6.3 – OLS in multiple regression

The population model:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_k X_{ki} + \epsilon_i \tag{6.1}$$

How to estimate the βs?

- Still want to minimize the sum of squared residuals (the sum of "vertical distances"):

$$\min_{b_0, b_1, \ldots, b_k} \sum_{i=1}^{n} e_i^2$$

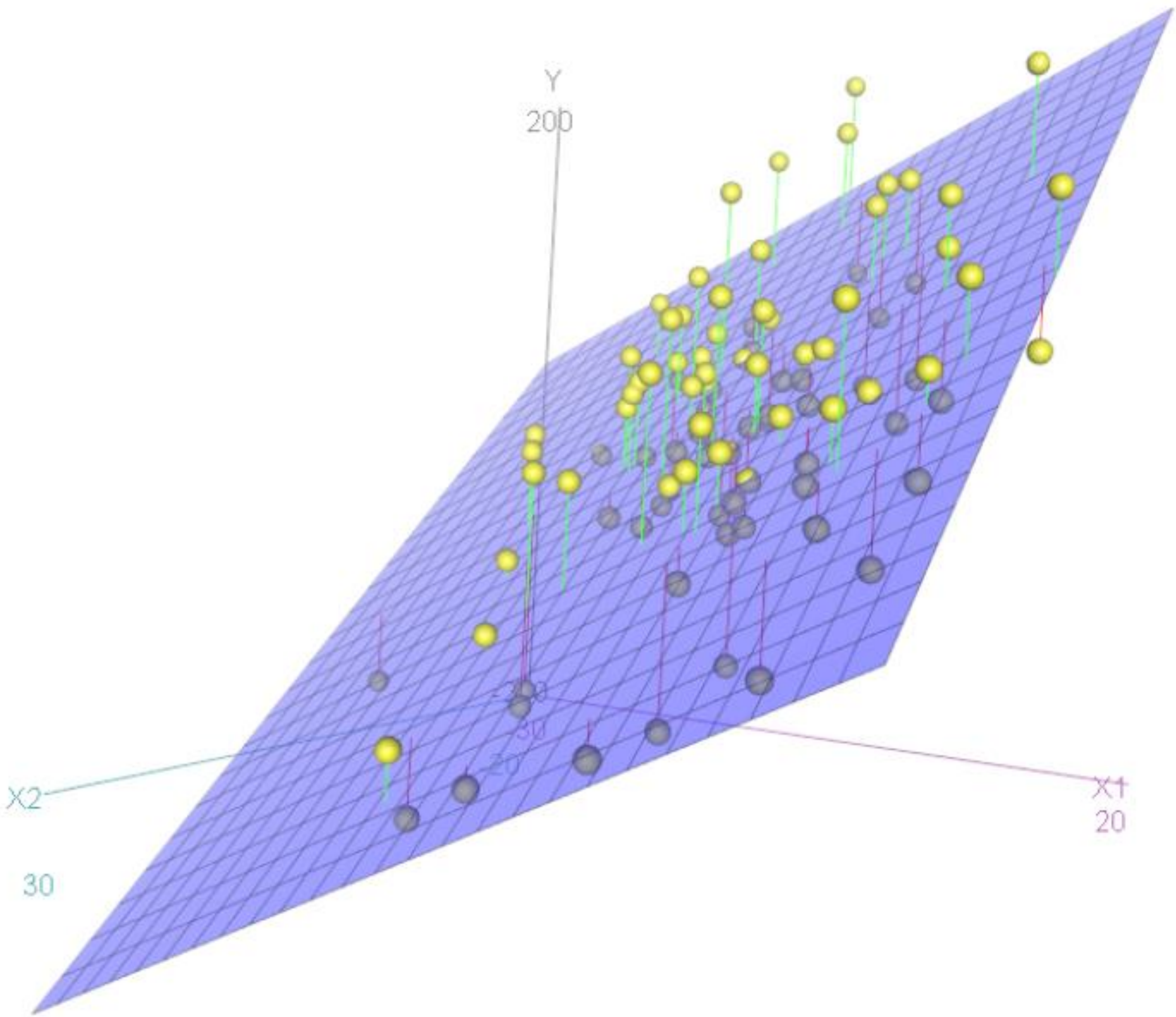
- Take (k + 1) derivatives, set them equal to zero, solve
- The new formula is too difficult to show (unless we use matrices, which we won't)

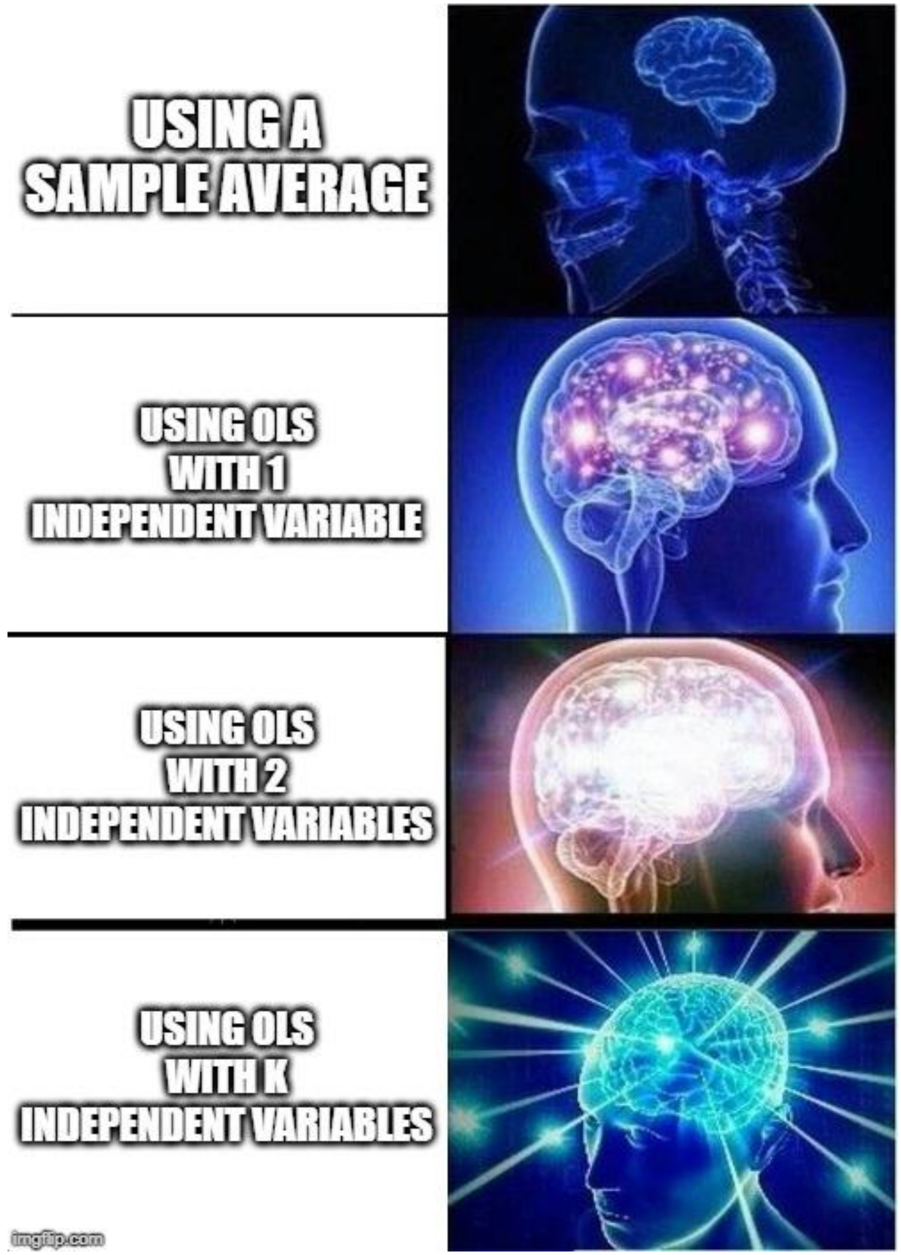
The resulting estimated model:

$$\hat{Y}_i = b_0 + b_1 X_{1i} + b_2 X_{2i} + \cdots + b_k X_{ki} \tag{6.2}$$

can't be interpreted as a line! (It's a k-dimensional hyperplane).

We can still try to visualize things if we have only 2 X variables, however:


Y
200
X2
30
20
X1
20


USING A
SAMPLE AVERAGE
USING OLS
WITH 1
INDEPENDENT VARIABLE
USING OLS
WITH 2
INDEPENDENT VARIABLES
USING OLS
WITH K
INDEPENDENT VARIABLES
imgflip.com

6.3.2 Interpretation

Let's look at a population model with two X variables:

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \epsilon_i \tag{6.3}$$

- Y is still the dependent variable
- X_1 and X_2 are the independent variables (the regressors)
- i still denotes an observation number
- β_0 is the population intercept
- β_1 is the effect of X_1 on Y, holding all else constant (X_2)
- β_2 is the effect of X_2 on Y, holding all else constant (X_1)
- ϵ is the regression error term (containing all the omitted factors that effect Y)

6.4: A2 – No perfect multicollinearity

Now that we have multiple *X* variables in our model, we need to make an additional assumption in order for OLS to work:

There is no perfect multicollinearity. This means:

- No two variables (or combinations of variables) are exactly linearly related
- No two variables are perfectly correlated

For example, exact linear relationships between Xs are:

- $X_1 = X_2$
- $X_1 = 100X_2$
- $X_1 = 1 + X_2 - 3X_3$

If you know X_1, you know X_2 in first two examples).

Including both variables would be redundant.

OLS can't handle it. (Like dividing by zero).

Some common examples of where the assumption of "no perfect multicollinearity" is violated in practice are when the same variable is measure in different units (such as square feet and square metres, or dollars and cents), and in the *dummy variable trap*.

The `Living.Area` variable measures the size of the house in square feet. Suppose that there was another variable in the data set that measured house size in square metres (1 square foot = 0.0929 square metre). We can create this variable in R using:

```
House.Size <- 0.0929 * Living.Area
```

and now let's include it in our OLS estimation:

```
summary(lm(Price ~ Fireplaces + Living.Area + House.Size))
```

```
Coefficients: (1 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 14.730146   5.007563   2.942  0.00331 **
Fireplaces   8.962440   3.389656   2.644  0.00827 **
Living.Area  0.109313   0.003041  35.951  < 2e-16 ***
House.Size         NA         NA      NA       NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 68980 on 1725 degrees of freedom
Multiple R-squared:  0.5095,	Adjusted R-squared:  0.5089
F-statistic: 895.9 on 2 and 1725 DF,  p-value: < 2.2e-16
```

6.4.1 The dummy variable trap

The dummy variable trap occurs when one too many dummy variables are included in the equation. For example, suppose that we have a dummy variable `female` that equals 1 if the worker is female. Suppose that we also have a variable `male` that equals 1 if the worker is male. There is an exact linear combination between the two variables:

$$female = 1 - male$$

OLS won't work for:

$$wage = \beta_0 + \beta_1 \times male + \beta_2 \times female + \epsilon$$

Much easier to fall into the trap for "categorical variables"

$$
\begin{aligned}
Alberta &= 1 \text{ if } Location = AB; 0 \text{ otherwise} \\
British.Columbia &= 1 \text{ if } Location = BC; 0 \text{ otherwise} \\
Manitoba &= 1 \text{ if } Location = MB; 0 \text{ otherwise} \\
&\vdots \\
Yukon &= 1 \text{ if } Location = YT; 0 \text{ otherwise}
\end{aligned}
$$

- We would create 13 dummy variables using “location”, but only include 12 of them in our equation
- The group that is left out becomes the “base group”
- We could also drop the intercept (but this isn’t usually done)

Final note:

Non-linear transformations are ok! We will do this in chapter 8.

6.4.2 Imperfect multicollinearity

Imperfect multicollinearity is when two (or more) variables are *almost* perfectly related (highly correlated).

Example

Pretend we know the pop. model:

$$Y = 2X_1 + 2X_2 + \epsilon$$

and that the correlation between X_1 and X_2 is 0.99.

```
summary(lm(Y ~ X1))
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -4.4165     3.8954  -1.134    0.263
X1            4.0762     0.4698   8.676 2.13e-11 ***
```

The estimated standard error is small, so that the t-statistic is large, and we are sure that X_1 is statistically significant. However, the estimated β_1 is twice as big as it should be. This is because of omitted variable bias.

```r
summary(lm(Y ~ X1 + X2))
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   -4.676      3.956  -1.182    0.243
X1             1.958      4.075   0.481    0.633
X2             2.128      4.066   0.523    0.603
```

Now, the estimated βs are closer to their true value of 2, but both appear to be statistically insignificant! (Note the large standard errors and small t-statistics.)

The problem:

- Because X_1 and X_2 are correlated, difficult to attribute changes in X_1 to changes in Y (same for X_2)
- X_1 and X_2 are almost always changing together in a similar way
- *ceteris paribus* assumption is not feasible
- β_1 is the effect of X_1 on Y, holding X_2 constant

How imperfect multicollinearity affects estimation

- large standard errors, wide confidence intervals
- adding and dropping variables results in large swings of the estimated values
- overall – makes us unsure about our results
- problem is difficult to address
- can’t drop variables (OVB)
- if you don’t need to interpret the affected variables, it’s not a problem