3.2.2 EM-Clustering

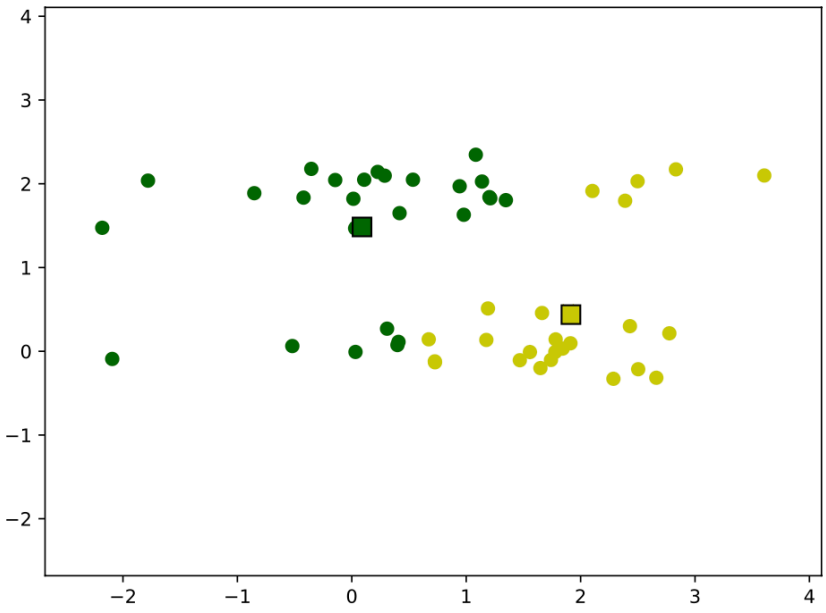

Figure 3.3: K-means fails to recognize the elongated structures in the data.

To address the difficulties mentioned in the last chapter with the K-means algorithm, we consider a stochastic approach. We assume that the data is distributed according to a Gaussian Mixture Model, i.e. that the points of a cluster are normally distributed around a mean m_k. In mathematical terms, this means that the data of the k-th group C_k has a density function of the form

3.2.9

$$p_k(x) := \frac{1}{\sqrt{(2\pi)^d \det \Sigma_k}} \exp\left(-\frac{1}{2}(x-m_k)^T \Sigma_k^{-1}(x-m_k)\right), \quad x \in \mathbb{R}^d$$

where $m_k \in \mathbb{R}^d$ is a mean of cluster k and $\sum_k \in \mathbb{R}^{d \times d}$ is a covariance matrix for $k = 1, \ldots, K$ (describes shape and spread of the cluster). $(x-m_k)^T \sum_k{}^{-1}(x-m_k)$ measures the **Mahalanobis** distance, which generalizes Euclidian distance by considering correlations between variables.

$\frac{1}{\sqrt{(2\pi)^d \det \sum_k}}$ normalizes the Gaussian function so that the probability integrates to 1

Note that in constrast to Section 2.2.1, we must work with multivariate normal distributions here. In the case of $d = 1$ and for $\sum_k := \sigma_k^2$ (3.2.9) simplifies to the familiar density of a scalar normal distribution with mean $m_k \in \mathbb{R}$ and variance σ_k^2:

$$p_k(x) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{(x-m_k)^2}{2\sigma_k^2}\right)$$

The distribution of all data can then be modeled by a density function of the form
(Instead of a single Gaussian function, the whole dataset is modeled as a **mixture** of multiple Gaussian distributions)

3.2.10

$$p(x) := \sum_{k=1}^{K} \pi_k p_k(x), \quad x \in \mathbb{R}^d$$

where $0 \leq \pi_k \leq 1$ with $\sum_{k=1}^{K} \pi_k = 1$.

Explanation

- π_k
  - These are the **mixing coefficients**, representing the probability that a randomly chosen data point belongs to cluster k
  - They must satisfy $0 \leq \pi_k \leq 1$ and sum to 1!
- $p_k(x)$
  - The probability density function for cluster k
  Thus the full data distribution $p(x)$ is a weighted sum of **individual** Gaussian distributions, where each cluster contributes proportionally to its prior probability π_k

As in Bayesian classification, the numbers π_k represent the probabilities of the k-th group C_k. Our goal is to calculate the means m_k, the covariance matrices $\sum_k$, and the group probabilities π_k by utilizing the data

For this purpose, the so-called ***responsibilities*** (determine probability that point x belongs to cluster k) of the k-th group for the point x, denoted as $\gamma(x,k)$ are helpful. These are defined as

3.2.11

$$\gamma(x,k) := \mathbb{P}(C_k|x) = \frac{\pi_k p_k(x)}{p(x)} = \frac{\pi_k p_k(x)}{\sum_{k=1}^{K} \pi_k p_k(x)}$$

where we have used Bayes' theorem for densities.

Exkurs

$$\mathbb{P}(C_k|x) = \frac{\mathbb{P}(x|C_k)\mathbb{P}(C_k)}{\mathbb{P}(x)}$$

- $\mathbb{P}(x|C_k) \to p_k(x)$:
  - The likelihood of observing the point x given that it belongs to cluster C_k. Modeled as Gaussian distribution for cluster k
- $\mathbb{P}(C_k) \to \pi_k$
  - The prior probability of cluster k, representing the proportion of data expected in cluster k
- $\mathbb{P}(x) \to p(x)$
  - The total probability of observing x across **all clusters**, which serves as the denominator

Explanation

The numerator $\pi_k p_k(x)$ represents the probability that x belongs to cluster k, considering the prior π_k and the likelihood $p_k(x)$
The denominator is

$$p(x) = \sum_{k=1}^{K} \pi_k p_k(x)$$

This means that $p(x)$ is the weighted sum of the likelihoods of x under each cluster, weighted by their respective priors π_k

The so-called EM (Expectation Maximization) algorithm now starts from an initial initialization of the parameters $\pi_k, m_k, \sum_k$ and updates these using the following rules:

Initialization

Basically figure out the square or space you are in and either place points randomly or if you have two clusters, you compute a mean point, and pertubate this point into two points then and thus initialize the parameters. Elsewise often K-Means is just used for initialization

First, we define a kind of weight for the k-th group as the sum of $\gamma(x, k)$ over all data points via

3.2.12

$$N(k) \leftarrow \sum_{i=1}^{N} \gamma(x_i, k)$$

Explanation: weight for k-th group is calculated by summing the responsibilities for each point x_i with respect to cluster k. $N(k)$ is the total weight assigned to cluster k, based on how much each point belongs to that cluster. Remember $\gamma(x_i, k)$ is the **responsibility**, which indicates the probability that the point x_i, belongs to cluster k.

We can now update the group probabilities π_k by

3.2.13

$$\pi_k \leftarrow \frac{N(k)}{N}$$

Explanation: update prior probability for each cluster. N is the total number of data points, $N(k)$ is the sum of responsibilities for the k-th cluster (see above!)

Next, we update the means through a weighted average of the points, where the weights are the responsibilities

3.2.14

$$m_k \leftarrow \frac{1}{N(k)} \sum_{i=1}^{N} \gamma(x_i, k) x_i \in \mathbb{R}^d$$

Thus, the points contribute significantly to the k-th mean for which the k-th group has a large responsibility. (Points that have a higher responsibility for cluster k will have more influence on the new mean of that cluster).

Finally, we update the covariance matrices using a weighted empirical covariance matrix:

3.2.15

$$\Sigma_k \leftarrow \frac{1}{N(k)} \sum_{i=1}^{N} \gamma(x_i, k)(x_i - m_k)(x_i - m_k)^T \in \mathbb{R}^{d \times d}$$

Explanation

The covariance matrix is updated to reflect the spread of the data points around the new mean for each cluster, weighted by their responsibilities, where $(x_i - m_k)$ is the vector of deviations from the mean point for x_i. Multiplying this with its transposed form gives a covariance matrix! This is then weighed by the responsibility. We divide by the total effective weight of cluster k because this normalizes the sum, resulting in the **weighted empirical covariance matrix** for cluster k

An iteration of the EM algorithm thus executes the steps 3.2.12 to 3.2.15 for all $k = 1, .., K$. In the next iteration, the responsibilities $\gamma(x, k)$ defined in 3.2.11 are calculated using the updated parameters of the Gaussian Mixture Model. This is iterated for a fixed number of iterations or until the parameters change only slightly. See figure below:

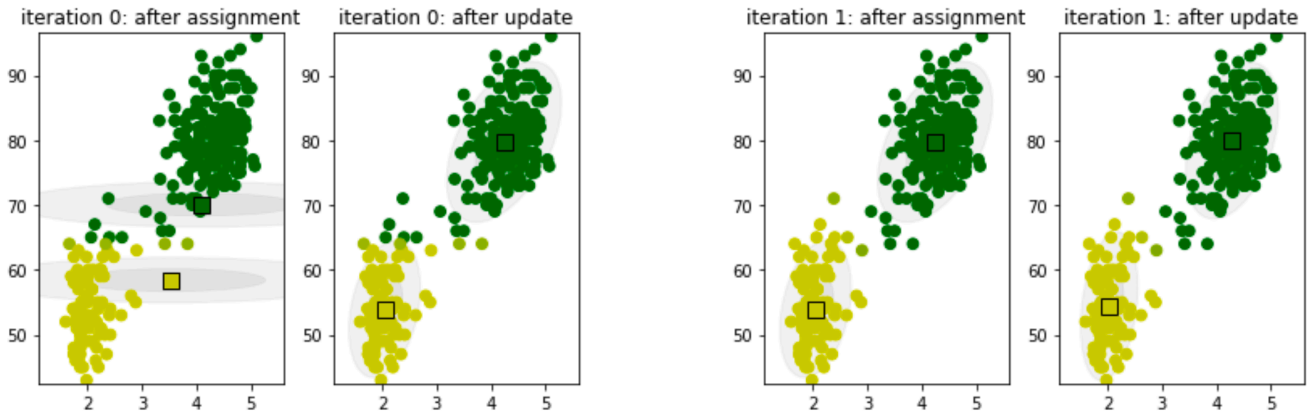

Figure 3.4: Two iterations of the EM algorithm for $K = 2$.

The main difference between K-means and EM lies in the covariance matrices Σ_k, which allow the algorithm to adapt locally to the data, similar to what we observed in principal component decomposition. It should also be noted that the vector $(\gamma(x, k))_{k=1}^{K}$ is the sought probability vector that contains the group probabilities for a fixed data point $x \in \mathbb{R}^d$