BACS - HW12

109006241

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
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)

```
cars = read.table("auto-data.txt", 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(weight), log(acceleration),
                                 weight, model_year, origin))
head(cars_log, 5)
```

```
##   log.mpg. log.cylinders. log.weight. log.acceleration. weight model_year
## 1 2.890372       2.079442    8.161660          2.484907   3504         70
## 2 2.708050       2.079442    8.214194          2.442347   3693         70
## 3 2.890372       2.079442    8.142063          2.397895   3436         70
## 4 2.772589       2.079442    8.141190          2.484907   3433         70
## 5 2.833213       2.079442    8.145840          2.351375   3449         70
##   origin
## 1      1
## 2      1
## 3      1
## 4      1
## 5      1
```

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

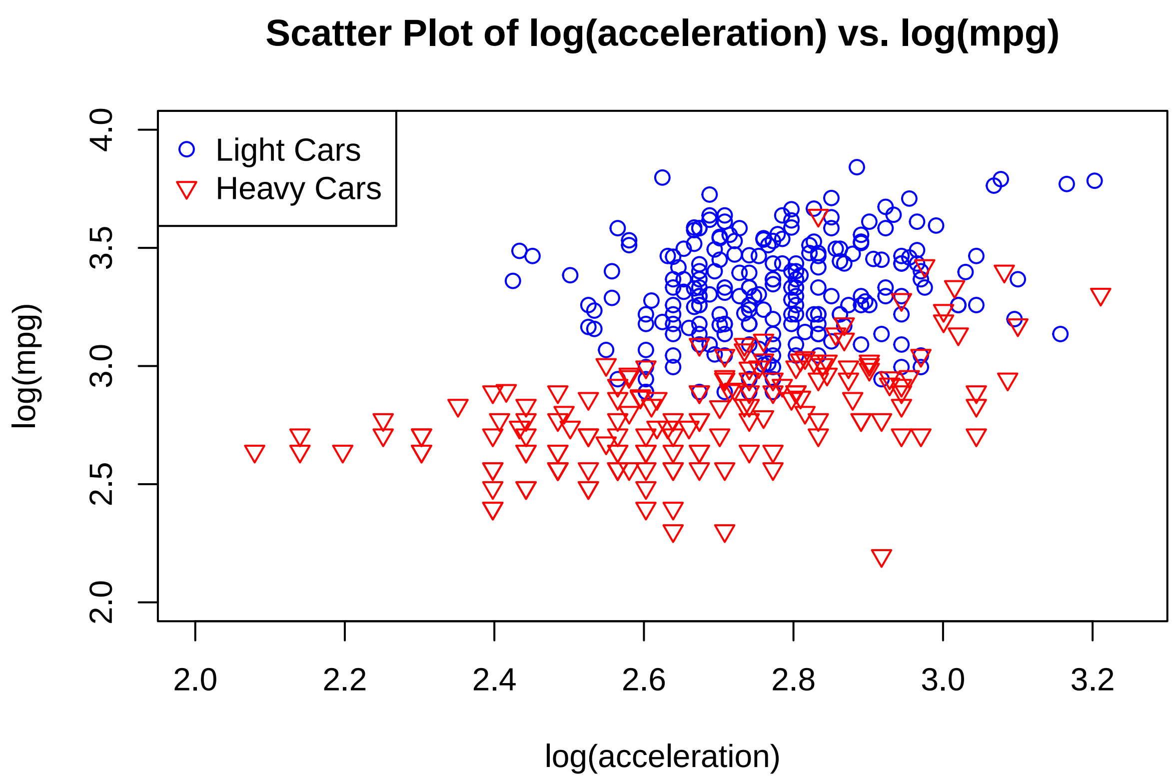
a. Let's visualize how weight might moderate the relationship between acceleration and mpg:

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

```
mean_wt = mean(cars$weight)
cars_log_light = subset(cars_log, weight < mean_wt)
cars_log_heavy = subset(cars_log, weight >= mean_wt)
```

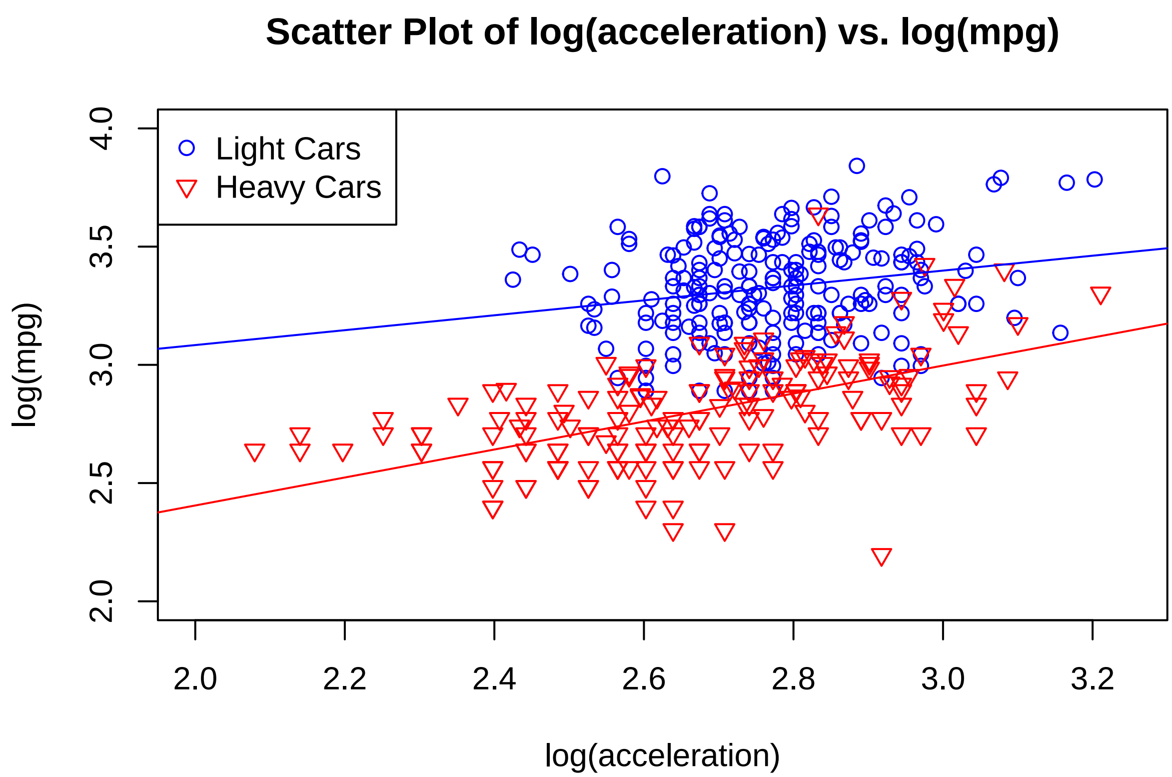
ii. Create a single scatter plot of acceleration vs. mpg, with different colors and/or shapes for light versus heavy cars

```
plot(cars_log_light$log.acceleration., cars_log_light$log.mpg., col="blue", pch=1,
     main="Scatter Plot of log(acceleration) vs. log(mpg)", xlab="log(acceleration)", ylab="log(mpg)",
     xlim=c(2,3.25), ylim=c(2,4))
points(cars_log_heavy$log.acceleration., cars_log_heavy$log.mpg., col="red", pch=6)
legend("topleft", legend=c("Light Cars", "Heavy Cars"), col=c("blue", "red"), pch=c(1, 6))
```



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)

```r
plot(cars_log_light$log.acceleration., cars_log_light$log.mpg., col="blue", pch=1,
     main="Scatter Plot of log(acceleration) vs. log(mpg)", xlab="log(acceleration)", ylab="log(mpg)",
     xlim=c(2,3.25), ylim=c(2,4))
points(cars_log_heavy$log.acceleration., cars_log_heavy$log.mpg., col="red", pch=6)
abline(lm(log.mpg. ~ log.acceleration., cars_log_light), col="blue")
abline(lm(log.mpg. ~ log.acceleration., cars_log_heavy), col="red")
legend("topleft", legend=c("Light Cars", "Heavy Cars"), col=c("blue", "red"), pch=c(1, 6))
```



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

```r
# Regression for light cars
summary(lm(log.mpg. ~ log.weight. + log.acceleration. + model_year + factor(origin), cars_log_light))
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin), data = cars_log_light)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36464 -0.07181  0.00349  0.06273  0.31339
##
## Coefficients:
```

```
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         6.86661    0.52767  13.013   <2e-16 ***
## log.weight.        -0.83437    0.05662 -14.737   <2e-16 ***
## log.acceleration.   0.10956    0.05630   1.946   0.0529 .
## model_year          0.03383    0.00198  17.079   <2e-16 ***
## factor(origin)2     0.05129    0.01980   2.590   0.0102 *
## factor(origin)3     0.02621    0.01846   1.420   0.1571
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1112 on 221 degrees of freedom
## Multiple R-squared:  0.7292, Adjusted R-squared:  0.7231
## F-statistic:   119 on 5 and 221 DF,  p-value: < 2.2e-16
```

```r
# Regression for heavy cars
summary(lm(log.mpg. ~ log.weight. + log.acceleration. + model_year + factor(origin), cars_log_heavy))
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin), data = cars_log_heavy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36811 -0.06937  0.00607  0.06969  0.43736
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.188679   0.759983   9.459  < 2e-16 ***
## log.weight.       -0.822352   0.077206 -10.651  < 2e-16 ***
## log.acceleration.  0.040140   0.057380   0.700   0.4852
## model_year         0.030317   0.003573   8.486 1.14e-14 ***
## factor(origin)2    0.091641   0.040392   2.269   0.0246 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1212 on 166 degrees of freedom
## Multiple R-squared:  0.7179, Adjusted R-squared:  0.7111
## F-statistic: 105.6 on 4 and 166 DF,  p-value: < 2.2e-16
```

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

   The mpg (miles-per-gallon) values for heavier cars are generally lower than the mpg values of lighter cars.

Question 2) Use the transformed dataset from above (cars_log), to test whether we have moderation.

a. (not graded) Considering weight and acceleration, use your intuition and experience to state which of the two variables might be a moderating versus independent variable, in affecting mileage.

I think the weight variable might be a moderating variable.

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

```
summary(lm(log.mpg. ~ log.weight. + log.acceleration. + model_year + factor(origin), cars_log))
```

```
##
## 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
```

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

```
summary(lm(log.mpg. ~ log.weight. + log.acceleration. + model_year + factor(origin) + log.weight.*log.ac
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin) + log.weight. * 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 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

```
log.weight._mc = scale(cars_log$log.weight., center=TRUE, scale=FALSE)
log.acceleration._mc = scale(cars_log$log.acceleration., center=TRUE, scale=FALSE)
summary(lm(cars_log$log.mpg. ~ log.weight._mc + log.acceleration._mc + cars_log$model_year + factor(cars
```

```
##
## Call:
## lm(formula = cars_log$log.mpg. ~ log.weight._mc + log.acceleration._mc +
##     cars_log$model_year + factor(cars_log$origin) + log.weight._mc *
##     log.acceleration._mc)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37807 -0.06868  0.00463  0.06891  0.39857
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          0.518882   0.132944   3.903 0.000112 ***
## log.weight._mc                      -0.880393   0.028585 -30.799  < 2e-16 ***
## log.acceleration._mc                 0.072596   0.037567   1.932 0.054031 .
## cars_log$model_year                  0.033685   0.001735  19.411  < 2e-16 ***
## factor(cars_log$origin)2             0.058737   0.017789   3.302 0.001049 **
## factor(cars_log$origin)3             0.028179   0.018266   1.543 0.123704
## log.weight._mc:log.acceleration._mc -0.287170   0.123866  -2.318 0.020943 *
## ---
## 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

```
interaction_regr = lm(log.weight.*log.acceleration. ~ log.weight. + log.acceleration., cars_log)
interaction_ortho = interaction_regr$residuals
summary(lm(cars_log$log.mpg. ~ cars_log$log.weight. + cars_log$log.acceleration. + cars_log$model_year +
```

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

```
##
## 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)                 7.377176   0.311392  23.691  < 2e-16 ***
## cars_log$log.weight.       -0.876967   0.028539 -30.729  < 2e-16 ***
## cars_log$log.acceleration.  0.046100   0.036524   1.262  0.20764
## cars_log$model_year         0.033685   0.001735  19.411  < 2e-16 ***
## factor(cars_log$origin)2    0.058737   0.017789   3.302  0.00105 **
## factor(cars_log$origin)3    0.028179   0.018266   1.543  0.12370
## 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?

```r
# Raw
cor_raw_weight = cor(cars_log$log.weight.*cars_log$log.acceleration., cars_log$log.weight.)
cor_raw_acceleration = cor(cars_log$log.weight.*cars_log$log.acceleration., cars_log$log.acceleration.)

# Mean-centered
cor_mc_weight = cor(log.weight._mc*log.acceleration._mc, cars_log$log.weight)
cor_mc_acceleration = cor(log.weight._mc*log.acceleration._mc, cars_log$log.acceleration.)

# Orthogonalized
cor_ortho_weight = cor(interaction_ortho, cars_log$log.weight)
cor_ortho_acceleration = cor(interaction_ortho, cars_log$log.acceleration.)

correlations = matrix(
  c(cor_raw_weight,
    cor_raw_acceleration,
    cor_mc_weight,
    cor_mc_acceleration,
    cor_ortho_weight,
    cor_ortho_acceleration),
  nrow=2
)
rownames(correlations) = c("Weight", "Acceleration")
colnames(correlations) = c("Raw", "Mean-centered", "Orthogonalized")

round(correlations, 4)
```

```
##                 Raw Mean-centered Orthogonalized
## Weight       0.1083       -0.2027              0
## Acceleration 0.8529        0.3512              0
```

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)

```
model1 = lm(log.weight. ~ log.cylinders., cars_log)
summary(model1)
```

```
##
## 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
```

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)

```
model2 = lm(log.mpg. ~ log.weight. + log.acceleration. + model_year + factor(origin), cars_log)
summary(model2)
```

```
##
## 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
```

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

```
model1$coefficients["log.cylinders."] * model2$coefficients["log.weight."]
```

```
## log.cylinders.
##     -0.7189275
```

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

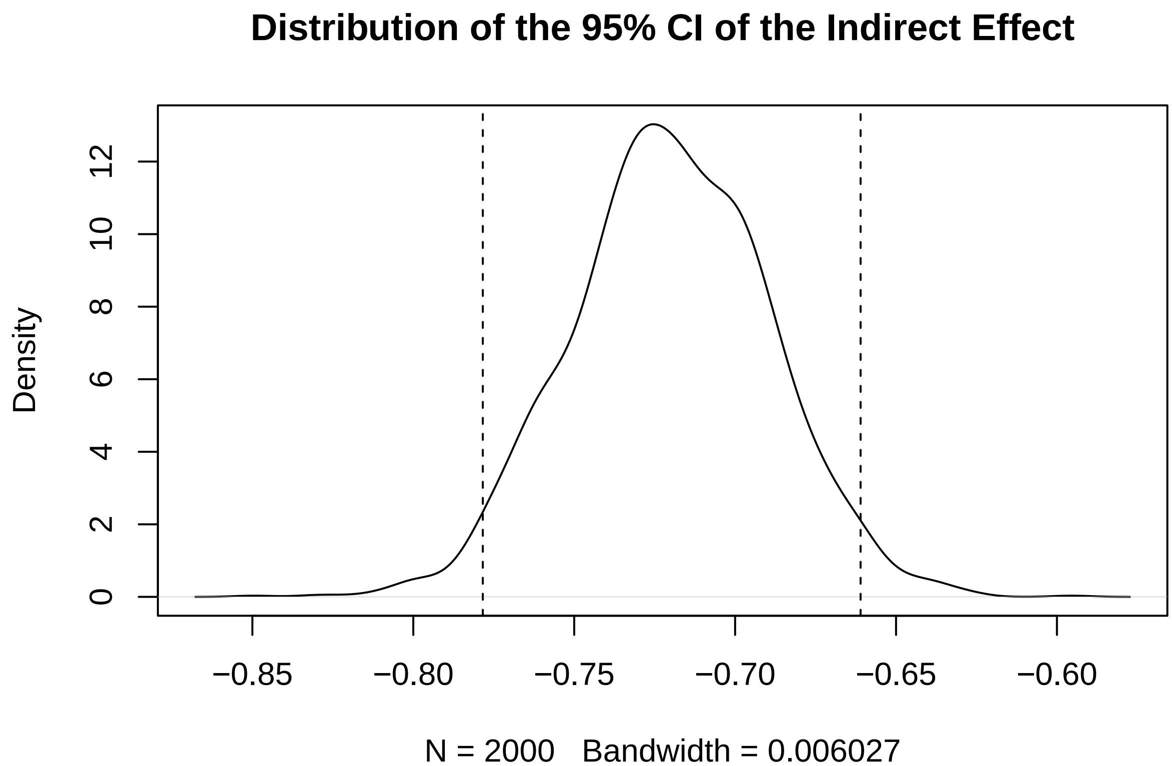
i. Bootstrap regression models 1 & 2, and compute the indirect effect each time: What is its 95% CI of the indirect effect of log.cylinders. on log.mpg.?

```
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(model1, model2, cars_log))

quantile(indirect, probs=c(0.025, 0.975))
```

```
##       2.5%      97.5%
## -0.7784044 -0.6610106
```

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

```
plot(density(indirect), main="Distribution of the 95% CI of the Indirect Effect")
abline(v=quantile(indirect, probs=c(0.025, 0.975)), lty="dashed")
```


Distribution of the 95% CI of the Indirect Effect
Density
0
2
4
6
8
10
12
−0.85
−0.80
−0.75
−0.70
−0.65
−0.60
N = 2000   Bandwidth = 0.006027