Logistic Regression

Logistic regression is a supervised learning classification algorithm used to predict the probability of a target variable. The dependent variable is binary in nature having data coded as either 1 (stands for success/yes) or 0 (stands for failure/no).

A logistic regression model predicts P(Y=1) as a function of X. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

Types of logistic Regression:

1) Binary/ Binomial: Two possible values for dependent variable, either 1 and 0. For example: Yes/No, Win/Loss, Present/Absent etc.

2) Multinomial: Dependent variable can have 3 or more unordered values. For example: Type1, Type2, type3 etc.

3) Ordinal: Dependent variables can have 3 or more ordered values. For example: 1/2/3, Low/Medium/High

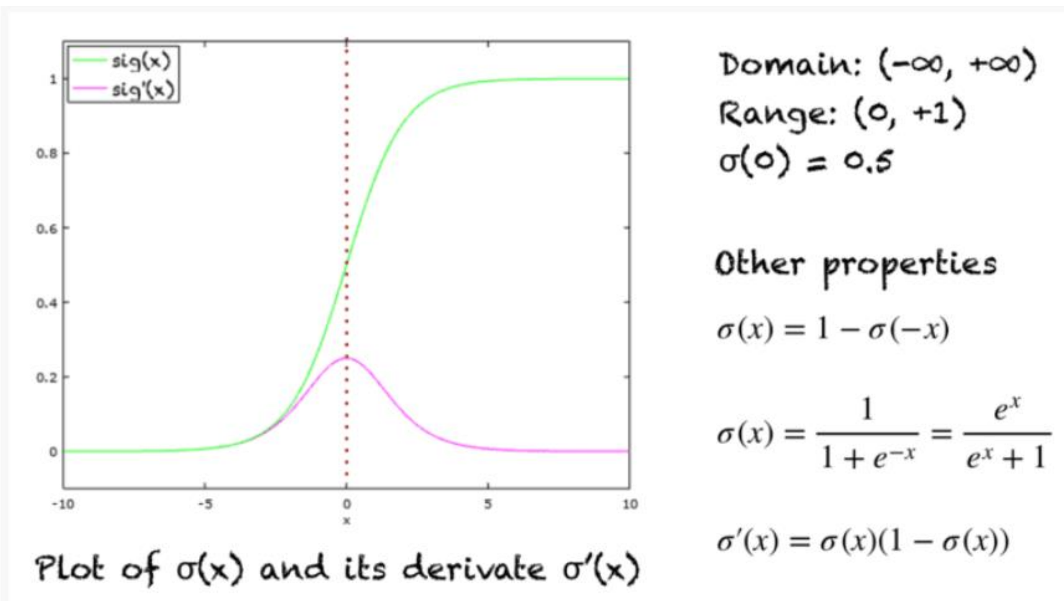
For linear regression, the dependent variable follows a normal distribution N (µ, s) where µ is a linear function of the explanatory variables. For logistic regression, the dependent variable, also called the response variable, follows a Bernoulli distribution for parameter p (p is the mean probability that an event will occur) when the experiment is repeated once, or a Binomial (n, p) distribution if the experiment is repeated n times (for example the same dose tried on n insects). The probability parameter p is here a linear combination of explanatory variables.

Logistic regression use function known as Sigmoid Function.

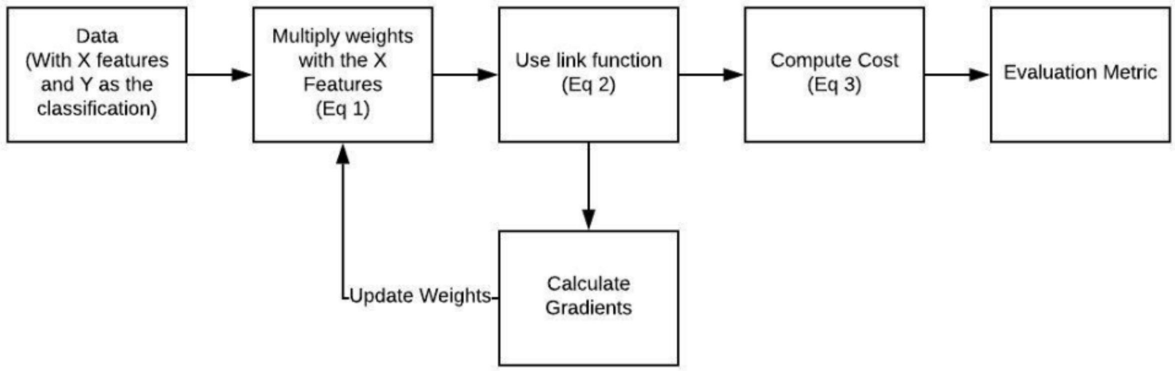
Sigmoid Function:
This function is used to map the predicted values to probabilities. It maps any input real value to another value between 0 and 1.

Range of Sigmoid: [0,1]

In Logistic regression, we use the concept of the threshold value. Value above threshold value tends to 1 and value below the threshold values tends to 0.



Logistic regression Algorithm:



Logistic regression Flow

1) Primarily, we create a weight matrix with random initialization. Then we multiply it by features.

$$a = w_0 + w_1x_1 + w_2x_2 + \ldots w_nx_n$$

2) We then pass this output a to sigmoid function.

$$y\hat{i} = 1/(1+e^{-a})$$

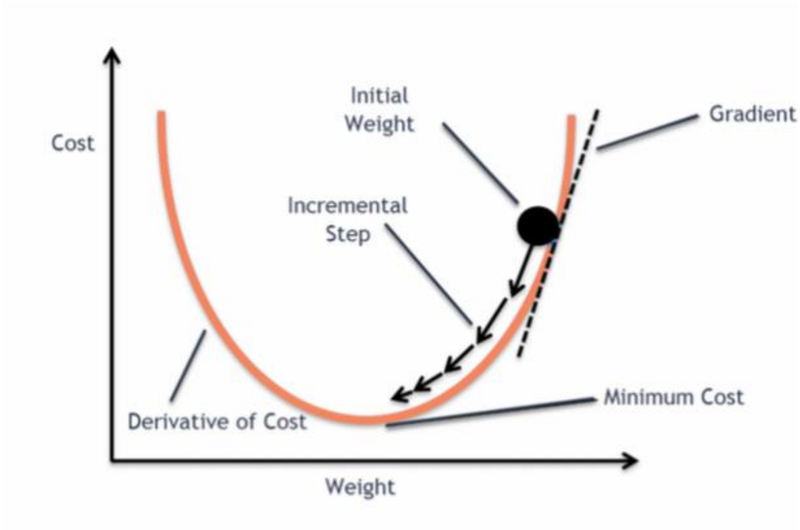
3) Calculate the cost function: [Cross entropy error]

$$cost(w) = (-1/m)\sum_{i=1}^{i=m} y_i log(\hat{y_i}) + (1-y_i) log(1-\hat{y_i}))$$

Now, our main aim is to minimize this cost function. For this we basically use gradient descent method.

Gradient Descent:

Gradient descent is a first-order iterative optimization algorithm for finding a local minimum of a differentiable function.



In order to determine minimum cost, we will find derivative of this cost function, check for few conditions:

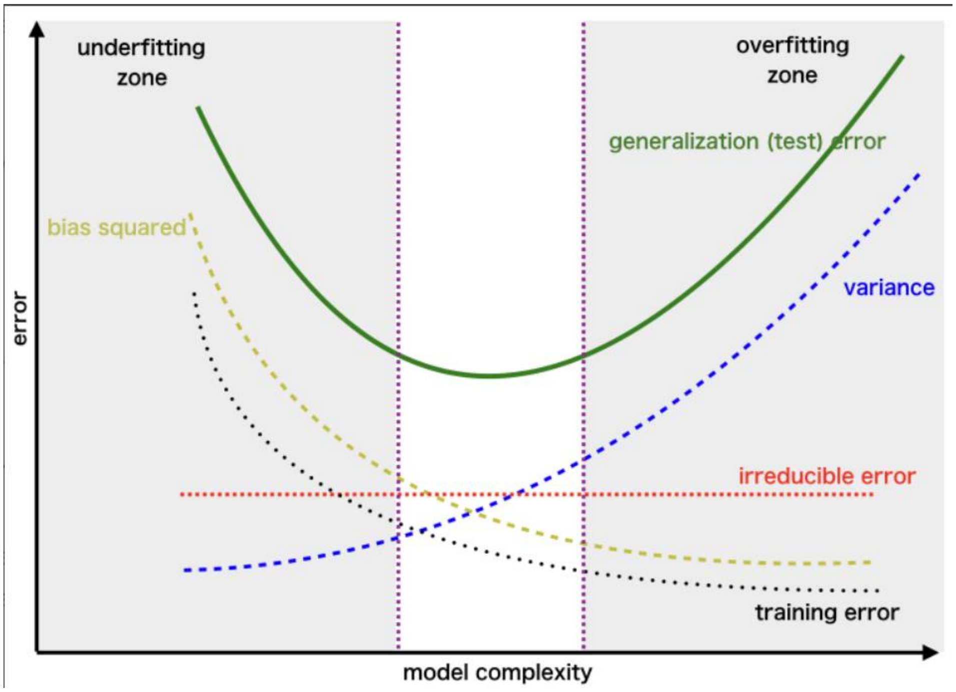
- Check if the derivative is +ve or -ve. If it is positive, it means in order to minimize cost, we have to decrease value of weight. Similarly, if derivative is negative, we have to increase value of weight.

There are two more important terms related to cost function (test error): **Bias and Variance**

Bias: It is the difference between the prediction of the values by the ML model and the correct value.

High bias means underfitting and low bias means overfitting.

Variance: Tells us about the spread of our data (variability of data)
High Variance means overfitting and low variance means underfitting.



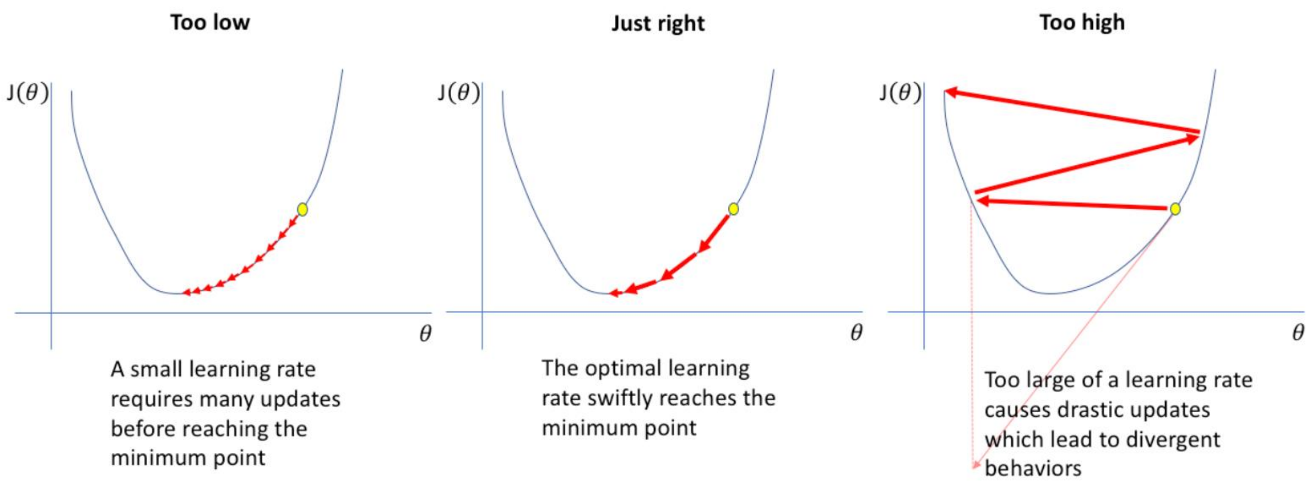
4) Find derivative of this cost function

$$dw_j = \sum_{i=1}^{i=n} (\hat{y} - y) x_j^i$$

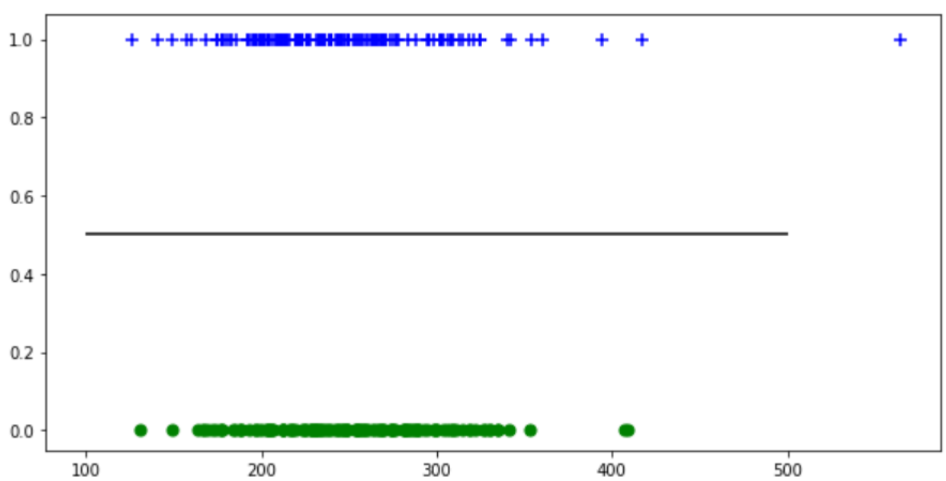
5) Update weight:

$$w_i = w_j - (\alpha * dw_j)$$

Here, α is learning rate. It should neither be very high nor too low. Otherwise, it will take large amount of time to converge.



Now, the goal of the logistic regression algorithm is to create a linear decision boundary separating two classes from one another. This decision boundary is given by a conditional probability.



The line in between is the decision boundary with two classes above and below it.

Let us assume that the class above the black line (decision boundary) i.e. the *'+'* is classified as *'1'* and class below the decision boundary *'o'* is defined as *'0'. What logistic regression does is that it calculates a conditional probability i.e.*

$P(y = 1 \,|x; w)$, Probability for class '1'
$P(y = 0 \,|x; w)$, Probability for class '0'

$P(y = 1 \,|x; w)$ + $P(y = 0 \,|x; w)$ **= 1**

If sigmoid of input function is greater than 0.5, it will predict output as 1.
If sigmoid of input function is less than 0.5, it will predict output as 0.
If sigmoid of input function is equal to 0.5, it can predict 0 or 1.

Reference:

https://www.analyticsvidhya.com/blog/2021/08/conceptual-understanding-of-logistic-regression-for-data-science-beginners/