BACS_HW_Week13_106071041

106071041

2021/5/23

```
cars <- read.table("auto-data.txt", header = F, na.strings = "?")
names(cars) <- c("mpg", "cylinders", "displacement", "horsepower", "weight", "acceleration", "model_year", "origin", "car_na
me")
```

```
cars_log <- with(cars, data.frame(log(mpg), log(weight), log(acceleration), model_year, origin))
```

Question 1 | Visualization

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

(i) Create two subsets of your data, light and heavy

```
light_cars <- subset(cars_log, log.weight. < mean(log.weight.))
```

```
heavy_cars <- subset(cars_log, log.weight. > mean(log.weight.))
```

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

1. Transform the scale to nominal scale

```
weight_mean <- apply(cars_log["log.weight."], 2, mean)
```

```
transform <- as.data.frame(ifelse(cars_log["log.weight."] > weight_mean, 1, 2))
```

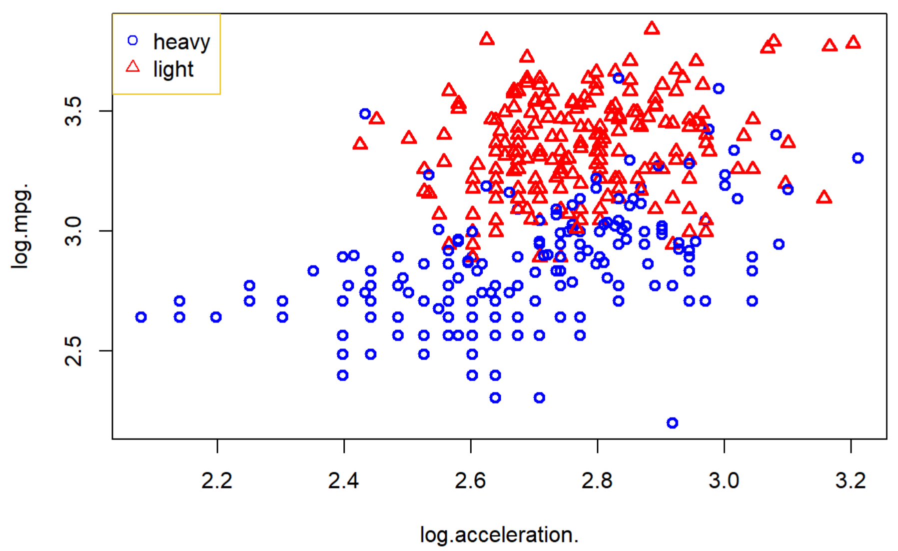
```
transformed <- cbind(cars_log[-c(2)], transform)
```

```
transformed$log.weight.[1:20]
```

```
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2
```

2. Plot

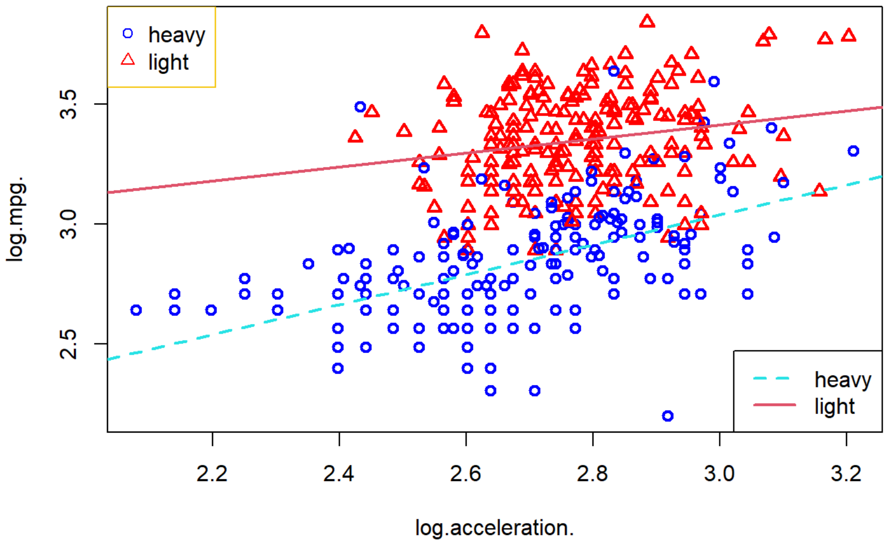
```
weight_colors <- c("blue", "red")
with(transformed, plot(log.acceleration., log.mpg., pch=log.weight., col=weight_colors[log.weight.], lwd = 2, ))
legend("topleft", c("heavy", "light"), col = weight_colors, pch = c(1,2), box.col = 23)
```



(iii) Draw two slopes of acceleration-vs-mpg over the scatter plot: one slope for light cars and one slope for heavy cars (use different line styles)

```
ac_regr_light <- lm(log.mpg. ~ log.acceleration., data = light_cars)
ac_regr_heavy <- lm(log.mpg. ~ log.acceleration., data = heavy_cars)
```

```
weight_colors <- c("blue", "red")
with(transformed, plot(log.acceleration., log.mpg., pch=log.weight., col=weight_colors[log.weight.], lwd = 2, ))
legend("topleft", c("heavy", "light"), col = weight_colors, pch = c(1,2), box.col = 23)
abline(ac_regr_heavy, col = 5, lwd=2, lty="dashed")
abline(ac_regr_light, col = 10, lwd=2)
legend("bottomright", c("heavy", "light"), col = c(5,10), lwd = 2, lty = c(2, 1))
```



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

1. Light Cars

```
with(light_cars, lm(log.mpg. ~ log.weight.+log.acceleration.+model_year+origin));summary(.Last.value)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     origin)
##
## Coefficients:
##       (Intercept)        log.weight.  log.acceleration.         model_year
##          7.019008          -0.840576           0.107638           0.032605
##            origin
##          0.009573
```

```
##           Length Class  Mode
## help_type 0      -none- NULL
```

2. Heavy Cars

```
with(heavy_cars, lm(log.mpg. ~ log.weight.+log.acceleration.+model_year+origin));summary(.Last.value)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     origin)
##
## Coefficients:
##       (Intercept)        log.weight.  log.acceleration.         model_year
##           7.03753           -0.82224            0.05697            0.03089
##            origin
##           0.06414
```

```
##           Length Class  Mode
## help_type 0      -none- NULL
```

c. (not graded) What do you observe about light versus heavy cars so far?

Light cars with larger mpg.

Question 2 | Moderation

a. State which variable might be a moderating versus independent variable, in affecting mileage.

1. Weight: heavy cars needs more fuels to go.
2. Model_year: Maybe in different year will have different revisions on cars.

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

(i) Report a regression without any interaction terms

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

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     origin)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39581 -0.07037  0.00014  0.06984  0.39638
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.539281   0.314707  23.956   <2e-16 ***
## log.weight.       -0.889384   0.028466 -31.243   <2e-16 ***
## log.acceleration.  0.062145   0.036679   1.694   0.0910 .
## model_year         0.032106   0.001690  18.999   <2e-16 ***
## origin             0.018352   0.009165   2.002   0.0459 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1164 on 393 degrees of freedom
## Multiple R-squared:  0.8836, Adjusted R-squared:  0.8825
## F-statistic: 746.1 on 4 and 393 DF,  p-value: < 2.2e-16
```

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

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

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     origin + log.weight. * log.acceleration.)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38147 -0.06870  0.00120  0.06595  0.39570
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    1.773573   2.763699   0.642   0.5214
## log.weight.                   -0.179842   0.339101  -0.530   0.5962
## log.acceleration.              2.162941   1.001155   2.160   0.0313 *
## model_year                     0.032933   0.001728  19.057   <2e-16 ***
## origin                         0.016595   0.009164   1.811   0.0709 .
## log.weight.:log.acceleration. -0.261526   0.124550  -2.100   0.0364 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1159 on 392 degrees of freedom
## Multiple R-squared:  0.8849, Adjusted R-squared:  0.8835
## F-statistic:   603 on 5 and 392 DF,  p-value: < 2.2e-16
```

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

```
mc <- function(x){
  scale(x, center = TRUE, scale = FALSE)
}
```

```
summary(with(cars_log, lm(log.mpg. ~ mc(log.weight.)+mc(log.acceleration.)+model_year+origin+mc(log.weight.)*mc(log.accelera
tion.))))
```

```
##
## Call:
## lm(formula = log.mpg. ~ mc(log.weight.) + mc(log.acceleration.) +
##     model_year + origin + mc(log.weight.) * mc(log.acceleration.))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38147 -0.06870  0.00120  0.06595  0.39570
##
## Coefficients:
##                                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)                            0.566397   0.132258   4.283 2.33e-05 ***
## mc(log.weight.)                       -0.893616   0.028415 -31.448  < 2e-16 ***
## mc(log.acceleration.)                  0.082003   0.037725   2.174   0.0303 *
## model_year                             0.032933   0.001728  19.057  < 2e-16 ***
## origin                                 0.016595   0.009164   1.811   0.0709 .
## mc(log.weight.):mc(log.acceleration.) -0.261526   0.124550  -2.100   0.0364 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1159 on 392 degrees of freedom
## Multiple R-squared:  0.8849, Adjusted R-squared:  0.8835
## F-statistic:   603 on 5 and 392 DF,  p-value: < 2.2e-16
```

(iv) Report a regression with an orthogonalized interaction term

```
acc_x_weight <- cars_log$log.acceleration.*cars_log$log.weight.
interaction_regr <- lm(acc_x_weight ~ cars_log$log.acceleration.+cars_log$log.weight.)
interaction_ortho <- interaction_regr$residuals
```

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

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     origin + interaction_ortho, data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38147 -0.06870  0.00120  0.06595  0.39570
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.499651   0.313919  23.890   <2e-16 ***
## log.weight.       -0.890495   0.028349 -31.412   <2e-16 ***
## log.acceleration.  0.057873   0.036577   1.582   0.1144
## model_year         0.032933   0.001728  19.057   <2e-16 ***
## origin             0.016595   0.009164   1.811   0.0709 .
## interaction_ortho -0.261526   0.124550  -2.100   0.0364 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1159 on 392 degrees of freedom
## Multiple R-squared:  0.8849, Adjusted R-squared:  0.8835
## F-statistic:   603 on 5 and 392 DF,  p-value: < 2.2e-16
```

c. What is the correlation between that interaction term and the two variables that you multiplied together? (raw, mean-centered, orthogonalized)

1. Raw
   log.weight. VS interaction : 0.108
   log.acceleration. VS interaction : 0.853

```
round(with(cars_log, cor(log.weight., log.weight.* log.acceleration.)), 3)
```

```
## [1] 0.108
```

```
round(with(cars_log, cor(log.acceleration., log.weight.* log.acceleration.)), 3)
```

```
## [1] 0.853
```

2. Mean-centered
   log.weight. VS interaction : -0.203
   log.acceleration. VS interaction : 0.351

```
round(cor(mc(cars_log$log.weight.), mc(cars_log$log.weight.)*mc(cars_log$log.acceleration.)),3)
```

```
##        [,1]
## [1,] -0.203
```

```
round(cor(mc(cars_log$log.acceleration.), mc(cars_log$log.weight.)*mc(cars_log$log.acceleration.)),3)
```

```
##       [,1]
## [1,] 0.351
```

3. Orthogonalized
   log.weight. VS interaction : 0
   log.acceleration. VS interaction : 0

```
round(with(cars_log, cor(log.weight. , interaction_ortho)),3)
```

```
## [1] 0
```

```
round(with(cars_log, cor(log.acceleration. , interaction_ortho)),3)
```

```
## [1] 0
```