```
########################################
#### SUGGESTED EXERCISE SOLUTIONS ####
########################################

#########
## 7.1 ##
#########
#(a) As closely as you can, re-create the following plot:
plot(-3:3,7:13,type="n",xlab="",ylab="")
text(x=0,y=10,labels="SOMETHING\nPROFOUND")
abline(v=c(-3,3),lty=2,lwd=4,col=8)
abline(h=c(7,13),lty=2,lwd=4,col=8)
arrows(x0=c(-2.5,-2.5,-
2.5,2.5,2.5,2.5),y0=c(7.5,10,12.5,7.5,10,12.5),x1=c(-1,-1,-
1,1,1,1),y1=c(9.5,10,10.5,9.5,10,10.5))

#(b)With the following data, create a plot of weight on the x -axis and
height on the y -axis. Use different point characters or colors to
distinguish between males and females and provide a matching legend.
Label the axes and give the plot a title.
#----------------------------
# Weight (kg) Height (cm) Sex
#----------------------------
#     55   161   female
#     85   185   male
#     75   174   male
#     42   154   female
#     93   188   male
#     63   178   male
#     58   170   female
#     75   167   male
#     89   181   male
#     67   178   female
w <- c(55,85,75,42,93,63,58,75,89,67)
h <- c(161,185,174,154,188,178,170,167,181,178)
s <-
c("female","male","male","female","male","male","female","male","male","f
emale")
plot(w,h,type="n",xlab="Weight (kg)",ylab="Height (cm)",main="Height
against weight for 10 people")
points(w[s=="male"],h[s=="male"],pch=19)
points(w[s=="female"],h[s=="female"],pch=3,col=2)
legend("topleft",legend=c("male","female"),pch=c(19,3),col=c(1,2))

#########
## 7.2 ##
#########
# In Exercise 7.1 (b), you used base R graphics to plot some weight and
height data, distinguishing males and females using different points or
colors. Repeat this task using ggplot2.
w <- c(55,85,75,42,93,63,58,75,89,67)
h <- c(161,185,174,154,188,178,170,167,181,178)
s <-
factor(c("female","male","male","female","male","male","female","male","m
ale","female"))
library(ggplot2)
qplot(w,h,color=s,shape=s,xlab="Weight (kg)",ylab="Height
(cm)",main="Height against weight for 10 people") + geom_point(size=4)
```

```r
##########
## 8.1 ##
##########
#(a) In R's built-in datasets library is the data frame quakes. Make sure
you can access this object and view the corresponding help file to get an
idea of what this data represents.
##(i) Select only those records that correspond to a magnitude (mag) of
greater than or equal to 5 and write them to a table-format file called
"q5.txt" in an existing folder on your machine. Use a delimiting
character of ! and do not include any row names.
write.table(x=quakes[quakes$mag>=5,],file="q5.txt",sep="!",row.names=F)
getwd()         # where is the file
file.choose()  # let's check
##(ii) Read the file back into your R workspace, naming the object
q5.dframe.
q5.dframe <- read.table(file="q5.txt",sep="!",header=T)

#(b) In the contributed package "car", there's a data frame called
Duncan, which provides historical data on perceived job prestige in 1950.
Install the "car" package and access the Duncan data set and its help
file.
install.packages("car")
library("car")
data(Duncan)
##(i) Write R code that will plot education on the x-axis and income on
the y-axis, with both x- and y-axis limits fixed to be [0, 100]. Provide
appropriate axis labels. For jobs with a prestige value of less than or
equal to 80, use a black open circle as the point character. For jobs
with prestige greater than 80, use a blue filled circle.
plot(Duncan$education[Duncan$prestige<=80],Duncan$income[Duncan$prestige<
=80],xlim=c(0,100),ylim=c(0,100),xlab="Education",ylab="Income")
points(Duncan$education[Duncan$prestige>80],Duncan$income[Duncan$prestige
>80],pch=19,col="blue")

##(ii) Add a legend explaining the difference between the two types of
points and then save a 500 x 500 pixel .png file of the image.
png("dunc.png",width=500,height=500)
plot(Duncan$education[Duncan$prestige<=80],Duncan$income[Duncan$prestige<
=80],xlim=c(0,100),ylim=c(0,100),xlab="Education",ylab="Income")
points(Duncan$education[Duncan$prestige>80],Duncan$income[Duncan$prestige
>80],pch=19,col="blue")
legend("topleft",legend=c("prestige > 80","prestige <=
80"),pch=c(19,1),col=c("blue","black"))
dev.off()
file.choose()  # check if file is created

#(c) Create a list called exer that contains the three data sets quakes,
q5.dframe, and Duncan.
exer <- list(quakes,q5.dframe,Duncan)

##(i) Write the list object directly to disk, calling it Exercise8-1.txt.
Briefly inspect the contents of the file in a text editor.
dput(x=exer,file="Exercise8-1Data.txt")

##(ii) Read Exercise8-1.txt back into your workspace; call the resulting
object list.of.dataframes. Check that list.of.dataframes does indeed
contain the three data frame objects.
list.of.dataframes <- dget("Exercise8-1Data.txt")
list.of.dataframes
```

```
#(d) In Section 7.4.3, you created a ggplot2 graphic of 20 observations
displayed as the bottom image ofFigure 7-11 on page 144. Use ggsave to
save a copy ofthis plot as a .tiff file.
x <- 1:20
y <- c(-1.49,3.37,2.59,-2.78,-3.94,-0.92,6.43,8.51,3.41,-8.23,-12.01,-
6.58,2.87,14.12,9.63,-4.58,-14.78,-11.67,1.17,15.62)
ptype <- rep(NA,length(x=x))
ptype[y>=5] <- "too_big"
ptype[y<=-5] <- "too_small"
ptype[(x>=5&x<=15)&(y>-5&y<5)] <- "sweet"
ptype[(x<5|x>15)&(y>-5&y<5)] <- "standard"
ptype <- factor(x=ptype)
qplot(x,y,color=ptype,shape=ptype) + geom_point(size=4) +
geom_line(mapping=aes(group=1),color="black",lty=2) +
geom_hline(mapping=aes(yintercept=c(-5,5)),color="red") +
geom_segment(mapping=aes(x=5,y=-5,xend=5,yend=5),color="red",lty=3) +
geom_segment(mapping=aes(x=15,y=-5,xend=15,yend=5),color="red",lty=3)
ggsave(filename="elaborateqplot.tiff")
file.choose()  # check if file is created
```