Digital Skills Training for Student
Central Computer Center, KUET

FUTURE IS HERE

ENHANCING DIGITAL GOVERNMENT AND ECONOMY

Machine Learning with Python

Session 21: Logistic Regression

Shah Muhammad Azmat Ullah
Assistant Professor, Dept. of ECE
Khulna University of Engineering & Technology

Contents

Introduction

Logistic regression is a **supervised machine learning algorithm** used for **classification tasks** where the goal is to predict the probability that an instance belongs to a given class or not.

Logistic regression is used for binary **classification** where we use sigmoid function, that takes input as independent variables and produces a probability value between 0 and 1.

For example, we have two classes Class 0 and Class 1 if the value of the logistic function for an input is greater than 0.5 (threshold value) then it belongs to Class 1 otherwise it belongs to Class 0. It's referred to as regression because it is the extension of linear regression but is mainly used for classification problems.

Introduction

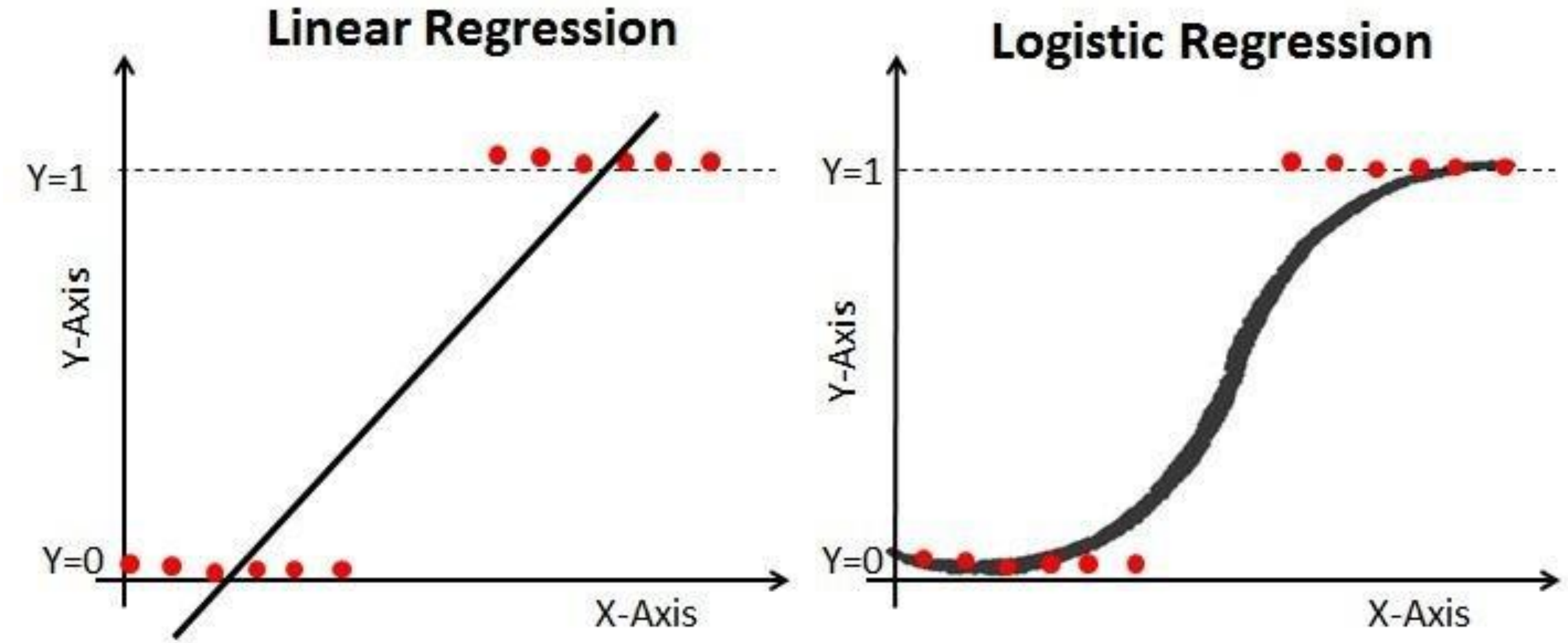
- Logistic regression predicts the output of a categorical dependent variable. Therefore, the outcome must be a categorical or discrete value.

- It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

- In Logistic regression, instead of fitting a regression line, we fit an “S” shaped logistic function, which predicts two maximum values (0 or 1).

Differences Between Linear and Logistic Regression

linear regression output is the continuous value that can be anything while logistic regression predicts the probability that an instance belongs to a given class or not.



Differences Between Linear and Logistic Regression

Linear Regression	Logistic Regression
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.
Linear regression is used for solving **regression problem**.	It is used for solving **classification problems.**
In this we predict the value of continuous variables	In this we predict values of categorical variables
In this we find best fit line.	In this we find S-Curve.
Least square estimation method is used for estimation of accuracy.	Maximum likelihood estimation method is used for Estimation of accuracy.

Differences Between Linear and Logistic Regression

Linear Regression	Logistic Regression
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.
It required linear relationship between dependent and independent variables.	It not required linear relationship.
There may be collinearity between the independent variables.	There should be little to no collinearity between independent variables.

Types of Logistic Regression

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

1. **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.

2. **Multinomial:** In multinomial Logistic regression, there can be 3 or more possible unordered types of the dependent variable, such as “cat”, “dogs”, or “sheep”

3. **Ordinal:** In ordinal Logistic regression, there can be 3 or more possible ordered types of dependent variables, such as “low”, “Medium”, or “High”.

Terminologies involved in Logistic Regression

Independent variables: The input characteristics or predictor factors applied to the dependent variable's predictions.

Dependent variable: The target variable in a logistic regression model, which we are trying to predict.

Logistic function: The formula used to represent how the independent and dependent variables relate to one another. The logistic function transforms the input variables into a probability value between 0 and 1, which represents the likelihood of the dependent variable being 1 or 0.

Odds: It is the ratio of something occurring to something not occurring. it is different from probability as the probability is the ratio of something occurring to everything that could possibly occur.

Terminologies involved in Logistic Regression

Log-odds: The log-odds, also known as the logit function, is the natural logarithm of the odds. In logistic regression, the log odds of the dependent variable are modeled as a linear combination of the independent variables and the intercept.

Coefficient: The logistic regression model's estimated parameters, show how the independent and dependent variables relate to one another.

Intercept: A constant term in the logistic regression model, which represents the log odds when all independent variables are equal to zero.

Maximum likelihood estimation: The method used to estimate the coefficients of the logistic regression model, which maximizes the likelihood of observing the data given the model.

Assumptions of Logistic Regression

- These assumptions is important to ensure that we are using appropriate application of the model. The assumption include:

1. **Independent observations**: Each observation is independent of the other. meaning there is no correlation between any input variables.
2. **Binary dependent variables**: It takes the assumption that the dependent variable must be binary or dichotomous, meaning 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 relationship between the independent variables and the log odds of the dependent variable should be linear.
4. **No outliers**: There should be no outliers in the dataset.
5. **Large sample size:** The sample size is sufficiently large

How does Logistic Regression work

The 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.

Let the independent input features be:

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

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

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

How does Logistic Regression work

then, apply the multi-linear function to the input variables X.

$$z = \left(\sum_{i=1}^{n} w_i x_i\right) + b$$

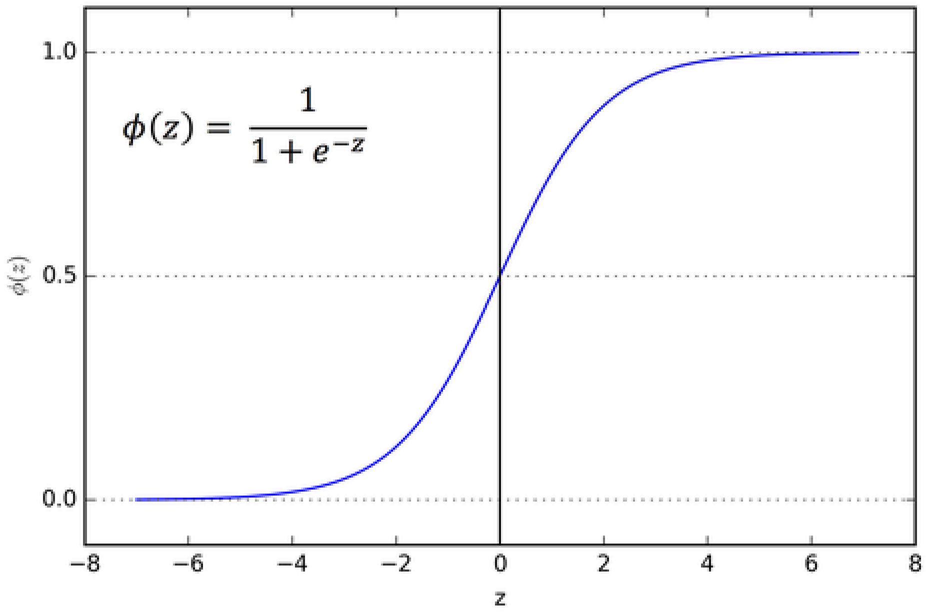
Here xi is the ith observation of X, wi=[w1,w2,w3,⋯,wm] is the weights or Coefficient, and b is the bias term also known as intercept. simply this can be represented as the dot product of weight and bias.

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

whatever we discussed above is the linear regression.

Sigmoid Function

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.



As shown above, the figure 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

Maximum Likelihood

- In logistic regression, ML is used to estimate the parameters *ww* by maximizing the likelihood of the observed data.

Log-Likelihood

To simplify calculations, the log-likelihood $\ell(\mathbf{w})$ is used:

$$\ell(\mathbf{w}) = \sum_{i=1}^{n} \left[y_i \log \sigma(\mathbf{w}^T \mathbf{x}_i) + (1 - y_i) \log(1 - \sigma(\mathbf{w}^T \mathbf{x}_i)) \right]$$

Objective

Maximizing the log-likelihood is equivalent to finding the optimal parameters:

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \ell(\mathbf{w})$$

Log Loss

- Log Loss, also known as Logistic Loss or Cross-Entropy Loss, is a common cost function for classification problems, particularly binary or multi-class classification.

- It measures the difference between predicted probabilities and actual class labels.

- Minimizing Log Loss is equivalent to maximizing the likelihood.

For Binary Classification:

- Predicted probability: $\hat{y} \in [0, 1]$
- Actual label: $y \in \{0, 1\}$
- Log Loss:

$$\text{Log Loss} = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)\right]$$

For Multi-Class Classification:

- Predicted probability vector: $\hat{\mathbf{y}} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_k]$
- One-hot encoded label vector: $\mathbf{y} = [y_1, y_2, \ldots, y_k]$
- Log Loss:

$$\text{Log Loss} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{k} y_{ij}\log(\hat{y}_{ij})$$

Gradient Descent

- To minimize Log Loss, we use **Gradient Descent** to iteratively update the parameters w.

Gradient of Log Loss

The gradient of $J(\mathbf{w})$ with respect to $\mathbf{w}$ is:

$$\nabla J(\mathbf{w}) = -\frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \left(y_i - \sigma(\mathbf{w}^T\mathbf{x}_i)\right)$$

Here:

- $y_i - \sigma(\mathbf{w}^T\mathbf{x}_i)$ is the error between the true label and predicted probability.

Update Rule

The parameters are updated iteratively as:

$$\mathbf{w} \leftarrow \mathbf{w} - \eta\nabla J(\mathbf{w})$$

Where:

- η: Learning rate (step size).
- $\nabla J(\mathbf{w})$: Gradient of Log Loss.

Connection Between Concepts

1. Maximum Likelihood: Logistic regression parameters are estimated by maximizing the likelihood of the observed data.

2. Log Loss: Equivalent to minimizing the negative log-likelihood, serving as the objective function.

3. Gradient Descent: An optimization method used to minimize the Log Loss function iteratively.

Log-Odds

The **log-odds** (or logit) in logistic regression refers to the logarithm of the odds that an event occurs, given the input features. It is a key concept underlying logistic regression as it forms the link between the linear combination of input features and the predicted probability.

1. Definition of Odds

The **odds** represent the ratio of the probability of an event occurring to the probability of it not occurring:

$$\text{Odds} = \frac{P(y=1|\mathbf{x})}{1 - P(y=1|\mathbf{x})}$$

Log-Odds

2. Log-Odds (Logit Function)

The **log-odds** is the natural logarithm of the odds:

$$\text{Log-Odds} = \log\left(\frac{P(y=1|\mathbf{x})}{1-P(y=1|\mathbf{x})}\right)$$

In logistic regression, the log-odds is modeled as a linear function of the input features:

$$\text{Log-Odds} = \mathbf{w}^T\mathbf{x} = w_0 + w_1x_1 + w_2x_2 + \cdots + w_kx_k$$

Where:

- $\mathbf{w} = [w_0, w_1, w_2, \ldots, w_k]$ are the model parameters.
- $\mathbf{x} = [1, x_1, x_2, \ldots, x_k]$ is the feature vector (including a bias term with value 1).

Log-Odds

3. Sigmoid Function and Probabilities

The sigmoid function maps the log-odds to a probability:

$$P(y=1|\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1+e^{-\mathbf{w}^T\mathbf{x}}}$$

Reversing the relationship:

$$\frac{P(y=1|\mathbf{x})}{1-P(y=1|\mathbf{x})} = e^{\mathbf{w}^T\mathbf{x}}$$

Taking the natural logarithm gives:

$$\log\left(\frac{P(y=1|\mathbf{x})}{1-P(y=1|\mathbf{x})}\right) = \mathbf{w}^T\mathbf{x}$$

Log-Odds

Example

Suppose:

$$\text{Log-Odds} = -1 + 2x_1 - 0.5x_2$$

- If $x_1 = 1$ and $x_2 = 0$:

$$\text{Log-Odds} = -1 + 2(1) - 0.5(0) = 1$$

Odds:

$$\text{Odds} = e^1 \approx 2.718$$

Probability:

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + e^{-1}} \approx 0.731$$

This process links the input features to the probability of the binary outcome via the log-odds.

Let me know if you'd like a step-by-step example or further explanation!

Precision-Recall Tradeoff in Logistic Regression Threshold Setting

Logistic regression becomes a classification technique only when a decision threshold is brought into the picture. The setting of the threshold value is a very important aspect of Logistic regression and is dependent on the classification problem itself.

The decision for the value of the threshold value is majorly affected by the values of precision and recall. Ideally, we want both precision and recall being 1, but this seldom is the case.

Precision-Recall Tradeoff in Logistic Regression Threshold Setting

In the case of a **Precision-Recall tradeoff**, we use the following arguments to decide upon the threshold:

1. **Low Precision/High Recall:** In applications where we want to reduce the number of false negatives without necessarily reducing the number of false positives, we choose a decision value that has a low value of Precision or a high value of Recall. For example, in a cancer diagnosis application, we do not want any affected patient to be classified as not affected without giving much heed to if the patient is being wrongfully diagnosed with cancer. This is because the absence of cancer can be detected by further medical diseases, but the presence of the disease cannot be detected in an already rejected candidate.

Precision-Recall Tradeoff in Logistic Regression Threshold Setting

In the case of a **Precision-Recall tradeoff**, we use the following arguments to decide upon the threshold:

2. High Precision/Low Recall: In applications where we want to reduce the number of false positives without necessarily reducing the number of false negatives, we choose a decision value that has a high value of Precision or a low value of Recall. For example, if we are classifying customers whether they will react positively or negatively to a personalized advertisement, we want to be absolutely sure that the customer will react positively to the advertisement because otherwise, a negative reaction can cause a loss of potential sales from the customer.

Precision-Recall Tradeoff in Logistic Regression Threshold Setting

In the case of a **Precision-Recall tradeoff**, we use the following arguments to decide upon the threshold:

2. High Precision/Low Recall: In applications where we want to reduce the number of false positives without necessarily reducing the number of false negatives, we choose a decision value that has a high value of Precision or a low value of Recall. For example, if we are classifying customers whether they will react positively or negatively to a personalized advertisement, we want to be absolutely sure that the customer will react positively to the advertisement because otherwise, a negative reaction can cause a loss of potential sales from the customer.

Practice

Logistic Regression Implementation:

Use **data_LogR.csv** and **logistic-regression-implementation.ipynb**

https://www.kaggle.com/code/kanncaa1/logistic-regression-implementation/notebook

Logistic Regression Classifier Tutorial

Use weatherAUS.csv and logistic-regression-classifier-tutorial.ipynb

https://www.kaggle.com/code/prashant111/logistic-regression-classifier-tutorial/notebook#1.-Introduction-to-Logistic-Regression-

LogisticRegression

https://scikit-learn.org/1.5/modules/generated/sklearn.linear_model.LogisticRegression.html

Thank You!