BACS HW12

106070020

2021年5月15日

Question 1

(a)Run a new regression on the cars_log dataset, with mpg.log. dependent on all other variables

(i)Which log-transformed factors have a significant effect on log.mpg. at 10% significance?

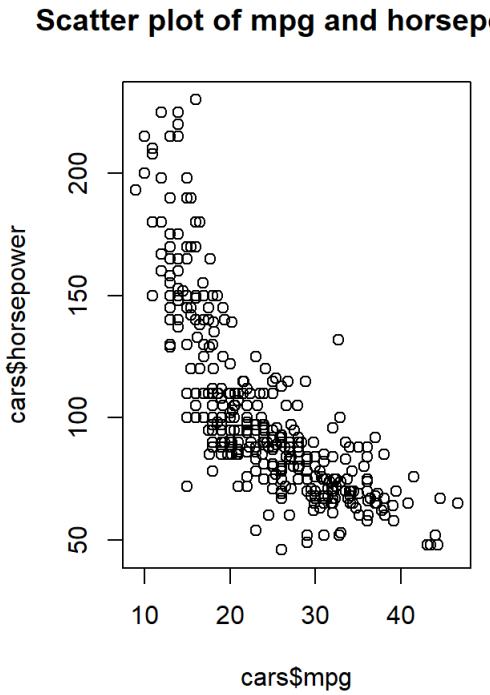
```
cars <- read.table("C:/Users/eva/Desktop/作業 上課資料(清大)/大四下/BACS/HW11 BACS/auto-data.txt", header=F, na.strings
 = "?")
names(cars) <- c("mpg", "cylinders", "displacement", "horsepower", "weight",
                 "acceleration", "model_year", "origin", "car_name")
cars_log <- with(cars, data.frame(log(mpg), log(cylinders), log(displacement), log(horsepower), log(weight), log(accel
eration), model_year, origin))

engine_regr <-lm(log.mpg. ~ log.cylinders. + log.displacement.+log.horsepower.+ log.weight.+log.acceleration.
                 +model_year+factor(origin), data=cars_log, na.action=na.exclude)
summary(engine_regr)
```

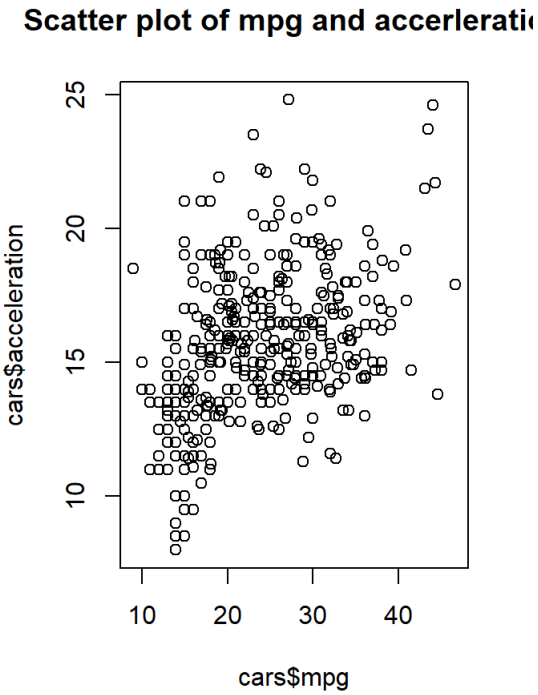
```
##
## Call:
## lm(formula = log.mpg. ~ log.cylinders. + log.displacement. +
##     log.horsepower. + log.weight. + log.acceleration. + model_year +
##     factor(origin), data = cars_log, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39727 -0.06880  0.00450  0.06356  0.38542
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.301938   0.361777  20.184  < 2e-16 ***
## log.cylinders.    -0.081915   0.061116  -1.340  0.18094
## log.displacement.  0.020387   0.058369   0.349  0.72707
## log.horsepower.   -0.284751   0.057945  -4.914 1.32e-06 ***
## log.weight.       -0.592955   0.085165  -6.962 1.46e-11 ***
## log.acceleration. -0.169673   0.059649  -2.845  0.00469 **
## model_year         0.030239   0.001771  17.078  < 2e-16 ***
## factor(origin)2    0.050717   0.020920   2.424  0.01580 *
## factor(origin)3    0.047215   0.020622   2.290  0.02259 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.113 on 383 degrees of freedom
##   (6 observations deleted due to missingness)
## Multiple R-squared:  0.8919, Adjusted R-squared:  0.8897
## F-statistic:   395 on 8 and 383 DF,  p-value: < 2.2e-16
```

> Under 10% significance, log.horsepower., log.weight., log.acceleration., model_year, and origin have a significant effect on log.mpg.

(ii)Do some new factors now have effects on mpg, and why might this be?

```
par(mfrow=c(1,2))
plot(cars$mpg, cars$horsepower, main='Scatter plot of mpg and horsepower')
plot(cars$mpg, cars$acceleration, main='Scatter plot of mpg and accerleration')
```

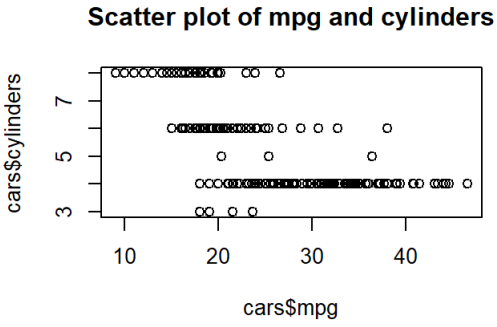


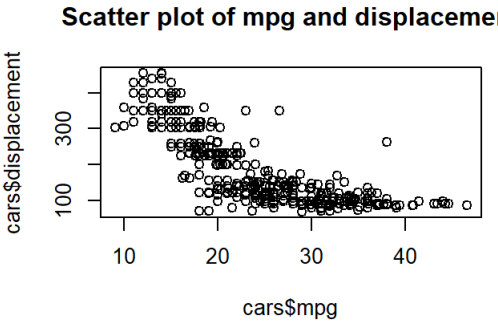


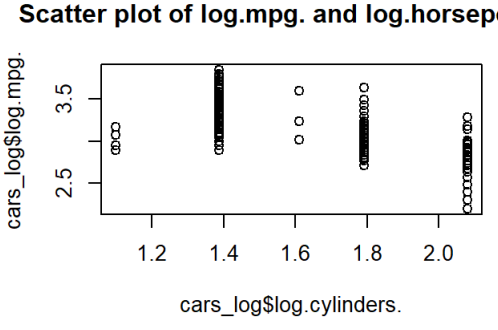
log.horsepower. and log.acceleration. are new factors now have effects on mpg. The reason of this situation may be that the relationship between these factors and mpg may be exponential relationship(e.g. the relationship between horsepower and mpg seems to be exponential).

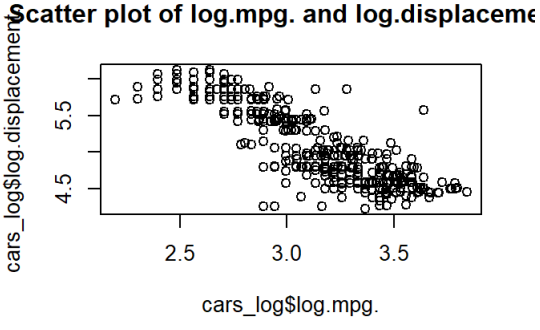
(iii)Which factors still have insignificant or opposite (from correlation) effects on mpg? Why might this be?

```
par(mfrow=c(2,2))
plot(cars$mpg, cars$cylinders, main='Scatter plot of mpg and cylinders')
plot(cars$mpg, cars$displacement, main='Scatter plot of mpg and displacement')
plot(cars_log$log.cylinders.,cars_log$log.mpg., main='Scatter plot of log.mpg. and log.horsepower.')
plot(cars_log$log.mpg.,cars_log$log.displacement.,main='Scatter plot of log.mpg. and log.displacement.')
```









> log.cylinders. and log.displacement. are still have insignificant effects on log.mpg. According to the plots above, cylinders and displacement seem not having the linear or exponential relationship with mpg. Also, the relationship of log.mpg. with log.cylinders. and log.mpg. with log.displacement are also not linear or exponential.

(b)Let's take a closer look at weight, because it seems to be a major explanation of mpg

(i)Create a regression (call it regr_wt) of mpg on weight from the original cars dataset

```
regr_wt<-lm(mpg~weight, data = cars)
summary(regr_wt)
```

```
##
## Call:
## lm(formula = mpg ~ weight, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -12.012  -2.801  -0.351   2.114  16.480
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 46.3173644  0.7952452   58.24   <2e-16 ***
## weight      -0.0076766  0.0002575  -29.81   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.345 on 396 degrees of freedom
## Multiple R-squared:  0.6918, Adjusted R-squared:  0.691
## F-statistic: 888.9 on 1 and 396 DF,  p-value: < 2.2e-16
```

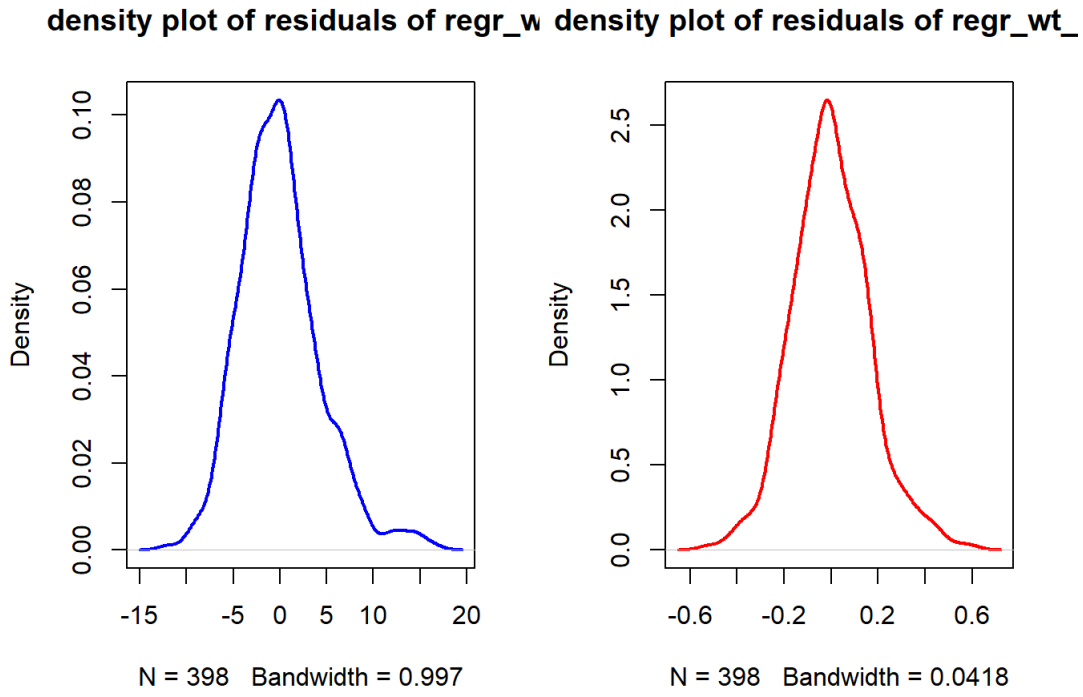
(ii)Create a regression (call it regr_wt_log) of log.mpg. on log.weight. from cars_log

```
regr_wt_log<-lm(log.mpg.~log.weight., data=cars_log)
summary(regr_wt_log)
```

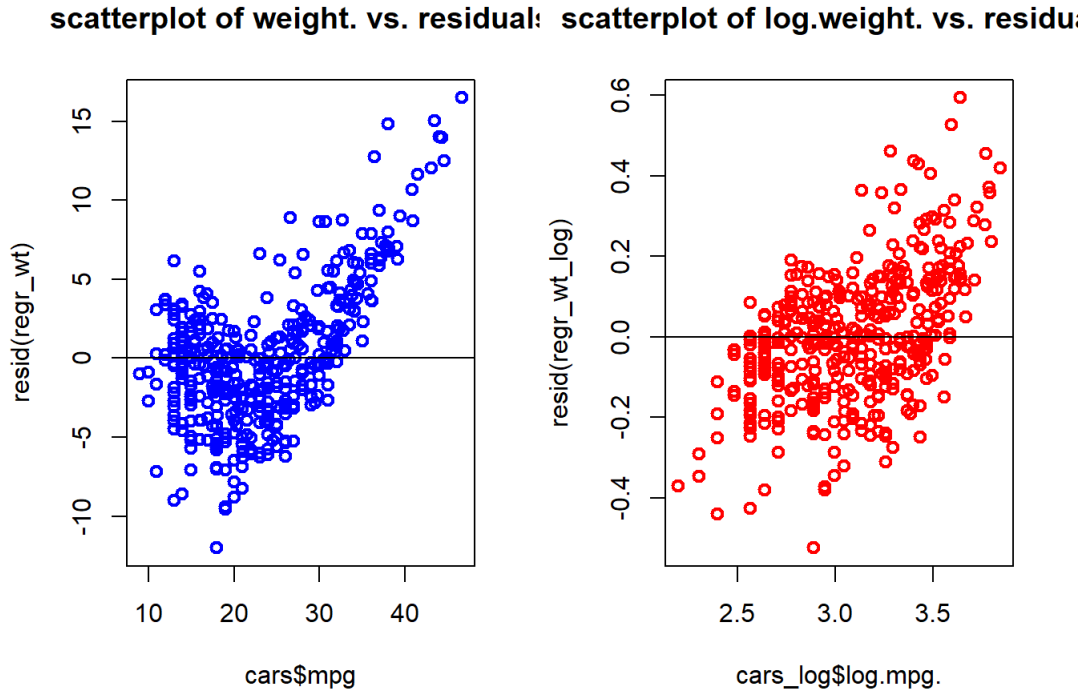
```
##
## Call:
## lm(formula = log.mpg. ~ log.weight., data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.52408 -0.10441 -0.00805  0.10165  0.59384
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11.5219     0.2349   49.06   <2e-16 ***
## log.weight.  -1.0583     0.0295  -35.87   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.165 on 396 degrees of freedom
## Multiple R-squared:  0.7647, Adjusted R-squared:  0.7641
## F-statistic:  1287 on 1 and 396 DF,  p-value: < 2.2e-16
```

(iii)Visualize the residuals of both regression models (raw and log-transformed)

```
par(mfrow=c(1,2))
plot(density(regr_wt$residuals), col='blue', lwd=2, main='density plot of residuals of regr_wt')
plot(density(regr_wt_log$residuals), col='red', lwd=2, main='density plot of residuals of regr_wt_log')
```




```
par(mfrow=c(1,2))
{plot(cars$mpg, resid(regr_wt), col='blue', lwd=2, main='scatterplot of weight. vs. residuals')
abline(h=0)
}
{plot(cars_log$log.mpg., resid(regr_wt_log), col='red', lwd=2, main='scatterplot of log.weight. vs. residuals')
  abline(h=0)
}
```



(iv)How would you interpret the slope of log.weight. vs log.mpg. in simple words?

```
#Q1(b)(iv)
regr_wt_log$coefficients
```

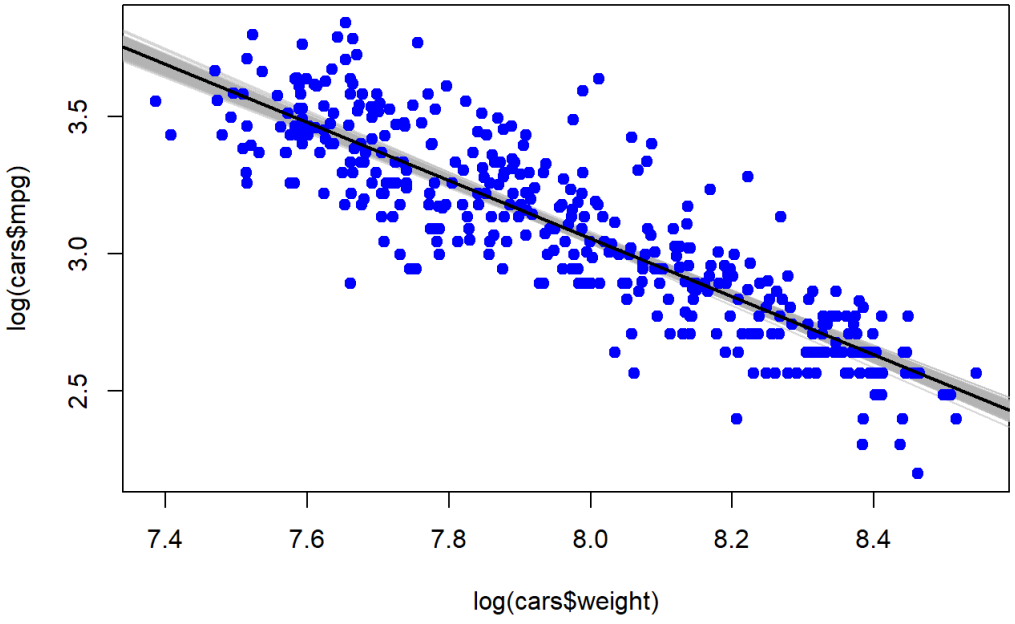
```
## (Intercept) log.weight.
##   11.521907   -1.058268
```

The slope of log.weight. vs log.mpg. is -1.058268, which means that they have negative relationship. (1 unit increase in log.weight. will cause 1.058268 decrease in log.mpg.)

(c)Let's examine the 95% confidence interval of the slope of log.weight. vs. log.mpg.

(i)Create a bootstrapped confidence interval

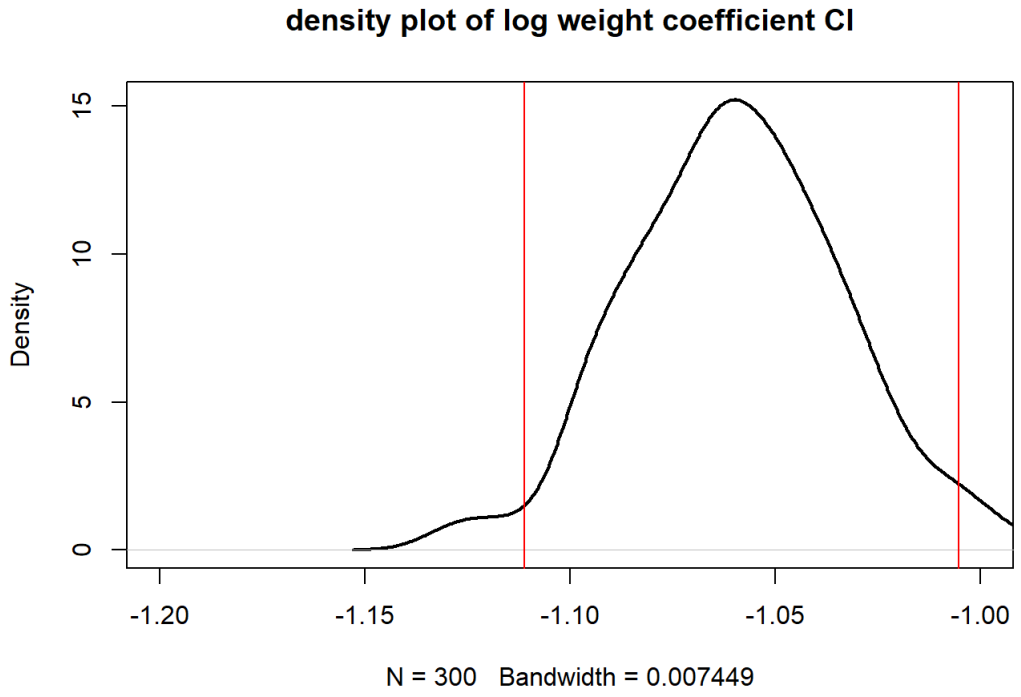
```
plot(log(cars$weight), log(cars$mpg), col=NA, pch=19)
# Function for single resampled regression line
boot_regr<-function(model, dataset) {
  boot_index<-sample(1:nrow(dataset), replace=TRUE)
  data_boot<-dataset[boot_index,]
  regr_boot<-lm(model, data=data_boot)
  abline(regr_boot,lwd=1, col=rgb(0.7, 0.7, 0.7, 0.5))
  regr_boot$coefficients
}
coeffs<-replicate(300,boot_regr(log(mpg) ~ log(weight), cars))

#Plot points and regression line
points(log(cars$weight), log(cars$mpg), col="blue",pch=19)
abline(a=mean(coeffs["(Intercept)",]),b=mean(coeffs["log(weight)",]),lwd=2)
```




```
#Confidence interval values
quantile(coeffs["log(weight)",], c(0.025, 0.975))
```

```
##      2.5%     97.5%
## -1.111066 -1.005158
```

```
#Plot confidence interval of coefficient
plot(density(coeffs["log(weight)",]),xlim=c(-1.2, -1),
     main='density plot of log weight coefficient CI', lwd=2)
abline(v=quantile(coeffs["log(weight)",], c(0.025, 0.975)), col='red')
```



(ii)Verify your results with a confidence interval using traditional statistics

```
hp_regr_log<-lm(log(mpg) ~ log(weight), cars)
confint(hp_regr_log)
```

```
##                 2.5 %    97.5 %
## (Intercept) 11.060154 11.983659
## log(weight) -1.116264 -1.000272
```

```
quantile(coeffs["log(weight)",], c(0.025, 0.975))
```

```
##      2.5%     97.5%
## -1.111066 -1.005158
```

> The results are very similar.

Q2

(a)Using regression and R2, compute the VIF of log.weight. using the approach shown in class

```
regr_log <- lm(log.mpg. ~ log.cylinders. + log.displacement. + log.horsepower. +
                log.weight. + log.acceleration. + model_year +
                factor(origin),  data=cars_log)

weight_regr<-lm(weight ~ cylinders + displacement + horsepower + acceleration +model_year+ factor(origin),data=cars,n
a.action=na.exclude)
r2_weight <-summary(weight_regr)$r.squared
vif_weight<-1 / (1-r2_weight)
sqrt(vif_weight)
```

```
## [1] 3.327814
```

(b)Let's try a procedure called Stepwise VIF Selection to remove highly collinear predictors.

(i)Use vif(regr_log) to compute VIF of the all the independent variables

```
library(car)
vif(regr_log)
```

```
##                       GVIF Df GVIF^(1/(2*Df))
## log.cylinders.    10.456738  1        3.233688
## log.displacement. 29.625732  1        5.442952
## log.horsepower.   12.132057  1        3.483110
## log.weight.       17.575117  1        4.192269
## log.acceleration.  3.570357  1        1.889539
## model_year         1.303738  1        1.141814
## factor(origin)     2.656795  2        1.276702
```

(ii)Eliminate from your model the single independent variable with the largest VIF score that is also greater than 5

> Eliminate log.displacement., whose GVIF is larger than 5.

(iii)Repeat steps (i) and (ii) until no more independent variables have VIF scores above 5

```
regr_log2 <- lm(log.mpg. ~ log.cylinders.  + log.horsepower. +
                log.weight. + log.acceleration. + model_year +
                factor(origin),  data=cars_log)

vif(regr_log2)
```

```
##                       GVIF Df GVIF^(1/(2*Df))
## log.cylinders.     5.433107  1        2.330903
## log.horsepower.   12.114475  1        3.480585
## log.weight.       11.239741  1        3.352572
## log.acceleration.  3.327967  1        1.824272
## model_year         1.291741  1        1.136548
## factor(origin)     1.897608  2        1.173685
```

```
regr_log3 <- lm(log.mpg. ~ log.cylinders.  +
                log.weight. + log.acceleration. + model_year +
                factor(origin),  data=cars_log)

vif(regr_log3)
```

```
##                      GVIF Df GVIF^(1/(2*Df))
## log.cylinders.    5.321090  1        2.306749
## log.weight.       4.788498  1        2.188264
## log.acceleration. 1.400111  1        1.183263
## model_year        1.201815  1        1.096273
## factor(origin)    1.792784  2        1.157130
```

```
regr_log4 <- lm(log.mpg. ~ log.weight. + log.acceleration. + model_year +
                factor(origin),  data=cars_log)

vif(regr_log4)
```

```
##                      GVIF Df GVIF^(1/(2*Df))
## log.weight.       1.926377  1        1.387940
## log.acceleration. 1.303005  1        1.141493
## model_year        1.167241  1        1.080389
## factor(origin)    1.692320  2        1.140567
```

(iv)Report the final regression model and its summary statistics

```
#Final regression model
regr_log4
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin), data = cars_log)
##
## Coefficients:
##       (Intercept)        log.weight.  log.acceleration.         model_year
##           7.43116           -0.87661            0.05151            0.03273
##   factor(origin)2    factor(origin)3
##           0.05799            0.03233
```

```
#summary
summary(regr_log4)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin), data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38275 -0.07032  0.00491  0.06470  0.39913
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.431155   0.312248  23.799  < 2e-16 ***
## log.weight.       -0.876608   0.028697 -30.547  < 2e-16 ***
## log.acceleration.  0.051508   0.036652   1.405  0.16072
## model_year         0.032734   0.001696  19.306  < 2e-16 ***
## factor(origin)2    0.057991   0.017885   3.242  0.00129 **
## factor(origin)3    0.032333   0.018279   1.769  0.07770 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1156 on 392 degrees of freedom
## Multiple R-squared:  0.8856, Adjusted R-squared:  0.8841
## F-statistic: 606.8 on 5 and 392 DF,  p-value: < 2.2e-16
```

(c)Using stepwise VIF selection, have we lost any variables that were previously significant? If so, how much did we hurt our explanation by dropping those variables?

```
a<-summary(regr_log4)
b<-summary(regr_log)
a$r.squared
```

```
## [1] 0.8855764
```

```
b$r.squared
```

```
## [1] 0.89191
```

> We lose log.cylinders., log.horsepower., and log.displacement. log.horsepower. used to be the significant variables. However, we just hurt our explanationa little by dropping those variables. The R-square of regr_log(0.89191) is a little bit larger than that of regr_log4(0.8855764).

(d)From only the formula for VIF, try deducing/deriving the following:

(i)If an independent variable has no correlation with other independent variables, what would its VIF score be?

> The VIF should be 1.

(ii)Given a regression with only two independent variables (X1 and X2), how correlated would X1 and X2 have to be, to get VIF scores of 5 or higher? To get VIF scores of 10 or higher?

1.

- VIF= 1/(1-r2)>5
- 5*(1-r2)<1
- (1-r2) < 1/5
- r2 > 4/5
- Ans: The R-square of two independent variable should larger than 0.8 to make VIF>5

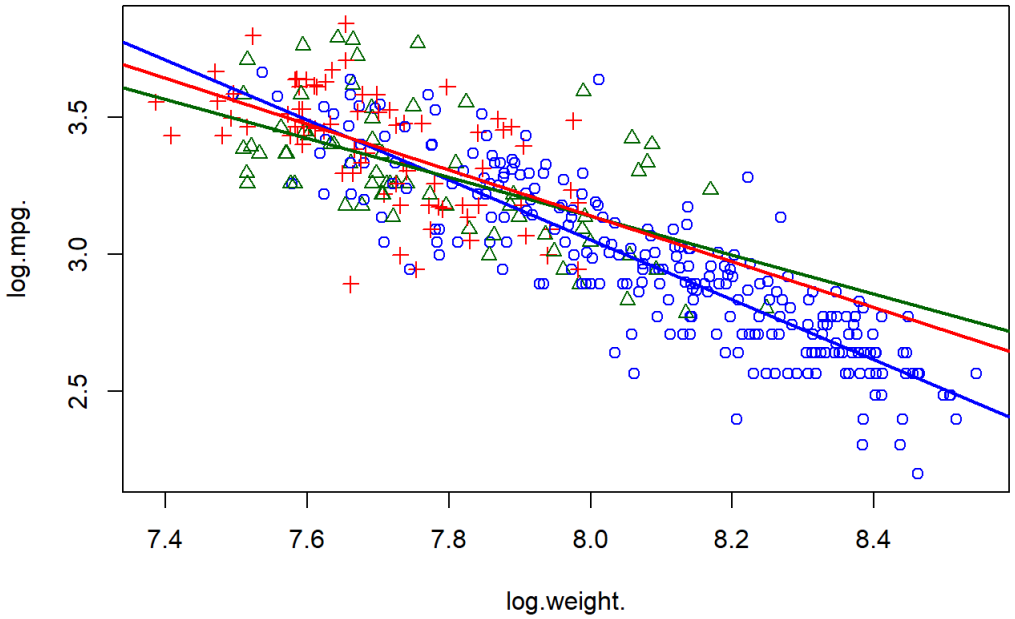
2.

- VIF = 1/(1-r2)>10
- 10*(1-r2)<1
- (1-r2) < 1/10
- r2 > 9/10
- Ans: The R-square of two independent variable should larger than 0.8 to make VIF>5

Q3

(a)Let’s add three separate regression lines on the scatterplot, one for each of the origins:

```
origin_colors = c("blue", "darkgreen", "red")
with(cars_log, plot(log.weight., log.mpg., pch=origin, col=origin_colors[origin]))

cars_us <- subset(cars_log, origin==1)
wt_regr_us <- lm(log.mpg. ~ log.weight., data=cars_us)
abline(wt_regr_us, col=origin_colors[1], lwd=2)

cars_eu <- subset(cars_log, origin==2)
wt_regr_eu <- lm(log.mpg. ~ log.weight., data=cars_eu)
abline(wt_regr_eu, col=origin_colors[2], lwd=2)

cars_jp <- subset(cars_log, origin==3)
wt_regr_jp <- lm(log.mpg. ~ log.weight., data=cars_jp)
abline(wt_regr_jp, col=origin_colors[3], lwd=2)
```



(b)[not graded] Do cars from different origins appear to have different weight vs. mpg relationships?

It seems like the weight vs. mpg relationships between US and Europe and Japan appear to be slioghtly different, as the scatter plot shows that the distribution and the slope in the group 2 and group 3 are close, while they are slightly different with group 1.