Toggle code

drawing

Week 10 - Clustering

Dr. David Elliott

1. Hierarchical Clustering
2. DBSCAN

4. Hierarchical Clustering

Hierarchical clustering is an approach to clustering which does not require that we commit to a apriori number of clusters.

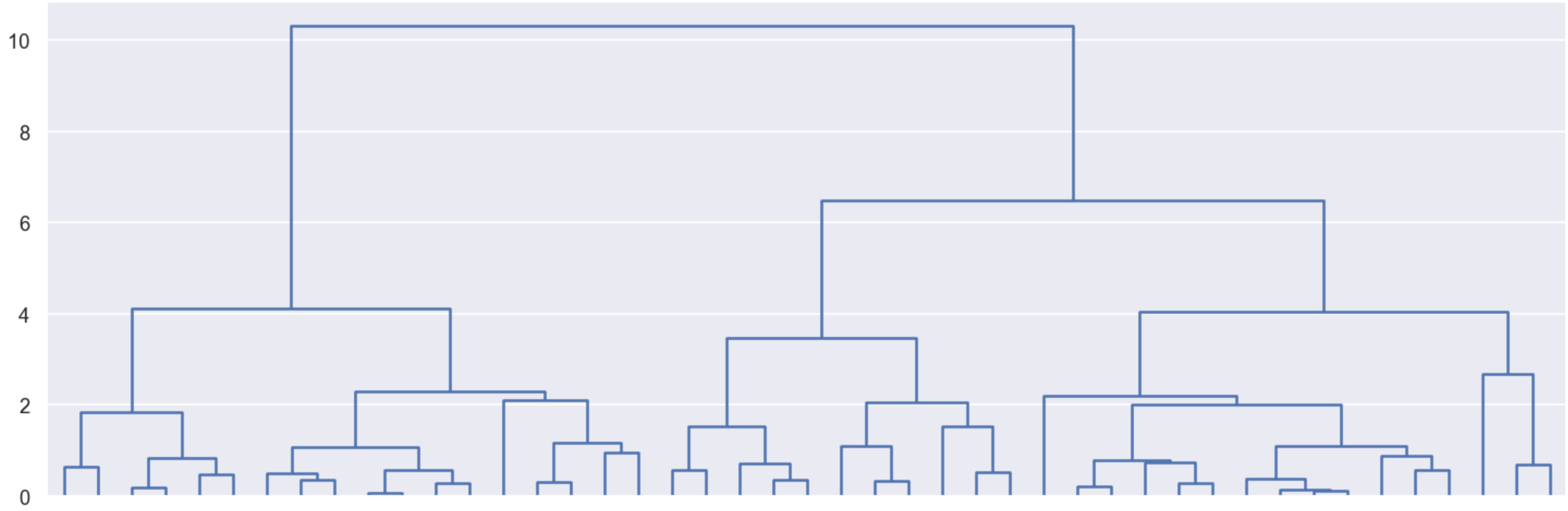
Instead we can determine the number of clusters using a tree-based representation of the observations, called a dendrogram[1].

Agglomerative (or bottom-up) clustering is the most common type of hierarchical clustering and refers to the fact that a dendrogram (generally depicted as an upside-down tree) is built starting from the leaves and combining clusters up to the trunk[1].

Notes

- Each leaf of the dendrogram represents an observation and as we move up the tree, some leaves begin to fuse into branches. These correspond to observations that are similar to each other[1].
- As we move higher up the tree, branches themselves fuse, either with leaves or other branches[1].
- The earlier (lower in the tree) fusions occur, the more similar the groups of observations are to each other. On the other hand, observations that fuse later (near the top of the tree) can be quite different[1].

Figure 11: Basic Dendrogram

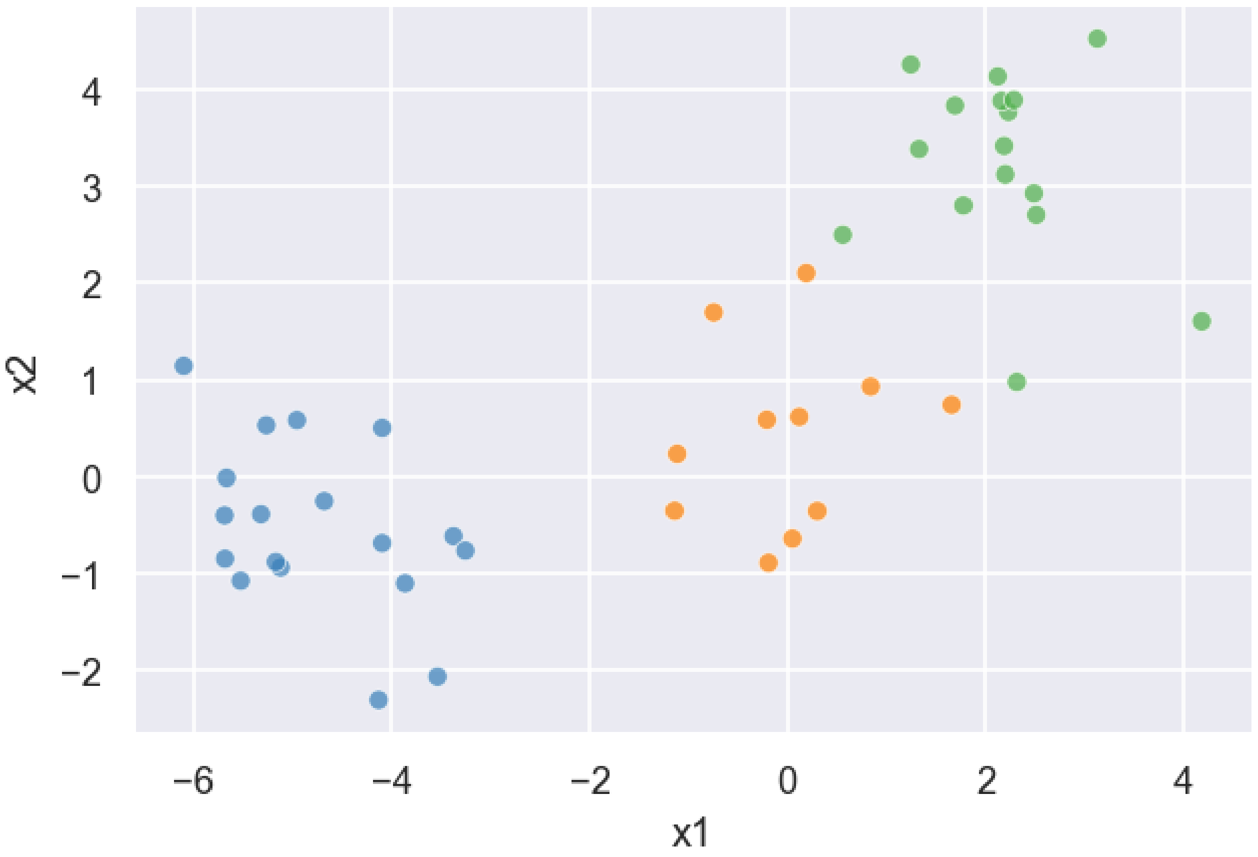


Data Example

We will use the following data generated data.

There are three distinct classes, shown in separate colors. However, we will treat these class labels as unknown and will seek to cluster the observations in order to discover the classes from the data[1].

Figure 12: Forty-five observations generated in two-dimensional space.



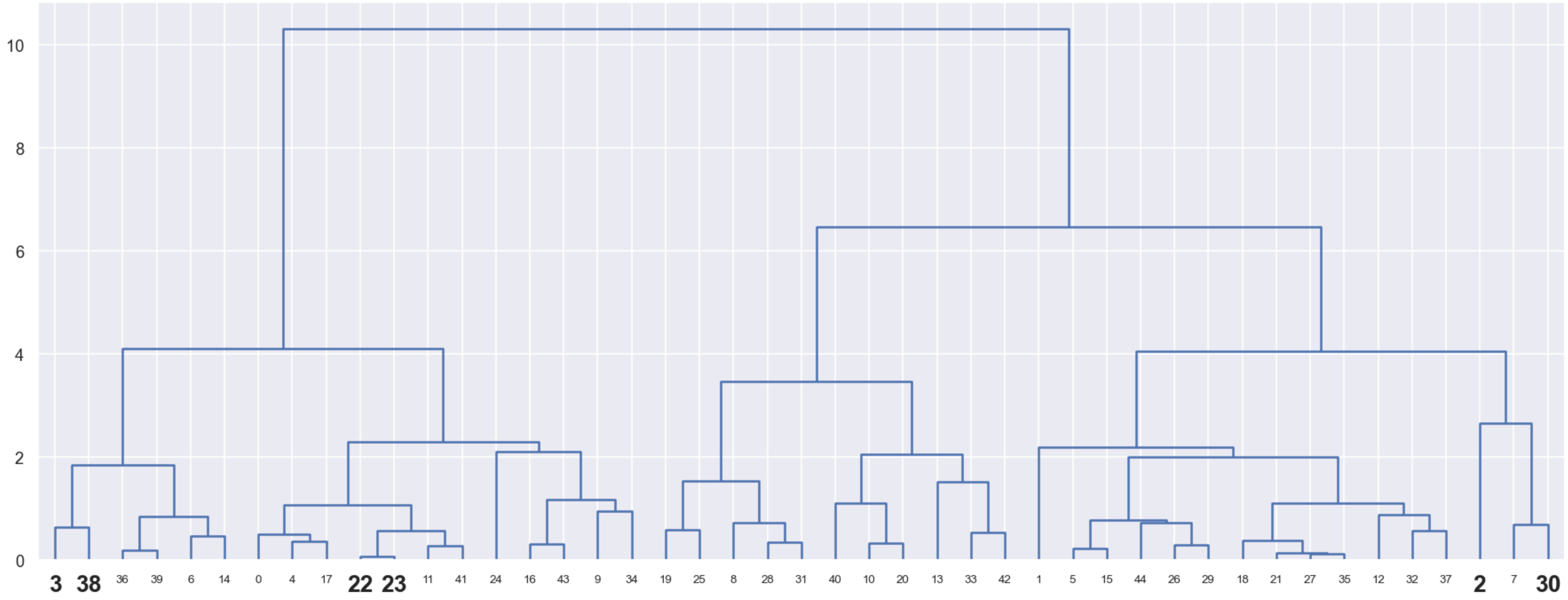
Each leaf of the dendrogram represents one of the 45 observations.

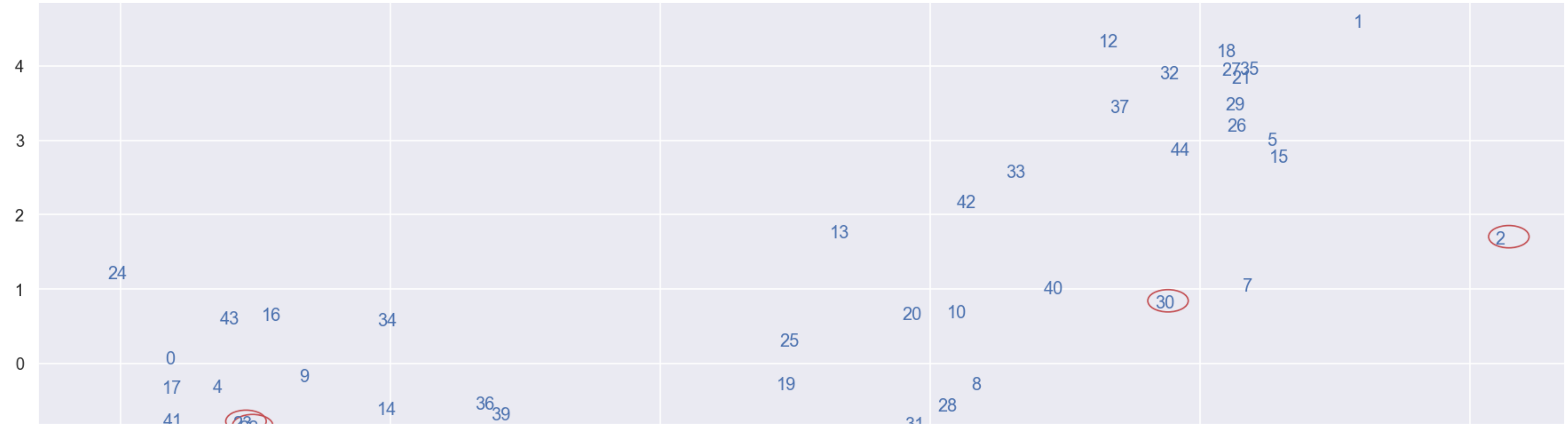
For any two observations, we can look for the point in the tree where branches containing those two observations are first fused.

The *height* of this fusion, indicates how different the two observations are. Observations that fuse at the bottom of the tree are quite similar to each other, whereas observations that fuse close to the top will generally be quite different[1].

- Observations close to each other on the *hoizontal axis* are not similar, we need to use the *vertical axis* to see where they are fused (see observation 37 and 2 for example)

Figure 13: Complete Linkage





In order to identifying clusters using a dendrogram we make a horizontal cut across it. The distinct sets of observations beneath the cut can be interpreted as clusters.

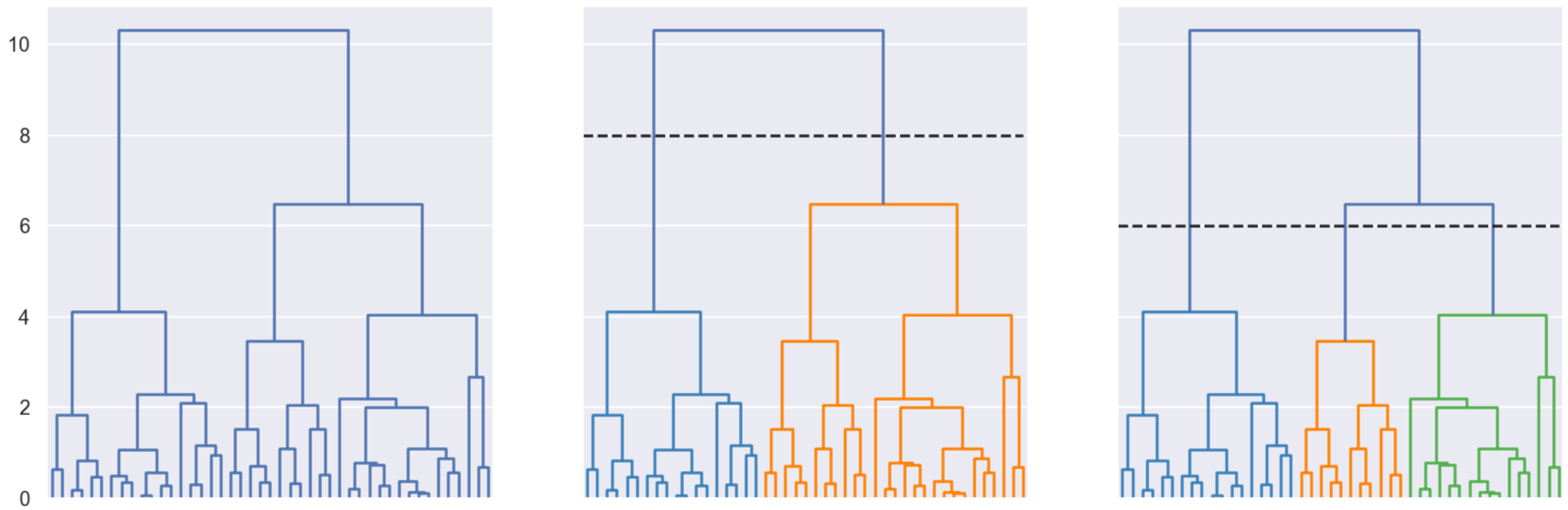
A dendrogram can be used to obtain any number of clusters. In practice, it is common select by eye a sensible number of clusters based on the heights of the fusions and the number of clusters desired.

However, often the choice of where to cut the dendrogram is not so clear.

Notes

- *"The height of the cut to the dendrogram serves the same role as the K in K-means clustering: it controls the number of clusters obtained."*[1]
- *"the dendrogram from the left-hand panel, cut at a height of nine (indicated by the dashed line). This cut results in two distinct clusters, shown in different colors. Right: the dendrogram from the left-hand panel, now cut at a height of five. This cut results in three distinct clusters, shown in different colors."*[1]

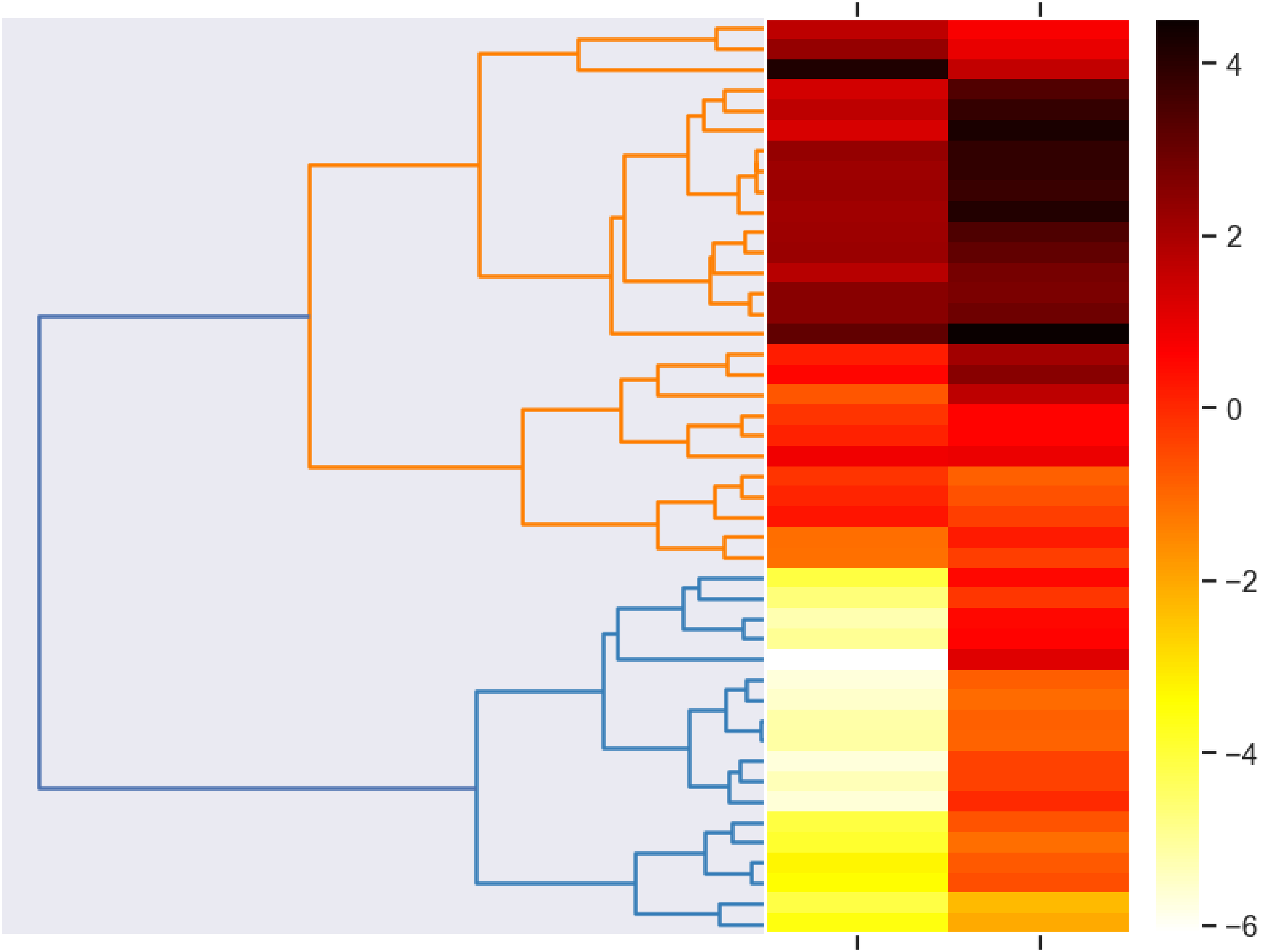
Figure 14: Dendrogram obtained from hierarchical clustering with complete linkage



	row label 1	row label 2	distance	no. of items in clust.
cluster 1	22.0	23.0	0.076208	2.0
cluster 2	27.0	35.0	0.125870	2.0
cluster 3	21.0	46.0	0.135057	3.0
cluster 4	36.0	39.0	0.191957	2.0
cluster 5	5.0	15.0	0.223856	2.0

Heirarchical clustering dendrograms are oten used in combination with a **heat map**, which represent each individual value with a color[2].

Figure 15: Dendrogram with Heatmap



The Hierarchical Clustering Algorithm[1]

1. Begin with n observations and a measure (such as Euclidean distance) of all the $\binom{n}{2} = \frac{n(n-1)}{2}$ pairwise dissimilarities. Treat each observation as its own cluster.

2. For $i = n, n-1, \ldots, 2$:

   (a) Examine all pairwise inter-cluster dissimilarities among the i clusters and identify the pair of clusters that are least dissimilar (that is, most similar). Fuse these two clusters. The dissimilarity between these two clusters indicates the height in the dendrogram at which the fusion should be placed.

   (b) Compute the new pairwise inter-cluster dissimilarities among the $i - 1$ remaining clusters.

Notes

- The algorithm uses a dissimilarity measure between each pair of observations (e.g. Euclidean distance).

- It then proceeds iteratively starting at the bottom of the dendrogram where each observation is treat as its own cluster. Then the two clusters that are most similar to each other are fused so that there now are $n - 1$ clusters. The algorithm then proceeds in this fashion until all of the observations belong to one single cluster.

Dissimilarity Measure

For this to work, the dissimilarity between a pair of observations also needs to be extended to a pair of groups of observations (linkage).

The four most commonly-used types of linkage in hierarchical clustering are[1]:

Complete

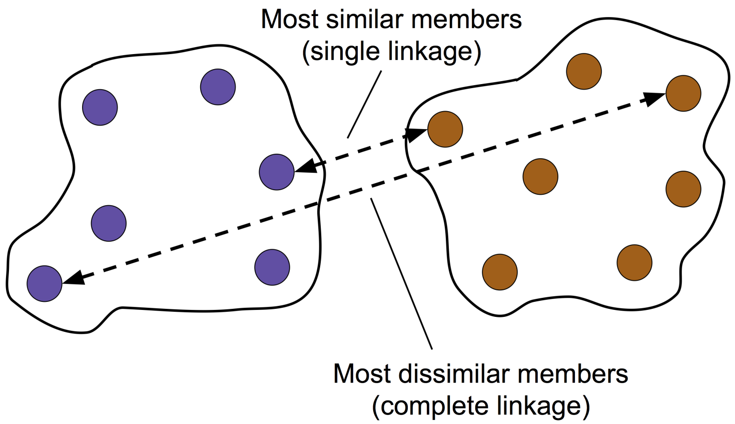
- Maximal intercluster dissimilarity
- Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the largest of these dissimilarities.

Single

- Minimal intercluster dissimilarity.

- Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the smallest of these dissimilarities.
- Single linkage can result in extended, trailing clusters in which single observations are fused one-at-a-time.

Figure 16: Different Linkage Methods



Average

- Mean intercluster dissimilarity.
- Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the average of these dissimilarities.

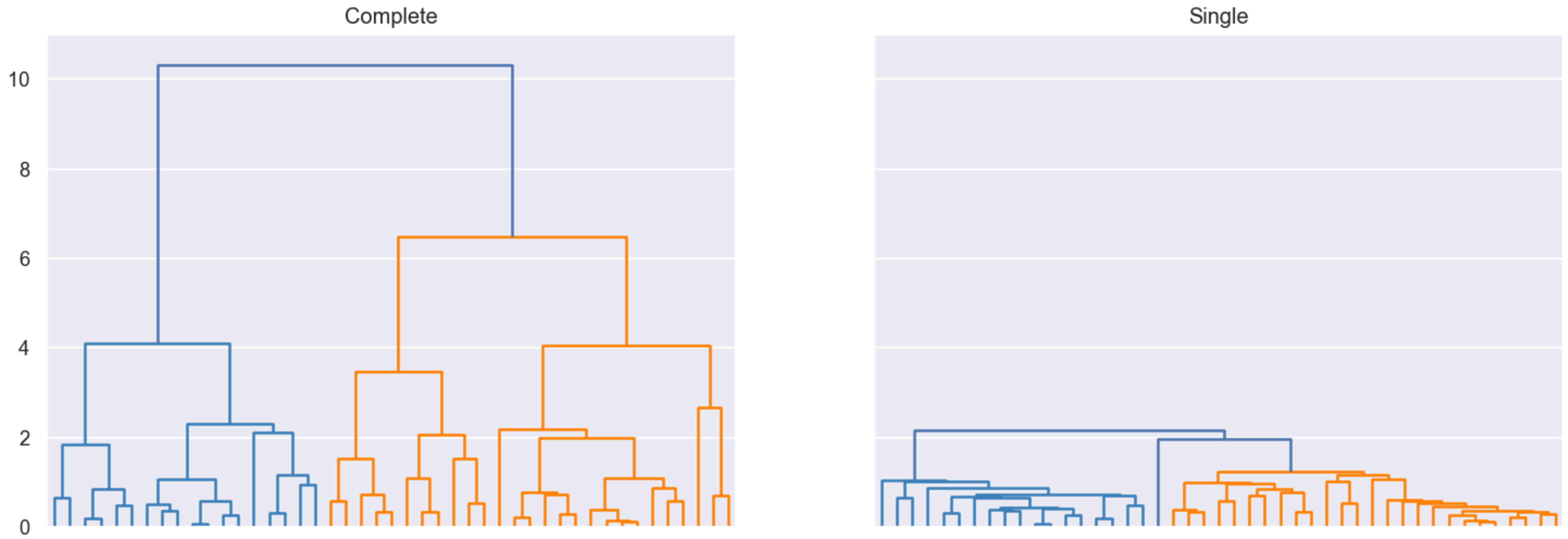
Centeroid

- Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B.
- Centroid linkage can result in undesirable inversions.

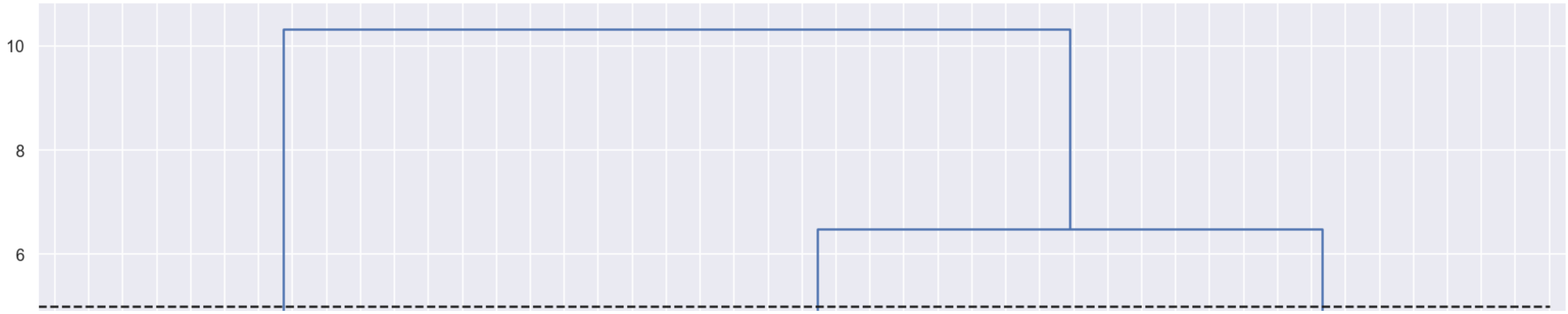
Notes

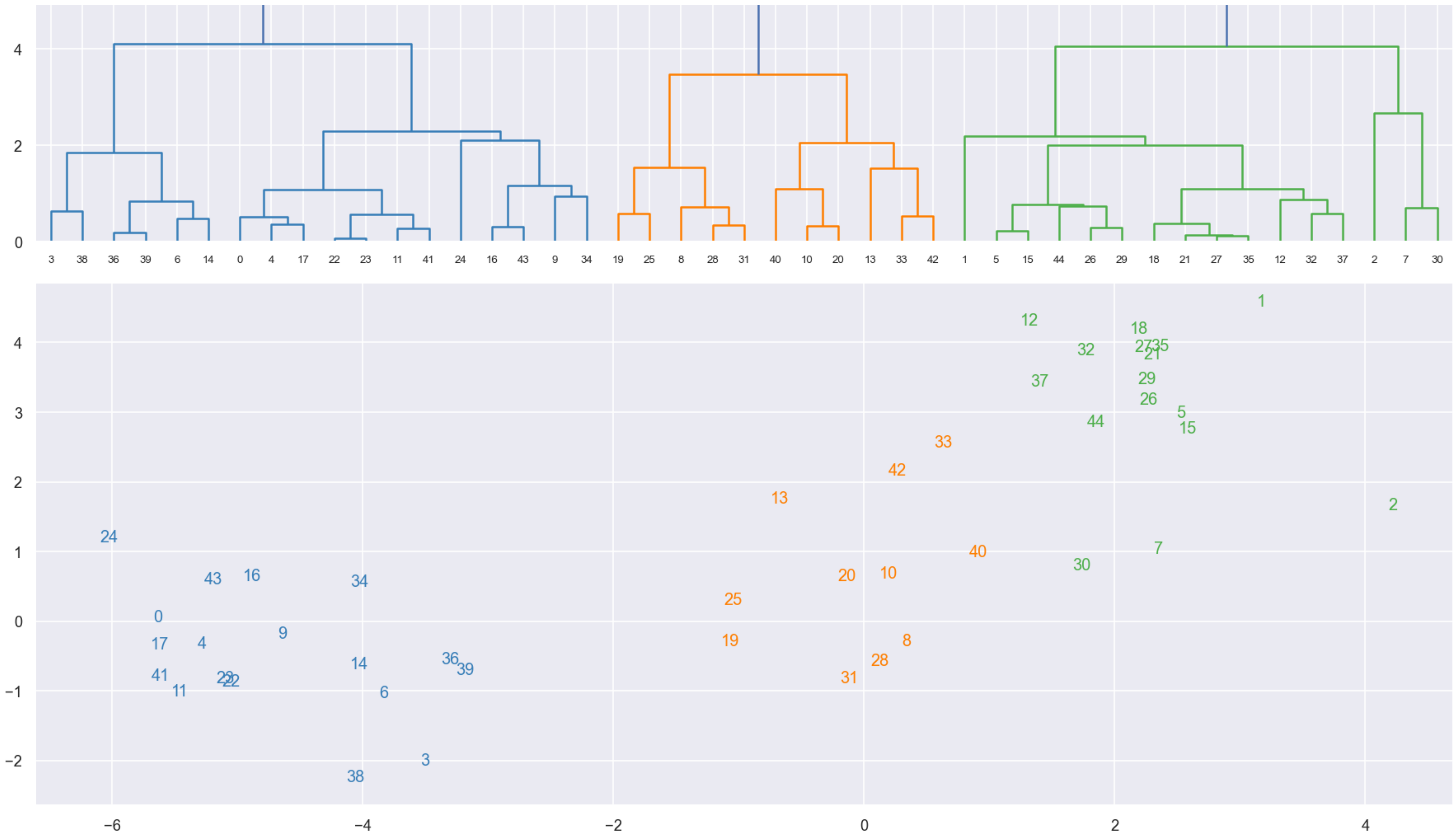
- *"Average, complete, and single linkage are most popular among statisticians. Average and complete linkage are generally preferred over single linkage, as they tend to yield more balanced dendrograms."*[1]

Figure 17: Dendrograms with different types of linkage



Extra Figure: Complete Linkage





Distance Measure

So far we have used Euclidean distance but other metrics exist.

Correlation-based distance[1]

- Considers two observations to be similar if their features are highly correlated.
- Values may be similar dispite being far apart in terms of Euclidean distance.

Choice is important but depends on data being clustered and the scientific question at hand.

5. DBSCAN

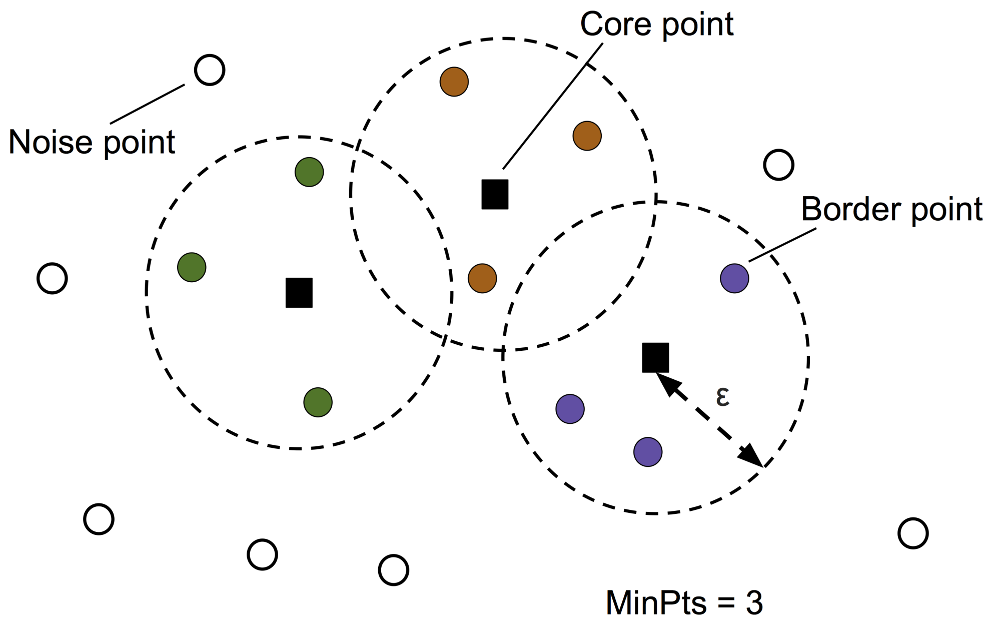
Density-based Spatial Clustering of Applications with Noise (DBSCAN) is density based clustering method.

For each instance:

- The algorithm counts how many instances are located within a small distance, ϵ, of it[3].
- If an instance has at least `min_samples` instances around it, then it is considered a **core instance**[3].
- If a point has fewer neighbours than `min_samples` within ϵ, but lies within the ϵ of a core instance, its considered a **border point**[2].
- All points not a core point or border point are considered **noise points** (or anomaly)[2].

Notes

- Clusters are defined as continuous regions of high density[3].
- The region around each instance is called an ϵ-neighbourhood[3].
- Core instances are instances located in dense regions[3].

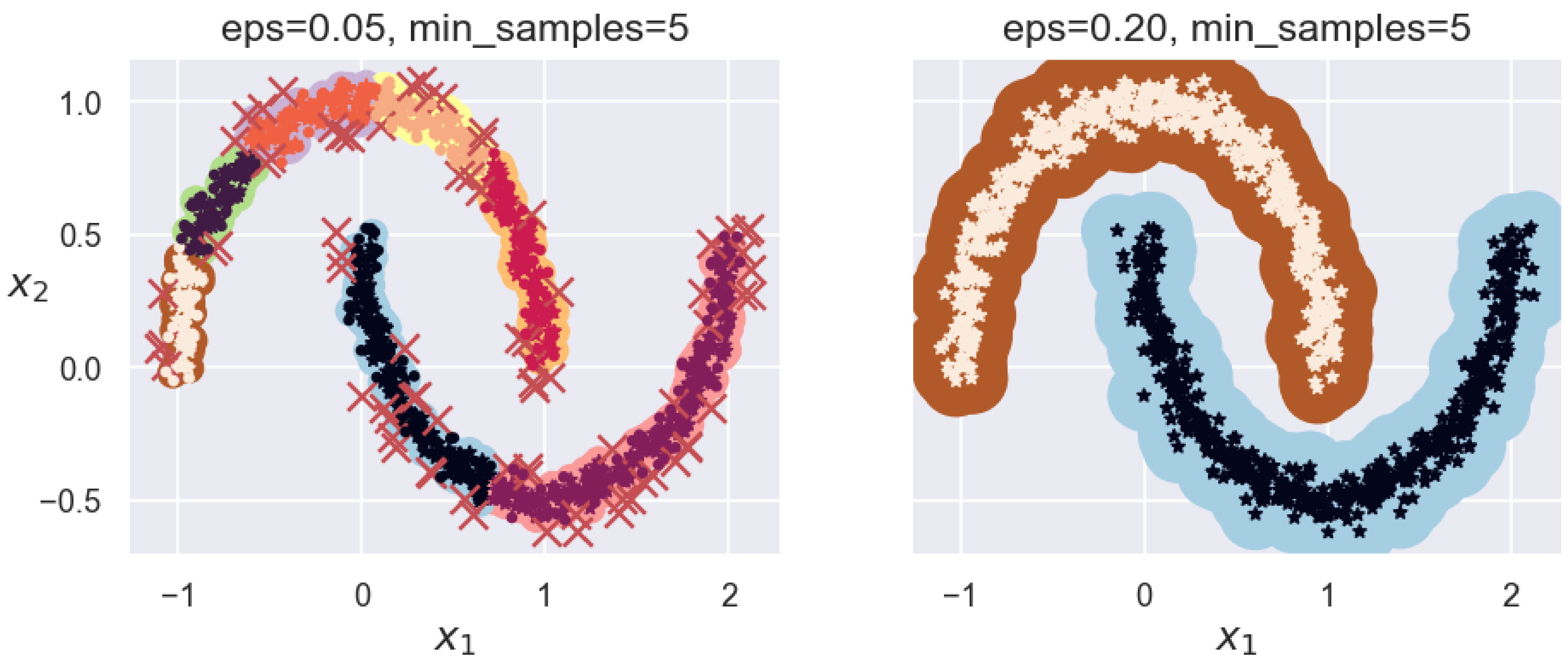


Once each point is labelled as either core, border, or noise, the algorithm[2]:

- Forms a separate cluster for each core point or connected group of core points provided they are no father than ϵ away from each other.
- Assigns each border point to the cluster of its corresponding core point.

Notes

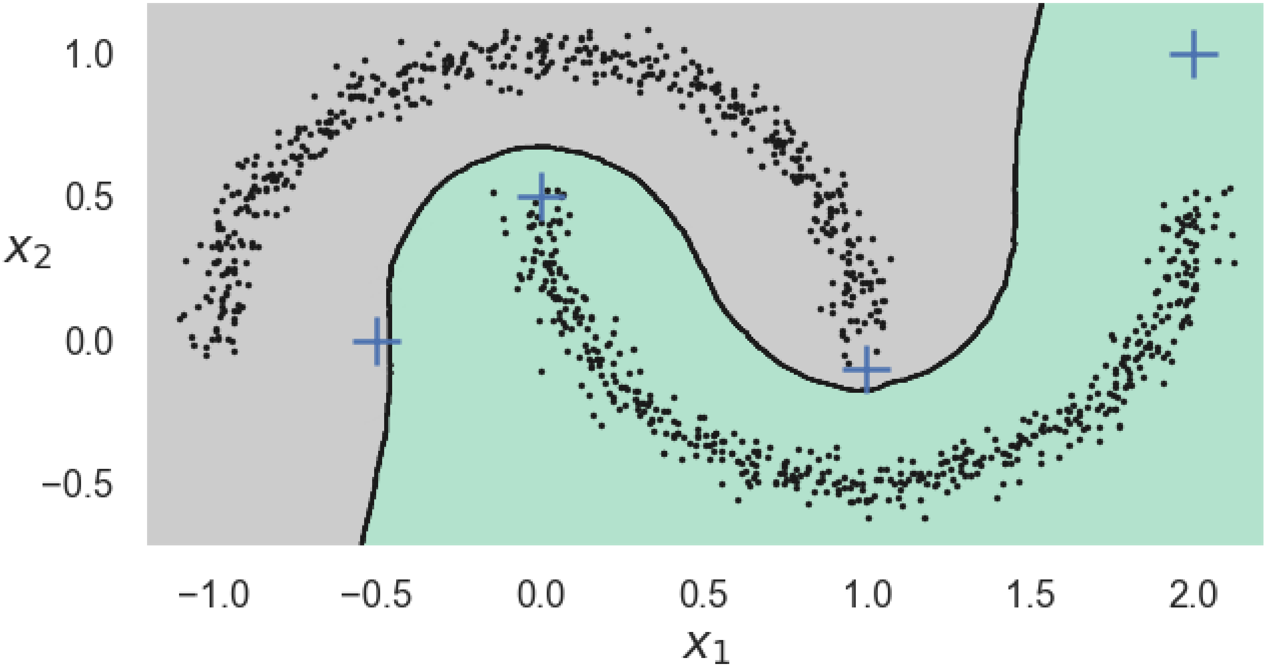
- As neighourhood can have multiple core instances, you can form a long sequence for a cluster[3].
- All instances in the neighbourhood of a core instance belong to a cluster, including other core instances[3].



DBSCAN is not very good at making prediction of cluster assignment for new observations, so is more commonly used in a pipeline with another classification model[3].

Notes

- We could train a model only using the core instances, all instances, or just the anomolies depending on what we are trying to achieve.
- If used in combination with an algorithm such as k-nearest neighbors, we can also set a maximum distance from clusters for them to be considered anomolie; see pg. 256 of Géron (2019).



Extra

You may want to try HDBSCAN which turns DBSCAN into a hierarchical clustering algorithm.

Associated Exercises

Now might be a good time to try exercise 3.

References

1. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.
2. Raschka, S., & Mirjalili, V. (2019). Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow 2. Packt Publishing Ltd.

3. Géron, A. (2019). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media.