Lecture: 69

Introduction

Tuesday, June 15, 2021 12:07 PM

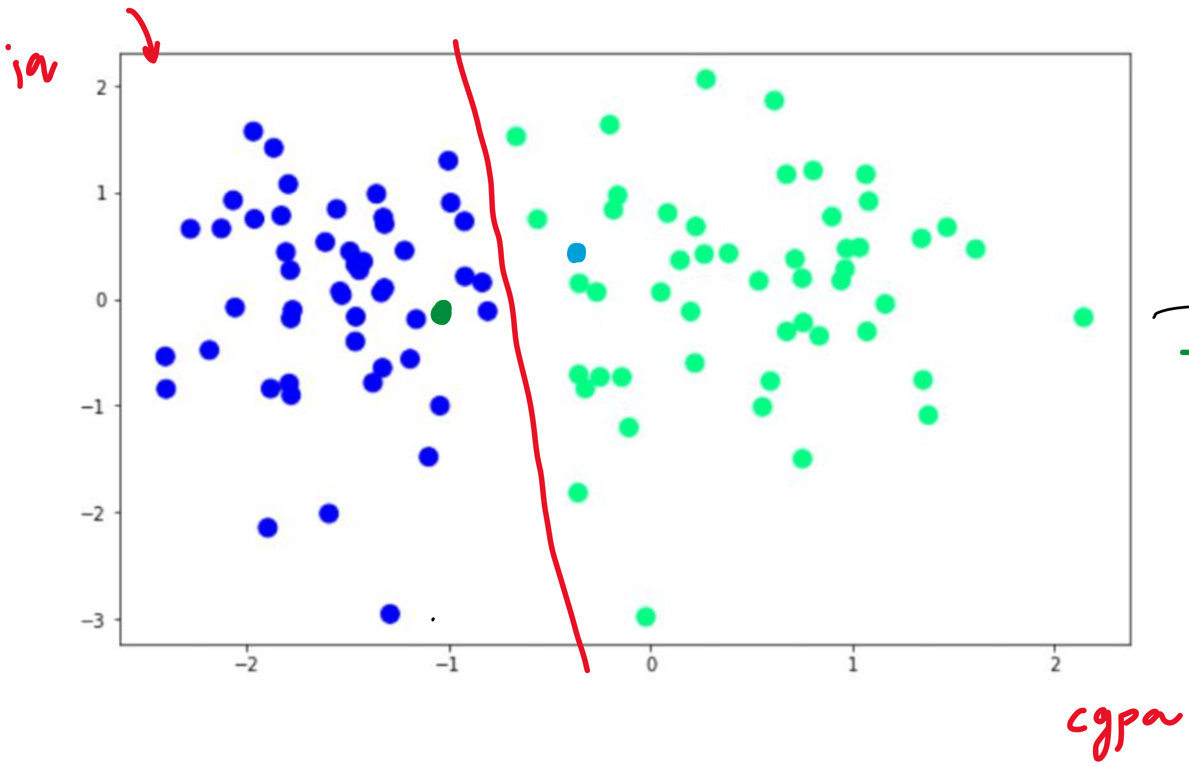
2টা perspectives এ logistics Regression implemented

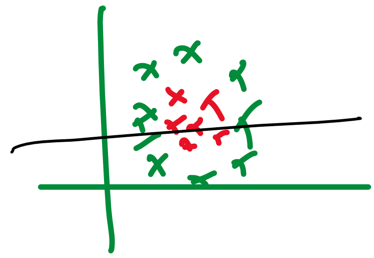
- geometric
- $\boxed{\text{Probability}}$ → আমরা probability টা পড়বো।

Deep learning এর fundamentals logistics regression থেকেই শুরু।

Requirement

Tuesday, June 15, 2021 12:07 PM





→ Linearly separable data থাকলে সেইটাতে Logistics Regression apply করতে পারবো।

⊛ Linearly separable or almost linearly separable ⊛

→ এখানেও আমরা Linear Regression এর মত line draw করি।

[প্রথমে perceptron পড়েছিলাম DL এ/সেইটা থেকে পড়বো।]

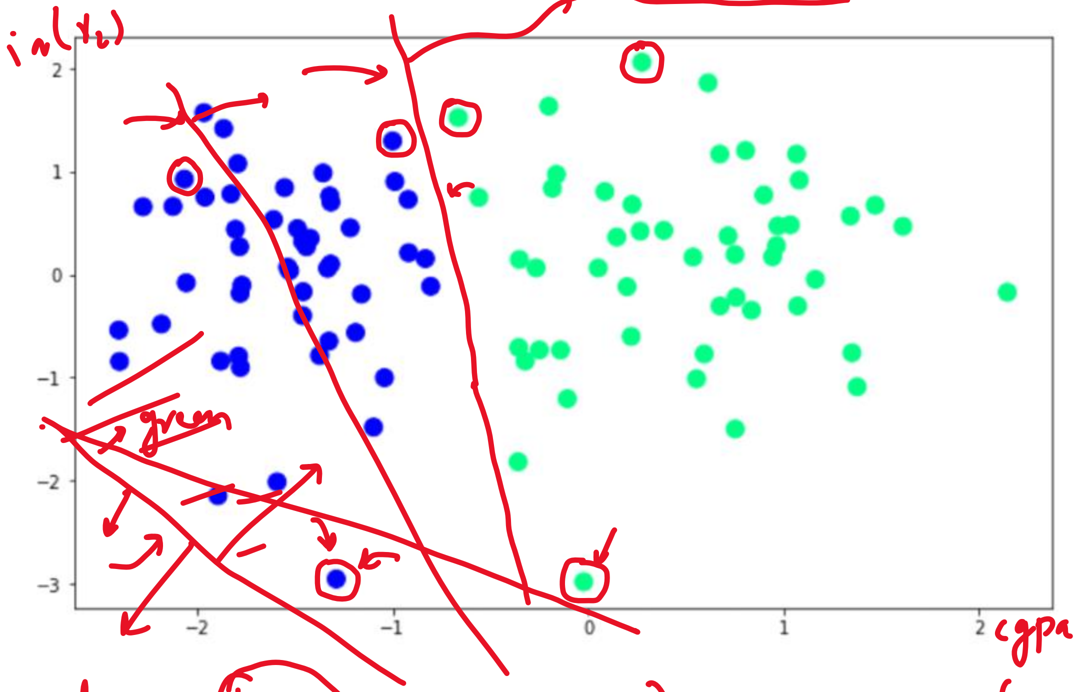
https://github.com/yasin-arafat-05/100DaysDL/blob/main/note/5_perceptron_trick.md

Perceptron Trick

Tuesday, June 15, 2021 12:08 PM

$y = mx + b$

$Ax + By + c = 0$



$Ax_1 + Bx_2 + C = 0$

$Ax_1 + Bx_2 + Cx_3 + d = 0$

$\boxed{A, B, C}$

$A = 1, B = 1$

$C = 0$

loop → 1000

→ random student

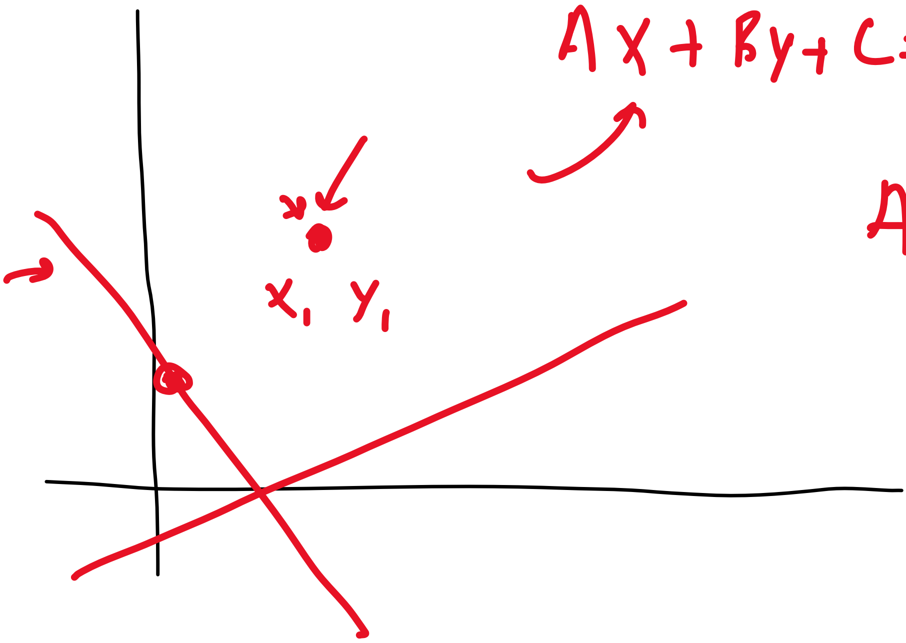
1000

while converge

How to label regions?

Tuesday, June 15, 2021 1:12 PM

$$Ax + By + C = 0$$



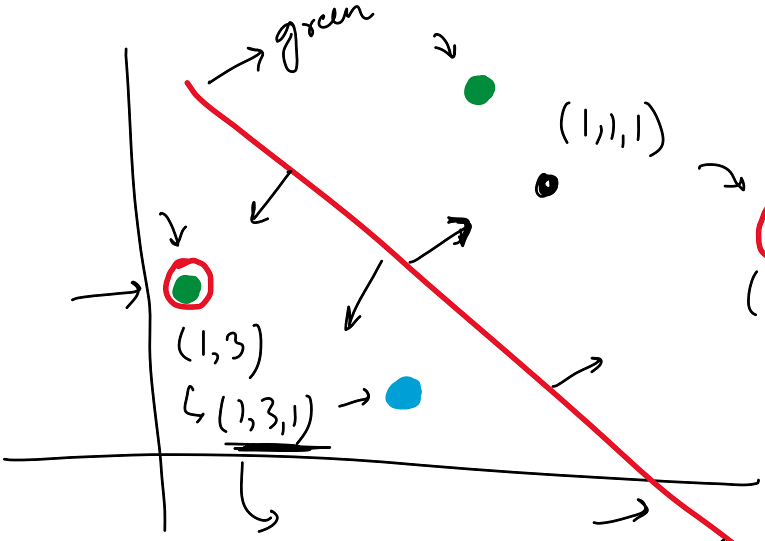
$$Ax_1 + By_1 + C > 0$$

$$<$$

$$= 0$$

Transformations

Tuesday, June 15, 2021 1:31 PM



$2x + 3y + 5 = 0$

blue

2	3	5
4	5	1

$-$

$-2x - 2y + 4 = 0$

2	3	5
0.01	0.03	0.01

$+$

$3x + 6y + 6 = 0$

learning rate → 0.01

$\text{coef} = \boxed{\text{coef}} - \boxed{\eta \text{ coordinate}}$

Algorithm

Tuesday, June 15, 2021 2:31 PM

x_0	(x_1) cgpa	(x_2) iq	y placed
1	7.5	81	1
1	8.9	109	1
1	7.0	81	0

$Ax + By + c = 0$

$\boxed{w_0 + w_1 x_1 + w_2 x_2 = 0}$

$\sum_{i=0}^{2} w_i x_i = 0$

$w_0 = C, \quad w_1 = A \quad w_2 = B$

$\boxed{w_0 \; w_1 \; w_2}$

$w_0 x_0 + w_1 x_1 + w_2 x_2 = 0$

$w_0 \times 1 + w_1 \times 7.5 + w_2 \times 81 \rightarrow \sum_{i=0}^{2} w_i x_i = 0$

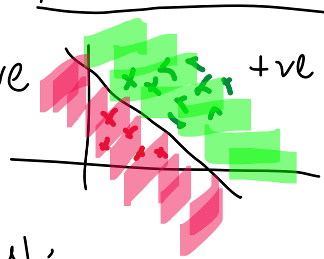
$[w_0 \;\; w_1 \;\; w_2] \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$

$= \quad \geq 0 \rightarrow 1$

$< 0 \rightarrow 0$

$\boxed{w_0 x_0 + w_1 x_1 + w_2 x_2} \rightarrow 1$

$\quad \downarrow 0$



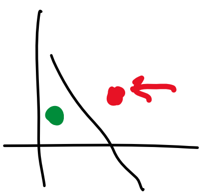
$x_i \in P$

$x_i \in N$

epoch → 1000 ; $\eta = 0.01$

for i in range(epochs):

randomly select a student



$[w_0 \; w_1 \; w_2]$

- if $x_i \in N$ and $\sum_{i=0}^{2} w_i x_i \geq 0$ { (+ve)

  $W_{new} = W_{old} - \eta x_i$

- if $x_i \in P$ and $\sum_{i=1} w_i x_i < 0$

  $W_{new} = W_{old} + \eta x_i$

{ $x_i \in N \quad w_i x_i < 0$

$x_i \in P \quad w_i x_i \geq 0$

Simplified Algo

Tuesday, June 15, 2021 2:44 PM

if $x_i \in N$ and $\sum w_i x_i \geq 0$

$$w_n = w_0 - \eta x_i$$

if $x_i \in P$ and $\sum w_i x_i < 0$

$$w_n = w_0 + \eta x_i$$

+ − একই সাথে control করতে পারবো নিচের equation এ

for i in 1000

random student

$$\boxed{w_n = w_0 + \eta(y_i - \hat{y}_i) x_i}$$

0

$$w_n = w_0$$

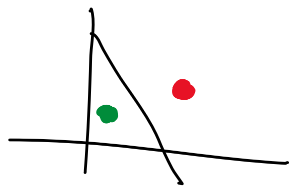
1

−1

$$w_n = w_0 + \eta x_i$$

$$w_n = w_0 - \eta x_i$$

y_i	$\hat{y}_i$	$y_i - \hat{y}_i$
1	1	0
0	0	0
1	0	1
0	1	−1



```
for i in range(epochs):
    select a random student (i)
    w_n = w_0 + η (y_i - ŷ_i) x_i
```

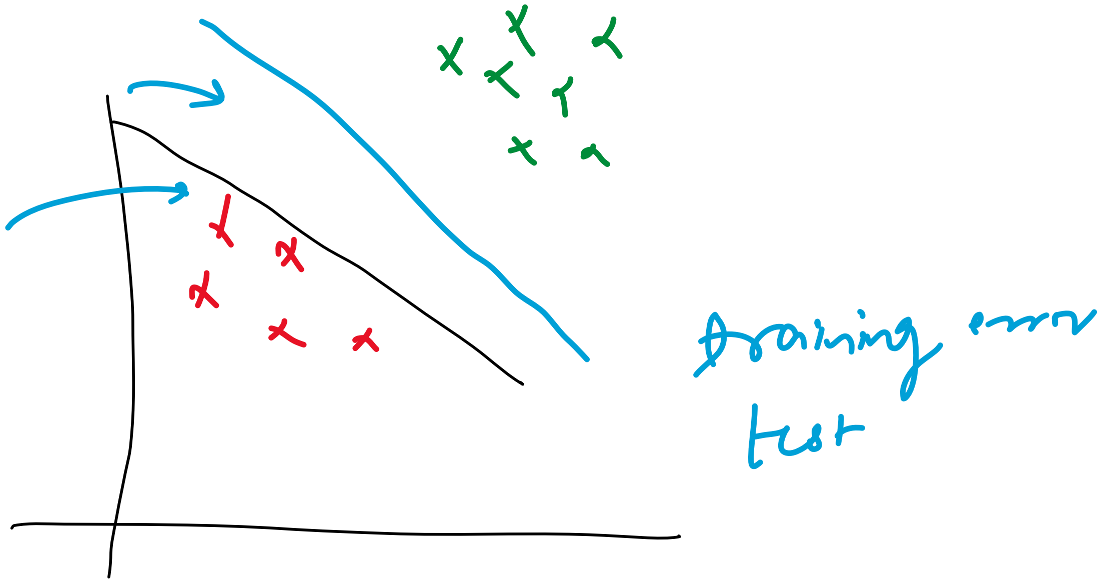
A, B, C

$$Ax + By + C = 0 \rightarrow y = mx + b$$

$$\boxed{m} = -\frac{A}{B} \qquad \text{coef}[\_ \;\; \_]$$

$$c = -\frac{C}{B} \qquad [\,,\,]$$

KS



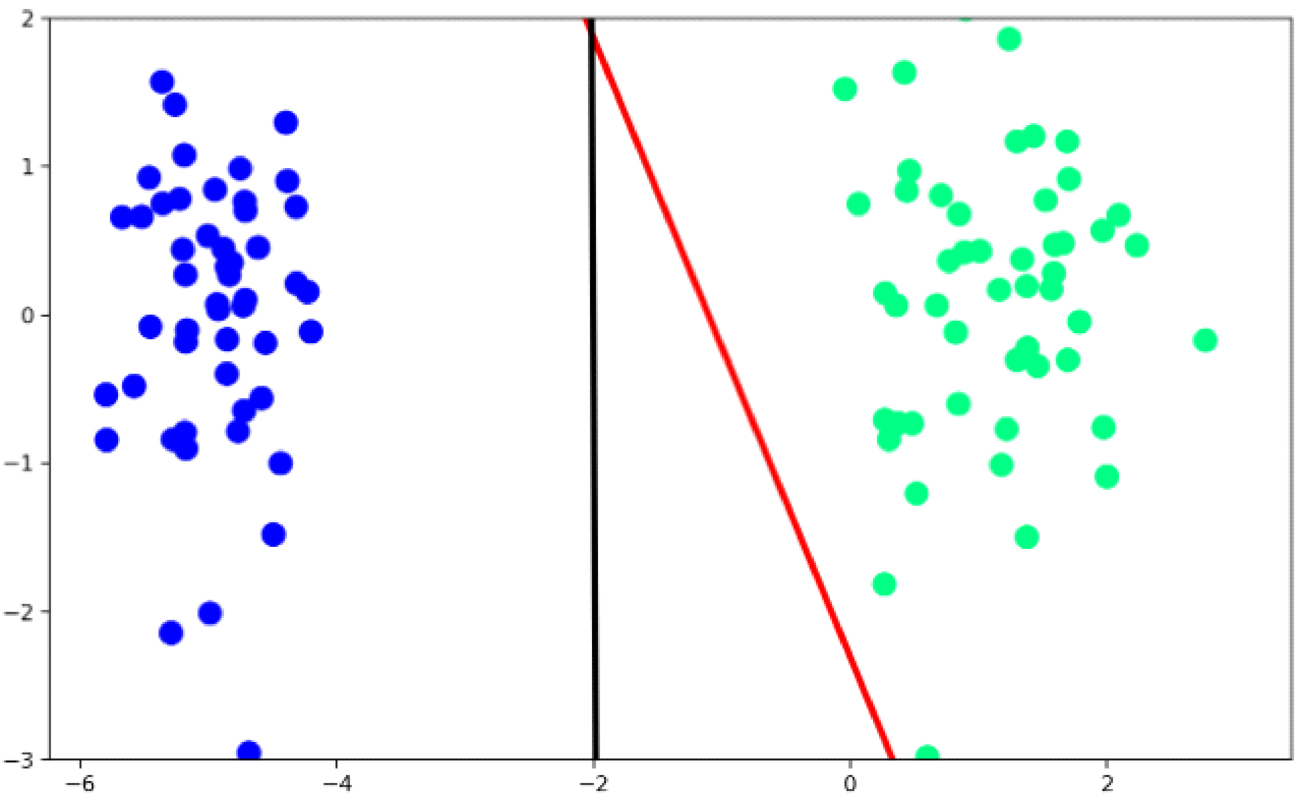
Lecture-70

- Code perceptron trick.
- Animation how fit the line.

Lecture-71

Problem with Perceptron

Thursday, June 17, 2021 12:18 PM

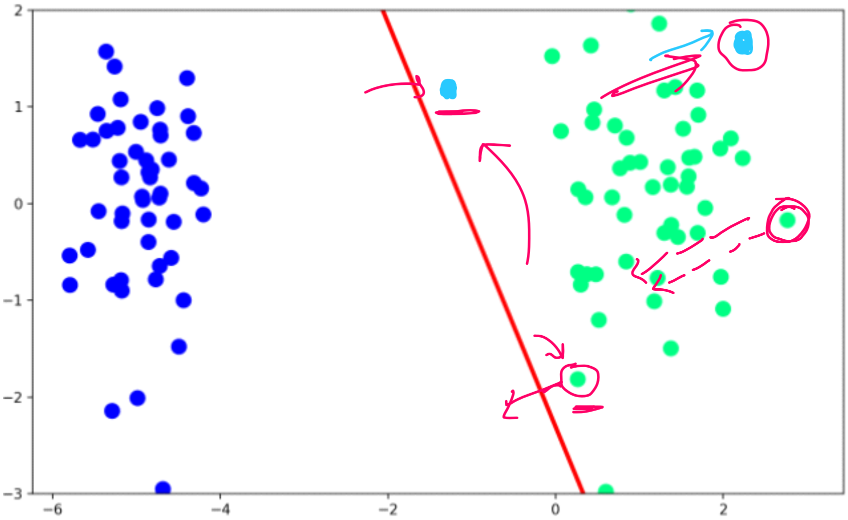


আমরা Perceptron trick এ randomly 1টা পয়েন্ট select করার পর, সেই কি তার সঠিক অবস্থানে আছে কি নেই তার উপর ভিত্তি করে parameter update করি। কিন্তু, perceptron trick এর চেয়ে logistics Regression ভালো কাজ করে। এখন কি করলে আমাদের এই result logistics regression এর মতো ভালো আসবে?

→ আগে, আমরা miscllsified point এর জন্য line কে pull করেছি।
→ এবার clssified point-গুলোর জন্য line কে দূরে পাঠাবো। + আগের টা। তাহলে, আমাদের line equilibrium এ আসবে।

Possible Solution?

Thursday, June 17, 2021 12:18 PM



$$\boxed{W_n = W_0 + (y_i - \hat{y}_i) x_i}$$

→ misclassified line-pull

→ correctly
└ line push

push pull

$$\underbrace{W_n = W_0} + \eta \underbrace{(y_i - \hat{y}_i)}_{0} x_i$$

y_i	$\hat{y}_i$	$y_i - \hat{y}_i$
1	1	0
0	0	0
1	0	1
0	1	-1

$(y_i - \hat{y}_i) \neq 0$ model predict

$\sum w_i x_i = \boxed{0, 1}$

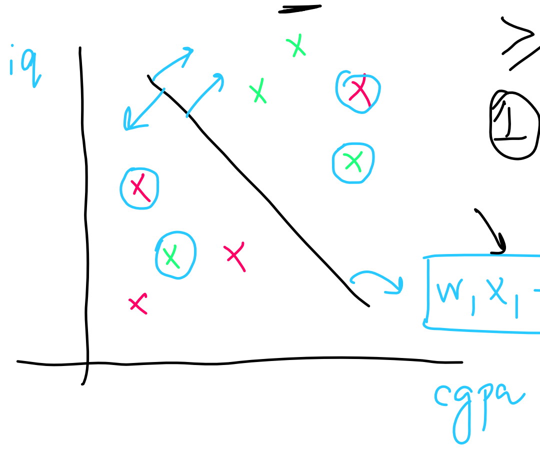
$(y_i - \hat{y}_i)$

$w_1 \times 8.1 + w_2 \times 81 + w_0 =$

>0 <0

1 0

$\{8.1, 81\}$



$\boxed{w_1 x_1 + w_2 x_2 + w_0 = 0}$

cgpa	iq	placed (x_i)
9	91	0
8.8	78	1
8.1	102	1
7.9	98	1

⊛ কোন misclassified point, line এর দূরে থাকলে বেশি distance এ নিজের দিকে pull করবে। line এর কাছে থাকলে কম distance এ নিজের দিকে pull করবে। আর, classified point এর জন্য কাছে থাকলে বেশি distance আর দূরে থাকে কম distance এ push করবে। (আমাদের target, $(y_i - \hat{y}_i)$ যেন 0 না হয়) (এটা case এর মধ্যে ২টা case এ।)

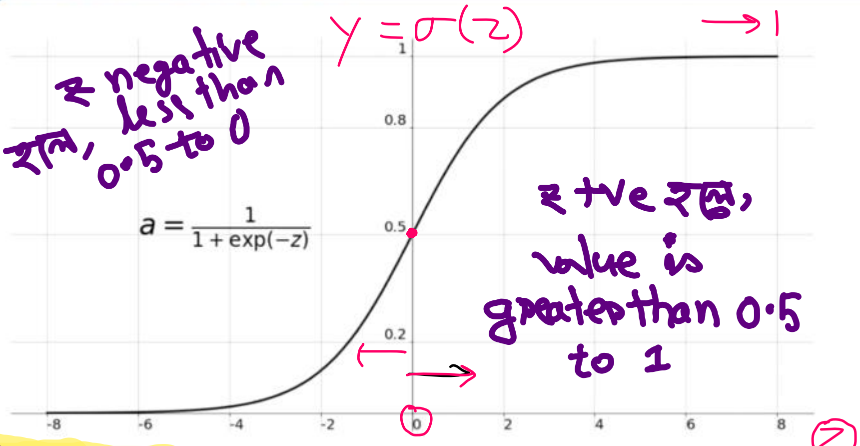
line এর equation এ মান বসানোর পর আমরা step fn use করেছিলাম। 0< আসলে, 0 আর greater than 0 আসলে 1 দিয়েছিলাম।

আমরা step fn এর পরিবর্তে যদি Sigmoid fn ব্যবহার করি তাহলে আমরা যা discuss করেছি তা achieve করতে পারবো। চলো sigmoid fn সম্পর্কে জানি।

The Sigmoid Function

Thursday, June 17, 2021 12:19 PM

Sigmoid Function



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

$$\boxed{-\infty < z < \infty}$$

$$0 < y < 1$$

For this, Points: $\{7.5, 81\} \to w_1 \; w_2 \; w_0$

$$\hat{y}_i = \boxed{w_1 \times 7.5 + w_2 \times 81 + w_0} = $$

$z \geq 0 \to 1$

$z < 0 \to 0$

$\sigma > 0.5 \leftarrow z = \sum w_i x_i = \sigma(z)$

$\sigma(z) < 0.5$

$\{7.5, 81\} \to z \to \sigma(z) = 0.5 \geq$

if 1 $P(N) = 1 - P(y)$

else 0

$0.5 <$ $P(y) =$

placement

Y P(N)

$P(y) = 0.7$

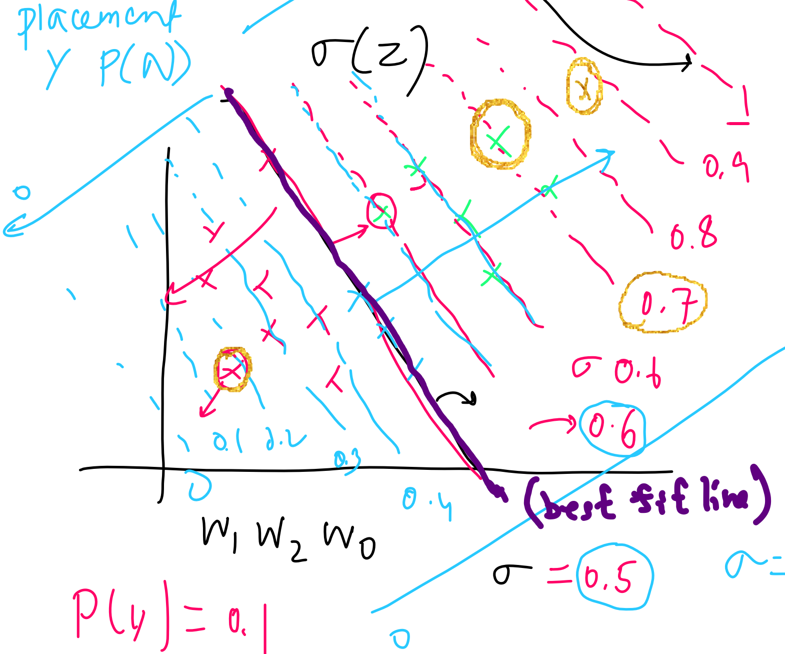
$\sum w_i x_i = 0$ $\sigma = 0.5$ $P(y) = 0.6$

$\sum w_i x_i > 0$ $\sigma > 0.5$

$\sum w_i x_i$ $\sigma > 0.6$

P(y) P(N)

$\sigma(\;) = P(\;)$



$\sigma = 0.5$ $\boxed{P(0.5)} \to P(y) = 0.5$

σ এর output-কে আমরা Probability হিসেবে কল্পনা করতে পারি। Best fit line এর উপর কোন point থাকলে $\sum w_i x_i = 0$, $\sigma(\sum w_i x_i) = 0.5$, best fit line এর উপর থেকে যত সামনের দিকে যাবো তত $\sigma(\sum w_i x_i)$ এর output বাড়তে থাকবে। আর তার probability ও বেশি হবে। নিচের দিকে probability তত কম হবে।

How sigmoid will behave for non classified & classified point :-

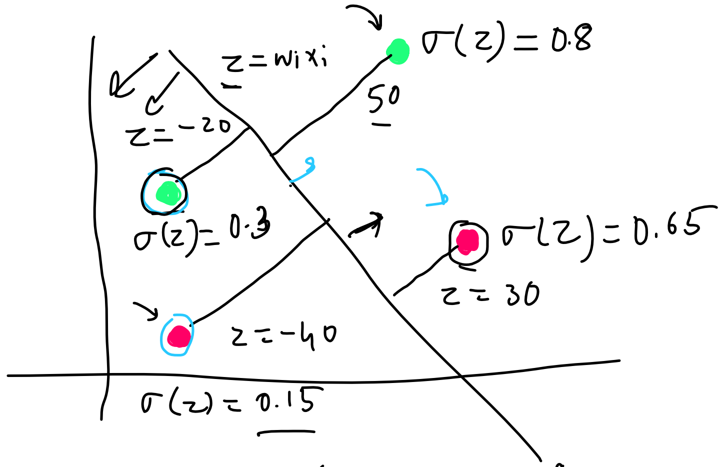
Impact of Sigmoid

Thursday, June 17, 2021 3:27 PM

$$w_n = w_0 + \eta(y_i - \hat{y}_i)x_i$$

$$\hat{y}_i = \sigma(z)$$

where $z = \sum w_i x_i$



y_i	$\hat{y}_i$	$y_i - \hat{y}_i$
1	0.8	0.2
0	0.65	-0.65
1	0.3	0.7
0	0.15	-0.15

$z=50$, $w_n = w_0 + \eta \times 0.2 \times x_i$ (Push line)

$z=30$, $w_n = w_0 - \eta * 0.65 * x_i$ (pull line)

$z=-20$, $w_n = w_0 + \eta * 0.7 * x_i$ (Push)

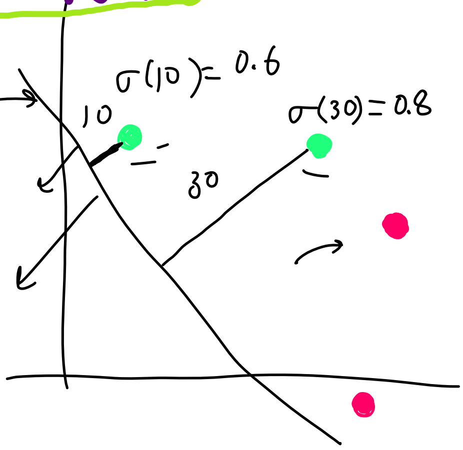
$z=-40$, $w_n = w_0 - \eta * 0.15 * x_i$ (Pull)

কে কত distance এ push / pull করবে? ???

$$w_n = w_0 + \eta(y_i - \hat{y}_i)x_i$$

$$\hat{y}_i = \sigma(z)$$

where $z = \sum w_i x_i$



y_i	$\hat{y}_i$	$y_i - \hat{y}_i$
1	0.6	0.4
1	0.8	0.2
0		
0		

1st

$w_n = w_0 + \boxed{\eta * 0.4 * x_i} = x_1$

$w_n = w_0 + \boxed{\eta * 0.2 * x_i} = x_2$

$x_1 > x_2$ → code implement

$\eta\, 0.4 x_i > \eta\, 0.2 x_i$;

$0.4 > 0.2$;

0.4 এর জন্য best fit line কাছে, তাই এটা long distance এ best fit line কে ধাক্কা দিবে। mathmatically proofs.

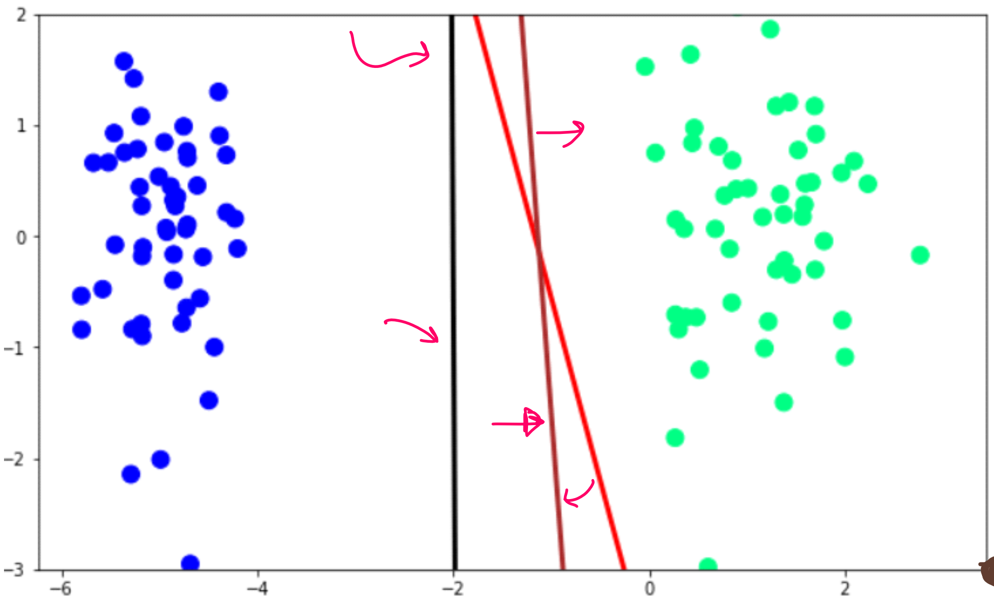
Sigmoid implement করার পর, Result ভালো হয়েছে। কিন্তু, এর থেকেও ভালো Result অর্জন করা সম্ভব।

Lecture: 72

(Problem with sigmoid f^n approach)

The Problem

Friday, June 18, 2021 2:20 PM



Color	technique used
red →	step
brown →	sigmoid
black →	sklearn

for i in 1000
random

Loss function
error function

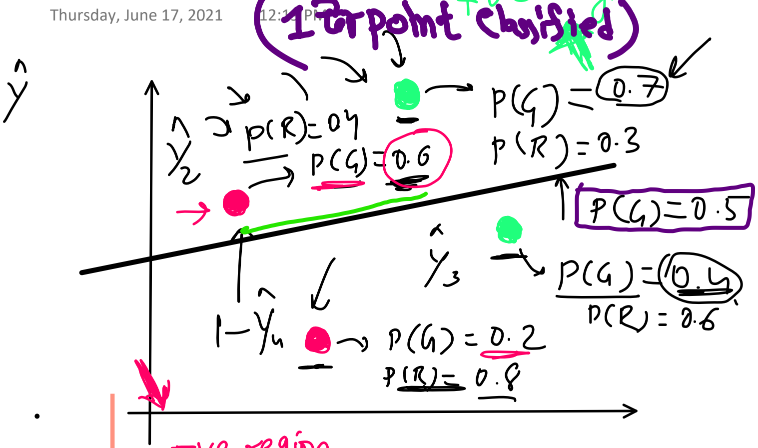
আমরা sigmoid f^n approach এ, randomly point select করে, লাইন তার সঠিক জায়গায় আছে কি না তা চেক করি। কিন্তু, প্রথমবারে randomly যে point গুলো select হয়েছে, পরেরবার যদি randomly এমন point গুলো select হয় যা আগের গুলোর থেকে ভালো result দেয়। এমন ঘটনা ঘটতেই পারে। তো, ML আমাদেরকে বলে, একটা f^n খোঁজে যেইটা loss calculation করে। যেইটা বলে, আমাদের ML model কতটুকু ভুল করতেছে। তারপর, আমরা সেই loss function এর minima থেকে parameter এর গুলোর মান বের করি।

এখানে, দিতে, model-1 দিয়ে 1 টা, model-2 দিয়ে 2 টা point classified হচ্ছে। এখন, কথা হচ্ছে, model-2 ভালো, এটা আমরা maximum likelihood দিয়ে প্রমান করবো। sigmoid f^n দিয়ে। লাইনের উপরে গেলে sigmoid এর output 0.5 থেকে 1 হবে আর নিচের দিকে আসলে (0 ~ 0.5) এমন (যেমন আগে) দেখেছি।

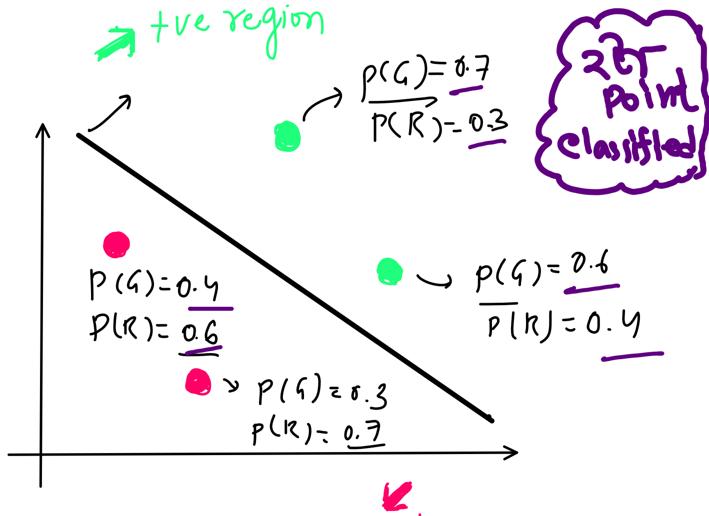
Maximum likelihood বলে ঃ [Prediction ($\hat{y}$) output of sigmoid fn] সব গুলোকে গুন করে যে model এর value বেশি তার output বেশি হবে।

Maximum Likelihood and Cross-Entropy

Thursday, June 17, 2021 12:1? PM



$\hat{y}_i$



$\hat{y} = \sigma(z)$

$\hookrightarrow z = \sum w_i x_i$

model 2 → 0.7 × 0.6 × 0.6 × 0.7

= 0.176

Green Green red red

model 1 → 0.7 × 0.4 × 0.4 × 0.8 = 0.089

$\log(ab) = \log a + \log b$

data point গুলো অনেক বেশি হলে product অনেক কাছাকাছি হবে তাই log ব্যবহার করবো

log (0.1) > log(0.9)

-1   -0.04

$\log(\max) = \boxed{-\log(0.7)} - \log(0.4) - \log(0.4) - \log(0.8)$

0-1 = -ve

cross entropy   minimize

-ve log এর maximum likelihood এর Σ কে cross entropy বলে।

NO

$-\log(\hat{y}_1) - \log(\hat{y}_2) - \log(\hat{y}_3) - \log(\hat{y}_4)$

cross entropy কে কমাবো না বাড়াবো? কমাবো কারণ, log(0.1) থেকে log(0.9) এর থেকে বড়।

আমরা হুবহু এই formula নিতে পারবো না। কারণ p(red) = 0.8 হলে, p(green) = 1 - p(red) = 1 - 0.8 = 0.2

$(1-\hat{y})$ modified formula $y_i = 1$

$$-y_i \log(\hat{y}_i) - (1-y_i)\log(1-\hat{y}_i)$$

model-1 এর 1st point এর জন্য $y_i = 1$,

$-1\log(0.7) - 0\log(0.3)$

$$-y_1 \log(\hat{y}_1) = -\log(\hat{y}_1)$$

$$= -\log(0.7)$$

$y_2 = 0$ $y_3 = 1$

[এটা আমরা ১ম এ পেয়েছি] = log(0.7)

$= -\log(0.7)$

$\boxed{y_2 = 0}$ $\quad y_3 = 1$

$-y_2 \log(\hat{y}_2) - (1-y_2)\log(1-\hat{y}_2)$

$-\log(1-\hat{y}_2) = -\log 0.4$

$\boxed{y_3}$ $\quad -y_3 \log(\hat{y}_3)$

$-\log(\hat{y}_3) = -\log(0.4)$

$y_4 = 0$

$-(1-y_4)\log(1-\hat{y}_4)$

$-\log(1-\hat{y}_4)$

$-\log(0.8)$

final formula

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

MSE

closed form

$$\boxed{L = -\frac{1}{n}\sum_{i=1}^{n} y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)}$$

gradient descent

min $w_1 w_2 w_0$

বলে → log-loss error

→ Binary cross entropy

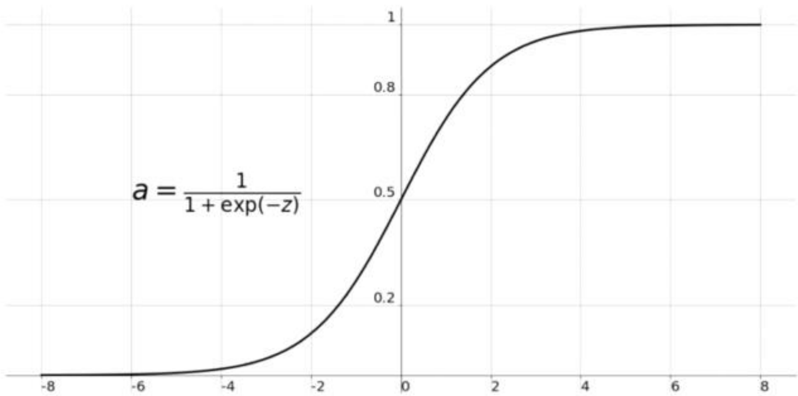
log-loss বা Binary cross-entropy এর minimum value বের করার জন্য কোন formula নেই। আমরা gradient descent apply করি।

Derivative of Sigmoid

Thursday, June 17, 2021 12:20 PM

Lecture÷73 → [Gradient Descent on Derivative of Sigmoid]

Sigmoid Function



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

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

$$\frac{d}{dx}\left(\frac{1}{x}\right) = \frac{d}{dx}(x)^{-1}$$

$$= -x^{-2} = -\frac{1}{x^2}$$

$$\frac{d}{dx}\left[\frac{1}{1+e^{-x}}\right] = \frac{d}{dx}\left[(1+e^{-x})^{-1}\right] = -\frac{1}{(1+e^{-x})^2}\frac{d}{dx}(1+e^{-x})$$

$e^{-x} = e^{-x}$ $\quad -x = -1$

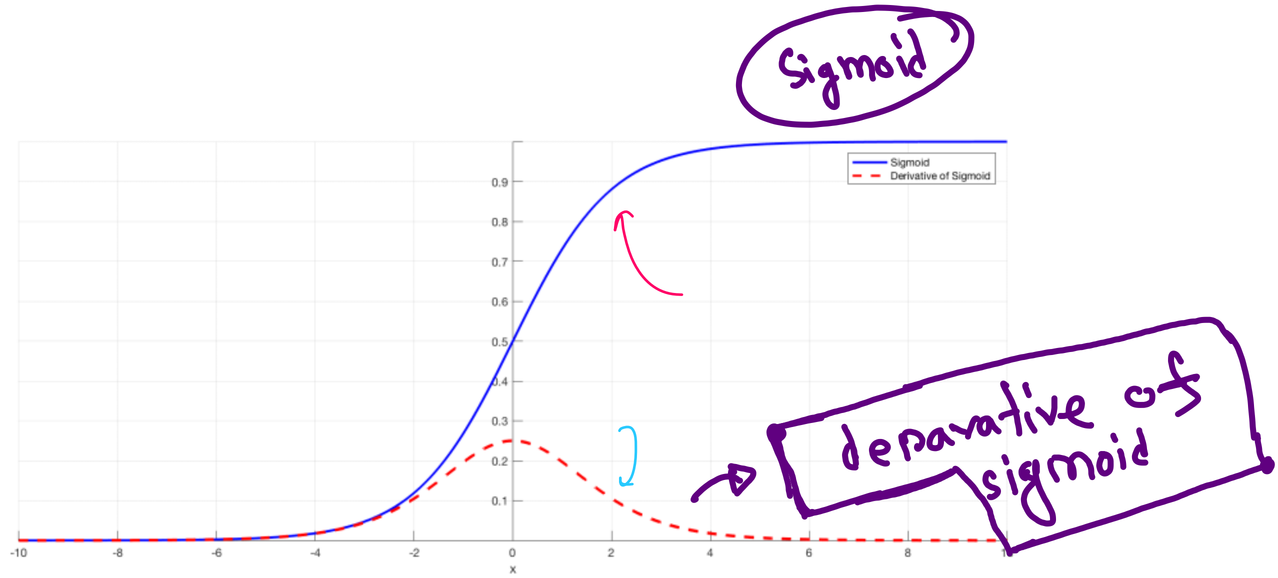
$$= -\frac{1}{(1+e^{-x})^2}\frac{d}{dx}(e^{-x}) = -\frac{e^{-x}}{(1+e^{-x})^2}\frac{d}{dx}(-x) = \frac{e^{-x}}{(1+e^{-x})^2}$$

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

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

$$\sigma(x)\left[1-\sigma(x)\right] \Rightarrow \sigma'(x) = \boxed{\sigma(x)\left[1-\sigma(x)\right]}$$

(final derivative)


Sigmoid
Sigmoid
Derivative of Sigmoid
derivative of sigmoid
x

Lecture: 74 (Gradient Descent)

Gradient Descent

Monday, June 21, 2021 11:50 AM

$\rightarrow \boxed{\text{working}}$ $\rightarrow$ classification $\{x_1, x_2\} \rightarrow y$

$x_0 \quad x_1 \quad x_2 \quad y = \{0, 1\}$

$w_1 x_1 + w_2 x_2 + w_0 = 0$

$\hat{y}_i = \sigma(z) =$

$\hat{y}_i = \sigma(z) \quad z = \sum_{j=0}^{n} w_j x_j$

$w_1 x_1 + w_2 x_2 + w_0 x_0$

$\sum_{j=0}^{2} w_j x_j = z$

$$L = -\frac{1}{m} \sum_{i=1}^{m} y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)$$

gradient descent

$L(w_1, w_2, w_0) =$

minimum

for i in epochs

0.01

argmin $(w_0\ w_1\ w_2)$

$w_1\ w_2\ w_0 = 1 \rightarrow w_{new} = w_{old} - \eta \frac{\partial L}{\partial w_{old}}$

$$\underbrace{w_0 = w_0 - \eta \frac{\partial L}{\partial w_0}, \quad w_1 = w_1 - \eta \frac{\partial L}{\partial w_1}, \quad w_2 = w_2 - \eta \frac{\partial L}{\partial w_2}}$$

n cols $\rightarrow$ n+1 derivatives

$w_n = w_n - \eta \frac{\partial L}{\partial w_n}$ $\qquad j = 0, 1, 2 \ldots n$

$$L = -\frac{1}{m} \sum_{i=1}^{m} y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)$$

$$\frac{\partial L}{\partial w_j} = -\frac{1}{m} \sum_{i=1}^{m} \left[\frac{\partial L}{\partial w_j} \left(y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i) \right) \right]$$

$\hat{y}_i = \sigma(z) \quad z = \sum_{j=0}^{n} w_j x_j$

$\frac{\partial L}{\partial w_j} y_i \log(\hat{y}_i) = \frac{\partial L}{\partial w_j} y_i \log(\sigma(z))$

$- y_i \quad \frac{\partial L}{} \quad \sigma\left(\sum^{n} w_j x_j\right)$

$$\frac{\partial L}{\partial w_j} y_i \log(\hat{y}_i) \qquad \frac{\partial}{\partial w_j} y_i \log \qquad = \frac{y_i}{\hat{y}_i} \frac{\partial L}{\partial w_j} \sigma\left(\sum_{j=0}^{n} w_j x_j\right)$$

$$\frac{\partial L}{\partial w_j} y_i \log\left(\sigma\left(\sum_{j=0}^{n} w_j x_j\right)\right)$$

$$= y_i \frac{\partial L}{\partial w_j} \log\left(\sigma\left(\sum_{j=0}^{n} w_j x_j\right)\right) \quad \hat{y}_i \qquad = \frac{y_i}{\cancel{\hat{y}_i}} \cancel{\hat{y}_i}(1-\hat{y}_i) \frac{\partial L}{\partial w_j} \sum_{j=0}^{n} w_j x_j$$

$$y_i(1-\hat{y}_i) \sum_{j=0}^{n} \frac{\partial L}{\partial w_j} w_j x_j = \boxed{y_i(1-\hat{y}_i) \sum_{j=0}^{n} x_j}$$

$$\frac{\partial L}{\partial w_j} (1-y_i)\log(1-\hat{y}_i) \Rightarrow (1-y_i) \frac{\partial L}{\partial w_j} \log(1-\hat{y}_i) \qquad \hat{y}_i = \sigma(z) \quad \sigma\left(\sum_{j=0}^{n} w_j x_j\right)$$

$$\frac{(1-y_i)}{(1-\hat{y}_i)} \frac{\partial L}{\partial w_j} (1-\hat{y}_i) \Rightarrow -\frac{(1-y_i)}{(1-\hat{y}_i)} \frac{\partial L}{\partial w_j} \hat{y}_i \Rightarrow -\frac{(1-y_i)}{(1-\hat{y}_i)} \frac{\partial L}{\partial w_j} \sigma(z)$$

$$\Rightarrow -\frac{(1-y_i)}{(1-\hat{y}_i)} \sigma(z)(1-\sigma(z)) \frac{\partial L}{\partial w_j} \sigma(z) \quad \hat{y}_i \qquad = -\frac{(1-y_i)}{\cancel{(1-\hat{y}_i)}} \hat{y}_i \cancel{(1-\hat{y}_i)} \frac{\partial L}{\partial w_j} \sigma(z)$$

$$\Rightarrow -\hat{y}_i(1-y_i) \frac{\partial L}{\partial w_j} \sum_{j=0}^{n} w_j x_j \Rightarrow -\hat{y}_i(1-y_i) \sum_{j=0}^{n} \frac{\partial L}{\partial w_j} (w_j x_j)$$

$$\Rightarrow \boxed{-\hat{y}_i(1-y_i) \sum_{j=0}^{n} x_j}$$

$$\frac{\partial L}{\partial w_j} = -\frac{1}{m} \sum_{i=1}^{m} \left[y_i(1-\hat{y}_i) \sum_{j=0}^{n} x_j - \hat{y}_i(1-y_i) \sum_{j=0}^{n} x_j \right]$$

$$\frac{\partial L}{} \quad \sum^{m} \left[\quad - \hat{y}(1-y_i) \right] \sum^{n} x_j$$

$$\frac{\partial L}{\partial w_j} = -\frac{1}{m}\sum_{i=1}^{m}\left[y_i(1-\hat{y}_i) - \hat{y}_i(1-y_i) \right]\sum_{j=0}^{n} x_j$$

$$= -\frac{1}{m}\sum_{j=1}^{m}\left[y_i - \cancel{y_i\hat{y}_i} - \hat{y}_i + \cancel{y_i\hat{y}_i} \right]\sum_{j=0}^{n} x_j$$

$$\boxed{\frac{\partial L}{\partial w_j} = -\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)\sum_{j=0}^{n} x_j}$$

x_1	x_2	y_i	$\hat{y}_i$
1	2	1	0
3	4	0	0

no. of rows $= m = 2$

no. of cols $= n = 2$

$$\frac{\partial L}{\partial w_0} = -\frac{1}{2}[1+0]\,[1+1]$$

$$= -\frac{1}{\cancel{2}}[1]\,[\cancel{2}] = -1$$

$$= -\frac{1}{2}\begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$= -\frac{1}{2}[1] = -\frac{1}{2}$$

Gradient Descent

Monday, June 21, 2021 1:53 PM

Dataset Rows = m Columns = n

n input

	1	2	3		n	y	$\hat{y}$
1	x_{11}	x_{12}	x_{13}	......	x_{1n}	y_1	$\hat{y}_1$
2	x_{21}	x_{22}	x_{23}	....	x_{2n}	y_2	$\hat{y}_2$
⋮	⋮						
m	x_{m1}	x_{m2}	x_{m3}	......	x_{mn}	y_m	$\hat{y}_m$

যেমন logistics regression এ input n হলে

$\{w_1 \; w_2 \; w_3 \; \ldots \; w_n \;\; w_0\}$

$(n+1) \Rightarrow$ Parameter (n+1) টা হবে।

Predicted output

$$\sigma(w_1 x_{11} + w_2 x_{12} + w_3 x_{13} + \ldots + w_n x_{1n} + w_0) = \hat{y}_1$$

$$\sigma(w_1 x_{21} + w_2 x_{22} + w_3 x_{23} + \ldots + w_n x_{2n} + w_0) = \hat{y}_2$$

$\hat{Y}$

$$\hat{Y} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_m \end{bmatrix} = \begin{bmatrix} \sigma(w_0 + w_1 x_{11} + w_2 x_{12} + \ldots + w_n x_{1n}) \\ \sigma(w_0 + w_1 x_{21} + w_2 x_{22} + \ldots + w_n x_{2n}) \\ \vdots \\ \sigma(w_0 + w_1 x_{m1} + w_2 x_{m2} + \ldots + w_n x_{mn}) \end{bmatrix}$$

(sigma কে বাইরে আনছি)

$$\hat{Y} = \sigma\left(\begin{bmatrix} w_0 + w_1 x_{11} + w_2 x_{12} + \ldots + w_n x_{1n} \\ w_0 + w_1 x_{21} + w_2 x_{22} + \ldots + w_n x_{2n} \\ \vdots \\ w_0 + w_1 x_{m1} + w_2 x_{m2} + \ldots + w_n x_{mn} \end{bmatrix}\right)$$

w_0 A X B W

$$\hat{Y} = \sigma\left(\begin{bmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1n} \\ 1 & x_{21} & x_{22} & \ldots & x_{2n} \\ \vdots & & & & \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \vdots \end{bmatrix}\right)$$

(input feature) X (Bias) W

$$\hat{y} = \sigma\left(\begin{bmatrix} 1 & x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & & & & \\ 1 & x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \begin{bmatrix} \vdots \\ w_n \end{bmatrix} \right)$$

Predicted value in matrix form

এখন, আমরা Loss function কে matrix আকারে প্রকাশ করবো

$$\boxed{\hat{y} = \sigma(XW)}$$

$$L = -\frac{1}{m}\sum_{i=1}^{m} y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)$$

$$L = -\frac{1}{m}\left[\sum_{i=1}^{m} y_i \log(\hat{y}_i) + \sum_{i=1}^{m}(1-y_i)\log(1-\hat{y}_i)\right]$$

$(1-y)\log$

$\sigma(1-XW)$

[প্রথমে 1st part টুকু করি]

1st $\sum_{i=1}^{m} y_i \log(\hat{y}_i) = y_1 \log \hat{y}_1 + y_2 \log \hat{y}_2 + y_3 \log \hat{y}_3 + \cdots + y_m \log \hat{y}_m$

2nd $\begin{bmatrix} y_1 & y_2 & y_3 & \cdots & y_m \end{bmatrix} \begin{bmatrix} \log \hat{y}_1 \\ \log \hat{y}_2 \\ \vdots \\ \log \hat{y}_m \end{bmatrix}$

3rd $\begin{bmatrix} y_1 & y_2 & y_3 & \cdots & y_m \end{bmatrix} \log\left(\begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_m \end{bmatrix} \right)$

(Actual output)

Predicted o/p

$$\boxed{Y \log \hat{Y} = y \log(\sigma(XW))}$$

$\hat{Y}$ এর মান বসিয়ে

সুতরাং

$$L = -\frac{1}{m}\left[y \log \hat{y} + (1-y)\log(1-\hat{y}) \right]$$

min

where $\hat{y} = \sigma(XW)$

GD [W] find

where $\hat{y} = \sigma(XW)$ L (GD) [W]...

Loss function in Matrix form

$$L = -\frac{1}{m}\left[y\log(\sigma(wx)) + (1-y)\log(1-\sigma(wx))\right]$$

W এর এমন মান বের করতে হবে যাতে eq. (minimum) মান হয়।

we will use gradient descent.

randomly initialize, $W = [\quad]$ then

$$\boxed{\begin{array}{l}\text{for i in epochs:}\\ \rightarrow W = W - \eta\dfrac{\Delta L}{\Delta W}\end{array}}$$

$W_0 \longrightarrow W_n$

$$\left\{\frac{\Delta L}{\Delta W}\right\} \qquad \frac{\Delta L}{\Delta W} = \left[\frac{\partial L}{\partial w_0}, \frac{\partial L}{\partial w_1}, \frac{\partial L}{\partial w_2}, \cdots \frac{\partial L}{\partial w_n}\right]$$

হলো $(n+1)$

$\frac{\Delta L}{\Delta W}$

$$L = -\frac{1}{m}\left[y\log\hat{y} + (1-y)\log(1-\hat{y})\right]$$

$$\frac{dL}{dW} =$$

$\boxed{\hat{y} = \sigma(xw) \text{ এই } y \text{ constant, আর, } \hat{y} \text{ না।}}$

$$\frac{d}{dW}\,y\log\hat{y} \Rightarrow y\frac{d}{dW}\log\hat{y} \Rightarrow \frac{y}{\hat{y}}\frac{d}{dL}(\hat{y})$$

(log x এর সূত্র)

$\boxed{\text{not actual } \times}$

σ এর derivative
$\sigma[1-\sigma]$

$\Rightarrow \hat{y}$

$$\Rightarrow \frac{y}{\hat{y}}\frac{d}{dL}\sigma(WX) \Rightarrow \frac{y}{\hat{y}}\,\sigma(wx)\left[1-\sigma(wx)\right]\frac{d}{dW}(WX)$$

$$= \frac{y}{\cancel{\hat{y}}}\,\cancel{\hat{y}}(1-\hat{y})\,x = \boxed{y(1-\hat{y})X} \qquad \hat{y} = \sigma(wx)$$

2nd term of log loss f^n

$$\frac{d}{dw}(1-y)\log(1-\hat{y}) \Rightarrow (1-y)\frac{d}{dw}\log(1-\hat{y}) \Rightarrow \frac{(1-y)}{(1-\hat{y})}\frac{d}{dw}[1-\hat{y}]$$

$$= -\frac{(1-y)}{(1-\hat{y})}\frac{d}{dw}\sigma(wx) \Rightarrow -\frac{(1-y)}{(1-\hat{y})}\left[\sigma(wx)[1-\sigma(wx)]\right]\frac{d}{dL}(wx)$$

$$\Rightarrow -\frac{(1-y)}{\cancel{(1-\hat{y})}}\hat{y}\cancel{(1-\hat{y})}X = \boxed{-\hat{y}(1-y)X}$$

overall

$$\frac{dL}{dw} = -\frac{1}{m}\left[y(1-\hat{y})X - \hat{y}(1-y)X\right]$$

$$= -\frac{1}{m}\left[y(1-\hat{y}) - \hat{y}(1-y)\right]X$$

$$= -\frac{1}{m}\left[y - \cancel{y\hat{y}} - \hat{y} + \cancel{y\hat{y}}\right]X$$

$$\boxed{\frac{\Delta L}{\Delta w} = -\frac{1}{m}(y-\hat{y})X}$$

final answer

gd

2

$$\boxed{w = w + \eta\frac{1}{m}(y-\hat{y})X}$$

যেখানে

$$\boxed{\underline{W} = \underline{W} + \eta \frac{1}{m} (\underline{Y - \hat{Y}}) X}$$

$(n+1), 1$

$$W = \begin{bmatrix} W_0 \\ W_1 \\ \vdots \\ W_n \end{bmatrix} \qquad X = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1n} \\ 1 & x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & & & & \\ 1 & x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \qquad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

$m, n+1$ $\qquad$ $m, 1$

$$\hat{Y} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_m \end{bmatrix} \leftarrow (m, 1)$$

$$\underline{W} = \underline{W} + \boxed{\frac{\eta}{m}} (\underline{Y - \hat{Y}}) \underline{X}$$

$$(n+1, 1) = \underbrace{(n+1, 1)}_{\to (n+1), 1} \quad (1, m) \; m, (n+1) \to (1, n+1)$$

(Transpose ব্যবহার করি)

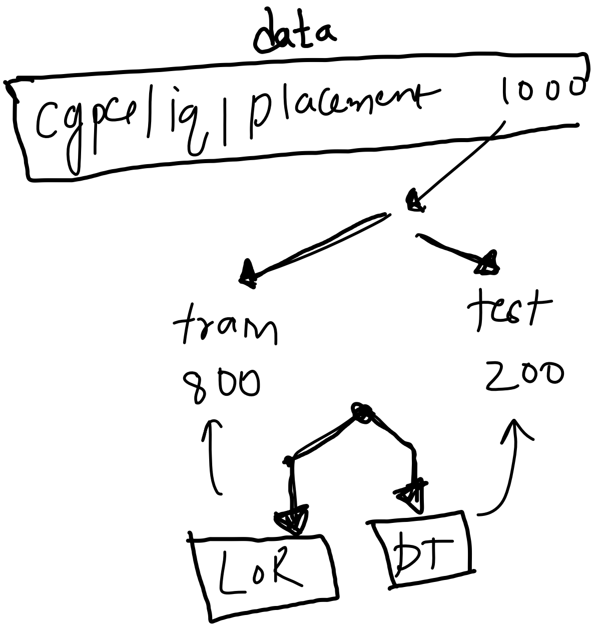
Lecture : 75 (Accuracy, confusion matrix, Classification Matrix)

(Binary Class classification)

Accuracy

Tuesday, June 22, 2021 1:12 PM

Actual Label	Logistic Regression Prediction	Decision Tree Prediction
1	✓ 1	✓ 1
0	✗ 1	✗ 1
0	✓ 0	✓ 0
0	✓ 0	✓ 0
1	✓ 1	✓ 1
1	✓ 1	✓ 1
0	✗ 1	✓ 0
0	✓ 0	✓ 0
0	✓ 0	✓ 0
1	✓ 1	✓ 1



$$\text{accuracy} = \frac{\text{no. of } ✓}{\text{total prediction}}$$

$$\frac{8}{10} = 0.8 \rightarrow 80\% \text{ (LoR)}$$

$$\frac{9}{10} = 90\% \text{ (DT)}$$

$$\boxed{\text{accuracy} = \frac{\text{no. of correct prediction}}{\text{total prediction}}}$$

Accuracy of multi-classification problem

Wednesday, June 23, 2021 8:44 AM

Actual Label	Logistic Regression Prediction	Decision Tree Prediction
0	0	0
0	0	0
0	0	0
2	2	2
0	0	0
2	2	2
0	0	0
2	2	2
1	1	1
1	1	1

iris

setosa, virginica | versicolor

0 1 2

$$\text{accuracy} = \frac{\#\text{ correct prediction}}{\#\text{ total}}$$

$$= \frac{10}{10} = 1 = 100\%$$

How much accuracy is good?

Wednesday, June 23, 2021 11:14 AM

(এইটা আমার Problem এর উপর নির্ভর করে।)

→ যদি medical related question এর জন্য model এর accuracy 100% হওয়াই লাগবে। যদি 99% হয় তাহলে 100 জনের মধ্যে 1 জন মারা যাওয়ার সম্ভবনা থাকবে।

The Problem with Accuracy

Wednesday, June 23, 2021 11:14 AM

90% → $\boxed{10\%}$ →

binary

Y | N

→ Y ✓ → X

→ N X → ✓

Y ✓ → X

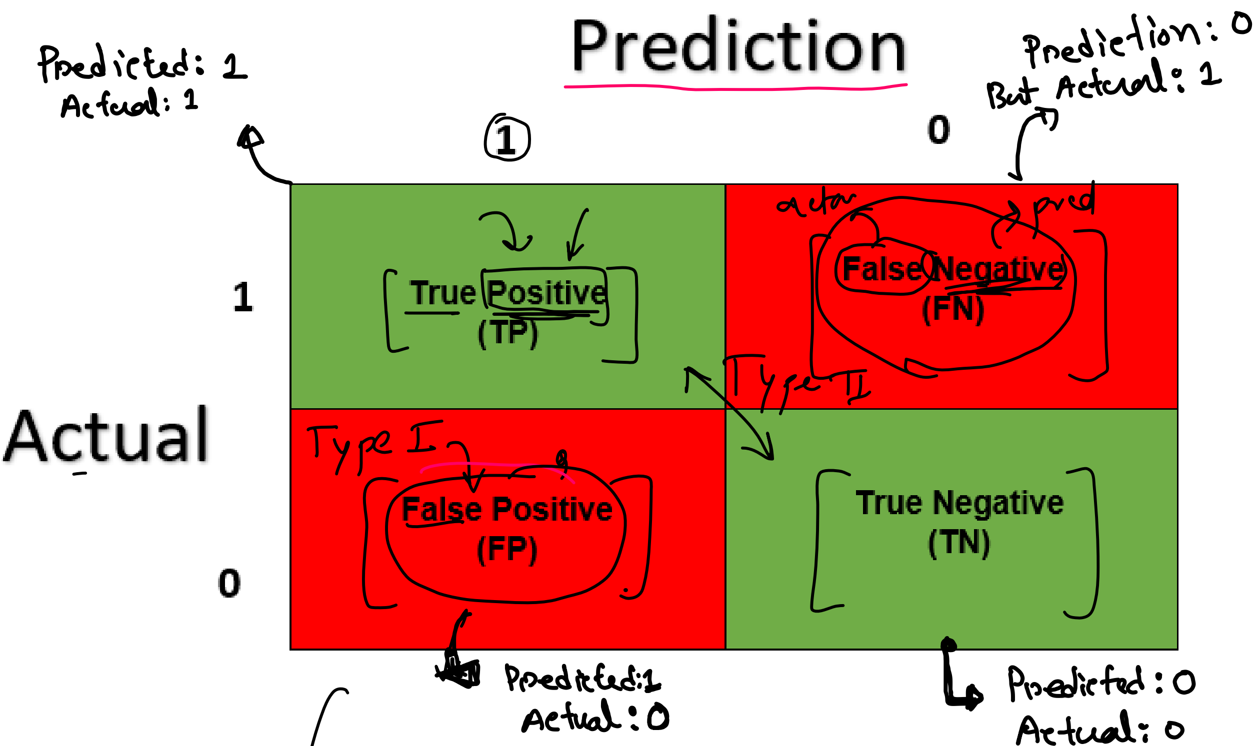
N X → ✓

Classification prob এর ক্ষেত্রে, আমরা কোনটাতে ভুল করতেছি সেটা accuracy বলে দিতে পারে না। Placement হবে, নাকি হবে না এমন, 90% accuracy বিশিষ্ট কোন model এর ক্ষেত্রে, আমাদের prediction হওয়ার কথা ছিল YES কিন্তু model এর output No এসেছে।

অর্থাৎ, nature of error কি সেইটা accuracy score বলতে পারে না। এর জন্য, আমরা confution matrix পড়বো যেইটা আমাদের confution দূর করবে।

Confusion Matrix

Tuesday, June 22, 2021 1:12 PM



$$acc = \frac{TP + TN}{TP + TN + FP + FN}$$

True Positive → Actual: 1, predicted value: 1

False Positive → Actual NO, Predicted Yes

True Positive, True negative → (সঠিক prediction হচ্ছে অর্থাৎ true আছে)

False Positive → Type-1 (error)

False Negative → Type-2 (error)

Extented

↳ { echo or not }

duplicate

↳ echo

$1+2+3=6 \rightarrow$ outcome

$\boxed{\frac{xy}{2y}}$

Type 1 Error

Wednesday, June 23, 2021 8:45 AM

(False-Positive কে type-1 error বলে)

Type 2 Error

Wednesday, June 23, 2021 8:45 AM

(False-Negative) Type-2 error

Heart-disease নেই (Predicted) ~~কিন্তু~~, actually আছে সেইটা Type-2 error।

Confusion Matrix for Multi-classification Problem

Wednesday, June 23, 2021 8:45 AM

0, 1, 2 →

$\boxed{10 \times 10}$

Predicted

3 x 3

binary

2 x 2

		Predicted 0	Predicted 1	Predicted 2
Actual	0	7	0	5
	1	2	21	6
	2	9	1	13

$$\frac{\text{True predict}}{\text{total}} = \text{accuracy}$$

3x3

When accuracy is misleading?

Wednesday, June 23, 2021 8:45 AM

Imbalanced Dataset

1000 people

999 Not terrorist

1 terrorist

model → No one is terrorist

		Predicted	
		1	0
Actual	1	0	1
	0	0	999

$$\text{Accuracy} = \frac{999}{999+1} = \boxed{99.9\%}$$

Imbalance data এর ক্ষেত্রে, যেমন, 1000 জনের মধ্যে 1 জন চোর এমন data এ থেকে, prediction, ৯৯৯ ঠিক 1 বার ভুল হলে,

$$\text{accuracy} = \frac{৯৯৯}{1000} = ৯৯.৯\%$$ কিন্তু, accuracy

0% হওয়ার কথা ছিল, কারণ, আমাদের, Dataset এ চোর কিন্তু একটাই। এ জন্য আমরা এখন, Precision, recall, F1 score আসবো।

Lecture: 76 (Precision, Recall, F1)

Precision

Wednesday, June 23, 2021 8:46 AM

{spam: 1, not spam: 0}

Predicted

Actual

A

	Sent to Spam	Not sent to spam
Spam	100	170 FN
Not Spam	30 FP	700

B

	Sent to Spam	Not sent to spam
Spam	100	190
Not Spam	10	700

$\frac{800}{1000} = 80\%$ $\frac{800}{1000} = 80\%$

FP and FN

২ ক্ষেত্রে accu. সেম, কিন্তু, কোনটা বেশি ভালো বুঝবো কিভাবে।

$P_A = \frac{100}{100+30}$ $P_B = \frac{100}{100+10}$

$P_A < P_B$

FP → Not spam → Spam

FN → spam → Not spam

FP আর FN এর মধ্যে কোনটা ভয়ানক?

FP Not spam email কে spam বলনাম

FP যার কম সেইটাই ভালো, তাই, model B ভালো

0

Binary

+ve

Precision

What proportion of predicted Positives is truly Positive?

1

যতগুলোকে আমরা +ve বলেছি তার মধ্যে actually কতগুলো +ve সেইটা হচ্ছে Precision।

	Sent to Spam	Not sent to spam
Spam	True Positive	False Negative
Not Spam	False Positive	True Negative

Precision= TP/(TP+FP)

(যার Precision বেশি তাকে Deploy করবো)

Precision এর ক্ষেত্রে, আমরা email spam exam নিয়েছি। এক্ষেত্রে, FP এর উপর ~~ভিত্তি~~ করে model deploy করেছি। কিন্তু, always, accuracy সমান হলে, আগে precision check করবো এমন নয়। আগে আমরা Problem statement দেখবো।

Recall

Wednesday, June 23, 2021 8:46 AM

has cancer : 1 , no cancer : 0

Predicted

Actual

	Detected Cancer	Not Detected
Has Cancer	1000	200 FN
No Cancer	800 FP	8000

	Detected Cancer	Not Detected
Has Cancer	1000	500 ← B
No Cancer	500	8000

A 90% B 90%

$$Recall = \frac{1000}{1200} \qquad R_B = \frac{1000}{1500}$$

FP

$$\boxed{R_A} > R_B$$

কত জনের cancer আছে, আর তার মধ্যে কাদের কে আলাদা করা গিয়েছে। এর ratio হচ্ছে Recall

What proportion of actual Positives is correctly classified?

	Detected Cancer	Not detected Cancer
Has Cancer	True Positive	False Negative
No Cancer	False Positive	True Negative

Recall= TP/(TP+FN)

(যার Recall বেশি সেইটা ভালো)

(F1 Score)

F1 Score

Wednesday, June 23, 2021 8:46 AM

$$F1\,score = \frac{2PR}{P+R}$$

$P = 0 \quad R = 100$

$F1\,score = 50$

$F1 = 0$

$$F1\,score = \frac{P+R}{2}$$

(A)

$P = R = 80$

$$\frac{80+80}{2} = 80 \qquad F1 = 80$$

$$\frac{2 \times 80 \times \cancel{80}^{40}}{\cancel{160}_{2}} = 80$$

(B) 75

$P = 60 \quad R = 100$

$$\frac{100+60}{2} = 80$$

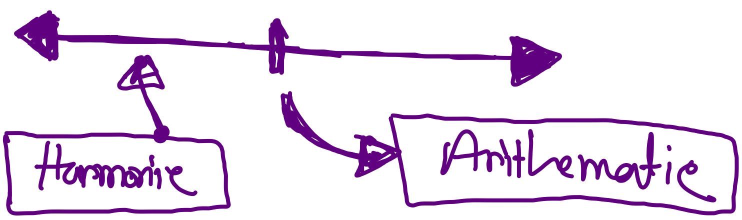
$$= \frac{2 \times 60 \times 100}{160}$$

$= 75$

F1 → Score, আমরা যদি বলতে না পারি, accuracy ভালো, কিন্তু Recall ভালো হবে নাকি Precision হবে সেক্ষেত্রে আমরা F1 score বের করি। এখানে, আমরা Harmonic mean নেই।

$$F1.score = \frac{2PR}{P+R} \;/\; \frac{P+R}{2}$$

(Harmonic mean) (Arithmetic mean)



(যার কম হবে, F1 score তার দিকে থাকে। এটারে example দেখানো হয়েছে।)

Multi-class Precision and Recall

Wednesday, June 23, 2021 8:46 AM

(For Multi-Class Classificition)

Binary Class-Classification এর ক্ষেত্রে, আমরা 1 বা +ve কে importance দেই। কিন্তু, আমরা, 1,0 ২টার জন্যই F1, Precision, Recall বের করি। average = None.
আমরা +ve for যদি, " = True

Positive ① Yes NO

Binary ② Class

Multi ③

Dog Cat Rabbit

$$\frac{2 \times 0.86 + 0.66}{0.86 + 0.66}$$

R_D, R_C, R_R (Recall)

$$F1_D = \frac{2 P_D R_D}{P_D + R_D}$$

$$F1_C = \frac{2 P_C R_C}{P_C + R_C}$$

$F1_R$

Actual total

Predicted

Actual		Dog	Cat	Rabbit	Total	Recall
	Dog	25	5	10	40	0.62
	Cat	0	30	4	34	0.88
	Rabbit	4	10	20	34	0.58
	Total	29	45	34		
	Precision	0.86	0.66	0.58		

Predicted total

$$R_D = \frac{25}{40} = 0.62, \quad R_C = \frac{30}{34} = 0.88, \quad R_R = \frac{20}{34} = 0.58$$

- macro recall
- weighted recall

$F1_D$, $F1_C$, $F1_R$

- macro f1
- weighted f1

Precision

আমার মডেল যতগুলোকে +ve বলেছে, তার মধ্যে actually কত গুলো +ve, $\frac{TP}{TP+FP}$ → Actually +ve, → Model +ve

$$P_{Dog} = \frac{25}{29} = 0.86 \quad P_{Cat} = \frac{30}{45} = 0.66 \quad P_{Rabbit} = \frac{20}{34} = 0.58$$

→ Overall Precision বের করার পদ্ধতি ২টা (Macro, Weighted)

→ Macro, $\frac{0.86 + 0.66 + 0.58}{3} = 0.70$

→ weighted, $\left(\frac{40}{108} \times 0.86\right) + \left(\frac{34}{108} \times 0.66\right) + \left(\frac{34}{108} \times 0.58\right) = 0.71$

সব level equal থাকলে, Macro
না থাকলে, weighted ব্যবহার করবো।

Multi-class F1 Score

Thursday, June 24, 2021 11:14 AM

Lecture: 77 (Softmax Regression)

Softmax Regression

Monday, June 28, 2021 3:35 PM

cgpa	iq	place?
7.1	71	0
8.5	85	1
9.5	95	2

→ classifier

{ Yes } 1
{ No } 0
{ Opt out } 2 → (অন্য কোন কারণে বের হয়ে গেছে)

{6.5, 65}

→ Yes / No / opt out

Softmax কে

- Softmax
- Multinomial
- LR (Logistics)

→ Deep Learning

Binary classification

softmax → general

↑ for 2class

$\boxed{0 < \sigma < 1}$

Softmax function

K = no. of class {3}

yes → 1
no → 2
opt → 3

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

$$\sigma(z)_3 = \frac{e^{z_3}}{e^{z_1} + e^{z_2} + e^{z_3}}$$

(Yes)

$$\sigma(z)_1 = \left\{ \frac{e^{z_1}}{e^{z_1} + e^{z_2} + e^{z_3}} \right\}$$

No

$$\sigma(z)_2 = \frac{e^{z_2}}{e^{z_1} + e^{z_2} + e^{z_3}}$$

Logistics Regression এর মতোই, কিন্তু Sigmoid এর পরিবর্তে softmax ব্যবহার করি। Softmax এ কোন একটা class এর value দিলে সেই class হওয়ার probability বের করে দেয়।

Training in softmax regression

Training Intuition

Monday, June 28, 2021 3:35 PM

{Y, N, D} → 0, 1, 2

cgpa	iq	place?
–	–	0
–	–	1
–	–	2

(Placement কে OHE দিয়ে transform করবো)

0 → 1 0 0
1 → 0 1 0
2 → 0 0 1

transform OHE

cgpa	iq	=0	=1	=2
✓	✓	1	0	0
		0	1	0
		0	0	1

এর পর 3টা Dataset বানাবো, cgpa, iq, output থাকবে 1 কলামে

D → D_1, D_2, D_3

3 coeff

$w_1\ w_2\ w_0$

D_1: cgpa | iq | =0 → YES → m_1 → $w_1^{(0)}, w_2^{(0)}, w_0^{(0)}$

D_2: cgpa | iq | =1 → NO → m_2 → $w_1^{(1)}, w_2^{(1)}, w_0^{(1)}$

D_3: cgpa | iq | =2 → optout → m_3 → $w_1^{(2)}, w_2^{(2)}, w_0^{(2)}$

{ Loss function, gradient descent } behind the scene

3টা Dataset নিয়ে 3টা LOR model train করবো। প্রতিটা model থেকে 3টা total 9 টা parameter পাবো।

Prediction: new student {7, 70}

$z_1 = 7 \times w_1^{(0)} + 70\, w_2^{(0)} + w_0^{(0)}$

$z_2 = 7 \times w_1^{(1)} + 70\, w_2^{(1)} + w_0^{(1)}$

$z_3 = 7 \times w_1^{(2)} + 70\, w_2^{(2)} + w_0^{(2)}$

$\sigma(\text{YES}) = \dfrac{e^{z_1}}{e^{z_1}+e^{z_2}+e^{z_3}}$ $\quad \sigma(\text{NO}) = \dfrac{e^{z_2}}{e^{z_1}+e^{z_2}+e^{z_3}}$ $\quad \sigma(\text{optout}) = \dfrac{e^{z_3}}{e^{z_1}+e^{z_2}+e^{z_3}}$

0.4 0.3 0.3 → 0.4 (placement Done)

Loss Function

Monday, June 28, 2021 3:47 PM

sk-learn এ কোনো মতে softmax regression implement করে না। চলো, sk-learn কীভাবে কাজ করে তা দেখি।

আমরা loss fn কে modify করবো যাতে একটা দিয়েই LoR, Softmax এর কাজ চালাতে পারি।

$$L = -\frac{1}{m}\sum_{i=1}^{m} y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)$$

Modified loss fn

$$L = -\frac{1}{m}\underbrace{\sum_{i=1}^{m}}_{\text{data}}\underbrace{\sum_{K=1}^{K}}_{\text{class}} y_K^{(i)} \log(\hat{y}_K^{(i)})$$

$K = \{1, 2, 3\}$ $i=1$

example

$w_1^{(1)}\ w_2^{(1)}\ w_0^{(1)}$

	X_1	X_2	Y	$Y_{K=1}$	$Y_{K=2}$	$Y_{K=3}$
1	X_{11}	X_{12}	1	1	0	0
2	X_{21}	X_{22}	2	0	1	0
3	X_{31}	X_{32}	3	0	0	1

transformation

0 → কতগুলো হয়ে যাবে dataset থেকে

1st row

$$y_1^{(1)}\log(\hat{y}_1^{(1)}) + \boxed{y_2^{(1)}\log(\hat{y}_2^{(1)})} + \boxed{y_3^{(1)}\log(\hat{y}_3^{(1)})} +$$

2nd row

$$+ \boxed{y_1^{(2)}\log(\hat{y}_1^{(2)})} + y_2^{(2)}\log(\hat{y}_2^{(2)}) + \boxed{y_3^{(2)}\log(\hat{y}_3^{(2)})} +$$

3rd row

$$+ \boxed{y_1^{(3)}\log(\hat{y}_1^{(3)})} + \boxed{y_2^{(3)}\log(\hat{y}_2^{(3)})} + y_3^{(3)}\log(\hat{y}_3^{(3)})$$

Simplified

$$L = y_1^{(1)}\log(\hat{y}_1^{(1)}) + y_2^{(2)}\log(\hat{y}_2^{(2)}) + y_3^{(3)}\log(\hat{y}_3^{(3)})$$

$\hat{y}_1^{(1)}, \hat{y}_2^{(2)}, \hat{y}_3^{(3)}$

Loss function এ $Y_1\ Y_2\ Y_3$ value জানি, কিন্তু $\hat{Y}$ এর value জানি না।

softmax

Now,

$$\hat{y}_1^{(1)} = \sigma(w_1^{(1)}x_{11} + w_2^{(1)}x_{12} + w_0^{(1)})$$

$$y_2^{(2)} = \sigma(w_1^{(2)}x_{21} + w_2^{(2)}x_{22} + w_0^{(2)})$$

$$y_3^{(3)} = \sigma(w_1^{(3)}x_{31} + w_2^{(3)}x_{32} + w_0^{(3)})$$

$$\begin{bmatrix} w_1^{(1)} & w_2^{(1)} & w_0^{(1)} \\ w_1^{(2)} & w_2^{(2)} & w_0^{(2)} \\ w_1^{(3)} & w_2^{(3)} & w_0^{(3)} \end{bmatrix}$$

L

Parameter matrix

এখন এই Parameter update করার জন্য gradient descent ব্যবহার করবো

$\frac{\partial L}{\partial w_1^{(1)}}, \frac{\partial L}{\partial w_2^{(1)}}, \frac{\partial L}{\partial w_0^{(1)}} \ldots \ldots$ 9 dn

gradient

9-values init = 1 $\begin{bmatrix} \text{---} \\ \text{---} \\ \text{---} \end{bmatrix} =$

update using Gradient Descent

loop → 1000 epochs

$$w_1^{(1)} = w_1^{(1)} - \eta \frac{\partial L}{\partial w_1^{(1)}}$$

$$w_2^{(1)} = w_2^{(1)} - \eta \frac{\partial L}{\partial w_2^{(1)}}$$

$$\vdots$$

$$w_9^{(1)} = w_9^{(1)} - \eta \frac{\partial L}{\partial w_9^{(1)}}$$

Prediction

Monday, June 28, 2021 3:35 PM

$S_x = \{7, 70\} \Rightarrow Y, N, Opt$

$\boxed{m_1}$ Yes

$\boxed{m_2}$ No

$\boxed{m_3}$ Opt out

$w_1^{(1)} \; w_2^{(1)} \; w_0^{(1)}$

$w_1^{(2)} \; w_2^{(2)} \; w_0^{(2)}$

$w_1^{(3)} \; w_2^{(3)} \; w_0^{(3)}$

$Z_1 = 7 \times w_1^{(1)} + 70 \times w_2^{(2)} + w_0^{(1)}$

$Z_2 = 7 \times w_1^{(2)} + 70 \times w_2^{(2)} + w_0^{(3)}$

$Z_3 = 7 \times w_1^{(3)} + 70 \times w_2^{(3)} + w_0^{(3)}$

$$\sigma(y) = \frac{e^{z_1}}{e^{z_1} + e^{z_2} + e^{z_3}}$$

$= \underline{0.40}$

$$\sigma(N) = \frac{e^{z_2}}{e^{z_1} + e^{z_2} + e^{z_3}}$$

$\underline{0.35}$

$$e(0) = \frac{e^{z_3}}{e^{z_1} + e^{z_2} + e^{z_3}}$$

$\underline{0.25}$

Sigmoid Vs Softmax

Monday, June 28, 2021 3:47 PM

Code Sample

Monday, June 28, 2021 3:36 PM

Lecture: 78

Polynomial Feature in LoR

আমরা জানি, linear data এর ক্ষেত্রে আমরা, (LR & LOR) নিয়ে কাজ করি। যদি, আমাদের data non-linear হয় তাহলেও আমরা LoR apply করতে পারি কিছু modification করে।

$\boxed{\text{modification হচ্ছে degree}}$

Polynomial Logistic Regression

Monday, June 28, 2021 6:43 PM

input target

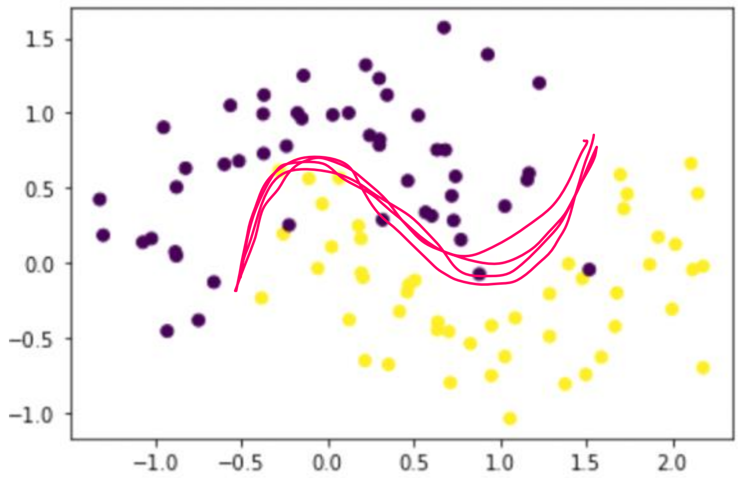
x_1 x_2 y $\{0,1\}$

প্রতিটার input এর জন্য তাকে Polynomial বানাবো।

$x_1 \to x^2$, x (w_1), x (w_2), (w_0)

১টা feature কে square করবো এটা value বা parameter আরো

degree



Linear

polynomial

degree = 2

x^0 x^1 x^2 x^3

x_1 x_2 → 2 cols

6 cols

x_1^0 x_1^1 x_1^2 x_2^0 x_2^1 x_2^2 y

w_0 w_1 w_2 w_3 w_6

feature-1 feature-2 Y

Lecture 79

Hyperparameter tuning in (LoR)