

a) Data Preprocess

b) EDA, Stats, Assumptions!

c) Split the dataset: 80-20 or 70-30!

d) Basic:

Logistic regression is a statistical method used for binary classification problems. It predicts the probability of an outcome that can only be one of two classes, typically labelled as 0 or 1.

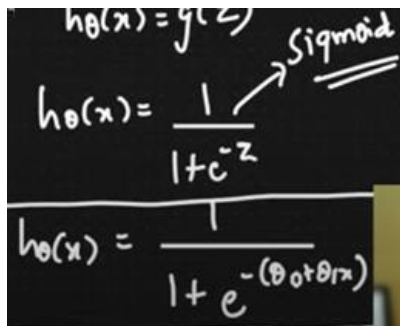
We try to apply linear regression over here, but can we? → No!

1) **Linear Regression Issue:** Linear regression doesn't constrain the output to the 0-1 range, which can result in predictions outside of the valid probability range (e.g., values greater than 1 or less than 0).

2) **Mismatch with Binary Outcomes:** Since binary outcomes (TRUE/FALSE) require probabilities between 0 and 1, linear regression is not suitable for binary classification without modification.

3) **Sigmoid Function:** To address this, logistic regression uses the Sigmoid function, which "squashes" the output of the linear model into the valid probability range of 0 to 1.

e) Sigmoid Function? What is it? How to get it? → **Decision boundary:**


$$h_{\theta}(x) = \frac{1}{1 + e^{-z}}$$
$$h_{\theta}(x) = \frac{1}{1 + e^{-(\theta_0 + \theta_1 x)}}$$

$h_{\theta}(x)$: predicted probability

z : $\theta_0 + (\theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 + \dots + \theta_n x_n)$ also known as log-odds!

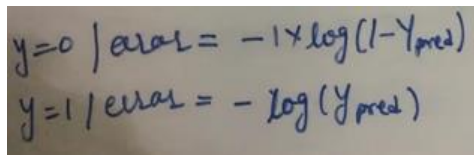
θ_0 : intercept term (bias)

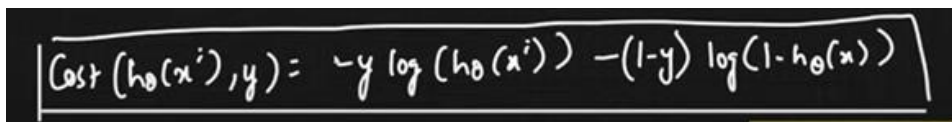
$\theta_1, \theta_2, \dots, \theta_n$: are the coefficients of the predictor variables x_1, x_2, \dots, x_n

e : base of the natural logarithm

f) Cost function:

Logistic regression uses Log Loss (or Binary Cross-Entropy) to measure the performance of the model.


$$\begin{aligned} y=0 & \text{ error} = -1 \times \log(1 - y_{\text{pred}}) \\ y=1 & \text{ error} = -\log(y_{\text{pred}}) \end{aligned}$$


$$\text{Cost}(h_{\theta}(x^i), y) = -y \log(h_{\theta}(x^i)) - (1-y) \log(1 - h_{\theta}(x^i))$$

i) cost function & loss function with no. of parameters will always be the same!

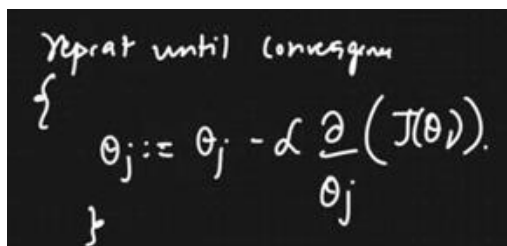
ii) as the cost function increases the penalty as the predicted probability moves further away from the true label. For example:

If $y = 1$ and $\hat{y} = 0.9$, the error is small, and the penalty is low.

If $y = 1$ and $\hat{y} = 0.1$, the error is large, and the penalty is high.

Hence, **Greater Distance \rightarrow Greater Penalty!**

g) minimize the cost function & solve the errors: **GRADIENT DESCENT'S CONVERGENCE ALGORITHM:**


$$\begin{aligned} & \text{Repeat until convergence} \\ & \left\{ \begin{aligned} & \theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} (J(\theta)) \end{aligned} \right. \\ & \left. \right\} \end{aligned}$$

h) Overfitting vs Underfitting:

i) **Overfitting:** The model fits too well to the training data, capturing noise and resulting in poor generalization to new data.

Solution: Regularization techniques (L1/L2), cross-validation, reducing model complexity.

ii) **Underfitting:** The model fails to capture the underlying trend and performs poorly even on the training data.

Solution: Adding more predictors, using non-linear transformations, increasing model complexity.

(for more, refer: Linear Regression's PDF)

i) Regularization: Regularization is especially useful when dealing with high-dimensional data to avoid overfitting.

(for more, refer: Linear Regression's PDF)

j) For the underfitting do the required process!

(for more, refer: Linear Regression Cheat sheet's PDF)

k) Cross-Validation:

K-Fold Cross-Validation: Used to assess the generalization ability of the logistic regression model by splitting the data into K subsets (folds), training on K-1 folds, and testing on the remaining fold.

Cross-validation ensures that the model is not overfitting to a particular subset of the data.

l) Performance Matrix: (Classification problems)

Prediction	Actual	
	1	0
1	TP	FP
0	FN	TN

$$\text{a) Accuracy: } \frac{TP+TN}{\text{Total}}$$

$$\text{b) Precision: } \frac{TP}{TP+FP}$$

e.g.: Spam Classification

$$\text{c) Recall: } \frac{TP}{TP+FN}$$

e.g.: Cancer Prediction

d) when both precision & recall is important, then?

F1 – Score:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

e.g.: Tomorrow stock market is going to crash!

If we think from the company point of view, it's: Precision (FP)!

If we think from the public point of view, it's: Recall (FN)!

when FP is important, will decrease β to 0.5, can say it's F0.5 score;

when FN is important, will increase β to 2, can say it's F2 score!

ON THE GIVEN PROBLEM STATEMENT, WILL GET TO KNOW WHICH TERMINOLOGY IS IMPORTANT

e) ROC Curve & AUC Curve: helps us to understand wheatear our model is the best or not.

m) Model Evaluation on Test Data:

Once the model has been trained, evaluate it using metrics such as **MSE**, **R^2** , **RMSE**, etc. This helps assess how well the model is generalizing to unseen data.

Keep checking all the below given assumption in the start, mid or in the end; whenever required!

n) Assumptions;

i) Linearity of Logit:

The relationship between the independent variables and the log-odds (logit) of the dependent variable is linear.

Check: Plot the logit (log-odds) vs. predictor variables or use Box-Tidwell test to check for non-linearity.

ii) Independence of Observations:

The observations should be independent of each other. Logistic regression assumes that the errors (or residuals) are independent.

Check: Durbin-Watson test (for autocorrelation), especially for time-series data.

iii) No Multicollinearity:

Predictor variables should not be highly correlated with each other.

Check: Correlation matrix or Variance Inflation Factor (VIF).

iv) Large Sample Size:

Logistic regression works best with large datasets, as smaller datasets can lead to overfitting or unreliable estimates.

Check: Sample size rule of thumb: The number of observations should be at least 10 times the number of predictors.

v) Binary Outcome (for Binary Logistic Regression):

The target variable should be binary (0 or 1).

Check: Ensure that the target variable only has two unique classes.