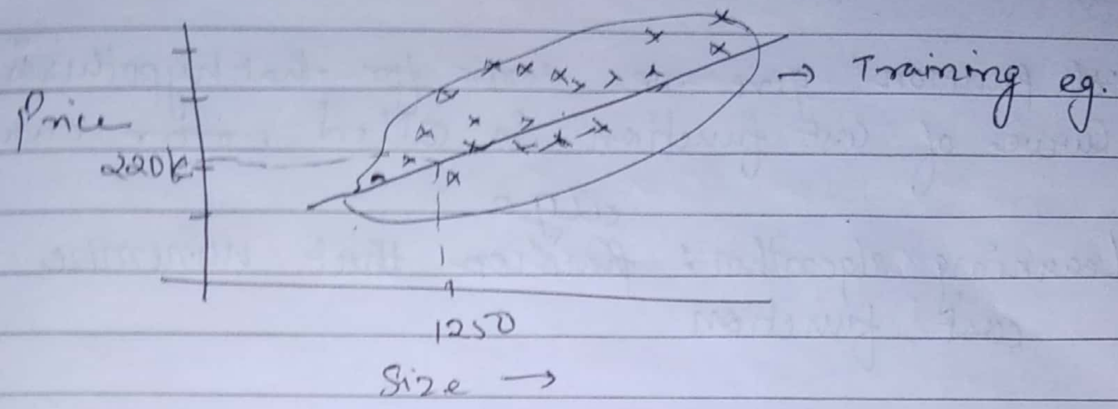
Linear Regression

→ One Variable



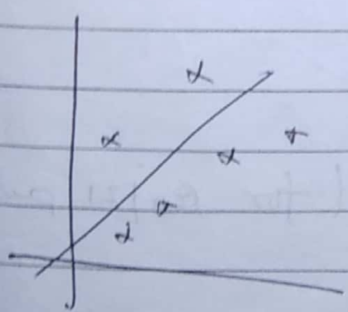
for a given size can find price for price

Training set :- use to catch relation f() in $y = f(x)$
~~Supervised~~

$x \longrightarrow h$ (hypothesis) $\rightarrow$ Estimated price

$h = \theta_0 + \theta_1 x$
univariate linear regression

Now we want best linear curve to fit



We have to minimize $\sum (h_\theta(x^i) - y^i)^2$
θ_0, θ_1
chose θ such that. h is close to y

$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2$ = Mean Square error

J_{θ_0, θ_1} ≡ Cost function hypothesis gives me curve

Cost function give me error for that hypothesis.
Curve of cost function is called error curve.

Learning algorithm: ~~function~~ algo that minimize the cost function

Training set
↓
Learning algorithm
↓
Size of house → hypothesis → Estimated price

Gradient Descent

Outline
- start with any θ_0, θ_1 value.
- keep θ_0, θ_1 to reduce $J(\theta_0, \theta_1)$ until we hopefully end up at minimum

repeat until convergence {

$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_j) \quad | \text{ for } j=1 \text{ and } j=0$$

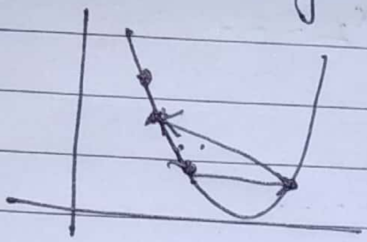
$$temp0 = \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$

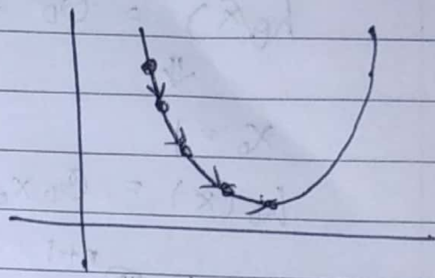
~~temp1 =~~

$$temp1 = \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$

$\theta_0 = temp0$
$\theta_1 = temp1$

As
when α is large | when α is small





Repeat until convergences
(Vector representation)

$$\theta = \theta - \alpha \cdot \frac{1}{m} \Sigma (h(x) - y)^2$$

Batch is size of training ~~set~~ ~~size~~.

When 'α' is large it may bounce around minima and finally diverse.

when 'α' is too low it may take high amount of time ~~and~~

So right ~~is~~ of 'α' is found by hit and trail method

plus as we move toward the minima Some researcher suggest to decrease the lr after some epochs.

Some suggest to keep 'α' same, as derivative decrease

Linear Regression with Multiple feature

Multivariate linear Regression.

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

$\Downarrow$

$x_0 = 1$

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

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \vdots \end{bmatrix} \mathbb{R}^{n+1} \qquad x = \begin{bmatrix} x_0 = 1 \\ x_1 \\ x_2 \\ \vdots \end{bmatrix}$$

$$h_\theta x = \theta^T x$$

Gradient descent

$$J(\theta_0, \theta_1, \dots) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2$$

Gradients

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

Feature Scaling

Make sure features are on same scale

Page No.
Date

Try to make $-1 \le x_j \le 1$

Ways to do that

1) Dividing max
2) Mean normalization

$$x_i = \frac{x_i - \mu}{x_{max}}$$

This makes new mean to zero.

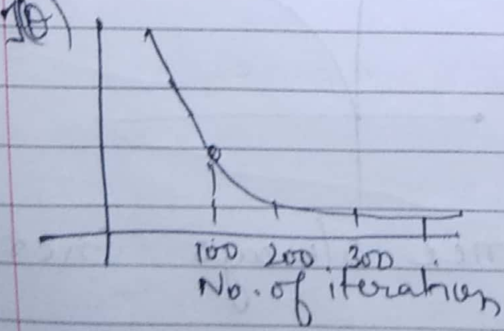
3) $$x_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}.$$

4) converting to their z-scores
as we know formula for z-scores

$$x_i = z_i = \frac{x_i - \mu}{\sigma}$$

σ → standard deviation of sample

How to choose 'α'

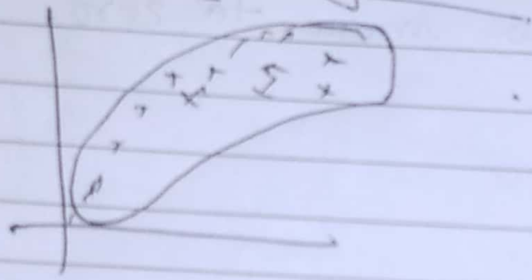


Declare convergence if $J(\theta)$ decreases by less than some ε

ε is generally taken 10^{-3}.

To choose α, try different α, $10^{-1}, 10^{-2}, 10^{-3}$ ... for small number of epochs.

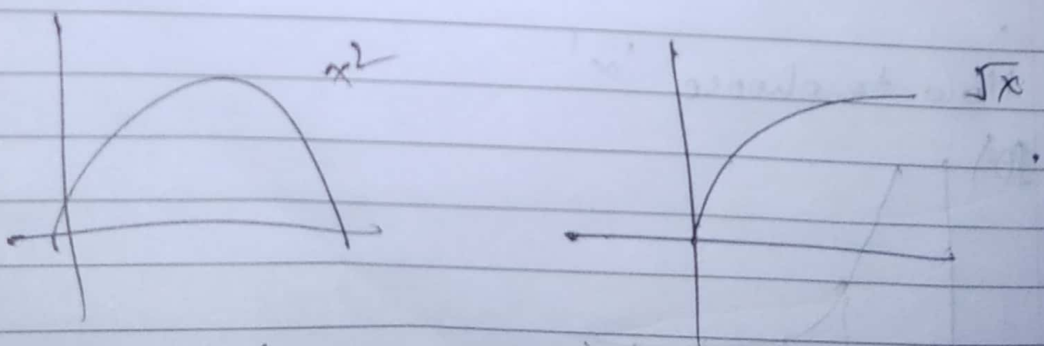
Polynomial Regression



$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$
$$= \theta_0 + \theta_1 x_1 + \theta_2 x^2 + \theta_3 x^3$$

Choice of feature

$\sqrt{x}$ instead of x^2



Common intuition: price always increse as size increse.

Page No.
Date

Loss function / Cost function

MSE/RMS

why ~~RM~~ ~~MSE/rms~~ is preferred over other

other possible cost function

$$MAE = \frac{\Sigma \, | h_\theta(x) - y^i |}{m}$$

$$\text{mean log error} = \frac{\Sigma \, \hat{y}_i \log h_\theta(x_i)}{m}$$

MSE is preffered over other losses because it penalizes object far from result more than object near to ~~lag~~ label.

MAE peanalizes all distance equally.

mean log error is genarally known as Cross entropy loss used in classification task. ~~it Generally how~~

MBE
Mean Bias error

Generally used when need to determine ~~weed~~ model have positive or negative mask

$$MBE = \frac{\Sigma_{i=1}^{n} (y_i - \hat{y}_i)}{n}$$

$$y_i = h_\theta(x_i)$$