Credit Card Users Churn Prediction
Project 6

Objectives

- To come up with a classification model that will help the bank improve its services so that customers do not renounce their credit cards..
- Identify which variables are most significant.
- Generate a set of insights and recommendations that will help the bank.
- Explore and visualize the dataset.
- Build a classification model to predict if the customer is going to churn or not.
- Optimize the model using appropriate techniques.
- Generate a set of insights and recommendations that will help the bank.

Data Information

Customer details:

Variable	Description
CLIENTNUM	Client number. Unique identifier for the customer holding the account
Attrition_Flag	Internal event (customer activity) variable - if the account is closed then "Attrited Customer" else "Existing Customer"
Customer_Age	Age in Years
Gender	Gender of the account holder
Dependent_count	Number of dependents
Education_Level	Educational Qualification of the account holder - Graduate, High School, Unknown, Uneducated, College(refers to a college student), Post-Graduate, Doctorate.
Marital_Status	Marital Status of the account holder
Income_Category	Annual Income Category of the account holder
Card_Category	Type of Card
Months_on_book	Period of relationship with the bank
Total_Relationship_Count	Total no. of products held by the customer
Months_Inactive_12_mon	No. of months inactive in the last 12 months
Contacts_Count_12_mon	No. of Contacts between the customer and bank in the last 12 months
Credit_Limit	Credit Limit on the Credit Card
Total_Revolving_Bal	The balance that carries over from one month to the next is the revolving balance
Avg_Open_To_Buy	Open to Buy refers to the amount left on the credit card to use (Average of last 12 months)
Total_Trans_Amt	Total Transaction Amount (Last 12 months)
Total_Trans_Ct	Total Transaction Count (Last 12 months)
Total_Ct_Chng_Q4_Q1	Ratio of the total transaction count in 4th quarter and the total transaction count in 1st quarter
Total_Amt_Chng_Q4_Q1	Ratio of the total transaction amount in 4th quarter and the total transaction amount in 1st quarter
Avg_Utilization_Ratio	Represents how much of the available credit the customer spent

Observations	Variables
10127	21

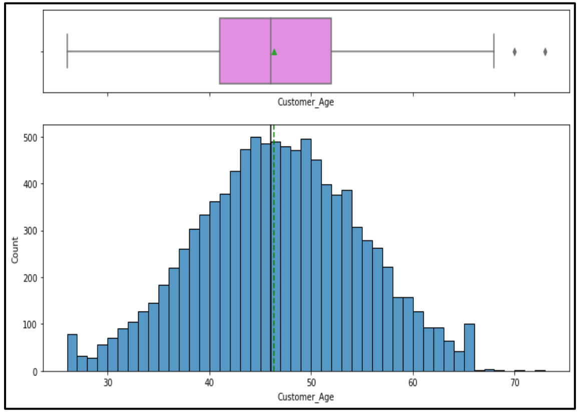
Note:

- There are 2 variables with missing values:
- [Education_Level and Marital_Status]
- We notice that for the Income_Category there is a non numerical range of data with 'abc' that will need to be addressed.
- Attrition_Flag is going to be our dependent variable and as such will need to be converted from string of Attrited and Existing customer to 0 and 1.
- 0 will be Attrited customer and 1 will be existing customer

greatlearning Power Ahead

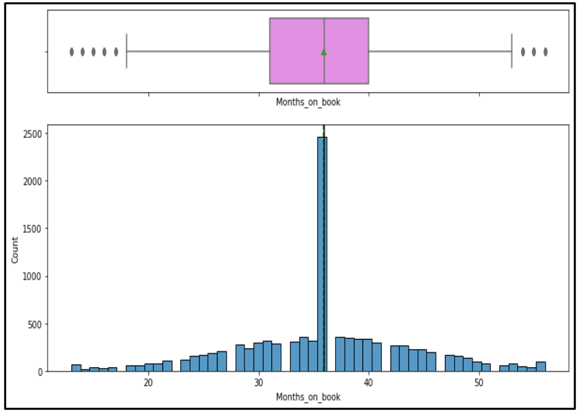
Exploratory Data Analysis – Customer Age, Months on Book, Credit Limit

Customer Age



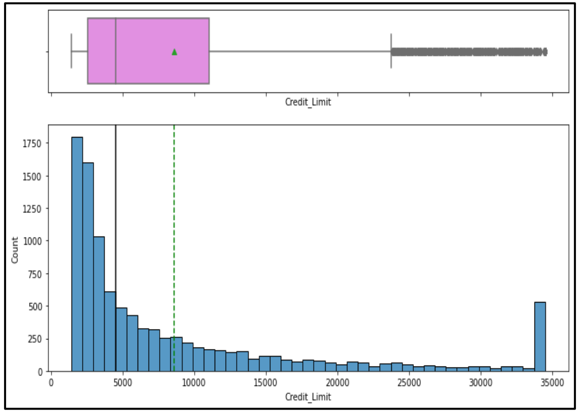
- The distribution of the customer age has a mostly normal distribution.
- The average customer age is around 46.
- The mean and the median are also close together.

Months on Book



- The distribution of the customer age has a mostly normal distribution.
- The average number of months on book is 36 months
- The mean and the median are also close together.

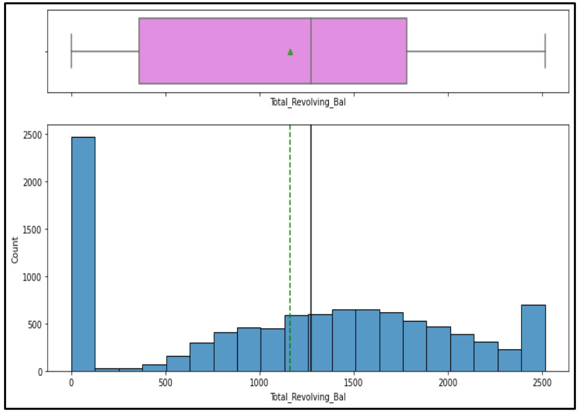
Credit Limit



- The distribution of Credit_Limit is heavily skewed to the right.
- There are outliers to the left of the distribution.

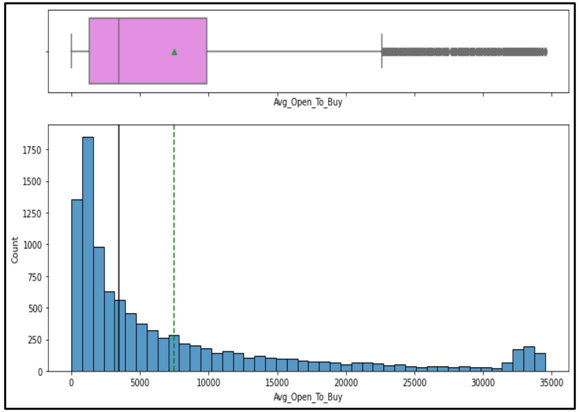
Exploratory Data Analysis – Total Revolving Bal, Average Open to Buy, Total Transaction Amount

Total Revolving Bal



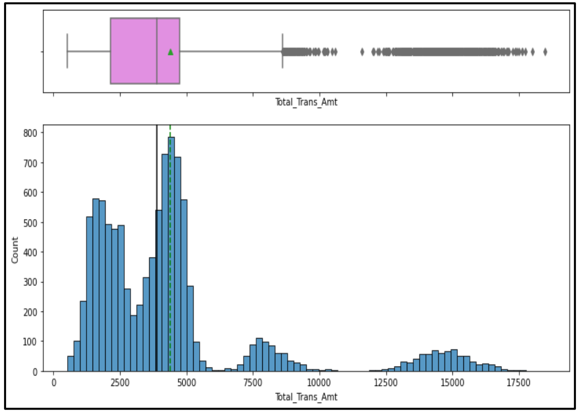
- The distribution of the total revolving balance is skewed heavily on both ends.
- A majority of the data is concentrated around 0.

Average Open to Buy



- The distribution of Avg_Open_To_Buy is heavily skewed to the right.
- There are outliers to the left of the distribution.

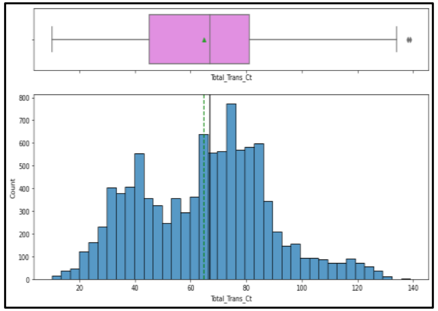
Total Transaction Amount



- The distribution of Total_Trans_Amt appears to be bimodal and heavily skewed to the right.
- There are outliers to the right of the distribution.
- The mean and the median are also close together.

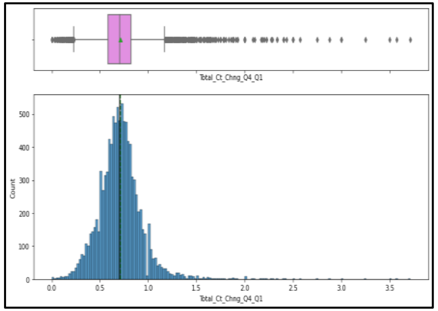
Exploratory Data Analysis – Total Transaction Count, Total Count Change, Total Amount Change, Avg Utilization Ratio

Total Transaction Count



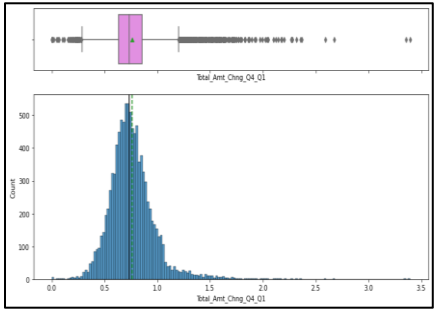
- The distribution of Total_Trans_Ct also appears to be bimodal.
- There are outliers to the right of the distribution.
- The mean and the median are also close together centered around 70.

Total Count Change



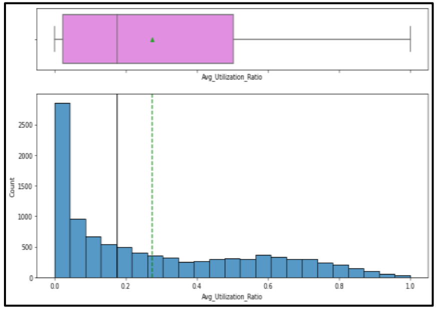
- The distribution of Total_Ct_Chng_Q4_Q1 is normally distributed.
- There are outliers to the left and right of the distribution.
- The mean and the median are also close together.

Total Amount Change



- The distribution of Total_Amt_Chng_Q4_Q1 is normally distributed.
- There are outliers to the left and right of the distribution.
- The mean and the median are also close together.

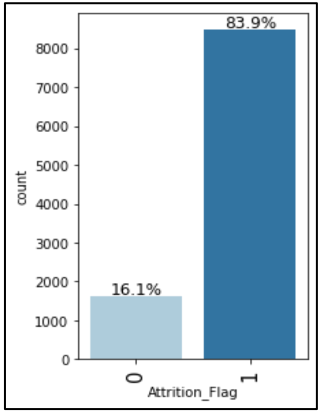
Average Utilization Ratio



- The distribution of the Avg_Utilization_Ratio is skewed heavily to the right.
- A majority of the data is concentrated around 0.

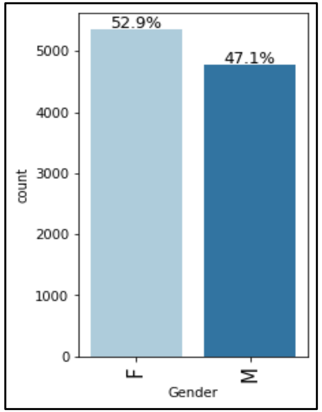
Exploratory Data Analysis – Attrition Flag, Gender, Dependent Count, Educational Level, Marital Status

Attrition Flag



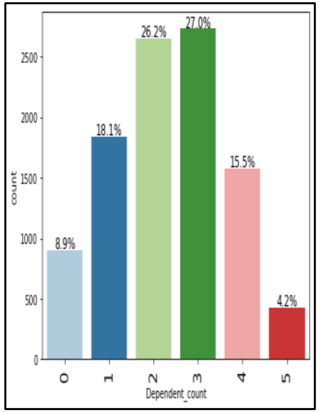
- There are approx 16.1% of customers who renounced their credit cards.

Gender



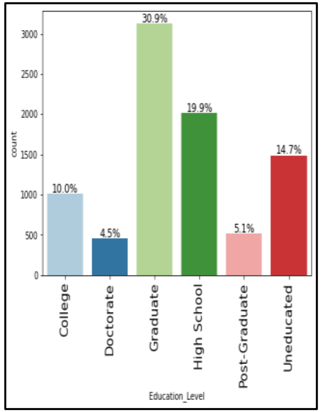
- More customers in this dataset are female at 52.9%.
- This is interesting to consider in the demographics.

Dependent Count



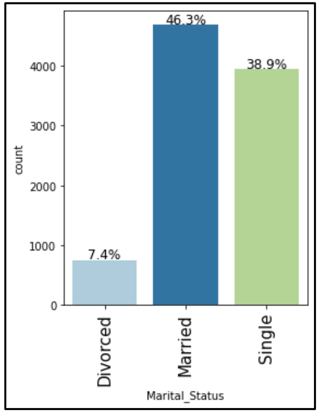
- Most customers have three dependents at 27% with 2 dependents being a close second at 26.2%.
- This shows most customers have a small family.

Education Level



- A majority of customers have a graduate level of education at 30.9%.
- This is interesting to target for potential customers.

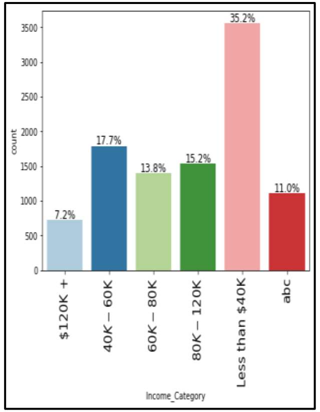
Marital Status



- A majority of customers are either Married or Single at 46.3% and 38.9% respectively.
- Divorced customers made the minority of the population in our data set.

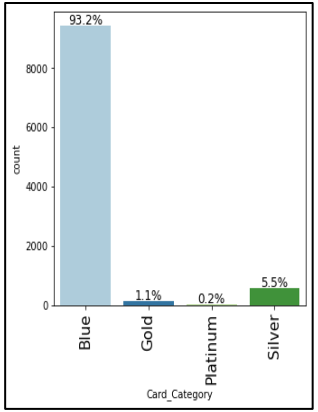
Exploratory Data Analysis – Income Category, Card Category, Total Relationship Count, Months Inactive, Contacts Count

Income Category



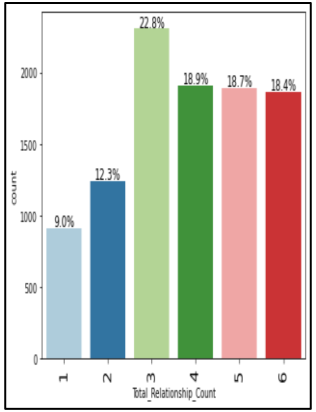
- Most customers make less than $40K income.
- This is interesting to target for potential customers.

Card Category



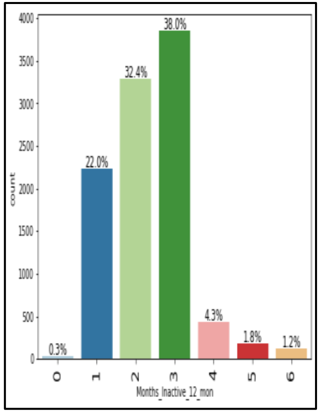
- An overwhelming majority of customers are in the Blue card category.
- This might indicate it is the easiest card to qualify.
- The bank may want to consider if they need to change the qualification criteria of the other cards to distribute the card categories.

Total Relationship Count



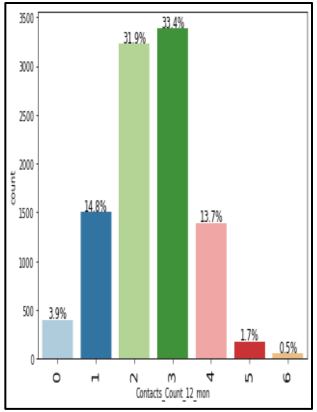
- Most customers have 3 or more products.

Months Inactive



- On average most customers are inactive for 1-3 months.

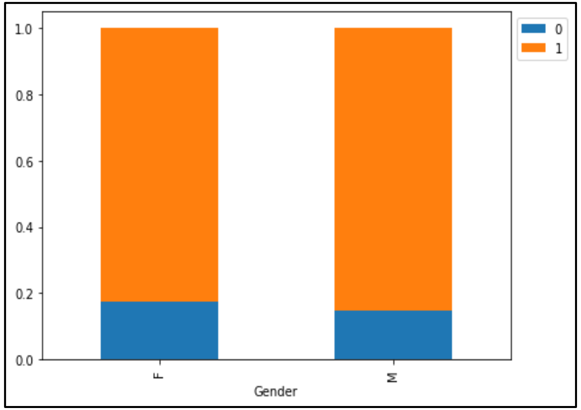
Contacts Count



- Most customers are contacted 2-3 times.

Exploratory Data Analysis – Gender, Dependent count, Education level vs Attrition flag

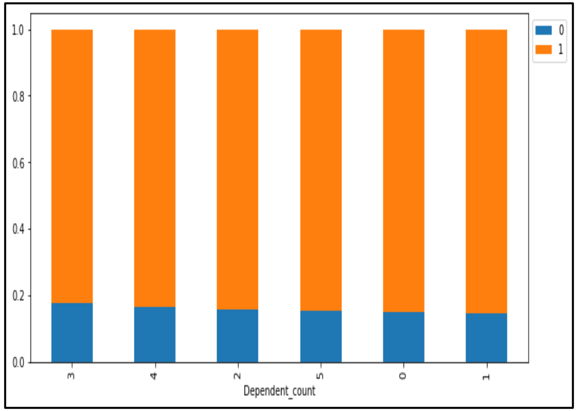
Gender vs Attrition_Flag



Attrition_Flag	0	1	All
Gender			
All	1627	8500	10127
F	930	4428	5358
M	697	4072	4769

- More female customers renounced their credit cards than male customers

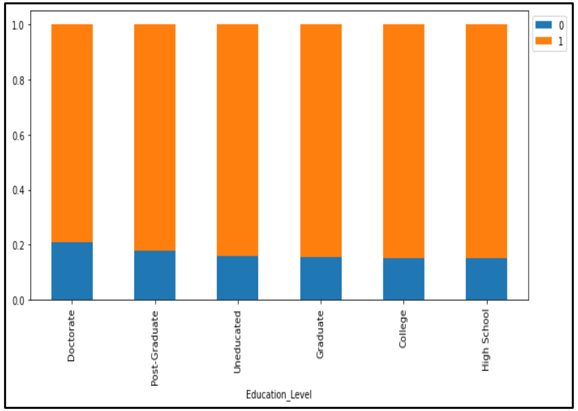
Dependent_count vs Attrition_Flag



Attrition_Flag	0	1	All
Dependent_count			
All	1627	8500	10127
3	482	2250	2732
2	417	2238	2655
1	269	1569	1838
4	260	1314	1574
0	135	769	904
5	64	360	424

- Customers with 2-3 dependents also were the ones who renounced their credic card accounts.

Education_Level vs Attrition_Flag

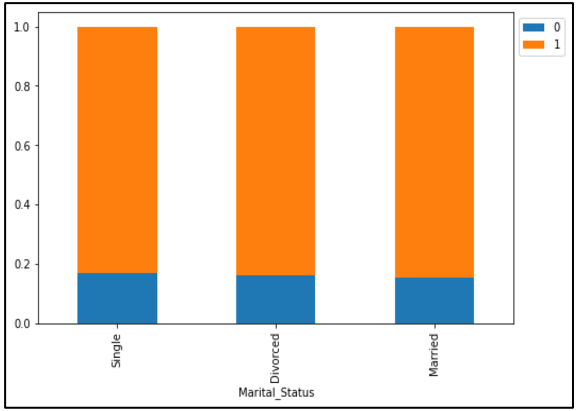


Attrition_Flag	0	1	All
Education_Level			
All	1371	7237	8608
Graduate	487	2641	3128
High School	306	1707	2013
Uneducated	237	1250	1487
College	154	859	1013
Doctorate	95	356	451
Post-Graduate	92	424	516

- Customers with a graduate level of education had the most customers renounce their credit cards with customers with a high school level of education at a close second.

Exploratory Data Analysis – Marital status, Income category, Card category vs Attrition flag

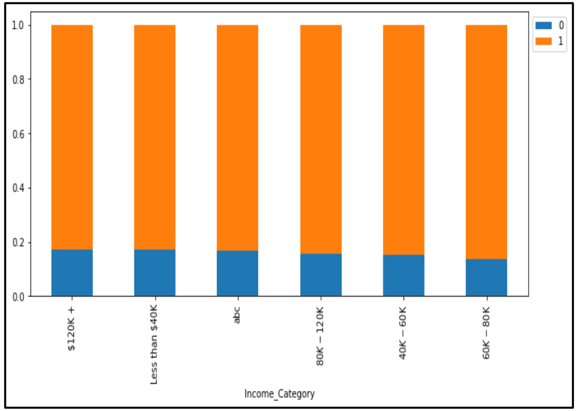
Marital_Status vs Attrition_Flag



Attrition_Flag	0	1	All
Marital_Status			
All	1498	7880	9378
Married	709	3978	4687
Single	668	3275	3943
Divorced	121	627	748

- Married and Single customers had the most cases of attrited customers.

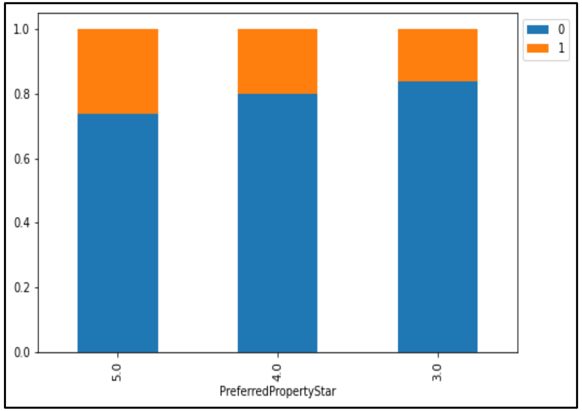
Income_Category vs Attrition_Flag



Attrition_Flag	0	1	All
Income_Category			
All	1627	8500	10127
Less than $40K	612	2949	3561
$40K - $60K	271	1519	1790
$80K - $120K	242	1293	1535
$60K - $80K	189	1213	1402
abc	187	925	1112
$120K +	126	601	727

- A majority of the attrited customers make less than $40k

Card_Category vs Attrition_Flag

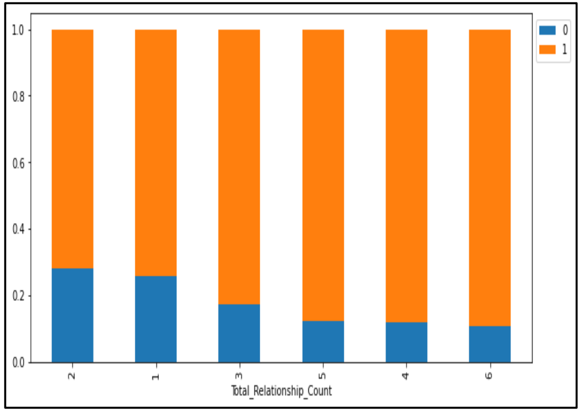


Attrition_Flag	0	1	All
Card_Category			
All	1627	8500	10127
Blue	1519	7917	9436
Silver	82	473	555
Gold	21	95	116
Platinum	5	15	20

- As expected, most customers who attrited had a Blue card
- This makes sense as a majority of the customers in the dataset also have Blue card

Exploratory Data Analysis – Total relationship count, Months inactive, Contacts count vs Attrition flag

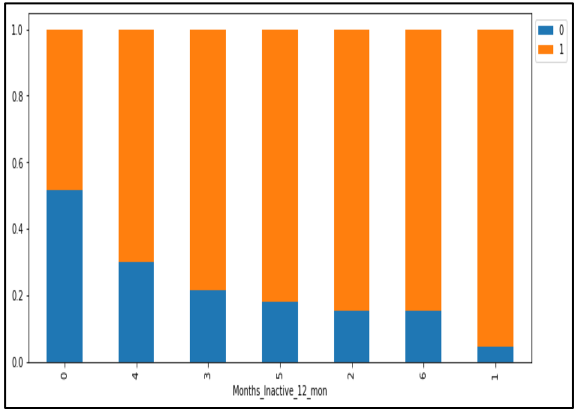
Total_Relationship_Count vs Attrition_Flag



Attrition_Flag Total_Relationship_Count	0	1	All
All	1627	8500	10127
3	400	1905	2305
2	346	897	1243
1	233	677	910
5	227	1664	1891
4	225	1687	1912
6	196	1670	1866

- Customers with 2 or 3 relationships with the bank had higher counts of attrited customers

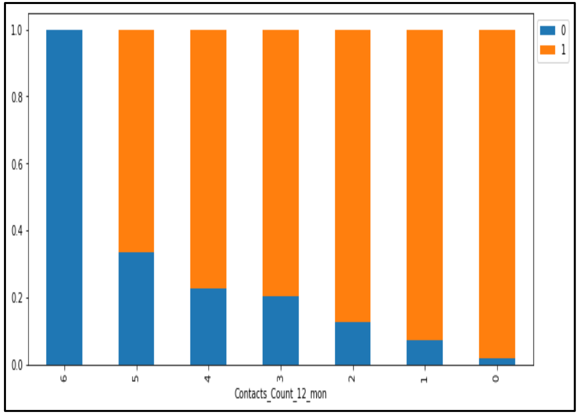
Months_Inactive_12_mon vs Attrition_Flag



Attrition_Flag Months_Inactive_12_mon	0	1	All
All	1627	8500	10127
3	826	3020	3846
2	505	2777	3282
4	130	305	435
1	100	2133	2233
5	32	146	178
6	19	105	124
0	15	14	29

- Customers with 3 inactive months had higher counts of attrited customers

Contacts_Count_12_mon vs Attrition_Flag

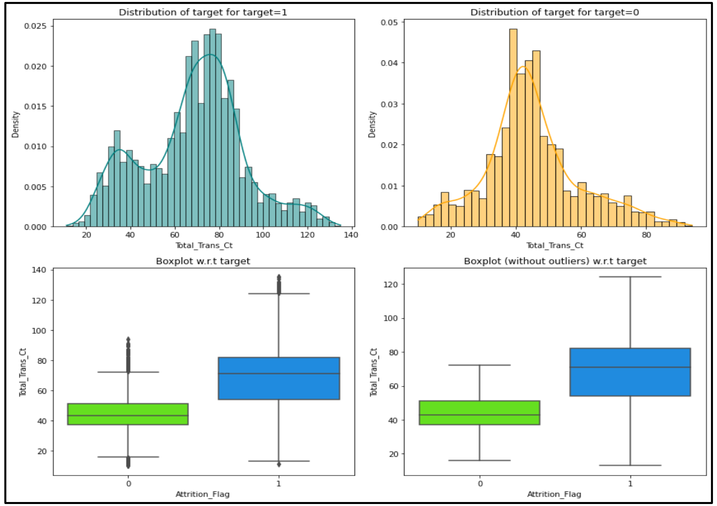


Attrition_Flag Contacts_Count_12_mon	0	1	All
All	1627	8500	10127
3	681	2699	3380
2	403	2824	3227
4	315	1077	1392
1	108	1391	1499
5	59	117	176
6	54	0	54
0	7	392	399

- Customers who were contacted 3 times in the last 12 months had higher counts of attrited customers

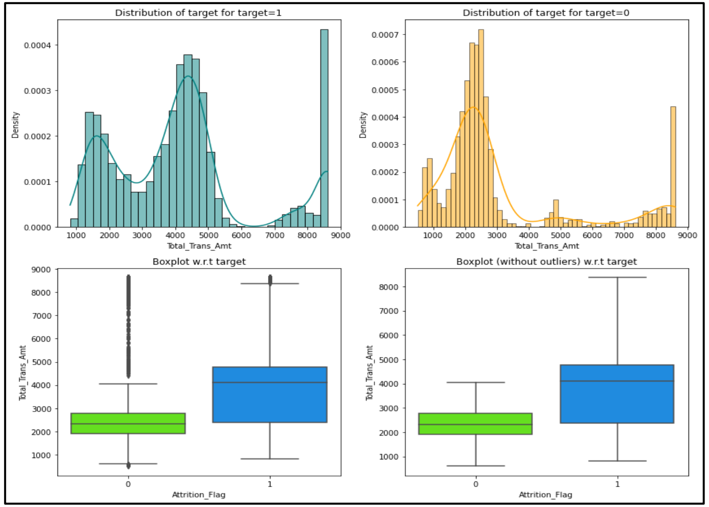
Exploratory Data Analysis

Total_Trans_Ct vs Attrition_Flag



- Customers who have a total transaction count between 0 and ~90 were found to have renounced their credit cards
- The highest concentration of customers who closed their accounts was seen to be betweek 40 and 50 transaction counts

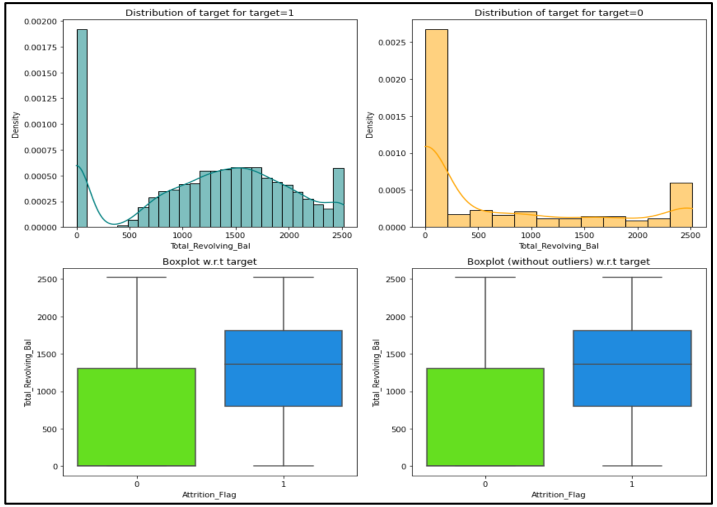
Total_Trans_Amt vs Attrition_Flag



- Total transaction amount between 2k and 3k had the highest concentration of attrited customers
- There were some outliers of attrited customers at about 8.5k total transaction count

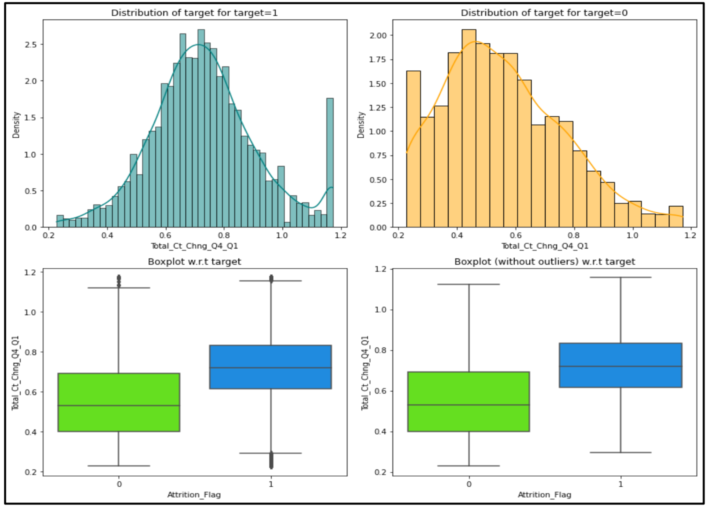
Exploratory Data Analysis

Total_Revolving_Bal vs Attrition_Flag



- The revolving balance for attrited customer had the highest concentration around 0
- The distribution of total revolving balance for attrited customers was mostly uniform aside from the peak at 0 and a small concentration at about 2.5k

Total_Ct_Chng_Q4_Q1 vs Attrition_Flag

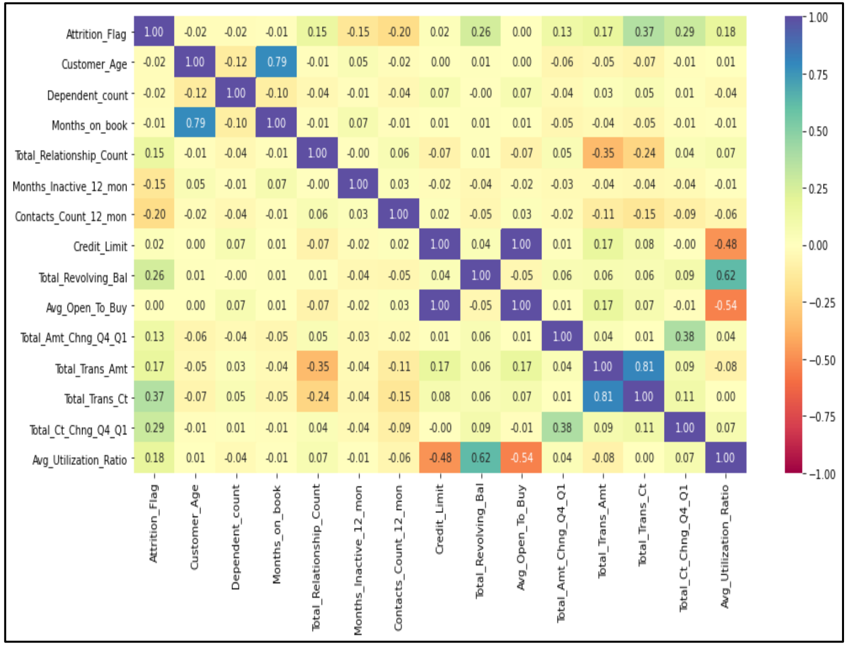


- The distribution of the ratio of the total transaction count in 4th quarter and the total transaction count in 1st quarter compared to attrited customers and existing customers are similar in shape but attrited customer had a slightly lower ratio on average
- Attrited customers mostly on average mostly had a ratio of ~0.4-0.5 while existing customers have a ratio of about 0.7

greatlearning
Power Ahead

Exploratory Data Analysis – Correlation

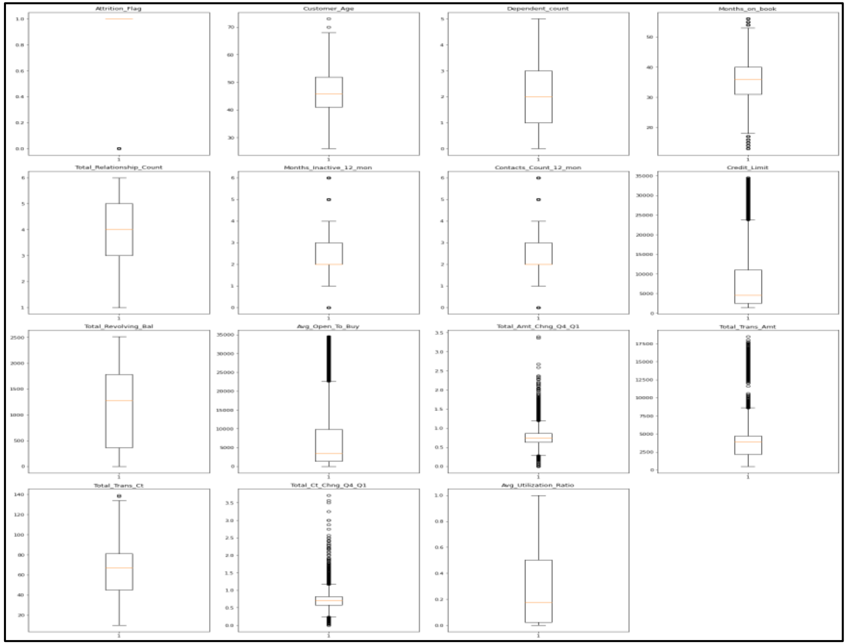
Heat Map Correlation



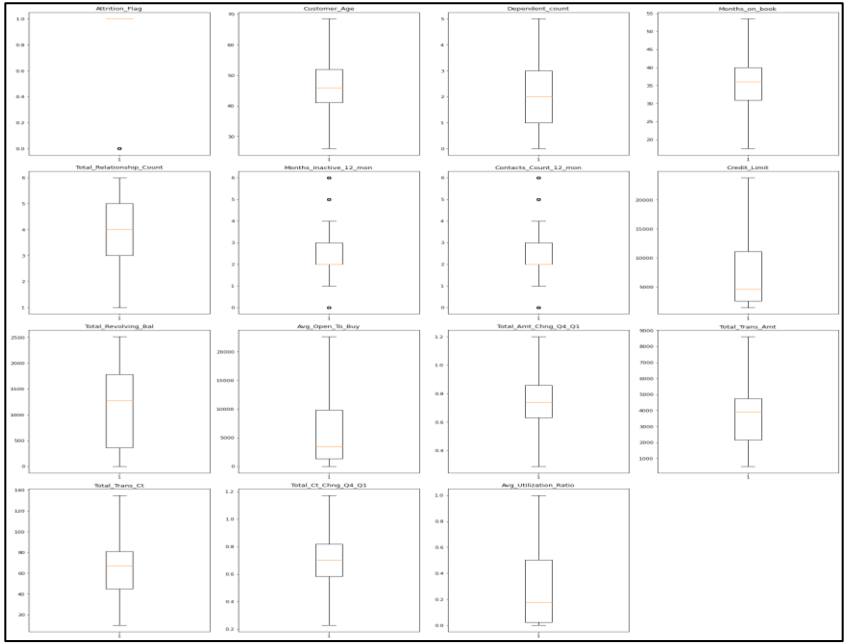
- Avg_Open_To_Buy and Credit_Limit are highly correlated.
- We do also notice some correlation between Total_Trans_Ct and Total_Trans_Amt which makes sense.
- Avg_Utilization_Ratio and Avg_Open_To_Buy have a strong negative correlation of -0.54.

Outlier Treatment

Pre Treatment



Post Treatment



- Eight of our variables appear to have outliers: Customer_Age, Months_on_book, Credit_Limit, Avg_Open_To_Buy, Total_Amt_Chng_Q4_Q1, Total_Trans_Amt, Total_Trans_Ct, and Total_Ct_Chng_Q4_Q1
- We will treat only these eight variable for outliers.
- We will exclude the other variables as they either do not have outliers or they have few values that are distributed.

- All numerical variables that had outliers have been treated.
- Only the target variables have been treated for outliers.

Model Evaluation Criteron

What does a bank want?

A bank wants to minimize the loss - it can face 2 types of losses here:

- A customer who would leave the bank's credit card services.
- A customer not opening credit card services with the bank.

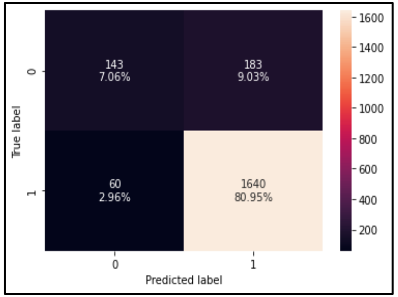
Which loss is greater ?

- A customer who would leave the bank's credit card services.

Since we want to reduce the number off customer attrition we should use Recall as a metric of model evaluation instead of accuracy.

Recall - It gives the ratio of True positives to Actual positives, so high Recall implies low false negatives, i.e. low chances of predicting attrited customer as a non-attrited customer.

Logistic Regression



Training performance:

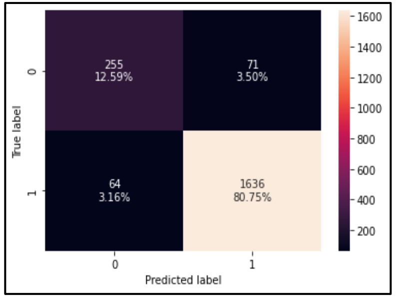
	Accuracy	Recall	Precision	F1
0	0.888066	0.964503	0.907883	0.935337

Validation performance:

	Accuracy	Recall	Precision	F1
0	0.880059	0.964706	0.899616	0.931025

- Logistic Regression has given a generalized performance on training and validation set.
- This is a strong model giving us comparable scores between training and validation set.
- Accuracy and precision are ok but may be improved.

Decision Tree Classifier



Training performance:

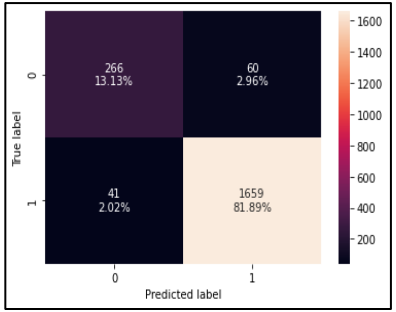
	Accuracy	Recall	Precision	F1
0	1.0	1.0	1.0	1.0

Validation performance:

	Accuracy	Recall	Precision	F1
0	0.933366	0.962353	0.958407	0.960376

- The model is not necessarily overfitting as the validation performance is close to the training performance but the training performance is has no variation. As such this model will not be trusted.

Bagging Classifier



Training performance:

	Accuracy	Recall	Precision	F1
0	0.997202	0.99745	0.999214	0.998332

Validation performance:

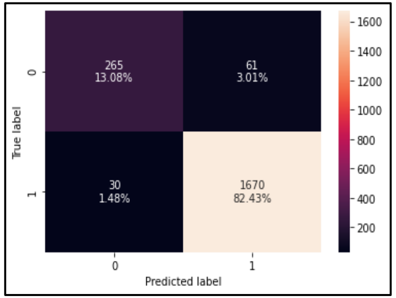
	Accuracy	Recall	Precision	F1
0	0.950148	0.975882	0.965096	0.970459

- With default parameters, the bagging classifier is performing well overall.
- The model is not overfitting the data.

Model Evaluation Cont

greatlearning
Power Ahead

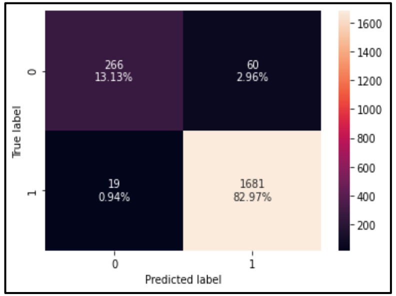
AdaBoost Classifier




```
Training performance:
   Accuracy    Recall  Precision        F1
0  0.961317  0.982546   0.971683  0.977084
Validation performance:
   Accuracy    Recall  Precision        F1
0  0.955084  0.982353    0.96476  0.973477
```

- The model is not overfitting the data and the overall performance is good.
- Both test and validation set are performing well.

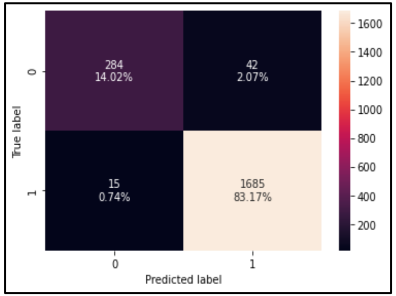
Gradient Boosting Classifier




```
Training performance:
   Accuracy    Recall  Precision       F1
0   0.97893  0.993136   0.981966  0.98752
Validation performance:
   Accuracy    Recall  Precision        F1
0  0.961007  0.988824   0.965537  0.977042
```

- Overall model performance is pretty good.
- Performance on both validation and training performance is comparable.

XGBoost Classifier




```
Training performance:
   Accuracy  Recall  Precision   F1
0       1.0     1.0        1.0  1.0
Validation performance:
   Accuracy    Recall  Precision        F1
0  0.971866  0.991176    0.97568  0.983367
```

- The model is not necessarily overfitting as the validation performance is close to the training performance but the training performance is has no variation. Similar to the performance of the decision tree model, this model will not be trusted.

Model Evaluation Comparison

greatlearning
Power Ahead

Training performance comparison:

	Logistic Regression	Decision Tree	Bagging Classifier	AdaBoost Classifier	Gradient Boosting Classifier	XGBoost Classifier
Accuracy	0.888066	1.0	0.997202	0.961317	0.978930	1.0
Recall	0.964503	1.0	0.997450	0.982546	0.993136	1.0
Precision	0.907883	1.0	0.999214	0.971683	0.981966	1.0
F1	0.935337	1.0	0.998332	0.977084	0.987520	1.0

Validation performance comparison:

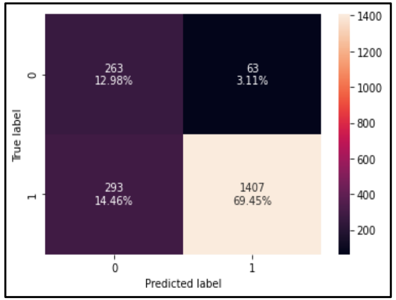
	Logistic Regression	Decision Tree	Bagging Classifier	AdaBoost Classifier	Gradient Boosting Classifier	XGBoost Classifier
Accuracy	0.880059	0.933366	0.950148	0.955084	0.961007	0.971866
Recall	0.964706	0.962353	0.975882	0.982353	0.988824	0.991176
Precision	0.899616	0.958407	0.965096	0.964760	0.965537	0.975680
F1	0.931025	0.960376	0.970459	0.973477	0.977042	0.983367

Conclusions:

- Overall all models performed well.
- Decision tree and XGBoost training performance were perfect across the board which is not what we want in our model.
- AdaBoost Classifier, Bagging Classifier, and Gradient Boosting Classifier had the strongest overall models in terms of fit and performance.

Oversampling train data using SMOTE

Logistic Regression



Training performance:

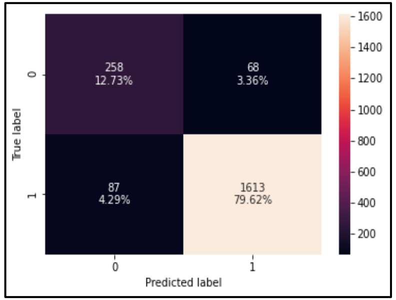
	Accuracy	Recall	Precision	F1
0	0.827319	0.818396	0.833267	0.825764

Validation performance:

	Accuracy	Recall	Precision	F1
0	0.824284	0.827647	0.957143	0.887697

- Model has given a generalized performance on training and validation set.
- Overall model performance is lower in all aspects.
- Would consider using original model over oversampled model.

Decision Tree



Training performance:

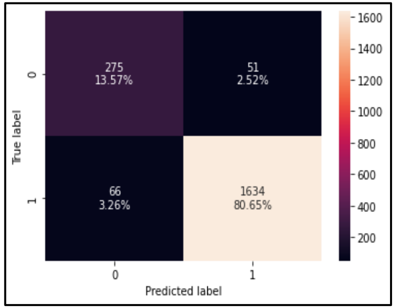
	Accuracy	Recall	Precision	F1
0	1.0	1.0	1.0	1.0

Validation performance:

	Accuracy	Recall	Precision	F1
0	0.923495	0.948824	0.959548	0.954156

- The model is not necessarily overfitting as the validation performance is close to the training performance but the training performance is has no variation. As such this model will not be trusted.

Bagging Classifier



Training performance:

	Accuracy	Recall	Precision	F1
0	0.997058	0.995293	0.998819	0.997053

Validation performance:

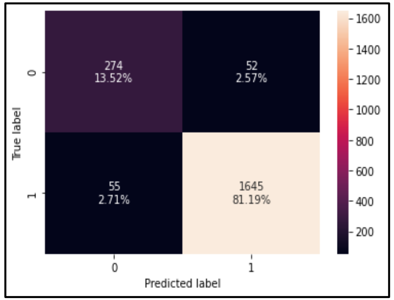
	Accuracy	Recall	Precision	F1
0	0.942251	0.961176	0.969733	0.965436

- Model performance overall is doing well on the training data
- Validation performance is slightly lower than the training data.
- Overall the normal bagging model performance is better than over sampled model.

greatlearning
Power Ahead

Oversampling train data using SMOTE Cont

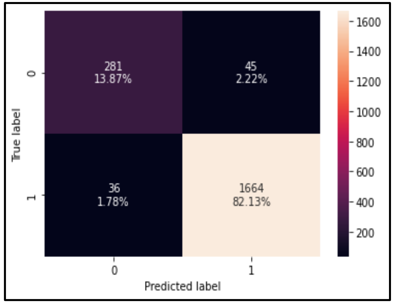
AdaBoost




```
Training performance:
    Accuracy  Recall  Precision        F1
0    0.9697  0.96921  0.970161  0.969685
Validation performance:
    Accuracy    Recall  Precision        F1
0  0.947187  0.967647   0.969358  0.968502
```

- The model is not overfitting the data and the overall performance is good.
- Both test and validation set are performing well although accuracy on the validation set is slightly lower compared to the rest of the performance.
- Normal model is performing slightly better than the oversampled model.

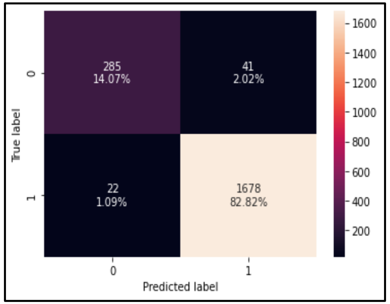
Gradient Boosting Classifier




```
Training performance:
   Accuracy    Recall  Precision        F1
0  0.983428  0.983134   0.983713  0.983423
Validation performance:
   Accuracy    Recall  Precision        F1
0   0.96002  0.978824   0.973669  0.976239
```

- Overall model performance is pretty good.
- Performance on both validation and training performance is comparable.
- Model performance is just as good as performance on the normal dataset.

XGBoost Classifier




```
Training performance:
   Accuracy  Recall  Precision   F1
0       1.0     1.0        1.0  1.0
Validation performance:
   Accuracy    Recall  Precision        F1
0  0.968904  0.987059   0.976149  0.981574
```

- The model is not necessarily overfitting as the validation performance is close to the training performance but the training performance is has no variation. Similar to the performance of the decision tree model, this model will not be trusted.

Oversampled Model Evaluation Comparison

```
Over Sampled Training performance comparison:
```

	Logistic Regression	Decision Tree	Bagging Classifier	AdaBoost Classifier	Gradient Boosting Classifier	XGBoost Classifier
Accuracy	0.827319	1.0	0.997058	0.969700	0.983428	1.0
Recall	0.818396	1.0	0.995293	0.969210	0.983134	1.0
Precision	0.833267	1.0	0.998819	0.970161	0.983713	1.0
F1	0.825764	1.0	0.997053	0.969685	0.983423	1.0

```
Over Sampled Validation performance comparison:
```

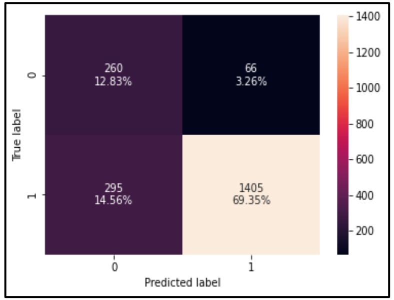
	Logistic Regression	Decision Tree	Bagging Classifier	AdaBoost Classifier	Gradient Boosting Classifier	XGBoost Classifier
Accuracy	0.824284	0.923495	0.942251	0.947187	0.960020	0.968904
Recall	0.827647	0.948824	0.961176	0.967647	0.978824	0.987059
Precision	0.957143	0.959548	0.969733	0.969358	0.973669	0.976149
F1	0.887697	0.954156	0.965436	0.968502	0.976239	0.981574

Conclusions:

- Overall all oversampled models performed well but slightly lower in general compared to the normal models.
- Decision tree and XGBoost training performance were perfect across the board which is not what we want in our model.
- Bagging Classifier and Gradient Boosting Classifier had the strongest overall models in terms of fit and performance.

Undersampling train data using RandomUnderSampler

Logistic Regression



Training performance:

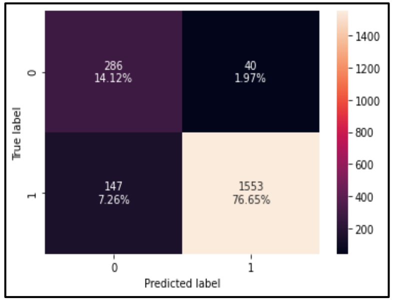
	Accuracy	Recall	Precision	F1
0	0.80584	0.803279	0.807415	0.805342

Validation performance:

	Accuracy	Recall	Precision	F1
0	0.821816	0.826471	0.955133	0.886156

- Model appears to be underfitting on the validation set slightly.
- Model performance has not improved using downsampling.
- We maintain that the normal model is the better model.

Decision Tree



Training performance:

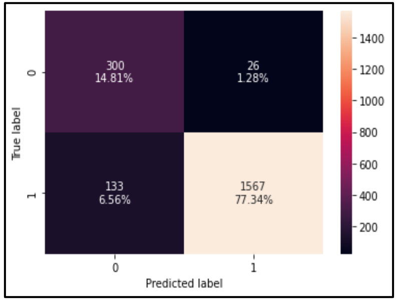
	Accuracy	Recall	Precision	F1
0	1.0	1.0	1.0	1.0

Validation performance:

	Accuracy	Recall	Precision	F1
0	0.9077	0.913529	0.97489	0.943213

- The model is not necessarily overfitting as the validation performance is close to the training performance but the training performance is has no variation. As such this model will not be trusted.

Bagging Classifier



Training performance:

	Accuracy	Recall	Precision	F1
0	0.996926	0.993852	1.0	0.996917

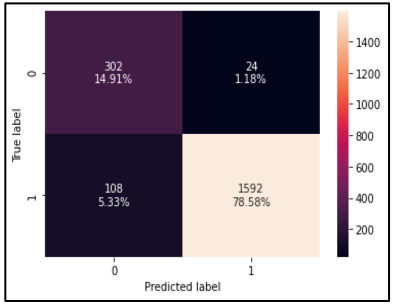
Validation performance:

	Accuracy	Recall	Precision	F1
0	0.92152	0.921765	0.983679	0.951716

- Model performance overall is doing well on the training data
- Validation performance is slightly lower than the training data, especially in accuracy and recall.
- Overall the normal bagging model performance is better than over sampled model.

Undersampling train data using RandomUnderSampler

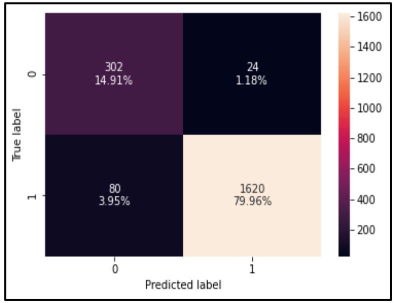
AdaBoost




```
Training performance:
    Accuracy    Recall  Precision        F1
0  0.947234  0.941598   0.952332  0.946935
Validation performance:
    Accuracy    Recall  Precision        F1
0  0.934847  0.936471   0.985149  0.960193
```

- The model is not overfitting the data and the overall performance is good.
- Overall model performance is lower than the oversampled and normal model.

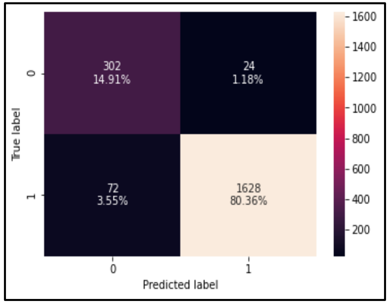
Gradient Boosting Classifier




```
Training performance:
    Accuracy    Recall  Precision        F1
0    0.97541  0.971311   0.979339  0.975309
Validation performance:
    Accuracy    Recall  Precision      F1
0  0.948667  0.952941   0.985401  0.9689
```

- Overall model performance is pretty good.
- Performance on both validation and training performance is lower than the previous two models.
- This model will likely not be considered as a strong model.

XGBoost Classifier




```
Training performance:
    Accuracy  Recall  Precision   F1
0        1.0     1.0        1.0  1.0
Validation performance:
    Accuracy    Recall  Precision       F1
0  0.952616  0.957647   0.985472  0.97136
```

- The model is is not necesserily overfitting as the validation performance is close to the training performance but the training performance is has no variation. Similar to the performance of the decision tree model, this model will not be trusted.

Undersampled Model Evaluation Comparison

```
Under Sampled Training performance comparison:
```

	Logistic Regression	Decision Tree	Bagging Classifier	AdaBoost Classifier	Gradient Boosting Classifier	XGBoost Classifier
Accuracy	0.805840	1.0	0.996926	0.947234	0.975410	1.0
Recall	0.803279	1.0	0.993852	0.941598	0.971311	1.0
Precision	0.807415	1.0	1.000000	0.952332	0.979339	1.0
F1	0.805342	1.0	0.996917	0.946935	0.975309	1.0

	Logistic Regression	Decision Tree	Bagging Classifier	AdaBoost Classifier	Gradient Boosting Classifier	XGBoost Classifier
Accuracy	0.821816	0.907700	0.921520	0.934847	0.948667	0.952616
Recall	0.826471	0.913529	0.921765	0.936471	0.952941	0.957647
Precision	0.955133	0.974890	0.983679	0.985149	0.985401	0.985472
F1	0.886156	0.943213	0.951716	0.960193	0.968900	0.971360

Conclusions:

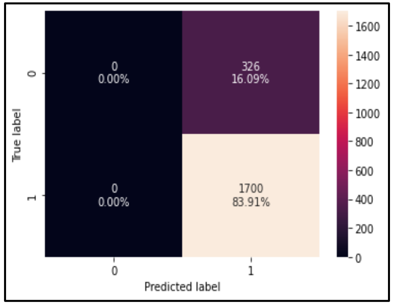
- Overall all undersampled models performed lower than the normal models.
- Decision tree and XGBoost training performance were perfect across the board which is not what we want in our model.
- AdaBoost Classifier and Gradient Boosting Classifier had the strongest overall models in terms of fit and performance.

Model Selection and Hyper Parameter Tuning

Model Selection

- Based on the performance of the models above, we have selected to focus on the three top performing models
- Based on our observations, the three models we feel performed the best to tune are Bagging Classifier, AdaBoost Classifier, and Gradient Boosting Classifier using the normal data.
- Performance of the under and over sampled data were comparable to the normal model performance but slightly lower across the board.

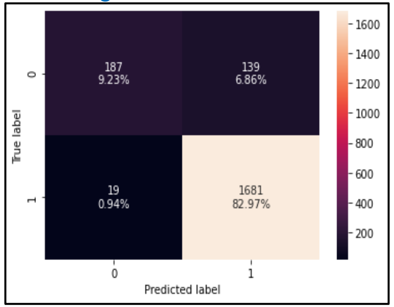
Tuning Bagging Classifier




```
Training performance:
   Accuracy  Recall  Precision        F1
0  0.839342     1.0   0.839342  0.912654
Test performance:
   Accuracy  Recall  Precision        F1
0  0.839585     1.0   0.839585  0.912798
Validation performance:
   Accuracy  Recall  Precision        F1
0  0.839092     1.0   0.839092  0.912507
```

- Overall model performance decreased for accuracy and precision after tuning the model
- The recall for all three models is 1.0 which is unusual; this will likely not be our final model

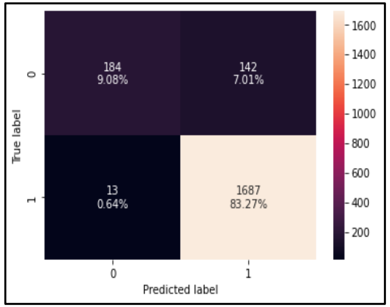
Tuning AdaBoost Classifier




```
Training performance:
   Accuracy    Recall  Precision        F1
0  0.931358  0.988821   0.933358  0.960289
Test performance:
   Accuracy    Recall  Precision        F1
0  0.923495  0.989418   0.924725  0.955978
Validation performance:
   Accuracy    Recall  Precision        F1
0  0.922014  0.988824   0.923626  0.955114
```

- Overall model performance increased slightly with very good recall and accuracy
- We would potentially consider this to be our final model

Tuning Gradient Boosting Classifier




```
Training performance:
   Accuracy    Recall  Precision        F1
0  0.931029  0.992744   0.929831  0.960258
Test performance:
   Accuracy    Recall  Precision        F1
0  0.919052  0.992357    0.91789  0.953672
Validation performance:
   Accuracy    Recall  Precision        F1
0  0.923495  0.992353   0.922362  0.956078
```

- Overall model performance increased slightly with very good recall and accuracy
- Accuracy and recall was very similar to the tuned AdaBoost Classifier model

Tuned Model Evaluation Comparison

greatlearning Power Ahead

```
Tuned training performance comparison:
```

	Tuned Bagging Classifier	Tuned AdaBoost Classifier	Tuned Gradient Boosting Classifier
Accuracy	0.839342	0.931358	0.931029
Recall	1.000000	0.988821	0.992744
Precision	0.839342	0.933358	0.929831
F1	0.912654	0.960289	0.960258

```
Tuned test performance comparison:
```

	Tuned Bagging Classifier	Tuned AdaBoost Classifier	Tuned Gradient Boosting Classifier
Accuracy	0.839585	0.923495	0.919052
Recall	1.000000	0.989418	0.992357
Precision	0.839585	0.924725	0.917890
F1	0.912798	0.955978	0.953672

```
Tuned validation performance comparison:
```

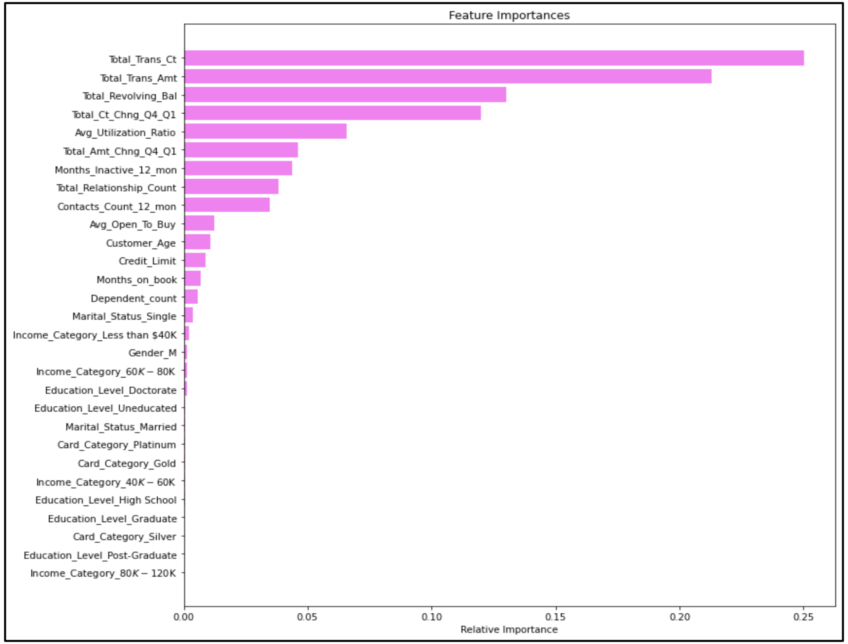
	Tuned Bagging Classifier	Tuned AdaBoost Classifier	Tuned Gradient Boosting Classifier
Accuracy	0.839092	0.922014	0.923495
Recall	1.000000	0.988824	0.992353
Precision	0.839092	0.923626	0.922362
F1	0.912507	0.955114	0.956078

Conclusions:

- Overall performance of all three models were very good.
- The tuned bagging classifier had overall the lowest performance but a validation recall of ~1.00.
- Out of the three the best model that gave the best performance between AdaBoost and Gradient Boosting Classifiers is the Tuned Gradient Boosting Classifier

Tuned Gradient Boosting Classifier Feature Importance

Tuned Gradient Boosting Classifier Feature Importance



- Here we can see from the feature importance of our model that the transaction data and revolving balance are the three variables of highest importance of determining if a customer will renounce their credit cards.

Conclusion

After all the analysis, we have been able to conclude:

- The CLIENTNUM attribute does not add any information to our analysis as all the values are unique. There is no association between a person's customer ID and Attrition_Flag, also it does not provide any general conclusion for future potential travel package customers. We can neglect this information for our model prediction.
- We notice a couple values that need to be treated in 2 different columns which were Education_Level and Marital_Status.
- There were a few variables that were heavily skewed that needed to be treated for outliers.
- Here we can see from the feature importance of our model that the transaction data and revolving balance are the three variables of highest importance of determining if a customer will renounce their credit cards.
- We noticed several features that appeared to correlate to customer attrition during EDA.
- Based on feature importance they mostly did correlate to what we saw during our EDA.
- The top features of importance were related to the customer’s transaction data.
- After exploring several models with different samples of data, we were able to find a strong model in the gradient boosting classifier.
- With some tuning, we were able to improve the model performance slightly to make it more reliable.
- Once we had our model selected and tuned we productionized the model using pipelines.

Recommendations

After all the analysis, we suggest the following recommendations:

- The best model's test recall is ~99% and the test precision is ~91%. This means that the model is good at identifying attrited customers, therefore, the bank can more accurately identify potential customers who might renounce their credit cards.
- The model performance can be improved if wanted to improve the precision and accuracy, although both were very good at ~91% each.
- We saw in our analysis that customers who have about 40-50 transaction counts are more likely to renounce their credit cards. The bank can offer a plan that can cater to customers who will spend in this range.
- Total amount spent around 2k and 3k had the highest concentration of attrited customers. The bank can offer a lower credit limit card for customers who may be low spenders.
- We saw that customers who had a revolving balance of 0 also were more likely to renounce their credit card. The bank should be more strict on a month to month basis to flag these customers.
- Our analysis showed that customers with a lower transaction ration were more likely to renounce their credit cards. On average a ratio of ~0.4-0.5 while existing costumers have a ratio of about 0.7 The bank can alter its policies to suppress this or also monitor this as a metric for customers who might be thinking of renouncing their credit cards.
- The bank can set up metrics for transaction data to further drill down and flag customers that are at risk of renouncing their credit cards.
- Based on the feature importance breakdown, the bank can potentially create a custom card category for customers that are at risk for renouncing their credit cards with the bank based on what their need and usage will be.

greatlearning

Power Ahead

Happy Learning !