1 Clustering

Clustering is one of the central tasks in machine learning. Given a set of data points, the purpose of clustering is to partition the data into a set of clusters where data points assigned to the same cluster correspond to similar data points.

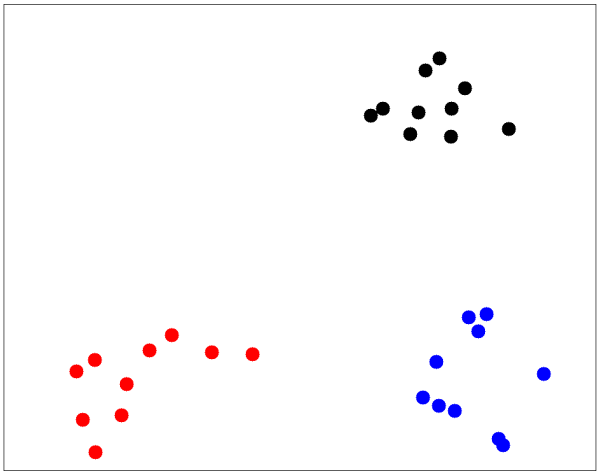

Figure 1: Given a set of data points, we want to partition the data into a set of clusters where data points assigned to the same cluster correspond to similar data points.

An intuitive definition of clustering would consist in trying to partition of objects (data points) into subsets such that subset consists of "similar" objects.

Typical examples where clustering arises are:

1. Newspaper articles: cluster them into articles of similar topics.
2. Medicine: cluster patients into groups with similar symptoms.
3. Social network: cluster into different communities.
4. Marketing: cluster customers into groups of similar buying habits.
5. Images: cluster into groups with specific contents.

To make clustering defined above actually workable, we need a notion of similarity. Mathematically speaking, we need to choose an appropriate distance (norm). Often we use Euclidean norm. Once the norm is chosen, we need to set a certain criterion for how to select clusters based on this norm.

2 The k-means criterion

The most common criterion for defininig clusters is the *k-means* criterion. It is based on minimizing the pairwise distance of data points within the same cluster.

Formally, given $x_1, \ldots, x_n \in \mathbb{R}^d$, we partition these points into k clusters $C_1, \ldots, C_k$ based on the objective

$$\min_{C_1,\ldots,C_k} \sum_{l=1}^{k} \frac{1}{|C_l|} \sum_{i,j \in C_l} \|x_i - x_j\|^2. \tag{1}$$

This cost function is a weighted average of the cluster variances, with weights proportional to cluster size in terms of number of points $|C_l|$. We derive a more tractable form. Let $\mu_l = \frac{1}{|C_l|} \sum_{i \in C_l}^{n} x_i$ be the centroid, the center of mass of points, in C_l. Note that

$$\begin{aligned}
\sum_{i,j \in C_l} \|x_i - x_j\|^2 &= \sum_{i,j \in C_l} \Big(\|x_i\|^2 + \|x_j\|^2 - 2\langle x_i, x_j \rangle \Big) \\
&= \sum_{i \in C_l} \Big(|C_l| \|x_i\|^2 + \sum_{j \in C_l} \|x_j\|^2 - 2|C_l| \langle x_i, \mu_l \rangle \Big) \\
&= 2|C_l| \sum_{i \in C_l} \|x_i\|^2 - 2|C_l|^2 \|\mu_l\|^2.
\end{aligned}$$

Furthermore,

$$\begin{aligned}
\sum_{i \in C_l} \|x_i - \mu_l\|^2 &= \sum_{i \in C_l} \Big(\|x_i\|^2 + \|\mu_l|^2 - 2\langle x_i, \mu_l \rangle \Big) \\
&= \sum_{i \in C_l} \Big(\|x_i\|^2 + |C_l| \|\mu_l\|^2 - 2|C_l| \|\mu_l\|^2 \Big) \\
&= \sum_{i \in C_l} \|x_i\|^2 - |C_l| \|\mu_l\|^2.
\end{aligned}$$

Hence,

$$\frac{1}{2} \sum_{i,j \in C_l} \|x_i - x_j\|^2 = |C_l| \sum_{i \in C_l} \|x_i - \mu_l\|^2,$$

and therefore minimzing $\sum_{l=1}^{k} \frac{1}{|C_l|} \sum_{i,j \in C_l} \|x_i - x_j\|^2$ is equivalent to solving

$$\min_{\substack{C_1,\ldots,C_k \\ \mu_1,\ldots,\mu_k}} \sum_{j=1}^{k} \sum_{i \in C_j} \|x_i - \mu_j\|_2^2. \tag{2}$$

Note that finding the solution to the k-means objective (2) is a highly non-convex problem, and it finding its solution is NP-hard. Thus, assuming the conjecture $P \neq NP$, there is no polynomial-time algorithm to find the solution to the k-means objective (2). The following lemma suggests a two-step approach.

Lemma 1. *Let $\{x_1 \ldots, x_n\}$ be a set of points. The expression $\sum_{i=1}^{n} \|x_i - \mu\|_2^2$ is minimized when μ is the centroid, i.e. $\mu = \frac{1}{n} \sum_{i=1}^{n} x_i$.*

Proof. Set $\alpha_i = 0$ and $V = 0$ in the first step of the PCA-proof. □

Thus, if we knew the correct centers μ_j, we could easily assign each x_i to the right clusters by solving

$$\min_j \|x_i - \mu_j\|_2^2 \tag{3}$$

This suggests an iterative algorithm, known as Lloyd's algorithm (and sometimes referred to as k-means algorithm):

1. Choose k initial cluster centers $\mu_1, \ldots, \mu_k$.
2. Assign each point x_i to its correct cluster C_j according to $j = \text{argmin} \|x_i - \mu_j\|_2^2$.
3. Update the centers μ_j based on the new clusters.
4. Repeat step 2 and 3 until convergence to some stopping criterion.

Despite its popularity, Lloyd's algorithm suffers from some major drawbacks:

1. Verification if computed solution is the global optimum is in general very expensive.
2. Different initializations will in general give different solutions.
3. Lloyd's algorithm is not guaranteed to converge to the true solutions.

The main advantage of Lloyd's algorithm is its computational efficiency.

There are also some issues with the k-means objective (2), regardless of the shortcomings of Lloyd's algorithm:

- (D.Mixon) Consider two circles, each of radius 1, their centers are a distance d apart. As long as $d > 2.08$, K-means yields correct answer. But if $d \leq 2.08$, K-means fails. This is the failure of K-means and not of Lloyd's algorithm.
- K-means will always produce convex clusters, thus it can only work if clusters can be linearly separated.

The k-means objective function fails sometimes even in cases that are linearly separable and appear easy, see the examples in Figure 5, where k-means fails once the data set gets more and more dilated.

Therefore, we would like a clustering algorithm that can detect the underlying geometry of the data. This leads us to graphs, spectral clustering, and diffusion maps.

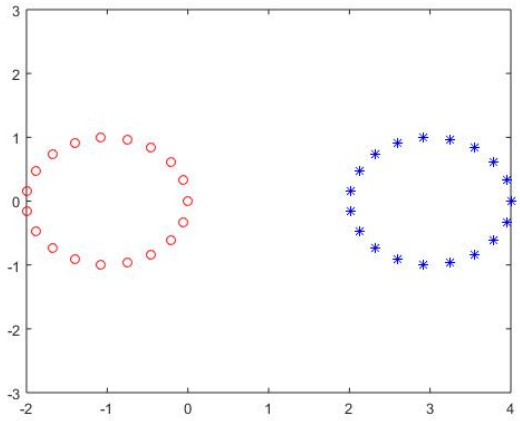

Figure 2: $d > 2.08$

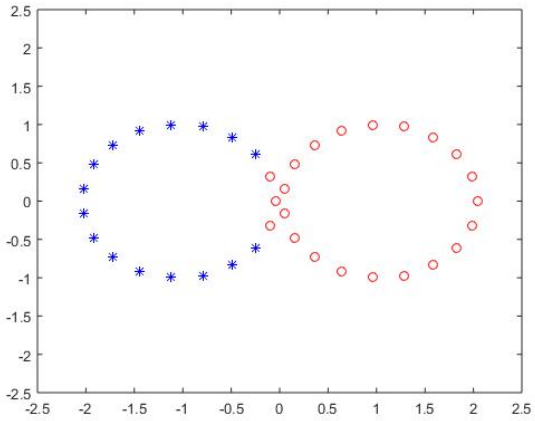

Figure 3: $d \leq 2.08$

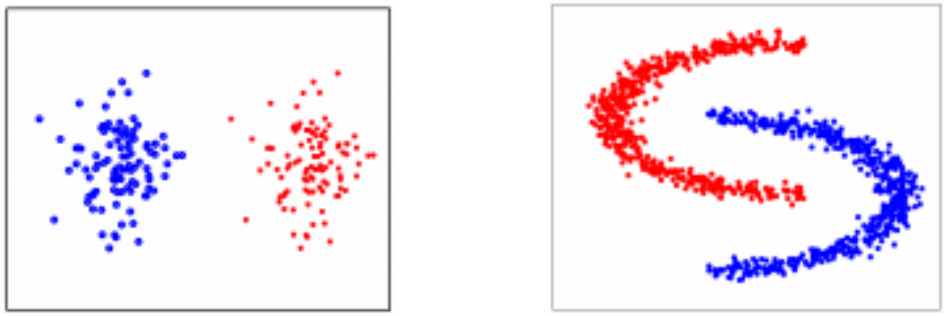

Figure 4: The k-means solution produces the expected two clusters for the case depicted in the left image, but fails to produce the expected result for the example depicted in the right image.

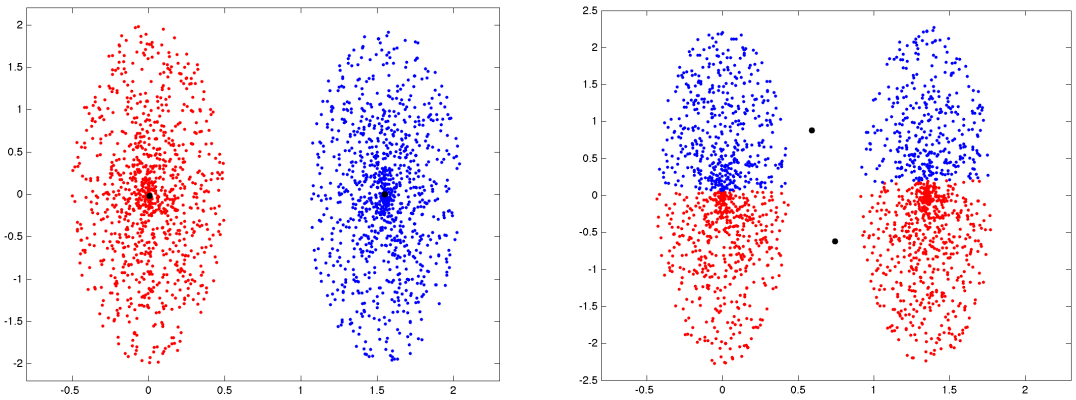

Figure 5: The k-means solution produces the expected two clusters for the case depicted in the left image, but fails to produce the expected result for the example depicted in the right image.