Dillards Returns

Hanxi (Elsie) Lin, Judy Zhu, Yuanzhu (Zoe) Li & Xenia Vrettakou

12.08.2023
MLDS 400 - Everything Starts with Data
Group 8

Table of Contents

EXECUTIVE SUMMARY

This report explores the issue of reducing return rates for Dillard's retail chain. It emphasizes the significance of comprehending the factors that contribute to high return rates and aims to identify inherent product attributes associated with returns using data analysis techniques. The document also includes background information on Dillard's and presents findings from exploratory data analysis and data modeling. The conclusion underlines the importance of ongoing improvement and proposes future steps for the company.

INTRODUCTION

Dillard’s is a prominent retail chain operating across numerous stores. We want to address the intricate challenge of reducing return rates across its diverse product inventory. High return rates present multifaceted challenges, indicating potential mismatches between consumer preferences and products, operational inefficiencies, and financial implications. Understanding and addressing these return rates is a pivotal concern, suggesting the need for a nuanced comprehension of the underlying factors that lead to increased returns. In an evolving retail environment where customer experience and satisfaction reign supreme, return rates are a crucial barometer of the effectiveness of product selection strategies and the alignment of products with consumer demands. This analysis aims to unravel the complex factors associated with high return rates in Dillard's inventory. Utilizing sophisticated data analysis techniques, specifically feature selection using Random Forests and Logistic Regression Analysis, the goal is to identify intrinsic product attributes that are strongly associated with high rates of return. By obtaining a comprehensive understanding of these influential features, our goal is to refine its product selection strategy, optimize inventory, potentially reduce return rates, and improve operational efficiency while increasing overall customer satisfaction.

BACKGROUND

Our analysis focuses on Dillard's department store chain. Dillard’s was founded in 1938 in Nashville, Arkansas and it is currently headquartered in Little Rock, AR. It is an American fashion retail company that offers both in-person and online shopping experiences to its customers. Their department stores sell various products including clothing, home goods, accessories, beauty products, and more. In 1969 Dillards went public and as of 2023, it has more than 270 retail stores in 29 States.

EXPLORATORY DATA ANALYSIS (EDA) & DATA CLEANING

The comprehensive analysis of the overall return rates, as presented in Table 1, reveals that the highest rates occur predominantly in the latter half of the year, particularly in December. Notably, these rates are comparable to those observed in January. The findings suggest a trend where return rates during the first half of the year surpass those of the second half, with a significant spike identified in June.

A more extensive examination of the historical return rate data dating back to 2004, detailed in Table 2, underscores variations across states. The majority of states exhibit return rates below 4%, while Florida (FL) and Texas (TX) stand out with rates exceeding 5%.

These insights contribute to a nuanced understanding of the temporal patterns in return rates and the distinctive variations observed across states. The detailed examination allows for strategic considerations in managing return rates, particularly in regions with higher incidence, enhancing the overall effectiveness of the business approach.

In the EDA process, it was found that 1205238 rows in the transaction dataset have $0 Orgprice, an uncommon occurrence. Further investigation uncovered inconsistencies where equivalent products with the same SKU had different non-zero prices. To address this, a data cleaning method replaced Orgprice values of 0 with the maximum value for each unique item. After that we got 12363 rows with 0 values, further investigation uncovered remaining 0 values for certain SKUs. To address this, the skstinfo table was checked, replacing 0s in the Retail feature with NaN. The mean of each retail item, grouped by SKU, was calculated and used to replace the remaining 0 values in the transactions table, ensuring accurate pricing representation.

According to the correlation matrix in Table 3, we observe a high correlation between ORGPRICE, AMT, and amt_diff and retail_diff. This observation is something we accounted for during our modeling process.

SKU Table: We cleaned and replaced all empty values, N/A, NA with NaN, and removed any rows with missing values. We converted all columns to appropriate types, all are categorical. The columns we keep are `['BRAND', 'DEPT', 'SKU']`

Transact Table: We cleaned and replaced all empty values, N/A, NA with NaN, and removed any rows with missing values. We converted all columns to appropriate types, including numeric, categorical, and datetime. The columns we keep are `['SKU', 'STYPE', 'STORE', 'ORGPRICE', 'QUANTITY', 'SALEDATE', 'AMT']`.

FEATURE ENGINEERING

Firstly, Binning and Categorization have been employed to create a feature named `BRAND_dummy`. This involves using the `pd.cut` function to categorize `ORGPRICE` into distinct groups: 'Cheap', 'Affordable', and 'Luxury', based on quartiles (q25 and q75). This categorization provides a useful way to group prices into meaningful ranges, facilitating more nuanced analysis and modeling.

Secondly, the process of Filtering Data has been applied, specifically targeting rows where

`ORGPRICE` exceeds 1000. By removing these outliers or extreme values, the analysis becomes more focused on a specific range of prices, which can enhance the performance and accuracy of predictive models.

In the realm of Date Manipulation, the `SALEDATE` column is converted to a datetime format using `pd.to_datetime`, and a new feature, `Salemonth`, is created by extracting the month from `SALEDATE`. This step is crucial for enabling analysis of seasonality and identifying sales patterns that vary month-to-month.

The fourth step involves Calculating Differences, resulting in the creation of two features: `amt_diff` and `retail_diff`. This process calculates the differences between `AMT` and `ORGPRICE`, as well as `RETAIL` and `ORGPRICE`, thereby creating features that reflect the discrepancies between the actual transaction amounts and the original price. These differences are insightful for understanding various pricing strategies and customer purchasing behaviors.

Lastly, Calculating Return Rate involves determining the average return rate for each state, creating a feature named `Return rate`. This is achieved by utilizing the `Stype` column to discern whether a transaction is a purchase or a return. The purpose here is to analyze how return rates vary across different geographical regions, offering a lens into regional consumer behavior and product performance.

Overall, each of these steps in feature engineering plays a vital role in transforming raw data into a more analytically valuable form, aiding in both the enhancement of machine learning models and the deepening of data-driven insights.

The feature engineering process applied various transformations and creations of new features in the dataset. These engineered features can potentially enhance the analysis and predictive capabilities of models by providing additional insights and information from the original dataset.

CUSTOMER DATA ANALYSIS

A. Final Data Preparation

Prior to data modeling we decided to randomly select 100 thousand samples from our cleaned dataset of about 66 million samples. This decision was made so we can prevent future inefficiencies like computationally intense calculations that would slow down our analysis. After this initial step, we used the hold-out cross-validation methodology to split our data into a train set (80%) and a test set (20%). The last step in terms of splitting the data was to take 15 thousand random subsets from the 80 thousand training set of samples that *Returns* equal to 0.

Then we proceeded by addressing the issue of class imbalance. As we previously mentioned the focus of our analysis is to predict the probability of an item being returned. The target feature is

Returns which takes two values (1 if the item was returned and 0 if it was purchased). It is intuitively expected and confirmed through our EDA that when the *Returns* feature equals 1 the total number of returns is significantly less than the total number of purchases. This is something we accounted for by using SMOTE, a technique that generates synthetic samples based on data. To be more precise SMOTE, oversamples the minority class (*Returns* equal 1) by increasing the number of samples to 15 thousand for the training set only. This was done only on the training set because our goal is to train the model on balanced training data so that it learns how to separate the two classes really well. Then we evaluated our model's abilities on the test set.

B. Logistic regression

We applied logistic regression as our baseline model, typically used for binomial classification problems. Our target feature is the categorical variable *Returns*. Our first model (Model 1) consisted of all 8 of the variables that we had in our final dataset after we cleaned them and featured engineering some of them. Model 1 did not perform well as only half of the variables were significant. The remaining 4 were highly correlated with ORGPRICE and Salemonth was simply insignificant. We then attempted to improve our model by excluding amt_diff, retail_diff and AMT (Model 2), to account for multicollinearity issues with ORGPRICE. This second model improved the significance of the coefficients remaining in the model but Salemonth was still insignificant.

We applied the confusion matrix to evaluate both of our models. Table 4 depicts the confusion matrix for model 1. We observed that accuracy is relatively good if we base the performance of our model on that metric. However, it is important to mention that when dealing with imbalance data, accuracy is not the best metric to use. Further, precision and F1 are very low in classifying *Returns* equal to 1. In terms of model 2 even though we expected the model to improve after accounting for multicollinearity issues we observed a slight reduction in all the confusion matrix indicators (Table 5), making the model worse.

Lastly, it is important to mention that for logistic regression only, we tried to resolve the class imbalance problem in an alternative way as well, aiming to improve our model. Instead of generating synthetic samples we randomly selected from the dataset with 100 thousand samples, 15 thousand samples that *Returns* equal to 1 and another 15 thousand when *Returns* equal to 0. However, this did not improve our logistic regression results and we attribute that to the fact that SMOTE can help the model learn to find returns within a range better.

C. Random Forest

Our goal is to use the Random Forest model to forecast return occurrences accurately. It is meticulously fine-tuned to harness deterministic modeling while addressing class imbalances. Parameters such as bootstrap=False and class_weight='balanced' ensure a robust approach, accompanied by stringent criteria of min_samples_leaf=2 and min_samples_split=10, preventing overfitting without compromising crucial patterns. The ensemble of 200 n_estimators amplifies predictive capabilities, fortifying the model's overall robustness.

About Table 6, Feature importance analysis from this model reaffirms pivotal factors influencing return predictions. 'Salemonth' retains dominance, showcasing temporal trends significantly impacting return occurrences. Concurrently, the persistent influence of 'ReturnRate' underscores the substantial role played by historical return rates in guiding proactive strategies. Transactional attributes like 'AMT' and 'ORGPRICE' continue to wield considerable impact, signifying their crucial roles in return forecasts. Additionally, 'retail_diff' and 'amt_diff' highlight variations in transactional elements, contributing meaningfully to return predictions.

This iteration of our model exhibits a commendable 77.02% accuracy on the test set, demonstrating its efficacy in discerning return occurrences within our dataset. While the observed Area Under the Curve (AUC) slightly decreased to 0.5926 (Table 7), this nuanced shift emphasizes interpreting predictive performance beyond accuracy metrics. Precision, the accuracy of positive predictions, is 10.71% of the time when the model identifies a return. Also, recall shows that our model captures about 26.42% of all actual return occurrences. The F1 score is 15.24%, describing a balanced and evolving representation of our model's predictive capabilities.

While our model demonstrates adeptness in overall return predictions, the nuanced precision and recall metrics highlight areas for continued refinement. The precision's lower value hints at potential false positives, while the moderate recall suggests space for improvement in capturing more actual return cases. Therefore, our model, while showcasing commendable aptitude, invites continuous refinement to achieve a more precise and comprehensive description of return predictions.

CONCLUSION

The identified correlation between seasonal trends, specific months, and return rates underscores the pivotal role of timing in the precise prediction of return occurrences. This discovery aligns with the insights gathered from the Exploratory Data Analysis (EDA) section, which illustrated an obvious seasonal return trend in June, July, and December. Importantly, this temporal pattern coincides with intervals characterized by promotional activities within the retail companies such as the Christmas season.

Thus, based on our random forest model and the most important features that affect the probability of an item being returned we created a detailed ROI analysis that expresses the total amount of cost that Dillard’s can reduce if the strategic recommendations below are applied. To optimize seasonal inventory management, businesses can strategically adjust inventory levels by anticipating projected return rates during distinct seasons or specific months. Additionally, implementing dynamic pricing strategies allows for the adjustment of prices during high-return periods, ensuring profitability remains robust even in the event of potential returns.

REFERENCES

Dillards Retail Store. *Summerlin*, https://summerlin.com/directory/stores/dillards/. Accessed 15 Nov. 2023.

Smith, Craig. “Interesting Dillards Statistics and Facts.” *DMR*, 16 Mar. 2023, expandedramblings.com/index.php/dillards-statistics-and-facts/.

“Number of Dillard’s Locations in the USA in 2023.” *ScrapeHero*, 25 Oct. 2023, www.scrapehero.com/location-reports/Dillards-USA/#:~:text=How%20many%20Dillards%20stores%20are,Dillards%20stores%20in%20the%20US.

“Our History: Dillard’s Careers.” *Our History | Dillard’s*, careers.dillards.com/Careers/Information/history. Accessed 16 Nov. 2023.

“A more than $761 billion dilemma: Retailers’ returns jump as online sales grow.” Melissa Repko, Jan, 25th, https://www.cnbc.com/2022/01/25/retailers-average-return-rate-jumps-to-16point6percent-as-online-sales-grow-.html

Ward, Colton. “The Hidden Cost of Free Returns.” Elite EXTRA, 14 Apr. 2023, eliteextra.com/the-hidden-cost-of-free-returns/#:~:text=That%20means%20the%20average%20retailer,go%20as%20high%20as%2030%25.

“DILLARD’S, IN C . 2004 ANNUAL REPORT.” Dillard’s. https://www.annualreports.com/HostedData/AnnualReportArchive/d/NYSE_DDS_2004.pdf

APPENDIX

Table 1: Line plot for returns rate across 29 states

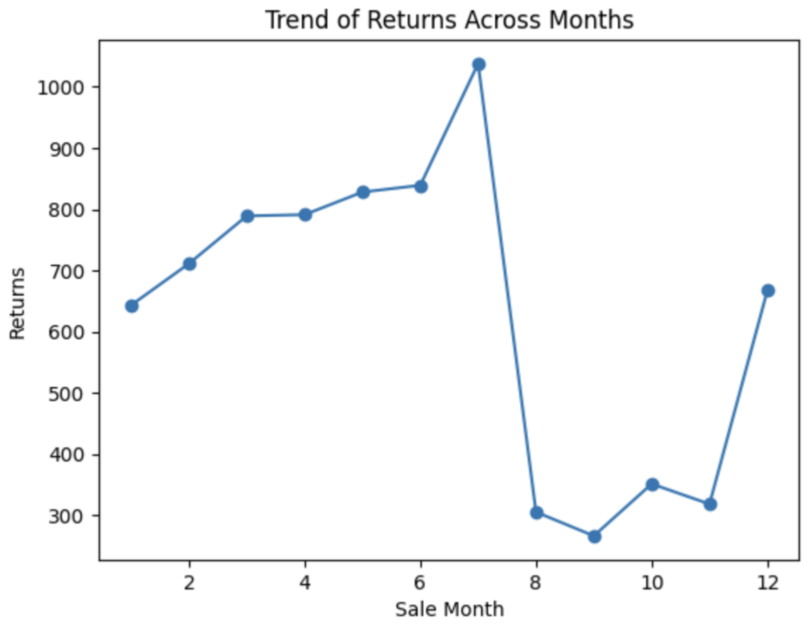


Table 2: Bubble plot of return rate across 29 states

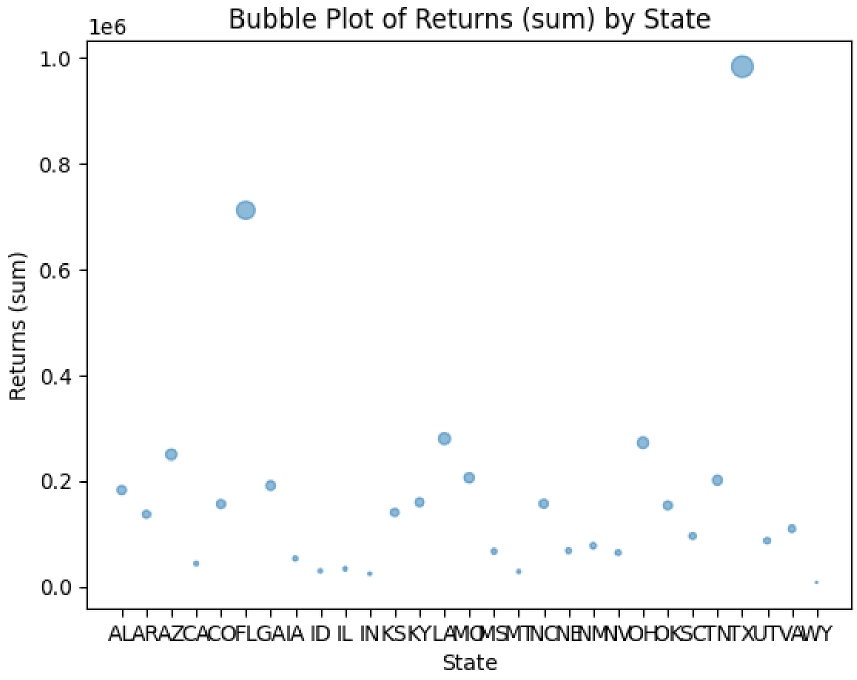
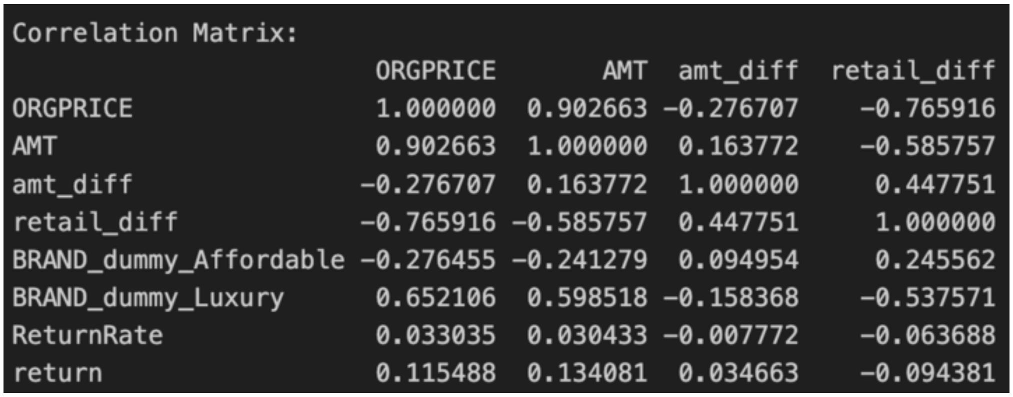
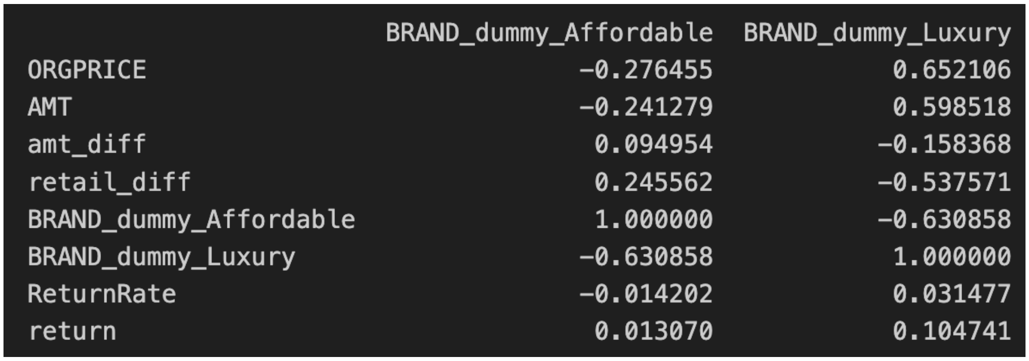


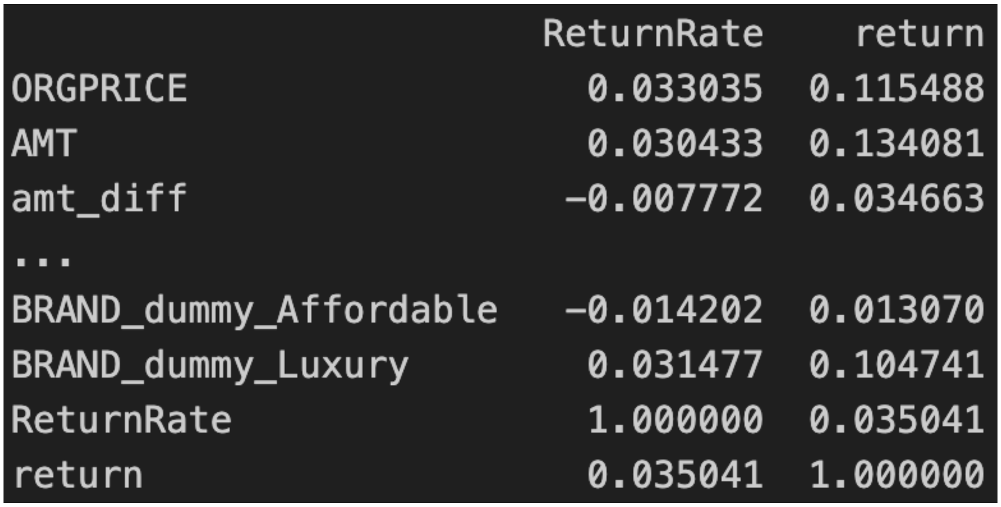
Table 3: Correlation Matrix



```
Correlation Matrix:
                       ORGPRICE       AMT  amt_diff  retail_diff
ORGPRICE               1.000000  0.902663 -0.276707    -0.765916
AMT                    0.902663  1.000000  0.163772    -0.585757
amt_diff              -0.276707  0.163772  1.000000     0.447751
retail_diff           -0.765916 -0.585757  0.447751     1.000000
BRAND_dummy_Affordable -0.276455 -0.241279  0.094954     0.245562
BRAND_dummy_Luxury     0.652106  0.598518 -0.158368    -0.537571
ReturnRate             0.033035  0.030433 -0.007772    -0.063688
return                 0.115488  0.134081  0.034663    -0.094381
```

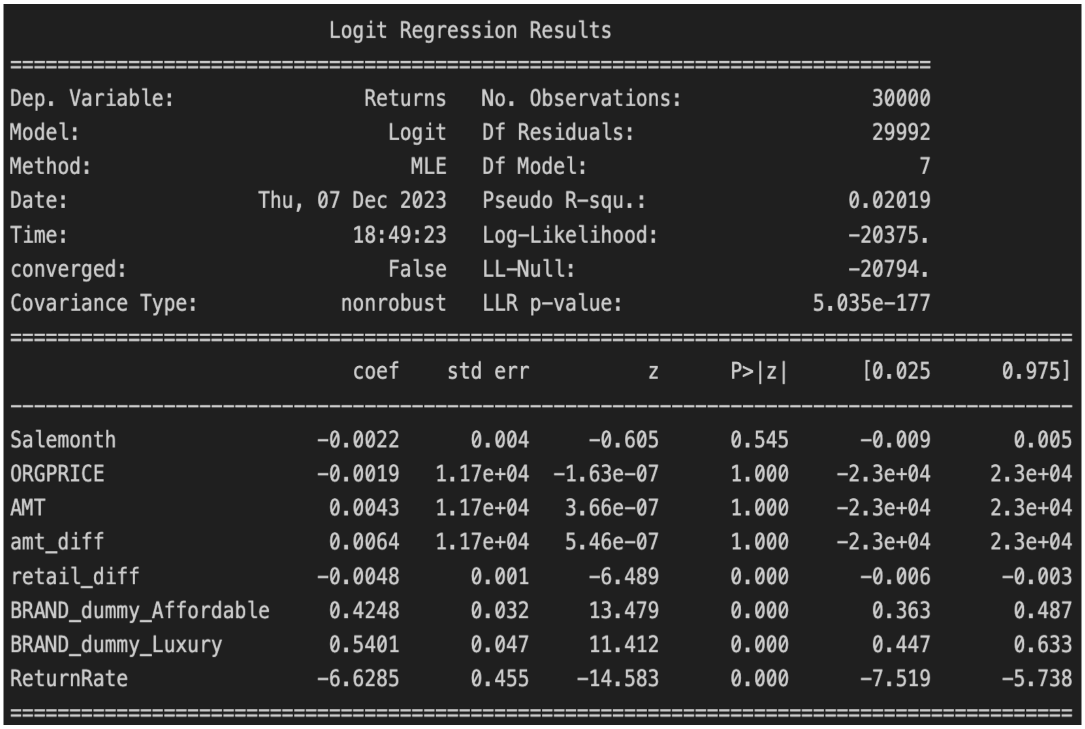


```
                        BRAND_dummy_Affordable  BRAND_dummy_Luxury
ORGPRICE                             -0.276455            0.652106
AMT                                  -0.241279            0.598518
amt_diff                              0.094954           -0.158368
retail_diff                           0.245562           -0.537571
BRAND_dummy_Affordable                1.000000           -0.630858
BRAND_dummy_Luxury                   -0.630858            1.000000
ReturnRate                           -0.014202            0.031477
return                                0.013070            0.104741
```



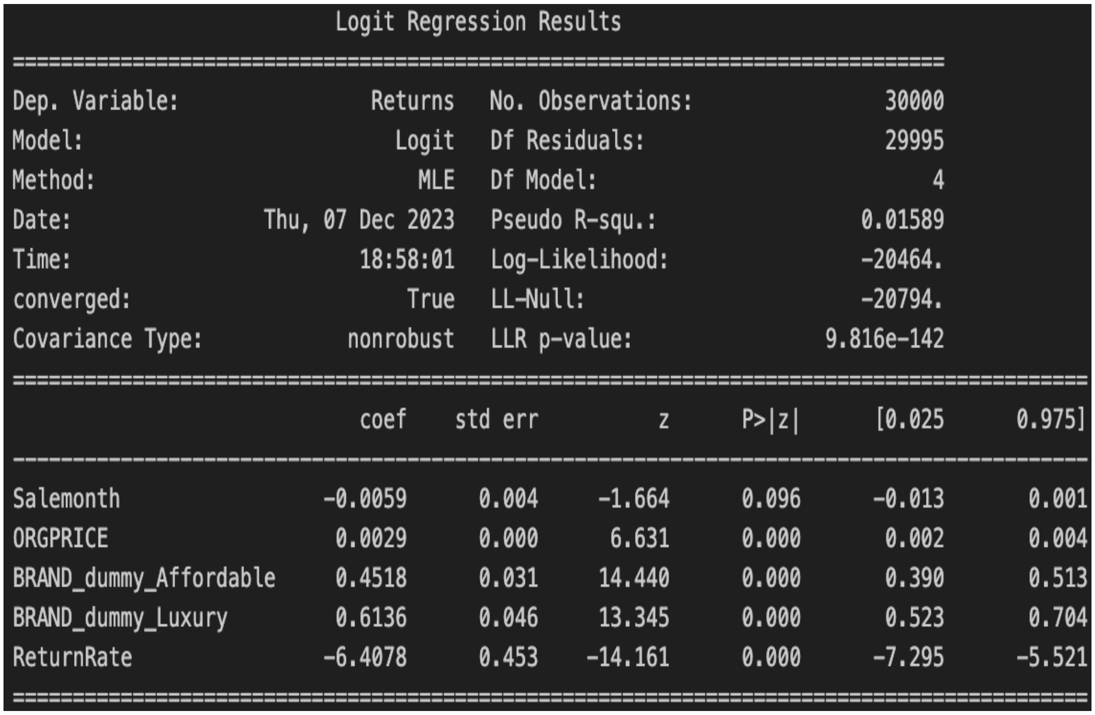
```
                        ReturnRate    return
ORGPRICE                  0.033035  0.115488
AMT                       0.030433  0.134081
amt_diff                 -0.007772  0.034663
...
BRAND_dummy_Affordable   -0.014202  0.013070
BRAND_dummy_Luxury        0.031477  0.104741
ReturnRate                1.000000  0.035041
return                    0.035041  1.000000
```

Model 1: Logistic Regression - Full Model



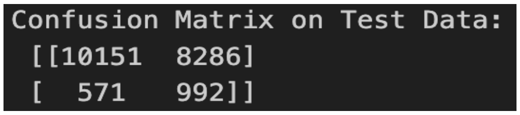
```
                           Logit Regression Results
==============================================================================
Dep. Variable:                Returns   No. Observations:                30000
Model:                          Logit   Df Residuals:                    29992
Method:                           MLE   Df Model:                            7
Date:                Thu, 07 Dec 2023   Pseudo R-squ.:                 0.02019
Time:                        18:49:23   Log-Likelihood:                -20375.
converged:                      False   LL-Null:                       -20794.
Covariance Type:            nonrobust   LLR p-value:                 5.035e-177
==========================================================================================
                            coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------------------
Salemonth              -0.0022      0.004     -0.605      0.545      -0.009       0.005
ORGPRICE               -0.0019   1.17e+04  -1.63e-07      1.000    -2.3e+04     2.3e+04
AMT                     0.0043   1.17e+04   3.66e-07      1.000    -2.3e+04     2.3e+04
amt_diff                0.0064   1.17e+04   5.46e-07      1.000    -2.3e+04     2.3e+04
retail_diff            -0.0048      0.001     -6.489      0.000      -0.006      -0.003
BRAND_dummy_Affordable  0.4248      0.032     13.479      0.000       0.363       0.487
BRAND_dummy_Luxury      0.5401      0.047     11.412      0.000       0.447       0.633
ReturnRate             -6.6285      0.455    -14.583      0.000      -7.519      -5.738
==========================================================================================
```

Model 2: Logistic Regression - Reduced Model

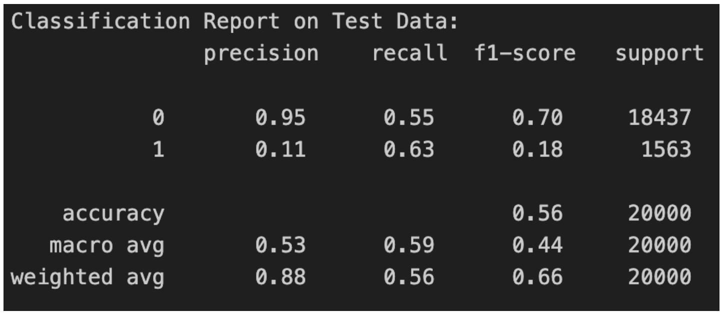


```
                           Logit Regression Results
==============================================================================
Dep. Variable:                Returns   No. Observations:                30000
Model:                          Logit   Df Residuals:                    29995
Method:                           MLE   Df Model:                            4
Date:                Thu, 07 Dec 2023   Pseudo R-squ.:                 0.01589
Time:                        18:58:01   Log-Likelihood:                -20464.
converged:                       True   LL-Null:                       -20794.
Covariance Type:            nonrobust   LLR p-value:                 9.816e-142
==========================================================================================
                            coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------------------
Salemonth              -0.0059      0.004     -1.664      0.096      -0.013       0.001
ORGPRICE                0.0029      0.000      6.631      0.000       0.002       0.004
BRAND_dummy_Affordable  0.4518      0.031     14.440      0.000       0.390       0.513
BRAND_dummy_Luxury      0.6136      0.046     13.345      0.000       0.523       0.704
ReturnRate             -6.4078      0.453    -14.161      0.000      -7.295      -5.521
==========================================================================================
```

Table 4: Confusion Matrix - Full Model

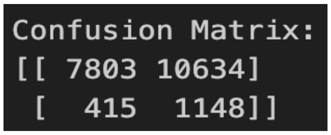


```
Confusion Matrix on Test Data:
 [[10151  8286]
 [  571   992]]
```

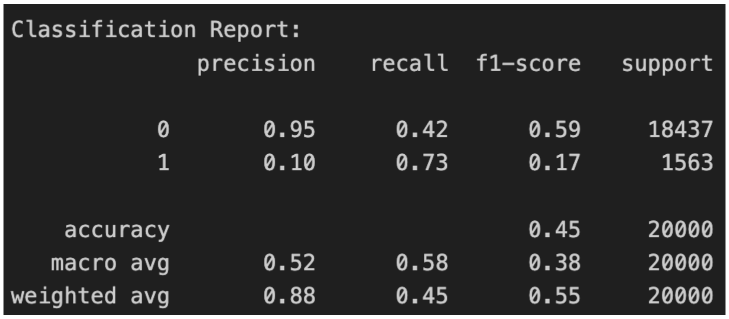


```
Classification Report on Test Data:
              precision    recall  f1-score   support

           0       0.95      0.55      0.70     18437
           1       0.11      0.63      0.18      1563

    accuracy                           0.56     20000
   macro avg       0.53      0.59      0.44     20000
weighted avg       0.88      0.56      0.66     20000
```

Table 5: Confusion Matrix - Full Model



```
Confusion Matrix:
[[ 7803 10634]
 [  415  1148]]
```



```
Classification Report:
              precision    recall  f1-score   support

           0       0.95      0.42      0.59     18437
           1       0.10      0.73      0.17      1563

    accuracy                           0.45     20000
   macro avg       0.52      0.58      0.38     20000
weighted avg       0.88      0.45      0.55     20000
```

Table 6: Random Forest - Feature Importance

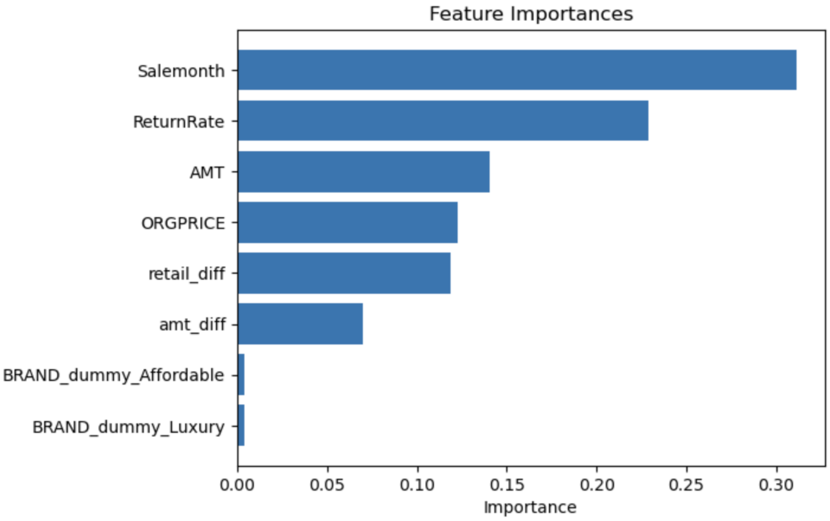


Table 7: Random Forest - ROC Curve

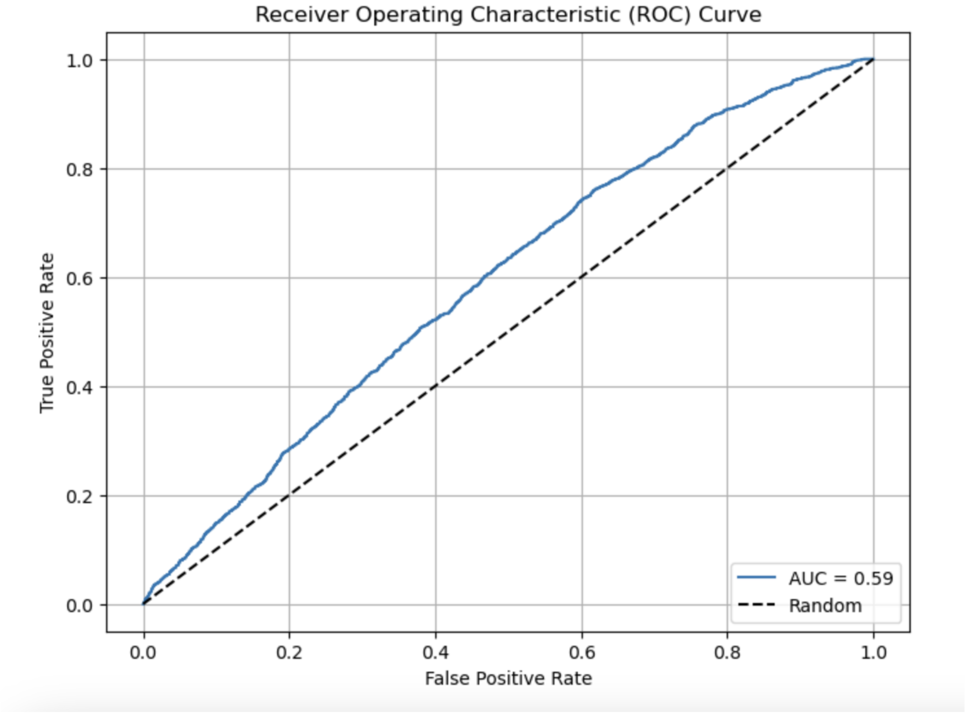
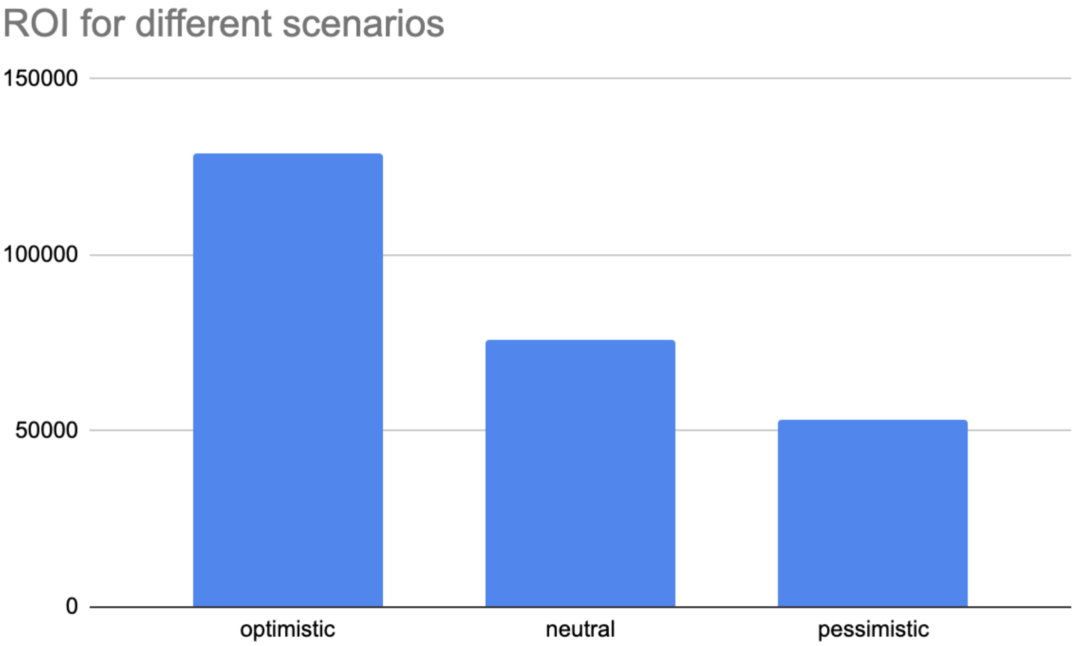


Table 8: ROI Analysis


ROI for different scenarios
optimistic
neutral
pessimistic