

# STAT-685 Directed Studies: The Computational Fulfilment of MLE

Weimin Zhang  
Aug. 1, 2001

## Abstract

In this note, we introduce the methods for the computational fulfilment of MLE. The methods include Newton-Raphson, Fisher scoring, Fletche-Reeves, Polak-Ribiere, Davidon-Fletcher-Powell and Broyden-Fletcher-Goldfarb-Shanno algorithms.

## 1. Introduction:

A probability function is a means to estimate data on the basis of given parameters. Using the same formula for the probability function we can reverse the emphasis to the estimation of parameters on the basis of given data. Essentially, given a dataset, we determine the most likely parameters that give rise to the given data within the scope of the probability function. This is called the likelihood function. Whereas the joint iid probability is given as:

$$f(\mathbf{x}; \boldsymbol{\theta}) = \prod_{i=1}^n f(x_i; \theta) \quad (1)$$

The likelihood is understood as:

$$L(\boldsymbol{\theta}; \mathbf{x}) = \prod_{i=1}^n f(\theta; x_i) \quad (2)$$

Moreover, since the overall likelihood of the function is multiplicative, we log-transform the likelihood function,  $L(\boldsymbol{\theta}; \mathbf{x})$ , to make it additive. This is known as the log-likelihood,  $\mathcal{L}(\boldsymbol{\theta}; \mathbf{x})$ , and it's the foundation for all maximum likelihood theory.

For each sample point  $\mathbf{x}$ , let  $\hat{\boldsymbol{\theta}}(\mathbf{x})$  be a parameter value at which  $\mathcal{L}(\boldsymbol{\theta}; \mathbf{x})$  attains its maximum likelihood estimator (MLE) of the parameter  $\boldsymbol{\theta}$  based on a sample  $\mathbf{X}$  is  $\hat{\boldsymbol{\theta}}(\mathbf{X})$ . We also use the abbreviation MLE to stand for maximum likelihood estimate, when we are talking of the realized value of the estimator.

To find estimates, we can use Newton's method. This method is a linear Taylor series approximation where we write the derivative of the log-likelihood in a Taylor series expansion. In the following, we use  $\mathcal{L}' = \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}}$  and  $\mathcal{L}'' = \frac{\partial^2 \mathcal{L}}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^T}$ . We wish to solve:

$$\mathcal{L}' = 0 \quad (3)$$

Solving this equation provides estimates of  $\theta$ . Thus, this is called the estimating equations. We may expand this as a Taylor series as:

$$0 = \mathcal{L}'(\hat{\theta}^{(0)}) + (\theta - \hat{\theta}^{(0)})\mathcal{L}''(\hat{\theta}^{(0)}) + \frac{(\theta - \hat{\theta}^{(0)})^2}{2!}\mathcal{L}'''(\hat{\theta}^{(0)}) + \dots \quad (4)$$

The first two terms reduce to the linear equation

$$0 \approx \mathcal{L}'(\hat{\theta}^{(0)}) + (\theta - \hat{\theta}^{(0)})\mathcal{L}''(\hat{\theta}^{(0)}) \quad (5)$$

Such that we may write (solving for  $\theta$ )

$$\theta \approx \hat{\theta}^{(0)} - \frac{\mathcal{L}'\hat{\theta}^{(0)}}{\mathcal{L}''(\hat{\theta}^{(0)})} \quad (6)$$

We may iterate this estimation using:

$$\begin{aligned} \hat{\theta}^{(r+1)} &= \hat{\theta}^{(r)} - \frac{\mathcal{L}'\hat{\theta}^{(r)}}{\mathcal{L}''(\hat{\theta}^{(r)})} \\ &= \hat{\theta}^{(r)} + (-H(\hat{\theta}^{(r)}))^{-1}g(\hat{\theta}^{(r)}) \end{aligned} \quad (7)$$

for  $r=1,2,\dots$  and a reasonable starting value  $\theta^{(0)}$ . Where **gradient**  $g$  is the derivative of log-likelihood function, and  $g(0)$  is often called the score function.  $H = \mathcal{L}''$ , is **Hessian** matrix.

This linearized Taylor series approximation is exact if the function is truly quadratic and only one iteration is needed.

**Example 1:** Give:

$$\mathcal{L}(\theta; \mathbf{x}) = 4 - \theta_1^2 - \theta_2^2$$

We have:

$$\begin{aligned} g &= (-2\theta_1, -2\theta_2) \\ H &= \begin{bmatrix} -2 & 0 \\ 0 & -2 \end{bmatrix} \end{aligned}$$

Now, let  $\theta_0 = (3, 5)$ , then,

$$\begin{aligned} \hat{\theta}^{(1)} &= \hat{\theta}^{(0)} + (-H(\hat{\theta}^{(0)}))^{-1}g(\hat{\theta}^{(0)}) \\ &= (3, 5) + (-6, -10)\left(-\begin{bmatrix} -2 & 0 \\ 0 & -2 \end{bmatrix}^{-1}\right) \\ &= (3, 5) + (-3, -5) \\ &= (0, 0) \end{aligned}$$

$$\begin{aligned}
\hat{\theta}^{(2)} &= \hat{\theta}^{(1)} + (-H(\hat{\theta}^{(1)})^{-1}g(\hat{\theta}^{(1)})) \\
&= (0,0) + (0,0)\left(-\begin{bmatrix} -2 & 0 \\ 0 & -2 \end{bmatrix}^{-1}\right) \\
&= (0,0)
\end{aligned}$$

We can rewrite (7) as

$$\hat{\theta}^{(r+1)} = \hat{\theta}^{(r)} + \delta g \Delta \tag{8}$$

With different  $\delta, \Delta$ , or  $g\Delta$  there are different algorithms to calculate MLE. In the following sections we present some estimation methodologies.

## 2. Newton-Raphson

To find maximum likelihood estimates, we proceed with the derivation originating with the log-likelihood and utilize the first derivatives. We now derive the (observed) matrix of second derivatives. This will then give us the necessary ingredients for the Newton-Raphson methodology to find the maximum likelihood estimates. The Newton-Raphson algorithm implements (7) without change. In order to use the Newton-Raphson algorithm, one needs an initial estimate of  $\theta$ ,  $\hat{\theta}^{(0)}$ . In fact, in some cases this initial estimate is critical as the algorithm will not always converge for a given  $\hat{\theta}^{(0)}$ .

If it is not clear that the algorithm will converge to the maximizer of the likelihood function then several different initial estimates can be tried.

**Example 2:** We consider i.i.d.  $N(\mu, \sigma^2)$  normal random variables  $X_1, \dots, X_n$  with density function:

$$f(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Given outcomes  $x_1, \dots, x_n$ , the log-likelihood function is

$$\mathcal{L}(\mu, \sigma^2|x) = -\frac{n}{2}\log 2\pi - \frac{n}{2}\log \sigma^2 - \frac{1}{2} \sum_{i=1}^n \frac{(x_i - \mu)^2}{\sigma^2}$$

The partial derivatives, with respect to  $\mu$  and  $\sigma^2$  are

$$\frac{\partial \mathcal{L}(\mu, \sigma^2|x)}{\partial \mu} = \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu)$$

$$\frac{\partial \mathcal{L}(\mu, \sigma^2|x)}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^n (x_i - \mu)^2$$

That is, gradient,  $g$

$$g = \left( \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu), -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^n (x_i - \mu)^2 \right)$$

The second-order partial derivatives are

$$\frac{\partial^2 \mathcal{L}(\mu, \sigma^2 | x)}{\partial \mu^2} = -\frac{n}{\sigma^2}$$

$$\frac{\partial^2 \mathcal{L}(\mu, \sigma^2 | x)}{\partial (\sigma^2)^2} = \frac{n}{2\sigma^4} - \frac{1}{\sigma^6} \sum_{i=1}^n (x_i - \mu)^2$$

$$\frac{\partial^2 \mathcal{L}(\mu, \sigma^2 | x)}{\partial \mu \partial \sigma} = -\frac{1}{\sigma^4} \sum_{i=1}^n (x_i - \mu)$$

That is, Hessian matrix,  $H$

$$H = \begin{bmatrix} -\frac{n}{\sigma^2} & -\frac{1}{\sigma^4} \sum_{i=1}^n (x_i - \mu) \\ -\frac{1}{\sigma^4} \sum_{i=1}^n (x_i - \mu) & \frac{n}{2\sigma^4} - \frac{1}{\sigma^6} \sum_{i=1}^n (x_i - \mu)^2 \end{bmatrix}$$

We take a sample of 100 observations with  $\mu = 1.9$  and  $\sigma^2 = 1.53$ . To find  $\hat{\boldsymbol{\theta}}^{(0)}$ , we must first find a reasonable initial estimate of  $\boldsymbol{\theta}$ . Successive value of  $\hat{\boldsymbol{\theta}}^{(r+1)}$  are defined by (7). The choice of  $\hat{\boldsymbol{\theta}}^{(0)}$  is crucial here.

Table 1: Iterates of the Newton-Raphson algorithm for Normal data in Example 2

r	$\mu$	$\sigma^2$	$\mathcal{L}$	$g$	$H^{-1}$
0	-1.06306	0.830769	-250.510712		
1	-0.54519	1.238011	-222.529209	10.21618	0.031926   -0.01032
				18.73480	-0.01032   0.011907
2	-0.27455	1.470137	-218.546636	1.878127	0.036349   -0.00537
				3.145905	-0.00537   0.015321
3	-0.19747	1.526604	-218.379878	0.100275	0.043628   -0.00154
				0.155778	-0.00154   0.018871
4	-0.19287	1.529701	-218.379406	0.000309	0.046025   -0.00009
				0.000471	-0.00009   0.019938

### 3. The Fisher scoring algorithm

A simple modification of the Newton-Raphson algorithm is the Fisher scoring algorithm. This algorithm replaces  $H$  by  $H^*$  where the (i, j) element of  $H^*$  is

$$\begin{aligned} (H^*)_{ij}(\boldsymbol{\theta}) &= E_{\boldsymbol{\theta}}[H_{ij}\boldsymbol{\theta}] \\ &= E_{\boldsymbol{\theta}}\left[\frac{\partial^2 \mathcal{L}}{\partial \theta_i \partial \theta_j}\right] \end{aligned} \quad (9)$$

The expected value above is computed assuming that  $\theta$  is the true value of the parameter. ( $H$  is observed Fisher information matrix while  $H^*$  is the expected Fisher information matrix.) Now if  $\hat{\theta}^{(r+1)}$  is the estimate of  $\theta$  after  $r$  iterations, we define  $\hat{\theta}^{(r+1)}$  by (7).

The important distinction between Newton-Raphson and Fisher scoring algorithms is the fact that  $H^*(\theta)$  depends on the observed value of  $X, x$ , only through the value of  $\theta$  while  $H(\theta)$  depends, in general on both  $\theta$  and  $x$ .

**Example 3:** As in Example 2, let  $X_1 \dots X_n$  be i.i.d. Normal random variables. From before, we have

$$H(\theta) = \begin{bmatrix} -\frac{n}{\sigma^2} & -\frac{1}{\sigma^4} \sum_{i=1}^n (x_i - \mu) \\ -\frac{1}{\sigma^4} \sum_{i=1}^n (x_i - \mu) & \frac{n}{2\sigma^4} - \frac{1}{\sigma^6} \sum_{i=1}^n (x_i - \mu)^2 \end{bmatrix}$$

and so

$$H^*(\theta) = E(H(\theta)) = \begin{bmatrix} \frac{n}{\sigma^2} & 0 \\ 0 & -\frac{n}{2\sigma^4} \end{bmatrix}$$

Use (7), we can get values for  $\hat{\theta}^{(r+1)}$ . See Table 2.

Table 2: Iterates of the Fisher scoring algorithm for Normal data in Example 3

r	$\mu$	$\sigma^2$	$\mathcal{L}$	$g$	$H^{-1}$
0	-1.06306	0.830769	-250.510712		
1	-0.82262	0.956453	-236.038262	24.19905	0.014781   -0.00477
				46.32173	-0.00477   0.00457
2	-0.62065	1.116267	-226.304793	14.01033	0.017073   -0.00455
				28.59756	-0.00455   0.005831
3	-0.44089	1.286727	-220.832121	6.849986	0.021807   -0.00439
				14.60189	-0.00439   0.008115
4	-0.29595	1.430631	-218.760219	2.465481	0.029267   -0.00380
				5.334796	-0.00380   0.011639
5	-0.21541	1.508326	-218.39655	0.499152	0.038197   -0.00255
				1.086392	-0.00255   0.015745
6	-0.19483	1.527666	-218.379553	0.042945	0.044341   -0.00143
				0.102369	-0.00143   0.018459
7	-0.19297	1.529584	-218.379406	0.002443	0.046041   -0.00108
				0.006327	-0.00108   0.019191
8	-0.19286	1.529703	-218.379406	0.000142	0.046214   -0.00105
				0.000361	-0.00105   0.019265

When using Newton-Raphson or Fisher scoring, you can also try to optimize the stepsize (The  $\delta$  in (8), for example, set  $\delta = \frac{1}{16}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}, 1, 2, 4, 8, \dots$ . In (7),  $\delta = 1$ ). With these techniques, optimizing the stepsize overcomes bad steps where the algorithm attempts to step too far or not far enough toward the solution. Regardless, the methods also employ

(automatically) Marquardt's modification such that the diagonals of the negative Hessian are increased if the matrix is singular.

#### 4. The Fletcher-Reeves and Polak-Ribiere algorithms:

We have seen that the key to the success of Newton-type methods is the curvature information, provided by the Hessian matrix. There are different methods to build up curvature information and one family goes under the name "Conjugate Gradient Methods" as typified by the Fletcher-Reeves algorithm and Polak-Ribiere algorithm.

In Fletcher-Reeves algorithm, we replace  $g\Delta$  in (8) by  $h_i$ , where

$$h_i = g_i + \gamma_{i-1}h_{i-1} \quad (10)$$

$$h_0 = g_0 \quad (11)$$

$$\gamma_i = \frac{\langle g_{i+1}, g_{i+1} \rangle}{\langle g_i, g_i \rangle} \quad (12)$$

**Example 4:** As in Example 2, let  $X_1 \dots X_n$  be i.i.d. Normal random variables. From before, we have

$$g = \left( \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu), -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^n (x_i - \mu)^2 \right)$$

now,

$$h_0 = g_0$$

$$h_1 = g_1 + \frac{\langle g_1, g_1 \rangle}{\langle g_0, g_0 \rangle} g_0$$

$$h_{i+1} = g_i + \frac{\langle g_{i+1}, g_{i+1} \rangle}{\langle g_i, g_i \rangle} h_i$$

Hence, the Fletche-Reeves algorithm is:

$$\hat{\theta}^{(r+1)} = \hat{\theta}^{(r)} + \delta_r h_r$$

Let  $\hat{\theta}^{(0)} = (-1.06306, 0.807692)$ , values for  $\hat{\theta}^{(r)}$  are given in Table 3.

Table 3: Iterates of the Fletche-Reeves algorithm for Normal data in Example 4

r	$\mu$	$\sigma^2$	$\mathcal{L}$	$g$	$H^{-1}$
0	-1.06306	0.807692	-252.090091		
1	-0.19286	1.529706	-218.379406	-0	0   0
				0.000251	0   0

Polak-Ribiere algorithm introduced one tiny but sometimes significant change. They proposed using the form

$$\gamma_i = \frac{\langle (g_{i+1} - g_i), g_{i+1} \rangle}{\langle g_i, g_i \rangle} \quad (13)$$

instead of equation (12). The rest, will keep the same with Fletcher-Reeves.

**Example 5:** As in Example 2, let  $X_1 \dots X_n$  be i.i.d. Normal random variables. From before, we have

$$g = \left( \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu), -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^n (x_i - \mu)^2 \right)$$

$$h_0 = g_0$$

now, we have

$$h_1 = g_1 + \frac{\langle (g_1 - g_0), g_1 \rangle}{\langle g_0, g_0 \rangle} g_0$$

$$h_{i+1} = g_i + \frac{\langle (g_{i+1} - g_i), g_{i+1} \rangle}{\langle g_i, g_i \rangle} h_i$$

Hence, the Polak-Ribiere algorithm is:

$$\hat{\theta}^{(r+1)} = \hat{\theta}^{(r)} + \delta_r h_r$$

Let  $\hat{\theta}^{(0)} = (-1.06906, 0.784615)$ , values for  $\hat{\theta}^{(r)}$  are given in Table 4.

Table 4: Iterates of the Polak-Ribiere algorithm for Normal data in Example 5

r	$\mu$	$\sigma^2$	$\mathcal{L}$	$g$	$H^{-1}$
0	-1.06906	0.784615	-253.972581	-	-
1	-0.19286	1.529706	-218.379406	0	0   -0
				0.000251	-0   0

## 5. The Davidon-Fletcher-Powell (DFP) algorithm and Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm

Another family which build up curvative information goes under the names “quasi-Newton” or “variable metric” methods, as typified by the Davidon-Fletcher-Powell (DFP) or the closely related Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm. Like conjugate gradient methods, these methods require that you are able to compute your function’s gradient, or first partial derivative, at arbitrary points. The basic idea of the variable metric

method is to build up, iteratively, a good approximation to the inverse Hessian matrix, without explicitly forming the Hessian matrix.

In Davidon-Fletcher-Powell (DFP) algorithm, we replace  $(-H(\hat{\boldsymbol{\theta}}^{(r)}))^{-1}g(\hat{\boldsymbol{\theta}}^{(r)})$  in (7) with  $h_r$  where

$$\hat{\boldsymbol{\theta}}^{(r+1)} = \hat{\boldsymbol{\theta}}^{(r)} + \delta_r h_r$$

and

$$H_{r+1} = H_r + \frac{(\boldsymbol{\theta}^{(r+1)} - \boldsymbol{\theta}^{(r)}) \otimes (\boldsymbol{\theta}^{(r+1)} - \boldsymbol{\theta}^{(r)})}{(\boldsymbol{\theta}^{(r+1)} - \boldsymbol{\theta}^{(r)}) \cdot (g_{r+1} - g_r)} - \frac{[H_r \cdot (g_{r+1} - g_r)] \otimes [H_r \cdot (g_{r+1} - g_r)]}{(g_{r+1} - g_r) \cdot H_r \cdot (g_{r+1} - g_r)} \quad (14)$$

Where  $\otimes$  denotes the “outer” or “direct” product of two vectors, a matrix: The  $ij$  component of  $\mathbf{u} \otimes \mathbf{v}$  is  $u_i v_j$ . We can choice  $H_0 = I$  or any symmetric positive definite matrix.

The BFGS updating formular is exactly the same, but with one additional term,

$$\cdots + [(g_{r+1} - g_r) \cdot H_r \cdot (g_{r+1} - g_r)] \mathbf{u} \otimes \mathbf{u} \quad (15)$$

where  $\mathbf{u}$  is defined as the vector

$$\mathbf{u} \equiv \frac{\boldsymbol{\theta}^{(r+1)} - \boldsymbol{\theta}_r}{(\boldsymbol{\theta}_{r+1} - \boldsymbol{\theta}_r) \cdot (g_{r+1} - g_r)} - \frac{H_r \cdot (g_{r+1} - g_r)}{(g_{r+1} - g_r) \cdot H_r \cdot (g_{r+1} - g_r)} \quad (16)$$

**Example 6:** As in Example 2, let  $X_1 \dots X_n$  be i.i.d. Normal random variables. From before, we have

$$g = \left( \frac{1}{\sigma^2} \sum_{i=1}^n (x_i - \mu), -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^n (x_i - \mu)^2 \right)$$

Give  $\hat{\boldsymbol{\theta}}^{(0)} = (-1.79579, 1.084615)$ , use DFP algorithm, values for  $\hat{\boldsymbol{\theta}}^{(r)}$  are given in Table 5

Table 5: Iterates of the Davidon-Fletcher-Powell algorithm for Normal data in Example 6

r	$\mu$	$\sigma^2$	$\mathcal{L}$	$g$	$H^{-1}$
0	-1.79579	1.084615	-267.58353		
1	-0.19286	1.529706	-218.379406	0	0.634937   -0.48144
				0.00025	-0.48144   0.365066

Give  $\hat{\boldsymbol{\theta}}^{(0)} = (1.597597, 3.784615)$ , use Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm, values for  $\hat{\boldsymbol{\theta}}^{(r)}$  are given in Table 6

Table 6: Iterates of the Broyden-Fletcher-Goldfarb-Shanno algorithm for Normal data in Example 6

r	$\mu$	$\sigma^2$	$\mathcal{L}$	$g$	$H^{-1}$
0	1.597597	3.784615	-290.01014		
1	-0.19286	1.529706	-218.379406	0 0.000251	1   -0.09893 -0.09893   0.009781

## References

- [1] James Hardin and Joe Hilbe, *Generalized Linear Models and Extensions* Stata Press, 2001.
- [2] William H. Press, Saul A. Teukosky, Brian P. Flannery *Numerical Recipes in C*, second edition, Cambridge University Press, 1992.
- [3] E. Polak, *Computational Methods in Optimization*, A Unified Approach, University of California, Berkely, California, 1971.
- [4] Philippe Gill, Walter Murray, Margaret H. Wright *Practical Optimization*, System Optimization Laboratory, 1981.
- [5] Keith Knight, *Mathematical Statistics*
- [6] George R. Terrell *Mathematical Statistics*, a Unified Introduction, 1999.
- [7] George Casella, Roger L. Berger *Statistical inference*, North Carolina State University, 1990.