# The Levenberg-Marquardt Method:

# Nonlinear Least Squares Minimization and Curve Fitting

November, 2010 Guanqing Ou and Arash Ushani

## Introduction

The solution to the Nonlinear Least Squares Minimization problem is the minimum of a function that follows the form

|  |  |
| --- | --- |
|  | (1) |

where *x* is a vector , and are functions that map from to . The s are referred to as residuals and . The Levenberg-Marquardt algorithm provides a method of solving this problem, which can be used to fit a set of data points to a function with nonlinear parameters. The following paper discusses this problem and methods of solving it, including the gradient descent method, the Gauss-Newton method, and the Levenberg-Marquardt method. These three methods are compared using an example application, and a brief discussion of other applications of the LM method is included.

## nonlinear least squares minimization

All three methods of solving this problem make use of the gradient and Hessian of the function . This makes sense, as these terms?????? give us information about the behavior of the function as changes, and thus can be used to find the values of for which is minimized. The following discusses how the gradient and Hessian may be expressed in a way that facilitates numerically finding this minimum.

Let us consider equation (1). If we think of all s as a vector of functions: , then the function can be rewritten as

|  |  |
| --- | --- |
|  | (2) |

This allows us to express the gradient and Hessian of in a fairly concise manner. Recall that for any function , where , the gradient , and the Hessian is a matrix with dimensions where the entry .

First let us consider. In the most general form,. But *,* so we can express each as

|  |  |
| --- | --- |
|  | (3) |

Using this equation, we can restate , as

|  |  |
| --- | --- |
|  | (4) |

We can further simplify this by remembering that the Jacobian of a set of functions is

|  |  |
| --- | --- |
|  | (5) |

This is the transpose of the left matrix in equation 4. The right matrix, , is simply , as . Thus,

|  |  |
| --- | --- |
|  | (6) |

The Hessian can be found similarly. Recall each term . We already know that , to find , we simply take the partial derivative to :

|  |  |
| --- | --- |
|  | (7) |

which, using the Chain Rule, can be expressed as

|  |  |
| --- | --- |
|  | (8) |

If we assume that all is linear in or simply that the residuals are small, the second term can be ignored. And the first term is simply the terms of the Jacobian of multiplied by each other, so we can say

|  |  |
| --- | --- |
|  | (8) |

With these expressions of the gradient and Hessian of , we can move onto a discussion of the methods used to solve the nonlinear least squares minimization problem.

### Computational Approach

Parameter and Curve Fitting

In each of the following methods, there is assumed to exist a function where is an independent variable, and **p** is a vector of n parameters that when applied to the function is presumed to provide a fit of the function to a set of m points . The goodness-of-fit between this parameterized function and the set of data points is determined by calculating the chi-squared error

|  |  |
| --- | --- |
|  | (9) |

weights the errors, in case the user wants to optimize the function at certain points . A weighting matrix is defined with the diagonals equal to .

The goal of any parameter/curve fitting method is to iteratively find a perturbation **h** to the vector **p** until some is the minimum possible value for the points and function **.**

Equation 9 can be rewritten in terms of matrices:

|  |  |
| --- | --- |
|  | (10) |

which, after factoring, gives

|  |  |
| --- | --- |
|  | (11) |

The Gradient Descent Method

Gradient descent is a fairly simple method, which updates the parameter vector **p** by subtracting the scaled gradient at each step, such that

|  |  |
| --- | --- |
|  | (12) |

The gradient of can be found using equation 12:

|  |  |
| --- | --- |
|  | (13) |

So the perturbation **h** is simply

|  |  |
| --- | --- |
|  | (14) |

The Gauss-Newton Method

The Gauss-Newton method assumes that the function is quadratic near the optimal parameter values, and locally approximates the parameters using a first-order Taylor series expansion:

|  |  |
| --- | --- |
|  | (15) |

This allows us to restate equation 11 as follows:

|  |  |
| --- | --- |
|  | (16) |

Since we are assuming that the behavior of the parameter vector is quadratic around the optimal value, to find the perturbation **h** that minimizes , we find where :

|  |  |
| --- | --- |
|  | (17) |

Rearrangement gives us:

|  |  |
| --- | --- |
|  | (18) |

The Levenberg-Marquardt Method

### IMplementation

The above methods were implemented in MATLAB as shown below.

### Evaluation and comparison

*Stability*

For the finite difference method, we expect the stability to be determined by the eigenvalues of the matrices . For collocation and Galerkin, the matrix that determines stability is