Data science

Supermarket sales analysis

Rifqi arrayan

TABLE OF CONTENT

Background

A supermarket company is keen to understand various aspects of their operations and sales to improve business strategy and customer satisfaction. With fierce competition in the retail industry, the company needs to identify key factors that influence sales, customer preferences, and operational efficiency. Through the analysis of supermarket sales datasets, the company can gain in-depth insights into sales performance, product preferences, customer behavior, and marketing strategy effectiveness.

Goals

1.Understanding Sales Trends:
Identify peak sales times to optimize operating hours and resource allocation.
View sales performance by branch and city to determine more effective local marketing strategies.

2.Analyze Product Preferences:
Identify the products that customers are most interested in.
Analyze sales by product line to identify products that need to be improved or promoted further.

3.Assess Customer Satisfaction:
Analyze customer rating distribution to understand their level of satisfaction.
Identify areas of improvement based on customer feedback.

4.Customer Segmentation:
Analyze sales differences based on customer type (member vs. non-member) and gender to adjust marketing strategies.
Understand customer preferences based on payment methods used.

TOOLS

Google colab is used as platform for analyzing data

Google Sheets is used to make data easier preprocessing is like removing duplicates

pyhton is used data for data analysis processes

TABLE OF CONTENT

Data Preparation

data is taken from kaggle where this data contains sales data on a supermarket to be analyzed to get insights that are useful for the company, and also get maximum profit.

Rows
1000

Columns
17

```
from google.colab import files
df = files.upload()
```

Choose Files supermarke... Sheet1.csv

- **supermarket_sales - Sheet1.csv**(text/csv)

```
Saving supermarket_sales - Sheet1.csv
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 17 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Invoice ID               1000 non-null   object 
 1   Branch                   1000 non-null   object 
 2   City                     1000 non-null   object 
 3   Customer type            1000 non-null   object 
 4   Gender                   1000 non-null   object 
 5   Product line             1000 non-null   object 
 6   Unit price               1000 non-null   float64
 7   Quantity                 1000 non-null   int64  
 8   Tax 5%                   1000 non-null   float64
 9   Total                    1000 non-null   float64
 10  Date                     1000 non-null   object 
 11  Time                     1000 non-null   object 
 12  Payment                  1000 non-null   object 
 13  cogs                     1000 non-null   float64
 14  gross margin percentage  1000 non-null   float64
 15  gross income             1000 non-null   float64
 16  Rating                   1000 non-null   float64
dtypes: float64(7), int64(1), object(9)
```

Exploraty data analysis

	Unit price	Quantity	Tax 5%	Total	cogs	gross margin percentage	gross income	Rating
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.00000	1000.000000	1000.000000	1000.00000
mean	55.672130	5.510000	15.379369	322.966749	307.58738	4.761905	15.379369	6.97270
std	26.494628	2.923431	11.708825	245.885335	234.17651	0.000000	11.708825	1.71858
min	10.080000	1.000000	0.508500	10.678500	10.17000	4.761905	0.508500	4.00000
25%	32.875000	3.000000	5.924875	124.422375	118.49750	4.761905	5.924875	5.50000
50%	55.230000	5.000000	12.088000	253.848000	241.76000	4.761905	12.088000	7.00000
75%	77.935000	8.000000	22.445250	471.350250	448.90500	4.761905	22.445250	8.50000
max	99.960000	10.000000	49.650000	1042.650000	993.00000	4.761905	49.650000	10.00000

```
df.duplicated().sum()
```

```
0
```

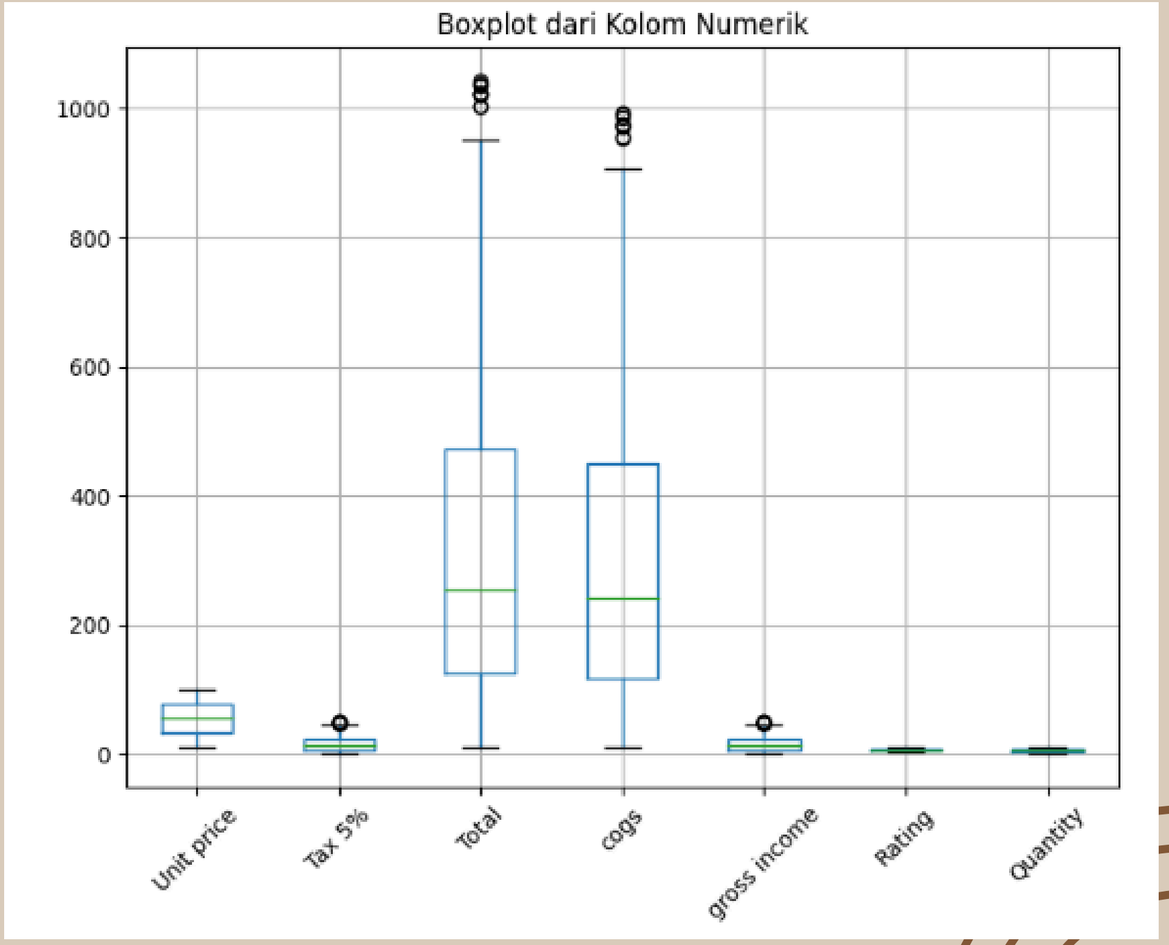
```
df.isna().sum().sort_values(ascending=False)
```

```
Invoice ID                0
Total                     0
gross income              0
gross margin percentage   0
cogs                      0
Payment                   0
Time                      0
Date                      0
Tax 5%                    0
Branch                    0
Quantity                  0
Unit price                0
Product line              0
Gender                    0
Customer type             0
City                      0
Rating                    0
dtype: int64
```

Processing data

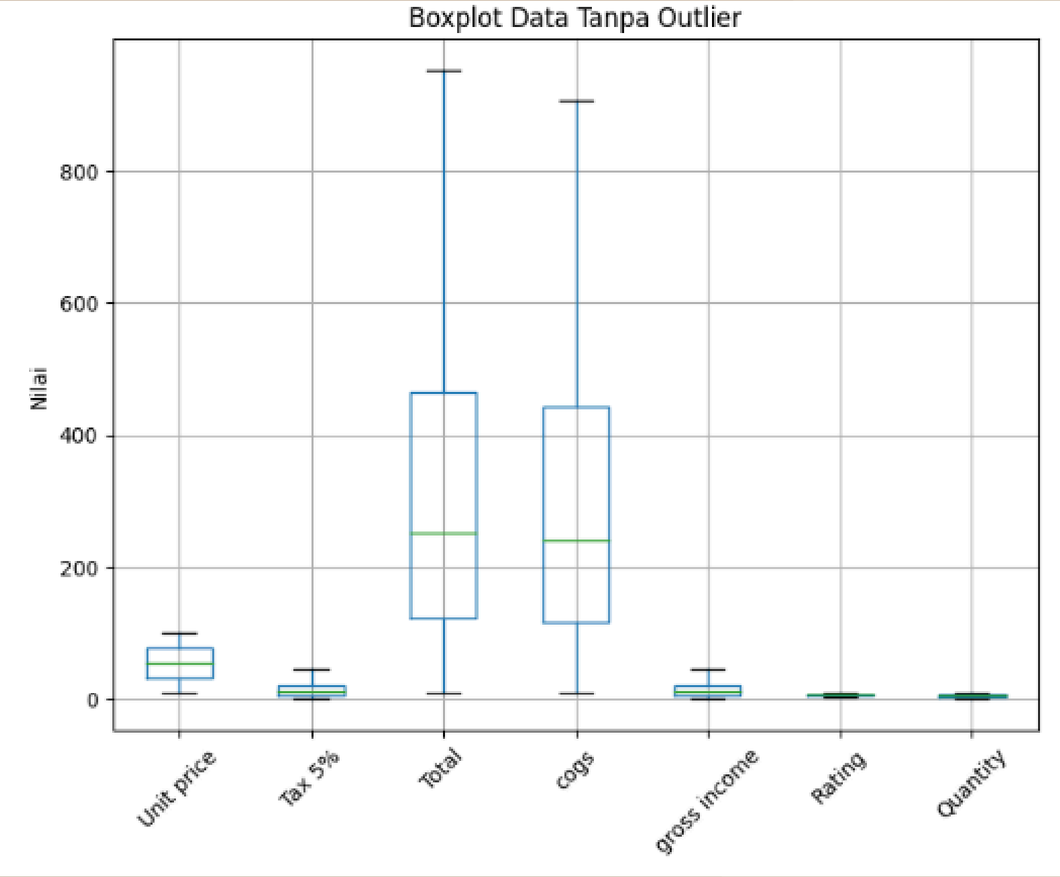
Checking outlier

```
Data outlier:
Unit price      0
Tax 5%          9
Total           9
cogs            9
gross income    9
Rating          0
Quantity        0
dtype: int64
```



It can be seen that there are outliers in this dataset, although the number is not too significant.

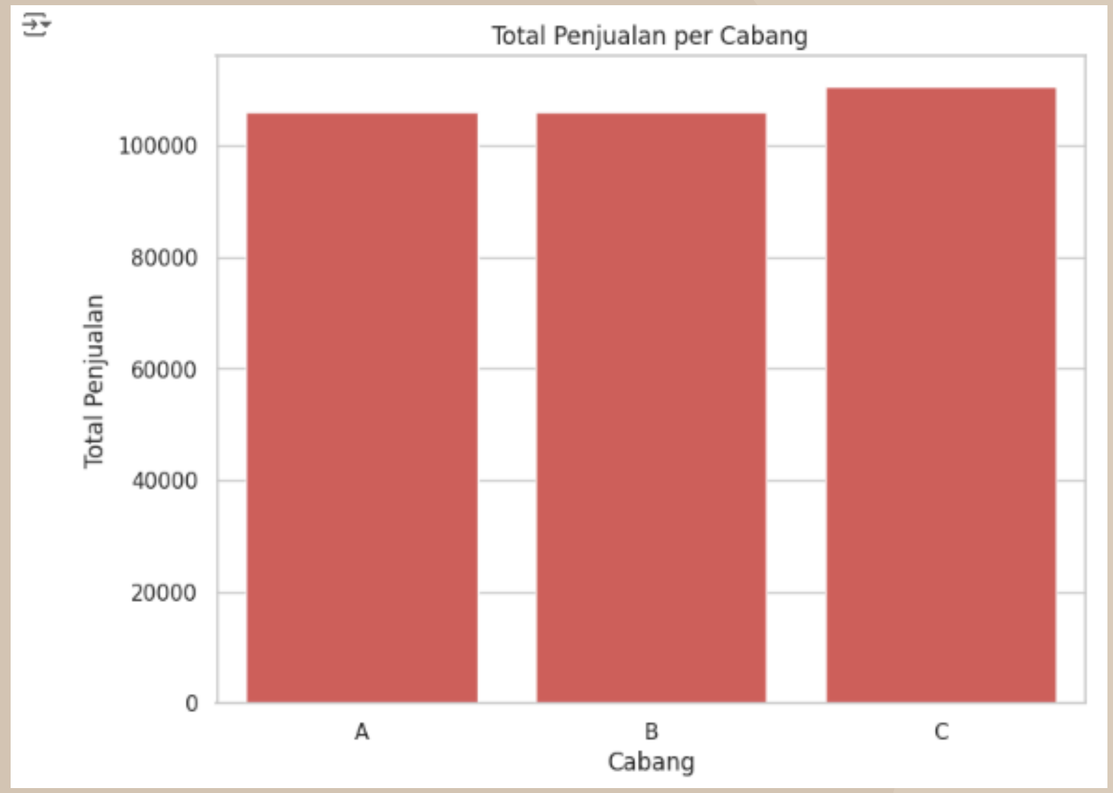
Deleting outlier



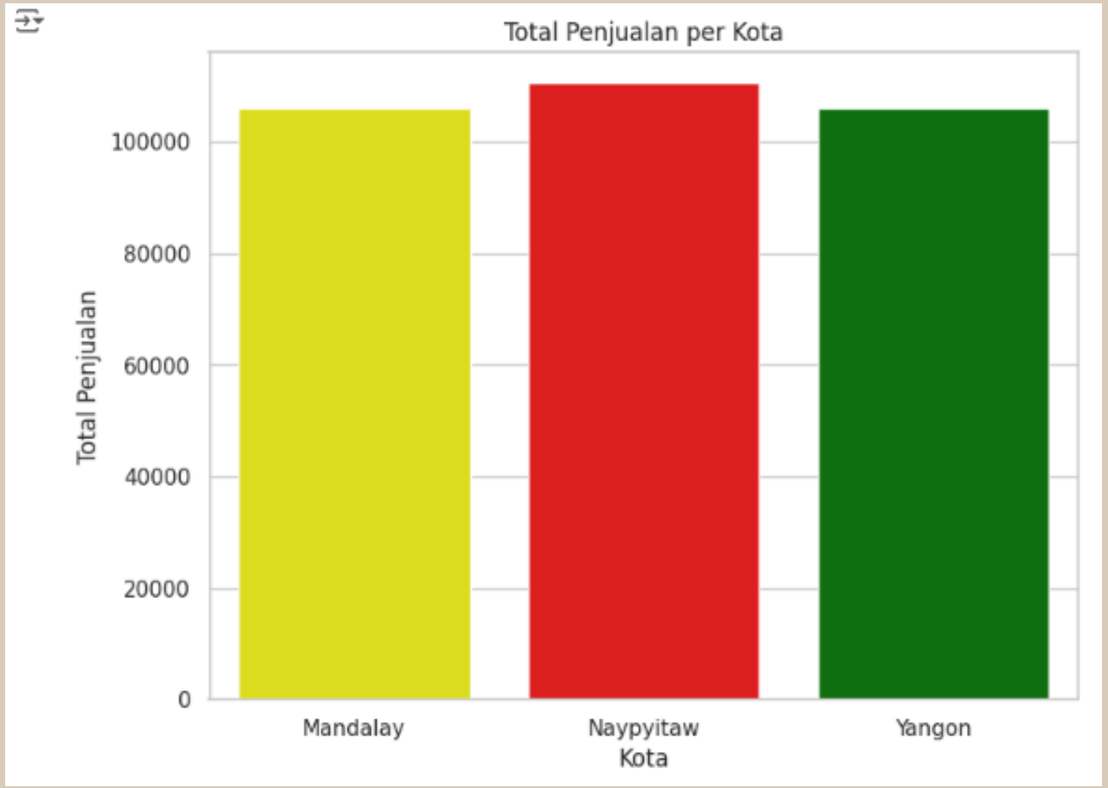
Because the number of outliers in this dataset is not too significant (many), I removed the outliers.

Data insights

analysis of total sales per branch

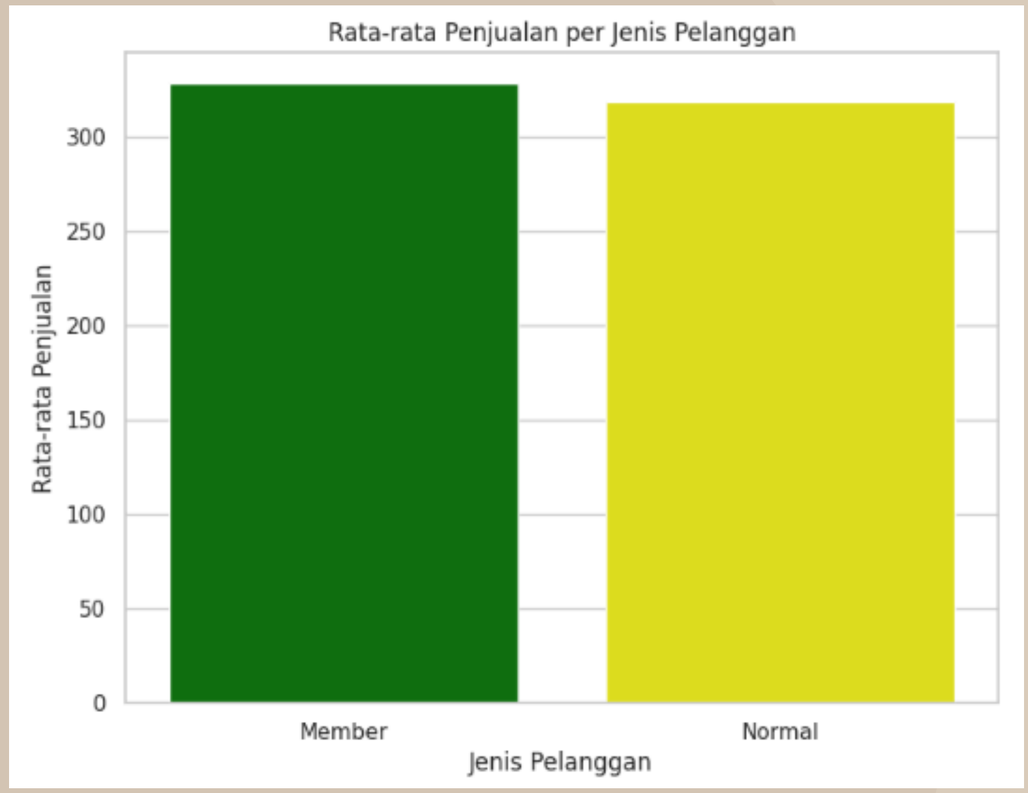


analysis of total sales per city

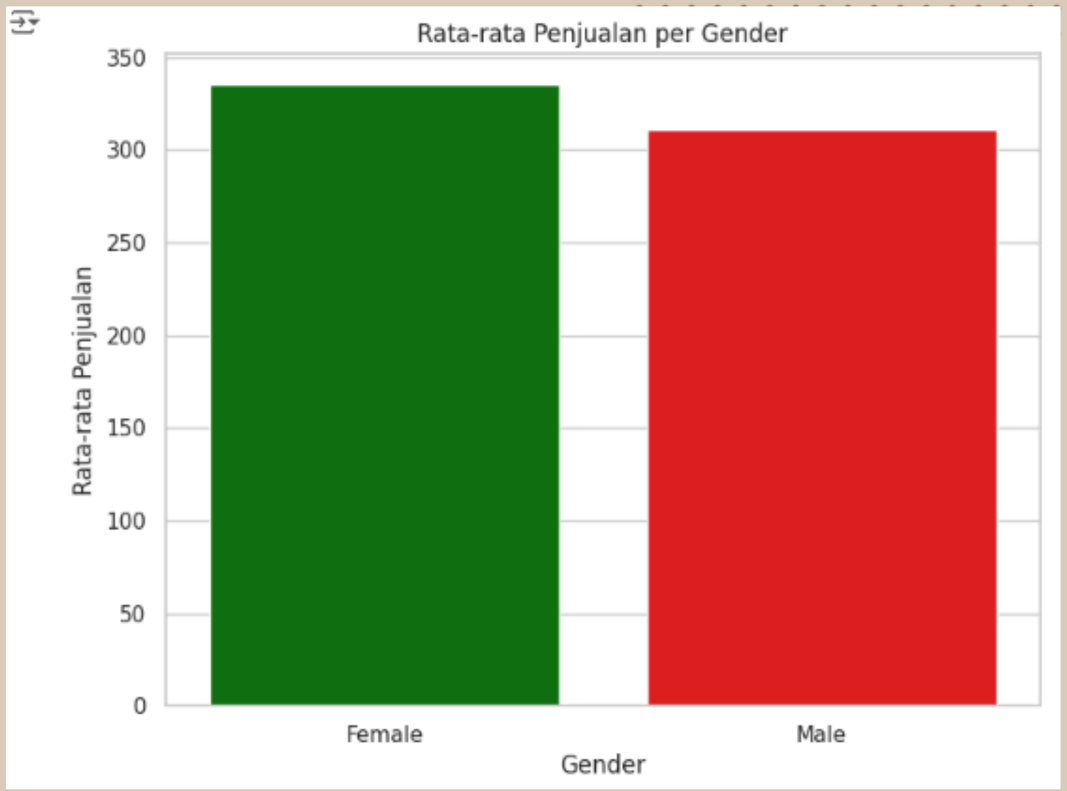


Analysis results:
-The branch with the highest sales is branch C.
The city with the highest sales is Naypyitaw.
Recommendation:
-Focus on promotions in branches and cities with high sales such as branch C in Naypyitaw to capitalize on the large customer base.
-Conduct marketing campaigns to increase sales in branches and cities with lower sales.

analysis of average sales per customer type

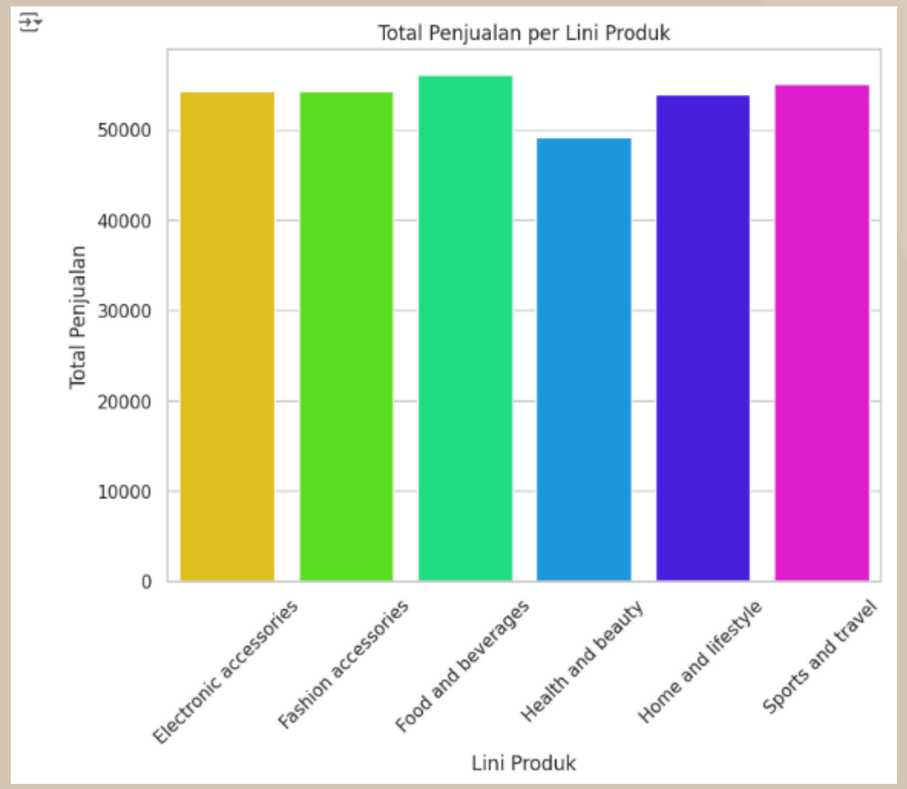


analysis of average sales based on gender

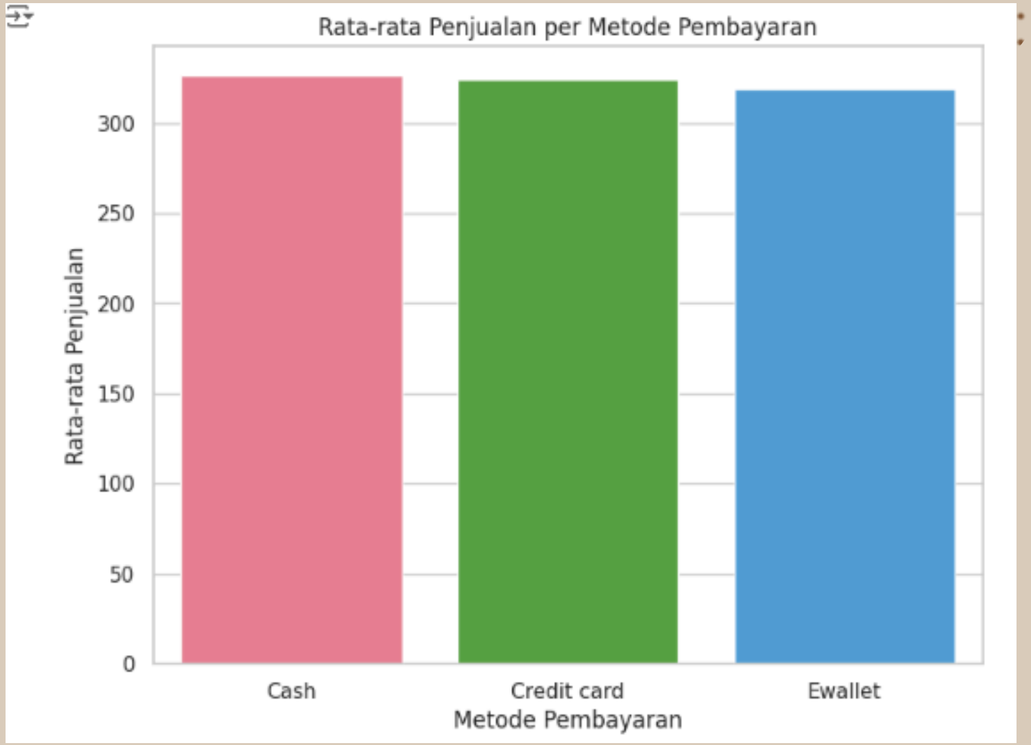


Analysis results:
-Average sales per transaction is higher for member customers than normal customers.
-Average sales per transaction is higher for women than men.
Recommendation:
-Increase stock and promotion for men's product line

sales analysis by product

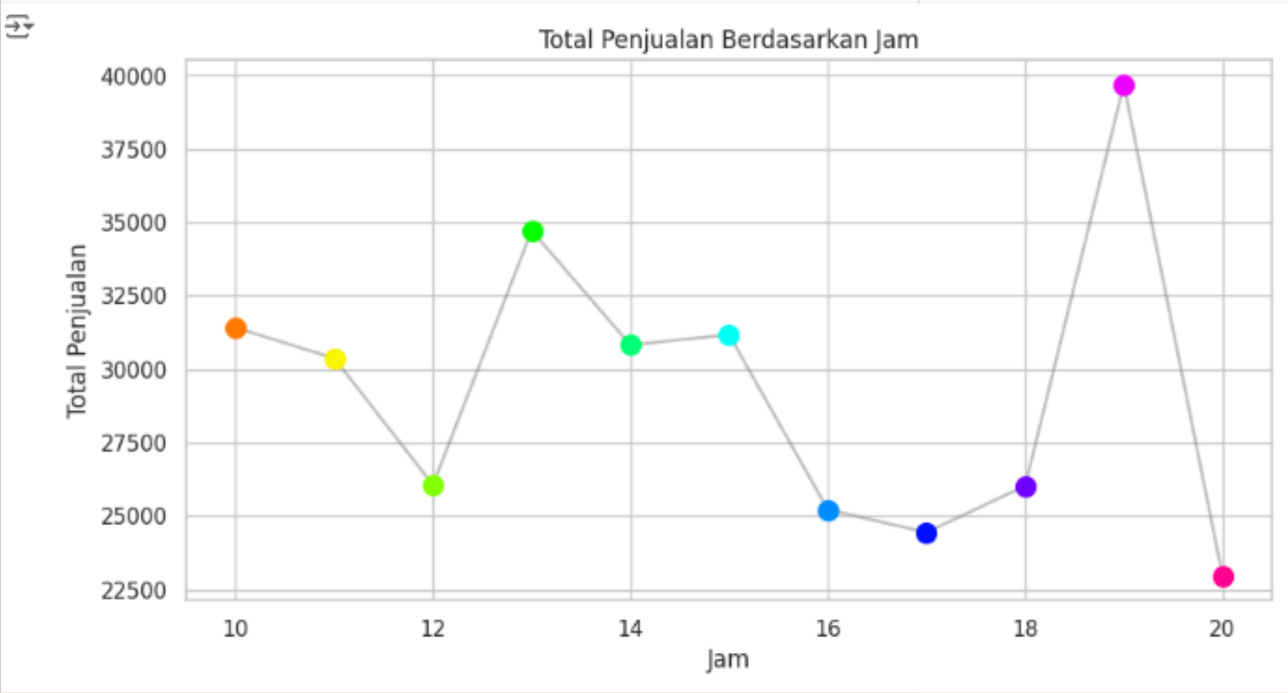


sales analysis by payment method



Analysis results:
- “food and beverages” is the product line with the highest total sales.
-The highest sales average is using cash, credit card then e-wallet.
Recommendation:
-Provide discounts or cashback for Ewallet payments to encourage the use of this popular payment method.

hourly sales analysis



Analysis results:

-The highest sales occur during 13:00 to 15:00 and 18:00 to 20:00.

There are certain hours where it drops drastically from 10am to 12am

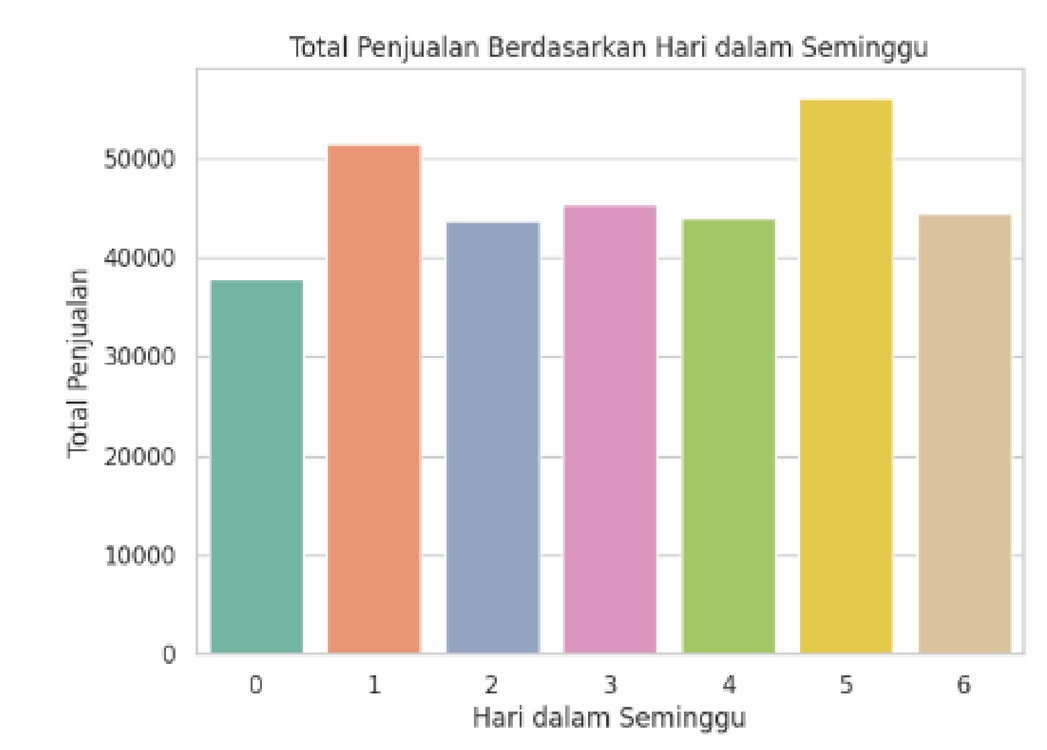
Recommendation:

-Offer limited-time promotions during low sales hours to attract more customers.

Increase staff during peak hours to improve service efficiency.

analysis of total sales by month

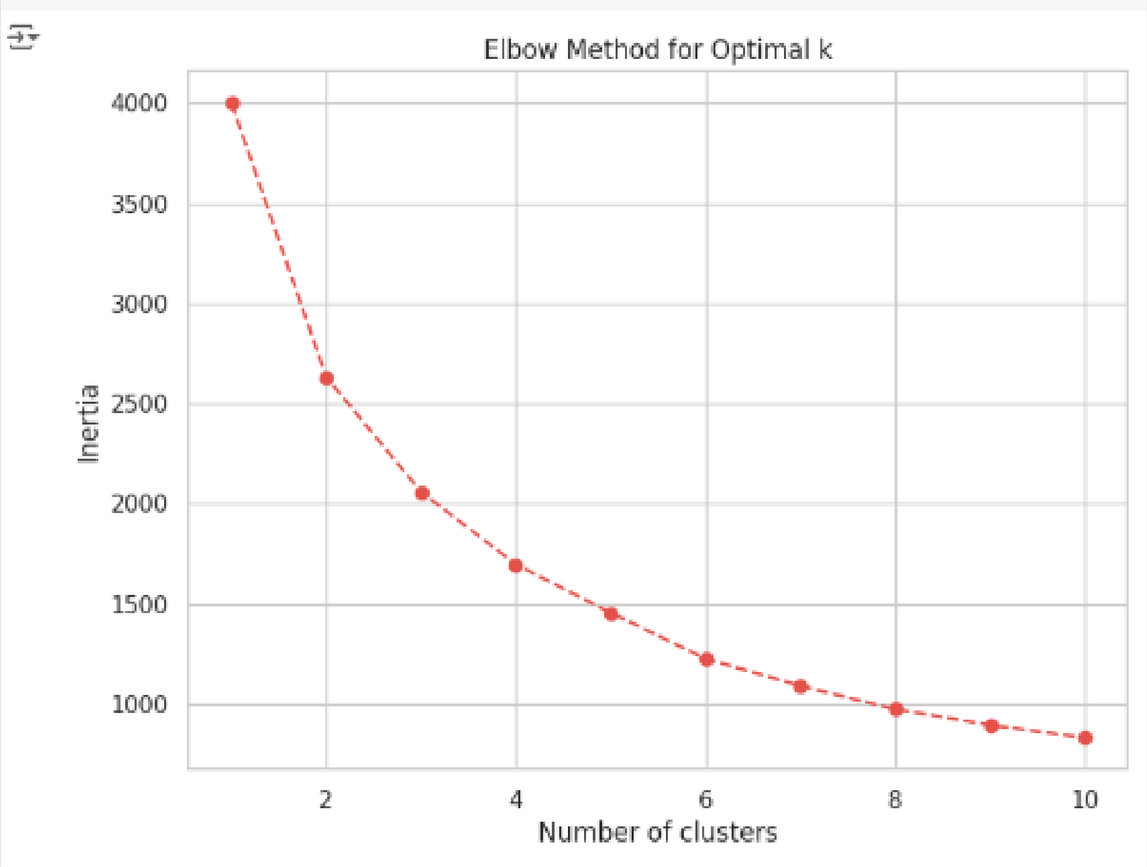
sales analysis by day of the week



Analysis results:
-Total sales in the second month decreased dramatically
-and on Monday (0) the lowest total sales
Recommendation:
-do offers with discounted prices in the second month and on Mondays
-do promotions and attractive offers such as buy 1 free 1

Modelling data

Elbow method

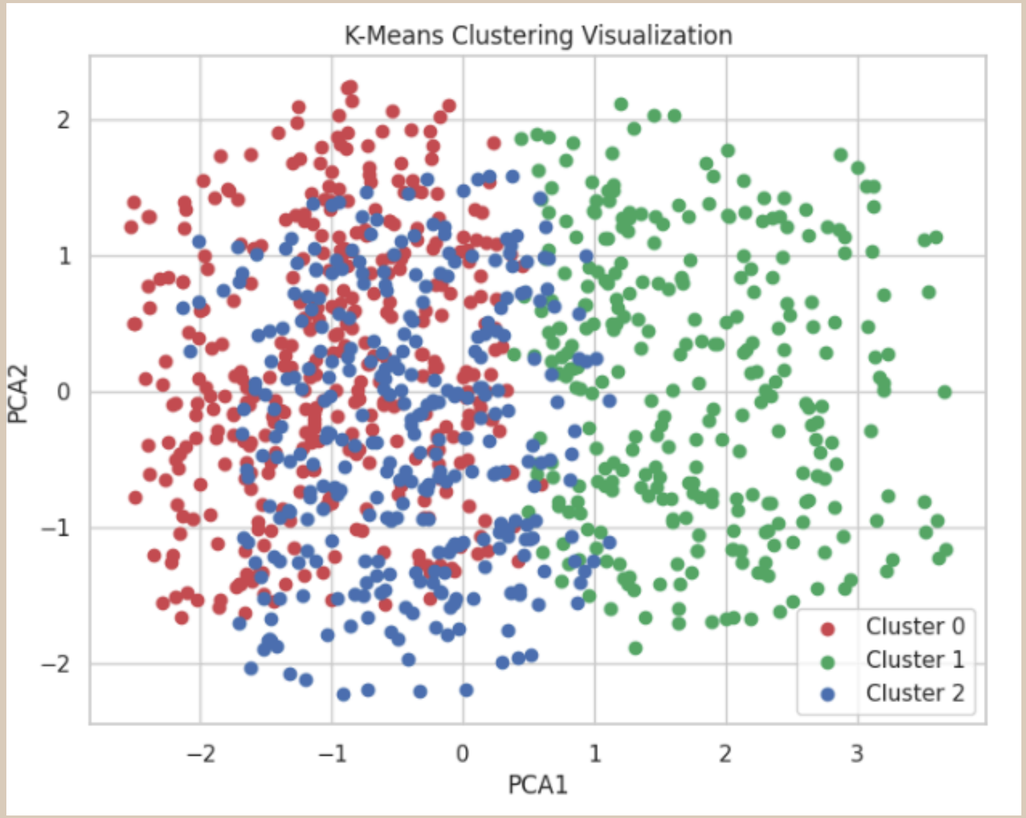


in the elbow method on the side I can see that a drastic decrease is seen at point (cluster) to 3 the rest is constant

application of K-means clustering

```python
from sklearn.cluster import KMeans

# menerapkan k-means clustering dengan membagi data menjadi 3 cluster
kmeans = KMeans(n_clusters=3, random_state=42)
kmeans.fit(scaled_features)

#  mengambil label klaster dari kmeans.labels_ dan menambahkannya sebagai kolom baru bernama 'Cluster'
df['Cluster'] = kmeans.labels_

# mencetak lima baris pertama dari DataFrame yang telah di cluster sebanyak 3 cluster
print(df.head())
```

```
   Invoice ID Branch       City Customer type  Gender  \
0  750-67-8428      A     Yangon        Member  Female
1  226-31-3081      C  Naypyitaw        Normal  Female
2  631-41-3108      A     Yangon        Normal    Male
3  123-19-1176      A     Yangon        Member    Male
4  373-73-7910      A     Yangon        Normal    Male

             Product line  Unit price  Quantity   Tax 5%     Total       Date
0       Health and beauty       74.69         7  26.1415  548.9715   1/5/2019
1  Electronic accessories       15.28         5   3.8200   80.2200   3/8/2019
2      Home and lifestyle       46.33         7  16.2155  340.5255   3/3/2019
3       Health and beauty       58.22         8  23.2880  489.0480  1/27/2019
4       Sports and travel       86.31         7  30.2085  634.3785   2/8/2019

    Time      Payment    cogs  gross margin percentage  gross income  Rating  \
0  13:08      Ewallet  522.83                 4.761905       26.1415     9.1
1  10:29         Cash   76.40                 4.761905        3.8200     9.6
2  13:23  Credit card  324.31                 4.761905       16.2155     7.4
3  20:33      Ewallet  465.76                 4.761905       23.2880     8.4
4  10:37      Ewallet  604.17                 4.761905       30.2085     5.3

   Hour  Cluster
0    13        1
1    10        2
2    13        2
3    20        1
4    10        1
```




```
Cluster
0    381
2    320
1    299
Name: count, dtype: int64
  Cluster Unit price                Quantity                Total                \
                mean        std       mean       std        mean         std
0       0  57.665302  23.926375   2.467192  1.238462  152.165614   99.147288
1       1  79.635652  14.006307   7.775920  1.796703  641.440119  166.010701
2       2  30.908094  13.226590   7.015625  1.943498  228.753295  114.325898

     Rating
       mean       std
0  6.934383  1.678188
1  6.860535  1.774176
2  7.123125  1.708167
```

clustering conclusion

Cluster 0:
Unit Price: Average 57.67 with a standard deviation of 23.93.
Quantity: Average 2.47 with a standard deviation of 1.24.
Total: Average 152.17 with a standard deviation of 99.15.
Rating: Mean 6.93 with a standard deviation of 1.68.

Characteristics: Customers in this cluster tend to purchase products at moderate unit prices and in smaller quantities. The total purchase value is also relatively lower than other clusters.

Strategy: Focus on selling products with medium unit prices. It may be necessary to increase customer purchase volume by offering discounts for larger purchases.

clustering conclusion

Cluster 1:

Unit Price: Average 79.64 with a standard deviation of 14.01.
Quantity: Average 7.78 with a standard deviation of 1.80.
Total: Average 641.44 with a standard deviation of 166.01.
Rating: Average 6.86 with a standard deviation of 1.77.

Characteristics: Customers in this cluster tend to purchase products with high unit prices and large quantities. The total purchase value is also the highest among all clusters.

Strategy: Premium customers who buy in bulk. Focus on loyalty programmes and exclusive offers to retain these customers and increase their satisfaction.

clustering conclusion

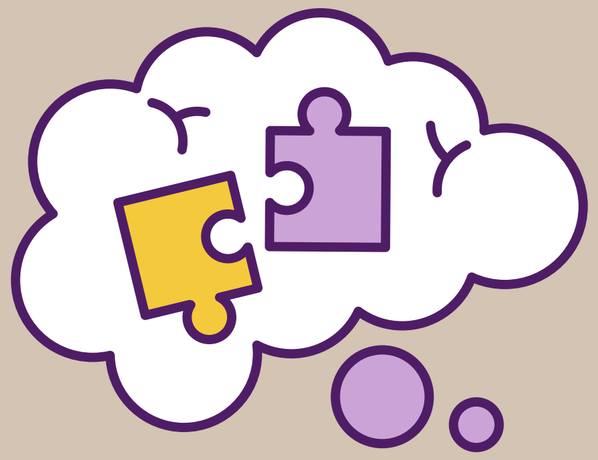
Cluster 2:

Unit Price: Average 30.91 with a standard deviation of 13.23.
Quantity: Average 7.02 with a standard deviation of 1.94.
Total: Average 228.75 with a standard deviation of 114.33.
Rating: Average 7.12 with a standard deviation of 1.71.

Characteristics: Customers in this cluster tend to buy products with low unit prices and large quantities. The total purchase value is medium compared to other clusters.

Strategy: Price-sensitive buyers, but buy in large quantities. Price promotion strategies and product bundling packages can be effective to increase sales in this cluster.

Conclusion



-The company is expected to implement the results of the analysis that I conducted and can carry out the recommendations that I provide
-the company can also apply some attractive offers or discounts when sales decline
-Companies can also see the characteristics, and also the strategies of the 3 clusters
-It is hoped that the company will gain insight from the analysis I conducted to advance the company and increase revenue.

VIEW IN GITHUB

VIEW IN COLAB

Thank you!