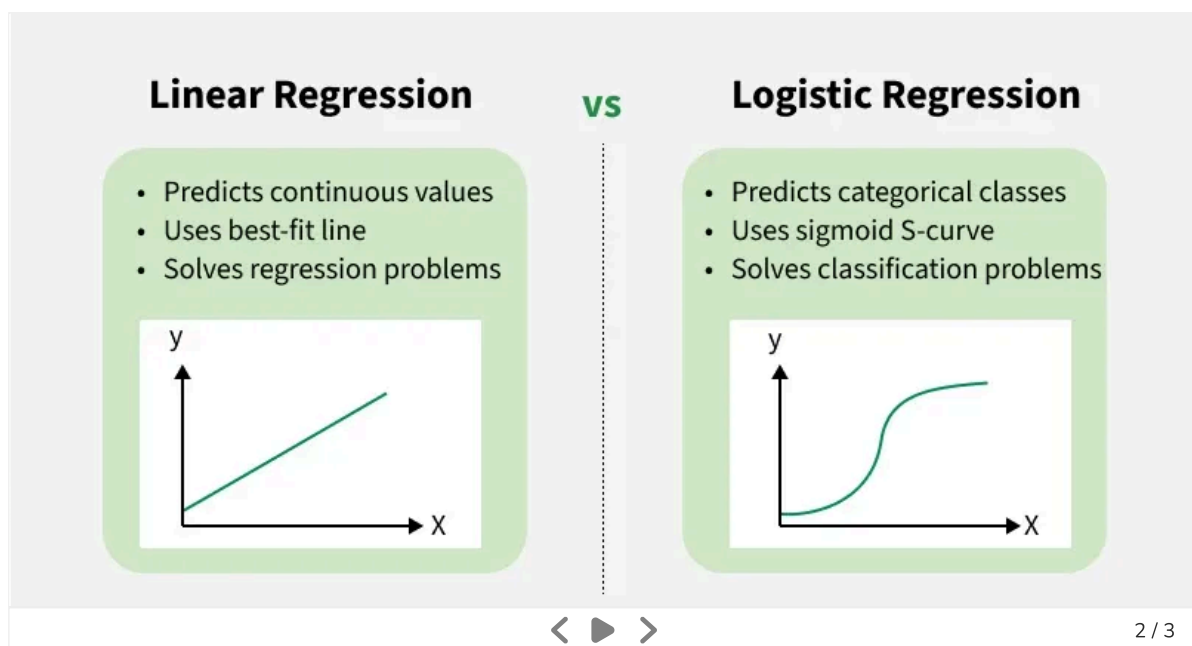Last Updated : 23 Dec, 2025

Logistic Regression is a supervised machine learning algorithm used for classification problems. Unlike linear regression which predicts continuous values it predicts the probability that an input belongs to a specific class.

- It is used for binary classification where the output can be one of two possible categories such as Yes/No, True/False or 0/1.
- It uses sigmoid function to convert inputs into a probability value between 0 and 1.

Types of Logistic Regression

Logistic regression can be classified into three main types based on the nature of the dependent variable:

1. **Binomial Logistic Regression**: This type is used when the dependent variable has only two possible categories. Examples include Yes/No, Pass/Fail or 0/1. It is the most common form of logistic regression and is used for binary classification problems.
2. **Multinomial Logistic Regression**: This is used when the dependent variable has three or more possible categories that are not ordered. For example, classifying animals into categories like "cat," "dog" or "sheep." It extends the binary logistic regression to handle multiple classes.
3. **Ordinal Logistic Regression**: This type applies when the dependent variable has three or more categories with a natural order or ranking. Examples include ratings like "low," "medium" and "high." It takes the order of the categories into account when modeling.

Assumptions of Logistic Regression

Understanding the assumptions behind logistic regression is important to ensure the model is applied correctly, main assumptions are:

1. **Independent observations**: Each data point is assumed to be independent of the others means there should be no correlation or dependence between the input samples.
2. **Binary dependent variables**: It takes the assumption that the dependent variable must be binary, means it can take only two values. For more than two categories SoftMax functions are used.
3. **Linearity relationship between independent variables and log odds**: The model assumes a linear relationship between the independent variables and the log odds of the dependent variable which means the predictors affect the log odds in a linear way.
4. **No outliers**: The dataset should not contain extreme outliers as they can distort the estimation of the logistic regression coefficients.
5. **Large sample size**: It requires a sufficiently large sample size to produce reliable and stable results.

Understanding Sigmoid Function

1. The sigmoid function is a important part of logistic regression which is used to convert the raw output of the model into a probability value between 0 and 1.

2. This function takes any real number and maps it into the range 0 to 1 forming an "S" shaped curve called the sigmoid curve or logistic curve. Because probabilities must lie between 0 and 1, the sigmoid function is perfect for this purpose.

3. In logistic regression, we use a threshold value usually 0.5 to decide the class label.

- If the sigmoid output is same or above the threshold, the input is classified as Class 1.
- If it is below the threshold, the input is classified as Class 0.

This approach helps to transform continuous input values into meaningful class predictions.

How does Logistic Regression work?

Logistic regression model transforms the linear regression function continuous value output into categorical value output using a sigmoid function which maps any real-valued set of independent variables input into a value between 0 and 1. This function is known as the logistic function.

Suppose we have input features represented as a matrix:

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ x_{21} & \cdots & x_{2m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{bmatrix}$$

and the dependent variable is Y having only binary value i.e 0 or 1.

$$Y = \begin{cases} 0 & \text{if } Class\ 1 \\ 1 & \text{if } Class\ 2 \end{cases}$$

then, apply the multi-linear function to the input variables X.
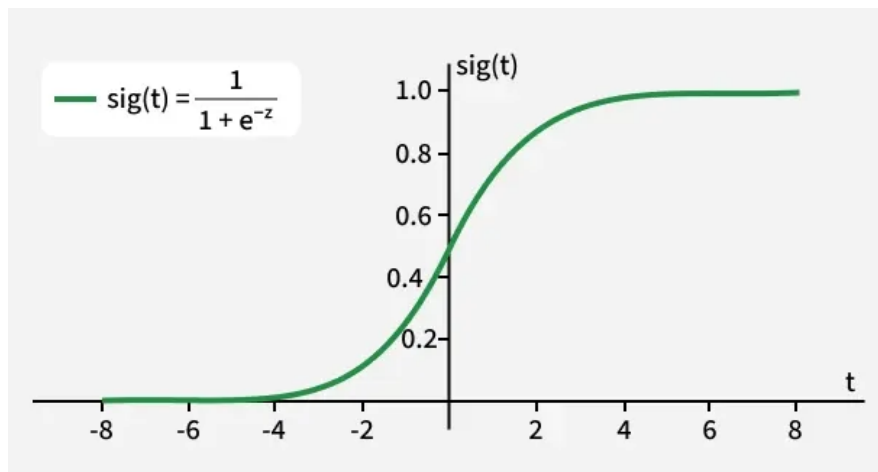
$$z = \left(\sum_{i=1}^{n} w_i x_i \right) + b$$

Here x_i is the ith observation of X, $w_i = [w_1, w_2, w_3, \cdots, w_m]$ is the weights or Coefficient and b is the bias term also known as intercept. Simply this can be represented as the dot product of weight and bias.

$$z = w \cdot X + b$$

At this stage, z is a continuous value from the linear regression. Logistic regression then applies the sigmoid function to z to convert it into a probability between 0 and 1 which can be used to predict the class.

Now we use the [sigmoid function](#) where the input will be z and we find the probability between 0 and 1. i.e. predicted y.

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

<p align="center">Sigmoid function</p>

As shown above the sigmoid function converts the continuous variable data into the probability i.e between 0 and 1.

- $\sigma(z)$ tends towards 1 as $z \to \infty$
- $\sigma(z)$ tends towards 0 as $z \to -\infty$
- $\sigma(z)$ is always bounded between 0 and 1

where the probability of being a class can be measured as:

$$P(y = 1) = \sigma(z)$$
$$P(y = 0) = 1 - \sigma(z)$$

Logistic Regression Equation and Odds:

It models the odds of the dependent event occurring which is the ratio of the probability of the event to the probability of it not occurring:

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

Taking the natural logarithm of the odds gives the log-odds or logit:

$$\log\left[\frac{p(x)}{1-p(x)}\right] = z$$
$$\log\left[\frac{p(x)}{1-p(x)}\right] = w \cdot X + b$$
$$\frac{p(x)}{1-p(x)} = e^{w \cdot X + b} \quad \cdots \text{Exponentiate both sides}$$
$$p(x) = e^{w \cdot X + b} \cdot (1 - p(x))$$
$$p(x) = e^{w \cdot X + b} - e^{w \cdot X + b} \cdot p(x))$$
$$p(x) + e^{w \cdot X + b} \cdot p(x)) = e^{w \cdot X + b}$$
$$p(x)(1 + e^{w \cdot X + b}) = e^{w \cdot X + b}$$
$$p(x) = \frac{e^{w \cdot X + b}}{1 + e^{w \cdot X + b}}$$

then the final logistic regression equation will be:

$$p(X; b, w) = \frac{e^{w \cdot X + b}}{1 + e^{w \cdot X + b}} = \frac{1}{1 + e^{-w \cdot X + b}}$$

This formula represents the probability of the input belonging to Class 1.

Likelihood Function for Logistic Regression

The goal is to find weights w and bias b that maximize the likelihood of observing the data.

For each data point i

- for $y = 1$, predicted probabilities will be: p(X;b,w) $=p(x)$
- for $y = 0$ The predicted probabilities will be: 1-p(X;b,w) $= 1 - p(x)$

$$L(b, w) = \prod_{i=1}^{n} p(x_i)^{y_i} (1 - p(x_i))^{1-y_i}$$

Taking natural logs on both sides:

$$
\begin{aligned}
\log(L(b, w)) &= \sum_{i=1}^{n} y_i \log p(x_i) \ + \ (1 - y_i) \log(1 - p(x_i)) \\
&= \sum_{i=1}^{n} y_i \log p(x_i) + \log(1 - p(x_i)) - y_i \log(1 - p(x_i)) \\
&= \sum_{i=1}^{n} \log(1 - p(x_i)) + \sum_{i=1}^{n} y_i \log \frac{p(x_i)}{1 - p(x_i)} \\
&= \sum_{i=1}^{n} - \log 1 - e^{-(w \cdot x_i + b)} + \sum_{i=1}^{n} y_i (w \cdot x_i + b) \\
&= \sum_{i=1}^{n} - \log 1 + e^{w \cdot x_i + b} + \sum_{i=1}^{n} y_i (w \cdot x_i + b)
\end{aligned}
$$

This is known as the log-likelihood function.

Gradient of the log-likelihood function

To find the best w and b we use gradient ascent on the log-likelihood function. The gradient with respect to each weight w_j is:

$$
\begin{aligned}
\frac{\partial J(l(b, w))}{\partial w_j} &= - \sum_{i=n}^{n} \frac{1}{1 + e^{w \cdot x_i + b}} e^{w \cdot x_i + b} x_{ij} + \sum_{i=1}^{n} y_i x_{ij} \\
&= - \sum_{i=n}^{n} p(x_i; b, w) x_{ij} + \sum_{i=1}^{n} y_i x_{ij} \\
&= \sum_{i=n}^{n} (y_i - p(x_i; b, w)) x_{ij}
\end{aligned}
$$

Terminologies involved in Logistic Regression

Here are some common terms involved in logistic regression:

1. **Independent Variables:** These are the input features or predictor variables used to make predictions about the dependent variable.
2. **Dependent Variable**: This is the target variable that we aim to predict. In logistic regression, the dependent variable is categorical.
3. **Logistic Function**: This function transforms the independent variables into a probability between 0 and 1 which represents the likelihood that the dependent variable is either 0 or 1.
4. **Odds**: This is the ratio of the probability of an event happening to the probability of it not happening. It differs from probability because probability is the ratio of occurrences to total possibilities.

5. **Log-Odds (Logit)**: The natural logarithm of the odds. In logistic regression, the log-odds are modeled as a linear combination of the independent variables and the intercept.
6. **Coefficient**: These are the parameters estimated by the logistic regression model which shows how strongly the independent variables affect the dependent variable.
7. **Intercept**: The constant term in the logistic regression model which represents the log-odds when all independent variables are equal to zero.
8. **Maximum Likelihood Estimation (MLE)**: This method is used to estimate the coefficients of the logistic regression model by maximizing the likelihood of observing the given data.

Implementation for Logistic Regression

Now, let's see the implementation of logistic regression in Python. Here we will be implementing two main types of Logistic Regression:

1. Binomial Logistic regression:

In binomial logistic regression, the target variable can only have two possible values such as "0" or "1", "pass" or "fail". The sigmoid function is used for prediction.

We will be using sckit-learn library for this and shows how to use the breast cancer dataset to implement a Logistic Regression model for classification.

```python
from sklearn.datasets import load_breast_cancer
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

X, y = load_breast_cancer(return_X_y=True)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=23)

clf = LogisticRegression(max_iter=10000, random_state=0)
clf.fit(X_train, y_train)

acc = accuracy_score(y_test, clf.predict(X_test)) * 100
print(f"Logistic Regression model accuracy: {acc:.2f}%")
```

Output:

> *Logistic Regression model accuracy (in %): 96.49%*

This code uses logistic regression to classify whether a sample from the breast cancer dataset is malignant or benign.

2. Multinomial Logistic Regression:

Target variable can have 3 or more possible types which are not ordered i.e types have no quantitative significance like "disease A" vs "disease B" vs "disease C".

In this case, the softmax function is used in place of the sigmoid function. Softmax function for K classes will be:

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Here K represents the number of elements in the vector z and i, j iterates over all the elements in the vector.

Then the probability for class c will be:

$$P(Y = c | \overrightarrow{X} = x) = \frac{e^{w_c \cdot x + b_c}}{\sum_{k=1}^{K} e^{w_k \cdot x + b_k}}$$

Below is an example of implementing multinomial logistic regression using the Digits dataset from scikit-learn:

```python
from sklearn.model_selection import train_test_split
from sklearn import datasets, linear_model, metrics

digits = datasets.load_digits()

X = digits.data
y = digits.target

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=1)

reg = linear_model.LogisticRegression(max_iter=10000, random_state=0)
reg.fit(X_train, y_train)

y_pred = reg.predict(X_test)

print(f"Logistic Regression model accuracy: {metrics.accuracy_score(y_test, y_pred) * 100:.2f}%")
```

Output:

Logistic Regression model accuracy: 96.66%

This model is used to predict one of 10 digits (0-9) based on the image features.

How to Evaluate Logistic Regression Model?

Evaluating the logistic regression model helps assess its performance and ensure it generalizes well to new, unseen data. The following metrics are commonly used:

1. Accuracy: Accuracy provides the proportion of correctly classified instances.

$$Accuracy = \frac{True\,Positives + True\,Negatives}{Total}$$

2. Precision: Precision focuses on the accuracy of positive predictions.

$$Precision = \frac{True\,Positives}{True\,Positives + False\,Positives}$$

3. Recall (Sensitivity or True Positive Rate): Recall measures the proportion of correctly predicted positive instances among all actual positive instances.

$$Recall = \frac{True\,Positives}{True\,Positives + False\,Negatives}$$

4. F1 Score: F1 score is the harmonic mean of precision and recall.

$$F1\,Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

5. Area Under the Receiver Operating Characteristic Curve (AUC-ROC): The ROC curve plots the true positive rate against the false positive rate at various thresholds. AUC-ROC measures the area under this curve which provides an aggregate measure of a model's performance across different classification thresholds.

6. Area Under the Precision-Recall Curve (AUC-PR): Similar to AUC-ROC, AUC-PR measures the area under the precision-recall curve helps in providing a summary of a model's performance across different precision-recall trade-offs.

Differences Between Linear and Logistic Regression

Logistic regression and linear regression differ in their application and output. Here's a comparison:

Aspect	Linear Regression	Logistic Regression
Definition	Linear regression is used to predict the continuous dependent variable using a given set of independent variables.	Logistic regression is used to predict the categorical dependent variable using a given set of independent variables.
Problem Type	It is used for solving regression problem.	It is used for solving classification problems.
Output Type	In this we predict the value of continuous variables.	In this we predict values of categorical variables.
Curve/Model Fitting	In this we find best fit line.	In this we find S-Curve.
Estimation Method	Least square estimation method is used for estimation of accuracy.	Maximum likelihood estimation method is used for estimation of accuracy.
Output Example	The output must be continuous value such as price, age etc.	Output must be categorical value such as 0 or 1, Yes or No, etc.
Relationship Requirement	It required linear relationship between dependent and independent variables.	It not required linear relationship.
Collinearity	There may be collinearity between the independent variables.	There should be little to no collinearity between independent variables.

Comment	K **kartik** + Follow	125

Article Tags: Machine Learning   AI-ML-DS   python   AI-ML-DS With Python

Explore

Machine Learning Basics

Python for Machine Learning

Feature Engineering

Supervised Learning

Unsupervised Learning

Model Evaluation and Tuning

GeeksforGeeks
Sanchhaya Education Private Limited

Corporate & Communications Address:
A-143, 7th Floor, Sovereign Corporate Tower, Sector- 136, Noida, Uttar Pradesh (201305)

Registered Address:
K 061, Tower K, Gulshan Vivante Apartment, Sector 137, Noida, Gautam Buddh Nagar, Uttar Pradesh, 201305

Company
About Us
Legal
Privacy Policy
Contact Us
Advertise with us
GFG Corporate Solution
Training Program

Explore
POTD
Job-A-Thon
Blogs
Nation
Skill Up

Tutorials
Programming Languages
DSA
Web Technology
AI, ML & Data Science
DevOps
CS Core Subjects
Interview Preparation
Software and Tools

Courses
ML and Data Science
DSA and Placements
Web Development
Programming Languages
DevOps & Cloud
GATE
Trending Technologies

Videos
DSA
Python
Java
C++
Web Development
Data Science
CS Subjects

Preparation Corner
Interview Corner
Aptitude
Puzzles
GfG 160
System Design