9: Graphics

John H Maindonald

March 7, 2018

```
doFigs <- TRUE
```

```
fig9.1 <- function(){
## ---- brain-body ----
plot(Brainwt ~ Bodywt, xlim=c(0, 300),
    ylim=c(0,1500), data=primates,
    xlab="Body weight (kg)",
    ylab="Brain weight (g)")
# Specify xlim to allow room for the labels
with(primates,
    text(Brainwt ~ Bodywt,
        labels=rownames(primates), pos=4))
# pos: pos=1 (below), 2 (left), 3 (above)
}
```

```
supp9.1 <- function(){
if(!require(RColorBrewer))stop("RColorBrewer must be installed")
## ---- col-plots ----
opt <- options(width=75)
colpal <- rev(list(
    "Default palette" = palette()[1:8],  cm.colors = cm.colors(12),
    terrain.colors = terrain.colors(12), heat.colors = heat.colors(12),
    blueRamp = colorRampPalette(c(blues9, "white"))(12),
    "Brewer-Set1" = brewer.pal(8, "Set1"),
    "Brewer-Dark2" = brewer.pal(8, "Dark2")))
palnam <- names(colpal)
plot(1, 1, xlim=c(0.5,12.5), ylim=c(0,length(palnam)+0.5), type="n",
    axes=FALSE, xlab="", ylab="")
for(i in 1:length(palnam)){
    len <- length(colpal[[i]])
    points(1:len, rep(i,len), pch=15, col=colpal[[i]], cex=5.5)
    legend(1, i+0.025, palnam[i], adj=0, box.col="white", bg="white",
        x.intersp=0, y.intersp=0, yjust=0)
```

```r
}
options(opt)
}
```

```r
fig9.2 <- function(){
## ---- alpha-ex ----
## Sample from the 15992 rows
dfsamp <- cps1[sample(nrow(cps1), 3000), ]
plot(re78 ~ re75, data=dfsamp, pch=20, cex=0.5,
     col="black")
mtext(side=3, line=0.5, "A: 100% opacity", adj=0)
plot(re78 ~ re75, data=dfsamp, pch=20, cex=0.5,
     col=adjustcolor("black", alpha=0.4))
mtext(side=3, line=0.5, "B: 40% opacity", adj=0)
blueRamp <- colorRampPalette(c("white", blues9))
with(dfsamp, smoothScatter(re75~re74,
                           colramp=blueRamp))
mtext(side=3, line=0.5, "C: Color density plot",
      adj=0)
}
```

```r
fig9.3 <- function(){
## ---- aspect ----
plot((1:30)*0.92, sin((1:30)*0.92),
     xlab="", ylab="")
}
```

```r
supp9.2 <- function(){
## ---- Animals-body-brain ----
## Supplementary figure 9.2
library(MASS)
oldpar <- par(pch=16, pty="s", mfrow=c(2,2))
with(Animals, {      # bracket several R statements
  plot(body, brain)
  plot(sqrt(body), sqrt(brain))
  plot(body^0.1, brain^0.1)
  plot(log(body), log(brain))
})                   # close both sets of brackets
par(oldpar)          # Restore former settings
}
```

```r
fig9.4 <- function(){
## ---- poss-hist ----
ftotlen <- subset(possum, sex=="f")[, "totlngth"]
## Left panel: breaks at 72.5, 77.5,..
hist(ftotlen, breaks = 72.5 + (0:5)*5, freq=FALSE,
     xlab="Total length", ylim=c(0,0.11),
     main ="A: Breaks at 72.5, 77.5,...")
## Now superimpose a density curve, as in Fig. 7.3
lines(density(ftotlen))
##
## Panel B: breaks at 75, 80, ...
hist(ftotlen, breaks = 75 + (0:5)*5, freq=FALSE,
     xlab="Total length", ylim=c(0,0.11),
     main="B: Breaks at 75, 80, ...")
}
```

```r
supp9.3 <- function(){
## ---- fposs-density ----
## Supplementary figure 9.3
with(subset(possum, sex=="f"),
     plot(density(totlngth), type="l"))
}
```

```r
fig9.5 <- function(){
## ---- boxplot ----
## Code
with(subset(possum, sex=="f"),
     {boxplot(totlngth, horizontal=TRUE)
      rug(totlngth)} )
}
```

```r
fig9.6 <- function(){
## ---- possum-qqn ----
ftotlen <- subset(possum, sex == "f")[, "totlngth"]
ylabel <- expression(bold("Data"))
xy <- qqnorm(ftotlen, xlab="", ylab=ylabel)
usr <- par()$usr
fillcol <- adjustcolor("gray60", alpha=0.2)
rect(usr[1], usr[3],
     usr[2], usr[4], col=fillcol)
for(i in 1:7) qqnorm(rnorm(43), xlab="",
                     ylab="Simulated",
                     main="Q-Q: Simulated")
```

```r
# rnorm() generates random samples from a normal
# distribution with, by default, mean 0 and SD=1.
}


fig9.7 <- function(){
## ---- plot-expr ----
yl <- expression("Area = " * pi * r^~2)
plot(1:5, pi*(1:5)^2, xlab="Radius (r)", ylab=yl)
}


fig9.8 <- function(){
## ---- cline-gph ----
## On the command line: Create and print object
xyplot(Brainwt ~ Bodywt, data=primates)
}


fig9.9 <- function(){
## ---- rowSwim ----
xyplot(ht ~ wt | sport, groups=sex, data=ais,
       par.settings=simpleTheme(pch=c(4,1)),
       scales=list(tck=0.5),
       auto.key=list(space="right"),
       subset=sport%in%c("Row","Swim"))
}


fig9.10 <- function(){
## ---- grog-simple ----
## Simple version of plot
grogplot <- xyplot(
             Beer+Spirit+Wine ~ Year | Country,
             data=grog, outer=FALSE,
             auto.key=list(space="right"))
## ---- grog-update ----
## Update trellis object, then print
ylab <- "Amount consumed (per person)"
parset <- simpleTheme(pch=c(1,3,4))
finalplot <- update(grogplot, ylim=c(0,5.5),
                    xlab="", ylab=ylab,
                    par.settings=parset)
print(finalplot)
}
```

```r
fig9.11 <- function(){
## ---- strip-grob ----
library(grid)
plotnam <- "Stripplot of cuckoo data"
stripplot(species ~ length, xlab="", data=cuckoos,
  legend=list(top=list(fun=textGrob,
                       args=list(label=plotnam,
                                 x=0)))))
}


fig9.12 <- function(){
## ---- jobs-basic ----
## 1. Create a basic version of the graphics object
jobsB.xyplot <-
  xyplot(Ontario+Quebec+BC+Alberta+Prairies+Atlantic ~ Date,
         data=jobs, type="b", layout=c(3,2), outer=TRUE,
         ylab="Number of jobs",
         scales=list(y=list(relation="sliced", log=TRUE)))
## ---- jobs-enhance ----
## 2. Code for the enhancements to jobsB.xyplot
ylabpos <- exp(pretty(log(unlist(jobs[,-7])), 100))
ylabels <- paste0(round(ylabpos),"\n(", log(ylabpos), ")")
## Create a date object 'startofmonth'; use instead of 'Date'
atdates <- seq(from=95, by=0.5, length=5)
datelabs <- format(seq(from=as.Date("1Jan1995", format="%d%b%Y"),
                       by="6 month", length=5), "%b%y")
update(jobsB.xyplot, xlab="", between=list(x=0.5, y=0.5),
       scales=list(x=list(at=atdates, labels=datelabs),
                   y=list(at=ylabpos, labels=ylabels), tck=0.6) )
}


fig9.13 <- function(){
## ---- strip-bw ----
stripplot(species ~ length, data=cuckoos,
          xlab="Cuckoo egg length (mm)")
bwplot(species ~ length, data=cuckoos,
       xlab="Cuckoo egg length (mm)")
}


fig9.14 <- function(){
## ---- lattice-density ----
## Code
colset <- c("gray","black")
```

```r
densityplot(~ earconch | sex, groups=Pop,
            data=possum,
            par.settings=simpleTheme(col=colset),
            auto.key=list(space="right"))
}


fig9.15 <- function(){
## ---- gph-from ----
gph <- xyplot(Brainwt ~ Bodywt,  data=primates,
              xlim=c(0,300))
## ---- my-panel ----
my.panel <- function(x,y){
  panel.xyplot(x,y)
  panel.text(x,y, labels=rownames(primates), pos=4)
}
update(gph, panel=my.panel)
}


fig9.16 <- function(){
print(c("The Figure 9.16 screen display was obtained using the playwith GUI.",
        "See text for details."), quote=FALSE)
}


fig9.17 <- function(){
## ---- qplot-smooth ----
## Default loess smooth, with SE bands added.
ggplot2::quickplot(Year, mdbRain, data=DAAG::bomregions2015,
          geom=c("point","smooth"), se=TRUE,
          method="loess", span=0.5, xlab="",
          ylab="Av. rainfall, M-D basin")
}


fig9.18 <- function(){
## ---- overlay-dens-simple ----
## Overlay with boxplots and density contours
ggplot2::quickplot(wt, ht, data=DAAG::ais,
          geom=c("boxplot", "point", "density2d"),
          facets = . ~ sex)
}
```

```r
supp9.4 <- function(){
## ---- twenty5080 ----
## Supplementary figure 4
ggplot2::quickplot(Year, mdbRain, data=DAAG::bomregions2015,
          geom=c("point", "quantile"),
          formula = y ~ ns(x,5),
          quantiles=c(0.2,0.5,0.8) )
}
```

```r
fig9.19 <- function(){
## ---- aisRS ----
## Extract from ais data for rowers and swimmers
aisRS <- subset(DAAG::ais, sport %in% c("Row","Swim"))
aisRS$sport <- droplevels(aisRS$sport)
ggplot2::ggplot(aisRS) +
  ggplot2::geom_point(aes(wt, ht,
            color=sex,
            shape=sport),
            size=2)
}
```

```r
figset9 <- function(){
  if(!require(DAAG, quietly=TRUE))stop('DAAG must be installed')
  if(!require(MASS, quietly=TRUE))stop('MASS must be installed')
  if(!require(latticeExtra, quietly=TRUE))stop('latticeExtra must be installed')
  if(!require(ggplot2, quietly=TRUE))stop('ggplot2 must be installed')
  if(!require(splines, quietly=TRUE))stop('splines must be installed')
  if(!require(quantreg, quietly=TRUE))stop('quantreg must be installed')
  if(!require(RColorBrewer, quietly=TRUE))stop('RColorBrewer must be installed')
  }
```

```r
figset9()
```

```
Attaching package:  'MASS'
The following object is masked from 'package:DAAG':
      hills

Attaching package:  'ggplot2'
The following object is masked from 'package:latticeExtra':
      layer

Attaching package:  'SparseM'
The following object is masked from 'package:base':
      backsolve
```

```
fig9.1()
```
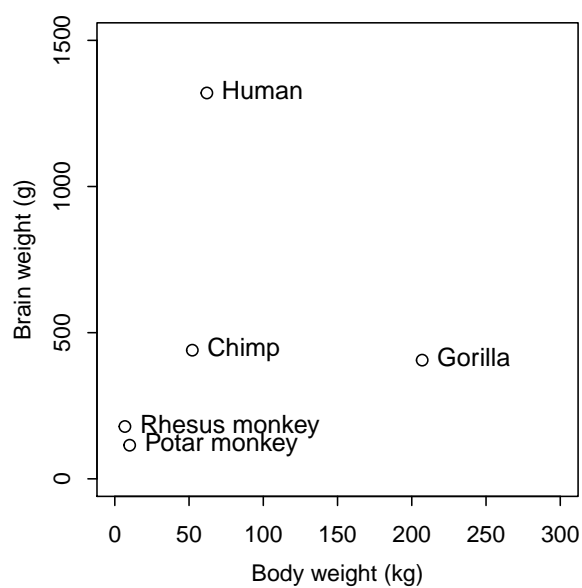

Figure 1: Plot of brain weight against body weight, for selected primates.

```
supp9.1()
```
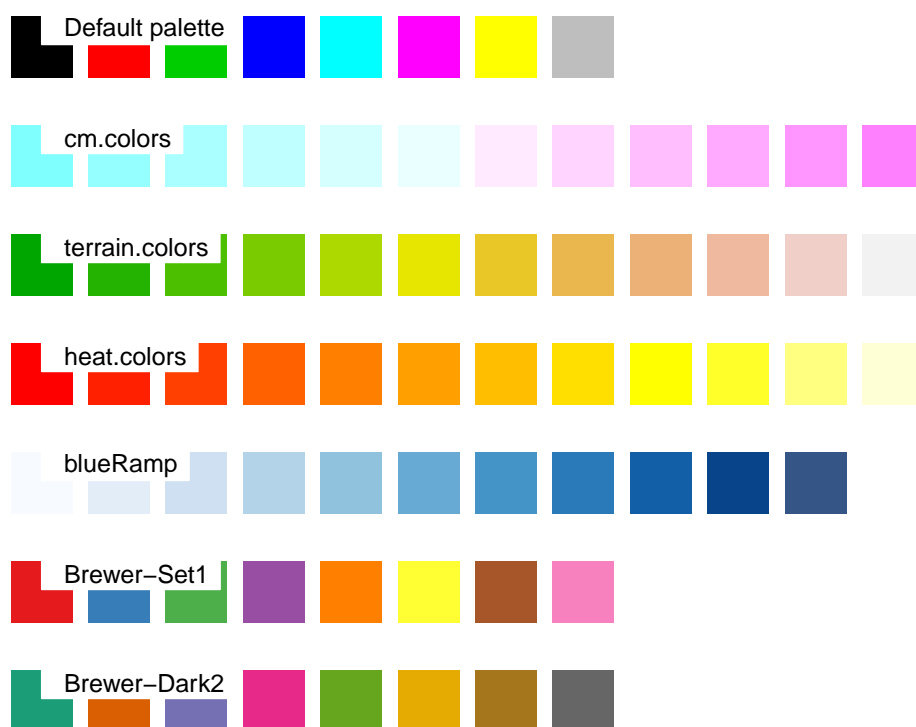

Supplementary Figure 1: Color palettes.

```
fig9.2()
```
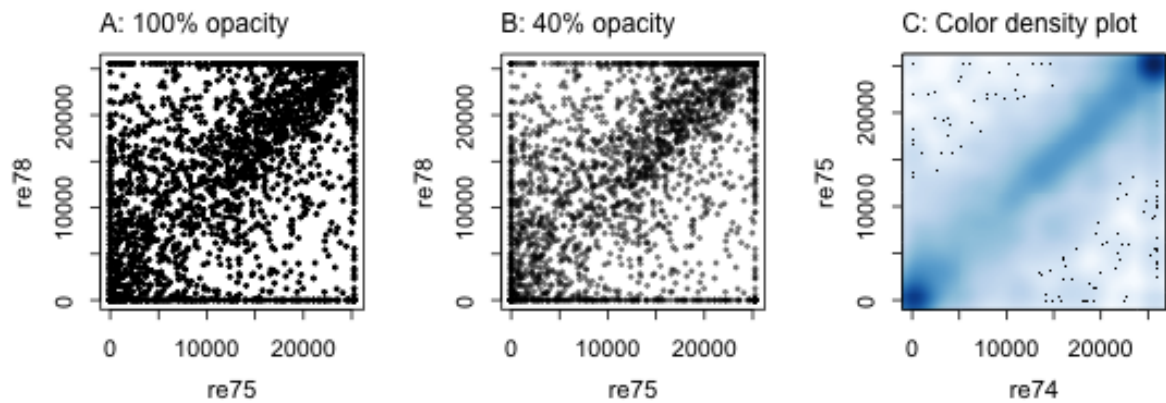

Figure 2: In Panel A, points are plotted with the 100% opacity, i.e., no transparency. In Panel B, `alpha=0.4`, i.e., 40% opacity. Panel C uses the function `smoothScatter()` to show a smoothed color density representation of the data. .

```
par(fig=c(0,0.265, 0,1))
fig9.3()
mtext(side=3, line=0.5, "A: 1:1 aspect ratio", adj=0)
par(fig=c(0.265, 1, 0,1), new=TRUE)
fig9.3()
mtext(side=3, line=0.5, "B: 1:3.5 aspect ratio", adj=0)
```
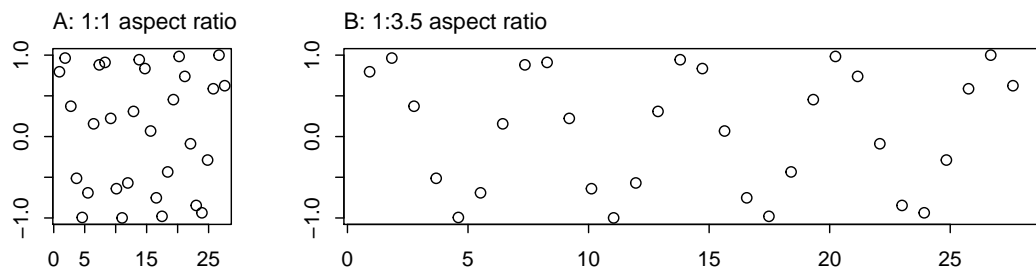

Figure 3: Both figures show the same data, but with very different aspect ratios.
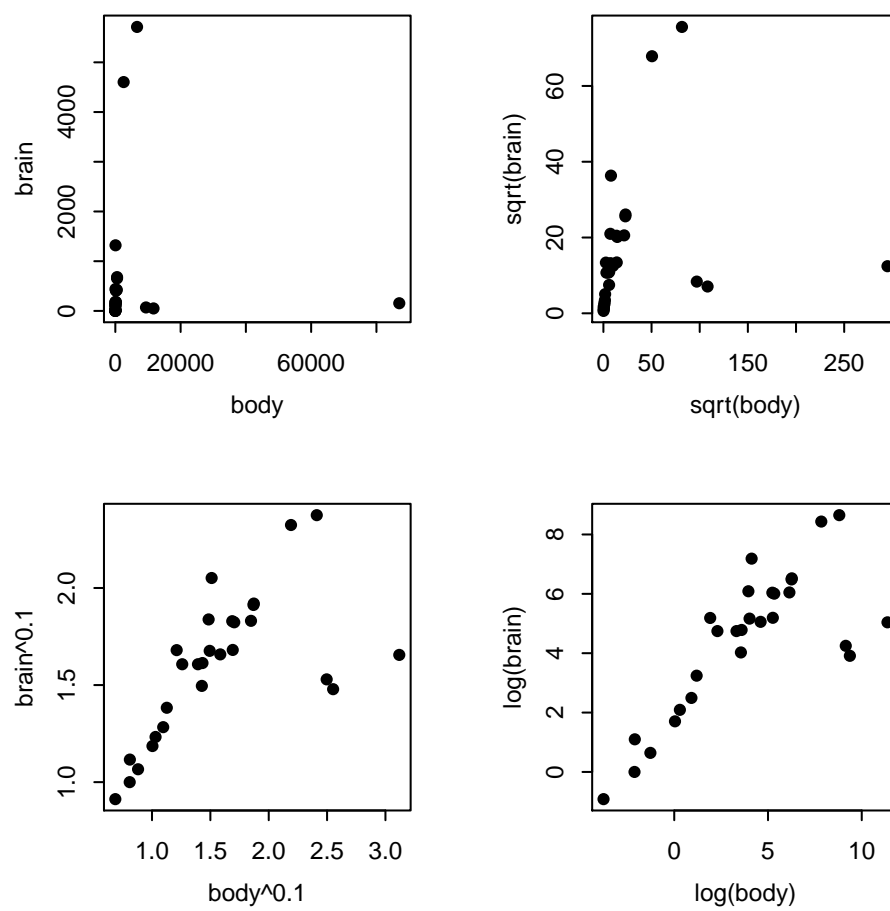
Supplementary Figure 2: The figure shows a 2 by 2 layout of plots.

`fig9.4()`

A: Breaks at 72.5, 77.5,...   **B: Breaks at 75, 80, ...**
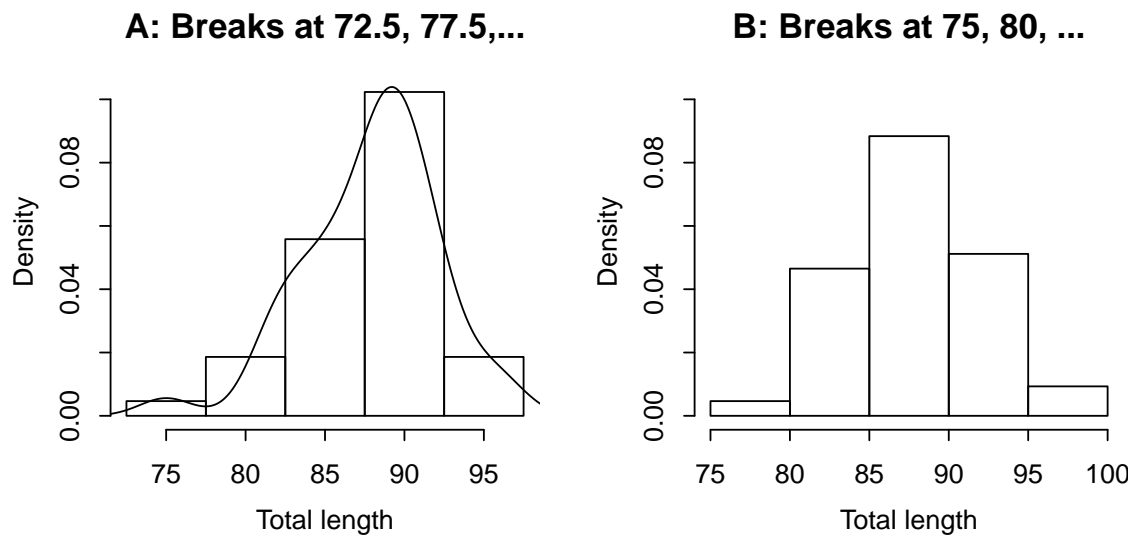
Figure 4: The two panels show the same data, but with a different choice of breakpoints.

`supp9.3()`

density.default(x = totlngth)

N = 43   Bandwidth = 1.662
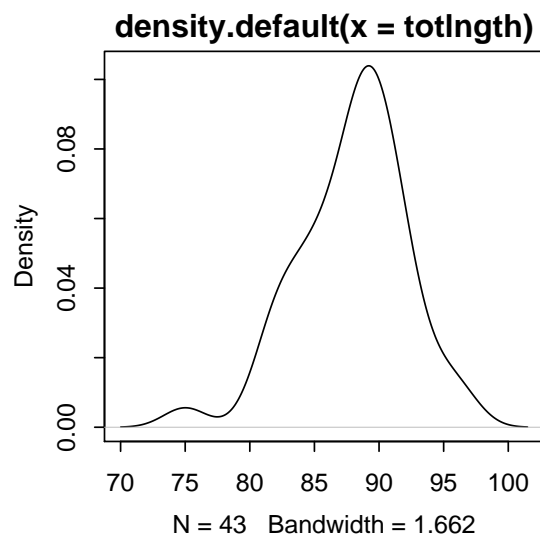
Supplementary Figure 3: Density plot for lengths of female possums.

```
fig9.5()
```
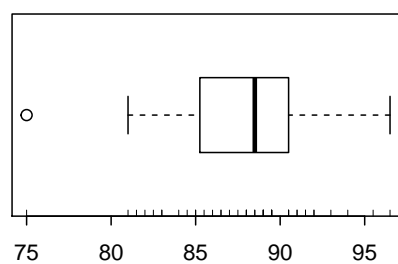

Figure 5: Distribution of lengths of female possums. The bars (together making up a 'rug') show actual data values.

```
fig9.6()
```
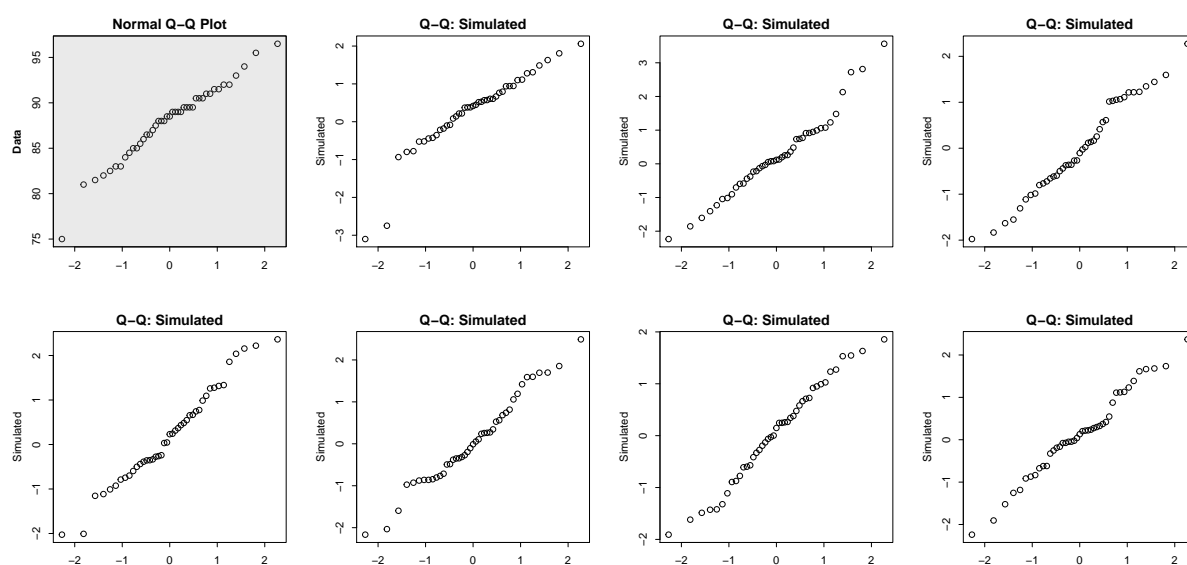

Figure 6: Normal probability plots. The top left panel shows the 43 lengths of female possums. Other panels are for independent normal random samples of size 43.

```
fig9.7()
```
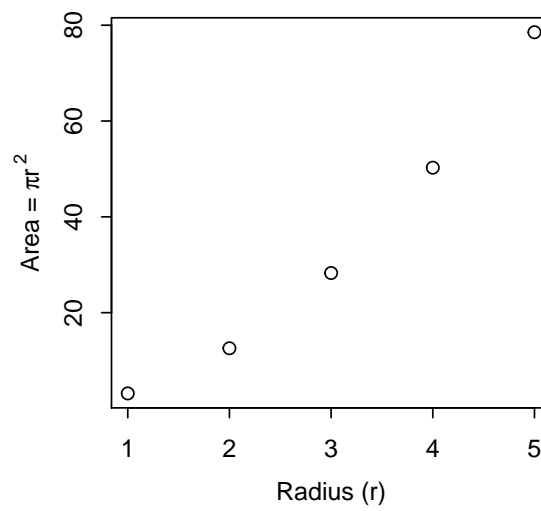

Figure 7: A mathematical expression is included as part of the y-axis label..

```
fig9.8()
```
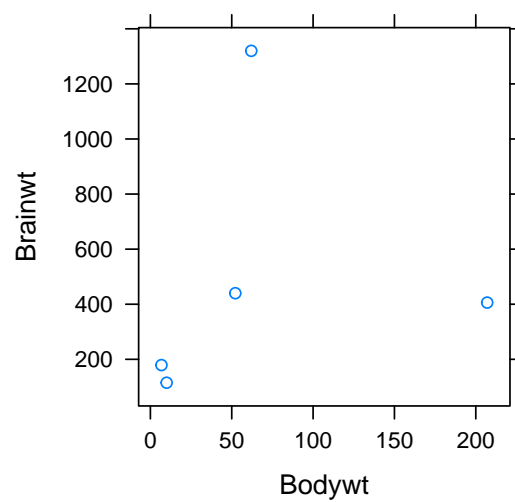

Figure 8: Use of lattice function `xyplot()` to give a graph.

```
fig9.9()
```
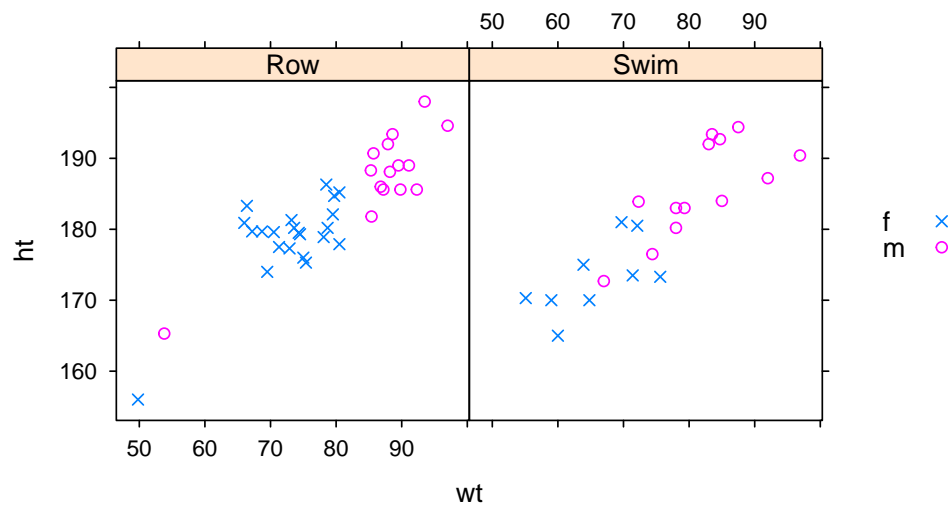

Figure 9: Height (`ht`) versus Weight (`wt`), for rowers (`Row`) and swimmers (`Swim`). Different plotting symbols are used to distinguish males from females.

```
fig9.10()
```
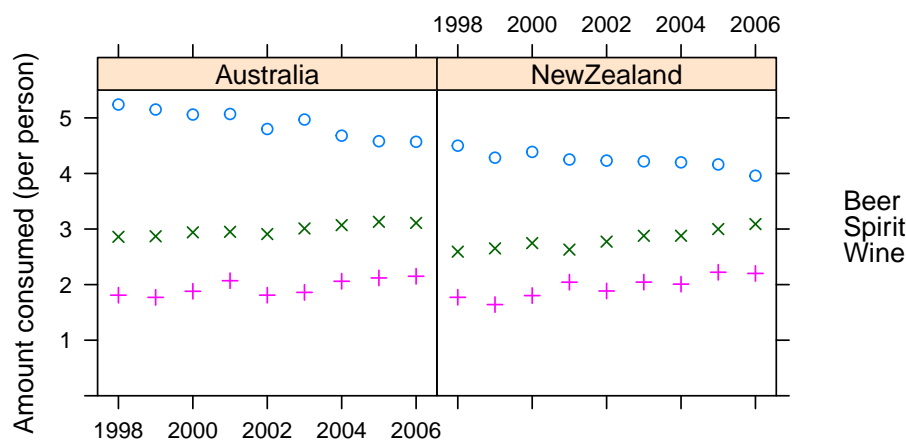

Figure 10: Australian and New Zealand apparent per person annual consumption (in liters) of the pure alcohol content of liquor products, for 1998 to 2006.

`fig9.11()`

Stripplot of cuckoo data
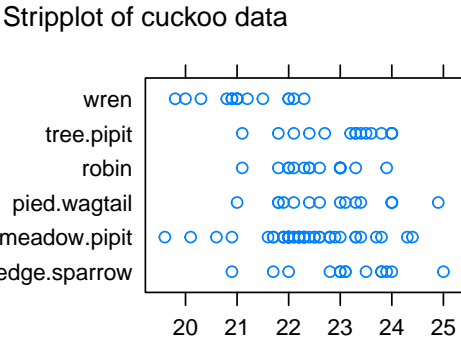
Figure 11: The argument `legend` has been used to add text, supplied as a 'grob'.

`fig9.12()`
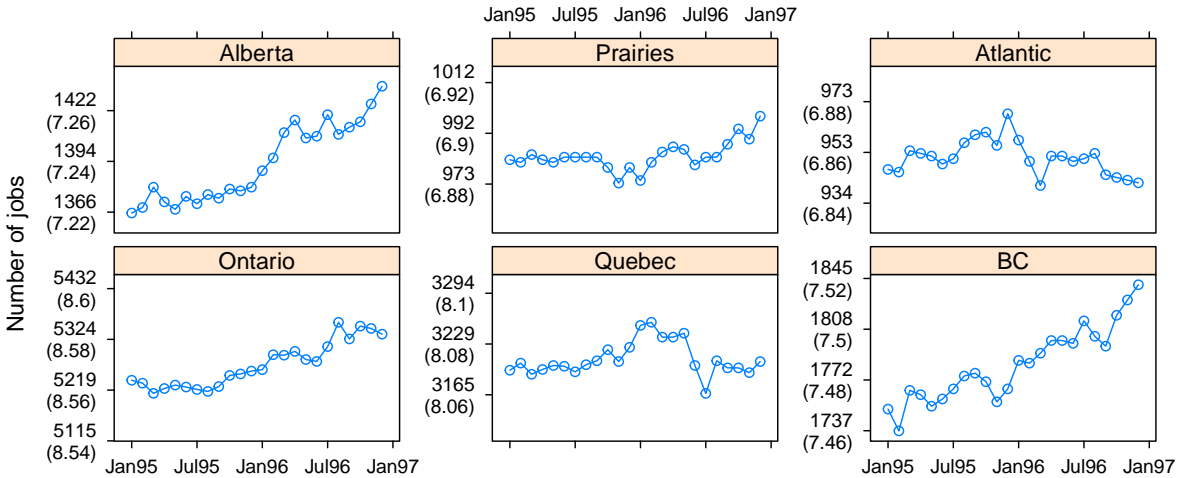
Figure 12: Jobs growth in Canadian provinces, between January 1995 and December 1996.

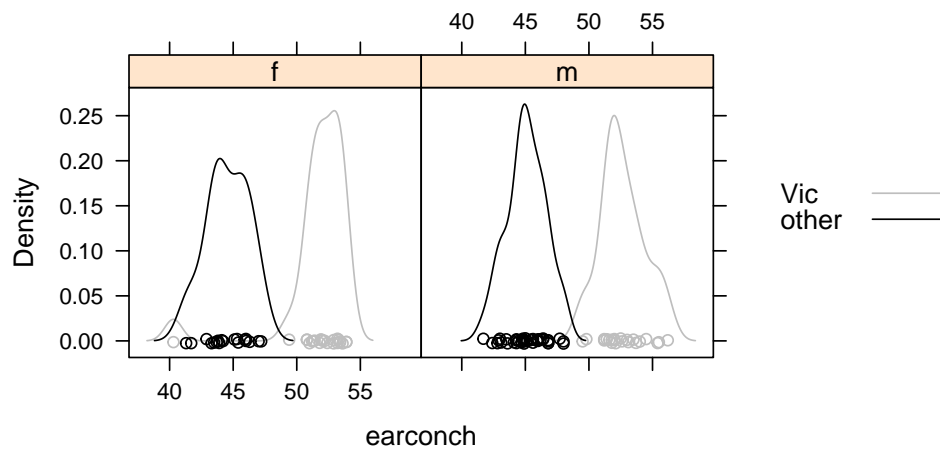Figure 13: A stripplot and a dotplot, side by side for comparison.

```
fig9.14()
```


Figure 14: Lattice style density plot comparing possum earconch measurements, separately for males and females, between Victorian and other populations. Observe that the scatter of data values is shown along the horizontal axis.
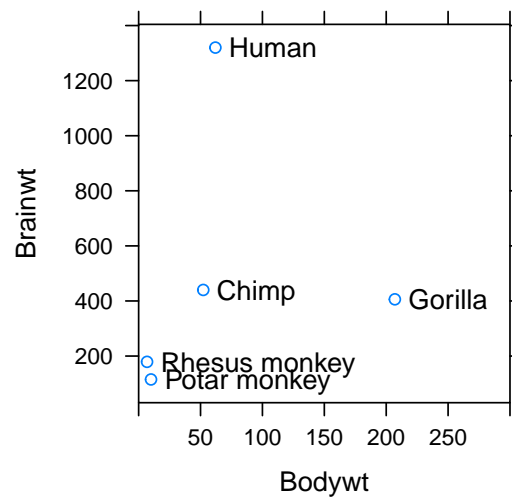
```
fig9.15()
```


Figure 15: Here, labels have been added to the points. This can be done by updating a graph that has the points, by use of a panel function that both plots points and and adds labels, or by adding a new layer to a graph that shows the points.
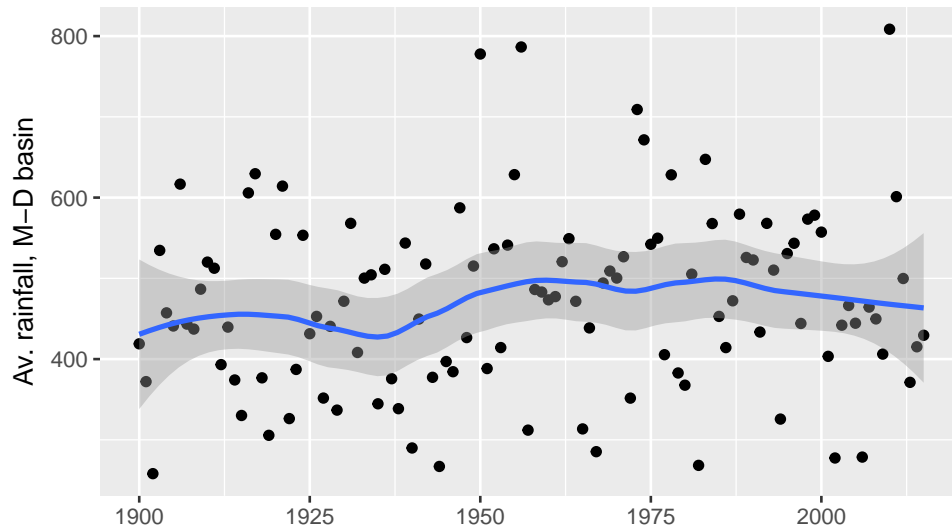
```
fig9.17()
```


Figure 17: Annual rainfall, from 1901 to 2012, for the Murray-Darling basin region of Australia. The curve is fitted using the default loess smoother. The pointwise standard error bands assume that errors about the curve are independent; this is unlikely to be strictly true. To suppress these bands, specify `se=FALSE`.
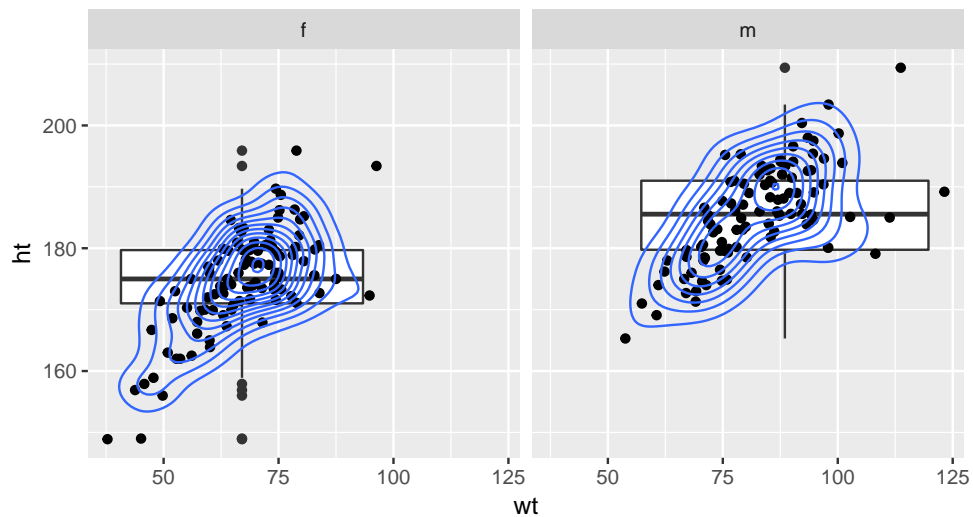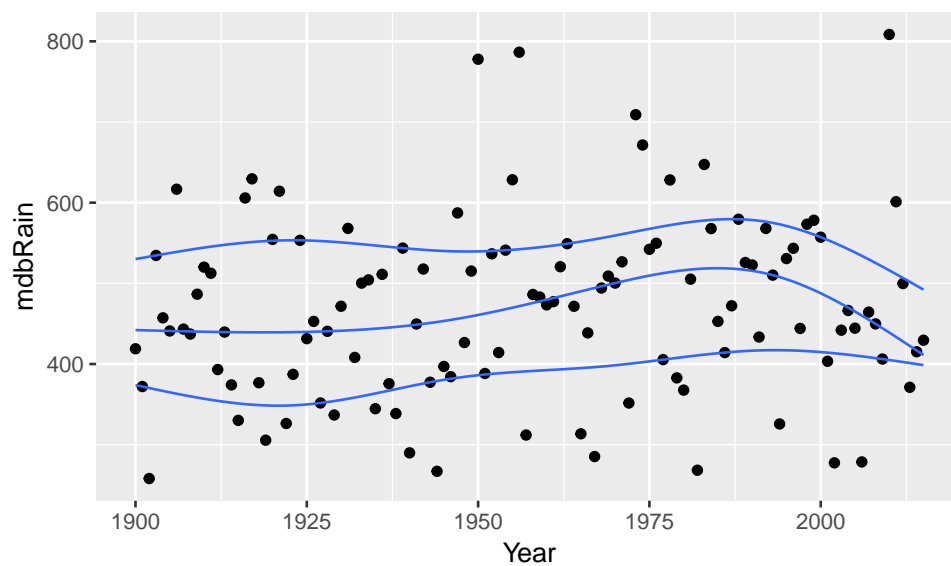
```
fig9.18()
```


Figure 18: Height versus weight, by sex, for Australian athletes in the `ais` data set. Boxplots that show the distributions of heights, and two-dimensional density contours have been added.

```
supp9.4()
```


Supplementary Figure 4: Annual rainfall, from 1901 to 2012, for the Murray-Darling basin region of Australia. Curves that estimate the 20%, 50% and 80% quantiles have been fitted.
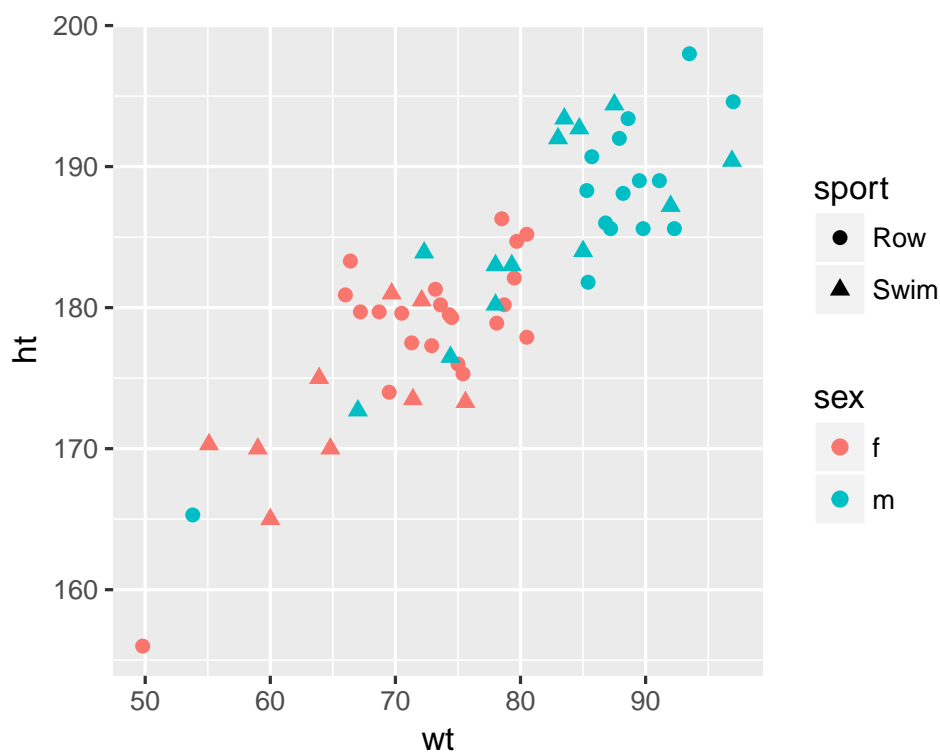
```
fig9.19()
```


Figure 19: Use `color` for distinguishing `sex`es, `shapes` for `sport`s.