -


cm_nv - NumPy

	0	1
0	1312	91
1	48	20

20:80 Ratio

	0	1
0	1657	97
1	61	24

25:75 Ratio

	0	1
0	646	47
1	29	14

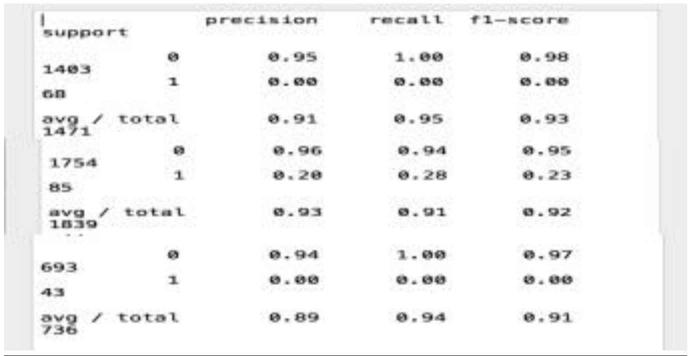
10:90 Ratio

Error rate for Test-Train Data Ratio is 10:90 = $\frac{76}{736} * 100 = 10.33\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{139}{1471} * 100 = 9.45\%$

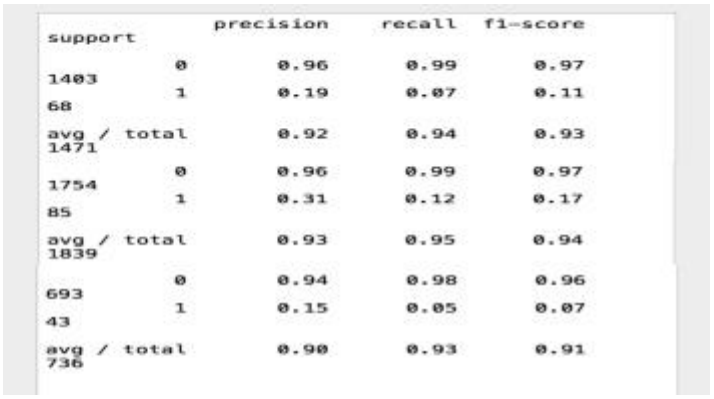
Error rate for Test-Train Data Ratio is 25:75 = $\frac{158}{1839} * 100 = 8.6\%$

Classification Report for Logistic Regression: -


```
|             precision    recall  f1-score
support

          0       0.95      1.00      0.98
1403
          1       0.00      0.00      0.00
68

avg / total       0.91      0.95      0.93
1471
          0       0.96      0.94      0.95
1754
          1       0.20      0.28      0.23
85

avg / total       0.93      0.91      0.92
1839

          0       0.94      1.00      0.97
693
          1       0.00      0.00      0.00
43

avg / total       0.89      0.94      0.91
736
```

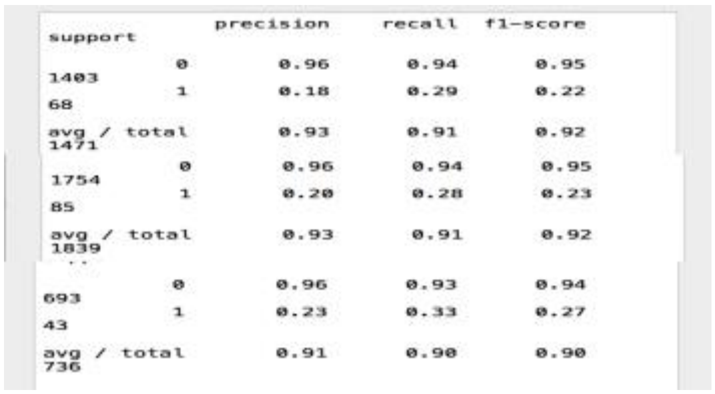
Saheli Saha
8123698628
sahelisaha06@gmail.com

Classification Report for Support Vector Machine: -



```
              precision    recall  f1-score
support

          0       0.96      0.99      0.97
1403
          1       0.19      0.07      0.11
68

avg / total       0.92      0.94      0.93
1471
          0       0.96      0.99      0.97
1754
          1       0.31      0.12      0.17
85

avg / total       0.93      0.95      0.94
1839
          0       0.94      0.98      0.96
693
          1       0.15      0.05      0.07
43

avg / total       0.90      0.93      0.91
736
```

Classification Report for Naïve Bayes: -



```
              precision    recall  f1-score
support

          0       0.96      0.94      0.95
1403
          1       0.18      0.29      0.22
68

avg / total       0.93      0.91      0.92
1471
          0       0.96      0.94      0.95
1754
          1       0.20      0.28      0.23
85

avg / total       0.93      0.91      0.92
1839
 ..
          0       0.96      0.93      0.94
693
          1       0.23      0.33      0.27
43

avg / total       0.91      0.90      0.90
736
```

As we know for 10:90, 20:80 and 25:75 there are 736, 1471 and 1839 observation in the test set respectively. According to that average F score is given above.

Classification Score for Support Vector Machine: -

When Test-Train Data Ratio is 10:90: -
```
0.929
```
When Test-Train Data Ratio is 20:80: -
```
0.943
```
When Test-Train Data Ratio is 25:75: -
```
0.944
```

Classification Score for Logistic Regression: -

When Test-Train Data Ratio is 10:90: -
```
0.942
```

Saheli Saha
8123698628
sahelisaha06@gmail.com

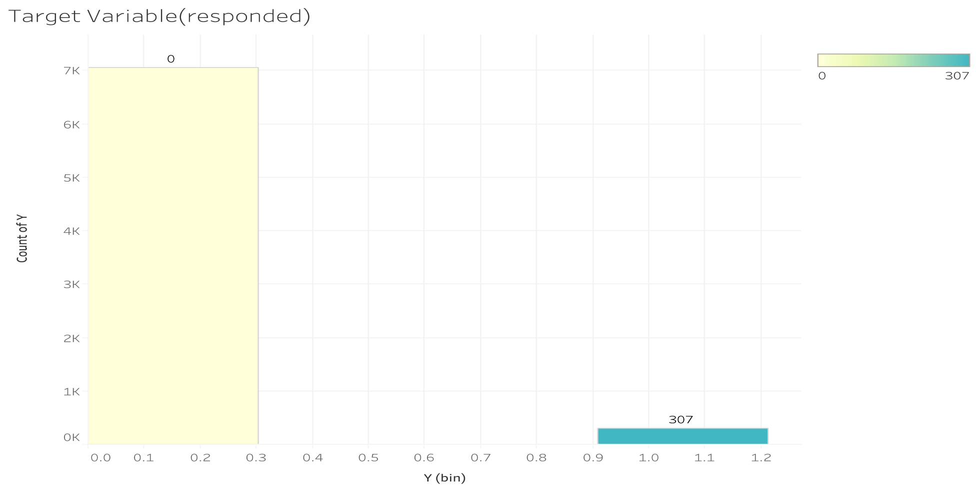
When Test-Train Data Ratio is 20:80: -
0.954
When Test-Train Data Ratio is 25:75: -
0.954

Classification Score for Naïve Bayes: -

When Test-Train Data Ratio is 10:90: -
0.89
When Test-Train Data Ratio is 20:80: -
0.90
When Test-Train Data Ratio is 25:75: -
0.91

For Support Vector Machine, 25:75 ratio observation has F score = 0.94 but it has error rate 5.7% and classification score is also 0.944. Whereas for Logistic Regression 20:80 ratio observation, error rate 4.6% and classification score is 0.954 hence, Logistic Regression with 20:80 test train ratio is the best model. Naive Bayes error rate, F score and classification are not good enough compared to Logistic Regression and Support Vector Machine.

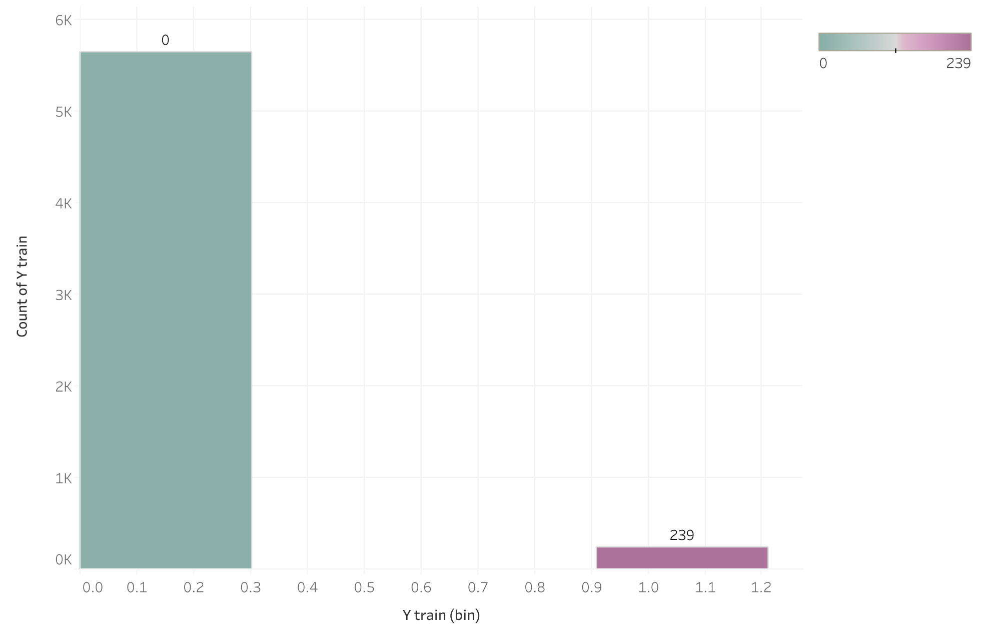
Visualization: -



The trend of count of Y for Y (bin). Color shows sum of Y. The marks are labeled by sum of Y.

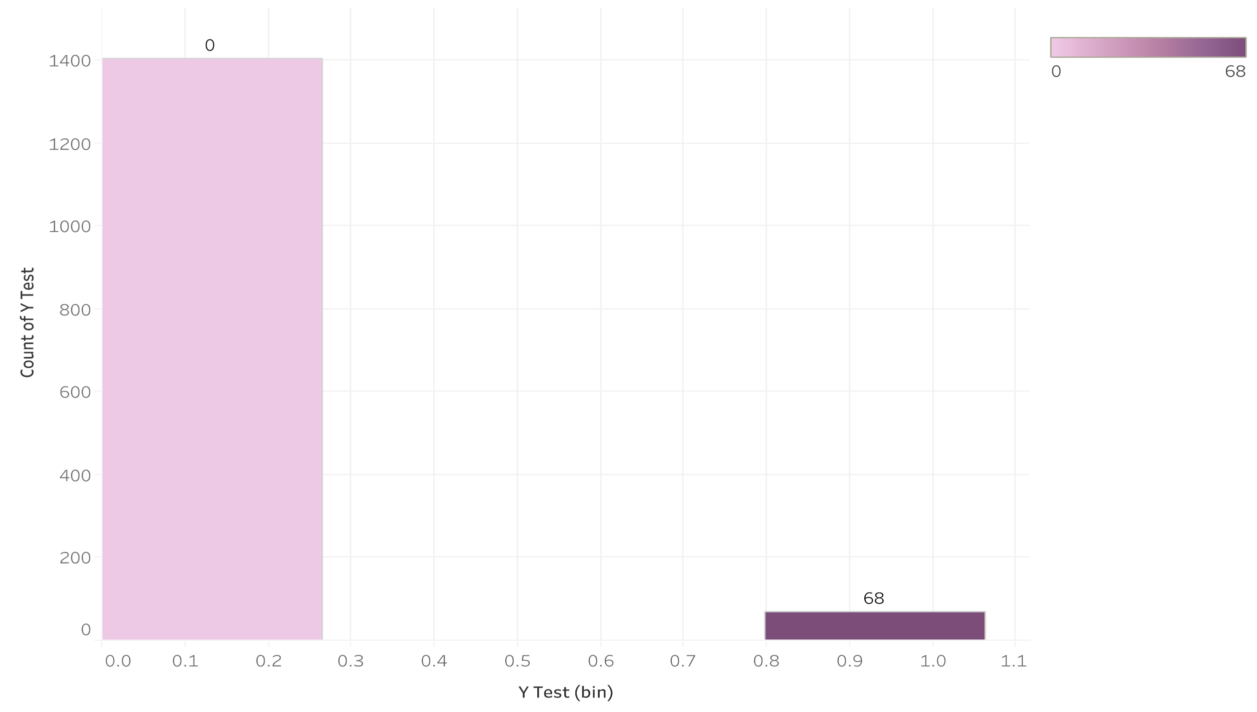
Saheli Saha
8123698628
sahelisaha06@gmail.com

Target Variable(responded) in Train Data



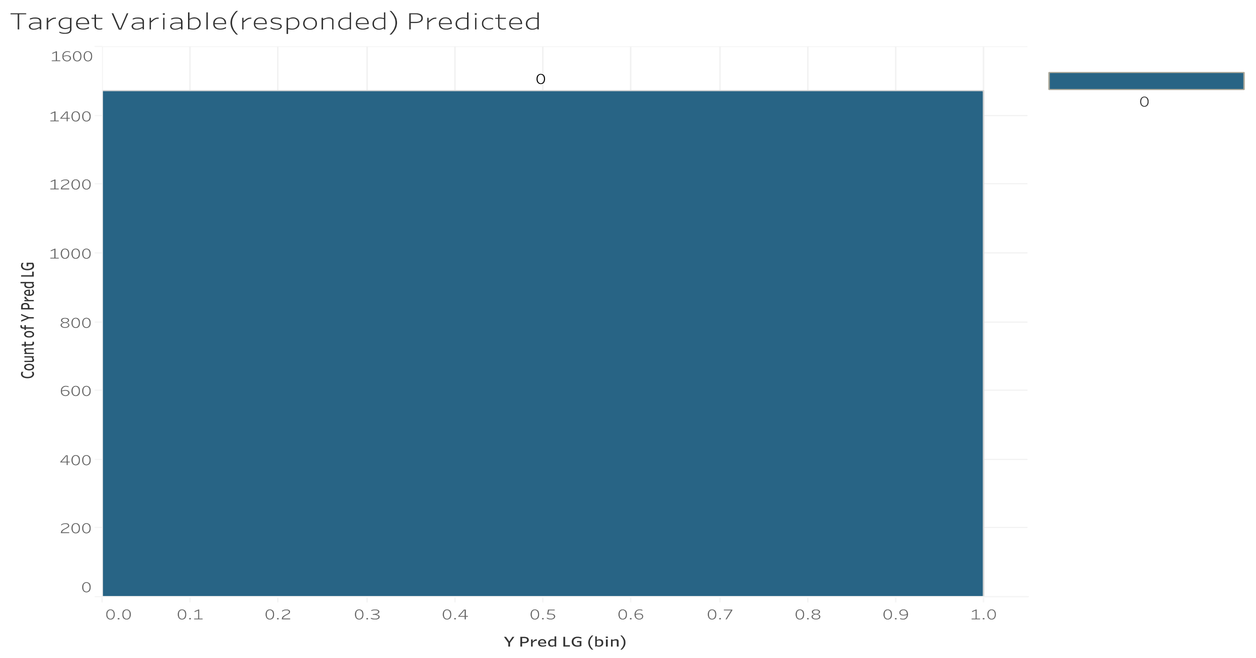
The trend of count of Y train for Y train (bin). Color shows sum of Y train. The marks are labeled by sum of Y train.

Target Variable(responded) in Test Data



The trend of count of Y Test for Y Test (bin). Color shows sum of Y Test. The marks are labeled by sum of Y Test.

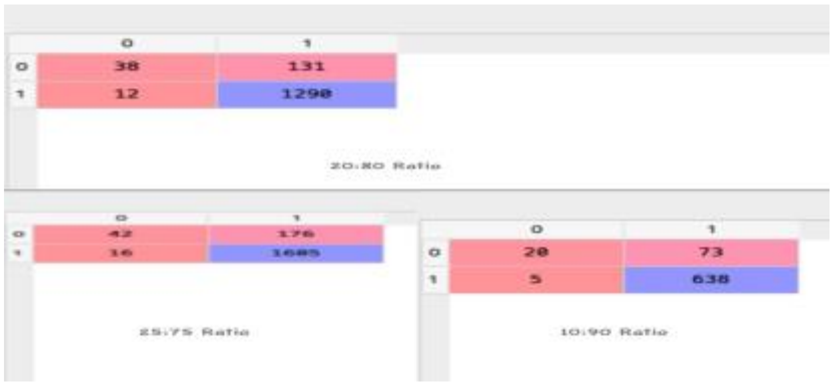
Saheli Saha
8123698628
sahelisaha06@gmail.com



The trend of count of Y Pred LG for Y Pred LG (bin). Color shows sum of Y Pred LG. The marks are labeled by sum of Y Pred LG. The view is filtered on Y Pred LG (bin), which keeps non-Null values only.

Model Test for the First Approach (Missing data replacement and application of Classification Algorithms): -

<u>Confusion Matrix for Support Vector Machine: -</u>



	0	1
0	38	131
1	12	1298

20:80 Ratio

	0	1
0	42	176
1	16	1605

25:75 Ratio

	0	1
0	20	73
1	5	638

10:90 Ratio

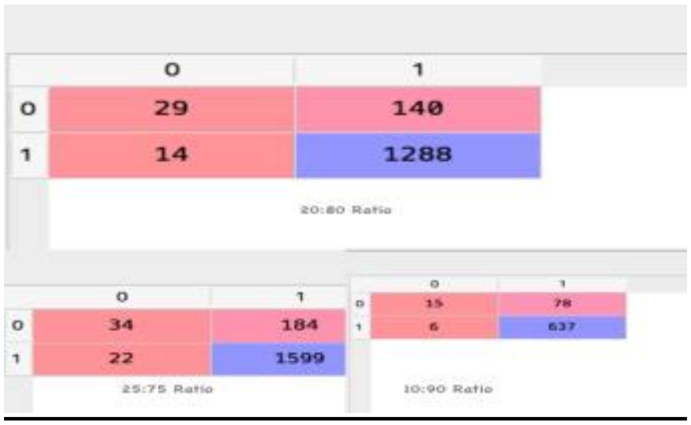
Error rate for Test-Train Data Ratio is 10:90 = $\frac{78}{736} * 100 = 10.61\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{143}{1471} * 100 = 9.72\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{192}{1839} * 100 = 10.44\%$

Saheli Saha
8123698628
sahelisaha06@gmail.com

Confusion Matrix for Logistic Regression: -



	0	1
0	29	140
1	14	1288

20:80 Ratio

	0	1
0	34	184
1	22	1599

25:75 Ratio

	0	1
0	15	78
1	6	637

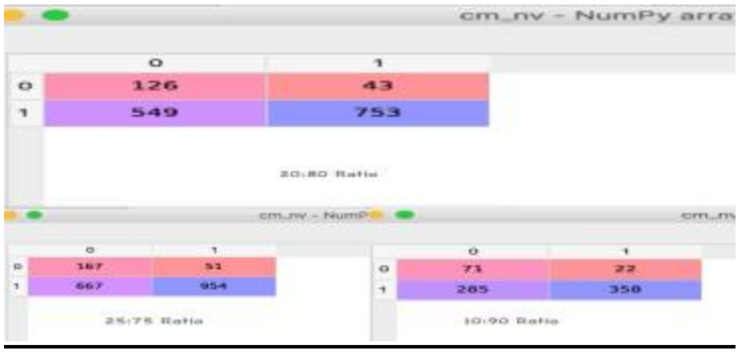
10:90 Ratio

Error rate for Test-Train Data Ratio is 10:90 = $\frac{84}{736} * 100 = 11.41\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{154}{1471} * 100 = 10.47\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{206}{1839} * 100 = 11.20\%$

Confusion Matrix for Naive Bayes: -



	0	1
0	126	43
1	549	753

20:80 Ratio

	0	1
0	167	51
1	667	954

25:75 Ratio

	0	1
0	71	22
1	285	358

10:90 Ratio

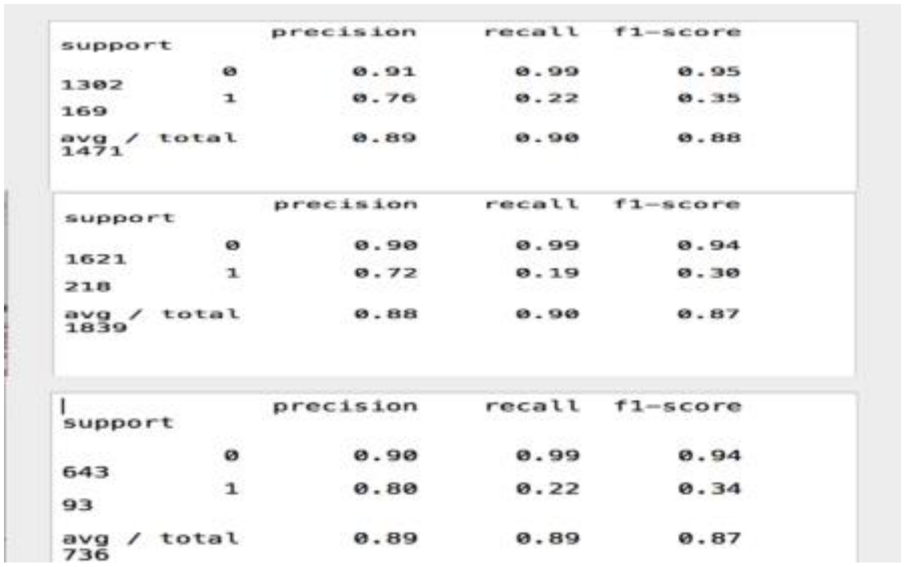
Error rate for Test-Train Data Ratio is 10:90 = $\frac{307}{736} * 100 = 41.71\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{592}{1471} * 100 = 40.24\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{718}{1839} * 100 = 39.04\%$

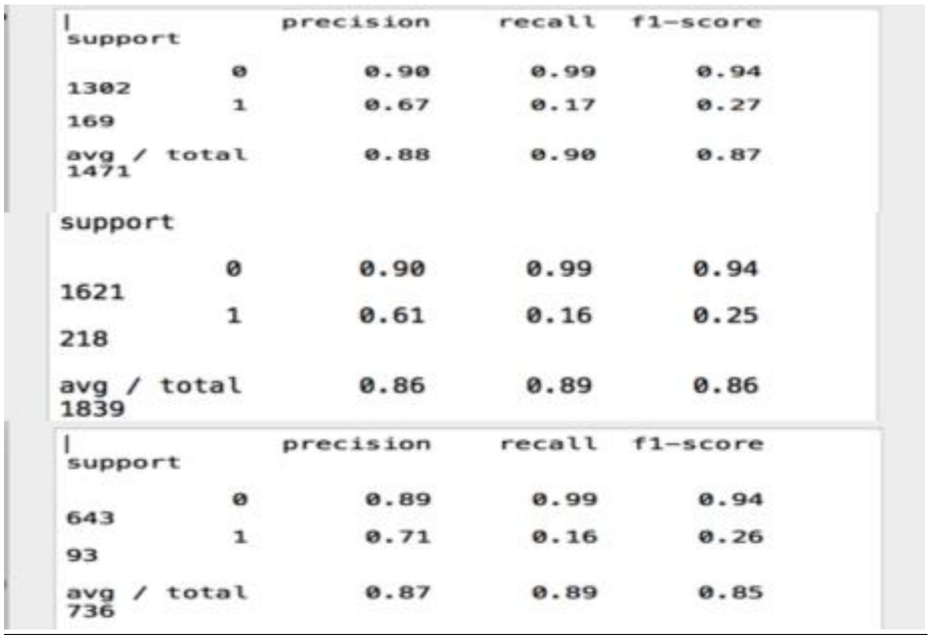
Saheli Saha
8123698628
sahelisaha06@gmail.com

Classification Report for Support Vector Machine: -



```
                   precision    recall  f1-score
support
             0         0.91      0.99      0.95
1302
             1         0.76      0.22      0.35
169
avg / total            0.89      0.90      0.88
1471

                   precision    recall  f1-score
support
             0         0.90      0.99      0.94
1621
             1         0.72      0.19      0.30
218
avg / total            0.88      0.90      0.87
1839

|                  precision    recall  f1-score
support
             0         0.90      0.99      0.94
643
             1         0.80      0.22      0.34
93
avg / total            0.89      0.89      0.87
736
```

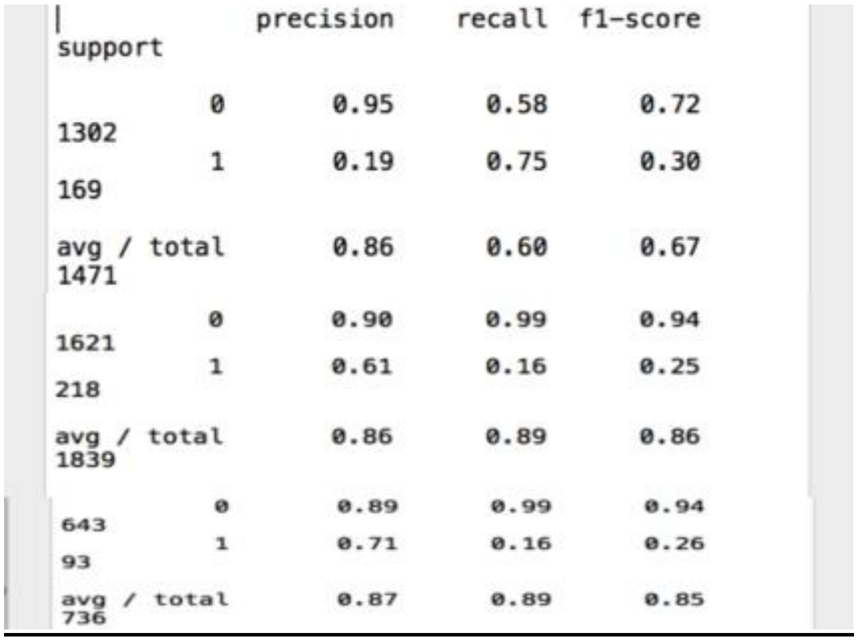
Classification Report for Logistic Regression: -



```
|                  precision    recall  f1-score
support
             0         0.90      0.99      0.94
1302
             1         0.67      0.17      0.27
169
avg / total            0.88      0.90      0.87
1471
support
             0         0.90      0.99      0.94
1621
             1         0.61      0.16      0.25
218
avg / total            0.86      0.89      0.86
1839
|                  precision    recall  f1-score
support
             0         0.89      0.99      0.94
643
             1         0.71      0.16      0.26
93
avg / total            0.87      0.89      0.85
736
```

Saheli Saha
8123698628
sahelisaha06@gmail.com

Classification Report for Naïve Bayes: -



```
|             precision    recall  f1-score
support

          0       0.95      0.58      0.72
1302
          1       0.19      0.75      0.30
169

avg / total       0.86      0.60      0.67
1471
          0       0.90      0.99      0.94
1621
          1       0.61      0.16      0.25
218

avg / total       0.86      0.89      0.86
1839
          0       0.89      0.99      0.94
643
          1       0.71      0.16      0.26
93

avg / total       0.87      0.89      0.85
736
```

Classification Score for Support Vector Machine: -

When Test-Train Data Ratio is 10:90: -
0.894
When Test-Train Data Ratio is 20:80: -
0.903
When Test-Train Data Ratio is 25:75: -
0.896

Classification Score for Logistic Regression: -

When Test-Train Data Ratio is 10:90: -
0.886
When Test-Train Data Ratio is 20:80: -
0.895
When Test-Train Data Ratio is 25:75: -
0.888

Classification Score for Naïve Bayes: -

When Test-Train Data Ratio is 10:90: -
0.894

Saheli Saha
8123698628
sahelisaha06@gmail.com

When Test-Train Data Ratio is 20:80: -

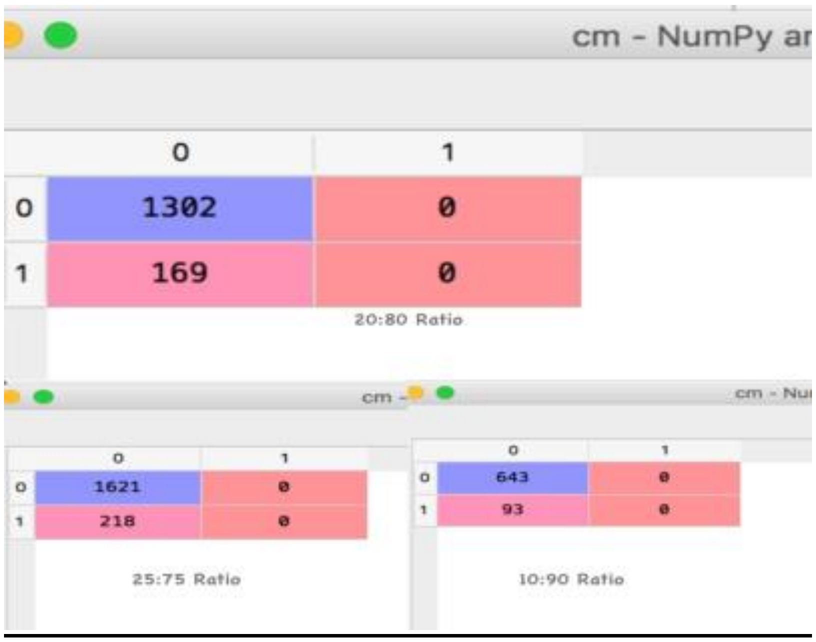
```
0.903
```

When Test-Train Data Ratio is 25:75: -

```
0.896
```

Model Test for the Second Approach (Missing data replacement and application of Factor Analysis and Classification Algorithms): -

<u>Confusion Matrix for Support Vector Machine: -</u>



	0	1
0	1302	0
1	169	0

20:80 Ratio

	0	1
0	1621	0
1	218	0

25:75 Ratio

	0	1
0	643	0
1	93	0

10:90 Ratio

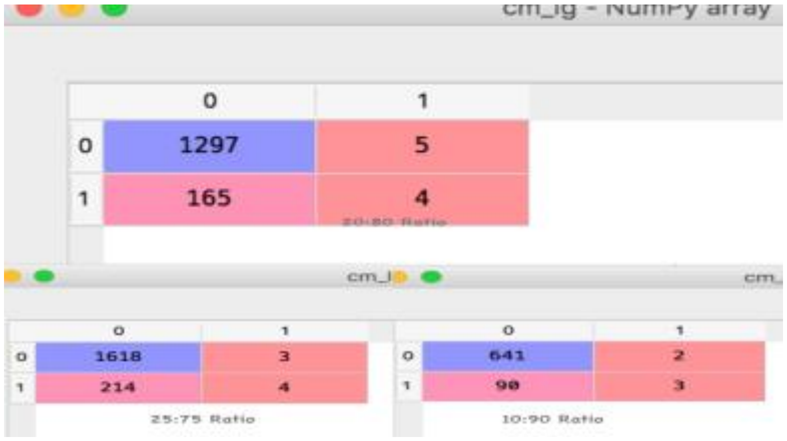
Error rate for Test-Train Data Ratio is 10:90 = $\frac{93}{736} * 100 = 12.63\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{169}{1471} * 100 = 11.49\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{218}{1839} * 100 = 11.85\%$

Saheli Saha
8123698628
sahelisaha06@gmail.com

<u>Confusion Matrix for Logistic Regression: -</u>



	0	1
0	1297	5
1	165	4

20:80 Ratio

	0	1
0	1618	3
1	214	4

25:75 Ratio

	0	1
0	641	2
1	90	3

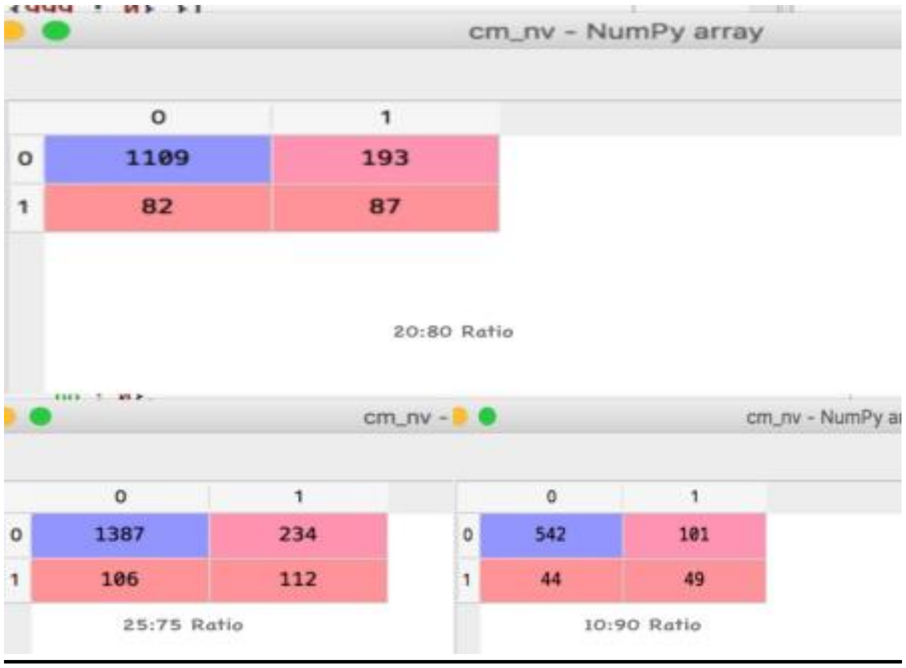
10:90 Ratio

Error rate for Test-Train Data Ratio is 10:90 = $\frac{92}{736} * 100 = 12.5\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{170}{1471} * 100 = 11.56\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{217}{1839} * 100 = 11.80\%$

<u>Confusion Matrix for Naïve Bayes: -</u>



	0	1
0	1109	193
1	82	87

20:80 Ratio

	0	1
0	1387	234
1	106	112

25:75 Ratio

	0	1
0	542	101
1	44	49

10:90 Ratio

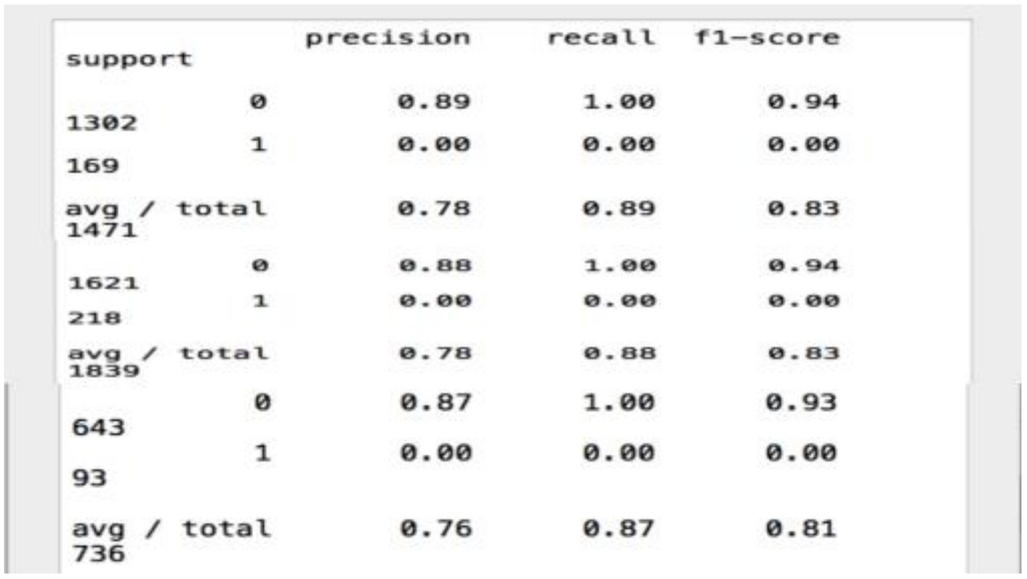
Error rate for Test-Train Data Ratio is 10:90 = $\frac{145}{736} * 100 = 19.70\%$

Saheli Saha
8123698628
sahelisaha06@gmail.com

Error rate for Test-Train Data Ratio is 20:80 = $\frac{275}{1471} * 100 = 18.70\%$

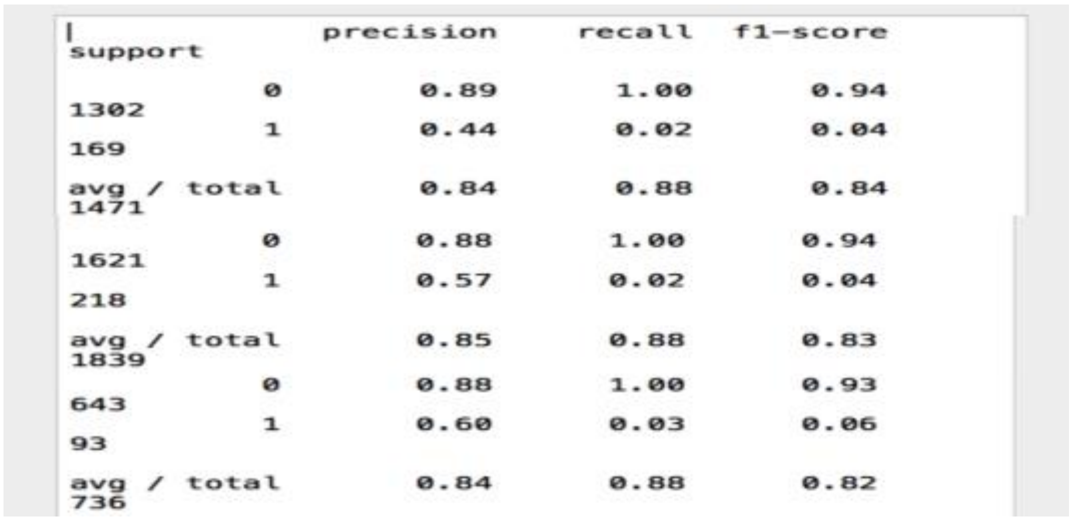
Error rate for Test-Train Data Ratio is 25:75 = $\frac{340}{1839} * 100 = 18.48\%$

Classification Report for Support Vector Machine: -



```
                 precision    recall  f1-score
support

            0       0.89      1.00      0.94
1302
            1       0.00      0.00      0.00
169

avg / total         0.78      0.89      0.83
1471
            0       0.88      1.00      0.94
1621
            1       0.00      0.00      0.00
218
avg / total         0.78      0.88      0.83
1839
            0       0.87      1.00      0.93
643
            1       0.00      0.00      0.00
93

avg / total         0.76      0.87      0.81
736
```

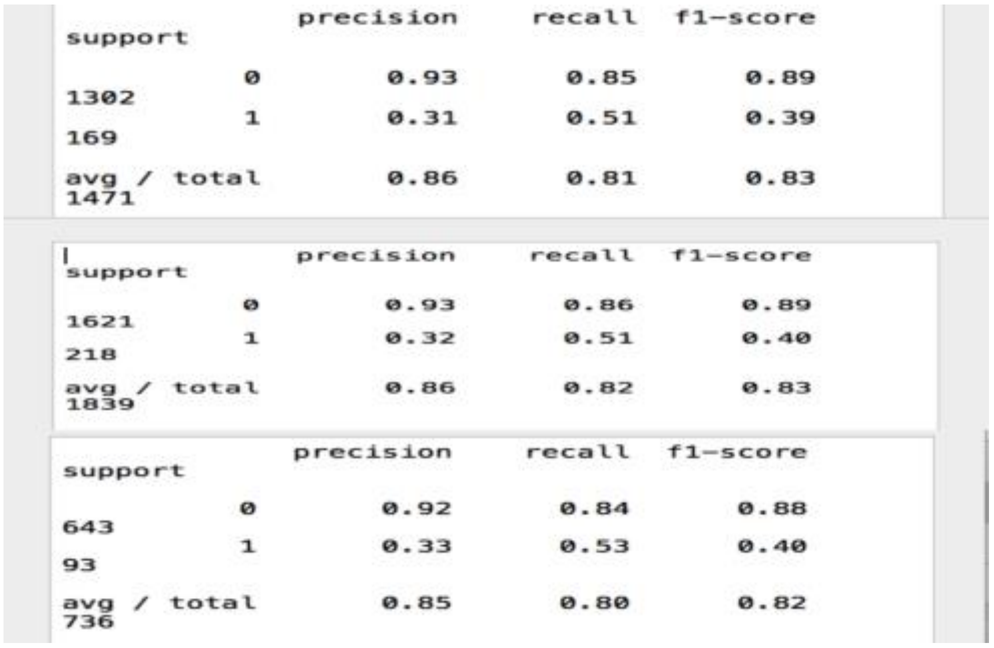
Classification Report for Logistic Regression: -



```
|                precision    recall  f1-score
support

            0       0.89      1.00      0.94
1302
            1       0.44      0.02      0.04
169

avg / total         0.84      0.88      0.84
1471
            0       0.88      1.00      0.94
1621
            1       0.57      0.02      0.04
218

avg / total         0.85      0.88      0.83
1839
            0       0.88      1.00      0.93
643
            1       0.60      0.03      0.06
93

avg / total         0.84      0.88      0.82
736
```

Saheli Saha
8123698628
sahelisaha06@gmail.com

<u>Classification Report for Naïve Bayes: -</u>



```
             precision    recall  f1-score
support

          0       0.93      0.85      0.89
1302
          1       0.31      0.51      0.39
169

avg / total       0.86      0.81      0.83
1471
```

```
             precision    recall  f1-score
support
          0       0.93      0.86      0.89
1621
          1       0.32      0.51      0.40
218
avg / total       0.86      0.82      0.83
1839
```

```
             precision    recall  f1-score
support

          0       0.92      0.84      0.88
643
          1       0.33      0.53      0.40
93

avg / total       0.85      0.80      0.82
736
```

<u>Classification Score for Support Vector Machine: -</u>

When Test-Train Data Ratio is 10:90: -
`0.874`

When Test-Train Data Ratio is 20:80: -
`0.885`

When Test-Train Data Ratio is 25:75: -
`0.881`

<u>Classification Score for Logistic Regression: -</u>

When Test-Train Data Ratio is 10:90: -
`0.802`
When Test-Train Data Ratio is 20:80: -
`0.883`
When Test-Train Data Ratio is 25:75: -
`0.815`

Saheli Saha
8123698628
sahelisaha06@gmail.com

Classification Score for Naïve Bayes: -

When Test-Train Data Ratio is 10:90: -

```
0.803
```

When Test-Train Data Ratio is 20:80: -

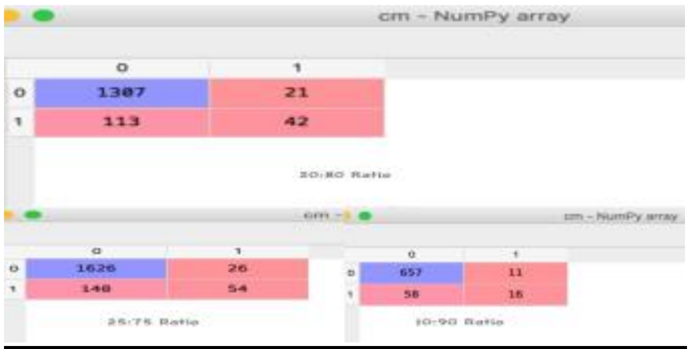
```
0.813
```

When Test-Train Data Ratio is 25:75: -

```
0.815
```

Model Test for the Third Approach (Application of KNN for missing value imputation and Classification Algorithms): -

Confusion Matrix for Support Vector Machine: -



cm – NumPy array

	0	1
0	1307	21
1	113	42

20:80 Ratio

cm – NumPy array

	0	1
0	1626	26
1	148	54

25:75 Ratio

cm – NumPy array

	0	1
0	657	11
1	58	16

10:90 Ratio

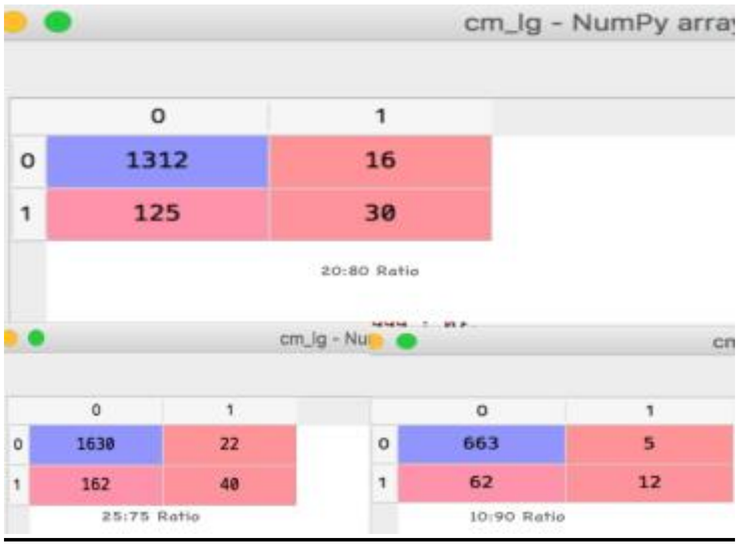
Error rate for Test-Train Data Ratio is 10:90 = $\frac{69}{742} * 100 = 9.29\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{134}{1483} * 100 = 9.03\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{174}{1854} * 100 = 9.38\%$

Saheli Saha
8123698628
sahelisaha06@gmail.com

Confusion Matrix for Logistic Regression: -



	0	1
0	1312	16
1	125	30

20:80 Ratio

	0	1
0	1630	22
1	162	40

25:75 Ratio

	0	1
0	663	5
1	62	12

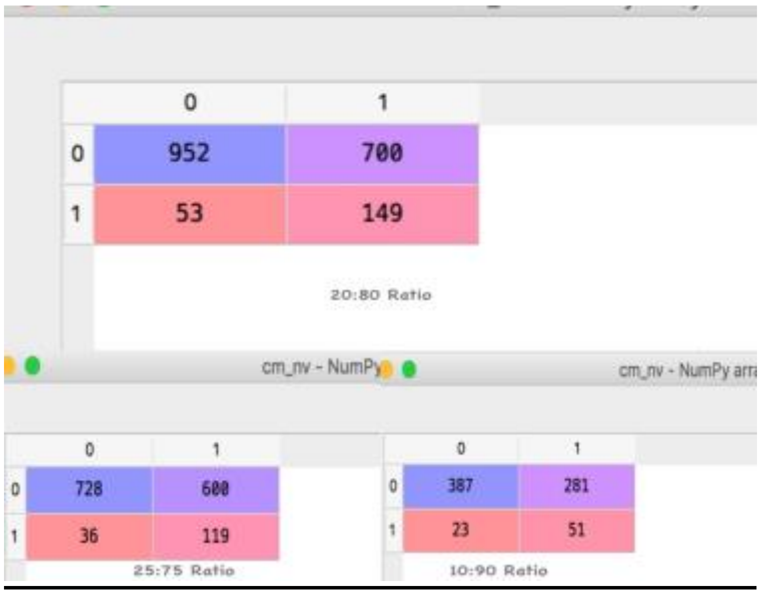
10:90 Ratio

Error rate for Test-Train Data Ratio is 10:90 = $\frac{67}{742} * 100 = 9.03\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{141}{1483} * 100 = 9.50\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{184}{1854} * 100 = 9.92\%$

Confusion Matrix for Naïve Bayes: -



	0	1
0	952	700
1	53	149

20:80 Ratio

	0	1
0	728	600
1	36	119

25:75 Ratio

	0	1
0	387	281
1	23	51

10:90 Ratio

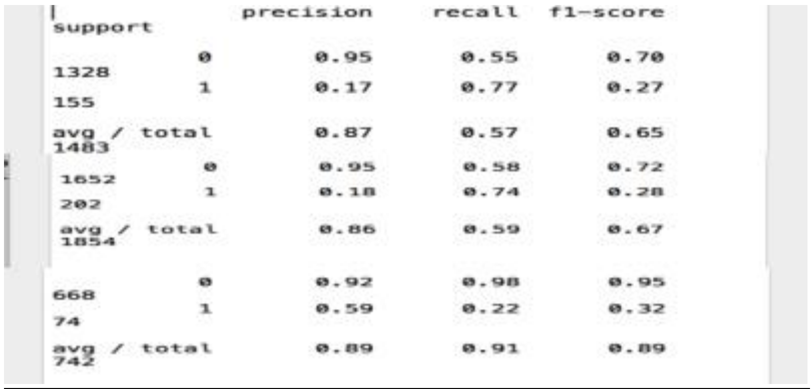
Error rate for Test-Train Data Ratio is 10:90 = $\frac{304}{742} * 100 = 40.97\%$

Saheli Saha
8123698628
sahelisaha06@gmail.com

Error rate for Test-Train Data Ratio is 20:80 = $\frac{753}{1483} * 100 = 50.78\%$

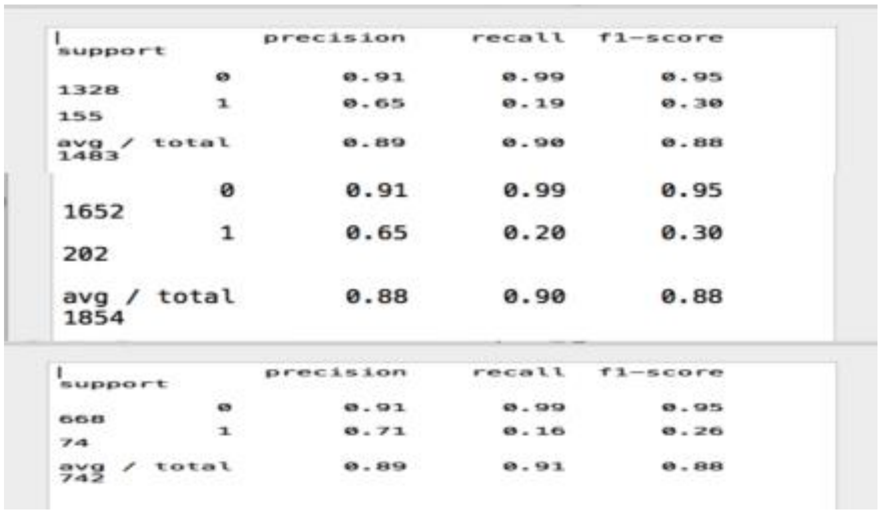
Error rate for Test-Train Data Ratio is 25:75 = $\frac{636}{1854} * 100 = 34.30\%$

Classification Report for Support Vector Machine: -



```
|                precision    recall  f1-score
support
          0       0.95      0.55      0.70
1328
          1       0.17      0.77      0.27
155

avg / total       0.87      0.57      0.65
1483
          0       0.95      0.58      0.72
1652
          1       0.18      0.74      0.28
202

avg / total       0.86      0.59      0.67
1854

          0       0.92      0.98      0.95
668
          1       0.59      0.22      0.32
74

avg / total       0.89      0.91      0.89
742
```

Classification Report for Logistic Regression: -

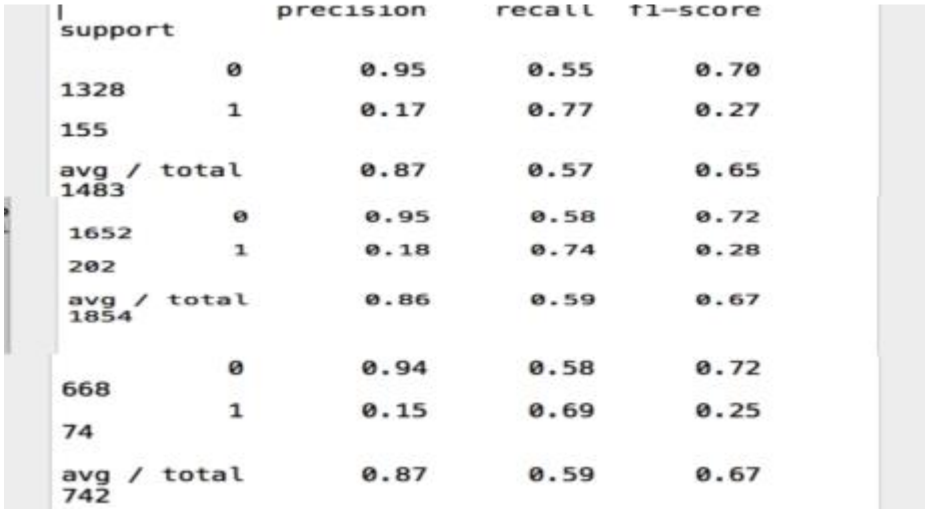


```
|                precision    recall  f1-score
support
          0       0.91      0.99      0.95
1328
          1       0.65      0.19      0.30
155

avg / total       0.89      0.90      0.88
1483
          0       0.91      0.99      0.95
1652
          1       0.65      0.20      0.30
202

avg / total       0.88      0.90      0.88
1854
```

```
|                precision    recall  f1-score
support
          0       0.91      0.99      0.95
668
          1       0.71      0.16      0.26
74

avg / total       0.89      0.91      0.88
742
```

Saheli Saha
8123698628
sahelisaha06@gmail.com

Classification Report for Naïve Bayes: -



```
|              precision    recall  f1-score
support

          0       0.95      0.55      0.70
1328
          1       0.17      0.77      0.27
155

avg / total       0.87      0.57      0.65
1483
          0       0.95      0.58      0.72
 1652
          1       0.18      0.74      0.28
 202

 avg / total      0.86      0.59      0.67
 1854

          0       0.94      0.58      0.72
668
          1       0.15      0.69      0.25
74

avg / total       0.87      0.59      0.67
742
```

Classification Score for Support Vector Machine: -

When Test-Train Data Ratio is 10:90: -
0.907
When Test-Train Data Ratio is 20:80: -
0.910
When Test-Train Data Ratio is 25:75: -
0.906

Classification Score for Logistic Regression: -

When Test-Train Data Ratio is 10:90: -
0.910
When Test-Train Data Ratio is 20:80: -
0.905
When Test-Train Data Ratio is 25:75: -
0.901

Classification Score for Naïve Bayes: -

When Test-Train Data Ratio is 10:90: -
0.590
When Test-Train Data Ratio is 20:80: -
0.571

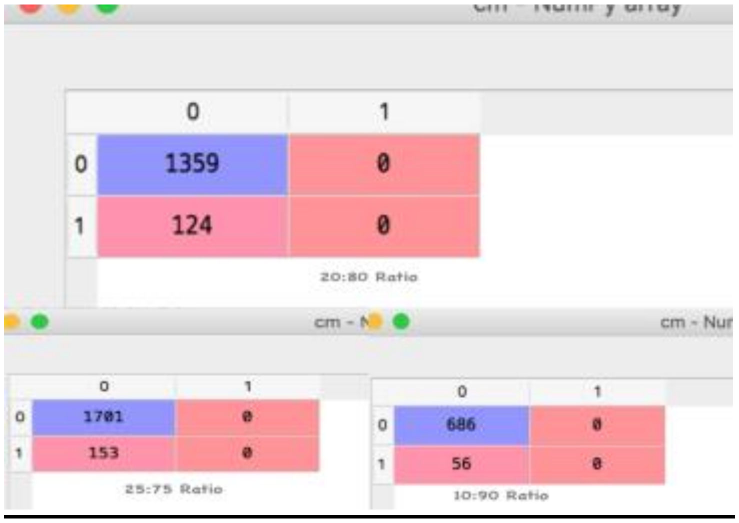
Saheli Saha
8123698628
sahelisaha06@gmail.com

When Test-Train Data Ratio is 25:75: -

```
0.594
```

Model Test for the Fourth Approach (Application of KNN, Dummy Variables, Factor Analysis and Classification Algorithms): -

<u>Confusion Matrix for Support Vector Machine: -</u>



	0	1
0	1359	0
1	124	0

20:80 Ratio

	0	1
0	1701	0
1	153	0

25:75 Ratio

	0	1
0	686	0
1	56	0

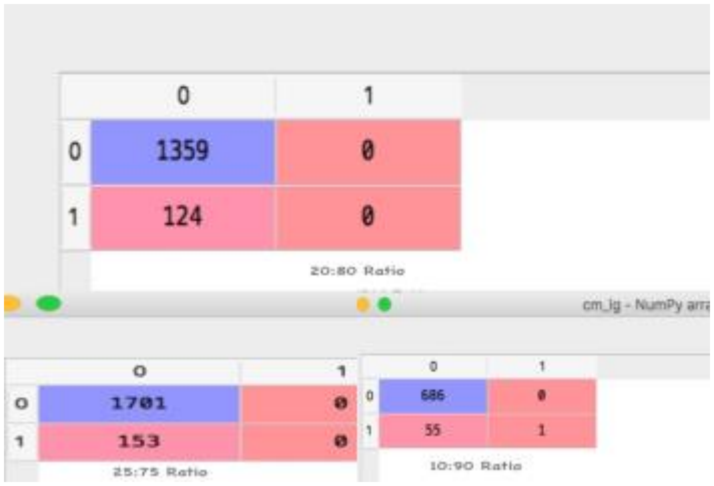
10:90 Ratio

Error rate for Test-Train Data Ratio is 10:90 = $\frac{56}{742} * 100 = 7.55\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{124}{1483} * 100 = 8.36\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{153}{1854} * 100 = 8.25\%$

<u>Confusion Matrix for Logistic Regression: -</u>



	0	1
0	1359	0
1	124	0

20:80 Ratio

	0	1
0	1701	0
1	153	0

25:75 Ratio

	0	1
0	686	0
1	55	1

10:90 Ratio

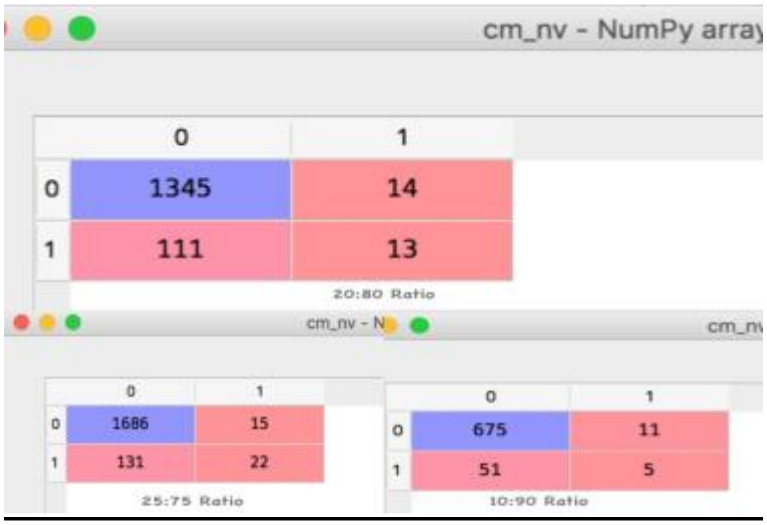
Saheli Saha
8123698628
sahelisaha06@gmail.com

Error rate for Test-Train Data Ratio is 10:90 = $\frac{55}{742} * 100 = 7.41\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{124}{1483} * 100 = 8.36\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{153}{1854} * 100 = 8.25\%$

<u>Confusion Matrix for Naïve Bayes: -</u>


cm_nv - NumPy array

	0	1
0	1345	14
1	111	13

20:80 Ratio

cm_nv - N

	0	1
0	1686	15
1	131	22

25:75 Ratio

cm_nv

	0	1
0	675	11
1	51	5

10:90 Ratio

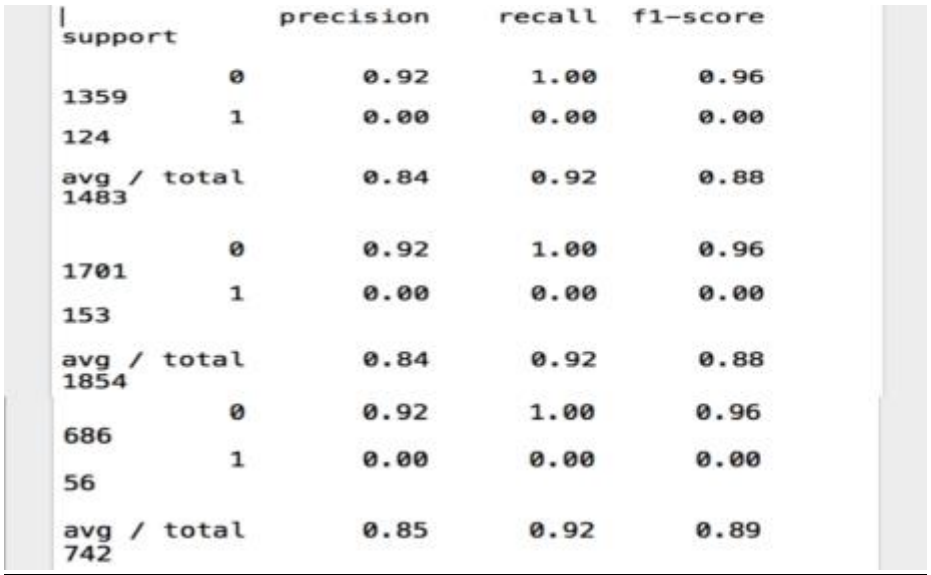
Error rate for Test-Train Data Ratio is 10:90 = $\frac{62}{742} * 100 = 8.36\%$

Error rate for Test-Train Data Ratio is 20:80 = $\frac{125}{1483} * 100 = 8.42\%$

Error rate for Test-Train Data Ratio is 25:75 = $\frac{146}{1854} * 100 = 7.87\%$

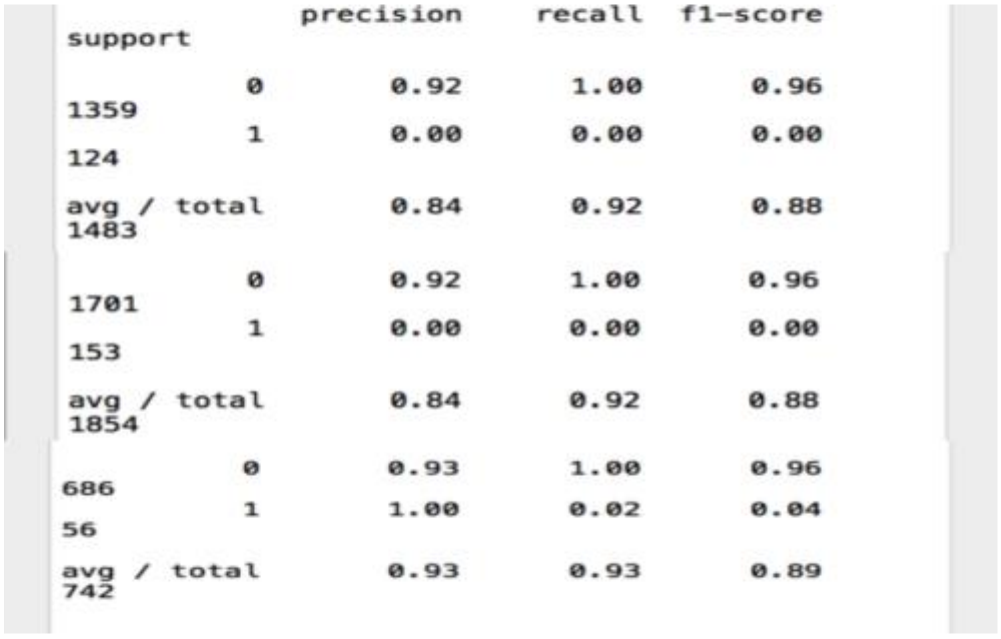
Saheli Saha
8123698628
sahelisaha06@gmail.com

Classification Report for Support Vector Machine: -



```
|             precision    recall  f1-score
support
          0        0.92      1.00      0.96
1359
          1        0.00      0.00      0.00
124

avg / total        0.84      0.92      0.88
1483

          0        0.92      1.00      0.96
1701
          1        0.00      0.00      0.00
153

avg / total        0.84      0.92      0.88
1854
          0        0.92      1.00      0.96
686
          1        0.00      0.00      0.00
56

avg / total        0.85      0.92      0.89
742
```

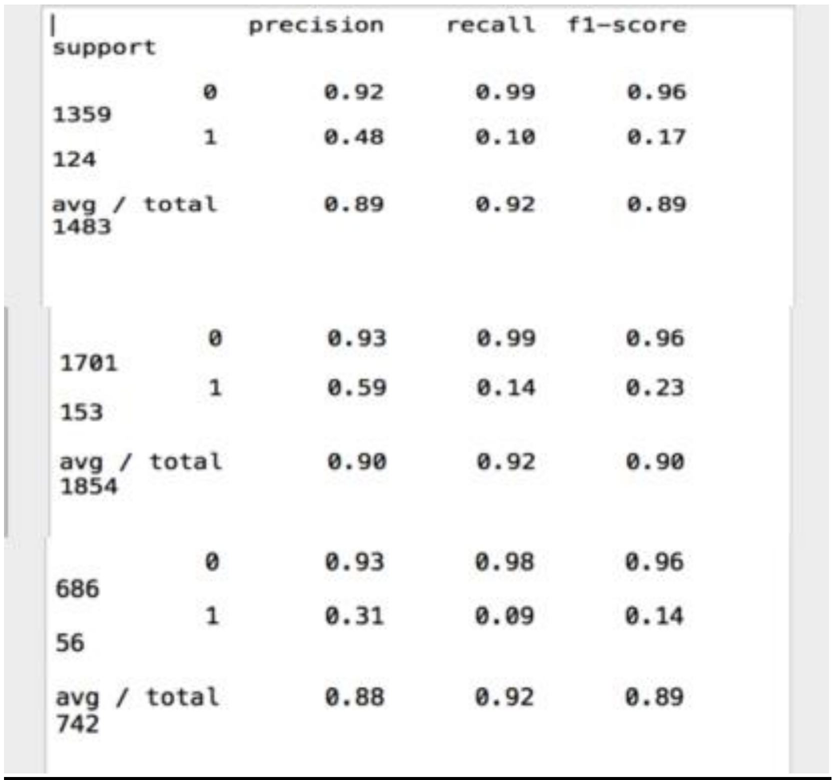
Classification Report for Logistic Regression: -



```
              precision    recall  f1-score
support
          0        0.92      1.00      0.96
1359
          1        0.00      0.00      0.00
124

avg / total        0.84      0.92      0.88
1483

          0        0.92      1.00      0.96
1701
          1        0.00      0.00      0.00
153

avg / total        0.84      0.92      0.88
1854
          0        0.93      1.00      0.96
686
          1        1.00      0.02      0.04
56

avg / total        0.93      0.93      0.89
742
```

Saheli Saha
8123698628
sahelisaha06@gmail.com

Classification Report for Naïve Bayes: -



```
|             precision    recall  f1-score
support

          0       0.92      0.99      0.96
1359
          1       0.48      0.10      0.17
124

avg / total       0.89      0.92      0.89
1483


          0       0.93      0.99      0.96
1701
          1       0.59      0.14      0.23
153

avg / total       0.90      0.92      0.90
1854

          0       0.93      0.98      0.96
686
          1       0.31      0.09      0.14
56

avg / total       0.88      0.92      0.89
742
```

Classification Score for Support Vector Machine: -

When Test-Train Data Ratio is 10:90: -
0.925
When Test-Train Data Ratio is 20:80: -
0.916
When Test-Train Data Ratio is 25:75: -
0.917

Classification Score for Logistic Regression: -

When Test-Train Data Ratio is 10:90: -
0.926
When Test-Train Data Ratio is 20:80: -
0.916
When Test-Train Data Ratio is 25:75: -
0.917

Saheli Saha
8123698628
sahelisaha06@gmail.com

Classification Score for Naïve Bayes: -

When Test-Train Data Ratio is 10:90: -
0.916

When Test-Train Data Ratio is 20:80: -
0.916
When Test-Train Data Ratio is 25:75: -
0.921

Conclusion: -

Comparing the results, we can see in general Naïve Bayes is not a good classifier for the dataset, error rate was very high. For 1st and 2nd approach it is 40% and 18 % in average. Logistic Regression and Support Vector Machine performed almost equally well. In my first and second approach error rate was around 11% which is quite good. But that did not stop me from experimenting more. So, I tried with dummy variables. There I got quite a good improvement of my model. So, replacing missing data with numerical values are not a good idea for this dataset.
For my first and second approach average error rate was around 11% where average accuracy level is 90%.

Then I applied KNN algorithm for missing value imputation. In 3rd approach I applied it and then replaced categorical values with numerical values and in 4th approach with KNN I applied dummy variables for handling categorical data. I got better accuracy in my 4th approach. So replacing data with numerical values without using dummy variable is not preferable here. For 3rd approach average error rate was 10% and average accuracy level is 90%, where in 5th approach it is 8% and 91% respectively. Which is better than my 1st and 2nd approach.

Finally, in my model error rate is around 5% to be precise it is 4.6% and accuracy level is 96%, Which is far better than my other approaches. That is why I choose this model over other models.