09-11-2025

→ Code implementation → DT & RF → classification

DT ↓ regression

RF → regression

→ gradient Boosting (concept)

→ XG Boost (concept) + code (if time permits)

Gradient Boosting (Regression)

→ Decision Tree ⟶ Random Forest ⟶ Bagging ⟶ non-linear data

↳ use full-depth trees → (strong models) → parallely

→ models are parallel

majority → o/p

Adaboost →

m100

→ Stump (weak learner)

→ level - 1

weak learners ensembled in sequence

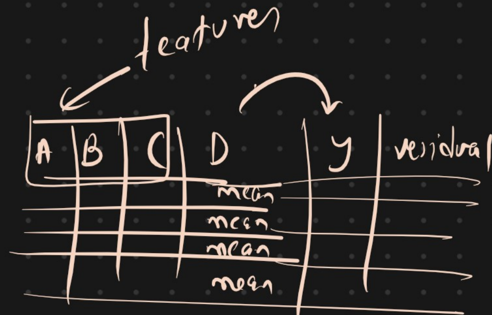
XGBoost

extreme Gradient Boosting

f(0) → mean

Gradient Boosting

→ Dataset
- → target column → mean → $f(0)$
- → create a new column for $f(0)$ predict (mean)
- → compute residuals : $r_i = y_i - F(0)$
- → for each candidate split (features)
  - → for each midpoint
    - → mean of left
    - → mean of right
    - → sum of mean square (SSE)
- → choose split with lowest SSE
- → $h1(x)$



$$F(x) = F(0) + h1(x) + h2(x) + h3(x)$$

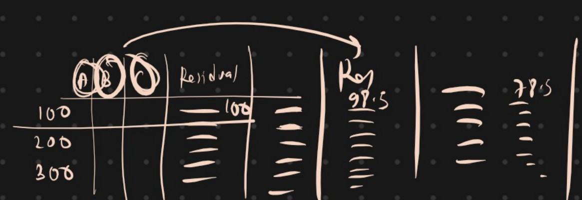
F(0) ↓ mean

h1(x) ↓ DT ↓
- → stump
- → level-2, 3

A < 150 → SSE
A < 250 → SSE
> low → A < 150

B < x → SSE
B < y → SSE
> low → SSE

C ⟶ SSE



A < 150, 250

100 → mean²

200, 300 → mean²

sum

SSE

$\boxed{A < 150}$

Dataset

ID	SIZE	BEDS	price (y)
1	800	2	120
2	900	2	130
3	1000	3	150
4	1100	3	170
5	1200	4	200

learning-rate : lr : 0.1

step 0 : initialize model F(0)

$$f(0) = \text{mean}(y) \rightarrow 120 + 130 + 150 + 170 + 200/5 = 154$$

step 1 : compute residual : y - F(0)

ID	SIZE	BEDS	price (y)	prediction (F(0))	r1
1	800	2	120	154	-34
2	900	2	130	154	-24
3	1000	3	150	154	-4
4	1100	3	170	154	16
5	1200	4	200	154	46

step 2 : Candidate splits

: size

: size split: [850, 950, 1050, 1150]

: size < 1050 (calculation)

A ↙ L ↘ B R

800, 900, 1000 → [-34, -24, -4] → $-62/3 = -20.66$

1100, 1200 → [+16, +46] → $62/2 \rightarrow 31.0$

$$A(L) : SSE : (-34-(-20.66))^2 + (-24-(-20.66))^2 + (-4-(-20.66))^2 \rightarrow 466.66$$

$$B(R) : SSE \rightarrow (16-31)^2 + (46-31)^2 \rightarrow 450.$$

Total SSE = SSE(A) + SSE(B)
= 916.66

Feature	Threshold	SSE
Size	850	2675
size	950	1316.66
size	1050	916.68 → Smallest SSE
size	1150	1475.0
Beds	2.5	1316.66
Beds	3.5	1475.0

$\boxed{\text{Size} < 1050}$

depth-level - 1
(stump)

→ 2 leavel

800, 900, 1000
↓ mean(L)
$\boxed{-20.66}$

1100, 1200
↓ mean(R)
$\boxed{31.0}$

$$F1(x) = F(0) + lr \times h1(x)$$

$$= 154 + 0.1 \times \text{left} \to -20.66 \to$$

$$\quad\quad\quad\quad\quad\; \times \text{right} \to 31 \to$$

ID	SIZE	BEDS	Price (y)	prediction (F(0))	r1	F1	r2 (y - F1)
1	800	2	120	154	-34	151.93	-31.93
2	900	2	130	154	-24	151.93	-21.93
3	1000	3	150	154	-4	151.93	-1.93
4	1100	3	170	154	16	157.10	+12.90
5	1200	4	200	154	46	157.10	+42.90

$F1(ID(1)) \to 154 + 0.1 \times (-20.66) \to$

$154 - 2.066 \longrightarrow 151.93$

$F2(ID(1)) \to 154 + \underline{0.1} \times (-20.66) + \underline{0.1} \times h2(x) \to$

$h1(x)$

$h2(x)$

$h3(x)$

$h4(x)$

$$F(x) \rightarrow f(0) + h1(x) + \dots\dots\ h10(x)$$ if 10 tree selected.

→ each iteration creates a new stump (DT with depth - 1)

→ weak learner

---

pruning: removing split that does not reduce loss enough.

Xgboost

: stronger version of gradient Boosting

Advantages:

- → cross-validation split
- → gradient (1st order derivate)
- → Hessians (2nd order derivate)
- → has regularization (λ, α, γ)
- → parallelism

Formula:

$$w = -\frac{G}{H + \lambda} \rightarrow -\frac{\text{sign}(G) \cdot \max(|G| - \alpha, 0)}{H + \lambda}$$

$$\text{Gain} = \text{Score}_L + \text{Score}_R - \text{Score}_P - \gamma$$

ID	SIZE	BEDS	price (y)	prediction (F(0))
1	800	2	120	154
2	900	2	130	154
3	1000	3	150	154
4	1100	3	170	154
5	1200	4	200	154

Step 1: initialize:

$F(0) = \text{mean}(y) \rightarrow 154$

Step 2: for each possible split:

Step 2.1 → compute

L: G_L, H_L

R: G_R, H_R

Step 2.2: Calculate Gain, $\gamma = 0$

Gain: Score(L) + Score(R) − Score(parent)

: Score $= -\dfrac{G^2}{H+\lambda}$ ⟶ xgboost formula reference ⟶ $\dfrac{1}{2}\left[\dfrac{G_L^2}{H_L+\lambda} + \dfrac{G_R^2}{H_R+\lambda} - \dfrac{G^2}{H+\lambda}\right] - \gamma$

Step 2.3: Select the split with Highest gain

Step 2.4: For each leaf compute final leaf weight:

$$w = -\frac{G}{H+\lambda}$$

Step 2.5: update prediction:

$$F(x) = F(0) + \eta \cdot f_i(x)$$

$\eta = 0.1$ (ideal)

$\eta = 1$ (simple calculation)

ID	SIZE	BEDS	Price (y)	prediction ($F(0)$)	$g: \hat{y} - y$	h
1	800	2	120	154	34	1
2	900	2	130	154	24	1
3	1000	3	150	154	4	1
4	1100	3	170	154	-16	1
5	1200	4	200	154	-46	1

step 1: initialization → $\hat{y}_i = 154$

step 2: Build Tree 1

split : size < 950 (Threshold) → $\lambda = 1$

L: 800, 900 → 34, 24 → 58 (g); 1, 1 → 2 (h)

R: 1000, 1100, 1200 → 1, 1, 1 → 3 (h); 4, -16, -46 → -58 (g)

$$\text{Score}(L) = \frac{-58^2}{2+1} = -1121.33$$

$$\text{Score}(R) = \frac{-(-58)^2}{3+1} = -841$$

parent:

$g = 34 + 24 + 4 - 16 - 46 = 0$

$h = 5$

$$\text{Score}(p) = \frac{-0^2}{5+1} = 0$$

$$\text{Gain} = \text{Score}_L + \text{Score}_R - \text{Score}_p - \gamma$$

γ controls pruning, turning it off creates more deeper trees, more expressive tree

$$\text{Gain} = -1121.33 - 841$$

$$= -1962.33$$

Split	Gain
Size < 850	1156
Size < 950	1962.33 → Highest gain
Size < 1050	707
Size < 1150	1156
Beds ≤ 2.5	1156
Beds ≤ 3.5	707

Step : leaf weights

$$w = -\frac{G}{H+\lambda}$$

Size < 950

L: $= -\frac{58}{3} = -19.33$

R: $-\frac{(-58)}{4} = 14.5$

ID	SIZE	BEDS	Price (y)	prediction (f(0))	$g: \hat{y}-y$	h	w	new $\hat{y}$
1	800	2	120	154	34	1	-19.33	134.67
2	900	2	130	154	24	1	-19.33	134.67
3	1000	3	150	154	4	1	14.5	168.5
4	1100	3	170	154	-16	1	14.5	168.5
5	1200	4	200	154	-46	1	14.5	168.5

$$\text{new } \hat{y} = f(0) + \eta(w)$$

→ GB → class

→ XGB → classification + practice