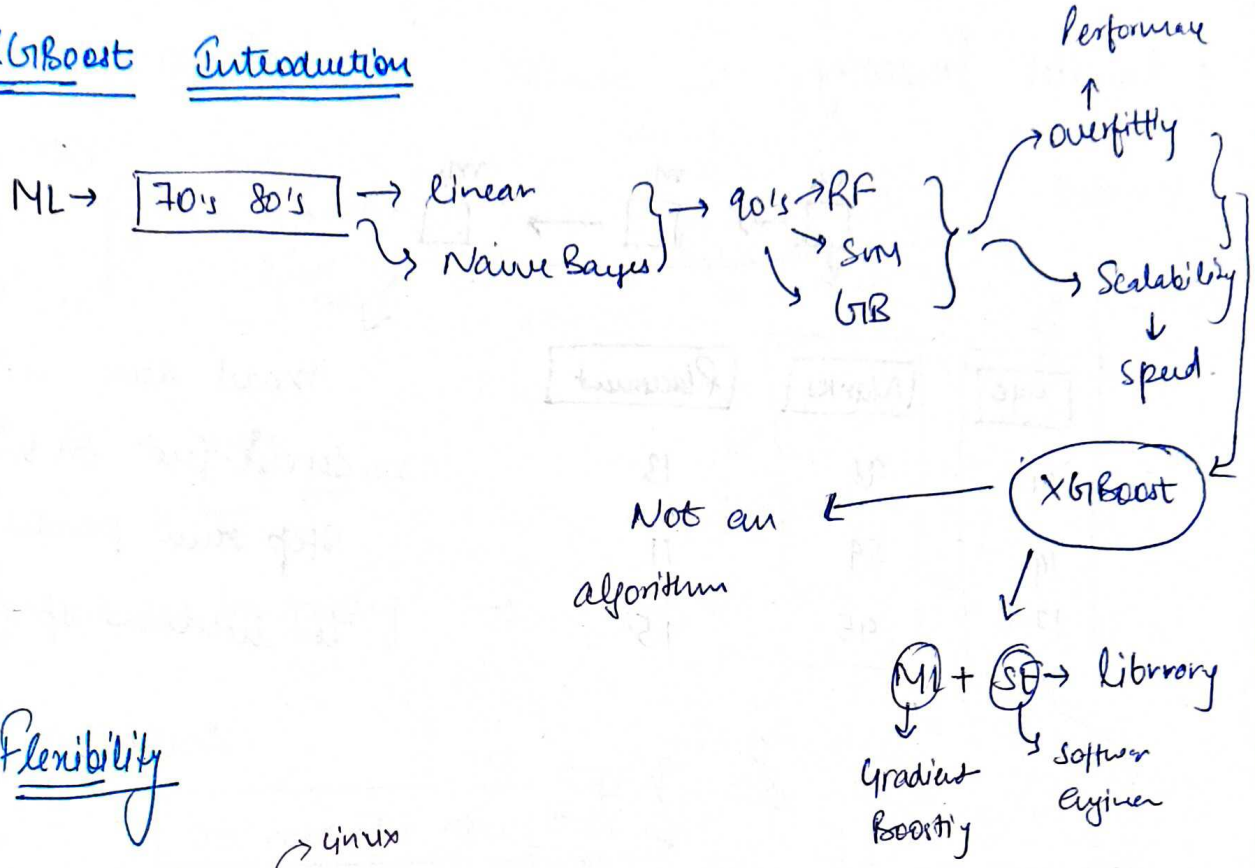
XGBoost

XGBoost Introduction



Flexibility

1. Cross platform → Linux, Window
2. Multiple language support → Python, R, matlab

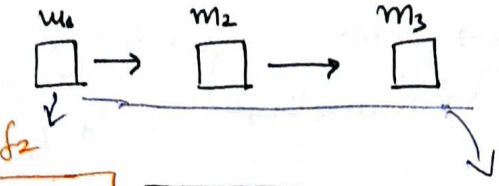
Python → API → JAVA

3. Integration with other libraries and tools → numpy/sklearn/pandas, model building, distributed → spark, model interpretible, model deployment → docker, work flow → mlflow

4. Support all kind of ML problems → Reg, Classification → Binary, Multi, Timeseries, Ranking

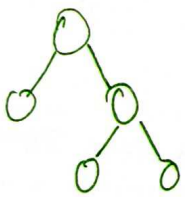
2. Speed

1. Parallel Processing



age	Marks	Placement
21	98	13
19	89	11
17	95	15

model run in sereise. but in model, step run parallelly to increase speed.



98, 89, 95 → f2 → f2 > 97

17, 19, 21 → 18, 20 → entropy / gini

f1 > 18

This process run parallely in XGBoost. So, speed increase.

2. Optimized Data Structures

cgpa	iq	placement
7	70	7
8	80	8

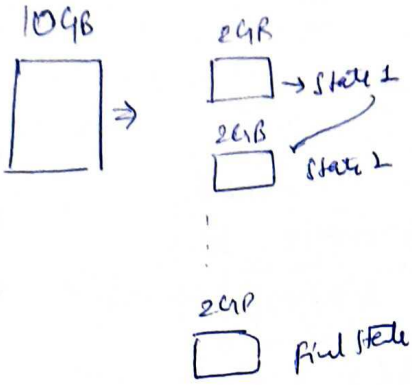
Row by Row

In optimized data structure → Step → Column by Column

Column block

3. Cache Awareness

4. Out of Core Computing



XgBoost out of core

↓

Tree method → hist → feature in libraries

↓

Cache

5. Distributed Computing → multiple machine

6. GPU support

tree_method = gpu_hist

Performance

1. Regularized learning objective

→ Linear reg → (L_1 & L_2)

→ Train → Test
✓ overfitting ✗

Gradient

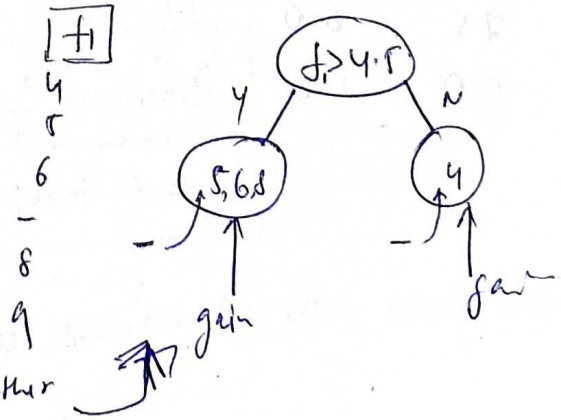
(L) + reg

learning rate

tree construction

2. Handling Missing Values

3. Sparsity Aware split finding



* Missing value one time in Y and another time in N and then check gain

4. Efficient Split Finding (Weighted Quantile Sketch + Approximate Tree Learning)

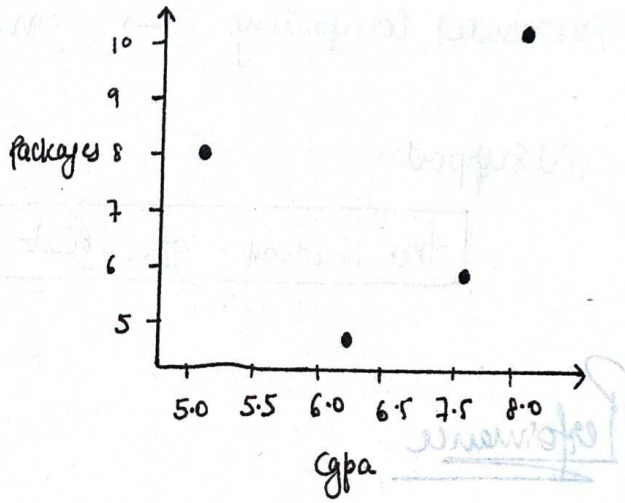
5. Tree Pruning

⇉ LightGBM

⇉ CatBoost

The Problem Statement

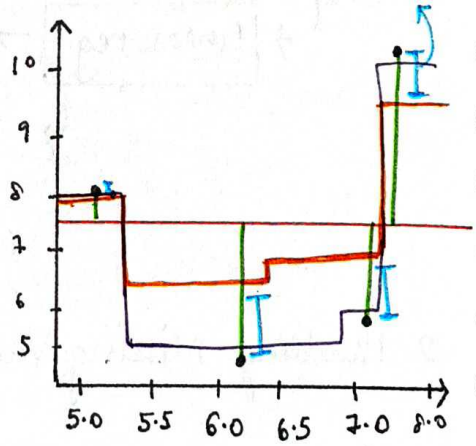
Cgpa	Packages
6.7	4.5
9.0	11.0
7.5	6.0
5.0	8.0



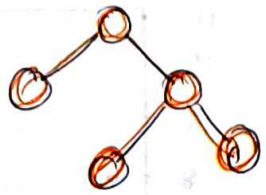
Solution Overview

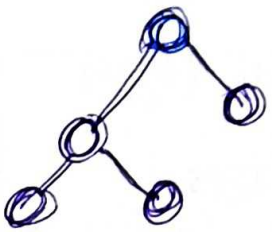
Cgpa	Package	$f_0(x)$	residual	$f_1(x)$	residual
6.7	4.5	7.5	-3	–	–
9.0	11.0	7.5	3.5	–	–
7.5	6.0	7.5	-1.5	–	–
5.0	8.0	7.5	0.5	–	–

New Residual



Stage I → 7.5 + Stage II + Stage III





decision Tree different from Gradient Boosting → Vanilla dt, gini/entropy criteria

XGBoost use different criteria

→ XGBoost follow same flow as Gradient boosting but diff criteria in dt.

Step by Step Calculation

Cgpa	Package	Model 1	residual1
6.7	4.5	7.3	-2.8
9.0	11.5	7.3	3.7
7.5	6.0	7.3	-1.3
5.0	8.0	7.3	0.7

Cgpa | residual1 → decision Tree

leaf node ← -2.8, 3.7, -1.3, 0.7

Calculate quantity to leaf node is Similarity Score in XGBoost.

* Same as we calculate gini impurity in Gradient boosting of DT.

$$SS = \frac{(\text{Sum of residual})^2}{\text{no. of residual} + \lambda}$$

Let $\lambda = 0$

$$= \frac{(-2.8 + 3.7 + 1.3 + 0.7)^2}{4 + 0}$$

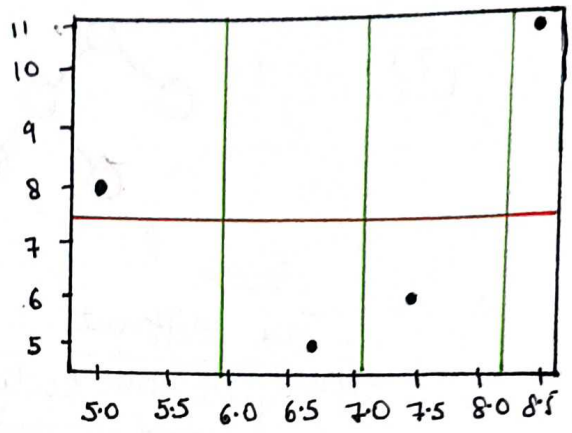
$= 0.2$ → residual similar to each other score

* We need to split residual with the help of cgpa column.

Ascending order (cgpa)

5.0] 5.85
6.7]
7.5] 7.1
9.0] 8.25

cgpa	residual 1
5.0	0.7
6.7	-2.8
7.5	-1.3
9.0	3.7



Splitting criteria (1) → 5.85

cgpa < 5.85

0.7 | -2.8, -1.3, 3.7

$SS_L = \frac{(0.7)^2}{1+0} = 0.49$

$SS_R = \frac{(-2.8+(-1.3)+3.7)^2}{3+0} = 0.05$

gain = $(SS_L + SS_R) - SS_{root}$

= (0.49 + 0.5) - 0.02

= 0.52 ← SS score increase

Splitting criteria (2) → 7.1

cgpa < 7.1

0.7, -2.8 | -1.3, 3.7

$SS_L = \frac{(0.7-2.8)^2}{2} = 2.20$

$SS_R = \frac{(-1.3+3.7)^2}{2} = 2.88$

gain = $(SS_L + SS_R) - SS_{root}$ ← SS score of all residual

= 2.20 + 2.88 - 0.02

= 5.06 ← SS score increase

Splitting criteria 3 → 8.25

cgpa < 8.25

0.7, -2.8, -1.3 | 3.7

$SS_L = 3.85$

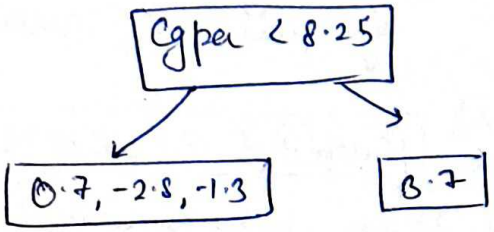
$SS_R = 13.69$

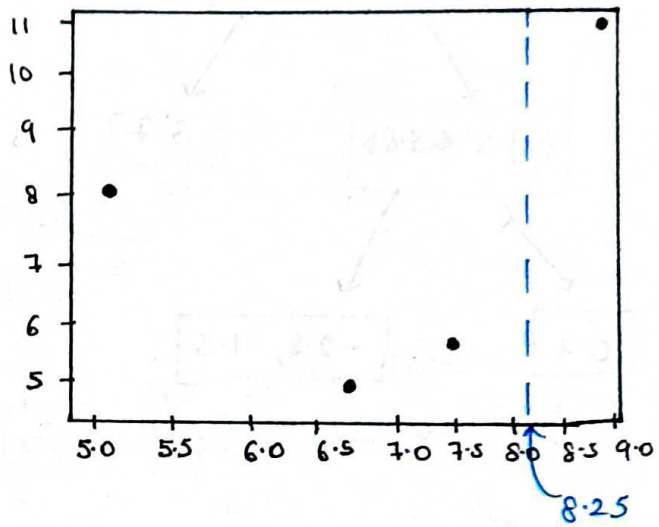
gain = $(SS_L + SS_R) - SS_{root}$

= 17.52 ← SS score increase

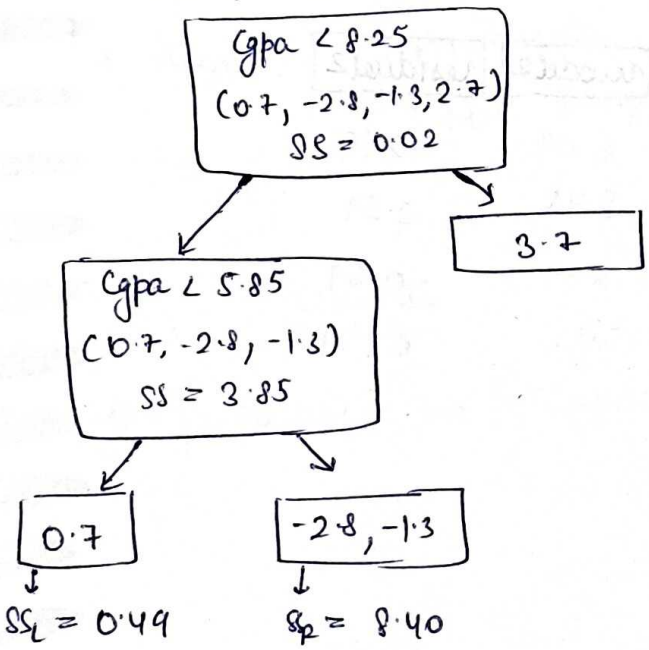
→ Take ~~Stage 2~~ Criteria 3 because higher SS score increase than other.

Stage 2 → decision tree





① Spliting Criteria → 5.85

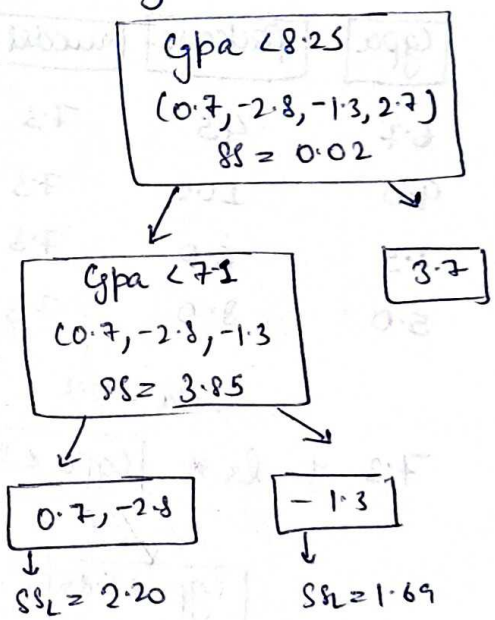


$SS_L = 0.49$ $SS_R = 8.40$

gain = (0.49 + 8.40) − 3.85
= 5.04

SS score accuracy

② Spliting Criteria → 7.1

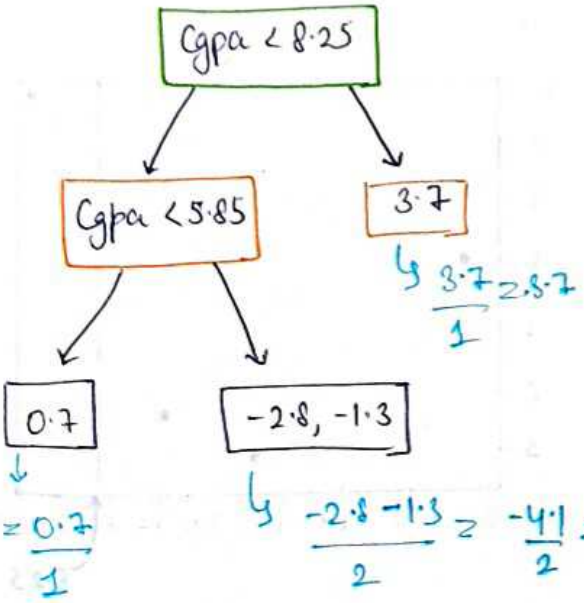


$SS_L = 2.20$ $SS_R = 1.69$

gain = 6.04

SS score accuracy

The final Decision Tree

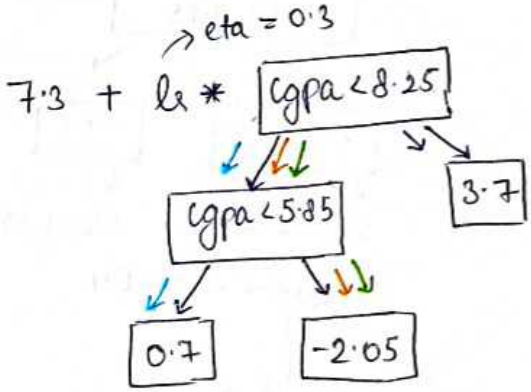


→ Now, we have to find out the output of all leaf nodes.

Output = sum of residual / no. of residual + λ

let λ = 0

Cgpa	Package	model 1	residual 1	model 2	residual 2
6.7	4.5	7.3	-2.8	6.69	-2.19
9.0	11.0	7.3	3.7	8.41	2.59
7.5	6.0	7.3	-1.3	6.69	-0.69
5.0	8.0	7.3	0.7	7.51	0.49



7.3 + 0.3 * (-2.05) = 6.69

7.3 + 0.3 * (3.7) = 8.41

7.3 + 0.3 * (-2.05) = 6.69

7.3 + 0.3 * (0.7) = 7.51

Stage ③

cgpa | res2

↳ decision Tree

→ if residual is going to near 0. then our dt is good

mean + lr * dt1 + lr * dt2

no_estimator → decide no. of tree

↳ Algo 1: Exact Greedy Algorithm for split finding (we cover this method) for small data set

↳ Algo 2 :- Approximate Algorithm for split finding (for large dataset)

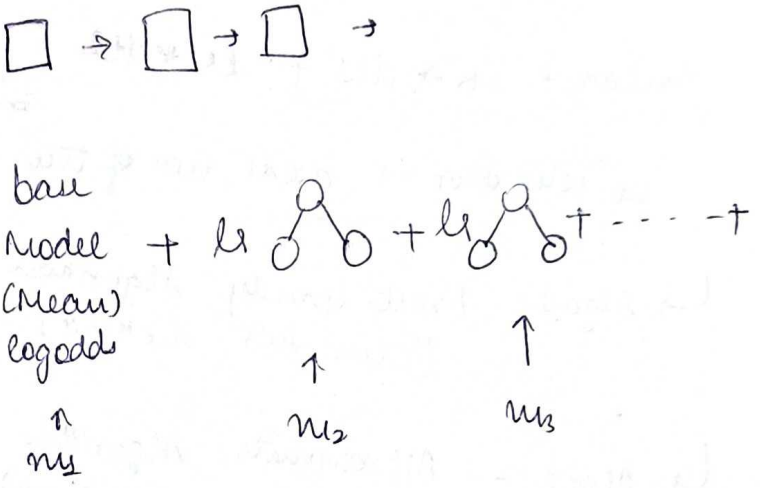
Research

cgpa	package	→	cgpa (f1)	12th mark (f2)	package

or categorical

XGBoost for Classification

cgpa	placed
5.70	0
6.25	1
7.10	0
8.15	1
9.60	1



xgboost → different decision tree { similarity score }

Step by Step Solution

cgpa	placed	Pred 1
5.70	0	0.405
6.25	1	0.405
7.10	0	0.405
8.15	1	0.405
9.60	1	0.405

Stage 1:

base estimator → mean

Log(odds) → $\log\left(\frac{p}{1-p}\right)$ p → prob of the class

$= \log\left(\frac{3/5}{2/5}\right) = \log\left(\frac{3}{2}\right) = 0.4$

$= 0.405$

0.405 → log(odds)

we have to change into probability

$$p = \frac{e^{\log(odds)}}{1 + e^{\log odds}} = \frac{e^{0.405}}{1 + e^{0.405}} = 0.60$$

gpa	placed	Pred 1	pred 1 (prob)	res1
5.70	0	0.405	0.6	-0.6
6.25	1	0.405	0.6	0.4
7.10	0	0.405	0.6	-0.8
8.15	1	0.405	0.6	0.4
9.60	1	0.405	0.6	0.4

Stage 1 → error, pseudo residual

m1 + ☐ → decision Tree

Xgboost

gpa	res1	pred 1 (prob)
5.70	-0.6	0.6
6.25	0.4	0.6
7.10	-0.6	0.6
8.15	0.4	0.6
9.60	0.4	0.6

leaf node [-0.6, 0.4, 0.6, 0.4, 0.4]

↳ similarity score

$$SS = \frac{(\sum \text{residual}_i)^2}{\sum \text{prev-prob}_i(1-\text{prev-prob}_i) + \lambda}$$

$$= \frac{(-0.6+0.4+(-0.6)+0.4+0.4)^2}{5 \times [0.6(1-0.6)] + 0}$$

$$= 0 \rightarrow \text{similarity score}$$

gpa

5.70] 5.97

6.25]

7.10] 7.62

8.10]

9.60] 8.87

Splitting criteria = 5.97

gpa < 5.97

-0.6 | 0.4, -0.6, 0.4, 0.4

$$SS_L = \frac{(-0.6)^2}{0.6(1-0.6)} = 1.5$$

$$SS_R = \frac{0.4-0.6+0.4+0.4}{4 \times 0.6 \times 0.4} = 0.37$$

$$\text{gain} = (SS_L + SS_R) - SS_{root}$$

$$= 1.87$$

Splitting Criteria = 6.67

Cgpa < 6.67

-0.6, 0.4 | -0.6, 0.4, 0.4

SSL = $\frac{(-0.2)^2}{2[0.6 \times 0.4]}$ = 0.08

SSR = $\frac{(0.2)^2}{3 \times 0.24}$ = 0.05

gain = 0.08 + 0.05 - 0

= 0.13

Splitting Criteria = 7.62

Cgpa < 7.62

-0.6, 0.4, -0.6 | 0.4, 0.4

= $\frac{(-0.8)^2}{3 \times 0.24}$ = 0.88

= $\frac{(0.8)^2}{2 \times 0.24}$ = 1.33

gain = 0.88 + 1.33 - 0

= 2.22

Splitting Criteria = 8.87 → Same steps for this decision tree.

Cgpa < 8.87

-0.6, 0.4, -0.6, 0.4 | 0.4

→ Splitting Criteria = 7.62 is a largest Similar Score

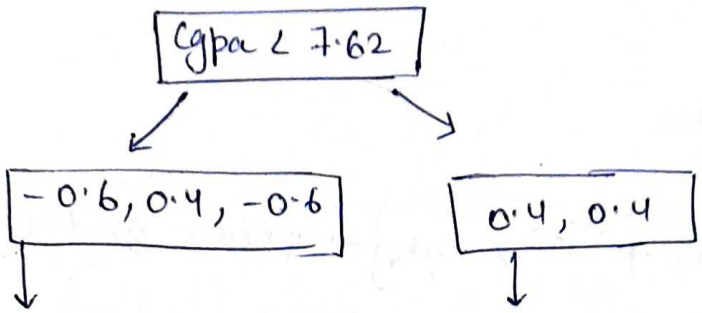
Stage 2 model

Cgpa < 7.62

-0.6, 0.4, -0.6 | 0.4, 0.4

* We can also derive or find next node of these leaf node

Stage 2 model



output = $\sum$ residual;

$\sum$ prev_probi (1 - prev_pro)

$= \frac{-0.6 + 0.4 + (-0.6)}{3 \times 0.6 \times 0.4} = \frac{-0.8}{3 \times 0.24} = \boxed{-1.11}$

$= \frac{0.4 + 0.4}{2 \times 0.24} = \boxed{1.66}$

Cgpa	Placed	Pred1 (log odds)	pred1 (prob)	res1	pred2 (log)	Pred2 (prob)	res2
5.70	0	0.405	0.6	-0.6	0.072	0.518	-0.51
6.25	1	0.405	0.6	0.4	0.072	0.518	0.48
7.10	0	0.405	0.6	-0.8	0.072	0.518	-0.51
8.15	1	0.405	0.6	0.4	0.903	0.712	0.28
9.60	1	0.405	0.6	0.4	0.903	0.712	0.28

learning rate

m1 + 0.3 × m2

log(odds) (decisionTree)

$0.405 + 0.3 \times$ [Cgpa < 7.62]

-1.11 | 1.66

$= 0.405 + 0.3 * (-1.11)$
$= 0.072$

$= 0.405 + 0.3 (-1.11)$
$= 0.072$

$= 0.405 + 0.3 \times 1.66$
$= 0.903$

$Prob = \frac{e^{log\,odds}}{1 + e^{log\,odds}}$

Stage 3

cgpa | res2

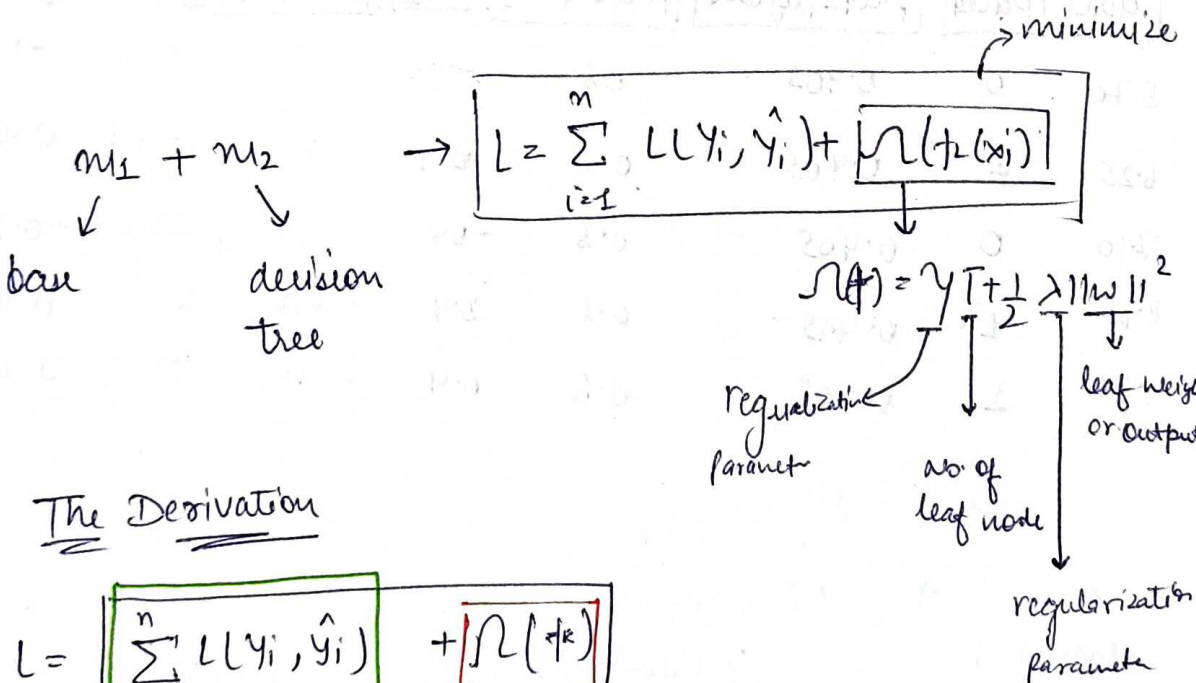
↳ decision Tree

$[m_1 + 0.3 \times m_2 + 0.3 \times m_3] \rightarrow$ pred → prod → res3

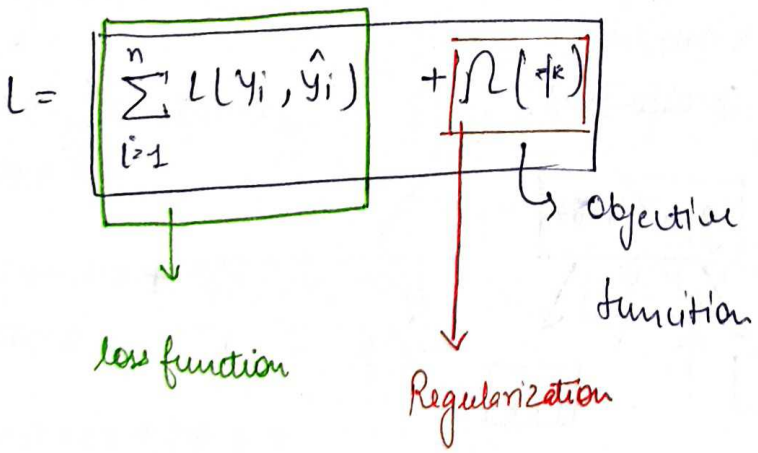
Maths of XGBoost

Loss function → regularization parameter

XGBoost loss function different from Gradient Boost



The Derivation



mean → m_1 → $f_1(x)$ — loss function → $\sum_{i=1}^{n} L(y_i, \hat{y}_i)$ [∵ we have only mean not any decision tree]

Case 1 →

Case 2 → mean m_1 → $f_1(x_i)$; m_2 → decision tree → $f_2(x_i)$ → $L = \sum_{i=1}^{n} L(y_i, f_1(x_i) + f_2(x_i)) + \Omega(f_2(x_i))$

$\hat{y}_i = f_1(x_i) + f_2(x_i)$

Case 3 → mean m_1 → $f_1(x_i)$; decision tree m_2 → $f_2(x_i)$; decision tree m_3 → $f_3(x_i)$

$L = \sum_{i=1}^{n} (y_i, f_1(x_i) + f_2(x_i) + f_3(x)) + \Omega(f_3(x_i))$

Case 4:-

mean m_1 → $f_1(x_i)$; decision tree m_2 → $f_2(x_i)$ ---- m_t → $f_t(x_i)$ → $L = \sum_{i=1}^{n} y_i, f_1(x_i) + f_2(x_i) \cdots f_t(x_i) + \Omega(f_t(x_i))$

↓ simplify

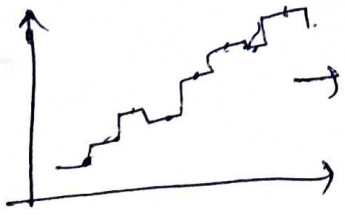
$L^{(t)} = \sum_{i=1}^{n} L(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t)$

At any stage t

$L^{(t)} = \sum_{i=1}^{n} L(y_i, (\hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t(x_i))$

→ we have to the value of $f_t(x_i)$ which reduce value of $L^{(t)}$.

but it is difficult to reduce the value because in XGBoost we use decision tree and decision tree plot →



non-differentiable

* we have to smooth the line

Solution of the problem → we use taylor Series

Complex function → approx → polynomial

Applying Taylor Series

$$f(x) \simeq f(a) + f'(a)(x-a) + \frac{f''(a)}{2!}(x-a)^2 \cdots$$

$$\alpha^{\langle t \rangle} = \left[\sum_{i=1}^{n} L\left(y_i, \hat{y}_i^{\langle t-1 \rangle} + f_t(x_i)\right) \right] + \lambda(f_t(x_i))$$

$$x = \hat{y}_i^{\langle t-1 \rangle} + f_t(x_i) \qquad a \to \hat{y}_i^{\langle t-1 \rangle}$$

$$f(a) = \sum_{i=1}^{n} L\left(y_i, \underbrace{\hat{y}_i^{\langle t-1 \rangle} + f_t(x_i)}_{x}\right)$$

↳ replace with a

$$f(a) = \sum_{i=1}^{n} L\left(y_i, \hat{y}_i^{\langle t-1 \rangle}\right)$$

Gradient (g_i)

$$f'(a)(x-a) = \sum_{i=1}^{n} \frac{\partial L\left(y_i, \hat{y}_i^{\langle t-1 \rangle}\right)}{\partial \hat{y}_i^{\langle t-1 \rangle}} f_t(x_i)$$

$$\frac{f''(a)}{2}(x-a)^2 = \sum_{i=1}^{n} \frac{\partial^2 L\left(y_i, \hat{y}_i^{\langle t-1 \rangle}\right)}{\partial \hat{y}_i^{\langle t-1 \rangle 2}} f_t^2(x_i)$$

↳ Hessian

$$L^{(t)} = \sum_{i=1}^{n} \left[L(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t(x_i))$$

$$L^{(t)} = \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t(x_i))$$

↳ expand ↳ t leaf node

$$L^{(t)} = \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \gamma T + \frac{1}{2}\lambda \left[w_1^2 + w_2^2 + w_3^2 + \cdots + w_T^2 \right]$$

$$L^{(t)} = \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

$i = i+n$; $j = 1-T$

So, there are 2 summation. We have make a common summation.

$$L^{(t)} = \sum_{j=1}^{T} \left[\sum_{i \in I_j} g_i w_j + \frac{1}{2} \sum_{i \in I_j} h_i w_j^2 \right] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

* Explaining How we make a common summation

$w_j = f_t(x_i)$

$\sum_{i=1}^{n} \rightarrow \sum_{j=1}^{T} \sum_{i \in I_j}$

$$L^{(t)} = \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

$$L^{(t)} = \sum_{j=1}^{T} \left[\sum_{i \in I_j} g_i w_j + \frac{1}{2} \sum_{i \in I_j} h_i w_j^2 \right] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2$$

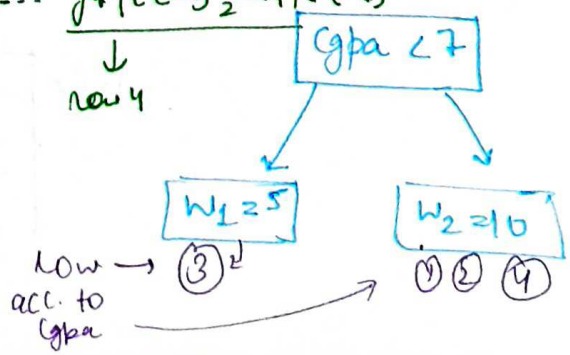
cgpa	package
7.5	4.5 → g_1
8.2	6.9 → g_2
6.4	5.1 → g_3
9.1	10 → g_4

expand all row one by one

→ $\left(g_1 f_t(x_3) + \frac{1}{2} h_1 f_t^2(x_1)\right) + \left(g_2 f_t(x_1) + \frac{1}{2} h_1 f_t^2(x_1)\right) \cdots g_4 f_t(x_4) + \frac{1}{2} h_4 f_t^2(x_4)$

expan

↓ row 1 ↓ row 2 ↓ row 4



$j=1$, $j=2$

$$g_3 ft(x_3) + \frac{1}{2} h_3 ft^2(x_3) + g_1 ft(x_1) + \frac{1}{2} h_1 ft^2(x_1) + \cdots\cdots g_4 ft(x_4) + \frac{1}{2} h_4 ft^2(x_4)$$

(row 3) (row 1) (row 2) (row 4)

In this method, we can also cover all.

$ft(x_i) \rightarrow$ Output of decision Tree (w_i)

So, we can write

$$g_3 ft(x_3) + \frac{1}{2} h_3 w_1^2 + g_1 ft(x_1) + \frac{1}{2} h_1 (w_2)^2 + \cdots\cdots$$

(row 3) (row 1) (row 4)

$$L^{(t)} = \sum_{j=1}^{T} \left[\sum_{i \in I_j} g_i w_j + \frac{1}{2} \sum_{i \in I_j} h_i w_j^2 \right] + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2$$

We have to simplify this ↑

$$L^{(t)} = \sum_{j=1}^{T} \left[\sum_{i \in I_j} g_i w_j^2 + \frac{1}{2} \sum_{i \in I_j} h_i w_j^2 + \frac{1}{2} \lambda w_j^2 \right] + \gamma T$$

$$L^{(t)} = \sum_{j=1}^{T} \left[\sum_{i \in I_j} g_i w_i + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) w_j^2 \right] + \gamma T$$

$$\frac{\partial L^{(t)}}{\partial w_j} = \sum_{i \in I_j} g_i + \left(\sum_{i \in I_j} h_i + \lambda \right) w_j = 0$$

$$w_j = \boxed{\frac{-\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}}$$

Output value for Regression

$$w_j = \frac{-\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}$$

λ → Regularization

$$g = \frac{\partial L(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}} \qquad h = \frac{\partial^2 L(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)2}}$$

Reg → mse → $L = \frac{1}{2}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2$

Loss → Reg, Classi

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

$g_i = \sum_{i \in I_j} (y_i - \hat{y}_i^{(t-1)})$ → sum of residual

↳ residual for every point

$$h_i = \frac{\partial}{\partial y_i} \Rightarrow \frac{\partial L}{\partial \hat{y}_i} = \frac{\partial}{\partial \hat{y}} (\hat{y} - y_i) = 1$$

$$w_j = \frac{\sum_{i \in I_j} (y_i - \hat{y}_i^{(t-1)})}{\sum_{i \in I_j} (1) + \lambda} = \frac{\text{Sum of residual}}{\text{no. of residual} + \lambda}$$

if no. of residual 2 then 2

$$\boxed{w_j = \frac{\text{Sum of residual}}{\text{no. of residual} + \lambda}}$$

For Regression.

Output value for Classification

$$w_j = \frac{-\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \qquad g = \frac{\partial L(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}} \qquad h = \frac{\partial^2 L(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)2}}$$

Loss – log loss → gradient boosting

$$w_j = \frac{\text{Sum of residual}}{\sum_{i \in I_j} p_i(1-p_i) + \lambda}$$

p_i → Pred Prob of previous timestep

Derivation of Similarity Score

$$L^{(t)} = \sum_{j=1}^{t} \left[\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) w_j^2 \right] + \gamma T$$

$$w_j = \frac{-\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}$$

$$L^{(t)}(q) = \sum_{j=1}^{T} \left[\left(\sum_{i \in I_j} g_i \right) \frac{\left(-\sum_{i \in I_j} g_i \right)}{\sum_{i \in I_j} h_i + \lambda} + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) \frac{\left(\sum_{i \in I_j} g_i \right)^2}{\left(\sum_{i \in I_j} h_i + \lambda \right)^2} \right] + \gamma T$$

$$L^{(t)}(q) = \sum_{j=1}^{T} \left[\frac{-\left(\sum_{i \in I_j} g_i \right)^2}{\sum_{i \in I_j} h_i + \lambda} + \frac{1}{2} \frac{\left(\sum_{i \in I_j} g_i \right)^2}{\sum_{i \in I_j} h_i + \lambda} \right] + \gamma T + \gamma T$$

$$= \sum_{j=1}^{T} \left[\frac{-1}{2} \frac{\left(\sum_{i \in I_j} g_i\right)^2}{\sum_{i \in I_j} h_i + \lambda} \right] + \gamma T$$

↳ No. of ~~leaf~~ Node. parent / child

$$\alpha^{(t)}(q) = -\frac{1}{2} \sum_{j=1}^{T} \left[\frac{\left(\sum_{i \in I_j} g_i\right)^2}{\sum_{i \in I_j} h_i + \lambda} \right] + \gamma T$$

we have replace gini/entropy

reg

↓

$g \rightarrow$ sum of residual

$h \rightarrow \sum N \rightarrow$ No. of residual

classi

↓

$g \rightarrow$ residual

$h = p_i (1 - p_i)$