Engineering and Applied Science Programs for Professionals
Whiting School of Engineering
Johns Hopkins University
685.621 Algorithms for Data Science
Computational Statistics

This document provides a rollup of the Computational Statistics specifically in the analysis of the algorithm associated with the Expectation Maximization algorithm. In this module the Expectation Maximization methods is introduced along with the associated mixture models. This methods however is not a classifier on its own, a Bayes decision theory method is use to make a two-class classifiers.

Contents

1 Computational Statistics

Computational statistics is a branch of mathematical sciences concerned with efficient methods for obtaining numerical solutions to statistically formulated problems. A complete course is dedicated for the study of a variety of computationally intensive statistical techniques and the role of computation as a tool of discovery. Topics include **numerical optimization in statistical inference [expectation-maximization (EM) algorithm, Fisher scoring, etc.], random number generation**, Monte Carlo methods, randomization methods, jackknife methods, bootstrap methods, tools for identification of structure in data, estimation of functions (orthogonal polynomials, splines, etc.), and graphical methods. The Fisher scoring was previously introduced as an application to feature ranking of two and multi class problems. The specific topic covered in this module is the Expectation Maximization method which is used to identify a maximum likelihood or maximum a posteriori estimate of an observation belonging to 1 of k clusters set by the user.

1.1 K-means Clustering

1.2 Expectation Maximization

The idea behind the EM algorithm (Dempster et al., 1977) is that even though the data values of $\mathbf{x}$, feature vectors in $\mathbf{x}_n \in \Re^D$, are unknown/incomplete the distribution $f(\mathbf{x}|p)$ can be used to determine an estimate for the maximum likelihood of class label association (Tomasi, 2006). In maximum likelihood estimation, the estimate to be modeled is the parameter(s) for which the observed data are the most likely. This is done by iteratively estimating the data parameters, then using the data to update the estimated parameters (mean, standard deviation and mixing probabilities), until a desired convergence is met. The two major steps of the EM algorithm are the expectation step (E-Step) and the maximization step (M-Step).

The EM algorithm consists of choosing initial parameters for the means, $\mu_k^{(j)}$, standard deviations, $\sigma_k^{(j)}$, and mixing probabilities, $p_k^{(j)}$, for a user defined number of clusters, K, then performing the E-Step and M-Step successively until convergence, where j is the current iteration and n is the number of samples/observations. The convergence criteria is determined by examining when the parameters quit changing, i.e., when $\left|\mu_k^{(j)} - \mu_k^{(j+1)}\right| < \epsilon$ & $\left|\sigma_k^{(j)} - \sigma_k^{(j+1)}\right| < \epsilon$ & $\left|p^{(j)}(k|n) - p^{(j+1)}(k|n)\right| < \epsilon$ for some epsilon (ϵ) and distance calculation (Euclidian distance). The maximum likelihood estimation is a method of estimating the parameters of the distributions based upon the observed data.

The expectation step (E-Step) calculates the membership probabilities, $p(k|n)$ (Tomasi, 2006). The mixing probabilities p_k are viewed as the sample mean of the membership probabilities $p(k|n)$ assuming a uniform distribution over all the data points. The Gaussian function, $g(\mathbf{x}_n; \mu_k^{(j)}, \sigma_k^{(j)})$, is used to compute mixture of Gaussian functions as shown in the denominator of $p(k|n)$.

$$p^{(j)}(k|n) = \frac{p_k^{(j)} g(\mathbf{x}_n; \mu_k^{(j)}, \sigma_k^{(j)})}{\sum\limits_{k=1}^{K} p_k^{(j)} g(\mathbf{x}_n; \mu_k^{(j)}, \sigma_k^{(j)})} \tag{1}$$

$$g(\mathbf{x}_n; \mu_k^{(j)}, \sigma_k^{(j)}) = \frac{1}{\left(\sqrt{2\pi}\sigma_k\right)^D} \exp\left\{ -\frac{1}{2}\left(\frac{\|\mathbf{x}_n - \mu_k\|}{\sigma_k}\right)^2 \right\} \tag{2}$$

Note that $\|\mathbf{x}_n - \mu_k\|$ is the vector norm in which the distance between the observation vector and the cluster mean vector is calculated. To account for the individual values of the vectors $\mathbf{x}_n$ and μ_k the vectors are temporarily written as $\mathbf{x}_{n_d}$ and μ_{k_d} in the following equation to account for the dimension of the vectors:

$$\|\mathbf{x}_{n_d} - \mu_{k_d}\| = \sqrt{\sum_{d=1}^{D} (\mathbf{x}_{n_d} - \mu_{k_d})^2} = \sqrt{(\mathbf{x}_{n_1} - \mu_{k_1})^2 + (\mathbf{x}_{n_2} - \mu_{k_2})^2 + \cdots + (\mathbf{x}_{n_D} - \mu_{k_D})^2} \tag{3}$$

The maximization step (M-Step) uses the data from the expectation step as if it were measured data to determine the maximum likelihood estimate of the parameter (Tomasi, 2006). This estimated data is often referred to as the "imputed" data. This step is dependent upon the membership probabilities $p(k|n)$ which are computed in the E-Step. The EM algorithm consists of iterating the mean, standard deviation, and mixing probabilities until convergence. The mixing probabilities are the sample mean of the conditional probabilities $p(k|n)$ assuming a uniform distribution over all the data points.

$$\mu_k^{(j+1)} = \frac{\sum\limits_{n=1}^{N} p^j(k|n)\mathbf{x}_n}{\sum\limits_{n=1}^{N} p^j(k|n)} \tag{4}$$

$$\sigma_k^{(j+1)} = \sqrt{\frac{1}{D}\frac{\sum\limits_{n=1}^{N} p^j(k|n)\left\|\mathbf{x}_n - \mu_k^{(j+1)}\right\|^2}{\sum\limits_{n=1}^{N} p^j(k|n)}} \tag{5}$$

$$p_k^{(j+1)} = \frac{1}{N}\sum_{n=1}^{N} p^j(k|n) \tag{6}$$

1.2.1 Example

Given a matrix and generated mean and standard deviation the following variables are assigned:

$$\mathbf{x} = \begin{bmatrix} 1 & 2 \\ 4 & 2 \\ 1 & 3 \\ 4 & 3 \end{bmatrix}, \bar{x}_{col} = \begin{bmatrix} 2.5 & 2.5 \end{bmatrix}^T, \sigma_{col} = \begin{bmatrix} 1.7321 & 0.57735 \end{bmatrix}^T$$

The variable $K \equiv$ number of sub populations. This is a priori knowledge the use will need to know or guess. Now a set of initial parameters are generated to ensure convergence is reached for each subpopulation. Notice that the column standard deviation is multiplied by random numbers from -1 to 1 which is used to calculate the initial mean for the subpopulations. For the standard deviation the average is used for simplicity. The membership probabilities are derived by the number of desired sub populations.

$$\mu = \bar{x}_{col}\begin{bmatrix} 1 & 1 \end{bmatrix} + \sigma_{col}\begin{bmatrix} randn(1,K) \end{bmatrix} = \begin{bmatrix} 2.5 \\ 2.5 \end{bmatrix}\begin{bmatrix} 1 & 1 \end{bmatrix} + \begin{bmatrix} 1.7321 \\ 0.57735 \end{bmatrix}\begin{bmatrix} -0.1867 & 0.7257 \end{bmatrix} = \begin{bmatrix} 2.1766 & 3.7571 \\ 2.3922 & 2.9190 \end{bmatrix}$$

$$\sigma = \bar{\sigma}_{col}\begin{bmatrix} 1 & 1 \end{bmatrix} = \begin{bmatrix} 1.1547 & 1.1547 \end{bmatrix}\begin{bmatrix} 1 & 1 \end{bmatrix} = \begin{bmatrix} 1.1547 & 1.1547 \end{bmatrix}$$

$$p_k = \frac{\begin{bmatrix} 1 & 1 \end{bmatrix}}{K} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

The Expectation Step (E-Step) $p^{(j)}(k|n)$ is the conditioning argument (n) to the expectation (k) and is regarded as fixed. Computes the expected value of the x_n data using the current estimation of the parameter and the observed data. This step uses the prior probabilities of the sub population centroids, $p_k^{(j)}$. The Gaussian function, $g()$, is used to compute mixture of Gaussian functions as shown in the denominator of $p^{(j)}(k|l)$.

$$p^{(j)}(k|n) = \frac{p_k^{(j)} g(\mathbf{x}_n; \mu_k^{(j)}, \sigma_k^{(j)})}{\sum\limits_{k=1}^{K} p_k^{(j)} g(\mathbf{x}_n; \mu_k^{(j)}, \sigma_k^{(j)})} \tag{7}$$

Now we rewrite Equation 8 for the individual samples/observations as follows:

$$g(\mathbf{x}_n;\mu_k^{(j)},\sigma_k^{(j)}) = \frac{1}{\left(\sqrt{2\pi}\sigma_k\right)^D}\exp\left\{-\frac{1}{2}\left(\frac{\|\mathbf{x}_n-\mu_k\|}{\sigma_k}\right)^2\right\} \tag{8}$$

Now lets add some numbers to the equations based on the give matrix $\mathbf{x}$ and initial conditions set.
For $n = k = j = 1$,

$$g(x_1;\mu_1^{(1)},\sigma_1^{(1)}) = \frac{1}{(\sqrt{2\pi}1.1547)^2}\exp\left\{-\frac{1}{2}\left(\frac{\|\begin{bmatrix}1 & 2\end{bmatrix}-\begin{bmatrix}2.1766 & 2.3922\end{bmatrix}\|}{1.1547}\right)^2\right\} = \frac{1}{8.3776}\exp\left\{-\frac{1}{2}\left(\frac{1.5382}{1.3333}\right)\right\} = 0.0670$$

$$p_i^{(1)}g(x_1;\mu_1^{(1)},\sigma_1^{(1)}) = (0.5)(0.0670) = 0.0335$$

For $n = 2, k = j = 1$,

$$g(x_2;\mu_1^{(1)},\sigma_1^{(1)}) = \frac{1}{(\sqrt{2\pi}1.1547)^2}\exp\left\{-\frac{1}{2}\left(\frac{\|\begin{bmatrix}4 & 2\end{bmatrix}-\begin{bmatrix}2.1766 & 2.3922\end{bmatrix}\|}{1.1547}\right)^2\right\} = \frac{1}{8.3776}\exp\left\{-\frac{1}{2}\left(\frac{3.4786}{1.3333}\right)\right\} = 0.0324$$

$$p_i^{(1)}g(x_2;\mu_1^{(1)},\sigma_1^{(1)}) = (0.5)(0.0324) = 0.0162$$

For $n = 3, k = j = 1$,

$$g(x_3;\mu_1^{(1)},\sigma_1^{(1)}) = \frac{1}{(\sqrt{2\pi}1.1547)^2}\exp\left\{-\frac{1}{2}\left(\frac{\|\begin{bmatrix}1 & 3\end{bmatrix}-\begin{bmatrix}2.1766 & 2.3922\end{bmatrix}\|}{1.1547}\right)^2\right\} = \frac{1}{8.3776}\exp\left\{-\frac{1}{2}\left(\frac{1.7538}{1.3333}\right)\right\} = 0.0618$$

$$p_i^{(1)}g(x_3;\mu_1^{(1)},\sigma_1^{(1)}) = (0.5)(0.0618) = 0.0309$$

For $n = 4, k = j = 1$,

$$g(x_4;\mu_1^{(1)},\sigma_1^{(1)}) = \frac{1}{(\sqrt{2\pi}1.1547)^2}\exp\left\{-\frac{1}{2}\left(\frac{\|\begin{bmatrix}4 & 3\end{bmatrix}-\begin{bmatrix}2.1766 & 2.3922\end{bmatrix}\|}{1.1547}\right)^2\right\} = \frac{1}{8.3776}\exp\left\{-\frac{1}{2}\left(\frac{3.6942}{1.3333}\right)\right\} = 0.0299$$

$$p_i^{(1)}g(x_4;\mu_1^{(1)},\sigma_1^{(1)}) = (0.5)(0.0299) = 0.0149$$

For $n = 1, k = 2, j = 1$,

$$g(x_1;\mu_2^{(1)},\sigma_2^{(1)}) = \frac{1}{(\sqrt{2\pi}1.1547)^2}\exp\left\{-\frac{1}{2}\left(\frac{\|\begin{bmatrix}1 & 2\end{bmatrix}-\begin{bmatrix}3.7571 & 2.9190\end{bmatrix}\|}{1.1547}\right)^2\right\} = \frac{1}{8.3776}\exp\left\{-\frac{1}{2}\left(\frac{8.4463}{1.3333}\right)\right\} = 0.0050$$

$$p_i^{(1)}g(x_1;\mu_2^{(1)},\sigma_2^{(1)}) = (0.5)(0.0050) = 0.0025$$

For $n = 2, k = 2, j = 1$,

$$g(x_2;\mu_2^{(1)},\sigma_2^{(1)}) = \frac{1}{(\sqrt{2\pi}1.1547)^2}\exp\left\{-\frac{1}{2}\left(\frac{\|\begin{bmatrix}4 & 2\end{bmatrix}-\begin{bmatrix}3.7571 & 2.9190\end{bmatrix}\|}{1.1547}\right)^2\right\} = \frac{1}{8.3776}\exp\left\{-\frac{1}{2}\left(\frac{0.9036}{1.3333}\right)\right\} = 0.0851$$

$$p_i^{(1)}g(x_2;\mu_2^{(1)},\sigma_2^{(1)}) = (0.5)(0.0851) = 0.0425$$

For $n = 3, k = 2, j = 1$,

$$g(x_3;\mu_2^{(1)},\sigma_2^{(1)}) = \frac{1}{(\sqrt{2\pi}1.1547)^2}\exp\left\{-\frac{1}{2}\left(\frac{\|\begin{bmatrix}1 & 3\end{bmatrix}-\begin{bmatrix}3.7571 & 2.9190\end{bmatrix}\|}{1.1547}\right)^2\right\} = \frac{1}{8.3776}\exp\left\{-\frac{1}{2}\left(\frac{7.6082}{1.3333}\right)\right\} = 0.0069$$

$$p_i^{(1)}g(x_3;\mu_2^{(1)},\sigma_2^{(1)}) = (0.5)(0.0069) = 0.0034$$

For $n = 4, k = 2, j = 1$,

$g(x_4; \mu_2^{(1)}, \sigma_2^{(1)}) = \frac{1}{(\sqrt{2\pi}1.1547)^2} \exp\left\{ -\frac{1}{2}\left(\frac{\| \begin{bmatrix} 4 & 3 \end{bmatrix} - \begin{bmatrix} 3.7571 & 2.9190 \end{bmatrix} \|}{1.1547}\right)^2 \right\} = \frac{1}{8.3776} \exp\left\{ -\frac{1}{2}\left(\frac{0.0656}{1.3333}\right) \right\} = 0.1165$

$p_i^{(1)} g(x_4; \mu_2^{(1)}, \sigma_2^{(1)}) = (0.5)(0.1165) = 0.0582$

Now we calculate the membership probabilities of $p^{(1)}(k|n)$ as follows:

$p^{(1)}(k|n) = \frac{p_k^{(1)} g(\mathbf{x}; \mu_k^{(1)}, \sigma_k^{(1)})}{\sum_{k=1}^{K} p_k^{(1)} g(\mathbf{x}; \mu_k^{(1)}, \sigma_k^{(1)})}$

For $n = 1, k = 1, j = 1$,

$p^{(1)}(1|1) = \frac{p_1^{(1)} g(x_1; \mu_1^{(1)}, \sigma_1^{(1)})}{\sum_{k=1}^{K} p_k^{(1)} g(x_1; \mu_k^{(1)}, \sigma_k^{(1)})} = \frac{0.0335}{0.0360} = 0.9302$

For $n = 2, k = 1, j = 1$,

$p^{(1)}(1|2) = \frac{p_1^{(1)} g(x_2; \mu_1^{(1)}, \sigma_1^{(1)})}{\sum_{k=1}^{K} p_k^{(1)} g(x_2; \mu_k^{(1)}, \sigma_k^{(1)})} = \frac{0.0162}{0.0587} = 0.2758$

For $n = 3, k = 1, j = 1$,

$p^{(1)}(1|3) = \frac{p_1^{(1)} g(x_3; \mu_1^{(1)}, \sigma_1^{(1)})}{\sum_{k=1}^{K} p_k^{(1)} g(x_3; \mu_k^{(1)}, \sigma_k^{(1)})} = \frac{0.0309}{0.0344} = 0.8998$

For $n = 4, k = 1, j = 1$,

$p^{(1)}(1|4) = \frac{p_1^{(1)} g(x_4; \mu_1^{(1)}, \sigma_1^{(1)})}{\sum_{k=1}^{K} p_k^{(1)} g(x_4; \mu_k^{(1)}, \sigma_k^{(1)})} = \frac{0.0149}{0.0732} = 0.2041$

For $n = 1, k = 2, j = 1$,

$p^{(1)}(2|1) = \frac{p_2^{(1)} g(x_1; \mu_2^{(1)}, \sigma_2^{(1)})}{\sum_{k=1}^{K} p_k^{(1)} g(x_1; \mu_k^{(1)}, \sigma_k^{(1)})} = \frac{0.0025}{0.0360} = 0.0693$

For $n = 2, k = 2, j = 1$,

$p^{(1)}(2|2) = \frac{p_2^{(1)} g(x_2; \mu_2^{(1)}, \sigma_2^{(1)})}{\sum_{k=1}^{K} p_k^{(1)} g(x_2; \mu_k^{(1)}, \sigma_k^{(1)})} = \frac{0.0425}{0.0587} = 0.7242$

For $n = 3, k = 2, j = 1$,

$p^{(1)}(2|3) = \frac{p_2^{(1)} g(x_3; \mu_2^{(1)}, \sigma_2^{(1)})}{\sum_{k=1}^{K} p_k^{(1)} g(x_3; \mu_k^{(1)}, \sigma_k^{(1)})} = \frac{0.0034}{0.0344} = 0.1002$

For $n = 4, k = 2, j = 1$,

$p^{(1)}(2|4) = \frac{p_2^{(1)} g(x_4; \mu_2^{(1)}, \sigma_2^{(1)})}{\sum_{k=1}^{K} p_k^{(1)} g(x_4; \mu_k^{(1)}, \sigma_k^{(1)})} = \frac{0.0582}{0.0732} = 0.7959$

Now we can calculate the following sum of mixture probabilities for each k as follows:

For $N = 4, k = 2$ and $j = 1$,

$$\sum_{n=1}^{4} p^1(1|n) = 0.9302 + 0.2758 + 0.8998 + 0.2041 = 2.3100 \qquad (9)$$

For $N = 4, k = 2$ and $j = 1$,

$$\sum_{n=1}^{4} p^1(2|n) = 0.0698 + 0.7242 + 0.1002 + 0.7959 = 1.6900 \tag{10}$$

Now we can calculate the M Step values as follows:

For $N = 4, k = 1$ and $j = 1$

$$\mu_1^{(2)} = \frac{\sum\limits_{n=1}^{4} p^1(1|n)\mathbf{x}_n}{\sum\limits_{n=1}^{4} p^1(1|n)} = \frac{0.9302\begin{bmatrix}1 & 2\end{bmatrix} + 0.2758\begin{bmatrix}4 & 2\end{bmatrix} + 0.8998\begin{bmatrix}1 & 3\end{bmatrix} + 0.2041\begin{bmatrix}4 & 3\end{bmatrix}}{0.9302 + 0.2758 + 0.8998 + 0.2041} = \begin{bmatrix}1.6232 & 2.4779\end{bmatrix} \tag{11}$$

$$\sigma_1^{(2)} = \sqrt{\frac{1}{2}\frac{\sum\limits_{n=1}^{4} p^1(1|n)\left\|\mathbf{x}_n - \mu_1^{(2)}\right\|^2}{\sum\limits_{n=1}^{4} p^1(1|n)}} = \sqrt{\frac{1}{2}\frac{0.9302(0.6168) + 0.2758(5.8774) + 0.8998(0.6610) + 0.2041(5.9216)}{0.9302 + 0.2758 + 0.8998 + 0.2041}} = 0.9303 \tag{12}$$

$$p_1^{(2)} = \frac{1}{4}\sum_{n=1}^{4} p^1(1|n) = \frac{0.9302 + 0.2758 + 0.8998 + 0.2041}{4} = 0.5775 \tag{13}$$

For $N = 4, k = 2$ and $j = 1$

$$\mu_2^{(2)} = \frac{\sum\limits_{n=1}^{4} p^1(2|n)\mathbf{x}_n}{\sum\limits_{n=1}^{4} p^1(2|n)} = \frac{0.0698\begin{bmatrix}1 & 2\end{bmatrix} + 0.7242\begin{bmatrix}4 & 2\end{bmatrix} + 0.1002\begin{bmatrix}1 & 3\end{bmatrix} + 0.7959\begin{bmatrix}4 & 3\end{bmatrix}}{0.0698 + 0.7242 + 0.1002 + 0.7959} = \begin{bmatrix}3.6984 & 2.5302\end{bmatrix} \tag{14}$$

$$\sigma_2^{(2)} = \sqrt{\frac{1}{2}\frac{\sum\limits_{n=1}^{4} p^1(2|n)\left\|\mathbf{x}_n - \mu_2^{(2)}\right\|^2}{\sum\limits_{n=1}^{4} p^1(2|n)}} = \sqrt{\frac{1}{2}\frac{0.0698(7.5623) + 0.7242(0.3721) + 0.1002(7.5020) + 0.7959(0.3117)}{0.0698 + 0.7242 + 0.1002 + 0.7959}} = 0.7290 \tag{15}$$

$$p_1^{(2)} = \frac{1}{4}\sum_{n=1}^{4} p^1(2|n) = \frac{0.0698 + 0.7242 + 0.1002 + 0.7959}{4} = 04225 \tag{16}$$

Now the convergence criteria can be determined using $\left|\mu_k^{(j)} - \mu_k^{(j+1)}\right| < \epsilon$ & $\left|\sigma_k^{(j)} - \sigma_k^{(j+1)}\right| < \epsilon$ & $\left|p^{(j)}(k|n) - p^{(j+1)}(k|n)\right| < \epsilon$ for some epsilon (ϵ) and distance calculation (Euclidian distance).

1.2.2 Mixture Models

In mixture models, also known as model-based Gaussian clustering, the multivariate Gaussian normal is used as a density function similarly described in Equation 8. The general multivariate normal density for n dimensions is

$$g(\mathbf{x};\mu_k,\Sigma_k) = \frac{\exp\Big\{-\frac{1}{2}(\mathbf{x}-\mu_k)^T\Sigma_k^{-1}(\mathbf{x}-\mu_k)\Big\}}{\Big(\sqrt{2\pi}\Big)^n\Big|\Sigma_k\Big|^{1/2}}. \tag{17}$$

The geometric characteristics (size, shape and orientation) of the clusters are determined by the covariance matrix Σ_k which is generated in terms of eigenvalue decomposition described in Martinez and Martinez (2002). The decomposition of the covariance matrix Σ_k is used as a suitable model for the geometric characteristics of the cluster. The structure of the covariance matrix is as follows:

$$\Sigma_k = \lambda_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k^T \tag{18}$$

where λ_k is a scalar, D_k is the orthogonal matrix of eigenvectors and A_k is a diagonal matrix whose elements are proportional to the eigenvalues of Σ_k. Note that in EM the values p_k, μ_k, and σ_k are updated after each iteration and in the mixture models σ_k is replaced by Σ_k to represent the geometric characteristics of the clusters.

The eigenvalue decomposition can be modeled as various clustering arrangements. Celeux and Govaert (1995), describe in detail fourteen models based on the eigenvalue decomposition. Allowing for variations in the orientation, volume, shape and size of the clusters; six of these models are shown in Table 1 (Martinez and Martinez, 2002).

Table 1: Parameterization for Mixture Models

Model	Σ_k	**Geometric Shapel**	**Volume**	**Shape**	**Orientation**
1	$\lambda\mathbf{I}$	Spherical	Equal	Equal	N/A
2	$\lambda_k\mathbf{I}$	Spherical	Variable	Equal	N/A
3	$\lambda\mathbf{DAD}^T$	Ellipsoid	Equal	Equal	Equal
4	$\lambda_k\mathbf{D}_k\mathbf{A}_k\mathbf{D}_k^T$	Ellipsoid	Variable	Variable	Variable
5	$\lambda\mathbf{D}_k\mathbf{A}\mathbf{D}_k^T$	Ellipsoid	Equal	Equal	Variable
6	$\lambda_k\mathbf{D}_k\mathbf{A}\mathbf{D}_k^T$	Ellipsoid	Variable	Equal	Variable

The eigenvalue decomposition can be modeled as various clustering arrangements, i.e., spheres, ellipsoids and rotations of ellipsoids. Allowing the orientation, volume, shape and size of the clusters define the various models used. Figure 1 shows the mixture model using rotated ellipsoids (Model 4) to generate the decision boundary around each class.

1.2.3 Bayes Classifier

The EM algorithm can be used to find a class label for an input sample. Classification uses input samples described by feature vectors $\mathbf{x}_0 \in \Upsilon_n$ to assign the samples to a given class $\mathbf{C} = C_j = [C_1, C_2, \cdots, C_c], j = 1, 2, \cdots, c$. The Bayes classifier extends a general multivariate normal case where the covariance matrix Σ_j for each class is different. For the multi-class classifier each class must have individual conditional probability densities where the densities are modeled as normal distributions. The classes C_j are defined as normal distributions centered about the mean vector μ_j. The mean vector, μ_j, and the covariance matrix, Σ_j, are calculated using the EM algorithm. The vector $\mathbf{x}_0$ is a n-dimensional vector of the observed data, and $|\Sigma_i|$ and Σ_i^{-1} are the determinants and inverse covariance matrix of the given class. The posterior probability of class membership can be calculated by Bayes rule if C_j is defined as the event of belonging to population j. Using the density function $g(\mathbf{x}; \mu_k^{(i)}, \sigma_k^{(i)})$ (Tomasi, 2006), the Bayes classifier can be expressed in terms of the prior probabilities, $P(C_i)$, and posterior probability of class membership as follows:

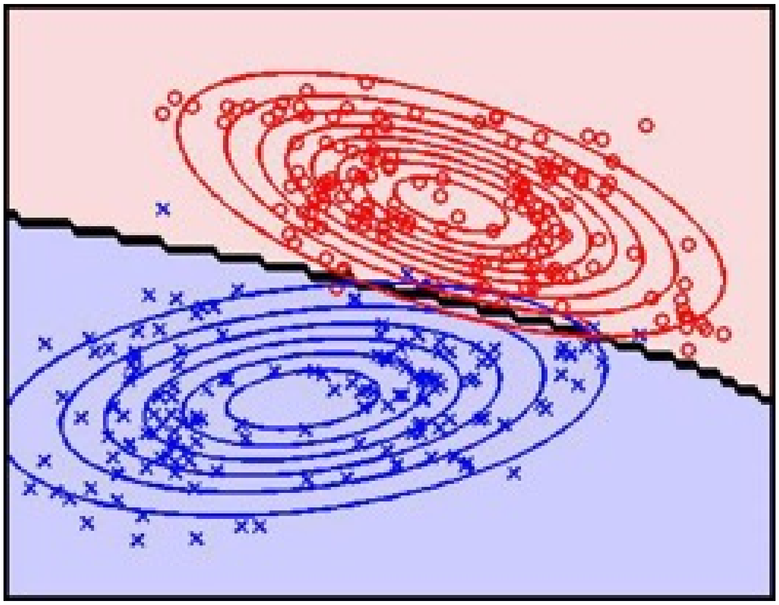

Figure 1: Expectation Maximization using mixture models with Decision Boundary

$$P(C_j|x_0) = \frac{P(C_j)\frac{1}{\sqrt{(2\pi)^n|\Sigma_j|}}\exp\Big[-\frac{1}{2}\big(\mathbf{x}_0-\mu_j\big)^T\Sigma_j^{-1}\big(\mathbf{x}_0-\mu_j\big)\Big]}{\sum_{i=1}^{c}P(C_i)\frac{1}{\sqrt{(2\pi)^n|\Sigma_i|}}\exp\Big[-\frac{1}{2}\big(\mathbf{x}_0-\mu_i\big)^T\Sigma_i^{-1}\big(\mathbf{x}_0-\mu_i\big)\Big]} \tag{19}$$

where the a priori probabilities P(Cj) are the estimates of belonging to a class and under the assumption that $\Sigma_j = \Sigma$ for $\forall j$.

2 References

[1] Bishop, Christopher M., *Neural Networks for pattern Recognition*, Oxford University Press, 1995

[2] Bishop, Christopher M., *Pattern Recognition and Machine Learning*, Springer, 2006

[3] Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronal L., and Stein, Clifford, *Introduction to Algorithms*, 3rd Edition, MIT Press, 2009

[4] Dempster, A. P., Laird, N. M. and Rubin, D. B., *Maximum likelihood from incomplete data via the EM algorithm*, Journal of the Royal Statistical Society B, Volume 39, Number 1, pp.1–22, 1977

[5] Duda, R.O., Hart, P.E. and Stork, D.G., *Pattern Classification* (2nd. Ed.), New York, NY: John Wiley & Sons, 2001

[6] Duin, Robert P.W., Tax, David and Pekalska, Elzbieta, *PRTools*, http://prtools.tudelft.nl/

[7] Franc, Vojtech and Hlavac, Vaclav, *Statistical Pattern Recognition Toolbox*, https://cmp.felk.cvut.cz/cmp/software/stprtool/index.html

[8] Fukunaga, Keinosuke, *Introduction to Statistical Pattern Recognition*, Academic Press, 1972

[9] Jaakola, T. S. and Haussler, D., *Exploring Generative Models in Discriminative Classifiers*, Advances in Neural Information Processing Systems, Kearns, M.S., Soll, S. A. and Cohn, D. A. (Eds.), Volume 11, Cambridge, MA: MIT Press, 1998

[10] Machine Learning at Waikato University, *WEKA*, https://www.cs.waikato.ac.nz/ ml/index.html

[11] Martinez W. L. and Martinez, A. R., *Computational Statistics Handbook with MATLAB*, Boca Raton, FL: Chapman & Hall/CRC, 2002

[12] Parzen, E., *On the Estimation of a Probability Density Function and Mode*, Annals of Mathematical Statistics, Volume 33 pp. 1065-1076, 1962

[13] Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery, Brian P., *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press, Jan 31, 1986

[14] Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery, Brian P., *Numerical Recipes: The Art of Scientific Computing*, 3rd Edition, Cambridge University Press, September 10, 2007

[15] Press, William H., Teukolsky, Saul A., Vetterling, William T., and Flannery, Brian P., *Numerical Recipes: The Art of Scientific Computing*, 3rd Edition, http://numerical.recipes/

[16] Press, William H., *Opinionated Lessons in Statistics*, http://www.opinionatedlessons.org/

[17] Taboga, Marco, *EM algorithm*, Lectures on probability theory and mathematical statistics. Kindle Direct Publishing, https://www.statlect.com/fundamentals-of-statistics/EM-algorithm, 2021

[18] Tomasi, C., *Estimating Gaussian Mixture Densities with EM – A Tutorial*, Duke University Course Notes, 2006, http://www.cs.duke.edu/courses/spring04/cps196.1/handouts/EM/tomasiEM.pdf, Retrieved Sept 2006