2.1.2 Linear Discriminant Analysis

Now assume that we have K classes, each with prior probability π_k and $\sum_{k=1}^{K} \pi_k = 1$. Suppose $f_k(\mathbf{x})$ is the class-conditional density of a given feature vector $\mathbf{x}$ in class $G = k$. A simple application of Bayes theorem gives us the posterior probability:

$$Pr(G = k|\mathbf{x}) = \frac{f_k(\mathbf{x})\pi_k}{\sum_{l=1}^{K} f_l(\mathbf{x})\pi_l} \tag{2.6}$$

Therefore, the maximum a posterior (MAP) estimate of the class label of $\mathbf{x}$ is:

$$\hat{G(\mathbf{x})} = \arg\max_k f_k(\mathbf{x})\pi_k \tag{2.7}$$

Linear and quadratic discriminant analysis assumes that each class density is multivariate Gaussian.

$$f_k(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu_k)^T\Sigma_k^{-1}(\mathbf{x}-\mu_k)} \tag{2.8}$$

where μ_k and Σ_k are the mean and covariance matrix of the Gaussian distribution, respectively.

Linear discriminant analysis (LDA) arises in the special case when we assume that all classes have a common covariance matrix: $\Sigma_k = \Sigma, \forall k$. In other words, the density functions of different classes have the same covariance but with different means, *i.e.*, they are shifted versions of each other.

Let's first look at two-class cases. By looking at the log-ratio of the posterior probability of $\mathbf{x}$ being in class k or l:

$$\begin{aligned} \log \frac{Pr(G = k|\mathbf{x})}{Pr(G = l|\mathbf{x})} &= \log \frac{f_k(\mathbf{x})}{f_l(\mathbf{x})} + \log \frac{\pi_k}{\pi_l} \\ &= \log \frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k + \mu_l)^T\Sigma^{-1}(\mu_k - \mu_l) + \mathbf{x}^T\Sigma^{-1}(\mu_k - \mu_l) \end{aligned} \tag{2.9}$$

We know that $\mathbf{x}$ belongs to class k if equation (2.9) > 0; otherwise $\mathbf{x}$ belongs to class l. In other words, the decision boundary between class k and class l is a hyperplane in p-dimensional place, *i.e.*, the decision boundary i slinear.

In practice, we do not know the parameters of the Gaussian distributions, and will need to estimate using our training data:

- $\hat{\pi_k} = N_k/N$, where N_k is the number of training data in class k among all N training data.
- $\hat{\mu_k} = (\sum_{g_i=k} \mathbf{x}_i)/N_k$, the mean of the class-k training data.

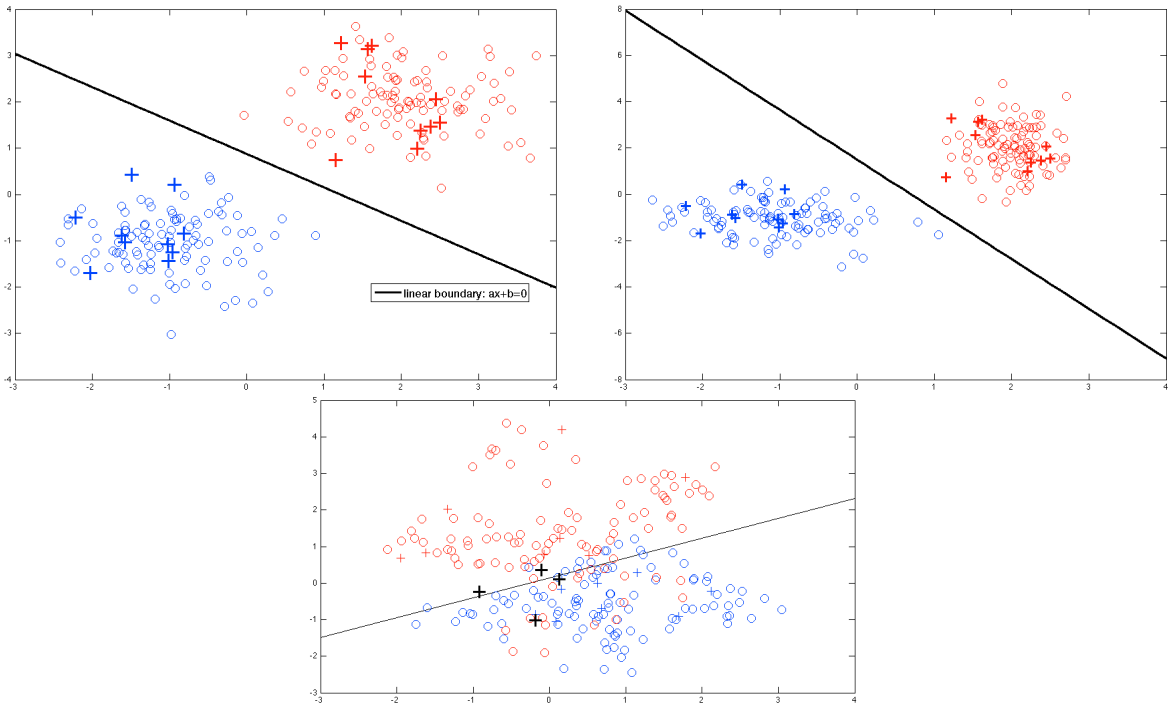

Figure 2.6: Training (circle) and test (plus sign) data of the 2-class examples. Top left: Gaussians of equal variances. Top right: Gaussians of different variances. Bottom: Each class being mixed Gaussians with equal variances

- $\hat{\Sigma} = (\sum_{k=1}^{K} \sum_{g_i=k} (\mathbf{x}_i - \hat{\mu_k})(\mathbf{x}_i - \hat{\mu_k})^T)/(N-K)$

Once we have the Gaussian density functions of all classes estimated from training data, given a test data $\mathbf{x}$, we can classify it according to equation (2.9). More specifically, the hyperplance decision boundary has a constant (intercept) $b = \log \frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k + \mu_l)^T \Sigma^{-1}(\mu_k - \mu_l)$, and a slope $\mathbf{a} = \Sigma^{-1}(\mu_k - \mu_l)$. Given an $\mathbf{x}$, it can be classified to k if $\mathbf{x}^T\mathbf{a} + b > 0$, and to j otherwise.

Example: Implement LDA for 2-class classification.

1. Test it on the first 2-class example used in Fig. 2.1 (left) (the two classes do have equal variance). You should expect a 100% classification accuracy, with a linear decision boundary as shown in the figure below ($b = 0.5439, \mathbf{a} = (-0.4517 \quad -0.6257)^T$).

2. Now test it on another 2-class example where the 2 classes have un-equal variance (data is provided in "2Duv.txt"). Even though the assumption of equal variance is violated, LDA still works well in this case with 100% classification accuracy. The linear decision boundary is $b = 0.5186, \mathbf{a} = (-0.7452 \quad -0.3472)^T$

3. Test it on the second 2-class example used in Fig. 2.1 (right) (each class is a mixture of equal-variance Gaussian distributions). You should expect a 80% classi-

fication accuracy, with a linear decision boundary as shown in the figure below ($b = 0.1212, \mathbf{a} = (0.4703 \quad -0.8688)^T$). As explained earlier, for this type of classes, linear decision boundary is unlikely to be optimal.

Now how do we deal with cases that have more than 2 classes? It is a straightforward application of equation (2.9).

$$\hat{G(\mathbf{x})} = \arg\max_k f_k(\mathbf{x})\pi_k = \arg\max_k \{\mathbf{x}^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + \log\pi_k\} \tag{2.10}$$

In other words, given a test data $\mathbf{x}$, we will calculate the linear discriminant function $\delta_k(\mathbf{x}$ as defined in equation (2.10) for each of the class k. $\mathbf{x}$ will then be classified to the class that gives the largest value of δ_k.

$$\hat{G}(\mathbf{x}) = \arg\max_k \delta_k(\mathbf{x}) \tag{2.11}$$

Evidently, this applies to $K \geq 2$ classes including 2-classes cases.

Example: Implement general LDA for K-class classification.

1. Test it on the 3-class example used in Fig. 2.4 (the three classes do have equal variance). You should expect a 100% classification accuracy. Note how with linear decision boundaries, LDA is able to correctly classify the three classes while linear regression (in the last section) failed unless a higher order polynomial expansion is used.

2. Test on the first 3-class example used in Fig. 2.3 (each class is a mixture of equal-variance Gaussian). You will encounter a 13.3% classification error.

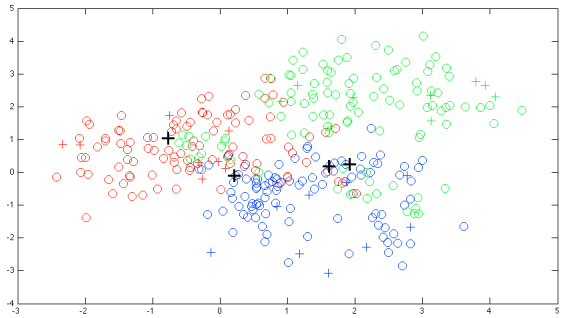

Figure 2.7: Training (circle) and test (plus sign) data of the 2D example.

$*$ To extend to quadratic decision boundaries, there are two alternative approaches:

- Similar to linear regression, we can apply basis expansion on the feature vectors to include higher order terms, and then apply LDA on the extended vectors. Apply

LDA on the two examples where each class is generated by a mixture of Gaussians (*i.e.*, 2-class example in Fig. 2.6 bottom panel and the 3-class example in Fig. 2.7), with f eature vectors expanded with the second-order term, you will see the classification accuracy increases to 85% (compared to 80%) and 90% (compared to 86.7%), respectively.

- Quadratic discriminant analysis (QDA), where the covariance matrix Σ_k is not assumed to be equal and the decision function becomes: $\delta_k(\mathbf{x}) = -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(\mathbf{x} - \mu_k)^T\Sigma_k^{-1}(\mathbf{x} - \mu_k) + \log\pi_k$