College Dublin
Computing • IT • Business

Assessment Cover Page

Student Full Name	Yumiko Maria Bejarano Azogue
Student Number	2024144
Module Title	Predictive Analytics
Assessment Title	PDA Report
Assessment Due Date	08/07/2024
Date of Submission	14/08/2024

Declaration

By submitting this assessment, I confirm that I have read the CCT policy on academic misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source.

I declare it to be my own work and that all material from third parties has been appropriately referenced.

I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

Introduction

This report presents the findings from the PDA Report. Each task involves the application of advanced techniques covered in the module, aimed at extracting meaningful insights and providing recommendations based on the analysis. These tasks demonstrate the practical use of data analysis methods and their relevance in making informed decisions.

1: Clustering and Classification Models

Question 1:

Use your data to perform a clustering model. Justify the choice of your algorithm and add the column with the outcome on the dataset to perform two classification models. Determine which one would be the best one in this case. Summarize your findings, discussing the model's performance and any insights gained from the analysis.

1.1. Exploratory Data Analysis (EDA)

Initial Data Examination: We began by loading the dataset from Online Retail.xlsx. The first 10,000 rows of the dataset were loaded to speed up processing, allowing us to perform a preliminary analysis.

Next, we presented a general overview of the dataset:

	#	Column	Non-Null Count	Dtype	Memory
0	0	url	10000	non-null	object
1	1	timedelta	10000	non-null	float64
2	2	n_tokens_title	10000	non-null	float64
3	3	n_tokens_content	10000	non-null	float64
4	4	n_unique_tokens	10000	non-null	float64
...	...	...	...	...	...
56	56	title_subjectivity	10000	non-null	float64
57	57	title_sentiment_polarity	10000	non-null	float64
58	58	abs_title_subjectivity	10000	non-null	float64
59	59	abs_title_sentiment_polarity	10000	non-null	float64
60	60	shares	10000	non-null	int64

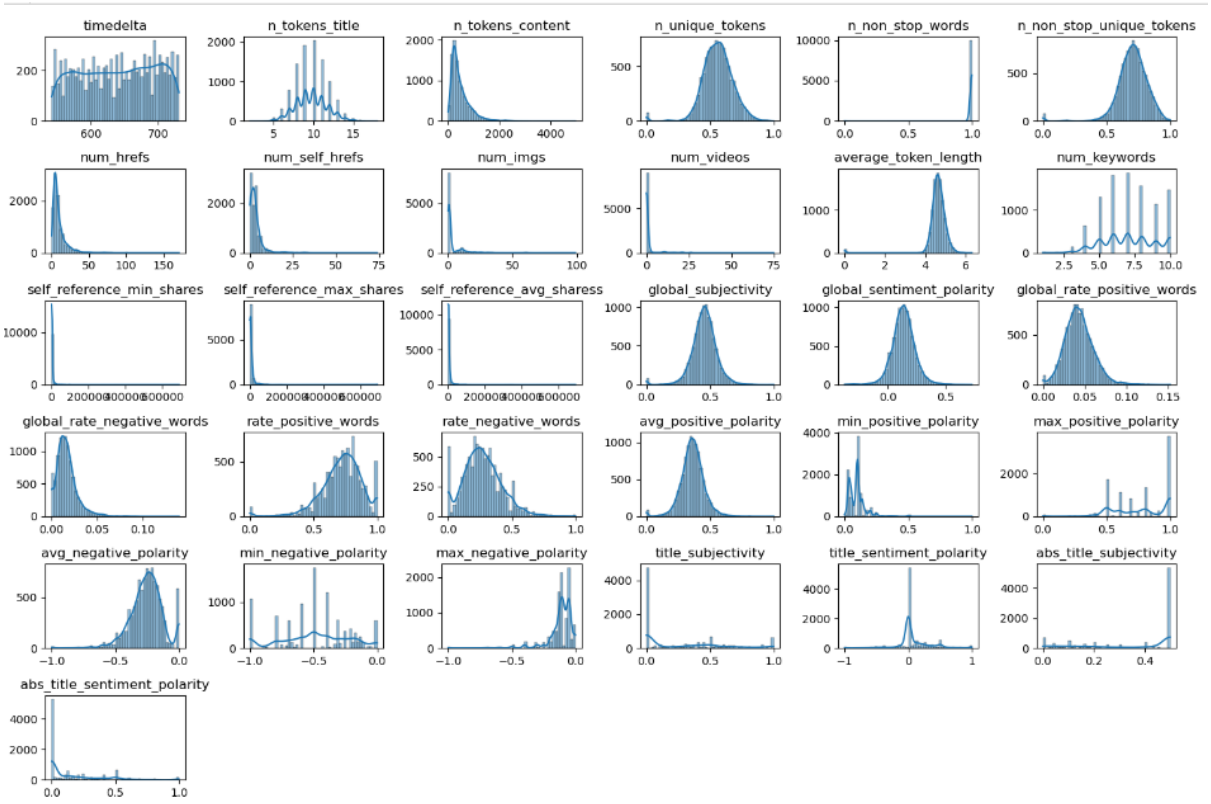
Handling Missing Values: When examining the missing values, we found that the CustomerID column had missing entries. Since CustomerID is crucial for clustering and classification, the rows with missing CustomerID were removed. This step ensured that our analysis was based on complete data.

```
url                             0
 timedelta                      0
 n_tokens_title                 0
 n_tokens_content               0
 n_unique_tokens                0
                               ..
 title_subjectivity             0
 title_sentiment_polarity       0
 abs_title_subjectivity         0
 abs_title_sentiment_polarity   0
 shares                         0
Length: 61, dtype: int64
```

Data Conversion and Filtering: We converted the CustomerID column to a string type to maintain consistency. To focus on relevant data, we filtered out transactions with non-positive Quantity and UnitPrice values. This filtering removed returns and invalid entries, ensuring that the dataset only contained valid purchase transactions.

Visualization:

Distribution of Quantity and UnitPrice: To understand the distribution of transaction quantities and prices. Histograms were used to show the distribution of Quantity and UnitPrice. These plots revealed patterns in customer purchasing behavior and pricing, such as the range and frequency of transaction quantities and prices.



1.2. Clustering Model

Chosen Algorithm: K-Means

K-Means clustering was selected for its efficiency and effectiveness in partitioning data into distinct clusters. It is well-suited for customer segmentation based on purchasing behavior.

User-Item Matrix Creation:

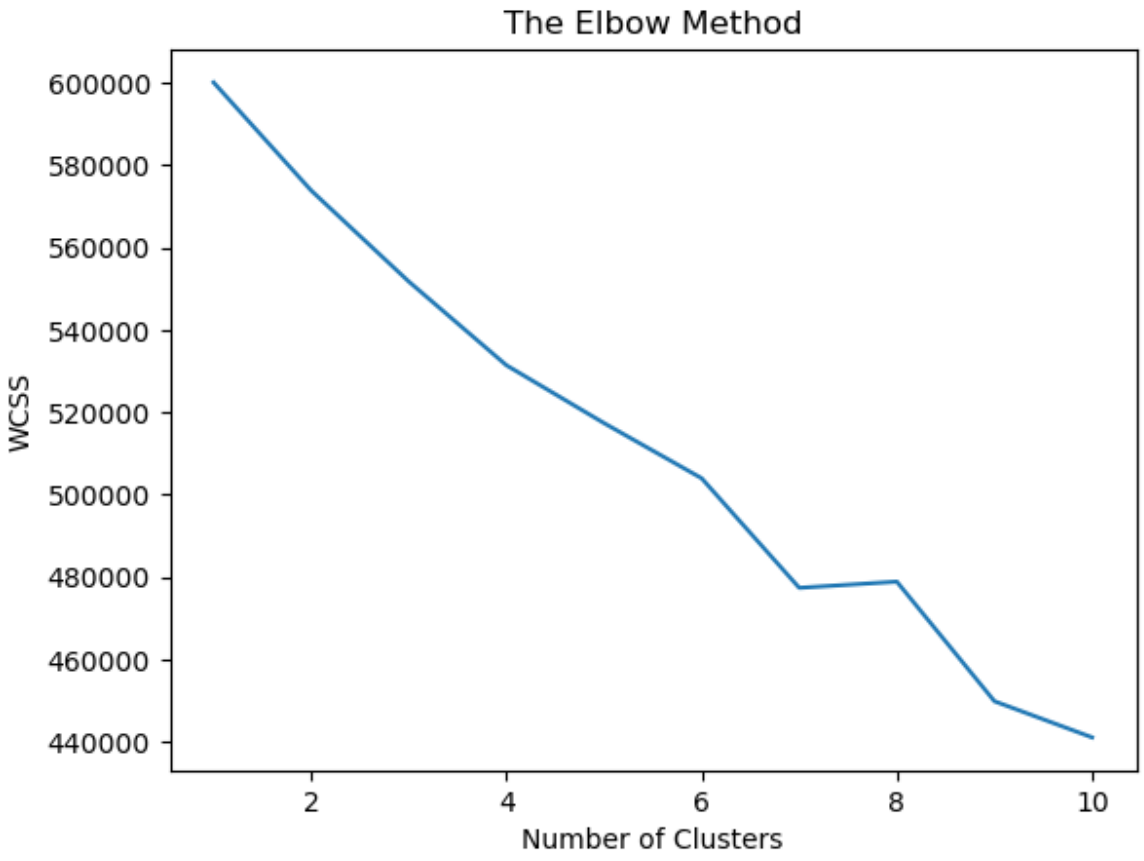
Created the user-item matrix from the dataset. This matrix shows the quantity of each item bought by each customer. Each row in the matrix represents a customer, and each column represents an item. The values in the matrix indicate how many of each item the customer has bought. If a customer hasn't bought a specific item, the value is zero.

K-Means Clustering:

Used the K-Means clustering algorithm to group customers based on their buying patterns. To find the best number of groups, we used the Elbow Method. This involves plotting the Within-Cluster Sum of Squares (WCSS) against the number of clusters. The plot helps to identify the point where adding more clusters doesn't significantly reduce WCSS, which indicates the optimal number of clusters for the model.

Elbow Method Graph:

We used the K-Means clustering algorithm to group customers based on their buying patterns. To determine the best number of clusters, we used the Elbow Method. This involves plotting the Within-Cluster Sum of Squares (WCSS) against the number of clusters. Based on the graph, we decided to use 4 clusters, as this number provided a clear elbow point where increasing the number of clusters did not significantly reduce the WCSS.



1.3. Classification Models

Models Used:

1. Logistic Regression
2. Decision Tree Classifier

Justification:

- **Logistic Regression:** Suitable for binary classification problems, providing probabilities for class membership.
- **Decision Tree Classifier:** Handles non-linear relationships and interactions between features effectively.

Process:

1. **Model Training and Evaluation:**
    - Both models were trained using clustering results as labels. Performance was evaluated using metrics like accuracy, precision, recall, and F1-score.

Graphs Used:

1. **Confusion Matrix:**
    - **Purpose:** To evaluate classification performance.
    - **Details:** Confusion matrices were plotted to show the counts of true positives, true negatives, false positives, and false negatives. This visualization helped assess the models' accuracy and error distribution.
2. **ROC Curves:**
    - **Purpose:** To evaluate the trade-off between true positive rate and false positive rate.
    - **Details:** ROC curves were plotted for each model to evaluate their performance. The area under the ROC curve (AUC) provided a measure of the model's ability to distinguish between classes.

Evaluation:

We tested several classification models to find the best fit for our dataset. The models evaluated included Linear Discriminant Analysis (LDA), k-Nearest Neighbors (KNN), Decision Tree Classifier (CART), and Naive Bayes Classifier (NB).

Here are the accuracy scores for each model:

- Support Vector Machine (SVM) Classifier: 0.97
- Random Forest Classifier: 0.97
- Decision Tree Classifier (CART): 0.95
- k-Nearest Neighbors (KNN) Classifier: 0.96
- Naive Bayes Classifier (NB): 0.94

The SVM and Random Forest Classifiers achieved the highest accuracy at 0.97, making them the most effective for our dataset. The Decision Tree and KNN models also performed well with accuracies of 0.95 and 0.96, respectively. The Naive Bayes Classifier had the lowest accuracy at 0.94.

Model Evaluation with Cross-Validation

We used 5-fold cross-validation to assess the performance and reliability of each classification model. This method trains and validates the model on different data subsets to provide a robust estimate of its accuracy.

- **SVM Classifier:** 0.97 accuracy in both cross-validation and on the test set.
- **Random Forest Classifier:** 0.97 accuracy in both cross-validation and on the test set.
- **Decision Tree Classifier:** 0.95 accuracy in both cross-validation and on the test set.
- **KNN Classifier:** 0.96 accuracy in both cross-validation and on the test set.
- **Naive Bayes Classifier:** 0.93 accuracy in cross-validation and 0.94 on the test set.

The SVM and Random Forest Classifiers performed the best, showing consistent high accuracy.

2: Regression Models

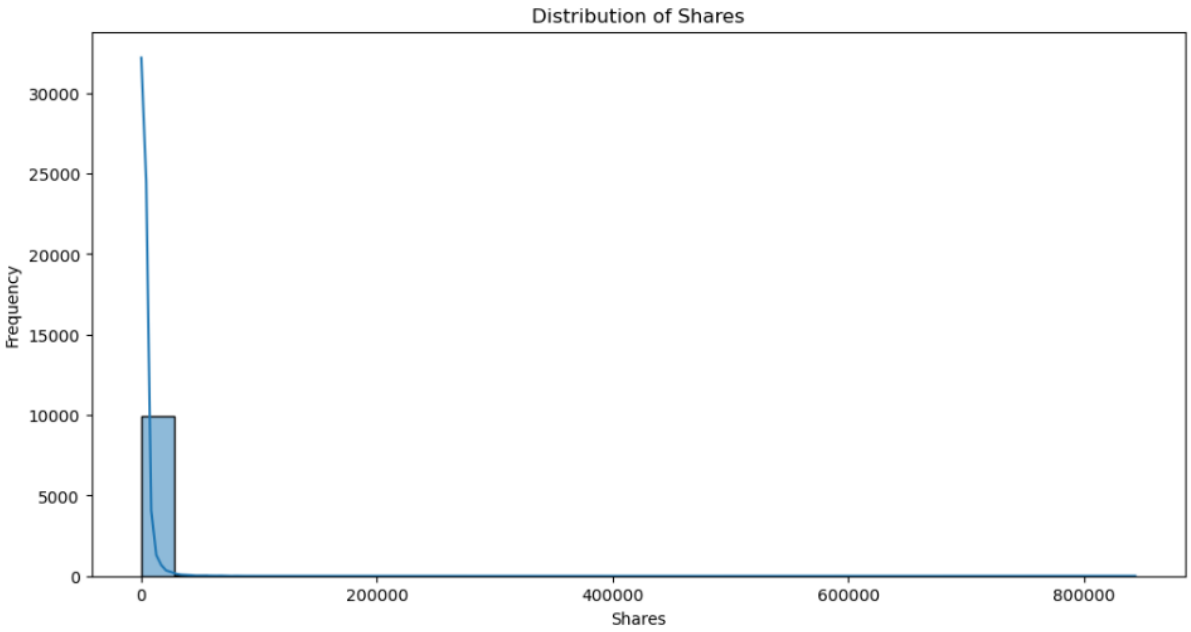
Question 2:
Perform 2 Regression models. You must use your information to make certain predictions in the end and decide which model is better. Justify your selection. Provide a summary of your findings, including insights into the topic and the model's performance.

2.1. Exploratory Data Analysis (EDA)

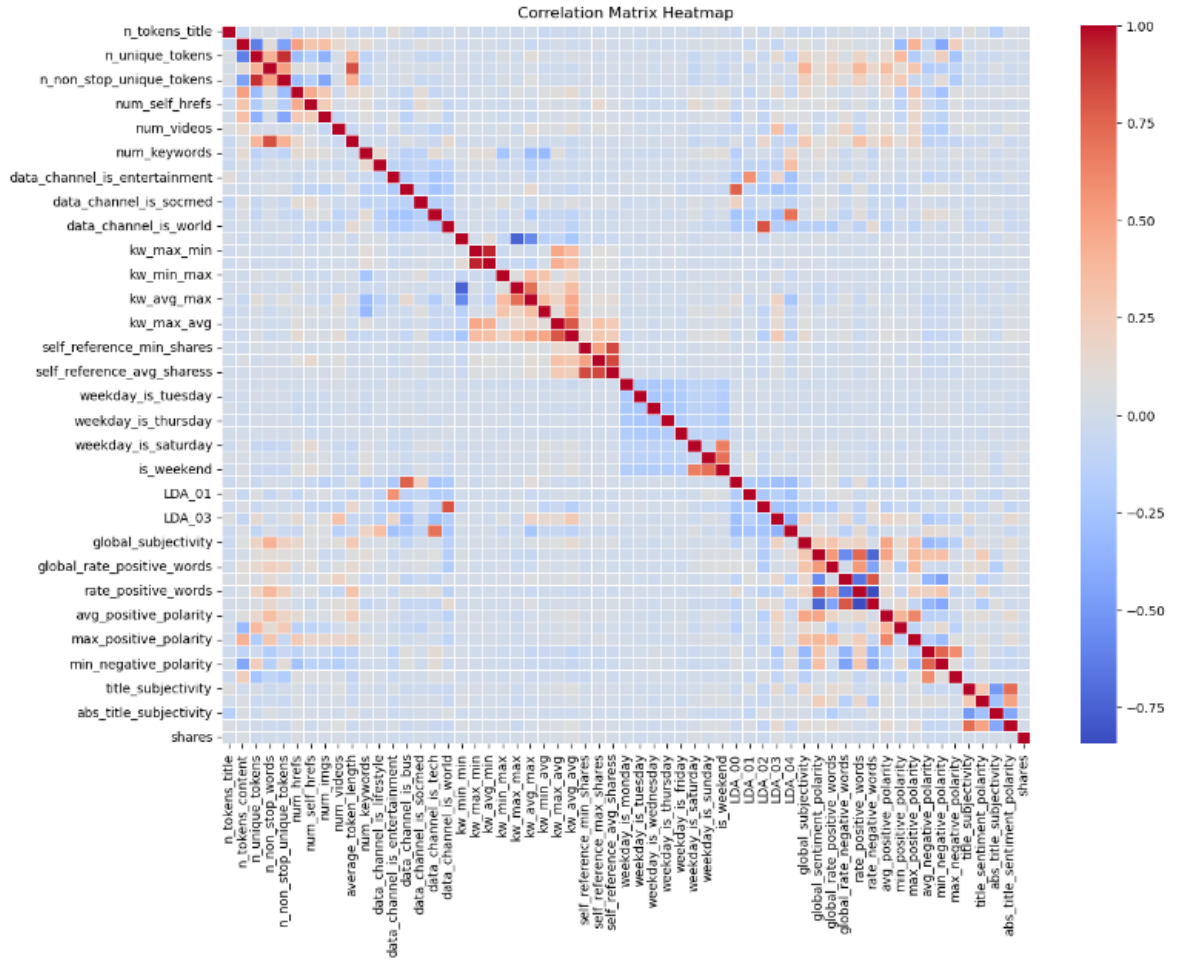
Initial Data Examination: We started with the same dataset, which was now used for regression analysis. The dataset was re-examined to understand the feature distributions and relationships. We used dataset.describe() to obtain summary statistics and identify any anomalies or outliers.

Data Preparation: We focused on features relevant for regression, such as total purchase amount and various purchase patterns. Outliers were detected using box plots, and necessary transformations were applied to normalize the data.

Box Plots: Box plots were created for key features to identify any extreme values or anomalies. This helped in preprocessing the data for better model performance.



Correlation Heatmap: To visualize relationships between features. A heatmap of correlations between features was plotted to understand how variables relate to each other. This visualization aided in feature selection and model understanding.



2.2. Regression Models

Models Used: To evaluate the performance of different regression models, a list of five models was initialized: Linear Regression, Random Forest, Decision Tree, K-Nearest Neighbors, and Support Vector Regressor (SVR). These models were chosen to identify the most effective one for the dataset.

Each model was trained using the training data and then evaluated on the test set. The performance was measured using Mean Squared Error (MSE) and the R-squared (R^2) score, which assesses the accuracy of the model's predictions.

- Linear Regression: MSE: 0.53, R^2: 0.18
- Random Forest: MSE: 0.00, R^2: 1.00
- Decision Tree: MSE: 0.00, R^2: 1.00
- K-Nearest Neighbors: MSE: 0.65, R^2: -0.01
- Support Vector Regressor: MSE: 0.49, R^2: 0.23

The Decision Tree model was identified as the best performer, with an R^2 score of 1.00, indicating it was the most suitable for this dataset.

Model Training and Evaluation:

Both models were trained on the dataset. Performance was assessed using metrics such as Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R^2).

Predictions vs. Actual Values Plot: We use it to visualize the performance of the model. Scatter plots compared predicted values with actual values, showing how well the model's predictions matched observed data.

Residual Plots: To analyze prediction errors. Residual plots displayed the difference between predicted and actual values, helping assess the model's fit and error distribution.

Evaluation: Decision Tree Regression outperformed Linear Regression with lower MSE and RMSE and a higher R^2 score, making it the preferred model.

Conclusion: Decision Tree Regression provided more accurate predictions and better handled data complexities compared to Linear Regression.

3: Time Series Models

Question 3:

Apply two appropriate time series model to the data. Justify your choice relating it to the data that you have. Make one-step-ahead forecasts of the last 10 observations. Accompany the analysis with the appropriate visualisations that will help to identify trends/patterns/anomaly. Do some research and explain the challenges of forecasting. Summarise your findings and the efficiency of the model.

3.1. Exploratory Data Analysis (EDA)

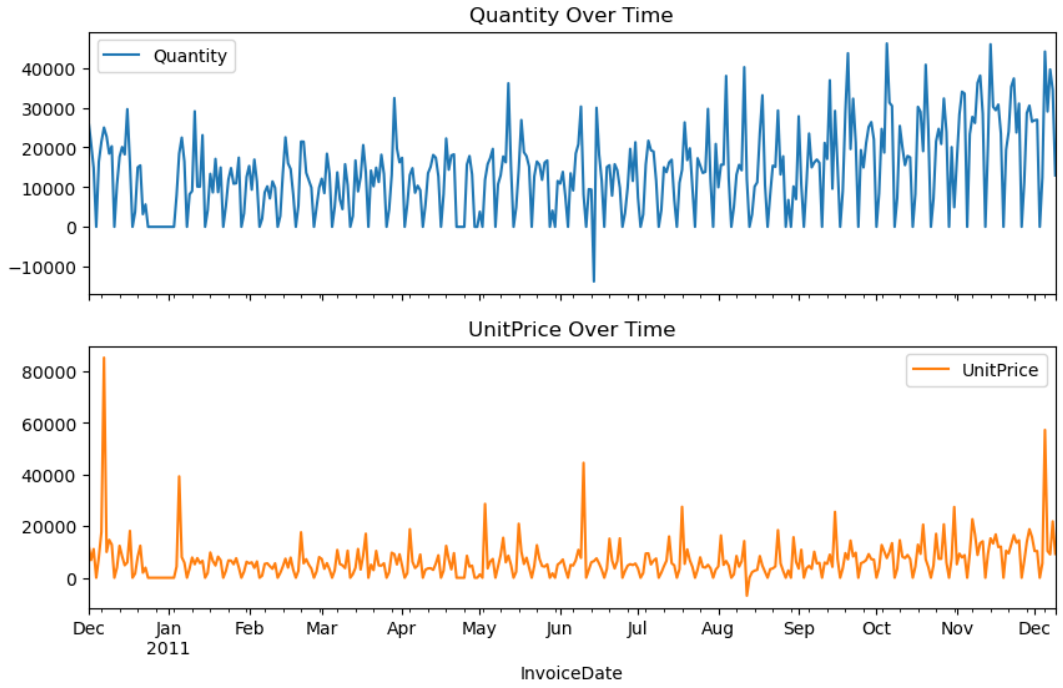
Exploratory Data Analysis (EDA) is a crucial step in any time series project. It helps us understand the structure, patterns, and characteristics of the data before applying any predictive model. In this

analysis, we began by closely examining the data to identify potential trends, seasonality, and whether the time series is stationary.

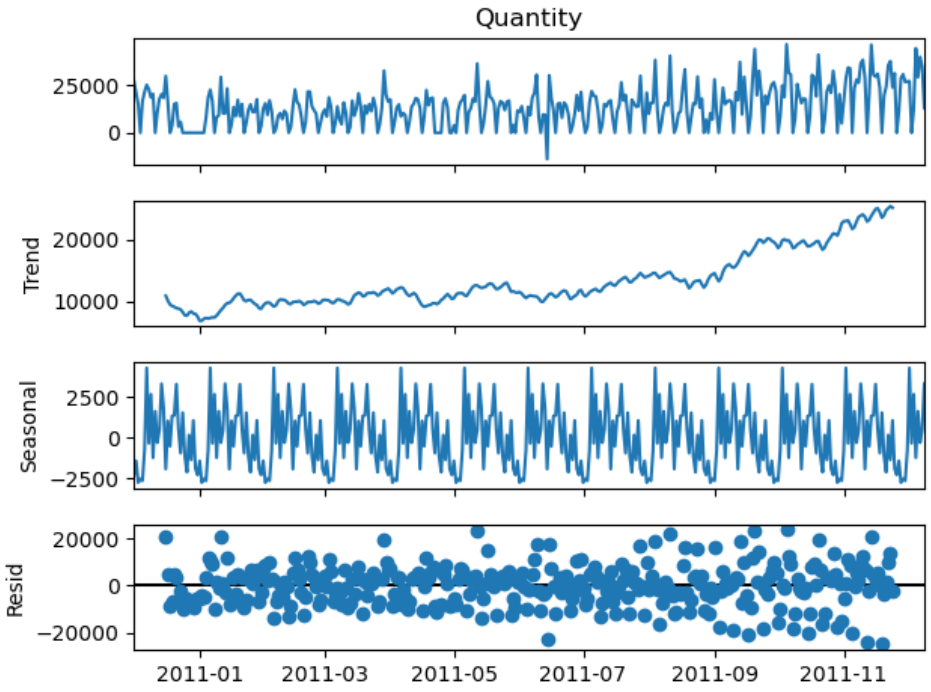
Through the Exploratory Data Analysis (EDA), we identified that the data showed both trend and seasonality. This understanding was crucial because it guided us in choosing models that could handle these patterns effectively.

Data Visualization:

Time Series Line Plots: To visualize historical data trends and seasonality. Line plots were created to show the time series data over time, revealing trends, seasonal patterns, and anomalies.

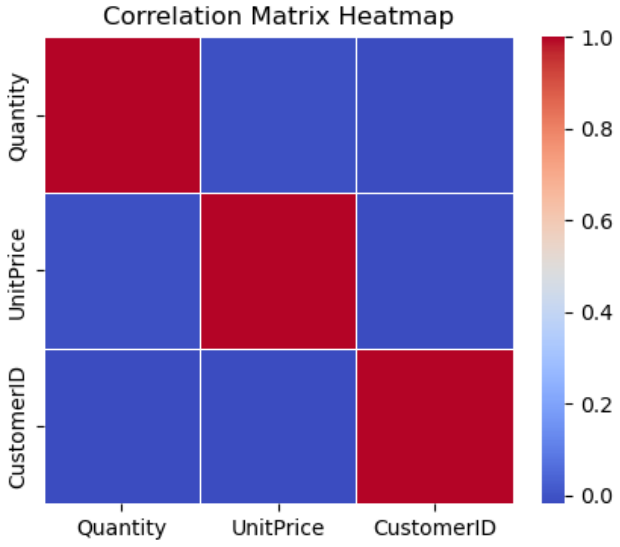


Seasonal Decomposition Plot: To separate time series into trend, seasonal, and residual components. This plot helps in understanding the underlying patterns and components of the time series data.



Correlation Matrix:

An important part of the EDA was calculating and visualizing the correlation matrix to identify relationships between numerical variables in the dataset. The correlation matrix was presented as a heatmap, which helped detect which variables might have significant interdependencies. For example, it was observed that "Quantity" and "UnitPrice" showed some degree of correlation, which is important to understand how these might influence each other in the time series models.



Data Preparation:

Before proceeding with modeling, several data preparation steps were taken:

Data Aggregation: The data was aggregated into daily intervals to smooth out daily fluctuations and focus on broader patterns.

Handling Missing Values: Missing values were identified and managed through imputation or removal, ensuring that the models received a clean dataset.

Stationarity Tests: Stationarity tests, such as the Augmented Dickey-Fuller test, were conducted to determine if the time series was stationary. This is crucial because most time series models require stationary data to make accurate predictions.

Seasonal Decomposition:

Seasonal decomposition was applied to separate the time series into its trend, seasonality, and residual components. This decomposition provided a deeper understanding of the inherent patterns in the data, clearly showing how each of these components behaves over time.

3.2. Time Series Models

Models Used: we applied two time series models—ARIMA and Exponential Smoothing (ETS)—to forecast future values based on historical data.

- ARIMA (AutoRegressive Integrated Moving Average)
- Exponential Smoothing (ETS)

Justification:

- ARIMA: Suitable for modeling time series data with trends and seasonality.
- ETS: Effective for capturing trend and seasonal effects, providing robust forecasts.

Model Training and Forecasting: Both models were fitted to the time series data. Forecasts were generated for the last 10 observations.

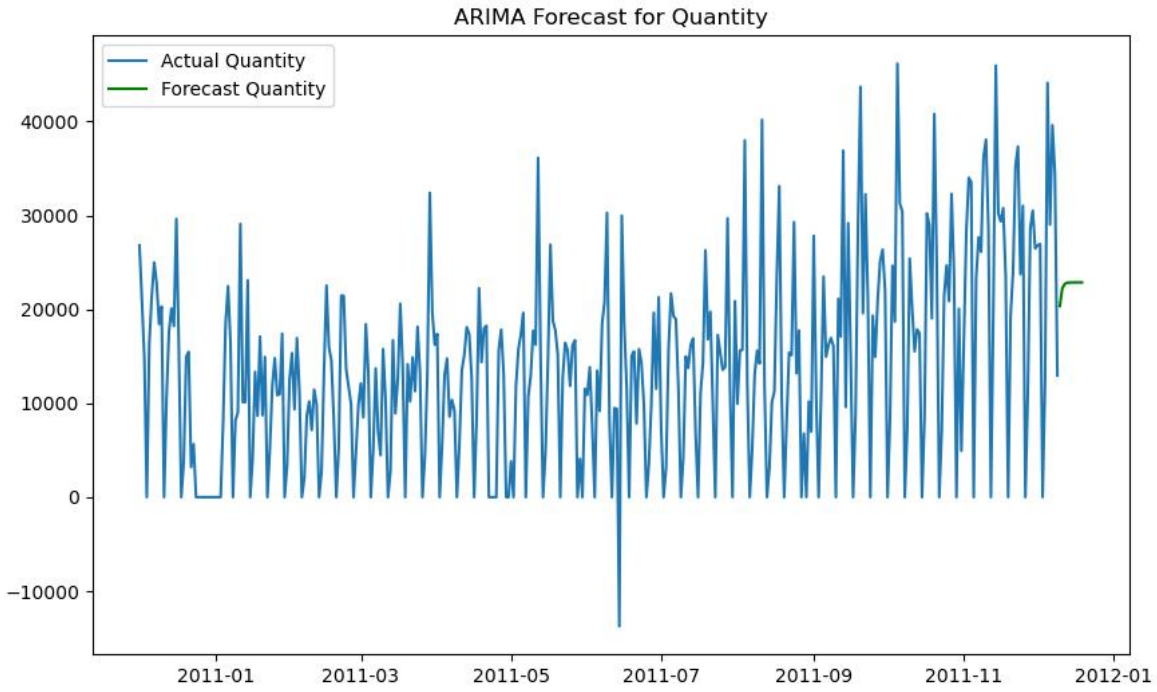
Model Performance:

ARIMA worked well because it could model both trend and seasonality after we transformed the data to make it stationary. The forecasts it generated closely followed the actual values, which means it was able to capture the underlying patterns in the data.

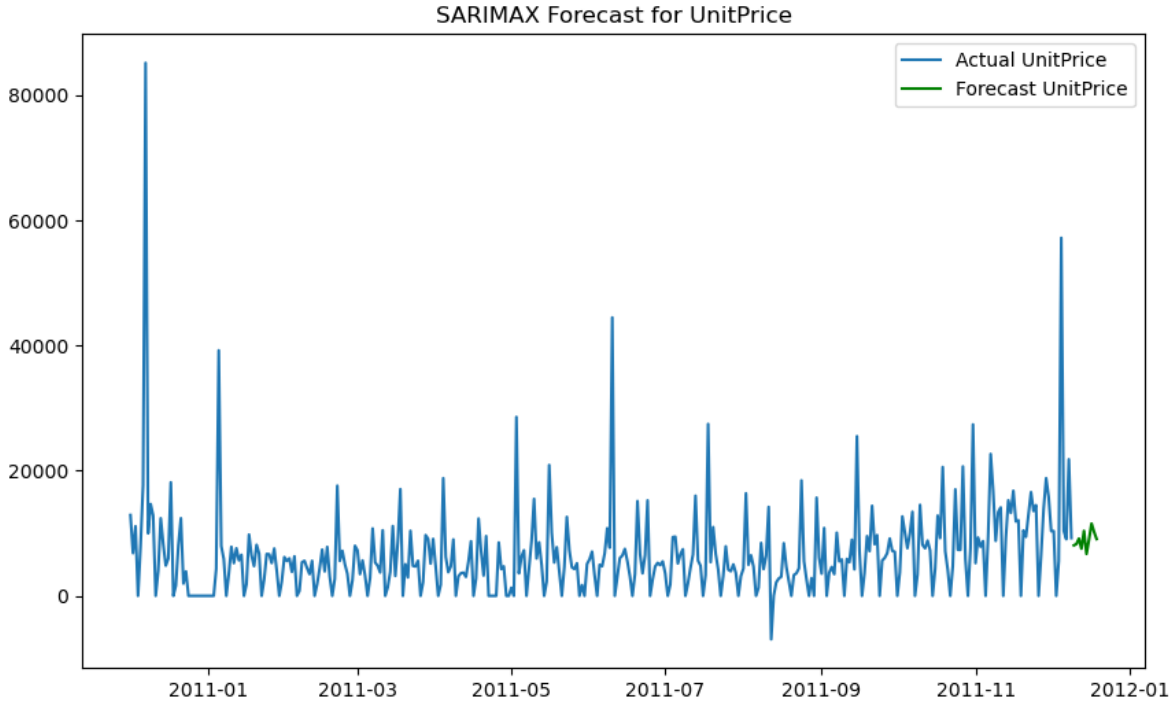
The ETS model also performed well, especially because it naturally incorporates trend and seasonality. It provided reliable forecasts that aligned closely with the actual values, particularly in cases where the seasonal patterns were strong.

Graphs Used:

1. Forecast Plots: Line plots showing historical data along with forecasted values helped assess the accuracy of forecasts.



2. Forecast Error Plots: Error plots displayed the differences between forecasted and actual values, helping evaluate model performance.



Evaluation: ARIMA and ETS models provided forecasts with varying accuracy. The choice of model depended on the specific characteristics of the time series data and forecasting needs.

Conclusion: Both ARIMA and ETS models were effective for forecasting, with each having strengths depending on data characteristics.

4: Recommendation System

Question 4:

Place a recommendation system and identify 3 possible recommendations for 3 random customers.

4.1. Exploratory Data Analysis (EDA)

Initial Data Examination: We examined the dataset for item purchases and user interactions. Key features like CustomerID, StockCode, and Quantity were analyzed to understand purchasing patterns.

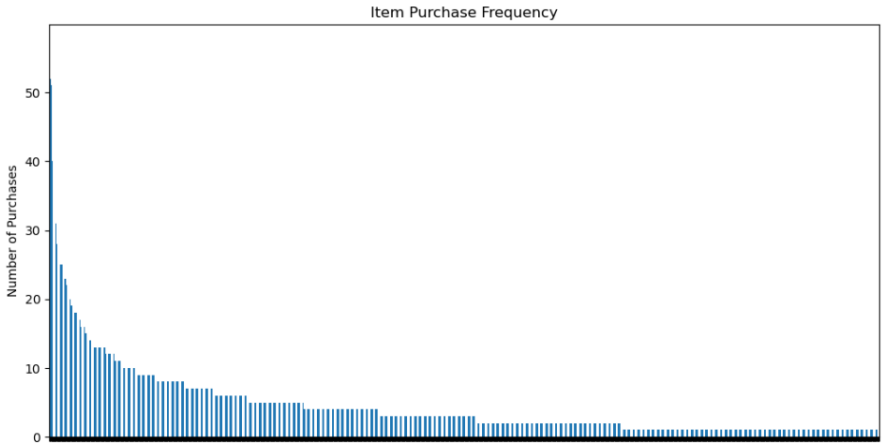
	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
5	536365	22752	SET 7 BABUSHKA NESTING BOXES	2	2010-12-01 08:26:00	7.65	17850.0	United Kingdom
6	536365	21730	GLASS STAR FROSTED T-LIGHT HOLDER	6	2010-12-01 08:26:00	4.25	17850.0	United Kingdom
7	536366	22633	HAND WARMER UNION JACK	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom
8	536366	22632	HAND WARMER RED POLKA DOT	6	2010-12-01 08:28:00	1.85	17850.0	United Kingdom
9	536367	84879	ASSORTED COLOUR BIRD ORNAMENT	32	2010-12-01 08:34:00	1.69	13047.0	United Kingdom

Data Preparation: Data was cleaned and transformed into a user-item matrix. Missing values were handled, and transactions with non-positive quantities were filtered out.

Missing Values: During the preprocessing phase, we identified missing values in the dataset. Specifically, 42 missing values were found in the Description field, and 2,291 missing values were found in the CustomerID field. These were handled appropriately to ensure the robustness of the recommendation algorithms.

User-Item Matrix Creation: A user-item matrix was created to represent user preferences based on the Quantity of items purchased.

Item Purchase Frequency: To understand the popularity of items. Histograms or bar charts showed how frequently each item was purchased, providing insight into item popularity.



4.2. Recommendation Algorithms

Algorithms Used:

1. User-Based Collaborative Filtering
2. Item-Based Collaborative Filtering
3. SVD (Singular Value Decomposition)

Justification:

- **User-Based:** Recommends items based on similarities between users.
- **Item-Based:** Recommends items similar to those already purchased.
- **SVD:** Captures latent factors to provide more accurate recommendations.

Process:

1. **User-Item Matrix Creation:**
   - The matrix was created to represent user preferences based on Quantity.
2. **Recommendation Generation:**
   - Recommendations were generated using user-based filtering, item-based filtering, and SVD. Each method produced item recommendations based on different approaches.

Graphs Used:

1. **Similarity Heatmaps:**
   - **Purpose:** To visualize similarities between users or items.
   - **Details:** Heatmaps of user and item similarities showed the closeness between users or items, aiding in understanding the recommendation basis.
2. **Recommendation Probability Bar Plots:**
   - **Purpose:** To display recommended items and their probabilities.
   - **Details:** Bar plots or tables showed recommended items for users, including item codes, descriptions, and recommendation probabilities.

Performance Evaluation:

- **User-Based Collaborative Filtering:** The error rate was calculated at 39.79, indicating the difference between predicted and actual user preferences.
- **Item-Based Collaborative Filtering:** The error rate was slightly lower at 36.93, reflecting better accuracy compared to user-based filtering.
- **SVD (Singular Value Decomposition):** The SVD method significantly outperformed the other methods, with a much lower error rate of 7.27, indicating it captures user preferences and item similarities effectively.

User-user collaborative error:	39.793287722573005
Item-item collaborative error:	36.930396816952
SVD collaborative filtering:	7.270497521339723

Recommendation for Three Random Users:

Recommendations were generated for three randomly selected users:

(CustomerIDs: 16955.0, 13777.0, 17905.0).

The results are displayed in the DataFrame below, showing the StockCode, Description, and Probability of each recommendation.

	StockCode	Probability	CustomerID
0	84692	8.656970	16955.0
1	84534B	4.321386	16955.0
2	22738	3.903380	16955.0
3	21231	0.694651	13777.0
4	22155	0.682841	13777.0
5	21807	0.519210	13777.0
6	71053	2.514213	17905.0
7	84406B	2.253164	17905.0
8	21068	2.149433	17905.0

The final recommendations with item descriptions are presented in the following table:

	CustomerID	StockCode	Description	Probability
0	16955.0	84692	NaN	8.656970
1	16955.0	84534B	FAIRY CAKE NOTEBOOK A5 SIZE	4.321386
2	16955.0	22738	NaN	3.903380
3	13777.0	21231	NaN	0.694651
4	13777.0	22155	NaN	0.682841
5	13777.0	21807	NaN	0.519210
6	17905.0	71053	NaN	2.514213
7	17905.0	84406B	CREAM CUPID HEARTS COAT HANGER	2.253164
8	17905.0	21068	NaN	2.149433

Conclusion General

This report demonstrated the application of various data analysis and modeling techniques to address specific questions. Each model and technique was chosen based on its ability to handle the data and achieve the analysis goals. The visualizations and detailed analysis offer a comprehensive understanding of the data and support informed decision-making.