Logistic Regression:

Abstract:

Logistic Regression is a very versatile and important form of machine learning, it is valuable as the basic form of machine learning used in industry, and as a baseline of performance of different models.

Binary Classification Problems:

Where the prediction is between 2 discrete classes: {0,1}.

How?

We use a sigmoid function to compress the result of a model into values within 0 and 1:

Where x can be any model, polynomial regression, linear regression or whatever.

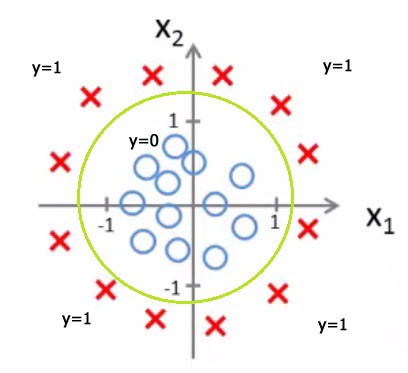
The resultant graph:



The range of output that we receive from the function: 0 <f(x) <1

Is the probability: P(y = 1|x;0)

If we plot a graph with respect to the features:



Where the circle represents the decision boundary that the model takes to distinguish between 2 classes.

Now for the implementation:

Given a model g(x), logistic function will compose over it:

The cost function:

Recap, a cost function is to determine how well a model performs.

The cost function for Logistic Regression is:

The gradient is derived:

The gradient descent function:

More sophisticated algorithms asides from Gradient descent:

* Conjugate Gradient
* BFGS
* L-BFGS

Advantages:

* is selected for you based on the algorithm.

Disadvantage:

* More complex and hence harder to implement directly, instead use modules based on code optimized by experts in numerical computation.

Multiclass Classification:

While Logistic Regression itself is a binary classification algorithm, it can be extended to be used to distinguish multiple classes, through a One-vs-rest classification.

The model thus is a combination of k number of logistic models based on k number of classes.

And thus the output that is picked is the function that maximises the output value:

Regularization:

Regularization is a method used in machine learning to reduce the size of the parameters and thus reduce the possibility of overfitting a ML model.

A underfit model is a Biased model

A overfit model is a high variance model

Overfitting occurs when there are more features than there are datapoints in a model.

Hence this is done through penalizing the size of the parameters .

Note that is not affected by this penalization.

This effect is controlled by the value of lambda.

If lambda is too high, the model becomes underfit, and if lambda too low, the model overfits the data.

Regularization in Logistic Regression:

Cost Function:

Gradient Function: