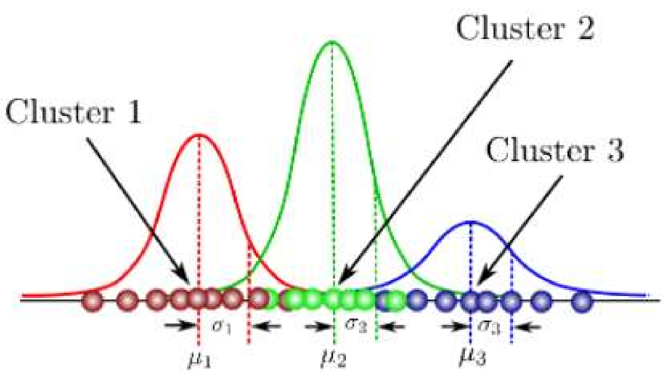
Go Back to Data Analysis & Visualization

Course Content

Gaussian Mixture Models

Gaussian Mixture Models (GMMs) are an unsupervised machine learning technique used for clustering data. K-means performs clustering taking only the mean of the data into consideration, but Gaussian Mixture Models use the mean as well as the variance of the data for clustering.

Gaussian Mixture Models assume that the underlying data represents a mixture of Gaussians, as shown in the image below.



Gaussian Mixture Models are considered probabilistic models for representing normally distributed subpopulations within an overall population. The Gaussian Mixture Model tries to learn the parameters (mean, variance) of each Gaussian in the dataset to cluster the data points. The parameters mean (μ) and variance (σ^2), of each Gaussian, are learned using the **Expectation-Maximization (EM) algorithm**.

How does the EM algorithm work?

The EM algorithm has 2 steps:

- The E-step
- The M-step

The E-Step and the M-Step are repeated until the parameters mean (μ) and variance (σ^2) **converge**.

E-step:

1. It will start by placing K Gaussians randomly with mean μ_k and variance $\sum_k$ for the **k**th Gaussian as shown below. The value of k is passed as a parameter.

2. Then for each Gaussian/cluster we calculate the cluster weights π_k which show us how much each cluster is represented over all data points (each cluster's relative size) and can be given as,

$$\pi_k = \frac{Total\ Number\ Of\ Points\ Assigned\ To\ Gaussian/Cluster}{Total\ Number\ Of\ Points}$$

3. Then for each datapoint assign a soft assignment,

$$r_{ik} = \frac{\pi_k\ N(x_i\ |\ \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j\ N(x_i\ |\ \mu_j,\ \Sigma_j)}$$

Where, r_{ik} is the probability that x_i is generated from the **k**th cluster and K is the total number of clusters.

M-step:

After computing the soft assignments for the current iteration, **we use it to update parameters - mean and variance**.

First, we calculate the sum of probabilities of each data point for each cluster.

Let's say we have a point x_i, its probability vector can be given as

$$[r_{i1}, r_{i2}, \dots, r_{ik}]$$

Where,

r_{i1} = probability of point i belonging to cluster 1

r_{i2} = probability of point i belonging to cluster 2

r_{ik} = probability of point i belonging to cluster k

This vector is calculated for each and every data point in the dataset.

Next, we calculate the sum of probabilities for all points belonging to a particular cluster. It can be mathematically represented as,

$$N_k^{soft} = \sum_{i=1}^{N} r_{ik}$$

N_k^{soft} = sum of all probabilities for all points belonging to cluster k

Then, the cluster weights $\pi_1,\ \pi_2,\ \dots,\ \pi_k$, show us how much each cluster is represented over all data points (each cluster's relative size). The cluster weight of each cluster is given by the ratio of the soft count $N_1^{soft},\ N_2^{soft},\ \dots,\ N_k^{soft}$ over the total number of data points N_k. It's mathematical representation can be given as,

$$\pi_k = \frac{N_k^{soft}}{N_k}$$

Now we re-estimate the mean $\mu_1,\ \mu_2,\ \dots,\ \mu_k$ of each cluster using the formula below,

$$\mu_k = \frac{1}{N_k^{soft}} \sum_{i=1}^{N} r_{ik} x_i$$

Where,

$\sum_{i=1}^{N} r_{ik} x_i$ = sum of all probabilities of cluster for which we are re-estimating the mean multiplied by its true value

Now we re-estimate the variance $\Sigma_1, \Sigma_2, \ldots, \Sigma_k$, of each cluster using the formula below,

$$\Sigma_k = \frac{1}{N_k^{soft}} \sum_{i=1}^{N} r_{ik} (x_i - \mu_k)(x_i - \mu_k)^T$$

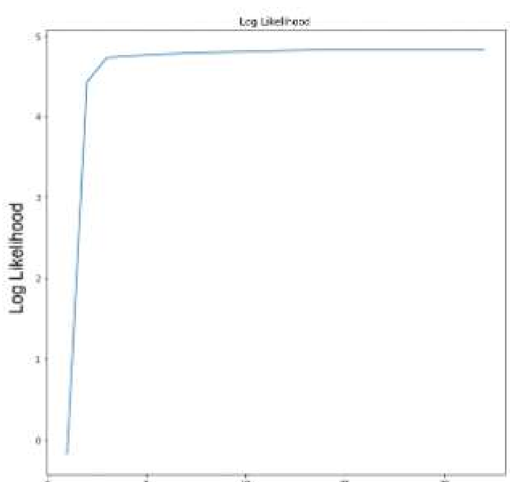
Finally, we repeat the E-step and the M-step until the estimates of parameters converge.

Convergence

We use the below function, called log-likelihood, to decide the convergence of parameters.

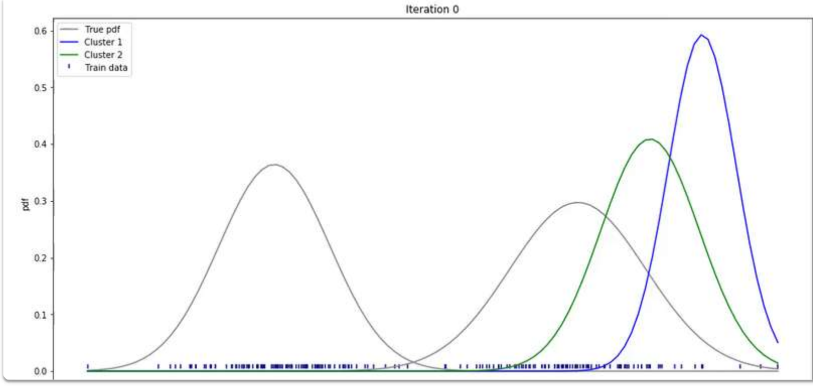
$$ln(p(X|\pi, \mu, \Sigma)) = \sum_{i=1}^{N} ln\{\sum_{j=1}^{K} \pi_j N(x_i \mid \mu_j, \Sigma_j)\}$$

When the value of this formula does not improve by a tolerance/threshold level then we can say that the algorithm has converged.



In the above graph, we can see that the change in value after 15 iterations is very small, hence we can say that the values have converged and the model has fit successfully.

The below graphic is a representation of the Gaussian Mixture Model learning the parameters at each iteration to resemble the true distribution.


Iteration 0
True pdf
Cluster 1
Cluster 2
Train data
pdf
0.6
0.5
0.4
0.3
0.2
0.1
0.0