CENG 499 - Introduction to Machine Learning
Fall 2022
Homework 2

Necat Kılıçarslan

December 2022

1 PART 1

1.1 Knn Experiment

In this part, I created 9 different configurations for my model and test each of them. The different model parameters and their results are given below.

- Confidence interval for MinkowskiDistance K=5: 93.466 +/- 1.222
- Confidence interval for MinkowskiDistance K=10: 95.066 +/- 0.326
- Confidence interval for MinkowskiDistance K=30: 93.333 +/- 0.421
- Confidence interval for MahalanobisDistance K=5: 89.333 +/- 1.0327
- Confidence interval for MahalanobisDistance K=10: 86.8 +/- 2.0396
- Confidence interval for MahalanobisDistance K=30: 83.2 +/- 1.146
- Confidence interval for CosineDistance K=5: 93.866 +/- 0.266
- Confidence interval for CosineDistance K=10: 95.466 +/- 0.498
- Confidence interval for CosineDistance K=30: 94.533 +/- 0.653

The best classifier is 8 $->$ Confidence interval for CosineDistance K=10: 95.466 +/- 0.498

Since the 8th model means is the highest, this model has the best-performing hyperparameter values, The best model is using the CosineDistance function and the K value is 10.

2 PART 1

2.1 Kmeans Experiment

In this part, I created 9 different configurations according to the homework pdf for each dataset. Then, I tried all of the configurations. Here is the result of the experiment.

2.1.1 Dataset 1

- K = 2 dataset1, confidence interval: 157.99416732788086 +- 0.0
- K = 3 dataset1, confidence interval: 73.47527694702148 +- 0.0
- K = 4 dataset1, confidence interval: 33.955621004104614 +- 0.0
- K = 5 dataset1, confidence interval: 10.893421173095703 +- 0.0
- K = 6 dataset1, confidence interval: 9.990493834018707 +- 0.0
- K = 7 dataset1, confidence interval: 9.254283118247987 +- 0.00048756325584260507
- K = 8 dataset1, confidence interval: 8.518688905239106 +- 0.005241351363521859
- K = 9 dataset1, confidence interval: 7.821717211604119 +- 0.003626174480241597
- K = 10 dataset1, confidence interval: 7.176218894124031 +- 0.05157818699540444

According to the elbow method from figure 1 we can easily see that K = 5 which is the best number of clusters.

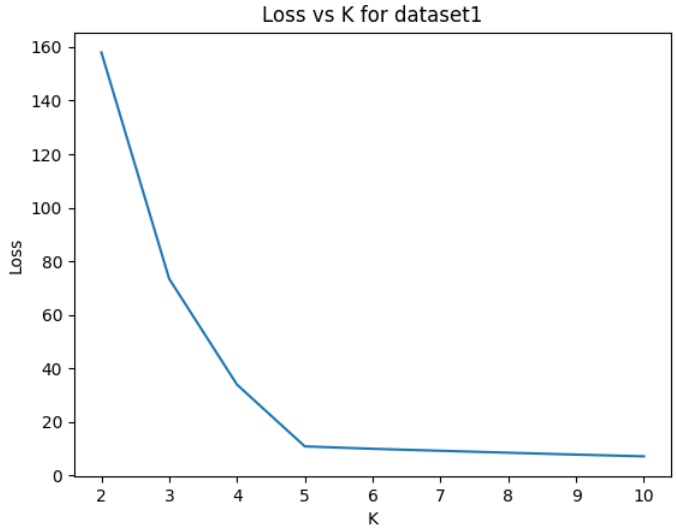


Figure 1: Configurations and their confidence intervals

2.1.2 Dataset 2

- K = 2 dataset2, confidence interval: 130.6921501159668 +- 0.0
- K = 3 dataset2, confidence interval: 51.02644729614258 +- 0.0
- K = 4 dataset2, confidence interval: 23.652430534362793 +- 0.0
- K = 5 dataset2, confidence interval: 22.69369034767151 +- 0.000732454466185045
- K = 6 dataset2, confidence interval: 21.747010707855225 +- 0.0
- K = 7 dataset2, confidence interval: 20.870836567878722 +- 0.029065904869613322
- K = 8 dataset2, confidence interval: 20.043905091285705 +- 0.009791316027201123
- K = 9 dataset2, confidence interval: 19.384672510623933 +- 0.01495733347952498
- K = 10 dataset2, confidence interval: 18.849976086616515 +- 0.08699568828694135

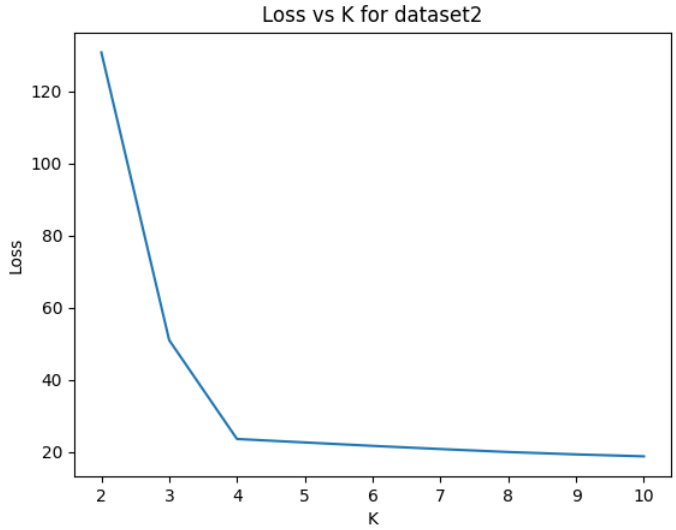


Figure 2: Configurations and their confidence intervals

According to the elbow method from figure 2, we can easily see that K = 4 which is the best number of clusters.

Now let's analyze the time complexity for the means. Firstly, my code run l times which is the actual total number of count to the data that will be converged. This is the outermost loop. Inside this loop, I iterated over the dataset count which is the number of counts in the dataset and then I iterated over the number of K times, and then inside this for loop I am calculating the distance. Therefore if we increase the number of points the time complexity will be increased sharply, also inside this loop, I iterated over the number of K values, if the K values increases, the time complexity will be increased. In addition to that information, if we increase the data sample vector dimension, the distance calculation is will be much longer then the time complexity affected highly. Also, if we increase the number of iterations the whole calculations will be computed more times, thus the time complexity will be increased.

2.2 Kmeans Plus Plus Experiment

Again for this part, I created 9 different models for each dataset. Here is the result of the Kmeans Plus Plus experiment.

2.2.1 Dataset 1

- K = 2 dataset1, confidence interval: 157.99416732788086 +- 0.0
- K = 3 dataset1, confidence interval: 73.47527694702148 +- 0.0
- K = 4 dataset1, confidence interval: 33.955621004104614 +- 0.0
- K = 5 dataset1, confidence interval: 10.893421173095703 +- 0.0
- K = 6 dataset1, confidence interval: 9.990493834018707 +- 0.0
- K = 7 dataset1, confidence interval: 9.25377243757248 +- 0.000998373429716879
- K = 8 dataset1, confidence interval: 8.520891380310058 +- 0.004787525884272671
- K = 9 dataset1, confidence interval: 7.793565797805786 +- 0.005336477856760529
- K = 10 dataset1, confidence interval: 7.059733808040619 +- 0.0

kmeans_plusplus_dataset1.png.png

Figure 3: Configurations and their confidence intervals

According to the elbow method from figure 1 we can easily see that K = 5 which is the best number of clusters.

2.2.2 Dataset 2

- K = 2 dataset2, confidence interval: 130.6921501159668 +- 0.0
- K = 3 dataset2, confidence interval: 51.02644729614258 +- 0.0
- K = 4 dataset2, confidence interval: 23.652430534362793 +- 0.0
- K = 5 dataset2, confidence interval: 22.699701976776122 +- 0.006698903316276858
- K = 6 dataset2, confidence interval: 21.79466621875763 +- 0.006783107837539607
- K = 7 dataset2, confidence interval: 20.84448652267456 +- 0.00034658765333536805
- K = 8 dataset2, confidence interval: 20.008964824676514 +- 0.05991631202268079
- K = 9 dataset2, confidence interval: 19.362406659126282 +- 0.021019716639909477
- K = 10 dataset2, confidence interval: 18.77463014125824 +- 0.03580286981021955

According to the elbow method from figure 2, we can easily see that K = 4 which is the best number of clusters.

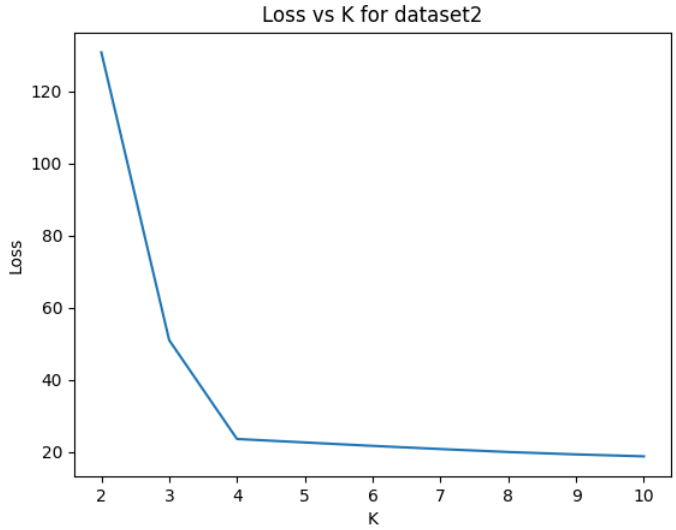


Figure 4: Configurations and their confidence intervals

3 PART 3

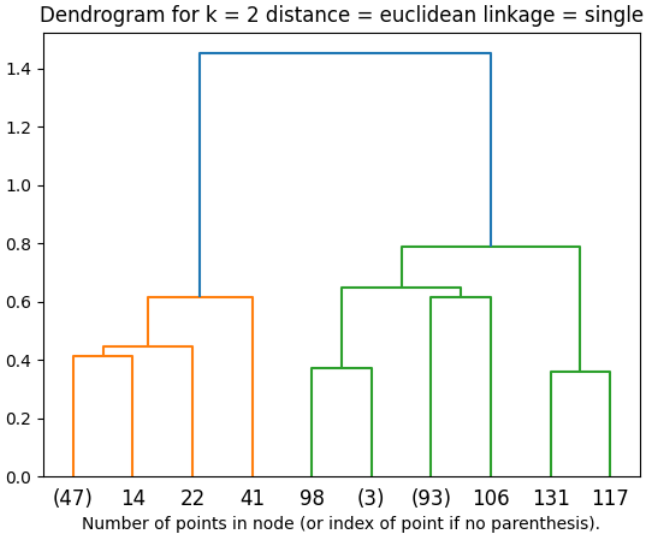
3.1 Result Part

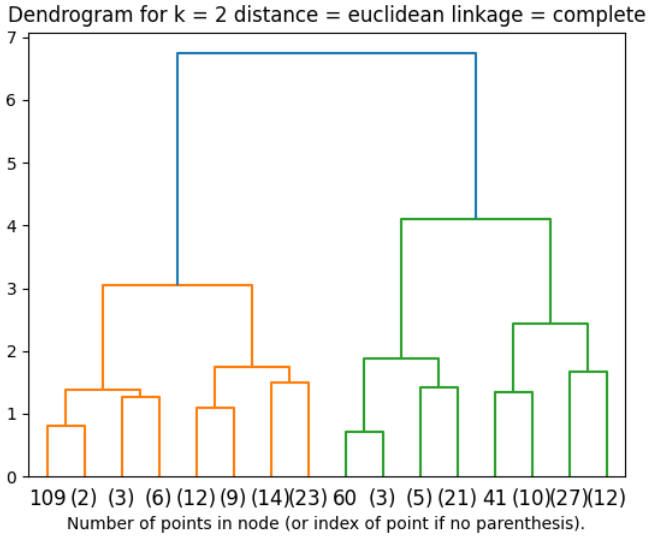
Firstly I want to share my average silhouette score for each configuration.

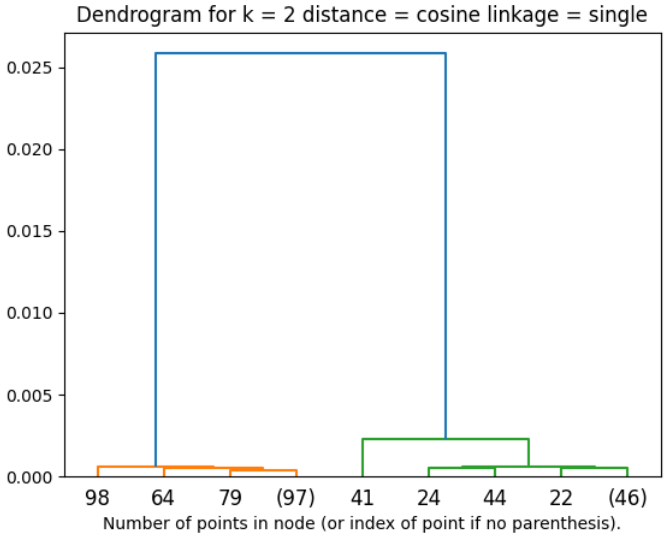
- For 2 clusters, euclidean distance and single linkage The average silhouette score is: 0.68810517
- For 2 clusters, euclidean distance and complete linkage The average silhouette score is: 0.5023174
- For 2 clusters, cosine distance and single linkage The average silhouette score is: 0.68810517
- For 2 clusters, cosine distance and complete linkage The average silhouette score is: 0.68810517
- For 3 clusters, euclidean distance and single linkage The average silhouette score is: 0.53133893
- For 3 clusters, euclidean distance and complete linkage The average silhouette score is: 0.5191346
- For 3 clusters, cosine distance and single linkage The average silhouette score is: 0.56061506
- For 3 clusters, cosine distance and complete linkage The average silhouette score is: 0.3834758
- For 4 clusters, euclidean distance and single linkage The average silhouette score is: 0.39444217
- For 4 clusters, euclidean distance and complete linkage The average silhouette score is: 0.51993114
- For 4 clusters, cosine distance and single linkage The average silhouette score is: 0.376524
- For 4 clusters, cosine distance and complete linkage The average silhouette score is: 0.15615867
- For 5 clusters, euclidean distance and single linkage The average silhouette score is: 0.31220323
- For 5 clusters, euclidean distance and complete linkage The average silhouette score is: 0.3702072
- For 5 clusters, cosine distance and single linkage The average silhouette score is: 0.2478568
- For 5 clusters, cosine distance and complete linkage The average silhouette score is: 0.09422624

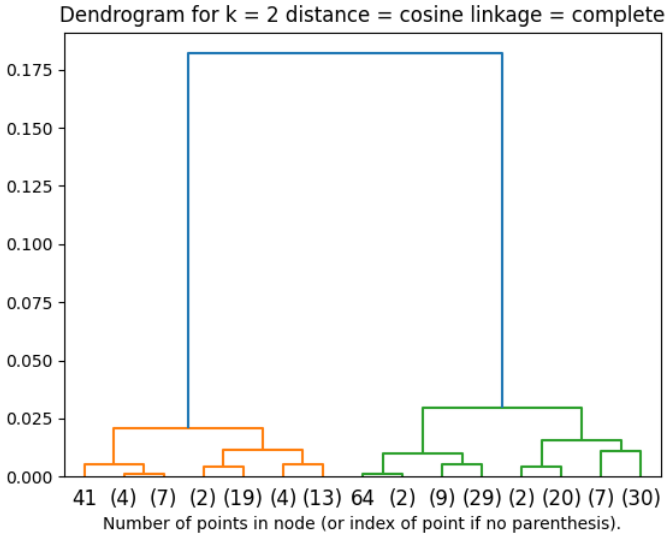
3.2 Dendrogram

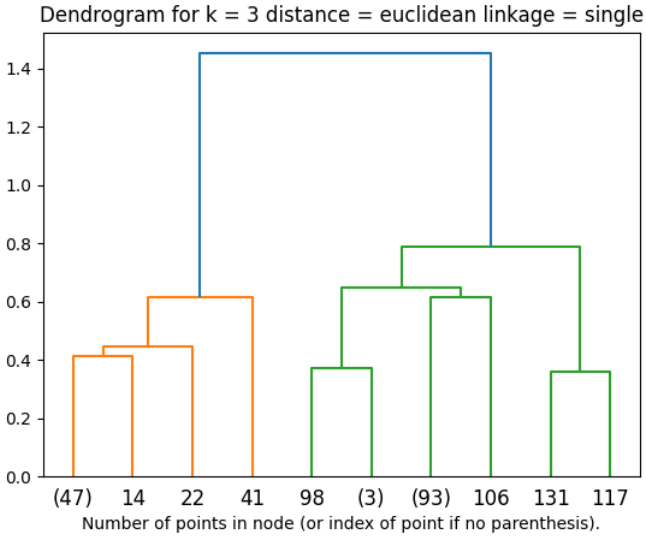
In the following, I put the dendrogram result for each given parameter value one by one.

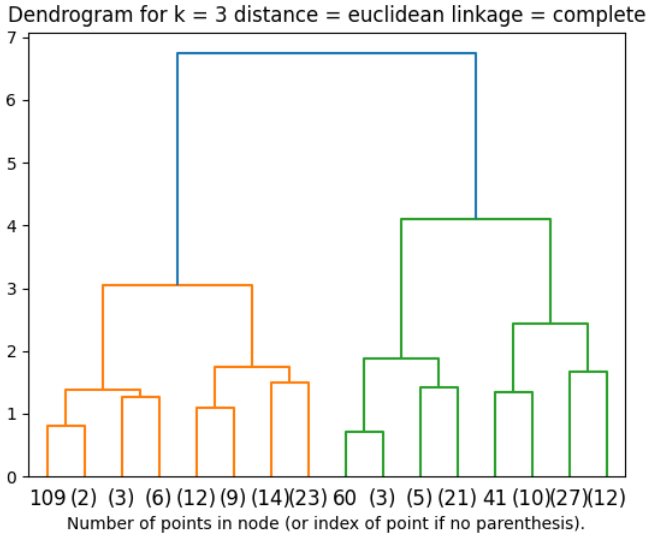

Dendrogram for k = 2 distance = euclidean linkage = single
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
(47) 14 22 41 98 (3) (93) 106 131 117
Number of points in node (or index of point if no parenthesis).

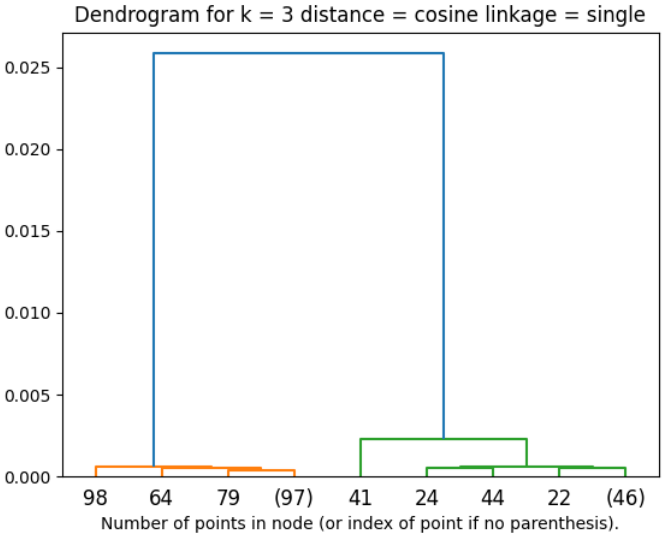

Dendrogram for k = 2 distance = euclidean linkage = complete
7
6
5
4
3
2
1
0
109 (2) (3) (6) (12) (9) (14) (23) 60 (3) (5) (21) 41 (10) (27) (12)
Number of points in node (or index of point if no parenthesis).

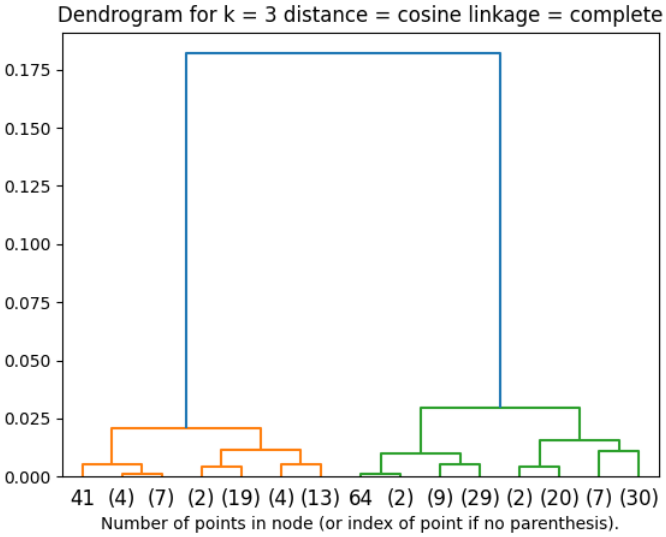

Dendrogram for k = 2 distance = cosine linkage = single
0.025
0.020
0.015
0.010
0.005
0.000
98 64 79 (97) 41 24 44 22 (46)
Number of points in node (or index of point if no parenthesis).

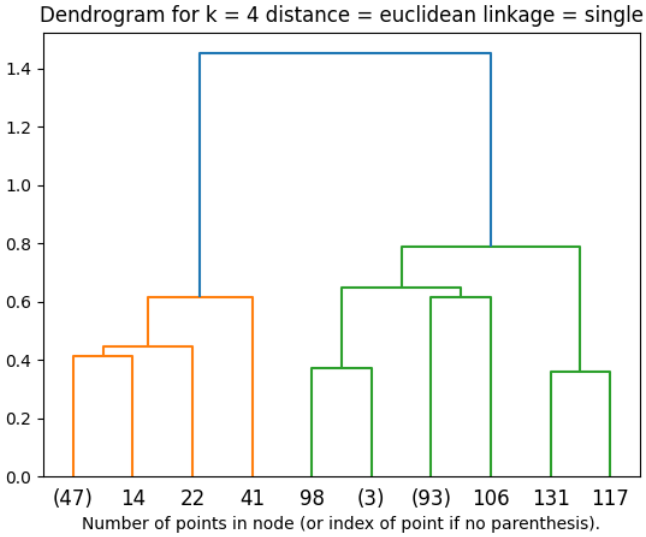

Dendrogram for k = 2 distance = cosine linkage = complete
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
41 (4) (7) (2) (19) (4) (13) 64 (2) (9) (29) (2) (20) (7) (30)
Number of points in node (or index of point if no parenthesis).

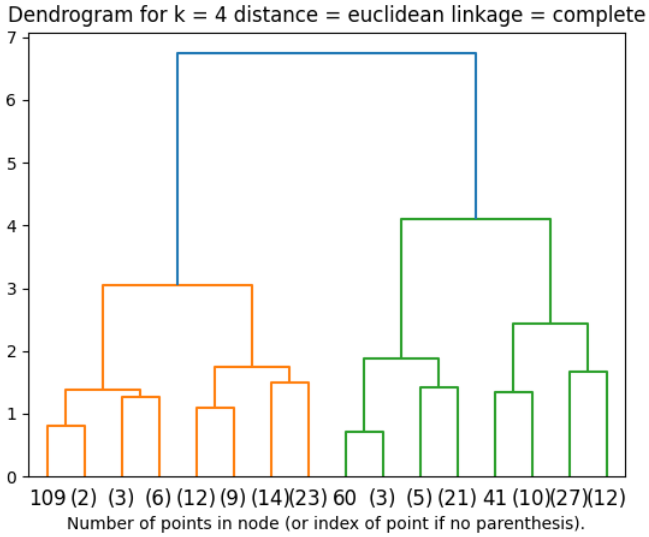

Dendrogram for k = 3 distance = euclidean linkage = single
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
(47) 14 22 41 98 (3) (93) 106 131 117
Number of points in node (or index of point if no parenthesis).

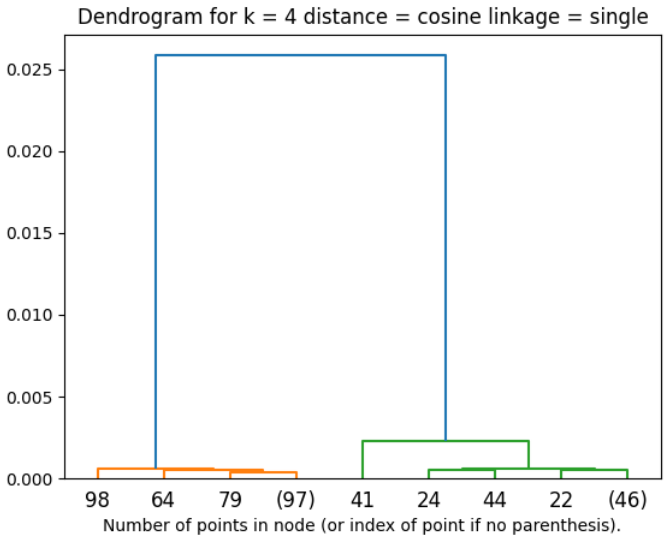

Dendrogram for k = 3 distance = euclidean linkage = complete
7
6
5
4
3
2
1
0
109 (2) (3) (6) (12) (9) (14) (23) 60 (3) (5) (21) 41 (10) (27) (12)
Number of points in node (or index of point if no parenthesis).

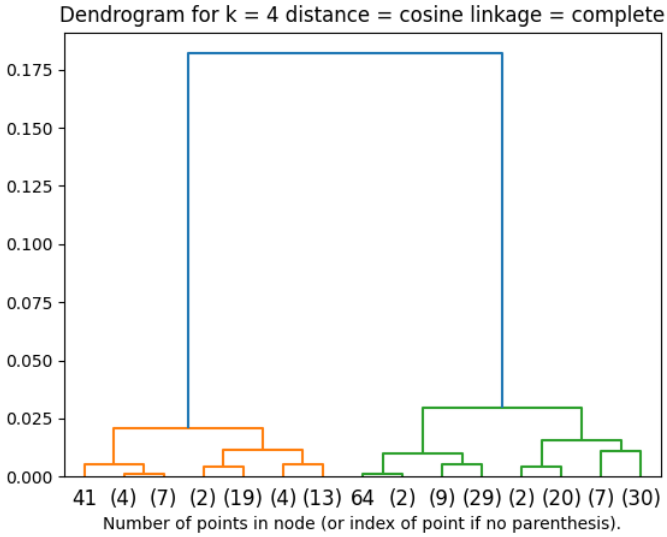

Dendrogram for k = 3 distance = cosine linkage = single
0.025
0.020
0.015
0.010
0.005
0.000
98 64 79 (97) 41 24 44 22 (46)
Number of points in node (or index of point if no parenthesis).

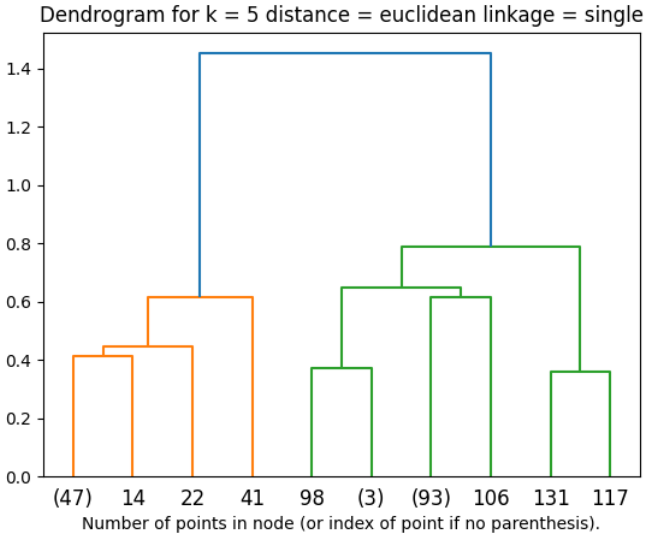

Dendrogram for k = 3 distance = cosine linkage = complete
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
41 (4) (7) (2) (19) (4) (13) 64 (2) (9) (29) (2) (20) (7) (30)
Number of points in node (or index of point if no parenthesis).

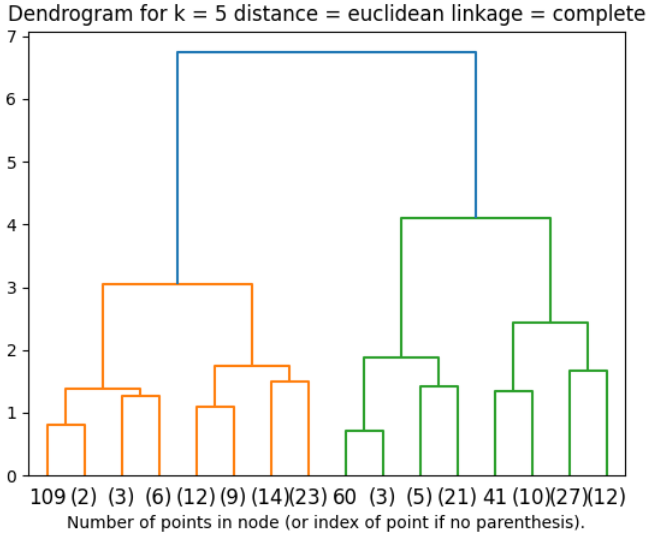

Dendrogram for k = 4 distance = euclidean linkage = single
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
(47) 14 22 41 98 (3) (93) 106 131 117
Number of points in node (or index of point if no parenthesis).

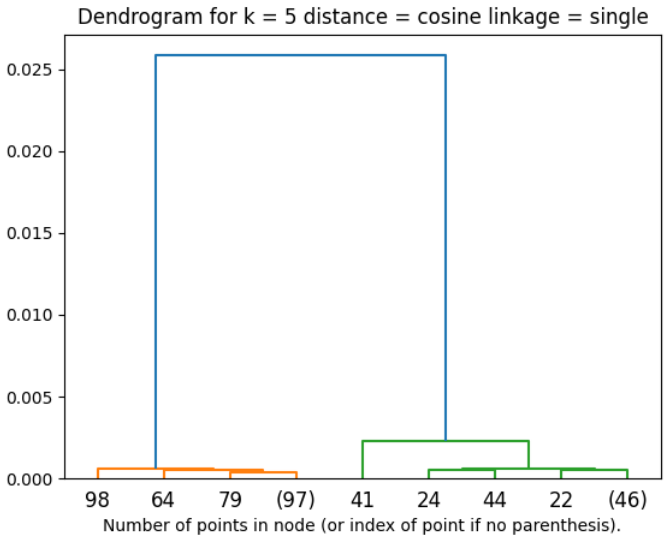

Dendrogram for k = 4 distance = euclidean linkage = complete
7
6
5
4
3
2
1
0
109 (2) (3) (6) (12) (9) (14) (23) 60 (3) (5) (21) 41 (10) (27) (12)
Number of points in node (or index of point if no parenthesis).

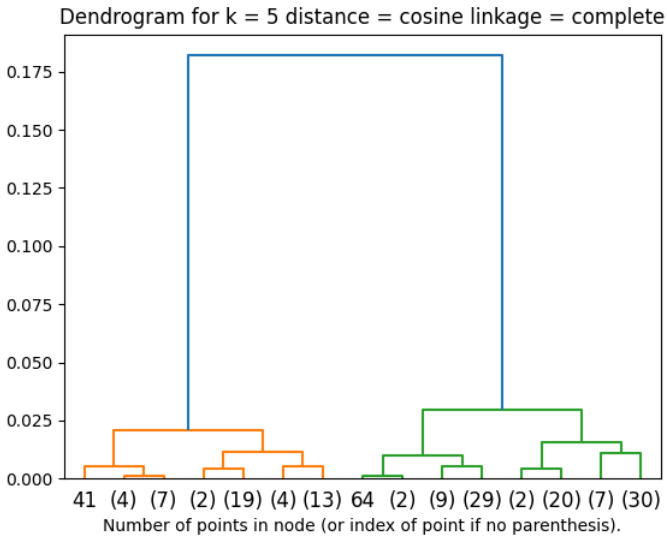

Dendrogram for k = 4 distance = cosine linkage = single
0.025
0.020
0.015
0.010
0.005
0.000
98 64 79 (97) 41 24 44 22 (46)
Number of points in node (or index of point if no parenthesis).


Dendrogram for k = 4 distance = cosine linkage = complete
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
41 (4) (7) (2) (19) (4) (13) 64 (2) (9) (29) (2) (20) (7) (30)
Number of points in node (or index of point if no parenthesis).


Dendrogram for k = 5 distance = euclidean linkage = single
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
(47) 14 22 41 98 (3) (93) 106 131 117
Number of points in node (or index of point if no parenthesis).

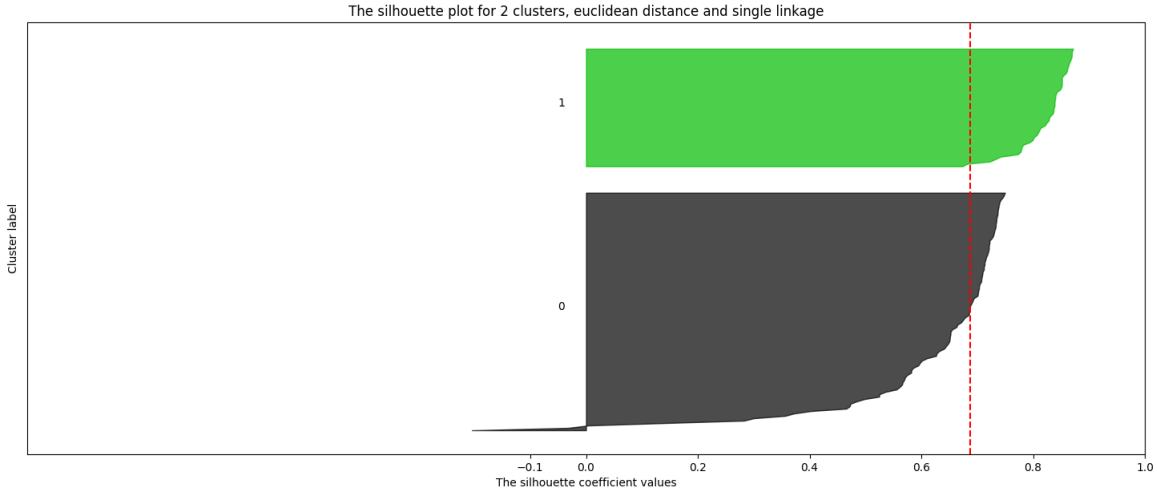

Dendrogram for k = 5 distance = euclidean linkage = complete
7
6
5
4
3
2
1
0
109 (2) (3) (6) (12) (9) (14) (23) 60 (3) (5) (21) 41 (10) (27) (12)
Number of points in node (or index of point if no parenthesis).

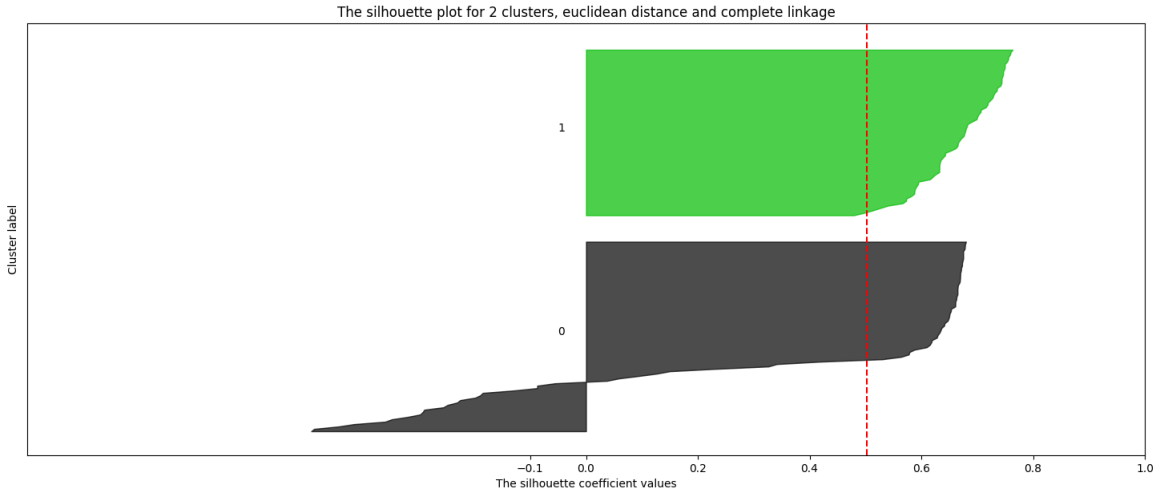


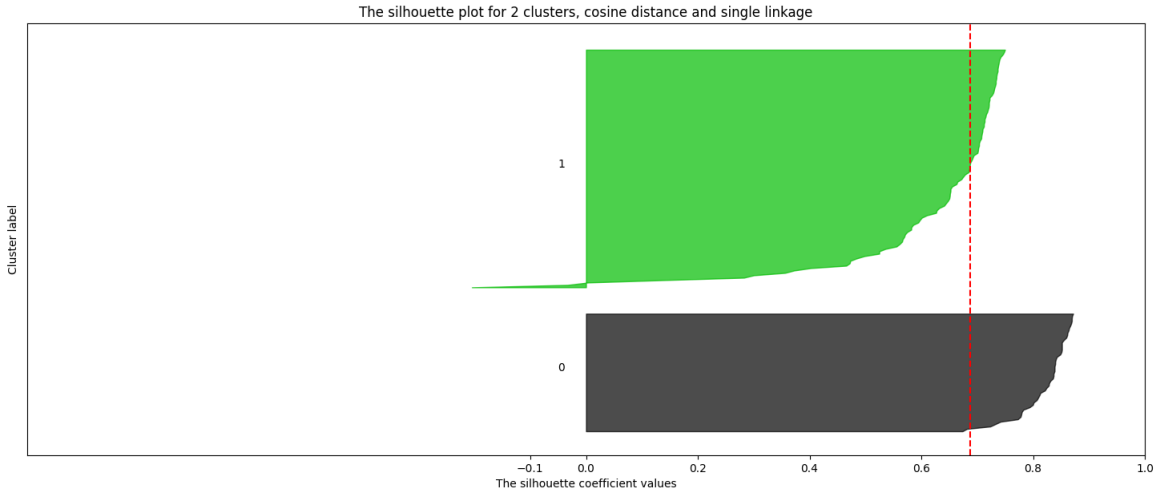


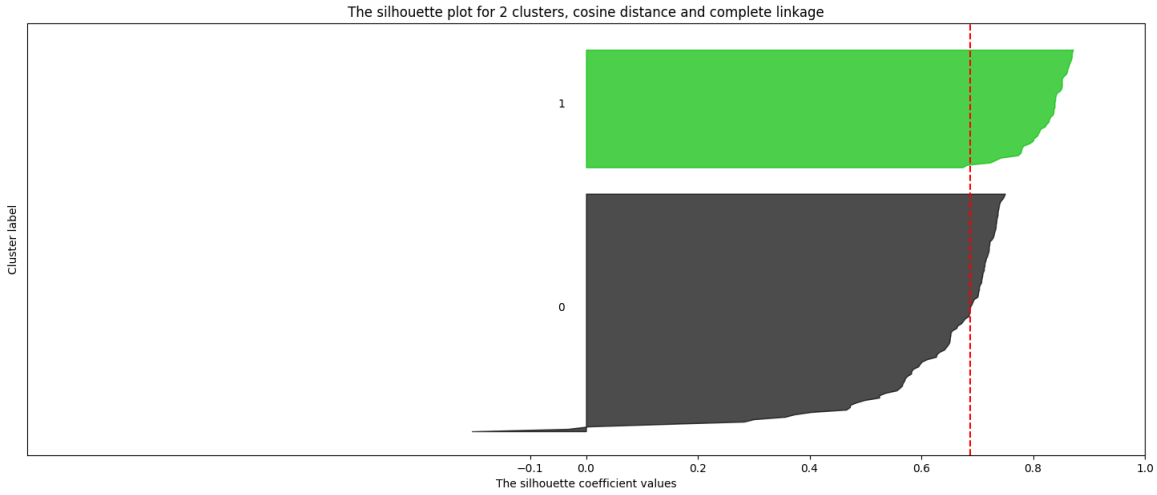
3.3 Silhouette

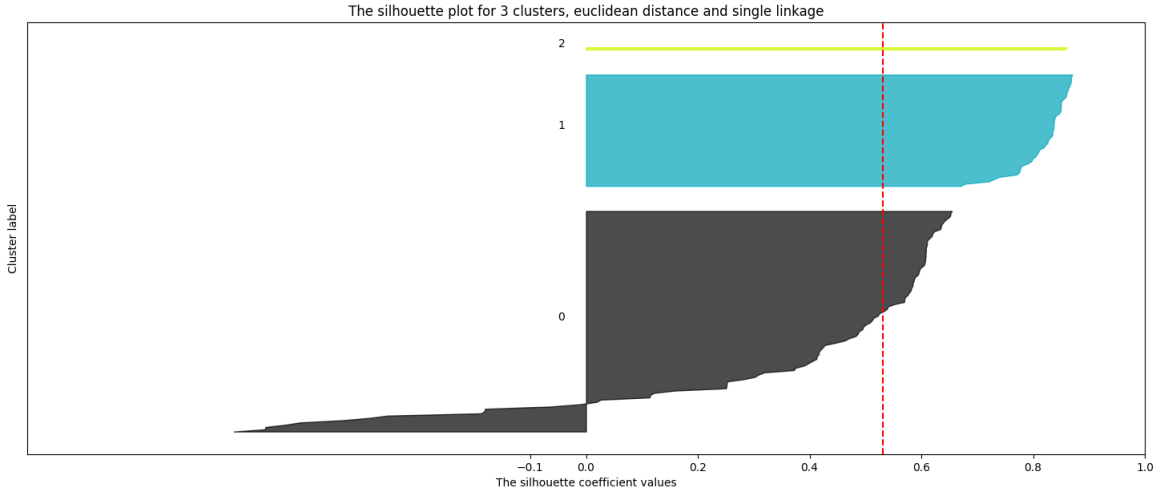
In the following, I put the silhouette result for each given parameter value one by one.

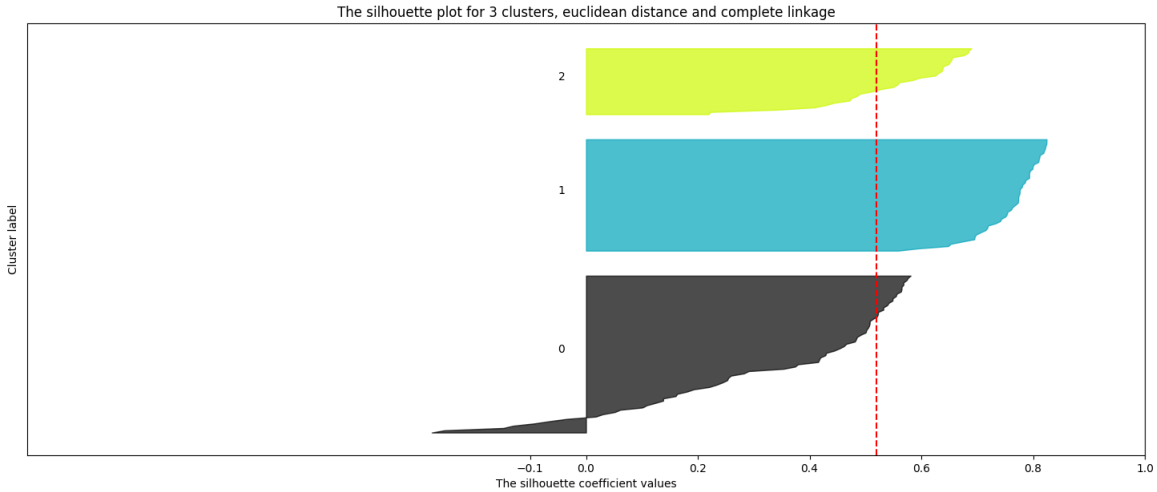

The silhouette plot for 2 clusters, euclidean distance and single linkage
Cluster label
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values

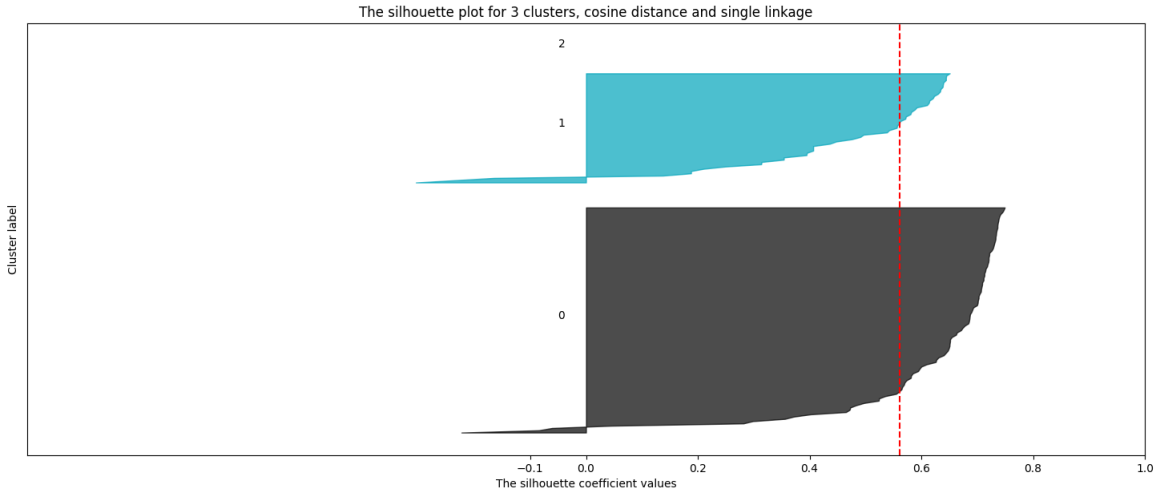

The silhouette plot for 2 clusters, euclidean distance and complete linkage
Cluster label
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values

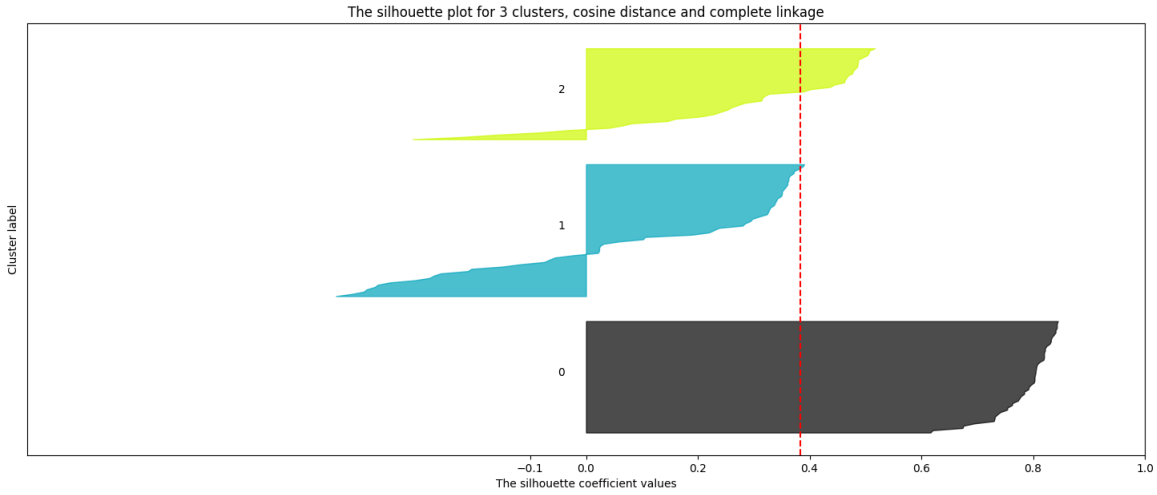

The silhouette plot for 2 clusters, cosine distance and single linkage
Cluster label
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values

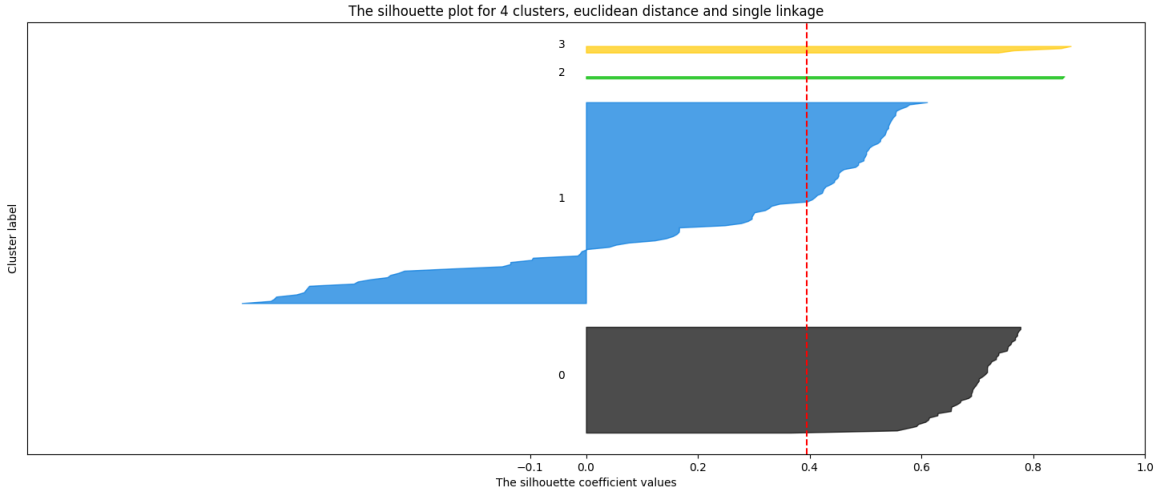

The silhouette plot for 2 clusters, cosine distance and complete linkage
Cluster label
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values

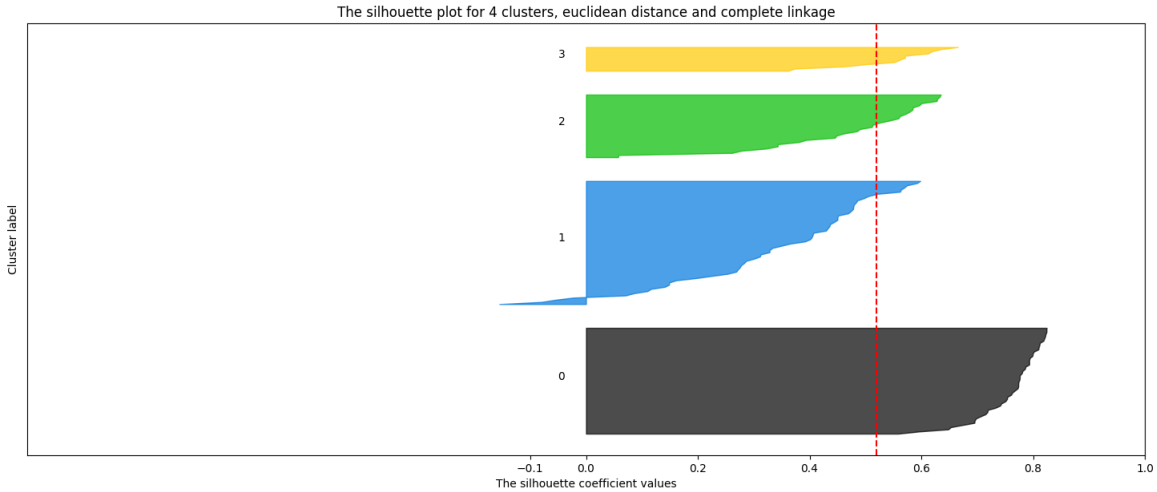

The silhouette plot for 3 clusters, euclidean distance and single linkage
Cluster label
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values

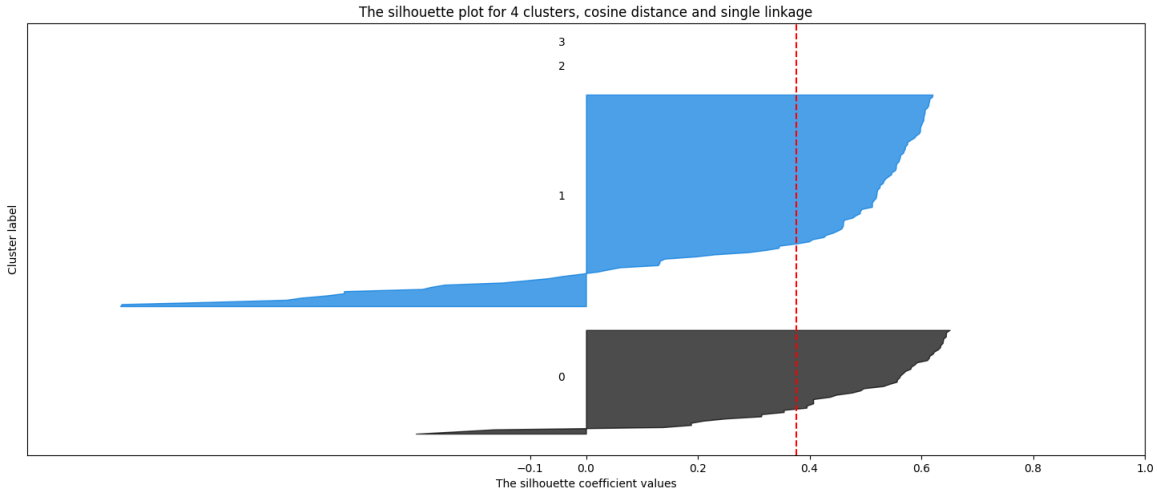

The silhouette plot for 3 clusters, euclidean distance and complete linkage
Cluster label
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values

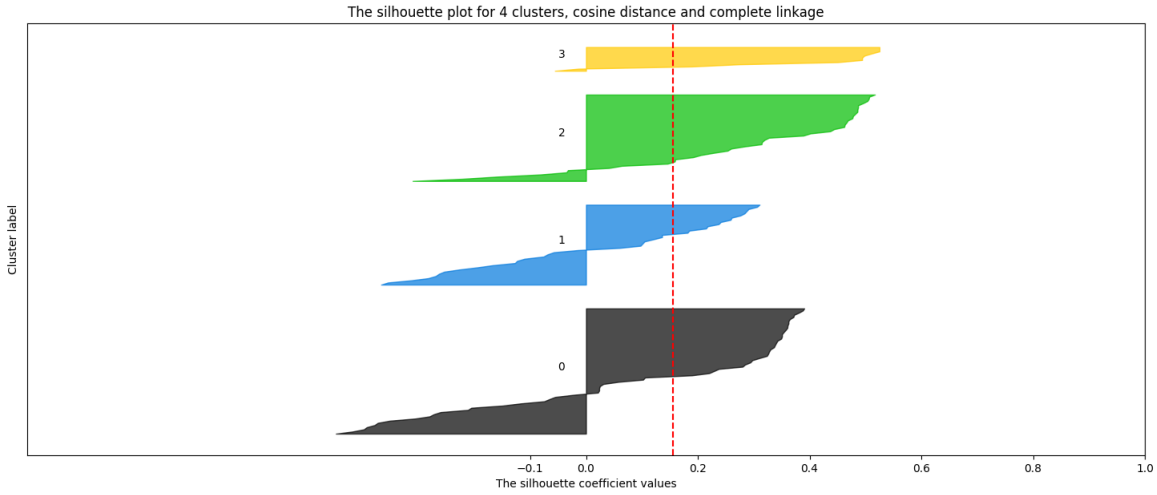

The silhouette plot for 3 clusters, cosine distance and single linkage
Cluster label
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values

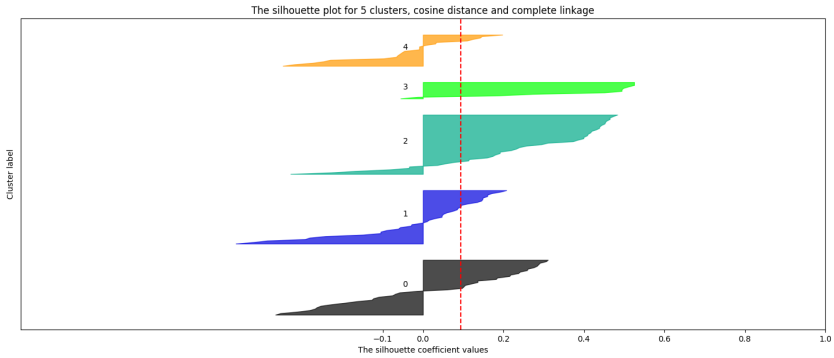

The silhouette plot for 3 clusters, cosine distance and complete linkage
Cluster label
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values

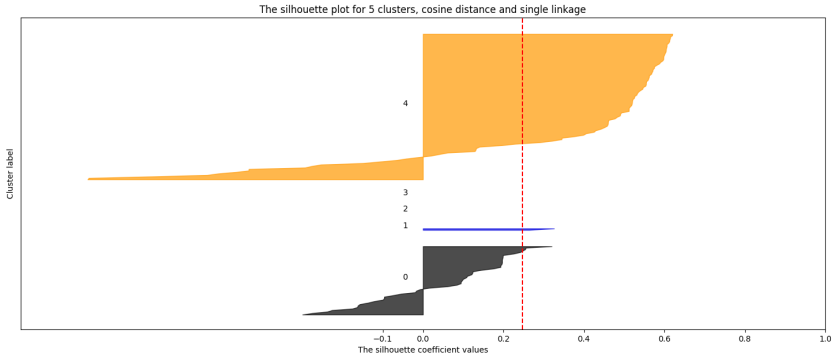

The silhouette plot for 4 clusters, euclidean distance and single linkage
Cluster label
3
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values

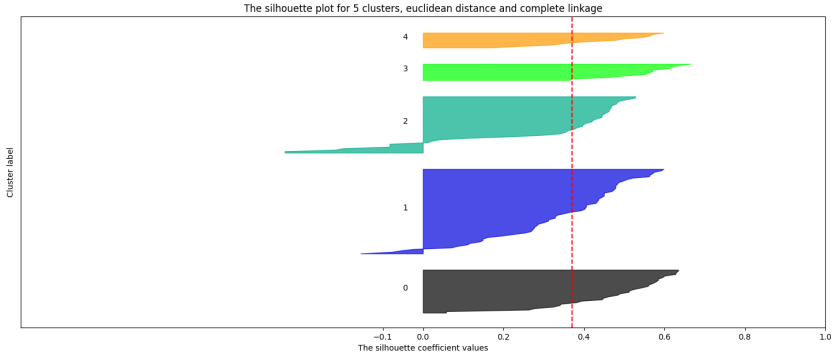

The silhouette plot for 4 clusters, euclidean distance and complete linkage
Cluster label
3
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values


The silhouette plot for 4 clusters, cosine distance and single linkage
Cluster label
3
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values


The silhouette plot for 4 clusters, cosine distance and complete linkage
Cluster label
3
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values





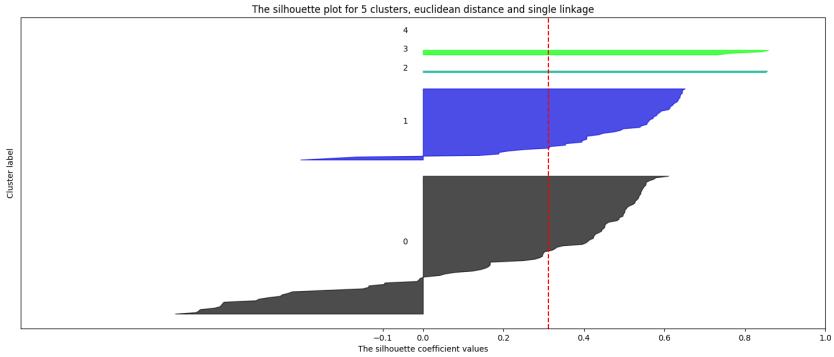


3.4 Comments

- For 2 clusters, euclidean distance and single linkage The average silhouette score is: 0.68810517
- For 2 clusters, cosine distance and single linkage The average silhouette score is: 0.68810517
- For 2 clusters, cosine distance and complete linkage The average silhouette score is: 0.68810517

These are the best silhouette scores. If we look at the dendrograms of these parameters we can easily see that from the figures. Since these values are on the right of the ”0” lines with compared to other modeles, we can understand them these are the best parameters.

Since the complexity is highly related to number of data points if we increased the number of data points the complexity of the whole system will increase. Moreover again distance with related to the number of dimensions, if we increase the dimension number, the complexity of the system will be highly affected. I would choose HAC with a dataset consisting of 1 million data points each of which has a dimension of 120000. Because kmeans depend on the initial cluster centers which are randomly generated because of that reasons sometimes the convergence can be too long depending on the initial data points. In other words, in kmeans if we choose a bad initial value the time need to converge can be too long. We can conclude that k means works in a non-deterministic way. In order to avoid this situation happening, it would be a good choice to use HAC in this experiment.


The silhouette plot for 5 clusters, euclidean distance and single linkage
Cluster label
4
3
2
1
0
−0.1
0.0
0.2
0.4
0.6
0.8
1.0
The silhouette coefficient values