Capstone Project Submission

Instructions:

i) Please fill in all the required information.
ii) Avoid grammatical errors.

Team Member's Name, Email and Contribution:

- Karan Rawat (rawatkarankr99@gmail.com)

- Data Reading
- Exploratory Data Analysis
- Data Preparation before Modelling
- Handling Class Imbalance
- Metrics Selection
- Model Explainability and Feature Importance
- Modelling
- Feature Engineering and Modelling
- Final Selected Model

Other than this I prepared summary, PPT report and Technical document for this

- Bhaskar Mendhe (bhaskarmendhe358@gmail.com)

- Data Reading
- Exploratory Data Analysis
- Data Preparation before Modelling
- Handling Class Imbalance
- Metrics Selection
- Model Explainability and Feature Importance
- Modelling
- Feature Engineering and Modelling
- Final Selected Model

- Other than this I prepared summary, PPT report and Technical document for this study.

Please paste the GitHub Repo link.

Github Link:- https://github.com/rawatkarankr99/Capstone-Project---3.git

Summary

- The objective is to create a machine learning model to characterize the mail that is ignored ; read ; acknowledged by the reader.
- There are total 12 features out of which the feature "Email_Status " is a response variable and rest are predictor variables.
- The features "Customer_Location", "Total_Past_Communications", "Total_Links" and "Total_Images" have missing values. We need to take care of these missing values.
- There are no duplicate rows in the dataset.
- There is high imbalance in class distribution of response variable. The majority of the data, 54941 data points which is 80.38 % belongs to "class 0", 11039 points which is 16.15 % belongs to class "1" and very small amount of data, 2373 data points which is 3.47% belongs to "class 2".
- The features "Email_Type", "Email_Source_Type", "Email_Campaign_Type" and "Time_Email_sent_Category" contains categorical information , so we have changed the datatype according to that.
- There were five categorical and numerical features each afterwards.
- The numerical features 'Subject_Hotness_Score','Total_Links' and 'Total_Images' were positively skewed.
- The Log, Square Root and Cube Root transformations used for removing skewness and these were able to remove the skewness , but Power transformation removed skewness outstandingly and also it standardize the data as well.
- All the numerical features except 'Word_Count' had outliers.
- The numerical features "Total_Images" and "Total_Links" were highly correlated with correlation value as 0.78.
- There are two subcategories under "Email_Type" and "Email_Source_Type" features, three subcategories under "Email_Campaign_Type" and "Time_Email_Sent_Category" features, and seven subcategories under "Customer_Location" feature.
- For the feature "Email_Type" ,there are more number of points of Type 1 rather than Type 2 .For "Email_Source_Type",there is slight difference in distribution of points of Type 1 and Type 2. For "Customer_Location", majority of the points belongs to category G. For "Email_Campaign_Type" , majority of the points belongs to Type 2. For the "Time_Email_Sent_Category" also, majority of the points belongs to Type 2.
- The distribution of Email_Status is similar in all categorical features except "Email_campaign_Type".There are more number of points related to majority class in each feature.For "Email_campaign_Type" as Type 1 ,the distribution of points w.r.t to classes can be seen similar, as there are very less number of points for the same.
- The "Email_Id" feature contains identity information. So dropping this feature.
- As there is high class imbalance and we have very few points of class "2" and class "1" , so we have removed only those outliers which belongs to class "1".
- We have used Iterative Imputer to fill missing values of numerical features and treating missing values of categorical feature as separate category using Simple Imputer.
- To remove collinearity combination of collinear features as a single feature have been taken.
- We have used Power Transformer for transforming numerical features as it helps in removing the skewness and standardizing as well and One hot Encoder for Categorical Features.
- The shape of Train Set becomes - (50075, 22) and Test Set Shape becomes =(12519, 22)
- There is high class imbalance in the dataset. To solve this we will provide different weights to both the majority and minority classes. The difference in weights will influence the classification of the classes during the training phase. The whole purpose is to penalize the misclassification made by the minority class by setting a higher class weight and at the same time reducing weight for the majority class. The weights can be assigned according to classes simply by using parameter "class_weight" as "balanced" while defining the machine learning models.

- There is high imbalance of classes in the dataset, and also our objective is to classify the mails as ignored ; read and acknowledged as correctly as possible to corresponding classes.So fo this task we will look upon weighted Precision,Recall and F1 score, as Precision and Recall account for true positives which is nothing but correctly classified points belonging to respective classes and F1 score is just harmonic mean of these two which is a combined single metric to look for. We will also look for ROC_AUC score.
- The objective is to classify the mails as ignored ; read and acknowledged. For this , we need to find the reasons why the mails being ignored , how many being read and finally how many being acknowledged.So wee need a model which can explain the reasons for classifications , so that we can improve the content for the mails such that mails could get read and acknowledged in the future which helps the owners stay connected with their prospective customers. The feature importance is also very important in this case as we need to know which are the most important features for classification, so we can focus on those to improve the content for mails.
- The training and test scores don't differ much for tuned Logistic Regression which is a good sign. It gives score of precision as 0.762 and roc_auc as 0.779 but recall score is low which is 0.629 and thus f1 score is also low which is 0.68. It is able to correctly classify 7010 points out of 9836 of class 1. It is able to correctly classify 298 points out of 475 of class 2, but it still classified only 569 points out of 2208.
- The training and test scores for tuned Naive Bayes don't differ much so the model is not overfitting or underfitting. The Naive Bayes gives test score of precision as 0.727 and roc_auc as 0.714 ,recall score as 0.69, and f1 score as 0.71. It is able to correctly classify 7521 points out of 9836 of class 0. It is able to correctly classify 1128 points out of 2208 of class 1 which is a great job done , but no points of class 2 have been classified which is not a good sign.
- The difference between training and test scores of tuned Decision Tree are very large . The model gives very high scores for train set but not for test set , so it is clearly overfitting. The Decision Tree gives test score of precision as 0.727 and roc_auc as 0.646 ,recall score as 0.684, and f1 score as 0.703. It is able to correctly classify 7664 points out of 9836 of class 0. It is able to correctly classify 846 points out of 2208 of class 1 and only 63 points out of 475 of class 2.
- The difference between training and test scores of tuned Random Forest are very large. The model gives very high scores for train set but not for test set , so it is clearly overfitting. The Random Forest gives test score of precision as 0.747 and roc_auc as 0.777 ,recall score as 0.798, and f1 score as 0.758. It is able to correctly classify 9450 points out of 9836 of class 0 which is great job done. It is able to correctly classify 531 points out of 2208 of class 1 and only 13 points out of 475 of class 2.
- The training and test scores for XGBoost don't differ much so the model is not overfitting or underfitting. The XGBoost gives test score of precision as 0.768 and roc_auc as 0.819 ,recall score as 0.814, and f1 score as 0.768. It is able to correctly classify 9668 points out of 9836 of class 0 which is a great job done. It is able to correctly classify 532 points out of 2208 of class 1, but only 2 points of class 2 have been classified which is not a good sign.
- Then we have done Feature engineering by transforming numerical features into polynomials of degree 5.
- The training and test scores for tuned Logistic Regression don't differ much which is a good sign. It gives score of precision as 0.773 and roc_auc as 0.791 but recall score is low which is 0.642 and thus f1 score is also low which is 0.693. It is able to correctly classify 7132 points out of 9836 of class 1. It is able to correctly classify 291 points out of 475 of class 2, but it still classified only 622 points out of 2208.
- The difference between training and test scores of Decision Tree are very large . The model gives very high scores for train set but not for test set , so it is clearly overfitting. The Decision Tree gives test score of precision as 0.725 and roc_auc as 0.632 ,recall score as 0.70, and f1 score as 0.711. It is able to correctly classify 7905 points out of 9836 of class 0. It is able to correctly classify 787 points out of 2208 of class 1 and only 75 points out of 475 of class 2.
- The difference between training and test scores of Random Forest are very large. The model gives very high scores for train set but not for test set , so it is clearly overfitting. The Random Forest gives test score of precision as 0.765 and roc_auc as 0.801 ,recall score as 0.782, and f1 score as 0.773. It is able to correctly classify 8809 points out of 9836 of class 0 which is great job done. It is able to correctly classify 943 points out of 2208 of class 1 and only 40 points out of 475 of class 2.

- The training and test scores for XGBoost don't differ much so the model is not overfitting or underfitting. The XGBoost gives test score of precision as 0.772 and roc_auc as 0.816 ,recall score as 0.812, and f1 score as 0.762. It is able to correctly classify 9701 points out of 9836 of class 0 which is a great job done. It is able to correctly classify 475 points out of 2208 of class 1, but only 1 point of class 2 have been classified which is not a good sign.

Conclusions

- The final model selected is Logistic Regression.
- Taking Scores into consideration, XGBoost outperforms all models , but it hardly classify 1 or 2 points correctly from the minority class. At other hand Logistic Regression being simplest model able to correctly classify most number of points from minority class other than any model. Also Logistic Regression is very easy to interpret , as it fits a hyperplane for a classification.
- Also We can derive feature importance from the coefficient of Logistic Regression very easily. To get the features which are important to classify points of class 0, we will look at the first array of coefficient. The more larger values of coefficients corresponding to features , then more the feature is important. Similarly we fetch the features which are important for classification of points belong to class 1 and class 2.
- The "Email_Campaign" as a whole feature is the most important individual feature for classification. The other features "Total_Past_Communications","Word_Count","Subject_Hotness_Score","Total_Link_Images" which is combination of "Total_Links" and 'Total_Images", are next important features which are used in combinations of polynomials for classifications. So we can look upon these features to improve the business work.

Challenges Faced

- The dataset contains missing values.
- The dataset contains outliers.
- Some of the features mapped to wrong datatype.
- The dataset contains correlated features also.