Gov 2000 - 6. What is Regression?

Matthew Blackwell

October 20, 2015

1. Relationships Between Two Variables
2. Conditional Expectation
3. Conditional Expectations with Continuous Covariates
4. Lines of Best Fit
5. Least Squares

Where are we? Where are we going?

- What we've been up to: estimating parameters of population distributions. Generally we've been learning about a single variable.
- This week and for the rest of the term, we'll be interested in the relationships between variables. How does one variable change we change the values of another variable? These will be the bread and butter of the class moving forward.

1/ Relationships Between Two Variables

What is a relationship and why do we care?

- Most of what we want to do in the social science is learn about how two variables are related
- Examples:
  - Does turnout vary by types of mailers received?
  - Is the quality of political institutions related to average incomes?
  - Does conflict mediation help reduce civil conflict?

Notation and conventions

- Y_i - the dependent variable or outcome or regressand or left-hand-side variable or response
  - Voter turnout
  - Log GDP per capita
  - Number of battle deaths
- X_i - the independent variable or explanatory variable or regressor or right-hand-side variable or treatment or predictor
  - Social pressure mailer versus Civic Duty Mailer
  - Average Expropriation Risk
  - Presence of conflict mediation

Joint distribution review

- (Y_i, X_i) are draws from an i.i.d. joint distribution $f_{Y,X}$
  - Y_i and X_i are measured on the same unit i
- Regression tries to understand how Y_i varies as a function of X_i:
$$Y_i = f(X_i) + \text{error}$$
- **WARNING** different than our use of Y_i and X_i as r.v.s for different groups.
  - There, Y_i and X_i corresponded to different units.

Three uses of regression

1. **Description** - parsimonious summary of the data
2. **Prediction/Estimation/Inference** - learn about parameters of the joint distribution of the data
3. **Causal Inference** - evaluate counterfactuals

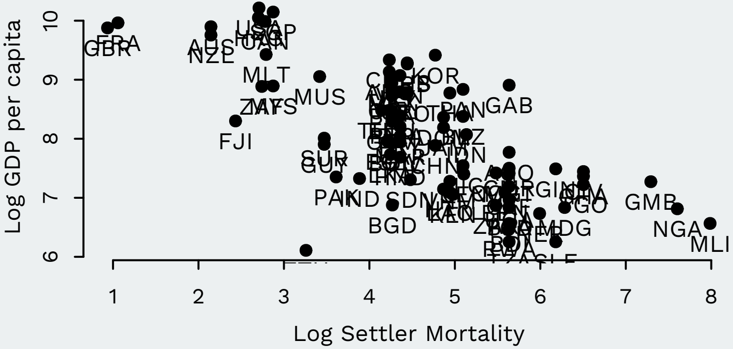
Describing relationships

- Remember that we had ways to summarize the relationship between variables in the population.
- Joint densities, covariance, and correlation were all ways to summarize the relationship between two variables.
- But these were population quantities and we only have samples, so we may want to estimate these quantities using their sample analogs

Scatterplots

- Sample version of joint probability density.
- Shows graphically how two variables are related

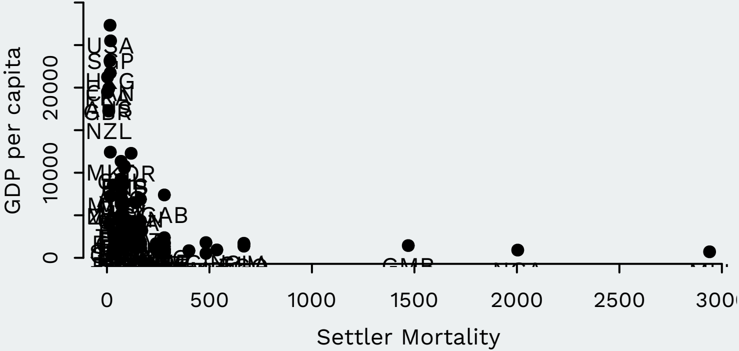
```
plot(ajr$logem4, ajr$logpgp95, xlab = "Log Settler Mortality",
     ylab = "Log GDP per capita", pch = 19, bty = "n")
text(ajr$logem4, ajr$logpgp95, ajr$shortnam, pos = 1)
```



Non-linear relationship

- Example of a non-linear relationship, where we use the unlogged version of GDP and settler mortality:

```
plot(exp(ajr$logem4), exp(ajr$logpgp95), xlab = "Settler Mortality",
     ylab = "GDP per capita", pch = 19, bty = "n")
text(exp(ajr$logem4), exp(ajr$logpgp95), ajr$shortnam, pos = 1)
```



Sample covariance

- Population covariance:

$$\text{Cov}(X, Y) = \mathbb{E}[(X_i - \mathbb{E}[X])(Y_i - \mathbb{E}[Y])]$$

- **Defintion** The **sample covariance** between Y_i and X_i is

$$\widehat{\text{Cov}}(X, Y) = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X}_n)(Y_i - \overline{Y}_n)$$

```
## tell cov() to use only the pairwise complete observations:
cov(ajr$logem4, ajr$logpgp95, use = "pair")
```

```
## [1] -0.9881
```

Sample correlation

- Population correlation:

$$\rho = \mathrm{Cov}(X, Y)/\sigma_X \sigma_Y$$

- **Defintion** The **sample correlation** between Y_i and X_i is

$$\hat{\rho} = r = \frac{\widehat{\mathrm{Cov}}(X, Y)}{S_X S_Y} = \frac{\sum_{i=1}^{n}(X_i - \overline{X}_n)(Y_i - \overline{Y}_n)}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X}_n)^2 \sum_{i=1}^{n}(Y_i - \overline{Y}_n)^2}}$$

```r
## and has the same solution to NAs:
cor(ajr$logem4, ajr$logpgp95, use = "pair")
```

```
## [1] -0.7048
```

2/ Conditional Expectation

Conditional expectation review

- **Definition** The population **conditional expectation function** (CEF), $\mathbb{E}[Y_i|X_i = x]$, is the function that gives the mean of Y at various values of x.
  - Also called the regression function.
  - The CEF is a function of x: $\mu(x)$.
- $\mathbb{E}[Y_i|X_i = x]$ is a feature of the population distribution.
- We will want to produce estimates $\widehat{\mathbb{E}}[Y_i|X_i = x]$
- Regression at its most fundamental is about how the mean of Y changes as a function of X

CEF for binary covariates

- We've been writing μ_y and μ_x for the means in different groups.
- Different approach:
  - Y_i is the outcome for every unit in either group.
  - $X_i = 1$ for women, $X_i = 0$ for men.
- Then the mean in each group is just a conditional expectation:

$$\mu_w = E[Y_i | X_i = 1]$$
$$\mu_m = E[Y_i | X_i = 0]$$

- Notice here that since X_i can only take on two values, 0 and 1, then these two conditional means completely summarize the CEF.

Estimating the CEF for binary covariates

- How do we estimate $\widehat{\mathbb{E}}[Y_i|X_i = x]$?
- Sample means within each group:

$$\widehat{\mathbb{E}}[Y_i|X_i = 1] = \frac{1}{n_1} \sum_{i:X_i=1} Y_i$$

$$\widehat{\mathbb{E}}[Y_i|X_i = 0] = \frac{1}{n_0} \sum_{i:X_i=0} Y_i$$

- $n_1 = \sum_{i=1}^{n} X_i$ is the number of women in the sample.
- $n_0 = n - n_1$ is the number of men.
- $\sum_{i:X_i=1}$ sum only over the i that have $X_i = 1$, meaning that i is a woman.
- $\rightsquigarrow$ estimate the mean of Y_i conditional on X_i by just estimating the means within each group of X_i.

Binary covariate example

```r
## mean of log GDP among non-African countries
mean(ajr$logpgp95[ajr$africa == 0], na.rm = TRUE)
```

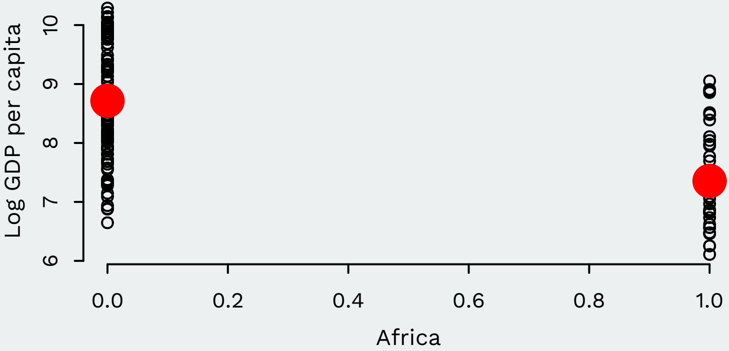
```
## [1] 8.716
```

```r
## mean of log GDP among African countries
mean(ajr$logpgp95[ajr$africa == 1], na.rm = TRUE)
```

```
## [1] 7.355
```

Binary covariate CEF plot

```r
plot(ajr$africa, ajr$logpgp95, ylab = "Log GDP per capita", xlab = "Africa",
    bty = "n")
points(x = 0, y = mean(ajr$logpgp95[ajr$africa == 0], na.rm = TRUE),
    pch = 19, col = "red", cex = 3)
points(x = 1, y = mean(ajr$logpgp95[ajr$africa == 1], na.rm = TRUE),
    pch = 19, col = "red", cex = 3)
```



Discrete covariate: estimating the CEF

- What if X_i isn't binary, but takes on > 2 discrete values?
- The same logic applies, we can still estimate $\mathbb{E}[Y_i|X_i = x]$ with the sample mean among those who have $X_i = x$:

$$\widehat{\mathbb{E}}[Y_i|X_i = x] = \frac{1}{n_x} \sum_{i:X_i=x} Y_i$$

Discrete covariate example

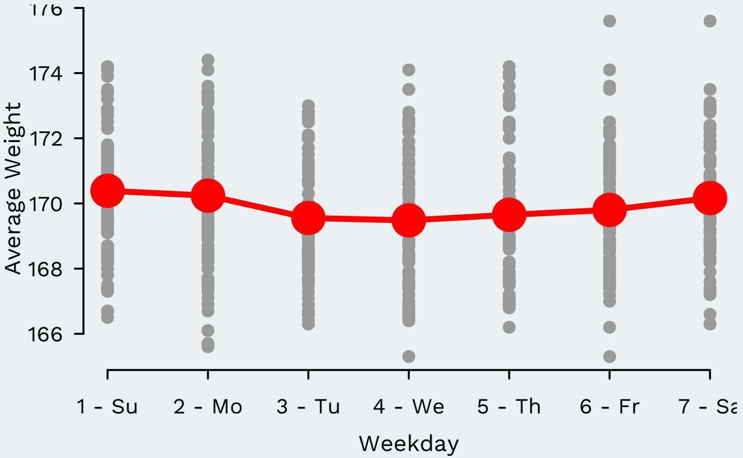
- I've been collecting data on my own weight for a while.
- How does my weight (Y_i) varied by the day of the week (X_i)?
- Calculate the mean weight for each day of the week:

```
weight <- read.csv("weight.csv", stringsAsFactors = FALSE)
weight$weekday <- as.numeric(format(as.Date(weight$date, format = "%m/%d/%y%n%H:%M"),
    "%w")) + 1
weight$date <- as.Date(weight$date, format = "%m/%d/%y%n%H:%M")
day.means <- rep(NA, times = 7)
names(day.means) <- c("1 - Su", "2 - Mo", "3 - Tu", "4 - We", "5 - Th",
    "6 - Fr", "7 - Sa")
for (i in 1:7) {
    day.means[i] <- mean(weight$weight[weight$weekday == i])
}
day.means
```

```
## 1 - Su 2 - Mo 3 - Tu 4 - We 5 - Th 6 - Fr 7 - Sa
##  170.4  170.2  169.6  169.5  169.7  169.8  170.2
```

Discrete covariate CEF plot

```r
plot(x = weight$weekday, y = weight$weight, xaxt = "n", xlab = "Weekday",
    ylab = "Average Weight", pch = 19, col = "grey60")
points(x = 1:7, y = day.means, pch = 19, col = "red", cex = 3)
lines(x = 1:7, y = day.means, pch = 19, col = "red", lwd = 3)
axis(side = 1, at = 1:7, labels = names(day.means))
```



3/ Conditional Expectations with Continuous Covariates

Continuous covariate (I): each unique value gets a mean

- What if X_i is continuous? Can we calculate a mean for every value of X_i?
- Not really, because remember the probability that two values will be the same in a continuous variable is 0.
- Thus, we'll end up with a very "jumpy" function, $\widehat{\mathbb{E}}[Y_i|X_i = x]$, since n_x will be at most 1 for any value of x.

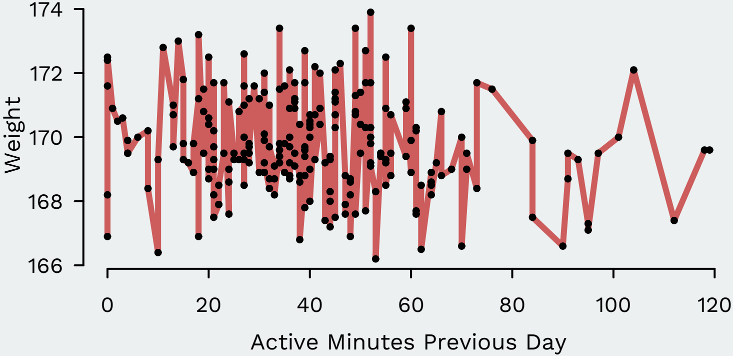
Continuous covariate (I) example

- I also wear an activity tracker and that collects how active I am during the day
- Let's look at the relationship between my weight and my active minutes in the previous day using this approach.

```
fitbit <- read.csv("fitbit.csv", stringsAsFactors = FALSE)
fitbit$date <- as.Date(fitbit$date, format = "%m/%d/%y")
## lag fitbit by one day
fitbit$date <- fitbit$date + 1
## merge fitbit and weight data
weight <- merge(weight, fitbit, by = "date")
```

Continuous covariate (I) CEF plot

```
plot(weight$active.mins[order(weight$active.mins)],
     weight$weight[order(weight$active.mins)], type = "l", lwd = 3, pch = 19,
     col = "indianred",xlab = "Active Minutes Previous Day", ylab = "Weight")
points(weight$active.mins, weight$weight, pch = 19, cex = 0.5)
```



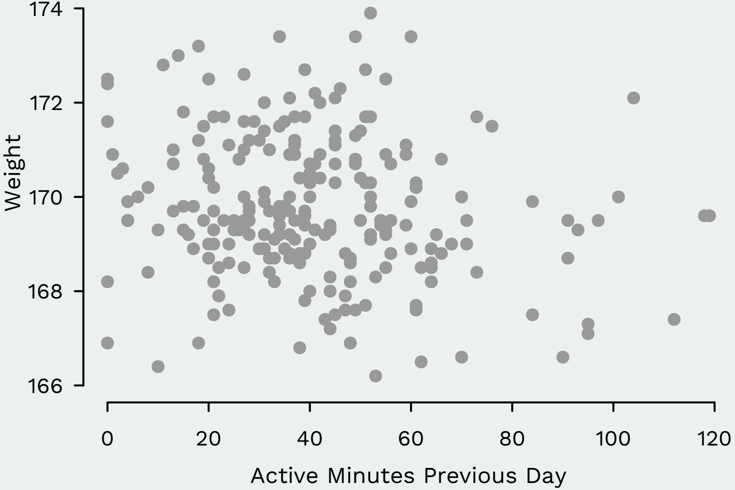
- The estimates, $\widehat{\mathbb{E}}[Y_i|X_i = x]$, will jump around a lot from sample to sample and have high sampling variance.

Continuous covariate (II): stratify and take means

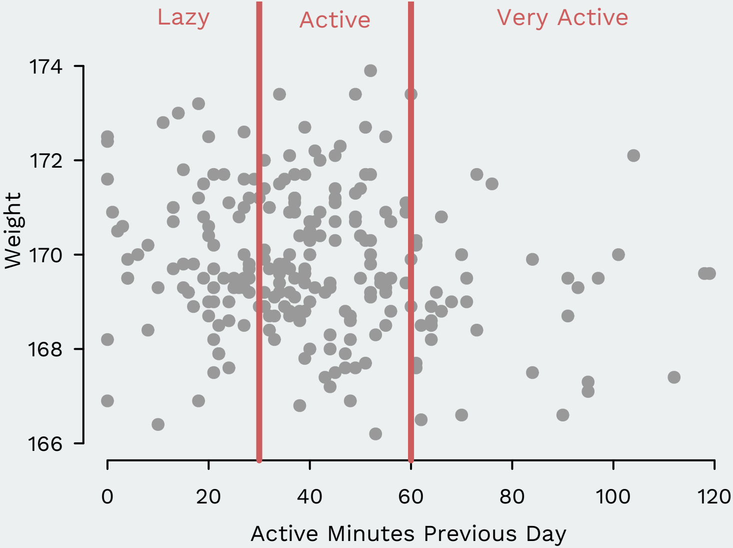
- So, that seems like each value of X_i won't work, but maybe we can take the continuous variable and turn it into a discrete variable. We call this **stratification**.
- Once it's discrete, we can just calculate the means within each **strata**.
- For instance, we could break up the "Active Minutes" variable into 3 categories: lazy ($<$ 30mins), active (30-60mins), and very active ($>$60min).

```
lowactivity.mean <- mean(weight$weight[weight$active.mins < 30])
medactivity.mean <- mean(weight$weight[weight$active.mins >= 30 & weight$active.mins
    60])
hiactivity.mean <- mean(weight$weight[weight$active.mins >= 60])
```

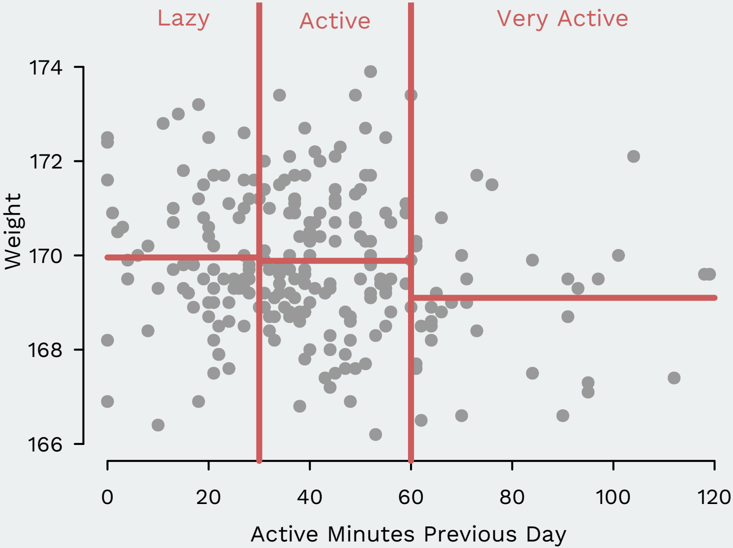
Continuous covariate (II) stratified CEF



Continuous covariate (II) stratified CEF



Continuous covariate (II) stratified CEF



Continuous covariate (III): model relationship as a line

- The stratification approach was fairly crude: it assumed that means were constant within strata, but that seems wrong.
- Can we get a more global model for the regression function? Well, maybe we could **assume** that it is linear:

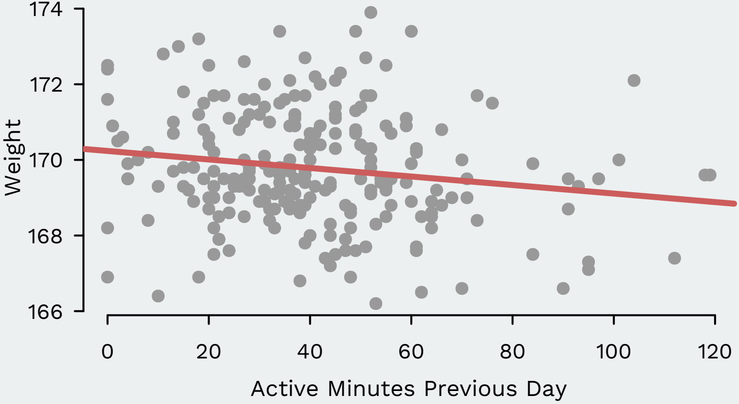
$$\mathbb{E}[Y_i|X_i = x] = \beta_0 + \beta_1 x$$

- Why might we do this? Parsimony, first and foremost: 2 numbers to predict any value.
- Some other nice properties we'll talk about in the coming weeks.

Continuous covariate (III)

- Estimated linear CEF:

```r
plot(weight$active.mins, weight$weight, pch = 19, col = "grey60",
     xlab = "Active Minutes Previous Day", ylab = "Weight")
abline(lm(weight ~ active.mins, data = weight), col = "indianred", lwd = 3)
```



Interpretation of the regression slope

- When we model the regression function as a line, we can interpret the parameters of the line in appealing ways:

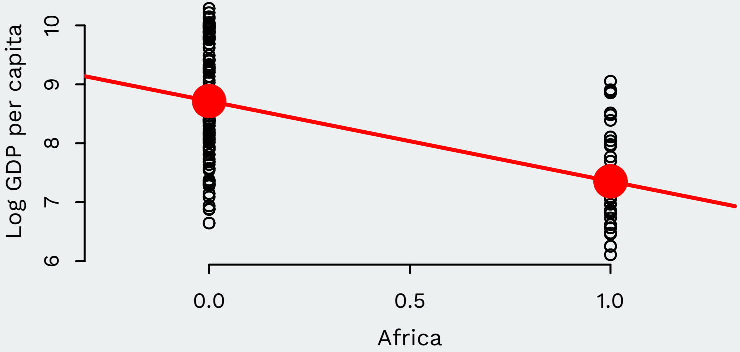
  1. **Intercept**: the average outcome among units with $X_i = 0$ is β_0:
  $$\mathbb{E}[Y_i|X_i = 0] = \beta_0 + \beta_1 0 = \beta_0$$
  2. **Slope**: a one-unit change in X_i is associated with a β_1 change in Y_i
  $$\begin{aligned}\mathbb{E}[Y_i|X_i = x+1] - \mathbb{E}[Y_i|X_i = x] &= (\beta_0 + \beta_1(x+1)) - (\beta_0 + \beta_1 x) \\ &= \beta_0 + \beta_1 x + \beta_1 - \beta_0 - \beta_1 x \\ &= \beta_1\end{aligned}$$

Linear regression with a binary covariate

- Using the two facts above, it's easy to see that when X_i is binary, then we have the following:
  1. **Intercept**: $\mathbb{E}[Y_i|X_i = 0] = \beta_0$
  2. **Slope**: average difference between $X_i = 1$ group and $X_i = 0$ group: $\beta_1 = \mathbb{E}[Y_i|X_i = 1] - \mathbb{E}[Y_i|X_i = 0]$
- Thus, we can read off the difference in means between two groups as the slope coefficient on a linear regression

Linear CEF with a binary covariate

```r
plot(ajr$africa, ajr$logpgp95, xlab = "Africa", ylab = "Log GDP per capita",
    xlim = c(-0.25, 1.25), bty = "n")
points(x = 0, y = mean(ajr$logpgp95[ajr$africa == 0], na.rm = TRUE),
    pch = 19, col = "red", cex = 3)
points(x = 1, y = mean(ajr$logpgp95[ajr$africa == 1], na.rm = TRUE),
    pch = 19, col = "red", cex = 3)
abline(lm(logpgp95 ~ africa, data = ajr), col = "red", lwd = 2)
```



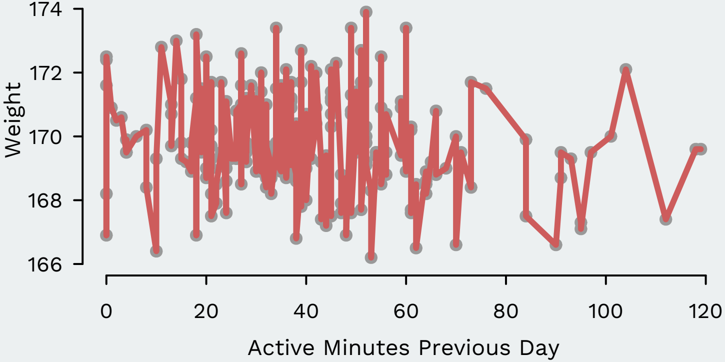
Parametric vs. nonparametric models

$$\widehat{\mathbb{E}}[Y_i|X_i = x] = \frac{1}{n_x} \sum_{i:X_i=x} Y_i$$

- Conditional sample mean: **nonparametric** because there are no assumptions about how $\mathbb{E}[Y_i|X_i = x]$ changes as we change x.
  - We just estimate the mean among each value of x.
  - Breaks down with continuous independent variables.
- A **parametric model** makes assumptions about the functional form of $\mathbb{E}[Y_i|X_i = x]$.
  - Suppose we assume the linear model $\mathbb{E}[Y_i|X_i = x] = \beta_0 + \beta_1 x$.
  - We are **assuming** that $\mathbb{E}[Y_i|X_i = x + 1] - \mathbb{E}[Y_i|X_i = x] = \beta_1$ at every value of x.

Bias-variance tradeoff

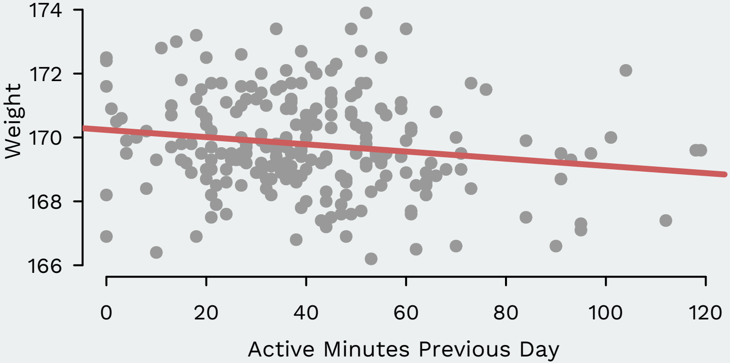
- How we model the regression function, $\mathbb{E}[Y_i|X_i = x]$, affects our the behavior of our estimates:



- Low bias (function "nails" every point)
- High variance (drastic changes from sample to sample)

Bias-variance tradeoff

- How we model the regression function, $\mathbb{E}[Y_i|X_i = x]$, affects our the behavior of our estimates:



- Higher bias (misses "local" variation)
- Low variance (slope and intercept will only change slightly from sample to sample)

4/ Lines of Best Fit

Back up and review

- To review our approach:
  - We wanted to estimate the CEF/regression function $\mathbb{E}[Y_i|X_i = x]$, but found that it was hard to do nonparametrically
  - So we're going to *model* it: place restrictions on its functional.
  - Easiest functional form is a line:

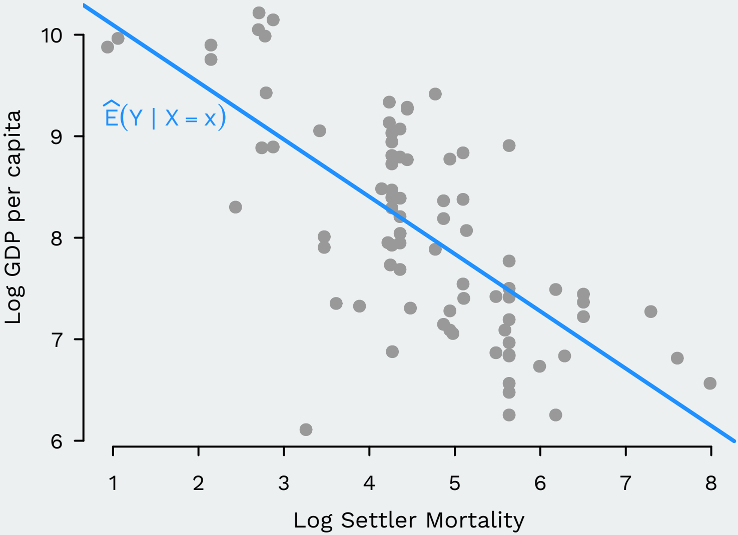
$$\mathbb{E}[Y_i|X_i = x] = \beta_0 + \beta_1 x$$

- β_0 and β_1 are population parameters just like μ or σ^2!
- Need to estimate them in our samples! But how?

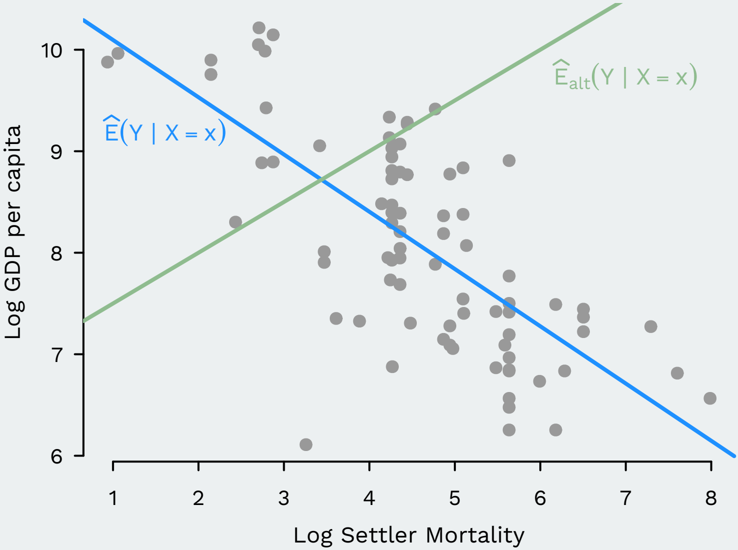
Simple linear regression model

- We'll need some terms and concepts first. Let's write our model:
$$Y_i = \mathbb{E}[Y_i|X_i = x] + u_i$$
$$Y_i = \beta_0 + \beta_1 X_i + u_i$$
- Now, suppose we have some estimates of the slope, $\hat{\beta}_1$, and the intercept, $\hat{\beta}_0$. Then the fitted or sample regression line is
$$\widehat{\mathbb{E}}[Y_i|X_i = x] = \widehat{\beta}_0 + \widehat{\beta}_1 x$$
- We want our estimate to predict outcomes very well so that $(Y_i - \widehat{\mathbb{E}}[Y_i|X_i])$ are small.

Fitted linear CEF



Fitted linear CEF



Fitted values and residuals

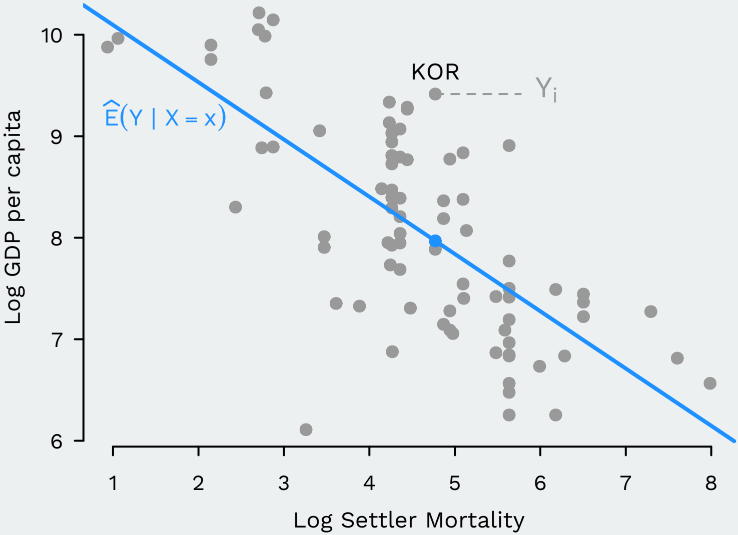
- **Definition** A **fitted value** or **predicted value** is the estimated conditional mean of Y_i for a particular observation with independent variable X_i:

$$\widehat{Y}_i = \widehat{\mathbb{E}}[Y_i|X_i] = \widehat{\beta}_0 + \widehat{\beta}_1 X_i$$

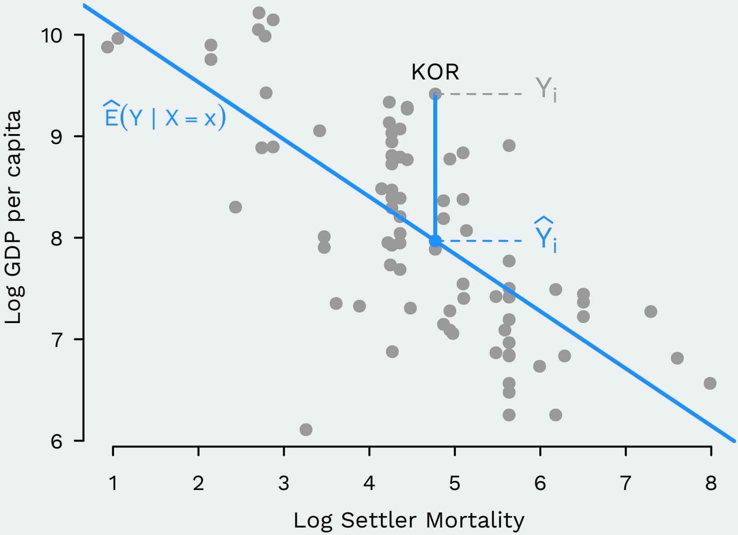
- **Definition** The **residual** is the difference between the actual value of Y_i and the predicted value, $\widehat{Y}_i$:

$$\widehat{u}_i = Y_i - \widehat{Y}_i = Y_i - \widehat{\beta}_0 - \widehat{\beta}_1 X_i$$

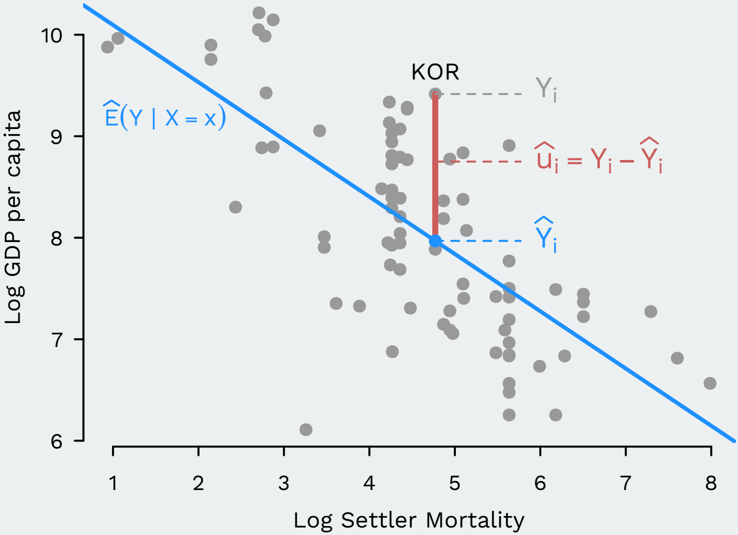
Fitted linear CEF



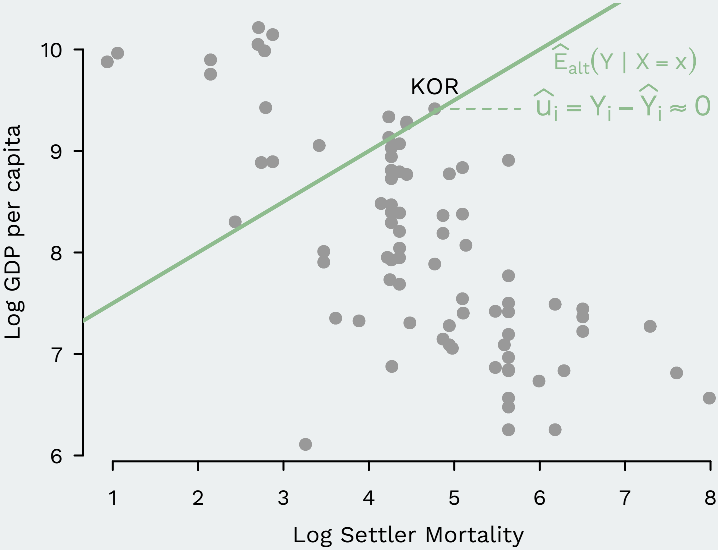
Fitted linear CEF



Fitted linear CEF



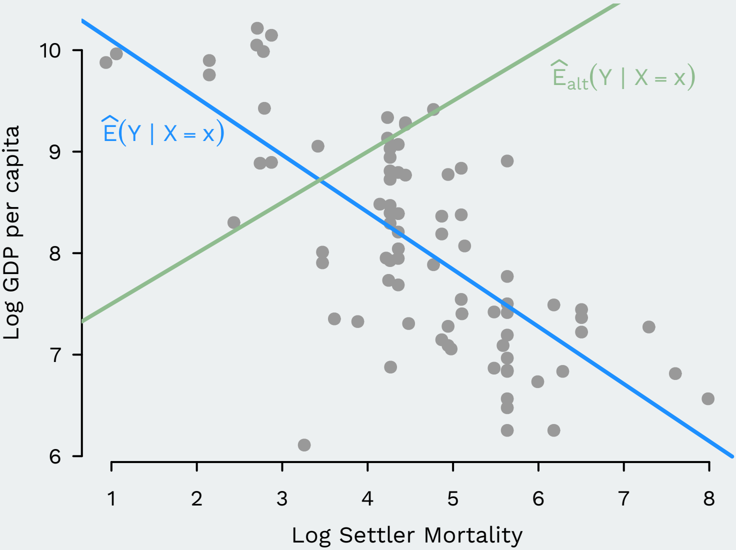
Why not this line?



Minimize the residuals

- The residuals, $\widehat{u}_i = Y_i - \widehat{\beta}_0 - \widehat{\beta}_1 X_i$, tell us how well the line fits the data.
  - Larger magnitude residuals means that points are very far from the line
  - Residuals close to 0 mean points very close to the line
- The smaller the magnitude of the residuals, the better we are doing at predicting Y_i
- Choose the line that minimizes the residuals

Which is better at minimizing residuals?



5/ Least Squares

Minimizing the residuals

- Let $\tilde{\beta}_0$ and $\tilde{\beta}_1$ be possible values of the intercept and slope
- **Least absolute deviations** (LAD) regression:

$$(\widehat{\beta}_0^{LAD}, \widehat{\beta}_1^{LAD}) = \underset{\tilde{\beta}_0, \tilde{\beta}_1}{\arg\min} \sum_{i=1}^{n} |Y_i - \tilde{\beta}_0 - \tilde{\beta}_1 X_i|$$

- **Least squares** (LS) regression:

$$(\widehat{\beta}_0, \widehat{\beta}_1) = \underset{\tilde{\beta}_0, \tilde{\beta}_1}{\arg\min} \sum_{i=1}^{n} (Y_i - \tilde{\beta}_0 - \tilde{\beta}_1 X_i)^2$$

- Sometimes called **ordinary least squares** (OLS)

Why least squares?

Figure: Our man Gauss

- Easy to derive a closed-form expression for the least squares estimator.
- East to investigate the properties of the least squares estimator.
- Least squares is optimal in a certain sense that we'll see in the coming weeks.

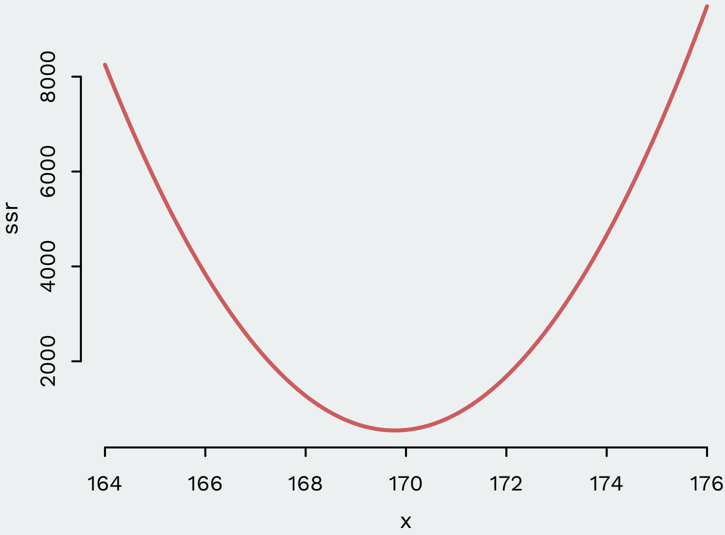
Least squares and the mean

- Let's derive a simpler least squares estimator first, for $\widehat{E}[Y_i]$.
- $\widehat{\mathbb{E}}[Y_i]$ should be a good predictor of Y_i.
- $\rightsquigarrow$ find the value that minimizes the **sum of squared residuals** (SSR)

$$S(\tilde{\mu}) = \sum_{i=1}^{n}(Y_i - \tilde{\mu})^2$$

- How do we solve this?
  1. Calculate the derivative of S with respect to $\tilde{\mu}$
  2. Set the derivative equal to 0
  3. Solve for $\tilde{\mu}$ and replace $\tilde{\mu}$ with the solution
- What does the sum of the squared residuals (SSR) function look like?

Sum of the squared residuals function



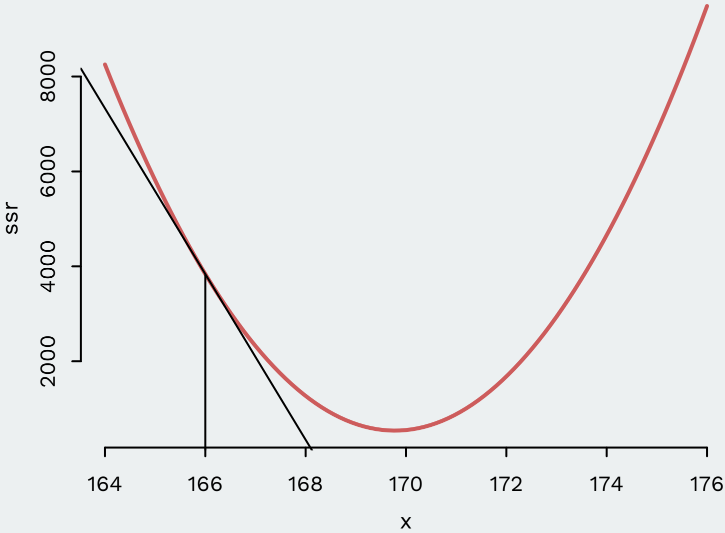
Minimize the SSR

1. Calculate the derivative

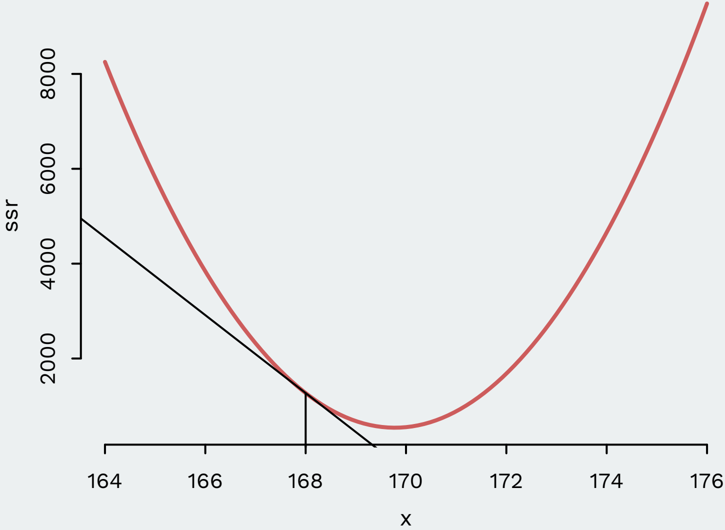
$$
\begin{aligned}
S(\tilde{\mu}) &= \sum_{i=1}^{n} (Y_i - \tilde{\mu})^2 \\
&= \sum_{i=1}^{n} (Y_i^2 - 2Y_i\tilde{\mu} + \tilde{\mu}^2)
\end{aligned}
$$

$$
\frac{\partial S(\tilde{\mu})}{\partial \tilde{\mu}} = \sum_{i=1}^{n} (-2Y_i + 2\tilde{\mu}) \qquad \text{(linearity + product rule)}
$$

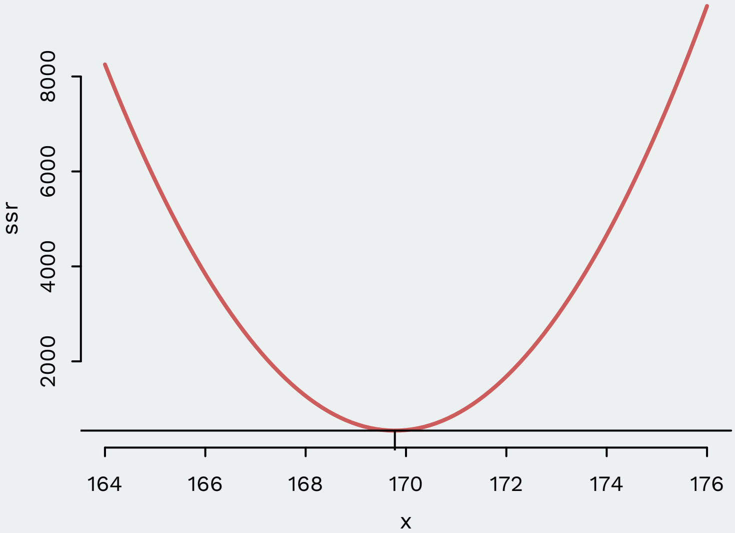
Sum of the squared residuals derivative



Sum of the squared residuals derivative



Sum of the squared residuals derivative



Minimize the SSR

1. Calculate the derivative

$$S(\tilde{\mu}) = \sum_{i=1}^{n}(Y_i - \tilde{\mu})^2$$

$$\frac{\partial S(\tilde{\mu})}{\partial \tilde{\mu}} = \sum_{i=1}^{n}(-2Y_i + 2\tilde{\mu})$$

2. Setting it to zero:

$$0 = \sum_{i=1}^{n}(-2Y_i + 2\tilde{\mu})$$

3. And solve:

$$\widehat{\mu} \;\equiv\; \tilde{\mu} \;=\; \frac{1}{n}\sum_{i=1}^{n} Y_i$$

Deriving the OLS estimator

- Now we want to estimate $\mathbb{E}[Y_i|X_i = x] = \beta_0 + \beta_1 x$.
- $\widehat{\mathbb{E}}[Y_i|X_i] = \widehat{\beta}_0 + \widehat{\beta}_1$ should be a good predictor of Y_i
- Let $\{\tilde{\beta}_0, \tilde{\beta}_1\}$ be candidate estimates for $\{\beta_0, \beta_1\}$
- Define the least squares objective function:

$$S(\tilde{\beta}_0, \tilde{\beta}_1) = \sum_{i=1}^{n} (Y_i - \tilde{\beta}_0 - \tilde{\beta}_1 X_i)^2.$$

- How do we derive the LS estimators for β_0 and β_1?
    1. Take partial derivatives of S with respect to $\tilde{\beta}_0$ and $\tilde{\beta}_1$.
    2. Set each of the partial derivatives to 0
    3. Solve for $\{\tilde{\beta}_0, \tilde{\beta}_1\}$ and replace them with the solutions

Taking the partial derivatives

$$
\begin{aligned}
S(\tilde{\beta}_0, \tilde{\beta}_1) &= \sum_{i=1}^{n} (Y_i - \tilde{\beta}_0 - \tilde{\beta}_1 X_i)^2 \qquad \text{(the SSR)} \\
&= \sum_{i=1}^{n} (Y_i^2 - 2Y_i\tilde{\beta}_0 - 2Y_i\tilde{\beta}_1 X_i + \tilde{\beta}_0^2 + 2\tilde{\beta}_0\tilde{\beta}_1 X_i + \tilde{\beta}_1^2 X_i^2) \\
&\text{(taking the product)}
\end{aligned}
$$

- Taking partial derivatives:

$$
\begin{aligned}
\frac{\partial S(\tilde{\beta}_0, \tilde{\beta}_1)}{\partial \tilde{\beta}_0} &= \sum_{i=1}^{n} (-2Y_i + 2\tilde{\beta}_0 + 2\tilde{\beta}_1 X_i) \\
\frac{\partial S(\tilde{\beta}_0, \tilde{\beta}_1)}{\partial \tilde{\beta}_1} &= \sum_{i=1}^{n} (-2Y_i X_i + 2\tilde{\beta}_0 X_i + 2\tilde{\beta}_1 X_i^2)
\end{aligned}
$$

First order conditions

- The first order conditions are when we set the derivatives equal to 0:

$$0 = \sum_{i=1}^{n}(-2Y_i + 2\tilde{\beta}_0 + 2\tilde{\beta}_1 X_i)$$

$$0 = \sum_{i=1}^{n}(-2Y_i X_i + 2\tilde{\beta}_0 X_i + 2\tilde{\beta}_1 X_i^2)$$

- Now solving for $\tilde{\beta}_0$ and $\tilde{\beta}_1$ yields the **normal equations**:

$$\widehat{\beta}_0 n = \left(\sum_{i=1}^{n} Y_i\right) - \widehat{\beta}_1 \left(\sum_{i=1}^{n} X_i\right)$$

$$\widehat{\beta}_1 \sum_{i=1}^{n} X_i^2 = \left(\sum_{i=1}^{n} X_i Y_i\right) - \widehat{\beta}_0 \left(\sum_{i=1}^{n} X_i\right)$$

Normal equations and the OLS estimator

- We can take the **normal equations**:

$$\hat{\beta}_0 n = \left(\sum_{i=1}^{n} Y_i\right) - \hat{\beta}_1 \left(\sum_{i=1}^{n} X_i\right)$$

$$\hat{\beta}_1 \sum_{i=1}^{n} X_i^2 = \left(\sum_{i=1}^{n} X_i Y_i\right) - \hat{\beta}_0 \left(\sum_{i=1}^{n} X_i\right)$$

- And rearrange them to get the **OLS estimators**:

$$\hat{\beta}_0 = \overline{Y} - \hat{\beta}_1 \overline{X}$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sum_{i=1}^{n}(X_i - \overline{X})^2}$$

- Let's see how we get these!

Normal equations to OLS estimators

- For the intercept, just divide by n:

$$\widehat{\beta}_0 n = \left(\sum_{i=1}^{n} Y_i\right) - \widehat{\beta}_1 \left(\sum_{i=1}^{n} X_i\right)$$

$$\widehat{\beta}_0 \frac{n}{n} = \left(\frac{1}{n}\sum_{i=1}^{n} Y_i\right) - \widehat{\beta}_1 \left(\frac{1}{n}\sum_{i=1}^{n} X_i\right)$$

$$\widehat{\beta}_0 = \overline{Y} - \widehat{\beta}_1 \overline{X}$$

Normal equations to OLS estimators

- Now, for the slope, we need to rearrange a bit:

$$\widehat{\beta}_0 = \overline{Y} - \widehat{\beta}_1 \overline{X}$$

$$\widehat{\beta}_0 \left(\sum_{i=1}^{n} X_i \right) = \overline{Y} \left(\sum_{i=1}^{n} X_i \right) - \widehat{\beta}_1 \overline{X} \left(\sum_{i=1}^{n} X_i \right)$$

- Plug this into the second normal equation:

$$\widehat{\beta}_1 \sum_{i=1}^{n} X_i^2 = \left(\sum_{i=1}^{n} X_i Y_i \right) - \widehat{\beta}_0 \left(\sum_{i=1}^{n} X_i \right)$$

$$\widehat{\beta}_1 \sum_{i=1}^{n} X_i^2 = \left(\sum_{i=1}^{n} X_i Y_i \right) - \overline{Y} \left(\sum_{i=1}^{n} X_i \right) + \widehat{\beta}_1 \overline{X} \left(\sum_{i=1}^{n} X_i \right)$$

$$\widehat{\beta}_1 \sum_{i=1}^{n} X_i^2 = \left(\sum_{i=1}^{n} X_i Y_i \right) - \left(\sum_{i=1}^{n} \overline{Y} X_i \right) + \widehat{\beta}_1 \left(\sum_{i=1}^{n} \overline{X} X_i \right)$$

- Let's rearrange:

$$\hat{\beta}_1 \sum_{i=1}^{n} X_i^2 = \left(\sum_{i=1}^{n} X_i Y_i\right) - \left(\sum_{i=1}^{n} \overline{Y} X_i\right) + \hat{\beta}_1 \left(\sum_{i=1}^{n} \overline{X} X_i\right)$$

$$\hat{\beta}_1 \sum_{i=1}^{n} \left(X_i^2 - \overline{X} X_i\right) = \sum_{i=1}^{n} \left(X_i Y_i - \overline{Y} X_i\right)$$

$$\hat{\beta}_1 \sum_{i=1}^{n} X_i \left(X_i - \overline{X}\right) = \sum_{i=1}^{n} X_i (Y_i - \overline{Y})$$

- Remember that deviations from the mean sum to 0: $\sum_{i=1}^{n} (Z_i - \overline{Z}) = 0$

$$\hat{\beta}_1 \sum_{i=1}^{n} X_i \left(X_i - \overline{X}\right) - \underbrace{\hat{\beta}_1 \overline{X} \sum_{i=1}^{n} \left(X_i - \overline{X}\right)}_{=0} = \sum_{i=1}^{n} X_i (Y_i - \overline{Y}) - \underbrace{\overline{X} \sum_{i=1}^{n} (Y_i - \overline{Y})}_{=0}$$

$$\hat{\beta}_1 \sum_{i=1}^{n} \left(X_i (X_i - \overline{X}) - \overline{X}(X_i - \overline{X})\right) = \sum_{i=1}^{n} \left(X_i (Y_i - \overline{Y}) - \overline{X}(Y_i - \overline{Y})\right)$$

$$\hat{\beta}_1 \sum_{i=1}^{n} \left(X_i - \overline{X}\right) \left(X_i - \overline{X}\right) = \sum_{i=1}^{n} (X_i - \overline{X}_i)(Y_i - \overline{Y})$$

OLS estimators

- Isolate $\widehat{\beta}_1$ to get the **OLS estimator** for the slope:

$$\widehat{\beta}_1 = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sum_{i=1}^{n}(X_i - \overline{X})^2}$$

- Note that this is the following:

$$\widehat{\beta}_1 = \frac{\text{Sample Covariance between } X \text{ and } Y}{\text{Sample Variance of } X}$$

- Combine this with the intercept estimator:

$$\widehat{\beta}_0 = \overline{Y} - \widehat{\beta}_1 \overline{X}$$

AJR Example in R

- Let’s use those simple formulas we just learned:

```
ajr <- na.omit(ajr[, c("logem4", "logpgp95")])
cov.xy <- cov(ajr$logem4, ajr$logpgp95)
var.x <- var(ajr$logem4)
cov.xy/var.x
```

```
## [1] -0.5641
```

```
mean(ajr$logpgp95) - cov.xy/var.x * mean(ajr$logem4)
```

```
## [1] 10.66
```

- Compare it to what `lm()`, the OLS function in R produces:

```
coef(lm(logpgp95 ~ logem4, data = ajr))
```

```
## (Intercept)      logem4
##     10.6602     -0.5641
```

Mechanical properties of least squares

- The residuals will be 0 on average:

$$\sum_{i=1}^{n} \widehat{u}_i = 0$$

- The residuals will be uncorrelated with the predictor:

$$\sum_{i=1}^{n} X_i \widehat{u}_i = 0 \rightsquigarrow \widehat{\text{Cov}}(X_i, \widehat{u}_i) = 0$$

- The residuals will be uncorrelated with the fitted values:

$$\sum_{i=1}^{n} \widehat{Y}_i \widehat{u}_i = 0 \rightsquigarrow \widehat{\text{Cov}}(\widehat{Y}_i, \widehat{u}_i) = 0$$

Mechanical properties of least squares in R

```
mod <- lm(logpgp95 ~ logem4, data = ajr)
mean(residuals(mod))
```

```
## [1] -2.624e-18
```

```
cor(ajr$logem4, residuals(mod))
```

```
## [1] -3.185e-17
```

```
cor(fitted(mod), residuals(mod))
```

```
## [1] -1.16e-16
```