Density Estimates & Density Based Clustering

Remember *k*-means. With *k*-means it is not possible to discover any clusters without specifying how many *k* clusters there should be. This is a serious problem, as often it is very hard to determine how many clusters there are in an unknown dataset.

This is where density based clustering comes in handy. Density based clustering is a means to fix this issue by creating clusters based on the density.

Flat Clustering

k-means

K-means is a flat clustering algorithm. It knows the number of *k* clusters it should find, and thus does not create any hierarchy of clusters.

Hierarchical Clustering

An analogy for hierarchical clustering could be the animal kingdom. Here animals are grouped into mammals, reptiles and insects. These groups contains subgroups that are species. Each species often also can be grouped into variants of this specie. This is very similar to hierarchical clustering where clusters may have subclusters, with may also contain subclusters and so on.

Finding a hierarchy of a structure can be done using two different approaches. The top-down and the bottom-up approach.

Top-down

The top-down approach starts by having all objects in one cluster. Then it divides this cluster into several subclusters, each with the same density. When the subclusters has been made, it looks for subclusters inside each subcluster. It keeps on doing this until it makes no sense to continue.

Bottom-up

The bottom-up approach starts by having all objects as singletons and then starts merging them based on density. Each time it merges a set of object, a cluster has been defined.

Outcome?

The outcome of either the top-down or bottom-up approach should ideally be identical. The uppermost cluster contains all subclusters and the bottom-most is a series of singletons (objects) each in their own cluster.

Single-link

Single link clustering is a bottom-up approach. Here all objects are checked how far they are from another closest object. Afterwards objects farther away are checked to see which of the current high-density clusters they are closest to.

Pros & Cons

Pros:

- Does not require any knowledge about number of clusters.
- Not only flat partition but hierarchy of clusters.
- A single partition can be retrieved by some horizontal cut.

Cons:

- If you want a flat partition, where is a good place to cut.

- Greedy heuristic; cannot correct bad decision.
- Single-link effect, complete-link effect.
- In general: inefficient.