Kampus Merdeka

INDONESIA JAYA

KREDENSIAL
MIKRO
MAHASISWA
INDONESIA

MACHINE LEARNING

UNIVERSITAS CIPUTRA

METRODATA ACADEMY

PROJECT REPORT

- CRISP-DM Methodology
- Business Understanding and Data Understanding
- Data Preparation and Modelling
- Evaluation

https://kmmi.kemdikbud.go.id

DAFTAR ISI

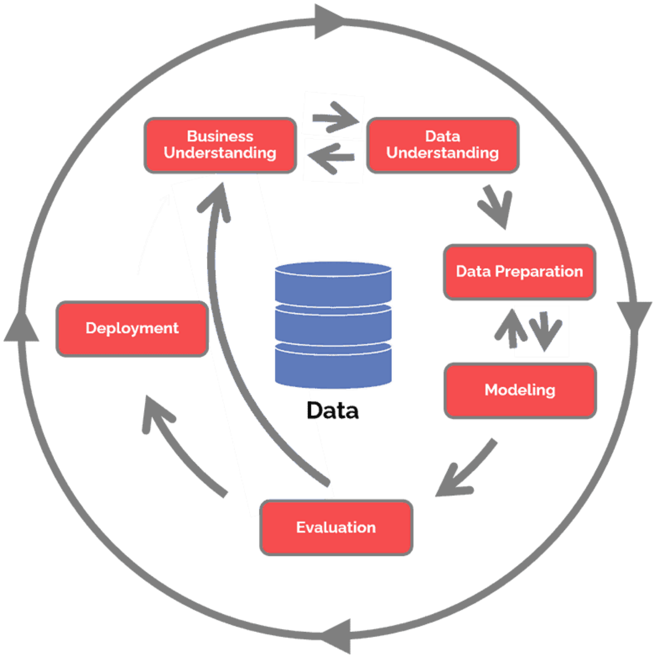
CRISP-DM METHODOLOGY

Cross Industry Standard Process for Data Mining (CRISP-DM) is a six-phase process model that naturally describes the life cycle of data science using Machine Learning. This methodology will help you plan, organize, and implement projects.

CRISP-DM and IBM's Data science methodology begins with a Business Understanding activity which is a process of understanding the problem to be solved. The activity also includes a mapping process between business problems and analytical tasks (appropriate data science tasks).

The next activity is understanding the data (Data Understanding) which includes determining data needs, collecting data and exploring data. In the IBM Methodology, each sub-activity is made into a separate process.

The next step is Data Preparation which is carried out to improve the quality of the data to match the Modeling process that will be carried out next. The quality of the resulting model is evaluated before being deployed into an operational system. The series of activities ended with a feedback and reporting process.



Gambar 1. Siklus Metode CRISP-DM

BUSINESS UNDERSTANDING

In running a business, a company will certainly conduct transactions with various types of individual characteristics that are different and unique. These various individuals can be used as labels by companies regarding their target market. By recognizing the type of market segmentation, a company can adjust their business strategy in various ways, from providing attractive promotions, personalized ads, etc. Therefore, our group wants to create a model that is able to show the characteristics of each market segmentation that exists in a company based on data in the form of transaction history.

To answer the above problems, group 5 used the K-Means Clustering method to group the data. The K-Means method was chosen as the main method because K-Means is a very popular and easy to understand clustering method. The K-Means method is considered suitable for the type of dataset with a very large number of records. Our group also realizes that the marketing segmentation problems faced by various companies can be very different from one another, so the use of K-Means can help these companies because the use of K-Means is easy to understand and easy to adapt. K-Means will be more suitable to be used in making this model when compared to other clustering methods such as Hierarchical Clustering. Our consideration lies in the K-Means process which requires a value of k that represents the desired number of clusters. While in Hierarchical itself, the process continues until only 1 type of cluster is left, and the output given is in the form of a dendrogram. With the scale of this very large dataset, companies will find it difficult to see the dendogram graph and determine the number of clusters they want.

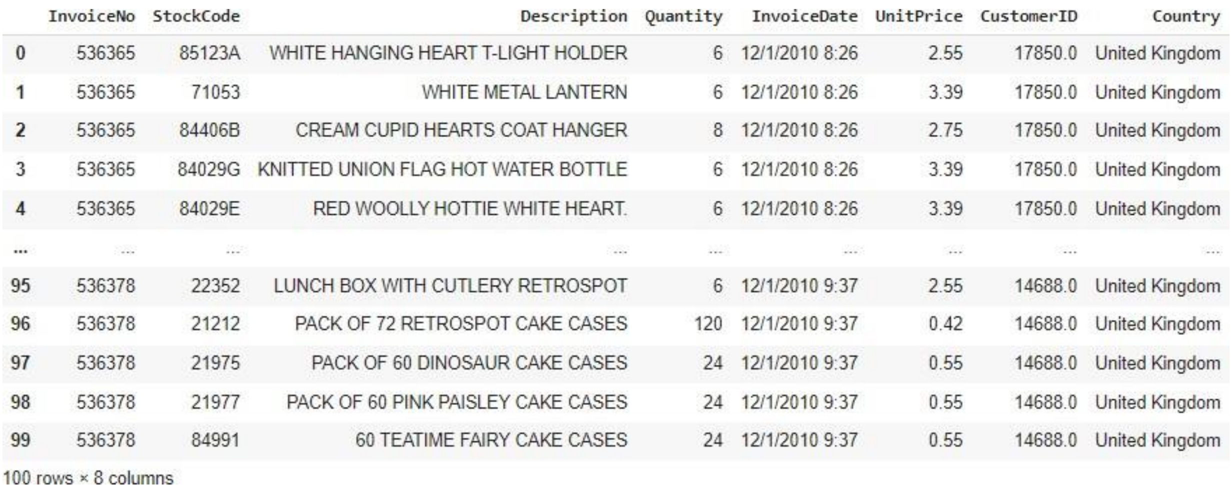
In assessing the feasibility of the model that we will design, of course there is an evaluation process. In this case, we use evaluation methods in the form of the Elbow Method and Silhouette Score. But for now, we will prioritize the use of the Silhouette Score method because in the process itself, the Silhouette Score takes into account the similarity of a point to its own cluster compared to other clusters. While in the Elbow Method, cluster determination is done by looking at the faults of the Elbow Graph in finding the most optimal point. However, the use of these two evaluation methods can be combined to find the number of clusters that best suits the company's needs.

DATA UNDERSTANDING

By observation of the data, we see that the data provided is historical data on sales transactions at a shopping mall. This data stores information in the form of invoice number, item ID, item description, number of items purchased, date of purchase, price per unit, buyer ID, and buyer's country of origin. Our group also found that there was blank data in the description column and customer ID. This indicates that there are items that do not have a description, and there are also transactions where the buyer is not recognized. In the column for the number of items and the price per unit, there are some negative values that should not be.

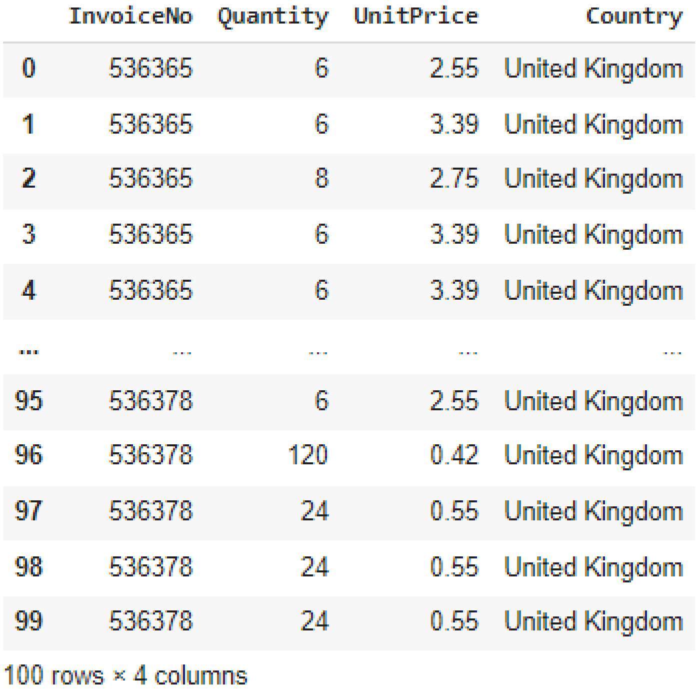
DATA PREPARATION

a model we need to understand the available data. The first step we took was to load the dataset from the csv file into a pandas dataframe, from the given dataset there are 8 data columns which can be seen in Figure 1.A. based on discussion and business understanding, the group decided to delete the Stock Code, Description, Invoice Date, CustomerID columns which can be seen in Figure 1.B.



	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	12/1/2010 8:26	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	12/1/2010 8:26	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	12/1/2010 8:26	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	12/1/2010 8:26	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	12/1/2010 8:26	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
95	536378	22352	LUNCH BOX WITH CUTLERY RETROSPOT	6	12/1/2010 9:37	2.55	14688.0	United Kingdom
96	536378	21212	PACK OF 72 RETROSPOT CAKE CASES	120	12/1/2010 9:37	0.42	14688.0	United Kingdom
97	536378	21975	PACK OF 60 DINOSAUR CAKE CASES	24	12/1/2010 9:37	0.55	14688.0	United Kingdom
98	536378	21977	PACK OF 60 PINK PAISLEY CAKE CASES	24	12/1/2010 9:37	0.55	14688.0	United Kingdom
99	536378	84991	60 TEATIME FAIRY CAKE CASES	24	12/1/2010 9:37	0.55	14688.0	United Kingdom

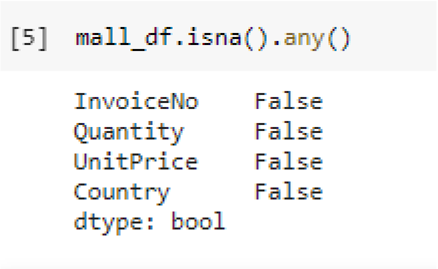
100 rows × 8 columns



	InvoiceNo	Quantity	UnitPrice	Country
0	536365	6	2.55	United Kingdom
1	536365	6	3.39	United Kingdom
2	536365	8	2.75	United Kingdom
3	536365	6	3.39	United Kingdom
4	536365	6	3.39	United Kingdom
...	...	...	...	...
95	536378	6	2.55	United Kingdom
96	536378	120	0.42	United Kingdom
97	536378	24	0.55	United Kingdom
98	536378	24	0.55	United Kingdom
99	536378	24	0.55	United Kingdom

100 rows × 4 columns

After deleting the columns that are not needed in determining the model, we check the null data contained in the dataframe. From the coding results in Figure 2.A, it can be seen that the InvoiceNo, Quantity, Unit Price, and Country columns do not have empty data.

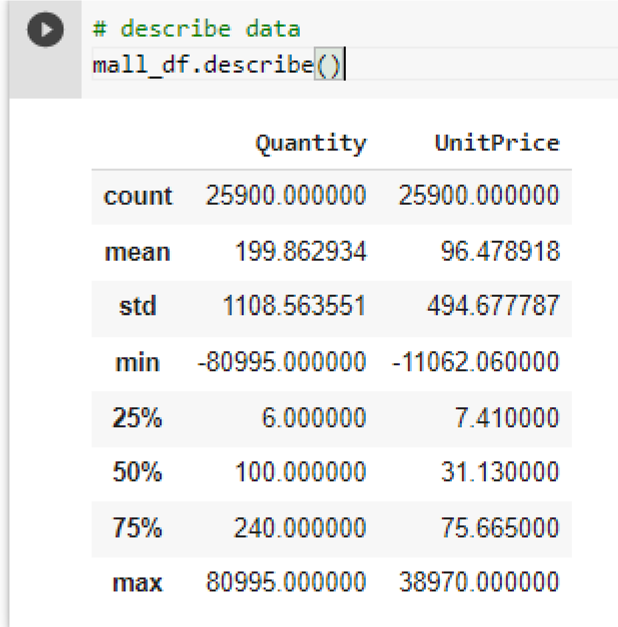


```
[5]  mall_df.isna().any()
```

```
InvoiceNo    False
Quantity     False
UnitPrice    False
Country      False
dtype: bool
```

DATA TRANSFORMATION

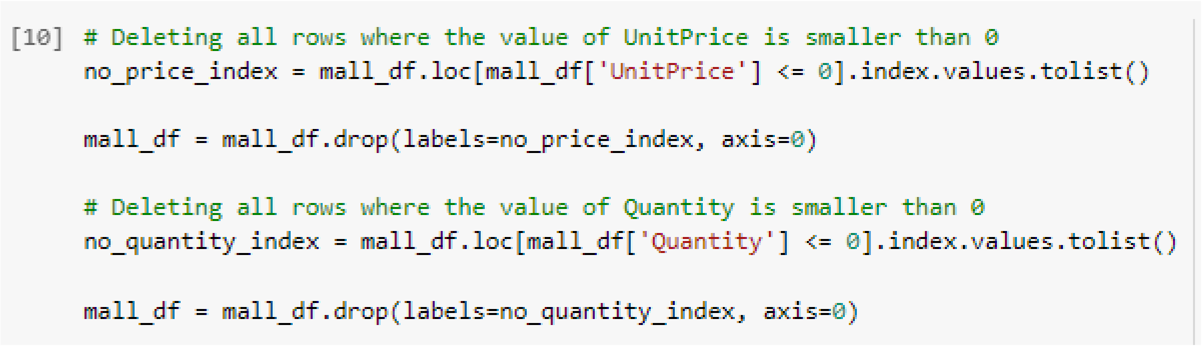
The following is the result of the data that has been described, it can be seen that the minimum value for Quantity and Unit Price has a minus value



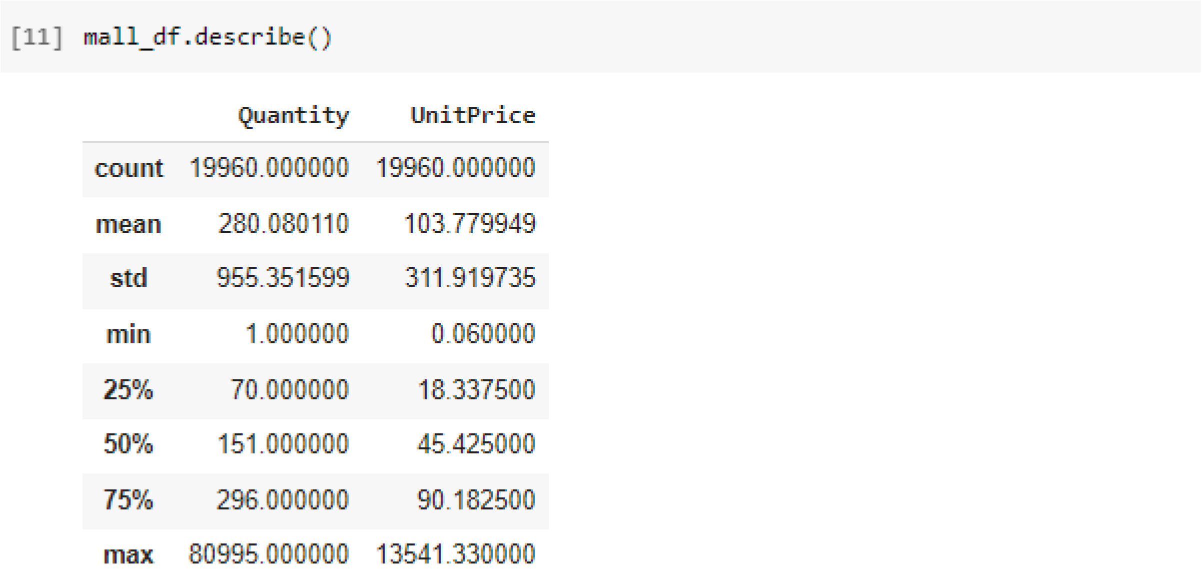
```
# describe data
mall_df.describe()
```

	Quantity	UnitPrice
count	25900.000000	25900.000000
mean	199.862934	96.478918
std	1108.563551	494.677787
min	-80995.000000	-11062.060000
25%	6.000000	7.410000
50%	100.000000	31.130000
75%	240.000000	75.665000
max	80995.000000	38970.000000

After knowing that there is a minus value, the value quantity or unit price below 0 will be deleted. As a result, after being deleted, the data is again described to check whether the data is still below 0 or not.



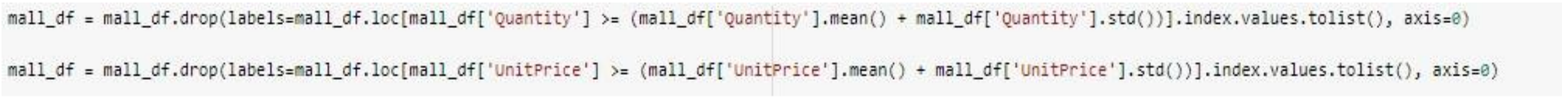
```
[10] # Deleting all rows where the value of UnitPrice is smaller than 0
     no_price_index = mall_df.loc[mall_df['UnitPrice'] <= 0].index.values.tolist()

     mall_df = mall_df.drop(labels=no_price_index, axis=0)

     # Deleting all rows where the value of Quantity is smaller than 0
     no_quantity_index = mall_df.loc[mall_df['Quantity'] <= 0].index.values.tolist()

     mall_df = mall_df.drop(labels=no_quantity_index, axis=0)
```



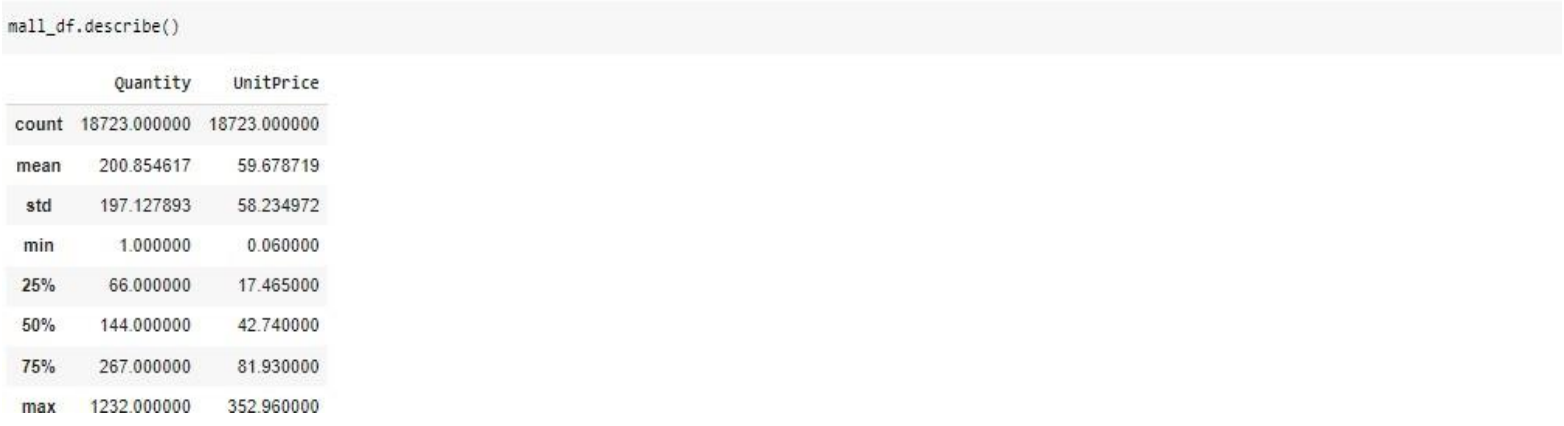
```
[11] mall_df.describe()
```

	Quantity	UnitPrice
count	19960.000000	19960.000000
mean	280.080110	103.779949
std	955.351599	311.919735
min	1.000000	0.060000
25%	70.000000	18.337500
50%	151.000000	45.425000
75%	296.000000	90.182500
max	80995.000000	13541.330000

We use the mean value added to the standard deviation to determine the value of the row that is classified as very large (outliers). Then if there is a value in the quantity or unit price column that is greater than the mean value added with a standard deviation, then we will delete that row.



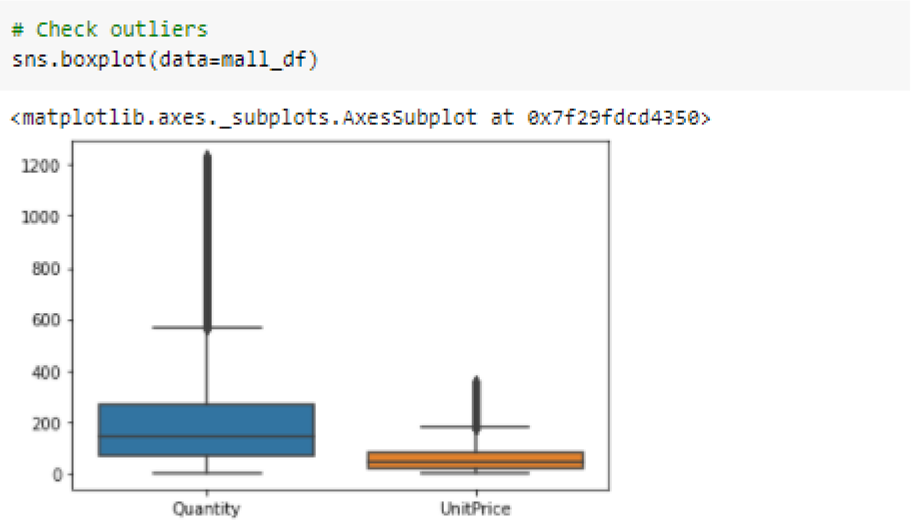
```
mall_df = mall_df.drop(labels=mall_df.loc[mall_df['Quantity'] >= (mall_df['Quantity'].mean() + mall_df['Quantity'].std())].index.values.tolist(), axis=0)

mall_df = mall_df.drop(labels=mall_df.loc[mall_df['UnitPrice'] >= (mall_df['UnitPrice'].mean() + mall_df['UnitPrice'].std())].index.values.tolist(), axis=0)
```



```
mall_df.describe()
```

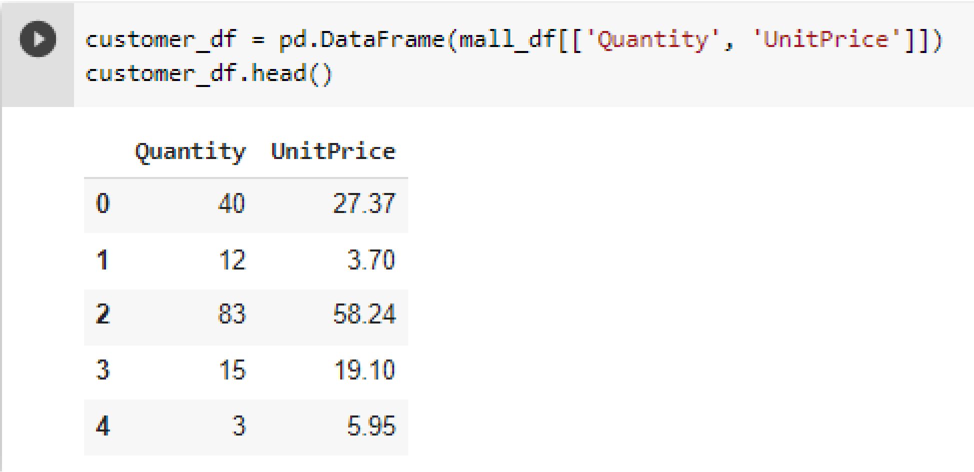
	Quantity	UnitPrice
count	18723.000000	18723.000000
mean	200.854617	59.678719
std	197.127893	58.234972
min	1.000000	0.060000
25%	66.000000	17.465000
50%	144.000000	42.740000
75%	267.000000	81.930000
max	1232.000000	352.960000

In this process we return the list of quantity values and Unit Price values that we have dropped earlier.



```
# Check outliers
sns.boxplot(data=mall_df)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f29fdcd4350>
```

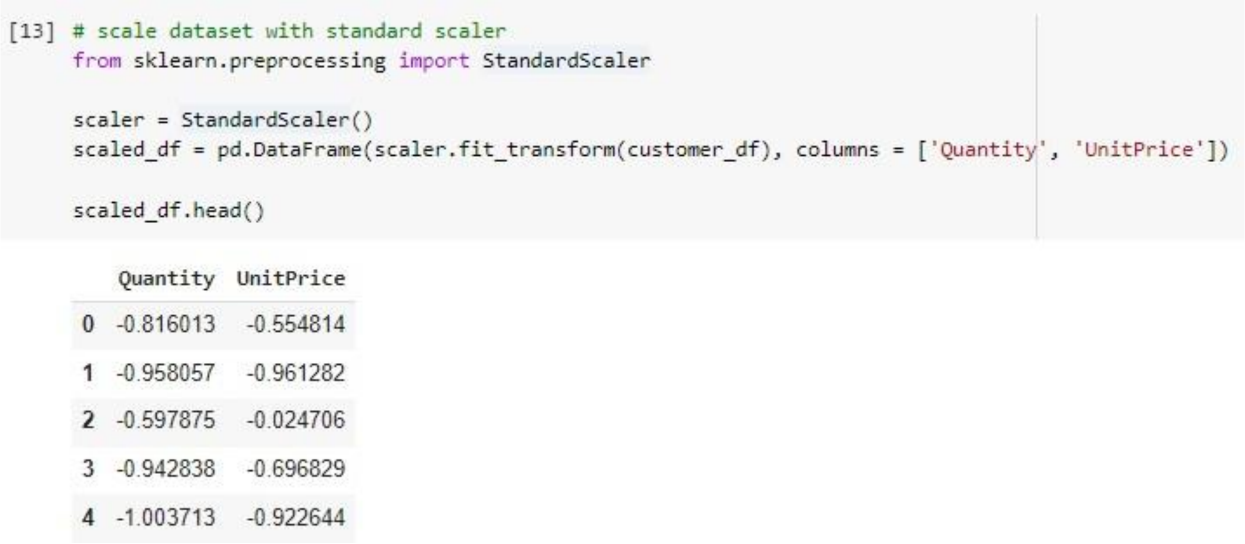


```
customer_df = pd.DataFrame(mall_df[['Quantity', 'UnitPrice']])
customer_df.head()
```

```
   Quantity  UnitPrice
0        40      27.37
1        12       3.70
2        83      58.24
3        15      19.10
4         3       5.95
```

FEATURE ENGINEERING

Gambar 1.C

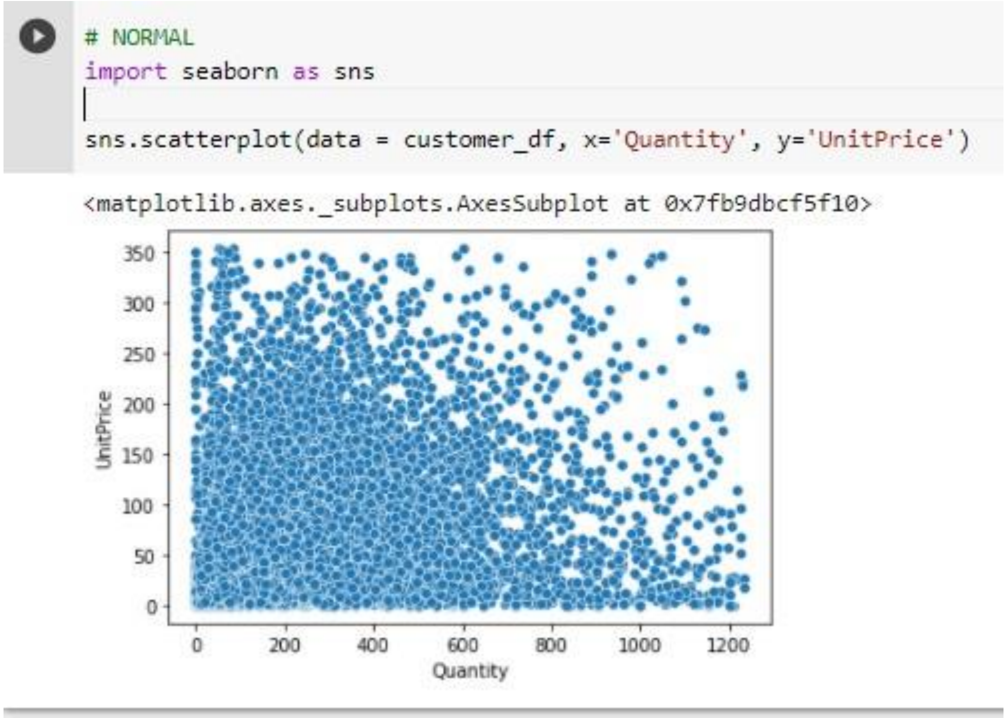


```
[13] # scale dataset with standard scaler
     from sklearn.preprocessing import StandardScaler

     scaler = StandardScaler()
     scaled_df = pd.DataFrame(scaler.fit_transform(customer_df), columns = ['Quantity', 'UnitPrice'])

     scaled_df.head()
```

```
   Quantity   UnitPrice
0  -0.816013  -0.554814
1  -0.958057  -0.961282
2  -0.597875  -0.024706
3  -0.942838  -0.696829
4  -1.003713  -0.922644
```

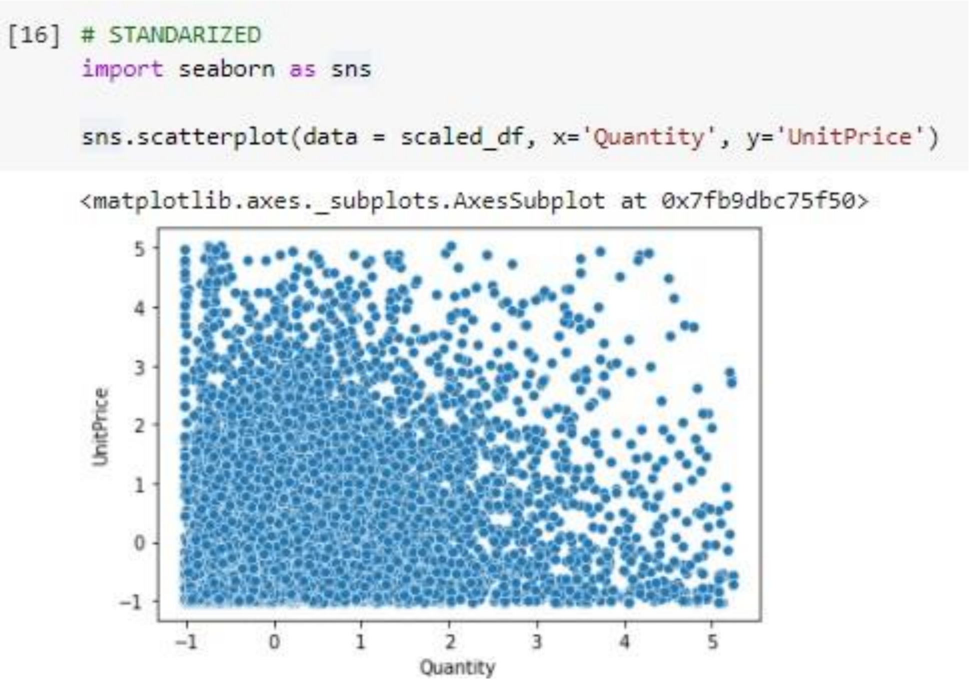
Here we have created a new data frame named customer_df which consists of Quantity and Unit Price data. With the help of boxplots we can visualize the data. Here we can see for ourselves how the two data when visualized.Scaling dataset *customer_df* using the Standard Scaler

After we have scaled the *customer_df* we check the starting position for our sample visually using *a scatter plot*.examination *scatter plot* is; *customer_df* and scaled_df(data that has been scaled from *customer_df*)



```
# NORMAL
import seaborn as sns

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

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fb9dbcf5f10>
```

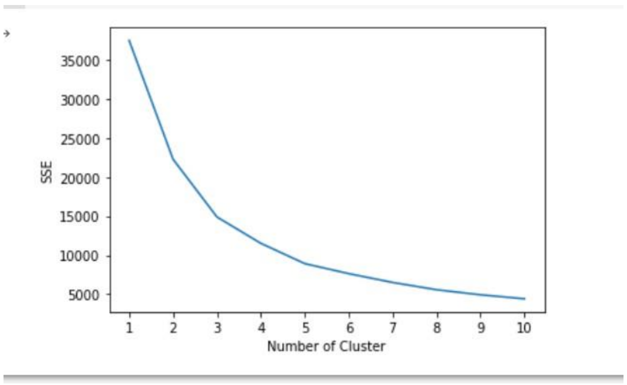


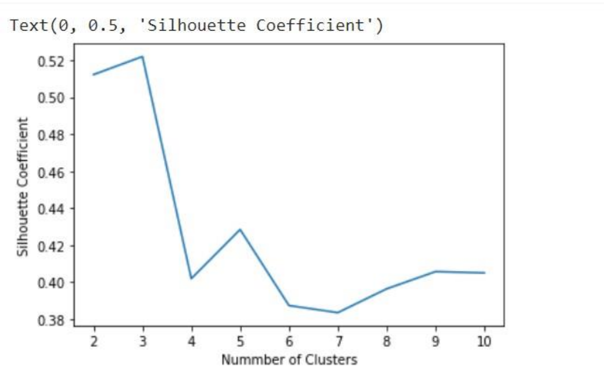
```
[16] # STANDARIZED
     import seaborn as sns

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

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fb9dbc75f50>
```

MODELLING

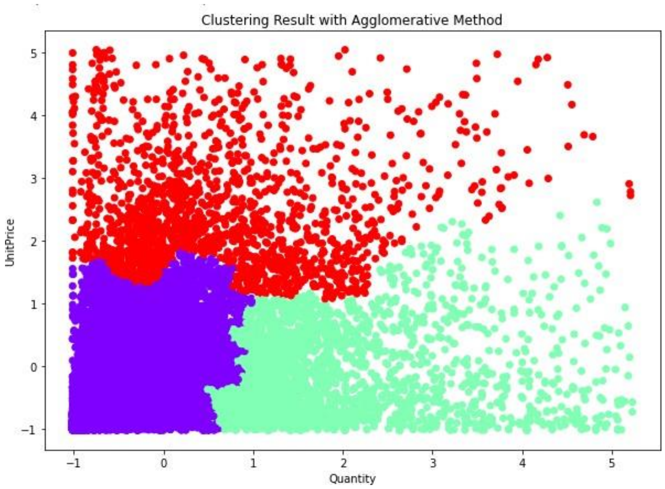
In determining the group evaluation method, we used 2 methods to determine the K Means clustering model, namely the Elbow Method and the Silhouette Score. The Elbow Method calculation uses the average amount of distortion or distance to the center of the graph, while the Silhouette Score calculation uses the average of the distance from one cluster to another nearby cluster. Based on the graph obtained by calculations using the Elbow Method *(Figure 3.A)* the highest distortion value is in the number of 3 clusters and for the Silhouette Score*(Figure 3.B)* the highest value is obtained from the number of 3 clusters.





```
Text(0, 0.5, 'Silhouette Coefficient')
```

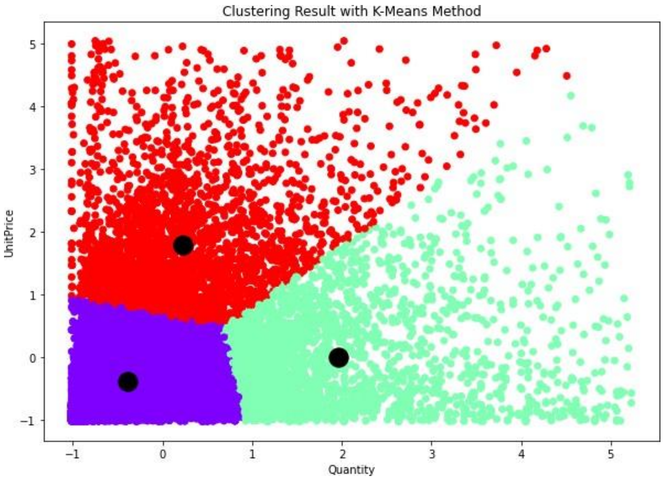
Our group creates 2 types of machine learning models using different methods. In the first model, we used a hierarchical agglomerative clustering method. The parameters we use in making this agglomerative clustering model are the Euclidean metric, criterion ward, and the number of n clusters 3. Then we use the model to divide the clusters, and here are the results we get:



We use a combination of euclidean and ward because criterion ward is indeed only according to euclidean. When we tried various other methods, the clustering results were also not as good as this Euclidean and Ward. Our consideration of using the number of clusters of 3 is because based on the SSE graph, the knee value or the most optimum is located in the number of clusters of 3.

Then our second model uses the K-means Algorithm method. The parameters we use are random init, the number of n clusters is 3, the maximum iteration is 300, and the random state is 42. Based on these parameters we get a clustering graph as shown below:

Same as before, we set our cluster number to be 3 clusters. Because our SSE and Silhouette Score values are at the optimum point when the cluster value is 3. We also use a random state of 42 to ensure that every time we reshape the model, the results will always be the same.



EVALUATION

Based on the Machine Learning model that our group created, the mall manager can see that there are 3 groups of customers classified based on the number of items purchased and the number of items purchased. the first group is customers who buy goods with relatively low quantity and price of goods, the second group is customers who buy goods with high quantity but with low price of goods (Wholesale goods category), the third group is customers who buy goods with low quantity of goods But the relatively high price of goods (luxury goods category) from this data is expected that mall managers can develop sales strategies that are in accordance with customer targets in order to get higher profits.

The difficulty of our group project clustering is that the preprocessing dataset of this dataset includes some data that our group considers irrelevant, for example, data on the quantity and price of goods that are negative, outlier data is also found which is considered irrelevant, for example there is data on customers who buy goods with quantity 80995pcs. Because of these irrelevant data, our group had difficulties in sorting out the data to be used in the model.

PROFILES

Hello my name is M HAIKAL FEBRIAN P I was born in (meulaboh 05 february 2002), I study at Telkom University and I majored in computer engineering, the motto of life is 'the important thing is to stay calm and try your best and surrender to Gusti Allah ‘.

.

.

REFERENCES

https://www.geeksforgeeks.org/difference-between-k-means-and-hierarchical-clustering/#:~:text=Agglomerative%20methods%20begin%20with%20%27n,only%20one%20cluster%20is%20obtained.&text=In%20K%20Means%20cl ustering%2C%20since,algorithm%20many%20times%20may%20differ

https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html