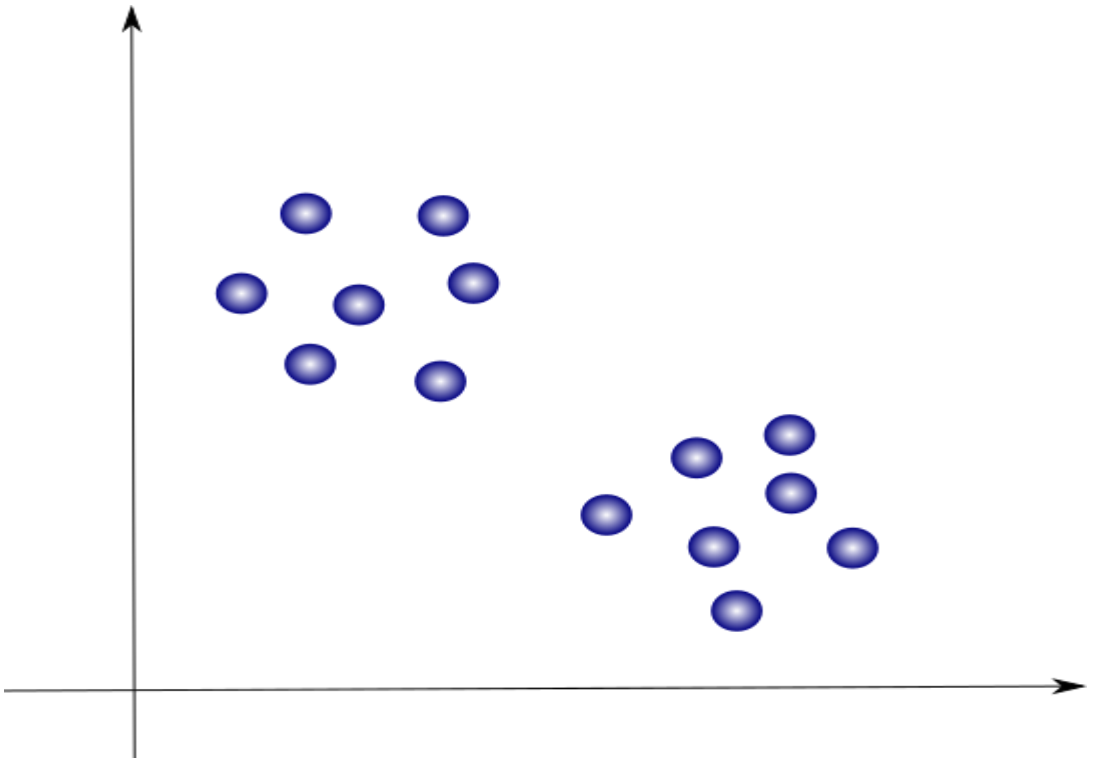
**Gaussian Mixture Model**

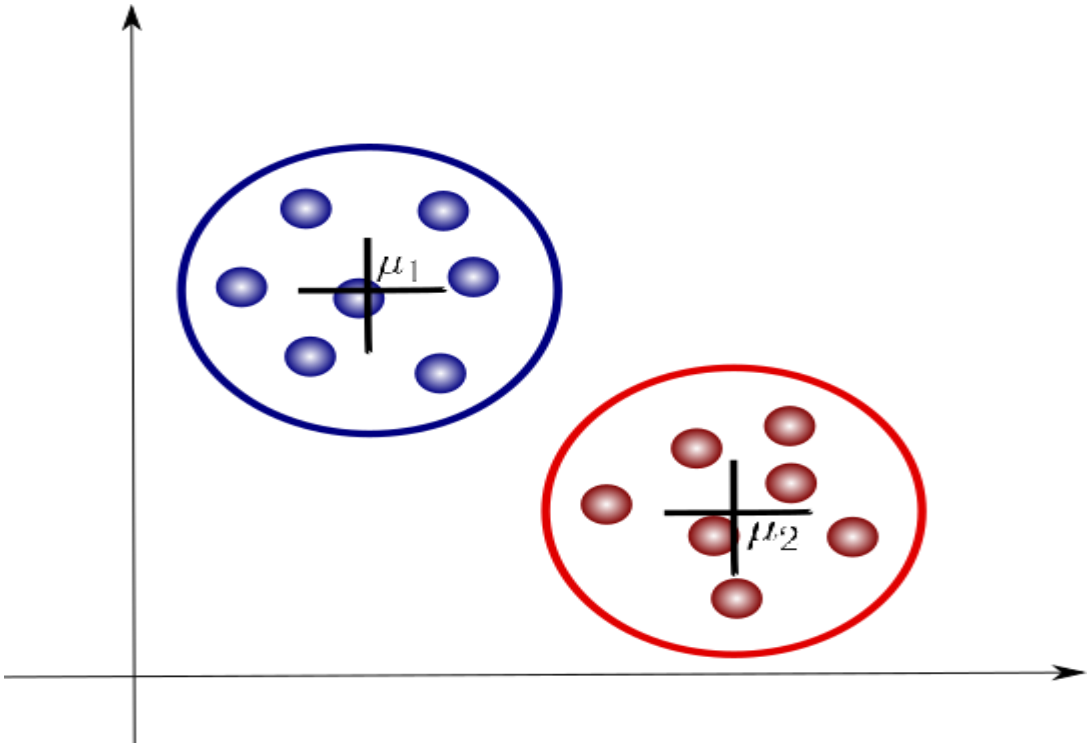
Supervised and unsupervised learning. The main difference between both lies in the nature of the data as well as the approaches used to deal with it. Clustering is an unsupervised learning problem where we intend to find clusters of points in our dataset that share some common characteristics.

**Gaussian mixture models** are a probabilistic model for representing normally distributed subpopulations within an overall population. Mixture models in general don't require knowing which subpopulation a data point belongs to, allowing the model to learn the subpopulations automatically. Since subpopulation assignment is not known, this constitutes a form of unsupervised learning.

Let's suppose we have a dataset that looks like this:



Our job is to find sets of points that appear close together. In this case, we can clearly identify two clusters of points which we will colour blue and red, respectively:




Please note that we are now introducing some additional notation. Here, μ1 and μ2 are the centroids of each cluster and are parameters that identify each of these. A popular clustering algorithm is known as K-means, which will follow an iterative approach to update the parameters of each clusters. More specifically, what it will do is to compute the means (or centroids) of each cluster, and then calculate their distance to each of the data points. The latter are then labeled as part of the cluster that is identified by their closest centroid. This process is repeated until some convergence criterion is met, for example when we see no further changes in the cluster assignments.

One important characteristic of K-means is that it is a *hard clustering method*, which means that it will associate each point to one and only one cluster. A limitation to this approach is that there is no uncertainty measure or *probability* that tells us how much a data point is associated with a specific cluster. So what about using a soft clustering instead of a hard one? This is exactly what Gaussian Mixture Models, or simply GMMs, attempt to do.

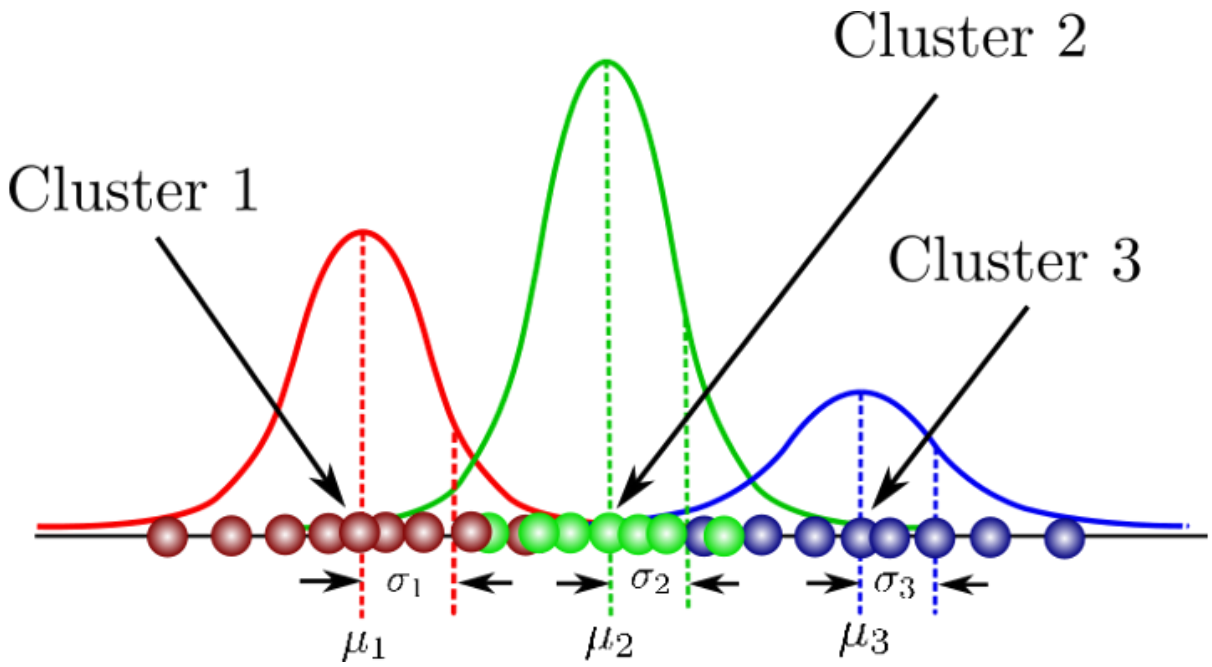
**Definitions**

A *Gaussian Mixture* is a function that is comprised of several Gaussians, each identified by $k \in \{1,\ldots, K\}$, where $K$ is the number of clusters of our dataset. Each Gaussian $k$ in the mixture is comprised of the following parameters:

- A mean μ that defines its centre.

- A covariance Σ that defines its width. This would be equivalent to the dimensions of an ellipsoid in a multivariate scenario.
- A mixing probability π that defines how big or small the Gaussian function will be.

Let us now illustrate these parameters graphically:




Here, we can see that there are three Gaussian functions, hence $K = 3$. Each Gaussian explains the data contained in each of the three clusters available. The mixing coefficients are themselves probabilities and must meet this condition:

$$\sum_{k=1}^{K} \pi_k = 1 \qquad (1)$$

Now how do we determine the optimal values for these parameters? To achieve this we must ensure that each Gaussian fits the data points belonging to each cluster. This is exactly what maximum likelihood does.

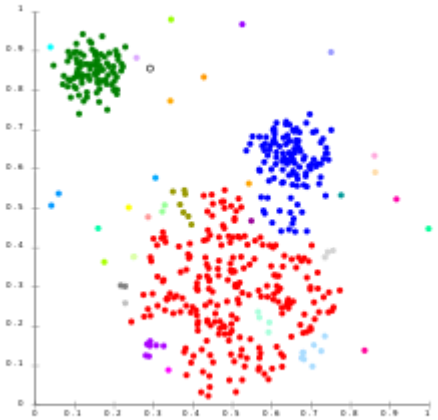
In general, the Gaussian density function is given by:

$$\mathcal{N}(\mathbf{x}|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\right)$$

Where **x** represents our data points, $D$ is the number of dimensions of each data point. μ and Σ are the mean and covariance, respectively. If we have a dataset comprised of $N = 1000$ three-dimensional points ($D = 3$), then **x** will be a $1000 \times 3$ matrix. μ will be a $1 \times 3$ vector, and Σ will be a $3 \times 3$ matrix. For later purposes, we will also find it useful to take the log of this equation, which is given by:

$$\ln \mathcal{N}(\mathbf{x}|\mu,\Sigma) = -\frac{D}{2}\ln 2\pi - \frac{1}{2}\ln \Sigma - \frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu) \qquad (2)$$

If we differentiate this equation with respect to the mean and covariance and then equate it to zero, then we will be able to find the optimal values for these parameters
Once the EM algorithm has run to completion, the fitted model can be used to perform various forms of inference. The two most common forms of inference done on GMMs are **density estimation** and **clustering**.



Clustering using a Gaussian mixture model. Each color represents a different cluster according to the model.[3]

**Density Estimation**

Since the GMM is completely determined by the parameters of its individual components, a fitted GMM can give an estimate of the probabilities of both in-sample and out-of-sample data points, known as density estimation. Furthermore, since numerically sampling from an individual Gaussian distribution is possible, one can easily sample from a GMM to create synthetic datasets.

Sampling from a GMM consists of the following steps:

1. Sample the Gaussian component according to the distribution defined by p(Cs)=ϕs.p(Cs)=ϕs.

2. Sample xx from the distribution for component CsCs, according to the distribution defined by N(x ∣ μs,σs).N(x∣μs,σs).

**Clustering**

Using Bayes' theorem and the estimated model parameters, one can also estimate the posteriori component assignment probability. Knowing that a data point is likely from one component distribution versus another provides a way to learn clusters, where cluster assignment is determined by the most likely component assignment. Clustering has many uses in machine learning, ranging from tissue differentiation in medical imaging to customer segmentation in market research.