Report: CAP4770 Final Project – Project B: Retail

Ethan Alexander, Maiah Jaffa

Introduction

Utilizing the UCI Machine Learning Repository's Online Retail II dataset, we aim to study consumer e-commerce data to create predictive models and extract useful insights, to find trends in consumer behavior, optimize marketing initiatives, and guide corporate plans to improve client retention, boost revenue, and customize marketing strategies.

To make these conclusions, we will use a variety of visualizations, clustering techniques, classification models, regression models, and association rule mining techniques.

Data Preprocessing

To derive the insights desired, the dataset required cleaning, preprocessing, and feature engineering.

For one, the first column name, 'Invoice,' was read in with dummy characters. This required us to rename the column to be compatible with our program.

We also removed all items with an empty 'Customer ID' field. The items with an empty 'Customer ID' field were inner company transactions like bank deposits. Since our focus is to analyze consumer behavior, we chose to remove all transactions that did not represent customer purchases or returns. This way, our analyses would not be affected by irrelevant data.

The dataset contained many sets of multiple rows that represent the same transaction. To eliminate these duplicates and ensure all transactions held an equal weight in our analyses, we kept one item in each set that had the same values for 'Quantity', 'Invoice', 'InvoiceDate', 'Customer ID', 'Price' and 'Stock Code'.

To provide reliable and precise analyses of the data based on time features, we modified and added attributes. First, we ensured that 'InvoiceDate' was of type 'DateTime'. Then, we extracted the hour of day and the day of week when the transaction was made.

To create a feature that encapsulates the weight of each transaction in terms of monetary value, 'TotalSalesAmount' is derived from multiplying the price of the item being bought by the quantity purchased.

The dataset is originally divided into two spreadsheets: one representing transactions from 2009 to 2010 and the other representing transactions from 2010 to 2011. While comparing these two to each other proved insightful and warranted separation in a few aspects, most of our analyses would benefit from a conglomeration of both datasets. As such, we created a separate DataFrame to contain all data from both sets. This way, we would be able to utilize the separation when fruitful as well as use all the data.

Another distinction between transactions was whether they were purchases or returns. In some analyses, weighing both together is insightful, as the combination of both can represent net value. However, some analyses require looking at particularly one of purchases or returns, and so we created two separate DataFrame objects. To determine which transactions represented purchases and which represented returns, we split the dataset by whether 'Quantity' contained a positive or negative value. This approach aligns with the conventions of the original dataset.

Methodology

Clustering

To divide the clientele by preferences and purchasing habits, K-Means Clustering was employed.

Model Architecture

K-Means Clustering is a partition-based clustering algorithm that groups data into k clusters. The architecture revolves around centroids, Euclidean Distance, and dimensionality.

Optimization Policy

K-Means minimizes inertia or the within-cluster sum of squared distances. Inertia represents the compactness of clusters by measuring the total squared distance between each data point in a cluster and its cluster centroid. A lower inertia value implies that the cluster is more compact and has a strong identity. The Elbow Method is used to determine the value for k which optimizes the inertia in respect to a balance between compactness and simplicity.

Training Process

To ensure that all features utilized in the K-Means Clustering contribute equally to the distance metric, the values for the features were standardized. The clustering algorithm initially selects random centroids for a predefined number of clusters. Then the centroids are updated as the mean of all data points assigned to each cluster. These steps are repeated until convergence is reached. The final cluster assignments represent groups of data points that are as compact and distinct as possible.

Classification

Model Architecture

This model uses the RandomForestClassifier in predicting purchase behavior and cart abandonment. Constructing decision trees in training and then outputting the mode of classes for classification tasks. The model uses one hundred estimator trees for thorough predictions. The model uses session characteristics, time features, customer history, and the current session to predict purchasing. While for card abandonment prediction the model uses features relation to the cart, session time data, and whether the customer is returning to predict cart abandonment.

Optimization Policy

The RandomForestClassifier minimized Gini impurity therefore helping to reduce the probability of incorrect classification in the initial labeling of elements in the subset. Through having a lower Gini impurity value, the model can better separate different classes.

Training Process

In training the classification model, the features are scaled to prepare the data for analysis. The data is then split into two sets: training and testing sets. After that the data is fitted using decision trees which helps improve predicted accuracy. The probability of likelihood of a purchase is then followed out and the model performance is compared to the test set for accuracy.

Regression

Model Architecture

In the regression model the RandomForestRegression is used to create a regression model to predict future sales. Incorporating items in the cart, the cart value, session length, time of day and week, and it the customer has prior purchases. This type of model works as it is not likely to be overfit.

Optimization Policy

The RandomForestClassifier helps with optimization as mean squared error is minimized. Further the model uses decision trees which help average the predictions and generalize results.

Training Process

In training the regression model the features were normalized with StandardScaler. The data is then split into two sets: training and testing sets. The model is trained on these features, the patterns from the customer data and the resulting sales values.

Experiments

Countries by Revenue

This algorithm determines the top countries to total revenue by grouping the data based 'Country' and aggregating the revenue values within each group. It calculates the sum of revenues for each country, providing a summary of total contributions. The results are then sorted in descending order to identify the top five countries with the highest aggregated revenue.

Transactions by Hour

This algorithm analyzes transaction patterns by grouping the dataset based the hour of day at which the transaction was performed and counting the number of transactions within each group. For each hour, it calculates the total number of invoices, representing the transaction volume during that period. The result is a summary of transaction counts categorized by hour, which can reveal patterns in activity levels throughout the day.

Sales by Day of Week

This algorithm calculates total sales for each day of the week to analyze sales patterns over time. It groups the dataset based on the day of the week when the transaction was performed and computes the sum of sales values within each

group. The result is a summary showing the total sales amount for each day, providing insights into daily sales performance.

Top Products Sold

This algorithm identifies the top-performing products based on the total quantity sold. It begins by grouping the dataset by product descriptions, aggregating the sales quantities for each product to calculate the total units sold. The aggregated results are then sorted in descending order, prioritizing products with the highest sales volume. Finally, the algorithm extracts the top ten entries, representing the most popular or best-selling products.

Distribution of Purchase Quantities

This algorithm generates a histogram that provides a visual representation of how purchase quantities are distributed, highlighting trends such as the most common purchase sizes. It is focused on transactions with quantities lower than 50 units to highlight typical consumer transactions without skew caused by a few high-quantity transactions.

Total Sales by Date

This algorithm calculates the total sales amount for each day represented in the dataset. First, the dates are extracted from the invoice date information. Then, the algorithm groups the data by dates, aggregating the total sales for each day. The resulting DataFrame contains the total sales for each unique date. This data and its complementary visualization provide a clear trend of total sales over time, helping identify seasonal and yearly patterns and fluctuations in daily sales performance.

Clustering Customers by Sales and Quantities Purchased

This algorithm performs customer segmentation using K-Means clustering based on total quantity purchased and total sales amount. First, it groups the data by customer ID, summing the quantity and total sales for each customer. The data is then scaled to standardize the features. The elbow method is used to determine the optimal number of clusters by calculating the inertia for different values of k. The optimal k is chosen at the "elbow" point, where inertia starts to decrease at a slower rate. K-Means clustering is applied with k=4, and the customers are assigned to clusters. The results are visualized using a scatter plot of total quantity purchased vs. total sales amount, colored by cluster. Finally, the average quantity and sales per cluster are displayed by inverting the scaled centroids.

Clustering Customers by Day of Week and Time of Purchases

Similarly to clustering based on sales and quantities purchased, this algorithm segments customers based on the day of the week and time of day of their purchases using K-Means clustering. First, it maps the day of week to numerical values to encode the days as integers, allowing for numeric computation. The data is then grouped by customer ID, and the average day of the week and the average time of day are computed for each customer. These values are standardized to ensure that both features contribute equally to the clustering process. The elbow method is applied to determine the optimal k. The K-Means algorithm is then used to assign customers to clusters, and the results are visualized using a scatter plot of average day of the week vs. average time of day. Finally, the centroids of the clusters are decoded back to the days of the week to interpret the cluster centers in terms of actual days and times.

Discussion

Countries by Revenue

Out of the 40 countries included in the dataset, the vast majority of revenue stems from the United Kingdom. Ireland, Netherlands, Germany, and round out the top five countries by revenue. This could imply where advertising efforts should be prioritized. With countries outside of the UK, more exposure to the online store would boost traffic greatly.

Transactions by Hour

Transactions hit their peak in the early afternoon, with noon being the most common hour for purchases. This provides insight on when promotional offers or discounts should be scheduled to maximize engagement and sales. Curiously, the hour ranges for data for 2009-2010 and 2010-2011 are slightly different. The 2009-2010 data contain transactions from hour 07 to hour 21, while 2010-2011 contains transactions from hour 06 to hour 20. Depending on how transactions are performed and if the online store has hours of operation akin to a brick-and-mortar store, this trend is important to track when the optimal hours of operation are.

Sales by Day of Week

Sales are relatively uniform across the five weekdays, with Thursday having the most sales. The weekends are considerably lower, with Sunday having about half

the sales of the weekdays and Saturday having a miniscule fraction. The 2010-2011 data contain no entries for Saturday transactions, implying that the store may not be active on Saturdays. Advertising campaigns revolving around weekend discounts could boost Sunday sales.

Top Products Sold

The top products graph shows which store products have been the most popular in terms of quantity purchased. World War II Gliders are the most popular item by a near 20,000-item margin. To capitalize off the gliders' popularity, it would be wise to create similar items that appeal to a similar audience, such as other gliders or other World War II themed toys and items.

Distribution of Purchase Quantities

The distribution of quantity in purchases graph shows a large skew towards small purchase of under 5 items. To increase the average number of items bought purchase, discounts involving multiple items such as Buy Two Get One Free or online advertisements recommending other items when customers are about to checkout.

Total Sales by Date

The total sales by date graph shows a relatively consistent daily sales rate, but the months of October, November, and December appear to have the highest sales. This is likely due to the holiday season and gift-giving traditions. Deals and sales related to the holidays could build on this popularity even more, and sales at other times of the year may build additional traffic in the less momentous months.

Clustering Customers by Sales and Quantities Purchased

Clustering by sales and quantities purchased provides a segmentation by which customers with accounts can be sent promotions targeted to them. The customers in clusters with lower sales and quantities can be sent deals and rewards to inspire them to shop a second time and lead to a consistent relationship. The customers in clusters with higher sales and quantities purchased can be sent promotions such as discounts if a friend of theirs makes a purchase on the website. It is likely that these customers are loyal to the store and would both be happy and be incentivized to recommend it to the people in their lives.

Clustering Customers by Day of Week and Time of Purchases

Clustering customers by the day and time at which they tend to shop can lead to valuable insights as to when advertisements should be distributed to customers. For example, for the cluster of people who tend to shop on Tuesday mid-afternoon, advertisements could be sent via email in the early afternoon to remind customers of the store at a time at which they are likely able and willing to shop.

Conclusion

The analysis revealed several critical insights into customer behavior and sales trends. Revenue is predominantly generated from the United Kingdom, with Ireland, the Netherlands, and Germany following, indicating potential for growth by increasing visibility in non-UK markets. Cart abandonment does not happen frequently but does occur during the beginning and end of the day. Transactions peak around noon, suggesting that promotional campaigns should target early afternoon to maximize engagement. Weekday sales are relatively uniform, with Thursday being the strongest, while weekends, especially Saturday, show significantly lower activity. Product analysis identified top-selling items like World War II Gliders, presenting opportunities to develop similar products to appeal to that customer segment. Additionally, clustering customers by purchase behavior and shopping times provided actionable insights for targeted promotions, such as incentivizing repeat purchases for low-frequency shoppers or timing advertisements based on customers' preferred shopping periods. These findings can guide strategic decisions to enhance sales and customer retention.