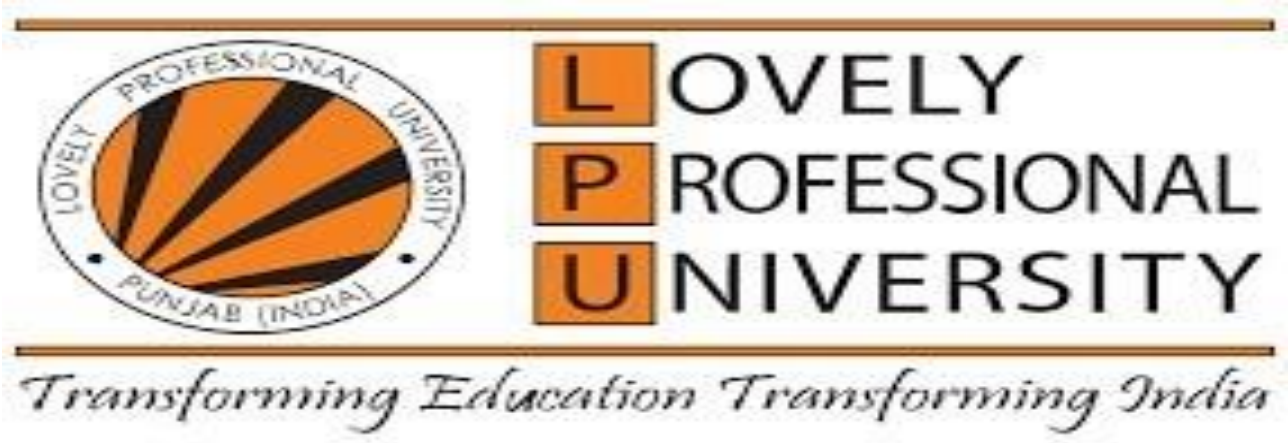


Supermarket Sales Analysis

EDA Project

Submitted by:

Dharani K.S ,

B. Tech Computer Science and Engineering with specialization inData Science [AI & ML],

Lovely Professional University

Punjab.

TABLE OF CONTENT

Introduction:

The exploration and analysis of the provided supermarket sales dataset are crucial for gaining valuable insights into the business operations, customer behavior, and overall performance of the supermarket. Exploratory Data Analysis (EDA) serves as a fundamental step in understanding the underlying patterns, trends, and anomalies within the data. The project aims to uncover meaningful information that can be utilized for strategic decision-making and process optimization in the supermarket industry.

Purposes:

- To know the distribution of sales data and supermarket market segmentation.
- To know supermarket customer satisfaction based on categorical and numerical indicators.
- To know how is the distribution of sales data and supermarket market segmentation.
- To know how is supermarket customer satisfaction based on categorical and numerical indicators.

Domain Knowledge:

Super Market – Retail Domain:

The retail industry involves the sale of goods and services to consumers, typically through physical stores or online platforms. Supermarkets are a subset of the retail industry, specializing in the sale of a wide range of consumer products, including groceries, household items, and more. A retail superstore is a large-scale retail establishment that offers a diverse range of consumer products, often encompassing multiple categories, all conveniently located under one roof. These sprawling retail spaces are designed to provide customers with a one-stop shopping experience, where they can find a wide variety of items, from groceries and electronics to clothing, household goods, furniture, and more.

Industry Relevance:

The retail industry is a crucial component of the economy, serving as a direct link between manufacturers and consumers. Supermarkets, in particular, play a pivotal role in providing everyday essentials to the public. Understanding consumer behavior, managing inventory efficiently, and optimizing pricing strategies are essential for success in this industry.

Important Things to Know:

- Consumer segmentation and targeted marketing.
- Promotional strategies and campaigns.
- Customer relationship management (CRM).
- Branding and positioning in the market.
- Inventory management and demand forecasting.
- Supply chain logistics and distribution.
- Supplier relationships and negotiations.
- Operational efficiency and process optimization.
- Cost of Goods Sold (COGS) and gross margin analysis.
- Tax implications and financial compliance.
- Budgeting and financial forecasting.
- Payment methods and transaction analysis

Key Features:

Some of key features to know for performing analysis are,

•**Technology Integration:** Modern superstores utilize advanced point-of-sale (POS) systems, inventory management software, and e-commerce platforms to streamline operations and enhance the shopping experience.

•**Customer Loyalty Programs:** To foster customer loyalty, superstores often

implement loyalty programs, offering discounts, rewards, or exclusive offers to repeat customers.

•**Marketing and Advertising**: Superstores invest in marketing campaigns to promote their brand and attract shoppers. They may use a combination of digital marketing, traditional advertising, and in-store promotions to reach their audience.

•**Diverse Product Range:** Superstores stock a broad selection of merchandise, often spanning groceries, electronics, home and garden items, clothing, automotive supplies, sporting goods, and more. The variety of products caters to a wide demographic of customers.

Reason for choosing dataset:

Name of the dataset : supermarket_sales.csv

About Dataset:

The growth of supermarkets in most populated cities are increasing and market competitions are also high. The dataset is one of the historical sales of supermarket company which has recorded in 3 different branches for 3 months data from Jan 2019 to Mar 2019.

This dataset is a collection of data from a supermarket company in Myanmar, which has three branches in Yangon, Mandalay, and Naypyitaw. We want to evaluate customer satisfaction from several factors by doing exploratory data analysis to see some of the relationships of the available variables on their supermarket sales dataset.

This supermarket dataset is a collection of data that provides information about the transactions that took place in a supermarket. This dataset typically includes information such as the date and time of the transaction, the products that were purchased, the price of each product, the total amount spent on the transaction, and other relevant details.

This dataset is used to perform data analysis and gain insights into the behavior of the supermarket customers. For example, we might use this supermarket dataset to study the relationship between the day of the week and the total amount spent on transactions, or to analyze the factors that influence customer spending patterns. We might also use this dataset to identify trends and patterns in customer behavior, to optimize pricing and product placement, or to develop marketing and advertising strategies.

Overall, this supermarket dataset can be a valuable tool for understanding and predicting the behavior of supermarket customers, and for making data-driven decisions that can improve the performance of a supermarket business.

Data Description:

The dataset is one of the historical sales of supermarket company which has recorded in 3 different branches for 3 months data from Jan 2019 to Mar 2019. This dataset is a collection of data from a supermarket company in Myanmar, which has three branches in Yangon, Mandalay, and Naypyitaw. We want to evaluate customer satisfaction from several factors by doing exploratory data analysis to see some of the relationships of the available variables on their supermarket sales dataset. It has 1000 rows and 17 columns.

Attributes Information :

- Invoice id: Computer generated sales slip invoice identification number
- Branch: Branch of supercenter (3 branches are available identified by A, B and C).
- City: Location of supercenters
- Customer type: Type of customers, recorded by Members for customers using member card and Normal for without member card.

- Gender: Gender type of customer
- Product line: General item categorization groups - Electronic accessories, Fashion accessories, Food and beverages, Health and beauty, Home and lifestyle, Sports and travel
- Unit price: Price of each product in $
- Quantity: Number of products purchased by customer
- Tax: 5% tax fee for customer buying
- Total: Total price including tax
- Date: Date of purchase (Record available from January 2019 to March 2019)
- Time: Purchase time (10am to 9pm)
- Payment: Payment used by customer for purchase (3 methods are available – Cash, Credit card and Ewallet)
- COGS: Cost of goods sold
- Gross margin percentage: Gross margin percentage
- Gross income: Gross income
- Rating: Customer stratification rating on their overall shopping experience (On a scale of 1 to 10)

Data Type :

- Invoice ID object
- Branch object
- City object
- Customer type object
- Gender object
- Product line object

- Unit price float64
- Quantity int64
- Tax 5% float64
- Total float64
- Date object
- Time object
- Payment object
- cogs float64
- gross margin percentage float64
- gross income float64
- Rating float64

Work Flow and Steps of EDA :

- The workflow of this EDA project is,
- Importing the required libraries
- Reading the data
- About the Dataset
- Data preprocessing
- Data Cleaning
- Exploring the Dataset
- Univariate analysis
- Bivariate analysis
- Multivariate analysis
- Distributions
- Hypothesis Testing
- Conclusion

Importing the required libraries:

This step involves importing programming libraries such as Pandas, NumPy, Matplotlib, Seaborn, or other relevant libraries. These libraries provide tools and functions for data manipulation, analysis, and visualization.

Reading the data:

Load the dataset into your programming environment using appropriate functions from the chosen libraries. Common file formats include CSV, Excel, or SQL databases.

About the Dataset:

Understand the basic information about the dataset, such as the number of rows and columns, data types, and a brief overview of the variables. This step helps in getting a preliminary understanding of what the dataset contains.

Data Preprocessing:

Handle missing values, outliers, and any inconsistencies in the data. This step ensures that the dataset is ready for analysis and model building.

Data Cleaning:

Clean the data by addressing issues like duplicates, irrelevant columns, or any other anomalies that might affect the analysis.

Exploring the Dataset:

Conduct a preliminary exploration to identify patterns, trends, and general insights. This step helps in formulating hypotheses for more in-depth analysis.

Univariate Analysis:

Analyze individual variables in the dataset. This includes summarizing statistics, visualizing distributions, and understanding the central tendency and spread of each variable.

Bivariate Analysis:

Explore relationships between pairs of variables. This step helps in understanding how variables interact with each other and if there are any patterns or correlations.

Multivariate Analysis:

Extend the analysis to more than two variables simultaneously. This step provides a more comprehensive view of the relationships within the dataset.

Distributions:

Examine the distributions of variables, especially the target variable if it's a predictive modeling task. This step aids in selecting appropriate models and understanding the nature of the data.

Hypothesis Testing:

If relevant, perform statistical hypothesis tests to validate or reject assumptions and claims about the data. Common tests include t-tests, chi-square tests, or ANOVA.

Conclusion:

Summarize the key findings from the analysis. This can include insights gained, patterns identified, and any recommendations for further analysis or actions based on the results.

Libraries used:

I have shared the libraries used and approaches below here:

seaborn:

Seaborn is a statistical data visualization library based on Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Seaborn comes with several built-in themes and color palettes to make it easy to create aesthetically pleasing visualizations.

matplotlib.pyplot:

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. pyplot is a collection of functions that provide a convenient MATLAB-like interface for creating a variety of plots.

scipy.stats.boxcox:

The boxcox function in the SciPy library is used for performing Box-Cox

transformation. Box-Cox transformation is applied to stabilize the variance and make the data more closely approximate a normal distribution.

scipy.stats.normaltest:

The normaltest function in SciPy tests whether a sample differs from a normal distribution. It combines skewness and kurtosis to produce an omnibus test of normality.

numpy:

NumPy is a fundamental package for scientific computing with Python. It provides support for large, multi-dimensional arrays and matrices, along with mathematical functions to operate on these arrays.

scipy.stats:

The stats module in SciPy contains a wide range of statistical functions and tests. It includes functions for hypothesis testing, probability distributions, and statistical analysis.

pandas:

Pandas is a powerful data manipulation and analysis library. It provides data structures like Series and DataFrame, which are essential for cleaning, transforming, and analyzing structured data.

plotly.express:

Plotly Express is a high-level interface for creating a variety of interactive plots. It is built on top of Plotly, making it easy to create interactive visualizations with just a few lines of code.

Data Preprocessing:

- The initial step involved dropping the "Invoice ID" column as it was deemed irrelevant for the analysis. Subsequently, the data types of the "Date" and "Time" columns were converted to datetime for proper temporal analysis.

- To extract more valuable information from the temporal data, additional features such as "year," "Days," "month," "weekday," and "hour" were derived from the "Date" and "Time" columns. This facilitates a more granular exploration of sales patterns over different time periods.
- The dataset's "Branch" and "City" columns were found to be identical, leading to the decision to drop one of them to avoid redundancy. The "Branch" column was removed, and numerical columns were separated from non-numerical ones to facilitate targeted analysis.
- Furthermore, the dataset was divided into numerical and non-numerical columns to enable specific analyses based on data types. This demarcation streamlines the application of statistical measures and visualization techniques, enhancing the efficiency of the exploratory data analysis process.
- Overall, the data preprocessing steps conducted in this project lay the groundwork for a comprehensive exploratory analysis of supermarket sales. The transformations and feature engineering performed contribute to a more insightful understanding of customer behavior, sales trends, and other critical aspects.

Data Cleaning:

- Handling Missing Data: Checked for missing values using df.isnull().sum(), revealing no missing data in any column.Visualized the absence of missing values with a heatmap using sns.heatmap(df.isnull()).

- Checking for Duplicates:Utilized df.duplicated() to identify duplicate rows. Confirmed the absence of duplicate rows with print(df[duplicates]).

- Handling Outliers:Created a copy of the dataframe (df_copy) to preserve the original data.Employed a loop to iteratively filter out outliers for each

numerical column, keeping values below the 98th percentile.Displayed the resulting dataframe (df_copy) after outlier removal.

- Visualizing Numerical Data Before and After Outlier Removal: Plotted boxplots for each numerical column before outlier removal using Plotly Express (px.box).Repeated the boxplot visualization for numerical columns in the cleaned dataframe (df_copy) to showcase the impact of outlier removal.

- Checking Correlation: Generated a correlation matrix heatmap using Seaborn (sns.heatmap(df.corr(), annot=True)) to explore relationships between numerical variables.

- Descriptive Statistics: Presented descriptive statistics for numerical columns using df.describe().T, providing key metrics such as mean, standard deviation, minimum, maximum.

Data Exploration :

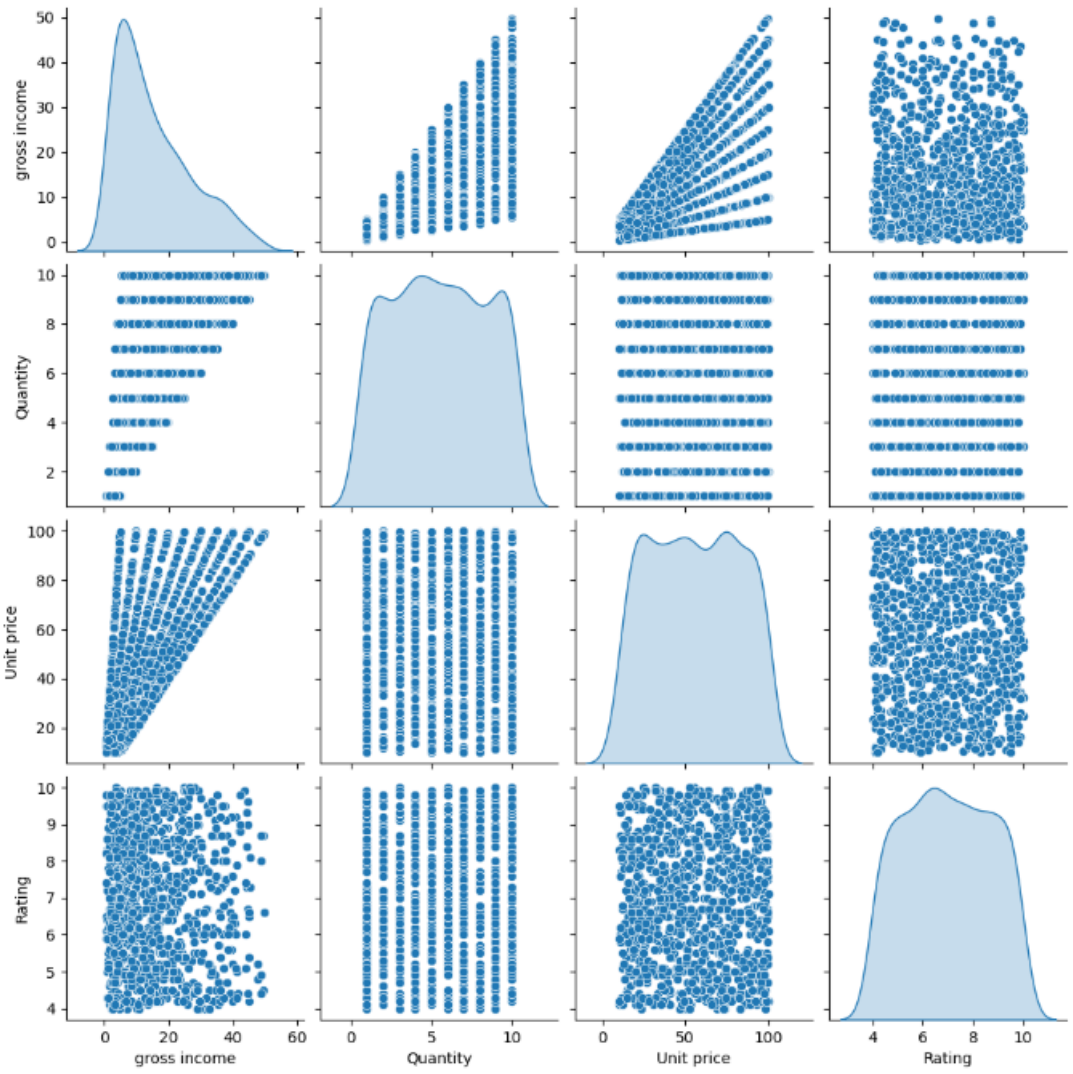
In the initial stages of data exploration, various visualizations were employed to uncover trends and patterns within the supermarket sales dataset. A pair plot was generated for selected columns, including "gross income," "Quantity," "Unit price," and "Rating," offering a comprehensive view of the relationships between these numerical variables. Histograms for key columns, such as "gross income," provided insights into their distributions, helping to identify potential skewness or central tendencies.

Further exploration delved into categorical variables, beginning with a count of payment channels in each city. The resulting count plot revealed the distribution of payment methods across different cities. Additionally, a detailed analysis of

payment methods per product line was conducted, visualized through a count plot. This exploration aids in understanding how customers across different cities prefer various payment channels and how these preferences vary across different product lines. Overall, these visualizations serve as an initial exploration, paving the way for deeper insights and more targeted analyses in subsequent stages of the data exploration process.

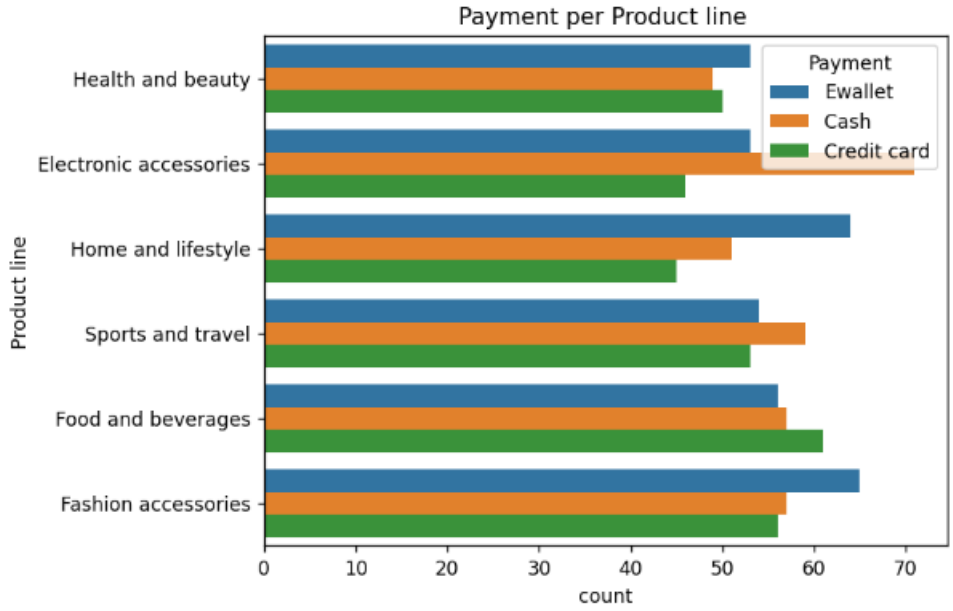
Pairplot for few columns

```
In [66]: 1 sns.pairplot(df[["gross income", "Quantity", "Unit price", "Rating"]], diag_kind="kde")
Out[66]: <seaborn.axisgrid.PairGrid at 0x23aafea63a0>
```



Payment per product line

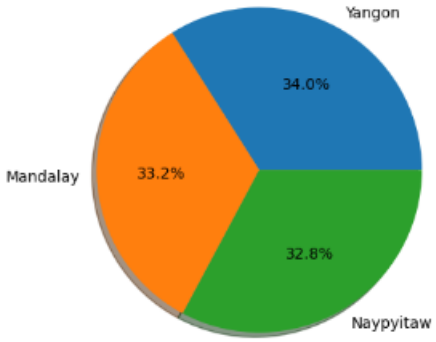
```
In [71]:  1 plt.figure(dpi=125)
          2 sns.countplot(y = "Product line",  hue = "Payment", data =df).set_title('Payment per Product line ')
Out[71]: Text(0.5, 1.0, 'Payment per Product line ')
```



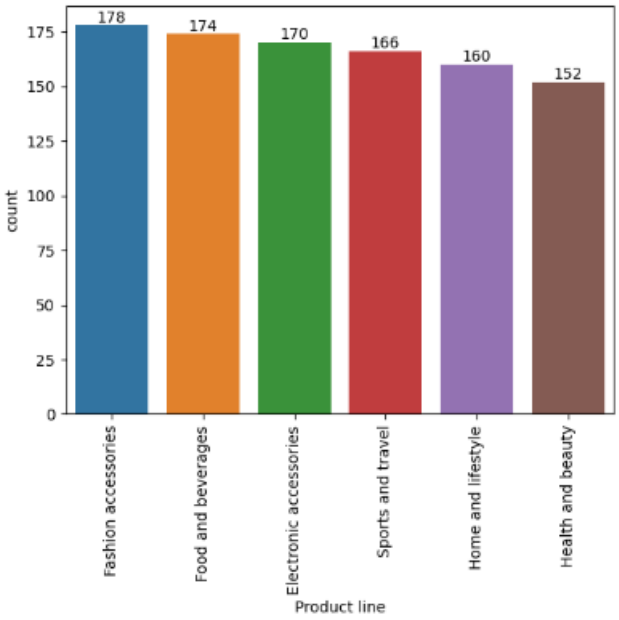
Univariate Analysis:

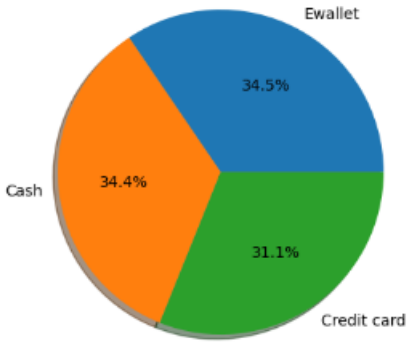
Univariate analysis involves the examination of a single variable or feature at a time, providing insights into its distribution, central tendencies, and other key characteristics. In the provided code, the univariate analysis focused on various columns within the supermarket sales dataset. The functions count_plot, pie_plot, and box_plot were created to facilitate the visualization of categorical columns, pie charts for categorical data, and box plots for numeric data, respectively.

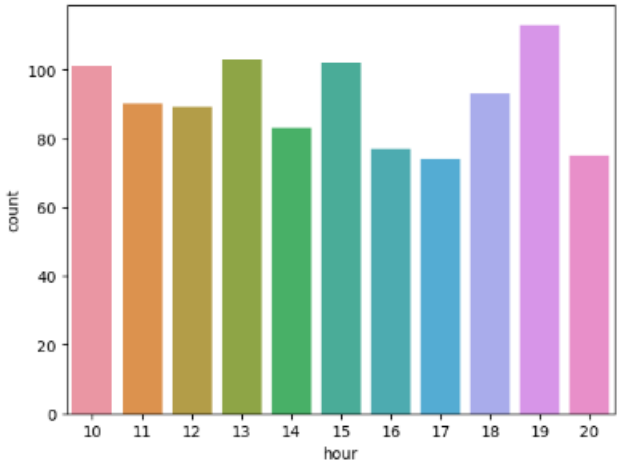
The analysis began by exploring categorical columns such as "City," "Customer type," "Gender," "Product line," "Payment," "Rating," and "Month." The count_plot and pie_plot functions were employed to visualize the distribution and proportions of these categorical variables. Notable insights were derived, such as the prevalence of sales in the city of Yangon, the majority of customers holding membership cards, and a nearly equal gender distribution among shoppers.

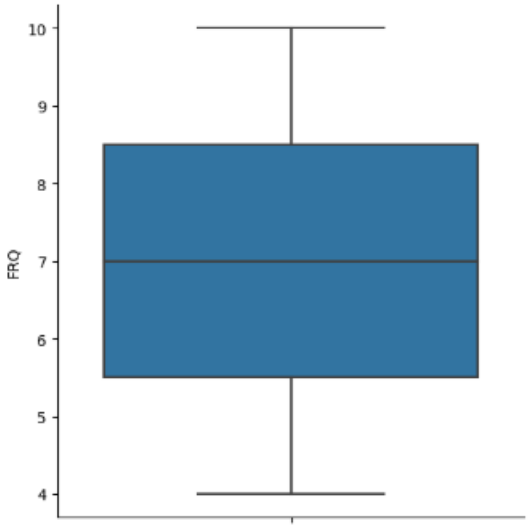


Numeric variables, like the "Rating" column, were examined using the box_plot function, revealing an average rating of approximately 7 across products. The distribution of sales across different months was also explored, showcasing a peak in January. Additionally, the code identified the peak hour for sales, occurring at 7:00 PM.





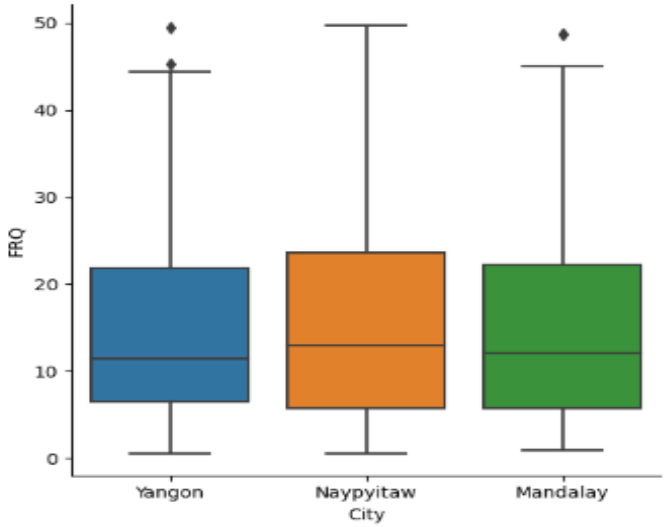


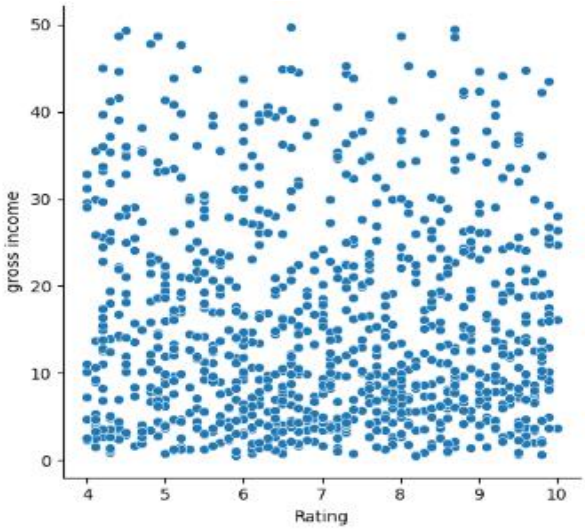


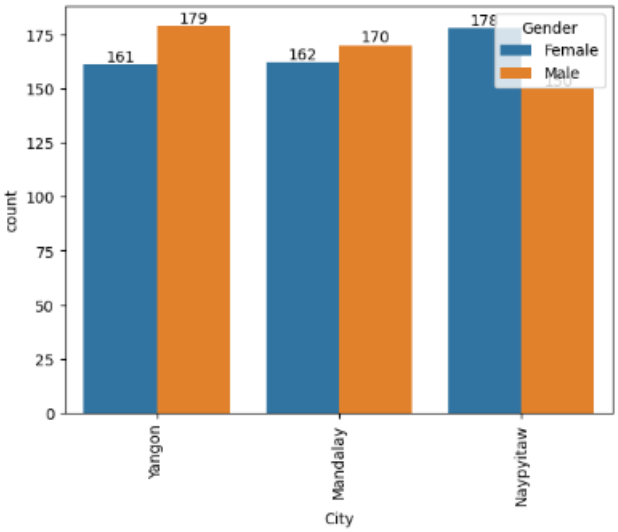
In summary, the univariate analyses conducted in the provided code shed light on the distribution and characteristics of individual variables, offering valuable insights into the supermarket sales dataset.

Bivariate Analysis:

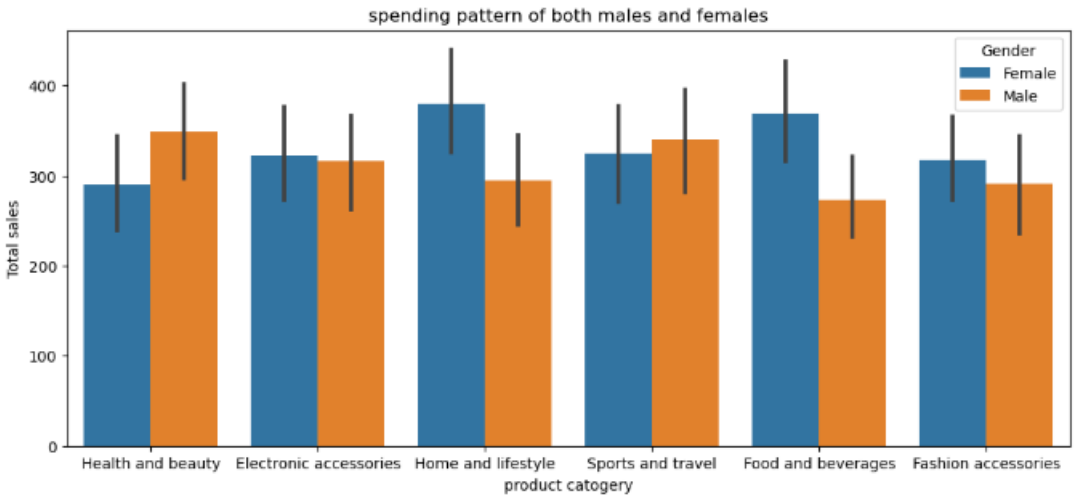
Bivariate analysis involves the examination of the relationships between two variables to uncover patterns, correlations, or dependencies. In the provided code, various bivariate analyses were conducted on the supermarket sales dataset. The relationships between gross income and customer ratings were explored using a scatter plot, revealing no clear correlation between the two variables. Branch-wise comparisons were made to understand the gross income distribution, demonstrating that Branch C outperforms others in terms of profitability. The interaction between gender and customer presence in each branch was visualized, showcasing variations in gender distribution across different branches.

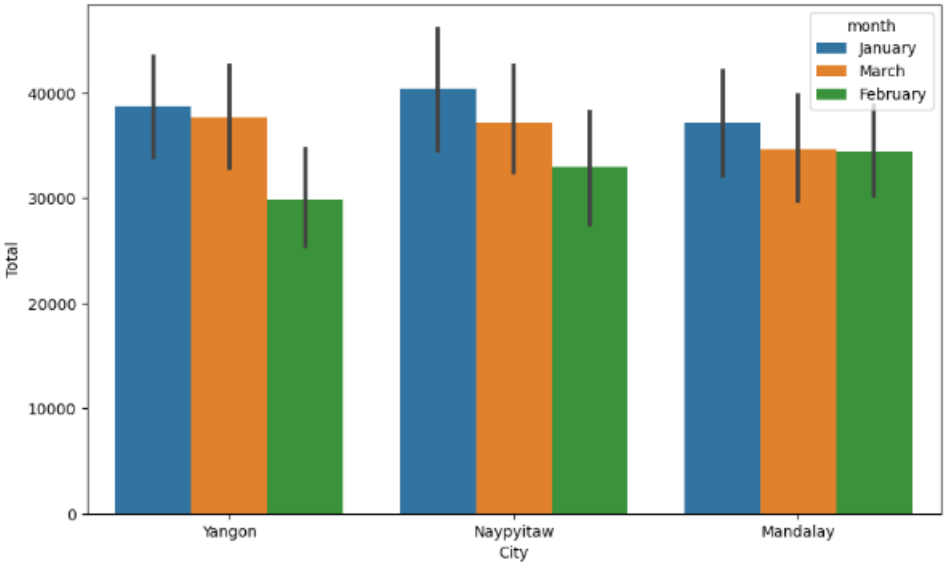


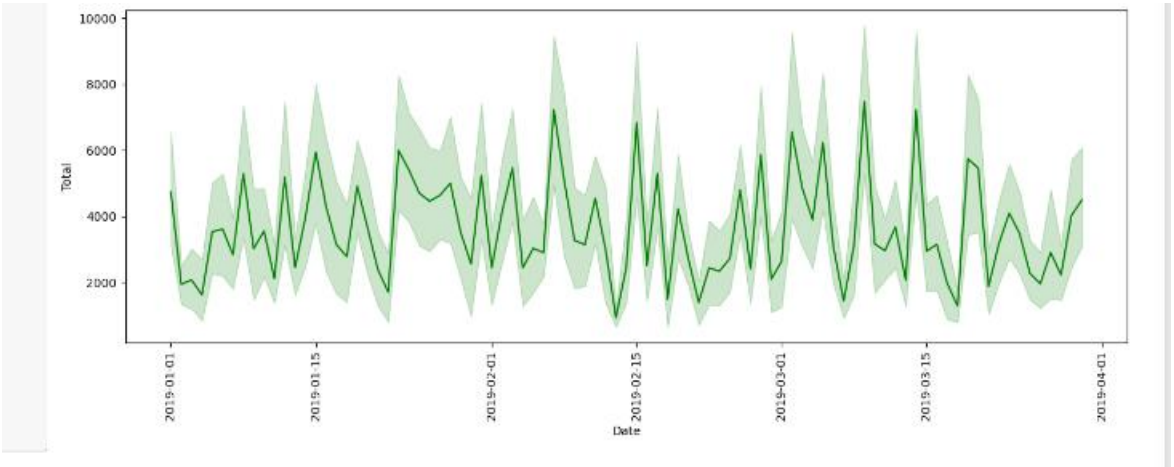




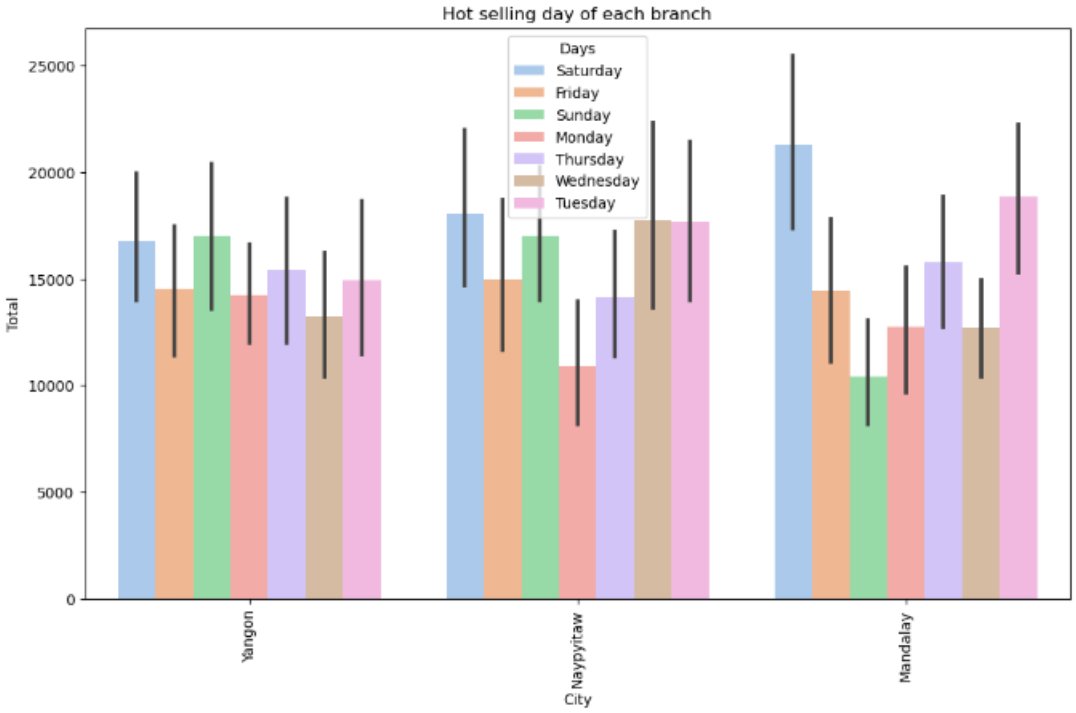
Furthermore, the code investigated the gross income disparity between males and females in each branch, emphasizing that females tend to contribute more to the gross income. The product line sales across branches were analyzed, indicating varying preferences in different cities. The distribution of customer types in each branch revealed that while Yangon and Mandalay had more normal customers, Naypyitaw had a higher number of members. A boxplot illustrated the relationship between gender and gross income, with females contributing more to the supermarket's revenue.

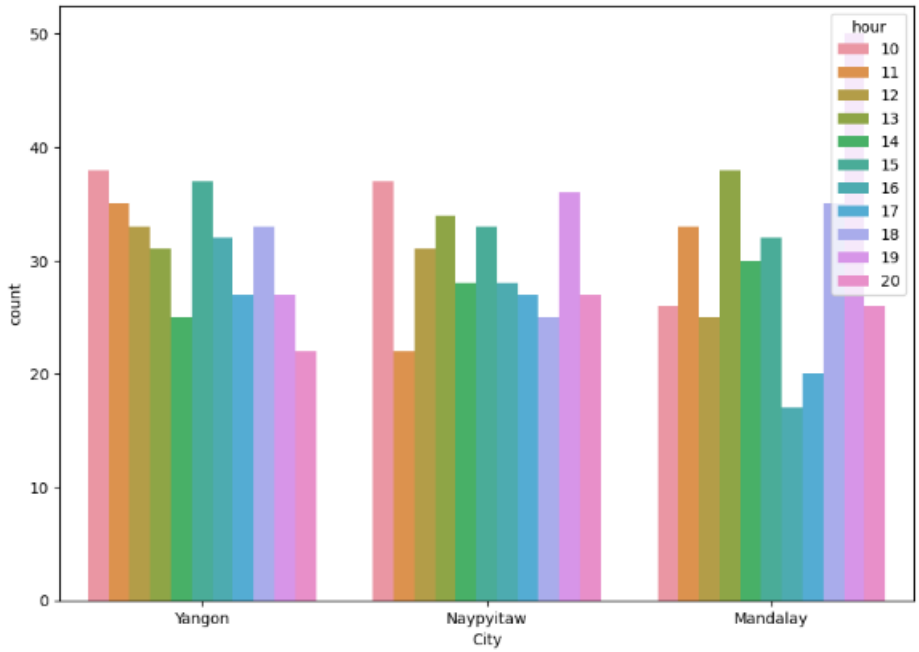






The spending patterns of males and females were scrutinized by product category, demonstrating that females tended to purchase more home and lifestyle products, while males leaned towards health and beauty and sports and travel. The analysis also identified the day of the week with the maximum sales (Saturday) and the hottest selling month (January). Furthermore, the code delved into the trends of each branch with respect to dates and identified the most populated product category (fashion accessories).





In summary, the bivariate analyses performed in the code provided valuable insights behaviors, branch performance, and overall sales trends in the supermarket dataset.

Multivariate Analysis:

Multivariate analysis involves the simultaneous examination and

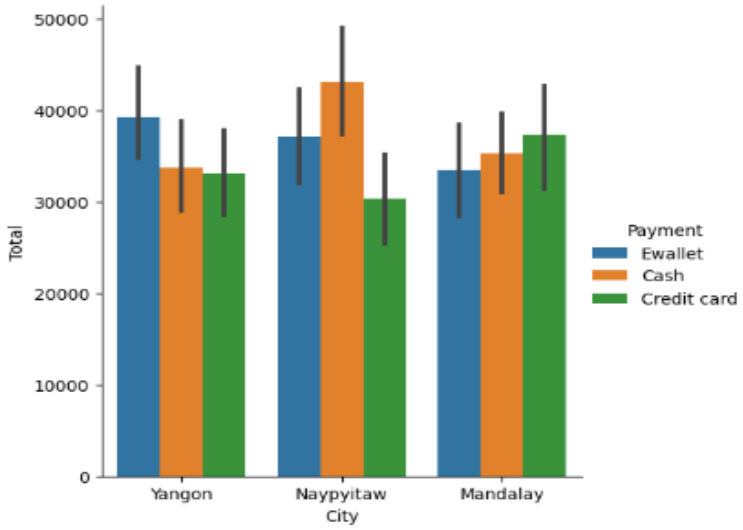
interpretation of relationships between multiple variables in a dataset. It goes beyond bivariate analysis, which explores relationships between two variables, and provides a more comprehensive understanding of the interplay among several factors. In the provided code, various multivariate analyses were conducted on the supermarket sales dataset to unveil complex patterns and interactions.

Customer Type and Total Sales by Branch:

The bar plot with the hue set to customer type illustrates the total sales for each branch, providing insights into which customer type contributes more to sales in each city.

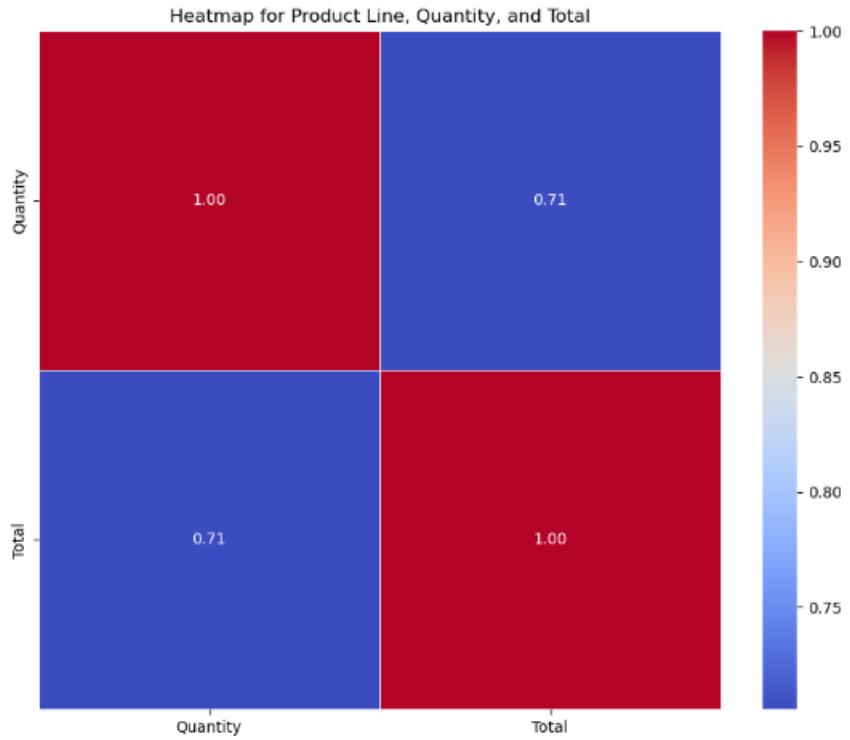
Payment Method and Total Sales by Branch:

Another bar plot was used to visualize the total sales across different payment methods for each branch. This analysis helps identify the preferred payment method in each city and suggests strategies for encouraging certain payment methods.



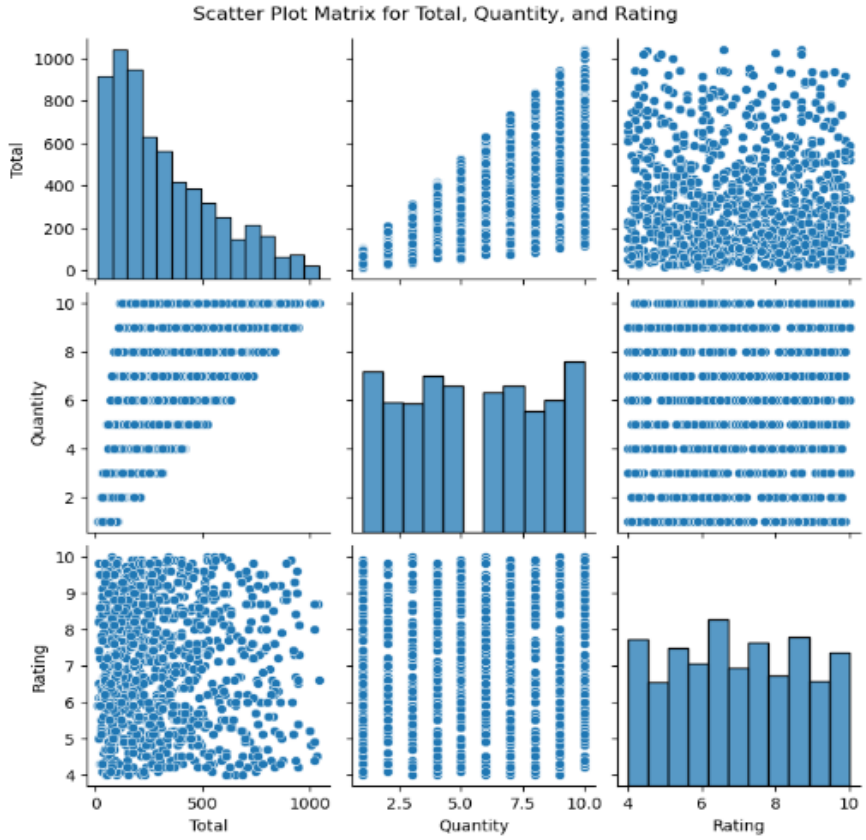
Heatmap for Product Line, Quantity, and Total:

The heatmap depicts the correlation matrix between product line, quantity, and total sales. This allows for a quick understanding of the relationships among these variables.



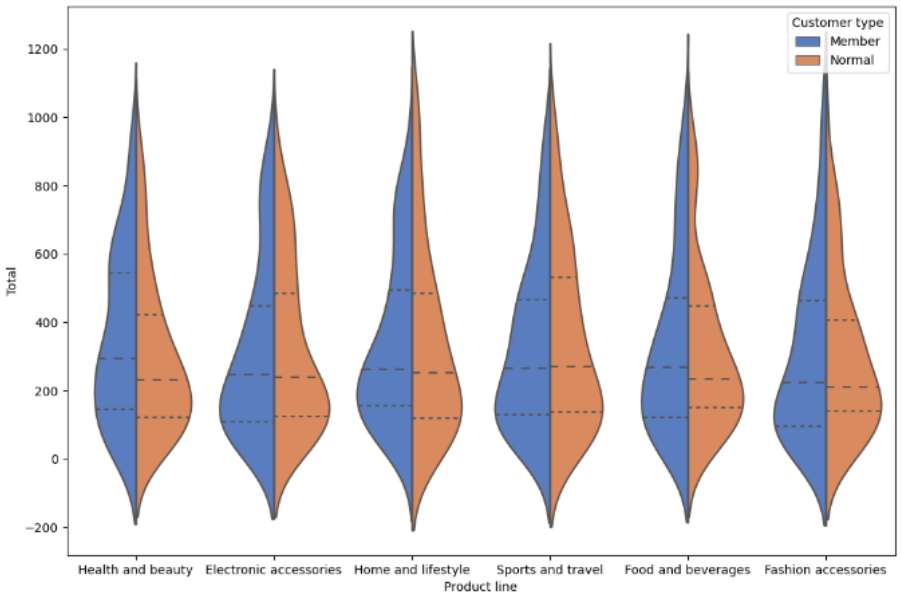
Scatter Plot Matrix for Total, Quantity, and Rating:

The scatter plot matrix explores the relationships between total sales, quantity, and customer ratings. It provides a visual representation of how these variables interact with each other.



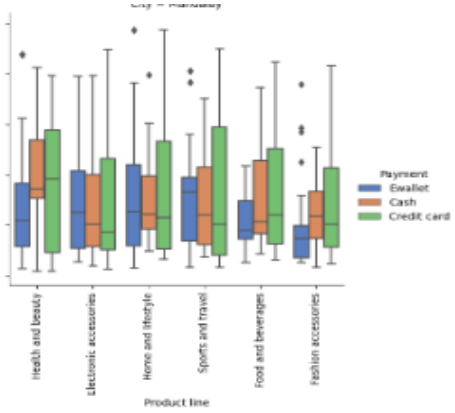
Distribution of Total Sales Across Product Lines, Branches, and Customer Types:

Violin plots showcase the distribution of total sales across different product lines, branches, and customer types. This analysis can reveal patterns and variations in sales based on these factors.



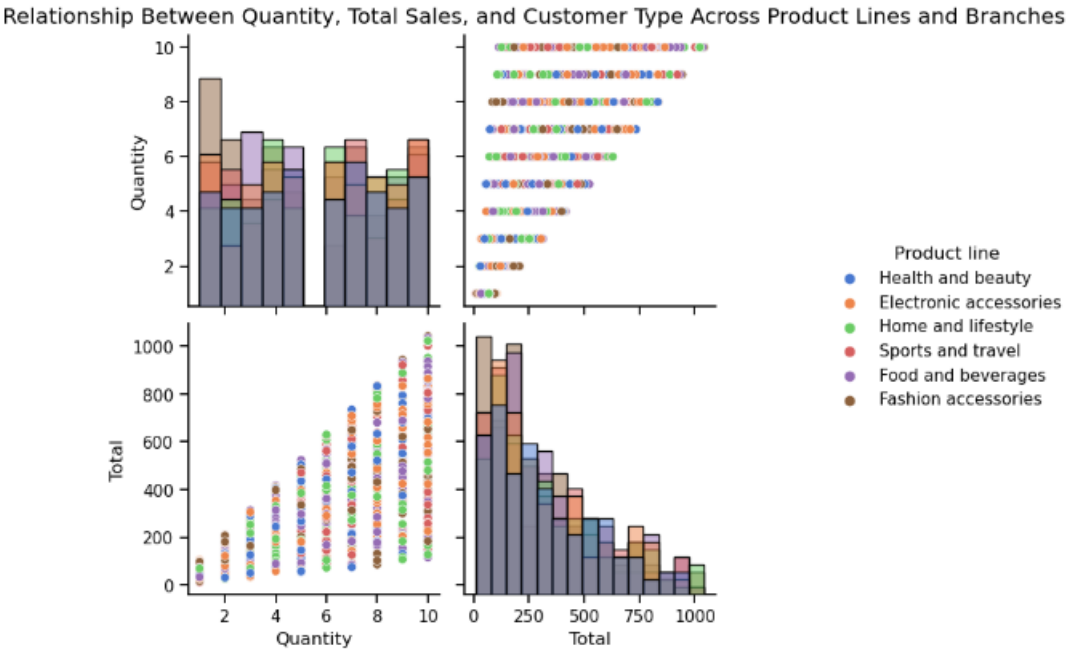
Average Gross Income Across Product Lines, Branches, and Payment Methods:

Box plots were used to represent the average gross income across different product lines for each branch, with further differentiation based on the payment method. This provides insights into the variability in gross income concerning these factors.



Relationship Between Quantity, Total Sales, and Customer Type Across Product Lines and Branches:

The scatter matrix explores the relationship between quantity, total sales, and customer types across different product lines and branches. This can help identify trends and correlations among these variables.



Insights from these multivariate analyses include understanding the impact of customer types on total sales, recognizing preferences in payment methods, and exploring the relationships between various factors like product lines, quantities, and total sales across different branches and customer types. These insights can guide strategic decision-making and marketing efforts for the supermarket.

Distributions:

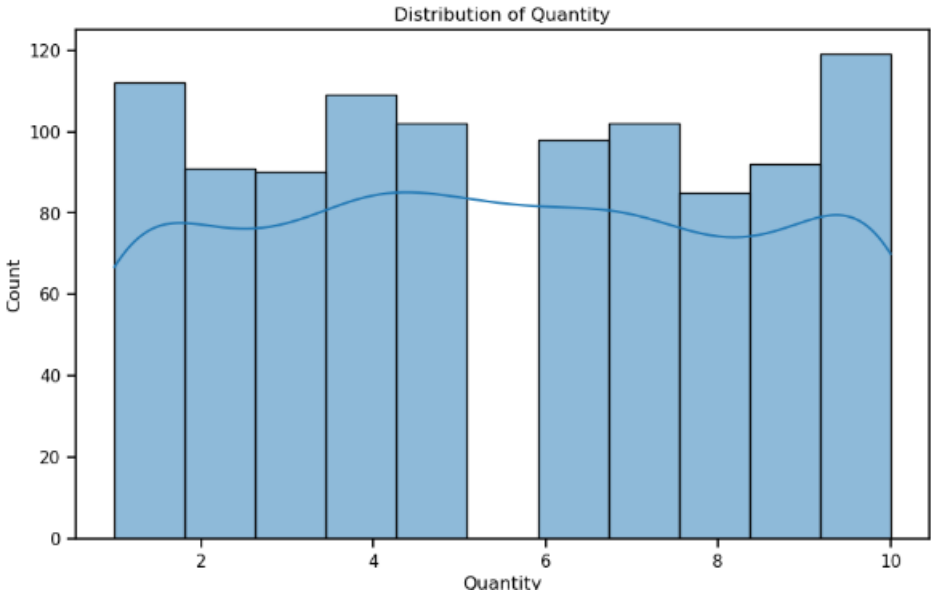
Normal Distribution:

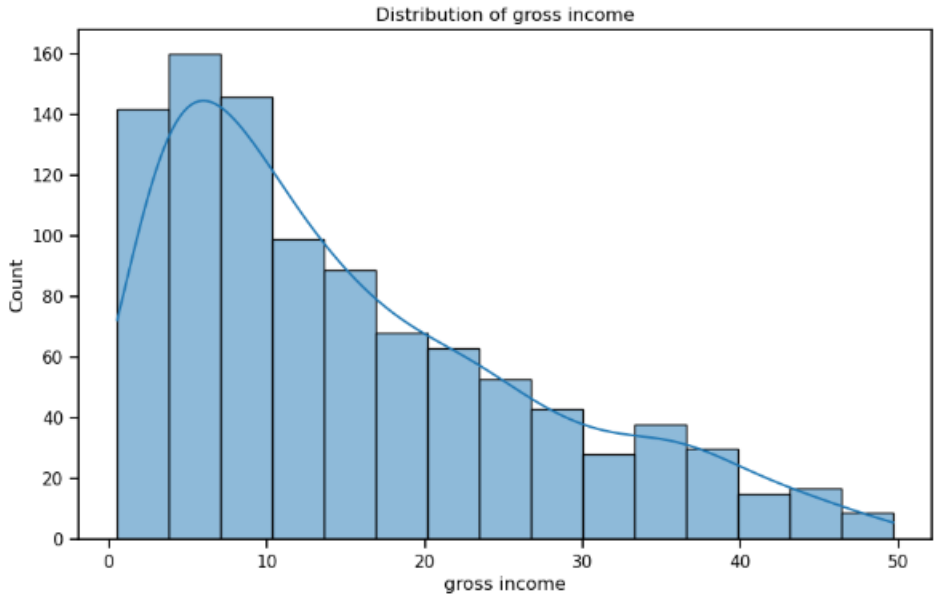
A normal distribution, also known as a Gaussian distribution or bell curve, is a symmetric probability distribution characterized by a bell-shaped curve. In a normal distribution, data is evenly distributed around the mean, and the majority of observations cluster near the center, with fewer occurrences as values deviate further from the mean. The distribution is defined by two parameters: mean (μ) and standard deviation (σ). Many natural phenomena and random processes tend to exhibit a normal distribution, making it a fundamental concept in statistics.

Uses of Normal Distribution in EDA:

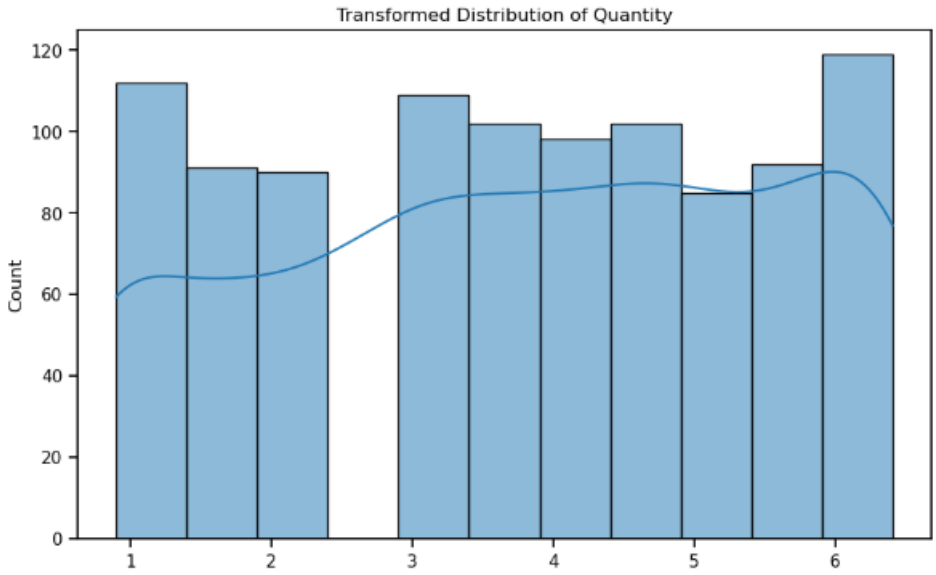
Understanding whether a dataset follows a normal distribution is crucial in exploratory data analysis (EDA). Many statistical techniques assume a normal distribution, and deviations from normality can impact the reliability of statistical tests and model assumptions. Normal distributions facilitate the application of parametric statistical methods, such as t-tests and ANOVA, which rely on assumptions of normality. Additionally, normality is a key consideration when performing transformations or normalization of data to make it more amenable to certain analyses.

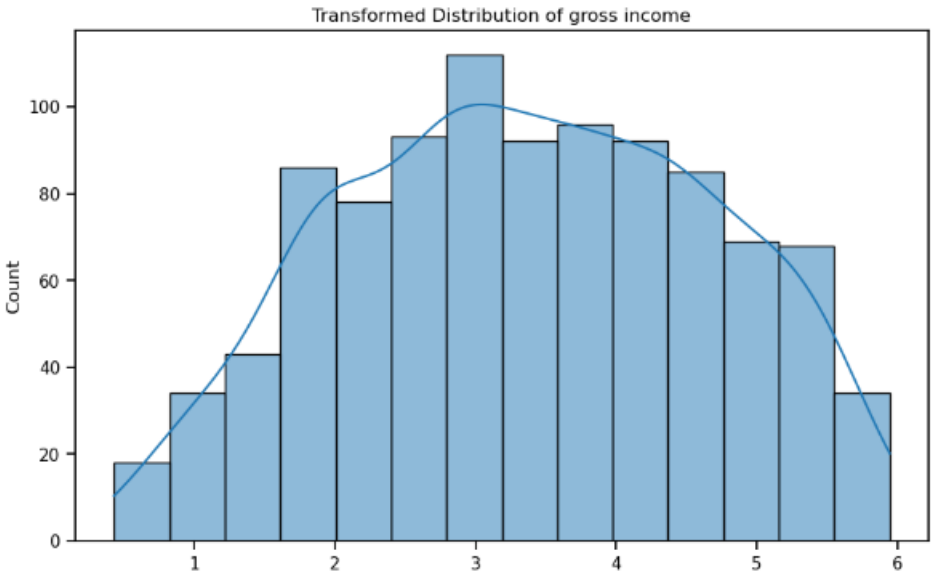
Distribution in the project:





The provided code aims to analyze and normalize the distribution of selected columns ('Total', 'Quantity', 'gross income') in the supermarket sales dataset. It begins by visualizing the original distribution using histograms with kernel density estimates. Following this, a normality test (the Shapiro-Wilk test) is conducted, producing a p-value that indicates whether the data follows a normal distribution. If the p-value is below a significance level (commonly 0.05), the data is considered non-normally distributed. In such cases, a Box-Cox transformation is applied to make the distribution more normal. The transformed data is then visualized again to observe the impact of normalization. The normality test results and transformation process are reported for each column.





Above images are the distributions after normalizing.

Hypothesis Testing:

Hypothesis testing is a statistical method used to make inferences about population parameters based on a sample of data. It involves formulating a null hypothesis (H0) and an alternative hypothesis (H1), then using statistical tests to assess the evidence against the null hypothesis. The objective is to determine whether the observed data provides enough evidence to reject the null hypothesis in favor of the alternative hypothesis. Commonly used tests include t-tests for means, chi-square tests for proportions, and ANOVA for comparing means of multiple groups.

Explanation in the Code:

The provided code demonstrates two different hypothesis tests:

Total Sales and Member Type:

Null Hypothesis (H0): There is no significant difference in total sales between Members and Normal customers.

Alternative Hypothesis (H1): There is a significant difference in total sales between

Members and Normal customers.The code uses a two-sample t-test to compare the total sales of Members and Normal customers.

Result: The p-value (0.53) is greater than the significance level (0.05), so the null hypothesis is not rejected. The conclusion is that there is no significant difference in total sales between Members and Normal customers.

Average Ratings and City:

Null Hypothesis (H0): There is no significant difference in average ratings between the branches (Cities).

Alternative Hypothesis (H1): There is a significant difference in average ratings between the branches.

The code uses an ANOVA (Analysis of Variance) test to compare the average ratings among the branches (Cities).

Result: The F-statistic and p-value are calculated, but the p-value is NaN (Not a Number), which may indicate issues with the data or sample sizes. The code still reports that there is no significant difference in average ratings between the branches.

This hypothesis testing provides a formalized way to assess whether observed differences or patterns in the data are statistically significant, allowing researchers to draw conclusions about population parameters based on sample data.

Conclusion:

In this project, a comprehensive analysis of sales data from three branches of a supermarket was conducted. The exploration covered various aspects, including customer demographics, product preferences, sales patterns across months, days, and hours, as well as multivariate relationships among different variables. Univariate analyses provided insights into the distribution and trends of

individual variables, while bivariate analyses explored relationships between pairs of variables. Multivariate analyses delved deeper into the interplay of multiple factors. Additionally, the distribution of key variables was examined for normality, and hypothesis tests were performed to assess differences in total sales between customer types and average ratings across branches. The findings suggest that there is no significant difference in total sales between Member and Normal customers, and no significant difference in average ratings among the branches. This project contributes valuable insights for strategic decision-making, such as identifying popular products, optimizing inventory, and tailoring marketing strategies to specific customer segments and locations.

Insights:

Based on the analysis of the supermarket dataset, it is recommended to focus on

- Increasing the average unit price and gross margin percentage in order to maximize profits.
- Promoting certain product lines that have high quantity sold can also increase revenue.
- By targeting specific cities and customer types, the number of transactions can be increased.
- Furthermore, improving the rating of the supermarket by gender can lead to increased customer satisfaction and loyalty.
- Implementing strategies to minimize the amount of tax paid can also improve overall profitability.
- Based on the high performance of the Random Forest regression algorithm in predicting the total sales for each city, it is recommended to utilize this algorithm in future sales forecasting.
- By incorporating the other attributes in the dataset, the accuracy of the predictions can be further improved. This can be useful in making informed

business decisions and allocating resources effectively.

- Based on the relatively low performance of the Random Forest classification algorithm in predicting customer ratings, it is recommended to explore other algorithms and/or incorporate additional relevant data to improve the accuracy of the predictions. This can be useful in understanding customer preferences and implementing strategies to improve customer satisfaction and loyalty.
- From this dataset we can see that the percent of male and female are equal by 50-50
- The highest rating recorded amount all customers is between 6 to 7
- The Male customers are using payment method of Ewallet more then the female customers
- The peek sales are at weekends.
- The company could potensonally increase credit card payments by offering incenѲves to customers who currently prefer cash or eWallet payments.
- This could help diversify the modes of payment and possibly increase overall sales.
- The peak business hour is 7 PM.
- This insight can be used to manage staff schedules and ensure maximum efficiency during this one.
- Fashion accessories are the top-selling products, contribuing to 17.80% of sales.

Limitations:

Data Completeness and Quality: The analysis relies on the assumption that the provided dataset is complete and accurate. If there are missing or inaccurate values, it could impact the validity of the conclusions drawn. Further, the dataset's representativeness in terms of the supermarket's overall operations and customer

base is assumed but not guaranteed.

Temporal Scope: The dataset's limited temporal scope might restrict the identification of long-term trends or seasonal patterns. Expanding the timeframe or having data from multiple years could provide a more comprehensive understanding of sales dynamics.

External Factors: The analysis does not account for external factors that could influence sales, such as economic conditions, competitor activities, or local events. Incorporating external data sources could enhance the robustness of the analysis.

Causation vs. Correlation: While the analysis identifies correlations between variables, establishing causation requires more in-depth investigation and controlled experiments. The observed relationships do not necessarily imply a cause-and-effect relationship.

Homogeneity of Branches: The assumption that the three branches are similar in terms of customer behavior and preferences may not hold true. Each branch may have unique characteristics and local influences that are not captured in the dataset.

Normality Assumption: The normality transformations and tests conducted assume that the underlying data distributions are normal. However, the appropriateness of these transformations may be questionable, and the normality tests' results should be interpreted cautiously.

Hypothesis Testing Assumptions: The hypothesis tests, particularly the t-test and ANOVA, rely on assumptions like the normality of data and homogeneity of variances. Violations of these assumptions can impact the reliability of the test results.

Limited Contextual Information: The dataset lacks detailed contextual information, such as marketing promotions, customer feedback, or specific product details. Incorporating such information could provide a more nuanced understanding of the factors influencing sales.

Considering these limitations, it is essential to interpret the findings with caution and recognize that additional data and more sophisticated analyses may be needed for a comprehensive understanding of the supermarket's operations.

Recommendations:

Payment Method Diversification: Encourage and incentivize customers, especially those currently using cash or eWallets, to explore credit card payments. This initiative aims to diversify payment methods, potentially leading to increased overall sales and a broader customer base.

Strategic Business Hour Management: Acknowledging that the peak business hour is at 7 PM, the company can optimize staff schedules to ensure adequate coverage during this busy period. Efficient staffing during peak hours can enhance customer service and satisfaction.

Product Promotion and Stock Management: Given that fashion accessories are the top-selling products, accounting for 17.80% of sales, the company should consider strategic product promotions and marketing campaigns for fashion accessories. Additionally, ensuring sufficient stock availability for these popular items is crucial to meet customer demand.

Targeted Marketing for Credit Card Users: Since male customers are more inclined to use credit cards, targeted marketing campaigns can be designed to attract and retain male customers. Special promotions or discounts for credit card users may be implemented to further encourage this payment method.

Rating Improvement Initiatives: Although the highest recorded ratings fall between 6 to 7, there is potential to improve customer satisfaction further. The company could implement initiatives to enhance customer experience, such as personalized services, loyalty programs, or addressing specific areas identified as opportunities for improvement.

Weekend Sales Promotions: Since weekends experience peak sales, the company can strategically plan sales promotions and events during these periods to capitalize on increased customer traffic. This could include limited-time offers, discounts, or exclusive weekend promotions to attract more customers.

Data Collection and Feedback: To gain a more comprehensive understanding of customer preferences and satisfaction, the company could implement data collection mechanisms and seek direct feedback from customers. This qualitative information can provide valuable insights for future decision-making and improvements.

Explore Partnerships for Fashion Accessories: Considering the significant contribution of fashion accessories to total sales, exploring partnerships with popular brands or designers in this category could further enhance the company's product offerings and attract a wider audience.

By implementing these recommendations, the company can not only capitalize on existing strengths but also address areas for improvement, fostering growth and sustainability in the competitive retail market. Regular monitoring and adaptation of

strategies based on customer feedback and market trends will be essential for ongoing success

References:

- PPT LINK: https://docs.google.com/presentation/d/1Dvts8OyhlB8YHZGJxkK6qLWejr05f5XL/edit?usp=drive_link&ouid=115028152639430586782&rtpof=true&sd=true
- Kaggle: Supermarket sales (kaggle.com)
- CODE LINK: https://drive.google.com/file/d/1sH2RZsC-AEBIvOk7ZxwwSxHrPRNqr0mv/view?usp=drive_link

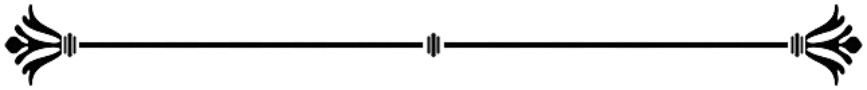
- Python Libraries:
  - Pandas: McKinney, W. (2010). Data Structures for Statistical Computing in Python. In Proceedings of the 9th Python in Science Conference (pp. 51–56).
  - Matplotlib: Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering, 9(3), 90–95.
  - Seaborn: Waskom, M. (2021). seaborn: statistical data visualization. Journal of Open Source Software, 6(60), 3021.
  - Plotly: Sievert, C. (2021). Interactive Web-Based Data Visualization with R, plotly, and shiny. Chapman and Hall/CRC.
  - NumPy: Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., … Oliphant, T. E. (2020). Array Programming with NumPy. Nature, 585(7825), 357–362.

These references were instrumental in accessing, cleaning, and analyzing the IPL dataset, as well as visualizing the insights gained. The combination of reliable data sources and powerful Python libraries facilitated a comprehensive exploratory data analysis (EDA) of the Indian Premier League.

Acknowledgement:

I would like to express my gratitude to Kaggle for providing the comprehensive Supermarket Sales dataset, which served as the backbone of this analysis. Additionally, I extend thanks to the creators of Pandas, Matplotlib, Seaborn, Plotly, NumPy, Ipywidgets, and IPython for developing the powerful libraries and tools that facilitated data manipulation, visualization, and interactive widgets in this project.

Special thanks to the community of data enthusiasts and analysts whose shared knowledge and experiences have been invaluable throughout this exploration.



Thank You!