11-OCT-2025

Agenda:

Logistic Regression

Implementation of LR

ROC, AUC

Cross-validation

logistic regression: $y = mx + c$

→ classification

Activation f^n / logistic function → sigmoid → input range

→ $-\infty$ to ∞

→ output range → 0 to 1

$z = mx + c$ → sigmoid → 0, 1 } threshold > 0.5 → 0, 1

$z = mx + c$

height → short → 0 } binary
→ tall → 1

sigmoid $(z) \rightarrow \hat{y} \rightarrow$ prob → 0.2 → prob of tall is 20%.
→ 0.8 → prob of tall is 80%.

Linear regression.

$y = 80$
$\hat{y} = 91$ } mse → $(y - \hat{y})^2$

logistic regression.

$z = 80$
sigmoid$(z) \rightarrow 0.8 \leftarrow \hat{y}$ → 0 − 1
$1 \leftarrow y$ } mse will not work
↓
0 1

To solve this prob, we need a different cost function.

Modified MSE (Linear Regression)

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

Logistic Regression:

$$J(\beta) = -\frac{1}{m}\sum_{i=1}^{m}\left(y_i \cdot \log(\hat{y_i}) + (1-y_i)\cdot \log(1-\hat{y_i})\right)$$

cost function =

$m=1, \; y=0, \; y\_pred = 0.9$ (case 1)

$$= -\left(y_i \cdot \log(\hat{y_i}) + (1-y_i)\cdot\left(\log(1-\hat{y_i})\right)\right)$$

$$= -\left[0 \cdot \log(0.9) + (1-0)\cdot\left(\underline{\log(1-0.9)}\right)\right)$$

$$= -\left[0 + 1 \cdot \log(0.1)\right]$$

$$= -\log(0.1)$$

$$= -(-2.3025)$$

$$= 2.3025$$

$$\boxed{MSE = 0.81}$$

$m=1, \; y=0, \; y\_pred = 0.1$ (case 1)

$$= -\left[0 \cdot \log(0.1) + (1-0)\cdot \log(1-0.1)\right]$$

$$= -\left[0 + 1 \cdot \log(0.9)\right]$$

$$= -\log(0.9)$$

$$= -(-0.10536)$$

$$= 0.10536$$

case 1 $(y=0,\ y\text{-pred}=0.9)$: cost $\approx$ 2.3025 (High cost)

case 2 $(y=0,\ y\text{-pred}=0.1)$: cost $\approx$ 0.10536 (low cost)

Flow of logististic regression.

1) x_1, x_2, x_3, x_4 $\longrightarrow$ linear equation for intermediate o/p $\rightarrow z$
2) $z \rightarrow$ sigmoid $\rightarrow \hat{y}$
3) loss-function gradient $(y, y\text{-pred}) \rightarrow$ update weight/parameters.
4) calculate final loss using loss/cost function.
5) Iterate until satisfice
6) Done

weight/parameters $\downarrow$ m, β $\downarrow$ learable parameters

Out of context topic.

Data card:

- → info about data
- → sources
- → rows & columns
- → Features → info → range of each feature, its type
- → Missing values

Model card:

- → performance
- → link of test set on which we did evaluation
- → sklearn
- → confusion matrix
- → date

imputation = [1, 2, 3, 4, 6]

numerical = [1, 2, 3, 4, 5, 6]

transformer = [ ] → object of sklearn pipeline

if imputation:

① pipeline → imputation on imputation list

→ scaling on imputation list

append ① to transformer list

② [if numerical is not present in] → [5]

pipeline → scaling → [5]

else:

pipeline → scaling → [numerical (col)]

1, 2, 3, 4, 5, 6, 7, 8, 9 remainder = "drop")

imputation + scaling

scaling

transform()

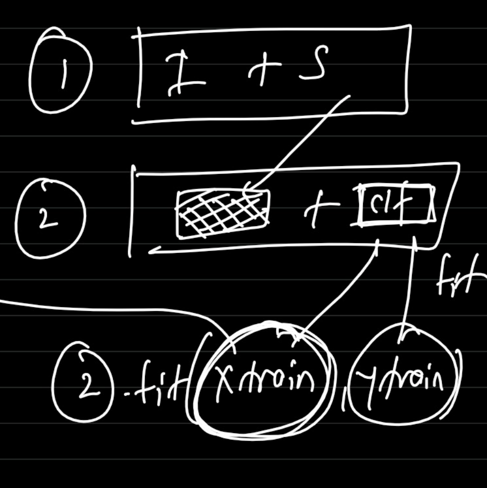
remainder = "passthrough"

→ 1, 2, 3, 4, 5, 6, 7, 8, 9

X train → imputation
X train → scaling

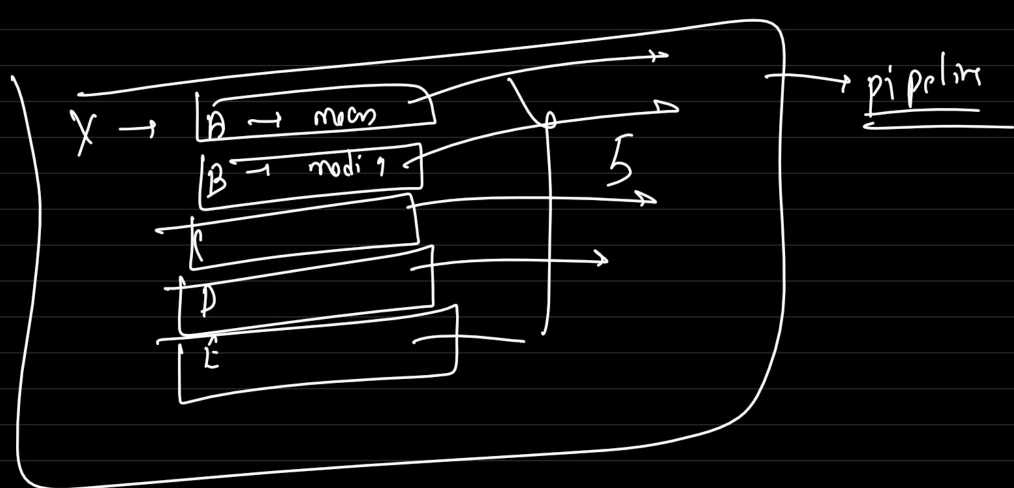
clf = logistic Reg

clf.fit(X train, y-train)



fit

transform



predict

$$z = X\_train.dot(weights) + bias$$

$$p = sigmoid(z)$$

$$\hat{y} = (p >= 0.5)\ 1\ else\ 0$$