regression.chpt5

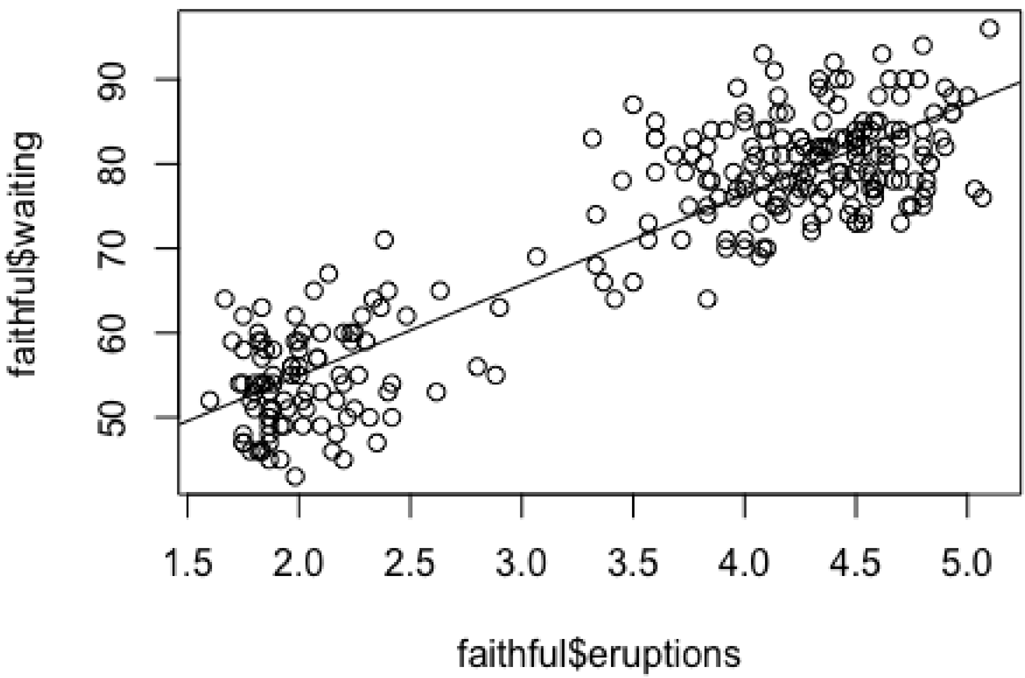
(Note that this was edited post knitting to help with formatting.)

fitting a line to the data

A line is determined by _________ and ___________:

$$y = b + m * x$$

```
plot(faithful$eruptions,faithful$waiting) # 1
cor(faithful$eruptions,faithful$waiting)

## [1] 0.9008112

w=lm(faithful$waiting~faithful$eruptions)  #given the linear model or best
fitting line for waiting as a function of eruptions
#gives the y intercept and then the slope
#linear model
#NOTE the difference in order from plot( ____, _____) # in (1)
#However one can also use
#plot(waiting~eruptions,data=faithful)
abline(w)#adds the the best fitting line
```



Let's see what this variable w holds: (lots of stuff!)

```
str(w)

## List of 12
##  $ coefficients : Named num [1:2] 33.5 10.7
##   ..- attr(*, "names")= chr [1:2] "(Intercept)" "faithful$eruptions"
##  $ residuals    : Named num [1:272] 6.9 1.21 4.76 4.03 2.89 ...
##   ..- attr(*, "names")= chr [1:272] "1" "2" "3" "4" ...
##  $ effects      : Named num [1:272] -1169.26 201.6 4.36 3.59 2.54 ...
##   ..- attr(*, "names")= chr [1:272] "(Intercept)" "faithful$eruptions" ""
"" ...
##  $ rank         : int 2
##  $ fitted.values: Named num [1:272] 72.1 52.8 69.2 58 82.1 ...
##   ..- attr(*, "names")= chr [1:272] "1" "2" "3" "4" ...
##  $ assign       : int [1:2] 0 1
##  $ qr           :List of 5
##   ..$ qr   : num [1:272, 1:2] -16.4924 0.0606 0.0606 0.0606 0.0606 ...
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : chr [1:272] "1" "2" "3" "4" ...
##   .. .. ..$ : chr [1:2] "(Intercept)" "faithful$eruptions"
##   .. ..- attr(*, "assign")= int [1:2] 0 1
##   ..$ qraux: num [1:2] 1.06 1.09
```

```
##   ..$ pivot: int [1:2] 1 2
##   ..$ tol  : num 1e-07
##   ..$ rank : int 2
##   ..- attr(*, "class")= chr "qr"
##  $ df.residual  : int 270
##  $ xlevels      : Named list()
##  $ call         : language lm(formula = faithful$waiting ~
faithful$eruptions)
##  $ terms        :Classes 'terms', 'formula'  language faithful$waiting ~
faithful$eruptions
##   .. ..- attr(*, "variables")= language list(faithful$waiting,
faithful$eruptions)
##   .. ..- attr(*, "factors")= int [1:2, 1] 0 1
##   .. .. ..- attr(*, "dimnames")=List of 2
##   .. .. .. ..$ : chr [1:2] "faithful$waiting" "faithful$eruptions"
##   .. .. .. ..$ : chr "faithful$eruptions"
##   .. ..- attr(*, "term.labels")= chr "faithful$eruptions"
##   .. ..- attr(*, "order")= int 1
##   .. ..- attr(*, "intercept")= int 1
##   .. ..- attr(*, "response")= int 1
##   .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
##   .. ..- attr(*, "predvars")= language list(faithful$waiting,
faithful$eruptions)
##   .. ..- attr(*, "dataClasses")= Named chr [1:2] "numeric" "numeric"
##   .. .. ..- attr(*, "names")= chr [1:2] "faithful$waiting"
"faithful$eruptions"
##  $ model        :'data.frame':   272 obs. of  2 variables:
##   ..$ faithful$waiting  : num [1:272] 79 54 74 62 85 55 88 85 51 85 ...
##   ..$ faithful$eruptions: num [1:272] 3.6 1.8 3.33 2.28 4.53 ...
##   ..- attr(*, "terms")=Classes 'terms', 'formula'  language
faithful$waiting ~ faithful$eruptions
##   .. .. ..- attr(*, "variables")= language list(faithful$waiting,
faithful$eruptions)
##   .. .. ..- attr(*, "factors")= int [1:2, 1] 0 1
##   .. .. .. ..- attr(*, "dimnames")=List of 2
##   .. .. .. .. ..$ : chr [1:2] "faithful$waiting" "faithful$eruptions"
##   .. .. .. .. ..$ : chr "faithful$eruptions"
##   .. .. ..- attr(*, "term.labels")= chr "faithful$eruptions"
##   .. .. ..- attr(*, "order")= int 1
##   .. .. ..- attr(*, "intercept")= int 1
##   .. .. ..- attr(*, "response")= int 1
##   .. .. ..- attr(*, ".Environment")=<environment: R_GlobalEnv>
##   .. .. ..- attr(*, "predvars")= language list(faithful$waiting,
faithful$eruptions)
##   .. .. ..- attr(*, "dataClasses")= Named chr [1:2] "numeric" "numeric"
##   .. .. .. ..- attr(*, "names")= chr [1:2] "faithful$waiting"
"faithful$eruptions"
##  - attr(*, "class")= chr "lm"
```

The coefficients are the y-intercept and the slope:

```r
#w$coefficients
b=w$coefficients[1]  #the first element of coefficients in the linear model
is the y intercept
m=w$coefficients[2]  #the second is the slope
b

## (Intercept)
##     33.4744

m

## faithful$eruptions
##           10.72964
```

EXERCIZE: compute these by hand from the formula:

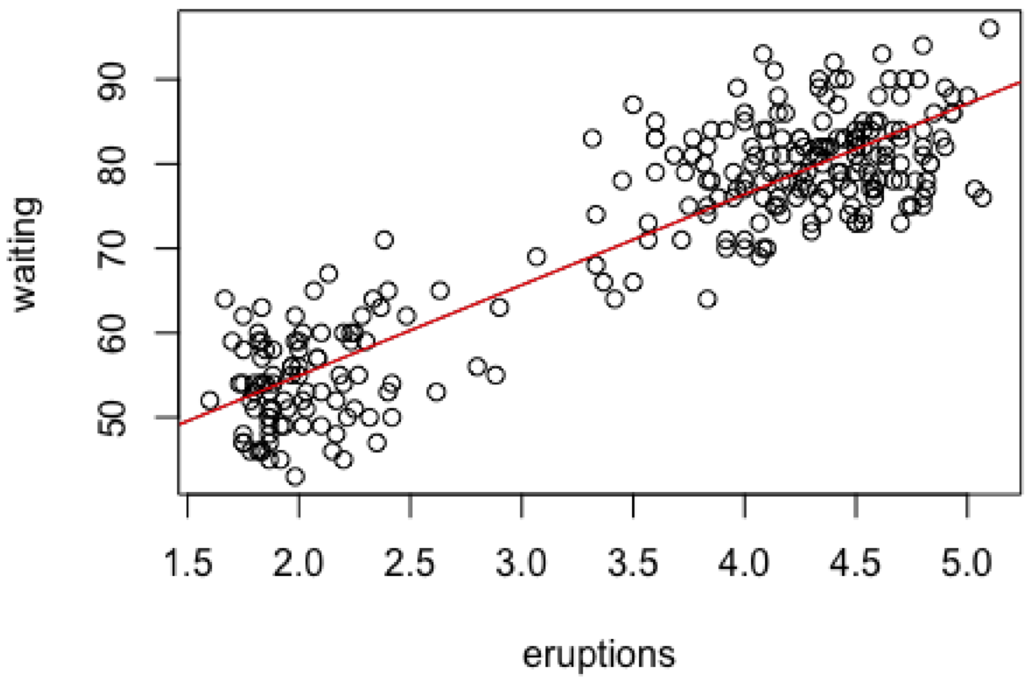
We do this below using R. $m1 = r * s_y / s_x$

$b1={y}-b1*{x} $ (Note that this doesn't look right in word)

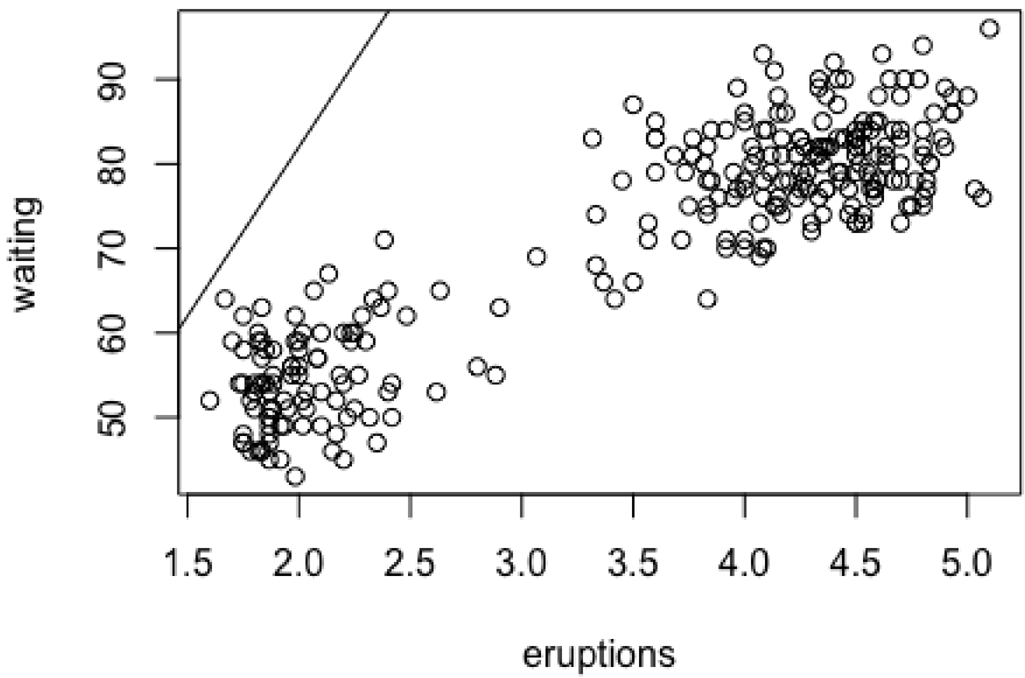
```r
m1=cor(faithful$eruptions,faithful$waiting)*sd(faithful$waiting)/sd(faithful$
eruptions)
m1

## [1] 10.72964

b1=mean(faithful$waiting)-m1*mean(faithful$eruptions)
b1

## [1] 33.4744
```

Now add the regression line to the scatter plot:

```r
plot(waiting~eruptions,data=faithful)
abline(w)   #adds a line with above y intercept and slope
abline(b1,m1,col='red') #another way to add a line is to give y intercept and
then the slope
```

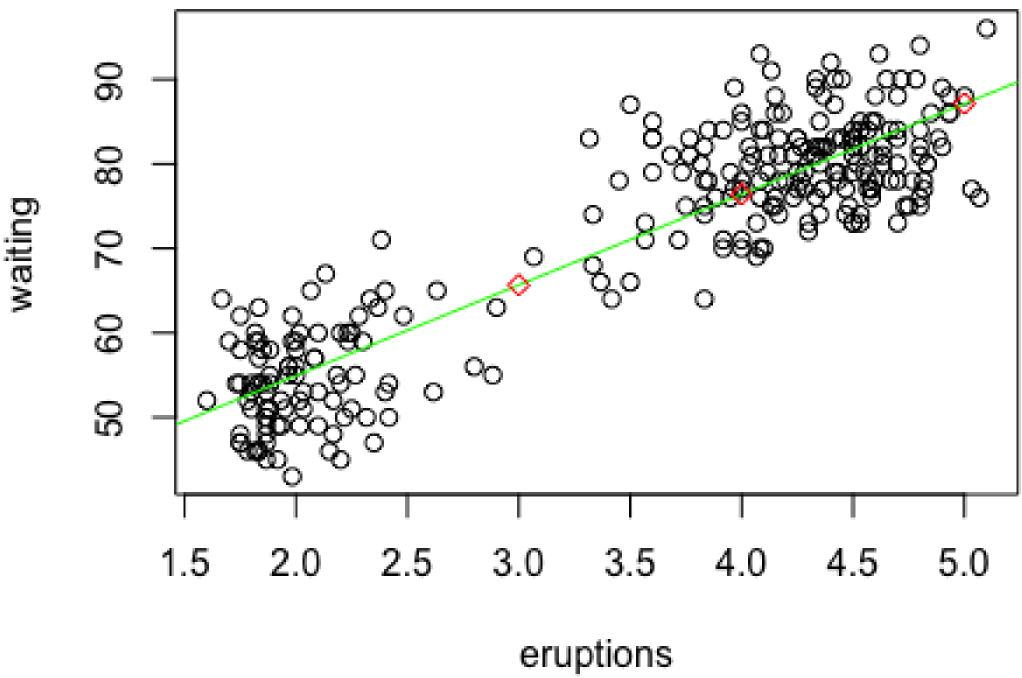



```
plot(waiting~eruptions,data=faithful)
abline(2,40) #well that's not the right line
```



What are the values predicted by the regression line for eruptions of length 3, 4 and 5?

```
u=c(3,4,5)

v=b+m*u #creates a vector of the resulting three values on the regression
line
v

## [1] 65.66332 76.39296 87.12260

plot(waiting~eruptions,data=faithful)
abline(w,col='green')
points(u,v,pch=23  ,col='red')   #add those predicted values to graph
```



Our variable waiting has total variation:

```
sd(faithful$waiting)^2

## [1] 184.8233
```

Part of this variation measures the variation of $\widehat{waiting}$ (In general, for data x and y with a least squares line the predicted value is denoted $\hat{y}$). Let's compute the proportion of the total variation in y given by the variation in $\hat{y}$ for waiting. First the variation in $\widehat{waiting}$.

```
sd(w$coefficients[1]+w$coefficients[2]*faithful$eruptions)^2

## [1] 149.9769
```

Next the proportion of the total variation in waiting given by the variation in $\widehat{waiting}$.

```
sd(w$coefficients[1]+w$coefficients[2]*faithful$eruptions)^2/sd(faithful$wait
ing)^2

## [1] 0.8114608

cor(faithful$waiting,faithful$eruptions)^2
```

```
## [1] 0.8114608
```

A conjecture?

$r^2 = (predicted\ variation\ in\ y\ from\ x)/(variation\ in\ y)$ ________ Now we will discuss the question of how good is our linear model. So we are asking how far it is from our actual data. So for each value x of eruptions we want to compute the difference between the waiting for that x and the value predicted by the regression line for that x. Let's look at one data point:

```
faithful$eruptions[5]

## [1] 4.533

faithful$waiting[5]

## [1] 85
```

The value predicted by the linear model is

```
w$coefficients[1]+w$coefficients[2]*faithful$eruptions[5] #remember w is from
the linear model

## (Intercept)
##    82.11186
```

Plot this on the graph

```
#difference between 5th data point
faithful$waiting[5]-(w$coefficients[1]+w$coefficients[2]*faithful$eruptions[5
])

## (Intercept)
##    2.888139
```

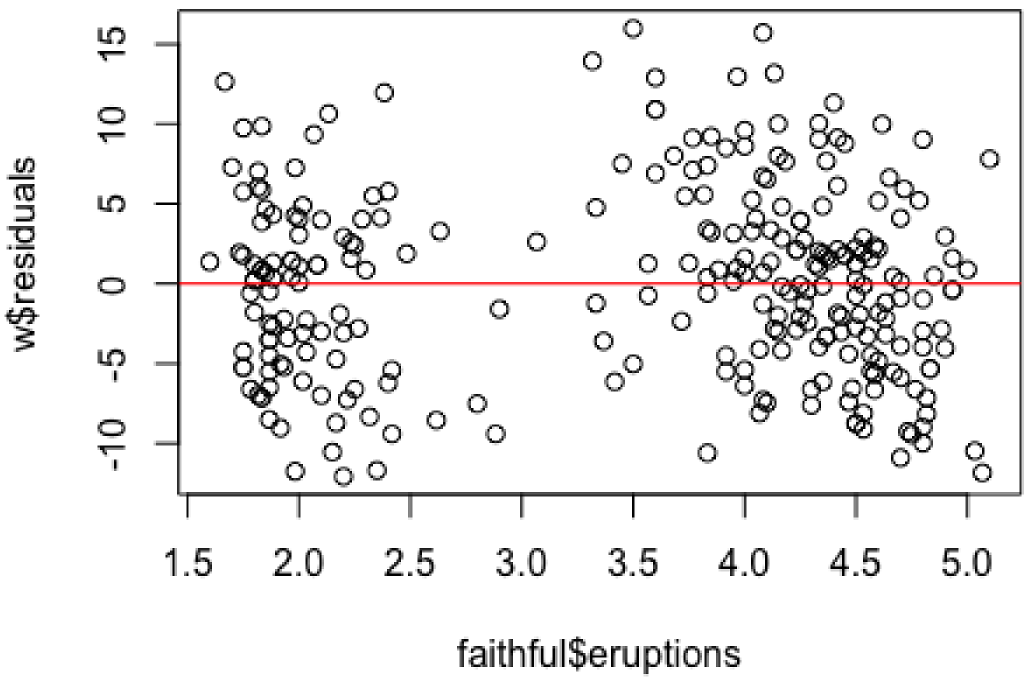
The differences between a data point and our line is called a **residual.**

residual = observed - predicted = the difference between the actual plotted point to the point on the regression line

The formula for the regression line is to minimize the sum of the squares of these differences.

Plot all the residues for our faithful data and the line y=0

```
plot(faithful$eruptions,w$residuals)
abline(h=0,col='red')# compare this graph to that of the regression line
```

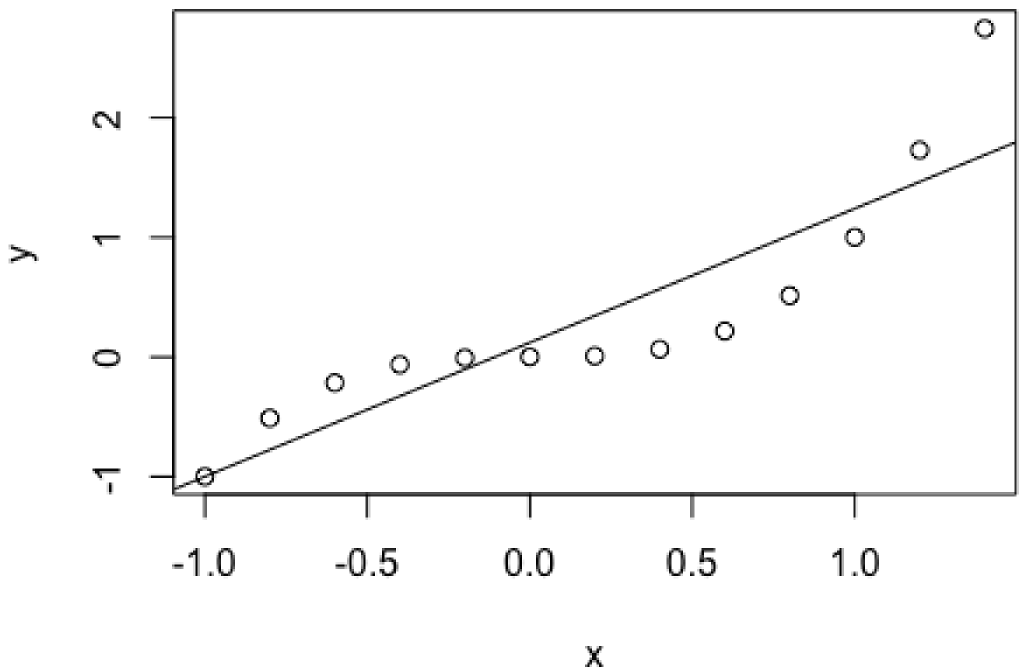


Note the range for the residuals. We see the residuals balance about y=0. The mean of the residuals is zero. One can check this mathematically-but we won't. Check this with R code:

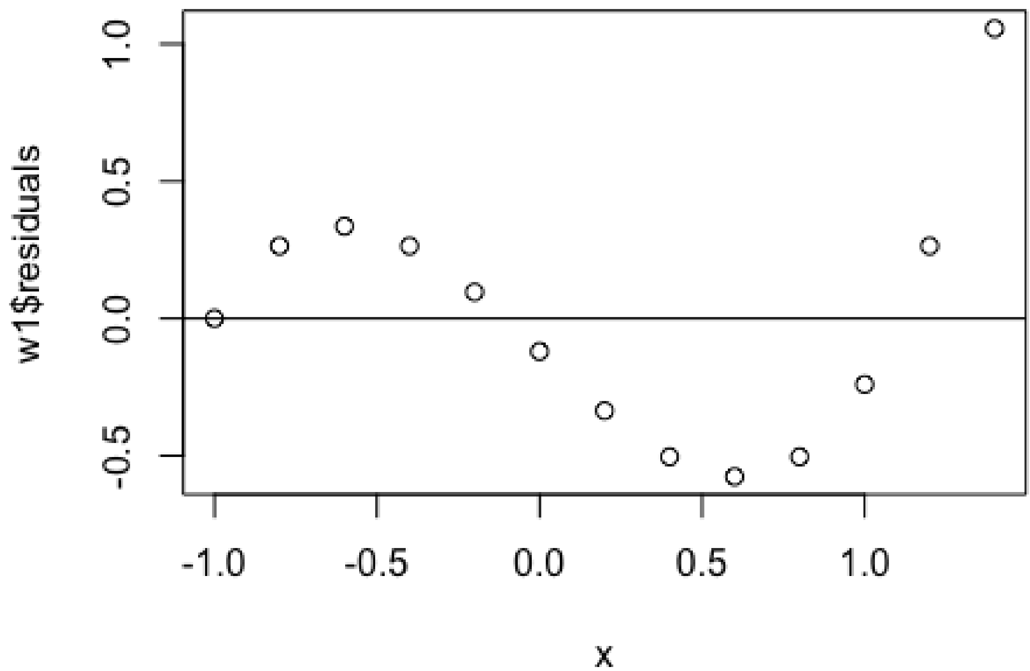
```
mean(w$residuals)

## [1] 1.387779e-16
```

An example where the residual plot is important to give us further information about the relation between the two variables.If the relation is linear we should see a random pattern in the residuals. We are only studying linear regression between two variables. There are other types of regression.

```
x<-seq(-1,1.5,0.2)
y<-x^3
cor(x,y) #Note this is pretty strong

## [1] 0.8871503

w1=lm(y~x)
plot(x,y)
abline(w1)
```

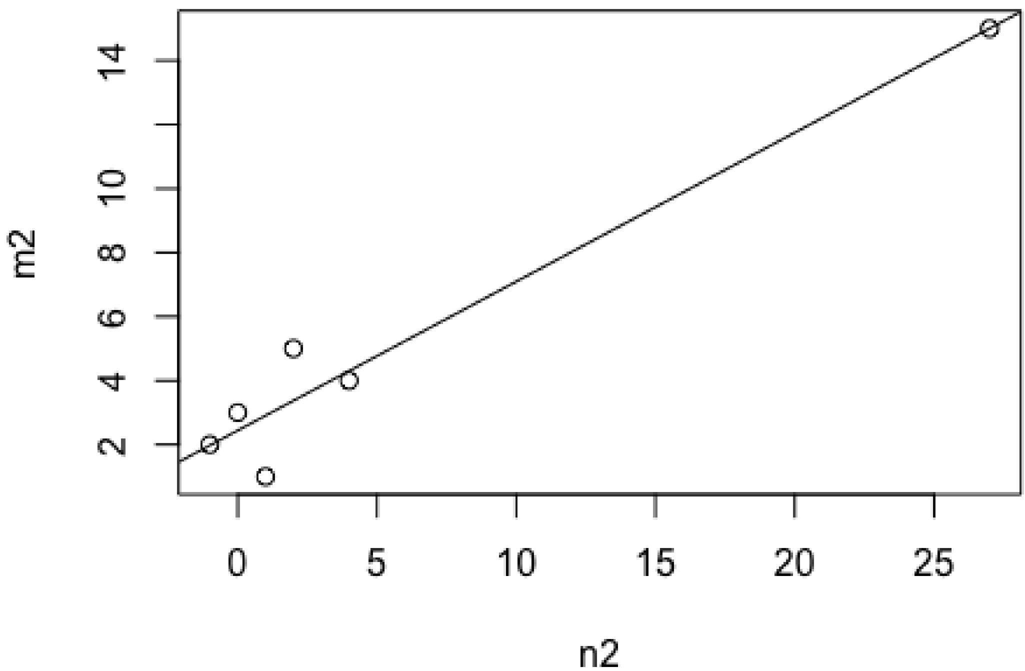



```
plot(x,w1$residuals)
abline(h=0)
```

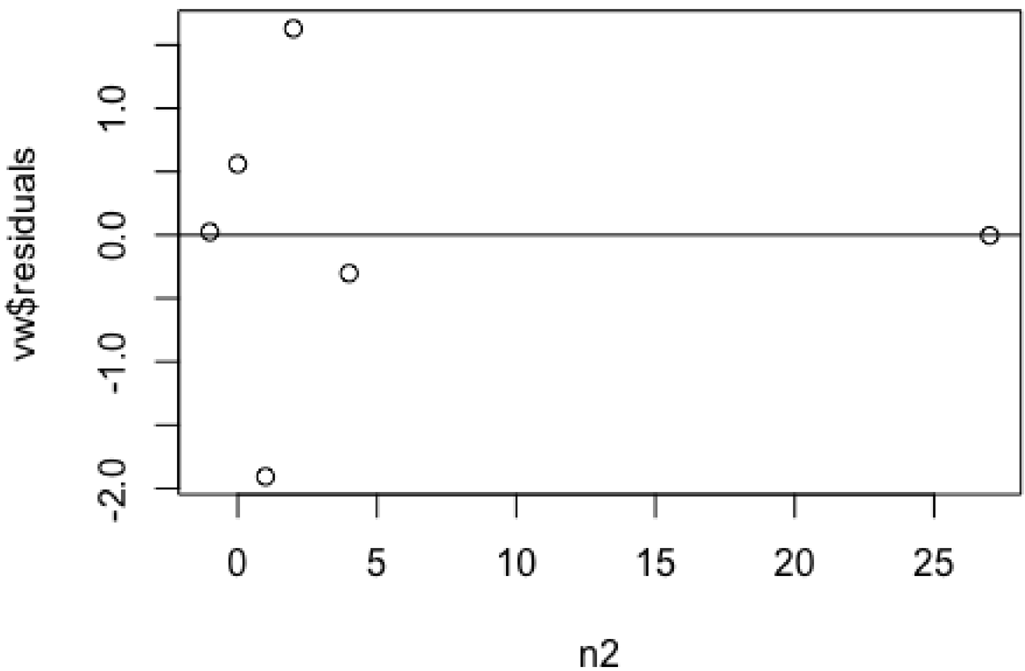


Influential observation-changes the calculations

```
m2=c(1,2,3,4,5,15)
 n2=c(1,-1,0,4,2,27)
 cor(n2,m2)

## [1] 0.9739291

 plot(n2,m2)
 vw=lm(m2~n2)
 vw

##
## Call:
## lm(formula = m2 ~ n2)
##
## Coefficients:
## (Intercept)           n2
##      2.4407       0.4653

 abline(vw)
```




```
plot(n2,vw$residuals)
abline(h=0)
```

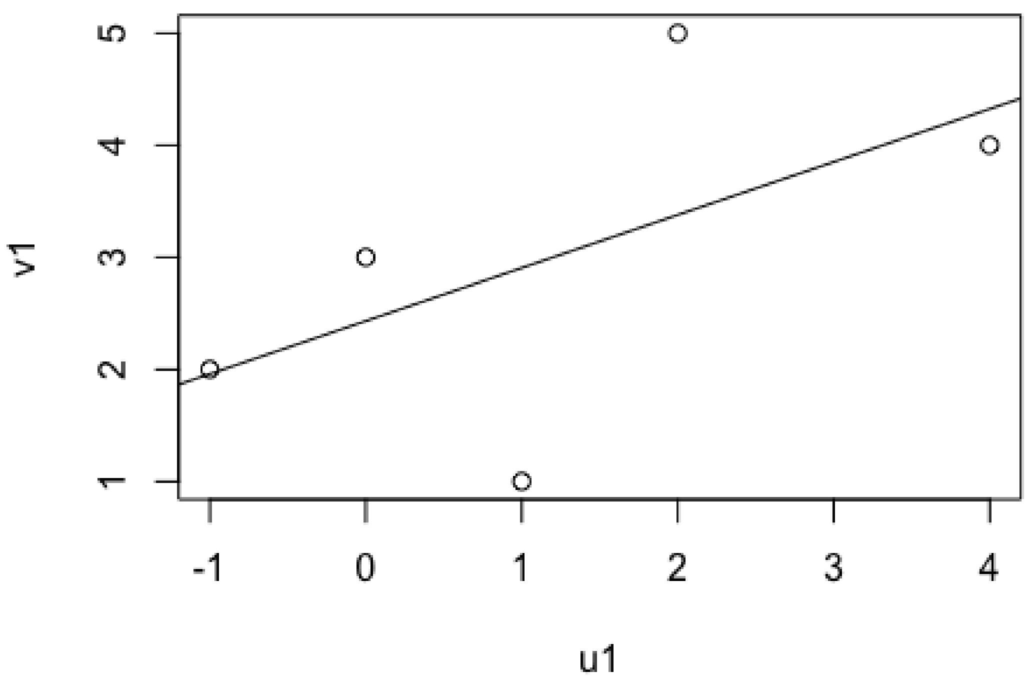


Getting rid of an influential observation

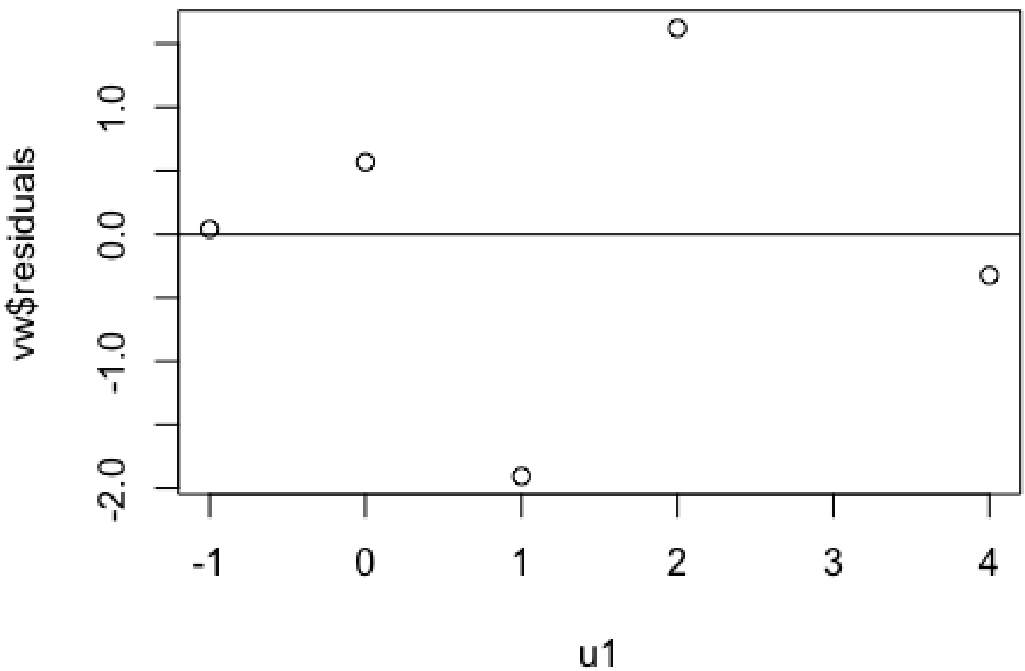
```
u1=n2[-6]  #remember that a -6 in the index says use everything but the sixth
entry
v1=m2[-6]
u1

## [1]  1 -1  0  4  2

v1

## [1] 1 2 3 4 5

cor(u1,v1)

## [1] 0.5753965

 plot(u1,v1)
 vw=lm(v1~u1)
 vw

##
## Call:
## lm(formula = v1 ~ u1)
```

```
##
## Coefficients:
## (Intercept)           u1
##       2.432        0.473
```

```
abline(vw)
```



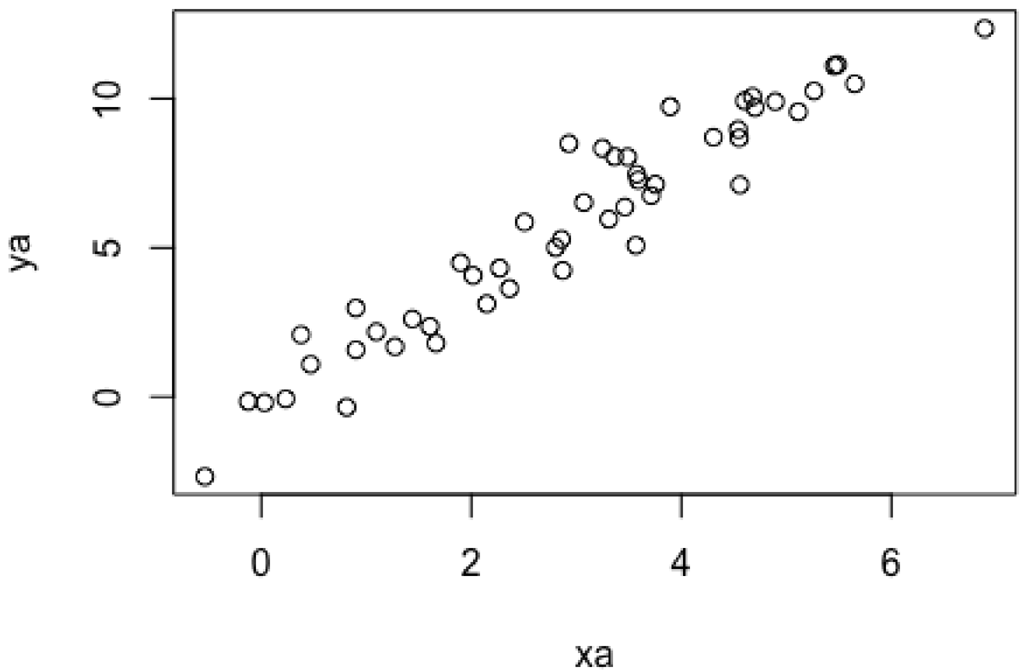

```
plot(u1,vw$residuals)
abline(h=0)
```



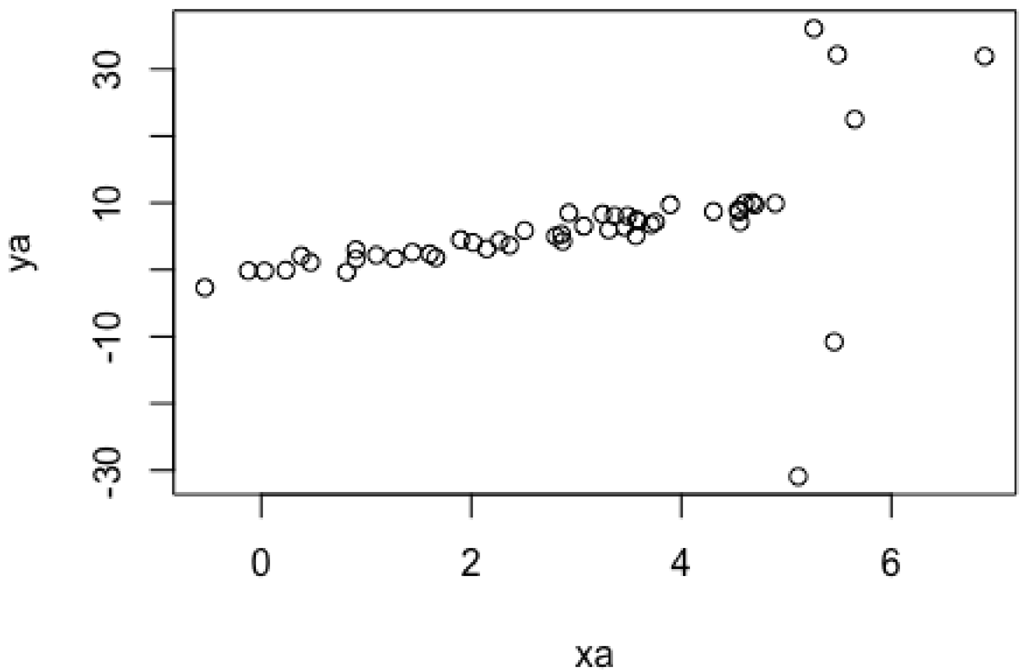
Other examples of influential observations (move a point up or down-do for above) ____ Example below of how residuals change in a certain region and so prediction is not as valid in that region.

Try varying the sd in the line ya[xa>3]=ya[xa>3]+rnorm(c,3,3) and rerunning. Notice where the residuals get bigger.

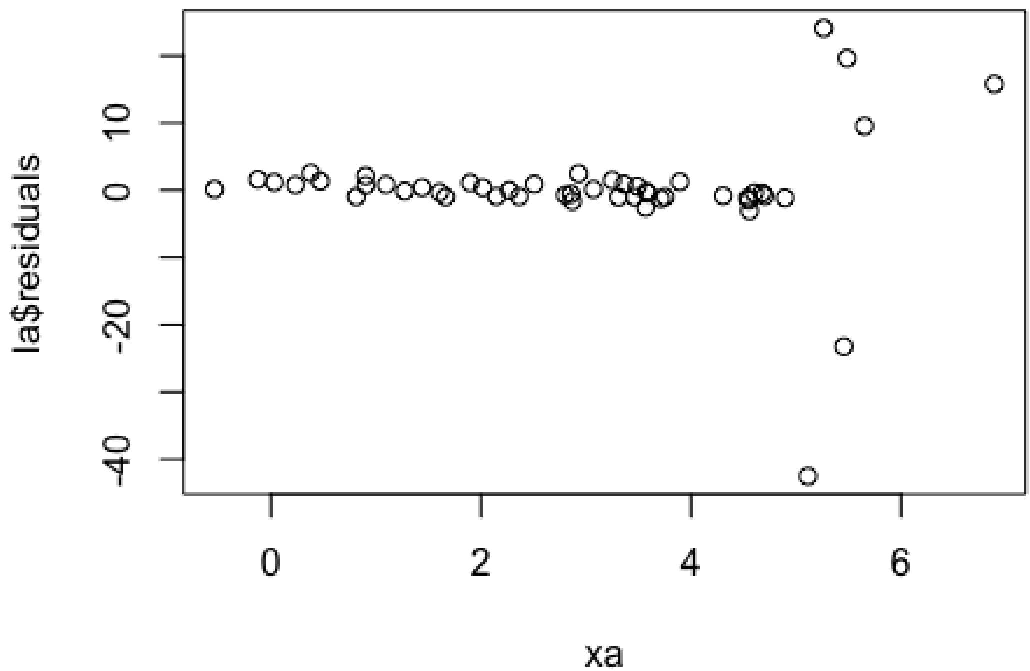
```
xa=rnorm(50,3,2)
ya=xa*2+rnorm(50)
num=table(ya[xa>5])
num

##
##  9.5561391339406 10.2552391062385 10.4972841943192 11.1005391405755
##                1                1                1                1
## 11.1358052557809 12.3474695366579
##                1                1

plot(xa,ya)
```




```
c=sum(num) #I'll change points that have x coordinates greater than 5.This is how many points I am changing.
c

## [1] 6

ya[xa>5]=ya[xa>5]+rnorm(c,3,30) # the rnorm(c,3,30) with sd 30 adds in scattershot.
cor(xa,ya)

## [1] 0.4568334

plot(xa,ya)
```




```
la=lm(ya~xa)
plot(xa,la$residuals)
```



Try it:

a.PLot the scatter plot for u52016 as a function of life expectancy.(May also need to load unicef data set.)

b.Overlay the plot of the regression line for u52016 as a function of litrt.

c.Determine the equation of this regression line.

d. Use it to predict what u52016 value would correspond to a life expectancy of 75, of 56.

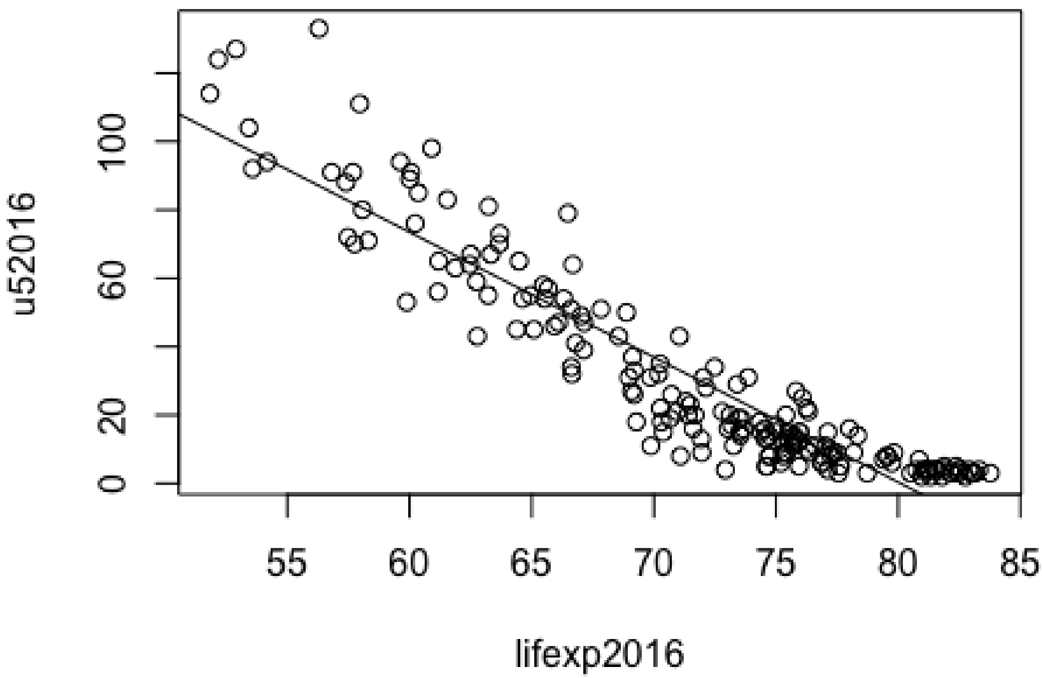
e. What are acceptable values of life expectancy for using for extrapolation?

f. What percent of the variation in u52016 is explained by the variation in life expectancy?

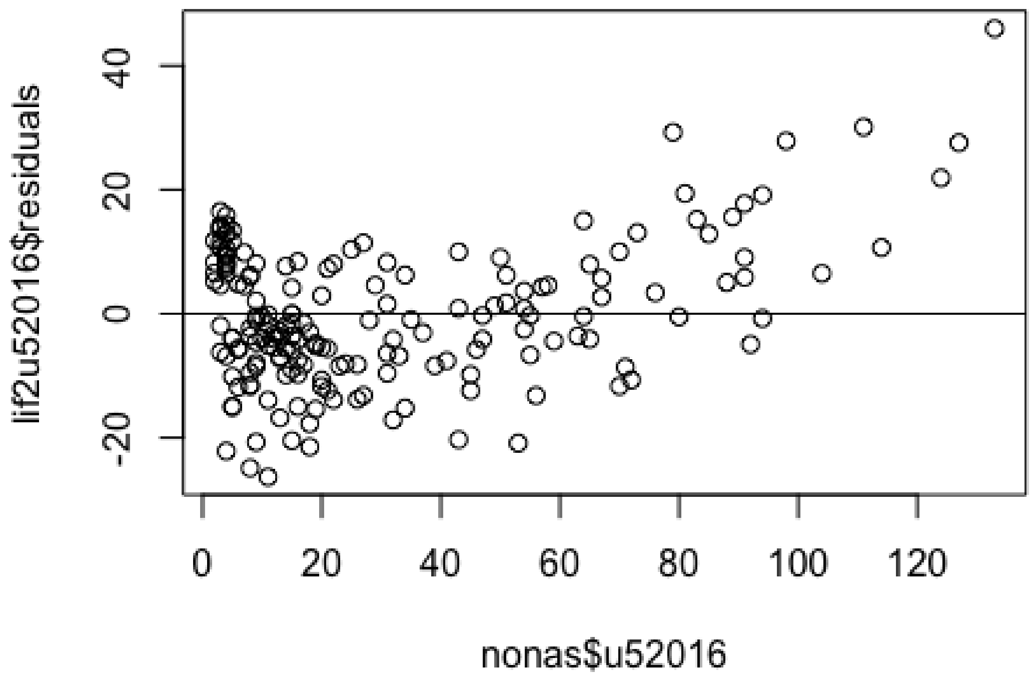
g. plot the residual plot

Other questions…

```
lifu52016=lm(u52016~lifexp2016,data=unicefb2016)
plot(u52016~lifexp2016,data=unicefb2016)
abline(lifu52016)
```




```
#plot(unicefb2016$lifexp2016,lifu52016$residuals) this gives error
#abline(h=0)
#length(lifu52016$residuals)
nonas=unicefb2016%>%filter(!is.na(lifexp2016)&!is.na(u52016))
#otherwise have problems plotting the residual plot
#length(nonas$u52016)
lif2u52016=lm(u52016~lifexp2016,data=nonas)
plot(nonas$u52016,lif2u52016$residuals)
abline(h=0) # comment on the graph of the residuals
```



Is the u52016 rate caused by the life expectancy?

Try it:

Write code to draw the scatter plot for the variable Education in swiss as a function of Agricultural and to add to this the least square regression line.

Write code to print out the slope and the y intercept for the least square regression line for the variable Education (education beyond primary school) in swiss as a function of Agricultural. Use this to predict what the percent of draftees with education beyond primary school would be for a province with 60% of men involved in Agriculture.

Is the percent of draftees with Primary School education caused by the percent of men involved in Agriculture?

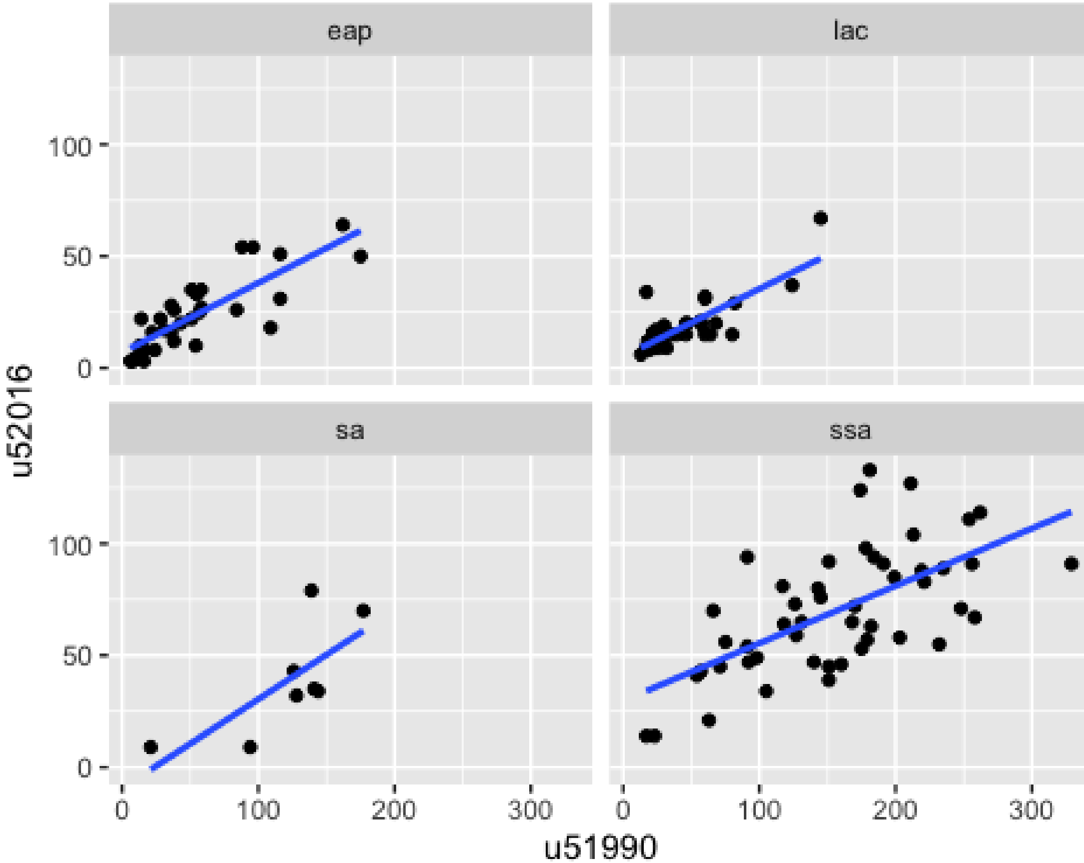
Something to consider: Compare the regression lines. You would just copy and modify this code.

```
unicefb2016%>%filter(classf%in%c('ssa','sa','eap','lac'))%>%ggplot(aes(u51990
,u52016))+geom_point()+geom_smooth(method='lm',se=F)+facet_wrap(vars(classf))
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 5 rows containing non-finite values (`stat_smooth()`).
```

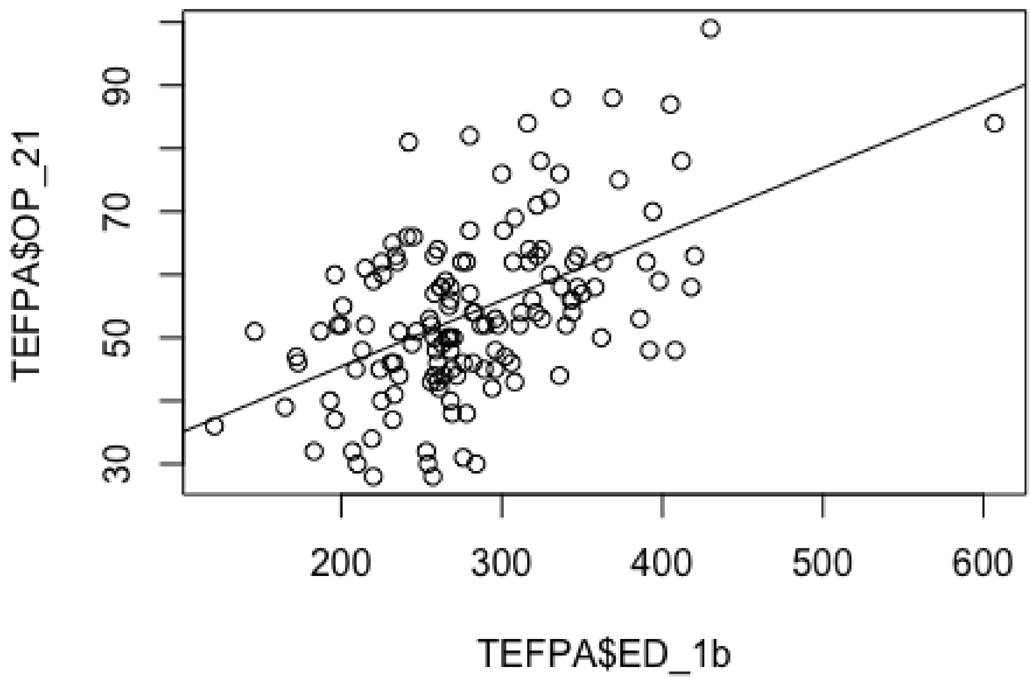
```
## Warning: Removed 5 rows containing missing values (`geom_point()`).
```



Try it: Do another example from your data or Unicef data.

A new dataset: TEF Timely and effective care at hospital. See DOB code in class resources or list given below.

```
#add your own version of this load statement
load("~/Desktop/Rnotes/TEF.rda")

#View(TEF) Commented off to knit. Uncomment to view the dataset

TEFPA=TEF%>%filter(State=='PA')
plot(TEFPA$ED_1b,TEFPA$OP_21)#OP_21 time to pain med,ED_1b time before
admitted as inpatient
cor(TEFPA$ED_1b,TEFPA$OP_21,use='pairwise.complete.obs')

## [1] 0.51944

abline(lm(TEFPA$OP_21~TEFPA$ED_1b))
```

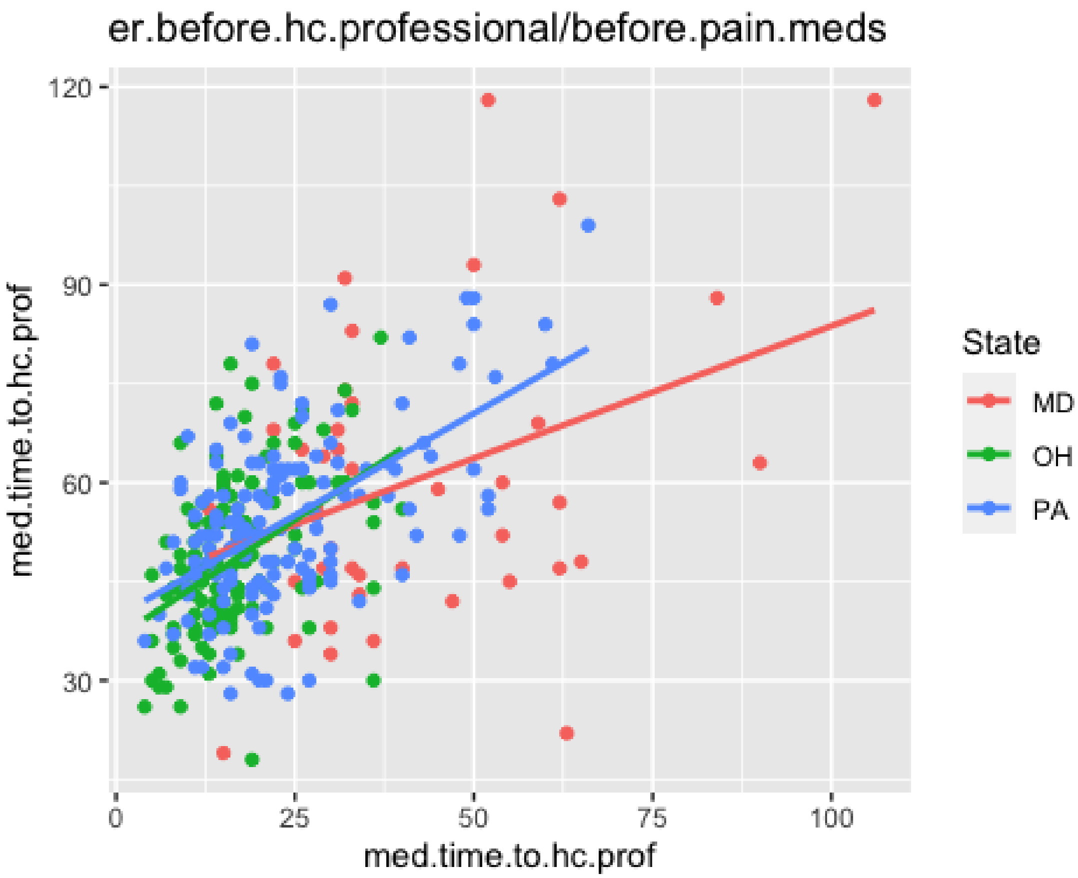


What are some questions/observations you have about this data and this graph? Put this is real terms.

Try working with some other variables or states.

An example of plotting multiple regression lines. Not strong correlation.

```
TEFPAOHMD=TEF%>%filter(State%in%c('PA','OH','MD')) #Just want the observations from these three states
ggplot(TEFPAOHMD,aes(OP_20,OP_21,col=State))+geom_point()+geom_smooth(method='lm',se=F)+labs(title='er.before.hc.professional/before.pain.meds',x='med.time.to.hc.prof',y='med.time.to.hc.prof')

## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 68 rows containing non-finite values (`stat_smooth()`).

## Warning: Removed 68 rows containing missing values (`geom_point()`).
```




```
TEFPAOHMD%>%group_by(State)%>%summarize(cr=cor(OP_20,OP_21,use="pairwise.comp
lete.obs"))
```

```
## # A tibble: 3 × 2
##   State    cr
##   <chr> <dbl>
## 1 MD    0.375
## 2 OH    0.462
## 3 PA    0.566
```

```
with(TEFPAOHMD,cor(OP_20,OP_21,use="pairwise.complete.obs"))
```

```
## [1] 0.522351
```

Information on the TEF dataset ED-1b Average (median) time patients spent in the emergency department, before they were admitted to the hospital as an inpatient (alternate Measure ID: ED-1) ED-2b Average (median) time patients spent in the emergency department, after the doctor decided to admit them as an inpatient before leaving the emergency department for their inpatient room (alternate Measure ID: ED-2)

EDV Emergency department volume (alternate Measure ID: EDV-1)

IMM-2 Patients assessed and given influenza vaccination

IMM-3

Healthcare workers given influenza vaccination (alternate Measure ID: IMM-3_OP_27_FAC_ADHPCT) OP-1 Median time to fibrinolysis. *This measure is only found in the downloadable database, it is not displayed on Hospital Compare

OP-2 Outpatients with chest pain or possible heart attack who got drugs to break up blood clots within 30 minutes of arrival OP-3b

Average (median) number of minutes before outpatients with chest pain or possible heart attack who needed specialized care were transferred to another hospital OP-4 Outpatients with chest pain or possible heart attack who received aspirin within 24 hours of arrival or before transferring from the emergency department OP-5

Average (median) number of minutes before outpatients with chest pain or possible heart attack got an ECG OP-18b Average (median) time patients spent in the emergency department before leaving from the visit (alternate Measure ID: OP-18)

OP-18c Average time patients spent in the emergency department before being sent home *This measure is only found in the downloadable database, it is not displayed on Hospital Compare OP-20 Average (median) time patients spent in the emergency department before they were seen by a healthcare professional OP-21 Average (median) time patients who came to the emergency department with broken bones had to wait before getting pain medication OP-22 Percentage of patients who left the emergency department before being seen OP-23 Percentage of patients who came to the emergency department with stroke symptoms who received brain scan results within 45 minutes of arrival OP-29 Percentage of patients receiving appropriate recommendation for follow-up screening colonoscopy OP-30 Percentage of patients with history of polyps receiving follow-up colonoscopy in the appropriate timeframe Percentage of patients who had cataract surgery and had improvement in visual function within 90 days following the surgery

OP-31 Percentage of patients who had cataract surgery and had improvement in visual function within 90 days following the surgery

OP-33 Percentage of patients receiving appropriate radiation therapy for cancer that has spread to the bone PC-01 Percent of mothers whose deliveries were scheduled too early (1-2 weeks early), when a scheduled delivery was not medically necessary SEP-1 Severe Sepsis and Septic Shock

VTE-6 Patients who developed a blood clot while in the hospital who did not get treatment that could have prevented it