Statistical Learning: Chapter 4
Classification

4.1 Introduction

Supervised learning with a categorical (Qualitative) response

Notation:
- Feature vector X,
- qualitative response Y, taking values in $\mathcal{C}$
- We want to build classifier C(X) that uses X to predict class label for Y

drug disc $\mathcal{C} = \{active, inactive\}$

Often we may be just as interested in estimating **probability** of each class,
i.e. P(Y=y | X=X) for y in $\mathcal{C}$

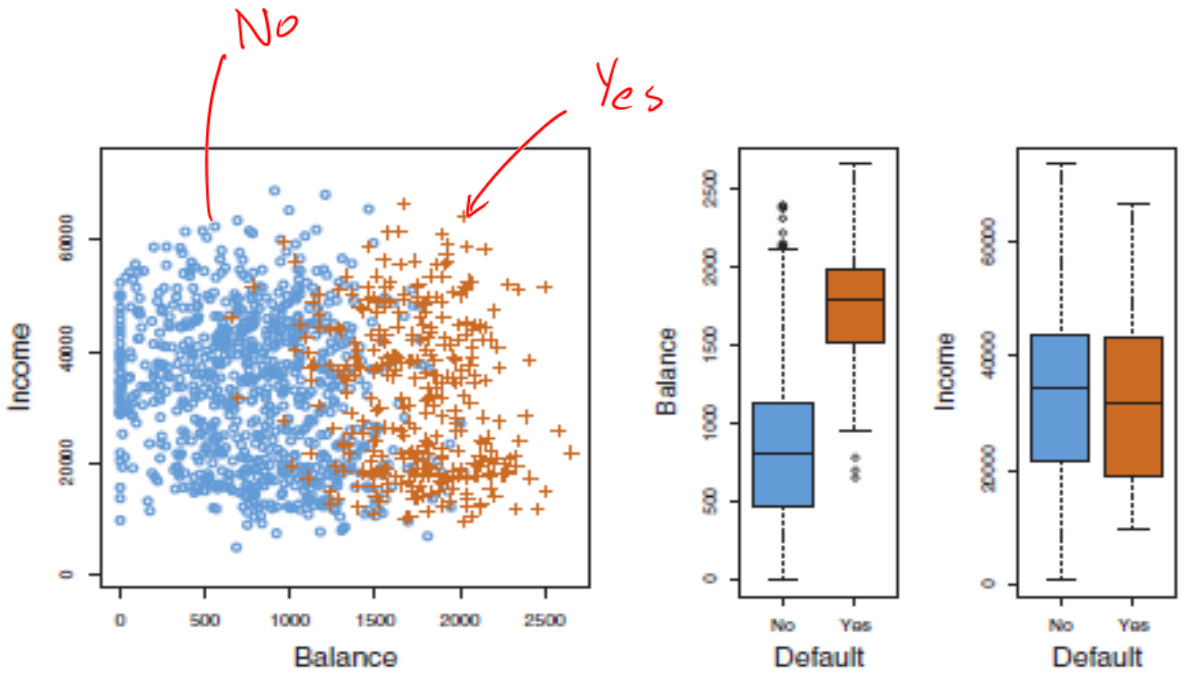
Example: Credit card default, we may be more interested in predicting the probabilty of a default than classifying individuals as default or not.

Simulated dataset used in book for illustration: "Default" in ISLR library

X = (student, balance, income)
Y = default (taking values Yes and No)
n=10,000 values

```
> head(Default)
  default student   balance    income
1      No      No  729.5265 44361.625
2      No     Yes  817.1804 12106.135
3      No      No 1073.5492 31767.139
4      No      No  529.2506 35704.494
> table(Default$default)

  No  Yes
9667  333
```



Some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R" (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani

4.2 Why not just use linear regression?

It is technically possible, for binary data:

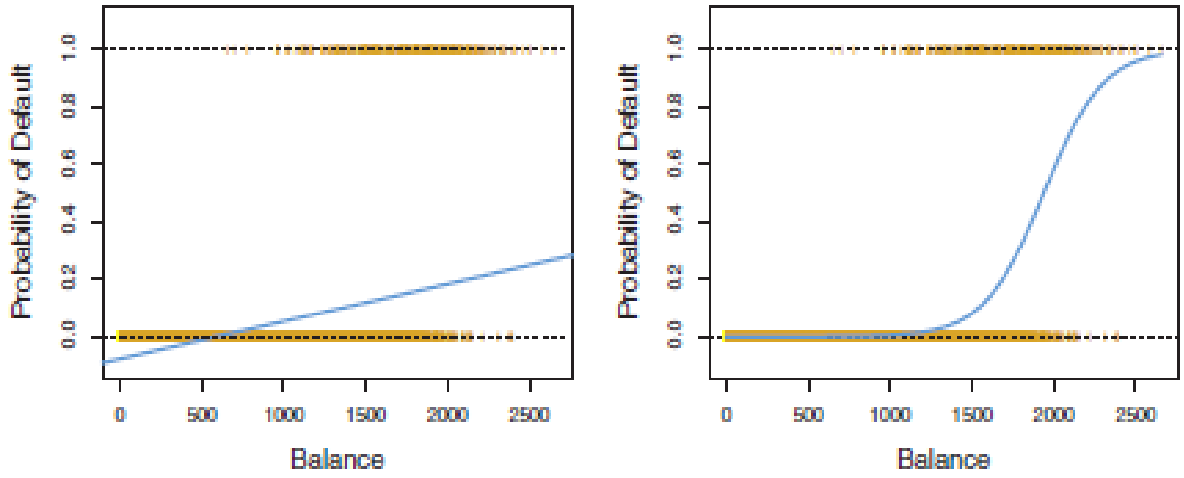
Example of how we might do it for binary data:

Convert Y from categorical to numeric:
Yes = 1, No = 0

Least squares Linear regression estimates E(Y|X) and for a binary variable Y, we know
E(Y) = 0*Pr(Y=0) + 1*Pr(Y=1) = Pr(Y=1)

We could always produce a classification by saying C(X) = "yes" if our predicted Y is > 0.5

But there are problems, even for the binary case:
- We can get predictions that are < 0 or > 1 (linear regression case is on the left of Fig 4.2):



So if we want probability estimates, a linear function is not a good idea.

Other problems: With more than 2 classes, the conversion to 0/1 is problematic, since it implies an ordering of the different classes:

Movie categories
$\mathcal{C}$ = { Drama, Action, Comedy, Documentary }

Choosing
Y = 1 if Drama, 2 if Action, 3 if Comedy, 4 if Documentary
would imply an ordering of the categories, and give us different results in a linear regression than

Y = 1 if Action, 2 if Comedy, 3 if Drama, 4 if Documentary

Also:
Normal error model is wrong here (binomial or multinomial)

In the next two sections we will learn two methods for classiciation,
Logistic regression and **Linear Discriminant Analysis**

4.3 Logistic Regression

4.3.1 The logistic model with one X

We will focus mostly on the **binary** case, in which the two classes are labelled as $\mathcal{C} = \{0,1\}$.
This is NOT the same as fitting a least squares regression with 0 / 1 response.

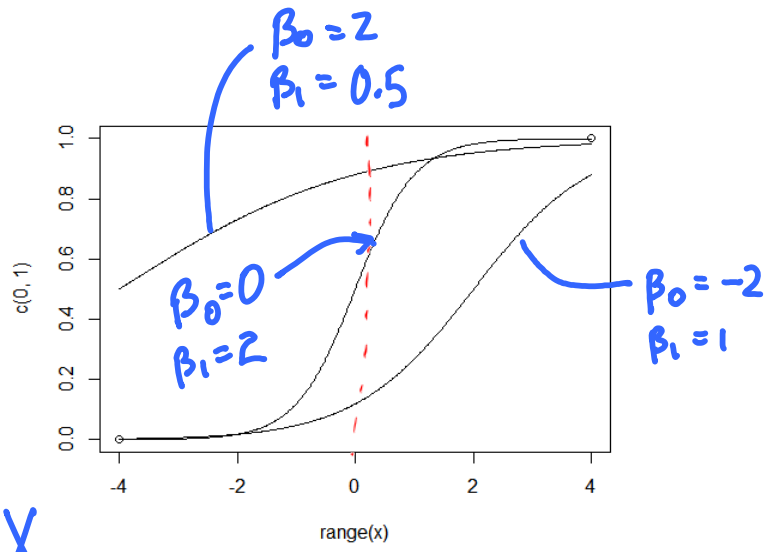
First, consider the case of a single predictor, X, which we assume takes numeric values.

Logistic regression writes p(X) = Pr(Y=1|X), and uses the form

$$(4.2) \qquad p(x) = \frac{e^{\beta_0 + \beta_1 x}}{1 + e^{\beta_0 + \beta_1 x}} \qquad e = 2.71828\ldots$$

What does this look like?

One curve is (beta0,beta1) = (-2,1)
(0,2), or (2,0.5)



We can re-express (4.2) as:

$$(4.4) \qquad \log\left(\frac{p(x)}{1-p(x)}\right) = \beta_0 + \beta_1 X$$

Log odds or "logit" function

If $\beta_0 = 0$, $P(Y=1 \mid X=0) = \frac{1}{2}$

So a unit change in X results in a change of beta_1 in the log-odds.
- Not as nice an interpretation as linear regression
- But the sign of the coefficients still gives an indication of the direction of the effect.
- From plot of the logit function:beta_0 determines p(0) = exp(beta0) / (1+exp(beta0))

Thought provoking question: If our model is

log(p(x) / (1-p(x)) = beta0 + beta1 x,

where's the error term? It's not on the right of the formula

linear reg $Y = \beta_0 + \beta_1 X + \varepsilon$

Y is Bernoulli $E(Y) = p(x)$ $Var(Y) = p(x)(1-p(x))$

Y is Normal $E(Y) = \beta_0 + \beta_1 X$, $Var(Y) = \sigma^2$

4.3.2 Estimating the regression coefficients in a logistic regression

We use maximum likelihood to estimate the parameters.

Likelihood function = probability of the observed training data.

Example: (n = 5)

```
Y       X   likelihood =
0     1.4         P(Y)  = Pr(Y1=0)*Pr(Y2=0)*Pr(Y3=1)*Pr(Y4=0)*Pr(Y5=1)
0     2.4               = [1-p(1.4)] * [1-p(2.4)] * p(3.5) * [1-p(3.6)] * p(5.0)
1     3.5
0     3.6
1     5.0
```

In the above, expressions like p(1.4) are given by equation (4.2) for p(x) on previous page. These depend on beta0 and beta1.

The general form of the likelihood function is

$$\ell(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i':y_{i'}=0} (1 - p(x_{i'})). \qquad (4.5)$$

Maximum likelihood estimation : choose beta0 and beta1 to maxiize the likelihood function.
- That is, what values of beta0 and beta1 make the observed data most probable?

This is easily done in R, using the "glm" function.

	Coefficient	Std. Error	Z-statistic	P-value
Intercept	β_0 -10.6513	0.3612	-29.5	< 0.0001
balance	β_1 0.0055	0.0002	24.9	< 0.0001

Both different from 0.

For the "Default" dataset the above estimates indicate what?

4.3.3 Making Predictions

Default probability for an individual with a balance of $1,000 is

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 X}}{1+e^{\hat{\beta}_0+\hat{\beta}_1 X}} = \frac{e^{-10.6513+0.0055\times 1{,}000}}{1+e^{-10.6513+0.0055\times 1{,}000}} = 0.00576,$$

For a balance of $2,000:

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 X}}{1+e^{\hat{\beta}_0+\hat{\beta}_1 X}} = \frac{e^{-10.6513+0.0055\times 2000}}{1+e^{-10.6513+0.0055\times 2000}} = 0.586$$

We can also make predictions for the effect of being a student on defaulting on your loan. First fit the logistic regression model:

	Coefficient	Std. error	Z-statistic	P-value
Intercept	−3.5041	0.0707	−49.55	<0.0001
student[Yes]	0.4049	0.1150	3.52	0.0004

$$\widehat{\Pr}(\texttt{default=Yes}|\texttt{student=Yes}) = \frac{e^{-3.5041+0.4049\times 1}}{1+e^{-3.5041+0.4049\times 1}} = 0.0431,$$

$$\widehat{\Pr}(\texttt{default=Yes}|\texttt{student=No}) = \frac{e^{-3.5041+0.4049\times 0}}{1+e^{-3.5041+0.4049\times 0}} = 0.0292.$$

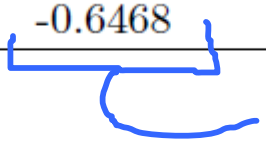
It appears that students are more likely to default on a loan than non-students. And the effect is significant (small p-value).

4.3.4. Multiple Logistic Regression
(logistic regression with more than one variable)

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

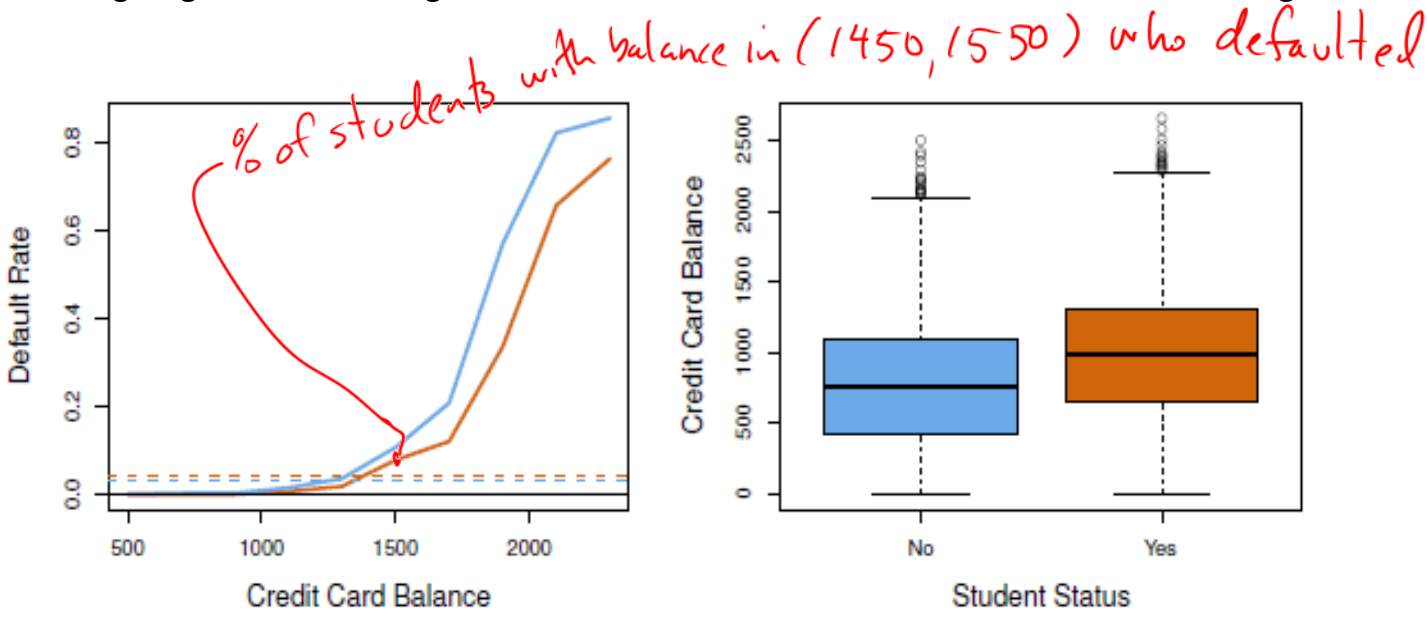
$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

	Coefficient	Std. Error	Z-statistic	P-value
Intercept	-10.8690	0.4923	-22.08	< 0.0001
balance	0.0057	0.0002	24.74	< 0.0001
income	0.0030	0.0082	0.37	0.7115
student[Yes]	-0.6468	0.2362	-2.74	0.0062



What?? Negative??

What's going on with the negative "student" coefficient? ... the idea of "Confounding"



Left: Average default rates for nonstudents (blue) and students (orange)

Right: Boxplots of credit card balance for nonstudents and students.

Students tend to have higher balances than non-students, and default rates increase with credit card balance. So students tend to default more because they have higher balances.

The two predictors "student" and "balance" are confounded.

A simple linear regression with only "student" ignores balance, and since students have a higher balance, the estimated effect is that being a student increases the chance of defaulting.

Note that for each level of balance (in left plot), students have a lower default rate than nonstudents.

The multiple logistic regression is identifying the correct interpretation. That is, after adjusting for the effect of credit card balance, students are less likely to default.

4.3.5 Logistic regression for > 2 response classes

Note R^2 not used in logistic. Instead we look at correct / incorrect classification

Logistic regression is easily generalized to more than two response classes.

One version, in the "glmnet" package in R, uses a linear function for each class:

$$\Pr(Y = k|X) = \frac{e^{\beta_{0k}+\beta_{1k}X_1+\ldots+\beta_{pk}X_p}}{\sum_{\ell=1}^{K} e^{\beta_{0\ell}+\beta_{1\ell}X_1+\ldots+\beta_{p\ell}X_p}}$$

k = 1, ..., K

# of classes

This is also referred to as multinomial regression

4.4 Linear Discriminant Analysis (LDA)
Another way to estimate P(Y|X) using linear function.

Different from, but related to logistic regression.

In some settings (well-separated classes, or small samples in which the within class-distribution of Xs is roughlyt normal), logistic regression estimates can be unstable. LDA can be more stable.

LDA is also more popular than logistic methods for multiclass problems.

Key idea:
- Model the distribution of X within each class (i.e. model P(X|Y)).
- Then use Bayes' theorem to turn this around and get an estimate of P(Y|X)

4.4.1 Bayes Theorem:

$$\Pr(Y = k|X = x) = \frac{\Pr(X = x|Y = k) \cdot \Pr(Y = k)}{\Pr(X = x)}$$

$$P(Y=k|X=x)\,P(X=x) = P(X=x|Y=k)\,P(Y=k)$$

Definitions:

$f_k(x) = Pr(X=x|Y=k)$ = probability density function of x for an observation from class k

$\pi_k = Pr(Y=k)$ = proportion of observations for which $Y=k$

"prior probabilities"

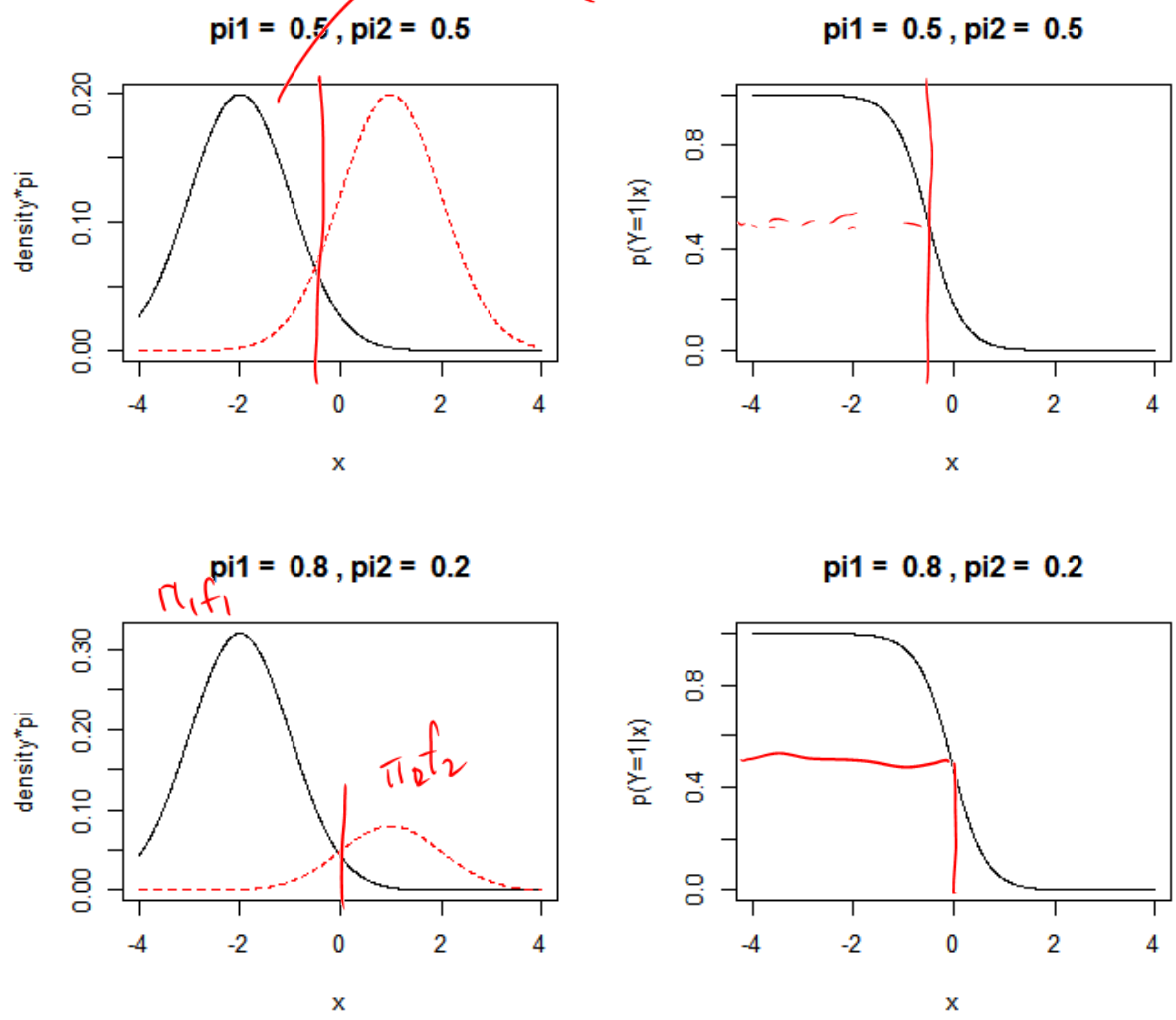
Bayes' Theorem is written in a slightly different form:

$$\underbrace{\Pr(Y = k|X = x)}_{p_k(x)} = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}. \qquad (4.10)$$

We will typically use a normal distribution for f_k(x)

4.4.2 LDA for p=1 dimension

Example:
- Both rows: x in one class is N(-2,1), the other N(1,1)
- Top row: pi1 = pi2 = 0.5
- Bottom row: pi1 = .8, pi2 = .2



Mathematical details:

The Gaussian density has the form

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

μ_k = mean in class k

σ_k^2 = variance " " "

NOTE: we assume for now that variances in all classes are equal. $\sigma_k^2 = \sigma \quad \forall k$

Plugging this density into Bayes formula gives the expression below for P(Y=k | X=x):

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

$f_k(x)$

To classify at the value X=x, we must calculate p_k(x) for all k=1, 2, ..., K classes, and then choose the class with the largest p_k(x).

If we take logs of the above expression, and simplify, it turns out that p_k(x) is largest whenever the following discriminant score is largest (it is also calculated at a specific x for all k).

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

Not that this discriminant score is a **linear** function of x. This is where the "linear" in LDA comes from.

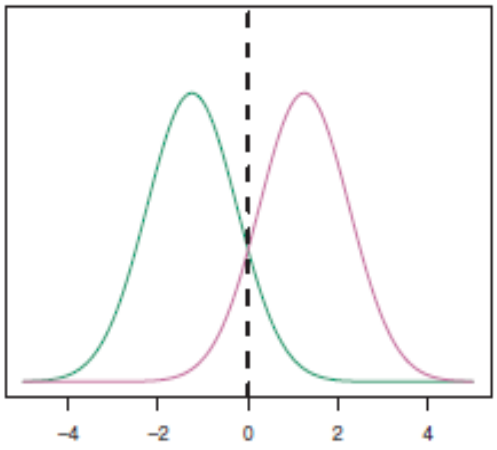
If we have K=2 classes and equal class probabilities pi_1 = pi_2 = 0.5, it's possible to show that the decision boundary is at the midpoint of the class means:

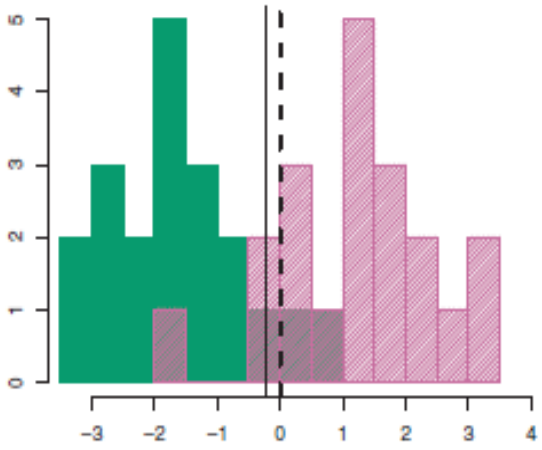
$$x = \frac{\mu_1 + \mu_2}{2}$$

Example with class means 1.5 and -1.5, and equal class probabilities:

Left: Theoretical model, with ideal boundary

Right: 20 observations sampled in each class, solid line = estimated decision boundary





Estimation:

The above derivations assumed we knew the population means, the common variance, and the prior probabilities of each class.
In practice, we estimate these from the training data and "plug in" the values to the above formulas.

$$\hat{\pi}_k = \frac{n_k}{n}$$

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:\, y_i = k} x_i$$

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

$$= \sum_{k=1}^{K} \frac{n_k - 1}{n-K} \cdot \hat{\sigma}_k^2$$

$$\hat{\sigma}_k^2 = \frac{1}{n_k - 1} \sum_{i:\, y_i = k} (x_i - \hat{\mu}_k)^2$$

Above formula is the usual sample variance for x values belonging to class k.

Note: These estimates are fast to compute, even for very large samples.

Estimation of an LDA model is generally faster than a logistic regression, which uses iterative algorithms to fine the maximum likelihood estimates.

4.4.3 LDA with p > 1 dimensions

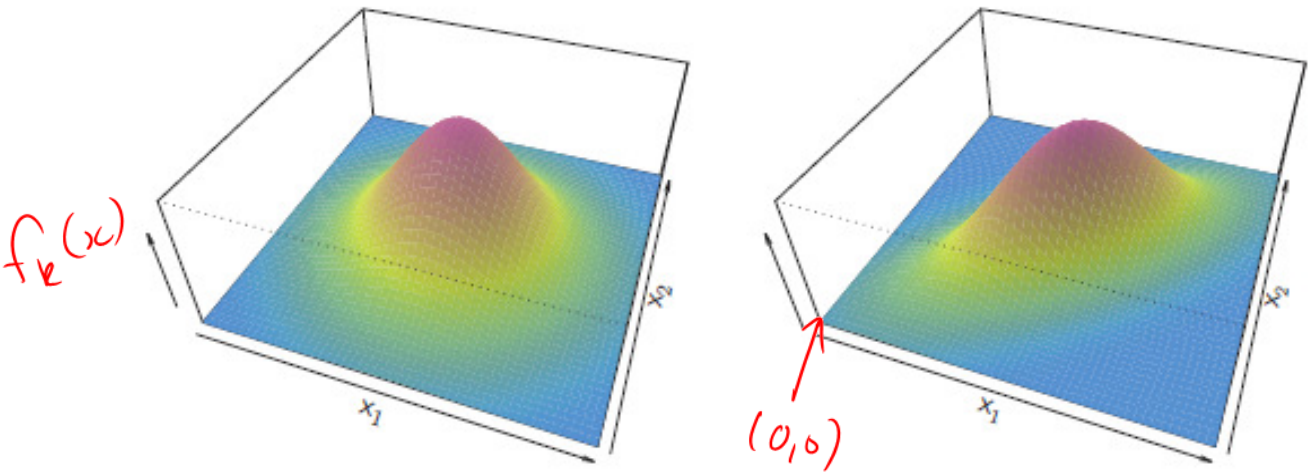


FIGURE 4.5. *Two multivariate Gaussian density functions are shown, with* $p = 2$. Left: *The two predictors are uncorrelated.* Right: *The two variables have a correlation of* 0.7.

The univariate normal densities in the previous section are replaced by multivariate normal density functions. They have the form:

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

$\Sigma_{jk} = \Sigma_{kj} = Cov(X_j, X_k)$

Σ = covariance matrix $(p \times p)$

μ = mean vector $(p \times 1)$

$\Sigma_{jj} = Var(X_j)$ (j^{th} diag element)

j^{th} element of p-vector X.

The covariance matrix determines the shape of the density. It also determines the correlations between the different elements of the vector x. The plot on the previous page shows two cases, one where the covariance matrix is the identity, the other where it has diagonal elements of 1 and off-diagnoal elements of 0.7.

Diagonal elements of the covariance matrix are the variances of the elements of x.

Calculations using the above density, similar to what we did for p=1 dimension, give the discriminant function

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

$x^T a_k + b_k$ (a_k is $p \times 1$, b_k is 1×1)

Σ = common covar for all classes

μ_k = mean for class k

This may look messy, but careful examination indicates it is still a linear function of the elements of the vector x

Illustration: K = 3 classes and p=2 dimensions

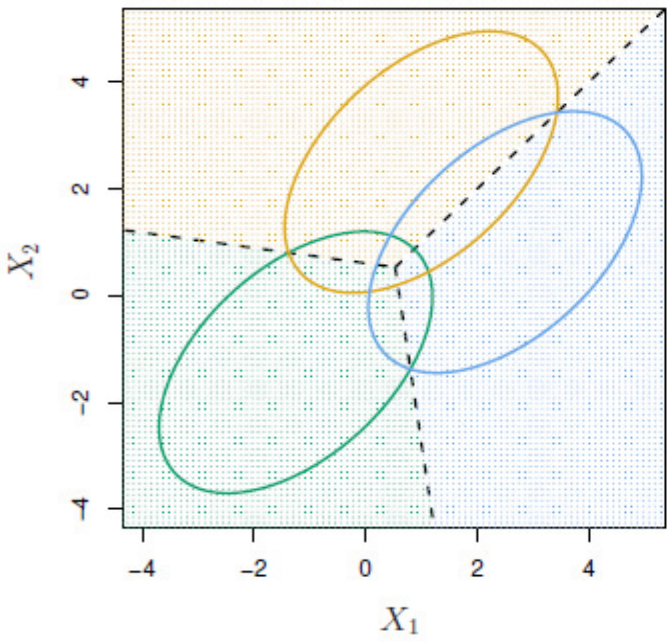
We assume pi1 = pi2 = pi3 = 1/3

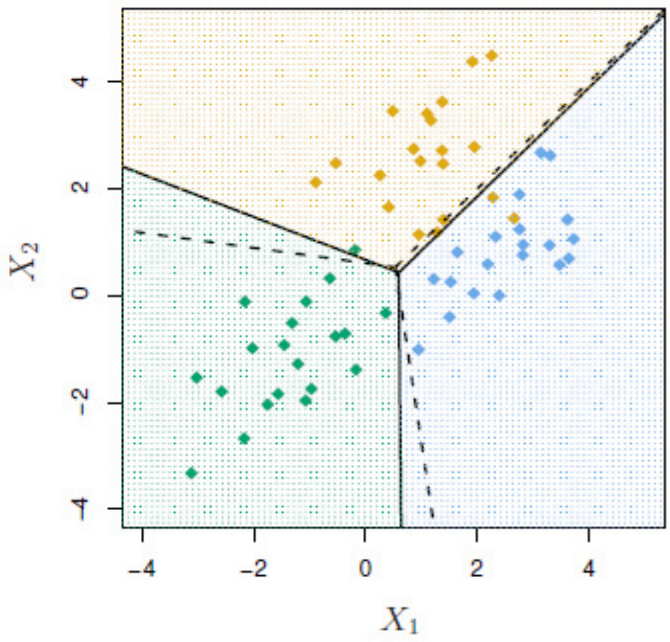
For plots on next page:
1. The dashed lines below are known as the Bayes decision boundaries. They come from the delta function above, and would give minimum possible error. Of course they are not known unless the mu's and Sigma are known.

2. The ellipses correspond to contours of equal height of the Multivariate Gaussian density function, and give an idea of the distribution of the data within each class. Notice how the shape, size and orientation of the ellipses are the same within each class.

3. In the right plot, samples of size 20 are simulated from each class. The color corresponds to the **actual** class from which they were simulated, so some points will fall outside the optimal decision boundaries (i.e. be misclassified).

Black line = estimated bounaries





Calculating class probabilities for LDA

Once we have estimates of the discriminant functions above (delta_k(x)), we can use these to calculate the probabilities of the various classes:

$$\widehat{\Pr}(Y = k|X = x) = \frac{e^{\hat{\delta}_k(x)}}{\sum_{l=1}^{K} e^{\hat{\delta}_l(x)}}.$$

The "estimated" delta_k functions require estimation of multivariate means and covariance matrices. Formulas are similar to 1D case.

Aside: Inferential statements, such as Confidence Intervals and Hypothesis Tests, are not as common in LDA as they are in logistic regression. This is partly because logistic regression is based on a model for random Y with fixed X, while LDA is based on a model for random X with fixed Y.

Credit card default example:
- Use two predictors (balance and student)
- LDA gives the following results, presented as a "confusion matrix" comparing actual and predicted class:

		True Default class		
		No	Yes	Total
Predicted class	No	9644	252	9896
	Yes	23	81	104
	Total	9667	333	10000

Overall misclassification rate = (252+23) / 10000 = 275/10000 = 2.75%

BUT, if we predicted "everyone will not default" our misclassification rate would be....

333/10000 = 3.33%

Of the true No's we make 23/9667 = 0.2% errors (false positive rate)
Of the true Yes's, we make 252/333 = 75.5% errors!!! (false negative rate)

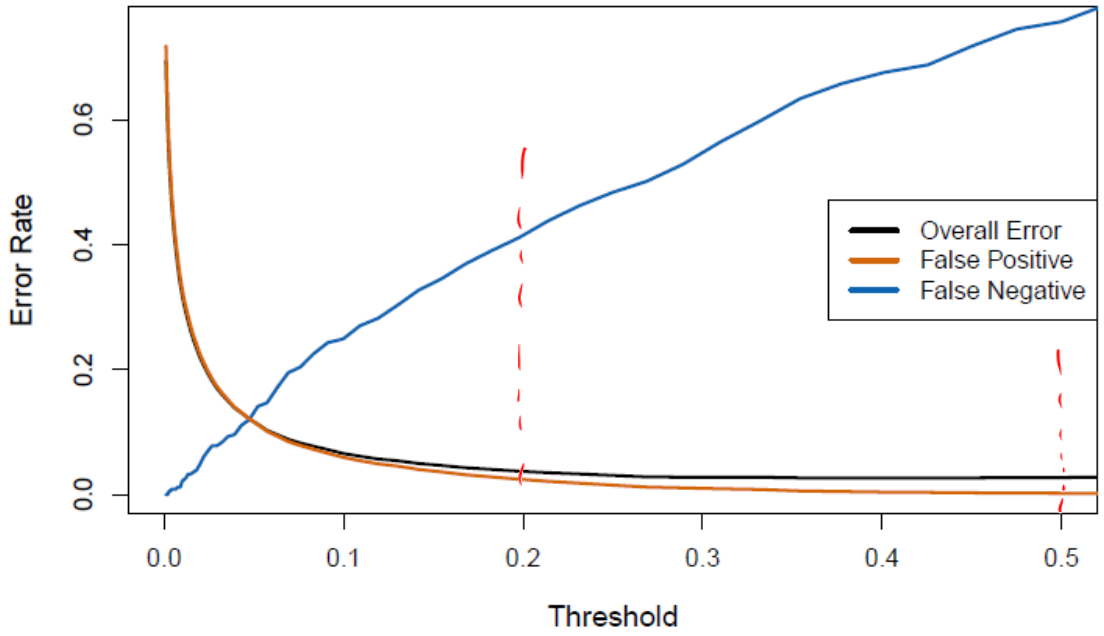
Problem: we predicted classes using a threshold of 0.5 for the predicted class probability.
- This minimize overall misclassification rate
- But it is doesn't give accurate predictions for the "yes" class.

Solution: use different threshold value (below: threshold = 0.2)

		True default status		
		No	Yes	Total
Predicted default status	No	9,432	138	9,570
	Yes	235	195	430
	Total	9,667	333	10,000

Our overall error rate has increased to (235 + 138) /10000 = 3.73%.
- False negative rate (i.e. error rate for "yes") = 138/333 = 41.4% (better than 75.5% before)
- False positive = 235/9667 = 2.4% (was 0.2%)

We could consider varying the threshold between 0 and 0.5, and look at how our error rates change

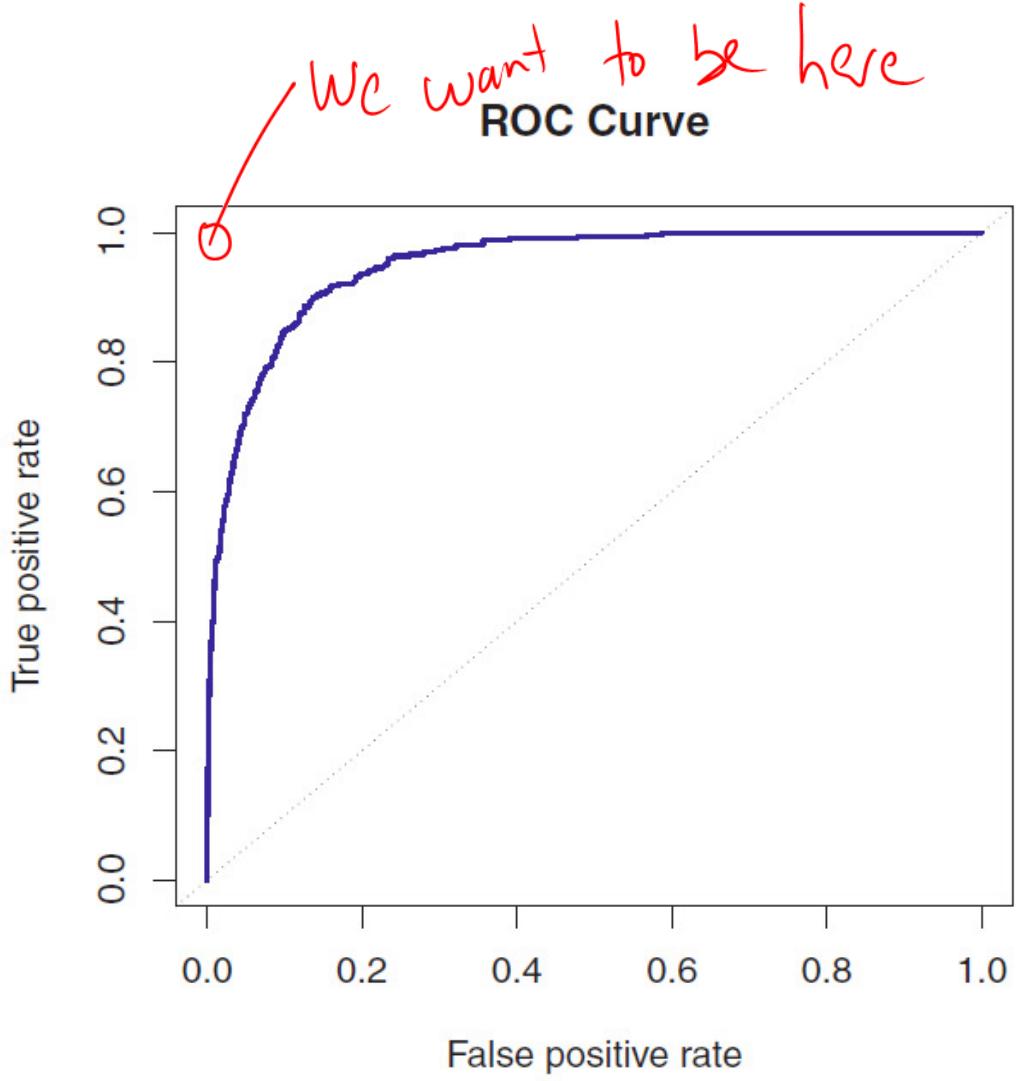


Another popular way of visualizing the tradeoff for different thresholds is the ROC curve
- We vary the threshold from 0 to 1.
- At each threshold value we get a false positive and a false negative rate (as in above plot)
- Calculate "true positive" rate = (1 - false negative) = fraction of defaulters correctly identified.
- Then we plot false positive vs. true positive
- Put another way, we plot values on the orange curve vs. 1-values on the blue curve
- So we want False positive = 0 and true positive = 1. This is the upper ~~right~~ left corner in the plot below.

We want to be here

True positive rate = proportion of defaulters correctly identified

this is
1-false negative rate



Area under the ROC curve (AUC) is often used as a performance measure. High AUC is good.

Dotted 45-degree line = result if "student" & "balance" had no relationship with Y

In this example, the results for logistic regression are nearly identical to LDA.

Note that the LDA model isn't completely sensible for this data.
- "Student" = 1 if student, and 0 if non-student.
- Why is this "not sensible"? What assumptions is LDA making about the Xs within each class?

↓ Normality

4.4.4 Quadratic Discriminant Analysis (QDA)
Recall that LDA assumes
- multivariate normal distribution within each class k
- each class has a different mean vector
- all classes have the same covariance matrix

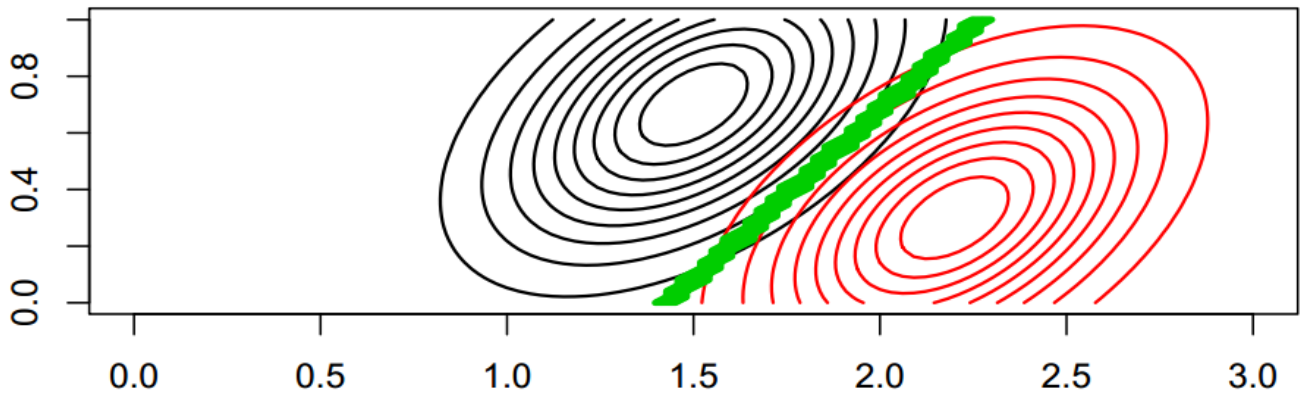
LDA: $X \sim MVN(\mu_k, \Sigma)$ given that $Y = k$

Quadratic discriminant analysis allows the covariance matrices to be different within each class:

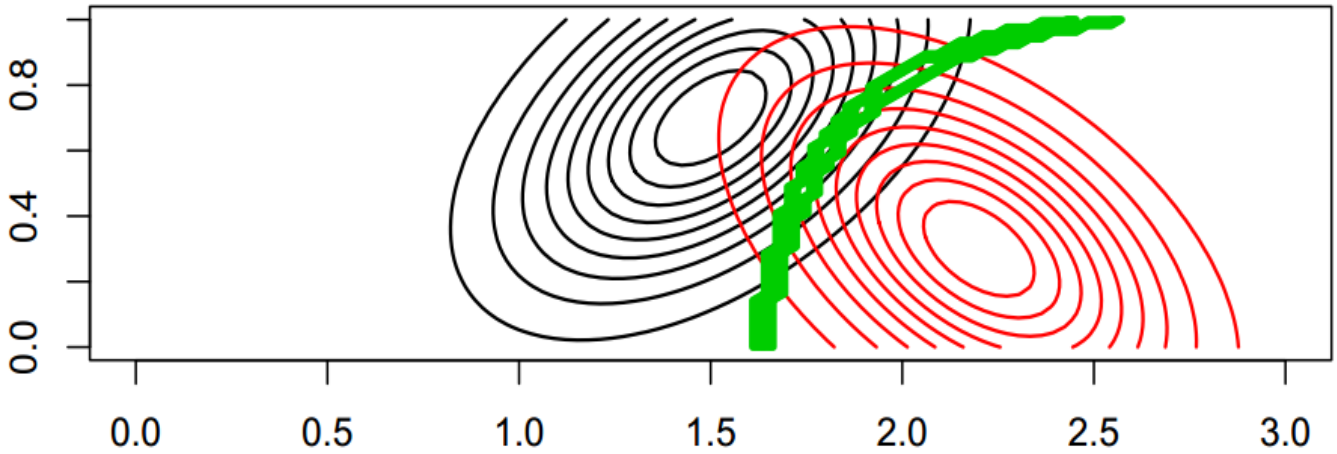
QDA $X \sim MVN(\mu_k, \Sigma_k)$ given that $Y = k$

LDA vs. QDA:

LDA:



QDA:



By allowing the covariance matrices to be different, we get decision boundaries that are nonlinear.

Another variation of LDA, not mentioned in the book is **Naive Bayes**
- Naive Bayes assumes that each class has a different covariance matrix, like QDA
- But it also assumes that the within-class covariance matrix is **diagonal**.
- That is, it assumes that conditional on class, the predictors are uncorrelated.
- This is a strong assumption
- The assumption can be useful when we have very high dimensional data (many Xs, large p)

QDA must estimate K Σ's, each with $p + \binom{p}{2}$ parameters

NB → p diagonals.

A naive Bayes model can also be used for non-normal data, or data with some normal and some non-normal (e.g. qualitative) variables:
- we just assume that within each class the pdf is a product of one-dimensional pdfs:

$$P(X \mid Y = k) = \prod_{j=1}^{p} f_{kj}(x_j)$$

$$X = \begin{bmatrix} x_1 \\ \vdots \\ x_j \\ \vdots \\ x_p \end{bmatrix}$$

each 1 dimensional pdf can be normal, non-normal, discrete, etc.