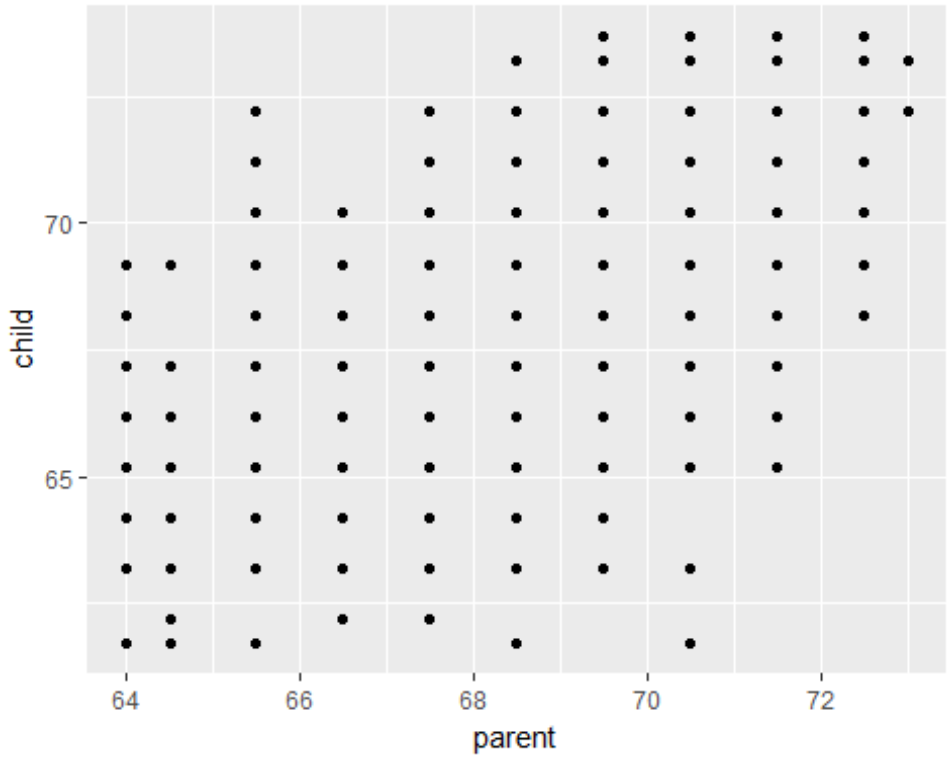
Regression Models (Part III)

F. Comparing children's heights and their parent's heights

Looking at either the parents or children on their own isn't interesting. We're interested in how the relate to each other. Let's plot the parent and child heights.

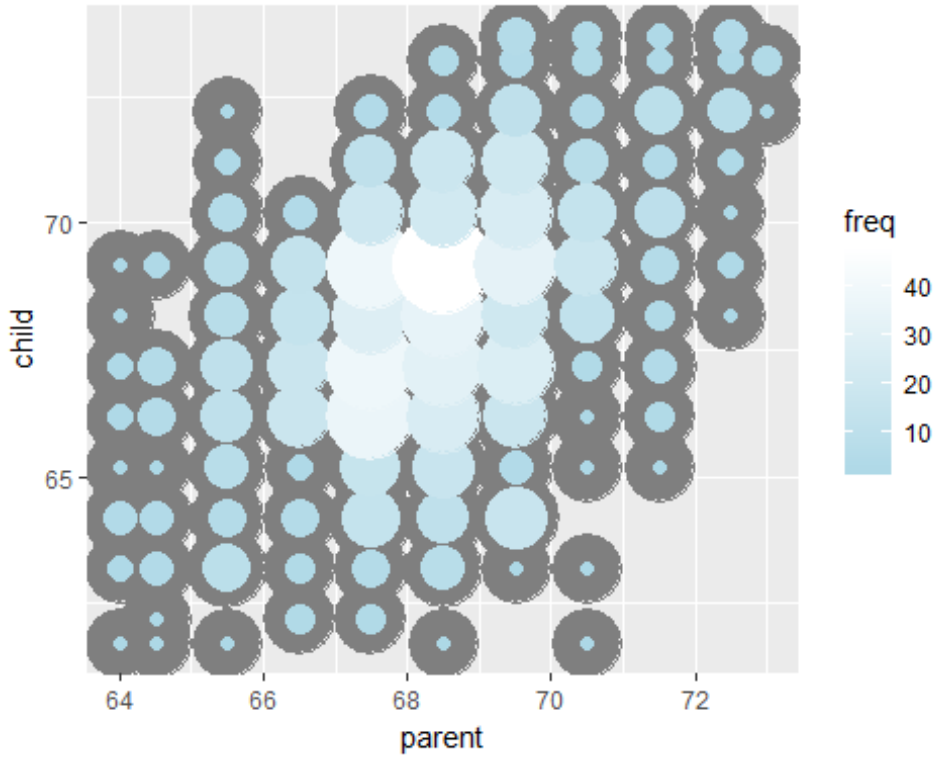
```
library(UsingR)
ggplot(galton, aes(x = parent, y = child)) + geom_point()
```



The overplotting is clearly hiding some data. Here you can get the code to make the size and color of the points be the frequency.

```
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))
## Warning: Ignoring unknown aesthetics: show_guide
g <- g + geom_point(aes(colour=freq, size = freq))
g <- g + scale_colour_gradient(low = "lightblue", high="white")
g
```



G. Regression through the origin

A line requires two parameters to be specified, the intercept and the slope. Let's first focus on the slope. We want to find the slope of the line that best fits the data. However, we have to pick a good intercept. Let's subtract the mean from bot the parent and child heights so that their subsequent means are 0. Now let's find the line that goes through the origin (has intercept 0) by picking the best slope.

Suppose that X_i are the parent heights with the mean subtracted. Consider picking the slope β that minimizes

$$\sum_{i=1}^{n} (Y_i - X_i\beta)^2$$

Each $X_i\beta$ is the vertical height of a line through the origin at point X_i. Thus, $Y_i - X_i\beta$ is the vertical distance between the line at each observed X_i point (parental height) and the Y_i (child height).

Our goal is exactly to use the origin as a pivot point and pick the line that minimizes the sum of the squared vertical distances of the points to the line. Use R studio's manipulate function to experiment Subtract the means so that the origin is the mean of the parent and children heights.

```
y <- galton$child - mean(galton$child)
x <- galton$parent - mean(galton$parent)
freqData <- as.data.frame(table(x, y))
names(freqData) <- c("child", "parent", "freq")
freqData$child <- as.numeric(as.character(freqData$child))
freqData$parent <- as.numeric(as.character(freqData$parent))
myPlot <- function(beta){
    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 = FAL
SE))
    g <- g + geom_point(aes(colour=freq, size = freq))
    g <- g + scale_colour_gradient(low = "lightblue", high="white")
    g <- g + geom_abline(intercept = 0, slope = beta, size = 3)
    mse <- mean( (y - beta * x) ^2 )
    g <- g + ggtitle(paste("beta = ", beta, "mse = ", round(mse, 3)))
    g
}
```

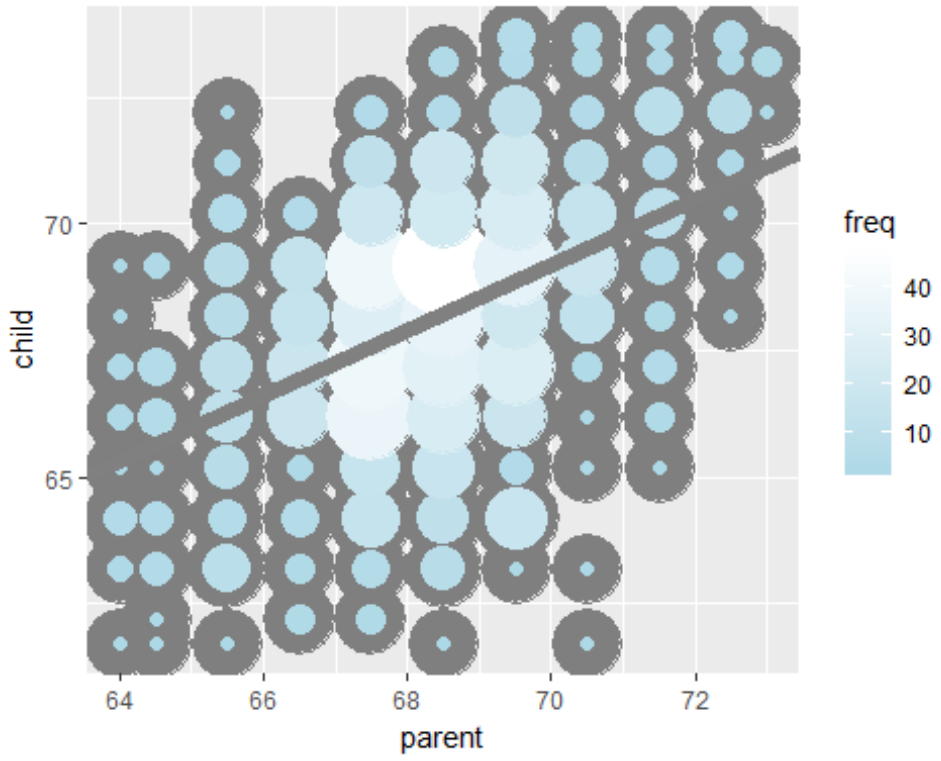
H. The solution

In the next few lectures we'll talk about why this is the solution. But, rather than leave you hanging, here it is:

```
lm(I(child - mean(child))~ I(parent - mean(parent)) - 1, data = galton)
##
## Call:
## lm(formula = I(child - mean(child)) ~ I(parent - mean(parent)) -
##     1, data = galton)
##
## Coefficients:
## I(parent - mean(parent))
##                   0.6463
```

Let's plot the best fitting line. In the subsequent chapter we will learn all about creating, interpreting and performing inference on such mode fits. (Note that I shifted the origin back to the means of the original data.) The results suggest that to every every 1 inch increase in the parents height, we estimate a 0.646 inch increase in the child's height.

```
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))
## Warning: Ignoring unknown aesthetics: show_guide
g <- g + geom_point(aes(colour=freq, size = freq))
g <- g + scale_colour_gradient(low = "lightblue", high="white")
lm1 <- lm(galton$child ~ galton$parent)
g <- g + geom_abline(intercept = coef(lm1)[1], slope = coef(lm1)[2], size = 3
, colour = grey(.5))
g
```


70
65
child
64
66
68
70
72
parent
freq
40
30
20
10