Data Analysis Report: Online Retail Dataset

---

1. Introduction

This report aims to explore and analyze the Online Retail dataset, focusing on key aspects such as basic statistics, distributions of quantities and prices, relationships between attributes, and dimensionality reduction techniques to uncover underlying patterns. The data was initially explored, and various statistical analyses were performed to understand the data structure, identify outliers, and visualize trends.

---

2. Dataset Overview

The dataset comprises transactions from an online retail store, with the following primary attributes:

- **InvoiceNo**: Unique identifier for each transaction.
- **StockCode**: Product identifier.
- **Description**: Product description.
- **Quantity**: Number of items purchased in each transaction.
- **UnitPrice**: Price per unit of the product.
- **CustomerID**: Unique identifier for each customer.
- **Country**: Country where the customer is based.

---

3. Exploratory Data Analysis (EDA)

3.1. Initial Exploration

After loading the dataset, we examined the first few records to understand its structure:

```
print(data.head())
```

The dataset contains important attributes like `Quantity`, `UnitPrice`, and `InvoiceNo`, which are crucial for analyzing purchasing behavior.

3.2. Basic Statistics

Summary statistics for key columns, such as `Quantity` and `UnitPrice`, were generated using the `describe()` method. This gives us a high-level view of the data's central tendency, spread, and potential anomalies:

```
summary_stats = data.describe()
```

From the summary statistics, we identified that the dataset has a wide range of `Quantity` and `UnitPrice`, suggesting the presence of both high-volume and high-priced transactions.

3.3. Median Calculation

The median `UnitPrice` was computed, which provides the middle value in the price distribution:

```
median = data['UnitPrice'].median()
```

The median helps in understanding the typical price point for products, reducing the influence of extreme outliers.

---

4. Distribution Analysis

4.1. Quantity Distribution

A histogram was plotted to observe the distribution of the `Quantity` attribute. We sampled 1000 records to get a better understanding of the range of quantities, limiting the x-axis to a maximum of 500 for better visualization:

```
plt.hist(random_sample['Quantity'], bins=1000, color='blue',
edgecolor='black')
```

This histogram shows a skewed distribution with a larger frequency of smaller quantities purchased.

4.2. UnitPrice Distribution

Similarly, the `UnitPrice` distribution was plotted. The histogram of prices was limited to values between 0 and 100 for a clearer view of the price range:

```
plt.hist(random_sample['UnitPrice'], bins=100, color='blue')
```

The distribution appears right-skewed, indicating that most products are priced relatively low, with a few higher-priced items.

---

5. Outlier Detection

5.1. Boxplot Analysis

A boxplot of the `Quantity` attribute was created to visually identify potential outliers:

```
sns.boxplot(x=data['Quantity'])
```

Outliers are marked by dots outside the whiskers, and these extreme values may be due to errors or rare high-volume transactions that need further investigation.

5.2. Scatter Plot Analysis

Scatter plots were used to investigate the relationship between `UnitPrice` and `Quantity`. We first visualized the entire dataset and then zoomed into specific price ranges:

```
sns.scatterplot(x=data['UnitPrice'], y=data['Quantity'])
```

From these plots, we observed that as the `UnitPrice` increases, the `Quantity` tends to decrease. A few extreme points were visible in the scatter plot, which are potential outliers.

---

6. Statistical Measures

6.1. Mean and Centered Data

The mean of the `Quantity` column was calculated, and the data was centered around this mean for further analysis:

```
mean = data['Quantity'].mean()
centered_data = data['Quantity'] - mean
```

This centered data helps in understanding how each transaction deviates from the average transaction.

6.2. Variance Calculation

The **total variance** of the dataset was computed to quantify the spread of the data. Variance gives insight into the variability of transactions across customers:

```
total_variance = np.sum(centered_data ** 2) / (data_matrix.shape[0] *
data_matrix.shape[1])
```

The total variance is important for understanding the overall spread of the data and its potential impact on analysis.

---

7. Covariance Matrices

Covariance matrices were computed using two methods to analyze the relationships between variables:

Inner Product Method:

```
inner_product_covariance = np.dot(centered_data_matrix.T,
centered_data_matrix) / (centered_data_matrix.shape[0] - 1)
```

-

Outer Product Method:

```
outer_product_covariance = np.cov(centered_data_matrix, rowvar=False)
```

These covariance matrices help in understanding how different attributes, such as `Quantity` and `UnitPrice`, vary together.

---

8. Dimensionality Reduction: PCA and t-SNE

To explore high-dimensional relationships, we applied **Principal Component Analysis (PCA)** and **t-SNE** (t-distributed Stochastic Neighbor Embedding) techniques.

8.1. PCA Visualization

PCA was applied to reduce the dataset to 2 dimensions, which were then visualized using a scatter plot:

```
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X)
```

This plot helps to identify whether there are any clear clusters or patterns in the data based on the principal components.

8.2. t-SNE Visualization

t-SNE was applied to preserve local relationships while reducing the data to 2 dimensions:

```
tsne = TSNE(n_components=2, perplexity=30)
X_tsne = tsne.fit_transform(X)
```

t-SNE visualization helped to reveal local groupings based on `Quantity` and `UnitPrice`, providing further insight into the dataset's structure.

---

9. Conclusion

- **Outliers and Anomalies**: Outliers were identified in both `Quantity` and `UnitPrice`, which might represent erroneous data or rare purchase behaviors. These should be examined further to determine their impact.
- **Relationships Between Variables**: There is an inverse relationship between `UnitPrice` and `Quantity` — as the price of an item increases, the quantity purchased decreases.
- **Dimensionality Reduction**: PCA and t-SNE helped visualize the dataset in 2D and uncover underlying patterns. PCA focused on the variance in the data, while t-SNE revealed local relationships and potential groupings.
- **Statistical Analysis**: Mean, variance, and covariance calculations provided insights into the data's central tendency and spread.

Further data cleaning is recommended to address outliers and potential errors, which would improve the reliability of any predictive modeling or further analysis.

---

10. Recommendations for Further Analysis

- **Outlier Treatment**: Investigate extreme outliers and assess whether they should be removed or treated.
- **Segmentation**: Segment the data by customer or product categories to uncover more specific trends.
- **Predictive Modeling**: With cleaned data, predictive models could be built to forecast sales, customer behavior, or inventory needs.

This report provides a foundational understanding of the dataset, which can be further expanded with deeper analysis or modeling techniques.