Logistic Regression

1. Logistic Regression is a fundamental machine learning algorithm used for binary classification tasks, where the target variable takes on only two possible outcomes
2. Despite its name, logistic regression is a classification algorithm rather than a regression algorithm.
3. It's named as such because it utilizes the logistic function (also known as the sigmoid function) to model the probability of a certain class.

Logistic Regression → To solve Classification problem

Classification → O/P → category

Binary → 0, 1

Logistic Regression
→ Binary Classification
→ Multi-Class "

Dataset → student → IIT

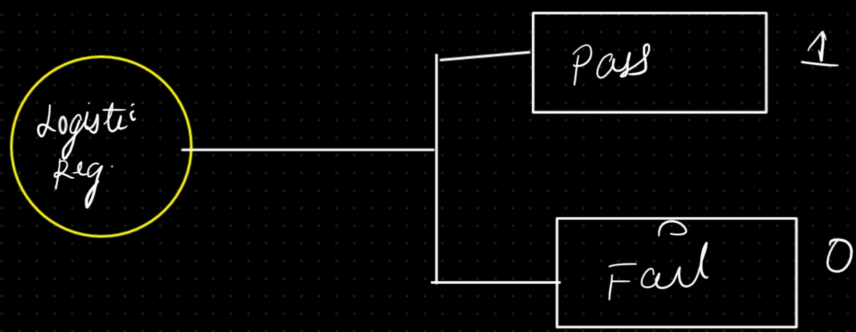
Binary → 0, 1

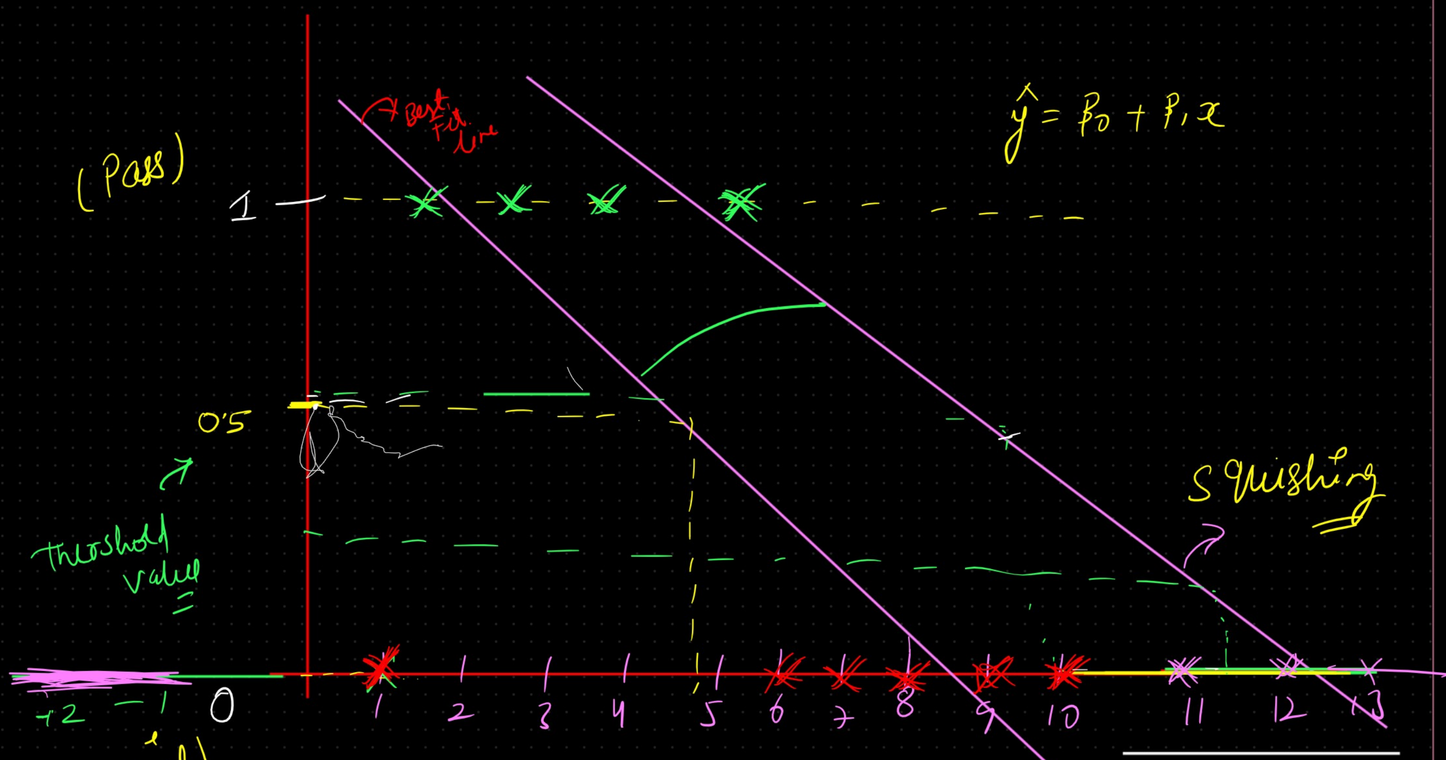
No. of play hours	Pass/Fail	
10	Fail	0
9	Fail	0
8	Fail	0
7	Fail	0
6	Fail	0
5	Pass	1
4	Pass	1
3	Pass	1

O/P
↓
categories
↓
Yes/No
Pass/Fail
will buy/won't buy

2 Pass 1

1 Fail 0





13 $\underbrace{[0, 1]}$ → prob. outliers

① Sigmoid Activation Function

(Logistic Function)

$\sigma \rightarrow [0,1]$

$$\boxed{\sigma(z) = \frac{1}{1 + e^{-z}}}$$

$$Z = \beta_0 + \beta_1 x_1$$

or

$$Z = \theta_0 + \theta_1 x_1$$

Best fit line $\hat{y} = \underbrace{\beta_0 + \beta_1 x_1}$

or

$$h_\theta(x) = \theta_0 + \theta_1 x_1$$

$$\boxed{h_\theta(x) = \sigma(\theta_0 + \theta_1 x_1)}$$

$$\Downarrow$$

$$z$$

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

$$h_\theta(x) = \frac{1}{1+e^{-(\theta_0+\theta_1 x_1)}}$$

where z is the input to the function

→ The sigmoid function maps any real-valued number to the range [0, 1].
and its output can be interpreted as a probability.

When z is positive, the sigmoid function outputs a value closer to 1, indicating a high probability of the positive class.

when z is negative, the sigmoid function outputs a value closer to 0, indicating a high probability of the negative class.

② Hypothesis Function

$$\boxed{h_\theta(x) = \sigma(\theta^T x)}$$

$\therefore$ $h_\theta(x) \rightarrow$ predicted value $\rightarrow \hat{y}$

$\theta \rightarrow$ parameter vector (weights)

$x \rightarrow$ feature vector

$\sigma \rightarrow$ sigmoid activation Function

$\theta^T x \rightarrow$ dot product of the parameter vector and the feature vector

Log Loss (Binary Cross-Entropy Loss)

Log loss, also known as binary cross-entropy loss, measures the performance of a classification model whose output is a probability value between 0 and 1. For logistic regression, the log loss function is defined as:

$$J_{\theta_0, \theta_1} = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$

$$\boxed{J(\theta_0, \theta_1) = -y \log(h_\theta(x)) - (1-y)\log(1 - h_\theta(x))}$$

if $y=1$

$$= -y \log(h_\theta(x))$$

if $y=0$

$$= -\log(1-h_\theta(x))$$

Log Loss → cost func^n

$$J(\theta) = \frac{-1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log(h_\theta(x^{i})) + (1-y^{(i)}) \log(1-h_\theta(x^{(i)})) \right]$$

The cost function is the average of the log loss over all training examples. The goal of training a logistic regression model is to minimize this cost function by adjusting the parameters (θ)

�

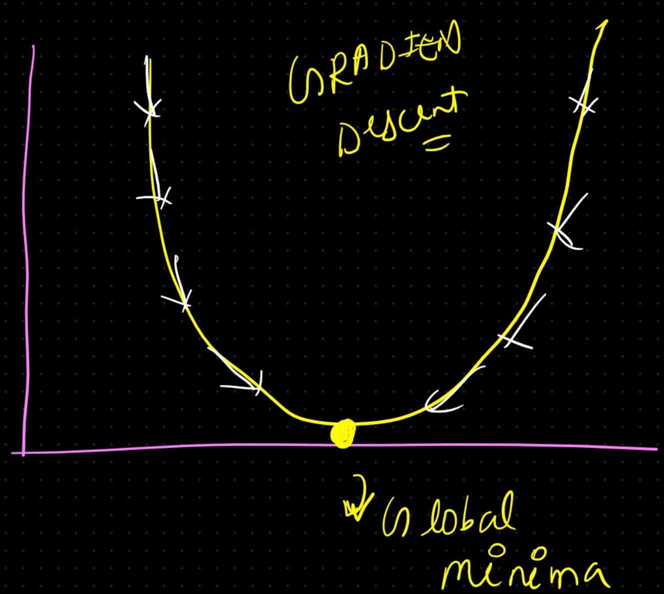
Gradient Descent

Gradient descent is an optimization algorithm used to minimize the cost function. In logistic regression, we update the parameters (θ) using the gradient of the cost function with respect to the parameters. The update rule for gradient descent is:

$$\boxed{\theta_j := \theta_j - \alpha \frac{\partial J(\theta)}{\partial \theta_j}}$$

α – Learning Rate

$\frac{\partial J(\theta)}{\partial \theta_j}$ → partial derivate of the cost function



$x \rightarrow$ student's exam score

based $\rightarrow y$

$\boxed{\text{model}}$

$\downarrow$

Pass / Fail

Suppose we have a dataset of students' exam scores (x) and whether they passed (y). We want to build a model to predict whether a student will pass based on their exam score.

- **Input Data (Features):** Exam score (x).
- **Target Variable:** Pass or fail (y).
- **Hypothesis Function:** $h_\theta(x) = \sigma(\theta_0 + \theta_1 x)$.
- **Cost Function:** $J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)}\log(h_\theta(x^{(i)})) + (1-y^{(i)})\log(1-h_\theta(x^{(i)}))\right]$.
- **Gradient Descent:** Update θ using gradient descent until convergence.

Eg $\rightarrow$ Email Spam Classification

Goal $\rightarrow$ Email $\rightarrow$ Spam = 1

$\rightarrow$ Not spam = 0

Features $\rightarrow$ Keywords $\rightarrow$ Word Freq.

$\rightarrow$ characteristics

(2) Medical Diagnosis

Patient $\rightarrow$ predict $\rightarrow$ diabetes

presence $\rightarrow$ 1   0 $\rightarrow$ absence

Feature $\rightarrow$ symptoms, test result, medical data

$\downarrow$

predict $\rightarrow$ prob $\rightarrow$ BMI, Glucose, Age

Logistic Regression with Regularization techniques

$$= \frac{1}{m}\left[-y \log(h_\theta(x)) - (1-y)\log(1-h_\theta(x))\right]$$

+ L2 Regularization (Ridge)

$$= \frac{1}{m}\left[-y \log(h_\theta(x)) - (1-y)\log(1-h_\theta(x))\right]$$

$$+ \lambda \sum_{i=1}^{n} (\theta)^2 \rightarrow \text{slope}$$

reduce overfitting

L1 Regularization (Lasso)

$$= \frac{1}{m}\left[-y \log(h_\theta(x)) - (1-y)\log(1-h_\theta(x))\right]$$

$$+ \lambda \sum_{i=1}^{n} |\theta| \rightarrow \text{slope / coeff}$$

Feature selection

Elastic Net

$$= \frac{1}{m}\left[-y \log(h_\theta(x)) - (1-y)\log(1-h_\theta(x))\right]$$

$$+ \lambda_1 \sum_{i=1}^{n} (\theta)^2 + \lambda_2 \sum_{i=1}^{n} |\theta|$$

$\lambda_1, \lambda_2 \rightarrow$ hyperparameter

$$\boxed{\lambda \propto \frac{1}{C} \quad \text{or} \quad C \propto \frac{1}{\lambda}}$$

Email Spam Classification:

In this example, the goal is to classify emails as either spam or not spam (ham). Features could include word frequencies, presence of certain keywords, or other characteristics of the email. The target variable

�

y would be binary, where 1 indicates spam and 0 indicates not spam. Logistic regression can be used to build a model that predicts the probability of an email being spam based on its features.

Medical Diagnosis:

Logistic regression can be used in medical diagnosis to predict whether a patient has a particular disease or condition based on their symptoms, test results, or other medical data. For instance, it could be used to predict the likelihood of a patient having diabetes based on factors like age, BMI, glucose levels, etc. The target variable would represent the presence or absence of the disease (1 for presence, 0 for absence).

Credit Risk Assessment:

Logistic regression is widely used in finance for credit risk assessment. Banks and financial institutions can use logistic regression models to predict the likelihood of a customer defaulting on a loan based on various factors such as credit score, income, employment status, debt-to-income ratio, etc. The target variable would indicate whether the customer defaulted (1) or not (0).

Customer Churn Prediction:

Logistic regression can be employed by businesses to predict whether a customer is likely to churn (cancel their subscription or stop using a service) based on their behavior, usage patterns, demographics, etc. Features could include frequency of purchases, length of time as a customer, customer demographics, etc. The target variable would indicate whether the customer churned (1) or not (0).

Social Media Sentiment Analysis:

Logistic regression can be used for sentiment analysis on social media platforms to classify posts or comments as positive, negative, or neutral. Features could include text features extracted from the posts, such as word frequencies, presence of emoticons, sentiment scores, etc. The target variable would represent the sentiment category (e.g., 1 for positive, 0 for negative).