Solving Numerical Optimization Model of Neural Network

Weam Abbas Obaid
University of Babylon, wiaamabbas123@gmail.com

Dr. Ahmed Sabah Ahmed Aljilawi
University of Babylon, aljelawy2000@yahoo.com

Abstract

Neural networks are usually trained using gradient-based methods, in this paper the attempts to reduce the size of the cost function in order to ensure that it is suitable for any monitoring or observation provided, where the model adjusts its weight and slope, so it uses the cost function and enhanced learning to reach the convergence point or the local minimum. Gradient descent is the process by which algorithm adjusts its weights, allowing the model to determine the trend towards reducing errors(reducing the cost function) with each training of the model, the model coefficients are adjusted to gradually converge to a minimum.

Keywords: *Numerical Optimization, Neural Network, Algorithm, Learning rate, Cost function.*

1-INTRODUCTION

The neural network is a system with interconnected node consisting of several layers, namely the input layer, the hidden layers and the output layer.

Neural networks perform their tasks like neurons in the human brain by certain algorithms and those algorithms recognize the hidden patterns in the data and divide them into group and classify them and over time those networks learn and their performance gragually improves [1].

In 1943,McCulloch(a scientist specializing in neuro-science) and Bates (in mathematics) created a neuron model that acts as a switch in an electrical circuit that receives inputs from other neurons and the activity of the cell depends on the total weight of these inputs they called this model (logic threshold) this model paved the way for neural network research to divide it into two distinct methods, one focused on biological processes in the brain and the other on the application of neural networks to artificial intelligence[2].

Then in 1960, a study proved that this model of neural networks has cell-like properties in the brian, ht can recognize a variety of patterns, and although these networks are model up of artificial neurons, they continue to work even if some of those cells are destroyed[2].

The original goal of this neural network is to create a computer system that has the ability to simulate the human brain in solving problems, so neural networks important for their remarkable ability to extract meaning from complex and inaccurate data, which gives them the ability to understand patterns and observe tendencies that neither humans nor other computing technologies can notice, so they help humans solv complex problems in their daily lives[3].

These networks can also learn and model complex and non-linear relationships between data inputs and outputs, in addition, they make generalizations and inferences to reveal hidden relationships between inputs and outputs, patterns and predictions.

Now a neural network is a system that learns how to make predictions by following these steps.

1// Taking input data.

2//Making a prediction.

3//Comparing the forecast with the desired output.

4//Modifying the vectors to oredict correctly (more accurately) the next time.

The process continues until the difference between the expectation and the required goals becomes minimal, knowing when to stop training and what the accuracy target should be set is an important a spect in training neural networks[4].

2-Optimization:

Whatever the real-word problem may be, it can be expressed as bounded optimization peoblems in the following general form.

Maximize/minimize

$$f(x) \ ,x \in R^n \qquad (1)$$

where $x = (x_1, x_2, \cdots, x_n)^T \in R^n$

Subject to $g_i(x) = 0\ ,\quad (i = 1,2, \dots, N)$

$$h_j(x) \geq 0, \quad (j = 1,2, \dots, M)$$

Where $f(x), g_i(x)$ and $h_j(x)$ are scalar function of the real column vector x.x_i is the components of $x = (x_1, x_2, \dots, x_n)^T$ are called design variables or decision variables,they can be either continuous, intermittent or mixed the vector x, is called a decision vector which of n-dimensional space R^n.The function f(x) is called objective function or cost function, g_i (x)are constraints in terms of N equalities, and h_j (x) are constraints in terms M inequalities. So there are N+M constraints in total[5].

The objective functionf(x) can be either linear or nonlinear. If constraintsg_i (x) and h_j (x) are all linear, it becomes a linearly constraints problem(a linear programming problem). If objective function is at most quadratic and the constraints is linear, then it is called quadratic programming. If all values of the decision variables can be integers, then a linear programming is called integer programming or integer linear programming[6].

3-The cost function:

It is the difference between the expected value and true value to achieve the least error ratio(minumim cost). The cost function is used to calculate the loss based on the expectations made in the linear regression and the mean squared error(MSE) is used to calculatethe loss the equation is as follows[7].

$$minimize\ \frac{1}{m} \sum_{i=1}^{m} \left(y_{true} - y_{pred}\right)^2 \qquad (2)$$

where m number of input,$y_{true}$the real value, y_{pred} the value to be predicted.

4-Mean Squared Error(MSE):

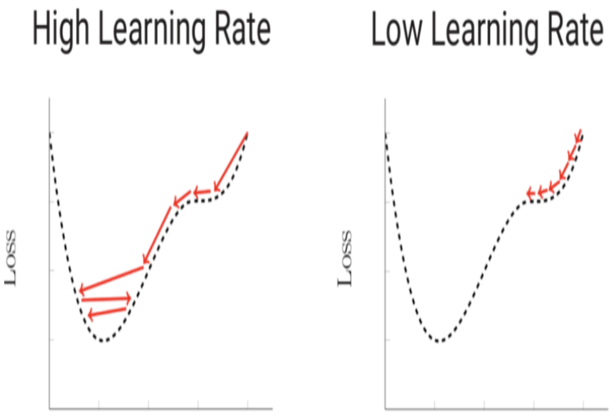
The mean squared error is the sum of the differences between the expected and true values[4].

5-Learning rate:

This update is very important because it represents the core of machine learning applications and determines how quickly the model reaches the convergence point the point where the error is perfect=0), and is also called the step length(iterations).

Therefore, if the learning rate(step length) is large,the steps are fast and less accurate, but if it is small, the steps are slower and more accurate[8], as in the figure(1).

Fig(1) (Learning Rate)



6- Independent variables:

Independent variables are variables that do not depend on any other variable in the scope of a given experiment, such as time, spase, density and mass, or are variables that are not affected by any other variables called prediction variables independent variables[9].

7- Dependent variable:

Is a variable that changes as a result of a change an independent variable is a variable whose value is studied according to an assumption, law or rule depending on the values of other variables[9].

8- Linear Neural Network:

The simplest type of neuron network is a linear neural network, which is used for problems such as logistic regression, polynomial regression, linear regression. Machine learning techniques are often used[10].

8.1-Linear Regression:

We use regression to explain the relationship between one or more independent variables (denoted by X) (input) and a dependent variable (denoted by Y) (output) and predicts. the output of all regression problems is a real number (y ∈ R).

The idea of regression we have data related to each other and I want to identify new data values that can be applied in stock prices on the stock exchange, house prices, the amount that the client will buy, the weather[10].

Must use the following in order to take advantage of regression.

Features are input data that are either numerical values or vectors. There are several pairs (x_i,y_i) in the training examples, which represent the output for each input.

Feature (model). This describes how the inputs and outputs are related.

The cost (loss) function or objective function of our model measures its accuracy.

Reduction of the cost (loss) function or objective function through optimization

It is also called one-variable regression or unvariate regression[10].

In linear regression many points are given (x_i , y_i) . Finding a font that complements the identified data is our aim (the most suitable font), the most important value in the linear equation is the slope, if the value of the slope is positive, the straight line is upward, if it is negative, the straight line is downward, if the straight line is horizontal, the slope equal zero and if it is vertical, the slope equal infinite (an unknown quantity)[10].

8.2-Linear Equation:

Suppose we have the following linear equation.

$$y = mx + c \qquad (3)$$

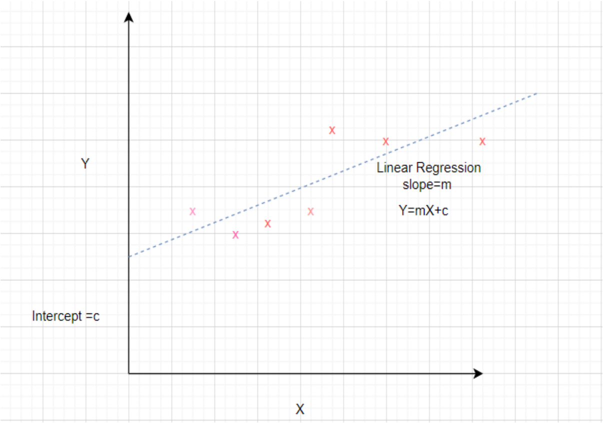
Where m : is the slop (or gradient).

c : is the point of intersection with the $y-axis$.

We found the slop using two point on the line using the equation.

$$m = \frac{y_2 - y_1}{x_2 - x_1} \qquad (4)$$

Fig (2) (Linear equation)



Alinear equation is an equation whose graph is a straight line, and it is a relationship between two variables that are both of the first order[11], figure (2).

8.3-Mathematicals Model of Application Based on Linear Regression:.

The linear regression equation can be written as follows and is similar to the equation of a straight line [12][13].

Hypothesis: $h_{\theta(x)= \theta_0+\theta_1 x}$ . . . (5)

Parameters: θ_0 , θ_1

$$\text{Cost function: } j(\theta_0 , \theta_1) = \frac{1}{2\,m} \sum_{i=1}^{m} (h_\theta(x_i) - y_i)^2 \qquad (6)$$

$$\text{Objective function: } minimize_{\theta_0 , \theta_1} \; j(\theta_0 , \theta_1)$$

Where θ_0 , θ_1 hypothetical values.

$h_\theta(x_i)$ is the expected value of linear equation, y_i is the real value.

The goal is to minimize the difference between the value of $h_\theta(x_i)$ and the value of y_i,and find the value θ_0, θ_1 that make the cost function (cost error function) as low as possible divide by $2\,m$ to relate the error value to the number of sample values[10].

Example (1)

Let's take the linear equation and the following data.

$$h(x) = \theta_0 + \theta_1\, x \qquad (5)$$

$$h(x) = 5 + 2\,x$$

x	y	$h(x)$	$h(x) - y$	$(h(x) - y)^2$
1	7	7	0	0
2	8	9	1	1
2	7	9	2	4
3	9	11	2	4
4	11	13	2	4
5	10	15	5	25
5	12	15	3	9

Where $\theta_0 = 5, \theta_1 = 2$, x : input, $m = 7$ number of input, y : expected value(output), $h_\theta(x)$: the true values obtained from the offset of the values of x (input)by the linear equation $h_\theta(x)$, $h_\theta(x) - y$ the difference between expected value and true value , $(h_\theta(x) - y)^2$ the difference square of them,

$\sum_{i=1}^{m}(h_\theta(x_i) - y_i)^2$ sum squared error.

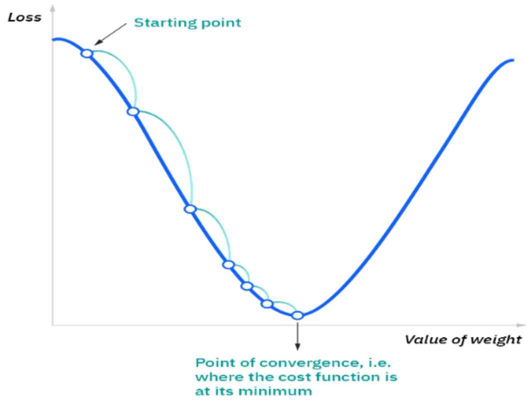
$$j(\theta_0 , \theta_1) = \frac{1}{2\,m} \sum_{i=1}^{m}(h_\theta(x_i) - y_i)^2 \qquad (6)$$

$$j(\theta_0, \theta_1) = \frac{1}{2(7)}\,(47) = 3.3$$

When choosing random values for θ_0, θ_1 the error value is very large, the error value is too large. θ_0, θ_1 values must be updated continuously to reduce the error.

The closer the imposed values are to the real values, the lower the error m meaning that we reach the optimal solution, see figure(3)

Fig(3) (The optimal solution)



The a bove steps can be applied using the python programming language terms of data represention and drawing the line fit (the most a ppropriate line) when the imposed values are close to the real values[8].

Now : How to choose value θ_0 , θ_1 ?

The equation of the straight line $y = mx + c$ is similar to $h(x) = \theta_0 + \theta_1\, x$

Where θ_0 : it represents the point of intersection with $y - axis$.

θ_1 : it is slope.

θ_0 , θ_1 it determines the most a ppropriate line (the smaller the differences, the lower the error rate, and vice versa).

To demonstrate how to choose θ_0 , θ_1. Iet's discuss the following examples.

Example (2):.

Let $x = 1 , 2 , 3 \quad y = 1 , 2 , 3 \quad m = 3$

x	y	$h(x)$	$h(x) - y$	$(h(x) - y)^2$
1	1	0	-1	1
2(	2	0	-2	4
3	3	0	-3	9

$\theta_0 = \theta_1 = 0$ then $h(x) = 0$

$$j(\theta_0 , \theta_1) = \frac{1}{2(3)}(1 + 4 + 9) = \frac{14}{6} = 2.33333$$

θ_0 , θ_1 are small values, but the error value is large. If θ_1 value increases, what happens to cost function ? as in the following example.

Example (3):

Let $\theta_0 = 0$, $\theta_1 = 0.5$, $h(x) = 0.5\, x$

x	y	$h(x)$	$h(x) - y$	$(h(x) - y)^2$
1	1	0.5	- 0.5	0.25
2	2	1	-1	1
3	3	1.5	-1.5	2.25

$$j(\theta_0 , \theta_1) = \frac{1}{2(3)}(0.25 + 1 + 2.25) = 0.5833$$

Example (4):

Let $\theta_0 = 0$, $\theta_1 = 1$, $h(x) = x$

x	y	$h(x)$	$h(x) - y$	$(h(x) - y)^2$
1	1	0	0	0
2	2	0	0	0
3	3	0	0	0

$j(\theta_0 , \theta_1) = 0$

Note that when $\theta_1 = 1$ then the value cost function = 0, this mean that the true value applies to the assumed value, and this means the optimal solution, see figure(4)

But if the theta value increases, the error value also increases for example.

Example (5):

$\theta_0 = 0$, $\theta_1 = 1.5$, $h(x) = 1.5\, x$

x	y	$h(x)$	$h(x) - y$	$(h(x) - y)^2$
1	1	1.5	0.5	0.25
2	2	3	1	1
3	3	4.5	1.5	2.25

$$j(\theta_0 , \theta_1) = \frac{1}{2(3)}(\ 0.25 + 1 + 2.25) = 0.58333$$

Fig (4)

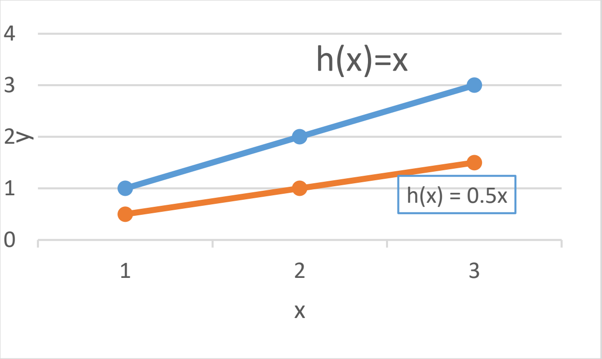
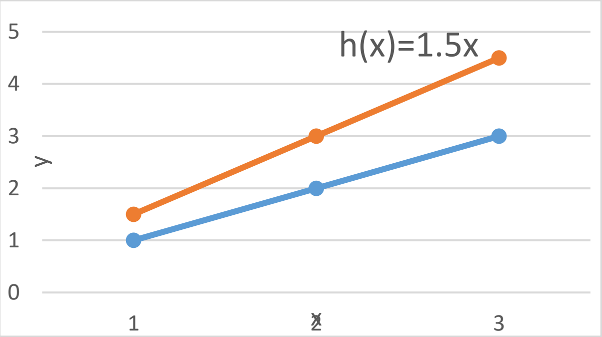


Fig (5)



In example(2) a small θ_0,θ_1 values were chosen after calculating cost function, their value would be large, if θ_1 increased(as in example 3)the value of error function became less than the previous value, if θ_1 value was updated again and it's value increased(as example 4),the valueof cost function=0,meaning that true value applies to expected value(optimal solution),

In example (5), large θ_1 value were chosen,the value of cost function is also large, so θ_1values are minimized to get the lowest value of cost function(objective function), figure(5).

8.5-The Gradient Descent:.

To find a mechanism to reduce the cost function, percentage we use the gradient desent method, which is it iterative process to determine the best parameters in order to lower the cost function while modeling the relationship between a dependent variable and one or more independent variables[6].

To implement the gradient descent algorithm, we need to reduce the cost function, the number of repetitions, the learing rate to determine the step size in each repetition while moving towards the minimum, the partial derivatives of theta to update the parameters in each repetition and the prediction function[7].

After each repetition, the cost is upgraded in proprtion to the error, the error value changes very quickly because it have a square in the error function, so we derive the error function and show it mathematically .

$$h_\theta(x_i) = \theta_0 + \theta_1\, x_i \qquad (5)$$

$$j(\theta_0\,,\theta_1) = \frac{1}{2\,m}\sum_{i=1}^{m}(h_\theta(x_i) - y_i)^2 \qquad (6)$$

$$= \frac{1}{2\,m}\sum_{i=1}^{m}(\theta_0 + \theta_1 x_i - y_i)^2$$

$$\frac{\partial \mathrm{j}}{\partial \theta_\mathrm{i}} = \frac{\partial}{\partial \theta_\mathrm{i}}\left(\frac{1}{2\,\mathrm{m}}\ \sum_{\mathrm{i}=1}^{\mathrm{m}}(\theta_0\,,\theta_1\,\mathrm{x_i}) - \mathrm{y_i})^2\right)$$

$$= \frac{1}{2\,m}\ \sum_{i=1}^{m} 2 * (\theta_0 + \theta_1 x_i) - y_i) * x_i^{\theta}$$

$$= \frac{1}{m}\sum_{i=1}^{m}(h_\theta(x_i) - y_i) * x_i^{\theta}$$

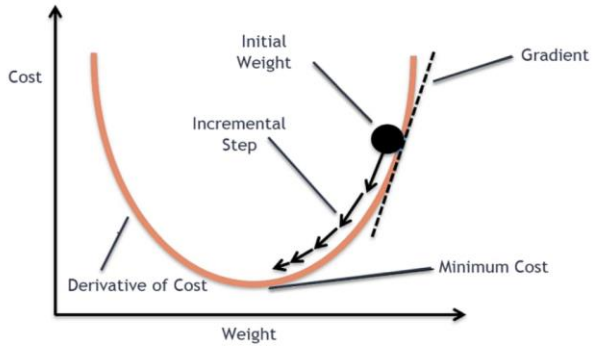
Where $x_i^{\theta} = \begin{cases} 1 & i = 0 \\ x_i & o.w \end{cases}$

$$\theta_i\ = \theta_i - \alpha * \left(\frac{\partial}{\partial \theta}\ cost\ (\theta_i\ )\right)$$

$$= \theta_i - \alpha\ \frac{1}{m}(\sum_{i=1}^{m}(h_\theta(x_i) - y_i)) * x_i^{\theta} \qquad (7)$$

This update is called the learning rate, which determines the speed at which the model reaches the convergence point, the point at which the error is ideal (as little as possible).

Fig (6) (Gradient Descent)



The alpha value that determines the step length, as shown in the the larger the alpha value ,the less accurate the result, and the smaller the alpha value, the more accurate the result figure(6). The above procedures can be used with Python's Example)1(to optimize theta0, theta1 values. The results were as follows after selecting the learning rate (α=0.01) and the number of iterations (100), see figure(7),figure(8).

Estimated theta1: 1.2783066626429578, Estimated theta0: 5.035844310464995

Fig (7) (Best fit line)

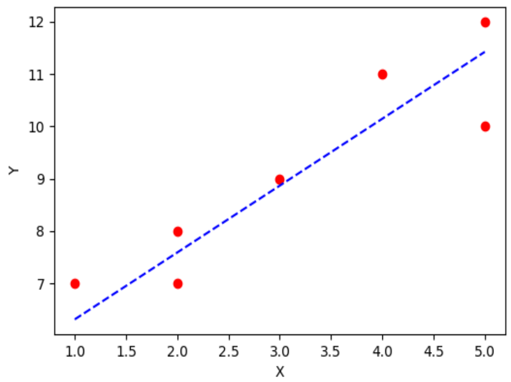
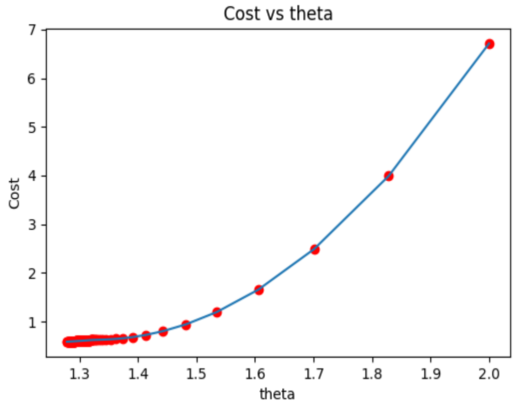


Fig (8) (Relation between cost and theta)



Multiple Linear Regression:.

We talked earlier a bout the expectation of one variable (input x ana output y) now we are dealing with more than one variable in the sense that the internal data has more than one information, instead of entering the area of the house to find out its price, we enter the area of the house, its location, the number of rooms, its age and others to determine its price. these inputs are called features. Each independent variable in this situation has an effect on the expected result [10].

For each entry, the corresponding Theta that informs the model will be determined based on the data points that best indicate the importance of each entry. The formula is as follows.

$$h_\theta(x) = \theta_0 x_0 + \theta_1 x_1 + \quad . \quad . \quad . \quad + \theta_n x_n \qquad (8)$$

Where $x_0 = 1$, x , θ are vectors

$$x = \begin{bmatrix} x_0 \\ \vdots \\ x_n \end{bmatrix} \in R^{n+1} \qquad \theta = \begin{bmatrix} \theta_0 \\ \vdots \\ \theta_n \end{bmatrix} \in R^{n+1}$$

The above equation can be written as follows .

$$h_\theta(x_i) = \theta_j^T x_i \qquad (9)$$

$$= [\theta_0 \quad ... \quad \theta_n] \begin{bmatrix} x_0 \\ \vdots \\ x_1 \end{bmatrix}$$

where $i = 1, \dots, m$, $j = 0, \dots, n$, θ_i^T is transpose(is obtained by interchanging the row and column)[5].

Cost function (loos): $j(\theta_j) = \frac{1}{2\,m}\sum_{i=1}^{m}(h_\theta(x_i) - y_i)^2$ (6)

Repeat until convergence : $\theta_j = \theta_j - \alpha\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x_i) - y_i) * x_i$ (7)

Example (6):

Assume that we have number of goods and each commodity it has three features as shown in the table.

Solution:

Let $\theta_0 = 5 , \theta_1 = 2 , \theta_2 = 3 , \theta_3 = 4$

Where: $m = 5 , n = 4$

x_0	x_1	x_2	x_3	y	$h(x) - y$
1	3	20	13	120	123-120=3
1	6	16	11	113	109-113=-4
1	5	12	8	85	83-85=-2
1	7	17	9	100	106-100=6
1	4	10	7	60	71-60=11

We $n + 1$ features because that are $x_0 = 0$

$$x_1 = \begin{bmatrix} 1 \\ 3 \\ 20 \\ 13 \end{bmatrix} ,x_2 = \begin{bmatrix} 1 \\ 6 \\ 16 \\ 11 \end{bmatrix} , x_3 = \begin{bmatrix} 1 \\ 5 \\ 12 \\ 8 \end{bmatrix} , x_4 = \begin{bmatrix} 1 \\ 7 \\ 17 \\ 9 \end{bmatrix} , x_5 = \begin{bmatrix} 1 \\ 4 \\ 10 \\ 7 \end{bmatrix}$$

$h(x_i) = \theta_j^T * x_i$

$h(x_1) = [5 \quad 2 \quad 3 \quad 4] \begin{bmatrix} 1 \\ 3 \\ 20 \\ 13 \end{bmatrix}$ =$[5 + 6 + 60 + 52] = 123$

$h(x_2) = 109 , h(x_3) = 83 , h(x_4) = 106 , h(x_5) = 71$

repeat until convergence : suppose $\alpha = 0.01$, $m = 5$

$\theta_0 = 5 - \frac{0.01}{5}(14) * 1 = 4.972$, $\theta_1 = 2 - \frac{0.01}{5}(14) * 3 = 1.916$

$\theta_2 = 3 - \frac{0.01}{5}(14) * 20 = 2.44$, $\theta_3 = 4 - \frac{0.01}{5}(14) * 13 = 3.636$

the theta values began decrease, we repert the process several times until we reach the optimal value, can use python(2) .

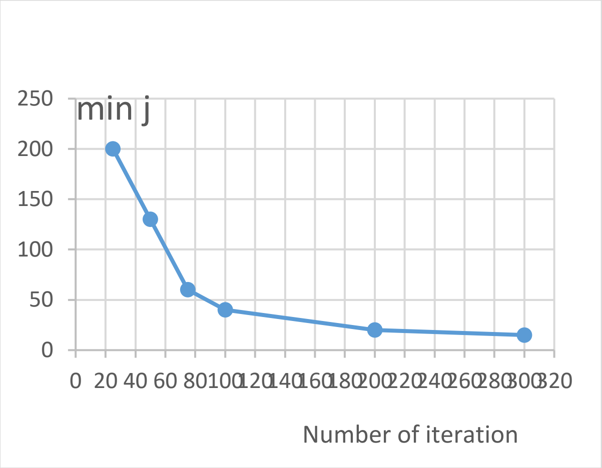
Now, What is the right number for the number of attempts ?

The following figure (9), shows the relationship between the number of attempts (iterations) and the reduction of the cost function.

It's clear from the drwing that the more the number of attempts, the lower cost function, but after a certain period the tendency approaches zero, so the number of attempts becomes large with a slight change in the value of the cost function[13].

So, we must stop because it is considered a wast of time and effort.

Fig (9) (Relation between min j and numberof iteration)



Can stop after (10) or more attempts, so each case is different from the others.

Also, the choice of the alpha value affects the accuracy of the data the number of attempts, so the speed and the number of attempts shoud be appropriate to find the optimal solution.

Normal equation:.

It is an analytical approach used for optimization an alternative to linear regression, the equation is reduced without the need for repetition, this approach is an effective and time-saving option[6].

The normal equation method is based on the mathematical concept of small and large values, where the partial derivative of any function is zero at the minimum and maximum.

It can be said that they are equations obtained by determining the partial derivatives of the sum of squared errors or the cost function equal to zero, where a person can estimate the coefficients of multiple linear regression[7].

$$h(\theta) = \theta_0 x_0 + \theta_1 x_1 + . . . + \theta_n x_n \quad . \; . \; . \; (8)$$

$$h(\theta) = \theta^T x \qquad (9)$$

$$j(\theta) = \frac{1}{2\,m} \sum_{i=1}^{m}(h_\theta(x_i) - y_i)^2$$

where m: features , $i = 1, \ldots, m$, x_i: input value , y_i: expected value, represent the cost function as vector, $x_0 = 1$

$$\begin{bmatrix} h_\theta(x_0) \\ h_\theta(x_1) \\ \vdots \\ h_\theta(x_m) \end{bmatrix} - \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_m \end{bmatrix}$$

$\frac{1}{2\,m}$ was ignored because it doesn't make any difference in the work. it was used mathematically during the calculation of gradient descent.

$$\begin{bmatrix} \theta^T(x_0) \\ \theta^T(x_1) \\ \vdots \\ \theta^T(x_m) \end{bmatrix} - y$$

$$\Rightarrow \begin{bmatrix} \theta_0(x_{00}) + \theta_1(x_{01}) + \ldots + \theta_n(x_{0n}) \\ \theta_0(x_{10}) + \theta_1(x_{11}) + \ldots + \theta_n(x_{1n}) \\ \vdots \\ \theta_0(x_{m0}) + \theta_1(x_{m1}) + \ldots + \theta_n(x_{mn}) \end{bmatrix} - y$$

It can be written as $[X\theta - y]$ since the formula a bove can not squared since the square of the vector (matrix) is not equal to the square of each of it's values to obtain the squared value the vector can be multiplied by transpose.

Therefore the cost function is. $(X\theta - y)^T(X\theta - y)$

Cost function : $j(\theta) = (X\theta - y)^T(X\theta - y)$

$$\frac{\partial j(\theta)}{\partial \theta} = \frac{\partial}{\partial x}[(X\theta - y)^T(X\theta - y)]$$

$= 2\,X^T X\theta - 2X^T\,y$, cost'$(\theta) =$ 0

$$0 \quad = 2\,X^T X\theta - 2X^T\,y$$

$$2X^T X\theta = 2X^T\,y$$

$$(X^T X)^{-1}\,(X^T X)\theta = \,(X^T X)^{-1}\,(X^T y)$$

$\theta = (X^T X)^{-1}(X^T y)$, where $(X^T X)^{-1} \neq (X^T X)$

$(X^T X)^{-1}.(X^T X) = I(I$ identity matrix)

This is the normal equation with θ giving the minimum cost value[6].

Example (7):

Let's the following data.

x_1	x_2	x_3	y
3	20	13	120
6	16	11	113
5	12	8	85
7	17	9	100
4	10	7	60

$$X = \begin{bmatrix} 1 & 3 & 20 & 13 \\ 1 & 6 & 16 & 11 \\ 1 & 5 & 12 & 8 \\ 1 & 7 & 17 & 9 \\ 1 & 4 & 10 & 7 \end{bmatrix} \quad , \quad X^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 3 & 6 & 5 & 7 & 4 \\ 20 & 16 & 12 & 17 & 10 \\ 13 & 11 & 8 & 9 & 7 \end{bmatrix}$$

$$X^T X = \begin{bmatrix} 5 & 25 & 75 & 48 \\ 25 & 135 & 375 & 236 \\ 75 & 375 & 1189 & 755 \\ 48 & 236 & 755 & 484 \end{bmatrix} \quad ,$$

$$(X^T X)^{-1} = \begin{bmatrix} \frac{85841}{7870} & \frac{-8283}{7870} & \frac{410}{787} & \frac{-1087}{787} \\ \frac{-8283}{7870} & \frac{1299}{7870} & \frac{-70}{787} & \frac{128}{787} \\ \frac{410}{787} & \frac{-70}{787} & \frac{108}{787} & \frac{-175}{787} \\ \frac{-1087}{787} & \frac{128}{787} & \frac{-175}{787} & \frac{320}{787} \end{bmatrix}$$

$$(X^T X)^{-1}.X^T = \begin{bmatrix} \frac{841}{3935} & \frac{-17827}{7870} & \frac{3333}{3935} & \frac{-27}{787} & \frac{17619}{7870} \\ \frac{-873}{3935} & \frac{2391}{7870} & \frac{26}{3935} & \frac{43}{787} & \frac{-1127}{7870} \\ \frac{85}{787} & \frac{-207}{787} & \frac{-44}{787} & \frac{181}{787} & \frac{-15}{787} \\ \frac{-43}{787} & \frac{401}{787} & \frac{13}{787} & \frac{-286}{787} & \frac{-85}{787} \end{bmatrix}$$

$$(X^T X)^{-1}.X^T.y = \begin{bmatrix} -27.42833545 \\ 5.208907581 \\ 0.3418043202 \\ 9.603557814 \end{bmatrix} = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}$$

Can be use python.

9-The difference between gradient descent and normal equation:

Gradient descent

1\\Be slow and need to choose the Learning rate.

2\\It's an iterative algorithm.

3\\Works well with a large number of features.

4\\The scaling feature can be used

Normal equation

1\\Be fast and do not need to choose the Learning rate.

2\\It's an analytical approach.

3\\Works well with a small number of features.

4\\Does not use the scaling feature.

Sometimes a problem occurs when applying the normal equation because the matrix is not invertible or it is singular, it has no inverse ,or the number of rows is less than the number of columns, so we need to increase the number of rows or reduce the number of columns, or there are data, one of which is dependent on the other m this leads to the determinant of the matrix =0 (it has no inverse)[6].

10-Conclusion:

1-Neural networks rely on training data in order to learn and improve their accuracy over time, and this is done once the learning algorithms are accurately adjusted.

2- Reducing the cost function is the driving force behind linear regression.

3-In order for the regression to reach the local minimum, the learning rate must be set to an approprite value (neither too small nor too large).

4-Partial derivation the regression coefficients can reduce cost function.

5-Normal equation can be applied to get the best update of regression coefficients in fewer steps.

Reference

[1] Picton, P. (1994). What is a neural network?. In Introduction to Neural Networks (pp. 1-12). Palgrave, London.

[2] Wasserman, P. D., & Schwartz, T. (1988). Neural networks. II. What are they and why is everybody so interested in them now?. IEEE expert, 3(1), 10-15.

[3] VOHRADSKY, J. (2001). Neural network model of gene expression. the FASEB journal, 15(3), 846-854.

[4] Shanmuganathan, S. (2016). Artificial neural network modelling: An introduction. In Artificial neural network modelling (pp. 1-14). Springer, Cham.

[5] Yang, X. (2008). Introduction to mathematical optimization. From linear programming to metaheuristics.

[6] Lubis, F. F., Rosmansyah, Y., & Supangkat, S. H. (2014, September). Gradient descent and normal equations on cost function minimization for online predictive using linear regression with multiple variables. In 2014 International Conference on ICT For Smart Society (ICISS) (pp. 202-205). IEEE.

[7] Freeman, J. A., & Skapura, D. M. (1991). Neural networks: algorithms, applications, and programming techniques. Addison Wesley Longman Publishing Co., Inc..

[8] Higham, C. F., & Higham, D. J. (2019). Deep learning: An introduction for applied mathematicians. Siam review, 61(4), 860-891.

[9] Boyce, W. E., DiPrima, R. C., & Meade, D. B. (2021). Elementary differential equations and boundary value problems. John Wiley & Sons.

[10] Dawani, J. (2020). Hands-On Mathematics for Deep Learning: Build a solid mathematical foundation for training efficient deep neural networks. Packt Publishing Ltd.

[11] Lanczos, C. (1952). Solution of Systems of Linear Equations by. Journal of research of the National Bureau of Standards, 49(1), 33.

[12] Aris, R. (1994). Mathematical modelling techniques. Courier Corporation.

[13] Kelleher, J. D. (2019). Deep learning. MIT press.