**BACS HW (Week 10)**

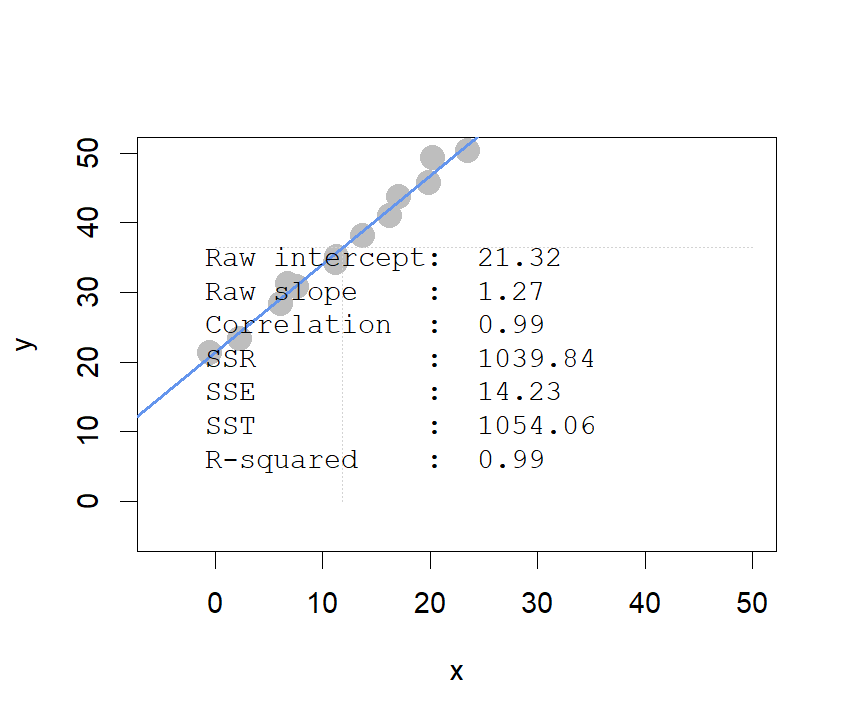
108070023

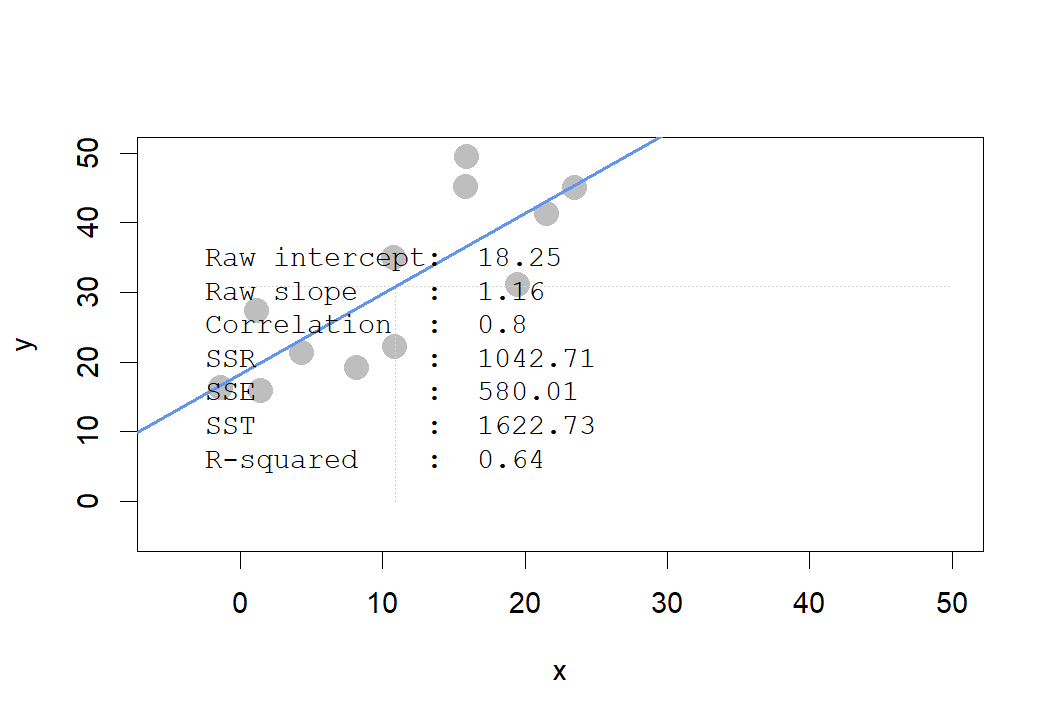
**Question 1)** Download demo\_simple\_regression\_rsq.R from Canvas – it has a function that runs a regression simulation. This week, the simulation also reports R2 along with the other metrics from last week.

To answer the questions below, understand each of these four scenarios by simulating them:  
Scenario 1: Consider a very narrowly dispersed set of points that have a negative or positive steep slope  
Scenario 2: Consider a widely dispersed set of points that have a negative or positive steep slope  
Scenario 3: Consider a very narrowly dispersed set of points that have a negative or positive shallow slope  
Scenario 4: Consider a widely dispersed set of points that have a negative or positive shallow slope

|  |  |  |  |
| --- | --- | --- | --- |
| *Scenario 1* | *Scenario 2* | *Scenario 3* | *Scenario 4* |
| https://lh6.googleusercontent.com/I9NBmFOibW1rvO4ICrBOS99uhEp738Y1r_bgbJmBcl6PDrionDFR6X6vEQfAdwbWBkfSXmfkwGQeFMvSFNX7ap1nWXGwcl1p_Io1uC4cdpP0mdVpkdFwAqA4YrGXyJV07OwnmrOc | https://lh6.googleusercontent.com/CNsOfLyGwdAwJwDhRhINKLqOBVaX5Zk54azBMwB5JmeI5M71ruVMQDXnNeJnjMhsX5khqqEzDka_zMQV1g-LCr23JYug2T5limVhEmjQnpJwlpdhCH_VTEZY_9UNFrbNuC31jLe6 | https://lh4.googleusercontent.com/agdy0vcuH1PgX-UxBoSKa9NozjLqf8UJbAIf0FZJlTLR5PROwEMY3VLJVP9JkP-zWnGiDeAXk06gX6ZO0veTgd5uQXujW2Zi3E3c2vCajK9iEiQa7whyZgQ93sNmwoHcOJ5RXhKE | https://lh3.googleusercontent.com/xV33WaPJNRz97VUeFDulp9XAcqCawhEfDWtidCoMPu6BNub1NJKRcGOZwZaUdBWZBwI7yJHqH8jVfKH_3OyrSZfan5Qlvkw7SLjuuVEIpN5blAc5gPIkNG3GIY0AQlABWHvCkjNv |

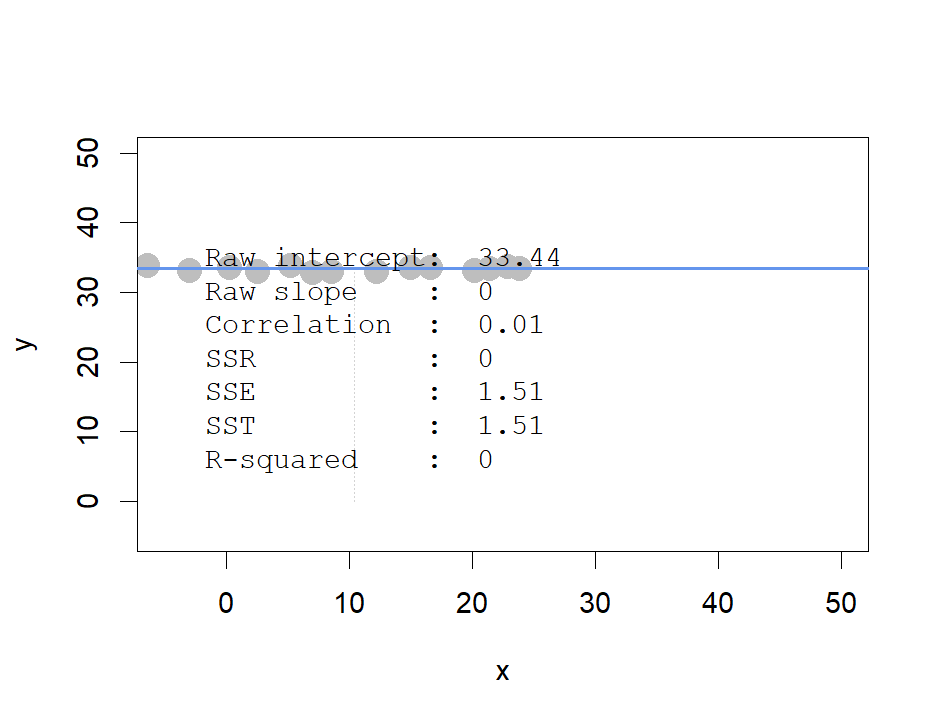
1. Comparing scenarios 1 and 2, which do we expect to have a stronger R2 ?

Ans: We’ll expect scenario 1 to have a stronger R. Since R is the portion that SSR take part in SST(how much SST is explained by regression), so if we explain it intuitively, the narrower the set of points lie, the closer the set of points to the line; therefore, SST is more explainable by regression line.

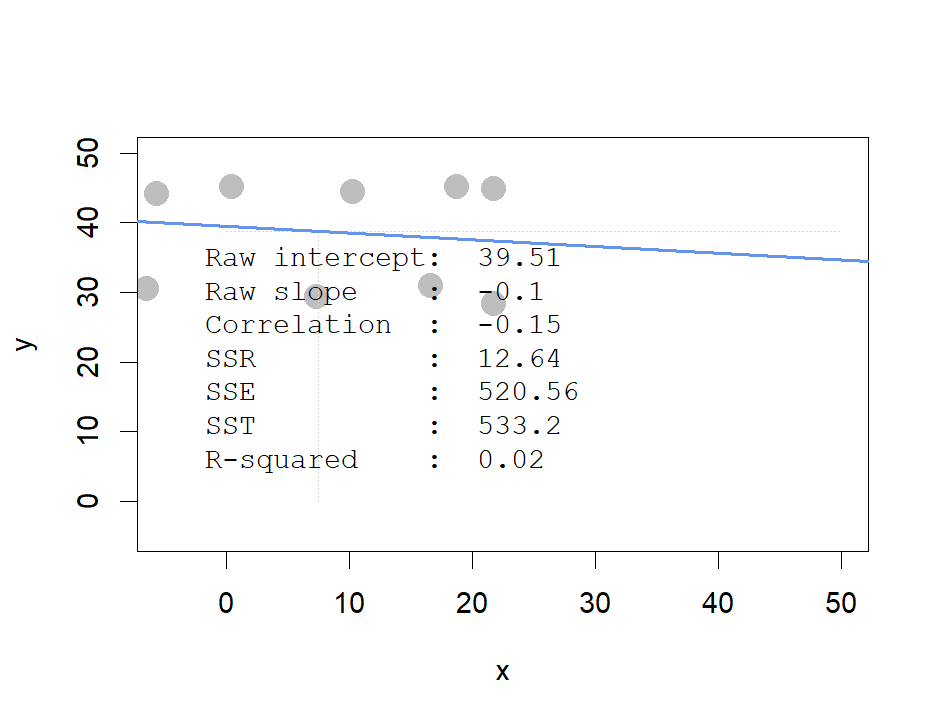


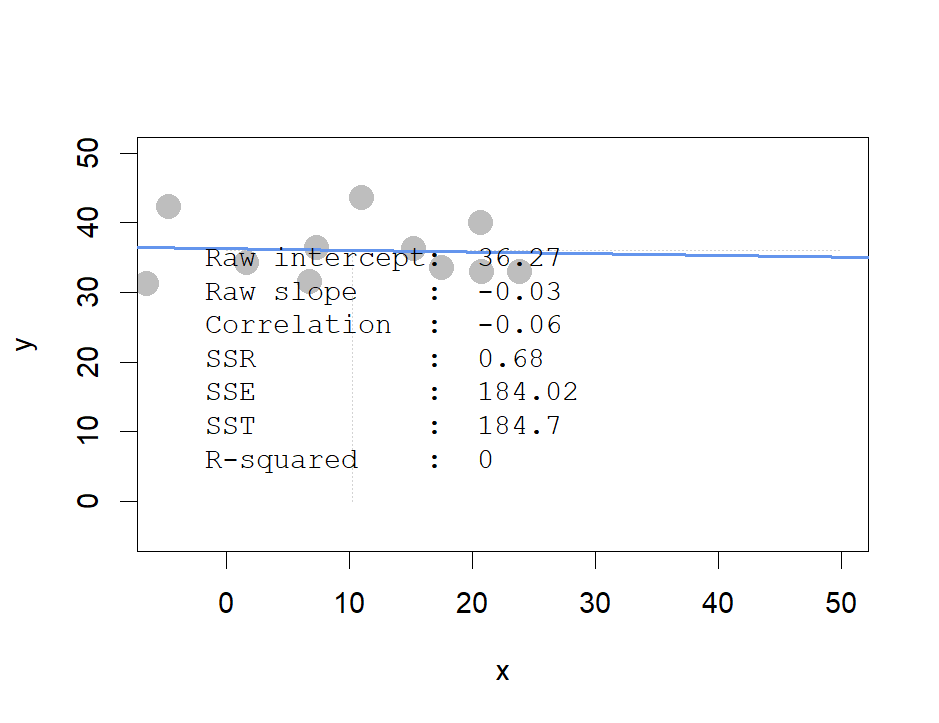
1. Comparing scenarios 3 and 4, which do we expect to have a stronger R2 ?

Ans: I’ll say that they both have similar R2  close to 0, but if points of scenario 4 are widely dispersed enough, R2  of scenario 4 will be stronger. Let’s look at figures below carefully, SSR of scenario 2 is a little bit bigger than scenario 1, while the amount of it is minor compared to SST. Therefore, we’ll observe the two scenarios to have similar R2 that is close to 0.



< scenario 1 >



 < two kinds of scenario 2 >

1. Comparing scenarios 1 and 2, which do we expect has bigger/smaller SSE, SSR, and SST? (intuitively)

Ans: We’ll expect scenario 2 has the bigger SST and SSE, since points in scenario 2 are more dispersed than scenario 1, and they may result in a bigger SST. Also, points in scenario 2 lie widely compared to scenario 1(lie farther to the line), which comes up with a bigger SSE(minimum sum of square of distance between y-hat and y ). However, we’ll expect SSR of scenario 2 to be stronger. As we discussed above, SSR means the

how much SST is explained by the regression line, if the points lie narrowly, the trend of them will look more similar to the regression line, and result in stronger SSR.

1. Comparing scenarios 3 and 4, which do we expect has bigger/smaller SSE, SSR, and SST? (intuitively)

Ans: We’ll expect scenario 4 to have higher SSE and SST, since its points disperse more widely. As for SSR, if the points are widely dispersed enough, it may result in the bigger SSR compared to scenario 3.

**Question 2)** Let’s perform regression ourselves on the programmer\_salaries.txt dataset we saw in class  
You can read the file using read.csv("programmer\_salaries.txt", sep="\t")

1. First, use the lm() function to estimate the model Salary ~ Experience + Score + Degree   
   (show the beta coefficients, R2 and the first 5 values of y  ($fitted.values) and ($residuals)

> #2-a

> sal<-read.csv("programmer\_salaries.txt", sep="\t")

> reg<-lm(Salary~Experience+Score+Degree,data = sal)

> head(sal,)

Experience Score Degree Salary

1 4 78 0 24.0

2 7 100 1 43.0

3 1 86 0 23.7

4 5 82 1 34.3

5 8 86 1 35.8

6 10 84 1 38.0

> summary(reg)

Call:

lm(formula = Salary ~ Experience + Score + Degree, data = sal)

Residuals:

Min 1Q Median 3Q Max

-3.8963 -1.7290 -0.3375 1.9699 5.0480

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 7.9448 7.3808 1.076 0.2977

Experience 1.1476 0.2976 3.856 0.0014 \*\*

Score 0.1969 0.0899 2.191 0.0436 \*

Degree 2.2804 1.9866 1.148 0.2679

---

Signif. codes:

0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.396 on 16 degrees of freedom

Multiple R-squared: 0.8468, Adjusted R-squared: 0.8181

F-statistic: 29.48 on 3 and 16 DF, p-value: 9.417e-07

> head(reg$residual,n=5)

1 2 3 4 5

-3.8962605 5.0479568 -2.3290112 2.1879860 -0.5425072

> head(reg$fitted.values,n=5)

1 2 3 4 5

27.89626 37.95204 26.02901 32.11201 36.34251

1. Use only linear algebra (and the geometric view of regression) to estimate the regression yourself:
   1. Create an X matrix that has a first column of 1s followed by columns of the independent variables

> #2-b-i

> salx<-data.matrix(sal[,1:3])

> salx<-cbind(c(1),salx)

> colnames(salx)[1]<-"intercept"

* 1. Create a y vector with the Salary values *(only show the code)*

> #2-b-ii

> saly<-sal$Salary

* 1. Compute the beta\_hat vector of estimated regression coefficients *(show the code and values)*

> #2-b-iii

> beta\_h<-(solve(t(salx)%\*%salx))%\*%t(salx)%\*%saly

> beta\_h

[,1]

intercept 7.944849

Experience 1.147582

Score 0.196937

Degree 2.280424

* 1. Compute a y\_hat vector of estimated y values, and a res vector of residuals   
     *(show the code and the first 5 values of y\_hat and res)*

> #2-b-iv

> saly\_h<-salx%\*%beta\_h

> head(saly\_h,n=5)

[,1]

[1,] 27.89626

[2,] 37.95204

[3,] 26.02901

[4,] 32.11201

[5,] 36.34251

> res<-saly-saly\_h

> head(res,n=5)

[,1]

[1,] -3.8962605

[2,] 5.0479568

[3,] -2.3290112

[4,] 2.1879860

[5,] -0.5425072

* 1. Using only the results from (i) – (iv), compute SSR, SSE and SST *(show the code and values)*

> #2-b-v

> SST<-sum((saly-mean(saly))^2)

> SSR<-sum((saly\_h-mean(saly))^2)

> SSE<-sum((saly-saly\_h)^2)

> data.frame(SST,SSR,SSE)

SST SSR SSE

1 599.7855 507.896 91.88949

1. Compute R2 for in two ways, and confirm you get the same results *(show code and values)*:
2. Use any combination of SSR, SSE, and SST

> #2-c-i

> R\_square\_1<-SSR/SST

> R\_square\_1

[1] 0.8467961

ii.Use the squared correlation of vectors y and y

> #2-c-ii

> R\_square\_2<-cor(saly,saly\_h)^2

> R\_square\_2

[,1]

[1,] 0.8467961

*(see question 3 on next page)*

**Question 3)** We’re going to take a look back at the early heady days of global car manufacturing, when American, Japanese, and European cars competed to rule the world. Take a look at the data set in file auto-data.txt. We are interested in explaining what kind of cars have higher fuel efficiency (mpg).

1. Let’s first try exploring this data and problem:
   1. Visualize the data in any way you feel relevant (report only relevant/interesting ones)

> #3-a-i

> library(dplyr)

> library(ggplot2)

> #install.packages("cowplot")

> library(cowplot)

> auto <- read.table("auto-data.txt", header=FALSE, na.strings = "?")

> names(auto) <- c("mpg", "cylinders", "displacement", "horsepower", "weight",

+ "acceleration", "model\_year", "origin", "car\_name")

> auto$brand <- as.character(lapply(auto$car\_name,function(x){unlist(strsplit(x," "))[1]}))

> my\_summary <- auto %>%

+ count(brand, sort = TRUE)

> my\_summary #too many categories of car\_name and brand, not valuable for linear regression model

brand n

1 ford 51

2 chevrolet 43

3 plymouth 31

4 amc 28

5 dodge 28

6 toyota 25

7 datsun 23

8 buick 17

9 pontiac 16

10 volkswagen 15

11 honda 13

12 mercury 11

13 mazda 10

14 oldsmobile 10

15 fiat 8

16 peugeot 8

17 audi 7

18 chrysler 6

19 volvo 6

20 vw 6

21 renault 5

22 opel 4

23 saab 4

24 subaru 4

25 chevy 3

26 bmw 2

27 cadillac 2

28 maxda 2

29 mercedes-benz 2

30 capri 1

31 chevroelt 1

32 hi 1

33 mercedes 1

34 nissan 1

35 toyouta 1

36 triumph 1

37 vokswagen 1

> auto['car\_name']<-NULL #abandon car\_name column

> auto['brand']<-NULL

> # Scatter plot

> cylinders <- ggplot(auto, aes(x = cylinders, y = mpg))+

+ geom\_point(size=0.8)

> displacement <- ggplot(auto, aes(x = displacement, y = mpg))+

+ geom\_point(size=0.8)

> horsepower <- ggplot(auto, aes(x = horsepower, y = mpg))+

+ geom\_point(size=0.8)

> weight <- ggplot(auto, aes(x = weight, y = mpg))+

+ geom\_point(size=0.8)

> acceleration <- ggplot(auto, aes(x = acceleration, y = mpg))+

+ geom\_point(size=0.8)

> model\_year <- ggplot(auto, aes(x = model\_year, y = mpg))+

+ geom\_point(size=0.8)

> origin <- ggplot(auto, aes(x = origin, y = mpg))+

+ geom\_point(size=0.8)

> ggdraw() +

+ draw\_plot(cylinders, 0, .6, .33, .35) +

+ draw\_plot(displacement, .33, .6, .33, .35) +

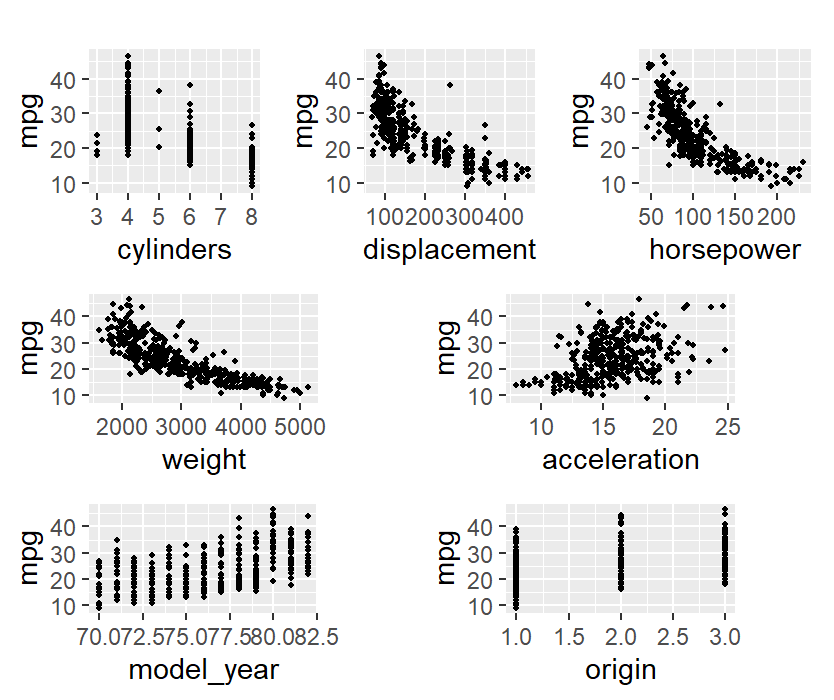
+ draw\_plot(horsepower, .66, .6, .33, .35) +

+ draw\_plot(weight, 0, .3, .4, .3) +

+ draw\_plot(acceleration, .5, .3, .4, .3) +

+ draw\_plot(model\_year, 0, 0, .4, .3) +

+ draw\_plot(origin, .5, 0, .4, .3)



<relationship between mpg and each variable>

* 1. Report a correlation table of all variables, rounding to two decimal places  
     (in the cor() function, set use="pairwise.complete.obs" to handle missing values)

> #3-a-ii

> round(cor(auto[,c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration", "model\_year", "origin")], use = "pairwise.complete.obs"),digits=2)

mpg cylinders displacement horsepower

mpg 1.00 -0.78 -0.80 -0.78

cylinders -0.78 1.00 0.95 0.84

displacement -0.80 0.95 1.00 0.90

horsepower -0.78 0.84 0.90 1.00

weight -0.83 0.90 0.93 0.86

acceleration 0.42 -0.51 -0.54 -0.69

model\_year 0.58 -0.35 -0.37 -0.42

origin 0.56 -0.56 -0.61 -0.46

weight acceleration model\_year origin

mpg -0.83 0.42 0.58 0.56

cylinders 0.90 -0.51 -0.35 -0.56

displacement 0.93 -0.54 -0.37 -0.61

horsepower 0.86 -0.69 -0.42 -0.46

weight 1.00 -0.42 -0.31 -0.58

acceleration -0.42 1.00 0.29 0.21

model\_year -0.31 0.29 1.00 0.18

origin -0.58 0.21 0.18 1.00

* 1. From the visualizations and correlations, which variables seem to relate to mpg?

A: ‘cylinders’, ‘displacement’,’horsepower’,’weight’ have correlation higher than 0.75 or smaller than -0.75,so I’ll say they seems to relate to mpg.

* 1. Which relationships might not be linear? *(don’t worry about linearity for rest of this HW)*

Ans: ‘acceleration’, ‘model year’, ‘origin’ may not have linear relationship with mpg since they have lower correlation, and the distribution of origin and model year seem like several vertical lines, I can’t observe linear relationship in these graphs.

* 1. Are there any pairs of independent variables that are highly correlated (*r > 0.7*)?

> #3-a-v

> library(reshape2)

> diag(cor\_table) <- 0

> cor\_melt <- melt(cor\_table)

> hight\_cor <- cor\_melt[order(abs(cor\_melt$value),decreasing = T) & abs(cor\_melt$value) >0.7,]

>

> #### eliminate the same combination of variable

> #sort two variable by first character order

> hight\_cor[1:2] <- t( apply(hight\_cor[1:2], 1, sort) )

> #eliminate the same variable combination

> hight\_cor[!duplicated(hight\_cor[1:2]),]

Var1 Var2 value

2 cylinders mpg -0.78

3 displacement mpg -0.80

4 horsepower mpg -0.78

5 mpg weight -0.83

11 cylinders displacement 0.95

12 cylinders horsepower 0.84

13 cylinders weight 0.90

20 displacement horsepower 0.90

21 displacement weight 0.93

29 horsepower weight 0.86

 Ans: The above result shows pairs of independent variables that are highly correlated (*r > 0.7*)

1. Let’s create a linear regression model where mpg is dependent upon all other suitable variables *(Note: origin is categorical with three levels, so use factor(origin) in lm(...)  to split it into two dummy variables)*
2. Which independent variables have a ‘significant’ relationship with mpg at 1% significance?

> #3-b

> mpg\_reg<-lm(mpg~cylinders+displacement+horsepower+weight+acceleration+model\_year+factor(origin),data = auto)

> summary(mpg\_reg)

Call:

lm(formula = mpg ~ cylinders + displacement + horsepower + weight +

acceleration + model\_year + factor(origin), data = auto)

Residuals:

Min 1Q Median 3Q Max

-9.0095 -2.0785 -0.0982 1.9856 13.3608

Coefficients:

Estimate Std. Error t value

(Intercept) -1.795e+01 4.677e+00 -3.839

cylinders -4.897e-01 3.212e-01 -1.524

displacement 2.398e-02 7.653e-03 3.133

horsepower -1.818e-02 1.371e-02 -1.326

weight -6.710e-03 6.551e-04 -10.243

acceleration 7.910e-02 9.822e-02 0.805

model\_year 7.770e-01 5.178e-02 15.005

factor(origin)2 2.630e+00 5.664e-01 4.643

factor(origin)3 2.853e+00 5.527e-01 5.162

Pr(>|t|)

(Intercept) 0.000145 \*\*\*

cylinders 0.128215

displacement 0.001863 \*\*

horsepower 0.185488

weight < 2e-16 \*\*\*

acceleration 0.421101

model\_year < 2e-16 \*\*\*

factor(origin)2 4.72e-06 \*\*\*

factor(origin)3 3.93e-07 \*\*\*

---

Signif. codes:

0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.307 on 383 degrees of freedom

Multiple R-squared: 0.8242, Adjusted R-squared: 0.8205

F-statistic: 224.5 on 8 and 383 DF, p-value: < 2.2e-16

 Ans: ’Intercept’, ‘displacement’, ’ weight’, ’model\_year’, ’origin’ have significant relationship with mpg at 1% significance?

1. Looking at the coefficients, is it possible to determine which independent variables are the *most effective* at increasing mpg? If so, which ones, and if not, why not? (hint: units!)

Ans: No, because those variable are in different scales, it's not possible to compare their magnitude directly. We should standardize them and compare between them.

1. Let’s try to resolve some of the issues with our regression model above.
2. Create fully standardized regression results: are these slopes easier to compare?  
   (note: consider if you should standardize origin)

> #3-c-i

> auto\_std <- cbind(scale(auto[1:7]),auto$origin)#origin is categorical variable

> colnames(auto\_std) <- colnames(auto[1:8])

> auto\_std <- data.frame(auto\_std)

> mpg\_stdreg<-lm(mpg~cylinders+displacement+horsepower+weight+acceleration+model\_year+factor(origin),data = auto\_std)

> summary(mpg\_stdreg)

Call:

lm(formula = mpg ~ cylinders + displacement + horsepower + weight +

acceleration + model\_year + factor(origin), data = auto\_std)

Residuals:

Min 1Q Median 3Q Max

-1.15270 -0.26593 -0.01257 0.25404 1.70942

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -0.13323 0.03174 -4.198 3.35e-05

cylinders -0.10658 0.06991 -1.524 0.12821

displacement 0.31989 0.10210 3.133 0.00186

horsepower -0.08955 0.06751 -1.326 0.18549

weight -0.72705 0.07098 -10.243 < 2e-16

acceleration 0.02791 0.03465 0.805 0.42110

model\_year 0.36760 0.02450 15.005 < 2e-16

factor(origin)2 0.33649 0.07247 4.643 4.72e-06

factor(origin)3 0.36505 0.07072 5.162 3.93e-07

(Intercept) \*\*\*

cylinders

displacement \*\*

horsepower

weight \*\*\*

acceleration

model\_year \*\*\*

factor(origin)2 \*\*\*

factor(origin)3 \*\*\*

---

Signif. codes:

0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.423 on 383 degrees of freedom

Multiple R-squared: 0.8242, Adjusted R-squared: 0.8205

F-statistic: 224.5 on 8 and 383 DF, p-value: < 2.2e-16

Ans: Yes, it is easier to interpret the coefficient after standardization. According the report above, we can find out that weight is the most effective at increasing mpg since it has the highest correlation with mpg.

1. Regress mpg over each *nonsignificant* independent variable, individually.  
   Which ones become significant when we regress mpg over them individually?

> #3-c-ii

> #unsignificant var:cylinders,horsepower,acceleration

> cy<-lm(mpg~cylinders,data=auto)

> summary(cy)

Call:

lm(formula = mpg ~ cylinders, data = auto)

Residuals:

Min 1Q Median 3Q Max

-14.2607 -3.3841 -0.6478 2.5538 17.9022

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 42.9493 0.8330 51.56 <2e-16 \*\*\*

cylinders -3.5629 0.1458 -24.43 <2e-16 \*\*\*

---

Signif. codes:

0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.942 on 396 degrees of freedom

Multiple R-squared: 0.6012, Adjusted R-squared: 0.6002

F-statistic: 597.1 on 1 and 396 DF, p-value: < 2.2e-16

> hp<-lm(mpg~horsepower,data=auto)

> summary(hp)

Call:

lm(formula = mpg ~ horsepower, data = auto)

Residuals:

Min 1Q Median 3Q Max

-13.5710 -3.2592 -0.3435 2.7630 16.9240

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 39.935861 0.717499 55.66 <2e-16 \*\*\*

horsepower -0.157845 0.006446 -24.49 <2e-16 \*\*\*

---

Signif. codes:

0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.906 on 390 degrees of freedom

(因為不存在，6 個觀察量被刪除了)

Multiple R-squared: 0.6059, Adjusted R-squared: 0.6049

F-statistic: 599.7 on 1 and 390 DF, p-value: < 2.2e-16

> ac<-lm(mpg~acceleration,data=auto)

> summary(ac)

Call:

lm(formula = mpg ~ acceleration, data = auto)

Residuals:

Min 1Q Median 3Q Max

-18.007 -5.636 -1.242 4.758 23.192

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 4.9698 2.0432 2.432 0.0154 \*

acceleration 1.1912 0.1292 9.217 <2e-16 \*\*\*

---

Signif. codes:

0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.101 on 396 degrees of freedom

Multiple R-squared: 0.1766, Adjusted R-squared: 0.1746

F-statistic: 84.96 on 1 and 396 DF, p-value: < 2.2e-16

Ans: The above figures show the regression line for three nonsignificant independent variable: **cylinders, horsepower, acceleration**. According to the p-value of each regression, all of them are significant if we regress mpg over them individually.

iii.Plot the density of the *residuals*: are they normally distributed and centered around zero?  
(get the residuals of a fitted linear model, e.g. regr <- lm(...), using regr$residuals

> #3-c-iii

> plot(density(cy$residuals),main = "cylinders$residuals")

> plot(density(hp$residuals),main = "horsepower$residuals")

> plot(density(ac$residuals),main = "acceleration$residuals")

Ans: Yes, they all center around 0 and their distribution look similar to normal distribution.

