Statistical Learning: Customer Segments

Thi Thuy Le - S0564314

February 27, 2022

Contents

1 Introduction

1.1 What is Customer Segments?

Customer segmentation is the process of dividing customers into groups based on common characteristics so companies can market to each group effectively and appropriately.

A customer segmentation always has a concrete goal, e.g. to better understand customer groups, to create selections for marketing campaigns or to estimate potential. The customer segment is created on the basis of one- or multi-dimensional information that is relevant to the achievement of the goal.

Figure 1: Customer Segments

1.2 Why is customer segmentation important?

- Firstly, identify the right potential customers: Customer segmentation helps the company to identify and understand the characteristics of the target customer, thereby effectively supporting marketing strategies to the target audience. Customers have appropriate age groups, geographical locations, gender, purchasing habits or interests.

- Second, discover new opportunities in the market: It can be said that customer segmentation is also the basis for marketers to identify and evaluate the market at different times, thereby monitoring progress of the market, predicting the next possible changes in the market in order to anticipate the needs of the market.

- Third, create specific and relevant messages: By segmenting customers, you can clearly understand what each segment needs from you and from there can create messages that directly address their problems. encountered.

- Fourth, improve brand loyalty: When customers feel that your products and services are right for them at the right time in their lives, they are more likely to stick with your brand. and recommend it to others.

- Fifth, make a difference compared to competitors: Companiescan give specific messages to each customer segment, making the brand more prominent and imprinted in the hearts of customers.

2 Data analysis

Sources we use: https://www.kaggle.com/shwetabh123/mall-customers
Github: https://github.com/ThiThuyLe/Project-Statistical-Learning

2.1 Data sets

CustomerID: Unique ID assigned to the customer.
Gender: Gender of the customer.
Age: Age of the customer.
Annual Income (k$): Annual Income of the customer.
Spending Score: Score assigned by the mall based on customer behavior and spending nature.

First I will include all the important and required Python libraries Import modules and existing data from Excel into Python.

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
import sklearn.metrics as sm
from collections import Counter
from sklearn.cluster import KMeans
from pandas import DataFrame
from sklearn.cluster import AgglomerativeClustering
```

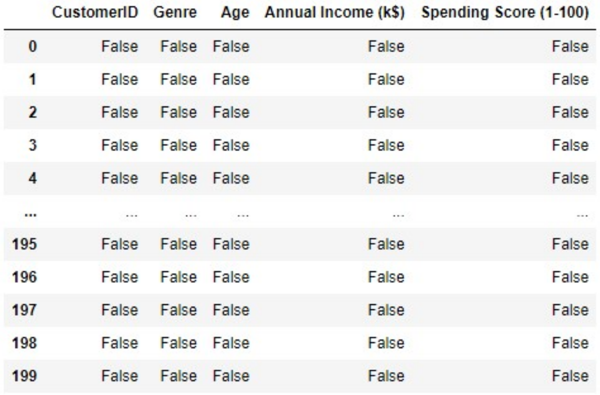
Then we give a quick reference. And this is what the Dataframe I need to create looks like.

```
df.head()
```

CustomerID	Genre	Age	Annual income (k$)	Spending score (1-100)
0001	male	19	15	39
0002	male	21	15	81
0003	female	20	16	6
0004	female	23	16	77
0005	female	31	17	40

2.2 Check MissingValues NaN

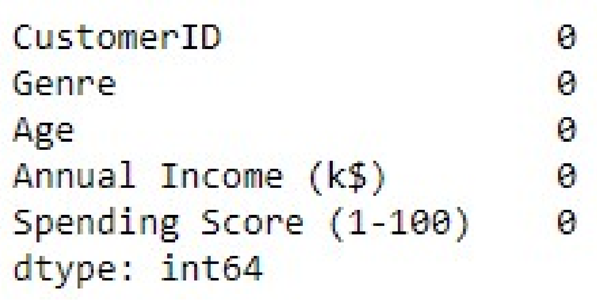
In order for all data to run smoothly, I have to check if any data is missing.



	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
...	...	...	...	...	...
195	False	False	False	False	False
196	False	False	False	False	False
197	False	False	False	False	False
198	False	False	False	False	False
199	False	False	False	False	False

Figure 2: Check MissingValues

```
df.isnull().sum()
```



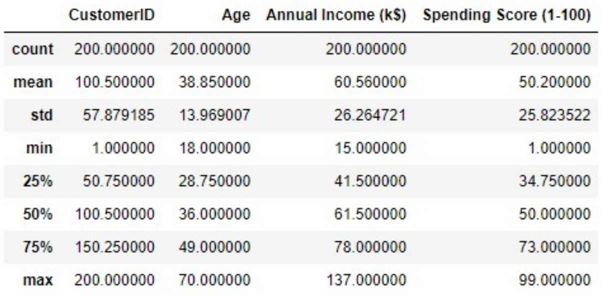
```
CustomerID                0
Genre                     0
Age                       0
Annual Income (k$)        0
Spending Score (1-100)    0
dtype: int64
```

Figure 3: Sum of the MissingValues

Here it can be seen that there are no missing values.

2.3 Mean, Min, Max, Median, Standard Deviation of data

```
df.describe()
```



	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
count	200.000000	200.000000	200.000000	200.000000
mean	100.500000	38.850000	60.560000	50.200000
std	57.879185	13.969007	26.264721	25.823522
min	1.000000	18.000000	15.000000	1.000000
25%	50.750000	28.750000	41.500000	34.750000
50%	100.500000	36.000000	61.500000	50.000000
75%	150.250000	49.000000	78.000000	73.000000
max	200.000000	70.000000	137.000000	99.000000

Figure 4: Mean, Min, Max, Median, Standard Deviation of data

Standard deviation of Annual Income and Spending Score are higher than Age. They can contribute to cluster formation.

3 Visualizing data

3.1 Annual Income

```
fig = plt.figure(figsize = (20, 10))
plt.subplot(1, 2, 1)
sns.set(style = 'whitegrid')
sns.distplot(df['Annual Income (k$)'])
plt.title('Distribution of Annual Income', fontsize = 22)
plt.xlabel('Range of Annual Income')
plt.ylabel('Count of Annual Income')

plt.show()
```

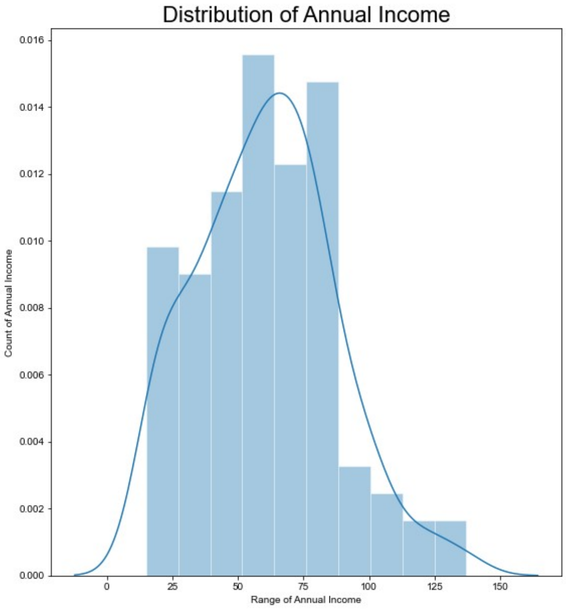


Figure 5: Distribution of Annual Income

While in histogram mode, it is also possible to add a KDE (Kernal Density Estimation) curve.

This Annual Income parameter has a maximum value of 137 and a minimum value of 15. The maximum distribution is at the value 70.

The blue lines represent the normal distribution. The distribution of Annual Income is right-skewed and normally distributed.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)$$

We have a more detailed look at this chart (Figure 6).

```
fig = plt.figure(figsize = (20, 10))
sns.countplot(df['Annual Income (k$)'], palette = 'rainbow')
plt.title('Distribtuion of Annual Income', fontsize = 22)

plt.show()
```

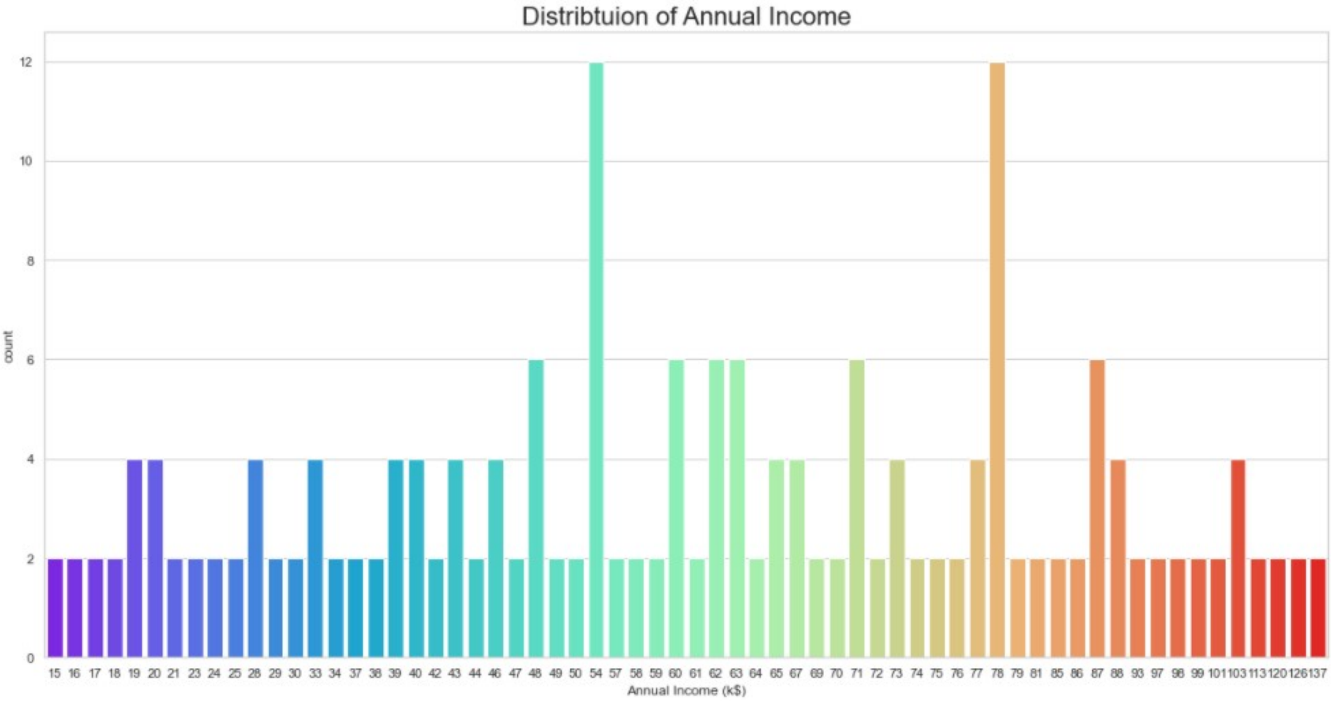


Figure 6: Distribution of Annual Income

There are many people earning at 54(k$) and 78(k$).

3.2 Age

```
fig = plt.figure(figsize = (20, 10))
plt.subplot(1, 2, 2)
sns.set(style = 'whitegrid')
sns.distplot(df['Age'], color = 'green')
plt.title('Distribution of Age', fontsize = 22)
plt.xlabel('Range of Age')
plt.ylabel('Count of Age')

plt.show()
```

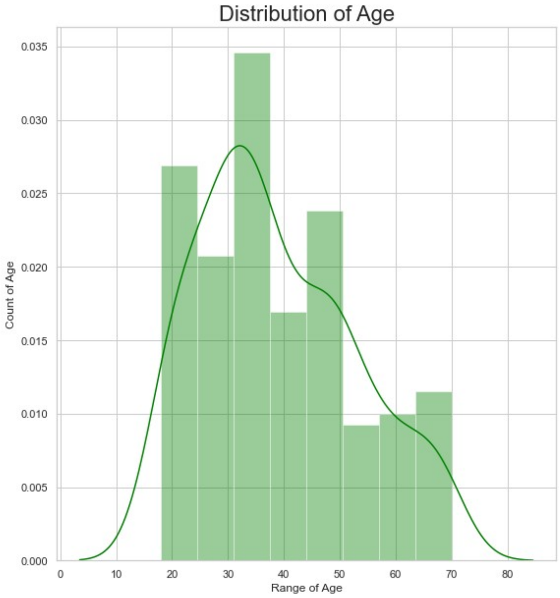


Figure 7: Distribution of Age

This Age parameter has a maximum value of 70 and a minimum value of 18. The maximum distribution is between 30 and 35. Through this we can also see, the majority of customers are young people.

We have a more detailed look at this chart (Figure 8).

```
fig = plt.figure(figsize = (20, 10))
sns.countplot(df['Age'], palette = 'rainbow')
plt.title('Distribtuion of Age', fontsize = 22)

plt.show()
```

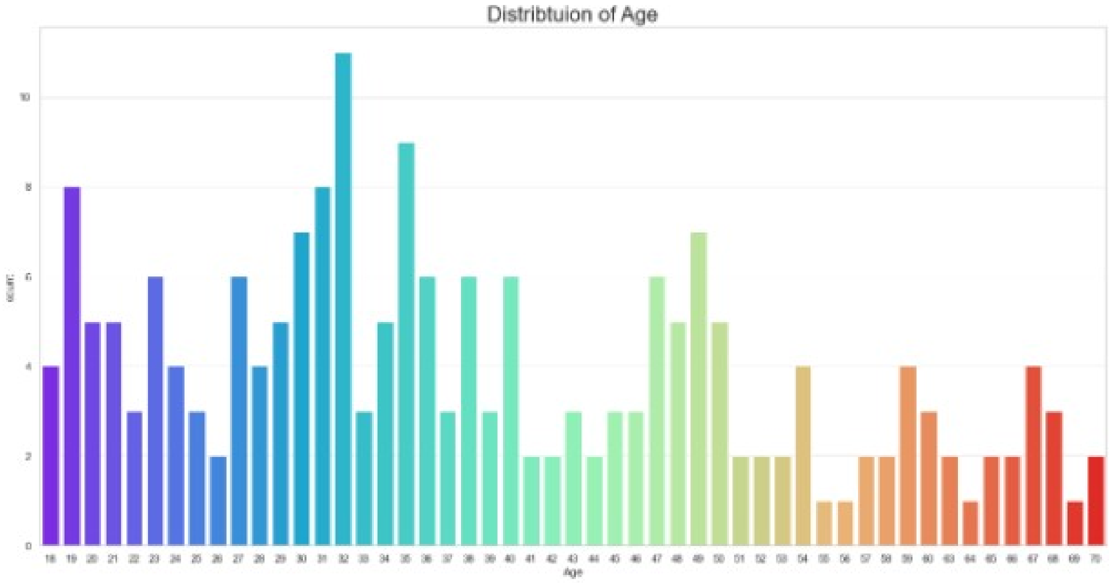


Figure 8: Distribution of Age

3.3 Spending Score (1-100)

```
fig = plt.figure(figsize = (20, 10))
plt.subplot(1, 2, 2)
sns.set(style = 'whitegrid')
sns.distplot(df['Spending Score (1-100)'], color = 'brown')
plt.title('Distribution of Spending Score (1-100)', fontsize = 22)
plt.xlabel('Range of Spending Score (1-100)')
plt.ylabel('Count of Spending Score (1-100)')

plt.show()
```

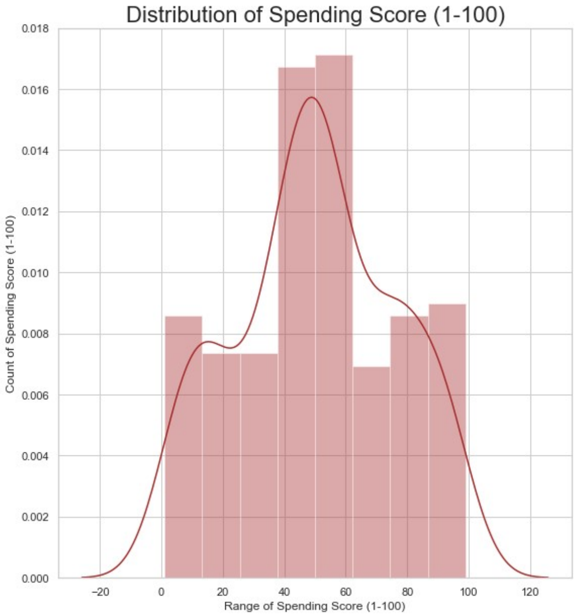


Figure 9: Distribution of Spending Score

This Spending Score parameter has a maximum value of 99 and a minimum value of 1. The maximum distribution is between 40 and 50 and average is 50,20.

If we take a deep dive into the features, it is observed that spending score has 3 peaks(0-20,40-60,80-100)

We have a more detailed look at this chart (Figure 10).

```
fig = plt.figure(figsize = (20, 10))
sns.countplot(df['Spending Score (1-100)'], palette = 'rainbow')
plt.title('Distribtuion of Spending Score (1-100)', fontsize = 22)

plt.show()
```

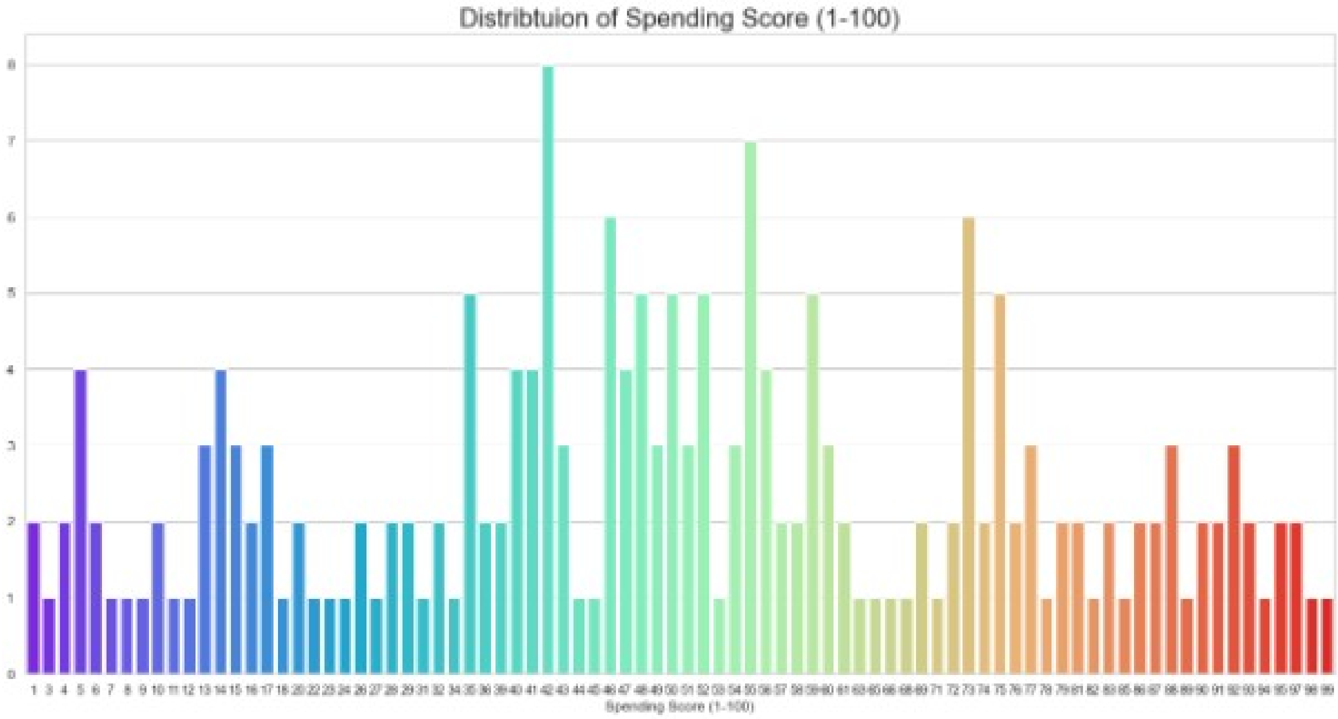


Figure 10: Distribution of Spending Score

Most people have an average score. Those with low or high scores are very few.

3.4 Gender

```
labels= ['Female', 'Male']
size= df['Genre'].value_counts()
plt.rcParams['figure.figsize'] = (6,6)
plt.pie(size, labels=labels,autopct='%1.1f%%')
plt.title('Distribution of Gender', fontsize = 20)
plt.legend()
plt.show()
```

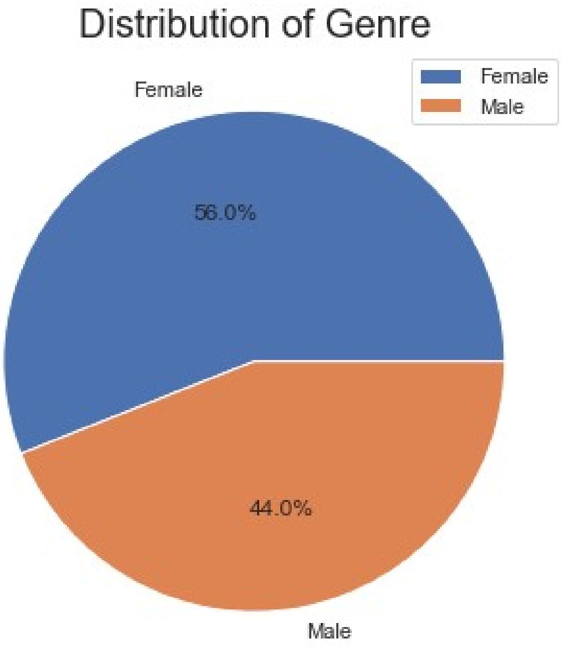


Figure 11: Distribution of Gender Score

We divide the Gender into 2 groups. Women make up the majority with 56% and men with 44%. In this pie chart we can easily see that women have more passion for shopping than men.

Compare between Gender and parameters

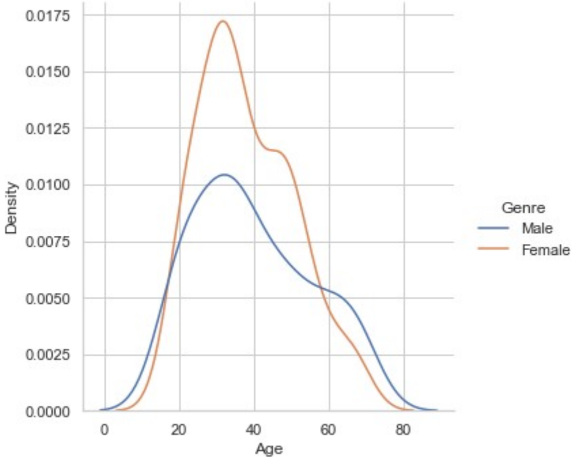


Figure 12: Compare Gender vs Age

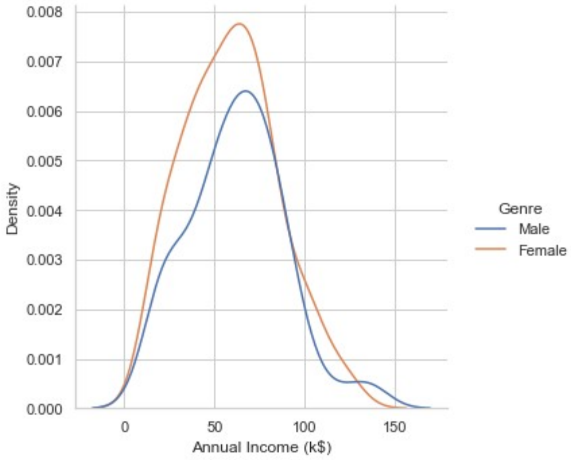


Figure 13: Compare Gender vs Annual Income

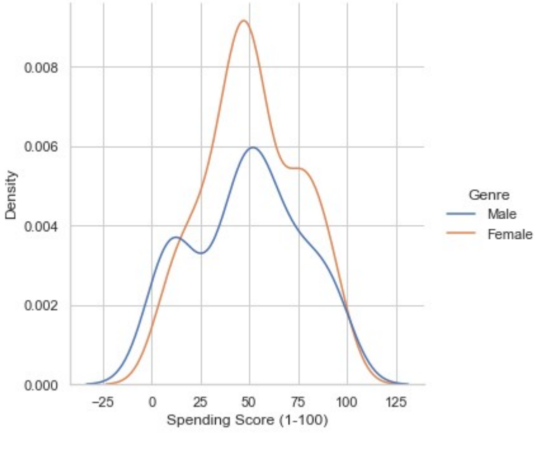


Figure 14: Compare Gender vs Spending Score

Through 3 charts (Figur 12, Figur 13, Figur 14), we can see, the percentage of women is higher than men in all aspects such as Annual Income, Age and Spending Score.

Compare between Annual Income and Age and Spending Score

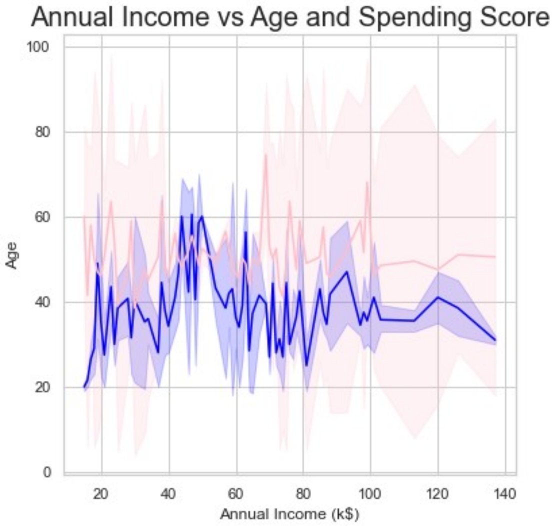


Figure 15: Compare between Annual Income and Age and Spending Score

The above Plot Between Annual Income and Age represented by a blue color line, and a plot between Annual Income and the Spending Score represented by a pink color. shows how Age and Spending Score varies with Annual Income.

3.5 Correlations of the variables

First, we want to get an overview of the correlations of the dataset's features. From the plot data it can be seen that between some features from the data is sightly correlated.

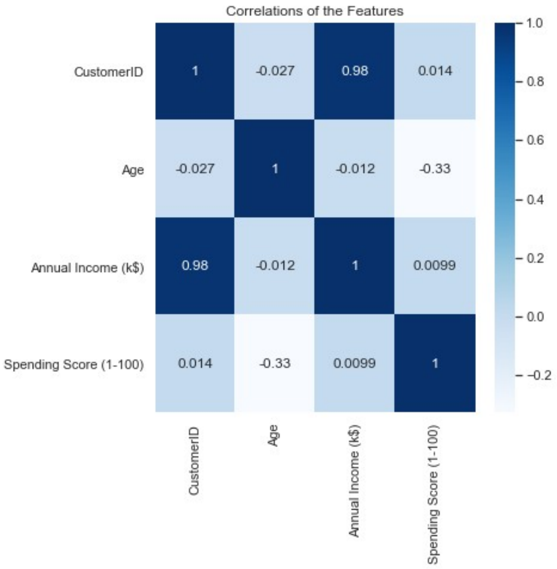


Figure 16: Correlations of the variables

4 Clustering Methode

4.1 K-Means

Definition: K-Means algorithm is a calculation method that can be used for grouping objects, the so-called cluster analysis.

4.2 Elbow Method

Definition: In cluster analysis, the elbow method is a heuristic used in determining the number of clusters in a data set. The method consists of plotting the explained variation as a function of the number of clusters, and picking the elbow of the curve as the number of clusters to use. The same method can be used to choose the number of parameters in other data-driven models, such as the number of principal components to describe a data set.

We will use the elbow method to check the optimal number of clusters.

```
inertias = []
for i in range(1, 11):
    km = KMeans(n_clusters=i).fit(feature)
    inertias.append(km.inertia_)

plt.plot(range(1, 11), inertias, 'bx-')
plt.xlabel('Values of K')
plt.ylabel('Inertia')
plt.title('The Elbow Method using Inertia')
plt.show()
```

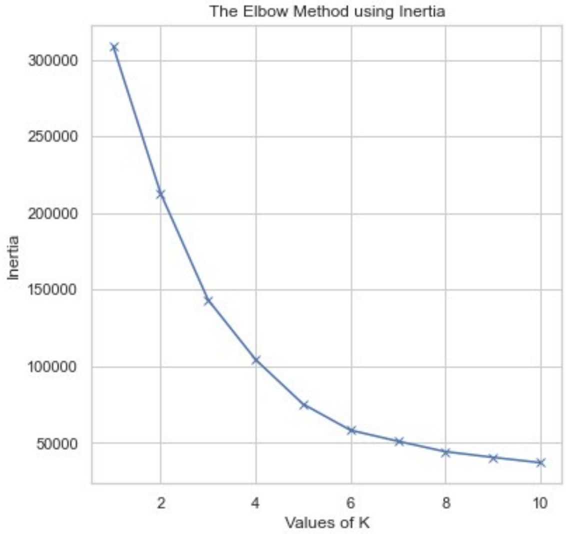


Figure 17: Elbow Method

From the diagram we can see that 5 is the optimal number of K.

We divide customer segments into 5 groups.

```
km = KMeans(n_clusters=5).fit(feature)
y_km = km.fit_predict(feature)
n_cluster, km_count = np.unique(y_km, return_counts=True)
plt.bar(n_cluster, km_count)
plt.ylabel('No of customer')
plt.xlabel('Clustering')
plt.title('Customer segmentation by 5 groups')
```

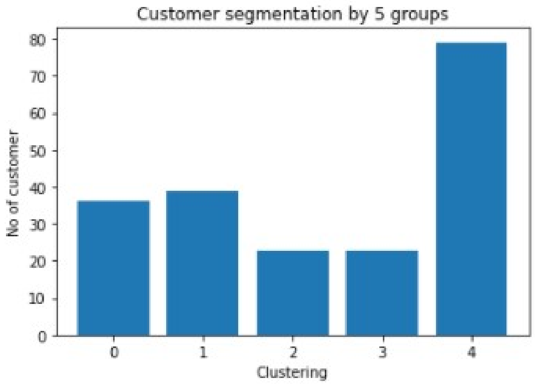


Figure 18: Customer Segmentation by 5 groups

Through the chart (Figur 19), we can analyze the following:

- **Cluster 1:** The customers have average annual income as well as average spending score.
- **Cluster 2:**The customers have lower spending score but have a high annual income.
- **Cluster 3:**The customers have lower annual income and lower spending score.
- **Cluster 4:** The customers with lower annual income but higher spending score.
- **Cluster 5:** The customers have both higher annual income and higher spending score

```
plt.scatter(X[y_kmeans == 0, 0], X[y_kmeans == 0, 1], s = 100, c = 'orange', label =
    'Cluster 1')
plt.scatter(X[y_kmeans == 1, 0], X[y_kmeans == 1, 1], s = 100, c = 'brown', label =
    'Cluster 2')
plt.scatter(X[y_kmeans == 2, 0], X[y_kmeans == 2, 1], s = 100, c = 'red', label =
    'Cluster 3')
plt.scatter(X[y_kmeans == 3, 0], X[y_kmeans == 3, 1], s = 100, c = 'black', label =
    'Cluster 4')
plt.scatter(X[y_kmeans == 4, 0], X[y_kmeans == 4, 1], s = 100, c = 'lightpink', label =
    'Cluster 5')
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s = 300, c =
    'yellow', label = 'Centroids')
plt.title('Clusters of customers')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```

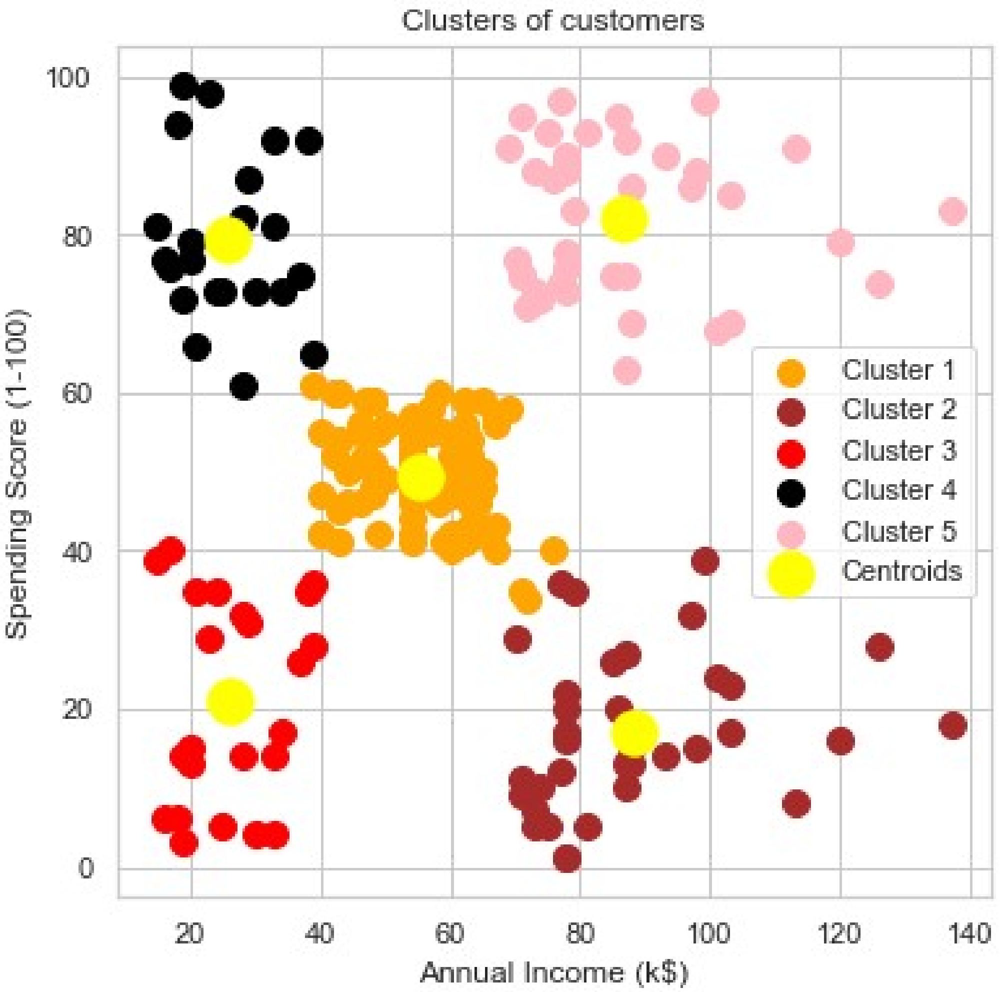


Figure 19: Clusters of customers

5 Summary

K-means clustering is a type of unsupervised learning, it is usually used when we don't know their groups/categories. The algorithm assign each data point to one of K-groups based on the features similarity.

It is useful to find groups which have not been explicitly labeled in the data. This can be used to confirm business assumptions about what types of groups exist or to identify unknown groups in complex data sets.

6 Source

1. https://www.kaggle.com/shwetabh123/mall-customers
2. Lecture script (Dr. Alla Petukhina)
3. https://www.kaggle.com/shwetabh123/mall-customers/code
4. https://www.shopify.com/encyclopedia/customer-segmentation
5. https://datasolut.com/kundensegmentierung/
6. https://hedima.vn/journey/tai-sao-cac-cong-ty-nen-phan-khuc-khach-hang/
7. https://translate.google.com
8. https://de.wikipedia.org/wiki/Cluster(*Datenanalyse*)
9. https://de.wikipedia.org/wiki/K-Means-Algorithmus
10. https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/
11. https://www.youtube.com/watch?v=r-uOLxNrNk8t=1772s
12. Bild https://datasolut.com/kundensegmentierung/

Declaration of Authorship

We hereby confirm that we have authored this Seminar paper independently and without use of others than the indicated sources. All passages (and codes) which are literally or in general matter taken out of publications or other sources are marked as such.

Berlin, 27.02.2022

Thi Thuy Le