Linear Discriminant Analysis and Quadratic Discriminant Analysis

COGNOAB

November 2024

1 Linear Discriminant Analysis for Dimensions Reduction

- Consider a pattern classification problem, where we have C-classes, e.g. seabass, tuna, salmon ...
- Each class has N_i m-dimensional samples, where i = *1,2, ..., C*
- Hence, we have a set of m-dimensional samples $\{x_1, x_2, \ldots, x_{N_i}\}$ belonging to class ω_i.
- Stacking these samples from different classes into one big fat matrix $\mathbf{X}$ such that each column represents one sample
- We seek to obtain a transformation of X to Y through projecting the samples in X onto a hyperplane with dimension C-1.

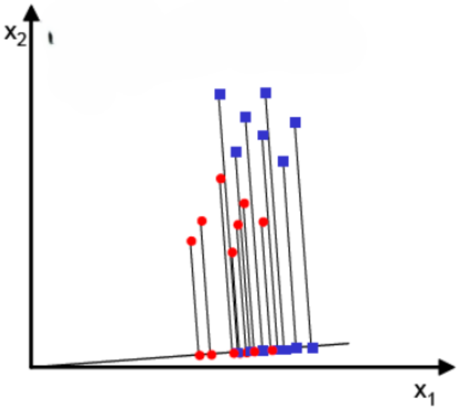
1.1 LDA for Two Class

Assume we have m-dimensional samples $\{x_1, x_2, \ldots, x_N\}$, N_1 of which belong to class ω_1 and N_2 belong to class ω_2.
We seek to obtain a scalar $\mathbf{y}$ by projecting the samples $\mathbf{x}$ onto a line (C-1 space, C = 2).

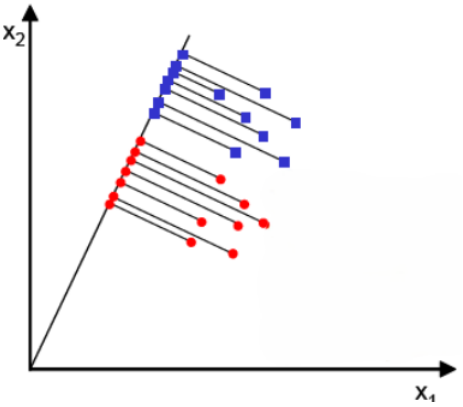
$$y = w^T x \quad \text{where} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \quad \text{and} \quad w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}$$

- where $\mathbf{w}$ is the projection vector used to project $\mathbf{x}$ to $\mathbf{y}$.

⇒ Of all the possible lines, we would like to select the one that maximizes the separability of the scalars.



The bad projection



The good projection

- Figure 1:The two classes are not well separated when projected onto this line
- Figure 2:This line succeeded in separating the two classes and in the meantime reducing the dimensionality of our problem from two features (x_1,x_2) to only a scalar value **y**.

In order to find a good projection vector, we need to define a measure of separation between the projections.
The mean vector of each class in **x** and **y** feature space is:

$$\mu = \frac{1}{N_i} \sum_{x \in \omega_i} x$$

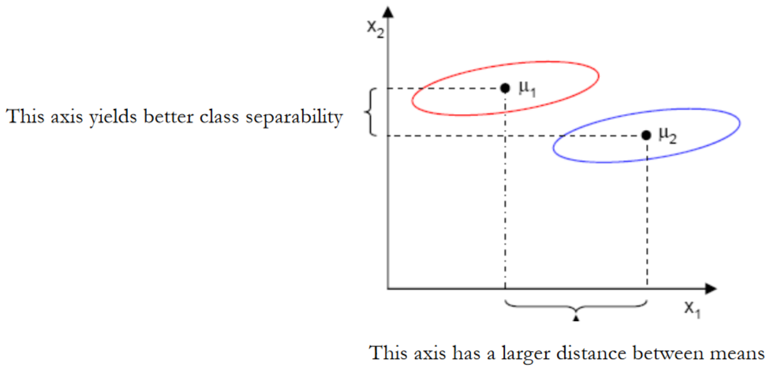
and

$$\tilde{\mu} = \frac{1}{N_i} \sum_{y \in \omega_i} y = \frac{1}{N_i} \sum_{x \in \omega_i} w^T x = w^T \frac{1}{N_i} \sum_{x \in \omega_i} x = w^T \mu$$

We could then choose the distance between the projected means as our objective function

$$J(w) = |\tilde{\mu}_1 - \tilde{\mu}_2| = |w^T \mu_1 - w^T \mu_2| = |w^T (\mu_1 - \mu_2)|$$

However, the distance between the projected means is not a very good measure since it does not take into account the standard deviation within the classes.


x₂
μ₁
μ₂
x₁
This axis yields better class separability
This axis has a larger distance between means

Axis Comparing Distance

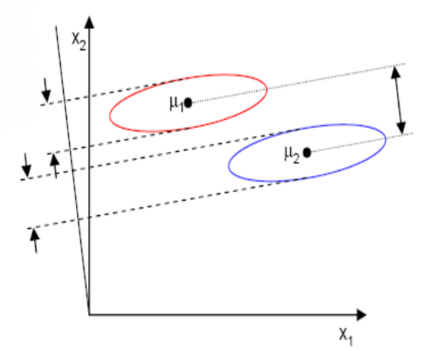
⇒ The solution proposed by Fisher is to **maximize a function that represents the difference between the means, normalized by a measure of the within-class variability**, or the **so-called scatter**.
For each class we define the scatter, an equivalent of the variance, as:

$$\tilde{s_i}^2 = \sum_{y \in \omega_i} (y - \tilde{\mu}_i)^2$$

$\tilde{s_i}^2$ measures the variability within class ω_i after projecting it on the y-space.
⇒ Thus $\tilde{s_1}^2 + \tilde{s_2}^2$ measures the variability within the two classes at hand after projection, hence it is called within class scatter of the projected samples.

The Fisher linear discriminant is defined as the linear function $w^T x$ that maximizes the criterion function: (the distance between the projected means normalized by the within- class scatter of the projected samples.

$$J(w) = \frac{|\tilde{\mu}_1 - \tilde{\mu}_2|^2}{\tilde{s_1}^2 + \tilde{s_2}^2}$$



$\Rightarrow$ Therefore, we will be looking for a projection where examples from the same class are projected very close to each other and, at the same time, the projected means are as farther apart as possible

In order to find the optimum projection w^*, we need to express $J(w)$ as an explicit function of w.
We will define a measure of the scatter in multivariate feature space x which are denoted as ***scatter matrices***.

$$S_i = \sum_{x \in \omega_i} (x - \mu_i)(x - \mu_i)^T$$

$$S_w = S_1 + S_2$$

S_i: covariance matrix of class ω_i
S_w: The **within-class scatter matrix**

Now, the scatter of the projection y can then be expressed as a function of the scatter matrix in feature space x.

$$\begin{aligned}
\tilde{S}_i &= \sum_{y \in \omega_i} (y - \tilde{u}_i)^2 \\
&= \sum_{x \in \omega_i} (w^T x - w^T \mu_i)^2 \\
&= \sum_{x \in \omega_i} w^T (x - \mu_i)(x - \mu_i)^T w \\
&= w^T \left(\sum_{x \in \omega_i} (x - \mu_i)(x - \mu_i)^T \right) w \\
&= w^T S_i w
\end{aligned}$$

$$\Rightarrow \tilde{S}_W = \tilde{S}_1 + \tilde{S}_2 = w^T S_1 w + w^T S_2 w = w^T (S_1 + S_2) w = w^T S_w w$$

$\tilde{S_W}$:within-class scatter matrix of the projected samples **y**

Similarly, the difference between the projected means (in y space) can be expressed in terms of the means in the original feature space (x-space)

$$\tilde{S}_B = (w^T \tilde{\mu}_1 - w^T \tilde{\mu}_2)^2 = w^T (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T w = w^T S_B w$$

S_B :between-class scatter of the original samples
$\tilde{S_B}$:between-class scatter of the projected samples **y**

We can finally express the Fisher criterion in terms of S_W and S_W as:

$$J(w) = \frac{|\tilde{\mu}_1 - \tilde{\mu}_2|^2}{\tilde{s_1}^2 + \tilde{s_2}^2} = \frac{w^T S_B w}{w^T S_W w}$$

⇒ Hence J(w) is a measure of the difference between class means (encoded in the between-class scatter matrix) normalized by a measure of the within-class scatter matrix.

To find the maximum of J(w), we differentiate and equate to zero:

$$\begin{aligned}
&\frac{d}{dw} J(w) = \frac{d}{dw}\left(\frac{w^T S_B w}{w^T S_W w}\right) = 0 \\
&\Rightarrow (w^T S_W w)\frac{d}{dw}(w^T S_B w) - (w^T S_B w)\frac{d}{dw}(w^T S_W w) = 0 \\
&\Rightarrow (w^T S_W w)(S_B + S_B^T)w - (w^T S_B w)(S_W + S_W^T) = 0 \\
&\Rightarrow (w^T S_W w) 2 S_B w - (w^T S_B w) 2 S_W w = 0 \\
&\text{Note:} \quad S_B = S_B^T \quad \text{and} \quad S_W = S_W^T
\end{aligned}$$

Next, Dividing by $2w^T S_w w$, we have:

$$\begin{aligned}
&\Rightarrow \frac{w^T S_W w}{w^T S_W w} S_B w - \frac{w^T S_B w}{w^T S_W w} S_W w = 0 \\
&\Rightarrow S_B w - J(W) S_W w = 0 \quad \text{Note} : J(w) \quad \text{is scalar.} \\
&\Rightarrow S_W^{-1} S_B w - J(w) w = 0
\end{aligned}$$

Next, Solving the generalized eigen value problem:

$$S_W^{-1} S_B w = J(w) w \quad \text{where} \quad \lambda = J(w) = scalar$$

yields

$$w^* = \arg\max_w J(w) = \arg\max_w \left(\frac{w^T S_B w}{w^T S_W w}\right) = S_W^{-1}(\mu_1 - \mu_2)$$

⇒ This is known as Fisher's Linear Discriminant, although it is not a discriminant but rather a specific choice of direction for the projection of the data down to one dimension

1.2 LDA for C-Classes

Now, we have **C-classes** instead of just two
We are now seeking $C-1$ projections $\{y_1, y_2, \ldots, y_{C-1}\}$ by means of $C-1$ projection vectors $\mathbf{w}_i$, where $i = 1, 2, \ldots, C-1$.
The projection vectors $\mathbf{w}_i$ can be arranged by columns into a projection matrix

$$\mathbf{W} = [\mathbf{w}_1 \mid \mathbf{w}_2 \mid \cdots \mid \mathbf{w}_{C-1}]$$

such that:

$$y_i = w_i^T x \Rightarrow y = W^T x$$

where:

$$x \in \mathbb{R}^{m\times 1} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}, \quad y \in \mathbb{R}^{(C-1)\times 1} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{C-1} \end{bmatrix}, \quad \mathbf{W} \in \mathbb{R}^{m\times(C-1)} = [\mathbf{w}_1 \mid \mathbf{w}_2 \mid \cdots \mid \mathbf{w}_{C-1}]$$

If we have n-feature vectors, we can stack them into one matrix as follows:

$$Y = W^T X$$

where:

$$X_{mxn} = \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1n} \\ x_{21} & x_{22} & \ldots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \ldots & x_{mn} \end{bmatrix} \quad , Y_{C-1xn} = \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ y_{(C-1)1} & y_{(C-1)2} & \cdots & y_{(C-1)n} \end{bmatrix}$$

and:

$$\mathbf{W} \in \mathbb{R}^{m\times(C-1)} = [\mathbf{w}_1 \mid \mathbf{w}_2 \mid \cdots \mid \mathbf{w}_{C-1}]$$

In two-classes case, the **within-class scatter** was computed as:

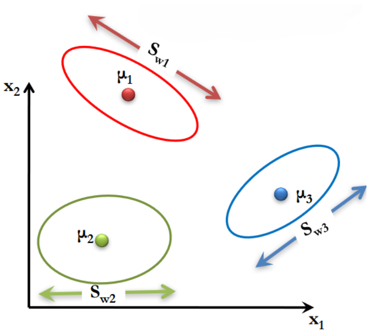
$$S_W = S_1 + S_2$$

This can be generalized in the C-classes case as:

$$S_W = \sum_{i=1}^{C} S_i \quad \text{where} \quad S_i = \sum_{x\in\omega_i} (x-\mu_i)(x-\mu_i)^T$$

$$\mu_i = \frac{1}{N_i}\sum_{x\in\omega_i} x \quad \text{is the mean of class } \omega_i$$

N_i : The number of samples in class ω_i



The S_W example in two-dimensional features with three classes C = 3.

In two-classes case, the **between- class scatter** was computed as:

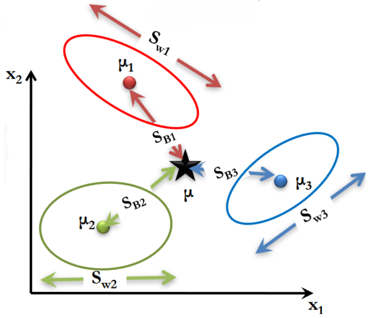
$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

For C-classes case, we will measure the between-class scatter with respect to the mean of all classes as follows:

$$S_B = \sum_{i=1}^{c} N_i(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$$

$$\text{where} \quad \frac{1}{N}\sum_{\forall x} x = \frac{1}{N}\sum_{\forall x} N_i\mu_i \quad \text{and} \quad \mu_i = \frac{1}{N_i}\sum_{x\in\omega_i} x$$

N:Number of all data
N_i:Number of data samples in class ω_i



The S_B example in two-dimensional features with three classes C = 3.

We can define mean vectors for projected sample **y** as:

$$\tilde{\mu}_i = \frac{1}{N_i} \sum_{y \in \omega_i} y \quad \text{and} \quad \tilde{\mu} = \frac{1}{N} \sum_{y \in \omega_i} y$$

While the scatter matrices for the projected samples **y** will be:

$$\tilde{S}_W = \sum_{i=1}^{C} \tilde{S}_i = \sum_{i=1}^{C} \sum_{y \in \omega_i} (y - \tilde{\mu}_i)(y - \tilde{\mu}_i)^T$$

In two-classes case, we have expressed the scatter matrices of the projected samples in terms of those of the original samples as:

$$\tilde{S}_W = W^T S_W W$$

$$\tilde{S}_B = W^T S_B W$$

⇒ It still hold in C-Classes case

In that we are looking for a projection that maximizes the ratio of between-class to within-class scatter, but the projection is no longer a scalar (it has C-1 dimensions), we then use the determinant of the scatter matrices to obtain a scalar objective function:

$$J(W) = \frac{|\tilde{S}_B|}{|\tilde{S}_W|} = \frac{|W^T S_B W|}{|W^T S_W W|}$$

And we will seek the projection $\mathbf{W}^*$ that maximizes this ratio.
To find the maximum of J(W), we differentiate with respect to W and equate to zero. In two-classes case:

$$S_W^{-1} S_B w = \lambda w \quad \text{where} \quad \lambda = J(W) = \text{scalar}$$

For C-classes case, we have C-1 projection vectors, hence the eigen value problem can be generalized to the C-classes case as:

$$S_W^{-1} S_B w_i = \lambda_i w_i \quad \text{where} \quad \lambda_i = J(w_i) = \text{scalar} \quad \text{and} \quad \text{i = 1,2,3,...,C-1}$$

Thus, It can be shown that the optimal projection matrix W* is the one whose columns are the eigenvectors corresponding to the largest eigen values of the following generalized eigen value problem:

$$S_W^{-1} S_B w = \lambda W^*$$

where:

$$\lambda = J(W^*) = \text{scalar} \quad \text{and} \quad W^* = \left[\mathbf{w}_1^* \mid \mathbf{w}_2^* \mid \cdots \mid \mathbf{w}_{C-1}^*\right]$$

2 Linear Discriminant Analysis for Classifications

Linear Discriminant Analysis use Bayes' Theorem for Classifications. Before jumping into **LDA**, let talk about some basic concepts about Bayes' Theorem.

Suppose that we wish to classify an observation into one of **K** classes(K $\geq$ 2). In other words, the qualitative response variable Y can take on K possible distinct and unordered values.
We have the *density function* of **X**:

$$f_k(x) = Pr(X = x|y = K)$$

In other words, $f_k(x)$ is relatively large if there is a high probability that an observation in the kth class has $X \approx x$, and $f_k(x)$ is small if it is very unlikely that an observation in the kth class has $X \approx x$. Then Bayes' theorem states that:

$$Pr(y = K|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$

π_K:the overall or prior probability that a randomly chosen observation comes from the kth class

We will use the abbreviation $p_k(X) = Pr(Y = k|X)$, We refer to pk(x) as the *posterior* probability that an observation posterior $X = x$ belongs to the kth class. That is, it is the probability that the observation belongs to the kth class, given the predictor value for that observation.

Like the Bayes Classification, which classifies an observation to the class for which $p_k(X)$ is largest, has the lowest possible error rate out of all classifiers. Therefore, if we can find a way to estimate $f_k(X)$, then we can develop a classifier that approximates the Bayes classifier. Such an approach is the topic of the following sections.

2.1 Linear Discriminant Analysis for p=1

Assume that we have p=1(It means we have only one predictor or one-dimensional feature), We would like to obtain an estimate for $f_k(x)$ in order to estimate $p_k(X)$:

Suppose that we assume $f_k(x)$ has *normal or Gaussian* distribution, n the one-dimensional setting, the normal density takes the form

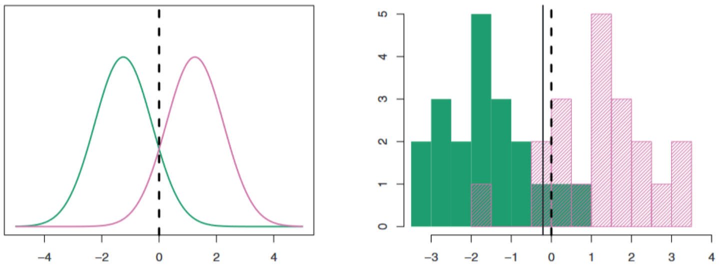
$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} exp\left(-\frac{1}{2\sigma_k^2}(x - \mu_k)^2\right)$$

where μ_k and σ^2 is mean and variance parameters for the kth class.

Next, let us further assume that $\sigma_1^2 \neq ... \neq \sigma_K^2$: that is, there is a shared variance term across all K classes, which for simplicity we can denote by σ^2

Plugging the form of $f_k(x)$ in Bayes' Theorem, we have:

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{1}{2\sigma^2}(x-\mu_k)^2\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{1}{2\sigma^2}(x-\mu_l)^2\right)}$$



Two one-dimensional normal density functions are shown. The dashed vertical line represents the Bayes decision boundary. Right: 20 observations were drawn from each of the two classes, and are shown as histograms. The Bayes decision boundary is again shown as a dashed vertical line. The solid vertical line represents the LDA decision boundary estimated from the training data.

The Bayes classifier involves assigning an observation $X = x$ to the class for which $p_k(x)$ is largest. Taking the log of $p_k(x)$ and rearranging the terms, it is not hard to show that this is equivalent to assigning the observation to the class for which

$$\delta_k(x) = x.\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + log(\pi_k)$$

is largest.

In particular, the following estimates are used:

$$\hat{\mu_k} = \frac{1}{n_k} \sum_{i:y_i=k} x_i$$

$$\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu_k})^2$$

where n is the total number of training observations, and n_k is the number of training observations in the kth class. In the absence of any additional information, LDA estimates π_k using the proportion of the training observations that belong to the kth class. In other words,

$$\hat{\pi_k} = \frac{n_k}{n}.$$

2.2 Linear Discriminant Analysis for $p > 1$

We now extend the LDA classifier to the case of multiple predictors. To do this, we will assume that $X = (X_1, X_2, ..., X_p)$ is drawn from a *variate Gaussian* distribution, with a class-specific covariance vector and a common covariance matrix.

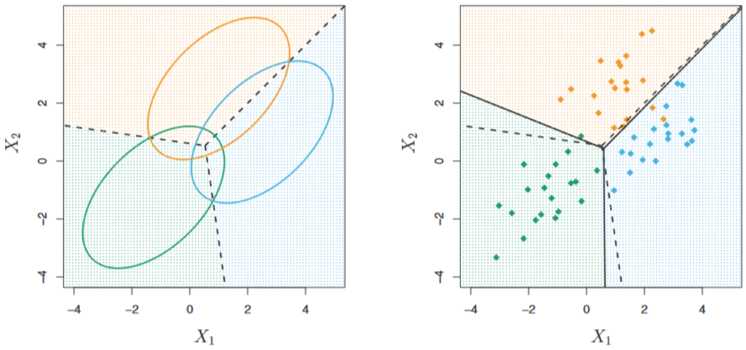
To indicate that a p-dimensional random variable X has a multi- variate Gaussian distribution, we write $X \sim (\mu, \Sigma)$, Here $E(X) = \mu$ is the mean of X (a vector with p components), and $Cov(X) = \Sigma$ is the pxp covariance matrix of X. Formally, the multivariate Gaussian density is defined as:

$$f(x) = \frac{1}{(x\pi)^{p/2}|\Sigma|^{1/2}} exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

In the case of $p > 1$ predictors, the LDA classifier assumes that the observations in the kth class are drawn from a multivariate Gaussian distribution $N(\mu_k, \Sigma)$ where μ_k is a class -specific mean vector, and Σ is a covariance matrix that is common to all K classes.Plugging the density function for the kth class $f_k(X = x)$ into the Bayes' Theorem and performing a little bit of algebra reveals that the Bayes classifier assigns an observation $X = x$ to the class for which

$$\delta_k(x) = x^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + log\pi_k$$

is largest.



n example with three classes. The observations from each class are drawn from a multivariate Gaussian distribution with p = 2, with a class-specific mean vector and a common covariance matrix. Left: Ellipses that contain 95% of the probability for each of the three classes are shown. The dashed lines are the Bayes decision boundaries. Right: 20 observations were generated from each class, and the corresponding LDA decision boundaries are indicated using solid black lines. The Bayes decision boundaries are once again shown as dashed lines.

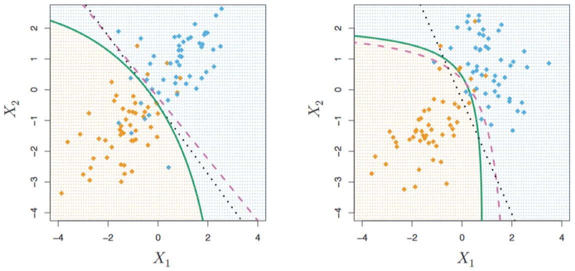
3 Quadratic Discriminant Analysis

In LDA, that assumes that the observations within each class are drawn from a multivariate Gaussian distribution with a class- specific mean vector and a covariance matrix that is common to all K classes, *Quadratic discriminant analysis* (QDA) provides an alternative quadratic discriminant analysis approach. Like LDA, the QDA classifier results from assuming that the observations from each class are drawn from a Gaussian distribution, and plugging estimates for the parameters into Bayes' theorem in order to perform prediction. However, unlike LDA, QDA assumes that each class has its own covariance matrix. That means it assumes that an observation from the kth class is of the form $X \sim (\mu_k, \Sigma_k)$ where Σ_K is the covariance matrix for kth class. Under this assumption, the Bayes classifier assigns an observation $X = x$ to the class for which:

$$\begin{aligned}\delta_k(x) &= -\frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) - \frac{1}{2}\log|\Sigma_k| + \log\pi_k \\ &= -\frac{1}{2}x^T\Sigma_k^{-1}x + x^T\Sigma_k^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma_k^{-1}\mu_k - \frac{1}{2}\log|\Sigma_k| + \log\pi_k\end{aligned}$$

is largest

So the QDA classifier involves plugging estimates for Σ_k, μ_k,and π_k into $\delta_k(x)$, and then assigning an observation X = x to the class for which this quantity is largest. Unlike in LDA $\delta_k(x)$, the quantity x appears as a quadratic function in QDA $\delta_k(x)$. This is where QDA gets its name.



Left: The Bayes (purple dashed), LDA (black dotted), and QDA (green solid) decision boundaries for a two-class problem with $\Sigma_1 = \Sigma_2$. The shading indicates the QDA decision rule. Since the Bayes decision boundary is linear, it is more accurately approximated by LDA than by QDA. Right: Details are as given in the left-hand panel, except that $\Sigma_1 \neq \Sigma_2$. Since the Bayes decision boundary is non-linear, it is more accurately approximated by QDA than by LDA.