Transformations

BIOE 498/598

2/5/2020

Linear Transformations: Scaling Predictor Variables

$$\text{earnings}[\$] = -6100 + 1300 \cdot \text{height}[\text{in}] + \text{error}$$

$$\text{earnings}[\$] = -6100 + 51 \cdot \text{height}[\text{mm}] + \text{error}$$

$$\text{earnings}[\$] = -6100 + 108 \cdot \text{height}[\text{ft}] + \text{error}$$

Linear Transformations: Scaling Predictor Variables

$$\text{earnings}[\$] = -6100 + 1300 \cdot \text{height}[\text{in}] + \text{error}$$

$$\text{earnings}[\$] = -6100 + 51 \cdot \text{height}[\text{mm}] + \text{error}$$

$$\text{earnings}[\$] = -6100 + 108 \cdot \text{height}[\text{ft}] + \text{error}$$

Scaling in General:

$$y = \beta_0 + \beta_1(kx) + \epsilon$$

$$\updownarrow$$

$$y = \beta_0 + (k\beta_1)x + \epsilon$$

Mean Centering

Recall an early model from our class:

$$\text{child.score} = \beta_0 + \beta_1 \text{mom.iq}$$

The effect size β_1 is the change in child score for every unit increase in mother IQ. The intercept β_0 was uninterpretable (the score of a child with a mother of IQ=0).

Mean Centering

Recall an early model from our class:

$$\text{child.score} = \beta_0 + \beta_1 \text{mom.iq}$$

The effect size β_1 is the change in child score for every unit increase in mother IQ. The intercept β_0 was uninterpretable (the score of a child with a mother of IQ=0).

Let's *mean center* the variable mom.iq:

$$\text{c.mom.iq} = \text{mom.iq} - \text{mean[mom.iq]}$$

Mean Centering (continued)

Our new model is

$$\text{child.score} = \beta_0 + \beta_1 \text{c.mom.iq}$$

Now both coefficients are interpretable:

- β_1 remains the increase in child score given a unit increase in mother's IQ.
- β_0 is the predicted child score for a child with mother of average IQ.

Standardization by Z-score

We can also center mom.iq and rescale it by the standard deviation:

$$\mathrm{z.mom.iq} = \frac{\mathrm{mom.iq} - \mathrm{mean}[\mathrm{mom.iq}]}{\mathrm{stdev}[\mathrm{mom.iq}]}$$

Standardization by Z-score

We can also center mom.iq and rescale it by the standard deviation:

$$\mathrm{z.mom.iq} = \frac{\mathrm{mom.iq} - \mathrm{mean}[\mathrm{mom.iq}]}{\mathrm{stdev}[\mathrm{mom.iq}]}$$

In the model

$$\mathrm{child.score} = \beta_0 + \beta_1 \mathrm{z.mom.iq}$$

the interpretation of β_0 is the same (predicted score for child with average mom.iq), but β_1 is the change in child score based on an increase of one standard devation in mother's IQ.

Why rescale by the standard deviation?

No scaling:

```r
lm(kid_score ~ mom_hs + c_mom_iq, child) %>% coefs()
```

```
## (Intercept)      mom_hs    c_mom_iq
##       82.12        5.95        0.56
```

Why rescale by the standard deviation?

No scaling:

```r
lm(kid_score ~ mom_hs + c_mom_iq, child) %>% coefs()
```

```
## (Intercept)      mom_hs    c_mom_iq
##       82.12        5.95        0.56
```

Z-scoring (1 stdev):

```r
lm(kid_score ~ z_mom_hs + z_mom_iq, child) %>% coefs()
```

```
## (Intercept)    z_mom_hs    z_mom_iq
##        86.8         2.4         8.5
```

What is the best scaling factor?

```r
lm(kid_score ~ mom_hs + c_mom_iq, child) %>% coefs()
```

```
## (Intercept)      mom_hs    c_mom_iq
##       82.12        5.95        0.56
```

Z-scoring (1 stdev):

```r
lm(kid_score ~ z_mom_hs + z_mom_iq, child) %>% coefs()
```

```
## (Intercept)    z_mom_hs    z_mom_iq
##        86.8         2.4         8.5
```

What is the best scaling factor?

```
lm(kid_score ~ mom_hs + c_mom_iq, child) %>% coefs()
```

```
## (Intercept)      mom_hs    c_mom_iq
##       82.12        5.95        0.56
```

Z-scoring (1 stdev):

```
lm(kid_score ~ z_mom_hs + z_mom_iq, child) %>% coefs()
```

```
## (Intercept)    z_mom_hs    z_mom_iq
##        86.8         2.4         8.5
```

Scaling by 2 stdev:

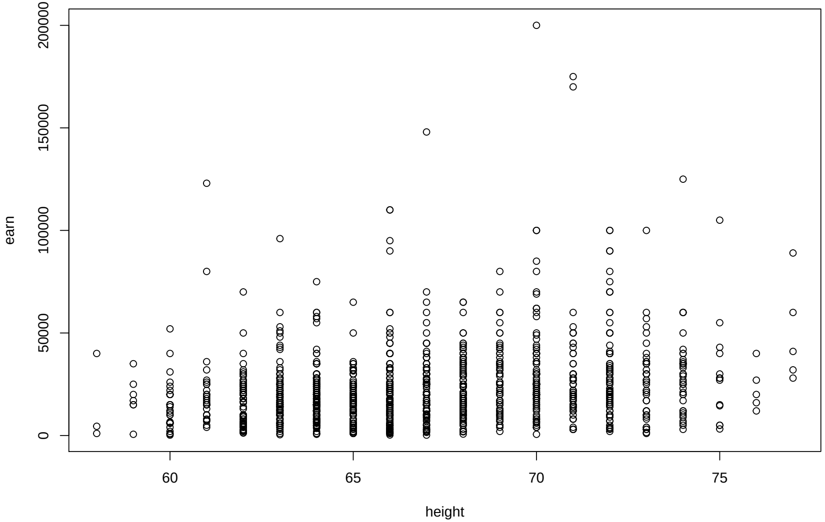
```
lm(kid_score ~ z2_mom_hs + z2_mom_iq, child) %>% coefs()
```

```
## (Intercept)   z2_mom_hs   z2_mom_iq
##        86.8         4.9        16.9
```

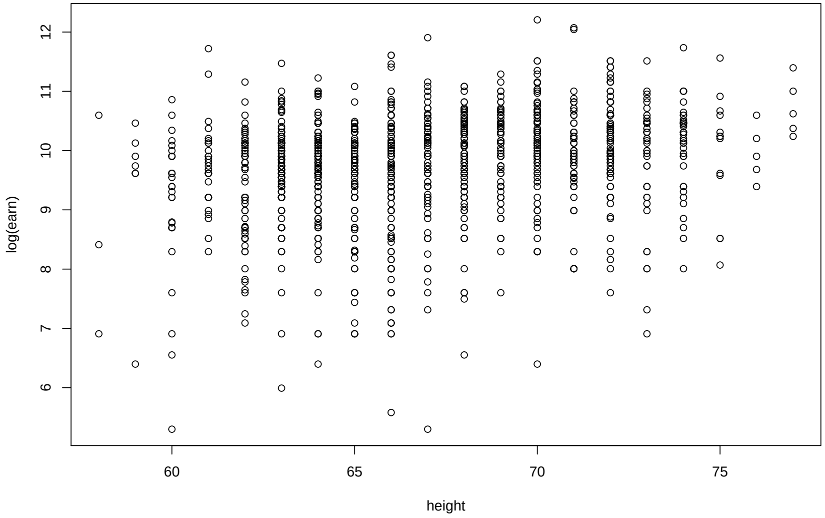
Our recommendations

- Leave binary (indicator) variables unscaled.
- Center and scale continuous variables by 2 stdev,
- *except* if the variable uses a scale widely accepted by your audience.
- If you need to compared effect sizes, build a rescaled model behind-the-scenes and present the conclusions.

Earnings vs. height?



log(Earnings) vs height?



Logarithmic transformation of the response

```r
lm(log(earn) ~ height, earnings) %>% coefs()
```

```
## (Intercept)      height
##        5.78        0.06
```

Logarithmic transformation of the response

```r
lm(log(earn) ~ height, earnings) %>% coefs()
```

```
## (Intercept)      height
##        5.78        0.06
```

This changes from an additive model to a multiplicative one:

$$\log(y) = \beta_0 + \beta_1 x_1 + \cdots + \epsilon$$

$$\downarrow$$

$$y = B_0 \cdot B_1^{x_1} \cdots E$$

What is the best response transformation?

A general family of nonlinear transformations are described by the Box-Cox transformation:

$$T(y) = \begin{cases} (y^\lambda - 1)/\lambda, & \lambda \neq 0 \\ \log(y), & \lambda = 0 \end{cases}$$

What is the best response transformation?

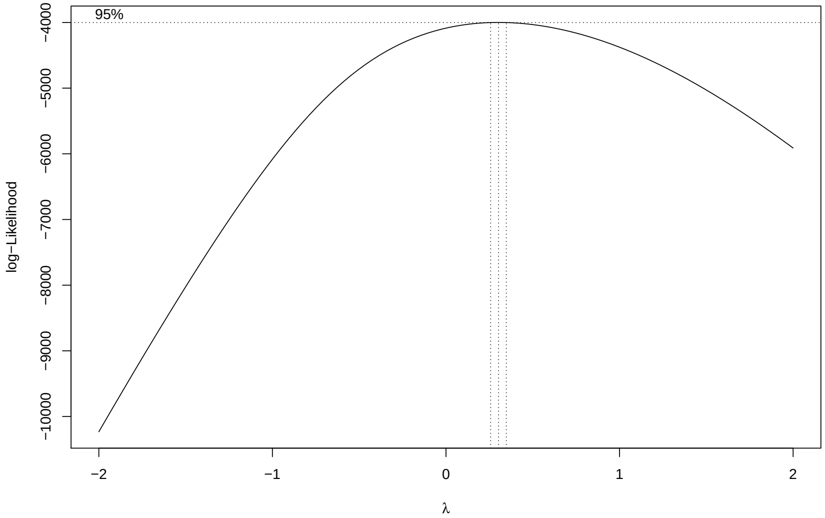
A general family of nonlinear transformations are described by the Box-Cox transformation:

$$T(y) = \begin{cases} (y^\lambda - 1)/\lambda, & \lambda \neq 0 \\ \log(y), & \lambda = 0 \end{cases}$$

- $\lambda = 2$ suggests $y \to y^2$
- $\lambda = 1$ suggests no transformation
- $\lambda = 1/2$ suggests $y \to \sqrt{y}$
- $\lambda = -1$ suggests $y \to 1/y$

Where do we get λ?

```r
MASS::boxcox(earn ~ height, data=earnings)
```



Shouldn't we always transform the response?

- Transforming the response will often improve the predictive power of the model.
- Models with transformed responses are more difficult to interpret.

Shouldn't we always transform the response?

- Transforming the response will often improve the predictive power of the model.
- Models with transformed responses are more difficult to interpret.

- In general, **there is always a tradeoff between prediction and interpretation**.
- Recommendation: perform the Box-Cox analysis, but only transform if
  - you only want the model for prediction, *or*
  - the Box-Cox suggests a common transformation (log, square root, inverse, square, etc.).