Regression Models (Part V)

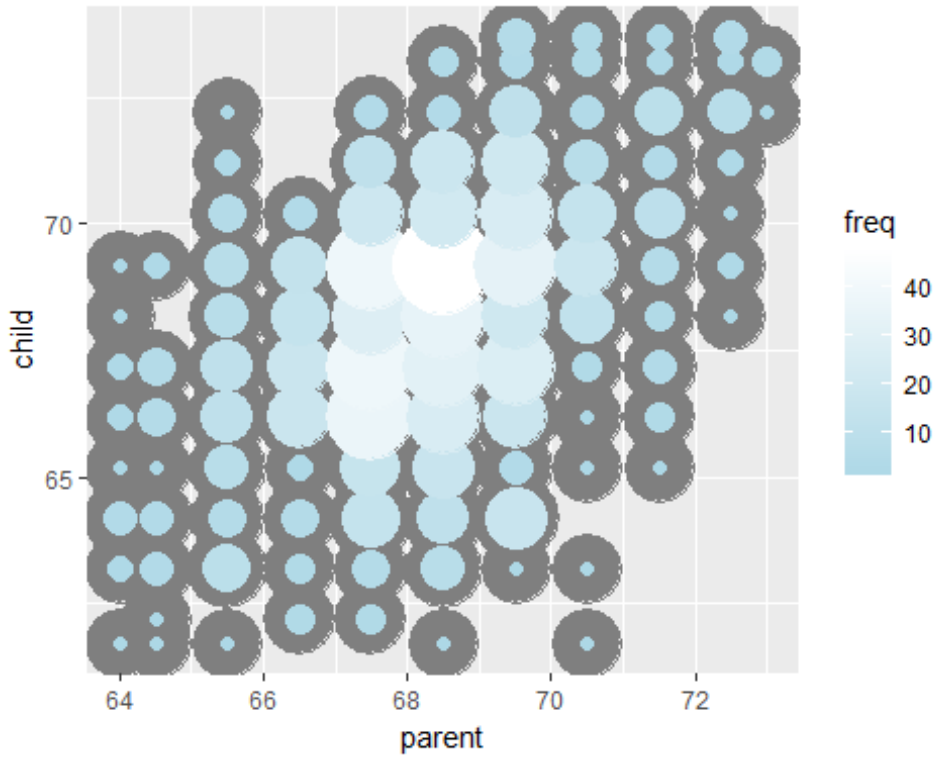
A. Ordinary least squares

Ordinary least squares (OLS) is the workhorse of statistics. It gives a way of taking complicated outcomes and explaining behavior (such as trends) using linearity. The simplest application of OLS is fitting a line.

B. General least squares for linear equations

Consider again the parent and child height data from Galton.

```
library(UsingR)
data(galton)
library(dplyr)
freqData <- as.data.frame(table(galton$child, galton$parent))
names(freqData) <- c("child", "parent", "freq")
freqData$child <- as.numeric(as.character(freqData$child))
freqData$parent <- as.numeric(as.character(freqData$parent))
g <- ggplot(filter(freqData, freq > 0), aes(x = parent, y = child))
g <- g  + scale_size(range = c(2, 20), guide = "none" )
g <- g + geom_point(colour="grey50", aes(size = freq+20, show_guide = FALSE))
g <- g + geom_point(aes(colour=freq, size = freq))
g <- g + scale_colour_gradient(low = "lightblue", high="white")
g
```



Let's try fitting the best line. Let Y_i be the i^{th} child's height and X_i be the i^{th} (average over the pair of) parental heights. Consider finding the best line of the form

$$\text{Child Height} = \beta_0 + \text{Parent Height}\beta_1,$$

Let's try using least squares by minimizing the following equation over β_0 and β_1:

$$\sum_{i=1}^{n} \{Y_i - (\beta_0 + \beta_1 X_i)\}^2.$$

Minimizing this equation will minimize the sum of the squared distances between the fitted line at the pareNnts heights ($\beta_1 X_i$) and the observed child heights (Y_i).

The result actually has a closed form. Specifically, the least squares of the line:

$$Y = \beta_0 + \beta_1 X,$$

through the data pairs (X_i, Y_i) with Y_i as the outcome obtains the line $Y = \hat{\beta}_0 + \hat{\beta}_1 X$ where:

$\hat{\beta}_1 = Cor(Y, X)\frac{Sd(Y)}{Sd(X)}$ and $\hat{\beta}_0 = \overline{Y} - \hat{\beta}_1 \overline{X}$.$

At this point, a couple of notes are in order. First, the slope, $\hat{\beta}_1$, has the units of Y/X.

Secondly, the intercept, $\hat{\beta}_0$, has the units of Y.

The line passes through the point $(\overline{X}, \overline{Y})$. If you center your Xs and Ys first, then the line will pass through the origin. Moreover, the slope is the same one you would get if you centered the data, $(X_i - \overline{X}, Y_i - \overline{Y})$, and either fit a linear regression or regression through the origin.

To elaborate, regression through the origin, assuming that $\beta_0 = 0$, yields the following solution to the least squares criteria:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} X_i Y_i}{\sum_{i=1}^{n} X_i^2},$$

This is exactly the correlation times the ratio in the standard deviations if the both the Xs and Ys have been centered first. (Try it out using R to verify this!)

It is interesting to think about what happens when you reverse the role of X and Y. Specifically, the slope of the regression line with X as the outcome and Y as the predictor is Cor(Y, X) Sd(X)/ Sd(Y).

If you normalized the data, $\{\frac{X_i - \overline{X}}{Sd(X)}, \frac{Y_i - \overline{Y}}{Sd(Y)}\}$, the slope is simply the correlation, $Cor(Y, X)$, regardless of which variable is treated as the outcome.

C. Revisiting Galton's data

Let's double check our calculations using R

```
y <- galton$child
x <- galton$parent
beta1 <- cor(y, x) *  sd(y) / sd(x)
beta0 <- mean(y) - beta1 * mean(x)
rbind(c(beta0, beta1), coef(lm(y ~ x)))
##      (Intercept)         x
## [1,]    23.94153 0.6462906
## [2,]    23.94153 0.6462906
```

We can see that the result of lm is identical to hard coding the fit ourselves. Let's Reversing the outcome/predictor relationship.

```
beta1 <- cor(y, x) *  sd(x) / sd(y)
beta0 <- mean(x) - beta1 * mean(y)
rbind(c(beta0, beta1), coef(lm(x ~ y)))
##      (Intercept)         y
## [1,]    46.13535 0.3256475
## [2,]    46.13535 0.3256475
```

Now let's show that regression through the origin yields an equivalent slope if you center the data first

```
yc <- y - mean(y)
xc <- x - mean(x)
beta1 <- sum(yc * xc) / sum(xc ^ 2)
c(beta1, coef(lm(y ~ x))[2])
##                   x
## 0.6462906 0.6462906
```

Now let's show that normalizing variables results in the slope being the correlation.

```
yn <- (y - mean(y))/sd(y)
xn <- (x - mean(x))/sd(x)
c(cor(y, x), cor(yn, xn), coef(lm(yn ~ xn))[2])
##                          xn
## 0.4587624 0.4587624 0.4587624
```

The image below plots the data again, the best fitting line and standard error bars for the fit. We'll work up to that point later. But, understanding that our fitted line is estimated with error is an important concept.