LIGHT CSW 2023 Quantitative Analysis Workshop

Exercises (solutions)

Alex Richards, Marc Henrion

1 March 2023

Exercise 1

Go to the course website on GitHub:

https://github.com/mlw-stats/LIGHT_CSW2

From here, download the following files:

`btTBreg.csv`
`btTBregHospitals.csv`

1. Load the `btTBreg.csv` data table into R.
2. The variables `cd41`, `cd42` and `cd41.sk`, `cd42.sk` measure the same variables (`cd4` and `cd4.sk` respectively) in the same individuals at two different time point. This means the data are in wide format. Reformat to long format.
3. Save the reformatted data into a file called `btTBregLong.tab` in such a way that
   i. Columns are tab-separated.
   ii. Column names are saved.
   iii. No row number is saved in the resulting file.
4. Load the `btTBregHospitals.csv` data table. Join the data frames storing `btTBreg.csv` and `btTBregHospitals.csv`.
5. Compute the average patient age and the proportion of male patients for each hospital.
6. Write an R function that computes the following summary statistics, then, using your custom function, compute these for the `bmi`, `cd41`, `cd42` columns:
   i. mean
   ii. median
   iii. inter quartile range
   iv. minimum
   v. maximum
   vi. number of missing values
7. Do the same now, but only for female patients. Repeat for only male patients.

Exercise 1 (solution)

Go to the course website on GitHub:

https://github.com/mlw-stats/LIGHT_CSW2

From here, download the following files:

btTBreg.csv
btTBregHospitals.csv

1. Load the btTBreg.csv data table into R.

```
btDat<-read.csv("dataAndSupportDocs/btTBreg.csv")

head(btDat) # have a look at the data
##   id age sex hiv   bmi ses cd41 cd42   cd41.sk  cd42.sk hosp
## 1  1  44   2   0 26.32   4  346  519 313.11656 572.8906    1
## 2  2  32   2   0 20.79   5  237  337  43.12752 406.1971    5
## 3  3  32   1   0 19.21   1  198  328 338.32172 408.2427    2
## 4  4  20   1   0 21.34   4  246  525  77.08697 312.7572    3
## 5  5  30   1   0 23.98   4  270  444 169.02539 335.3739    3
## 6  6  32   1   0 17.97   4  283  372 255.45773 323.4773    4
dim(btDat) # check dimensions of data table
## [1] 3000   11
```

The variables cd41, cd42 and cd41.sk, cd42.sk measure the same variables (cd4 and cd4.sk respectively) in the same individuals at two different time point. This means the data are in wide format. Reformat to long format.

```
btDatLong.cd4<-btDat %>%
  pivot_longer(names_to="time", values_to="cd4", cols=c(cd41, cd42)) %>%
  select(id,age,sex,hiv,bmi,ses,time,cd4)

btDatLong.cd4sk<-btDat%>%
  pivot_longer(names_to="time", values_to="cd4.sk", cols=c(cd41.sk, cd42.sk)) %>%
  select(id,age,sex,hiv,bmi,ses,time,cd4.sk)

btDatLong<-data.frame(btDatLong.cd4,cd4.sk=btDatLong.cd4sk$cd4.sk)

rm(btDatLong.cd4,btDatLong.cd4sk)

btDatLong$time<-factor(case_when(btDatLong$time=="cd41"~"entry",btDatLong$time=="cd42"~"exit",TRUE~NA_c

head(btDatLong) # have a look at the data
##   id age sex hiv   bmi ses  time cd4    cd4.sk
## 1  1  44   2   0 26.32   4 entry 346 313.11656
## 2  1  44   2   0 26.32   4  exit 519 572.89062
## 3  2  32   2   0 20.79   5 entry 237  43.12752
## 4  2  32   2   0 20.79   5  exit 337 406.19707
## 5  3  32   1   0 19.21   1 entry 198 338.32172
## 6  3  32   1   0 19.21   1  exit 328 408.24267
dim(btDatLong) # check dimensions
## [1] 6000    9
```

An alternative function that can be used is reshape(). To get more information on this function, type ?reshape at the console.

```
btDatLong<-reshape(btDat,
                   direction="long",
                   varying=list(c("cd41","cd42"),c("cd41.sk","cd42.sk")),
                   ids="id",
                   v.names=c("cd4","cd4.sk"))
```

```
head(btDatLong) # have a look at the data
##     id age sex hiv   bmi ses hosp time cd4    cd4.sk
## 1.1  1  44   2   0 26.32   4    1    1 346 313.11656
## 2.1  2  32   2   0 20.79   5    5    1 237  43.12752
## 3.1  3  32   1   0 19.21   1    2    1 198 338.32172
## 4.1  4  20   1   0 21.34   4    3    1 246  77.08697
## 5.1  5  30   1   0 23.98   4    3    1 270 169.02539
## 6.1  6  32   1   0 17.97   4    4    1 283 255.45773
dim(btDatLong) # check dimensions
## [1] 6000   10
```

3. Save the reformatted data into a file called `btTBregLong.tab` in such a way that
   i. Columns are tab-separated.
   ii. Column names are saved.
   iii. No row number is saved in the resulting file.

```
dir.create("Exercises_output",showWarnings=F)
write.table(btDatLong,sep="\t",col.names=T,row.names=F,file="Exercises_output/btTBregLong.tab")
```

4. Load the `btTBregHospitals.csv` data table. Join the data frames storing `btTBreg.csv` and `btTBregHospitals.csv`.

```
btDatHosp<-read.csv("dataAndSupportDocs/btTBregHospitals.csv")

head(btDatHosp) # have a look at the data
##   HID ShortName                          FullName beds     city
## 1   1      QECH Queen Elizabeth Central Hospital 1000 Blantyre
## 2   2       KCH          Kamuzu Central Hospital 1000 Lilongwe
## 3   3       ZCH           Zomba Central Hospital  400    Zomba
## 4   4       MCH           Mzuzu Central Hospital  350    Mzuzu
## 5   5    Mlambe          Mlambe Mission Hospital  254    Lunzu
dim(btDatHosp) # check dimensions of the data table
## [1] 5 5

btDatJoined<-btDat %>%
  inner_join(btDatHosp,by=c("hosp"="HID"))

head(btDatJoined) # have a look
##   id age sex hiv   bmi ses cd41 cd42   cd41.sk  cd42.sk hosp ShortName
## 1  1  44   2   0 26.32   4  346  519 313.11656 572.8906    1      QECH
## 2  2  32   2   0 20.79   5  237  337  43.12752 406.1971    5    Mlambe
## 3  3  32   1   0 19.21   1  198  328 338.32172 408.2427    2       KCH
## 4  4  20   1   0 21.34   4  246  525  77.08697 312.7572    3       ZCH
## 5  5  30   1   0 23.98   4  270  444 169.02539 335.3739    3       ZCH
## 6  6  32   1   0 17.97   4  283  372 255.45773 323.4773    4       MCH
##                          FullName beds     city
## 1 Queen Elizabeth Central Hospital 1000 Blantyre
## 2          Mlambe Mission Hospital  254    Lunzu
## 3          Kamuzu Central Hospital 1000 Lilongwe
## 4           Zomba Central Hospital  400    Zomba
## 5           Zomba Central Hospital  400    Zomba
## 6           Mzuzu Central Hospital  350    Mzuzu
dim(btDatJoined) # check dimensions
## [1] 3000   15
```

5. Compute the average patient age and the proportion of male patients for each hospital.

Useful functions for this are `aggregate()` and `group_by()`. You can however also do it manually.

- Manually:

```
# initialise new variables
btDatHosp$avgAge<-NA
btDatHosp$propMale<-NA

# iterate over hospitals
for(i in 1:nrow(btDatHosp)){
  btDatHosp$avgAge[i]<-mean(btDatJoined$age[btDatJoined$ShortName==btDatHosp$ShortName[i]],na.rm=T)
  btDatHosp$propMale[i]<-sum(btDatJoined$sex==1 &
                             btDatJoined$ShortName==btDatHosp$ShortName[i]) /
                         sum(btDatJoined$ShortName==btDatHosp$ShortName[i])
}

print(btDatHosp)
##   HID ShortName                          FullName beds     city   avgAge
## 1   1      QECH Queen Elizabeth Central Hospital 1000 Blantyre 33.14020
## 2   2       KCH          Kamuzu Central Hospital 1000 Lilongwe 32.80067
## 3   3       ZCH           Zomba Central Hospital  400    Zomba 32.99310
## 4   4       MCH           Mzuzu Central Hospital  350    Mzuzu 32.87382
## 5   5    Mlambe          Mlambe Mission Hospital  254    Lunzu 32.89950
##    propMale
## 1 0.4763514
## 2 0.4757119
## 3 0.4948276
## 4 0.4731861
## 5 0.5242881
```

- Using `aggregate()`

```
btDat$hosp<-factor(btDat$hosp)
btDatHosp$avgAge<-aggregate(btDatJoined$age,FUN=mean,by=list(btDat$hosp))$x
btDatHosp$propMale<-aggregate(ifelse(btDatJoined$sex==1,1,0),FUN=mean,by=list(btDat$hosp))$x

print(btDatHosp)
##   HID ShortName                          FullName beds     city   avgAge
## 1   1      QECH Queen Elizabeth Central Hospital 1000 Blantyre 33.14020
## 2   2       KCH          Kamuzu Central Hospital 1000 Lilongwe 32.80067
## 3   3       ZCH           Zomba Central Hospital  400    Zomba 32.99310
## 4   4       MCH           Mzuzu Central Hospital  350    Mzuzu 32.87382
## 5   5    Mlambe          Mlambe Mission Hospital  254    Lunzu 32.89950
##    propMale
## 1 0.4763514
## 2 0.4757119
## 3 0.4948276
## 4 0.4731861
## 5 0.5242881
```

- Using `group_by()`

```
tmp<-btDat %>%
  group_by(hosp) %>%
  summarise(avgAge=mean(age,na.rm=T))
```

```
btDatHosp$avgAge<-tmp$avgAge

tmp<-btDat %>%
  group_by(hosp) %>%
  summarise(propMale=mean(ifelse(sex==1,1,0),na.rm=T))
btDatHosp$propMale<-tmp$propMale

print(btDatHosp)
##   HID ShortName                          FullName beds     city   avgAge
## 1   1      QECH Queen Elizabeth Central Hospital 1000 Blantyre 33.14020
## 2   2       KCH          Kamuzu Central Hospital 1000 Lilongwe 32.80067
## 3   3       ZCH           Zomba Central Hospital  400    Zomba 32.99310
## 4   4       MCH           Mzuzu Central Hospital  350    Mzuzu 32.87382
## 5   5    Mlambe          Mlambe Mission Hospital  254    Lunzu 32.89950
##    propMale
## 1 0.4763514
## 2 0.4757119
## 3 0.4948276
## 4 0.4731861
## 5 0.5242881
```

6. Write an R function that computes the following summary statistics, then, using your custom function, compute these for the `bmi`, `cd41`, `cd42` columns:
   i. mean
   ii. median
   iii. interquartile range
   iv. minimum
   v. maximum
   vi. number of missing values

```
summaryFun<-function(x){
  return(c(
    mean(x,na.rm=T),
    median(x),
    paste(sep="","(",paste(collapse=",",quantile(x,probs=c(0.25,0.75))),")"),
    min(x,na.rm=T),
    max(x,na.rm=T),
    sum(is.na(x))
  ))
}

res<-apply(btDat[,c("bmi","cd41","cd42")],MARGIN=2,FUN=summaryFun)
rownames(res)<-c("mean","median","IQR","min","max","num_MV")
print(res)
##        bmi                cd41               cd42
## mean   "23.0574333333333" "248.794333333333" "448.003"
## median "23.05"            "249"              "447"
## IQR    "(21.34,24.74)"    "(216,281)"        "(381,515)"
## min    "12.64"            "57"               "81"
## max    "31.14"            "447"              "843"
## num_MV "0"                "0"                "0"
```

7. Do the same now, but only for female patients. Repeat for only male patients.

```
resF<-apply(btDat[btDat$sex==2,c("bmi","cd41","cd42")],MARGIN=2,FUN=summaryFun)
rownames(resF)<-c("mean","median","IQR","min","max","num_MV")
print(resF)
##         bmi                 cd41                cd42
## mean    "23.1218644067797" "248.473924380704" "446.675358539765"
## median  "23.14"             "250"              "447.5"
## IQR     "(21.365,24.82)"   "(215,281)"        "(379,512)"
## min     "12.64"            "57"               "138"
## max     "31.14"            "447"              "820"
## num_MV  "0"                "0"                "0"

resM<-apply(btDat[btDat$sex==1,c("bmi","cd41","cd42")],MARGIN=2,FUN=summaryFun)
rownames(resM)<-c("mean","median","IQR","min","max","num_MV")
print(resM)
##         bmi                 cd41                cd42
## mean    "22.9900136425648" "249.129604365621" "449.392223738063"
## median  "22.98"            "248"              "447"
## IQR     "(21.3,24.66)"     "(216,282)"        "(383,519.75)"
## min     "14.44"            "71"               "81"
## max     "30.9"             "414"              "843"
## num_MV  "0"                "0"                "0"
```

Exercise 2

Using the `iris` dataset (type `?iris` to get more information about this dataset) that comes pre-loaded with R, produce the following figures:

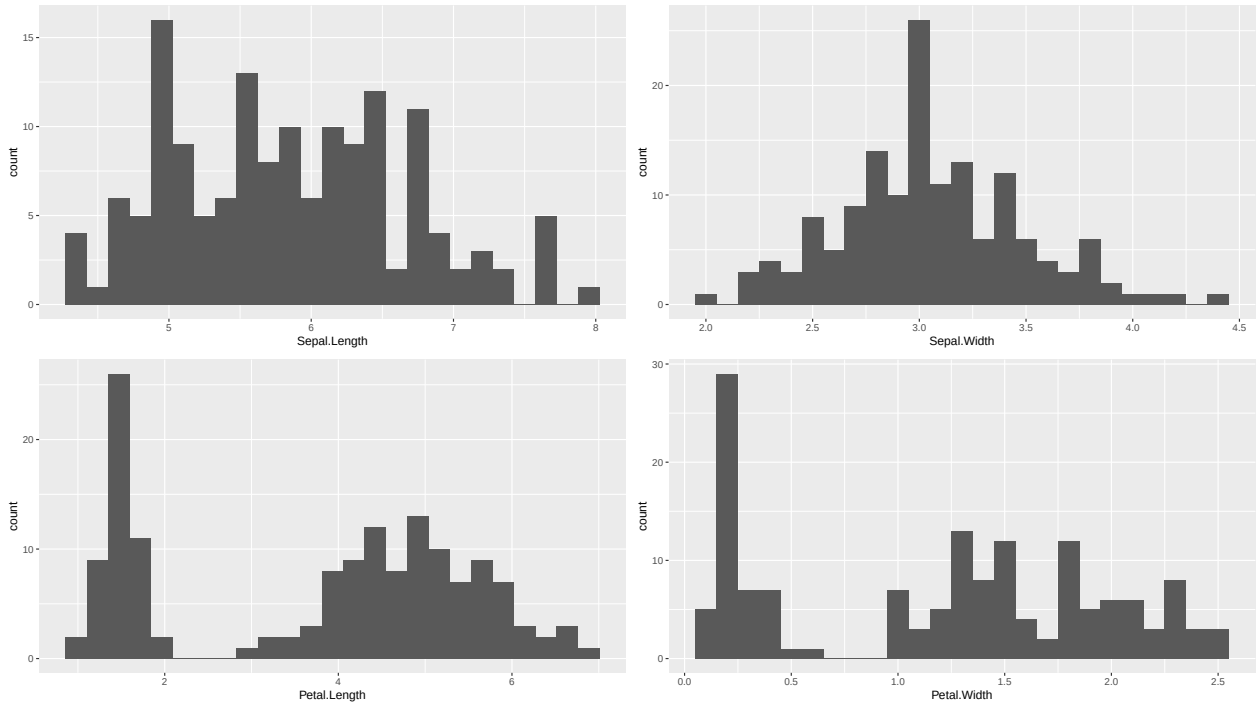
- Produce histograms for each of `Sepal.Length`, `Sepal.Width`, `Petal.Length`, `Petal.Width`.
- Produce a bar plot for `Species`.
- Produce box and whisker plots for each of the 4 continuous variables. Put them all on a single, multi-panel figure.
- Repeat for just `Sepal.Length` using a violin plot, stratifying by `Species`.
- Produce a single graph (not multi-panel) that has histograms for `Sepal.Length` for each of the 3 flower species.
- There are 4 continuous variables. This means there are 6 possible pairs of these. For each such pair, produce a scatter plot of one variable against the other and highlight the different flower species by using a different colour for each species.
- For one of these 6 scatter plots: estimate the bivariate probability density and add density contour lines to the figure.

Exercise 2 (solution)

Using the `iris` dataset (type `?iris` to get more information about this dataset) that comes pre-loaded with R, produce the following figures:

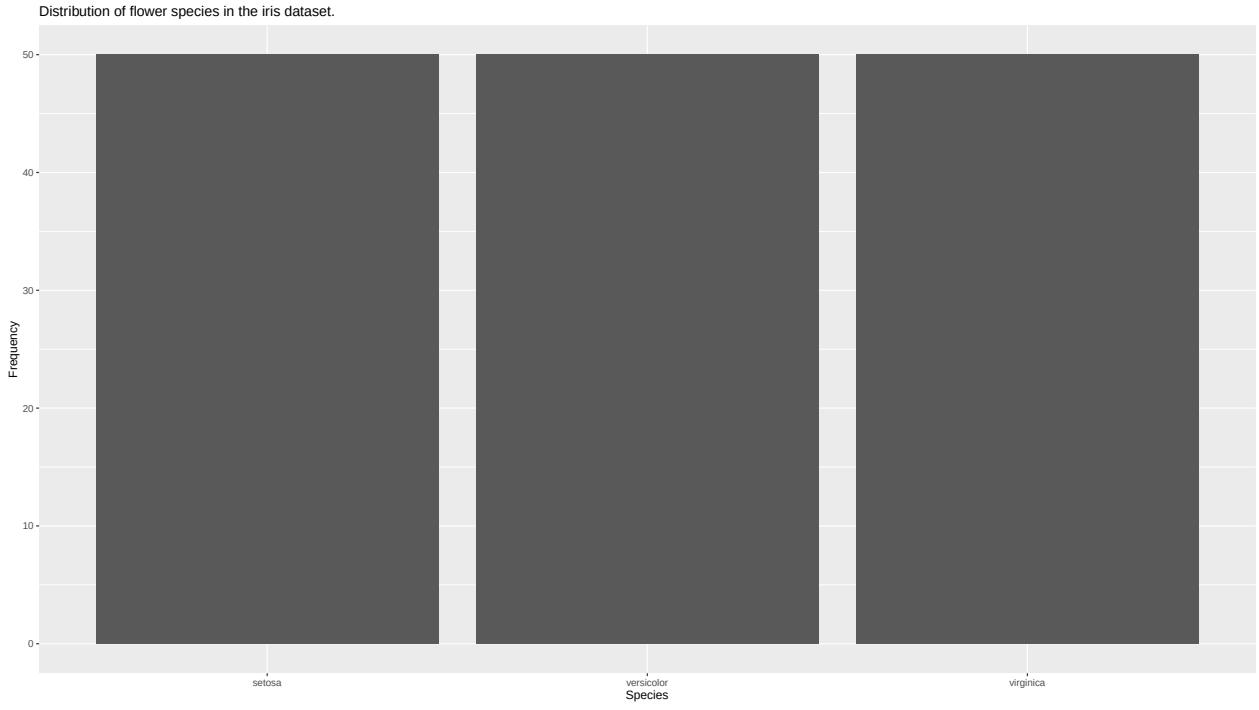
- Produce histograms for each of `Sepal.Length`, `Sepal.Width`, `Petal.Length`, `Petal.Width`.

```
g<-list()
g[[1]]<-ggplot(data=iris,mapping=aes(x=Sepal.Length)) + geom_histogram(bins=25)
g[[2]]<-ggplot(data=iris,mapping=aes(x=Sepal.Width)) + geom_histogram(bins=25)
g[[3]]<-ggplot(data=iris,mapping=aes(x=Petal.Length)) + geom_histogram(bins=25)
g[[4]]<-ggplot(data=iris,mapping=aes(x=Petal.Width)) + geom_histogram(bins=25)
grid.arrange(g[[1]],g[[2]],g[[3]],g[[4]],nrow=2)
```



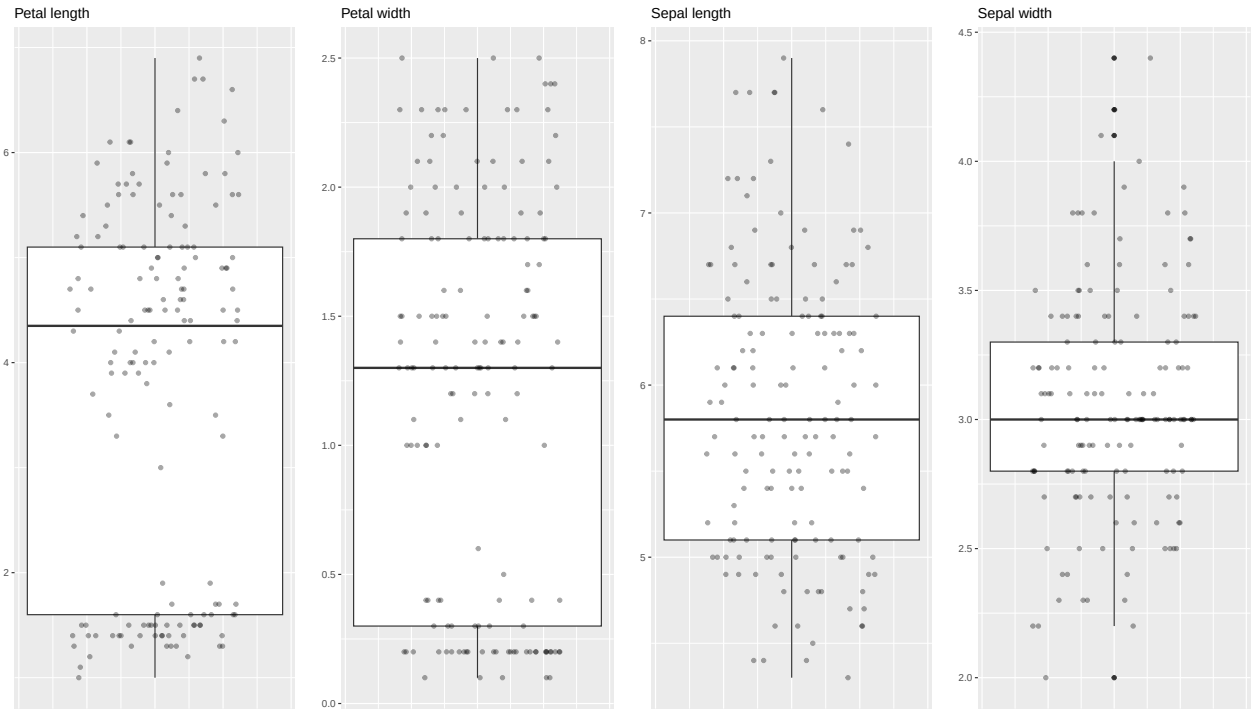
- Produce a bar plot for `Species`.

```
iris %>%
  ggplot(mapping=aes(x=Species)) +
  geom_bar() +
  labels(title="Distribution of flower species in the iris dataset.") +
  ylab("Frequency")
```



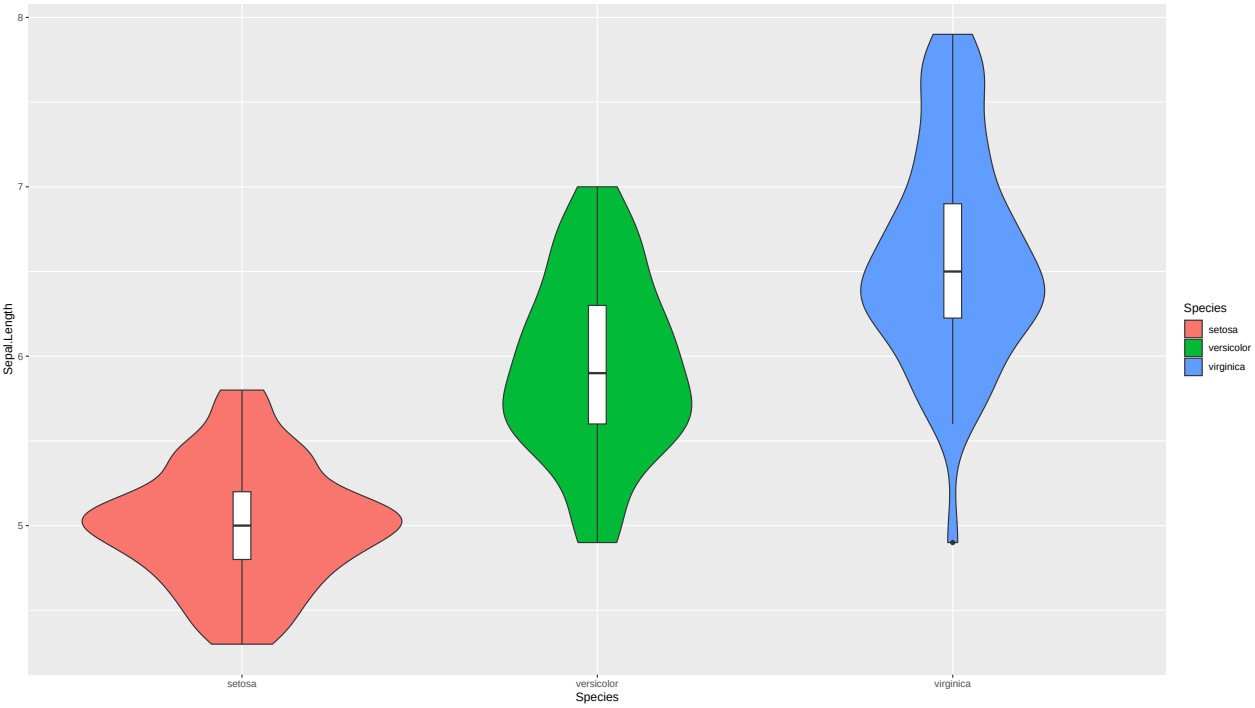
- Produce box and whisker plots for each of the 4 continuous variables. Put them all on a single, multi-panel figure.

```
g1<-iris %>%
  ggplot(mapping=aes(x=1,y=Petal.Length)) +
  geom_boxplot() +
  geom_jitter(height=0,width=0.25,alpha=0.35) +
  labs(title="Petal length") +
  ylab("") +
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())

g2<-iris %>%
  ggplot(mapping=aes(x=1,y=Petal.Width)) +
  geom_boxplot() +
  geom_jitter(height=0,width=0.25,alpha=0.35) +
  labs(title="Petal width") +
  ylab("") +
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())

g3<-iris %>%
  ggplot(mapping=aes(x=1,y=Sepal.Length)) +
  geom_boxplot() +
  geom_jitter(height=0,width=0.25,alpha=0.35) +
  labs(title="Sepal length") +
  ylab("") +
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())

g4<-iris %>%
  ggplot(mapping=aes(x=1,y=Sepal.Width)) +
  geom_boxplot() +
  geom_jitter(height=0,width=0.25,alpha=0.35) +
  labs(title="Sepal width") +
  ylab("") +
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())

grid.arrange(g1,g2,g3,g4,nrow=1)
```



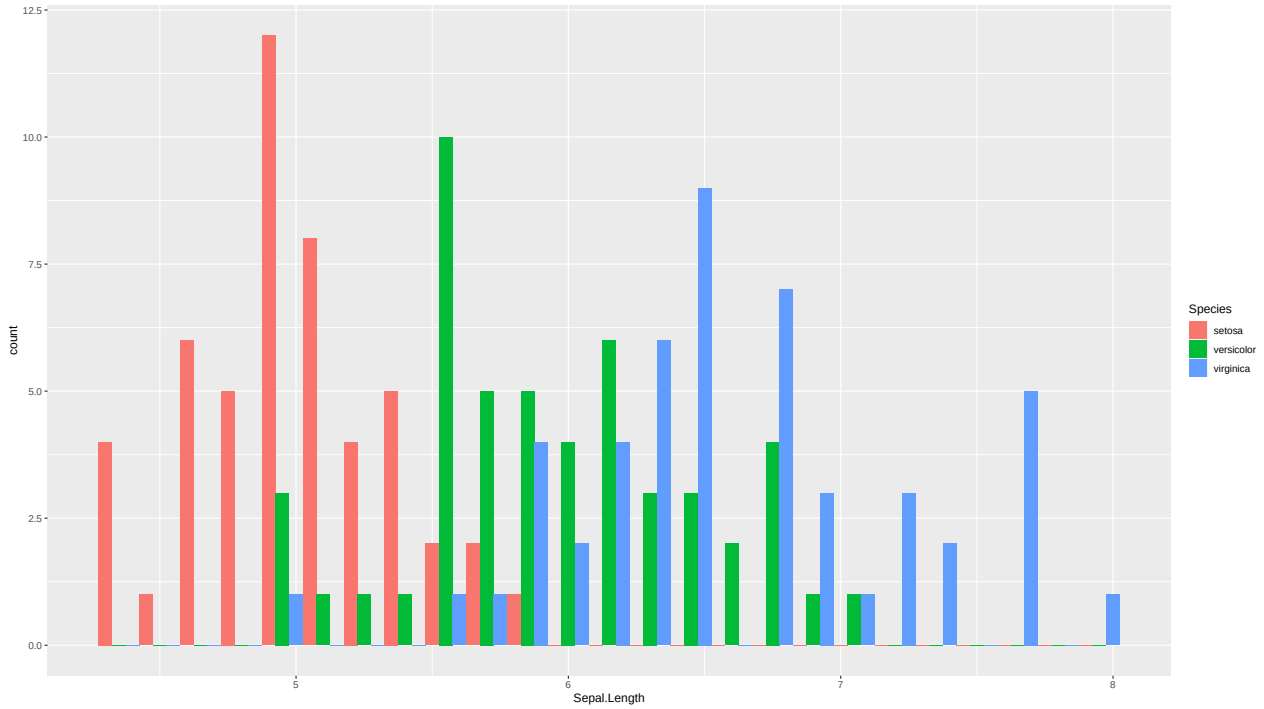
- Repeat for just `Sepal.Length` using a violin plot, stratifying by `Species`.

```
ggplot(data=iris,mapping=aes(x=Species,y=Sepal.Length,fill=Species)) +
  geom_violin() +
  geom_boxplot(width=0.05, fill="white")
```



- Produce a single graph (not multi-panel) that has histograms for `Sepal.Length` for each of the 3 flower species.

```
ggplot(data=iris,mapping=aes(x=Sepal.Length,fill=Species)) +
  geom_histogram(binwidth=0.15,position="dodge")
```



- There are 4 continuous variables. This means there are 6 possible pairs of these. For each such pair, produce a scatter plot of one variable against the other and highlight the different flower species by using a different colour for each species.

```
g<-list()
counter<-0

for(i in 1:3){
  for(j in min(c(i+1),4):4){
    counter<-counter+1

    g[[counter]]<-iris %>%
      ggplot(mapping=aes(x=get(colnames(iris)[i]),y=get(colnames(iris[j])),col=Species)) +
      geom_point() +
      scale_color_manual(values=c("steelblue","orange","salmon")) +
      xlab(colnames(iris)[i]) +
      ylab(colnames(iris)[j])
  }
}
```

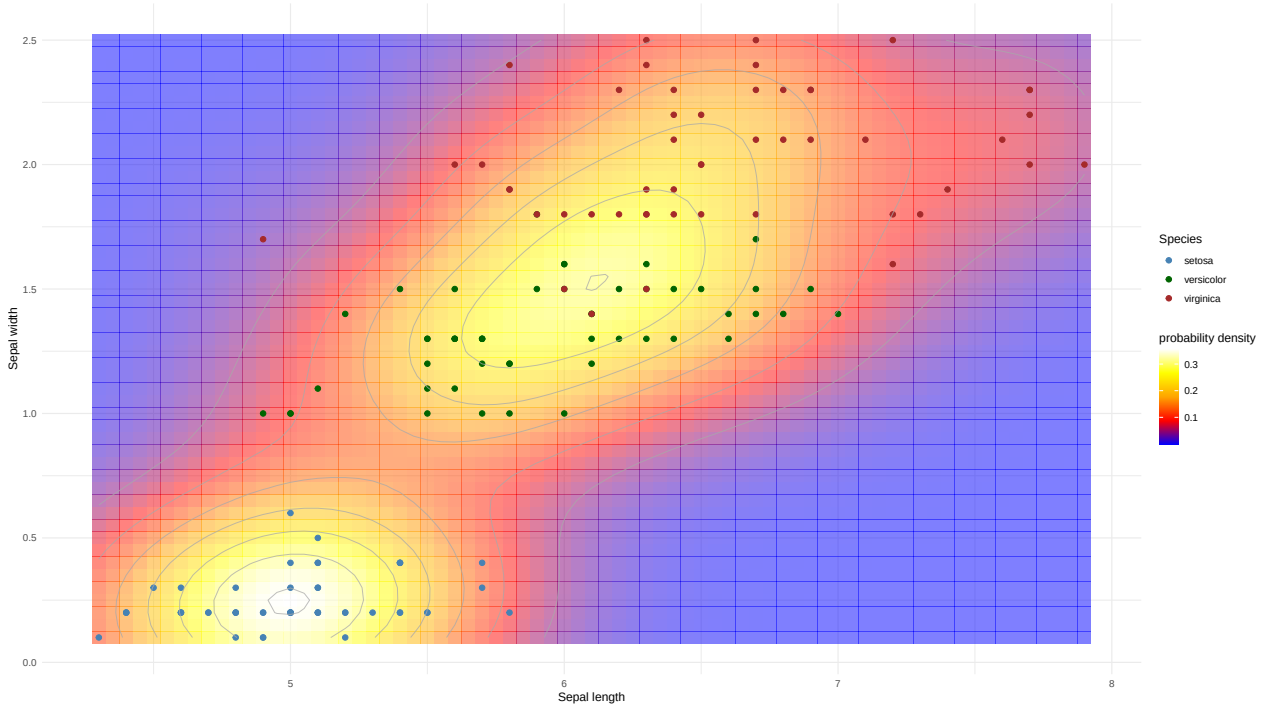
- For one of these 6 scatter plots: estimate the bivariate probability density and add density contour lines to the figure.

This requires a bit of extra work and so you have likely found this harder:

1. Estimate the 2-dimensional density.
2. Using multiple geoms with different datasets.

```
library(MASS)
##
```

```
## Attaching package: 'MASS'
## The following object is masked from 'package:dplyr':
##
##     select
clrs<-colorRampPalette(c("blue","red","orange","yellow","white"))
dens <- kde2d(iris$Sepal.Length,
              iris$Petal.Width,
              n=c(length(seq(min(iris$Sepal.Length),max(iris$Sepal.Length),by=0.05)),
                  length(seq(min(iris$Petal.Width),max(iris$Petal.Width),by=0.05))))

df<-expand.grid(dens$x,dens$y)
df$z<-as.vector(dens$z)
colnames(df)<-c("x","y","z")

ggplot() +
  geom_tile(data=df,mapping=aes(x=x,y=y,fill=z,z=z),width=0.05,height=0.05,alpha=0.5) +
  geom_point(data=iris,mapping=aes(x=Sepal.Length,y=Petal.Width,col=Species),size=2) +
  geom_contour(data=df,mapping=aes(x=x,y=y,fill=z,z=z),col="darkgrey",lwd=0.35,alpha=0.75) +
  scale_fill_gradientn(colours = clrs(200),name="probability density") +
  scale_color_manual(values=c("steelblue","darkgreen","brown")) +
  theme_minimal() +
  xlab("Sepal length") +
  ylab("Sepal width")
```



Exercise 3

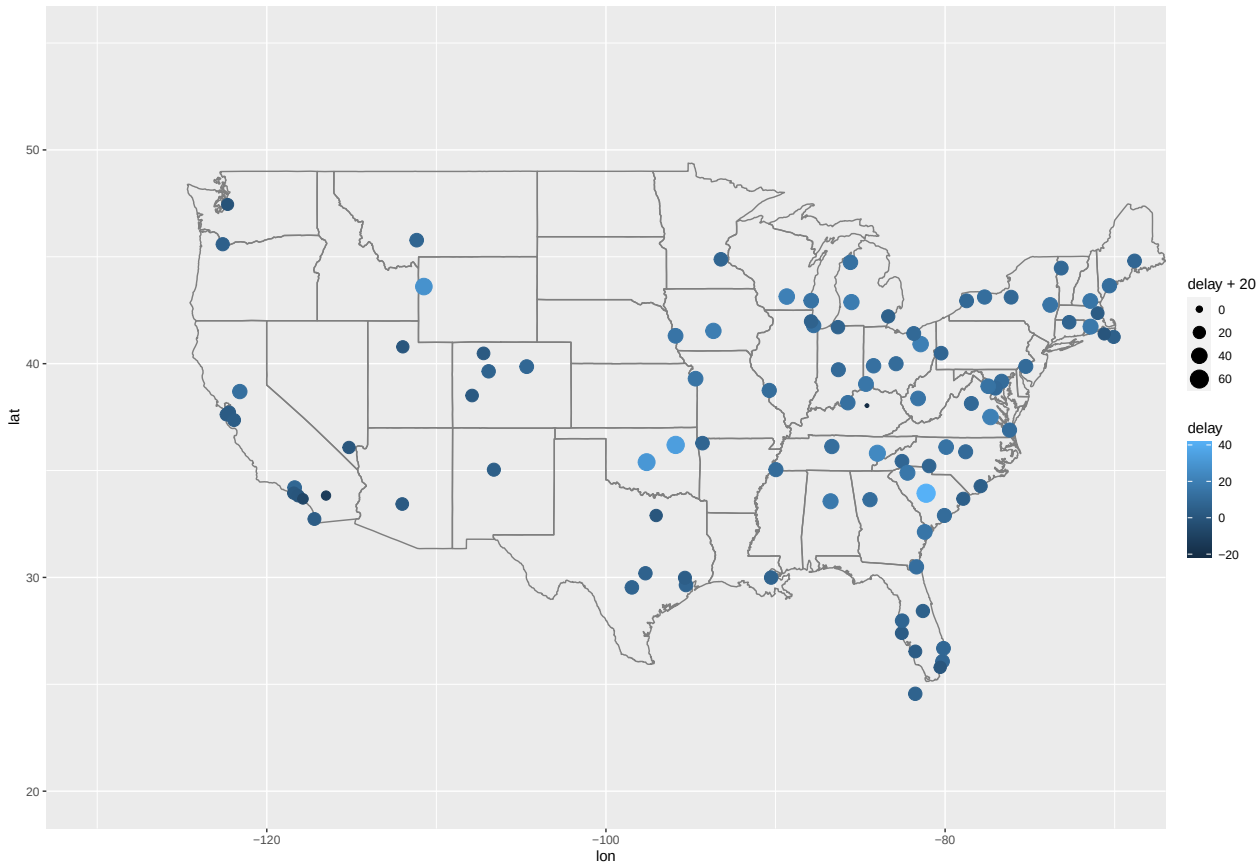
Install the package `nycflights13`, then load it. This has data on flights that took off in the US during 2013. There are 5 data tables: + `airlines`, data on airlines + `airports`, data on airports + `planes`, data on planes + `weather`, hourly weather data at NYC airports for 2013 + `flights`, data on flights leaving NYC airports during 2013

- Compute the average delay by destination, then join the airports data frame to get the longitude and latitude of delays. Plot this (if you are using ggplot2, then the functions `borders()` and `coord_quickmap()` can be useful for a nicer figure).
- Construct data frames giving average delay per wind speed / temperature / precipitation / visibility. Produce scatter plots of each of these against delay and add an average trend line.

Exercise 3 (solution)

- Compute the average delay by destination, then join the airports data frame to get the longitude and latitude of delays. Plot this (if you are using ggplot2, then the functions `borders()` and `coord_quickmap()` can be useful for a nicer figure).

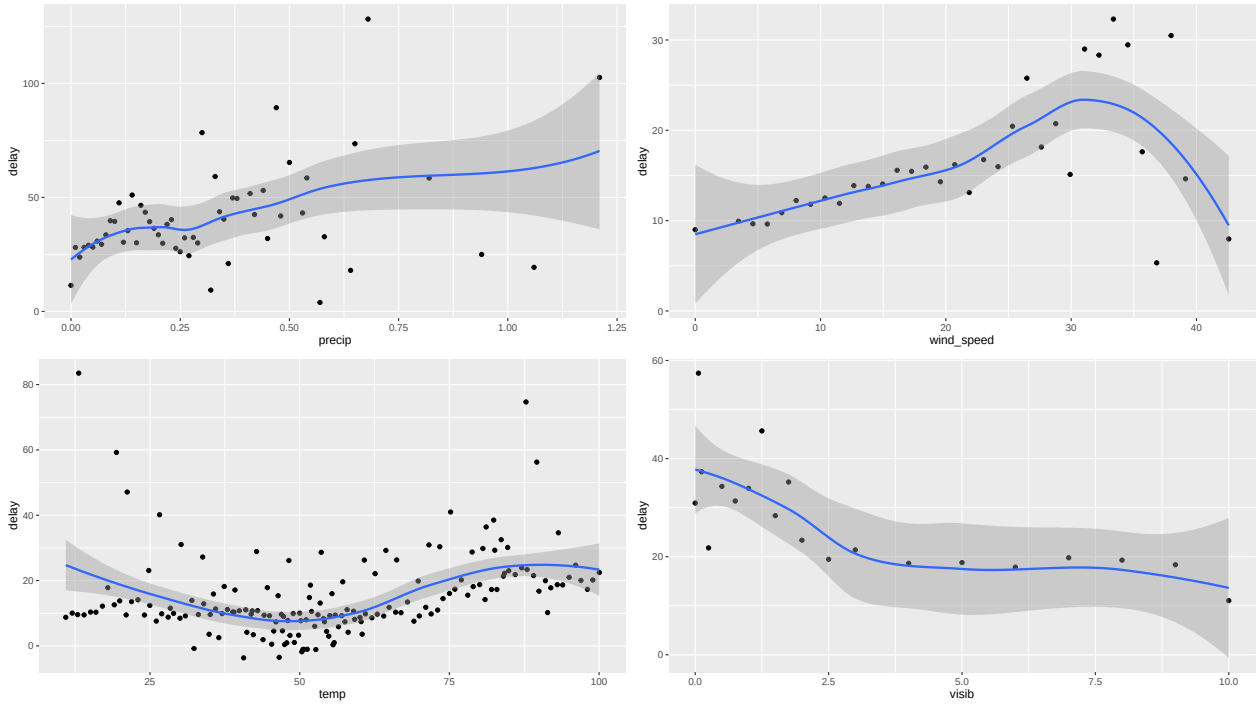
```
library(nycflights13)

# Compute the average delay by destination, then join the airports data frame
# to get the longitude and latitude of delays.
avg_dest_delays <-
    flights %>%
    group_by(dest) %>%
    summarise(delay = mean(arr_delay, na.rm = TRUE)) %>% # arrival delay NA's are cancelled flights
    inner_join(airports, by = c(dest = "faa"))

# stratify by origin airport
avg_dest_delays_by_origin <-
    flights %>%
    group_by(origin,dest) %>%
    summarise(delay = mean(arr_delay, na.rm = TRUE)) %>% # arrival delay NA's are cancelled flights
    inner_join(airports, by = c(dest = "faa"))

# plotting this
ggplot(data=avg_dest_delays,mapping=aes(lon,lat,colour=delay,size=delay+20)) +
  borders("state") +
  geom_point() +
  coord_quickmap(xlim=c(-130,-70),ylim=c(20,55)) # xlim, ylim to hide Alaska and Hawaii
```



- Construct data frames giving average delay per wind speed / temperature / precipitation / visibility. Produce scatter plots of each of these against delay and add an average trend line.

```
flights_weather <-flights %>% left_join(weather,by=c("year","month","day","hour"))

flights_precip <- flights_weather %>%
  group_by(precip) %>%
  summarise(delay=mean(dep_delay,na.rm=T))

flights_wind <- flights_weather %>%
  group_by(wind_speed) %>%
  summarise(delay=mean(dep_delay,na.rm=T))

flights_temp <- flights_weather %>%
  group_by(temp) %>%
  summarise(delay=mean(dep_delay,na.rm=T))

flights_visib <- flights_weather %>%
  group_by(visib) %>%
  summarise(delay=mean(dep_delay,na.rm=T))

g1<-ggplot(data=flights_precip,mapping=aes(x=precip,y=delay)) +
  geom_point() +
  geom_smooth()

g2<-ggplot(data=flights_wind,mapping=aes(x=wind_speed,y=delay)) +
  geom_point() +
```

```
  geom_smooth()

g3<-ggplot(data=flights_temp,mapping=aes(x=temp,y=delay)) +
  geom_point() +
  geom_smooth()

g4<-ggplot(data=flights_visib,mapping=aes(x=visib,y=delay)) +
  geom_point() +
  geom_smooth()

grid.arrange(g1,g2,g3,g4)
```



Exercise 4

Decide what design could be used to answer the following research questions:

1. What is the prevalence of HIV in urban Blantyre in 2018?
2. Do men experience higher mortality compared to women once they start ART?
3. Does smoking increase the chance of having lung cancer?
4. What is the effect of providing oral HIV self-test kits on the uptake of HIV testing?
5. What interventions may improve linkage to ART following community based HIV testing?

Exercise 4 (Solution)

The answers below may not be the only valid answers - there may be several alternative designs for a given question.

1. If today was 2018, then taking a cross-sectional, random sample from the Blantyre population in 2018 will allow you to answer the question. Given that 2018 is in the past now, a retrospective design will need to be used.

2. A longitudinal design where a cohort of equal numbers of men and women, recruited at ART initiation, are followed over time will be appropriate for this question.
3. You could again recruit a cohort for a longitudinal study. However you will need a big budget and a lot of time: lung cancer is rare and to develop lung cancer takes years. So here a case-control study may be more efficient: recruit lung cancer patients from a hospital, then recruit matched (by age, sex and other known factors to impact the risk of lung cancer) or unmatched controls. Then by comparing smoking habits between controls and cases, you may be able to answer the research question (somewhat - the causality implied by the question will be tricky to resolve).
4. The appropriate design depends on the practical circumstances. If there is a government programme distributing self-test kits, then a pragmatic before-after study design will need to be used. However if no such programmes exist, then an intervention study, specifically a randomised controlled trial, where participants (or more likely health centres where these kits would be distributed, making this a cluster design) are randomised to either receiving HIV self-test kits or not, will be an appropriate design.
5. The question implies that there are a number of potential interventions and the idea is to both identify effective interventions and evaluate their effect. This suggests an adaptive interventional design, such as a multi-arm multi-stage design, could be useful.

Exercise 5

Take the iris dataset and explain how you would, in a formal statistical way, compare the following:

1. `Petal.Width` between the flower species `virginica` and `setosa`.
2. `Sepal.Length` between all 3 flower species.

For each comparison, state which test you will use (there may be more than one valid option!), state the null and alternative hypotheses, do the test and interpret the results.

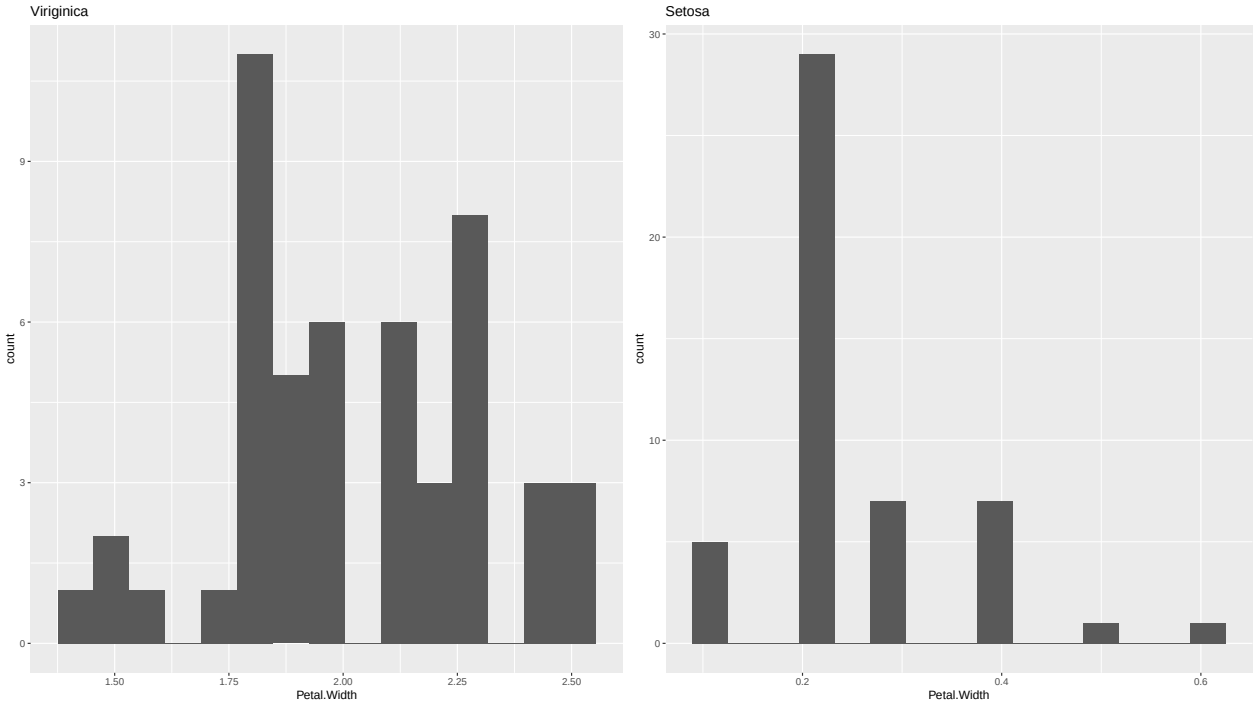
Exercise 5 (solution)

1. `Petal.Width` between the flower species `virginica` and `setosa`.

We have 50 observations for each flower type – large enough for the Central Limit Theorem (CLT) to guarantee that sample means are approximately normally distributed as long as the data are not too severely non-normal (outliers etc).

Let's quickly check the distribution of `Petal.Width` in the 2 flower species:

```
g1<-iris %>%
  filter(Species=="virginica") %>%
  ggplot(mapping=aes(x=Petal.Width)) +
  geom_histogram(bins=15) +
  ggtitle("Viriginica")

g2<-iris %>%
  filter(Species=="setosa") %>%
  ggplot(mapping=aes(x=Petal.Width)) +
  geom_histogram(bins=15) +
  ggtitle("Setosa")

grid.arrange(g1,g2,nrow=1)
```



The data do not look particularly normally distributed, but there is no instance of severe non-normality either. The t-test should be OK to use, given the CLT.

As we only want to assess whether `Petal.Width` is the same or not across the 2 flower species, we will do a two-sided test. We have no reason to believe one or the other flower species should have larger values.

For a 2 sample t-test, the null and alternative hypotheses are:

$$H_0 : \mu_v = \mu_s$$
$$H_1 : \mu_v \neq \mu_s$$

where μ_v, μ_s are the population means for the virginica and setosa flower species respectively.

We can now proceed to do the two-sided, two-sample t-test:

```
t.test(Petal.Width~Species,data=iris %>% filter(Species %in% c("virginica","setosa")))
##
##  Welch Two Sample t-test
##
## data:  Petal.Width by Species
## t = -42.786, df = 63.123, p-value < 2.2e-16
## alternative hypothesis: true difference in means between group setosa and group virginica is not equ
## 95 percent confidence interval:
##  -1.863133 -1.696867
## sample estimates:
##    mean in group setosa mean in group virginica
##                   0.246                   2.026
```

The p-value is essentially 0, so we reject the null hypothesis that the mean `Petal.Width` is the same in both groups. Under the null hypothesis H_0 it is very unlikely that we would have observed the data we collected, hence we H_0.

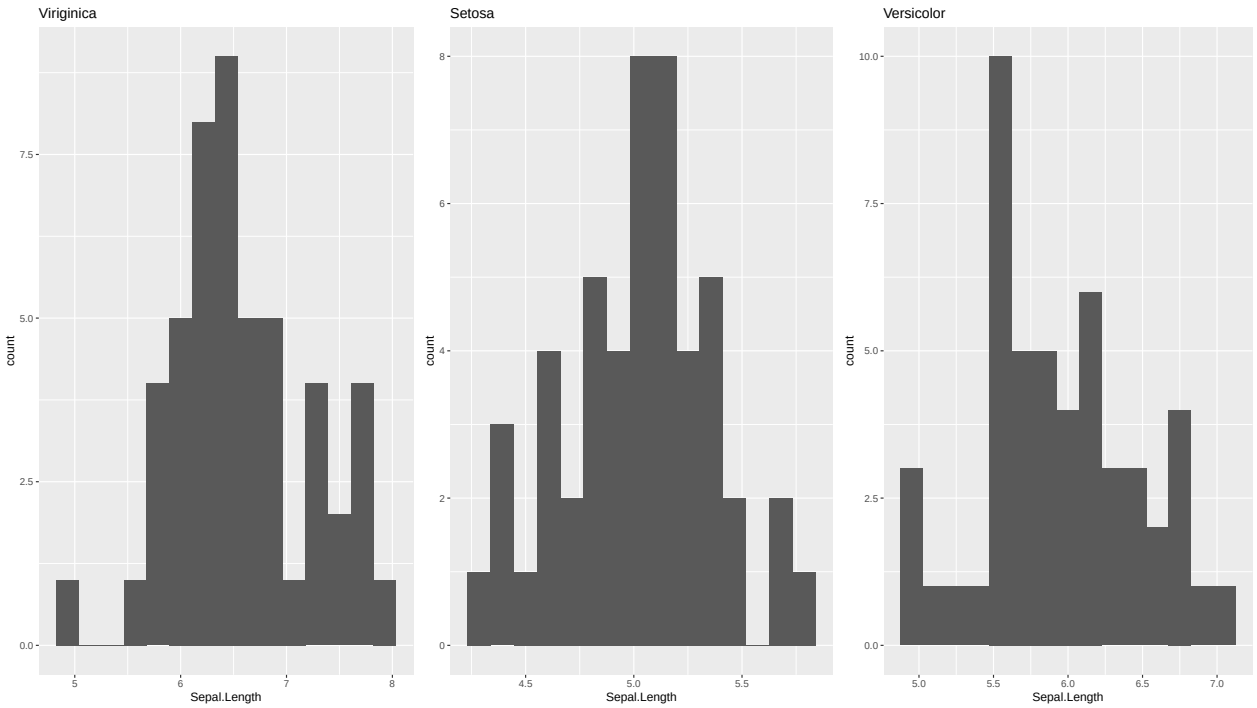
Note: it would also be OK to do a Wilcoxon rank-sum test and this gives the same result (p-value essentially 0, reject H_0):

```
wilcox.test(Petal.Width~Species,data=iris %>% filter(Species %in% c("virginica","setosa")))
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Petal.Width by Species
## W = 0, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

2. `Sepal.Length` between all 3 flower species.

Now we are comparing 3 groups, not 2. To check whether we can use ANOVA or need to use the Kruskal-Wallis test, we need inspect that the data are not severely non-normal:

```
g1<-iris %>%
  filter(Species=="virginica") %>%
  ggplot(mapping=aes(x=Sepal.Length)) +
  geom_histogram(bins=15) +
  ggtitle("Viriginica")

g2<-iris %>%
  filter(Species=="setosa") %>%
  ggplot(mapping=aes(x=Sepal.Length)) +
  geom_histogram(bins=15) +
  ggtitle("Setosa")

g3<-iris %>%
  filter(Species=="versicolor") %>%
  ggplot(mapping=aes(x=Sepal.Length)) +
  geom_histogram(bins=15) +
  ggtitle("Versicolor")

grid.arrange(g1,g2,g3,nrow=1)
```



This looks OK to use ANOVA.

The null and alternative hypotheses will be:

$$H_0 : \mu_s = \mu_{ve} = \mu_{vi}$$
$$H_1 : \mu_i \neq \mu_j \quad \text{for some i,j}$$

```
oneway.test(Sepal.Length~Species,data=iris)
##
##  One-way analysis of means (not assuming equal variances)
##
## data:  Sepal.Length and Species
## F = 138.91, num df = 2.000, denom df = 92.211, p-value < 2.2e-16
```

The p-value is again essentially 0 and so we reject the null hypothesis H_0. We conclude that there is enough evidence to suggest that the mean values for `Sepal.Length` are different across the 3 flower species.

Note: as above, we can always do the non-parametric test. Nonparametric tests have slightly less power than the parametric tests if the parametric assumptions are met, but that does not mean that it's not possible to use the non-parametric test when you can use an equivalent parametric test.

Here, if we used Kruskal-Wallis:

The null and alternative hypotheses are somehwat different:

$$H_0 : \text{Sepal.Length in all groups has the same distribution.}$$
$$H_1 : \text{The distribution of Sepal.Length is not the same across all groups.}$$

```
kruskal.test(Sepal.Length~Species,data=iris)
##
##  Kruskal-Wallis rank sum test
##
```

```
## data:  Sepal.Length by Species
## Kruskal-Wallis chi-squared = 96.937, df = 2, p-value < 2.2e-16
```

The p-value is essentially 0, so we reject the null hypothesis.

Exercise 6

In a drug trial, researchers are assessing overall in-hospital mortality as the primary outcome. The new drug is compared against the standard-of-care treatment (SOC). Patients are randomised 1:1 to the new drug and SOC. At trial conclusion, the researchers observe that out of 250 SOC patients, 61 have died and out of 250 patients on the new drug arm, 48 have died.

Perform a statistical test to conclude whether or not there is a difference between the new drug and the SOC. State the test you use, the null and alternative hypotheses, perform the test and interpret the results.

Exercise 6 (solution)

Here we need to compare the proportions of patients that die in hospital during the study period, so we will need to do a two-sample test for proportions.

Let p_{drug}, p_{SOC} be the proportion of patients dying on the new drug regime and on the SOC arm respectively.

The null and alternative hypotheses are:

$$H_0 : p_{drug} = p_{SOC}$$
$$H_1 : p_{drug} \neq p_{SOC}$$

Performing the test, we get:

```
res<-prop.test(x=c(48,61),n=c(250,250))
res
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  c(48, 61) out of c(250, 250)
## X-squared = 1.6894, df = 1, p-value = 0.1937
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.12823737  0.02423737
## sample estimates:
## prop 1 prop 2
##  0.192  0.244
```

The p-value is $0.1936817 > 0.05$, so we do not reject H_0, there is not enough evidence (at the 5% significance level) to suggest that the proportions in both groups are different,

Note that a Fisher's exact test could also be used here (note the different specification of the data for this test):

```
res<-fisher.test(x=matrix(c(48,61,250-48,250-61),byrow=T,nrow=2))
res
##
##  Fisher's Exact Test for Count Data
##
## data:  matrix(c(48, 61, 250 - 48, 250 - 61), byrow = T, nrow = 2)
## p-value = 0.1935
```

```
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.4686843 1.1531506
## sample estimates:
## odds ratio
##  0.7367018
```

The p-value is 0.1935381, almost the same as for the two-proportion test (which uses a normal distribution approximation) and since this is > 0.05, we do not reject H_0. As for the two-proportion test, we conclude that there is not enough evidence (at the 5% significance level) to suggest that the proportions in both groups are different.

Exercise 7

Test whether the 2 variables from Table 1 below are independent or not. State the test you use, the null and alternative hypotheses, do the test and interpret the results.

Table 1: Summary of patient outcomes for different health centers.

	alive	dead
Hospital1	92	29
Hospital2	54	15
Hospital3	31	3

What when you repeat your analysis for Table 2 below?

Table 2: Summary of patient outcomes for different health centers.

	alive	dead
Hospital1	920	290
Hospital2	540	150
Hospital3	310	30

Comment on the results from your analyses for both tables.

Exercise 7 (solution)

We are assessing whether 2 categorical variables are independent or not. If possible, we would use the exact Fisher test for this.

The null and alternative hypotheses are:

$$H_0: \text{ Health centre and outcome are independent.}$$

$$H_0: \text{ Health centre and outcome are not independent.}$$

Note that we could also express this as $H_0 : p_1 = p_2 = p_3$ and $H_1 : p_i \neq p_j$ for some i, j.

We can proceed to do the test:

```
res<-fisher.test(matrix(c(92,29,54,15,31,3),byrow=T,ncol=2))
res
##
##  Fisher's Exact Test for Count Data
##
## data:  matrix(c(92, 29, 54, 15, 31, 3), byrow = T, ncol = 2)
## p-value = 0.1483
## alternative hypothesis: two.sided
```

The p-value is $0.1483358 > 0.05$, so at the 5% significance level we do not reject H_0, there is not enough evidence to suggest that outcome depends on the health centre.

Everything (test to use, null and alternative hypotheses) remains to same for the table with larger counts (note that Table 2 has the same cell counts as Table 1, just multiplied by 10 – the sample size is 10 times larger), but we get a different result.

```
res<-fisher.test(matrix(c(920,290,540,150,310,30),byrow=T,ncol=2))
res
##
##  Fisher's Exact Test for Count Data
##
## data:  matrix(c(920, 290, 540, 150, 310, 30), byrow = T, ncol = 2)
## p-value = 5.149e-10
## alternative hypothesis: two.sided
```

Now the p-value is $5.1493727 \times 10^{-10} < 0.05$, so we reject H_0 at the 5% significance level. There is considerable evidence that the outcome depends on the health centre.

What has changed is simply the sample size: with the larger sample size we can conclude that the same differences in outcome proportions between health centres are not due to random chance, but are likely a real feature. For the table with less counts, we did not have enough evidence to conclude this – it was reasonably possible to observe the data even under the null hypothesis H_0.

Exercise 8

Using the `adolescent_small.csv` data, fit a linear model regressing weight (variable `a104wt`) on age (variable `a12age`).

Test if the regression coefficient of age $\beta_{age} = 0$.

Note:

- deviance = sum of squares
- residual = error

Exercise 8 (solution)

```
ado<-read.csv("dataAndSupportDocs/adolescent_small.csv")

mod1<-glm(a104wt~a12age,data=ado)
# print(mod1)

summary(mod1)
##
## Call:
```

```
## glm(formula = a104wt ~ a12age, data = ado)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -26.1077   -6.5219   -0.3766    6.1234   29.1485
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -12.2481     3.0327  -4.039 7.05e-05 ***
## a12age        3.5562     0.2213  16.067  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 95.29828)
##
##     Null deviance: 49760  on 265  degrees of freedom
## Residual deviance: 25159  on 264  degrees of freedom
##   (35 observations deleted due to missingness)
## AIC: 1971
##
## Number of Fisher Scoring iterations: 2
```

The output above contains all the information we need for this exercise.

The model equation for the model above is simply:

$$Y = \beta_0 + \beta_1 \cdot X + \epsilon$$

where Y is the response variable weight (`a104wt`) and X is the predictor variable age (`a12age`).

We are meant to test the null hypothesis H_0 against the alternative H_1:

$$H_0 : \beta_1 = 0$$
$$H_1 : \beta_1 \neq 0$$

In the table above, we get this from the section `Coefficients`, by finding the row for `a12age` and the p-value for the test of H_0 is the one given in the column `Pr(>|t|)`. Here this is $5.5524318 \times 10^{-41}$ (shown as `<2e-16` in the output; i.e. essentially 0).

Since this p-value is < 0.05, we reject H_0 at the 5% significance level – there is sufficient evidence that the coefficient of `a12age` is not equal to 0. In other words, we reject the null hypothesis of no association between `a104wt` and `a12age`.

Alternatively, you could also calculate the p-value also manually, using an F-test:

```
F<-((mod1$null.deviance-mod1$deviance)/1)/((mod1$deviance)/mod1$df.residual)
P<-1-pf(F,df1=1,df2=mod1$df.residual)
print(P)
## [1] 0
```

Exercise 9

Using the `adolescent_small.csv` data, fit the following GLM model:

Weight `a104wt` as a function of

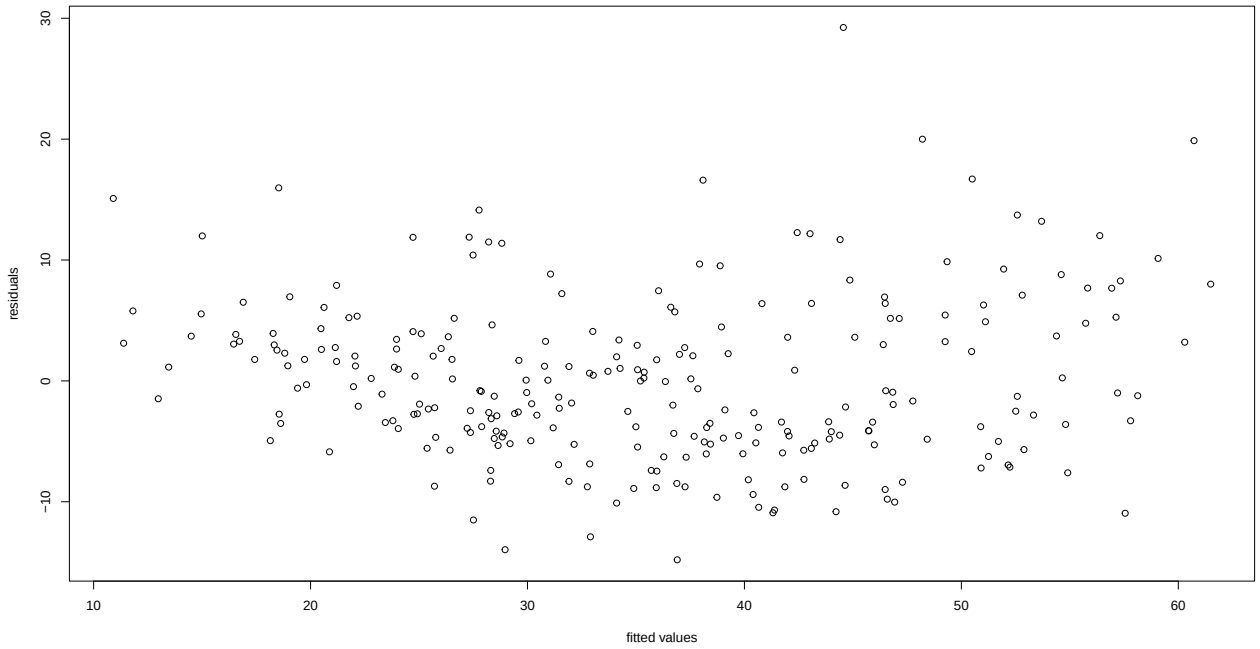
- age `a12age`
- height `a103ht`
- hiv `hiv`
- sex `a13sex`

Produce:

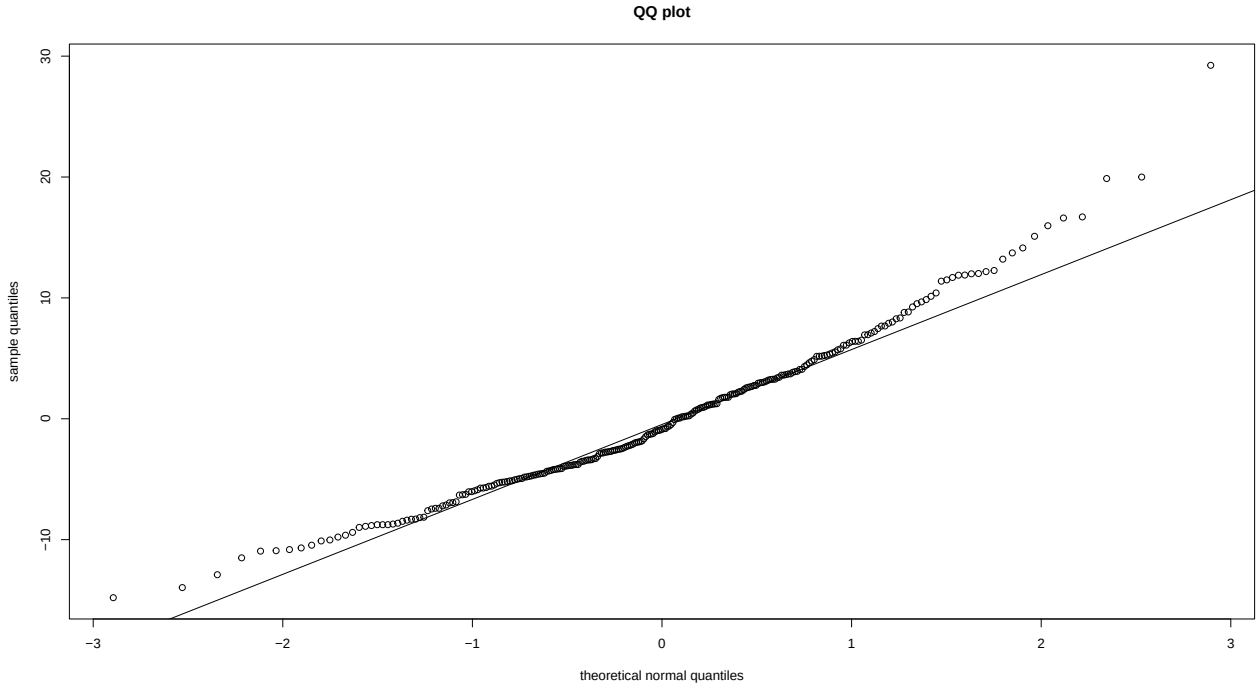
- a residuals vs. fitted values graph
- histogram of the residuals
- a QQ plot.

Exercise 9 (solution)

```
mod3 <- glm(a104wt ~ a12age + a103ht + as.factor(hiv) + as.factor(a13sex), data = ado)
summary(mod3)
##
## Call:
## glm(formula = a104wt ~ a12age + a103ht + as.factor(hiv) + as.factor(a13sex),
##     data = ado)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -14.8023   -4.6543   -0.8692    3.7079   29.2399
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -50.87129    4.84188 -10.507  < 2e-16 ***
## a12age                     1.32731    0.23284   5.700 3.26e-08 ***
## a103ht                     0.49929    0.04457  11.203  < 2e-16 ***
## as.factor(hiv)positive    -6.41914    0.90752  -7.073 1.42e-11 ***
## as.factor(a13sex)Male     -1.26779    0.84335  -1.503    0.134
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 45.36643)
##
##     Null deviance: 47549  on 262  degrees of freedom
## Residual deviance: 11705  on 258  degrees of freedom
##   (38 observations deleted due to missingness)
## AIC: 1756.6
##
## Number of Fisher Scoring iterations: 2

plot(predict(mod3, data = ado), residuals(mod3), xlab = "fitted values", ylab = "residuals")
```




```
qqnorm(residuals(mod3), xlab = "theoretical normal quantiles", ylab = "sample quantiles",
    main = "QQ plot")
qqline(residuals(mod3))
```



Exercise 10

You are given the following data:

$$\mathbf{x} = (-6, -6, -4, -1, 0.5, 2, 8, 8, 11, 11.5)^T$$

$$\mathbf{y} = (-3.7, -4.3, -3.9, -4.6, 0.5, -6.9, 10.2, 16.1, 6, 19.5)^T$$

a. Fit a linear regression model to these data and show the model output.

b. Describe the resulting regression line:

- What is the relationship between variables X and Y?
- How much (on average) does Y change when X changes by 1?
- What value does Y take (on average) when $X = 0$?

c. Compute the coefficient of determination R^2, the adjusted R^2, the likelihood and the AIC. Which of these tell you how good your model fits the data?

d. Compute the residuals $r_i = y_i - \hat{y}_i$ and do a normal distribution QQ plot.

e. What other diagnostic check(s) could you do? Do this and explain whether you think this is a good model.

f. Re-fit the model, but now including a term for X^2: $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon$. Check and discuss the resulting model and compare it to the previous one. Which model would you recommend for this dataset?

Exercise 10 (solution)

a.

Using R, we need to code up the data, then fit the model using either the `lm()` or the `glm()` function (`glm()` used below):

```
x<-c(-6,-6,-4,-1,0.5,2,8,8,11,11.5)
y<-c(-3.7,-4.3,-3.9,-4.6,0.5,-6.9,10.2,16.1,6,19.5)

mod<-glm(y~x)

summary(mod)
##
## Call:
## glm(formula = y ~ x)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -9.3238  -2.6891   0.7262   3.0511   6.6826
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.09255    1.87070   0.049  0.96175
## x            1.16560    0.27157   4.292  0.00264 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 30.74697)
```

```
##
##     Null deviance: 812.39  on 9  degrees of freedom
## Residual deviance: 245.98  on 8  degrees of freedom
## AIC: 66.405
##
## Number of Fisher Scoring iterations: 2
```

b.

- From the regression line, we conclude there is a *positive* relationship between X and Y since $\hat{\beta}_1 > 0$: as X increases, so Y increases. The p-value for testing the null hypothesis $h_0 : \beta_1 = 0$ against a two-sided alternative $H_1 : beta_1 \neq 0$ is low (0.0026), well below the usual statistical significance threshold of 0.05. We have therefore evidence that this positive relationship is real and not just due to random noise in the (finite) dataset.
- When X increases by 1, then Y increases - on average - by 1.17. Let $x_2 = x_1 + 1$, then:

$$\hat{y}_2 - \hat{y}_1 = 0.09 + 1.17x_2 - (0.09 + 1.17x_1) = 1.17(x_2 - x_1) = 1.17$$

- When $X = 0$, then, on average, we expect Y to be 0.09. Let $x = 0$, then:

$$\hat{y} = 0.09 + 1.17x = 0.09 + 1.17 * 0 = 0.09$$

c.

You can extract most of these quite conveniently from the R model object. `glm()` does not compute R^2, so you could just refit the model using `lm()` to get this. Likewise `lm()` will not compute a model likelihood (hence also no AIC), so you will need to get this from `glm()`. Recall that a simple linear regression does not make any distributional assumption and hence cannot yield a likelihood, only when you assume the errors/residuals to be normally distributed, i.e. by using `glm()`, will you be able to compute a likelihood. The log-likelihood can be extracted using the function `logLik()` on the GLM model object. To get the actual likelihood, just exponentiate.

```
modLm<-lm(y~x)
R2<-summary(modLm)$r.squared
R2adj<-summary(modLm)$adj.r.squared
likelihood<-exp(logLik(mod))
AIC<-mod$aic

print(paste(sep="","The coefficient of determination R2 is ",R2,"."))
## [1] "The coefficient of determination R2 is 0.697219249916526."
print(paste(sep="","The adjusted R2 is ",R2adj,"."))
## [1] "The adjusted R2 is 0.659371656156092."
print(paste(sep="","The model likelihood is ",likelihood,"."))
## [1] "The model likelihood is 7.64129162540487e-14."
print(paste(sep="","The AIC is ",AIC,"."))
## [1] "The AIC is 66.4052493042211."
```

You can however also calculate all of these manually (same results up to rounding errors):

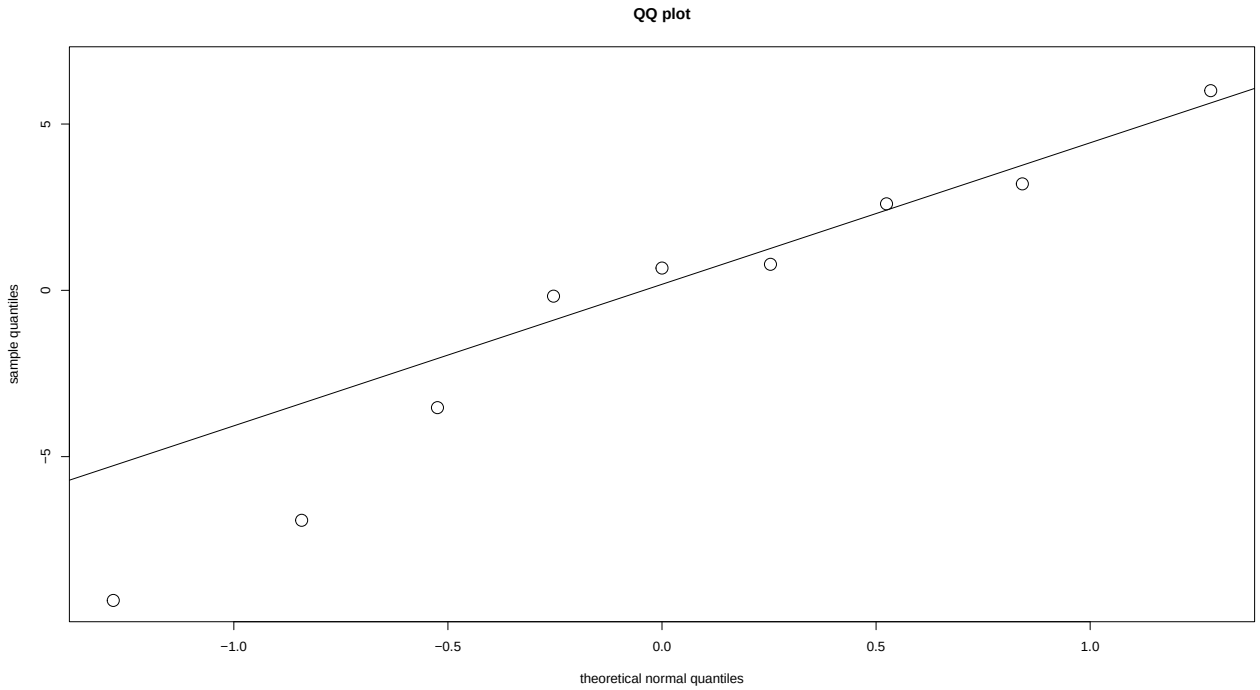
```
beta<-coef(mod)
r<-y-(beta[1]+beta[2]*x) # same as y<-resid(mod)

R2<-beta[2]^2*sum((x-mean(x))^2)/sum((y-mean(y))^2)
R2adj<-1-(1-R2)*(length(x)-1)/(length(x)-1-1)
```

```
    # p in the lecture notes is the number of predictors, not number of parameters
likelihood<-prod(dnorm(r,sd=sd(r)))
AIC<-(-2*log(likelihood))+2*(2+1)

print(paste(sep="","The coefficient of determination R2 is ",R2,"."))
## [1] "The coefficient of determination R2 is 0.697219249916526."
print(paste(sep="","The adjusted R2 is ",R2adj,"."))
## [1] "The adjusted R2 is 0.659371656156092."
print(paste(sep="","The model likelihood is ",likelihood,"."))
## [1] "The model likelihood is 7.43920561909125e-14."
print(paste(sep="","The AIC is ",AIC,"."))
## [1] "The AIC is 66.4588544607993."
```

d.

The individual residuals are listed in the table / calculation worksheet given under a. above. We can easily produce a QQ plot:

```
r<-resid(mod) # had already computed this manually above; just redoing it here
theoQ<-qnorm(order(order(r))/length(r))

plot(theoQ,r,
     xlab="theoretical normal quantiles",
     ylab="sample quantiles",
     main="QQ plot", cex=2)
qqline(r) # just adds the line
```



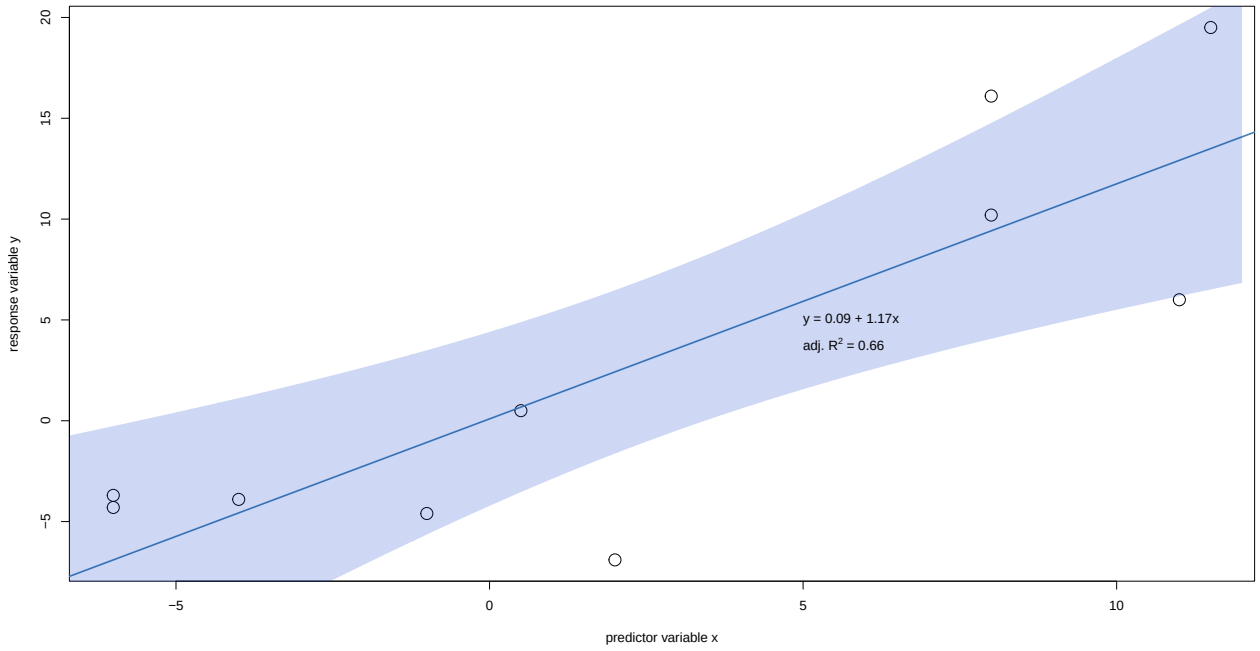
With only 10 data points, it is a bit difficult to interpret this. Overall it looks OK, but perhaps some deviation at the lower end - this could indicate that the normality of residuals assumption is not fully met.

e.

Goodness of fit

We have not yet actually looked at the actual model fit. This should be the first check you do: does the model seem to fit?

```
xx<-seq(-12,12,length=500)
fit<-as.data.frame(predict(modLm,newdata=data.frame(x=xx),interval="confidence"))

plot(x,y,cex=2,xlab="predictor variable x",ylab="response variable y")
abline(a=beta[1],b=beta[2],col="steelblue",lwd=2)
polygon(c(xx,xx[length(xx):1]),c(fit$lwr,fit$upr[length(xx):1]),
        border=NA,col=rgb(0,50,200,alpha=50,maxColorValue=255))
text(x=5,y=5,adj=0,
     labels=paste(sep="","y = ",round(beta[1],digits=2)," + ",round(beta[2],digits=2),"x"))
text(x=5,y=3.75,adj=0,
     labels=substitute(paste("adj. ",R^2," = ",R2adj),list(R2adj=round(R2adj,digits=2))))
```

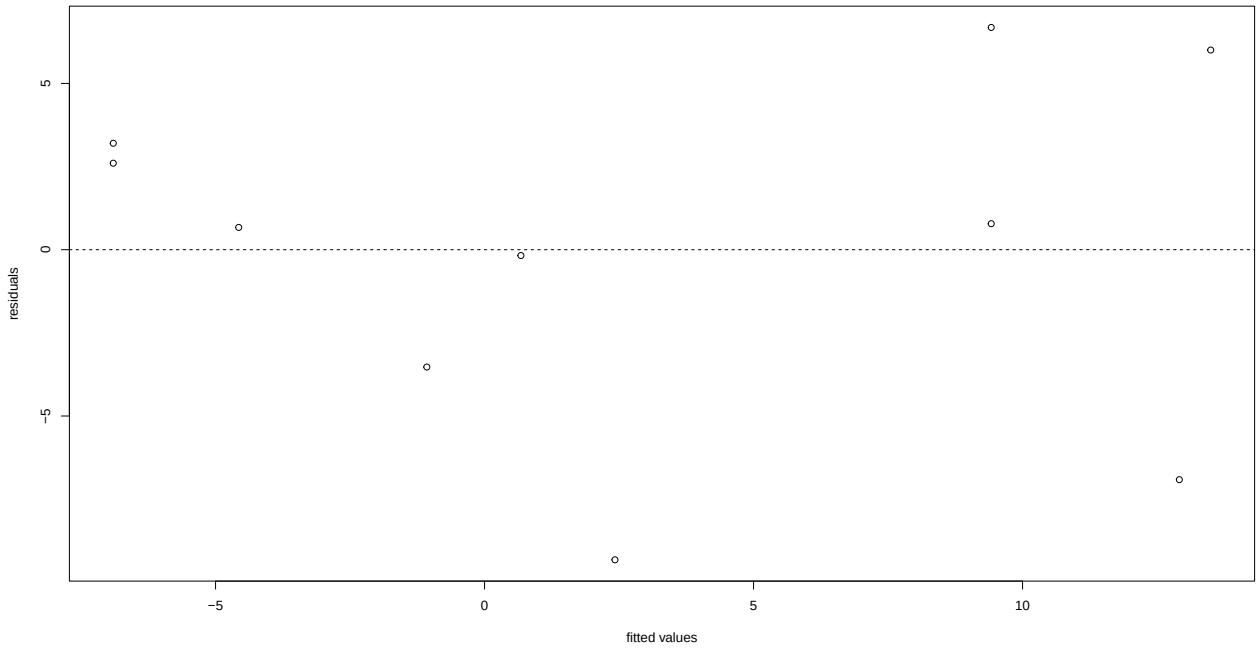


With so few data points, it's difficult to say much. Overall the line seems a reasonable fit, but you could argue that from $X = -5$, to $X = 5$, a better fit would be just a flat line, to be followed by a steeper increase in Y with X for $X > 5$.

Residuals vs. fitted values

Next, we can plot residuals against fitted values and check if the residuals look to be randomly distributed, are homoscedastic and that there are no obvious outliers.

```
yhat<-beta[1]+beta[2]*x # same as predict(mod)

plot(yhat,r,xlab="fitted values",ylab="residuals")
abline(h=0,lty=2)
```

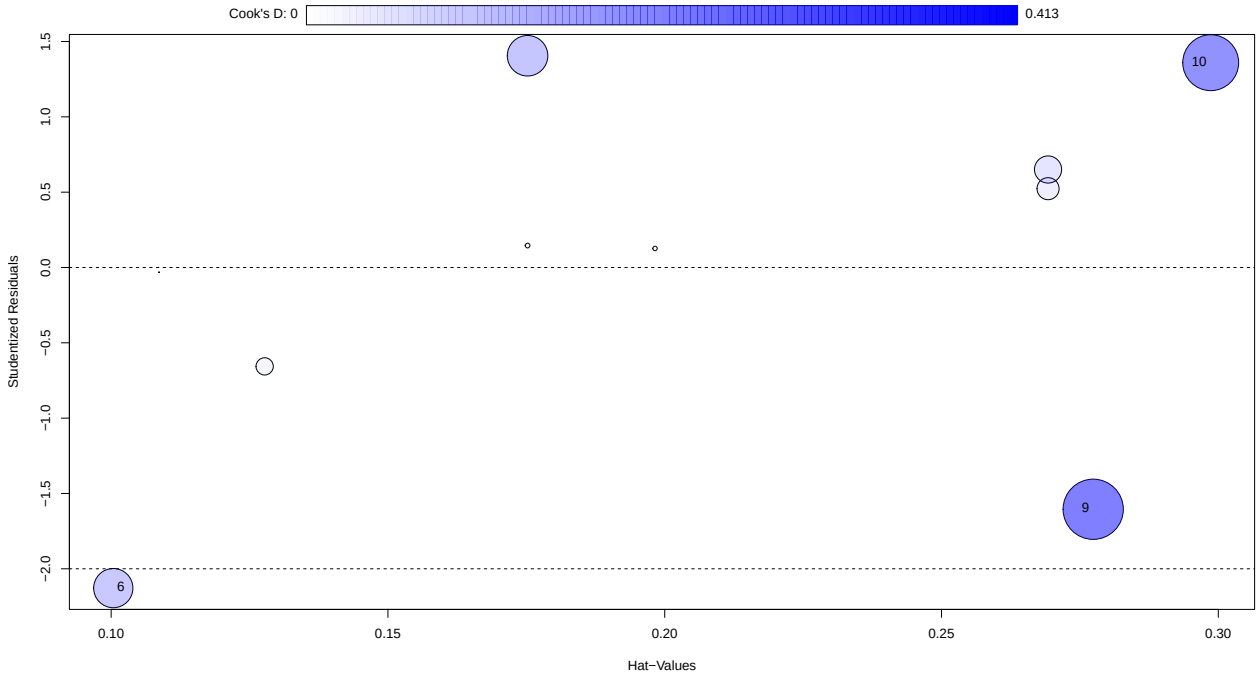


Too little data points to say anything definite, but it seems there is a wider spread around 0 for larger predicted values.

Influential observations

Finally we can check for outliers & influential observations.

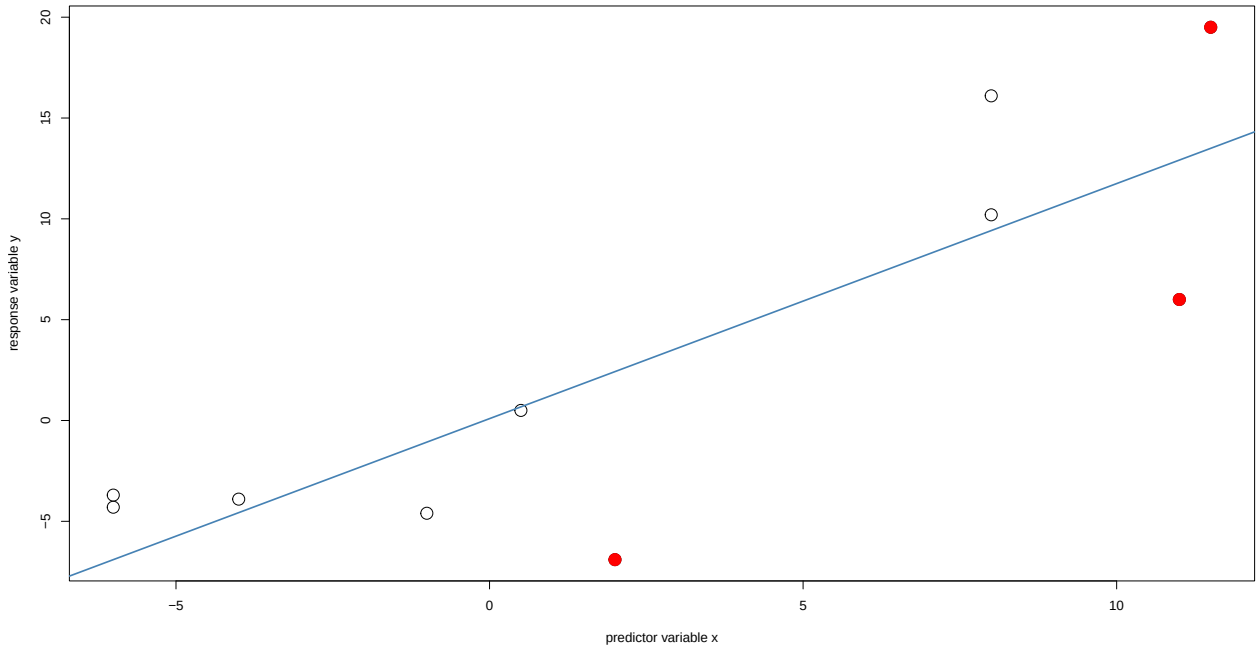
```
require(car)
influencePlot(mod)
```




```
##      StudRes       Hat     CookD
## 6  -2.128233 0.1003838 0.1753472
## 9  -1.604823 0.2774047 0.4130223
## 10  1.359498 0.2986328 0.3557561
```

This suggests 3 potentially influential observations: observations number 6, 9 and 10. We highlight them in red below.

```
plot(x,y,cex=2,xlab="predictor variable x",ylab="response variable y")
points(x[c(6,9,10)],y[c(6,9,10)],pch=19,cex=2,col="red")
abline(a=beta[1],b=beta[2],col="steelblue",lwd=2)
```



f.

Let's fit the model with a term for X^2:

```
df <- data.frame(y = y, x = x, x2 = x^2)
mod2 <- glm(y ~ x + x2, data = df)

summary(mod2)
##
## Call:
## glm(formula = y ~ x + x2, data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -8.8650  -1.0886   0.6105   2.2989   7.6625
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.38487    2.70194  -0.883   0.4067
## x            0.77845    0.40903   1.903   0.0987 .
## x2           0.07179    0.05808   1.236   0.2563
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 28.84427)
##
##     Null deviance: 812.39  on 9  degrees of freedom
```

```
## Residual deviance: 201.91  on 7  degrees of freedom
## AIC: 66.431
##
## Number of Fisher Scoring iterations: 2
```

We note that none of the regression coefficients for X or X^2 are statistically significant anymore (using $p < 0.05$ as threshold).

Goodness of fit

Let's compute goodness of fit metrics:

```
mod2Lm<-lm(y~x+x2,data=df)
R2<-summary(mod2Lm)$r.squared
R2adj<-summary(mod2Lm)$adj.r.squared
likelihood<-exp(logLik(mod2))
AIC<-mod2$aic

print(paste(sep="","The coefficient of determination R2 is ",R2,"."))
## [1] "The coefficient of determination R2 is 0.751461526840908."
print(paste(sep="","The adjusted R2 is ",R2adj,"."))
## [1] "The adjusted R2 is 0.680450534509739."
print(paste(sep="","The model likelihood is ",likelihood,"."))
## [1] "The model likelihood is 2.05040639246456e-13."
print(paste(sep="","The AIC is ",AIC,"."))
## [1] "The AIC is 66.4311363903853."
```

We note that according to R^2, the new model explains slightly more of the variation in the dataset (75% vs 70% on standard R^2, 68% vs. 66% on adjusted R^2).

The likelihood is also slightly better (larger) but the AICs are virtually the same, with the model without the X^2 term having a marginalyl better (lower) AIC.

From this we conclude, that both models have similarly good fit, and in such a case we would usually prefer the more parsimonious (simpler) model. We would therefore prefer the model without the X^2 term.

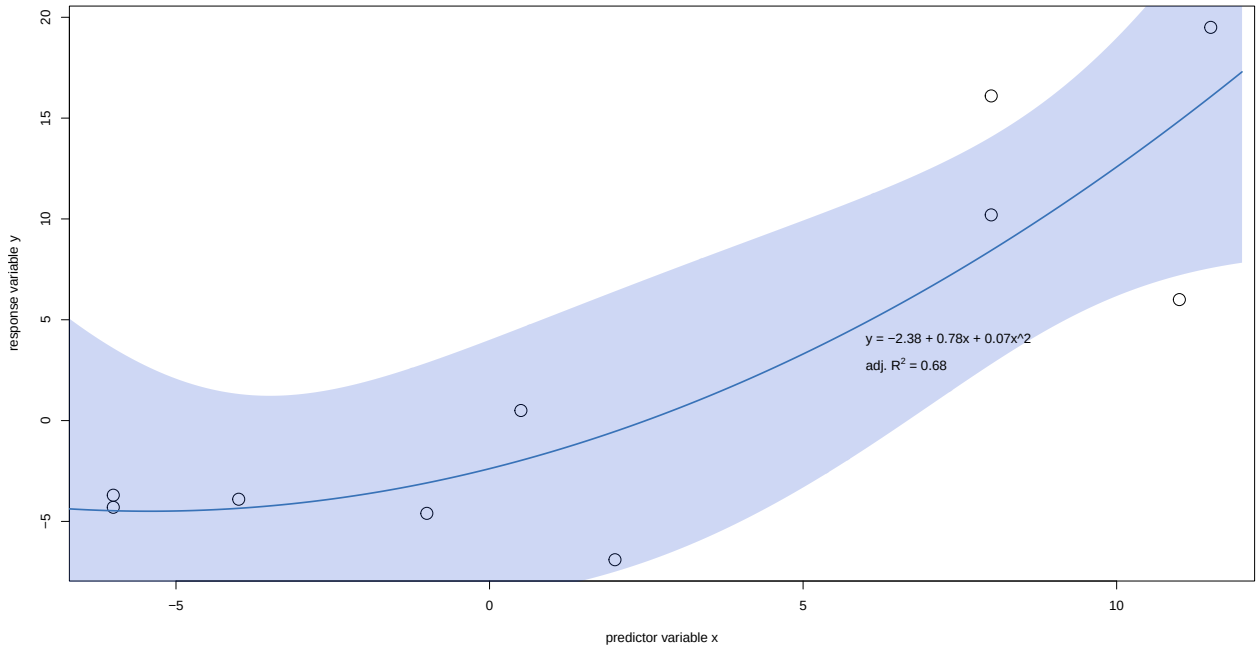
We can probe this further, byt comparing both models (which are an example of nested models) by using a likelihood ratio test. This confirms (p=0.16) that the inclusion of the X^2 term does not significantly improve model fit.

```
library(lmtest)
lrtest(mod2, mod)
## Likelihood ratio test
##
## Model 1: y ~ x + x2
## Model 2: y ~ x
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   4 -29.216
## 2   3 -30.203 -1 1.9741       0.16
```

Looking at model diagnostics for the second model, we can start by inspecting the model fit visually (we already computed various goodness-of-fit metrics above).

```
beta2<-round(digits=2,coef(mod2))
fit2<-as.data.frame(predict(mod2Lm,newdata=data.frame(x=xx,x2=xx^2),interval="confidence"))

plot(x,y,cex=2,xlab="predictor variable x",ylab="response variable y")
lines(xx,fit2$fit,col="steelblue",lwd=2)
```

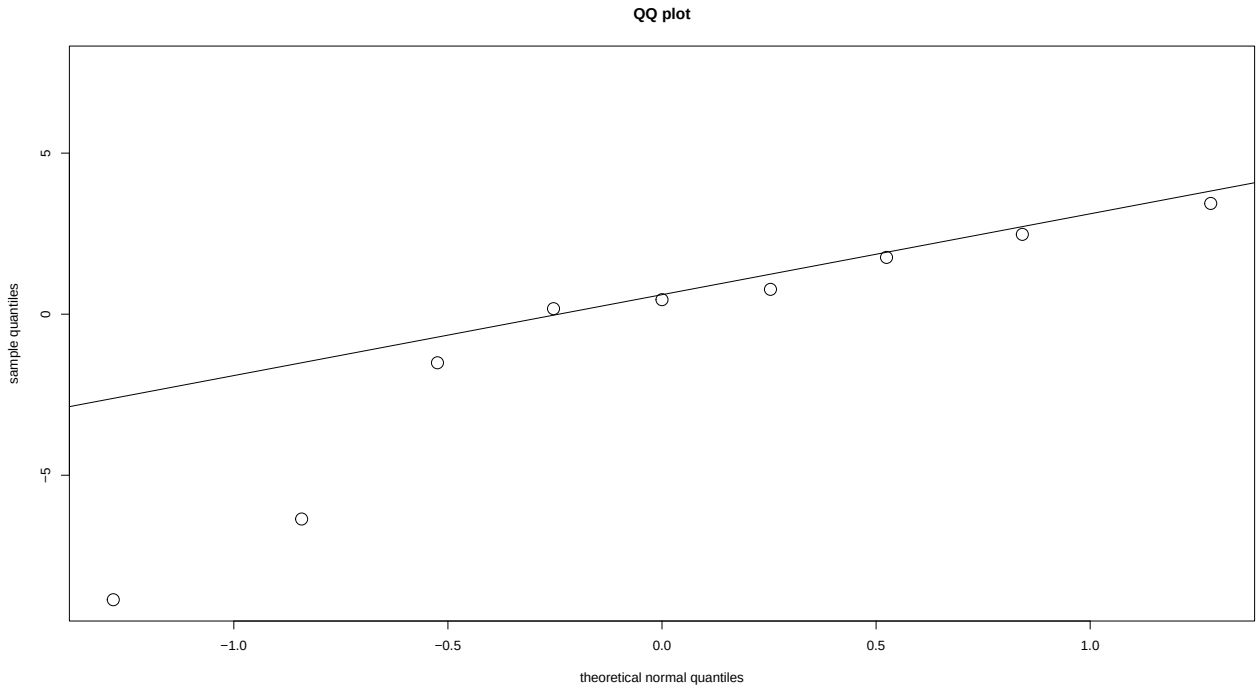
```
polygon(c(xx,xx[length(xx):1]),c(fit2$lwr,fit2$upr[length(xx):1]),
        border=NA,col=rgb(0,50,200,alpha=50,maxColorValue=255))
text(x=6,y=4,adj=0,
     labels=substitute(paste("y = ",b1," + ",b2,"x + ",b3,"x^2"),list(b1=beta2[1],b2=beta2[2],b3=beta2[
text(x=6,y=2.75,adj=0,
     labels=substitute(paste("adj. ",R^2," = ",R2adj),list(R2adj=round(R2adj,digits=2))))
```



As previously, difficult to make definite statements with so few datapoints, but compared to earlier this model seems to slightly better capture the initial flat relationship between X and Y, followed by a positive increase at larger X values.

QQ plot

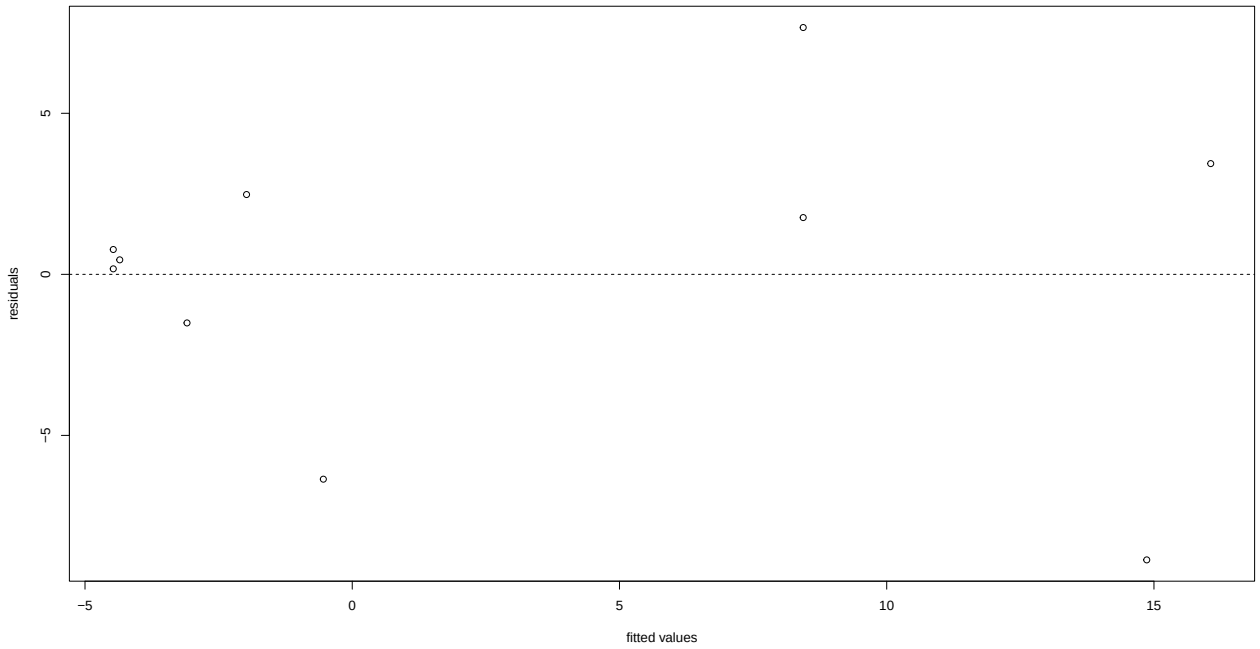
```
r2<-resid(mod2)
theoQ2<-qnorm(order(order(r2))/length(r2))

plot(theoQ2,r2,
     xlab="theoretical normal quantiles",
     ylab="sample quantiles",
     main="QQ plot", cex=2)
qqline(r2) # just adds the line
```



The residuals hug the diagonal even more closely at the upper end, but this just makes the departure from the line at the lower end more marked.

Residuals against predicted values

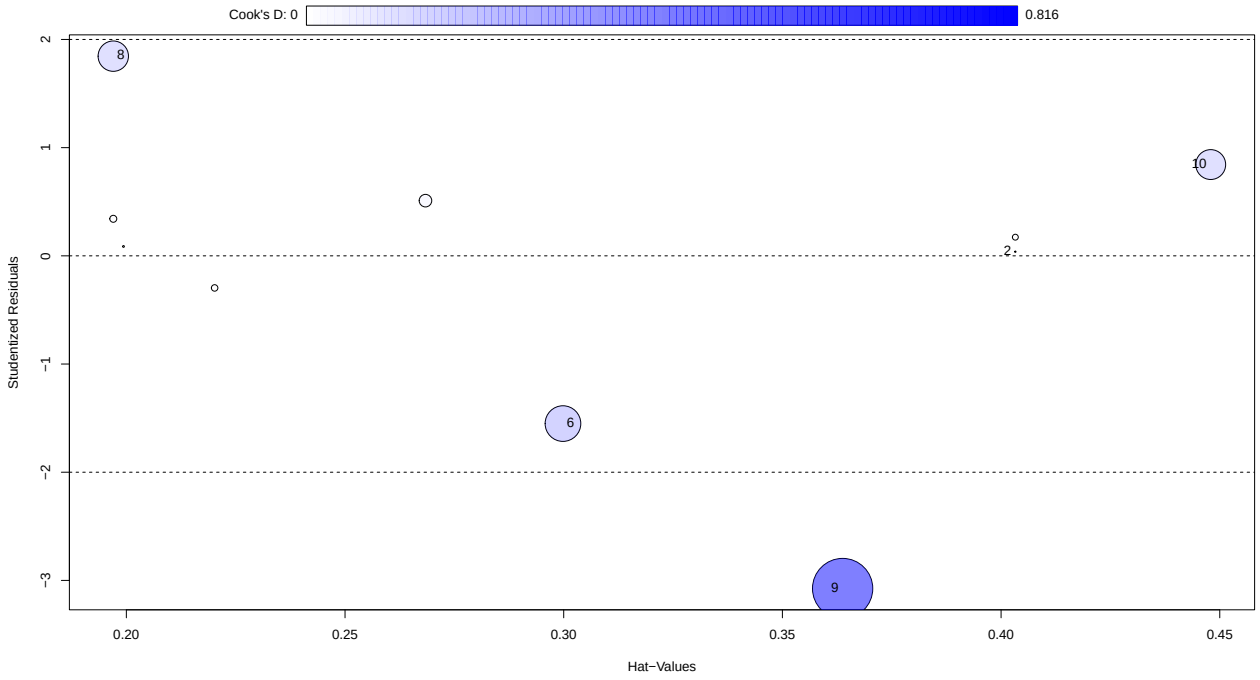
```
yhat2<-predict(mod2)

plot(yhat2,r2,xlab="fitted values",ylab="residuals")
abline(h=0,lty=2)
```



This looks fairly similar to the former model: overall OK, but perhaps more spread around 0 at larger fitted values.

Influential observations

```
influencePlot(mod2)
```




```
##         StudRes       Hat        CookD
## 2    0.03815863 0.4032611 0.0003825671
## 6   -1.55046605 0.2998273 0.2858143733
## 8    1.84565972 0.1970122 0.2073184661
## 9   -3.07488307 0.3637716 0.8161750794
## 10   0.84363573 0.4479316 0.2007574248
```

In addition to the same 3 potentially observations, there is now a fourth one: observation 8.

As an overall conclusion, we would recommend the simpler model $Y = \beta_0 + \beta_1 X + \epsilon$ to the model with a term for X^2. Both models fit the data similarly well. One could argue that the more complex model better captures the real relationship between X and Y (steeper increase in Y as X gets larger), but there are too few data points to confirm this. Other model diagnostics are quite similar for both models. Sticking to the principle of parsimony, we recommend the simpler model.