General Formulation for LDA

- Fisher's LD maximizes between-class scatter while minimizing within-class scatter.

- LDA assumes Gaussian distribution and identical covariance matrices for groups.
  - The general formulation of LDA is equivalent to Fisher's LD for two groups.

Mathematical Formulation

- Let $\mathbf{X}_i = (X_{i1}, X_{i2}, \ldots X_{ip})'$ denote the $p-$dimensional vector of obs.
- Let π_k denote the overall or *prior* probability that a randomly chosen observation comes from the k-th class.
- Let $f_k(\mathbf{X}) := P(\mathbf{X}|Y = k)$ denote the *density function* of $\mathbf{X}$ for an observation that comes from the k-th class.
- Then Bayes rule states that

$$P(Y = k|\mathbf{X} = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{\ell=1}^{K} \pi_\ell f_\ell(\mathbf{x})}$$

- $p_k(\mathbf{x}) := P(Y = k|\mathbf{X} = \mathbf{x})$ is the so-called *posterior* probability that an observation $\mathbf{X} = \mathbf{x}$ belong to the k-th class, **given** the predictor value for that observation.

Bayes classifier

- A *Bayes classifier* is a rule that assigns an observation $\mathbf{x}$ to the class with largest $p_k(\mathbf{x})$.
- The problem is that it is difficult to estimate f_k.
- Three classifiers are suggested to approximates the Bayes classifier with different estimates of f_k:
  - linear discriminant analysis (LDA)
  - quadratic discriminant analysis (QDA)
  - naive Bayes

LDA

- Assumption: $\mathbf{x}_i \sim N_p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma})$ if $\mathbf{x}_i$ belongs to the kth class, where $\boldsymbol{\mu}_k$ is class-specific mean and $\boldsymbol{\Sigma}$ is a covariance matrix that is common for all K classes.

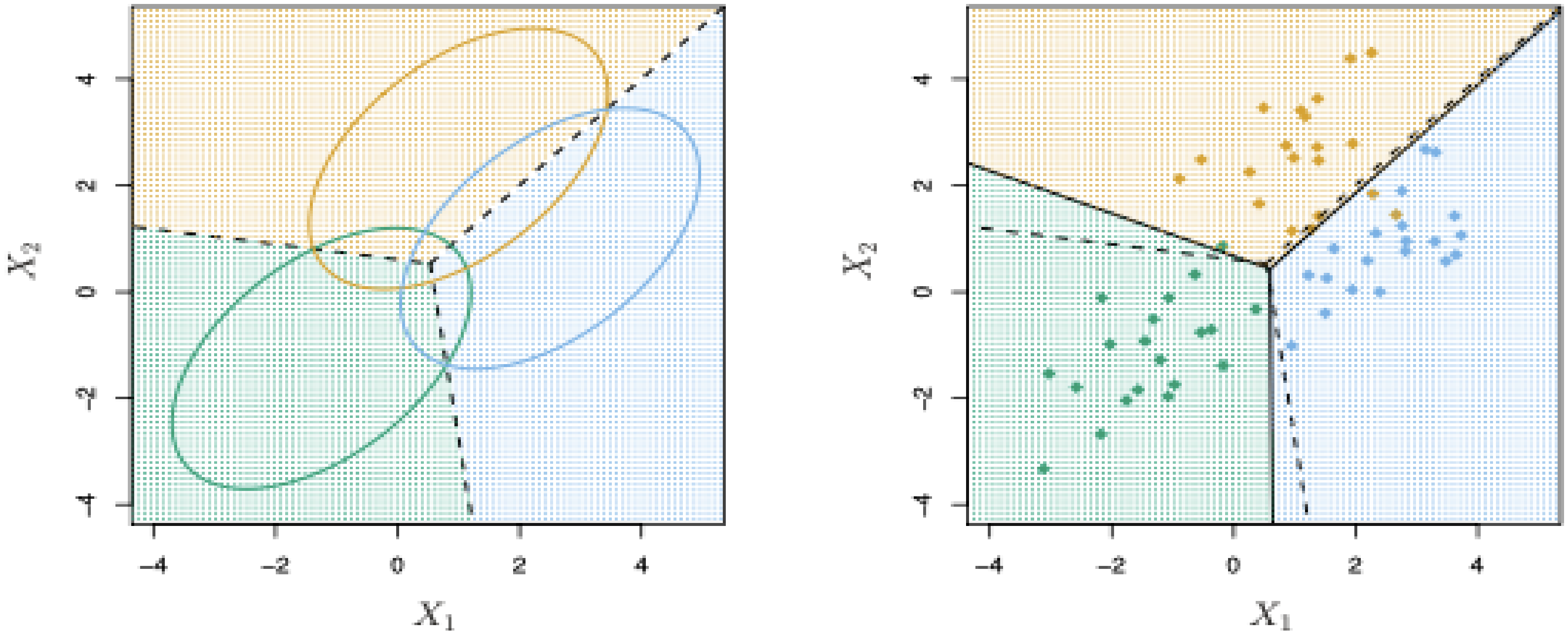
- LDA classifier assigns an observation $\mathbf{X} = \mathbf{x}$ to the class for which

$$\delta_k(\mathbf{x}) = \mathbf{x}^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \boldsymbol{\mu}_k^\top \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \log \pi_k$$

  is largest.

- $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}$ will be replaced by their sample estimates.

Example



- Left panel: Ellipses that contain the 95% of the probability for each of the three classes.
- Right panel: 20 observations were generated from each class, and the corresponding LDA decision boundaries are indicated using solid black lines.
- The dashed lines indicates the Bayes classifier (when the truth is known).

Example: Default Dataset

- We want to predict whether or not an individual will default on the basis of credit card balance and student status.
- LDA is fitted to 1000 training samples.

	default	student	balance	income
1	No	No	729.52650	44361.625
2	No	Yes	817.18041	12106.135
3	No	No	1073.54916	31767.139
4	No	No	529.25060	35704.494
5	No	No	785.65588	38463.496
6	No	Yes	919.58853	7491.559
7	No	No	825.51333	24905.227
8	No	Yes	808.66750	17600.451
9	No	No	1161.05785	37468.529
10	No	No	0.00000	29275.268
11	No	Yes	0.00000	21871.073
12	No	Yes	1220.58375	13268.562

```
require(ISLR2)
data("Default")

default.lda = lda(default ~ student + balance + income,
data=Default)
default.lda

Call:
lda(default ~ student + balance + income, data = Default)
```

```
Prior probabilities of groups:
    No    Yes
0.9667 0.0333

Group means:
    studentYes   balance   income
No   0.2914037  803.9438 33566.17
Yes  0.3813814 1747.8217 32089.15

Coefficients of linear discriminants:
                     LD1
studentYes -1.746631e-01
balance     2.243541e-03
income      3.367310e-06
```

Confusion Matrix

The confusion matrix can be obtained as

```
table(Default$default, predict(default.lda)$class)

     No   Yes
  No  9645   22
  Yes  254   79
```

		True default status		
		No	Yes	Total
Predicted	No	9644	252	9896
default status	Yes	23	81	104
	Total	9667	333	10000

The Problem

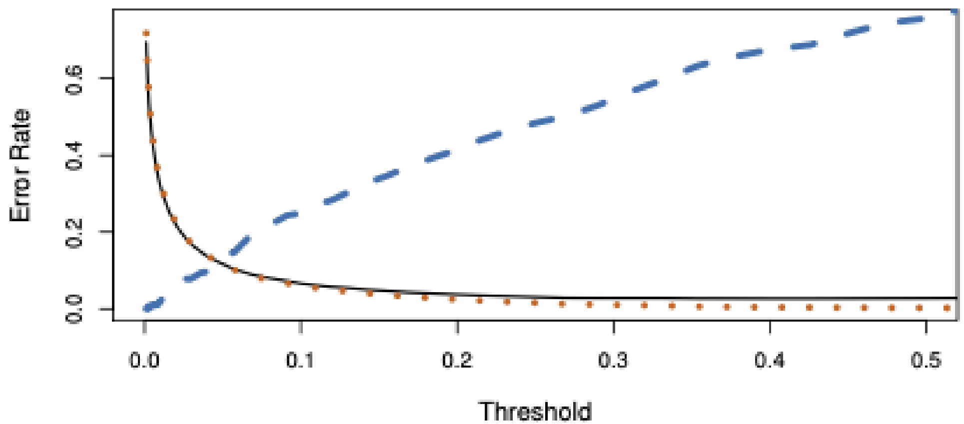
- Suppose a useless classifier that always predicts that an individual will not default.
- The resulting error rate is 3.33% - a very small error but useless result
- Note that of the 333 individuals who defaulted, LDA missed 252 (or 75.7%).

The Problem

- Sensitivity is the percentage of true defaulters that are identified. (24.3%)
- Specificity is the percentage of non-defaulters that are correctly identified. (99.8%)
- Why does LDA do such a poor job of classifying the customers who default?
- It's because LDA is trying to approximate the Bayes classifier which has the lowest *total* error rate out of all classifiers.
  - Some misclassifications will result from incorrectly assigning a customer who does not default to the default class;
  - Others will result from incorrectly assigning a customer who defaults to the non-default class.

Revisiting the Bayes Classifier

$Pr(\text{default} = \text{Yes}|\mathbf{X} = \mathbf{x}) > 0.5 \rightarrow \mathbf{x}$ is classified as the "default" class



- x-axis varies the threshold from 0 to 0.5 in the Bayes classifier.
- The black solid line displays the overall error rate.
- The blue dashed line represents the fraction of defaulting customers that are incorrectly classified
- the orange dotted line indicates the fraction of errors among the non-defaulting customers

Measures of Errors

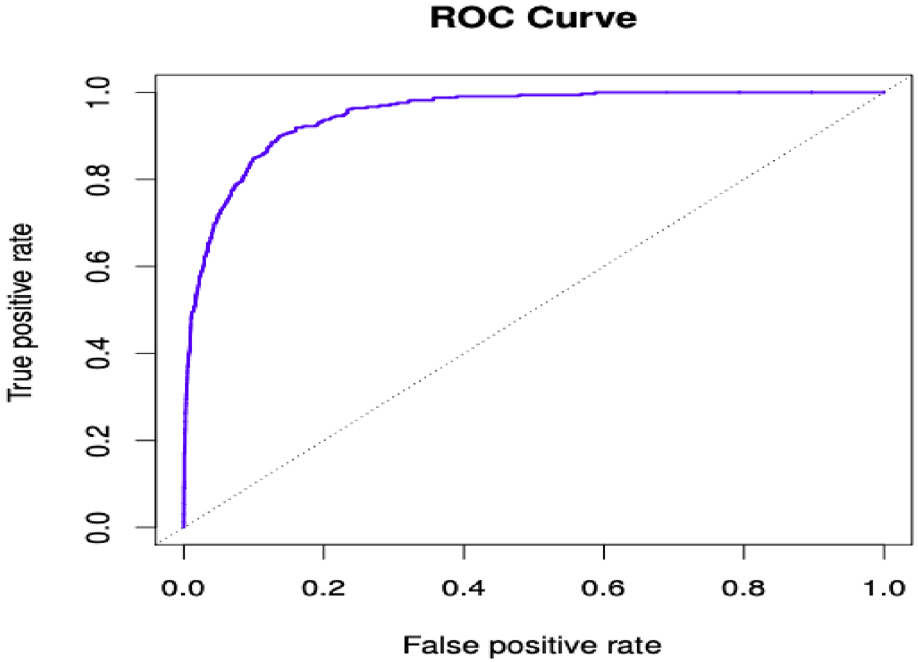
		True class		
		− or Null	+ or Non-null	Total
Predicted class	− or Null	True Neg. (TN)	False Neg. (FN)	N^*
	+ or Non-null	False Pos. (FP)	True Pos. (TP)	P^*
	Total	N	P	

Name	Definition	Synonyms
False Pos. rate	FP/N	Type I error, 1−Specificity
True Pos. rate	TP/P	1−Type II error, power, sensitivity, recall
Pos. Pred. value	TP/P^*	Precision, 1−false discovery proportion
Neg. Pred. value	TN/N^*	

- The ROC (receiver operating characteristics) curve is a popular graphic for simultaneously displaying the two types of errors for all possible thresholds.

ROC

An ideal ROC curve will hug the top left corner, so the larger the *area under the (ROC) curve* (AUC) the better the classifier.



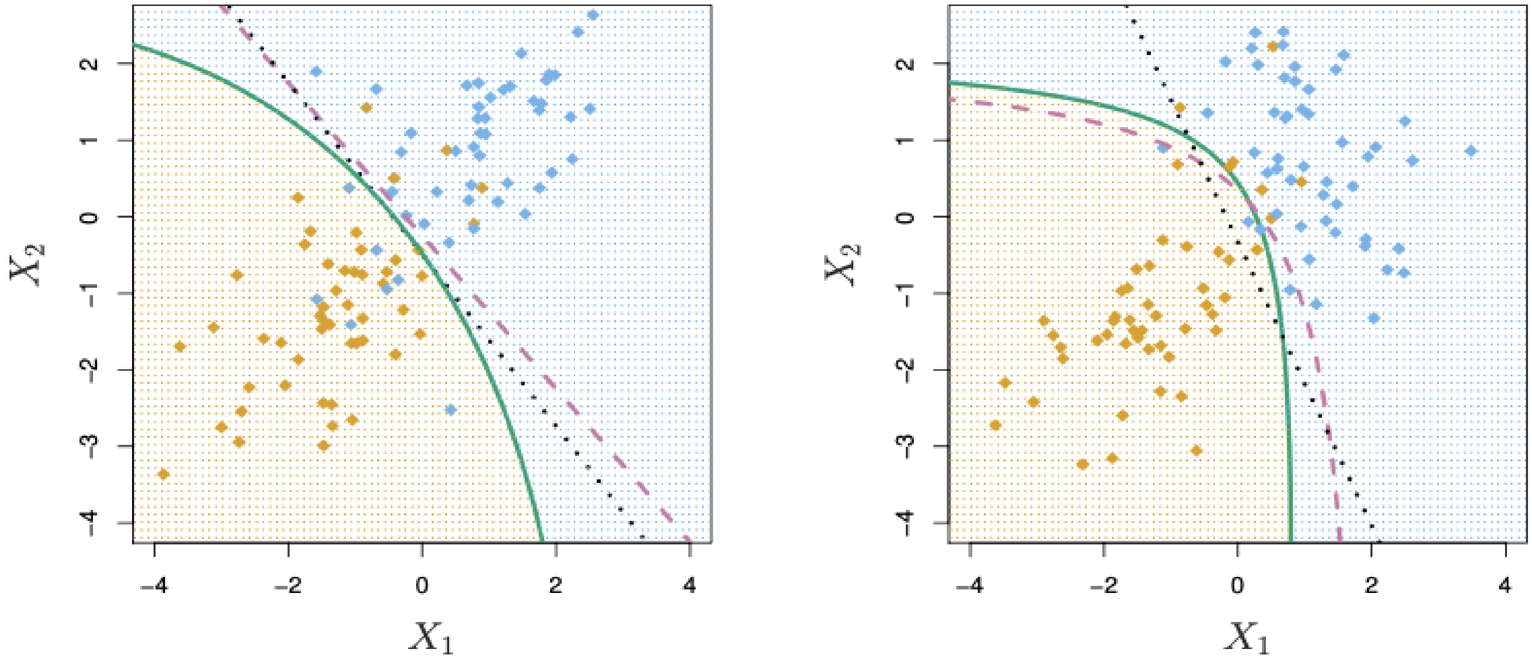
Quadratic Discriminant Analysis (QDA)

- Assumption: $\mathbf{X} \sim N_p(\boldsymbol{\mu}_k, \Sigma_k)$ (Gaussianity, unequal covariance)
- QDA classifier assigns an observation $\mathbf{X} = \mathbf{x}$ to the class for which

$$\delta_k(\mathbf{x}) = -\frac{1}{2}\mathbf{x}^\top \Sigma_k^{-1}\mathbf{x} + \mathbf{x}^\top \Sigma_k^{-1}\boldsymbol{\mu}_k - \frac{1}{2}\boldsymbol{\mu}_k^\top \Sigma_k^{-1}\boldsymbol{\mu}_k - \frac{1}{2}\log|\Sigma_k| + \log \pi_k$$

  is largest.
- All population parameters are replaced by their sample estimates.

Example



- Left: The Bayes (purple dashed), LDA (black dotted), and QDA (green solid) decision boundaries for a two-class problem with $\Sigma_1 = \Sigma_2$.
- Same as in the left except that $\Sigma_1 \neq \Sigma_2$.

QDA in R

```
default.qda = qda(default ~ student + balance + income,
data=Default)

Call:
qda(default ~ student + balance + income, data = Default)

Prior probabilities of groups:
    No    Yes
0.9667 0.0333

Group means:
    studentYes   balance   income
No   0.2914037  803.9438 33566.17
Yes  0.3813814 1747.8217 32089.15
```

Naive Bayes Classifier

$$P(Y = k|\mathbf{X} = \mathbf{x}) = \frac{\pi_k \times f_{k1}(x_1) \times f_{k2}(x_2) \times \cdots \times f_{kp}(x_p)}{\sum_{\ell=1}^{K} \pi_\ell \times f_{\ell 1}(x_1) \times f_{\ell 2}(x_2) \times \cdots \times f_{\ell p}(x_p)}$$

for $k = 1, \ldots, K$.

- Each f_{kj} can be any distribution (not necessarily normally distributed).
- It introduces some bias due to independence assumption, but reduces variance.
- It often performs well in practice.
- Use the package `naivebayes`:

  `naive_bayes`