Lecture 13: Non-linear least squares and the Gauss-Newton method

Michael S. Floater

November 12, 2018

1 Non-linear least squares

A minimization problem that occurs frequently is the minimization of a function of the form

$$f(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{m} r_i(\mathbf{x})^2, \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$ and $r_i : \mathbb{R}^n \to \mathbb{R}$, $i = 1, \ldots, m$. Such a minimization problem comes from curve fitting by least squares, where the r_i are the residuals.

1.1 Linear case

Suppose we are given data (t_j, y_j), $j = 1, \ldots, m$, and we want to fit a straight line,

$$p(t) = x_1 + x_2 t.$$

Then we would like to find x_1 and x_2 that minimize

$$\frac{1}{2}\sum_{i=1}^{m}(y_i - p(t_i))^2 = \frac{1}{2}\sum_{i=1}^{m}(y_i - x_1 - x_2 t_i)^2.$$

This problem can be formulated as (1) with $n = 2$ where the residuals are

$$r_i(\mathbf{x}) = r_i(x_1, x_2) = y_i - p(t_i) = y_i - x_1 - x_2 t_i.$$

More generally, we could fit a polynomial

$$p(t) = \sum_{j=1}^{n} x_j t^{j-1},$$

or even a linear combination of basis functions $\phi_1(t), \ldots, \phi_n(t)$,

$$p(t) = \sum_{j=1}^{n} x_j \phi_j(t).$$

These are again examples of (1), where

$$r_i(\mathbf{x}) = r_i(x_1, \ldots, x_n) = y_i - p(t_i).$$

In all these cases, the problem is linear in the sense that the solution is found by solving a linear system of equations. This is because f is quadratic in $\mathbf{x}$. We can express f as

$$f(\mathbf{x}) = \frac{1}{2}\|A\mathbf{x} - \mathbf{b}\|^2,$$

where $A \in \mathbb{R}^{m,n}$ is the Vandermonde matrix

$$A = \begin{bmatrix} \phi_1(t_1) & \phi_2(t_1) & \cdots & \phi_n(t_1) \\ \phi_1(t_2) & \phi_2(t_2) & \cdots & \phi_n(t_2) \\ \vdots & \vdots & & \vdots \\ \phi_1(t_m) & \phi_2(t_m) & \cdots & \phi_n(t_m) \end{bmatrix},$$

$\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$ is the vector of coefficients of p, and $\mathbf{b} = [y_1, y_2, \ldots, y_m]^T$ is the vector of data observations.

We have seen that we can then find $\mathbf{x}$ from the QR decomposition of A or from the normal equations, for example.

1.2 Non-linear case

It might be more appropriate to fit a curve $p(t)$ that does not depend linearly on its parameters $x_1, \ldots, x_n$. An example of this is the rational function

$$p(t) = \frac{x_1 t}{x_2 + t}.$$

Another is the exponential function

$$p(t) = x_1 e^{x_2 t}.$$

In both cases we would again like to find x_1 and x_2 to minimize

$$\frac{1}{2}\sum_{i=1}^{m}(y_i - p(t_i))^2.$$

As for the linear case we can reformulate this as the minimization of f in (1) with the residuals

$$r_i(\mathbf{x}) = r_i(x_1, x_2) = y_i - p(t_i).$$

In these cases the problem is non-linear since f is no longer a quadratic function (the residuals are no longer linear in the parameters $x_1, \ldots, x_n$). One approach to minimizing such an f is to try Newton's method. Recall that Newton's method for minimizing f is simply Newton's method for solving the system of n equations, $\nabla f(\mathbf{x}) = \mathbf{0}$, which is the iteration

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - (\nabla^2 f(\mathbf{x}^{(k)}))^{-1}\nabla f(\mathbf{x}^{(k)}). \tag{2}$$

The advantages of Newton's method are:

1. If f is quadratic, it converges in one step, i.e., $\mathbf{x}^{(1)}$ is the global minimum of f for any initial guess $\mathbf{x}^{(0)}$.

2. For non-linear least squares it converges quadratically to a local minimum if the initial guess $\mathbf{x}^{(0)}$ is close enough.

The disadvantage of Newton's method is its lack of robustness. For non-linear least squares it might not converge. One reason for this is that the search direction

$$\mathbf{d}^{(k)} = -(\nabla^2 f(\mathbf{x}^{(k)}))^{-1}\nabla f(\mathbf{x}^{(k)})$$

might not even be a descent direction: there is no guarantee that it fulfills the descent condition,

$$\nabla f(\mathbf{x}^{(k)}))^T \mathbf{d}^{(k)} < 0.$$

One way to improve robustness is to use the Gauss-Newton method instead. The Gauss-Newton method is also simpler to implement.

2 Gauss-Newton method

The Gauss-Newton method is a simplification or approximation of the Newton method that applies to functions f of the form (1). Differentiating (1) with respect to x_j gives

$$\frac{\partial f}{\partial x_j} = \sum_{i=1}^{m} \frac{\partial r_i}{\partial x_j} r_i,$$

and so the gradient of f is

$$\nabla f = J_r^T \mathbf{r},$$

where $\mathbf{r} = [r_1, \ldots, r_m]^T$ and $J_r \in \mathbb{R}^{m,n}$ is the Jacobian of $\mathbf{r}$,

$$J_r = \left[\frac{\partial r_i}{\partial x_j} \right]_{i=1,\ldots,m, j=1,\ldots,n}.$$

Differentiating again, with respect to x_k, gives

$$\frac{\partial^2 f}{\partial x_j \partial x_k} = \sum_{i=1}^{m} \left(\frac{\partial r_i}{\partial x_j} \frac{\partial r_i}{\partial x_k} + r_i \frac{\partial^2 r_i}{\partial x_j \partial x_k} \right),$$

and so the Hessian of f is

$$\nabla^2 f = J_r^T J_r + Q,$$

where

$$Q = \sum_{i=1}^{m} r_i \nabla^2 r_i.$$

The Gauss-Newton method is the result of neglecting the term Q, i.e., making the approximation

$$\nabla^2 f \approx J_r^T J_r. \tag{3}$$

Thus the Gauss-Newton iteration is

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - (J_r(\mathbf{x}^{(k)})^T J_r(\mathbf{x}^{(k)}))^{-1} J_r(\mathbf{x}^{(k)})^T \mathbf{r}(\mathbf{x}^{(k)}).$$

In general the Gauss-Newton method will not converge quadratically but if the elements of Q are small as we approach a minimum, we can expect fast convergence. This will be the case if either the r_i or their second order partial derivatives

$$\frac{\partial^2 r_i}{\partial x_j \partial x_k}$$

are small as we approach a minimum.

An advantage of this method is that it does not require computing the second order partial derivatives of the functions r_i. Another is that the search direction, i.e.,

$$\mathbf{d}^{(k)} = -(J_r(\mathbf{x}^{(k)})^T J_r(\mathbf{x}^{(k)}))^{-1} \nabla f(\mathbf{x}^{(k)}),$$

is always a descent direction (as long as $J_r(\mathbf{x}^{(k)})$ has full rank). This is because $J_r^T J_r$ is positive semi-definite, which implies that $(J_r^T J_r)^{-1}$ is also positive semi-definite, which means that

$$\nabla f(\mathbf{x}^{(k)}))^T \mathbf{d}^{(k)} = -\nabla f(\mathbf{x}^{(k)}))^T (J_r(\mathbf{x}^{(k)})^T J_r(\mathbf{x}^{(k)}))^{-1} \nabla f(\mathbf{x}^{(k)}) \leq 0.$$

If $J_r(\mathbf{x}^{(k)})$ has full rank this inequality is strict. This suggests that the Gauss-Newton method will typically be more robust than Newton's method.

There is still no guarantee, however, that the Gauss-Newton method will converge in general. In pratice, one would want to incorporate a step length $\alpha^{(k)}$ into the iteration:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha^{(k)} \mathbf{d}^{(k)},$$

using some rule like the Armijo rule, in order to ensure descent at each iteration.

3 Example

In a biology experiment studying the relation between substrate concentration [S] and reaction rate in an enzyme-mediated reaction, the data in the following table were obtained.

i	1	2	3	4	5	6	7
[S]	0.038	0.194	0.425	0.626	1.253	2.500	3.740
rate	0.050	0.127	0.094	0.2122	0.2729	0.2665	0.3317

It is desired to find a curve (model function) of the form

$$\text{rate} = \frac{V_{\max}[S]}{K_M + [S]}$$

that best fits the data in the least-squares sense, with the parameters $V_{\max}$ and K_M to be determined.

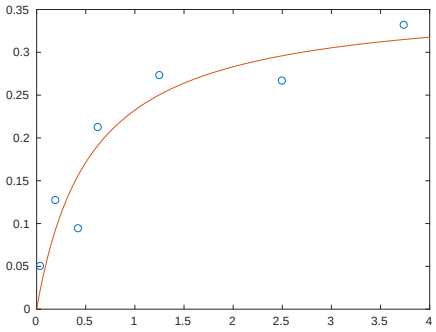

Figure 1: Curve model

We can rewrite this problem as finding x_1 and x_2 such that

$$p(t) = \frac{x_1 t}{x_2 + t}$$

best fits the data (t_i, y_i), $i = 1, 2, \ldots, 7$, of the table where t_i is the i-th concentration [S] and y_i is the i-th rate. We will find x_1 and x_2 that minimize the sum of squares of the residuals

$$r_i = y_i - \frac{x_1 t_i}{x_2 + t_i}, \quad i = 1, \ldots, 7.$$

The Jacobian J_r of the vector of residuals r_i with respect to the unknowns x_1 and x_2 is a 7×2 matrix with the i-th row having the entries

$$\frac{\partial r_i}{\partial x_1} = -\frac{t_i}{x_2 + t}, \qquad \frac{\partial r_i}{\partial x_2} = \frac{x_1 t_i}{(x_2 + t)^2}.$$

Starting with the initial estimates of $x_1 = 0.9$ and $x_2 = 0.2$, and using the stopping criterion

$$\|\nabla f\|_2 \leq 10^{-15}, \tag{4}$$

the method converges in 14 iterations, yielding the solution $x_1 = 0.3618$, $x_2 = 0.5563$. The sum of squares of residuals decreased from the initial value of 1.445 to 0.0078. The plot in Figure 1 shows the curve determined by the model for the optimal parameters with the observed data.

We can alternatively try the (full) Newton method. We then also need the second order partial derivatives,

$$\frac{\partial^2 r_i}{\partial x_1^2} = 0, \qquad \frac{\partial^2 r_i}{\partial x_1 \partial x_2} = \frac{t_i}{(x_2 + t_i)^2}, \qquad \frac{\partial^2 r_i}{\partial x_2^2} = \frac{-2 x_1 t_i}{(x_2 + t_i)^3}.$$

Starting with the same initial estimates of $x_1 = 0.9$ and $x_2 = 0.2$, Newton's method does not converge. However, if we change the initial estimates to $x_1 = 0.4$ and $x_2 = 0.6$ we find that both the Gauss-Newton and Newton methods converge. Moreover, using again the stopping criterion of (4), the Gauss-Newton method needs 11 iterations while Newton's method needs only 5.