Classification: Logistic regression & discriminant analysis

Hedibert F. Lopes & Paulo Marques
INSPER Institute of Education and Research
São Paulo, Brazil

Outline

Logistic regression

- Binary response
- Generalized linear model
- Maximum likelihood
- `default` dataset
- Bayesian logistic regression
- `spam` dataset

Discriminant analysis

- Discriminante rule
- Bayes discriminante rule
- Discriminant function
- Admissibility
- Decision theory and unequal costs
- `iris` dataset
- `admission` dataset

Outline

Binary response

We are still interested in learning about y via a set of predictors $x_1, x_2, \ldots, x_p$

Problem: y is a qualitative binary variable (1/0, yes/no, sucess/failure, etc).

The `Default` dataset (ISLR package) contains $n = 10,000$ observations.

- (y) default: yes/no indicating whether the customer defaulted on their debt
- (x_1) student: yes/no indicating whether the customer is a student
- (x_2) balance: Average balance after making their monthly payment
- (x_3) income: Income of customer

```
library(ISLR)
data(Default)
dim(Default)
Default[1:10,]
   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
5       No      No  785.6559 38463.496
6       No     Yes  919.5885  7491.559
7       No      No  825.5133 24905.227
8       No     Yes  808.6675 17600.451
9       No      No 1161.0579 37468.529
10      No      No    0.0000 29275.268
```

Logistic regression

A sample of size n of responses y_i and characteristics $x_i = (1, x_{i1}, x_{i2}, \ldots, x_{ip})'$, for $i = 1, \ldots, n$, is collected in order to construct a classifier.

A logistic regression assumes that

$$P(y_i = 1|x_i) = \frac{\exp\{\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}\}}{1 + \exp\{\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}\}} = \frac{\exp\{x_i'\beta\}}{1 + \exp\{x_i\beta\}},$$

where $\beta = (\beta_0, \beta_1, \ldots, \beta_p)'$.

It follows immediately that,

$$\log\left(\frac{P(y_i = 1|x_i)}{P(y_i = 0|x_i)}\right) = x_i'\beta,$$

since $P(y_i = 0|x_i) = 1 - P(y_i = 1|x_i)$.

Generalized linear model

The logistic regression belongs to the broad class of generalized linear models (GLM) where responses y_i are Gaussian, binomial, gamma, Poisson, etc.

In this case the responses (binary variables) are

$$y_i \sim Bernoulli(\pi_i)$$

where the "sucess" probabilities are individual-specific and related to predictors as

$$\log\left(\frac{\pi_i}{1-\pi_i}\right) = x_i'\beta.$$

Since, y_i is Bernoulli, it follows that

$$\begin{aligned} E(y_i|x_i) &= \pi_i = g(x_i'\beta) \\ V(y_i|x_i) &= \pi_i(1-\pi_i) = g(x_i'\beta)(1-g(x_i'\beta)), \end{aligned}$$

where

$$g(\mu) = \frac{e^\mu}{1+e^\mu} \quad \text{and} \quad g^{-1}(\pi) = \log\left(\frac{\pi}{1-\pi}\right).$$

Maximum likelihood estimation

It is easy to see that

$$p(y_1, \ldots, y_n | \pi_1, \ldots, \pi_n) = \prod_{i=1}^{n} \pi_i^{y_i} (1 - \pi_i)^{1-y_i},$$

or

$$p(y_{1:n} | x_{1:n}, \beta) = \prod_{i=1}^{n} [g(x_i'\beta)]^{y_i} (1 - g(x_i'\beta))^{1-y_i}.$$

In general, the MLE of β is

$$\hat{\beta}_{MLE} = \arg\max_{\beta} \prod_{i=1}^{n} [g(x_i'\beta)]^{y_i} (1 - g(x_i'\beta))^{1-y_i},$$

or

$$\begin{aligned} \hat{\beta}_{MLE} &= \arg\max_{\beta} \sum_{i=1}^{n} y_i \log\{g(x_i'\beta)\} + \sum_{i=1}^{n} (1 - y_i) \log\{1 - g(x_i'\beta)\} \\ &= \arg\max_{\beta} \sum_{i:y_i=1}^{n} \log\{g(x_i'\beta)\} + \sum_{i:y_i=0}^{n} \log\{1 - g(x_i'\beta)\}. \end{aligned}$$

When $g(\mu) = \frac{e^\mu}{1+e^\mu}$

Score equations: In this case, to maximize the log-likelihood, we set its derivatives to zero

$$\frac{\partial l(\beta)}{\partial \beta} = \sum_{i=1}^{n} x_i(y_i - g(x_i'\beta)) = 0$$

which are $p + 1$ equations nonlinear in β.

Newton-Raphson: The NR algorithm uses the matrix of 2nd derivatives (Hessian matrix) to find $\hat{\beta}_{MLE}$

$$\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta'} = -\sum_{i=1}^{n} x_i x_i' g(x_i'\beta)(1 - g(x_i'\beta))$$

Starting with β^{old}, a single Newton-Raphson update is

$$\beta^{\text{new}} = \beta^{\text{old}} - \left(\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta'}\right)^{-1} \frac{\partial l(\beta)}{\partial \beta},$$

where the derivatives are evaluated at β^{old}.

Iterratively reweighted least squares (IRLS)

Let

$$X'(y-g) = \frac{\partial l(\beta)}{\partial \beta} \quad \text{and} \quad -X'WX = \frac{\partial^2 l(\beta)}{\partial \beta \partial \beta'}.$$

The Newton-Raphson step is thus

$$\begin{aligned} \beta^{\text{new}} &= \beta^{\text{old}} + (X'WX)^{-1}X'(y-p) \\ &= (X'WX)^{-1}X'W(X\beta^{\text{old}} + W^{-1}(y-g)) \\ &= (X'WX)^{-1}X'Wz, \end{aligned}$$

which looks like weighted least squares with *adjusted response*

$$z = X\beta^{\text{old}} + W^{-1}(y-g).$$

At each iteration we solve the weighted least squares problem:

$$\beta^{\text{new}} = \arg\min_{\beta} (z - X\beta)'(z - X\beta)$$

Model diagnostics

Pearson's residuals

$$r_i = \frac{y_i - \hat{\pi}_i}{\sqrt{\hat{\pi}_i(1 - \hat{\pi}_i)}}$$

Deviance residuals

$$d_i = \text{sign}(y_i - \hat{\pi}_i)\sqrt{2\left[y_i \log\left(\frac{y_i}{\hat{\pi}_i}\right) + (1 - y_i)\log\left(\frac{1 - y_i}{1 - \hat{\pi}_i}\right)\right]}$$

Under the null hypothesis that the model is correct, if follows that

$$\chi^2 = \sum_{i-1}^{n} r_i \sim \chi^2_{n-p} \quad \text{and} \quad D = \sum_{i=1}^{n} d_i \sim \chi^2_{n-p}.$$

Classifying a new individual

A new individual $i = n + 1$ is classified as an "yes" individual or a "no" individual based on his characteristics $x_{n+1} = (x_{n+1,1}, \ldots, x_{n+1,p})'$ by comparing

$$P(y_{n+1} = 1 | x_{n+1}, y_{1:n}, x_{1:n}) \quad \text{and} \quad P(y_{n+1} = 0 | x_{n+1}, y_{1:n}, x_{1:n}),$$

which are "estimated" by

$$\widehat{P}(y_{n+1} = 1 | x_{n+1}, y_{1:n}, x_{1:n}) = \frac{\exp\{x'_{n+1}\hat{\beta}_{MLE}\}}{1 + \exp\{x'_{n+1}\hat{\beta}_{MLE}\}}.$$

and by

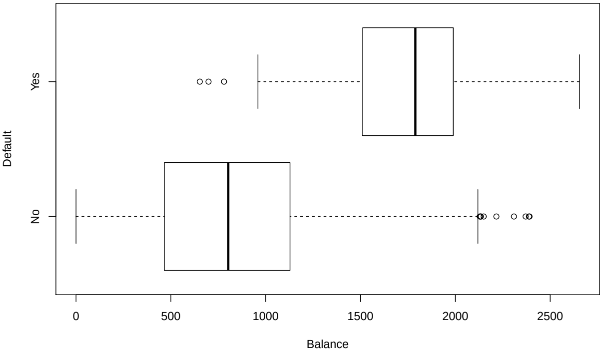
$$1 - \widehat{P}(y_{n+1} = 1 | x_{n+1}, y_{1:n}, x_{1:n}),$$

respectively.

Default dataset

(y) default: yes/no indicating whether the customer defaulted on their debt
(x) balance: Average balance after making their monthly payment

There are 9667 $y_i = 0$ and 333 $y_i = 1$



Linear and generalized linear models

Linear model (LM) via ordinary least squares (OLS)

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -7.519e-02  3.354e-03  -22.42   <2e-16 ***
balance      1.299e-04  3.475e-06   37.37   <2e-16 ***

Residual standard error: 0.1681 on 9998 degrees of freedom
Multiple R-squared:  0.1226,Adjusted R-squared:  0.1225
F-statistic:  1397 on 1 and 9998 DF,  p-value: < 2.2e-16
```

Generalized linear model (GLM) via iterative weighted least squares (IWLS)

```
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.065e+01  3.612e-01  -29.49   <2e-16 ***
balance      5.499e-03  2.204e-04   24.95   <2e-16 ***

Null deviance: 2920.6  on 9999  degrees of freedom
Residual deviance: 1596.5  on 9998  degrees of freedom
AIC: 1600.5
```

R code

```
install.packages("ISLR")
library(ISLR)
data(Default)
n = nrow(Default)
attach(Default)

default.binary = rep(0,n)
default.binary[default=="Yes"]=1

lm.fit = lm(default.binary~balance)
summary(lm.fit)

glm.fit = glm(default.binary~balance,family=binomial)
summary(glm.fit)
```

Classifying a new customer

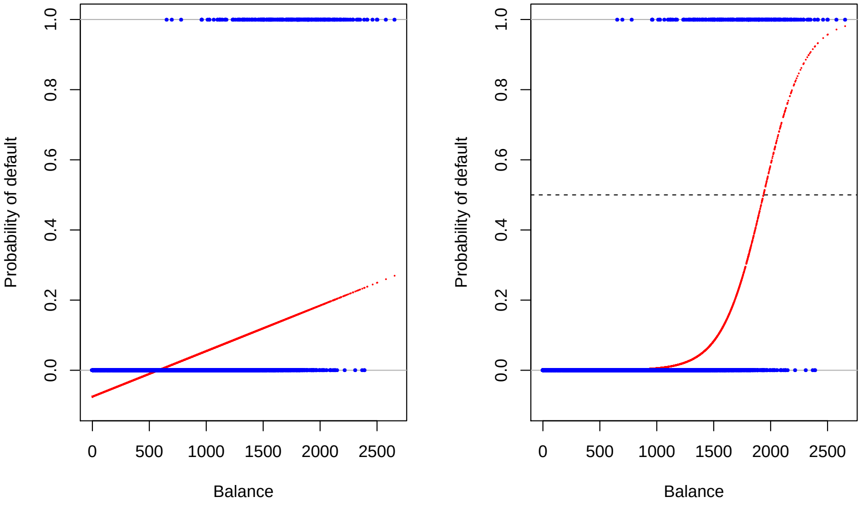
For a new customer $(n+1)$ with a balance of 1000 US dollars, it follows that

$$\widehat{P}(y_{n+1} = 1|x = 1000) = \frac{\exp\{-10.6513 + 0.0055(1000)\}}{1 + \exp\{-10.6513 + 0.0055(1000)\}} = 0.58\%,$$

while for a new customer with a balance of 2000 US dollars, it follows that

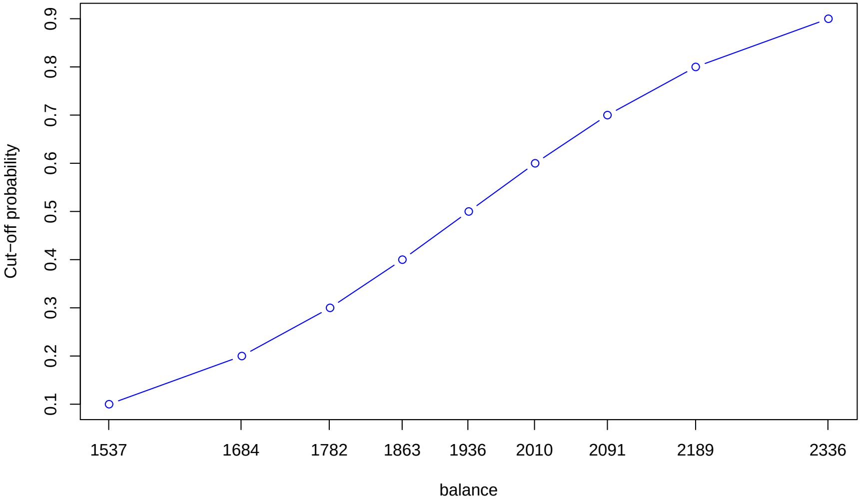
$$\widehat{P}(y_{n+1} = 1|x = 1000) = \frac{\exp\{-10.6513 + 0.0055(2000)\}}{1 + \exp\{-10.6513 + 0.0055(2000)\}} = 58.6\%.$$

Fitted OLS and GLS fits



`balance` as classifier

If one wants to use a cut-off probability of 50% to classify a new customer as a YES for default, then this translates into checking whether `balance` is below or above 1936 US dollars.



Categorical predictor

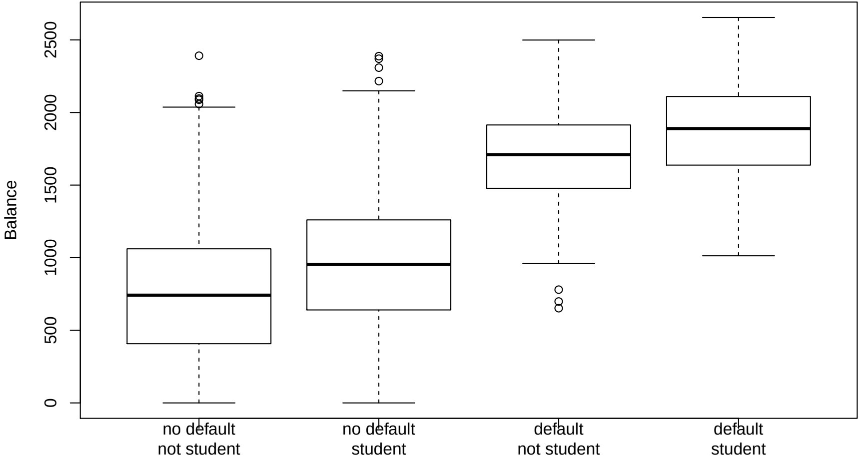
Here we want to use the binary variable `student` as a predictor for the binary variable `default`

	student		Total
default	No	Yes	Total
No	6850	2817	9667
Yes	206	127	333
Total	7056	2944	10000

```
               Estimate Std. Error z value Pr(>|z|)
(Intercept)    -3.50413    0.07071  -49.55  < 2e-16 ***
student.binary  0.40489    0.11502    3.52 0.000431 ***

Null deviance: 2920.6  on 9999  degrees of freedom
Residual deviance: 2908.7  on 9998  degrees of freedom
AIC: 2912.7
```

$$\widehat{P}(y_{n+1} = 1|\texttt{student=Yes}) = 0.43\%$$
$$\widehat{P}(y_{n+1} = 1|\texttt{student=No}) = 0.29\%$$

Two predictors: `balance` and `student`



GLS fit and prediction

```
               Estimate Std. Error z value Pr(>|z|)
(Intercept)    -1.075e+01  3.692e-01 -29.116  < 2e-16 ***
balance         5.738e-03  2.318e-04  24.750  < 2e-16 ***
student.binary -7.149e-01  1.475e-01  -4.846 1.26e-06 ***

Null deviance: 2920.6  on 9999  degrees of freedom
Residual deviance: 1571.7  on 9997  degrees of freedom
AIC: 1577.7
```

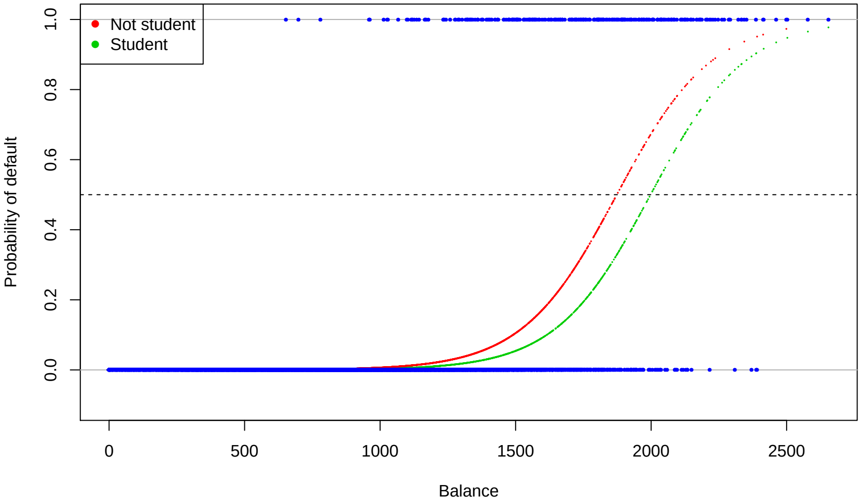
Therefore,

$$\widehat{P}(y_{n+1} = 1|\texttt{balance} = 1500, \texttt{student=Yes}) = 5.4\%$$

and

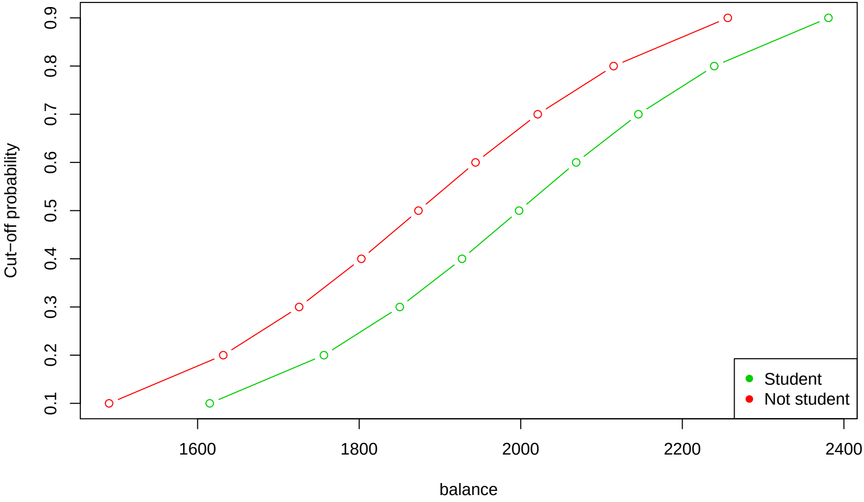
$$\widehat{P}(y_{n+1} = 1|\texttt{balance} = 1500, \texttt{student=No}) = 10.5\%$$

GLS fit


Not student
Student
Probability of default
0.0
0.2
0.4
0.6
0.8
1.0
0
500
1000
1500
2000
2500
Balance

`balance` as classifier

If one wants to use a cut-off probability of 50% to classify a new customer as a YES for default, then this translates into checking whether `balance` is below or above 1873 (1998) US dollars for students (non-students).



Bayesian logistic regression

Like everything Bayesian, the probability of default of a new customer, y_{n+1}, conditionally on his/her characteristics x_{n+1}, is obtained by integrating out the unknown parameters β based on its most current information assessment, that is based on its posterior distribution $p(\beta|y_{1:n}, x_{1:n})$:

$$P(y_{n+1} = 1|x_{n+1}, y_{1:n}, x_{1:n}) = \int P(y_{n+1} = 1|x_{n+1}, \beta)p(\beta|y_{1:n}, x_{1:n})d\beta$$

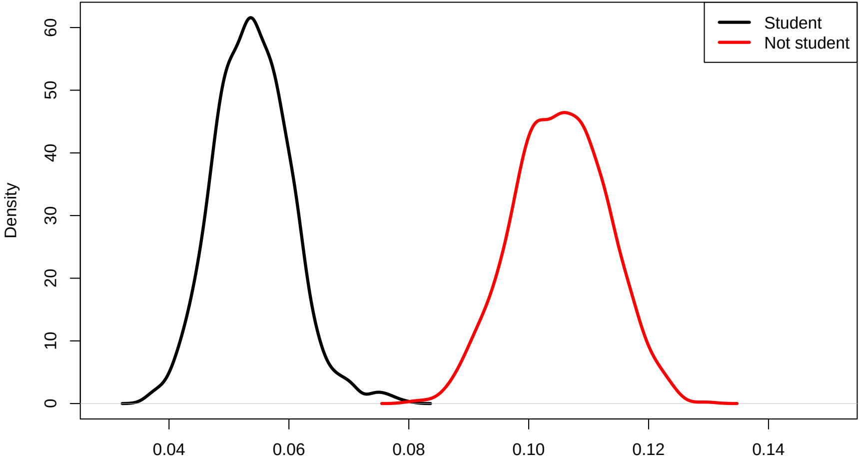
Notice that the MLE basically puts 100% of its mass at $\hat{\beta}_{MLE}$ and the above probability would approximated by

$$\widehat{P}(y_{n+1} = 1|x_{n+1}, y_{1:n}, x_{1:n}) = P(y_{n+1} = 1|x_{n+1}, \hat{\beta})$$

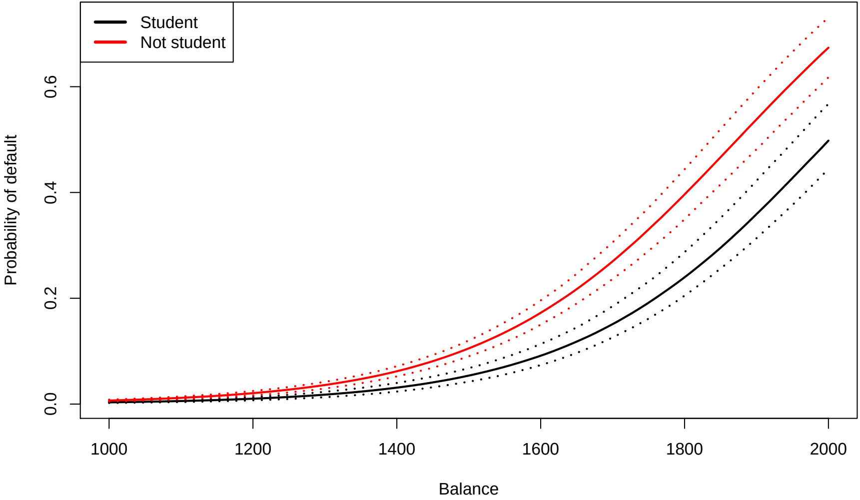
Bayesian logistic regression

```
install.packages("BayesLogit")
library(BayesLogit)
X = cbind(1,balance,student.binary)
bayesfit = logit(default.binary,X)
```

$P(y_{n+1} = 1 | \texttt{balance} = 1500, \texttt{student})$



Bayes fit



Student
Not student
Probability of default
0.0
0.2
0.4
0.6
1000
1200
1400
1600
1800
2000
Balance

Regularized logistic regression

Recall that the classic Gaussian elasticnet estimates β as that

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{n}(y_i - x_i'\beta)^2 + \lambda\left[(1-\alpha)||\beta||_2^2 + \alpha||\beta||_1\right],$$

with $\alpha = 1$ being the lasso penalty, and $\alpha = 0$ the ridge penalty.

It can be shown that the logistic elasticnet estimates β as

$$\hat{\beta} = \arg\min_{\beta} -\sum_{i=1}^{n}\left(y_i x_i'\beta - \log(1 + \exp\{x_i'\beta\})\right) + \lambda\left[(1-\alpha)||\beta||_2^2 + \alpha||\beta||_1\right]$$

R package glmnet

Logistic LASSO:

```
glmnet(x=X,y=Y,family="binomial",alpha=1)

cv.glmnet(x=X,y=Y,family="binomial",alpha=1)
```

Classifications/probabilities:

```
predict(out,X_0,s="lambda.min",type="class")

predict(out,X_0,s="lambda.min",type="response")
```

spam dataset[1]

A researcher labeled 4601 *of his emails as either* `spam` *or* `ham`, *say*

$$y_i = \begin{cases} 1 & \textit{if email } i \textit{ is } \texttt{spam} \\ 0 & \textit{if email } i \textit{ is } \texttt{ham} \end{cases}$$

40% of the messages were `spam`*.*

57 *predictors: most frequently used words/tokens.*

The goal of the study is to predict whether future emails are `spam` *or* `ham` *using these keywoords; that is to build a customized spam filter.*

[1] Text from Efron and Hastie (2016) *Computer Age Statistical Inference: Algorithms, Evidence, and Data Science*, pages 113-115.

Predictors

Predictors: x_{ij} is the relative frequency of a keyword j in email i

1 make	16 free	31 telnet	46 edu
2 address	17 business	32 857	47 table
3 all	18 email	33 data	48 conference
4 3d	19 you	34 415	49 char;
5 our	20 credit	35 85	50 char(
6 over	21 your	36 technology	51 char
7 remove	22 font	37 1999	52 char!
8 internet	23 000	38 parts	53 char$
9 order	24 money	39 pm	54 char#
10 mail	25 hp	40 direct	55 cap.ave
11 receive	26 hpl	41 cs	56 cap.long
12 will	27 george	42 meeting	57 cap.tot
13 people	28 650	43 original	
14 report	29 lab	44 project	
15 addresses	30 labs	45 re	

Observed predictors for the first 4 emails

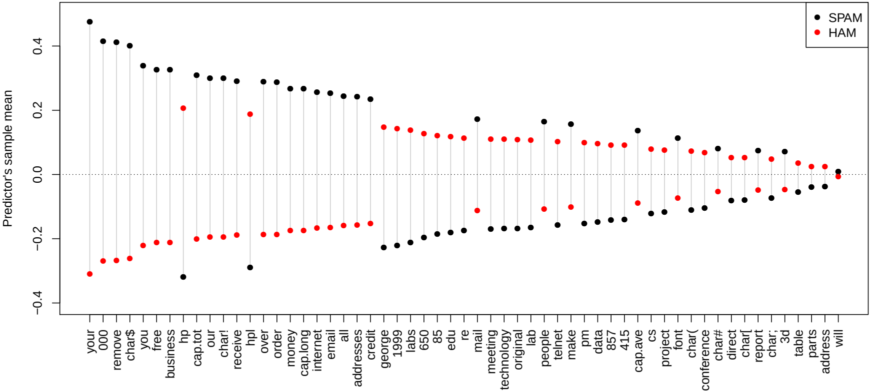
	make	address	all	3d	our	over	remove	internet	order	mail	receive	will	people	report	addresses	free	business	email
[1,]	0.00	0.64	0.64	0	0.32	0.00	0.00	0.00	0.00	0.00	0.00	0.64	0.00	0.00	0.00	0.32	0.00	1.29
[2,]	0.21	0.28	0.50	0	0.14	0.28	0.21	0.07	0.00	0.94	0.21	0.79	0.65	0.21	0.14	0.14	0.07	0.28
[3,]	0.06	0.00	0.71	0	1.23	0.19	0.19	0.12	0.64	0.25	0.38	0.45	0.12	0.00	1.75	0.06	0.06	1.03
[4,]	0.00	0.00	0.00	0	0.63	0.00	0.31	0.63	0.31	0.63	0.31	0.31	0.31	0.00	0.00	0.31	0.00	0.00

	you	credit	your	font	000	money	hp	hpl	george	650	lab	labs	telnet	857	data	415	85	technology	1999	parts	pm	direct	cs
[1,]	1.93	0.00	0.96	0	0.00	0.00	0	0	0	0	0	0	0	0	0	0	0	0	0.00	0	0	0.00	0
[2,]	3.47	0.00	1.59	0	0.43	0.43	0	0	0	0	0	0	0	0	0	0	0	0	0.07	0	0	0.00	0
[3,]	1.36	0.32	0.51	0	1.16	0.06	0	0	0	0	0	0	0	0	0	0	0	0	0.00	0	0	0.06	0
[4,]	3.18	0.00	0.31	0	0.00	0.00	0	0	0	0	0	0	0	0	0	0	0	0	0.00	0	0	0.00	0

	meeting	original	project	re	edu	table	conference	char;	char(	char[	char!	char$	char#	cap.ave	cap.long	cap.tot
[1,]	0	0.00	0	0.00	0.00	0	0	0.00	0.000	0	0.778	0.000	0.000	3.756	61	278
[2,]	0	0.00	0	0.00	0.00	0	0	0.00	0.132	0	0.372	0.180	0.048	5.114	101	1028
[3,]	0	0.12	0	0.06	0.06	0	0	0.01	0.143	0	0.276	0.184	0.010	9.821	485	2259
[4,]	0	0.00	0	0.00	0.00	0	0	0.00	0.137	0	0.137	0.000	0.000	3.537	40	191

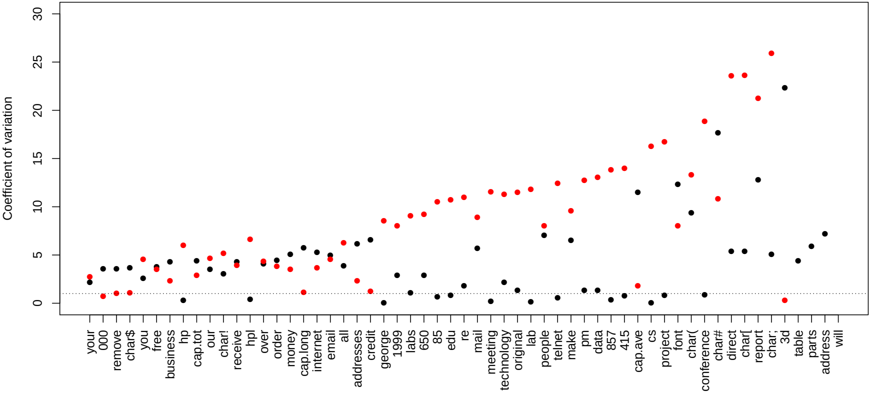
Exploratory data analysis

Predictor's sample means



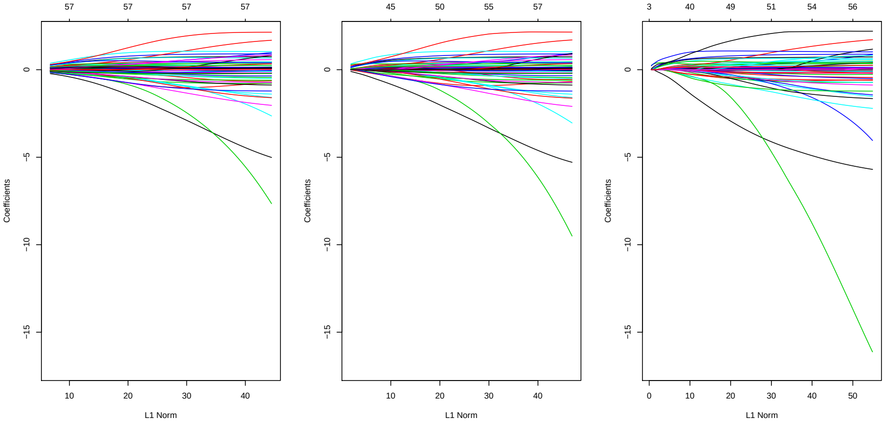
Exploratory data analysis

Predictor's coefficient of variations



Regularized logistic regression

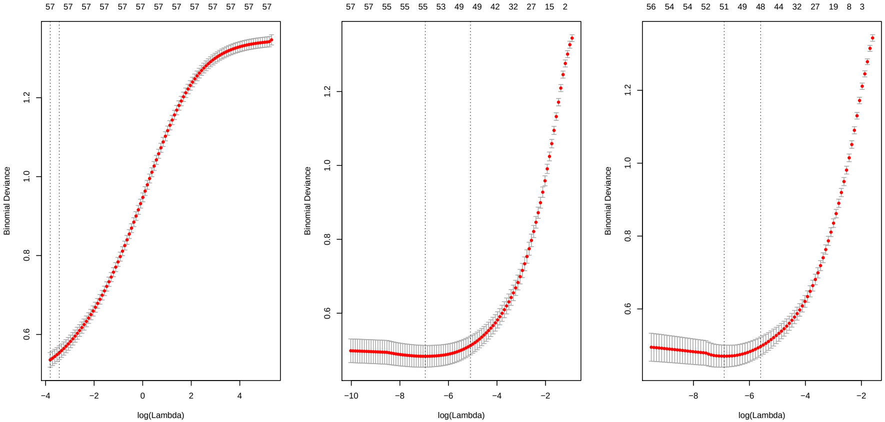
Ridge, elasticnet and lasso



Regularized logistic regression

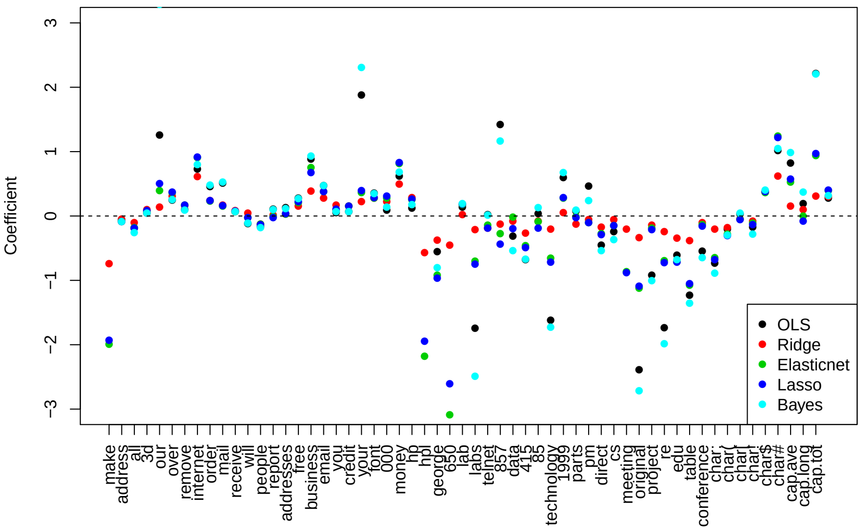
Choosing penalty parameter: Ridge, elasticnet and lasso

Training size is 2300 (testing is 2301)



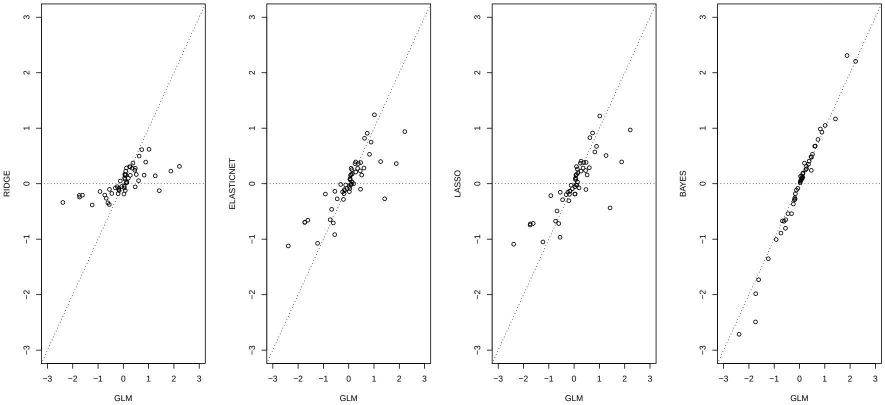
Estimation

Comparing estimated predictor's coefficients



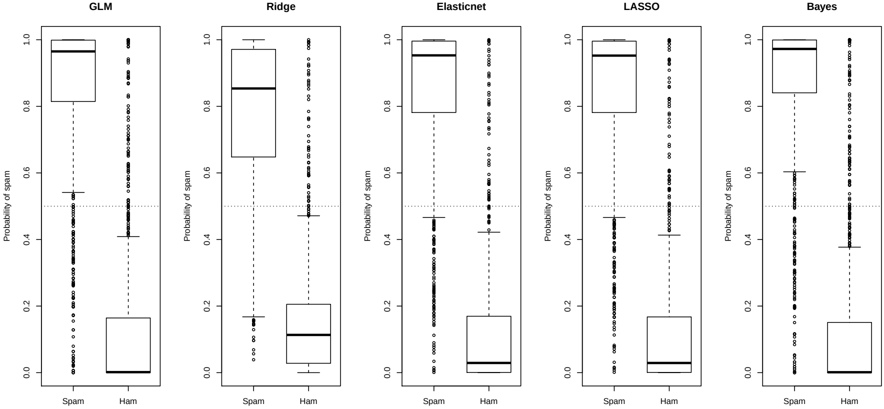
Estimation

Shrinkage effect



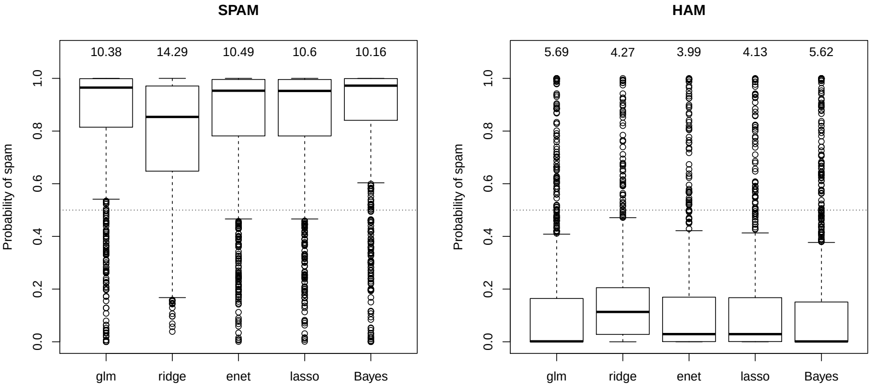
Misclassification rates

Probability of SPAM - testing sample



Misclassification rates

Probability of SPAM - testing sample



Classification tables

	y=0,yhat=0	y=0,yhat=1	y=1,yhat=0	y=1,yhat=1	error
glm	1325	80	93	803	7.52
ridge	1345	60	128	768	8.17
enet	1349	56	94	802	6.52
lasso	1347	58	95	801	6.65
bayes	1326	79	91	805	7.39

False negatives: Classifying a `spam` $(y = 1)$ as a `ham`
False positives: Classifying a `ham` $(y = 0)$ as a `spam`
False discoveries: False positives over positives.

	false.negative	false.positive	false.discovery
glm	10.38	5.69	8.20
ridge	14.29	4.27	6.28
enet	10.49	3.99	5.88
lasso	10.60	4.13	6.08
bayes	10.16	5.62	8.10

Outline

Discriminant analysis

Logistic regression (LR) models $P(y = k|x)$ directly.

Discriminant analysis (DA) models the predictors x within each class k of y and then uses Bayes' rule to estimate $P(y = k|x)$.

When the xs within each class of y are Gaussian, LR and DA are quite similar.

Why bother?

- LR are unstable when classes are well-separated
- LR are unstable when n is small
- LR is not too popular when $k > 2$

Discriminant rule[2]

A discriminant rule d corresponds to a division of $\mathbb{R}^p$ into disjoint regions $R_1, \ldots, R_\kappa$ such that $\cap_{k=1}^{\kappa} = \mathbb{R}^p$.

The rule d is defined by

$$\text{allocate } x \text{ to group } k \text{ if } x \in R_k,$$

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

[2]Based on Mardia, Kent and Bibby's *Multivariate Analysis*, Chapter 11.

Bayes discriminant rule

π_k : probability that an observation comes from class k, for $j = 1, \ldots, \kappa$.
$p(x|y = k)$: probability density function of x from class k, for $k = 1, \ldots, \kappa$.
$P(y = k|x)$: Bayes' theorem states that

$$P(y = k|x) = \frac{\pi_k p(x|y = k)}{\sum_{j=1}^{\kappa} \pi_j p(x|y = j)}$$

Bayes discriminant rule: Allocate observation x to the population k^* such that

$$k^* = \arg \max_{k \in 1, \ldots, \kappa} P(y = k|x)$$

is maximized. Alternatively, via allocation functions

$$\phi_k(x) = \begin{cases} 1 & \text{if } \pi_k p(x|y = k) = \max_j \pi_j p(x|y = j) \\ 0 & \text{otherwise,} \end{cases}$$

Maximum likelihood discriminant rule: $\pi_1 = \cdots = \pi_\kappa = 1/\kappa$.

Example: 0-1 predictor

Let x be a Bernoulli random variable, with

$$\begin{aligned} x|y=1 &\sim Bernoulli(1/2) \\ x|y=2 &\sim Bernoulli(3/4) \end{aligned}$$

The ML discriminant rule allocates x to class 1 when $x=0$ and allocates x to class 2 when $x=1$, since

$$\begin{aligned} p(x=0|y=1)=1/2 &> 1/4=p(x=0|y=2) \\ p(x=1|y=1)=1/2 &< 3/4=p(x=1|y=2) \end{aligned}$$

Example: Multinomial

Suppose x is a multinomial random variable, with

$$\begin{aligned} x|y=1 &\sim Multinomial(\alpha_1,\ldots,\alpha_\kappa) \\ x|y=2 &\sim Multinomial(\beta_1,\ldots,\beta_\kappa) \end{aligned}$$

where

$$\sum_{k=1}^{\kappa}\alpha_k = \sum_{k=1}^{\kappa} = 1 \quad \text{and} \quad \sum_{k=1}^{\kappa} x_k = n.$$

The likelihood functions are

$$\begin{aligned} p(x|y=1) &= \frac{n!}{x_1!\cdots x_\kappa!}\alpha_1^{x_1}\cdots\alpha_\kappa^{x_\kappa} \\ p(x|y=2) &= \frac{n!}{x_1!\cdots x_\kappa!}\beta_1^{x_1}\cdots\beta_\kappa^{x_\kappa} \end{aligned}$$

The ML discriminant rule allocates x to class 1 if

$$\sum_{k=1}^{\kappa} x_i \log\frac{\alpha_k}{\beta_k} < 0.$$

Univariate Gaussian models

Suppose x is a Gaussian random variable, with

$$\begin{aligned} x|y = 1 &\sim N(\mu_1, \sigma_1^2) \\ x|y = 2 &\sim N(\mu_2, \sigma_2^2) \end{aligned}$$

where $\mu_1 < \mu_2$ and $\sigma_1 > \sigma_2$.

$p(x|y = 1) > p(x|y = 2)$ if

$$\frac{\sigma_2}{\sigma_1} \exp\left\{ -\frac{1}{2} \left[\left(\frac{x - \mu_1}{\sigma_1} \right)^2 - \left(\frac{x - \mu_2}{\sigma_2} \right)^2 \right] \right\} > 1,$$

or

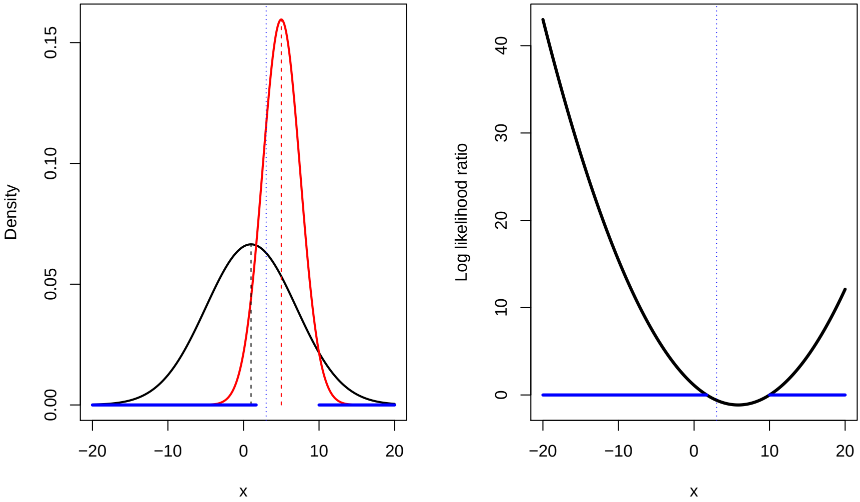
$$x^2 \left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_2^2} \right) - 2x \left(\frac{\mu_1}{\sigma_1^2} - \frac{\mu_2}{\sigma_2^2} \right) + \left(\frac{\mu_1^2}{\sigma_1^2} - \frac{\mu_2^2}{\sigma_2^2} \right) < 2 \log \frac{\sigma_2}{\sigma_1} < 0.$$

If $\sigma_1 = \sigma_2$, then $p(x|y = 1) > p(x|y = 2)$ when

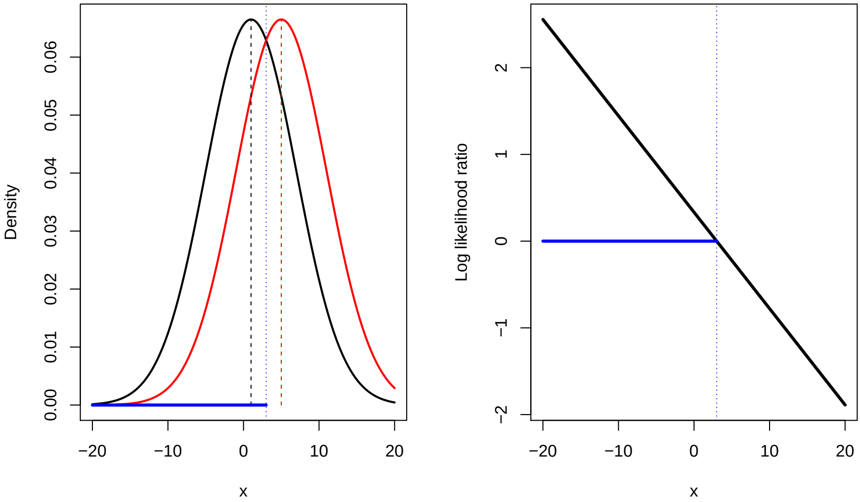
$$|x - \mu_2| > |x - \mu_1|$$

or when $x < (\mu_1 + \mu_2)/2$.

Example: $\mu_1 = 1, \mu_2 = 5, \sigma_1 = 6, \sigma_2 = 2.5$



Example: $\mu_1 = 1, \mu_2 = 5, \sigma_1 = 6, \sigma_2 = 6$



Gaussian populations with common variances

- Let $(x|y = k)$ be the $N_p(\mu_k, \Sigma)$, for $k = 1, \ldots, \kappa$ and $\Sigma > 0$[3].

- The ML discrimination rule allocates x to class k^* such that

$$k^* = \arg \min_{k \in 1,\ldots,\kappa} (x - \mu_k)' \Sigma^{-1} (x - \mu_k),$$

  i.e., k^* minimzes the square of the Mahalanobis distance between x and μ_k.

- When $\kappa = 2$, the rule allocates x to class $k = 1$ if

$$\alpha'(x - \mu) > 0,$$

  where $\alpha = \Sigma^{-1}(\mu_1 - \mu_2)$ and $\mu = (\mu_1 + \mu_2)/2$.

- The discriminant function for two Gaussians with the same covariance matrix is linear. Quadratic Discriminant Analysis assumes distinct covariance matrices.

---

[3] $\Sigma > 0$ if $z'\Sigma z > 0, \forall z \neq 0$.

Discriminant function

When there are just $\kappa = 2$ classes, the ML discriminant rule is defined in terms of the discriminant function

$$h(x) = \log p(x|y = 1) - \log p(x|y = 2)$$

and the ML rule takes the form

$$\text{Allocate } x \text{ to class 1 if } h(x) > 0$$
$$\text{Allocate } x \text{ to class 2 if } h(x) < 0,$$

while the Bayes discriminant rule takes the form of

$$\text{Allocate } x \text{ to class 1 if } h(x) > \log \pi_2/\pi_1$$
$$\text{Allocate } x \text{ to class 2 if } h(x) < \log \pi_2/\pi_1$$

Admissibility

The probability of allocating an individual to class i, when in fact she comes from class j, is given by

$$p_{ij} = \int \phi_i(x) p(x|y=j) dx$$

Say that one discriminant rule d with probabilities of correct allocation $\{p_{kk}\}$ is as good as another rule d' with probabilities $\{p'_{kk}\}$ if

$$p_{kk} \geq p'_{kk} \qquad \text{for all } k = 1, \ldots, \kappa.$$

Say that d is better than d' if at least one of the inequalities is strict. If d is a rule for which there is no better rule, say that d is admissible.

Theorem: All Bayes discriminant rules (including the ML rule) are admissible.

Theorem: If populations $k = 1, \ldots, \kappa$ have prior probabilities $\pi_1, \ldots, \pi_\kappa$, then no discriminant rule has a larger posterior probability of correct allocation than the Bayes rule with respect to this prior.

Decision theory and unequal costs

The discrimination problem can be seen as a decision problem. Let

$$K(i,j) = \begin{cases} 0, & i = j, \\ c_{ij} & i \neq j. \end{cases}$$

be a loss function representing the cost or loss incurred when an observation is allocated to class i when in fact it comes from class j, assuming $c_{ij} > 0 \; \forall i \neq j$.

If d is a rule with allocation function $\phi_k(x)$, then the risk function is defined by

$$\begin{aligned} R(d,k) &= E(K(d(x),k)|y=k) \\ &= \sum_{j=1}^{\kappa} K(j,k) \int \phi_j(x) p(x|y=k) dx = sum_{j=1}^{\kappa} c_{jk} p_{jk} \end{aligned}$$

If prior probabilities exist then the Bayes risk can be defined by

$$r(d,\pi) = \sum_{k=1}^{\kappa} \pi_k R(d,k)$$

and represents the posterior expected loss.

Theorems

Theorem 1: All Bayes discrimination rules are admissible for the risk function R.

Theorem 2: If the classes $k = 1, \ldots, \kappa$ have prior probabilities $\pi_1, \ldots, \pi_\kappa$, then no discriminant rule has smaller Bayes risk for the risk function R than the Bayes rule with respect to π.

The advantage of the decision theory approach is that it allows us to attach varying levels of importance to different sorts of errors.

For example, in medical diagnosis it might be regarded as more harmful to a patient's survival for polio to be misdiagnosed as flu than for flu to be misdiagnosed as polio.

`iris` dataset

This famous (Fisher's or Anderson's) iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are Iris setosa, versicolor, and virginica.

`iris` is a data frame with 150 cases (rows) and 5 variables (columns) named `Sepal.Length`, `Sepal.Width`, `Petal.Length`, `Petal.Width`, and `Species`.

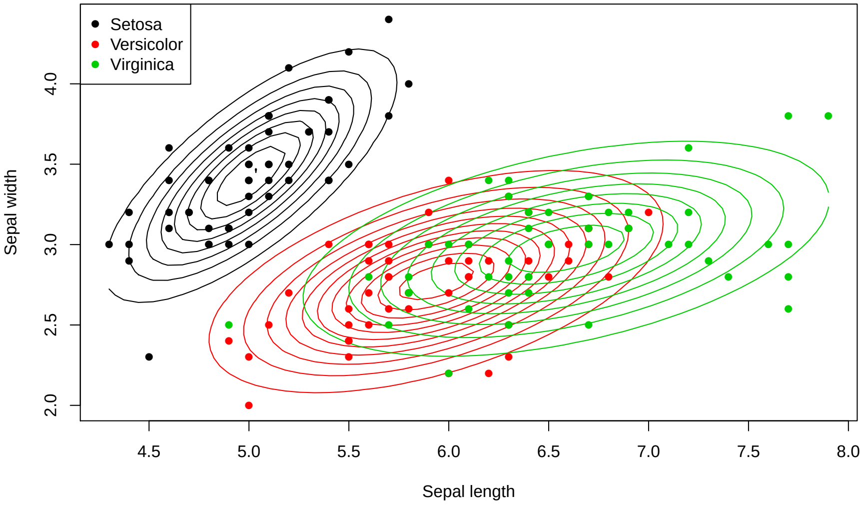
Anderson (1935) The irises of the Gaspe Peninsula
Bulletin of the American Iris Society, 59, 2-5.

Fisher (1936) The use of multiple measurements in taxonomic problems
Annals of Eugenics, 7, Part II, 179-188.

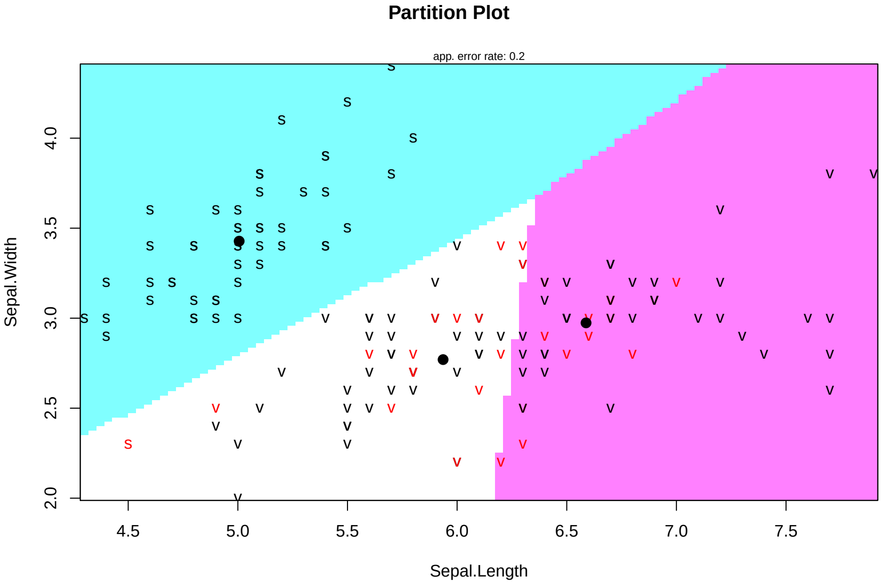
Summary statistics

```
data(iris)

iris[c(1,51,101),]
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
1            5.1         3.5          1.4         0.2     setosa
51           7.0         3.2          4.7         1.4 versicolor
101          6.3         3.3          6.0         2.5  virginica

y = rep(0,nrow(iris))
y[iris[,5]=="setosa"]=1
y[iris[,5]=="versicolor"]=2
y[iris[,5]=="virginica"]=3
x = as.matrix(iris[,1:2])
n = nrow(x)
n1 = sum(y==1)
n2 = sum(y==2)
n3 = sum(y==3)

xbar1 = apply(x[y==1,],2,mean)
xbar2 = apply(x[y==2,],2,mean)
xbar3 = apply(x[y==3,],2,mean)
S1 = var(x[y==1,])*(n1-1)/n1
S2 = var(x[y==2,])*(n2-1)/n2
S3 = var(x[y==3,])*(n3-1)/n3

cbind(xbar1,xbar2,xbar3)
             xbar1 xbar2 xbar3
Sepal.Length 5.006 5.936 6.588
Sepal.Width  3.428 2.770 2.974

round(cbind(S1,S2,S3),3)
             Sepal.Length Sepal.Width Sepal.Length Sepal.Width Sepal.Length Sepal.Width
Sepal.Length        0.122       0.097        0.261       0.083        0.396       0.092
Sepal.Width         0.097       0.141        0.083       0.096        0.092       0.102
```

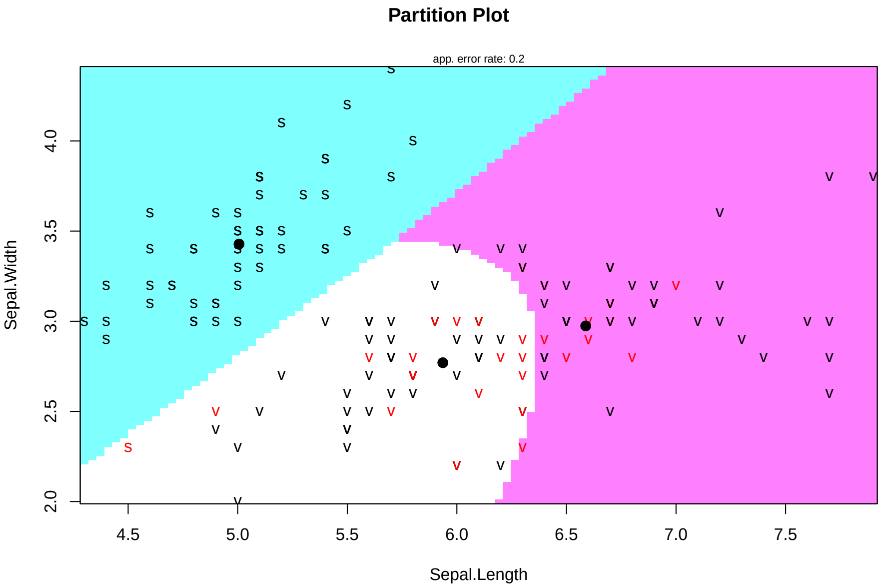
Discrimination between three species of iris



LDA


Partition Plot
app. error rate: 0.2
Sepal.Width
Sepal.Length
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5

QDA



Partition Plot
app. error rate: 0.2
Sepal.Width
Sepal.Length
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
6.5
7.0
7.5

admission dataset[4]

Admission data for applicants to graduate schools in business.

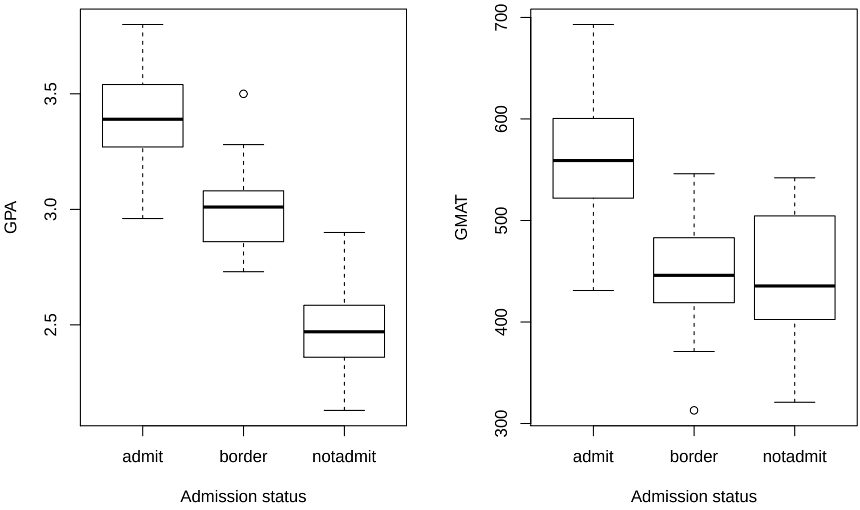
Objective: Predict likelihood of admission via GPA and GMAT scores.

Admission levels: admit, notadmit, and borderline

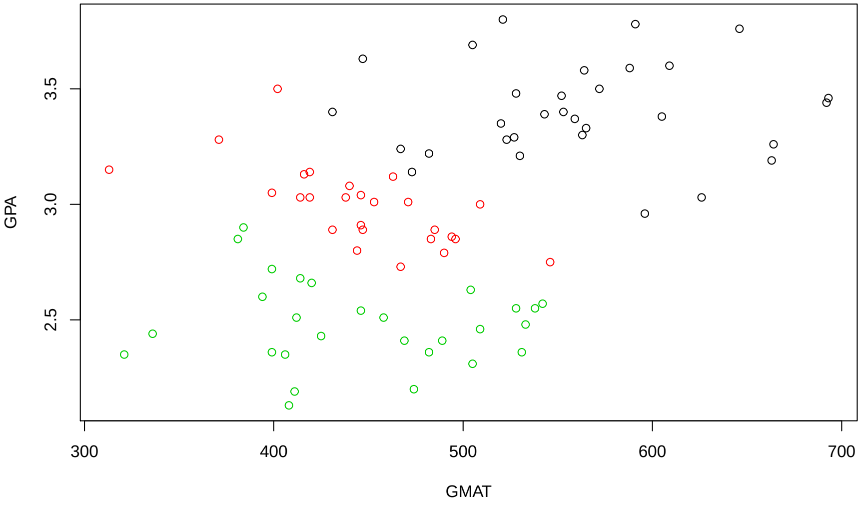
```
url <- "http://www.biz.uiowa.edu/faculty/jledolter/DataMining/admission.csv"
admit <- read.csv(url)
dim(admit)
adm=data.frame(admit)

par(mfrow=c(1,2))
boxplot(GPA~De,data=admit)
boxplot(GMAT~De,data=admit)

par(mfrow=c(1,1))
plot(adm$GPA,adm$GMAT,col=adm$De)
```

[4] Example from Johannes Ledolter: https://www.biz.uiowa.edu/faculty/jledolter

Boxplots


GPA
3.5
3.0
2.5
admit
border
notadmit
Admission status
GMAT
700
600
500
400
300
admit
border
notadmit
Admission status

Analysis



Linear discriminant analysis

```
> m1=lda(De~.,data=adm)
> m1
Call:
lda(De ~ ., data = adm)

Prior probabilities of groups:
    admit    border  notadmit
0.3647059 0.3058824 0.3294118

Group means:
              GPA     GMAT
admit    3.403871 561.2258
border   2.992692 446.2308
notadmit 2.482500 447.0714

Coefficients of linear discriminants:
             LD1         LD2
GPA  5.008766354  1.87668220
GMAT 0.008568593 -0.01445106

Proportion of trace:
   LD1    LD2
0.9673 0.0327

> predict(m1,newdata=data.frame(GPA=3.21,GMAT=497))
$class
[1] admit
Levels: admit border notadmit

$posterior
      admit    border     notadmit
1 0.5180421 0.4816015 0.0003563717

$x
       LD1      LD2
1 1.252409 0.318194
```

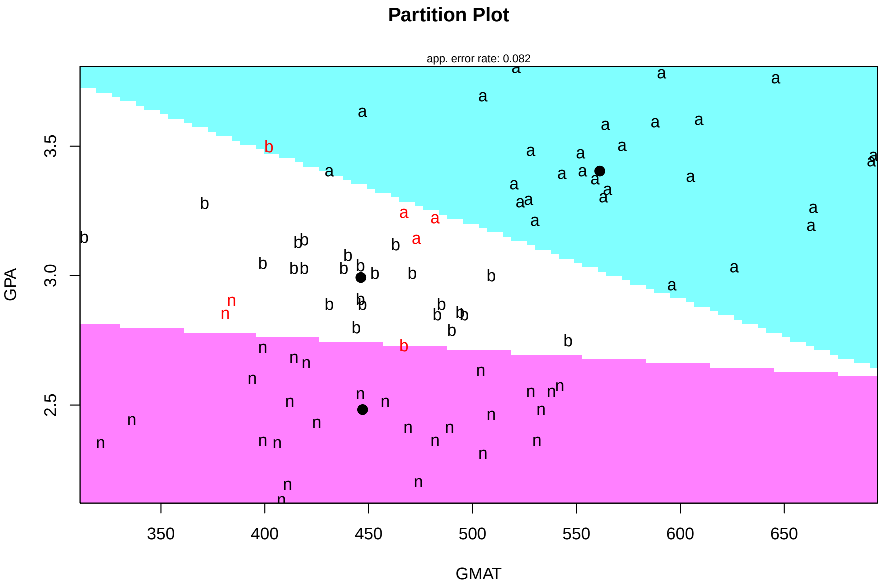
Quadratic discriminant analysis

```
> m2=qda(De~.,adm)
> m2
Call:
qda(De ~ ., data = adm)

Prior probabilities of groups:
    admit    border  notadmit
0.3647059 0.3058824 0.3294118

Group means:
              GPA     GMAT
admit    3.403871 561.2258
border   2.992692 446.2308
notadmit 2.482500 447.0714
> predict(m2,newdata=data.frame(GPA=3.21,GMAT=497))
$class
[1] admit
Levels: admit border notadmit

$posterior
      admit    border     notadmit
1 0.9226763 0.0768693 0.0004544468
```

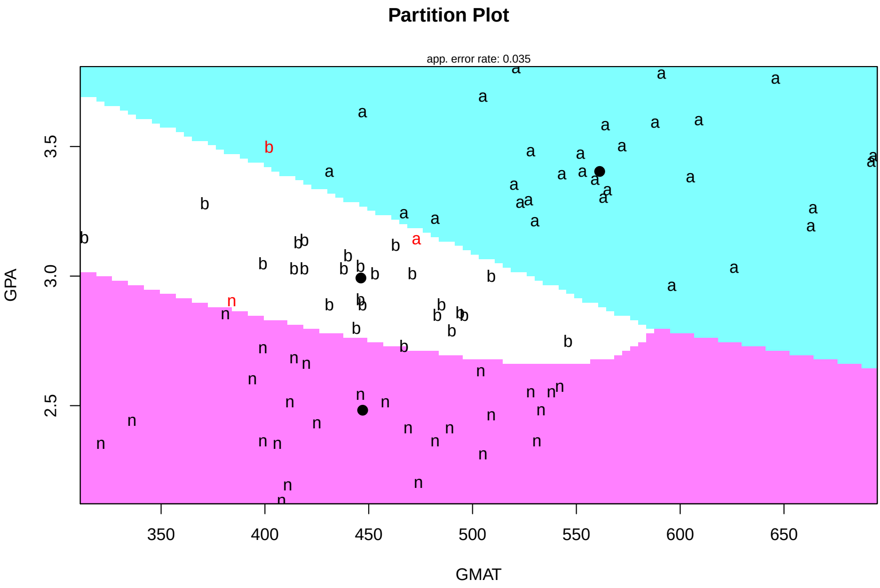
Exploratory Graph for LDA or QDA

```
install.packages('klaR')
library(klaR)
partimat(De~.,data=adm,method="lda")
partimat(De~.,data=adm,method="qda")
```

LDA


Partition Plot
app. error rate: 0.082
GPA
GMAT
3.5
3.0
2.5
350
400
450
500
550
600
650

QDA


Partition Plot
app. error rate: 0.035
GPA
GMAT
3.5
3.0
2.5
350
400
450
500
550
600
650