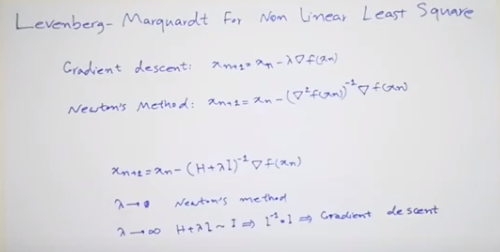
Levenberg-Marquardt algorithm is an optimization algorithm which is a combination of gradient descent method and Gauss Newton method. This algorithm finds the minimum of linear or non - linear function having an array of parameters.

The gradient descent method is useful when the initial point is far from minimum point of the function. The Gauss Newton algorithm is useful when initial point is close to minimum point of function. The combined method of the two gives the ability to algorithm to become independent of initial point choosing and hence Levenberg-Marquardt algorithm is derived.

The example to use Levenberg-Marquardt algorithm for non-linear least square problem can be shown as



Suppose F(x) is the multi variable non linear function.

H is hessian matrix and I is identity matrix , lambda is given at starting of the problem.

N is the iteration 0, 1, 2,3, …..

Starting value is 0 for all the variables of the function F(x) at n = 0 iteration.

The next new value is present at iteration Xn+1 for all the variables present in function F(x).

 is partial derivative of F(x) at n th iteration.

When lambda is = 0 then only hessian matrix remains thus the Levenberg-Marquardt algorithm converts to Newton’s method and when lambda is equal to infinity the algorithm then converts to Gradient descent. Thus in this way both methods are combined.

 is tolerance of functional value and is given at the starting of the problem which is following this algorithmic approach for solution.

If mod of ( f(xn+1 ) ) <= 

Then Xn+1 isthe final minimum point of the function F(x) found at n+1 th iteration.

Then stop the iteration.

Else if f(xn+1 ) < f(xn ) then

lambda n+1 = lambdan / 2

And continue the iteration on  to find new Xn+1

Else if f(xn+1 ) > f(xn ) then

lambda n+1 = 2 \* lambdan

And continue the iteration on  to find new Xn+1