Green Banana Company Project

Background and Business Understanding

The firm has three million active customers with about 30% yearly churn. This yearly churn is offset by acquiring 900,000 new customers, hence **ZERO net growth.** The dataset provided consists of **857178 records** and **53 columns**. The critical client objectives are:

1. To develop a marketing plan to **retain** more customers or **rescue churning** customers.
2. The marketing team needs a model to:
    a. Understand **which factors contribute** to customer churn
    b. **Identify prospective churn candidates** by proposing a churn model

In verifying the churn from the data set, I discovered **127,474 records with the label 'X'** in the dataset. A further breakdown of the churn data revealed that the churn rate in the dataset was **31.3785%.** The figure below shows the breakdown of churn in the dataset and per snapshot.

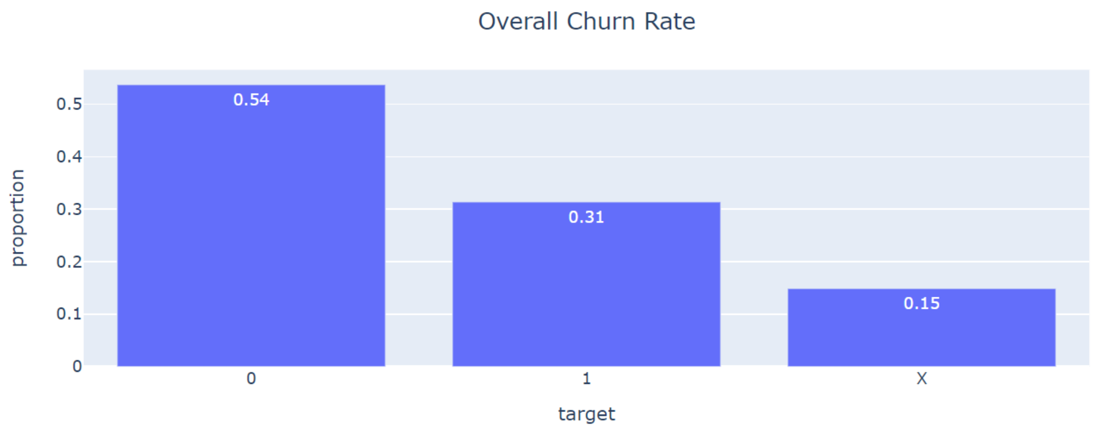


Figure 1: Churn Rate

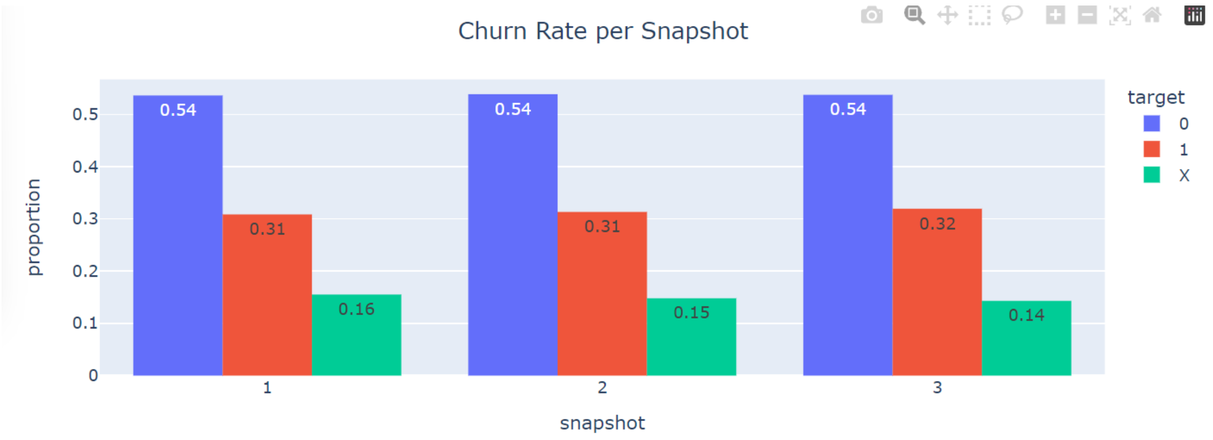


Figure 2: Churn Rate per Snapshot

After observing the X labels, I removed all these records from the dataset and continued my analysis. The resulting dataset had **729,704 records**, and the churn rate used for my analysis **was 36.86%.**

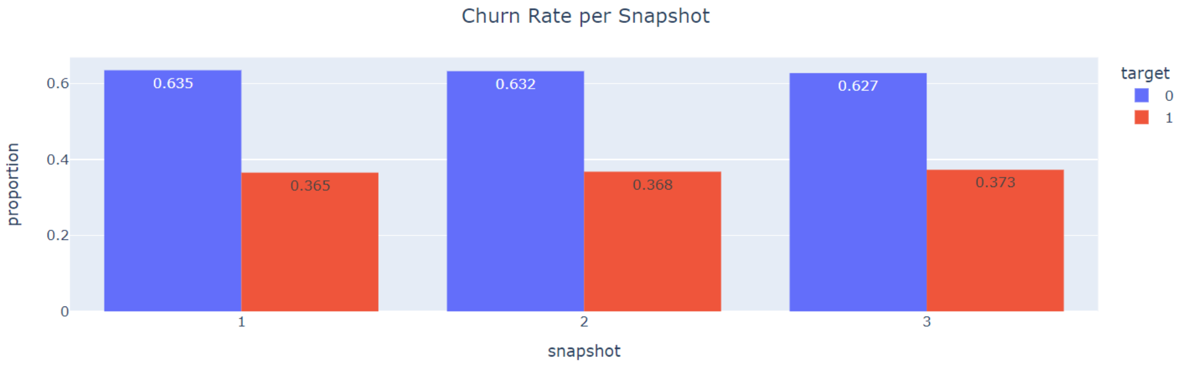


Figure 3: Churn rate after data cleaning

Data Understanding and Building a Business Case for the Project

Across Snapshot Analysis

From the data, I realised that the **average customer spend per shipment is USD49.57**. Other relevant spending metrics are:

Avg Customer Gross Spend = USD 932.12

Avg Customer Net Spend (Item Value) = USD 892.32

Avg Customer Shipping Spend = USD 13.05

Avg Customer Spend last year = USD 43.27

Avg Total Spend last three months = USD 104.73

Avg total spend in last 6 months = USD 229.72

Now, let's take a closer look at churned customers' profiles. On average, churned customers spend **USD 37.83 per shipment**, and the total number of active shipments for churned customers is **348,476**. This means that, on average, Green Banana Company loses **USD 13,203,755.64** yearly**.**

Given that not all churned customers can be salvaged, I run multiple scenarios in the table below to show the savings that could be made at various thresholds.

Salvage Perc	Savings
2.50%	$ 329,571.18
5%	$ 659,142.35
10%	$ 1,318,284.71
12.50%	$ 1,647,855.89
15%	$ 1,977,427.06

Table 1: Scenario Planning to understand the impact of salvage percentages on Revenue

Snapshot Specific Analysis

Average Spend Analysis

We repeat the analysis above for each snapshot

	Avg Spend per Shipment per Churned Customer	The active number of shipments for Churned Customers	Average Loss
Snapshot 1	USD 38.94	113,256	4,410,188.64
Snapshot 2	USD 37.61	115,855	4,357,306.55
Snapshot 3	USD 36.99	119, 365	4,415,311.35

Table 2: Average Spend Analysis per Snapshot

From the above table, using a salvage threshold of 5%, i.e., the max percentage of churned customers we can salvage by extra-marketing efforts, we save about USD 220,509.43.

Total Spend Analysis

A careful look at the spend data revealed that the total spend of customers who churned in the last 3 and 6 months are as follows:

	Total Spend by Churned Customers in the past three months	Total Spend by Churned Customers in the past six months
Snapshot 1	6,574,947.33	14,533,823.35
Snapshot 2	6,065,951.73	13,895,170.23
Snapshot 3	5,883,089.62	13,642,302.07

Table 3: Total Spend Analysis per Snapshot

To put these figures in perspective, the churned customers in each snapshot contributed about **25%** of total spending. The figure below shows the breakdown.

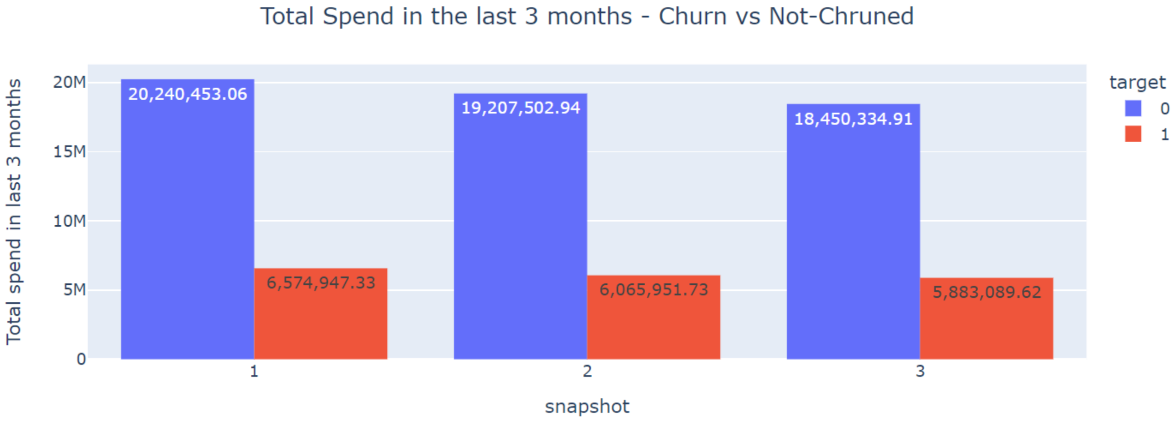


Figure 4: Total Spend in the Last 3 Months Churned vs Not Churned

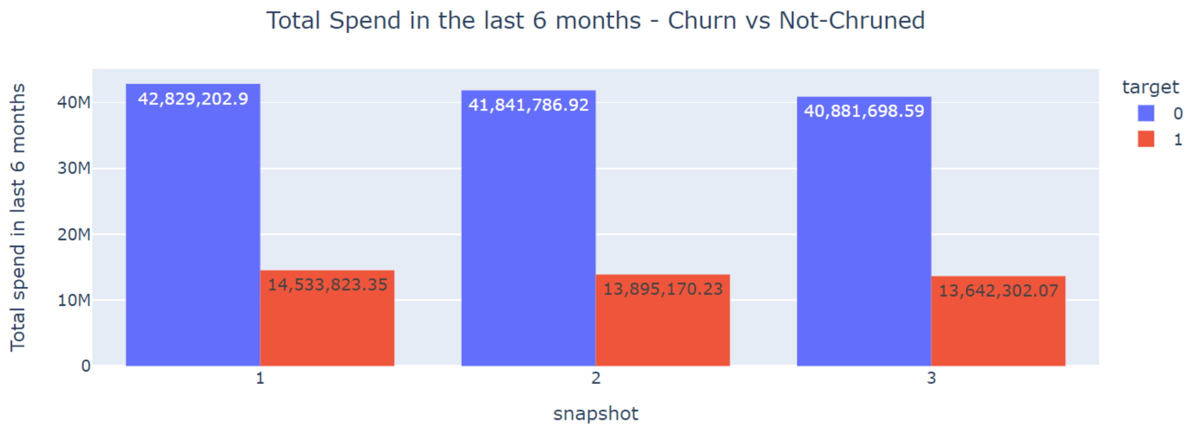


Figure 5: Spend in the Last six months Churned vs Not Churned

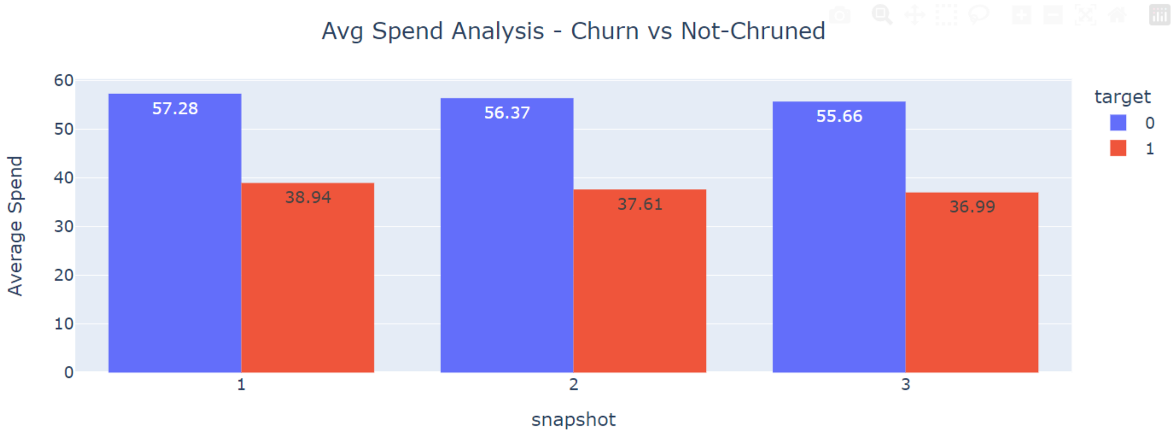
From the above charts, it can be deduced that the firm loses a significant chunk of actual spending revenue by looking at the expenditure profile of churned customers. This shows that significantly investing in salvaging a percentage of churned customers will be of tremendous value to the firm.

Customer Lifetime Value Analysis

The table below shows the average spend analysis by churn categories and reveals that, on average, retained customers spend more than churned customers. The figure also indicates this analysis by snapshot.

target	mean	median	max
Not Churned	USD56.43	USD 48.79	USD 957.22
Churned	USD 37.83	USD 31.56	USD 753.33

Table 4: Spend Profile of Customers (Churned vs Not-Churned)



As the adage goes, it costs less to retain an existing customer than it does to retain a new one. To calculate the CLV, we use the **average purchase value, average purchase frequency**, and **the average lifespan of the customer**, i.e.

$$Customer\ Lifetime\ Value = \ AVG\ Purchase\ Freq * AVG\ Customer\ Value * AVG\ Customer\ Lifespan$$

To determine the **average customer value**, I found the average customer spend in the entire dataset, **USD 51.13**. I also used the total number of shipments as a proxy for **purchase frequency** and found the average to be **1.55 shipments**. Lastly, I computed the average customer lifespan using the inverse of the churn rate, **2.71 months**.

From the above, a customer's **Average Lifetime Value is USD 215.25**. Given this computation, for every customer churned, the company loses **USD 215.25** of Revenue that could have been potentially gained had the customer been retained over their lifetime (i.e., repeat purchases). The figure below shows the potential Revenue lost for churned customers over the customer's lifetime.

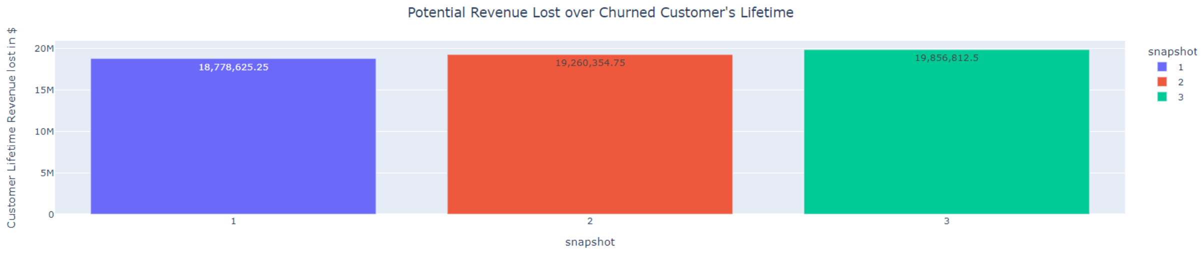


*Figure 6: Potential Revenue lost over the lifetime of the Churned Customer (CLV * Number of Churned Customers)*

We will use the customer's average lifetime value to justify the DS project's business value.

Building the Churn Model

There are two (2) objectives to building this churn model, i.e.

- Determine factors that contribute to churn
- Build a business case for the churn model

Determining Factors Contributing to Churn

I fitted multiple linear classification models to the dataset to determine the factors contributing to customer churn in Green Banana Co. I chose these models not to maximise the model's predictive power but to focus on model interpretability. I used the sklearn library for this section.

The Logistic Regression Model

I first prepared the data to fit the logistic regression model and removed all null values. That is, I excluded all variables with more than 100,000 missing values. In addition, I removed the missing records in the dataset corresponding to the 'f_i_max_progress' and 'f_i_max_shipments' columns. The final dataset for building the model contained 721195 records and 45 columns.

I also standardised all the variables using the MinMax scaler to build the logistic regression model. This was done because the solvers in sklearn work faster with all variables on the same scale. In addition, scaling the predictor variables is a prerequisite to enabling comparisons of predictors to understand which predictor is influential. Furthermore, I utilised the cross-validation (cv=10) technique to ensure my results were generalisable. I report the results of each fold of the logistic regression classifier below. **Please note that the evaluation metric for all models reported is accuracy.**

Fold	fit_time	score_time	estimator	test_score	train_score
0	16.299	0.186	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.669	0.713
1	21.419	0.214	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.679	0.712
2	15.695	0.200	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.712	0.708
3	22.346	0.139	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.700	0.709
4	19.506	0.142	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.691	0.710
5	16.814	0.152	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.703	0.709
6	19.944	0.194	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.693	0.710
7	19.509	0.146	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.637	0.715
8	21.098	0.183	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.416	0.726
9	15.442	0.147	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregression', LogisticRegression(max_iter=1000))])	0.467	0.726

Table 5: CV results of the Logistic Regression Classifier

The **training accuracy score** of this model is **0.714 ± 0.00658**, whilst the testing score accuracy is **0.637 ± 0.106**. From the results, the logistic regression model overfits the data as the training accuracy is higher than the test accuracy.

Furthermore, I plot the distribution of the features' coefficients across each fold to interpret the logistic regression model. The boxplot below shows the distribution of the predictors across each fold.

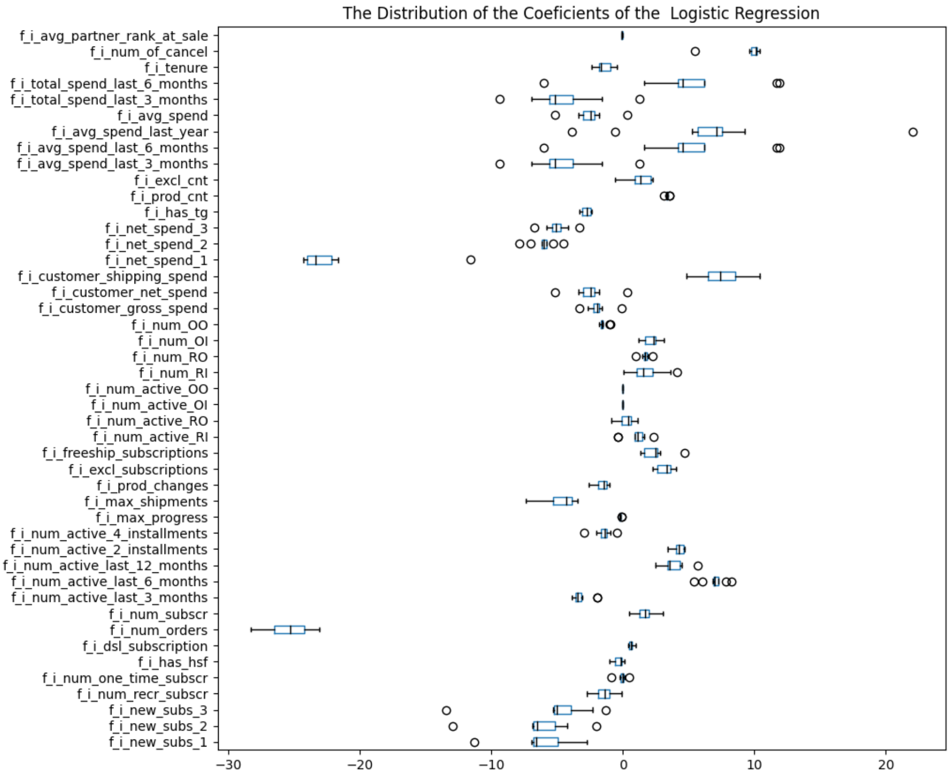


Figure 7: Distribution of Logistic Regression Coefficients across CV Folds

From the above, it can be deduced that:

1. The **total number of orders** variable has the most significant influence on churn. This is followed by the **net spend in the last 30-day window variable** and the **number of cancelled orders.**
2. The variables **new_subsriptions in the 30, 60, and 90-day windows**, as well as **total spend in these windows and customer spend on shipping**, are variables with some influence of churn.

I decided to fit a finetuned Logistic regression because there were so many variables with little to no influence on churn (with a coefficient of zero). By this, I finetune the **regularisation parameter, C,** of the logistic regression classifier. In the table below, I report the finetuned Logistic regression model results.

fit_time	score_time	estimator	test_score	train_score
226.909	0.168	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv', LogisticRegressionCV(cv=5, max_iter=1000))])	0.637	0.637
214.172	0.191	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv',	0.637	0.637

		LogisticRegressionCV(cv=5, max_iter=1000))])		
225.955	0.146	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv', LogisticRegressionCV(cv=5, max_iter=1000))])	0.637	0.637
240.844	0.145	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv', LogisticRegressionCV(cv=5, max_iter=1000))])	0.637	0.637
218.020	0.170	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv', LogisticRegressionCV(cv=5, max_iter=1000))])	0.637	0.637
214.848	0.165	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv', LogisticRegressionCV(cv=5, max_iter=1000))])	0.637	0.637
219.882	0.181	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv', LogisticRegressionCV(cv=5, max_iter=1000))])	0.637	0.637
239.004	0.146	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv', LogisticRegressionCV(cv=5, max_iter=1000))])	0.627	0.715
223.566	0.158	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv', LogisticRegressionCV(cv=5, max_iter=1000))])	0.410	0.727
178.088	0.149	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('logisticregressioncv', LogisticRegressionCV(cv=5, max_iter=1000))])	0.579	0.645

Table 6: VS results of the Finetuned Logistic Regression Classifier

This model's **training accuracy score is 0.655 ± 0.0352**, while the **testing score accuracy** is **0.608 ± 0.0716**. By finetuning, the accuracy of the training is reduced. However, the model also overfitting slightly reduces.

Furthermore, to interpret the finetined logistic regression model, I plot the distribution of the coefficients across the folds. The boxplot below shows the distribution of the predictors in each fold.

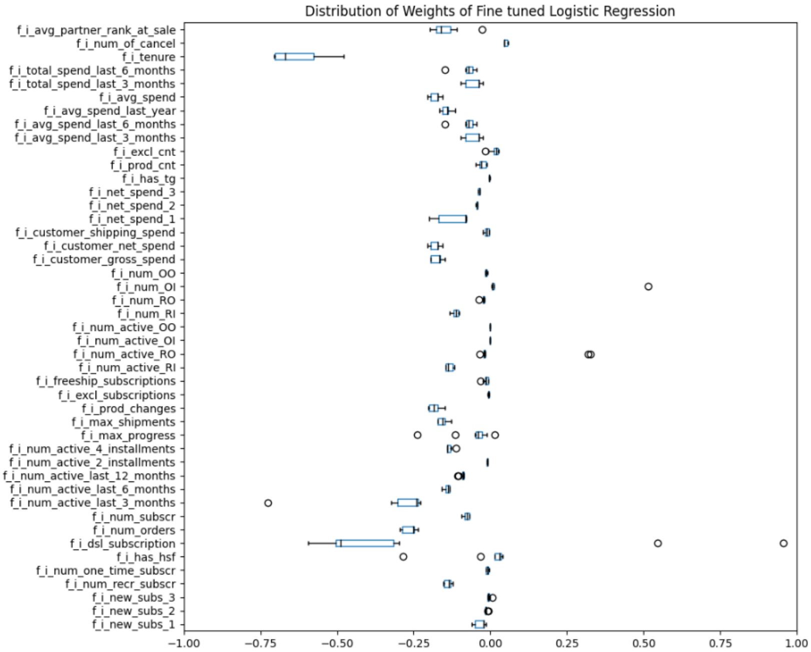


Figure 8: Distribution of Finetuned Logistic Regression Coefficients across CV Folds

From the finetuned logistic regression model, we can deduce the following:

1. The **tenure, measured in days and the days since the last subscription**, are influential predictors of churn.
2. The **number of orders, active shipments in the last three months, and customer spending characteristics** are also influential predictors.

In addition, I tested a logistic regression model with polynomial features (degree=2). I report the results of this model in the final model results table. To further understand the effect of the other predictors on churn, I decided to test the ridge regression model.

The Ridge Classifier Model

I followed the same data preparation technique as outlined previously to build the ridge classification model. The results of the ridge regression model are shown below:

Fold	fit_time	score_time	estimator	test_score	train_score
0	5.504	0.241	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridge classifier, RidgeClassifier(alpha=100))])	0.648	0.694
1	4.476	0.212	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridgeclassifier',	0.653	0.693

			RidgeClassifier(alpha=100))])		
2	3.528	0.217	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridgeclassifier', RidgeClassifier(alpha=100))])	0.673	0.690
3	3.672	0.186	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridgeclassifier', RidgeClassifier(alpha=100))])	0.665	0.692
4	3.617	0.186	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridgeclassifier', RidgeClassifier(alpha=100))])	0.672	0.691
5	3.681	0.180	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridgeclassifier', RidgeClassifier(alpha=100))])	0.695	0.687
6	3.320	0.185	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridgeclassifier', RidgeClassifier(alpha=100))])	0.702	0.688
7	3.741	0.195	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridgeclassifier', RidgeClassifier(alpha=100))])	0.692	0.690
8	3.909	0.193	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridgeclassifier', RidgeClassifier(alpha=100))])	0.473	0.704
9	4.262	0.205	Pipeline(steps=[('minmaxscaler', MinMaxScaler()), ('ridgeclassifier', RidgeClassifier(alpha=100))])	0.455	0.708

Table 7: CV Results of the Ridge Classifier Model

This **model's training accuracy score is 0.694 ± 0.0069**, while the **testing score accuracy is 0.633 ± 0.0907**. The ridge regression model also overfits the data.

Furthermore, to interpret the ridge classifier model, I plot the distribution of the coefficients across the folds. The boxplot below shows the distribution of the predictors in each fold.

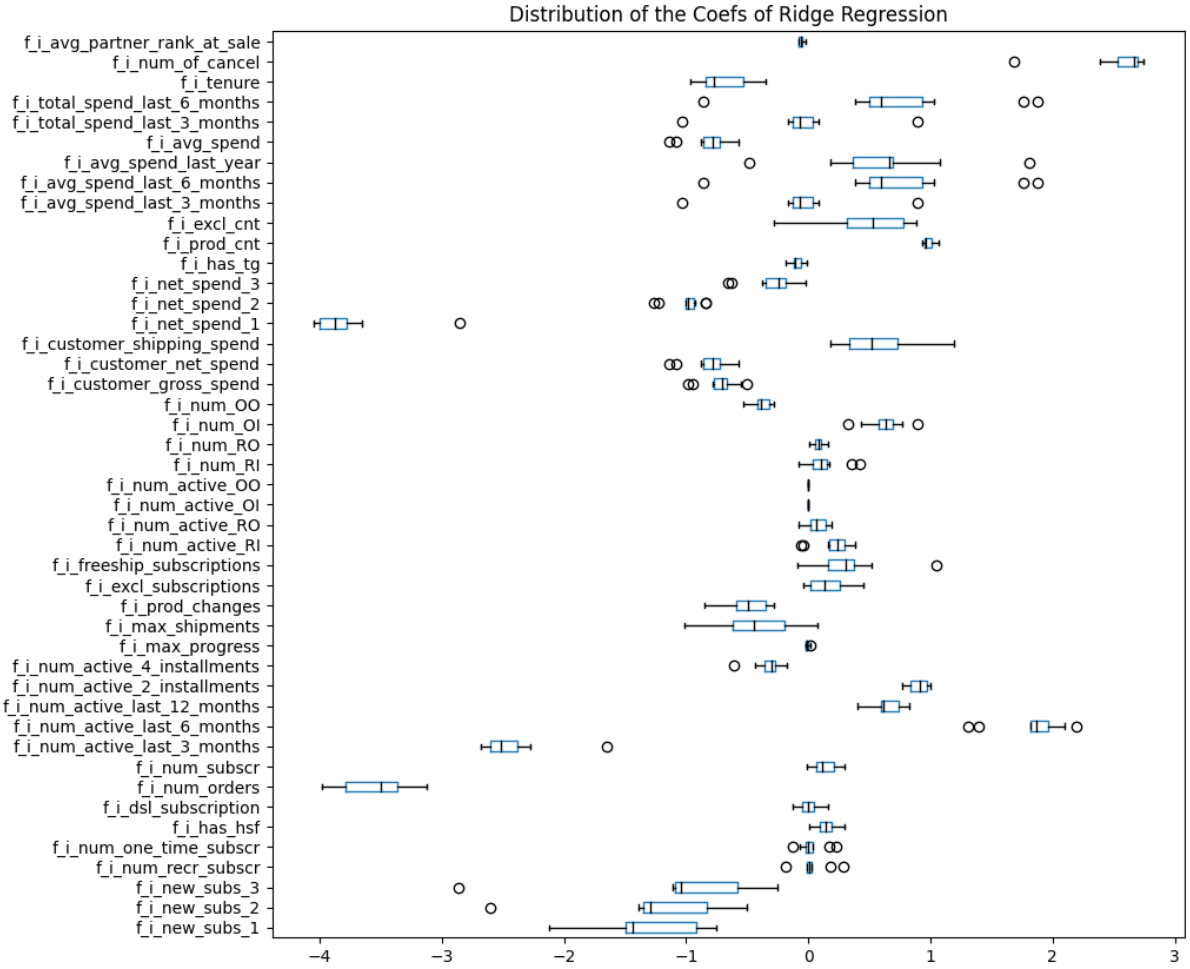


Figure 9: Distribution of Ridge Classifier Coefficients across CV Folds

The ridge classifier model reveals the following:

1. The most critical features to consider when reducing spending are the number of orders, active orders in the last 3 months, customer net spend, and new subscriptions in the last 30 days.
2. This model also reveals that the number of cancelled orders is critical in churn.

In addition, I tested a finetuned ridge classification model to improve the model's accuracy and predictive power. The results are reported below

Fold	fit_time	score_time	test_score	train_score
0	47.347	0.270	0.654	0.701
1	44.728	0.210	0.661	0.700
2	44.322	0.178	0.683	0.696
3	43.976	0.207	0.670	0.697
4	43.832	0.191	0.672	0.697

5	43.263	0.198	0.689	0.696
6	44.178	0.177	0.697	0.700
7	44.758	0.218	0.653	0.704
8	43.903	0.205	0.395	0.713
9	44.038	0.193	0.486	0.710

Table 8: CV results of the finetuned ridge classification model

The training accuracy of this model is 0.701 ± 0.00585, whilst the testing accuracy is 0.626 ± 0.101. This model still overfits the data. The distribution of coefficients is found below:

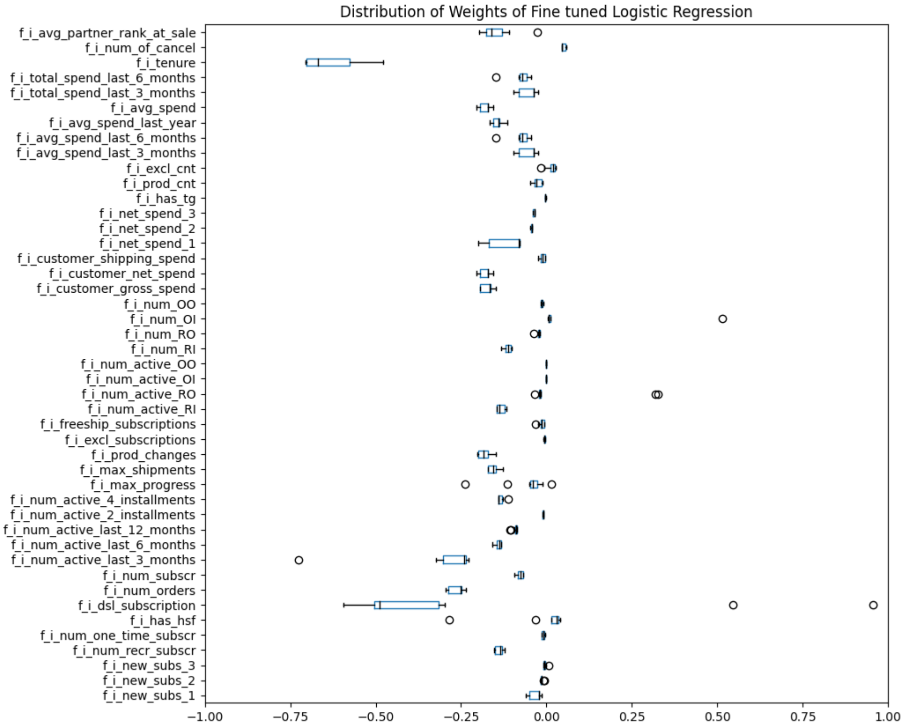


Figure 10: Distribution of Finetuned Ridge Classifier Coefficients across CV Folds

The above image reveals the following:

- The number of cancelled orders, customers' tenure in days, customer spending (shipping, gross, and net), active orders in the last three months, and the number of orders and days since the last subscription was created are influential in predicting churn.

The table below summarises all the findings from the modelling phase of the project.

Model	Training Accuracy	Test Accuracy	Interpretation	Decision
Baseline (Dummy Classifier)	0.637 ± 0.00	0.637 ± 0.00	This is the baseline model.	
Logistic Regression Classifier	0.714 ± 0.00658	0.637 ± 0.106	Model overfits the data. However, the std dev of test accuracy is high	Accept
Finetuned Logistic Regression Model	0.655 ± 0.0352	0.608 ± 0.0716	Model overfits the data with low test accuracy	Reject
Ridge Classification Model	0.694 ± 0.0069	0.633 ± 0.0907	Model overfits the data with low test accuracy	Reject
Finetuned Ridge Classification Model	0.701 ± 0.00585	0.626 ± 0.101	Model overfits the data with low test accuracy	Reject
Logistic Regression Model with Polynomial Features (degree=2)	0.732 ± 0.00665	0.624 ± 0.128	Model overfits the data. However, the standard deviation of the test accuracy is high.	Reject

Table 9: Table of results from the model

Final Model and Factors Contributing to Churn

From the analysis above, the characteristics that contribute to churn are the following:

1. Number of orders
2. Net spend in the last 30-day window
3. Number of order cancellations
4. Customer total Spend on shipping

I then built a logistic regression model using these four features as predictors. The results are shown below:

Fold	fit_time	score_time	test_score	train_score
0	2.246	0.121	0.677	0.695
1	2.125	0.097	0.681	0.694
2	1.572	0.111	0.682	0.693
3	2.531	0.116	0.674	0.695
4	2.099	0.133	0.652	0.696
5	2.488	0.112	0.693	0.694

6	1.976	0.129	0.776	0.691
7	1.749	0.127	0.800	0.688
8	2.013	0.122	0.704	0.691
9	1.889	0.132	0.479	0.711

Table 10: CV results of the Final Model

The final model's **training accuracy is 0.695 ± 0.00622**, and the **testing accuracy is 0.682 ± 0.0853**. This model reduces the overfitting problem identified earlier. I also plot the coefficient's distribution and make recommendations below.

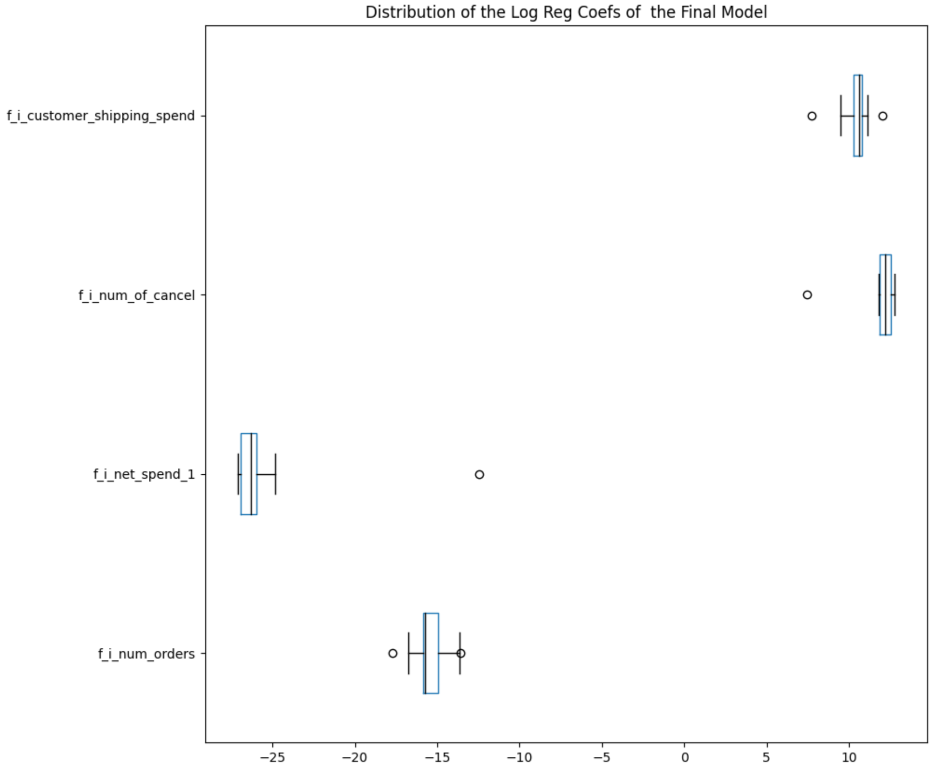


Figure 11: Distribution of Coefficients in the Final Logistic Regression Model

Recommendations:

1. A reduction in customer shipping costs will reduce the probability of churning. Can the firm outsource its logistics services to reduce this cost?
2. The number of orders cancelled is a good predictor of churn. Can the firm institute a policy of automatically recommending promotions/deals upon order cancellations to reduce the probability of churn?
3. The firm can increase promotional strategies to increase net spend and total customer orders. As customers spend more and order more shipments, their churn probability reduces.

The figure below shows the confusion matrix of the final logistic regression model.

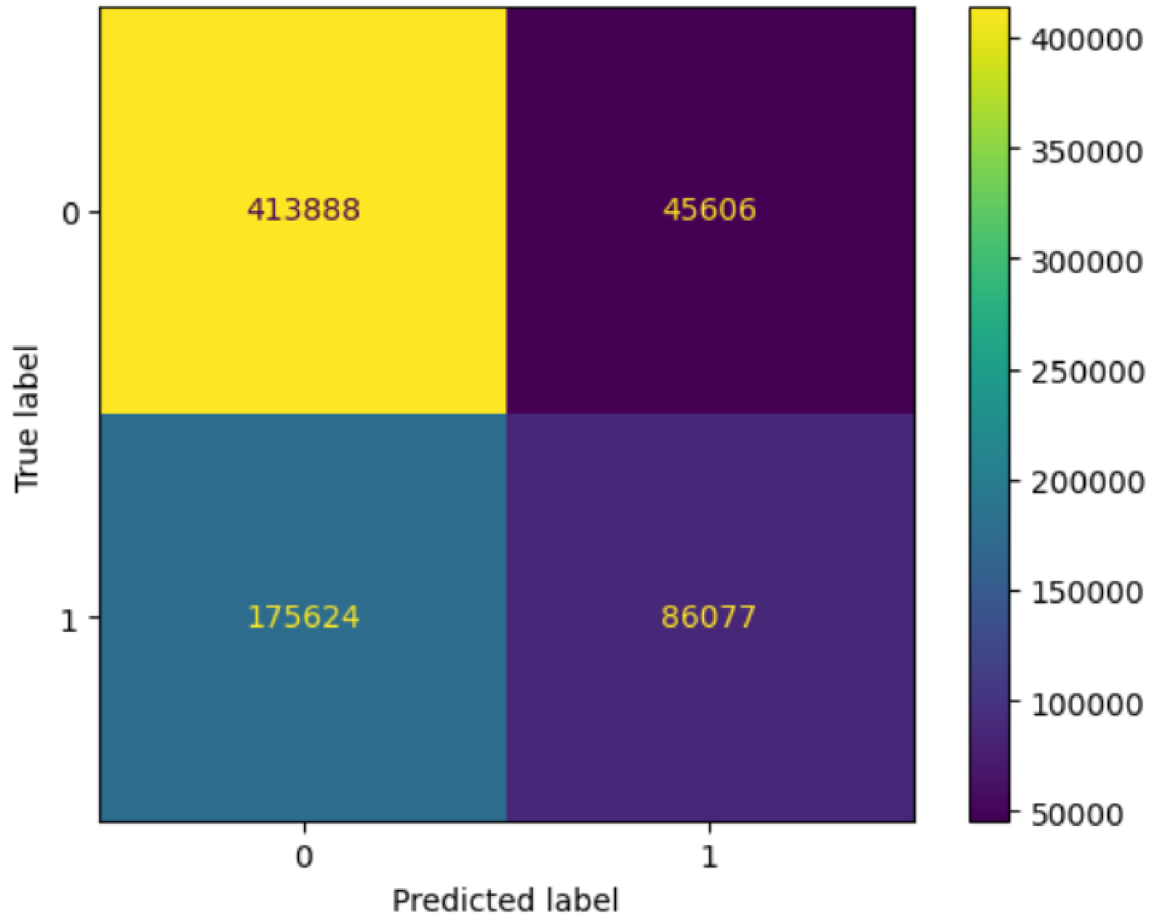


Figure 12: The Confusion Matrix of the Final Log Regression Classifier

Business Case for Churn Model

I computed the model's sensitivity to analyse the economic impact of the developed model. Thus, the probability that the model will correctly predict churned customers from those who churn.

$$Sensitivity = \frac{86077}{86077 + 175624} = 0.33$$

Using 33% as the probability of correctly predicting churned customers, the table below summarises my business case for this model.

Snapshot Analysis

Snapshot no:		**3**
Number of churned Customers		**92,250**
Customer Lifetime Value		**215.25**

Potential Revenue lost by the firm	**Customer Lifetime Value * Number of churned Customers**	**19,856,812.50**
Sensitivity of the Churn Model		**33.00%**
Number of Churned Customers Correctly Predicted	**Sensitivity of the Churn Model * Number of churned Customers**	**30443**
Potential Revenue that could be salvaged after prediction with the Churn Model	**Number of Churned Customers Correctly Predicted * Customer Lifetime Value**	**6,552,748.13**

Sensitivity Analysis

Salvage Percentage	Potential Savings	Decision
1.00%	$ 65,527.48	Loss
1.50%	$ 98,291.22	
2.00%	$ 131,054.96	
2.50%	$ 163,818.70	
3.00%	$ 196,582.44	
3.50%	$ 229,346.18	
4.00%	$ 262,109.93	Profit
4.50%	$ 294,873.67	
5.00%	$ 327,637.41	
5.50%	$ 360,401.15	
6.00%	$ 393,164.89	
6.50%	$ 425,928.63	
7.00%	$ 458,692.37	

From the table above, it can be concluded that if 4% or more of customers predicted to be churned are salvaged per snapshot (using the recommendations above), Green Banana Co will benefit from the model positively. Against this background, I recommend that the DS project be implemented.

References

1. https://www.quora.com/What-are-some-of-the-ways-to-calculate-a-customers-lifetime-value-LTV-that-do-not-rely-on-churn-rate