Chapter 3

Multivariate Normal Model

3.1 Introduction

In this lecture we will examine the use of the multivariate normal model and its use in pattern classification. A point in feature space will be represented by a row vector $\mathbf{x}$ of M dimensions. $\mathbf{x} = x_1, \ldots, x_M$, where x_i is the value of feature number i. Particular points in feature space may be referred to as $\mathbf{x}_1$, $\mathbf{x}_2$ and so on. We will illustrate some of the concepts by using figures based on $M = 2$, but real problems often use much larger values of M. The mathematics is the same in higher dimensions, but we can't draw the figures.

The multivariate normal distribution has a relatively simple expression that is almost independent of the number of dimensions. This and other properties make it very flexible as a modeling tool. The expression is

$$p_N(\mathbf{x}) = \frac{1}{(2\pi)^{M/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}\left[(\mathbf{x}-\mu)\Sigma^{-1}(\mathbf{x}-\mu)'\right]} \tag{3.1}$$

We will use the notation $p_N(\mathbf{x}, \mu, \Sigma)$ to refer to the above expression when we want to make the parameters evident but don't want to write out the whole equation. The subscript N means that the distribution is normal.

The parameter μ is a vector that represents the mean value of $\mathbf{x}$. The value μ_r is the mean value of x_r. The parameter Σ is the covariance matrix. The ij element of Σ is the covariance

$$s_{ij} = E[(x_i - \mu_i)(x_j - \mu_j)] \tag{3.2}$$

Since $s_{ij} = s_{ji}$ and $s_{ii} > 0$, the matrix Σ is positive definite. All of the eigenvalues of Σ are real and positive and all of the eigenvectors are linearly independent. These properties are especially useful in certain numerical calculations.

The contours on which $p_N(\mathbf{x}) = c$, where c is some constant, are of particular interest. These are the contours of constant probability. If we set the expression in (3.1) equal to a constant, we see that the required condition for a contour is that the exponent equal a constant. This leads to

$$(\mathbf{x} - \mu)\Sigma^{-1}(\mathbf{x} - \mu)' = \ln\left(c(2\pi)^{M/2}\sqrt{|\Sigma|}\right) = c_1 \tag{3.3}$$

This defines an ellipsoid in M-dimensional space.

It can be shown that the term on the left satisfies the requirements to be a distance measure. This distance measure, which is commonly used in statistics, is called the Mahalanobis distance.

Mahalanobis Distance The Mahalanobis distance between two points x and y is

$$d_\Sigma(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})\Sigma^{-1}(\mathbf{x} - \mathbf{y})'} \tag{3.4}$$

where Σ is a positive defininte symmetric matrix.

Bivariate Normal Distribution

The bivariate normal distribution has $M = 2$. The mean is a vector of length $M = 2$ and the covariance is a 2×2 matrix. This is the simplest case and one that has figures that we can draw. We can write out the various vectors and matrices in symbolic form to see their behaviour. We will calculate estimated values for these symbols in examples where we have numerical feature data. Let the mean vector and covariance matrix be represented by

$$\mu = [m_1, m_2] \text{ and } \Sigma = \begin{bmatrix} s_{11} & s_{12} \\ s_{21} & s_{22} \end{bmatrix} \tag{3.5}$$

The inverse covariance matrix is

$$\Sigma^{-1} = \frac{\begin{bmatrix} s_{22} & -s_{12} \\ -s_{21} & s_{11} \end{bmatrix}}{s_{11}s_{22} - s_{12}s_{21}} \tag{3.6}$$

When we multiply out the vector products, we find that the probability density function is

$$p_X(\mathbf{x}) = \frac{1}{2\pi\sqrt{s_{11}s_{22} - s_{12}s_{21}}} \cdot \exp\left(-\frac{(x_1 - m_1)^2 s_{22} - (s_{12}+s_{21})(x_1 - m_1)(x_2 - m_2) + s_{11}(x_2 - m_2)^2}{2(s_{11}s_{22} - s_{12}s_{21})}\right)$$

The exponent is a quadratic form in the components of $\mathbf{x} = [x_1, x_2]$. Setting the exponent to a constant will provide contours of constant probability. The shape of these contours will be ellipses. They will be concentric about the mean value of the distribution. This will be illustrated with an example.

Numerical Example Consider the $M = 2$ case with mean and covariance given by

$$\mu = [1, 1] \text{ and } \Sigma = \begin{bmatrix} 1 & -0.7 \\ -0.7 & 1 \end{bmatrix} \tag{3.7}$$

We find that $|\Sigma| = 0.51$ and

$$\Sigma^{-1} = \frac{1}{0.51}\begin{bmatrix} 1 & 0.7 \\ 0.7 & 1 \end{bmatrix} = \begin{bmatrix} 1.9608 & 1.3725 \\ 1.3725 & 1.9608 \end{bmatrix} \tag{3.8}$$

A plot of the probability function and contours of constant probability is shown in Figure 3.1

Here the contours of constant probability are ellipses that are centered on the mean feature values. Each of the contours has an equation of the form

$$1.9608(x_1 - 1)^2 + 2.755(x_1 - 1)(x_2 - 1) + 1.9608(x_2 - 1)^2 = c_1 \tag{3.9}$$

where c_1 is an appropriate constant. This is the form of an equation of an ellipse. Note that the ellipses are concentric about the mean vector $\mu = [1, 1]$ and are tilted because of the nonzero terms off of the diagonal in the covariance matrix. The direction of the tilt of the ellipse axis is determined by the sign of the off-diagonal terms.

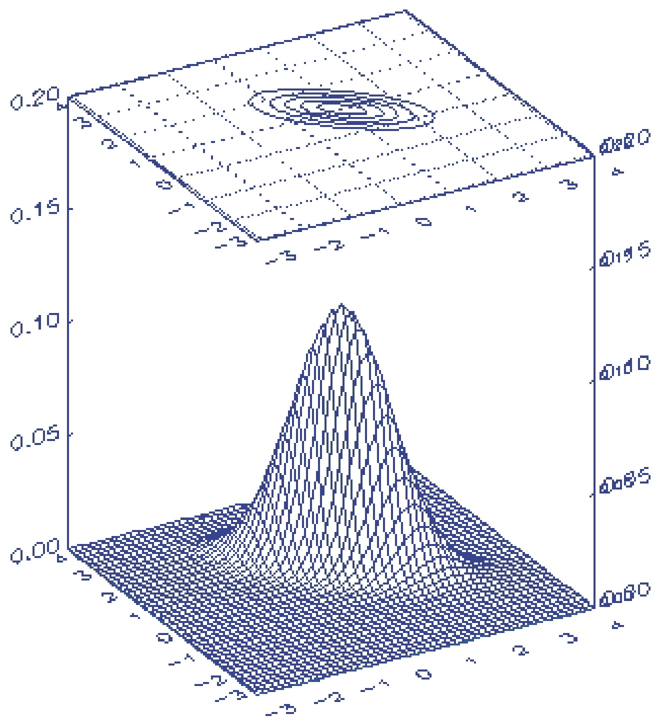

Figure 3.1: A bivariate normal distribution centered on (1,1) with negatively correlated features.

3.1.1 Modeling Experimental Data

The normal distribution may be used to model experimental data by computing the mean and covariance functions for that data. Of course, the model will only give reasonable results when the data is clustered in a normal fashion.

When the data is distributed in three or fewer dimensions it is possible to display the distribution of feature points and get an idea about the quality of the cluster for each class.

3.1.2 Example: Three Classes with Two Features

Some experimental data involving $M = 2$ features are displayed in Figure 3.2. Note that this data falls into three relatively distinct clusters. We would therefore expect some success in modeling each cluster with a normal distribution. The contour lines of a normal distribution with the same mean and covariance are overlaid on the data clusters.

The data used in this example is tabulated in Table 3.1. There are fifteen examples of each type of object, and each object has two features.

The mean and covariance for each class can be readily calculated. The method is shown in the next section.

$$\mu_1 = [4.074, 3.855] \quad \Sigma_1 = \begin{bmatrix} 0.727 & 0.087 \\ 0.087 & 0.609 \end{bmatrix} \quad \Sigma_1^{-1} = \begin{bmatrix} 1.4 & -0.2 \\ -0.2 & 1.67 \end{bmatrix} \tag{3.10}$$

Example 3

$$\mu_2 = [-4.087, -4.07] \quad \Sigma_2 = \begin{bmatrix} 1.65 & 0.6347 \\ 0.634 & 0.974 \end{bmatrix} \quad \Sigma_2^{-1} = \begin{bmatrix} 0.806 & -0.525 \\ -0.525 & 1.368 \end{bmatrix} \tag{3.11}$$

$$\mu_3 = [-3.87, 3.63] \quad \Sigma_3 = \begin{bmatrix} 0.569 & -0.263 \\ -0.263 & 0.683 \end{bmatrix} \quad \Sigma_3^{-1} = \begin{bmatrix} 2.14 & 0.823 \\ 0.823 & 1.78 \end{bmatrix} \tag{3.12}$$

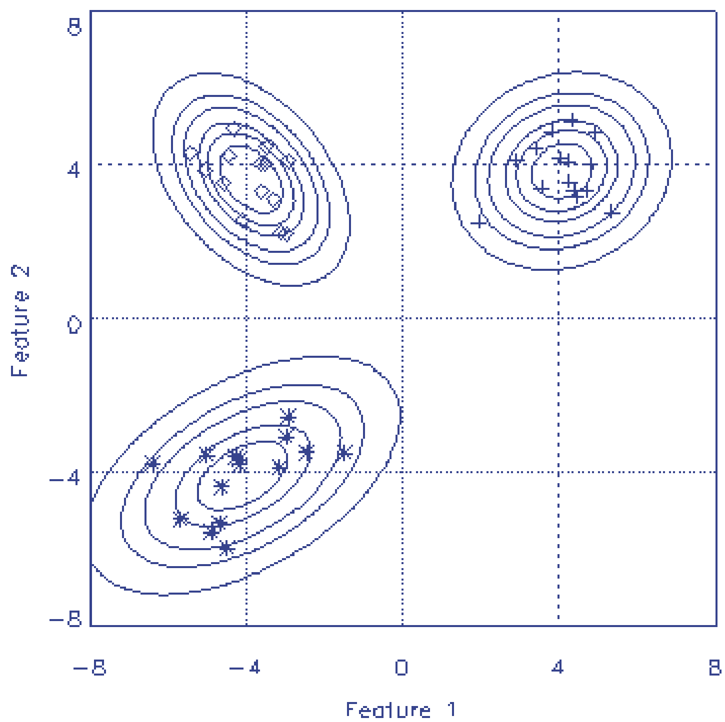

Figure 3.2: A scatter plot of feature data with contours corresponding to normal distributions fit to each cluster.

Class α_1		Class α_2		Class α_3	
Feature 1	Feature 2	Feature 1	Feature 2	Feature 1	Feature 2
3.41	4.44	-4.26	-3.56	-4.31	4.97
3.81	4.89	-2.92	-2.54	-4.58	3.53
4.40	3.35	-2.96	-3.07	-3.61	3.33
5.34	2.77	-4.89	-5.54	-3.64	4.04
4.24	4.09	-2.45	-3.47	-2.90	4.13
4.25	3.56	-4.15	-3.77	-5.04	3.89
2.92	4.12	-5.02	-3.54	-3.50	4.05
4.92	4.85	-1.49	-3.49	-2.97	2.22
3.56	3.41	-4.68	-5.31	-3.47	4.48
4.44	3.18	-4.49	-5.97	-3.13	2.24
4.03	4.17	-4.16	-3.64	-4.11	2.56
4.33	5.14	-6.39	-3.76	-5.42	4.34
4.80	4.00	-3.14	-3.85	-4.45	4.24
1.96	2.52	-4.63	-4.35	-3.31	3.13
4.70	3.33	-5.68	-5.19	-3.61	3.32

Table 3.1: Table data for three classes of objects described by two features.

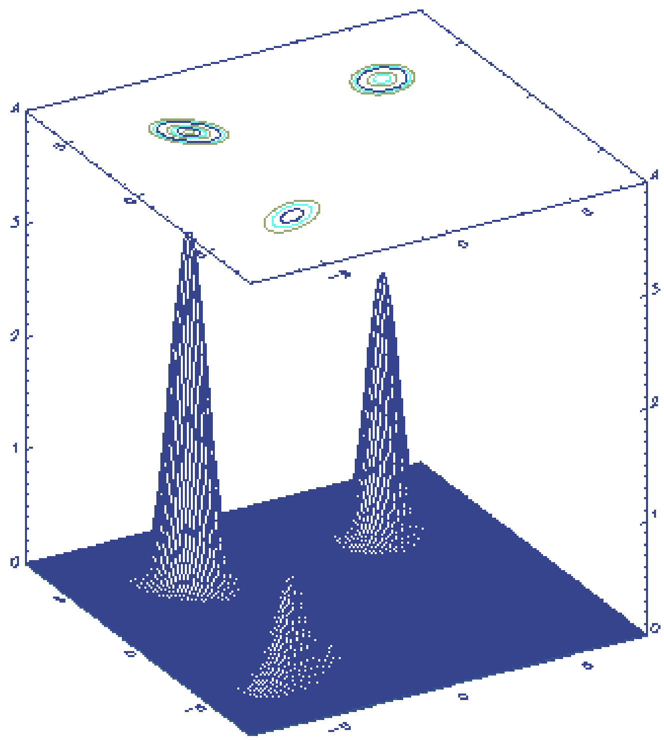

Figure 3.3: An illustration of the normal distribution models for three data clusters showing the surfaces and contours of the models.

Note that in Σ_1 and Σ_2 the off-diagonal terms are positive, showing a positive correlation between the feature values, while Σ_3 has negative off-diagonal terms, showing negative correlation between its features. The contours in Figure 3.2 for class α_1 are nearly circular, showing that the cross correlation is very small (0.087). The contours for α_2 are quite elongated and have a positive slope in the coordinate plane, showing a fairly strong positive correlation (0.635). The contours for α_3 are less elongated and show a negative slope, corresponding to the negative correlation (-0.26). (You can find the cluster for each class by looking at the coordinates of the mean feature vector.)

3.1.3 Calculation of Model Parameters

The advantage of using the normal distribution is that each class can be modeled by a simple parametric function. The mean vector and covariance matrix for each class are easly calculated from labeled data.

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N_c} \mathbf{x}_n \tag{3.13}$$

where N_k is the number of examples of class k feature vectors. This calculation has to be done separately for each class since each has a different mean vector. We can also calculate the terms of the covariance matrix using the data for each class.

$$s_{ij} = \frac{1}{N_k - 1} \sum_{n=1}^{N_k} (x_{ni} - \mu_i)(x_{nj} - \mu_j)$$

where x_{ni} is component i of the feature vector $\mathbf{x}_n$ in class k.

In IDL it is convenient to arrange the data for a particular class as a matrix X in which each row represents a feature vector for a different data point. There are as many rows as feature vectors for a given class and as many columns as there are features. Then the mean is given by the function `mean`(X) and the covariance is given by `cov`(X). The matrix X would correspond to the pair of columns in Table 3.1 associated with the class of interest.

3.1.4 Minimum Error Decisions

The expected loss in associating a given observation $\mathbf{x}$ with class ω_j is given by

$$L_j(\mathbf{x}) = \sum_{i=1}^{c} \lambda(\omega_j|\alpha_i)P(\alpha_i)p(\mathbf{x}|\alpha_i) \tag{3.14}$$

If $\lambda(\omega_i|\alpha_i) = 0$ and $\lambda(\omega_j|\alpha_i) = 1$ for all $j \neq i$, the above reduces to

$$L_j(\mathbf{x}) = 1 - P(\mathbf{x}|\alpha_j)$$

Hence, the probability of error is minimized by associating $\mathbf{x}$ with the class that maximizes $P(\mathbf{x}|\alpha_j)$. The probability $P(\mathbf{x}|\alpha_j)$ is modeled as a normal distribution with mean μ_j and covariance Σ_j. That is, we assume that we know the distribution once we know which class was chosen. The expression for the probability is given by specializing Equantion 3.1 for each class.

$$P(\mathbf{x}|\alpha_j) = \frac{1}{(2\pi)^{M/2}|\Sigma_j|^{1/2}} e^{-\frac{1}{2}\left[(\mathbf{x}-\mu_j)\Sigma_j^{-1}(\mathbf{x}-\mu_j)'\right]} \tag{3.15}$$

When we are using the normal distribution as a model, it is more convenient to maximize $\ln(P(\mathbf{x}|\alpha_j))$. If we eliminate terms that are the same for all j, we find that we should maximize the function

$$g_j(\mathbf{x}) = -\ln|\Sigma_j| - (\mathbf{x}-\mu_j)\Sigma_j^{-1}(\mathbf{x}-\mu_j)' \tag{3.16}$$

This function is a *discriminant function.* Choosing the value of j that maximizes $g_j(\mathbf{x})$ is equivalent to choosing the j that minimizes the loss. In the above equation μ_j and Σ_j are the mean and covariance for class j. The discriminant functions for the previous example are shown in Figure 3.4. The decision boundaries correspond to the contours where the decision functions meet. These are shown on a contour plot in Figure 3.5. It is easy to see why the boundaries are curved contours in this example.

The term on the right is the square of the Mahalanobis distance measure from the point $\mathbf{x}$ to the point μ_j. If we write

$$d_j^2(\mathbf{x}, \mu_j) = (\mathbf{x}-\mu_j)\Sigma_j^{-1}(\mathbf{x}-\mu_j)'$$

we have a distance measure in Σ_j units. The discriminant function can be written as

$$g_j(\mathbf{x}) = c_j - d_{\Sigma_j}^2(\mathbf{x}, \mu_j) \tag{3.17}$$

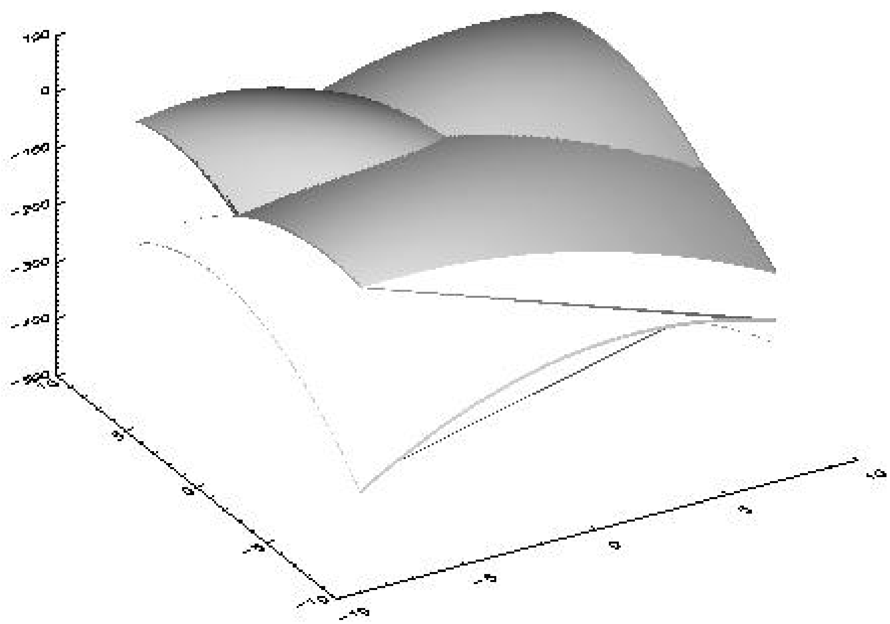

Figure 3.4: Discriminant function for the three data classes shown in the previous example.

where

$$c_j = -\ln|\Sigma_j|$$

is a constant, with a different value for each j. For a given point $\mathbf{x}$, we need to simply calculate $g_j(\mathbf{x})$ for $j = 1, \ldots, k$ and pick the one that is greatest. This will give us the decision that minimizes the probability of error. The computation of $g_j(\mathbf{x})$, in turn, involves calculation of $d_{\Sigma_j}(\mathbf{x}, \mu_j)$. Because of the minus sign, moving the point $\mathbf{x}$ closer to μ_j increases $g_j(\mathbf{x})$. Note, however, that because the constants c_j are different for each class we don't always associate $\mathbf{x}$ with the closest μ_j. The constants will not vary with class when all the Σ_j are equal. In that case, some even nicer simplifications occur, as discussed below.

3.1.5 Decision Regions and Decision Boundaries with minumum error decisions

A decision region F_i consists of all the points such that $L_i(\mathbf{x}) < L_j(\mathbf{x})$ for all $j \neq i$. One can find the decision boundary between regions F_i and F_j by finding the points that fall on the contour

$$L_i(\mathbf{x}) = L_j(\mathbf{x}) \tag{3.18}$$

Tracing the boundaries between regions is helpful in visualizing the layout of feature space. However, one has to be careful, because a point on the boundary between F_i and F_j may actually belong to F_m. This will happen if $L_m(\mathbf{x})$ is the lowest loss. Equation (3.18) only looks for the points that are on the boundary between two particular regions. Further examination is usually needed.

We can find the decision boundary for the minimum error criterion by solving

$$g_i(\mathbf{x}) = g_j(\mathbf{x}) \tag{3.19}$$

In general, these boundaries are quadratic surfaces of dimension $M - 1$ in the M-dimensional feature space. If we substitute (3.18) into (3.19) and rearrange we find

$$(\mathbf{x} - \mu_i)\Sigma_i^{-1}(\mathbf{x} - \mu_i)' - (\mathbf{x} - \mu_j)\Sigma_j^{-1}(\mathbf{x} - \mu_j)' = \ln \frac{|\Sigma_j|}{|\Sigma_i|} \tag{3.20}$$

An example of decision boundaries in a 2D feature space is shown in Figure 3.5. The boundaries are nonlinear, and serve to separate the three classes on a pair-by-pair basis. Note that the three boundaries cross at a single point.

Case of equal covariance matrices

When the classes all have the same covariance, we can drop the subscripts on the Σ matrices and write (3.20) as

$$(\mathbf{x} - \mu_i)\Sigma^{-1}(\mathbf{x} - \mu_i)' - (\mathbf{x} - \mu_j)\Sigma^{-1}(\mathbf{x} - \mu_j)' = 0 \tag{3.21}$$

This is the equation of a $M - 1$ dimensional plane that divides the M dimensional feature space into regions.

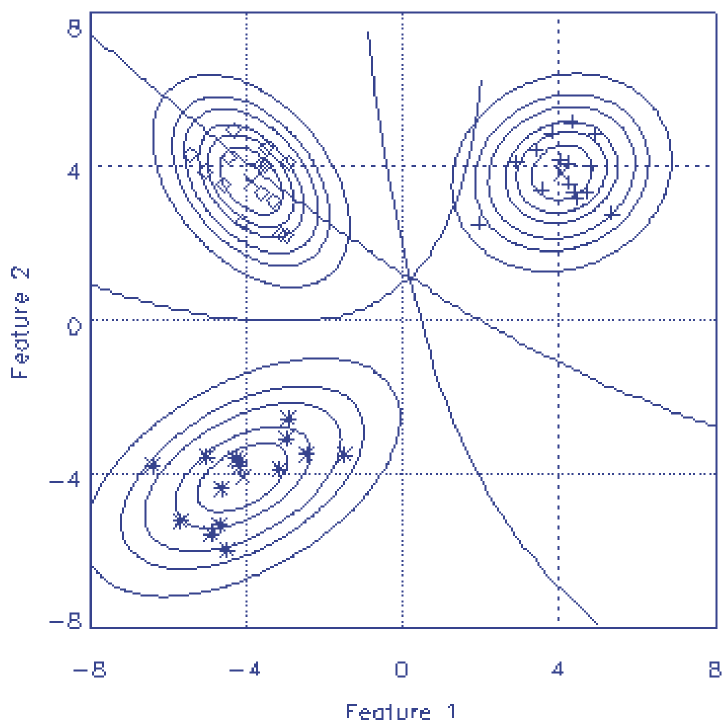

Figure 3.5: An illustration of the boundaries between the decision regions for a system with three classes. The individual covariance functions were used to compute the discriminant functions, leading to the curved boundaries.

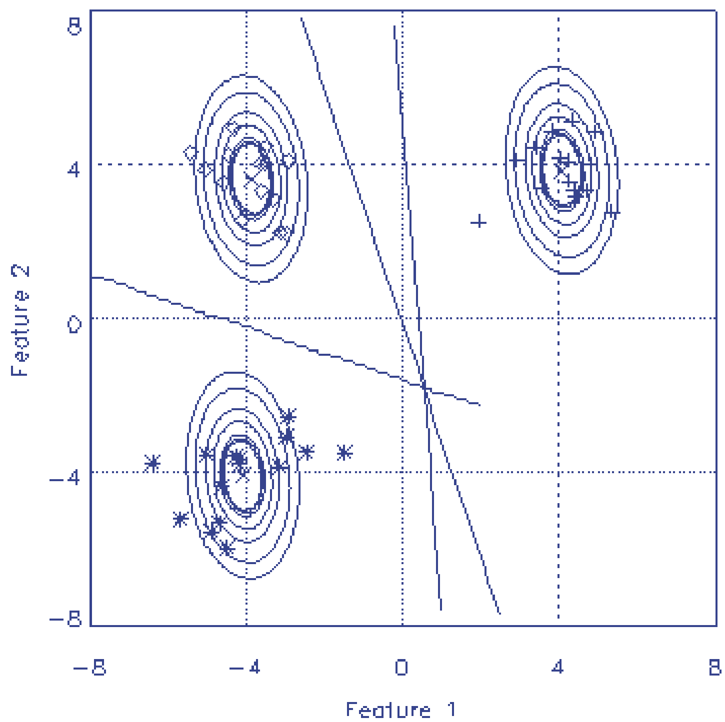

Figure 3.6: An illustration of the boundaries between the decision regions when a pooled covariance matrix is used. This produces linear boundaries that are not necessarily in the optimum positions.

If there is reason to believe that all of the classes have equal covariance, then one can compute a single covariance matrix by subtracting the mean value from the data for each class and pooling the results. When that is done, we find that the covariance function for the sample data of Table 3.1 is

$$\Sigma = \begin{bmatrix} 0.939 & 0.146 \\ 0.146 & 0.721 \end{bmatrix}$$

We can now see in Figure 3.6 that all of the boundaries are straight lines. This will always be the case with a multivariate normal distribution when the classes have equal covariance matrices. It is worth emphasizing that in this problem the covariance matrices of the individual classes are really not equal. However, by using a covariance matrix that is computed with the pooled data we are forced to use the same covariance matrix with each class. The effect can be seen by comparing Figures 3.3 and 3.6.

3.2 Problems

1. The feature space for this pattern recognition problem is two dimensional and there are two pattern classes denoted by α_1 and α_2. The probability density function for the feature vector $\mathbf{x}$, for each of the classes is normal. The parameters for each class are

$$\mu_1 = [4, 0] \quad \Sigma_1 = \begin{bmatrix} 1 & .2 \\ .2 & 1 \end{bmatrix}$$

$$\mu_2 = [0, -4] \quad \Sigma_2 = \begin{bmatrix} 1 & .2 \\ .2 & 1 \end{bmatrix}$$

(a) Sketch the equal probibility contours for $P(\mathbf{x}|\alpha_1)$ and $P(\mathbf{x}|\alpha_2)$.

(b) Assume that $P(\alpha_1) = P(\alpha_2)$ and that a minimum-error decision rule is to be used. Find the equation of the boundary between the regions F_1 and F_2.

(c) Assume that $P(\alpha_1) = 2P(\alpha_2)$ and that a minimum-error decision rule is to be used. Find the equation of the boundary between the regions F_1 and F_2.

(d) Assume that $P(\alpha_1) = P(\alpha_2)$ and that the decision risks are given by $\lambda(\omega_1|\alpha_2) = 1$, $\lambda(\omega_1|\alpha_2) = 2$, and $\lambda(\omega_1|\alpha_1) = \lambda(\omega_2|\alpha_2) = 0$. Find the equation of the boundary between the regions F_1 and F_2.

(e) Assume that $P(\alpha_1) = 2P(\alpha_2)$ and that the decision risks are given by $\lambda(\omega_1|\alpha_2) = 1$, $\lambda(\omega_1|\alpha_2) = 2$, and $\lambda(\omega_1|\alpha_1) = \lambda(\omega_2|\alpha_2) = 0$. Find the equation of the boundary between the regions F_1 and F_2.

2. The feature space for this pattern recognition problem is two dimensional and there are two pattern classes denoted by α_1 and α_2. The probability density function for the feature vector $\mathbf{x}$, for each of the classes is normal. The parameters for each class are

$$\mu_1 = [4, 0] \qquad \Sigma_1 = \begin{bmatrix} 1 & .2 \\ .2 & 1 \end{bmatrix}$$

$$\mu_2 = [0, -4] \quad \Sigma_2 = \begin{bmatrix} 1 & -.2 \\ -.2 & 1 \end{bmatrix}$$

(a) Sketch the equal probability contours for $P(\mathbf{x}|\alpha_1)$ and $P(\mathbf{x}|\alpha_2)$.

(b) Assume that $P(\alpha_1) = P(\alpha_2)$ and that a minimum-error decision rule is to be used. Find the equation of the boundary between the regions F_1 and F_2.

(c) Assume that $P(\alpha_1) = 2P(\alpha_2)$ and that a minimum-error decision rule is to be used. Find the equation of the boundary between the regions F_1 and F_2.

(d) Assume that $P(\alpha_1) = P(\alpha_2)$ and that the decision risks are given by $\lambda(\omega_1|\alpha_2) = 1$, $\lambda(\omega_1|\alpha_2) = 2$, and $\lambda(\omega_1|\alpha_1) = \lambda(\omega_2|\alpha_2) = 0$. Find the equation of the boundary between the regions F_1 and F_2.

(e) Assume that $P(\alpha_1) = 2P(\alpha_2)$ and that the decision risks are given by $\lambda(\omega_1|\alpha_2) = 1$, $\lambda(\omega_1|\alpha_2) = 2$, and $\lambda(\omega_1|\alpha_1) = \lambda(\omega_2|\alpha_2) = 0$. Find the equation of the boundary between the regions F_1 and F_2.

3. The feature space for this pattern recognition problem is three dimensional and there are two pattern classes denoted by α_1 and α_2. The probability density function for the feature vector $\mathbf{x}$, for each of the classes is normal. The parameters for each class are

$$\mu_1 = [-1, -1, -1] \quad \Sigma_1 = \begin{bmatrix} 1 & 0.5 & 0 \\ 0.5 & 1 & 0.5 \\ 0 & 0.5 & 1 \end{bmatrix}$$

$$\mu_2 = [1, 1, 1] \quad \Sigma_2 = \begin{bmatrix} 1 & 0.5 & 0 \\ 0.5 & 1 & 0.5 \\ 0 & 0.5 & 1 \end{bmatrix}$$

Assume that $P(\alpha_1) = P(\alpha_2)$ and that a minimum-error decision rule is to be used. Find the equation of the boundary between the regions F_1 and F_2.

4. Reproduce equations (3.10), (3.11) and (3.12) by using the data in Table 3.1. You may want to write a computer program using functions such as those given at the end of this chapter.

5. Find the mean and covariance matrix for each class for the jockey, football player and swimmer data and plot the equal probability contours for the feature vectors for each class assuming that the data can be fin with a bivariate normal distribution. Plot the decision boundaries on the feature space for the minum error probability decision rule.

6. Find the pooled covariance matrix for the jockey, football player and swimmer data. Plot the equal probability contours and the decision region boundaries for the minum error probability decision rule under the assumption that each class has a covariance equal to the one you calculated from the pooled data. What is the effect on the location of the decision boundaries compared with your answer to the previous problem?

3.3 Computer programs

Computer programs to calculate the mean and covariance can easily be written in IDL. Listed below are probrams for functions mean(H) and cov(H) where H is an array of feature data. All of the data in H is assumed to be from the same class. To find the mean and covariance for all of the classes it is necessary to run the computation with a data array for each class. H is assumed to be arranged with each feature vector as a row. There are as many rows as there are examples and as many columns as there are features.

```
FUNCTION MEAN,H
;+
; MU=MEAN(H) returns an array that has one less array dimension
 ; than does H. That is, if H is a 2D array then MU is a 1D
 ; vector. If H is a 3D array then MU is a 2D array formed by
 ; averaging over the third dimension of H. Each value of MU is
 ; the average value over the highest dimension of H.
 ; If H is a scalar then H is returned. MU is always of type FLOAT
 ; unless H is of type DOUBLE, COMPLEX or DOUBLE COMPLEX, in
 ; which case MU is the same type as H.
 ;-
 S=SIZE(H)
 ; S[0] is the number of dimensions of H
 ; S[S[0]] is the number of elements in the highest dimension
 ; of H.
 IF S[0] EQ 0 THEN RETURN,H $
 ELSE RETURN,TOTAL(H,S[0])/FLOAT(S[S[0]])
 END
 ===========================================================
 FUNCTION COV,H
 ;+
 ; C=COV(H) returns the covariance matrix for a table of data
 ; in an array H. H is assumed to be arranged with each example
 ; as a row and each column as a feature. C is an array of size
 ; NxN where N is the number of columns of H. C is always of type
 ; FLOAT unless H is of type DOUBLE, COMPLEX or DOUBLE COMPLEX,
 ; in which case C is the same type as H.
 ;-
 S=SIZE(H)
 MU=MEAN(H)
 MUV=(FLTARR(1,S[2])+1)##MU
 X=H-MUV
```

```
RETURN,X#TRANSPOSE(X)/FLOAT(S[2]-1)
END
```