How Does Logistic Regression Work?

Image by Editor

Logistic Regression Algorithm.

This is a machine learning technique used for classification or prediction in data sets which have many features, but where most of them have little value and should be ignored.

For example, imagine that you have several algorithms for analyzing your photos. Some are good at converting images into text, others are good at detecting objects in photos but not so much for recognizing people’s faces, and others still focus on finding patterns in colour schemes. Now say that you have an image with two people in it: one smiling and one frowning—which one is the happier of the two? Assuming this isn’t something to be kept as a secret (which it probably shouldn’t be), then you need more than just judging by appearances. You want to learn what makes people happy and sad, so instead of having various algorithms that each specialize in identifying different emotions—you want to create a single algorithm that knows how to recognize both emotions equally well.

Classification

For example, an email may be labelled as spam or ham (not spam) for each instance. This is used with binary classification problem.

What is Logistic Regression?

It uses Sigmoid function

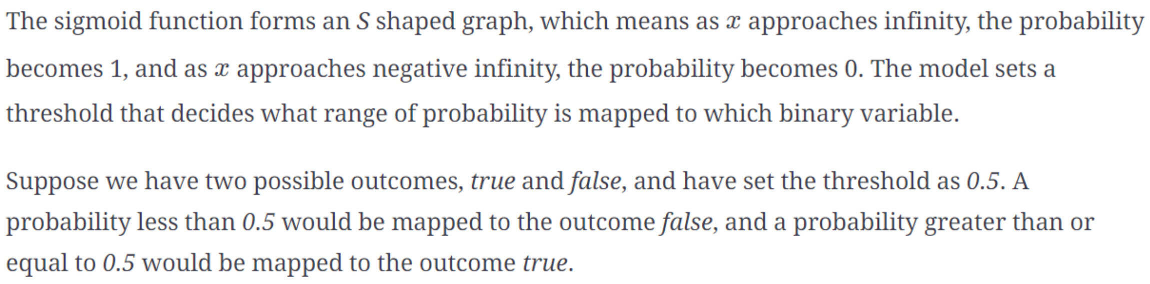
What is a sigmoid function?

The logistic function in linear regression is a type of sigmoid, a class of functions with the same specific properties.

Sigmoid is a mathematical function that takes any real number and maps it to a probability between 1 and 0.

The formula of the sigmoid function is:

$$f(x) = \frac{1}{1+e^{-x}}$$



The sigmoid function forms an S shaped graph, which means as x approaches infinity, the probability becomes 1, and as x approaches negative infinity, the probability becomes 0. The model sets a threshold that decides what range of probability is mapped to which binary variable.

Suppose we have two possible outcomes, *true* and *false*, and have set the threshold as *0.5*. A probability less than *0.5* would be mapped to the outcome *false*, and a probability greater than or equal to *0.5* would be mapped to the outcome *true*.

Logistic regression is a Machine Learning classification algorithm that is used to predict the probability of certain classes based on some dependent variables. In short, the logistic regression model computes a sum of the input features (in most cases, there is a bias term), and calculates the logistic of the result.

The output of logistic regression is always between (0, and 1), which is suitable for a binary classification task. The higher the value, the higher the probability that the current sample is classified as class=1, and vice versa.

$$\mathrm{h}_{\Theta}(\mathrm{X}) = \frac{1}{1+\mathrm{e}^{-\Theta \mathrm{X}}}$$

As the formula above shows, Θ is the parameter we want to learn or train or optimize and $\mathbf{X}$ is the input data. The output is the prediction value when the value is closer to 1, which means the instance is more likely to be a positive sample(y=1). If the value is closer to 0, this means the instance is more likely to be a negative sample(y=0).

To optimize our task, we need to define a loss function(cost or objective function) for this task. In logistic regression, we use the log-likelihood loss function.

$$\mathrm{J}(\Theta) = -\frac{1}{\mathrm{m}}\sum_{\mathrm{m}}^{\mathrm{i}=1}(\mathrm{y}^{\mathrm{i}}\log(\mathrm{p}^{\mathrm{i}}) + (1 - \mathrm{y}^{\mathrm{i}})\log(1 - \mathrm{p}^{\mathrm{i}}))$$

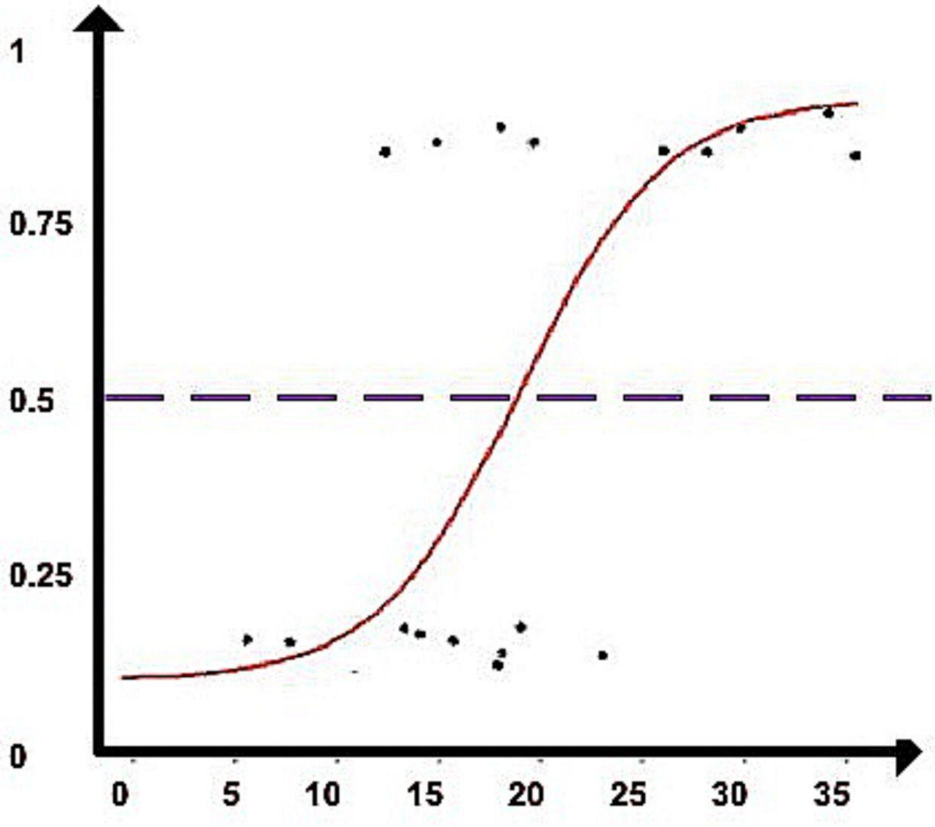
m is the number of samples in the training data. y^{i} is the label of the i-th sample, p^{i} i is the prediction value of the i-th sample. When the current sample's label is 1, then the second term of the formula is 0. We hope the larger the first term, the better, and vice versa. Finally, we add the loss of all samples, take the average, and add a negative sign. We want to minimize the quadratic cost function $\mathrm{J}(\Theta)$. When $\mathrm{J}(\Theta)$ is smaller, it means that the model fits better on the data set.

How Does Logistic Regression Work?

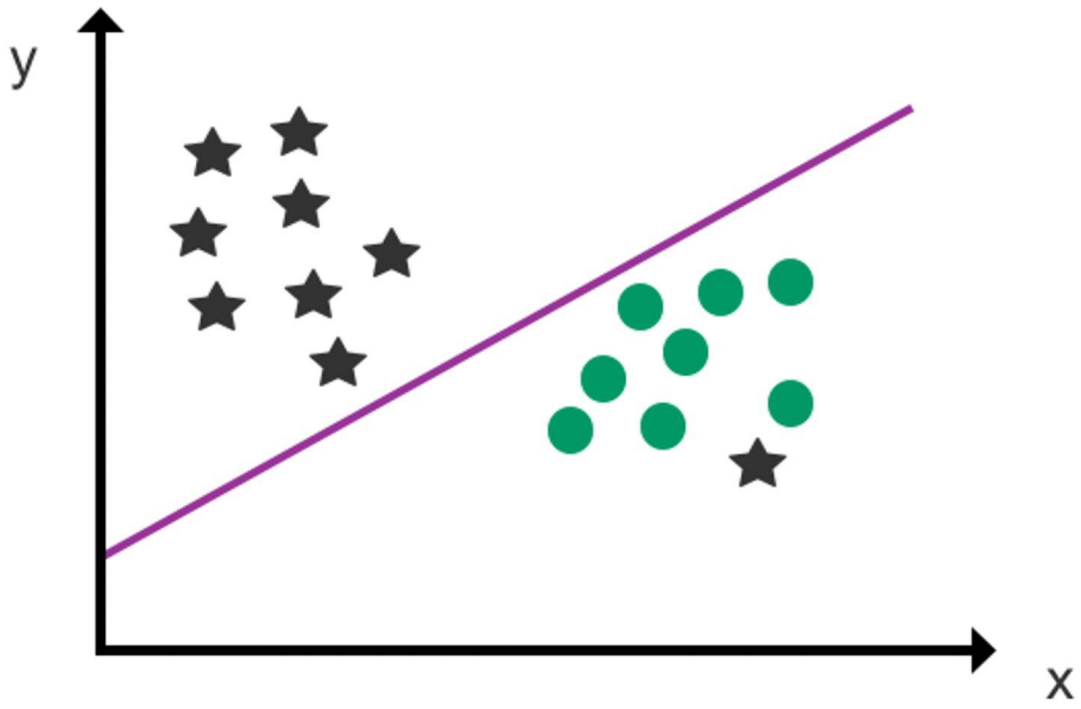
Machine learning generally involves predicting a quantitative outcome or a qualitative class. The former is commonly referred to as a regression problem. In the scenario of linear regression, the input is a continuous variable, and the prediction is a numerical value. When predicting a qualitative outcome (class), the task is considered a classification problem. Examples of classification problems include predicting what products a user will buy or if a target user will click on an online advertisement.

Not all algorithms fit cleanly into this simple dichotomy, though, and logistic regression is a notable example. Logistic regression is part of the regression family as it involves predicting outcomes based on quantitative relationships between variables. However, unlike linear regression, it accepts both continuous and discrete variables as input and its output is qualitative. In addition, it predicts a discrete class such as “Yes/No” or “Customer/Non-customer”.

In practice, the **logistic regression algorithm** analyzes relationships between variables. It assigns probabilities to discrete outcomes using the Sigmoid function, which converts numerical results into an expression of probability between 0 and 1.0. Probability is either 0 or 1, depending on whether the event happens or not. For binary predictions, you can divide the population into two groups with a cut-off of 0.5. Everything above 0.5 is considered to belong to group A, and everything below is considered to belong to group B.



A hyperplane is used as a decision line to separate two categories (as far as possible) after data points have been assigned to a class using the Sigmoid function. The class of future data points can then be predicted using the decision boundary.



Assumptions for Logistic Regression:

- The dependent variable must be categorical in nature.
- The independent variable should not have multi-collinearity.

Logistic Regression Equation:

The Logistic regression equation can be obtained from the Linear Regression equation. The mathematical steps to get Logistic Regression equations are given below:

- We know the equation of the straight line can be written as:

$$y= b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n$$

- In Logistic Regression y can be between 0 and 1 only, so for this let's divide the above equation by (1-y):

$$\frac{y}{1-y}\text{; 0 for y= 0, and infinity for y=1}$$

- But we need range between -[infinity] to +[infinity], then take logarithm of the equation it will become:

$$\boldsymbol{log\left[\frac{y}{1-y}\right] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n}$$

The above equation is the final equation for Logistic Regression.

Type of Logistic Regression:

On the basis of the categories, Logistic Regression can be classified into three types:

- **Binomial:** In binomial Logistic regression, there can be only two possible types of the dependent variables, such as 0 or 1, Pass or Fail, etc.
- **Multinomial:** In multinomial Logistic regression, there can be 3 or more possible unordered types of the dependent variable, such as "cat", "dogs", or "sheep"
- **Ordinal:** In ordinal Logistic regression, there can be 3 or more possible ordered types of dependent variables, such as "low", "Medium", or "High".