***Logistic Regression from Scratch Objective***

The main objective of the Logistic Regression from Scratch is to predict the value of a variable based on the value of another variable. The variable we want to predict is called the dependent variable.

The variable we are using to predict the dependent variable’s value is called the independent variable.

The simplest form of the regression equation with one dependent and one

independent variable.

y = 1/(1+e\*\*(m \* x + b))

where,

• y = estimated dependent value.

• b = constant or bias.

• m = regression coefficient or slope.

• x = value of the independent variable.

Logistic Regression from Scratch

In this article, we will implement the Logistic Regression from scratch using only Numpy.

1. Understanding Loss Function

While there are many loss functions to implement, we will use the Mean Squared Error function as our loss function

A mean squared error function as the name suggests is the mean of squared sum of difference between true and predicted value.

As the predicted value of y depends on the slope and constant, hence our goal is to find the values for slope and constant that minimize the loss function or in other words, minimize the difference between y predicted and true values.

2. Optimization Algorithm

Optimization algorithms are used to find the optimal set of parameters given a training dataset that minimizes the loss function, in our case we need to find the optimal value of slope (m) and constant (b).

One such Algorithm is Gradient Descent. Gradient descent is by far the most popular optimization algorithm used in machine learning.

Using gradient descent, we iteratively calculate the gradients of the loss function with respect to the parameters and keep on updating the parameters till we reach the local minima

***Logistic Regression Intuition:***

It is a supervised model use in machine learning. It is mostly used to predict dependent variables in binary either 0 or 1(yes/no). Here we use sigmoid function.formula for sigmoid function: 1/(1+e^(-f(x))) Due to use of sigmoid function we can easily predict possibility. If a prediction have possibility of less than 0.5 and greater than 0 is equal to 0 and greater than 0.5 is equal to 1.

**Types of Logistic regression:**

Binary logistic regression:

It have only two classes in target feature 0 and 1 Ex:cancer prediction,detecting frauds,loan eligibility use in banks etc.

Multinomial logistic regression:

It have three or more classes. Ex:detecting whether animal eat meat,grass,or other item .student whether he go to college,trade school or into workforce like this. Ordinal logistic regression: It have three or more classes but according to ranking.

Usage:

Ranking restaurents between 0 to 5 stars,ranking colleges bases upon courses offered,podium results of Olympic. In three of this mostly use regression is binary.

***Sigmoid Function:***

A **sigmoid function** is a [mathematical function](https://en.wikipedia.org/wiki/Mathematical_function) having a characteristic "S"-shaped curve or **sigmoid curve**.

A common example of a sigmoid function is the [logistic function](https://en.wikipedia.org/wiki/Logistic_function) and is defined by the formula:[[1]](https://en.wikipedia.org/wiki/Sigmoid_function#cite_note-Han-Morag_1995-1)

{\displaystyle S(x)={\frac {1}{1+e^{-x}}}={\frac {e^{x}}{e^{x}+1}}=1-S(-x).}

Other standard sigmoid functions are given in the [Examples section](https://en.wikipedia.org/wiki/Sigmoid_function#Examples). In some fields, most notably in the context of [artificial neural networks](https://en.wikipedia.org/wiki/Artificial_neural_network), the term "sigmoid function" is used as an alias for the logistic function.

Special cases of the sigmoid function include the [Gompertz curve](https://en.wikipedia.org/wiki/Gompertz_curve" \o "Gompertz curve) (used in modeling systems that saturate at large values of x) and the [ogee curve](https://en.wikipedia.org/wiki/Ogee_curve) (used in the [spillway](https://en.wikipedia.org/wiki/Spillway) of some [dams](https://en.wikipedia.org/wiki/Dam)). Sigmoid functions have domain of all [real numbers](https://en.wikipedia.org/wiki/Real_number), with return (response) value commonly [monotonically increasing](https://en.wikipedia.org/wiki/Monotonically_increasing) but could be decreasing. Sigmoid functions most often show a return value (y axis) in the range 0 to 1. Another commonly used range is from −1 to 1.

A wide variety of sigmoid functions including the logistic and [hyperbolic tangent](https://en.wikipedia.org/wiki/Hyperbolic_tangent) functions have been used as the [activation function](https://en.wikipedia.org/wiki/Activation_function) of [artificial neurons](https://en.wikipedia.org/wiki/Artificial_neuron). Sigmoid curves are also common in statistics as [cumulative distribution functions](https://en.wikipedia.org/wiki/Cumulative_distribution_function) (which go from 0 to 1), such as the integrals of the [logistic density](https://en.wikipedia.org/wiki/Logistic_density), the [normal density](https://en.wikipedia.org/wiki/Normal_density), and [Student's *t* probability density functions](https://en.wikipedia.org/wiki/Student%27s_t-distribution). The logistic sigmoid function is invertible, and its inverse is the [logit](https://en.wikipedia.org/wiki/Logit) function.

## PROPERTIES:-

In general, a sigmoid function is [monotonic](https://en.wikipedia.org/wiki/Monotonic_function), and has a first [derivative](https://en.wikipedia.org/wiki/Derivative) which is [bell shaped](https://en.wikipedia.org/wiki/Bell_shaped_function). Conversely, the [integral](https://en.wikipedia.org/wiki/Integral) of any continuous, non-negative, bell-shaped function (with one local maximum and no local minimum, unless degenerate) will be sigmoidal. Thus the [cumulative distribution functions](https://en.wikipedia.org/wiki/Cumulative_distribution_function) for many common [probability distributions](https://en.wikipedia.org/wiki/Probability_distribution) are sigmoidal. One such example is the [error function](https://en.wikipedia.org/wiki/Error_function), which is related to the cumulative distribution function of a [normal distribution](https://en.wikipedia.org/wiki/Normal_distribution); another is the [arctan](https://en.wikipedia.org/wiki/Arctan) function, which is related to the cumulative distribution function of a [Cauchy distribution](https://en.wikipedia.org/wiki/Cauchy_distribution).

A sigmoid function is [convex](https://en.wikipedia.org/wiki/Convex_function) for values less than a particular point, and it is [concave](https://en.wikipedia.org/wiki/Concave_function) for values greater than that point: in many of the examples here, that point is 0.

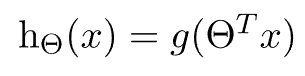
## COMMONLY USED EXAMPLES:-

1. [Logistic function](https://en.wikipedia.org/wiki/Logistic_function) [Hyperbolic tangent](https://en.wikipedia.org/wiki/Hyperbolic_tangent)

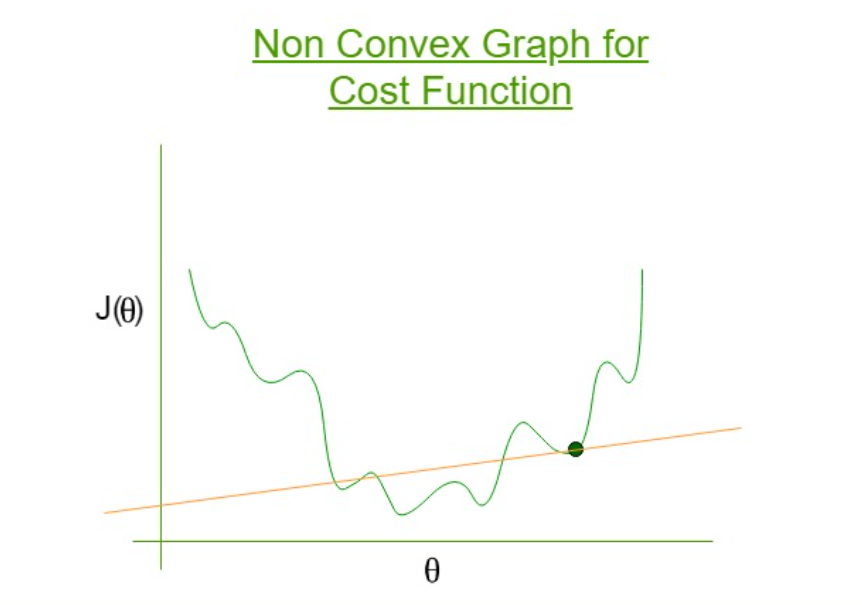
 

**COST FUNCTION OF LOGISTIC REGRESSION**

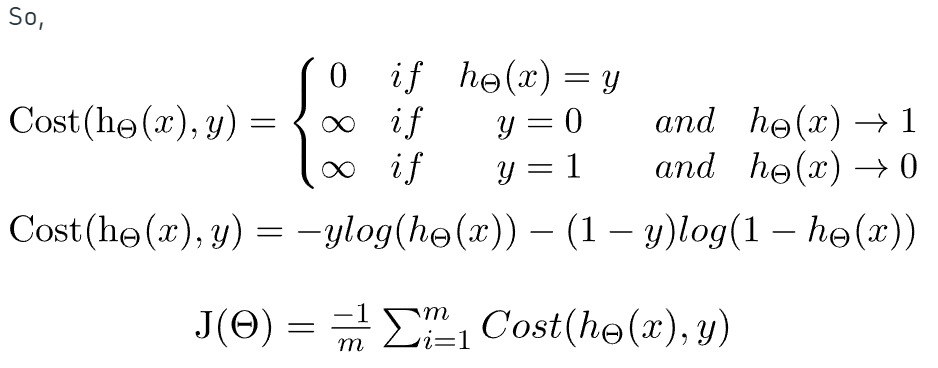
In the case of Logistic Regression, the Cost function is –



It will result in a non-convex cost function. But this results in cost function with local optima’s which is a very big problem for Gradient Descent to compute the global optima.

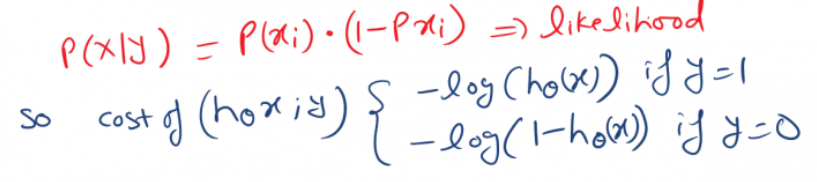


So, for Logistic Regression the cost function is defined as:



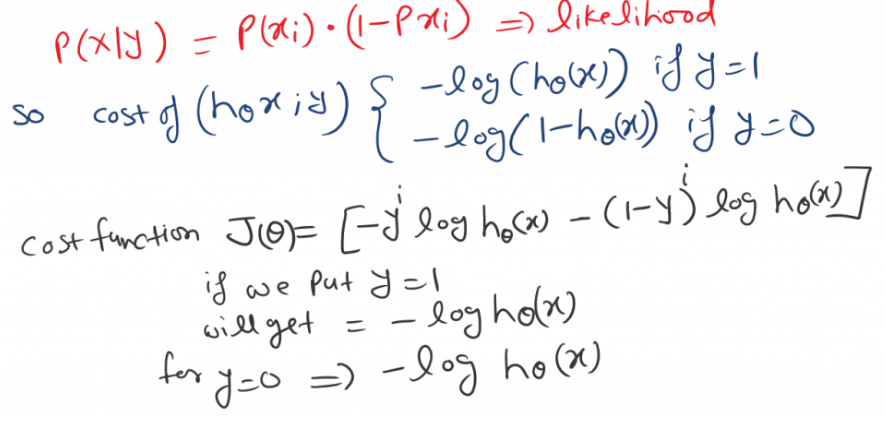
To fit parameter θ, J(θ) has to be minimized and for that Gradient Descent is required.

As we can see in logistic regression the H(x) is nonlinear (Sigmoid function). And for linear regression, the cost function is convex in nature. For linear regression, it has only one global minimum. In nonlinear, there is a possibility of multiple local minima rather the one global minima. So to overcome this problem of local minima. And to obtain global minima, we can define new cost function. We will take the same reference as we saw in Likelihood.

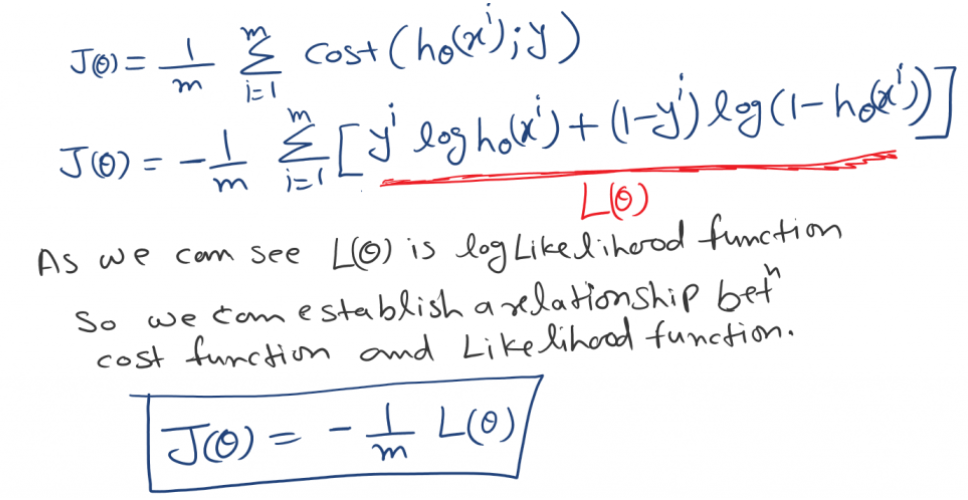


Cross entropy loss or log loss or logistic regression cost function. cross entropy loss measure the performance of the classification model. And the output is probability value between 0 to 1

So the cost function is as bellow.

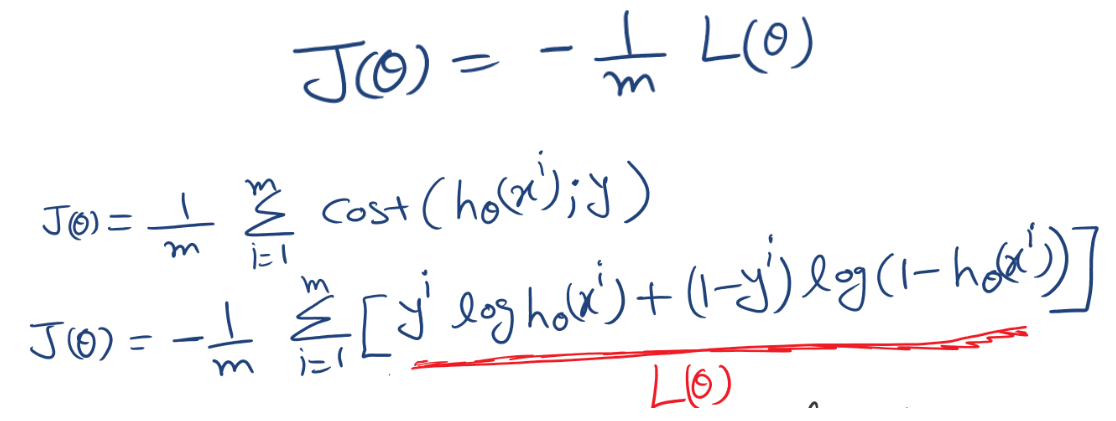


Now we can put this expression into Cost function



As we can see L(θ) is a log-likelihood function in above figure. So we can establish a relation between Cost function and Log-Likelihood function. You can check out Maximum likelihood estimation in detail.

Maximization of L(θ) is equivalent to min of -L(θ), and using average cost over all data point, out cost function would be.



Choosing this cost function is a great idea for logistic regression. Because Maximum likelihood estimation is an idea in statistics to finds efficient parameter data for different models. And it has also the properties that are convex in nature.

*Gradient Descent:*

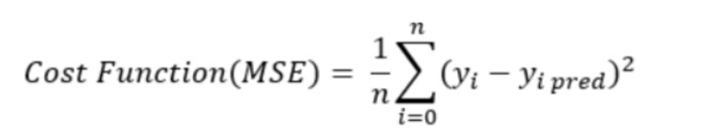
Gradient descent is an optimization algorithm used to minimize some function by iteratively moving in the direction of steepest descent as defined by the negative of the gradient. In machine learning, we use gradient descent to update the parameters of our model.

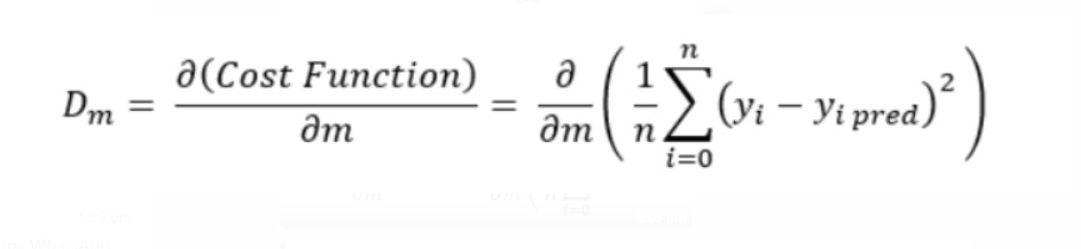
When there are more than one inputs you can use a process of optimizing values of coefficients by iteratively minimizing error of model on your training data. This is called Gradient Descent and works by starting with random values for each coefficient. The sum of squared errors are calculated for each pair of input and output variables.

A learning rate is used for each pair of input and output values. It is a scalar factor and coefficients are updated in direction towards minimizing error. The process is repeated until a minimum sum squared error is achieved or no further improvement is possible.

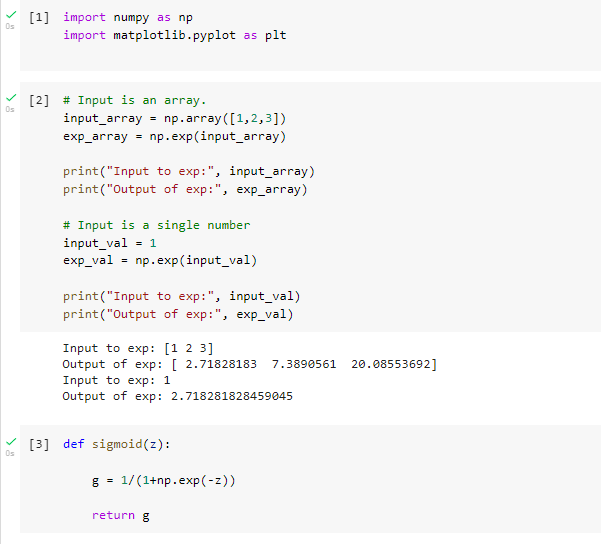
When using this method, learning rate alpha determines the size of improvement step to take on each iteration of procedure. In practise, Gradient Descent is useful when there is a large dataset either in number of rows or number of columns.

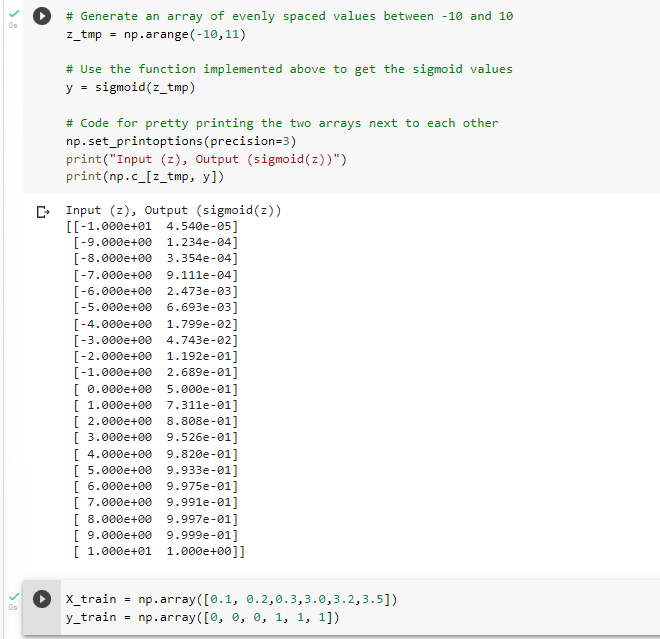
In short ,it is a minimization algorithm meant for minimizing a given activation function.

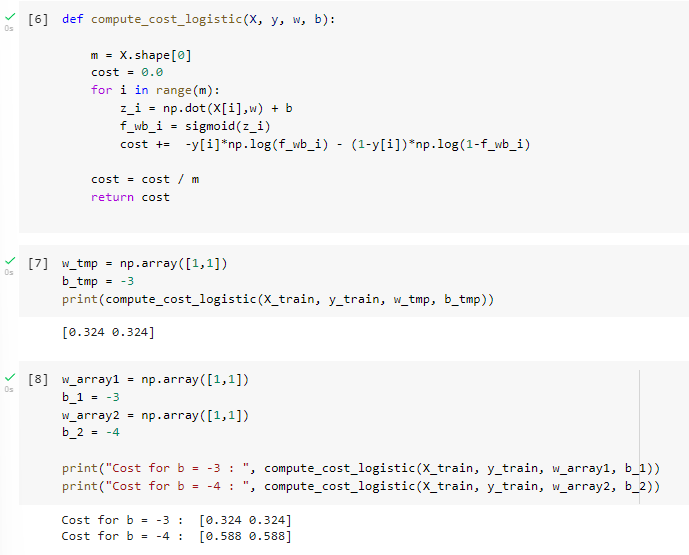


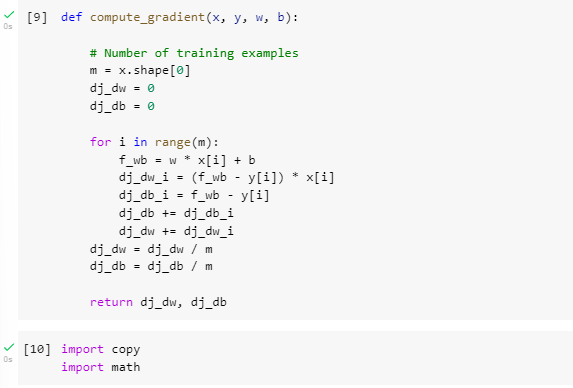


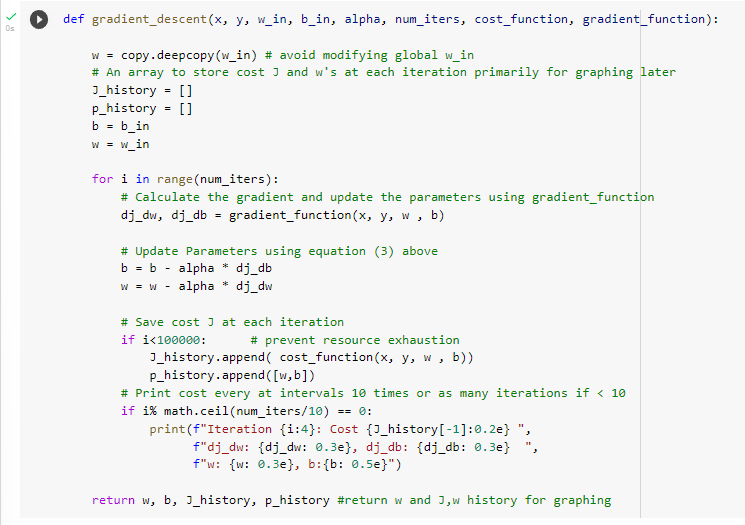
***CODE:***

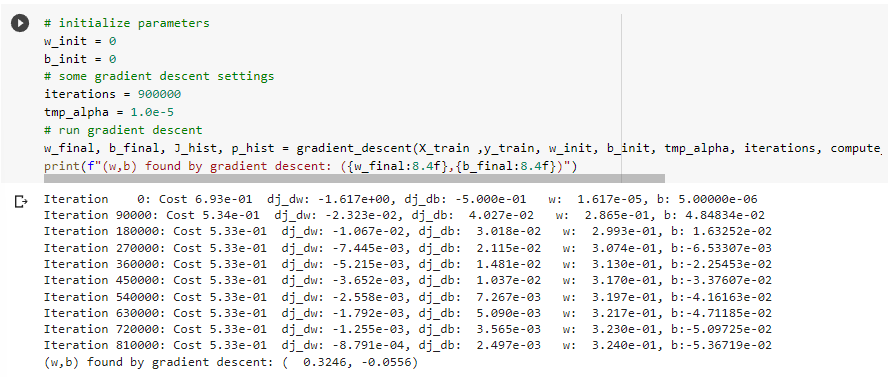












***References:***

[https://www.mastersindatascience.org/learning/machin e-learning-algorithms/logistic-regression/](https://www.mastersindatascience.org/learning/machin%20e-learning-algorithms/logistic-regression/)

<https://www.geeksforgeeks.org/understanding-logistic-regression>

<https://medium.com/@minions.k/gradient-descent-algo-d7b180605d7c>

<https://www.analyticsvidhya.com/blog/2021/04/gradient-descent-in-linear-regression/>