Research Methods in English Linguistics
Inferential Statistics 2: Simple Linear Regression

Hyunah Ahn

October 29, 2020

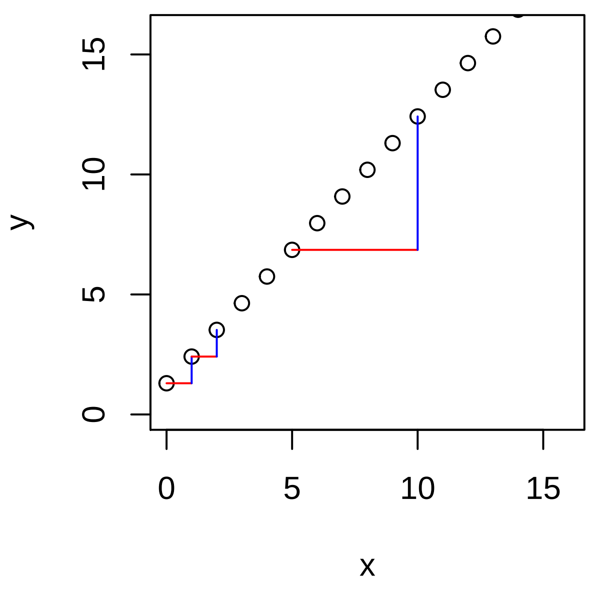
1 Change of Y over change of X

Regression, especially simple linear regression is about finding the rate at which Y changes as X changes. We first need to know the Y value when X is zero (the intercept of Y) and the amount of change in Y per unit of X (slope). If you remember simple formulas from elementary and middle school math classes as below, simple linear regression is quite simple and easy. a is the intercept and b is the coefficient (or slope).

$Y = a + bX$

The coefficient b is expressed as change (or difference) in Y over change in X. The Δ sign is for 'difference' and read 'delta.'

$b = \frac{\Delta Y}{\Delta X}$



As you can see from the plot above, y changes (increases) as x increases. What do you think is the intercept? Also, for each change in x (red line), what is your estimate for change in y (blue)?

Your guess for the intercept (a):

Your guess for the slope (b):

Using your estimates for a and b, write a formula below:

$y = a + bx$

$y =$

R has a basic function for simple linear regression `lm()`. You can find the intercept and coefficient using the following code.

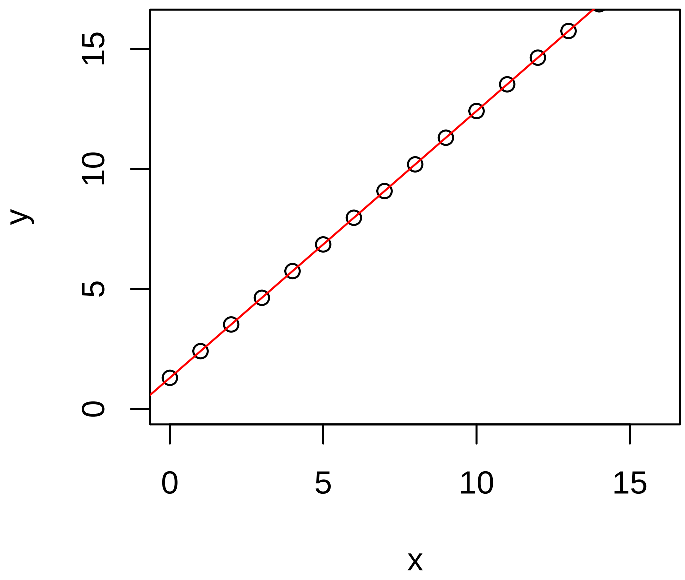
```
lm(y~x)
```

```
Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
      1.300        1.112
```

Is the output close to your estimation? We can add a regression line to the plot above.

```
plot(y~x, ylim=c(0, 16), xlim=c(0, 16)) # plot change of y over change of x
abline(lm(y~x), col=”red”) # draw a regression line
```



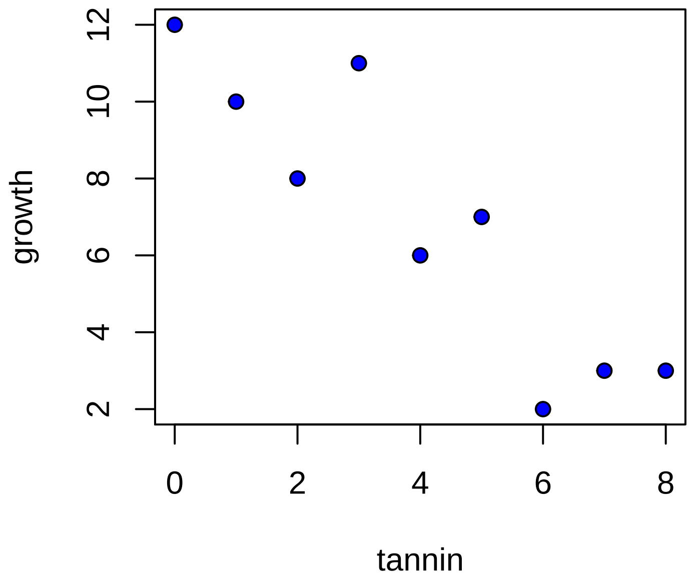
As you can see from the plot, all data points are on the regression line. No data deviates from the red line. Unfortunately, no natural data align this perfectly on a regression line. Let’s take a look at real data below.

2 Handling actual data

As you can see in the plot below, growth decreases as tannin increases. You can see the general tendency but when you compare each data point, increase in tannin doesn't always lead to decrease in growth.

```
data←read.csv("C:/Users/hyuna/OneDrive/Documents/01SNU/03GraduateSeminar/data/tannin.csv") # load
     'tannin.csv'

attach(data) # attach data
plot(growth~tannin, pch=21, bg="blue") #plot growth over tannin
```



```
lm(growth~tannin) #
```

```

Call:
lm(formula = growth ~ tannin)

Coefficients:
(Intercept)       tannin
     11.756       -1.217
```

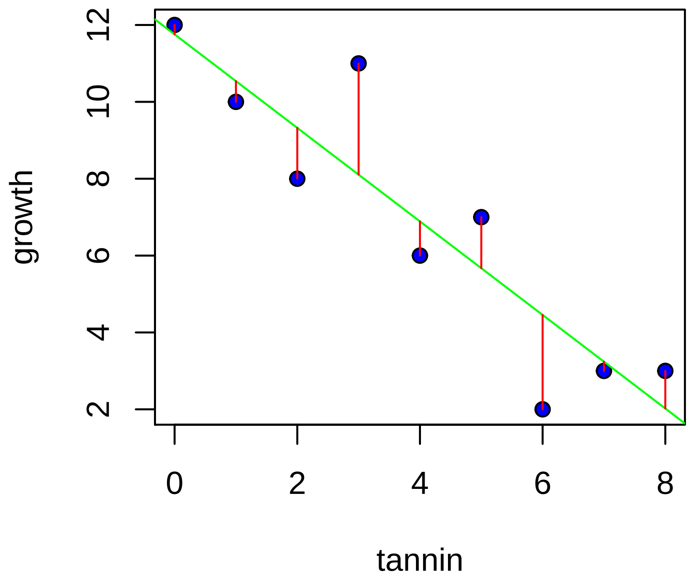
When you want to report the general tendency of the data, you want to find the a and b values for the formula $y = a + bx$ that fit the data the best, that maximizes the likelihood of data (maximum likelihood estimation. In doing so, what is important is to see 'residuals' or 'errors.' Errors in regression mean the distance between observed data and predicted data. From the output of the function `lm(growth~tannin)`, we obtained the estiamte for the intercept and coefficient. When the tannin level is 0, the growth value is 11.756 and for every unit increase of tannin, the growth level decreases about 1.217. We can fit a regression line and measure distances between true data points and the fitted (predicted) line.

```
plot(growth~tannin, pch=21, bg="blue") #plot growth over tannin
abline(lm(growth~tannin), col="green") # draw the regression line

fitted ←predict(lm(growth~tannin)) # estimate growth values based on the regression output
fitted # Print the predicted values
```

```
        1         2         3         4         5         6         7         8
11.755556 10.538889  9.322222  8.105556  6.888889  5.672222  4.455556  3.238889
        9
 2.022222
```

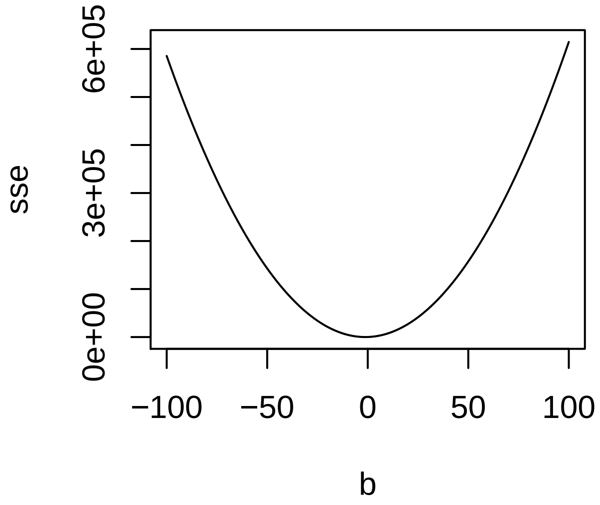
```
for(i in 1:9)lines(c(tannin[i], tannin[i]), c(growth[i], fitted[i]), col="red") # draw lines
    between observed and fitted data points
```



The task at hand is to find the regression line that will minimize the length of the red lines in the plot. If you remember previous lessons, calculating the lengths of the red lines should be quite similar to calculating the variance. Here, what's important is to figure out the regression line that will minimizes the sum of squares (SSE: error sum of squares). You will remember from your high school mass class that you can calculate the slope of an infinitesimal point on a curved plot as below. When given a set of data, you can make a range of wild guesses as to what regression line will fit your data the best. When you calculate all possibilities of regression lines, and it turns out, the slope lines are the derivations of the curve.

```
b←seq(-100, 100, 0.02) # Let's make extremely wild guesses that the plot we see above can have a
    slope of -100 to 100.
sse←numeric(length(b)) # Create a numeric vector that has the same length as b.
for (i in 1:length(b)){
  a←mean(growth)-b[i]*mean(tannin) # a = mean(y) - b * mean(x); from y = a+bx; For each esimated
      coefficient, calculate a new intercept
  residual←growth-(a+b[i]*tannin) # observed data - fitted data (a+bx); measure the distance
      between true data and fitted data
  sse[i]←sum(residual^2) # Square the residual (error), sum all the squares of residuals, and
      fill the sse vector.
}

plot(sse~b, type ="l")
```

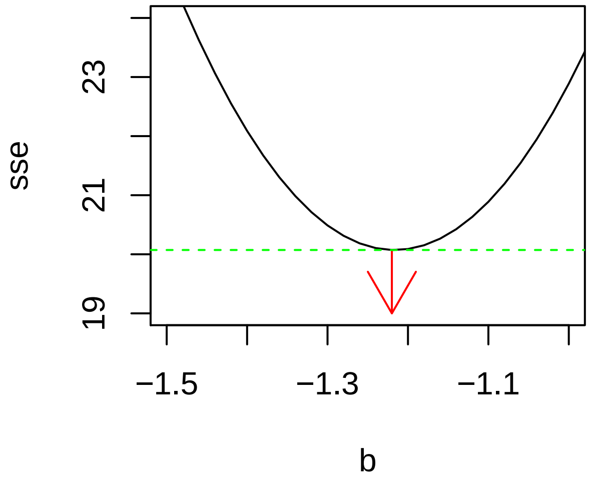


Now, zoom in on the lowest point of the curve (that is where the smallest value of sse is.)

```
plot(sse~b, type ="l", xlim=c(−1.5, −1), ylim=c(19, 24)) # plot the bottom tip
smallest.b←b[which(sse==min(sse))] # Find the coefficient estimate where the error sum of squres
    is the smallest
smallest.b # print it (it's actually the same as the coefficient value from the output table)
```

```
[1] -1.22
```

```
arrows(smallest.b, min(sse), smallest.b, 19, col = "red") # draw an arrow
abline(h=min(sse), col="green", lty=2) # an horizontal line (a derivated slope line where the sse
    is the smallest)
```



3 Maximum likelihood estimate of the slope

The error sum of squares is calculated as below. You first need to measure the distance between each data point and estiamted data point. The estimated y value is marked as $\hat{y}$. The distance $(y - \hat{y})$ should be squared (as we always have in calculating variance) and summed. Hence, the name is error sum of squares.

(1) $\sum(y - \hat{y})^2$

Now, one thing you should remember is that the estimated y value ($\hat{y}$) is a function of x with the intercept a and the slope b.

(2) $\hat{y} = a + bx$

Now, let's combine (1) and (2) to make (3).

(3) $\sum(y - (a + bx))^2 \rightarrow \sum(y - a - bx)^2$

We now need to calculate the value of b when the error sum of squares in the formula above (3) becomes the smallest, which can be done through partial derivatives. In derivatives, we know that when $f(x) = x^n$, its derivative is $f'(x) = nx^{n-1}$.

$f(x) = x^3 + 5x^2 + 6$ $\frac{\partial f}{\partial x} = \frac{\partial}{\partial x} f = 3x + 10x$

However, things can be a bit more complicated when we have more than two variables x and y. If we have the function of x as in (4) but need to get the derivative of it in terms of b as in (5), we need to use the partial derivation (6).

(4) $SSE = f(x) = (y - a - bx)^2$

(5) $\frac{\partial f}{\partial b}$

(6) $\frac{\partial f}{\partial u} * \frac{\partial u}{\partial b}$

Let's temporarilly replace the parenthesized part of (4) with u, then, (6) can be calculated in two steps of (7) and (8).

(7) $\frac{\partial f}{\partial u} = \frac{\partial}{\partial u} u^2 = \sum 2u$

(8) $\frac{\partial u}{\partial b} = \frac{\partial}{\partial b}(y - a - bx) = -x$

Then, following (6), you can get the product of (7) and (8) as in (9).

$(9) 2u * -x = -2x(u) = -2x(y - a - bx)$

With the original sum symbol, the derivative of $\sum(y - a - bx)^2$ in terms of b is expressed as (10).

(10) $\frac{\partial SSE}{\partial b} = -2\sum x(y - a - bx) = -2\sum xy - ax - bx^2$

When the change of SSE is infinitely small given the change of the slope (b), the value will be infinitely close to zero as in (11).

(11) $-2\sum xy - ax - bx^2 = 0$

By multiplying $-\frac{1}{2}$ on both sides, you can get (12).

(12) $\sum xy - \sum ax - \sum bx^2 = 0$

Remember that a is the intercept in the regression? Then, you can easily use the function $\bar{y} = a + b\bar{x}$ to get $a = \bar{y} - b\bar{x}$ and replace a with it as in (13).

(13) $\sum xy - (\bar{y} - b\bar{x})\sum x - b\sum x^2 = 0$

$\bar{y}$ is calculated by dividing the sum of all y values by the number of y values (14) and the same goes for $\bar{x}$.

(14) $\bar{y} = \frac{\sum y}{n}$

(15) $\bar{x} = \frac{\sum x}{n}$

Then, (13) can be re-written as (16).

(16) $\sum xy - [\frac{\sum y}{n} - b\frac{\sum x}{n}]\sum x - b\sum x^2 = 0$

(17) results from multiplying out the bracketed term.

(17) $\sum xy - \frac{\sum x \sum y}{n} + b\frac{(\sum x)^2}{n} - b\sum x^2 = 0$

The latter two terms with b can be moved to the right (18)

(18) $\sum xy - \frac{\sum x \sum y}{n} = b\sum x^2 - b\frac{(\sum x)^2}{n}$

(19) and (20) will show the process of finding b.

(19) $\sum xy - \frac{\sum x \sum y}{n} = b(\sum x^2 - \frac{(\sum x)^2}{n})$

(20) $b = \frac{\sum xy - \sum x \sum y/n}{\sum x^2 - (\sum x)^2/n}$

The numerator of (20) happens to be the corrected sum of the products of x and y (SSXY) and the denominator is the corrected sum of x.

(21) $b = \frac{SSXY}{SSX}$

We can use the function in (20) and calculate the slope on our own.

```
(sum(tannin*growth)-(sum(tannin)*sum(growth))/length(tannin))/(sum(tannin^2)-(sum(tannin)^2)/
    length(tannin))
```

```
[1] -1.216667
```

The value you get is the same as the slope you found from the function `lm(growth tannin)`.

4 How much data is explained by the regression?

Even if you can find the least squares estimate of the regression slope, it is not meaningful if data are scattered too far away from the regression line. For more information, let's summarize the linear model as below.

```
model<-lm(growth~tannin)
summary(model)
```

```

Call:
lm(formula = growth ~ tannin)

Residuals:
    Min      1Q  Median      3Q     Max
-2.4556 -0.8889 -0.2389  0.9778  2.8944

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  11.7556     1.0408  11.295 9.54e-06 ***
tannin       -1.2167     0.2186  -5.565 0.000846 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.693 on 7 degrees of freedom
Multiple R^2:  0.8157,   Adjusted R^2:  0.7893
F-statistic: 30.97 on 1 and 7 DF,  p-value: 0.0008461
```

Growth decreased as a funciton of tannin. When there is 0 amount of tannin, the growth level is 11.7556 `(Intercept)`. For every unit of tannin increase, the growth level goes 1.2167 unit down. The two parameters are shown in the estimate column above. The standard error column shows the ranges of error in the intercept and slope estimates. The t value column shows that both the intercept and the tannin factor are significantly different from 0.

Below the coefficients table is the list of other statistics. R-squared values show the percentage of data explained by the regression line. Let's take a closer look at the ANOVA statistics.

```
summary.aov(model)
```

```
            Df Sum Sq Mean Sq F value   Pr(>F)    
tannin       1  88.82   88.82   30.97 0.000846 ***
Residuals    7  20.07    2.87                     
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

The ANOVA table shows the ratio of data change as a function of a variable (SSX) to data change not explained by the variable (SSE).

The sum of squares for the variable `tannin` is 88.82, the sum of squares of error `Residuals` is 20.07. The degrees of freedom are calculated as below. To estimate the slope, you only needed to work out one parameter (b), so the degree of freedom is 1. SSE was calculated by working on $(y - a - bx)$, where two parameters a and b were estimated. Then, the degree of freedom is the total number of data points minus two. The mean of squares is obtained by dividing the sum of squares by degrees of freedom. Finally, the F-value is calculated by dividing the mean of squares of the regression by the mean of squares of the error (residuals). You can find the threshold F value using the degrees of freedom.