Lecture 3

Hisam Sabouni, PhD

2/24/2020

Overview

The goal of this lecture is to intruce the Ordinary Least Squares (OLS) estimator. We will first describe how the estimator works then derive its statistical properties. We will pay particular attention to how the estimator is estimated and the underlying assumptions that need to hold in order for the interpretation of the estimator to be unbiased and consistent.

Ordinary Least Squares (OLS)

As the name implies the objective function of OLS is the squared error between a set of predicted values and the actual data. In particular the goal of OLS is to model the relationship between a **dependent variable Y** and a set of **independent, or explanatory, variables** $X_1, X_2, \dots$ by assigning weights, or coefficients, to each indpendent variables. Essentially we are estimating the following relationship:

$$Y_i = f(X_{i,1}, X_{i,2}, \dots)$$

Here each observation i has a dependent variable Y_i and a set of associated depdent varibles $X_{i,j}$. Typically in econometrics we assume that the the function that maps each of our indpendent variables to our dependent variable takes on a linear form:

$$Y_i = \beta_0 + \beta_1 X_{i,1} + \beta_2 X_{i,2} + \dots$$

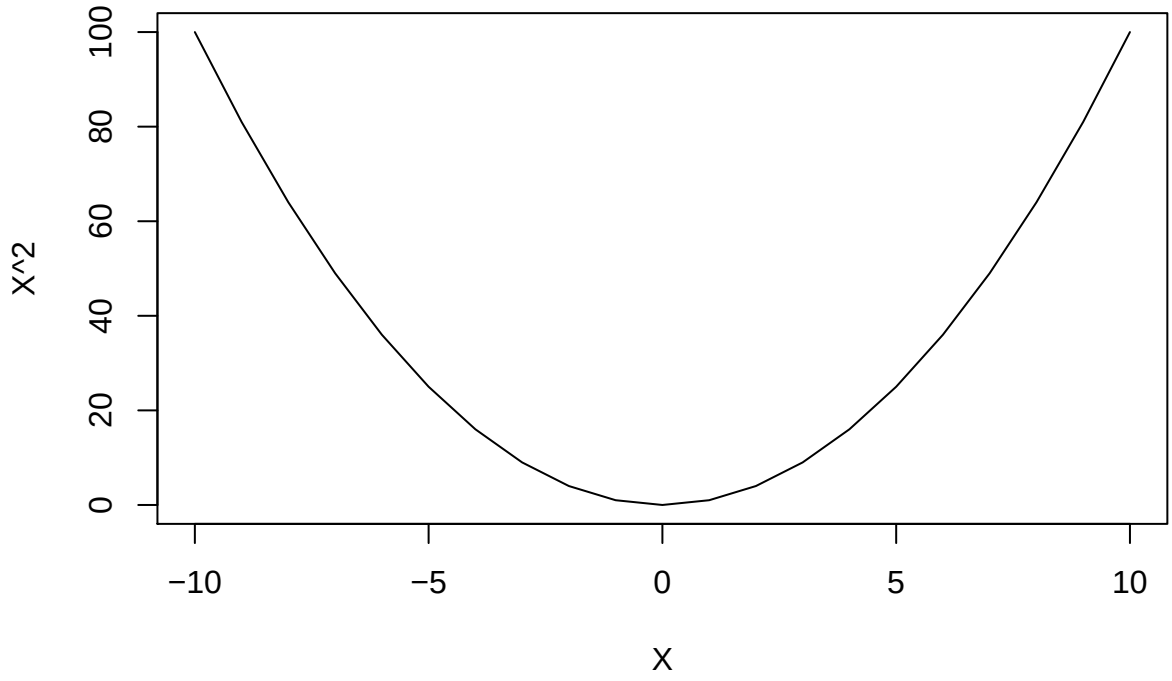
Assuming linear relationships allows for clear interpretation of our model. Furthermore, most real world relationships can in fact be modeled in a linear framework. In the above model assigns weights (the β terms) to each independent variable $X_{i,j}\ \forall\ j$. The first weight (β_0) is a special term,

which is called the intercept of the model. Now in a perfect world we would be able to assign the weights to each independent variable such that there is no error. That is a perfect model that takes in a set of indepdnet variables and perfectly predicts the dependent variable. In most economic settings this isn't the case. As a result, we assume that our model usually contains some error:

$$Y_i = \beta_0 + \beta_1 X_{i,1} + \beta_2 X_{i,2} + \cdots + \epsilon_i$$

Here whatever is not captured by our model we toss into the error term (denoted by ϵ, could be any letter, ϵ isn't special..). The goal of Ordinary Least Squares is the assign the weights β to each dependent variable $X_{i,j}$ to minimize the squared error ϵ. Why squared error? Why not absolute error? Well, when we square a variable we are actually mapping the data into a parabola which has a well defined minimum:

```
plot(seq(-10, 10, by = 1), seq(-10, 10, by = 1)^2,
    typ = "l", xlab = "X", ylab = "X^2")
```



As you can see there is a nice well defined minimum when we use a parabola. This makes it easy for us to think about how we may want to derive our optimal set of weights to minimize the error

(take the derivative and set it equal to zero!). Note that for a given set of weights and a given set of data our measure of the error will vary. Going forward we will denote the true, or, population relationship between our dependent and independent variables as follows:

$$Y_i = \beta_0 + \beta_1 X_{i,1} + \beta_2 X_{i,2} + \cdots + \epsilon_i$$

For a given set of **estimate** we will place a $\hat{\cdot}$ over the weight:

$$Y_i = \hat{\beta}_0 + \hat{\beta}_1 X_{i,1} + \hat{\beta}_2 X_{i,2} + \cdots + \hat{\epsilon}_i$$

One Inpendent Variable

Lets think about how we would go about minimizing the squared error in a linear model where we have only on independent variable X. Let our population model be:

$$(1) \quad Y_i = \beta_0 + \beta_1 X_{i,1} + \epsilon_i$$

Our goal is to estimate $\{\beta_0, \beta_1\}$ to minimize the residual sum of squared error $\sum_i \epsilon_i^2$. We can state this as a minimization problem as follows:

$$(2) \quad min_{\{\hat{\beta}_0, \hat{\beta}_1\}} \sum_i \hat{\epsilon}_i^2 = \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i,1})^2$$

Here we took equation (1) and isolated the error term for each observation i that we have in our sample (subtract over $\beta_0 + \beta_1 X_{i,1}$). Each error term for each observation i is then squared. To get the residual squared error we simply sum across the squared error of each observation, which gets us to equation (2). In the minimization problem presented in equation (2) we need to find $\{\hat{\beta}_0, \hat{\beta}_1\}$ to minimize the residual sum of squared error. Given that the squared error has a well defined minimum, we can simply take partial derivatives with respect to each of our choice parameters ($\{\hat{\beta}_0, \hat{\beta}_1\}$) and set our derivatives equal to zero. Recall that when we are taking a derivative we are esentially estimating the slope of lines that are tangent with respect to our objective function. The only point along a parabola that has a slope of zero is the minimum point (smallest error in our case)..Lets do that:

$$(2) \quad min_{\{\hat{\beta}_0,\hat{\beta}_1\}} \sum_i \hat{\epsilon}_i^2 = \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i,1})^2$$

Let $\hat{E} = \sum_i \hat{\epsilon}_i^2$:

First we will take the partial derivative with respect to $\hat{\beta}_0$ (we will hold all other variables as constants). We will have to use the chain-rule here to take our derivative (derivative of the outside function multiplied by the derivative of the inside):

$$\frac{\partial \hat{E}}{\partial \hat{\beta}_0} : -2 \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i,1})$$

Here we brought the two down from the outside function and multiplied by the derivative of the inside (-1).

Setting equal to zero:

$$-2 \sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i,1}) = 0$$

$$\sum_i (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i,1}) = 0$$

Now, lets use a little trick. We know that $\bar{Y} = \frac{1}{N} \sum_i Y_i$ and that $\bar{X}_{i,1} = \frac{1}{N} \sum_i X_{i,1}$, therefore we can re-write this derivative as:

$$N\bar{Y}_i - N\hat{\beta}_0 - N\hat{\beta}_1 \bar{X}_{i,1} = 0$$

Which implies that:

$$\hat{\beta}_0 = \bar{Y}_i - \hat{\beta}_1 \bar{X}_{i,1}$$

Now lets go back to our objective function, E, and take the first order partial derivative with respect to our second choice parameter $\hat{\beta}_1$:

$$\frac{\partial \hat{E}}{\partial \hat{\beta}_1} : \sum_i -2X_{i,1}(Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i,1})$$

Here we brought the two down from the outside function and multiplied by the derivative of the inside ($-X_{i,1}$).

Setting equal to zero:

$$\sum_i -2X_{i,1}(Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i,1}) = 0$$

Lets get rid of the -2 that is being multiplyed by everything and distribute the X_{i,1}:

$$\sum_i X_{i,1}Y_i - \hat{\beta}_0 \sum_i X_{i,1} - \hat{\beta}_1 \sum_i X_{i,1}^2 = 0$$

Now we can plug in our first oder condition for $\hat{\beta}_0$ to isloate only $\hat{\beta}_1$:

$$\sum_i X_{i,1}Y_i - (\bar{Y}_i - \hat{\beta}_1 \bar{X}_{i,1}) \sum_i X_{i,1} - \hat{\beta}_1 \sum_i X_{i,1}^2 = 0$$

$$\sum_i X_{i,1}Y_i - \bar{Y}_i \sum_i X_{i,1} + \hat{\beta}_1 \bar{X}_{i,1} \sum_i X_{i,1} - \hat{\beta}_1 \sum_i X_{i,1}^2 = 0$$

Lets re-arrange:

$$\sum_i X_{i,1}Y_i - \bar{Y}_i \sum_i X_{i,1} = \hat{\beta}_1 \sum_i X_{i,1}^2 - \hat{\beta}_1 \bar{X}_{i,1} \sum_i X_{i,1}$$

Now we can use our average trick once again:

$$\sum_i X_{i,1}Y_i - N\bar{Y}_i\bar{X}_{i,1} = \hat{\beta}_1 \sum_i X_{i,1}^2 - N\hat{\beta}_1 \bar{X}_{i,1}\bar{X}_{i,1}$$

$$\sum_i X_{i,1}Y_i - N\bar{Y}_i\bar{X}_{i,1} = \hat{\beta}_1 (\sum_i X_{i,1}^2 - N\bar{X}_{i,1}^2)$$

Which implies that:

$$\hat{\beta}_1 = \frac{\sum_i X_{i,1} Y_i - N\bar{Y}_i \bar{X}_{i,1}}{\sum_i X_{i,1}^2 - N\bar{X}_{i,1}^2}$$

Now this formulation is pretty ugly. But we can actually show that this is equivalent to:

$$\hat{\beta_1 *} = \frac{\sum_i (X_{i,1} - \bar{X}_{i,1})(Y_i - \bar{Y}_i)}{\sum_i (X_{i,1} - \bar{X}_{i,1})^2}$$

Given this solution we have our solution for $\hat{\beta}_0$!

$$\hat{\beta *}_0 = \bar{Y}_i - \frac{\sum_i (X_{i,1} - \bar{X}_{i,1})(Y_i - \bar{Y}_i)}{\sum_i (X_{i,1} - \bar{X}_{i,1})^2} \bar{X}_{i,1}$$

Now given these two solutions we have solved for a general solution of weights to minimize the residual sum of squared error between our dependent variable and our independent variable! Lets create a small simulation to see how this works.

We will create some population data and define the underlying data generating process. We will then take a random sample from the population data and check our estimated parameters if they match our true underlying data generate process. We will simulate our one dependent variable $X_{i,1} \sim N(120, 5^2)$. We will model the data generate process as:

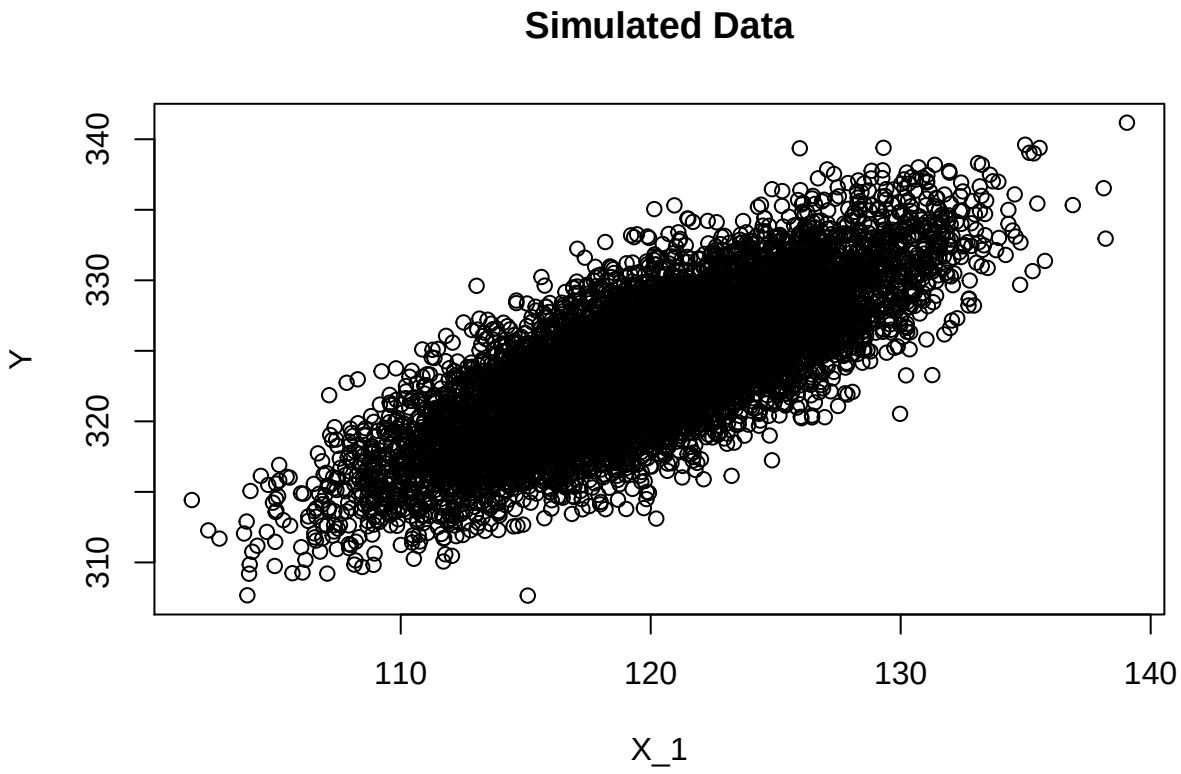
$$Y_i = 240 + 0.7 X_{i,1} + \epsilon, \text{ where } \epsilon \sim N(0, 3^2)$$

We will simulate 10,000 observations as our true data generating process. We will then take a sample of 1,000 observations and see if we can measure our true $\beta_0 = 240$ and our true $\beta_1 = 0.7$ using our derived formulas to minimize the residual sum of squared error.

```
# Set the seed#
set.seed(1)
# First lets make some X_{i,1} data:
X_1 <- rnorm(10000, 120, 5)  #10,000 obs from a random normal with mean 120 and standar
Y <- 240 + 0.7 * X_1 + rnorm(10000, 0, 3)  #Create a true Y, that has a true intercept
our_data <- cbind.data.frame(Y, X_1)
dim(our_data)  #check dimensions
```

[1] 10000 2

```
# Generate a plot
plot(our_data$X_1, our_data$Y, xlab = "X_1", ylab = "Y",
    main = "Simulated Data")
```




```
# Lets take a random sample of the full
# dataset of 1000 observations:
our_random_sample <- our_data[sample(1:nrow(our_data),
    1000), ]
dim(our_random_sample)  #check dimensions
```

[1] 1000 2

Now lets write a function that takes in one dependent variable and one independent variable. The function will use the equations we derived to give us our estimated weights to minimize the residual sum of squared error.

$$\hat{\beta}_1 = \frac{\sum_i (X_{i,1} - \bar{X}_{i,1})(Y_i - \bar{Y}_i)}{\sum_i (X_{i,1} - \bar{X}_{i,1})^2}$$

$$\hat{\beta}*_0 = \bar{Y}_i - \frac{\sum_i (X_{i,1} - \bar{X}_{i,1})(Y_i - \bar{Y}_i)}{\sum_i (X_{i,1} - \bar{X}_{i,1})^2} \bar{X}_{i,1}$$

```
baby_ols <- function(dependent, independent) {
    # indpendent = X_1 dependet = Y Given beta_1
    # is a bit complicated we will break it up
    # into two calculations one for the numerator:
    beta_1_numerator <- sum((independent - mean(independent,
        na.rm = T)) * (dependent - mean(dependent,
        na.rm = T)))
    # one for the denominator:
    beta_1_denominator <- sum((independent - mean(independent,
        na.rm = T))^2)
    # then put it together!
    beta_1 <- beta_1_numerator/beta_1_denominator
    # Given that we have beta_1 we can plug into
    # our derived equation and solve for beta_0
    # (our intercept)
    beta_0 <- mean(dependent, na.rm = T) - beta_1 *
        mean(independent, na.rm = T)
    # Store our results in a named vector
    results <- c(beta_0 = beta_0, beta_1 = beta_1)
    # return the results
    return(results)
}

# Now we can pass in our random sample of data
# and see the estimated results. The function
# should return for us an estimated intercept
# (beta_0) and an estimated coefficent for X_1
# (beta_1):
estimated_ols_weights <- baby_ols(dependent = our_random_sample$Y,
    independent = our_random_sample$X_1)
```

```
estimated_ols_weights
```

```
 beta_0      beta_1
```

237.3102200 0.7230976

Not bad! The function we created does indeed get pretty close to estimating the true population parameters. Recall that the population parameter $\beta_0 = 240$ our sample estimate using our derived formula is $\hat{\beta}_0 = 237.31022$. Similarly our true population parameter $\beta_1 = 0.7$ and our estimate is $\hat{\beta}_1 = 0.7230976$. Lets check a few things from here. A natural next question to ask is what is the residual sum of squared error given these predictions? Recall that the residual sum of squared errors is in essence the difference between the actual outcome variable (Y) our predicted values. Lets write another function that takes in an indpendent variable and a vector of weights, where the function will use the weights to generate predicted values:

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 X_{i,1}$$

Given the predicted values we can come up with our residual sum of squared errors:

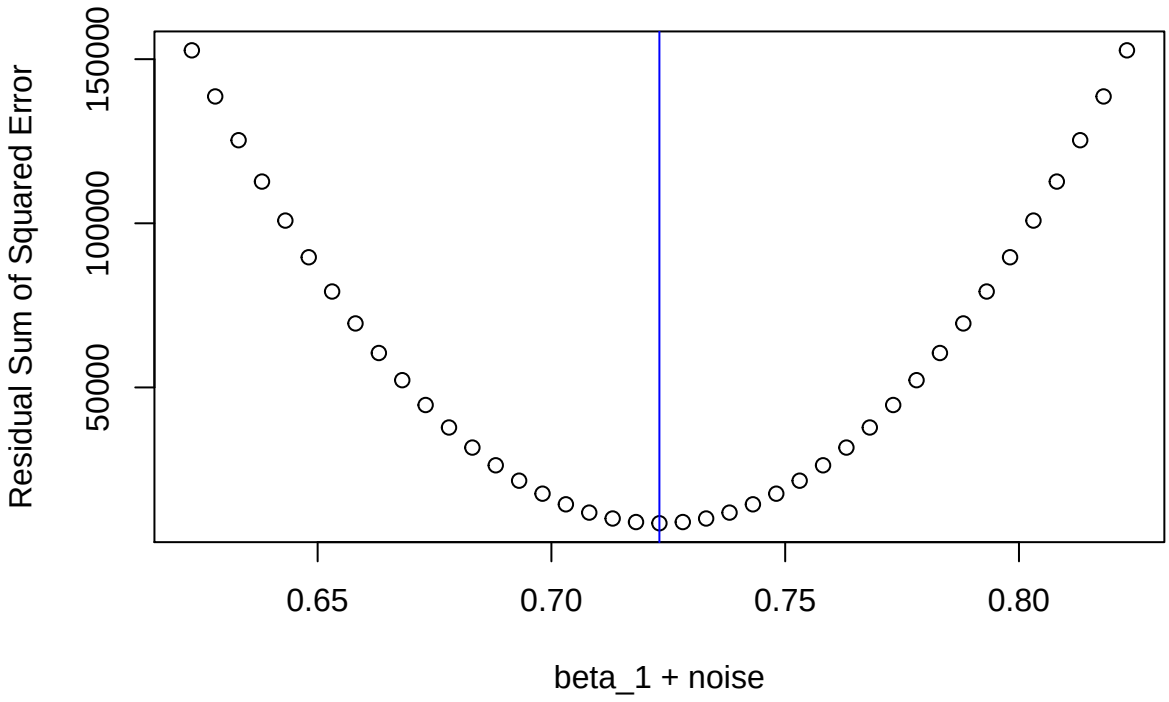
$$\sum_i \hat{\epsilon}^2 = \sum_i (Y_i - \hat{Y}_i)^2$$

```
baby_ols_predictions <- function(indpendent, weights) {
    # Function assumes that the first element of
    # weights is beta_0 (intercept) and the second
    # element of weights is beta_1 (weight on
    # indpendent)
    preds <- weights[1] + weights[2] * indpendent
    return(preds)
}
predicted_values <- baby_ols_predictions(indpendent = our_random_sample$X_1,
    weights = estimated_ols_weights)

residual_sum_squared_error <- sum((our_random_sample$Y -
    predicted_values)^2)
residual_sum_squared_error
```

[1] 8626.55

Great. Now we have the residual sum of squared errors associated with our estimated OLS weights. What happens to the residual sum of squared errors as we keep one weight constant and very one of the other weights?

```
# create vector that is a sequence from -0.1
# to 0.1 in steps of 0.01. This will be added
# to beta_1
noise_seq <- seq(-0.1, 0.1, by = 0.005)
# We're going to add the noise and see how the
# residual_sum_squared_error changes:
rsse_holder <- c()
for (i in 1:length(noise_seq)) {
    predicted_values_temp <- baby_ols_predictions(indpendent = our_random_sample$X_1,
        weights = c(estimated_ols_weights[1],
            estimated_ols_weights[2] + noise_seq[i]))

    rsse_holder[i] <- sum((our_random_sample$Y -
        predicted_values_temp)^2)
}
# :)
plot(estimated_ols_weights[2] + noise_seq, rsse_holder,
    xlab = "beta_1 + noise", ylab = "Residual Sum of Squared Error")
abline(v = estimated_ols_weights[2], col = 4)
```



Note that as we vary the underlying sample we run through our ordinary least squares estimator we will get a different estimate of our coefficients. Below we will run 1,000 simulations, each time drawing a random sample from our population data. We will record the estimated $\hat{\beta}_0$ and $\hat{\beta}_1$.

```
set.seed(123)
store_betas <- matrix(NA, 1000, 2)
for (i in 1:1000) {
    our_random_sample <- our_data[sample(1:nrow(our_data),
        1000), ]
    store_betas[i, ] <- baby_ols(dependent = our_random_sample$Y,
        independent = our_random_sample$X_1)
}
head(store_betas)
```

```
     [,1]      [,2]
```

[1,] 237.4452 0.7197276 [2,] 240.5039 0.6954642 [3,] 240.0772 0.6998583 [4,] 239.0042 0.7082725 [5,] 237.9237 0.7167581 [6,] 237.8870 0.7179903

```
mean(store_betas[, 1])
```

[1] 239.6989

```
sd(store_betas[, 1])
```

[1] 2.13737

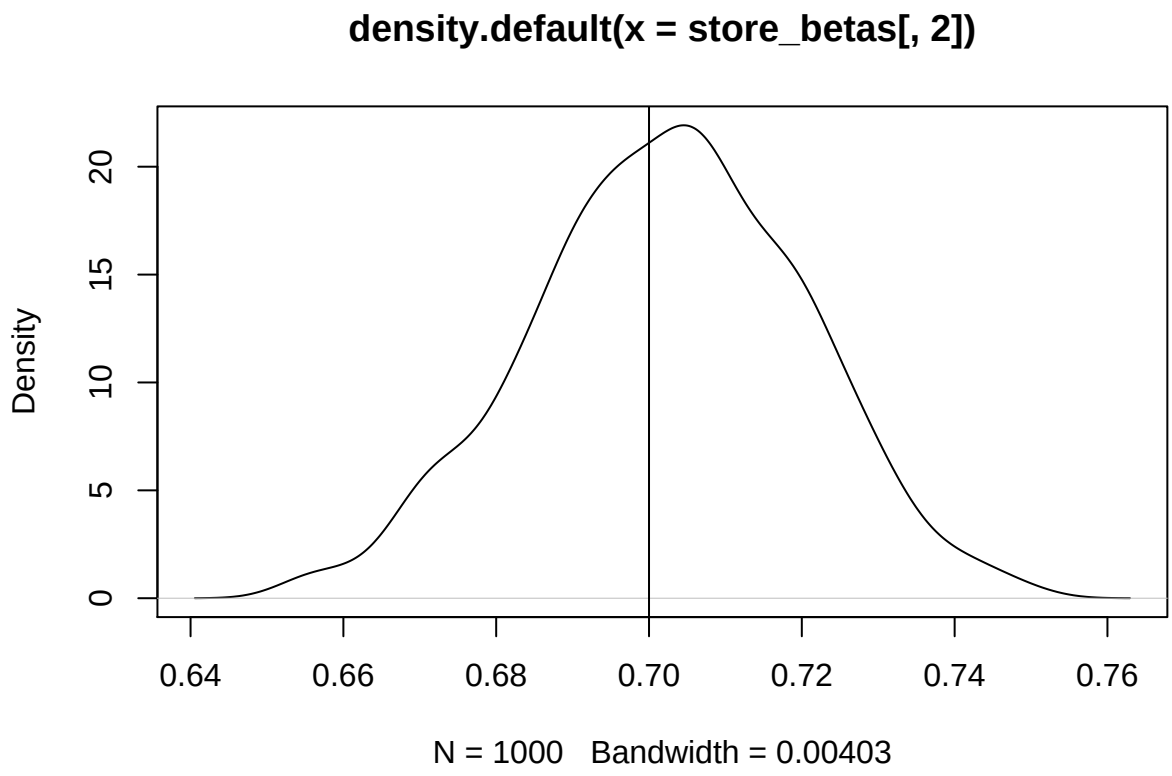
```
mean(store_betas[, 2])
```

[1] 0.7023855

```
sd(store_betas[, 2])
```

[1] 0.01782509

```
plot(density(store_betas[, 2]))
abline(v = 0.7)
```



Notice that as we vary the sample through our estimator we get a different set of coefficients. The properties of estimators we covered before are still valid for evaluating ordinary least squares. Recall

that the standard deviation of an estimator is called the standard error and recall that when an the expected value of an estimator is equal to the true population parameter of interest we state that the estimator is unbiased. Our simulation above actually allows us to measure the standard deviation of our estimator (standard deviation of the estimated coefficients) and allows us to estimate the expected value of our estimator (mean of the estimated coefficients).

Given our estimated coefficients and our estimated standard errors, we can actually construct t-statistics for our regression coefficients. For example, we can test if there is a non-zero relationship between $X_{i,1}$ and Y_i as follows:

$$H_0 : \beta_1 = 0$$

$$H_1 : \beta_1 \neq 0$$

First we would construct our t-statistic:

$$T = \frac{\hat{\beta}_1 - 0}{SE(\hat{\beta}_1)} = \frac{0.70}{0.018} \approx 38$$

Given that the estimated t-statistic in absolute value is well in excess of a traditional statistical significance level of say 1.96, a 5% chance of type-1 error (reject a null hypothesis that is in fact true), we can state that we 'reject our null hypothesis'. Now given that the sign of our t-statistic is positive we can also state that there is a positive 'statistically significant relationship' between $X_{i,1}$ and Y_i.

This is a pretty powerful statement. We know the true underlying relationship between $X_{i,1}$ and Y_i (we made it up) and through a small sample we are able to estimate the true underlying data generating process linking $X_{i,1}$ and Y_i. Through our estimated model:

$$\hat{Y}_i = 2\hat{4}0 + 0.\hat{7}0X_{i,1}$$

We are also making the statement that there is a constant linear relationship between Y_i and $X_{i,1}$. Notice that regardless of the level of $X_{i,1}$, if we increase $X_{i,1}$ by 1 we will increase our estimated prediction of Y_i by $\hat{\beta}_1 = 0.7$. Similarly, if we decrease $X_{i,1}$ by 1 we will decrease our estimated prediction of Y_i by $\hat{\beta}_1 = 0.7$. The first order conditions of the minimization problem of the residual

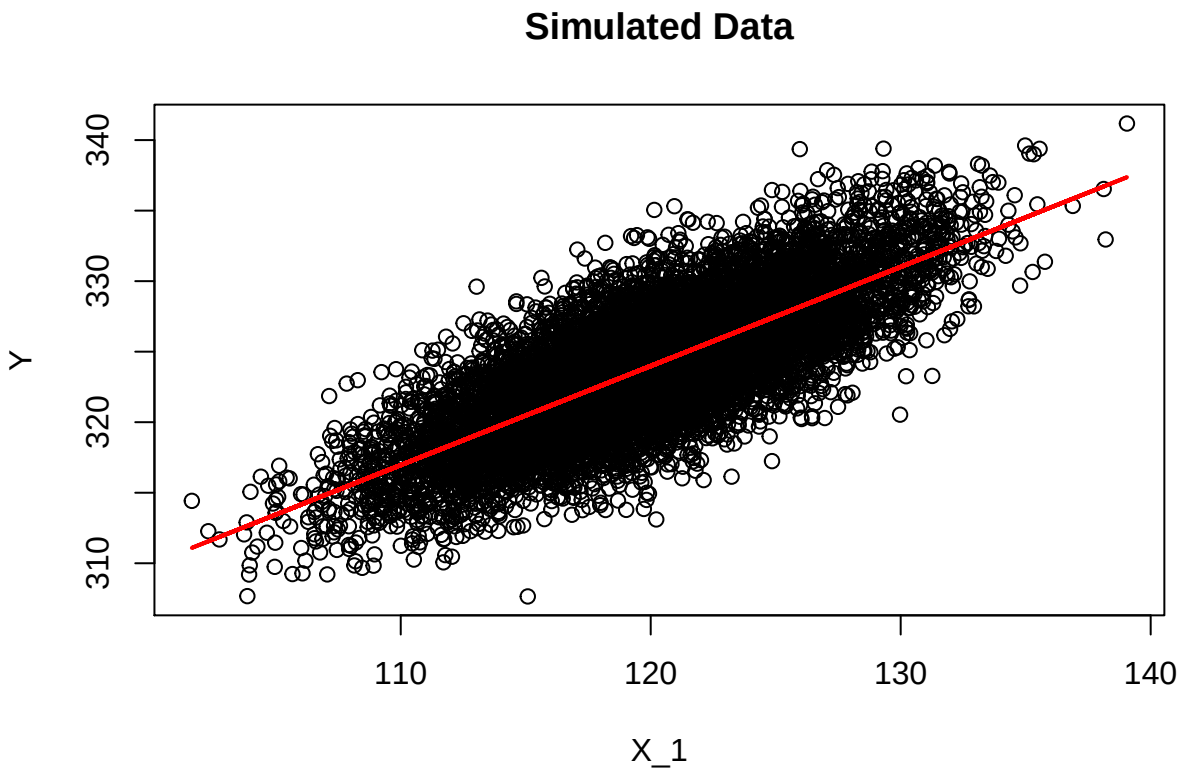
sum of squares summarizes this nicely:

$$\hat{\beta}_0 = \bar{Y}_i - \hat{\beta}_1 \bar{X}_{i,1}$$

$$\bar{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 \bar{X}_{i,1}$$

Here we are showing that the estimated $\hat{\beta}$ is capturing the underlying relationship between the average value of $\bar{X}_{i,1}$ and $\bar{Y}_i$. We can visualize this estimated relationship:

```
# Generate a plot
plot(our_data$X_1, our_data$Y, xlab = "X_1", ylab = "Y",
    main = "Simulated Data")
lines(our_data$X_1, mean(store_betas[, 1]) + mean(store_betas[,
    2]) * our_data$X_1, col = 2, lwd = 2)
```



More formally stated, we can show mathematicall how our predictions of Y_i vary as we vary $X_{i,1}$ through a partial derivative:

$$\frac{\partial Y_i}{\partial X_{i,1}} : \hat{\beta}_1$$

Goodness-of-fit

Define the total sum of squares (SST), the explained sum of squares (SSE), and the residual sum of squares (SSR) (also known as the sum of squared residuals), as follows:

$$SST = \sum_i (Y_i - \bar{Y})^2$$

$$SSE = \sum_i (\hat{Y}_i - \bar{Y})^2$$

$$SSR = \sum_i (\hat{Y}_i - Y_i)^2$$

SST is a measure of the total sample variation in the Y_i; that is, it measures how spread out the Y_i are in the sample. If we divide SST by $n-1$, we obtain the sample variance of Y.Similarly, SSE measures the sample variation in the $\hat{Y}_i$ (where we use the fact that $\bar{\hat{Y}} = \bar{Y}$), and SSR measures the sample variation in the residuals. The total variation in Y can always be expressed as the sum of the explained variation and the unexplained variation SSR.

$$SST = SSE + SSR$$

We can prove this as follows:

$$SST = \sum_i (Y_i - \bar{Y})^2 = \sum_i (Y_i - \hat{Y}_i + \hat{Y}_i - \bar{Y})^2$$

$$SST = \sum_i ((Y_i - \hat{Y}_i) + (\hat{Y}_i - \bar{Y}))^2$$

$$SST = \sum_i (\hat{\epsilon} + (\hat{Y}_i - \bar{Y}))^2$$

$$SST = \sum_i \hat{\epsilon}^2 + 2\sum_i \hat{\epsilon}(\hat{Y}_i - \bar{Y}) + \sum_i (\hat{Y}_i - \bar{Y})^2$$

$$SST = SSR + 2\sum_i \hat{\epsilon}(\hat{Y}_i - \bar{Y}) + SSE$$

Almost there! We just have this one additional term: $2\sum_i \hat{\epsilon}(\hat{Y}_i - \bar{Y})$, which is equivalent to the sample covariance between $\hat{\epsilon}$ and our fitted values $\hat{Y}$ if we were to divide by $n-1$. Lets think this through:

$$Cov(\hat{Y}_i, \hat{\epsilon}) = Cov(\hat{\beta}_0 + \hat{\beta}_1 X_{i,1}, \hat{\epsilon}) = Cov(\hat{\beta}_0, \hat{\epsilon}) + Cov(\hat{\beta}_1 X_{i,1}, \hat{\epsilon}) =$$

$$= 0 + \hat{\beta}_1 Cov(X_{i,1}, \hat{\epsilon}) = \hat{\beta}_1 Cov(X_{i,1}, Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_{i,1}) =$$

$$= \hat{\beta}_1 (Cov(X_{i,1}, Y_i) - Cov(X_{i,1}, \hat{\beta}_0) - \hat{\beta}_1 Cov(X_{i,1}, X_{i,1})) =$$

$$= \hat{\beta}_1 (Cov(X_{i,1}, Y_i) - \frac{Cov(X_{i,1}, Y_i)}{Var(X_{i,1})} Cov(X_{i,1}, X_{i,1})) = \hat{\beta}_1 (Cov(X_{i,1}, Y_i) - \frac{Cov(X_{i,1}, Y_i)}{Var(X_{i,1})} Var(X_{i,1})) =$$

$$= \hat{\beta}_1 (Cov(X_{i,1}, Y_i) - Cov(X_{i,1}, Y_i)) = 0$$

$$\therefore Cov(\hat{Y}_i, \hat{\epsilon}) = 0$$

which implies that:

$$SST = SSR + 2\sum_i \hat{\epsilon}(\hat{Y}_i - \bar{Y}) + SSE = SSR + SSE$$

It is often useful to compute a number that summarizes how well the OLS regression line fits the data. If we divide both sides of our total sum of squares equation by the total sum of squares we

will have:

$$\frac{SST}{SST} = 1 = \frac{SSR}{SST} + \frac{SSE}{SST}$$

Re-written, what we have is the $R-squared$ or coefficient of determination of our estimated model:

$$R^2 = \frac{SSE}{SST} = 1 - \frac{SSR}{SST}$$

R^2 is the ratio of the explained variation compared to the total variation; thus, it is interpreted as the fraction of the sample variation in Y_i that is explained by $X_{i,1}$. Note that R^2 is bounded by $[0, 1]$ based on the definition and proof we derived above of the total sum of squared error. A higher R^2 usually indicates a better fitting model (this isn't always true and is actually sometimes a sign of a problem in time-series).

Example

Now that we have an idea of how the OLS estimator works using simulated data, lets take go back to our example data on the school performance for reading and math scores.

```
# Load in the package
library(AER)
# Load in the data from the package into
# memory
data("CASchools")
# Preview the data
head(CASchools)
```

district school county grades students 1 75119 Sunol Glen Unified Alameda KK-08 195 2 61499 Manzanita Elementary Butte KK-08 240 3 61549 Thermalito Union Elementary Butte KK-08 1550 4 61457 Golden Feather Union Elementary Butte KK-08 243 5 61523 Palermo Union Elementary Butte KK-08 1335 6 62042 Burrel Union Elementary Fresno KK-08 137 teachers calworks lunch computer expenditure income english read 1 10.90 0.5102 2.0408 67 6384.911 22.690001 0.000000 691.6 2 11.15 15.4167 47.9167 101 5099.381 9.824000 4.583333 660.5 3 82.90 55.0323 76.3226 169 5501.955 8.978000 30.000002 636.3 4 14.00 36.4754 77.0492 85 7101.831 8.978000 0.000000 651.9 5 71.50 33.1086 78.4270 171 5235.988 9.080333 13.857677 641.8 6 6.40 12.3188 86.9565

25 5580.147 10.415000 12.408759 605.7 math 1 690.0 2 661.9 3 650.9 4 643.5 5 639.9 6 605.4

Suppose we want to study the relationship between classroom size and total test scores. Let us define total test score as follows:

$$\text{Total Test Score = Reading Score + Math Score}$$

Lets estimate classroom size by creating a student to teacher ratio (how many students per teacher):

$$STR = \frac{Students}{Teachers}$$

The lower STR the smaller the estimated classroom size (less students per teacher) and the higher STR the larger the estimated classroom size (more students per teacher).

Lets create two new columns that contain total test scores and our student to teacher ratio.

```
CASchools$total_test_score <- CASchools$read +
    CASchools$math
CASchools$STR <- CASchools$students/CASchools$teachers
```

Many universities/schools like to advertise that they have small classroom sizes, where the small classroom usually indicates that students will recieve a higher quality of education as the teachers will be able to manage the classroom better and to provide a more tailored education to individuals. We can try to measure the relationship between test scores and classroom size as follows:

$$\text{Total Test Score} = \beta_0 + \beta_1 STR$$

If the advertisements of universities/schools are in fact true we could formulate a testable hypothesis:

$$H_0 : \beta_1 < 0$$

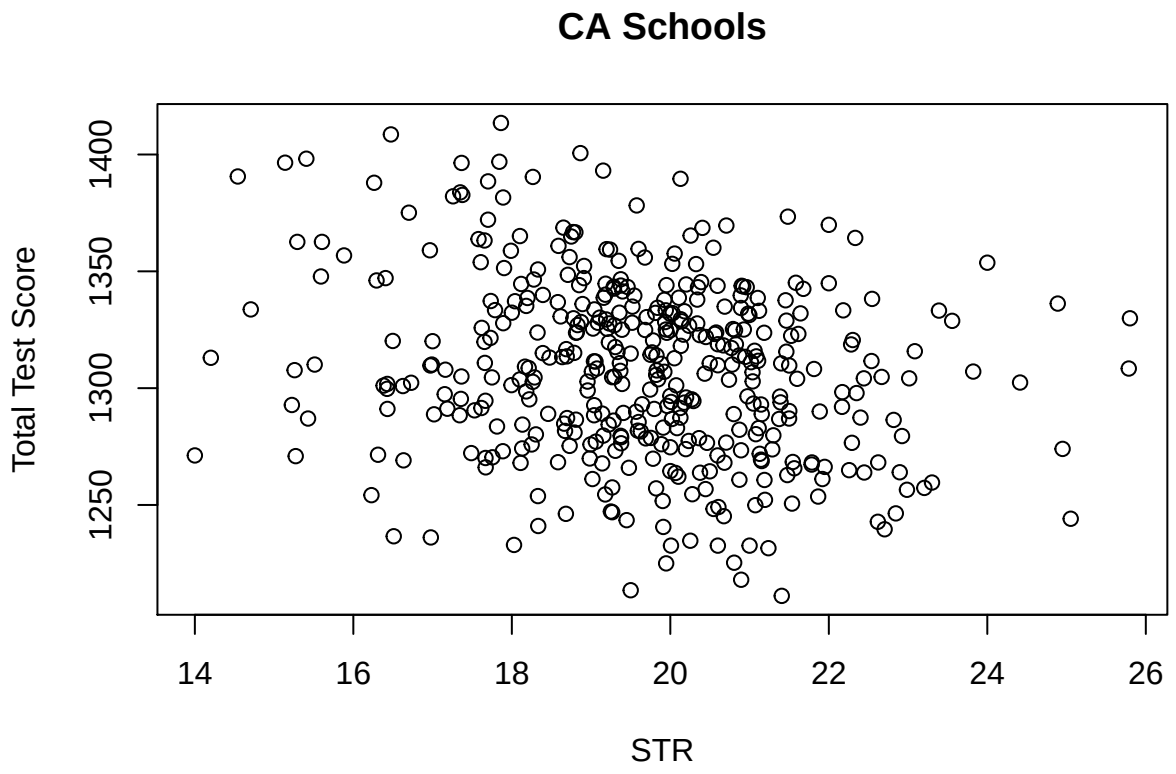
$$H_1 : \beta_1 \geq 0$$

Here our null hypothesis (H_0) is that the relationship between student to teacher ratio's and test

scores is negative: as STR increases (more students per teacher $\rightarrow$ larger classroom size) then test scores should decrease. Our alternative hypothesis (H_1) is that there is either no relationship between student to teacher ratio's and test scores or there is in fact a positive relationship between student to teacher ratio's and test scores. For example, a larger classroom may create a more diverse classroom where students can learn from one another.

Before jumping to estimating this relationship, lets visualize the relationship between test scores and our estimate of classroom size:

```
plot(CASchools$STR, CASchools$total_test_score,
    xlab = "STR", ylab = "Total Test Score", main = "CA Schools")
```



Hmm..no clear relationship. Lets run our regression and see what we come up with.

```
mdl <- baby_ols(dependent = CASchools$total_test_score,
    independent = CASchools$STR)
mdl
```

```
 beta_0      beta_1
```

1397.865899 -4.559616 Here our estimated coefficient on STR is -4.5596163. The sign of the coefficient is in fact negative and implies that a one unit incrase in the student to teacher ratio leads to a decline in the total testing score of approximately 5 points. On average a school in our sample has a student to teacher ratio of 20, which implies an average school test score as predicted by our model of 1308. If the student to teacher ratio increases by one unit (one extra student per teacher) this would lead to a predicted average test score of 1304.

Now recall that to conduct a hypothesis test we need a measure of a standard error for our estimator. In our simulation we estimated the standard error by repeatedly re-sampling from our true population data. We can actually conduct a similar process when dealing with real world data. We can simply re-sample rows from our datset with replacement and generate an estimate of the standard error.

```
# Set seed
set.seed(1)
# Create an empty matrix to store our
# coefficients
store_betas_str <- matrix(NA, nrow = 1000, ncol = 2)
for (i in 1:nrow(store_betas_str)) {
    # Loop through and take random samples from
    # our full dataset
    sample_data <- CASchools[sample(1:nrow(CASchools),
        nrow(CASchools), replace = T), ]
    # Estimate our model
    sample_mdl <- baby_ols(dependent = sample_data$total_test_score,
        independent = sample_data$STR)
    # Store the estimated Coefficients
    store_betas_str[i, ] <- sample_mdl
}
# Print Stats for Intercept (Beta_0)
mean(store_betas_str[, 1])
```

[1] 1396.648

```
sd(store_betas_str[, 1])
```

[1] 20.25405

```
# Print Stats for Slope (Beta_1)
mean(store_betas_str[, 2])
```

[1] -4.5029

```
sd(store_betas_str[, 2])
```

[1] 1.014793

Here our **bootstrapped** estimate of the standard error for β_1 is 1.0147926. We can use this estimate to conduct our hypothesis test:

$$T = \frac{-4.5029 - 0}{1.015} = -4.44$$

Given that our generated t-statistic is negative and large in absolute terms, we do not have enough information to reject our null hypothesis that larger classrooms lead to lower test scores. Note that this evidence supports the idea of having smaller classroom sizes improves test scores. We could have also re-formulated our hypothsis as stating larger classroom sizes improve test scores:

$$H_0 : \beta_1 > 0$$

$$H_1 : \beta_1 \leq 0$$

In which case we would state that we reject our null hypothesis as we found a statistically significant negative relationship between student to teacher ratios and overall test scores.

OLS using Matrices

Now that we have a general idea of how to use Ordinary Least Squares we are going to re-formulate the OLS estimator using Matrix algebra. This will allow us to more easily write out the properties of the estimator when dealing with several variables (more x's!!). We can write out the same equations we have been dealing with as follows:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}_{n\ x\ 1} = \begin{bmatrix} 1 & x_{11} & x_{12} & x_{13} & \dots & x_{1j} \\ 1 & x_{21} & x_{22} & x_{23} & \dots & x_{2j} \\ 1 & x_{n1} & x_{n2} & x_{n3} & \dots & x_{dj} \end{bmatrix}_{n\ x\ j} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_j \end{bmatrix}_{jx1} + \begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix}_{n\ x\ 1}$$

Which can be written as simply

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

Now previously we showed that the ordinary least squares solution is the set of β's that minimizes the residual sum of squares. In matrices we can write this out as:

$$\sum_i \hat{\epsilon}_i^2 = \epsilon'\epsilon = \begin{bmatrix} \epsilon_0 & \epsilon_1 & \dots & \epsilon_n \end{bmatrix}_{1\ x\ n} \begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix}_{n\ x\ 1} = \begin{bmatrix} \epsilon_0 * \epsilon_0 + \epsilon_1 * \epsilon_1 + \dots + \epsilon_n * \epsilon_n \end{bmatrix}_{1\ x\ 1}$$

Going back to our matrix formulation we then have that:

$$\epsilon'\epsilon = (Y - X\hat{\beta})'(Y - X\hat{\beta})$$

$$= Y'Y - \hat{\beta}'X'Y - Y'X\hat{\beta} + \hat{\beta}'X'X\hat{\beta}$$

$$= Y'Y - 2\hat{\beta}'X'Y + \hat{\beta}'X'X\hat{\beta}$$

where this development uses the fact that the transpose of a scalar is the scalar i.e. $Y'X\hat{\beta} = (Y'X\hat{\beta})' = \beta'X'Y$

To find the $\hat{\beta}$ that minimizes the sum of squared residuals, we need to take the derivative with respect to $\hat{\beta}$:

$$\frac{\partial \epsilon'\epsilon}{\partial \hat{\beta}} = -2X'Y + 2X'X\hat{\beta} = 0$$

$$\frac{\partial \epsilon' \epsilon}{\partial \hat{\beta}} = X'X\hat{\beta} = X'Y$$

$$X'X\hat{\beta} = X'Y$$

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

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

Where I is an identity matrix (a matrix of all 0's off diagonal and 1's along the diagnoal; basically 1 in matrix algebra).

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

With this formulation we now are able to estimate a vector of $\hat{\beta}$ using matrices! This formulation allows us to have up to j indpendent variables!

Using this formulation we can also derive a number of implied properties of Ordinary Least Squares that had previously skipped over. For example, if we go back to:

$$X'X\hat{\beta} = X'Y$$

If we substitue in for $Y = X\hat{\beta} + \epsilon$, we have:

$$X'X\hat{\beta} = X'(X\hat{\beta} + \epsilon)$$

$$X'X\hat{\beta} = X'X\hat{\beta} + X'\epsilon$$

$$X'X\hat{\beta} = X'X\hat{\beta} + X'\epsilon$$

$$0 = X'\epsilon$$

This tells us the following:

1) The observed values of X are uncorrelated with the residuals.

$X'\epsilon$ implies that every column of x_j of X, $x_j'\epsilon = 0$. In other words, each regressor has zero sample correlation with the residuals.

2) The sum of the residuals is zero.

If there is a constant, then the first column in X (i.e. X_1) will be a column of ones. This means that for the first element in the $X'\epsilon$ vector must be zero.

3) The sample mean of the residuals is zero (from 2, just divide by n).

4) The regression hyperplane passes through the means of the observed values.

This follows from the fact that $\bar{\epsilon} = 0$. Recall that $\epsilon = Y - X\hat{\beta}$, dividing by n we have $\bar{\epsilon} = \bar{Y} - \bar{X}\hat{\beta} = 0 \rightarrow \bar{Y} = \bar{X}\hat{\beta}$. This shows that the regression hyperplane goes through the point of means of the data.

5) The predicted values of Y are uncorrelated with the residual (Same proof we did for deriving R-Squared).

6) The mean of the predicted Y's for the sample will equal the mean of the observed Y's i.e. $\bar{\hat{Y}} = \bar{y}$