Data Mining
W4240 Section 001

Giovanni Motta

Columbia University, Department of Statistics

October 19, 2015

Logistic Regression

Recall from last time:

- have binary responses (0 or 1)
- $Y_i \sim Ber(p(x_i))$
- use a linear model for log odds:

$$\log\left(\frac{p(x_i)}{1-p(x_i)}\right) = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}$$

Can predict probability of $\{Y_i = 1\}$ as

$$p(x_i) = \frac{e^{\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}}}{1 + e^{\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}}}$$

Logistic Regression

Interpretation: one unit more of x_{ij} multiplies the probability of seeing $\{Y_i = 1\}$ by e^{β_j}.

Problems:

- estimator may not converge (common with colinear covariates)
- estimator may be unstable when classes are linearly separable

Logistic Regression

Example: is smoking associated with cancer?

Doll and Hill (1950)

- interviewed lung cancer patients newly admitted to hospitals
- randomly interviewed other newly admitted patients

	Control	**Cancer**
Smokers	1296	1350
Nonsmokers	61	7

$$\log\left(\frac{\pi(x_i)}{1-\pi(x_i)}\right) = \beta_0 + \beta_1 \mathbf{1}_{\{\text{patient } i \text{ is a smoker}\}}$$

Parameter	**Value**	**p-value**
β_0	-2.1650	$5.78e-08$
β_1	2.2058	$3.76e-08$

Logistic Regression

Doll and Hill (1954)

- surveyed smoking habits of 30,000 British doctors
- tracked deaths over 10 years

Cigarettes/day	**0**	**1-14**	**15-24**	**5+**
% of sample	10	40	30	20
% of lung cancer deaths	2	16	40	42

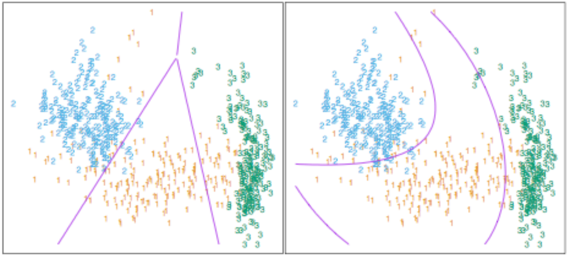
$$\log\left(\frac{p(x_i)}{1-p(x_i)}\right) = \beta_0 + \beta_1 \mathbf{1}_{\{\text{light}\}} + \beta_2 \mathbf{1}_{\{\text{moderate}\}} + \beta_3 \mathbf{1}_{\{\text{heavy}\}}$$

Parameter	**Value**	**p-value**
β_0	-5.5175	$< 2e-16$
β_1	0.6972	0.0231
β_2	1.9201	$9.50e-11$
β_3	2.3903	$7.24e-16$

Linear Models and Classification

Are there other ways to use linear models?

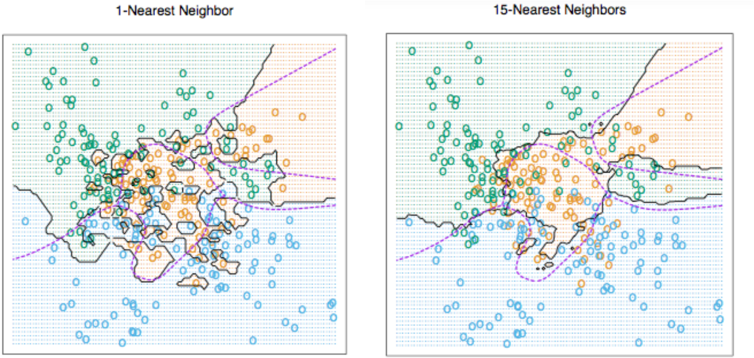
- logistic regression: linear in log odds
- what about *linear decision boundaries*?[1]



[1]Some figures reprinted from *The Elements of Statistical Learning* by Hastie, Tibshirani and Friedman.

Linear Models and Classification

$k-$nearest neighbors decision boundaries (and Bayes estimate)



when k increasing, more smooth

Linear Discriminant Analysis

Suppose we want to classify an observation into one of K classes, where $K \geq 2$. Let C_k denote the k-th class, $k = 1, \ldots, K$.

Define

$\pi_k = \mathbb{P}(Y = k)$	*prior* probability that an observation Y belongs to C_k
$p_k(x) = \mathbb{P}(Y = k \mid X = x)$	*posterior* probability that an observation $X = x$ belongs to C_k
$f_k(x) = \mathbb{P}(X = x \mid Y = k)$	*density* function of X for an observation that belongs to C_k

- $f_k(x)$ is the density for class k
- π_k is the probability of class k, with $\sum_{k=1}^{K} \pi_k = 1$

Linear Discriminant Analysis

Let's compute the probability of class k given covariates x:

$$\begin{aligned}
\mathbb{P}\left(Y = k \mid X = x\right) &= \frac{\mathbb{P}(Y = k, X = x)}{\mathbb{P}(X = x)} \\
&= \frac{\mathbb{P}(X = x \mid Y = k)\mathbb{P}(Y = k)}{\mathbb{P}(X = x)} \\
&= \frac{\mathbb{P}(X = x \mid Y = k)\mathbb{P}(Y = k)}{\sum_{\ell=1}^{K} \mathbb{P}(X = x \mid Y = \ell)\mathbb{P}(Y = \ell)} \\
&= \frac{f_k(x)\pi_k}{\sum_{\ell=1}^{K} f_\ell(x)\pi_\ell}
\end{aligned}$$

Issues:

- need to know $f_1(x), \ldots, f_K(x)$
- need to know $\pi_1, \ldots, \pi_K$

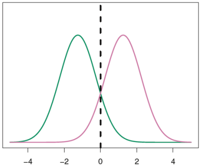
Linear Discriminant Analysis

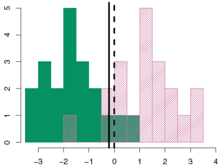
First, let's assume (for now) that

- $p = 1$ (we have only one covariate) and X is continuous
- $K = 2$ (we have only 2 classes)

Idea:

- model distribution of data for each class (fit distribution f_1, f_2, π_1, π_2)
- compare probabilities for each class at a location (using previous slide)
- select one with highest probability (Bayes Classifier)





Linear Discriminant Analysis

When a distribution is continuous, a *Gaussian distribution* is often a good fit

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

Here μ_k is the mean for the k^{th} component and σ_k^2 is the variance.

Let's also assume all of the classes have the same variance, $\sigma_1^2 = \cdots = \sigma_K^2 = \sigma^2$.

So, we can compute the probability of seeing the k^{th} class when $X = x$:

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu_k)^2\right)}{\sum_{\ell=1}^{K} \pi_\ell \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x-\mu_\ell)^2\right)} \tag{1}$$

Linear Discriminant Analysis

Bayes classifier: assign an observation $X = x$ to the class for which (1) is largest.

$$
\begin{aligned}
& p_1(x) \geq p_2(x) \\
\Rightarrow\ & \log(p_1(x)) \geq \log(p_2(x)) \\
\Rightarrow\ & \log(\pi_1) - \frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(x-\mu_1)^2 \\
& \quad \geq \log(\pi_2) - \frac{1}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(x-\mu_2)^2 \\
\Rightarrow\ & \log(\pi_1) - \frac{1}{2\sigma^2}(x-\mu_1)^2 \geq \log(\pi_2) - \frac{1}{2\sigma^2}(x-\mu_2)^2 \\
\Rightarrow\ & \log(\pi_1) + x\frac{\mu_1}{\sigma^2} - \frac{\mu_1^2}{2\sigma^2} \geq \log(\pi_2) + x\frac{\mu_2}{\sigma^2} - \frac{\mu_2^2}{2\sigma^2}
\end{aligned}
$$

Call one side of the last equation the *discriminant function*:

one good for use normal as can find discriminant funcion

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

Linear Discriminant Analysis

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

- if $K = 2$ and $\pi_1 = \pi_2$, the Bayes classifier assigns an observation
  - to class 1 if $2x(\mu_1 - \mu_2) > \mu_1^2 - \mu_2^2$,
  - to class 2 if $2x(\mu_1 - \mu_2) < \mu_1^2 - \mu_2^2$

  In this case, the Bayes decision boundary corresponds to the point where

  middle of two mean

$$x = \frac{\mu_1^2 - \mu_2^2}{2(\mu_1 - \mu_2)} = \frac{\mu_1 + \mu_2}{2}$$

- If we have discriminant functions for all of the $K > 2$ classes, choose the class with the highest value. That is, we assign the observation to the class for which $\delta_k(x)$ is largest.

Linear Discriminant Analysis

But how do we find the parameters?

- $\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i$
- $\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2$
- $\hat{\pi}_k = \frac{n_k}{n}$ n_k is the number of observation in k

$$\hat{\delta}_k(x) = \log(\hat{\pi}_k) + x\frac{\hat{\mu}_k}{\hat{\sigma}^2} - \frac{\hat{\mu}_k^2}{2\hat{\sigma}^2}$$

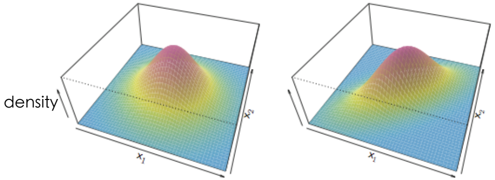
Linear Discriminant Analysis

Now suppose that $p > 1$....
Idea: model $f_k(x)$ as a multivariate Gaussian distribution

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

μ_k is a $p-$dimensional vector of means, $[\mathbb{E}[X_1], \mathbb{E}[X_2], \ldots, \mathbb{E}[X_p]]^T$

Σ_k is a $p \times p$ covariance matrix, $\Sigma_{k,ij} = \mathrm{Cov}[X_i, X_j]$



condition when p=2

Linear Discriminant Analysis

Estimating the parameters for a multivariate Gaussian (MLE):

$$\ell(\mu, \Sigma) = -\frac{np}{2}\log(2\pi) - \frac{n}{2}\log(|\Sigma|) - \frac{1}{2}\sum_{i=1}^{n}(x_i - \mu)^T\Sigma^{-1}(x_i - \mu)$$

$$\nabla_\mu\, \ell(\mu_j, \Sigma) = -\sum_{i=1}^{n}\Sigma^{-1}(x_i - \mu)$$

$$0 = \Sigma\, 0 = \sum_{i=1}^{n}(x_i - \mu)$$

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n}x_i$$ sample mean is the ML estimator

$$\hat{\Sigma} = \text{ some matrix calculus...}$$

$$= \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T \left(\approx \underbrace{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T}_{\text{unbaised}}\right)$$

Linear Discriminant Analysis

Back to estimating class probabilities...

$$\mathbb{P}(G = k \mid X = x) = \frac{f_k(x)\pi_k}{\sum_{\ell=1}^{K} f_\ell(x)\pi_\ell}$$

Idea 1: model data with Gaussians...that have the same covariance matrix

Idea 2: choose the class with the higher probability by comparing log probabilities

$$\log \frac{\mathbb{P}(G = k \mid X = x)}{\mathbb{P}(G = \ell \mid X = x)} = \log \frac{f_k(x)}{f_\ell(x)} + \log \frac{\pi_k}{\pi_\ell}$$

Linear Discriminant Analysis

$$
\begin{aligned}
\log\frac{\mathbb{P}(Y=k\,|\,X=x)}{\mathbb{P}(Y=\ell\,|\,X=x)} &= \log\frac{f_k(x)}{f_\ell(x)} + \log\frac{\pi_k}{\pi_\ell} \\
&= \log\frac{\pi_k}{\pi_\ell} + \log\left(\frac{(2\pi)^{-\frac{p}{2}}\,|\Sigma|^{-\frac{1}{2}}\,e^{-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)}}{(2\pi)^{-\frac{p}{2}}\,|\Sigma|^{-\frac{1}{2}}\,e^{-\frac{1}{2}(x-\mu_\ell)^T\Sigma^{-1}(x-\mu_\ell)}}\right) \\
&= \log\frac{\pi_k}{\pi_\ell} + \log\left(\frac{e^{-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)}}{e^{-\frac{1}{2}(x-\mu_\ell)^T\Sigma^{-1}(x-\mu_\ell)}}\right) \\
&= \log\frac{\pi_k}{\pi_\ell} + \log\left(e^{-\frac{1}{2}\left[(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)-(x-\mu_\ell)^T\Sigma^{-1}(x-\mu_\ell)\right]}\right) \\
&= \log\frac{\pi_k}{\pi_\ell} - \frac{1}{2}(\mu_k+\mu_\ell)^T\Sigma^{-1}(\mu_k-\mu_\ell) + x^T\Sigma^{-1}(\mu_k-\mu_\ell)
\end{aligned}
$$

Linear Discriminant Analysis

Comparing two classes

- we choose class k over class ℓ if $\log \frac{\mathbb{P}(Y=k \mid X=x)}{\mathbb{P}(Y=\ell \mid X=x)} > 0$
- we choose class ℓ over class k if $\log \frac{\mathbb{P}(Y=k \mid X=x)}{\mathbb{P}(Y=\ell \mid X=x)} < 0$
- we are indifferent when $\log \frac{\mathbb{P}(Y=k \mid X=x)}{\mathbb{P}(Y=\ell \mid X=x)} = 0$ (this is the decision boundary!)

What is the decision boundary?

$$0 = a + x^T b$$

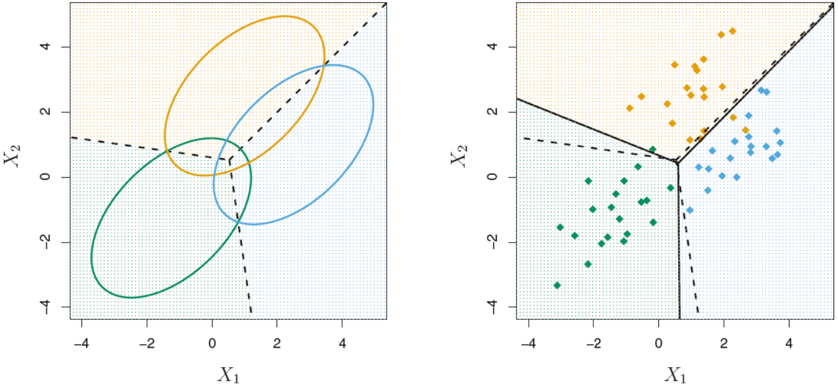
so boundary is linear form

$$a = \log \frac{\pi_k}{\pi_\ell} - \frac{1}{2}(\mu_k + \mu_\ell)^T \Sigma^{-1}(\mu_k - \mu_\ell)$$

$$b = \Sigma^{-1}(\mu_k - \mu_\ell)$$

Linear Discriminant Analysis

Gaussian observations, $K = 3$, $p = 2$.
Left: $\mu_1 \neq \mu_2 \neq \mu_3$, $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \mathbf{\Sigma}_3$, 95% ellipses; Dashed lines: Bayes decision boundaries.
Right: $n = 20$; Solid lines: LDA decision boundaries.



Linear Discriminant Analysis

Multiple classes:

- comparing pairs of log probabilities can get computationally intensive for large K
- use *discriminant functions* instead
- discriminant function is (condensed version of) $\log(f_k(x)\,\pi_k)$

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

Decision rule:

$$\hat{Y}(x) = \arg\max_k \delta_k(x)$$

Linear Discriminant Analysis

Estimating parameters:

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

$$\hat{\mu}_k = \sum_{g_i=k} \frac{x_i}{n_k}$$

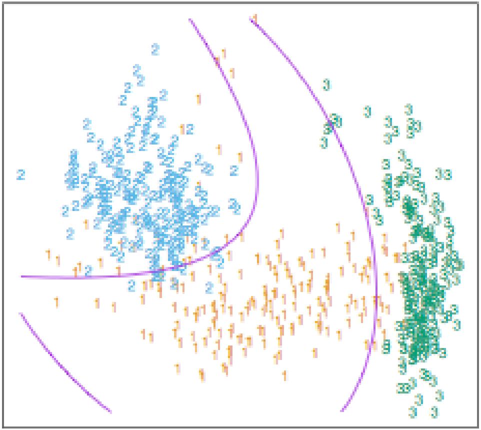
$$\hat{\Sigma} = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$$

K df lose by estimate mean for each k

Linear Discriminant Analysis

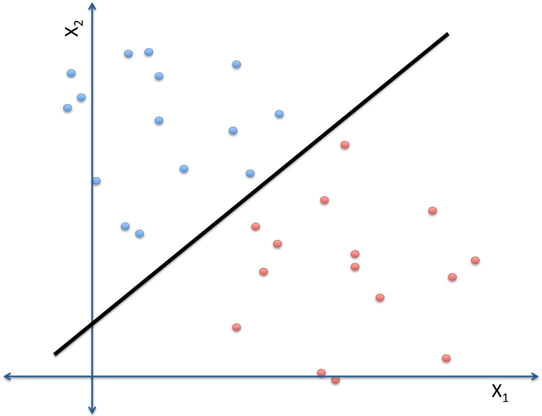
Extensions:

- can get non-linear boundaries by using non-linear basis functions
- example: $\tilde{X}_1 = X_1$, $\tilde{X}_2 = X_2$, $\tilde{X}_3 = X_1^2$, $\tilde{X}_4 = X_2^2$, $\tilde{X}_5 = X_1 X_2$ covariance



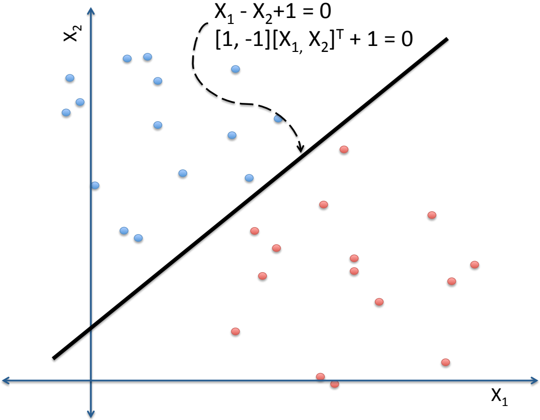
Linear Classifiers

A *linear classifier* is a classifier whose decision boundary is a line (or hyperplane)



Linear Classifiers

A linear classifier uses a linear combination of the features (or covariates) to make a classification decision

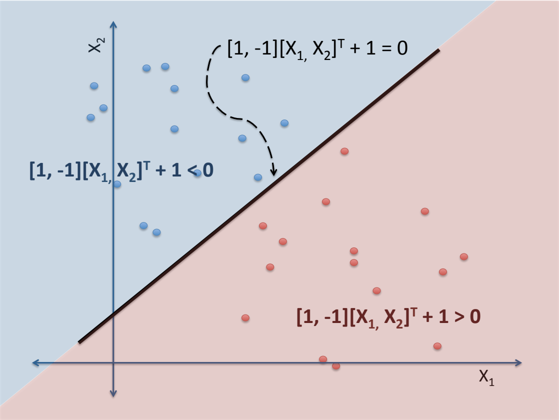


Linear Classifiers

Define a hyperplane as

$$\mathbf{a}^T\mathbf{x} - b = 0$$

Can divide space into set of $\mathbf{x}$ where $\mathbf{a}^T\mathbf{x} - b > 0$, $\mathbf{a}^T\mathbf{x} - b < 0$



Linear Classifiers

Define a hyperplane as

$$\mathbf{a}^T\mathbf{x} - b = 0$$

Can divide space into set of $\mathbf{x}$ where $\mathbf{a}^T\mathbf{x} - b > 0$, $\mathbf{a}^T\mathbf{x} - b < 0$

Classify the points: $(0, 1)$, $(-1, 1)$, $(2, 2)$, $(0, 0)$

Example 1: $\mathbf{a} = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$, $b = 1$

Example 2: $\mathbf{a} = \begin{bmatrix} -1/2 \\ -1/2 \end{bmatrix}$, $b = 1$

Example 3: $\mathbf{a} = \begin{bmatrix} -1/2 \\ -1/2 \end{bmatrix}$, $b = 0$

Linear Classifiers

Define a hyperplane as

$$\mathbf{a}^T\mathbf{x} - b = 0$$

Can divide space into set of $\mathbf{x}$ where $\mathbf{a}^T\mathbf{x} - b > 0$, $\mathbf{a}^T\mathbf{x} - b < 0$

This can be extended to higher dimensions

Classify the points: $(2, 2, 0, 0)$, $(1, 1, -1, -1)$, $(0, 0, 0, 0)$

Example 1: $\mathbf{a} = \begin{bmatrix} 0 \\ 2 \\ 0 \\ 1 \end{bmatrix}$, $b = 1$

Example 2: $\mathbf{a} = \begin{bmatrix} -1 \\ 2 \\ 1 \\ 1 \end{bmatrix}$, $b = 0$

Linear Discriminant Analysis

Let's implement LDA on the `Default` dataset with covariates `balance` and `income`.

```
> mu <- mat.or.vec(2,2)
> mu[1,1] <- mean(balance[default=="No"])
> mu[1,2] <- mean(balance[default=="Yes"])
> mu[2,1] <- mean(income[default=="No"])
> mu[2,2] <- mean(income[default=="Yes"])
> data.centered = rbind(cbind(balance[default=="No"]-mu[1,1],income[default=="No"]-mu[2,1]), +
cbind(balance[default=="Yes"]-mu[1,2],income[default=="Yes"]-mu[2,2]))
> Sigma <- mat.or.vec(2,2)
> Sigma[1,1] <- var(data.centered[,1])*10000/9998
> Sigma[2,2] <- var(data.centered[,2])*10000/9998
> Sigma[1,2] <- cov(data.centered[,1],data.centered[,2])*10000/9998
> Sigma[2,1] <- cov(data.centered[,1],data.centered[,2])*10000/9998
> pi.vec <- rep(0,2) # why not just pi?
> pi.vec[1] <- sum((default=="No"))/10000
> pi.vec[2] <- sum((default=="Yes"))/10000
```

n=10000, k =2

Linear Discriminant Analysis

Let's implement LDA on the `Default` dataset with covariates `balance` and `income`.

Let's make a function that takes μ and Σ and produces a selection for a vector of inputs.

```
my.lda <- function(pi.vec,mu,Sigma,x){
# Inputs:
# pi.vec : vector of class probabilities
# mu : matrix of means per class
# Sigma : covariance matrix
# x : vector of inputs
# Outputs:
# out.vec : vector of predicted classes
x.dims <- dim(x)
n <- x.dims[1]
Sigma.inv <- Sigma^(-1)

out.prod <- rep(2,n)

discrim.1 <- apply(x,1,function(y) y %*% Sigma.inv %*% mu[,1]
- 0.5*t(mu[,1]) %*% Sigma.inv %*% mu[,1] + log(pi.vec[1]))
discrim.2 <- apply(x,1,function(y) y %*% Sigma.inv %*% mu[,2]
- 0.5*t(mu[,2]) %*% Sigma.inv %*% mu[,2] + log(pi.vec[2]))

out.prod[discrim.1 >= discrim.2] <- 1

return(out.prod)
}
```

Linear Discriminant Analysis

So it turns out that when credit card balance and student status are used as covariates, we get an error rate of 2.75% (=23+252/10,000) on the training set.

Here we predict "Default" if our model says it has a probability higher than "No Default."

- Let's compare to the naive estimator: always predict the majority class.
- *Null* estimator has an error rate of 3.33% (=81+252/10,000).
- In this case, predictor usually says "No Default." This is common in problems with *unbalanced classes*.

our model not better a lot than simple null estimate, this because the unbalanced class, i.e. the difference of number of observations in difference class too large

		True default status		
		No	Yes	Total
Predicted default status	No	9,644	252	9,896
	Yes	23	81	104
	Total	9,667	333	10,000

Linear Discriminant Analysis

Here we predict "Default" if our model says it has a probability higher than "No Default."

Is this the best way if our classes are unbalanced? What if we lower the probability threshold for "Default", to say a probability of 0.2 or greater?

Old ($\mathbb{P}(Y = Default \,|\, X = x) > 0.5)$):

		True default status		
		No	Yes	Total
Predicted default status	No	9,644	252	9,896
	Yes	23	81	104
	Total	9,667	333	10,000

New ($\mathbb{P}(Y = Default \,|\, X = x) > 0.2)$):

as default' s probability less than 0.5 in sample, so change the decision rule

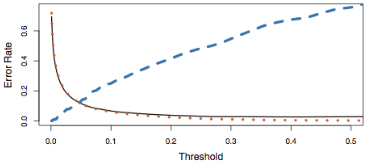
		True default status		
		No	Yes	Total
Predicted default status	No	9,432	138	9,570
	Yes	235	195	430
	Total	9,667	333	10,000

Linear Discriminant Analysis

By changing the threshold, we can change the *sensitivity* and *specificity* of our classifier:

- Sensitivity: the percentage of true defaulters identified by the test
- Specificity: the percentage of non-defaulters correctly identified

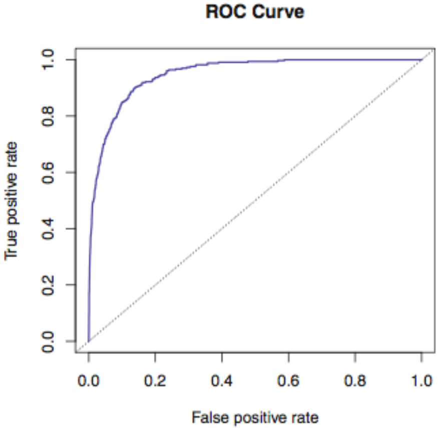
Bayes rate: the lowest *total* possible error rate out of all classifiers. But this may not be our goal...



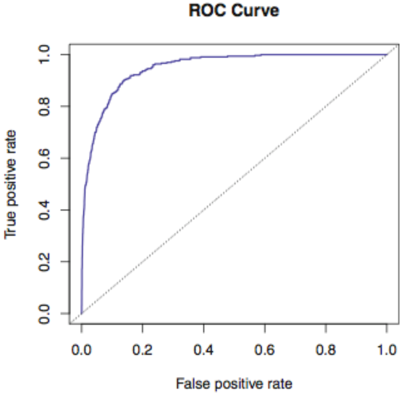
Black solid line: overall error rate. Blue dashed line: fraction of defaulting customers that are incorrectly classified. Orange dotted line: fraction of errors among the non-defaulting customers.

ROC Curves

We can characterize the tradeoff between sensitivity and specificity with an **ROC (receiver operating characteristics) curve**, which plots the true positive rate against the false positive rate.



ROC Curves



One way to describe the overall performance of a classifier is with the area under the ROC curve (AUC). A classifier no better than chance should have an AUC of 0.5. Here the AUC is 0.95, indicating a very good classifier.

Quadratic Discriminant Analysis

the boundary will be curve

Well, what happens if we let every class have its own covariance matrix?

- compare pairwise log probabilities
- ...which is the same as comparing log probabilities for all classes
- compute log probability

$$\log f_k(x)\pi_k = -\frac{p}{2}\log(2\pi) - \frac{1}{2}\log(|\Sigma_k|) - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \log \pi_k$$

- remove common terms to get discriminant functions

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

Quadratic Discriminant Analysis

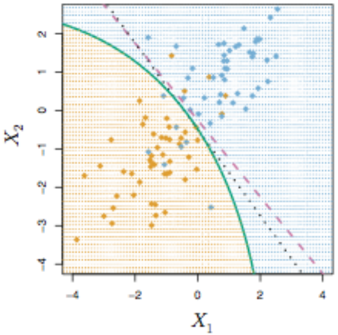
New discriminant function:

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

What type of boundaries?

- compute $\delta_k(x) - \delta_\ell(x) > 0$ to get class k region

Quadratic Discriminant Analysis



LDA vs QDA, where the Bayes boundary is linear (left) and quadratic (right)