HW13

106022103

2021/5/22

Assist

- 106000199
  - Discussion about the interaction.

Set up

import libary

```
library(ggplot2)
require(qqplotr)
library(plyr)
library(gridExtra)
library(ggcorrplot)
library(magrittr)
library(ggpubr)
library(car)
```

Read file

```
cars <- read.table("data/auto-data.txt", header=FALSE, na.strings = "?")
names(cars) <- c("mpg", "cylinders", "displacement", "horsepower", "weight",
                 "acceleration", "model_year", "origin", "car_name")
cars <- cars[complete.cases(cars), ] # remove missing value
cars[,'origin']<-factor(cars[,'origin']) # convert to factor
cars[,'car_name']<-factor(cars[,'car_name']) # convert to factor
cars_value <- cars[,-9] # drop the class data
cars_log <- with(cars_value, data.frame(log(mpg), log(cylinders), log(displacement), log(horsepower), lo
```

Q1 Let's visualize how weight and acceleration are related to mpg.

a. Let's visualize how weight might moderate the relationship between acceleration and mpg:

i. Create two subsets of your data, one for light-weight cars (less than mean weight) and one for heavy cars (higher than the mean weight)

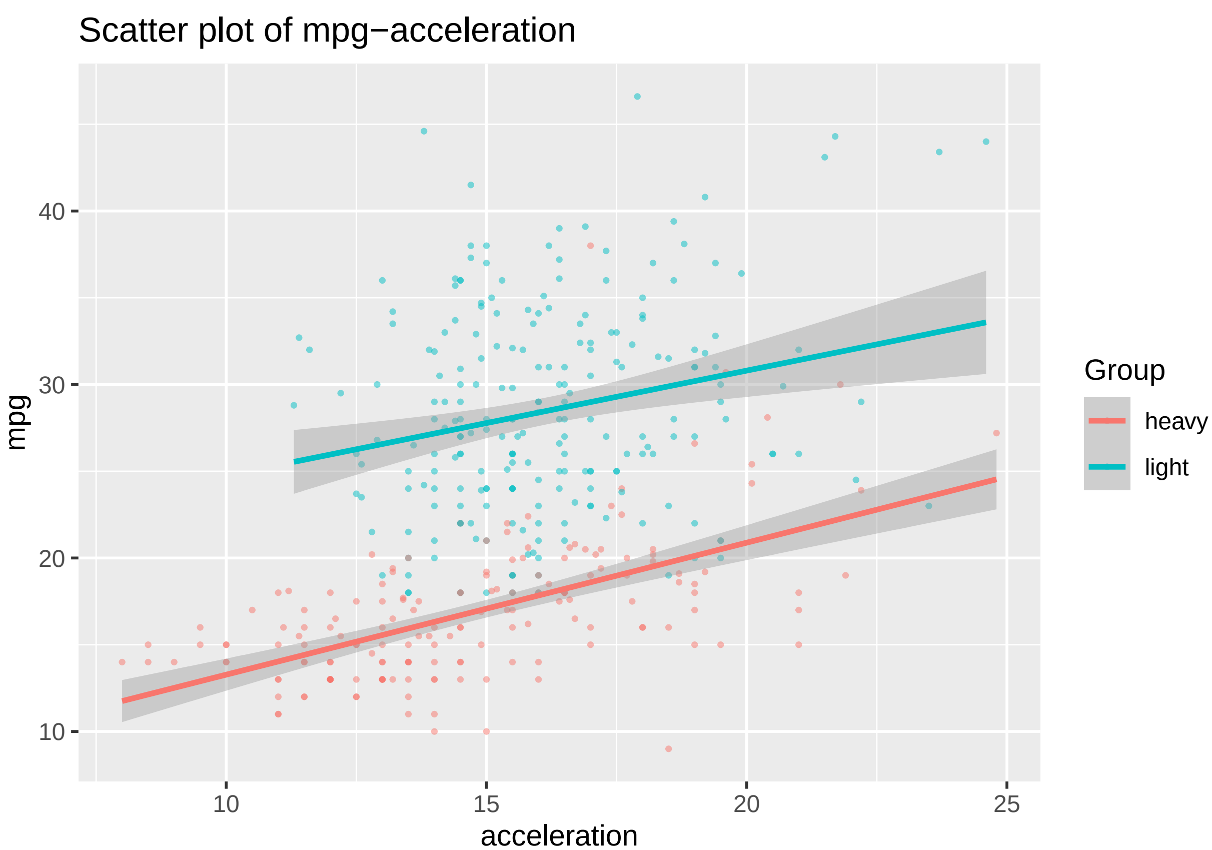
```
weight_mean <- mean(cars$weight)
weight_mean_log <- mean(cars_log$log.weight.)
light_weight_cars <- cars[cars$weight<weight_mean,]
heavy_weight_cars <- cars[cars$weight>weight_mean,]
light_weight_cars_log <- cars_log[cars_log$log.weight. < weight_mean_log,]
heavy_weight_cars_log <- cars_log[cars_log$log.weight. > weight_mean_log,]
```

```
cars$Group <- "No"
cars[cars$weight > weight_mean,]$Group = "heavy"
cars[cars$weight < weight_mean,]$Group = "light"

cars_log$Group <- "No"
cars_log[cars_log$log.weight. > weight_mean_log,]$Group = "heavy"
cars_log[cars_log$log.weight. < weight_mean_log,]$Group = "light"
```

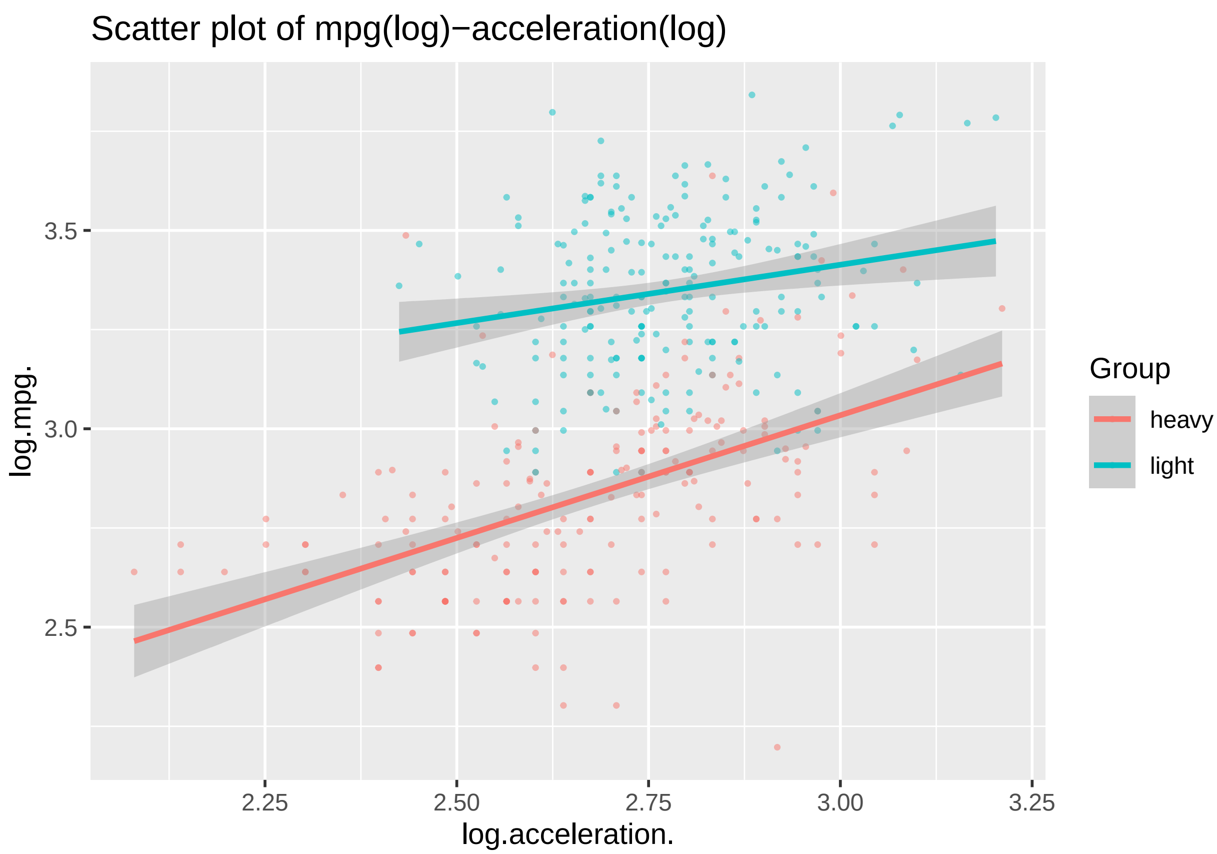
ii.~iii. Create a single scatter plot of `acceleration` vs. `mpg`, with different colors and/or shapes for light versus heavy cars

```
p <- ggplot(data = cars, mapping = aes(x=acceleration, y=mpg,color=Group)) +
    geom_point(size = 0.5,alpha=0.5) +
    geom_smooth(method=lm)+
    labs(title = paste("Scatter plot of","mpg-acceleration"))
p
```



origin scale

```
p <- ggplot(data = cars_log, mapping = aes(x=log.acceleration., y=log.mpg., color=Group)) +
    geom_point(size = 0.5,alpha=0.5) +
    geom_smooth(method=lm)+
    labs(title = paste("Scatter plot of","mpg(log)-acceleration(log)"))
p
```



log scale

b. Report the full summaries of two separate regressions for light and heavy cars where

log.mpg. is dependent on log.weight., log.acceleration., model_year and origin

```
summary(lm(log.mpg. ~ log.weight. + log.acceleration. + model_year + origin,  data = light_weight_cars_l
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     origin, data = light_weight_cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36684 -0.06688  0.00620  0.06448  0.31576
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        6.817512   0.606080  11.249   <2e-16 ***
## log.weight.       -0.820783   0.066717 -12.302   <2e-16 ***
## log.acceleration.  0.111434   0.058800   1.895   0.0595 .
## model_year         0.033109   0.002096  15.798   <2e-16 ***
## origin2            0.039695   0.021455   1.850   0.0658 .
## origin3            0.020798   0.019458   1.069   0.2864
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1109 on 196 degrees of freedom
```

```
## Multiple R-squared:  0.7034, Adjusted R-squared:  0.6958
## F-statistic: 92.97 on 5 and 196 DF,  p-value: < 2.2e-16
```

```
summary(lm(log.mpg. ~ log.weight. + log.acceleration. + model_year + origin,  data = heavy_weight_cars_l
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     origin, data = heavy_weight_cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37106 -0.07150  0.00276  0.06702  0.42505
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.096619   0.690120  10.283  < 2e-16 ***
## log.weight.       -0.824266   0.069657 -11.833  < 2e-16 ***
## log.acceleration.  0.031170   0.056250   0.554  0.58017
## model_year         0.032086   0.003325   9.649  < 2e-16 ***
## origin2            0.098291   0.034250   2.870  0.00459 **
## origin3            0.061596   0.066222   0.930  0.35351
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.122 on 184 degrees of freedom
## Multiple R-squared:  0.754,  Adjusted R-squared:  0.7473
## F-statistic: 112.8 on 5 and 184 DF,  p-value: < 2.2e-16
```

c. (not graded) Using your intuition only: What do you observe about light versus heavy cars so far?

ANSWER: Lighter cars often have higher `mpg` at the same `acceleration` level.

Q2 Using the fully transformed dataset from above (cars_log), to test whether we have moderation.

a. (not graded) Between weight and acceleration ability, use your intuition and experience to state which variable might be a moderating versus independent variable, in affecting mileage.

ANSWER: I think `acceleration` might be a moderating versus independent variable, in affecting `mpg`.

b. Use various regression models to model the possible moderation on log.mpg.

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

i. Report a regression without any interaction terms

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin), data = cars_log)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -0.38259 -0.07054  0.00401  0.06696  0.39798
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         7.410974   0.316806  23.393  < 2e-16 ***
## log.weight.        -0.875499   0.029086 -30.101  < 2e-16 ***
## log.acceleration.   0.054377   0.037132   1.464  0.14389
## model_year          0.032787   0.001731  18.937  < 2e-16 ***
## factor(origin)2     0.056111   0.018241   3.076  0.00225 **
## factor(origin)3     0.031937   0.018506   1.726  0.08519 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1163 on 386 degrees of freedom
## Multiple R-squared:  0.8845, Adjusted R-squared:  0.883
## F-statistic: 591.1 on 5 and 386 DF,  p-value: < 2.2e-16
```

```
regr_weight_acc <- lm(log.mpg. ~ log.weight. + log.acceleration. + log.weight.*log.acceleration.+
                        model_year + origin, data = cars_log)
summary(regr_weight_acc)
```

ii. Report a regression with a raw interaction between weight and acceleration

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + log.weight. *
##     log.acceleration. + model_year + origin, data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37795 -0.06904  0.00367  0.06946  0.39735
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    1.084310   2.780784   0.390  0.69680
## log.weight.                   -0.097340   0.341054  -0.285  0.77548
## log.acceleration.              2.357003   1.006243   2.342  0.01967 *
## model_year                     0.033730   0.001771  19.051  < 2e-16 ***
## origin2                        0.056935   0.018145   3.138  0.00183 **
## origin3                        0.027512   0.018506   1.487  0.13793
## log.weight.:log.acceleration. -0.286724   0.125213  -2.290  0.02257 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1157 on 385 degrees of freedom
## Multiple R-squared:  0.886,  Adjusted R-squared:  0.8843
## F-statistic: 498.9 on 6 and 385 DF,  p-value: < 2.2e-16
```

```
mc_log_weight <- scale(cars_log$log.weight., center = TRUE, scale = FALSE)
mc_log_acc <- scale(cars_log$log.acceleration., center = TRUE, scale = FALSE)
mc_log_mpg <- scale(cars_log$log.mpg., center = TRUE, scale = FALSE)
```

```
summary(lm(mc_log_mpg ~ mc_log_acc + mc_log_weight + mc_log_acc * mc_log_weight+ model_year + origin, da
```

iii. Report a regression with a mean-centered interaction term

```
##
## Call:
## lm(formula = mc_log_mpg ~ mc_log_acc + mc_log_weight + mc_log_acc *
##     mc_log_weight + model_year + origin, data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37795 -0.06904  0.00367  0.06946  0.39735
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -2.584407   0.135617 -19.057  < 2e-16 ***
## mc_log_acc                0.074918   0.038003   1.971  0.04940 *
## mc_log_weight            -0.879375   0.028977 -30.348  < 2e-16 ***
## model_year                0.033730   0.001771  19.051  < 2e-16 ***
## origin2                   0.056935   0.018145   3.138  0.00183 **
## origin3                   0.027512   0.018506   1.487  0.13793
## mc_log_acc:mc_log_weight -0.286724   0.125213  -2.290  0.02257 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1157 on 385 degrees of freedom
## Multiple R-squared:  0.886,  Adjusted R-squared:  0.8843
## F-statistic: 498.9 on 6 and 385 DF,  p-value: < 2.2e-16
```

```
inter <- cars_log$log.weight. * cars_log$log.acceleration.
inter_regr <- lm(inter ~ cars_log$log.weight. + cars_log$log.acceleration.)
cor(inter_regr$residuals, cars_log$log.weight.)
```

iv. Report a regression with an orthogonalized interaction term

```
## [1] -1.347702e-16
```

```
cor(inter_regr$residuals, cars_log$log.acceleration.)
```

```
## [1] 4.089779e-17
```

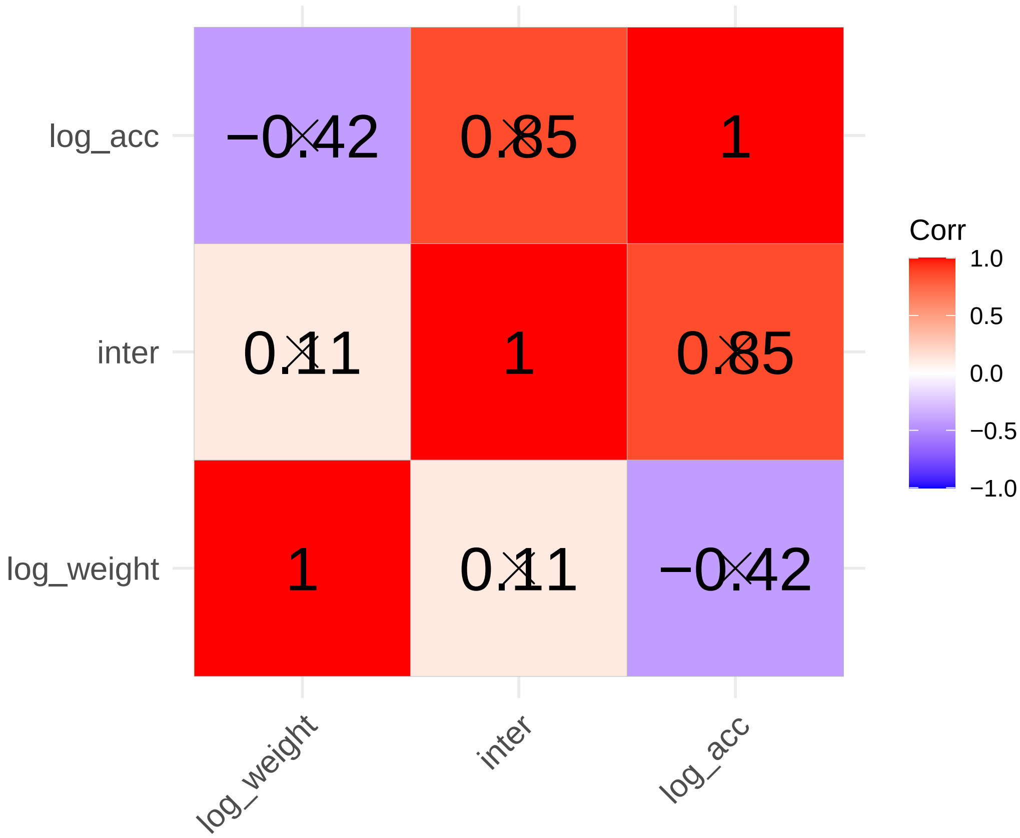
```
summary(lm(data = cars_log, log.mpg. ~ log.weight. + log.acceleration. + inter_regr$residuals+ model_yea
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + inter_regr$residuals +
##     model_year + origin, data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37795 -0.06904  0.00367  0.06946  0.39735
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           7.359447   0.315882  23.298  < 2e-16 ***
```

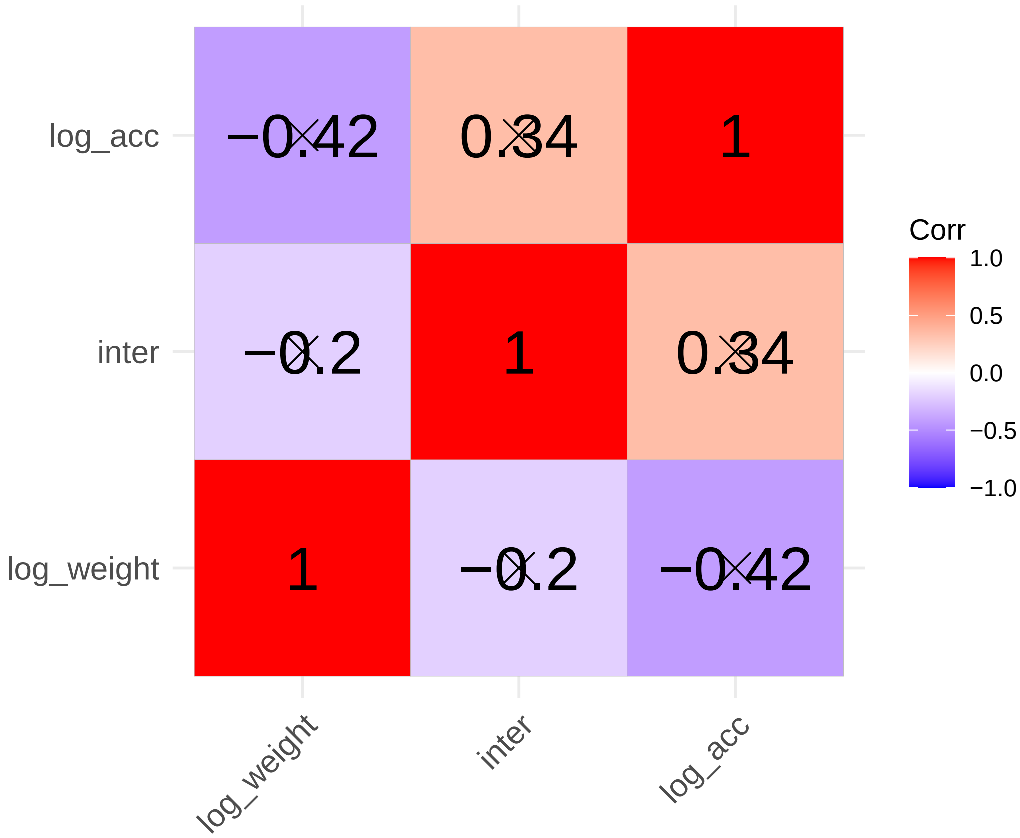
```
## log.weight.          -0.876082   0.028928 -30.285  < 2e-16 ***
## log.acceleration.     0.048960   0.037005   1.323  0.18659
## inter_regr$residuals -0.286724   0.125213  -2.290  0.02257 *
## model_year            0.033730   0.001771  19.051  < 2e-16 ***
## origin2               0.056935   0.018145   3.138  0.00183 **
## origin3               0.027512   0.018506   1.487  0.13793
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1157 on 385 degrees of freedom
## Multiple R-squared:  0.886,  Adjusted R-squared:  0.8843
## F-statistic: 498.9 on 6 and 385 DF,  p-value: < 2.2e-16
```

c. For each of the interaction term strategies above (raw, mean-centered, orthogonalized) what is the correlation between that interaction term and the two variables that you multiplied together?

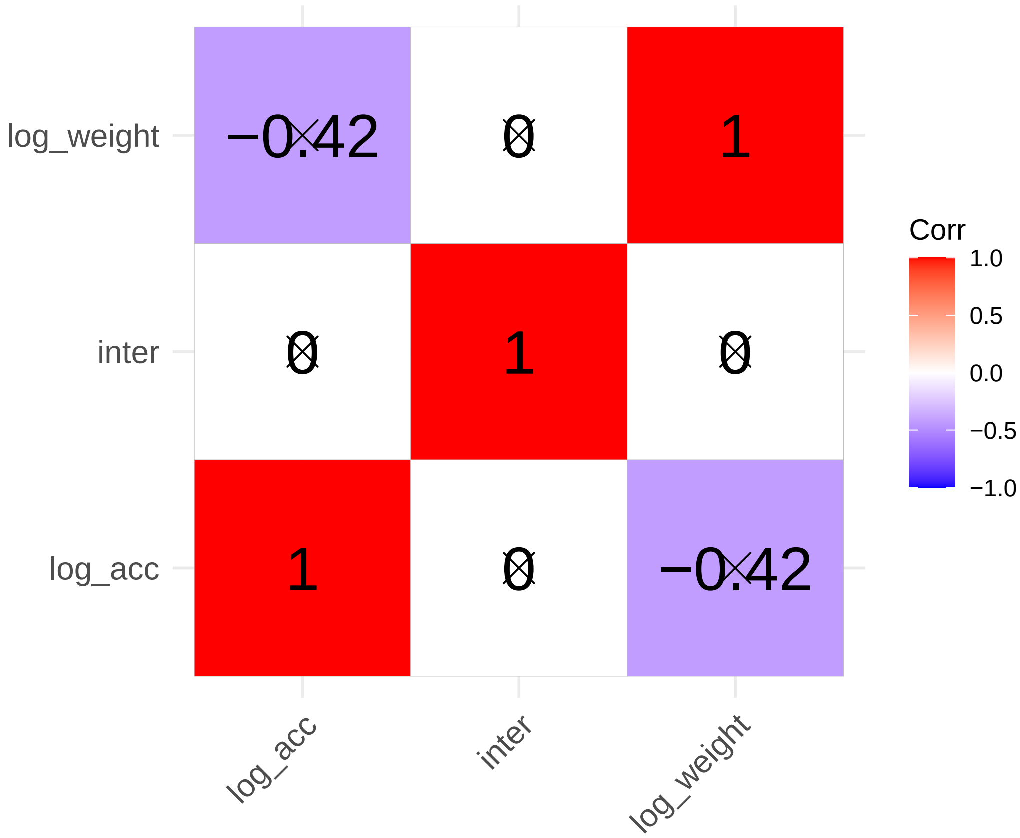
```r
# raw
inter_1 <- cars_log$log.weight. * cars_log$log.acceleration.
cor_raw <- round(cor(cbind(inter_1, cars_log$log.weight., cars_log$log.acceleration.)),2)
p.raw_mat <- cor_pmat(cor(cbind(inter_1, cars_log$log.weight., cars_log$log.acceleration.)))
colnames(cor_raw) <-  c("inter", "log_weight", "log_acc")
rownames(cor_raw) <-  c("inter", "log_weight", "log_acc")
colnames(p.raw_mat) <-  c("inter", "log_weight", "log_acc")
rownames(p.raw_mat) <-  c("inter", "log_weight", "log_acc")
ggcorrplot(t(cor_raw), hc.order = TRUE,
  type = "full", p.mat = t(p.raw_mat), lab = TRUE,lab_size = 8)
```




```r
# mean-centered
inter_2 <- mc_log_weight * mc_log_acc
cor_raw <- round(cor(cbind(inter_2, mc_log_weight, mc_log_acc)),2)
p.raw_mat <- cor_pmat(cor(cbind(inter_2, mc_log_weight, mc_log_acc)))
colnames(cor_raw) <-  c("inter", "log_weight", "log_acc")
rownames(cor_raw) <-  c("inter", "log_weight", "log_acc")
colnames(p.raw_mat) <-  c("inter", "log_weight", "log_acc")
rownames(p.raw_mat) <-  c("inter", "log_weight", "log_acc")
ggcorrplot(t(cor_raw), hc.order = TRUE,
  type = "full", p.mat = t(p.raw_mat), lab = TRUE,lab_size = 8)
```




```
# orthogonalized
inter_3 <- inter_regr$residuals
cor_raw <- round(cor(cbind(inter_3, cars_log$log.weight., cars_log$log.acceleration.)),2)
p.raw_mat <- cor_pmat(cor(cbind(inter_3, cars_log$log.weight., cars_log$log.acceleration.)))
colnames(cor_raw) <-  c("inter", "log_weight", "log_acc")
rownames(cor_raw) <-  c("inter", "log_weight", "log_acc")
colnames(p.raw_mat) <-  c("inter", "log_weight", "log_acc")
rownames(p.raw_mat) <-  c("inter", "log_weight", "log_acc")
ggcorrplot(t(cor_raw), hc.order = TRUE,
  type = "full", p.mat = t(p.raw_mat), lab = TRUE,lab_size = 8)
```



Reference Link

- ggplot2 scatter plots
- Multi-collinearity, Variance Inflationand Orthogonalization in Regression