Simple Linear Regression

Given dependent variable y and independent variable x, we assume the observed data follows the function:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

The mathematical problem to solve is:

$$min_{\hat{\beta}_0, \hat{\beta}_1} SSE = \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)]^2 = \sum_{i=1}^{n} e_i^2$$

In other words, we need to find estimators $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize the sum of squared errors (SSE). This is why the regression method is often also called **Ordinary Least Squared (OLS)**. Do note that in the above equations, $\epsilon \neq e$: the former refers to random, "natural" disturbances in the data (e.g. white noises) while the later refers to residuals or errors in the model.

To minimize a convex function, which the SSE above is, we must take the first-order conditions. In other words, we differentiate it with respect to $\hat{\beta}_0$ and $\hat{\beta}_1$, set both those derivatives to 0, and solve the system of two equations.

Taking the partial derivative with respect to $\hat{\beta}_0$:

$$\frac{\partial}{\partial \hat{\beta}_0} \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)]^2 = \sum_{i=1}^{n} \frac{\partial}{\partial \hat{\beta}_0} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)]^2 = (-2) \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)] = 0$$

Taking the partial derivative with respect to $\hat{\beta}_1$:

$$\frac{\partial}{\partial \hat{\beta}_1} \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)]^2 = \sum_{i=1}^{n} \frac{\partial}{\partial \hat{\beta}_1} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)]^2 = (-2) \sum_{i=1}^{n} x_i [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)] = 0$$

Therefore, the system of equations to be solved is:

$$\sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)] = 0$$
$$\sum_{i=1}^{n} x_i [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)] = 0$$

Solving the first equation for $\hat{\beta}_0$:

$$\sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)] = 0$$
$$\sum_{i=1}^{n} y_i - \sum_{i=1}^{n} \hat{\beta}_0 - \sum_{i=1}^{n} \hat{\beta}_1 x_i = 0$$
$$n\hat{\beta}_0 = \sum_{i=1}^{n} y_i - \hat{\beta}_1 \sum_{i=1}^{n} x_i$$
$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

Solving the first equation for $\hat{\beta}_1$ by plugging in $\hat{\beta}_0$:

$$\sum_{i=1}^{n} x_i [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)] = 0$$
$$\sum_{i=1}^{n} x_i [y_i - (\bar{y} - \hat{\beta}_1 \bar{x} + \hat{\beta}_1 x_i)] = 0$$
$$\sum_{i=1}^{n} x_i [y_i - \bar{y} - \hat{\beta}_1 (x_i - \bar{x})] = 0$$
$$\sum_{i=1}^{n} x_i (y_i - \bar{y}) = \sum_{i=1}^{n} x_i \hat{\beta}_1 (x_i - \bar{x})$$
$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} x_i (y_i - \bar{y})}{\sum_{i=1}^{n} x_i (x_i - \bar{x})}$$

Note that:

$$\sum_{i=1}^{n} (y_i - \bar{y}) = \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} \bar{y} = \sum_{i=1}^{n} y_i - n\bar{y} = 0$$
$$\sum_{i=1}^{n} (x_i - \bar{x}) = \sum_{i=1}^{n} x_i - \sum_{i=1}^{n} \bar{x} = \sum_{i=1}^{n} x_i - n\bar{x} = 0$$

as $\bar{x}$ and $\bar{y}$ - being mean values of x and y - are constant. This also means x_i values cannot be identical since $\hat{\beta}_1$ will be invalid in that case - as the denominator in the formula above will be 0.

Using these equalities, we can rewrite the numerator and denominator of $\hat{\beta}_1$:

$$\sum_{i=1}^{n} x_i (y_i - \bar{y}) = \sum_{i=1}^{n} x_i (y_i - \bar{y}) - 0 = \sum_{i=1}^{n} x_i (y_i - \bar{y}) - \sum_{i=1}^{n} \bar{x} (y_i - \bar{y}) = \sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})$$
$$\sum_{i=1}^{n} x_i (x_i - \bar{x}) = \sum_{i=1}^{n} x_i (x_i - \bar{x}) - 0 = \sum_{i=1}^{n} x_i (x_i - \bar{x}) - \sum_{i=1}^{n} \bar{x} (x_i - \bar{x}) = \sum_{i=1}^{n} (x_i - \bar{x})^2$$
$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

This is the form that many books on Statistics and Machine Learning use, including the textbook. If you remember your statistics coursework, you can notice that the numerator and the denominator look similar to the covariance of x between y and the variance of x respectively:

$$cov(x, y) = E[(X - E[X])(Y - E[Y])] = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{n} = \frac{\sum_{i=1}^{n} x_i y_i}{n} - \frac{\sum_{i=1}^{n} x_i}{n} \frac{\sum_{i=1}^{n} y_i}{n} = E[XY] - E[X]E[Y]$$

$$\sigma_x^2 = E[(X - E[X])^2] = \frac{\sum_{i=1}^{n} (x_i - \bar{x})^2}{n} = \frac{\sum_{i=1}^{n} x_i^2}{n} - (\frac{\sum_{i=1}^{n} x_i}{n})^2 = E[X^2] - (E[X])^2$$

As such, we can further rewrite $\hat{\beta}_1$ as:

$$\hat{\beta}_1 = \frac{cov(x, y)}{\sigma_x^2} = \frac{n^2 cov(x, y)}{n^2 \sigma_x^2} = \frac{n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2}$$

Exercise 1: Write & Test a Simple Linear Regression function

Import numpy, pandas, and sklearn:

In [ ]:
```
import numpy as np
import pandas as pd
import sklearn
```

Load the California housing data set using sklearn.datasets:

In [ ]:
```
from sklearn.datasets import fetch_california_housing
data = fetch_california_housing(as_frame = True).frame
data.head()
```

Using sklearn's built-in LinearRegression function, regress **MedHouseVal** on **MedInc**:

In [ ]:
```
from sklearn.linear_model import LinearRegression
X = data['MedInc'].values.reshape(-1, 1)
y = data['MedHouseVal'].values
reg = LinearRegression().fit(X, y)
```

Check the model's coefficient and intercept:

In [ ]:
```
print("Intercept: %f" % (reg.intercept_))
print("Slope: %f" % (reg.coef_))
```

Based on the derivations of $\hat{\beta}_0$ and $\hat{\beta}_1$ above, fill in the blank to complete the function below. You can use pandas and numpy's mean function to calculate the means of the variables:

In [ ]:
```
def simple_linear_regression (x, y):
    x_bar = ...
    y_bar = ...
    slope = ...
    intercept = ...
    return (intercept, slope)
```

Check if the output intercept and the slope match with the ones above:

In [ ]:
```
X = data['MedInc'].values
y = data['MedHouseVal'].values
simple_linear_regression (X, y)
```

Multivariate Linear Regression

With multiple independent variables $x_1, x_2, \dots, x_k$ for $k \geq 2$, our regression model will become:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_k x_{ik} + \epsilon_i$$

The mathematical problem to solve become:

$$min_{\beta_0, \dots, \beta_k} SSE = \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \sum_{j=1}^{k} \hat{\beta}_j x_{ij})]^2$$

We can repeat the same approach above. However, we now must take $k+1$ first-order conditions instead of only two as before. This can potentially create a very complex system of equations, which will mostly be very difficult to untangle and computationally intensive to solve by hand. Thankfully, we have computers to do that for us - at least, most of the time.

As an example, we can quickly examine the simplest case where $k = 2$. In that case, the SSE is:

$$SSE = \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})]^2 = \sum_{i=1}^{n} e_i^2$$

Similar to above, take the partial derivative with respect to $\hat{\beta}_0$, $\hat{\beta}_1$, and $\hat{\beta}_2$:

$$\frac{\partial}{\partial \hat{\beta}_0} \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})]^2 = \sum_{i=1}^{n} \frac{\partial}{\partial \hat{\beta}_0} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})]^2 = (-2) \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})] = 0$$

$$\frac{\partial}{\partial \hat{\beta}_1} \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})]^2 = \sum_{i=1}^{n} \frac{\partial}{\partial \hat{\beta}_1} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})]^2 = (-2) \sum_{i=1}^{n} x_{i1} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})] = 0$$

$$\frac{\partial}{\partial \hat{\beta}_2} \sum_{i=1}^{n} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})]^2 = \sum_{i=1}^{n} \frac{\partial}{\partial \hat{\beta}_2} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})]^2 = (-2) \sum_{i=1}^{n} x_{i2} [y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2})] = 0$$

It is easy to notice that similar to above, we can trivially derive $\hat{\beta}_0$:

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}_1 - \hat{\beta}_2 \bar{x}_2$$

Like before, to proceed, we can substitute this term into the second and third inequalities. However, since there are still two unknowns remaining, we must solve either $\hat{\beta}_1$ or $\hat{\beta}_2$ in terms of the other and plug the solution into the remaining variable's inequality. Ultimately, $\hat{\beta}_1$ and $\hat{\beta}_2$ are:

$$\hat{\beta}_1 = \frac{(\sum_{i=1}^{n} x_{i2}^2)(\sum_{i=1}^{n} x_{i1} y_i) - (\sum_{i=1}^{n} x_{i1} x_{i2})(\sum_{i=1}^{n} x_{i2} y_i)}{(\sum_{i=1}^{n} x_{i1}^2)(\sum_{i=1}^{n} x_{i2}^2) - (\sum_{i=1}^{n} x_{i1} x_{i2})^2}$$

$$\hat{\beta}_2 = \frac{(\sum_{i=1}^{n} x_{i1}^2)(\sum_{i=1}^{n} x_{i2} y_i) - (\sum_{i=1}^{n} x_{i1} x_{i2})(\sum_{i=1}^{n} x_{i1} y_i)}{(\sum_{i=1}^{n} x_{i1}^2)(\sum_{i=1}^{n} x_{i2}^2) - (\sum_{i=1}^{n} x_{i1} x_{i2})^2}$$

A simpler, somewhat brute-force solution would be to directly solve the derived system of equations above with the given data:

$$\sum_{i=1}^{n} y_i = \sum_{i=1}^{n} (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2}) = n\hat{\beta}_0 + \hat{\beta}_1 \sum_{i=1}^{n} x_{i1} + \hat{\beta}_2 \sum_{i=1}^{n} x_{i2}$$

$$\sum_{i=1}^{n} y_i x_{i1} = \sum_{i=1}^{n} x_{i1} (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2}) = \hat{\beta}_0 \sum_{i=1}^{n} x_{i1} + \hat{\beta}_1 \sum_{i=1}^{n} x_{i1}^2 + \hat{\beta}_2 \sum_{i=1}^{n} x_{i1} x_{i2}$$

$$\sum_{i=1}^{n} y_i x_{i2} = \sum_{i=1}^{n} x_{i2} (\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \hat{\beta}_2 x_{i2}) = \hat{\beta}_0 \sum_{i=1}^{n} x_{i2} + \hat{\beta}_1 \sum_{i=1}^{n} x_{i1} x_{i2} + \hat{\beta}_2 \sum_{i=1}^{n} x_{i2}^2$$

by calculating $\sum_{i=1}^{n} x_{i1}^2$, $\sum_{i=1}^{n} x_{i2}^2$, $\sum_{i=1}^{n} x_{i1} x_{i2}$, $\sum_{i=1}^{n} x_{i1} y$, and $\sum_{i=1}^{n} x_{i2} y$ directly from the data.

Exercise 2: Creating new columns from existing ones by using pandas

Using sklearn's built-in LinearRegression function, regress **MedHouseVal** on **MedInc** and **HouseAge**:

In [ ]:
```
X = data[['MedInc', 'HouseAge']].values
y = data['MedHouseVal'].values
reg = LinearRegression().fit(X, y)
```

In [ ]:
```
print("Intercept: %f" % (reg.intercept_))
print("Coefficient for MedInc: %f" % (reg.coef_[0]))
print("Coefficient for HouseAge: %f" % (reg.coef_[-1]))
```

Add columns named **MedInc^2**, **HouseAge^2**, **MedInc*HouseAge**, **MedInc*MedHouseVal**, and **HouseAge*MedHouseVal** - containing values as indicated by their names using pandas:

In [ ]:
```
data['MedInc^2'] = ...
data['HouseAge^2'] = ...
data['MedInc*HouseAge'] = ...
data['MedInc*MedHouseVal'] = ...
data['HouseAge*MedHouseVal'] = ...
```

Sum up the columns above along with **MedInc**, **HouseAge**, and **MedHouseVal**. Use these sums to check the coefficients and intercept by plugging in values in the system above as appropriate (e.g. see eMathHelp's System of Linear Equations Calculator). See if your coefficients match sklearn's results above. The number of observations or n is 20640.

In [ ]:
```
print("Sum of MedInc: %f" % (data['MedInc'].sum()))
print("Sum of HouseAge: %f" % (data['HouseAge'].sum()))
print("Sum of MedHouseVal: %f" % (data['MedHouseVal'].sum()))
print("Sum of MedInc^2: %f" % (data['MedInc^2'].sum()))
print("Sum of HouseAge^2: %f" % (data['HouseAge^2'].sum()))
print("Sum of MedInc*HouseAge: %f" % (data['MedInc*HouseAge'].sum()))
print("Sum of MedInc*MedHouseVal: %f" % (data['MedInc*MedHouseVal'].sum()))
print("Sum of HouseAge*MedHouseVal: %f" % (data['HouseAge*MedHouseVal'].sum()))
```

Closed Form Solution of Linear Regression (a.k.a The Matrix Form)

You can easily notice that the system of equations in linear regression:

$$y_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \dots + \beta_k x_{1k} + \epsilon_1$$
$$y_2 = \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \dots + \beta_k x_{2k} + \epsilon_2$$
$$\vdots$$
$$y_k = \beta_0 + \beta_1 x_{k1} + \beta_2 x_{k2} + \dots + \beta_k x_{kk} + \epsilon_k$$

can be written as:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & x_{k1} & x_{k2} & \cdots & x_{kk} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_k \end{bmatrix}$$

$$Y = X\beta + \epsilon$$

with Y and ϵ being n-by-1 vectors, β being a k-by-1 vector, and X being an n-by-k matrix.

The n-by-1 vector of residuals, e, is therefore:

$$e = Y - X\hat{\beta} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix} - \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & x_{k1} & x_{k2} & \cdots & x_{kk} \end{bmatrix} \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \\ \hat{\beta}_2 \\ \vdots \\ \hat{\beta}_k \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_k \end{bmatrix}$$

where $\hat{\beta}$ is a k-by-1 vector of estimators.

The Closed Form solution of Linear Regression is written as:

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

a form which many advanced Machine Learning books and courses use.

The proof for this equation is quite mathematically rigorous as it involves differentiation with matrice and vectors. This blog post explains the mathematics behind it very well - much better than I can. An alternative explanation is based on geometric intuition: consider the plane containing all $X\hat{\beta}$ vectors for all possible $\hat{\beta}$ vectors. We want to minimize the distance from this plane to Y, which is equal to the length of the residual vector e and the sum of squared errors as:

$$e = Y - X\hat{\beta}$$
$$|e| = \sqrt{\sum_{i=1}^{n} e_i^2}$$

To do so, we must find $\hat{\beta}$ such that e is perpendicular to the $X\hat{\beta}$-plane. For this reason, this form is usually also called the **Normal Equation**.

To end the notes, I will prove that the Closed Form solution will have the same results as the solution(s) above - specifically with Simple Linear Regression, which is the easiest to observe.

In Simple Linear Regression, the matrix X and $X^\intercal$ are:

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_k \end{bmatrix} \longleftrightarrow X^\intercal = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_k \end{bmatrix}$$

As such, $X^\intercal X$ is:

$$X^\intercal X = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_k \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_k \end{bmatrix} = \begin{bmatrix} n & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 \end{bmatrix}$$

Its inverse is therefore:

$$(X^\intercal X)^{-1} = \frac{1}{n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2} \begin{bmatrix} \sum_{i=1}^{n} x_i^2 & -\sum_{i=1}^{n} x_i \\ -\sum_{i=1}^{n} x_i & n \end{bmatrix}$$

Lastly, $X^\intercal Y$ is:

$$X^\intercal Y = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_k \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \end{bmatrix}$$

From there, we can expand the Closed Form:

$$\hat{\beta} = (X^\intercal X)^{-1} X^\intercal Y = \frac{1}{n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2} \begin{bmatrix} \sum_{i=1}^{n} x_i^2 & -\sum_{i=1}^{n} x_i \\ -\sum_{i=1}^{n} x_i & n \end{bmatrix} \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \end{bmatrix}$$

$$\hat{\beta} = \frac{1}{n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2} \begin{bmatrix} \sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_i y_i \\ -\sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i + n \sum_{i=1}^{n} x_i y_i \end{bmatrix}$$

Notice that the value on the second row is:

$$\frac{n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2}$$

which, as above, is equal to:

$$\hat{\beta}_1 = \frac{cov(x, y)}{\sigma_x^2} = \frac{n^2 cov(x, y)}{n^2 \sigma_x^2} = \frac{n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2}$$

Once again, consider the value of $\hat{\beta}_0$ in Simple Linear Regression:

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

Plugging in $\hat{\beta}_1$ as above:

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

$$\hat{\beta}_0 = \frac{\sum_{i=1}^{n} y_i}{n} - \frac{n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2} \frac{\sum_{i=1}^{n} x_i}{n}$$

$$\hat{\beta}_0 = \frac{\sum_{i=1}^{n} y_i [n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2]}{n^2 \sum_{i=1}^{n} x_i^2 - n(\sum_{i=1}^{n} x_i)^2} - \frac{n \sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_i y_i - (\sum_{i=1}^{n} x_i)^2 \sum_{i=1}^{n} y_i}{n^2 \sum_{i=1}^{n} x_i^2 - n(\sum_{i=1}^{n} x_i)^2}$$

This is reduced to:

$$\hat{\beta}_0 = \frac{n[\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_i y_i]}{n[n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2]} = \frac{\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_i y_i}{n \sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2}$$

which is the same as the value on the first row of $(X^\intercal X)^{-1} X^\intercal Y$. As such:

$$\hat{\beta} = (X^\intercal X)^{-1} X^\intercal Y = \begin{bmatrix} \hat{\beta}_0 \\ \hat{\beta}_1 \end{bmatrix}$$