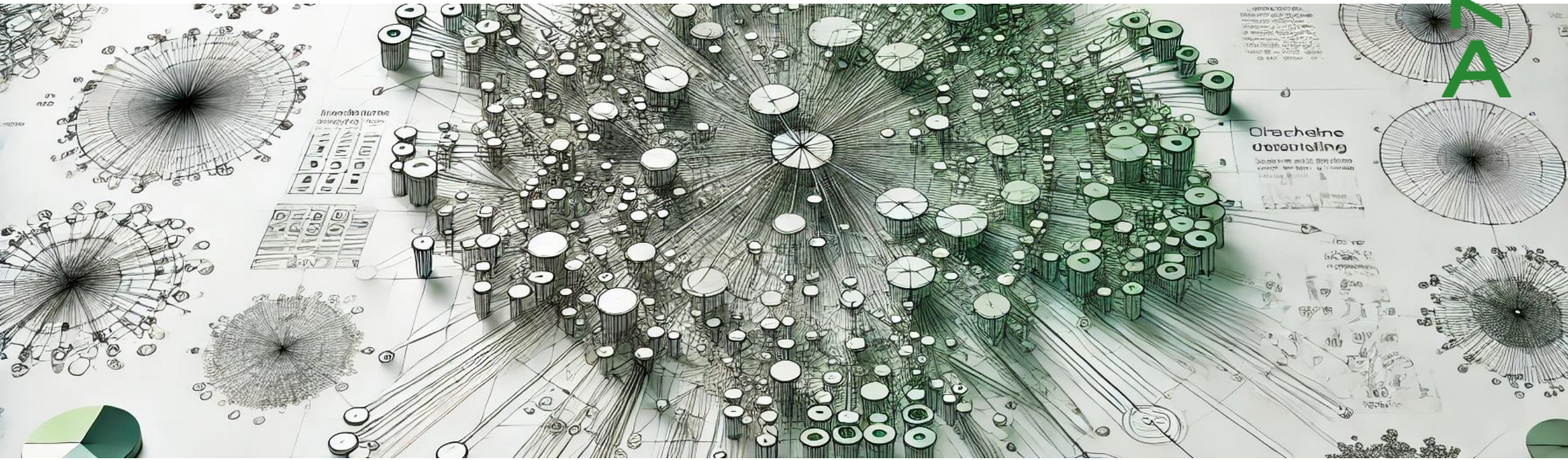
Hochschule Karlsruhe
University of
Applied Sciences

Fakultät für
Elektro- und
Informationstechnik

Clustering



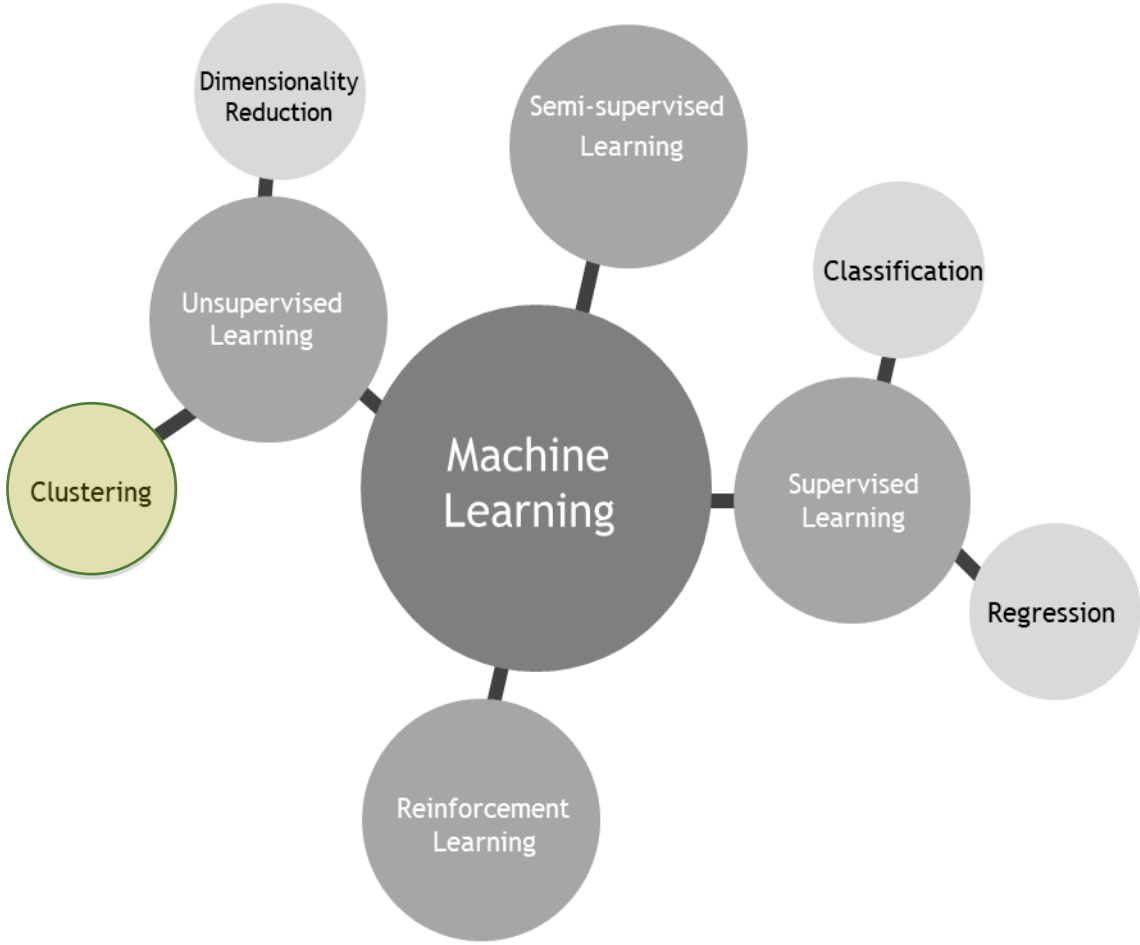
Source: DALL.E

Kawther Aboalam

Clustering

Basic Idea

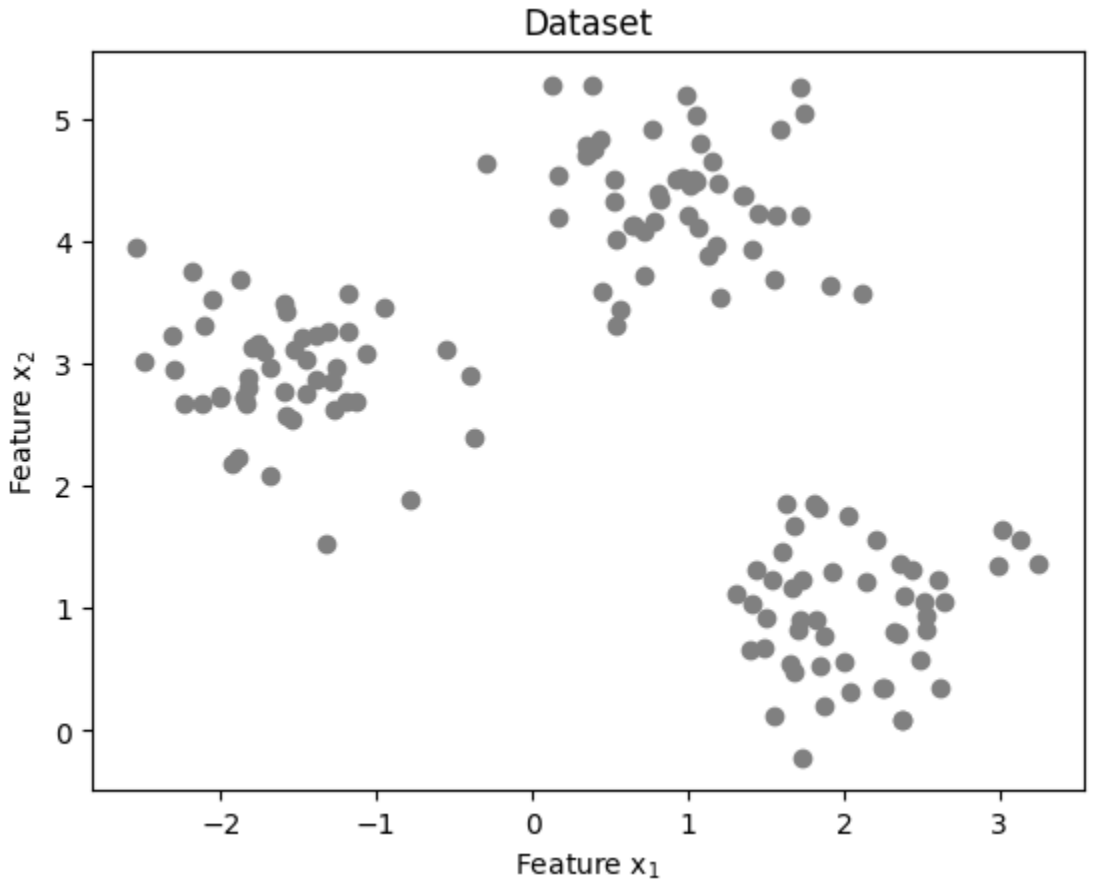
- Cluster analysis is used to discover similarity structures in datasets, whereby the group assignment is referred to as clustering
- Unlike regression and classification, clustering methods belong to the group of unsupervised learning
- Mechanisms for identifying similarity are based on different principles
  - Distance-based
  - Density-based
- Technical applications of clustering methods include anomaly detection in condition monitoring and pattern recognition in image processing
- Clustering methods can be used for pre-sorting of data, which is the starting point for labeling data points into classes



Clustering

K-Means Algorithm

+ K-Means algorithm is one of the most widely used clustering methods, it can be described clearly and implemented easily
+ Procedure searches for cluster centers that represent specific regions within the data
+ K-Means algorithm divides the one label N data points X into a predefined number of clusters K

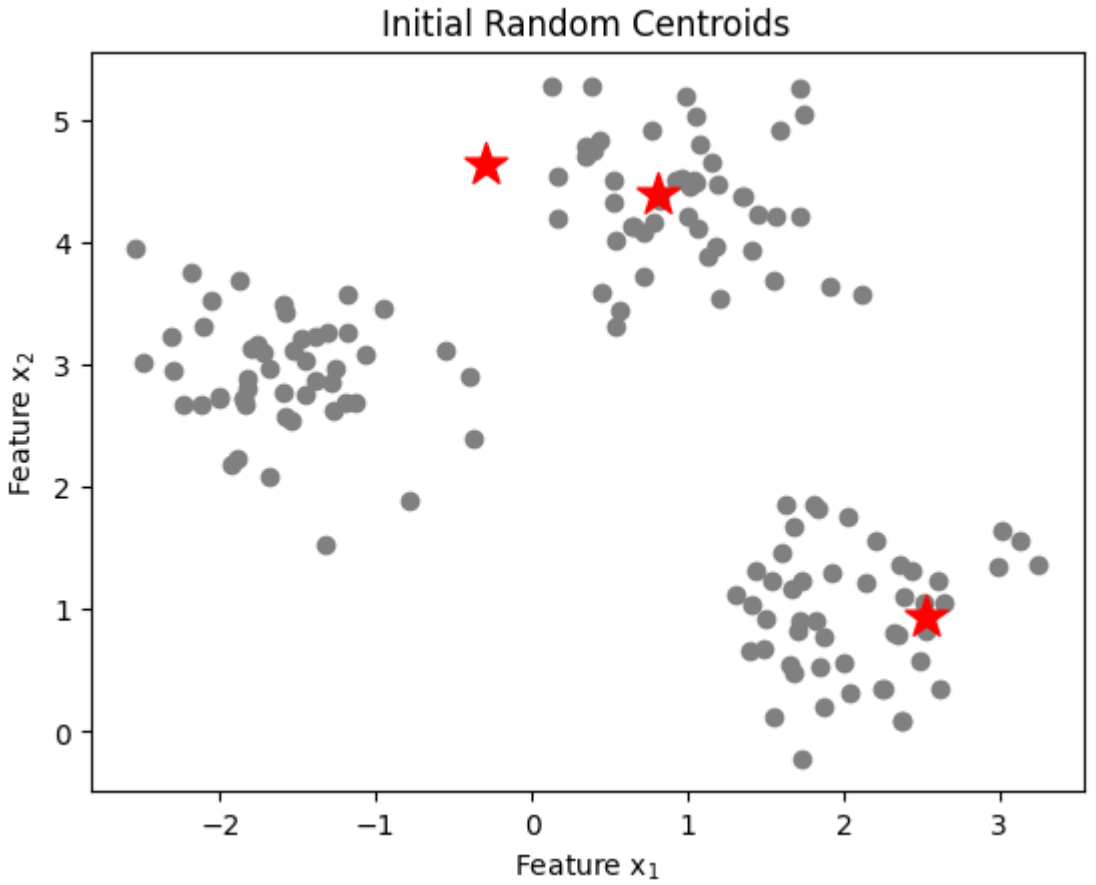


Clustering

K-Means Algorithm

- K-Means algorithm is one of the most widely used clustering methods, it can be described clearly and implemented easily
- Procedure searches for cluster centers that represent specific regions within the data
- K-Means algorithm divides the one label data points X into a predefined number of clusters K
- Steps:

1. Select randomly K data points as initial centroids



Clustering

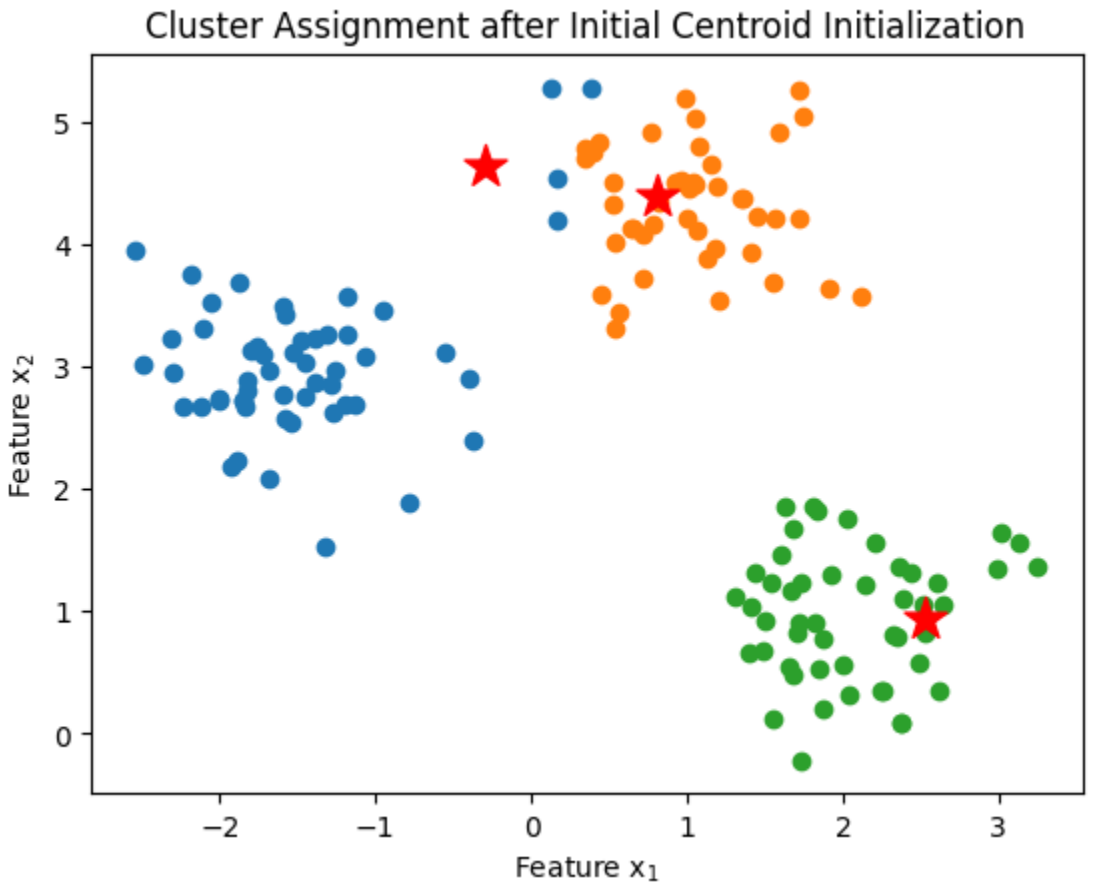
K-Means Algorithm

- K-Means algorithm is one of the most widely used clustering methods, it can be described clearly and implemented easily
- Procedure searches for cluster centers that represent specific regions within the data
- K-Means algorithm divides the one label data points X into a predefined number of clusters K
- Steps:

1. Select randomly K data points as initial centroids
2. Assign each data point x_i to the nearest centroid c_k. This forms K clusters C_k, where k $\in \{1, 2, \dots, K\}$ and x_i is the i[th] data point in the dataset

$$x_i \in C_k \quad If \quad k = \arg min_k \|x_i - c_k\|^2$$

Where, $\|x_i - c_k\|^2$, is the Squared Euclidean distance



Clustering

K-Means Algorithm

+ Steps:

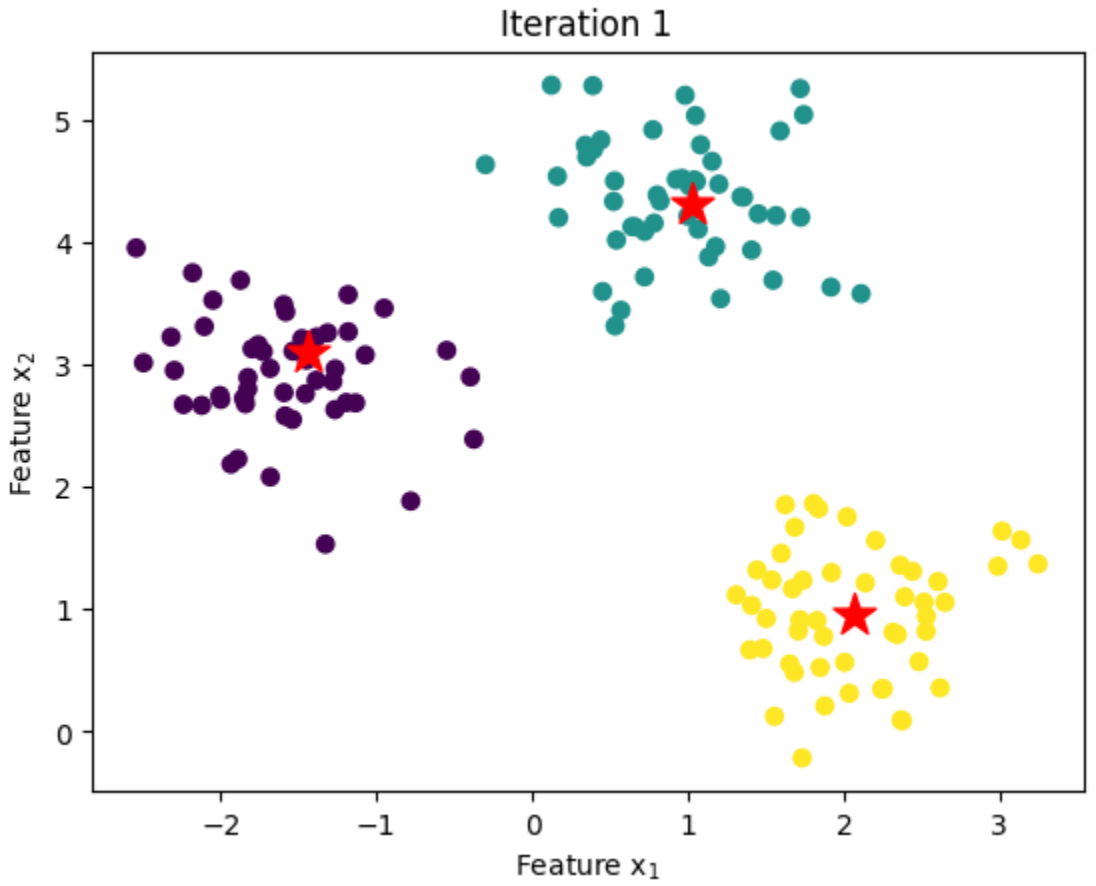
1. Select randomly K data points as initial centroids
2. Assign each data point x_i to the nearest centroid c_k. This forms K clusters C_k, where k $\in \{1, 2, \dots, K\}$ and x_i is the i[th] data point in the dataset

   $$x_i \in C_k \quad If \quad k = \arg min_k \|x_i - c_k\|^2$$

   Where, $\|x_i - c_k\|^2$, is the Squared Euclidean distance
3. Update each cluster center c_k to be the average of all data points assigned to that cluster C_k

$$c_k = \frac{1}{|C_k|} \cdot \sum_{x_i \in C_k} x_i$$

+ Where, $|C_k|$ is the number of data points in cluster number k

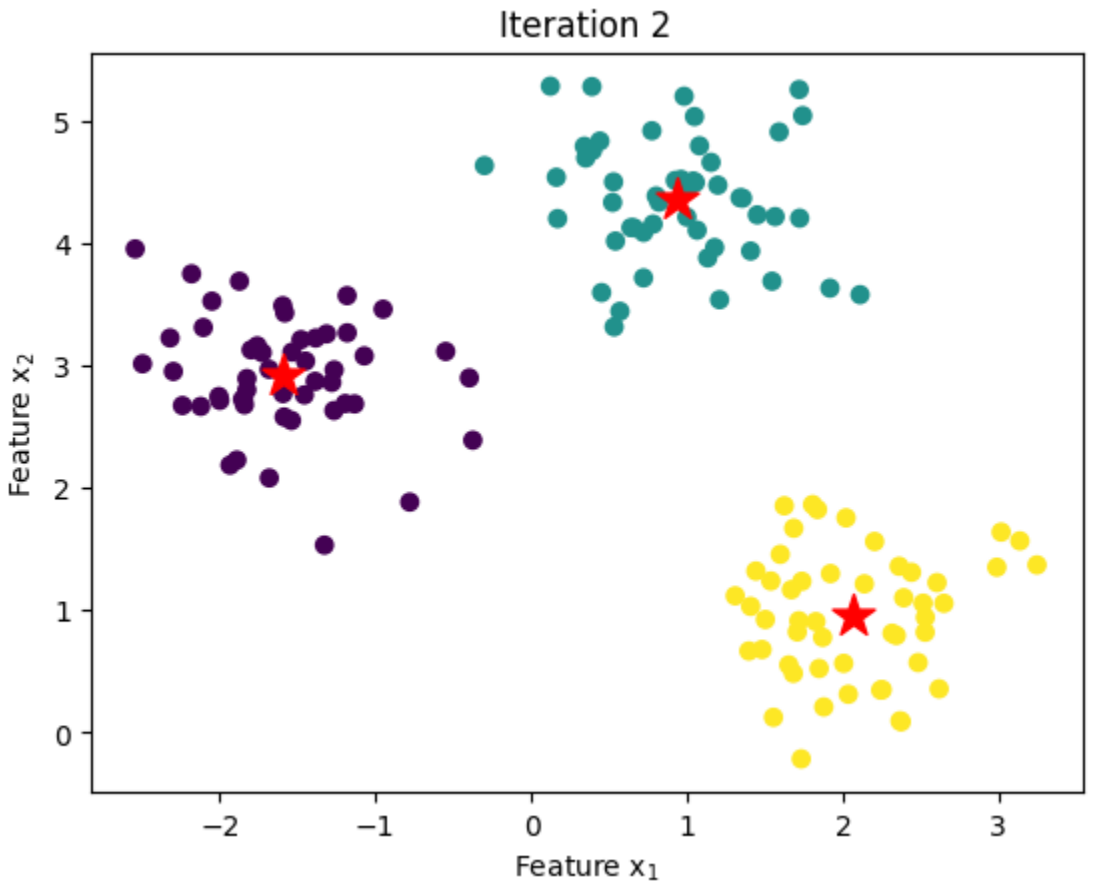


Clustering

K-Means Algorithm

+ Steps:

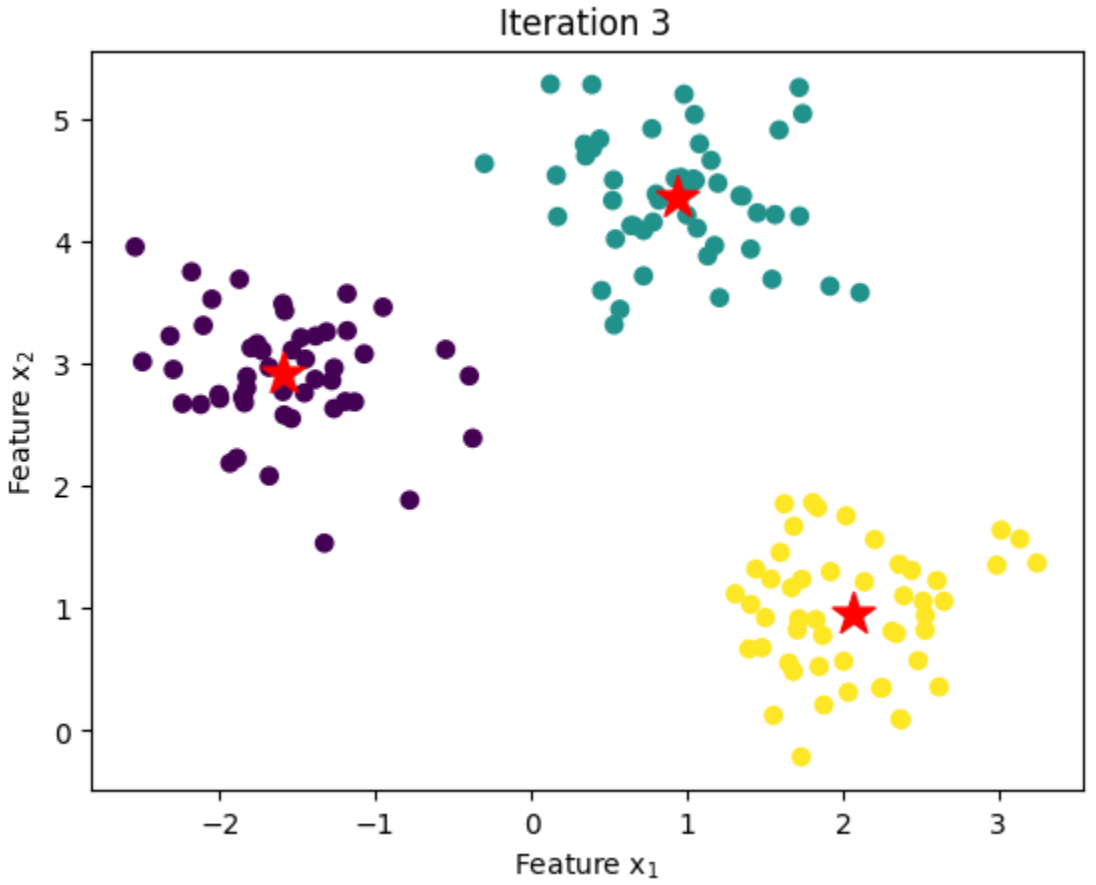
    4. Repeat the steps 2 and 3



Clustering

K-Means Algorithm

+ Steps:
    4. Repeat the steps 2 and 3
    5. Convergence Check: repeat steps 2 and 3 until the cluster centers do not change significantly. That means that the difference in the cluster centers of two consecutive iterations is zero or a small value called tolerance or sometimes called threshold.
+ After iteration number two there is no significant change for the cluster centers by the shown example.



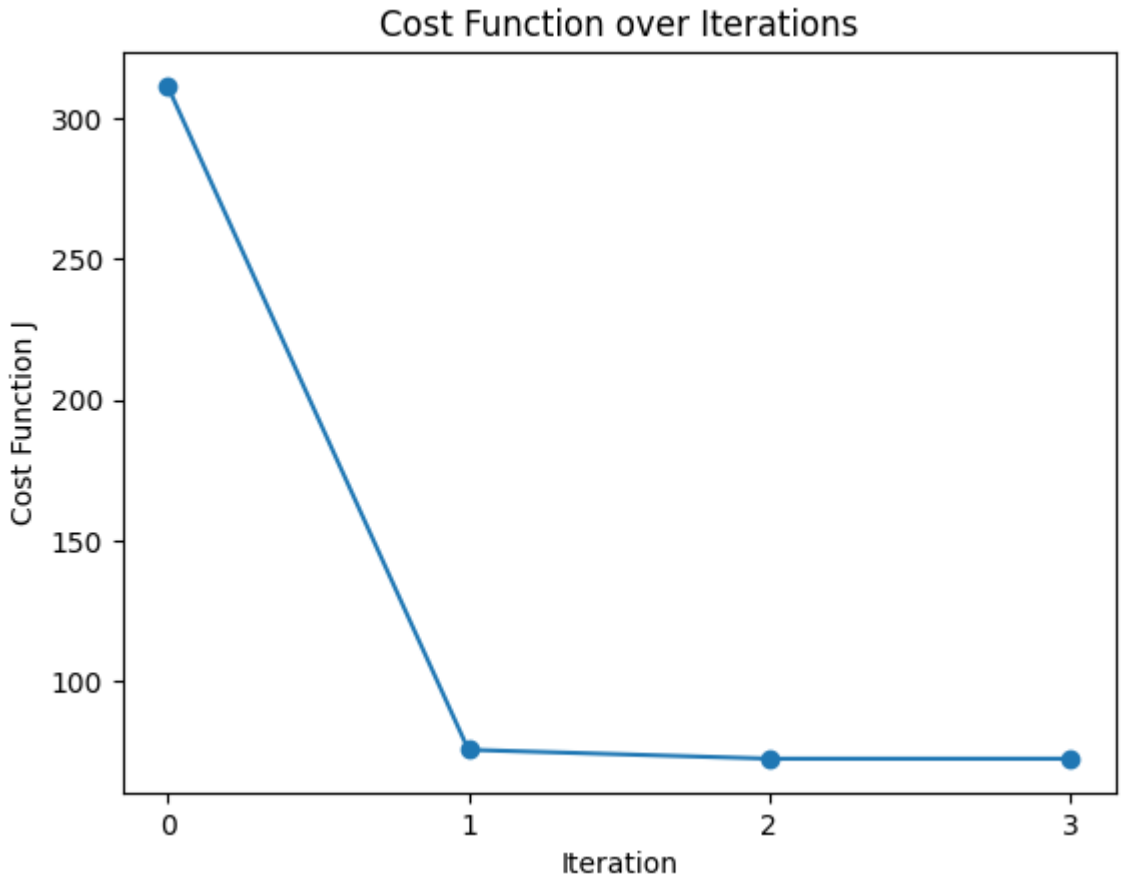
Clustering

K-Means Algorithm — Optimization Objective

- The optimization objective of k-means algorithm is to minimize the cost function, which can be defined as the squared distances between each data point and its dominating centroid.

$$J = \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - c_k\|^2$$

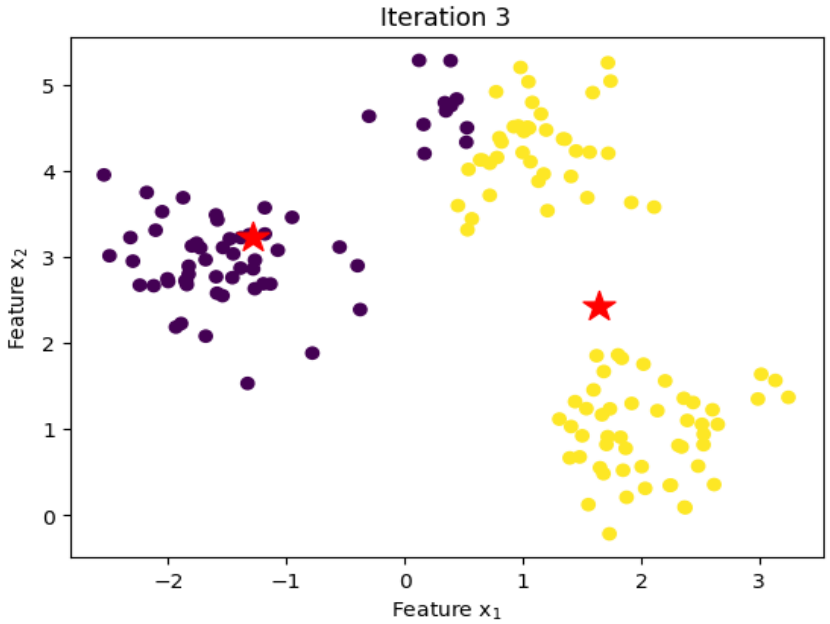
- The average of this cost function over the number of the data points is called distortion
- Each step of the k-means algorithm refines the choices of centroids to reduce the cost function and the distortion
- Changing the cost function insignificantly is an indication that the cluster centers do not change significantly
- Regarding the discussed example, the cost function changes insignificantly after iteration number two as the cluster centers do not change significantly as well.

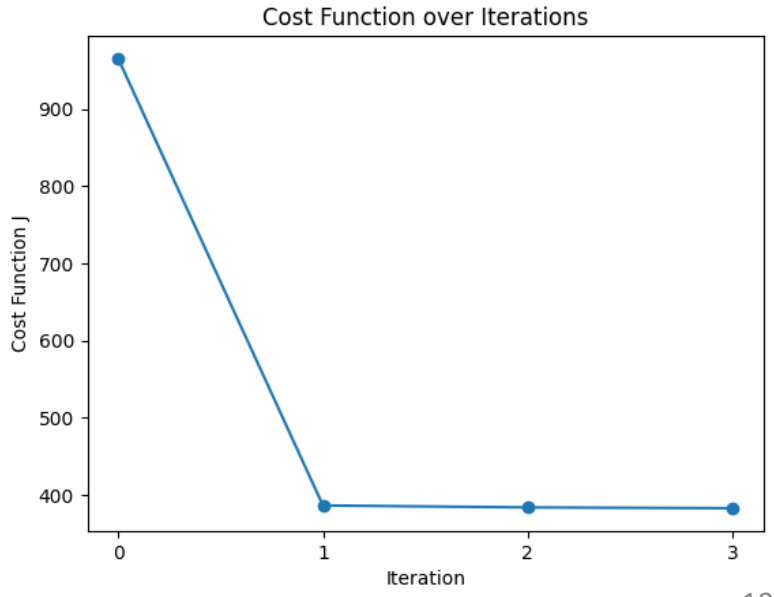


Clustering

K-Means Algorithm — Cost Function

+ Algorithm requires a prior determination of the number of clusters K, but it is often not known
+ Example was performed intuitively with K = 3 clusters, different **result for K = 2 clusters**

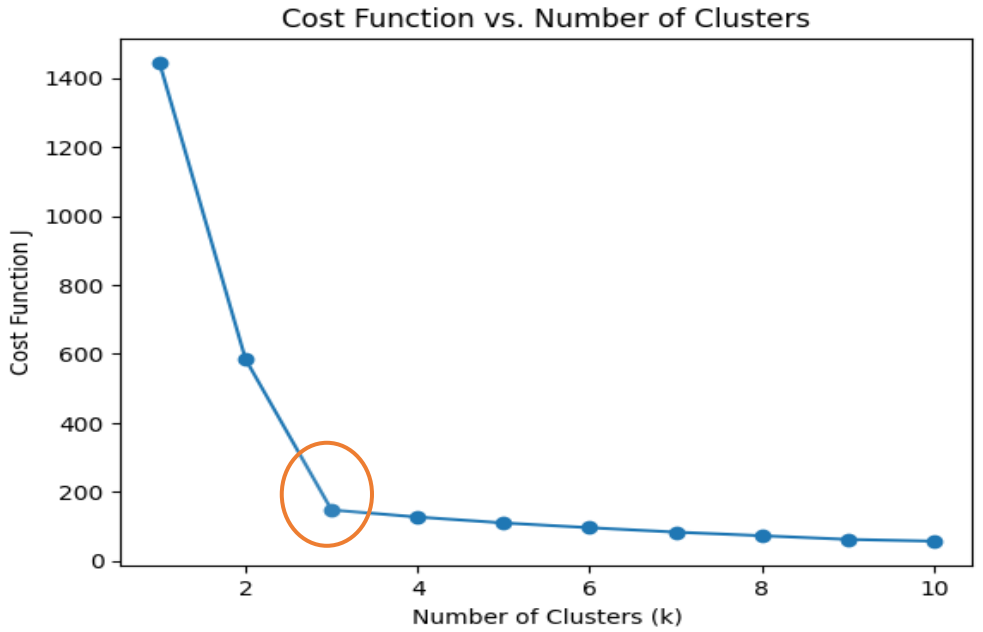




Clustering

K-Means Algorithm — Cost Function

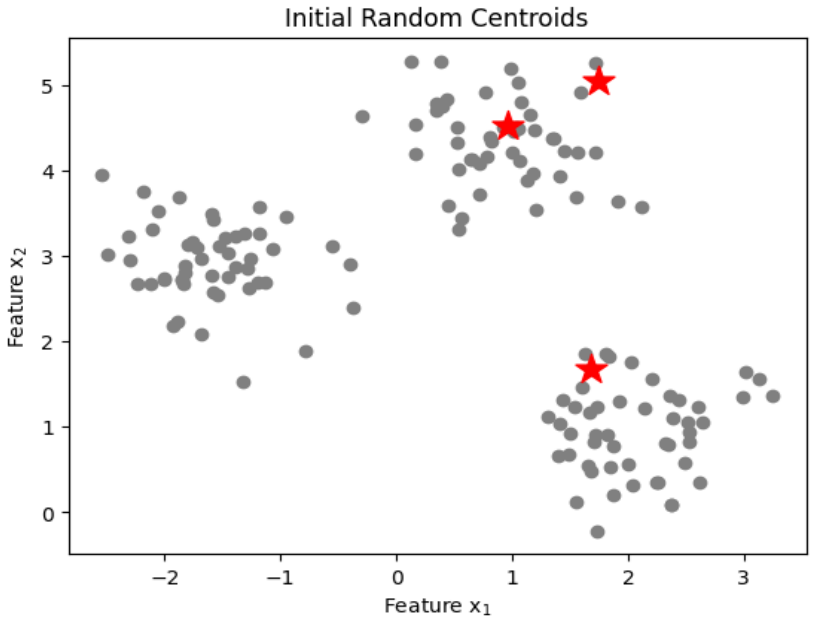
+ To identify a suitable number of clusters, the cost function is determined as a function of the number of clusters K
  - The graph looks like an elbow
  - When k=1 the cost function has the highest value but with increasing k value its value starts to decrease
  - We choose that value of k from where the graph starts to look like a straight line
  - K=3 represents the suitable number of clusters in this case

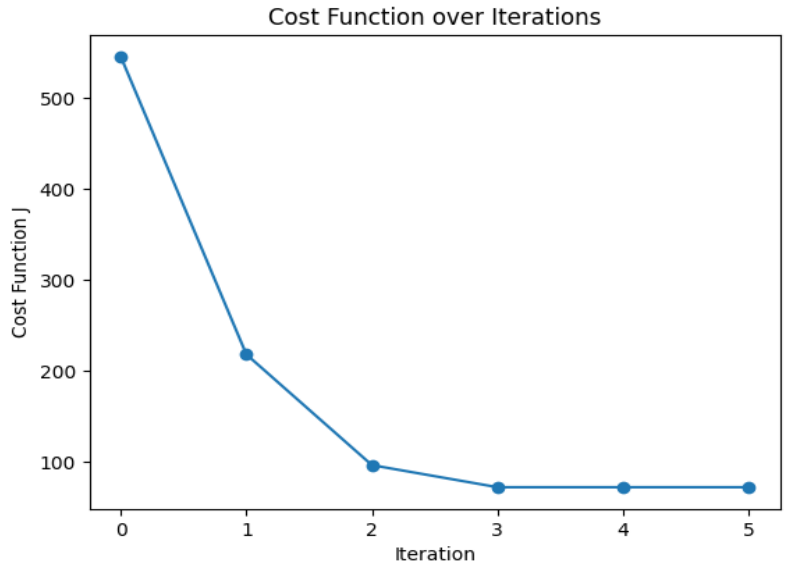


Clustering

K-Means Algorithm

+ Multiple runs of the algorithm with different initialization values can change the results, because the initial values of the centroids are chosen randomly all at once without any constrains
+ If they are close to each other, that can slow down convergence

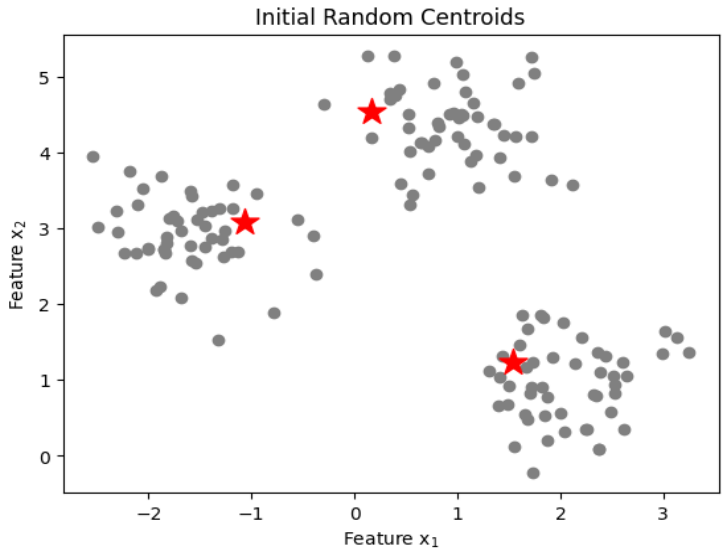


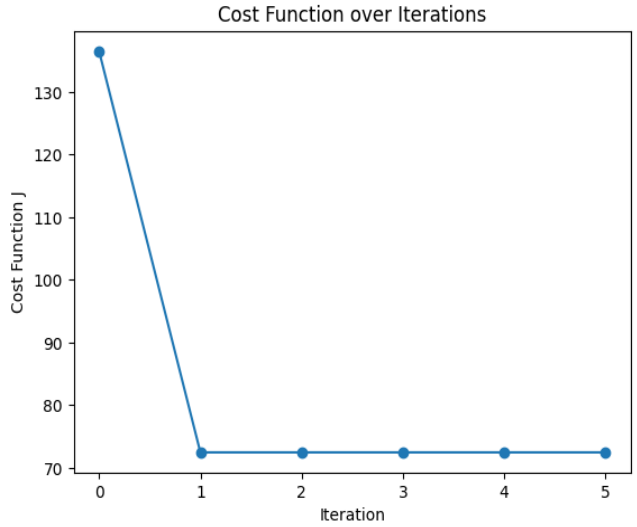


Clustering

K-Means Algorithm — K-Means++

+ More robust initialization with the K-Means++ algorithm
  - one value is randomly selected from the sample as center c_1
  - remaining K - 1 initialization values are also randomly determined, probability of selection increases with increasing distance from the already selected centers
  - Initialization values are determined randomly, but are far apart from each other
  - That helps to speed up convergence and improve clustering quality

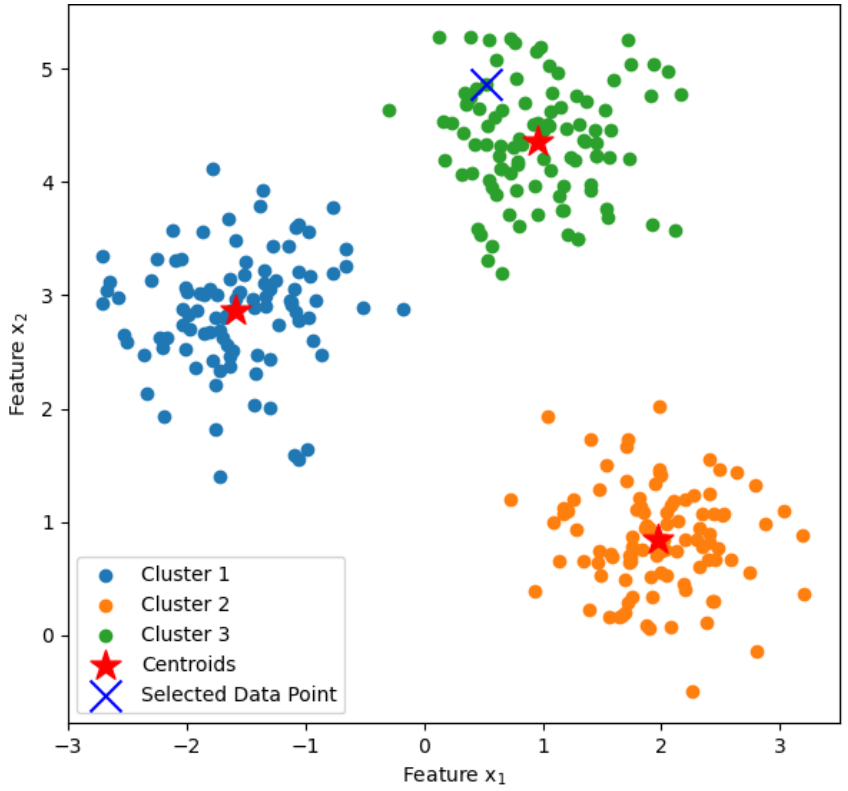




Clustering

K-Means Algorithm – Evaluation of Clustering Results

- In practice, the ideal number of clusters is often unknown
- The ideal value ensures that the objects in a cluster are
  - arranged close to each other (Cohesion: how similar that sample is to its own cluster) and
  - the different clusters are well distinguishable from each other (separation: How far apart clusters are from each other)
- To determine the best K, the silhouette coefficient can be used.
- Two parameters for each data point are calculated:
  - Intra-cluster distance (a(x_i)): The average distance between a data point x_i and all other points in the same cluster
  - Nearest-cluster distance (b(x_i)): The average distance between a data point x_i and all points in the nearest cluster that the sample is not a part of



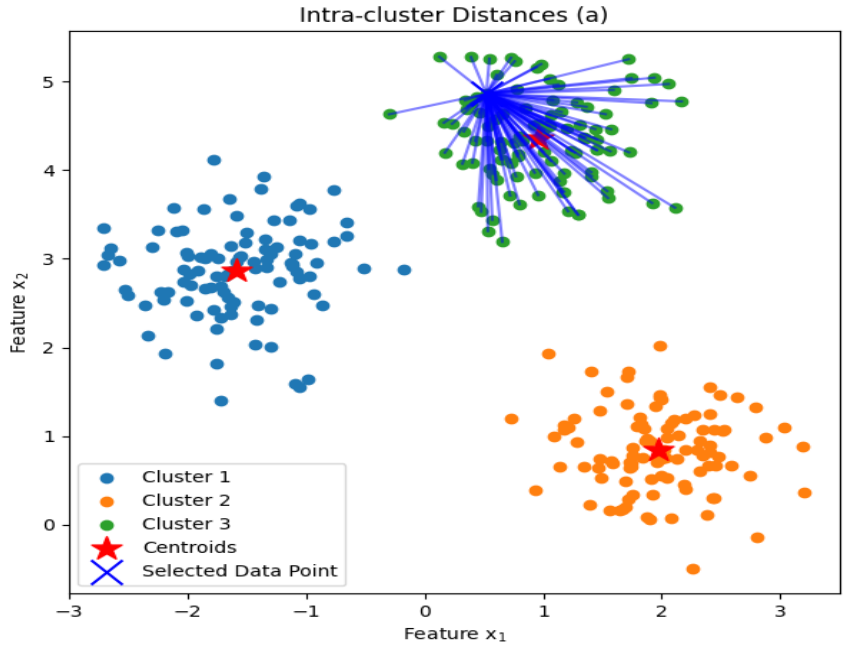
Clustering

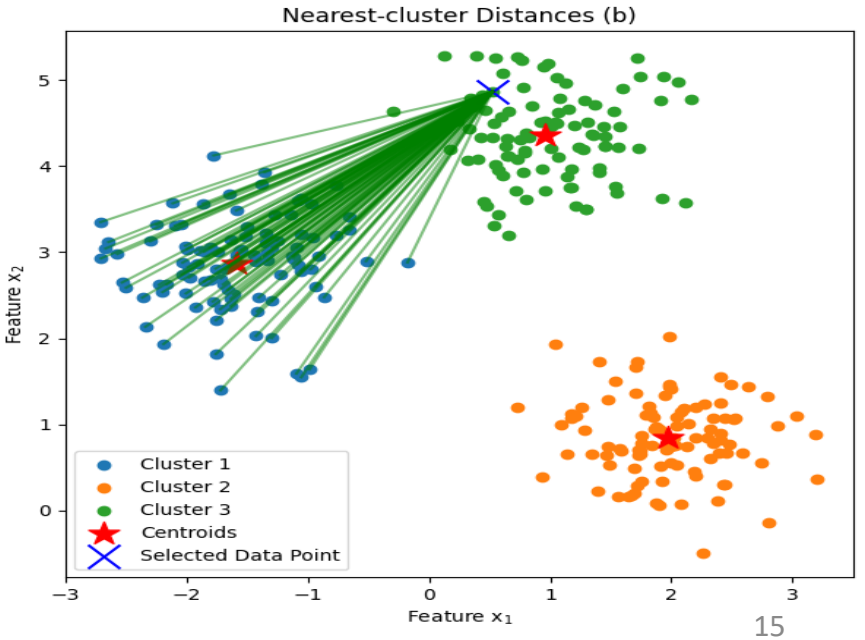
K-Means Algorithm — Evaluation of Clustering Results

+ For each data point x_i, the Silhouette Coefficient is computed as:

$$S(x_i) = \frac{\mathrm{b}(x_i) - \mathrm{a}(x_i)}{\max(\mathrm{a}(x_i), \mathrm{b}(x_i))}$$

+ For the illustrated data point:
  - Intra-cluster distance (a) = 0.88
  - Nearest-cluster distance (b) = 2.95
  - Silhouette Coefficient = 0.70
+ The Silhouette Coefficient ranges from -1 to 1
  - A value close to 1 indicates that the sample is well clustered,
  - a value close to 0 indicates that the sample is on or very close to the decision boundary between two neighboring clusters,
  - and a value close to -1 indicates that the sample might have been assigned to the wrong cluster.



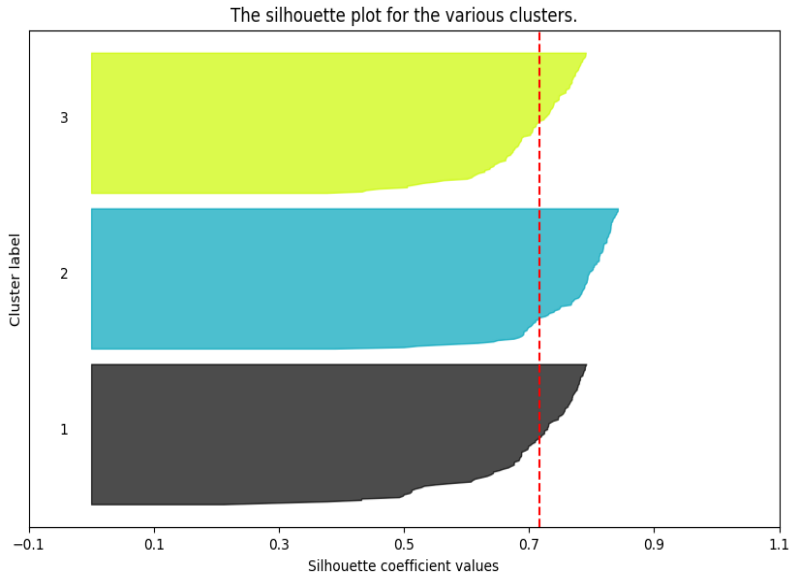


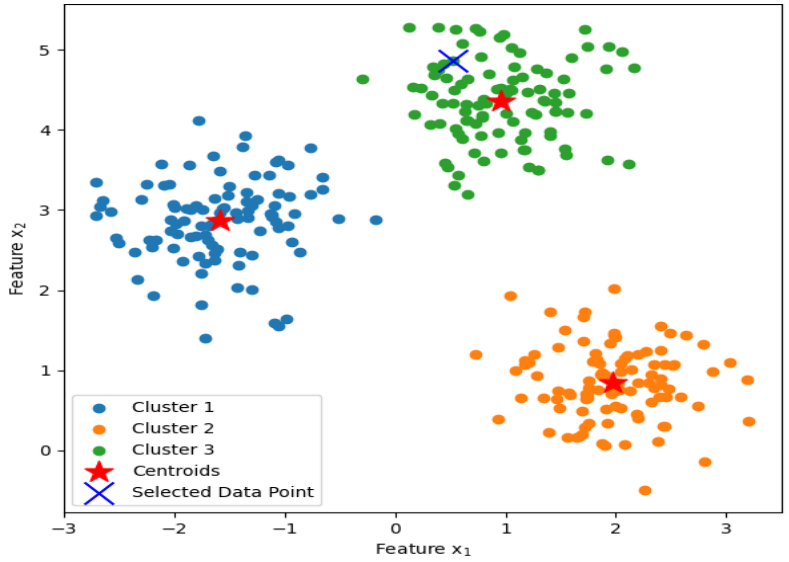
Clustering

K-Means Algorithm — Evaluation of Clustering Results

- The overall silhouette score for the dataset is the mean Silhouette Coefficient for all points in the dataset
- Overall Silhouette Score: 0.72
- From the thickness of the silhouette plot the cluster size can be visualized.

Overall Silhouette Score	Interpretation
0.71 – 1.00	a well-defined, strong clustering structure
0.51 – 0.70	a reasonable clustering structure with good separation
0.26 – 0.50	a weak structure
< 0.25	little to no underlying clustering structure

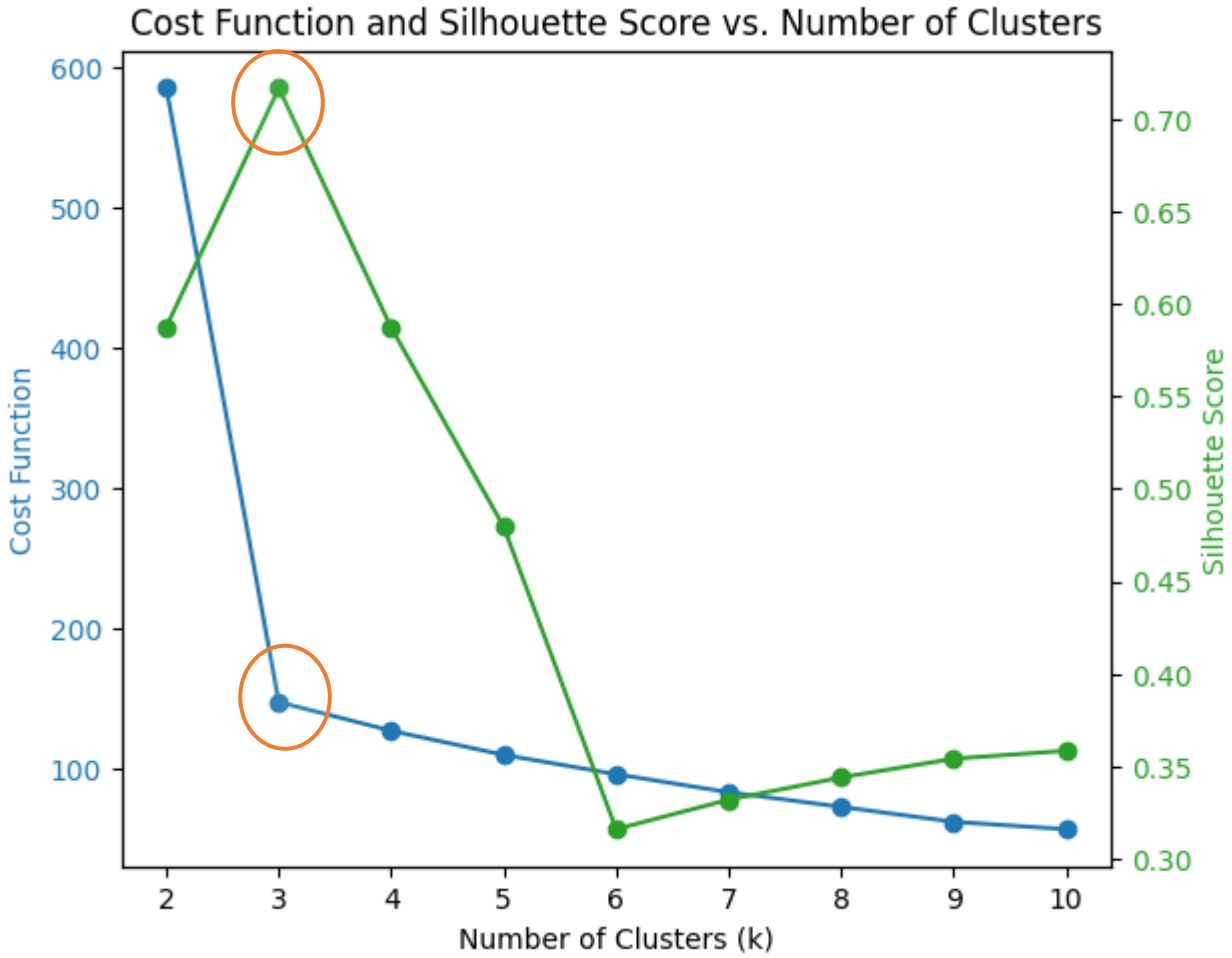




Clustering

K-Means Algorithm — Evaluation of Clustering Results

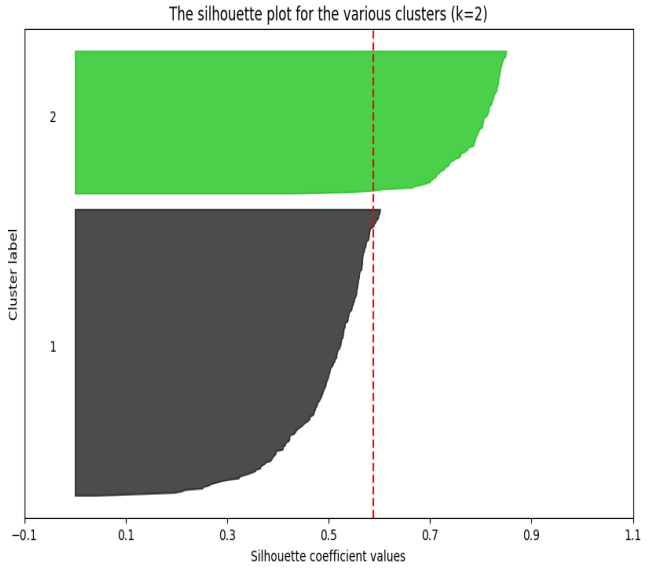
- In this example the silhouette analysis is used to choose an optimal value for the number of clusters
- For k = 3, the silhouette score has the largest value and equals to 0.72. Therefore, k=3 is selected as the best value
- K=3 is selected also according to the analysis of the cost function with respect to the number of clusters

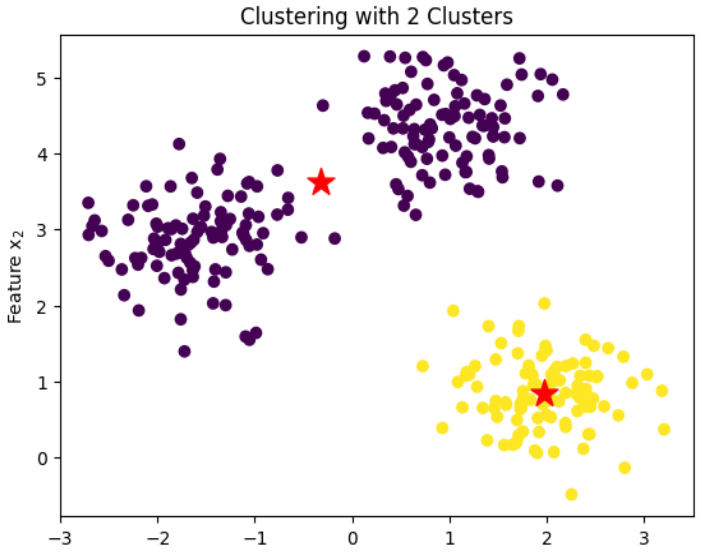


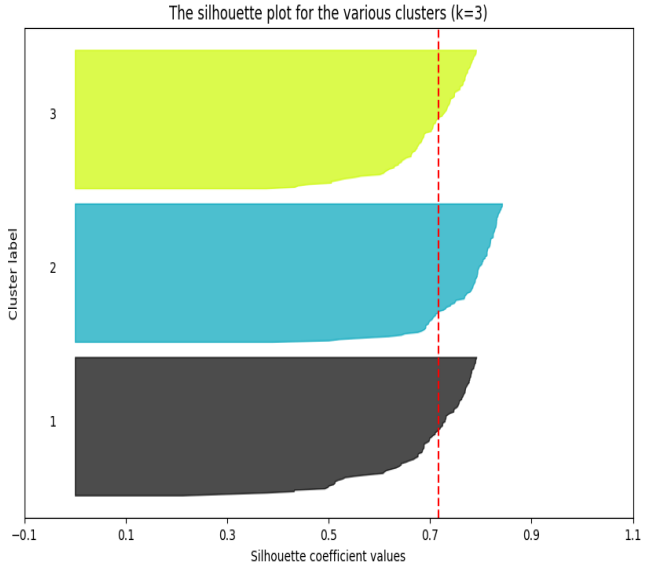
Clustering

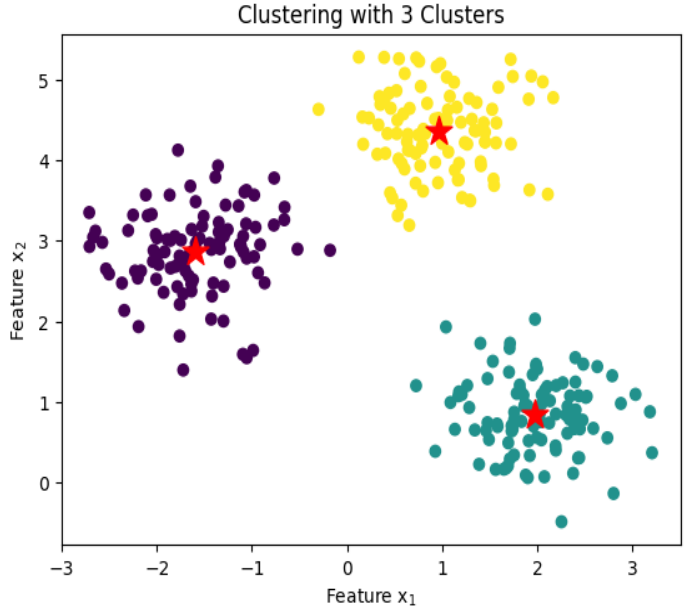
K-Means Algorithm — Evaluation of Clustering Results

- From the thickness of the silhouette plot the cluster size can be visualized
- The silhouette plot for cluster 1 when k is equal to 2, is bigger in size. It groups 2 sub clusters into one big cluster.
- However when the K is equal to 3, 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
- Wide and High Silhouettes: Indicate well-defined clusters.
- Narrow and Low Silhouettes: indicate that clusters are not well-separated and may overlap.





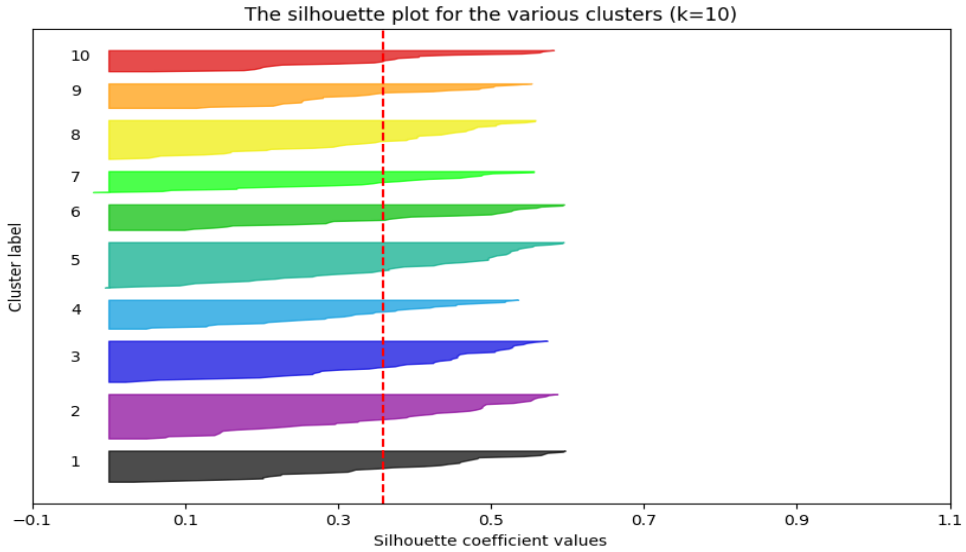


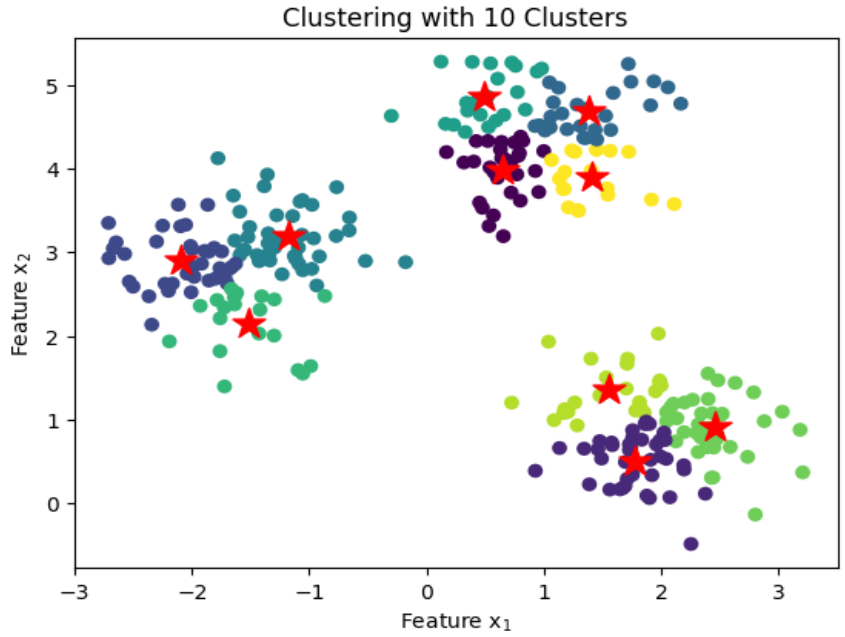


Clustering

K-Means Algorithm — Evaluation of Clustering Results

+ Wide and High Silhouettes: Indicate well-defined clusters
+ Narrow and Low Silhouettes: Suggest that clusters are not well-separated and may overlap

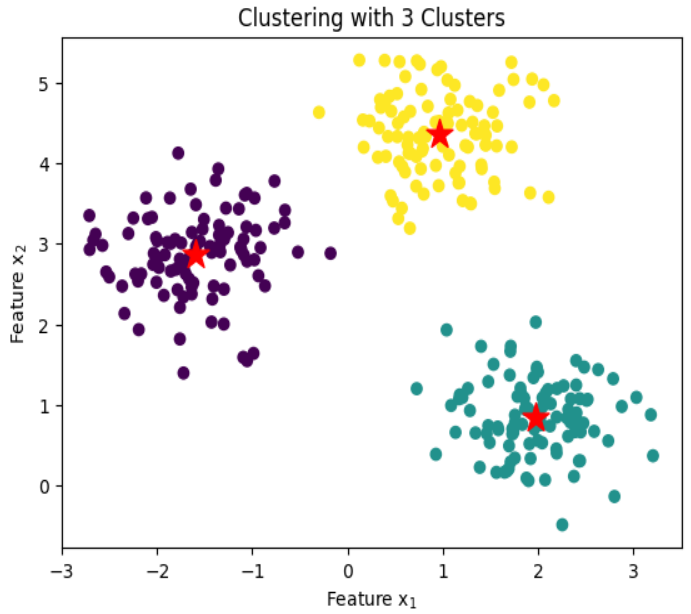




Clustering

K-Means Algorithm — New Data Points

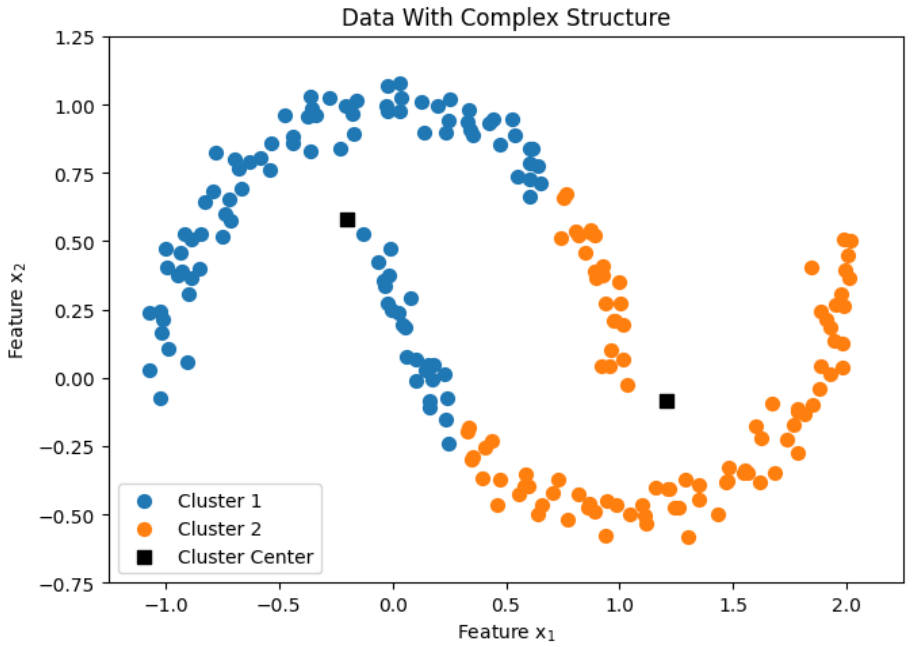
- If there is a new data point that is to be assigned to the clusters, the complete algorithm must basically be run to redetermine the clusters based on the entire data points.
- If the previous sample size N is large, the new data point will have little effect on the clusters and the associated cluster centers.
- In that case, the new data point can be provisionally assigned to a cluster by distance from the existing cluster centers before a time-consuming run of the K-Means algorithm is performed.



Clustering

K-Means Algorithm

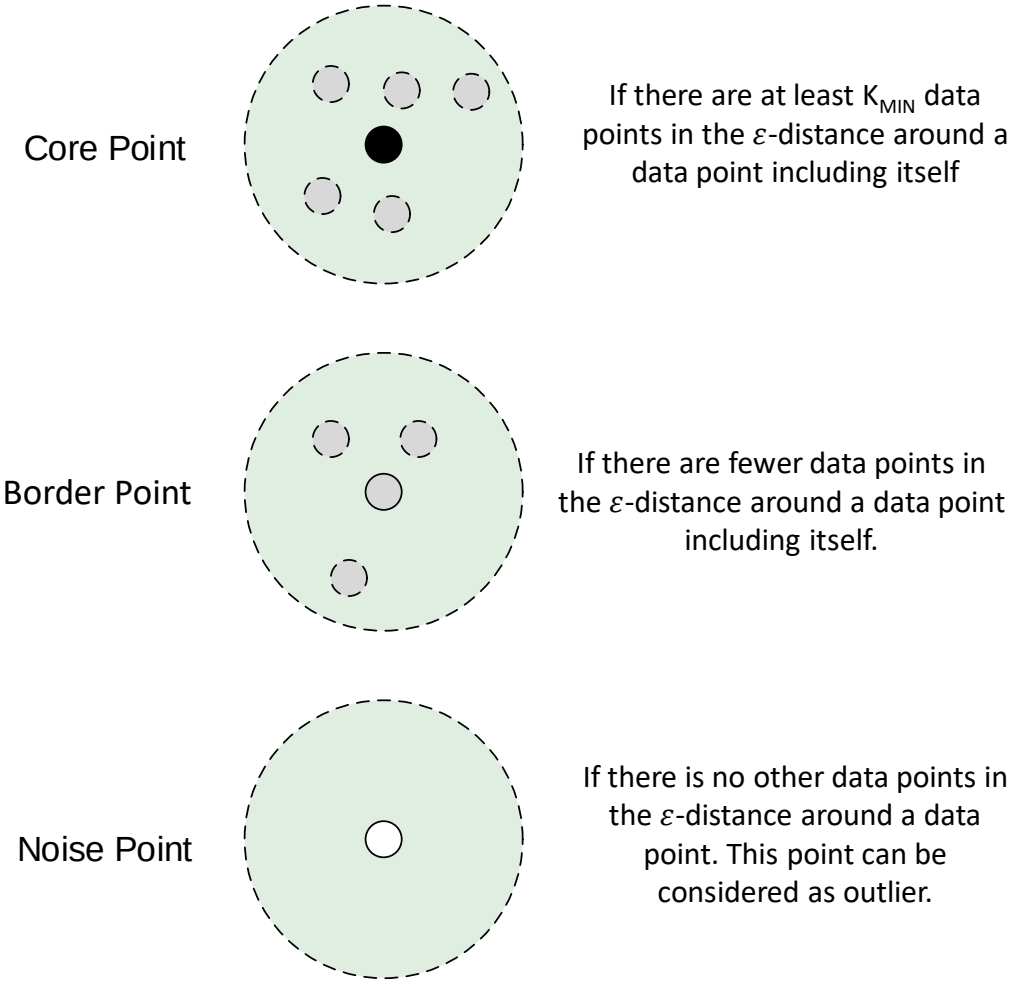
- The cost function uses squared Euclidean distance measure which affects the formation of clusters.
- For dataset with different spatial extent, using Euclidean distance causes some sample values to be assigned differently than expected.
- Standardization of data can reduce but not avoid this effect.
- Principal component analysis of sample values beneficial
- Two-moon samples may not be recognized as contiguous clusters by Euclidean distance assessment
- Other clustering methods required, e.g. DBSCAN



Clustering

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

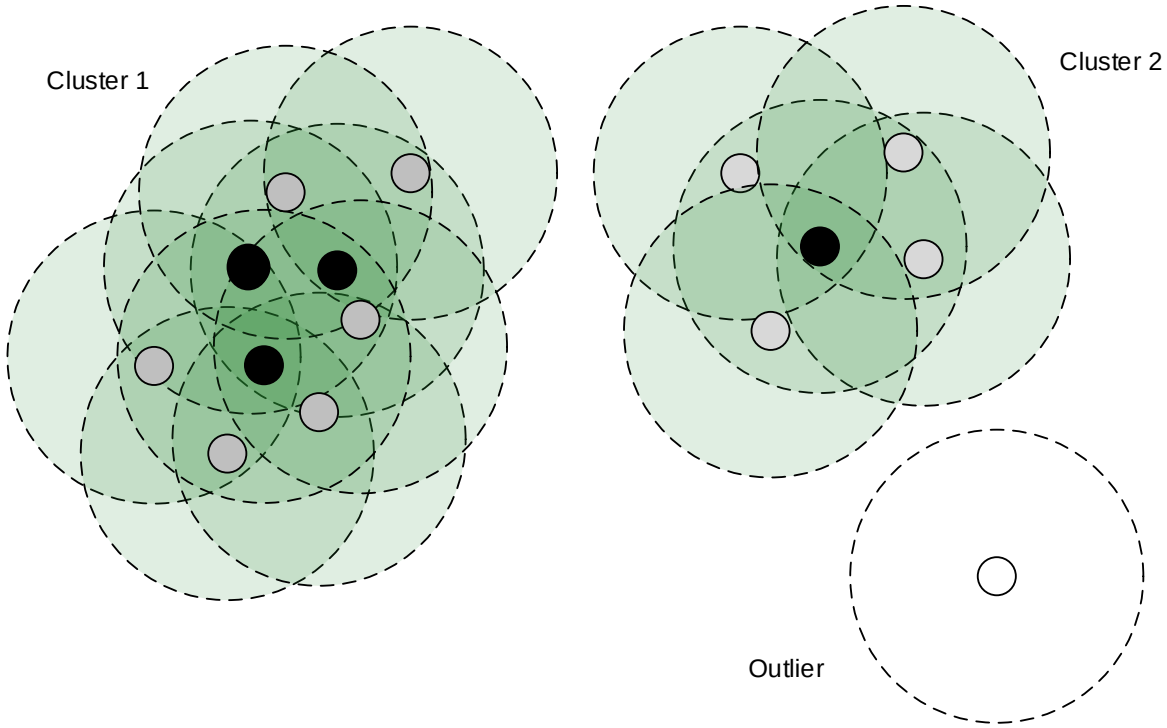
+ DBSCAN method is a density-based cluster method
+ Advantage: number of clusters does not have to be known in advance, algorithm also recognizes complex cluster structures
+ DBSCAN method can be used to identify outliers
+ Algorithm checks whether and which data points are in the distance of other data points
  - Evaluation of an ε-distance around each data point
  - Classification into core, border and noise points depending on the number of neighboring data points



Clustering

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

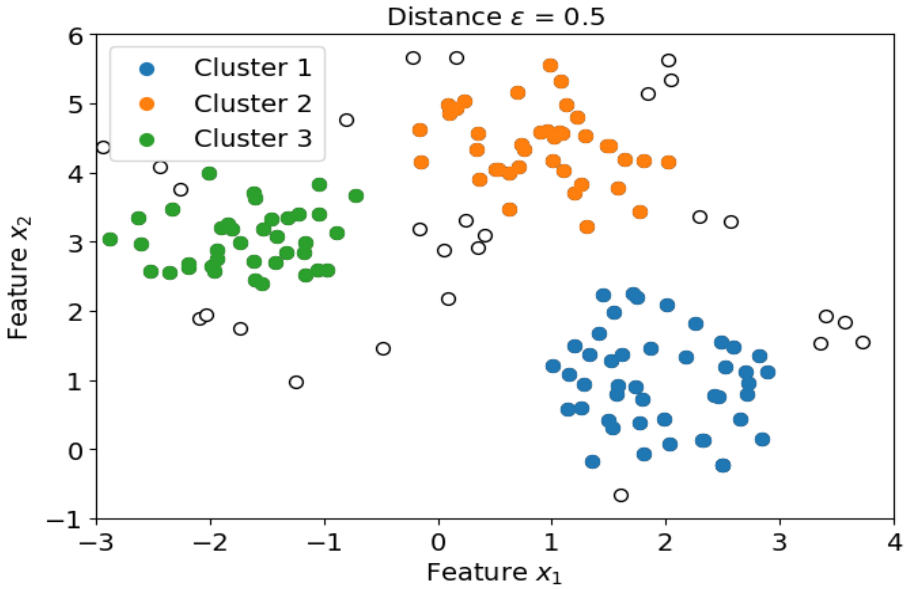
+ Algorithm has two hyperparameters:
  - Dimension of ε-distance
  - Required minimum number of K_{MIN} for a core point
+ Classification of points:
  - Search for core point and label all other points within the ε-distance as border points
  - Check for each of these points if it is also a core point
  - Repeat until all points are characterized
+ In this example:
  + K_{MIN} is equal to five
  + Two clusters 1 and 2 are formed
  + One point has no other data points in its ε-distance, it is classified as an outlier
+ Try your self: https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/

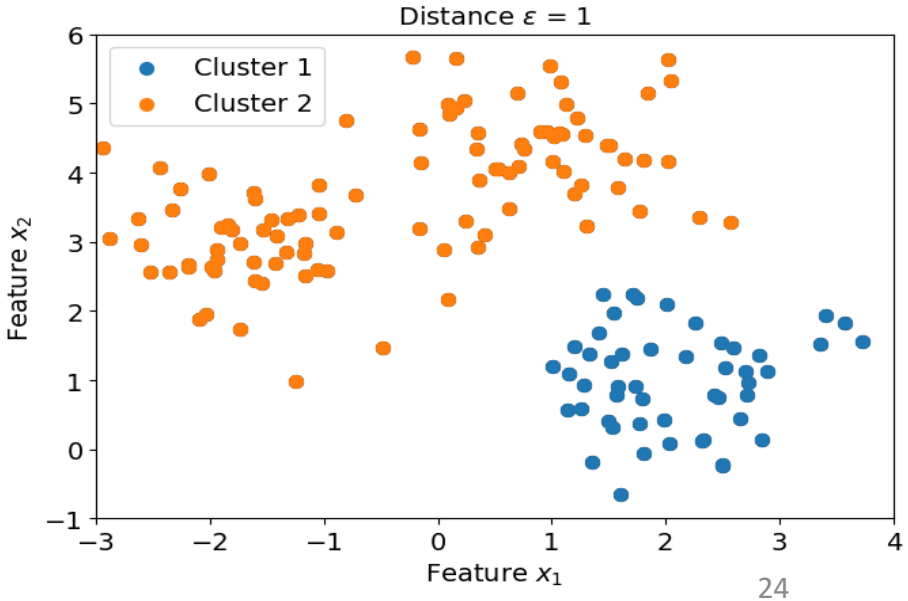


Clustering

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

+ Size of the ε-distance is essential in the procedure for the formation of clusters
+ Demonstration on a small dataset with different parameterization
  - ε = 0.5:
    three clusters are formed, some data points are too far away from the clusters and are classified as noise points
  - ε = 1:
    both upper clusters are connected to one big cluster, also data points lying far away are assigned to one cluster

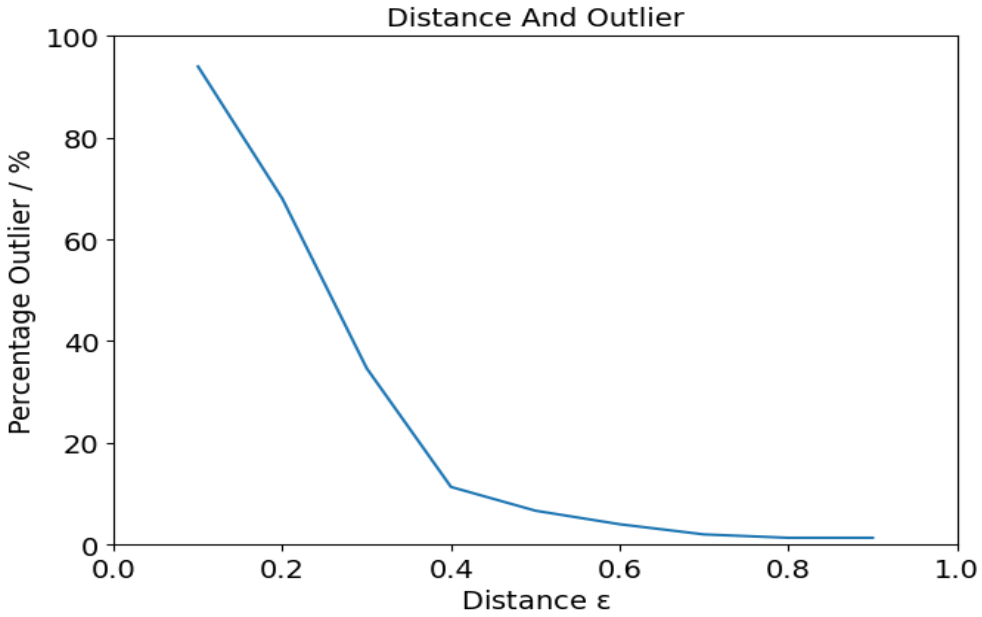


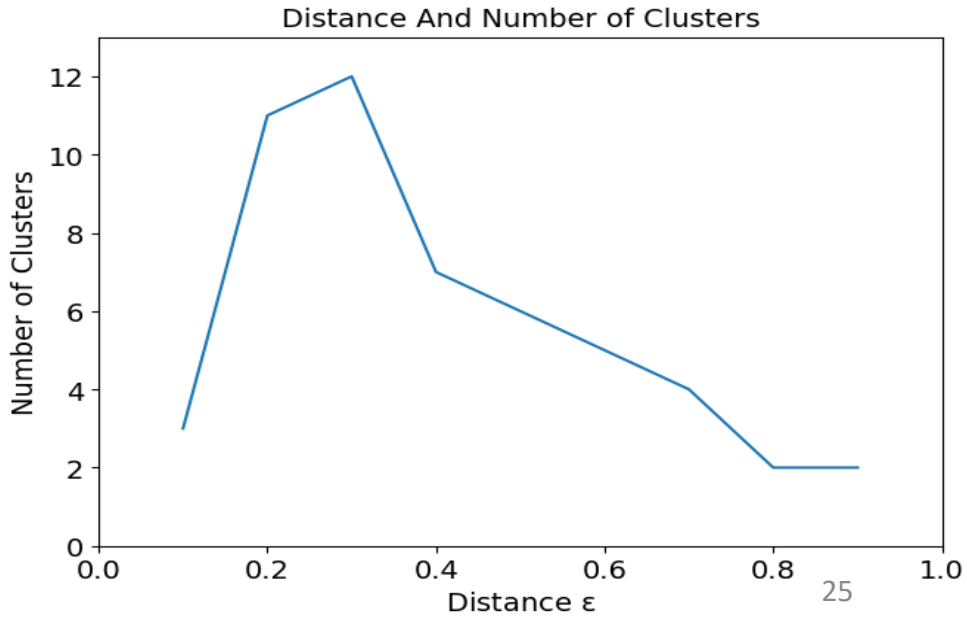


Clustering

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

- As the ε -distance increases, the proportion of outliers decreases because the catch range of the distances increases
- Number of clusters initially increases as the ε -distance increases because the critical size of KMIN points falls more often in an distance
- If the ε -distance continues to increase, several small clusters combine to form large clusters and the number of clusters decreases

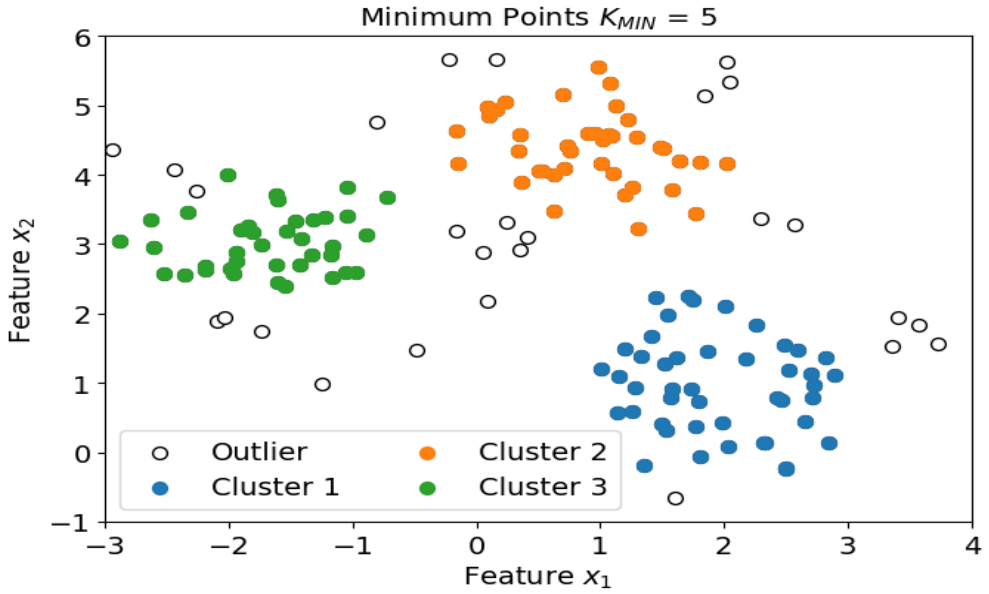


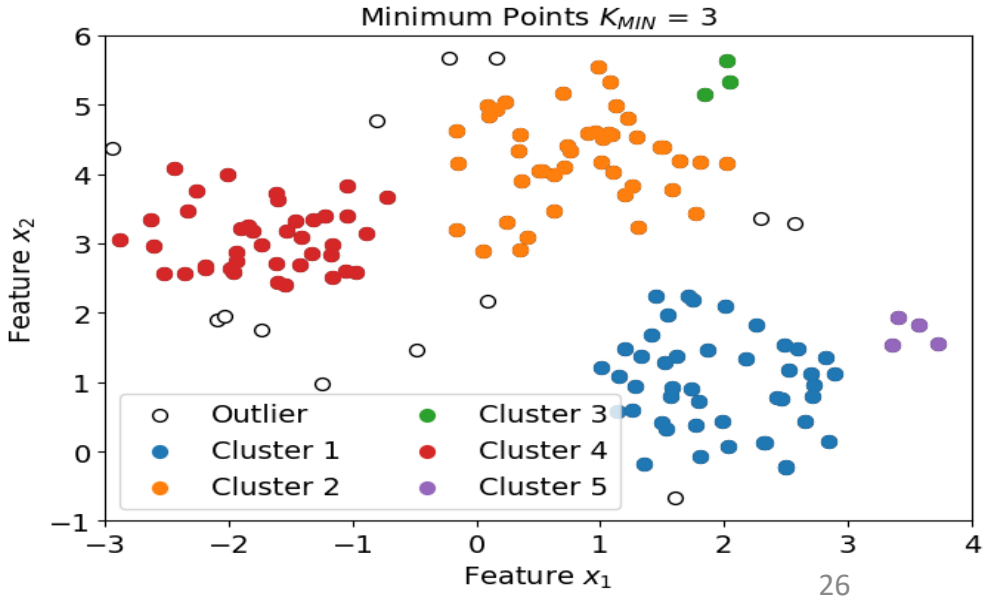


Clustering

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

+ Cluster result depends on the minimum number of neighbors for a core point K_{MIN} for identical ε -distance.
+ If the number of required neighbors is reduced, groups of outliers form new clusters, so the number of clusters is increased and the number of outliers is reduced
  - K_{MIN} = 5: this results in 3 clusters and 28 outliers
  - K_{MIN} = 3: it results in 5 clusters and 13 outliers
+ In the following, parameter study of the number of K_{MIN} required neighbors for an own cluster

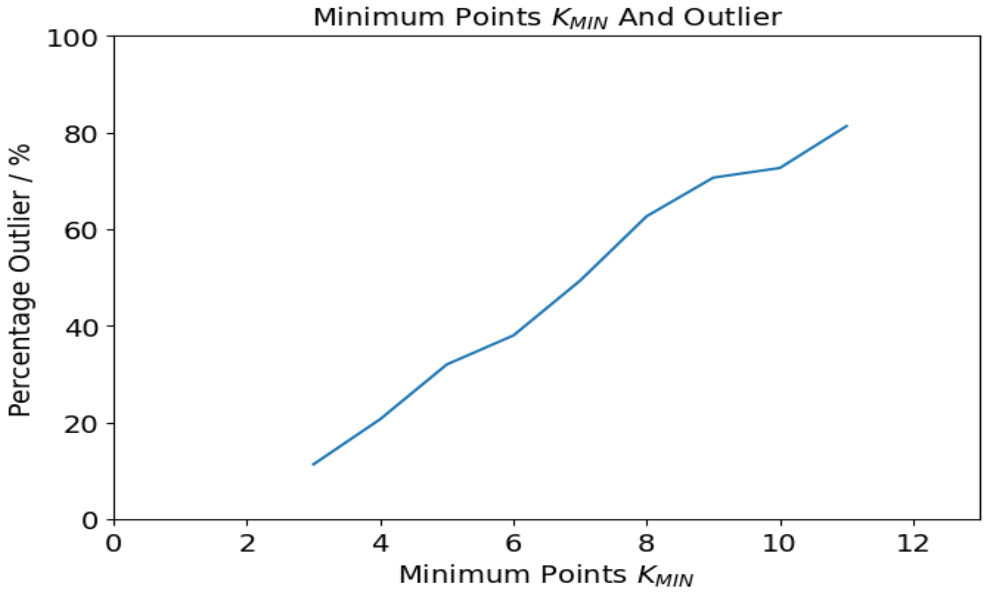


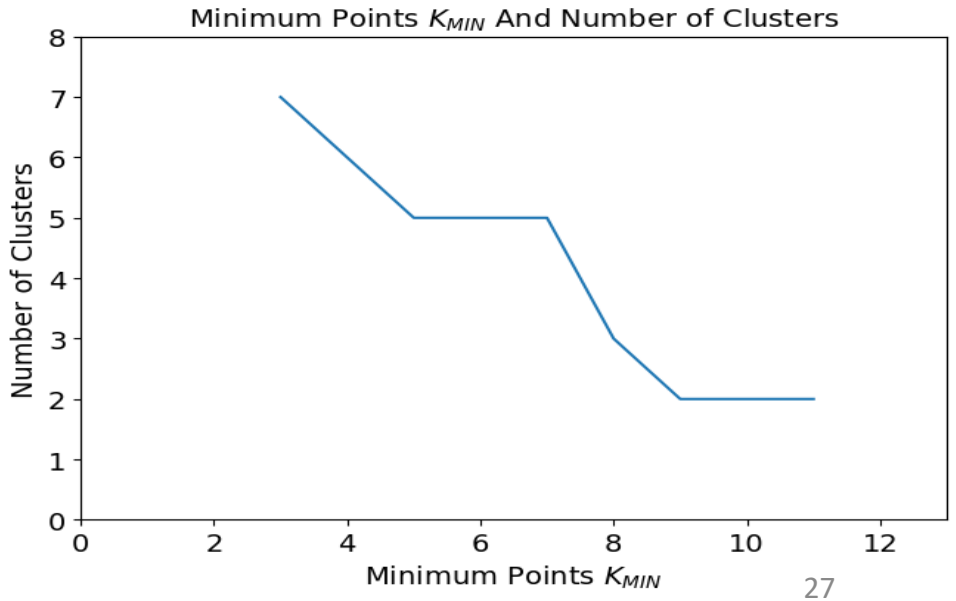


Clustering

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

- The more neighbors are needed for a core point, the more likely no cluster is reached
- Proportion of outliers therefore increases with increasing K_{MIN}
- Size of clusters increases with increasing K_{MIN}, therefore number of clusters decreases with increasing K_{MIN}
- Parameter studies show that the two hyperparameters can be used to,
  - the size and number of clusters and
  - adjust the proportion of outliers
- Optimization must be specific for each application

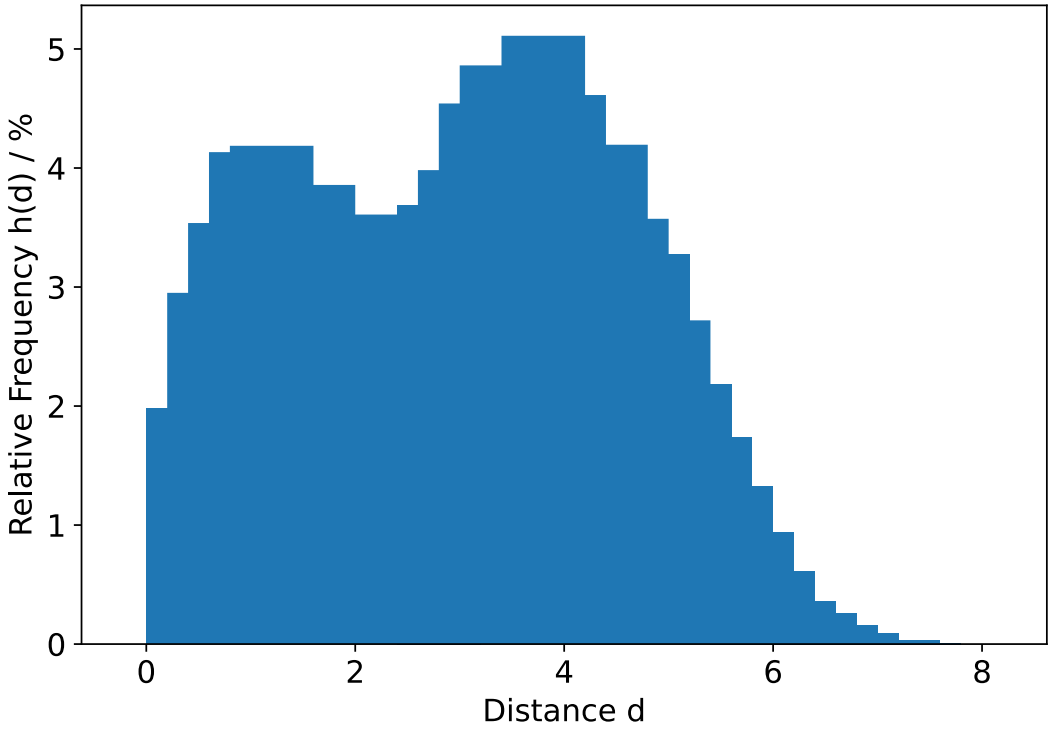




Clustering

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

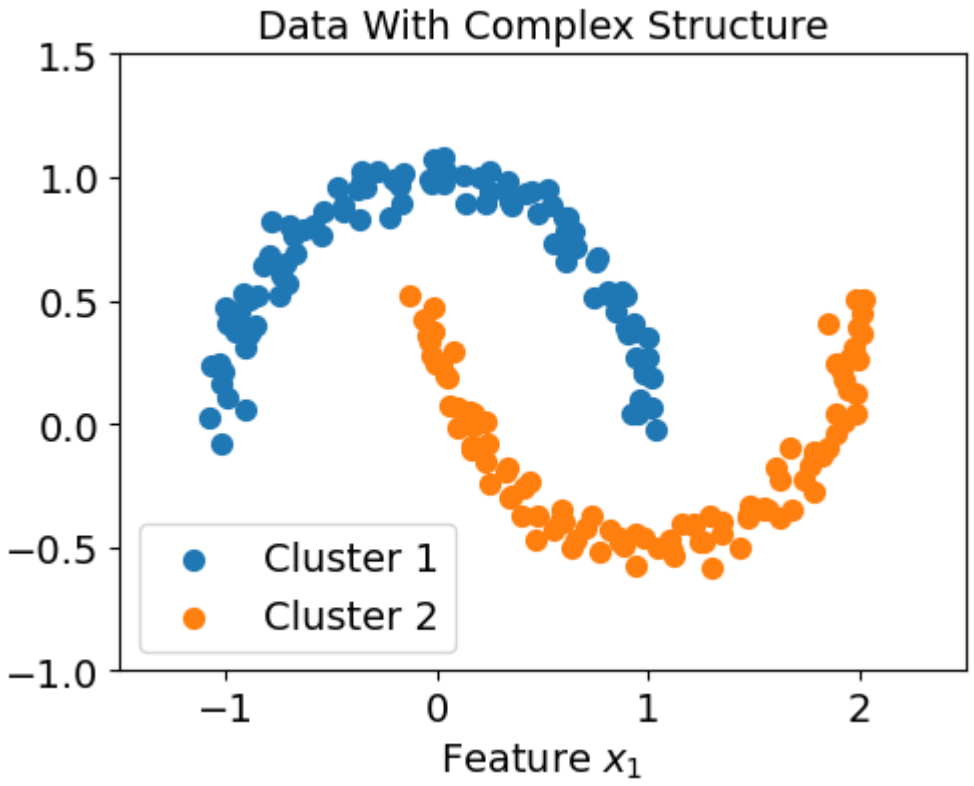
+ Estimation of reasonable values for the parameter based on the distance of all points to each other
+ Representation as histogram of the distances of data points to each other
+ Essential for the determination of the ε -distance is the minimum distance with significant proportion
  - 2 % of the parts have a distance of 0.2 or less in this example
  - Parameter selected slightly above this range
+ Further optimization of the hyperparameter is done by parameter studies



Clustering

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

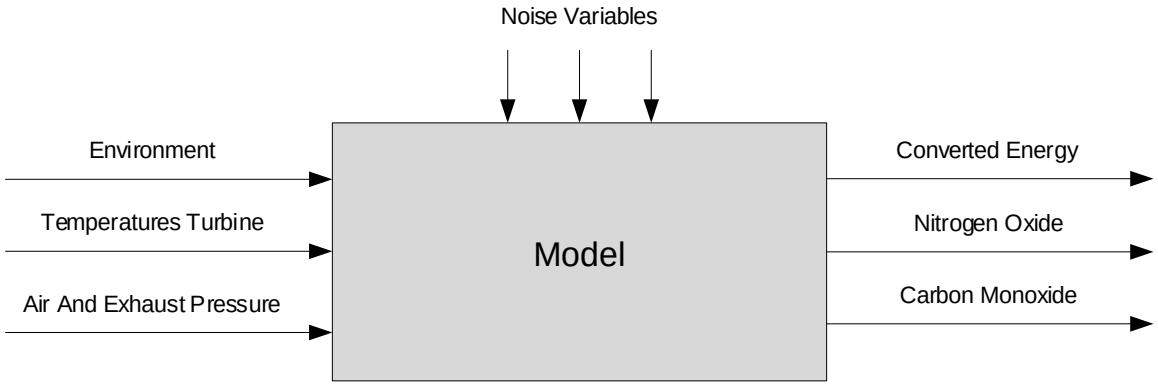
+ Two-moon data points can be recognized as contiguous clusters by DBSCAN



Clustering

DBSCAN - Example With Outlier Detection

+ Outliers indicate a failure of the underlying process.
+ However, misbehavior of complex processes can often only be detected by specific combinations of different variables.
+ Even if each variable is within the typical scatter range, the combination of these variables can indicate process misbehavior.
+ Detection of the misbehavior of a gas turbine using the DBSCAN cluster method.
+ Dataset from the Faculty of Engineering, Namik Kemal University.



Clustering

DBSCAN - Example With Outlier Detection

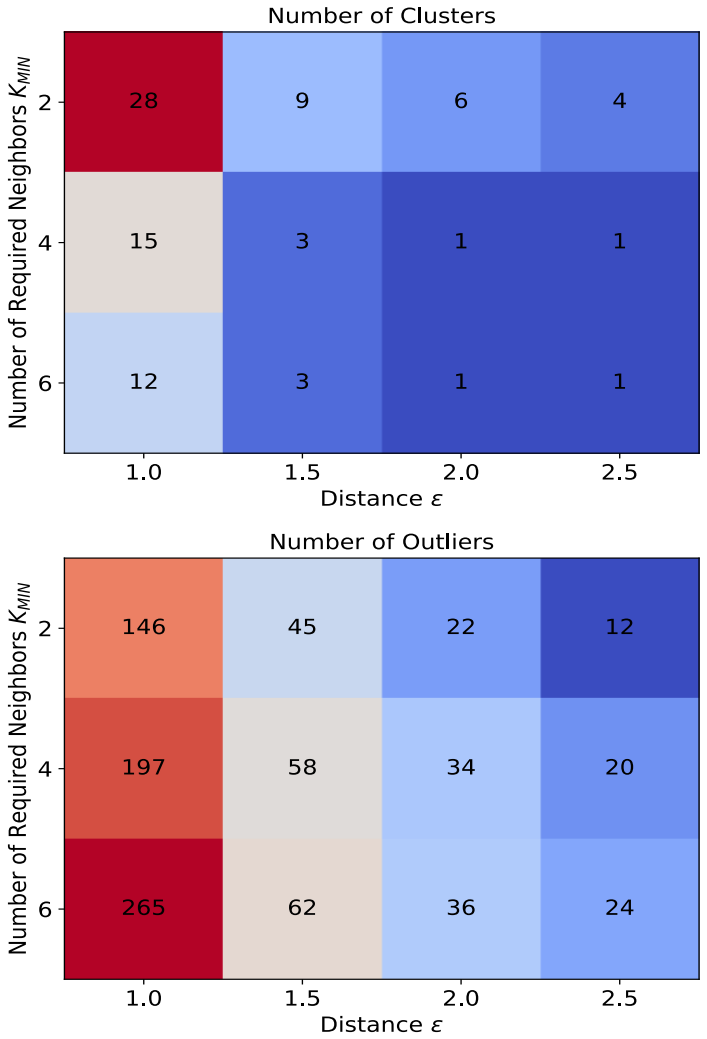
- Emissions of a gas turbine as a function of power output and ambient conditions.
- The turbine is controlled by the requirement of a converted energy per acquisition interval.
- Output signals are the converted energy per unit time as well as nitrogen oxide and carbon monoxide emissions.
- Temperature, pressure and humidity are recorded at different locations as environmental and process variables.
- Data have been standardized because of the different magnitudes.

Name	Unit	Minimum	Maximum	Average
Ambient Temperature AT	°C	- 6.23	37.10	17.71
Ambient Pressure AP	mbar	985.85	1036.56	1013.07
Ambient Humidity AH	%	24.08	100.20	77.87
Air Filter Pressure Drop AFDP	mbar	2.09	7.61	3.93
Exhaust Gas Back Pressure GTEP	mbar	17.70	40.72	25.56
Turbine Inlet Temperature TIT	°C	1000.85	1100.89	1081.43
Turbine Outlet Temperature TAT	°C	511.04	550.61	546.16
Compressure Pressure CDP	mbar	9.85	15.16	12.06
Converted Energy TEY	MWh	100.02	179.50	133.51
Carbon Monoxide CO	mg/m^3	0.00	44.10	2.37
Nitrogen Oxides NOx	mg/m^3	25.90	119.91	65.29

Clustering

DBSCAN - Example With Outlier Detection

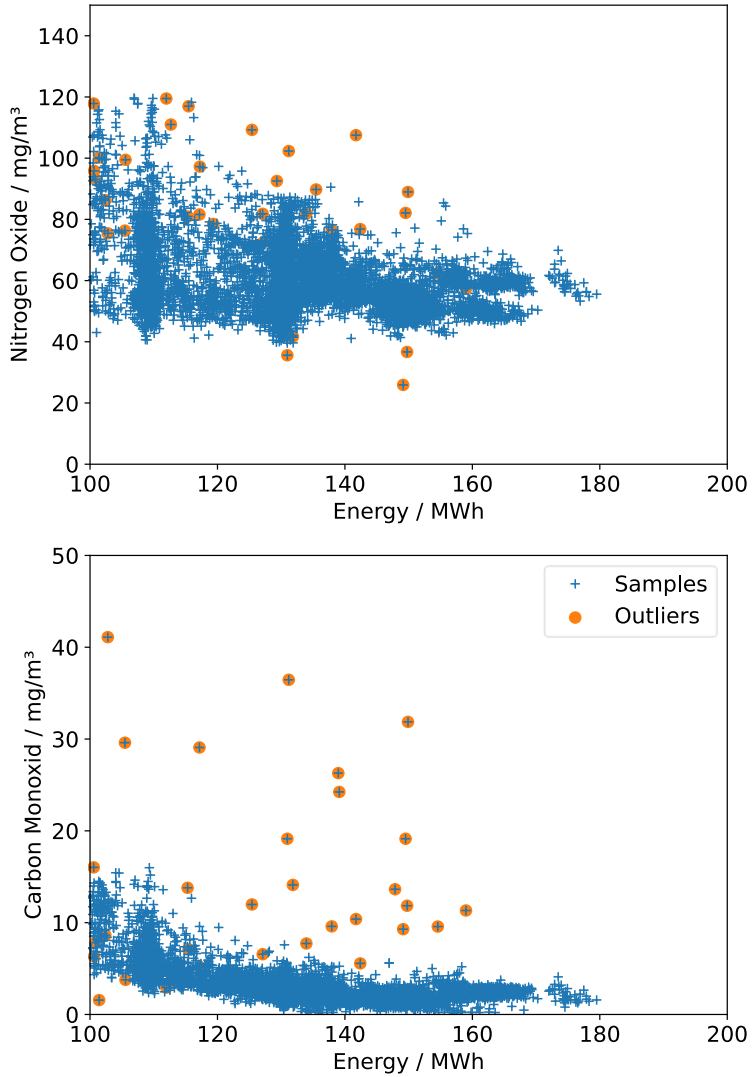
- Cluster analysis to separate typical operating points from non-typical operating points.
- DBSCAN method is to be parameterized to produce a single cluster of typical operating points, the noise points not belonging to this cluster then represent outliers.
- ε-distance and the number of required neighbors for core points K_{MIN} are varied to obtain the desired cluster result.
- For an environment with ε = 2 and ε = 2.5 as well as K_{MIN} = 4 or K_{MIN} = 6 a cluster number one is obtained.
- Depending on the parameterization between 20 and 36 outliers.



Clustering

DBSCAN - Example With Outlier Detection

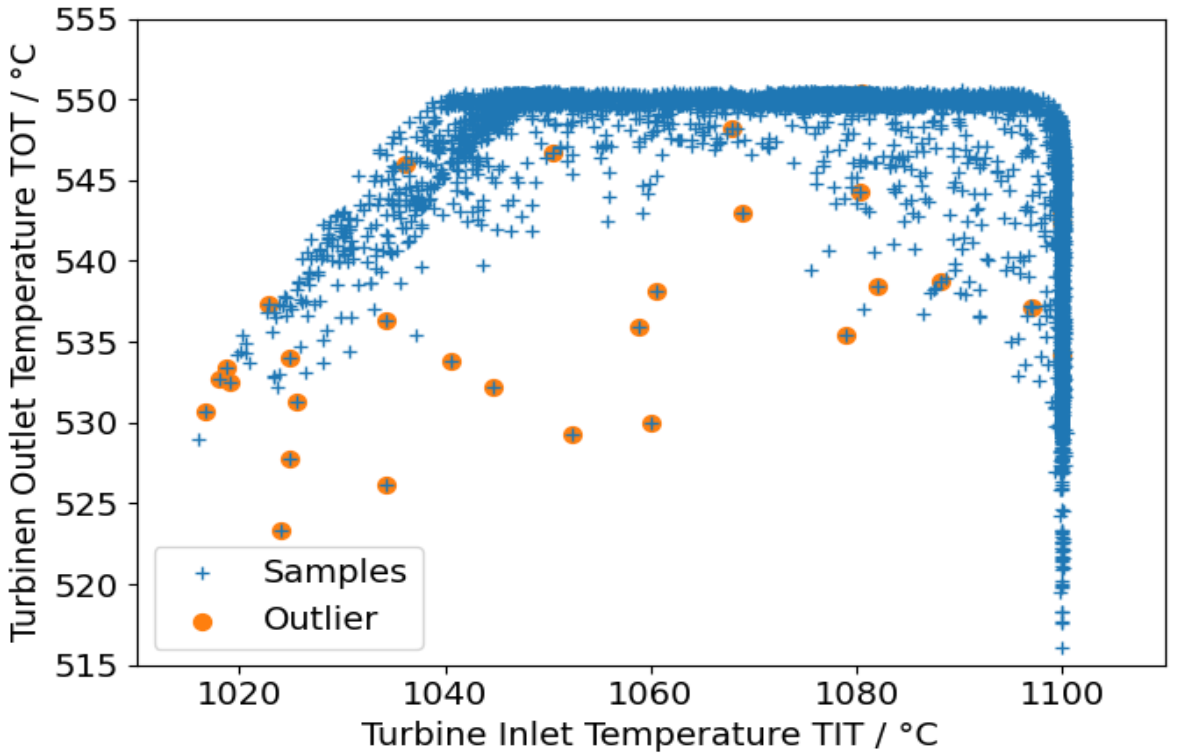
+ Representation of the operating points for the parameter combination and $\varepsilon = 2$ and KMIN = 6, outliers are highlighted in color.
+ Outliers are mainly due to increased carbon monoxide emissions, only a few outliers with increased nitrogen oxide emissions.
+ Appropriate operation management should avoid the strongly increased carbon monoxide emissions.
+ Analysis of critical operating points shows a clear behavior of turbine temperatures.



Clustering

DBSCAN - Example With Outlier Detection

- Operating points have input temperatures between 1010 and 1100 °C and output temperatures between 515 and 552 °C.
- Turbine is operated so that input temperature does not exceed 1100 °C and turbine output temperature does not exceed 552 °C.
- Operating points with low emissions are close to these limits.
- Operating points with high carbon monoxide emissions have temperature combinations well below these limits, they are to be avoided.
- Cluster analysis allows identification of outliers for further process analysis.



Hochschule Karlsruhe
University of
Applied Sciences

Fakultät für
Elektro- und
Informationstechnik

www.h-ka.de

Manfred Strohrmann