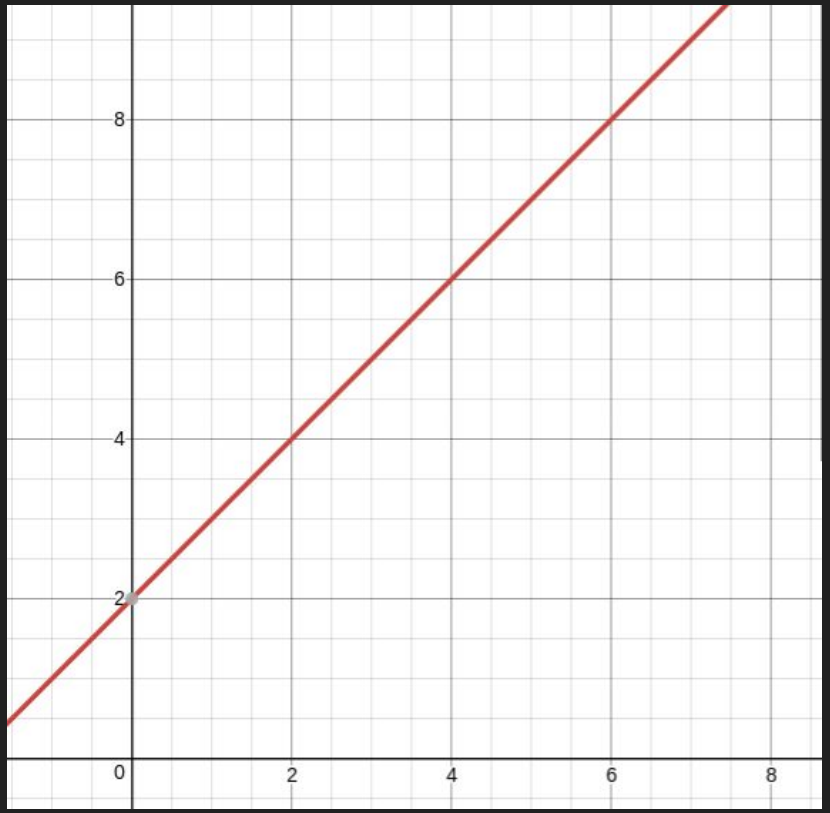
Session 1: Linear Regression with One Variable

And a Python review

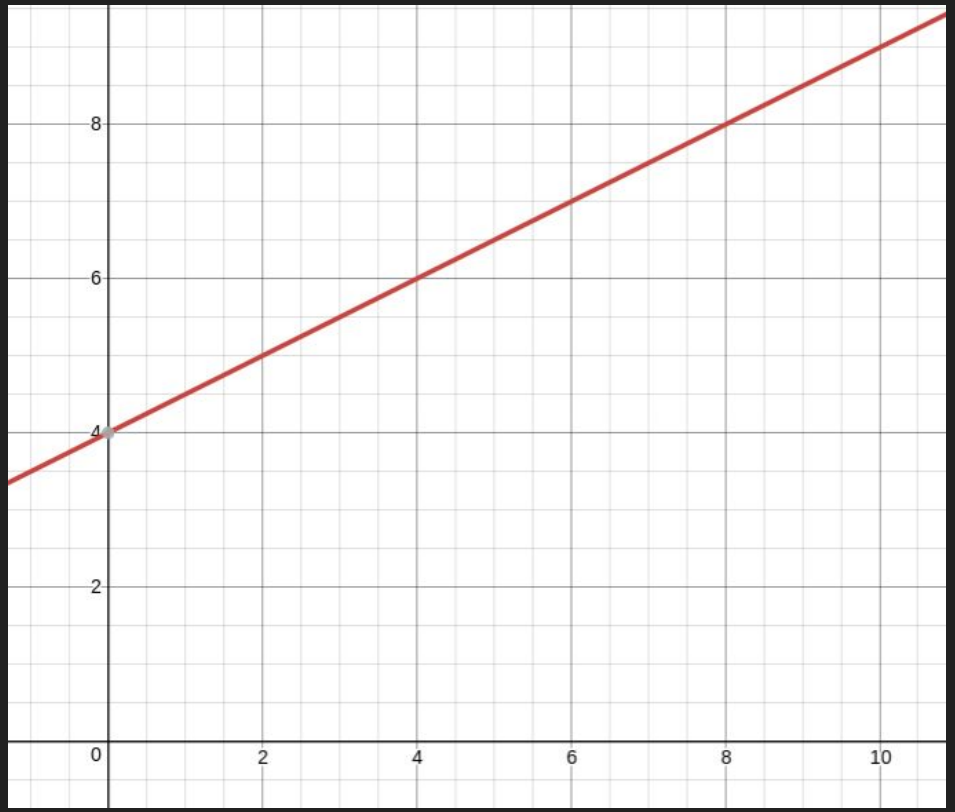
A linear equation

$$y = 1x + 2$$

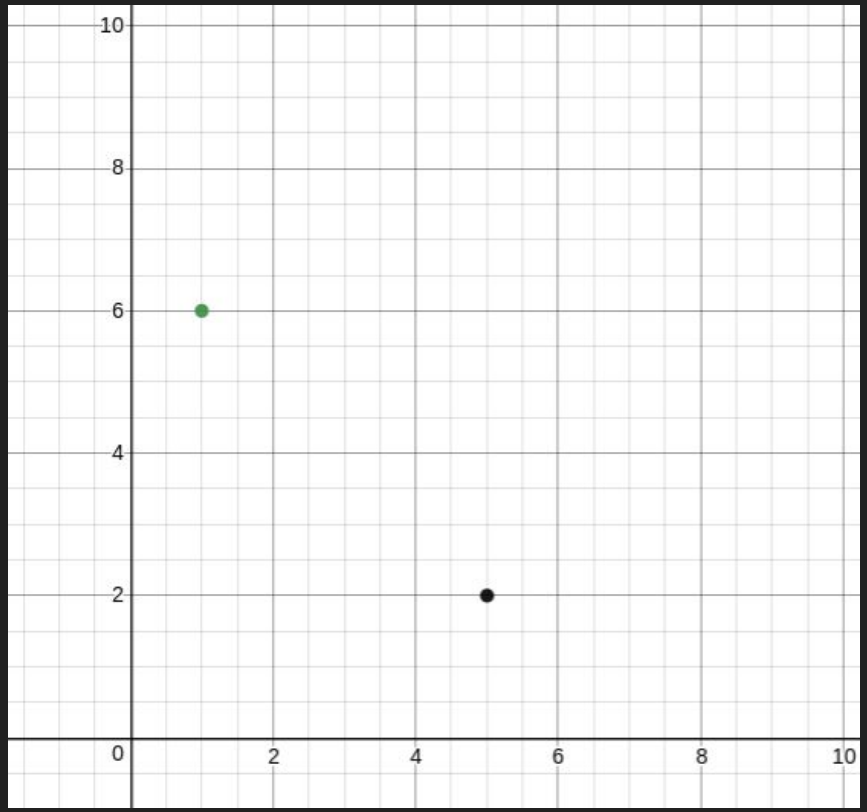


Another linear equation

$$y = 0.5x + 4$$



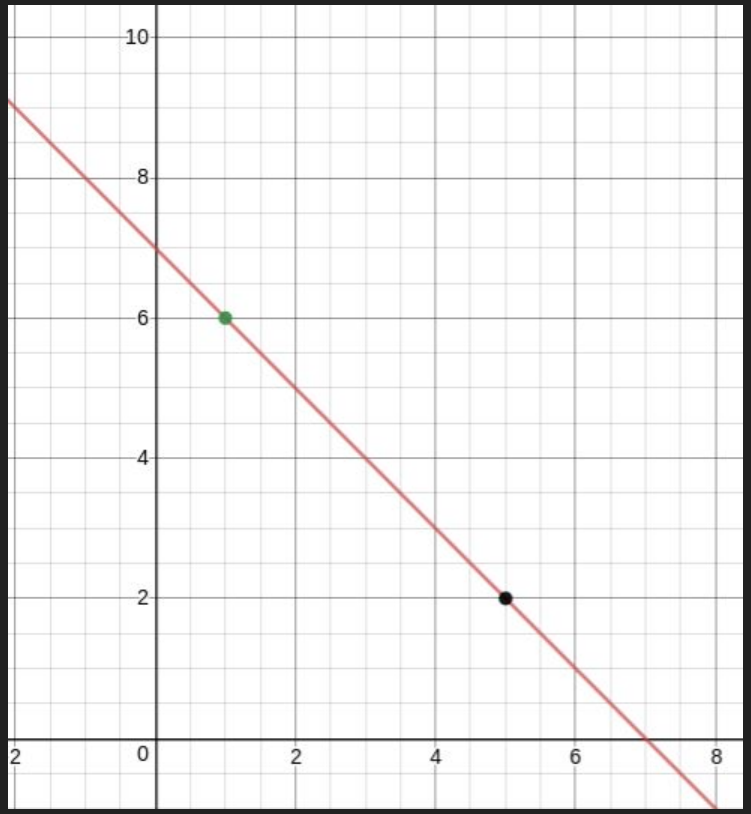
What if we instead have two values?



$$y = mx + b$$

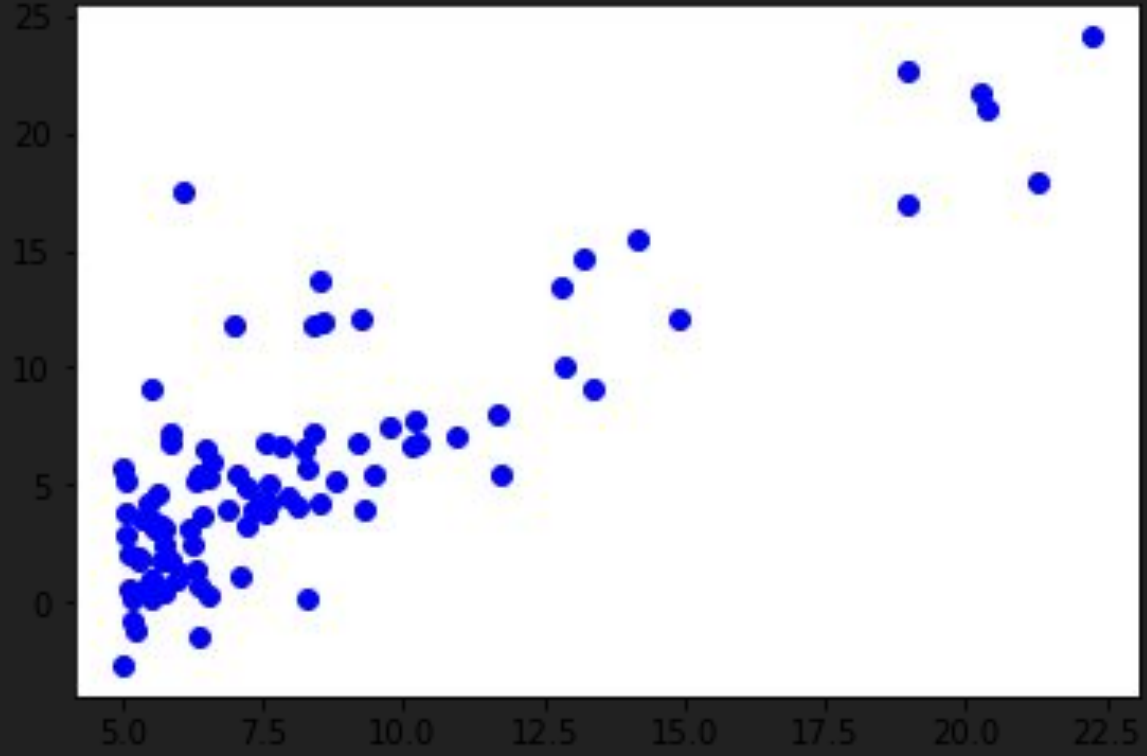
$$slope = \frac{(y_2 - y_1)}{(x_2 - x_1)}$$

We solve the equations and find the line equation



$$y = -x + 7$$

What if we have a bunch of values?



We need a way to generalize that behaviour

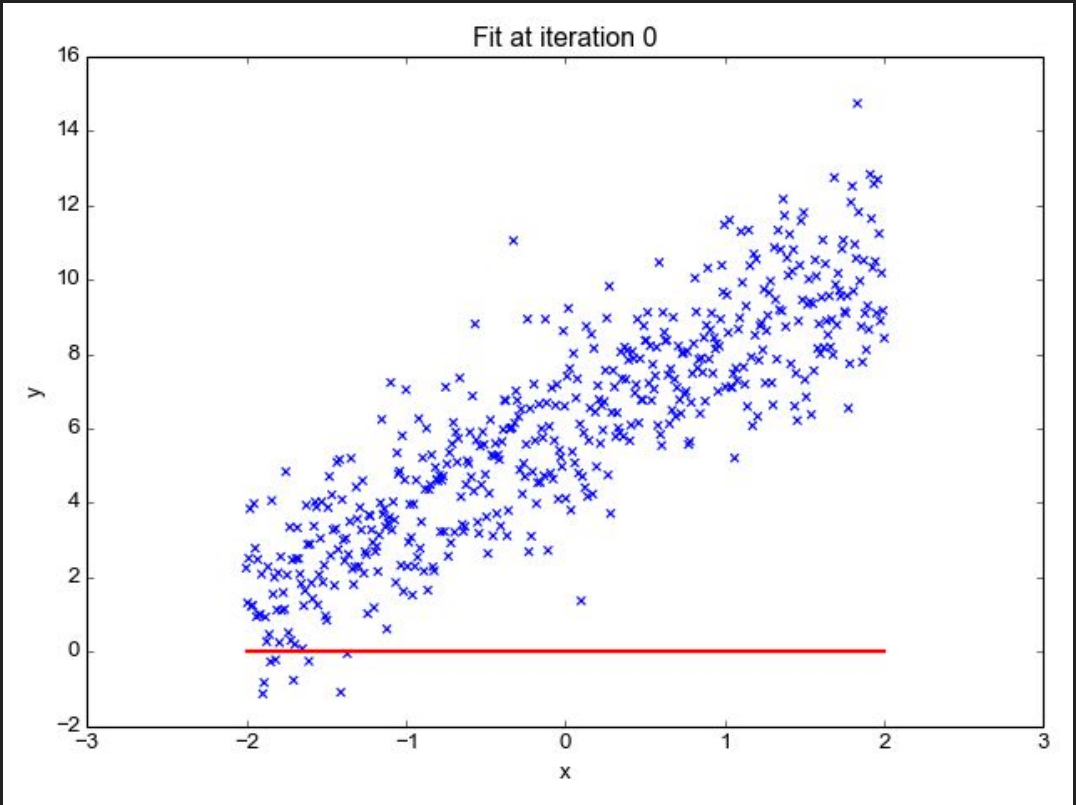
Enter linear regression!

For a set of values (x, y) we want to find the best model $y = \theta x + \beta$ such that it minimizes the error.

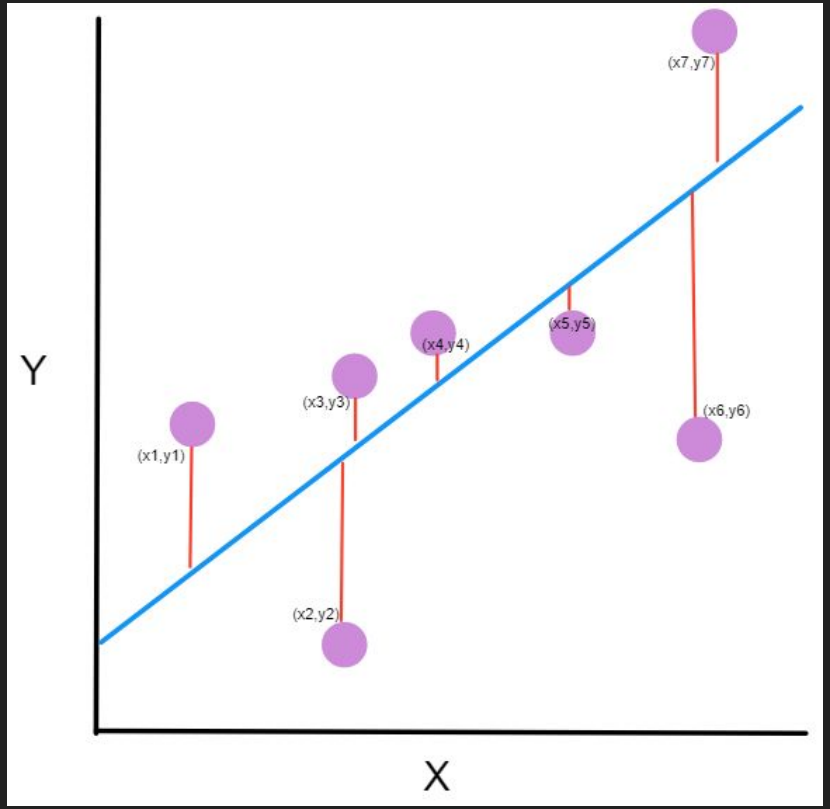
We will measure the error with the Mean Squared Error (MSE):

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2.$$

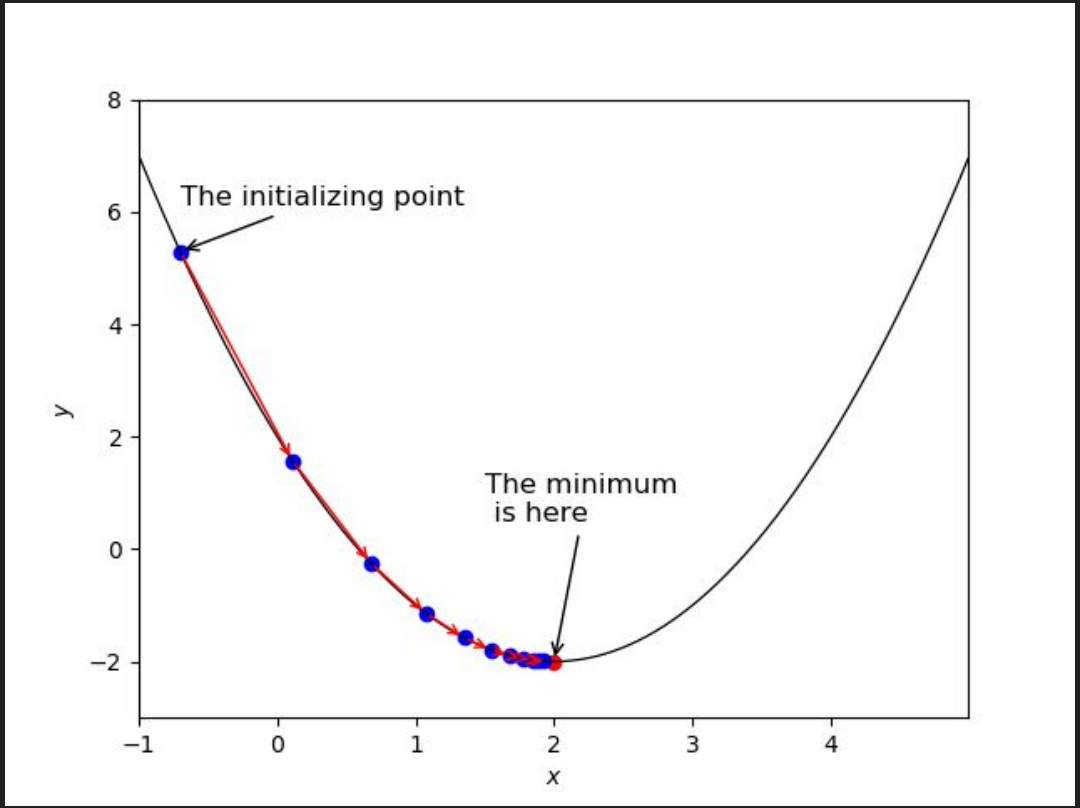
This is a more visual explanation of that definition



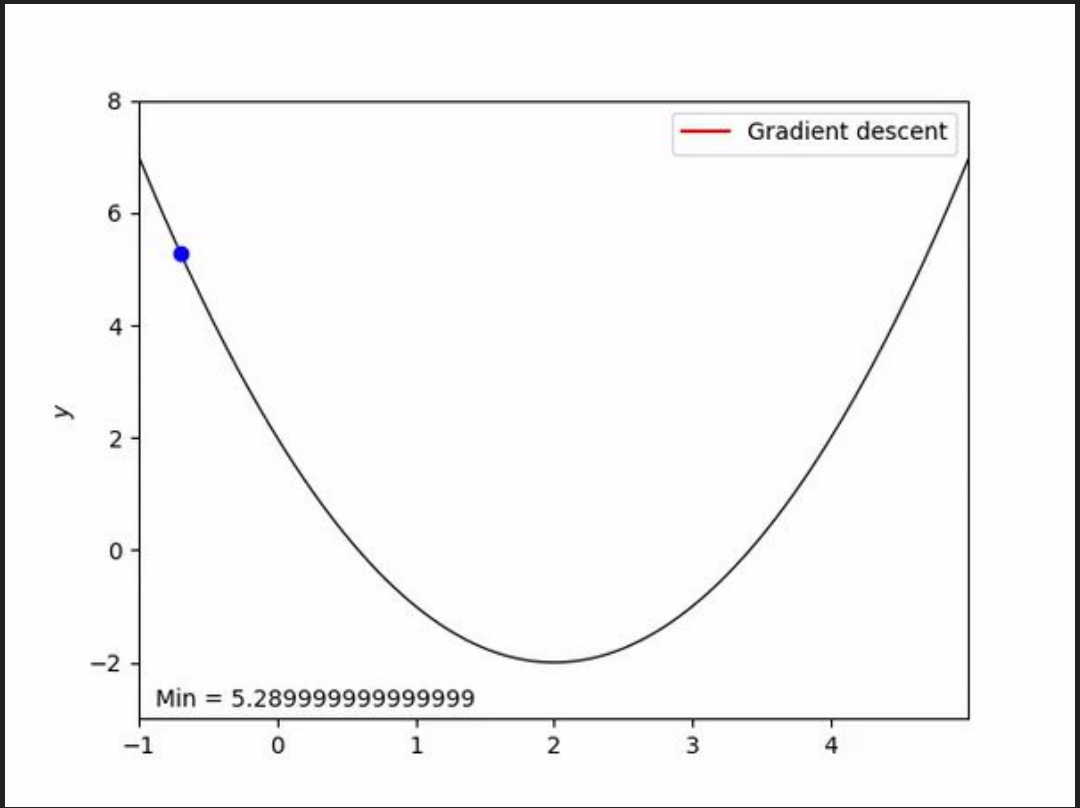
And we try to minimize the length of the red lines



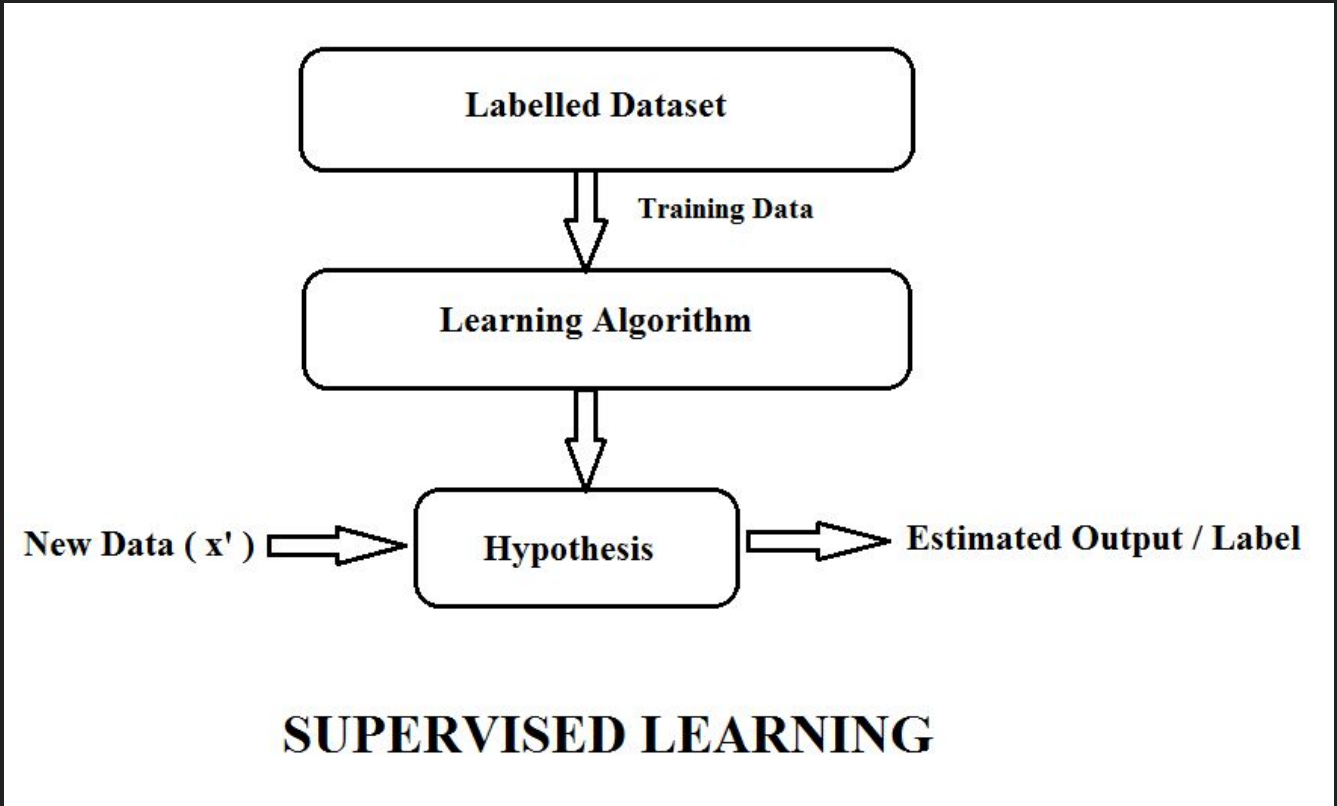
To do that we use gradient descent



An animation of gradient descent



How does this fit in machine learning?



What about unsupervised learning?

We throw data to the computer and hope that it gives us something good.

Now let’s play with Python! :D