BIOS7345 Lab 3

Sarah Lotspeich

21 September 2018

With the `Cereal.csv` data (on the GitHub),

```
cereal <- read.csv("~sarahlotspeich/Dropbox/Vanderbilt/Fall 2018/TA (Hakmook)/BIOS7345_Labs/Cereal.csv"
```

use linear regression to regress cereal rating on calories and sugar without an interaction (use an intercept). You may use either `lm()` or `ols()`.

```
mod <- ols(rating ~ calories + sugars, data = cereal)
```

Give the interpretation of the parameter estimates. Try to use the `mod` object created above to dynamically code your coefficient estimates instead.

```
#access model coefficients
mod$coefficients
```

```
##  Intercept   calories     sugars
## 84.1141671 -0.2764392 -1.7193933
```

```
coefficients(mod)
```

```
##  Intercept   calories     sugars
## 84.1141671 -0.2764392 -1.7193933
```

- β_0 : for a cereal with 0 calories and 0 sugars (gross) we expect a rating of 84.1141671
- β_1 : for every 1-unit increase in calories, we expect a -0.2764392 change in rating
- β_2 : for every 1-unit increase in calories, we expect a -1.7193933 change in rating

Pop quiz: what do you call it when you estimate based on values that are far outside the observed (or realistic) range for your predictors?

Does this intercept interpretation make sense?

> No, because there are not any cereals with 0 calories and 0 sugars. This is extrapolation!

Now, let's try centering both `sugars` and `calories` at their mean values

```
cereal %<>% mutate(sugars_centered = sugars - mean(sugars),
                   calories_centered = calories - mean(calories))
```

and refit the model above for `rating ~ calories + sugars`.

```
mod_cent <- ols(rating ~ calories_centered + sugars_centered, data = cereal)
```

Given an interpretation of the intercept estimate.

- β_0: for a cereal with 106.8831169 calories and 6.9220779 sugars we expect a rating of 42.665705

After centering, we should have that $\hat{\beta}_0 = \bar{y}$. Check that this is true.

```
mean(cereal$rating)
```

```
## [1] 42.6657
```

How do the effects of `sugars` and `calories` change with centering?

```
mod$coefficients
```

```
##  Intercept   calories     sugars 
## 84.1141671 -0.2764392 -1.7193933
```

```
mod_cent$coefficients
```

```
##         Intercept calories_centered   sugars_centered 
##        42.6657050        -0.2764392        -1.7193933
```

> Trick question – they don't! As we saw last week, Rencher Theorem 7.3e. tells us that linear transformations of the predictors (in our case centering) does not change the estimated coefficients.

What about the standard errors of these effects? How can we get the standard errors directly from the model object (e.g. not printing it or just typing `mod`)?

```
diag(sqrt(mod$var))
```

```
## Warning in sqrt(mod$var): NaNs produced
```

```
##  Intercept   calories     sugars 
## 5.44513407 0.05754591 0.25225205
```

```
diag(sqrt(mod_cent$var))
```

```
## Warning in sqrt(mod_cent$var): NaNs produced
```

```
##         Intercept calories_centered   sugars_centered 
##        0.92111278        0.05754591        0.25225205
```

> Standard errors also do not change after centering.

Round the covariance matrix of the centered model. What do we notice about the some of the off-diagonal entries?

```
round(mod_cent$var, 5)
```

```
##                   Intercept calories_centered sugars_centered
## Intercept           0.84845           0.00000         0.00000
## calories_centered   0.00000           0.00331        -0.00816
## sugars_centered     0.00000          -0.00816         0.06363
```

What is happening here? Think about the [1,2] element of the covariance matrix.

$$\begin{aligned} Cov(\hat{\beta}_0, \hat{\beta}_1) &= Cov(\bar{y}, \hat{\beta}_1) \text{ since we saw that } \hat{\beta}_0 = \bar{y} above \\ &\overset{def}{=} E[(\bar{y} - E(\bar{y}))(\hat{\beta}_1 - E(\hat{\beta}_1))] \\ &= 0 \end{aligned}$$

Now, divide the centered predictors by their standard deviations (to standardize them)

```
cereal %<>% mutate(calories_stand = calories_centered/sd(calories),
                   sugars_stand = sugars_centered/sd(sugars))
```

and refit the model above for `rating ~ calories + sugars`.

```
mod_stand <- ols(rating ~ calories_stand + sugars_stand, data = cereal)
```

Given an interpretation of the intercept estimate.

- β_0: for a cereal with 106.8831169 calories and 6.9220779 sugars we expect a rating of 42.665705

What do we notice about the intercept for the standardized model?

> It's the same as from the centered model! Why? Well, the centered (e.g. $= 0$) values of the predictors did not change when we divided them by their standard deviations.

Compare the R^2 values for the original, centered, and standardized models. What do you see?

```
round(mod$stats,5)
```

```
##          n Model L.R.       d.f.         R2          g      Sigma
##   77.00000   87.16918    2.00000    0.67763   13.11454    8.08273
```

```
round(mod_cent$stats,5)
```

```
##          n Model L.R.       d.f.         R2          g      Sigma
##   77.00000   87.16918    2.00000    0.67763   13.11454    8.08273
```

```
round(mod_stand$stats,5)
```

```
##          n Model L.R.       d.f.         R2          g      Sigma
##   77.00000   87.16918    2.00000    0.67763   13.11454    8.08273
```

> Linearly transforming our predictors (either centering or standardizing) did not change the model fit from the original.

Show how to obtain the coefficients from the centered/scaled model as a function of the centered/unscaled coefficients and the standard deviation of the predictor variables (e.g. obtain $\boldsymbol{\beta}_{cent,scaled}$ from $\boldsymbol{\beta}_{cent,unscaled}$ and s^2_{xi}).

```
round(mod_cent$coefficients, 5)
```

```
##         Intercept calories_centered   sugars_centered
##          42.66570          -0.27644          -1.71939
```

```
round(mod_stand$coefficients, 5)
```

```
##      Intercept calories_stand   sugars_stand
##       42.66570       -5.38618       -7.64251
```

```
round(mod_cent$coefficients, 5)*c(1, sd(cereal$calories), sd(cereal$sugars))
```

```
##         Intercept calories_centered   sugars_centered
##         42.665700         -5.386190         -7.642491
```

Do the same for the SEs.

```
round(sqrt(diag(mod_cent$var)), 5)
```

```
##         Intercept calories_centered   sugars_centered
##           0.92111           0.05755           0.25225
```

```
round(sqrt(diag(mod_stand$var)), 5)
```

```
##      Intercept calories_stand   sugars_stand
##        0.92111        1.12123        1.12123
```

```
round(sqrt(diag(mod_cent$var)), 5)*c(1, sd(cereal$calories), sd(cereal$sugars))
```

```
##         Intercept calories_centered   sugars_centered
##          0.921110          1.121311          1.121222
```