\_\_\_\_\_\_\_\_\_\_\_\_Regression\_\_\_\_\_\_\_\_\_\_\_\_

# Introduction

Regression is the statistical process of determining the relationship between one or more independent variables and a dependent variable. In Machine Learning, it is used for forecasting and making predictions. Regression can be classified into 3 types:

* Linear Regression: This involves fitting a best fit line, or a function of degree 1 to a given data set to make accurate predictions.
* Polynomial Regression: Like Linear Regression, except we fit a function of degree 2 or more to the data set.
* Logistic Regression: Different from the previous two types, it is used to classify data points.

# Linear Regression

The objective of Linear Regression is to find a linear relationship between one or many independent variables and a dependent variable, to predict outcomes within a range. The simplest of Linear Regression is Univariate Linear Regression. Here we find the best fit line of the form:

where y = Dependent Variable, x = Independent Variable, m = slope of the line, and c = intercept. This line must be the minimization of the cost function, which will be defined in a further section. There can also be multivariate linear regression, where using multiple variables, the dependent variable can be predicted. The equation to be solved for multivariate linear regression, with *t* variables is:

Here is the prediction of the linear regression model. As convention, we take , and its coefficient becomes the constant term. Therefore, is the intercept of this function. The input of the function is a matrix of the form:

[]

Notice how is not inputted as we have already defined its value to be 1.

Chart, scatter chart

Description automatically generated

Figure

Figure 1 shows a scatterplot that shows the dependence of the price of a car on its horsepower. The line formed is the best fit line, whose intercept, c, is equal to -3664.57488879, while the slope, m, is 162.57053957.

# Cost Function

The Cost Function, , of a Linear Regression problem is the mean squared value of the error between predicted value, , and actual values at the point where , given by , of the whole dataset. For Univariate Linear Regression, it is given by:

Here, *n* is the total number of data points. Now, a question arises: Why is the cost function the mean squared value, and not just the mean? The answer to that question lies in outliers. A mean squared function accounts for outliers much better than a mean function does. The multiplied in front of the mean squared function makes it the equation easier to solve, when we compute the best fit line.

Chart, line chart

Description automatically generated

Figure 2

In figure 2, note how the orange line (which was obtained through a mean squared C.F.) caters better to the outlier at point D than the green line.

# Gradient Descent

Gradient descent is a method used to determine the values of . It is iteration based and determines the minimum value of the cost function.

Chart, line chart

Description automatically generated

Figure 3

Take the case in figure 3. Assume in this equation. Here the y-axis denotes the cost function, C(m), for a univariate linear regression problem, while the x-axis denotes the slope for the regression line. We know that the slope for the best fit line exists at the point where C.F. is minimum. Therefore, we need to move towards that point from an initial point (h, k). How do we find this minimum? By applying the gradient descent algorithm.

To understand this algorithm, proceed step-by-step. We define the current point to be the point on the curve where the value of x is the current value of . We initialize a value for which usually is kept as zero. It keeps updating after each iteration of the algorithm. Next, we look at the term α, which is also called the learning rate. The learning rate is the parameter that decides the step size when moving towards the global minimum (there is only one minimum in a linear regression model). It is a constant value and does not change throughout the whole process. Finally, we look at the partial derivative of the cost function w.r.t . This is basically the slope of the line the current point in the loop. As the current point gets closer to the minimum, the slope gets smaller. This reiterates the fact that α remains constant for the whole process. If the initial point is set as the minimum, the slope will be zero, and the point will not shift.

**Note**: The value of α should be set optimally. If it is too low, the algorithm will take too long to complete. If it is too high, the value of may never converge at the minimum (see figure 4).

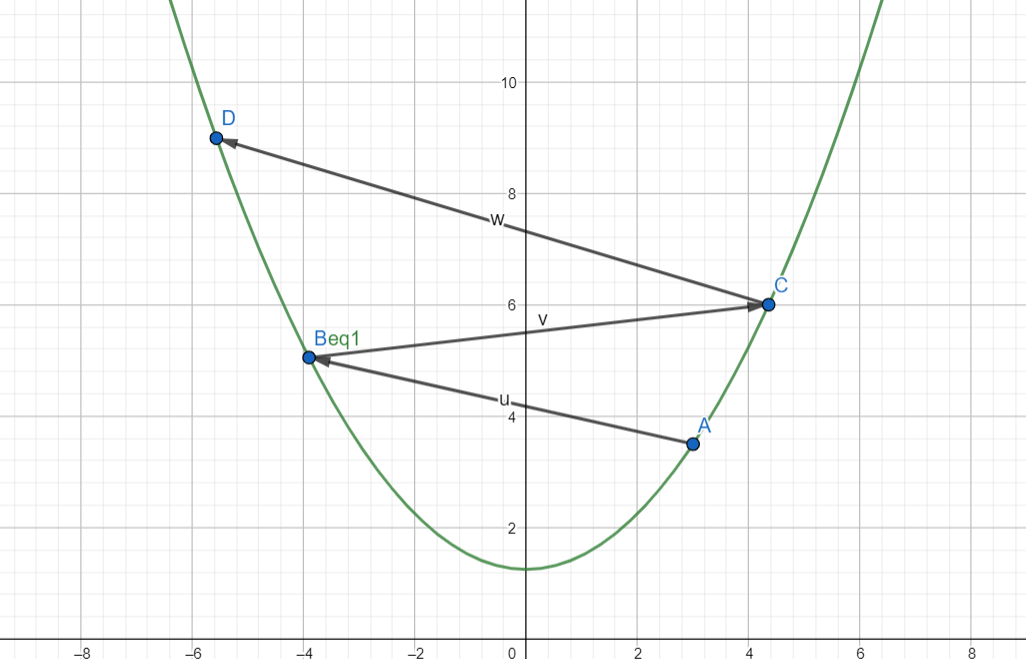


Figure 4

# Logistic Regression

Logistic regression is the model used to classify and predict the probability of an object to belong to that class in a classification problem. Unlike linear regression, it is used when the dependent variable is categorical. For example, in a dataset containing the characteristics of cars and their price, we use linear regression to predict their prices, while we may use logistic regression to check the type of fuel system it uses. There are 3 types of logistic regression:

* Binary Logistic Regression: Classifies into 2 categories; positive (1) or negative (0)
* Multiple Logistic Regression: Classifies into multiple, unordered categories. For example, into species of iris flower.
* Ordinal Logistic Regression: Classifies into multiple, ordered categories. For example, into Low, Medium, or High.

# Sigmoid Function

To find the probability of a certain data point belonging in a category, we need to first map the whole dataset into a function that lies between zero and one. This can be achieved by a sigmoid or a logistic function:

Here, y(x) is a function containing the independent parameters of the model. It is of the form . The decision boundary will be discussed in the next section. As seen in figure 1, the value of this function always lies between zero and one. The value S(x) is the probability of the object belonging in the positive class (1).

Chart, line chart

Description automatically generated

Figure 1

# Decision Boundary

This function sets a tipping point to classify the object into a class. In a binary classification problem, this boundary function is set at S(x) = 0.5, greater than which, a positive value is taken, and below which, a negative value is assumed. This boundary can be linear or non-linear, depending on the arrangement of data points. The decision boundary is a characteristic of the function y(x). It does not depend upon data points.

# Cost Function

In logistic regression, the cost function is used to find the parameters of y(x) (the parameters m). Recall the cost function, C(m), for linear regression:

Here, define the term as equal to Cost(y(x), k). If we use this cost function for logistic regression, we get a non-convex function. However, since we use gradient descent to solve for the values of m, this is not optimal, since the function should have only one minimum for gradient descent to work. Therefore, for logistic regression, we define the Cost(h(x), y) as follows:

Note how the value of k can be either zero or one only as this is a classification problem. The simplified form of the cost function is as follows:

This function is a convex function, and we can minimize this through gradient descent. Gradient descent for logistic regression works almost exactly like that for linear regression. The algorithm is the same and the only real difference is the cost function we have to minimize, which has already been defined.

# Overfitting and Underfitting

Overfitting and underfitting are two of the biggest causes of poor model performance. Choosing the degree of the function often involves a variance-bias trade off. And in many cases, this leads to a huge tilt towards one or the other, making the model inaccurate.

* Overfitting: When the function corresponds too closely with the given data points, it is known as overfitting. This is the high variance case. The problem that arises here is that the function may fail to predict future observations accurately, as these maybe outliers. Here the order of the chosen function may be too high. See figure 5 for an example.
* Underfitting: Here the function can neither model given data, nor can it predict future observations. Here the order of the given function may be too low. This is the high bias case. See figure 5 for an example.

Chart, scatter chart

Description automatically generated

Figure 5

# References

* [Machine Learning by Stanford | Coursera](https://www.coursera.org/learn/machine-learning)
* [Holehouse.org](http://www.holehouse.org/mlclass/01_02_Introduction_regression_analysis_and_gr.html)
* [Stanford CS229: Machine Learning](http://cs229.stanford.edu/syllabus.html)
* [Towards Data Science | Medium](https://towardsdatascience.com/machine-leaning-cost-function-and-gradient-descend-75821535b2ef)
* [Mathematics for Machine Learning by Deisenroth, Faisal and Ong](https://mml-book.github.io/)
* [Machine Learning Mastery](https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/)
* [Wikipedia](https://en.wikipedia.org/wiki/Logistic_regression)
* [StatQuest | YouTube](https://www.youtube.com/watch?v=yIYKR4sgzI8)