Optimization Techniques

Individual Assignment – Eduardo Razo

This document aims to describe and explain **Gradient Descent** as optimization technique, applied on a Generalized Linear Model **(logistic regression)** to optimize the likelihood of a classification problem. In this case, estimating the direction (up or down) of the stock market based on the daily percentage returns for the S&P 500 stock index.

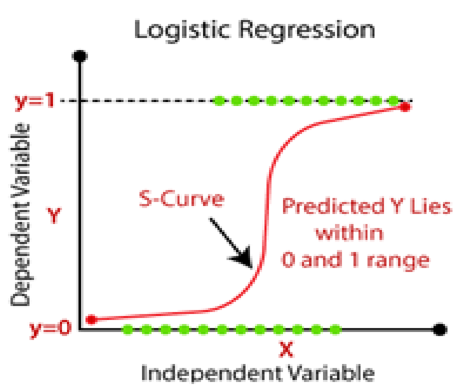
What is logistic regression?

It is one of the most popular Machine Learning algorithms which come under supervised learning technique. We leverage this methodology when the target variable is categorical, and it is necessary to solve a classification task. It is used to predict the dependent variable based on the independent variables contained in a dataset. The output of the logistic regression should be categorical such as 0 or 1, Yes or No.

For example: Whether the stock market is going to go up (1) or down (0).

It is worthy to mention that the logistic regression is a linear method, but the predictions are transformed using the sigmoid function, which is a mathematical function that has a characteristic "S"-shaped.

Logistic regression becomes a classification technique only when a decision threshold is set. The setting of the threshold value is a particularly important aspect of logistic regression. The decision for the value of the threshold value is majorly affected by the values of precision and recall. Ideally, we look for precision and recall to be 1.



Logistic function

$$f(x) = \frac{1}{1 + e^{-x}}$$

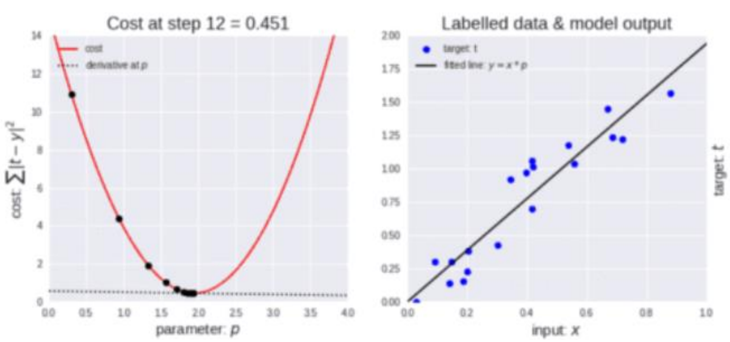
What is gradient descent?

Gradient Descent is one of the most popular techniques used to minimize a function by following the gradients of the cost function.

This involves knowing the form of the cost as well as the derivative so that from a given point someone can know the gradient and can move in that direction, e.g. downhill towards the minimum value.

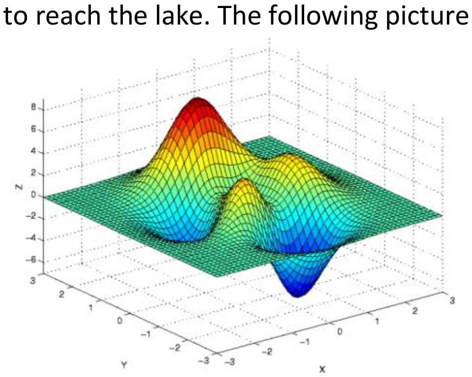
In machine learning, we can use a technique that evaluates and updates the coefficients every iteration, called gradient descent to minimize the error of a model on our data. The way this optimization algorithm works is that each training instance is shown to the model one at a time. The model makes a prediction for a training instance, the error is calculated, and the model is updated in order to reduce the error for the next prediction. This procedure can be used to find the set of coefficients in a model that result in the smallest error for the model on the data.

In these graphs we can observe first how gradient descent iterates from the initial point and then goes down to minimize the cost, and how the model fits the optimal line among the data in the classification problem.



To understand in a simpler way, we can think of an example and suppose we are at the top of a mountain, and we must reach a lake which is at the lowest point of the mountain. A condition is that we have zero visibility to see where we are headed. So, what approach will you take to reach the lake?

The best way is to check the ground near us and observe where the land tends to descend. This will give an idea in what direction take the first step. If we follow the descending path, it is highly likely to reach the lake. The following picture illustrates this idea.



Implementation of gradient descent

For this project, I decided to address a classification task and optimizing the corresponding likelihood associated to the result using the gradient descent algorithm.

The classification problem consists of predicting whether the direction of the stock market will go up or down based on the daily percentage returns for the S&P 500 stock index. The data set is taken from the available data sets In R.

Data description

Variable	Description
Year	The year that the observation was recorded
Lag 1	Percentage return for previous day
Lag 2	Percentage return for 2 days previous
Lag 3	Percentage return for 3 days previous
Lag 4	Percentage return for 4 days previous
Lag 5	Percentage return for 5 days previous
Volume	Volume of shares traded (in billions)
Today	Percentage return for today
Direction	A factor with levels Down and Up indicating whether the market had a positive or negative return on a given day

Data processing

It only was necessary to create an additional variable (Dir) to obtain a dummy variable (1=”Up”, 0=”Down”) indicating the direction of the market and in order to be able to execute the code.

Gradient descent model

The first steps were the definition of the sigmoid function, which is the main component in logistic regression as describe in the first paragraph of this document, and the definition of the cost function which we want to minimize.

Secondly, the gradient descent function was defined, however as a previous activity in this same step, it was necessary to define the gradient descent parameters such as learning rate which controls how large of a step we take downhill during each iteration, the number of iterations and beta.

The detail of the implementation code can be found in the R file submitted together with this document.

Results

The output of the function is the values of the parameters (coefficients) that are part of the linear regression function and in this case, the ones that minimizes the cost function. The parameters are the intercept, the beta values for each independent variable and latest cost.

In this problem the target variable (Y) was **Dir (1 =" Up", 0 = "Down")** and the independent variables or predictors were **Lag 1, Lag 2, Lag 3, Lag 4, Lag 5** and **Volume.** The gradient descent function found the optimal value for the cost function at iteration 4000

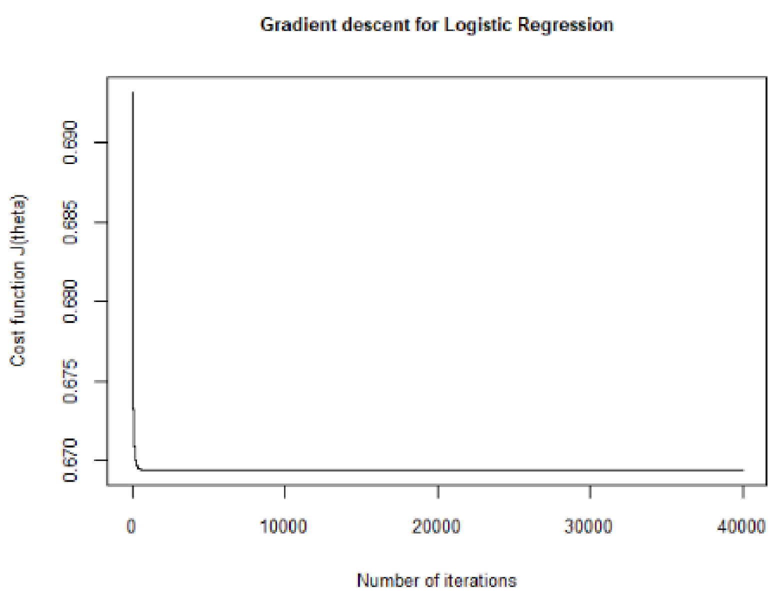
Outputs

Intercept
-1.423

Coefficients (beta)
-0.073, -0.214, -0.015, -0.084, 0.362, 0.777

Latest cost
0.66944

The optimal cost and the number of iterations can be observed graphically as follows:



Finally, as validation, I ran the conventional logistic regression model considering the same variables and the result was the same set of coefficients which are the same than the obtained with the gradient descent model.

Group assignment evaluation

Kurt Kusterer grade = 16
Mario Castillo grade = 16
Alejandro López grade = 16
Xinghao Zong grade = 16
Eduardo Razo grade = 16

References

https://en.wikipedia.org/wiki/Logistic_regression

https://medium.com/meta-design-ideas/linear-regression-by-using-gradient-descent-algorithm-your-first-step-towards-machine-learning-a9b9c0ec41b1