

# Linear regression 2

STA 032: Gateway to data science Lecture 24

Jingwei Xiong

May 8, 2023

# Recap

Last lecture we covered how the linear regression was invented.

Today we will go over the mathematics form of regression and it's derivation.

# Basics of Regression

- We observe a **response** or **dependent variable** (Y): Son's height
- With each Y, we also observe **regressors** or **predictors**  $\{X_1, \dots, X_n\}$  when we have  $n$  variables. *In this case,  $n=1$ , only Father's height*
- Goal: determine the mathematical relationship between response variables and regressors:

$$Y = h(X_1, \dots, X_n)$$

■ Which is to find the blue line!

- Function can be non-linear. When function is linear, the form will be:

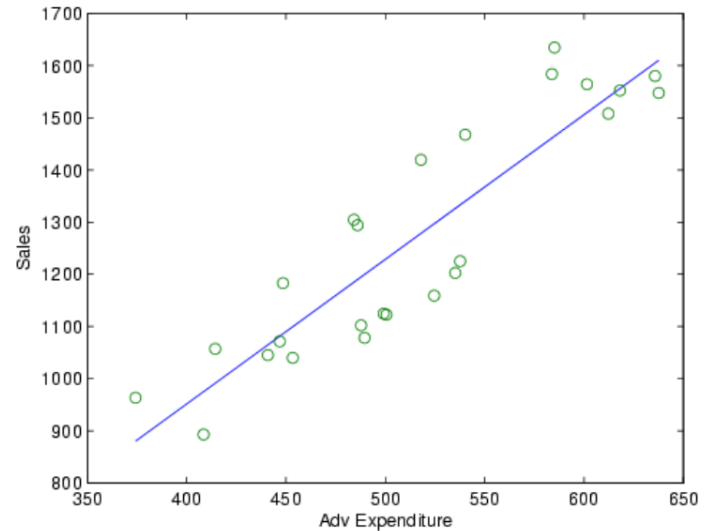
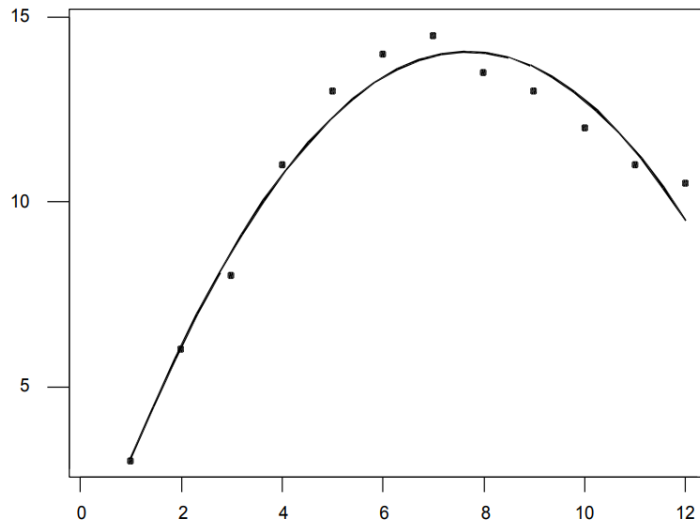
$$Y = h(X_1, \dots, X_n) = \beta_0 + \beta_1 X_1 + \dots + \beta_n X_n$$

# Response and Predictor Variables in Regression Analysis

- In regression analysis, we distinguish between two types of variables: the response variable and the predictor variable(s).
- The response variable, also known as the dependent variable, is the variable we want to predict or explain.
- The predictor variable(s), also known as independent variables or regressors, are the variables used to predict or explain the response variable.

- Function can be non-linear. When function is linear, the form will be:

$$Y = h(X_1, \dots, X_n) = \beta_0 + \beta_1 X_1 + \dots + \beta_n X_n$$



Question: If the association between response and predictor is positive then the slope is:

A. Positive. B Negative. C. Cannot determine

# Simple Linear Regression Model

Galton's father-and-son data inspire the following assumptions for the simple linear regression (SLR) model:

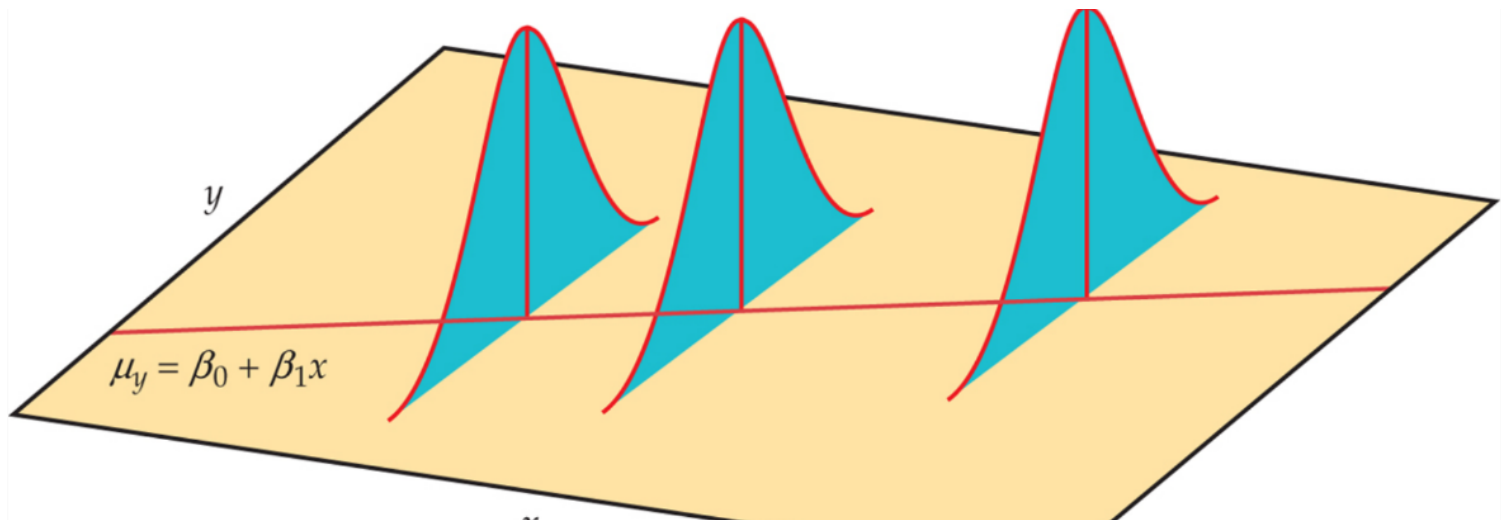
1. The means of  $Y$  is a linear function of  $X$ , i.e.

$$E(Y|X = x) = \beta_0 + \beta_1 x$$

2. The SD of  $Y$  does not change with  $x$ , i.e.,

$$SD(Y|X = x) = \sigma, \text{ for every } x$$

1. Within each subpopulation, the distribution of  $Y$  is normal.



# Simple Linear Regression Model

Equivalently, the SLR model asserts the values of  $X$  and  $Y$  for individuals in a population are related as follows

$$Y = \beta_0 + \beta_1 X + \varepsilon$$

- The value of  $\varepsilon$ , called the **error**, varies from observation to observation, follows a normal distribution:

$$\varepsilon \sim N(0, \sigma^2)$$

In this model, the line

$$y = \beta_0 + \beta_1 x$$

is called the **population regression line**.

- In other words, our model assume:

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2) \text{ for all } i$$

# Estimate regression parameters

Intuitively, last lecture we use the correlation to model the linear relationship between response and predictor. But why is this case?

- To estimate  $(\beta_0, \beta_1)$ , we find values that minimize squared error  $L$ : (Also called RSS residual sum of squares)

$$L = \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i))^2$$

- Where each term of the squared error is the square of the distance of one  $Y_i$  to the regression line!

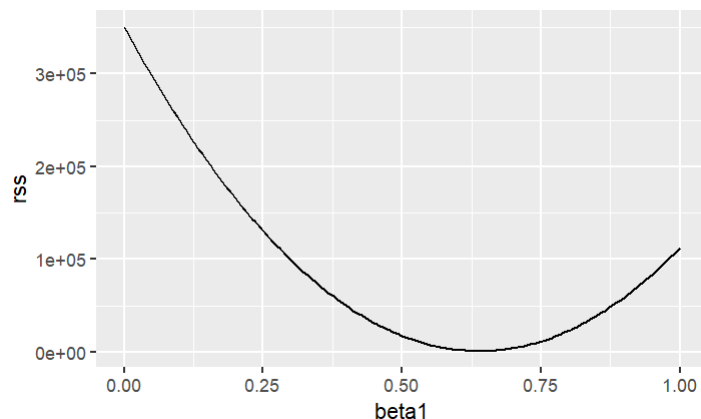


To show how to minimize the squared error  $L$ , we use the last lecture Galton's height data. Let's write a function that computes the RSS for any pair of values  $\beta_0$  and  $\beta_1$ :

```
rss <- function(beta0, beta1, data){  
  resid <- galton_heights$son - (beta0+beta1*galton_heights$father)  
  return(sum(resid^2))  
}
```

So for any pair of values, we get an RSS. Here is a plot of the RSS as a function of  $\beta_1$  when we keep the  $\beta_0$  fixed at 25.

```
beta1 = seq(0, 1, len=nrow(galton_heights))  
results <- data.frame(beta1 = beta1,  
                      rss = sapply(beta1, rss, beta0 = 25))  
results |> ggplot(aes(beta1, rss)) + geom_line() +  
  geom_line(aes(beta1, rss))
```



We can see a clear minimum for  $\beta_1$  at around 0.65. However, this minimum for  $\beta_1$  is for when  $\beta_0 = 25$ , a value we arbitrarily picked. We don't know if (25, 0.65) is the pair that minimizes the equation across all possible pairs.

Trial and error is not going to work in this case.

But we can use calculus: take the partial derivatives, set them to 0 and solve for  $\beta_1$  and  $\beta_2$ .

# Method of least squares

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i, \quad i = 1, 2, \dots, n$$

To estimate  $(\beta_0, \beta_1)$ , we find values that minimize squared error:

$$L = \sum_{i=1}^n \epsilon_i^2 = \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2$$

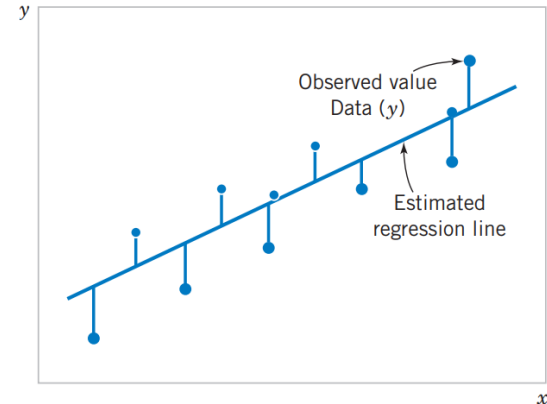
The least squares estimators of  $\beta_0$  and  $\beta_1$ , say,  $\hat{\beta}_0$  and  $\hat{\beta}_1$ , must satisfy

$$\left. \frac{\partial L}{\partial \beta_0} \right|_{\hat{\beta}_0, \hat{\beta}_1} = -2 \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) = 0$$

$$\left. \frac{\partial L}{\partial \beta_1} \right|_{\hat{\beta}_0, \hat{\beta}_1} = -2 \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i) x_i = 0$$

Least square normal equations

$$\begin{aligned} n\hat{\beta}_0 + \hat{\beta}_1 \sum_{i=1}^n x_i &= \sum_{i=1}^n y_i \\ \hat{\beta}_0 \sum_{i=1}^n x_i + \hat{\beta}_1 \sum_{i=1}^n x_i^2 &= \sum_{i=1}^n y_i x_i \end{aligned}$$



# Least square estimates

The **least squares estimates** of the intercept and slope in the simple linear regression model are

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \quad (11-7)$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n y_i x_i - \frac{\left(\sum_{i=1}^n y_i\right)\left(\sum_{i=1}^n x_i\right)}{n}}{\sum_{i=1}^n x_i^2 - \frac{\left(\sum_{i=1}^n x_i\right)^2}{n}} \quad (11-8)$$

where  $\bar{y} = (1/n) \sum_{i=1}^n y_i$  and  $\bar{x} = (1/n) \sum_{i=1}^n x_i$ .

# Alternative notation

$$S_{xx} = \sum_{i=1}^n (x_i - \bar{x})^2 = \sum_{i=1}^n x_i^2 - \frac{\left(\sum_{i=1}^n x_i\right)^2}{n} \quad (11-10)$$

$$S_{xy} = \sum_{i=1}^n (y_i - \bar{y})(x_i - \bar{x}) = \sum_{i=1}^n x_i y_i - \frac{\left(\sum_{i=1}^n x_i\right)\left(\sum_{i=1}^n y_i\right)}{n} \quad (11-11)$$

$$\left. \begin{array}{l} \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \\ \hat{\beta}_1 = \frac{S_{xy}}{S_{xx}} \end{array} \right\} \hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i \quad \text{Fitted (estimated) regression model}$$

20

Where  $\frac{S_{xy}}{S_{xx}} = r \frac{s_y}{s_x}$ ,

Where  $\frac{S_{xy}}{S_{xx}} = r \frac{s_y}{s_x}$ ,

So this is the same with our last lecture formula replacing population parameters by sample estimations.

$$y = b + mx \text{ with slope } m = \rho \frac{\sigma_y}{\sigma_x} \text{ and intercept } b = \mu_y - m\mu_x$$

- To interpret the result: 1 unit increase in X will result in  $\hat{\beta}_1$  unit increase in Y!

# Caution: Sample V.S. Population

Note the population regression line:

$$y = \beta_0 + \beta_1 x$$

is **different** from the least square regression line: (Where we estimate  $\beta$ s using LSE):

$$y = \hat{\beta}_0 + \hat{\beta}_1 x$$

- The latter is merely the least square line for our **sample**. The former is the regression relationship for entire **population**.
- The values of  $\hat{\beta}_0, \hat{\beta}_1$  will change from sample to sample:

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}} = r \frac{s_y}{s_x}, \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

- We are interested in the population parameter  $\beta_0, \beta_1$ , **not** the sample counterparts  $\hat{\beta}_0, \hat{\beta}_1$ .

- How close is  $\hat{\beta}_1$  to  $\beta_1$ ?

By some algebra (You will learn it in STA108), the slope of the least square line is:

$$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}} = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sum_i (x_i - \bar{x})^2}$$

Under the SLR model:  $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$ , replacing  $y_i$  in the  $\hat{\beta}_1$  formula, we can show after some algebra that:

$$\hat{\beta}_1 = \beta_1 + \frac{\sum_i (x_i - \bar{x})\varepsilon_i}{\sum_i (x_i - \bar{x})^2}$$

From the above, one can get the mean, the SD and the sampling distribution of  $\hat{\beta}_1$ : (Detail in STA108)

- $E\hat{\beta}_1 = \beta_1$ ,  $\hat{\beta}_1$  is an **unbiased** estimate of  $\beta_1$

$$SD(\hat{\beta}_1) = \frac{\sigma}{\sqrt{\sum_i (x_i - \bar{x})^2}} = \frac{\sigma}{s_x \sqrt{n-1}}$$

Where  $s_x$  is the sample SD of  $x_i$ 's.



# Estimate of $\sigma$

- **Residuals** are the value between SLR prediction and true observation:

$$e_i = y_i - \hat{y}_i = y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i)$$

- By the property of Least Square estimation,  $\sum_i e_i = 0$
- We use the "**sample SD**" of the residuals  $e_i$  to estimate error variance  $\sigma$ :

$$s_e = \sqrt{\frac{\sum_i (e_i - \bar{e})^2}{n - 2}} = \sqrt{\frac{\sum_i e_i^2}{n - 2}}$$

- Mean of residuals is 0:  $\bar{e} = \sum_i e_i / n = 0$
- Note here we divide by  $n - 2$ , not  $n - 1$ , because we lose two degrees of freedom as we estimate two parameters  $\beta_0$  and  $\beta_1$ .
  - In multiple linear regression, it would be  $n - p$ , when we have  $p$  parameters.

# Sampling distribution of $\hat{\beta}_1$

Recall that

$$SD(\hat{\beta}_1) = \frac{\sigma}{\sqrt{\sum_i (x_i - \bar{x})^2}}$$

But  $\sigma$  is unknown, so we estimate it with  $s_e$ . Then the estimated SD of  $\hat{\beta}_1$  is called the **standard error (SE)** of  $\hat{\beta}_1$ :

$$SE(\hat{\beta}_1) = \frac{s_e}{\sqrt{\sum_i (x_i - \bar{x})^2}}$$

The **sampling distribution** of  $\hat{\beta}_1$  is normal:

$$\hat{\beta}_1 \sim N(\beta_1, \frac{\sigma^2}{\sum_i (x_i - \bar{x})^2})$$

$$z = \frac{\hat{\beta}_1 - \beta_1}{\sigma / \sqrt{\sum_i (x_i - \bar{x})^2}} \sim N(0, 1)$$

$$z = \frac{\hat{\beta}_1 - \beta_1}{\sigma / \sqrt{\sum_i (x_i - \bar{x})^2}} \sim N(0, 1)$$

This is (approx.) valid:

- either if the errors  $\varepsilon_i$  are i.i.d  $N(0, \sigma^2)$
- or if the errors  $\varepsilon_i$  are independent and the sample size  $n$  is large.

As  $\sigma$  is unknown, if replaced with  $s_e$  (sample standard deviation of error), the t-statistic below has a t-distribution with  $n - 2$  degrees of freedom:

$$T = \frac{\hat{\beta}_1 - \beta_1}{s_e / \sqrt{\sum_i (x_i - \bar{x})^2}} = \frac{\hat{\beta}_1 - \beta_1}{SE(\hat{\beta}_1)} \sim t_{n-2}$$

# Confidence intervals for $\beta_1$

Then the  $(1 - \alpha)$  **confidence interval** for  $\beta_1$  is given as:

$$\hat{\beta}_1 - t^* SE(\hat{\beta}_1), \hat{\beta}_1 + t^* SE(\hat{\beta}_1)$$

Where  $t^*$  is the critical value for the  $t_{n-2}$  distribution at confidence level  $1 - \alpha$

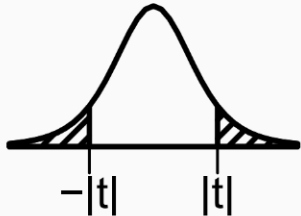
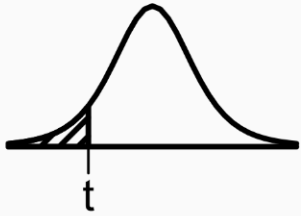
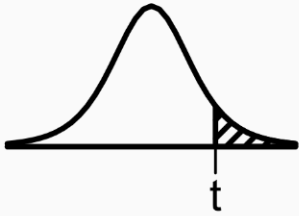
`t_star = qt(1 - alpha / 2, df = n - 2)`

## Tests for $\beta_1$

To test the hypothesis  $H_0 : \beta_1 = a$ , we use the  $t$ -statistic

$$t = \frac{b_1 - a}{SE(b_1)} \sim t_{n-2}$$

The  $p$ -value can be computed using the  $t$ -table based on the  $H_a$ :

$H_a$	$\beta_1 \neq a$	$\beta_1 < a$	$\beta_1 > a$
$P$ -value			

Observe that testing  $H_0 : \beta_1 = 0$  is equivalent to testing whether  $x$  is useful in predicting  $y$  linearly.

- It is possible that  $r$  is small but  $\beta_1$  is significantly different from 0.

# The lm function (and it's summary)

- Regression in R is as simple as **lm(y ~ x)**, in which "lm" stands for "linear model".

To fit the model:

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$$

with  $Y_i$  the son's height and  $x_i$  the father's height, we can use this code to obtain the least squares estimates.

```
fit <- lm(son ~ father, data = galton_heights)
fit$coef
```

```
(Intercept)      father
  37.287605      0.461392
```

The most common way we use `lm` is by using the character `~` to let `lm` know which is the variable we are predicting (left of `~`) and which we are using to predict (right of `~`). The intercept is added automatically to the model that will be fit.

# Summary of lm function

The object `fit` includes more information about the fit. We can use the function `summary` to extract more of this information (not shown):

```
summary(fit)
```

Call:

```
lm(formula = son ~ father, data = galton_heights)
```

Residuals:

Min	1Q	Median	3Q	Max
-9.3543	-1.5657	-0.0078	1.7263	9.4150

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	37.28761	4.98618	7.478	3.37e-12 ***
father	0.46139	0.07211	6.398	1.36e-09 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.45 on 177 degrees of freedom

Multiple R-squared: 0.1878, Adjusted R-squared: 0.1833

F-statistic: 40.94 on 1 and 177 DF, p-value: 1.36e-09

To understand some of the information included in this summary we need to remember that the LSE are random variables. Mathematical statistics gives us some ideas of the distribution of these random variables

## LSE ( $\hat{\beta}_0, \hat{\beta}_1$ ) are random variables

The LSE (  $\hat{\beta}_0, \hat{\beta}_1$  ) is derived from the data  $y_1, \dots, y_N$ , which are a realization of random variables  $Y_1, \dots, Y_N$ .

This implies that our estimates are random variables.

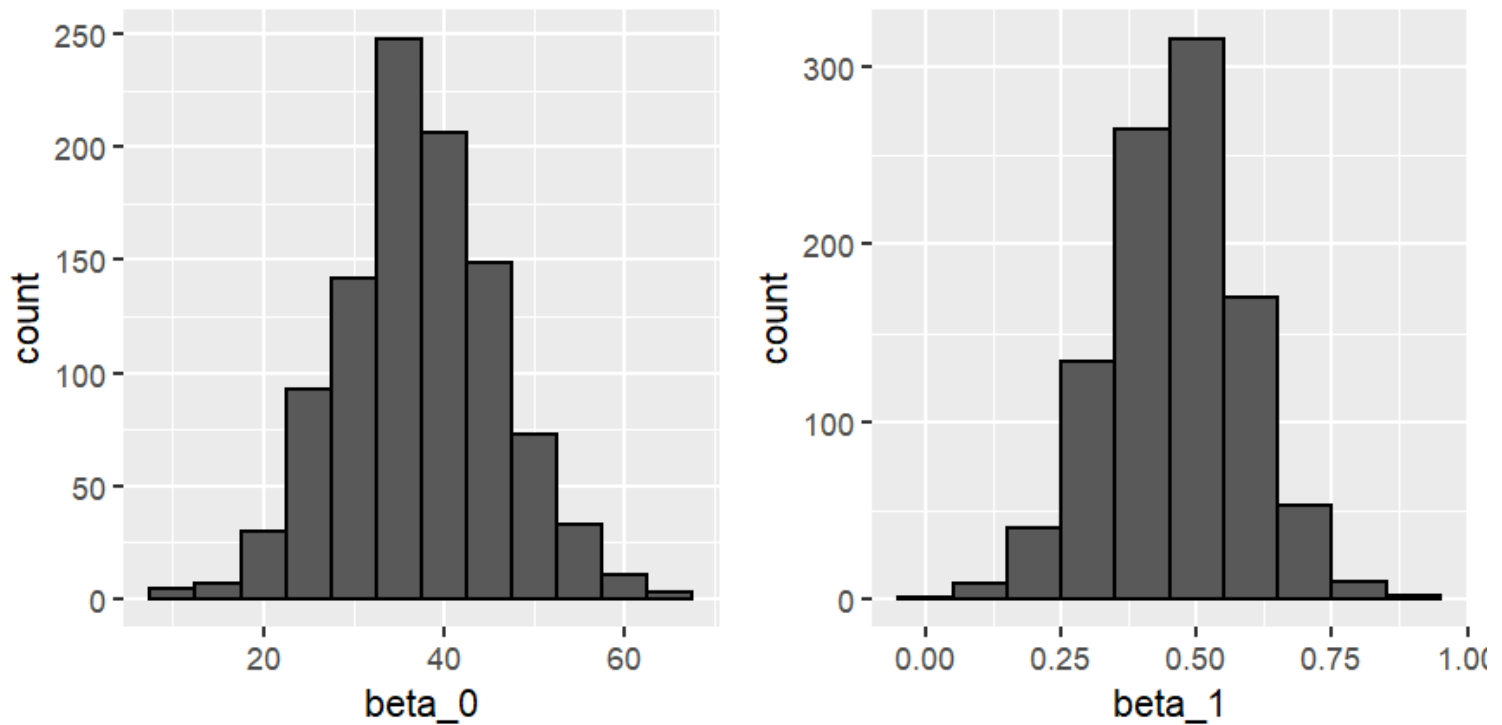
To see this, we can run a Monte Carlo simulation in which we assume the son and father height data defines a population, take a random sample of size  $N = 50$ , and compute the regression slope coefficient for each one:

```
library(tidyverse)
B <- 1000
N <- 50
lse <- replicate(B, {
  sample_n(galton_heights, N, replace = TRUE) %>%
    lm(son ~ father, data = .) %>%
    coef()
})
lse <- data.frame(beta_0 = lse[,1], beta_1 = lse[,2])
```

We can see the variability of the estimates by plotting their distributions:



We can see the variability of the estimates by plotting their distributions:



The reason these look normal is because the central limit theorem applies here as well: for large enough  $N$ , the least squares estimates will be approximately normal with expected value  $\beta_0$  and  $\beta_1$ , respectively.

For the standard deviation of the limit distribution of the  $\beta_0$  and  $\beta_1$ , they are called: **standard errors**

The standard errors are a bit complicated to compute, but mathematical theory does allow us to compute them and they are included in the summary provided by the `lm` function. Here it is for one of our simulated data sets:

```
sample_n(galton_heights, N, replace = TRUE) %>%  
  lm(son ~ father, data = .) %>%  
  summary() %>%  
  coef()
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	19.2791952	11.6564590	1.653950	0.1046637693
father	0.7198756	0.1693834	4.249977	0.0000979167

You can see that the standard errors estimates reported by the summary are close to the standard errors from the simulation:

```
lse |> summarize(se_0 = sd(beta_0), se_1 = sd(beta_1))
```

	se_0	se_1
1	8.83591	0.1278812

```
sample_n(galton_heights, N, replace = TRUE) %>%
  lm(son ~ father, data = .) %>%
  summary() %>%
  coef()
```

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	60.2849066	10.9564442	5.5022328	1.430103e-06
father	0.1355222	0.1573947	0.8610342	3.935000e-01

- The summary function also reports **t-statistics** (t value) and **p-values** (Pr(>|t|)).
- The t-statistic is not actually based on the central limit theorem but rather on the assumption that the  $\varepsilon$ s follow a normal distribution.
- Under this assumption, mathematical theory tells us that the LSE divided by their standard error,  $\hat{\beta}_0/\hat{SE}(\hat{\beta}_0)$  and  $\hat{\beta}_1/\hat{SE}(\hat{\beta}_1)$ , follow a t-distribution with  $N - p$  degrees of freedom, with  $p$  the number of parameters in our model.
- In the case of height  $p = 2$ , the two p-values are testing the null hypothesis that  $\beta_0 = 0$  and  $\beta_1 = 0$ , respectively.
- hypothesis testing with regression models is commonly used in epidemiology and economics to make statements such as "the effect of A on B was statistically significant after adjusting for X, Y, and Z".

# Predicted values are random variables

Once we fit our model, we can obtain prediction of  $Y$  by plugging in the estimates into the regression model. For example, if the father's height is  $x$ , then our prediction  $\hat{Y}$  for the son's height will be:

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

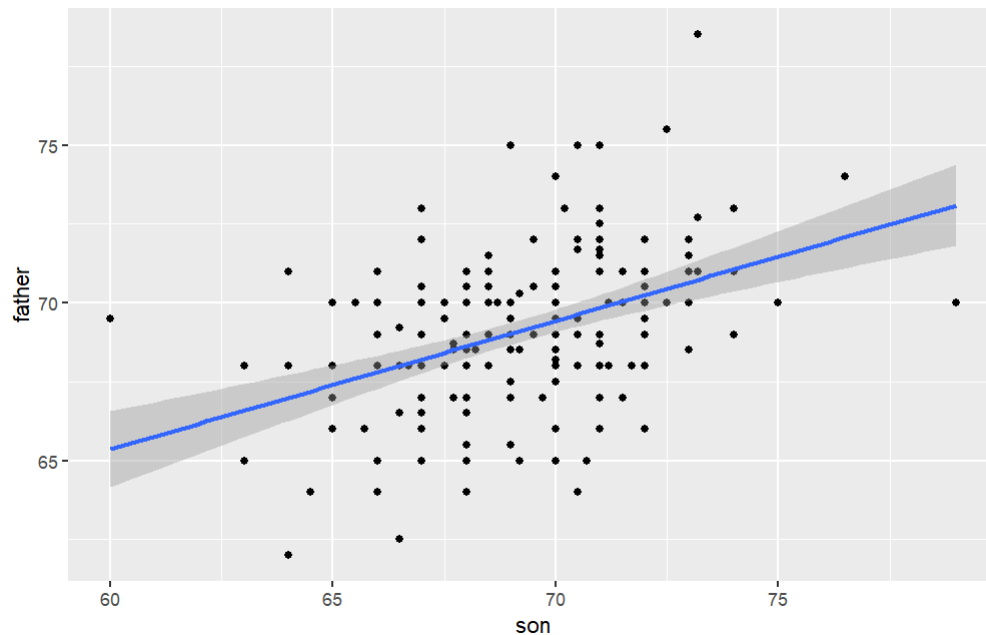
When we plot  $\hat{Y}$  versus  $x$ , we see the regression line.

Keep in mind that the prediction  $\hat{Y}$  is also a random variable and mathematical theory tells us what the standard errors are.

It is because  $\hat{\beta}_0, \hat{\beta}_1$  are random variables, has their own standard error. Thus the  $\hat{Y}$  is also a random variable and has it's own standard error.

If we assume the errors are normal, or have a large enough sample size, we can use theory to construct confidence intervals as well. In fact, the **ggplot2** layer `geom_smooth(method = "lm")` that we previously used plots  $\hat{Y}$  and surrounds it by confidence intervals:

```
galton_heights |> ggplot(aes(son, father)) +  
  geom_point() +  
  geom_smooth(method = "lm")
```



The R function `predict` takes an `lm` object as input and returns the prediction. If requested, the standard errors and other information from which we can construct confidence intervals is provided:

```
fit <- galton_heights %>% lm(son ~ father, data = .)
y_hat <- predict(fit, se.fit = TRUE)
names(y_hat)
```

```
[1] "fit"                "se.fit"             "df"                 "residual.scale"
```

