bacs_hw10

110071010

2024-04-27

Pre-setting

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", header=FALSE, na.strings = "?")
names(cars) <- c("mpg", "cylinders", "displacement", "horsepower", "wei
ght","acceleration", "model_year", "origin", "car_name")
cars_log <- with(cars, data.frame(log(mpg), log(weight), log(accelerati
on), model_year, origin))
```

Question 1

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

1a

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

i

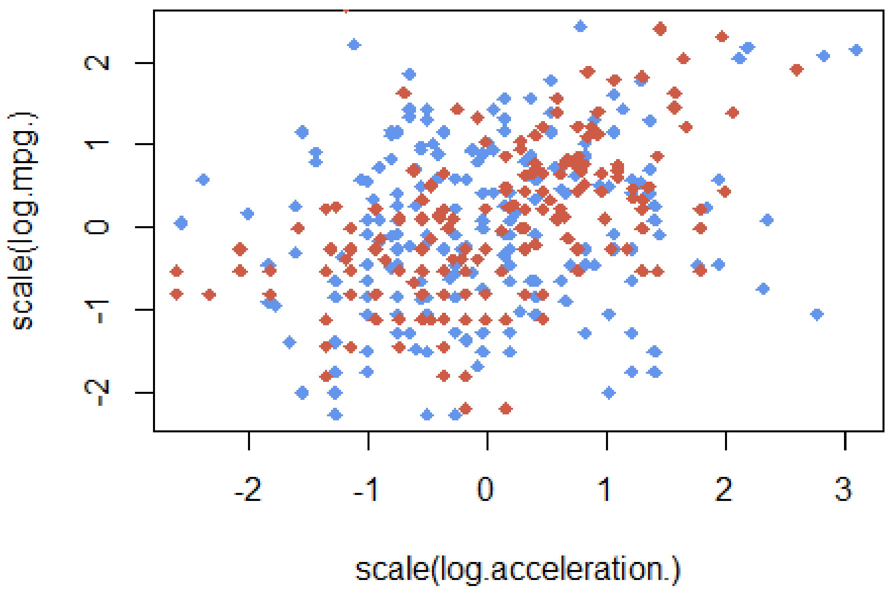
Create two subsets of your data, one for light-weight cars (less than mean weight) and one for heavy cars (higher than the mean weight)

```
mm <- mean(cars_log$log.weight.)
cars_light <- subset(cars_log, log.weight. < mm)
cars_heavy <- subset(cars_log, log.weight. > mm)
```

ii

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

```
weight_colors = c("cornflowerblue", "coral3")
with(cars_light, plot(scale(log.acceleration.), pch = 18, scale(log.mp
g.), col=weight_colors[1]))
with(cars_heavy, points(scale(log.acceleration.), pch = 18, scale(log.m
pg.), col=weight_colors[2]))
```

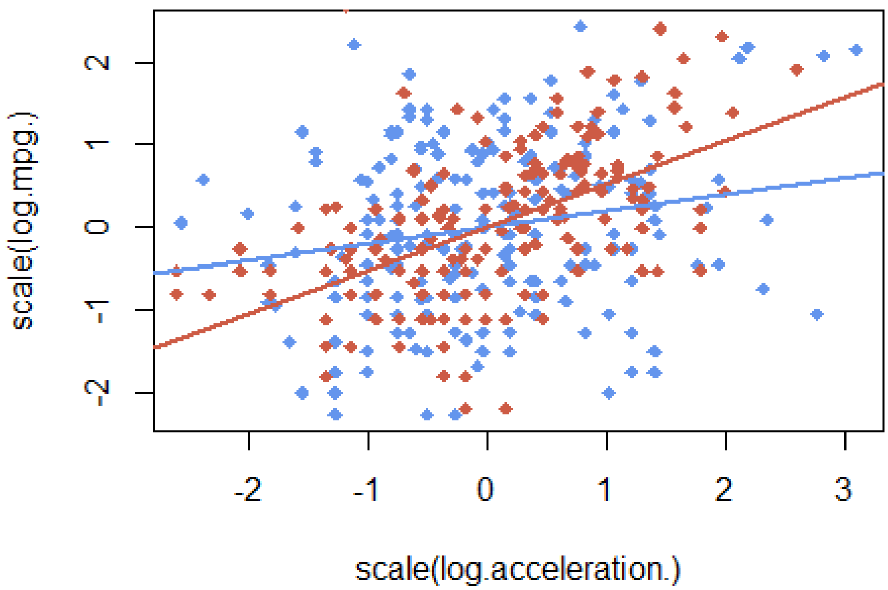


iii

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)

```
weight_colors = c("cornflowerblue", "coral3")
with(cars_light, plot(scale(log.acceleration.), pch = 18, scale(log.mp
g.), col=weight_colors[1]))
with(cars_heavy, points(scale(log.acceleration.), pch = 18, scale(log.m
pg.), col=weight_colors[2]))

with(cars_light, abline(lm(scale(log.acceleration.)~scale(log.mpg.)), c
ol=weight_colors[1], lwd=2))
with(cars_heavy, abline(lm(scale(log.acceleration.)~scale(log.mpg.)), c
ol=weight_colors[2], lwd=2))
```



1b

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

```
# Light cars
summary(lm(log.mpg. ~ log.weight.+log.acceleration.+model_year+factor(o
rigin),data=cars_light))

##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year
 +
##     factor(origin), data = cars_light)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.36590 -0.06612  0.00637  0.06333  0.31513
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        6.809014   0.598446  11.378   <2e-16 ***
## log.weight.       -0.821951   0.065769 -12.497   <2e-16 ***
## log.acceleration.  0.111137   0.058297   1.906   0.0580 .
## model_year         0.033344   0.002049  16.270   <2e-16 ***
## factor(origin)2    0.042309   0.020926   2.022   0.0445 *
## factor(origin)3    0.020923   0.019210   1.089   0.2774
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1102 on 199 degrees of freedom
## Multiple R-squared:  0.7093, Adjusted R-squared:  0.702
## F-statistic:  97.1 on 5 and 199 DF,  p-value: < 2.2e-16
```

```r
# Heavy cars
summary(lm(log.mpg. ~ log.weight.+log.acceleration.+model_year+factor(o
rigin),data=cars_heavy))
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year
 +
##     factor(origin), data = cars_heavy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.37099 -0.07224  0.00150  0.06704  0.42751
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.132892   0.677740  10.525  < 2e-16 ***
## log.weight.       -0.825517   0.068101 -12.122  < 2e-16 ***
## log.acceleration.  0.031221   0.055465   0.563  0.57418
## model_year         0.031735   0.003254   9.752  < 2e-16 ***
## factor(origin)2    0.099027   0.033840   2.926  0.00386 **
## factor(origin)3    0.063148   0.065535   0.964  0.33650
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1212 on 187 degrees of freedom
## Multiple R-squared:  0.7585, Adjusted R-squared:  0.752
## F-statistic: 117.4 on 5 and 187 DF,  p-value: < 2.2e-16
```

1c

Question

Using your intuition only: What do you observe about light versus heavy cars so far?

Answer

Heavy cars display stronger acceleration-mpg relationship.

Question 2

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

2a

Instruction

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.

Answer

Weight is the moderator,and acceleration is the independent variable.

2b

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

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

##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year
 +
##     factor(origin), data = cars_log)
##
## Coefficients:
##       (Intercept)        log.weight.  log.acceleration.         mode
l_year
##           7.43116           -0.87661            0.05151            0.
03273
##   factor(origin)2    factor(origin)3
##           0.05799            0.03233
```

ii

Report a regression with an interaction between weight and acceleration

```
lm(log.mpg. ~ log.weight.+log.acceleration.+model_year+factor(origin)+l
og.weight.* log.acceleration., data=cars_log)

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

```
## Coefficients:
##                     (Intercept)                        log.weight.
##                         1.08964                           -0.09663
##               log.acceleration.                         model_year
##                         2.35757                            0.03368
##                 factor(origin)2                    factor(origin)3
##                         0.05874                            0.02818
## log.weight.:log.acceleration.
##                        -0.28717
```

iii

Report a regression with a mean-centered interaction term

```
log.weight._mean_centered <- scale(cars_log$log.weight., center=TRUE, s
cale=FALSE)
log.acceleration._mean_centered <- scale(cars_log$log.acceleration., ce
nter=TRUE, scale=FALSE)

lm(log.mpg. ~ log.weight._mean_centered +log.acceleration._mean_centere
d +model_year+factor(origin)+log.weight._mean_centered *log.acceleratio
n._mean_centered , data=cars_log)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight._mean_centered + log.acceleration.
_mean_centered +
##     model_year + factor(origin) + log.weight._mean_centered *
##     log.acceleration._mean_centered, data = cars_log)
##
## Coefficients:
##                                                       (Intercept)
##                                                           0.51888
##                                         log.weight._mean_centered
##                                                          -0.88039
##                                   log.acceleration._mean_centered
##                                                           0.07260
##                                                        model_year
##                                                           0.03368
##                                                   factor(origin)2
##                                                           0.05874
##                                                   factor(origin)3
##                                                           0.02818
## log.weight._mean_centered:log.acceleration._mean_centered
##                                                          -0.28717
```

iv

Report a regression with an orthogonalized interaction term

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

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year
 +
##     factor(origin) + interaction_ortho, data = cars_log)
##
## Coefficients:
##       (Intercept)        log.weight.  log.acceleration.          mode
l_year
##           7.37718           -0.87697            0.04610            0.
03368
##   factor(origin)2    factor(origin)3  interaction_ortho
##           0.05874            0.02818           -0.28717
```

2c

Question

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?

Answer

interaction_term – log.weight. : 0.108305532

interaction_term – log.acceleration. : 0.852881042

```r
cor(cbind(interaction_term = cars_log$log.weight * cars_log$log.acceler
ation.,cars_log))
```

```
##                     interaction_term     log.mpg. log.weight. log.acce
leration.
## interaction_term       1.000000000  0.007445392   0.1083055
0.8528810
## log.mpg.               0.007445392  1.000000000  -0.8744686
0.4640533
## log.weight.            0.108305532 -0.874468594   1.0000000        -
0.4256194
## log.acceleration.      0.852881042  0.464053310  -0.4256194
1.0000000
## model_year             0.185345672  0.576342261  -0.2840090
0.3107471
## origin                -0.107848822  0.558329285  -0.6048831
0.2210906
```

```
##                      model_year     origin
## interaction_term   0.1853457 -0.1078488
## log.mpg.           0.5763423  0.5583293
## log.weight.       -0.2840090 -0.6048831
## log.acceleration.  0.3107471  0.2210906
## model_year         1.0000000  0.1806622
## origin             0.1806622  1.0000000
```

Question 3

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.

3a

Let's try computing the direct effects first:

i

Instruction

Model 1: Regress log.weight. over log.cylinders. only

Check whether number of cylinders has a significant direct effect on weight

Observation

Yes. The coefficient of 0.8201 reflects a strong positive relationship.

```
cars_log1 <- with(cars, data.frame(log(mpg), log(weight), log(accelerat
ion),log(cylinders), model_year, origin))
A <- lm(log.weight. ~ log.cylinders., data=cars_log1)
A

##
## Call:
## lm(formula = log.weight. ~ log.cylinders., data = cars_log1)
##
## Coefficients:
##    (Intercept)  log.cylinders.
##         6.6037          0.8201
```

ii

Instruction

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

Observation

Yes. The coefficient of -0.87661 reflects a strong negative relationship.

```
B <- lm(log.mpg. ~ log.weight.+log.acceleration.+model_year+factor(orig
in), data=cars_log1)
B

##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year
 +
##     factor(origin), data = cars_log1)
##
## Coefficients:
##       (Intercept)        log.weight.  log.acceleration.         mode
l_year
##           7.43116           -0.87661            0.05151            0.
03273
##   factor(origin)2    factor(origin)3
##           0.05799            0.03233
```

3b

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

```
(A$coefficients[2]) * (B$coefficients[2])

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

3c

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(model_a, model_b, dataset) {
  boot_index <- sample(1:nrow(dataset), replace=TRUE)
  data_boot <- dataset[boot_index, ]
  regr1 <- lm(model_a, data_boot)
  regr2 <- lm(model_b, data_boot)
  return(regr1$coefficients[2] * regr2$coefficients[2])
}
```

```
set.seed(12341234)
indirect <- replicate(2000, boot_mediation(A, B, cars_log1))
quantile(indirect, probs=c(0.025, 0.975))
```

```
##       2.5%      97.5%
## -0.7823354 -0.6604392
```

ii

Show a density plot of the distribution of the indirect effect, and mark its 95% CI

```
plot(density(indirect),main = "Indirect Effect Distribution", col = "co
rnflowerblue", lwd = 2)
abline(v = quantile(indirect, probs=c(0.025, 0.975)), col = "gray", lwd
 = 2)
```