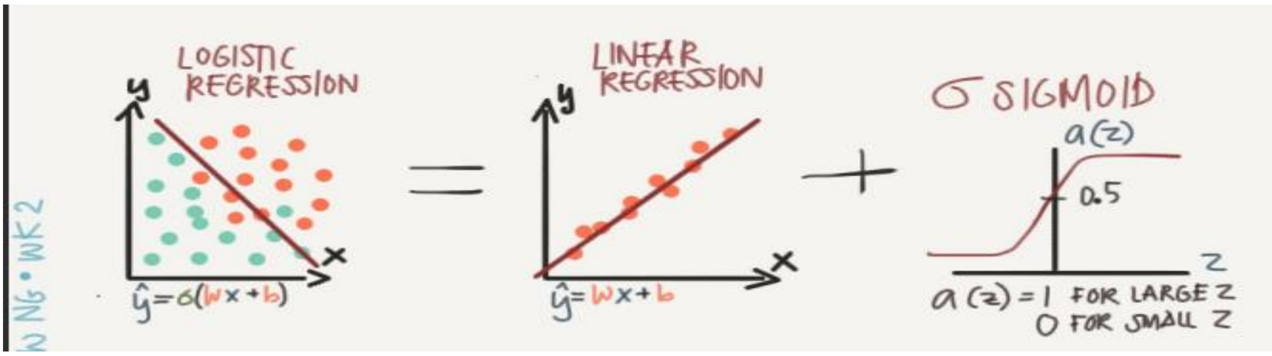
AUTHOR: SYEDA DARAQSHAN

# DAY 5:

# WEEK 2 NEURAL NETWORKS AND DEEP LEARNING:

Logistic Regression is given by

$$\hat{y} = \sigma(w^T x + b), \text{where } \sigma(z) = \frac{1}{1+e^{-z}}$$




## Loss Error Function is L(y^ ,y)

It computes error on a single training example.

Loss (error) function: $\mathcal{L}(\hat{y}, y) = \frac{1}{2}(\hat{y}-y)^2$

$$\boxed{\mathcal{L}(\hat{y}, y)} = -\left(y \log \hat{y} + (1-y)\log(1-\hat{y})\right)$$

If y =1,then Loss= -log y^ => y^ should be large

If y=0,then Loss= -(log(1-y^)) => log(1-y^) should be large. => y^ should be small.

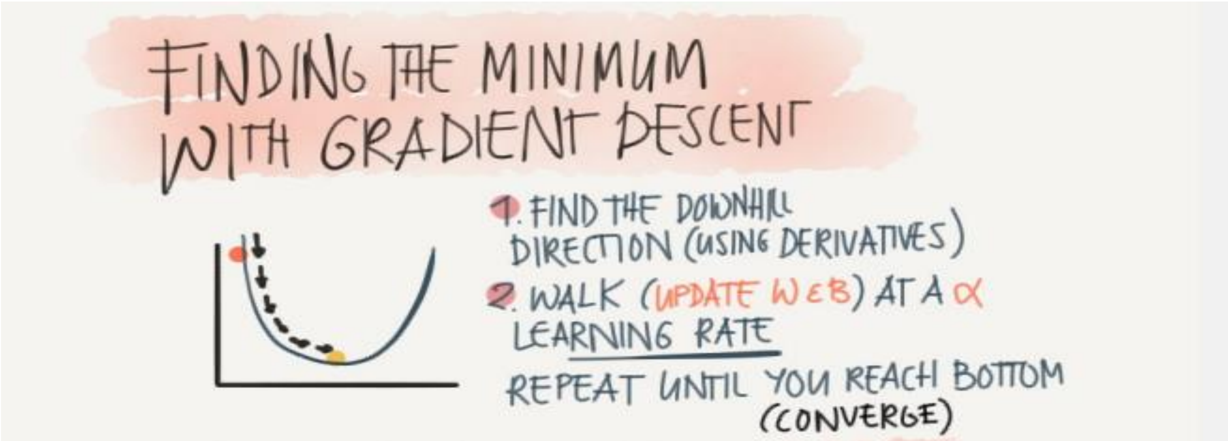
## COST FUNCTION:

It computes error on the entire training set.

It is the average of the loss functions on the entire training set.

$$\boxed{\text{Cost}} \text{ function: } J(w,b) = \frac{1}{m}\sum_{i=1}^{m} \mathcal{L}(\hat{y}^{(i)}, y^{(i)}) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log\hat{y}^{(i)} + (1-y^{(i)})\log(1-\hat{y}^{(i)})\right]$$

# GRADIENT DESCENT:




W:= w-α dJ(w)/dw

α =Learning Rate

J(w)>0 then W is large and the value decreases.

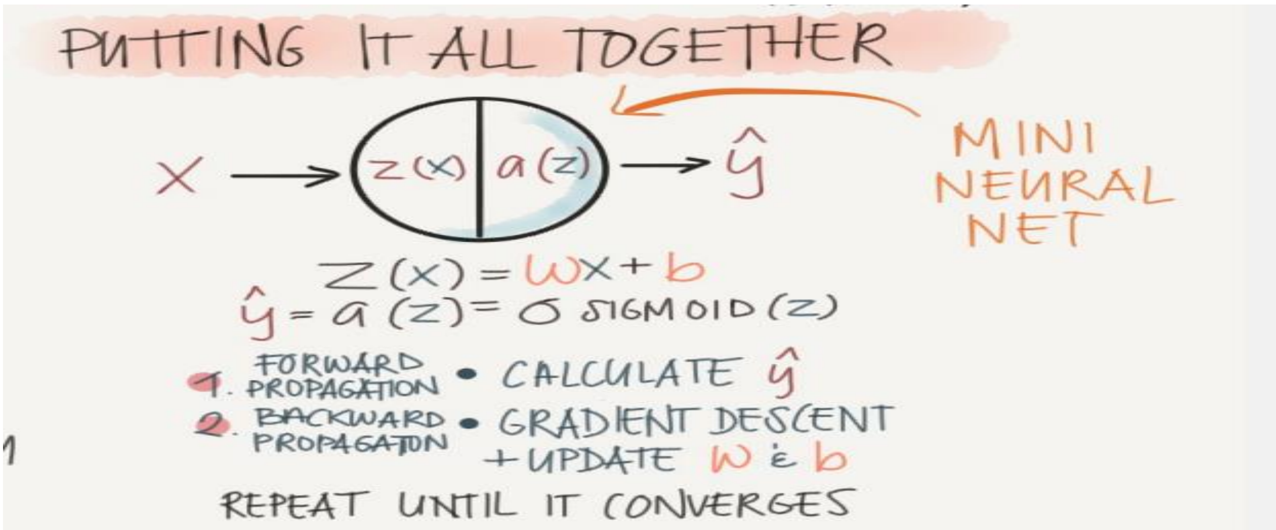
J(w)<0 then W is small and the value decreases.


w=w-α dJ(w,b)/dw

b=b-α dJ(w,b)/dw

A convex funtion has one local optima.

# <u>FORWARD AND BACKWARD PROPAGATION:</u>




One step of backward propagation on a computational graph yields derivative of final output variable.