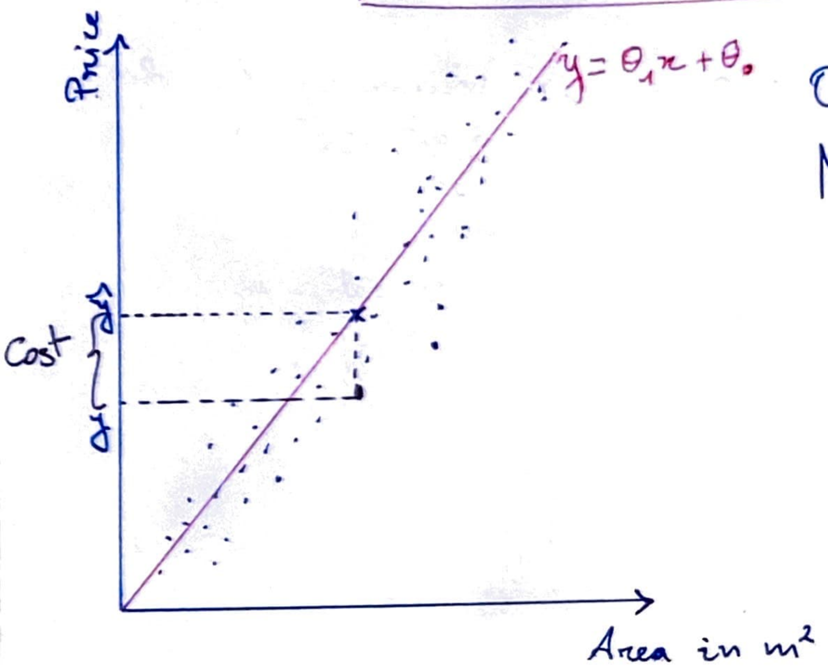
Linear Regression Analysis



Our goal is to MINIMIZE the cost function.

<u>The "cost" function:</u>

If y_i is the real value and $\hat{y}_i$ is the predicted value for the i-th datapoint, we define the cost function as follows:

$$\boxed{Cost = \frac{1}{2m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2}$$

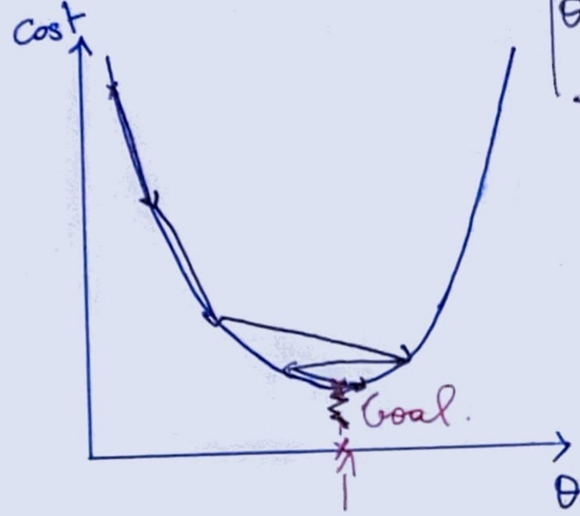
where m is the number of datapoints

For "one features" dataframes (like in the example above), we have a line of equation : $\hat{y} = \theta_1 x + \theta_0$.

In general, we will have: $\hat{y} = \theta_n x_n + \dots + \theta_1 x_1 + \theta_0$ for n features.

Our goal is, therefore, find the vector $\theta = \begin{pmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_n \end{pmatrix}$ that minimizes the cost.

The plot of the cost function, respectively to θ looks like:



$$\boxed{\theta = \theta - \alpha \frac{\partial cost}{\partial \theta}} \quad , \alpha > 0$$

This works because if the plot is decreasing, $\frac{\partial cost}{\partial \theta} < 0$ then

θ will increase,

otherwise, θ will decrease.

To find this value of theta, we use the Gradient descent algorithm.

Let X be the matrix of the features.

We will concatenate a "1" vector to X (in order to obtain θ_0).

$$\Rightarrow \quad X = \begin{bmatrix} 1 & x_{1,1} & x_{2,1} & \cdots & x_{n,1} \\ 1 & x_{1,2} & & & \vdots \\ 1 & \vdots & & & \vdots \\ \vdots & \vdots & & & \vdots \\ 1 & x_{1,m} & & & x_{n,m} \end{bmatrix} \qquad \hat{y} = \begin{pmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_m \end{pmatrix}$$

X is now a $(n+1) \times m$ matrix.

Note that if $\hat{y}_i = \theta_0 + \theta_1 x_{1,i} + \dots + \theta_n x_{n,i}$ $\forall i$, then

$$\hat{y} = X \cdot \theta \, ,$$

By calculus results, $cost = \frac{1}{2m} \sum (y_i - \hat{y}_i)^2$

$$cost = \frac{1}{2m} \operatorname{Sum}\left[\| y - \hat{y} \|_2^2 \right]$$

$$= \frac{1}{2m} \operatorname{Sum}\left(\| y - X\theta \|_2^2 \right)$$

$$\frac{\partial cost}{\partial \theta} = \frac{1}{m} X^T \cdot (Y - X\theta)$$