**Methodology**

## **1. Data Loading and Initial Exploration**

The analysis began by sourcing data from two Excel sheets containing online retail transaction records from 2009 to 2011. These datasets were concatenated into a single, combined dataset to enable a comprehensive analysis over the entire period.

Initial exploration of the data involved displaying the first few rows to inspect the structure and identify potential issues. Basic statistical summaries, such as mean, median, and standard deviation, were calculated for numerical columns to provide an overview of the data distribution, helping to understand the central tendencies and spread within the dataset.

To ensure data quality, several common issues were addressed:

* **Missing Values**: Missing values were examined across the dataset, with a particular focus on essential columns such as CustomerID, which is crucial for customer-level analysis. Rows with missing or incomplete CustomerID values were removed to maintain the accuracy of customer behavior insights.
* **Duplicates**: Duplicate rows were identified and removed, as they could distort results and analyses if left unaddressed.
* **Irregularities**: Invalid or inconsistent values, such as negative quantities or prices, were identified. These were likely due to data entry errors and were removed to maintain data integrity.

Additionally, columns that were irrelevant to the objectives of the analysis were removed, ensuring that only relevant features remained. This streamlined the dataset, preparing it for further processing, feature engineering, and model development.

**Key Observations**

During the cleaning process, several observations were noted:

* A significant portion of the data lacked CustomerID values, which were essential for customer-specific analyses. These entries were appropriately handled to ensure the reliability of the results.
* Instances of negative quantities and prices were observed, indicating possible data entry errors. These records were removed to prevent inaccuracies.
* Certain columns, determined to be unnecessary for the analysis, were dropped to simplify the dataset.

With these issues addressed, the dataset was then prepared and optimized for the subsequent stages of feature engineering and model development.

## **Data Preprocessing and Feature Engineering**

The first step was to clean the data. Duplicate rows and any records missing important information, like “CustomerID”, were removed. Entries with negative values in the “Quantity” or “Price” columns were also excluded because these likely represented mistakes or returns that could make the results less accurate.

After cleaning, new features were created to make the data more useful. For example:

* **Total\_Amount**: Calculated by multiplying “Price” by “Quantity”, this shows the total value of each transaction.
* **Hour**, **Day\_of\_Week**, and **Month**: These features were extracted from the transaction dates to identify patterns in when customers shop.
* **Is\_Weekend**: A new feature that shows whether the purchase happened on a weekend, helping to compare weekday and weekend shopping habits.

Better understand how customers behave, **Recency, Frequency, and Monetary (RFM) Analysis** was done:

* **Recency**: How long ago the customer made their last purchase.
* **Frequency**: How many times a customer has purchased.
* **Monetary Value**: The total amount of money a customer has spent.

Other metrics, like the average, smallest, and largest purchase amounts, as well as how much these vary, were also calculated for each customer. This provided a clearer picture of their buying habits.

Another step was to calculate how often customers make purchases. For each customer, the average time between purchases and how much this varied were measured. This showed whether customers shop regularly or occasionally.

Finally, outliers (very high or very low values) were handled to avoid them affecting the models. These extreme values were limited to a reasonable range to keep the data clean and reliable.

With these steps, the dataset was fully prepared for creating predictive models. It included useful new features and was cleaned to ensure accurate and meaningful results.

* 1. **Data Splitting and Feature Scaling**

Dividing the data into **training**, **validation**, and **test sets** is an essential step in machine learning to ensure the model learns effectively and performs well on new data. This was done using an 80-10-10 split:

* **80% Training: The majority of the data is used for training so the model has enough examples to learn patterns effectively.**
* **10% Validation: A smaller portion is used for validation to ensure the model generalizes well without wasting too much data that could be used for training.**
* **10% Test: Another 10% is reserved to give a clear, unbiased estimate of how well the model will perform on future data.**

Next, feature scaling is applied to make all the numbers in the dataset have a similar range. This helps the models focus on the patterns in the data instead of the size of the numbers. StandardScaler is used to adjust the data so that every feature had an average of zero and a consistent range. This step was done separately for the training, validation, and test sets to prevent any information mixing between them.

These steps ensured the data was clean, evenly scaled, and ready for the machine learning models to perform well.

* 1. **Model Development**

**XGBoost Model**

The first model developed was an XGBoost model, a powerful algorithm that excels in handling structured data. To further enhance its performance, hyperparameter tuning was conducted using Optuna. Optuna systematically explored different hyperparameter settings, such as the maximum depth of decision trees (max\_depth) and the learning rate (learning\_rate), to identify the optimal configuration for achieving high accuracy.

**Neural Network Model**

The second model utilized a Neural Network consisting of three layers of neurons designed to efficiently process the data. To prevent overfitting, an early stopping technique was implemented. This technique monitors the model's performance on a validation set and stops the training process when performance on the validation set stops improving. This approach helps prevent the model from fitting too closely to the training data and enhances its ability to generalize.

Both models were trained on the training dataset, where they learned patterns and relationships within the data. After training, their performance was evaluated on the validation dataset to assess their ability to accurately predict customer value and generalize to new, unseen data.

These carefully developed models underwent a rigorous evaluation process to identify the most effective model for predicting customer value.

The XGBoost and Neural Network models were evaluated on a test dataset. Several key metrics were calculated to assess their performance: Mean Squared Error, Root Mean Squared Error, Mean Absolute Error, and R-squared. These metrics helped determine the accuracy of the models in predicting values.

**Mean Squared Error (MSE):**

where:

* n: number of data points
* yi​: actual value of the i-th data point
* y^​i​: predicted value of the i-th data point

A lower MSE indicates a more accurate model, as it minimizes the average squared difference between predicted and actual values.

**Root Mean Squared Error (RMSE):**

RMSE=

​

RMSE is the square root of MSE, providing a more interpretable error metric in the same units as the target variable.

**Mean Absolute Error (MAE):**

MAE calculates the average absolute difference between predicted and actual values, making it less sensitive to outliers compared to MSE and RMSE.

**R-squared (R²) Score:**

where:

* yˉ​: mean of the actual values

R² represents the proportion of variance in the target variable explained by the model. A higher R² score signifies a better-fitting model.

By calculating these metrics, a comprehensive evaluation of the models' predictive accuracy and overall performance was achieved.

To evaluate the effectiveness of our XGBoost and Neural Network models, we delved into a comprehensive analysis of their performance on the test dataset. We calculated several key metrics, including:

* **Mean Squared Error (MSE):** This metric measures the average squared difference between predicted and actual values. A lower MSE indicates a more accurate model.
* **Root Mean Squared Error (RMSE):** The square root of MSE, RMSE provides a more interpretable measure of error in the same units as the target variable.
* **Mean Absolute Error (MAE):** This metric calculates the average absolute difference between predicted and actual values. MAE is less sensitive to outliers compared to MSE and RMSE.
* **R-squared (R²) Score:** R² represents the proportion of variance in the target variable explained by the model. A higher R² score signifies a better-fitting model.

**Additional Diagnostic Steps**

To gain deeper insights into our models' performance and identify potential areas for improvement, we conducted the following additional analyses:

* **Feature Importance Analysis (XGBoost):** By visualizing the feature importance scores, we were able to understand which features contributed most significantly to the model's predictions. This information can be invaluable for feature selection and model interpretation.
* **Learning Curve Analysis (Neural Network):** Plotting learning curves helped us assess the model's training and validation performance as the training set size increased. This analysis allowed us to identify potential overfitting or underfitting issues.
* **Predicted vs. Actual Value Plots:** Visualizing the relationship between predicted and actual values provided a clear picture of the model's accuracy and any systematic biases.

Through this rigorous evaluation process, we were able to select the optimal model for our specific task, ensuring the highest level of predictive accuracy and reliability.