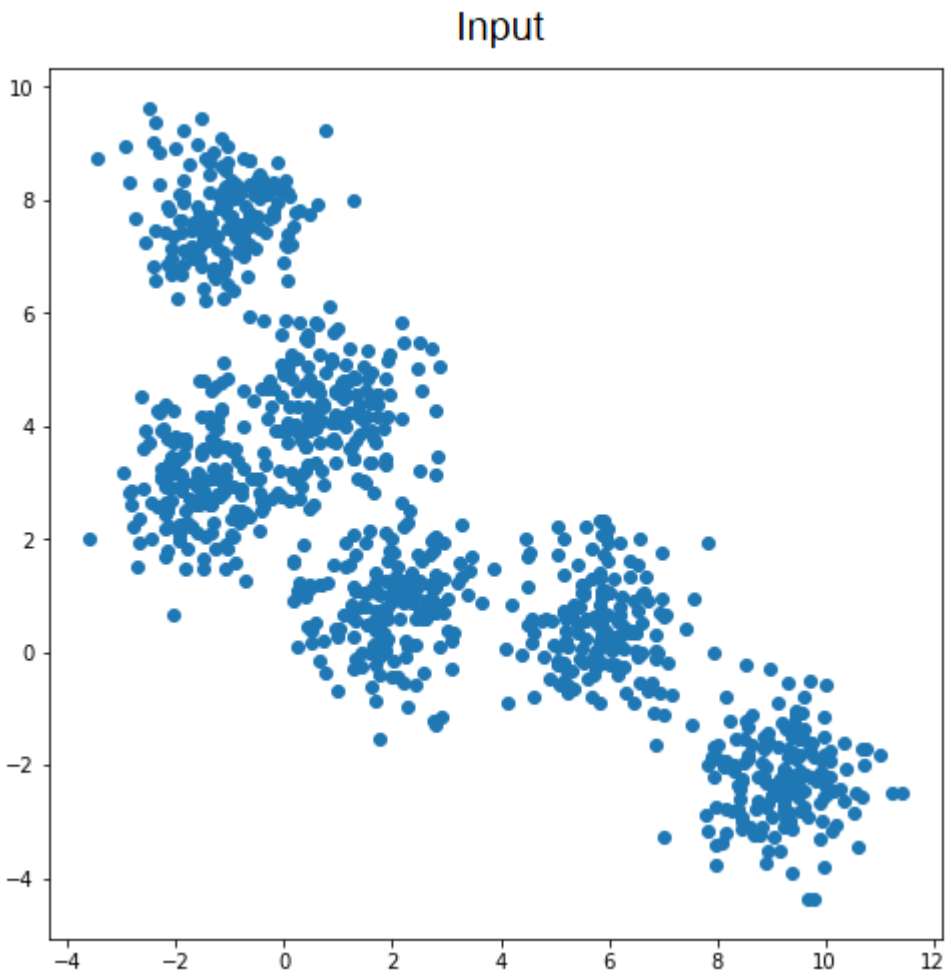
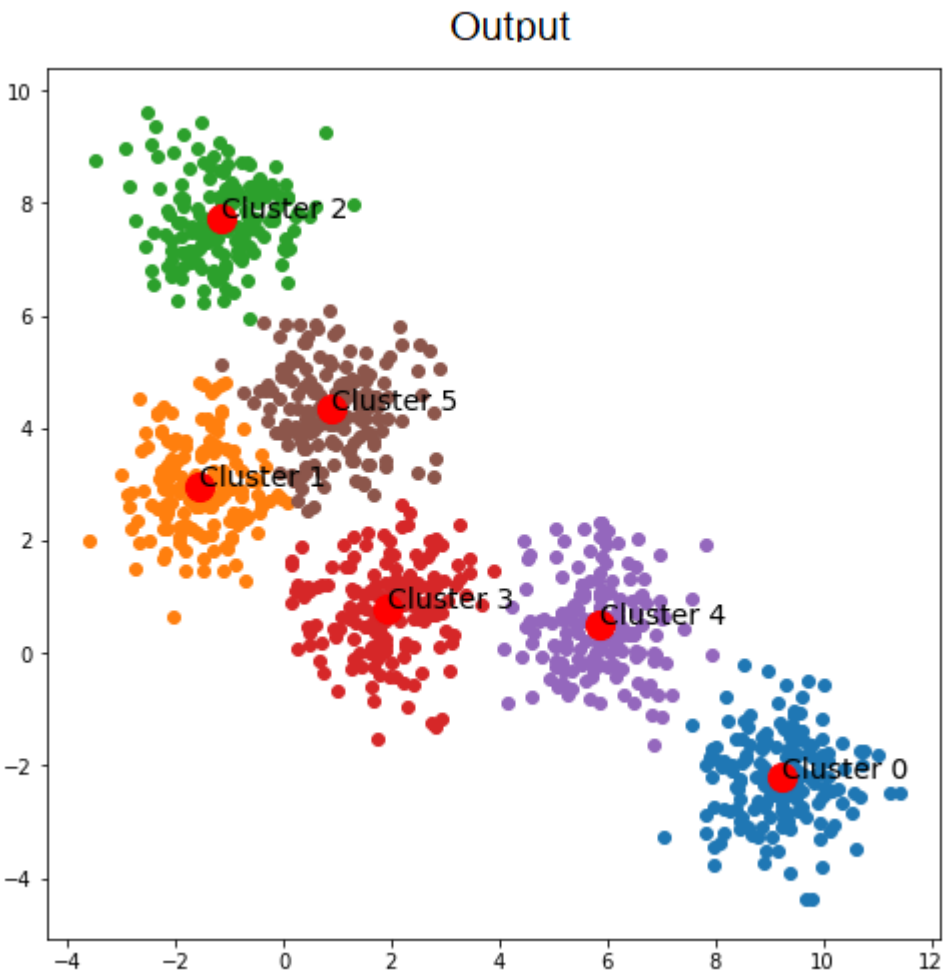
Cluster Analysis
Clustering

Clustering


Input
10
8
6
4
2
0
−2
−4
−4
−2
0
2
4
6
8
10
12


Output
Cluster 2
Cluster 5
Cluster 1
Cluster 3
Cluster 4
Cluster 0
10
8
6
4
2
0
−2
−4
−4
−2
0
2
4
6
8
10
12

Applications of Clustering

- Behavioural segmentation:
  - Segment by purchase history
  - Segment by activities on application, website, or platform
  - Define personas based on interests
  - Create profiles based on activity monitoring
- Inventory categorization:
  - Group inventory by sales activity
  - Group inventory by manufacturing metrics

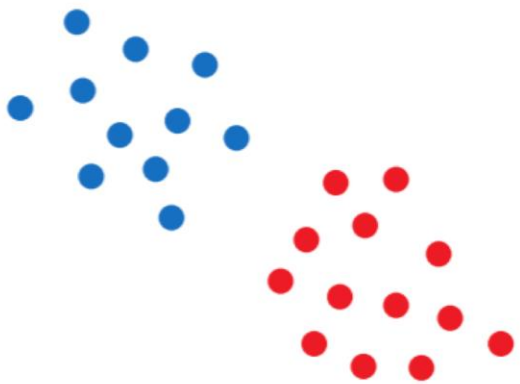
Applications of Clustering

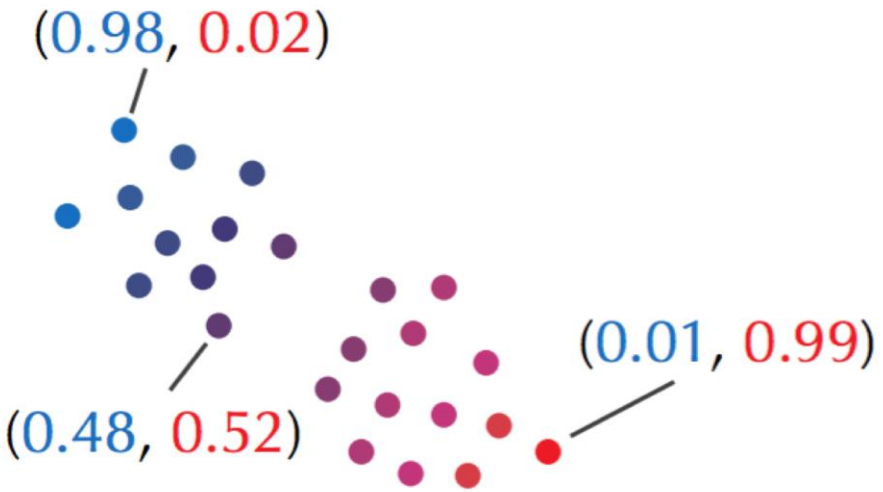
- Sorting sensor measurements:
  - Detect activity types in motion sensors
  - Group images
  - Separate audio
  - Identify groups in health monitoring
- Detecting bots or anomalies:
  - Separate valid activity groups from bots
  - Group valid activity to clean up outlier detection

Types of Clustering

- Clustering can be broadly divided into two subgroups:
- **Hard clustering:** in hard clustering, each data object or point either belongs to a cluster completely or not. For example in the Uber dataset, each location belongs to either one borough or the other.
- **Soft clustering:** in soft clustering, a data point can belong to more than one cluster with some probability or likelihood value. For example, you could identify some locations as the border points belonging to two or more boroughs.

Types of Clustering



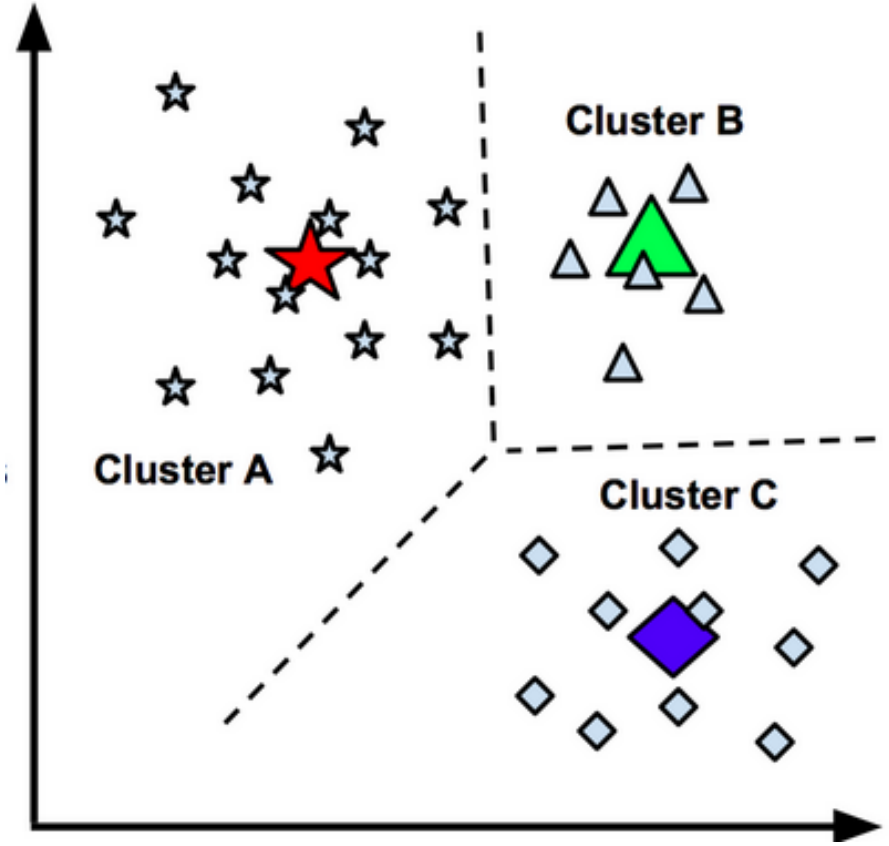


Hard choices: points are colored red or blue depending on their cluster membership.

Soft choices: points are assigned "red" and "blue" *responsibilities* r_{blue} and r_{red} ($r_{blue} + r_{red} = 1$)

1. Centroid-based clustering

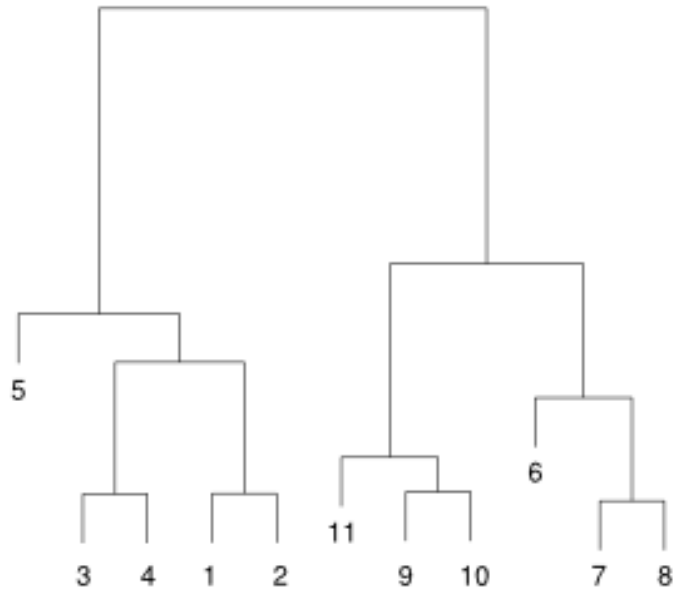
- In this type of clustering, clusters are represented by a central vector or a centroid.
- This centroid might not necessarily be a member of the dataset.
- This is an iterative clustering algorithms in which the notion of similarity is derived by how close a data point is to the centroid of the cluster.
- k-means is a centroid based clustering.

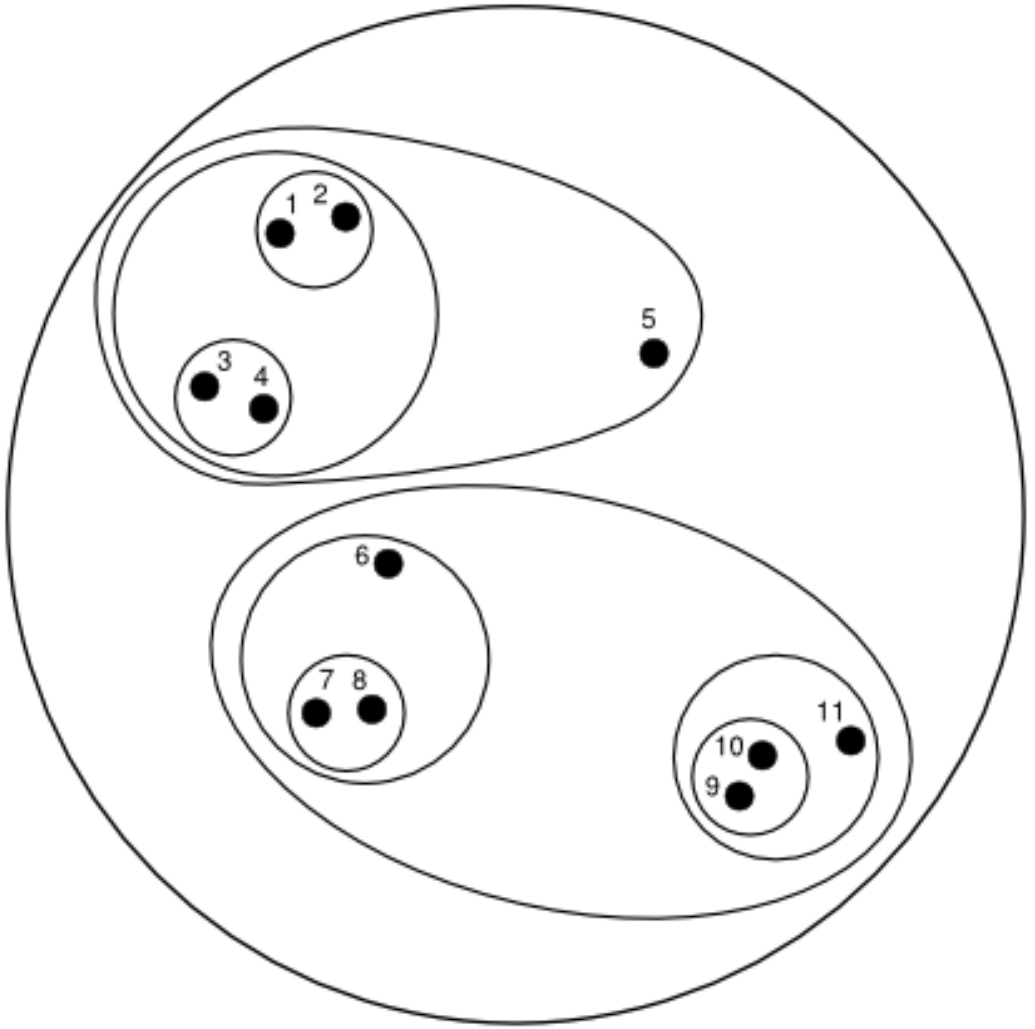


2. Connectivity-based clustering

- The main idea behind this clustering is that data points that are closer in the data space are more related (similar) than to data points farther away.
- The clusters are formed by connecting data points according to their distance.
- At different distances, different clusters will form and can be represented using a dendrogram, which gives away why they are also commonly called "**hierarchical clustering**".
- These methods do not produce a unique partitioning of the dataset, rather a hierarchy from which the user still needs to choose appropriate clusters by choosing the level where they want to cluster.
- They are also not very robust towards outliers, which might show up as additional clusters or even cause other clusters to merge.

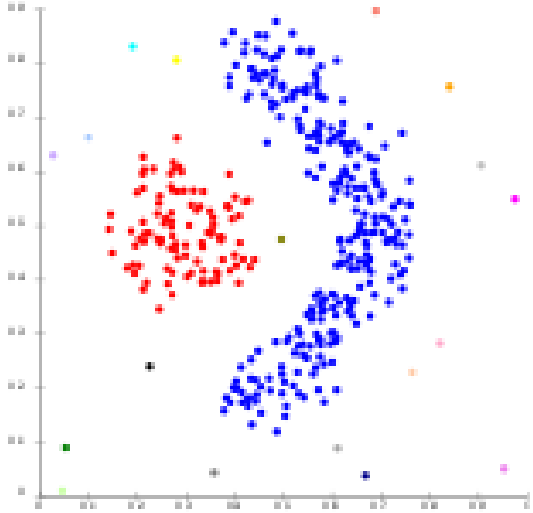
2. Connectivity-based clustering





3. Density-based clustering

- Density-based methods search the data space for areas of varied density of data points.
- Clusters are defined as areas of higher density within the data space compared to other regions.
- Data points in the sparse areas are usually considered to be noise and/or border points.
- The drawback with these methods is that they expect some kind of density guide or parameters to detect cluster borders.
- DBSCAN and OPTICS are some prominent density based clustering.

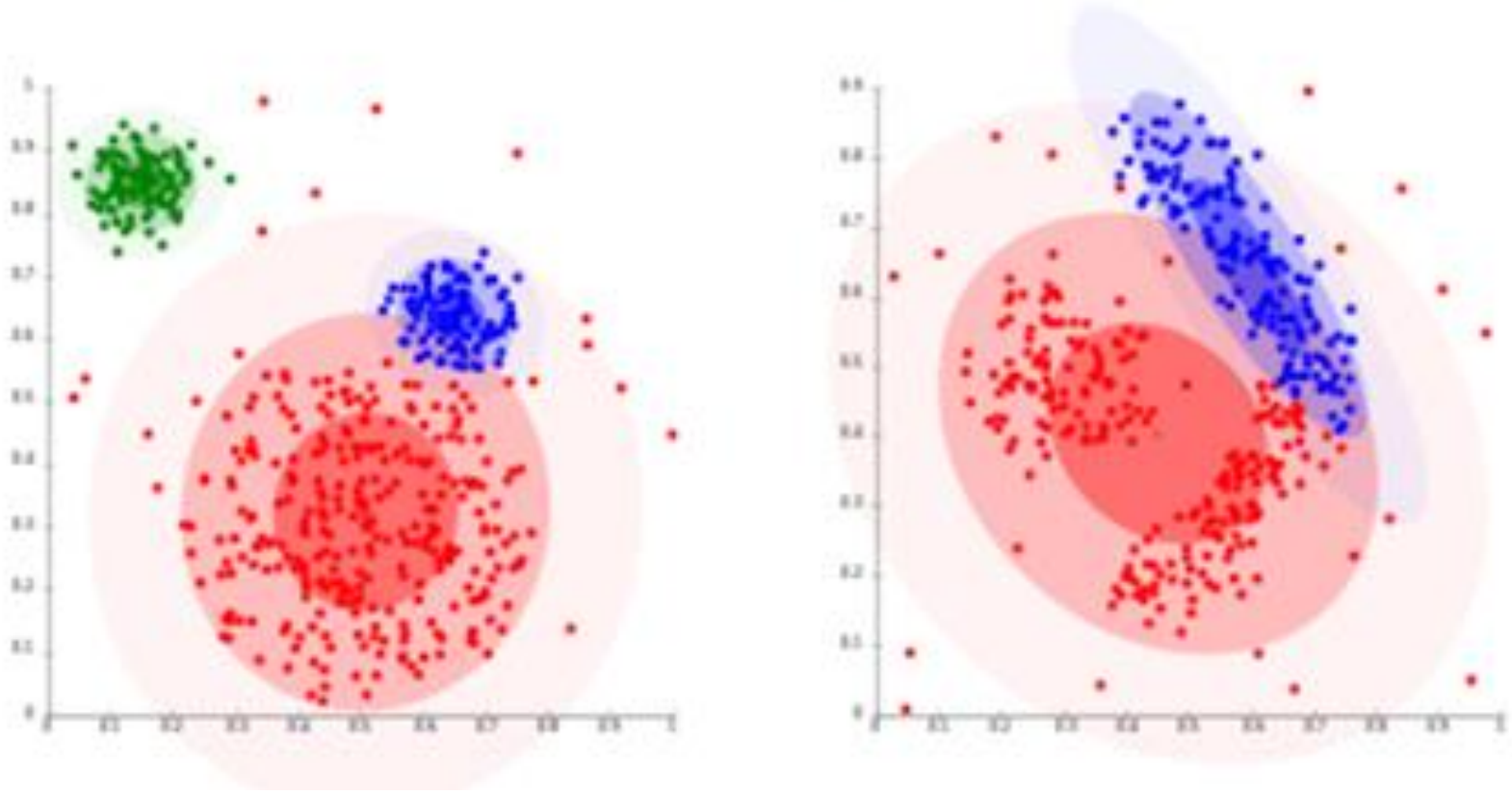


4. Distribution-based clustering

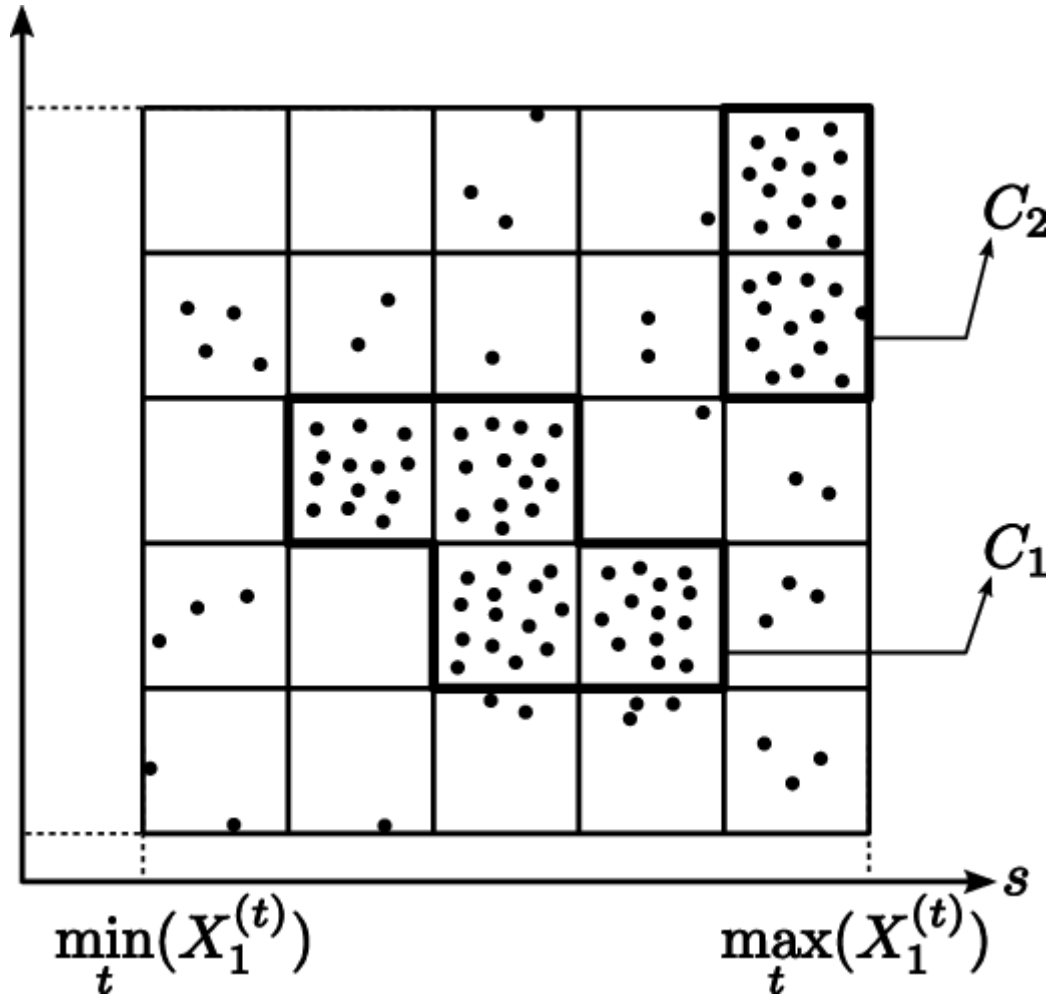
- This clustering is very closely related to statistics: distributional modeling.
- Clustering is based on the notion of how probable is it for a data point to belong to a certain distribution, such as the Gaussian distribution, for example.
- Data points in a cluster belong to the same distribution. These models have a strong theoritical foundation, however they often suffer from overfitting.
- Gaussian mixture models, using the expectation-maximization algorithm is a famous distribution based clustering method.

4. Distribution-based clustering

Multivariate Distribution-based Clustering



Grid-based clustering



Nandri Vanakkam