Data Analytics
CS390
Modeling: Formal Basics

Fall 2017

Oliver Bonham-Carter

What does it mean to regress *Y* on *X*?

- A function defines one variable in terms of another.
- The statement "y is a function of x" (denoted y = y(x)) means that y varies according to whatever value x takes on.
- A causal relationship is often implied (i.e. "x causes y"), but does not *necessarily* exist.

Extracting Info

- Create model object to look for “What If?” patterns.
- Run functions on model object to get details

  Try these commands

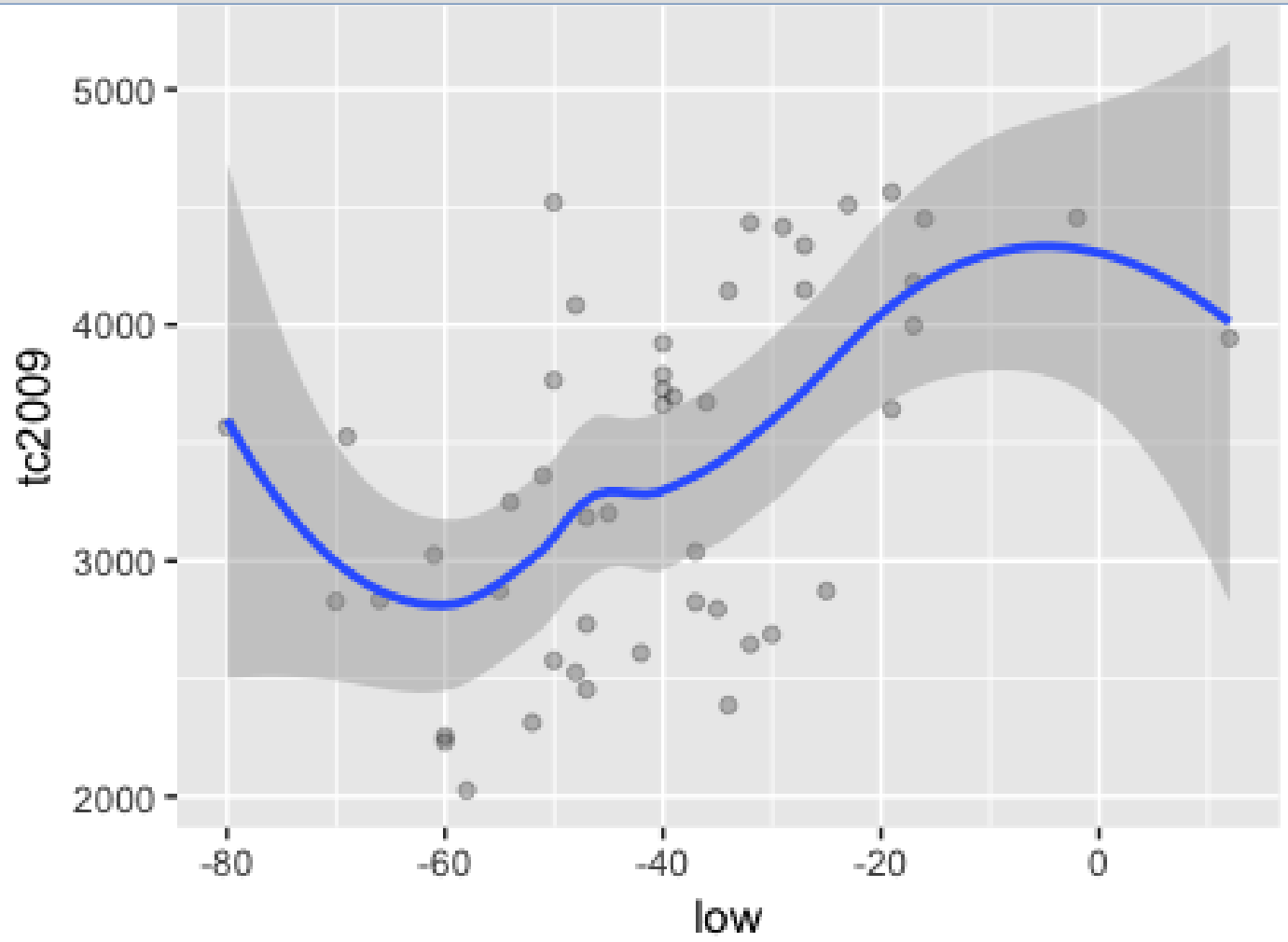
```
summary(mod)
predict(mod) # predictions at original vals
resid(mod) # residuals
```

Let's Hit the Code

```
#plot the data

crime %>% ggplot(aes(x = low, y = tc2009)) +
geom_point(alpha = I(1/4)) + geom_smooth(method =
lm)

crime %>% ggplot(aes(x = low, y = tc2009)) +
geom_point(alpha = I(1/4)) + geom_smooth()

#Build the model

mod1 <- lm(tc2009 ~ low, data = crime)
```

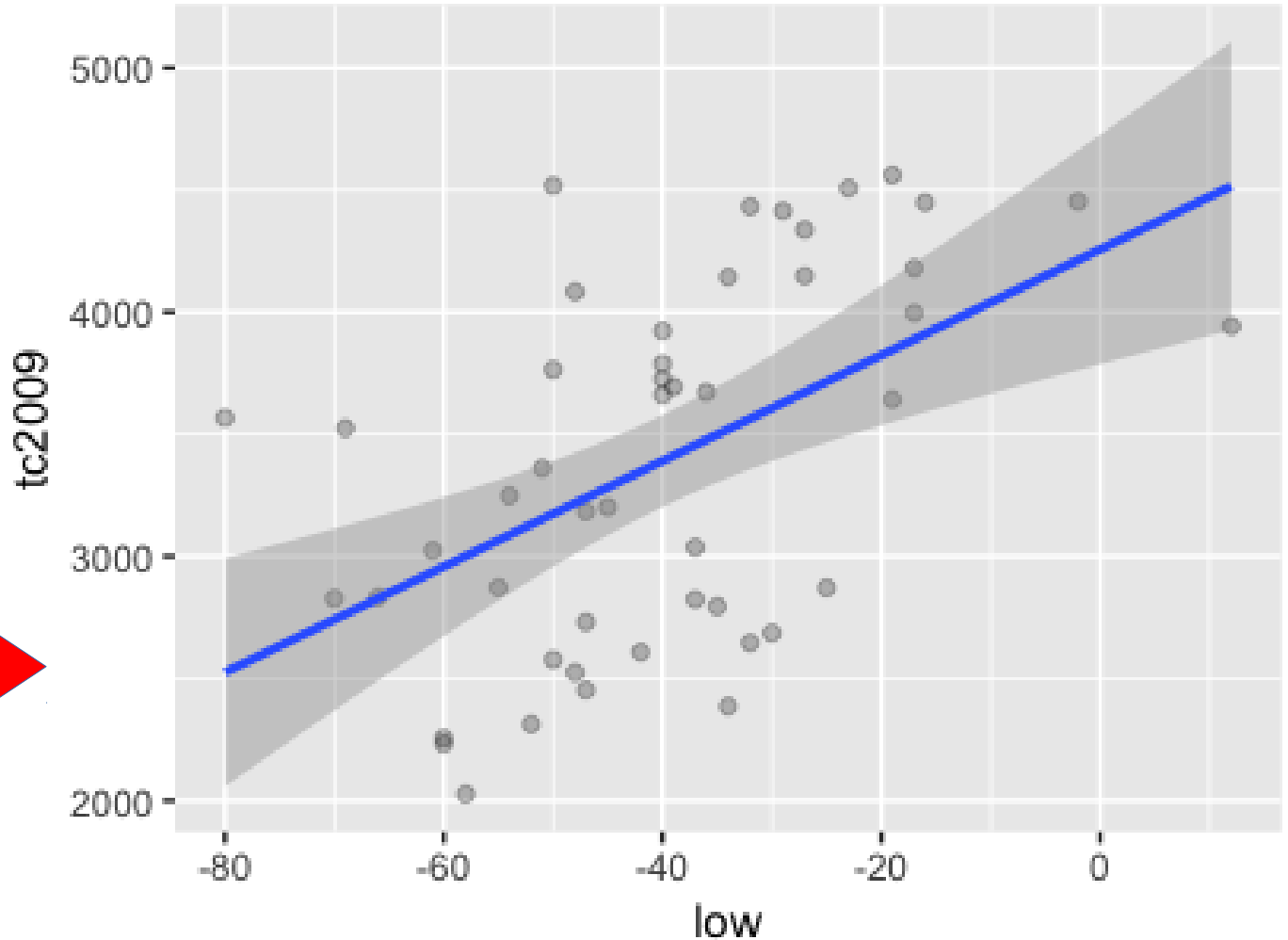
Plots

```
crime %>% ggplot(aes(x = low, y = tc2009)) +
geom_point(alpha = I(1/4)) + geom_smooth()
```



Plots

```
crime %>% ggplot(aes(x = low, y = tc2009)) +
geom_point(alpha = I(1/4)) + geom_smooth(method = lm)
```



This is the model's line here!

Build A Model
To Play With

```
mod1 <- lm(tc2009 ~ low, data = crime)
```

```
Call:
lm(formula = tc2009 ~ low, data = crime)

Coefficients:
(Intercept)          low
    4256.86        21.65
```

Coef

- Shows the model's coefficients (I.e., intercept, slopes)

```
coef(mod)
coefficients(mod)
# (Intercept)                low
#  4256.86158    21.64725
```

α β

Interpreting Models

Linear models are very easy to interpret

$$y = \alpha + \beta x + \epsilon$$

α is the expected value of y when x is 0.

β is the expected increase in y associated with a one unit increase in x

Coef

```
coef(mod)
coefficients(mod)
#  (Intercept)                low
#    4256.86158       21.64725
```

The best estimate of
tc2009 for a state with low = -10 is
4256.86 + **21.6** * (-10) = 4040.86

(x,y) ← (-10, 4040.86)

Coef Calculator

This function is now my data!!

```
# create function to find y for x
tellMeY <- function(x_int){
  #function to get the y value for an entered x value
  # The best estimate of tc2009 for a state with low of inputted value x_int
  cat("  intercept :",mod1$coefficients[1] )
  cat("\n  slope     :",mod1$coefficients[2] )
  y = mod1$coefficients[1] + x_int * mod1$coefficients[2]
  cat("\n  y = ",y)
}

tellMeY(-10) # note: x = -10 also, my “what if?” enabler
```

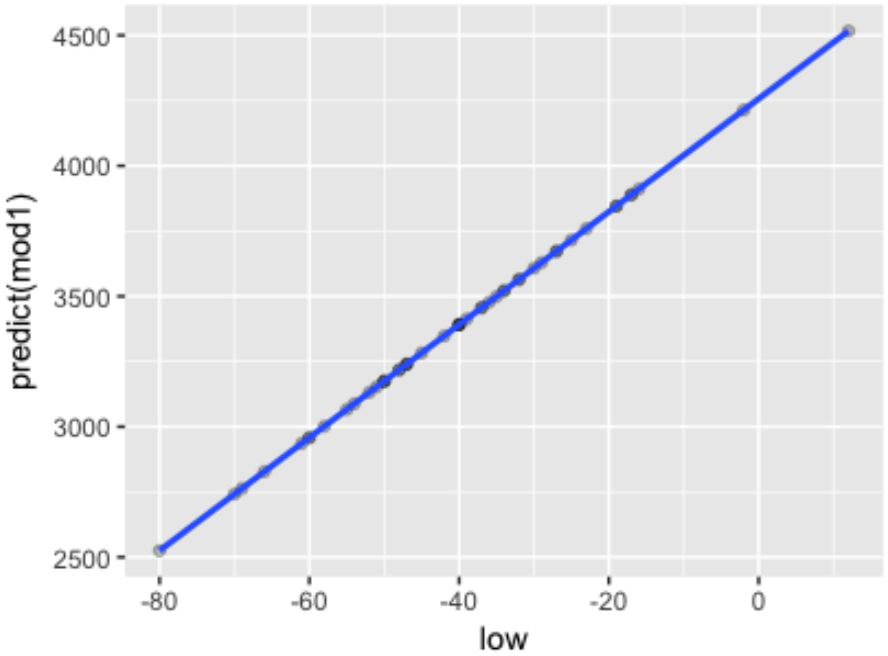
The best estimate of
tc2009 for a state with low = -10 is
4256.86 + 21.6 * (-10) = 4040.86

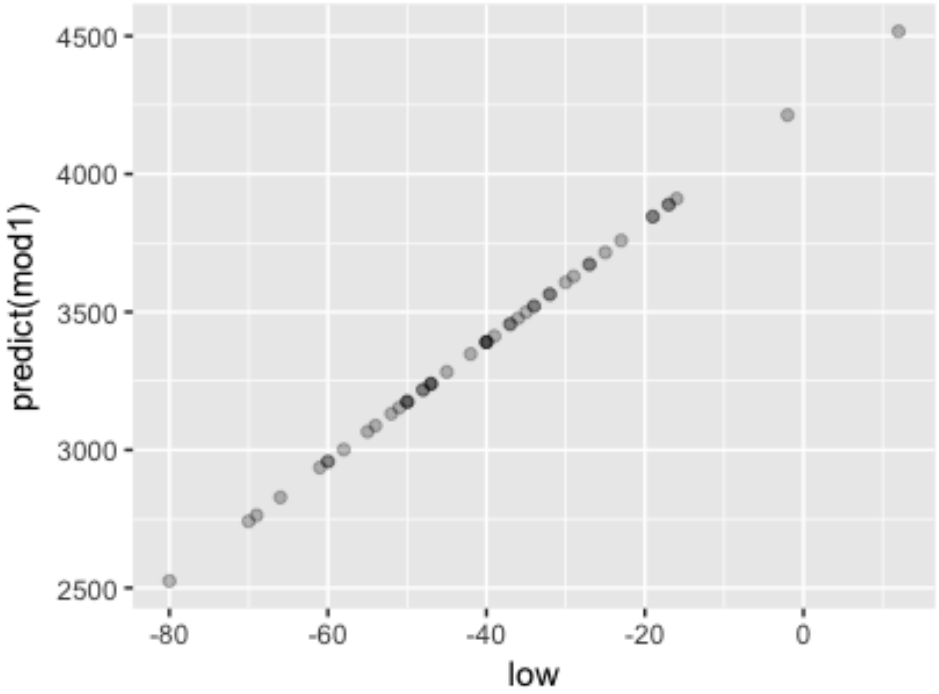
Due to error, there is a slight difference between this and our value.

The Model's Line

```
crime %>% ggplot(aes(x = low, y = predict(mod1))) +
geom_point(alpha = I(1/4))
```

```
crime %>% ggplot(aes(x = low, y = predict(mod1))) +
geom_point(alpha = I(1/4)) + geom_smooth()
```





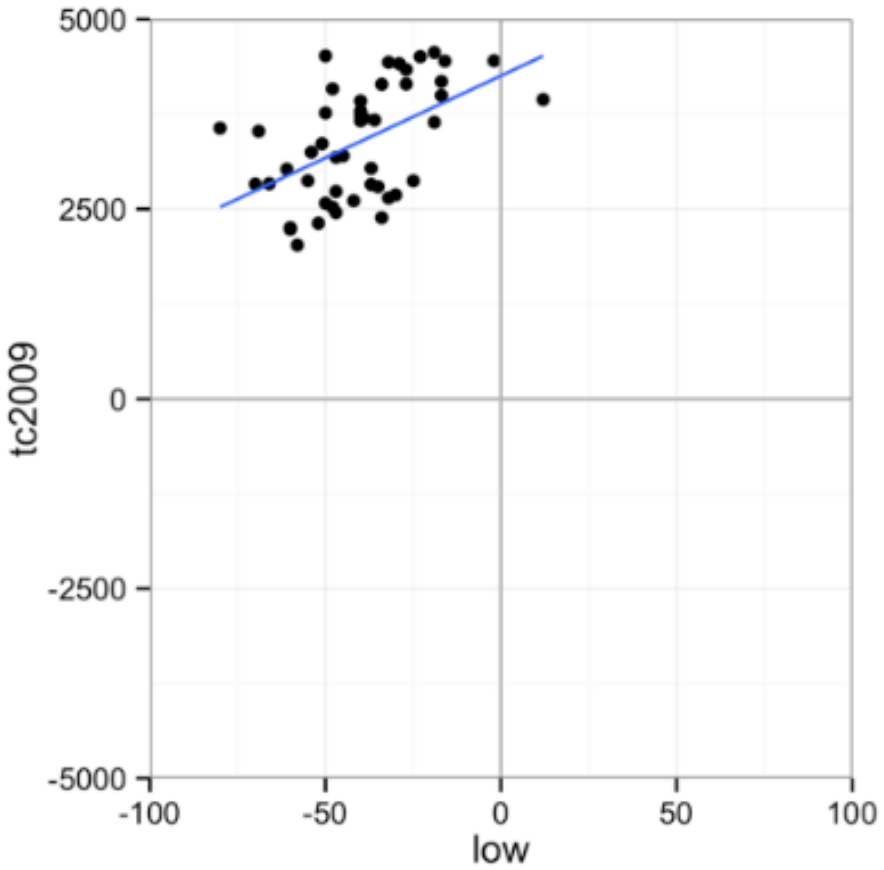
Aside: intercept terms

R includes an intercept term in each model by default

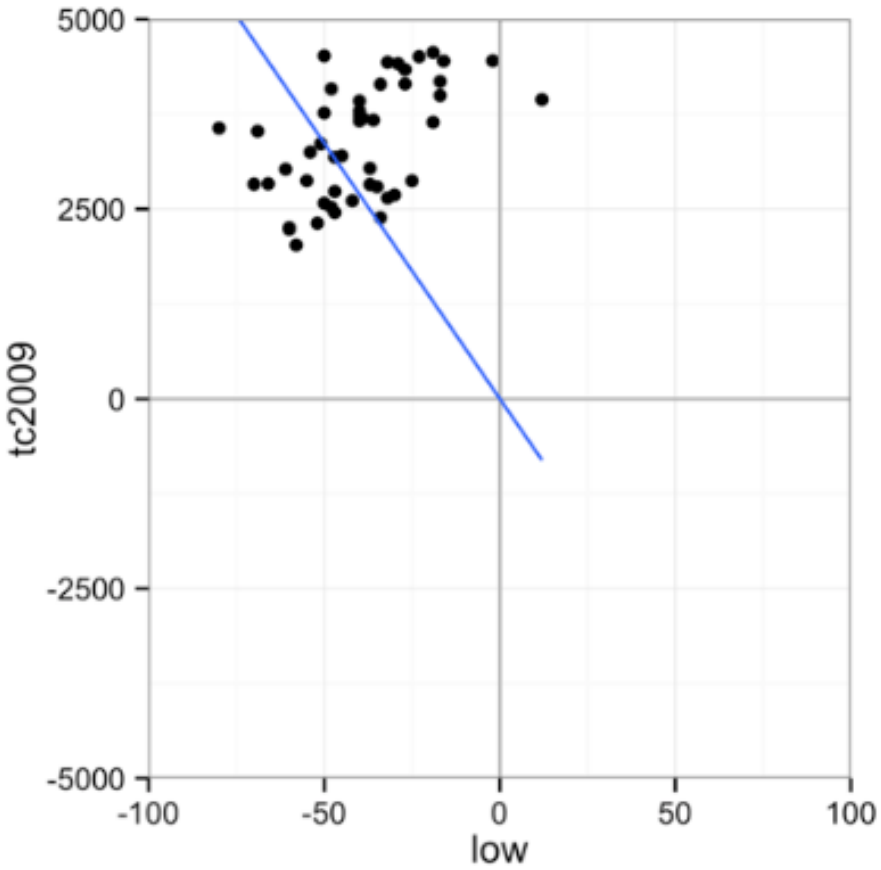
$$y = \alpha + \beta x + \epsilon$$

```
y ~ x
```

Want the Zeros or Not?



With α



Without α

Every linear model has a y intercept. Including α lets this term vary. Not including α forces the intercept to (0, 0).

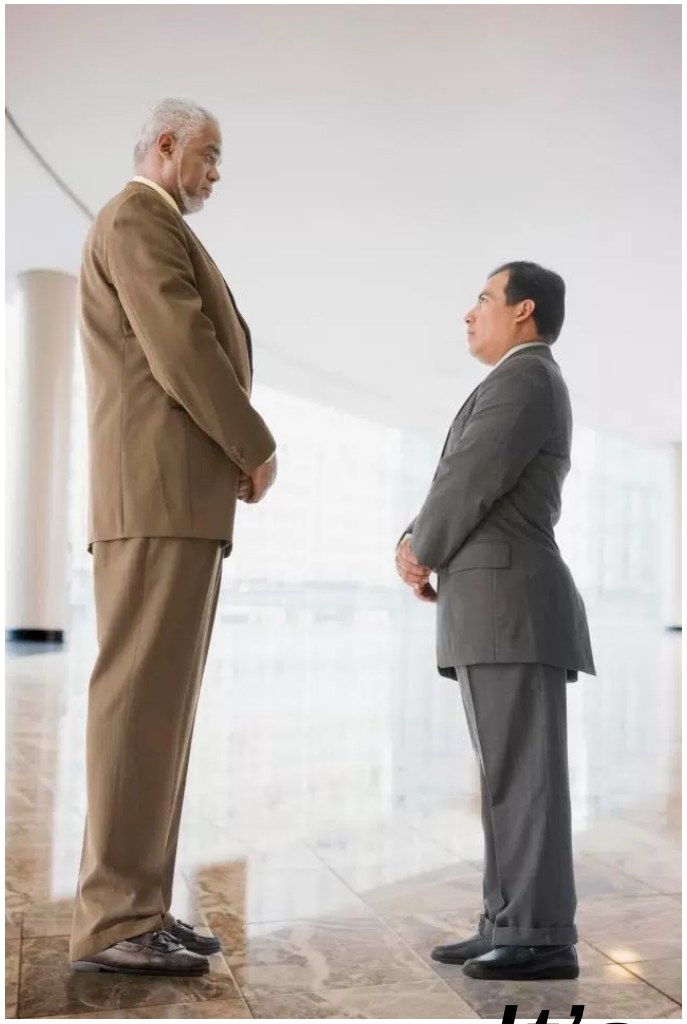
An Intercept Term: To Use or Not?

ALLEGHENY COLLEGE

You can explicitly ask for an intercept by including the number one, 1, as a formula term. You can remove the intercept by including a zero or negative 1.

```
# equivalent - includes intercept
lm(tc2009 ~ 1 + low, data = crime)
lm(tc2009 ~ low, data = crime)

# equivalent - removes intercept
lm(tc2009 ~ low - 1, data = crime)
lm(tc2009 ~ 0 + low, data = crime)
```

Now, Back to Our Question...



Do you think that taller people make more money?

File: wages.csv

Remember:

It's not you, it's your data.

Consider This!

Fit a linear model to the wages data set that predicts ***earn*** with ***height***.

How do you interpret the relationship between **height** and **earnings**?

```
wages <- read.csv("wages.csv")
```

Dep And Indep Vars

- #make your model
- hmod <- lm(dependent ~ independent)
- Where **dependent** var is ***earn***
- And **independent** var is ***height***

$$y = \alpha + \beta x + \epsilon$$

Earn Regressed Over *height*

- #make your model
- hmod <- lm(earn ~ height)
- Where **dependent** var is ***earn***
- And **independent** var is ***height***

$$earn = \alpha + \beta \times height + \epsilon$$

Earn Regressed Over *height*

```
hmod <- lm(earn ~ height, data = wages)
coef(hmod)
## (Intercept)      height
## -126523.359    2387.196
```

$$earn = \alpha + \beta \times height + \epsilon$$

$$earn = -126523.36 + 2387.20 \times height + \epsilon$$

Earn Regressed Over *height*

The best estimate of earn for someone 68 inches tall is

$$earn = -126523.36 + 2387.20 \times 68 + \epsilon$$

$$earn = 35806.24$$

Do Tall People Make More?

```
wages %>% ggplot(aes(x = height, y = earn)) +
geom_point(alpha = I(1/4)) + geom_smooth()

wages %>% ggplot(aes(x = height, y = earn)) +
geom_point(alpha = I(1/4)) + geom_smooth(method = lm) #
regression line
```