MM Implementation

June 28, 2017

Theory

A mixture model will find the $\{(f_k, \xi_k)\}\, k = (1, ..., K)$ set of K classes defined by their multi-dimensional probability density function $f_k(X, \Theta_k)$where $\Theta_k$denotes a set of parameters of the kernel k and X the vector of variates and $\xi_k = (w_1, ..., w_k)$ a well known or estimation of their relative weight. K and other parameters could be determined empirically with other various techniques not detailed here. If probability distributions follow normal distributions, and if X is taken from a sample of N vectors, then we can say $f_k(X, \Theta_k) \approx \mathcal{N}(\mu_k, \Sigma_k)$ the multivariate normal distribution where p is the number of variates taken into account, $\mu_k$and Σ_k the respective mean and positive semi definite covariance matrix of the kernel k, previsously estimed or well known.

Expectation-Maximization Algorithm

An iteration of the algorithm first evaluates the different probabilities p_n for each sample X_n by evaluating the different density functions of each kernel f_k multiplied by their currently estimated weight w_k.

$$0 \leq n < N, p_n = \sum_{k=0}^{K} w_k.f_k(X_n)$$

The next step is to compute the relative probabilities:

$$p_{k,n} = \frac{w_k.f_k(X_n)}{p_n}$$

and adjust the weights:

$$w_k = \frac{\sum_{n=0}^{N} p_{k,n}}{N}$$

and update the parameters of each single distribution:

E.g. to update the parameters of a normal distribution whose pdf is:

$$f_k(X) = \frac{1}{\sqrt{(2\pi)^k \left|\Sigma\right|}}.e^{-\frac{1}{2}(X-\mu)^T\Sigma^{-1}(X-\mu)}$$

then maximization is done this way:

$$\mu_{k,i} := \frac{\sum_{n=0}^{N} p_{k,n} X_{n,i}}{N.w_k}$$

and:

$$\Sigma_{k,i,j} := \frac{\sum_{n=0}^{N} p_{k,n}.(X_{n,i} - \mu_i)(X_{n,j} - \mu_j)}{N.w_k}$$