Linear Regression

Model Representation:

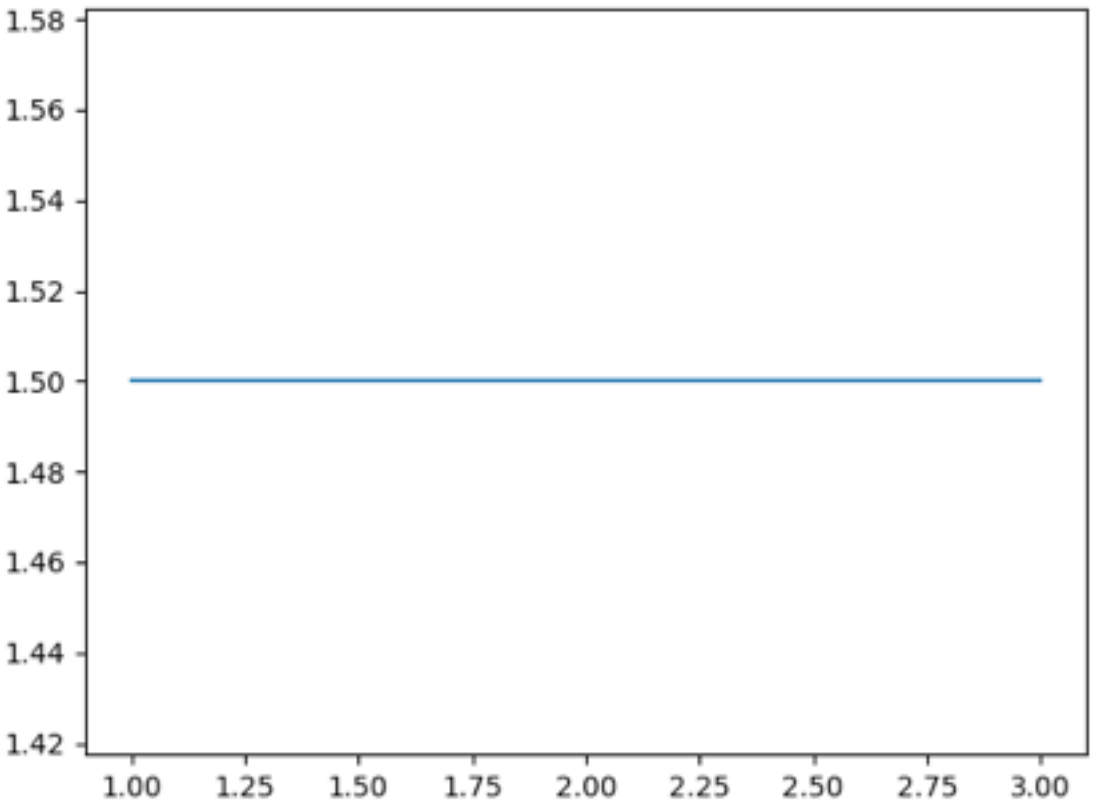
The goal of linear regression machine learning algorithm is to construct a model: a hypothesis that can be used to estimate Y based on X. The hypothesis, or model, maps inputs to outputs. So, for example, say I train a model based on a bunch of housing data that includes the size of the house and the sale price. By training a model, I can give you an estimate on how much you can sell your house for based on it's size. This is an example of a regression problem given some input, we want to predict a continuous output.

The hypothesis is usually presented as where theta values are the parameters.:

$$h_\theta(x) = \theta_0 + \theta_1 x$$

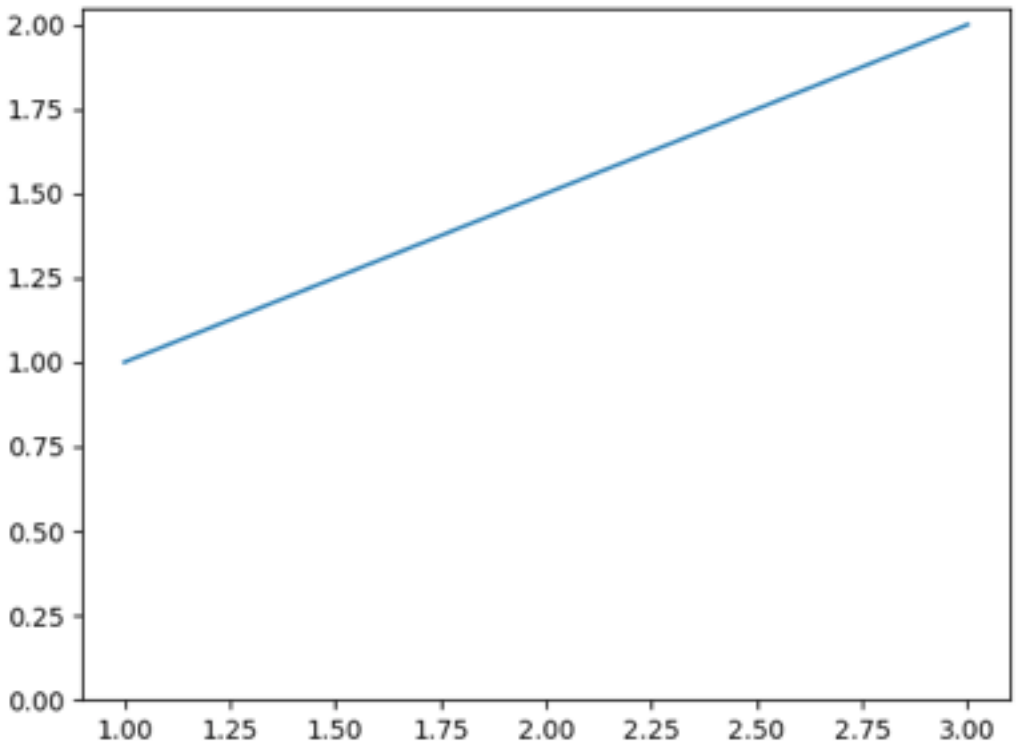
The goal of creating a model is to choose parameters, or theta values, so that h(x) is close to y for the training data, x and y.

For theta0 = 1.5 and theta1= 0 , h(x) = 1.5 + 0x. 0x means no slope, and y will always be the constant 1.5



Linear Regression

For theta0 = 1 and theta1= 0.5 ,



Cost Function:

We need a function that will minimise the parameters over our dataset. One common function that is often used is mean squared error, which measure the difference between the estimator (the dataset) and the estimated value (the prediction).

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

Adjusting the equation it looks like,

$$\frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

Gradient Descent:

Gradient Descent is a general function for minimising a function, in this case the Mean Squared Error cost function. Change the theta values, or parameters, bit by bit, until we hopefully arrived a minimum.

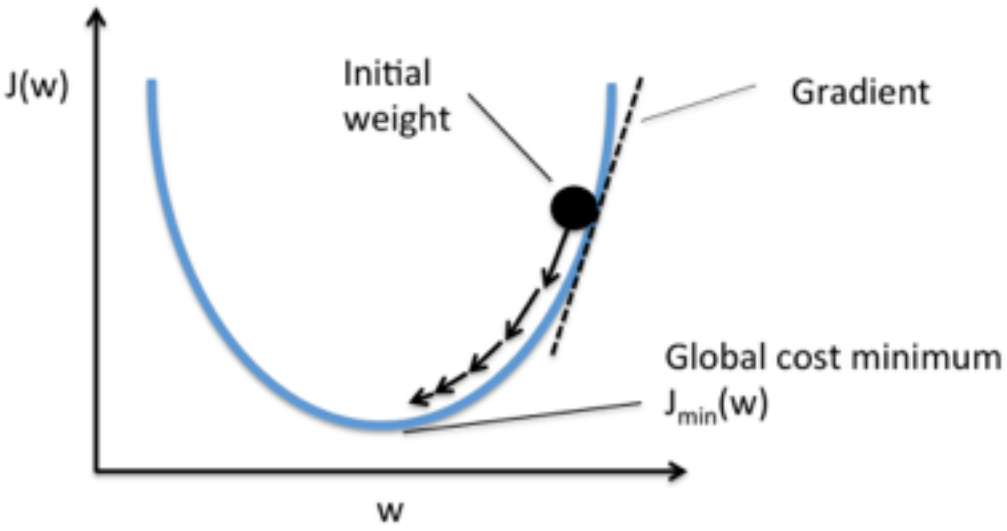
Linear Regression

Start by initialising theta0 and theta1 to any two values, say 0 for both, and go from there. The algorithm is as follows:

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \quad (\text{for } j = 0 \text{ and } j = 1)$$

where α, alpha, is the learning rate, or how quickly we want to move towards the minimum. If α is too large, however, we can overshoot. Also, we need to solve the partial derivate of the the cost function for implementing gradient descent.

$$\begin{aligned}
&\text{repeat until convergence } \{ \\
&\quad \theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right) \\
&\quad \theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right) \cdot x^{(i)} \\
&\}
\end{aligned}$$



PROBLEMS:

Given the documentation above do the following tasks:

a) Load the "data.txt" file
b) Plot the graph
c) Compute cost
d) Compute the gradient Descent
e) Plot the Contour