Probabilistic Clustering

Mixture of Gaussians is a probabilistic technique to perform unsupervised clustering. It can be interpreted as a probabilistic version of K-Means clustering, as it works similar to the K-Means algorithm with a difference that this method assigns probabilities or so called 'responsibilities' to each data point that it has come from a particular cluster. Clustering using Mixture of Gaussians uses EM Algorithm at its heart, and the process is illustrated in the following steps:

(i) Means, Covariance matrices and Mixture coefficients are randomly initialized as per the number of components or groups (K) considered, for K number of gaussians.

(ii) For each data point, the posterior probabilities that the point has come from k^{th} gaussian given the point x, are calculated. This is called E -step of the EM algorithm.

$$\text{posterior} \propto \text{likelihood} \times \text{prior}$$

$$\gamma(znk) = \frac{\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)} \pi k$$

(iii) Using the posterior probabilities calculated in E-step, updated Means, Covariance matrices and mixture coefficients for all the gaussians corresponding to each cluster are calculated. This is called M-step of the EM algorithm.

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) \mathbf{x}_n$$

$$N_k = \sum_{n=1}^{N} \gamma(z_{nk}).$$

$$\boldsymbol{\Sigma}_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^{\mathrm{T}}$$

- Ref. : Pattern Recognition and Machine Learning, Bishop, 2006

Verification of Algorithm in Python

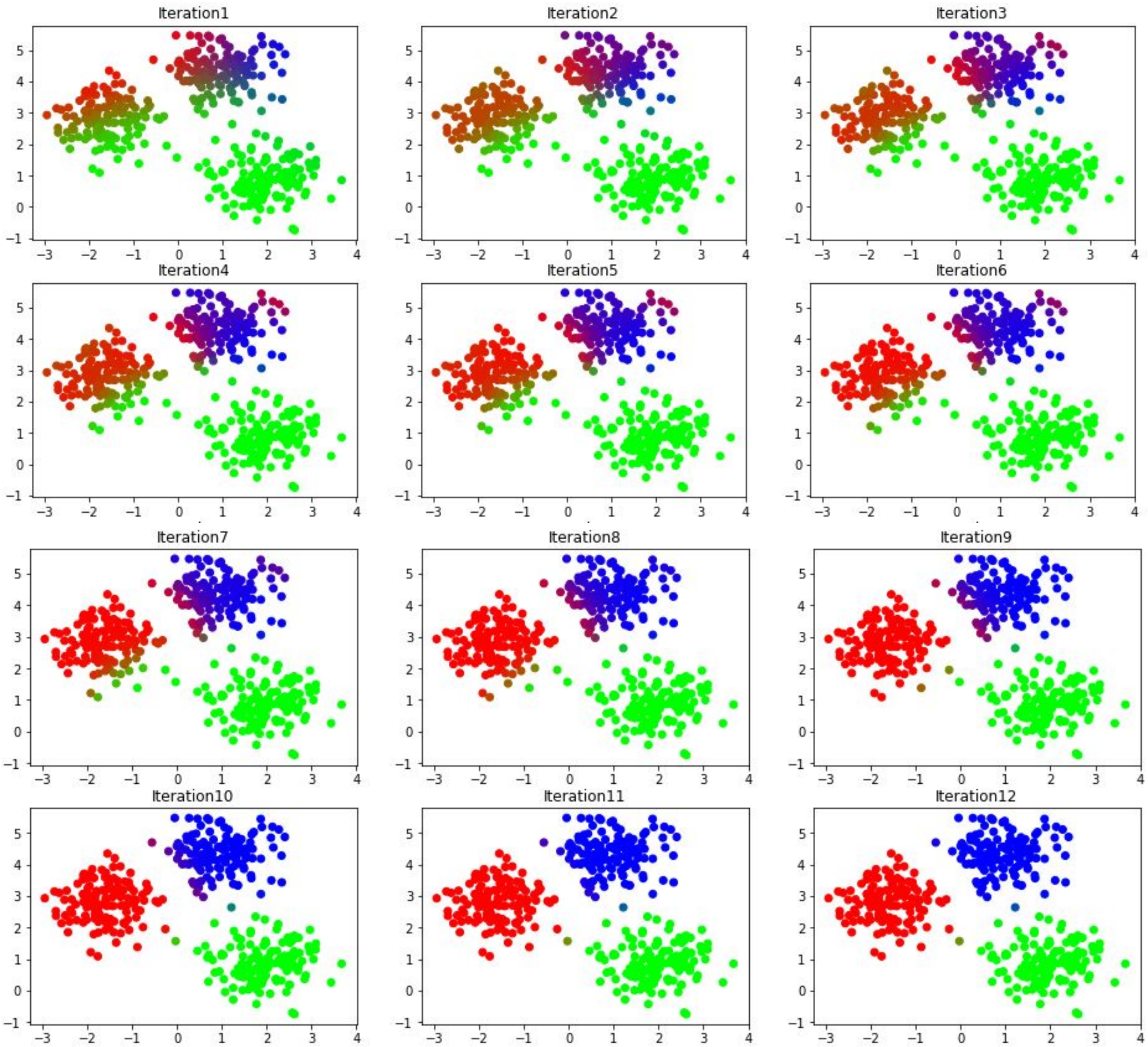

Fig. *Scatter Plots illustrating sequential workflow of EM algorithm for mixture of Gaussians for K=3*

Application of Algorithm for different values of K