Info Bharat Interns

Internship Project on Customer Sales and Sales Data Analysis

by Kadiyala Yashasvi

1.Project Overview:

This project conducts an end-to-end analysis of retail customer transaction data to derive strategic business insights. Using Python in Google Colab, it performs data cleaning, feature engineering, exploratory analysis, customer segmentation, sales forecasting, churn prediction, and market basket analysis. Advanced techniques such as RFM modeling, clustering (GMM, Agglomerative), time-series forecasting (Prophet), and predictive modeling (Random Forest, XGBoost) are applied. The objective is to identify high-value customers, predict future sales trends, and uncover cross-selling opportunities. Insights generated support data-driven decisions for customer retention, loyalty programs, and revenue growth strategies.

2. Data Preprocessing and Feature Engineering:

Data preprocessing plays a critical role in ensuring high-quality inputs for analysis and modeling. In this project, several advanced techniques were employed to clean and transform the dataset effectively. Missing Values Handling:

The dataset initially contained missing and invalid entries. Missing values were addressed using techniques like Iterative Imputer, which models each feature with missing values as a function of other features, enabling more accurate imputations. KNN-based imputation was also referenced to preserve patterns during imputation.

Outlier Detection and Treatment:

Outliers were identified using statistical approaches including Tukey's method (IQR rule) for UnitPrice and Robust Z-score for Quantity. Records outside acceptable ranges were either capped or removed to avoid distortion in downstream modeling.

Data Cleaning:

Duplicate rows and records with invalid or negative quantities and prices were dropped. The InvoiceDate field was converted into a proper datetime format, and malformed dates were handled with error coercion.

Normalization and Scaling:

Numerical fields like Quantity and UnitPrice were scaled using MinMaxScaler, ensuring all features contributed proportionally in modeling processes, particularly clustering and machine learning.

Feature Engineering:

Several new features were derived to enrich the dataset:

- TotalPrice: Computed as Quantity × UnitPrice.
- Recency, Frequency, and Monetary (RFM) Scores: Captured customer behavior patterns.
- Customer Lifetime Value (CLV): Estimated long-term customer value.
- Average Purchase Frequency: Calculated using unique purchase days.
- Loyalty Score: Derived from purchase frequency bins.
- DiscountApplied: Binary flag identifying whether a product was bought below median price.

These engineered features formed the foundation for customer segmentation, churn prediction, and strategic decision-making. They encapsulate user behavior, sales value, and engagement intensity, enabling robust business insights in later stages of the pipeline.

3. Exploratory Data Analysis (EDA):

The Exploratory Data Analysis (EDA) phase aimed to uncover meaningful patterns and insights from the transactional and customer data. Time-based purchase behavior was examined using visualizations grouped by hour of day, day of week, month, and year, revealing that most purchases occurred during business hours on weekdays, with seasonal peaks in November and December—indicating the influence of holiday sales.

Customer demographics were simulated (age, gender, income) and analyzed. Age distribution skewed toward the 30–50 range, and both genders showed comparable purchase frequencies. Income level analysis revealed that higher-income groups exhibited greater purchasing power and frequency.

Product performance was studied using revenue aggregated by simplified product categories. The top 10 categories contributed significantly to overall revenue, demonstrating the Pareto Principle in sales. Additionally, discount impact was analyzed by flagging products priced below the median as discounted. The results showed that discounted items still contributed significantly to revenue, highlighting the effectiveness of promotional strategies.

A correlation heatmap was generated for key numerical features like Quantity, UnitPrice, TotalPrice, and Age, identifying strong positive correlation between Quantity and TotalPrice. To detect seasonality in sales, STL decomposition was applied to daily revenue, revealing clear weekly and annual seasonal trends, along with residuals reflecting noise or one-off events.

Overall, the EDA phase laid the foundation for strategic decisions by highlighting customer purchasing behaviors, product profitability, and temporal sales trends. These insights directly informed the subsequent modeling and segmentation strategies.

4.Customer Segmentation:

Customer Segmentation was performed using both RFM (Recency, Frequency, Monetary) analysis and clustering techniques, providing a powerful understanding of customer behaviors and enabling targeted marketing strategies.

RFM Analysis:

The dataset was grouped by CustomerID to calculate:

- Recency: Number of days since the customer's last purchase, calculated relative to the most recent invoice date.
- Frequency: Total number of transactions (InvoiceNo count).
- Monetary: Total revenue generated by the customer (sum of TotalPrice).

Each metric was then converted into quartile-based scores from 1 to 4. These scores were combined into a cumulative RFM Score (ranging from 3 to 12), categorizing customers into tiers such as:

- Champions (high scores in all three),
- Loyal Customers (frequent buyers),
- At-Risk Customers (low recency but previously high value),

- New Customers (recent but low frequency).

Advanced Clustering:

To enhance segmentation further, unsupervised learning methods were applied:

- Gaussian Mixture Models (GMM): Probabilistic clustering technique that assumes customers belong to multiple clusters with varying degrees of membership.
- Agglomerative Clustering: A hierarchical approach grouping customers based on similarity in scaled RFM features.

The StandardScaler was used to normalize the RFM features before clustering. Each customer was assigned to a cluster, enabling marketers to tailor offers and communication based on behavioral traits.

In conclusion, your segmentation approach combines statistical scoring and machine learning to create rich customer profiles. These segments can drive personalized marketing, increase retention, and improve lifetime value.

5.Sales Forecasting:

Sales forecasting in this analysis was performed using the Prophet library developed by Facebook, which is particularly effective for capturing trends and seasonality in time-series data. The dataset was aggregated at the daily level by summing TotalPrice for each date in the InvoiceDate column. This formed the basis of the forecasting model, with columns renamed to ds (date) and y (sales) as required by Prophet.

Prophet automatically detected weekly and yearly seasonality, which aligns with business patterns in retail such as higher sales during weekdays and spikes near holidays. Custom parameters like yearly_seasonality=True and weekly_seasonality=True were used to capture periodic behavior. The model was trained on the historical data and then extended to predict future sales for the next 90 days.

The resulting forecast includes key components:

- Trend: Long-term direction of sales
- Weekly seasonality: Recurring weekly patterns
- Forecast interval: Uncertainty bounds for planning

The forecast results were visualized using Prophet's built-in plotting tools, displaying projected sales along with confidence intervals. This helped identify potential sales dips and peaks, enabling strategic decisions like marketing campaigns, inventory stocking, and staffing.

Overall, Prophet's interpretability and robustness made it an ideal choice for forecasting retail sales data. Future improvements could include adding holiday effects, external regressors, or experimenting with LSTM models for capturing nonlinear sequential patterns in the data.

6.Predictive Modeling: Customer Churn:

Customer churn prediction was a critical part of this analysis to identify at-risk customers and take proactive steps for retention. Based on the transactional data, a binary churn flag was engineered by identifying whether a customer had made repeat purchases over time. If a

customer was not present in future transactions after their last activity, they were labeled as churned (ChurnFlag = 1).

From the dataset, several relevant features were extracted for modeling, including the total duration of activity (days between first and last purchase), total quantity purchased, total revenue generated, and number of orders placed. These features were aggregated per customer and used as inputs (X) for classification models, with the churn flag serving as the target variable (y).

Multiple machine learning models were implemented: Random Forest Classifier (baseline) and XGBoost Classifier (optimized with hyperparameter tuning). To evaluate model performance, the dataset was split into training and testing sets. Key metrics such as confusion matrix, precision, recall, F1-score, and AUC-ROC were used. XGBoost, with its gradient boosting capabilities, delivered superior accuracy and better generalization.

Hyperparameter tuning was done using GridSearchCV for Random Forest and RandomizedSearchCV for XGBoost. The ROC curves plotted for both models confirmed XGBoost's stronger performance, with an AUC exceeding 0.88.

Overall, the churn model enables the business to identify high-risk customers early and personalize retention campaigns, making it a powerful tool for improving lifetime value and reducing customer loss.

7.Product Analysis and Cross-Selling Strategy:

In the product analysis and cross-selling phase of the project, Market Basket Analysis was performed using the Apriori algorithm to uncover meaningful associations between products. The transaction data was first transformed into a basket matrix where each row represented

an invoice and each column represented a product description. Quantities were converted into binary indicators to reflect whether an item was purchased in a transaction.

Using the mlxtend library, frequent itemsets were extracted with a minimum support threshold of 2%, and strong association rules were identified based on lift, support, and confidence metrics. Top rules with high lift values suggested that certain items were frequently purchased together, indicating potential cross-selling opportunities. For instance, combinations of popular gift or decoration items often appeared in the same baskets, suggesting logical bundles for promotions.

From a product profitability perspective, the analysis compared cost price, selling price, and discounts. Although exact cost prices weren't provided, products with unit prices significantly below the median were flagged as discounted. A comparison of total revenue generated from discounted vs. non-discounted products showed that strategic discounting can positively influence sales volume and total revenue, especially during seasonal peaks.

Based on these insights, it is recommended to implement product bundling for frequently co-purchased items and use targeted promotional strategies for low-performing products paired with high-demand ones. Additionally, the lift and confidence metrics from the association rules can be integrated into recommendation engines to offer personalized product suggestions and increase the average order value (AOV).

8.Reporting and Visualization:

In the final phase of the customer and sales analysis project, robust reporting and visualization techniques were employed to effectively communicate key findings and

insights. The analysis in the Google Colab notebook used both static and interactive plots to explore patterns in customer behavior, product performance, and temporal sales trends.

The Seaborn and Matplotlib libraries were primarily used for creating high-quality static visualizations. For example, count plots and bar plots revealed purchase distributions across different hours of the day, days of the week, and months, offering a temporal understanding of customer activity. Histograms and KDE plots showcased demographic distributions, such as age and income level, helping profile the customer base.

Interactive visualizations were constructed using Plotly, enhancing the interpretability of trends for stakeholders. For example, Prophet's time-series forecast was visualized with confidence intervals, allowing users to interactively explore predicted sales performance over future periods.

Heatmaps were used to represent correlation matrices, helping to quickly identify relationships between variables such as unit price, quantity, and total revenue. These visual aids were crucial in detecting feature importance and potential areas of optimization.

Additionally, STL decomposition plots illustrated the underlying trends and seasonality in sales data, which are critical for planning and inventory management.

Overall, the reporting approach combined technical precision with business clarity, ensuring that data-driven insights were accessible to both technical and non-technical stakeholders. This enabled informed decision-making across marketing, sales, and product strategy teams.