# Algorithmic Machine Learning
# Final Project Fall 2021.

## Shopping Mall Customer Clustering Report

**Project Group G**
Wenting Wang
Harshal Utekar
Mansi Khanna

**Dataset:** https://www.kaggle.com/roshansharma/mall-customers-clustering-analysis/notebook

We explore the Shopping Mall Customers dataset to gather insights about the customer's spending behavior. We consider age and income to be key factors in determining the spending behavior of the customers.

We aim to identify the clusters based on the income vs spending score rather than age vs spending score as we consider income to be a good factoring and clustering measure to classify the spending behavior.

We will be using 3 clustering methods: k-means, Gaussian mixtures, and SVM and will be determining which method works best as a clustering technique for our data.

The complete analysis and code can be found in the ipynb notebook that we have shared. This report gives a brief about the whole analysis and concludes the best clustering method that is observed.

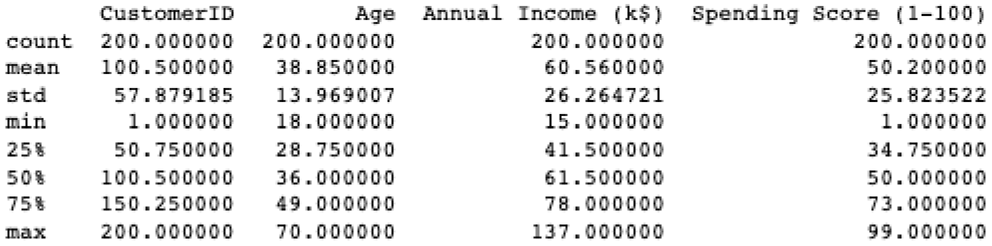
Regarding the work done, everyone had a fair share in the completion of the project. Everyone had a key area they focused on, but the work distribution regarding the exploratory data analysis, the summary of it and the measures has been fairly equal.

**Introduction:**

Clustering is an unsupervised learning algorithm. The goal is to group similar instances together into clusters. Clustering is a great tool for data analysis, customer segmentation, and more.

Our team was analyzing a shopping mall customer data set. The data set includes four features, Gender, Age, Income, and Spending score and 200 observations. We can cluster the customers based on their information. This is useful to understand who our customers are and what they need. Based on that useful information, we can adapt the products and marketing policy.

**Data preprocessing**



```
       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
```

Table 1

Our data set doesn’t include any missing data. So, the only thing we have to do is feature scaling. Machine learning algorithms don’t perform well when the input numerical attributes have very different scales. The annual income range from 15k to 137k, while the spending score only ranges from 1 to 99. We use min-max scaling (normalization). All values are shifted and rescaled so that they end up ranging from 0 to 1.

## Data Visualization

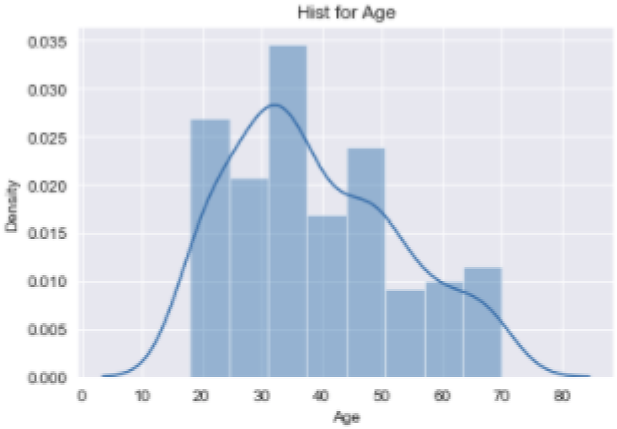


Figure 1

To better understand our data set. We use the histogram to display the Age distribution. 70% of our customers are within the range from 18 to 50 years old. Age 30-40 years old is the majority of the customers.

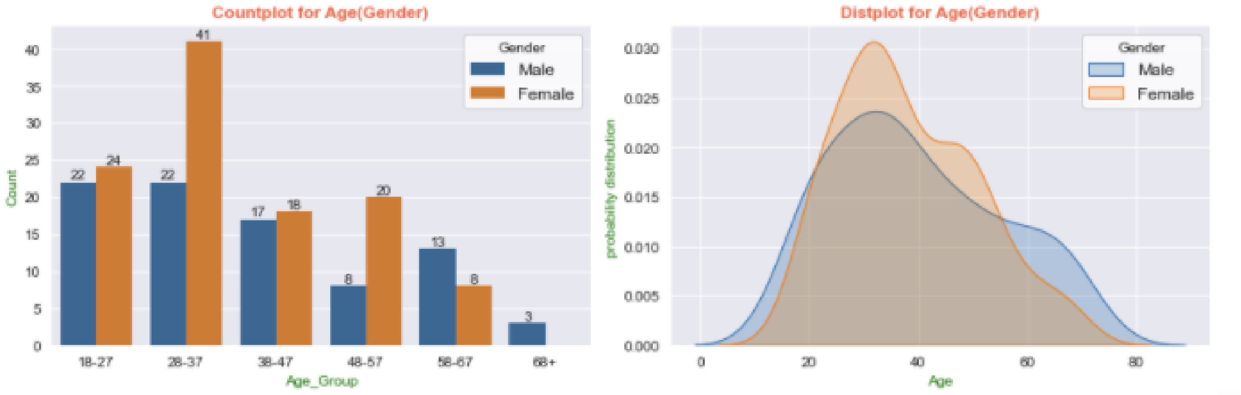


Figure 2

Figure 2 shows the Age groups by gender. We can easily conclude that the number of female customers is higher than male customers especially in the age group 28-37 years old.

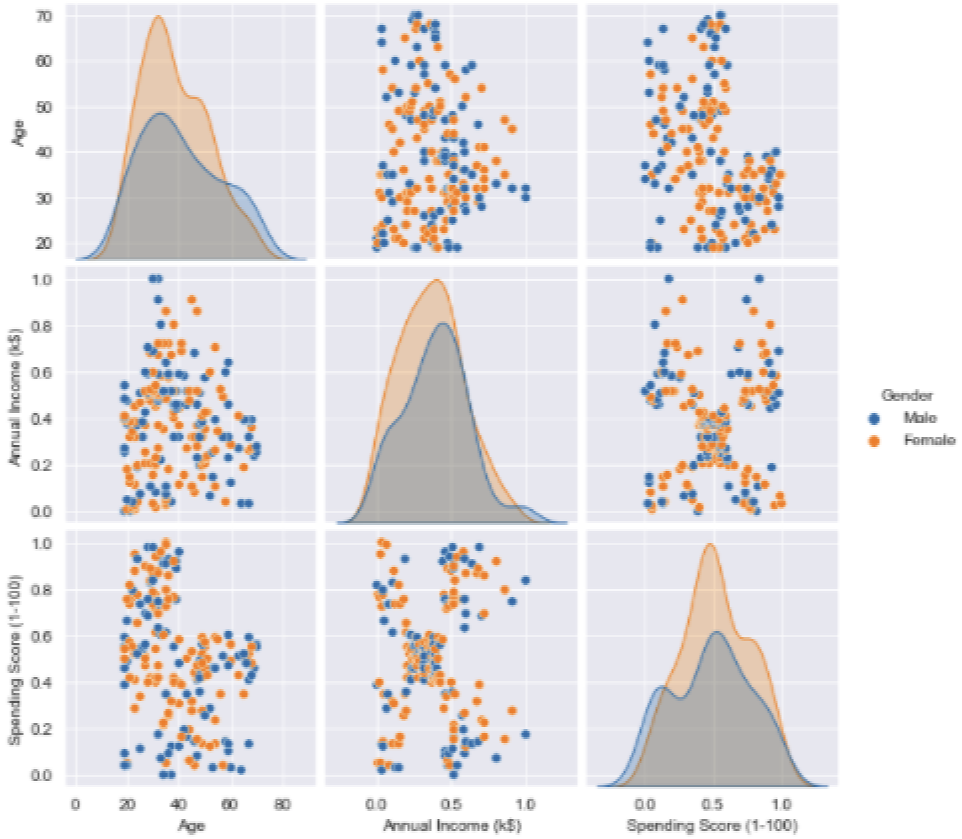


Figure 3

The pairs plot builds on two basic figures, the histogram and the scatter plot. The histogram on the diagonal allows us to see the distribution of a single variable while the scatter plots on the upper and lower triangles show the relationship (or lack thereof) between two variables.

From Figure 3, we can clearly see five blobs of instances from the scatter plot of Annual income versus the Spending score. But we will try different k to make sure we get the accurate result.

## K-Mean Clustering

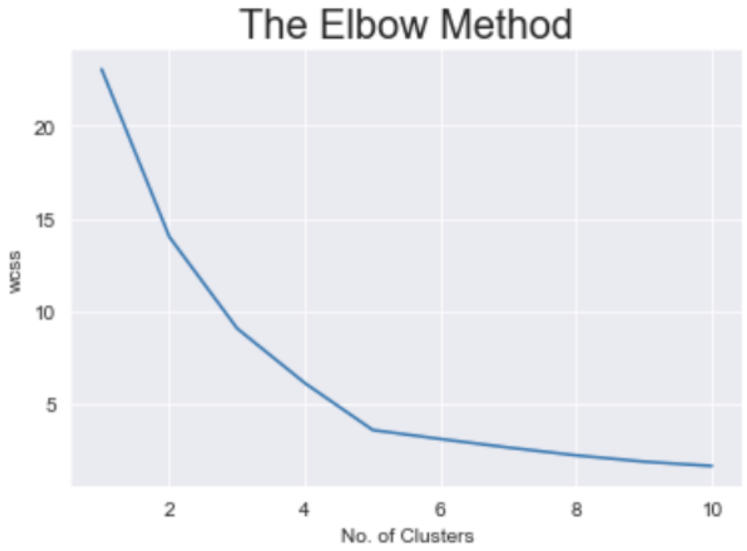


Figure 4

Figure 4 shows the results of K-mean clustering with different k, ranging from 1 to 10. With the k increase, the WCSS, which is also called inertia, keeps getting lower. So, it is not a good performance metric to choose k. Indeed, the more clusters there are, the closer each instance will be to its closest centroid, and therefore the lower the inertia will be.

In the beginning, the WCSS decreases very quickly as we increase k up to 5, but then it decreases much slowly as we keep increasing k. According to this elbow plot, the good choice is k=5. Just like our assumption at the beginning according to the scatter plot.

A more precise approach to choosing k is to use the silhouette score, which is the mean silhouette coefficient over all the instances.

Figure 5 shows the silhouette scores for different numbers of clusters. This plot is much clear than the elbow plot. It confirms that k=5 is a very good choice.

The left plot of Figure 6 is the silhouette diagram which is a more informative visualization that shows every instance's silhouette coefficient. The height of each block shows the number of instances the cluster contains and the width represents the sorted silhouette coefficients. And the dashed line is the mean silhouette coefficient. (Silhouette diagram for different k can be found in appendix)

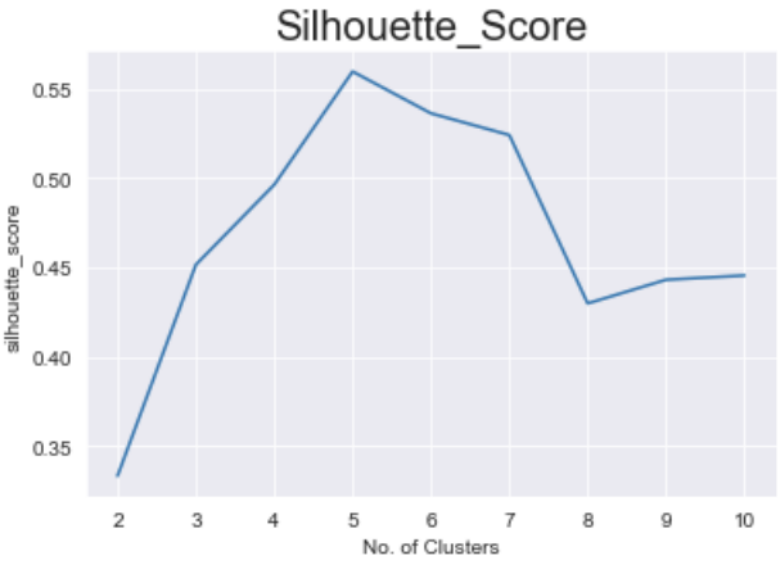


Figure 5

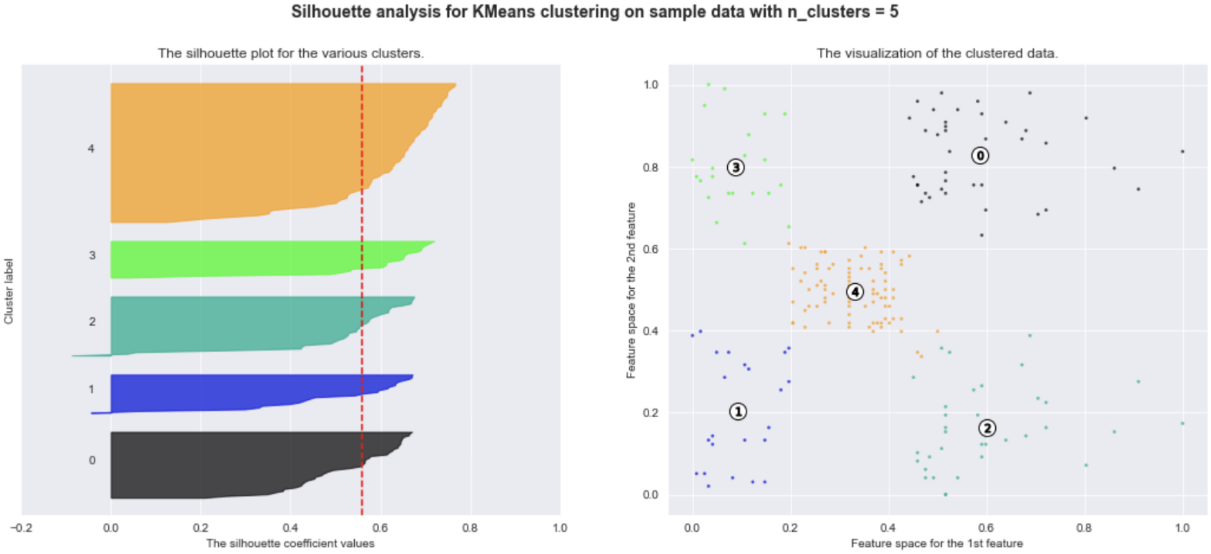


Figure 6

## Gaussian Mixtures

For the Gaussian mixture clustering, we performed two methods to gauge the output of clustering.

1. With Gender feature consideration in the clustering.
2. Without Gender feature consideration in the clustering.

To find the accurate number of clusters in both methods we ran AIC and BIC models to get the best value for clusters.

We also ran an anomaly detection model to determine if the gaussian mixture has any anomalies while clustering the points.

### 1. Gaussian mixture with Gender Feature:

We ran the Gaussian mixture with AIC and BIC for a range of 2 to 10 to check for the ideal number of clusters for the data. The below figure shows the AIC and BIC values for the model.

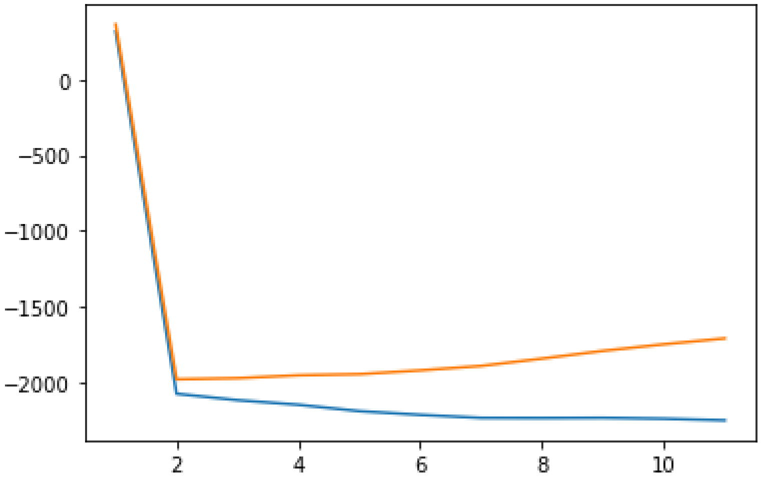


**Figure 7**

As we can see the BIC(yellow line) value is the lowest for k=2 and it seems to be increasing as the value of clusters keeps on increasing indicating that the ideal number of clusters in this case would be 2.

With cluster size as 2 we ran the Gaussian mixture model after fitting and predicting the model.

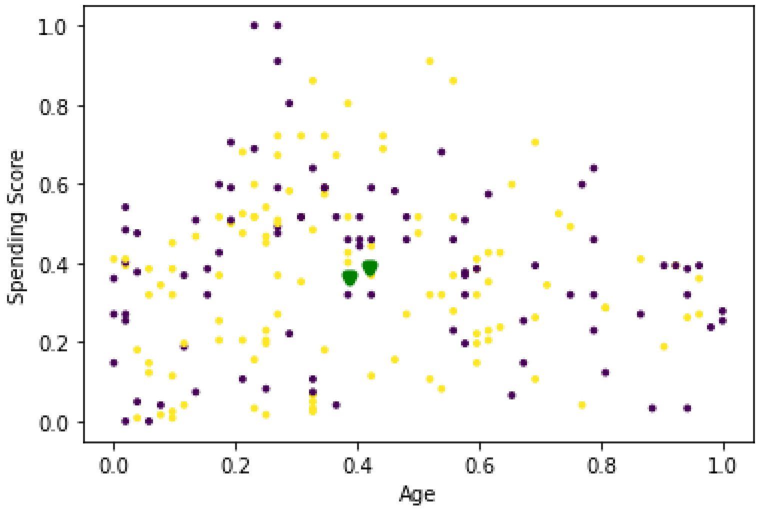


**Figure 8**

As we can see the clustering is not that precise, the central points for both the clusters(marked in green) are very close to each other and the clustering seems scattered. Having a gender feature does not classify the clustering properly.

### 2. Gaussian Mixture without Gender feature:

So, we move on to Gaussian mixture modeling without the Gender feature. We again ran the Gaussian mixture with AIC and BIC to determine the best number of clusters for the model within the range of 2 to 10.

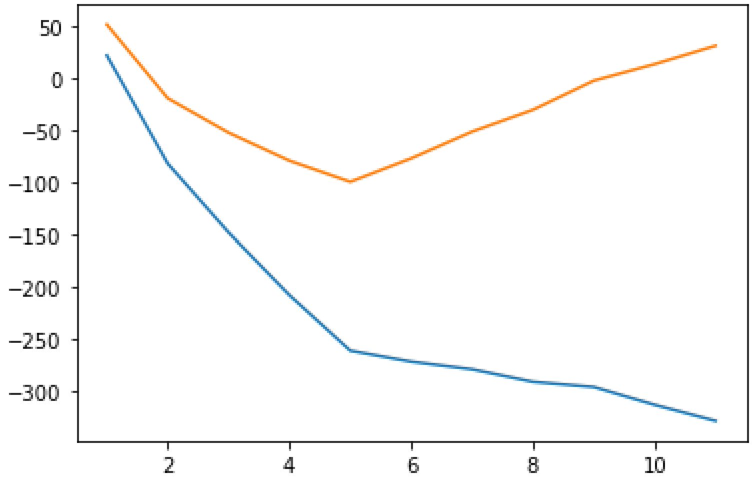


**Figure 9**

From the figure we can clearly see that excluding the ‘Gender’ feature had a significant impact on the AIC and BIC values of the whole model as well. We can see that the BIC(yellow line) has the least value for 5 clusters. The value keeps on decreasing till it reaches for cluster 5 and then starts increasing again. Thus, having 5 cluster components seems to be the best way to approach the model.

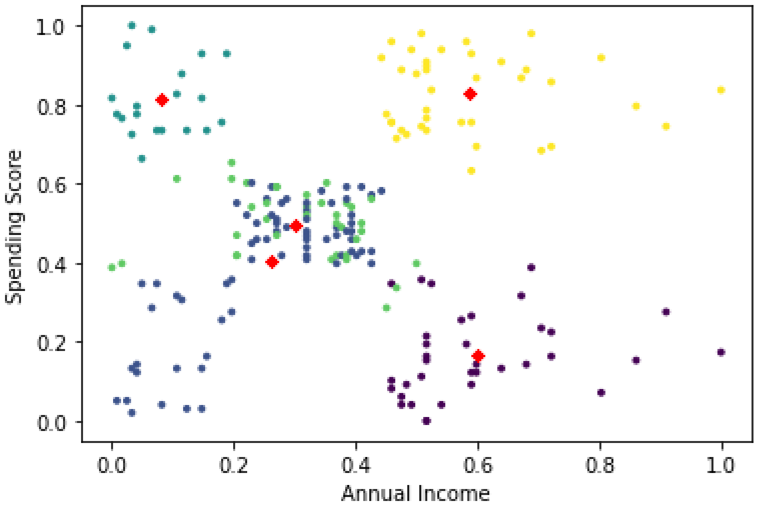


**Figure 10**

From the figure we see the Gaussian mixture modeling for 5 clusters. The clusters seem fairly distributed with the centers (marked in red) fairly spread out, except for the 2 clusters(green and dark blue) which are fairly close and overlap each other. These 2 clusters form the maximum concentration of points which lies between the annual income of about $20k-40k range.

To address this issue we proceeded to find out if it has any anomalies and plotted the matrix to visually address the anomalies.

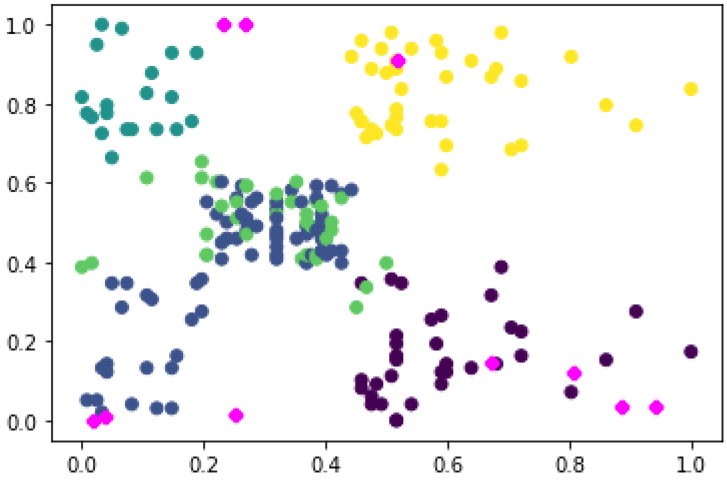


**Figure 11**

The anomalies that we see(marked in Pink) are not directly pertained to the green or purple clusters, where we thought the anomalies to show up for the most part. The anomalies are scattered and very few. So, we can safely say that the data and modeling does not have any major anomalies.

However, this clustering as we can see is not a good measure as the distinction between the age vs spending ratio at its highest concentration in the age group of 40-60 has 2 different clusters.

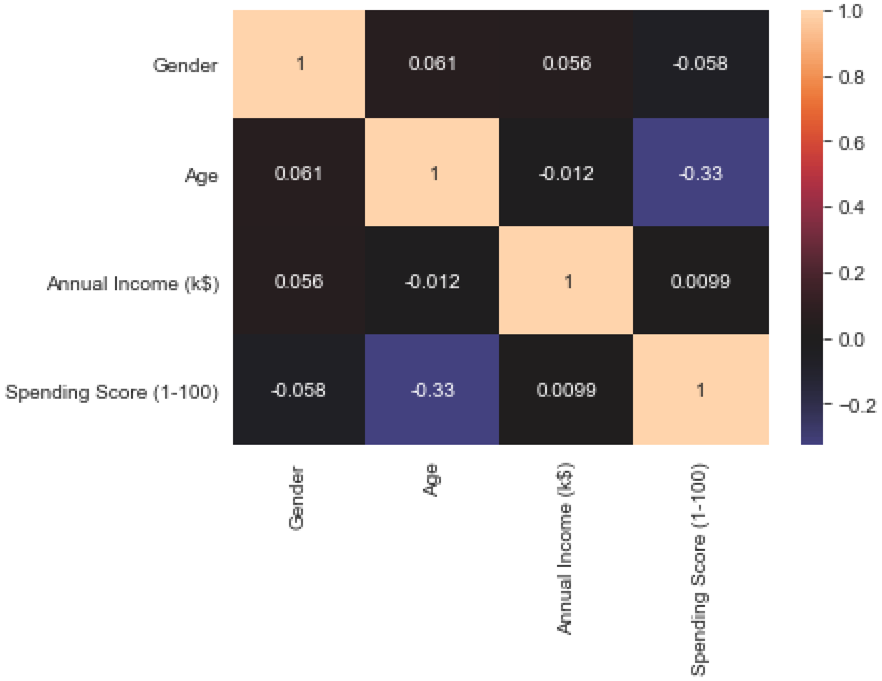
## Support Vector Machine

A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for two-group classification problems. After giving an SVM model sets of labeled training data for each category, they're able to categorize new text.

Compared to newer algorithms, they have two main advantages: higher speed and better performance with a limited number of samples (in the thousands). This makes the algorithm very suitable for text classification problems, where it's common to have access to a dataset of at most a couple of thousands of tagged samples.

### Support Vector Clustering

A natural way to put cluster boundaries is in regions in data space where there is little data, i.e. in "valleys" in the probability distribution of the data. This is the path taken in support vector clustering (SVC).



| | Gender | Age | Annual Income (k$) | Spending Score (1-100) |
|---|---|---|---|---|
| Gender | 1 | 0.061 | 0.056 | -0.058 |
| Age | 0.061 | 1 | -0.012 | -0.33 |
| Annual Income (k$) | 0.056 | -0.012 | 1 | 0.0099 |
| Spending Score (1-100) | -0.058 | -0.33 | 0.0099 | 1 |

**Figure 12**

Above is the heat-map for the Mall-Clustering data. This data helps us distinguish and identify the clusters to be formed to support our method

SVC data points are mapped from data space to a high dimensional feature space using a kernel function.

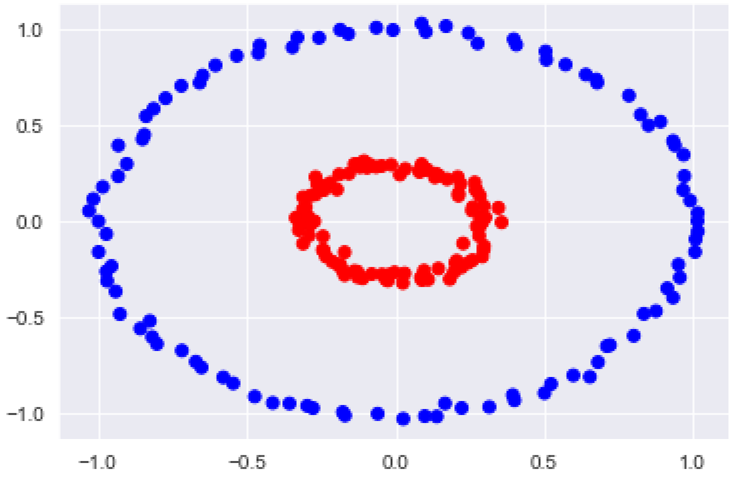


**Figure 13**

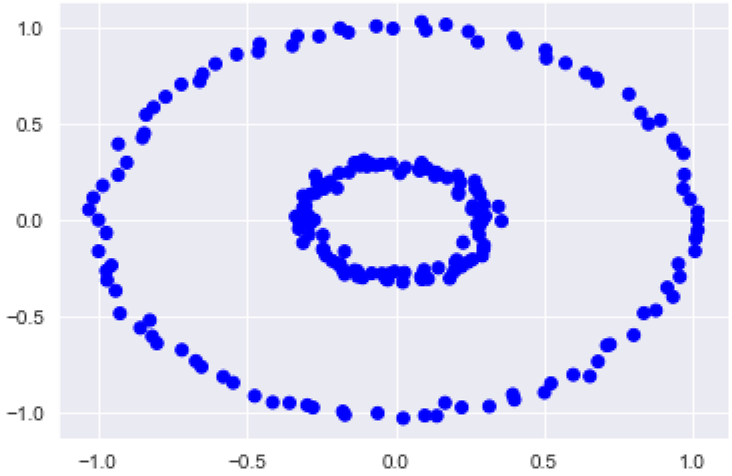


**Figure 14**

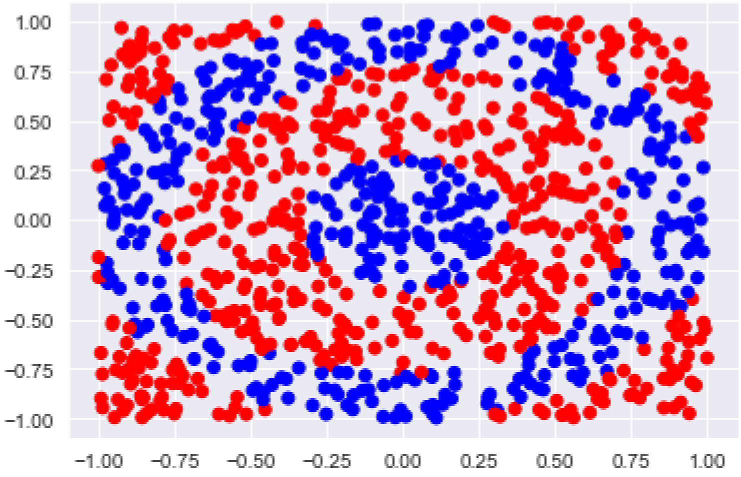


**Figure 15**

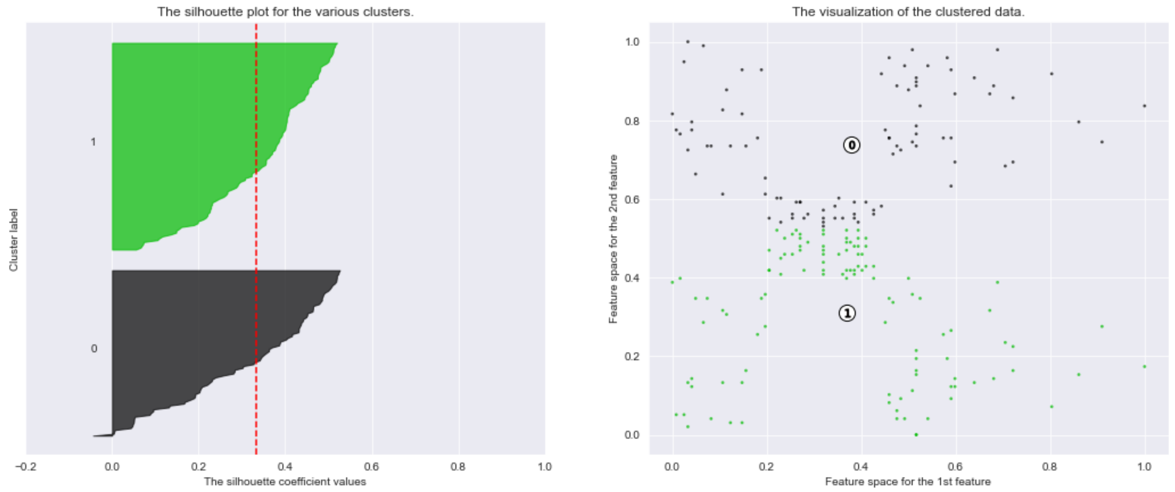
In the kernel's feature space the algorithm searches for the smallest sphere that encloses the image of the data using the Support Vector Domain Description algorithm.

This sphere, when mapped back to data space, forms a set of contours which enclose the data points as shown in the above figure.

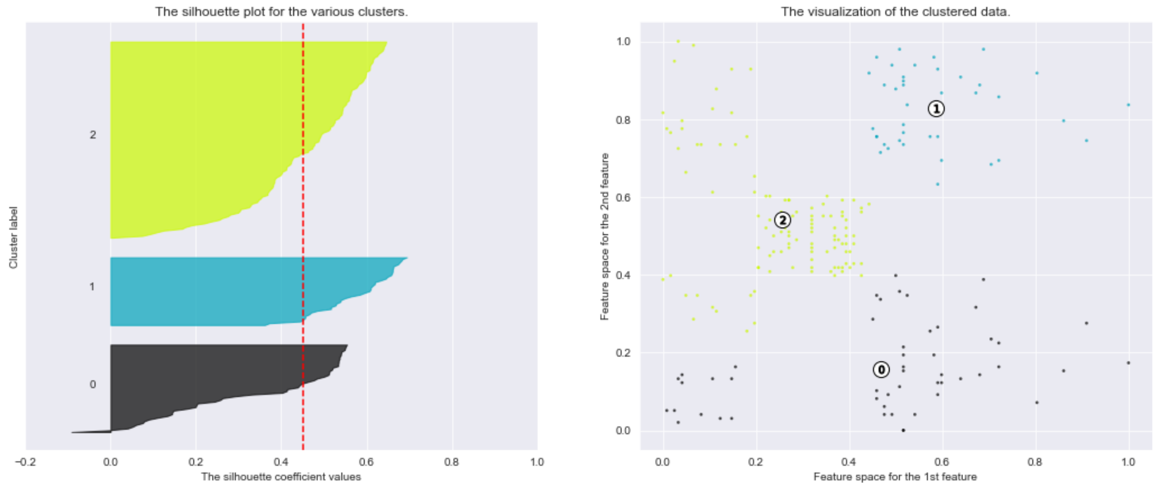
Those contours are then interpreted as cluster boundaries, and points enclosed by each contour are associated by SVC to the same cluster.

# Appendix:

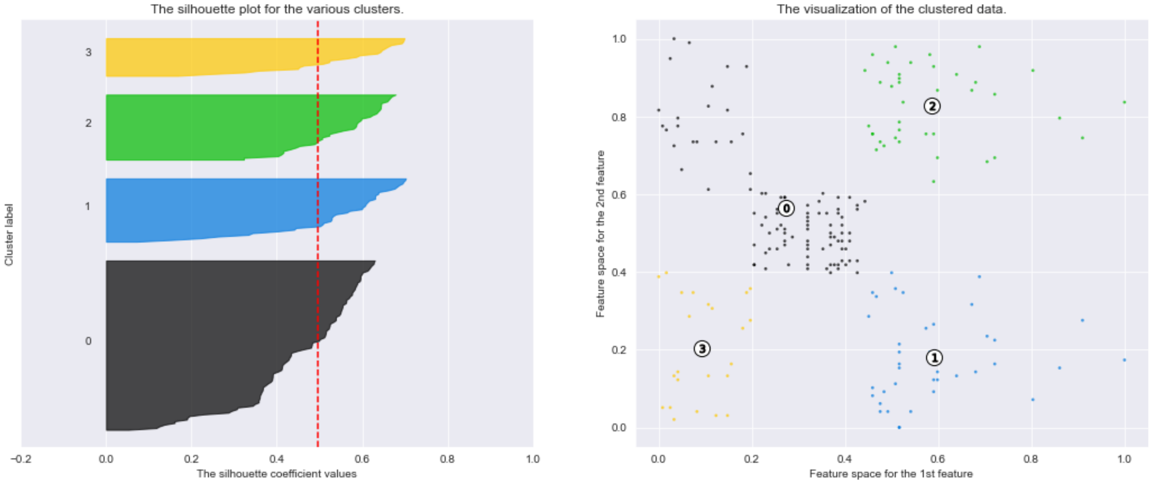
Silhouette analysis for KMeans clustering on sample data with n_clusters = 2



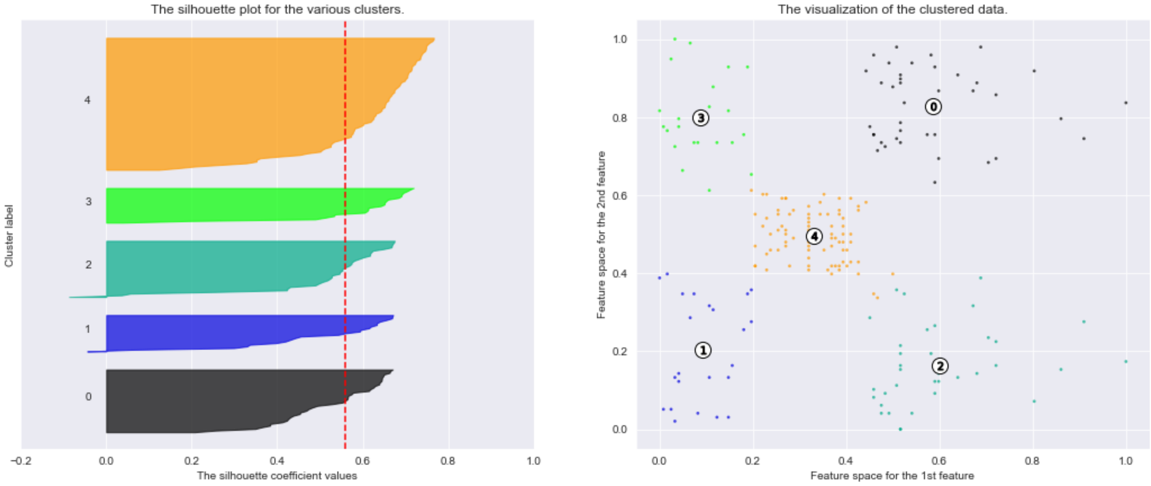
Silhouette analysis for KMeans clustering on sample data with n_clusters = 3



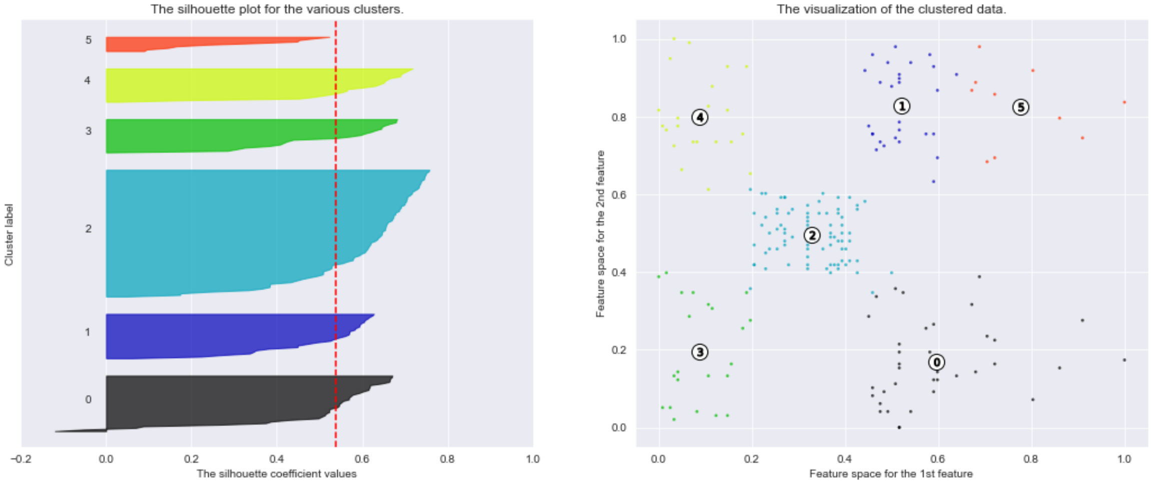
**Silhouette analysis for KMeans clustering on sample data with n_clusters = 4**



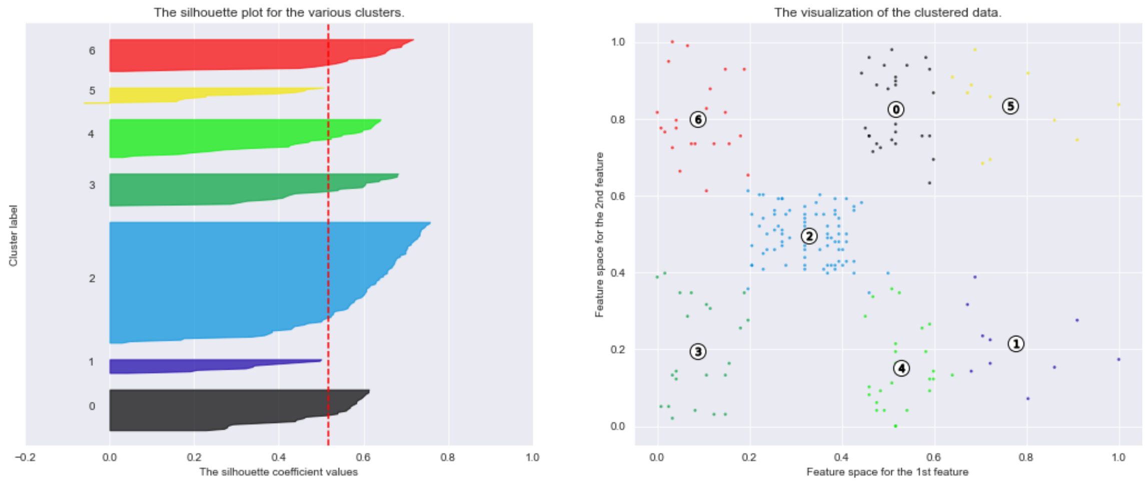
**Silhouette analysis for KMeans clustering on sample data with n_clusters = 5**



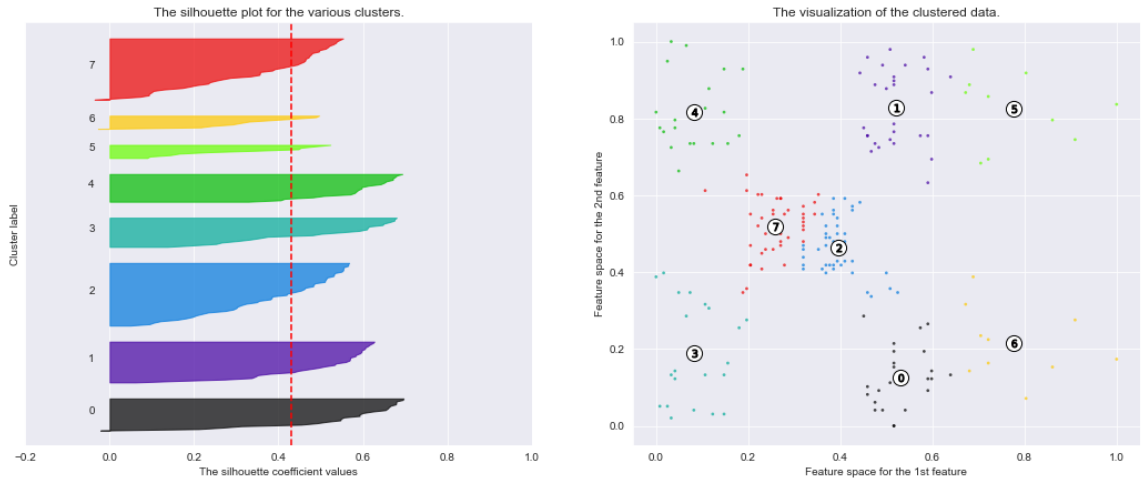
**Silhouette analysis for KMeans clustering on sample data with n_clusters = 6**



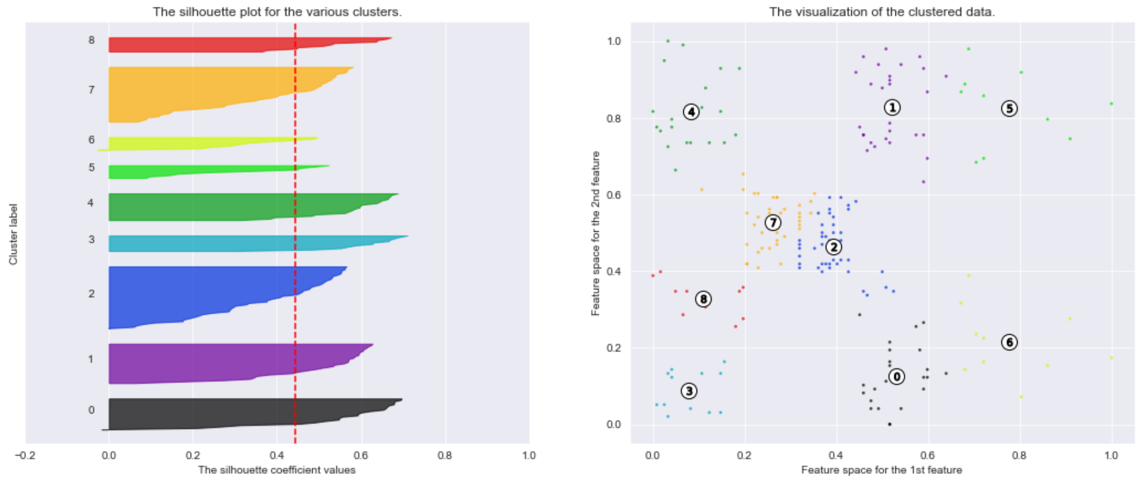
**Silhouette analysis for KMeans clustering on sample data with n_clusters = 7**

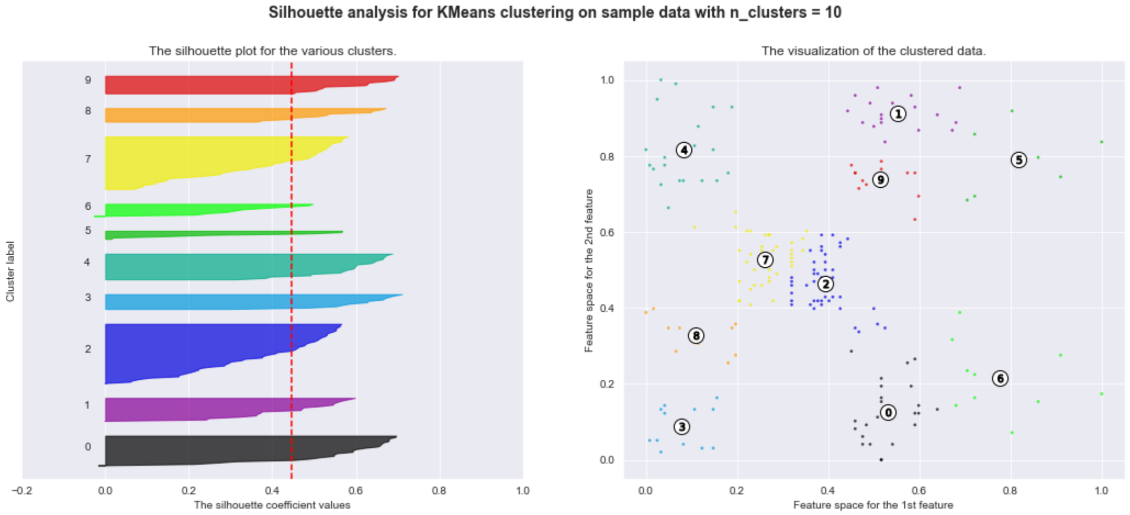


**Silhouette analysis for KMeans clustering on sample data with n_clusters = 8**



**Silhouette analysis for KMeans clustering on sample data with n_clusters = 9**




Silhouette analysis for KMeans clustering on sample data with n_clusters = 10
The silhouette plot for the various clusters.
Cluster label
9
8
7
6
5
4
3
2
1
0
−0.2
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values
The visualization of the clustered data.
Feature space for the 2nd feature
1.0
0.8
0.6
0.4
0.2
0.0
1
4
5
9
7
2
8
6
0
3
0.0
0.2
0.4
0.6
0.8
1.0
Feature space for the 1st feature