# House prices

1) Implement by hand equations of derivative using LaTeX (I will use similar format in LibreOffice/ODF, sorry)

$$f_x=\frac{\partial f}{\partial x}=\lim_{h\to 0}\frac{f(x+h,y)-f(x,y)}{h}$$

$$f_x=\frac{\partial f}{\partial y}=\lim_{h\to 0}\frac{f(x,y+h)-f(x,y)}{h}$$

why did I wrote them here?

2) Writing equations for housings:

$$dw\,linear:\frac{\delta}{\delta W}[W\cdot x+b]=x$$

$$dx\,linear:\frac{\delta}{\delta x}[W\cdot x+b]=W$$

$$db\,linear:\frac{\delta}{\delta b}[W\cdot x+b]=1$$

3) Cost function

We will use mean squared error cost function:

$$J(\theta_0,\theta_1)=L_{MSE}=\frac{1}{N}\cdot\sum_{i=0}^{N}(h_\theta(x_i)-y_i)^2$$

where $h_\theta(x_i)=\theta_0+\theta_1\cdot x_i$ (is our model)

We need to minimize cost function's result.

4) Gradient descent

The goal of this is to update $\theta_0,b$ and $\theta_1,W$ to minimize cost function result.

$$\theta_0:=\theta_0-\alpha\cdot\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1)$$

$$b:=b-\alpha\cdot\frac{\partial}{\partial b}J(b,W)$$

$$\theta_1:=\theta_1-\alpha\cdot\frac{\partial}{\partial\theta_1}J(\theta_0,\theta_1)$$

$$W:=W-\alpha\cdot\frac{\partial}{\partial W}J(b,W)$$

but to simultaneously update both variables we need to assign them to temporary variables first, so b doesn't affect calculation of the W.

$\alpha$ - is a learning rate, which defines how fast we are going to change b and W.

By intuition what is going to happen – partial derivative of the cost function will be a positive slope (will return a positive number) if the mse will on the right side of the local minima, as a result we will subtract a slope value multiplied by learning rate from W (or b). Otherwise we will add it. In ideal situation once MSE is 0 – we don't move anymore.

So let's try to figure out partial derivatives:

For $\theta_0$ or $b$ :

$$\frac{\partial}{\partial\theta_0}J(\theta_0,\theta_1)=\frac{\partial}{\partial\theta_0}\cdot\frac{1}{N}\cdot\sum_{i=0}^{N}(h_\theta(x_i)-y_i)^2=\frac{\partial}{\partial\theta_0}\cdot\frac{1}{N}\cdot\sum_{i=0}^{N}(\theta_0+\theta_1 x_i-y_i)^2=\frac{2}{N}\sum_{i=0}^{N}(\theta_0+\theta_1 x_i-y_i)\cdot 1$$

equals to:

$$\frac{2}{N}\sum_{i=0}^{N}(\theta_0+\theta_1 x_i-y_i)\cdot 1=\frac{2}{N}\sum_{i=0}^{N}(b+W\cdot x_i-y_i)\cdot 1$$

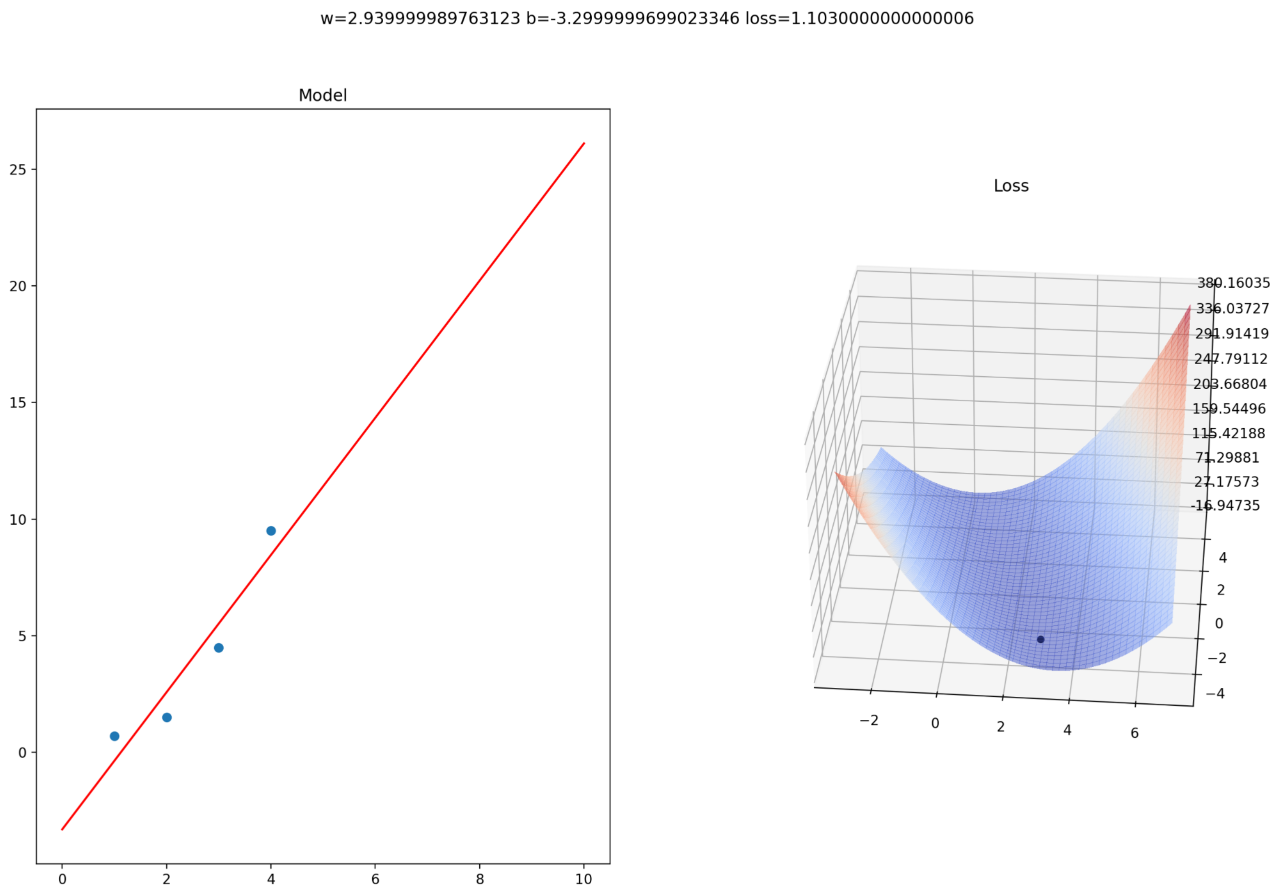
and for $\theta_1$ or $W$ :

$$\frac{\partial}{\partial\theta_1}J(\theta_0,\theta_1)=\frac{\partial}{\partial\theta_1}\cdot\frac{1}{N}\cdot\sum_{i=0}^{N}(h_\theta(x_i)-y_i)^2=\frac{\partial}{\partial\theta_1}\cdot\frac{1}{N}\cdot\sum_{i=0}^{N}(\theta_0+\theta_1 x_i-y_i)^2=\frac{2}{N}\sum_{i=0}^{N}(\theta_0+\theta_1 x_i-y_i)\cdot x_i$$

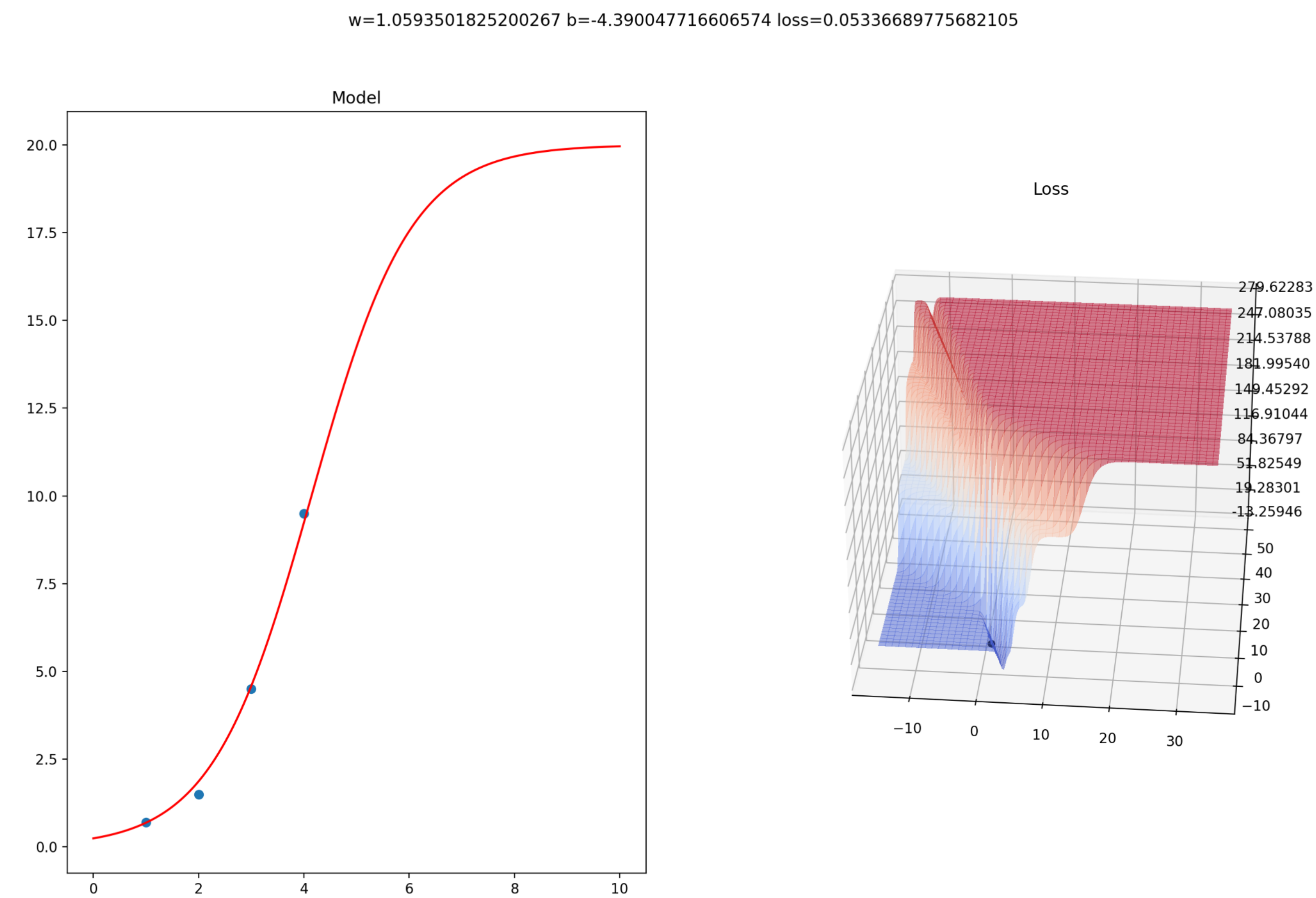
equals to:

$$\frac{2}{N}\sum_{i=0}^{N}(\theta_0+\theta_1 x_i-y_i)\cdot x_i=\frac{2}{N}\sum_{i=0}^{N}(b+W\cdot x_i-y_i)\cdot x_i$$

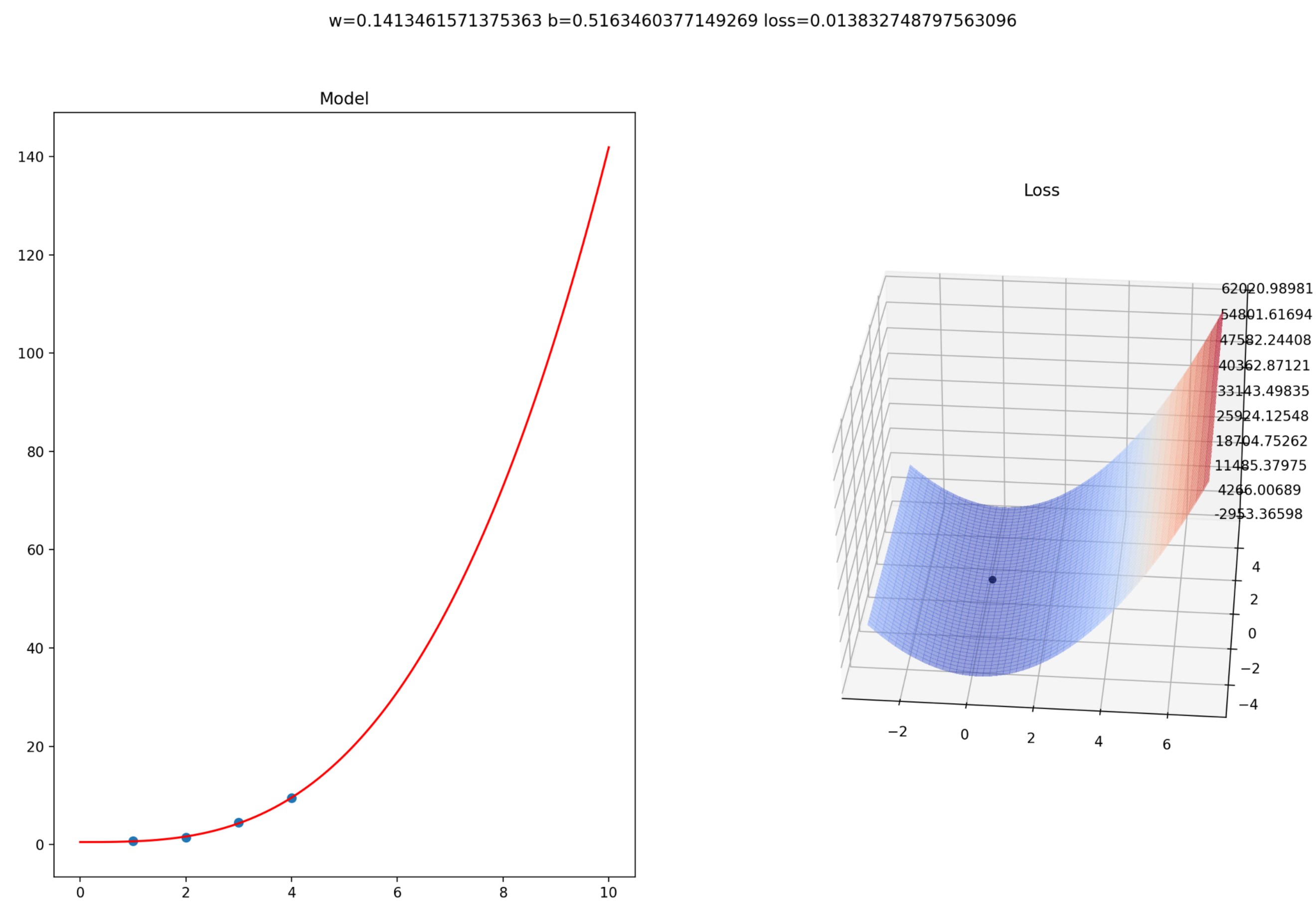
For linear model:




For sigmoid model:




I don’t find sigmoid logical here because there is no chance that 10 floor building may cost same as 100 floor. What about cubic?




That’s much better.

Sorry, I still don’t get how to write the rest. I have watched tons of Khan Academy videos on youtube regarding derivatives. I have watched lessons on Coursera regarding Machine Learning, Cost function, Gradient Descent… All the different notations of the same things drive me crazy.