XGBoost

— install

Bagging Boosting

Base Models

— High Bias & Low Variance

XG Boost.

Extreme ← → Boosting

Gradient

— pseudo residuals

— Parallelized the execution

— Row sampling / Column sampling - (feature sampling)

$\hat{y} = f(x)$ → Model.

$(y - \hat{y})^2$ → Any loss

$-\frac{\partial L}{\partial \hat{y}}$

⇒ -ve gradient of the loss fn.

⇒ pseudo residual

$\text{loss}_{(L)} = (y - \hat{y})^2$ → Squared.

$\frac{\partial L}{\partial \hat{y}} = -2(y - \hat{y})$

$(y - \hat{y}) = -\frac{1}{2} \partial L / \partial \hat{y}$

$(y - \hat{y}) \propto -\partial L / \partial \hat{y}$

$\hat{y} = h_0(x) + \alpha_1 h_1(x) + \alpha_2 h_2(x) + \cdots$

$h_0(x)$ → Rough prediction

$\alpha_1 h_1(x)$ → Error from prev. stage.

$\alpha_2 h_2(x)$ → Error from prev. to prev. stage

Drug Dosage (x)	Efficiency (y)	$\hat{y}$	ε_0	$\hat{\varepsilon}_0$	ε_1
12	−10	0.5	−10.5	−2.65	−7.85
15 ← 18	7	0.5	6.5	+2.60	+3.90
22 ← 27	8	0.5	7.5	+2.60	+4.90
30 ← 33	−7	0.5	−7.5	−1.75	5.75

$h_0(x) = 0.5$

$h_1(x)$

4.0 [−10.5, 6.5, 7.5, −7.5]

→ Dosage ≤ 15

yes — ① [−10.5] → 110.25 ↓ −10.5

No — 3 residuals [6.5, 7.5, −7.5] → 14.08

Dosage ≤ 30

② [6.5, 7.5] → 7.0 ; [−7.5] → −7.5

③

Similarity Score = $\dfrac{(\Sigma \text{residuals})^2}{n + \lambda}$ → λ ↳ Regularization parameter

Parent Node: $\dfrac{(-10.5 + 6.5 + 7.5 - 7.5)^2}{4}$

$= (-4)^2/4 = 4$

Left Node: $\dfrac{(-10.5)^2}{1} = 110.25$

Right Node: $\dfrac{(6.5 + 7.5 - 7.5)^2}{3} = 14.08$

Gain = Left Sim. + Right Sim. − Parent Sim.

$= 110.25 + 14.08 - 4.0$

$= 110.25 + 10.08$

$= 120.33$

$\alpha_1 = 0.3$

$\Rightarrow h_0(x) + \alpha_1 h_1(x)$

$\Rightarrow 0.5 + 0.3 \times (-10.5)$

$\Rightarrow 0.5 + -3.15$

$\Rightarrow -2.65$

$= 0.5 + 0.3 \times 7.0$

$= 0.5 + 2.1$

$= 2.60$

$= 0.5 + 0.3(-7.5)$

$= 0.5 - 2.25$

$= -1.75$

$h_2(x) \Rightarrow \alpha_2$

$-7.85, +3.90, 4.90, 5.75$

Dosage ≤ 22

$-7.85, 3.90$

$(-7.85 + 3.90)/2$

$4.90, 5.75$

$(4.90 + 5.75)/2$

$$y = h_0(x) + \alpha_1 h_1(x) + \alpha_2 h_2(x) + \cdots$$

$\log(\text{odds}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots$

$y_i =$

$\leftarrow g(z)$ $\rightarrow \log(\text{odds})$

$$g(z) = \frac{1}{1+e^{-z}} = p$$

$$\text{odds} = \left[\frac{p}{1-p}\right]$$

CatBoost → Categorical.

LightGBM.

Pink:
- Histograms/binning ✓ → Reduces no. of rows
- Effective Feature Bundling → Reduces no. of Columns
- Goss. (Gradient Based one Side Sampling) — Reduces no. of rows.

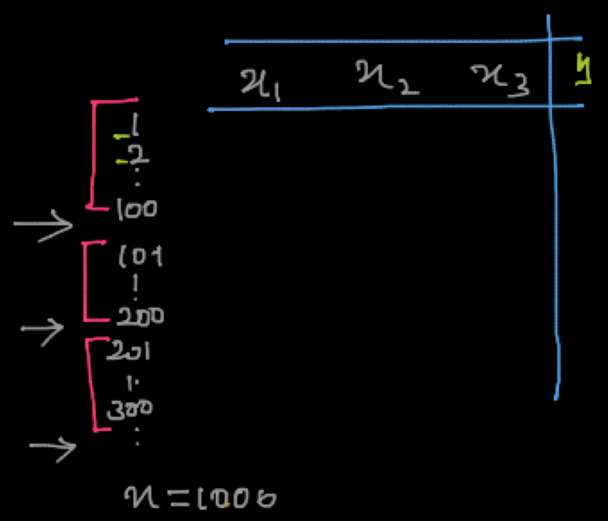
EFB — Categorical.

F_1	F_2
1	0
2	0
3	0
0	10
0	20
0	30

F
1
10

$f_1=1$; $f_2=0$.

$f_1=0$ $f_2=10$.



10 → Bins → 10 Split
Each → 100

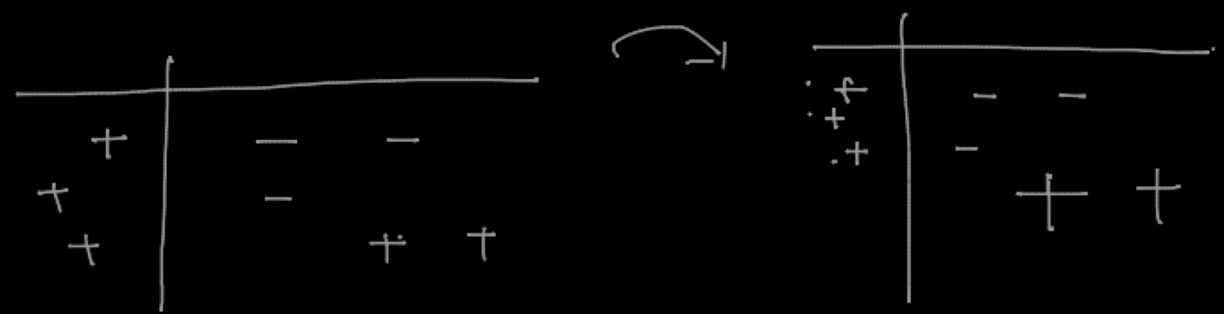
← 3-feature Sampling →

outlook f_1	Humidity f_2	Temp f_3	Windy f_4
→	→		
→	→		
→	→		
→	→		

Drug x.	Efficiency y.
→	

10.20 am.

Goss → Gradient Based one Sided Sample.



$n = 1000$

400 → Correctly → 100

600 → Wrongly classify → 600

$h_1(x)$

Gradient (Error) → High — Wrong
→ less — Correctly.

(700) → 1

$h_2(x)$

100 → 80
520

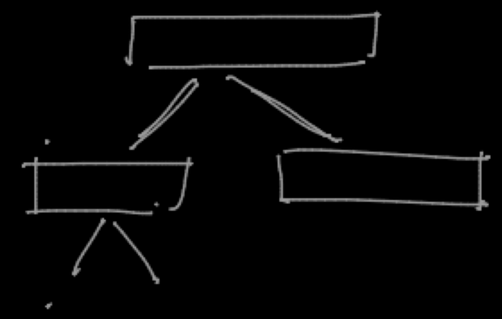
520
80
600

$h_3(x)$

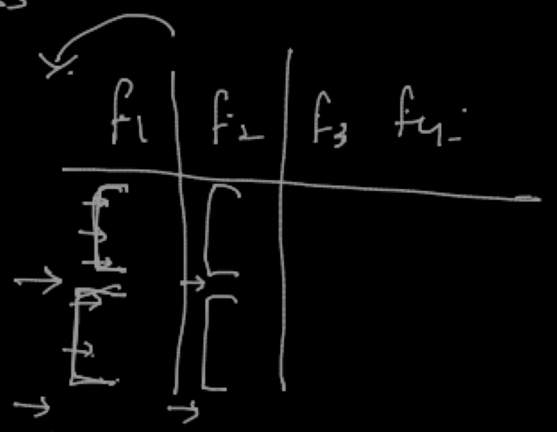
XGBM → LGBM.

- Feature samp → Histograms
- → parallelizi. → EFB
- → Goss

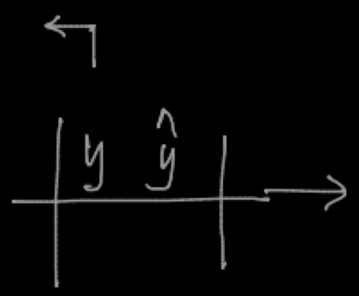
Gbdt:



Sub-features



Confusion Matrix.



Metrics

Accuracy

Recall / precision

f1-score

go

[500 → 1000.

classification_report

Categorical.
— Nominal.
— Ordinal.
— discrete data.
→ 'G', 'M' Numbers
OHE
Label En
Rati.
1
2
3
4
5
Category.

Parse dates
• CSV.
2, 3.5, 1990-1201
2, :), 2000-01-0,
→ Python →
— → Object
— int32

1 0 0
0 1 0
OHE → class ( )
pd.get_dum( )
CM

float.

categorical (object)

Date (object)

integer int32.

C1	C2	C3	C4
2.0	M,	2020-01-01,	4
3.0,	F,	1998-01-02,	5
4.5			
:			
6.6	↓	↓	↓
7.5			

(object) object obj.

↓ ↳ Parse-

↳ Encode

float → as. NaN

M
F
M
F.

↳ Encode.

↳ ONE →
→ label →