Programming Exercise 16
Linear Regression

C# Step by Step

In this programming exercise, you will write a program that calculates the Y intercept and slope of the “line of best fit” to implement linear regression.

Suppose you had a series of data points that were correlated in some manner. For example, weights and heights, square footages and home prices, time in service and rank, age and income, and so on. You would expect, generally speking, that people who weigh more are taller, homes that are larger cost more, service members with longer service have higher rank, older people have higher incomes, etc. Suppose, given your series of data points, you were instructed to write a program that would predict, given the first value (e.g., weight, house size), the second value (e.g., height, house price). That is, given a person’s weight to predict his height, given square footage to predict the home price, given time in service to predict the serviceman’s rank, and so on. How would you do it?

One way to think about the problem is to picture the data points plotted on a two dimensional surface, the well known coordinate plane, with an X axis and a Y axis. If you plot the predictor variable on the X axis, weight, square footage, etc., and the predicted variable on the Y axis, you should be able to see the trend — that as X increases, Y will also increase (or decrease, if there is an inverse correlation). If you could draw a line of best fit, you could easily take a new value of X and see where the X value is located on the line, then match that value to the Y axis to arrive at your prediction. This method is called *linear regression*. But, how to you draw a line of best fit? After all, there are an infinite number of lines that can be drawn on a two dimensional surface. How do you choose one line from an infinite number of lines?

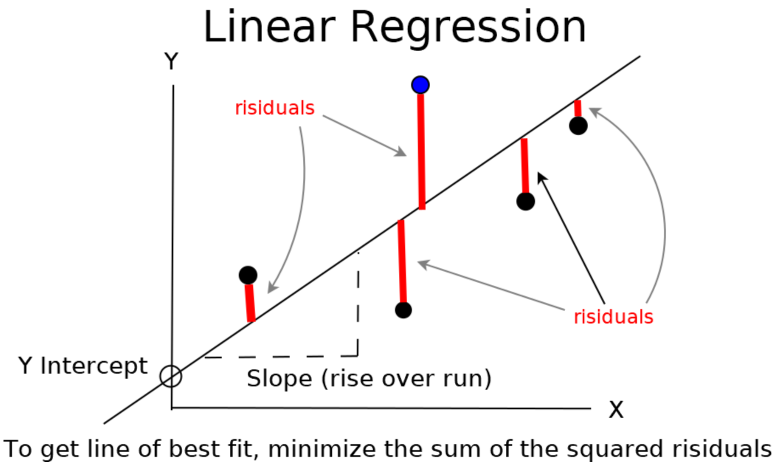

Figure 1: Linear regression illustration

We can see this with a picture, figure 1. This figure contains five points, and the X values appear to be positively correlated with the Y values. We have drawn a sloping line, rising from left to right, that seems to “fit” the points in some manner. The line is merely a hypothesis, we do not know if it is a good fit or not. It appears that all of points are some distance from the line, and we say that the vertical distance from each point to the line is a *residual*. If a point were exactly on the line, the value of the residual would be zero. The greater the vertical distance from the line, the greater the value of the residual. If somehow we could draw a line that minimized the sum of the (squared) residuals,[1] we could call it the "line of best fit."

Let's think about this problem in two parts: (1) how do you describe a line, and (2) how do you find the line of best fit.

Describing a line A line is described mathematically by the Y intercept plus the slope of the line (a coefficient) times some value, see equation 1. The Y intercept and the slope are constants. Here, alpha (α) is the Y intercept, beta (β) is the slope coefficient, X is the predictor variable, and Y is the predicted variable. Given X, we can easily plug it into this formula, if only we knew α and β. So, how do we calculate α and β?

$$Y = \alpha + \beta(X) \tag{1}$$

Calculating best fit In the equations below, n is the number of data points, x represents the predictor variables, and y represents the predicted variables. Equation 2 gives the value of α. Equation 3 gives the value of β. Make sure that you understand the difference between $\sum x^2$ and $(\sum x)^2$.

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

$$\beta = \frac{n \sum xy - \sum x \sum y}{n \sum x^2 - (\sum x)^2} \tag{3}$$

1 Data representation — 70 points

First, you will need to implement an appropriate data representation. You will have a set of points: (X, Y). You will need to access these points in your program. What kind of data structure will allow you to access these points? Here are some data points to get you started:

Name	Weight	Height
Andre	165	66
Aaron	185	71
James	190	70
Charles	210	72

2 Calculating values — 80 points

To use the formulas, you will need to find the required values. Calculate $\sum x$, $\sum x^2$, $\sum y$, $\sum xy$, and n.

3 Calculating and returning α and β — 90 points

Implement the formulas using the calculated values. Your program should output something like this, assuming α is 33 and β is 44:

```
The value of alpha is: 33
The value of beta is: 44
The formula is: y = 33 + 44 * x
```

4 Prediction — 100 points

Write a program implementing linear regression, which should take a given data set, calculate the formula of the line of best fit, request the user to enter a value of a predictor, and output the predicted value.

[1] We square the residuals to avoid negative numbers and to give a greater weight to the points further away from the regression line.