

DTH CUSTOMER CHURN

PROJECT NOTES 1



November 28, 2021

sAI kOMARAVOLU

1. [Problem Understanding.](#ProblemUnderstanding)
2. Data Report
3. Exploratory Data Analysis
4. Data Cleaning and Pre Processing
5. Model Building
6. Model Validation
7. Final Interprettation/Recommendation

AGENDA

1. **Need of the study**
   1. A DTH company is facing a lot of competition in the current market, and it has become a challenge to retain existing customers.
   2. Company wants to develop a model through which they can do churn prediction of the accounts and provide segmented offers to the potential churners.
   3. In this company, account churn is a major thing because 1 account can have multiple customers. hence by losing one account the company might be losing more than one customer.
   4. I have been assigned to develop a churn prediction model for this company and provide business recommendations on the campaign.
   5. I need to suggest a very clear idea on the campaign offer, which should not negatively impact the revenue of the company.
2. **Business/Social Opportunity**
   1. Company’s motive is to decrease the Customer’s churn away rate by offering good deals and cashbacks.
   2. Essential features must be identified, that help company to achieve less churn.
   3. Identify similar cluster of customers who churn, this will help to curate offers and retain them.
   4. By studying Company’ data I can help to retain the right customers and accordingly provide lucrative promotional offers to make them permanent customers.

Problem understanding

1. Data collection
   1. Data was provided to me in a excel from customer.
   2. Dataset consists of all customer accounts, and this is not a timeseries dataset.
2. Visual inspection, Data shape & summary
   1. Dataset is presented with 11260 rows and 19 features.
   2. AccountID is the identifier that uniquely points to a customer.
   3. Churn feature is a categorical column, which with value 0 is not churned and 1 is churned.
   4. Dataset has 5 float, 2 int and 12 object typed features, however few object typed columns need to be converted to numeric.
3. Data cleansing
   1. Features have been converted to upper case for the ease of analysis.
   2. Features like Tenure, Account\_user\_count, Rev\_per\_month, rev\_growth\_yoy, coupon\_used\_for\_payment and day\_since\_cc\_connect are converted to numeric.

DATA REPORT

1. Missing Values
   1. Dataset has missing values with most of the features which will be imputed.
   2. Tenure has invalid category “#” which is imputed by max category “1”.
   3. Gender feature has F and M which were imputed to be “Female” and “Male”.
   4. Account\_User\_count has “@” which was imputed to category “4”.
   5. Coupon\_used\_for\_payment has “\*”, “#” and “$” which were imputed to category “1”.
   6. Login\_Device has “&&&&” which was imputed to be “Mobile”.
   7. Day\_Since\_cc\_connect has “$” which was imputed to be category “3”.
   8. Cashback column had many empty cells, which were imputed to value 0.
   9. Cashback column had “$” for 2 cells, which were imputed to value 0.
2. Categorical Features
   1. Payment feature has 5 categories; UPI, Cash on Delivery, E wallet, Credit Card and Debit Card. Maximum account holders have debit card and next maximum is credit card.
   2. Gender feature has 2 categories, Male and Female. Male category accounts for maximum account holders.
   3. Account\_Segment feature has 7 categories, Super category caters to maximum account holders.
   4. Marital\_Status features has 3 categories, Married category caters to maximum account holders.
   5. Login\_Device feature has 2 categories, where in Mobile category has maximum account holders.

exploratory data analysis

Chart, histogram

Description automatically generated

Account\_ID feature is normally distributed, box plot does not hold any importance as this feature is a primary key for all transactions.

Chart

Description automatically generatedCity\_tier feature has customers from City\_tier “1” and less customers from City\_tier “2”.

eDA – Univariate analysis

Graphical user interface

Description automatically generated with medium confidence1. CC\_Contacted\_12m feature is a right skewed, at peak, 20 times customer had contacted customer care. This feature also has outliers with some customers reaching customer cate in excess of 120 times.

2. SERVICE\_SCORE feature is categorical and many customers felt that service is average.

eDA – Univariate analysis

Chart, box and whisker chart

Description automatically generated

Chart, box and whisker chart

Description automatically generated

1. Service\_Score feature is categorical, and customer’s popular opinion is that the service is average.
2. Tenure feature is a right skewed feature. Most of the customers are new and long time customers are present as outliers.

eDA – Univariate analysis

Chart

Description automatically generated

1. Account\_user\_count feature is categorical and many customers have 4 co-customers using the DTH.
2. Rev\_per\_month feature is normally distributed with outliers, some customers seem to bring huge revenue in comparison with others.
3. Rev\_growth\_yoy is consistent and right skewed with declining growth in the last 12 months.

eDA – Univariate analysis

Chart, histogram

Description automatically generated

1. Coupon\_used\_for\_payment feature is a right skewed feature without consistent peaks. Many customers have used 1 or 2 coupons in the time with DTH. Some customers have heavily used coupons which can be seen as outliers.
2. Day\_since\_cc\_connect is a right skewed feature without consistent peaks and outliers. Some customers have even contacted customer care 40+ times, this could be the reason for churn.

eDA – Univariate analysis

Graphical user interface, square

Description automatically generated with medium confidence

1. Cc\_agent\_score have rated the service to be average.
2. Complain\_12m, early customers have raised many complaints.
3. Cashback is high for customers is normally distributed, some customers have received maximum cashback.

eDA – Univariate analysis

Chart, bar chart

Description automatically generatedChart, bar chart

Description automatically generated

1. Customers who hold debit cards are higher than other payment modes.
2. Male customers are higher among customers.
3. Account\_segment super have many customers next to regular\_plys account\_segment.
4. Married customers are higher.

Chart, bar chart

Description automatically generated

1. Customers that are logged in to dth by mobile are higher.

eDA – Univariate analysis

A picture containing histogram

Description automatically generated1. All features are scattered and do not show any correlations with each other.

eDA – BIVARIATE analysis

A picture containing chart

Description automatically generated1. No features have strong correlations with each other.

2. Some features have very weak correlations like coupon\_used\_for\_payment.

eDA – BIVARIATE analysis

1. REV\_GROWTH\_YOY, COUPON\_USED\_FOR\_PAYMENT, CITY\_tIER, CC\_CONTACTED\_LY, SERVICE\_SCORE, REV\_PER\_MONTH and CASHBACK have near to 0 correlation with CHURN, need to investigate to curate more features from them or to drop them.
2. Assumptions:
   1. Outliers were obtained for Cashback, day\_since\_cc\_connect, coupon\_used\_for\_payment, rev\_per\_month, tenure, cc\_contacted\_ly.
   2. These outliers are acceptable and I believe they show the real situation of the customers hold by DTH and should not transform the outliers.
3. Data Balance:
   1. Churned customers are 16.83% and Not-Churned customers are 83.16%. This is highly imbalanced target class.
   2. Usage of SMOTE or K-Means to under or oversample needs to be identified.

eDA – Data IMPORTANCE

1. Yes the data is unbalanced, we need to get the balance for Y to 45% and N to 55% for optimal modelling and prediction.
2. We can use SMOTE to oversample Y or use K-Means to undersample N.
3. Need to perform feature engineering, apply PCA and see combined effect of features on the target class.
4. We can perform LDA to generate features by considering the target class.
5. From the analysis, customer service, features play an important role in churn, many customers have contacted customer care in excess of 120 times.

Business insights

**Project Notes -2**

**Dataset Encoding/Outlier Handling/Missing Value analysis**

Chart, bar chart

Description automatically generated

Summary:

1. COMPLAIN\_LY feature has highest invalid records and next to it is the DAY\_SINCE\_CC\_CONNECT. Reason might be the customer never contacted customer care on a yearly basis.
2. Mode is used to impute the values for COMPLAN\_LY.
3. Similarly other missing values have been imputed using Mean, Mode and Median appropriately.

**Outliers:**

Chart

Description automatically generated

Summary:

1. Barring ACCOUNT\_ID and CHURN, some features have outliers.
2. All outliers have been treated using IQR logic.
3. Categorical columns were encoded using One Hot Encoding to proceed with Machine Learning.

**1(a, b, and c) Model Building and Interpretation**

**(Target Variable Imbalance 85% and 15%)**

**Preface:**

1. Minority churn variable prediction will be given a priority.
2. We will use the recall score as the parameter to determine best model.

**Naïve Bayes with Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Naïve Bayes - With Outliers** | | | | | | | | |
| **Accuracy Train: 0.79** | | | | **Accuracy Test: 0.79** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.92 | 0.91 | 0.82 | 0.84 | 0.87 | 0.87 | 7023 | 2341 |
| 1 | 0.42 | 0.42 | 0.63 | 0.59 | 0.5 | 0.5 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. With Outliers, Naïve Bayes has same accuracy score of 79% for both train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 63% and Test data is 59%.
3. On train data 63 of 100 times algorithm can predict customer churned, on test data only 59 times of 100 algorithm is able to predict.
4. This score is not the best one to perform prediction, this algorithm will not be considered.

**Naïve Bayes without Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Naïve Bayes - Without Outliers** | | | | | | | | |
| **Accuracy Train: 0.77** | | | | **Accuracy Test: 0.78** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.93 | 0.93 | 0.79 | 0.80 | 0.86 | 0.86 | 7023 | 2341 |
| 1 | 0.41 | 0.41 | 0.70 | 0.70 | 0.51 | 0.52 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. Without Outliers, Naïve Bayes has same accuracy score of 77% and 78% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 70% and Test data is 70%.
3. On train data 70 of 100 times algorithm can predict customer churned, on test data only 70 times of 100 algorithm is able to predict.
4. Train and Test sensitivity is good having same score for both, will be considered as a best model, and further evaluated.**Linear Discriminant Analysis with Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Linear Discriminant Analysis - With Outliers** | | | | | | | | |
| **Accuracy Train: 0.87** | | | | **Accuracy Test: 0.86** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.98 | 0.88 | 0.97 | 0.97 | 0.93 | 0.92 | 7023 | 2341 |
| 1 | 0.71 | 0.69 | 0.40 | 0.35 | 0.51 | 0.46 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. With Outliers, Linear Discriminant Analysis has accuracy score of 87% and 86% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 40% and Test data is 35%.
3. On train data 40 of 100 times algorithm can predict customer churned, on test data only 35 times of 100 algorithm is able to predict.
4. Train and Test sensitivity is poor and this model will not be considered further.

**Linear Discriminant Analysis without Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Linear Discriminant Analysis - Without Outliers** | | | | | | | | |
| **Accuracy Train: 0.79** | | | | **Accuracy Test: 0.79** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.90 | 0.90 | 0.97 | 0.97 | 0.93 | 0.93 | 7023 | 2341 |
| 1 | 0.76 | 0.75 | 0.49 | 0.46 | 0.59 | 0.57 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. Without Outliers, Linear Discriminant Analysis has accuracy score of 79% and 79% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 49% and Test data is 46%.
3. On train data 49 of 100 times algorithm can predict customer churned, on test data only 46 times of 100 algorithm is able to predict.
4. Train and Test sensitivity is poor and this model will not be considered further.

**KNN With Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: KNN - With Outliers** | | | | | | | | |
| **Accuracy Train: 0.84** | | | | **Accuracy Test: 0.81** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.86 | 0.84 | 0.98 | 0.97 | 0.92 | 0.90 | 7023 | 2341 |
| 1 | 0.69 | 0.32 | 0.18 | 0.07 | 0.29 | 0.11 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. With Outliers, KNN algorithm has accuracy score of 84% and 81% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 18% and Test data is 7%.
3. Train and Test sensitivity is poor and this model will not be considered further.

**KNN Without Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: KNN - Without Outliers** | | | | | | | | |
| **Accuracy Train: 0.85** | | | | **Accuracy Test: 0.81** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.86 | 0.84 | 0.98 | 0.97 | 0.92 | 0.90 | 7023 | 2341 |
| 1 | 0.70 | 0.32 | 0.19 | 0.07 | 0.30 | 0.11 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. Without Outliers, KNN algorithm has accuracy score of 85% and 81% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 19% and Test data is 7%.
3. Train and Test sensitivity is poor and this model will not be considered further.

**Decision Tree with Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Decision Tree - With Outliers** | | | | | | | | |
| **Accuracy Train: 1** | | | | **Accuracy Test: 0.92** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 1 | 0.96 | 1 | 0.95 | 1 | 0.96 | 7023 | 2341 |
| 1 | 1 | 0.77 | 1 | 0.79 | 1 | 0.78 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. With Outliers, Decision Tree has accuracy score of 100% and 92% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 100% and Test data is 79%.
3. On train data 100 of 100 times algorithm can predict customer churned, on test data only 79 times of 100 algorithm is able to predict.
4. Deviation of 20% for recall score between is train and test is not a good parameter for a prediction model.
5. However, Train and Test sensitivity is good having same score for both, will be considered as a best model, and further evaluated.

**Decision Tree without Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Decision Tree - Without Outliers** | | | | | | | | |
| **Accuracy Train: 1** | | | | **Accuracy Test: 0.94** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 1 | 0.96 | 1 | 0.95 | 1 | 0.96 | 7023 | 2341 |
| 1 | 1 | 0.82 | 1 | 0.80 | 1 | 0.81 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. Without Outliers, Decision Tree has accuracy score of 100% and 94% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 100% and Test data is 80%.
3. On train data 100 of 100 times algorithm can predict customer churned, on test data only 80 times of 100 algorithm is able to predict.
4. Marginally improved by 1% after addressing outliers. Deviation of 20% for recall score between is train and test is not a good parameter for a prediction model.
5. However, Train and Test sensitivity is good having same score for both, will be considered as a best model, and further evaluated.

**Random Forest with Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Random Forest - With Outliers** | | | | | | | | |
| **Accuracy Train: 1** | | | | **Accuracy Test: 0.96** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 1 | 0.96 | 1 | 1 | 1 | 0.98 | 7023 | 2341 |
| 1 | 1 | 0.99 | 1 | 0.80 | 1 | 0.89 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. With Outliers, Random Forest algorithm has accuracy score of 100% and 96% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 100% and Test data is 80%.
3. On train data 100 of 100 times algorithm can predict customer churned, on test data only 80 times of 100 algorithm is able to predict.
4. Deviation of 20% for recall score between is train and test are not a good parameter for a prediction model.
5. However, Train and Test sensitivity is good having same score for both, will be considered as a best model, and further evaluated.

**Random Forest without Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Random Forest - Without Outliers** | | | | | | | | |
| **Accuracy Train: 1** | | | | **Accuracy Test: 0.96** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 1 | 0.96 | 1 | 1 | 1 | 0.98 | 7023 | 2341 |
| 1 | 1 | 0.99 | 1 | 0.79 | 1 | 0.88 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
| Chart  Description automatically generated |  |

Summary:

1. Without Outliers, Random Forest algorithm has accuracy score of 100% and 96% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 100% and Test data is 79%.
3. On train data 100 of 100 times algorithm can predict customer churned, on test data only 79 times of 100 algorithm is able to predict.
4. No impact of Outlier treatment is observed. Deviation of 20% for recall score between is train and test are not a good parameter for a prediction model.
5. However, Train and Test sensitivity is good having same score for both, will be considered as a best model, and further evaluated.

**Logistic Regression with Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Logistic Regression - With Outliers** | | | | | | | | |
| **Accuracy Train: 0.88** | | | | **Accuracy Test: 0.88** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.90 | 0.89 | 0.97 | 0.97 | 0.93 | 0.93 | 7023 | 2341 |
| 1 | 0.78 | 0.74 | 0.44 | 0.40 | 0.56 | 0.52 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. With Outliers, Logistic Regression algorithm has accuracy score of 88% and 88% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 44% and Test data is 40%.
3. On train data 44 of 100 times algorithm can predict customer churned, on test data only 40 times of 100 algorithm is able to predict.
4. Train and Test sensitivity is poor and this model will not be considered further.

**Logistic Regression without Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Logistic Regression - Without Outliers** | | | | | | | | |
| **Accuracy Train: 0.89** | | | | **Accuracy Test: 0.89** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.91 | 0.90 | 0.97 | 0.97 | 0.94 | 0.94 | 7023 | 2341 |
| 1 | 0.78 | 0.78 | 0.51 | 0.49 | 0.62 | 0.60 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. With Outliers, Logistic Regression algorithm has accuracy score of 89% and 89% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 51% and Test data is 49%.
3. On train data 51 of 100 times algorithm can predict customer churned, on test data only 49 times of 100 algorithm is able to predict.
4. Though recall improved after outlier treatment, Train and Test sensitivity is poor and this model will not be considered further.

**2(a, b, and c) Model Tuning and Business Implication**

**Ensemble Modelling**

**AdaBoost With Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Ada Boost - With Outliers** | | | | | | | | |
| **Accuracy Train: 0.89** | | | | **Accuracy Test: 0.90** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.92 | 0.92 | 0.96 | 0.96 | 0.94 | 0.94 | 7023 | 2341 |
| 1 | 0.76 | 0.77 | 0.60 | 0.59 | 0.67 | 0.67 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. With Outliers, AdaBoost algorithm has accuracy score of 89% and 90% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 60% and Test data is 59%.
3. On train data 60 of 100 times algorithm can predict customer churned, on test data only 59 times of 100 algorithm is able to predict.
4. Train and Test score is average, and this model will be evaluated after removing outliers.

**AdaBoost Without Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Ada Boost - Without Outliers** | | | | | | | | |
| **Accuracy Train: 0.89** | | | | **Accuracy Test: 0.90** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.92 | 0.92 | 0.96 | 0.97 | 0.94 | 0.94 | 7023 | 2341 |
| 1 | 0.77 | 0.79 | 0.58 | 0.60 | 0.66 | 0.68 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. Without Outliers, AdaBoost algorithm has accuracy score of 89% and 90% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 58% and Test data is 60%.
3. On train data 58 of 100 times algorithm can predict customer churned, on test data only 60 times of 100 algorithm is able to predict.
4. Train and Test score is average, and this model will not be considered for Prediction.

**Gradient Boost with Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Gradient Boost - With Outliers** | | | | | | | | |
| **Accuracy Train: 0.91** | | | | **Accuracy Test: 0.91** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.93 | 0.93 | 0.97 | 0.97 | 0.95 | 0.95 | 7023 | 2341 |
| 1 | 0.82 | 0.80 | 0.63 | 0.62 | 0.71 | 0.70 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
| Chart, treemap chart  Description automatically generated | Chart, treemap chart  Description automatically generated |

Summary:

1. With Outliers, Gradient Boost algorithm has accuracy score of 91% and 91% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 63% and Test data is 62%.
3. On train data 63 of 100 times algorithm can predict customer churned, on test data only 62 times of 100 algorithm is able to predict.
4. Train and Test score is average, and this model will be evaluated after removing outliers.

**Gradient Boost without Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Gradient Boost - Without Outliers** | | | | | | | | |
| **Accuracy Train: 0.92** | | | | **Accuracy Test: 0.91** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.93 | 0.93 | 0.97 | 0.97 | 0.95 | 0.95 | 7023 | 2341 |
| 1 | 0.83 | 0.81 | 0.62 | 0.62 | 0.71 | 0.70 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. Without Outliers, AdaBoost algorithm has accuracy score of 92% and 91% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 62% and Test data is 62%.
3. On train data 62 of 100 times algorithm can predict customer churned, on test data only 62 times of 100 algorithm is able to predict.
4. Train and Test score is average and they are same, further evaluation is required to consider this model for prediction.

**Bagging With Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Bagging - With Outliers** | | | | | | | | |
| **Accuracy Train: 1** | | | | **Accuracy Test: 0.96** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 1 | 0.97 | 1 | 0.99 | 1 | 0.98 | 7023 | 2341 |
| 1 | 1 | 0.95 | 1 | 0.83 | 1 | 0.89 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. With Outliers, Bagging algorithm has accuracy score of 100% and 96% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 100% and Test data is 83%.
3. On train data 100 of 100 times algorithm can predict customer churned, on test data only 83 times of 100 algorithm is able to predict.
4. Deviation of 17% for recall score between is train and test are not a good parameter for a prediction model.
5. However, Train and Test sensitivity is good having same score for both, will be considered as a best model, and further evaluated.

**Bagging Without Outliers**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Bagging - Without Outliers** | | | | | | | | |
| **Accuracy Train: 1** | | | | **Accuracy Test: 0.96** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 1 | 0.97 | 1 | 0.99 | 1 | 0.98 | 7023 | 2341 |
| 1 | 1 | 0.95 | 1 | 0.83 | 1 | 0.89 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
| Chart  Description automatically generated | Chart, treemap chart  Description automatically generated |

Summary:

1. Without Outliers, Bagging algorithm has accuracy score of 100% and 96% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Outlier treatment has no effect.
3. Recall score for Train data is 100% and Test data is 83%.
4. On train data 100 of 100 times algorithm can predict customer churned, on test data only 83 times of 100 algorithm is able to predict.
5. Deviation of 17% for recall score between is train and test are not a good parameter for a prediction model.
6. However, Train and Test sensitivity is good having same score for both, will be considered as a best model, and further evaluated.

**Overall model comparison**

|  |  |  |
| --- | --- | --- |
| **Algorithm** | **Sensitivity Train-Data** | **Sensitivity Test-Data** |
| Bagging-With-Outliers | 1 | 0.83 |
| Bagging-Without-Outliers | 1 | 0.83 |
| Random Forest-With-Outliers | 1 | 0.80 |
| Decision Tree-Without-Outliers | 1 | 0.80 |
| Decision Tree-With-Outliers | 1 | 0.79 |
| Random Forest-Without-Outliers | 1 | 0.79 |
| Naive-Bayes-Without-Outliers | 0.70 | 0.70 |
| Gradient Boost-Without-Outliers | 0.62 | 0.62 |
| Gradient Boost-With-Outliers | 0.63 | 0.62 |
| ADA Boost-Without-Outliers | 0.58 | 0.60 |
| Naive-Bayes-With-Outliers | 0.63 | 0.59 |
| ADA Boost-With-Outliers | 0.60 | 0.59 |
| LDA-Without-Outliers | 0.49 | 0.46 |
| LDA-With-Outliers | 0.40 | 0.35 |
| KNN-With-Outliers | 0.18 | 0.07 |
| KNN-Without-Outliers | 0.19 | 0.07 |

Summary:

1. Bagging algorithm provided best Recall score of 83% on test data, followed by Random Forest, Decision Tree and Naïve Bayes.
2. Having a high recall score does not mean that the algorithm is efficient, error of ~20% between Train and Test predictions specify algorithm did not do a good job predicting the Test data. As the variation increases in test data, Type 2 errors will increase there by incorrect minority class predictions.
3. Naïve Bayes, Gradient Boost and Ada Boost have equal Train and Test sensitivities, prediction results are always assumed to be consistent.
4. We will try to improve Random Forest and Gradient Boost by decreasing the Churn variable imbalance using SMOTE and come to a conclusion.

**SMOTE to optimize results**

**SMOTE on Naïve Bayes**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Naïve Bayes with SMOTE** | | | | | | | | |
| **Accuracy Train: 80** | | | | **Accuracy Test: 0.68** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 0.86 | 0.88 | 0.71 | 0.71 | 0.78 | 0.79 | 7023 | 2341 |
| 1 | 0.75 | 0.27 | 0.88 | 0.53 | 0.81 | 0.36 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. Naïve Bayes has same accuracy score of 80% and 68% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 88% and Test data is 53%.
3. On train data 88 of 100 times algorithm can predict customer churned, on test data only 53 times of 100 algorithm is able to predict.
4. Train and Test sensitivity has been degraded, SMOTE did not help improving models’ efficiency, hence it will not be a prediction model.

**SMOTE on Random Forest**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Algorithm: Random Forest with SMOTE** | | | | | | | | |
| **Accuracy Train: 1** | | | | **Accuracy Test: 0.95** | | | | |
| **Churn** | **Precision** | | **Sensitivity/Recall** | | **F1-Score** | | **Support** | |
| **Train** | **Test** | **Train** | **Test** | **Train** | **Test** | **Train** | **Test** |
| 0 | 1 | 0.96 | 1 | 0.99 | 1 | 0.97 | 7023 | 2341 |
| 1 | 1 | 0.92 | 1 | 0.82 | 1 | 0.86 | 1422 | 474 |

|  |  |
| --- | --- |
| **Confusion Matrix** | |
| **Train** | **Test** |
|  |  |

Summary:

1. Random Forest algorithm has accuracy score of 100% and 95% respectively for train and test data. As we have imbalanced data, we will not use this metric to determine if the model is a good model.
2. Recall score for Train data is 100% and Test data is 80%.
3. On train data 100 of 100 times algorithm can predict customer churned, on test data only 82 times of 100 algorithm is able to predict.
4. Deviation of 18% for recall score between is train and test are not a good parameter for a prediction model.
5. However, Train and Test sensitivity increased by 2% and it is good to see this score improving after SMOTE, will be considered as a best model for Prediction.

**Business Implications and Conclusion**

1. From my analysis, Random Forest Algorithm is considered as a best performing algorithm even on imbalanced data.
2. It is not affected by outliers and prediction results are consistent.
3. Company can use this algorithm to Predict and further optimize using Hyper parameters of Random Forest Algorithm.
4. Dataset has very less Churn rate, however, company has to look out for customer service complaints and call backs from customer, reduction of this complaints is linked to the churn rate.