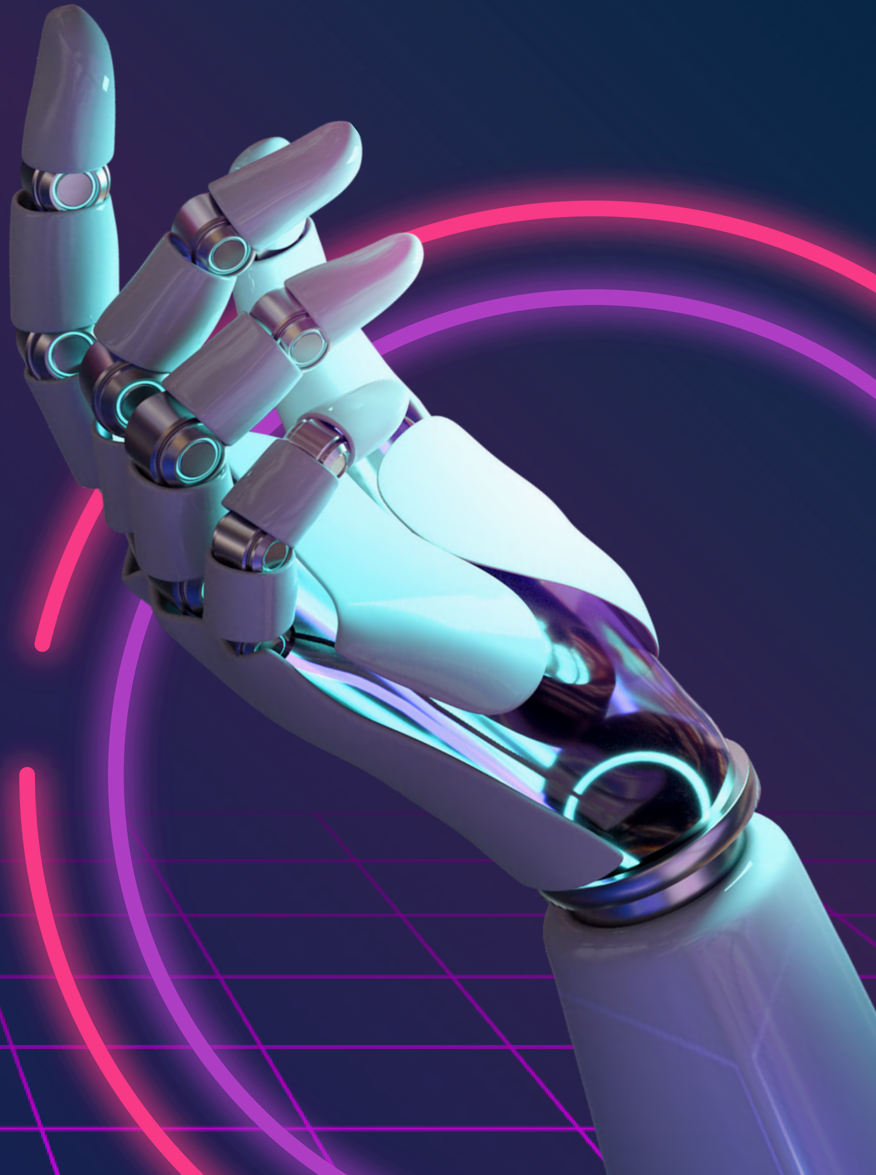
TRAINING CELL SESSION 2

LINEAR REGRESSION
LOGISTIC REGRESSION

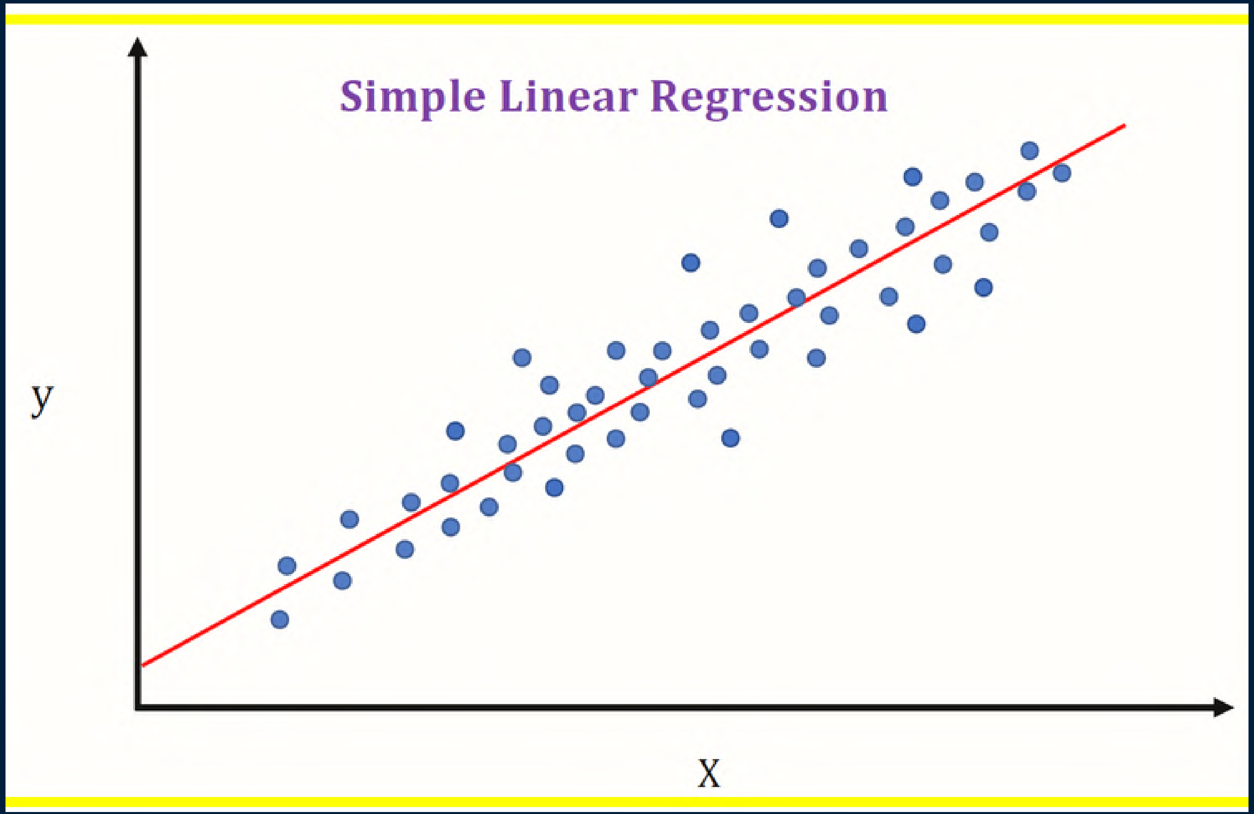
ADIL OUBAIBOU
ADNANE MAJDOUB

PLAN

1. Linear regression algorithm
2. Loss function vs Cost function
3. Gradient Descent
4. Practice Lab 1
5. Logistic Regression algorithm
6. Practice Lab 2



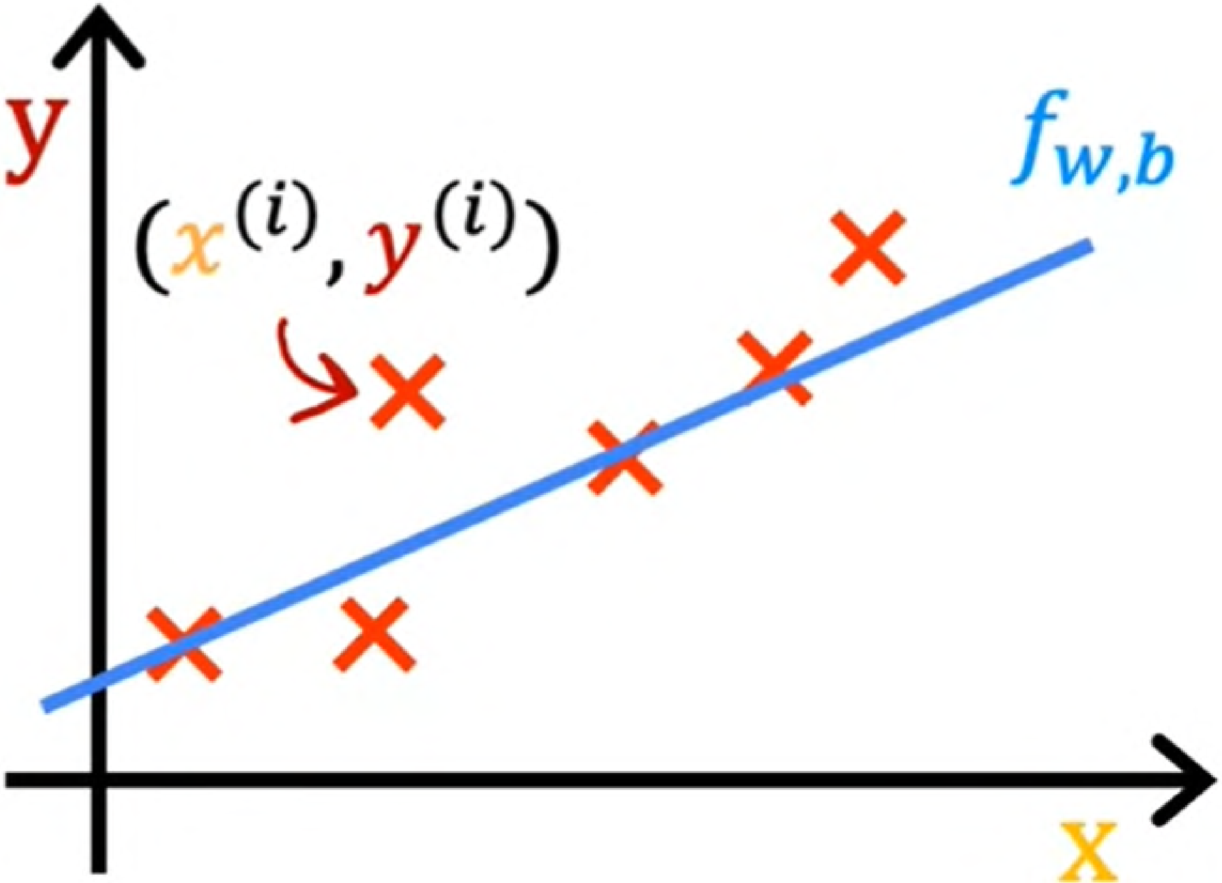
What’s our objective ?

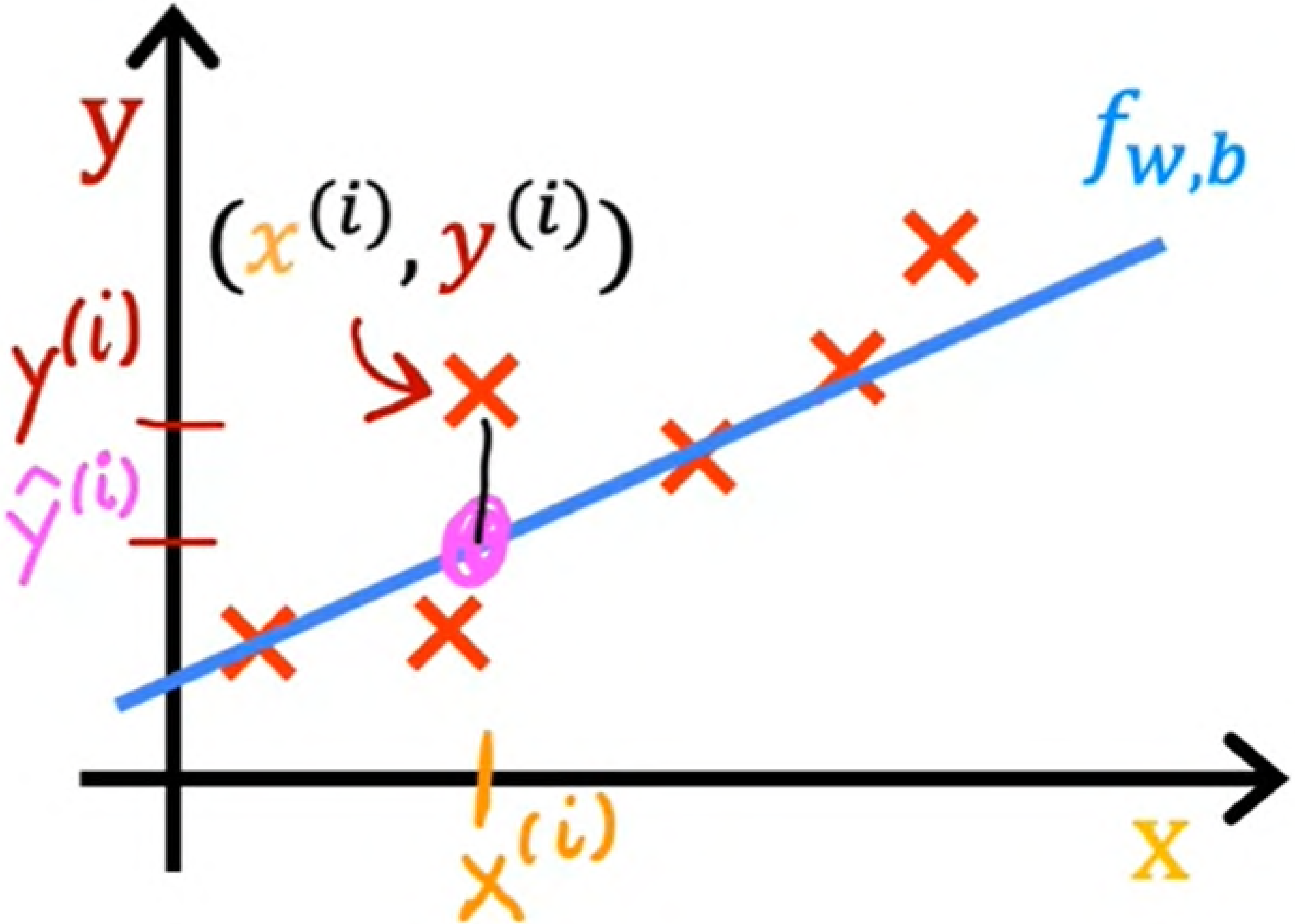


We want to fit a linear equation that’s the closer to all the data points

Simple Linear regression $f_{w,b}(x) = wx + b$

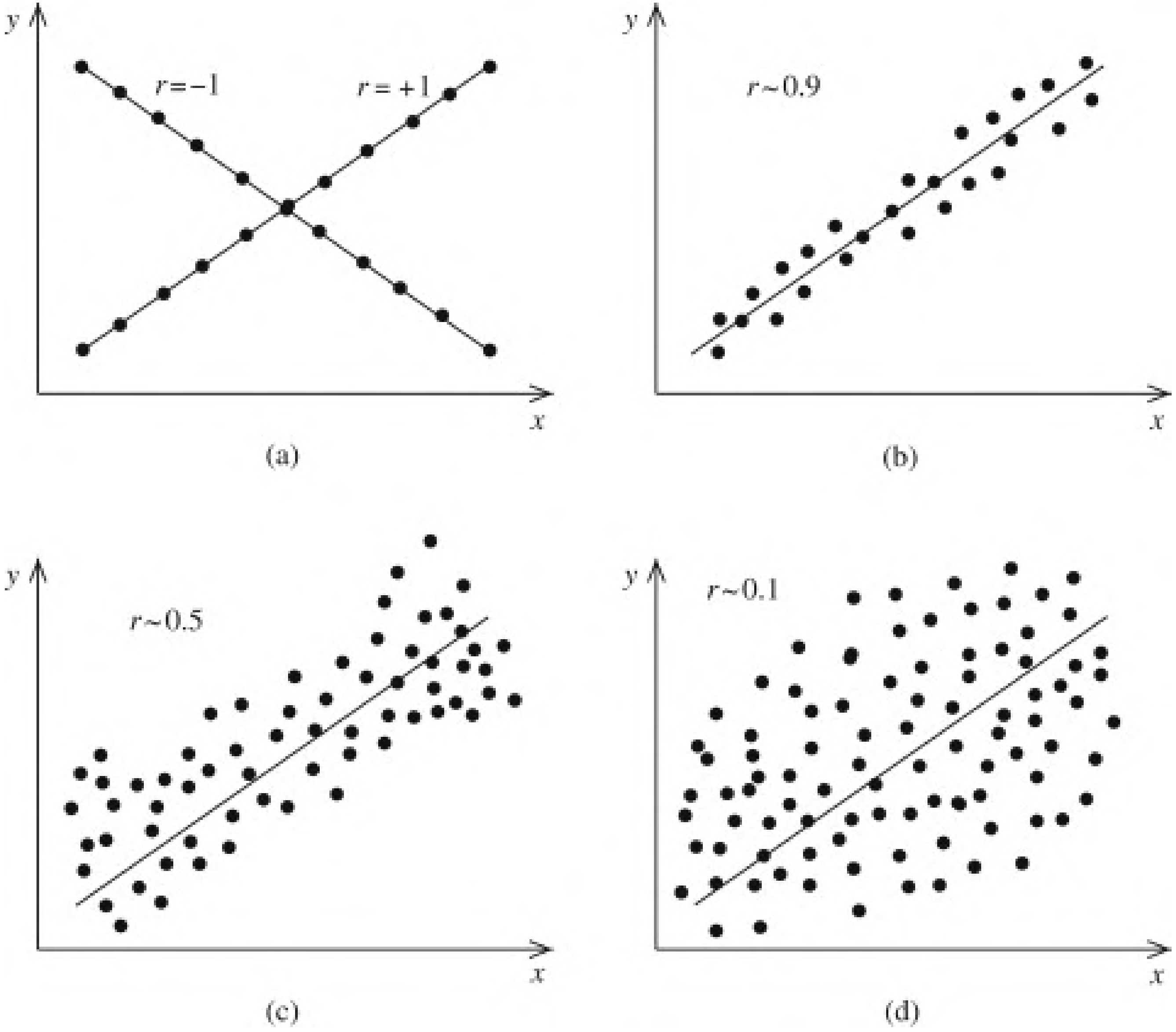
How the model find the best w , b ?


y
$(x^{(i)}, y^{(i)})$
$f_{w,b}$
x



y
$f_{w,b}$
$(x^{(i)}, y^{(i)})$
$y^{(i)}$
$\hat{y}^{(i)}$
$x^{(i)}$
x

Correlation



$$r = \frac{cov(x,y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2 \sum_{i=1}^{N}(y_i - \bar{y})^2}}$$

Loss vs Cost function

$$\text{Loss Function} = (y_i - \hat{y})^2$$

Cost Function $$J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(f_{w,b}(x^{(i)}) - y^{(i)})^2$$

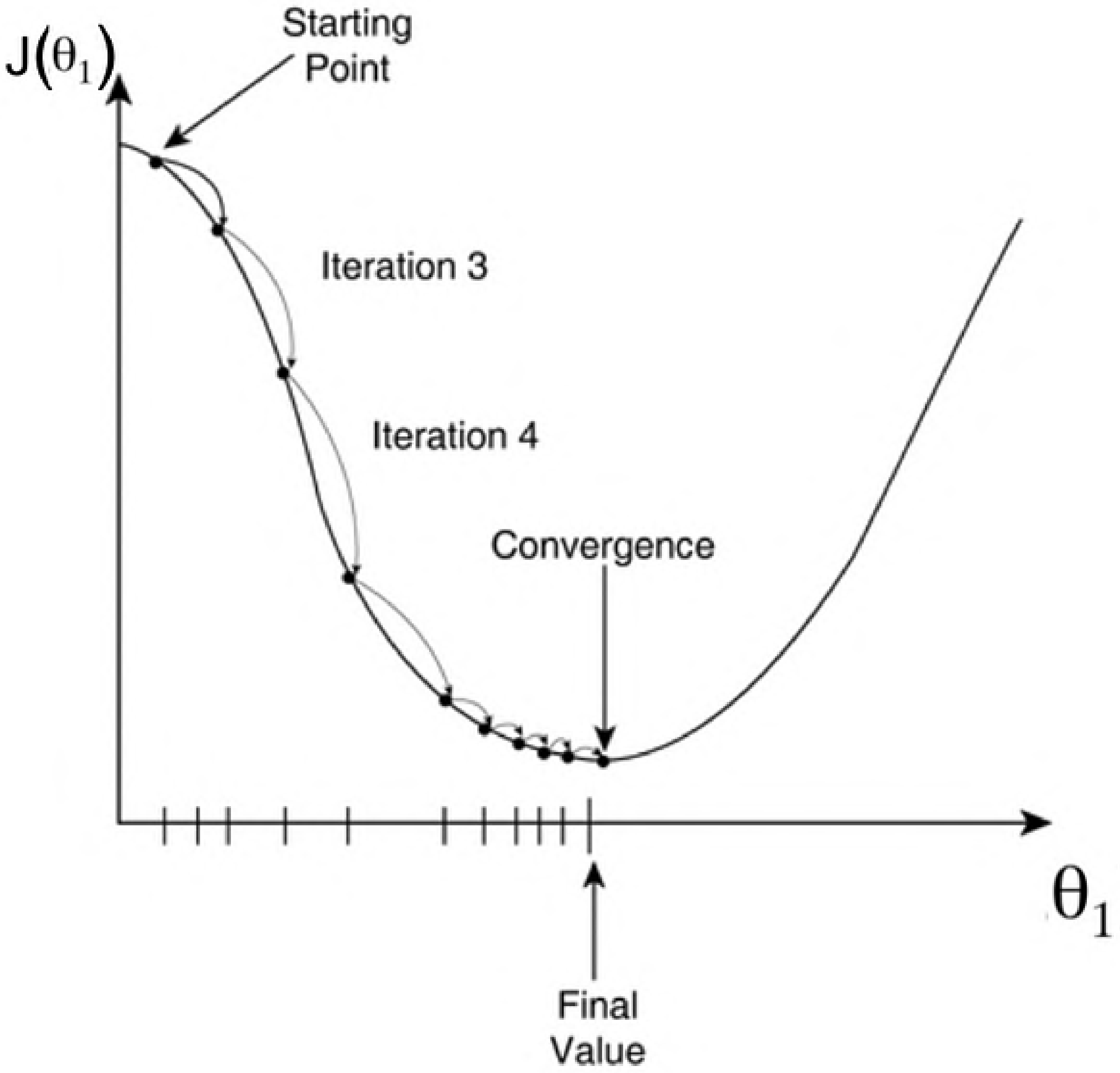
Cost Function $$J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(f_{w,b}(x^{(i)}) - y^{(i)})^2$$

Objective $$\underset{w,b}{\text{minimize}}\, J(w,b)$$

Gradient Descent

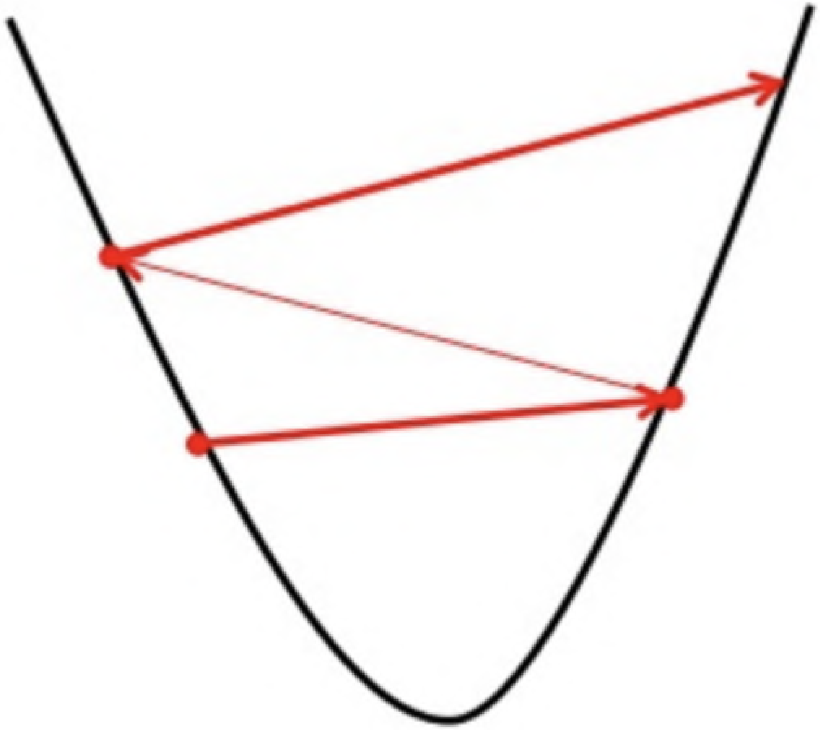
Gradient Descent formula

$$
\begin{aligned}
&\text{Repeat until converge } \{ \\
&\qquad \text{w} = \text{w} - \alpha \left[\frac{\partial \, J(w,b)}{\partial w} \right] \\
&\qquad \text{b} = \text{b} - \alpha \left[\frac{\partial \, J(w,b)}{\partial b} \right] \\
&\}
\end{aligned}
$$

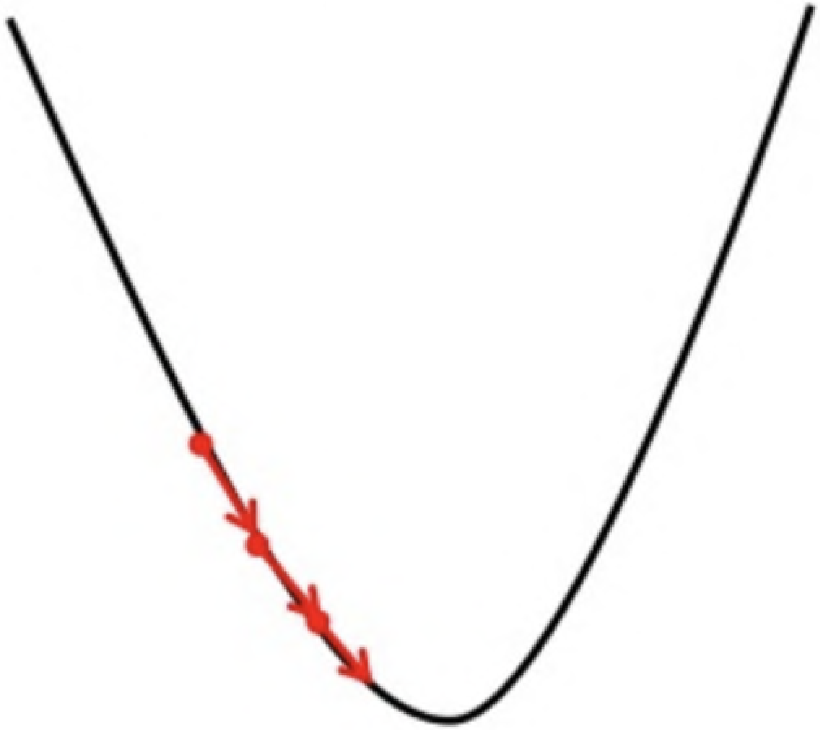


J(θ1)
Starting Point
Iteration 3
Iteration 4
Convergence
θ1
Final Value

Big learning rate



Small learning rate



Algorithm 1 Ordinary Least Squares (OLS) linear regression

Input: Training data $S = (x_i, y_i)$ such that $x_i \in \mathbf{R^m}$ for $i = 1, 2, \ldots, n$, learning rate α, tolerance δ, max iteration number N_{max}

Initialize weights vector $\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_m \end{bmatrix}$ and $iter = 0$ and we define the function :

$$f_w(x) = w^T x = \sum_{i=0}^{m} w_i x_i$$

Compute mean square error: $MSE = \frac{1}{n} \sum_{i=1}^{n} (f_w(x_i) - y_i)^2$

while $iter \leq N_{max}$ or $MSE > \delta$ **do**

Update coefficients:

$$w \leftarrow w - \alpha \cdot \frac{\partial MSE}{\partial w}$$

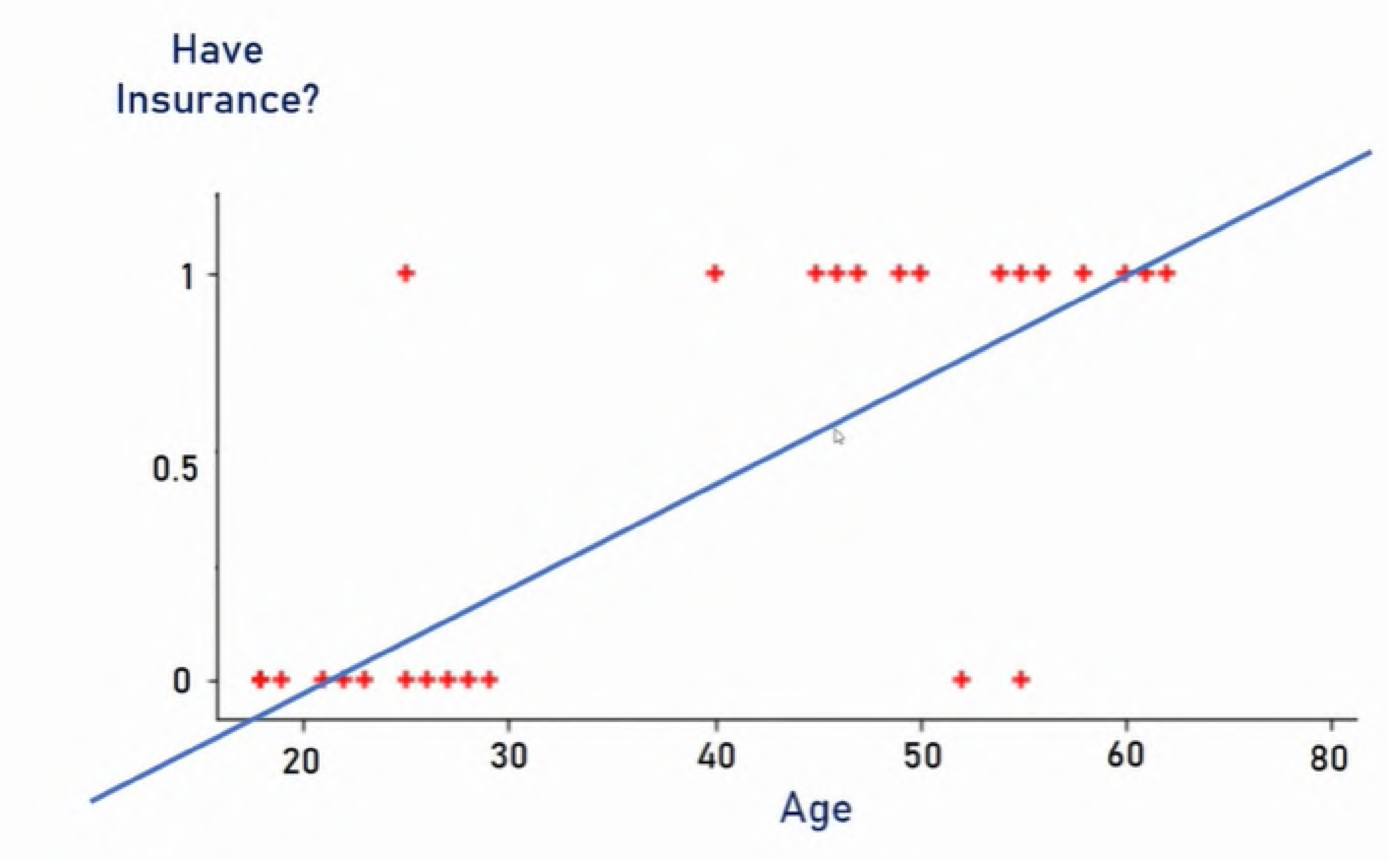
$iter \leftarrow iter + 1$

end while

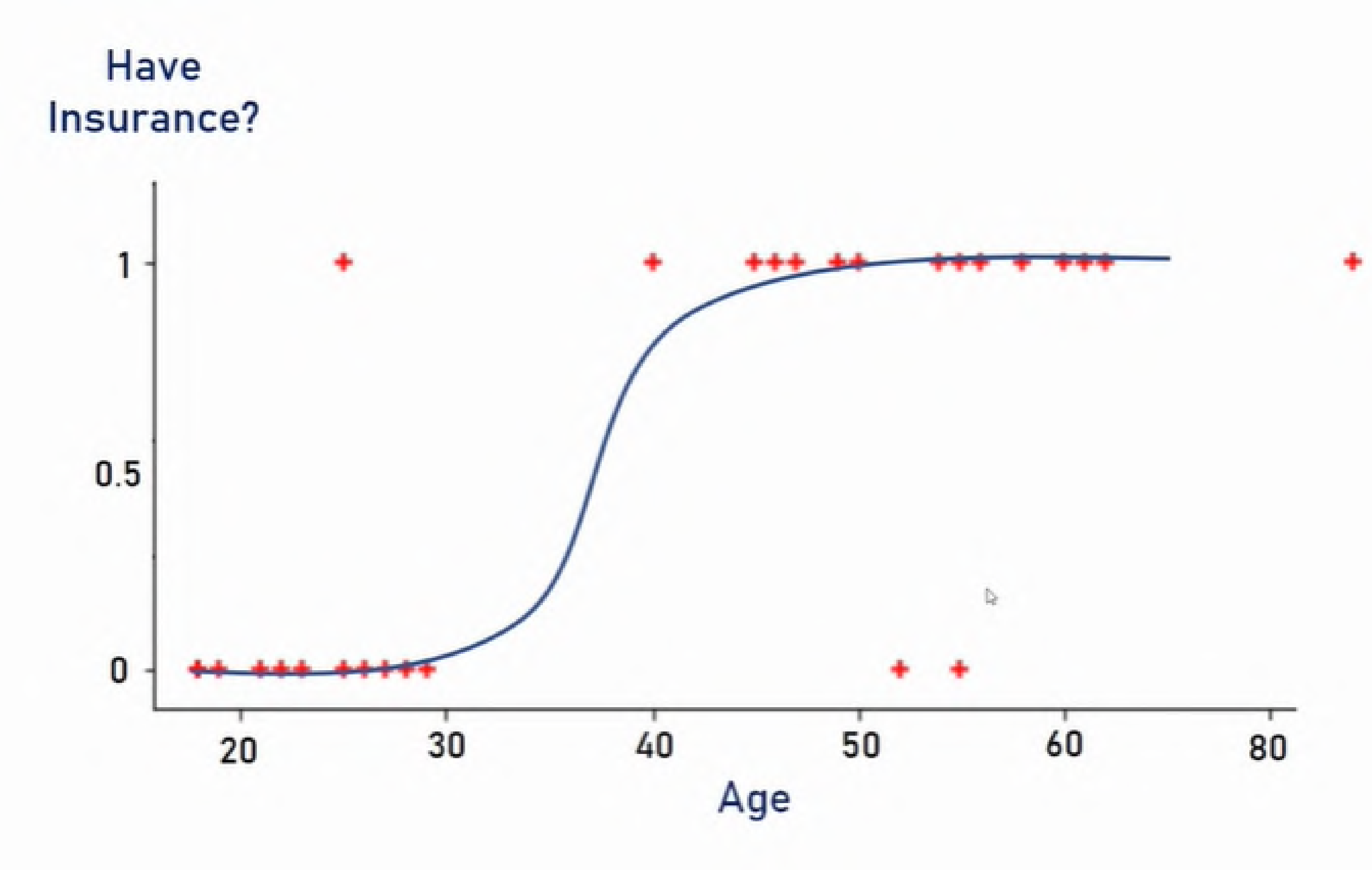
Output: weights vector w

Logistic Regression

Is linear regression good for this type of data ?



This one is better



Logistic Regression = **CLASSIFIER**

Sigmoid function

$$y = \frac{1}{1 + e^{-(m*x+b)}}$$

Cost function

$$\text{Log Loss} = \sum_{(x,y)\in D} -y\log(y') - (1-y)\log(1-y')$$

Thank you !