Capstone Project – 3
Email Campaign Effectiveness Prediction

SUPERVISED ML-CLASSIFICATION ALGORITHM

Submitted BY:

Name: Rahul Kumar Gupta

Content

- Problem statement
- Data Summary
- Data Cleaning
- Imputing missing values
- Analysis of categorical features
- Analysis of Continuous features
- Outlier treatment
- Feature Engineering
- Understanding target variable
- Handling Imbalanced data.
- Different Models
- Challenges
- Conclusion

Problem Statement

Most of the small to medium business owners are making effective use of Gmail-based Email marketing Strategies for offline targeting of converting their prospective customers into leads so that they stay with them in Business.

The main objective is to create a machine learning model to characterize the mail and track the mail that is ignored; read; acknowledged by the reader.

Data Summary

- The dataset comprised of 12 features including the target variable **Email_Status**.
- The **5 numerical variables** were :

  Word_Count

  Total_Past_Communications

  Subject_Hotness_Score

  Total_Links

  Total_Images
- The **5 categorical variables** were:

  Email_Type

  Email_Source_Type

  Customer_Location

  Email_Campaign_Type

  Time_Email_Sent_Catergory
- The total no. of records in our dataset is 68353

Data Cleaning

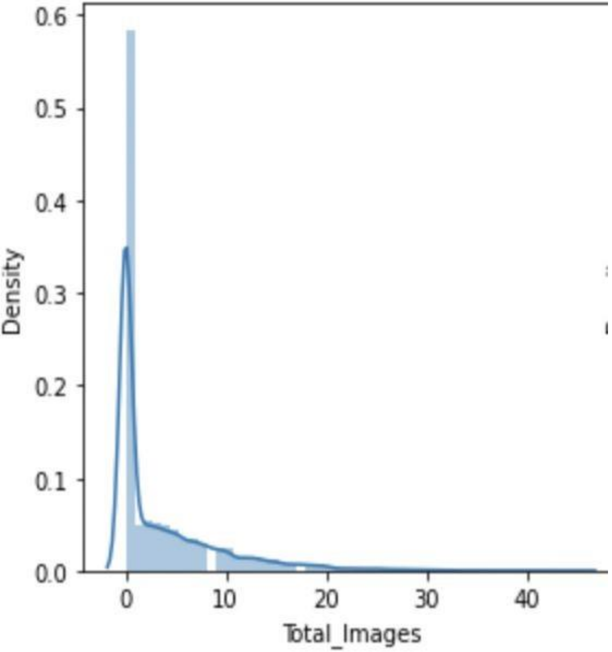
1. Null Value Imputation:

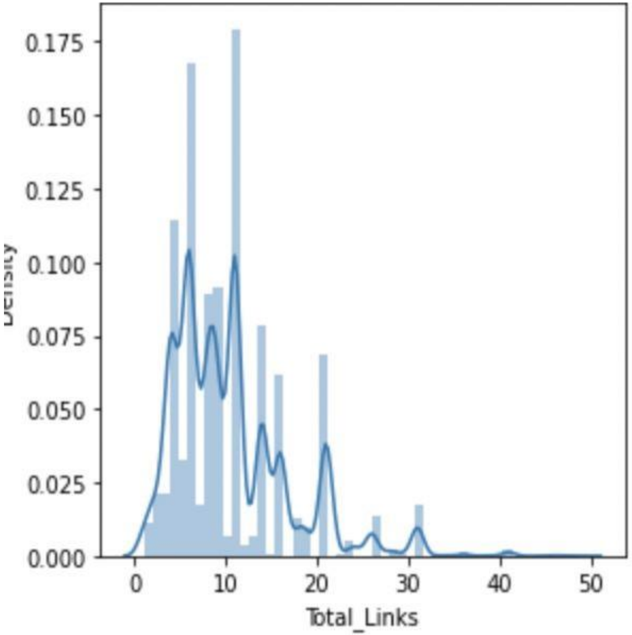
```
Email_ID                        0
Email_Type                      0
Subject_Hotness_Score           0
Email_Source_Type               0
Customer_Location           11595
Email_Campaign_Type             0
Total_Past_Communications    6825
Time_Email_sent_Category        0
Word_Count                      0
Total_Links                  2201
Total_Images                 1677
Email_Status                    0
dtype: int64
```

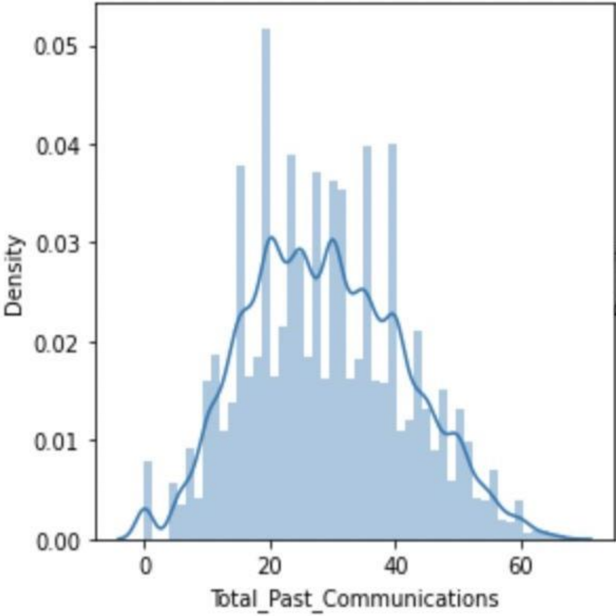
Here we saw see clearly there are total 4 features which is having a null value , so we will try to fill by analyzing it

Imputing missing values

- Impute the missing values for Total_Past_Communication by the mean
- Impute the missing values for Total_Links & Total_Images by the mode



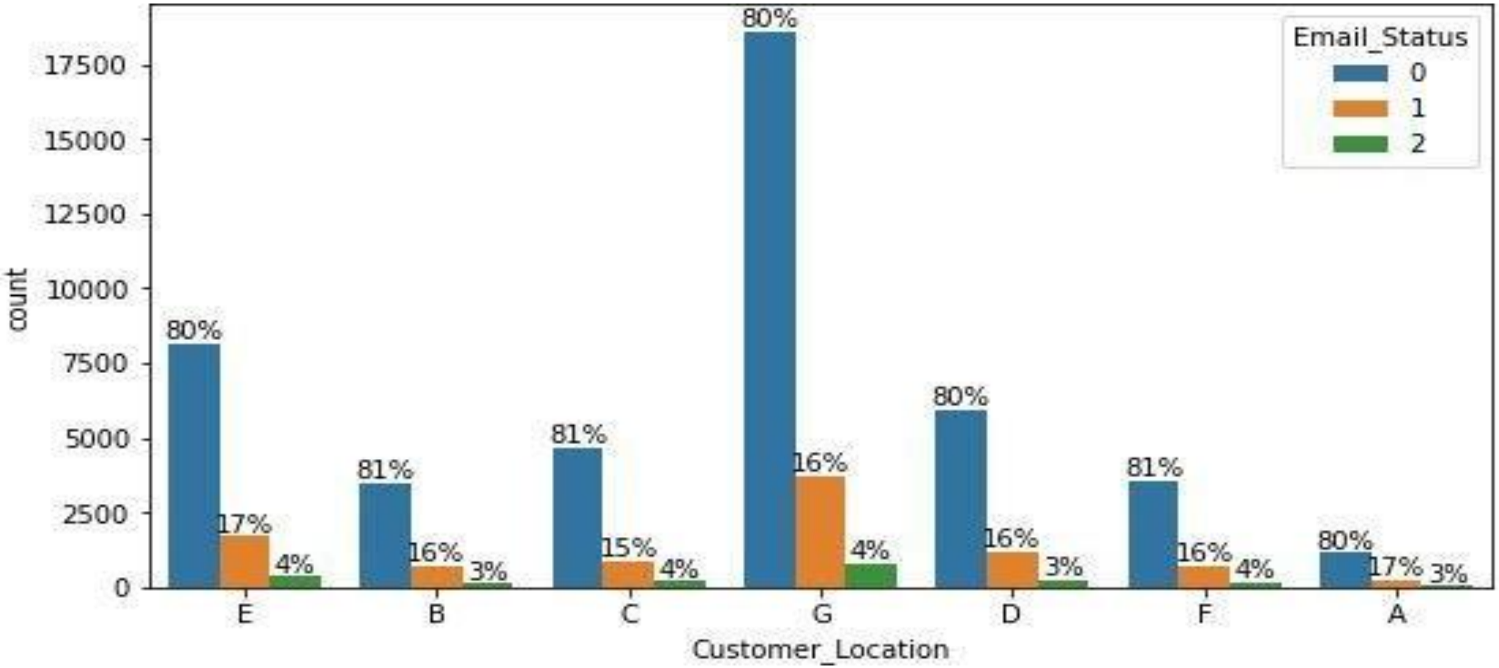




Analysis of Categorical features

- **Customer_Location w.r.t Email_Status**

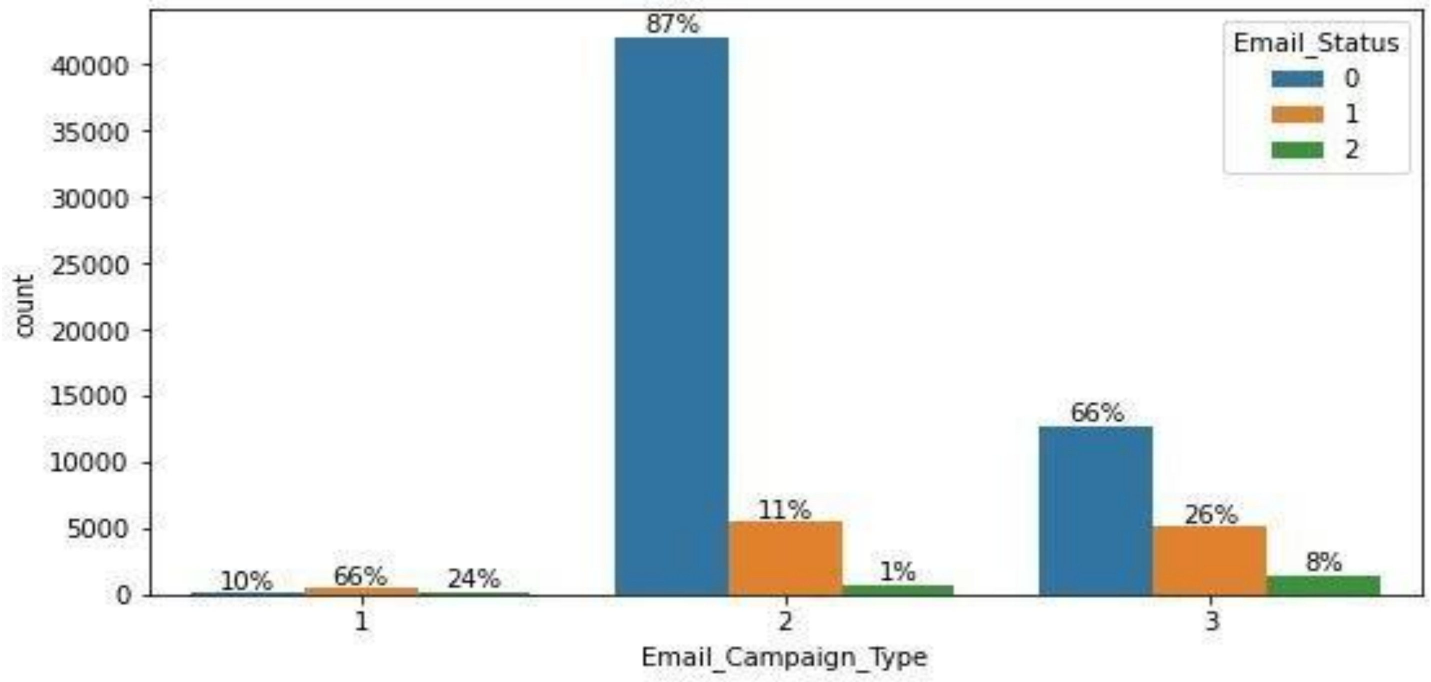
  Inference: same ratio of Email_Status for different demographics



Analysis of Categorical features

- **Email_Campaign_Type w.r.t. Email_Status**

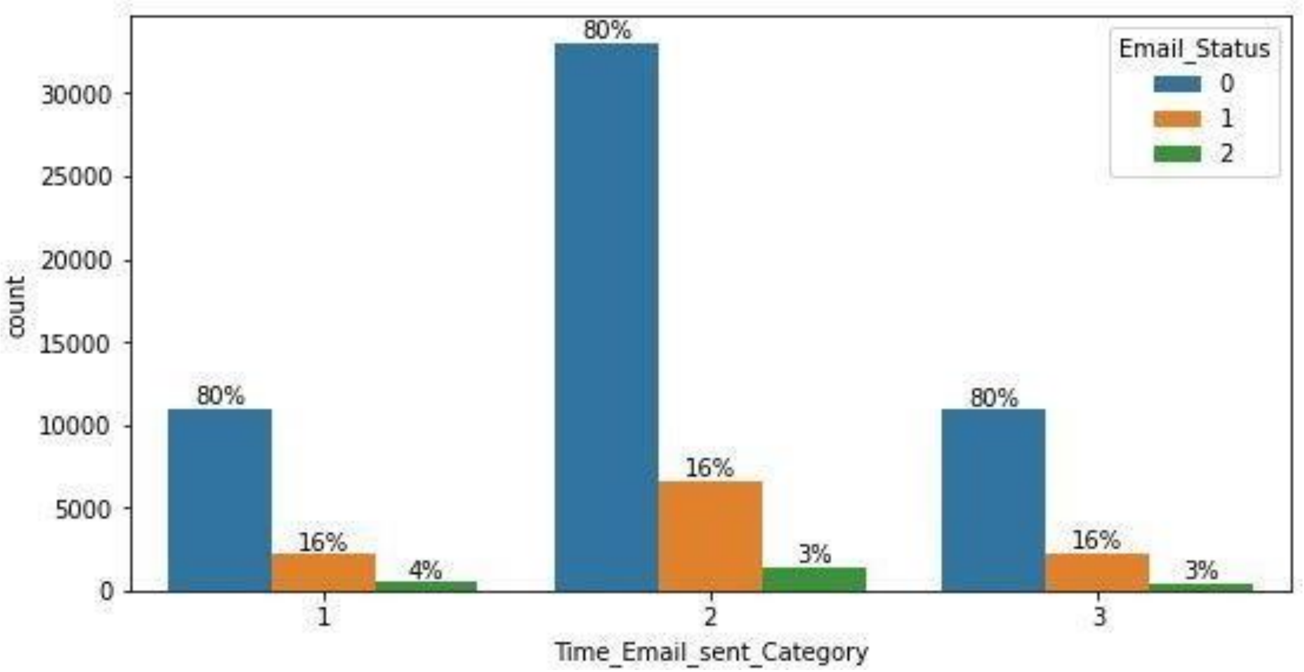
  90% of the time Email gets read or acknowledged if Campaign_Type is 1



Analysis of Categorical features

- **Time_Email_Sent_Catagory**

  Time Email Sent has no influence over Email_Status



OBSERVATION FROM CATAGORICAL VARIABLE AND TARGET EMAIL STATUS

The email type 1 which may be considered as promotional emails are sent more than email type 2 and hence are read and acknowledged more than the other type otherwise the proportion of ignored, read, acknowledged emails are kind of same in both email types. Email source type shows kind of a similar pattern for both the categories.

In the customer location feature we can find that irrespective of the location, the percentage ratio of emails being ignored, read and acknowledge are kind of similar. It does not exclusively influence our target variable. It would be better to not consider location as factor in people ignoring, reading or acknowledging our emails. Other factors should be responsible in why people are ignoring the emails not location.

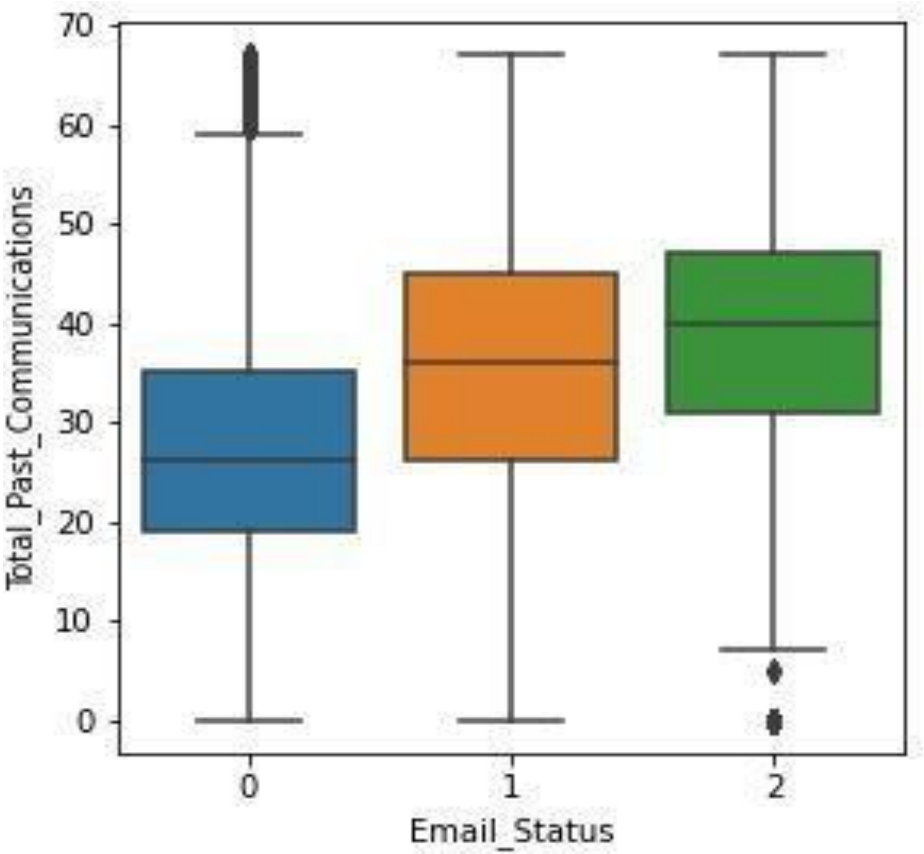
In the Email Campaign Type feature, it seems like in campaign type 1 very few emails were sent but has a very high likelihood of getting read. Most emails were sent under email campaign type 2 and most ignored. Seems like campaign 3 was a success as even when less number of emails were sent under campaign 3, more emails were read and acknowledged.

If we consider 1 annd 3 as morning and night category in time email sent feature, it is obvious to think 2 as middle of the day and as expected there were more emails sent under 2nd category than either of the others, sending emails in the middle of the day could lead to reading and opening the email as people are generally working at that time and they frequently checkup their emails, but it cannot be considered as the major factor in leading to acknowledged emails.

Analysis of Continuous features

- **Total_Past_Communications**

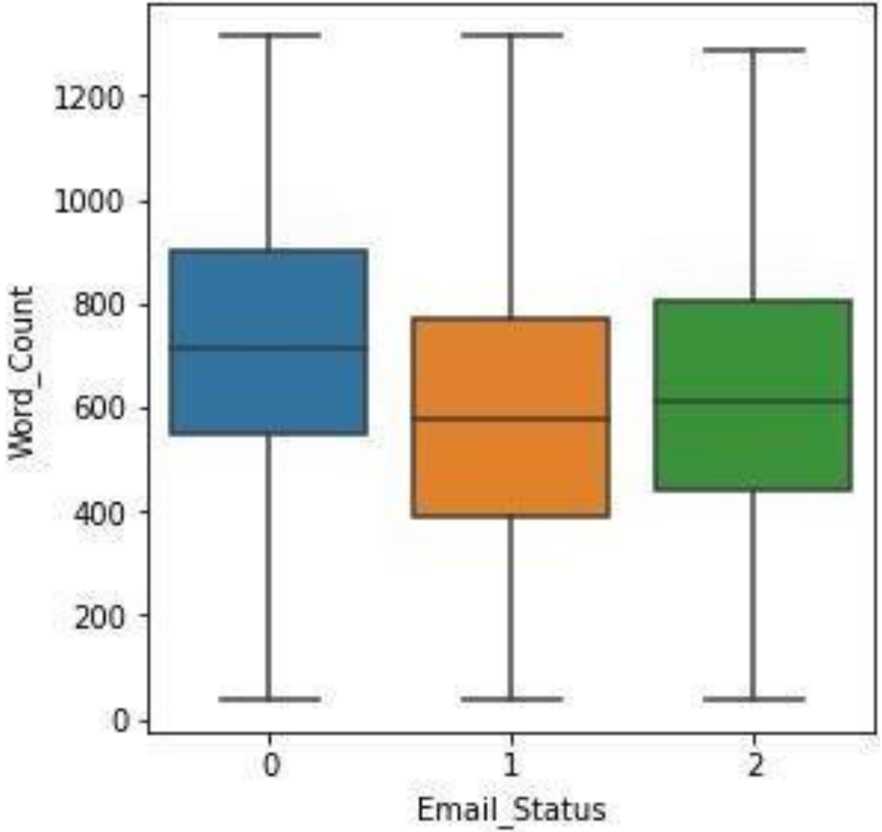
  As no. of past communication is increasing, Email is less ignored.



Analysis of Continuous features

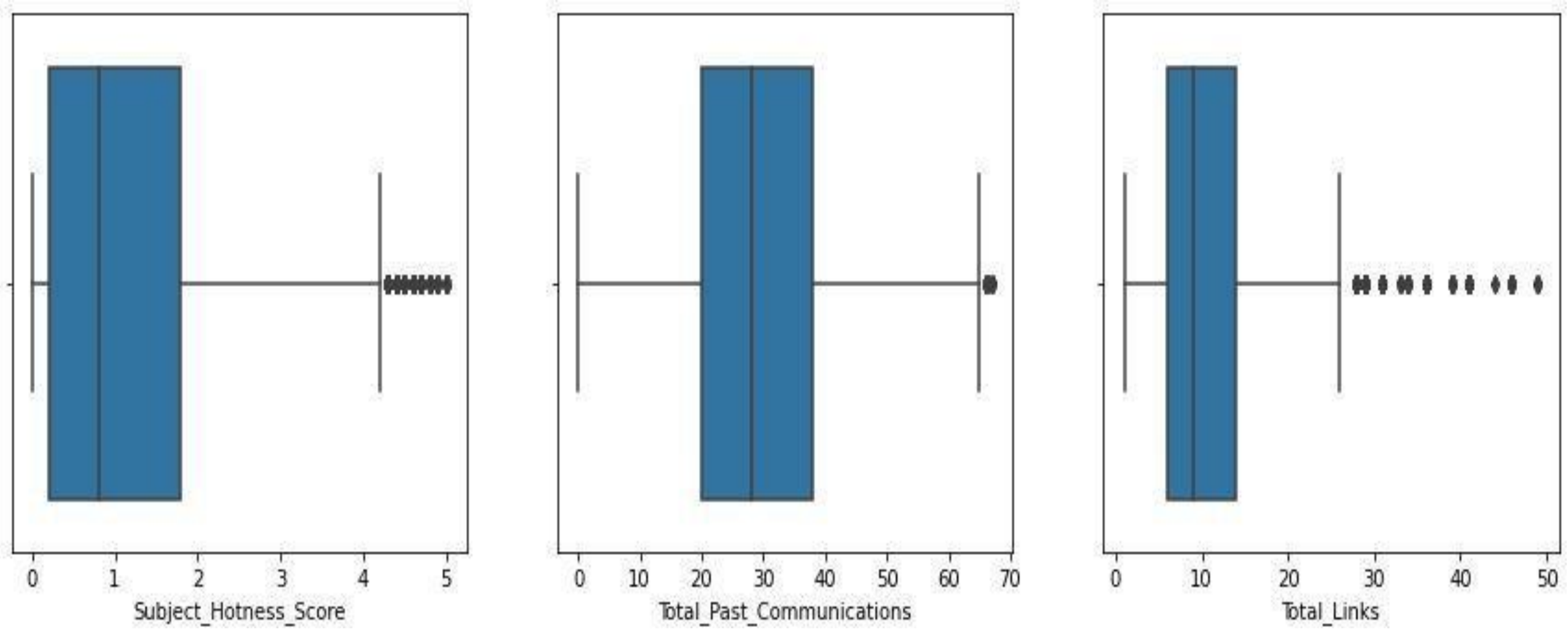
- **Word_Count**

  No one is interested in reading Emails that are too long!!

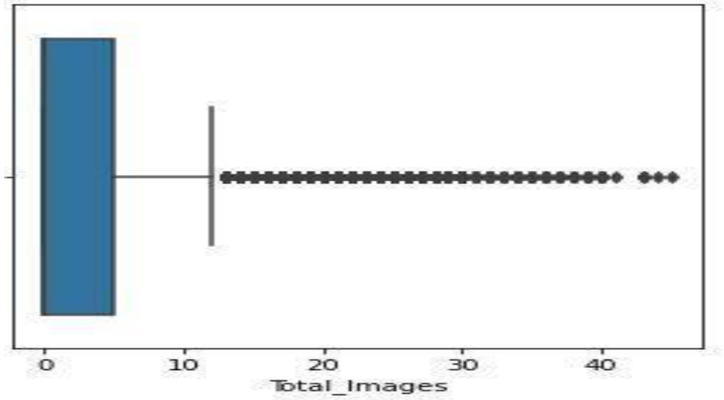


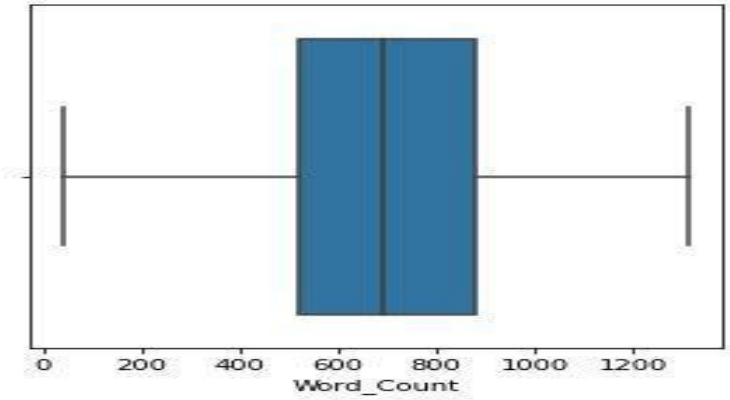
Analysis of Continuous features

- **Outliers in different continuous features**



- **Outliers in different continuous features**

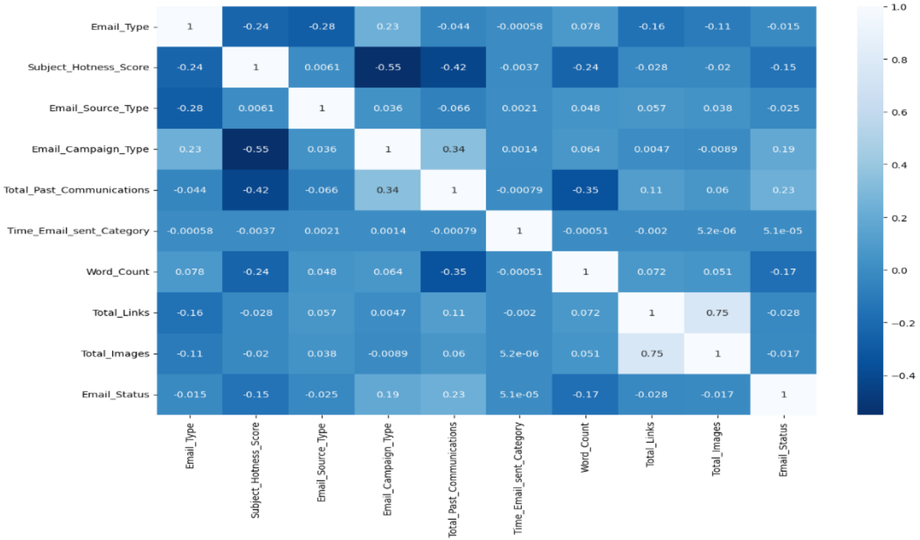


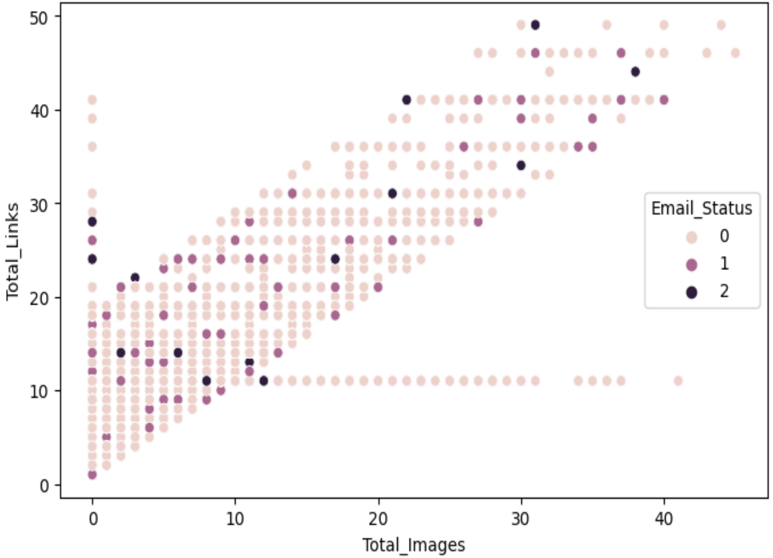


We have more than 5% outliers in minority section and hence to avoid lack of information, we decide against deleting them.

Feature Engineering

1. Combining Total_Images and Total_Links:





High **positive correlation** observed and hence **Links_Images =Total_Images + Total_Links=(0.75)**

Feature Engineering

2. Multicollinearity Check:

- Multicollinearity checked using **VIF Factor**

Why ?

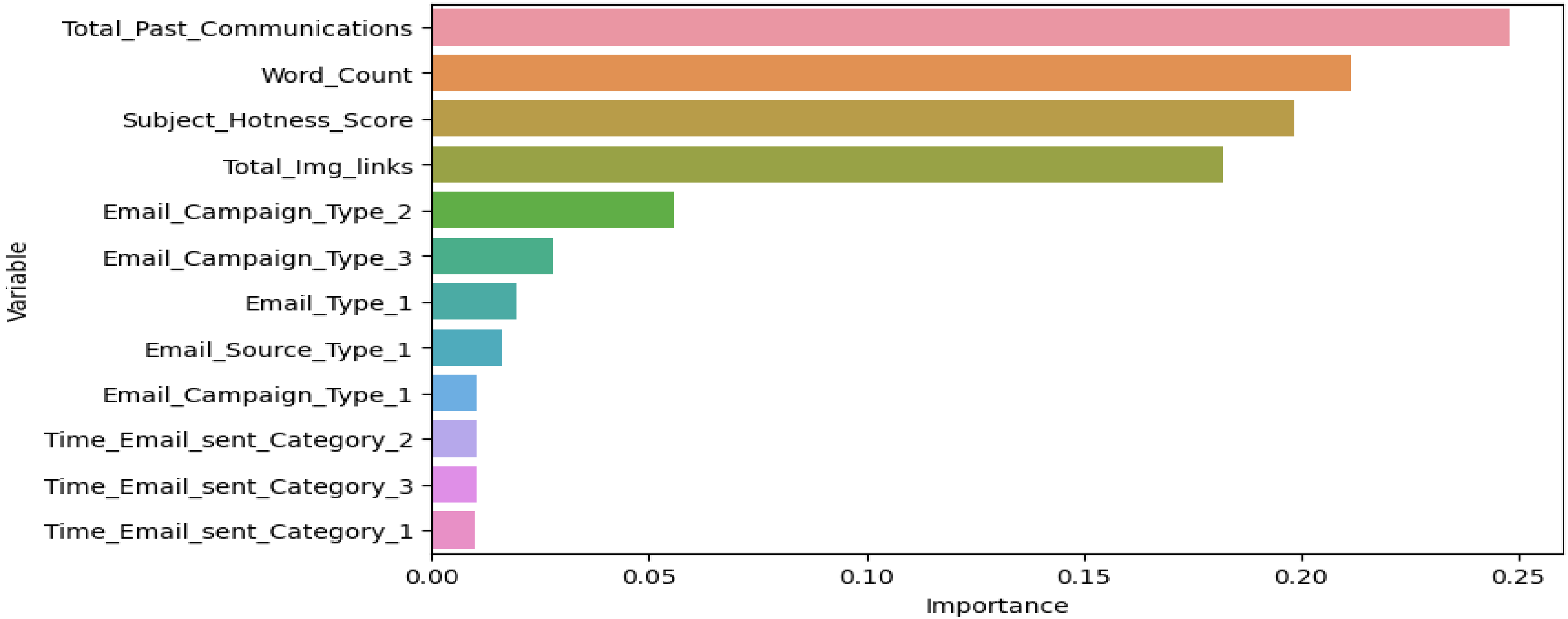
- Variables with high multicollinearity can adversely affect the model and removing highly correlated independent variables can help in reducing curse of dimensionality as well

- We can observe that all numerical variables are within the threshold(i.e. 5).

	variables	VIF
0	Subject_Hotness_Score	1.734531
1	Total_Past_Communications	3.430879
2	Word_Count	3.687067
3	Links_Images	2.629047

Feature Engineering

3. Understanding Feature Importance:



Feature Engineering

3. Understanding Feature Importance:

- The concept used to understand feature importance is **Information Gain**.

  Why?

- It explains which feature has maximum impact in classification based on the **notion of Entropy**.
- It works well for **numeric** as well as **categorical** data

- From the graph we understand that **Total_Past_Communications** and **Email_Campaign_Type** have **high importance**.
- **Time_Email_Sent_Category and Customer_Location are not important** and hence we decide to drop the feature.

Feature Engineering

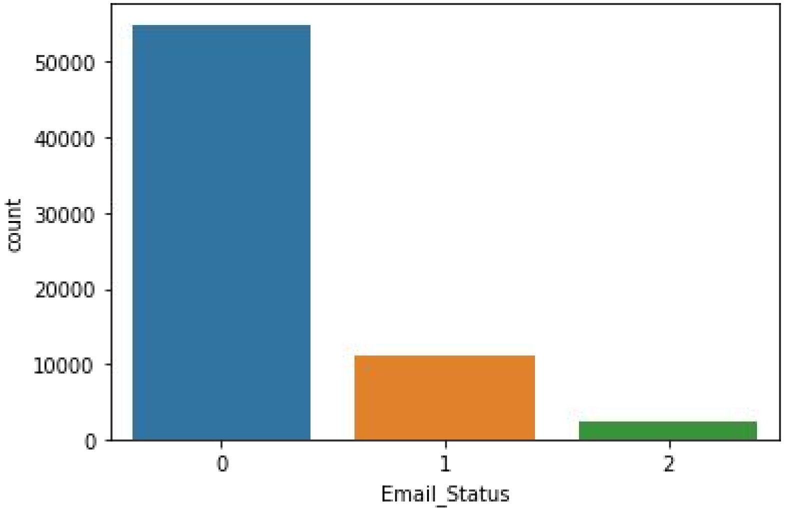
- **Numerical variables** were scaled using **MinMaxScaler**.

  Why?
  The numerical features of the dataset do not have a certain range and they differ from each other.

- **Categorical variables** were encoded using **One-Hot Encoding**.

  Why?
  This method changes categorical data to a numerical format and enables you to group your categorical data without losing any information.

Understanding Target Variable



The target variable consists of 3 classes:

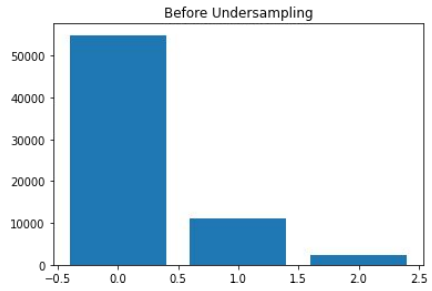
- 0 - ignored - 54941
- 1- read - 11039
- 2 - acknowledged - 2373

Target Variable was **highly imbalanced**.

Handling Imbalanced data

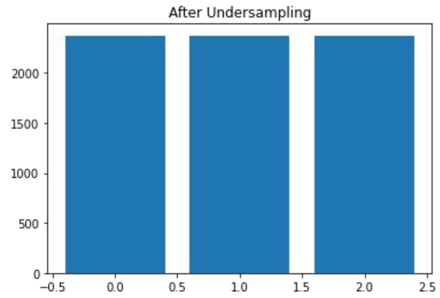
1. Undersampling Technique:

- Technique used was **Random UnderSampler**
- Created balanced data with **2373** records for each class.



Why it didn’t work?

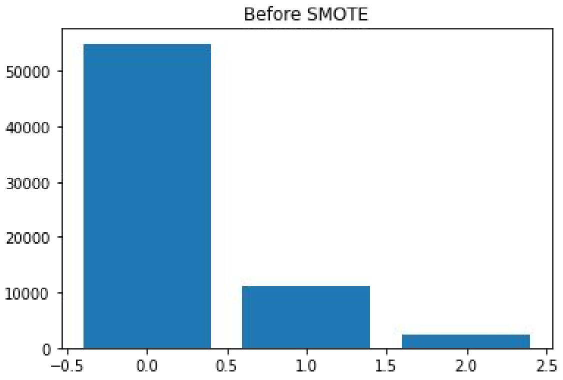
Created baseline models with undersampled data and it was observed that they underperformed primarily due to **loss of information.**

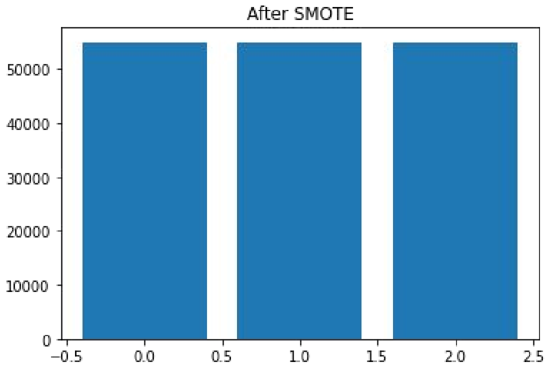


Handling Imbalanced data

2. Oversampling Technique:

- Technique used was **SMOTE**
- Created balanced data with **54941** records for each class.

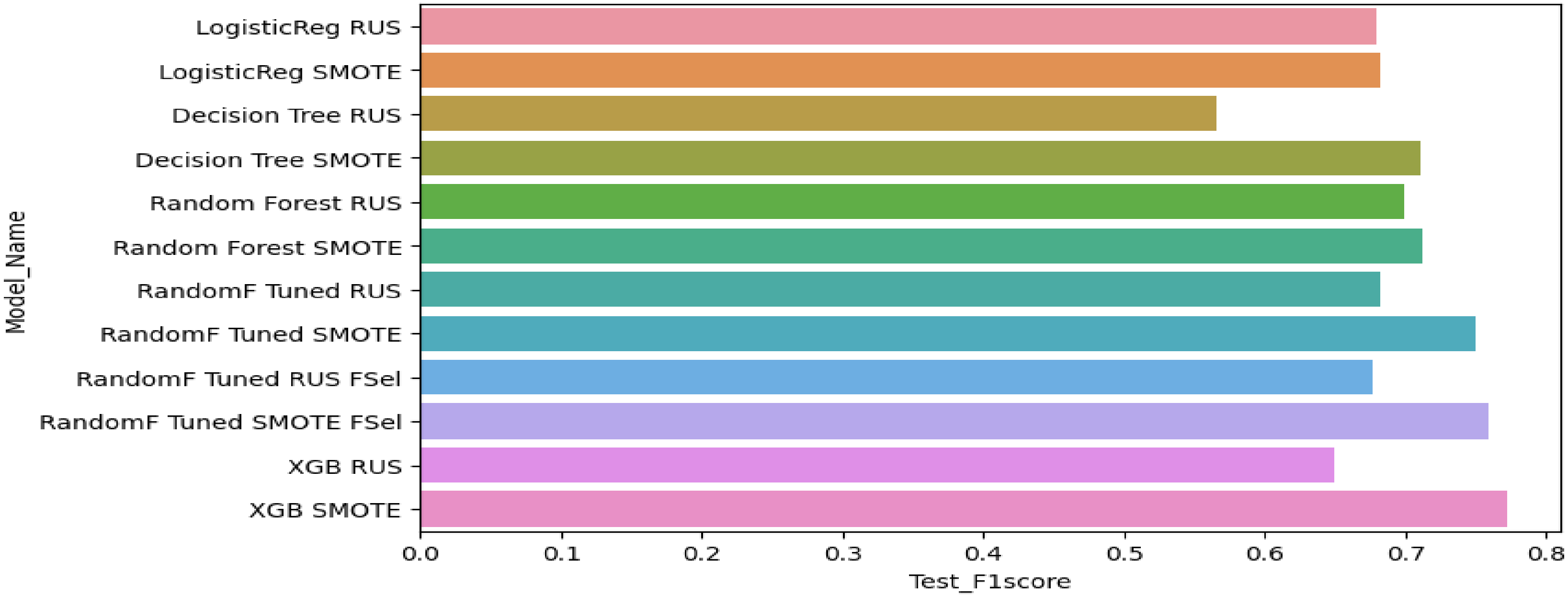




Different Models

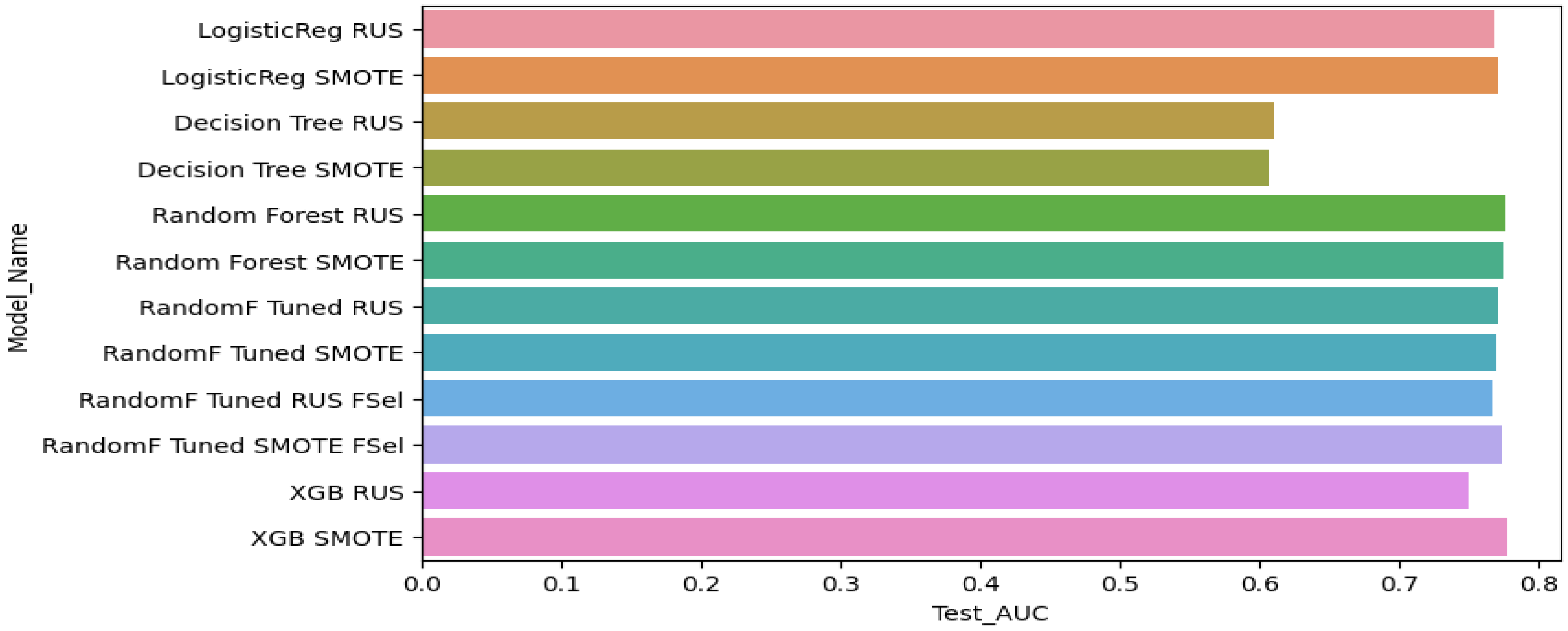
Evaluation Metrics:

1. F1_Score



Different Models

2. **ROC_AUC_Score**



Winner Model

XGBoost SMOTE:

- Robust to outliers.
- Supports regularization.
- Works well on small to medium dataset.
- F1 score for train & test set were 89% & 81% respectively

Conclusion

- In EDA, we observed that Email_Campaign_Type was the most important feature. If your Email_Campaign_Type was 1,there is a 90% likelihood of your Email to be read/acknowledged.

- It was observed that both Time_Email_Sent and Customer_Location were insignificant in determining the Email_status. The ratio of the Email_Status was same irrespective of the demographic location or the time frame the emails were sent on.

- As the word_count increases beyond the 600 mark we see that there is a high possibility of that email being ignored. The ideal mark is 400-600. No one is interested in reading long emails !

- For modelling, it was observed that for imbalance handling Oversampling i.e. SMOTE worked way better than undersampling as the latter resulted in a lot of loss of information.

- Based on the metrics, XGBoost Classifier worked the best giving a train score of 89% and test score of 81% for F1 score.

Challenges

- Choosing the appropriate technique to handle the imbalance in data was quite challenging as it was a tradeoff b/w information loss vs risk of overfitting.

- Overfitting was another major challenge during the modelling process.

- Understanding what features are most important and what features to avoid was a difficult task.

- Decision making on missing value imputations and outlier treatment was quite challenging as well.

Thank You
AlmaBetter Team