Mall Customers Segmentation

1. Introduction

Customer segmentation is the practice of dividing a customer base into groups of individuals that are similar in specific ways relevant to marketing, such as age, gender, interests and spending habits.

Companies employing customer segmentation operate under the fact that every customer is different and that their marketing efforts would be better served if they target specific, smaller groups with messages that those consumers would find relevant and lead them to buy something.

Benefits of customer segmentation include:

1. *Personalisation*
   - Personalisation ensures that you provide exceptional customer experience.
2. *Customer Retention*
   - It is 16 times as costly to build a long-term business relationship with a new customer than simply to cultivate the loyalty of an existing customer.
3. *Better ROI for marketing*
   - Affirmations that right marketing messages are sent to the right people based on their life cycle stage.
4. *Reveal new opportunities*
   - Customer segmentation may reveal new trends about products and it may even give the first mover's advantage in a product segment.

2. Case Study

The goal of this project is to perform a mall customers segmentation using Machine Learning algorithms. Two techniques will be presented and compared: KMeans and DBSCAN, those will help us to maximize the similarity of observation and identify customer's segments.

For this project I used the Mall Customers Segmentation Data. The fictional dataset contains 200 observations and 5 features:

- **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 (1–100):** Score assigned by the mall based on customer behaviour and spending nature.

3. Feature Engineering

No null (*NaN*) or undefined values were found in the dataset.

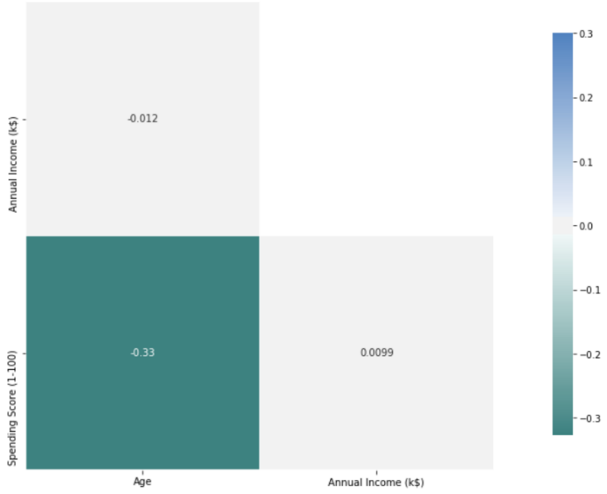
The feature 'CustomerID' was not relevant for our analysis therefore the column has been removed.

4. Data Exploration - Key Findings and Insights

The below *Heatmap* graphically illustrates correlations among all variables.

Heatmaps are appealing to the eyes, and they tend to send clear messages about data almost immediately:

- **Grey** fields represent no correlation
- Green fields reveal a **negative** or indirect correlation - the variation of a characteristic inversely affects that of the other
- Blue reveals a **positive** or direct correlation - the variation of one characteristic directly affects the other

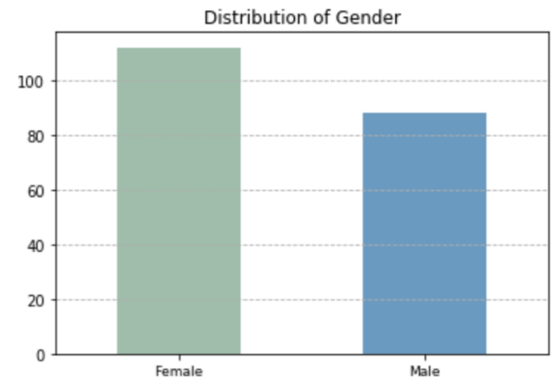


The correlations in the heatmap shows the following characteristics:

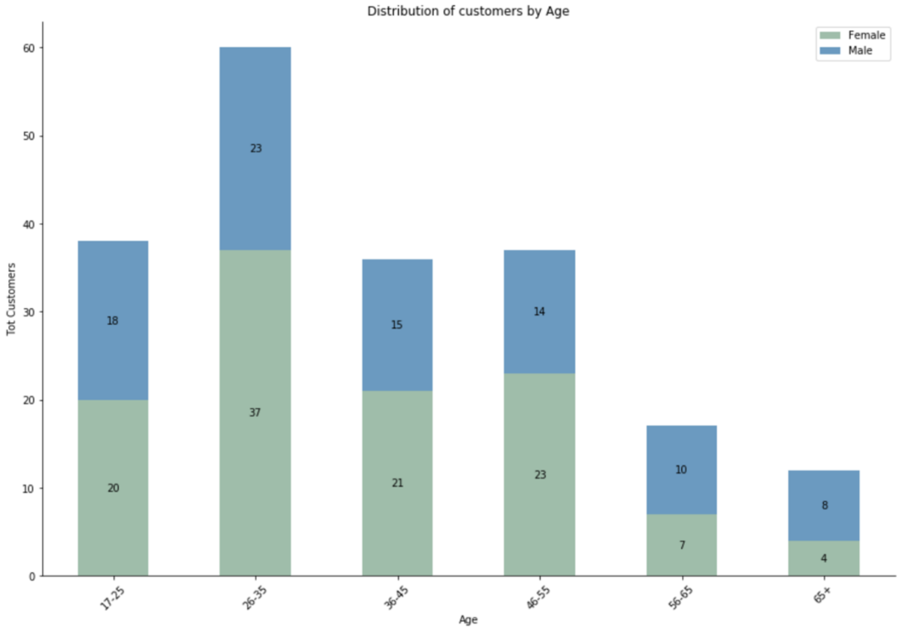
- There is a negligible correlation between age and annual income of customers
- There are week negative correlations (<0.5) between age and spending score
- There is a negligible correlation between annual income and spending score of customers

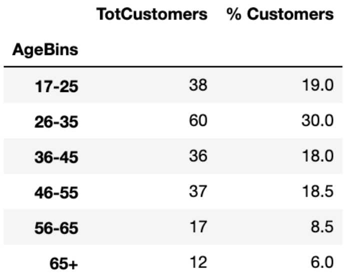
5. Descriptive Analysis

There are slightly more female customers than male (112 vs. 87). Females are 56% of total customers:



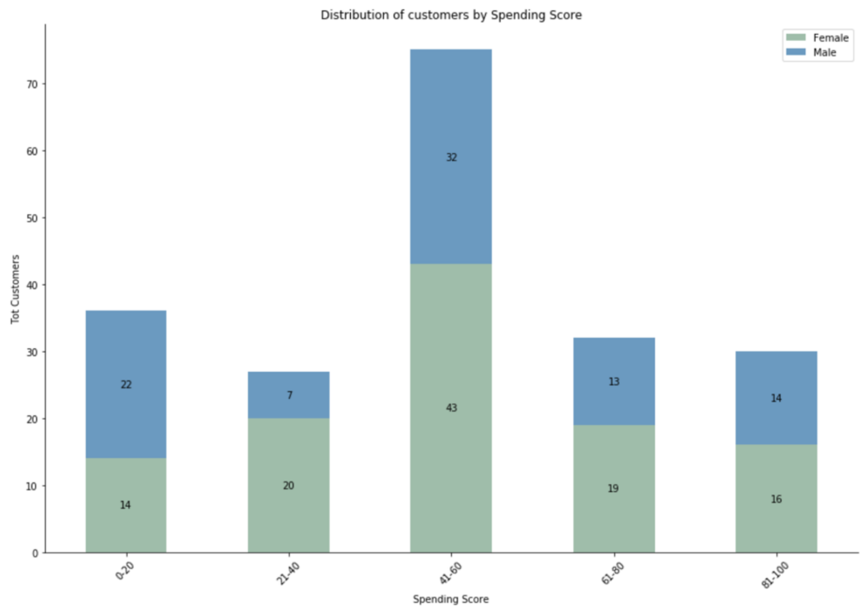
It appears that in both groups (males & females) there is a strong activity in the ages 26-35, while the data shows another frequent group from the female part in the range 46-55. In contrast, both of the groups' curve declines as the age reaches 65.

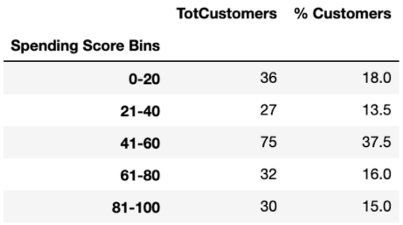




	TotCustomers	% Customers
AgeBins		
17-25	38	19.0
26-35	60	30.0
36-45	36	18.0
46-55	37	18.5
56-65	17	8.5
65+	12	6.0

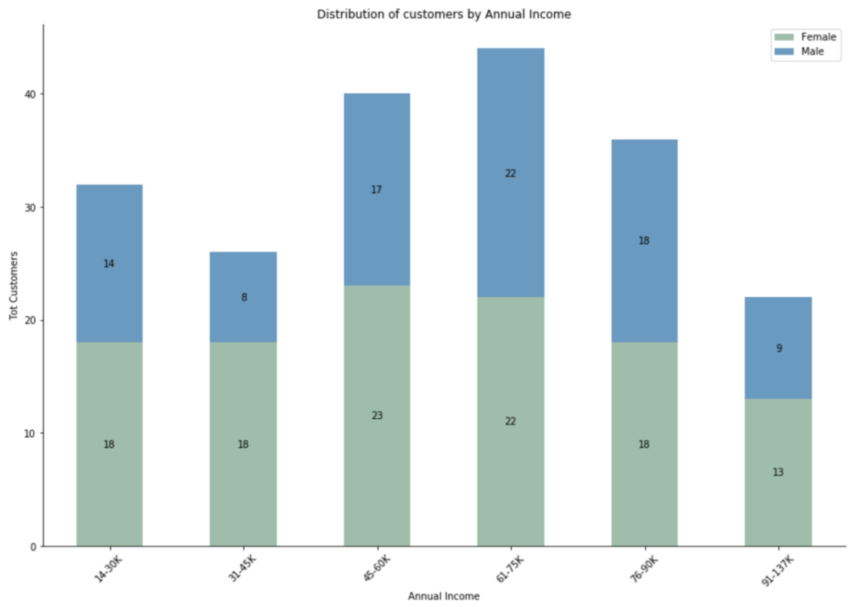
The majority of the customers have spending score in the range 41–60 with females being the group with majority of spending.

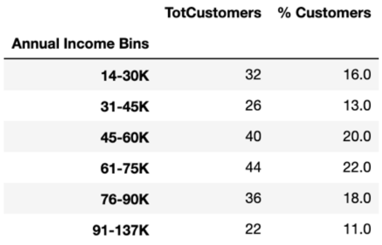




	TotCustomers	% Customers
Spending Score Bins		
0-20	36	18.0
21-40	27	13.5
41-60	75	37.5
61-80	32	16.0
81-100	30	15.0

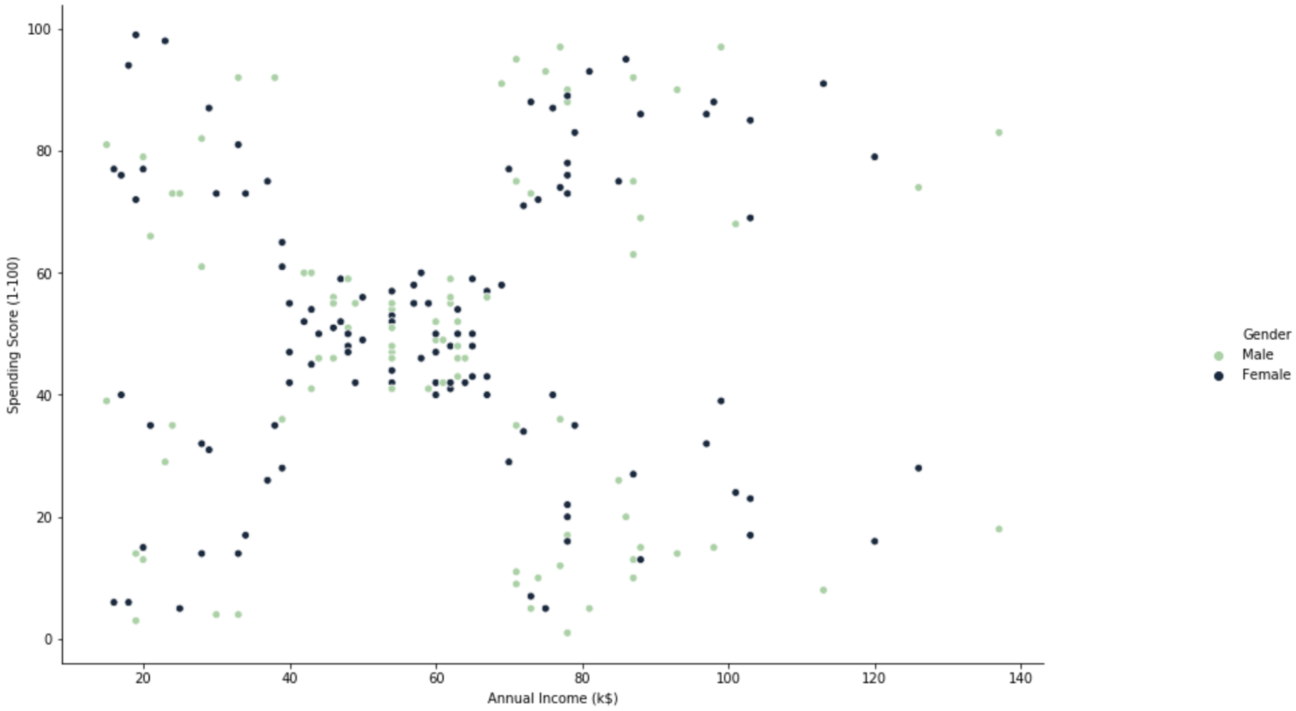
It appears that in both groups (males & females) the majority of the customers have an annual income in the range of 45-75K ($):





	TotCustomers	% Customers
Annual Income Bins		
14-30K	32	16.0
31-45K	26	13.0
45-60K	40	20.0
61-75K	44	22.0
76-90K	36	18.0
91-137K	22	11.0

The ‘Annual Income’ and the ‘Spending Score’ variables are the ones that most interests us since are the one we will use to perform the clustering. We can see that they behave normally and not any anomalies are detected.

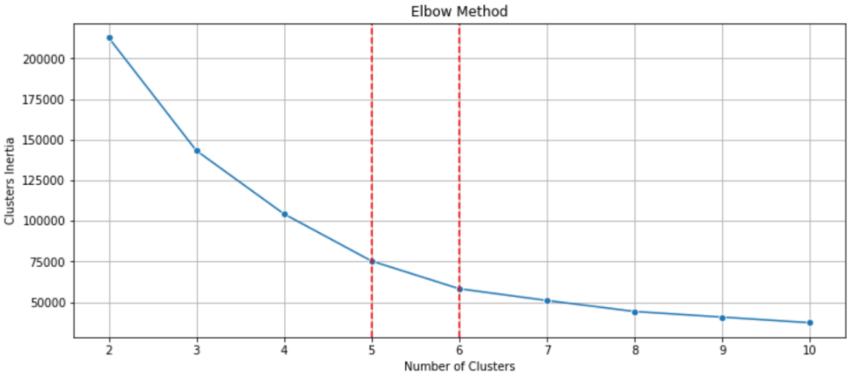


6. Clustering

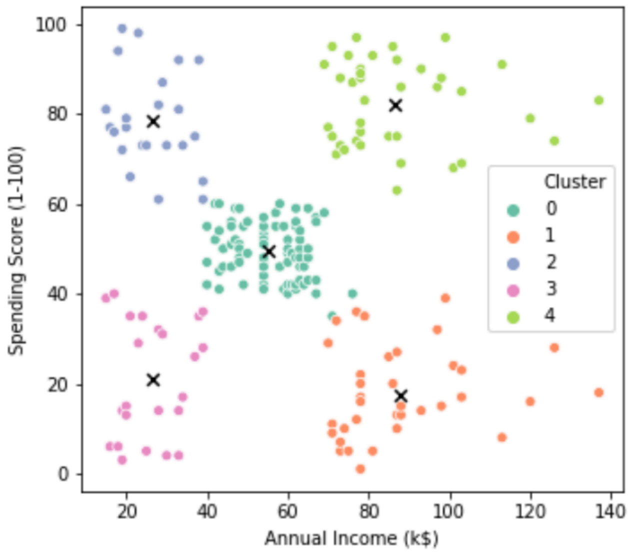
K-Means

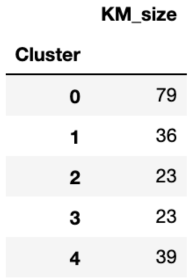
K-means is a simple unsupervised machine learning algorithm that groups a dataset into a user-specified number (*k*) of clusters. As it is important to determine whether they are using the right number of clusters, one method to validate the number of clusters is the elbow method.

The idea of the elbow method is to run k-means clustering on the dataset for a range of values of k (say, k from 2 to 10 in our specific case), and for each value of k calculate the sum of squared errors (SSE):



It can be seen from the graph above that a reasonable selection for the K value would be the k = 5. Hence, we are going to create 5 clusters to generate our segments.





	KM_size
Cluster	
0	79
1	36
2	23
3	23
4	39

K-Means algorithm generated the following 5 clusters:

- Cluster 0 - clients with medium annual income and medium spending score
- Cluster 1 - clients with high annual income and low spending score
- Cluster 2 - clients with low annual income and high spending score
- Cluster 3 - clients with low annual income and low spending score
- Cluster 4 - clients with high annual income and high spending score

The biggest cluster is Cluster 0 with 79 observations ("medium-medium" clients). There are two smallest ones, each containing 23 observations (cluster 2 "low-high" and cluster 3 "low-low" clients).

DBSCAN

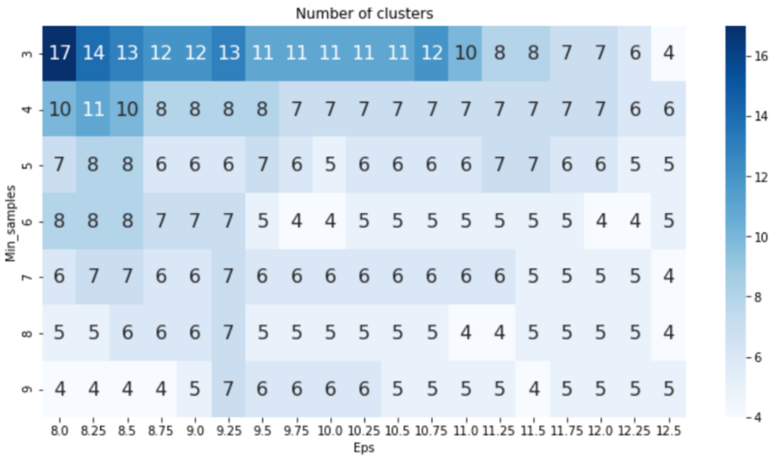
Density-based spatial clustering of applications with noise (DBSCAN) groups together observations that are close to each other based on a distance measurement (usually Euclidean distance) and a minimum number of points. It also marks as outliers the points that are in low-density regions.

The DBSCAN algorithm basically requires 2 parameters:

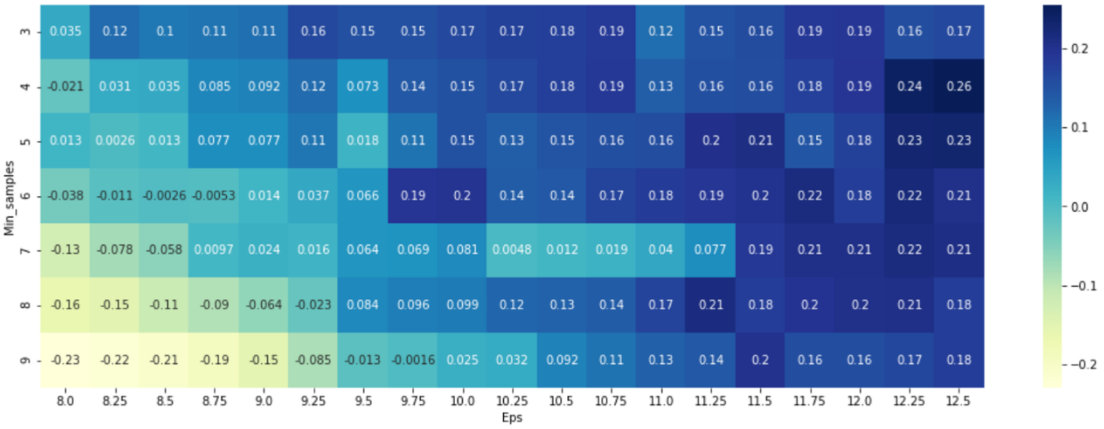
- eps: specifies how close points should be to each other to be considered a part of a cluster. It means that if the distance between two points is lower or equal to this value (eps), these points are considered neighbours.

- Min_sample: the minimum number of points to form a dense region. For example, if we set the minPoints parameter as 5, then we need at least 5 points to form a dense region.

To choose the best combination of the algorithm parameters we will first create a matrix of investigated combinations. The heatplot below shows how many clusters were generated by the algorithm for the respective parameter's combinations: the number of clusters vary from 17 to 4 and global maximum is 0.26 for eps=12.5 and min_samples=4:

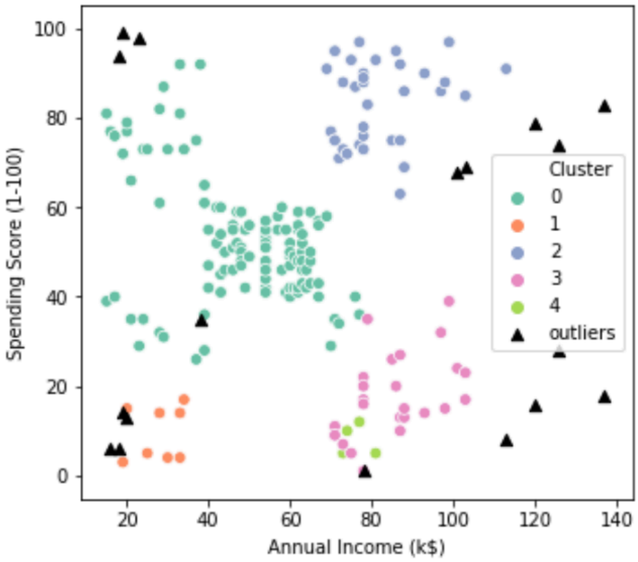


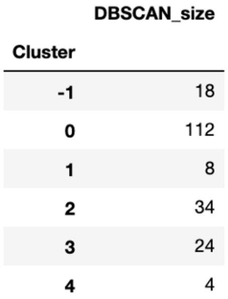
	8.0	8.25	8.5	8.75	9.0	9.25	9.5	9.75	10.0	10.25	10.5	10.75	11.0	11.25	11.5	11.75	12.0	12.25	12.5
3	17	14	13	12	12	13	11	11	11	11	11	12	10	8	8	7	7	6	4
4	10	11	10	8	8	8	8	7	7	7	7	7	7	7	7	7	7	6	6
5	7	8	8	6	6	6	7	6	5	6	6	6	6	7	7	6	6	5	5
6	8	8	8	7	7	7	5	4	4	5	5	5	5	5	5	5	4	4	5
7	6	7	7	6	6	7	6	6	6	6	6	6	6	6	5	5	5	5	4
8	5	5	6	6	6	7	5	5	5	5	5	5	4	4	5	5	5	5	4
9	4	4	4	4	5	7	6	6	6	6	5	5	5	5	4	5	5	5	5



	8.0	8.25	8.5	8.75	9.0	9.25	9.5	9.75	10.0	10.25	10.5	10.75	11.0	11.25	11.5	11.75	12.0	12.25	12.5
3	0.035	0.12	0.1	0.11	0.11	0.16	0.15	0.15	0.17	0.17	0.18	0.19	0.12	0.15	0.16	0.19	0.19	0.16	0.17
4	-0.021	0.031	0.035	0.085	0.092	0.12	0.073	0.14	0.15	0.17	0.18	0.19	0.13	0.16	0.16	0.18	0.19	0.24	0.26
5	0.013	0.0026	0.013	0.077	0.077	0.11	0.018	0.11	0.15	0.13	0.15	0.16	0.16	0.2	0.21	0.15	0.18	0.23	0.23
6	-0.038	-0.011	-0.0026	-0.0053	0.014	0.037	0.066	0.19	0.2	0.14	0.14	0.17	0.18	0.19	0.2	0.22	0.18	0.22	0.21
7	-0.13	-0.078	-0.058	0.0097	0.024	0.016	0.064	0.069	0.081	0.0048	0.012	0.019	0.04	0.077	0.19	0.21	0.21	0.22	0.21
8	-0.16	-0.15	-0.11	-0.09	-0.064	-0.023	0.084	0.096	0.099	0.12	0.13	0.14	0.17	0.21	0.18	0.2	0.2	0.21	0.18
9	-0.23	-0.22	-0.21	-0.19	-0.15	-0.085	-0.013	-0.0016	0.025	0.032	0.092	0.11	0.13	0.14	0.2	0.16	0.16	0.17	0.18

By using these parameters, we can see in the graph below that there are some outliers - these points do not meet distance and minimum samples requirements to be recognised as a cluster:



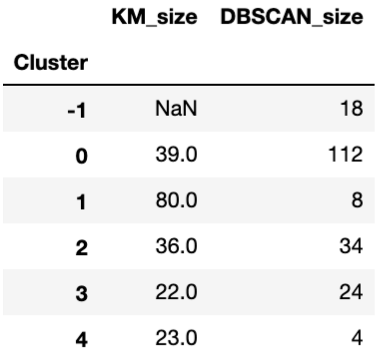


	DBSCAN_size
Cluster	
-1	18
0	112
1	8
2	34
3	24
4	4

DBSCAN created 5 clusters plus outliers' cluster (-1). Sizes of clusters 0-4 vary significantly - some have over 100 observations others only 4 or 8. There are in total 18 outliers.

7. Conclusion and suggestions

When comparing the clustering results of both K-Means and DBSCAN, we can safely say that K-Means performed the best while DBSCAN failed to generate reasonable clusters. This is manly due to its problems in recognising clusters of various densities (which are present in this case).



	KM_size	DBSCAN_size
Cluster		
-1	NaN	18
0	39.0	112
1	80.0	8
2	36.0	34
3	22.0	24
4	23.0	4

Having a better understanding of the customers segmentation, the company could use these results to make better and more informed decisions. As we saw in the analysis, there are customers with a high annual income but a low spending score. It would be interesting to target this segment with marketing actions in order to convert them in higher spenders. The focus should also be on the "loyal" customers and maintain their satisfaction.

For this analysis we used a fictional dataset that only contains two main variables, 'Annual Income' and 'Spending Score'. In a typical business scenario, there will be more variables that would generate more business insight, like frequency of the purchase, family size, distance from the mall, etc.