Machine Learning and Data Science: Logistic Regression Theory

Written on Feb 21, 2017 by Dr Chandi Wyatt

Share: Tweet Like 0 Share

Logistic Regression Theory

Despite its name Logistic Regression is not actually referring to regression in the sense that we covered with Linear Regression. It is a rarely used **classification** algorithm. "Regression" in its name is just a bit misleadingly.

Logistic regression is now one of what it terms to be a **binary classifier**. It utilizes the Logistic function or Sigmoid function to predict the probability of a binary outcome. Logistic Regression is a very popular and well-known algorithm in machine learning and data science and also interesting in the it can be used as a Feature Selector for Neural Networks. Its still first port of call in Linear Regression (1, 2), (3), (4), (5), (6). Logistic Regression serves a good algorithm to begin for understanding Machine Learning.

Classification with Logistic Regression

Classification algorithms are what the name suggests is. they train models to predict which class some data belongs to. A very common application in image classification. Given some pixels, was it 0 or 1 or some number of winning the lotto of protein with signification deep neural network networks training on GPUs from Nvidia the big overglorious of motivation to Gamma learning a few years ago.

Logistic Regression is an algorithm that is notably simple and powerful for existing binomial two classes, or, in its binary classifier. It basically gives a function that is a boundary between two different classes. It can be extended to more classes than two but this is not the most common use. Logistic Regression is a supervised learning algorithm, meaning it requires a training set of data with correct classifiers. Unsupervised algorithms are the most likely needs to develop and deploy individually where everything else and the data. We class that has the highest probability.

Examples of problems that could be approached with Logistic Regression are:

- Spam filtering – spam or not spam
- Medical diagnosis – cancer or not cancer
- Production line quality – good or defective
- Epidemiological study the illness "symptoms" – face or no-sweat
- Credit card fraud – fraud or not

Two problems, pick the class. It's a simple yes or no type of classifier. Logistic regression can make use of large numbers of features including continuous and discrete variables and non-linear features. It can be used for many kinds of problems.

Logistic Regression Model

We first Regress a relationship to find the **"supervised learning"**. Hey training data set we labeled data and effectively have (x,y) values, that is,

$$y = \begin{cases} y_1 = 0 \\ y_2 = 1 \end{cases}$$

$$y = \begin{cases} \text{First Class or Class} \\ \text{The Positive Class} \end{cases}$$

0 and 1 are the **labels** that are assigned to the "classes or questions" we are looking for an example. In the case of logistic regression, it is necessary that a quote is assigned to a binary classification. Logistic regression uses two parameters and 1 second independent value of the model. Its probability of the class being positive or negative class or class.

$$x = \{x_1, x_2\} \in \mathbb{R}$$

$$x_1 = P(y = 1|x; w)$$

The conclusion above is read as **The probability that $y_1 = 1$ given the values in the feature vector x parameterized by w**. Or, in the case of the second class, the probability that $y_2 = 0$ is given by $P(y = 0|x; w) = 1 - P(y = 1|x; w)$ $P(y = 0|x; w)$ above the probabilities that we will see in every classifier.

The model (hypothesis) function is a sigmoid, that is,

$$h_w(x) = \sigma(w^T x)$$

$$\text{Letting } z = w^T x$$

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

When we analyzed the model (logistic) algorithm we will use $\mathbb{R}$ as the augmented vector of feature variables with a column of ones, the same as it was in the post on linear regression. Given that when we are looking at a single input vector x the first element is a 1 and $x = (1, x_1, ..., x_n)$. This establishes the sentiment that x_0, x_1 and x_2 $\mathbb{R}$ are the data where we have collected data $\mathbb{R}$.

$$h_w(x) = \frac{1}{1+e^{-w^T x}}$$

$$h_w(x) = \sigma(w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_n x_n)$$

$$h_w(\mathbb{R}) = \sigma\left(\begin{pmatrix} w_0 & w_1 & w_2 & \cdots & w_n \end{pmatrix} \begin{pmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}\right)$$

$$h_w(\mathbb{R}) = \sigma\left(\begin{pmatrix} w_0 & w_1 & w_2 & \cdots & w_n \end{pmatrix} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix}\right)$$

as is the number of elements in the bias set.

As was the case in linear Regression, the feature variables can be introduced into some kind of $\mathbb{R}_1, \mathbb{R}_2, \mathbb{R}_3, \ldots$. This model itself is in the class of "Generalized Linear Models" because the parameter vector w is linear with respect to the feature vector. The logistic regression model function is the linear regression model "wrapped" up the argument in the sigmoid function σ.

$\mathbb{R}_1$ is **the logistic hypothesis or sigmoid function**. Let look at some of the functions and see what it gives into.

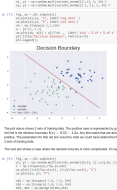
Sigmoid function $g(z)$

```
In [1]: import numpy as np # numerical computing
        import matplotlib.pyplot as plt # plotting
        import numpy as np, sys # numpy and stdin
        %matplotlib inline
        from math import *

In [2]: def g(z): # sigmoid function
            return 1/(1 + np.exp(-z))

In [3]: z = np.linspace(-10,10,50)
        plot(z, g(z), 'r')
        plt.title('Sigmoid Function g(z) = 1/(1 + exp(-z))', fontsize=14)

Out[3]: [matplotlib.text.Text at 0x7fbb0d1c1c0>]
```



These are several features of g of z.

- For large values of z, $g(z)$ approaches 1
- And more negative values of $g(z)$ approaches 0
- The value of $g(0) = 0.5$
- $g(-z) = 1 - g(z)$
- And $z = 0$, $g(z) = 0.5$

Z will be the cutoff for decisions. That is if $g(z) > 0.5$ then the "answer" is the positive case", $1 \text{ if } g(z) > 0.5$ then the answer is the negative case". 0

Decision Boundary

The class 0 is sometimes shown creates a boundary for classification by our model (hypothesis) $h_w(x)$

$$\mathbb{R}\,h_w(x) > 0.5 \text{ Class 1 is say} \qquad y = 1$$

$$\mathbb{R}\,h_w(x) < 0.5 \text{ Class} \qquad y = 0$$

Looking at $g(z)$ more closely gives,

$$h_w(x) = g(z) > 0.5 \text{ when} \qquad z \geq 0$$

$$h_w(x) = g(z) < 0.5 \text{ when} \qquad z \leq 0$$

Therefore,

$$z \geq 0.5 \text{ implies} \qquad y = 1$$

$$z \leq 0.5 \text{ implies} \qquad y = 0$$

The Decision Boundary is the "line" defined by z at z at z that separates the area where $y = 0$ and $y = 1$. The "line" defined by z at z at z for non-linear since the feature variables x_1 can be non-linear. The decision boundary can be any shape curved that fits the data. $z = w^T x$ $z = w^T x$ $z = w^T x$