Ordinary Least Squares

Chi-Chao Hung *

latest updated: April 3, 2025

Contents

*This note borrows heavily from Hansen (2022). I was also supported by ChatGPT. All errors are mine. If you find any errors or have any suggestions, please contact me via email at r13323021@ntu.edu.tw.

1 Population and Sample

In Lecture Note 1, we discussed the linear regression from the population point of view. The outcome variable Y is a random variable, and the regressors X is a random vector containing k random variables.

$$Y, \quad X = \begin{pmatrix} X^1 \\ \vdots \\ X^k \end{pmatrix}$$

The best linear predictor (BLP) is $X'\beta$ where β is given by

$$\beta = \mathrm{E}[XX']^{-1}\,\mathrm{E}[XY] \tag{1}$$

Now we consider a random sample drawn from the population. The sample is a realization of the random variables (Y, X). The sample consists of n observations $(Y_1, X_1), \ldots, (Y_n, X_n)$. For example, the i-th observation is (Y_i, X_i), where

$$Y_i = Y_i, \quad X_i = \begin{pmatrix} X_i^1 \\ \vdots \\ X_i^k \end{pmatrix}$$

Definition 1.1 (Sample Mean)**.** The sample mean of a random vector X is defined as

$$\bar{X} := \frac{1}{n}\sum_{i=1}^{n} X_i. \tag{2}$$

Here we provide some more useful natations. The sample mean of the random vector XY is defined as

$$\underset{k\times 1}{\hat{\mathbf{Q}}_{XY}} := \frac{1}{n}\sum_{i=1}^{n} X_i Y_i. \tag{3}$$

The sample mean of the random matrix XX' is defined as

$$\underset{k\times k}{\hat{\mathbf{Q}}_{XX}} := \frac{1}{n}\sum_{i=1}^{n} X_i X_i'. \tag{4}$$

Remark 1. Since the true BLP is unobservable, our goal is to *estimate* β using the random sample. Figure 1 shows the big picture of the estimation problem. In order to estimate β, we need to construct an *estimator* $\hat{\beta}$ of β.

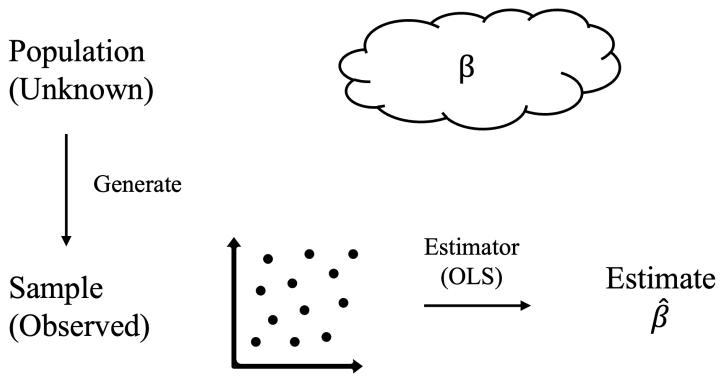


Figure 1: The Big Picture

2 Ordinary Least Square

2.1 Least Square Estimator

Recall the BLP problem in Lecture Note 1.

$$S(\beta) := \mathrm{E}\left[(Y - X'\beta)^2\right] \tag{5}$$

$$\beta = \arg\min_{\beta \in \mathbb{R}^k} S(\beta). \tag{6}$$

The least square problem is the "sample analogues" of the best linear predictor problem.

$$\hat{S}(\beta) := \frac{1}{n}\sum_{i=1}^{n}(Y_i - X_i'\beta)^2 \tag{7}$$

$$\hat{\beta} = \arg\min_{\beta \in \mathbb{R}^k} \hat{S}(\beta). \tag{8}$$

Remark 2. What is $\hat{\beta}$? The multiple identities of $\hat{\beta}$:

- $\hat{\beta}$ is the *solution* to the least square problem.
- $\hat{\beta}$ is a function of the random sample $(Y_1, X_1), \ldots, (Y_n, X_n)$. Since (Y_i, X_i) are random, $\hat{\beta}$ is also a *random variable*.
- $\hat{\beta}$ is an *estimator* of β, the projection coefficient. β is non-random.
- After the sample is realized, $\hat{\beta}$ is a *estimate* (guess) for β.

This is an application of the *plug-in principle*. The estimator defined in equation 8 is also commonly refered to as the ordinary least squares (OLS) estimator. It turns out that finding the OLS estimator is equivalent to finding a best fitted line(or plane) through the data points.

2.2 OLS Derivation

The derivation of the OLS estimator is similar to the derivatio of the BLP in Lecture Note 1. I highly recommend you to review the derivation of the BLP before reading this section.

First, we define sum of squared errors (SSE) as

$$\text{SSE}(\beta) := \sum_{i=1}^{n} (Y_i - X_i'\beta)^2. \tag{9}$$

Expand the expression of SSE, we have

$$\text{SSE}(\beta) = \sum_{i=1}^{n} Y_i^2 - 2\beta' \sum_{i=1}^{n} X_i Y_i + \beta' \left(\sum_{i=1}^{n} X_i X_i' \right) \beta.$$

Take the first-order condition (FOC) with respect to β:

$$0 = \frac{\partial}{\partial \beta} \text{SSE}(\hat{\beta}) = -2 \sum_{i=1}^{n} X_i Y_i + 2 \sum_{i=1}^{n} X_i X_i' \hat{\beta}.$$

Solve for $\hat{\beta}$:

$$\sum_{i=1}^{n} X_i X_i' \hat{\beta} = \sum_{i=1}^{n} X_i Y_i. \tag{10}$$

If $\sum_{i=1}^{n} X_i X_i'$ is invertible, then we can solve for $\hat{\beta}$:

$$\hat{\beta} = \underset{(k \times k)}{\left(\sum_{i=1}^{n} X_i X_i' \right)^{-1}} \underset{k \times 1}{\left(\sum_{i=1}^{n} X_i Y_i \right)}. \tag{11}$$

The solution can also be written as the function of sample moments:

$$\hat{\beta} = \hat{\mathbf{Q}}_{XX}^{-1} \hat{\mathbf{Q}}_{XY}. \tag{12}$$

Comparing $\hat{\beta}$ with β, we can see that $\hat{\beta}$ is the sample analogue of β.

$$\beta = \mathbf{Q}_{XX}^{-1} \mathbf{Q}_{XY}.$$

2.2.1 Simple OLS

Let us consider the simple linear regression model where X_i is a random variable. The model includes a constant term α:

$$\mathcal{P}(Y_i \mid X_i) = \alpha + \beta X_i \tag{13}$$

We can derive the OLS estimator $\hat{\alpha}, \hat{\beta}$ using previous results.

$$\hat{\mathbf{Q}}_{XX} = \frac{1}{n}\begin{pmatrix} n & \sum_{i=1}^n X_i \\ \sum_{i=1}^n X_i & \sum_{i=1}^n X_i^2 \end{pmatrix},$$
$$\hat{\mathbf{Q}}_{XY} = \frac{1}{n}\begin{pmatrix} \sum_{i=1}^n Y_i \\ \sum_{i=1}^n X_i Y_i \end{pmatrix}$$

Equation 10 becomes

$$\begin{cases} \hat{\alpha} + \bar{X}_1\hat{\beta} = \bar{Y} \\ \bar{X}_1\hat{\alpha} + \bar{X}_1^2\hat{\beta} = \bar{X}_1 Y \end{cases}$$

The solution is then

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

2.3 OLS Residuals

As a by-product of estimation we define the fitted value and the residual.

Definition 2.1 (Fitted Value)**.** The fitted value is defined as

$$\hat{Y}_i = X_i'\hat{\beta}. \tag{14}$$

Definition 2.2 (OLS Residual)**.** The residual is defined as

$$\hat{e}_i = Y_i - \hat{Y}_i = Y_i - X_i'\hat{\beta}. \tag{15}$$

Remark 3. What is the difference between e_i and $\hat{e}_i$?
Notice that the projection **error** e_i is defined as

$$e_i = Y_i - X_i'\beta,$$

where $X_i'\beta$ is the linear CEF or BLP. Since CEF or BLP are unobservable, e_i is also unobservable. On the other hand, as shown if Figure 2.3, the **residual** $\hat{e}_i$ is the deviation of Y_i from the fitted value $\hat{Y}_i$. Since $\hat{Y}_i$ is calculated using the sample data, $\hat{e}_i$ is observable.
In future lessons, we will investigate the *variance* of the OLS estimator, which is related to e_i. We will use the residual to estimate the error.

Property 2.1. Let $\hat{e}_i$ be the residual.

$$\sum_{i=1}^n X_i\hat{e}_i = 0.$$

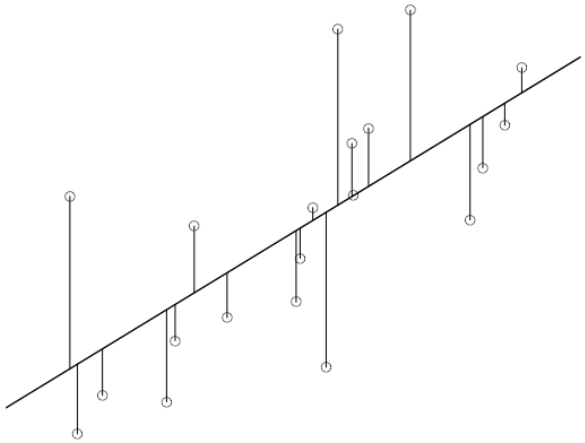


Figure 2: Illustration of Fitted Values and Residuals

Recall that in Lecture Note 1, we have shown that the projection error is orthogonal to the regressors.

$$\mathrm{E}[X_i e_i] = 0.$$

Property 2.1 is the sample analogue of this property.

Proof.

$$\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n} X_i \hat{e}_i &= \frac{1}{n}\sum_{i=1}^{n} X_i (Y_i - X_i'\hat{\beta}) \\
&= \hat{\mathbf{Q}}_{XY} - \hat{\mathbf{Q}}_{XX}\hat{\beta} \\
&= \hat{\mathbf{Q}}_{XY} - \hat{\mathbf{Q}}_{XX}\hat{\mathbf{Q}}_{XX}^{-1}\hat{\mathbf{Q}}_{XY} \\
&= \hat{\mathbf{Q}}_{XY} - \hat{\mathbf{Q}}_{XY} \\
&= 0.
\end{aligned}$$

□

3 OLS in Practice using R

You might be confused by all the math(there are more ahead) and wondering how to implement the OLS estimator in practice. In this section, we will show you how to use R to estimate the OLS estimator. We provide examples for two of the common usage of OLS, making **prediction** and **comparison**.

3.1 Predicting Miles per Galon of Cars

The data set `mtcars` was extracted from the 1974 Motor Trend US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

```
# Load necessary libraries
library(dplyr)

# View the first few rows of the dataset
head(mtcars)
```

```
##                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

Let's fit a simple linear regression model with the Weight(`wt`) as the regressor and Miles/(US) gallon(`mpg`) as the outcome variable.

```
# Fit simple linear regression model
ols1 <- lm(mpg ~ wt, data = mtcars)
coef(ols1)
```

```
## (Intercept)          wt
## 37.285126   -5.344472
```

We try adding more regressors to the model. `hp` is the Horsepower and `disp` is the Displacement.

```
# Fit multiple linear regression model
ols2 <- lm(mpg ~ wt + hp + disp, data = mtcars)
coef(ols2)
```

```
## (Intercept)          wt          hp        disp
## 37.1055052690 -3.8008905826 -0.0311565508 -0.0009370091
```

We can calculate the fitted values $\hat{Y}_i$ and residuals $\hat{e}_i$ add them to the data frame.

```
# Create a new dataframe with fitted values and residuals
mtcars_new <- mtcars %>%
  mutate(
    fitted_value = fitted(ols2),
    Residuals = residuals(ols2)
  )

# Display selected columns
head(mtcars_new %>% select(mpg, wt, hp, disp, fitted_value,
    Residuals))
```

```
##                    mpg    wt  hp disp fitted_value  Residuals
## Mazda RX4         21.0 2.620 110  160     23.57003 -2.5700299
## Mazda RX4 Wag     21.0 2.875 110  160     22.60080 -1.6008028
## Datsun 710        22.8 2.320  93  108     25.28868 -2.4886829
## Hornet 4 Drive    21.4 3.215 110  258     21.21667  0.1833269
## Hornet Sportabout 18.7 3.440 175  360     18.24072  0.4592780
## Valiant           18.1 3.460 105  225     20.47216 -2.3721590
```

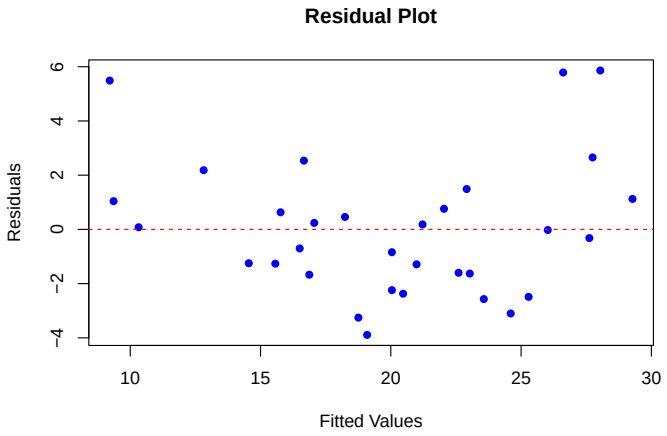
We can check whether Preperty 2.1 holds by calculating the inner product of the residuals with the regressors.

```
# Check orthogonality of residuals with wt
sum( mtcars$wt * mtcars$Residuals )

sum( mtcars$hp * mtcars$Residuals )

sum( mtcars$disp * mtcars$Residuals )
```

```
## [1] 0
## [1] 0
## [1] 0
```

A direct consequence of Property 2.1 is that the iner product of fitted values and residuals is also zero. Let's plot the residuals against the fitted values.

```
# Residual plot
plot(mtcars_new$fitted_value, mtcars_new$Residuals,
        main = "Residual Plot",
        xlab = "Fitted Values",
        ylab = "Residuals",
        col = "blue", pch = 16)
abline(h = 0, col = "red", lty = 2)  # Add reference line at zero
```



Finaaly, we can use the `predict` function to predict the `mpg` for new data. Suppose that we have a car with weight 3, horsepower 100, and displacement 200. What is the predicted Miles/(US) gallon for this car using our model?

```
# Predict mpg for new data
new_data <- data.frame(wt = 3, hp = 100, disp = 200)
predict(ols2, newdata = new_data)
```

```
##        1
## 22.39978
```

Our model predicts that the car will have 22.4 Miles/(US) gallon.

4 OLS (Matrix Ver.)

4.1 Matrix Notation

We now present the OLS estimator in matrix notation. Matrix is a powerful tool in presenting data, accelerating computation, and simplify notations. Though it might be though to get used to at first, it is a must-have skill for amyone who aims to have a career in data analysis.

Define the following matrices:

$$\underset{n\times 1}{\mathbf{Y}} = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}, \quad \underset{n\times k}{\mathbf{X}} = \begin{pmatrix} X_1' \\ \vdots \\ X_n' \end{pmatrix}, \quad \underset{n\times 1}{\beta} = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}.$$

where X_i' is the *transpose* of the i-th observation X_i. Notice that $\mathbf{X}$ is a $n \times k$ matrix,

$$\mathbf{X} = \begin{pmatrix} X_1^1 & X_1^2 & \cdots & X_1^k \\ \vdots & \vdots & \ddots & \vdots \\ X_n^1 & X_n^2 & \cdots & X_n^k, \end{pmatrix}$$

which is exactly how data are stored in Excel or R.

Sample sums can be written in matrix notation. For example

$$\hat{\mathbf{Q}}_{XX} = \frac{1}{n}\sum_{i=1}^{n} X_i X_i' = \frac{1}{n}\mathbf{X}'\mathbf{X}$$

$$\hat{\mathbf{Q}}_{XY} = \frac{1}{n}\sum_{i=1}^{n} X_i Y_i = \frac{1}{n}\mathbf{X}'\mathbf{Y}.$$

Therefore the OLS estimator can be written as

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}. \tag{16}$$

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am
1	21.0	6	160.0	110	3.90	2.620	16.46	0	1
2	21.0	6	160.0	110	3.90	2.875	17.02	0	1
3	22.8	4	108.0	93	3.85	2.320	18.61	1	1
4	21.4	6	258.0	110	3.08	3.215	19.44	1	0
5	18.7	8	360.0	175	3.15	3.440	17.02	0	0
6	18.1	6	225.0	105	2.76	3.460	20.22	1	0
7	14.3	8	360.0	245	3.21	3.570	15.84	0	0
8	24.4	4	146.7	62	3.69	3.190	20.00	1	0
9	22.8	4	140.8	95	3.92	3.150	22.90	1	0
10	19.2	6	167.6	123	3.92	3.440	18.30	1	0
11	17.8	6	167.6	123	3.92	3.440	18.90	1	0

Figure 3: Matrix $\mathbf{X}$ is exactly how data is stored in R.

Important
Don't confuse the matrix $\mathbf{X}$ with the vector X. The matrix $\mathbf{X}$ contains all the data, while the vector X is a single observation. In vector notation the OLS estimator is

$$\hat{\beta} = \underbrace{\left(\sum_{i=1}^{n} X_i X_i'\right)^{-1}}_{(k\times k)} \underbrace{\left(\sum_{i=1}^{n} X_i Y_i\right)}_{k\times 1}.$$

The expression is different from equation 16!

4.2 OLS as Projection

A great advantage of matrix notation is that it allows us to see the OLS process as a projection. However, this requires some knowledge on linear transformation and projection.

Definition 4.1 (Column Space)**.** The column space of a matrix $\mathbf{X}$ is the set of all possible linear combinations of the columns of $\mathbf{X}$.

$$\text{span}(\mathbf{X}) = \left\{\mathbf{X}v \text{ such that } v \in \mathbb{R}^k\right\}.$$

Definition 4.2 (Rank)**.** The rank of a matrix $\mathbf{X}$ is the dimension of the column space of $\mathbf{X}$.

$$\text{rank}(\mathbf{X}) = \dim(\text{span}(\mathbf{X})).$$

The dimension of a vector space is the number of **linearly independent vectors** that span the space.

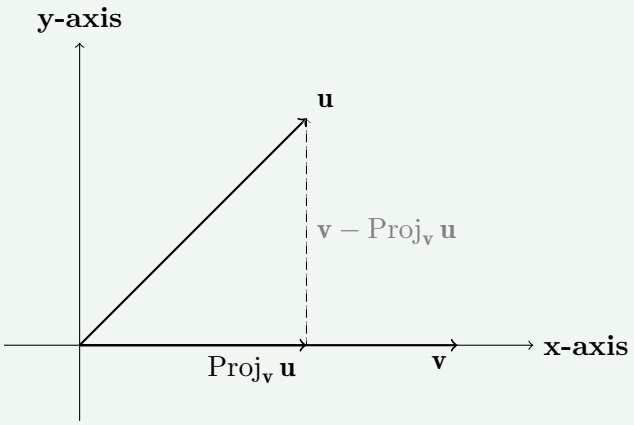
A linear transformation T is a linear function that maps a vector space V to another vector space W. Projection is a special type of linear transformation. It projects a vector u onto a subspace V.

Let's first review the projection of a vector onto a line from our high school math class.

Remark 4. Recall from your high school math class that the projection of a vector u onto a vector v is given by

$$\text{Proj}_v\, u = v\frac{v'u}{v'v}.$$

where $v'u$ is the *inner product* of v and u.



Once you made this connection you will realize that the OLS estimator $\hat{\beta}$ is the projection of Y onto the column space of X.

$$\mathbf{X}'\hat{\beta} = \mathbf{X}'(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{Y}$$

where $(\mathbf{X}'\mathbf{X})^{-1}$ plays the role of $\frac{1}{v'v}$ in the projection formula.

Definition 4.3. The projection matrix $\mathbf{P}$ is defined as

$$\mathbf{P} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'.$$

The fitted value $\hat{\mathbf{Y}}$ is the projection of Y onto the column space of X.

$$\hat{\mathbf{Y}} = \mathbf{X}\hat{\beta} = \mathbf{P}\mathbf{Y}.$$

Here we list some properties of the projection matrix $\mathbf{P}$.

Property 4.1. The projection matrix $\mathbf{P}$ has the following properties:

1. $\mathbf{P}$ is idempotent. $\mathbf{PP} = \mathbf{P}$.

2. $\mathbf{P}$ is symmetric. $\mathbf{P}' = \mathbf{P}$.

3. $\text{rank}(\mathbf{P}) = \text{rank}(\mathbf{X})$.

Property 4.1.1 has a very intuitive interpretation. The projection matrix $\mathbf{P}$ projects a vector onto a subspace. If you project the vector again, it will not change.

The geometric interpretation of Property 4.1.2 is more invloved. Property 4.1.3 basically says that the rank of the projection matrix is the same as the rank of the matrix $\mathbf{X}$.

Now we turn to the propert of the residuals.

Definition 4.4 (Annihilator Matrix)**.** The annihilator matrix $\mathbf{M}$ is defined as

$$\mathbf{M} = \mathbf{I} - \mathbf{P}.$$

By definition,

$$\mathbf{Y} = \mathbf{PY} + \mathbf{MY},$$

where $\mathbf{PY}$ is the fitted value and $\hat{\mathbf{e}} := \mathbf{MY}$ is the residual.

Property 4.2.

$$\mathbf{MX} = (\mathbf{I} - \mathbf{P})\mathbf{X} = \mathbf{X} - \mathbf{PX} = 0.$$

Definition 4.5 (Orthogonality)**.** Two vectors u and v are orthogonal if their inner product is zero.

$$u \perp v \iff u'v = 0.$$

Thus $\mathbf{M}$ and $\mathbf{X}$ are orthogonal.

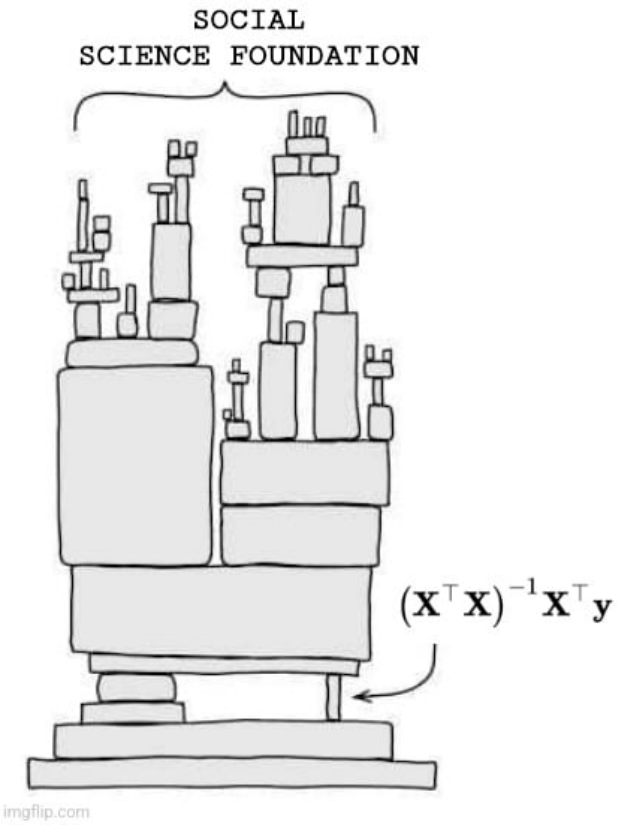


Figure 4: The reason of learning OLS

To be continued...

References

Hansen, Bruce E., *Econometrics*, Princeton: Princeton University Press, 2022.

Acronyms

BLP	best linear predictor. 2–5
FOC	first-order condition. 4
OLS	ordinary least squares. 3–6, 9–11
SSE	sum of squared errors. 4