Practicals to Accompany *Statistical Models*

©A. C. Davison
Institute of Mathematics
Swiss Federal Institute of Technology
Lausanne 1015
Switzerland
Anthony.Davison@epfl.ch

January 9, 2005

Chapter 1

Introduction

This document contains practicals to accompany the book *Statistical Models* (Davison, 2003), to which references below to Examples, Tables, and Figures refer.

It is taken for granted that you have access to a current version of the statistical package `R` and that the libraries `MASS`, `survival`, `ellipse`, and `mgcv` are installed. If not, see

`http://www.r-project.org/`

which describes the packages and from which `R` may be obtained free of charge. You will also need to install the library `SMPracticals`, which may be downloaded from the page

`http://statwww.epfl.ch/davison/SM/`

from which you obtained this file. The practicals assume that these libraries are safely installed, and that you have launched `R` and invoked the library by typing

```
library(SMPracticals)
```

This document is **not** intended as a guide to `R` or to using the `S` language, on which numerous excellent books are available in addition to the documentation you will find at

`http://www.r-project.org/`

See for example Dalgaard (2002), Venables and Ripley (2002), or MainDonald and Braun (2003).

Once `R` is launched, online help is available by typing `help.start()`, and help on a specific command or dataset (e.g. `whatsit`) is obtained by typing `help(whatsit)` or, more simply, `?whatsit`, at the `R` prompt. Much information about `R` can be obtained from the `R` mailing list; see the site above.

Dislaimer and warning

These practicals are provided to help learn the material described in Davison (2003). While every effort has been made to ensure their correctness, their author gives no warranty for their behaviour, and accepts no liability of any sort for any consequences that arise, directly or indirectly, from using them.

Suggestions and comments about the practicals themselves are welcome, and may be incorporated into future versions of this document. The author will answer questions about them at his discretion; answers to general questions about `R` or the `S` language should be sought in the sources above.

Conventions

In the `R` code below, the characters following `#` are comments and need not be typed.

Chapter 2

Variation

1. Dataframe `morley` contains data from the Michelson–Morley experiment on the speed of light. Twenty consecutive measurements were made during each of five experiments. To see the data, and make its histogram, empirical distribution function, and some boxplots:

```
library(MASS)
data(morley)
help(morley)
morley
attach(morley)
par(mfrow=c(2,2))
hist(Speed)
plot(sort(Speed),seq(from=0.01,to=1,by=0.01),ylab="EDF",
     xlab="Speed",type="s")
plot(Speed~factor(Expt))
qqnorm(Speed)
```

   For some basic statistics:

```
summary(Speed)  # summaries for all 100 observations
mean(Speed)
sqrt(var(Speed))
tapply(Speed,Expt,mean)  # and for each experiment separately
sqrt(tapply(Speed,Expt,var))
tapply(Speed,Expt,summary)
```

   Do you think there are differences among the experiments? Briefly summarize what you have discovered.

   As the observations were taken in time order within experiments, they may be dependent. To see if this is the case, we make scatter plots of successive observations (y_j, y_{j+1}) for each series and give the correlation between y_j and y_{j+1}:

```
par(mfrow=c(2,3))
succ <- function(y,s)
{ j <- 1:(length(y)-1)
  co <- cor(y[j],y[j+1])
  plot(y[j],y[j+1],xlab="y_j",ylab="y_{j+1}",
```

```
      main=paste("Experiment",s,", correlation",round(co,2)))
  invisible() }
for (i in 0:4) succ( Speed[20*i+1:20],i+1 )
detach("morley")
```

Do you think there is dependence between successive observations? If there is, how will it affect inferences for the true speed of light?

(Section 2.1)

2. Dataframe `mathmarks` contains results for five mathematics exams for each of 88 students. To see and plot them:

```
data(mathmarks)
summary(mathmarks)
apply(mathmarks,2,mean)
sqrt(apply(mathmarks,2,var))
boxplot(mathmarks)
pairs(mathmarks)  # see all scatterplots in matrix
```

For a simpler summary, put results for applied maths and for analysis and statistics together, and brush-and-spin the result:[1]

```
attach(mathmarks)
appl <- (vectors+mechanics)/2
anst <- (analysis+statistics)/2
brush(data.frame(appl,algebra,anst),hist=TRUE)
```

Play with this display a bit. Can you see the worst and best students? Try `brush(mathmarks,hist=T)`. What do you learn?

(Section 2.1; Mardia *et al.*, 1979, pp. 3–4)

3. To get a feel for the information in a probability plot, here is a function to generate samples, standardize them and make a normal probability plot of them:

```
tp <- function(n=c(10,20,50), line=F, ran.gen=rnorm, lims=c(-4,4), ...)
{ m <- length(n)
  for (i in 1:m)  for (j in 1:m)
  { y <- ran.gen(n[i],...)
    y <- (y-mean(y))/sqrt(var(y))
    qqnorm(y,xlim=lims,ylim=lims,main=paste("n=",n[i]))
    if (line) abline(0,1,lty=2) }
 invisible()  }
```

(a) To produce plots for normal samples of sizes 10, 20 and 50:

```
par(mfrow=c(3,3),pty="s")
tp() # without line
tp(line=TRUE) # with line
```

[1]The command `brush` is unavailable in `R`.

Repeat the last two commands a few times. Is the line useful? What effect has sample size on the variability of the plot?

(b) To assess what happens for non-normal data, here are samples from the gamma distribution with shape parameter 4 and from the t distribution with 5 degrees of freedom:

```
tp(ran.gen=rgamma,shape=4,line=TRUE)
tp(ran.gen=rt,df=5,line=TRUE)
```

Try each several times, with and without lines and with various values of `shape` and `df`.

Write a short summary of your findings.

(c) The function below generates data that are either (1) normal, (2) heavy-tailed, (3) skewed, (4) light-tailed, (5) have outliers, or (6) rounded.

```
ran.gen <- function(i,n,m=5)
switch(i,rnorm(n),rt(n,df=m),rgamma(n,shape=m)/m,
         (rbeta(n,m,m)-0.5)*m,c(rnorm(n-m),rcauchy(m)),
          round(m*rnorm(n)))
par(mfrow=c(3,3),pty="s")
gen <- sample(x=1:6,size=9,replace=TRUE)  # make data type
for (i in 1:9) qqnorm(ran.gen(gen[i],500))  # make data and plot them
```

Which normal scores plot(s) correspond to which types of data? Type `gen` to see if you're right. Try the last two lines again, with 50 replaced by 25, 100, or 500.

(Section 2.1.4)

4. To make a Q-Q plot of 1000 Cauchy variables against 1000 averages of $n = 5$ Cauchy variables:

```
n <- 500; R <- 1000
y <- rcauchy(R)
ybar <- apply(matrix(rcauchy(n*R),R,n),1,mean)
par(pty="s",mfrow=c(1,2))    # square panel with equal axes

lims <- range(y,ybar)
qqplot(y,ybar,xlim=lims,ylim=lims)  # note the outliers
abline(0,1,lty=2)

lims <- c(-20,20)
qqplot(y,ybar,xlim=lims,ylim=lims)  # close-up
abline(0,1,lty=2)
```

Repeat this a few times. Try also with $n = 10, 100$. Does this make any difference? Do the same for normal and exponential variables, replacing `rcauchy` with `rnorm` and `rexp`. What is the theoretical explanation?

(Section 2.1.4)

5. Intervals between failures (hours of operating time) for air-conditioning equipment in a Boeing 720 jet aircraft were

55 320 56 104 220 239 47 246 176 182 33 15 104 35.

If failures occur as a Poisson process while the planes are running, the intervals will be independent with a common exponential distribution. Check this using an exponential probability plot, and give a graphical estimate of the average time between failures. You may find this code useful:

```
qqexp <- function(y,...) {
 y <- sort(y[!is.na(y)])
 n <- length(y)
 x <- qexp((1:n)/(n+1))
 plot(x,y,xlab="Exponential plotting position",
      ylim=range(0,y),xlim=range(0,x),...)
 abline(0,mean(y),lty=2)
 invisible() }
y <- c(55,320,56,104,220,239,47,246,176,182,33,15,104,35)
qqexp(y,ylab="Air-conditioning data")
```

(Section 2.1.4; Proschan, 1963; Cox and Snell, 1981, p. 143)

Chapter 3

Uncertainty

1. If Z has the standard normal distribution, $W = |Z|$ has the *half-normal distribution*. Show that $\Pr(W \le w) = 2\Phi(w) - 1$ for $w \ge 0$, and zero otherwise, and deduce that its p quantile is $\Phi^{-1}\{\frac{1}{2}(1+p)\}$.

   If $y_1, \ldots, y_n$ is thought to be a random sample from the $N(0, \sigma^2)$ density and n is small, it may be useful to replace a normal scores plot of the y_j with a *half-normal plot*, i.e. a plot of the ordered $|y_j|$ against half-normal plotting positions. How you would expect this to appear if the data were (i) normal, (ii) had heavier tails, and (iii) had lighter tails? To verify your conjectures:

```
qqhnorm <- function(y, line=F, ...)
{  y <- y[!is.na(y)]  # drops any NAs
   n <- length(y); i <- c(1:n); r <- range(c(y,0))
   o <- qnorm(0.5*(1+i/(n+1)))
   qqplot(o,y,ylim=r,xlab="Half-normal quantile",...)
   if (is.numeric(line)) abline(0,line,lty=2)
   invisible() }
par(mfrow=c(2,2))
qqhnorm(abs(rnorm(20)),line=1,ylab="Normal data")
qqhnorm(abs(rt(20,df=5)),line=1,ylab="t_5 data")
qqhnorm(abs(rexp(20)-rexp(20)),line=1,ylab="Laplace data")
qqhnorm(rgamma(20,2),line=1,ylab="Gamma(2) data")
```

   Were you right? Write a short summary of your findings.

   (Sections 2.1.4, 3.2)

2. The t statistic for a normal random sample was derived in 1908 by a mixture of simulation, mathematics and guesswork. Before arriving at a mathematical derivation,[1] Student wrote measurements from 3000 criminals on pieces of card, shuffled them thoroughly, and divided them at random into 750 groups of size $n = 4$. He then calculated the average and sample variance, $\overline{Y}$ and S^2, for each group and looked at their empirical distributions and that of $Z = n^{1/2}(\overline{Y} - \mu)/S$; he assumed that μ equalled the average of all 3000 observations. Finding that the fit of the t_3 density to the empirical distribution of Z was excellent, he then set about a mathematical derivation. We can mimic his simulation and check the distributions of $\overline{Y}$, S^2 and Z with much less effort:

[1] His derivation was incomplete, because he assumed but did not prove independence of $\overline{Y}$ and S^2.

```
n <- 4; R <- 750
y <- matrix(rnorm(n*R),R,n)
ybar <- apply(y,1,mean)
s2 <- apply(y,1,var)
z <- ybar/sqrt(s2/n)
par(mfrow=c(2,2),pty="s")
plot(ybar,log(s2),pch=".")
qqnorm(ybar,pch="."); abline(0,1/2,lty=2)
oc <- qchisq(c(1:R)/(R+1),df=n-1)
qqplot(oc,(n-1)*s2,xlab="Chi-squared quantile",pch=".")
abline(0,1,lty=2)
ot <- qt(c(1:R)/(R+1),df=n-1)
qqplot(ot,z,xlab="Student t quantile",pch=".")
abline(0,1,lty=2)
```

Does the upper left panel support independence of $\overline{Y}$ and S^2?

With $Y_1, \ldots, Y_n \overset{\text{iid}}{\sim} N(0,1)$ and $n = 4$, we have $\overline{Y} \sim N(0, \frac{1}{4})$, $3S^2 \sim \chi^2_3$, and $Z \sim t_3$. Do the other panels support this?

To repeat your experiment with data from other densities, replace `rnorm(n*R)` in the second line above with, for example, `rt(n*R,df=3)` or `rexp(n*R)`. Do $\overline{Y}$ and S^2 now seem correlated? How does the distribution of Z change?

(Section 3.2; Student, 1908)

3. Let $Y_1, \ldots, Y_n$ be a random sample from a distribution with zero mean and variance σ^2, and define $W_u = n^{-1}(Y_1 + \cdots + Y_{\lfloor nu \rfloor})$, for $0 < u \leq 1$.[2] Let $0 < u_1 < \cdots < u_k < 1$. Show that as $n \to \infty$ the joint limiting distribution of $W_{u_1}, W_{u_2} - W_{u_1}, \ldots, W_1 - W_{u_k}$ is multivariate normal with mean zero and covariance matrix $\sigma^2 \text{diag}\{u_1, u_2 - u_1, \ldots, 1 - u_k\}$, and hence find the joint distribution of $W_{u_1}, W_{u_2}, \ldots, W_{u_k}, W_1$.[3] This is the distribution of the points of a *Wiener process* $\{W_u\}$ with $W_0 = 0$.

   The random process $\{W_u\}$, $0 < u \leq 1$ given that $W_0 = W_1 = 0$ is called a *Brownian bridge*; denote it $\{B_u\}$. Thus $B_0 = B_1 = 0$. Show that the joint distribution of $B_{u_1}, \ldots, B_{u_k}$ is multivariate normal with mean zero and covariances given by $\text{cov}(B_u, B_v) = \sigma^2 u(1 - v)$, $u < v$. Show also that $B_u \overset{D}{=} W_u - uW_1$.

   To see what sample paths of a Wiener process and Brownian bridge look like when $\sigma = 1$, repeat the last three lines below a few times.

```
par(mfrow=c(1,2))
wiener <- function(n) cumsum(rnorm(n))/sqrt(n)
n <- 5000; u <- c(1:n)/n; w <- wiener(n); b <- w-u*w[n]
plot(u,w,ylim=c(-2,2),type="l",ylab="Wiener process")
plot(u,b,ylim=c(-2,2),type="l",ylab="Brownian bridge")
```

[2] $\lfloor x \rfloor$ is the largest integer smaller than x.

[3] Brownian motion is named after Robert Brown, a Scottish botanist who in 1827 observed that pollen grains suspended in water move erratically and apparently randomly. Such experiments led many nineteenth-century scientists to believe in the existence of molecules, but only in 1905 did Albert Einstein (1879–1955) confirm this decisively. The random process that models Brownian motion is called a Wiener process, after Norbert Wiener (1894–1964) who put it on a firm mathematical footing.

The sample paths are the limit of a random walk and are highly irregular: in fact they are continuous everywhere but nowhere differentiable.

(Section 3.2)

4. If $U \sim U(0,1)$, show that its mean and variance are $\frac{1}{2}$ and $\frac{1}{12}$, and that all its odd moments are zero.

   One approach to generating standard normal variables is to set $Z = U_1 + \cdots + U_{12} - 6$. Explain the rationale behind this. To generate 10,000 such Zs and see if they appear normal:

```
u <- matrix(runif(12*10000),10000,12) # 10,000x12 matrix of Us
z <- apply(u,1,sum)-6  # add across rows
par(pty="s")  # square plot
qqnorm(z,pch=".",xlim=c(-4,4),ylim=c(-4,4))
abline(0,1,lty=2)
```

   Should the algorithm be used?

   (Section 3.3)

5. Here is an `S` implementation of a linear congruential generator:

```
congru <- function(n, M=2^31,a=1,c=1,seed=1)
{ x <- rep(NA,n+1)
  x[1] <- seed
  for (i in 2:(n+1)) x[i] <- (a*x[i-1] + c) %% M
  list(x=x,u=x/M) }
congru(16,M=13,a=2,c=0)
```

   Give the periods when (a) $M = 13$, $a = 2$ and $c = 0$; (b) $M = 13$, $a = 5$ and $c = 0$; and $M = 16$, $a = 5$ and $c = 1$. Taking $a = c = 1$ always gives period M. Is it wise?

   The once-popular algorithm `RANDU` used $M = 2^{31}$, $a = 2^{16} + 3$, and $c = 0$. To make and display triples (U_i, U_{i+1}, U_{i+2}) from this, [4]

```
n <- 10000
u <- congru(n=n+2,M=2^(31),a=2^(16)+3,c=0)$u
randu <- cbind(u[1:n],u[2:(n+1)],u[3:(n+2)])
spin(randu)  # help(spin) gives help
```

   Inspect its lattice structure. Why is `RANDU` no longer trusted?

   To see how lattice structure can interact with transformations to other distributions, we take $M = 2024$, $a = 65$ and $c = 1$:

```
u <- matrix(congru(2024,2048,65,1)$u,2,1024)
u1 <- u[1,]
u2 <- u[2,]  # u1 and u2 are successive pairs
r <- sqrt(-2*log(u1))
theta <- 2*pi*u2
```

[4] This part uses `spin` and cannot be performed in `R`.

```
x <- r*cos(theta)
y <- r*sin(theta)
par(pty="s")
plot(x,y)  # successive pairs for Box-Muller algorithm
```

Try this with the values for RANDU also.

Does the built-in $U(0,1)$ generator `runif` show the same problems?

(Section 3.3)

6. Verify that the following code fragment does the calculations necessary for Example 3.24, with the simulations for the normal, gamma, and Cauchy cases. Check some of numbers in Table 3.2 of the book.

```
cover.sim <- function(n, ran.gen=rnorm, a=c(0.95,0.975), R=1600, ...)
{  z <- matrix( ran.gen(n*R,...),R,n)
  zbar <- apply(z,1,mean)
  sz <- sqrt(apply(z,1,var))
  t <- sqrt(n)*zbar/sz
  c1 <- c2 <- NULL
  for (i in 1:length(a)) {
  c1 <- c(c1,mean(t<qt(1-a[i],df=n-1)))
  c2 <- c(c2,mean(t<qt(a[i],df=n-1))) }
  100*matrix(rbind( c1, c2, c2-c1 ),nrow=1,byrow=F) }
rgam <- function(n, shape ) rgamma(n,shape)-shape
rmix <- function(n, p=0.1) rnorm(n,sd=sqrt(1+8*(runif(n)<p)))
rlaplace <- function(n) rexp(n)-rexp(n)
cover.sim(10)
cover.sim(10, ran.gen=rgam, shape=2)
cover.sim(10, ran.gen=rcauchy)
```

(Section 3.3)

Chapter 4

Likelihood

1. Following Example 4.21, we assess the effect of rounding on the usual estimates for a normal random sample $Y_1, \ldots, Y_n$, rounded to $X_1, \ldots, X_n$. The rounded data have average and variance $\overline{X}$ and $S^2 = (n-1)^{-1}\sum(X_j - \overline{X})^2$, whose means are κ_1 and κ_2, where the κ_r are the cumulants of X. Their variances are κ_2/n and $\kappa_4/n + 2\kappa_2^2/(n-1)$. The following function calculates the efficiencies and means of these estimators relative to those based on $Y_1, \ldots, Y_n$, when the $Y_j \sim N(\mu, \sigma^2)$.

```
I.moments <- function(delta, n=10, m=0, s=1, dig=4)
{ k <- seq(from=-5,to=5,by=delta)
  pi <- pnorm( (k+0.5*delta-m)/s ) - pnorm( (k-0.5*delta-m)/s )
  k1 <- sum(pi*k); k2 <- sum(pi*k^2)
  k3 <- sum(pi*k^3); k4 <- sum(pi*k^4)
  k4 <- k4 - 4*k1*k3 - 3*k2^2 + 12*k1^2*k2 - 6*k1^4
  k3 <- k3 - 3*k1*k2+2*k1^3
  k2 <- k2 - k1^2
  eff.m <- s^2/k2
  eff.v <- (2*s^4/(n-1))/(k4/n + 2*k2^2/(n-1))
  round(c(100*eff.m,100*eff.v,k1,k2),digits=dig) }
I.moments(0.01)
```

   Read the function carefully to check it works as advertized.

   Obtain the efficiencies for various values of δ/σ and n. Is Table 4.2 of the book misleading?

2. `births` contains the data in Table 2.1 on times spent on delivery suite by 95 women. A possible model for these data is that the number of women arriving each day is Poisson, with mean θ, and that the time spent by each of them has a gamma distribution with mean μ and shape parameter α.

   (a) For a sample $m_1, \ldots, m_k$ from the Poisson distribution

$$f(m;\theta) = \frac{\theta^m}{m!}e^{-\theta}, \quad \theta > 0, m = 0, 1, \ldots,$$

   find the maximum likelihood estimate and gives its asymptotic variance.

   (b) Show that the gamma distribution parametrized as

$$f(y;\mu,\alpha) = \frac{\alpha^\alpha}{\Gamma(\alpha)\mu^\alpha}y^{\alpha-1}\exp(-\alpha y/\mu), \quad \alpha, \mu > 0,\ y > 0,$$

has mean μ and variance μ^2/α, and find the information matrix $I(\mu, \alpha)$ based on a sample $y_1, \ldots, y_n$. Show that given α, the maximum likelihood estimate of μ is $\widehat{\mu}_\alpha = \overline{y}$, and hence verify that the following `S-Plus` functions yield the profile log likelihood $\ell_p(\alpha) = \ell_p(\alpha, \widehat{\mu}_\alpha)$:

```
logL.gam <- function(alpha, y)
{ muhat <- mean(y)
  L <- dgamma( alpha*y/muhat, shape=alpha )*alpha/muhat
  -sum(log(L)) }
```

(c) Now apply the ideas from (a) and (b) to the birth data:

```
data(births)
fit <- nlm(logL.gam, 1, hessian=TRUE, y=time)
alphahat <- fit$estimate
se.alphahat <- sqrt(diag(solve(fit$hessian)))  # sqrt(diagonal of inverse hessian)
muhat <- mean(time)
```

to obtain the maximum likelihood estimates. Obtain the standard error for $\widehat{\mu}$, and give 95% confidence intervals for θ, α, and μ.

(d) Gamma probability plot to assess the quality of the model:

```
par(pty="s")
n <- length(time)
qqplot( mean(time)*qgamma(c(1:n)/(n+1), shape=alphahat)/alphahat, time,
       xlab="Quantile of fitted gamma distribution" )
abline(0,1,lty=2)
```

(Sections 4.1–4.4)

3. Once upon a time, on a spring afternoon in Portland, Oregon, these data were collected on the sizes of different groups of people in the park, in the street, and so forth:

Group size y	1	2	3	4	5	6
Frequency n	1486	694	195	37	10	1

A possible model is that the group sizes have truncated Poisson density

$$f(y; \theta) = \frac{\theta^y e^{-\theta}}{y!(1 - e^{-\theta})}, \quad y = 1, 2, \ldots, \quad \theta > 0.$$

To plot the log likelihood $\ell(\theta)$ over $0.1 \leq \theta \leq 2$:

```
y <- c(1:6)
n <- c(1486,694,195,37,10,1)
nlogL <- function(x, y, n.obs)     # x is theta
{  f <- dpois(y,x)/(1-dpois(0,x))  # dpois is Poisson PDF
   -sum(n*log(f))  }               # negative log likelihood
theta <- seq(from=0.1,to=2,length=200)
L <- rep(NA,200)
for (i in 1:200) L[i] <- -nlogL(theta[i],y,n)
plot(theta,L,type="l",ylab="Log likelihood")
```

You may like to plot this on a smaller interval. What values of θ seem plausible?

To minimize $-\ell(\theta)$ using `nlm`, find the observed information $J(\widehat{\theta})$ numerically and the 95% confidence interval using $z_1(\theta) = J(\widehat{\theta})^{1/2}(\widehat{\theta} - \theta)$:

```
x0 <- 1 # initial value
fit <- nlm(nlogL, x0, hessian=TRUE, y=y, n.obs=n)
fit  # see results of minimization
thetahat <- fit$estimate
J.inv <- diag(solve(fit$hessian)) # inverse of Hessian
thetahat-qnorm(c(0.975,0.025))*sqrt(J.inv) # 95% confidence interval based on thetahat
```

For a 95% confidence interval based on the signed likelihood ratio statistic $z_2(\theta) = \text{sign}(\widehat{\theta} - \theta)w(\theta)^{1/2}$:

```
theta <- seq(from=0.8,to=1,length=200)
for (i in 1:200) L[i] <- -nlogL(theta[i], y, n)
w <- -2*(L+fit$minimum)
z2 <- sign(thetahat-theta)*sqrt(w)
plot(theta,z2,type="l",ylab="Signed likelihood ratio statistic")
lines(theta,(thetahat-theta)/sqrt(J.inv),lty=2)
abline(h=c(-1.96,1.96),lty=3)
abline(h=0)
```

What do you deduce from the close agreement between the diagonal lines showing $z_2(\theta)$ and $z_1(\theta)$ in the range of interest? Does your deduction agree with the confidence interval obtained from the likelihood ratio statistic, by `lik.ci(theta, L)`?

For a chi-squared goodness of fit statistic:

```
E <- sum(n)*dpois(y,thetahat)/(1-dpois(0,thetahat))
P <- sum( (n-E)^2/E )  # Pearson's statistic
```

How many degrees of freedom has this? To get the significance level, `1-pchisq(P,df)`, where `df` is your chosen value. Is the fit good?

(Sections 4.1–4.5)

4. The data below are from an experiment of Rutherford and Geiger in which the number of α-particles registered by a counter in periods of one-eighth of a minute were recorded; n represents the number of particles and y the number of times n was observed:

n	0	1	2	3	4	5	6	7	8	≥ 9	Total
y	57	203	383	525	532	408	273	139	45	43	2608

On the supposition that the events form a Poisson process of rate 8λ events per minute, show that the probability of 9 or more events is $\exp(-\lambda)\sum_{r=9}^{\infty}\lambda^r/r!$, and hence write down the likelihood for the observed data. Maximize it numerically[1] and give a confidence interval for 8λ. How well does the model fit?

(Sections 4.1–4.4; Moore, 1952)

[1] You may find it useful to look at the preceding practical.

5. `blood` contains data on the incidence of blood groups O, A, B, and AB in 12 different studies on people living in Britain or of British origin living elsewhere. To fit the single-locus model used in Example 4.38, we use the following code, which computes the fitted probabilities and then the negative log likelihood starting from $(\log \lambda_A, \log \lambda_B)$:

```
make.prob <- function(log.lam)
{
  lamA <- exp(log.lam[1])
  lamB <- exp(log.lam[2])
  lamO <- 1-lamA-lamB
  piA <- lamA*(lamA+2*lamO)
  piB <- lamB*(lamB+2*lamO)
  piAB <- 2*lamA*lamB
  piO <- lamO^2
  c(piO,piA,piB,piAB)
}
nlogL <- function(log.lam, y)  -sum(y*log(make.prob(log.lam)))
```

To fit this to each of the studies separately:

```
fits <- list(neg.loglik=rep(NA,12),lambda=matrix(NA,12,3),
             fitted=matrix(NA,12,4))
for (i in 1:nrow(blood))
{
  fit <-  nlm(nlogL, c(-1,-3),y=blood[i,])
  fits$neg.loglik[i] <- fit$minimum
  fits$lambda[i,1:2] <- exp(fit$estimate)
  fits$lambda[i,3] <- 1- sum(fits$lambda[i,-3])
  fits$fitted[i,] <- sum(blood[i,])*make.prob(fit$estimate)
}
fits
```

The overall chi-squared statistic is computed by

```
sum( (fits$fitted-blood)^2/fits$fitted )
```

How many degrees of freedom has this? Does the model seem to fit?

To see a model with the same values of λ_A, λ_B fitted to the entire dataset:

```
fit <- nlm(nlogL, c(-1,-3),y=apply(blood,2,sum))
2*(fit$minimum-sum(fits$neg.loglik))
```

What does this last calculation give? Which model seems to fit best?

(Chapter 4; Taylor and Prior, 1938)

6. The generalized Pareto distribution function is[2]

$$G(y;\sigma,\xi) = \begin{cases} 1-(1+\xi y/\sigma)_+^{-1/\xi}, & \xi \neq 0, \\ 1-\exp(-y/\sigma), & \xi = 0. \end{cases} \tag{4.1}$$

[2] x_+ equals x if $x > 0$, and otherwise equals zero.

Note that if $\xi < 0$, then the support[3] of the generalized Pareto density function is $(0, \gamma)$, where $\gamma = -\sigma/\xi$. What implications has this for estimation?

(a) Check that this function implements simulation by inversion from this distribution:

```
rgpd <- function(n, s=1, xi=0)
  ifelse(rep(xi,n)==0,-s*log(runif(n)),s*(runif(n)^(-xi)-1)/xi)
```

and perform some comparisons of data generated from (4.1) with the exponential distribution:

```
qqexp(rgpd(100))
qqexp(rgpd(100, xi=1))
qqexp(rgpd(100, xi=-1))
```

(b) We now set $\sigma = 1$, and consider maximum likelihood estimation of ξ. By noting that $y/(1+\xi y) = 1 - \xi^{-1}(1+\xi y)^{-1}$, show that the score function has finite expectation only if $\xi > -1$, and finite variance only if $\xi > -1/2$. Deduce that a necessary condition for maximum likelihood estimation to be regular is that $\xi > -1/2$.

(c) To see what (b) implies in practice, we plot log likelihoods for some samples from (4.1):

```
par(mfrow=c(3,3),pty="s")
negL <- function(xi, y) (1/xi+1)*sum(log(1+y*xi)) # negative log likelihood
L <- xi <- seq(from=-1,to=3,length=100)
xi.sim <- 1; n <- 50
for (i in 1:9)    # repeat from here
{ y <- rgpd(n, xi=xi.sim)
  for (i in 1:length(xi)) L[i] <- -negL(xi[i],y)
  plot(xi,L,type="l") }
```

Repeat this last command with `xi.sim` equal to 1, 0, -0.2, -0.5, and -1, and discuss how the maximum likelihood estimator $\widehat{\xi}$ will behave. Does it help to take $n = 200$, say?

(d) Let $Y_1, \ldots, Y_n$ denote a random sample from (4.1), and set $M_n = \max\{Y_1, \ldots, Y_n\}$. By writing (4.1) as a function of ξ and γ, show that $n^{-\xi}(\gamma - M_n)/\gamma$ has a non-degenerate limiting distribution as $n \to \infty$, and deduce that if $\xi < -1/2$ then the rate of convergence of M_n to γ is faster than $n^{1/2}$. This implies that the estimator $\tilde{\gamma} = M_n$ can be treated as fixed equal to γ, at least asymptotically. Hence suggest the form of an estimator $\tilde{\xi}$ for ξ, and use simulated data to assess the performance of $\tilde{\gamma}$ and $\tilde{\xi}$ based on samples of size $n = 50$ when $\xi = -1, -0.7, -0.5, -0.3$.

(Section 4.6; Smith, 1985)

[3]The set where the density is positive.

Chapter 5

Models

1. `forbes` contains data due to James Forbes on the relation between atmospheric pressure (inches of mercury) and the boiling point of water (degrees Fahrenheit) at 17 locations in the Alps; he sought a formula for predicting pressure y from boiling point x. To see the data and plot them on various scales:

   ```
   data(forbes)
   attach(forbes)
   forbes
   par(mfrow=c(2,2))
   plot(bp,pres)
   plot(log(bp),pres)
   plot(bp,log(pres))
   plot(log(bp),log(pres))
   ```

   Are all the points close to a straight line? Which plot seems closest to linear?

   To fit a straight-line model and print estimates and their standard errors:

   ```
   fit <- lm(pres~bp)
   summary(fit)
   ```

   Give a careful interpretation of the estimates, including their units, and compute the predicted value when $x = 200°$F. For a different (better?) parametrization:

   ```
   x <- bp-190
   fit <- lm(pres~x)
   summary(fit)
   ```

   Again interpret the estimates. Are any equal to those for the first fit? Why? What about the standard errors and t statistics?

   For a graphical summary and a look at the residuals:

   ```
   plot(x,pres)
   abline(fit)
   plot(fitted(fit),resid(fit)) # plot residuals against fitted values
   ```

   To see the residuals $y_j - \widehat{y}_j$, type `resid(fit)`. Can you see any problems with the model? For a (possibly) better model:

```
fit <- lm(log(pres)~bp,subset=-12)
summary(fit)
plot(bp,log(pres))
abline(fit)
plot(bp[-12],resid(fit))
```

Do the residuals look better now? What is your prediction equation for $\mathrm{E}(\log Y)$ in terms of x? Give a standard error and 95% confidence interval for $\beta_0 + 200\beta_1$, and a 95% prediction interval for a new value of Y taken when $x = 200$.

(Section 5.1; Atkinson, 1985)

2. `aml` contains remission times in weeks for two groups of patients with acute myelogeneous leukaemia, one of which received maintenance chemotherapy. There are censored times in each group. To see the data and compare the product-limit estimates for the two groups:

```
library(survival)
data(aml)
aml
?survfit   # check use of function survfit for fitting survival models
fit <- survfit(Surv(time, status)~x,data=aml)
plot(fit,lty=c(1,2))
par(mfrow=c(1,2))
plot(survfit(Surv(time, status)~x,data=aml[1:11,]),xlim=c(0,180))
plot(survfit(Surv(time, status)~x,data=aml[12:23,]),xlim=c(0,180))
```

Do you think there is difference in survival between the two groups?

The function `survreg` is used to fit parametric models. To fit separate exponential means to the two groups, for example:

```
fit.exp <- survreg(Surv(time,status)~x, dist="exp",data=aml)
summary(fit.exp)
```

The model is parametrized so that the mean of the first (maintained) group is $\exp(\beta_1)$ and that of the second group is $\exp(\beta_1 + \beta_2)$. In the output `z` is $\widehat{\beta}/\mathrm{SE}(\widehat{\beta})$ and `Chisq` is the likelihood ratio statistic for a difference between groups. Is the effect of chemotherapy significant? Does the sign of $\widehat{\beta}_2$ make sense? Give a 95% confidence interval for β_2, based on asymptotic likelihood results.

To assess the fit of the model:

```
par(mfrow=c(1,1))
plot(fit,lty=c(1,2))
x <- seq(from=0,to=180,by=1)
lines(x,1-pexp(x,exp(-coef(fit.exp)[1])))
lines(x,1-pexp(x,exp(-sum(coef(fit.exp)))),lty=2)
```

Is there a significant difference between the two groups? Which has better survival?

To fit a Weibull density in which both groups have the same shape parameter (confusingly called `scale` in the output):

```
fit.wei <- survreg(Surv(time,status)~x,data=aml)
summary(fit.wei)
2*(fit.wei$loglik[2]-fit.exp$loglik[2])
```

This last calculation compares the Weibull and exponential models using their likelihood ratio statistic. Does the Weibull model fit better?

To add the lines for the fitted Weibull model to the graph:

```
lines(x,1-pweibull(x,fit.wei$scale,exp(coef(fit.wei)[1])),lty=3)
lines(x,1-pweibull(x,fit.wei$scale,exp(sum(coef(fit.wei)))),lty=4)
```

Is the fit better?

Are matters improved if the shape parameter is allowed to vary between the two groups?

```
fit.wei1 <- survreg(Surv(time,status)~x,data=aml[1:11,])
fit.wei2 <- survreg(Surv(time,status)~x,data=aml[12:23,])
summary(fit.wei1)
summary(fit.wei2)
2*(fit.wei1$loglik[1]+fit.wei2$loglik[1]-fit.wei$loglik[2])
```

Look at the help for `survreg`, and try fitting with some of the other distributions. Does any fit the data better?

(Section 5.4; Miller, 1981, p. 49)

3. (a) Following Example 5.36, show that the complete-data log likelihood for a p-component mixture of exponential families can be written as

$$\delta(u-r)\left\{\log \pi_r + s_r(y)\theta_r(\omega_r) - b_r(\omega_r)\right\},$$

and deduce that $\pi_r^\dagger = n^{-1}\sum_j w_r(y_j;\theta')$.

Find the expressions for $\widehat{\mu}_r^\dagger$ for the exponential and Poisson densities $\mu^{-1}e^{-y/\mu}$ and $\mu^y e^{-\mu}/y!$, and show that these are always positive for samples $y_1,\ldots,y_n$ in which at least one observation is positive.

(b) To illustrate these calculations we fit a mixture model to the HUS counts of Example 4.40, ignoring their time dependence. The functions `wei`, `update` and `loglik` below respectively compute the weights, update from θ' to $\theta^\dagger$ and compute the log likelihood, while `EM` iterates the algorithm until convergence.

```
wei <- function(th, y)
{ # produce n times p matrix of weights
  w <- matrix(NA,length(y),ncol(th))
 for (r in 1:ncol(th)) w[,r] <- th[1,r]*dpois(y,th[2,r])
  sweep(w,1,apply(w,1,sum),"/") }

update <- function(th, y)
{ # update theta to theta.dagger
  w <- wei(th, y)
  th[1,] <- apply(w,2,mean)
  th[2,] <- crossprod(w,y)/apply(w,2,sum)
```

```
  th }

loglik <- function(th, y)
{ # compute log likelihood at current theta
  d <- rep(0,length(y))
  for (r in 1:dim(th)[2])  d <- d + th[1,r]*dpois(y,th[2,r])
  sum(log(d)) }

EM <- function(init, y, tol=10^(-6))
{ # iterate EM algorithm starting from init
  # stopping criterion is relative change in log lik < tol
  L.init  <- loglik(init, y)
  th <- update(init, y)
  L.th <- loglik(th, y)
  R <- 1
  while (abs((L.th-L.init)/L.th)>tol)
  { init <- th
    L.init <- L.th
    th <- update(init, y)
    L.th <- loglik(th, y)
    R <- R+1 }
   list(theta=th, L=L.th, R=R) }
```

To apply these to the data:

```
y <- hus$birmingham
p <- 1   # 1-component mixture, with starting-values pi=1, mu=2
init1 <- matrix(c(1,5), 2, p)
fit1 <- EM(init1, y)
p <- 2  # 2-component mixture with pi1=pi2=0.5, mu1=5, mu2=1
init2 <- matrix(c(0.5, 5, 0.5, 1), 2, p)
fit2 <- EM(init2, y)
fit1  # compare fits
fit2
plot(wei(fit2$theta, y)[,1])  # plot final weights for group 1
```

For which y_j are the $\pi_r(y_j;\widehat{\theta}) > 0.5$? Does this make sense?

Compare the fits using log likelihood and AIC. Does a three-component mixture improve matters? Which do you think best? Do your conclusions differ for the Newcastle data?

(c) Implement expression (5.40) for this example with $p = 2$ and obtain the standard errors of $\widehat{\pi}_1, \widehat{\mu}_1$ and $\widehat{\mu}_2$.

(d) Which lines of the code above would have to be modified to fit a mixture of exponential densities?

(Section 5.5.2)

4. A simple (perhaps overly simple) model for outliers in normal samples is that data arise from the mixture density

$$(1-p)\sigma^{-1}\phi\{(y-\mu)/\sigma\} + p(k\sigma)^{-1}\phi\{(y-\mu)/(k\sigma)\},$$

where $k > 1$ is known but μ, σ and p are not. Under this model outliers arise with probability p from a normal density with mean μ but variance larger than σ^2. Show how μ, σ^2 and p may be estimated using the EM algorithm and modify the code in the previous practical to fit this model. Try out your code on the data in the final column of Table 1.1, using $k = 1, 2, 3$. How do the estimates depend on k?

(Section 5.5.2)

Chapter 6

Stochastic Models

1. We fit stationary Markov chains up to third order using the function `MClik`. Study the code for `MClik` carefully, and explain how it works. To apply this to the daily rainfall data on the Pacific island of Alofi:

   ```
   alofi.fit <- MClik(alofi)
   ```

   What order of chain appears to give the best fit, (i) using likelihood ratio tests, (ii) using AIC?

   Discuss how the model might fail, and say how you could assess how well it fits the data.

   (Section 6.1; Avery and Henderson, 1999)

2. `frets` contains data from Frets (1921) on the length and breadth of the heads of the first and second adult sons in a sample of 25 families. For an exploratory analysis:

   ```
   data(frets)
   cor(frets)
   pairs.mod(frets)
   ```

   How should the scatterplot matrix be interpreted?

   The distribution of a correlation coefficient R based on a random sample of size from a bivariate normal distribution with correlation ρ may be approximated using Fisher's z-transformation:

   $$Z = \tfrac{1}{2}\log\left(\frac{1+R}{1-R}\right) \quad \dot{\sim} \quad N\left(\tfrac{1}{2}\log\left(\frac{1+\rho}{1-\rho}\right), \frac{1}{n-3}\right).$$

   This can be used to assess whether individual partial correlation coefficients estimated from the full model are significantly different from zero:

   ```
   fisher <- function(r, n) c(0.5*log((1+r)/(1-r)), sqrt(1/(n-3)))
   fisher(0.13,25)
   ```

   What do you conclude? Using this crude approach, which partial correlations appear to be zero? A better approach is to use the `R` library `ggm` for fitting graphical Gaussian models:

```
library(ggm)    # you may need to download this R library
gr <- UG(~ l1*b1+l1*l2+l1*b2+b1*l2+b1*b2+l2*b2) ## Graph with all edges
fitConGraph(gr, cor(frets), n=25)
gr <- UG(~ l1*b1+l1*l2+b1*b2+l2*b2) # Drop edges for partial correlations < 0.2
fitConGraph(gr, cor(frets), n=25)
gr <- UG(~ l1*b1+l2*b2) # Independent heads,so just two edges
fitConGraph(gr, cor(frets), n=25)
```

If the model provides an adequate fit, the deviance should have an approximate chi-squared distribution with the given degrees of freedom. Try fitting some other models, and give the interpretation of the best-fitting one.

(Section 6.3; Whittaker, 1990, pp. 254–260)

3. `manaus` contains the monthly average river height or stage of the Rio Negro near the central Amazonian city of Manaus from 1903–1992. To see the data and its correlogram and autocorrelogram:

```
data(manaus)
par(mfrow=c(2,2))
plot(manaus)
acf(manaus)
pacf(manaus)
```

To fit autoregressive models and to see the corresponding values of AIC:

```
fit <- ar(manaus)
fit$aic
plot(fit$aic)
plot(fit$aic[-1]) # AIC without initial huge value
```

Compare carefully the form of the AIC and the partial autocorrelogram. Where do the big drops in AIC come?

ARMA models may be fitted with the `R` routine `arima`. For example,

```
fit1 <- arima(manaus,order=c(1,0,1))
fit1
```

fits an ARMA model with $p = q = 1$ to the original data, and

```
fit2 <- arima(manaus,order=c(1,1,1))
fit2
```

fits such a model to the differences of the original data. To assess the quality of the fit, use `tsdiag(fit1)`, which shows the residuals from a fitted model, their correlogram, and P-values for an overall or portmanteau goodness of fit test. Try fitting various models, and say which you think is a good compromise between simplicity and fit. Discuss the interpretation of your chosen model.

(Section 6.4)

4. `lake`, defined below, gives data on major freezes of Lake Constance for the years 875–1974 AD; in each the 7–14km wide upper lake could be crossed on foot or by vehicle. The code plots the cumulative number of freezes, gives a kernel estimate of the intensity, and produces an exponential probability plot of the intervals between them. Do you think they follow a homogeneous Poisson process? Give reasons.

```
lake <- c(
875,895,
928,
1074,1076,
1108,
1217,1227,1277,
1323,1325,1378,1379,1383,
1409,1431,1435,1460,1465,1470,1479,1497,
1512,1553,1560,1564,1565,1571,1573,
1684,1695,
1763,1776,1788,1796,
1830,1880,
1963
)
n <- length(lake)
u <- c(0:(n-1),n-1)
par(pty="s",mfrow=c(2,2))
plot(c(lake,1974),u,xlim=c(800,1974),ylim=c(0,40),
     ylab="Cumulative number of freezes", xlab="Year",type="s")
d <- density(lake,from=875,to=1974)
d$y <- d$y*length(lake)/(1974-875)
plot(density(lake))  # kernel intensity estimate

d <- diff(c(lake,1974))
qqexp(d, line=TRUE)
```

We now fit a Poisson process model with intensity given by $\log \lambda(t) = \theta_1 + \theta_2(t - \theta_3)^2$. The integral

$$\int_{875}^{1974} \lambda(t)\, dt$$

is approximated by summation, though of course more sophisticated approaches are possible, including exact computation — particularly easy when $\theta_2 < 0$.

```
logL <- function(theta, d=c(lake,1974))
{
  n <- length(d)-1
  u <- 0.01*seq(from=min(d),to=max(d),by=1)
  lambda <- exp(theta[1]+theta[3]*(u-theta[2])^2 )
  sum(lambda) - sum(log(lambda[d[1:n]-d[1]+1]))
}
logL( c(0.1,15,-10^(-5)) )
fit <- nlm(logL, c(1.25,15,-5*10^(-2)), hessian=TRUE,d=c(lake,1974))
fit$estimate   # estimates
sqrt(diag(solve(fit$hessian)))  # standard errors
```

Note how the parameters are rescaled to avoid numerical difficulties. Give a careful interpretation of the estimates and their standard errors. Is it clear that the intensity function is not constant?

To transform the data to uniformity, graph them on the uniform scale, and test uniformity with a Kolmogorov–Smirnov test:

```
U <- function(t, theta, t.bot=875, t.top=1974, step=1)
{
  u <- 0.01*seq(from=t.bot, to=t.top, by=step)
  lambda <- exp(theta[1]+theta[3]*(u-theta[2])^2 )
  sum(lambda[((100*u)<=t)])/sum(lambda)
}
t.trans <- c(lake,1)
for (i in 1:n) t.trans[i] <- U(lake[i],fit$estimate)
plot(t.trans,c(0:n)/n,xlim=c(0,1),
     ylab="Cumulative distribution", xlab="Transformed time",type="s")
abline(0,1,lty=2)
ks.test(t.trans,"punif")  # Kolmogorov-Smirnov test of uniformity
```

Does the model fit these data well? Do you think such a model can be a reasonable basis for extrapolation?

(Section 6.5.1; Steinijans, 1976)

5. `ftse` contains data from Example 6.23 on the Financial Times Stock Exchange index of daily London closing prices from 1991–1998. The data in `ftse` are the daily percentage returns $y_t = 100 \log(x_t/x_{t-1})$, where x_t is the closing price on day t. To see the original time series and the return series:

```
par(mfrow=c(1,2))
plot(exp(cumsum(ftse/100)),main="Closing prices",type="l",ylab="Closing price")
plot(ftse,main="Daily returns (%)")
```

We fit extreme-value models using the `R` library `evir`:

```
library(evir)   # may need to be installed
meplot(ftse)  # sample mean residual life (= mean excess) plot
```

This plot has a clear change at around $u = 1$, and thresholds greater than this seem indicated. To check this, we see how the estimated shape parameter varies with u:

```
shape(ftse)
```

Here values $u > 0.8$ seem indicated. We fit the generalized Pareto distribution to exceedances of $u = 1$:

```
fit <- gpd(ftse,threshold=1)
plot(fit)
```

This threshold seems a little low for a good fit, so we try $u = 1.5$:

```
fit <- gpd(ftse,threshold=1.5)
plot(fit)
```

Does this seem adequate? Try varying u and seeing how the fit changes. When you are satisfied,

```
quant(ftse)
```

shows how estimates of the 0.99 quantile change as a function of u: seems to make little difference in this case.

Repeat this for the lower tail, by applying the calculations above to `-ftse`, and discuss your conclusions.

(Section 6.4)

Chapter 7

Theory

1. `paulsen` contains data collected as part of an investigation into the quantal nature of neurotransmission in the brain, by Dr O. Paulsen of the Department of Pharmacology, University of Oxford, in collaboration with Professor P. Heggelund of the Department of Neurophysiology, University of Oslo. Two models have been proposed to explain such data. The first model suggests that the data are drawn from an underlying skewed unimodal distribution. The alternative model suggests that the data are drawn from a series of distributions with modes equal to integer multiples of a unit size. To get some insight into this, we make density estimates with various bandwidths h:

```
library(MASS)  # Venables-Ripley library
library(boot)  # remove this when paulson is in SM library
attach(paulsen)
h <- 1.5
hist(y,probability=TRUE,breaks=c(0:30))
lines(density(y,bw=h,from=0,to=30))
rug(y)
h.lcv <- ucv(y)
h.lcv # value of h chosen by least squares cross-validation
lines(density(y,bw=h.lcv,from=0,to=30),lty=2)
h.bcv <- bcv(y)
h.bcv # value of h chosen by biased cross-validation
lines(density(y,bw=h.bcv,from=0,to=30),lty=2)
h.SJ <- width.SJ(y)
h.SJ # value of h chosen by Sheather-Jones procedure (preferable to CV)
lines(density(y,bw=h.SJ,from=0,to=30),lty=2)
```

   Try other values of h. How large must it be before a unimodal estimate is obtained? Which theory do you think most plausible? Discuss how you might construct a test of the theory predicting unimodality.

   (Section 7.1.2; Paulsen and Heggelund, 1994; Silverman, 1981)

2. (a) Consider a kernel density estimator (7.4) Show that its exact integrated mean squared error (7.9) may be expressed as

$$\begin{aligned}\mathrm{IMSE}(h) &= \frac{1}{nh}\int w(u)^2\,du + (1-n^{-1})\left[\int \{(w_h * f)(u)\}^2\,du\right]^2 \\ &\quad -2\int (w_h * f)(u) f(u)\,du + \int f(u)^2\,du,\end{aligned}$$

where $w_h(u) = h^{-1}w(u/h)$ and $(w_h * f)(u) = \int w_h(x-y)f(y)\,dy$ is the convolution of w_h and f.

(b) Mixtures of normal densities

$$f(y) = \sum_{i=1}^{k} \frac{\pi_i}{\sigma_i}\phi\left(\frac{y-\mu_i}{\sigma_i}\right)$$

can take a wide variety of forms and are useful because exact calculations for kernel density estimation are possible. To see this, show (i) that[1]

$$\frac{1}{h\sigma}\int \phi\left(\frac{y-u}{h}\right)\phi\left(\frac{u-\mu}{\sigma}\right)\,du = \frac{1}{(\sigma^2+h^2)^{1/2}}\phi\left\{\frac{y-\mu}{(h^2+\sigma^2)^{1/2}}\right\},$$

and (ii) deduce that IMSE(h) for a normal mixture f using the normal kernel $w_h(u) = h^{-1}\phi(u/h)$ equals

$$\frac{1}{2nh\pi^{1/2}} + (1-n^{-1})U(h,2) - 2U(h,1) + U(h,0),$$

where

$$U(h,q) = \sum_{i=1}^{k}\sum_{l=1}^{k} \frac{\pi_i\pi_l}{(\sigma_i^2+\sigma_l^2+qh^2)^{1/2}}\phi\left\{\frac{\mu_i-\mu_l}{(\sigma_i^2+\sigma_l^2+qh^2)^{1/2}}\right\}.$$

(iii) Otherwise, or by explicit calculation, show that the integrated squared bias of $\widehat{f}$ equals $U(k,2) - 2U(k,1) + U(k,0)$. Does the variance depend on n? Does the bias?

(c) Here is some code to make normal mixture densities, followed by plots of five example densities:

```
mix <- function(y, pi=1, mu=0, sig=1)
{ out <- rep(0,length(y))
  for (i in 1:length(pi)) out <- out + pi[i]*dnorm(y,mu[i],sig[i])
  out }
y <- seq(from=-3,to=3,length=1000)
plot(y,mix(y),type="l",ylab="f(y)")   # standard normal
pi2 <- c(1/5,1/5,3/5); mu2 <- c(0,1/2,13/12); sig2 <- c(1,2/3,5/9)
plot(y,mix(y,pi2,mu2,sig2),type="l",ylab="f(y)")  # skewed unimodal
pi3 <- c(1/2,1/2); mu3 <- c(-3/2,3/2); sig3 <- c(1/2,1/2)
plot(y,mix(y,pi3,mu3,sig3),type="l",ylab="f(y)")  # separated bimodal
pi4 <- c(3/4,1/4); mu4 <- c(0,3/2); sig4 <- c(1,1/3)
plot(y,mix(y,pi4,mu4,sig4),type="l",ylab="f(y)")  # asymmetric bimodal
pi5 <- c(1/2,rep(0.1,5)); mu5 <- c(0,(0:4)/2-1); sig5 <- c(1,rep(0.1,5))
plot(y,mix(y,pi5,mu5,sig5),type="l",ylab="f(y)")  # claw
```

`sim.mix` simulates from such a density. To use it to generate data and then estimate using a kernel estimator, with bandwidth chosen by cross-validation:

```
library(MASS)
sim.mix <- function(n, pi=1, mu=0, sig=1)
{ i <- sample(1:length(pi),size=n,replace=TRUE,prob=pi)
```

[1] Recall Exercise 3.2.16 and note the hint for Exercise 7.1.5.

```
  rnorm(n,mu[i],sig[i])  }
y <- sim.mix(1000, pi3, mu3, sig3)
plot(density(y,ucv(y)),type="l") # cross-validation bandwidth
lines(density(y,bcv(y)),type="l",lty=2) # biased cross-validation
lines(density(y,width.SJ(y)),type="l",lty=3) # Sheather-Jones bandwidth
```

Try repeating the last four lines with different n and mixtures, and seeing how well the three bandwidth estimates do. Which do you prefer?

To see how the theoretical integrated squared bias and integrated mean squared error depend on h, n and the mixture:

```
u <- function(h, q, pi=1, mu=0, sig=1)
{ k <- length(pi)
  out <- 0
  for (i in 1:k) for (j in 1:k)
  out <- out + pi[i]*pi[j]*dnorm(mu[i]-mu[j],0,sqrt(sig[i]^2+sig[j]^2+q*h^2))
  out }
isb <- function(h, pi=1, mu=0, sig=1)
        u(h,2,pi,mu,sig) - 2*u(h,1,pi,mu,sig) + u(h,0,pi,mu,sig)
imse <- function(h, n, pi=1, mu=0, sig=1)
        1/(n*h*3.545) + isb(h,pi,mu,sig) - u(h,2,pi,mu,sig)/n
log.h <- seq(from=-1.2,to=0.2,length=20) # h best viewed on log scale
isb1 <- isb( 10^log.h )
imse1 <- imse( 10^log.h, 100 )   # N(0,1), sample size 100
plot(log.h, isb1,type="l")
lines(log.h, imse1)
lines(log.h, imse1-isb1)
```

Note how the bias and variance balance each other out. Try repeating this with other n and the other mixtures above.

The IMSE for the fifth mixture (the claw) can have local minima. To see this:

```
isb5 <- isb(10^log.h,pi5,mu5,sig5)
imse5 <- imse(10^log.h,53,pi5,mu5,sig5)  # n = 53
plot(log.h, isb5,type="l")
lines(log.h, imse5)
lines(log.h, imse5-isb5)
```

Try this with smaller and larger n. Try simulating from this density, and using the two bandwidths. To what do they correspond? What do you conclude?

(Section 7.1.2; Marron and Wand, 1992; Sheather and Jones, 1991)

3. `beetle` contains the numbers of Japanese beetle larvae in the upper foot of soil of a 18×8 array of foot-square plots in a field of maize. The maize plants were planted in rows four feet apart.

   A possible model is that these counts are independent Poisson variables. Discuss what alternatives might be sensible here, and outline how you might construct Monte Carlo exact tests against them. Perform tests aaginst what seem to you the most plausible alternatives, and state carefully your conclusions.

   (Section 7.3)

4. Show that the exact upper α confidence limit μ^α for the mean μ of a Poisson variable Y satisfies

$$\Pr(Y \leq y; \mu) = \sum_{u=0}^{y} \frac{\mu^u}{u!} e^{-\mu},$$

and by arguing that this is the probability of seeing y or fewer events in the interval $[0, \mu]$ in a Poisson process of unit rate, show that μ^α is the $1-\alpha$ quantile of the gamma distribution with shape parameter $y+1$ and unit scale, $g_{y+1}^{1-\alpha}$, say. Show that the lower α confidence limit is g_y^α when $y > 0$ and zero otherwise. Hence check that the functions `exact` and `iexact` below give the limits of the exact interval and indicate whether μ is contained in it. Try `exact` for a few values of y.

```
exact <- function(y, a=0.025)
         c(switch(1+(y>0),0,qgamma(a,y)),qgamma(1-a,y+1))
iexact <- function(y, mu, a=0.025)
         ((ppois(y,mu)>a)&(ppois(y-1,mu)<1-a))
```

Find the score, maximum likelihood and signed likelihood ratio statistics for tests on μ, and check this code:

```
mle <- function(y, mu) ifelse(y>0,(y-mu)/sqrt(y),-Inf)
z <- function(y, mu)
     sign(y-mu)*ifelse(y>0,sqrt(2*(y*log(y/mu)+mu-y)),sqrt(mu))
zs <- function(y, mu) z(y, mu)+log(mle(y,mu)/z(y,mu))/z(y,mu)
```

We now compute the exact coverages for these intervals with $0 \leq \mu \leq 50$.

```
y <- 0:400;  eps <- 10^(-8); a <- 0.025; za <- qnorm(1-a)
mu <- seq(from=eps,to=50,length=1000)
cov <- matrix(NA,length(mu),4)
for (i in 1:length(mu)) {
   cov[i,1] <- sum( dpois(y,mu[i])[iexact(y,mu[i])] )
   cov[i,2] <- sum( dpois(y,mu[i])[abs(mle(y,mu[i]))<=za] )
   cov[i,3] <- sum( dpois(y,mu[i])[abs(z(y,mu[i]))<=za] )
   cov[i,4] <- sum( dpois(y,mu[i])[abs(zs(y,mu[i]))<=za] )  }
plot(mu,cov[,1],type="l",ylim=c(0.7,1),ylab="Exact coverage")
abline(h=1-2*a)  # nominal level
lines(mu,cov[,2],lty=2) # MLE
lines(mu,cov[,3],lty=3) # signed LR statistic
lines(mu,cov[,4],lty=4) # z*
```

Are the qualitative conclusions the same as those in Example 7.38?

(Section 7.3.4)

Chapter 8

Linear Models

1. `trees` contains data on the volume, height and girth (diameter) of 31 felled black cherry trees; girth is measured four feet six inches above ground. The problem is to find a simple formula for predicting volume from height and girth.

   ```
   data(trees)
   pairs(trees,panel=panel.smooth)
   pairs(log(trees),panel=panel.smooth)
   ```

   `coplot` generates conditioning plots, in which the relationship between two variables is displayed conditional on subsets of values of other variables. This is useful to see if the relationship is stable over the range of other variables. To assess this for the relationship of log volume and log girth, conditional on height:

   ```
   attach(trees)
   coplot(log(Volume)~log(Girth)|Height,panel=panel.smooth)
   ```

   Try this on the orginal scale also.

   For an initial fit, we take a linear model and assess model fit using diagnostic plots:

   ```
   summary(fit <- glm(Volume~Girth+Height))
   plot.glm.diag(fit)
   ```

   What do you make of the fit?

   To assess the possibility of transformation:

   ```
   library(MASS)
   boxcox(fit)
   ```

   Both $\lambda = 1$ and $\lambda = 0$ lie outside the confidence interval, though the latter is better supported. One possibility is to take $\lambda = 1/3$, corresponding to response $\texttt{Volume}^{1/3}$. What transformations for `Girth` and `Height` are then needed for dimensional compatibility? Fit this model, give interpretations of the parameter estimates, and discuss its suitability.

   An alternative is to suppose that a tree is conical in shape, in which case $\texttt{Volume} \propto \texttt{Height} \times \texttt{Girth}^2$. Equivalently, we fit

```
summary(fit <- glm(log(Volume)~log(Girth)+log(Height)))
library(boot)
plot.glm.diag(fit)
```

Are the parameter estimates consistent with this model? Does it fit adequately? What advantage has it over the others for prediction of future volumes?

(Chapter 8; Atkinson, 1985, p. 63)

2. `salinity` contains $n = 28$ observations on the salinity of water in Pamlico Sound, North Carolina. The response `sal` is the bi-weekly average of salinity. The next three columns contain values of the covariates, respectively a lagged value of salinity `lag`, a trend indicator `trend`, and the river discharge `dis`. Using the techniques of the previous problem as a guide, find a model suitable for prediction of salinity from the covariates. The data contain at least one outlier.

   (Chapter 8; Ruppert and Carroll, 1980; Atkinson, 1985, p. 48)

3. `burt` contains IQ scores `y` and `x` for pairs of identical twins, the first raised by foster parents and the second raised by natural parents.[1] Cases are divided into groups according to parents' social class, A, B and C, labelled 1, 2 and 3. The general objective is to assess the impact of social class on IQ, and in particular the effect of environment. A simple approach is to regard the second twin's IQ x as a standard with which to compare the first twin's IQ, y. Then the simplest models are those relating y to x through linear equations of the form $y = \alpha + \beta x + \varepsilon$, where ε is a random error and α and β are coefficients which may depend on social class. In the absence of environmental and social class effects, one would expect that $\alpha = 0$ and $\beta = 1$. To inspect the data:

   ```
   attach(burt)
   par(pty="s")
   plot(x,y,type="n",xlim=c(60,140),ylim=c(60,140))
   text(x,y,class,cex=0.8)
   abline(0,1,lty=2)
   ```

   Do linear models seem reasonable? To fit the full model, and print the corresponding analysis of variance table:

   ```
   burt.glm <- glm(y~x*class,data=burt)
   anova(burt.glm,test="F")
   library(boot)
   burt.diag <- glm.diag(burt.glm)
   ```

   The analysis of variance suggests no effect of class after adjusting for x and no interaction of class and x. To examine the coefficients, we reformulate the model in form $y = \alpha_c + \beta_c x + \varepsilon$:

[1] Cyril Burt (1883–1971) was educated at the universities of Oxford and Würzburg, and worked for London County Council and later the University of London. He was the first psychologist to be knighted. He founded the field of educational psychology in Britain, helped to establish the Eleven-Plus testing program, expanded the statistical technique of factor analysis, and investigated differences in intelligence among social classes, gender and race. He was a strong advocate for the view that intelligence is an inherited characteristic, though there is good evidence that he fabricated data on identical twins to support his position; see `www.indiana.edu/~intell/burt.shtml`.

```
summary(fit <- glm(y~class+x:class-1))
```

which shows striking similarities among the $\widehat{\beta}_c$.

To fit and examine the model $y = \alpha + \beta x + \varepsilon$:

```
summary(fit <- glm(y~x))
plot.glm.diag(fit)
```

What do you conclude?

(Section 8.5)

4. When is a condition number worrying? For a small simulation to compare with Example 8.28, we consider the condition number using the correlation matrix of the covariates.

```
cond <- function(X) # computes condition number
{ X <- as.matrix(X)
  ei <- eigen(cor(X))$values
  sqrt(max(ei)/min(ei)) }
cond(cement[,1:4])  # condition number for cement data
R <- 1000
sim.cond <- rep(NA,R)
for (r in 1:R)  sim.cond[r] <- cond(matrix(rnorm(13*4),13,4))
summary(sim.cond)
```

Modify `cond` for the other definitions of condition number, and repeat the experiment. What do you conclude about the cement data?

(Section 8.7.2)

5. To assess the properties of stepwise model selection methods, we use the function `sim`, which generates data with n responses that are independent of the five covariates. generates

```
sim <- function(n)
{ X <- scale(matrix(rnorm(n*5),n,5))
  y <- 4+rnorm(n)
  data.frame(y,x1=X[,1],x2=X[,2],x3=X[,3],x4=X[,4],x5=X[,5]) }
```

We now repeat stepwise model selection R times, and count how often the correct model (with just one covariate, the overall mean) is obtained:

```
library(MASS)
R <- 100; n <- 10; co <- rep(NA,R)
for (r in 1:R) {
s <- step(lm(y~x1+x2+x3+x4+x5,data=sim(n)),direction="both",trace=0)
co[r] <- length(coef(s)) }
table(co)    # frequencies of number of covariates
```

Note the high proportion of fits with three or more spurious covariates. Repeat this with $n = 20$, 50, 100. Do the properties of `step` improve? Discuss your findings.

(Section 8.7.3)

6. `pollution` contains data on weather (variables 1–3, 15), socio-economic factors (variables 4–11), and pollution (variables 12–14) for 60 Standard Metropolitan Statistical Areas in the USA. The response (variable 16) is the age-adjusted mortality rate from all causes, expressed as deaths per 100,000 persons. For an initial look at the data:

```
pairs(pollution)
pairs(pollution[,c(1:3,15:16)])   # association of mortality with weather
pairs(pollution[,c(4:11,16)])     # and social factors
pairs(pollution[,c(12:14,16)])    # and pollution measures
```

Examine these plots carefully, and comment. Are there outliers? Should covariates and/or the response be transformed? What difficulties might arise in accounting for the effect of air pollution on mortality?

Try using `step` to eliminate weather and social variables from the regression:

```
fit <- step(glm(mort~.-hc-nox-so,data=pollution))
boxcox(fit)
plot.glm.diag(fit)  # model adequate?
fit <- update(fit,log(mort)~.)  # try log transform of response
plot.glm.diag(fit)  # model adequate?
```

Should all the variables be included? Try various models, choose one or perhaps a few that you think are similarly adequate, give careful interpretations of the covariate effects, and discuss their plausibility. Check the adequacy of your model.

For an initial assessment of the relation between the pollution variables and mortality, after adjustment for the other variables, we use `resid` and `lm` to make added variable plots:

```
pairs(resid(lm(cbind(log(mort),hc,nox,so)~.,data=pollution)))
```

The top line of this scatterplot matrix contains the added variable plots for log mortality and the pollution variables. What difficulties do you foresee for regression on all three pollution variables? Are outliers present? Try adding in these variables, or suitable transformations of them, to your chosen best model (or models) from above, and discuss the interpretation and fit of the various models.

One possible approach to dealing with some of the problems above would be to use ridge regression. Try using the `ridge.lm` function, for example by:

```
rfit <- lm.ridge(mort~.-hc-nox,data=pollution,lambda=seq(0,20,0.01))
plot(rfit)
select(rfit)
```

Discuss the interpretability of the resulting parameter estimates.

Try using the functions `lqs` in `library(lqs)` for least trimmed squares regression, and `rlm` in `library(MASS)` for robust M-estimation, and see if your conclusions change.

(Section 8.7; McDonald and Schwing, 1973)

Chapter 9

Designed Experiments

1. `chicks` contains the data of Table 9.8. To fit the linear model given in Example 9.4 without first eliminating the pair parameters by differencing, display the analysis of variance table, and view the estimates:

```
chicks   # print data
summary(aov(y~Pair+Treat,data=chicks))  # analysis of variance
options(contrasts=c("contr.treatment","contr.poly"))
chicks.lm <- lm(y~Pair+Treat,data=chicks)   # fit linear model
summary(chicks.lm,correlation=F)  # summary w/o correlations of estimates
```

   The `options` statement gives the parametrization used in the example. Note the parameter estimates for the amino acids and their standard errors, and the estimate of error variance.

   To perform the intra- and inter-block analyses directly, we make two new dataframes `chick1` and `chick2`:

```
attach(chicks)
i <- 2*c(1:15)
chick1 <- data.frame(y=y[i]-y[-i],
          His=(Treat[i]=="His-")-(Treat[-i]=="His-"),
          Arg=(Treat[i]=="Arg-")-(Treat[-i]=="Arg-"),
          Thr=(Treat[i]=="Thr-")-(Treat[-i]=="Thr-"),
          Val=(Treat[i]=="Val-")-(Treat[-i]=="Val-"),
          Lys=(Treat[i]=="Lys-")-(Treat[-i]=="Lys-"))
chick2 <- data.frame(y=y[-i]+y[i],
          His=(Treat[i]=="His-")+(Treat[-i]=="His-"),
          Arg=(Treat[i]=="Arg-")+(Treat[-i]=="Arg-"),
          Thr=(Treat[i]=="Thr-")+(Treat[-i]=="Thr-"),
          Val=(Treat[i]=="Val-")+(Treat[-i]=="Val-"),
          Lys=(Treat[i]=="Lys-")+(Treat[-i]=="Lys-"))
detach("chicks")
chick1   # new data frame suitable for intra-block fit
chick1.lm <- glm(y~Arg+His+Thr+Val+Lys-1,data=chick1)
summary(chick1.lm)
chick2   # new data frame suitable for inter-block fit
chick2.lm <- glm(y~Arg+His+Thr+Val+Lys,data=chick2)
summary(chick2.lm)
```

Again note the estimates and their standard errors. Why are the estimates of error variance different for these fits and that above?

To compute the intra-block effects on the original scale, and 95% confidence intervals:

```
nu <- 10
est <- matrix(coef(chick1.lm),5,3)
chick1.sum <- summary(chick1.lm)
se <- sqrt(diag(chick1.sum$cov.unscaled*chick1.sum$dispersion))
10^( est + outer(se*qt(0.975,df=nu),c(0,-1,1)) )
```

For the pooled analysis:

```
t1 <- coef(chick1.lm)
s1 <- sqrt(diag(chick1.sum$cov.unscaled*chick1.sum$dispersion))
chick2.sum <- summary(chick2.lm)
t2 <- coef(chick2.lm)[-1]
s2 <- sqrt(diag(chick2.sum$cov.unscaled*chick2.sum$dispersion))[-1]
w <- 1/s1^2/(1/s1^2+1/s2^2)
est <- w*t1 + (1-w)*t2
s <- sqrt(1/(1/s1^2+1/s2^2))
cbind(est,s)
```

To compute the degrees of freedom using the calculation in Exercise 9.2.3 and then to compute the estimates and confidence intervals on the original scale:

```
n1 <- 10; n2 <- 9; s1 <- s1[1]; s2 <- s2[1]
nu <- (s1^2+s2^2)^2/(s1^4/n1+s2^4/n2)
nu # equivalent df for variance estimate
10^( est + outer(s*qt(0.975,df=nu),c(0,-1,1)) )
```

The balanced nature of the experiment is seen by computing the $X^{\mathrm{T}}X$ matrix for the intra-block model, and its inverse:

```
X <- as.matrix(chick1[,-1])
X
crossprod(X)
solve(crossprod(X))
```

The effect of a lack of balance is seen by dropping certain pairs:

```
solve(crossprod(X[-15,]))
```

Explain the change to the matrix $(X^{\mathrm{T}}X)^{-1}$ in terms of which treatment pairs have here been dropped from the model.

(Section 9.2; Cox and Snell, 1981, pp. 95–97)

2. An experiment was conducted with millet to determine the optimum distance between plants in rows; the rows were one foot apart. The treatments were distances 2, 4, 6, 8 and 10 inches between plants. The design was a 5×5 Latin square, and the data in `millet` are the average yields (gm) of three central rows 15 feet long after allowing for discards, from each plot.

   To inspect the data:

```
attach(millet)
par(mfrow=c(2,2))
plot(row,y)
plot(col,y)
plot(dist,y)
```

What do you infer? To fit the usual linear model:

```
millet.lm <- lm(y~row+col+dist,data=millet)
aov(millet.lm)
```

Verify that the analysis of variance is unchanged by fitting the terms in a different order, but not when the intercept is omitted — for example, compare the analyses of variance for the model above and

```
anova(lm(y~row+col+dist-1,data=millet),test="F")
anova(lm(y~dist+col+row-1,data=millet),test="F")
```

Carefully interpret and compare the estimates for the models

```
summary(millet.lm,correlation=F)
summary(lm(y~dist+row+col-1,data=millet),correlation=F)
```

and explain why the standard error for each of the distance effects is $(2s^2/5)^{1/2}$ for the first fit but not for the second.

To plot residuals, identify outliers, and refit without case 25, for example:

```
screen(4)
plot(fitted(millet.lm),resid(millet.lm))
identify(fitted(millet.lm),resid(millet.lm))
millet.lm <- lm(y~row+col+dist,data=millet,subset=c(-25))
```

How does dropping the outlier(s) affect the analysis of variance, the estimates, and their standard errors?

As in Example 9.12, the levels of distance d are quantitative, and it is useful to fit a polynomial in d:

```
millet.lm <- lm(y~poly(dist,4)+col+row,data=millet)
summary(millet.lm,correlation=F)
```

The coefficients here suggest that a quadratic polynomial will be adequate, and for ease of interpretation we do not fit the orthonormalized polynomials that would be generated by `poly(dist,2)`:

```
d <- as.vector(dist)
d2 <- d^2
millet.lm <- lm(y~d+d2+col+row,data=millet)
fit <- fitted(millet.lm)
md <- tapply(fit,dist,mean)
screen(3,new=F)
lines(2*c(1:5),md)
points(2*c(1:5),md,pch="+")
```

Use the coefficients of this polynomial model to estimate the value of d that maximizes the response; see Exercise 9.3.3.

How does the outlier affect this analysis?

(Section 9.3)

3. The data in `marking` are from an experiment designed to compare how different markers assessed examination scripts, in which four markers took part. Normally each marker had a different batch of scripts, but for the experiment one script was taken at random from each batch and replaced after three copies of it had been made. The three copies were sent to the other three markers who assessed them, while the original was replaced and assessed in the usual way. Each of the four copies was therefore assessed by a single marker, but the three markers who had a copy knew that the script was part of the experiment, while the person marking the original did not know it to be part of the experiment. The experiment was repeated at another examination, with the same examiners, but different scripts. Analyse the data and report on your conclusions.

   (Section 9.3; Lindley, 1961)

4. The data in `teak` are from an experiment on the growth of teak plants, in which two planting methods, pits (A) and crowbar holes (B), and three root lengths (4, 6 and 8 inches) were used. Each of the 2×3 factor combinations was applied to a plot chosen at random within each of four blocks, with the response y the average height in inches of the 50 plants grown on each plot. Analyse the data and report on your conclusions.

   (Section 9.2)

Chapter 10

Nonlinear Models

1. `shuttle` contains the data in Table 1.3 on O-ring failures for the space shuttle. To fit a binomial logistic regression model with covariate temperature:

   ```
   data(shuttle)
   fit <- glm(cbind(r,m-r)~temperature,data=shuttle,binomial)
   anova(fit)
   summary(fit)
   ```

   Try fitting with and without both covariates, and compare your output with Examples 4.27 and 4.33.

   To assess model fit, try `plot.glm.diag(fit)`. Do you find these diagnostics useful?

   (Sections 10.1–10.4; Dalal *et al.*, 1989)

2. `bliss` contains data on mortality of flour-beetles as a function of dose of a poison. To plot the death rates:

   ```
   bliss
   attach(bliss)
   plot(log(dose),r/m,ylim=c(0,1),ylab="Proportion dead")
   fit <- glm(cbind(r,m-r)~log(dose),binomial)
   summary(fit)
   points(log(dose),fitted(fit),pch="L")
   ```

   Does the fit seem good to you? Try again with the `probit` and `cloglog` link functions, for example:

   ```
   fit <- glm(cbind(r,m-r)~log(dose),binomial(cloglog))
   points(log(dose),fitted(fit),pch="C")
   ```

   Which fits best? Give a careful interpretation of the resulting model.

   (Sections 10.1–10.4; Bliss, 1935)

3. Table 10.11 and `ulcer` give data to compare five new surgeries (A, B, M, N, S) for stomach ulcer with a previous surgery (C), 'success' being non-occurrence of recurrent bleeding. Here is a likelihood analysis of the data, treated as 40 2×2 tables, in which we first fit a single parameter for all five new surgeries, and then separate parameters for each:

```
attach(ulcer)
ulcer
fit <- glm(cbind(r,m-r)~table+(treat!=" C")+treat,binomial,data=ulcer)
anova(fit,test="Chisq")
```

Note the large residual deviance and the apparent lack of evidence for differences among the new surgeries. For a look at diagnostic plots:

```
# library(boot)  # needs to be replaced eventually - errors for Pearson residual
diag <- glm.diag(fit)
glm.diag.plot(fit)
```

There are two obvious outliers, and printing `diag$res` shows that trial 39, which has 100% success rate for the new therapy and 0% for the old one, contributes them both. For now we ignore this, and make an adjustment for overdispersion:

```
fit <- glm(cbind(r,m-r)~table+(treat!=" C")+treat,quasibinomial)
anova(fit,test="Chisq")
```

How does the new analysis of deviance table compare with the old one? Do conclusions about significance of effects change?

We now analyse the data without study 39:

```
i <- c(1:80)[-c(39,79)]
fit <- glm(cbind(r,m-r)~table+(treat!=" C")+treat,quasibinomial,subset=i)
anova(fit,test="Chisq")
```

Now there seem to be differences among the new treatments, even after adjusting for the overdispersion. To see the coefficients:

```
fit <- glm(cbind(r,m-r)~table+treat,quasibinomial,subset=i)
s <- summary(fit)  # SEs adjusted for overdispersion
coef(s)
```

Note the estimate and standard error for study 20. Check the data to see what has happened. Note also the coefficients for the treatment effects: what do you conclude about treatment S? Note that trial 39 has treatment S. Whence does the evidence that S is the same as the other treatments come?

Now we merge levels A, B, N, and M of the factor `treat`, and refit:

```
tr <- as.character(ulcer$treat)
tr[tr=="B"] <- "A"   # merge levels A, B, N, M of treat
tr[tr=="N"] <- "A"
tr[tr=="M"] <- "A"
tr <- factor(tr)  # factor with merged levels
fit <- glm(cbind(r,m-r)~table+tr,quasibinomial,subset=i)
anova(fit,test="Chisq")
s <- summary(fit)
coef(s)
```

Does this model fit as well as the one with the original factors? Give 95% confidence intervals for the effect of the combined treatments and of S, on the odds scale.

Discuss how you might assess if the variance function is suitable.

(Sections 10.1–10.4, 10.6; Sacks *et al.*, 1990; Efron, 1996)

4. In 1961 and 1962 an experiment was conducted in Sweden to assess the effect of a speed limit on the accident rate on motorways. The experiment was conducted on 92 days in each year, matched so that day j in 1961 was comparable to day j in 1962. On some days the speed limit was in effect and enforced, and not on other days. The number of accidents was recorded daily.

   (a) Let Y_{ij} be the number of accidents on day j in 196i, and let I_{ij} indicate whether the speed limit was in effect that day. A simple model is that Y_{1j} has a Poisson distribution with mean $\lambda_j \exp(\beta I_{1j})$, while the corresponding mean for 1962 is $\lambda_j \exp(\alpha + \beta I_{2j})$. Show that there is a cut in the likelihood, and deduce that inference for α and β may be performed using the binomial conditional distribution of Y_{1j} given $Y_{1j} + Y_{2j}$, for $j = 1, \ldots, 92$; give this distribution.

   (b) The data are in `limits`. The following code fits and displays the binomial model:

```
data(limits)
limits.bin <- glm(cbind(y1,y2)~delta,binomial,data=limits)
summary(limits.bin)
plot.glm.diag(limits.bin)
```

   To fit the Poisson model:

```
attach(limits)
acc <- c(y1,y2)
d <- c(i61,i62)
detach("limits")
year <- rep(c(1,2),c(92,92))
day <- factor(rep(1:92,2))
limits.poi <- glm(acc~day+year+day,poisson)
summary(limits.poi)
```

   Compare the output from the two fits, and verify that the estimates of the year and speed limit effects and their standard errors are equal.

   (Sections 10.4–10.5; Svensson, 1981)

5. `lizards` contains the numbers of two species of lizard, *grahami* and *opalinus*, observed in Whitehouse, Jamaica, at two different heights, two perch diameters, in sun or shade, at three times of the day. The data were collected by observing occupied perches and recording the appropriate height, perch diameter, and so forth.

   Interest lies in the difference in habitat choice between the species. Analyze the data by approaches (i)-(iii) below, comparing their relative advantages and disadvantages, and paying particular attention to the interpretation of the models you choose.

   (i) A conventional analysis of variance.

   (ii) A generalized linear model with binomial error, based on the probability that a perch is occupied by one species, given that it is occupied. Discuss the similarities and disssimilarties with (i).

(iii) A log-linear model.

(Sections 10.4–10.5; Bishop *et al.*, 1975, p. 164)

6. `Titanic` contains data on survival after the shipwreck of the Titanic. How does survival depend on social class, age (adult/child), and sex?

   (Sections 10.4–10.5; Simonoff, 1998)

7. `seeds` contains data on the germination of seeds in a 2×2 factorial experiment in which two types of seed and two root extracts, bean and cucumber, were compared. Seeds were placed on a plate containing root extract and the numbers subsequently germinating were counted. For an initial fit of a standard binomial logistic regression:

```
data(seeds)
fit <- glm(cbind(r,m-r)~seed*root,binomial,data=seeds)
anova(fit,test="Chisq")
plot.glm.diag(fit)
```

   Both residual deviance and residuals are over-dispersed. To allow for this:

```
fit <- glm(cbind(r,m-r)~seed*root,quasibinomial,data=seeds)
anova(fit,test="F")
summary(fit)
plot.glm.diag(fit)
```

   To fit a beta-binomial model (Exercise 10.6.4), we set up a negative log likelihood for a random sample from the distribution, parametrized in terms of μ and δ:

```
nlogL <- function(th, r, m)
{  # computes negative log likelihood, ignoring constant terms
   mu <- th[1]
   del <- th[2]
   a <- mu*(1/del-1)
   b <- (1-mu)*(1/del-1)
   sum( lbeta(a,b)-lbeta(r+a,m-r+b) ) }
init <- c(0.5,0.1)
nlogL( init, r[1:5], m[1:5] )
fit <- nlm(nlogL, init, hessian=TRUE, r=r[1:5], m=m[1:5])  # fit to first group
fit$minimum   # negative of maximised log likelihood
fit$estimate  # parameter estimates
sqrt(diag(solve(fit$hessian)))  # and their standard errors
```

   Find the estimates and standard errors for the other groups. Do you think that they are a good summary of the data in this case? Do the data support constant δ for all four groups? Discuss your findings.

   (Section 10.6; Crowder, 1978; Cox and Snell, 1989, Section 3.2)

8. `coal` contains dates of coal-mining disasters in Britain from 1850–1962, in which more than ten people were killed. To see the cumulative number of disasters, and the numbers per year:

```
data(coal)
attach(coal)
par(mfrow=c(1,2))
plot(date,1:length(date),type="s",ylab="Cumulative number of accidents")
cc <- hist(date,breaks=1850:1963,main="Accidents per year, 1850-1962")
```

Do the times look homogeneous? When does change occur?

We assume that the data are a realization of an inhomogeneous Poisson process, and try estimating the rate using a generalized additive model:

```
library(mgcv)
fit <- gam(cc$counts~s(cc$mids),poisson)  # automatic choice of df
fit
lines(cc$mids,fitted(fit))
fit <- gam(cc$counts~s(cc$mids,df=3),poisson)  # 3 degrees of freedom
fit
lines(cc$mids,fitted(fit),lty=2)
```

Legislation on safety in mines was strengthened in 1872, and legislation on liability of employers for accidents was strengthened in 1878 and 1897. How does this knowledge affect the type of model you might fit? Try fitting suitable models, and discuss their fit (see also Practical 5.3).

What other factors, external to the data here, might affect the numbers of disasters?

(Section 10.7; Maguire *et al.*, 1952; Jarrett, 1979)

9. `urine` contains data in which the binary response indicates the presence or not of calcium oxalate crystals in 79 samples of urine; there are six covariates related to chemical and physical properties of the samples. Cases 1 and 55 have missing values. Analyse the data and report on your conclusions, paying particular attention to the interpretation of any final models. You might start from:

```
data(urine)
pairs(urine)
fit <- step(glm(r~.-case,subset=-c(1,55),data=urine,binomial))
```

Are transformation of covariates or smoothing needed?

(Section 10.4, 10.7; Andrews and Herzberg, 1985, pp. 249–251)

10. To fit the proportional hazards model to the AML data of Practical 5.2:

```
library(survival)
data(aml)
fit <- coxph(Surv(time,cens)~group,data=aml)
summary(fit)
plot(survfit(fit))  # estimated baseline survivor function
```

Compare this fit with those Practical 5.2, and discuss the reasons for any differences.

(Section 10.8; Miller, 1981, p. 49)

11. To analyze the motorette data of Table 11.10,

```
library(survival)
data(motorette)
fit <- survfit(Surv(y, cens)~x,data=motorette)
plot(fit,lty=1:4)
```

To fit models with Weibull responses:

```
fit.wei <- survreg(Surv(y, cens)~log(x),data=motorette)
summary(fit.wei)
fit.wei <- survreg(Surv(log(y), cens)~log(x),data=motorette)
summary(fit.wei)
```

Try variants of this model, using for example the log-logistic distribution, and discuss how you might construct residuals to assess their fit.

(Section 10.8; Nelson and Hahn, 1972)

12. `pbc` contains the data on primary biliary cirrhosis from Example 10.39. Analyse the data and report on your conclusions, starting perhaps from:

```
library(survival)
data(pbc)
pbcm <- pbc[(pbc$trt!=-9),]    # now clean up missing data
pbcm$copper[(pbcm$copper==-9)] <- median(pbcm$copper[(pbcm$copper!=-9)])
pbcm$platelet[(pbcm$platelet==-9)] <- median(pbcm$platelet[(pbcm$platelet!=-9)])
attach(pbcm)  # new dataframe  without missing data
plot(survfit(Surv(time,status)~trt),ylim=c(0,1),lty=c(1,2),
     ylab="Survival probability",xlab="Time (days)")
plot(survfit(coxph(Surv(time,status)~trt+strata(sex))),ylim=c(0,1),
     lty=c(1,2),ylab="Survival probability",xlab="Time (days)")
lines(survfit(coxph(Surv(time,status)~trt)),lwd=2)
```

Now try adding the covariates. Do you agree with the conclusions of Example 10.39?

(Section 10.8; Fleming and Harrington, 1991)

Chapter 11

Bayesian Models

1. When a coin is spun on its edge, the proportion of tails may differ appreciably from $\frac{1}{2}$ due to irregularities that do not matter when the coin is tossed in the air.

   (a) Find and inspect a coin, and without spinning it, use a mixture of beta densities to give your personal prior for the likely proportion of heads when it is spun, briefly outlining your reasoning. You may find it useful to compare plots of priors. You can do this as follows in R:

```
coin.spin <- function( para, r=0, n=0, n.points=199)
{   # compute posterior density for n coin spins of which r heads,
    # with prior in para
  para <- as.matrix(para)
  para[,1] <- para[,1]/sum(para[,1])
  k <- nrow(para)
  w <- rep(0,k)
  for (i in 1:k)
w[i] <- (para[i, 1] * beta(para[i, 2] + r, para[i, 3] + n - r))/
        beta(para[i, 2], para[i, 3])
  para[,1] <- w/sum(w)
  x <- c(1:n.points)/(n.points +1)
  y <- rep(0,n.points)
  for (i in 1:k)
    y <- y + para[i,1]*dbeta(x, shape1=para[i,2]+r, shape2=para[i,3]+n-r )
  list( x=x, y=y )
}
para <- matrix( c(0.5, 10, 20, 0.5, 20, 10), nrow=2, ncol=3, byrow=TRUE)
prior <- coin.spin(para)
plot(prior, xlab="theta",ylab="PDF", type="l",ylim=c(0,6))
```

   The matrix `para` consists of the probabilities attached to each component of the mixture (first column) and the parameters a and b for the mixture components (second and third columns); the number of rows of `para` is the number of components of the mixture. The function `coin.spin` calculates the corresponding mixture density.

   (b) Now find two or three other people and try to produce a group prior for the proportion of heads by combining your priors from (a). You might try to agree on a suitable mixture by discussion, or to use a mixture of your individual priors. Briefly summarize the difficulties you see in finding a prior that satisfactorily represents group beliefs.

(c) Take your coin and use the number of heads recorded in 10 spins to update your prior from (a). If you get $r = 4$ heads in the $n = 10$ spins:

```
post <- coin.spin(para, r=4, n=10)
lines(post, lty=2)
```

Produce a graph that compares your prior and posterior densities for θ.

(d) Can you use the information from your co-investigators' spins to further update your results from (c)? To what extent do your results support the initial assertion that spinning gives asymmetric outcomes?

(Sections 11.1, 11.2)

2. (a) Suppose that y is the number of events occurring when a Poisson process of rate λ is watched for a period of length x. Show that if λ has a gamma distribution with shape and scale parameters α and β, the marginal density of y is generalized negative binomial,

$$f(y \mid \alpha, \beta) = \frac{\Gamma(y+\alpha)}{y!\Gamma(\alpha)}(1-\pi)^y\pi^\alpha, \quad y = 0, 1, 2, \ldots,$$

where $\pi = \beta/(\beta + x)$.

(b) Now suppose that data pairs $(x_1, y_1), \ldots, (x_n, y_n)$ are available, and that the prior density for β is gamma with shape and scale parameters ν and ϕ; suppose also that α is known. The prior density of β when $\nu = \phi = 1$ is plotted as follows:

```
par(mfrow=c(2,2))
beta <- seq(from=0,to=10,length=1000)
nu <- phi <- 1
f <- dgamma(beta*phi,shape=nu)*phi
plot(beta,f,xlab="beta",ylab="Prior",type="l")
```

`cloth` give the numbers of faults y in lengths x of cloth, for which a reasonable model is given in (a). Compute the Laplace approximation for the posterior density of β, and check that the function below calculates this approximation:

```
poi.beta.laplace <- function( data, alpha=get.alpha(data), phi=1, nu=0.1,
                              beta=seq(from=0,to=7,length=1000) )
{  # compute Laplace approximation
  h <- function(x, P)
    P$phi*x - (P$n*P$alpha+P$nu-1)*log(x) + sum( (P$alpha+P$y)*log(P$X+x) )
  y <- data$y
  X <- data$x
  n <- length(data$y)
  P <- list(phi=phi, nu=nu, n=n, y=y, X=X, alpha=alpha)
  h.min <- optim(par=1, fn=h, method="L-BFGS-B",
                 hessian=TRUE, lower=0, upper=Inf, P=P)
  u <- h.min$parameters
  conv <- h.min$message
  H2 <- det(h.min$hessian)
  out <- sum( lgamma(alpha+y)-lgamma(y+1)-lgamma(alpha)+y*log(X)) -
       lgamma(nu) + nu*log(phi) -
```

```
      h.min$value+0.5*log( 2*pi/H2 )
  PDF <- beta
  for (i in 1:length(beta))
  PDF[i] <- - h(beta[i],P) - (- h.min$value + 0.5*log( 2*pi/H2 ) )
  int <- sum(exp(PDF))*(beta[2]-beta[1])

  list( int=int, conv=conv, x=beta, y=exp(PDF) )
}
get.alpha <- function(d)
{  # estimate alpha from data
  rho <- d$y/d$x
  n <- length(d$y)
  mean(rho)^2/( (n-1)*var(rho)/n - mean(rho)*mean(1/d$x) )
}
```

To plot the data, then use `poi.beta.laplace` and plot its output for $\nu = 1, 5$:

```
data(cloth)
attach(cloth)
plot(x,y)
beta.post <- poi.beta.laplace(cloth,beta=beta,nu=1)
plot(beta.post,type="l",xlab="beta",ylab="Posterior density")
beta.post <- poi.beta.laplace(cloth,beta=beta,nu=5)
lines(beta.post,lty=2)
```

(A crude value for the integral of the posterior density is given in `beta.post$int`.) Assess the sensitivity of the posterior to the values of ν and ϕ, and explain in qualitative terms how the posterior depends on these hyperparameters.

(Section 11.3.1; Gaver and O'Muircheartaigh, 1987)

3. (a) If U_1 and U_2 have conditional densities

$$\begin{aligned} f(u_1 \mid u_2) &= \frac{u_2 \exp(-u_2 u_1)}{1 - \exp(-u_2 B)}, \quad 0 < u_1 < B, \\ f(u_2 \mid u_1) &= \frac{u_1 \exp(-u_1 u_2)}{1 - \exp(-u_1 B)}, \quad 0 < u_2 < B, \end{aligned}$$

show that $\pi(u_1) \propto u_1^{-1}\{1 - \exp(-Bu_1)\}$, for $0 < u_1 < B$.

(b) Verify that the function below performs Gibbs sampling for this problem, and that `add.exp.lines` draws the density estimates based on equation (11.41):

```
exp.gibbs <- function(u1=NULL, u2=NULL, B, I=100, S=100 )
{
  H <- function(u0, B, S=ncol(u0) )
  {
    rexp.trunc <- function( S, lambda, B)
            -log( 1-runif( n=S, max=1-exp(-lambda*B)) )/lambda
    u1 <- rexp.trunc( S, u0[2,], B)
    u2 <- rexp.trunc( S, u1, B)
    rbind( u1, u2 )
```

```
  }
  out <- array(NA, dim = c(2, S, I))
  if (is.null(u1)) u1 <- runif(n=S, max=B)
  if (is.null(u2)) u2 <- runif(n=S, max=B)
  out[,,1] <- rbind( u1, u2 )
  for (i in 2:I)  out[,,i] <- H( out[,,i-1], B )
  out
}
add.exp.lines <- function( exp.out, i, B=10)
{
  dexp.trunc <- function( u, lambda, B )
     dexp(u, rate=lambda)/(1-exp(-lambda*B))
  S <- dim(exp.out)[2]
  I <- dim(exp.out)[3]
  u <- seq(0.0001,B,length=1000)
  fu <- rep(0,1000)
  for (s in 1:S) fu <- fu + dexp.trunc(u,exp.out[3-i,s,I],B)/S
  lines(u,fu,col="red")
  invisible()
}
```

To use these functions:

```
par(mfrow=c(3,2))
B <-10; I <- 15; S <- 500
exp.out <- exp.gibbs(B=B,I=I,S=S)
hist(exp.out[1,,I],prob=TRUE,nclass=15,xlab="u1",ylab="PDF",xlim=c(0,B),ylim=c(0,1))
add.exp.lines(exp.out,1)
hist(exp.out[2,,I],prob=TRUE,nclass=15,xlab="u2",ylab="PDF",xlim=c(0,B),ylim=c(0,1))
add.exp.lines(exp.out,2)
```

To see how the output of the sampler varies from run to run, try repeating

```
exp.out <- exp.gibbs(B=B,I=I,S=S)
hist(exp.out[1,,I],prob=TRUE,nclass=15,xlab="u1",ylab="PDF",xlim=c(0,B))
add.exp.lines(exp.out,1)
hist(exp.out[2,,I],prob=TRUE,nclass=15,xlab="u2",ylab="PDF",xlim=c(0,B))
add.exp.lines(exp.out,2)
```

What is the effect of different values of I and S?

(c) Show that if $B \to \infty$, no proper density can solve the integral equations in (a), and deduce that the Gibbs sampler does not converge. Assess what this means in practical terms by taking some large values of B in (b).

(Section 11.3.3; Casella and George, 1992)

4. Using Example 11.23 as a template, write code to generate Cauchy data using $N(u, \sigma^2)$ proposals centred on the current value u. For what values of σ is (a) the acceptance probability highest, (b) the autocorrelation between successive values lowest?

   (Section 11.3.3)

5. Use our previous discussions of the hierarchical Poisson model to give forms for the full conditional densities for the pumps data of Example 11.19, and use them to check that the function `poi.gibbs` below implements the Gibbs sampler for this problem.

```
poi.gibbs <- function(d, alpha, gamma, delta, I, S)
{
  poi.sim <- function(y, x, alpha, gamma, delta, theta)
  {
    n <- length(y)
    lambda <- theta[1:n]
    beta <- theta[n + 1]
    out1 <- rgamma(n, alpha + y)/(x + 1/beta)
    out2 <- (sum(out1) + delta)/rgamma(1, gamma + n * alpha)
    c(out1, out2)
  }
  n <- length(d$y)
  out <- array(NA, dim = c(I, S,n+1))
  out[1, ,  ] <- rexp((n + 1) * S)
  for(s in 1:S)
  for(i in 2:I)
   out[i,s,] <- poi.sim(d$y, d$x, alpha, gamma, delta, out[i - 1, s,])
  out
}
```

The output is an $I \times S \times (n+1)$ array, where I is the number of iterations, S the number of independent replicates of the sampler, and n the number of pumps. Run the sampler and assess how quickly it converges:

```
data(pumps)
system.time( pumps.sim <- poi.gibbs(pumps, alpha=1.8, delta=0.1, gamma=1,
             I=1000, S=5) )
```

puts an $I \times S \times (n+1)$ array of output into `pumps.sim`, and we can assess convergence using time series functions:

```
par(mfrow=c(2,3))
plot.ts(pumps.sim[,1,1])
acf(pumps.sim[,1,1])
pacf(pumps.sim[,1,1])
plot.ts(pumps.sim[,1,11])
acf(pumps.sim[,1,11])
pacf(pumps.sim[,1,11])
```

which gives the first replicate of the sampler for the parameters λ_1 and β. Use these and the other time series tools in Rto assess how quickly the simulation converges in this example, for the various parameters.

More sophisticated analysis is provided by the `coda` library, which can be downloaded from CRAN:

```
library(coda)
library(help="coda")  # list of functions for which help can be sought
pumps.mcmc <- mcmc(pumps.sim[,1,])
plot(pumps.mcmc)
effectiveSize(pumps.mcmc)
summary(pumps.mcmc)
```

`coda` includes functions for assessment of convergence, using the replicate series:

```
pumps.list <- mcmc.list(mcmc(pumps.sim[,1,]), mcmc(pumps.sim[,2,]),
          mcmc(pumps.sim[,3,]), mcmc(pumps.sim[,4,]), mcmc(pumps.sim[,5,]))
gelman.diag(pumps.list)
gelman.plot(pumps.list)
geweke.diag(pumps.list)
geweke.plot(pumps.list)
raftery.diag(pumps.list)
```

You will need to check the help files to understand the output from these functions. Make any modifications to the number of iterations that seem useful, and re-run the sampler for long enough for the output to be reliable.

(Section 11.3.3; Gaver and O'Muircheartaigh, 1987)

6. A simple model for the HUS data of Example 4.40 is that the counts $Y_1, \ldots, Y_n$ are independent Poisson variables with

$$\mathrm{E}(Y_j) = \begin{cases} \lambda_1, & j = 0, \ldots, \tau, \\ \lambda_2, & j = \tau + 1, \ldots, n. \end{cases}$$

For a Bayesian formulation, we suppose *a priori* that λ_1, λ_2 are independent gamma variables with parameters a_1, β_1 and a_2, β_2, where a_1 and a_2 are specified and β_1, β_2 are independent variables from the gamma density with specified parameters c and d, and τ is uniform on $1, \ldots, n$.

(a) Construct the directed acyclic and moral graphs for this model.

(b) Show that the full conditional distributions are

$$\begin{aligned} \lambda_1 \mid \beta_1, \tau, y &\sim G(a_1 + s_\tau, \tau + \beta_1), \\ \lambda_2 \mid \beta_2, \tau, y &\sim G(a_2 + s_n - s_\tau, n - \tau + \beta_2), \\ \beta_1 \mid \lambda_1 &\sim G(a_1 + c, \lambda_1 + d), \\ \beta_2 \mid \lambda_2 &\sim G(a_2 + c, \lambda_2 + d), \\ \Pr(\tau = k \mid \lambda_1, \lambda_2, y) &\propto \exp\left\{ s_k \log(\lambda_1/\lambda_2) + k(\lambda_2 - \lambda_1) \right\}, \quad k = 1, \ldots, n, \end{aligned}$$

where $s_\tau = y_1 + \cdots + y_\tau$ and $G(a, b)$ indicates the gamma density with parameters a and b, that is, $b^a y^{a-1} e^{-by} / \Gamma(a)$, for $y > 0$ and $a, b > 0$.

(c) Check that the following code applies this algorithm, with R iterations.

```
hus.gibbs <- function(init, y, R=10, a1=1, a2=1, c=0.01, d=0.01)
{  # Gibbs sampler starting from init with R iterations
  sim.tau <- function(y, lam1, lam2)
  {
```

```
  s <- cumsum(y)
  tau <- 1:length(y)
  p <- exp( s*log(lam1/lam2) + tau*(lam2-lam1) )
  sample(1:length(y), size=1, prob=p)
}
lik <- function(y, para, a1, a2, c, d)
{  # computes log likelihood, log prior at current parameters
  lam1 <- para[1]
  lam2 <- para[2]
  beta1 <- para[3]
  beta2 <- para[4]
  tau <- para[5]
  lam <- c(rep(lam1,tau),rep(lam2,length(y)-tau))
  L <- sum( dpois(y, lam, log=TRUE) )
  P <- dgamma(lam1, shape=a1, rate=beta1, log=TRUE) +
       dgamma(lam2, shape=a2, rate=beta2, log=TRUE) +
       dgamma(beta1, shape=c, rate=d, log=TRUE) +
       dgamma(beta2, shape=c, rate=d, log=TRUE)
  c(L, P)
}
step <- function(j, para, y, a1, a2, c, d)
{  # single step of random scan Gibbs sampler: update para[j]
  lam1 <- para[1]
  lam2 <- para[2]
  beta1 <- para[3]
  beta2 <- para[4]
  tau <- para[5]
  s.tau <- sum(y[1:tau])
  s.n <- sum(y)
  n <- length(y)
  sim <- switch(j,
           rgamma(1, shape=a1+s.tau, rate=tau+beta1),            # lam1
           rgamma(1, shape=a2+s.n-s.tau, rate=n-tau+beta2),      # lam2
           rgamma(1, shape=a1+c, rate=lam1+d),                   # beta1
           rgamma(1, shape=a2+c, rate=lam2+d),                   # beta2
           sim.tau(y, lam1, lam2))                               # tau
  para[j] <- sim
  para
}
paras.out <- matrix(init, R, length(init), byrow=TRUE)
log.lik <- lik(y, init, a1, a2, c, d)
lik.out <- matrix(log.lik, R, length(log.lik), byrow=TRUE)
for (r in 2:R)
{  j <- sample(1:5)  # random order for five possible updates
   para <- paras.out[r-1,]
   for (i in 1:length(j)) para <- step(j[i], para, y, a1, a2, c, d)
   paras.out[r,] <- para
   lik.out[r,] <- lik(y, para, a1, a2, c, d)  }
cbind(paras.out, lik.out)   # output is parameters, log likelihood, log prior
```

```
}
hus <- c(1,5,3,2,2,1,0,0,2,1,1,7,11,4,7,10,16,16,9,15)

system.time( gibbs.out <- hus.gibbs(c(5, 5, 1, 1, 2), hus, R=1000))
plot.ts(gibbs.out[,1], main="lambda1") # time series plot for lam1
plot.ts(gibbs.out[,2], main="lambda1") # time series plot for lam2
plot.ts(gibbs.out[,6], main="log lik") # and of log likelihood
table(gibbs.out[,5])  # tabulate observed values of tau
```

Run the code with different initial values and with different values for a_1, a_2, c and d. What do you conclude? Do your conclusions agree with those based only on the likelihoood function?

(d) Try this with different HUS data, from Newcastle over the same period:

```
hus <- c(6,1,0,0,2,0,1,8,4,1,4,0,4,3,3,13,14,8,9,19)
system.time( gibbs.out <- hus.gibbs(c(5, 5, 1, 1, 2), hus, R=10000))
plot.ts(gibbs.out[,1], main="lambda1") # time series plot for lam1
plot.ts(gibbs.out[,2], main="lambda1") # time series plot for lam2
plot.ts(gibbs.out[,6], main="log lik") # and of log likelihood
table(gibbs.out[,5])  # tabulate observed values of tau
```

What do you conclude?

(Section 11.3.3; Henderson and Matthews, 1993)

7. Consider Bayesian analysis of the data in Example 6.22, with an unknown changepoint. Suppose for now that consecutive observations are independent normal variables with variance σ^2 and means μ_1 for $j = 1, \ldots, \gamma$ and μ_2 for $j = \gamma + 1, \ldots, n$, and that we take uniform prior densities for μ_1 and μ_2, that $\sigma^2 \sim IG(a, b)$ for specified a and b, and that γ is assumed uniform on $\{1, \ldots, n-1\}$ *a priori*.

(a) Show that

$$\pi(\mu_1, \mu_2, \sigma^2, \gamma \mid y) \propto (\sigma^2)^{-n/2-a-1} \exp\left\{-h(\mu_1, \mu_2, \sigma^2, \gamma)\right\}, \gamma = 1, \ldots, n-1,$$

where

$$\begin{aligned} 2\sigma^2 h(\mu_1, \mu_2, \sigma^2, \gamma) &= 2b + \sum_{j=1}^{\gamma}(y_j - \mu_1)^2 + \sum_{j=\gamma+1}^{n}(y_j - \mu_2)^2 \\ &= 2b + \sum_{j=1}^{\gamma}(y_j - \overline{y}_1)^2 + \sum_{j=\gamma+1}^{n}(y_j - \overline{y}_2)^2 \\ &\quad + \gamma(\overline{y}_1 - \mu_1)^2 + (n-\gamma)(\overline{y}_2 - \mu_2)^2, \end{aligned}$$

with $\gamma\overline{y}_1 = \sum_{j=1}^{\gamma} y_j$ and $(n-\gamma)\overline{y}_2 = \sum_{j=\gamma+1}^{n} y_j$. Hence find the full conditional densities for the four parameters.

(b) Complete this code, which runs a Gibbs sampler for the parameters:

```
beaver.gibbs <- function(init, y, R=10, a=1, b=0.05)
{  # Gibbs sampler starting from init with R iterations
  sim.g <- function(y, mu1, mu2, lam)
```

```
  {
    n <- length(y)
    p <- rep(NA,n-1)
    for (g in 1:(n-1))
     { mu <- c(rep(mu1,g),rep(mu2,n-g))
       p[g] <- sum((y-mu)^2) }
    sample(1:(n-1), size=1, prob=exp(-lam*p/2))
  }
  lik <- function(y, para, a, b)
  {  # computes log likelihood, log prior at current parameters
    mu1 <- para[1]
    mu2 <- para[2]
    lam <- para[3]
    g <- para[4]
    n <- length(y)
    mu <- c(rep(mu1,g),rep(mu2,n-g))
    L <- sum( dnorm(y, mu, sd=sqrt(1/lam), log=TRUE) )
    P <- dgamma(lam, shape=a, rate=b, log=TRUE)
    c(L, P)
  }
  step <- function(j, para, y, a, b)
  {  # single step of random scan Gibbs sampler: update para[j]
    mu1 <- para[1]
    mu2 <- para[2]
    lam <- para[3]
    g <- para[4]
    n <- length(y)
    mu <- c(rep(mu1,g),rep(mu2,n-g))
    SS <- sum((y-mu)^2)
    sim <- switch(j,
              rnorm(1, mean(y[1:g]), 1/sqrt(lam*g)),           # mu1
              rnorm(1, mean(y[(g+1):n]), 1/sqrt(lam*(n-g))),   # mu2
              rgamma(1, shape=a+n/2, rate=b+SS/2),             # lambda
              sim.g(y, mu1, mu2, lam))                         # gamma
    para[j] <- sim
    para
  }
  paras.out <- matrix(init, R, length(init), byrow=TRUE)
  log.lik <- lik(y, init, a, b)
  lik.out <- matrix(log.lik, R, length(log.lik), byrow=TRUE)
  for (r in 2:R)
  {  j <- sample(1:4)  # random order for four possible updates
     para <- paras.out[r-1,]
     for (i in 1:length(j)) para <- step(j[i], para, y, a, b)
     paras.out[r,] <- para
     lik.out[r,] <- lik(y, para, a, b)  }
  cbind(paras.out, lik.out)   # output is parameters, log likelihood, log prior
}
system.time( gibbs.out <- beaver.gibbs(c(36, 40, 3, 38), beaver$temp, R=1000))
```

```
par(mfrow=c(2,3))
plot.ts(gibbs.out[,1],main="mu1") # time series plot for mu1
plot.ts(gibbs.out[,2],main="mu2") # time series plot for mu2
plot.ts(gibbs.out[,3],main="lambda") # time series plot for lambda
plot.ts(gibbs.out[,4],main="gamma") # time series plot for gamma
plot.ts(gibbs.out[,5],main="log likelihood")  # and of log likelihood
```

(c) Choose a suitable R, run the Gibbs sampler, and use the output to give posterior standard errors and 90% posterior credible intervals for the changepoint γ, for the change in body temperature $\mu_2 - \mu_1$, and for σ^2. Does body temperature change before the beaver leaves the lodge, or at the same time?

(d) In fact successive values of y_j are positively autocorrelated. Discuss briefly how you would deal with this in practice, and describe qualitatively the effect it will have on your inference.

(Section 11.3.3; Reynolds, 1994)

8. (a) Suppose that a posterior density $\pi(\theta \mid y)$ is awkward but has a much simpler form $\pi(\theta \mid u, y)$ if the data y are augmented to (u, y). Show that

$$\pi(\theta \mid y) = \int \pi(\theta \mid u, y)\pi(u \mid y)\, du, \quad \pi(u \mid y) = \int \pi(u \mid \theta, y)\pi(\theta \mid y)\, d\theta,$$

and deduce that $\pi(\theta \mid y)$ satisfies an integral equation

$$\pi(\theta \mid y) = \int K(\theta, \theta'; y)\pi(\theta' \mid y)\, d\theta'.$$

Explain the relation between this and the *data augmentation* algorithm:

- choose an initial guess $\pi^{(0)}(\theta)$ for $\pi(\theta \mid y)$.
- Then for $i = 0, \ldots, I-1$ repeat:
  - generate $\theta_1, \ldots, \theta_S \overset{\text{iid}}{\sim} \pi^{(i)}(\theta)$;
  - generate $U_s \overset{\text{ind}}{\sim} \pi(u \mid \theta_s, y)$, for $s = 1, \ldots, S$; then
  - increment i, and set $\pi^{(i)}(\theta) = S^{-1} \sum_s \pi(\theta \mid u_s, y)$.

(b) Consider a multinomial variable $(U_0, \ldots, U_4)$ with denominator m and probabilities $(\pi_0, \ldots, \pi_4) = (1/2, \theta/4, (1-\theta)/4, (1-\theta)/4, \theta/4)$. Suppose that $y_1 = u_0 + u_1 = 125$, $y_2 = u_2 = 18$, $y_3 = u_3 = 20$ and $y_4 = u_4 = 34$ are observed, so u_0 and u_1 are unknown. Take prior $\pi(\theta) = 1$, $0 < \theta < 1$. Show that

$$\begin{aligned} \pi(\theta \mid u, y) &= \frac{\Gamma(u_1 + y_2 + y_3 + y_4 + 1)}{\Gamma(u_1 + y_4 + 1)\Gamma(y_2 + y_3 + 1)} \theta^{u_1 + y_4}(1-\theta)^{y_2 + y_3}, \quad 0 < \theta < 1, \\ \pi(u \mid \theta, y) &= \binom{y_1}{u_1} \frac{\theta^{u_1} 2^{y_1 - u_1}}{(\theta + 2)^{y_1}}, \quad u_1 = 0, \ldots, y_1. \end{aligned}$$

Check that the following code implements this:

```
y1 <- 125; y2 <- 18; y3 <- 20; y4 <- 34  # data
S <- 1000; I <- 10                       # I iterations, S simulations
u <- th <- matrix(NA, I, S)              # store u and theta
for (i in 1:I)                           # data augmentation
```

```
{  th[i,] <- if (i==1) runif(S)
              else rbeta(S, shape1=u[i-1,]+y4+1, shape2=y2+y3+1)
   u[i,] <- rbinom(S, size=y1, prob=th[i,]/(th[i,]+2))  }

# plot output values of theta for iterations 1, 2, and I
plot(density(th[1,]),type="l",xlab="theta",ylab="Posterior",ylim=c(0,10))
lines(density(th[2,]),col=2)
lines(density(th[I,]),col=3)

# exact calculation for comparison
m <- 500
x <- seq(from=0,to=1,length=m)
d <- (0.5+x/4)^y1*(1-x)^(y2+y3)*x^y4
d <- d/mean(d)
plot(x,d,type="l",xlab="theta",ylab="Posterior", ylim=c(0,10))
lines(density(th[I,]),col=2)

# better to use Rao-Blackwellized formula
d <- rep(0,m)
for (s in 1:S) d <- d + dbeta(x, shape1=u[I,s]+y4+1, shape2=y2+y3+1)/S
lines(x,d,col=3)
```

Experiment with different values of I and S. Discuss your findings. Can you see a link to the Gibbs sampler?

(Tanner, 1996, Chapter 5)

Chapter 12

Marginal and Conditional Likelihood

The practicals in this chapter are based on Alessandra Brazzale's work and in particular her libraries for conditional inference in R and in S-PLUS ; see

`http://www.isib.cnr.it/~brazzale/lib.html`

You will need to download and install these libraries first.

1. (a) Here is code for generating random samples from the t_ν distribution and then performing conditional inference for the location parameter, allowing for variation in the scale parameter:

```
nu <- 3; n <- 10
y <- rt(n, df=nu)
qqnorm(y)
library(cond)
library(marg)
y.rsm <- rsm(y~1, family=student(nu))
y.cond <- cond(y.rsm, offset=1)
y.cond
plot(y.cond)
```

   Asses how strongly the difference between the large-sample and conditional inferences depends on n and ν as they vary, by generating different samples. What seems to be the effect of the ancillary?

   (b) Here is code for the regression analysis of the data on tides in Yarmouth of Example 6.33, using Gumbel errors.

```
data(tide)
y <- tide$Yarmouth[!is.na(tide$Yarmouth)] # Yarmouth data, no missing values
year <- tide$year[!is.na(tide$Yarmouth)] # and corresponding years
yar.rsm <- rsm(y~year,family=extreme)  # Fit regression model with Gumbel errors
yar.cond <- cond(yar.rsm, offset=year) # Conditional inference for coeff. of year
yar.cond
plot(yar.cond)
```

Is the small difference between the large-sample and conditional analyses in this example coherent with what you observed in (a)?

(Section 12.2)

2. Verify that a random variable Y with Laplace density $\frac{1}{2}\exp(-|y-\theta|)$, $-\infty < y < \infty$, has cumulant generator[1]

$$K(t) = t\theta - \log(1-t^2).$$

Hence calculate the saddlepoint approximations to the density and distribution functions of an average of n independent Laplace variables. Implement them as functions in `R`, and compare them visually with simulations obtained by, for example,

```
n <- 1; R <- 20000
y <- rexp(R)-rexp(R)
y <- sort(c(y,-y))  # to symmetrize the sample and hence output
par(mfrow=c(1,2))
plot(density(y,width=0.2,n=200),type="l",ylab="PDF")
CDF <- seq(1,2*R,1)/(2*R+1)
plot(y,CDF,type="s")
```

(Section 12.3.2)

3. For a conditional analysis of the coefficients for the `urine` data of Practical 10.9, we set

```
data(urine)
fit <- glm(r~.,subset=-c(1,55),data=urine,binomial)
fit.cond <- cond(fit,offset=calc)  # conditional analysis for coefficient of calc
summary(fit.cond)
plot(fit.cond)
```

Try conducting conditional inference for coefficients of other covariates. Does this change your conclusions for any of them? Discuss.

(Section 12.3.3; Andrews and Herzberg, 1985, pp. 249–251)

[1] Note that Y may be expressed as $\theta + X_1 - X_2$, where the X_j are independent standard exponential variables.

Bibliography

Andrews, D. F. and Herzberg, A. M. (1985) *Data: A Collection of Problems from Many Fields for the Student and Research Worker*. New York: Springer.

Atkinson, A. C. (1985) *Plots, Transformations, and Regression*. Oxford: Clarendon Press.

Avery, P. J. and Henderson, D. A. (1999) Fitting Markov chain models to discrete state series such as DNA sequences. *Applied Statistics* **48**, 53–61.

Bishop, Y. M., Fienberg, S. E. and Holland, P. W. (1975) *Discrete Multivariate Analysis*. Cambridge, Massachussetts: MIT Press.

Bliss, C. I. (1935) The calculation of the dosage-mortality curve. *Annals of Applied Biology* **22**, 134–167.

Casella, G. and George, E. I. (1992) Explaining the Gibbs sampler. *The American Statistician* **46**, 167–174.

Cox, D. R. and Snell, E. J. (1981) *Applied Statistics: Principles and Examples*. London: Chapman & Hall.

Cox, D. R. and Snell, E. J. (1989) *Analysis of Binary Data*. Second edition. London: Chapman & Hall.

Crowder, M. J. (1978) Beta-binomial anova for proportions. *Applied Statistics* **27**, 34–37.

Dalal, S. R., Fowlkes, E. B. and Hoadley, B. (1989) Risk analysis of the space shuttle: Pre-Challenger prediction of failure. *Journal of the American Statistical Association* **84**, 945–957.

Dalgaard, P. (2002) *Introductory Statistics with R*. New York: Springer.

Davison, A. C. (2003) *Statistical Models*. Cambridge: Cambridge University Press.

Efron, B. (1996) Empirical Bayes methods for combining likelihoods (with Discussion). *Journal of the American Statistical Association* **91**, 538–565.

Fleming, T. R. and Harrington, D. P. (1991) *Counting Processes and Survival Analysis*. New York: Wiley.

Frets, G. P. (1921) Heredity of head form in man. *Genetica* **3**, 193–384.

Gaver, D. P. and O'Muircheartaigh, I. G. (1987) Robust empirical Bayes analysis of event rates. *Technometrics* **29**, 1–15.

Henderson, R. and Matthews, J. N. S. (1993) An investigation of changepoints in the annual number of cases of haemolytic uraemic syndrome. *Applied Statistics* **42**, 461–471.

Jarrett, R. G. (1979) A note on the intervals between coal-mining disasters. *Biometrika* **66**, 191–193.

Lindley, D. V. (1961) An experiment in the marking of an examination (with Discussion). *Journal of the Royal Statistical Society series A* **124**, 285–313.

Maguire, B. A., Pearson, E. S. and Wynn, A. H. A. (1952) The time intervals between industrial accidents. *Biometrika* **39**, 168–180.

MainDonald, J. and Braun, J. (2003) *Data Analysis and Graphics using R: An Example-Based Approach*. Cambridge: Cambridge University Press.

Mardia, K. V., Kent, J. T. and Bibby, J. M. (1979) *Multivariate Analysis*. London: Academic Press.

Marron, J. S. and Wand, M. P. (1992) Exact mean integrated squared error. *Annals of Statistics* **20**, 712–736.

McDonald, G. C. and Schwing, R. C. (1973) Instabilities of regression estimates relating air pollution to mortality. *Technometrics* **15**, 463–481.

Miller, R. G. (1981) *Survival Analysis*. New York: Wiley.

Moore, P. G. (1952) The estimation of the Poisson parameter from a truncated distribution. *Biometrika* **39**, 247–251.

Nelson, W. D. and Hahn, G. J. (1972) Linear estimation of a regression relationship from censored data. Part 1 — simple methods and their application (with Discussion). *Technometrics* **14**, 247–276.

Paulsen, O. and Heggelund, P. (1994) The quantal size at retinogeniculate synapses determined from spontaneous and evoked EPSCs in guinea-pig thalamic slices. *Journal of Physiology* **480**, 505–511.

Proschan, F. (1963) Theoretical explanation of observed decreasing failure rate. *Technometrics* **5**, 375–383.

Reynolds, P. S. (1994) Time-series analyses of beaver body temperatures. In *Case Studies in Biometry*, eds N. Lange, L. Ryan, L. Billard, D. R. Brillinger, L. Conquest and J. Greenhouse, pp. 211–228. New York: Wiley.

Ruppert, D. and Carroll, R. J. (1980) Trimmed least squares estimation in the linear model. *Journal of the American Statistical Association* **75**, 828–838.

Sacks, H. S., Chalmers, T. C., Blum, A. L., Berrier, J. and Pagano, P. (1990) Endoscopic hemostasis: An effective therapy for bleeding peptic ulcers. *Journal of the American Medical Association* **264**, 494–499.

Sheather, S. J. and Jones, M. C. (1991) A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society series B* **53**, 683–690.

Silverman, B. W. (1981) Using kernel density estimates to investigate multimodality. *Journal of the Royal Statistical Society series B* **43**, 97–99.

Simonoff, J. S. (1998) Logistic regression, categorical predictors, and goodness-of-fit: It depends on who you ask. *The American Statistician* **52**, 10–14.

Smith, R. L. (1985) Maximum likelihood estimation in a class of non-regular cases. *Biometrika* **72**, 67–92.

Steinijans, V. W. (1976) A stochastic point process model for the occurrence of major freezes of Lake Constance. *Applied Statistics* **25**, 58–61.

Student (1908) The probable error of a mean. *Biometrika* **6**, 1–25. Reprinted in *Breakthroughs in Statistics*, volume II, eds S. Kotz and N. L. Johnson, pp. 33–57. New York: Springer.

Svensson, A. (1981) On a goodness-of-fit test for multiplicative Poisson models. *Annals of Statistics* **9**, 697–704.

Tanner, M. A. (1996) *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*. Third edition. New York: Springer.

Taylor, G. L. and Prior, A. M. (1938) Blood groups in England. *Annals of Eugenics* **8**, 343–355.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S*. Fourth edition. New York: Springer.

Whittaker, J. (1990) *Graphical Models in Applied Multivariate Statistics*. New York: Wiley.