Clustering

Expectation-Maximization Algorithm

Expectation-Maximization Algorithm

- Real-world applications of machine learning
  - there are many relevant features available for learning but only a small subset of them are observable.

- If the variables are observable - use the instances for the purpose of learning
- when it is not observable, then we can predict its value in the instances.

Expectation-Maximization Algorithm

- Expectation-Maximization algorithm can be used for the latent variables

- Latent variables - Variables that are not directly observable and are actually inferred from the values of the other observed variables too

- To predict their values, we can use probability distribution.

Expectation-Maximization Algorithm

This algorithm is base for many unsupervised clustering algorithms.

It was proposed in 1977 by Dempster, Nan Laird, and Donald Rubin.

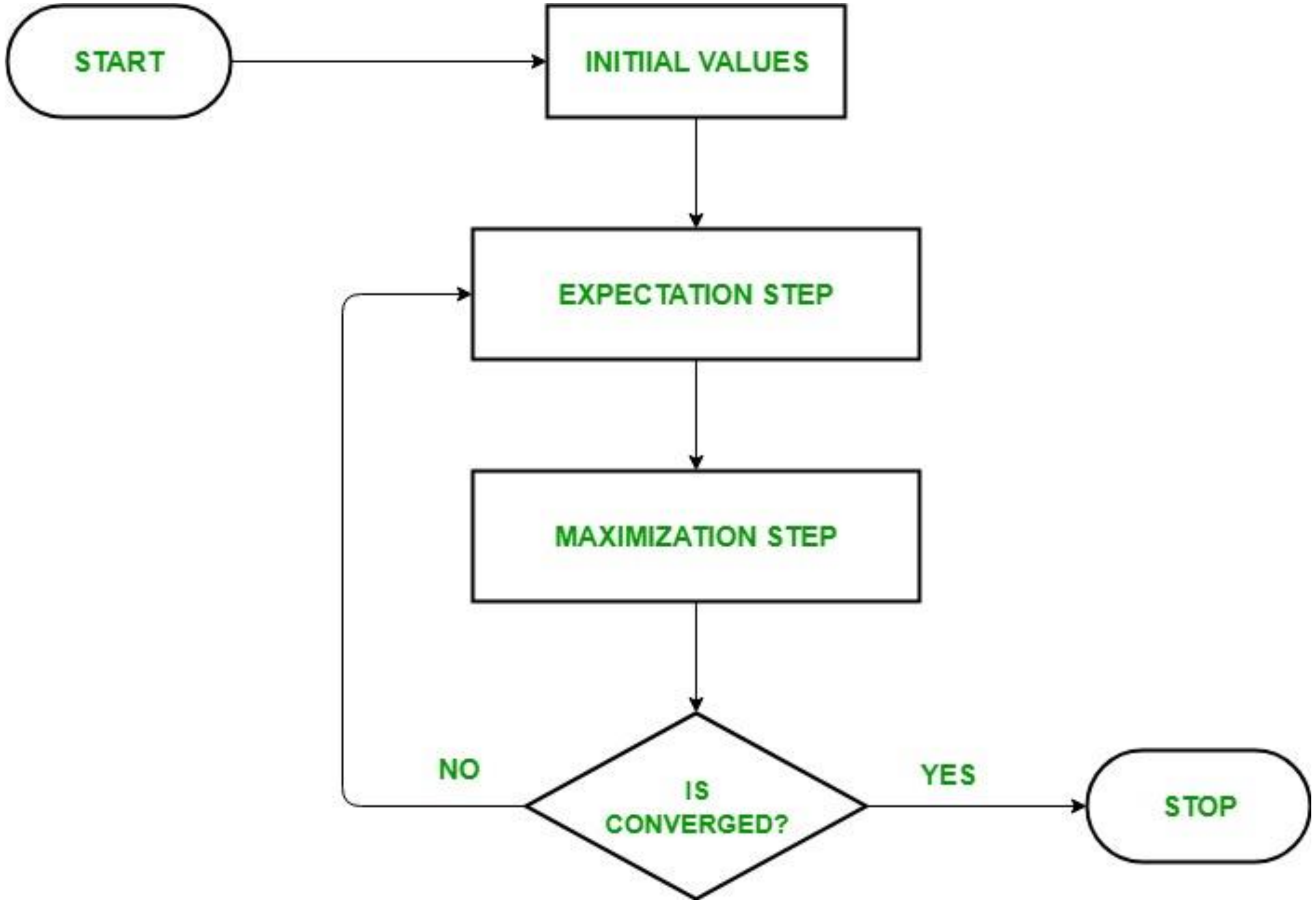
It is used to find the local maximum likelihood parameters in the cases where latent variables are involved, and the data is missing or incomplete.

Algorithm

1. Given a set of incomplete data, consider a set of initial parameters.
2. Expectation step (E – step):
    1. Using the observed available data of the dataset, estimate (guess) the values of the missing data.
    2. The algorithm computes the latent variables i.e. expectation of the log-likelihood using the current parameter estimates.
3. Maximization step (M – step):
    1. Complete data generated after the expectation (E) step is used in order to update the parameters.
    2. The algorithm determines the parameters that maximize the expected log-likelihood obtained in the E step, and corresponding model parameters are updated based on the estimated latent variables.
4. Repeat step 2 and step 3 until convergence.

Expectation-Maximization (EM) Algorithm Works:

- The Expectation-Maximization algorithm use the available observed data of the dataset to estimate the missing data and then use that data to update the values of the parameters.



Example

Let C1 and C2 be two coins.

Θ_1 be probability of getting head with C1
Θ_2 be probability of getting head with C2

Find values of Θ_1 and Θ_2 by tossing C1 and C2 for multiple times.

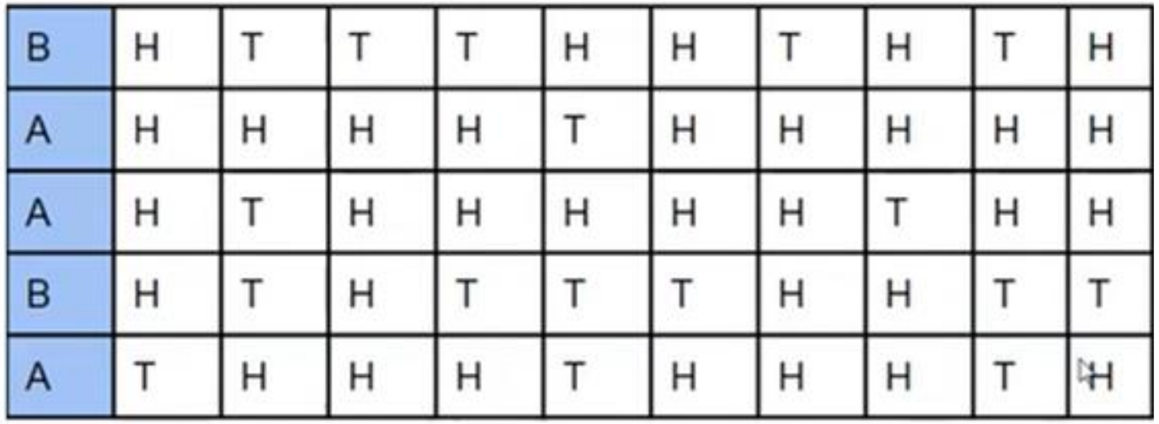
$$\Theta_1 = \frac{no.of\ heads\ with\ C1}{Total\ no.of\ flips\ using\ C1} \qquad \Theta_2 = \frac{no.of\ heads\ with\ C2}{Total\ no.of\ flips\ using\ C2}$$

Example

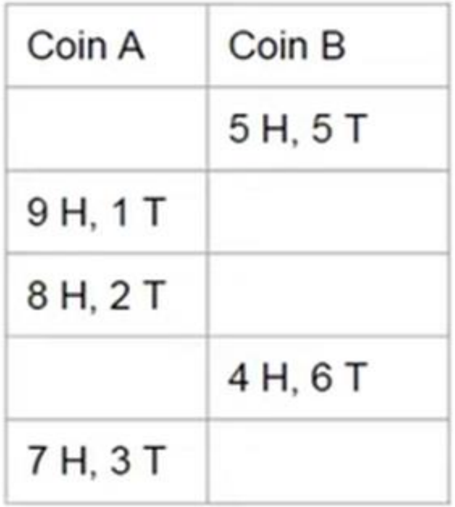
- Two coins A and B
- Choosing any of the coins randomly five times.
- Each selected coin must be tossed 10 times.

$$\Theta_1 = \frac{no.\, of\ heads\ with\ C1}{Total\ no.\, of\ flips\ using\ C1}$$

$$\Theta_2 = \frac{no.\, of\ heads\ with\ C2}{Total\ no.\, of\ flips\ using\ C2}$$



B	H	T	T	T	H	H	T	H	T	H
A	H	H	H	H	T	H	H	H	H	H
A	H	T	H	H	H	H	H	T	H	H
B	H	T	H	T	T	T	H	H	T	T
A	T	H	H	H	T	H	H	H	T	H



Coin A	Coin B
	5 H, 5 T
9 H, 1 T	
8 H, 2 T	
	4 H, 6 T
7 H, 3 T	

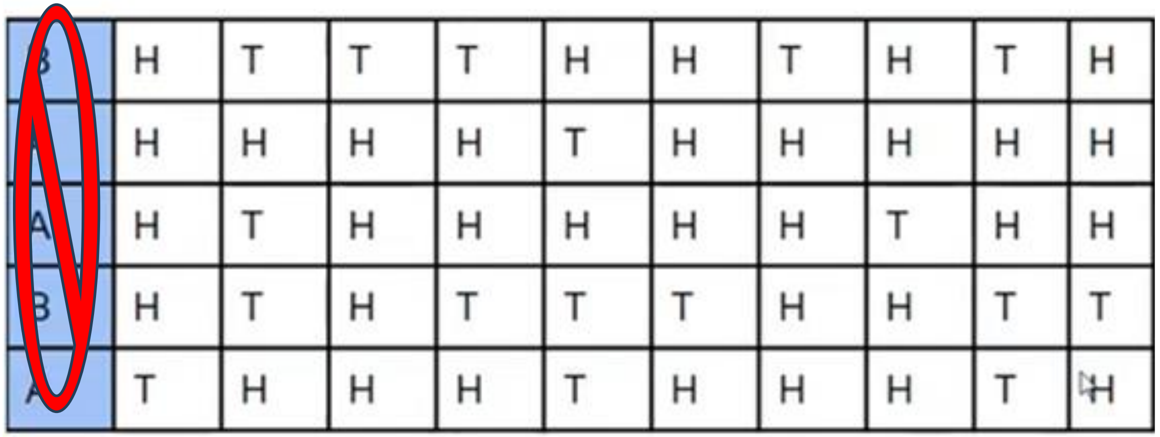
24H, 6T

9H, 11T

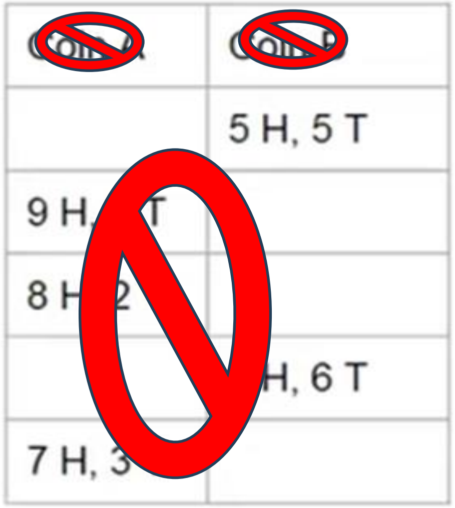
$$\Theta_1 = \frac{24}{(24+6)} = 0.8$$

$$\Theta_2 = \frac{9}{9+11} = 0.45$$

- If we don’t know the identity of the coin label, then we will assume or estimate the probabilities.



B	H	T	T	T	H	H	T	H	T	H
A	H	H	H	H	T	H	H	H	H	H
A	H	T	H	H	H	H	H	T	H	H
B	H	T	H	T	T	T	H	H	T	T
A	T	H	H	H	T	H	H	H	T	H



Coin A	Coin B
	5 H, 5 T
9 H, 1 T	
8 H, 2 T	
	4 H, 6 T
7 H, 3 T	

24H, 6T

9H, 21T

$$\Theta_1 = \frac{no.of\ heads\ with\ C1}{Total\ no.of\ flips\ using\ C1}$$

$$\Theta_2 = \frac{no.of\ heads\ with\ C2}{Total\ no.of\ flips\ using\ C2}$$

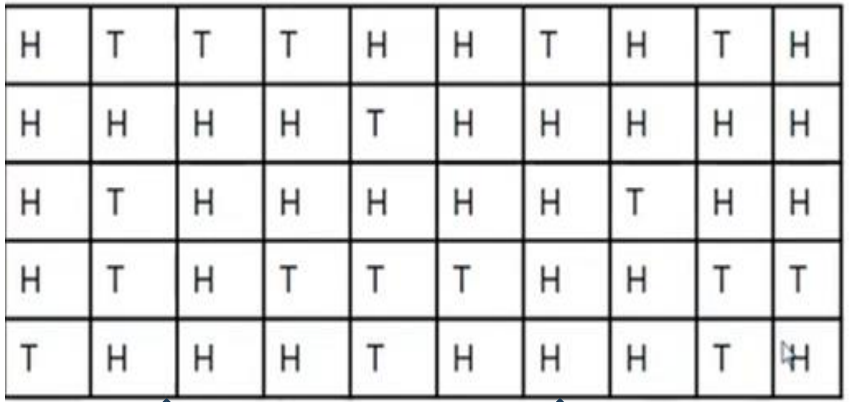
$$\Theta_1 = \frac{24}{(24+6)} = 0.8$$

$$\Theta_2 = \frac{9}{9+21} = 0.45$$

Solution is E-M algorithm

First value: 5H 5T for coin A
P(A)*5H = L(H) = 0.45*5 = 2.2
P(A)*5T = L(T) = 0.45*5 = 2.2

First value: 5H 5T for coin B
P(B)*5H = L(H) = 0.55*5 = 2.8
P(B)*5T = L(T) = 0.55*5 = 2.8



H	T	T	T	H	H	T	H	T	H
H	H	H	H	T	H	H	H	H	H
H	T	H	H	H	H	T	H	H	H
H	T	H	T	T	T	H	H	T	T
T	H	H	H	T	H	H	H	T	H

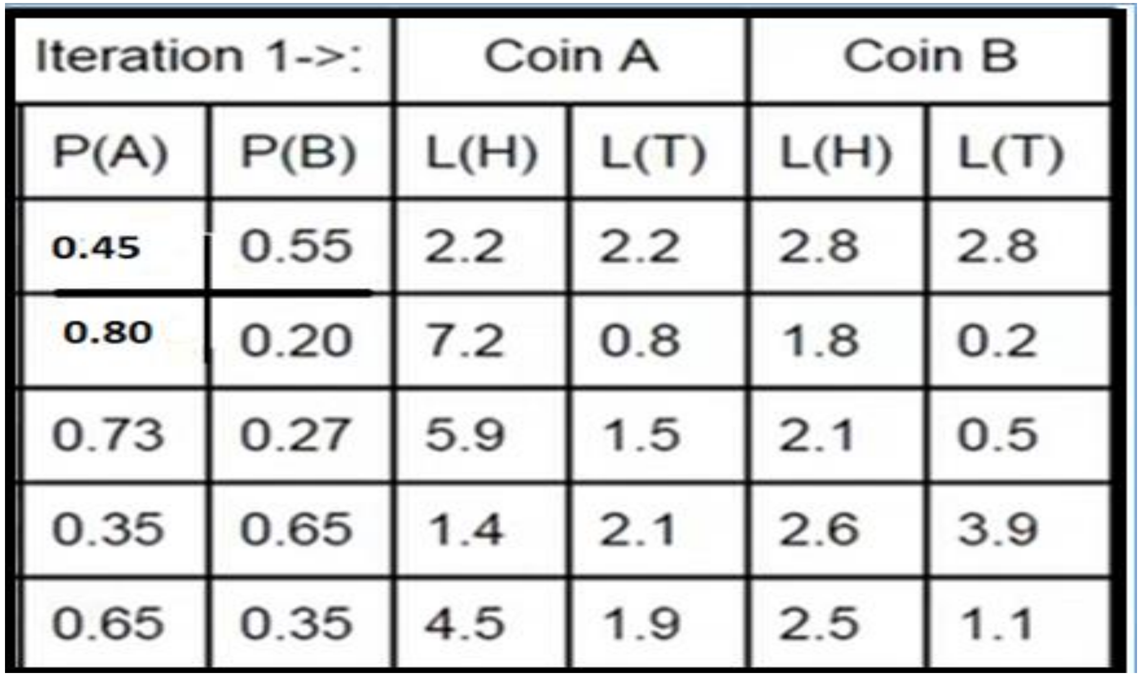
1

3

Initial values of
$\theta_A^{(0)}$ = 0.6
$\theta_B^{(0)}$ = 0.5

$$\theta_A^{(1)} = \frac{21.3}{(21.3 + 8.6)} = 0.71$$

$$\theta_B^{(1)} = \frac{11.7}{11.7 + 8.4} = 0.58$$



Iteration 1->:		Coin A		Coin B	
P(A)	P(B)	L(H)	L(T)	L(H)	L(T)
0.45	0.55	2.2	2.2	2.8	2.8
0.80	0.20	7.2	0.8	1.8	0.2
0.73	0.27	5.9	1.5	2.1	0.5
0.35	0.65	1.4	2.1	2.6	3.9
0.65	0.35	4.5	1.9	2.5	1.1

$$\sum L(H) = 21.3$$

$$\sum L(T) = 8.6$$

$$\sum L(H) = 11.7$$

$$\sum L(T) = 8.4$$

$\theta_A^{(10)}$ = 0.80

$\theta_B^{(10)}$ =0.52

Example

If we don’t know the identity of the coin label, then we will assume or estimate the probabilities.

Assume initial values of

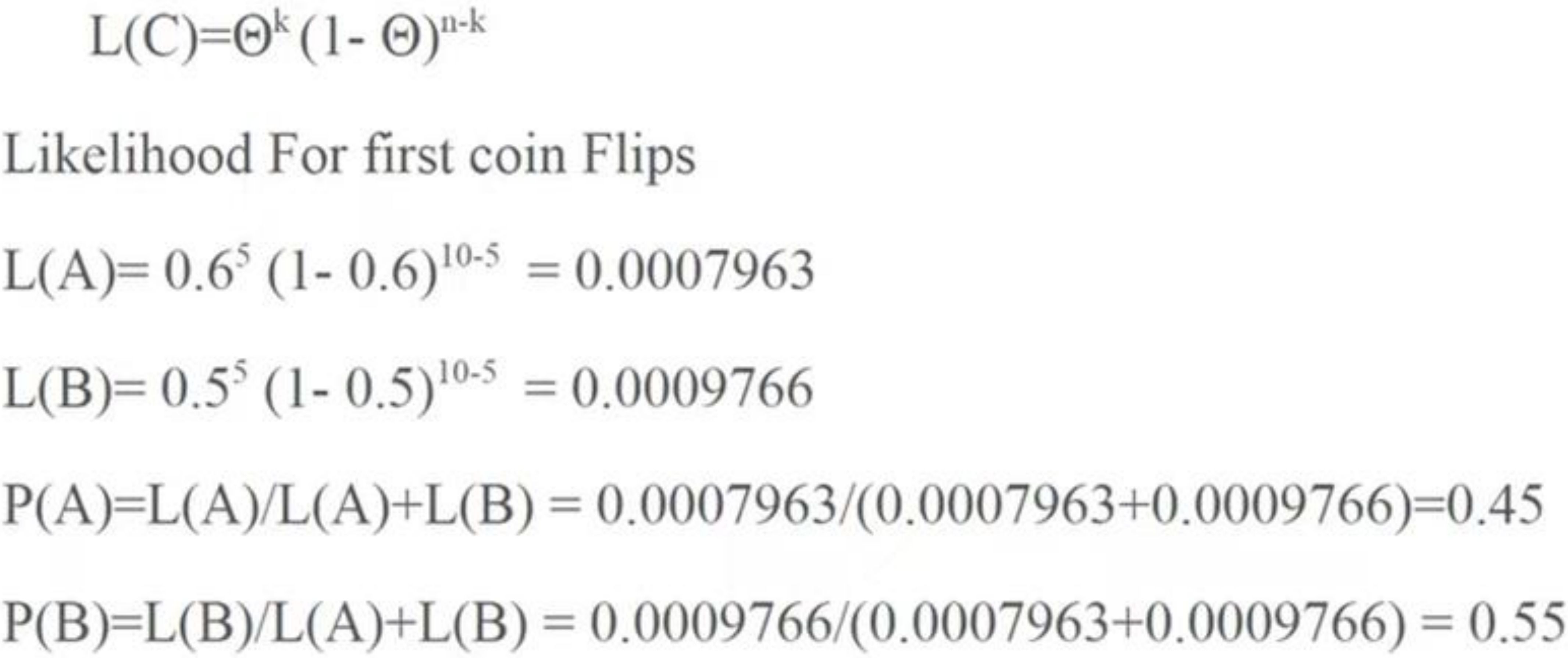
$Θ_A^{(0)}$ = 0.6

$Θ_B^{(0)}$ = 0.5

Then, we use Binomial distribution to find likelihood

$$L(C) = \theta^k (1-\theta)^{n-k}$$

Example



$$L(C)=\Theta^k (1- \Theta)^{n-k}$$

Likelihood For first coin Flips

$$L(A)= 0.6^5 (1- 0.6)^{10-5} = 0.0007963$$

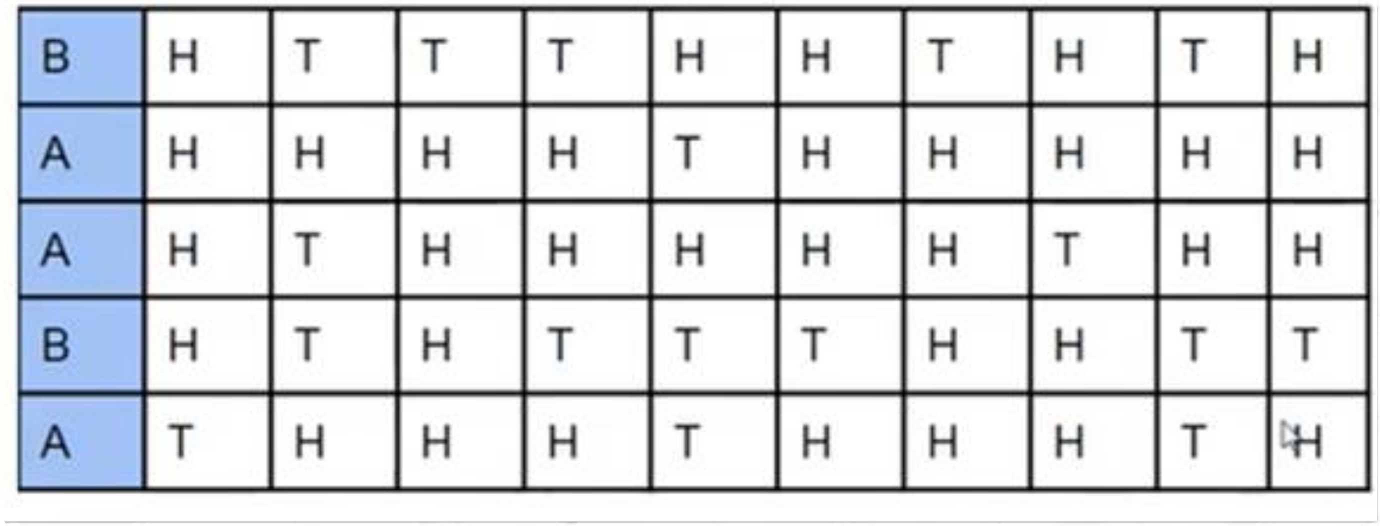
$$L(B)= 0.5^5 (1- 0.5)^{10-5} = 0.0009766$$

$$P(A)=L(A)/L(A)+L(B) = 0.0007963/(0.0007963+0.0009766)=0.45$$

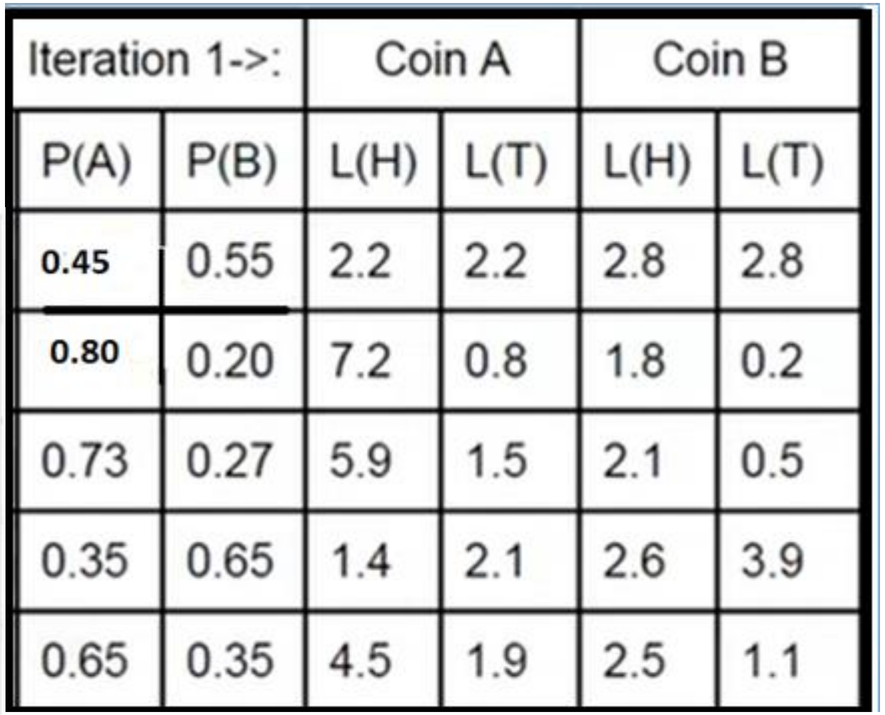
$$P(B)=L(B)/L(A)+L(B) = 0.0009766/(0.0007963+0.0009766) = 0.55$$

Example

Find probability of all coins with all flips.
It will be as follows:



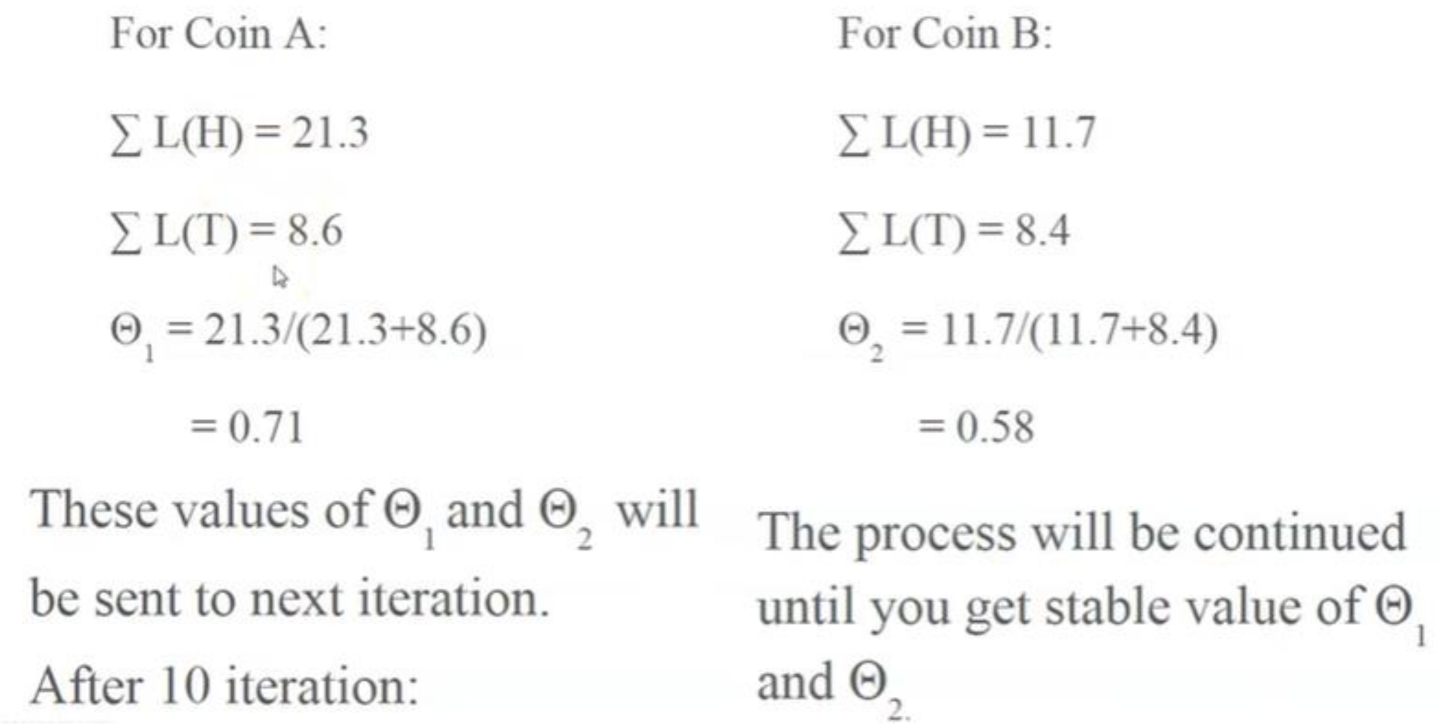
B	H	T	T	T	H	H	T	H	T	H
A	H	H	H	H	T	H	H	H	H	H
A	H	T	H	H	H	H	H	T	H	H
B	H	T	H	T	T	T	H	H	T	T
A	T	H	H	H	T	H	H	H	T	H



Iteration 1->:		Coin A		Coin B	
P(A)	P(B)	L(H)	L(T)	L(H)	L(T)
0.45	0.55	2.2	2.2	2.8	2.8
0.80	0.20	7.2	0.8	1.8	0.2
0.73	0.27	5.9	1.5	2.1	0.5
0.35	0.65	1.4	2.1	2.6	3.9
0.65	0.35	4.5	1.9	2.5	1.1

L(H) : Likely number of heads
L(T): Likely number of tails

Example



For Coin A:

$\sum L(H) = 21.3$

$\sum L(T) = 8.6$

$\Theta_1 = 21.3/(21.3+8.6)$

$= 0.71$

For Coin B:

$\sum L(H) = 11.7$

$\sum L(T) = 8.4$

$\Theta_2 = 11.7/(11.7+8.4)$

$= 0.58$

These values of Θ_1 and Θ_2 will be sent to next iteration.

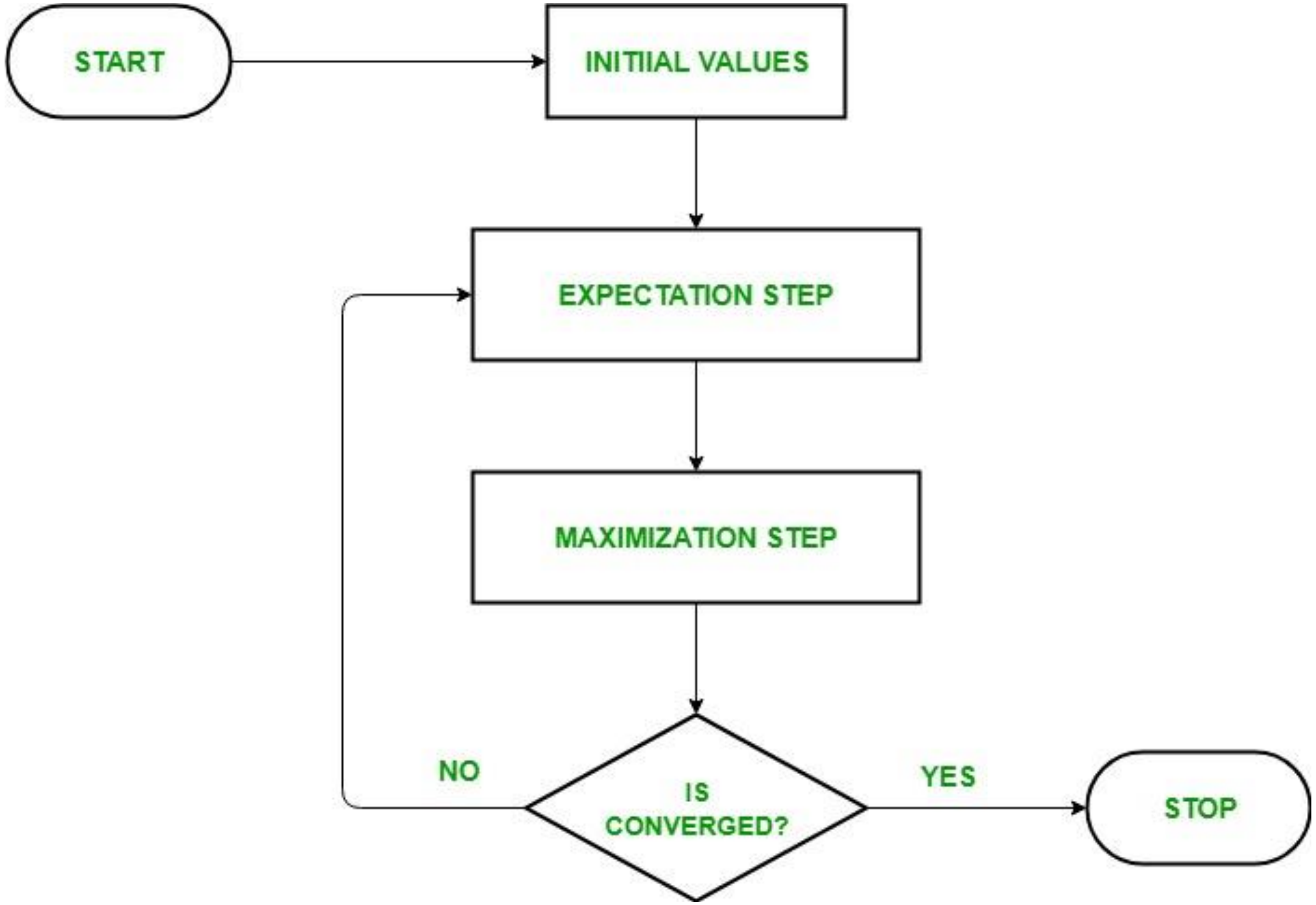
After 10 iteration:

The process will be continued until you get stable value of Θ_1 and $\Theta_{2.}$

θ_A= 0.80

θ_B =0.52

EM algorithm



Usage of EM algorithm

It can be used to fill the missing data in a sample.

It can be used for the purpose of estimating the parameters of Hidden Markov Model (HMM).

It can be used for discovering the values of latent variables.

Advantages of EM algorithm

It is always guaranteed that likelihood will increase with each iteration.

The E-step and M-step are often easy for many problems in terms of implementation.

Disadvantages of EM algorithm

It has slow convergence.

It makes convergence to the local optima only.

It requires both the probabilities, forward and backward.

Clustering Metrics in Machine Learning

Clustering Metrics in Machine Learning

- Clustering - unsupervised machine-learning approach
- Group similar data points based on specific attributes.
- It is critical to evaluate the quality of the clusters created when using clustering techniques.
- These metrics are quantitative indicators used to evaluate the performance and quality of clustering algorithms.

Clustering Metrics in Machine Learning

- Silhouette score – will discuss
- Davies–Bouldin Index - self study
- Calinski-Harabasz Index – self study

Selecting the number of clusters with silhouette analysis

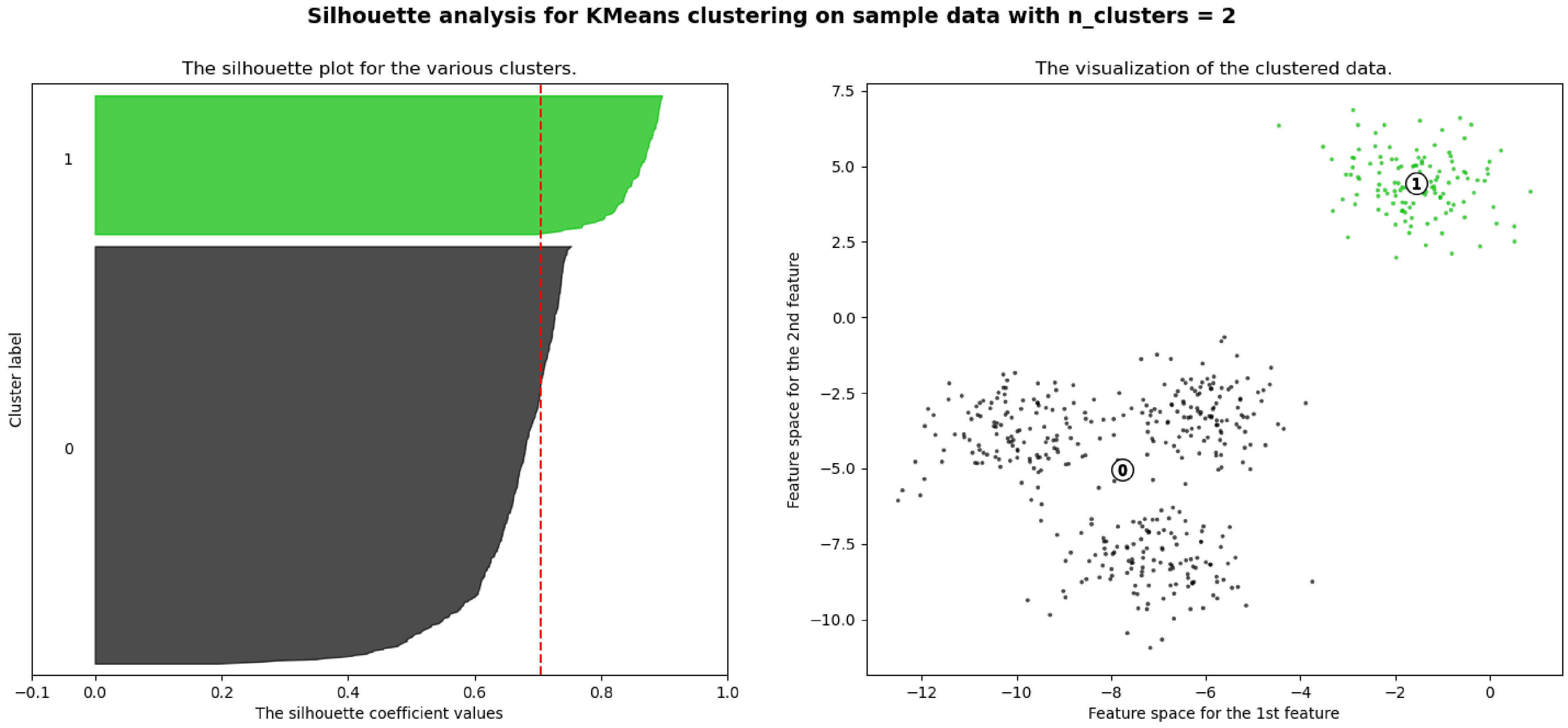
- Silhouette analysis can be used to study the separation distance between the resulting clusters.
- The silhouette plot displays a measure of how close each point in one cluster is to points in the neighboring clusters and thus provides a way to assess parameters like number of clusters visually.
- This measure has a range of [-1, 1].

Silhouette coefficients values & meaning

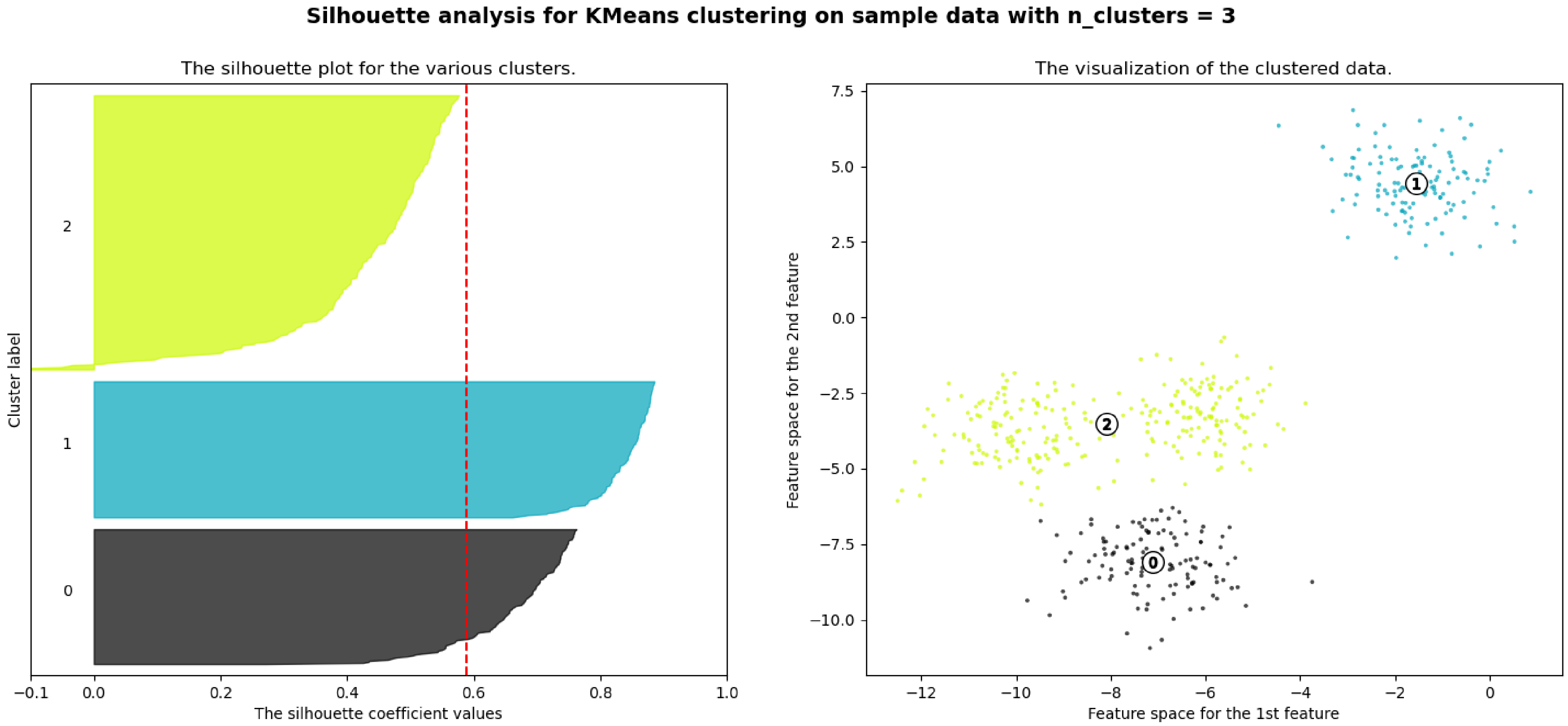
- Value near +1 indicate that the sample is far away from the neighboring clusters.

- A value of 0 indicates that the sample is on or very close to the decision boundary between two neighboring clusters

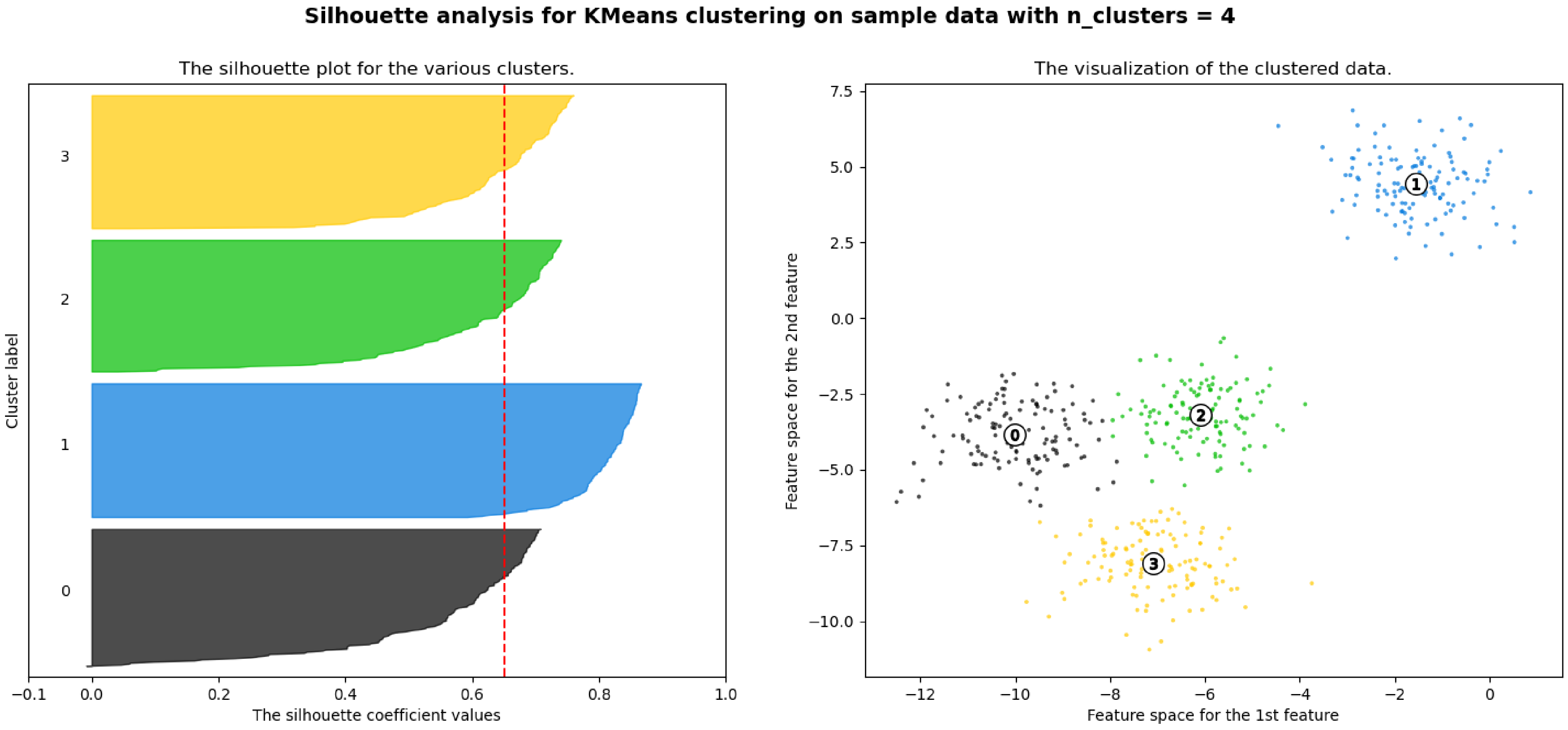
- Negative values indicate that those samples might have been assigned to the wrong cluster.



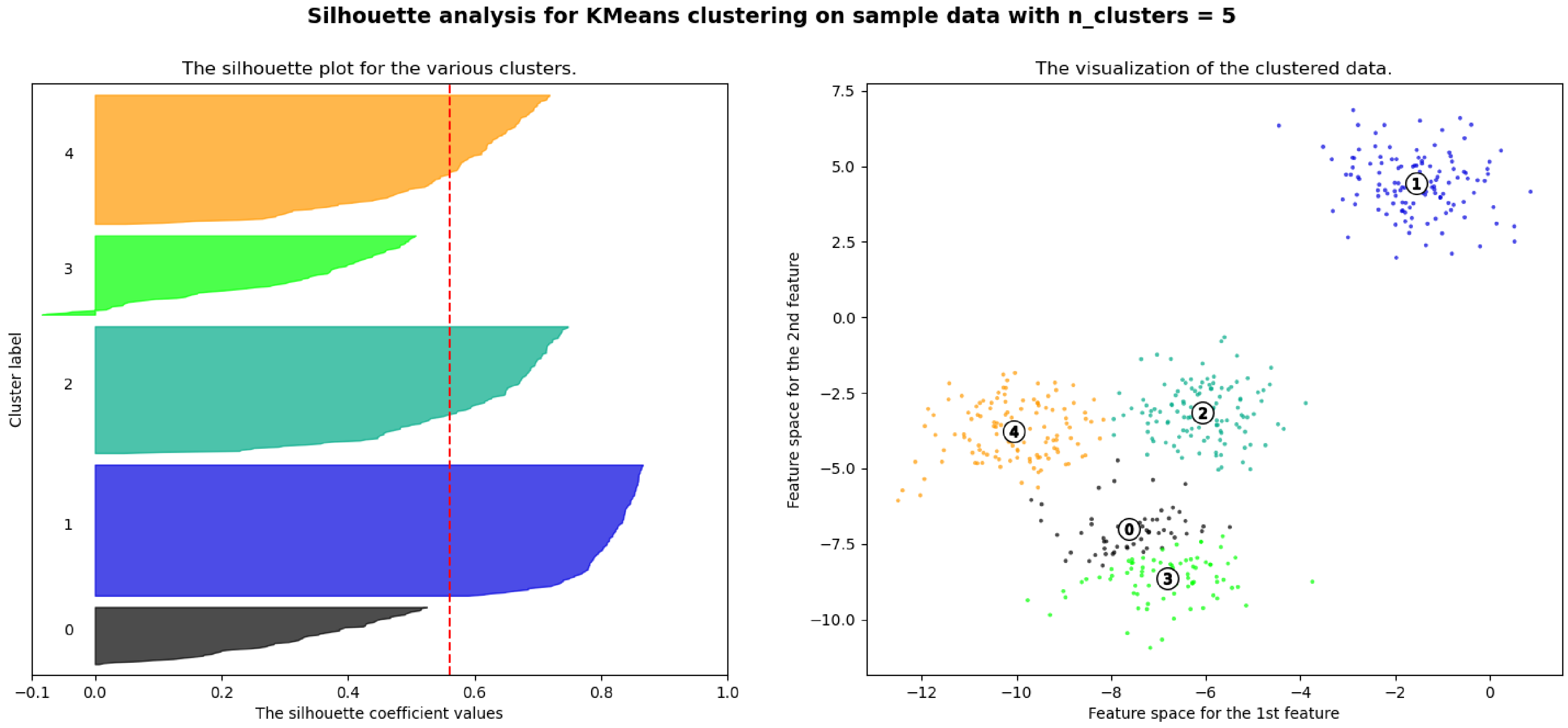
For n_clusters = 2 The average silhouette_score is : 0.7049787496083262



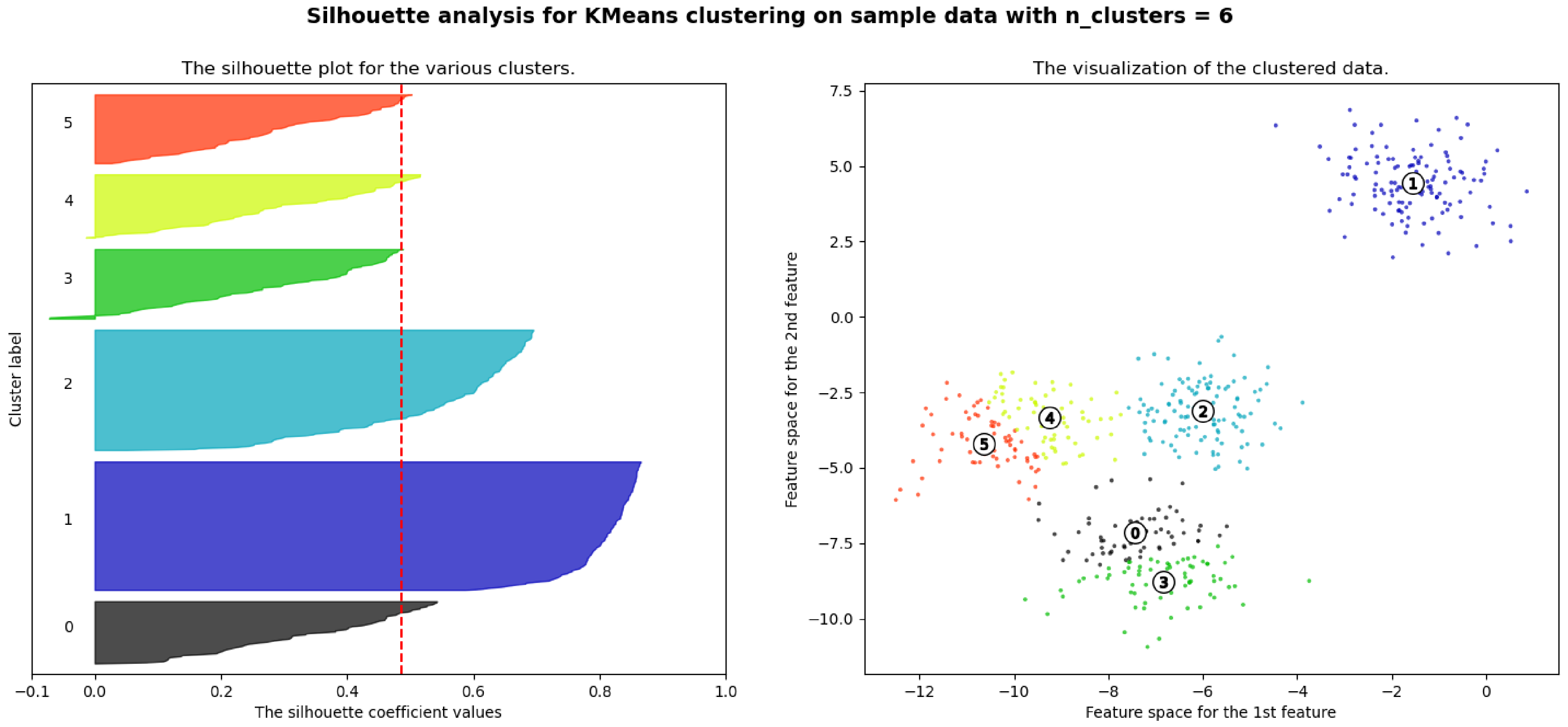
For n_clusters = 3 The average silhouette_score is : 0.5882004012129721



For n_clusters = 4 The average silhouette_score is : 0.6505186632729437



For n_clusters = 5 The average silhouette_score is : 0.561464362648773



For n_clusters = 6 The average silhouette_score is : 0.4857596147013469

- The silhouette plot shows that the n_clusters value of 3, 5 and 6 are a bad pick for the given data due to the presence of clusters with below average silhouette scores and also due to wide fluctuations in the size of the silhouette plots.

- Silhouette analysis is more ambivalent in deciding between 2 and 4.

- From the thickness of the silhouette plot the cluster size can be visualized.

- The silhouette plot for cluster 0 when n_clusters is equal to 2, is bigger in size owing to the grouping of the 3 sub clusters into one big cluster.

- However when the n_clusters = 4 all the plots are more or less of similar thickness and hence are of similar sizes as can be also verified from the labelled scatter plot on the right.

Calculating Silhouette Coefficient

P1 and P2 are in cluster1
P3 and P4 are in cluster2

Point	Cluster label
P1	1
P2	1
P3	2
P4	2

Distance matrix:

Point	P1	P2	P3	P4
P1	0	0.1	0.65	0.55
P2	0.1	0	0.7	0.6
P3	0.65	0.7	0	0.3
P4	0.55	0.6	0.3	0

Compute the Silhouette coefficient for each point, each cluster, and overall cluster.

- Silhouette coefficient can be calculated using $SC = 1 - \frac{a}{b}$
  - Here *a* is the average distance to the point to other points in the same cluster.
  - *b* represent the average distance of the point to the other points in the other clusters.

- **Consider Point P1:**
  - To calculate “a”, consider the points in this cluster.
  - This cluster has only two points P1 and P2.
  - Therefore, a = distance (P1 to P2) = 0.1
  - To calculate b, find the distance of P1 to P3 and P1 to P4.

- P1 to P3 = 0.65
- P1 to P4 = 0.55
- Average distance b from P1 to cluster C2 = $\frac{0.65+0.55}{2} = 0.6$

$$SC = 1 - \frac{a}{b} = 1 - \frac{0.1}{0.6} = 0.833$$

- **Consider Point P2:**
    - To calculate “a”, consider the points in this cluster.
    - This cluster has only two points P1 and P2.

a = distance (P1 to P2) 0.1

    - To calculate b, find the distance of P2 to P3 and P2 to P4.

- P2 to P3 = 0.7
- P2 to P4 = 0.6
- Average distance b from P2 to cluster C2 = $\frac{0.7+0.6}{2} = 0.65$

$$SC = 1 - \frac{a}{b} = 1 - \frac{0.1}{0.65} = 0.846$$

- **Consider Point P3:**
  - To calculate “a”, consider the points in this cluster.
  - This cluster has only two points P3 and P4.

a = distance (P3 to P4) 0.3

  - To calculate b, find the distance of P3 to P1 and P3 to P2.

- P3 to P1 = 0.65
- P3 to P2 = 0.7
- Average distance b from P3 to cluster C1 = $\frac{0.65+0.7}{2} = 0.675$

$$SC = 1 - \frac{a}{b} = 1 - \frac{0.3}{0.675} = 0.556$$

- **Consider Point P4:**
  - To calculate “a”, consider the points in this cluster.
  - This cluster has only two points P3 and P4.

a = distance (P3 to P4) 0.3

  - To calculate b, find the distance of P4 to P1 and P4 to P2.

- P4 to P1 = 0.55
- P4 to P2 = 0.6
- Average distance b from P4 to cluster C1 = $\frac{0.55+0.6}{2} = 0.575$

$$SC = 1 - \frac{a}{b} = 1 - \frac{0.3}{0.575} = 0.478$$

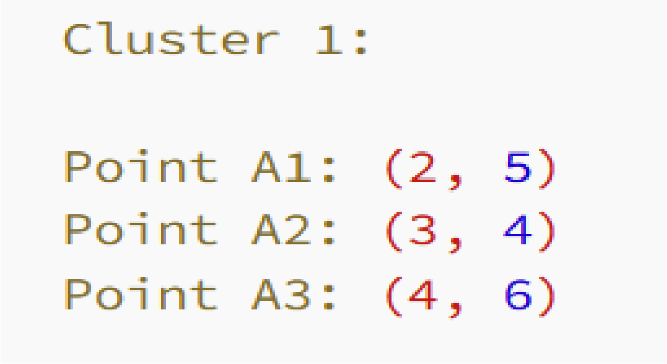
$$Average\ of\ SC for\ cluster1 = \frac{0.833 + 0.846}{2} = 0.84$$

$$Average\ of\ SC for\ cluster2 = \frac{0.556 + 0.478}{2} = 0.517$$

$$Overall\ average\ of\ SC = \frac{0.84 + 0.517}{2} = 0.68$$

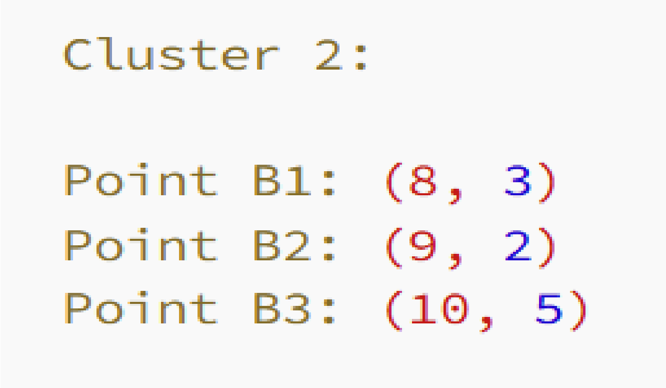
Assignment

Consider a scenario with 3 clusters, where each cluster has 3 points.



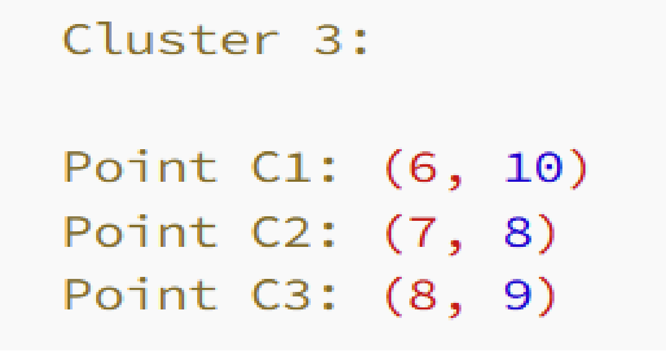
```
Cluster 1:

Point A1: (2, 5)
Point A2: (3, 4)
Point A3: (4, 6)
```



```
Cluster 2:

Point B1: (8, 3)
Point B2: (9, 2)
Point B3: (10, 5)
```



```
Cluster 3:

Point C1: (6, 10)
Point C2: (7, 8)
Point C3: (8, 9)
```

Calculate the silhouette coefficient for cluster 1, cluster 2, and cluster 3.

Thank you