Machine Learning in Data Science Project Week 7

Eva Aßmann, Paul Vogler

Dezember 2019

1 Data Generation

Five different data sets were generated using empirically determined parameters (Fig. 1): Using the scikit-learn make_classification function, a non-cluster data set was generated using random state 2. The data set contained 1000 samples with two informative and non-redundant features. Two classes were predetermined which were separated by the factor 0.3, distributed equally over all data points and each held two clusters. There was no label exchange allowed between the two classes.
A non-spherical data set was generated using the make_moons function from scikit-learn. Out of 1000 equally distributed samples two interleaving half circles of data points were scattered. Gaussian noise with standard deviation 0.1 was added to the data.
A data set with many clusters close to each other was generated using the make_blobs function from scikit-learn. At random state 37, out of 1000 equally distributed samples isotropic Gaussian blobs were created around six centroids. Each sample was assigned 2 features and the standard deviation of the clusters was set to 1.
Using the scikit-learn make_blobs function, a data set with clusters of different sizes was generated. Three blobs holding 750, 200 and 50 samples with two features each were created at random states 5, 12 and 43, respectively. Each cluster got a standard deviation of 0.8.
A data set with clusters of different densities was generated using the scitki-learn make_blobs function. Three blobs with 333, 333 and 334 samples and two features each were created at random states 16, 12 and 34, respectively. For the clusters standard deviations of 0.8, 1.2 and 0.3 were set.

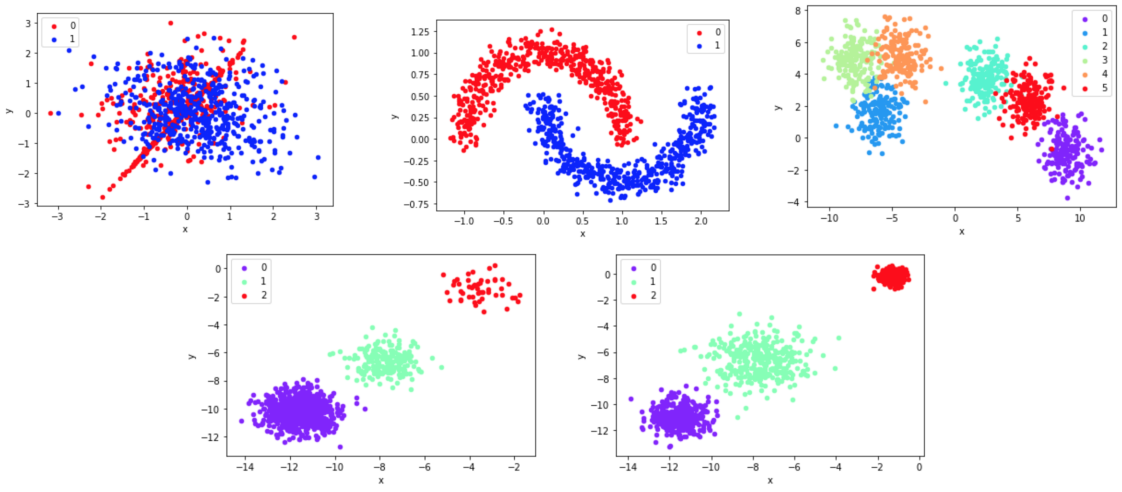

Figure 1: Generated data sets from top to bottom, from left to right: non-cluster, non-spherical, many clusters, clusters of different sizes and clusters of different densities.

2 K-means Clustering

For the k-means algorithm, the best number of clusters k had to be found first. The visually observed best number of clusters k1 was 2 for the non-cluster data, 2 for non-spherical, 6 for the many clusters set, 3 for the different sizes set and 3 for the different densities set. All of these were based on the original number of classes of the data sets.
In a second approach, k2 was chosen using the elbow method. The sum of squared distances was plotted against a range of possible k2 values. If the resulting plot looked like an arm, then the 'elbow' on the arm yielded the optimal k2 (Fig. 2). The best k2 values were 5 for the non-cluster data, 4 for non-spherical, 3 for the many clusters set, 3 for the different sizes set and 3 for the different densities set.

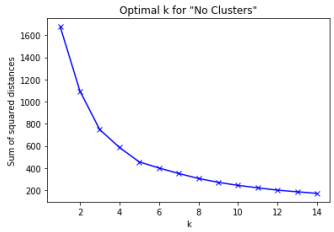

Figure 2: Best k2=5 for non-cluster data set determined by elbow method.

For each of the 5 data sets and for the respective k1 and k2 values, the data points, the cluster centroids and the cluster boundaries derived by the K-Means clustering algorithm from scikit-learn were plotted.
For the non-cluster set both k1=2 and k2=5 clusters only split the data in a regular shape that could easily be moved without providing a much better or worse clustering.
For the non-spherical set neither k1=2, nor k2=4 clusters could differentiate the two moon shapes (Fig. 3).

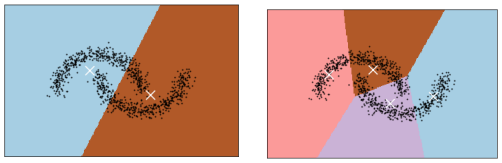

Figure 3: KMeans clustering result for the non-spherical data. Left: visually determined k, right: elbow-determined k

For the many-clusters set, k1=6 clusters accurately distinguished the 6 original blobs and the k2=3 clusters found at least sufficiently distinct groups (Fig. 4).

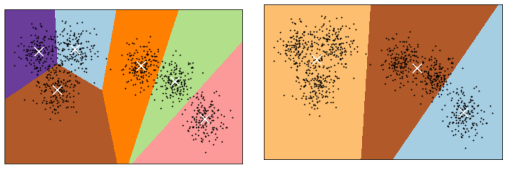

Figure 4: KMeans clustering result for the many-clusters data. Left: visually determined k, right: elbow-determined k

For both the different cluster sizes and densities sets the 3 original blobs were accurately distinguished by both optimal k values (Fig. 5).

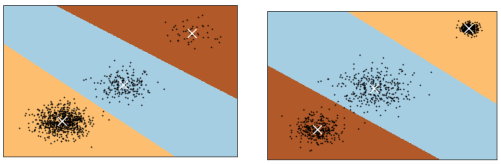

Figure 5: KMeans clustering result for the data clusters of different sizes (left) and densities (right).

In conclusion, the non-cluster and non-spherical sets could not be clustered accurately by the K-means for all used k, while blobs of data points with enough distance between them could be accurately distinguished.

3 Hierarchical Clustering

The five generated data sets were also analysed using bottom up hierarchical clustering with two different linkage methods, respectively. The minimum ('single') and maximum ('complete') linkage methods were chosen. For each data set the best number of clusters which was previously determined using the elbow method was used again in order to perform the unsupervised learning. Hierarchical clustering was performed using the AgglomerativeClustering method from scitkit-learn with the respective data set, number of clusters k and linkage method as input. The color coded clusters were plotted for every combination of data set and linkage method.
For the non-cluster set, the complete linkage method divided the data points into five differently sized clusters. The single linkage method yielded one big cluster and 4 very small clusters, each containing only one sample (Fig. 6).

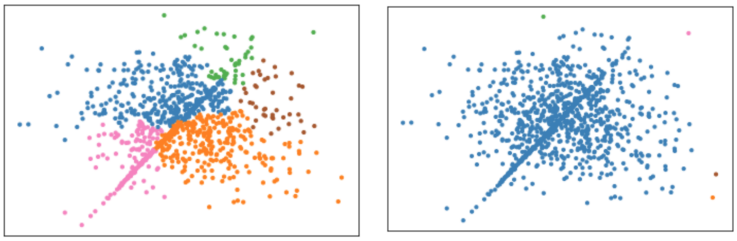

Figure 6: Hierarchically clustered non-cluster data with complete linkage (left) and single linkage method (right).

For the non-spherical set, the complete linkage split the points into four equally distributed clusters, whereas the single linkage put most of the two half spheres into two cluster with only one ore two samples left for each of the remaining two clusters (Fig. 7).

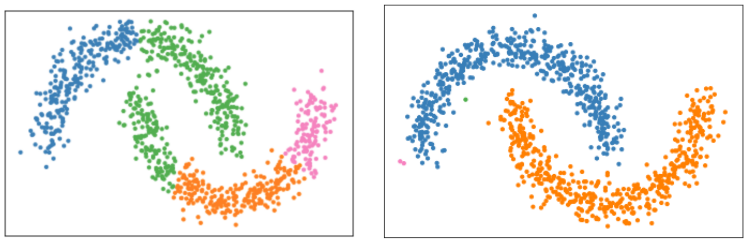

Figure 7: Hierarchically clustered non-spherical data using complete linkage (left) and single linkage method(right).

For the many-clusters set, the complete linkage method found three distinct clusters, while the single linkage only resulted in two larger clusters and a third single outlier point cluster (Fig. 8). For the different-sizes

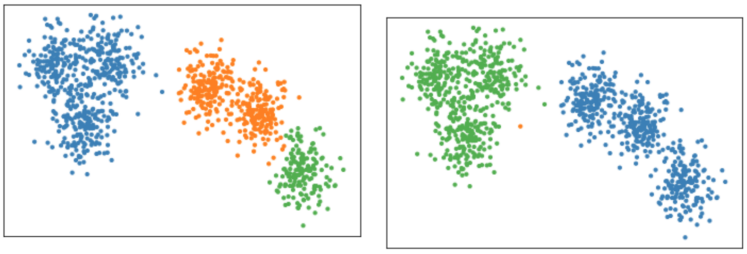

Figure 8: Hierarchically clustered many-clusters data using complete linkage (left) and single linkage method(right).

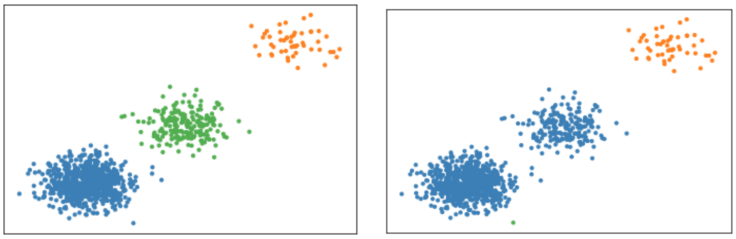

Figure 9: Hierarchically clustered data with clusters of different sizes using complete linkage (left) and single linkage method(right).

set and different-densities set, complete linkage generated three distinct clusters as well. Under the single linkage method, in both data sets two clusters stayed the same as with the complete linkage, while the prior third cluster was absorbed by one of these two (Fig. 9).
Overall, the complete linkage could distinguish more circular point clouds even with small distances between the clouds, while the single linkage could even differentiate groups with non-circular shapes like the moons, but needed a significant distance between point clouds to distinguish them.

4 Discussion

Both K-means and the hierarchical clustering can accurately determine blobs of data points with some distance between them, because they both use (for hierarchical only using the average linkage) a centroid based distance measure that is optimal in a circular shape. But other than K-means, the hierarchical clusters can group non-circular point shapes together for some of the distance measures, like minimum distance linkage. This works because as long as the distances inbetween the points of the non-circular cloud are smaller than the minimal length of the gap between two clouds, the points will be clustered into the same group.