MOF to improve clustering algorithm

Jiraphat Chaiyabut 6234307323

Advisor: Associate Professor Krung Sinapiromsaran

Department of Mathematics and Computer Science,

Faculty of Science, Chulalongkorn University

Presentation date:

Agenda

- ❑ Artificial intelligence, Machine learning, Unsupervised learning
- ❑ Clustering analysis
- ❑ Sillouette
- ❑ Anomaly scoring algorithm
- ❑ MOF to improve clustering algorithms
- ❑ MOF apply to real world dataset
- ❑ Conclusion

What is artificial intelligence (AI)?

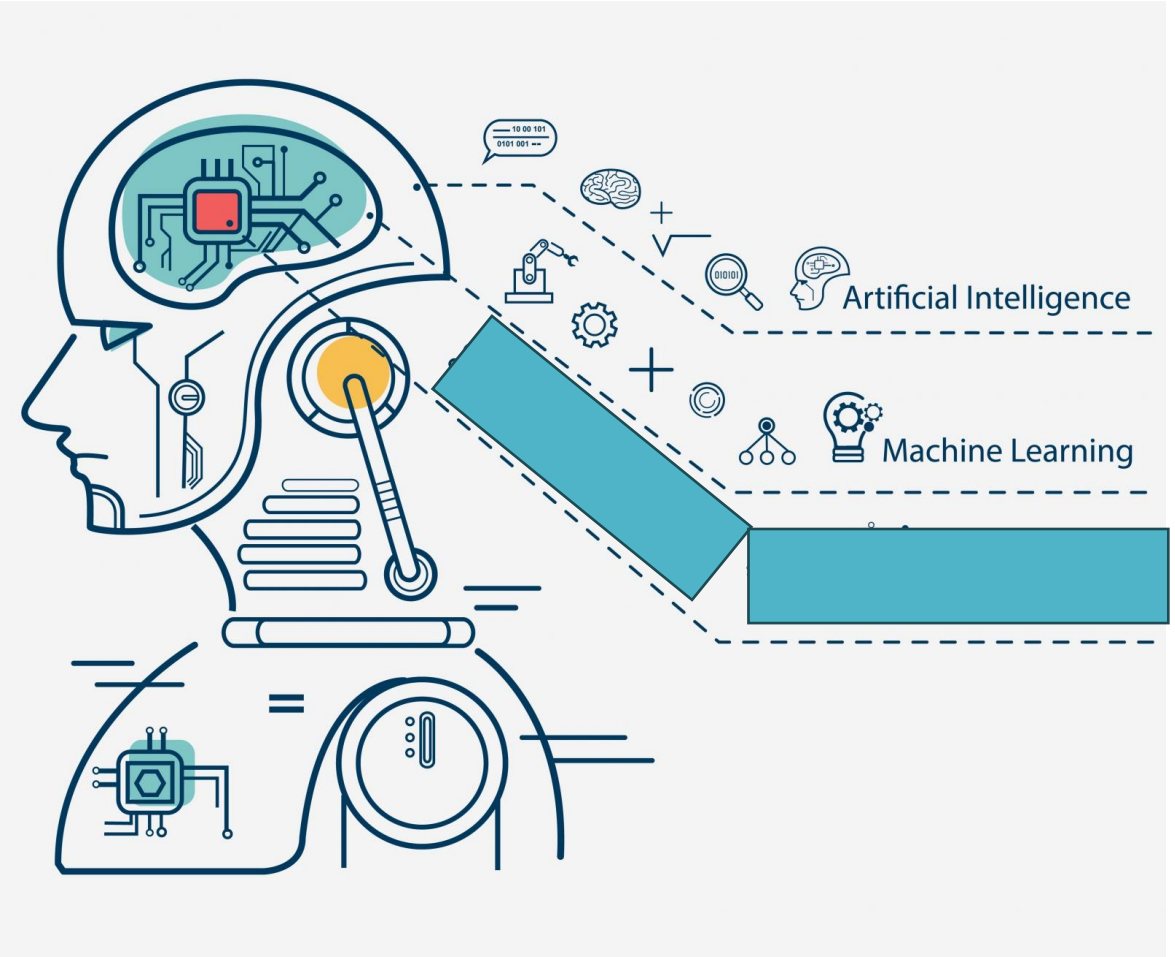


Fig source : https://tips.thaiware.com/1746.html

The main purpose of Machine learning

- Learning phase
- Inference from the model

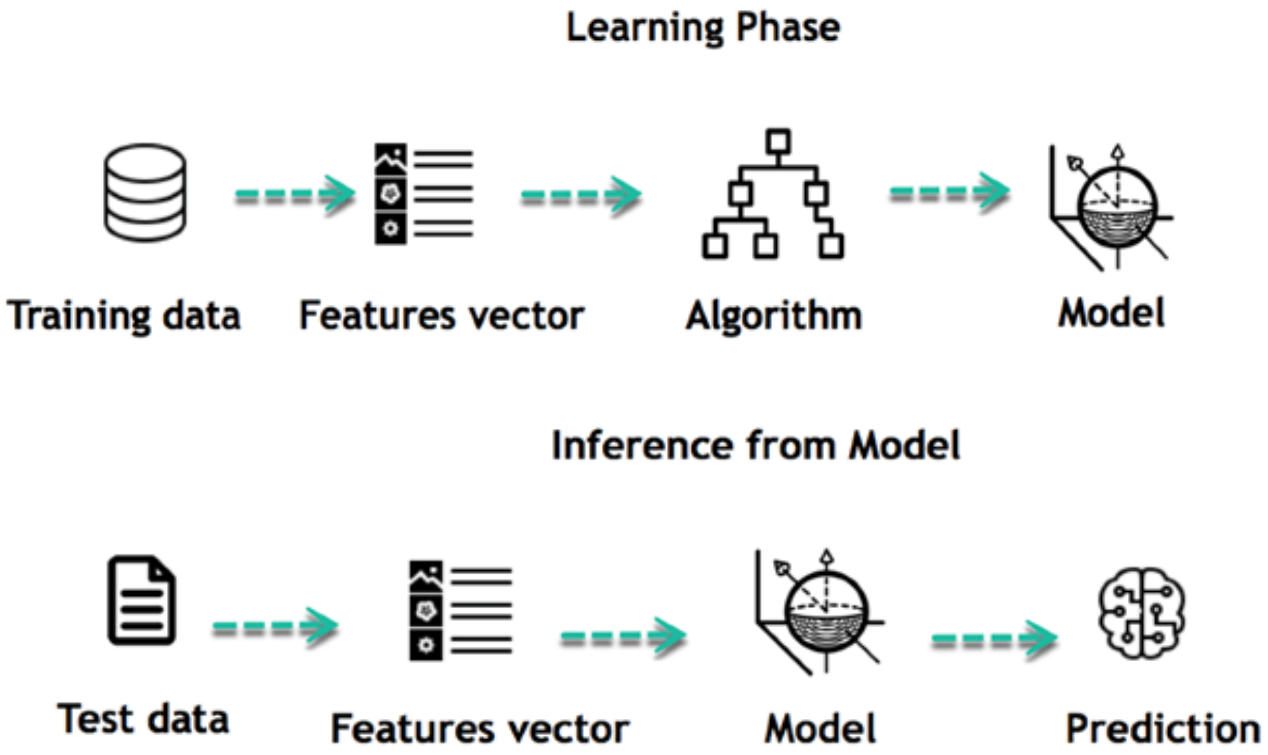


Fig source : https://www.thaiprogrammer.org/2018/12/อะไรคือ-การเรียนรู้ของเครื่องmachine-learning/

Type of Machine Learning

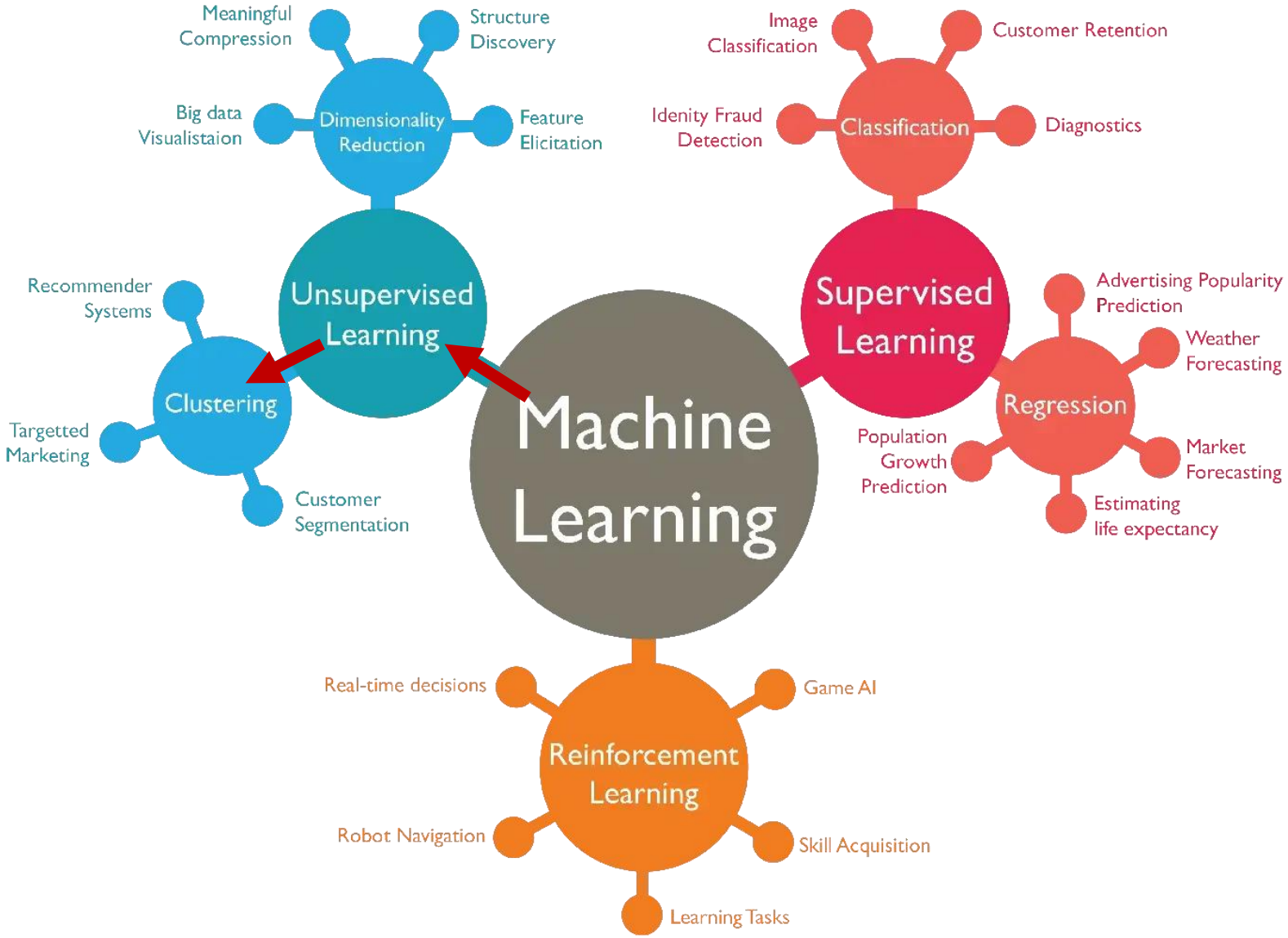


Fig source : https://medium.com/investic/machine-learning-fa8bf6663c07

Clustering analysis

- **Cluster analysis** or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar to each other than to those in other groups

K-mean algorithm

The way k-means algorithm works.

1. Initialization: Choose the number of clusters (k) you want to create.
2. Assignment: Assign each data point in the dataset to the nearest centroid.
3. Update: Calculate the mean (average) of all the data points in each cluster.
4. Iteration: Repeat steps 2 and 3 until the centroids' positions no longer change significantly

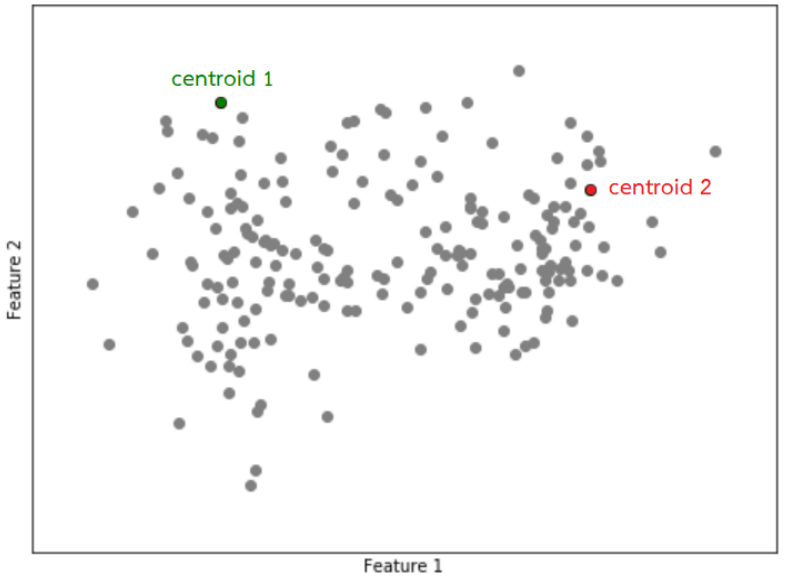


Figure 1

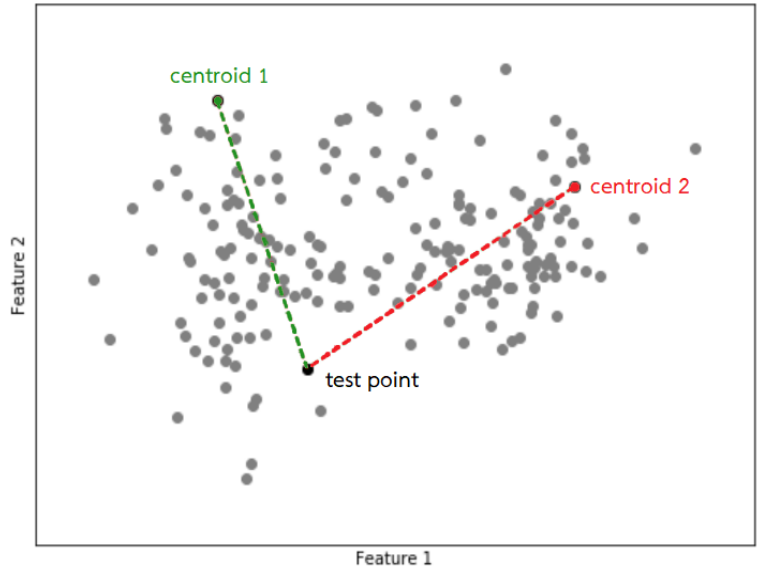


Figure 2

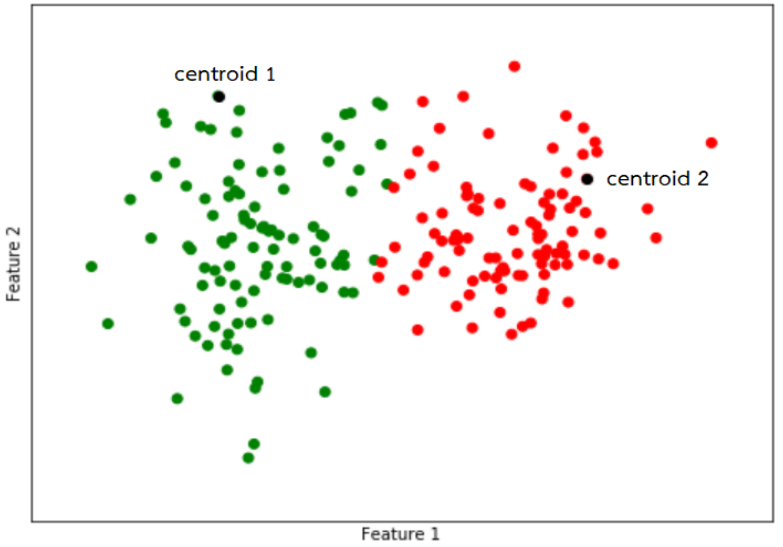


Figure 3

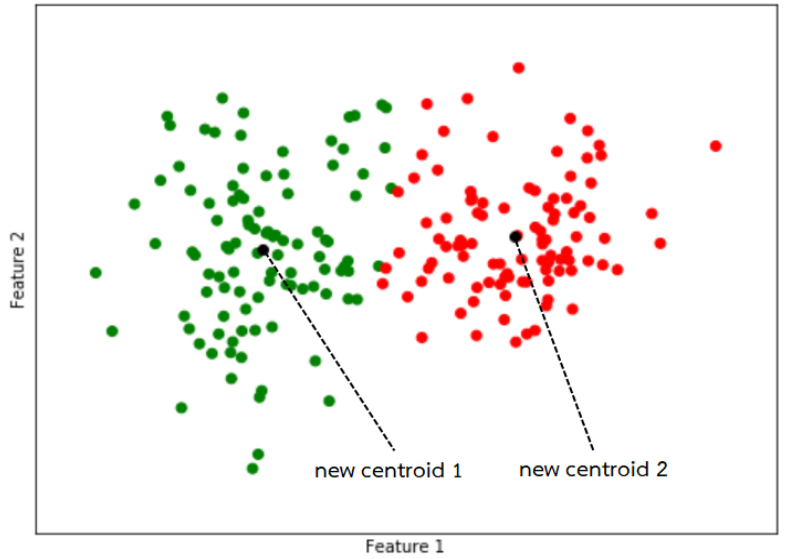


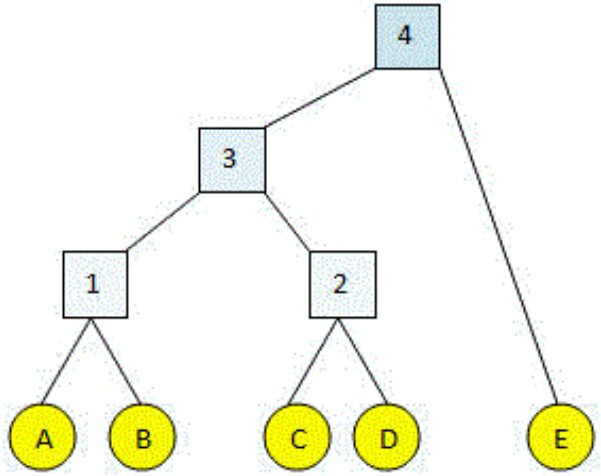
Figure 4

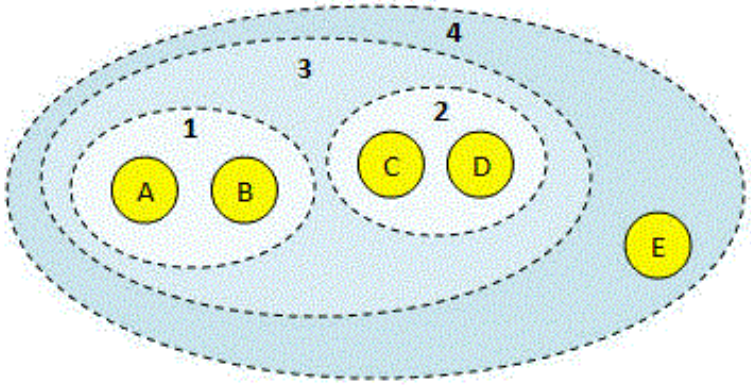
Fig source : https://bigdata.go.th/big-data-101/k-means-algorithm-for-clustering-large-data-sets-with-categorical-values/

Hierarchical clustering

The hierarchical clustering technique has two approaches

- Agglomerative: Agglomerative is a bottom-up approach, in which the algorithm starts with taking all data points as single clusters and merging them until one cluster is left.
- Single Linkage: the distance between two clusters is the minimum distance between members of the two clusters





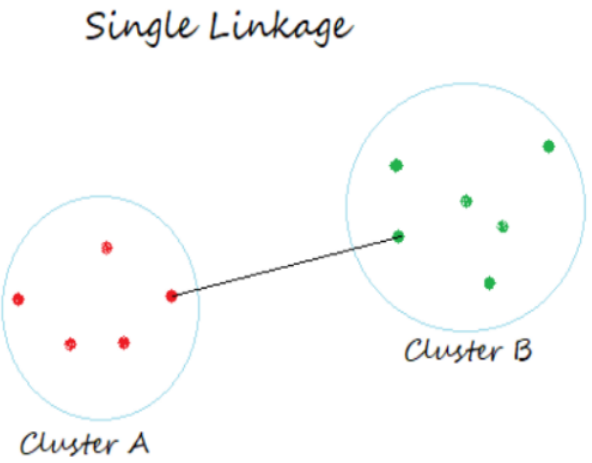


Fig source : https://www.analyticsvidhya.com/blog/2021/06/single-link-hierarchical-clustering-clearly-explained/

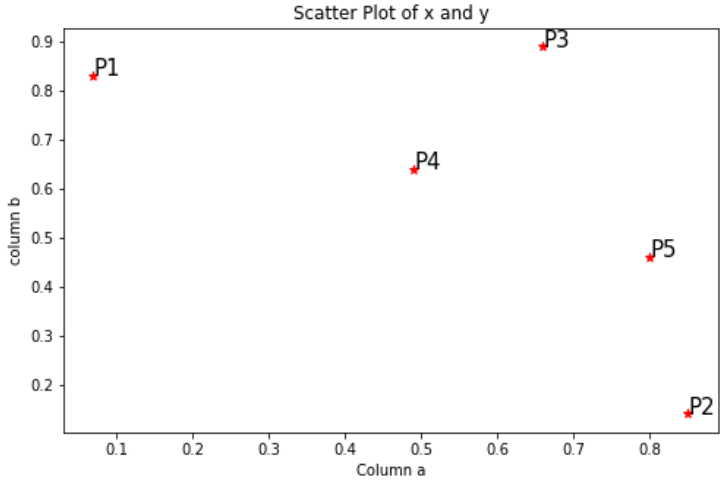
The way k-means algorithm works.

Step 1: Calculating the distance matrix in Euclidean method

Step 2: Look for the least distance and merge those into a cluster

Step 3: Re-compute the distance matrix after forming a cluster

Repeat steps 2,3 until we are left with one single cluster.



	P1	P2	P3	P4	P5
P1	0				
P2	1.04139	0			
P3	0.59304	0.77369	0		
P4	0.46098	0.61612	0.30232	0	
P5	0.81841	0.32388	0.45222	0.35847	0

	P1	P2	P3	P4	P5
P1	0				
P2	1.04139	0			
P3	0.59304	0.77369	0		
P4	0.46098	0.61612	0.30232	0	
P5	0.81841	0.32388	0.45222	0.35847	0

	P1	P2	P3,P4	P5
P1	0			
P2	1.04139	0		
P3,P4	0.46098	0.61612	0	
P5	0.81841	0.32388	0.35847	0

	P1	P2,P5	P3,P4
P1	0		
P2,P5	0.81841	0	
P3,P4	0.46098	0.35847	0

	P1	P2,P5,P3,P4
P1	0	
P2,P5,P3,P4	0.46098	0

Sillouette

- $|C_i|$= The number of points belonging to cluster i
- $a(i)$ = Average distance inside cluster
- $b(i)$ = Average distance to nearest other cluster

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \text{ if } |C_i| > 1$$

and

$$s(i) = 0, \text{ if } |C_i| = 1$$

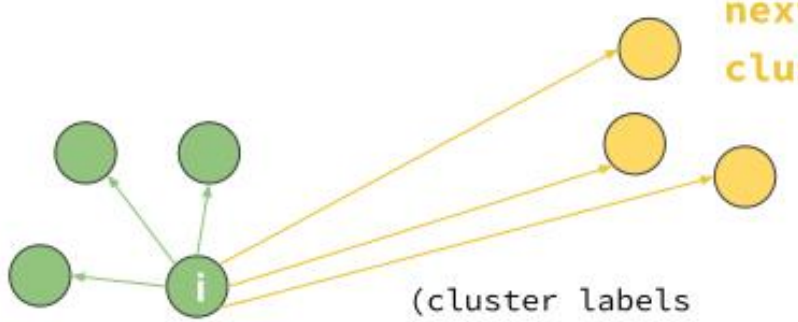
Which can be also written as:

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{if } a(i) > b(i) \end{cases}$$

From the above definition it is clear that

$$-1 \leq s(i) \leq 1$$

b(i): distance between i and its **next nearest cluster centroid**



(cluster labels assigned from ground-truth classes)

a(i): average distance between i and all of the other points in **its own cluster**

For a single point, i

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

$$-1 \leq s(i) \leq 1$$

Sprawling, overlapped clusters

Tight, well-separated clusters

Fig source : https://www.platform.ai/post/the-silhouette-loss-function-metric-learning-with-a-cluster-validity-index

Example

Now, let's put this into practice with a simple example:

Consider two clusters:

Cluster 1: [2, 3, 4]

Cluster 2: [8, 9, 10]

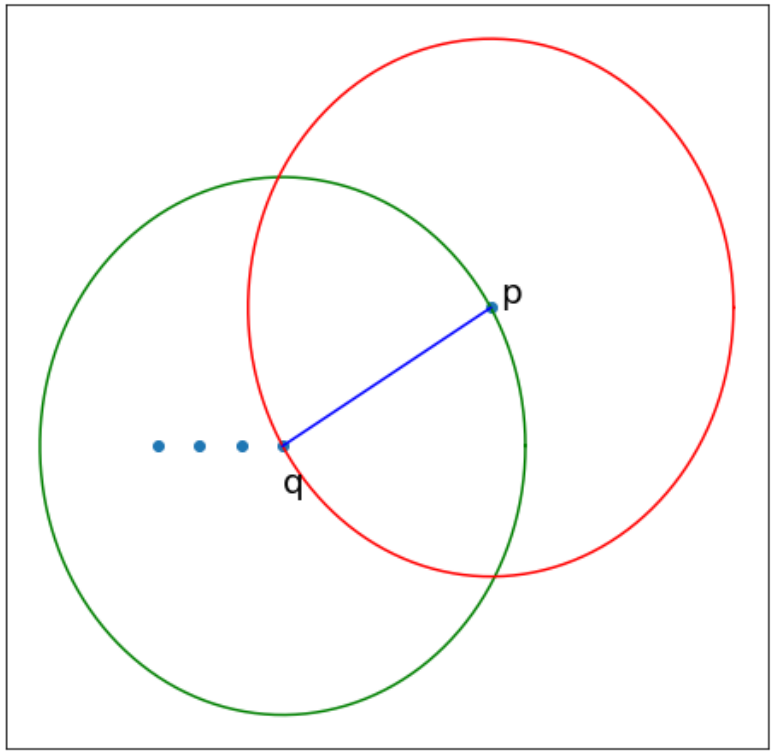
Let's calculate the silhouette score for data point 3 in Cluster 1.

1. $a(3) = (|3-2| + |3-4|)/2 = (1 + 1)/2 = 1$
2. $b(3) = (|3-8| + |3-9| + |3-10|)/3 = (5 + 6 + 7)/3 \approx 6$
3. $s(3) = (6 - 1) / \max(1, 6) = 5/6 \approx 0.83$

So, the silhouette score for the data point 3 is approximately 0.83

MOF algorithm

Mass ratio variance based outlier factor (MOF 2020)



Volume

- Volume of hypersphere

Mass

- The number of data points inside a hypersphere

Mass ratio of *q* w.r.t *p (red line)*

- Ratio of the number of data points inside green and red hyperspheres

Source : P. Changsakul. (2022). Parameter-free outlier scoring using mass ratio variance for static and streaming data [Slide presentation]

MOF algorithm

- Calculation of mass

Definition 1 *(Distance between **p** and **q**)*

Finding radius of the hypersphere by Euclidean distance.

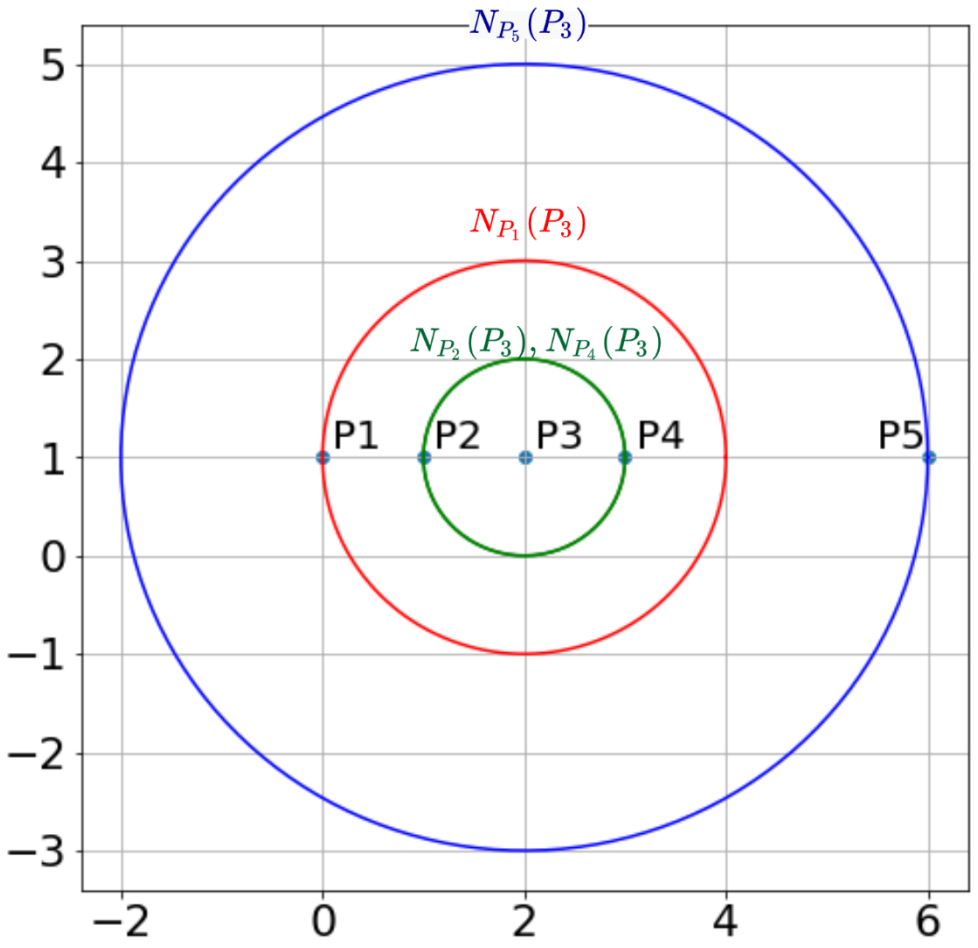
$$\mathrm{d}(\boldsymbol{p}, \boldsymbol{q}) = \sqrt{\sum_{i=1}^{d} (\boldsymbol{p}_i - \boldsymbol{q}_i)^2}$$

Definition 2 *(Neighborhoods of data point **q** with respect to data point **p**)*

Finding the set of data points inside hypersphere on center data point **q**

$$\mathrm{N}_p(\boldsymbol{q}) = \{ \boldsymbol{o} \in D \mid \mathrm{d}(\boldsymbol{q}, \boldsymbol{o}) \leq \mathrm{d}(\boldsymbol{q}, \boldsymbol{p}) \}$$

- How to calculate $|N_{P3}(P5)|$, $|N_{P3}(P4)|$, ..., $|N_{P3}(P1)|$



- $|N_{P3}(P5)| = |\{P3,P2,P4,P1,P5\}| = 5$
- $|N_{P3}(P1)| = |\{P3,P2,P4,P1\}| = 4$
- $|N_{P3}(P4)|, |N_{P3}(P2)| = |\{P3,P2,P4\}| = 3$

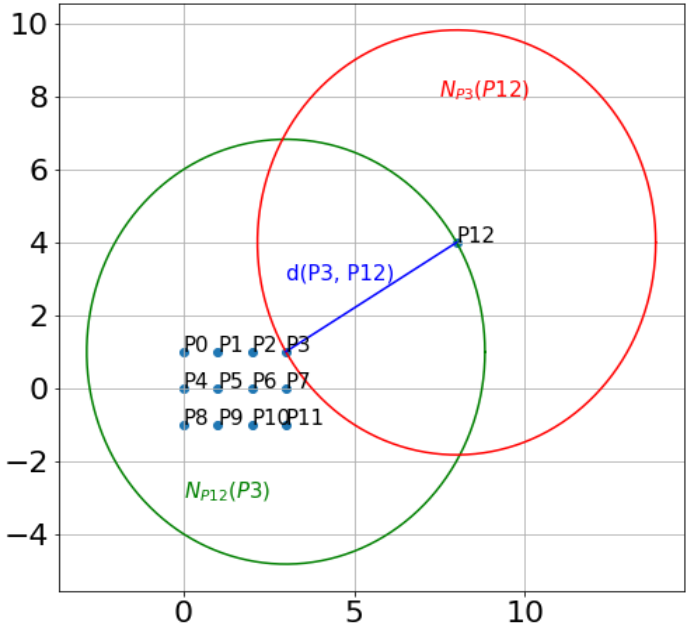
Source : P. Changsakul. (2022). Parameter-free outlier scoring using mass ratio variance for static and streaming data [Slide presentation]

- Calculation of mass ratio

Definition 3 *(The mass-ratio of data point **q** with respect to data point **p**)* Ratio of the number of data points inside hypersphere with same volume

$$massR_p(q) = \frac{|N_p(q)|}{|N_q(p)|}$$

For example : calculation of $massR_{P12}(P3)$



$$\mathrm{N_{P12}(P3)} = \{\, \mathrm{P0, P1, \ldots\ldots, P12} \} \text{ (all points)}$$

$$|\mathrm{N_{P12}(P3)}| = 13$$

$$\mathrm{N_{P3}(P12)} = \{\mathrm{P3, P12}\}$$

$$|\mathrm{N_{P3}(P12)}| = 2$$

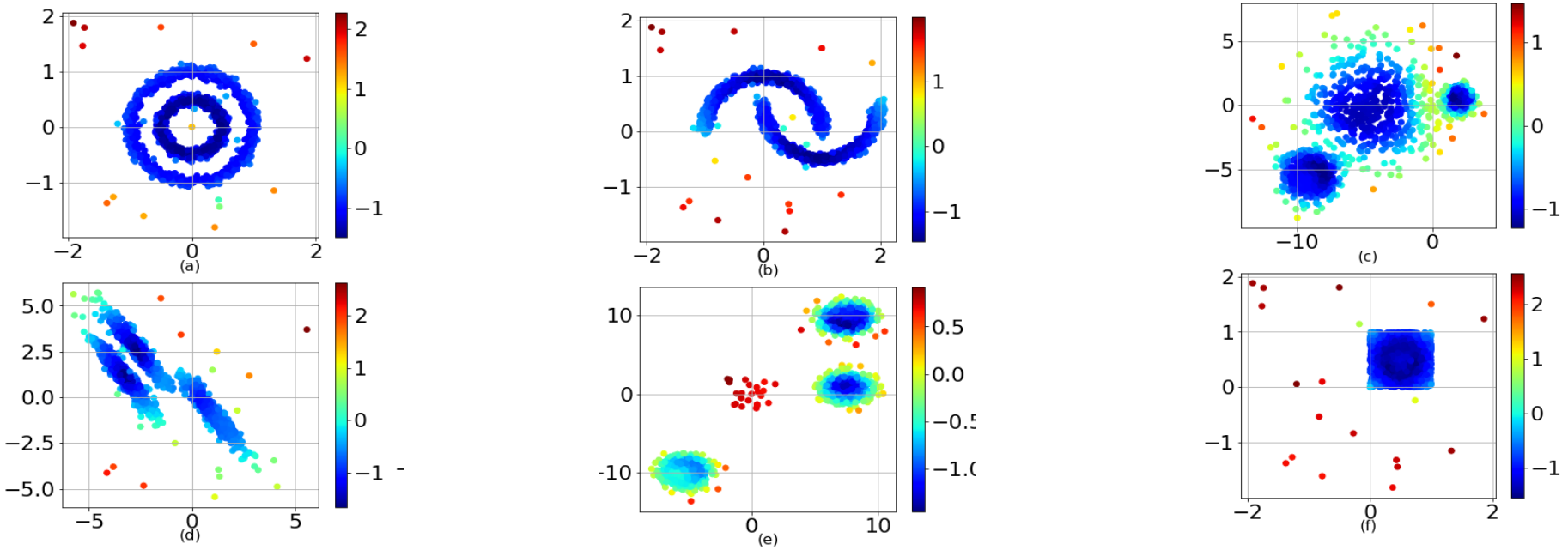
$$\mathrm{massR_{P12}(P3)} = \frac{|\mathrm{N_{P12}(P3)}|}{|\mathrm{N_{P3}(P12)}|} = \frac{13}{2}$$

- Calculation of MOF score

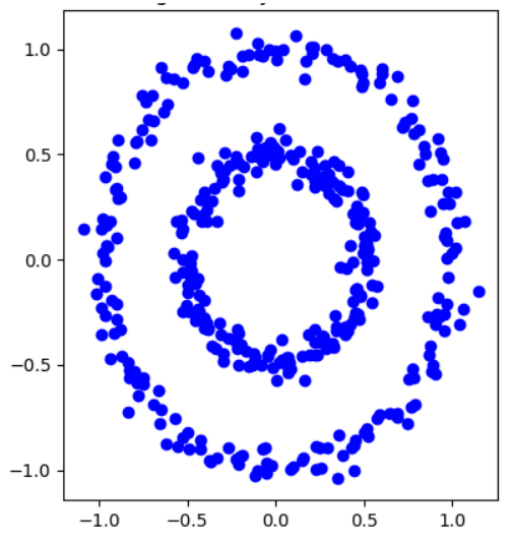
Definition 4 (Mass-ratio-variance based Outlier Factor of data point p) Variance of mass ratio of other data points with respect to data point ***p***

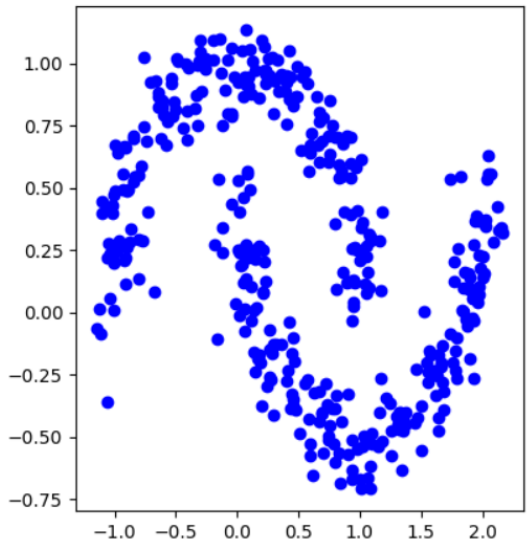
$$\mu_p = \frac{\sum_{i=1,\ q_i \neq p}^{n} \mathrm{massR}_p(q_i)}{n-1}$$

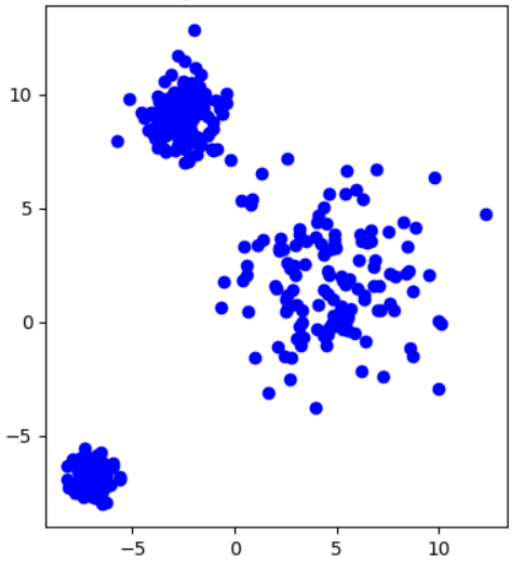
$$\mathrm{MOF}(p) = \frac{\sum_{i=1,\ q_i \neq p}^{n} \left(\mathrm{massR_p}(q_i) - \mu_p\right)^2}{n-1}$$

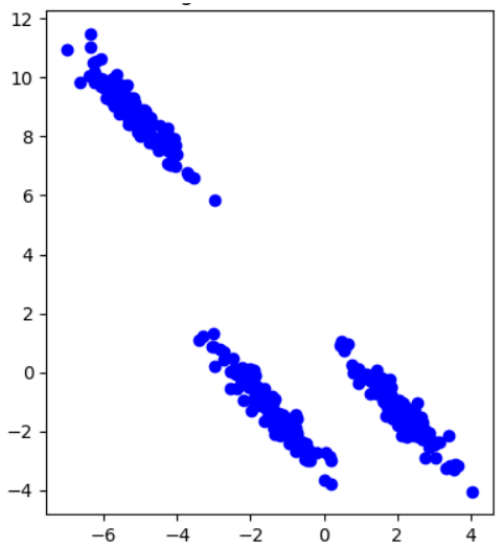


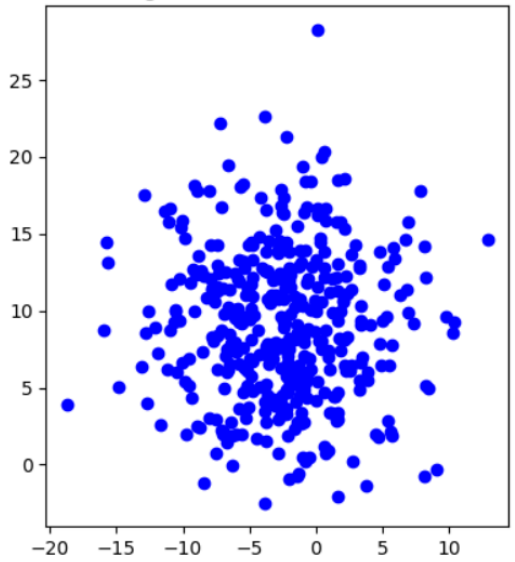
Synthesis datasets





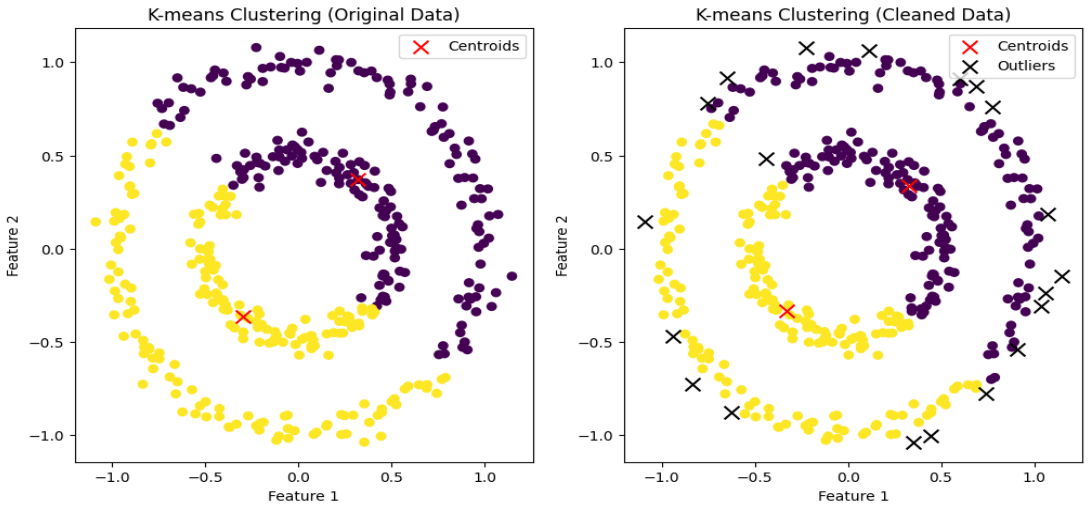


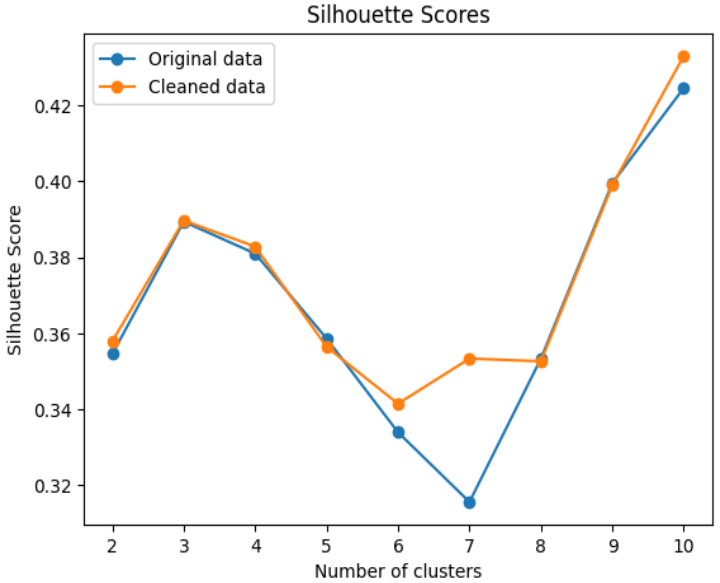




K-mean with 95 percentile threshold

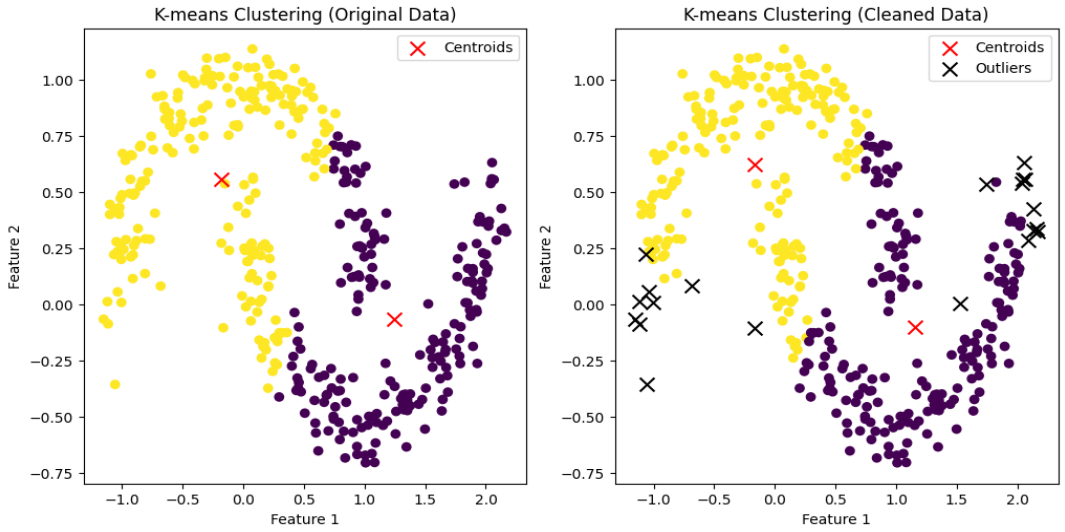
- Datasets contain double circle clusters





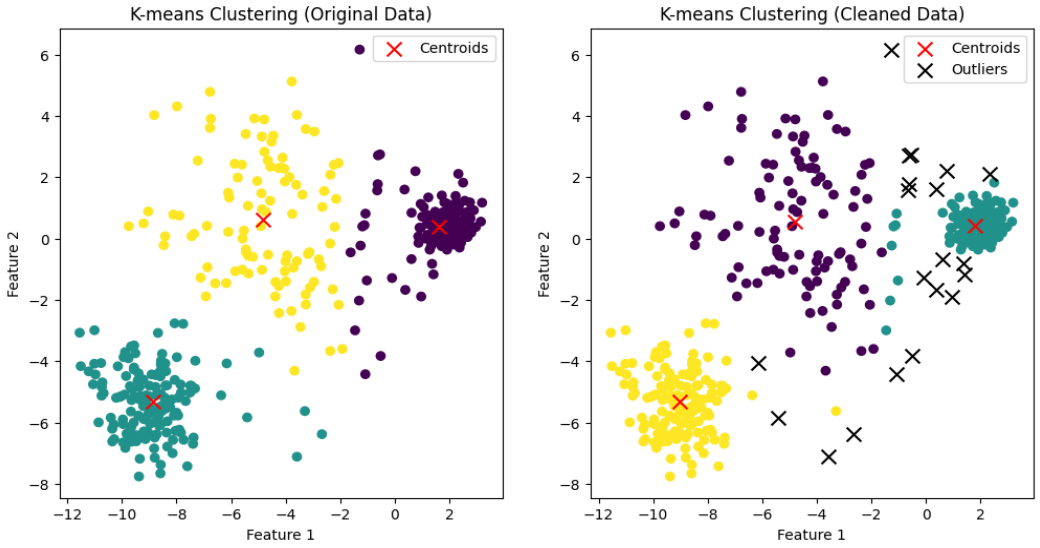
Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.3545	0.3579
3	0.3894	0.3897
4	0.3810	0.3828
5	0.3586	0.3564
6	0.3339	0.3415
7	0.3156	0.3533
8	0.3533	0.3526
9	0.3995	0.3990
10	0.4245	0.4330

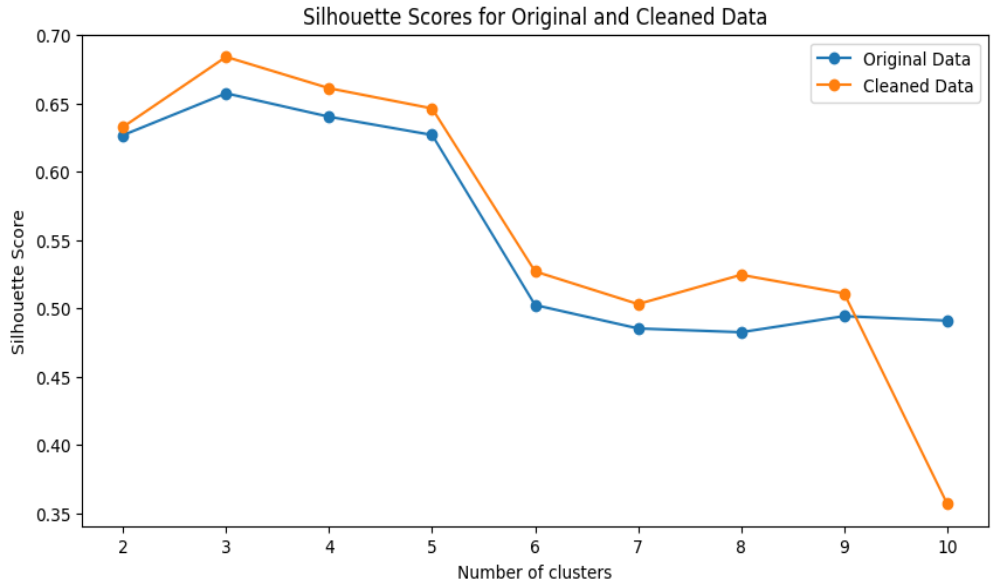
Datasets contain double moon clusters



Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.4831	0.4855
3	0.4192	0.4240
4	0.4424	0.4386
5	0.4666	0.4386
6	0.5068	0.5122
7	0.5028	0.5044
8	0.4974	0.4966
9	0.5032	0.5055
10	0.5020	0.5120

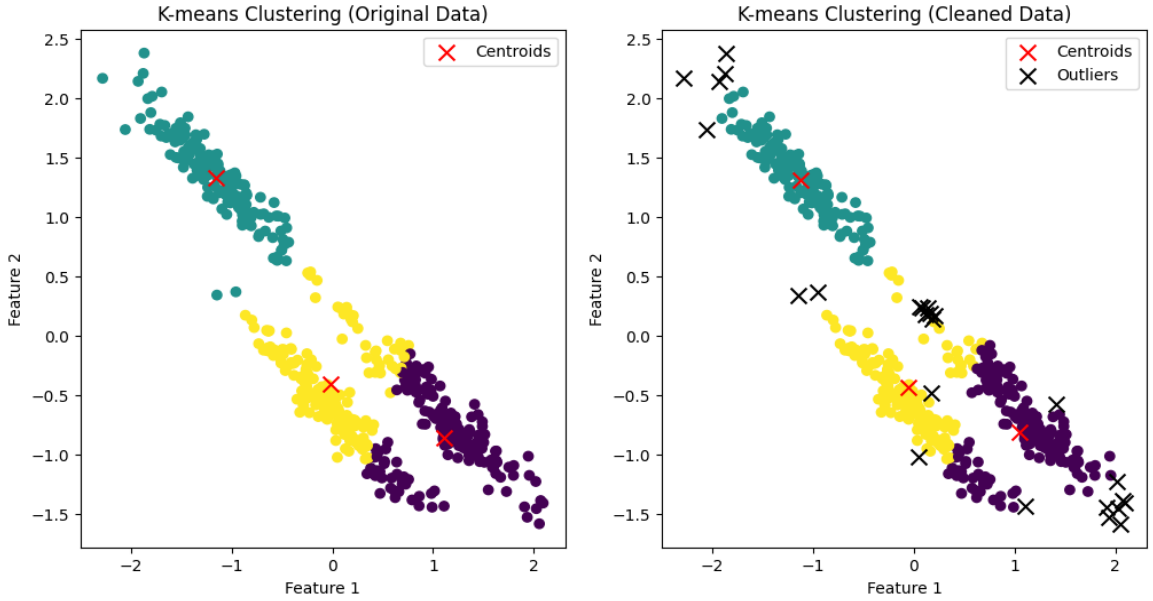
- Datasets contain three distributed data groups

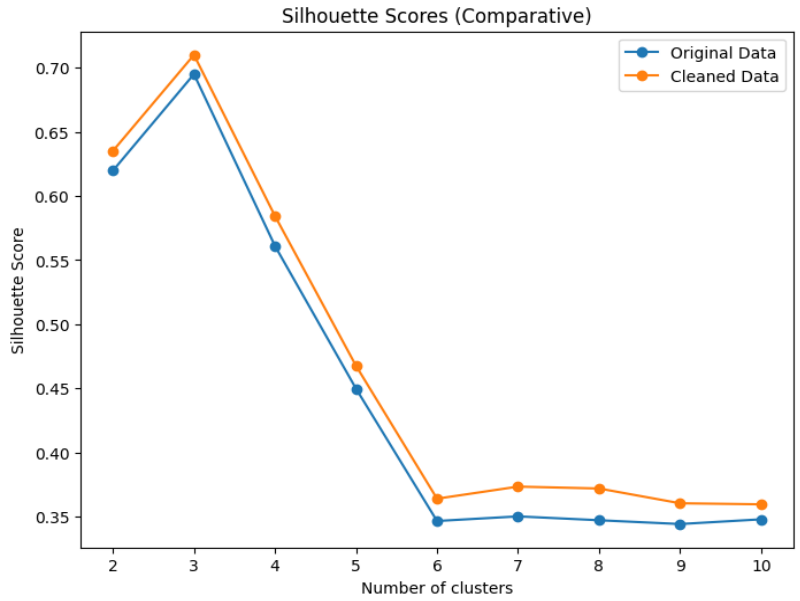




Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.6268	0.6328
3	0.6575	0.6842
4	0.6403	0.6612
5	0.6270	0.6464
6	0.5025	0.5270
7	0.4853	0.5032
8	0.4825	0.5246
9	0.4943	0.5110
10	0.4910	0.3571

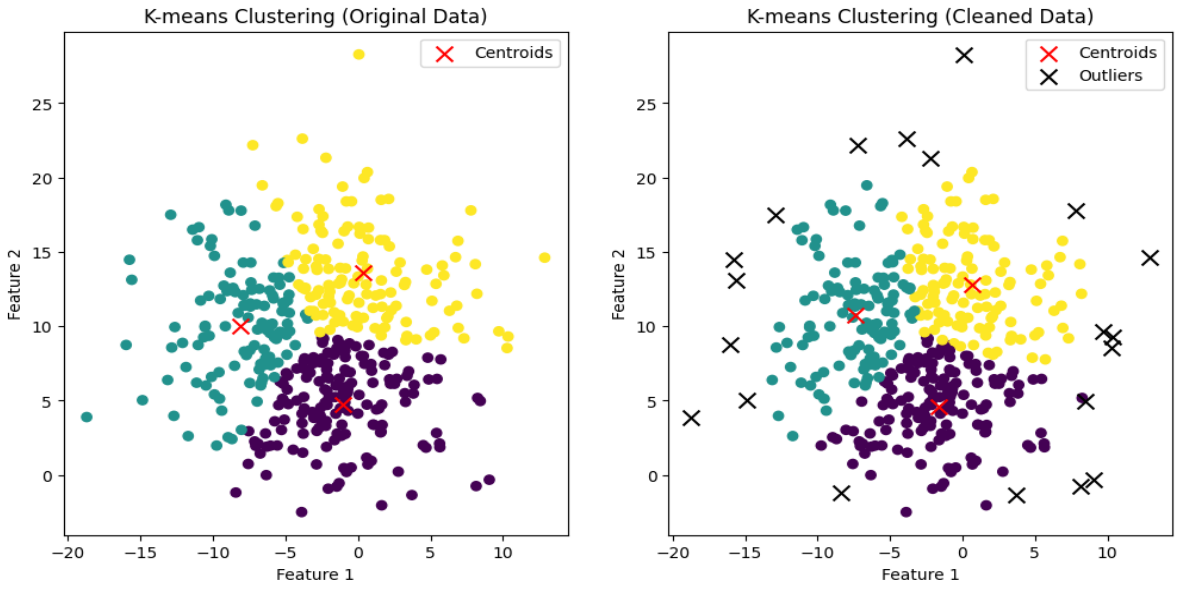
■ Datasets contain a group of data arranged in three lines

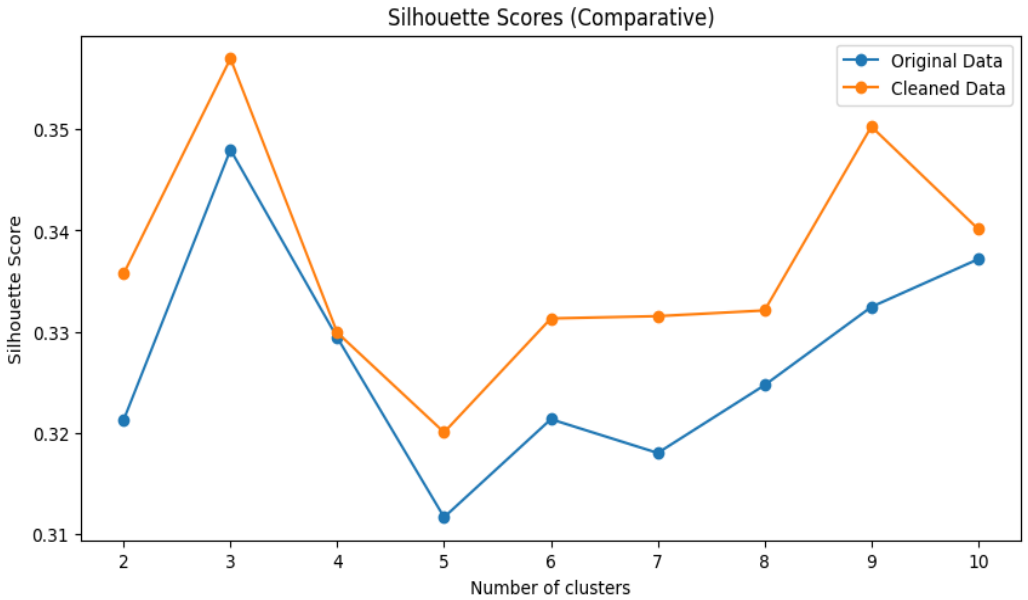




Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.6197	0.6351
3	0.6948	0.7099
4	0.5610	0.5844
5	0.4499	0.4677
6	0.3468	0.3642
7	0.3504	0.3736
8	0.3474	0.3721
9	0.3444	0.3606
10	0.3481	0.3598

■ Datasets contain a large group of distributed data

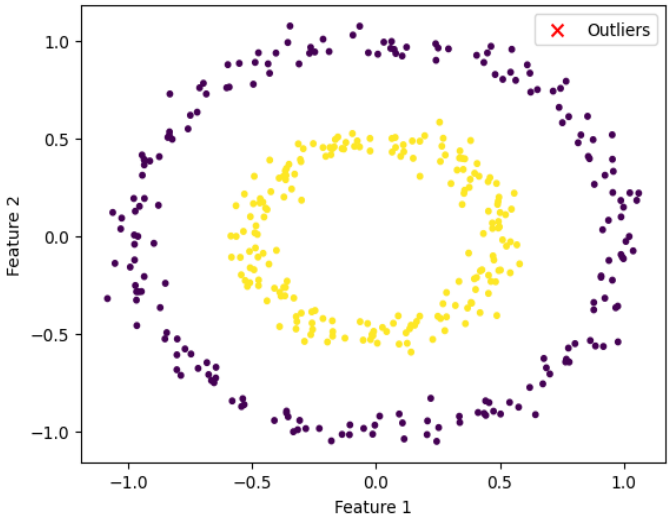


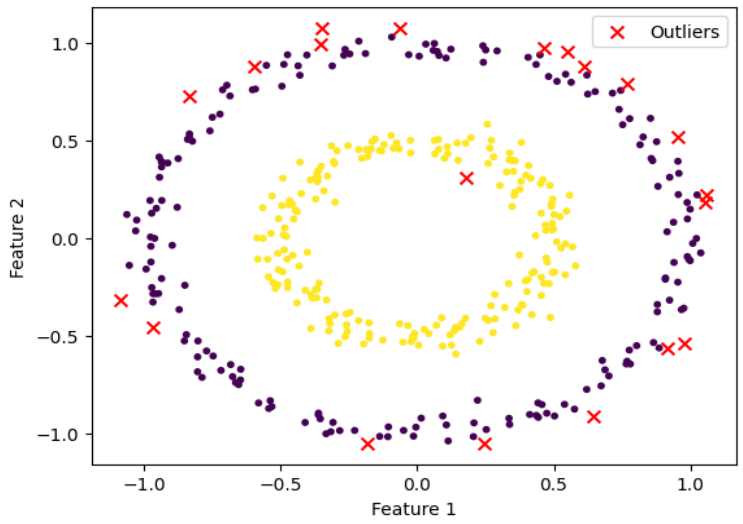


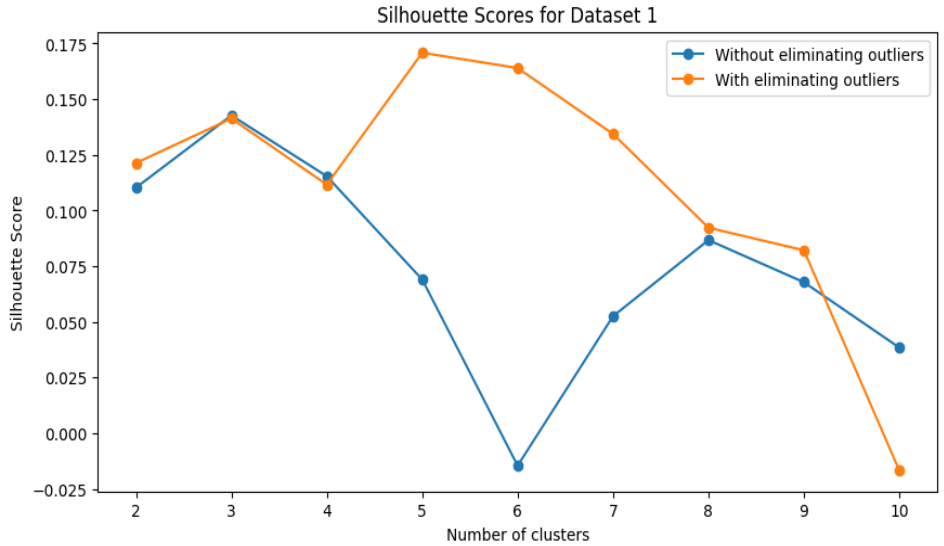
Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.3212	0.3357
3	0.3480	0.3570
4	0.3294	0.3299
5	0.3116	0.3200
6	0.3213	0.3313
7	0.3180	0.3315
8	0.3247	0.3321
9	0.3324	0.3503
10	0.3372	0.3401

Agglomerative single-linkage clustering with 95 percentile threshold

- Datasets contain double circle clusters

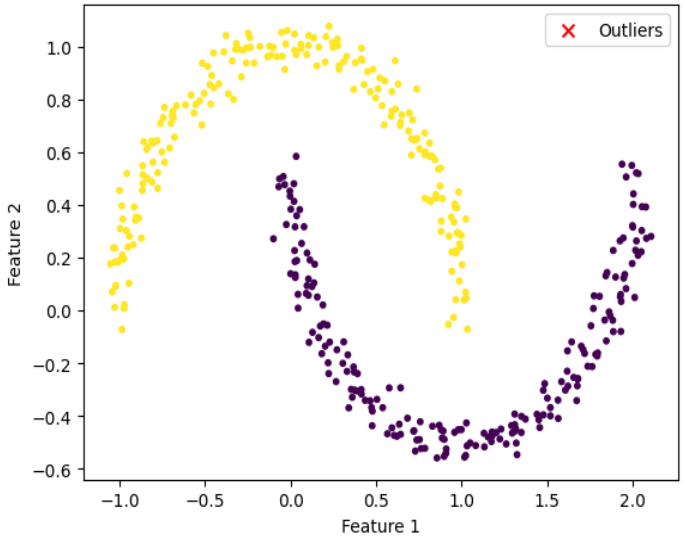


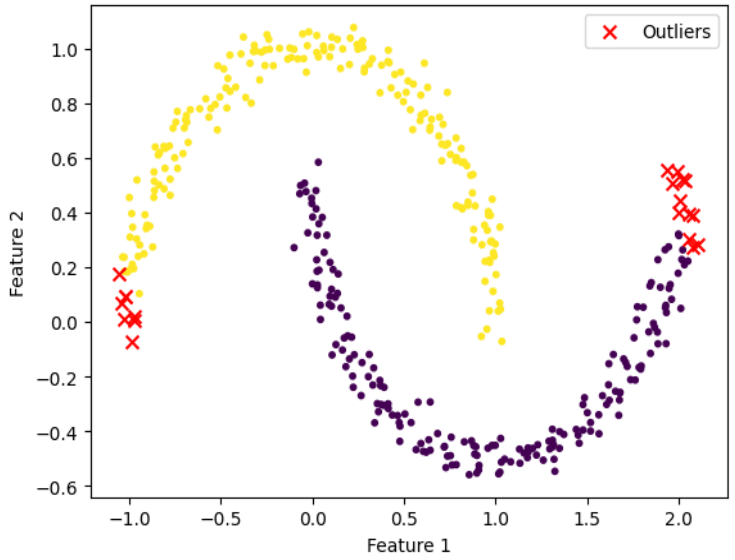


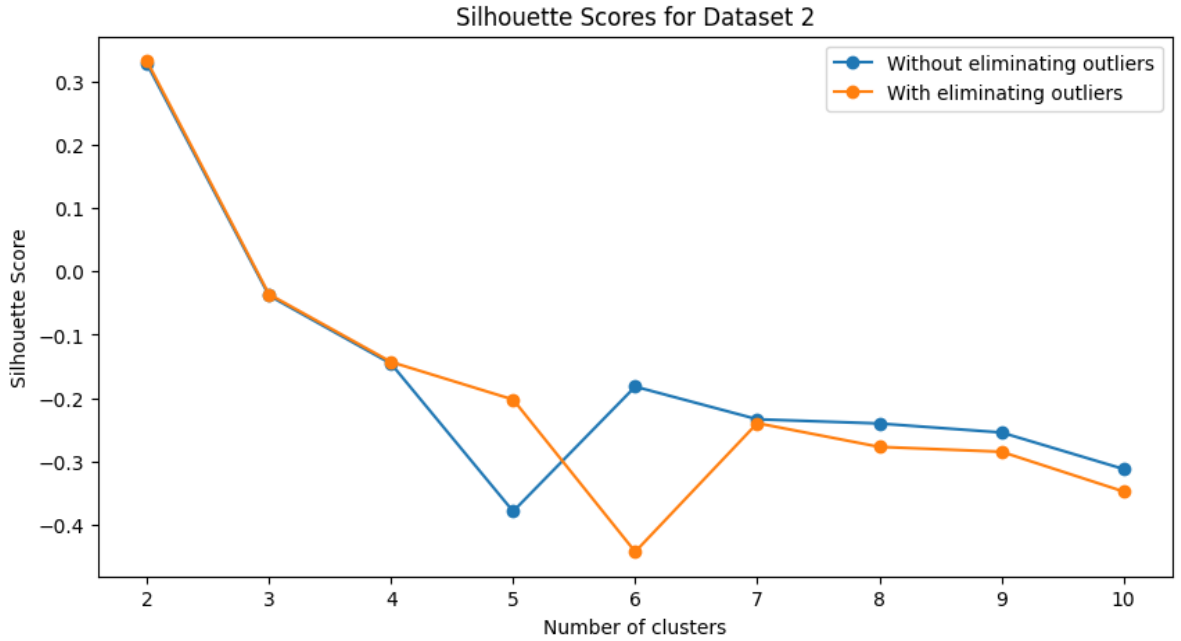


Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.1103	0.1213
3	0.1427	0.1413
4	0.1154	0.1116
5	0.0690	0.1708
6	-0.0145	0.1634
7	0.0526	0.1344
8	0.0867	0.0923
9	0.0678	0.0821
10	0.0385	-0.0167

■ Datasets contain double moon clusters

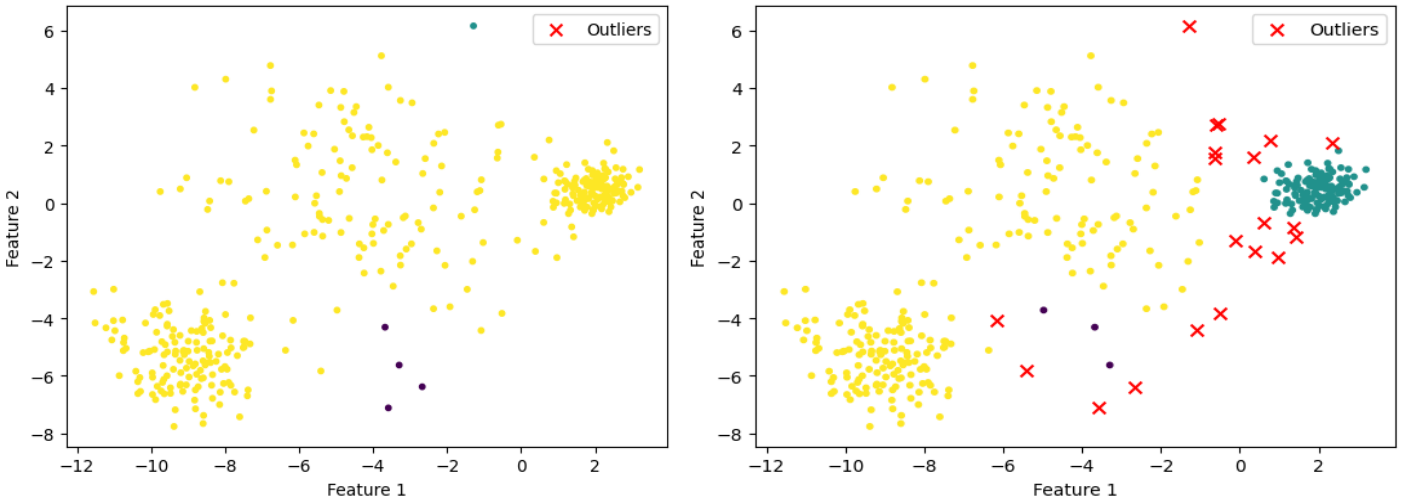


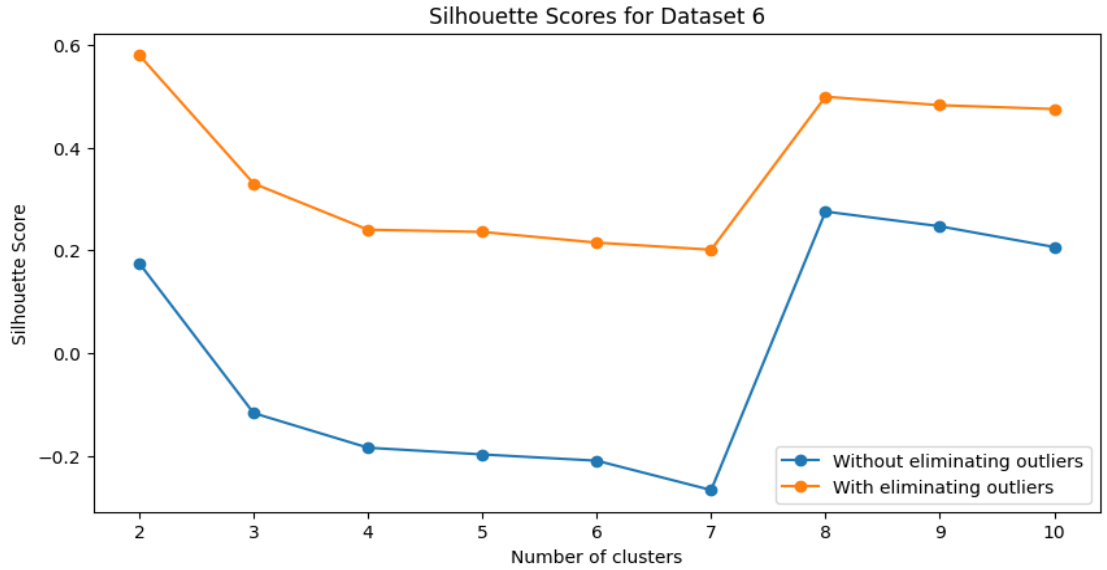




Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.3281	0.3325
3	-0.0381	-0.0359
4	-0.1457	-0.1430
5	-0.3790	-0.2024
6	-0.1820	-0.4429
7	-0.2336	-0.2397
8	-0.2402	-0.2772
9	-0.2544	-0.2849
10	-0.3121	-0.3477

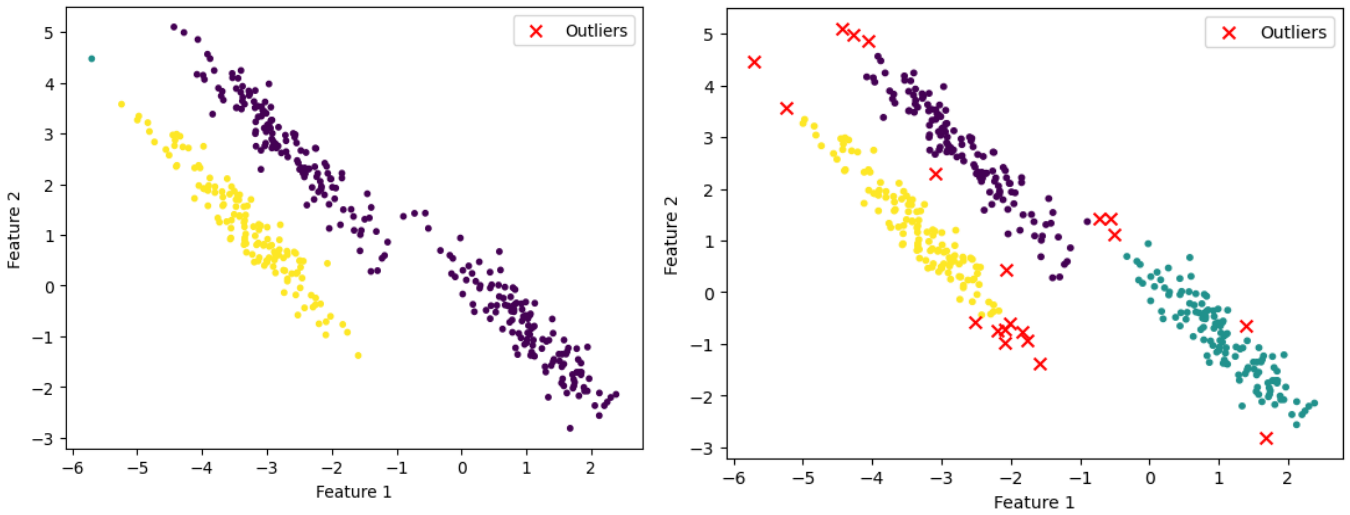
■ Datasets contain three distributed data groups

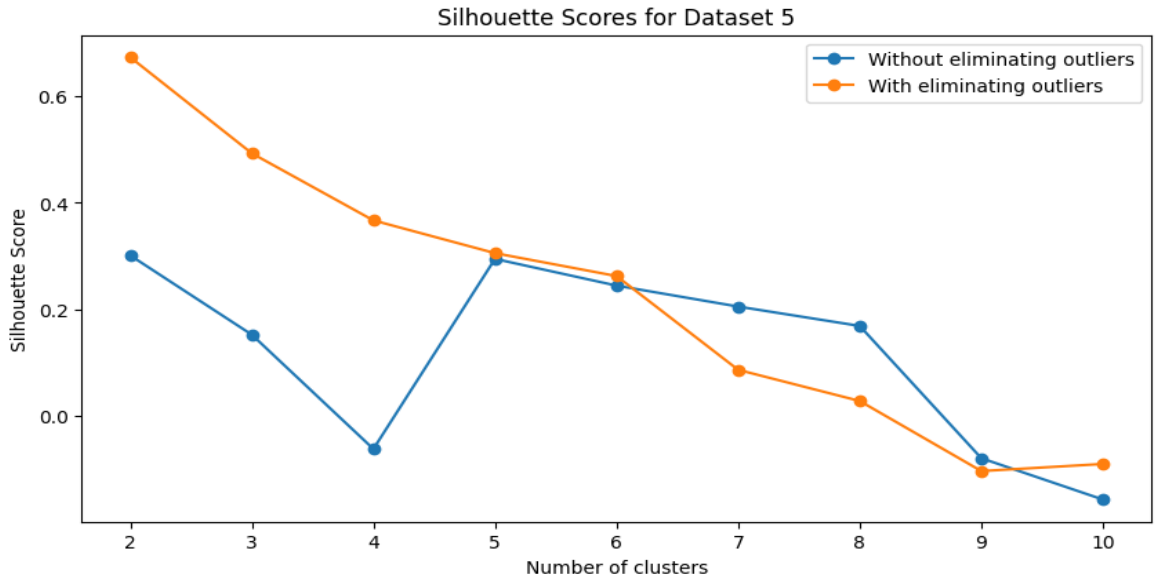




Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.1750	0.5797
3	-0.1165	0.3300
4	-0.1838	0.2402
5	-0.1969	0.2358
6	-0.2091	0.2150
7	-0.2662	0.2013
8	0.2755	0.4988
9	0.2468	0.4824
10	0.2064	0.4749

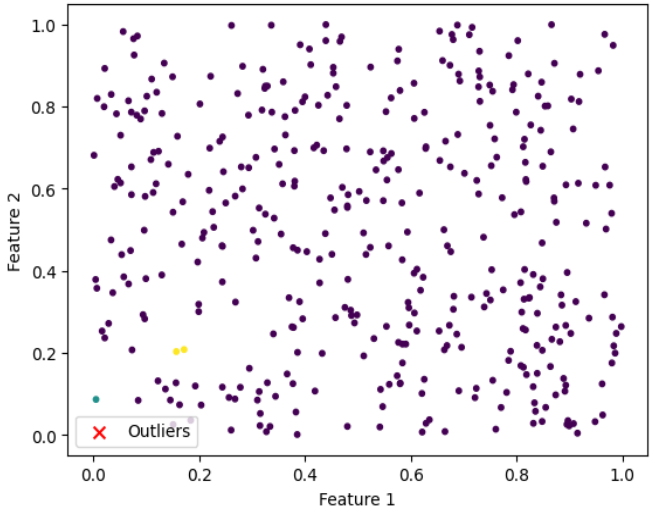
■ Datasets contain double moon clusters

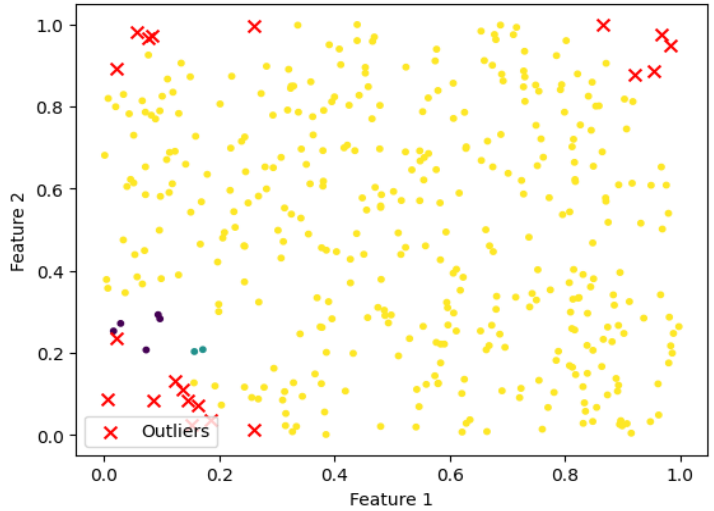


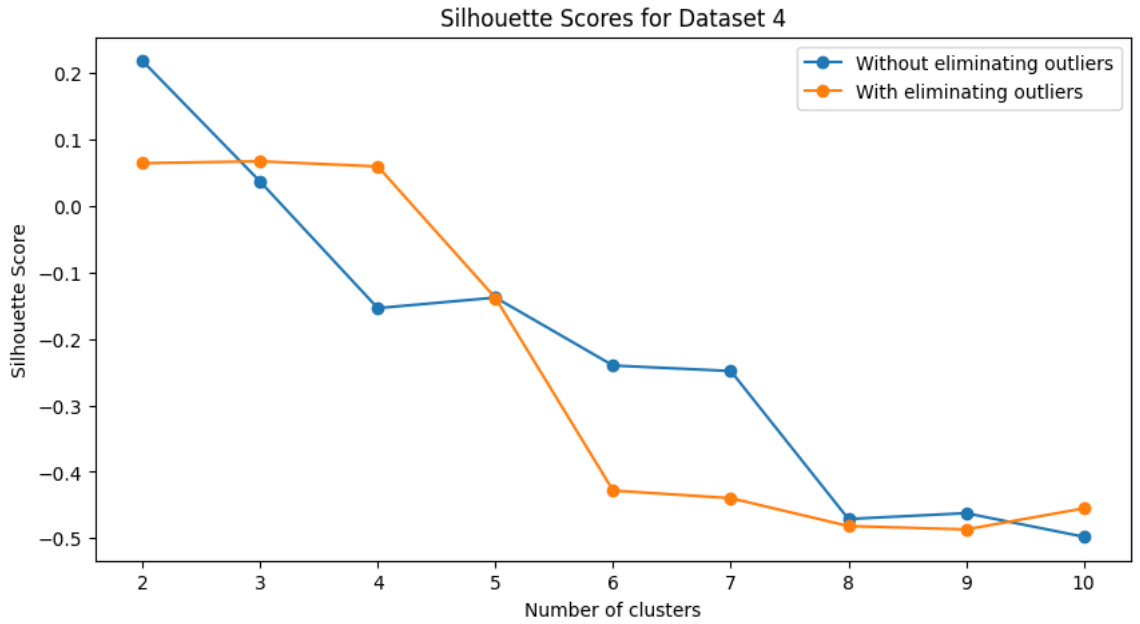


Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.3015	0.6728
3	0.1529	0.4928
4	-0.0612	0.3671
5	0.2949	0.3057
6	0.2448	0.2628
7	0.2056	0.0870
8	0.1693	0.0286
9	-0.0786	-0.1026
10	-0.1560	-0.0893

■ Datasets contain a large group of distributed data







Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.2183	0.0644
3	0.0368	0.0673
4	-0.1536	0.0595
5	-0.1378	-0.1383
6	-0.2240	-0.4281
7	-0.2483	-0.4396
8	-0.4709	-0.4816
9	-0.4622	-0.4867
10	-0.4977	-0.4549

Spotify dataset

This Spotify data comes from Kaggle and contains 42,305 records.

Feature of Spotify dataset:

- Danceability (0-1): This is a measure of how suitable a song is for dancing.
- Energy (0-1): Represents the intensity and activity of a song.
- Key (0-11): Indicates the estimated key of the song, using the pitch class notation.
- Loudness (-60 to 0 dB): Measures the overall loudness of a track in decibels (dB).
- Mode (0 or 1): Indicates the modality (major or minor) of a track.
- Speechiness (0-1): Measures the presence of spoken words in a track.
- Acousticness (0-1): A measure of how acoustic a song is.
- Instrumentalness (0-1): Predicts whether a track contains vocals or not.
- Liveness (0-1): Detects the presence of a live audience in the recording.
- Valence (0-1): Describes the musical positiveness or happiness conveyed by a track.
- Tempo (0-250 BPM): The overall estimated tempo of a song in beats per minute (BPM).

Other data Features

- Type, id, uri, track_href, analysis_url: These fields contain metadata about the song, such as its type, unique identifiers, and URLs
- Duration_ms: The duration of the track in milliseconds.
- Time_signature (1-7): Indicates the estimated number of beats in each bar of the song.
- Song_name, title, Unnamed: 0: These fields provide information about the song's title or name and may contain other identifying information or indexes.
- Genre: A categorization designation describing the song's musical style or genre. Underground rap, Dark trap, Trance, Techhouse, Psytrance, Dnb, Trap, Hiphop, Techno, RnB, Trap metal, Emo, Rap, Hardstyle, and Pop are some of the music genres available on Spotify.

	Daceability	Energy	Loudness	Speechiness	Acousticness	Intrumentalness	Liveness	valence	Tempo	Genre
0	0.831	0.814	-7.364	0.4200	0.0598	0.0134	0.0556	0.3890	156.98	Dark Trap
1	0.719	0.493	-7.230	0.0794	0.4010	0	0.1180	0.1240	115.08	Dark Trap
2	0.850	0.893	-4.783	0.0623	0.0138	0	0.3720	0.0391	218.05	Dark Trap
3	0.476	0.781	-4.710	0.1030	0.0237	0	0.1140	0.1750	186.95	Dark Trap
4	0.798	0.624	-7.668	0.2930	0.2170	0	0.1660	0.5910	147.99	Dark Trap
...	...	...	...	...	...	...	...	...	...	...
42300	0.528	0.693	-5.148	0.0304	0.0315	0.003	0.1210	0.3940	150.01	Hardstyle
42301	0.517	0.768	-7.922	0.0479	0.0225	0	0.0205	0.3830	149.93	Hardstyle
42302	0.361	0.821	-3.102	0.0505	0.0260	0.002	0.3850	0.1240	154.93	Hardstyle
42303	0.477	0.921	-4.777	0.0392	0.0005	0.0296	0.0575	0.4880	150.04	Hardstyle
42304	0.529	0.945	-5.862	0.0615	0.0019	0	0.4140	0.1340	155.05	Hardstyle

Systematic random sampling data

Systematic random sampling is a way to select a subset from a larger group. Follow three steps:

1. Sampling Interval (k): Take the total number of items (N) and divide by how many you want (n), so k = N/n. If k isn't whole, round it.
2. Random Start: Pick a random starting number between 1 and k. This is your first item.
3. Select Items: From the start, select every k-th item. If you hit the end, loop back.

This method ensures a mathematically balanced representation of the larger group.

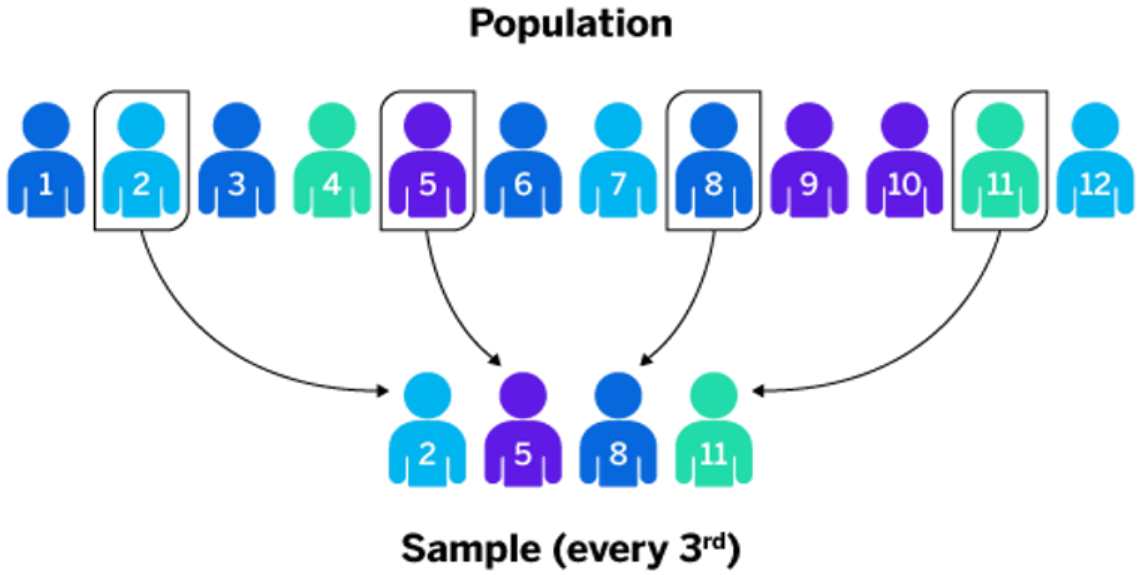
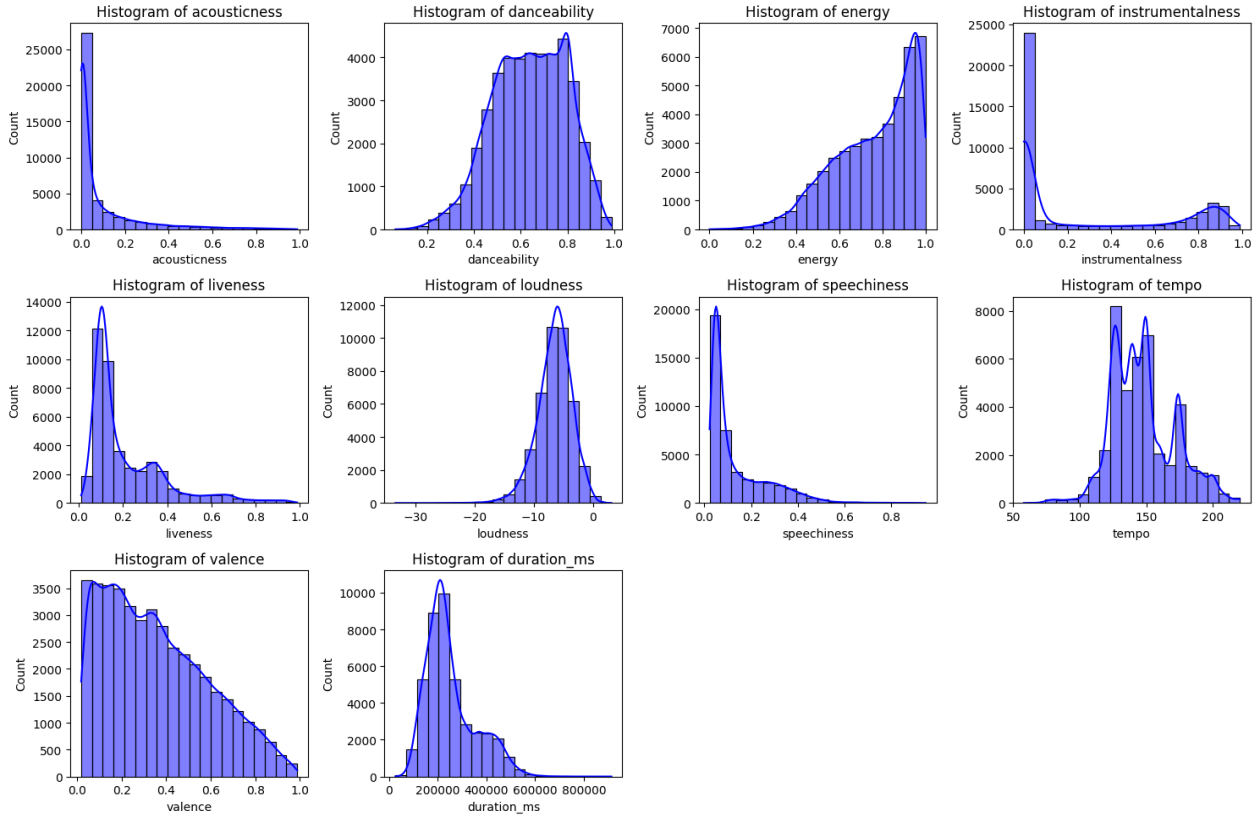
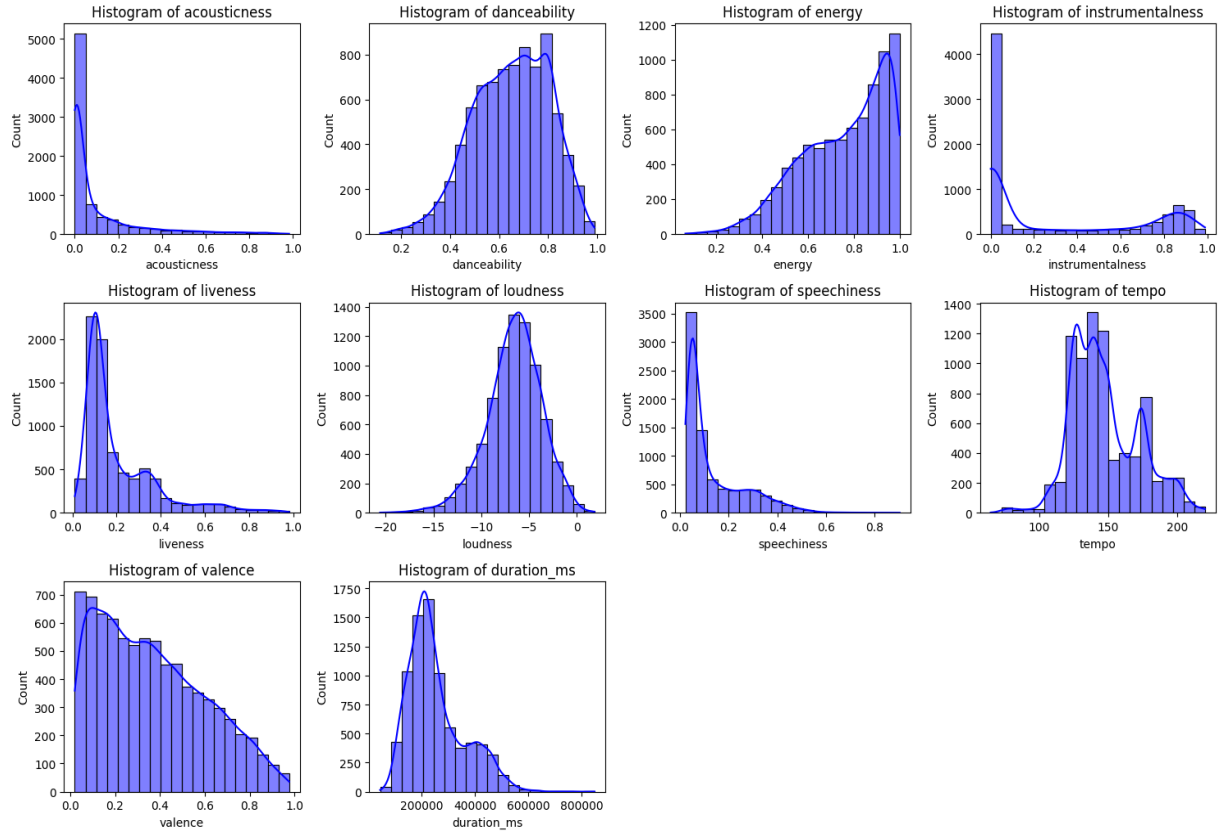


Fig source : https://www.qualtrics.com/uk/experience-management/research/systematic-random-sampling/



Histogram before sampling data



Histogram after sampling data

Remove outlier

- Set the MOF algorithm's outlier detection threshold to the 95th percentile.

	Daceability	Energy	Loudness	Speechiness	Acousticness	Intrumentalness	Liveness	valence	Tempo
Count	400	400	400	400	400	400	400	400	400
Mean	0.6621	0.5763	-9.4299	0.1725	0.2259	0.1769	0.2014	0.3450	133.5978
Std	0.1676	0.2320	5.5590	0.1460	0.2649	0.3324	0.1573	0.2335	43.8383
Min	0.1470	0.0127	-32.9290	0.0260	0	0	0.0338	0.0294	61.3090
Max	0.9610	0.996	-3.0800	0.7650	0.987	0.968	0.943	0.9700	220.2160

	Daceability	Energy	Loudness	Speechiness	Acousticness	Intrumentalness	Liveness	valence	Tempo
Count	7600	7600	7600	7600	7600	7600	7600	7600	7600
Mean	0.65	0.76	-6.44	0.14	0.09	0.30	0.21	0.36	148.21
Std	0.15	0.18	2.70	0.12	0.17	0.38	0.17	0.24	22.79
Min	0.12	0.12	-18.13	0.02	0.00	0.00	0.01	0.02	107.93
Max	0.99	1.00	1.42	0.90	0.98	0.99	0.98	0.98	209.88

Min-Max Scaling data

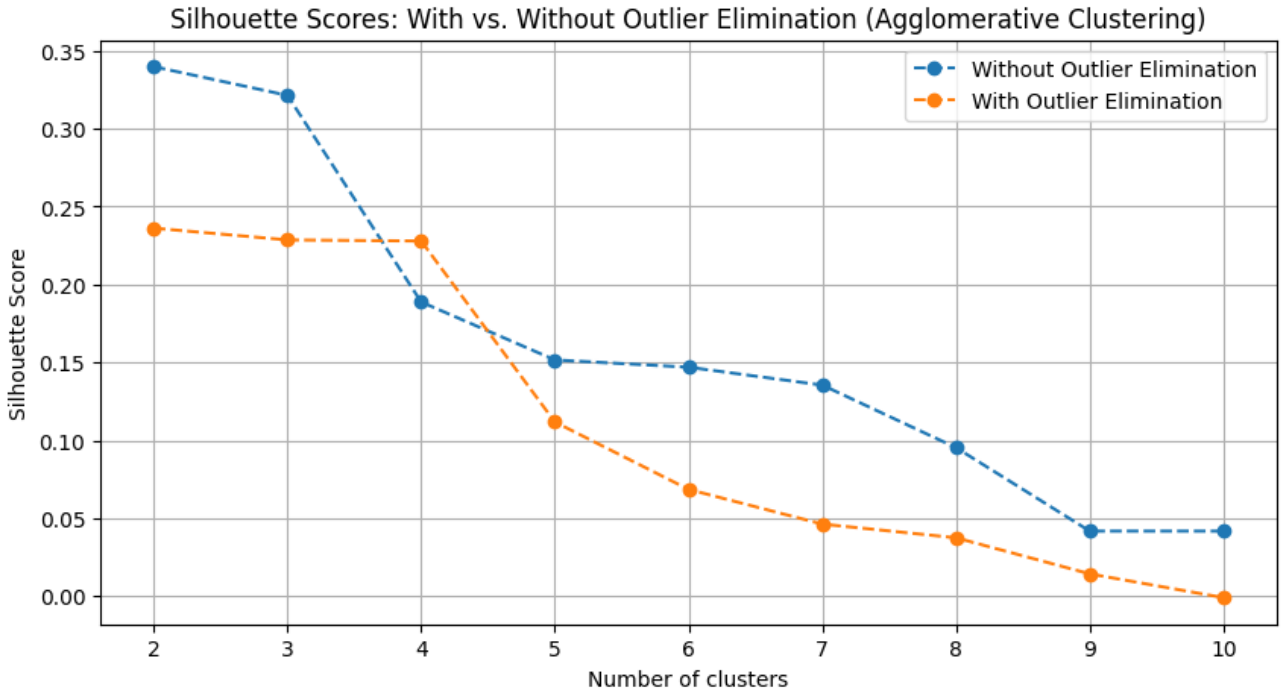
The Min-Max Scaling is one of the simplest methods to scale the features, where the values are scaled to a fixed range between 0 to 1. The formula used in Min-Max scaling is:

$$X_{scaled} = (X - X_{min}) / (X_{max} - X_{min})$$

Where:

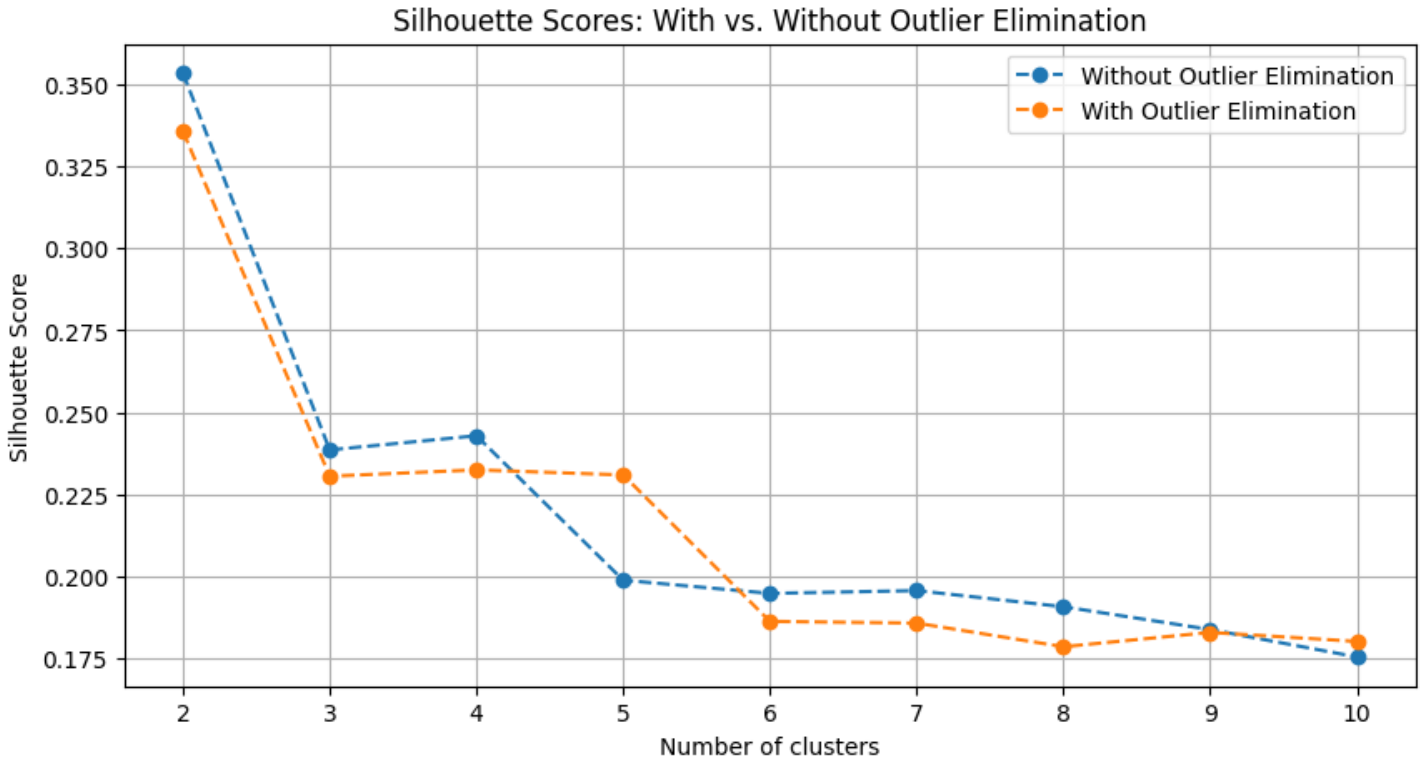
- X is the original feature value.
- X_{min} is the minimum value in that feature.
- X_{max} is the maximum value of that feature.
- X_{scaled} is the new value for that feature after scaling.

Agglomerative single-linkage clustering

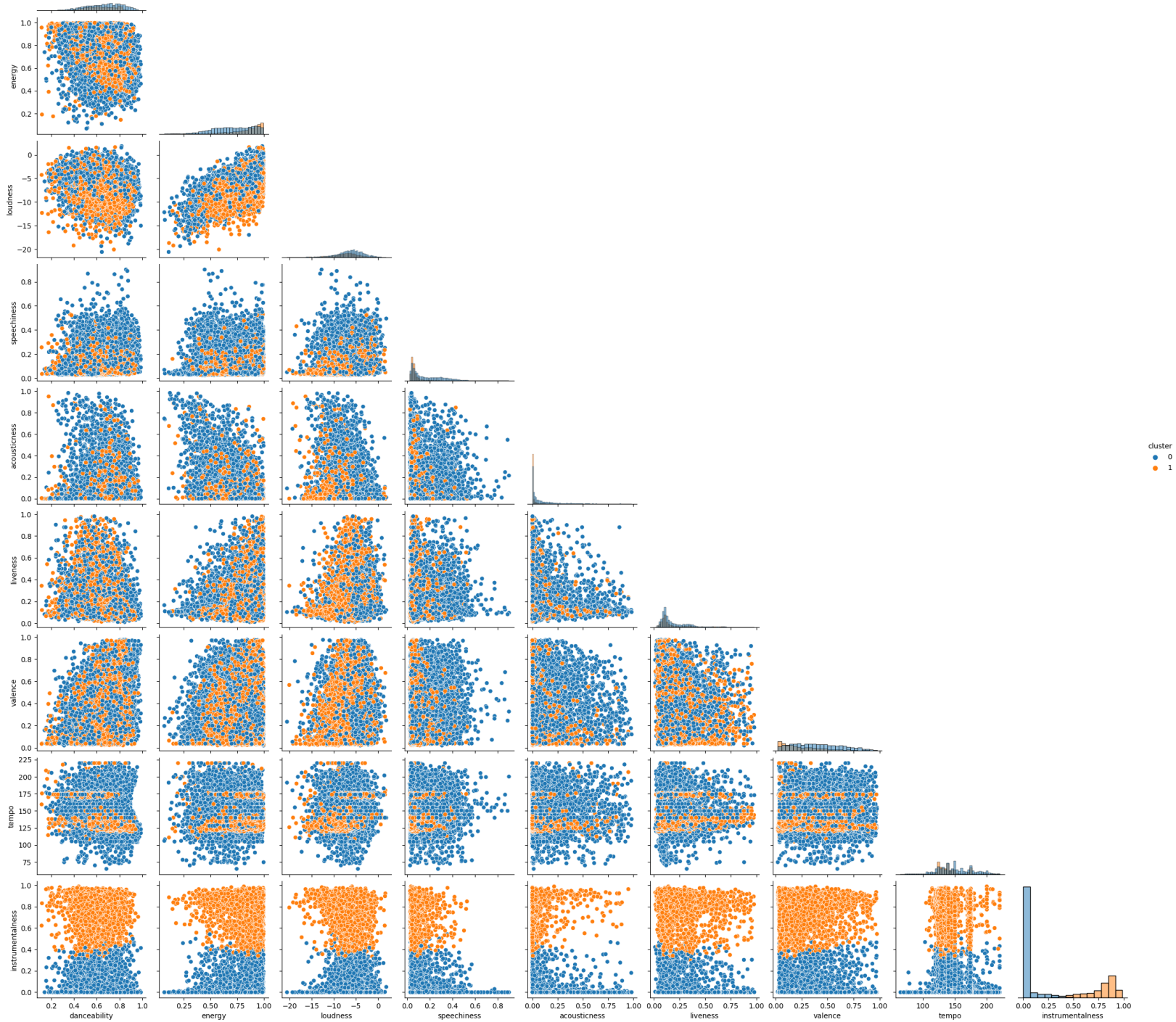


Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.3398	0.2362
3	0.3214	0.2286
4	0.1888	0.2279
5	0.1516	0.1117
6	0.1469	0.0684
7	0.1354	0.0462
8	0.0953	0.0374
9	0.0417	0.0143
10	0.0417	-0.0010

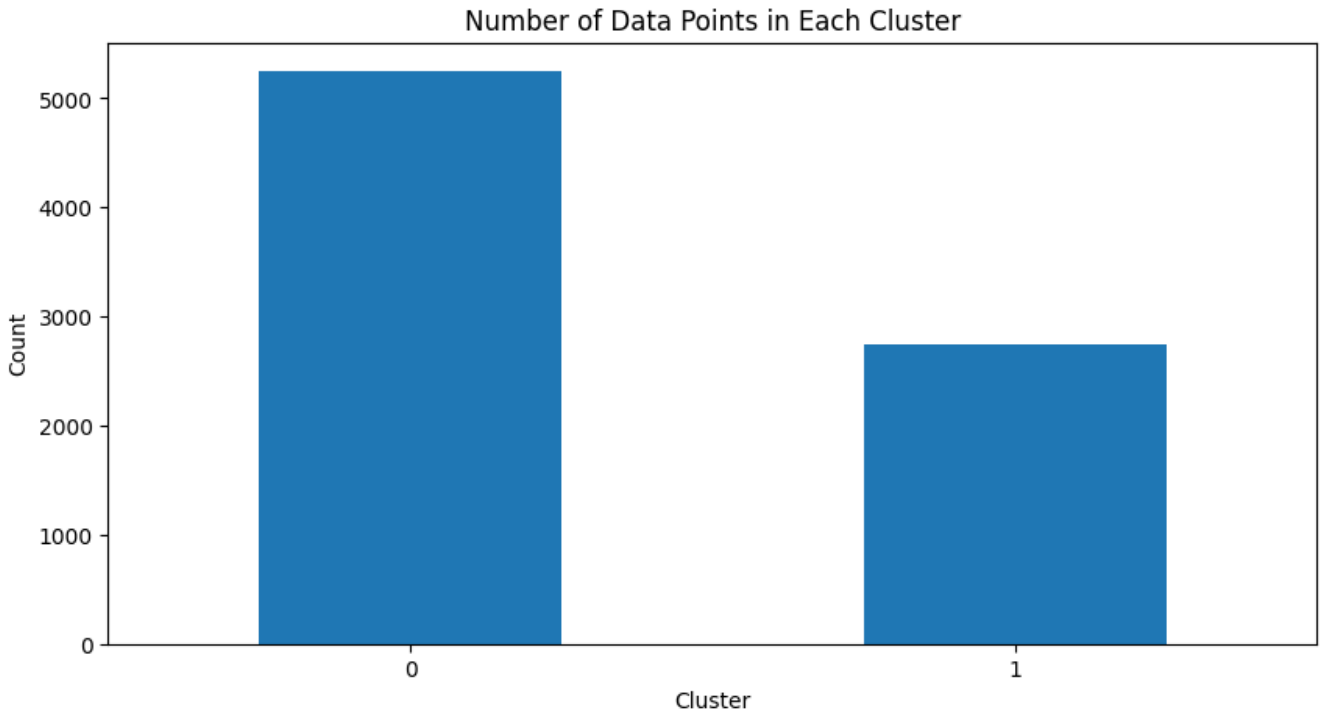
K-mean



Number of cluster	Without eliminating outliers	With eliminating outliers
2	0.3534	0.3356
3	0.2385	0.2305
4	0.2428	0.2324
5	0.1988	0.2309
6	0.1948	0.1862
7	0.1956	0.1857
8	0.1907	0.1785
9	0.1838	0.1829
10	0.1754	0.1801


cluster
0
1
danceability
energy
loudness
speechiness
acousticness
liveness
valence
tempo
instrumentalness

Analysis spotify dataset



Cluster	Number of data points
0	5249
1	2751

Danceability and Energy features

Range of Danceability	Cluster 0	Cluster 1
0.12 – 0.29	62	45
0.29 – 0.46	573	295
0.46 – 0.64	1558	1078
0.64 – 0.81	2056	1170
0.81 – 1.00	998	163

Range of Energy	Cluster 0	Cluster 1
0.07 – 0.25	42	10
0.25 – 0.44	374	62
0.44 – 0.63	1353	248
0.63 – 0.81	1623	563
0.81 – 1.00	1857	1868

Loudness and Speechiness features

Range of Loudness	Cluster 0	Cluster 1
-20.58 - -16.1	16	18
-16.1 - -11.61	220	168
-11.61 - -7.12	1577	1113
-7.12 - -2.64	2967	1317
-2.64 - 1.85	469	135

Range of Speechiness	Cluster 0	Cluster 1
0.02 - 0.2	3332	2648
0.2 - 0.37	1420	80
0.37 - 0.55	452	23
0.55 - 0.73	34	0
0.73 - 0.9	11	0

Acousticness and Liveness features

Range of Acousticness	Cluster 0	Cluster 1
0.0 - 0.2	4074	2622
0.2 - 0.39	629	65
0.39 - 0.59	302	33
0.59 - 0.79	164	20
0.79 - 0.98	80	11

Range of Liveness	Cluster 0	Cluster 1
0.01 - 0.2	3458	1887
0.2 - 0.4	1251	513
0.4 - 0.59	315	150
0.59 - 0.79	170	143
0.79 - 0.98	55	58

Valence and Tempo features

Range of Valence	Cluster 0	Cluster 1
0.02 - 0.21	1251	1402
0.21 - 0.4	1473	673
0.4 - 0.59	1252	379
0.59 - 0.79	883	204
0.79 - 0.98	390	93

Range of Tempo	Cluster 0	Cluster 1
64.95 - 95.98	80	0
95.98 - 127.01	955	653
127.01 - 158.04	2290	1666
158.04 - 189.07	1378	387
189.07 - 220.1	546	45

Instrumentalness feature

Range of Instrumentalness	Cluster 0	Cluster 1
0.0 - 0.2	4891	0
0.2 - 0.4	324	42
0.4 - 0.59	34	316
0.59 - 0.79	0	664
0.79 - 0.99	0	1729

Genre of songs

Genre of songs	Cluster 0
Underground rap	1144
Dark Trap	631
Hiphop	594
Trap	459
RnB	418
Trap Metal	369
Rap	367
Emo	332
Trance	252
Dnb	250
Techhouse	206
Hardstyle	102
Pop	93
Psytrance	21
Techno	11

Genre of songs	Cluster 1
Techno	580
Psytrance	571
Techhouse	389
Trance	348
Dnb	343
Dark Trap	285
Trap	139
Underground rap	31
hardstyle	24
Trap Metal	22
Hiphop	11
Emo	4
Rap	3
RnB	3
Pop	0

Cluster 0 Traits

- Most dancability is in the range of 0.46 – 0.99, representing 87.86%.
- Most energy is in the range of 0.44 – 1, representing 92.07%.
- Most loudness is in the range of -11.61 to -2.64, representing 86.57%.
- Most speechiness is in the range of 0.02 – 0.37, representing 90.53%.
- Most acousticness is in the range of 0 – 0.2, representing 77.61 %.
- Most liveness is in the range of 0.01 – 0.4 , representing 89.71%.
- Most valence is in the range of 0.02 – 0.59 , representing 75.75 %.
- Most tempo is in the range of 95.98 – 189.07 , representing 88.07 %.
- Most intrumentalness is in the range of 0 – 0.2 , representing 93.18 %.
- Underground Rap, Dark Trap, Hiphop, trap, RnB, Trap Metal and Rap make up 75.86 % of the music genres.

Cluster 1 Traits

- Most dancability is in the range of 0.46 – 0.81, representing 81.71%.
- Most energy is in the range of 0.63 – 1 , representing 88.37%.
- Most loudness is in the range of -11.61 to -2.64 , representing 86.57%.
- Most speechiness is in the range of 0.02- 0.2 , representing 96.25%.
- Most acousticness is in the range of 0 - 0.2, representing 95.31%.
- Most liveness is in the range of 0.01 - 0.4, representing 87.24%.
- Most valence is in the range of 0.02 – 0.4, representing 75.43%.
- Most tempo spread in the range of 95.98 – 158.04, representing 84.30%.
- Most intrumentalness is in the range of 0.59 – 0.99, representing 86.99%.
- techno, psytrance, techhouse, trance and dnb make up 81.10% of the music genres.

Conclusion

Synthesis datasets

- The MOF anomaly detection algorithm contributes to clustering. However, this does not apply to large groups of data with a distributed shape that have been tested with Agglomerative Single-Linkage Clustering.

Sportify dataset

- The MOF anomaly detection algorithm may not operate effectively with k-means and agglomerative clustering single-linkage clustering.