FLIP ROBO

PROJECT REPORT
ON
RATINGS PREDICTION

Submitted by

MANISHA

Introduction

Business Problem Framing:

The client who has a website where people write different reviews for technical products. Now they are adding a new feature to their website i.e. The reviewer will have to add stars(rating) as well with the review. The rating is out 5 stars and it only has 5 options available 1 star, 2 stars, 3 stars, 4 stars, 5 stars.

Conceptual Background of the Domain Problem:

Here we have no data, so first we scrap the data by Selenium technique and form the dataset for further analysis on it. And then fit a model to check which one is the better.

Review of Literature:

The dataset have 21738 rows and 3 columns. The ratings are 1, 2, 3, 4 and 5 for the product but we merge 3 and 4 as zero and 5 consider as 1 in the data. The difference between 0 and 1 is not less than 50%, se do not consider it as imbalanced .

Motivation for the Problem Undertaken:

We want to predict ratings for the reviews which were written in the past and they don’t have a rating. So, we have to build an application which can predict the rating by seeing the review.

Analytical Problem Framing

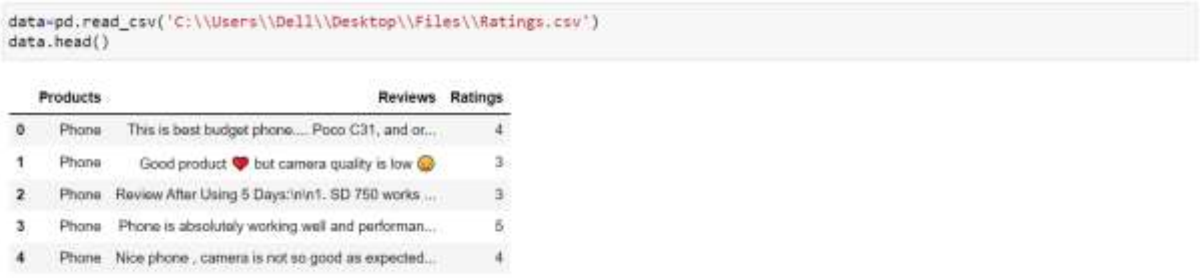
Mathematical/ Analytical Modeling of the Problem:

We use Statistical techniques and analytics modeling in our projects, such as:

- describe() : use to calculate the statistical values that are mean, standard deviation, quantile deviation, minimum and maximum values.
- corr(): use to calculate the relation between feature variable with the target variable.

Data Source and their formats:

The data set of the ecommerce rating as show in the fig:



```
data=pd.read_csv('C:\\Users\\Dell\\Desktop\\Files\\Ratings.csv')
data.head()
```

	Products	Reviews	Ratings
0	Phone	This is best budget phone.... Poco C31, and or...	4
1	Phone	Good product ❤ but camera quality is low 😞	3
2	Phone	Review After Using 5 Days:\n\n1. SD 750 works ...	3
3	Phone	Phone is absolutely working well and performan...	5
4	Phone	Nice phone , camera is not so good as expected...	4

There are 21738 rows and 3 columns. ‘Ratings’ column is our target variable and others are feature variable. There is one column of numerical data and two columns of object type.

Data Pre-processing

There are no null values in the dataset. Apply data processing technique on Reviews using NLP technique. Convert object data into numerical data. There are more object data so we do not consider StandardScaler() in the dataset.

Data inputs-Logic-Output Relationships

The Ratings column is our target variable. The correlation of product and reviews are 15% and 7% respectively with the target variable.

Hardware and Software Requirements and Tools used

Hardware:

- Memory 16GB minimum
- Hard Drive SSD is preferred 500GB
- Processor intel i5 minimum
- Operating system Windows 10

Software:

- Jupyter notebook (Python)

Libraries:

pandas	(used to create the data and read the data)
numpy	(used with the mathematical function)
seaborn	(used to create a different types of graphs)
matplotlib	(used to plot the graph)
accuracy_score	(used to calculate accuracy score for train and test)
classification_report	(to display accuracy, precision, f1 scores)
confusion_matrix	(form the matrix)
roc_curve	(use to draw a graph)
roc_auc_score	(calculate the area under the curve)

Model/s Development and Evaluation

Identification of possible problem:

We approach to both statistical and analytical problem

- Plot a bar graph for nominal data and distribution graph for continuous data
- describe () use to calculate mean, standard deviation, minimum, maximum and quantile deviation
- corr() used to calculate the correlation of input variable with the output variable.
-

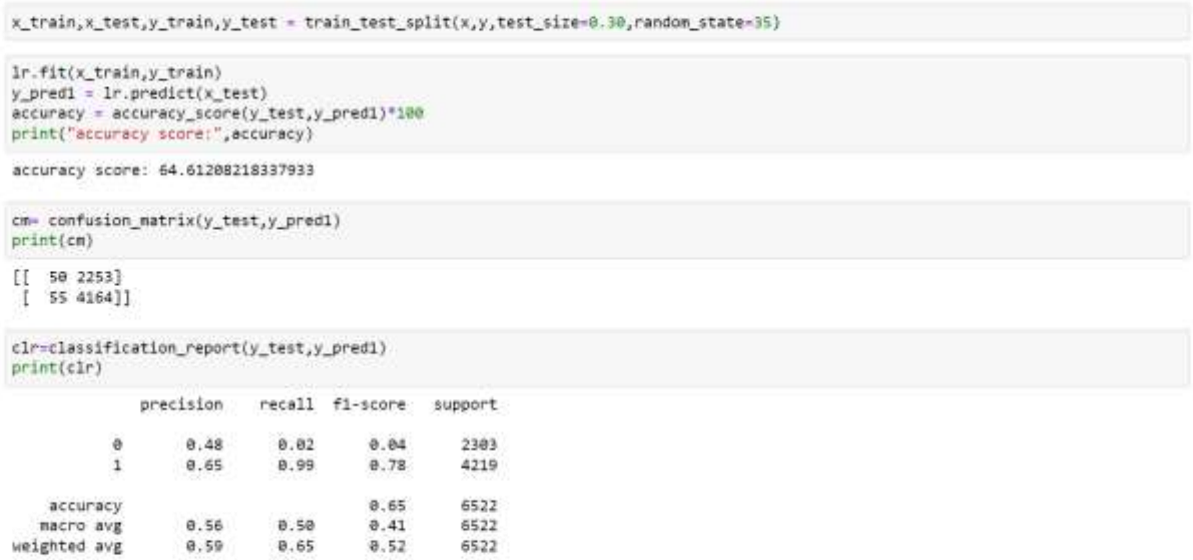
Testing of Identified Approaches:

Here we work on the classification problem so the machine learning models are:

- Logistic Regression
- K Neighbors Classification
- Random Forest Classification
- Decision Tree Classification

Run and Evaluate selected models:

- Logistic Regression



```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.30,random_state=35)
```

```
lr.fit(x_train,y_train)
y_pred1 = lr.predict(x_test)
accuracy = accuracy_score(y_test,y_pred1)*100
print("accuracy score:",accuracy)
```

```
accuracy score: 64.61208218337933
```

```
cm= confusion_matrix(y_test,y_pred1)
print(cm)
```

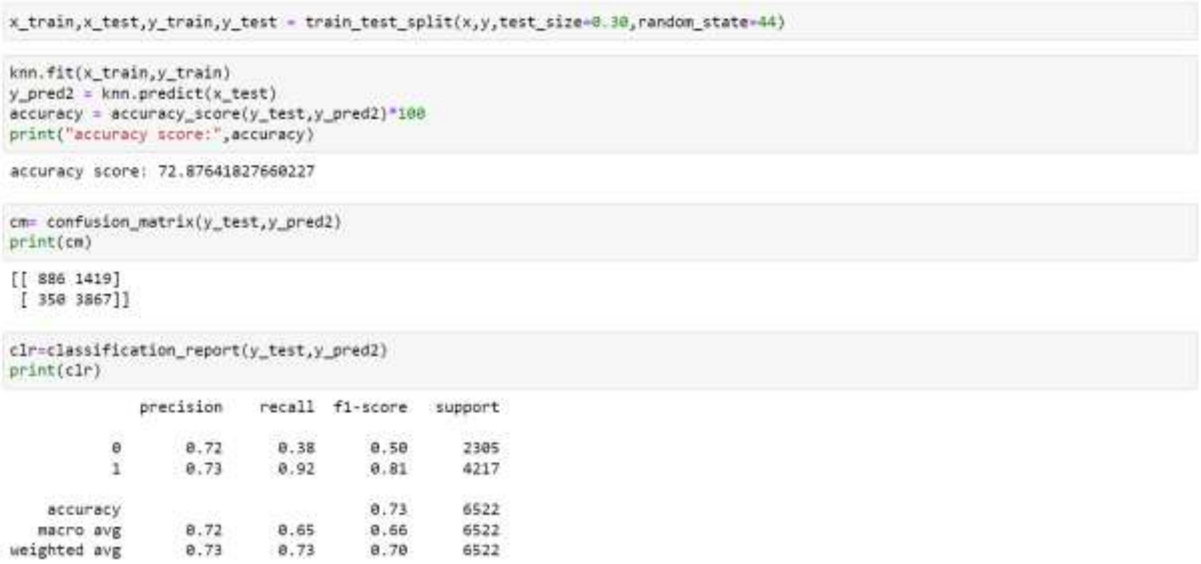
```
[[  50 2253]
 [  55 4164]]
```

```
clr=classification_report(y_test,y_pred1)
print(clr)
```

```
              precision    recall  f1-score   support

           0       0.48      0.02      0.04      2303
           1       0.65      0.99      0.78      4219

    accuracy                           0.65      6522
   macro avg       0.56      0.50      0.41      6522
weighted avg       0.59      0.65      0.52      6522
```

The accuracy score of logistic regression is 65%. And the precision is 65, recall is 99 and f1-score is 78.

- K Neighbors Classification



```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.30,random_state=44)
```

```
knn.fit(x_train,y_train)
y_pred2 = knn.predict(x_test)
accuracy = accuracy_score(y_test,y_pred2)*100
print("accuracy score:",accuracy)
```

```
accuracy score: 72.87641827660227
```

```
cm= confusion_matrix(y_test,y_pred2)
print(cm)
```

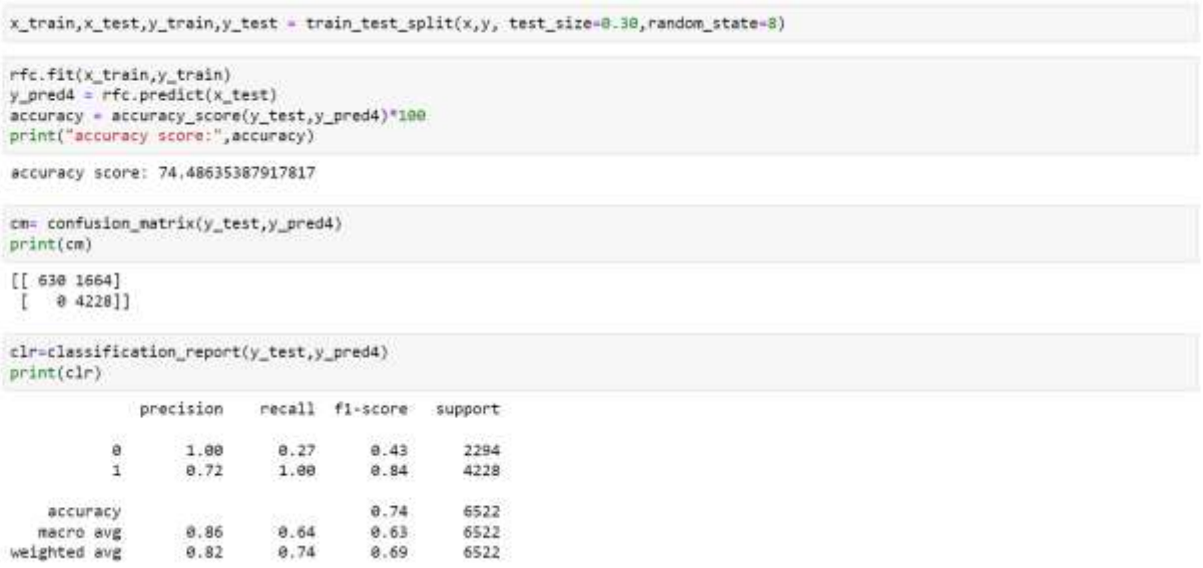
```
[[ 886 1419]
 [ 350 3867]]
```

```
clr=classification_report(y_test,y_pred2)
print(clr)
```

```
              precision    recall  f1-score   support

           0       0.72      0.38      0.50      2305
           1       0.73      0.92      0.81      4217

    accuracy                           0.73      6522
   macro avg       0.72      0.65      0.66      6522
weighted avg       0.73      0.73      0.70      6522
```

The accuracy score of K Neighbors classification is 73%. And the precision is 73, recall is 92 and f1-score is 81.

- Random Forest Classification



```
x_train,x_test,y_train,y_test = train_test_split(x,y, test_size=0.30,random_state=8)
```

```
rfc.fit(x_train,y_train)
y_pred4 = rfc.predict(x_test)
accuracy = accuracy_score(y_test,y_pred4)*100
print("accuracy score:",accuracy)
```

```
accuracy score: 74.48635387917817
```

```
cm= confusion_matrix(y_test,y_pred4)
print(cm)
```

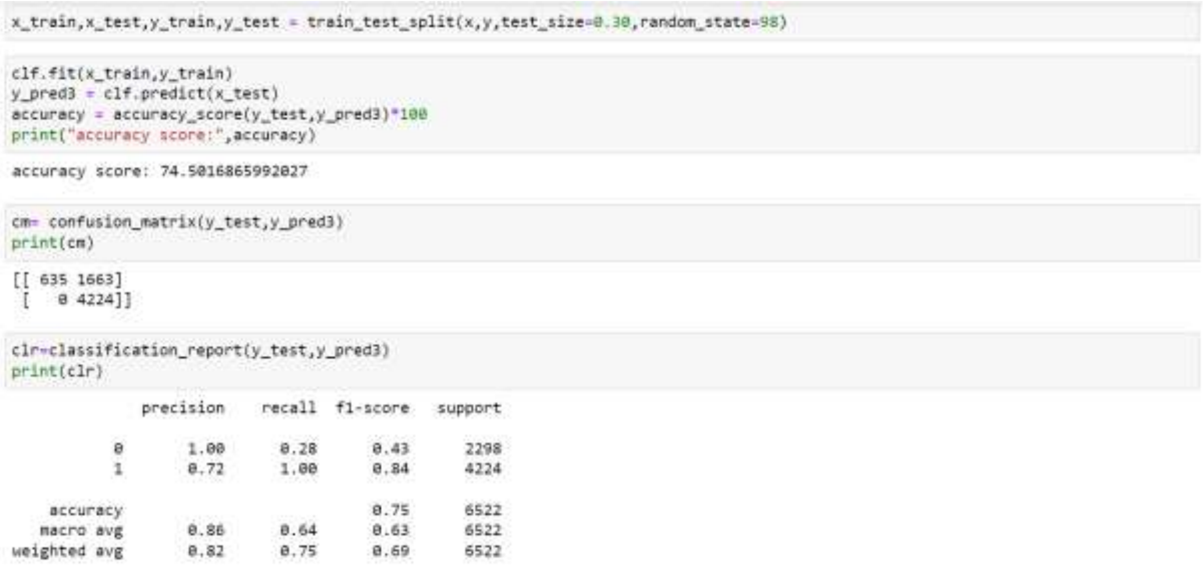
```
[[ 630 1664]
 [   0 4228]]
```

```
clr=classification_report(y_test,y_pred4)
print(clr)
```

```
              precision    recall  f1-score   support

           0       1.00      0.27      0.43      2294
           1       0.72      1.00      0.84      4228

    accuracy                           0.74      6522
   macro avg       0.86      0.64      0.63      6522
weighted avg       0.82      0.74      0.69      6522
```

The accuracy score of Random Forest classification is 74. And the precision is 72, recall is 100 and f1-score is 84.

- Decision Tree Classification



```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.30,random_state=98)
```

```
clf.fit(x_train,y_train)
y_pred3 = clf.predict(x_test)
accuracy = accuracy_score(y_test,y_pred3)*100
print("accuracy score:",accuracy)
```

```
accuracy score: 74.5016865992027
```

```
cm= confusion_matrix(y_test,y_pred3)
print(cm)
```

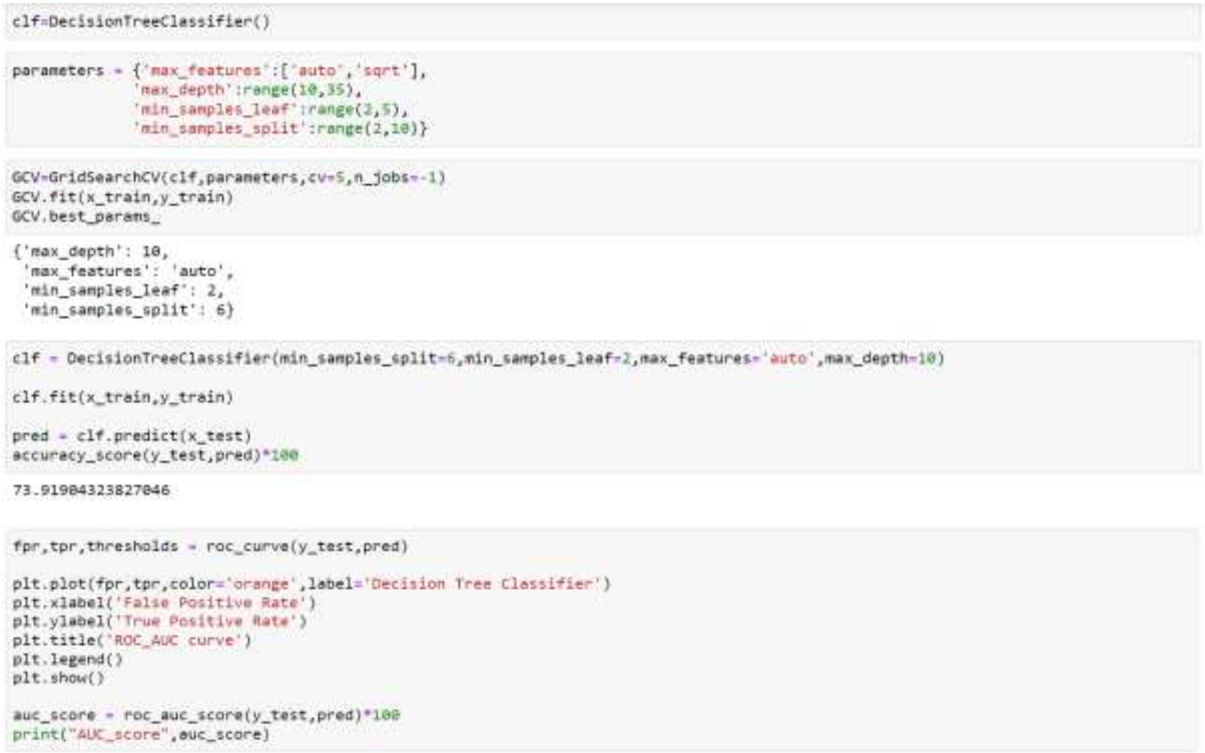
```
[[ 635 1663]
 [   0 4224]]
```

```
clr=classification_report(y_test,y_pred3)
print(clr)
```

```
              precision    recall  f1-score   support

           0       1.00      0.28      0.43      2298
           1       0.72      1.00      0.84      4224

    accuracy                           0.75      6522
   macro avg       0.86      0.64      0.63      6522
weighted avg       0.82      0.75      0.69      6522
```

The accuracy score of Decision Tree classification is 75%. And the precision is 72, recall is 100 and f1 score is 84.

This model give better score and total of True Negative and False Negative in the confusion matrix is also less in the same model. So, we tune the parameters of Decision Tree and try to get better score for it.



```
clf=DecisionTreeClassifier()
```

```
parameters = {'max_features':['auto','sqrt'],
              'max_depth':range(10,35),
              'min_samples_leaf':range(2,5),
              'min_samples_split':range(2,10)}
```

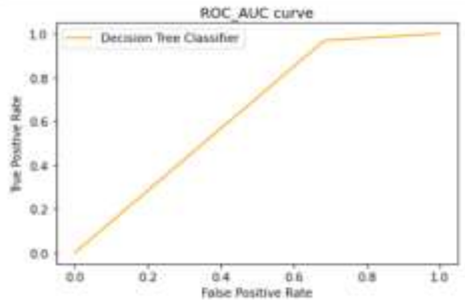
```
GCV=GridSearchCV(clf,parameters,cv=5,n_jobs=-1)
GCV.fit(x_train,y_train)
GCV.best_params_
```

```
{'max_depth': 10,
 'max_features': 'auto',
 'min_samples_leaf': 2,
 'min_samples_split': 6}
```

```
clf = DecisionTreeClassifier(min_samples_split=6,min_samples_leaf=2,max_features='auto',max_depth=10)

clf.fit(x_train,y_train)

pred = clf.predict(x_test)
accuracy_score(y_test,pred)*100
```

```
73.91984323827046
```

```
fpr,tpr,thresholds = roc_curve(y_test,pred)

plt.plot(fpr,tpr,color='orange',label='Decision Tree Classifier')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC_AUC curve')
plt.legend()
plt.show()

auc_score = roc_auc_score(y_test,pred)*100
print("AUC_score",auc_score)
```



```
AUC_score 64.24107065529839
```

The accuracy is not increase after tuning so, the score will be 74% and the AUC score is 64%.

Visualisation:

On visualising the data we see that there are types of ecommerce product in the dataset. There are three types of ratings such as 3,4 and 5 this means that the product is not much cheap.

Interpretation of the Results:

On our analysis basis we go through various models and then we conclude better model on the basis of accuracy score, precision and f1-score.Then use parameter tuning to improve the score but it is not happen means that is the actual score. And then we will predict the test data.

Conclusion

Key Findings and Conclusions of the Study:

On study product data set we train the data first and then test to predict the target values. The zero values are combined of 3 and 4 ratings whereas 5 means one.

Learning Outcomes of the Study in respect of Data Science:

Here we apply data processing technique on Reviews using NLP technique. Visualise the categorical data and we see that different products have different-different reviews on their point of views. In analysis we do describe the statistical values and correlation. Fit some classification models and find the better one i.e. Decision Tree Model. Calculate confusion metrics and classification report.

Limitations of this work and Scope for Future Work:

Limitations:

- Chances of customer dissatisfaction.
- Sampling errors can exclude your most valuable customers

Scope:

- Measure the customer’s satisfaction on the products.
- Improve customer retention and loyalty rates.
- Reduce cost and constraints.