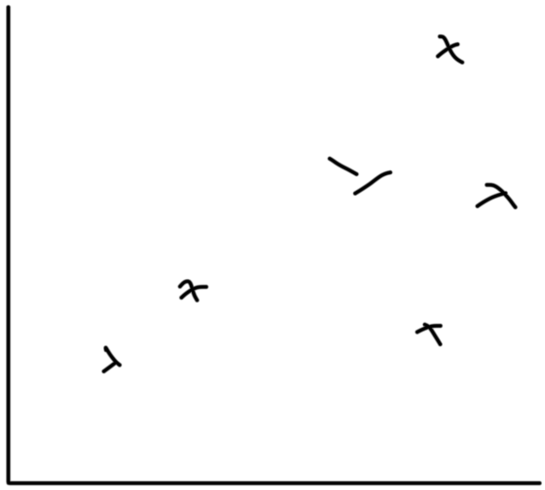
Lecture-1

#Supervised Learning

(feet²) Living area	Price ($'s)
2100	400
1600	330
2400	370
⋮	⋮



$h(x) \longrightarrow$ hypothesis

Notation:

$x^{(i)} \longrightarrow$ "input variable" (input feature)

$y^{(i)} \longrightarrow$ "label" value (target variable)

y → Output variable (Target variable)

$(x^{(i)}, y^{(i)})$ → training example

$\{(x^{(i)}, y^{(i)}) ; i = 1, \dots m\}$ → training set

x, y

$x \in \mathcal{X} ; y \in \mathcal{Y}$

$$\mathcal{X} = \mathcal{Y} = \mathbb{R}$$

h → hypothesis

$$\begin{tikzcd} & \text{Training set} \arrow[d] & \\ x \arrow[r] & \boxed{h} \arrow[r] & \text{predicted } y \end{tikzcd}$$

$$\begin{tikzcd} & \text{Regression} \\ \text{target variable} \arrow[ur, "\text{continuous}"] \arrow[dr, "\text{Discrete}"'] & \\ & \text{Classification} \end{tikzcd}$$

Linear Regression

Ex: $\overbrace{\text{Living area} \mid \text{\# bedrooms}}^{x} \mid \overbrace{\text{Price}}^{y}$

Living area	# bedrooms	Price
2100	3	400
1600	3	330
⋮	⋮	⋮

$x \longrightarrow$ 2 dimensional vector in $\mathbb{R}^2$

$x_1^{(i)} \longrightarrow$ living area of the i^{th} house

$x_2^{(i)} \longrightarrow$ no. of bedrooms " "

Lets assume that we decide to approximate y as a linear function of x.

$$h_\theta(x) = \theta_0 + \theta_1 \underline{\underline{x_1}} + \theta_2 \underline{\underline{x_2}}$$

θ_i's $\longrightarrow$ parameters (weights)

Lets just assume $\exists\, x_0 = 1$

$$h_\theta(x) = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2$$

$$= \sum_{i=0}^{n} \theta_i x_i = \theta^T x$$

$\downarrow \downarrow$

vectors

vectors

n → no. of input variable

m → no. of elem in training set

- Costfunction:

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

LMS Algorithm

goal → minimise $J(\theta)$

start with some initial guess for θ, and then move in steps st $J(\theta)$ decreases. → Gradient

$$\theta_j = \theta_j - \alpha \boxed{\frac{\partial}{\partial\theta_j} J(\theta)} \rightarrow ①$$

α → learning rate

Assumption → we have just 1 training example

Calculation of ①:

$$\frac{\partial}{\partial\theta_j} J(\theta) = \frac{\partial}{\partial\theta_j}\frac{1}{2}\left(h_\theta(x) - y\right)^2$$

$$\frac{\partial}{\partial\theta_3}\left(\cancel{\theta_1 x_1} + \cancel{\theta_2 x_2} + \theta_3 x_3 + \cancel{y}\right) = \frac{1}{2}\times 2\left(h_\theta(x)-y\right)\cdot\frac{\partial}{\partial\theta_j}\left(h_\theta(x)-y\right)$$

x_3

$$= \left(h_\theta(x)-y\right)\cdot\frac{\partial}{\partial\theta_j}\underline{\underline{\left(\sum_{i=0}^{n}\theta_i x_i - y\right)}}$$

$$= \boxed{\left(h_\theta(x)-y\right)\underline{\underline{x_j}}}$$

$$\theta_j = \theta_j + \alpha\left(\underline{\underline{y^{(i)}}} \overset{2}{-} h_\theta(x^{(i)})\right)$$

-ve ↳ LMS updaterule

Batch wise gradient descent

Stochastic grad descent

1) Batch wise grad descent:

Repeat until convergence {

$$\theta_j = \theta_j + \alpha\sum_{i=1}^{m}\left(y^{(i)} - h_\theta(x^{(i)})\right)x_j^{(i)}$$

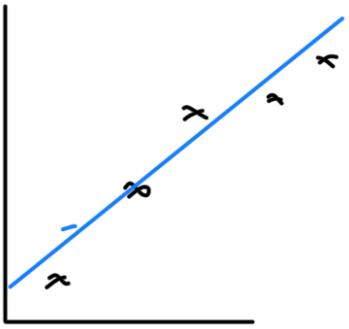
}

↳ for every j

2) Stochastic gradient descent:

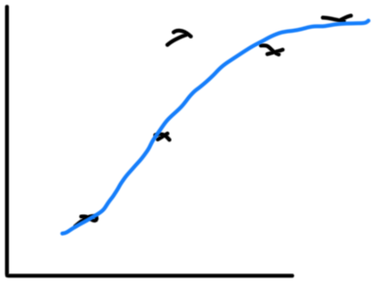
```
loop {
    for i=1 to m {
        $\theta_j = \theta_j + \alpha(y^{(i)} - h_\theta(x^{(i)})) x_j^{(i)}$
                                        (for every j)
    }
}
```

generally $m >> n$.



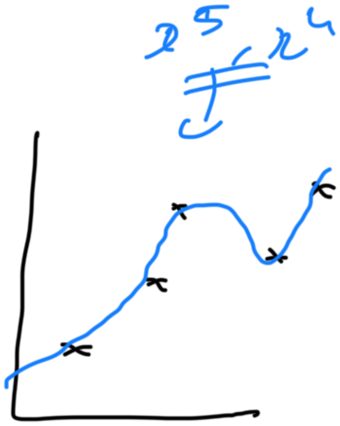
(i)

underfitting



(ii)

Best



(iii)

overfitting

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2$$

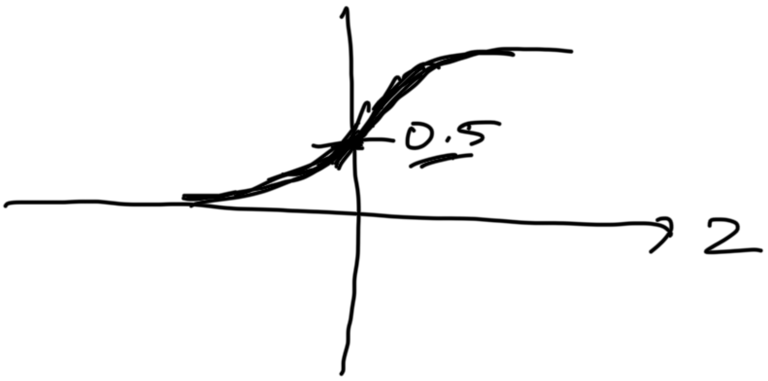
#Logistic Regression / Classification

- $y \in \{0, 1\}$

$0 \rightarrow$ -ve class

$1 \rightarrow$ +ve class

$$g(z) = \frac{1}{1+e^{-z}}$$



$$\underline{\underline{h_\theta(x)}} = g(\theta^T x) = \frac{1}{1+e^{-\underline{\underline{\theta^T x}}}}$$

- $g'(z) = \frac{\partial}{\partial z}\left(\frac{1}{1+e^{-z}}\right)$

$$= \frac{1}{(1+e^{-z})^2}\left(e^{-z}\right)$$

$$= \frac{1}{\quad}\left(1 - \frac{1}{\quad}\right)$$

$$1+e^{-z} \quad (\quad 1+e^{-z})$$

$$\boxed{g'(z) = g(z)\,(1-g(z))}$$

$$p(y=1|x;\theta) = h_\theta(x)$$

$$p(y=0|x;\theta) = 1-h_\theta(x)$$

$$p(y|x;\theta) = (h_\theta(x))^{y}\,(1-h_\theta(x))^{1-y}$$

$$L(\theta) = p(\vec{y}\,|\,X;\theta)$$

$$= \prod_{i=1}^{m} p(y^{(i)}\,|\,x^{(i)};\theta)$$

$$= \prod_{i=1}^{m} (h_\theta(x^{(i)}))^{y^{(i)}}\,(1-h_\theta(x^{(i)}))^{1-y^{(i)}}$$

$$\ell(\theta) = \log L(\theta)$$

$$= \quad {}^{m}$$

$$= \sum_{i=1} y^{(i)} \log h(x^{(i)}) + (1-y^{(i)}) \log(1-h(x^{(i)}))$$

→ maximise ⇒ $\theta = \theta + \alpha \underline{\underline{\nabla_\theta l(\theta)}}$

$$\frac{\partial}{\partial\theta_j} l(\theta) = \left(y \frac{1}{g(\theta^T x)} - \frac{(1-y)}{1-g(\theta^T x)} \right) \underline{\underline{\frac{\partial}{\partial\theta_j} g(\theta^T x)}}$$

$$= \left(\frac{y}{g(\theta^T x)} - (1-y)\frac{1}{1-g(\theta^T x)} \right) g(\theta^T x)(1-g(\theta^T x)) \left(\frac{\partial}{\partial\theta_j} \theta^T x\right)$$

$$= \left\{ y(1-g(\theta^T x)) - (1-y) g(\theta^T x) \right\} x_j$$

$$= \{ y - \underbrace{h_\theta(x)}_{\to\, g(\Sigma\theta_i x_i)} \} x_j$$

[illegible]

$$\boxed{\theta_j = \theta_j + \alpha \left(y^{(i)} - \underbrace{h_\theta(x^{(i)})}_{\text{"Non linear"}} \right) x_j^{(i)}}$$

→ Softmax Regression

→ Regularization (Linear Regression)(