

2 Jan, 19

Machine Learning

feeding machine with necessary data to make it learn.

There are 2 types of machine learning techniques

Supervised Learning / predictive Learning

Unsupervised learning / Descriptive Learning

- We have a trainer, teacher to sort/ classify samples to a particular type.
- no supervisor self classify objects into some patterns.
- e.g.: classification of fishes in a pond in a particular class label.

{ Ideal conditions }

NOTE:-

Parameters should be such $y = \hat{y}$ → estimate

features $\rightarrow x_1, x_2, \dots, x_{10}$ CL actual value

1

A

2

A

3

B

;

A

;

B

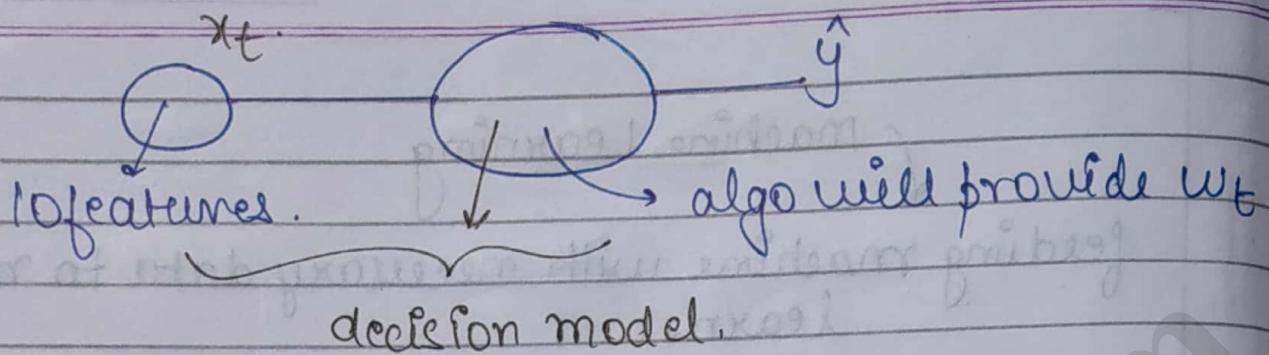
;

B

;

;

(class label)



$$w_0 + w_1 x_1 + w_2 x_2 + \dots + w_{10} x_{10} = \hat{y}$$

where $w_0 + w_1 + \dots + w_{10} = 1$

Supervised learning

Classification/
pattern recognition

Regression/
estimate.

- Class label is categorical.
- discrete values of class labels to be predicted.
- here we can differentiate objects into diff categories/ discrete values & not range of values.

- used to predict continuous scale
- continuous value of class labels.

eg:- predicting price of a property based upon some features.

eg:- Types of fishes
(A, B, C, ...)

eg :- weight (cont. variable).

- y is an integral value

- y is a real valued data value to be predicted.

NOTE :-

Classifiers to be used in supervised learning are :-

- (1) Naive Bayes classifier
- (2) linear Reg.
- (3) Logistic
- (4) Neural Net.

• Unsupervised Learning :- \downarrow (Clustering).

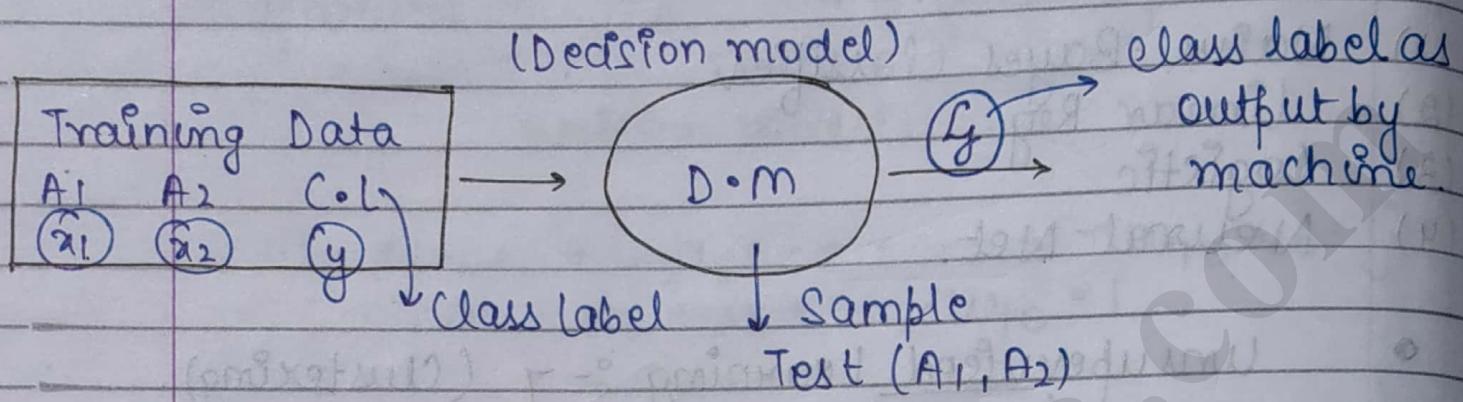
Here we aren't given labels but some grouping in some patterns useful patterns. (association)

eg :- Basket-market example / Bread butter clustering.

$$(A|B) = \frac{(A \cap B)}{B}$$

4th Jan 19.

- Naïve Bayes :- { supervised Learning }



Suppose samples = 500

out of 500 outputs, 480 are correct.

$$\therefore \text{accuracy} = \frac{480}{500} \text{ and error} = \frac{20}{500}$$

$$P(A \cap B) = \underbrace{P(A|B)}_{\downarrow} \cdot P(B) \rightarrow \text{probability of } B.$$

Joint probability.

probability of A given B
 (conditional probability)

$$= \underbrace{P(B|A)}_{\downarrow} \cdot P(A)$$

probability of B given A.

$$P(A|B) \cdot P(B) = P(B|A) \cdot P(A)$$

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

↓
 Bayes formula.

→ Suppose an object has x_1, \dots, x_n features.

$X = x_1, x_2, \dots, x_n$
and can be classified into k classes
 C_1, C_2, \dots, C_k .

$P(C_i | X) \rightarrow$ probability of class C_i for given values of X .

Using Bayes formula

$$P(X|C_i) P(C_i)$$

$= P(X) \rightarrow$ Evidence

for $i = 1, 2, 3, \dots$

Denominator is fixed.
and numerator varies.

$$P(C_i | X) \propto P(X|C_i) P(C_i)$$

↓ ↓ ↗
Posterior Likelihood Prior

Find \downarrow

$$0.6 = P(C = C_1 | \langle x_1, x_2 \rangle)$$

$$\begin{aligned} \text{Let } 0.3 &= P(C = C_2 | \langle x_1, x_2 \rangle) \\ \text{max. } 0.1 &= P(C = C_3 | \langle x_1, x_2 \rangle) \end{aligned} \quad \left. \begin{array}{l} \text{Sum = 1} \\ \{ \end{array} \right.$$

argument

$$g = \max. \arg P(C_i | X)$$

for $P(C_i) \rightarrow$ if nothing is given, probability is distributed equally among priori all C_i 's.
(Derivation not imp.)

OR

Calculate it from training data

Sample

A (400/1000)

1000

B (500/1000)

C (100/1000)

$X = x_1, x_2, \dots, x_n$

$$P(X|c_i) = P(x_1, x_2, \dots, x_n | c_i)$$

$$P(X|c_i) P(c_i) = \frac{P(x_1, x_2, \dots, x_n | c_i)}{P(c_i)}$$

$$= P(x_1, x_2, \dots, x_n | c_i)$$

breaking. every variable
in intersection.

$$P(x_1 | x_2, \dots, x_n | c_i) P(x_2, \dots, x_n | c_i)$$

$$= P(x_1 | x_2, \dots, x_n | c_i) P(x_2 | x_3, \dots, x_n | c_i) P(x_3, \dots, x_n | c_i)$$

$$\text{Let } X = x_1 \cdot x_2 \cdot x_3$$

$$= P(x_1 | x_2, x_3 | c_i) P(x_2 | x_3 | c_i) P(x_3 | c_i)$$

$$= P(x_1 | x_2, x_3 | c_i) P(x_2 | x_3 | c_i) P(x_3 | c_i) P(c_i)$$

①

NOTE: Naïve Bayes classification applies on those attributes that are independent of each other.

(Length and weight are not independent)
 ↓
 And increase in length increases weight.

$$P(A \cap B) = P(A|B) \cdot P(B)$$

Given - If A and B are independent events
 then $P(A \cap B) = P(A) \cdot P(B)$.

eqn ① can be written as :-

$$= P(x_1 | c_i) \cdot P(x_2 | c_i) P(x_3 | c_i) P(c_i)$$

(as x_1 , x_2 independent of x_3 and x_3) { for
 Naïve's Bayes
 classification? }

$$= \prod_{j=1}^n (P(x_j | c_i)) P(c_i)$$

~~2mb~~

$$V_{NB} = \operatorname{argmax}_i P(v_j) \prod_i P(a_i | v_j)$$

(Here $v_j = c_i$ and $a_i = x_j$)

Example

	x_1	x_2	x_3	x_4	C.L(y)
Day	outlook	Temp	Humidity	Wind	PlayTennis
D ₁	sunny	Hot	High	Weak	No
D ₂	Sunny	Hot	High	Strong	No
D ₃	Overcast	Hot	High	Weak	Yes
D ₄	Rain	Mild	High	Weak	Yes
D ₅	Rain	Cool	Normal	Weak	Yes
D ₆	Rain	Cool	Normal	Strong	No

D7	Overcast	Cool	Normal	Strong	Yes	4
D8	Sunny	Mild	High	Weak	No	
D9	Sunny	Cool	Normal	Weak	Yes	5
D10	Rain	Mild	Normal	Weak	Yes	6
D11	Sunny	Mild	Normal	Strong	Yes	7
D12	Overcast	Mild	High	Strong	Yes	8
D13	Overcast	Hot	Normal	Weak	Yes	9
D14	Rain	Mild	High	Strong	No	

$$V_{NB} = \arg \max_{V_j^o} \rightarrow P(V_j^o) \prod_i P(a_i | V_j^o)$$

$V_j^o \in \{ \text{Yes}, \text{No} \}$

Here from Training Data,

$$P(\text{Yes}) = \frac{9}{14} = \max.$$

$$P(\text{No}) = 5/14.$$

Given \rightarrow (Outlook = Sunny
Humidity = High
Wind = Strong
Temp = Cool)

→ Decision model will have 22 parameters irrespective of any size of data.

$$P(\text{Yes}), P(\text{No}), P(\text{Sunny} | \text{Yes}), P(\text{Sunny} | \text{No})$$

$$P(\text{Overcast} | \text{Yes}), P(\text{Overcast} | \text{No})$$

$$P(\text{Rain} | \text{Yes}), P(\text{Rain} | \text{No})$$

This way we will have

$$2 + 6 + 6 + 4 + 4 = \underline{\underline{22}}$$

CLASSTIME	Page No.
Date	/ /

$$P(\text{Yes}) = \frac{9}{14}$$

$$P(\text{No}) = \frac{5}{14}$$

$$P(\text{outlook} = \text{sunny} | \text{Yes}) = \frac{2}{9}$$

$$P(\text{sunny} | \text{No}) = \frac{3}{5} \rightarrow (14-9)$$

$$P(\text{outlook} = \text{sunny}) = \frac{5}{14} \quad P(\text{sunny} | \text{Yes}) = \frac{2}{9}$$

$$P(\text{Yes} | \text{outlook} = \text{sunny}) = \frac{\frac{2}{9}}{\frac{5}{14}} = \frac{2}{9} \times \frac{14}{5} = \frac{28}{45}$$

$$P(\text{Temp} = \text{cool} | \text{Yes}) = \frac{3}{9} = \frac{2}{9} \times \frac{9}{14} = \frac{2}{14} = \frac{1}{7}$$

$$P(\text{cool} | \text{No}) = \frac{1}{5} = \frac{2}{14}$$

$$P(\text{Wind} = \text{Strong} | \text{Yes}) = \frac{3}{9}$$

$$P(\text{Wind} | \text{No}) = \frac{3}{5}$$

$$P(\text{High} | \text{Yes}) = \frac{3}{9} \quad P(\text{High} | \text{No}) = \frac{4}{5}$$

(Yes)

$$P(\text{Yes} | X') = \frac{9}{14} \times \frac{2}{9} \times \frac{3}{9} \times \frac{3}{9} \times \frac{3}{9} = \frac{2}{378}$$

$$\text{proportion } \alpha = 0.005291$$

$$P(\text{No} | X') = \frac{5}{14} \times \frac{3}{5} \times \frac{1}{5} \times \frac{3}{5} \times \frac{4}{5} = \frac{18}{875}$$

$$\text{Proportion } \alpha = 0.00205$$

CLASSTIME	Page No.
Date	/ /

Final answer ↴

$$V_{NB} = \max \text{ of } (\text{Yes or no}) \\ = \underline{\text{No}}$$

class label = No

Probability of No and Yes

$$P(X) = \sum_{i=1}^n P(X|C_i) P(C_i)$$

$$P(C_k | X) = \frac{P(X|C_k) \cdot P(C_k)}{P(X)}$$

$$P(\text{Yes} | X) = \frac{0.00529}{0.00529 + 0.0205} = \underline{0.0205}$$

$$P(\text{No} | X) = \frac{0.0205}{0.00529 + 0.0205} = \underline{0.794}$$

4 Jan. 19

Practical

> help.start()

> a = 1:5 (1D array of length 5)
1 to 5
> a[1] = 7

> a = seq(from=1, to=10, by=2) {1 3 5 7 9}
> a = seq(1, 10, 2)

> ? seq (Help) // ? name.

> a // print
[1] 7 2 3 4 5.

→ If we don't know exact function.
?? name \Rightarrow displays everything related to this.

To know about function's declaration and definition.

> fncname
{ outputs full function}.

- For initializing 1D array / column vector.

> a = c(20, 25, 24, 23) {no need to define datatype but only one type of array}
> a
[1] 20 25 24 23

- For sequential array {1D array}

① {
 > a = seq (from=1, to=10, by 2)
 > a
 [1] 1 3 5 7 9
 or

② > a = 1:5 (1 to 5)
 > a
 [1] 1 2 3 4 5

{
 > a = 1:100
 > a
 [1] 1 2 3 4 5 ----- 100.

> b = c ("Everyone", "Likes", "machines")
> b

[1] "Everyone" "Likes" "machines"

if > b = c (2, "Everyone")
> b
[1] "2" "Everyone".

> example ("seq") // displays usage of seq.

> help (Seq) // tells about sequence & its methods.

> v = c (0, 1, 1, 2, 3, 5, 8, 13, 21, 34)

> x = v[3]

> x
[1]

// fetching subarray from index 1 to 5.

> v[1:5]
[1] 0 1 1 2 3

> $v[c(1, 5, 2, 8)]$ // extract elements of
these indexes.
↳ [1] 0 3 1 1 3

> $v[-1]$ // removes index value from
v and doesn't make change in v.
↳ [1] 1 1 2 3 5 8 13 21 34

> $v[-1:-2]$ // $v[-(1:2)]$ {remove index
↳ [1] 2 3 5 8 13 21 34 value from 1 to 2}

> $v[-c(1, 3)]$ // remove these particular
index values.
↳ [1] 3 18 13 21 24.

> $v_1 = c(1, 2, 4)$

> $v_2 = c(34, 56, 2)$

> $v_3 = c(v_1, v_2)$ // combine v_1 and v_2 .

> v_3
↳ [1] 1 2 4 34 56 2

> $c(1, 2, 3.1)$ // automatic conversion
↳ [1] 1.0 2.0 3.1 to higher DT.

> $v_1 = c(1, 2, 3)$

> $v_2 = c(1, 2, 4)$

> $v_1 == v_2$

↳ [1] TRUE TRUE FALSE

} Comparing index value of 1 with index value of 2 and then returning true.

$v_1 = 1$

$v_2 = "1" == 0 008.0 88.0 517$

$v_1 == v_2$

↳ [1] TRUE

→ Addition →

① If size is same for both arrays add corresponding index values.

② If size mismatched, then first add till same size and then start again. adding from start. ↓
and also display warning.

→ $V_1 = c(1, 2, 3)$
 $V_2 = c(1, 2, 4)$
 $V_1 + V_2$
[1] 2 4 7.

$V_1 = c(1, 2, 3)$
 $V_2 = c(1, 2, 3, 4)$
 $V_1 + V_2$ → Warning message.
[1] 2 4 6 5 → 4+1

> a] [1] 1 2 3.

> b] [1] 3 4 5

> a * b]

[1] 3 8 15
(1x3) ↓ (2x4) (5x3)

> a/b]

[1] 0.33 0.500 0.6666

} multiply &
divide in
similar
manner

$\frac{5}{3}$

TutorialsDuniya.com

Download FREE Computer Science Notes, Programs, Projects, Books PDF for any university student of BCA, MCA, B.Sc, B.Tech CSE, M.Sc, M.Tech at <https://www.tutorialsduniya.com>

- Algorithms Notes
- Artificial Intelligence
- Android Programming
- C & C++ Programming
- Combinatorial Optimization
- Computer Graphics
- Computer Networks
- Computer System Architecture
- DBMS & SQL Notes
- Data Analysis & Visualization
- Data Mining
- Data Science
- Data Structures
- Deep Learning
- Digital Image Processing
- Discrete Mathematics
- Information Security
- Internet Technologies
- Java Programming
- JavaScript & jQuery
- Machine Learning
- Microprocessor
- Operating System
- Operational Research
- PHP Notes
- Python Programming
- R Programming
- Software Engineering
- System Programming
- Theory of Computation
- Unix Network Programming
- Web Design & Development

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

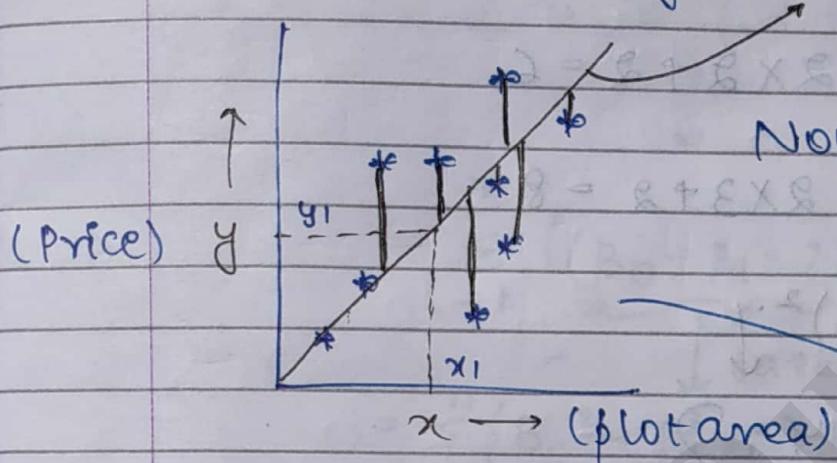
Telegram 

11 Jan, 19

* Linear Regression - ↴

Forming a line given the training data, we need to know slope and intercept

$$y = mx + c$$



Now if some x_i comes, we can estimate a price of y , denoted by y_i .

Ex-

I/P

x

0

1

2

3

O/P

y

4

7

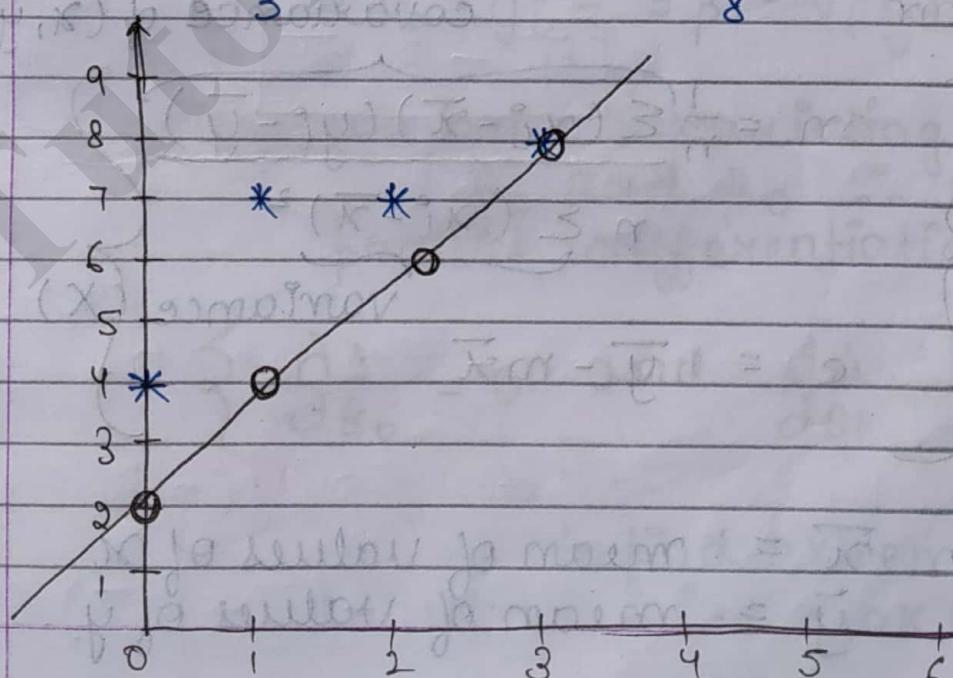
7

8

line must be such that sum of vertical distances must be least.

To predict the best suitable line

L →



Assuming $m=2$ and $c=2$

$$y = mx + c$$

(estimated val.) \hat{y}]

$$(0, 4) \quad \hat{y} = 2x_0 + 2 = 2$$

$$(1, 7) \quad \hat{y} = 2x_1 + 2 = 4$$

$$(2, 7) \quad \hat{y} = 2x_2 + 2 = 6$$

$$(3, 8) \quad \hat{y} = 2x_3 + 2 = 8$$

Now $(\hat{y} - y)^2$]

we can either
take 11 or
do $(\)^2$ of
errors.

$$\begin{matrix} 4 \\ 9 \\ 1 \\ 0 \end{matrix}$$

} adding them = 14.

$$\Delta = \sum_{i=1}^n (\hat{y}_i - y_i)^2$$

error

covariance of (x, y)

$$\left\{ m = \frac{1}{n} \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} \right\}$$

variance (x)

$$c = \bar{y} - m\bar{x}$$

where \bar{x} = mean of values of x

\bar{y} = mean of values of y .

Derivation -

$$\Delta = \sum_{i=1}^n (\hat{y}_i - y_i)^2$$

Considering $y = mx + c$

$$\text{Put } c = \beta_0$$

$$\text{and } m = \beta_1$$

$$\therefore \Delta = \sum_{i=1}^n ((\beta_0 + \beta_1 x_i) - y_i)^2$$

$$\begin{aligned} &= \sum_{i=1}^n ((\beta_0 + \beta_1 x_i)^2 + y_i^2 - 2y_i(\beta_0 + \beta_1 x_i)) \\ &\quad \downarrow (a+b)^2 \\ &= \sum_{i=1}^n (\beta_0^2 + \beta_1^2 x_i^2 + 2\beta_0 \beta_1 x_i + y_i^2 - \\ &\quad 2y_i \beta_0 - 2y_i x_i \beta_1) \end{aligned}$$

$$\begin{aligned} &= \sum_{i=1}^n \beta_0^2 + \beta_1^2 \sum_{i=1}^n x_i^2 + 2\beta_0 \beta_1 \sum_{i=1}^n x_i + \sum_{i=1}^n y_i^2 - \\ &\quad \int 2\beta_0 \sum_{i=1}^n y_i - 2\beta_1 \sum_{i=1}^n x_i y_i. \end{aligned}$$

Here we have 2 variating quantities β_0 and β_1 \therefore we perform partial differentiation

$$\left\{ \frac{\partial \Delta}{\partial \beta_0} = 0 \text{ and } \frac{\partial \Delta}{\partial \beta_1} = 0 \right\}$$

to find extreme pts.
i.e max or min pt.

Diff. wrt β_0



$$\frac{d\Delta}{d\beta_0} = 2 \sum_{i=1}^n \beta_0 + 2\beta_1 \sum x_i - 2 \sum y_i$$

(diff.
wrt β_1)

$$\frac{d\Delta}{d\beta_1} = 2 \sum_{i=1}^n \beta_1 x_i^2 + 2\beta_0 \sum x_i - 2 \sum x_i y_i$$

Now $\frac{d\Delta}{d\beta_0} = 0$

$$\sum_{i=1}^n \beta_0 + \beta_1 \sum x_i - \sum y_i = 0$$

$$n\beta_0 + \beta_1 \sum x_i - \sum y_i = 0$$

$$\beta_0 = \frac{\sum y_i - \beta_1 \sum x_i}{n}$$

$$\begin{aligned} \text{(mean of } y) &= \frac{\sum y_i}{n} - \frac{\beta_1 \sum x_i}{n} \\ &= \boxed{\bar{y} - \beta_1 \bar{x}} \end{aligned}$$

Now $\frac{d\Delta}{d\beta_1} = 0$

$$\beta_1 \sum_{i=1}^n x_i^2 + \beta_0 \sum x_i - \sum x_i y_i = 0.$$

$$\begin{aligned} \beta_1 \sum_{i=1}^n x_i^2 &= \sum x_i y_i - \beta_0 \sum x_i \\ &= \sum x_i y_i - (\bar{y} - \beta_1 \bar{x}) \sum x_i \end{aligned}$$

$$\beta_1 \sum x_i^2 + \left(\frac{\sum y_i}{n} - \beta_1 \frac{\sum x_i}{n} \right) \sum x_i - \sum x_i y_i = 0$$

$$\beta_1 \left(\sum x_i^2 - \left(\frac{\sum x_i}{n} \right)^2 \right) + \frac{\sum x_i \sum y_i}{n} - \sum x_i y_i = 0.$$

$$\beta_1 = \frac{\sum x_i y_i - \frac{\sum x_i \sum y_i}{n}}{\sum x_i^2 - \left(\frac{\sum x_i}{n} \right)^2}$$

(Slope = m)

$$= \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - \left(\sum x_i \right)^2}$$

$$\text{Now } \beta_1 = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$$

$$\frac{\sum x_i y_i - \sum x_i \bar{y} - \sum \bar{x} y_i + \sum \bar{x} \bar{y}}{\sum x_i^2 + \sum \bar{x}^2 - 2 \sum x_i \bar{x}}$$

$$\frac{\sum x_i y_i - \frac{\sum x_i \sum y_i}{n} - \frac{\sum x_i y_i}{n} + \frac{\sum x_i \sum y_i}{n}}{\sum x_i^2 + \frac{\sum x_i^2}{n^2} - 2 \frac{\sum x_i \sum x_i}{n}}$$

~~$$= \frac{\sum x_i y_i \left(1 - \frac{1}{n} \right) + \sum x_i \bar{y} \left(1 - \frac{1}{n} \right)}{\sum x_i^2 \left(1 + \frac{1}{n} \right) - 2 \left(\frac{\sum x_i}{n} \right)^2}$$~~

$$= \frac{\sum x_i y_i \left(1 - \frac{1}{n} \right)}{\sum x_i^2 - \frac{1}{n} (\sum x_i)^2}$$

$$= n \sum x_i y_i - \sum x_i \sum y_i$$

$$\overrightarrow{n \sum x_i^2 - (\sum x_i)^2}$$

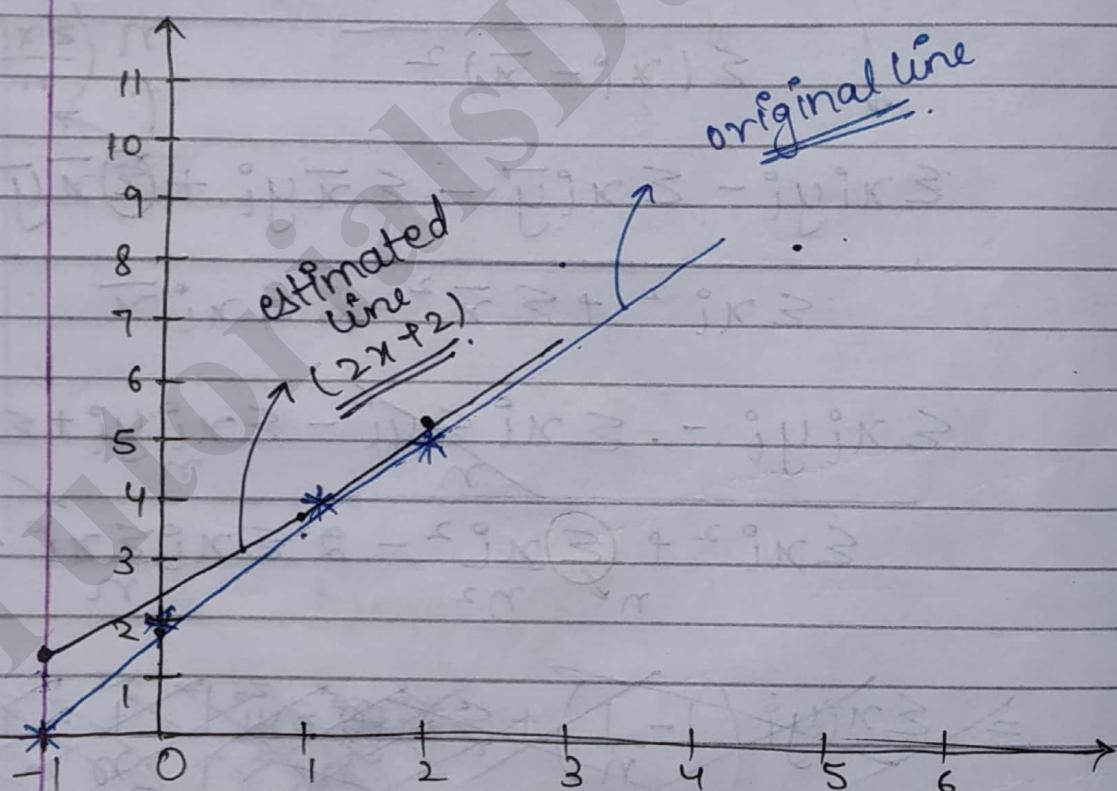
Q

(-1, 0), (0, 2), (1, 4), (2, 5)



Find the least regression line

x	y
-1	0
0	2
1	4
2	5



Choosing $m = 2$ and $c = 2$

$$\hat{y} = 2x + 2$$

x	y	$x_i - \bar{x}$	$y_i - \bar{y}$	$\sum (x_i - \bar{x})(y_i - \bar{y})$
-1	0	-3/2	-11/4	33/8
0	2	-1/2	-3/4	3/8
1	4	1/2	5/4	5/8
2	5	(3/2)	9/4	27/8

$$\bar{x} = \frac{1}{2}, \bar{y} = \frac{11}{4}$$

$$\sum (x_i - \bar{x})^2$$

$$\therefore \frac{33+3+5+27}{8} = \underline{\underline{\frac{68}{8}}}$$

$$\left. \begin{array}{c} 9/4 \\ 1/4 \\ 1/4 \\ 9/4 \end{array} \right\} \quad \frac{20}{4} = \underline{\underline{5}}.$$

$$\therefore m = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2} = \frac{68}{8 \times 5} = \underline{\underline{\frac{17}{10}}}.$$

$$c = \bar{y} - m\bar{x}$$

$$= \frac{11}{4} - \frac{17}{10} \left(\frac{1}{2} \right) = \frac{38}{20} = \underline{\underline{1.9}}$$

$$\text{Now } \hat{y} = 2x + 2$$

∴ Plotting line, we get:

x	y	?
-1	1.2	
0	3.6	
1	1.9	
2	5.3	

For predicting
value of y for
 $x = 3$

$$\hat{y} = 2(3) + 2 = \underline{\underline{8}}$$

{ Vector - 1D array }
{ Matrix - 2D array }

$a = c(2, 3, 5, 56, 7, 4)$
↓
Vector

→ $a = \text{matrix}(c(1, 2, 3, 4), 2, 2)$
↓
mxn

[1, 1] [1, 2]
[2, 1] [2, 2]

starts filling rowwise

$a = \text{matrix}(1:12, 2, 6)$

[1, 1] [1, 2] - - - [1, 6]
[2, 1] [2, 2] [2, 3] [2, 4] [2, 5] [2, 6]

→ $a = 1:12$

$\dim(a) = c(2, 2, 3)$

3 dimensions

1	3
2	4

5	7
6	8

9	11
10	12

, , 1 , , 2 , , 3

3 matrix of 2×2

* Control Loops -

help ("control") in source panel.
Syntax of all loops.

print (1:i)

↓
print values from 1 to 'i'

e.g:- for (i in 1:5) print (1:i)

[1] 1

(1 to 1)

[1] 1 2

(1 to 2)

[1] 1 2 3

(1 to 3)

[1] 1 2 3 4

(1 to 4)

[1] 1 2 3 4 5

(1 to 5)

{> if (a == 9) a = a + 2 (a = 2 initially)
> if (a == 9) a = a + 2 else a = a + 3
 ∴ a = 31.
 { and else }

• For Loop

for (i in 1:5) print (1:i).

for (i in c(2, 1, 3, 5)) print (1:i)

[1] 1 2

[1] 1

[1] 1 2 3

[1] 1 2 3 4 5

• While Loop :-

while (i <= 25) { print(i); i = i + 5 }

for 2 stmts in same lines,
use (;)

for ending infinite loop, use Esc.

func.
name ↗ repeat { if (i > 25) break else
{ print(i); i = i + 5 } }

> gcd = function (a, b) {

⊕ if (b == 0) return (a)

expects
more stmts. + else return (gcd(b, a % b))

+ }

> gcd(34, 15)

remainder
division/
modulo
division.

> gcd(12, 34)

This is made on
command
prompt \$.
Can be
directly used.

→ $\text{gcd } 2 = \text{function}(a, b)$

$\left\{ \begin{array}{l} \text{if } (b == 0) \text{ return } (a) \\ \text{else return } (\text{gcd } (b, a \% b)) \end{array} \right.$

If we write this in Source panel, then it is first necessary to run (compile) this code so that gcd2 comes into the library.

Now on Console panel → $\text{gcd } 2(34, 12)$ ✓

\downarrow
12

f1 = $\text{function}(a, b) \left\{ \begin{array}{l} \text{if } (a == 0) \text{ return } (b+2) \\ \text{else return } (b+3) \end{array} \right.$

$\left. \begin{array}{l} \\ \end{array} \right\}$

$\geq f1(0, 34) \rightarrow 36$

f factorial of a no. ↴

fact = $\text{function}(f) \left\{ \begin{array}{l} \text{if } (f == 0) \text{ return } (1) \\ \text{else return } (f * \text{fact}(f-1)) \end{array} \right.$

$\left. \begin{array}{l} \\ \end{array} \right\}$

$\rightarrow \text{fact}(5) \rightarrow \underline{120}$.

→ $\text{gcd 2} = \text{function}(a, b)$

if ($b == 0$) return (a)
else return ($\text{gcd}(b, a \% b)$)

If we write this in Source panel, then it is
first necessary to run (compile) this code so
that gcd 2 comes into the Library.

Now on Console panel → $\text{gcd 2}(34, 12)$ ✓

\downarrow
[L] 2

Q $f1 = \text{function}(a, b)$ {

if ($a == 0$) return ($b + 2$)
else return ($b + 3$)

≥ $f1(0, 34) \rightarrow 36$

Q Factorial of a no. ↴

$\text{fact} = \text{function}(f)$ {

if ($f == 0$) return (1)
else return ($f * \text{fact}(f - 1)$)

fact(5) → 120.

Q

Find whether a no. is prime

```
ch1 = "Y"  
ch2 = "N"  
prime = function(p) {  
    n = p/2  
    for (i in 2 : n) if ((p%7*i) == 0)  
        return (ch2)  
    return (ch1)}
```

}

prime(4)

[1] "N"

prime(3)

[1] "Y"

15 Jan, 19

Machine Learning

Q

$x_1 \quad x_2 \quad x_3 \quad x_4$

y

size (feet) ²	no. of bedrooms	no. of floors	Age of home (yrs)	Price (\$1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178

$$\{ m = 4 \}$$

$\xrightarrow{\text{training example}}$ $x_j^{(i)}$ → $j^{\text{th}} \text{ feature.}$

e.g. - $x_3^{(2)}$

2nd training set
and 3rd feature.

$$x^{(1)} = \begin{bmatrix} 2104 \\ 5 \\ 1 \\ 45 \\ 460 \end{bmatrix}$$

Suppose $X = \begin{bmatrix} 2104 & 5 & 1 & 45 \\ 1416 & 3 & 2 & 40 \\ 1534 & 3 & 2 & 30 \\ 852 & 2 & 1 & 36 \end{bmatrix}$

$$y = \beta_0 + \beta_1 x$$

for 1 variable

Now for multiple features that are to predicted

is

TutorialsDuniya.com

Download FREE Computer Science Notes, Programs, Projects, Books PDF for any university student of BCA, MCA, B.Sc, B.Tech CSE, M.Sc, M.Tech at <https://www.tutorialsduniya.com>

- Algorithms Notes
- Artificial Intelligence
- Android Programming
- C & C++ Programming
- Combinatorial Optimization
- Computer Graphics
- Computer Networks
- Computer System Architecture
- DBMS & SQL Notes
- Data Analysis & Visualization
- Data Mining
- Data Science
- Data Structures
- Deep Learning
- Digital Image Processing
- Discrete Mathematics
- Information Security
- Internet Technologies
- Java Programming
- JavaScript & jQuery
- Machine Learning
- Microprocessor
- Operating System
- Operational Research
- PHP Notes
- Python Programming
- R Programming
- Software Engineering
- System Programming
- Theory of Computation
- Unix Network Programming
- Web Design & Development

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

Telegram 

$$\rightarrow y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \theta_4 x_4$$

coeff. with features
(weightage of features)

4 features.

Constant
(represents a base value)

(same as $ax^2 + bx + c$)

$$y = \theta_0 + \underbrace{\frac{dy}{dx_1}}_{\theta_1} + \underbrace{\frac{dy}{dx_2}}_{\theta_2} + \dots + \underbrace{\frac{dy}{dx_4}}_{\theta_4}$$

$$\therefore \theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix} \quad \text{and } X = \begin{bmatrix} 1 & 2104 & 5 & 1 & 45 \\ 1 & 1416 & 3 & 2 & 40 \\ 1 & 1534 & 3 & 2 & 30 \\ 1 & 852 & 2 & 1 & 36 \\ 1 & & & & 14x5 \end{bmatrix}$$

$$\therefore y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix} \quad (4x1) \quad \text{assuming an } x_0 \text{ for uniformity } (\theta_0 x_0)$$

$$y = X\theta \quad (4x1) \quad (4x5) \quad (5x1)$$

$$\rightarrow ① \quad \boxed{\theta = (X^T X)^{-1} X^T Y}$$

Proof of ① :-

$$Y = X\theta$$

$$Y X^T = X X^T \theta$$

becomes a sq. matrix as $X \rightarrow 47 \times 5 \rightarrow 47 \times 47$
 $X^T \rightarrow 5 \times 47$

This is not possible as 47×5 is not

a Sq. matrix.

$$\left. \begin{array}{l} Y = X\theta \\ X^{-1}Y = X^{-1}X\theta \\ X^{-1}Y = I\theta \end{array} \right\}$$

$$\underline{X^{-1}Y = \theta}$$

Assuming, inverse can be found $(X^T X)^{-1} X^T Y = (X^T X)^{-1} X^T X \theta$

$$\therefore \theta = (X^T X)^{-1} X^T Y$$

$$X^T X \theta$$

I.

- If 2 features are related to each other, then $\Delta = 0 \therefore$ inverse can't be calculated.

→ Limitation of this method.

- The inverse operation on $X X^T$ is quite expensive as θ becomes of $O(n^3)$.

→ To overcome these problems, we have another method.

* Gradient Descent method :- (Used in neural network) (reducing).

calculating for variable

$$\hat{y} = \theta_0 + \theta_1 x$$

$$h_{\theta}(x) = \theta_0 + \theta_1 x. \quad || \text{Hypothesis function}$$

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

Average error

predicted

value.

where $\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}$ original value
and $m = \text{no. of features} = 1$.

and $J(\theta_0, \theta_1)$ rows

{error}

$\left\{ \begin{array}{c} m \end{array} \right\}$

cost function / Penalty func.

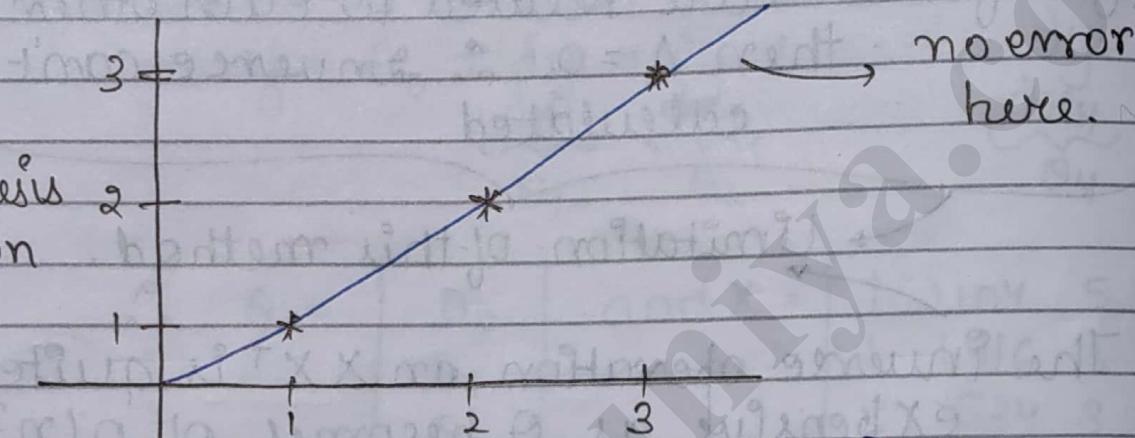
$$= \min_{\theta_0 \theta_1} \frac{1}{2m} \sum_{i=1}^m (y^{(i)} - (\theta_0 + \theta_1 x^{(i)}))^2$$

$$h_{\theta}(x) = \theta_1 x$$

{ assuming line passes through origin }

graph

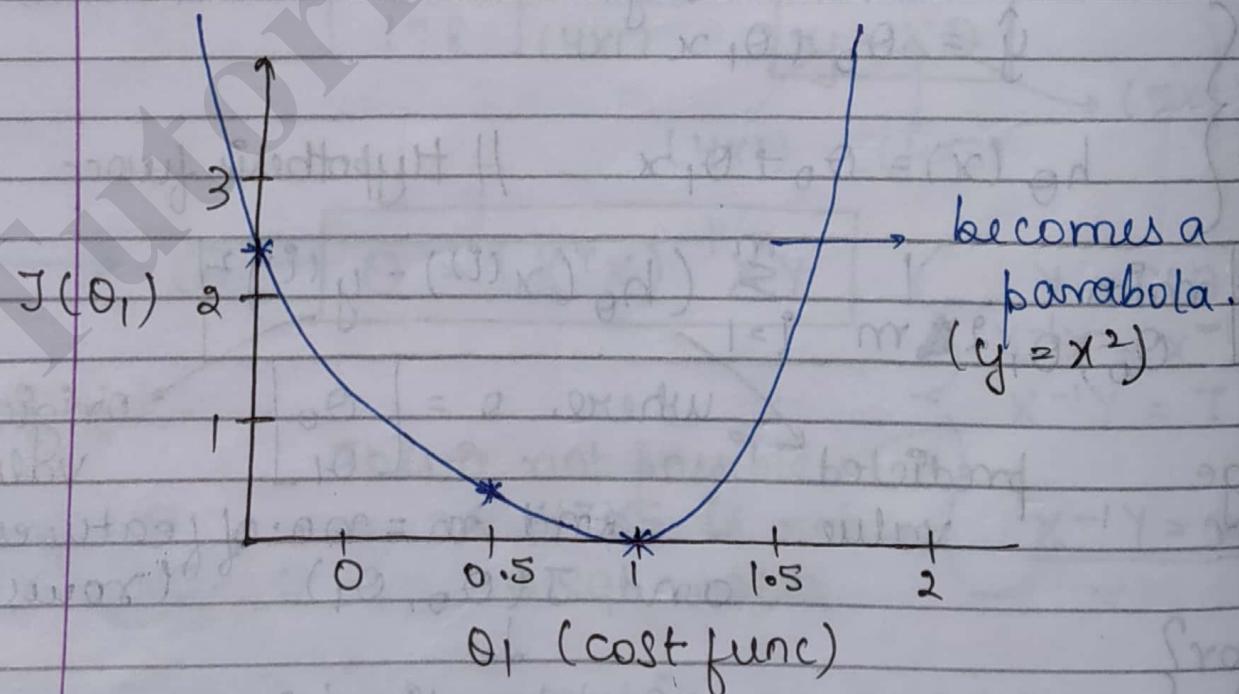
of
hypothesis
function



$$\left\{ \theta_1 = 1 = \tan 45^\circ \right\}$$

(m)

$$\therefore \min_{\theta_0 \theta_1} \frac{1}{2m} \sum_{i=1}^m (\theta_1(x^{(i)}) - y^{(i)})^2$$



(case1)

x	y
1	1
2	2
3	3

$$\theta_1 = 1$$

$$(\theta_1(x)^{(i)} - y^{(i)})^2$$

$$\left. \begin{array}{l} 1(1) - (1) = 0 \\ 1(2) - (2) = 0 \\ 1(3) - (3) = 0 \end{array} \right\}$$

$$\frac{0+0+0}{2} = 0$$

(case2)

Now if $\theta_1 = 0.5$

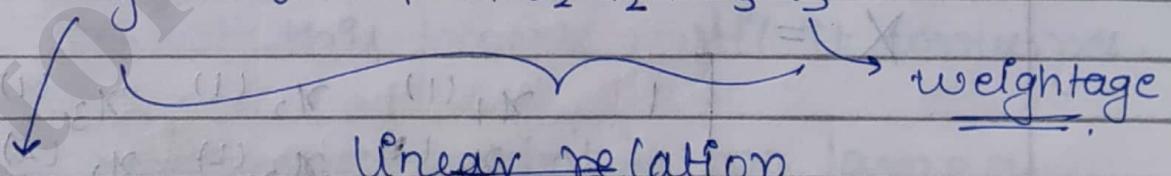
$$\begin{aligned} & (\theta_1(x)^{(i)} - y^{(i)})^2 \\ &= \frac{1}{2m} ((0.5x_1) - 1)^2 + (0.5x_2 - 2)^2 + (0.5x_3 - 3)^2 \\ &= \underline{0.76} \end{aligned}$$

For $\theta_1 = 0$ value = 2.3 (case3)

→ from graph, we need to choose min value of θ_1

23/11/19

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$

Linear relation 

As we keep on

↑ing degree of variables, it becomes a polynomial curve.

polynomial regression

$x_i \rightarrow$ where i represents feature no.

For n features, there will be

x_{n+1} weightage var.

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$

superscript
is sample
no.

subscript denotes
feature no.

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$

Given 2 samples

Given \rightarrow

x_1	x_2	x_3	y
1	2	-1	5
0	3	2	7

To find $\rightarrow \theta_0, \theta_1, \theta_2, \theta_3$.

Rewriting, $\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$ and

(4×1)

$$X = \begin{bmatrix} 1 & x_1^{(1)} & x_2^{(1)} & x_3^{(1)} \\ 0 & x_1^{(2)} & x_2^{(2)} & x_3^{(2)} \end{bmatrix} \quad (2 \times 4)$$

$$Y = \begin{bmatrix} y^{(1)} \\ y^{(2)} \end{bmatrix} \quad (2 \times 1)$$

$$\therefore Y = X\theta \quad (2 \times 4) \quad (4 \times 1)$$

{ Here inverse
of X is not
possible as X is
not a sq. matrix.

$$\therefore \theta = (X^T X)^{-1} X^T Y$$

Thus, if features increases, complexity increases.

$O(n^3)$

→ no. of features.

- To find min. in graph → use gradient descent method.

finds nearest local min.

- To find max. in graph → use gradient ascend method.

finds nearest local max.

converg- Gradient Descent method :- func. to be minimized
ence rate (new) (old)

$$\theta_j^{(t+1)} = \theta_j^{(t)} - \alpha \frac{\partial J(\theta)}{\partial \theta_j}$$

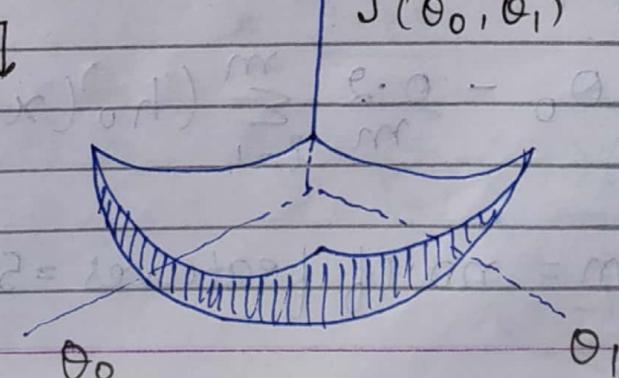
gives step size.

{ same parabola graph as drawn previously }.

NOTE: repeat this process until it converges.

$$J(\theta_0, \theta_1)$$

for 3-D ↴



$$\theta_0 = \theta_0 - \frac{\alpha}{m} \sum_{i=1}^m (\theta_0 + \theta_1 x^{(i)} - y^{(i)})$$

\downarrow $\sum_{i=1}^m$ $\sum_{i=1}^m$

$J(\theta_0, \theta_1)$.

Using chain rule

$$\textcircled{1} \quad \theta_0 = \theta_0 - \frac{\alpha}{m} \sum_{i=1}^m (h_0(x^{(i)}) - y^{(i)})$$

Similarly for θ_1 ,

$$\textcircled{2} \quad \theta_1 = \theta_1 - \frac{\alpha}{m} \left(\sum_{i=1}^m (h_0(x^{(i)}) - y^{(i)}) \right) x^{(i)}$$

Previous Year Question PPr

X	Y
0	3
1	4
2	5
3	4
4	6

Let $\alpha = 0.2$
and initial value
of $\theta_0 = 1$ and
 $\theta_1 = 1$

We show upto 2 iterations

Using formula $\textcircled{1}$.

$$\theta_0 = \theta_0 - \frac{0.2}{5} \sum_{i=1}^5 (h_0(x^{(i)}) - y^{(i)})$$

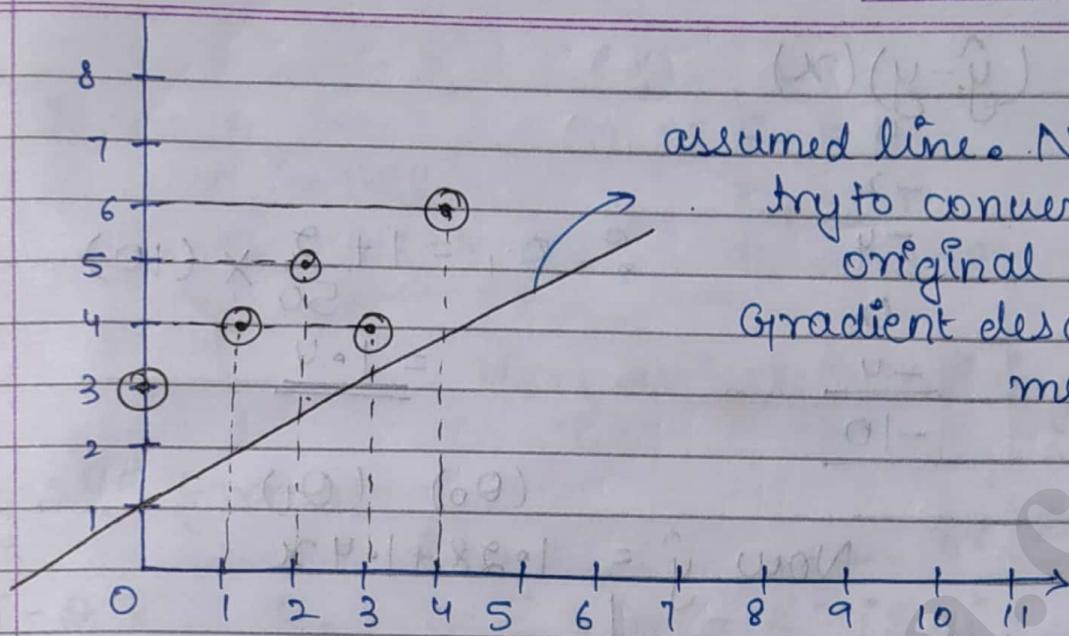
$m = \text{no. of features} = 5$

TutorialsDuniya.com

Get FREE Compiled Books, Notes, Programs, Books, Question Papers with Solution* etc of following subjects from <https://www.tutorialsduniya.com>

- C and C++
- Programming in Java
- Data Structures
- Computer Networks
- Android Programming
- PHP Programming
- JavaScript
- Java Server Pages
- Python
- Microprocessor
- Artificial Intelligence
- Machine Learning
- Computer System Architecture
- Discrete Structures
- Operating Systems
- Algorithms
- DataBase Management Systems
- Software Engineering
- Theory of Computation
- Operational Research
- System Programming
- Data Mining
- Computer Graphics
- Data Science

-
- ❖ Compiled Books: <https://www.tutorialsduniya.com/compiled-books>
 - ❖ Programs: <https://www.tutorialsduniya.com/programs>
 - ❖ Question Papers: <https://www.tutorialsduniya.com/question-papers>
 - ❖ Python Notes: <https://www.tutorialsduniya.com/python>
 - ❖ Java Notes: <https://www.tutorialsduniya.com/java>
 - ❖ JavaScript Notes: <https://www.tutorialsduniya.com/javascript>
 - ❖ JSP Notes: <https://www.tutorialsduniya.com/jsp>
 - ❖ Microprocessor Notes: <https://www.tutorialsduniya.com/microprocessor>
 - ❖ OR Notes: <https://www.tutorialsduniya.com/operational-research>



assumed line. Now we try to converge to original values using Gradient descent method.

1st iteration

$$\hat{y} \rightarrow \theta_0 + \theta_1 x \\ = (1+x)$$

$$\therefore \theta_0 = \theta_0 - \frac{\partial J}{\partial \theta_0} = \theta_0 - \frac{1}{5} \sum_{i=1}^5 (\hat{y} - y) \\ \begin{array}{c|c} i & \hat{y} - y \\ \hline 1 & -2 \\ 2 & -2 \\ 3 & -2 \\ 4 & 0 \\ 5 & -1 \end{array}$$

$$\therefore \theta_0 = \frac{1}{5} \sum_{i=1}^5 (-2) = \frac{-10}{5} = -2 \\ = \underline{\underline{1.28}}$$

$$\text{Now } \theta_1 = \theta_1 - \frac{\partial J}{\partial \theta_1} = \theta_1 - \frac{1}{5} \sum_{i=1}^5 (-7)(x^{(i)})$$

$$\underline{\underline{(\hat{y} - y)x}}$$

$$(\hat{y} - y)(x)$$

-2

-4

0

$$\begin{array}{r} -4 \\ \hline -10 \end{array}$$

$$\therefore \theta_1 = 1 + \frac{2}{50} \times (10) \\ = \underline{\underline{1.4}}$$

(θ_0) (θ_1)

$$\text{Now } \hat{y} = 1.28 + 1.4x$$



x	y	$\hat{y} = 1.28 + 1.4x$
0	3	1.28
1	4	2.68
2	5	4.08
3	4	5.48
4	6	6.88
		20.4

2nd iteration

x	y	\hat{y}	$\hat{y} - y$	$(\hat{y} - y)x$
0	3	1.28	-1.72	0
1	4	2.68	-1.32	-1.32
2	5	4.08	-0.92	1.84
3	4	5.48	+1.48	
4	6	6.88	0.88	

$$\sum (\hat{y} - y)$$

Now $\theta_0 = 1$
 $\theta_0 = 1.28$
 $\theta_0 = 1.344$

$(0.88, \theta_1) = 1$
 $\theta_1 = 1.04$
 $\theta_1 = 1.208$

Here we see θ is ↑ing

Using 2nd formula ↓

$$\theta_0 = \bar{y} - \theta_1 \bar{x} \quad \theta_1 = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x)^2}$$

or $\frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$

x	y	$\sum x_i y_i$	$\sum x_i^2$
0	3	0	0
1	4	4	1
2	5	10	4
3	4	12	9
4	6	24	16
		50	30

\therefore Putting values
in formula.

$n = 5$

$$\sum x_i \rightarrow 10$$

$$\sum y_i \rightarrow 22$$

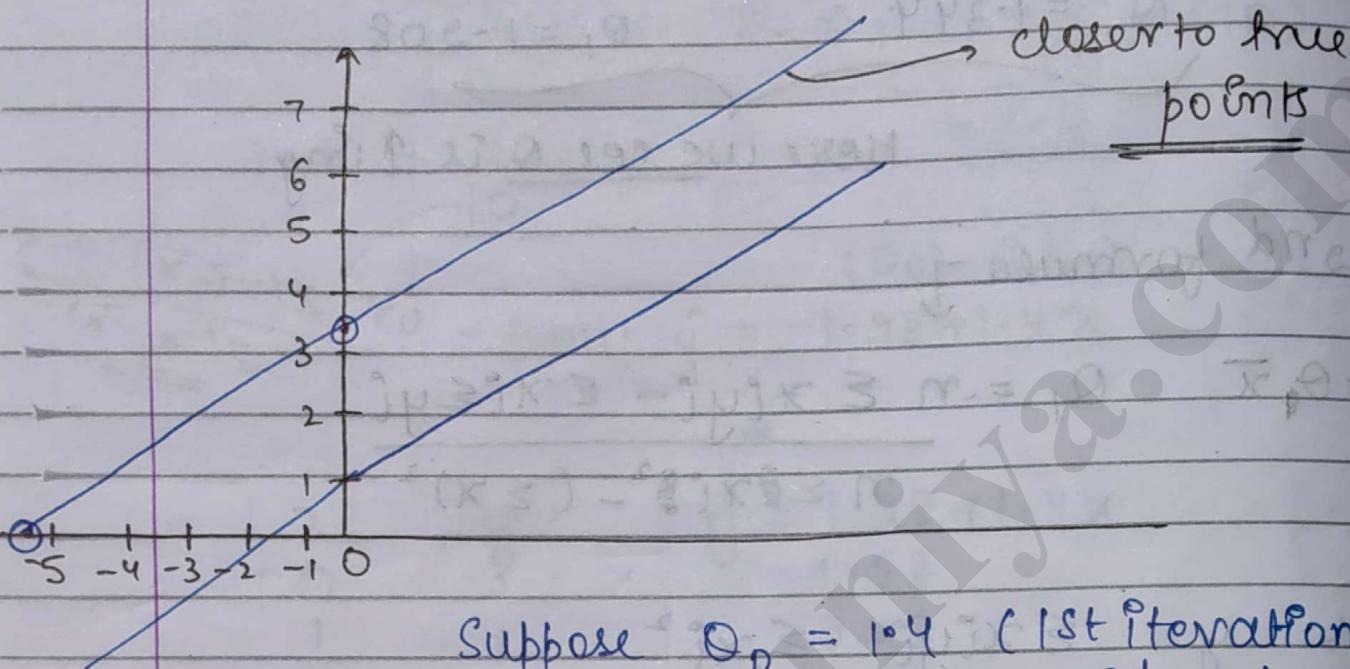
$$\text{Now } \bar{x} = \frac{10}{5} = 2$$

$$\bar{y} = \frac{22}{5}$$

$$\therefore \theta_1 = \frac{5(50) - 220}{5(30) - 100} = \underline{0.6}$$

$$\begin{aligned} \theta_0 &= \frac{22}{5} - (0.6)(2) \\ &= 4.4 - 1.2 \\ &= \underline{3.2} \end{aligned}$$

$$\therefore (0, 3.2) \\ (-5.33, 0)$$



Suppose $Q_0 = 1.4$ (1st iteration)
 $Q_0 = 5$ (2nd "
 $Q_0 = 2$ (3rd "

Here Q_0 is not in a particular dirⁿ
it keeps on moving back & forth. Thus this curve diverges.
∴ Try to reduce 'd'.

→ To plot points on graph in practical.

• plot (x, y, type = 'point')

in form of pts.

• abline (l, l) → to plot thru pt. on the line.

- Operators :-

$+$ → addition.

$-$ → subtraction

$*$ → multiplication.

$\% \cdot * \% \cdot$ → matrix multiplication

$/$ → real division.

$\% / \% /$ → integer division.

\wedge → exponential.

$\% \%$ → modulo.

$\&$ → and.

Xor → for XOR (exclusive OR).

→ comments.

- for displaying variables :-

$> a = 2$ // a gets value 2 but

$> (a = 2)$ won't get displayed

or a is assigned as well as displayed

$> a = 2$

$> a$ → 2

$> \text{print}(a) \rightarrow 2$

- * Multiple Regression -

$$h_{\theta}(x^i) = y = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3$$

eqⁿ of
multiple

regression.

n variables $\therefore (n+1)$ coeff.

Applying Gradient Descend method

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

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

multiple
linear
regression

partial derivative
wrt to $\theta_0, \theta_1, \dots, \theta_3$

Note:

$$x - \bar{x}$$

(S)

If this standard deviation, then
it is standardization.

- If S is range, then this method becomes mean normalization.

Q Normalize given data using mean normalization method

1, 2, 34, 45, 15, 40, 34.

TutorialsDuniya.com

Download FREE Computer Science Notes, Programs, Projects, Books PDF for any university student of BCA, MCA, B.Sc, B.Tech CSE, M.Sc, M.Tech at <https://www.tutorialsduniya.com>

- Algorithms Notes
- Artificial Intelligence
- Android Programming
- C & C++ Programming
- Combinatorial Optimization
- Computer Graphics
- Computer Networks
- Computer System Architecture
- DBMS & SQL Notes
- Data Analysis & Visualization
- Data Mining
- Data Science
- Data Structures
- Deep Learning
- Digital Image Processing
- Discrete Mathematics
- Information Security
- Internet Technologies
- Java Programming
- JavaScript & jQuery
- Machine Learning
- Microprocessor
- Operating System
- Operational Research
- PHP Notes
- Python Programming
- R Programming
- Software Engineering
- System Programming
- Theory of Computation
- Unix Network Programming
- Web Design & Development

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

Telegram 

1 Feb, 19

PLOTTING GRAPHS

$$y = f(x)$$

$$y' = af(bx'+c)+d \quad \text{--- (1)}$$

$$\frac{y'-d}{a} = f(bx'+c)$$

$$(x, y) \mapsto (bx'+c, \frac{y'-d}{a})$$

$$x = bx'+c$$

↓

$$y = \frac{y'-d}{a}$$

FINDING INVERSE

$$y' = ay+d$$

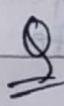
$$x' = \frac{x-c}{b}$$

eqⁿ of y' has
no change. It
is somewhat
same as (1)

$$= \frac{x}{b} - \frac{c}{b}$$

$$y' = ay+d$$

Multiply
by a scale of $\frac{1}{b}$.



Draw graph for $y = -2(3-2x)^2 + 5$.

$$\frac{y-5}{-2} = f(3-2x)$$

$$\therefore f(x) = x^2$$

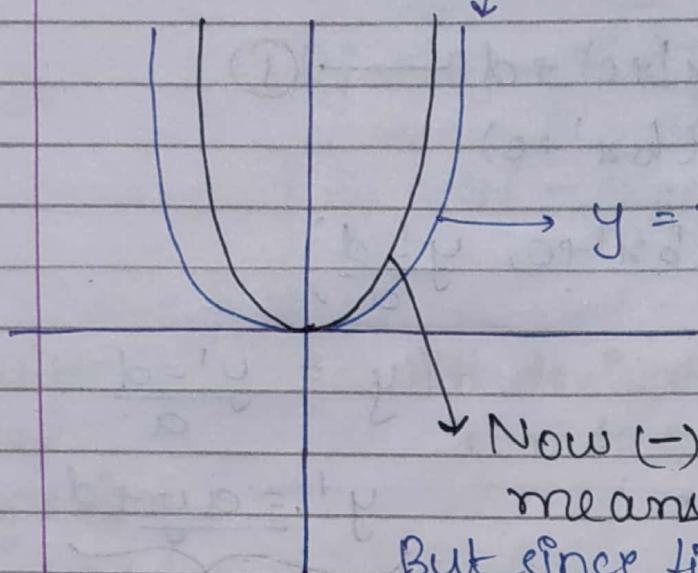
$$(x', y') \mapsto (3-2x, \frac{y-5}{-2})$$

$$\frac{x'-3}{-2}, \quad -2y'+5$$

$$-\frac{x'}{2} + 1.5$$

$$-2y'+5$$

Plotting for $\frac{-x^2}{2} + 1.5$

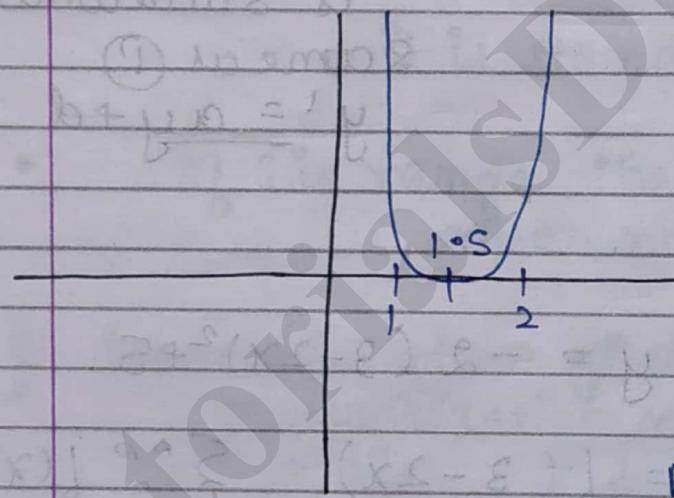


Now half the value of x
∴ reduce its width.

Now (-) sign means invert x and -x.

But since fig is symmetric, it would have no effect.

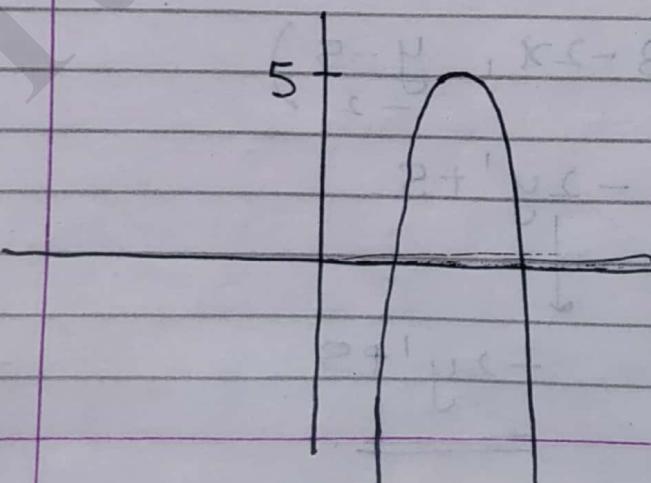
Now we perform translation of +1.5



Now for $-2y^2 + 1.5$

multiply y^2 by 2 and take into -ve

Multiplying y^2 by 2 means increasing its amplitude.



Now we need an increment of +5 in upward direction.

Logistic Regression

logistic function.

$$h_0(x) = g(\theta^T x)$$

Sigmoid func.

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

{ We can't use linear regression directly
so we use logistic regression }

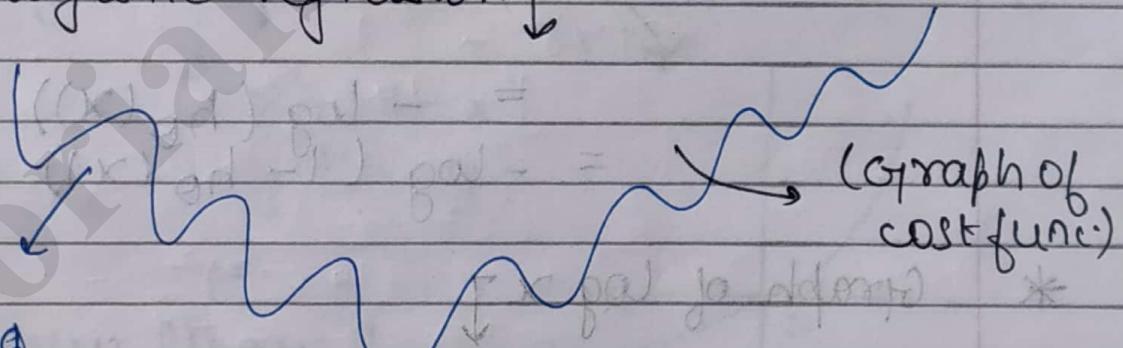
→ Cost function → (to be minimized)

$$\frac{1}{2m} \sum_{i=1}^m (h_0(x^{(i)}) - y^{(i)})^2$$

In linear regression, graph of this cost func. was a parabola where local minima = global minima using gradient descent.

For logistic regression ↴

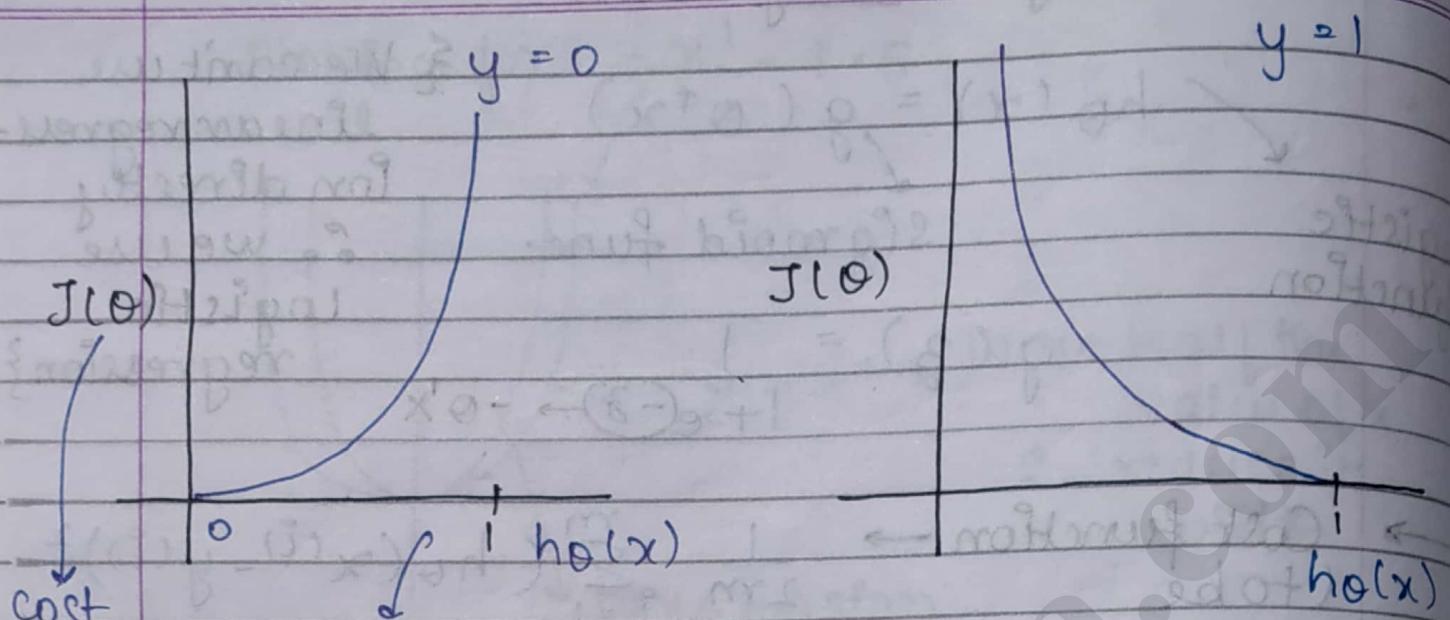
multiple maxima & minima



↳ We defined a new cost func.

$$0 \leq h_0(x) \leq 1$$

(assumption)



cost
func.
error.

Suppose $h_0(x) =$
 $y = 0$

predicted
value

∴ error is $\underline{\infty}$

Hence $y = 1$
and $h_0(x) =$
so no
error

- Cost func $\rightarrow \frac{1}{2m} \sum_{i=1}^m (h_0(x^{(i)}) - y^{(i)})^2$

$$= -\log(h_0(x)) \text{ when } y = 1$$

$$= -\log(1 - h_0(x)) \text{ when } y = 0$$

* Graph of $\log x$

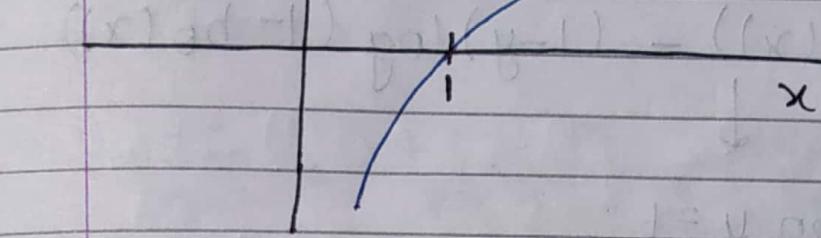
$$y = \log_{10} x$$

$$10^y = x$$

$$x = 0, y = 1 \quad (10, 1), (0.1, -1)$$

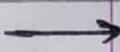
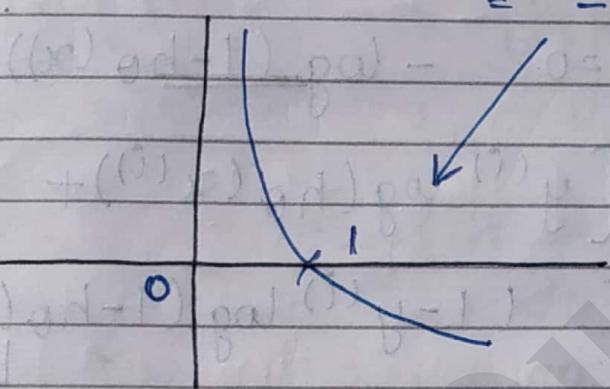
$$(3, 1/2)$$

$\log(x)$



$$y = f(x)$$

$$\begin{aligned} y &= -f(x) \\ &= -\log x. \end{aligned}$$

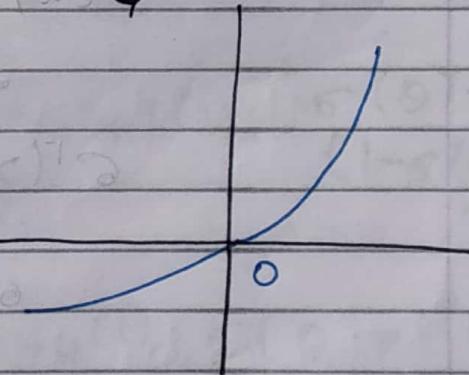


$$y = f(x)$$

$$y = -f(1-x)$$

$$x' = 1 - x \quad \rightarrow$$

$$y' = -y'$$



Dif. b/w linear & logistic regression.



Why linear regression can't be defined for classification.



What is logistic regression.



Derivation of cost func.



Numerical on logistic regression.

* Combined cost func. ↴

- y log (h₀(x)) - (1-y) log (1-h₀(x))

When y = 1

$$-\log(h_0(x)) \quad (0)$$

When y = 0 - log (1-h₀(x))

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \log(h_0(x^{(i)})) + (1-y^{(i)}) \log(1-h_0(x^{(i)}))] \quad \textcircled{1}$$

$$h_0(x^{(i)}) = \frac{1}{1+e^{-\theta^T x^{(i)}}} \rightarrow \text{sample } i$$

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

$$\sigma'(x) = \frac{-1(-e^{-x})}{(1+e^{-x})^2}$$

$$\sigma'(x) = \frac{e^{-x}}{(1+e^{-x})^2}$$

$$= \frac{1}{(1+e^{-x})} \cdot \frac{1+e^{-x}-1}{(1+e^{-x})} \cdot \frac{1}{(1+e^{-x})}$$

$$\sigma'(x) = \sigma(x) \left[\frac{1+e^{-x}}{(1+e^{-x})} - \frac{1}{(1+e^{-x})} \right]$$

$$\sigma'(x) = \sigma(x)(1-\sigma(x))$$

From ①,

$$\begin{aligned} \frac{\partial J}{\partial \theta_j} &= -\frac{1}{m} \sum_{i=1}^m y^{(i)} \frac{d}{\partial \theta_j} \log(h_0(x^{(i)})) + \\ &\quad (1-y^{(i)}) \frac{d}{\partial \theta_j} \log(1-h_0(x^{(i)})) \\ &= -\frac{1}{m} \sum_{i=1}^m \frac{y^{(i)} \frac{d}{\partial \theta_j} (h_0(x^{(i)}))}{h_0(x^{(i)})} + \frac{d(\log x)}{dx} \\ &\quad \frac{(1-y^{(i)}) \frac{d}{\partial \theta_j} (1-h_0(x^{(i)}))}{1-h_0(x^{(i)})} \end{aligned}$$

NOW

$$h_0(x^{(i)}) = \frac{1}{1+e^{-\theta^T x^{(i)}}}$$

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

$$\therefore h_0(x^{(i)}) = \sigma(\theta^T x^{(i)})$$

$$= -\frac{1}{m} \sum_{i=1}^m \frac{y^{(i)} \frac{d}{\partial \theta_j} (\sigma(\theta^T x^{(i)}))}{h_0(x^{(i)})} \rightarrow \frac{\sigma(\theta^T x^{(i)})}{(1-\sigma(\theta^T x^{(i)}))} x_j^{(i)}$$

$$= -\frac{1}{m} \sum_{i=1}^m \frac{y^{(i)} (1-h_0(x))}{1-h_0(x^{(i)}))} \rightarrow \frac{(1-y^{(i)}) \sigma(\theta^T x^{(i)})}{(1-\sigma(\theta^T x^{(i)}))} x_j^{(i)}$$

$$= -\frac{1}{m} \sum_{i=1}^m \frac{y^{(i)} (1-h_0(x))}{1-h_0(x^{(i)}))} \rightarrow \frac{(1-y^{(i)}) h_0(x^{(i)})}{1-h_0(x^{(i)}))} x_j^{(i)}$$

$$(1-y^{(i)}) h_0(x^{(i)}) x_j^{(i)}$$

common

$$\begin{aligned}
 &= -\frac{1}{m} \sum_{i=1}^m \left[y^{(i)} (1-h_0(x^{(i)})) - (1-y^{(i)}) \right. \\
 &\quad \left. h_0(x^{(i)}) \right] x_j^{(i)} \\
 &= -\frac{1}{m} \sum_{i=1}^m \left[y^{(i)} - y^{(i)} h_0(x^{(i)}) - h_0(x^{(i)}) \right. \\
 &\quad \left. + y^{(i)} h_0(x^{(i)}) \right] x_j^{(i)} \\
 &= -\frac{1}{m} \sum_{i=1}^m [y^{(i)} - h_0(x^{(i)})] x_j^{(i)}
 \end{aligned}$$

$$J(\theta) = \frac{1}{m} \sum_{i=1}^m [h_0(x^{(i)}) - y^{(i)}] x_j^{(i)}$$

same as linear regression cost func

$$\frac{1}{2m} \sum (h_0(x^{(i)}) - y^{(i)})^2$$

Upon derivative $\frac{1}{m} \sum (h_0(x^{(i)}) - y^{(i)})$

→ The only diff. b/w the 2 cost functions is that in linear regression, $h_0(x) = \theta^T x$ but in logistic regression we have

$$h_0(x) = g(\theta^T x)$$

8 feb, 19.

Q Apply logistic regression to find \hat{y}

$$\theta_0 = -2.16, \theta_1 = 0.425$$

$$\hat{P}(y = \text{yes} | x = \text{yes})$$

$$\Pr(y = \text{yes} | x = \text{no})$$

$$h_0(x) = P(y = 1) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$$

$$\begin{aligned} \text{probability} \\ \text{of } y = 1 \end{aligned} = \frac{1}{1 + e^{-(2.16 + 0.425x)}} \quad \textcircled{1}$$

$$\Pr(y = 1 | x = 1)$$

Using \textcircled{1}

$$\frac{1}{1 + e^{-(2.16 + 0.425)}} = \underline{\underline{0.85}} \quad \underline{\underline{0.149}}$$

$$\therefore P(y = 1 | x = 1) = \underline{\underline{0.85}} \quad \underline{\underline{0.149}}$$

Similarly $P(y = 1 | x = 0)$

$$\Pr(y = 1 | x = 0) \Rightarrow \frac{1}{1 + e^{-(2.16)}} = \underline{\underline{0.104}}$$

$\Pr(y = \text{no} | x = \text{no}/\text{yes})$

$$\downarrow -1 - P(y = \text{yes})$$

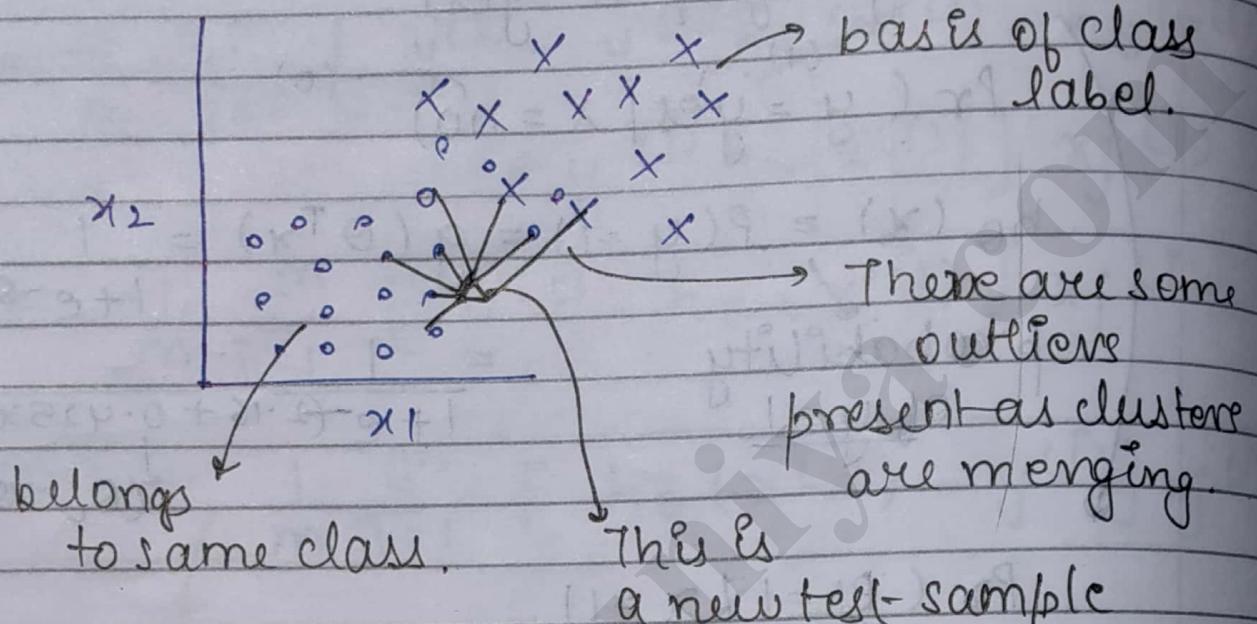
$$= 1 - \underline{\underline{0.149}} = \underline{\underline{0.851}}$$

→ K means classifier (eg - foriris data)

For a data with 2 features (x_1 and x_2)

Clusters on

basis of class
label.



(Using decision model, we need to decide
to which class it belongs)

We calculate distance of this training
set from all the other objects
present in data.

(using dist-formula),

Now sort all these pts. in increasing
order.

Now if $k = 5$, then output must be
5 nearest neighbours.

Now if 4 of them belong to class 1 and 1 to
class 2, then the training sample
belongs to class 1.

NOTE:- Take ' k ' as odd so that there is no ambiguity
relating to class.

CLASSTIME	Page No.:
Date	/ /

If $k = 2$ (even), then 1 may belong to class 1 & other to class 2.
Then, it is difficult to decide which class does training sample belongs.

TutorialsDuniya.com

Download FREE Computer Science Notes, Programs, Projects, Books PDF for any university student of BCA, MCA, B.Sc, B.Tech CSE, M.Sc, M.Tech at <https://www.tutorialsduniya.com>

- Algorithms Notes
- Artificial Intelligence
- Android Programming
- C & C++ Programming
- Combinatorial Optimization
- Computer Graphics
- Computer Networks
- Computer System Architecture
- DBMS & SQL Notes
- Data Analysis & Visualization
- Data Mining
- Data Science
- Data Structures
- Deep Learning
- Digital Image Processing
- Discrete Mathematics
- Information Security
- Internet Technologies
- Java Programming
- JavaScript & jQuery
- Machine Learning
- Microprocessor
- Operating System
- Operational Research
- PHP Notes
- Python Programming
- R Programming
- Software Engineering
- System Programming
- Theory of Computation
- Unix Network Programming
- Web Design & Development

Please Share these Notes with your Friends as well

facebook

WhatsApp 

twitter 

Telegram 