Multivariate and other worksheets for R (or S-Plus): a miscellany

P.M.E.Altham, Statistical Laboratory, University of Cambridge.

March 26, 2013

Contents

Preface

These worksheets were originally constructed for my graduate teaching before I retired in September 2005. Since then, I have a lot more time to be interested in lots more things! So I am continually (but slowly) adding more examples and graphs, and I have also made some minor editorial changes: this version of the worksheets includes a small index of the **commands** used, and also of the datasets.
If you have any comments or queries, please contact me at

`p.m.e.altham@statslab.cam.ac.uk`

`http://www.statslab.cam.ac.uk/~pat`

Special thanks must go to Dr R.J.Gibbens for his help in introducing me to S-Plus, and also to Professor B.D.Ripley for access to his S-Plus lecture notes. Several generations of keen and critical students for the Cambridge University Diploma in Mathematical Statistics, and since 1998 for the MPhil in Statistical Science, have made helpful suggestions which have improved these worksheets.

Most of the multivariate theory used is explained in my Lecture Notes at `http://www.statslab.cam.ac.uk/~pat/AppMultNotes.pdf`
These worksheets form a companion set to "Introduction to S-Plus for Generalized Linear Modelling", or (more recently) to my R worksheets for a similar course, which are available at `http://www.statslab.cam.ac.uk/~pat/redwsheets.pdf`
Nearly all of the examples given below will work in R, the free software (see link on my webpage).
Aristotle said
'For the things we have to learn before we can do them, we learn by doing them.'
This is a quotation I found at the start of the book by B.J.T.Morgan, 'Applied Stochastic Modelling', published by Arnold (2000).

References.
Agresti, A. (2002) Categorical Data Analysis. New York: Wiley.
Venables, W.N. and Ripley, B.D. (2002) Modern Applied Statistics with S-Plus. New York: Springer-Verlag. Also, any of the 3 previous editions this book.
Webb, A. (1999) Statistical Pattern Recognition. London: Arnold (this shows the relevance of multivariate analysis to the topic of Statistical Pattern Recognition.)
Note added April 2008 A very interesting article by Michael Friendly, which has a good online dataset and some marvellous graphics, is
'A.-M.Guerry's *Moral Statistics of France*: Challenges for Multivariable Spatial Analysis', in *Statistical Science*, **22**, 368-399.

This is based on a nineteenth-century dataset.
Note added May 2009 A very topical application of both Principal components analysis and hierarchical clustering is given in
Bollen J, Van de Sempel H, Hagberg E (2009) 'A principal component analysis of 39 scientific impact measures', see `http://xxx.lanl.gov/abs/0902.2183`.
This paper analyses 39×39 correlation matrix between 39 'scientific impact measures', one of which is the citation Impact Factor. The authors note that 'the notion of scientific impact is a multi-dimensional construct that cannot be adequately measured by any single indicator'.
Another way of rating the influence of journal is the 'Eigenfactor', which is not based on PCA but on the leading eigenvector of a particular Markov transition matrix, see `www.eigenfactor.org`

Chapter 1

Classical Statistics and Introduction to non-parametric methods

The 2-sample t-test and the 2-sample Wilcoxon test
Notation: let $(x_1, x_2, \ldots, x_m)$ and $(y_1, y_2, \ldots, y_n)$ be independent random samples from the distribution functions $F(.), G(.)$ respctively.
If we know that F, G correspond respectively to $N(\mu_1, \sigma^2)$ and $N(\mu_2, \sigma^2)$ then the optimum test of
$H_0 : \mu_1 = \mu_2$ against the alternative $H_1 : \mu_1 < \mu_2$ is achieved by the '2-sample t-test', and here is an example, for a very small and obvious set of data.

```
>x <- scan()
3.7 2.1 4.5 7.1

>y<- scan()
 6.1 7.9 10.3 11.4 13.7

>summary(x)
   Min. 1st Qu. Median Mean 3rd Qu. Max.
     2.1     3.3    4.1 4.35    5.15  7.1

>summary(y)
      Min. 1st Qu. Median Mean 3rd Qu. Max.
        6.1     7.9   10.3 9.88    11.4 13.7
>t.test(x,y, alt ="less")

  Standard Two-Sample t-Test

data:  x and y
t  = -3.1364, df  = 7, p-value  = 0.0082
alternative hypothesis: true difference in means is less than 0
```

```
  95 percent confidence interval:
  NA -2.189557

sample estimates:
 mean of x mean of y
   4.35      9.88
```

Thus here we reject H_0 in favour of H_1. Observe that here, $\mu_1 < \mu_2$ implies that $F(x) > G(x)$ for all x, ie the x's tend to be smaller than the y's.
But what if we want to test $HP_0 : F(x) = G(x)$ for all x against the alternative $HP_1 : F(x) > G(x)$ for all x, without making a specific assumption about the *shape* of F, G?
It turns out that we can get a long way (and in fact produce tests that are really rather efficient) simply by considering the *ranks* of the observations x_i, y_j.
This is what 'nonparametric' (or more accurately, 'distribution-free') statistical tests achieve, and as such they have a long history.
First, we find the ranks of $(x), (y)$ in the *combined sample*, which has $4 + 5 = 9$ elements.

```
> rank(c(x,y))
 [1] 2 1 3 5 4 6 7 8 9
```

Then we find W, the sum of the ranks of $(x_1, \ldots, x_m)$ in the combined sample; here $W = 2 + 1 + 3 + 5 = 11$. We reject HP_0 in favour of HP_1 if W is sufficiently SMALL, say if $W \leq c$, where $P(W \leq c|HP_0) = .05$, say.
The beauty of non-parametric statistics is that we can compute the 'null distribution' of W purely from m, n, the respective sample sizes.

```
>wilcox.test(x,y, alt ="less")

    Exact Wilcoxon rank-sum test

     data:  x and y
    rank-sum statistic W  = 11, n  = 4, m  = 5, p-value  = 0.0159
    alternative hypothesis: true mu is less than 0
```

How is the p-value computed? Note that under the null hypothesis, we can say by symmetry that all the orders of the $x_1, \ldots, x_m, y_1, \ldots, y_n$ are equally likely, and each such order must therefore have probability

$$1/\binom{m+n}{m} = q \text{ say.}$$

Here's how we build up the null distribution of W.
You can check that by definition, $q = .007936$. Further, by definition, $W \geq 10$, and $W = 10 (= 1 + 2 + 3 + 4)$ with probability q.
And $W = 11 = (1 + 2 + 3 + 5)$ with probability q also, hence

$$P(W \leq 11|HP_0) = 2 \times q = .0159.$$

Note that in general (for reasonable sorts of distributions, in fact) the non-parametric test is **conservative** with respect to the corresponding t-test (we are throwing away some data by using only ranks) so that we should expect that the non-parametric test will have a larger p-value than the corresponding t-test.

Now we consider a new problem, **tests for paired samples**.
Suppose we have data $(x_1, y_1), \ldots, (x_n, y_n)$, a random sample from the bivariate distribution function $F(x, y)$. We wish to test the hypothesis $HP_0 : F(x, y) = F(y, x)$ for all x, y against the alternative hypothesis HP_{alt} that the x's tend to be smaller than the corresponding y's. In the example given below, $n = 6$, and it is fairly obvious that the x's tend to be less than the y's, but the sample size is rather small. Now we know that if $F(.,.)$ is bivariate normal, then the optimum test of HP_0 against HP_{alt} is the paired sample t-test, carried out as follows:

```
> cbind(x,y,y-x)
        x     y
[1,] 12.3 12.43  0.13
[2,] 14.4 14.71  0.31
[3,]  2.3  2.97  0.67
[4,]  5.1  5.98  0.88
[5,]  6.7  6.12 -0.58
[6,]  9.1  9.99  0.89

>summary(y-x)
 Min. 1st Qu. Median   Mean 3rd Qu. Max.
-0.58   0.175   0.49 0.3833  0.8275 0.89


 > t.test(x,y,paired =T,alt = "less")

 Paired t-Test

 data:  x and y
  t  = -1.6687, df  = 5, p-value  = 0.078
  alternative hypothesis: true mean of differences is less than 0
  95 percent confidence interval:
          NA 0.07956285
  sample estimates:
  mean of x - y
       -0.3833333
```

Hence, the corresponding p-value is 0.078, so that at level 10% we reject HP_0 in favour of HP_{alt}.
How can we carry out the corresponding test if we make no assumption about the shape of $F(.,.)$?
Here's the way we do it.
Put $z_i = y_i - x_i$, then $z_1, \ldots, z_n$ is a random sample from the distribution function $G(.)$ say.
We test $H_0 : G(z) = 1 - G(-z)$, ie G corresponds to a pdf symmetric about 0,
against H_1, G corresponds to a pdf symmetric about a point > 0.

So, we compute $z_i = y_i - x_i$, find the ranks of $|z_i|, 1 \leq i \leq n$
and then compute as our test statistic V, defined as the sum of the ranks of the $|z_i|$ for which $z_i < 0$.

```
 > rank(abs(y-x))
[1] 1 2 4 5 3 6
 > abs(y-x)
[1] 0.13 0.31 0.67 0.88 0.58 0.89

 > wilcox.test(x,y,paired =T,alt ="less")

        Exact Wilcoxon signed-rank test

data:  x and y
signed-rank statistic V  = 3, n  = 6, p-value  = 0.0781
alternative hypothesis: true mu is less than 0
```

How is the p-value computed?
Here it is $P(V \leq 3|H_0)$ and so we see that it is $P(V = 0, 1, 2 \text{ or } 3|H_0)$.
Let $M =$ number out of $z_1, \ldots, z_n$ which are < 0. Then it can easily be seen that on H_0, M is distributed as $Bi(n, 1/2)$.
Hence ... it can be shown that, on H_0,

$$P(V = 0) = 1/2^6 = P(V = 1) = P(V = 2)$$

and

$$P(V = 3) = P(V = 1 + 2 \text{ or } V = 3) = 1/2^6 + 1/2^6$$

giving $P(V \leq 3|H_0) = 5/2^6 = .0781$ as given.
This way we can build up the null distribution of V, our test statistic, without even knowing the parent distribution $G()$.
For large n the asymptotic null distribution of V is normal, with mean and variance which are known functions of n, and a corresponding result holds for the 2-sample Wilcoxon test. You will find that R and SPlus use these asymptotic results to compute the p-values for large sample sizes.
Here is a very quick illustration, on the same very small sample, of **bootstrap methods**, here used to find 2 slightly different versions of a 95% confidence interval for the mean.

```
>z # this is our sample, of size 6.
[1]  0.13  0.31  0.67  0.88 -0.58  0.89
>t.test(z) # this will give a 95% confidence interval for mu,
# based on the assumption that the z's form a random sample
# from a Normal distribution, mean mu.
        One Sample t-test

data:  z
t = 1.6687, df = 5, p-value = 0.1560
alternative hypothesis: true mean is not equal to 0
```

```
95 percent confidence interval:
 -0.2071798  0.9738465
sample estimates:
mean of x
0.3833333
```

Now we will use the bootstrap library to find our confidence intervals for the mean. This construction does not depend on the assumption of normality. We generate 1000 bootstrap samples, and compute the mean for each such sample. Each sample is drawn *with* replacement from the original $z_1, \ldots, z_6$.

```
>library(boot)
>set.seed(1.7) # the arbitrary choice 1.7 ensures we get the same result each time
> z.boot = boot(data=z, statistic = function(x,i) mean(x[i]) , R =1000)
> boot.ci(z.boot, type=c("perc", "bca"))

BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 1000 bootstrap replicates

CALL :
boot.ci(boot.out = z.boot, type = c("perc", "bca"))

Intervals :
Level     Percentile            BCa
95%   (-0.0467,  0.7567 )   (-0.1350,  0.7183 )
Calculations and Intervals on Original Scale
```

Now a different example for you to try. The datafile
taxrevenue
contains, as rows, the taxrevenue for sales of
tobacco, spirits, beer, wine, cider and sherry, betting.
The columns are 1989-90,1990-91,1991-92.
These data are from " The Independent" Aug 18, 1993. (Note, data from a newspaper, while interesting and topical, does not usually constitute a "random sample". We press on regardless.)
Here is the dataset taxrevenue

```
  x       y       z
5035.3  5636.0  6289.5
1513.5  1703.0  1742.1
2074.2  2290.0  2324.9
 791.2   855.3   924.5
  58.8    68.6    73.8
 976.1  1006.4  1052.8
```

These data provide an opportunity for a tour through some S-Plus classical tests.

```
tax <-  read.table("taxrevenue", header  = T)
tax
```

```
attach(tax)
a <- (y-x)/x ; b <- (z-y)/y # we compare relative increases.
#first, one-sample tests on a.
a ; summary(a)
t.test(a,mu=.10)
names(t.test(a))
t.test(a)$conf.int
# Now the nonparametric version of this.
wilcox.test(a, mu =.10)
t.test(a,mu =.01) ; wilcox.test(a,mu =.01) #compare p-values.
```

Now compare a with b, but pretending a, b independent of each other.

```
t.test(a,b)             # This assumes the 2 variances are equal.
t.test(a,b,var.equal =F) # This doesn't assume the 2 variances equal.
```

But, the above were WRONGLY applied: they assumed independent a, b. So now we do it correctly, ie allowing for the PAIRING of a, b.

```
t.test(a,b,paired =T)
wilcox.test(a,b,paired =T)
```

Next we demonstrate 2 methods of testing the independence of a, b. The first, which uses the Pearson correlation coefficient, is effectively assuming that we have a random sample from a bivariate normal distribution. The second, constructed by Spearman in the context of intelligence-testing, tests for independence of a, b without making any assumption on the parent distribution $F(a, b)$, this is therefore a non-parametric test. It essentially replaces a_i, b_i by their ranks, eg $(1, 3), \ldots, (n, n)$ and works out the corresponding Pearson coefficient. On the null hypothesis of independence of a, b this has known distribution, depending only on n, the sample size.

```
cor.test(a,b)
cor.test(a,b,method ="spearman")
help(friedman.test) # experiment with this new non-parametric test.
# can you apply it to the data x, y, z ?
# Try some plots. Do they enlighten you ?
i <- 1:3 ; ttax <- t(tax)
matplot(i, ttax, type ="l") # might help
```

Here is another dataset, this time from The Independent, June 30, 1999, on the safety of multi-purpose vehicles (MPV's). The 8 types of vehicle were subjected to 'Front Impact' tests (in which the front impact takes place at 40mph (64kph)) and 'Side Impact' tests, in which the side impact takes place at 30mph (50kph)). The corresponding scores are given in the Table below: the higher the score, the better the vehicle.

	Frontal_score(%)	Side_Score(%)
RenaultEspace	67	100
ToyotaPicnic	61	93
Peugeot806	42	93
NissanSerena	34	100
VolkswagenSharan	36	96

MitsubishiSpWagon	24	96
Opel/VauxhallSintra	21	93
ChryslerVoyager	0	89

Questions for you:
i) Is the Frontal Score significantly less than the Side Score?
ii) Is there a positive association between these two scores?

'How the world is getting hungrier each year' is the headline in The Independent of 26 November, 2003, showing the following distressing data: for the following 40 countries, the percentage of the population that is undernourished, for the years 1999-2001, and for 1990-92.

		y99-01	y90-92
1	DRCongo	75	31
2	Somalia	71	68
3	Burundi	70	49
4	Afghanistan	70	58
5	Eritrea	61	63*
6	Mozambique	53	69
7	SierraLeone	50	46
8	Zambia	50	45
9	Haiti	49	65
10	Angola	49	61
11	CAR	44	50
12	Tanzania	43	35
13	Ethiopia	42	57*
14	Liberia	42	33
15	Rwanda	41	43
16	Zimbabwe	39	43
17	Mongolia	38	34
18	Cambodia	38	43
19	Kenya	37	44
20	Madagascar	36	35
21	Niger	34	42
22	Chad	34	58
23	NKorea	34	18
24	Yemen	33	35
25	Malawi	33	49
26	Bangladesh	32	35
27	Congo	30	37
28	Nicaragua	29	30
29	Guinea	28	40
30	PNewGuinea	27	25
31	Cameroon	27	33
32	Gambia	27	22
33	Iraq	27	7
34	Panama	26	20

```
 35 Guatemala   25      16
 36 Lesotho     25      27
 37 Togo        25      33
 38 DominicanR  25      27
 39 Sudan       25      31
 40 SriLanka    25      29
```

* corresponds to 1995-97, as the earlier figure was unavailable.

New for July 2008: The Tompkins Table for Cambridge Colleges Examinations results, 2000 − 2008. Each year The Independent publishes the examination rank order of the 29 Cambridge Colleges: Emmanuel has been at the top of the Table for each 2006 and 2007, but now (ie 2008) Selwyn is top.
Here is the Table of ranks for each of the last 9 year (note that certain colleges were only included in this Table from 2003 onwards). Suggestions for a non-parametric test, and a plot of the various college 'tracks' over the 8 years, are given below. First, here is the dataset.

```
College        y00  y01  y02  y03  y04  y05  y06  y07 y08
Christs         1   1   4   2   2   4   6   2 8
Churchill      15   9  10   9  19  18  13  15 6
Clare           9   6   3   6   4   9  12  17 13
CorpusC        10  20  18   7  10  16   8   8 9
Downing         8  10   8  12  17  15  11   3 12
Emmanuel        3   2   2   1   1   5   1   1 2
Fitzwilliam    21  13  20  20  15  13  19  14 21
Girton         18  17  16  17  25  24  22  21 22
G&Caius        12   8   7   4   5   2   2  10 4
Homerton       NA  NA  NA  25  24  26  25  26 25
HughesH        NA  NA  NA  27  27  29  29  29 26
Jesus          13  11   9  10   9   7  10   9 7
Kings          20  21  14  16  20  10  17  18 19
LucyC          NA  NA  NA  26  26  27  26  24 28
Magdalene      22  22  15  18  22  20  20  13 5
NewHall        16  23  24  24  23  25  24  23 23
Newnham        24  24  22  21  13  21  23  22 24
Pembroke        6   7   1   3   6   6   4   7 10
Peterhouse     14  19  23  22  21  22  21  25 17
Queens          5   5   5   5   8   8  14  11 16
Robinson       19  14  21  23  16  11  18  20 18
StCaths        11  18  12  11   7   1   3   5 11
StEdmunds      NA  NA  NA  29  29  28  28  28 29
StJohns         4   4  11  13  14  12  15  19 20
Selwyn          7  12  13  14  11  19   7   4 1
SidneyS        23  16  19  15  18  14   9  12 14
```

```
Trinity         2  3  6  8  3  3  5  6 3
TrinHall       17 15 17 19 12 17 16 16 15
Wolfson        NA NA NA 28 28 23 27 27 27
```

```
Tompkins <- read.table("Tompkins", header=T)
Tompkins <-Tompkins[-c(10,11,14,23,29),]#to remove the incomplete rows
matTomp <- as.matrix(Tompkins[,2:10])
friedman.test(t(matTomp)) #
```

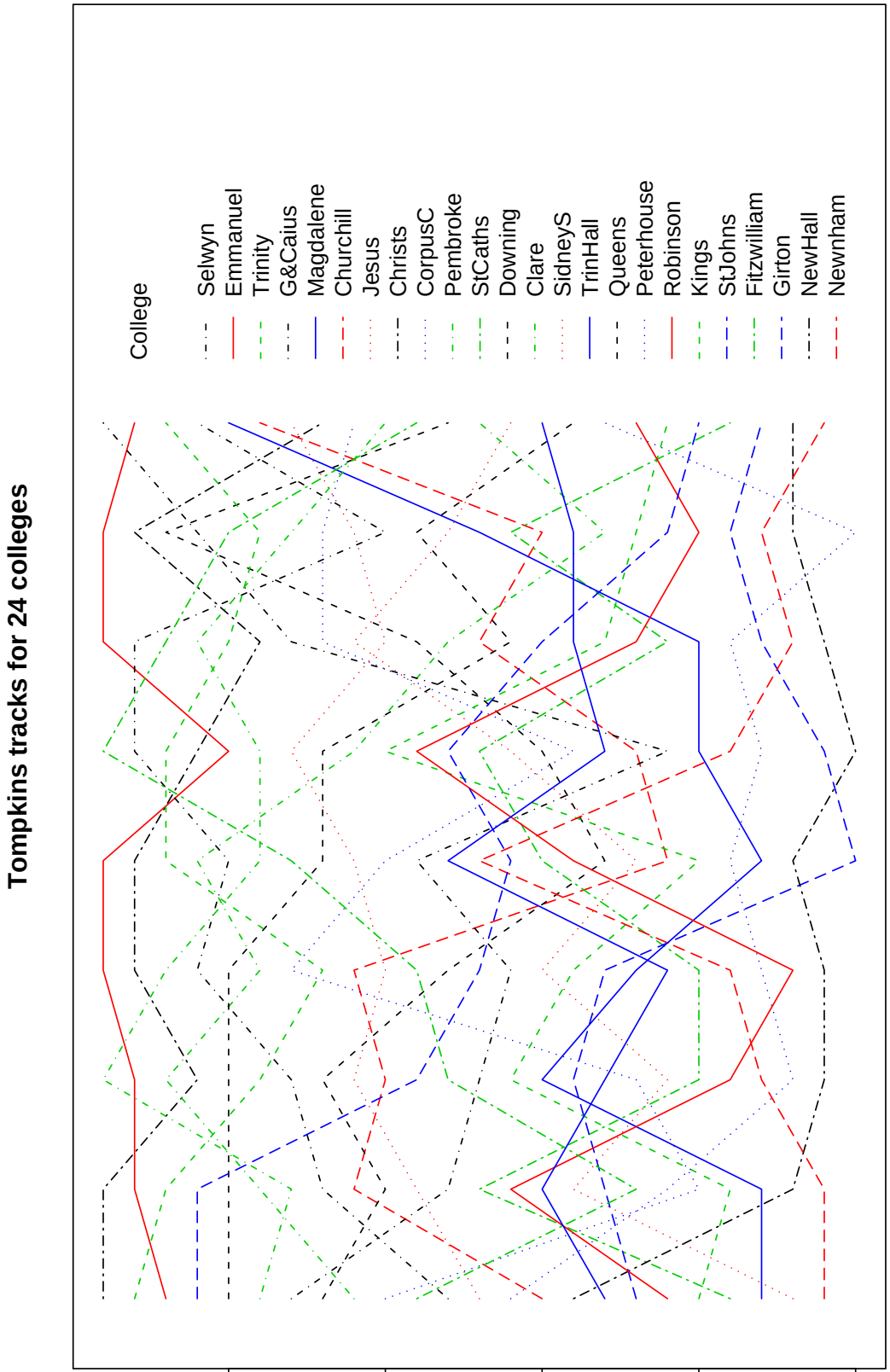
Note that we transpose the matrix in order to test for the differences between the 24 colleges. The Friedman test results in a chi-squared statistic of 150.82 on 23 df, apparently showing that there are indeed systematic differences between these 24 colleges. However, this use of the Friedman test may not be strictly valid, since consecutive years ('blocks' in the parlance of the Friedman test) will not be **independent**. Each Tompkins score, for a given year and a given college, is obtained from the examination results of students from years 1, 2 and 3 of that college. Thus typically a particular student, arriving in say autumn 2001, will contribute to the scores of his/her college in 2002, 2003 and 2004.

```
> round(cor(matTomp),2)
     y00  y01  y02  y03  y04  y05  y06  y07  y08
y00 1.00 0.80 0.76 0.75 0.70 0.55 0.66 0.62 0.50
y01 0.80 1.00 0.83 0.75 0.69 0.65 0.62 0.54 0.47
y02 0.76 0.83 1.00 0.91 0.72 0.71 0.72 0.67 0.60
y03 0.75 0.75 0.91 1.00 0.78 0.71 0.83 0.73 0.65
y04 0.70 0.69 0.72 0.78 1.00 0.81 0.79 0.67 0.55
y05 0.55 0.65 0.71 0.71 0.81 1.00 0.78 0.62 0.45
y06 0.66 0.62 0.72 0.83 0.79 0.78 1.00 0.86 0.78
y07 0.62 0.54 0.67 0.73 0.67 0.62 0.86 1.00 0.77
y08 0.50 0.47 0.60 0.65 0.55 0.45 0.78 0.77 1.00
```

This does indeed fit in with the suggestion of positive correlation between successive years. In this case I suspect that the Friedman test statistic of 150.82 should actually be 'deflated' by a suitable factor (but what is this?) before referring it to the χ^2 distribution. Now we show a method of plotting the 'time tracks' of the 24 colleges.

```
college <- Tompkins[,1] # to set up college names
# we could use 'matplot' to plot the tracks of the individual colleges,
# but 'interaction.plot' turns out to be quicker to use
v <- as.vector(matTomp) # this reads DOWN the rows
College <- gl(24,1, length=216, labels = college)
Year <- gl(9,24, length=216, labels=2000:2008)
y <- 25-v # to make graph give 'best' college at the TOP
interaction.plot(Year, College, y, col=c("black", "red", "green3", "blue"), ylab="")
```

This results in Figure 1.1. Some of the middling 'tracks' do seem to go all over the place.
Following a suggestion by Dr Richard Gibbens, we could also also plot the 'tracks' another way,


Tompkins tracks for 24 colleges
College
Selwyn
Emmanuel
Trinity
G&Caius
Magdalene
Churchill
Jesus
Christs
CorpusC
Pembroke
StCaths
Downing
Clare
SidneyS
TrinHall
Queens
Peterhouse
Robinson
Kings
StJohns
Fitzwilliam
Girton
NewHall
Newnham
2000
2001
2002
2003
2004
2005
2006
2007
2008
Year

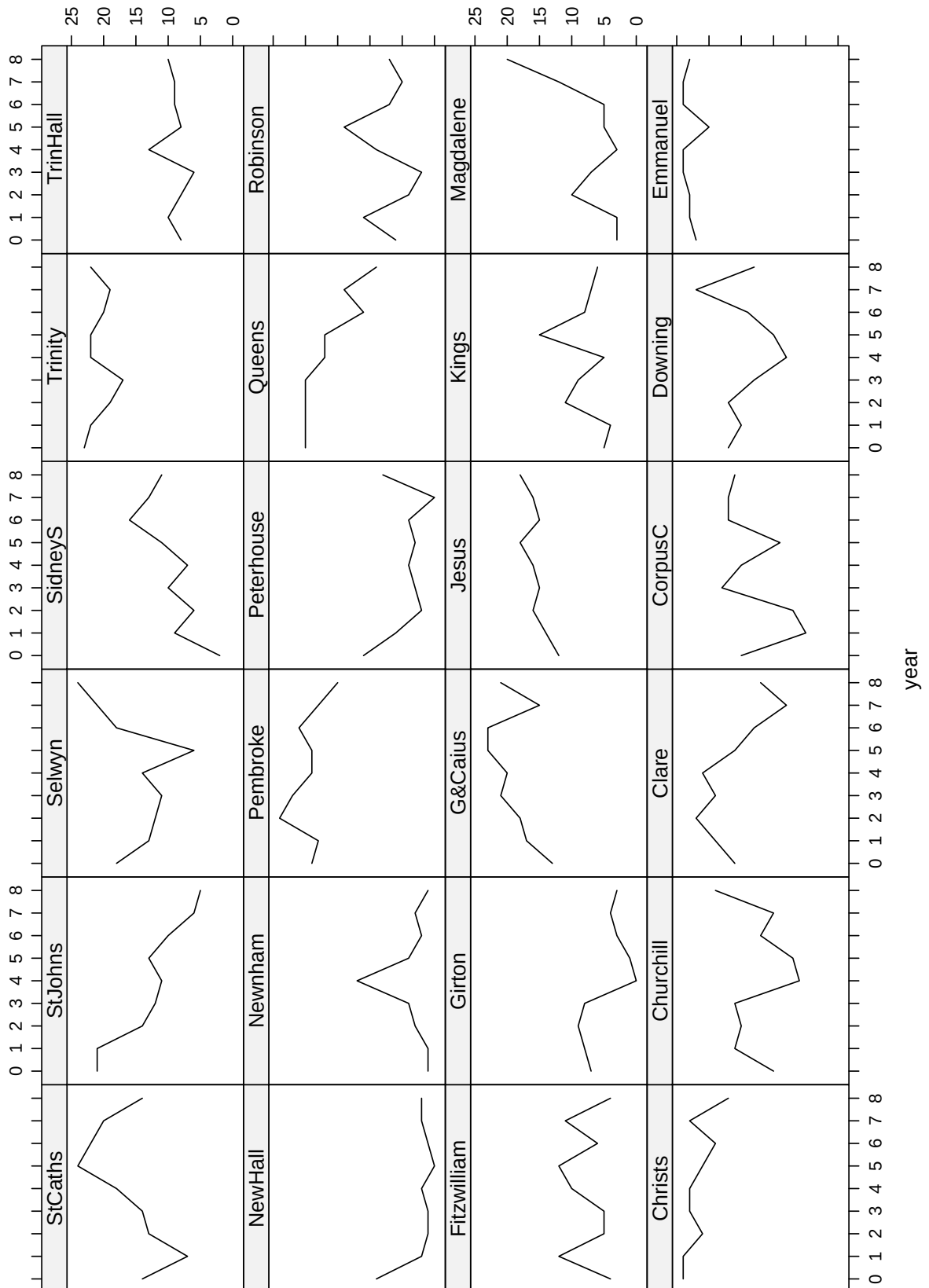
resulting in Figure 1.2.

```
library(lattice)
year <-  gl(9,24, length=216, labels=c(0:8)) # to reduce clutter on plot
xyplot(y~year|College, type="l")
```

Here is a fuller version of the Tompkins Table for 2008, in rank order.

```
 College score  %firsts
1 Selwyn   68.47   29.9
2 Emmanuel 68.30   30.6
3 Trinity  68.27   31.4
4 G&Caius  67.33   27.9
5 Magdalene 65.97  24.5
6 Churchill 65.72  27.1
7 Jesus     65.60  25.2
8 Christs   65.27  25.7
9 CorpusC   65.24  24.1
10 Pembroke 64.96  24.5
11 StCaths  64.63  23.5
12 Downing  64.48  22.8
13 Clare    64.44  22.5
14 SidneyS  64.22  20.9
15 TrinityH 63.76  19.3
16 Queens   63.58  22.3
17 Peterhouse 63.21 22.9
18 Robinson   63.20 20.6
19 Kings      63.07 22.5
20 StJohns    62.48 20.5
21 Fitzwilliam 61.08 18.2
22 Girton     60.84 15.3
23 NewHall    60.03 13.9
24 Newnham    59.96 13.3
25 Homerton   58.62 13.0
26 HughesHall 56.36 20.8
27 Wolfson    55.15  7.4
28 LucyC      52.61  8.7
29 StEdmunds  51.56 11.2
```

So you see that the two columns are correlated, but not perfectly correlated.
You may well want to know exactly how the Tompkins score is computed. The table allocates 5, 3, 2, 1, 0 points respectively for each of a First, a 2-1, a 2-2, a 3rd and 'granted an allowance'. (I think that complete failures are not counted at all.) Then to quote Wikipedia
"The scores in each subject are then weighted to a common average, to avoid the bias towards colleges with higher proportions of students entered for subjects which receive higher grades. The result is then expressed as a percentage of the total number of points available."
(Hence for construction of the Tompkins table, a First in Mathematics (for example) for college x


StCaths
StJohns
Selwyn
SidneyS
Trinity
TrinHall
NewHall
Newnham
Pembroke
Peterhouse
Queens
Robinson
Fitzwilliam
Girton
G&Caius
Jesus
Kings
Magdalene
Christs
Churchill
Clare
CorpusC
Downing
Emmanuel
0 5 10 15 20 25
0 1 2 3 4 5 6 7 8
year

'counts' less than a First in English for college x.)

Note added July 2009: Dr R.J.Gibbens has compiled a more complete set of data, including the results from 2009. Please see

`http://www.statslab.cam.ac.uk/~pat/RGdata.csv`

for a comma-separated file, from 1988-2009 (with some gaps).

The Independent, 5 August 2008, presents the following cricketing data, under the headline 'For better or Worse: England's captains' performances since Tony Greig. (This item follows the resignation of Michael Vaughan as England's Captain.) We can compare the batting average of an individual player when he was Captain with his batting average when he was not Captain.
With the help of my colleague Dr Richard Samworth (using cricinfo) I have corrected the figures given in the Independent for **Stewart**.

```
          MC AvC   MnC AvnC
Greig     14 38.04 44 41.32
Brearley  31 22.48  8 24.28
Botham    12 13.14 90 36.74
Willis    18 21.59 72 26.31
Gower     32 43.59 85 45.50
Gatting   23 44.05 56 32.21
Gooch     34 58.72 84 35.93
Atherton  54 38.73 61 35.25
Stewart   15 39.22 118 39.59
Hussain   45 36.04 51 38.10
Vaughan   37 36.02 45 50.98
Flintoff  11 33.23 58 32.32

key: MC= number of matches as Captain,
   AvC = batting average  as Captain
   MnC = number of matches not as Captain,
  AvnC = batting average not as Captain.
```

Here is my suggestion for plotting the data. You may also like to think of some suitable non-parametric tests: eg is the batting average of a Captain smaller than his batting average when not a Captain? Can you do anything useful with the information on the numbers of matches played?

```
 Cricket <- read.table("Cricket.data", header=T)
attach(Cricket)
captains <- row.names(Cricket)
plot(AvC ~ AvnC, type="n", xlab = "average not as Captain",
  ylab = "average as Captain", xlim=c(10,60), ylim=c(10,60))
text(AvC ~ AvnC, labels=captains)
```

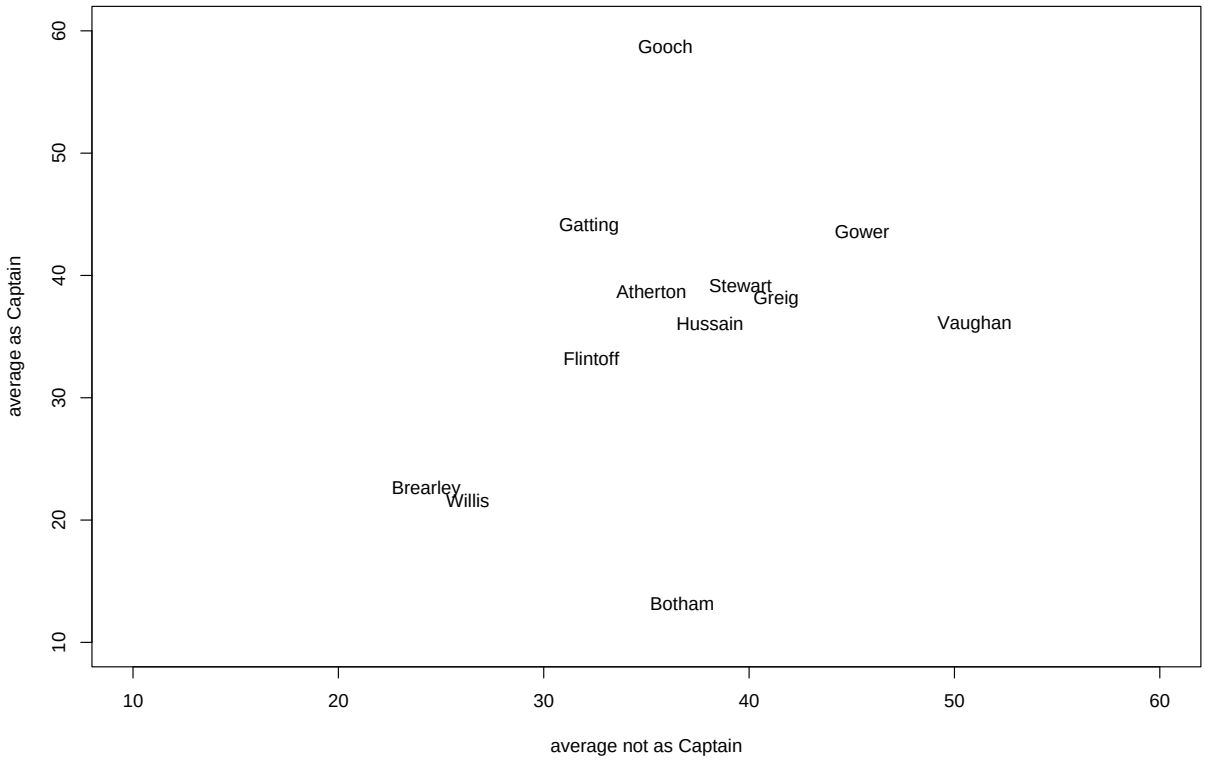


Figure 1.3: Comparing the batting averages of England Captains

This results in Figure 1.3 as shown.

New for June 2009: Cochran's Q statistic.
Here is a brief note on how to compute Cochran's Q for a table of binary data. Consider the set of data given below (taken from Siegel and Castellan 1988, *'Non-parametric statistics for the behavioral sciences'*). We have 3 different 'treatments', which here correspond to 3 different interviews. The yes/no responses (1 or 0) form the columns of this Table, and each row gives the responses of a particular subject: there are 18 subjects in all. Let us assume that y_{it} is the response of the ith subject to the tth treatment, where $i = 1, \ldots, n$, and $t = 1, \ldots, T$. For our example $n = 18,\ T = 3$. We follow the notation of Agresti 2002, *Categorical Data Analysis*, in particular Problem 11.22 on p488. Let us assume that the responses y_{it} are independent variables, with $Pr(y_{it} = 1) = p_{it} = 1 - Pr(y_{it} = 0)$. Assume further a linear logistic model, thus

$$\log((p_{it}/(1 - p_{it})) = \lambda_i + \mu_t.$$

Here the parameters of interest are μ_t. These correspond to the 'treatment' effects, and in particular we wish to test the null hypothesis

$$H_0 : \mu_t = 0 \text{ for all } t.$$

The parameters (λ_i) are the unknown subject effects, and in classical parlance these are 'nuisance' parameters.
Note that the distribution of (y_{it}) conditional on the statistics $(\sum_t y_{it}), 1 \leq i \leq n$ is free of the nuisance parameters (λ_i). Using this conditional distribution, Agresti explains why an appropriate statistic for testing H_0 is Cochran's Q statistic, defined by

$$Q = \frac{n^2(T-1)\sum_t(y_{.t} - y_{..})^2}{T\sum_i y_{i.}(1 - y_{i.})}$$

where

$$y_{.t} = \sum_i y_{it}/n,\ y_{i.} = \sum_t y_{it}/T,\ \text{and } y_{..} = \sum_i \sum_t y_{it}/nT.$$

In other words, to compute Q we just need the row means, the column means, and the overall mean of (y_{it}). The statistic Q is derived from the distribution of $(y_{.t})$ conditional on the subject means $(y_{i.})$, and on the null hypothesis H_0, Q is approximately χ^2 with $(T-1)$ degrees of freedom. First, here is the dataset.

```
sub r1  r2  r3
s1   0  0  0
s2   1  1  0
s3   0  1  0
s4   0  0  0
s5   1  0  0
s6   1  1  0
s7   1  1  0
s8   0  1  0
s9   1  0  0
s10  0  0  0
```

```
s11  1  1  1
s12  1  1  1
s13  1  1  0
s14  1  1  0
s15  1  1  0
s16  1  1  1
s17  1  1  0
s18  1  1  0
```

If you think about the construction of our test statistic, you will realise that any row of the datamatrix where all the responses are the same (for example the first row in our dataset) will make no contribution to Q.
And now, here is the program. You could do things in a more sophisticated way if you wish. I find our old friend 'tapply' very handy.

```
> Cochranq.data = read.table("Cochranq.data", header=T)
> y  = Cochranq.data[,2:4]
>Y = c(y$r1, y$r2, y$r3)
>Y
 [1] 0 1 0 0 1 1 1 0 1 0 1 1 1 1 1 1 1 0 1 1 0 0 1 1 1 0 0 1 1 1 1 1 1 1 1 0 0
[39] 0 0 0 0 0 0 0 0 1 1 0 0 0 1 0 0
>Treatment = gl(3,18, length= 54)
>Subject = gl(18, 1, length= 54)
> Tmean = tapply(Y, Treatment, mean)
> Tmean
        1         2         3
0.7222222 0.7222222 0.1666667
> Submean = tapply(Y,Subject, mean)
> Submean # as a check
        1         2         3         4         5         6         7         8
0.0000000 0.6666667 0.3333333 0.0000000 0.3333333 0.6666667 0.6666667 0.3333333
        9        10        11        12        13        14        15        16
0.3333333 0.0000000 1.0000000 1.0000000 0.6666667 0.6666667 0.6666667 1.0000000
       17        18
0.6666667 0.6666667
> n = 18; T = 3
> Ymean = mean(Y) ; Ymean
[1] 0.537037
> den = T* sum(Submean*(1-Submean)) ; den
[1] 8
>  num = (n^2)*(T-1) * sum((Tmean- Ymean)^2) ; num
[1] 133.3333
> 133.3333/8
> Q = num/dem ; this is Cochran's Q: refer it to chisq with 2 df
> Q
[1] 16.66666
```

The three Treatments are clearly significantly different.
Afterthought You may well wonder why we don't just use the linear logistic structure and try

```
summary(glm(Y ~ Subject + Treatment, binomial)) # and now check the increase in deviance with
summary(glm(Y ~ Subject, binomial))
```

The reason why the correct approach is to condition on the Subject totals is that we need a test-statistic whose distribution is asymptotic in n, the number of subjects. Thus we look at the distribution of the data conditional on the subject totals in order not to have the problem of the number of parameters going to infinity (as it would with the unconditional distribution).

Chapter 2

Getting started in multivariate normal analysis: simulating and plotting

We start by simulating a sample of 200 observations from a given 3-dimensional normal. (You could do this via the function rmvnorm() if you prefer.)

```
i <- 1:200
x <- rnorm(i)   # to generate 200 NID(0,1) rvs
y <- rnorm(i)   # to generate a further set of 200 NID rvs
z <- rnorm(i)   # and again.
v1 <- 2*x + y + 29
summary(v1)
v2 <- x+z+10
v3 <- 3*x + 72  # now (v1,v2,v3) form a r.s. of 200 observations
 # from a specified 3-variate normal distribution.
a <- cbind(v1,v2,v3) # a is the corresponding data-matrix
plot(v1,v2)
pairs(a)
brush(a)       # can you see what's going on ?
options(digits =4)   # makes things easier on the eye
b.cov <- var(a) # the sample covariance matrix
b.cov          # how close is it to the true covariance matrix?
b.cor <- cor(a) # the sample correlation matrix
b.cor
b.lm <- lm(v1~ x+y)
summary(b.lm)
hist(v1)       # does this look as you would expect ?
e.cov <- eigen(b.cov) ; names(e.cov) # eigen-values etc
e.cor <- eigen(b.cor)
e.cov ; e.cor  # why are these 2 sets of e-vals different ?
   #(Now we do DIY calculation of sample covariance matrix)
```

```
col.means <- apply(a,2,mean)
col.means ; help(apply)
col.resid <- sweep(a,2,col.means) ; help(sweep)
# "apply" & "sweep" are not terms you'ld ever have thought of !
cov.diy <- t(col.resid) %*% col.resid  # t( ) is transpose
# You can probably find a more elegant way of computing cov.diy.
cov.diy <- cov.diy/199                 # %*% is matrix mult'n
cov.diy ;b.cov                        # for comparison
 # Here's another useful function.
b <- scale(a,center =T,scale =T)         # NB U.S. spelling
pairs(b)
var(a) ; var(b)
cor(a) ; cor(b) # Now try a Hotelling T-Test.
```

Now we set up a function to compute the bivariate normal density function, for correlation coefficient ρ, calculate this density at each point in a 20×20 grid, and demonstrate three ways of plotting this density.
We compute

$$f(x, y) = \exp -(x^2 - 2\rho xy + y^2)/2(1 - \rho^2).$$

```
x <- seq(-2,2, length= 40); y <- x; rho <- .7
bivnd <- function(x,y){
exp(-(x^2 - 2*rho *x *y +y^2)/(2*(1- rho^2)))
}
z <- x %*% t(y)   # to set up z as a matrix of the right size
for (i in 1:40){
 for (j in 1:40){
  z[i,j] <- bivnd(x[i],y[j])
 }
}
contour(x,y,z)
image(x,y,z)
persp(x,y,z)
```

The resulting three plots are given respectively as Figures 2.1, 2.2 and 2.3.

Repeat, experimenting with different values of ρ. Think about the problem of simulating a sample of size n from this distribution, and then checking its empirical density.
In fact nested loops, while possible in SPlus, are to be avoided if possible (see Venables and Ripley's book). A little thought about matrix algebra shows us that they do not need to be used here. Try the following.

```
x2 <- x^2 ; y2 <- y^2
one <- rep(1, times= 40) # this is the unit vector, of length 40.
z0 <- (x2 %*% t(one) - 2 * rho * x %*% t(y) + one %*% t(y2))/(2*(1- rho^2))
z0 # to check that z0 is a matrix
z <- exp(-z0)
contour(x,y,z)  # and so on....
```

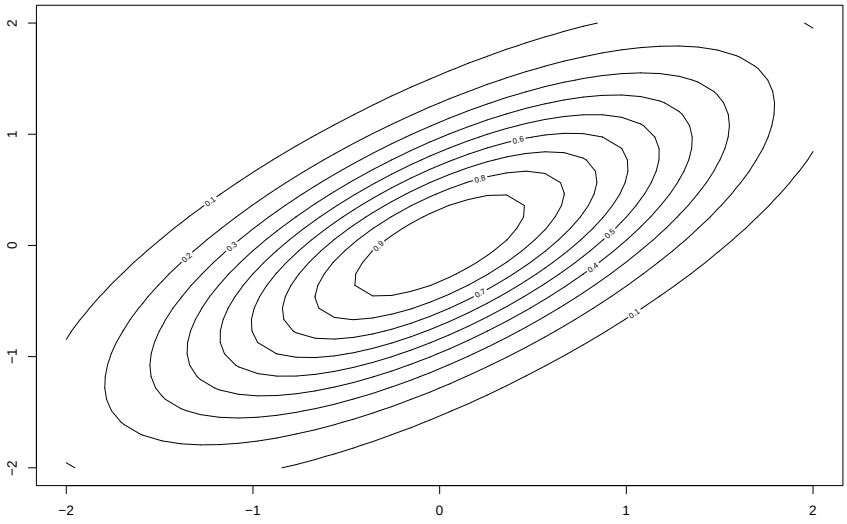


Figure 2.1: The bivariate normal density with $\rho = 0.7$, a contour plot

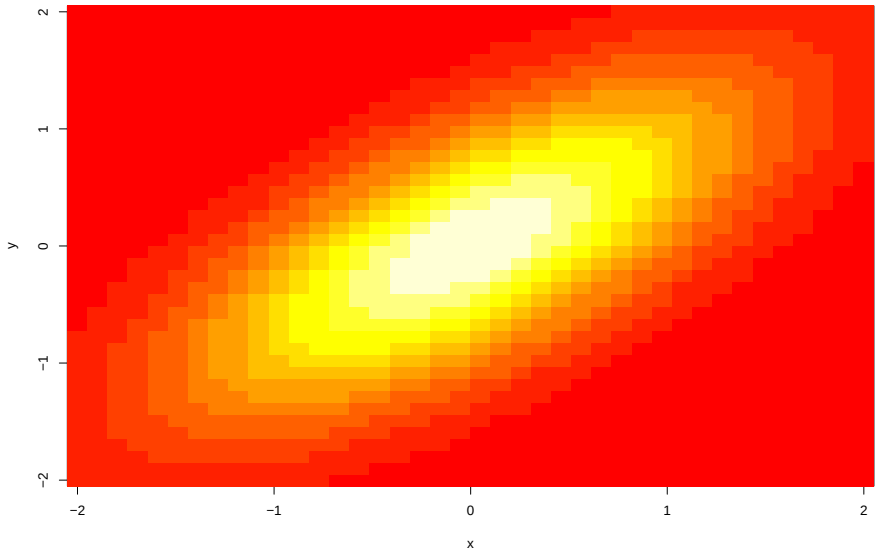


Figure 2.2: The bivariate normal density with $\rho = 0.7$, an image plot

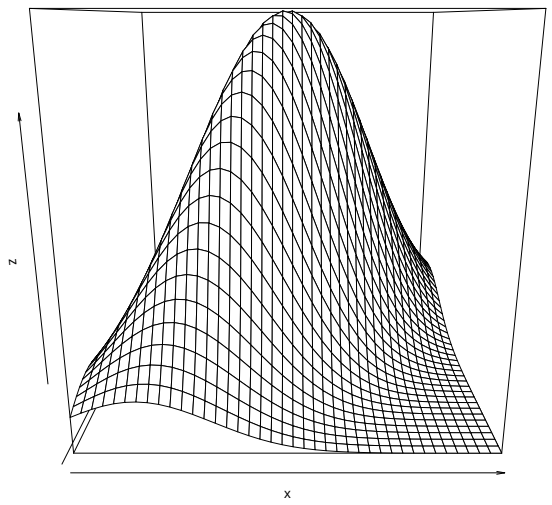


Figure 2.3: The bivariate normal density with $\rho = 0.7$, a perspective plot

Finally, as an optional extra, we plot an ellipse, as shown in Figure 2.4, to show the shape of a contour of the bivariate normal density function.
The ellipse will be centred at (x0, y0). What are a, b and alpha?

```
a <- 3 ; b <- 4; alpha <- pi/3; x0 <- 1 ; y0 <- 2
theta <- seq(0, 2*pi, length=1000)
x <- x0 + a*cos(theta)* cos(alpha) - b*sin(theta)*sin(alpha)
y <- y0 + a*cos(theta)* sin(alpha) + b*sin(theta)*cos(alpha)
plot(x,y, type="l")
points(x0, y0, pch=20) # to show the centre of the ellipse
```

Why does this correspond to a pdf with NEGATIVE correlation?
I got the code from a reply to Rhelp in October 2006 by Alberto Monteiro. If you eliminate θ from the expressions for x, y you should be able to write the equation of the above ellipse as in the usual form, and hence find the correlation coefficient ρ in terms of a, b, α.
Advanced Exercise Now you see how to draw an ellipse, choose six points at random on your ellipse, and draw seven lines to illustrate **Pascal's theorem**. (not a lot to do with statistics, I grant you, but good for your general education! I learnt Pascal's theorem when I was at school, and have somehow never forgotten it.)

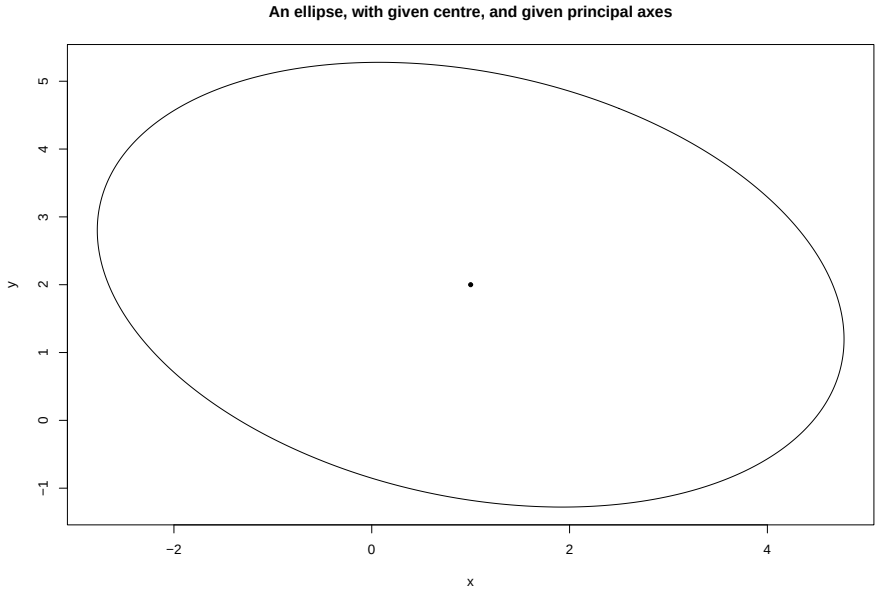


Figure 2.4: An ellipse

Chapter 3

Graphical models for normal distributions: a simulation, and the Times Online Good University Guide data

Note added March 2009: the R package ggm, graphical Gaussian models, by G.M.Marchetti provides very fully for graphical modelling of continuous multivariate data.
First we generate data from a multivariate normal distribution, with x_1, x_2, x_3 mutually independent, conditional on x_4. So the corresponding graphical model for dependencies is as shown in Figure 3.1.

We show that a more conventional multivariate analysis, ie principal components, will not pick up this structure.
(Would a factor analysis detect the structure?)

```
i <- 1:100 ; x4 <- rnorm(i) # first generate x4
x1 <- 7*x4 + (.5)*rnorm(i);x2 <- 8*x4+(.7)*rnorm(i)
x3 <- -10*x4 + rnorm(i)
library(ggm)
dag <- DAG(x1~x4, x2~x4, x3 ~x4) # for the corresponding directed acyclic graph
z = drawGraph(dag)
```

This gives the onscreen plot of the graphical dependence, for you to adjust by clicking near to each point in turn. When you have adjusted the position of the 4 points to your satisfaction click on middle button to quit. The final positions of the 4 points will be in z, as you can check by

```
points(z, pch=19)
a <- cbind(x1,x2,x3,x4) # This is our data matrix
pairs(a)                # for pairwise associations
v <- var(a)             # sample covariance matrix
```

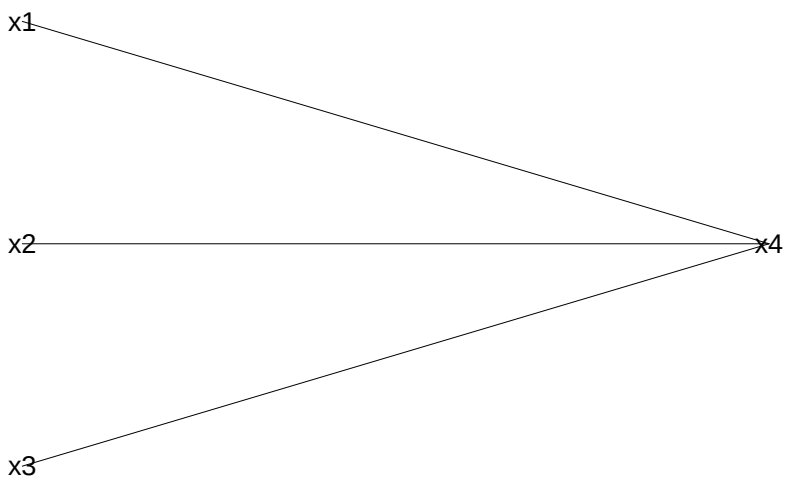


Figure 3.1: x_1, x_2, x_3 mutually independent, conditional on the variable x_4.

```
inv <- solve(v)          # inverse sample cov.matrix
round(v,2) ; round(inv,2) # to have a look at them.
```

Note:

$$var(x_4|\text{all remaining variables}) = 1/inv_{44}$$

where inv is the inverse of the covariance matrix of the $x's$.
Hence for example, we see that $var(x_4|x_1, x_2, x_3)$ is SMALL compared with $var(x_4)$, ie, x_4 is closely determined by x_1, x_2, x_3. Check this by

```
l.m <- lm(x4~x1+x2+x3) ; summary(l.m)
```

Note: standard theory also shows that, for example,

$$corr(x_1, x_2|\text{all remaining variables}) = -inv_{12}/\sqrt{inv_{11}inv_{22}}.$$

Inspection of inv also shows us for example, that $corr(x_1, x_2|$remaining variables) is LOW, and $corr(x_1, x_4|$remaining variables) is HIGH.
A more cunning way to find these conditional correlations is to use the linear model 'trick'.

```
y <- 1:100               #Invent a y-variable
trick.lm <- lm(y~x1+x2+x3+x4)
summary(trick.lm, cor= T)
```

This has given us MINUS the matrix of conditional correlations. (Ignore the column corresponding to 'intercept'.)
Now let's try principal components.

```
a.pr <- princomp(a)       #for principal components
first <- a.pr$x[,1];second <- a.pr$x[,2] # first 2 princ. comps
b <- cbind(a,first,second); pairs(b)
```

This plot will show us how x1,... are related to first and second principal components.

```
round(cor(b),2)        # x4 has no special role.
# We can use the D-matrix to compute appropriate test statistics:
d <- diag(inv) ; d<- 1/d ; d<- sqrt(d) #gives vector of 1/sqrt(d(i,i))
dd <- matrix(d)                     #turns it into 4X1 matrix
t.d <- dd %*% t(dd) #gives matrix of (1/sqrt(d(i,i)d(j,j)))
corr <- inv*t.d     # note, element by element multiplication
chi.sq <- 1 - corr^2
chi.sq <- -100*log(chi.sq)
chi.sq             # refer each term to chi-sq(1) to test sig.
                   # See Whittaker, p175
```

You could experiment with a 'heat map' picture of the correlation (or conditional correlation) matrix. You do have to remember that whereas a graph is indexed from the bottom left-hand corner, with a matrix we count from the top left-hand corner.

```
z = cor(a) # for the 4 by 4 correlation matrix
i = 1:4 ; j = 4:1
zz = z[i,j] # necessary 'flip'
ii= 1:4; jj = 1: 4 ; image(ii,jj,zz)
```

New for 2008: the Times Online Good University Guide data from April 28, 2008.
Firstly, here is the dataset for 2008, omitting the institution names. The column headings are R= Rank (there are some ties), StudSat = Student Satisfaction (with some NA's), ResQual= Research Quality,
ServSpend= Services and Facilities spend, Entry= Entry Standards, Compl= Completion rate, GoodH = percentage getting a 'Good Honours' degree
GradProsp = Graduate Prospects, Total= Total score.

	R	StudSat	ResQual	StudStaff	ServSpend	Entry	Compl	GoodH	GradProsp	Total
1	1	NA	6.2	13.0	2671	522	98.6	89.4	78.6	1000
2	2	NA	6.5	12.2	2097	530	97.9	84.5	87.9	995
3	3	3.9	5.8	9.7	2828	453	96.0	72.4	86.0	960
4	4	3.9	6.3	13.2	1416	471	96.9	74.6	83.0	915
5	5	4.1	5.3	15.4	1009	458	94.8	79.7	72.6	841
6	6	3.9	5.5	9.4	1623	429	94.3	72.5	79.8	832
7	7	NA	5.6	17.1	1724	453	96.7	79.3	73.0	813
8	8	3.9	5.2	14.9	1426	440	95.8	81.3	76.0	811
9	9	4.0	5.2	20.3	1250	452	96.4	75.4	75.8	810
10	10	4.0	4.7	12.4	1546	399	93.2	67.2	79.8	777
11	11	3.8	5.2	16.7	1235	443	95.3	76.8	78.9	764
12	12	4.1	4.3	17.6	1217	368	94.0	63.8	69.7	755
13	13	3.8	5.0	14.5	1254	458	92.2	77.8	76.4	742
14	14	4.0	5.4	16.7	1323	402	90.7	71.3	67.8	738
15	15	3.9	3.9	14.6	1701	346	88.4	64.3	76.9	734

16	16	3.9	5.5	15.0	1144	436	95.2	71.0	64.3	733
17	17	4.0	4.7	16.8	1016	387	94.8	74.6	65.6	726
18	18	3.8	5.3	9.5	1603	365	84.4	69.2	73.3	722
19	19	3.8	5.0	15.6	1368	431	96.2	74.5	70.7	721
20	20	4.1	5.0	17.0	1104	364	91.2	65.4	59.2	709
21	21	4.1	4.5	16.2	1149	371	92.9	62.5	64.6	705
22	22	3.9	4.5	15.1	1085	408	92.4	73.3	70.3	699
23	23	3.9	4.4	16.7	1402	390	92.3	67.6	72.9	694
24	24	3.9	5.2	13.9	1093	356	88.2	71.3	61.9	688
25	25	4.0	4.9	16.8	914	358	91.7	69.1	64.5	683
26	26	3.9	4.3	16.6	1200	389	92.4	69.6	70.2	678
27	27	3.9	5.4	15.8	1074	366	92.5	64.2	61.0	670
28	28	3.9	4.5	13.9	1136	377	90.1	66.3	70.1	658
29	29	3.8	5.1	14.6	1323	406	92.3	68.8	65.6	656
30	30	3.9	4.5	18.4	972	387	92.1	71.2	68.0	653
31	31	3.9	4.3	13.9	1130	425	85.5	64.8	68.0	650
32	32	NA	4.0	13.9	1174	447	78.8	64.3	75.8	648
33	33	3.9	4.3	17.2	1164	363	86.6	69.1	73.2	626
34	34	3.9	4.6	15.5	916	370	90.0	64.8	70.8	625
35	35	3.7	5.1	13.6	1136	382	88.3	70.3	61.6	621
36	36	3.9	4.8	14.7	1111	316	83.4	56.8	63.2	611
37	37	NA	3.9	14.5	1027	334	89.9	62.0	66.9	609
38	38	3.9	4.0	16.4	1093	324	87.0	60.5	65.1	608
39	39	4.1	4.0	19.2	976	299	89.8	61.1	56.1	607
40	40	3.7	4.7	15.9	1089	341	86.8	61.6	79.1	603
41	40	3.9	3.7	18.7	748	318	85.0	63.9	80.6	603
42	42	3.8	4.7	12.8	1088	342	88.5	61.6	72.6	599
43	42	4.0	3.2	19.0	761	294	83.9	57.3	71.1	599
44	44	NA	3.6	18.3	998	429	81.8	67.9	71.2	598
45	45	NA	4.2	16.8	1056	375	82.3	57.9	61.0	551
46	46	3.9	4.4	16.9	979	289	88.3	54.1	60.1	531
47	47	4.0	3.9	16.9	1042	290	77.0	50.0	65.8	530
48	48	3.9	3.4	18.6	1105	265	82.2	54.2	75.8	519
49	49	3.9	1.6	17.0	908	295	84.9	56.6	72.1	512
50	50	NA	4.2	17.6	939	382	68.7	58.6	74.6	502
51	51	3.7	3.1	16.7	1188	302	85.4	63.7	63.3	496
52	52	3.8	4.9	20.2	758	297	82.2	58.9	58.4	486
53	53	3.9	2.4	19.2	1345	279	77.0	61.8	59.2	485
54	54	3.8	4.0	18.1	693	314	84.3	56.0	64.1	484
55	55	NA	0.9	18.5	927	311	79.8	53.4	81.8	480
56	56	3.7	1.4	19.1	810	283	89.1	53.3	77.1	456
57	57	4.0	4.4	26.2	554	258	76.9	60.5	67.7	453
58	58	NA	1.5	21.3	710	333	77.7	60.3	75.2	447
59	59	3.5	4.4	18.7	873	391	87.1	57.9	51.3	442
60	60	3.9	1.6	17.5	1017	275	81.2	55.9	55.5	437
61	61	3.8	1.7	18.1	805	280	83.0	55.3	64.2	425
62	62	NA	1.2	18.1	767	354	74.8	60.0	61.6	419

63	62	3.7	0.7	18.5	851	278	85.2	59.8	63.7	419
64	64	3.9	1.1	18.1	943	241	79.8	52.2	53.3	399
65	65	3.9	1.3	16.3	1063	273	69.3	52.1	53.4	397
66	66	3.8	0.8	16.2	1024	252	77.3	57.7	62.5	396
67	67	NA	0.8	17.8	960	292	73.3	57.4	61.8	393
68	68	3.9	1.7	21.0	652	242	88.0	45.1	59.2	391
69	69	3.9	1.5	20.6	746	265	84.5	54.7	49.4	385
70	70	3.8	0.9	17.4	928	246	76.9	48.1	66.3	383
71	71	3.8	1.9	20.3	1257	237	79.5	52.7	55.6	382
72	72	3.8	1.6	19.8	867	266	80.6	56.7	59.8	380
73	73	3.8	1.1	21.9	1031	291	80.7	52.4	59.3	375
74	74	3.8	1.7	20.2	1000	242	80.5	51.3	53.7	371
75	75	3.8	0.8	20.7	1077	248	85.1	48.4	54.5	366
76	75	3.9	1.0	20.6	929	239	73.7	56.0	61.1	366
77	75	3.9	1.1	20.4	583	252	83.5	45.8	62.4	366
78	78	3.9	0.5	19.1	826	228	74.9	56.5	56.7	365
79	79	3.8	1.7	19.0	758	256	81.4	48.0	59.3	363
80	80	3.8	1.2	18.3	882	228	78.6	49.2	64.0	362
81	81	3.8	1.7	16.7	710	238	77.0	50.4	56.1	360
82	82	3.9	1.0	19.4	692	221	80.0	56.8	55.4	359
83	83	3.7	2.1	16.7	857	269	73.2	53.5	59.7	356
84	84	NA	0.6	23.4	1808	207	75.4	41.1	59.6	348
85	85	3.8	1.2	18.5	840	252	77.1	48.4	57.1	335
86	85	3.8	1.4	18.5	860	235	75.9	45.1	61.9	335
87	87	3.7	1.1	23.8	911	270	85.0	55.4	61.2	332
88	88	3.7	1.5	24.4	502	264	85.7	67.7	54.8	328
89	89	3.8	0.8	20.2	828	221	80.8	45.4	63.1	323
90	90	3.8	1.2	18.9	822	261	75.2	47.8	53.7	319
91	91	3.7	1.2	17.5	739	229	80.5	56.4	53.5	318
92	92	3.9	0.9	20.6	610	244	78.9	46.0	55.5	317
93	93	3.9	0.8	17.7	785	179	63.3	51.7	64.7	316
94	94	NA	0.7	20.0	1085	313	68.7	43.8	55.5	311
95	95	3.9	0.7	20.7	705	236	74.4	44.6	63.0	310
96	96	3.7	0.8	23.2	928	263	82.7	49.0	59.5	305
97	97	3.7	1.4	21.5	753	246	80.6	47.7	63.8	303
98	98	3.8	0.9	22.4	978	215	77.0	49.8	57.1	298
99	99	NA	1.4	22.3	1298	190	68.5	40.0	58.0	284
100	100	3.8	0.5	17.8	672	253	76.1	46.1	48.4	280
101	100	NA	0.7	19.8	1478	214	64.0	47.2	51.9	280
102	102	3.7	0.4	25.6	780	266	85.4	49.9	61.7	276
103	103	NA	1.3	23.1	800	191	72.0	51.7	61.9	270
104	104	3.8	0.5	19.9	615	242	64.8	52.7	61.5	262
105	105	3.7	0.5	21.6	940	224	78.2	43.9	48.8	247
106	106	3.9	0.4	22.8	248	243	78.3	38.1	55.7	242
107	107	3.6	0.6	22.8	626	257	76.6	46.5	70.7	229
108	108	3.7	1.3	29.6	1576	189	71.9	46.5	55.7	219
109	109	3.6	0.7	26.3	793	255	83.9	52.3	48.9	214

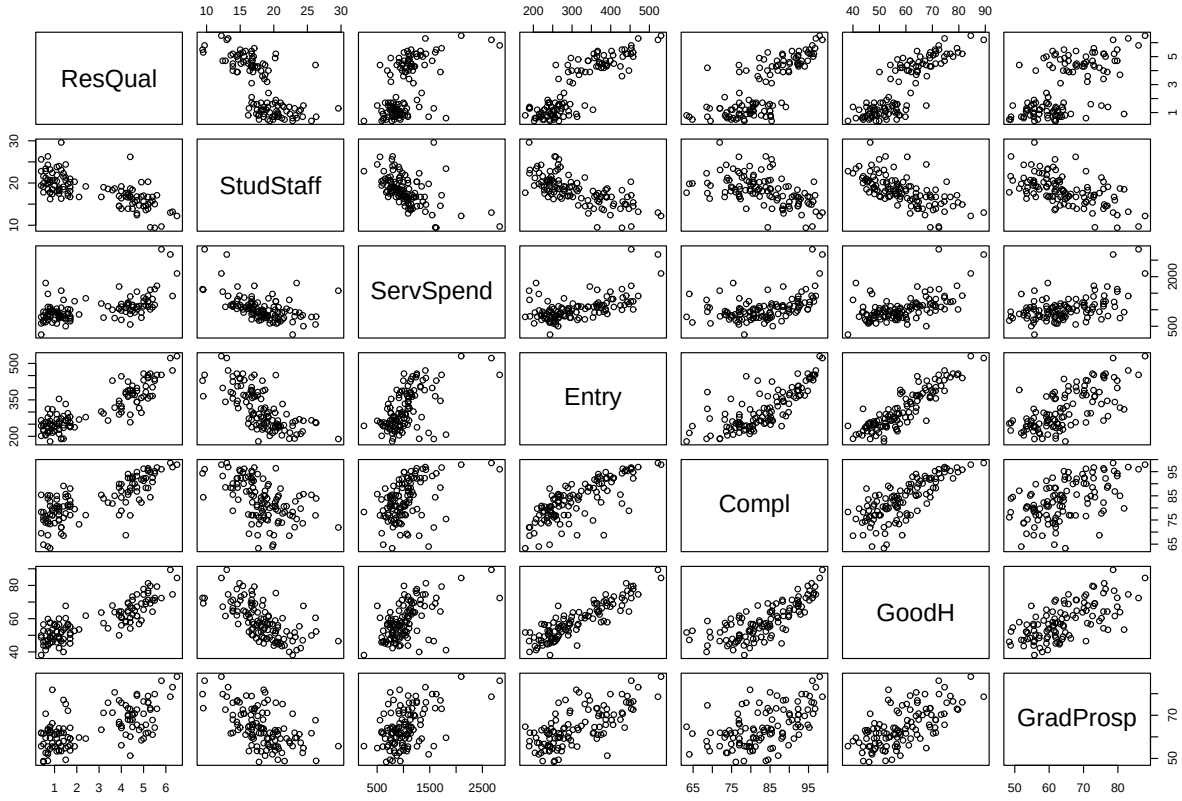


Figure 3.2: The pairs plot for the Times Online Good University Guide data, 2008

```
110 109     3.7      1.2      24.0       791   210  79.7  42.3      57.3   214
111 111     3.6      0.4      19.4       586   202  69.5  48.7      61.9   209
112 112     3.7      0.6      21.2       917   204  73.9  43.5      52.6   207
113 113     3.7      0.7      24.3       833   220  73.6  47.0      52.6   191
```

I used only the variables

```
ResQual, ..., GradProsp
```

for which the corresponding pairs plot is Figure 3.2. Finally, here is the matrix of conditional correlations, derived as above. You will see that the only two variables that have a strong correlation, conditional on the remaining five variables, are 'GoodHon' and 'Entry'.

```
           ResQual StudStaff ServSpend Entry Compl GoodH GradProsp
ResQual       1.00     -0.24      0.06  0.31  0.28  0.21      0.00
StudStaff    -0.24      1.00     -0.15 -0.17  0.14 -0.02     -0.12
ServSpend     0.06     -0.15      1.00  0.08 -0.03  0.07      0.12
Entry         0.31     -0.17      0.08  1.00  0.18  0.48      0.24
Compl         0.28      0.14     -0.03  0.18  1.00  0.21     -0.02
```

```
GoodH        0.21     -0.02      0.07  0.48  0.21  1.00      0.07
GradProsp    0.00     -0.12      0.12  0.24 -0.02  0.07      1.00
```

Afterthought: the above analysis was perhaps a bit simple-minded, since

```
hist(ResQual)
```

shows that this variable has a clearly bi-modal distribution. You could try using only the first 60 rows of the data-matrix in your analysis.
In case you feel you really HAVE to know the rank order of the 113 Universities concerned, here it is (I abbreviated some of the names). I see that Cambridge is **second**. Huh!

```
  [1] Oxford                    Cambridge
  [3] Imperial_College          London_School_of_Economics
  [5] St_Andrews                University_College_London
  [7] Warwick                   Bristol
  [9] Durham                    Kings_College_London
 [11] Bath                      Loughborough
 [13] Edinburgh                 Southampton
 [15] Aston                     York
 [17] Exeter                    S_O_A_S
 [19] Nottingham                East_Anglia
 [21] Leicester                 Sheffield
 [23] Newcastle                 Royal_Holloway
 [25] Reading                   Birmingham
 [27] Lancaster                 Cardiff
 [29] Manchester                Leeds
 [31] Glasgow                   Aberdeen
 [33] Queens_Belfast            Liverpool
 [35] Sussex                    Essex
 [37] Stirling                  Kent
 [39] Aberystwyth               Surrey
 [41] City                      Queen_Mary_London
 [43] Hull                      Strathclyde
 [45] Heriot-Watt               Swansea
 [47] Bangor                    Bradford
 [49] Oxford_Brookes            Dundee
 [51] Brunel                    Goldsmiths_London
 [53] Ulster                    Keele
 [55] Robert_Gordon             Nottingham_Trent
 [57] Lampeter                  Queen_Margaret_Edinburgh
 [59] Univ_of_the_Arts,London   Plymouth
 [61] Brighton                  Glasgow_Caledonian
 [63] Bournemouth               Staffordshire
 [65] Glamorgan                 UCE_Birmingham
 [67] Napier                    Chichester
 [69] Winchester                Central_Lancashire
 [71] Roehampton                West_of_England
 [73] Northumbria               Gloucestershire
```

```
 [75] UWIC_Cardiff               Coventry
 [77] Canterbury_Christ_Church   Newport
 [79] Portsmouth                 Kingston
 [81] Sunderland                 Northampton
 [83] Salford                    Bedfordshire
 [85] LiverpoolJohnMoores        Hertfordshire
 [87] Sheffield_Hallam           Bath_Spa
 [89] Worcester                  Manchester_Metropolitan
 [91] Westminster                Huddersfield
 [93] Bolton                     Paisley
 [95] Teesside                   Leeds_Metropolitan
 [97] DeMontfort                 Derby
 [99] East_London                Chester
[101] Abertay                    York_St_John
[103] London_South_Bank          Anglia_Ruskin
[105] Southampton_Solent         Edge_Hill
[107] Cumbria                    Middlesex
[109] Lincoln                    Greenwich
[111] Thames_Valley              Wolverhampton
[113] Liverpool_Hope
```

Chapter 4

Manova on 3 groups, using Fisher's classic Iris data

This dataset consists of 50 cases of each of 3 species, namely Iris setosa, Iris virginica, and Iris versicolor. Each case has 4 measurements on the length and width of its petals and sepals.

```
R
data(iris)
ir.species <- gl(3,50, length=150, labels=c("s", "c", "v"))
pairs(ir)  # not so revealing: we need to label the 3 species separately.
plot(ir[,1:2], type="n")
text(ir[,1:2], labels=as.character(ir.species)) # for a simple pairwise plot
# but, for a really good plot, we use the R example, thus
pairs(iris[1:4], main = "Anderson's Iris Data -- 3 species",
+  pch = 21, bg = c("red", "green3", "blue")[unclass(iris$Species)])
```

These pairwise plots result in Figures 4.1 and 4.2 respectively, and show some separation between the three groups.

```
summary(aov(ir[,1] ~ ir.species)) # for a 1-way anova on the 1st vector
iris.manova <- manova(ir~ ir.species) # for the manova
summary(iris.manova, univar=T) # compare with result of aov()
summary(iris.manova, test="wilk") # to look at the whole vector
liris.manova <- manova(log(ir)~ ir.species) #to try log-transform
summary(liris.manova, test="wilk")
```

The iris dataset works almost too well. For a fun dataset, where the separation between the groups is less clearcut, try the painters data (de Piles).

```
library(MASS)
?painters
```

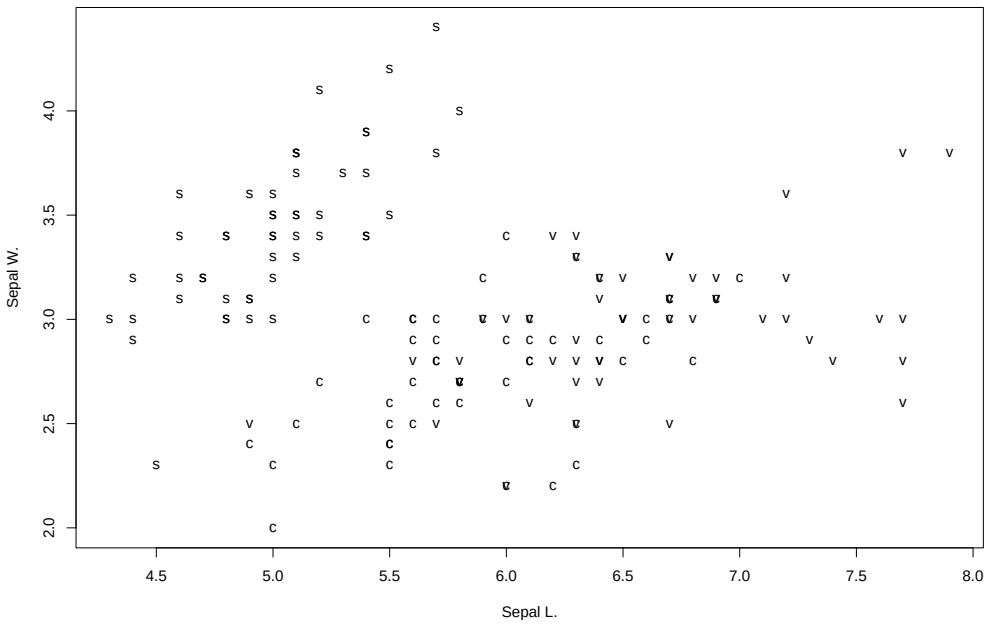


Figure 4.1: Fisher's Iris data: a simple pairs plot

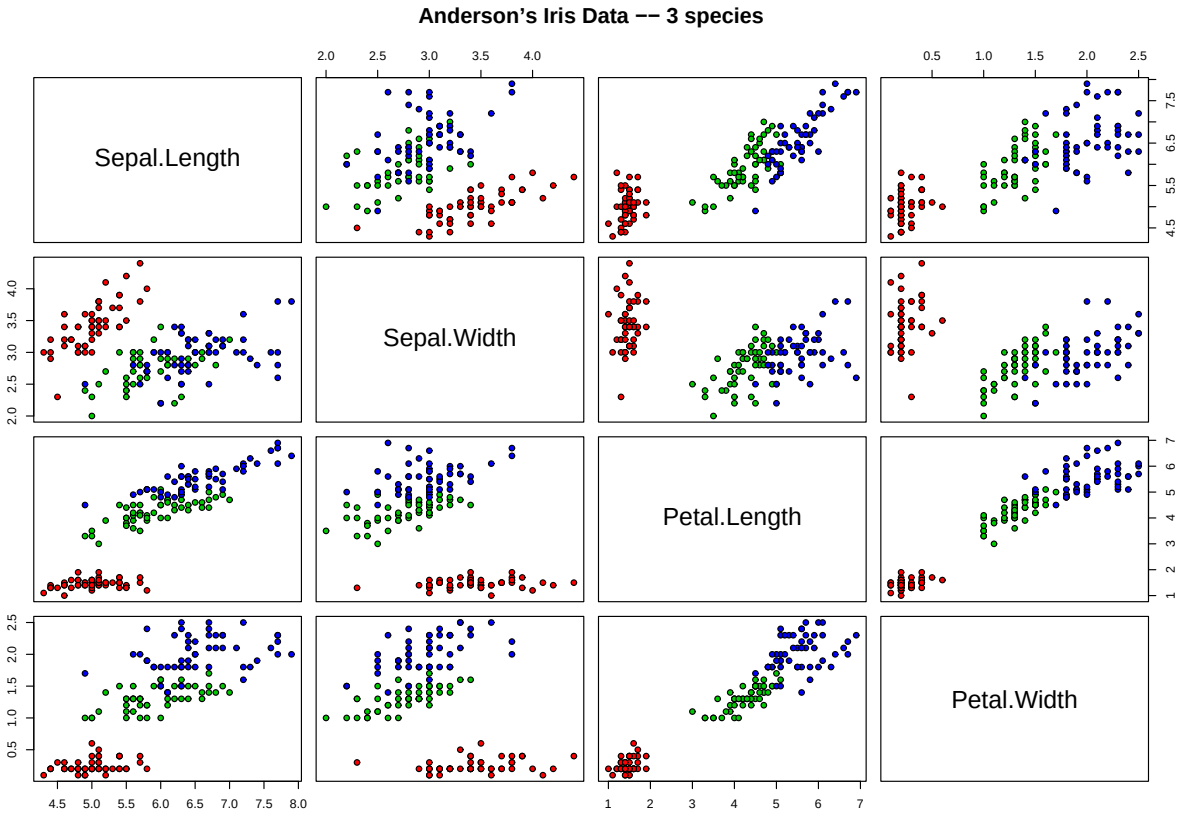


Figure 4.2: Fisher's Iris data: a full pairs plot

```
x <- painters[,1:4] ; x <- as.matrix(x)
 school <- painters[,5]
 is.factor(school) ; table(school)
 painters.manova <- manova(x~ school)
 summary(painters.manova, univar =T)
 summary(painters.manova,test ="wilk")
 for (i in 1:4){
 + cat(round(tapply(x[,i],school,mean),3),"\n")
 + }
 # This shows us the differences between the Schools.
```

Exercise: do a 'pairs'plot of the 'painters' data, with a different plotting symbol for each of the 8 Schools.

Chapter 5

Linear discrimination between groups

Let x be the original data vector, and consider doing a 1-way anova on the scalar quantity $y = u^T x$.
We want to choose a u such that the 1-way anova on y gives maximal separation between the groups:
hence you can see that we aim to solve the problem:
choose u to maximise $u^T B u$ subject to $u^T W u = 1$, where B, W are the between-groups and within-groups sums of (squares and products) matrices, respectively.
This gives us that $Bu = \lambda W u$, and so taking λ as the largest such value gives the maximum value of $u^T B u / u^T W u$.
Hence, for the transformed variable $y = u^T x$, in the 1-way anova, we find that
'between groups ss/(within groups ss)' $= \lambda$, and hence
'between groups ss/ total ss' $= \lambda/(\lambda + 1) = R^2$,
where R^2 is the usual multiple regression R^2: in this case it measures how well the separation into groups explains the overall variation.
In the example below, each of $B,\ W$ is a 4×4 matrix, and since there are just 3 groups, it follows that B is of rank 2, hence the final λ, and hence the final R^2, is effectively zero.

We use the Iris dataset defined above, and compare 2 methods. (The function lda() is also used on this dataset in Venables and Ripley, 4th edition.)

```
a <- log(ir) ; grou <- ir.species # for convenience
teeny.dis  <- discr(a,3 )
teeny.dis
teeny.dv <- a %*% teeny.dis$vars #new coords
teeny.x <- teeny.dv[,1]
tapply(teeny.x,grou,mean)
```

Now relate this to teeny.dis output. I have always had great difficulty interpreting

```
teeny.dis$groups
```

but in fact the ?discr does tell you what to expect. Here goes.
Taking the first column of the 3×3 matrix

```
teeny.dis$groups,
```

we set

```
x1 <- c(rep(0.8074378, times =50), rep(-0.2986802,times =50), rep( -0.5087577,times =50))
cor.test(x1, teeny.x)
```

and sure enough, this reveals to us the correlation 0.9887738, as we get for the first component of

```
teeny.dis$cor
```

above.

```
par(mfrow =c(3,1))                  # 3 plots on 1 page
hist(teeny.x[grou=="s"]);hist(teeny.x[grou =="c"]);hist(teeny.x[grou =="v"])
par(pty ="s")                       # to make graph frames SQUARE
par(mfrow =c(1,1))
teeny.y <- teeny.dv[,2]
plot(teeny.x,teeny.y,type ="n",xlab ="first disc var",ylab ="second disc var")
text(teeny.x,teeny.y, labels = as.character(ir.species))
```

We now compare with data in original co-ordinates

```
v1 <- a[,1] ; v2 <- a[,2]
plot(v1,v2,type ="n") ; text(v1,v2, labels= as.character(ir.species))
library(MASS)
?lda
ir.lda <- lda(log(ir), ir.species)
ir.lda
plot(ir.lda) # we'll do this another way now
ir.ld <- predict(ir.lda, dimension =2)$x
plot(ir.ld, type ="n", xlab = "first lin discr", ylab = "second lin discr"))
text(ir.ld, labels = as.character(ir.species), cex =1.0)
```

Here's how to apply it for the painters' dataset.

```
summary(painters)
table(School)
k <- scan()
10 6 6 10 7 4 7 4

x <- painters[,1:4]
first.dis <- discr(x,k) ; first.dis
```

Chapter 6

Principal Components Analysis

The data below are from Hartigan, 1975, "Clustering Algorithms" Our first object is to see whether the 9 points in 5 dimensions can be represented as 9 points in a plane. Here is Hartigan's data set.

```
       energy protein fat calcium iron
beef     180 22 10 17 3.7
chicken  170 25  7 12 1.5
clams     45  7  1 74 5.4
crabmeat  90 14  2 38 0.8
mackerel 155 16  9 157 1.8
salmon   120 17  5 159 0.7
sardines 180 22  9 367 2.5
tuna     170 25  7   7 1.2
shrimp   110 23  1  98 2.6
```

```
food <- read.table("food",header=T) ; food
attach(food)
a <- data.matrix(food) ; a
a.cov <- var(a) ; a.cov
a.corr <- cor(a) ; a.corr
pairs(a)
help(princomp)
a.pcp <- princomp(a) ; names(a.pcp)
a.pcp        # Can you understand what it's telling you ?
a.pcp$sdev   # What are these ?
help(eigen)  # We find out directly.
x <- eigen(a.cov) ; names(x)
x$values
z <- a.pcp$sdev  ; z <- z*z ; z
```

Do you see the connection? Let's get a plot of the 9 points using first 2 principal components.

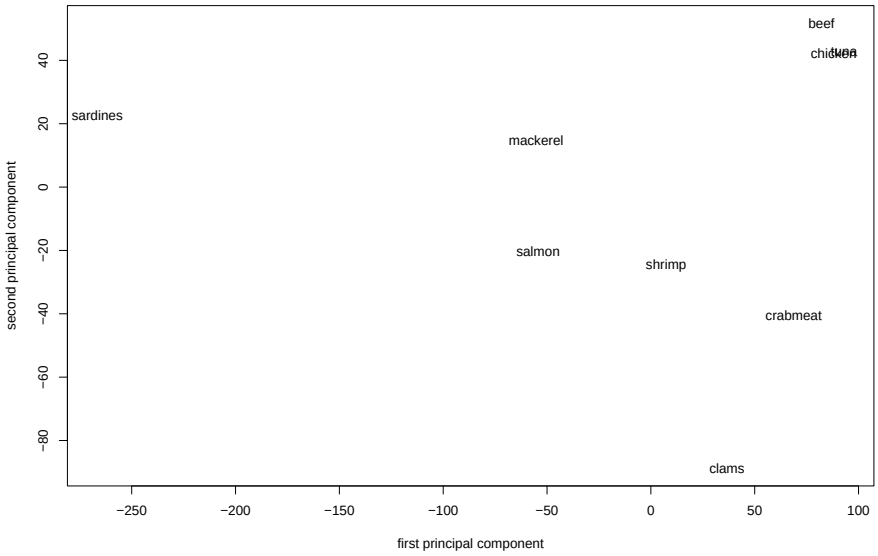


Figure 6.1: Principal components on the unstandardised food data

```
a.pcp            # for a reminder
x1 <- a.pcp$scores[,1] # first column
x2 <- a.pcp$scores[,2] # second column
plot(x1,x2)      # but we really need to label the points
a.lab <-  row.names(food)
plot(x1,x2,type="n",xlab ="first principal component",ylab="second principal component")
text(x1,x2,a.lab)
```

This gives us Figure 6.1. Because of the high variability of calcium relative to the other 5 variables, this variable will dominate the first principal component, as is shown by Figure 6.2, which is obtained by

```
plot(x1,calcium,type="n") ; text(x1,calcium,a.lab)
```

We may prefer to standardise all the original variables to have mean 0, variance 1 before we do the principal components analysis. Thus, in effect, we find the eigen-values of the **correlation** matrix rather than those of the **covariance** matrix. Of course, this gives each of the 5 variables "equal weight" in the analysis. The final plots may look completely different from the plots which result from the unstandardised variables.
A problem for you: compute the standardised data matrix from a above, and do the principal components analysis on this.
Compare the results of the above with what you get from princomp() . You should also try

```
a.pcp <- princomp(a,cor =T)
a.pcp$loadings
Loadings:
```

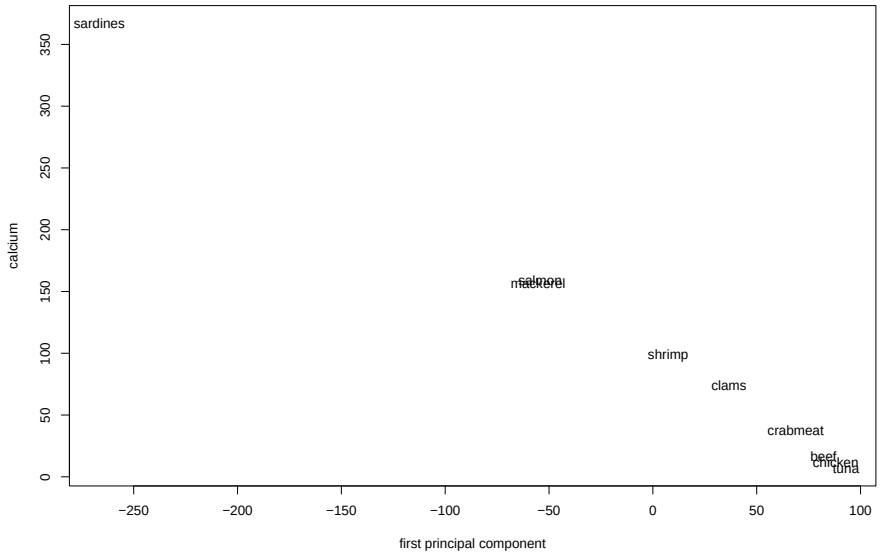


Figure 6.2: Showing that the first principal component is (almost) - calcium

```
        Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
energy  -0.601        -0.173         0.778
protein -0.516  0.277         0.714 -0.374
fat     -0.519 -0.286 -0.286 -0.561 -0.503
calcium -0.123 -0.793  0.533  0.268
iron     0.297 -0.458 -0.773  0.321

               Comp.1 Comp.2 Comp.3 Comp.4 Comp.5
SS loadings       1.0    1.0    1.0    1.0    1.0
Proportion Var    0.2    0.2    0.2    0.2    0.2
Cumulative Var    0.2    0.4    0.6    0.8    1.0
```

Note: Figure 6.3 shows a **biplot** of the data. This is an ingenious 2-dimensional picture of the data (using the standardised variables) is produced by

```
biplot(a.pcp)
```

The directions of the arrows in the biplot correspond to the 'loadings' of components 1 and 2.

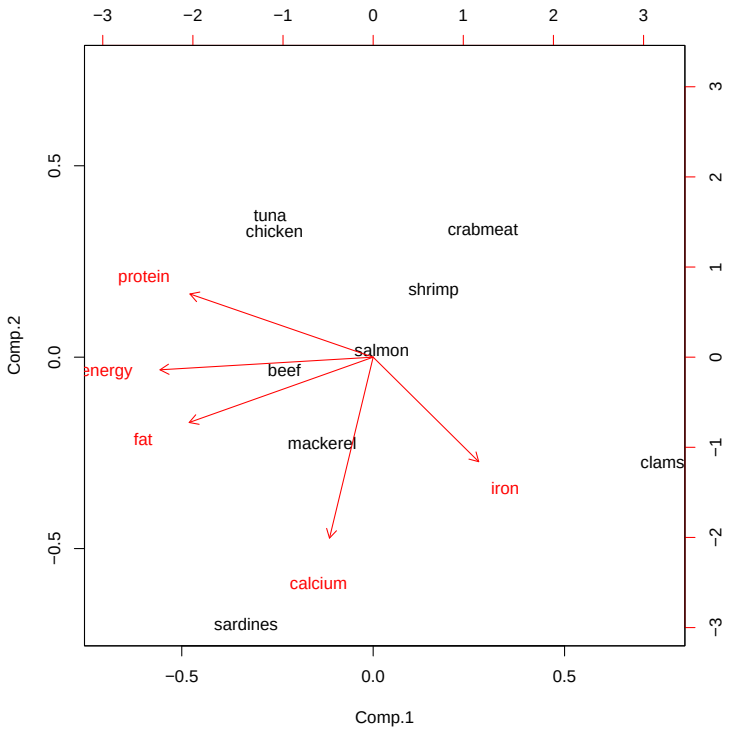


Figure 6.3: The biplot for the standardised food data

Chapter 7

Hierarchical Cluster Analysis

Here is a data set which represents 10 points in 3 dimensions, and I choose this very small data-set so that you can see what to expect from the result of the cluster analysis.

```
0 2 3
4 5 6
70 7 7
10 11 12
3   4  5
6   7 18
19 20 21
22 23 44
25 26 27
28 29 30
```

Here's how to analyse it in R (S-Plus will do the same, but with slightly different terminology). A fundamental problem with hierarchical cluster analysis is that there are several ways of choosing the **distance** function, and having made that particular choice, there are then several ways of choosing the particular method of clustering: this is because we can define the distance between two clusters in several different ways. You have to realise that cluster analysis is a 'data-analytic' method, ie a (sensible) way of reducing a complex dataset, but it does not depend on any fundamental statistical modelling ideas such as likelihood, parameters, goodness of fit etc.

```
a <- read.table("tinycluster") ; a
a <- data.matrix(a) ; a
```

Observe that R can cope with missing values in constructing a distance matrix.

```
d <- dist(a,method ="euclidean")
round(d,2) # which results in the interpoint distances below

       1     2     3     4     5     6     7     8     9
```

```
2   5.83
3  70.29 66.04
4  16.19 10.39 60.34
5   4.12  1.73 67.10 12.12
6  16.91 12.33 64.94  8.25 13.67
7  31.76 25.98 54.46 15.59 27.71 18.63
8  51.05 45.74 62.68 36.22 47.36 34.47 23.39
9  42.15 36.37 52.78 25.98 38.11 28.34 10.39 17.52
10 47.35 41.57 52.70 31.18 43.30 33.35 15.59 16.37  5.20
# You can see that points 2 and 5 are the closest of the 10.
par(mfrow=c(2,1))
h1 <- hclust(d,method ="complete") # this is the default method
names(h1)
plclust(h1)   # does this make sense to you ?
h2 <- hclust(d,method ="single")
plclust(h2)   # Observe differences from previous plot
# Now we'll put labels on the points
teeny.lab <- scan(,"")
a b c d e f g
h i j
             # NB,blank line
par(mfrow=c(1,1))
plclust(h2,labels =teeny.lab)
```

This results in the graph given in Figure 7.1.

Now we'll try an example of some BINARY data.
Here's my file for the Lent 2003 cohort of graduate students.
The questions are
1.do you eat eggs?
2.do you eat meat?
3.do you drink coffee?
4.do you like beer?
5. Are you a UK resident?
6. Are you a Cambridge graduate?
7. Are you female?
8. Do you play sports?
9. Do you have a full driving licence?
10.Are you left-handed?
The students gave the responses Yes or No, as 'y', 'n' respectively.
I admit these questions are BORING, but more interesting, personal questions might not be publicly usable, as these are.

```
.....................................................................
data for Lent 2003
        eggs  meat  coffee  beer  UKres  Cantab  Fem  sports driver Left-h
Vivienne  y    n      y      n      y      n      y     y      y      n
Taeko     y    y      y      n      y      y      y     n      n      n
```

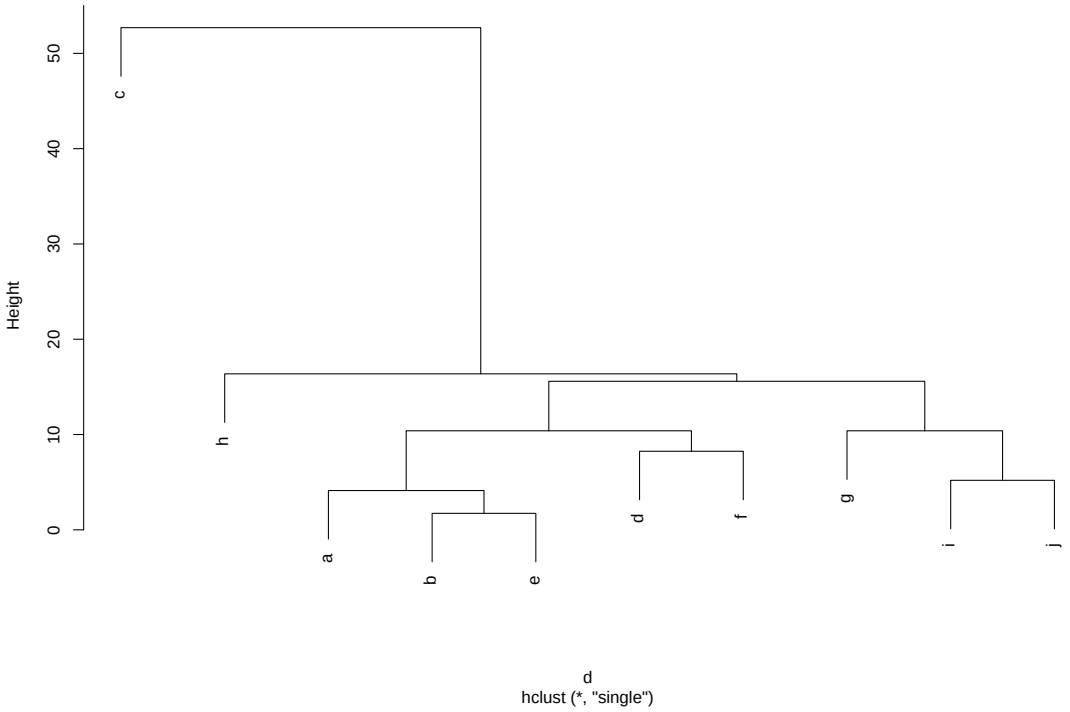


Figure 7.1: A dendrogram for the set of 10 points in 3 dimensions

```
Luitgard  y    n      y      n      n      n      y     y      y      n
Alet      y    y      y      y      n      n      y     n      y      n
Tom       y    y      y      y      y      y      n     y      y      n
LinYee    y    y      y      n      n      n      n     y      y      n
Pio       y    y      y      n      n      n      n     y      n      n
LingChen  y    y      n      n      n      n      y     y      n      n
HuiChin   y    y      y      n      n      n      y     y      y      n
Martin    y    y      y      y      y      n      n     y      y      n
Nicolas   y    y      y      y      n      n      n     y      y      y
Mohammad  y    y      y      n      n      n      n     n      y      n
Meg       y    y      y      n      n      n      y     y      n      n
Cindy     y    y      y      y      n      n      y     y      y      n
Peter     y    y      y      y      n      n      n     y      y      n
Paul      y    y      n      y      y      y      n     y      n      n
```

```
# What follows below was done in S-Plus
a <- read.table("students2003", header=T)
student.lab <- row.names(a)
a ; student.lab
a <- (a=="y")*1   # to convert to 0,1 data
a
d <- dist(a,"binary") ; d # can you understand it ?
s <- 1-d ; s              #s is the  SIMILARITY matrix
h <- hclust("compact",sim =s) # operating on the similarity matrix
plclust(h)               # does this make sense ?
h <- hclust(d,"compact") ; plclust(h) # now on the dissimilarity matrix
# essentially the same as the previous plot ?
# Now for fun with labels.
plclust(h,labels =student.lab)

ls()          # to show you all your S-Plus objects
# use rm() to remove unnecessary clutter
ls()          # shows you what you've done.
```

Exercise: do a cluster analysis on the 16 students using the FIRST 4 questions only.

Chapter 8

Decision trees

You are a trainee astronaut, learning how you should decide whether or not to use your autolander. You have a "training set" of 256 lines of data, telling you whether or not the autolander was used for all combinations of 6 factors(eg visibility yes/no) in the past. Here we show you how to use Splus to grow a "decision tree" to guide your actions in the future.

```
library(MASS)
help(shuttle)
shuttle
attach(shuttle)
summary(shuttle)
table(use,vis)
table(use,vis,error)   # and so on,for some useful summaries.
shuttle.tree <- tree(use~.,shuttle) # this grows a tree
 # making use of all 6 factors,if necessary.
summary(shuttle.tree)
shuttle.tree          # what is this telling you ?
 # Do we make use of "vis" in our decision ?
 # Do we make use of "wind" ?
plot(shuttle.tree)
text(shuttle.tree,srt =90)
```

Now try growing a tree using only the first 4 factors,and compare your results with the first tree obtained. For an interesting comparison with R, look at the function rpart() thus

```
library(rpart)  # rpart means 'recursive partitioning'
tree.rp <- rpart(use ~. , shuttle) ; tree.rp
plot(tree.rp,compress =T)
text(tree.rp,use.n =T)
post.rpart(shuttle.rp)  # for a  nice postscript graph
```

Note added June 2007. Now that I have discovered the new (ie 2007) book 'Data analysis and graphics using R: an example-based approach' by Maindonald and Braun, I realise that I should have also included the use of **the cross-validation error rate** to construct the best tree with

rpart().
A very simple use of the tree() function is given in Worksheet 15, below, for the Cushing's dataset. Here are the first 50 rows of the data-set from the Venables and Ripley library(MASS), originally from D.Michie (1989).

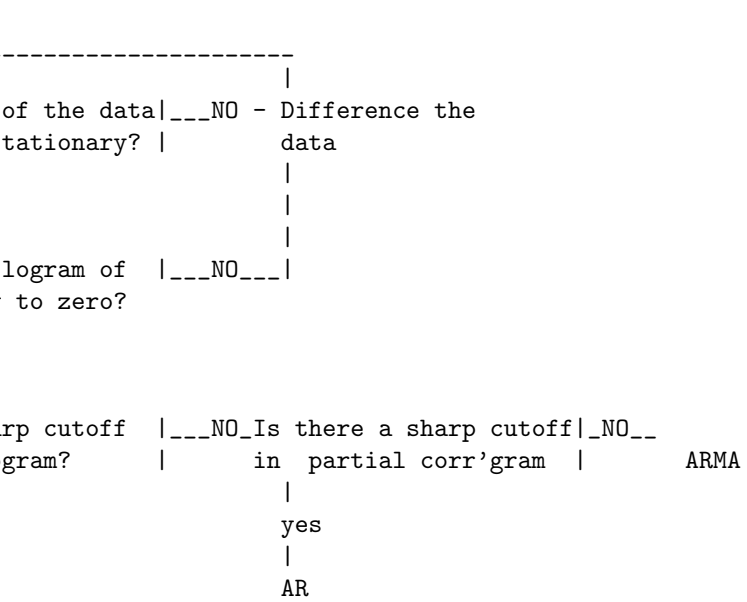
```
   stability error sign wind   magn vis  use
 1     xstab    LX   pp head  Light  no auto
 2     xstab    LX   pp head Medium  no auto
 3     xstab    LX   pp head Strong  no auto
 4     xstab    LX   pp tail  Light  no auto
 5     xstab    LX   pp tail Medium  no auto
 6     xstab    LX   pp tail Strong  no auto
 7     xstab    LX   nn head  Light  no auto
 8     xstab    LX   nn head Medium  no auto
 9     xstab    LX   nn head Strong  no auto
10     xstab    LX   nn tail  Light  no auto
11     xstab    LX   nn tail Medium  no auto
12     xstab    LX   nn tail Strong  no auto
13     xstab    XL   pp head  Light  no auto
14     xstab    XL   pp head Medium  no auto
15     xstab    XL   pp head Strong  no auto
16     xstab    XL   pp tail  Light  no auto
17     xstab    XL   pp tail Medium  no auto
18     xstab    XL   pp tail Strong  no auto
19     xstab    XL   nn head  Light  no auto
20     xstab    XL   nn head Medium  no auto
21     xstab    XL   nn head Strong  no auto
22     xstab    XL   nn tail  Light  no auto
23     xstab    XL   nn tail Medium  no auto
24     xstab    XL   nn tail Strong  no auto
25     xstab    MM   pp head  Light  no auto
26     xstab    MM   pp head Medium  no auto
27     xstab    MM   pp head Strong  no auto
28     xstab    MM   pp tail  Light  no auto
29     xstab    MM   pp tail Medium  no auto
30     xstab    MM   pp tail Strong  no auto
31     xstab    MM   nn head  Light  no auto
32     xstab    MM   nn head Medium  no auto
33     xstab    MM   nn head Strong  no auto
34     xstab    MM   nn tail  Light  no auto
35     xstab    MM   nn tail Medium  no auto
36     xstab    MM   nn tail Strong  no auto
37     xstab    SS   pp head  Light  no auto
38     xstab    SS   pp head Medium  no auto
39     xstab    SS   pp head Strong  no auto
40     xstab    SS   pp tail  Light  no auto
41     xstab    SS   pp tail Medium  no auto
42     xstab    SS   pp tail Strong  no auto
```

```
43     xstab    SS   nn head  Light  no auto
44     xstab    SS   nn head Medium  no auto
45     xstab    SS   nn head Strong  no auto
46     xstab    SS   nn tail  Light  no auto
47     xstab    SS   nn tail Medium  no auto
   stability error sign wind   magn vis  use
48     xstab    SS   nn tail Strong  no auto
49      stab    LX   pp head  Light  no auto
50      stab    LX   pp head Medium  no auto
```

Chapter 9

Introduction to Time-Series modelling

Diggle, 1990, p169, gives this excellent flowchart for guidance in arima modelling



We follow the approach in Venables and Ripley, and also use a PMEA data set.

```
library(MASS)
deaths    #  total UK monthly deaths from lung diseases for 1974-9
tsplot(deaths)
sablplot(sabl(deaths),title= "deaths") # seasonal components
acf(deaths)
acf(deaths,type= "partial")
```

```
spectrum(deaths)
spectrum(deaths,spans = 3)  # smoothed spectrum
spectrum(deaths,spans= c(3,3))
spectrum(deaths,"ar")
```

Now another dataset shown in Diggle, p42, on luteinising hormone.

```
lh ; tsplot(lh)
acf(lh)                   # looks like AR(1) or ARMA(1,1)
acf(lh,type= "partial")
spectrum(lh,"ar")
ar1 <- ar(lh,,1)
ar2 <- ar(lh)              #allowed free rein,chooses AR(3)
arima1 <- arima.mle(lh,model= list(order=c(1,0,0)))
                          # full MLE fit
2*arima1$loglik           # deviance - constant
arima.diag(arima1)        # diagnostics plot
arima3 <- arima.mle(lh,model=list(order= c(3,0,0)))
2*arima3$loglik           # not much better than AR(1)
arima.diag(arima3)
arima11 <- arima.mle(lh,model=list(order= c(1,0,1)))
2*arima11$loglik          #no better than AR(1)
arima.diag(arima11)
# Now use arima1 to forecast 12 steps
lh.fore <- arima.forecast(lh,n =12,model =arima1$model)
x <- lh.fore$mean ; sd <- lh.fore$std.err
tsplot(lh,x,x+2*sd,x-2*sd)
```

Now some popmusic data from 'The Independent', February 1994. First copy my files Splus/popmusic and Splus/popdata

```
source("popmusic")    # (this assumes you have BOTH files)
tsplot(ind)
acf(ind)
lind <- log(ind+1) ;tsplot(lind)  # and so on
```

Can you model the log-index? What is your prediction for 1994 ? Here is the popmusic file.
Data from 'The Independent', Wed Feb 23, 1994 "An Index of British penetration of the US singles market". The scoring system is :
give 30 points for the year's best-selling single, and go on down the scale to 1 point for the single that came 30th in that year's sale.
Thus the figure for 1993 is 28 pts for "UB40"(the 3rd best-seller)
+ 4 pts for "The Proclaimers"(27th)

```
pdata<- read.table("popdata", header=T); attach(pdata)
plot(year,ind)
# Here is the ``popdata" file.
year  ind
1960  0
```

```
1961   0
1962  14
1963   0
1964 179
1965 219
1966 131
1967 102
1968  48
1969  71
1970  61
1971  76
1972  38
1973  78
1974  36
1975 132
1976 105
1977 102
1978 166
1979  76
1980 142
1981  65
1982  36
1983 137
1984 111
1985 201
1986  70
1987  25
1988 170
1989  31
1990  38
1991  40
1992  48
1993  32
```

Chapter 10

Survival Data Analysis

We follow closely Venables and Ripley (1994) Chapter 11.
Two data-sets are used:
i)uncensored data on survival times for leukaemia (see Cox and Oakes, 1984, p9)
ii) The 2-sample Gehan data on remission times for leukaemia(Cox and Oakes, 1984, p7)

```
library(MASS)
attach(leuk) ; leuk
plot(log(time)~ag + log(wbc))       #log() is variance-stabilising here.
plot(survfit(Surv(time)~ag), lty= c(2,3))
```

These graphs suggest that
survival is BETTER with ag present than with ag absent, and
survival DECREASES as log(wbc) INCREASES.

```
legend(80,0.8,c("ag absent","ag present"),lty= c(2,3))
options(contrasts<-c("contr.treatment","contr.poly"))
leuk.glm <- glm(time ~ ag* log(wbc),Gamma(log))
```

Here we fit a gamma model, using the log-link. Check that you can write down the likelihood.

```
summary(leuk.glm,dispersion= 1)# sets df of gamma as 1. Thus, we have neg. exponential.
anova(leuk.glm)               # what is this telling us ?
# We drop the interaction term
leuk.glm <- update(leuk.glm, ~ . - ag:log(wbc))
summary(leuk.glm,dispersion= 1)
leuk.glmi <- glm(time ~ag*log(wbc),Gamma(inverse))
```

Does using the canonical link function improve the fit?

```
summary(leuk.glmi,dispersion= 1)
```

Again, we are forcing a neg exponential fit. Now we use survreg(), for exponential, Weibull and log-logistic regression analyses.

```
survreg(Surv(time) ~ag*log(wbc),dist= "exponential")
summary(survreg(Surv(time)~ag + log(wbc),dist= "exp"))
summary(survreg(Surv(time)~ag+log(wbc)))
summary(survreg(Surv(time)~ag+log(wbc),dist= "log"))
```

Now we will use a semi-parametric model, the Cox proportional hazards.

```
leuk.cox <- coxph(Surv(time)~ ag + log(wbc))
summary(leuk.cox)
detach("leuk")                  #to tidy our space.
```

Next we find the product-limit estimators of survival curves.

```
attach(gehan); gehan
plot.factor(gehan)
plot(log(time) ~ pair)          # variance- stabilising transformation again.
```

Now we will estimate the survivor function, using Greenwood's formula for standard errors. Some of what is written below is now out of date, since your version of Splus may have survfit() rather than surv.fit(). See Venables and Ripley, 1999, p371, for a method which replaces surv.fit() by survfit().

```
wt1 <- ifelse(treat=="control",1,NA) # to pick out control group
wt2 <- ifelse(treat=="6-MP",1,NA)    # to pick out treatment group
wt1 ; wt2                           # to check
fit1 <- surv.fit(time,cens,wt= wt1,type= "kaplan-meier",error= "greenwood")
fit1
fit2 <- surv.fit(time,cens,wt= wt2,type= "kaplan-meier",error= "greenwood")
fit2
surv.plot(time,cens,treat,lty= c(3,1),yscale= 100,
      xlab= "time of remission",ylab= "% survival")
legend(25,90,c("control","6-MP"),lty= c(1,3))

# or,a diy version,which has error-bars
plot(stepfun(fit1$time,fit1$surv),type= 'l',ylim= c(0,1),
      xlab= "time of remission",ylab= "survival")
t1 <- fit1$time ; s1 <- fit1$surv ; std1 <- fit1$std.err
t2 <- fit2$time ; s2 <- fit2$surv ; std2 <- fit2$std.err
lines(stepfun(t1,exp(log(s1) + 1.96*std1)),lty= 2)
lines(stepfun(t1,exp(log(s1) - 1.96*std1)),lty= 2)
lines(stepfun(t2,s2),lty= 3)
lines(stepfun(t2,exp(log(s2) + 1.96*std2)),lty= 2)
lines(stepfun(t2,exp(log(s2) - 1.96*std2)),lty= 2)
legend(1,0.2,c("control","6-MP","95% conf.int."),lty= c(1,3,2))
# or, use the packet-recipe
gehan.surv <- survfit(Surv(time, cens) ~ treat,conf.type= "log-log")
summary(gehan.surv)
plot(gehan.surv,conf.int= T,lty=c(3,2),log= T,
       xlab= "time of remission(weeks)",ylab=  "survival")
```

```
survreg(Surv(time,cens) ~ factor(pair)+treat,dist= "exp")
summary(survreg(Surv(time,cens)~treat,dist= "exp")
summary(survreg(Surv(time,cens)~treat))

help(surv.fit)
```

This enables us to find out about other options.
Now, to test for a difference between the 2 groups:

```
survdiff(Surv(time,cens) ~ treat, rho=0)    # This is the log-rank test
survdiff( Surv(time,cens) ~ treat,rho=1)    # almost Gehan-Wilcoxon test
                    # see Cox & Oakes p 124
```

Chapter 11

Survival analysis for the British monarchy

Under the heading
"How long British monarchs have lived",
the Independent on Sunday (26/11/95) gave the Table below. This gives, for each of 40 monarchs, the date of death, the lifetime, and a 0 or 1 according to whether the death was natural or not. (The list omits Lady Jane Grey, who was executed aged 16 in 1553, after 2 weeks on the throne. Mary, wife of William of Orange, is listed separately as she was Queen in her own right.)

	death	length	natural
WilliamI	1087	60	0
WilliamII	1100	40	1
HenryI	1135	67	0
Stephen	1154	53	0
HenryII	1189	56	0
RichardI	1199	42	1
John	1216	48	0
HenryIII	1272	65	0
EdwardI	1307	68	0
EdwardII	1327	43	1
EdwardIII	1377	64	0
RichardII	1399	33	1
HenryIV	1413	47	0
HenryV	1422	34	0
HenryVI	1471	49	1
EdwardIV	1483	40	0
EdwardV	1483	12	1
RichardIII	1485	32	1
HenryVII	1509	52	0
HenryVIII	1547	55	0
EdwardVI	1553	15	0
Mary	1558	42	0
ElizabethI	1603	69	0

```
JamesI      1625    58     0
CharlesI    1649    48     1
CharlesII   1685    54     0
JamesII     1701    67     0
WilliamIII  1702    51     1
Mary(II)    1694    32     0
Anne        1714    49     0
GeorgeI     1727    67     0
GeorgeII    1760    76     0
GeorgeIII   1820    81     0
GeorgeIV    1830    67     0
WilliamIV   1837    71     0
Victoria    1901    81     0
EdwardVII   1910    68     0
GeorgeV     1936    70     0
EdwardVIII  1972    77     0
GeorgeVI    1952    56     0
```

We use R to plot the Survivor function for the **natural** lifetimes (so that, for example, William II counts as a CENSORED observation.)

```
library(survival}
monarchy.data = read.table("monarchy.data", header=T)
attach(monarchy.data) ; cens= 1-natural
Surv(length, cens)
fit = survfit(Surv(length, cens)~ 1) ; fit
summary(fit)
plot(fit) ;  abline(.5,0)
```

and here is the resultant survivor function. Counting forward from age 0 years, the first observed natural death was for Edward VI, who died aged 15 years: there are only 39 monarchs at risk at this age as (poor little) Edward V has been 'censored' at 12 years old.

```
time n.risk n.event survival std.err   lower 95% CI upper 95% CI
   15     39       1   0.9744  0.0253       0.9260        1.000
   32     38       1   0.9487  0.0353       0.8820        1.000
   34     35       1   0.9216  0.0435       0.8402        1.000
   40     34       1   0.8945  0.0499       0.8018        0.998
   42     32       1   0.8666  0.0557       0.7640        0.983
   47     29       1   0.8367  0.0612       0.7249        0.966
   48     28       1   0.8068  0.0659       0.6874        0.947
   49     26       1   0.7758  0.0703       0.6495        0.927
   52     23       1   0.7420  0.0749       0.6088        0.904
   53     22       1   0.7083  0.0787       0.5696        0.881
   54     21       1   0.6746  0.0819       0.5317        0.856
   55     20       1   0.6408  0.0845       0.4950        0.830
   56     19       2   0.5734  0.0880       0.4244        0.775
   58     17       1   0.5397  0.0891       0.3905        0.746
```

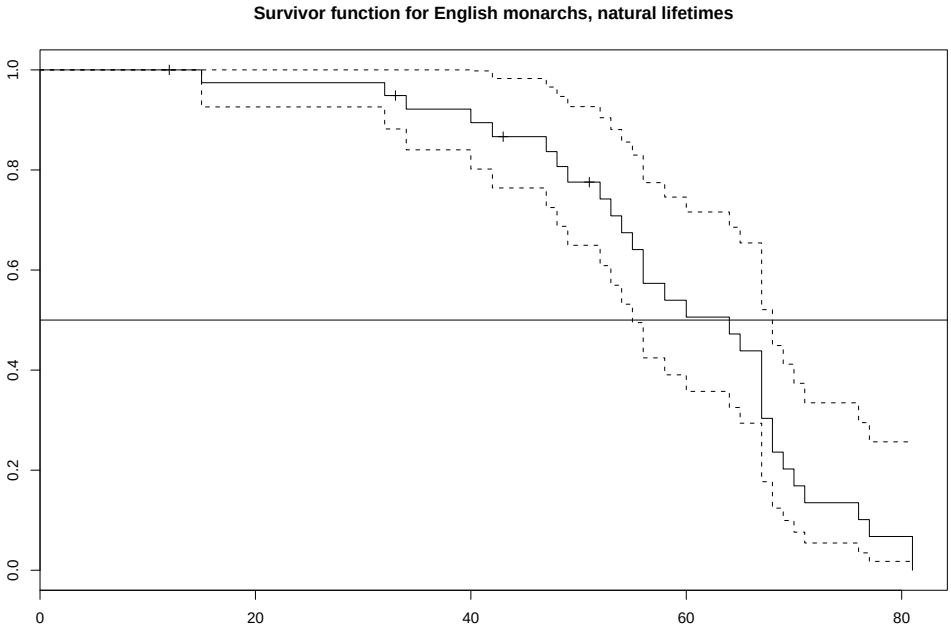


Figure 11.1: Survivor function for natural lifetime of English monarchs

60	16	1	0.5059	0.0897	0.3575	0.716
64	15	1	0.4722	0.0898	0.3253	0.686
65	14	1	0.4385	0.0895	0.2939	0.654
67	13	4	0.3036	0.0836	0.1769	0.521
68	9	2	0.2361	0.0774	0.1241	0.449
69	7	1	0.2024	0.0734	0.0994	0.412
70	6	1	0.1686	0.0684	0.0761	0.374
71	5	1	0.1349	0.0625	0.0544	0.335
76	4	1	0.1012	0.0552	0.0347	0.295
77	3	1	0.0675	0.0460	0.0177	0.257
81	2	2	0.0000	NA	NA	NA

The resulting plot is given in Figure 11.1.

Now compute the Kaplan-Meier estimates of the survivor function for the male monarchs and for the female monarchs, and try fitting parametric distributions to these. (Note, there are just 5 queens in the list.)

Chapter 12

Classical Metric Multidimensional Scaling and Chernoff's faces

```
a <- read.table("Dip97",header= T) # reads in the responses from 16 students
student.lab <- row.names(a)
a <- as.matrix(a)
d <- dist(a,metric= "binary")  # This  sets up the interstudent "distances"
new <- cmdscale(d,k= 2,eig= T) ; new
```

This finds the best 2-dimensional representation of the 16 points.

```
coord1 <- new$points[,1] # the first column
coord2 <- new$points[,2] # the second
par(pty="s")            # sets up a square plot
r <- range(new$points)
plot(coord1,coord2,type= "n")
text(coord1,coord2,seq(along= coord1)).
```

This labels the points by integers. Alternatively, we could use the default setting of a 2-dimensional representation, thus:

```
new <- cmdscale(d)
plot(new,type= "n")
text(new,labels= student.lab) # this time put the NAMES on the plot
faces(a, labels= student.lab)
```

How to insult your students!
Chernoff's faces (available in R via the package aplpack) represent up to 15 variables by features of cartoon faces as you will see in Figure 12.1.
The corresponding data set, 'Dip97', is given below.

```
       eggs meat coffee beer UKres Cantab Female Sports Driver Left.h
Anna      1    1      1    0     0      0      1      1      1      0
Rachel    1    1      1    1     1      1      1      1      1      0
```

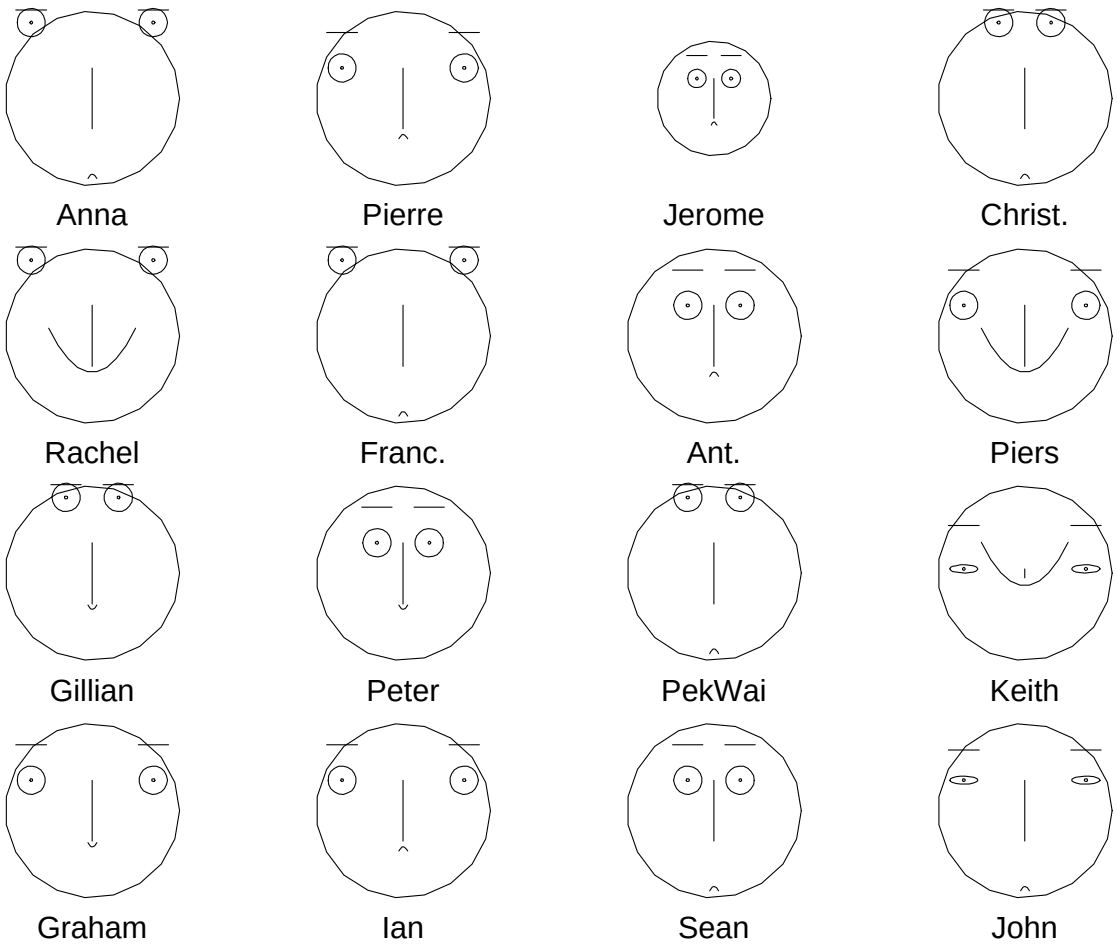


Figure 12.1: Chernoff's faces for the Diploma 1996-7 class

Gillian	1	1	1	1	1	0	1	0	1	0
Graham	1	1	1	1	1	0	0	1	1	0
Pierre	1	1	1	1	0	0	0	1	1	0
Franc.	1	1	1	0	0	0	1	1	1	0
Peter	1	1	1	1	1	0	0	0	1	0
Ian	1	1	1	1	0	0	0	1	1	0
Jerome	0	1	1	1	0	0	0	0	1	0
Ant.	1	1	1	1	0	0	0	0	1	0
PekWai	1	1	1	0	0	0	1	0	1	0
Sean	1	1	1	0	0	0	0	0	1	0
Christ.	1	1	1	0	0	0	1	0	1	0
Piers	1	1	1	1	1	1	0	1	1	0
Keith	1	1	0	1	1	1	0	1	1	1
John	1	1	1	0	0	0	0	1	1	1

And finally, new for 2002, the following data MPhil/Part III, applied multivariate analysis, Feb 2002.

	eggs	meat	coffee	beer	UKres	Cantab	Fem	sports	driver	Left-h	specs
Josh	y	y	y	y	y	n	n	y	n	n	y
TjunKiat	y	y	y	n	n	n	n	y	y	n	y
Flora	y	y	y	y	y	n	y	y	y	n	y
ChauLoong	y	y	y	n	n	n	n	y	n	n	n
Eleanor	y	y	y	y	y	n	y	y	y	n	n
Teresa	y	y	y	n	n	n	y	y	y	n	n
Jim	y	y	y	y	y	y	n	y	y	y	n
Mama	y	y	y	y	n	n	n	y	y	n	n
Chao	y	y	y	y	n	n	n	y	y	n	y
Qi	y	y	y	y	y	n	n	y	n	n	y
LeeLee	y	y	n	y	n	n	y	y	y	n	y
Karthi	y	y	y	n	n	n	n	n	n	n	y
David	y	y	y	y	y	n	n	y	y	n	y
Neeraj	y	y	n	n	y	n	n	n	n	n	y
Cosme	y	y	n	n	n	n	n	y	y	n	y
Arnaud	y	y	y	y	n	n	n	y	y	n	y
Jochen	y	y	y	n	n	y	n	y	y	n	y
Sophia	y	y	y	n	y	y	y	y	y	n	y
Stephane	y	y	n	n	n	n	n	y	y	n	n
JimmyL	y	y	y	y	n	n	n	y	y	n	y

Note, the first 2 columns turn out to be unhelpful, so you may prefer to omit them before trying, eg

dist() for use with hclust() or cmdscale()

The above data set is of course based on rather trivial questions.

By way of complete contrast, here is a data set from The Independent, Feb 13, 2002. on 'Countries with poor human rights records where firms with British links do business'. It occurs under the

headline

CORPORATE RISK: COUNTRIES WITH A BRITISH CONNECTION.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
SaudiArabia	1	0	0	0	0	1	0	1	0	0	1	1	0	1
Turkey	1	0	1	0	1	1	0	0	0	1	0	1	0	1
Russia	1	0	1	0	1	1	1	0	0	0	0	1	0	1
China	1	1	1	0	1	1	1	0	0	0	0	1	0	1
Philippines	1	1	1	0	0	0	0	0	1	0	0	1	1	0
Indonesia	1	1	1	0	0	1	1	1	0	0	0	1	0	0
India	1	0	1	0	1	0	0	1	1	0	1	1	0	0
Nigeria	0	0	1	0	0	0	1	0	0	0	0	1	1	0
Brazil	1	0	1	1	1	0	1	0	0	1	0	1	0	0
Colombia	1	1	1	1	1	0	0	0	0	1	0	1	0	0
Mexico	0	1	1	0	0	1	0	0	0	0	0	1	0	1

Key to the questions (1 for yes, 0 for no)
Violation types occurring in the countries listed

1 Torture
2 'Disappearance'
3 Extra-judicial killing
4 Hostage taking
5 Harassment of human rights defenders
6 Denial of freedom of assembly and association
7 Forced labour
8 Bonded labour
9 Bonded child labour
10 Forcible relocation
11 Systematic denial of women's rights
12 Arbitrary arrest and detention
13 Forced child labour
14 Denial of freedom of expression

Note that the *total* number of 1's in each row ranges from 4, for Nigeria, to 8, for China.
Figure 12.2 shows my 2-dimensional plot of the 11 countries, using the

```
method ="binary"
```

option in computing the between-countries distance matrix. (Of course, this treats the 14 different types of 'violation' as equally serious, which is not necessarily the correct thing to do.) In order to interpret the axes of this graph, I suggest the following:

```
a <- a[,-12] # to remove the 12th column from the matrix (it's all 1's)
b <- cbind(new, a) # new being the first 2 cmd co-ordinates
round(cor(b),2) # so that you can see, for example, which columns of a
are most closely correlated with new[,1]
```

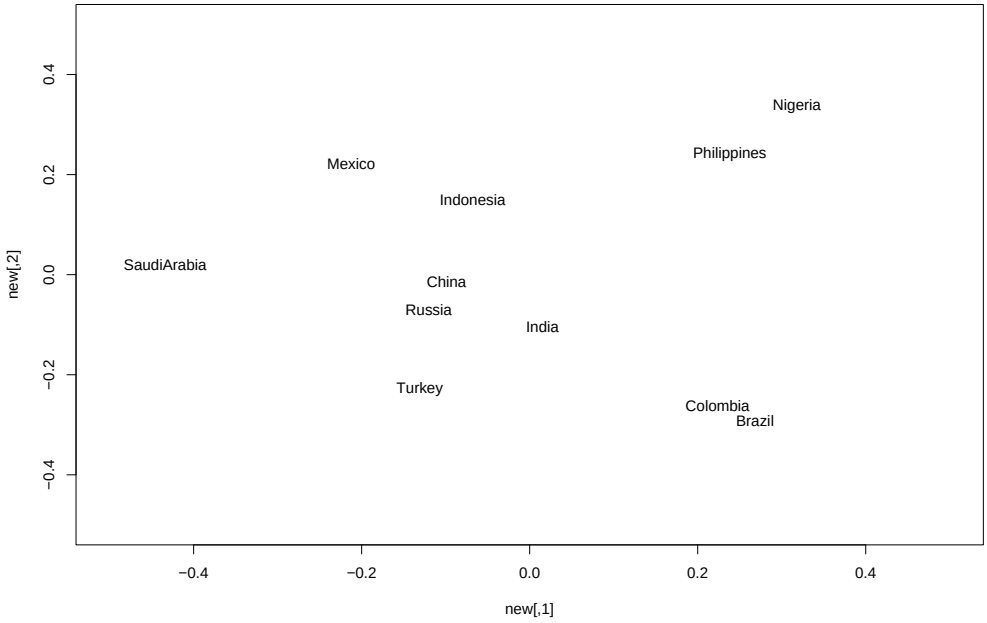


Figure 12.2: Countries with a British connection: human rights abuses

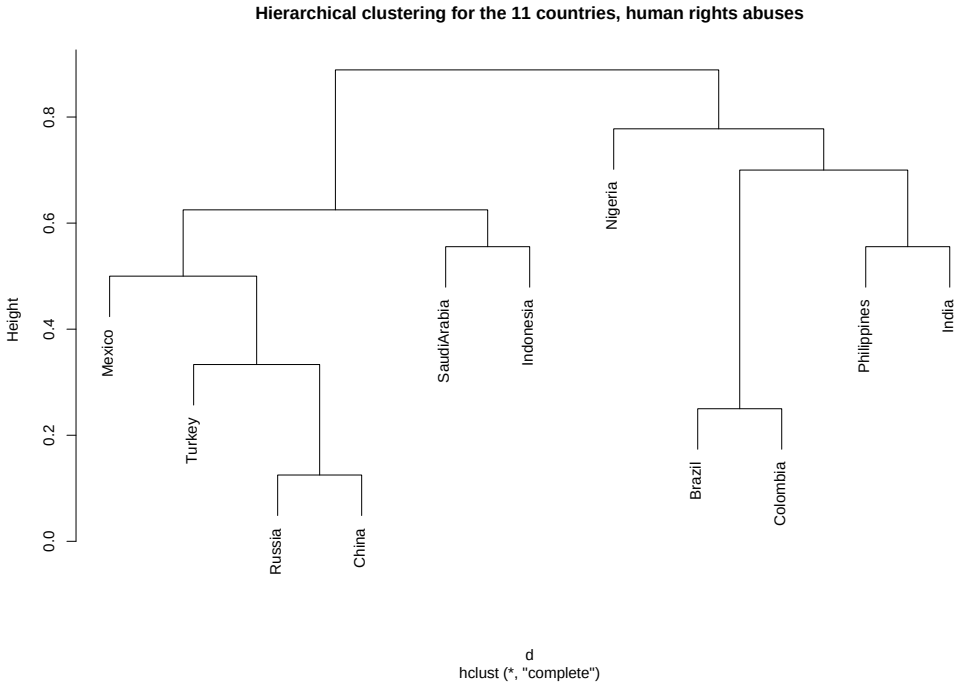


Figure 12.3: Hierarchical clustering for the countries data

You might like to compare the results of cmdscale with those of hierarchical clustering, as follows.

```
a = read.table("human.rights") # to read in the data
a = data.matrix(a)
d = dist(a, method="binary")
h = hclust(d, method="complete")
 # "complete" in R is same as "compact" in Splus
plclust(h)
```

The resulting graph is shown as Figure 12.3.

Finally, an interesting financial data set, as found on the BBC website in November 2011.
Under the headline 'Eurozone debt web: Who owes what to whom?' the BBC website `http://www.bbc.co.uk/news/business-15748696` gives the following figures on how much each of various major economies owes to each other, expressed in billions of Euros, from data collected from their various banks at the end of June 2001. (China is known to hold European debt, but no comprehensive figures are available.)
You have to read this table DOWN the columns: thus for example the UK owes 578.6 to the US, and Spain owes 74.9 to the UK. (I had to read the figures from 10 different graphs, so I can't be absolutely sure I read the figures correctly: you are encouraged to check for yourselves from the website.) The 10 countries given are UK, US, France, Spain, Portugal, Italy, Ireland, Greece, Japan, Germany.

	UK	US	Fra	Sp	Port	It	Ire	Gre	Jap	Ger
UK	NA	834.5	227	74.9	18.9	54.7	104.5	9.4	101.8	141.1
US	578.6	NA	202.1	49.6	3.6	34.8	39.8	6.2	244.8	174.4
Fra	209.9	440.2	NA	112	19.1	309	23.8	41.4	107.7	205.8
Sp	316.6	170.5	0	NA	65.7	29.5	0	0	0	0
Port	0	0	0	19.7	NA	0	0	7.5	0	0
It	0	0	37.6	22.3	2.9	NA	0	2.8	0	202.7
Ire	113.5	0	0	0	0	0	NA	0	0	0
Gre	0	0	0	0	0	0	0	NA	0	0
Jap	122.7	835.2	79.8	20	0	32.8	15.4	0	NA	108.3
Ger	379.3	414.5	123.5	131.7	26.6	120	82	15.9	42.5	NA

Can you think of a sensible way to use this matrix to construct a distance matrix between the 10 countries, and hence to obtain a single 2-dimensional plot of the countries?

Chapter 13

Analysis of a Repeated Measures design

You see below the data from p28 of M.J.Crowder and D.J.Hand (1990) 'Analysis of Repeated Measures'.
To quote from Crowder and Hand, 'The effect of a vitamin E diet supplement on the growth of guinea pigs was investigated as follows. For each animal the body-weight was recorded at the ends of weeks 1,3,4,5,6 and 7. All animals are given a growth-inhibiting substance during week 1, and the vitamin E therapy was started at the beginning of week 5. Three groups of animals, numbering five in each, received respectively zero, low and high doses of vitamin E.'

The body weights (in grams) are given in the table below. The rows correspond to Animals $1, \ldots 15$, respectively, and the columns to the weeks $1, 3, 4, 5, 6, 7$. The first 5 rows are Group 1, the next 5 are Group 2, and the final 5 are Group 3. We reconstruct the analysis given by Crowder and Hand on p34, following Venables and Ripley (1997) Chapter 10.
This model allows for *three* sources of random variation: one is that between the 15 animals, one is the random interaction effect animals $\times$ occasions and and finally one is the 'error' variation.
The model to be fitted is

$$x_{ij} = \mu_{ij} + \alpha_{ij} + +\epsilon_{ij}$$

for $i = 1, \cdots, 15, j = 1, \cdots, 6$, where we assume that

$$\alpha_{ij} = \alpha_i^I + \alpha_{ij}^{IO}$$

where $\alpha_i^I, \alpha_{ij}^{IO}, \epsilon_{ij}$ are independent, with variances $\sigma_I^2, \sigma_{IO}^2, \sigma^2$ respectively. (The first 2 of these 3 terms are known as variance components.)
We assume that

$$\mu_{ij} = \mu_j^{(g)} \text{ for } \; i\epsilon \text{ Group } g.$$

```
x <- scan()
455 460 510 504 436 466
```

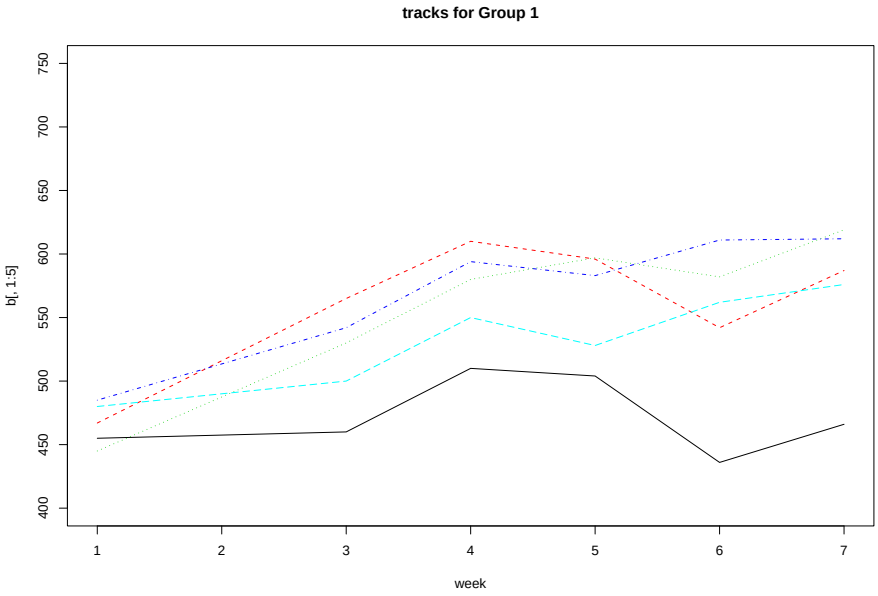


Figure 13.1: Time tracks showing growths for Group 1 guinea pigs

```
467 565 610 596 542 587
445 530 580 597 582 619
485 542 594 583 611 612
480 500 550 528 562 576
514 560 565 524 552 597
440 480 536 484 567 569
495 570 569 585 576 677
520 590 610 637 671 702
503 555 591 605 649 675
496 560 622 622 632 670
498 540 589 557 568 609
478 510 568 555 576 605
545 565 580 601 633 649
472 498 540 524 532 583

We <- c(1,3,4,5,6,7)
week <- We
```

First we plot the 15 'timetracks', on 3 separate plots, one for each of the 3 Groups. These are shown as Figures 13.1, 13.2 and 13.3 respectively.

```
a <- matrix(x,nrow= 15,ncol= 6,byrow= T)
b<- t(a)
par(mfrow=c(3,1))
matplot(week,b[,1:5],type= "l", ylim= c(400,750))
```

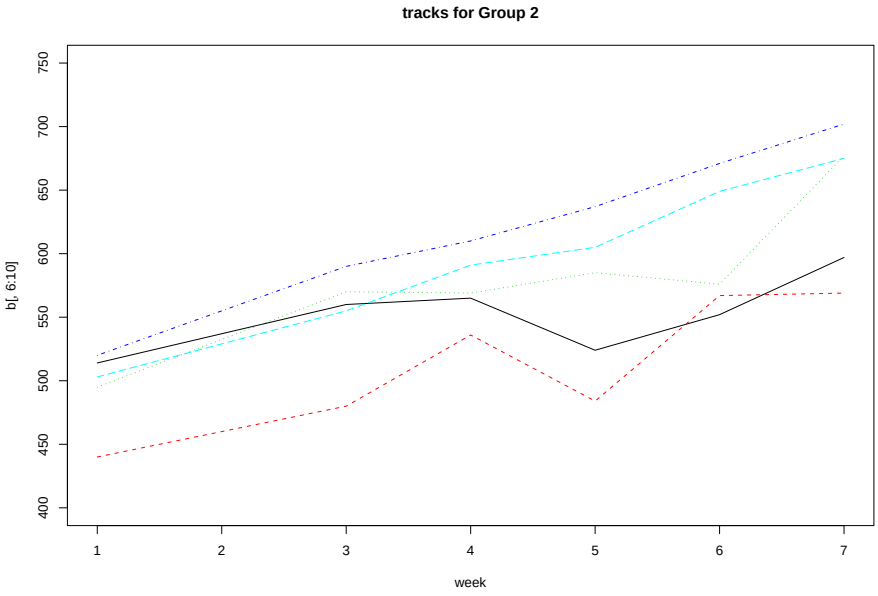


Figure 13.2: Time tracks showing growths for Group 2 guinea pigs

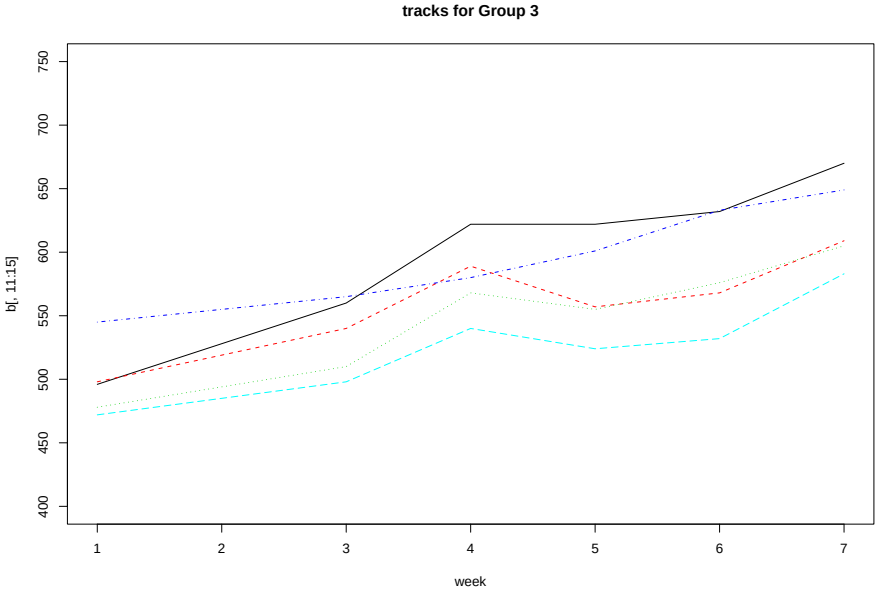


Figure 13.3: Time tracks showing growths for Group 3 guinea pigs

```
matplot(week,b[,6:10],type= "l", ylim= c(400,750))
matplot(week,b[,11:15],type= "l", ylim= c(400,750))
library(nlme)  # for use in R
Gr <- 1:3 ; An <- 1:15
y <- expand.grid(We,An)
Week <- y[,1] ; Animal <- y[,2]
Group <- gl(3, 30, length=30, labels=c("zero", "low", "high")
Week <- factor(Week); Animal <- factor(Animal)
first.aov <- aov(x~Week*Group + Error(Animal))
summary(first.aov)
```

This shows that the Group*Week interaction is non-significant. So next we try

```
sec.aov <- aov(x~ Week + Group + Error(Animal))
summary(sec.aov)
```

This results in the following output, where you can see that the original 89 df have been partitioned into $89 = (2+12)+(5+70)$, giving us the 'between Animals' comparisons and the 'Within Animals' comparisons, respectively.

```
Error: Animal
          Df Sum Sq Mean Sq F value Pr(>F)
Group      2  18548    9274  1.0555 0.3782
Residuals 12 105434    8786

Error: Within
          Df Sum Sq Mean Sq F value    Pr(>F)
Week       5 142555   28511  47.164 < 2.2e-16 ***
Residuals 70  42315     605

sec.lme <- lme(x ~ Week + Group, random= ~1 | Animal)
summary(sec.lme) # for comparison
> summary(sec.lme)
Linear mixed-effects model fit by REML
 Data: NULL
       AIC      BIC    logLik
  831.9379 856.0051 -405.9689

Random effects:
 Formula: ~1 | Animal
        (Intercept) Residual
StdDev:    36.92713 24.58668

Fixed effects: x ~ Week + Group
               Value Std.Error DF   t-value p-value
(Intercept) 466.2333 18.068104 70 25.804220  0.0000
Week3        48.8000  8.977786 70  5.435639  0.0000
Week4        88.0667  8.977786 70  9.809397  0.0000
```

```
Week5        80.6000  8.977786 70  8.977715  0.0000
Week6        93.0667  8.977786 70 10.366328  0.0000
Week7       126.8667  8.977786 70 14.131176  0.0000
Grouplow     33.1333 24.202181 12  1.369023  0.1961
Grouphigh    26.7667 24.202181 12  1.105961  0.2904
 Correlation:
................................................
Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-2.62053931 -0.51705480  0.01798091  0.62523764  2.09203976

Number of Observations: 90
Number of Groups: 15
```

Note: I have used 'Groups' to mean treatments zero, low, high here. This was not such a good choice of name, as lme() uses 'Groups' to mean Animals in this context.
Compare this also with

```
summary(lm(x ~ Week + Group)) # which assumes that all observations are independent
```

Note added March 2009: To fit the random effects model in R, I believe that what you need to do is

```
library(lattice); library(Matrix) ; library(lme4)
```

Now use the function

```
lmer()
```

Venables and Ripley show you how to look at residuals.
The current example follows the analysis given by Hand and Everitt. However 'Week' is definitely ordered in *time*, and so there may be a more suitable error structure than the symmetric one given here. See Venables and Ripley (1997) p312 for examples of other error structures.
Note that the function

```
glmmPQL()
```

available via library(MASS) provides a very general method of dealing with 'random effects' versions of generalized linear models. But beware: Hayley Jones, in her MPhil Applied Project, found that SPlus6 and R give different solutions when using this function on identical datasets.
(The problem seems to be connected with the fact that we are maximising a multi-modal log-likelihood function, and R may go off in the wrong direction.) For this reason we preferred to work with the SPlus version of the function. In either case, understanding all the output from glmmPQL() is tricky.

Chapter 14

Fitting a beta-binomial distribution to the IVF hospitals data

I checked the computations in Splus7 in July 2008.
E.C.Marshall and D.J.Spiegelhalter (1998) 'Reliability of league tables of *in vitro* fertilisation clinics: retrospective analysis of live birth rates', British Medical Journal, **316**, 1701-4 analyse the data from which the Table below has been constructed. To quote from E.C.Marshall's unpublished PhD thesis, which also includes these data, 'In July 1996 the Human Fertilisation and Embryology Authority reported on 25730 *in vitro* fertilisation treatments carried out in 52 clinics over the period from 1 April 1994 to 31 March 1995. An overall adjusted live birth rate of 14.5 % was found.'
In the Table below, r is the number of live births, and n the number of fertilisations. (r was computed from n and the observed percentage p, both of which were given in Marshall's PhD thesis.)

	r	n
Withington	7	147
ManchesterFS	41	506
Fazakerley	20	240
Ninewells	42	501
Hull	33	390
King'sColl	125	1453
BMIChiltern	13	149
Cromwell	39	427
Aberdeen	32	327
Walsgrave	45	458
Hartlepool	9	85
BUPALe'ster	12	110
UCH	41	366
WirralFC	17	141
GlasgowRI	105	876
SheffieldFC	80	661

```
    Le'sterRI  14  114
     LondonFC 100  786
     StMary's  82  627
     NewhamGH   9   68
 EdinburghACU  59  447
  BMIPortland  21  152
   Washington  42  307
    RoyalVicI  47  342
  BourneHallC 185 1315
      UHWales  24  168
     BridgeFC  81  568
   EsperanceH  31  212
     WessexFS  60  404
   ChurchillC  78  519
    MidlandFS 120  787
  UnivBristol 119  773
    WolfsonFC 160 1004
 RoyalMasonic 133  839
  Northampton  36  223
      NStaffs  19  116
 LondonWomens 105  643
  Guys&StThom  84  496
      BMIPark 111  640
   BUPARoding  38  211
    HollyHoFU  49  262
    BMIPriory  46  241
  S.Cleveland  20  104
    LeedsGenI 186  946
BMIChelsfield  42  208
    OxfordIVF 128  603
 SouthmeadGen  18   82
    Lister    244 1104
RMHBelfast    122  548
 StJames's    121  537
Birmingham     60  267
   NURTURE    204  861
```

First we will fit the binomial with constant probability p to these data, namely

$$r_i \sim \text{independent } \ Bi(n_i, p), \ 1 \leq i \leq 52.$$

This is easily achieved by

```
hdata <- read.table("hospitals.data", header= T)
attach(hdata)
first.glm <- glm(r/n ~ 1, binomial, weights= n)
summary(first.glm)
```

which shows a deviance of 390.76, with $df = 51$. So we have substantial overdispersion with respect to the model of constant binomial parameter p. We will compute the binomial residuals,

for comparison later with the betabinomial residuals.

```
p <- first.glm$fitted.values ; q <- 1-p
res <- (r-n*p)/sqrt(n*p*q)
sum(res^2) # as a check
chisq.test(cbind(r,n-r)) # as another check
# sqrt(n) * resid(first.glm) would give us the deviance residuals instead
```

Our next step is to allow one extra parameter: we assume that

$$r_i|p_i \sim Bi(n_i, p_i)$$

and assume further that p_i has the beta distribution, parameters θ, ϕ.
This has the consequence that each r_i then has a beta-binomial distribution, parameters n_i, θ, ϕ.
Again assume that all the r_i's are independent.
We pause to derive the frequency function for the beta-binomial. Now

$$f(r|p) = \binom{n}{r} p^r (1-p)^{n-r}, \ for\ r = 0, \cdots, n$$

where p has density $g(p)$ say, where

$$g(p) = \frac{\Gamma(\theta+\phi)}{\Gamma(\theta)\Gamma(\phi)} \ p^{\theta-1}(1-p)^{\phi-1}, \text{ for } 0 \le p \le 1.$$

Thus, integrating with respect to p, we find that

$$\int f(r|p)g(p)dp = \binom{n}{r} \frac{\Gamma(\theta+\phi)}{\Gamma(\theta)\Gamma(\phi)} \frac{\Gamma(\theta+r)\Gamma(\phi+n-r)}{\Gamma(\theta+\phi+n)}.$$

In the commands below, we compute

$$-\Sigma_i log f(r_i|\theta, \phi)$$

as MINUS the loglikelihood function, and then minimise it to find the maximum likelihood estimates of θ, ϕ. 'General optimization and maximum likelihood estimation' is given as Chapter 9 in Venables and Ripley (1997).

```
lbetabin <- function(p)
{
th <- p[1]
phi <- p[2]
sum( - lgamma(th + r) - lgamma(phi + n - r) + lgamma(th + phi + n) +
lgamma(th) + lgamma(phi) - lgamma(th + phi))
}
p <- c(.15,.85)
```

These are our initial estimates of θ, ϕ, taken from the binomial fit, and setting $\theta + \phi = 1$. One way to proceed is as follows

```
fit.first <- nlmin(lbetabin,p,print.level= 1) # this does not quite converge, but
fit.first$converged # shows that we have not yet reached convergence, but
fit.first$x         # shows that we have
#  estimates theta =10.73 , phi=63.07. So we use these as starting values, thus
p <-  fit.first$x
fit.next <-  nlmin(lbetabin,p,print.level= -1) # now quickly converges, giving
# the following estimates
fit.next$x
 10.89 63.04  # for theta, phi
```

Now we will try a different minimisation function.

```
p <- c(.15,.85) # the  same starting values as before
fit.betabin <- nlminb(start =  p, objective =  lbetabin, lower =  c(0, 0))
 # which gives convergence, and
fit.betabin$parameters
[1] 10.89 63.06            # and we need the corresponding se's, so
library(MASS)
vcov.nlminb(fit.betabin)   # gives us the approximate covariance matrix for these
                            parameter estimates
```

It is interesting that we find

$$\hat{\theta} = 10.89(se = 2.51), \hat{\phi} = 63.06(se = 14.85)$$

which corresponds to the beta-density for p, shown in Figure 14.1, which is quite sharply peaked. You can do this plot for yourself by

```
th <- 10.89; phi <- 63.06
p <- (1:100)/100
f <- dbeta(p,th,phi)
plot(p,f,type= "l")
```

We can use the parameter estimates to compute the correct estimated variance for r_i, and hence compute a χ^2 goodness of fit statistic for the model.

```
 th <- 10.89; phi <- 63.06; pi <- th/(th + phi)
betabin.resid <- (r - n*pi)/sqrt( n*pi *(1-pi)*(1+ (n-1)/(th + phi+1)))
plot(res,betabin.resid)
betabin.chi2 <- sum(betabin.resid^2)
```

This finds the χ^2 statistic as 50.35, with 50 df, showing that the inclusion of just 1 extra parameter gives a model that satisfactorily accounts for the 'over-dispersion' relative to the ordinary binomial. Here are the ordered binomial residuals.

```
round(sort(res),2)
# This shows us 'best' and 'worst' on crude 1-parameter binomial model
 King'sColl ManchesterFS Ninewells  Hull Withington Cromwell Walsgrave
      -6.85        -4.36     -4.16 -3.63      -3.48     -3.4     -3.11
 Fazakerley Aberdeen GlasgowRI BMIChiltern SheffieldFC   UCH LondonFC StMary's
```

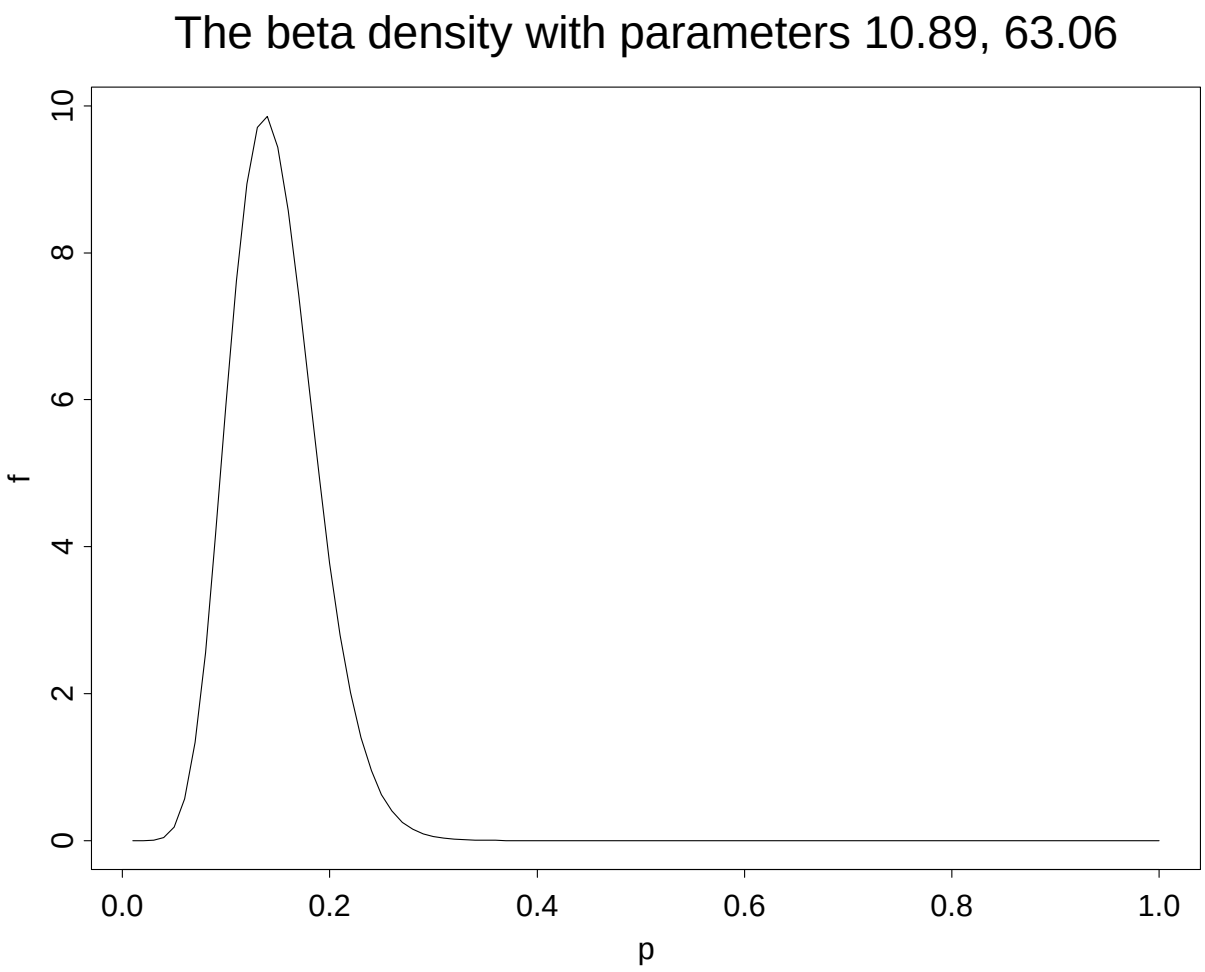


Figure 14.1: The beta density with parameters $10.89, 63.06$

```
      -2.9    -2.65     -2.51       -2.15        -2.1 -2.04     -1.8    -1.36
BUPALe'ster Hartlepool EdinburghACU WirralFC BourneHallC Le'sterRI RoyalVicI
      -1.21      -1.14        -1.08    -0.98       -0.97     -0.82     -0.66
Washington BridgeFC BMIPortland NewhamGH UHWales EsperanceH WessexFS
     -0.66    -0.51       -0.42    -0.41   -0.27      -0.16    -0.09
ChurchillC MidlandFS UnivBristol NStaffs Northampton RoyalMasonic WolfsonFC
      0.01      0.18        0.29    0.41        0.47         0.67      0.81
LondonWomens Guys&StThom S.Cleveland BUPARoding BMIPark HollyHoFU
        0.93        1.19         1.2       1.22    1.65      1.67
SouthmeadGen BMIPriory BMIChelsfield Birmingham LeedsGenI OxfordIVF
        1.76      1.77          2.09       3.41         4      4.27
RMHBelfast StJames's Lister NURTURE
      4.75      4.87   6.59    7.12
```

and here are the ordered beta-binomial residuals, which can also be compared to the standard normal

```
round(sort(betabin.resid),2) # for betabinomial residuals
Withington ManchesterFS King'sColl Ninewells Hull Fazakerley Cromwell
      -1.99        -1.51      -1.46     -1.45 -1.4      -1.37    -1.26

 BMIChiltern Walsgrave Aberdeen   UCH Hartlepool BUPALe'ster GlasgowRI
       -1.2     -1.11    -1.09 -0.79      -0.74       -0.72     -0.64

 SheffieldFC WirralFC LondonFC Le'sterRI StMary's EdinburghACU NewhamGH
       -0.61    -0.53    -0.47     -0.47    -0.38        -0.35    -0.25

 Washington RoyalVicI BMIPortland BourneHallC BridgeFC UHWales EsperanceH
      -0.23     -0.22       -0.18       -0.16    -0.11   -0.09      -0.02

 WessexFS ChurchillC MidlandFS UnivBristol RoyalMasonic WolfsonFC Northampton
    0.03       0.07      0.12        0.16         0.26      0.29         0.3

 NStaffs LondonWomens Guys&StThom BMIPark BUPARoding S.Cleveland HollyHoFU
   0.32         0.37         0.5    0.61       0.69        0.84      0.86

 BMIPriory BMIChelsfield LeedsGenI SouthmeadGen OxfordIVF Birmingham
      0.93          1.15      1.16         1.28       1.5       1.67

 RMHBelfast Lister StJames's NURTURE
       1.73   1.74      1.79     2.1
```

We could compare the 2 sets of residuals graphically via

```
par(mfrow= c(2,1))
qqnorm(res) ; qqline(res)
qqnorm(betabin.resid); qqline(betabin.resid)
```

This gives the graphs (note that the y-axes have different scales) shown in Figure 14.2.
Exercise:
The sample correlation matrix for $\hat{\theta}, \hat{\phi}$ suggests that we could find a much 'better' parametrisation, in which the two parameters are closer to being orthogonal. Experiment with the parametrisation

$$\pi = \theta/(\theta + \phi),\ \psi = \theta + \phi.$$

Afterword.
One of the objectives of Marshall and Spiegelhalter in looking at this table was to produce a 'reliable' ranking of the hospitals, since a ranking based only on the crude success rate can be quite misleading. How do we address this question with the benefit of our beta-binomial model?

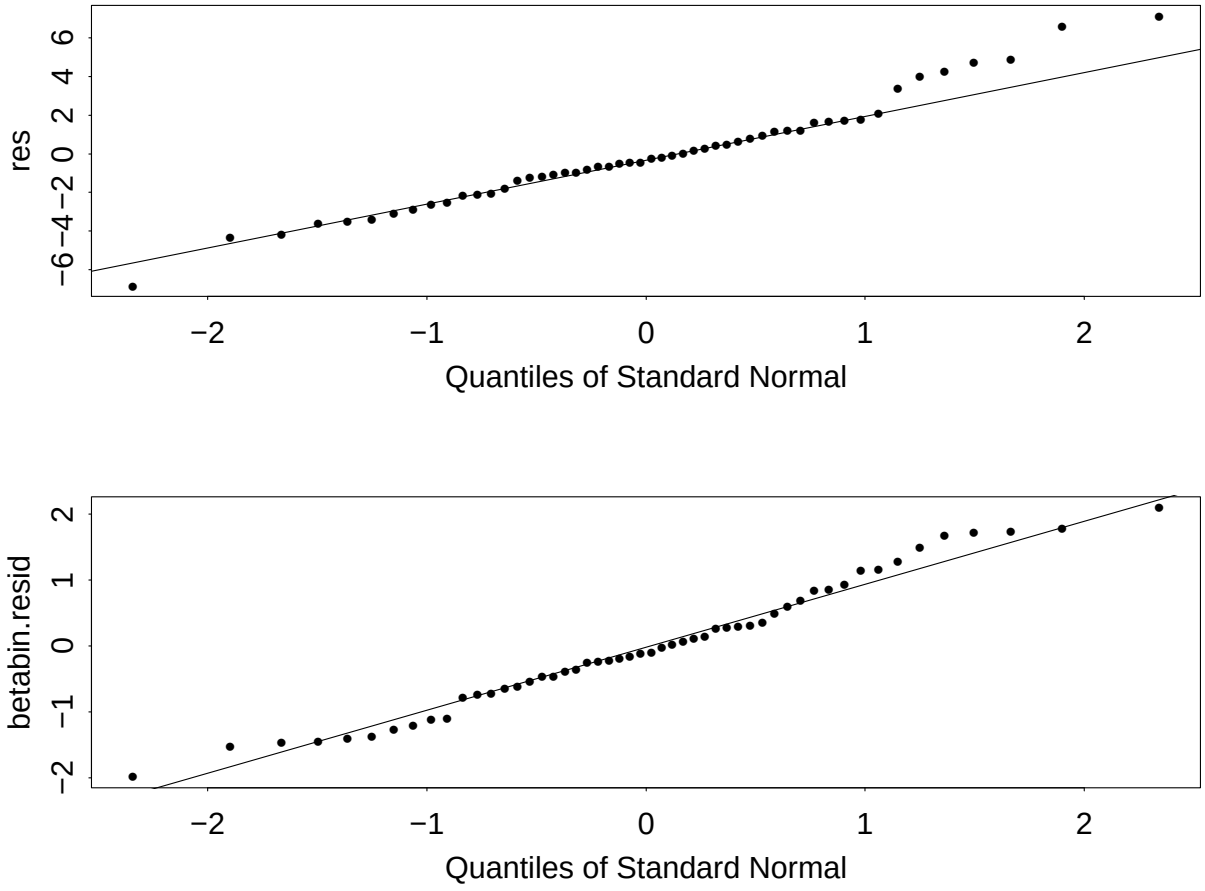


Figure 14.2: residuals for the binomial and the betabinomial distributions

Chapter 15

Multinomial logistic regression and classification

Here we follow the example on Cushing's syndrome, given in Venables and Ripley (1999) p350, and we give some supplementary explanation.
The dataset is given below.

	Tetrahydrocortisone	Pregnanetriol	Type
a1	3.1	11.70	a
a2	3.0	1.30	a
a3	1.9	0.10	a
a4	3.8	0.04	a
a5	4.1	1.10	a
a6	1.9	0.40	a
b1	8.3	1.00	b
b2	3.8	0.20	b
b3	3.9	0.60	b
b4	7.8	1.20	b
b5	9.1	0.60	b
b6	15.4	3.60	b
b7	7.7	1.60	b
b8	6.5	0.40	b
b9	5.7	0.40	b
b10	13.6	1.60	b
c1	10.2	6.40	c
c2	9.2	7.90	c
c3	9.6	3.10	c
c4	53.8	2.50	c
c5	15.8	7.60	c
u1	5.1	0.40	u
u2	12.9	5.00	u
u3	13.0	0.80	u
u4	2.6	0.10	u

```
u5              30.0          0.10    u
u6              20.5          0.80    u
```

The data 'are on diagnostic tests on patients with Cushing's syndrome, a hypersensitive disorder associated with over-secretion of cortisol by the adrenal gland. The dataset has three recognised types of the syndrome represented as

`a, b, c.`

(These encode 'adenoma', 'bilateral hyperplasia', and 'carcinoma', and represent the underlying cause of over-secretion. This can only be determined histopathologically.) The observations are urinary excretion rates (mg/24h) of the steroid metabolites tetrahydrocortisone and pregnanetriol, and are considered on a log scale.'
In the analysis given below, we do not use the last 6 rows of the data, for which the 'Type' was unknown. We fit the following logistic model

$$log(P(b|x)/P(a|x)) = \beta_2^T x, \quad log(P(c|x)/P(a|x)) = \beta_3^T x$$

with x as a 3-dimensional vector, having first element 1, and

$$P(a|x) + P(b|x) + P(c|x) = 1.$$

Thus, for example, if an object has covariate value x, we will predict it as b if $\beta_2^T x > 0$, & $\beta_2^T x > \beta_3^T x$. We use library(nnet) to maximise the resulting multinomial log-likelihood.

```
library(MASS)
library(nnet)
Cushings # to view the data
tp <- factor(Cushings$Type[1:21])
Cf <- data.frame(tp<-tp, Tetra <- log(Cushings[1:21,1]),
Pregna <- log(Cushings[1:21,2]))
attach(Cf)
Tetra <- Tetra- mean(Tetra) ; Pregna <- Pregna -mean(Pregna)
```

this improves the parametrisation, making convergence of maximisation algorithm faster.

```
cush.multinom <- multinom(tp ~ Tetra + Pregna, Hess =  T, maxit =  250)
cush.multinom
Call:
multinom(formula =  tp ~ Tetra + Pregna, Hess =  T, maxit =  250)

Coefficients:
  (Intercept)    Tetra    Pregna
b    7.288130 14.39930 -0.244936
c    2.385204 16.26469  3.358042

Residual Deviance: 12.30232
AIC: 24.30232
```

Note that the residual deviance is not an absolute measure of goodness of fit. In fact, the parameters are estimated rather imprecisely, as we see from

```
summary(cush.multinom)
Call:
multinom(formula =  tp ~ Tetra + Pregna, Hess = T, maxit = 250)

Coefficients:
  (Intercept)    Tetra    Pregna
b    7.288130 14.39930 -0.244936
c    2.385204 16.26469  3.358042

Std. Errors:
  (Intercept)    Tetra    Pregna
b    7.755119 13.73160 0.6692837
c    8.276217 13.38103 2.0996099

Residual Deviance: 12.30232
AIC: 24.30232
.......................................

round(predict(cush.multinom, type= "probs"),3)
      a    b    c
 1 0.89 0.01 0.10
 2 0.99 0.01 0.00
 3 1.00 0.00 0.00
 4 0.50 0.50 0.00
 5 0.43 0.56 0.00
 6 1.00 0.00 0.00
 7 0.00 0.99 0.01
 8 0.60 0.40 0.00
 9 0.58 0.42 0.00
10 0.00 0.99 0.01
11 0.00 1.00 0.00
12 0.00 0.29 0.71
13 0.00 0.97 0.03
14 0.00 1.00 0.00
15 0.01 0.99 0.00
16 0.00 0.91 0.09
17 0.00 0.10 0.90
18 0.00 0.06 0.94
19 0.00 0.63 0.37
20 0.00 0.13 0.87
21 0.00 0.03 0.97
```

The above shows that there is considerable uncertainty about the predicted class for some of the observations, eg numbers 8, 9.

```
predict(cush.multinom)
 [1] a a a a b a b a a b b c b b b b c c b c c
table(predict(cush.multinom),tp)
```

```
  a b c
a 5 2 0
b 1 7 1
c 0 1 4
```

which shows that the 'confusion matrix' is not so bad as we might have expected: the total of the offdiagonal terms is 5, so that the misclassification error rate with this method is 5/21, ie .24.
For this dataset, the logistic multinomial regression is actually less successful in prediction than the simple classification tree, which we can easily obtain as follows.

```
> first.tree <- tree(tp ~ Tetra + Pregna) # use rpart() if in R
> first.tree
node), split, n, deviance, yval, (yprob)
      * denotes terminal node

1) root 21 44.220 b ( 0.2857 0.4762 0.2381 )
  2) Tetra<-0.323364 8  8.997 a ( 0.7500 0.2500 0.0000 ) *
  3) Tetra>-0.323364 13 17.320 b ( 0.0000 0.6154 0.3846 )
    6) Pregna<0.582761 7  0.000 b ( 0.0000 1.0000 0.0000 ) *
    7) Pregna>0.582761 6  5.407 c ( 0.0000 0.1667 0.8333 ) *

> summary(first.tree)
Classification tree:
tree(formula = tp ~ Tetra + Pregna)
Number of terminal nodes:  3
Residual mean deviance:  0.8002 = 14.4 / 18
Misclassification error rate: 0.1429 = 3 / 21
```

What this is telling us is the following.
If you know neither Tetra nor Pregna, then you should predict all 21 cases to be 'b'.
But, this is not a 'terminal node' (in fact it is the root node), and we can improve our prediction. Our next step is
now look at Tetra, there are 8 cases for which Tetra < -0.323364, and all these cases should be predicted as 'a',
The remaining 13 cases have Tetra > -0.323364, and if you are allowed no further information, then predict all these cases as 'b'.
But this also is not a 'terminal' node: you can improve things further by
looking at Pregna for these 13 cases.
The 7 cases for whom Pregna < 0.582761 should be predicted as 'b' (this will be perfectly correct, and so must be a terminal node).
The remaining 6 cases for whom Pregna > 0.582761 should be predicted as 'c' (this will be not quite correct, but is a terminal node nonetheless).
You can check that this classification tree is then incorrect in exactly 3 out of the 21 cases, so the overall error rate is 0.1429.
I haven't given you the story here about the deviance, but that's something you can work out for yourself. The root deviance is easily seen to be

$$44.220 = -2n \sum p_i \log(p_i)$$

where $n = 21$, and $p_1 = 6/21, p_2 = 10/21, p_3 = 5/21$.

```
> post.tree(first.tree, file="tree.pdf", pointsize=6) # for a 'pretty' plot
```

We show how the sample space is divided up by the following plot, given as Figure 15.1.

```
> plot(Tetra, Pregna, type="n") # blank plot so far
> text(Tetra, Pregna,c("a","b","c")[tp]) # putting the points on with their labels
> abline(v= - 0.323364) # for the vertical dividing line
> abline(h=   0.582761) # for the horizontal dividing line
```

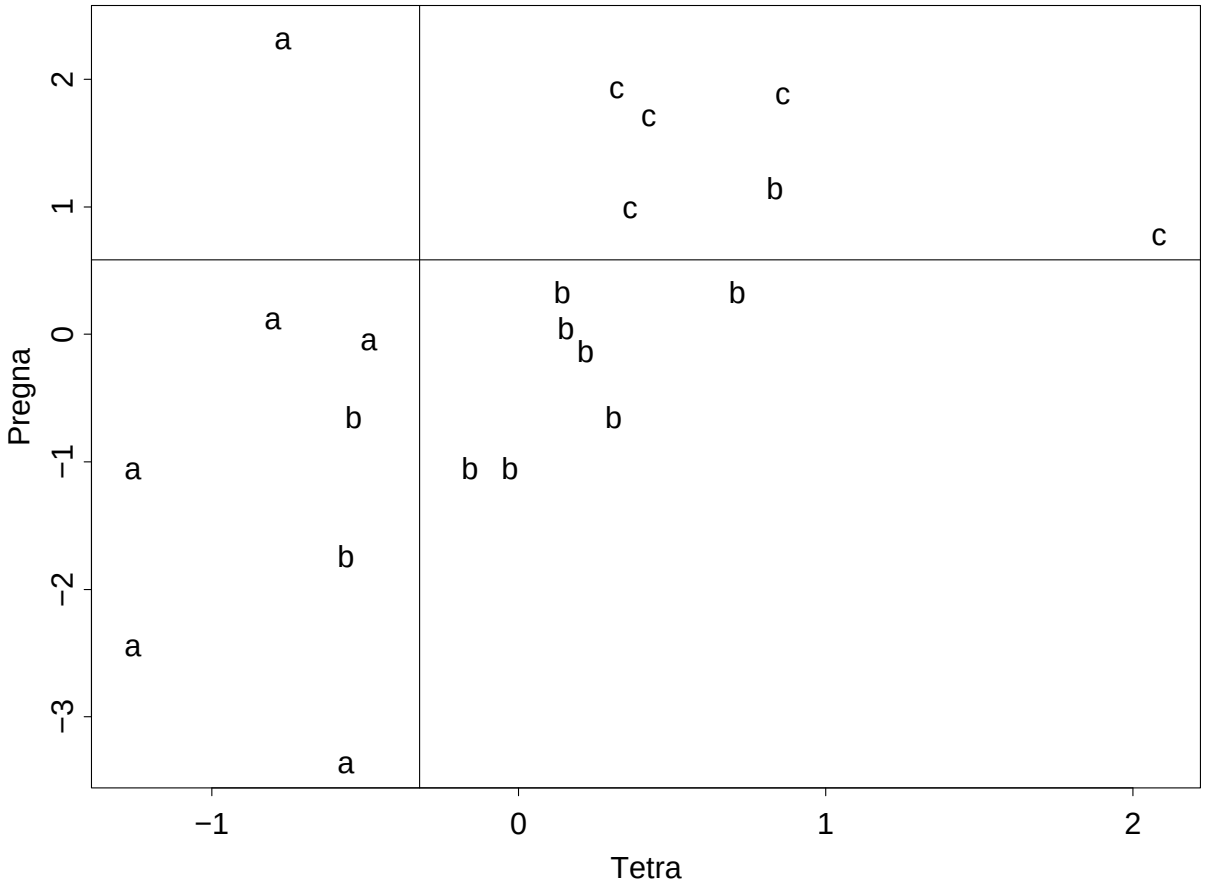


Figure 15.1: How the classification tree for the Cushings data is constructed

New for August 2008: Olympic medals data
The Independent, 6 August 2008, presents the dataset below on 'British medal hauls at the past 10 Olympics'.

```
           Gold Silver Bronze
Athens2004  9    9     12
Sydney2000 11   10      7
Atlanta96   1    8      6
Barcelona92 5    3     12
Seoul88     5   10      9
LosAngeles84 5  11     21
Moscow80    5    7      9
Montreal76  3    5      5
Munich72    4    5      9
MexicoCity68 5   5      3
```

Suppose we wish to find whether the distribution of medals into Gold, Silver, Bronze has changed over these 10 Games. Specifically we will fit

$$\log(p_{2i}/p_{1i}) = \alpha_2 + \beta_2 i, \text{ and } \log(p_{3i}/p_{1i}) = \alpha_3 + \beta_3 i,$$

for $i = 10, 9, \ldots, 1$ (the first row corresponding to 2004) with p_{1i}, p_{2i}, p_{3i} being the respective probabilities that in a given year, a medal is Gold, Silver, Bronze. ($p_{1i} + p_{2i} + p_{3i} = 1$, of course.) If you apply the analysis below, you will see that $\beta_2 = 0, \beta_3 = 0$ and that for the British teams, the probabilities of Gold, Silver, Bronze, respectively in any given year have remained more or less constant at $.24, .33, .42$.
Suggestion for analysis:

```
library(MASS) ; library(nnet)
Olympics <- read.table("Olympics.data", header=T)
Year <- 10:1 ; attach(Olympics) ; Olmat <- cbind(Gold, Silver, Bronze)
chisq.test(Olmat)
```

Strangely, the chisq statistic is 19.8 on 18 df, so really there's not a lot more to be said, but we will press on with the more complex multinomial logistic model as an exercise.

```
par(mfrow=c(1,2))
plot(Silver/Gold ~ Year) ; plot(Bronze/Gold ~ Year)
# these plots show no obvious trends
Total <- Gold + Silver + Bronze
first.multinom <- multinom(Olmat~ Year, Hess=T)
summary(first.multinom)
eigen(first.multinom$Hess) # to check Hessian is positive-definite
Olp <- predict(first.multinom, type="probs"); round(Olp,2) # for fitted probabilities
Olp <-  Olp*Total ; Olp<- round(Olp,2) # for fitted frequencies
cbind(Olp, Olmat) # for comparison
base.multinom <- multinom(Olmat ~1, Hess = T) # baseline model
#in which probabilities do not change with year
round(predict(base.multinom, type="probs"),2)
```

The resulting graphs are shown as Figure 15.2.
Note that there is a perceptible increase in the **Total** number of medals gained by Great Britain since 1968. This must be due in part to the increase in the number of Olympic events over the years; there were 172 events in 1968, and in 2008 there will be a total of 302 events. Try

```
plot(Total ~ Year)
```

But we see that Los Angeles 1984, in which there was a Total of 37 medals, was a 'strange' year, and in fact that was the Olympic Games which was boycotted by nearly all the Eastern Bloc countries. For this reason we now try

```
first.glm <- glm(Total[-6] ~ Year[-6], poisson) # to omit Los Angeles 1984
summary(first.glm) # shows a residual deviance of 6.83 on 7 df, hence a good fit
YYear <- 11:1
fv = exp(2.58660 +(0.07147*YYear))
plot(fv ~ YYEAR, type="l") # for fitted values, including for 2008
```

This gives
a predicted Total of 29.2 medals in 2008 (can you work out a confidence interval?), of which we expect
$7.1, 9.7, 12.4$ as Gold, Silver, Bronze respectively.
Contrast this with the rather upbeat prediction given before the start of the 2008 games by Nick Harris in The Independent. He predicted
$16, 17, 26$ as Gold, Silver, Bronze respectively.
As of August 19, 2008, it looks as though the actual outcome will be much better than my predictions of $7.1, 9.7, 12.4$**!**

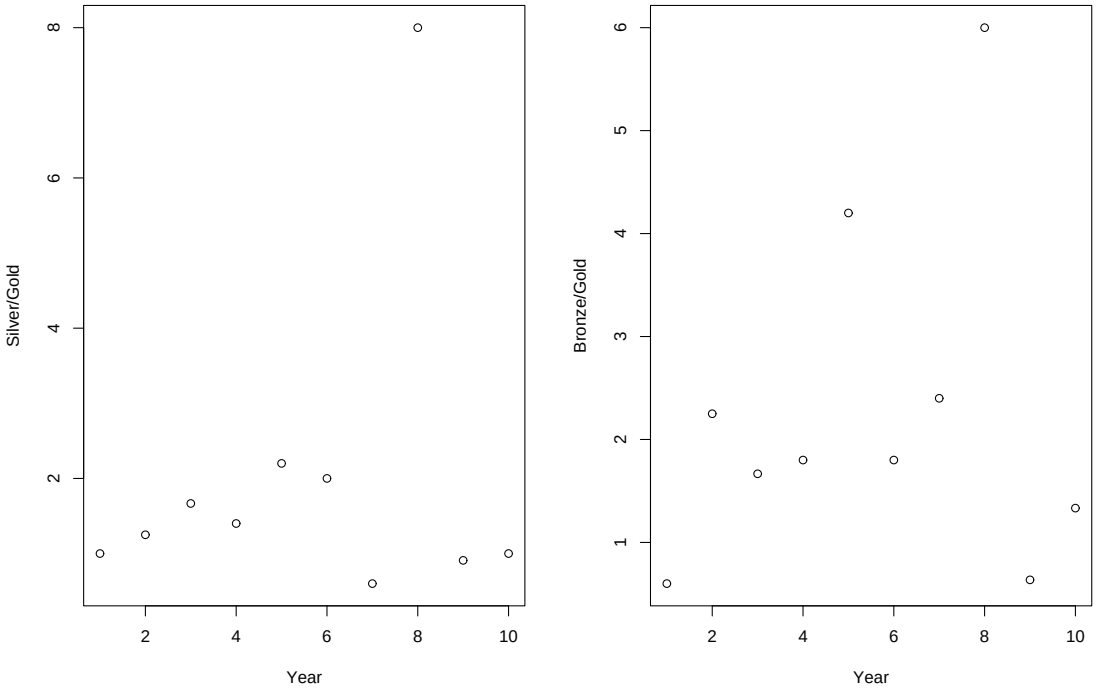


Figure 15.2: Have the British medals distributions changed from 1968 to 2004?

Chapter 16

Mohammad Raza's multivariate dataset on 50 famous films

Here is an interesting new dataset with which you can experiment, as collected by collected by Mohammad Raza of Wolfson College, Cambridge, for his May 2003 Mathematical Tripos Part III essay 'Analysis of a large and complex data set'. I am very grateful to Mohammad for his permission to use the data.
We have data for 50 famous recent movies, compiled via the Internet Movie Database (IMDB).
KEY to the variables
year= year when the film was released
gross= amount of money in millions of US dollars, the film made, in the US
budget= amount in millions of $ US spent in making the film,
(note that the 2 monetary figures given above are not 'adjusted' in any way, eg for inflation.)
rating (also male rating & female rating)= 'opinion of registered users of the IMDB website'. Each individual gives an integer score between 1 ('awful') and 10 ('excellent') for a given film. The rating for the film is then calculated as the average score. The 'male rating & female ratings' correspond to scores given by men, women, respectively.
AWARDS:
These are AA, GG, BAFTA for Academy Award, Golden Globe Award and British Academy of Film & Television Arts Awards, with a 1 indicating that the film won the 'Best Picture' award, and a 0 indicating that it did not.
GENRE:
Each movie is given a 1 or a 0 to indicate whether it was in a particular 'genre', eg comedy, scifi .... A film can be in more than one such genre.

	year	gross	budget	rating	malerating	femalerating
Titanic	1997	600.743	200.000	7.0	6.8	7.2
StarWars	1977	460.936	11.000	8.8	8.8	8.6
ET	1982	434.949	10.500	7.8	7.8	8.1
SWPhantomMenace	1999	431.065	115.000	6.7	6.7	6.5
SpiderMan	2002	403.706	139.000	7.7	7.7	7.8
JurassicPark	1993	356.763	63.000	7.3	7.3	7.1
ForrestGump	1994	329.452	55.000	8.0	8.0	8.0

```
                        HarryPotterI 2001 317.558 130.000    7.3        7.2          8.0
                      LOTRFellowship 2001 313.364 109.000    8.9        8.9          9.0
                         TheLionKing 1994 312.775  79.300    7.6        7.5          7.9
                         TheGodfather 1972 134.821   6.000    9.0        9.1          8.6
              TheShawshankRedemption 1994  28.341  25.000    9.0        9.0          9.0
                      TheGodfatherII 1974  57.300  13.000    8.9        8.9          8.4
                      SchindlersList 1993  96.067  25.000    8.8        8.8          8.9
                  ShichininNoSamurai 1954   0.187   0.500    8.9        9.0          7.7
                             Memento 2000  25.530   5.000    8.8        8.8          8.6
                        DrStrangelove 1964   9.164   1.800    8.7        8.8          8.1
                   SWReturnOfTheJedi 1983 309.064  32.500    8.1        8.1          8.0
                     IndependenceDay 1996 306.200  75.000    6.0        5.9          6.5
                      TheSixthSense 1999 293.502  55.000    8.3        8.2          8.4
                 SWEmpireStrikesBack 1980 290.159  18.000    8.7        8.7          8.4
                           HomeAlone 1990 285.761  15.000    6.2        6.2          6.7
                               Shrek 2001 267.652  60.000    8.1        8.1          8.3
          HowTheGrinchStoleChristmas 2000 260.031 123.000    6.0        5.9          6.4
                               Jaws 1975 260.000  12.000    8.2        8.3          7.9
           OneFlewOverTheCuckoosNest 1975 112.000   3.000    8.7        8.7          8.5
                          RearWindow 1954   1.559   1.000    8.7        8.7          8.6
                 RaidersOfTheLostArk 1981 242.374  20.000    8.6        8.7          8.4
                    TheUsualSuspects 1995  23.272   6.000    8.7        8.7          8.7
                    NorthByNorthwest 1959  13.275   4.000    8.6        8.7          8.5
                         PulpFiction 1994 107.930   8.000    8.6        8.7          7.9
                               Psycho 1960  32.000   0.800    8.6        8.6          8.3
               TheSilenceOfTheLambs 1991 130.727  22.000    8.5        8.6          8.5
                   LawrenceOfArabia 1962   0.342  12.000    8.6        8.6          8.4
                       Monsters,Inc 2001 255.870 115.000    8.1        8.0          8.4
                             Batman 1989 251.189  35.000    7.3        7.3          7.1
                          MenInBlack 1997 250.148  90.000    6.8        6.8          7.0
                           ToyStory2 1999 245.823  90.000    8.2        8.2          8.3
                             Twister 1996 241.700  92.000    5.9        5.8          6.3
                          GhostBusters 1984 238.600  30.000    7.4        7.4          7.5
                    BeverlyHillsCop 1984 234.760  15.000    7.1        7.1          7.1
                           CastAway 2000 233.630  90.000    7.3        7.3          7.4
          TheLostWorldJurassicPark 1997 229.074  73.000    5.4        5.4          5.2
                     AmericanBeauty 1999 130.058  15.000    8.5        8.5          8.3
                         Goodfellas 1990  46.836  25.000    8.5        8.6          8.0
                            Vertigo 1958   3.200   2.479    8.5        8.6          8.3
                       ApocalypseNow 1979  78.800  31.500    8.5        8.5          8.0
                           TheMatrix 1999 171.383  63.000    8.5        8.5          8.4
                          TaxiDriver 1976  21.100   1.300    8.4        8.5          7.9
                       SomeLikeItHot 1959  25.000   3.500    8.5        8.5          8.6

                              AA GG BAFTA comedy drama action horror fantasy
                          Titanic  1  1     1      0     1      0      0       0
                          StarWars  1  1     1      0     0      1      0       1
```

ET	1	1	1	0	0	0	0	1
SWPhantomMenace	0	0	0	0	0	1	0	0
SpiderMan	0	0	0	0	0	1	0	1
JurassicPark	0	0	0	0	0	1	1	0
ForrestGump	1	1	1	1	1	0	0	0
HarryPotterI	0	0	0	0	0	0	0	1
LOTRFellowship	1	1	1	0	0	0	0	1
TheLionKing	0	0	1	0	0	0	0	0
TheGodfather	1	0	1	0	1	0	0	0
TheShawshankRedemption	1	0	0	0	1	0	0	0
TheGodfatherII	1	0	1	0	1	0	0	0
SchindlersList	1	1	1	0	1	0	0	0
ShichininNoSamurai	0	1	0	0	1	1	0	0
Memento	0	0	0	0	1	0	0	0
DrStrangelove	1	1	0	1	0	0	0	0
SWReturnOfTheJedi	0	0	0	0	0	1	0	1
IndependenceDay	0	0	0	0	0	1	0	0
TheSixthSense	1	1	0	0	1	0	1	0
SWEmpireStrikesBack	0	0	0	0	0	1	0	1
HomeAlone	0	0	1	1	0	0	0	0
Shrek	0	1	1	1	0	0	0	1
HowTheGrinchStoleChristmas	0	0	0	1	0	0	0	1
Jaws	1	1	1	0	0	1	1	0
OneFlewOverTheCuckoosNest	1	1	1	0	1	0	0	0
RearWindow	0	1	0	0	0	0	0	0
RaidersOfTheLostArk	1	1	0	0	0	1	0	0
TheUsualSuspects	0	1	0	0	0	0	0	0
NorthByNorthwest	0	0	0	0	0	0	0	0
PulpFiction	1	1	1	0	1	1	0	0
Psycho	0	0	0	0	0	0	1	0
TheSilenceOfTheLambs	1	1	1	0	0	0	1	0
LawrenceOfArabia	1	1	1	0	1	0	0	0
Monsters,Inc	0	0	0	1	0	0	0	1
Batman	0	0	0	0	0	1	0	1
MenInBlack	0	0	1	1	0	1	0	0
ToyStory2	0	0	1	1	0	0	0	1
Twister	0	0	0	0	0	1	0	0
GhostBusters	0	0	1	1	0	0	0	1
BeverlyHillsCop	0	0	1	1	0	1	0	0
CastAway	0	0	0	0	1	0	0	0
TheLostWorldJurassicPark	0	0	0	0	0	1	1	0
AmericanBeauty	1	1	1	0	1	0	0	0
Goodfellas	1	1	1	0	1	0	0	0
Vertigo	0	0	0	0	1	0	0	0
ApocalypseNow	1	1	1	0	1	1	0	0
TheMatrix	0	0	0	0	0	1	0	0
TaxiDriver	1	1	0	0	1	0	0	0

```
             SomeLikeItHot  0  1     1      1     0      0      0       0

                           scifi romance thriller animation
                  Titanic      0       1        0         0
                 StarWars      1       0        0         0
                       ET      1       0        0         0
            SWPhantomMenace      1       0        0         0
                 SpiderMan      1       0        0         0
               JurassicPark      1       0        1         0
                ForrestGump      0       0        0         0
                HarryPotterI     0       0        0         0
             LOTRFellowship      0       0        0         0
                TheLionKing      0       0        0         1
               TheGodfather      0       0        0         0
    TheShawshankRedemption       0       0        0         0
            TheGodfatherII       0       0        0         0
             SchindlersList      0       0        0         0
         ShichininNoSamurai      0       0        0         0
                   Memento       0       0        1         0
              DrStrangelove      1       0        0         0
          SWReturnOfTheJedi      1       0        0         0
            IndependenceDay      1       0        0         0
              TheSixthSense      0       0        1         0
       SWEmpireStrikesBack       1       0        0         0
                  HomeAlone      0       0        0         0
                      Shrek      0       1        0         1
HowTheGrinchStoleChristmas       0       0        0         0
                       Jaws      0       0        1         0
 OneFlewOverTheCuckoosNest       0       0        0         0
                 RearWindow      0       0        1         0
       RaidersOfTheLostArk       0       0        0         0
          TheUsualSuspects       0       0        1         0
           NorthByNorthwest      0       1        1         0
                PulpFiction      0       0        1         0
                     Psycho      0       0        1         0
      TheSilenceOfTheLambs       0       0        1         0
          LawrenceOfArabia       0       0        0         0
              Monsters,Inc       0       0        0         1
                    Batman       0       0        1         0
                MenInBlack       1       0        0         0
                 ToyStory2       0       0        0         1
                   Twister       0       0        1         0
              GhostBusters       1       0        0         0
            BeverlyHillsCop      0       0        0         0
                  CastAway       0       0        0         0
  TheLostWorldJurassicPark       1       0        1         0
             AmericanBeauty      0       0        0         0
```

Goodfellas	0	0	0	0
Vertigo	0	0	1	0
ApocalypseNow	0	0	0	0
TheMatrix	1	0	1	0
TaxiDriver	0	0	1	0
SomeLikeItHot	0	1	0	0

Chapter 17

Fun and Games for British Union leaders (2004), and the Monetary Policy Committee (2007)

The dataset below was given in The Independent on May 29, 2004, under the headline
'One out of order, all out of order'.
'This week's unseemly brawl (at a barbecue in Hampstead) by Aslef officials continued a rich tradition of union leaders' excesses.'
The columns in the Table below give the 'out of order rating' for the categories
1. Fisticuffs, 2. Big Dinners, 3. Champagne socialist, 4. Luxury Travel,
5. Beer (no sandwiches), 6. Colourful Language, 7. Expenses Enthusiast, 8. Gender issues.
(By the way, if English is not your first language, you may need to get someone to explain some of the above (euphemisms) to you.)
The ratings are given for the 8 union leaders listed below (I omit their surnames).

	1	2	3	4	5	6	7	8
Joe 'The Cherub'	0	0	3	2	0	0	0	0
John 'Big Boss'	0	0	0	0	0	1	1	2
Andy 'Chasse-Spleen'	0	3	1	0	0	0	1	0
Derek	2	0	0	0	0	1	0	0
Shaun	2	0	0	0	2	1	0	0
Roger 'The Dodger'	0	1	1	0	0	0	2	0
Bollinger Bob	0	1	3	0	0	1	1	1
Raucous railwayman	0	1	2	0	2	2	0	0

You could, for example, construct a matrix to show the dissimilarities between all pairs of the 8 union bosses listed above.
Moving to a much more respectable scenario, but with data of the same structure, The Independent on July 2, 2007, gave the following data set on the voting of the 9 members of the Monetary Policy Committee with respect to the UK interest rates, under the heading 'Hawks, doves and pigeons: who influences UK interest rates?'. There were 9 successive monthly meetings, the first being 4/5 October, 2006.

Here we denote 1 to mean 'votes for an increase (of 0.25%), 0 to mean 'votes for no change in interest rate' and -1 to mean 'votes for a decrease (of 0.25%)'.
(For the record, interest rate was initially 4.75%, and at the end of the 6/7June meeting was 5.5%.)
We illustrate the positions of the 9 members of the MPC firstly with hierarchical clustering, and then by classical scaling. You may disagree with my (default) choice of metric. The individuals are plotted via hierarchical clustering in Figure 17.1, and via classical scaling in Figure 17.2. This latter is less successful as there are 3 pairs of coincident points, reflecting the fact that there are 3 pairs of individuals who vote identically. You could experiment with 'jitter' to improve the look of this plot.

	4/5Oct	8/9Nov	6/7Dec	10/11Jan	7/8Feb	7/8Mar	4/5Apr	9/10May	6/7Jun
Blanchflower	0	0	0	0	0	-1	0	1	0
Besley	1	1	0	1	1	0	1	1	1
Sentance	1	1	0	1	1	0	1	1	1
King	0	1	0	1	0	0	0	1	1
Gieve	0	1	0	1	0	0	0	1	1
Tucker	0	1	0	0	0	0	0	1	0
Bean	0	1	0	0	0	0	0	1	0
Barker	0	1	0	1	0	0	0	1	0
Lomax	0	0	0	0	0	0	0	1	0

```
a <- read.table("MPCdata.July2", header=T)
MPCnames <- row.names(a)
a <- as.matrix(a)
d <- dist(a)
clust.MPC <- hclust(d)
postscript("MPC.ps")
plclust(clust.MPC, hang=0.1,labels=MPCnames,main="Monetary Policy Committee")
dev.off()
loc <- cmdscale(d) ; x <- loc[,1] ; y <- loc[,2]
plot(x,y, type="n", main = "cmdscale for Monetary Policy Committee")
text(x,y, MPCnames, cex=1 )
```

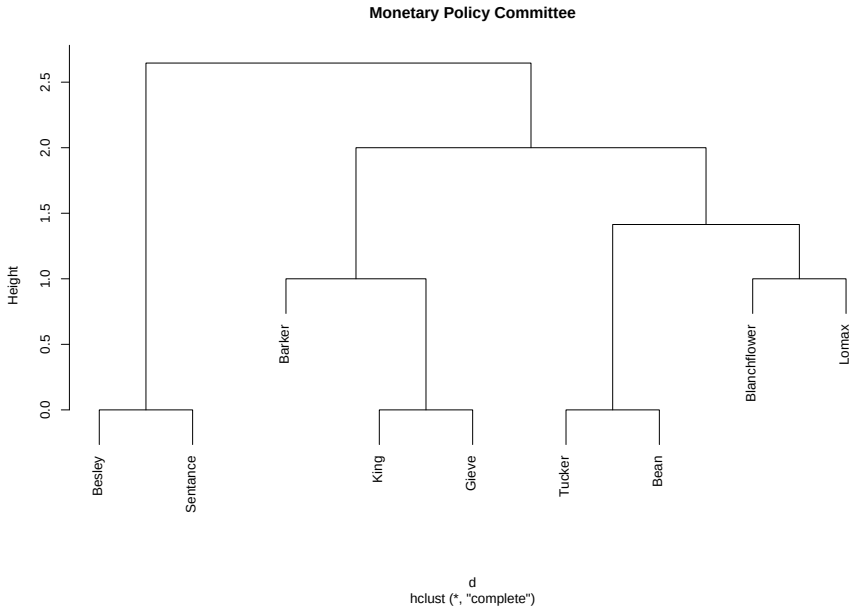


Figure 17.1: The monetary policy committee, hierarchical clustering

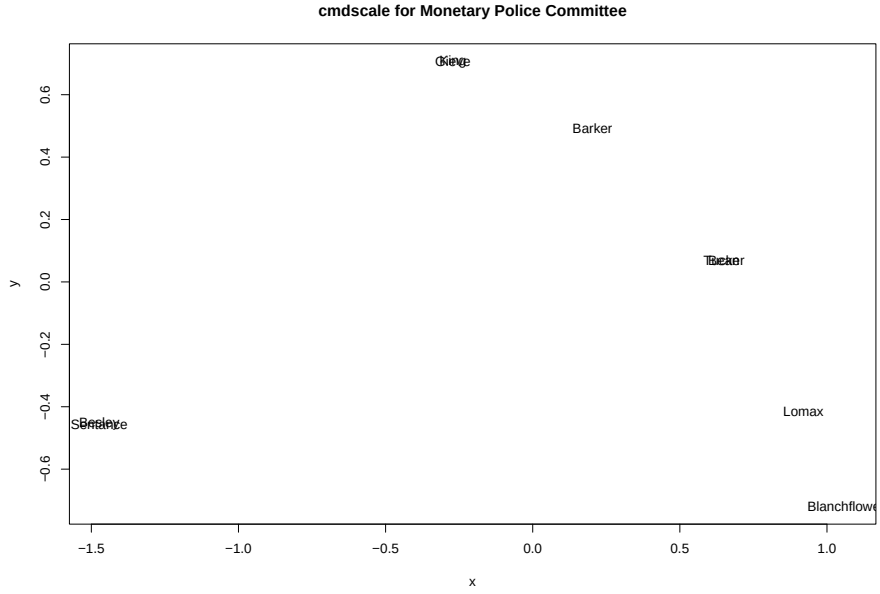


Figure 17.2: The monetary policy committee, classical multidimensional scaling

Chapter 18

Using capture-recapture data to estimate a total population size

Alan Agresti, in his 1994 *Biometrics* article
'Simple capture-recapture models permitting unequal catchability and variable sampling effort', Vol 50, pp 494-500, (and also in his 2002 book) gives a table of counts, first discussed by Cormack in 1985. This dataset was obtained from the results of a Capture-Recapture study of Snowshoe Hares, and consists of a 2^6 contingency table, with one missing entry, on the numbers of hares in a closed population which were trapped on each of 6 successive trapping days.
We need to set up suitable notation in order to describe the data precisely, thus
let $a = 0$ if an animal is NOT captured on the first day, and let $a = 1$ if it was captured on the first day.
Define $b = 0, 1, ..., f = 0, 1$ for the remaining sequence of 5 days.
The sequence of 64 entries in the variate 'count' follows the pattern

```
a= 0 1 0 1 0 1.................    0 1
b= 0 0 1 1 0 0 1 1 ............    1 1
c= 0 0 0 0 1 1 1 1 0 ...
d= 0 0 0 0 0 0 0 0 1 1  .....
e= 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 ..
f=  32 0's followed by 32 1's
```

We can set up this nested pattern of 0's and 1's using expand.grid() as shown below.

```
count <- scan()
NA 3 6 0 5 1 0 0
3  2 3 0 0 1 0 0
4  2 3 1 0 1 0 0
1  0 0 0 0 0 0 0
4  1 1 1 2 0 2 0
4  0 3 0 1 0 2 0
2  0 1 0 1 0 1 0
1  1 1 0 0 0 1 2
# data from Agresti (2002) p512
```

Thus you see that we don't know the number NOT caught on any of the 6 days, and for example, 6 animals were caught on day 2, but not on any of the other 5 days. (There were 2 wretched creatures who were caught on every one of the 6 days.)

```
x <- expand.grid(a=0:1, b=0:1, c=0:1,d=0:1, e=0:1,f=0:1)
x[1:10,] # as a check
attach(x)
sum(count[2:64])
```

This shows that a total of 68 animals were seen at least once each.
Our aim is to fit a model to this table of $2^6 - 1$ counts, in order to estimate the number of hares in this (closed) population which were never seen at all: this enable us to estimate the total population size.

```
A <- factor(a) ; B <- factor(b); C <- factor(c) ; D <- factor(d); E<- factor(e)
F<- factor(f)
```

First we fit a model which assumes mutual independence of the 6 catching occasions, but which does not assume equal catchability. (The output has been slightly reduced.)

```
> first.glm <- glm(count~ A+B+C+D+E+F, poisson)
> summary(first.glm)
 ......................

    Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.9553     0.2891   6.765 1.34e-11 ***
A1           -1.3061     0.2875  -4.543 5.55e-06 ***
B1           -0.5194     0.2491  -2.085 0.037052 *
C1           -1.0128     0.2681  -3.778 0.000158 ***
D1           -0.6351     0.2520  -2.520 0.011735 *
E1           -0.8170     0.2585  -3.160 0.001575 **
F1           -0.2970     0.2460  -1.207 0.227357

    Null deviance: 112.846  on 62  degrees of freedom
    Residual deviance:  58.314  on 56  degrees of freedom
    AIC: 154.50

> exp(1.9553)
[1] 7.066039
```

(You can see that a total of 16 animals were caught on day 1, compared with a total of 32 animals caught on day 6: perhaps the animals were getting tired, and/or the trappers were getting better at their task.)
The Residual deviance of 58.314 on 56 degrees of freedom shows us that the model does not fit very well, but we will still use the estimate of the intercept to provide us with an estimate of the count for which $a = 0,\ b = 0, \ldots f = 0$, giving us a value of $exp(1.9553) = 7.066039$, and hence an estimate of $68 + 7.1 = 75.1$ as our estimate of N, the total population size. (We could use $68 + exp(1.9553 + / - 2 * 0.2891)$ to give us our confidence interval for N.)
Agresti (1994) discusses various models which might fit better than the simple model of mutual

independence, and therefore which might provide more accurate estimates of N. For simplicity here, we discuss only one generalization of the independence model: namely the model which allows all 2-factor interactions between $A, B, \ldots, F$. There are 15 such interactions, each with 1 df. We again edit the resulting output somewhat.

```
next.glm <- glm(count~ (A+B+C+D+E+F)^2 , poisson)
summary(next.glm)

    Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.60498    0.51850   6.953 3.58e-12 ***
A1          -2.28082    0.57134  -3.992 6.55e-05 ***
B1          -1.82090    0.51966  -3.504 0.000458 ***
C1          -2.31716    0.55609  -4.167 3.09e-05 ***
D1          -2.14780    0.54364  -3.951 7.79e-05 ***
E1          -2.06819    0.53846  -3.841 0.000123 ***
F1          -2.09074    0.54177  -3.859 0.000114 ***
.....................
A1:E1        1.46053    0.66617   2.192 0.028348 *
.....................
C1:F1        1.59899    0.64082   2.495 0.012588 *
D1:F1        1.79685    0.60052   2.992 0.002770 **

    Null deviance: 112.846  on 62  degrees of freedom
    Residual deviance:  32.424  on 41  degrees of freedom
    AIC: 158.61

> exp(3.60498)
[1] 36.78095
```

Hence this more complex model (for which in fact only 3 of the 15 interactions are significant) gives us a point estimate of N as $36.8 + 68 = 104.8$.

Comment: I try to illustrate each of these worksheets with an appropriate graph. I haven't yet worked out what would be a suitable graph for this particular problem, so that is a challenge for you!

The dataset discussed above is by now rather old (though a classic, no doubt). Agresti's 1994 paper gave a GLIM program for the log-linear analysis, discussed a variety of possible models (including latent class models) for this dataset. If you want a new dataset as a challenge, try the following, taken from 'Capture-recapture methods to size alcohol-related problems in a population' by Corrao, Bagnardi, Vittadini and Favilli, *J.Epidemiol. Community Health* 2000;54;603-610.

Our object is to estimate the total number of individuals with alcohol-related problems (ARP) in the target population, by combining data from 4 different (and clearly non-independent) 'flagging' sources.

Here is the table of data, as published on p 606 by Corrao et al. (You may need to think a bit in order to put it into R/S-Plus.)

```
Gender   M   M  F   F
```

Age	Y	O	Y	O	Total
Patients flagged by exactly 1 source					
F1	30	40	6	12	88
F2	31	12	7	7	57
F3	12	5	3	2	22
F4	46	81	12	16	155
Patients flagged by exactly 2 sources					
F1&F2	2	1	0	0	3
F1&F3	1	0	0	0	1
F1&F4	1	2	0	0	3
F2&F3	2	2	0	1	5
F2&F4	3	3	1	0	7
F3&F4	3	1	0	0	4
Patients flagged by exactly 3 sources					
F1&2&3	0	0	0	0	0
F1&2&4	0	0	0	0	0
F1&3&4	1	0	0	0	1
F2&3&4	1	0	1	0	2
Patients flagged by all 4 sources					
F1&2&3&4	1	0	0	0	1
Total	134	147	30	38	349

Key to above table
Gender, M = male, F =female
Age, Y = under 50 yrs, O = 50 yrs or older
The sources for 'flagging' the patients were
F1 = self-help volunteering groups (similar to Alcoholics Anonymous)
F2 = psychiatric ambulatory
F3 = Public Alcohology Service
F4 = hospital discharges.
The catchment area was 'all residents in the area of Voghera, a Northern Italy rural area with an economy based on vinegrowing and wine production... with a resident population of 132618 over 15 years in age.'
Corrao et al discuss various log-linear models, and conclude that the target population contained approximately 2500 individuals with ARP.
How does this compare with your estimate? What is your confidence interval? Do you have to treat the table for men differently from that for women?
(The authors conclude that the answer to this question is No, but that Young and Old should be analysed separately.)

Chapter 19

The Ibrahim index, and links to other multivariate datasets

The Ibrahim Index of African Governance assesses the 48 countries of sub-Saharan Africa according to each of the following 5 features
Safety and Security
Rule of Law, Transparency and Corruption,
Participation and Human Rights,
Sustainable Economic Opportunity,
Human Development.
These 5 scores are given as the first 5 numeric columns of the datamatrix below,and are followed by the Ibrahim index and its Rank (so you see that Mauritius ranks best, and Somalia ranks worst). We use this dataset as an opportunity for a 'stars' plot, so we obtain a plot of each of the 5 attributes for each of the 48 countries.
Obviously there are many other multivariate techniques you could usefully apply to this dataset.
The figures below give the Ibrahim Index for 2008 (published October 6, 2008) which is based on data from 2006.

	Safety	Law	HumRights	EconOpp	HumDev	IbIndex	Rank
Angola	82.0	38.4	29.0	32.9	34.4	43.3	44
Benin	86.1	52.3	81.1	36.9	56.2	62.5	13
Botswana	75.0	81.6	87.4	58.2	68.0	74.0	4
Burkina_Faso	86.1	56.5	70.1	30.3	48.4	58.3	20
Burundi	62.0	48.3	60.4	35.9	43.3	50.0	35
Cameroon	77.7	43.3	54.7	44.7	56.5	55.4	25
Cape_Verde	100.0	86.1	77.7	47.2	62.6	74.7	3
Central_African_Republic	46.7	42.7	60.2	34.9	33.2	43.6	43
Chad	51.5	40.1	29.8	25.6	22.5	33.9	46
Comoros	94.4	52.8	73.1	37.5	51.5	61.9	14
Congo	68.5	43.5	48.6	48.7	57.1	53.3	28
Congo,_Dem_Republic	52.8	24.3	14.7	26.3	30.7	29.8	47
Cote_dIvoire	75.2	36.0	22.6	42.9	51.6	45.6	42

Djibouti	86.0	43.6	55.6	36.8	54.0	55.2	26
Equatorial_Guinea	86.0	43.3	32.2	48.5	36.0	49.2	36
Eritrea	81.0	56.6	10.6	35.4	48.8	46.5	41
Ethiopia	72.8	47.9	40.4	40.5	52.7	50.9	31
Gabon	100.0	56.4	61.2	61.6	67.8	69.4	8
Gambia	86.0	54.7	42.5	40.5	52.2	55.2	27
Ghana	86.1	72.7	80.2	47.3	64.3	70.1	7
Guinea	80.3	51.0	25.4	32.3	50.0	47.8	40
Guinea-Bissau	80.5	34.6	75.2	23.3	45.8	51.9	30
Kenya	63.3	56.0	63.3	48.6	64.5	59.1	17
Lesotho	75.0	69.3	75.5	42.9	53.7	63.3	12
Liberia	58.8	26.8	87.9	36.9	33.2	48.7	38
Madagascar	86.1	57.3	74.9	39.4	44.3	60.4	16
Malawi	86.1	64.0	69.1	40.7	59.7	63.9	11
Mali	77.8	50.0	74.7	31.4	45.8	55.9	23
Mauritania	71.0	58.8	30.8	36.8	56.5	50.8	32
Mauritius	91.7	80.5	92.2	71.4	89.9	85.1	1
Mozambique	86.1	50.4	70.4	36.7	41.7	57.1	22
Namibia	83.3	76.7	75.3	57.4	61.6	70.9	6
Niger	86.1	51.7	79.4	27.6	32.5	55.5	24
Nigeria	63.7	48.2	44.1	40.7	45.9	48.5	39
Rwanda	98.4	46.0	69.5	37.7	43.6	59.0	18
Sao_Tome&Principe	100.0	55.6	83.4	41.5	61.1	68.3	9
Senegal	85.4	66.2	81.7	42.3	54.9	66.1	10
Seychelles	83.2	80.4	76.9	70.0	88.4	79.8	2
Sierra_Leone	79.6	37.3	69.8	27.1	31.8	49.1	37
Somalia	38.8	8.2	6.4	26.0	15.2	18.9	48
South_Africa	61.1	78.1	86.3	63.5	68.7	71.5	5
Sudan	29.0	29.8	12.0	42.2	58.0	34.2	45
Swaziland	69.4	56.9	28.8	46.5	49.2	50.2	34
Tanzania	83.3	59.6	65.4	43.4	56.4	61.6	15
Togo	77.2	47.1	44.2	42.0	54.4	53.0	29
Uganda	75.1	55.8	61.0	42.2	57.4	58.3	19
Zambia	77.8	60.5	66.6	43.0	43.5	58.3	21
Zimbabwe	75.1	44.6	41.9	38.8	51.7	50.4	33

```
Africa = read.table("Ibrahim.data", header=T)
stars(Africa[, 1:5],full=FALSE,len = 0.6,key.loc = c(-1,10),
+ draw.segments = TRUE,nrow =8, cex = .8)
```

This results in the 'stars' plot shown as Figure 19.1. See how poor little Somalia is an almost invisible dot, while Mauritius and the Seychelles are large and visible, reflecting their relative attractiveness as places to live.

New for 2009: CO_2 **emissions data**
Here is another topical data set for you to play with.

Angola
Benin
Botswana
Burkina_Faso
Burundi
Cameroon
Cape_Verde
Central_African_Republic
Chad
Comoros
Congo
Congo,_Dem_Republic
Cote_dIvoire
Djibouti
Equatorial_Guinea
Eritrea
Ethiopia
Gabon
Gambia
Ghana
Guinea
Guinea−Bissau
Kenya
Lesotho
Liberia
Madagascar
Malawi
Mali
Mauritania
Mauritius
Mozambique
Namibia
Niger
Nigeria
Rwanda
Sao_Tome&Principe
Senegal
Seychelles
Sierra_Leone
Somalia
South_Africa
Sudan
Swaziland
Tanzania
Togo
Uganda
Zambia
Zimbabwe
EconOpp
HumDev
HumRights
Law
Safety

The Independent, under the headline 'The challenge facing the world's biggest polluters' on March 11, 2009, gave the following data, in an article beginning
'The clock is ticking in the race to agree a new treaty to cut the emissions that cause global warming. Michael McCarthy names and shames the offenders who must mend their ways.' The dataset consists of 'the 20 worst offenders' in terms of CO_2 emissions, which form the first column of numbers in million tonnes, presumably per annum.
The GDP is given in trillions of dollars,
GDPpc = GDP per capita, given in dollars,
EcG = Economic growth rate, as a percentage,
Pop = Population, in millions,
PopGr = Population growth rate, as a percentage.

```
           CO2   GDP   GDPpc   EcGr  Pop  PopGr
China     6018   7.8    6100    9.8  1330   0.62
US        5903  14.8   48000    1.4   303   0.88
Russia    1704   2.22  15800    6.0   140  -0.47
India     1293   3.32   2900    7.3  1140   1.57
Japan     1247   4.48  35300    0.7   127  -0.13
Germany    858   2.86  34800    1.7    82  -0.04
Canada     614   1.33  40200    0.7    33   0.83
UK         586   2.28  37400    1.1    61   0.27
S.Korea    515   1.27  26000    2.5    48   0.27
Iran       471   0.859 13100    6.6    65   0.79
Italy      468   1.8   31000    0.0    58  -0.02
S.Africa   444   0.506 10400    3.7    48   0.83
Mexico     436   1.57  14400    2.0   109   1.14
SaudiAr    424   0.600 21300    4.2    28   1.95
France     418   2.09   32700   0.7    62   0.57
Australia  417   0.825  39300   2.5    21   1.22
Brazil     377   2.03   10300   5.2   196   1.23
Spain      373   1.37   36500   1.3    40   0.10
Ukraine    329   0.360   6900   2.1    46  -0.65
Poland     303   0.684  17800   5.3    38  -0.04
```

New for 2012: The Happy Planet Index.
see `http://www.happyplanetindex.org` for a large multivariate dataset, easily downloadable. (use your desktop to reformat from a .xlsx file to the .csv file you want, in order to read it into R for appropriate analyses.)

New for 2012: Student Crime Statistics.
see `http://www.thecompleteuniversityguide.co.uk/student-crime-statistics/` for a 3-dimensional data set. (use your desktop to reformat from a .pdf file to the .txt file you can input into R, after some slight editing.)
Of the 3 crime rates (Violent, Burglary, Robbery) it appears that conditional on Robbery, the

other two are approximately independent. (Look at the pairwise conditional correlations, obtained by inverting the variance-covariance matrix.)

New for 2012: Olympic Triathlon results, men 2008.
see `http://www.sport.maths.org/content/olympic-triathlon`. I actually had to work a bit to read the dataset into R, since the times are displayed as 0:30:13.9, for example, ie hours:minutes:seconds. We have a 4-dimensional dataset, since the 'Total time' is the time for the 1.5km swim, plus the time for the 40 km cycle ride, plus the time for the 10 km run, PLUS the transition between events time (this is normally about a minute, we are told).
New for 2012: results from the London Olympics. There is a wealth of multivariate data available here, (although a given table is not always in the format most convenient for us, eg if we want to work out a correlation matrix).
Here is the link to a very simple example
`http://www.london2012.com/equestrian/event/eventing-individual/phase=eqx003400/index.html` Dressage is just one of three components in the equestrian eventing competition. Can you find the other two, and see how the three components are related to one another?
'Climate variability and conflict risk in East Africa, 1990–2009' by J.O'Loughlin and others, published in PNAS, 2012, see `http://www.pnas.org/content/109/45/18344.abstract` is a very substantial paper, illustrating both generalized additive modelling and negative binomial modelling. This paper has excellent online resources, including not only the full datasets, but also all the R programs used. To reproduce just a small part of the authors' calculations, showing full understanding of them, would make an excellent and challenging student project.

Chapter 20

A Bayesian analysis for matched pairs with incomplete data

'Using Bayesian p- values in a 2×2 table of matched pairs with incompletely classified data' by Lin, Lipsitz, Sinha, Gawande, Regenbogen and Greenberg appeared in *Appl. Statist.* (2009) **58**.
I was intrigued to see this paper, not least because it extends a result I published in 1971, specifically Altham, P.M.E. (1971) 'The analysis of matched proportions'. *Biometrika*, **58** 561-576.
Lin *et al.* give a numerical example, and show how to derive appropriate Bayesian p- values. However, they do not give computational details, except to say that a SAS macro may be obtained from the first author.
Naturally I wanted to see how we could do the computations in R. By happy chance Dr Robin Hankin gave a seminar in the Statistical Laboratory, University of Cambridge, on February 20, 2009, in which I realised that he had already developed a general method in R for dealing with his 'hyperdirichlet distributions' which would solve the computational problem here.
For convenience I present the practical example given in Lin *et al.*, and follow their notation. The dataset is given here as Table 20.1. These arise from 69 medical malpractice claims, and are the two Surgeon Reviewers' answers to the question: was there a communication breakdown in the hand-off between physicians caring for the patient? The rows of the Table correspond to the answers given by Reviewer 1, and the columns to the answers given by Reviewer 2.
As you will see, there were 69 cases in all, but in 11 cases where Reviewer 1 gave an answer, there was no answer given by Reviewer 2. Correspondingly in 8 cases where Reviewer 2 gave an answer, there was no answer given by Reviewer 1.
Following Lin *et al*, we adopt the notation given in Table 20.2 for the corresponding observed frequencies.

	Yes	No	Missing	Total
Yes	26	1	2	29
No	5	18	9	32
Missing	4	4	0	8
Total	35	23	11	69

Table 20.1: Two surgeon reviews of malpractice claims data

	Yes	No	Missing	Total
Yes	y_{11}	y_{10}	z_{1+}	$y_{1+} + z_{1+}$
No	y_{01}	y_{00}	z_{0+}	$y_{0+} + z_{0+}$
Missing	u_{+1}	u_{+0}	0	u_{++}
Total	$y_{+1} + u_{+1}$	$y_{+0} + u_{+0}$	z_{++}	n

Table 20.2: Notation for the data

What we want to know is whether Reviewer 2 is giving significantly higher proportion of 'Yes' responses than is Reviewer 1.
Note that it is easy to do the McNemar test on the 2×2 table of complete observations, thus:

```
>x <- matrix(c(26,1,5,18),nrow=2,byrow=T,
 dimnames=list("Reviewer 1"=c("Yes","No"),"Reviewer 2"= c("Yes", "No")))
> x
          Reviewer 2
Reviewer 1 Yes No
       Yes  26  1
       No    5 18

> mcnemar.test(x)

        McNemar's Chi-squared test with continuity correction

data:  x
McNemar's chi-squared = 1.5, df = 1, p-value = 0.2207
```

Not surprisingly (since this is just comparing 5 with 1) the answer is that there is no significant difference. We could easily do an exact test for this problem: it would be based on the Binomial distribution with parameters $6, 1/2$.
However, it is unsatisfactory not to make use of the observations in the 'Missing' row and column. To make use of these observations we note that, following equation (1) of Lin et al, the likelihood function of the data D may be taken as

$$L(\theta|D) \propto \prod_{ij} {\theta_{ij}}^{y_{ij}} \prod_{i} {\theta_{i+}}^{z_{i+}} \prod_{j} {\theta_{+j}}^{u_{+j}}$$

where $i = 0, 1$ and $j = 0, 1$.
If we take a Dirichlet prior for $(\theta_{ij}),\ i = 0, 1,\ j = 0, 1$, namely

$$\pi(\theta) \propto \prod_{ij} {\theta_{ij}}^{\alpha_{ij}-1},\ \text{with}\ \sum_{ij} \theta_{ij} = 1,$$

then the posterior density of θ given the data D is

$$\pi(\theta|D) \propto \prod_{ij} {\theta_{ij}}^{(y_{ij}+\alpha_{ij}-1)} \prod_{i} {\theta_{i+}}^{z_{i+}} \prod_{j} {\theta_{+j}}^{u_{+j}}.$$

We want to work out $Pr(\theta_{+1} > \theta_{1+}|D)$, equivalently we will find $Pr(\theta_{01} > \theta_{10}|D)$. We will do this for the special case of $\alpha_{ij} = 1$ for all i, j, which corresponds to the uniform prior density for (θ_{ij}).

The important thing to notice is that while the expression for $\pi(\theta|D)$ looks very awkward, it is in fact a special case of a hyperdirichlet, as defined by Robin Hankin. He has an R package which will manipulate this density, for example to find a given probability by integration. Here we seek the probability $Pr(\theta_{01} > \theta_{10}|D)$. To fit in with Robin's notation, it is a little easier if we rewrite the original 2×2 table of probabilities as Table 20.3.
In terms of (p_1, p_2, p_3, p_4) we seek the posterior probability $Pr(p_3 > p_2|D)$.

	Yes	No
Yes	$\theta_{11} = p_1$	$\theta_{10} = p_2$
No	$\theta_{01} = p_3$	$\theta_{00} = p_4$

Table 20.3: Rewriting the probabilities

The R program below first reads in all the necessary R packages (in the correct order), then set up the hyperdirichlet distribution, and finally does the required integration. The posterior density function is

$$\pi(p|D) \propto p_1^{26+1-1} p_2^{1+1-1} p_3^{5+1-1} p_4^{18+1-1} (p_1+p_2)^2 (p_3+p_4)^9 (p_1+p_3)^4 (p_2+p_4)^4$$

for $p_i \geq 0$, and $p_1 + p_2 + p_3 + p_4 = 1$.
Although the true Dirichlet distribution has many elegant analytic properties, it looks pretty hard to say anything elegant or analytic about this hyperdirichlet distribution. That's why we need R to do the computation.

```
library(mvtnorm)
library(adapt,lib.loc="~/Rlibs")
library(Brobdingnag,lib.loc="~/Rlibs")
library(polynom,lib.loc="~/Rlibs")
library(partitions,lib.loc="~/Rlibs")
library(aylmer,lib.loc="~/Rlibs")
library(abind, lib.loc="~/Rlibs")
library(hyperdirichlet,lib.loc="~/Rlibs")
# Note added June 2012. I realise that the above process (ie loading the necessary libraries,
# in the correct order, can be made much swifter, by the following two commands
# install.packages("hyperdirichlet", dependencies=TRUE)
# library(hyperdirichlet)

(b <- dirichlet(1:4)) # for useful starting point
     p1 p2 p3 p4 params powers
[1]   0  0  0  0      0      0
[2]   0  0  0  1      4      3
[3]   0  0  1  0      3      2
[4]   0  0  1  1      0      0
[5]   0  1  0  0      2      1
[6]   0  1  0  1      0      0
[7]   0  1  1  0      0      0
[8]   0  1  1  1      0      0
[9]   1  0  0  0      1      0
[10]  1  0  0  1      0      0
```

```
[11]  1  0  1  0      0      0
[12]  1  0  1  1      0      0
[13]  1  1  0  0      0      0
[14]  1  1  0  1      0      0
[15]  1  1  1  0      0      0
[16]  1  1  1  1      0      0

# Note that we are setting up the powers for a total
# of 16 possible combinations of p1, .., p4
#(with the restriction that the first and last rows
# have powers 0).

Normalizing constant:  3.30687830687831e-05
# Now edit b so that it corresponds to our data
> b[2] = 19; b[3]= 6; b[4] = 9; b[6] = 4
> b[9] = 27; b[11] = 4; b[13] = 2
> b
     p1 p2 p3 p4 params powers
[1]   0  0  0  0      0      0
[2]   0  0  0  1     19     18
[3]   0  0  1  0      6      5
[4]   0  0  1  1      9      9
[5]   0  1  0  0      2      1
[6]   0  1  0  1      4      4
[7]   0  1  1  0      0      0
[8]   0  1  1  1      0      0
[9]   1  0  0  0     27     26
[10]  1  0  0  1      0      0
[11]  1  0  1  0      4      4
[12]  1  0  1  1      0      0
[13]  1  1  0  0      2      2
[14]  1  1  0  1      0      0
[15]  1  1  1  0      0      0
[16]  1  1  1  1      0      0

> is.hyperdirichlet(b)
[1] TRUE
> x <- hyperdirichlet(b)
> f <- function(p){p[2]>p[3]}
> prob <-probability(x,disallowed=f)
# this evaluates prob(p[2]< p[3])
> prob
[1] 0.969555
> (probability(x,disallowed=f, eps=1e-3))
[1] 0.9689503 # for more accurate version

> max.like(b)
```

```
$MLE
        p1         p2         p3         p4
0.45121066 0.01799551 0.11125458 0.41953925
# We expect the response of Reviewer 1, 2 to be positively
# correlated, and indeed the mle's show (p1*p4)>> (p2*p3)
  ..................
```

Hence we see that $Pr(\theta_{01} > \theta_{10}|D) = 0.969$, ie Reviewer 2 is more likely to give a 'Yes' answer than is Reviewer 1.

Afterword i) In this instance, if we stop and think for a little, we see that it is not necessary to do 3-dimensional integration in order to find the normalization constant of the hyperdirichlet posterior density. For, using Hankin's notation for simplicity, we see that the posterior density is of the form

$$\pi(p|D) \propto p_1^a p_2^b p_3^c p_4^d (p_1+p_2)^A (p_3+p_4)^B (p_1+p_3)^C (p_2+p_4)^D$$

for suitably defined non-negative integers $a, b, c, d,\ A, B, C, D$. Now use 4 binomial expansions, eg

$$(p_1+p_2)^A = \sum_x \binom{A}{x} p_1^x p_2^{(A-x)},\ \ldots.$$

This gives

$$\pi(p|D) \propto \sum_{x,y,z,w} p_1^a p_2^b p_3^c p_4^d\ f(x,y,z,w)\ p_1^x p_2^{A-x} p_3^y p_4^{B-y} p_1^z p_3^{C-z} p_2^w p_4^{D-w}$$

where $x = 0, \ldots, A,\ y = 0, \ldots, B,\ z = 0, \ldots, C$, and $w = 0, \ldots, D$, $f(x,y,z,w)$ being a known product of binomial coefficients.
Now we can integrate over the simplex $p_i > 0$ for all i, $\sum p_i = 1$ and apply the general result

$$\Gamma(\nu) \int_p \prod_i p_i^{\nu_i - 1} dp = \prod_i \Gamma(\nu_i)$$

where $\nu = \sum \nu_i$, and hence find the normalisation constant.
Similarly, diligent integration of each term of the summation over the region ($p_i > 0$ for all i, $\sum p_i = 1$ and $p_2 < p_3$) should give us an exact formula for

$$Pr(p_2 < p_3|D)$$

in terms of sums of the beta-tails, ie binomial sums. For if θ has a beta distribution, then $Pr(\theta < 1/2)$ is a tail of the corresponding binomial distribution with parameter $1/2$.
This is shown by the following argument:
Suppose θ is beta, with parameters a, b, which are positive integers. To find $Pr(\theta < 1/2)$ as a binomial sum, you could integrate by parts. But a better way is to consider a random sample of $a+b-1$ points from the uniform distribution on $[0,1]$. Define θ as the distance from 0 to the ath point of the sample, then we see that θ has the beta distribution with parameters a, b. Further $Pr(\theta < 1/2) = Pr(N \geq a)$, where N is the number of points of the sample in the interval $[0, 1/2]$, hence $N \sim Bi(a+b-1, 1/2)$.

Afterword ii) For this example, it isn't very difficult to maximise the log-likelihood function explicitly, and hence find that $\log(\hat{p}_3/\hat{p}_2) = \log(0.11125458/0.01799551)$. From the matrix of second derivatives of the log-likelihood function, we can also find the approximate standard error of $\log(\hat{p}_3/\hat{p}_2)$. You see below a simple way of doing this in R.

```
a =  c(26,1,5,18)
A = 2; B = 9 ; C = 4; D = 4
x0 = rep(0.1, times=3) # starting values
lhyper = function(p)
+ {
+ p[4] = 1-p[1]-p[2]-p[3]
-sum(a*log(p)) - A* log(p[1] + p[2]) - B*log(p[3] + p[4])
 - C*log(p[1] + p[3]) - D*log(p[2] + p[4])
+ }
# The hardest thing here is to remember the MINUS signs!!
 fit.dir =optim(x0,lhyper, method="BFGS", hessian=T)
fit.dir$par   # gets the max likelihood  estimates
0.45118225 0.01800136 0.11127152
fit.dir$hessian
 V = solve(fit.dir$hessian) # for covariance matrix
eigen(V) # to check positive definite
 phat = fit.dir$par
# We now find the approx variance of log(phat[3]/phat[2])
> V[2,2]/(phat[2]^2) - 2*V[2,3]/(phat[2]*phat[3]) + V[3,3]/(phat[3]^2)
[1] 1.164936
> sqrt(1.164936)
[1] 1.079322
> (log(phat[3]/phat[2]))/1.079322
[1] +1.687658 # refer this to N(0,1)
```

To simulate the distribution of say $\log(p_2/(p_2+p_3))$ use

```
rhyperdirichlet()
```

The work given in this Chapter led to 'Correspondence: Using recently developed software on a 2×2 table of matched pairs with incompletely classified data' by Altham and Hankin, *Applied Statistics* **59**, (2010), 377–379.

Chapter 21

An Index of Child Well-Being in Europe

In 'An Index of Child Well-Being in Europe', Jonathan Bradshaw and Dominic Richardson (published online April 1, 2009 in *Child Indicators Research*) presented the data in the first 10 columns below. These 29 countries consist of the 27 European Union countries together with Norway and Iceland.

The first 6 columns for the various countries consist respectively of ranks for Health, Subjective wellbeing, Children's Relationships, Material Resources, Behaviour and Risk, and Education. Thus we see that for Health, Sweden is best (ie rank 1) but the UK has rank 24 out of 29.

The 'cwb' column is an Index of Child Well-being, constructed from several different aspects of children's lives. The Netherlands does best, with score of 117.25002.

The penultimate column is the Gini index for income distribution. This is a measure of income inequality, and here I have used the United Nations list given on Wikipedia. The Gini index for a given country is defined to be in $[0, 100]$. It is 0 if everyone in that country has the same income (eg \$20000), and is 100 in the case of extreme inequality of income, as when 1 person earns say \$2 million, and everyone else earns \$10. (The mathematical definition is given in Wikipedia.)

The final column (please do check it!) is also taken from Wikipedia. It gives the IMF figures in international \$ of the GDP (gross domestic product) at Purchasing Power per Capita. (This is a better measure of the wealth of an individual in a country than the simple per capita GDP).)

Consider the data in the Table below. Sadly, we have to ask why the UK does so badly: a view expressed in the Child Poverty Action Group report of Spring 2009 'Where the UK stands in the European table'.

	Rank	Country	Health	Subjective	Relationships	Material	Risk	Education
1	1	Netherlands	2	1	1	7	4	4
2	2	Sweden	1	7	3	10	1	9
3	3	Norway	6	8	6	2	2	10
4	4	Iceland	4	9	4	1	3	14
5	5	Finland	12	6	9	4	7	7
6	6	Denmark	3	5	10	9	15	12
7	7	Slovenia	15	16	2	5	13	11
8	8	Germany	17	12	8	12	5	6

```
9     9         Ireland     14         10         14        20   12         5
10   10      Luxembourg      5         17         19         3   11        16
11   11         Austria     26          2          7         8   19        19
12   12          Cyprus     10         NA         NA        13   NA        NA
13   13           Spain     13          4         17        18    6        20
14   14         Belgium     18         13         18        15   21         1
15   15          France     20         14         28        11   10        13
16   16  CzechRepublic      9         22         27         6   20         3
17   17        Slovakia      7         11         22        16   23        17
18   18         Estonia     11         20         12        14   25         2
19   19           Italy     19         18         20        17    8        23
20   20          Poland      8         26         16        26   17         8
21   21        Portugal     21         23         13        21    9        25
22   22        Hungary      23         25         11        23   16        15
23   23          Greece     29          3         23        19   22        21
24   24              UK     24         21         15        24   18        22
25   25         Romania     27         19          5        NA   24        27
26   26        Bulgaria     25         15         24        NA   26        26
27   27          Latvia     16         24         26        22   27        18
28   28       Lithuania     22         27         25        25   28        24
29   29           Malta     28         28         21        NA   14        NA
   Housing       cwb gini GDPppp
1        9 117.25002 30.9 40431
2        3 114.83973 25.0 37245
3        1 114.83537 25.8 53451
4        8 112.71459   NA 40025
5        4 110.96658 26.9 36217
6        5 109.56699 24.7 37266
7       19 107.12159 28.4 29472
8       16 106.08419 28.3 35442
9        2 105.32170 34.3 42539
10       7 104.76378   NA 82306
11       6 104.16369 29.1 39634
12      11 103.72564   NA 29830
13      13 103.64101 34.7 30621
14      12 102.96874 33.0 36235
15      10 100.86114 32.7 34208
16      22  98.89364 25.4 25395
17      15  98.71590 25.8 22040
18      25  96.90719 35.8 20259
19      20  96.08699 36.0 30581
20      23  94.64009 34.5 17482
21      18  94.49209 38.5 22190
22      21  94.32172 26.9 19499
23      14  93.99548 34.3 30535
24      17  92.92664 36.0 36523
25      NA  86.97584 31.0 12580
```

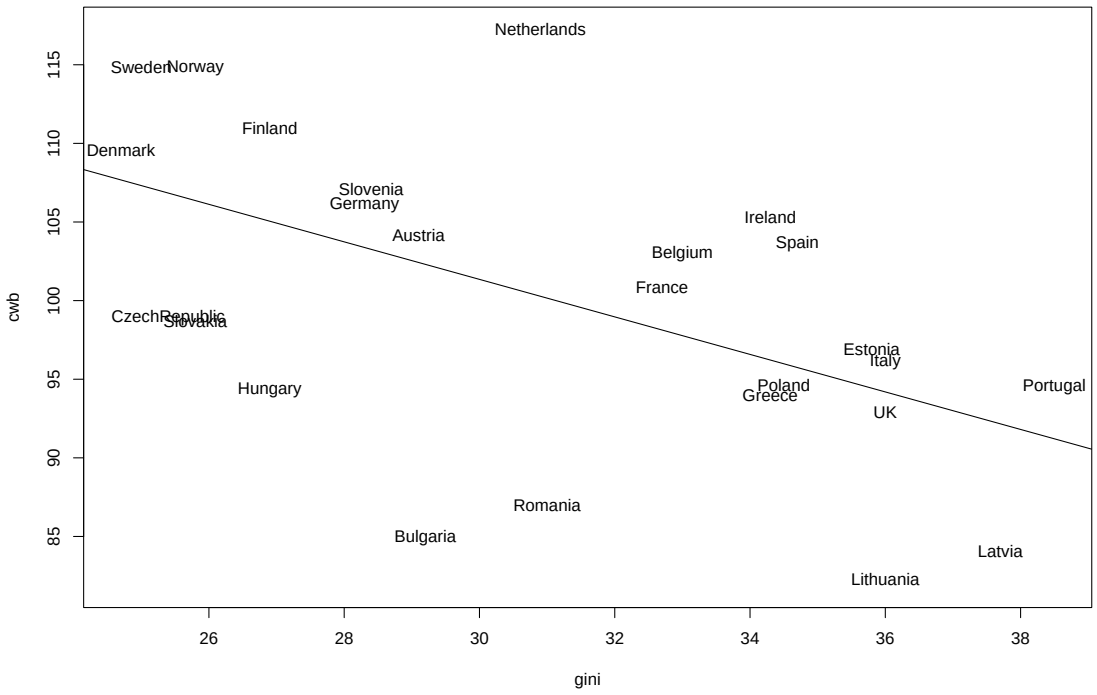


Figure 21.1: How does Child-Wellbeing depend on Income Inequality in Europe?

```
26      NA  84.94478 29.2 12341
27      26  84.06093 37.7 17071
28      24  82.32087 36.0 18946
29      NA  81.89308   NA 23760
```

Figures 21.1 and 21.2 show how Child Wellbeing is negatively related to the Gini Index and positively related to GDPppp for the countries given. (These graphs could be improved, with a little more trouble.)

```
> plot(gini, cwb, type="n")
> text(gini, cwb, Country)
>  points(gini, cwb)
> abline(lm(cwb ~ gini))
```

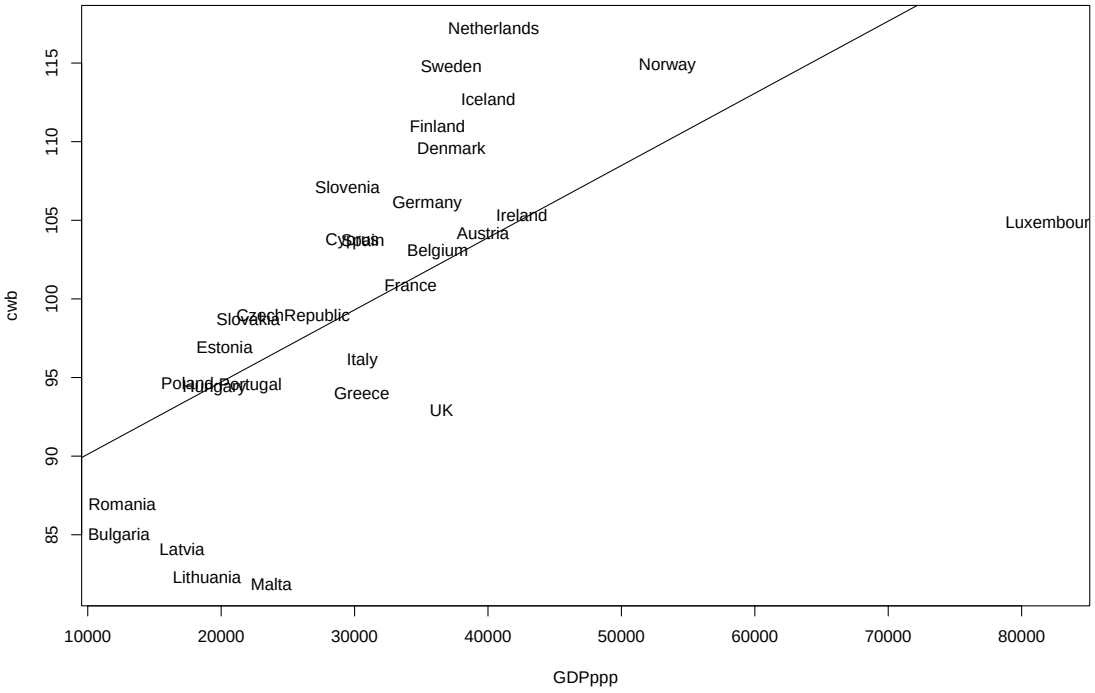


Figure 21.2: How does Child-Wellbeing depend on GDPppp in Europe?

Chapter 22

Estimating the number of dinosaur species not yet found.

This worksheet might form the basis for a graduate project.
While browsing the BBC website (science) I came across the paper 'Estimating the diversity of dinosaurs' (*Proceedings of the National Academy of Science, September 12, 2006*) by Steve C.Wang and Peter Dodson. There are 527 different species (genera) of dinosaurs currently described, and Wang and Dodson estimate from the observations taken so far, that at least 71% of dinosaur genera are yet to be described.
This estimation problem intrigued me, as I recalled that it must be essentially the same problem as that dealt with by Efron and Thisted in 1976 (Biometrika vol 63, p435-447). Their paper was 'Estimating the number of Unseen Species: How many words did Shakespeare know?'. To quote from Efron and Thisted 'Shakespeare wrote 31534 different words, of which 14376 appear only once, 4343 appear exactly twice,... and so on. The question considered is how many words he knew but did not use.'
Wang and Dodson used a non-parametric approach published by A.Chao and S-M Lee, *J.Amer. Statist. Ass*, 1992, pp210-217. In this worksheet, I briefly summarise the non-parametric estimators discussed by Chao and Lee, and present you with 2 examples of real data on which to try out the corresponding R programs.
First, we need some notation, and here we follow closely Chao and Lee.
Suppose we have a random sample of size n taken from a population of elements belonging to N different classes. Let p_i be the probability that any observation belongs to the ith class, and let X_i be the number of elements of the ith class that are observed in our sample ($i = 1, \ldots, N$); thus $(X_1, \ldots, X_N)$ is multinomially distributed.
Let f_i be the number of classes that have exactly i elements in the sample, so that for $i = 1, 2, \ldots, n$,

$$f_i = \sum_{j=1}^{j=N} I[X_j = i],$$

where $I(A)$ is the usual indicator function. The object of our inference is to estimate N, the total number of 'species' or classes, based on the data $(f_1, \ldots, f_n)$.
We define $D = \sum f_i$, the total number of distinct classes observed in the sample. Note that $n = \sum i f_i$.

Chao and Lee define/derive 4 estimators, of increasing sophistication. (You need to look at their paper to see the assumptions on which these estimators are based.)
Estimator 0. The simplest estimator is $\hat{N}_0$, defined as the solution of the equation

$$D = N[1 - \exp(-n/N)].$$

The asymptotic variance of $\hat{N}_0$ is

$$var(\hat{N}_0) = \frac{N}{[e^{n/N} - (n/N) - 1]}.$$

The estimator $\hat{N}_0$ was originally obtained as the approximate mle of N for the special case of $p_1 = p_2 = \ldots = p_N = 1/N$.
(However, if you look up
Harris, 1968, 'Statistical Inference in the Classical Occupancy Problem, unbiased estimation of the number of classes', *J.Amer. Statist. Ass.* **63** pp 837-847.
you will see that Harris takes $N = \theta$, and shows that the **exact** log-likelihood for this problem is

$$L(\theta) = \log \binom{\theta}{D} - n \log \theta + \text{a constant}.$$

Therefore, as a little exercise in R, we could directly maximise $L(\theta)$ and find $var(\hat{\theta})$, without bothering with the approximation to $L(\theta)$.)

Take the following example discussed by Chao and Lee, p213.
A hoard of 204 ancient coins was classified according to die types (eg pounds, 20p, 10p and 1p pieces respectively). The purpose was to estimate the total number of dies used in the coin minting process. The frequencies for the reverse side of the coins were
156 singletons (ie found only once each), 19 doubletons, 2 triplets and 1 quadruplet. The little R program below obtains (in a very simple-minded fashion) the estimate $\hat{N}_0$ together with its standard error.

```
 f = c(156,19,2,1)
D = sum(f)
i = 1:4 ; n = sum(i*f)
N= 200:1000
rhs = N*(1- exp(-n/N)) #Note very simple-minded way of solving the equation
# For a more sophisticated approach, try uniroot()
plot(rhs ~ N, type ="l") ; abline(D,0)
N= 600:800 ; rhs = N*(1- exp(-n/N))
plot(rhs ~ N, type ="l") ; abline(D,0)
N[abs(rhs - D) <.01]
```

This gives us that $\hat{N}_0 = 731$.

```
v0 = N/(exp(n/N) - (n/N) - 1) ; sqrt(v0)
```

This gives $se(\hat{N}_0) = 130.6$.
Estimator 1.

We first define C, the sample coverage, as the sum of the probabilities of the observed classes, ie

$$C = \sum_{i=1}^{i=N} p_i I[X_i > 0].$$

Hence C varies with the sample, and is a random variable. A 'widely used estimator' of C is

$$\hat{C} = 1 - f_1/n,$$

from which a 'natural estimator' of N is

$$\hat{N}_1 = D/\hat{C}.$$

This is asymptotically very efficient in the special case of $p_1 = p_2 = \ldots = p_N = 1/N$.
For the hoard of ancient coins example given above, you can check that $\hat{N}_1 = 757$.
Estimators 2 and 3
The assumption of equal values of $p_1, \ldots, p_N$ is of course in practice unrealistic. The estimators $\hat{N}_2$, $\hat{N}_3$ allow for the fact that the p_i's may vary by incorporating the **coefficient of variation** (or an estimate of it) into their definition.
Each of $\hat{N}_2$, $\hat{N}_3$ is of the form

$$\frac{D}{\hat{C}} + \frac{n(1-\hat{C})\gamma^2}{\hat{C}}.$$

For $\hat{N}_2$, we define $\gamma = \hat{\gamma}$, and for $\hat{N}_3$ we define $\gamma = \bar{\gamma}$. Here γ is defined as CV, the **coefficient of variation** of $(p_1, \ldots, p_N)$, and so

$$\gamma^2 = \frac{\sum_i (p_i - \bar{p})^2/N}{\bar{p}^2}$$

where $\bar{p}$ is the mean of $p_1, \ldots, p_N$, hence $\bar{p} = 1/N$. By definition, $\gamma^2 > 0$. There are two alternative estimators of γ^2. The first is

$$\hat{\gamma}^2 = \frac{\hat{N}_1 \sum i(i-1)f_i}{n(n-1)} - 1$$

if this is > 0, with $\hat{\gamma}^2 = 0$. otherwise,
'When the true value of CV is relatively large' (this seems to mean that $CV \geq 1$) Chao and Lee recommend the use of $\hat{N}_3$, where γ^2 is estimated by a 'bias-corrected' version of $\hat{\gamma}^2$. This is $\bar{\gamma}^2$, defined by

$$\bar{\gamma}^2 = \hat{\gamma}^2 (1 + \frac{n(1-\hat{C})\sum i(i-1)f_i}{n(n-1)\hat{C}})$$

if this is > 0, $\bar{\gamma}^2 = 0$ otherwise.

Exercise 1. Show that

$$E(f_1) = \sum_{i=1}^{N} np_i(1-p_i)^{n-1}.$$

Solution. Note that $f_1 = I_1 + \ldots + I_N$, where $I_1 = 1$ if species 1 is in the sample exactly once, $I_1 = 0$ otherwise (with $I_2, \ldots, I_N$ defined similarly). Hence $Pr(I_1 = 1) = np_1(1-p_1)^{n-1}$, and hence we have the given expression for $E(f_1)$.

Exercise 2. Similarly, show that

$$E(f_j) = \sum_{i=1}^{N} \binom{n}{j} p_i^j (1-p_i)^{n-j}.$$

Exercise 3. Hence show that since $D = \sum_j f_j$, it follows that

$$E(D) = N - \sum_{i=1}^{N} (1-p_i)^n,$$

and since $C = \sum_1^N p_i I(X_i > 0)$, it follows similarly that

$$E(C) = 1 - \sum_{1}^{N} p_i (1-p_i)^n.$$

Back to the practical example given by Chao and Lee.
With $f_1 = 156, f_2 = 19, f_3 = 2, f_4 = 1$,
they find $D = 178$, $\hat{\gamma} = .36$, and so $\hat{N}_2 = 844$ with $se = 186.6$.
Further, they give $\bar{\gamma} = 0.51$, and hence $\hat{N}_3 = 932$, with $se = 265.7$.
Check that you obtain the same estimates $\hat{N}_2, \hat{N}_3$.
Unfortunately, it is not possible to write down a simple expression for the standard errors, but Chao and Lee give the following asymptotic expression as an estimate of the variance of $\hat{N}_2$.

$$var(\hat{N}_2) \approx \sum_i \sum_j d_i d_j cov(f_i, f_j)$$

where $d_i = \frac{\partial \hat{N}_2}{\partial f_i}$, $\quad i = 1, \ldots, n$, and

$$cov(f_i, f_j) = f_i(1 - f_i/\hat{N}_2) \text{ if } i = j$$

$$cov(f_i, f_j) \ = -f_i f_j/\hat{N}_2 \ \text{ if } i \neq j.$$

In other words, the idea behind the variance derivation is quite simple, but its implementation is a bit painful.

Here is a new data set, for which we require the answer to the question 'What is the total number of species?' Statcounter.com provides me with a log of the last 2000 hits on my webpage, and in particular gives me the total number of hits sorted by country of access. On May 5, 2009 I found the following counts, by sorted by country:

```
594 322 109 75 69
65 56 51 45 33
32 26 26 22 22 22 22 20 20 18
18 17 16 15 14 14 14 14 14 13
12 12 12 12 11 11 10 10 10 9
9 8 8 6 6 5 5 5 5 4
3 3 3 2 2 2 2 2 1 1
1 1 1 1 1 1 1 1 1 1
1
```

I will take 'country' as 'species' in this example.
Thus you can see from the data that, there were 594 hits from the first country (which in fact was the USA), 322 from the next (the UK), 109 from the third (Italy), and so forth, finally ending up with a block of 13 countries each of which had exactly one entry. (These were Lesotho, Ghana, Mongolia, Ukraine, Bulgaria, UAE, Jamaica, Israel, Cuba, Indonesia, Guatemala, Cameroon and Belarus respectively. This is not something you need to know in order to answer the question, but I find this list of countries quite intriguing)
From this little example we might consider the problem of estimating N, the total number of species, or equivalently, the problem of estimating f_0, the number of countries not yet 'seen' to hit my webpage.
Chao and Shen 2004 'Nonparametric prediction in species sampling', *Journal of Agricultural, Biological and Environmental Statistics 9, pp 253-269* suggest modifying the estimators given above by **using only the 'rare' species**, ie those which appear at most 10 times each. Thus in all the expressions below, the summations are from $i = 1$ to $i = 10$ only. Simplifying their notation very slightly, we will take $\hat{f}_0$ as our estimator of the number of undetected species (countries), where

$$\hat{f}_0 = \frac{S_{rare}}{C_{rare}} + \frac{f_1 \hat{\gamma}^2}{C_{rare}} - S_{rare}$$

and $S_{rare} = \sum f_i$ is the total number of rare species in the sample, so here $S_{rare} = 35$.
The estimated sample coverage is C_{rare}, where

$$C_{rare} = 1 - f_1 / \sum i f_i$$

and

$$\hat{\gamma}^2_{rare} = \frac{S_{rare} \sum i(i-1) f_i}{C_{rare} (\sum i f_i)^2} - 1,$$

if this is strictly positive, $\hat{\gamma}^2_{rare} = 0$ otherwise.
Here is a little bit of R code to start you off.

```
count <- scan("statcounter.data")
ff <-  tabulate(count)
 f = ff[1:10]
```

Questions for you
I find $\hat{f}_0 = 12.087$. Do you agree?
Is there a painless way of deriving a reasonable estimate of the variance of $\hat{f}_0$?
Agresti (2002, p596) notes that if the N species have respective probabilities $p_1, \ldots, p_N$, then Simpson's coefficient of ecological diversity is defined as

$$I(p) = 1 - \sum p_i^2.$$

Show that $0 \leq I(p) \leq 1 - 1/N$ and state the conditions under which the bounds are attained. Suppose we take two items at random (sampling with replacement) from our population of N species. Show that the probability that these two animals are from different species is $I(p)$.
Note added November2009
Favaro, Lijoi, Mena and Prünster have recently published at least 3 papers on a Bayesian approach to this problem, with special application to 'expressed sequence tags (EST) analysis' in gene identification. See for example

'Bayesian non-parametric inference for species variety with a two-parameter Poisson-Dirichlet process prior' *J.R.Statistic. Soc* B (2009) **71** Part 5, pp 993-1008
'A Bayesian non-parametric method for prediction in EST analysis' *BMC Bioinformatics* 2007, **8:339**
They appear NOT to give a software reference (eg an R package) for their method but they do give a useful reference for the software for other methods, eg the methods of Chao and Shen. This is `http://chao.stat.nthu.edu.tw`
Note added November 2012. Ji-Ping Wang has written the R package SPECIES, with associated article in J. Statistical Software, 2011. This looks very promising as a source of useful methods and interesting data examples.

Chapter 23

Patterns of voting in the Eurovision Song Contest

The 2009 Eurovision Song Contest, held in Moscow in May, has some interesting voting data, available from Wikipedia. Table 23.1 shows the Scoreboard from Semi-final 1. There are 18 countries competing, namely Montenegro, CzechRep, Belgium, Belarus, Sweden, Armenia, Andorra, Switzerland, Turkey, Israel, Bulgaria, Iceland, Macedonia, Romania, Finland, Portugal, Malta, and Bosnia Herzegovina.
There are 20 countries voting, namely those 18 which are competing, plus Germany and the UK. (These latter are 2 out of the 'big four' countries, which together with Russia, the host country, have a guaranteed place in the final.)
No country can vote for itself, and each of the 20 countries voting assigns the votes

$$12, 10, 8, 7, 6, 5, 4, 3, 2, 1, 0$$

to the remaining 17 or 18 countries: Germany and the UK can vote for any of the 18 competing countries.
The row headings for the Table are the 18 competing countries given above, so that the first one you see is Montenegro.
The column headings for the Table are Total score, and then the scores given by the 20 countries listed above, so that the last one you see is the UK, which rates Turkey as the best (as do 7 other countries).
Missing values in the Table are denoted by *.
Try out the following commands and see what they tell you.

```
# I have taken the diagonal elements of the above Table as 'NA'
# You will need to put row and column headings in your datafile.
x = read.table("Eurodata", na.strings="*",  header=T)
x
Votes = x[,3:22] ; Votes
 tVotes = t(Votes) ; tVotes
country.lab = row.names(tVotes)
country.lab
d = dist(tVotes, method = "euclidean") # This has a 'fixup' for NA's
```

44	*	0	0	3	0	5	1	2	5	1	0	0	8	0	0	1	6	10	2	0
0	0	*	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	*	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	2	1	0	*	1	4	0	0	0	4	1	1	6	0	4	0	1	0	0	0
105	0	6	4	7	*	8	7	4	4	7	0	10	3	4	10	8	8	4	4	7
99	4	12	10	10	5	*	0	1	10	10	8	2	2	8	1	0	0	1	10	5
8	0	0	0	0	0	0	*	0	1	0	0	0	0	0	0	4	3	0	0	0
15	0	0	0	2	2	0	2	*	0	0	0	0	0	0	5	2	0	2	0	0
172	8	5	12	6	7	10	5	12	*	6	12	7	12	12	7	5	10	12	12	12
75	5	4	3	4	6	7	8	5	3	*	4	6	1	3	6	0	4	0	5	1
7	0	0	0	0	0	0	0	0	2	0	*	0	5	0	0	0	0	0	0	0
174	7	10	7	12	12	12	10	7	8	12	6	*	4	10	12	12	12	7	6	8
45	10	3	0	0	0	0	0	6	6	0	10	0	*	2	0	0	0	8	0	0
67	6	0	2	1	0	2	4	0	7	8	5	4	7	*	0	10	2	6	1	2
42	3	0	1	0	10	0	3	0	0	0	0	12	0	1	*	3	5	0	0	4
70	0	2	6	0	3	0	12	10	0	2	2	8	0	7	2	*	0	3	7	6
86	1	7	8	8	4	3	6	3	0	5	3	5	0	6	3	6	*	5	3	10
125	12	8	5	5	8	6	0	8	12	3	7	3	10	5	8	7	7	*	8	3

Table 23.1: 2009 Eurovision Song Contest, Scoreboard from Semi-final 1

```
round(d,2)
 library(MASS)
new = cmdscale(d, k=2, eig=T)
country.lab
new$points
x1 = new$points[,1]
x2 = new$points[,2]
 plot(x1,x2, xlim=c(-12,12), ylim=c(-12,12), type="n")
text(x1,x2, labels=country.lab)
dend = hclust(d)
dend
plclust(dend, labels = country.lab)
# Another, perhaps simpler approach is as follows
correlation = cor(Votes, use="pairwise.complete.obs")
# you may prefer to use Spearman's correlation coefficient here
round(correlation,2)
y = as.vector(tVotes)
y[1:18]
18*20
country.voting = gl(20, 1, length= 360, labels=country.lab)
country.voting[1:21]
is.factor(country.voting)
competing = gl(18,20,length=360, labels= country.lab[1:18])
competing[1:25]
Votes = x[,3:22]
tVotes = t(Votes)
```

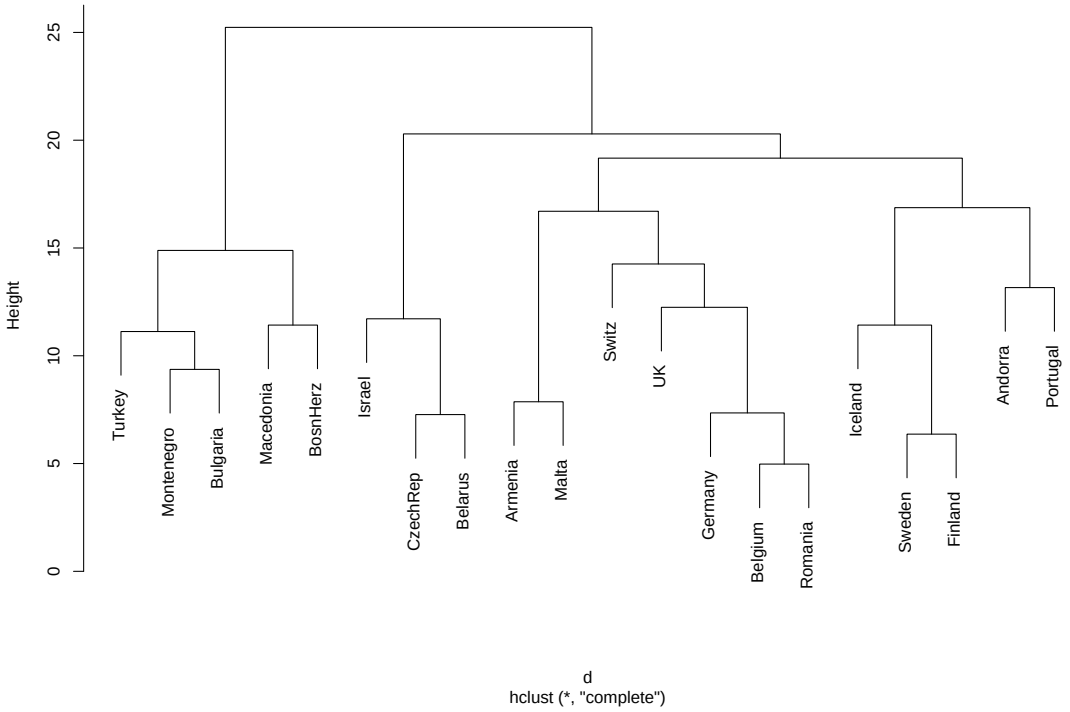


Figure 23.1: Eurovision Song Contest 2009, Semi-final 1, similarities between countries voting

```
y = as.vector(tVotes)# nb, retain *'s for this part
y
plot(y~ competing)
plot(y~ country.voting) # as a check
country.voting
tapply(y, competing, sum, na.rm=TRUE)
tapply(y, country.voting, sum, na.rm=TRUE)
first.lm = lm(y ~ competing ) ; summary(first.lm)
```

Figure 23.1 is the resulting dendrogram, showing which of the 20 countries are voting similarly to one another.

Chapter 24

An introduction to copulae

This worksheet might form the basis of a project for a graduate student.
The only multivariate distribution considered so far in these worksheets is the multivariate normal distribution. This is a very important distribution, with many elegant properties, but in several contexts, particularly those of finance and hydrology, the multivariate normal is too restrictive. This is why I now give a very brief introduction to the subject of Copulae. I will use the R package constructed by Ivan Kojadinovic and Jun Yan, who have gone into the subject very thoroughly.
I found the following paper a very helpful introduction to the subject of copulas:
'Everything you always wanted to know about copula modeling but were afraid to ask (2007) by C. Genest and A-C. Favre, *J. of Hydrologic Engineering* pp 347-368.
I paraphrase their introduction a little.
Suppose (X, Y) is a pair of continuous random variables, and $F(x), G(y)$ are the marginal distribution functions of X, Y respectively. Sklar's theorem shows that the joint distribution function $H(x, y) = Pr(X \leq x, Y \leq y)$ may be written in the form

$$H(x, y) = C(F(x), G(y))$$

for all real x, y. The function C maps the unit square $[0, 1]^2$ to the unit interval $[0, 1]$, and is called the **copula** function. In copula modelling, it is assumed that $C(\ ,\ )$ is the function of interest, as modelling the dependence between X, Y. Everything we say about bivariate distributions is readily extended to multivariate distributions.
An important special type of copula is called an **Archimedean** copula (why 'Archimedean' I have no idea) and for this we may write

$$C(u, v) = \phi^{-1}(\phi(u) + \phi(v))$$

for $0 < u, v < 1$, where ϕ is a convex decreasing function mapping $[0, 1]$ to $[0, \infty)$ such that $\phi(1) = 0$
The special case of $\phi(t) = (t^{-\alpha} - 1)$ for $\alpha \geq -1$ corresponds to a Clayton copula, introduced by David Clayton in 1978.
First, consider the following construction for a Gaussian or normal copula. This enables us to construct a particular bivariate distribution function for which each of the marginal distribution functions is a uniform distribution on $[0, 1]$ and the dependence parameter is the usual correlation coefficent ρ. Take

$$C_\rho(u, v) = \Phi_\rho(\Phi^{-1}(u), \Phi^{-1}(v))$$

for $0 \leq u, v \leq 1$, where Φ^{-1} is the inverse of the univariate standard normal distribution function, and $\Phi_\rho(x, y)$ is the bivariate normal distribution function for a 2−dimensional vector (X, Y) each with mean 0, variance 1 and with correlation coefficient ρ. Here $C_\rho(\ ,\ )$ is a particular example of a **copula** function. You can check that if

$$Pr(U \leq u, V \leq v) = C_\rho(u, v)$$

then the marginal distribution of each of $U,\ V$ is the rectangular distribution on $(0, 1)$. We will first generate a random sample from a bivariate Gaussian copula:

```
library(copula, lib.loc="~/Rlibs")
norm.cop <- normalCopula(0.7, dim=2) # here rho= 0.7
set.seed(26) # for repeatability
x <-  rcopula(norm.cop,1000)#to generate a sample from this distribution
plot(x)  # to show dependence
hist(x[,1] ) # to show uniform marginal distribution of first variable
hist(x[,2])  # to show uniform marginal distribution of second variable
cor.test(x[,1] , x[,2], method="kendall") # note, non-parametric correlation
# so this is unaffected by 1-1 transformations on each of x[,1], x[,2]
kendallsTau(norm.cop) # so this is the theoretical value
```

Now try changing the scale

```
qx <- qnorm(x)
plot(qx)
hist(qx[,1]) # to show N(0,1) distribution
cor.test(qx[,1],qx[,2], method="pearson")
cor.test(qx[,1],qx[,2], method="kendall") # same result as for (x[,1],x[,2])
# as we should expect.
```

Now let us generate a Clayton copula bivariate distribution, and assess the pairwise dependence by Kendall's τ. Note that for an Archimedean copula, there is a closed form expression for Kendall's τ: see for example p354 of Genest and Favre. In terms of the parameter used by the R package copula, Kendall's τ for the Clayton copula is given by

$$\tau = \alpha/(\alpha + 2).$$

First we will plot the probability density function for the bivariate Clayton copula with $\alpha = 1, \tau = 1/3$, shown in Figure 24.1.

```
clayton.cop1 = claytonCopula(1,dim=2)
persp(clayton.cop1,dcopula) # to look at the density function
```

Now we show the scatterplots for samples of size 1000 drawn from a Clayton copula with $\alpha = 3$, as Figure 24.2, and then with $\alpha = 4$, as Figure 24.3.

```
clayton.cop3 <- claytonCopula(3, dim=2)
kendallsTau(clayton.cop3) # we expect the answer 3/5
x3 <- rcopula(clayton.cop3,1000) # to generate a sample from this distribution
```

```
plot(x3)
cor.test(x3[,1] , x3[,2], method="kendall")
clayton.cop4 <- claytonCopula(4, dim=2)
kendallsTau(clayton.cop4) # we expect the answer 4/6
x4 <-  rcopula(clayton.cop4, 1000)
plot(x4)
cor.test(x4[,1] , x4[,2], method="kendall")
```

Now let's move up to 3 dimensions, and generate a sample from a 3-dimensional Clayton copula with $\alpha = 3$, with the resultant plot as Figure 24.4.

```
clayton.cop33 = claytonCopula(3, dim=3)
 x33 = rcopula(clayton.cop33, 1000)
scatterplot3d(x33)
# warning: when you've done this 3-dim plot, it's possible you can't immediately
# go back to doing a 2-dim plot
```

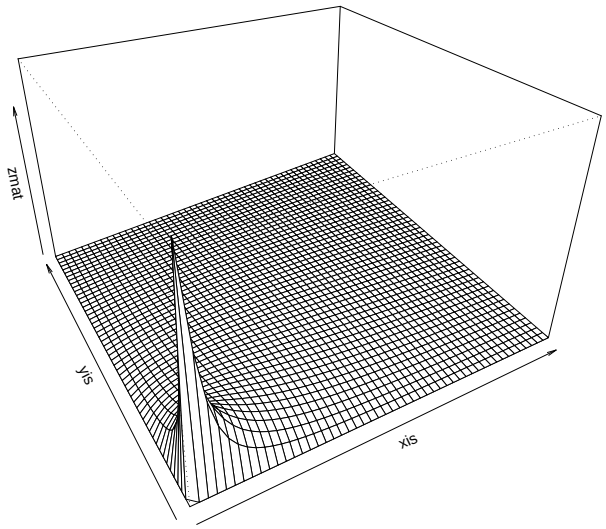


Figure 24.1: The probability density function for a Clayton copula with $\tau = 1/3$

Sometimes we want to introduce a dependence between the observed variables which is say, stronger when both variables are large and negative than when both are large and positive. As you can see from the scatterplots, a Clayton copula will do this nicely.
There's lots of scope for experimenting with the functions in this package, and the paper by Genest and Favre will give you suggestions for model-fitting.
But: be warned..... Wikipedia tells us that

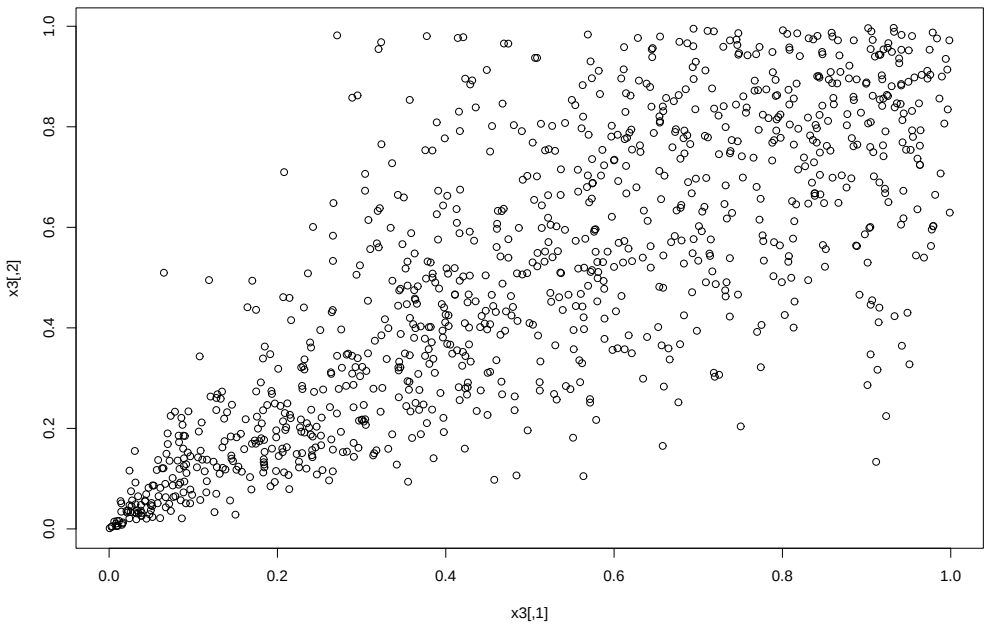


Figure 24.2: Scatter plot for 1000 points from a Clayton copula with $\tau = 3/5$

'The Methodology of applying the Gaussian copula to credit derivatives as developed by David X.Li is said to be one of the reasons behind the global financial crisis of 2008-2009.'

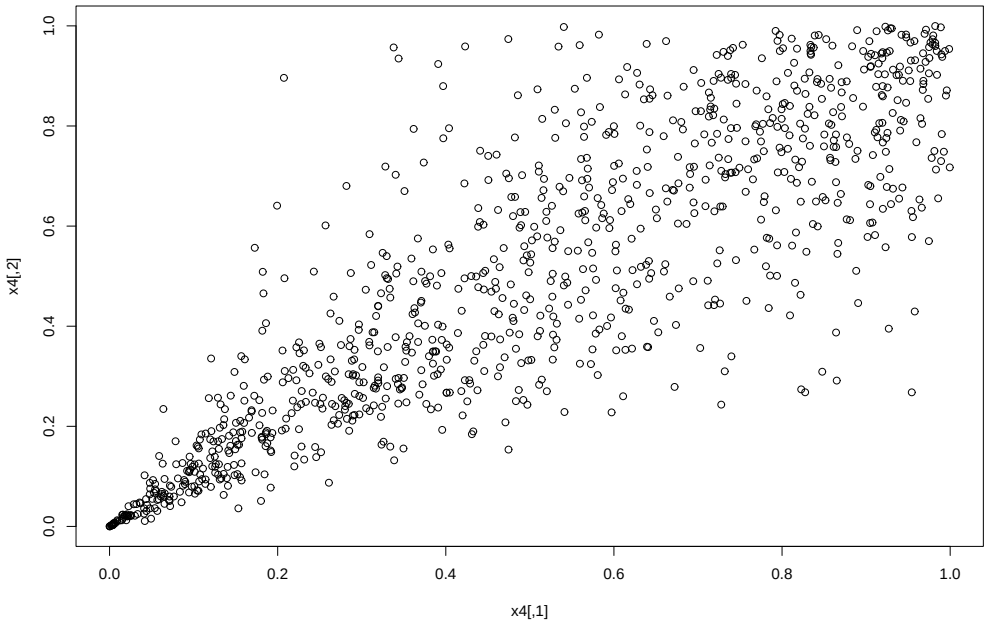


Figure 24.3: Scatter plot for 1000 points from a Clayton copula with $\tau = 4/6$

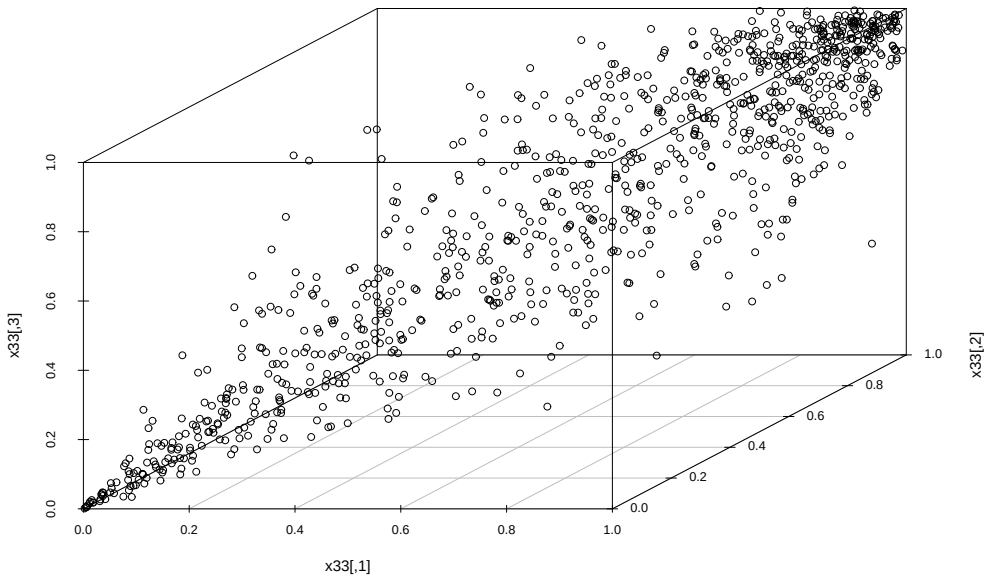


Figure 24.4: Scatter plot for 1000 points from a 3-dim Clayton copula with $\alpha = 3$

Chapter 25

Standardising or 'raking' a contingency table

Here we discuss how to 'rake' a contingency table, and also how to compare two contingency tables, using an example from historical demography.
One important feature of copula-modelling for say X, Y is the idea of standardising the marginal distribution for each of X, Y to be uniform. This same approach may be found helpful in assessing data from contingency tables with ordered rows and columns, as for example in the square table on social mobility given as Table 25.1 here. This dataset is taken from 'Comparing contingency tables: tools for analysing data from two groups cross-classified by two characteristics' (2007) by P.M.E.Altham and J.P.Ferrie, *Historical methods* **40**, 3-16. The dataset shows for a sample of 8999 US males,
as the **Columns** the social classes of Fathers' occupations in 1850, and
as the **Rows** their sons' occupations in 1880.
Clearly the sample size is too large to make fitting of subtle models a realistic proposition, so we focus on how to present the data as helpfully as possible. Thus we invoke **table standardisation**, also known as **raking** the table, as a helpful (we hope) way of presenting the data. This keeps the cross-ratios of the table as in the original table, but adjusts the margins so that each margin is a uniform distribution. Agresti (2002) p346 gives the computational 'trick' to be used in glm: here of course we implement it in R.

	White-collar	Farmer	Skilled/semi-skilled	Unskilled	Row total
White-collar	260	715	424	142	1541
Farmer	194	3245	454	247	4140
Skilled/semi-skilled	158	874	751	327	2110
Unskilled	70	664	246	228	1208
Column total	682	5498	1875	944	8999

Table 25.1: US Social Mobility: columns are Fathers in 1850, rows are Sons in 1880

```
n1 <- scan()
260 715 424 142
194 3245 454 247
```

```
158 874 751 327
70 664 246 228

Row <- gl(4,4,length=16) ; Column <- gl(4,1, length=16)
# Now invent a 'y-variable' which has the desired row and column totals
e <- 25*(n1/n1) # then e will have every row sum and every column sum equal to 100
first.glm <- glm(e ~ Row + Column + offset(log(n1)), poisson)
fv <- first.glm$fitted.values
# then fv will have every row sum and every column sum equal to 100,
# and fv will have same cross-ratios as n1
fv <- matrix(fv,byrow=T,  nrow=4) # so here is the 'raked' contingency table
round(fv,2)
     [,1]  [,2]  [,3]  [,4]
[1,] 42.58 16.15 25.42 15.85
[2,] 19.88 45.84 17.03 17.25
[3,] 20.35 15.52 35.42 28.71
```

Note that we can find an overall measure of the association between rows and columns in this table as follows.

```
aov(log(n1) ~  Row + Column)
                     Row   Column Residuals
Sum of Squares  1.546580 8.707645  2.051127
Deg. of Freedom        3        3         9
 ....................................
```

The residual sum of squares, here 2.051127, will be zero if and only if there is exact independence of rows and columns in the Table. Furthermore, this residual sum of squares is unaffected by the marginal totals in the Table: you can check this for example by

```
aov(log(fv) ~ Row +Column)
```

However, we must note that the residual sum of squares does of course depend on the numbers of rows and columns in the Table.
We may wish to compare the 1850-1880 transitions with those in a later Table, say those from 1880-1910, given here as Table 25.2.

	White-collar	Farmer	Skilled/semi-skilled	Unskilled	Row total
White-collar	1538	1622	1203	529	4892
Farmer	550	3371	363	409	4693
Skilled/semi-skilled	907	1486	1736	858	4987
Unskilled	500	1428	622	611	3161
Column total	3495	7907	3924	2407	17733

Table 25.2: US Social Mobility: columns are Fathers in 1880, rows are Sons in 1910

```
n2 <- scan()
 1538 1622 1203 529
```

```
  550 3371 363  409
  907 1486 1736 858
  500 1428 622  611

 aov(log(n2) ~Row + Column)
                      Row    Column Residuals
Sum of Squares  0.8874445 2.9285843 2.1440888
Deg. of Freedom         3         3         9
.....................................

 aov(log(n2/n1) ~ Row + Column)
                      Row    Column Residuals
Sum of Squares  2.0175237 2.9131499 0.2207554
Deg. of Freedom         3         3         9
.....................................
```

Let I be a 4×4 independence Table, for example a Table with every element 25. We now have a distance measure with which to compare the associations between rows and columns in the Tables $n1, n2, I$, where $n1, n2$ are the Tables for 1850-80 and 1880-1910 respectively. Thus

$$(d(I, n1))^2 = 2.051127,\ (d(I, n2))^2 = 2.1440888,\ (d(n1, n2))^2 = 0.2207554$$

and hence

$$d(I, n1) = 1.43217, d(I, n2) = 1.464270,\ d(n1, n2) = 0.469861\ .$$

This idea of a *metric* makes it possible for us to plot the 3 Tables $I,\ n1,\ n2$ onto a 2-dimensional graph, as in Figure 25.1. While this is a rather trivial exercise for just 3 Tables, it may become more useful if we have several 4×4 tables we wish to compare. Here's one way of putting the 3 Tables $n1, n2, I$ onto a 2-dimensional graph.

```
# First set up the 3 by 3 symmetric distance matrix d as
0    0.469861 1.43217
0.469861 0    1.464270
1.43217  1.464270  0

# Now set up labels for the 3 points
lab= c("1850-1880", "1880-1910","Indep")
x <- cmdscale(d, eig =T) ; x
plot(x$points, xlim=c(-1,1), ylim=c(-1,1), type="n")
text(x$points, xlim=c(-1,1), labels=lab)
```

This results in the graph shown as Figure 25.1. You will see that 1850-80 and 1880-1910 are pretty close to each other, relative to their distances from the Independence table, but 1850-80 is just a little nearer to Independence than is 1880-1910. (Three points with known interpoint distances can always be put exactly onto a plane; if we had say four points with known interpoint distances then we would generally have to make an approximation.)

Note that this is purely a 'data-analytic' way of presenting the the features of interest from the data: we could of course use glm() to do formal model-fitting if we preferred, for example to test

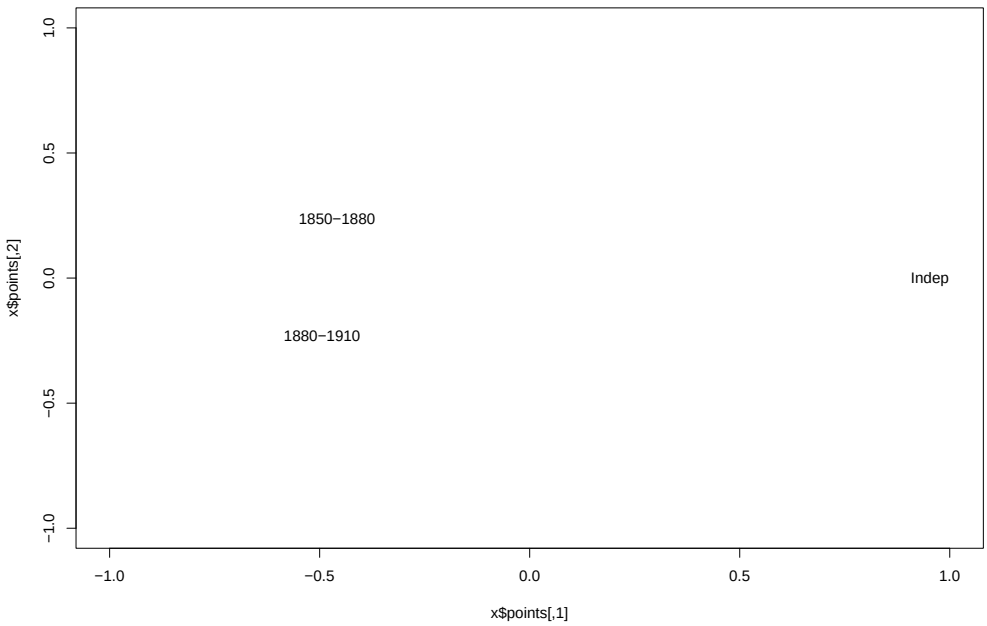


Figure 25.1: The associations in 3 social mobility Tables (each of which is 4×4) plotted by cmdscale

whether the interactions in the 1850-1880 Table are the same as those in the 1880-1910 Table.
.....

Here's another way of looking at the association between rows and columns in a contingency table.
Assuming that there is a natural ordering for the rows and the same natural ordering for the columns, we might want to use the original Table 25.1 to construct a **non-parametric correlation** between x, the father's social class, and y, his son's social class. We will then compute Kendall's τ as a measure of association of x and y. Note that this coefficient was originally constructed for continuous data, and the x, y here are highly discrete. Agresti (2002) p58 gives a formula for τ in the case of a contingency table, and Dr Laura Thompson gives suitable R code. I adopt a different approach, which is probably slower than Laura's (but it is a bit quicker to program).

```
# Let the 4 by 4 table be a matrix with elements n[i,j]
# To reconstruct the x, y variables, proceed as follows
> N <- sum(n) ; x <- 1:N ; y <- 1:N # to set up x,y as vectors
# with the right number of elements
# subtle triple loop coming up!
> u <-  0 # this will be our counter variable
> for (i in 1:4){
+ for (j in 1:4){
+ for (k in (u+1):(u + n[i,j])){
```

```
+ x[k]=i; y[k]= j
+ }
+ u= u + n[i,j]
+ }
+ }
# This process has a name but I've forgotten what it is
> n
     [,1] [,2] [,3] [,4]
[1,]  260  715  424  142
[2,]  194 3245  454  247
[3,]  158  874  751  327
[4,]   70  664  246  228
> table(x,y) # as a check
   y
x      1    2    3    4
  1  260  715  424  142
  2  194 3245  454  247
  3  158  874  751  327
  4   70  664  246  228

> cor.test(x,y, method= "kendall")
        Kendall's rank correlation tau
data:  x and y
z = 16.3894, p-value < 2.2e-16
alternative hypothesis: true tau is not equal to 0
sample estimates:
      tau
0.1514228
```

Chapter 26

The Independent (2009) data on UK universities.

This dataset was taken from The Independent, in August 2009, with the kind assistance of Julia Blackwell. It is given here in 'comma-separated variables' form to save horizontal space. (The dataset has been slightly edited for simplicity: I have omitted all the little comments about particular data-rows. You can find these on the original website).
I have made a few suggestions for the analysis and visual presentation of the data set below.

```
Rank2010,Rank2009,Institution,StudentSatisfaction,ResearchAssessment,EntryStandards,
StudentStaffRatio,AcademicServicesSpend,FacilitiesSpend,GoodHonours,
GraduateProspects,Completion,Score
1,-1,Oxford,4.11,3,524,10.9,2639,587,91.1,82.3,97.6,1000
2,-2,Cambridge,4.18,3,539,11.6,1755,657,87,85.5,99,947
3,-3,Imperial College,3.69,2.9,489,10.3,3036,575,68.5,88.4,97.1,910
4,-3,London School of Economics,3.77,3,483,14.1,1391,265,76,90.6,96.5,853
5,-6,Durham,3.96,2.7,459,15.2,1036,682,77.5,78.3,96.7,819
6,-5,Warwick,3.9,2.8,463,13.5,1630,354,79.7,79.2,95.9,817
7,-7,St Andrews,4.22,2.7,468,12.3,1152,357,85.1,77.8,94.2,801
8,-8,University College London,3.85,2.8,452,9.1,1586,220,80.4,82.9,92,793
9,-14,Bath,3.9,2.7,440,15.3,960,424,75.1,81.9,95.6,765
10,-11,York,3.96,2.8,434,13.4,1246,369,74.9,69.4,95.9,764
11,-21,Edinburgh,3.66,2.7,447,12.8,1660,335,80.6,76.9,90.4,756
12,-10,Lancaster,3.92,2.7,388,13.7,1036,488,69.6,64.3,93.3,749
13,-20,Southampton,3.81,2.7,407,14,1154,449,74.6,76.5,93.7,739
13,-23,Aston,4.02,2.4,365,16.6,895,675,63.6,78.1,91,739
15,-9,SOAS,3.82,2.6,378,10.7,1520,253,73.2,73.5,82.5,735
16,-16,Bristol,3.72,2.7,447,13.3,1386,307,81.5,82,95.7,734
17,-15,Kings College London,3.74,2.7,415,11.6,1589,320,72.8,83.2,92.3,724
18,-13,Loughborough,4.13,2.6,368,17.3,744,496,67.7,75.7,91.4,720
19,-16,Nottingham,3.79,2.7,408,13.8,972,392,74.3,76.3,95.7,713
20,-12,Leicester,4.13,2.5,360,14.8,970,460,71.6,76.2,93,706
21,-19,Exeter,4.15,2.6,394,17.6,987,384,79.4,71.7,91.5,704
```

```
22,-26,Sheffield,3.88,2.7,406,14.5,969,354,74.5,79.1,92.2,703
22,-22,Royal Holloway,3.89,2.7,365,14.6,868,389,70.3,69.8,92.9,703
24,-27,Newcastle,3.84,2.6,405,14.9,1097,398,72.2,79.4,92.3,700
25,-29,Sussex,3.88,2.6,378,15.6,785,333,81.7,70.6,90.1,697
26,-16,Glasgow,3.88,2.6,412,13.5,933,511,71.3,75.4,86.7,693
27,-24,Birmingham,3.89,2.6,403,15.3,1216,401,70.9,72.7,93.7,689
28,-27,Manchester,3.67,2.8,412,14.7,1136,362,70.4,73.8,91.6,685
28,-32,Leeds,3.82,2.7,392,14.4,851,385,73.4,71.1,91.9,685
30,-25,Essex,3.87,2.8,302,13.8,1050,413,61,62.7,87.4,671
31,-30,Surrey,3.8,2.6,352,17,858,347,65.1,80,88.8,653
32,-33,East Anglia,4.08,2.6,361,17.1,866,401,70.1,71.9,85.5,651
33,-34,Queens Belfast,3.8,2.6,358,15.3,905,511,69,78.3,85,648
34,-42,Liverpool,3.77,2.5,387,12.3,1196,285,68.8,72.5,91.2,647
35,-38,Reading,3.91,2.5,347,16.6,709,343,75.4,68.7,91.7,646
36,-37,Cardiff,3.75,2.7,394,14.9,927,253,66.8,77.6,92.5,645
37,-44,Goldsmiths College,3.79,2.6,318,12.8,641,178,65.2,69,85.6,635
38,-35,Kent,4,2.6,317,17,733,312,61.7,71.8,87.8,624
39,-49,Heriot-Watt,3.8,2.5,350,15.8,832,391,65.5,76.2,80.3,618
40,-31,Strathclyde,3.73,2.5,393,19.2,1118,254,74,78.1,83.2,615
41,-41,Brunel,3.74,2.3,319,17.5,1080,423,64.9,67.9,86.3,612
42,-40,Keele,3.92,2.4,319,14.5,808,331,64.4,70.4,89.1,611
43,-43,Queen Mary,3.82,2.7,346,13.4,878,338,64.2,77.3,88.5,609
44,-46,City,3.77,2.5,316,17.9,776,305,66.6,81.4,84.3,599
45,-54,Hertfordshire,3.71,2.4,244,15,822,850,47.3,65.3,82.3,597
46,-52,Nottingham Trent,3.81,2.2,276,16.6,795,399,55.7,74.1,86.2,592
47,-39,Aberdeen,3.97,2.6,363,15.1,929,224,67.6,74.4,77.3,590
48,-35,Stirling,3.82,2.4,324,14.3,873,153,64.3,69.8,81.4,589
49,-52,Dundee,3.84,2.5,371,14.9,935,262,65.8,75.6,71.3,574
50,-61,Robert Gordon,3.82,2.1,332,18.2,913,288,53.2,84.1,81.8,569
51,-51,Ulster,3.74,2.5,268,16.2,1160,356,61.4,65,77.4,568
52,-48,Hull,4,2.4,285,19,799,281,57.7,73.2,87,566
53,-59,Oxford Brookes,3.83,2.2,301,18.7,648,380,66.1,73,83.8,565
54,-56,Bournemouth,3.79,2.2,289,21.2,896,233,57.2,75.5,85.1,558
55,-45,Aberystwyth,4,2.5,310,17.7,746,364,61.1,53,87.5,548
56,-47,Swansea,3.88,2.4,304,15,778,303,50.3,62.5,87.4,536
57,-49,Bradford,3.73,2.3,270,15,830,279,62.8,70.4,82.4,535
58,-73,Northumbria,3.91,2.1,291,17.7,787,271,54.1,73,80.3,534
59,-69,Birmingham City,3.71,2.3,258,17.6,827,519,57.3,65.7,75.4,533
60,-56,U of the Arts London,3.47,2.5,322,22.1,687,90,61.6,63.6,85.5,523
61,-66,W of England Bristol,3.86,2.2,275,19.3,696,328,63.2,66.1,78.8,512
62,-60,Portsmouth,3.94,2.2,271,19,917,265,52.4,63.9,86.1,510
63,-67,Brighton>,3.79,2.4,278,19.8,579,239,58.5,65.3,84.7,506
63,-55,Bangor,3.85,2.4,283,19.2,751,200,55.7,67.8,83.8,506
65,-65,UWIC Cardiff,3.85,1.9,261,20.1,798,461,51.8,61.5,83.2,498
65,-64,De Montfort,3.92,2.3,248,17.2,731,232,51.1,65.7,81.5,498
67,-71,Chichester,4.04,2,234,18.8,567,377,49.3,64,89.2,491
68,-80,Sheffield Hallam,3.72,2,268,18.9,735,239,61.3,64.2,83.8,490
```

```
69,-74,Winchester,3.86,2.2,261,17.5,588,271,55.7,55.6,85.6,482
70, ,U for the Creative Arts,3.55,2.3,247,24.9,1167,284,51.2,58.2,85.3,480
71,-58,Gloucestershire,3.79,1.7,239,18.7,886,354,56.2,64.2,82.1,478
72,-85,Bedfordshire,3.81,2.1,215,15.7,662,426,47.8,66.7,75.1,474
72,-77,Edinburgh Napier,3.82,1.8,291,18.3,780,154,62.4,72.9,72.7,474
74,-96,Glasgow Caledonian,3.8,2,328,21.8,814,159,66.5,68.8,76.8,473
75,-70,Plymouth,3.78,2.2,268,15.9,787,251,61.4,58,84.2,468
76,-71,Salford,3.74,2.4,253,18.8,725,300,55.1,62.7,76.5,463
77,-101,Lampeter,3.99,2.1,252,15.5,658,149,52.8,61.8,82.7,460
78,-99,Lincoln,3.88,1.9,266,22.9,724,228,57.1,60.4,85.1,457
79,-63,Central Lancashire,3.84,2,252,20.2,826,361,49.3,70.1,72.8,456
80,-80,Staffordshire,3.81,1.6,232,17.9,897,311,53,69,78.1,455
81,-89,Kingston,3.83,2,236,19.4,723,310,60.1,61,79.3,454
82,-88,Manchester Metropolitan,3.65,2.2,266,19.9,779,242,57.2,63.8,79,448
82,-76,Bath Spa,3.92,1.9,287,24,418,133,67.7,59,87.1,448
84,-105,Huddersfield,3.77,2.1,265,16.3,671,205,51.9,56.9,81.2,447
85,-84,Coventry,3.69,1.9,280,20.8,798,294,61,67.5,75.2,445
86,-75,Queen Margaret,3.82,1.4,328,20.7,833,111,65.3,69.2,78.6,441
87,-77,Roehampton,3.65,2.2,251,18.9,1049,215,51,56.8,78.5,439
88,-82,Middlesex,3.72,2.2,194,24.1,1476,442,50.8,64.4,64.7,433
89,-92,Westminster,3.64,2.2,249,16.5,581,237,51.5,54.1,77.5,427
90,-90,Chester,3.8,1.7,267,19.3,534,210,54.8,65.1,78.2,422
91,-97,York St John,3.84,1.4,288,20.3,875,299,52.9,58.8,84.4,415
92,-102,Teesside,3.88,2,257,19.9,668,196,52.4,67.3,72.7,411
93,-79,Sunderland,3.93,1.9,226,17.1,554,311,50.2,59.3,75.7,410
94,-86,Glamorgan,3.75,2,263,18,701,202,52.4,60.4,69.1,409
95,-68,Abertay Dundee,3.82,1.8,279,20.5,1260,246,48.3,53.3,70.1,399
96,-93,Worcester,3.88,1.5,236,20.3,708,165,48.4,68.2,83.9,398
95,-68,Abertay Dundee,3.82,1.8,279,20.5,1260,246,48.3,53.3,70.1,399
96,-93,Worcester,3.88,1.5,236,20.3,708,165,48.4,68.2,83.9,398
97,-98,Leeds Metropolitan,3.54,2.1,257,23.1,825,120,53,59.9,82,397
98,-110,Thames Valley,3.76,1.7,197,16.8,878,432,50.4,59.7,63.6,395
98,-83,U of Wales Newport,3.8,2.4,229,26.4,530,304,54.8,55.8,72,395
100, ,Glyndwr,3.83,1.7,212,20,776,345,50.6,69.3,70.1,394
101,-103,Derby,3.75,2,231,19.2,1073,251,46.6,54.9,76.5,392
102,-107,Wolverhampton,3.7,2,204,19.2,800,419,44.6,60.9,73.4,391
103,-86,Buckinghamshire New,3.64,1.7,210,20.4,550,633,46.3,50.9,79.7,388
104,-100,Liverpool John Moores,3.72,2.2,244,19.7,790,212,45.7,58.8,76.2,387
104,-93,Northampton,3.83,1.7,233,20.8,693,209,53.7,59.5,81.4,387
106,-105,East London,3.63,2.2,191,22.7,725,404,44.4,62.7,71.6,371
107,-90,Canterbury Christ Church,3.8,1.8,238,19.1,537,148,49.7,59,80.3,358
108,-104,Cumbria,3.71,1.2,256,31.7,496,181,48.8,71.4,84.6,347
109,-107,Edge Hill,3.84,1.5,248,19.9,928,160,46.1,56.8,79.2,344
110,-95,Southampton Solent,3.58,1.5,211,21.6,699,351,45.7,56.6,74.5,337
111,-111,Greenwich,3.91,1.9,212,23.1,619,219,44.9,59.6,75.8,333
112,-112,Bolton,3.77,1.7,213,19.4,392,328,53.5,59.1,56,307
113,-113,London South Bank,3.63,2.2,179,27.9,652,149,53.7,63.2,69.4,280
```

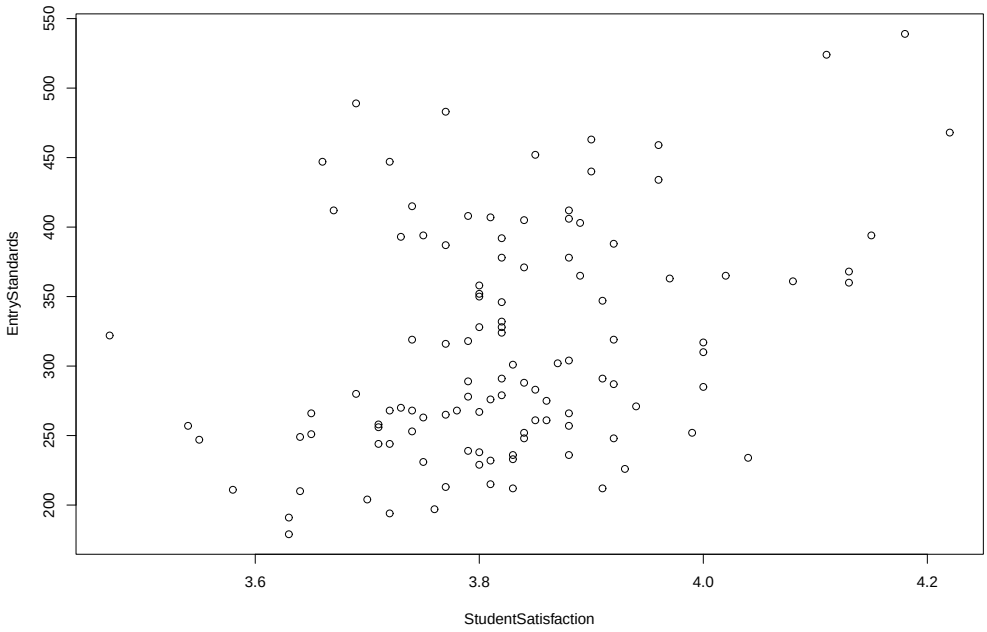


Figure 26.1: Scatterplot for StudentSatisfaction,EntryStandards data

Here are suggestions for your R-commands, including a 2-dimensional kernel density estimate.

```
> Univ.Data09 <-  read.csv("Univ.Data09", header=T)
> attach(Univ.Data09)
> plot(StudentSatisfaction,EntryStandards)
>library(MASS)
> f1  <- kde2d(StudentSatisfaction,EntryStandards, n=113)
> image(f1)
> plot(StudentSatisfaction,EntryStandards)
> image(f1, xlab="StudentSatisfaction", ylab="EntryStandards")
> contour(f1, xlab="StudentSatisfaction", ylab="EntryStandards")
> persp(f1, phi=30, theta=20,  xlab="StudentSatisfaction", ylab="EntryStandards")
```

We look at the relationship (if any) between StudentSatisfaction and EntryStandards, and we take this opportunity to use the Venables and Ripley computation of the two-dimensional probability density function of these two variables. As you will see from Figures 26.1, 26.2, 26.3 and 26.4, this probability density function is far from bivariate normal.
Finally, we show as Figure 26.5, the histogram of the 2008 ResearchAssessment scores. You will observe how this distribution differs from the previous ResearchAssessment distribution (see the data given for Worksheet 3) which was distinctly bimodal. (Note further that the 'scoring system' for 2008 is different from the previous RAE 'scoring system')

```
> truehist(ResearchAssessment)
```

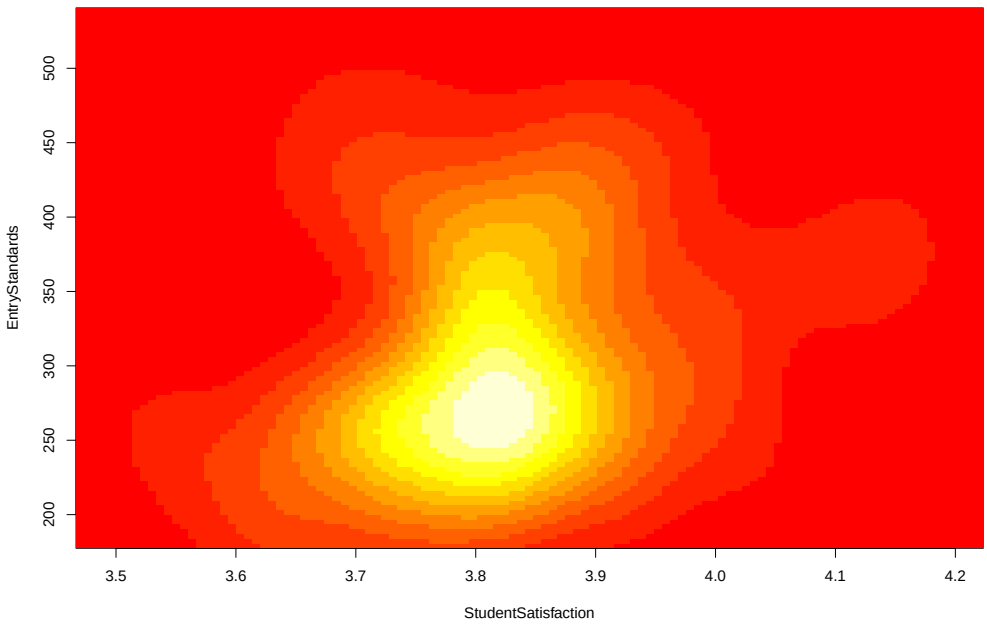


Figure 26.2: Image plot of 2-dimensional pdf for Universities 2009 data

Let us also fit a normal distribution to the StudentSatisfaction variable, and plot the corresponding density function.

```
 y <- Univ.Data09$StudentSatisfaction
> truehist(y)
> fitdistr(y,"normal")
      mean           sd
  3.826814159   0.131143814
 (0.012336972) (0.008723557)
> u <-  345:423 ; u <- u/100 # to set up suitable range
> f <- dnorm(u, mean=3.826814159, sd = 0.131143814)
> lines(u,f) # to superimpose the fitted normal density on the histogram
```

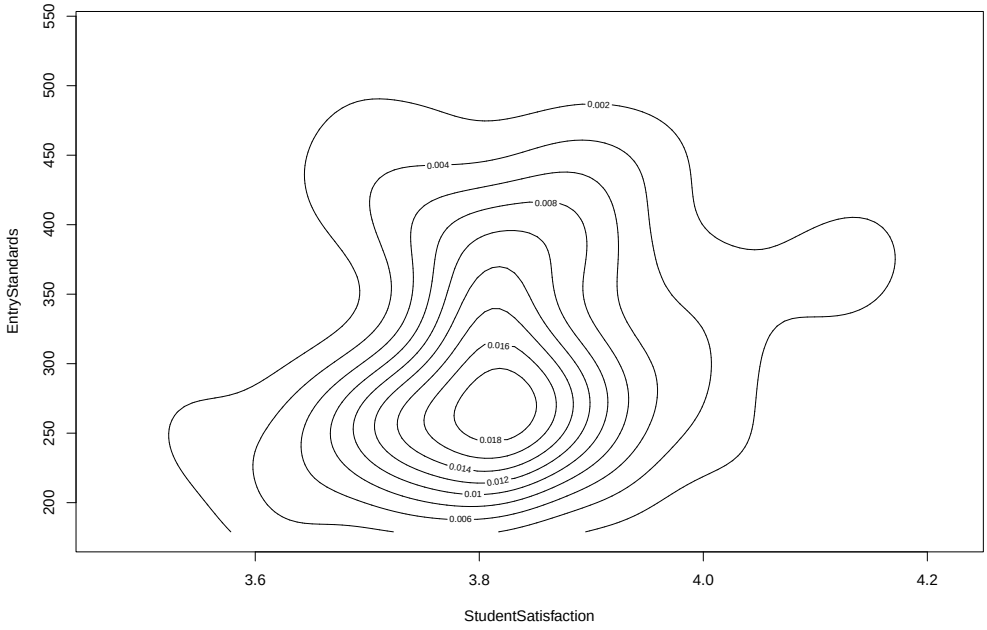


Figure 26.3: Contour plot of 2-dimensional pdf for Universities 2009 data

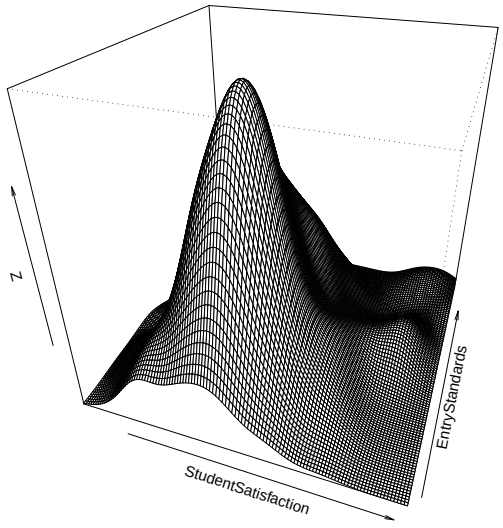


Figure 26.4: Perspective plot of 2-dimensional pdf for Universities 2009 data

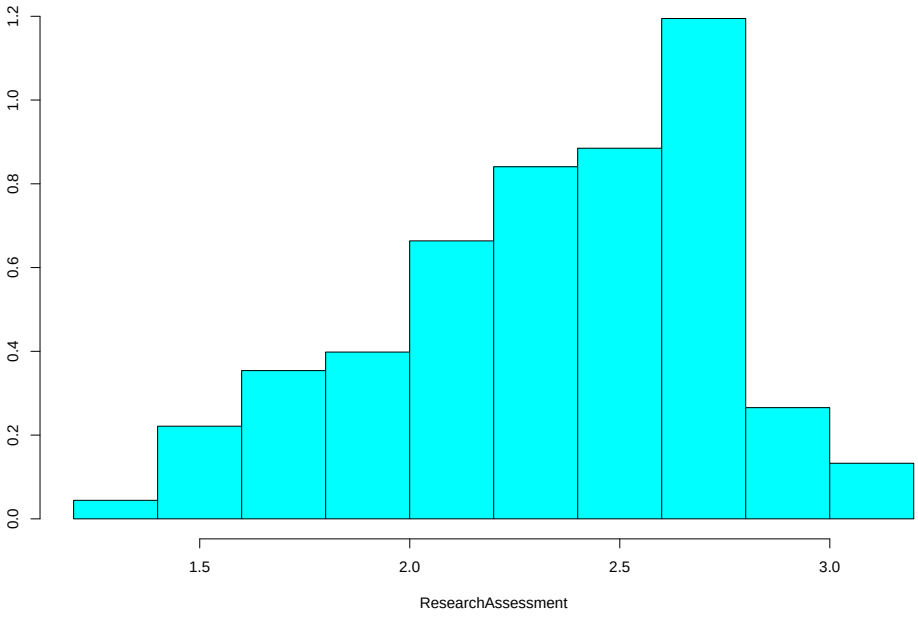


Figure 26.5: The histogram of the 2008 RAE scores

Chapter 27

An intriguing multivariate data-set: assessing the Habitability of Exoplanets

In 2011 Schulze-Makuch *et al* published the article 'A Two-Tiered Approach to Assessing the Habitability of Exoplanets' in Astrobiology, vol. 11, no. 10. They compute the ESI, or Earth Similarity index for various bodies. Their Table 2 gives 'Data and Calculations for the determination of the ESI as applied to selected solar system planets and satellites, and exoplanets'. This Table, slightly edited by me, is given below.

Planet	Rad	Density	EV	SurfTemp	IntESI	SurfESI	GlobESI
Earth	1.00	1.00	1.00	288	1.00	1.00	1.00
Mars	0.53	0.71	0.45	227	0.82	0.60	0.70
Mercury	0.38	0.98	0.38	440	0.84	0.42	0.60
Moon	0.27	0.60	0.21	220	0.67	0.46	0.56
Venus	0.95	0.95	0.93	730	0.98	0.20	0.44
Io	0.29	0.64	0.23	130	0.69	0.19	0.36
Callisto	0.38	0.33	0.22	134	0.58	0.20	0.34
Jupiter	10.97	0.24	5.38	152	0.36	0.24	0.29
Ganymede	0.41	0.35	0.25	110	0.60	0.14	0.29
Ceres	0.08	0.36	0.05	167	0.41	0.18	0.27
Europa	0.25	0.55	0.18	102	0.64	0.11	0.26
Saturn	9.14	0.12	3.23	134	0.28	0.22	0.25
Titan	0.40	0.34	0.24	94	0.59	0.10	0.24
Uranus	3.98	0.23	1.91	76	0.46	0.077	0.19
Neptune	3.87	0.30	2.11	72	0.51	0.067	0.18
Titania	0.12	0.31	0.07	60	0.43	0.025	0.10
Enceladus	0.04	0.31	0.02	75	0.32	0.028	0.094
Pluto	0.18	0.37	0.11	40	0.51	0.011	0.075
Triton	0.21	0.38	0.13	38	0.54	0.010	0.074
GJ581g	1.36	1.22	1.51	277	0.90	0.88	0.89
GJ581b	3.97	0.25	1.98	499	0.47	0.36	0.41

```
GJ581c    1.60 1.36 1.87 380 0.85 0.58 0.70
GJ581d    1.60 1.36 1.87 232 0.85 0.64 0.74
GJ581e    1.16 1.10 1.21 591 0.95 0.30 0.53
GJ581f    2.16 0.70 1.80 139 0.79 0.27 0.46
HD69830d  4.19 0.25 2.10 312 0.47 0.77 0.60
55Cncc    5.68 0.25 2.84 310 0.43 0.72 0.56
```

The column headings are Rad= radius, in Earth Units,
Density, also in Earth Units,
EV= Escape Velocity, also in Earth Units,
SurfTemp= Surface Temperature in deg K,
IntESI = Interior ESI,
SurfESI= Surface ESI,
GlobESI= Global ESI, computed from IntESI and SurfESI.
(You need to look at their Appendix A to see how the above variables are computed.)// Can you provide any useful plots and/or data summaries?
You may like to proceed to the data set given in their Table 3, which is headed 'Factors that affect the probability that life could exits on any planetary body, as applied to planetary bodies and satellites in our solar system and to selected exoplanets'.
PHI is the 'Planet Habitability Index' in the penultimate column of the dataset below.
You need to look at their Table 3 for the definitions of their other column headings.

```
Body Solid/Frozen Atmosphere Magnetosphere Light Heat Redox_chemistry
 Tidal_flexing Corg N S P Atmospheric Surface Sub-surface PHI PHIrel
Mercury   1 0 0.1 2 0 0 0 0 0 1 0 0 0 0 0.00 0.00
Venus     1 1 0 2 0 1 0 0.5 1 1 0 0.5 0 0 1.65 0.37
Earth     1 1 1 2 1 2 0.2 2 1 1 1 0.5 2 1 4.37 0.96
Moon      1 0 0 2 0 0 0.2 0 0 1 0 0 0 0 0.00 0.00
Mars      1 0.5 0 2 1 1 0 0.5 0 1 1 0 1 1 2.66 0.59
Ceres     1 0 0.1 1 0 0.5 0 0.5 1 0 0 0 0 0.5 1.05 0.23
Jupiter   0 1 1 1 0 1 0 1 1 1 1 0.5 0 0 1.68 0.37
Io        1 0.1 0.5 1 0 1 1 0.5 0 1 0 0 0.5 0 1.38 0.30
Europa    1 0.1 0.5 1 0 0.5 1 0.5 1 1 0 0 0.5 1 2.22 0.49
Saturn    0 1 1 1 0 1 0 1 1 1 1 0.5 0 0 1.68 0.37
Titan     1 1 1 1 0 1 0 1.5 1 0 0 0.5 2 1 2.89 0.64
Enceladus 1 0.1 0 1 0 0.5 0.5 0.5 1 0 0 0 1 1 1.60 0.35
Uranus    0 1 1 0 0 1 0 1 1 0 0 0.5 0 0 1.19 0.26
Titania   1 0 0 0 0 0.5 0.5 0.5 1 0 0 0 0 0.5 0.93 0.21
Neptune   0 1 1 0 0 1 0 1 1 0 0 0.5 0 0 1.19 0.26
Triton    1 0.1 0.1 0 0 0.5 0 1 1 0 0 0 0.5 0.5 1.05 0.23
Pluto     1 0 0 0 0 0.5 0 1 1 0 0 0 0 0.5 1.00 0.22
GJ581b    0 1 1 2 0 NA 1 NA 1 NA NA 0.5 0 NA 1.19 0.29
GJ581c    1 1 1 2 1 NA 1 NA 1 NA NA 0.5 0.5 NA 1.73 0.41
GJ581d    1 1 1 2 1 NA 0.5 NA 1 NA NA 0.5 0.5 NA 1.57 0.43
GJ581g    1 1 1 2 1 NA 1 NA 1 NA NA 0.5 1 NA 1.73 0.45
HD69830d  0 1 1 2 1 NA 0 NA 1 NA NA 0.5 0 NA 1.32 0.29
55Cncc    0 1 1 1 0 NA 1 NA 1 NA NA 0.5 0 NA 1.32 0.26
```

Thanks are due to Julia Blackwell for help in downloading these datasets.

I wanted to reproduce the authors' Figure 2, which plots Planetary Habitability (relative) against the Earth Similarity Index (ESI) as the x-axis. This is just a little awkward, since the first datatable has 27 rows, but the second has 23 rows, in a slightly different order. I found that the simplest thing to do was to edit the first table to remove the 4 'extra' rows, and then proceed as follows: a good revision of the R command order(). This gives Figure 27.1: you can probably improve the presentation here.

```
x <- read.table("redtable1", header=T) ; y <- read.table("edtable3", header=T)
# each of x and y has 23 rows, since I've removed 4 rows from the first table.
attach(x) ; attach(y)
i <- order(Planet) ; Planet[i]
j <- order(Body) ; Body[j]
Y <- PHIrel[j] ; X <- GlobESI[i]
name <- Planet[i]
plot(X,Y,type="n", xlab="Earth Similarity (ESI)", ylab= "Planetary Habitability (PHIrel)")
text(X,Y, label=name)
```

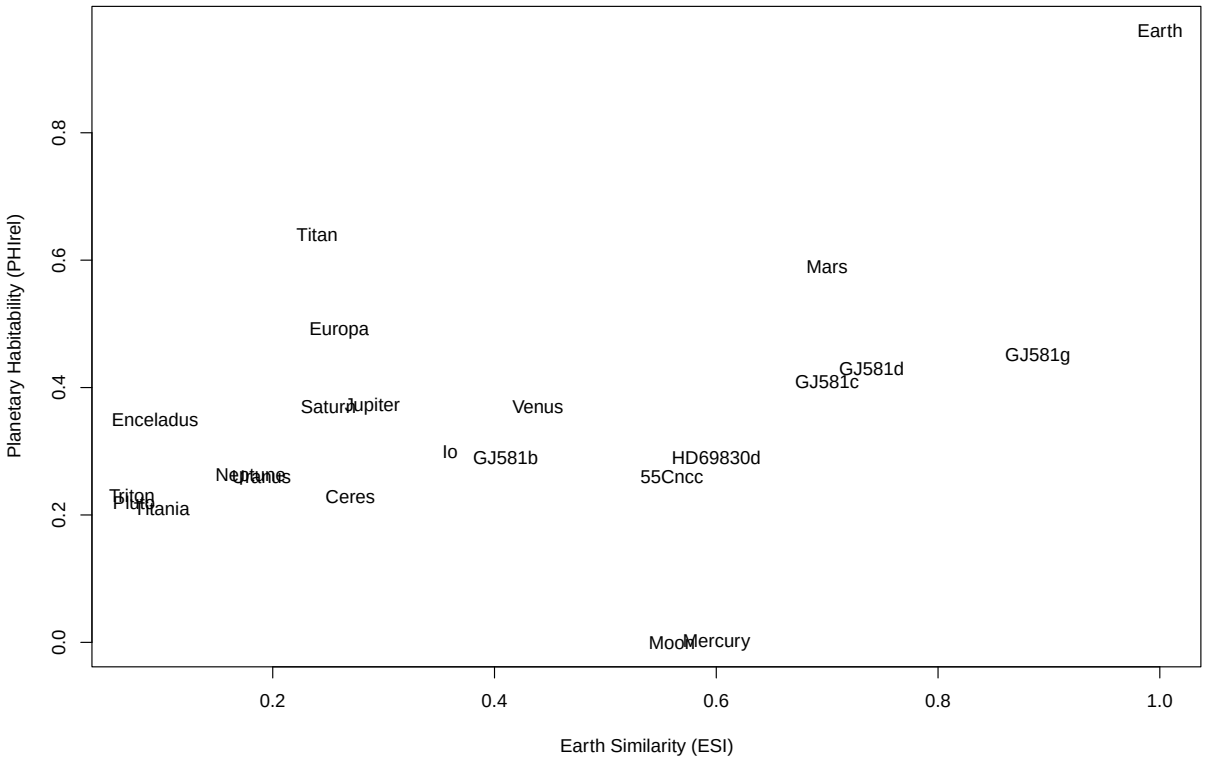


Figure 27.1: How does Planetary Habitability depend on the Earth Similarity Index?

Chapter 28

A 4-dimensional contingency table, revisited

This is a new analysis of a 2^4 table, formed from the dataset given as Table 28.1 on the incidence of schizophrenia in one or both members of sib-pairs hospitalized for mental disorder. I first encountered this dataset in 1975, through the visit of Professor Joel E. Cohen to the Statistical Laboratory. He published a paper using the dataset as an example in 1976: that paper is 'The distribution of the chi-squared statistic under clustered sampling from contingency tables' *J. Amer Statist. Ass.* 1976, **71** 665-670, and the dataset itself comes from a paper by M-T. Tsuang 'A study of pairs of sibs both hospitalized for mental disorder" *Brit. J. Of Psychiatry'* **113** (1967), 283-300.
When I encountered the dataset in 1975, I was completely unaware of the possibilities of modelling the four-way table which are so easily achieved in *R* via glm, and I do think that the modelling approach given below is more enlightening than the 1976 emphasis on hypothesis-testing.
Here is the dataset, which we may first view as a 4×4 contingency table, Table 28.1. Each person of each pair is classified into one of 4 categories, namely SM, SM, NM, NF, respectively, where
S/N refers to schizophrenic/ non-schizophrenic, and
M/F refers to male/female.

	SM	SF	NM	NF
elder sib SM	13	5	1	3
elder sib SF	4	6	1	1
elder sib NM	1	1	2	4
elder sib NF	3	8	3	15

Table 28.1: The dataset used by Joel E. Cohen; the columns are the status of the younger sibs

```
n = scan("JoelCohendata")
n
13 5 1 3 4 6 1 1 1 1 2 4 3 8 3 15

a <- matrix(n, nrow=T, byrow=T)
chisq.test(a) ; fisher.test(a, simulate.p.value=T)
```

This confirms what we can see at a glance from Table 28.1: that the (schizophrenia, sex) status of the younger sib is NOT independent of the (schizophrenia, sex) status of his/her elder sib. In particular there is an excess of sib-pairs which are (SM,SM) and likewise an excess of sib-pairs which are (NF,NF). We can in fact model this dependence more informatively, using the four 2-level variables
elder sib schizophrenic (yes/no)
elder sib sex (Male/Female)
younger sib schizophrenic (yes/no)
younger sib sex (Male/Female).

```
elderschiz  <- gl(2,8, length=16, labels= c("S", "N"))
eldersex    <- gl(2,4, length=16, labels= c("M", "F"))
youngerschiz<- gl(2,2, length=16, labels= c("S", "N"))
youngersex  <- gl(2,1, length=16, labels= c("M", "F"))
library(MASS)
big.glm <- glm(n ~ (elderschiz+eldersex+youngerschiz+youngersex)^3, poisson)
summary(big.glm)
stepAIC(big.glm)
```

At this stage I tried out various models, with my preferred final model as one with just 3 pairwise interactions, namely

```
Last.glm <- glm(n ~ elderschiz + eldersex + youngerschiz + youngersex +
+  eldersex:youngersex + elderschiz:eldersex +
+elderschiz:youngerschiz, family = poisson)
summary(Last.glm)
```

This model has residual deviance 7.7091 on 8 degrees of freedom.
How can we interpret this model? For convenience, let us write the 4 factors elderschiz, eldersex, youngerschiz, youngersex as A, B, C, D respectively, and p_{ABCD} as the corresponding cell probability. Then with our final model

$$p_{ABCD} = \alpha_{AB}\beta_{AC}\gamma_{BD}$$

for some positive $\alpha_{AB}, \beta_{AC}, \gamma_{BD}$ such that $\sum p_{ABCD} = 1$. Thus we can derive the following 3 2×2 tables as the sufficient statistics for the data.

```
tapply(n,list(elderschiz, eldersex), sum)
   M  F
S 22 12
N  8 29

tapply(n,list(elderschiz,youngerschiz),sum)
   S  N
S 28  6
N 13 24

tapply(n,list(eldersex,youngersex), sum)
  M  F
M 17 13
F 11 30
```

Hence, for example, in this sample of 71 hospitalized sib-pairs, there are 13 sib-pairs in which the elder is male and the younger is female.
Each of the 3 tables given above shows marked positive dependence.
Further, our final model tells us the following about the conditional probabilities of the younger sibling being (C, D), given that the elder sibling was (A, B):

$$p(CD|AB) = \frac{\beta_{AC}\gamma_{BD}}{\sum_{C'}\beta_{AC'}\sum_{D'}\gamma_{BD'}}.$$

So, conditional on (A, B), the variables C, D are independent, with C only depending on A and D only depending on B.

Chapter 29

World Oil: proved reserves for 53 countries

Having seen many newspaper references to 'World Energy', I was curious to see what datasets are publicly available.
From the BP website, `http://www.bp.com/`, 'Statistical Review of World Energy' I was able to download relevant data as a .xls file, namely

`Statistical_Review_of_World_Energy_2010.xls`

from which I saved the subset given below, as a .csv file, which can therefore be read directly into R.
But: WARNING: other websites refer to 'World Oil Reserves: The Problem of Reliable Data', so it is very likely that more research on the web is needed for better data.
Here is the BP table of 'Oil:Proved Reserves' as slightly edited by me.
First, an explanation of the column headings. There are 53 countries in all, and the last row of the dataset is 'Total World', presented here as a check. So you will need to delete this row before you carry out plots, summaries, regressions etc.
The columns 'end1989','end1999','end2008', 'barrels09' give the Proved reserves of oil, in Thousand million barrels, at the ends of 1989, 1999, 2008, 2009 respectively, while 'tonnes09' gives the the amount of oil in Thousand million tonnes, at the end of 2009.
The penultimate column, 'share' is the share of the total proved reserve for each country, as a percentage of the Total World proved reserves at the end of 2009.
The final column, 'R.to.P ratio', is the Reserves to Production ratio. Thus 'If the reserves remaining at the end of any year are divided by the production in that year, the result is the length of time that those remaining reserves would last if production were to continue at that rate'.
So the US Reserves to Production ratio of 10.8 means at the current production rate, the oil reserves of the US will run out in 10.8 years time.
Venezuela, Iraq, Kuwait and the United Arab Emirates all give a Reserves to Production ratio which exceeds 100 years, in these cases I have just written 100 as the corresponding entry, so this is not quite correct.

```
"Country","end1989","end1999","end2008","tonnes09","barrels09", "share",
 "R.to.P ratio"
"US",34.3,29.7,28.4,3.4,28.4,2.1 ,10.8
"Canada",11.6,18.3,33.2,5.2,33.2,2.5 ,28.3
"Mexico",52.0,21.5,11.9,1.6,11.7,0.9 ,10.8
"Argentina",2.2,3.1,2.5,0.3,2.5,0.2 ,10.2
"Brazil",2.8,8.2,12.8,1.8,12.9,1.0 ,17.4
"Colombia",2.0,2.3,1.4,0.2,1.4,0.1 ,5.4
"Ecuador",1.4,4.4,6.5,0.9,6.5,0.5 ,36.1
"Peru",0.8,0.9,1.1,0.2,1.1,0.1 ,21.1
"Trinidad & Tobago",0.6,0.8,0.8,0.1,0.8,0.1 ,15.1
"Venezuela",59.0,76.8,172.3,24.8,172.3,12.9 ,100
"Other S. & Cent. America",0.6,1.3,1.4,0.2,1.4,0.1 ,26.8
"Azerbaijan",NA,1.2,7.0,1.0,7.0,0.5 ,18.6
"Denmark",0.6,0.9,0.8,0.1,0.9,0.1 ,9.5
"Italy",0.8,0.9,1.0,0.1,0.9,0.1 ,27.2
"Kazakhstan",NA,25.0,39.8,5.3,39.8,3.0 ,64.9
"Norway",8.4,10.9,7.5,0.9,7.1,0.5 ,8.3
"Romania",1.2,1.2,0.5,0.1,0.5,0,14.2
"Russian Federation",NA,59.2,74.3,10.2,74.2,5.6 ,20.3
"Turkmenistan",NA,0.5,0.6,0.1,0.6,0,8.0
"United Kingdom",3.8,5.0,3.1,0.4,3.1,0.2 ,5.8
"Uzbekistan",NA,0.6,0.6,0.1,0.6,0,15.2
"Other Europe & Eurasia",69.4,2.3,2.1,0.3,2.2,0.2 ,14.9
"Iran",92.9,93.1,137.6,18.9,137.6,10.3 ,89.4
"Iraq",100.0,112.5,115.0,15.5,115.0,8.6 ,100
"Kuwait",97.1,96.5,101.5,14.0,101.5,7.6 ,100
"Oman",4.3,5.7,5.6,0.8,5.6,0.4 ,18.9
"Qatar",4.5,13.1,26.8,2.8,26.8,2.0 ,54.7
"Saudi Arabia",260.1,262.8,264.1,36.3,264.6,19.8 ,74.6
"Syria",2.0,2.3,2.5,0.3,2.5,0.2 ,18.2
"United Arab Emirates",98.1,97.8,97.8,13.0,97.8,7.3 ,100
"Yemen",2.0,1.9,2.7,0.3,2.7,0.2 ,24.5
"Other Middle East",0.1,0.2,0.1,0,0.1,0,9.4
"Algeria",9.2,11.3,12.2,1.5,12.2,0.9 ,18.5
"Angola",2.1,5.1,13.5,1.8,13.5,1.0 ,20.7
"Chad",NA,NA,0.9,0.1,0.9,0.1 ,20.9
"Rep. of Congo (Brazzaville)",0.7,1.7,1.9,0.3,1.9,0.1 ,19.4
"Egypt",4.3,3.8,4.2,0.6,4.4,0.3 ,16.2
"Equatorial Guinea",NA,0.6,1.7,0.2,1.7,0.1 ,15.2
"Gabon",1.0,2.6,3.7,0.5,3.7,0.3 ,44.1
"Libya",22.8,29.5,44.3,5.8,44.3,3.3 ,73.4
"Nigeria",16.0,29.0,37.2,5.0,37.2,2.8 ,49.5
"Sudan",0.3,0.3,6.7,0.9,6.7,0.5 ,37.5
"Tunisia",1.8,0.3,0.6,0.1,0.6,0,18.4
"Other Africa",0.9,0.7,0.6,0.1,0.6,0,11.0
"Australia",3.1,4.7,4.2,0.5,4.2,0.3 ,20.7
```

```
"Brunei",1.2,1.3,1.1,0.1,1.1,0.1 ,17.6
"China",16.0,15.1,14.8,2.0,14.8,1.1 ,10.7
"India",4.3,5.0,5.8,0.8,5.8,0.4 ,21.1
"Indonesia",5.1,5.2,3.7,0.6,4.4,0.3 ,11.8
"Malaysia",3.7,5.0,5.5,0.7,5.5,0.4 ,20.4
"Thailand",0.2,0.4,0.5,0.1,0.5,0,3.8
"Vietnam",0.1,1.8,4.7,0.6,4.5,0.3 ,35.7
"Other Asia Pacific",0.9,1.4,1.4,0.2,1.3,0.1 ,11.2
"Total World",1006.4,1085.6,1332.4,181.7,1333.1,100.0 ,45.7
```

Exercise: read in the dataset into a table, and set up

```
rrx
```

as a data frame, being the original dataset with the Countries as the row names, and omitting the redundant Tonnes column, and also omitting the final 'Total world' row, thus providing a dataframe with 53 rows in total. Now summarise the data in any way that seems suitable. You might for example like to provide a starsplot, such as the one I have given as Figure 29.1 for the first 9 rows of the dataframe. (If you also include Venezuela, it tends to dominate the picture, no doubt appropriately.)

```
stars(rrx[1:9,],ncol=4, len = 0.7, key.loc = c(8,2.5), draw.segments = TRUE)
```

You might also like to do a simple 'time tracks' plot, as in Figure 29.2. Here's one way to proceed.

```
z <- as.matrix(rrx[1:9, 1:4]) ; z <- t(z) ; Year <-c(1989,1999,2008,2009)
matplot(Year, z, type="b",ylim=c(0,70), ylab="proved reserves of oil,
 in thousand million barrels")
# the 'type="b" option means that we get Both lines and points together
```

It is not clear that formal statistical modelling, eg regressions, tests etc, would be relevant to this dataset, but you could try including the Oil consumption figures (available also on BP website) and see where that gets you.

May 2012. Here is another topical data set, from the BBC Business news, May 4, 2012. (I downloaded the data as an .xls file and converted it to a .csv file, using my Desktop machine.) It's easy to read it into R as follows

```
x <- read.table("eurozone_gdp.csv", header=T, sep=",")
```

Here is the financial data: you should start by plotting the timetrack for each row (ie country)

```
#GDP at market prices for the eurozone and the UK
#Annual figures - percentage change from previous period
#
"Country","1999","2000","2001","2002","2003","2004","2005","2006","2007","2008","2009","2010","201
"Belgium",3.5,3.7,0.8,1.4,0.8,3.3,1.7,2.7,2.9,1.0,-2.8,2.3,1.9
"Germany",1.9,3.1,1.5,0.0,-0.4,1.2,0.7,3.7,3.3,1.1,-5.1,3.7,3.0
"Estonia",-0.3,9.7,6.3,6.6,7.8,6.3,8.9,10.1,7.5,-3.7,-14.3,2.3,7.6
```

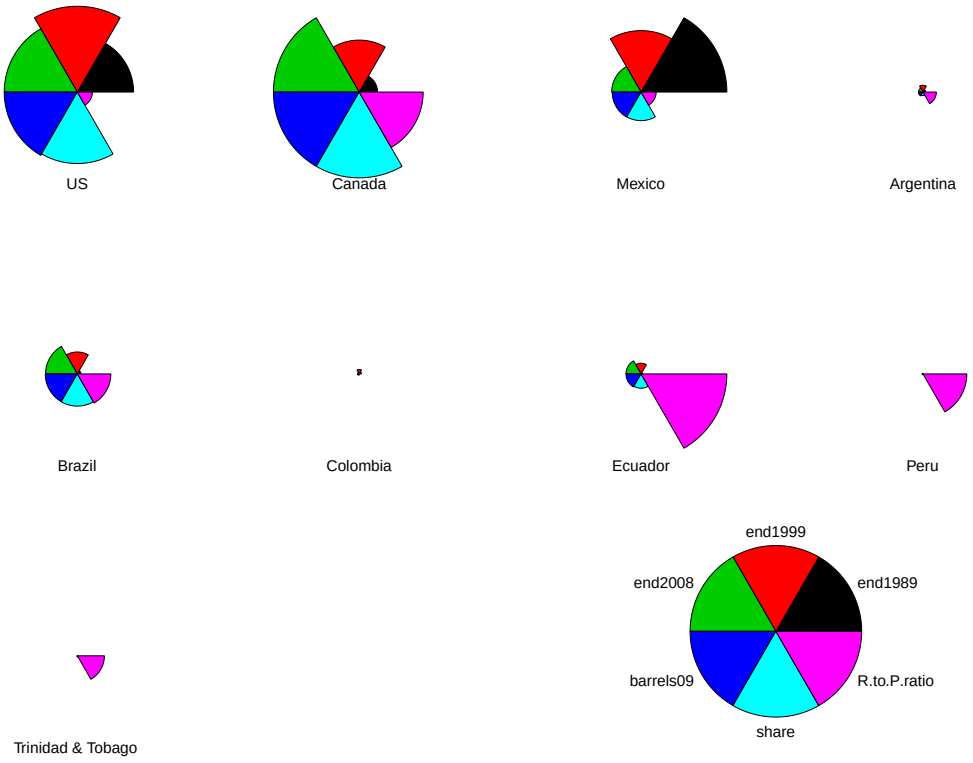


Figure 29.1: A stars plot for the Proved Oil Reserves of the first 9 countries in the Americas

```
"Ireland",9.9,9.3,4.8,5.9,4.2,4.5,5.3,5.3,5.2,-3.0,-7.0,-0.4,0.7
"Greece",3.4,3.5,4.2,3.4,5.9,4.4,2.3,5.5,3.0,-0.2,-3.3,-3.5,-6.9
"Spain",4.7,5.0,3.7,2.7,3.1,3.3,3.6,4.1,3.5,0.9,-3.7,-0.1,0.7
"Spain",4.7,5.0,3.7,2.7,3.1,3.3,3.6,4.1,3.5,0.9,-3.7,-0.1,0.7
"France",3.3,3.7,1.8,0.9,0.9,2.5,1.8,2.5,2.3,-0.1,-2.7,1.5,1.7
"Italy",1.5,3.7,1.9,0.5,0.0,1.7,0.9,2.2,1.7,-1.2,-5.5,1.8,0.4
"Cyprus",4.8,5.0,4.0,2.1,1.9,4.2,3.9,4.1,5.1,3.6,-1.9,1.1,0.5
"Luxembourg",8.4,8.4,2.5,4.1,1.5,4.4,5.4,5.0,6.6,0.8,-5.3,2.7,1.6
"Malta",NA,NA,-1.5,2.8,0.1,-0.5,3.7,2.9,4.3,4.1,-2.7,2.3,2.1
"Netherlands",4.7,3.9,1.9,0.1,0.3,2.2,2.0,3.4,3.9,1.8,-3.5,1.7,1.2
"Austria",3.5,3.7,0.9,1.7,0.9,2.6,2.4,3.7,3.7,1.4,-3.8,2.3,3.1
"Portugal",4.1,3.9,2.0,0.8,-0.9,1.6,0.8,1.4,2.4,0.0,-2.9,1.4,-1.6
"Slovenia",5.3,4.3,2.9,3.8,2.9,4.4,4.0,5.8,6.9,3.6,-8.0,1.4,-0.2
"Slovakia",0.0,1.4,3.5,4.6,4.8,5.1,6.7,8.3,10.5,5.8,-4.9,4.2,3.3
"Finland",3.9,5.3,2.3,1.8,2.0,4.1,2.9,4.4,5.3,0.3,-8.4,3.7,2.9
"United Kingdom",3.7,4.5,3.2,2.7,3.5,3.0,2.1,2.6,3.5,-1.1,-4.4,2.1,0.7
```

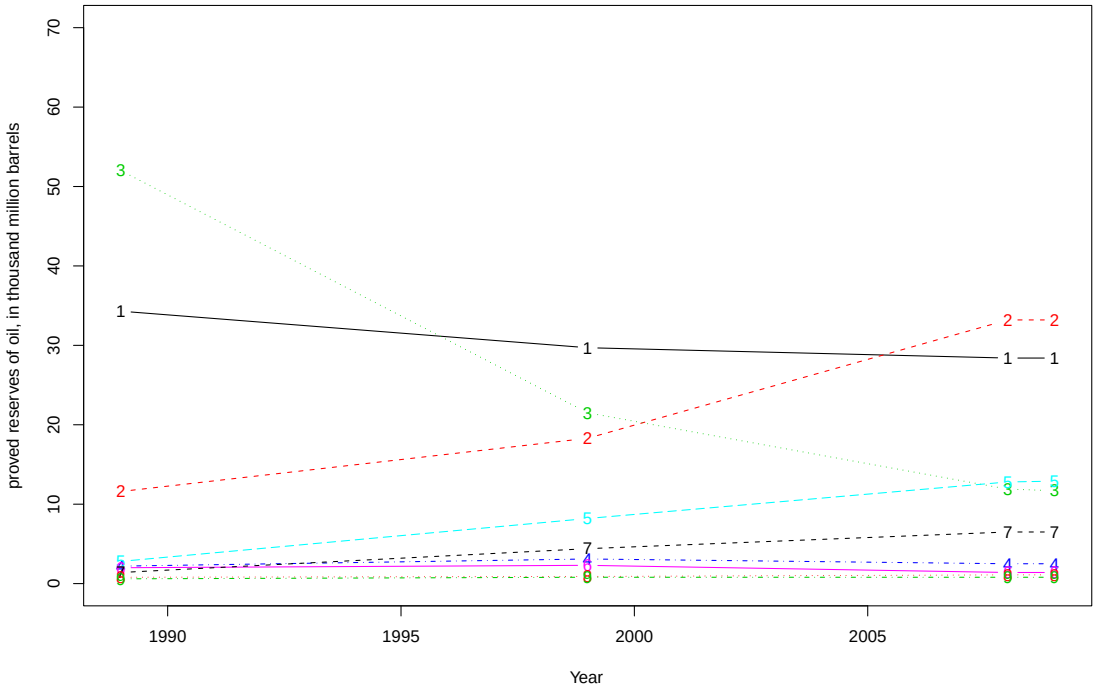


Figure 29.2: A 'time tracks' plot for the Proved Oil Reserves of the first 9 countries in the Americas

Chapter 30

A multiplicative generalization of the binomial distribution

In his book 'Discrete Distributions: Applications in the Health Sciences' (Wiley 2004) Daniel Zelterman discusses the 'Altham distribution', which I introduced in 1978 as an extension of the binomial distribution which could allow for under- or over-dispersion relative to the binomial.
First some motivation. Let Y_n be the number of affected individuals in a family of size n, so that we can take

$$Y_n = X_1 + \cdots + X_n$$

where X_i has values 0 or 1, and is the response of the ith member of the family of size n. We assume that $X_1, \ldots, X_n$ have a joint distribution which is symmetrical, but we allow for dependence between the $X_i's$ by taking the log-linear model

$$\log Pr(X_1 = x_1, \ldots X_n = x_n) = \sum \alpha_{x_i, x_j}$$

where $\alpha_{ij} = \alpha_{ji}$ and the summation extending over $1 \leq i, j \leq n$.
Then it can be shown (see Altham (1978)) that the frequency function of Y_n has the form

$$Pr(Y_n = i) \propto \binom{n}{i} p^i (1-p)^{n-i} \exp[-i(n-i)\theta]$$

for parameters $\theta > 0$ and p, where $0 < p < 1$. Thus if $\theta = 0$ we recover the usual binomial $Bi(n, p)$, while $\theta > 0, \theta < 0$ corresponds respectively to the $X_i's$ being positively, negatively co-dependent. Positive dependence will result in the over-dispersion of the distribution of Y_n relative to the pure binomial.
(This uses Zelterman's notation: see his equation (7.3) on p175 of his book).
From the point of view of fitting this model, the fact that the normalization constant can only be written in terms of a sum presents no problem. This follows from the convenient connection with glm() and the Poisson distribution with a log-linear model: see for example Altham and Lindsey (1998).
In June 2011, Daniel Zelterman kindly sent me his Sastry Brazilian Families data: this dataset on deaths of children in northeast Brazil was reported by N.Sastry in 1997.
I was able to fit multiplicative binomials, one for each family size, using the R commands shown

below.
Zelterman actually fitted the "Altham" distribution with the **same** θ parameter to all family sizes (see for example his p222). However for a given family size n, the θ parameter has an interpretation in terms of **conditional** crossratios. These are derived from

$$Pr(x_1, x_2 | x_3, \ldots, x_n).$$

Thus his use of a model with the same θ for all family sizes does not make sense.
Note that simple calculations for this dataset show that the probability that an individual in a family is affected varies with the family size, rising from .045 for families of size 1 to .292 for families of size 8. The extent to which individuals in a family are co-dependent also changes with the family size, so that for example individuals in a family of size 2 are more closely associated than individuals in a family of size 8.
Here is the Sastry data set.

```
famsize i y
1    0 255
1    1  12
2    0  239
2    1  44
2    2  2
3    0  143
3    1  41
3    2  15
3    3   3
4    0   69
4    1   30
4    2   9
4    3   2
4    4   0
5    0   43
5    1   34
5    2    15
5    3    9
5    4    3
5    5    0
6    0    15
6    1    18
6    2    8
6    3    5
6    4    3
6    5    0
6    6    1
7    0    4
7    1    4
7    2    7
7    3    4
7    4    2
```

```
7    5    0
7    6    0
7    7    0
8    0    1
8    1    2
8    2    4
8    3    3
8    4    1
8    5    1
8    6    0
8    7    0
8    8    0
```

Thus for example there are 30 families of size 4 in which exactly 1 of the siblings is affected.

Here is a program to fit multiplicative binomial models simultaneously to all family sizes (ie with one pair of parameters for each family size from $1, \ldots, 8$).

```
Z.data = read.table("zelterman.data", header=T)
attach(Z.data)
Off = lchoose(famsize,i) # to set up the term we will use as offset
x2 = i*(famsize-i)
Famsize = factor(famsize)
```

First we fit a binomial distribution, with the same parameter p, to all 8 family sizes, as a baseline.

```
> first.glm = glm(y ~ i + Famsize + offset(Off), poisson) ; summary(first.glm)
```

Note that including the additive term "Famsize", which is a factor, ensures that we get perfect agreement between the sum of observed and the sum of fitted for each family size.
We don't expect this baseline model to fit, and indeed as Zelterman shows (p222) we get a deviance of 152.10 with 35 df.
Now we'll gradually increase the complexity of the model.

```
next.glm = glm(y ~ i + x2 + Famsize + offset(Off), poisson) ; summary(next.glm)
 fv1 = first.glm$fitted.values
fv2 = next.glm$fitted.values
nnext.glm = glm(y ~ i*Famsize + x2*Famsize + Famsize + offset(Off), poisson)
summary(nnext.glm)
```

This fits a different Altham distribution for each of the 8 family sizes, so this model fits very well, as we might expect. The residual deviance is 10.599 with 21 df.
As we might expect, the families of size $1, 2$ are not actually contributing to this final model (we would get a perfect fit for each frequency in family size $1, 2$) and this you can check by

```
subnnext.glm = glm(y ~ i*Famsize + x2*Famsize + Famsize + offset(Off),
poisson, subset= (famsize>2))
```

and in particular, this final model will have exactly the same residual deviance, namely 10.599 with 21 df.

```
fv3 = nnext.glm$fitted.values
```

Now we verify that the observed and expected values of the sufficient statistics agree exactly for each family size, as the theory (for maximum likelihood estimation in exponential families) tells us.

```
tapply(i*fv3, Famsize, sum)
 1   2   3   4   5   6   7   8
12  48  80  54 103  66  38  28

tapply(i*y, Famsize, sum)
 1   2   3   4   5   6   7   8
12  48  80  54 103  66  38  28

tapply(i*i*fv3, Famsize, sum)
 1   2   3   4   5   6   7   8
12  52 128  84 223 168 100  86

tapply(i*i*y, Famsize, sum)
 1   2   3   4   5   6   7   8
12  52 128  84 223 168 100  86
```

Chapter 31

Fitting a Bivariate multiplicative binomial probability distribution

This example illustrates the paper by Altham and Hankin
"Multivariate generalizations of the multiplicative binomial distribution: Introducing the MM Package". This paper has been accepted for publication in the J. of Statistical Software.
For the draft version, see `http://www.statslab.cam.ac.uk/~pat/Gianfranco.pdf`.

Here I extend the multiplicative binomial, which was introduced in Altham (1978) to a bivariate version. Consider (X_1, X_2) as non-negative integers, where

$$P(X_1 = x_1, X_2 = x_2) = p(x_1, x_2)$$

for $0 \leq x_1 \leq k_1$ and $0 \leq x_2 \leq k_2$. (In the example below, $k_1 = 7$ and $k_2 = 3$.)
We define a distribution which is of exponential family form, and contains the following possibilities as special cases
i) (X_1, X_2) independent, each of multiplicative binomial form,
ii) (X_1, X_2) independent, each of binomial form.
We introduce a 5-parameter distribution, which has the property that at the maximum likelihood values of these parameters, the observed and fitted values of the means of X_1 will agree exactly, similarly for X_2, similarly for the observed and fitted values of the covariance matrix of X_1, X_2.
This distribution is extremely easy to fit to frequency data (using the Lindsey Poisson device), and the distribution has some nice properties, but there are no simple formulae for its moments.
Here is the proposed frequency function.

$$p(x_1, x_2) = C^{-1} \binom{k_1}{x_1} p_1^{x_1} q_1^{k_1 - x_1} \binom{k_2}{x_2} p_2^{x_2} q_2^{k_2 - x_2} \phi^{x_1 x_2} \theta_1^{x_1(k_1 - x_1)} \theta_2^{x_2(k_2 - x_2)},$$

where $p_1 + q_1 = 1, p_2 + q_2 = 1$, and all parameters are strictly positive.
Remarks.
1. C is the normalization constant, and has to be evaluated as a sum over x_1, x_2.
2. X_1, X_2 are independent if and only if $\phi = 1$.
3. Furthermore, if $\phi = 1$, then $\theta_1 = 1$ and $\theta_2 = 1$ corresponds to X_1, X_2 independent Binomial

$Bi(k_1, p_1), Bi(k_2, p_2)$ respectively.
4. It seems impossible to write down simpled closed form expressions for the expectations of X_1, X_2 etc, but a little progress can be made for certain special cases, eg $\theta_1 = 1$ and $\theta_2 = 1$, $\phi = 1 + \delta$, where δ is small.
Although it is not possible to give a simple expression say for the correlation between X_1, X_2, it is easily seen that ϕ controls their interdependence in a likelihood ratio fashion, thus

$$\frac{p(x_1, x_2)p(x_1 + 1, x_2 + 1)}{p(x_1 + 1, x_2)p(x_1, x_2 + 1)} = \phi.$$

If $\phi > 1$, then X_1, X_2 are positive likelihood ratio dependent, if $\phi < 1$ they are negative likelihood ratio dependent. For the properties of likelihood ratio dependence, see Lehmann (1966) who introduced this concept.
5. The conditional distribution say $X_1|X_2 = x_2$ will again be of multiplicative binomial form, since we can write

$$p(x_1|x_2) \propto \binom{k_1}{x_1} (p_1\phi^{x_2})^{x_1} q_1^{k_1 - x_1} \theta_1^{x_1(k_1 - x_1)}.$$

6. The minimal sufficient statistics for this distribution will essentially be the sample means of (x_1, x_2) and their covariance matrix.
The exponential family form of this model has the consequence that at the mle's of the parameters, the observed and fitted values of these sufficient statistics will exactly match one another. (This is a standard result for exponential families.)
7. Now we can easily see how to generalize the definition to higher dimensions, eg for a 3-variate distribution:

$$p(x_1, x_2, x_3) = C^{-1} \prod_{1 \le i \le 3} \binom{k_i}{x_i} p_i^{x_i} q_i^{k_i - x_i} \theta_i^{x_i(k_i - x_i)} \prod_{1 \le i < j \le 3} \phi_{ij}^{x_i x_j}$$

for $0 \le x_i \le k_i, 1 \le i \le 3$.
Exercise: find the form of

$$p(x_1, x_2|X_3 = x_3)$$

and show that the conditional dependence of $(x_1, x_2|X_3 = x_3)$ is governed by ϕ_{12}. (Compare this result with the trivariate normal distribution.)

References

Altham, PME. Two generalisations of the binomial distribution. *Applied Statistics* **27**, (1978), 162–167.
Altham, PME and Lindsey, JK. Analysis of the human sex ratio using overdispersion models. *Applied Statistics*. **47**, (1998), 149–157.
Lehmann, EL. Some concepts of dependence. *Ann Math Stat.* **37**, (1966) 1137–1153.
Lindsey, JK and Mersch, G. Fitting and comparing probability distributions with log linear models. *Comput. Statist. Data Anal.* **13**, (1992), 373–384.

Practical example: fitting the new distribution to data: does having more children under 5 years old make it harder to publish papers?
Here is our example of the new distribution showing a negative dependence between the 2 variables of interest. Zeileis, Kleiber and Jackman in their R package

`pcsl`

give as an example dataset various counts for each of a total of 915 biochemistry graduate students, from whom I selected the subsample which consisted of all males who had published 7 articles or less, and also had 3 or less children under 5 years old. This gave a total of of 486 men. Thus in our notation given above, we take
x_1 is number of articles published (7 or under) by each man, and
x_2 is number of children under 5 (3 or under) for each man.
(I am using the data for MEN; I tried the analysis first for women, but the negative dependence was not significant..... contrary to my prejudice!)

```
R
library(pscl, lib.loc="~/Rlibs")
data(bioChemists)
attach(bioChemists)
table(art[fem=="Men"], kid5[fem=="Men"])
```

We use only men who have published 7 articles or less.

```
n=scan()
    63 40 26  7
    69 42 19  5
    48 21 16  2
    33 11  9  1
    21  8  9  0
     7  8  1  0
     5  3  1  0
     9  2  0  0

x2 <- rep(0:3, times=8)
x1 <- gl(8, 4,length= 32 , labels=1:8)
x1 <- as.numeric(x1) ; x1 = x1 -1
# x1 is number of articles published (7 or under)
# x2 is number of children under 5   ( 3 or under)
> tapply(x1*n, x2, sum)/tapply(n, x2, sum)
       0        1        2        3
1.866667 1.637037 1.543210 0.800000
```

This shows that the MEAN number of articles produced decreases as the number of children under 5 increases.

```
Off <- lchoose(7,x1) + lchoose(3,x2)
y1 <- x1*(7-x1) ; y2 = x2*(3-x2)
first.glm <- glm(n ~ x1 + x2 + x1:x2 + y1 + y2 + offset(Off), poisson)
summary(first.glm)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9575  -0.8002  -0.3091   0.4648   2.4444
```

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.17575    0.10059  41.513  < 2e-16 ***
x1          -0.40122    0.03993 -10.049  < 2e-16 ***
x2          -0.70815    0.08282  -8.550  < 2e-16 ***
y1          -0.26094    0.01347 -19.375  < 2e-16 ***
y2          -0.37381    0.05906  -6.330 2.46e-10 ***
x1:x2       -0.08431    0.03403  -2.478   0.0132 *
---

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 1688.368  on 31  degrees of freedom
Residual deviance:   31.837  on 26  degrees of freedom
AIC: 155.44

Number of Fisher Scoring iterations: 5
```

Thus our model is not a bad fit. It shows significant negative dependence between X_1, X_2, since $\log(\hat{\phi}) = -0.08431(0.03403)$, so that $\hat{\phi} = 0.9191$.

```
fp <- first.glm$fitted.values/486
```

These are the fitted probabilities of the bivariate distribution.

```
> sum(x1*x2*n)/486
[1] 1.043210
> sum(x1*x2*fp)
[1] 1.043210
```

This provides a partial check of our statement that the observed and fitted values of the sufficient statistics will agree exactly.

```
 Fp <- matrix(fp,nrow=8, ncol=4, byrow=T))
 round(Fp,3)
      [,1]  [,2]  [,3]  [,4]
[1,] 0.134 0.094 0.046 0.016
[2,] 0.131 0.084 0.038 0.012
[3,] 0.093 0.055 0.023 0.007
[4,] 0.061 0.033 0.013 0.003
[5,] 0.041 0.021 0.007 0.002
[6,] 0.028 0.013 0.004 0.001
[7,] 0.018 0.007 0.002 0.000
[8,] 0.008 0.003 0.001 0.000
```

Note that within each column, the fitted probability decreases with row number (and correspondingly the same is true for each row).

```
interaction.plot(x1,x2,fp)
```

This provides a quick plot of the shape of the new distribution, shown in Figure 31.1.

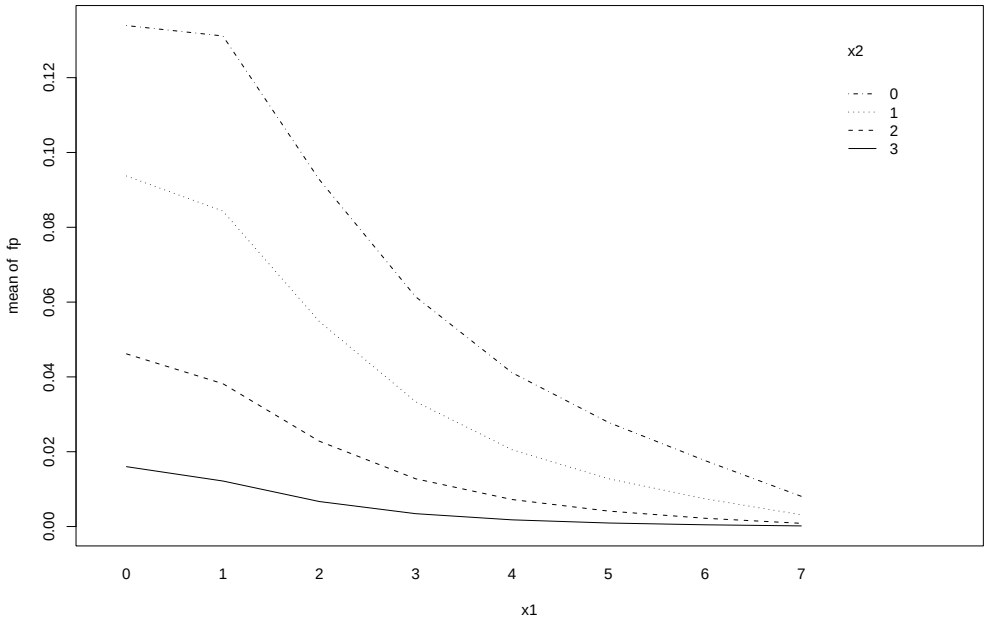


Figure 31.1: The fitted probabilities for the bivariate binomial, with x_1, x_2 as the number of publications, number of children under 5, respectively.

Here is a simple set of R commands to enable us to compare the effects of varying the 5 parameters of the distribution, when $k_1 = k_2 = 20$.
NB. I find that I need to type the formula in the function definition ALL ON ONE LINE, which is not how it's printed below (for reasons of page-width).

```
k=20
x1 = seq(0,20) ; x2 = x1
th1 =1 ; th2 = 1; p1 = .5 ; p2 = .5
q1 = 1- p1 ; q2  = 1 - p2 ; phi =  1
# first we generate 2 independent binomials
bivbin = function(x1,x2){
lchoose(k,x1)+ lchoose(k,x2) +x1*log(p1/q1) +x2*log(p2/q2)
 + x1*(k-x1)*log(th1)+x2*(k-x2)*log(th2)+x1*x2*log(phi)
}
z = x1%*%t(x2)
for (i in 1:21){
 for (j in 1:21){
  z[i,j] = exp( bivbin(x1[i],x2[j]))
 }
}
N = sum(z) ; z = z/N
par(mfrow=c(3,2))
contour(x1, x2, z)
persp(x1,x2,z)
# Now we introduce dependence beween x1, x2
phi = .94
bivbin = function(x1,x2){
lchoose(k,x1)+ lchoose(k,x2) +x1*log(p1/q1) +x2*log(p2/q2)
 + x1*(k-x1)*log(th1)+x2*(k-x2)*log(th2)+x1*x2*log(phi)
}
z = x1%*%t(x2)
for (i in 1:21){
 for (j in 1:21){
  z[i,j] = exp( bivbin(x1[i],x2[j]))
 }
}
N = sum(z) ; z = z/N
contour(x1, x2, z)
persp(x1,x2,z)
# Now set th1, th2 to be different from 1
th1 = .95; th2 =  .96
bivbin = function(x1,x2){
lchoose(k,x1)+ lchoose(k,x2) +x1*log(p1/q1) +x2*log(p2/q2)
 + x1*(k-x1)*log(th1)+x2*(k-x2)*log(th2)+x1*x2*log(phi)
}
z = x1%*%t(x2)
for (i in 1:21){
 for (j in 1:21){
```

```
  z[i,j] = exp( bivbin(x1[i],x2[j]))
 }
}
N = sum(z) ; z = z/N
contour(x1, x2, z)
persp(x1,x2,z)
```

Finally, here is a 'new' dataset, for which a bivariate distribution in the same family fits very well. The dataset comes from Table I , p424 of 'Accident Statistics and the Concept of Accident-Proneness' by A.G.Arbous and J.E.Kerrich, Biometrics vol 7, 1951.
Table I: Accidents among 122 experienced shunters

```
21 18 8 2 1 0 0
13 14 10 1 4 1 0
4 5 4 2 1 0 1
2 1 3 2 0 1 0
0 0 1 1 0 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0 0
0 1 0 0 0 0 0
```

There are 8 rows, corresponding to x1 = 0, 1,..., 7 accidents, which took place in the years 1943-1947. The 7 columns correspond to x2= 0,1,..,6 accidents, which took place in the years 1937-1942. Thus you see, for example, that of the total of 122 shunters, exactly 18 had 1 accident in the first time period and 0 accident in the second time period.
I find that fitting the full model we get a deviance of 32.209 on 50 degrees of freedom (note: there are 56 entries in the 8×7 table, and we are fitting 6 parameters.) There is strong positive association between x1, x2.

Chapter 32

Fitting a segmented line: the RAE 2008 data for Statistics and Operational Research

The journal 'Significance' in December 2012 contained an interesting article by Ralph Kenna and Bertrand Berche called 'Statistics of Statisticians: Critical masses for research groups'. This looked at the large online dataset from the UK Research Assessment Exercise of 2008. Here subjects were grouped by 'unit of assessment', and below I have reproduced a subset of the data for Unit 22, which is 'Statistics and Operational Research'. According to Kenna and Berche, the RAE 'score' for a department is calculated as
score $= X4s + (3 \times X3s + X2s)/7$
where for example $X4s$ is the percentage of FTEstaff in the $4*$ category (ie the top category).
It may be seen that, roughly speaking, this score increases as the number of FTEstaff increases, but the relationship is non-linear, and not really quadratic either. We download the R package 'segmented' to fit the following model. Let N = number of FTE staff. We assume
$score = a_1 + b_1 N$ for $N \leq N_{crit}$, and
$score = a_2 + b_2 N$ for $N > N_{crit}$.
Here $a_1, b_1, a_2, b_2, N_{crit}$ are all parameters to be estimated, with N_{crit} being of special interest: it is the 'critical mass' of a research group, also referred to as the Dunbar number.
Kenna and Berche find that for this dataset, $N_{crit} = 17$, with $se = 6$, and I get the same result, using the 'segmented' package of Vito Muggeo.
(Note that Kenna and Berche give no details of their calculation.)
First, the dataset (which I downloaded and then converted to a .csv file) with the university names shortened by me, for ease of plotting. This dataset was downloaded from
`http://www.rae.ac.uk/results/outstore/uoa22.xls`
(and then I slightly edited the various university names, to save space).

	FTEstaff	X4s	X3s	X2s	X1s	unclassified
Bath	15.00	20	40	35	5	0
Bristol	23.00	25	45	30	0	0
Brunel	10.00	15	35	40	10	0
Cambridge	16.00	30	45	25	0	0

```
Durham         11.60   5  45  45   5           0
Greenwich       2.00   0  40  40  20           0
ImpColl        13.90  25  50  25   0           0
Kent           12.00  20  45  30   5           0
Lancaster      21.65  15  45  35   5           0
Leeds          11.00  25  40  30   5           0
Liverpool       5.00   0  35  50  15           0
LSE            13.00  15  40  35   5           5
LondonMet       4.00   5  20  40  35           0
Manchester     10.90  20  35  30  15           0
Newcastle      13.00  10  45  40   5           0
Nottingham      9.00  20  50  30   0           0
Open            7.00  10  40  45   5           0
Oxford         24.50  40  50  10   0           0
Plymouth        4.00   0  30  45  25           0
QueenMary       8.20  10  30  45  15           0
Reading         7.70   5  30  55  10           0
Salford         9.80   0  35  55  10           0
Sheffield      10.70  10  50  30  10           0
Southampton    28.00  15  50  30   5           0
UCL            13.50  10  40  40  10           0
Warwick        24.00  25  45  30   0           0
Glasgow        13.00  15  35  40  10           0
StAndrews       7.00  10  50  35   5           0
Strathclyde    10.33  10  30  45  15           0
```

and here's what I did with the data: can you do better?

```
set.seed(12) # to give reproducible results
x <- read.csv("Reduoa22.csv", header=T)
x[1,] # we remove the Edinburgh/Heriot-Watt joint submission
rx <- x[-27, 1:6] ; rx# this is the dataset given above
attach(rx)
score <- X4s + (3*X3s + X2s)/7 # This is the formula used for the score
N <- FTEstaff
first.lm <- lm(score ~ N) ; summary(first.lm)
plot(score~ N) ; abline(first.lm)
N <- FTEstaff ; NN <- N*N ; next.lm <- lm(score ~ N + NN) ; summary(next.lm)
points(N, next.lm$fitted.values, pch=19)
pdf("RAE2008.pdf") # to send the plot to a .pdf
plot(score~ N) ; abline(first.lm)
points(N, next.lm$fitted.values, pch=19)
dev.off()
```

This results in Figure 32.1, which shows the linear and the quadratic fits. Now let's try fitting the segmented line.

```
install.packages("segmented")
library(segmented)
```

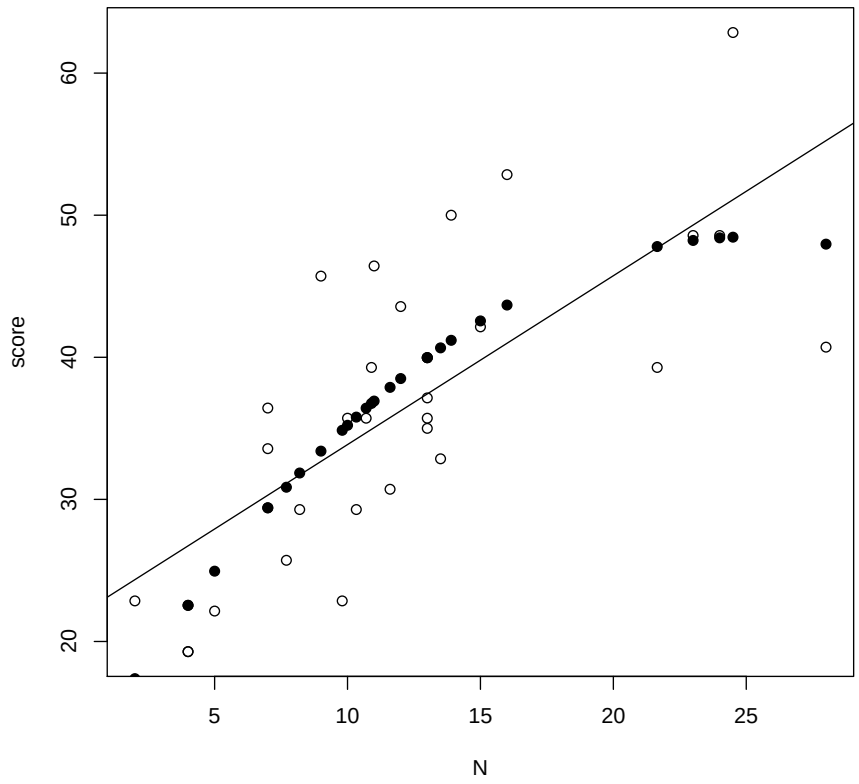


Figure 32.1: RAE 2008: Statistics and OR: Score against FTEstaff, linear and quadratic fits

```
first.glm <- glm(score ~ N)
o <- segmented(first.glm, seg.Z = ~N, psi =list(N=17))
summary(o)
slope(o) # for N above 17, the graph is FLAT
plot(N,score); plot(o, add=T,col=2,lwd=2)
```

You may like to see the corresponding dataset for Pure Maths (Unit of Assessment 20), again with the joint submission Edinburgh/Heriot-Watt removed. This dataset was downloaded from `http://www.rae.ac.uk/results/outstore/uoa20.xls`

	FTEstaff	X4s	X3s	X2s	X1s	unclassified
Bath	10.00	25	35	40	0	0
Birmingham	18.00	15	40	35	5	5
Bristol	34.53	30	40	25	5	0
Cambridge	55.00	30	45	25	0	0
Durham	15.00	20	40	35	0	5

East Anglia	7.00	15	45	35	5	0
Exeter	5.00	10	45	40	5	0
ImpCollLondon	21.80	40	45	15	0	0
Kent	6.00	0	35	55	10	0
King'sCollLondon	13.00	20	50	25	0	5
Lancaster	10.00	10	40	35	15	0
Leeds	23.20	10	45	40	5	0
Leicester	10.00	10	40	50	0	0
Liverpool	15.00	10	35	45	10	0
LSE	12.50	5	40	50	5	0
LondonMet	4.00	10	25	50	15	0
Loughborough	11.40	10	45	45	0	0
Manchester	27.00	20	40	35	5	0
Newcastle	10.00	5	30	60	5	0
Nottingham	15.00	15	35	45	5	0
Open	16.50	5	25	40	30	0
Oxford	55.16	35	40	25	0	0
Queen Mary	20.20	10	50	40	0	0
RoyalHolloway	26.60	0	25	35	20	20
Sheffield	17.25	15	40	45	0	0
Southampton	15.75	5	45	40	10	0
UCL	15.25	20	40	35	5	0
Warwick	32.00	35	45	20	0	0
York	12.34	10	35	50	5	0
Aberdeen	14.00	20	45	35	0	0
Glasgow	16.32	15	40	35	10	0
St Andrews	12.00	5	30	55	10	0
Aberystwyth	8.30	5	35	45	15	0
Cardiff	30.45	5	35	45	15	0
Swansea	20.50	5	35	50	10	0
Queen'sBelfast	8.20	5	40	50	5	0

Figure 32.2 is the pair of graphs of score against N, the number of FTE staff, for each of the 2 'units' Statistics and OR, and Pure mathematics. For this second dataset the graph shows considerable scatter, and use of

```
segmented()
```

will confirm what our eyes immediately tell us: there is no case for fitting a segmented line. If you do fit a segmented lines model to the Pure Mathematics data, you will find that the parameter estimates vary considerably according to the starting point of the iteration, and are then given with very large se's. I have shown the right-hand graph with just one of the possible solutions.

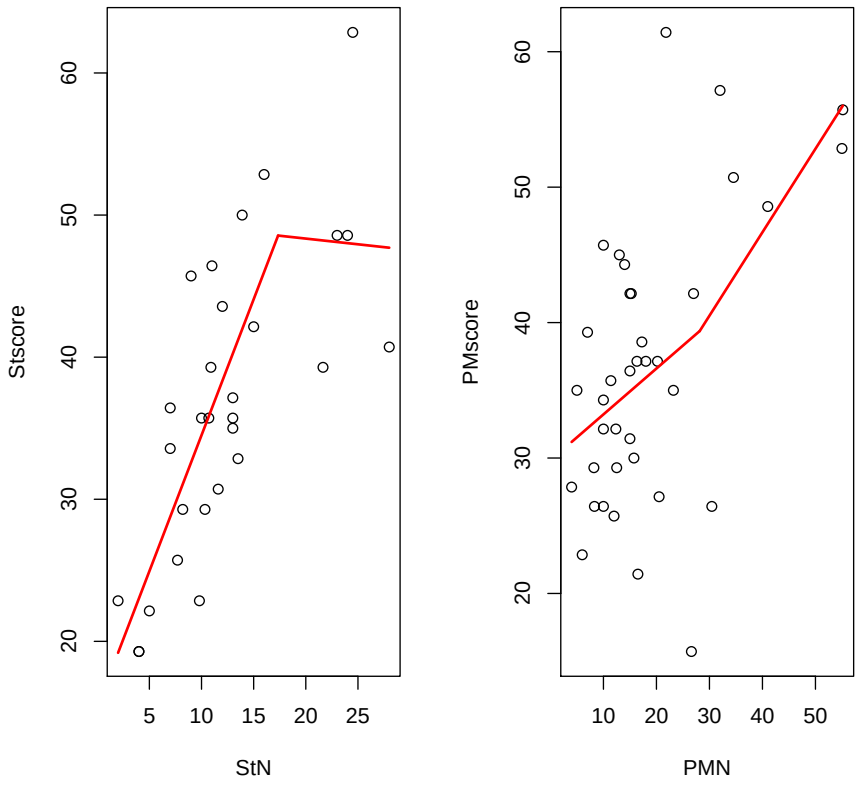


Figure 32.2: RAE 2008: fitting segmented lines for score against department size: Statistics and OR (left), Pure Mathematics (right)

Index