Project Report:
Predict Credit Card Customer Churn

Overview

I set out to develop a predictive model to identify the likelihood existing credit card customers will churn. A range of available customer history features were utilized to train and test a variety of models.

The final CatBoost Classifier model was able to use 70% of all customer data to train a model capable of identifying customers that would churn with 96% accuracy, 97.1% recall and 81.4% precision on the validation dataset. Out of the 488 churned customers in the validation dataset, only 14 were misclassified. Additionally, while 108 of the 582 predicted churns were misclassified, this produces a population of high risk customers most likely to churn. The resulting model possesses strong predictive power to identify most customers likely to churn, without significantly inflating the size of the class.

Problem

About 16% of credit card customers in an existing portfolio have churned over the past year. The portfolio manager needs to understand which customers are most likely to churn so that he can take proactive measures to mitigate churn and reduce the rate below 10%.

Approach

Data wrangling

The credit card portfolio contains 10,127 customers. 8,500 are existing customers and 1,627 customers have churned. The dataset was very clean, with no null values. All customer data was maintained for testing and training purposes.

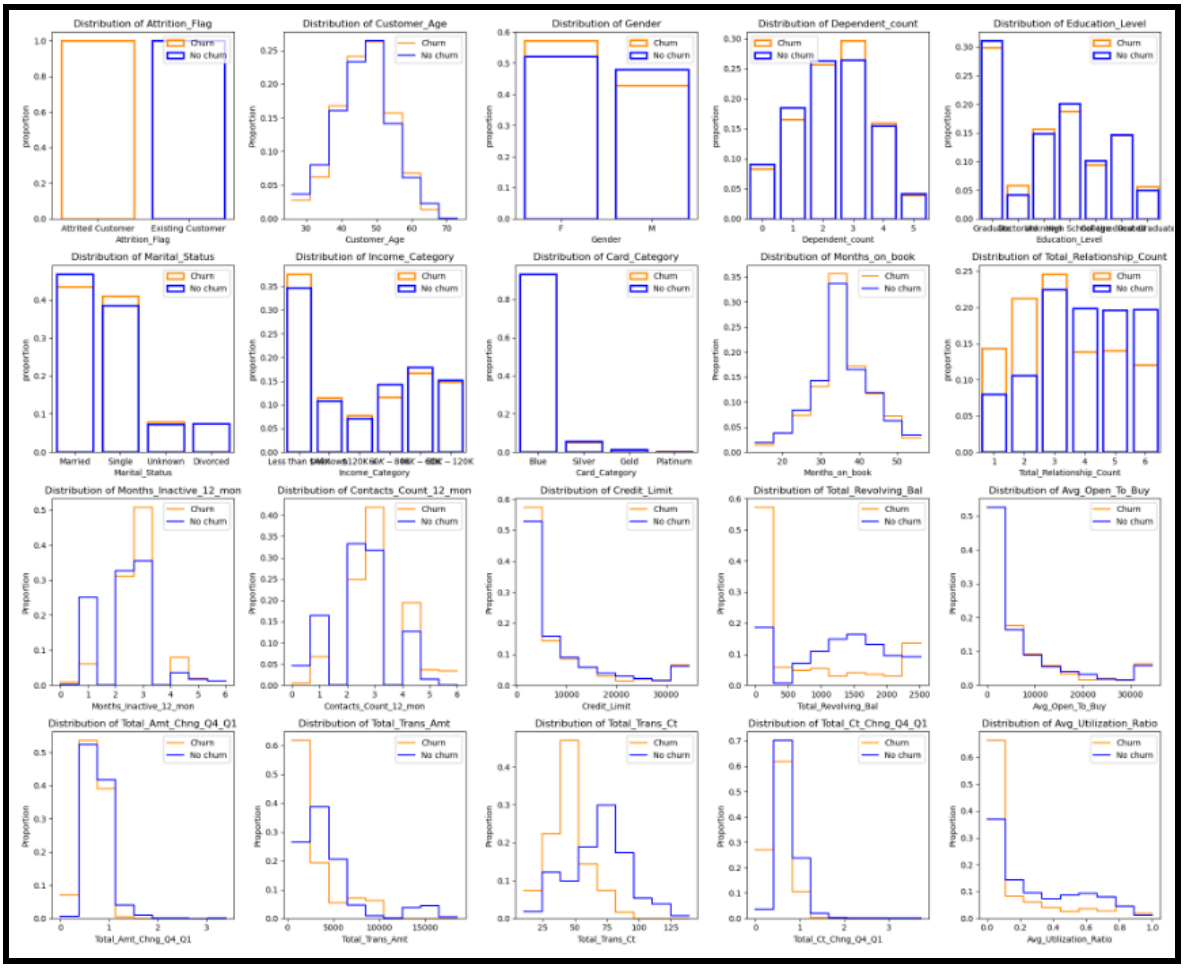
This data snapshot includes historical customer data from this point in time, features listed below:

Churn - flag indicating if customer relationship still exists
Age - customers age in years
Gender - male or female
Dependent_count - number of dependents
Education_Level - educational qualification of the account holder
Marital_Status - categorized as married, single, divorced, unknown

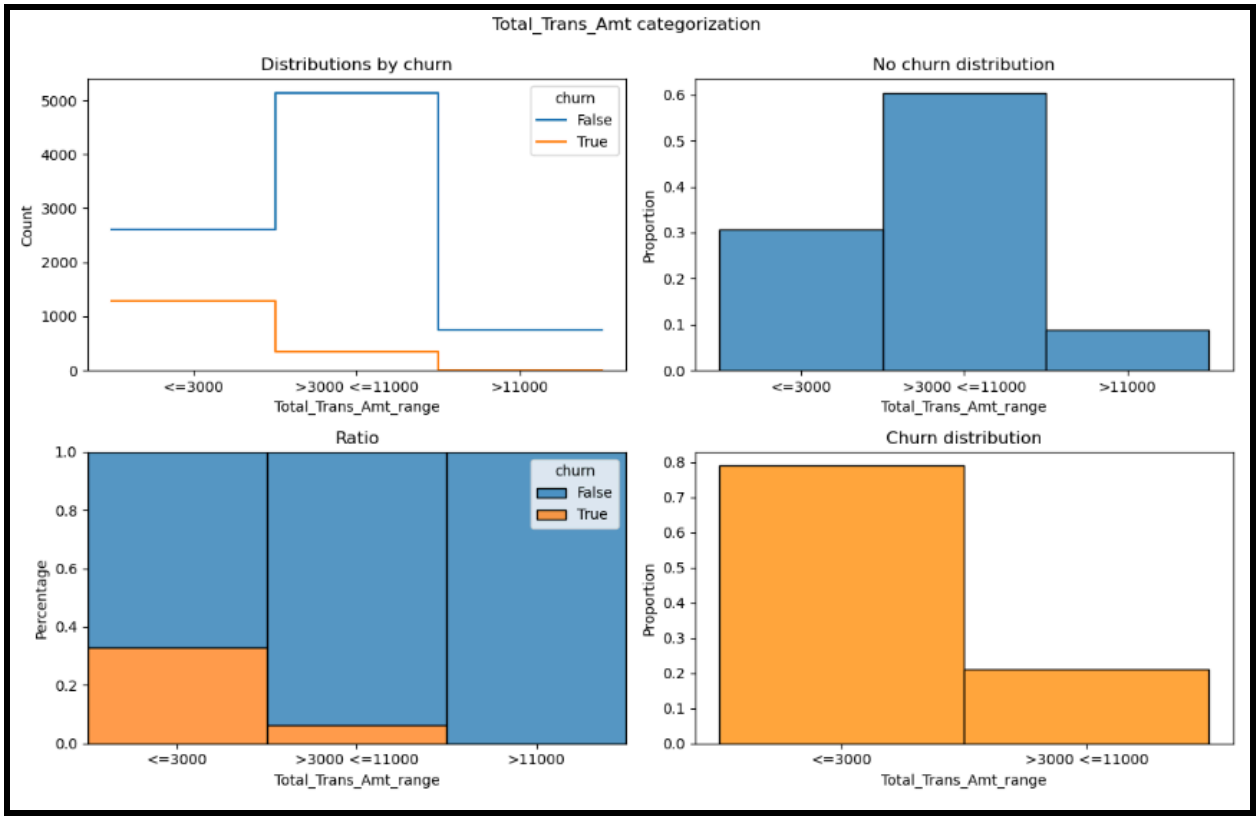
Income_Category - range of annual income
Card_Category - type of card
Total_Relationship_Count - total number of products held by customer
Months_on_book - months of relationship with company
Months_Inactive_12_mon - number of months inactive in the last 12 months
Contacts_Count_12_mon - number of contacts in the last 12 months
Credit_Limit - credit limit on the credit card
Total_Revolving_Bal - total revolving balance on the credit card
Avg_Open_To_Buy - open to buy credit line (average of last 12 months)
Total_Trans_Amt - total amount from transactions over the last 12 months
Total_Amt_Chng_Q4_Q1 - change in transaction amount (Q4 over Q1)
Total_Trans_Ct - total count of transactions over the last 12 months
Total_Ct_Chng_Q4_Q1 - change in transaction count (Q4 over Q1)
Avg_Utilization_Ratio - utilization ratio average over the last 12 months

Exploratory data analysis

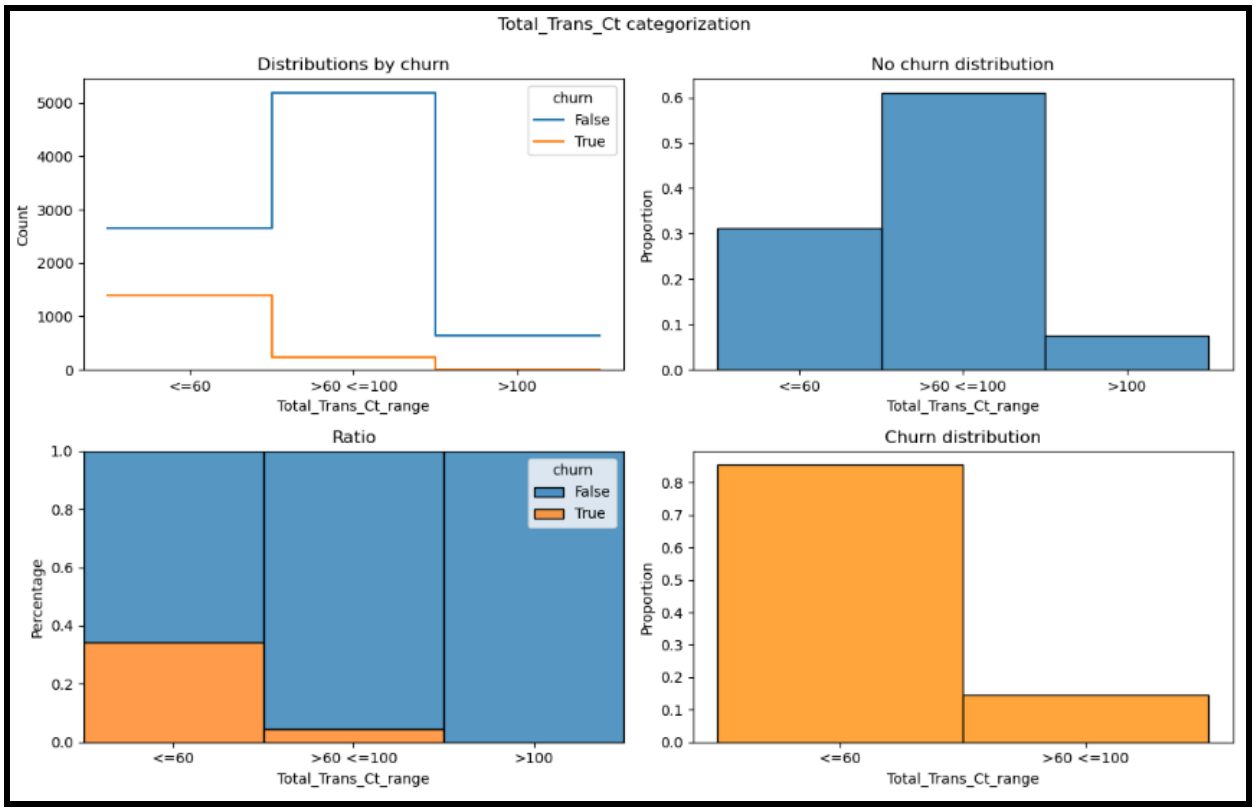
The first step was to review the proportional distribution of all features for the churn and non-churn populations.



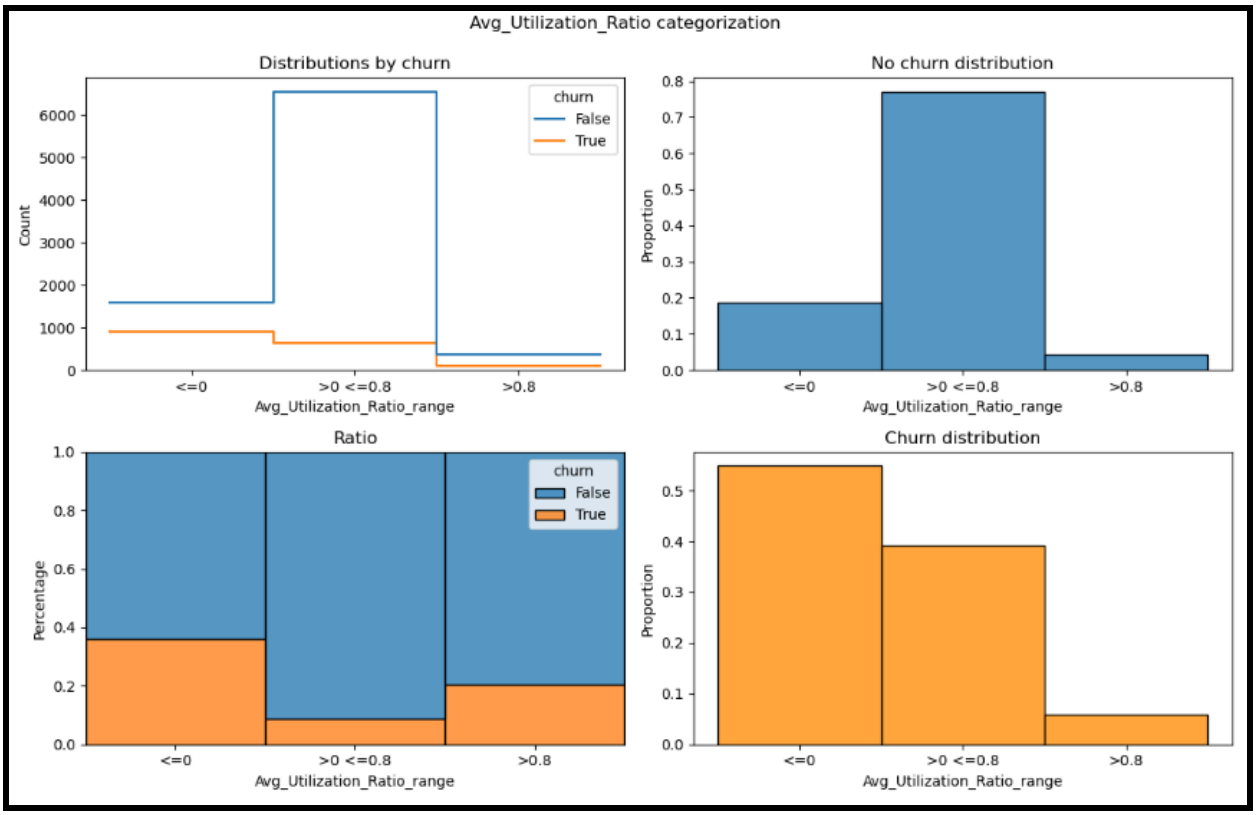
Upon review, there were certain features with visible differences in the distribution between the churn and non-churn populations. In particular, transaction amount, transaction count, and average utilization ratio appears to show the largest distinctions between churn and non-churn.



Transaction amounts show a clear skewed right churn population. As the total transaction amount increases, the proportion of churned customers decreases. All churned customers are below 11,000 in total transaction amount, while nearly 10% of existing customers exceed that threshold. High total transaction amounts appear to correlate with customers remaining with the company.



Transaction count also shows a clear skewed right churn population. As the total transaction count increases, the proportion of churned customers decreases. All churned customers are below 100 in total transactions, while nearly 10% of existing customers exceed that threshold. High total transaction counts appear to correlate with customers remaining with the company.



The average utilization ratio also showed interesting population trends. The churn population showed a clear skewed right churn population. However the proportion of churned to non-churned customers is lowest in the middle of the utilization range. The ratio of churned to non-churned customers is closer to 30% on the extremes, while only around 10% in the middle. Moderate levels of utilization appear to correlate with customers remaining with the company. Customers with no usage or maximizing as least 80% of their credit line have been more likely to churn.

Preprocessing

The dataset includes several categorical features. Before running the data through the model, one hot encoding was required to expand each categorical feature into unique columns for each value found in the feature. The expanded the dataset from 18 to 31 features for model training.

Before beginning any training, 30% of the dataset was randomly selected and set aside for validation. The remaining 70% was used to assess the performance of a variety of models.

Model selection

To measure which model would be most effective, the first step required was to define key metrics. The primary objective is to identify the population of customers that are likely to churn. I do not want to miss a customer that is likely to churn (false negatives), and can handle including a reasonable number of customers in this population that are not actually likely to churn (false positives). Therefore, recall is the primary metric to minimize false negatives.

I will also look at the Receiver Operating Characteristic (ROC) curve and the lift curve as secondary measures of performance.

Baseline

The first model was a simple decision tree. A decision tree is a non-parametric supervised learning algorithm, which can be used for classification tasks. It has a hierarchical tree structure, which consists of a root node, branches, internal nodes and leaf nodes.

The Decision Tree Classifier from sklearn is an easy to implement solution that provides a good foundation. It is simple to understand and to interpret and the tree can be visualized. Additionally, little data preparation is required as there is no need to normalize the data. However, these algorithms can have a tendency to overfit and create bias for unbalanced datasets.

Tuned Decision Tree

To address some of the disadvantages of decision trees, several variations of the Decision Tree Classifier were assessed using cross-validation and hypertuning of a few key variables. 324 iterations of variables were tested using 5 cross validation folds for each iteration.

Criterion was either ‘Gini’ or ‘Enthropy’.
Maximum tree depth was set to a range of values: None, 10, 20, 30, 40, 50.
Minimum samples per split was set to either 2, 5 or 10.
Minimum samples per leaf was set to either 1, 2 or 4.
Maximum number of features were set to either no maximum, the square root of all features or log squared of all features.

The recall metric from each of the 5 folds was averaged to assess the results of each iteration and determine the best set of parameters.

Since the population was very unbalanced, SMOTE and RandomUnderSampler from the imblearn library were utilized to oversample from the churn population and undersample from the non-churn population to create a balanced sample for training. Pipeline from the imblearn library enabled this sampling approach to be applied for each cross validation fold.

Random Forest

The next model assessed was a Random Forest Model. The RandomForestClassifier from sklearn was selected due to its proven success with classification predictions. It is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

Several variations of the model were assessed using cross-validation and random hypertuning of a few key variables. 24 iterations of variables were tested using 5 cross validation folds for each iteration.

Number of estimators was set to either 50, 100 or 200.
Maximum tree depth was set to either 30 or 50.
Minimum samples per split was set to either 2, or 4.
Minimum samples per leaf was set to 2.
Maximum number of features were set to either the square root of all features or log squared of all features.

The recall metric from each of the 5 folds was averaged to assess the results of each iteration and determine the best set of parameters.

Similar to the tuned decision tree, the sample populations were balanced with a Pipeline including SMOTE and RandomUnderSampler from the imblearn library.

CatBoost

The final model assessed was a CatBoost model. The CatBoostClassifier from catboost was selected due to its proven success with classification predictions, handling categorical data and unbalanced populations. It provides a gradient boosting framework which solves for categorical features using a permutation driven alternative compared to the classical algorithm.

This model is slightly different to implement than the previous models. The one hot encoded categorical columns are not needed. Instead the 18 original features are used directly as inputs for training the model.

Findings

Model results

All models used 70% of the assessment dataset for training and the remaining 30% for testing. The winning model based on recall was the CatBoost model.

Model	Recall
Baseline - Decision Tree	0.78
Tuned Decision Tree	0.81
Random Forest	0.84
CatBoost	0.86

After reviewing the results of the winning model, further assessment was given to the threshold used for predicting churn (instead of the default 50%). Reducing the threshold to 5% achieved almost 94% recall without massively inflating the size of the positive prediction class.

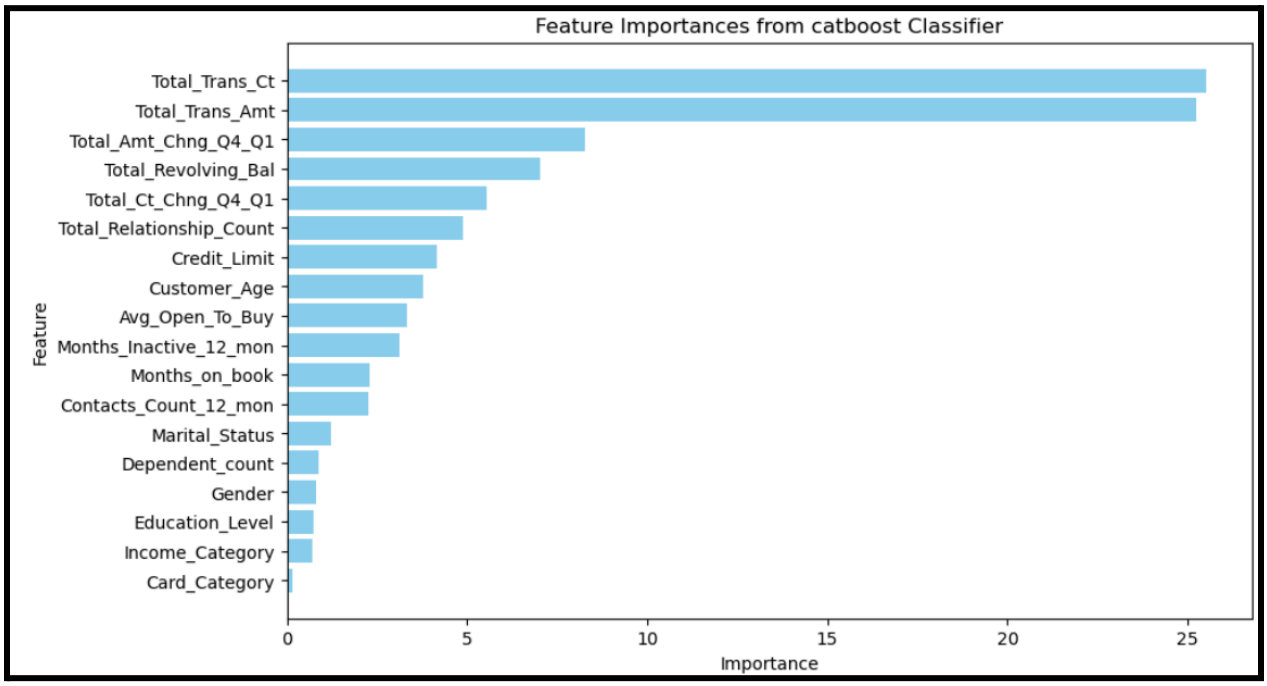
label	accuracy	precision	recall	F1	tp	fp	fn	tn	pred_pos	pred_neg	act_pos	act_neg
threshold_0	0.160	0.160	1.000	0.276	340	1787	0	0	2127	0	340	1787
threshold_5	0.961	0.839	0.938	0.886	319	61	21	1726	380	1747	340	1787
threshold_10	0.970	0.892	0.924	0.908	314	38	26	1749	352	1775	340	1787
threshold_15	0.971	0.904	0.918	0.911	312	33	28	1754	345	1782	340	1787
threshold_20	0.970	0.906	0.909	0.907	309	32	31	1755	341	1786	340	1787
threshold_25	0.972	0.924	0.900	0.912	306	25	34	1762	331	1796	340	1787
threshold_30	0.974	0.936	0.897	0.916	305	21	35	1766	326	1801	340	1787
threshold_35	0.972	0.940	0.882	0.910	300	19	40	1768	319	1808	340	1787
threshold_40	0.972	0.946	0.874	0.908	297	17	43	1770	314	1813	340	1787
threshold_45	0.971	0.946	0.871	0.907	296	17	44	1770	313	1814	340	1787
threshold_50	0.970	0.948	0.862	0.903	293	16	47	1771	309	1818	340	1787
threshold_55	0.969	0.951	0.853	0.899	290	15	50	1772	305	1822	340	1787
threshold_60	0.968	0.956	0.835	0.892	284	13	56	1774	297	1830	340	1787

The winning model was retrained with the full assessment dataset. The data set aside for validation was used as a final test. The validation results were measured using the 5% threshold for predicting churn.

CatBoost Testing Results at **5% threshold**													
Dataset	**accuracy**	**precision**	**recall**	**f1**	**tp**	**fp**	**fn**	**tn**	**pred_pos**	**pred_neg**	**act_pos**	**act_neg**	**total**
Assessment	0.961	0.839	0.938	0.886	319	61	21	1726	380	1747	340	1787	2127
Validation	0.96	0.814	0.971	0.886	474	108	14	2442	582	2456	488	2550	3038

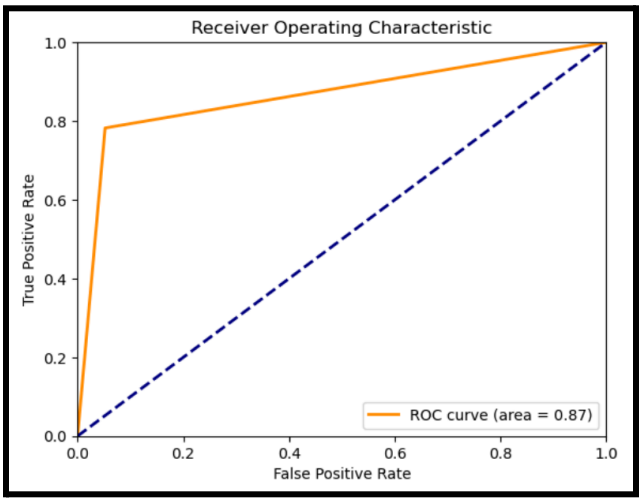
Reviewing the validation results, out of the 488 customers that churned, the model correctly included 474 in the positive class. Only 3% of the churned customers were missed. Conversely, of the 582 in the positive class, 19% did not churn. This is not a large enough number to cloud the whole class. Additionally, these may be key customers to focus on as potential churn candidates.

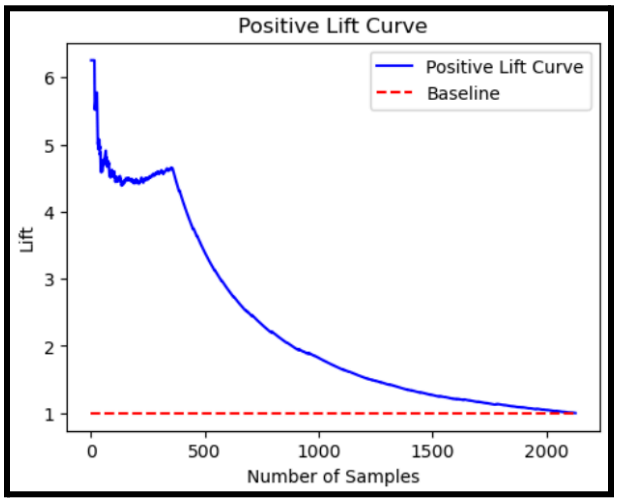
It was also interesting to note the total transaction count and amount were the most important features by a large margin. This helps support the conclusion that card activity is a key indicator of churn probability.



Baseline test results

	precision	recall	f1-score	support
Existing Customer	0.96	0.95	0.95	1787
Attrited Customer	0.74	0.78	0.76	340
accuracy			0.92	2127
macro avg	0.85	0.87	0.86	2127
weighted avg	0.92	0.92	0.92	2127

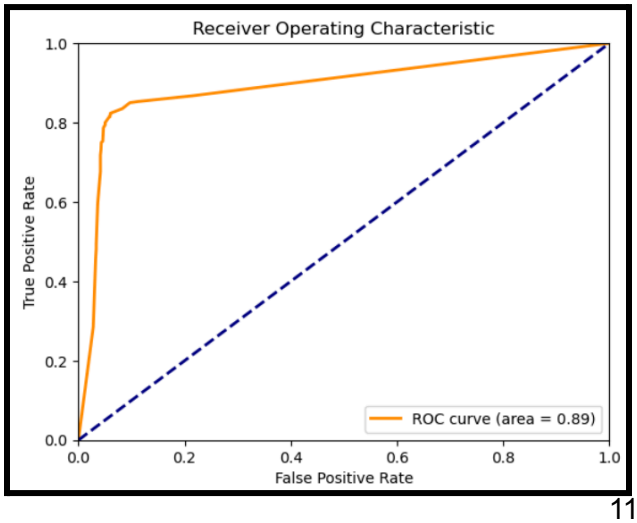


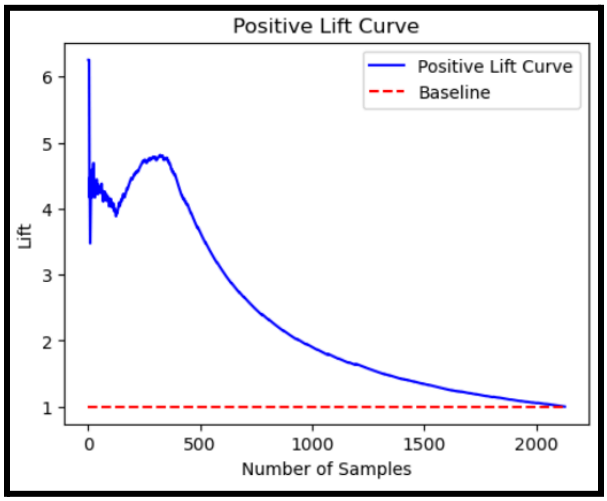


*Note: the ROC curve appears unusual as the model only predicted probabilities of 0, 0.5 or 1.0.

Tuned Decision Tree test results

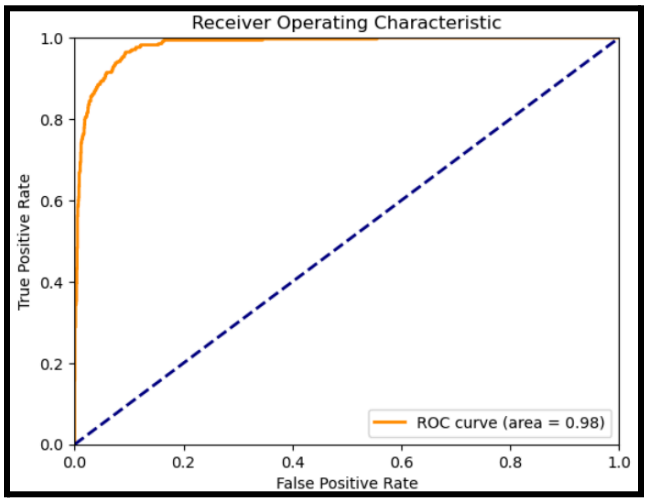
	precision	recall	f1-score	support
Existing Customer	0.96	0.94	0.95	1787
Attrited Customer	0.72	0.81	0.77	340
accuracy			0.92	2127
macro avg	0.84	0.88	0.86	2127
weighted avg	0.93	0.92	0.92	2127

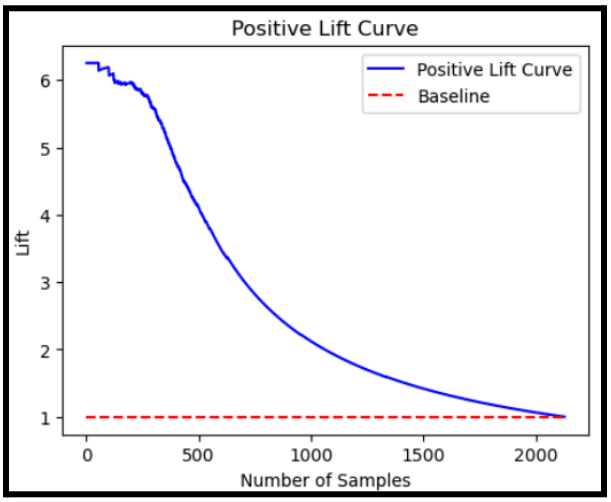




Random Forest test results

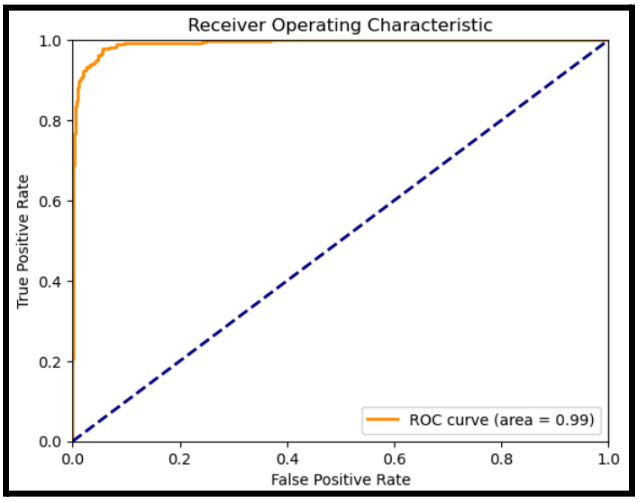
	precision	recall	f1-score	support
Existing Customer	0.97	0.97	0.97	1787
Attrited Customer	0.86	0.84	0.85	340
accuracy			0.95	2127
macro avg	0.91	0.91	0.91	2127
weighted avg	0.95	0.95	0.95	2127

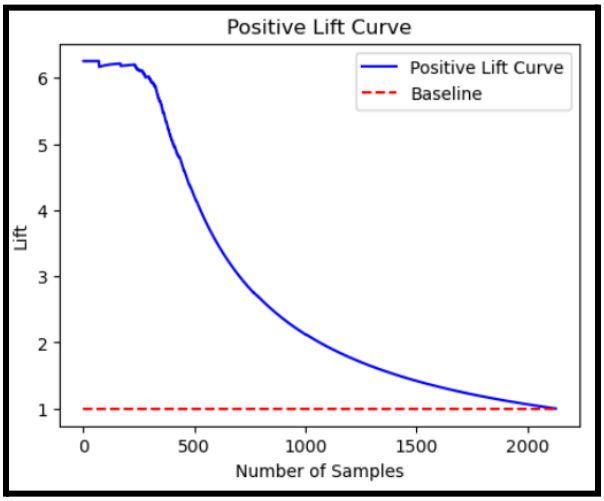




CatBoost test results

	precision	recall	f1-score	support
Existing Customer	0.97	0.99	0.98	1787
Attrited Customer	0.95	0.86	0.90	340
accuracy			0.97	2127
macro avg	0.96	0.93	0.94	2127
weighted avg	0.97	0.97	0.97	2127





Use cases

With the ability to identify churn risk customers, the portfolio manager will know the appropriate population to address with churn mitigating activities.

General reward programs

The company could increase card usage by implementing or enhancing existing reward programs to incentivize the size and volume of transactions.

Survey churn risk population

By identifying the segment of customers most likely to churn, proactive outreach could occur to understand what is preventing these customers from using their cards more. There may be key features missing from the card that competitors offer, or their could be concerns about the level of customer service. Collecting feedback would help to focus retention efforts in the most relevant areas.

Targeted incentives

Reducing interest rates and customized reward offerings could help to generate additional activity for the subset of customers identified to be in the churn risk population.

Future enhancements

1. Hypertuning model
2. Threshold monitoring
3. Automation
4. Data quality checks