日水研報告，(37)：259—277，1987.
Bull. Jap. Sea Reg. Fish. Res. Lab., (37):259—277, 1987.

Comparison of Algorithms of Several Methods for Estimating Parameters of a Mixture of Normal Distributions

TATSURO AKAMINE[1)]

Abstract

BASIC programs of HASSELBLAD (1966)'s method, GAUSS-SEIDEL method and a variation of it, and MARQUARDT's method for a mixture of normal distributions were compared. Although HASSELBLAD's method converged slowly to the same degree as GAUSS-SEIDEL method, it is useful because it requires less computation. At the present time, MARQUARDT's method is best for the speed of convergence and precision. GAUSS-SEIDEL method and a variation of it, and MARQUARDT's method need to calculate a quadratic differential in a Hessian matrix for bad conditional data.

Key words BASIC program, normal distribution, MARQUARDT, HASSELBLAD, GAUSS-SEIDEL

1. Introduction

HASSELBLAD (1966) provided a useful and highly precise algorithm to estimate parameters for a mixture of normal distributions with the maximum likelihood method. On the other hand, SHIMAZU (1980) used MARQUARDT's method for this problem with a subroutine for a large computer. But the objective functions of this were the least-squares method and the χ^2-minimum method. AKAMINE (1982, 1984) provided BASIC programs with the least-squares method for micro computers. AKAMINE (1985) compared several objective functions by MARQUARDT's method. Then it appeared that the maximum likelihood method is best for good conditional data, and the least-squares method and the χ^2-minimum method are better for bad conditional data (i. e. partly missing data).

AKAMINE (1984) compared MARQUARDT's method, GAUSS-SEIDEL method and the steepest descent method. He mentioned that MARQUARDT's method is the best, GAUSS-SEIDEL method is slow to converge, and the steepest descent method is too slow to use. But the objective function of this was the least-squares method.

In this paper, HASSELBLAD (1966)'s method, GAUSS-SEIDEL method and a variation of it, and MARQUARDT's method are compared. The objective function is the maximum likelihood method.

Received: 21 October, 1986. Contribution A No. 436 from the Japan sea Regional Fisheries Research Laboratory

1) Japan Sea Regional Fisheries Research Laboratory, Suido-cho, Niigata 951, Japan.
〒951 新潟市水道町1丁目5939-22　日本海区水産研究所

II. Algorithm

1. Model

The variables are according to AKAMINE (1985). The model of a mixture of normal distributions is defined as follows:

$$f=\sum_{i=1}^{n} K_i N_i \tag{1.1}$$

$$N_i=N(\mu_i, \sigma_i, x)=\frac{1}{\sqrt{2\pi}\ \sigma_i}\exp\left\{-\frac{1}{2}\left(\frac{x-\mu_i}{\sigma_i}\right)^2\right\}$$

Then, the probability distribution becomes as follows:

$$g=\frac{f}{S}=\sum_{i=1}^{n} P_i N_i \tag{1.2}$$

$$S=\int_{-\infty}^{\infty} f dx=\sum_{i}^{n} K_i, \qquad P_i=\frac{K_i}{S} \tag{1.3}$$

The number of parameters (K_i, μ_i, σ_i) is $3n$ for (1.1). On the other hand, that is $(3n-1)$ for (1.3) because of the restriction S.

Let the data be $x_1 \sim x_T$ (T: number of data). The objective function becomes as follows:

$$\ln L=\sum_{x=x_1}^{x_T} \ln g \tag{1.4}$$

But (1.4) is not efficient for large data, so frequency distribution data was used. Let the data be F_x, where the following relations exist:

$$T=\sum_{x=a}^{b} F_x, \quad G_x=\frac{F_x}{T} \tag{1.5}$$

$b=a+(m-1)h$

G_x: relative frequency distribution
m : number of classes
h : class width
a : minimum class mark
b : maximum class mark

Therefore, (1.4) becomes as follows:

$$\ln L=\sum_{x=a}^{b} F_x \ln g \tag{1.6}$$

In this paper, the objective function is

$$Y=-\ln L \quad (>0). \tag{1.7}$$

The BASIC programs will search for the minimum solution of Y (1.7). It is equivalent to searching for the solution of the following.

$$\boldsymbol{y}=\frac{\partial Y}{\partial \boldsymbol{\theta}}=0 \tag{1.8}$$

2. Iterative method

Non-linear optimization can be solved by iterative methods. Iterative methods are defined as follows:

$$\boldsymbol{y}(\boldsymbol{\theta})=0 \tag{2.1}$$

Solve (2.1) about parameters $\boldsymbol{\theta}$. (2.1) gives

$$\boldsymbol{\theta}=\boldsymbol{z}(\boldsymbol{\theta}). \tag{2.2}$$

(2.2) gives the following interative method.

$$\boldsymbol{\theta}^{\text{new}}=\boldsymbol{z}(\boldsymbol{\theta}^{\text{old}}) \tag{2.3}$$

There are many variations of (2.2). Each variation has a different convergence speed. The algorithms in this paper present how to obtain (2.2) from (2.1).
Generally, (2.3) is described as the following expression to get $\Delta\boldsymbol{\theta}$.

$$\boldsymbol{\theta}^{\text{new}}=\boldsymbol{\theta}^{\text{old}}+\Delta\boldsymbol{\theta} \tag{2.4}$$

3. HASSELBLAD's method

HASSELBLAD (1966) used the iterative method as follows:

$$P_n=1-\sum_{i=1}^{n-1}P_i \tag{3.1}$$

Exclude P_n by (3.1). (1.2) with (3.1) gives

$$\frac{\partial g}{\partial P_i}=N_i-N_n. \qquad (i=1\sim n-1) \tag{3.2}$$

Then he modified (1.8) to get the following expressions:

$$P_i^{\text{new}}=\frac{\left[\sum_x \frac{F_x}{g}N_i\right]^{\text{old}}}{T} \qquad (i=1\sim n-1) \tag{3.3}$$

$$\mu_i^{\text{new}}=\left[\frac{\sum_x \frac{F_x}{g}N_i\,x}{\sum_x \frac{F_x}{g}N_i}\right]^{\text{old}} \qquad (i=1\sim n) \tag{3.4}$$

$$\sigma_i^{2\,\text{new}}=\left[\frac{\sum_x \frac{F_x}{g}N_i\,(x-\mu_i)^2}{\sum_x \frac{F_x}{g}N_i}\right]^{\text{old}} \qquad (i=1\sim n) \tag{3.5}$$

(3.4) and (3.5) are natural generalization of a normal distribution ($n=1$). On the other hand, (3.3) is given skillfully by (3.2). It makes his method successful.

4. GAUSS-SEIDEL method

This method is based on NEWTON's method for 1 parameter. NEWTON's method for 1 parameter to solve $y(\theta)=0$ is as follows:

$$\Delta\theta=-\frac{y}{y'} \tag{4.1}$$

GAUSS-SEIDEL method iterates this along $\theta_1\to\theta_2\to\cdots\to\theta_p\to\theta_1\to\cdots$ ($p=3n-1$: number of parameters). Where

$$y=\frac{\partial Y}{\partial\theta}=-\sum_{x=a}^{b}\frac{F_x}{g}\frac{\partial g}{\partial\theta} \tag{4.2}$$

$$y' = \frac{\partial^2 Y}{\partial \theta^2} = \sum_{x=a}^{b} \frac{F_x}{g} \left\{ \frac{1}{g} \left(\frac{\partial g}{\partial \theta} \right)^2 - \frac{\partial^2 g}{\partial \theta^2} \right\} \tag{4. 3}$$

The second term of (4.3) is omitted for the least-squares method in general (GAUSS-NEWTON method). But when the condition of the data is bad, it is difficult to get good convergence without the second term for the maximum likelihood method. Where

$$\frac{\partial g}{\partial P_i} = N_i - N_n \tag{4. 4}$$

$$\frac{\partial^2 g}{\partial P_i^2} = 0 \qquad (i = 1 \sim n-1)$$

$$\frac{\partial g}{\partial \mu_i} = P_i N_i \frac{x - \mu_i}{\sigma_i^2} \tag{4. 5}$$

$$\frac{\partial^2 g}{\partial \mu_i^2} = P_i N_i \frac{(x-\mu_i)^2 - \sigma_i^2}{\sigma_i^4} \qquad (i = 1 \sim n)$$

$$\frac{\partial g}{\partial \sigma_i} = P_i N_i \frac{(x-\mu_i)^2 - \sigma_i^2}{\sigma_i^3} \tag{4. 6}$$

$$\frac{\partial^2 g}{\partial \sigma_i^2} = P_i N_i \frac{(x-\mu_i)^4 - 5(x-\mu_i)^2 \sigma_i^2 + 2\sigma_i^4}{\sigma_i^6} \qquad (i = 1 \sim n)$$

P_n is corrected by the following expression at the same time of the correction of P_i.

$$\Delta P_n = -\Delta P_i \qquad (i = 1 \sim n-1)$$

But this method is not suitable for the variation of GAUSS-SEIDEL method (next capter). Therefore, use K instead of P. P is calculated from K by (1.3). Use the next expressions instead of (4.4)

$$\frac{\partial Y}{\partial K_i} = -\sum \frac{F}{f} \frac{\partial f}{\partial K_i} + \frac{T}{S} \tag{4. 7}$$

$$\frac{\partial^2 Y}{\partial K_i^2} = \sum \frac{F}{f^2} \left(\frac{\partial f}{\partial K_i} \right)^2 + \frac{T}{S^2} \qquad (i = 1 \sim n)$$

$$\frac{\partial f}{\partial K_i} = N_i, \quad \frac{\partial^2 f}{\partial K_i^2} = 0$$

These are according to AKAMINE (1985). The order of parameters for calculation is as follows:

$$\rightarrow P_1 \rightarrow \cdots\cdots \rightarrow P_n \rightarrow \mu_1 \rightarrow \cdots\cdots \rightarrow \mu_n \rightarrow \sigma_1 \rightarrow \cdots\cdots \rightarrow \sigma_n$$

This program is rewritten from AKAMINE (1982). The objective function is changed from the least-squares method to the maximum likelihood method, and memory is used effectively to decrease the amount of computation. The same method as AKAMINE (1982) to give (4. 1~3) is as follows:

$$Y^* = \sum F_x \ln (g + \Delta g) \tag{4. 8}$$

$$\Delta g = \frac{\partial g}{\partial \theta} \Delta \theta + \frac{1}{2} \frac{\partial^2 g}{\partial \theta^2} (\Delta \theta)^2$$

Then

$$\frac{\partial Y^*}{\partial \Delta \theta} = \sum \frac{F_x \dfrac{\partial \Delta g}{\partial \Delta \theta}}{g + \Delta g}$$

$$\fallingdotseq \sum \frac{F_x}{g}\left(\frac{\partial \Delta g}{\partial \Delta \theta}-\frac{1}{g}\left(1-\frac{\Delta g}{g}\right)\right) \tag{4.9}$$

Substitute (4.8) into (4.9), and omit $(\Delta\theta)^2$ terms.

$$\frac{\partial Y^*}{\partial \Delta\theta} \fallingdotseq \sum \frac{F_x}{g}\left(\frac{\partial g}{\partial \theta}+\frac{\partial^2 g}{\partial \theta^2}\Delta\theta\right)\frac{1}{g}\left(1-\frac{1}{g}\frac{\partial g}{\partial \theta}\Delta\theta\right)$$

$$\fallingdotseq \sum \frac{F_x}{g}\left[\frac{\partial g}{\partial \theta}-\left\{\frac{1}{g}\left(\frac{\partial g}{\partial \theta}\right)^2-\frac{\partial^2 g}{\partial \theta^2}\right\}\Delta\theta\right]=0 \tag{4.10}$$

Then the same expression is given.

5. The steepest descent method

This method is not useful for models with many parameters, but this is the base for the methods presented in the following chapters.

$$\Delta\boldsymbol{\theta}=k\boldsymbol{g} \tag{5.1}$$

$$\boldsymbol{g}=-\boldsymbol{y}=-\frac{\partial Y}{\partial \boldsymbol{\theta}} \qquad \boldsymbol{g}: \text{steepest descent vector}$$

When $\Delta Y<0$, let $k^{\text{new}}=k^{\text{old}}*1.2$ and continue the calculaticn. When $\Delta Y\geqq 0$, iterate $k^{\text{new}}=k^{\text{old}}*0.8$ to get $\Delta Y<0$. $\boldsymbol{g}$ is affected by scaling of the parameters, so adequate scaling is needed. This method is stable, but the convergence is too slow to use (AKAMINE 1984).

6. Variation of GAUSS-SEIDEL method

There are many variations of GAUSS-SEIDEL method. In this paper, the steepest descent method was applied to decrease the time of sum calculation, as follows:

$$\Delta\boldsymbol{\theta}=k\boldsymbol{g}^* \tag{6.1}$$

$$\boldsymbol{g}^*=\left(-\frac{y}{y'}\right)=-\left(\frac{\partial Y}{\partial \theta_i}\Big/\frac{\partial^2 Y}{\partial \theta_i^2}\right)$$

The length of k is changed by the same operation as the steepest descent method. This method does not need scaling of the parameters.

7. MARQUARDT's method

This method is based on NEWTON's method for several parameters. NEWTON's method for several parameters to solve $\boldsymbol{y}(\boldsymbol{\theta})=0$ is as follows:

$$\boldsymbol{H}\,\Delta\boldsymbol{\theta}=\boldsymbol{g} \tag{7.1}$$

$$\boldsymbol{g}=-\boldsymbol{y}=-\frac{\partial Y}{\partial \boldsymbol{\theta}}$$

$$\boldsymbol{H}=\frac{\partial \boldsymbol{y}}{\partial \boldsymbol{\theta}}=\left(\frac{\partial y_i}{\partial \theta_j}\right)=\left(\frac{\partial^2 Y}{\partial \theta_i \partial \theta_j}\right) \qquad \boldsymbol{H}: \text{Hessian matrix}$$

Where, (4.2～6) and the following expressions are used.

$$\frac{\partial^2 Y}{\partial \theta_i \partial \theta_j}=\sum_{x=a}^{b}\frac{F_x}{g}\left(\frac{1}{g}\frac{\partial g}{\partial \theta_i}\frac{\partial g}{\partial \theta_j}-\frac{\partial^2 g}{\partial \theta_i \partial \theta_j}\right) \tag{4.3'}$$

$$\frac{\partial^2 g}{\partial P_i \partial \mu_i}=N_i\frac{x-\mu_i}{\sigma_i^2} \tag{7.2}$$

$$\frac{\partial^2 g}{\partial P_i \partial \sigma_i}=N_i\frac{(x-\mu_i)^2-\sigma_i^2}{\sigma_i^3}$$

$$\frac{\partial^2 g}{\partial \mu_i \partial \sigma_i} = P_i \, N_i \, \frac{(x-\mu_i)^3 - 3\,(x-\mu_i)\,\sigma_i^2}{\sigma_i^5}$$

The other quadratic differential terms are all 0. NEWTON's method solves these simultaneous equations about $\boldsymbol{\theta}$. Although this method converges quickly in the neighborhood of the solution, the domain of convergence is too small to use practically.

MARQUARDT's method enlarges the domain of convergence by λ as follows:

$$(\boldsymbol{H}+\lambda \boldsymbol{I})\Delta\boldsymbol{\theta} = \boldsymbol{g} \tag{7.3}$$

When λ is large, this approaches to the steepest descent method. On the other hand, when λ is small, this approaches to NEWTON's method. Therefore, let λ be large at first, and make it small step by step to get the solution. In this paper, the simplest way was used. Let $\nu=2$. When $\Delta Y<0$, let $\lambda^{\text{new}}=\lambda^{\text{old}}/\nu$ to continue the calculation. On the other hand, when $\Delta Y \geqq 0$, let $\lambda^{\text{new}}=\lambda^{\text{old}} * \nu$ to iterate the same procedure. When $\Delta Y \geqq 0$ after 10 times of enlarging λ continuously, judge it to be the solution to end the calculation.

Scaling of the parameters is necessary because MARQUARDT's method is like the steepest descent method at first. Scaling of the parameters is defined as follows:

$$\boldsymbol{S} = \begin{pmatrix} s_1 & & \\ & \ddots & \\ & & s_n \end{pmatrix} \tag{7.4}$$

$$\boldsymbol{\theta}^* = \boldsymbol{S}\boldsymbol{\theta}, \qquad \Delta\boldsymbol{\theta}^* = \boldsymbol{S}\Delta\boldsymbol{\theta}$$

$$\boldsymbol{g}^* = \boldsymbol{S}^{-1}\boldsymbol{g}, \qquad \boldsymbol{H}^* = \boldsymbol{S}^{-1}\boldsymbol{H}\boldsymbol{S}^{-1}$$

AKAMINE (1984, 1985) used the following scaling.

$$\boldsymbol{S}_2 = \begin{pmatrix} \frac{1}{\theta_1} & & \\ & \ddots & \\ & & \frac{1}{\theta_n} \end{pmatrix} \tag{7.5}$$

But, in this paper, MARQUARDT (1963)'s scaling is used as follows:

$$\boldsymbol{S}_1 = \begin{pmatrix} \sqrt{h_{11}} & & \\ & \ddots & \\ & & \sqrt{h_{nn}} \end{pmatrix}, \qquad \boldsymbol{H} = (h_{ij}) \tag{7.6}$$

III. Experiment

The BASIC program for each algorithm was tested. The programs for HASSELBLAD's method and MARQUARDT's method are shown in appendices. The other methods are omitted. The program for MARQUARDT's method is rewritten from "program 2" of AKAMINE (1985) according to AKAMINE (1986). The scaling method is according to MARQUARDT (1963), and decreases the number of iterations for convergence.

Table 1. The fork length distribution of porgy (data-1).

Class mark	Frequency	Class mark	Frequency	Class mark	Frequency
7.5	7	15.5	1439	23.5	310
8.5	79	16.5	921	24.5	228
9.5	509	17.5	448	25.5	168
10.5	2240	18.5	512	26.5	140
11.5	2341	19.5	719	27.5	114
12.5	623	20.5	673	28.5	64
13.5	476	21.5	445	29.5	22
14.5	1230	22.5	341	30.5	0

Table 2. Artificial data (data-2).

Class mark	Frequency	Class mark	Frequency	Class mark	Frequency
2	9	22	66	42	28
6	53	26	41	46	14
10	81	30	28	50	13
14	47	34	31	54	9
18	49	38	31		

Table 3. Initial values and solutions of the experiments by MARQUARDT's method.

		data-1		data-2	
		Initial value	Solution	Initial value	Solution
P	1	5000	.410596	200	.355643
	2	4000	.305661	150	.339813
	3	3000	.178877	75	.154197
	4	1000	.0823418	50	.109954
	5	500	.0225247	25	.040394
μ	1	11	11.001	10	9.30057
	2	15.5	15.2896	22	21.3996
	3	20	19.7065	33	33.66
	4	24	23.4508	43	41.902
	5	27	27.2564	51	51.6621
σ	1	1	.87321	3.5	3.42403
	2	1	1.13791	3.5	4.18102
	3	1.5	1.41597	3	4.37172
	4	1.5	1.6367	2.5	3.7115
	5	1.5	1.16227	2.5	2.15682
Y		37876.6	37605.8	1935.91	1925.06
Times of iteration		0	13	0	11

Table 4. Convergence process of each method for data-1 ($Y-37600$).

Number of iteration	HASSELBLAD	GAUSS-SEIDEL	(variation)	MARQUARDT
0	276.6	276.6	276.6	276.6
1	42.7	37.7	77.5	91.6
2	21.2	7.7	19.3	37.5
3	12.6	6.8	8.5	18.7
4	9.1	6.7	7.3	11.2
5	7.6	6.6	7.0	7.9
6	6.9	6.6	6.9	6.7
7	6.6	6.5	6.8	6.2
8	6.4	6.5	6.7	6.0
9	6.4	6.4	6.7	5.9
10	6.3	6.4	6.7	5.9
11				5.9
12				5.8
13				5.8
14				5.8 end
20	6.2	6.2	6.4	
30	6.2	6.1	6.2	
36			6.1 end	
40	6.1	6.1		
50	6.1	6.0		
60	6.1	6.0		
70	6.1	6.0		
80	6.0	6.0		
90	6.0	5.9		
100	6.0	5.9		
200	6.0			
300	5.9			
400	5.9			
500	5.9			
Time for 1 iteration (second)	5	50	10	30

The micro computer used in this experiment was a PC-9801F (8MHz, NEC). Although the N_{88}-BASIC (86) of this machine includes bugs in the "PRINT" statement and "SQR" function, among others (NISHIMI 1986), careful treatment will give correct results.

The data used for this experiment is shown in Tables 1 and 2. Data-1 is the fork length of porgy (TANAKA 1956). Data-2 is artificial data. The condition of data-2 is so bad that GAUSS-SEIDEL method and a variation of it and MARQUARDT's method could not reach a correct solution without the quadratic differential term of (4.3). Therefore, these methods need the quadratic differential term of (4.3) for the maximum likelihood method.

Table 5. Convergence process of each method for data-2 ($Y-1925$).

Number of iteration	HASSELBLAD	GAUSS-SEIDEL	(variation)	MARQUARDT
0	10.91	10.91	10.91	10.91
1	2.00	1.99	4.21	3.75
2	1.35	.71	2.70	1.76
3	1.02	.41	.72	.87
4	.81	.26	.49	.42
5	.66	.18	.34	.19
6	.55	.14	.25	.10
7	.47	.11	.21	.08
8	.40	.09	.16	.07
9	.34	.08	.14	.07
10	.30	.08	.12	.06 end
20	.12	.07	.08	
30	.08	.07	.07	
39			.07 end	
40	.07	.07		
50	.07	.07		
60	.07	.07		
70	.07	.07		
80	.07	.07		
90	.06	.07		
100	.06	.07		

The results of the experiment are shown in Tables 3～5. It shows that:

(1) The convergence of HASSELBLAD's method is the same degree of that of GAUSS-SEIDEL method. But the amount of computation is very small. The total time required for calculation is small in practice. The BASIC program and memory needed are also small. It converged to the correct solution for bad conditional data-2. Thus this is a useful method, but it required more than 200 iterations for a correct solution.

(2) MARQUARDT's method converges to the correct solution in a short time. It judges the end of convergence too. It seems the best method at the present time. Although the adequate initial value of λ is different by objective functions, etc., it is adequate to let it be 1 in this case.

(3) GAUSS-SEIDEL method needs a long time to converge because it needs a lot of computation. It is difficult to get a correct solution for bad conditional data-2. Although a variation of it was speeded up by decreasing the amount of computation, it did not converge to the correct solution. Many variations of GAUSS-SEIDEL method have been considered, but they do not seem to be better than MARQUARDT's method.

IV. Conclusion

MARQUARDT's method seems to be the best method for this problem at the present time. Although this method has many modifications (NAKAGAWA and OYANAGI 1982) and this program is not perfect, these seem to be adequate for present micro computers.

HASSELBLAD's method is useful because it does not need a lot of memory and computational time, but it is necessary to get a correct solution to iterate more than 200 times.

GAUSS-SEIDEL method and variations of it have no merit in practice at the present time.

Acknowledgement

The author is indebted to Mr. F. KATO of Seikai Regional Fisheries Research Laboratory and Mr. K. ISHIOKA of Nansei Regional Fisheries Research Laboratory for their kind advice and help. The author is also grateful to Mr. K. NOGAMI and Mr. K. TANAKA of Japan Sea Regional Fisheries Research Laboratory for their critical readings of the manuscript.

References

AKAMINE, T. (1982) A BASIC program to analyse the polymodal frequency distribution into normal distributions. *Bull. Jap. Sea Reg. Fish. Res. Lab.*, (33), 163–166. *in Japanese with English summary.*

——— (1984) The BASIC program to analyse the polymodal frequency distribution into normal distributions with MARQUARDT's method. *Bull. Jap. Sea Reg. Fish. Res. Lab.*, (34), 53–60. *in Japanese with English summary.*

——— (1985) Consideration of the BASIC programs to analyse the polymodal frequency distribution into normal distributions. *Bull. Jap. Sea Reg. Fish. Res. Lab.*, (35), 129–160. *in Japanese with English summary.*

——— (1986) Expansion of growth curves using a periodic function and BASIC programs by MARQUARDT's method. *Bull. Jap. Sea Reg. Fish. Res. Lab.*, (36), 77–107.

HASSELBLAD, V. (1966) Estimation of parameters for a mixture of normal distributions. *Technometrics*, 8(3), 431–444.

MARQUARDT, D. W. (1963) An algorithm for least-squares estimation of nonlinear parameters. *J. Soc. Indust. Appl. Math.*, 11 (2), 431–441.

NAKAGAWA, T. and Y. OYANAGI (1982) 最小二乗法による実験データ解析——プログラム **SALS**, 東京大学出版会, 東京, 206pp.

NISHIMI, J. (1986) N_{88}-BASIC (86) にある数学関数のバグと対策. インターフェース, (2), 280–289.

SHIMAZU, Y. (1980) 体長組成から年齢組成を推定する一方法. 昭和54年度漁業資源研究会議, 西日本底魚部会会議報告, 36–48.

TANAKA, S. (1956) A method of analysing the polymodal frequency distribution and its application to the length distribution of porgy, *Taius tumifrons* (T. & S.). *Bull. Tokai Reg. Fish. Res. Lab.*, (14), 1–13. *in Japanese with English summary.*

多重正規分布のパラメータを推定するアルゴリズムの比較

赤　嶺　達　郎

Hasselblad (1966) の方法，Gauss-Seidel 法とその改良法，Marquardt 法の BASIC プログラムを作成し比較検討した．Hasselblad の方法は Gauss-Seidel 法と同程度の収束を示したが，計算量が少ないので十分に実用的である．現状では Marquardt 法が速さと精度の両面で最良である．条件の悪いデータについては，Gauss-Seidel 法とその改良法および Marquardt 法ではヘシアン行列で二次微分項の計算が必要である．

Appendix A. The BASIC program to estimate parameters for the mixture of normal distributions by MARQUARDT's method.

```
10     '
20     '    Polymodal-model by Marquardt's method
30     '
40     '                    from Program 2 of Akamine(1985)
50     '                    Scaling method of Marquardt(1963)
60     '
100    '     main routine
110    '
120    GOSUB *VARIDEF
130    GOSUB *DATAREAD
140    GOSUB *INITIAL
150    FOR IREP=1 TO NIT
160       GOSUB *SUMUP
170       GOSUB *CALEQAT
180    NEXT IREP
190    PRINT "**WARNING**      Convergence was not completed."
200    PRINT
210    IREP=IREP-1
220    GOSUB *PRINTOUT
230    END
500   *PEND2
510    PRINT "Convergence was completed."
520    PRINT
530    IREP=IREP-1
540    GOSUB *PRINTOUT
550    END
1000  *VARIDEF
1010   '
1020   '    Definition of functions
1030   '
1040   DEFINT I-N
1050   PAI2=.3989423#
1060   DEF FNBND(I,X)=PAI2/SD(I)*EXP(-.5*(X-AM(I))*(X-AM(I))/SD(I)/SD(I))
1070   RETURN
2000  *DATAREAD
2010   '
2020   '    Reading of data and variables
2030   '
2040   READ CLAMBDA,CNU,NIT,NND,AMCM,NCL,CWD
2050   PRINT "Number of normal distributions =";NND
2060   PRINT "Minimum class mark             =";AMCM
2070   PRINT "Number of classes              =";NCL
2080   PRINT "Class width                    =";CWD
2090   PRINT "Number of iterations           =";NIT
2100   PRINT "           Lambda              =";CLAMBDA
2110   PRINT "             Nu                =";CNU
2120   PRINT
2130   NP=3*NND
2140   DIM F(NCL),X(NCL),GX(NCL),DIFFER(NP),BND(NND,NCL)
2150   DIM HESSIAN(NP,NP),GVECTOR(NP),PDELTA(NP)
2160   DIM SCALE(NP),PA(NND),AM(NND),SD(NND)
2170   DIM PAOLD(NND),AMOLD(NND),SDOLD(NND)
2180   SOWA=0
2190   FOR K=1 TO NCL
2200      X(K)=AMCM+(K-1)*CWD
2210      READ F(K)
2220      PRINT "F(";X(K);")=";F(K)
2230      SOWA=SOWA+F(K)
2240   NEXT K
2250   PRINT
2260   PRINT "Sum of data =";SOWA
2270   PRINT
2280   RETURN
```

```
3000 *INITIAL
3010  '
3020  '    Setting of initial condition
3030  '
3040  SOWA2=0
3050  FOR I=1 TO NND
3060     READ PA(I),AM(I),SD(I)
3070     AMOLD(I)=AM(I) : SDOLD(I)=SD(I)
3080     SOWA2=SOWA2+PA(I)
3090  NEXT I
3100  FOR I=1 TO NND : PA(I)=PA(I)/SOWA2 : PAOLD(I)=PA(I) : NEXT I
3110  GOSUB *CLL2
3120  Y1=Y2
3130  IREP=0 : YDELTA=0
3140  GOSUB *PRINTOUT
3150  RETURN
4000 *SUMUP
4010  '
4020  '   Calculation of HESSIAN and GVECTOR
4030  '
4040  FOR I=1 TO NP
4050     GVECTOR(I)=0
4060     FOR J=I TO NP
4070        HESSIAN(I,J)=0
4080     NEXT J
4090  NEXT I
4100  FOR K=1 TO NCL
4110     F7=F(K)/GX(K) : F6=F7/GX(K)
4120     FOR I=1 TO NND
4130        IS3=I : IS2=IS3+NND : IS1=IS2+NND
4140        P2=X(K)-AM(I) : PP2=P2*P2 : PPP2=PP2*PP2
4150        P3=SD(I)      : PP3=P3*P3 : PPP3=PP3*PP3 : PPQ3=PP3*PPP3
4160        P1=PA(I)
4170        BNDP=P1*BND(I,K) : F8=F7*BNDP
4180        DIFFER(IS1)=BND(I,K)-BND(NND,K)
4190        DIFFER(IS2)=BNDP*P2/PP3
4200        DIFFER(IS3)=BNDP*(P2*P2-PP3)/PP3/P3
4210        HESSIAN(IS3,IS3)=HESSIAN(IS3,IS3)-F8*(PPP2-5*PP2*PP3+2*PPP3)/PPQ3
4220        HESSIAN(IS3,IS2)=HESSIAN(IS3,IS2)-F8*P2*(PP2-3*PP3)/PPP3/P3
4230        HESSIAN(IS3,IS1)=HESSIAN(IS3,IS1)-F8*(PP2-PP3)/PP3/P3/P1
4240        HESSIAN(IS2,IS2)=HESSIAN(IS2,IS2)-F8*(PP2-PP3)/PPP3
4250        HESSIAN(IS2,IS1)=HESSIAN(IS2,IS1)-F8*P2/PP3/P1
4260     NEXT I
4270     '
4280     FOR I=1 TO NP
4290        GVECTOR(I)=GVECTOR(I)+F7*DIFFER(I)
4300        FOR J=I TO NP
4310           HESSIAN(I,J)=HESSIAN(I,J)+F6*DIFFER(I)*DIFFER(J)
4320        NEXT J
4330     NEXT I
4340  NEXT K
4350  '
4360  NP=NP-1
4370  '
4380  '   Scaling of parameter
4390  '
4400  FOR I=1 TO NP
4410     SCALE1=SQR(HESSIAN(I,I))
4420     SCALE(I)=.5*(SCALE1+HESSIAN(I,I)/SCALE1)
4430  '          for BUG of SQR-function of N88-BASIC(86)
4440  NEXT I
4450  FOR I=1 TO NP
4460     GVECTOR(I)=GVECTOR(I)/SCALE(I)
4470     FOR J=I TO NP
4480        HESSIAN(I,J)=HESSIAN(I,J)/SCALE(I)/SCALE(J)
4490     NEXT J
4500  NEXT I
```

```
4510   '
4520   '   Reserve of HESSIAN and GVECTOR
4530   '
4540   FOR I=2 TO NP
4550      HESSIAN(I,1)=GVECTOR(I)
4560   NEXT I
4570   FOR I=2 TO NP-1
4580      FOR J=I+1 TO NP
4590         HESSIAN(J,I)=HESSIAN(I,J)
4600      NEXT J
4610   NEXT I
4620   RETURN
5000  *CALEQAT
5010   '
5020   '   Solution of the simultaneous linear equations
5030   '
5040   K2=0
5050  *REPEAT
5060   K2=K2+1
5070   IF K2>11 GOTO *PEND2
5080   PRINT "K=";K2
5090   PRINT "      LAMBDA =";CLAMBDA
5100   PRINT
5110   FOR I=1 TO NP
5120      HESSIAN(I,I)=1+CLAMBDA
5130   NEXT I
5140   GOSUB *GAUSS
5150   '
5160   '   Correction of parameters
5170   '
5180   FOR I=1 TO NP
5190      PDELTA(I)=PDELTA(I)/SCALE(I)
5200   NEXT I
5210   FOR I=1 TO NND
5220      IS3=I : IS2=IS3+NND : IS1=IS2+NND
5230      PA(I)=PAOLD(I)+PDELTA(IS1)
5240      AM(I)=AMOLD(I)+PDELTA(IS2)
5250      SD(I)=SDOLD(I)+PDELTA(IS3)
5260   NEXT I
5270   SOWA3=0
5280   FOR I=1 TO NND-1
5290      SOWA3=SOWA3+PA(I)
5300   NEXT I
5310   PA(NND)=1-SOWA3
5320   '
5330   GOSUB *CLL2
5340   IF Y2>=Y1 GOTO *PREREP
5350   '
5360   CLAMBDA=CLAMBDA/CNU
5370   YDELTA=Y2-Y1
5380   Y1=Y2
5390   '
5400   NP=NP+1
5410   '
5420   FOR I=1 TO NND
5430      PAOLD(I)=PA(I) : AMOLD(I)=AM(I) : SDOLD(I)=SD(I)
5440   NEXT I
5450   GOSUB *PRINTOUT
5460   RETURN
```

```
5500  *PREREP
5510   '
5520   '   Calculation again with larger lambda
5530   '
5540   CLAMBDA=CLAMBDA*CNU
5550   FOR I=2 TO NP
5560      GVECTOR(I)=HESSIAN(I,1)
5570   NEXT I
5580   FOR I=2 TO NP-1
5590      FOR J=I+1 TO NP
5600         HESSIAN(I,J)=HESSIAN(J,I)
5610      NEXT J
5620   NEXT I
5630   GOTO *REPEAT
6000  *CLL2
6010   '
6020   '    Calculation of object function
6030   '
6040   Y2=0
6050   FOR K=1 TO NCL
6060      F1=0
6070      FOR I=1 TO NND
6080         BND(I,K)=FNBND(I,X(K))
6090         F1=F1+PA(I)*BND(I,K)
6100      NEXT I
6110      Y2=Y2-F(K)*LOG(F1)
6120      GX(K)=F1
6130   NEXT K
6140   RETURN
7000  *GAUSS
7010   '
7020   '   GAUSS' elimination method
7030   '
7040   FOR I=1 TO NP-1
7050      FOR K=I+1 TO NP
7060         Q1=HESSIAN(I,K)/HESSIAN(I,I)
7070         GVECTOR(K)=GVECTOR(K)-Q1*GVECTOR(I)
7080         FOR J=K TO NP
7090            HESSIAN(K,J)=HESSIAN(K,J)-Q1*HESSIAN(I,J)
7100         NEXT J
7110      NEXT K
7120   NEXT I
7130   '
7140   PDELTA(NP)=GVECTOR(NP)/HESSIAN(NP,NP)
7150   FOR I=NP-1 TO 1 STEP -1
7160      T1=GVECTOR(I)
7170      FOR J=I+1 TO NP
7180         T1=T1-PDELTA(J)*HESSIAN(I,J)
7190      NEXT J
7200      PDELTA(I)=T1/HESSIAN(I,I)
7210   NEXT I
7220   RETURN
8000  *PRINTOUT
8010   '
8020   '   Output for CRT
8030   '
8040   PRINT "IREP=";IREP
8050   PRINT "      logL      =";Y1
8060   PRINT "   Delta logL   =";YDELTA
8070   PRINT
8080   FOR I=1 TO NND
8090      PRINT "I=";I
8100      PRINT "Area=";PA(I),"Mean=";AM(I),"S.D.=";SD(I)
8110   NEXT I
8120   PRINT
8130   RETURN
```

```
10000  '
10010  DATA 1 , 2 , 20              :' lambda , nu , NIT
10020  '
10030  '   Example data of body length (Porgy)
10040  '
10050  DATA 5 , 7.5 , 24 , 1        :' NND , AMCM , NCL , CED
10060  '
10070  '           histogram
10080  '
10090  DATA 7 , 79 , 509 , 2240 , 2341 , 623 , 476 , 1230 , 1439 , 921
10100  DATA 448 , 512 , 719 , 673 , 445 , 341 , 310 , 228 , 168 , 140
10110  DATA 114 , 64 , 22 , 0
10120  '
10130  '   Example initial value of parameter
10140  '
10150  DATA 5000 , 11   , 1         :' PA(1) , AM(1) , SD(1)
10160  DATA 4000 , 15.5 , 1
10170  DATA 3000 , 20   , 1.5
10180  DATA 1000 , 24   , 1.5
10190  DATA  500 , 27   , 1.5       :' PA(5) , AM(5) , SD(5)
30000  '
30010  '  Notation of variables
30020  '
30030  '  NND        :  number of normal distributions
30040  '  AMCM       :  minimum class mark
30050  '  NCL        :  number of classes
30060  '  CWD        :  class width
30070  '  NIT        :  number of iterations
30080  '  CLAMBDA    :  lambda
30090  '  CNU        :  nu
30100  '  F(K)       :  F
30110  '  X(K)       :  x
30120  '  GX(K)      :  f
30130  '  SOWA       :  T
30140  '  SOWA3      :  sum of P(1 --- NND-1)
30150  '  DIFFER     :  df/d(parameter)
30160  '  HESSIAN    :  hessian matrix
30170  '  GVECTOR    :  gradient vector
30180  '  SCALE      :  scaling matrix
30190  '  PDELTA     :  delta (parameter)
30200  '  Y1         :  old Y
30210  '  Y2         :  new Y
30220  '  YDELTA     :  delta Y
30230  '  NP         :  number of parameters
30240  '  PA(I)      :  new P(i)
30250  '  AM(I)      :  new mu(i)
30260  '  SD(I)      :  new sigma(i)
30270  '  PAOLD(I)   :  old P(i)
30280  '  AMOLD(I)   :  old mu(i)
30290  '  SDOLD(I)   :  old sigma(i)
```

Appendix B. The BASIC program to estimate parameters for the mixture of normal distributions by HASSELBLAD (1966)'s method.

```
10    '
20    '   Polymodal-model by Hasselblad(1966)'s method
30    '
100   '   main routine
110   '
120   GOSUB *VARIDEF
130   GOSUB *DATAREAD1
140   GOSUB *INITIAL1
150   FOR IREP=1 TO NIT
160      GOSUB *CALCU1
170   NEXT IREP
190   END
1000 *VARIDEF
1010  '
1020  '   Definition of functions
1030  '
1040  DEFINT I-N
1050  PAI2=.3989423#
1060  DEF FNBND(I,X)=PAI2/SD(I)*EXP(-.5*(X-AM(I))*(X-AM(I))/SD(I)/SD(I))
1070  RETURN
2000 *DATAREAD1
2010  '
2020  '   Reading of data and variables
2030  '
2040  READ NIT,NND,AMCM,NCL,CWD
2050  PRINT "Number of normal distributions =";NND
2060  PRINT "Minimum class mark             =";AMCM
2070  PRINT "Number of classes              =";NCL
2080  PRINT "Class width                    =";CWD
2090  PRINT "Number of iterations           =";NIT
2100  PRINT
2110  DIM F(NCL),X(NCL),GX(NCL),PA(NND)
2120  DIM AM(NND),AM1(NND),AM2(NND)
2130  DIM SD(NND),SD1(NND),BND(NND,NCL)
2140  SOWA=0
2150  FOR K=1 TO NCL
2160     X(K)=AMCM+(K-1)*CWD
2170     READ F(K)
2180     PRINT "F(";X(K);")=";F(K)
2190     SOWA=SOWA+F(K)
2200  NEXT K
2210  PRINT
2220  PRINT "Sum of data =";SOWA
2230  PRINT
2240  RETURN
3000 *INITIAL1
3010  '
3020  '   Setting of initial condition
3030  '
3040  SOWA2=0
3050  FOR I=1 TO NND
3060     READ PA(I),AM(I),SD(I)
3070     SOWA2=SOWA2+PA(I)
3080  NEXT I
3090  FOR I=1 TO NND : PA(I)=PA(I)/SOWA2 : NEXT I
3100  GOSUB *CLL2
3110  Y1=Y2
3120  IREP=0 : YDELTA=0
3130  GOSUB *PRINTOUT
3140  RETURN
```

```
5000  *CALCU1
5010   '
5020   '    The main part of calculation
5030   '
5040   FOR I=1 TO NND
5050      AM1(I)=0 : AM2(I)=0 : SD1(I)=0
5060   NEXT I
5070   FOR K=1 TO NCL
5080      F7=F(K)/GX(K)
5090      FOR I=1 TO NND
5100         F9=F7*BND(I,K)
5110         AM1(I)=AM1(I)+F9*X(K)
5120         AM2(I)=AM2(I)+F9
5130         SD1(I)=SD1(I)+F9*(X(K)-AM(I))*(X(K)-AM(I))
5140      NEXT I
5150   NEXT K
5160   SOWA3=0
5170   FOR I=1 TO NND-1
5180      PA(I)=AM2(I)*PA(I)/SOWA  : SOWA3=SOWA3+PA(I)
5190   NEXT I
5200   PA(NND)=1-SOWA3
5210   FOR I=1 TO NND
5220      AM(I)=AM1(I)/AM2(I)
5230      ASD1=SD1(I)/AM2(I)
5240      ASD2=SQR(ASD1)
5250      SD(I)=.5*(ASD2+ASD1/ASD2)
5260      '       for BUG of SQR-function of N88-BASIC(86)
5270   NEXT I
5280   GOSUB *CLL2
5290   YDELTA=Y2-Y1
5300   Y1=Y2
5310   GOSUB *PRINTOUT
5320   RETURN
6000  *CLL2
6010   '
6020   '    Calculation of object function
6030   '
6040   Y2=0
6050   FOR K=1 TO NCL
6060      F1=0
6070      FOR I=1 TO NND
6080         BND(I,K)=FNBND(I,X(K))
6090         F1=F1+PA(I)*BND(I,K)
6100      NEXT I
6110      Y2=Y2-F(K)*LOG(F1)
6120      GX(K)=F1
6130   NEXT K
6140   RETURN
8000  *PRINTOUT
8010   '
8020   '   Output for CRT
8030   '
8040   PRINT"IREP=";IREP
8050   PRINT "     logL      =";Y1
8060   PRINT "   Delta logL  =";YDELTA
8070   PRINT
8080   FOR I=1 TO NND
8090      PRINT "I=";I
8100      PRINT "Area=";PA(I),"Mean=";AM(I),"S.D.=";SD(I)
8110   NEXT I
8120   PRINT
8130   RETURN
```

```
10000  '
10010  DATA 200                    :' NIT
10020  '
10030  '   Example data of body length (Porgy)
10040  '
10050  DATA 5 , 7.5 , 24 , 1       :' NND , AMCM , NCL , CED
10060  '
10070  '           histogram
10080  '
10090  DATA 7 , 79 , 509 , 2240 , 2341 , 623 , 476 , 1230 , 1439 , 921
10100  DATA 448 , 512 , 719 , 673 , 445 , 341 , 310 , 228 , 168 , 140
10110  DATA 114 , 64 , 22 , 0
10120  '
10130  '   Example initial value of parameter
10140  '
10150  DATA 5000 , 11   , 1        :' PA(1) , AM(1) , SD(1)
10160  DATA 4000 , 15.5 , 1
10170  DATA 3000 , 20   , 1.5
10180  DATA 1000 , 24   , 1.5
10190  DATA  500 , 27   , 1.5      :' PA(5) , AM(5) , SD(5)
30000  '
30010  '  Notation of variables
30020  '
30030  '  NND       :  number of normal distributions
30040  '  AMCM      :  minimum class mark
30050  '  NCL       :  number of classes
30060  '  CWD       :  class width
30070  '  NIT       :  number of iterations
30080  '  F(K)      :  F
30090  '  X(K)      :  x
30100  '  GX(K)     :  f
30110  '  SOWA      :  T
30120  '  SOWA3     :  sum of P(1 --- NND-1)
30130  '  Y1        :  old Y
30140  '  Y2        :  new Y
30150  '  YDELTA    :  delta Y
30160  '  PA(I)     :  P(i)
30170  '  AM(I)     :  mu(i)
30180  '  SD(I)     :  sigma(i)
```