Unsupervised Machine Learning: Clustering Project

Paulina Kossowska
September 2021

“One of the main objectives of this course is to help you gain hands-on experience in communicating insightful and impactful findings to stakeholders. In this project you will use the tools and techniques you learned throughout this course to train a few unsupervised machine learning algorithms on a data set that you feel passionate about, and communicate insights you found from your modeling exercise.”

Understanding the Data

Mall_Customers.csv

I have downloaded a fuel consumption dataset, **Mall_Customers.csv** from Kaggle.

Link

The columns in csv file contains information about:

CustomerID

Age

Gender

Annual Income

Spending Score

Goals of this analysis

There are two main goals for this analysis:

1. **Performed Exploratory Data Analysis which will help me understand with what type of data I work on.**
2. **Performed customer segmentation based on this data which will allow a marketing team prepare suited strategy plan for those customers.**

1. EDA:
    - 1.1. Understanding the Data
    - 1.2. Data Visualization
    - 1.3. Skewness and feature scaling

2. Clustering Modeling:
    - 2.1. 1st scenario
        - 2.1.1. KMeans
        - 2.1.2. Hierarchical Clustering
        - 2.1.3. DBSCAN
        - 2.1.4. Mean Shift
    - 2.2. 2nd scenario
        - 2.2.1. KMeans
        - 2.2.2. Hierarchical Clustering

Exploratory Data Analysis

Data Exploration

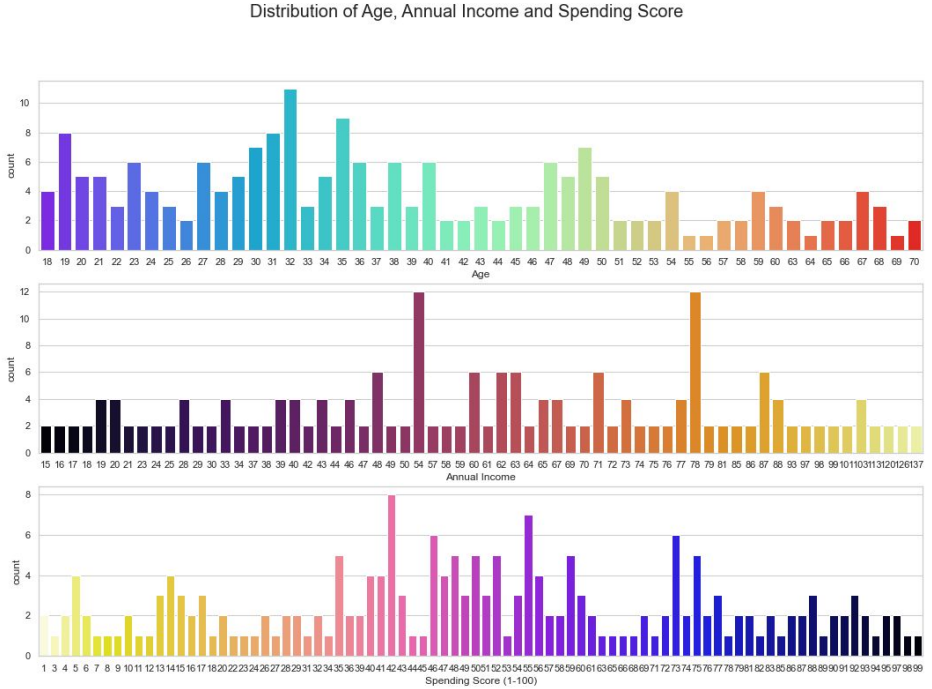
Using shape(), describe(), dtypes(), isnull() methods from pandas library I found out that in my dataset were 200 rows and 5 columns.

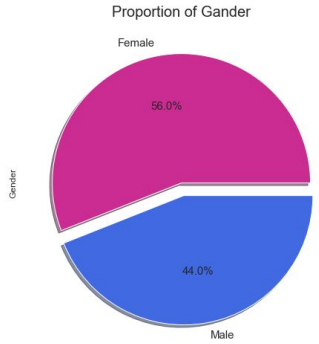
4 of them was an int64 columns and the one was object columns. I also learnt that there were no missing values in my dataframe so I didn't had to handle with this problem in this project.

Data Visualization

As my dataset is not so big, I decided go to visualization section. First I started from gender distribution at my dataset and also looked at distribution of variables. The next couple of slides will shows the results of my work.

Data Visualization





Data Visualization

- Pie chart explain distribution of gender in dataset. As we can see the female lead and got 56% share in gender proportion. It could be a prove of hypothesis that the woman's more often are shopping and visit shopping centres
- Visitors in the Mall are in range 18 - 70 years. The most frequent group in age range lies between 27 - 38 years. People at 32 years are the most frequent group at the Mall. Customers in olders age range start from 55+ are less frequent than others group. For age 55, 56, 64 and 69 we observed the smallest frequent values
- Again, at Annual Income distribution we can see in more details how income range looks like. The dataset contains imaginary informations about annual income. The ranges lies between 15 - 137 dollars. 54 and 78 are the most frequent values
- Spending Score distribution chart is very important. By looking at it we can get some intuition about spending score for customers who visited the Mall. On general I can conclude the the most results lies in range between 35 - 60. The range for this distribution in wide from 1 to 99, so conclude that there are a variety of customers who visited shopping centres and had different requirements

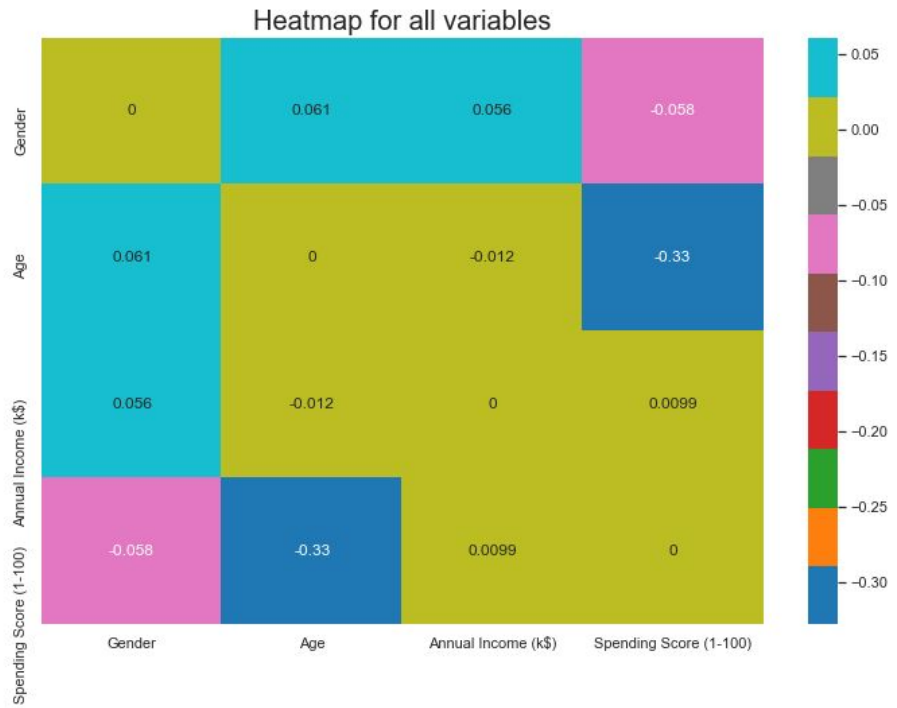
Data Visualization

Then, to perform some viz in more details I first created correlation matrix and heatmap. This help me to identify how variables were correlated. Before I did it, I needed to change Gender column in some binary format, so I used LabelBinarizer from sklearn.preprocessing

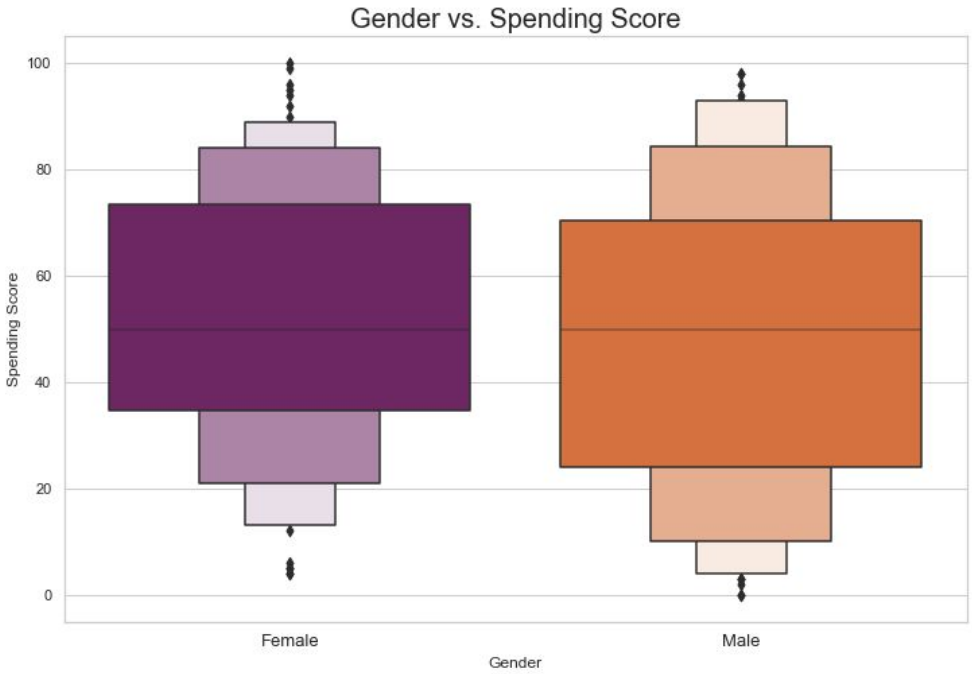
I found out that the highest correlation is between Age and Spending Score and it is a negative correlation equal to -0.33

The positive correlation occurs between Gender and Annual Income, Age and Gender and Annual Income and Spending Score

Data Visualization

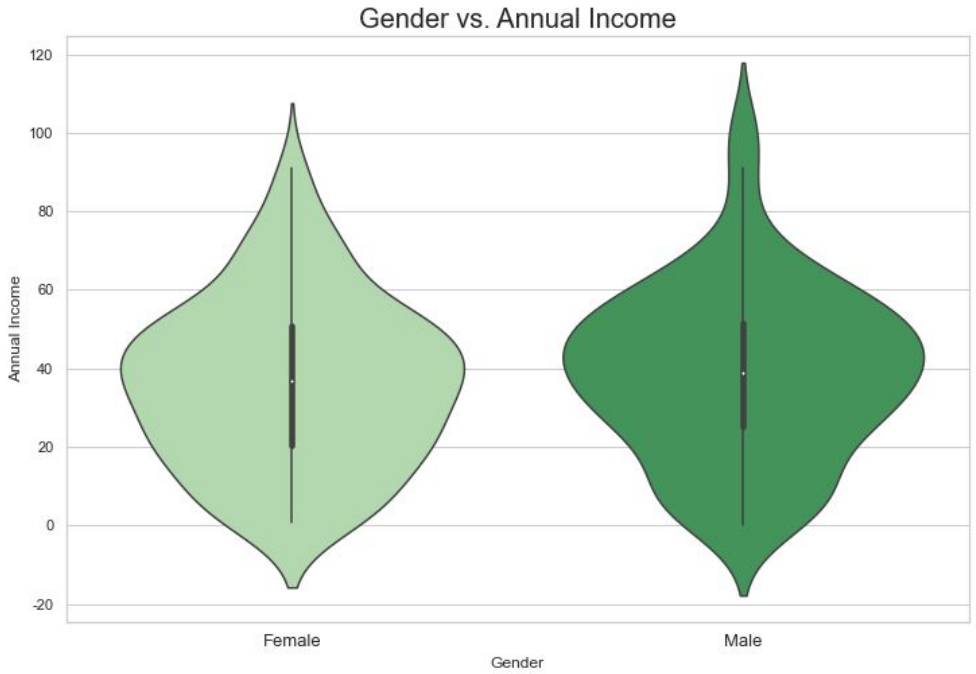


Data Visualization



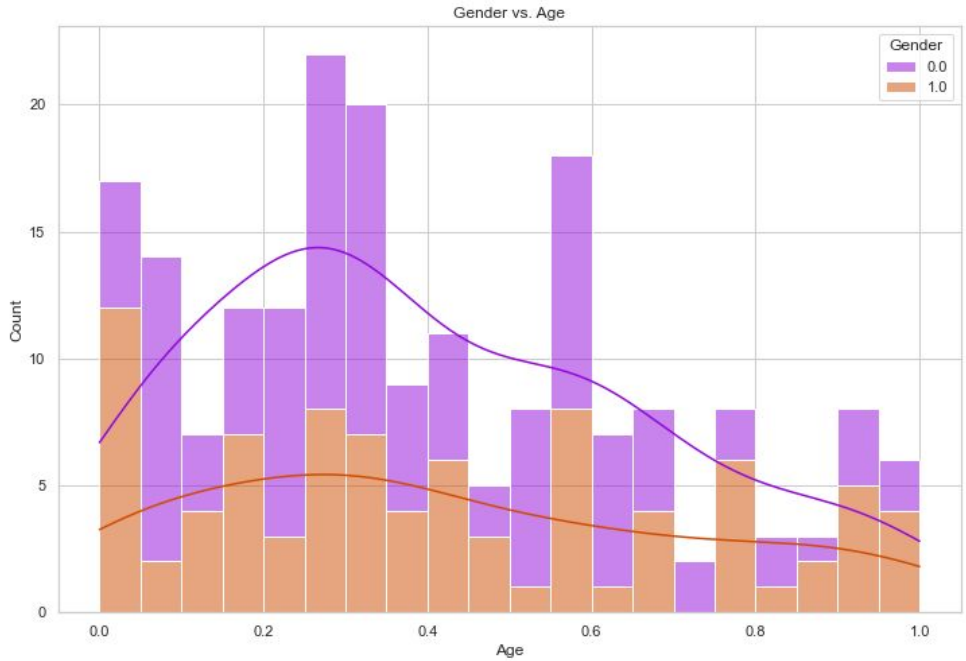
The range for Q1 and Q3 for Male are wider than for Female. Mostly, 50% of data for Male lies in range between 25 - 70 spending score. For Female this range is a little bit tighter: 38 - 75 spending score. The median for both is at the very similar level. More Male are getting lower spending scores than Female and more Female are in higher level of spending score, which could conclude then in general Female are better shopper.

Data Visualization



The interpretation for this graph is similar to interpretation of boxplot. In this case again the median for both gender lies at almost the same place but there are more Annual Income outliers for Male, which in general confirmed the theory that Male got higher income level than Female.

Data Visualization



0: Female
1: Male

Skewness and features scaling

Checking for skewness and taking limit as 0.75 told me that log transformation won't be needed in this case.

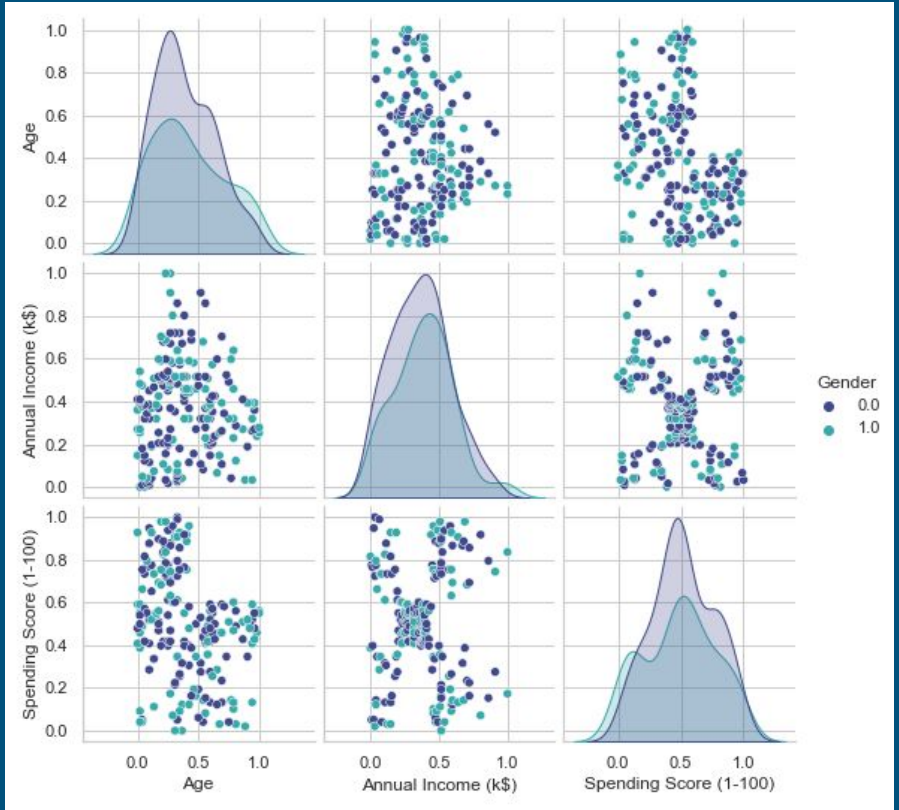
Anyway as my column were int64 data type I firstly converted them to float and then I used MinMaxScaler from sklearn.preprocessing to scale all my variables.

```
df.skew()

Gender                    0.243578
Age                       0.485569
Annual Income (k$)        0.321843
Spending Score (1-100)   -0.047220
dtype: float64
```

Skewness and features scaling

As all of pre processing work was done I used sns.pairplot function on my scaled data to visualize them.



Clustering Modeling

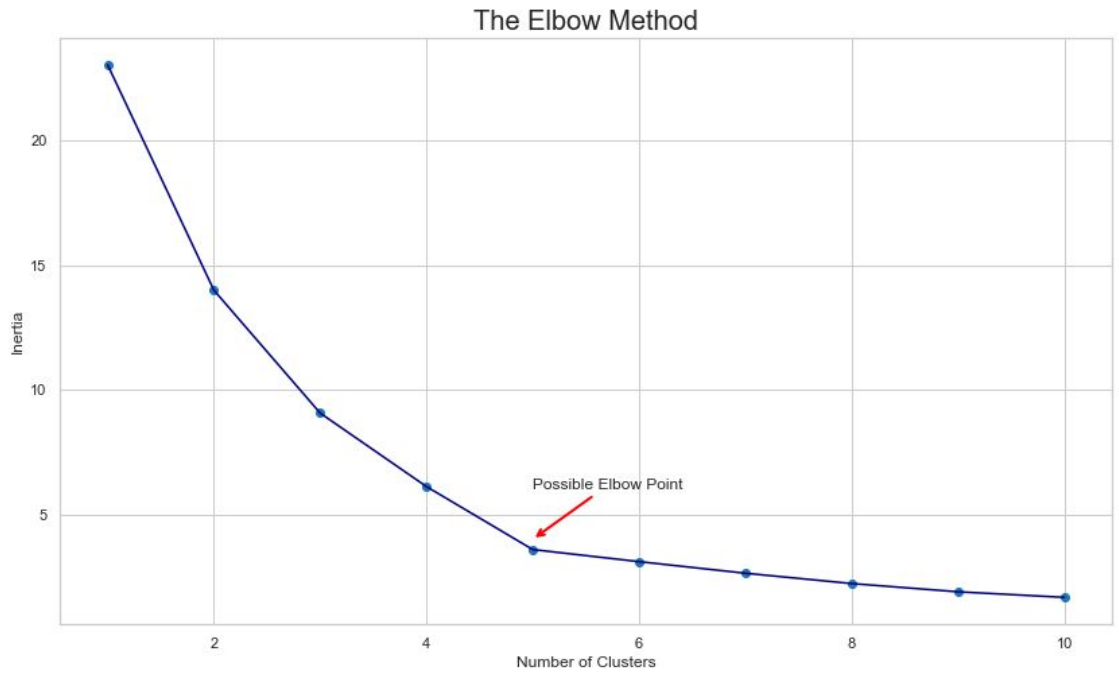
1st scenario: cluster analysis of Spending Score and Annual Income

KMeans

Before I start perform machine learning analysis and run some algorithms I decided to drop column Gender, because first it won't bring many valuable information to my clustering method and secondly by doing this I tried to avoid biases in my analysis due to gender factor.

Then, I want to check how many clusters are the most suitable for my data. To do this I will loop through 10 different cluster numbers on default KMeans algorithms and check which one perform the best and elbow method allowed me to see it.

KMeans

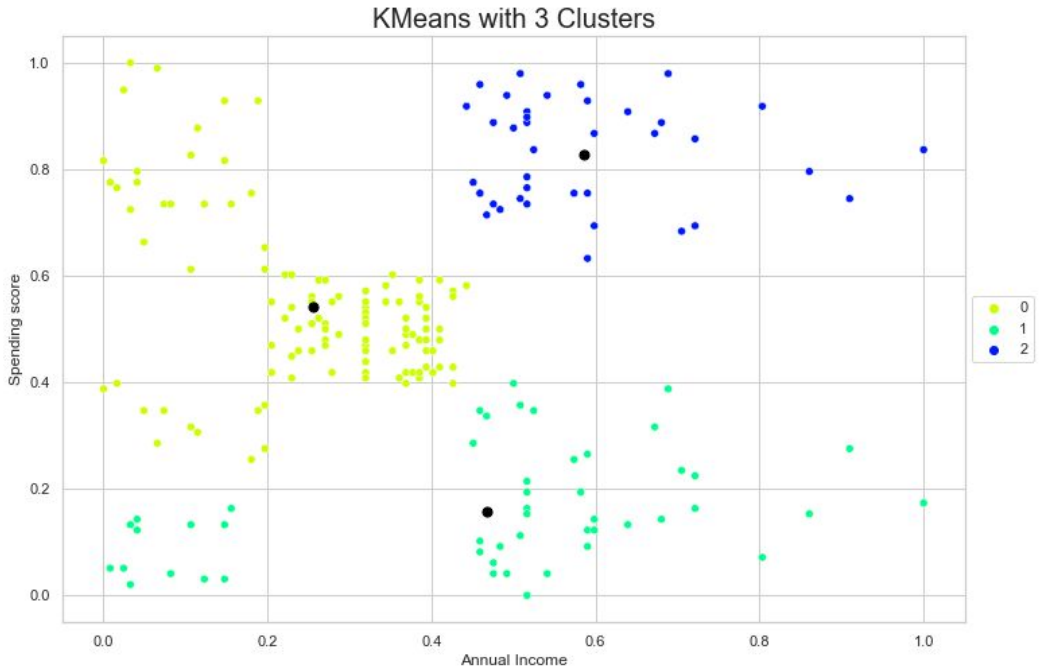

The Elbow Method
Inertia
20
15
10
5
Possible Elbow Point
2
4
6
8
10
Number of Clusters

KMeans

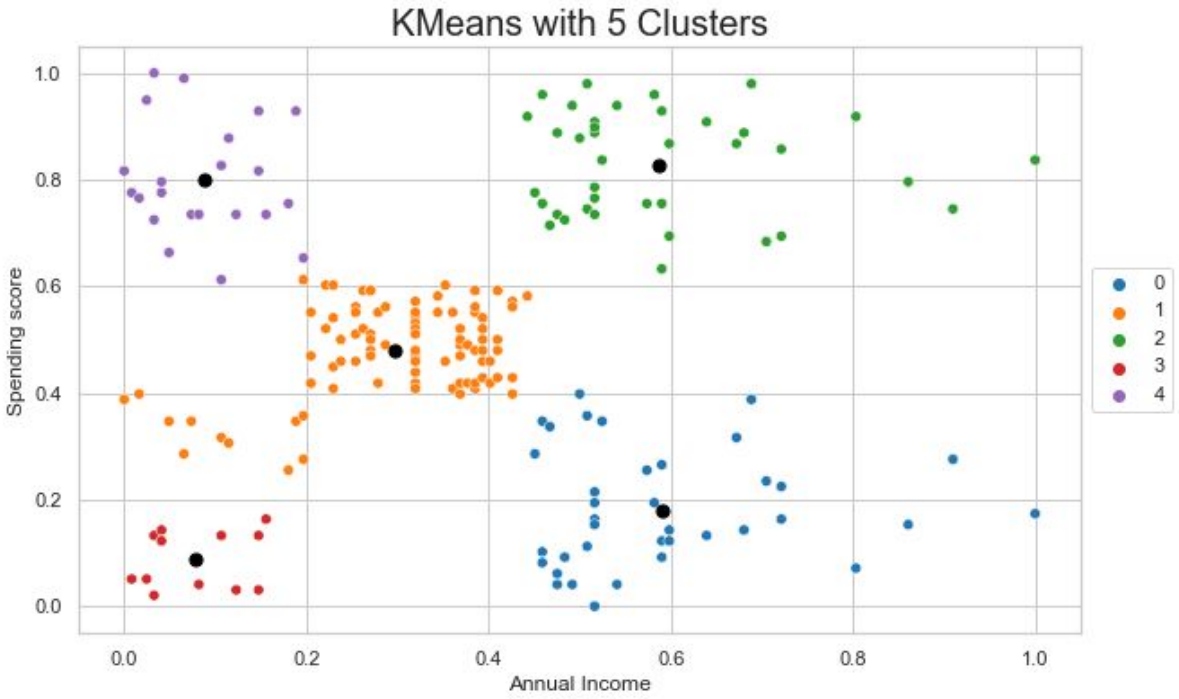
I visualize KMeans with 3 and 5 clusters to better understand which one perform better and split customers into more reasonable groups.

This segmentation with 5 clusters give me chance to more clearly identify customers segments.

KMeans with 3 clusters



KMeans with 5 clusters



Hierarchical Clustering

As I already found the best number of clusters which is 5, I will perform the next clustering machine learning algorithms based on that knowledge.

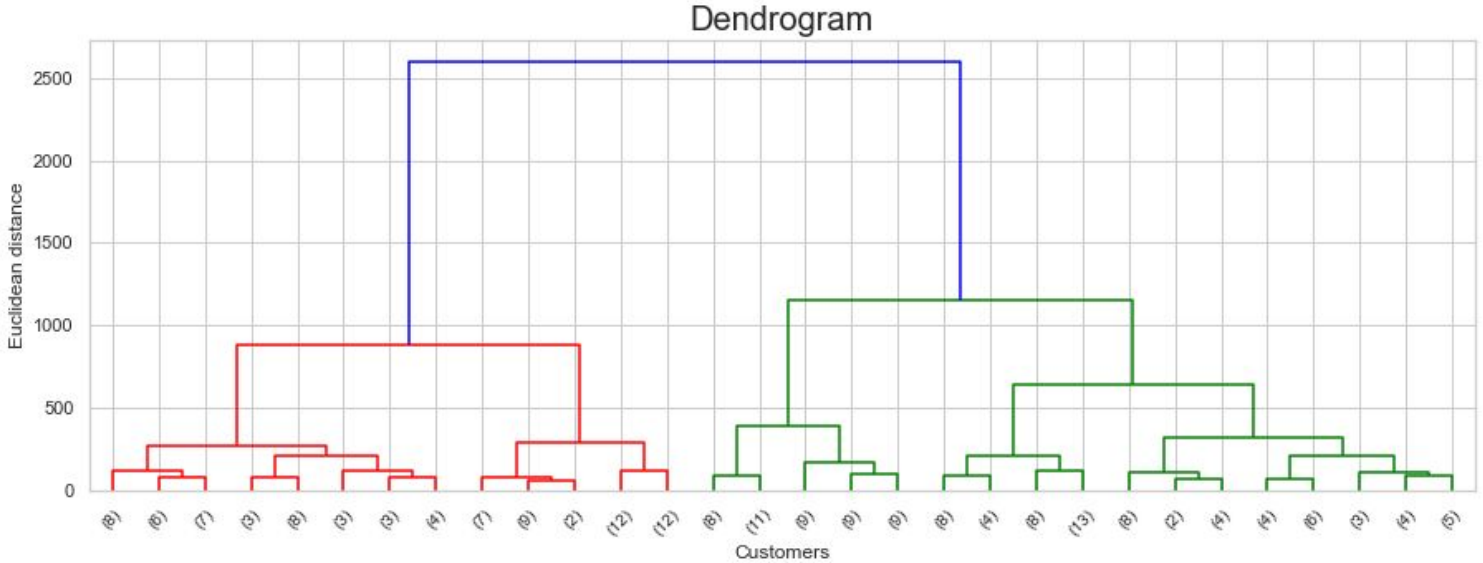
I want to check how one of linkage method - ward will perform.

The linkage criterion determines which distance to use between sets of observation. The algorithm will merge the pairs of cluster that minimize this criterion.

- ➔ **ward** minimizes the variance of the clusters being merged
- ➔ **average** uses the average of the distances of each observation of the two sets
- ➔ **complete** or maximum linkage uses the maximum distances between all observations of the two sets
- ➔ **single** uses the minimum of the distances between all observations of the two sets

Hierarchical Clustering

Both clustering algorithms: KMeans and Hierarchical Clustering did good job in labeling our customers into 5 different groups and both results are similar. I also plot a dendrogram created from agglomerative clustering which was also very helpful in confirming that 5 clusters for this scenario are optimal number.



DBSCAN

Most of the traditional clustering techniques, such as k-means, hierarchical and fuzzy clustering, can be used to group data without supervision.

However, when applied to tasks with arbitrary shape clusters, or clusters within cluster, the traditional techniques might be unable to achieve good results. That is, elements in the same cluster might not share enough similarity or the performance may be poor. Additionally, Density-based Clustering locates regions of high density that are separated from one another by regions of low density. Density, in this context, is defined as the number of points within a specified radius.

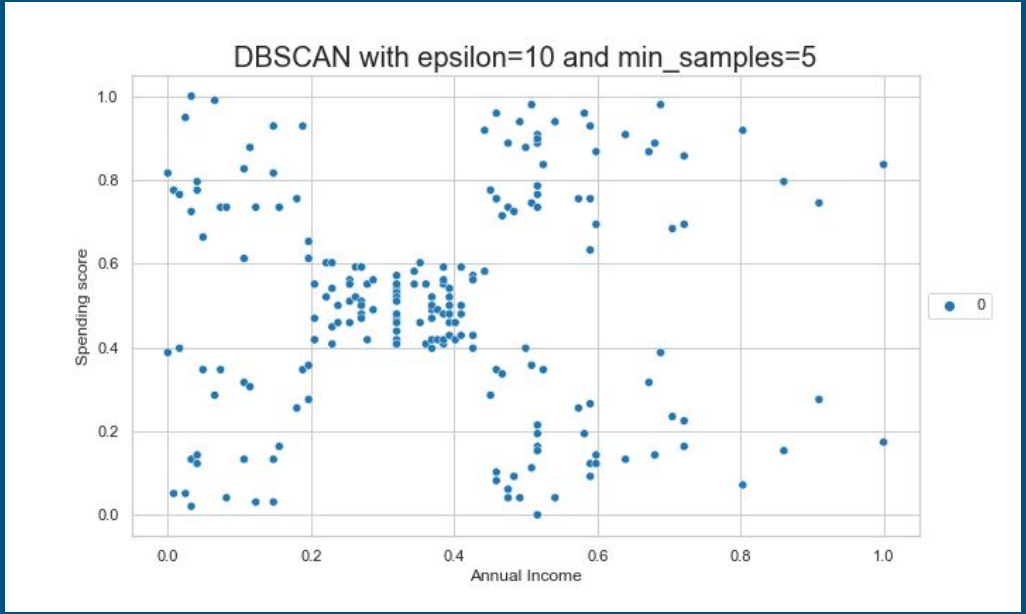
I focused on two parameters when performing this algorithms:

epsilon: the maximum distance between two samples for one to be considered as in the neighborhood of the other. This is not a maximum bound on the distances of points within a cluster. This is the most important DBSCAN parameter to choose appropriately for your data set and distance function

min_samples: the number of samples (or total weight) in a neighborhood for a point to be considered as a core point. This includes the point itself

DBSCAN

No matter what value of epsilon or min_samples I took, the algorithm performed in similar way - all data was classified as one group. The reason why DBSCAN doesn't perform very well is a fact that density in our data isn't so strong. Probably, if the dataset will be bigger DBSCAN will donne better job.



Mean Shift

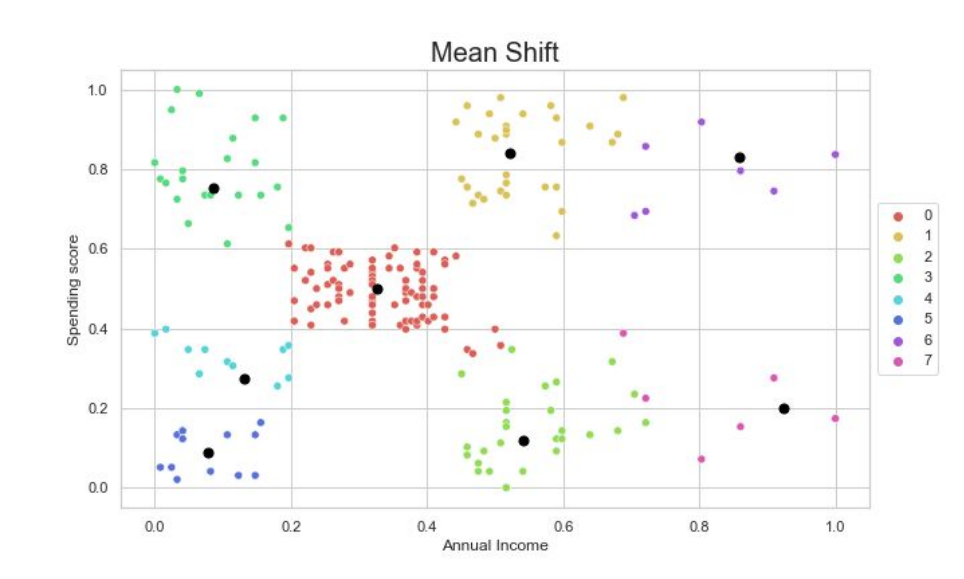
MeanShift clustering aims to discover blobs in a smooth density of samples. It is a centroid based algorithm, which works by updating candidates for centroids to be the mean of the points within a given region. These candidates are then filtered in a post-processing stage to eliminate near-duplicates to form the final set of centroids.

The algorithm automatically sets the number of clusters, instead of relying on a parameter bandwidth, which dictates the size of the region to search through. This parameter can be set manually, but can be estimated using the provided estimate_bandwidth function.

Mean Shift

Using following bandwidth allowed me to performed in the presented way.

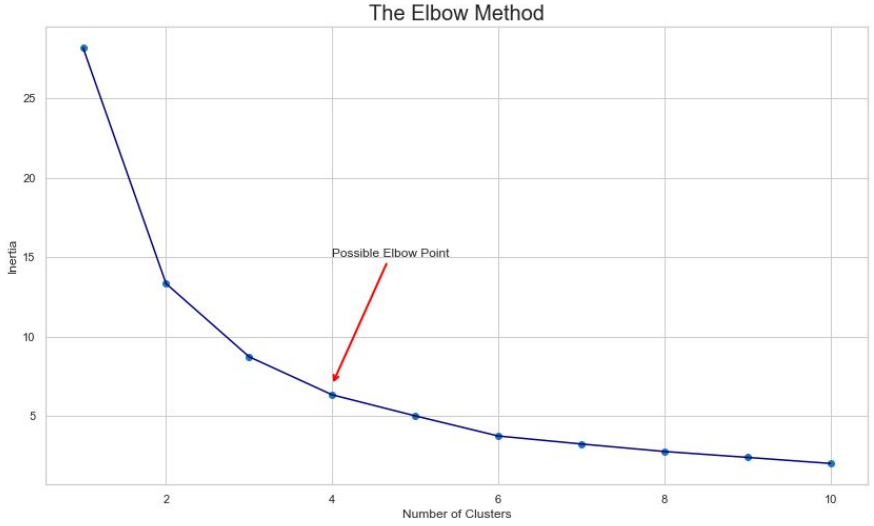
```
# the following bandwidth can be automatically detected using
bandwidth = estimate_bandwidth(X, quantile=0.1)
```



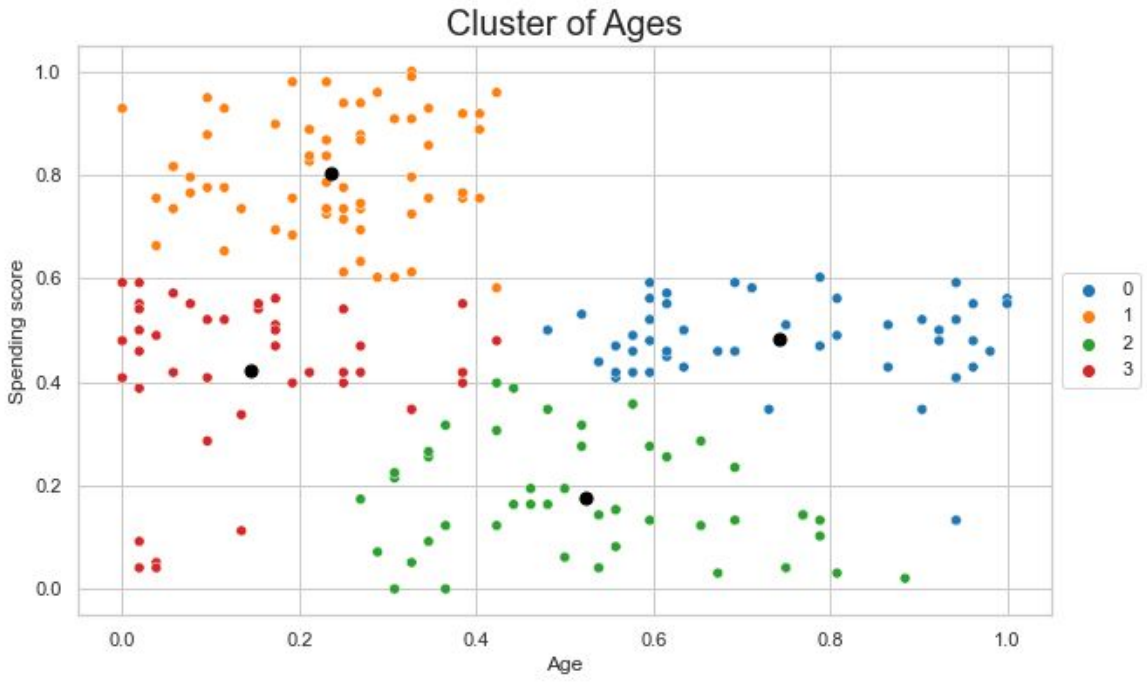
2nd scenario: cluster analysis of Spending Score and Age

KMeans

I did the same steps which like in scenario 1. First by looping through different numbers I found optimal number of clusters. In this case I could be 4.

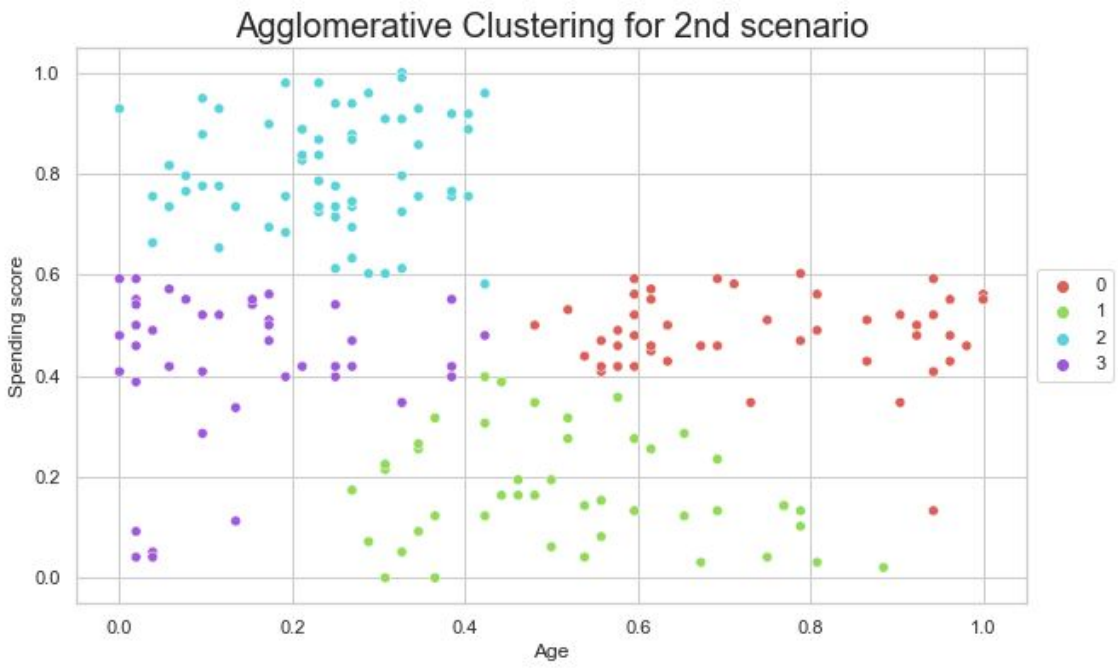


KMeans



Cluster of Ages gave me clear insight how customers can be segmented based on their age and spending score. As we can see there are 4 different groups.

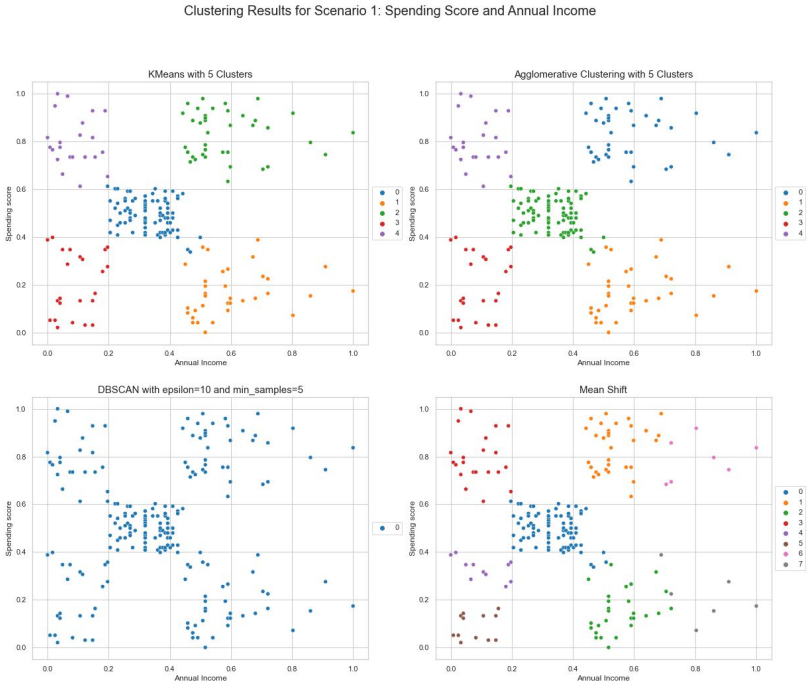
Hierarchical Clustering



As I did before, in this scenario I will also perform default linkage method ward for this algorithm. I will also set number of clusters equal to 4 as I previously found out that this number could be the more optimal value for dataset I work on right now.

Key Findings

1st scenario



Graph on the left contains in one place all visualizations presented in first section about 1st scenario.

It is a little bit hard to read this in format like this (some zooming will be needed) but all algorithms beside DBSCAN done job very well and show us 5 different clusters for customers who visit our Mall based on their spending score and annual income.

1st scenario

Based on that information I can split customers into 5 different groups:

- Label 0: mid income and mid spending score --> **general customers**
- Label 1: high income and low spending score --> **miser customers**
- Label 2: high income and high spending score --> **target customers**
- Label 3: low income and high spending score --> **spendthrift customers**
- Label 4: low income and low spending score --> **careful customers**

2nd scenario

Graph on the left contains in one place all visualizations presented in second section about 2nd scenario.

It is a little bit hard to read this in format like this (some zooming will be needed) but both KMeans and Hierarchical Clustering done job very well and identified 4 different ages groups.

2nd scenario

Based on that information I can split customers into 4 different groups:

- Label 0: low age and mid spending score --> **Young Customers**
- Label 1: high age and mid spending score --> **Senior Citizen Customers**
- Label 2: mid age and low spending score --> **Usual Customers**
- Label 3: low age and high spending score --> **Priority Customers**

Limitations

The overall job was done. The marketing team based on this analysis can identify and segment customers in two directions: first based on their income and spending score allowed them create marketing strategy for 5 different groups, and second direction involves segmentation based on customers age which allowed to prepared strategy tailored for 4 cluster ages.

But still there are some limitations worthy to mentions:

- ★ this dataset from Kaggle was really small one, because of it DBSCAN couldn't perform well
- ★ once the dataset will be updated, the algorithms could be review
- ★ the goal was to perform some basic clustering methods and was done pretty good but still models could be further improved and parameters could be tuned to perform even better