BACS-hw12-107070004

Let's take another look at interactions in our cars dataset. For this week, let's only use the following data:

1. mpg: miles-per-gallon (dependent variable)
2. weight: weight of car
3. acceleration: acceleration ability of car (seconds to achieve 0-60mph)
4. model_year: year model was released
5. origin: place car was designed (1: USA, 2: Europe, 3: Japan)
6. cylinders: cylinders in engine (only used in Question 3)

Create a data.frame called cars_log with log-transformed columns for mpg, weight, and acceleration (model_year and origin don't have to be transformed)

Question 1) Let's visualize how weight and acceleration are related to mpg.

```
cars <- read.table("auto-data.txt", header=FALSE, 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(weight), log(acceleration)
                                  , log(cylinders), model_year, origin))
head(cars_log, 10)
```

```
##    log.mpg. log.weight. log.acceleration. log.cylinders. model_year origin
## 1  2.890372    8.161660          2.484907       2.079442         70      1
## 2  2.708050    8.214194          2.442347       2.079442         70      1
## 3  2.890372    8.142063          2.397895       2.079442         70      1
## 4  2.772589    8.141190          2.484907       2.079442         70      1
## 5  2.833213    8.145840          2.351375       2.079442         70      1
## 6  2.708050    8.375860          2.302585       2.079442         70      1
## 7  2.639057    8.378850          2.197225       2.079442         70      1
## 8  2.639057    8.369157          2.140066       2.079442         70      1
## 9  2.639057    8.395026          2.302585       2.079442         70      1
## 10 2.708050    8.255828          2.140066       2.079442         70      1
```

(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)

```r
cars_log_sorted <- cars_log[order(cars_log$log.weight.),]
head(cars_log_sorted, 5)
```

```
##     log.mpg. log.weight. log.acceleration. log.cylinders. model_year origin
## 55  3.555348    7.385851          2.890372       1.386294         71      3
## 145 3.433987    7.407924          2.803360       1.386294         74      3
## 344 3.666122    7.470224          2.827314       1.386294         81      3
## 346 3.558201    7.473069          2.778819       1.386294         81      3
## 54  3.433987    7.480428          2.944439       1.386294         71      3
```

```r
len <- nrow(cars_log_sorted)
len
```

```
## [1] 398
```

```r
heavy_cars <- cars_log_sorted[c(0:len/2),]
tail(heavy_cars, 5)
```

```
##       log.mpg. log.weight. log.acceleration. log.cylinders. model_year origin
## 394   3.295837    7.933797          2.747271       1.386294         82      1
## 394.1 3.295837    7.933797          2.747271       1.386294         82      1
## 277   3.072693    7.935587          2.753661       1.386294         78      2
## 277.1 3.072693    7.935587          2.753661       1.386294         78      2
## 324   3.328627    7.937375          2.667228       1.386294         80      1
```

```r
light_cars <- cars_log_sorted[c(len/2+1:len),]
head(light_cars, 5)
```

```
##     log.mpg. log.weight. log.acceleration. log.cylinders. model_year origin
## 124 2.995732    7.939872          2.602690       1.791759         73      3
## 242 3.091042    7.942718          2.674149       1.791759         77      3
## 275 3.010621    7.948032          2.766319       1.609438         78      2
## 16  3.091042    7.949091          2.740840       1.791759         70      1
## 390 3.091042    7.949797          2.687847       1.791759         82      1
```

HINT: consider carefully how you compare log weights to mean weight

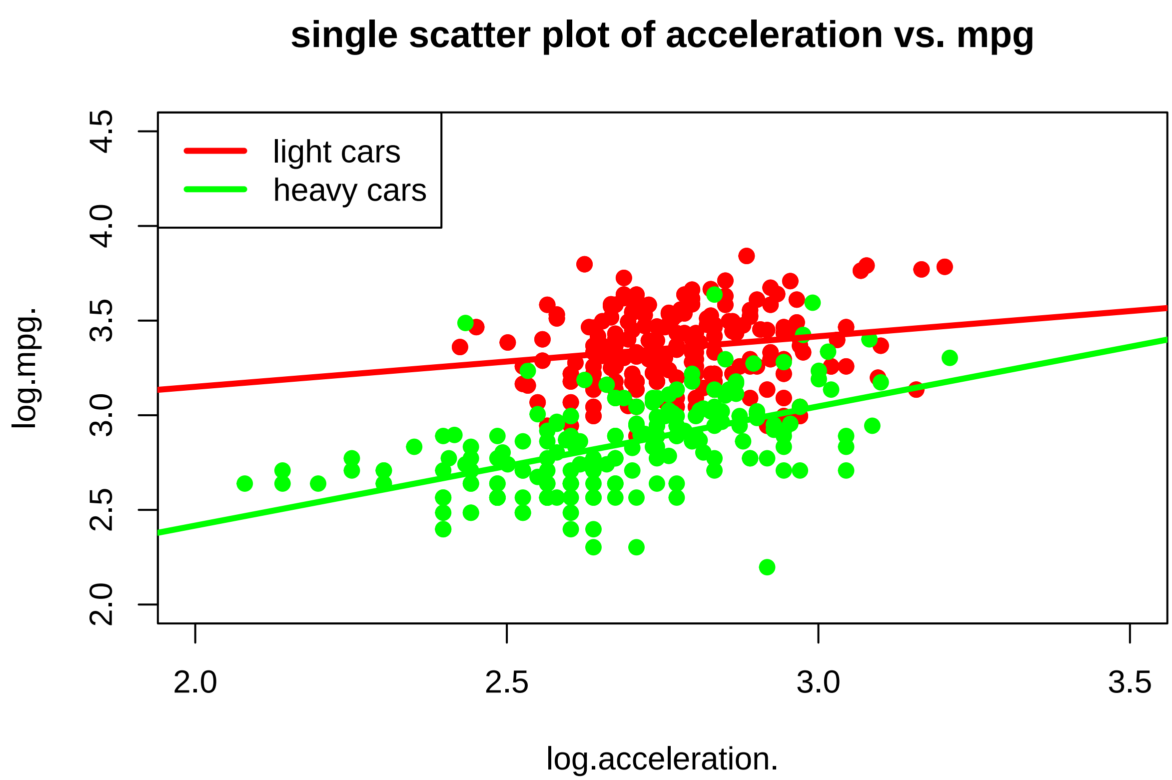
(ii) Create a single scatter plot of acceleration vs. mpg, with different colors and/or shapes for light versus heavy cars

plot is show in (iii)

(iii) Draw two slopes of acceleration-vs-mpg over the scatter plot: one slope for light cars and one slope for heavy cars (distinguish them by appearance)

```
plot(heavy_cars$log.acceleration., heavy_cars$log.mpg.
     , main="single scatter plot of acceleration vs. mpg"
     , xlab = "log.acceleration.", ylab = "log.mpg."
     , xlim=c(2, 3.5), ylim=c(2, 4.5)
     , col = "red", pch = 19)
points(light_cars$log.acceleration., light_cars$log.mpg.
     , col = "green", pch = 19)
legend("topleft", legend = c("light cars", "heavy cars"),
        lwd = 3, lty = c(1, 1), col = c("red", "green"))

regr_heavy_a <- lm(log.mpg. ~ log.acceleration.
              , data=heavy_cars, na.action=na.exclude)
regr_light_a <- lm(log.mpg. ~ log.acceleration.
              , data=light_cars, na.action=na.exclude)
abline(regr_heavy_a, col = "red", lwd = 3)
abline(regr_light_a, col = "green", lwd = 3)
```



(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

```
regr_heavy_b <- lm(log.mpg. ~ log.weight. + log.acceleration.
                  + model_year + factor(origin)
```

```r
                  , data=heavy_cars, na.action=na.exclude)
regr_light_b <- lm(log.mpg. ~ log.weight. + log.acceleration.
                  + model_year + factor(origin)
                  , data=light_cars, na.action=na.exclude)
summary(regr_heavy_b)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin), data = heavy_cars, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36266 -0.07107  0.00620  0.06230  0.31178
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        6.478000   0.429950  15.067  < 2e-16 ***
## log.weight.       -0.788980   0.047858 -16.486  < 2e-16 ***
## log.acceleration.  0.112449   0.040602   2.770 0.005881 **
## model_year         0.034264   0.001445  23.705  < 2e-16 ***
## factor(origin)2    0.050966   0.014727   3.461 0.000598 ***
## factor(origin)3    0.025503   0.013594   1.876 0.061382 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1078 on 391 degrees of freedom
## Multiple R-squared:  0.7092, Adjusted R-squared:  0.7055
## F-statistic: 190.7 on 5 and 391 DF,  p-value: < 2.2e-16
```

```r
summary(regr_light_b)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin), data = light_cars, na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36824 -0.06814  0.00337  0.06525  0.43452
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.207739   0.663252  10.867   <2e-16 ***
## log.weight.       -0.826540   0.066570 -12.416   <2e-16 ***
## log.acceleration.  0.049780   0.054979   0.905   0.3664
## model_year         0.030195   0.003141   9.614   <2e-16 ***
## factor(origin)2    0.084469   0.033071   2.554   0.0114 *
## factor(origin)3    0.048432   0.054382   0.891   0.3743
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.1219 on 193 degrees of freedom
##   (199 observations deleted due to missingness)
## Multiple R-squared:  0.7552, Adjusted R-squared:  0.7489
## F-statistic: 119.1 on 5 and 193 DF,  p-value: < 2.2e-16
```

(c) (not graded)

Question 2) Let's tackle multicollinearity next. Consider the regression model:

(a) (not graded)

(b) Use various regression models to model the possible moderation on log.mpg.: (use log.weight., log.acceleration., model_year and origin as independent variables)

(i) Report a regression without any interaction terms

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

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

(ii) Report a regression with an interaction between weight and acceleration

```r
regr_wa <- lm(log.mpg. ~ log.weight. + log.acceleration.
                  + model_year + factor(origin)
                  + log.weight.*log.acceleration.
              , data=cars_log, na.action=na.exclude)
summary(regr_wa)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin) + log.weight. * log.acceleration., data = cars_log,
##     na.action = na.exclude)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37807 -0.06868  0.00463  0.06891  0.39857
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     1.089642   2.752872   0.396  0.69245
## log.weight.                    -0.096632   0.337637  -0.286  0.77488
## log.acceleration.               2.357574   0.995349   2.369  0.01834 *
## model_year                      0.033685   0.001735  19.411  < 2e-16 ***
## factor(origin)2                 0.058737   0.017789   3.302  0.00105 **
## factor(origin)3                 0.028179   0.018266   1.543  0.12370
## log.weight.:log.acceleration. -0.287170   0.123866  -2.318  0.02094 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.115 on 391 degrees of freedom
## Multiple R-squared:  0.8871, Adjusted R-squared:  0.8854
## F-statistic: 512.2 on 6 and 391 DF,  p-value: < 2.2e-16
```

(iii) Report a regression with a mean-centered interaction term

```r
cars_log_mc <-  data.frame(scale(cars_log,center=TRUE,scale=FALSE))
regr_mc <- lm(log.mpg. ~ cars_log_mc$log.weight. + cars_log_mc$log.acceleration.
                  + cars_log_mc$model_year + factor(cars_log_mc$origin)
                  + cars_log_mc$log.weight.*cars_log_mc$log.acceleration.
              , data=cars_log)
summary(regr_mc)
```

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

```
##      Min       1Q   Median       3Q      Max
## -0.37807 -0.06868  0.00463  0.06891  0.39857
##
## Coefficients:
##                                                        Estimate Std. Error
## (Intercept)                                            3.079276   0.008442
## cars_log_mc$log.weight.                               -0.880393   0.028585
## cars_log_mc$log.acceleration.                          0.072596   0.037567
## cars_log_mc$model_year                                 0.033685   0.001735
## factor(cars_log_mc$origin)0.42713567839196             0.058737   0.017789
## factor(cars_log_mc$origin)1.42713567839196             0.028179   0.018266
## cars_log_mc$log.weight.:cars_log_mc$log.acceleration. -0.287170   0.123866
##                                                       t value Pr(>|t|)
## (Intercept)                                           364.765  < 2e-16 ***
## cars_log_mc$log.weight.                               -30.799  < 2e-16 ***
## cars_log_mc$log.acceleration.                           1.932  0.05403 .
## cars_log_mc$model_year                                 19.411  < 2e-16 ***
## factor(cars_log_mc$origin)0.42713567839196              3.302  0.00105 **
## factor(cars_log_mc$origin)1.42713567839196              1.543  0.12370
## cars_log_mc$log.weight.:cars_log_mc$log.acceleration.  -2.318  0.02094 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.115 on 391 degrees of freedom
## Multiple R-squared:  0.8871, Adjusted R-squared:  0.8854
## F-statistic: 512.2 on 6 and 391 DF,  p-value: < 2.2e-16
```

(iv) Report a regression with an orthogonalized interaction term

```r
interaction_regr <- lm(log.weight.*log.acceleration. ~ log.weight. + log.acceleration.
                      + model_year + factor(origin)
                      , data=cars_log)
interaction_ortho <- interaction_regr$residuals
regr_ortho <- lm(log.mpg. ~ log.weight. + log.acceleration.
                  + model_year + factor(origin)
                  + interaction_ortho
              , data=cars_log)
summary(regr_ortho)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin) + interaction_ortho, data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37807 -0.06868  0.00463  0.06891  0.39857
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         7.431155   0.310520  23.931  < 2e-16 ***
## log.weight.        -0.876608   0.028538 -30.717  < 2e-16 ***
```

```
## log.acceleration.  0.051508   0.036450   1.413  0.15841
## model_year         0.032734   0.001686  19.413  < 2e-16 ***
## factor(origin)2    0.057991   0.017786   3.260  0.00121 **
## factor(origin)3    0.032333   0.018178   1.779  0.07607 .
## interaction_ortho -0.287170   0.123866  -2.318  0.02094 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.115 on 391 degrees of freedom
## Multiple R-squared:  0.8871, Adjusted R-squared:  0.8854
## F-statistic: 512.2 on 6 and 391 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?

- raw

```
cor(cars_log$log.weight., cars_log$log.weight.*cars_log$log.acceleration.)
```

```
## [1] 0.1083055
```

```
cor(cars_log$log.acceleration., cars_log$log.weight.*cars_log$log.acceleration.)
```

```
## [1] 0.852881
```

- mean-centered

```
cor(cars_log_mc$log.weight., cars_log_mc$log.weight.*cars_log_mc$log.acceleration.)
```

```
## [1] -0.2026948
```

```
cor(cars_log_mc$log.acceleration., cars_log_mc$log.weight.*cars_log_mc$log.acceleration.)
```

```
## [1] 0.3512271
```

- orthogonalized

```
cor(cars_log$log.weight., interaction_ortho)
```

```
## [1] 2.084909e-17
```

```
cor(cars_log$log.acceleration., interaction_ortho)
```

```
## [1] 2.38378e-16
```

Question 3) We saw earlier that the number of cylinders does not seem to directly influence mpg when car weight is also considered. But might cylinders have an indirect relationship with mpg through its weight?

Let's check whether weight mediates the relationship between cylinders and mpg, even when other factors are controlled for. Use log.mpg., log.weight., and log.cylinders as your main variables, and keep log.acceleration., model_year, and origin as control variables (see gray variables in diagram).

(a) Let's try computing the direct effects first:

(i) Model 1: Regress log.weight. over log.cylinders. only (check whether number of cylinders has a significant direct effect on weight)

```r
regr_wc <- lm(log.weight. ~ log.cylinders., data=cars_log)
summary(regr_wc)
```

```
##
## Call:
## lm(formula = log.weight. ~ log.cylinders., data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.35473 -0.09076 -0.00147  0.09316  0.40374
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)     6.60365    0.03712  177.92   <2e-16 ***
## log.cylinders.  0.82012    0.02213   37.06   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1329 on 396 degrees of freedom
## Multiple R-squared:  0.7762, Adjusted R-squared:  0.7757
## F-statistic:  1374 on 1 and 396 DF,  p-value: < 2.2e-16
```

ans: Number of cylinders has a significant direct effect on weight.

(ii) Model 2: Regress log.mpg. over log.weight. and all control variables (check whether weight has a significant direct effect on mpg with other variables statistically controlled?)

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

```
##
```

```
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + log.cylinders. +
##     model_year + factor(origin), data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39866 -0.06888  0.00227  0.06718  0.40603
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.25316    0.34818  20.831   <2e-16 ***
## log.weight.       -0.83628    0.04523 -18.491   <2e-16 ***
## log.acceleration.  0.03997    0.03798   1.053   0.2932
## log.cylinders.    -0.05119    0.04438  -1.153   0.2495
## model_year         0.03240    0.00172  18.838   <2e-16 ***
## factor(origin)2    0.05298    0.01840   2.880   0.0042 **
## factor(origin)3    0.02984    0.01840   1.622   0.1057
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1156 on 391 degrees of freedom
## Multiple R-squared:  0.886,  Adjusted R-squared:  0.8842
## F-statistic: 506.3 on 6 and 391 DF,  p-value: < 2.2e-16
```

ans: Weight has a significant direct effect on mpg with other variables statistically controlled.

(b) What is the indirect effect of cylinders on mpg? (use the product of slopes between model 1 & 2)

```r
regr_wc$coefficients[2]*regr_mw$coefficients[2]
```

```
## log.cylinders.
##     -0.6858539
```

(c) Let's bootstrap for the confidence interval of the indirect effect of cylinders on mpg

(i) Bootstrap regression models 1 & 2, and compute the indirect effect each time: what is its 95% CI of the indirect effect of log.cylinders. on log.mpg.?

```r
boot_mediation <- function(model1, model2, dataset) {
  boot_index <- sample(1:nrow(dataset), replace=TRUE)
  data_boot <- dataset[boot_index, ]
  regr1 <- lm(model1, data_boot)
  regr2 <- lm(model2, data_boot)
  return(regr1$coefficients[2] *regr2$coefficients[2])
}
set.seed(42)
indirect <- replicate(2000, boot_mediation(regr_wc,regr_mw, cars_log))
quantile(indirect, probs=c(0.025, 0.975))
```

```
##       2.5%      97.5%
## -0.7607807 -0.6046015
```

(ii) Show a density plot of the distribution of the 95% CI of the indirect effect

```
plot(density(indirect), main="the distribution of the 95% CI of the indirect effect")
abline(v=quantile(indirect, probs=c(0.025, 0.975)))
```