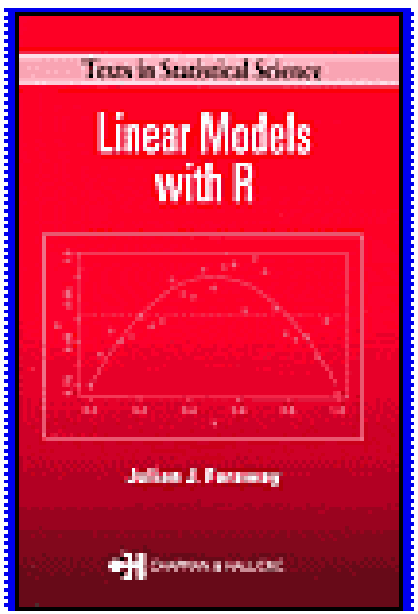
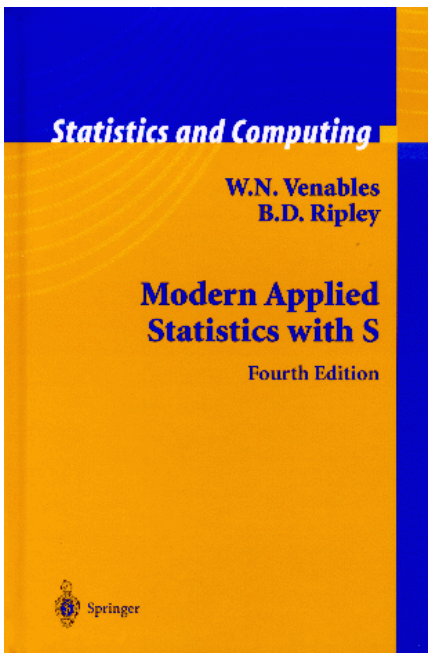
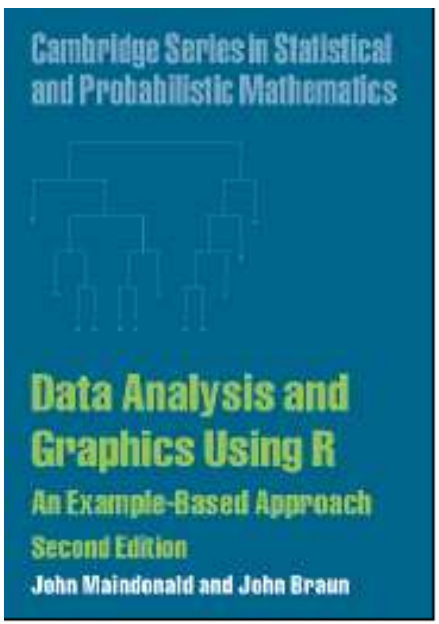
Administration

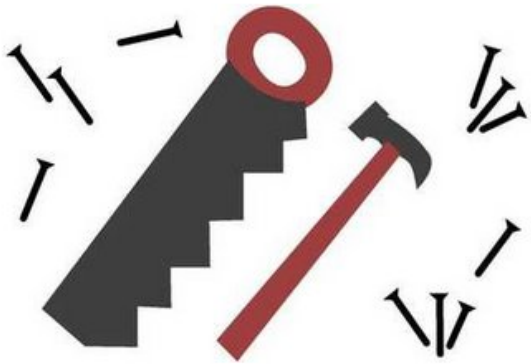
- Homework 1 on web page, due Feb 11
- NSERC summer undergraduate award applications due Feb 5
- Some helpful books







... administration



... administration

- collection of tools for regression and classification
- some old (least squares, discriminant analysis)
- some new (lasso, support vector machines)
- statistical justifications: loss, likelihood, mean squared error, classification error, posterior probabilities...
- statistical thinking
- framework for analysing new tools

Linear regression plus

- variable selection: forward, backward, stepwise, all possible subsets
- comparing models: adjusted R^2, C_p vs. p, AIC[1] and variants, K-fold cross-validation
- shrinkage methods: ridge regression, lasso, Least Angle Regression
- tuning parameter (amount of shrinkage): validation data, K-fold cross-validation
- derived variables: principal components regression, partial least squares

[1] C_p and AIC can be shown to be estimates of expected prediction error

Predictions with smoothing regression

- on training data: x's are centered and scaled when fitting Lasso, LAR, and ridge regression
- $\hat{\beta}_0$ is not included in shrinkage
- (3.41)

$$\hat{\beta}^{\text{ridge}} = \text{argmin}_\beta \sum (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

- (3.52)

$$\hat{\beta}^{\text{lasso}} = \text{argmin}_\beta \sum (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- $\hat{\beta}_0 = \bar{y} = \bar{y}_{train}$
- for predicting a new $y^0 = \hat{\beta}_0 + \sum_{j=1}^{p}(x_j^0 - \bar{x}_j)\hat{\beta}_j$: use $\hat{\beta}_0 = \bar{y}_{train}$, and $\bar{x}_j$ means average for jth feature on the training data
- see construction of `tx` and `mm` in `Table33R.txt`

...predictions

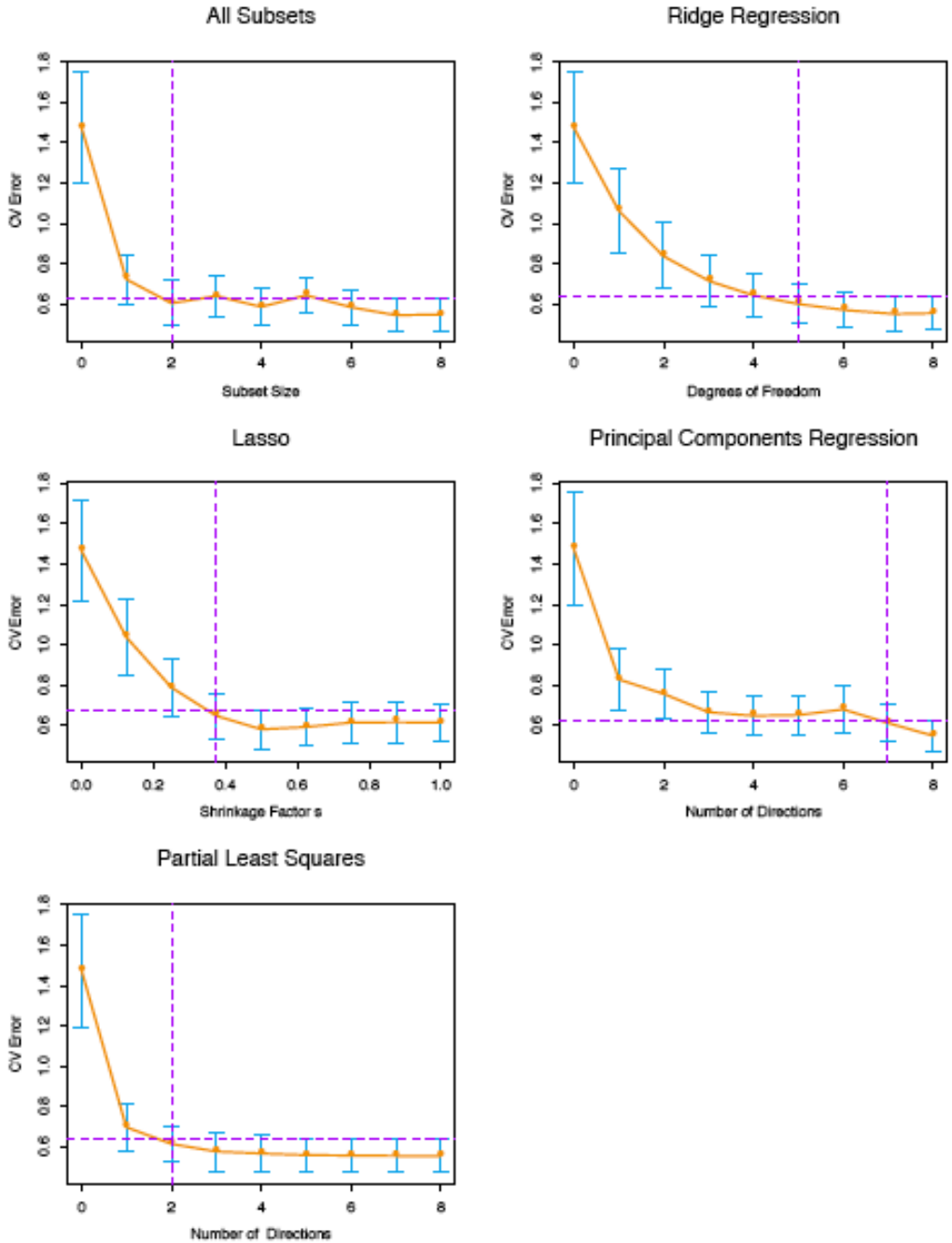
```
> options(digits=4)

> as.vector(predict.lars(pr.lars,newx=test[,1:8],type="fit",s=0.36, mode="fraction")$fit)
 [1] 2.094 1.443 1.780 2.292 2.695 2.009 2.283 1.731 1.969 1.694 2.588 2.486
[13] 2.712 2.492 2.554 2.345 2.053 2.954 3.189 1.976 2.946 3.035 2.746 2.550
[25] 2.636 2.697 3.135 3.095 3.233 3.698


> as.vector(tx %*% coef(pr.lars,s=0.36,mode="fraction"))+mean(train$lpsa)
 [1] 2.094 1.443 1.780 2.292 2.695 2.009 2.283 1.731 1.969 1.694 2.588 2.486
[13] 2.712 2.492 2.554 2.345 2.053 2.954 3.189 1.976 2.946 3.035 2.746 2.550
[25] 2.636 2.697 3.135 3.095 3.233 3.698
```

Derived features §3.5

- replace $\mathbf{x}_1, \ldots \mathbf{x}_p$ with linear combinations of columns
- principal components from SVD are natural candidates
- $X = UDV^T, \quad U^T U = I_N, \quad V^T V = I_p$
- $z_m = Xv_m, \quad m = 1, \ldots, M < p$
- z_m are orthogonal by construction
- $$\hat{y}^{pcr}_{(M)} = \bar{y}\mathbf{1} + \sum_{m=1}^{M} \hat{\theta}_m z_m$$
- $$\hat{\theta}_m = \frac{\langle z_m, y \rangle}{\langle z_m, z_m \rangle} = \frac{\sum_{i=1}^{n} z_{mi} y_i}{\sum_{i=1}^{n} z_{mi}^2}$$
- inputs should be scaled first (mean 0, variance 1)
- Angle brackets notation explained at (3.25).
  Exercise: $\bar{z}_m = 0$??

... derived features

- closely related method Partial least squares
- also constructs derived variables
- widely used in chemometrics, where often $p > N$
- see §3.6 for discussion


All Subsets
CV Error
Subset Size
Ridge Regression
CV Error
Degrees of Freedom
Lasso
CV Error
Shrinkage Factor s
Principal Components Regression
CV Error
Number of Directions
Partial Least Squares
CV Error
Number of Directions

Linear methods for classification

(Chapter 4)

- inputs $X = X_1, \ldots, X_p$ (notation x used on p.101)
- output Y takes values in one of K classes
- output G is a group label: values $1, \ldots, K$
- response Y as needed ($Y \leftrightarrow G$), e.g. $Y = 1, 0$ as $G =$ *blue*, *orange* Fig 2.1; eq. (2.7)
- data $(x_i, g_i), i = 1, \ldots N$
- goal to learn a model to predict the correct class for a future output, based on inputs

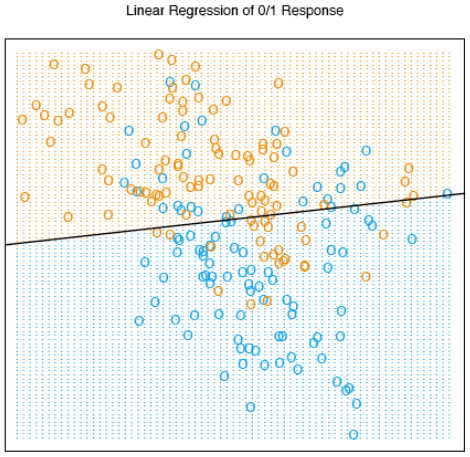


FIGURE 2.1. *A classification example in two dimensions. The classes are coded as a binary variable (BLUE = 0, ORANGE = 1), and then fit by linear regression. The line is the decision boundary defined by* $x^T\hat{\beta} = 0.5$. *The orange shaded region denotes that part of input space classified as ORANGE, while the blue region is classified as BLUE.*

code is in ElemStatLearn.pdf

Linear methods

- rule: $G = 2 \iff$ linear function of inputs $\geq$ something, else $G = 1$ (with extensions for $K > 2$)
- see Fig. 2.2 for a nonlinear method
- note that boundaries can be curved by including terms like X_j^2 and $X_j X_k$: see Fig. 4.1
- types of linear methods:
  - linear regression §4.2 $Y = \beta_0 + \beta^T X$
  - linear discriminant analysis §4.3
  - logistic regression §4.4

$$\log \frac{P(Y=1)}{P(Y=0)} = \beta_0 + \beta^T X$$

  - separating hyperplanes §4.5

Linear regression §4.2

- $$Y = \begin{pmatrix} y_{11} & \cdots & y_{1K} \\ y_{21} & \cdots & y_{2K} \\ \vdots & \vdots & \vdots \\ y_{N1} & \cdots & y_{NK} \end{pmatrix} = \begin{pmatrix} 0 & 0 & \ldots & 1 \\ 0 & 1 & \ldots & 0 \\ 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & \ldots & 0 \end{pmatrix}$$
- $y_i = (y_{i1}, \ldots, y_{iK})$ multivariate Bernoulli
- $E(Y \mid X) = XB$
- $\hat{B} = (X^T X)^{-1} X^T Y$: dimension?
- new observation $(1, x_0^T)$, new prediction $(1, x_0^T)\hat{B} = (\hat{f}_1, \ldots, \hat{f}_K)$: find the largest
- $$X = \begin{pmatrix} 1 & x_1^T \\ 1 & x_2^T \\ \vdots & \vdots \\ 1 & x_N^T \end{pmatrix}$$

... linear regression

- if $K = 1$ this is linear regression of $0/1$ response
- will produce predictions larger than 1, smaller than 0
- more natural to consider instead modelling $Pr(Y = 1 \mid X)$ and finding methods that predict in $(0, 1)$
- but if data supports $0.2 < Pr(Y = 1 \mid X) < 0.8$ results will not be very different
- Figure 4.2
- "targets" t_k (p. 104): $t_k = (0, \underbrace{\ldots, 1, \ldots,}_{k} 0)$
- find the largest $\{\hat{f}(x_0) - t_k\}^2$; another way to state the least squares solution

Discriminant analysis (§4.3)

- $G \in \{1, 2, \ldots, K\}$,
- $f_k(x) = f(x \mid G = k)$ =density of x in class k
- new ingredient: density of inputs
- Bayes Theorem:

$$\Pr(G = k \mid x) = \frac{f(x \mid G = k)\pi_k}{f(x)} \quad k = 1, \ldots, K$$

- associated classification rule: assign a new observation to class k if

$$\Pr(G = k \mid x) > \Pr(G = k' \mid x) \quad k' \neq k$$

(maximize the posterior probability)

special case - Normal

$$x \mid G = k \sim N_p(\mu_k, \Sigma_k)$$

$$\begin{aligned} & \Pr(G = k \mid x) \\ \propto \quad & \pi_k \frac{1}{(\sqrt{2\pi})^p |\Sigma_k|^{1/2}} \exp -\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1}(x-\mu_k) \end{aligned}$$

which is maximized by maximizing the log:

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

– if we further assume $\Sigma_k = \Sigma$, then

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

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

$$\Leftrightarrow \max_k \{\log \pi_k + x^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k\}$$

- Procedure: compute
  $\delta_k(x) = \log \pi_k + x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k$
- classify observation x to class k if $\delta_k(x)$ largest (see Figure 4.5, left)
- Estimate unknown parameters π_k, μ_k, Σ:

$$\hat{\pi}_k = \frac{N_k}{N}, \qquad \hat{\mu}_k = \sum_{i:g_i=k} \frac{x_i}{N_k}$$

$$\hat{\Sigma} = \sum_{k=1}^{K} \sum_{i:g_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T / (N - K)$$

- Figure 4.5, 4.1, 4.11

- Special case: 2 classes
- $\log \hat{\pi}_2 + x^T \hat{\Sigma}^{-1} \hat{\mu}_2 - \frac{1}{2} \hat{\mu}_2^T \hat{\Sigma}^{-1} \hat{\mu}_2 > (<)$
  $\log \hat{\pi}_1 + x^T \hat{\Sigma}^{-1} \hat{\mu}_1 - \frac{1}{2} \hat{\mu}_1^T \hat{\Sigma}^{-1} \hat{\mu}_1,$
- $\Leftrightarrow x^T \hat{\Sigma}^{-1} (\hat{\mu}_2 - \hat{\mu}_1) > (<)$
  $\frac{1}{2} \hat{\mu}_2 \hat{\Sigma}^{-1} \hat{\mu}_2 - \frac{1}{2} \hat{\mu}_1 \hat{\Sigma}^{-1} \hat{\mu}_1 + \log(N_1/N) - \log(N_2/N)$
- LHS is a linear combination of the inputs; RHS is the "cutpoint"
- If Σ_k not all equal, the discriminant function $\delta_k(x)$ defines a quadratic boundary; see Figure 4.6, left
- An alternative is to augment the original set of features with quadratic terms and use linear discriminant functions; see Figure 4.6, right

Another description of LDA (§4.3.2, 4.3.3)

- Let $\mathbf{W}$ = within class covariance matrix $\sum_{k=1}^{K} \sum_{i:g_i=k} (x_i - \bar{x}_k)(x_i - \bar{x}_k)^T$
- $\mathbf{B}$ = between class covariance matrix $\sum_{k=1}^{K} N_k (\bar{x}_k - \bar{x})(\bar{x}_k - \bar{x})^T$
- $$\mathbf{B} + \mathbf{W} = \sum_{k=1}^{K} \sum_{i:g_i=k} (x_i - \bar{x})(x_i - \bar{x})^T = \mathbf{T}$$
- linear classification rule $a^T x$

$$\max_a \frac{a^T \mathbf{B} a}{a^T \mathbf{W} a}$$

- equivalently

$$\max_a a^T \mathbf{B} a, \text{ subject to } a^T \mathbf{W} a = 1$$

... LDA

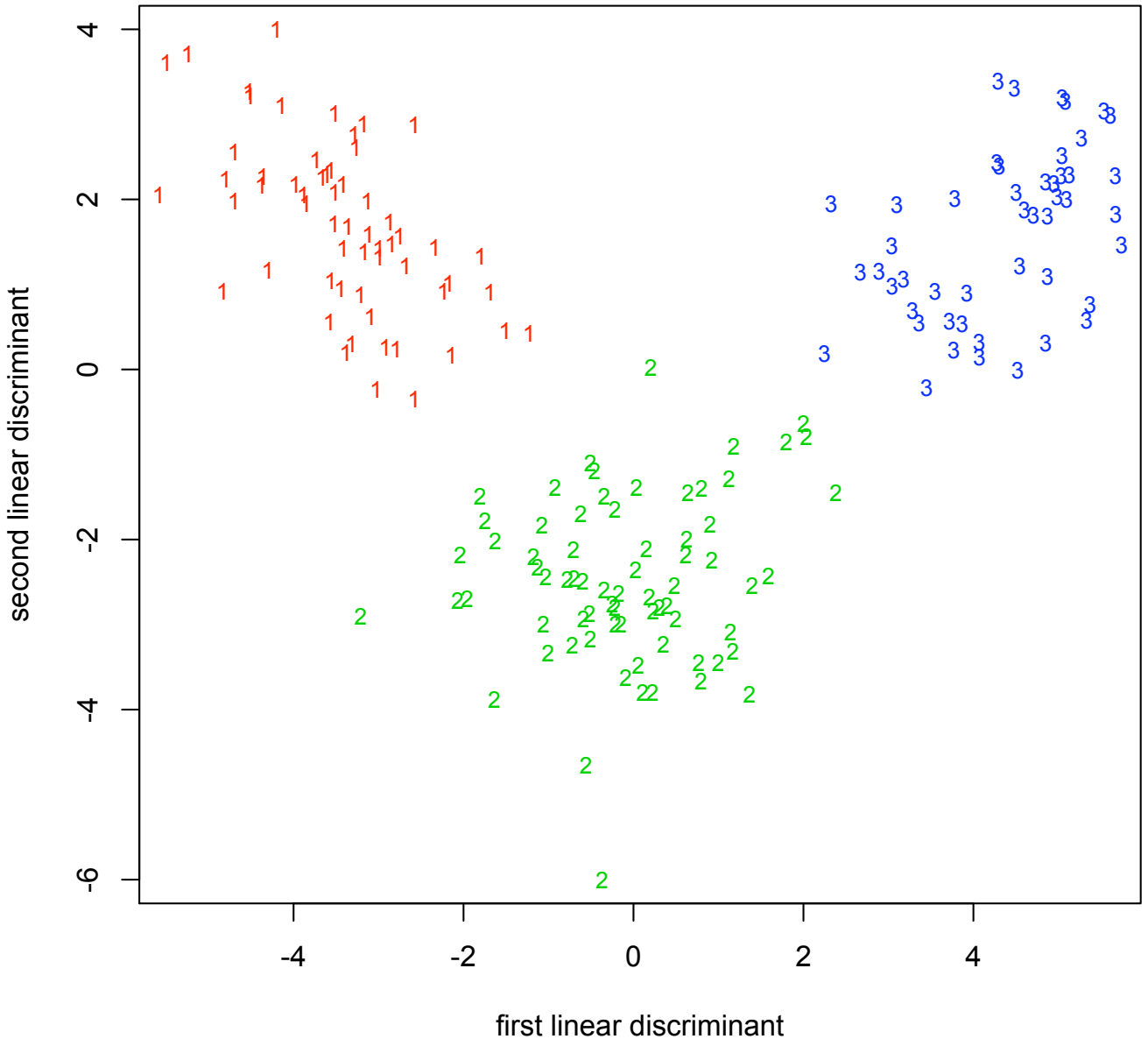
- Solution a_1, say, is the eigenvector of $\mathbf{W}^{-1}\mathbf{B}$ corresponding to the largest eigenvalue. This determines a line in R^p.
- continue, finding a_2, orthogonal (with respect to $\mathbf{W}$) to a_1, which is the eigenvector corresponding to the second largest eigenvalue, and so on.
- There are at most $\min(p, K-1)$ positive eigenvalues.
- These eigenvectors are the linear discriminants, also called canonical variates.
- This technique can be useful for visualization of the groups.
- Figure 4.11

... LDA

- (§4.3.3) write $\hat{\Sigma} = UDU^T$, where $U^T U = I$, D is diagonal (see p.109 for $\hat{\Sigma}$)
- $X^* = D^{-1/2} U^T X$, with $\hat{\Sigma}^* = I$
- classification rule is to choose k if $\hat{\mu}_k^*$ is closest (closest class centroid)
- only needs the K points $\hat{\mu}_k^*$, and the $K-1$ dimension subspace to compute this, since remaining directions are orthogonal (in the X^* space)
- if $K = 3$ can plot the first two variates (cf wine data)
- Figures 4.4 and 4.8
- algorithm on p.114

R code for the wine data

```
> library(MASS)
> wine.lda <- lda(class ~ alcohol + malic + ash + alcil + mag + totphen +
    flav + nonflav + proanth + col + hue + dil + proline, data = wine)
> wine.lda
Call:
lda.formula(class ~ alcohol + malic + ash + alcil + mag + totphen +
    flav + nonflav + proanth + col + hue + dil + proline, data = wine)

Prior probabilities of groups:
        1         2         3
0.3314607 0.3988764 0.2696629

Group means:
   alcohol    malic      ash    alcil      mag  totphen      flav  nonflav
1 13.74475 2.010678 2.455593 17.03729 106.3390 2.840169 2.9823729 0.290000
2 12.27873 1.932676 2.244789 20.23803  94.5493 2.258873 2.0808451 0.363662
3 13.15375 3.333750 2.437083 21.41667  99.3125 1.678750 0.7814583 0.447500
   proanth      col       hue      dil   proline
1 1.899322 5.528305 1.0620339 3.157797 1115.7119
2 1.630282 3.086620 1.0562817 2.785352  519.5070
3 1.153542 7.396250 0.6827083 1.683542  629.8958

Coefficients of linear discriminants:
                LD1           LD2
alcohol -0.403399781  0.8717930699
malic    0.165254596  0.3053797325
ash     -0.369075256  2.3458497486
alcil    0.154797889 -0.1463807654
mag     -0.002163496 -0.0004627565
totphen  0.618052068 -0.0322128171
flav    -1.661191235 -0.4919980543
nonflav -1.495818440 -1.6309537953
```


second linear discriminant
first linear discriminant
4
2
0
-2
-4
-6
-4
-2
0
2
4

Logistic regression

- data (x_i, g_i), $g_i = 1, 2$; equivalently (x_i, y_i), $y_i = 0, 1$
- natural starting point is Bernoulli distribution for y_i: $y_i = 1$ with probability $p_i = p_i(\beta)$
- likelihood function

$$L(\beta) = \prod_{i=1}^{N} p_i^{y_i}(1 - p_i)^{1-y_i}$$

- log-likelihood

$$\ell(\beta) = \Sigma_{i=1}^{N} y_i \log p_i(\beta) + (1 - y_i) \log\{1 - p_i(\beta)\}$$

- A common choice for $p_i(\beta)$ is the *logistic function*

$$p_i(\beta) = \frac{\exp(\beta^T x_i)}{1 + \exp(\beta^T x_i)}$$

x_i is a column vector: the ith row of X is x_i^T

Likelihood methods

- log-likelihood

$$\ell(\beta) = \Sigma_{i=1}^{N}\{y_i\beta^T x_i - \log(1 + e^{\beta^T x_i})\}$$

(constant term included in the set of inputs; β has length $p+1$)

- Maximum likelihood estimate of β:

$$\left.\frac{\partial \ell(\beta)}{\partial \beta}\right|_{\beta=\hat{\beta}} = 0 \iff \Sigma_{i=1}^{N} y_i x_{ij} = \Sigma_{i=1}^{N} p_i(\hat{\beta}) x_{ij}, \quad j = 1, \ldots, p$$

- Fisher information

$$-\left.\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T}\right|_{\beta=\hat{\beta}} = \Sigma_{i=1}^{N} x_i x_i^T \hat{p}_i(1 - \hat{p}_i)$$

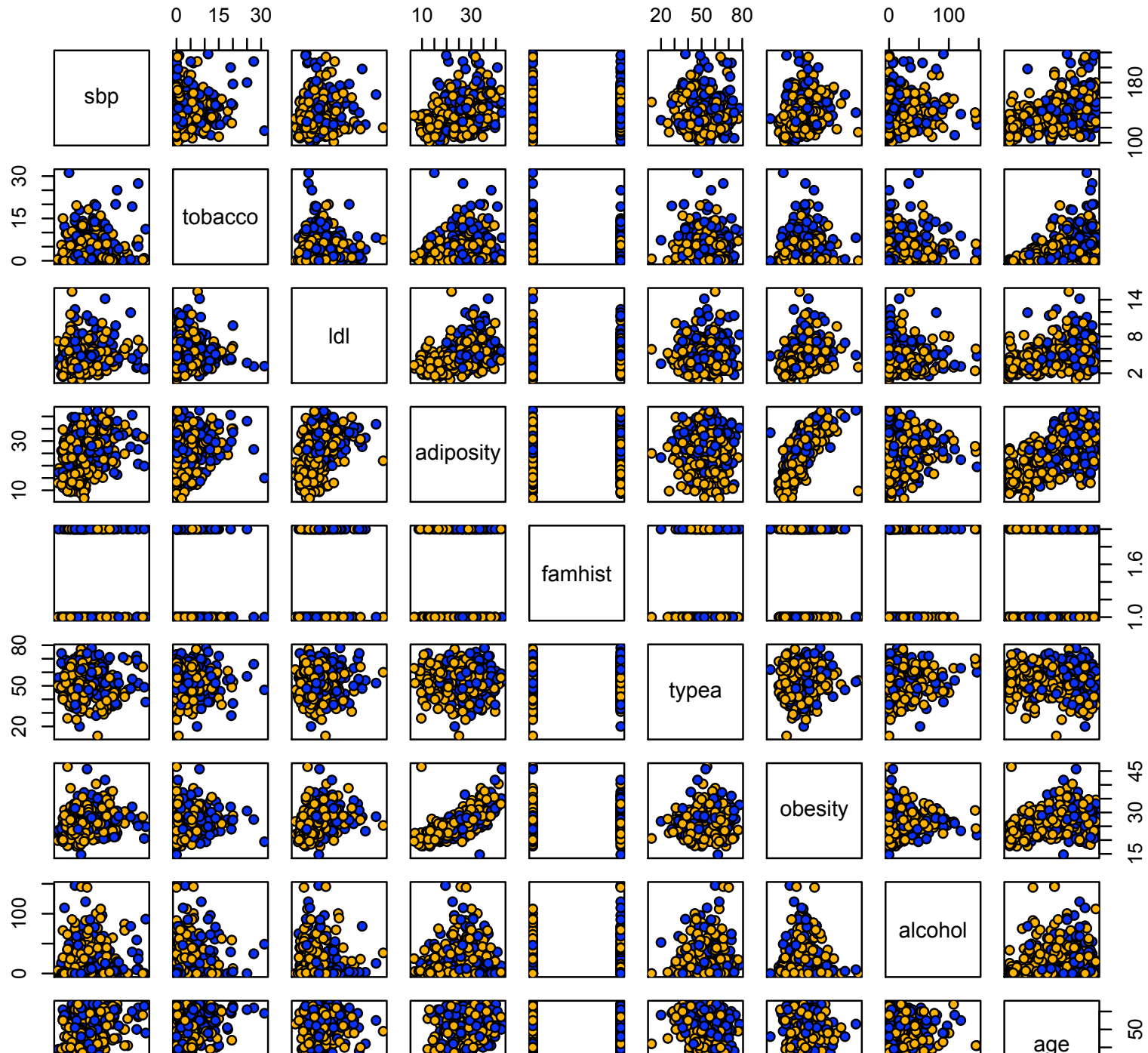
- Fitting: use an iteratively reweighted least squares algorithm; equivalent to Newton-Raphson; p.121
- Asymptotics: $\hat{\beta} \xrightarrow{d} N(\beta, \{-\ell''(\hat{\beta})\}^{-1})$

Inference

- Component: $\hat{\beta}_j \dot{\sim} N(\beta_j, \hat{\sigma}_j) \quad \hat{\sigma}_j^2 = [\{-\ell''(\hat{\beta})\}^{-1}]_{jj}$; gives a t-test (z-test) for each component
- $2\{\ell(\hat{\beta}) - \ell(\beta_j, \tilde{\beta}_{-j})\} \dot{\sim} \chi^2_{\dim \beta_j}$; in particular for each component get a χ^2_1, or equivalently
- $\text{sign}(\hat{\beta}_j - \beta_j)\sqrt{[2\{\ell(\hat{\beta}) - \ell(\beta_j, \tilde{\beta}_{-j})\}} \dot{\sim} N(0,1)$
- this gives two (asymptotically equivalent) ways to test if an input is statistically significant
- To compare 2 models $M_0 \subset M$ can use this twice to get $2\{\ell_M(\hat{\beta}) - \ell_{M_0}(\tilde{\beta}_q)\} \dot{\sim} \chi^2_{p-q}$ which provides a test of the adequacy of M_0
- LHS is the difference in (residual) deviances; analogous to SS in regression

Heart Data

```
> library(ElemStatLearn)
> data(SAheart)
> hr = SA.heart
> pairs(hr[1:9],pch=21,bg=c("red","green")[codes(factor(hr$chd))])
> hr.glm = glm(chd ~ ., data = hr, family=binomial)
> summary(hr.glm)

Call:
glm(formula = chd ~ ., family = binomial, data = hr)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7781  -0.8213  -0.4387   0.8889   2.5435

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)    -6.1507209  1.3082600  -4.701 2.58e-06 ***
sbp             0.0065040  0.0057304   1.135 0.256374
tobacco         0.0793764  0.0266028   2.984 0.002847 **
ldl             0.1739239  0.0596617   2.915 0.003555 **
adiposity       0.0185866  0.0292894   0.635 0.525700
famhistPresent  0.9253704  0.2278940   4.061 4.90e-05 ***
typea           0.0395950  0.0123202   3.214 0.001310 **
obesity        -0.0629099  0.0442477  -1.422 0.155095
alcohol         0.0001217  0.0044832   0.027 0.978350
age             0.0452253  0.0121298   3.728 0.000193 ***
---
Signif. codes:   *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 596.11  on 461  degrees of freedom
Residual deviance: 472.14  on 452  degrees of freedom
AIC: 492.14
```


sbp
tobacco
ldl
adiposity
famhist
typea
obesity
alcohol
age

extensions

- $E(y_i) = p_i$, $\text{var}(y_i) = p_i(1 - p_i)$ under Bernoulli
- Often the model is generalized to allow $\text{var}(y_i) = \phi p_i(1 - p_i)$; called over-dispersion
- Most software provides an estimate of ϕ based on residuals.

- if $y_i \sim Binom(n_i, p_i)$ same model applies
- $E(y_i) = n_i p_i$ and $\text{var}(y_i) = n_i p_i(1 - p_i)$ under Binomial

- Model selection uses AIC: $-2\ell(\hat{\beta}) + 2p$
- In R, use `glm` to fit logistic regression, `step` for model selection
- `glm` can be used for all exponential family models: uses iteratively reweighted least squares See also §4.4.3

step

```
> step(hr.glm)
Start:  AIC=492.14
chd ~ sbp + tobacco + ldl + adiposity + famhist + typea + obesity +
    alcohol + age

            Df Deviance    AIC
- alcohol    1   472.14 490.14
- adiposity  1   472.55 490.55
- sbp        1   473.44 491.44
<none>           472.14 492.14
- obesity    1   474.23 492.23
- ldl        1   481.07 499.07
- tobacco    1   481.67 499.67
- typea      1   483.05 501.05
- age        1   486.53 504.53
- famhist    1   488.89 506.89

Step:  AIC=490.14
chd ~ sbp + tobacco + ldl + adiposity + famhist + typea + obesity +
    age

            Df Deviance    AIC
- adiposity  1   472.55 488.55
- sbp        1   473.47 489.47
<none>           472.14 490.14
- obesity    1   474.24 490.24
- ldl        1   481.15 497.15
- tobacco    1   482.06 498.06
- typea      1   483.06 499.06
- age        1   486.64 502.64
- famhist    1   488.99 504.99
```

... step

```
Step:  AIC=488.55
chd ~ sbp + tobacco + ldl + famhist + typea + obesity + age

          Df Deviance    AIC
- sbp      1   473.98 487.98
<none>         472.55 488.55
- obesity  1   474.65 488.65
- tobacco  1   482.54 496.54
- ldl      1   482.95 496.95
- typea    1   483.19 497.19
- famhist  1   489.38 503.38
- age      1   495.48 509.48

Step:  AIC=487.98
chd ~ tobacco + ldl + famhist + typea + obesity + age

          Df Deviance    AIC
- obesity  1   475.69 487.69
<none>         473.98 487.98
- tobacco  1   484.18 496.18
- typea    1   484.30 496.30
- ldl      1   484.53 496.53
- famhist  1   490.58 502.58
- age      1   502.11 514.11
```

... step

```
Step:  AIC=487.69
chd ~ tobacco + ldl + famhist + typea + age

          Df Deviance    AIC
<none>         475.69 487.69
- ldl      1   484.71 494.71
- typea    1   485.44 495.44
- tobacco  1   486.03 496.03
- famhist  1   492.09 502.09
- age      1   502.38 512.38

Call:  glm(formula = chd ~ tobacco + ldl + famhist + typea + age, family = binomial,      data

Coefficients:
   (Intercept)         tobacco             ldl  famhistPresent
      -6.44644         0.08038         0.16199         0.90818
         typea             age
       0.03712         0.05046

Degrees of Freedom: 461 Total (i.e. Null);  456 Residual
Null Deviance:     596.1
Residual Deviance: 475.7  AIC: 487.7
```

Interpretation of coefficients

- e.g. tobacco (measured in kg): coeff= 0.081
- $\text{logit}\,\{p_i(\beta)\} = \beta^T x_i$
- increase in one unit of x_{ij}, say, leads to increase in $\text{logit}\ p_i$ of 0.081
- increase in $p_i/(1-p_i)$ of $\exp(0.081) = 1.084$.
- estimated s.e. 0.026, $\text{logit}\ p_i \pm 0.026$, $\exp(0.081 + 2 \times .026), \exp(0.081 - 2 \times .026))$ is $(1.03, 1.14)$.
- similarly for age: $\hat{\beta}_j = 0.044$; increased odds 1.045 for 1 year increase
- prediction of new values to class 1 or 0 according as $\hat{p} > (<) 0.5$

L_1 regularization

- eqn (4.31)

$$\max_{\beta_0,\beta} \left\{ \sum_{i=1}^{N} \left[y_i(\beta_0 + \beta^T x_i) - \log(1 + e^{\beta_0+\beta^T x_i}) \right] - \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

- note that intercept has been separated out
- Figure 4.13
- interpretation?
- recently developed software: `glmpath` and `glmnet`

Notes

- how to choose between logistic regression and discriminant analysis?
- classification error on the training data is overly optimistic
- logistic regression and generalizations to K classes doesn't assume any distribution for the inputs
- discriminant analysis more efficient if the assumed distribution is correct
- warning: in §4.3 x and x_i are $p \times 1$ vectors, and we estimate β_0 and β, the latter a $p \times 1$ vector
- in §4.4 they are $(p+1) \times 1$ with first element equal to 1 and β is $(p+1) \times 1$.

Misclassification

```
> hr.glm.class = predict(hr.glm)>0
> table(hr.glm.class,hr$chd)

hr.glm.class   0   1
       FALSE 256  77
       TRUE   46  83
> hr.glm2 = glm(chd ~ tobacco + ldl + famhist + age, data = hr, family=binomial)
> table(predict(hr.glm2)>0, hr$chd)

         0   1
  FALSE 254  76
  TRUE   48  84
> hr.lda = lda(chd ~ ., data = hr)
> table(predict(hr.lda)$class,hr$chd)

     0   1
  0 258  73
  1  44  87
```