マルカール法によるマルチコホートモデルの解法

誌名	日本海区水産研究所研究報告 = Bulletin of Japan Sea Regional Fisheries Research Laboratory
ISSN	00214620
著者	赤嶺, 達郎
巻/号	37号
掲載ページ	p. 225-257
発行年月	1987年1月

日水研報告，(37)：225—257，1987．
Bull. Jap. Sea Reg. Fish. Res. Lab., (37) : 225—257, 1987.

A Solution of the Multi-Cohort Model by MARQUARDT's Method

TATSURO AKAMINE[1)]

Abstract

The multi-cohort model was solved by the BASIC program rewritten from AKAMINE (1986). The neighborhood of the solution was examined by using artificial data. The multi-cohort model has too many parameters to produce a fast convergence and a stable solution. There are high positive correlations between virgin stocks and the natural mortality coefficient, and high negative correla tions between these and the fishing mortality coefficients.

Key words multi-cohort, BASIC program, MARQUARDT

I. Introduction

Multi-cohort models were introduced in detail by SHIMAZU (1983). He said that the methods of DOUBLEDAY (1976) and POPE and SHEPHERD (1982) still include problems and it is necessary to develop and apply robust methods of non-linear estimation. The author had already applied MARQUARDT's method to parameter estimations of a mixture of normal distributions and the growth curves (AKAMINE 1985, 1986). In this paper, MARQUARDT's method was applied to a multi-cohort model and all parameters including a natural mortality coefficient have been estimated. The Hessian matrix of the solution has been analysed to estimate the correlation of parameters in the neighborhood.

II. Model

The multi-cohort model in this paper is defined as follows:

$$\begin{cases} C_{ij}=N_{ij}\dfrac{F_{ij}}{F_{ij}+M}\{1-\exp\{-(F_{ij}+M)\}\} & (2.1) \\ N_{ij}=N_{i-1,j-1}\exp\{-(F_{i-1,j-1}+M)\} & (2.2) \end{cases}$$

where

C_{ij} : catch at age j in year i
N_{ij} : stock at age j in year i

Received: 21 October, 1986. Contribution A No. 435 from 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

F_{ij} : fishing mortality coefficient at age j in year i
M : natural mortality coefficient

Cohort analysis is the method of estimating N_{i1}, N_{1j}, F_{ij}, M (parameters) from C_{ij} (data). But this model has more parameters than data, so this model can not be solved. Therefore multi-cohort models use the next condition to decrease the number of parameters.

$$F_{ij}=f_i s_j \tag{2.3}$$

Using f_i and s_j instead of F_{ij} decreases the number of parameters from nm to $(n+m)$. But f_i and s_j have 1 more freedom as follows:

$$F_{ij}=f_i s_j=\frac{1}{k} f_i \cdot k s_j$$

Because F_{ij} is not affected by k, k can be ignored in calculations. But for the purpose of comparision, s is standardized in this paper as follows:

$$\sum_j^m s_j=1 \tag{2.4}$$

This restrictive condition defines f_i and s_j as unique.

Let the data of catch be C^0_{ij} $(i=1\sim n,\ j=1\sim m)$. Then the number of the data is nm. On the other hand, the parameters are N_{i1} $(i=1\sim n)$, N_{1j} $(j=2\sim m)$, f_i $(i=1\sim n)$, s_j $(j=1\sim m)$, M. But, the restrictive condition of (2.4) makes the freedom of the parameters 1 less, so the number of the parameters becomes $2(n+m)-1$. Then the condition for the solutions to exist becomes as follows:

$$\begin{aligned} &nm \geqq 2(n+m)-1 \\ &(n-2)(m-2) \geqq 3 \end{aligned} \tag{2.5}$$

Let the objective function be the least-squares method as follows:

$$Y=\sum_i^n \sum_j^m (C_{ij}-C^0_{ij})^2 \tag{2.6}$$

The other objective functions (i. e. the minimum χ^2 estimation method, the least-squares method for data transformed to a logarithmic base) are omitted. The problem as to which is the best objective function is difficult to answer, because it is concerned with what causes the errors in the catch.

The relation between the parameters and data is illustrated in Fig. 1. Each cohort advances to the lower right of the matrix. M is the only parameter concerned with all the data. About N, f and s, the larger i or j of the parameter becomes, the smaller the number of data concerned with it. Especially N_{1m}, N_{n1} is concerned with only 1 data C^0_{1m}, C^0_{n1}. This relation is easily understood by the expression of this model (2.1, 2.2) as follows:

$$C_{ij}=\begin{cases} G(N_{1,j-i+1},\ f_1,\ \sim,\ f_i,\ s_{j-i+1},\ \sim,\ s_j,\ M) & (i<j) \\ G(N_{i-j+1,1},\ f_{i-j+1},\sim,\ f_i,\ s_1,\ \sim,\ s_j,\ M) & (i\geqq j) \end{cases} \tag{2.7}$$

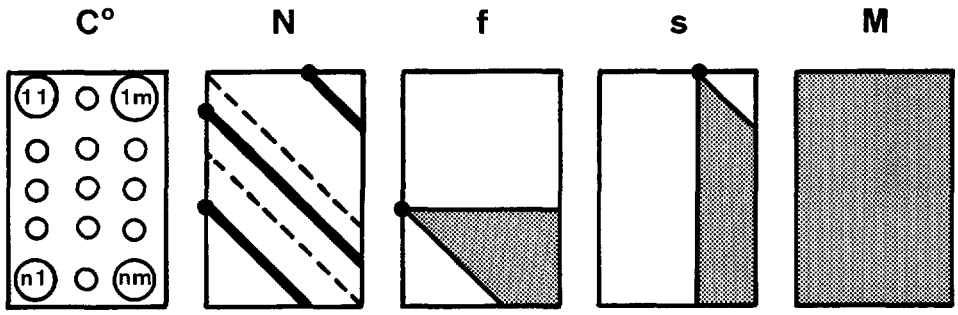


Fig. 1. The relation of data (C^0) and parameters (N, f, s, M).

III. The calculating method

1. MARQUARDT's method

The algorithm of MARQUARDT's method is described easily as follows: Let Y be the objective function, $\boldsymbol{\theta}$ be the parameters. It is expressed as follows in the case of searching the minimal point.

$$(\boldsymbol{H}+\lambda\boldsymbol{I})\,\Delta\boldsymbol{\theta}=\boldsymbol{g} \tag{3.1}$$

$$\boldsymbol{H}=\left(\frac{\partial^2 Y}{\partial\theta_i\partial\theta_j}\right),\quad \boldsymbol{g}=-\frac{\partial Y}{\partial\boldsymbol{\theta}}$$

$$\begin{cases} \boldsymbol{H} & : \text{Hessian matrix} \\ \boldsymbol{I} & : \text{unit matrix} \\ \boldsymbol{g} & : \text{gradient vector} \end{cases}$$

Let Y be as follows:

$$Y=Y(y),\quad y=C_{ij} \tag{3.2}$$

(3. 1) becomes as follows:

$$\frac{\partial Y}{\partial\theta_i}=\frac{\partial Y}{\partial y}\frac{\partial y}{\partial\theta_i} \tag{3.3}$$

$$\frac{\partial^2 Y}{\partial\theta_i\partial\theta_j}=\frac{\partial^2 Y}{\partial y^2}\frac{\partial y}{\partial\theta_i}\frac{\partial y}{\partial\theta_j}+\frac{\partial Y}{\partial y}\frac{\partial^2 y}{\partial\theta_i\partial\theta_j}$$

The second term of the second equation is usually omitted, because it is smaller than the first term for the least-squares method. Then the second equation becomes as follows (GAUSS-NEWTON method):

$$\frac{\partial^2 Y}{\partial\theta_i\partial\theta_j}=\frac{\partial^2 Y}{\partial y^2}\frac{\partial y}{\partial\theta_i}\frac{\partial y}{\partial\theta_j} \tag{3.4}$$

(2. 6) becomes as follows:

$$\begin{cases} \dfrac{\partial Y}{\partial y}=2\sum_{i}^{n}\sum_{j}^{m}(C_{ij}-C^{0}_{ij}) \\ \dfrac{\partial^2 Y}{\partial y^2}=2\sum_{i}^{n}\sum_{j}^{m} \end{cases} \tag{3. 5}$$

These are obtained formally, so it is necessary to be careful that $\sum$ affects the following terms. Therefore $\boldsymbol{H}$ and $\boldsymbol{g}$ becomes as follows for the least-squares method.

$$\boldsymbol{H}=(h_{pq})=2\sum_{i}^{n}\sum_{j}^{m}\frac{\partial C_{ij}}{\partial\theta_p}\frac{\partial C_{ij}}{\partial\theta_q}=2{}^t\boldsymbol{A}\boldsymbol{A}$$

$$\boldsymbol{g}=(g_p)=-2\sum_{i}^{n}\sum_{j}^{m}(C_{ij}-C^{0}_{ij})\frac{\partial C_{ij}}{\partial\theta_p}=2{}^t\boldsymbol{A}\boldsymbol{d}$$

$$\boldsymbol{A}=\begin{pmatrix} \dfrac{\partial C_{11}}{\partial\theta_1} & \cdots & \dfrac{\partial C_{11}}{\partial\theta_l} \\ \vdots & & \vdots \\ \dfrac{\partial C_{nm}}{\partial\theta_1} & \cdots & \dfrac{\partial C_{nm}}{\partial\theta_l} \end{pmatrix}, \qquad \boldsymbol{d}=\begin{pmatrix} C^{0}_{11}-C_{11} \\ \vdots \\ C^{0}_{nm}-C_{nm} \end{pmatrix}$$

$\boldsymbol{A}$: Jacobian matrix
${}^t\boldsymbol{A}$: transposed matrix of $\boldsymbol{A}$
$\boldsymbol{d}$: residual vector
$l = 2(n+m)-1$: number of parameters

When λ is large in (3. 1), it approaches the steepest descent method as follows:

$$\Delta\boldsymbol{\theta}\fallingdotseq\frac{1}{\lambda}\boldsymbol{g}$$

On the other hand, when λ is small it approaches NEWTON's method as follows:

$$\boldsymbol{H}\Delta\boldsymbol{\theta}\fallingdotseq\boldsymbol{g}$$

The steepest descent method is stable but has a slow convergence, NEWTON's method has the opposite character. Therefore, first let λ be large, then become smaller step by step. It is expected to produce a good convergence. In general, the practical procedure is as follows: Let $\nu=2$. When $\Delta Y<0$, let λ be smaller as $\lambda^{\text{new}}=\lambda^{\text{old}}/\nu$ and continue the calculation. On the other hand, when $\Delta Y\geqslant 0$, let λ be larger as $\lambda^{\text{new}}=\lambda^{\text{old}}*\nu$ and try again the same iterative routine of calculation. When $\Delta Y\geqslant 0$ with the condition that λ has been made larger 10 times continuously, it is regarded that parameters have reached the solution to end the calculation. The initial value of λ (λ_0) is concerned with the diagonal components of $\boldsymbol{H}$ because they are influenced by the scaling of the parameters, they are described in the next subchapter.

The general theory of the least-squares method for the above-mentioned is as follows: The linear approximate observation equations are as follows:

$$\Delta \boldsymbol{C} = \begin{pmatrix} \Delta C_{11} \\ \vdots \\ \Delta C_{nm} \end{pmatrix} = \begin{pmatrix} \frac{\partial C_{11}}{\partial \theta_1} & \cdots & \frac{\partial C_{11}}{\partial \theta_l} \\ \vdots & & \vdots \\ \frac{\partial C_{nm}}{\partial \theta_1} & \cdots & \frac{\partial C_{nm}}{\partial \theta_l} \end{pmatrix} \begin{pmatrix} \Delta \theta_1 \\ \vdots \\ \Delta \theta_l \end{pmatrix} = \boldsymbol{A} \Delta \boldsymbol{\theta}$$

Let the left term $\Delta \boldsymbol{C}$ be $\boldsymbol{d}$, and multiply ${}^t\boldsymbol{A}$ from the left.

$$ {}^t\boldsymbol{A}\boldsymbol{d} = ({}^t\boldsymbol{A}\boldsymbol{A})\, \Delta \boldsymbol{\theta} $$

Solving this equation iteratively about $\Delta \boldsymbol{\theta}$ is GAUSS-NEWTON method. Because this method is not stable, MARQUARDT's method adds $\lambda \boldsymbol{I}$ to ${}^t\boldsymbol{A}\boldsymbol{A}$ $(=\boldsymbol{H})$ to let it be stable. Then, it becomes equal to (3. 1).

2. Scaling of parameters

It is necessary for MARQUARDT's method to scale parameters, because it is similar to the steepest descent method at the first times of the iteration. Scaling of the parameters is defined as follows:

$$\boldsymbol{\theta}^* = \boldsymbol{S}\boldsymbol{\theta}, \quad \Delta \boldsymbol{\theta}^* = \boldsymbol{S} \Delta \boldsymbol{\theta}$$
$$\boldsymbol{g}^* = \boldsymbol{S}^{-1}\boldsymbol{g}, \quad \boldsymbol{H}^* = \boldsymbol{S}^{-1}\boldsymbol{H}\boldsymbol{S}^{-1}$$

$$\boldsymbol{S} = \begin{pmatrix} s_1 & & \\ & \ddots & \\ & & s_l \end{pmatrix}, \qquad \boldsymbol{S}^{-1} = \begin{pmatrix} \frac{1}{s_1} & & \\ & \ddots & \\ & & \frac{1}{s_l} \end{pmatrix}$$

MARQUARDT (1963)'s scaling $\boldsymbol{S}_1$ is used for this program.

$$\boldsymbol{S}_1 = \begin{pmatrix} \sqrt{h_{11}} & & \\ & \ddots & \\ & & \sqrt{h_{ll}} \end{pmatrix}, \quad \boldsymbol{H} = (h_{ij})$$

Then, the diagonal components of $\boldsymbol{H}^*$ are all 1. It is adequate to set $\lambda_0 = 0.01$ for good convergence.

On the other hand, $\boldsymbol{S}_2$ is used for error analysis (chapter IV) in the same way as AKAMINE (1985, 1986), because it is better to express errors as the ratio to their own parameters in general.

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

3. Calculation of a Jacobian matrix

MARQUARDT's method needs the values of a Hessian matrix. If the correct values of a Hessian matrix are given, it produces correct solutions in general. A Hessian matrix is calculated from a Jacobian matrix by (3. 4). In this program, the Jacobian matrix is calculated by the differential expressions, because the numerical differentiation by the finite difference method makes errors large and the convergence worse (NAKAGAWA and OYANAGI 1982). The multi-cohort model is the recurrence

formula (2.2). Then, the differential expressions are given as the recurrence formulas. Calculations should be made efficiently by using memory.

(2. 1) and (2. 2) are rewritten as follows:

$$\begin{cases} C_{ij}=N_{ij}g_{ij} & (3.\ 6) \\ N_{ij}=N_{i-1,j-1}\,h_{i-1,j-1} & (3.\ 7) \end{cases}$$

$$\begin{cases} g_{ij}=\dfrac{f_i s_j}{f_i s_j+M}\,(1-h_{ij}) & (3.\ 8) \\ h_{ij}=\exp\{-(f_i s_j+M)\} & (3.\ 9) \end{cases}$$

Then, the differential expressions are as follows:

$$\begin{cases} \dfrac{\partial C_{ij}}{\partial\theta}=\dfrac{\partial N_{ij}}{\partial\theta}\,g_{ij}+N_{ij}\dfrac{\partial g_{ij}}{\partial\theta} & (3.\ 10) \\ \dfrac{\partial N_{ij}}{\partial\theta}=\dfrac{\partial N_{i-1,j-1}}{\partial\theta}h_{i-1,j-1}+N_{i-1,j-1}\dfrac{\partial h_{i-1,j-1}}{\partial\theta} & (3.\ 11) \end{cases}$$

The concrete expressions for each parameter are as follows:

(1) N

$$\frac{\partial C_{ij}}{\partial N_{pq}}=\frac{\partial N_{ij}}{\partial N_{pq}}g_{ij} \qquad (3.\ 12)$$

$$\frac{\partial N_{ij}}{\partial N_{pq}}=\begin{cases} 1 & (i=1 \text{ or } j=1) \\ \dfrac{\partial N_{i-1,j-1}}{\partial N_{pq}}h_{i-1,j-1} & (i\neq 1 \text{ and } j\neq 1) \end{cases}$$

(2) f

$$\frac{\partial C_{ij}}{\partial f_p}=\begin{cases} N_{ij}\dfrac{\partial g_{ij}}{\partial f_p} & (p=i) \\ \dfrac{\partial N_{ij}}{\partial f_p}g_{ij} & (p\neq i) \end{cases} \qquad (3.\ 13)$$

$$\frac{\partial N_{ij}}{\partial f_p}=\begin{cases} N_{i-1,j-1}\dfrac{\partial h_{i-1,j-1}}{\partial f_p} & (p=i-1) \\ \dfrac{\partial N_{i-1,j-1}}{\partial f_p}h_{i-1,j-1} & (p\neq i-1) \end{cases}$$

(3) s

The standardization of s (2. 4) is not necessary for callcuation. Then, the treatment of s may be done in the same way as f, is as follows:

$$\frac{\partial C_{ij}}{\partial s_q}=\begin{cases} N_{ij}\dfrac{\partial g_{ij}}{\partial s_q} & (q=j) \\ \dfrac{\partial N_{ij}}{\partial s_q}g_{ij} & (q\neq j) \end{cases} \qquad (3.\ 14)$$

$$\frac{\partial N_{ij}}{\partial s_q}=\begin{cases} N_{i-1,j-1}\dfrac{\partial h_{i-1,j-1}}{\partial s_q} & (q=j-1) \\ \dfrac{\partial N_{i-1,j-1}}{\partial s_q}h_{i-1,j-1} & (q\neq j-1) \end{cases}$$

(4) M

$$\frac{\partial C_{ij}}{\partial M}=\frac{\partial N_{ij}}{\partial M}g_{ij}+N_{ij}\frac{\partial g_{ij}}{\partial M} \tag{3.15}$$

$$\frac{\partial N_{ij}}{\partial M}=\begin{cases} 0 & (i=1 \text{ or } j=1) \\ \dfrac{\partial N_{i-1,j-1}}{\partial M}h_{i-1,j-1}+N_{i-1,j-1}\dfrac{\partial h_{i-1,j-1}}{\partial M} & (i\neq 1 \text{ and } j\neq 1) \end{cases}$$

(5) Conclusion

These concrete expressions are as follows:

$$\begin{cases} \dfrac{\partial g_{ij}}{\partial f_i}=g_{ij}\left(\dfrac{1}{f_i}-\dfrac{s_j}{f_is_j+M}+\dfrac{s_jh_{ij}}{1-h_{ij}}\right) \\ \dfrac{\partial g_{ij}}{\partial s_j}=g_{ij}\left(\dfrac{1}{s_j}-\dfrac{f_i}{f_is_j+M}+\dfrac{f_ih_{ij}}{1-h_{ij}}\right) \\ \dfrac{\partial g_{ij}}{\partial M}=g_{ij}\left(-\dfrac{1}{f_is_j+M}+\dfrac{h_{ij}}{1-h_{ij}}\right) \end{cases} \tag{3.16}$$

$$\begin{cases} \dfrac{\partial h_{ij}}{\partial f_i}=-s_jh_{ij} \\ \dfrac{\partial h_{ij}}{\partial s_j}=-f_ih_{ij} \\ \dfrac{\partial h_{ij}}{\partial M}=-h_{ij} \end{cases} \tag{3.17}$$

It is necessary for N and M to be divided into two cases, and for f and s into three cases. It is also necessary for effective calculation to reserve the values of $\partial N_{i-1,j-1}/\partial\theta$ into memory. Then, it is better to save memory by calculating along the cohorts as $C_{ij}\longrightarrow C_{i+1,j+1}\longrightarrow C_{i+2,j+2}\longrightarrow\cdots\cdots$.

4. Program

The BASIC program for this model is rewritten from AKAMINE (1986). The main modified parts are calculations of the objective function and a Jacobian matrix. Although it is written only for the case of $n\geqq m$, it is easy to rewrite it for the case of $n<m$. It is better to set the number of iterations to 100 or more, because the number of parameters is too large to get a fast convergence. It is easy to rewrite for other objective functions according to AKAMINE (1985). In these cases, it is better for effective calculation to let CEST be the dimension CEST (I, J) to reserve of C_{ij}.

5. Example computations

(1) Data for computation

Artificial data ($n=10$, $m=5$) was used for the test computations. The data and its parameters (true values) are shown in Tables 1 and 2. Data–0 is true in column 6. Data–1 is rounded after the decimal point. Data–2 is the addition of the normal random numbers N (0,5) to data–1. The real data obtained in the fields look like data–2.

Table 1. The artificial data of catch for the tests of the program.

Data-0

year \ age	1	2	3	4	5
1	44.2398	101.250	82.4200	70.8245	47.2163
2	42.6744	79.9783	75.7490	65.0753	35.3296
3	98.3568	173.025	133.510	130.145	71.4247
4	42.2674	170.619	119.823	94.5508	56.5991
5	43.1970	145.625	234.366	164.149	80.9196
6	74.2305	59.6709	77.9712	125.974	52.6113
7	147.921	204.331	65.5569	86.2443	86.5561
8	40.2967	98.621	54.6682	18.4043	14.6962
9	29.1285	198.696	205.226	120.077	26.2489
10	76.7537	90.2756	248.974	261.420	96.0956

Data-1

year \ age	1	2	3	4	5
1	44	101	82	71	47
2	43	80	76	65	35
3	98	173	134	130	71
4	42	171	120	95	57
5	43	146	234	164	81
6	74	60	78	126	53
7	148	204	66	86	87
8	40	99	55	18	15
9	29	199	205	120	26
10	77	90	249	261	96

Data-2

year \ age	1	2	3	4	5
1	48	97	85	61	54
2	39	80	86	71	44
3	99	172	132	132	65
4	44	175	122	96	57
5	44	139	234	167	80
6	72	61	80	120	56
7	146	203	65	83	92
8	37	104	54	19	15
9	34	197	190	127	22
10	67	94	239	271	101

Tabl 2. Results of the program for the data of Table 1.

		Initial value	Solution			
			True value	Data-0	Data-1	Data-2
N	1	1000	1000	999.934	1122.74	2123.26
	2	1000	1200	1199.92	1338.11	2792.58
	3	1000	1500	1499.91	1670.59	3979.94
	4	1000	800	799.952	893.516	2412.39
	5	1000	500	499.969	563.040	1937.49
	6	1000	1300	1299.92	1469.56	6261.88
	7	1000	2000	1999.87	2276.33	11601.7
	8	1000	1800	1799.88	2051.34	12310.0
	9	1000	600	599.964	680.584	5215.54
	10	1000	1100	1099.94	1263.45	9621.20
N'	2	500	800	799.961	880.743	1439.10
	3	500	500	499.981	543.271	777.947
	4	500	300	299.992	325.279	377.692
	5	500	200	199.998	215.857	282.863
f	1	1	1.0	1.00002	.919873	.719316
	2	1	.8	.800017	.742029	.591020
	3	1	1.5	1.50003	1.39195	.924121
	4	1	1.2	1.20002	1.11576	.658857
	5	1	2.0	2.00004	1.84382	.882425
	6	1	1.3	1.30003	1.19101	.445178
	7	1	1.7	1.70004	1.54503	.472433
	8	1	.5	.500010	.455702	.124051
	9	1	1.1	1.10002	1.00259	.223946
	10	1	1.6	1.60003	1.44725	.260437
s	1	.2	.05	.0500017	.0486628	.0315890
	2	.2	.15	.150003	.148764	.117442
	3	.2	.2	.200002	.200248	.182250
	4	.2	.3	.299999	.301091	.317780
	5	.2	.3	.299994	.301234	.350940
M		.3	.2	.199980	.226127	.335011
Y				.000000293	1.87194	562.192
Times of iteration				37	39	95

(2) Results of computations

The initial values of the parameters for conputation and each solution are shown in Table 2. In data-0, it required 37 iterations to reach a true solution with $Y=0$. The changes of Y, λ, and M are shown in Fig. 2. In data-1, it required 39 iterations to reach a solution. The changes of λ and all parameters shown in Fig. 3. They shows that data-0 and -1 are similar to each other. It must be remarked that a small difference of data makes a large difference to the solution. In data-2, it required 95

iterations to reach a solution. The changes of λ and M are shown in Fig. 4. It reached a much different solution from the true solution, because the difference of data-2 from data-0 was too large.

Data-1 is regarded as the addition of data-0 and an error $-0.5\sim0.5$. This error is considered to follow the uniform distribution, so the variance of error becomes 1/12. Because the number of data is 50, the total variance of error becomes $50/12=4.17$. It coincides with $Y=4.21$ obtained by inputting the true values into data-1. Data-2 is regarded as the addition of data-1 and $N(0,5)$ which is independent of the above error. Then, the variance of error becomes 25, and the total variance of error becomes $50*25=1250$. $Y=1335$ is obtained by inputting the true values into data-2. In the both cases, MARQUARDT's method obtained a better solution with a smaller Y (Table 2).

(3) Computer

The computer used for these calculations was a PC-9801F (NEC, 16bits, 8MHz). The N_{88}-BASIC of this machine has bugs in the "PRINT" statement and the "SQR" function, among others (NISHIMI 1986), and all functions are single precision. It is also regrettable that defect of the "block IF" statement, etc. makes programs difficult to understand. But these faults are negligible for application of this subject.

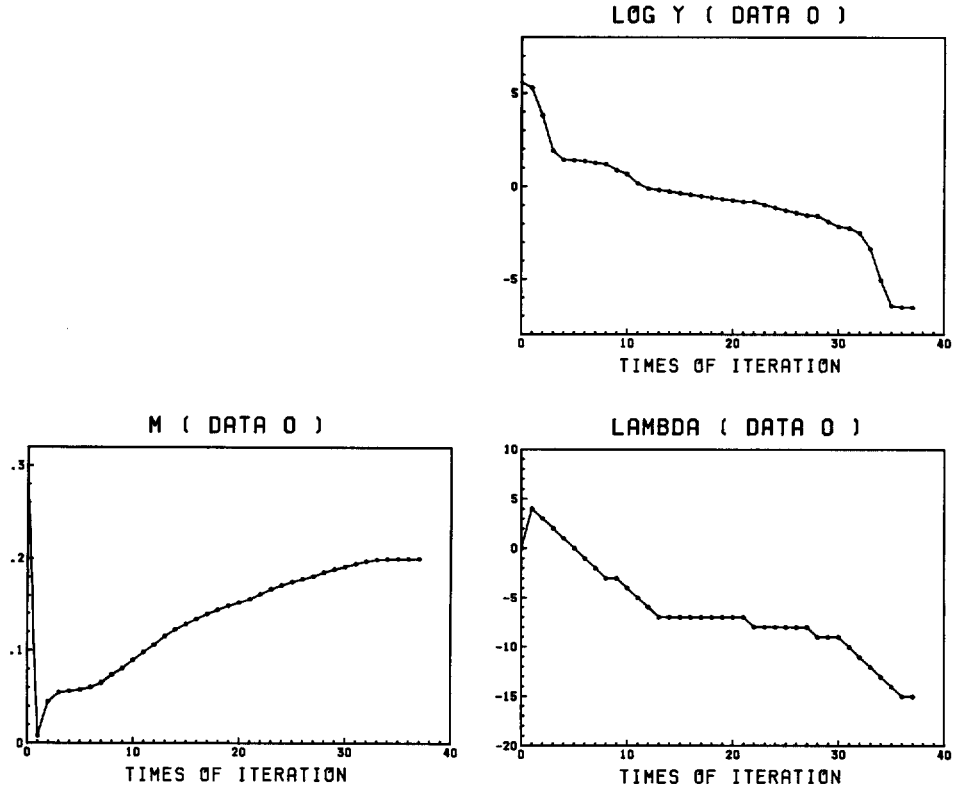


Fig. 2. Convergence of data-0

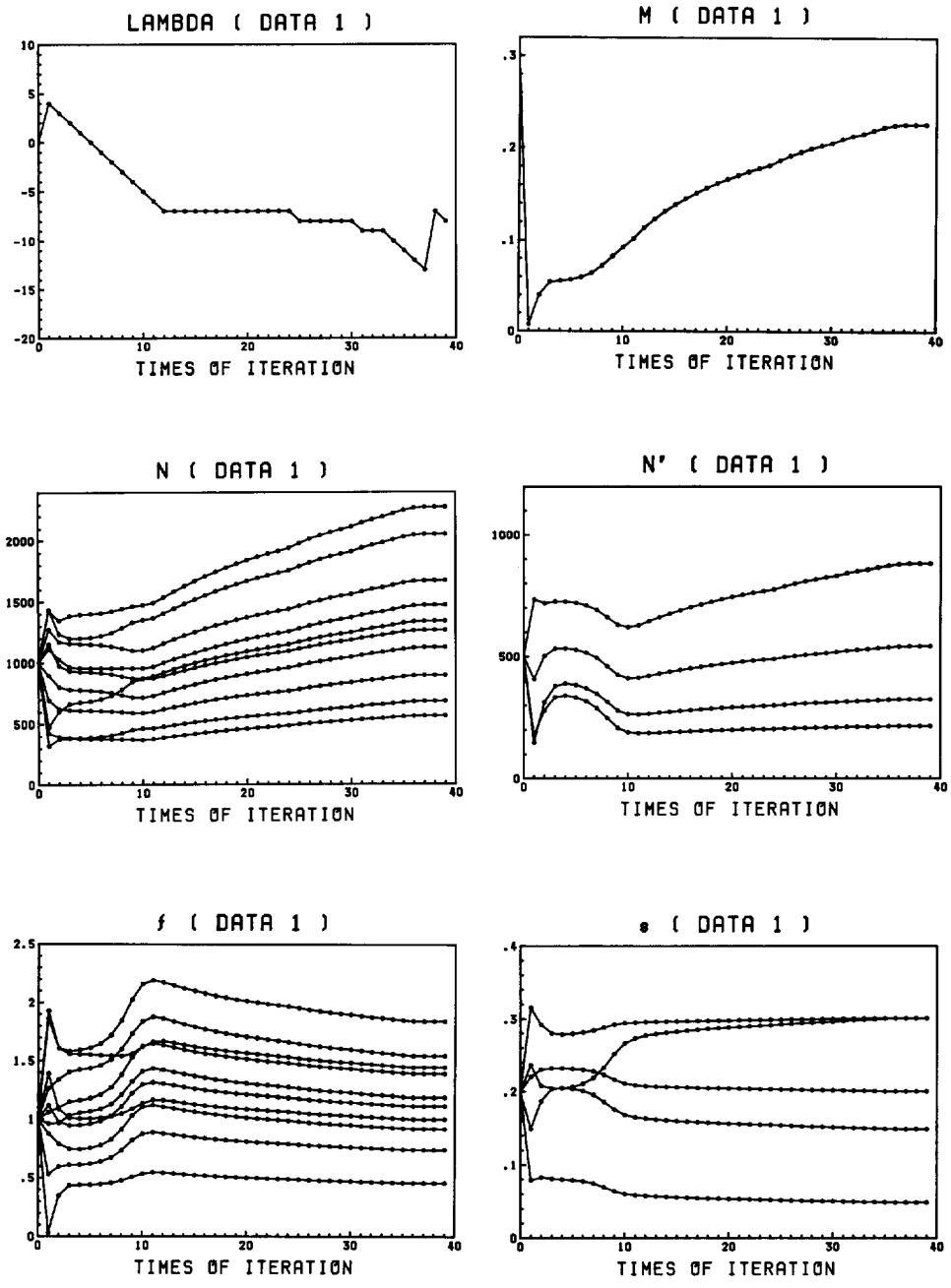


Fig. 3. Convergence of data-1

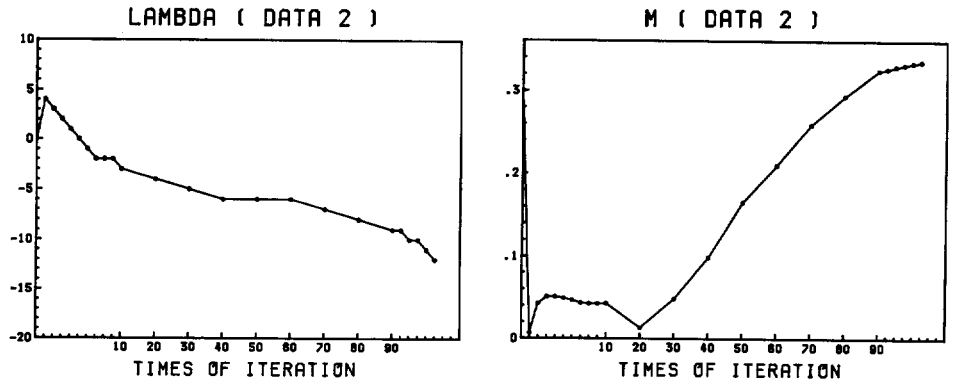


Fig. 4. Convergence of data-2.

(4) The computing time

One iteration required about 4 minutes by the PC-9801F. Then, the total time is about 160 minutes for data-0 and -1, and 380 minutes for data-2. Real data obtained in the fields may require more time, so it is better to use faster computers than this.

The change of λ in Figs. 2~4 shows little changes in the middle part. Namely it is difficult to search a smaller Y in the part between the steepest descent method and NEWTON's method. This is caused by the fact that the non-linearity is very strong in this model. This problem is considered synthetically in the next chapter.

IV. Error analysis

The procedures of error analysis are according to AKAMINE (1985, 1986). The values of the Hessian matrix $\boldsymbol{H}$ of the solution was used to estimate the state of the neighborhood. Data-0 and -2 are not adequate for this subject because the former's solution became $Y=0$ and the latter's solution was very different from the true values. Therefore, data-1 is used in this chapter.

For a test of the confidence area, the following expression applied in the linear case is used according to DRAPER and SMITH (1966).

$$\frac{\Delta Y}{Y_0}=\frac{p}{m-p}F(p, m-p, 1-\alpha) \tag{4.1}$$

m : number of data
p : number of parameters
α : confidence level

In this case, $m=50$, $p=29$, then $F(29, 21, 0.05) \fallingdotseq F(30, 20, 0.05)=2.039$, $Y_0=1.87194$, then $\Delta Y=5.3$ by (4. 1). Therefore, $\Delta Y<5.3$ is used as the condition of the confidence area.

1. Calculations of $\boldsymbol{H}$ and $\boldsymbol{H}^{-1}$

First, consider the values of $\boldsymbol{H}$. It is impossible to use the HESSIAN (I, J) values

directly because there is the restriction (2. 4). When values of $\boldsymbol{H}$ can not be obtained directly, general methods to obtain them are as follows:

(1) By fundamental vectors

In the neighborhood of the solution there is the following approximate expression:

$$\Delta Y \fallingdotseq -\frac{1}{2}\, {}^t\Delta\boldsymbol{\theta}\boldsymbol{H}\Delta\boldsymbol{\theta} \tag{4. 2}$$

Where the fundamenal vector is defined as follows:

$$\boldsymbol{e}_i = {}^t(\underset{1}{0},\cdots\cdots,0,\underset{i}{1},0,\cdots\cdots,\underset{l}{0})$$

Then, (4. 2) leads to the following relationship:

$$\Delta\boldsymbol{\theta} = \frac{\boldsymbol{e}_i}{s_i}, \qquad \Delta Y = -\frac{1}{2}\,\frac{h_{ii}}{s_i^2} \tag{4. 3}$$

$$\Delta\boldsymbol{\theta} = \frac{\boldsymbol{e}_i}{s_i} + \frac{\boldsymbol{e}_j}{s_j}, \qquad \Delta Y = -\frac{1}{2}\left(\frac{h_{ii}}{s_i^2} + \frac{h_{jj}}{s_j^2} + \frac{2h_{ij}}{s_i s_j}\right) \tag{4. 4}$$

These expressions provides all values of $\boldsymbol{H}$. For the diagonal elements ΔY is provided as the mean of $(+s_i)$ and $(-s_i)$, and for the other elements it is provided as the mean of $(+s_i,+s_j)$, $(+s_i,-s_j)$, $(-s_i,+s_j)$ and $(-s_i,-s_j)$. With $\boldsymbol{S}_2$ scaling it is adequate to move each parameter about 1%. But this meshod requires a large amount of computation and a high precision because of cancellings by the subtractions in the calculations.

(2) By MARQUARDT's method

The correct vector $\Delta\boldsymbol{\theta}$ of MARQUARDT's method in the last part provides the minimum eigenvector of $\boldsymbol{H}$. This was proved by NAKAGAWA and OYANAGI (1982) as follows:

The eigenvalue resolution of $\boldsymbol{H}$ is

$${}^t\boldsymbol{P}\boldsymbol{H}\boldsymbol{P} = \begin{pmatrix} \beta_1 & & \\ & \ddots & \\ & & \beta_l \end{pmatrix}, \qquad \boldsymbol{P} = (\boldsymbol{v}_1,\cdots\cdots,\boldsymbol{v}_l), \quad {}^t\boldsymbol{P} = \boldsymbol{P}^{-1}$$

β_i : eigenvalue

$\boldsymbol{v}_i$: eigenvector

Then, it leads to :

$${}^t\boldsymbol{P}(\boldsymbol{H}+\lambda\boldsymbol{I})\,\boldsymbol{P} = {}^t\boldsymbol{P}\boldsymbol{H}\boldsymbol{P} + \lambda\boldsymbol{I} = \begin{pmatrix} \beta_1+\lambda & & \\ & \ddots & \\ & & \beta_l+\lambda \end{pmatrix}$$

Then, it leads to :

$$(\boldsymbol{H}+\lambda\boldsymbol{I})^{-1} = \boldsymbol{P}\begin{pmatrix} \frac{1}{\beta_1+\lambda} & & \\ & \ddots & \\ & & \frac{1}{\beta_l+\lambda} \end{pmatrix}{}^t\boldsymbol{P}$$

$$=\frac{1}{\beta_1+\lambda}\ \boldsymbol{v}_1{}^t\boldsymbol{v}_1+\cdots\cdots\cdots+\frac{1}{\beta_l+\lambda}\boldsymbol{v}_l{}^t\boldsymbol{v}_l$$

(spectral resolution)

Let this expression be substituted in (3.1). It becomes

$$\Delta\boldsymbol{\theta}=(\boldsymbol{H}+\lambda\boldsymbol{I})^{-1}\boldsymbol{g}$$

$$=\boldsymbol{v}_1\left(\frac{1}{\beta_1+\lambda}{}^t\boldsymbol{v}_1\boldsymbol{g}\right)+\cdots\cdots\cdots+\boldsymbol{v}_l\left(\frac{1}{\beta_l+\lambda}{}^t\boldsymbol{v}_l\boldsymbol{g}\right)$$

Let the following relations assume.

$$\lambda,\ \beta_l \ll \beta_{l-1} \leqslant \beta_{l-2} \leqslant \cdots\cdots\cdots \leqslant \beta_1$$

Then it leads to :

$$\Delta\boldsymbol{\theta} \fallingdotseq \boldsymbol{v}_l\left(\frac{1}{\beta_l+\lambda}{}^t\boldsymbol{v}_l\boldsymbol{g}\right)+\sum_{i=1}^{l-1}\boldsymbol{v}_i\left(\frac{1}{\beta_i}{}^t\boldsymbol{v}_i\boldsymbol{g}\right) \qquad \frac{1}{\beta_l+\lambda} \gg \frac{1}{\beta_i}$$

Therefore, $\Delta\boldsymbol{\theta}$ becomes parallel with $\boldsymbol{v}_l$, where $\boldsymbol{v}_l$ is the minimum eigenvector of $\boldsymbol{S}_1^{-1}\boldsymbol{H}\boldsymbol{S}_1^{-1}$ not $\boldsymbol{H}$.

(3) The method in this paper

$\boldsymbol{H}$ was calculated from HESSIAN (I, J) by (2. 4) in this paper, s_m is eliminated by (2. 4) as follows:

$$s_m=1-\sum_{j=1}^{m-1}s_j,\ \Delta s_m=-\sum_{j=1}^{m-1}\Delta s_j \tag{4. 5}$$

In the neighborhood of the solution there is the following expression:

$$\boldsymbol{H} \fallingdotseq \left(\frac{\partial^2 Y}{\partial y^2}\ \frac{\partial y}{\partial\theta_i}\ \frac{\partial y}{\partial\theta_j}\right) \tag{3. 4}$$

Let $y=z\,(s)$. Then, it leads to:

$$\frac{\partial y}{\partial s_j}=\frac{\partial z}{\partial s_j}+\frac{\partial z}{\partial s_m}\ \frac{\partial s_m}{\partial s_j}=\frac{\partial z}{\partial s_j}-\frac{\partial z}{\partial s_m} \qquad \because \frac{\partial s_m}{\partial s_j}=-1 \tag{4. 6}$$

On the other hand, the linear approximation leads to the same relationship as follows:

$$\Delta y=\sum_{j=1}^{m-1}\frac{\partial z}{\partial s_j}\ \Delta s_j+\frac{\partial z}{\partial s_m}\ \Delta s_m=\sum_{j=1}^{m-1}\left(\frac{\partial z}{\partial s_j}-\frac{\partial z}{\partial s_m}\right)\Delta s_j \tag{4. 6'}$$

Then the real expression of $s_j\ (j=1\sim m-1)$ becomes as follows:

$$\frac{\partial y}{\partial\theta_i}\ \frac{\partial y}{\partial s_j}=\frac{\partial y}{\partial\theta_i}\ \frac{\partial y}{\partial s_j}-\frac{\partial y}{\partial\theta_i}\ \frac{\partial y}{\partial s_m}$$

$$\frac{\partial y}{\partial s_j}\ \frac{\partial y}{\partial s_k}=\frac{\partial y}{\partial s_j}\ \frac{\partial y}{\partial s_k}-\frac{\partial y}{\partial s_j}\ \frac{\partial y}{\partial s_m}-\left(\frac{\partial y}{\partial s_k}\ \frac{\partial y}{\partial s_m}-\frac{\partial y}{\partial s_m}\ \frac{\partial y}{\partial s_m}\right)$$

The real procedures to exclude s_m from $\boldsymbol{H}$ including s_m: HESSIAN (I, J) is as follows:

① Subtract the s_m column from the s_j ($j=1\sim m-1$) columns, and exclude the s_m column.

② Subtract the s_m row from the s_j ($j=1\sim m-1$) rows, and exclude the s_m row.

These procedures do not include scaling of parameters, so that must be done before scaling. Therefore the regular $\boldsymbol{H}$ is obtained so it is possible to calculate $\boldsymbol{H}^{-1}$ ($\sim \boldsymbol{V}$).

Input the values of the solution as the initial values, run the program and stop at line 4120. Do the procedure of ① and ②. Therefore values of $\boldsymbol{H}/2$ are in HESSIAN (I, J). In this paper, the values of $\boldsymbol{H}_2^*/2=\boldsymbol{S}_2^{-1}(\boldsymbol{H}/2)\boldsymbol{S}_2^{-1}$ are used for the analysis. Next, calculate the following :

HESSIAN (I, J) * [scale θ_i] * [scale θ_j]

[scale θ] : solution of data-1

The values of $\boldsymbol{H}_2^*/2$ of data-1 is shown in Table 3.

The values of s_m about the covariance matrix $\boldsymbol{V}=<\Delta\theta_i\Delta\theta_j>\sim\boldsymbol{H}^{-1}$ are obtained by (4.5) as the following expression:

$$<\Delta s_m \Delta\theta>=-\sum_{j=1}^{m-1}<\Delta s_j\Delta\theta> \tag{4.7}$$

$<\ >$: expected value

The values of Table 3 are already scaled, so the real expression becomes as follows:

$$<\Delta s_m \Delta\theta>=-\frac{\sum_{}^{m-1}[\text{scale } s_j]<\Delta s_j\Delta\theta>}{[\text{scale } s_m]} \tag{4.8}$$

The procedures are inverse to ① and ② as follows:

③ Add the s_j ($j=1\sim m-1$) columns multiplied by s_j, divide it by ($-s_m$), let it be the s_m column.

④ Add the s_j ($j=1\sim m-1$) rows multiplied by s_j, divide it by ($-s_m$), let it be the s_m row.

These procedures are the same way as AKAMINE (1985). This method depends on the property of $\boldsymbol{H}$ and $\boldsymbol{H}^{-1}$ to be expressed as the product of the two components as $\boldsymbol{A}=(\sum a_i a_j)$.

The inverse matrix is calculated by the BASIC program of AKAMINE et al. (1982). The values of $\boldsymbol{H}^{-1}$ ($\sim\boldsymbol{V}$) and the correlation matrix $\boldsymbol{R}$ are shown in Table 4.

The values of $\boldsymbol{H}_2^*$ becomes 10000 (100^2) times of the values obtained by the method of fundamental vectors ($\boldsymbol{e}_i$, $\boldsymbol{e}_j$) to move each parameter by about 1%. Therefore the values of $\boldsymbol{H}_2^*$ are regarded as the values of ΔY by linear approximation to move each parameter by about 100%. Then multiply the values of Table 3 by 1/10000. For example, if M moves by +1% of its value (with other parameters to be fixed), ΔY becomes about 21 and over the confidence area. But in the real cases the confidence area becomes larger because the correlations of parameters is very

Table 3-a. Hessian matrix of the solution (data-1).
Elements of s_5 are excluded by oparations ①, ②

		N										N'				f
		1	2	3	4	5	6	7	8	9	10	2	3	4	5	1
N	1	41878	0	0	0	0	0	0	0	0	0	0	0	0	0	109
	2		75883	0	0	0	0	0	0	0	0	0	0	0	0	0
	3			117046	0	0	0	0	0	0	0	0	0	0	0	0
	4				36785	0	0	0	0	0	0	0	0	0	0	0
	5					10805	0	0	0	0	0	0	0	0	0	0
	6						73733	0	0	0	0	0	0	0	0	0
	7							141851	0	0	0	0	0	0	0	0
	8								103202	0	0	0	0	0	0	0
	9									8941	0	0	0	0	0	0
	10										5929	0	0	0	0	0
N'	2											36126	0	0	0	6013
	3												15990	0	0	4455
	4													6266	0	4029
	5														2209	1929
f	1															22911
	2															
	3															
	4															
	5															
	6															
	7															
	8															
	9															
	10															
s	1															
	2															
	3															
	4															
	5															
M																

Table 3-b. Continued.

		f									*s*					*M*
		2	3	4	5	6	7	8	9	10	1	2	3	4	5	
N	1	2409	11288	5420	4986	0	0	0	0	0	−697	−53	7973	436	0	−25940
	2	−900	17912	6299	18849	2350	0	0	0	0	−1279	16751	4737	16500	0	−41870
	3	0	2027	13987	37032	10591	5988	0	0	0	1059	11030	33052	4606	0	−61057
	4	0	0	−157	14775	3593	5820	206	0	0	−190	14673	3456	5614	0	−16613
	5	0	0	0	996	2340	3391	225	592	0	900	2048	2998	−366	0	−4746
	6	0	0	0	0	1391	31154	727	9577	7426	191	27487	−4209	2155	0	−36518
	7	0	0	0	0	0	12032	1997	24468	54931	12032	1997	24468	54931	0	−82155
	8	0	0	0	0	0	0	−647	27593	53763	−647	27593	53763	0	0	−47765
	9	0	0	0	0	0	0	0	428	7290	428	7290	0	0	0	−2771
	10	0	0	0	0	0	0	0	0	5731	5731	0	0	0	0	−637
N′	2	2369	12401	2735	0	0	0	0	0	0	−442	4663	551	9667	0	−14880
	3	2644	4101	0	0	0	0	0	0	0	−662	−2025	1729	−1455	0	−4893
	4	1124	0	0	0	0	0	0	0	0	−182	−555	−747	2905	0	−934
	5	0	0	0	0	0	0	0	0	0	−312	−952	−1282	−1928	0	−229
f	1	−1231	−2958	−617	−223	0	0	0	0	0	1933	9312	5102	1471	0	1402
	2	18113	−4652	−1232	−1231	−85	0	0	0	0	1887	5695	4105	1150	0	872
	3		65768	−4909	−7801	−1204	−406	0	0	0	8360	22504	8498	4793	0	−13025
	4			51443	−12836	−2478	−1310	−11	0	0	527	24388	5415	666	0	−7672
	5				80429	−6410	−4111	−77	−53	0	−2447	6369	34271	8978	0	−40548
	6					27956	−5942	−131	−660	−430	4343	−1476	692	9587	0	−10595
	7						68472	−483	−4224	−5837	17704	29994	−4161	−6565	0	−15775
	8							14761	−3226	−5594	1506	9076	354	−3816	0	2806
	9								85809	−21644	−1957	32174	29416	2117	0	−25117
	10									108999	−1536	−9841	30955	36051	0	−83160
s	1										52335	−2314	−2268	271	0	16098
	2											186465	−6169	6255	0	−7121
	3												195570	4431	0	−50488
	4													161121	0	−60909
	5														0	0
M																208364

Table 4-a. The covariance matrix and the correlation matrix of the solution (data-1).
Elements of s_5 are added by operations ③, ④.

		N										N'				f
		1	2	3	4	5	6	7	8	9	10	2	3	4	5	1
N	1	216856	207073	203323	202035	207355	213522	216679	209722	198133	195721	170921	137789	104200	72191	−98031
	2	100	197784	194207	193001	198112	204043	207116	200537	189537	187303	163196	131514	99391	68753	−93483
	3	100	100	190746	189593	194673	200562	203655	197279	186571	184477	160263	129187	97684	67677	−91911
	4	100	100	100	188543	193647	199594	202772	196554	186048	184103	159284	128453	97208	67504	−91512
	5	100	100	100	100	199133	205278	208684	202460	191858	190054	163559	132023	100081	69828	−94313
	6	100	100	100	100	100	211867	215530	209307	198599	196960	168465	136046	103224	72210	−97336
	7	99	99	100	100	100	100	219493	213413	202804	201422	170924	137989	104643	73133	−98672
	8	99	99	99	99	100	100	100	207855	197905	196912	165402	133488	101172	70650	−95405
	9	98	98	98	99	99	99	100	100	189026	188390	156274	126144	95644	66909	−90248
	10	97	97	97	98	98	99	99	99	100	188448	154372	124617	94503	66184	−89212
N'	2	100	100	100	100	100	100	99	99	98	97	135111	109410	83459	59095	−78851
	3	99	99	99	99	99	99	98	98	97	96	100	89426	69206	50836	−65863
	4	95	95	95	95	95	95	95	94	94	93	97	99	55199	42923	−53035
	5	79	79	79	80	80	80	80	79	79	78	82	87	94	38167	−42471
f	1	−93	−93	−93	−93	−93	−93	−93	−92	−92	−91	−95	−97	−100	−96	51402
	2	−92	−92	−92	−93	−93	−93	−93	−92	−92	−91	−94	−97	−99	−96	100
	3	−92	−92	−92	−92	−93	−93	−92	−92	−91	−91	−94	−97	−99	−96	100
	4	−92	−92	−92	−92	−92	−93	−92	−92	−92	−91	−93	−96	−99	−96	100
	5	−91	−91	−91	−92	−92	−93	−92	−92	−92	−92	−93	−96	−99	−96	99
	6	−91	−91	−91	−92	−92	−93	−93	−93	−93	−93	−93	−95	−98	−95	99
	7	−90	−90	−90	−91	−92	−92	−93	−93	−93	−93	−92	−94	−97	−94	98
	8	−88	−88	−88	−89	−90	−91	−91	−92	−92	−93	−90	−92	−95	−93	96
	9	−83	−83	−84	−85	−86	−87	−88	−89	−90	−91	−85	−88	−91	−91	92
	10	−77	−77	−78	−79	−80	−82	−83	−84	−86	−87	−79	−82	−85	−86	87
s	1	−94	−94	−94	−94	−94	−93	−93	−93	−92	−91	−92	−89	−80	−55	75
	2	−90	−91	−91	−90	−90	−89	−89	−89	−88	−87	−88	−84	−73	−46	68
	3	−82	−82	−82	−81	−81	−80	−80	−79	−78	−77	−79	−73	−61	−31	55
	4	98	98	98	98	98	98	98	98	97	96	97	95	90	71	−87
	5	87	87	87	86	86	85	85	85	84	83	84	79	68	39	−62
M		99	99	99	99	99	99	98	98	97	96	99	97	91	72	−88

Table 4-b. Continued.

		f									*s*					*M*
		2	3	4	5	6	7	8	9	10	1	2	3	4	5	
N	1	−89645	−84976	−83842	−87524	−94269	−98939	−92842	−80941	−77410	−95490	−54066	−21664	9599	46933	266176
	2	−85491	−81054	−79993	−83541	−90019	−94528	−88758	−77455	−74184	−91330	−51725	−20731	9174	44910	254265
	3	−84072	−79731	−78726	−82264	−88693	−93202	−87601	−76579	−73515	−89604	−50714	−20299	9012	44006	249517
	4	−83733	−79446	−78490	−82086	−88567	−93168	−87691	−76845	−74009	−88921	−50283	−20081	8958	43593	247725
	5	−86334	−81967	−81047	−84843	−91639	−96531	−91031	−80041	−77419	−90988	−51362	−20438	9196	44459	253851
	6	−89139	−84684	−83806	−87830	−94966	−100199	−94684	−83539	−81166	−93578	−52761	−20934	9476	45617	261125
	7	−90402	−85940	−85138	−89348	−96747	−102273	−96883	−85797	−83791	−95145	−53622	−21253	9636	46348	264932
	8	−87461	−83217	−82552	−86792	−94161	−99784	−94814	−84372	−82940	−92287	−51985	−20570	9356	44903	256329
	9	−82800	−78876	−78381	−82597	−89815	−95466	−91059	−81527	−80794	−87224	−49064	−19337	8865	42315	241869
	10	−81909	−78111	−77748	−82098	−89457	−95350	−91273	−82163	−82001	−86256	−48443	−19032	8780	41733	238689
N′	2	−72135	−68431	−67531	−70503	−75899	−79640	−74784	−65353	−62604	−73918	−41599	−16468	7515	35921	208600
	3	−60305	−57277	−56550	−59051	−63520	−66628	−62643	−54973	−52815	−57615	−32046	−12389	5990	27382	166385
	4	−48614	−46270	−45772	−47761	−51308	−53793	−50687	−44795	−43251	−40810	−22150	−8124	4426	18508	123333
	5	−39033	−37339	−36975	−38674	−41459	−43447	−41157	−36949	−36102	−23368	−11637	−3406	2903	8884	80987
f	1	47127	44914	44409	46412	49847	52270	49326	43768	42405	37124	19875	7063	−4124	−16386	−114860
	2	43313	41250	40806	42674	45858	48125	45466	40425	39266	33818	18067	6387	−3770	−14863	−104869
	3	100	39371	38965	40787	43867	46088	43615	38894	37920	31868	16970	5950	−3571	−13915	−99174
	4	100	100	38638	40482	43597	45886	43526	38960	38175	31399	16686	5818	−3528	−13654	−97716
	5	99	100	100	42520	45866	48387	46034	41401	40826	32785	17395	6032	−3693	−14206	−101884
	6	99	99	100	100	49623	52451	50048	45211	44861	35489	18840	6533	−3996	−15385	−109757
	7	98	98	99	99	100	55684	53336	48469	48480	37373	19816	6845	−4209	−16166	−115105
	8	96	97	98	98	99	100	51435	47046	47523	34940	18430	6278	−3962	−14960	−107645
	9	93	94	95	96	97	98	99	43629	44714	29950	15596	5124	−3462	−12486	−93030
	10	87	88	90	92	93	95	97	99	46662	28366	14596	4619	−3336	−11526	−88249
s	1	75	74	73	73	73	73	71	66	60	47323	27749	11876	−4421	−24824	−121885
	2	68	67	66	66	66	65	63	58	53	99	16459	7174	−2536	−14846	−69925
	3	54	53	52	51	52	51	49	43	38	96	98	3236	−1040	−6574	−28743
	4	−86	−86	−85	−85	−85	−85	−83	−79	−73	−97	−94	−87	442	2216	11940
	5	−61	−60	−60	−59	−59	−59	−57	−51	−46	−98	−100	−99	91	13497	61395
M		−88	−87	−86	−86	−86	−85	−82	−77	−71	−97	−95	−88	99	92	331121

Table 5. Vectors of the solution (data-1).

$\boldsymbol{v}$: eigenvectors. $\boldsymbol{u}$: vectors from the solution to ith iteration times of MARQUARDT's method.

		$\boldsymbol{v}$		$\boldsymbol{u}$			
		1	2	10	20	30	35
N	1	.26493	.11996	−.25166	−.25511	−.26290	−.26701
	2	.25313	.11579	−.24153	−.24420	−.25135	−.25519
	3	.24884	.10449	−.23650	−.23983	−.24691	−.25056
	4	.24768	.09049	−.23380	−.23830	−.24548	−.24918
	5	.25483	.06965	−.23726	−.24429	−.25202	−.25594
	6	.26303	.05315	−.24239	−.25158	−.25974	−.26385
	7	.26755	.04405	−.24522	−.25574	−.26406	−.26826
	8	.25979	.02888	−.23734	−.24860	−.25632	−.26009
	9	.24651	.00265	−.22221	−.23547	−.24282	−.24637
	10	.24448	−.01846	−.21697	−.23287	−.24039	−.24395
N'	2	.20945	.03927	−.20483	−.20566	−.20878	−.21016
	3	.16975	−.04939	−.16840	−.16964	−.16952	−.16909
	4	.12966	−.15122	−.12752	−.13177	−.12920	−.12735
	5	.09222	−.31048	−.08166	−.09490	−.09043	−.08752
f	1	−.12265	.19638	.14138	.13847	.12612	.11971
	2	−.11236	.18826	.12686	.12586	.11516	.10955
	3	−.10680	.19105	.11763	.11879	.10908	.10399
	4	−.10571	.19784	.11423	.11705	.10771	.10266
	5	−.11078	.21659	.11801	.12261	.11276	.10738
	6	−.11972	.23752	.12743	.13312	.12200	.11605
	7	−.12628	.26038	.13332	.14102	.12873	.12219
	8	−.11938	.26972	.12086	.13231	.12114	.11511
	9	−.10549	.28426	.09613	.11453	.10585	.10092
	10	−.10258	.32079	.08273	.10951	.10188	.09738
s	1	−.11449	−.26265	.16247	.12642	.12108	.11810
	2	−.06424	−.19174	.09017	.06909	.06775	.06696
	3	−.02525	−.11138	.03737	.02666	.02679	.02687
	4	.01170	.01178	−.01541	−.01298	−.01226	−.01179
	5	(.05531)	(.19939)	(−.08022)	(−.05929)	(−.05857)	(−.05822)
M		.32254	.35146	−.41082	−.35210	−.33604	−.32759
λ (%)		3.0574 (95.89)	.10121 (3.174)				
$1/\lambda$		.32708	9.8804				

strong. In data-1, the convergence of Y indicates that after the first 10 times of iteration it enters the confidence area (Fig. 3). The fact that $M=0.092$ at that time shows that M is able to move by about 50% in the confidence area.

2. Calculations of eigenvectors

The confidence area and correlations of the parameters are evaluated by the diagonal elements of $\boldsymbol{H}^{-1}$ and the values of $\boldsymbol{R}$. The eigenvectors ($\boldsymbol{v}_i$) of $\boldsymbol{H}^{-1}$ were calculated to clear those characters. The BASIC programs (HAUSEHOLDER transform, bisection method, WIELANDT's inverse iteration) of GEN and IDA (1983) were used for this calculation. The values of $\boldsymbol{v}_1$, $\boldsymbol{v}_2$ are shown in Table 5. Items after $\boldsymbol{v}_3$ are omitted because their errors are too large to trust.

The conclusions obtained from the values of $\boldsymbol{v}_1$ are as follows:

1) The values of s are more stable than that of the other parameters, but they may be unstable when m is large.
2) The left part of s (i is small) and the right part of s (i is large) have a negative correlation. It seems to be caused by the restrictive condition $\sum s=1$.
3) The interior part of N and f have a high positive correlation.
4) N, M and the right parts of s have a high positive correlation. On the other hand, f and the left part of s have a high positive correlation too. These two sets have a high negative correlation with each other.

These conclusions are interpreted as follows: When virgin stocks increase, the natural mortality coefficient increases to restrain the change of stocks. The fishing mortality coefficient decreases to restrain the change of the total mortality coefficient. But the natural mortality coefficient is larger than the fishing mortality coefficient, so the total mortality coefficient increases to enlarge the change of stocks. Therefore the fishing ratio of old fish increases and that of young decreases to restrain the decrease of the catches of old fish and the increase of that of young fish. When virgin stocks decrease, all parameters change in the opposite direction to the above. Therefore the objective function (2.6) does not change so much. Although other objective functions may show another conclusion, these are considered general properties of the multi-cohort model (2.1, 2.2, 2.3).

The changes of ΔY on the eigenvectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are shown in Fig. 5. The scattering in $\Delta Y \leqslant 0.0001$ is considered to be caused by the cancelling error. The range of the following approximate expression is $k \leqslant 0.01$.

$$\Delta Y \fallingdotseq \frac{k^2}{2\lambda_i}, \qquad k \ : \ \text{vector length} \tag{4.9}$$

This expression is not so effective to estimate the confidence area in this case because the 95% confidence area is about $|k| < 0.1$.

The changes of ΔY on the vector $\boldsymbol{u}_i$ that is from the solution (39th interation) to the ith iteration are checked. These of $i=10, 20, 30$ are shown in Fig. 5. A typical illustration of the response surface is shown in Fig. 6. Generally the shape of the

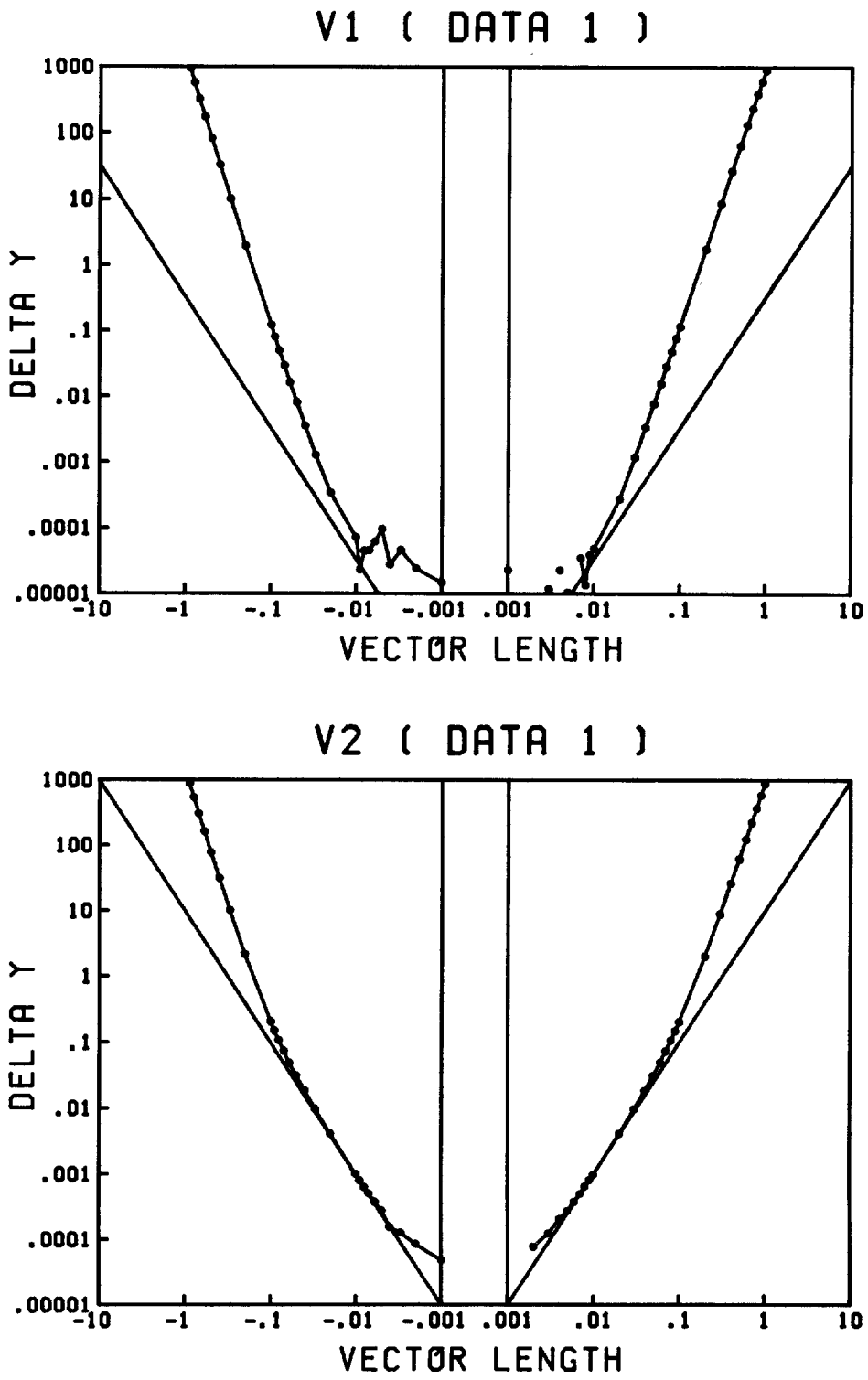
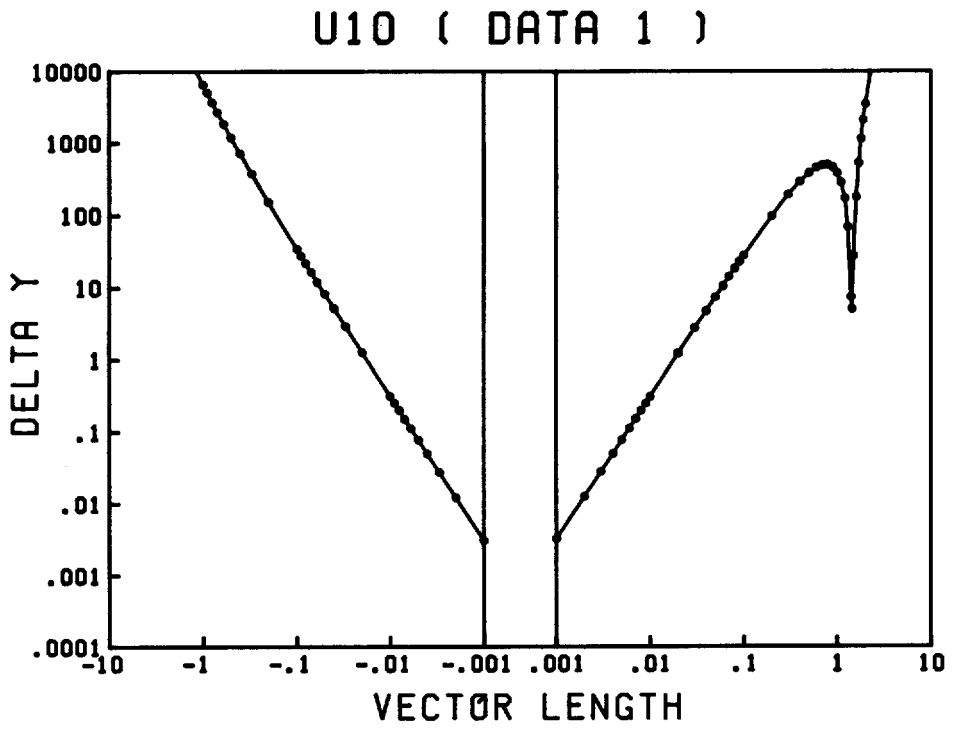


Fig. 5-a. ΔY on the vectors of Table 5.



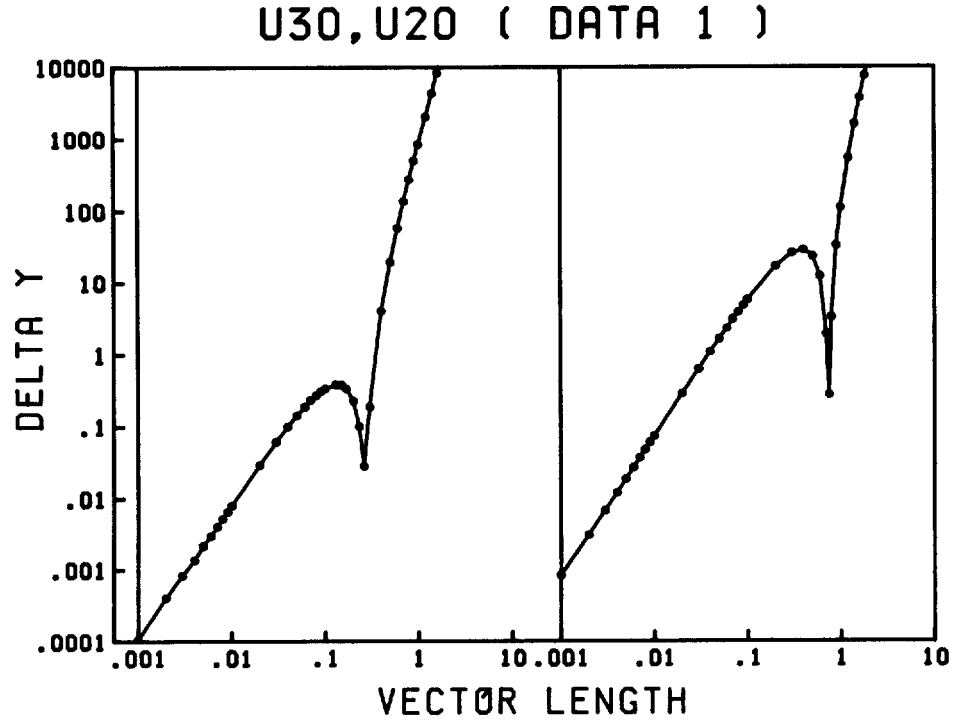


Fig. 5-b. Continued.

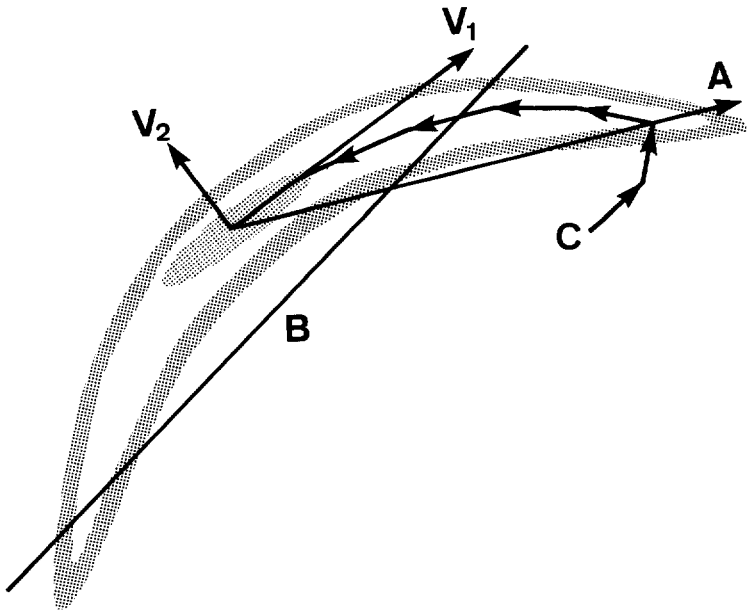


Fig. 6. Illustration of the response surface.
V : eigenvectors of the solution.
A : the vector from the solution to ith iteration times of MARQUARDT's method.
B : line of the mesh over parameter space.
C : trace of MARQUARDT's method.

confidence area looks like a long banana (DRAPER and SMITH 1968). MARQUARDT's method reaches the solution along the bottom of a ravine (NAKAGAWA and OYANAGI 1982). $\boldsymbol{u}_i$ is considered as the line A of Fig. 6. Two local minimums look as if they are on the line B of Fig. 6, but they are not true local minimums. SHIMAZU (1983) suggested that there is a danger of local minimums in multi-cohort models. But the question of whether this model has true local minimums or not must be cleared in the future.

$\boldsymbol{u}_{35}$ is shown in Table 5. It indicates the same tendency of $\boldsymbol{v}_1$. $\boldsymbol{v}_1$ is the minimum eigenvector of $\boldsymbol{H}$ scaled by $\boldsymbol{S}_2$, and $\boldsymbol{u}_{35}$ is parallel to that of $\boldsymbol{H}$ scaled by $\boldsymbol{S}_1$. Then $\boldsymbol{u}_{35}$ is not perfectly equal to $\boldsymbol{v}_1$.

See Table 2. The differences of the solutions of data–0 and –1 from the true values correspond with the results of this chapter. Linearity still remains in these data. But in data–2, the solution does not indicate the same tendency as data–0 and –1 because non-linearlty is relatively strong.

V. Consideration

Multi-cohort models have many parameters. Because the number of data is limited by the models, it is impossible to make the precision of the solutions high by increasing the amount of data. Then the confidence area becomes too long to use. The parameter concerned with all data is only M. Therefore, let the value of M be determined by external information, and fix it during calculation so that the confidence area becomes much smaller. However, a high precision is required for the value of M. It is better to decrease the numbers by fixing f and s to estimate only N, etc.

A high precision is required for the data because the confidence area becomes too long. It is also difficult to obtain good solutions for the data that do not satisfy the condition $F_{ij}=f_i s_j$. Therefore it may be impossible to use this model in practice, but it is still valuable to use this model by decreasing the number of parameters or by improving the model to check the results obtained by other studies. In this paper, it is treated as an application of MARQUARDT's method (AKAMINE 1986), so improvements in the models and programs, etc. are omitted.

Although MARQUARDT' s method takes a long time in this model, it is practical enough to get a high precision solution. The computing efficiency of this program is not so good because it was developed while aiming at an easily understood algorithm. Although it is possible to make the computing time shorter by improvement of the computing efficiency, it is easier by using a high speed computer or compilers. There is another method to decrease the amount of computation by using an approximation of a Hessian matrix, although it may lower the precision of the solutions. A new method different from MARQUARDT's might solve this model with a shorter computing time in the future.

VI. Conclusion

The program rewritten from AKAMINE (1986) for the multi-cohort model was tested using artificial data to obtain the following results.

1) It converged to true values of the solution for the high precision data.
2) N, M and the right part of s have a high positive correlation. On the other hand, f and the left part of s have a high positive correlation. These two sets have a high negative correlation with each other.
3) It converged to very different values for the data that added random numbers. Then the data must be required a high confidence and the data that do not satisfy the condition $F_{ij}=f_i s_j$ may not lead to good solutions.
4) A multi-cohort model has a high non-linearity. Thus it takes a long time to convergence even by MARQUARDT's method. It seems difficult to get a true solution by meshing over the parameter space because there are several local minimums on the mesh.

5) A multi-cohort model has too many parameters for the number of data. It is better to estimate the values for part of the parameters by another study, fix them, and calculate the values of the remaining parameters.

Acknowledgement

The author is indebted to Mr. K. ISHIOKA of Nansei Regional Fisheries Research Laboratory and Mr. F. KATO of Seikai 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 reading of the manuscript.

References

AKAMINE, T. (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.

AKAMINE, T., Y. NOSE and M. SHIMIZU (1982) A method of estimating salmon returning rate when the age compositions of returned groups are unknown. *Bull. Jap. Sea Reg. Fish. Res. Lab.*, (33), 141–145. *in Japanese with English summary.*

DOUBLEDAY, W. G. (1976) A least squares approach to analysing catch at age data. *Res. Bull. int. Commn NW Atlant. Fish.*, 12, 69–81.

DRAPER, N. R. and H. SMITH (1966) *Applied regression analysis.* John Wiley and Sons, New York. *translated into Japanese by* K. NAKAMURA (1968, 応用回帰分析, 森北出版株式会社, 東京, 378 pp).

GEN, M. and K. IDA (1983) パソコン会話型科学技術計算プログラム集. 工学図書株式会社, 東京, 163pp.

MARQUARDT, D. W. (1963) An algorithm for least-squares estimation of non-linear 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, 206 pp.

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.

POPE, J. G. and J. G. SHEPHERD (1982) A simple method for the consistent interpretation of catch-at-age data. *j. Cons. int. Explor. Mer*, 40, 176–184.

SHIMAZU, Y. (1983) Cohort analysis. *Population dynamics of fishery resources.* pp 30–45, 恒星社厚生閣, Tokyo, 149pp. *in Japanese.*

マルカール法によるマルチコホートモデルの解法

赤嶺達郎

赤嶺 (1986) のマルカール法の BASIC プログラムを書きかえてマルチコホートモデルを解いた. 人工データについて計算し解の近傍の状態を調べた. マルチコホートモデルはパラメータ数が多いため収束が遅く解も不安定である. 初期資源量と自然死亡係数には正の相関が, またこれらと漁獲死亡係数には負の相関が認められた.

Appendix. The BASIC program for the multi-cohort model by MARQUARDT's method.

```
10      '
20      '   Multi-cohort-model by Marquardt's method
30      '
40      '                        by  Tatsuro Akamine
50      '                                  1986-02-27
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 *SHUSEI9
230     GOSUB *PRINTOUT
240     END
500    *PEND2
510     PRINT "Convergence was completed."
520     PRINT
530     IREP=IREP-1
540     GOSUB *SHUSEI9
550     GOSUB *PRINTOUT
560     END
1000    *VARIDEF
1010     '
1020     '   Definition of functions
1030     '
1040     DEFINT I-N
1050     DEF FNG1(I,J)=FNFS2(I,J)/FNFSM2(I,J)*(1-FNH1(I,J))
1060     DEF FNH1(I,J)=EXP(-FNFSM2(I,J))
1070     DEF FNGF1(I,J)=FNG1(I,J)*(1/FI(I)-SJ(J)/FNFSM2(I,J)+SJ(J)*FNH2(I,J))
1080     DEF FNGS1(I,J)=FNG1(I,J)*(1/SJ(J)-FI(I)/FNFSM2(I,J)+FI(I)*FNH2(I,J))
1090     DEF FNGM1(I,J)=FNG1(I,J)*(-1/FNFSM2(I,J)+FNH2(I,J))
1100     DEF FNHF1(I,J)=-FNH1(I,J)*SJ(J)
1110     DEF FNHS1(I,J)=-FNH1(I,J)*FI(I)
1120     DEF FNHM1(I,J)=-FNH1(I,J)
1130     DEF   FNFS2(I,J)=FI(I)*SJ(J)
1140     DEF   FNFSM2(I,J)=FNFS2(I,J)+BMM
1150     DEF   FNH2(I,J)=FNH1(I,J)/(1-FNH1(I,J))
1160    RETURN
2000    *DATAREAD
2010     '
2020     '  Reading of data and variables
2030     '
2040     READ NIT,CLAMBDA,CNU
2050     PRINT "NUMBER OF ITERATION =";NIT
2060     PRINT "       LAMBDA       =";CLAMBDA
2070     PRINT "         NU         =";CNU
2080     PRINT
2090     READ N9,M9
2100     PRINT "NUMBER OF YEAR =";N9
2110     PRINT "NUMBER OF AGE  =";M9
2120     PRINT
2130     NP2=N9+M9-1 : NP3=NP2+N9 : NP4=NP3+M9
2140     NP=2*(N9+M9)
2150     DIM DNIJ(NP),DCEST(NP),PDELTA(NP)
2160     DIM COBS(N9,M9),BNIJ(N9,M9),DD1(N9,M9),FI(N9),SJ(M9)
2170     DIM HESSIAN(NP,NP),GVECTOR(NP),SCALE(NP)
2180     DIM BNIJ2(N9),BNIJ3(M9),FI2(N9),SJ2(M9)
2190     FOR I=1 TO N9
2200        FOR J=1 TO M9
2210           READ COBS(I,J)
2220        NEXT J
2230     NEXT I
```

```
2240   PRINT "DATA OF CATCH"
2250   FOR I=1 TO N9
2260      PRINT
2270      FOR J=1 TO M9 :  PRINT COBS(I,J), :NEXT J
2280   NEXT I
2290   PRINT
2300   RETURN
3000  *INITIAL
3010   '
3020   '  Setting of initial condition
3030   '
3040   FOR I=1 TO N9
3050      READ BNIJ(I,1) : BNIJ2(I)=BNIJ(I,1)
3060   NEXT I
3070   FOR J=2 TO M9
3080      READ BNIJ(1,J) : BNIJ3(J)=BNIJ(1,J)
3090   NEXT J
3100   FOR I=1 TO N9
3110      READ FI(I) : FI2(I)=FI(I)
3120   NEXT I
3130   FOR J=1 TO M9
3140      READ SJ(J) : SJ2(J)=SJ(J)
3150   NEXT J
3160   READ BMM : BMM2=BMM
3170   GOSUB *CALD2
3180   Y1=Y2
3190   IREP=0 : YDELTA=0
3200   GOSUB *PRINTOUT
3210   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   '
4110   GOSUB *SUMUP1
4120   '
4130   '   Scaling of parameters
4140   '
4150   FOR I=1 TO NP
4160      SCALE1=SQR(HESSIAN(I,I))
4170      SCALE(I)=.5*(SCALE1+HESSIAN(I,I)/SCALE1)
4180   '           for BUG of SQR-function of N88-BASIC(86)
4190   NEXT I
4200   FOR I=1 TO NP
4210      GVECTOR(I)=GVECTOR(I)/SCALE(I)
4220      FOR J=I TO NP
4230         HESSIAN(I,J)=HESSIAN(I,J)/SCALE(I)/SCALE(J)
4240      NEXT J
4250   NEXT I
4260   '
4270   '  Reserve of HESSIAN and GVECTOR
4280   '
4290   FOR I=2 TO NP
4300      HESSIAN(I,1)=GVECTOR(I)
4310   NEXT I
4320   FOR I=2 TO NP-1
4330      FOR J=I+1 TO NP
4340         HESSIAN(J,I)=HESSIAN(I,J)
4350      NEXT J
4360   NEXT I
4370   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)    :'  Scaling
5200  NEXT I
5210  FOR I=1 TO N9 : BNIJ(I,1)=BNIJ2(I)+PDELTA(I) : NEXT I
5220  FOR J=2 TO M9 : BNIJ(1,J)=BNIJ3(J)+PDELTA(N9+J-1) : NEXT J
5230  FOR I=1 TO N9 : FI(I)=FI2(I)+PDELTA(NP2+I) : NEXT I
5240  FOR J=1 TO M9 : SJ(J)=SJ2(J)+PDELTA(NP3+J) : NEXT J
5250  BMM=BMM2+PDELTA(NP)
5260  '
5270  GOSUB *CALD2
5280  IF Y2>=Y1 GOTO *PREREP
5290  '
5300  CLAMBDA=CLAMBDA/CNU
5310  YDELTA=Y2-Y1
5320  Y1=Y2
5330  FOR I=1 TO N9 : BNIJ2(I)=BNIJ(I,1) : NEXT I
5340  FOR J=2 TO M9 : BNIJ3(J)=BNIJ(1,J) : NEXT J
5350  FOR I=1 TO N9 : FI2(I)=FI(I) : NEXT I
5360  FOR J=1 TO M9 : SJ2(J)=SJ(J) : NEXT J
5370  BMM2=BMM
5380  GOSUB *PRINTOUT
5390  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 *CALD2
6010  '
6020  '  Calculation of object function
6030  '
6040   Y2=0
6050   FOR I=1 TO N9
6060     FOR J=1 TO M9
6070       IF I<>1 THEN IF J<>1 THEN BNIJ(I,J)=BNIJ(I-1,J-1)*FNH1(I-1,J-1)
6080       CEST=BNIJ(I,J)*FNG1(I,J)
6090       D1=COBS(I,J)-CEST
6100       DD1(I,J)=D1
6110       Y2=Y2+D1*D1
6120     NEXT J
6130   NEXT I
6140  RETURN
```

```
7000  *GAUSS
7010   '
7020   '  GAUSS's 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
8050    PRINT "IREP=";IREP
8060    PRINT "     D2    =";Y1
8070    PRINT " DELTA-D2 =";YDELTA
8080    PRINT
8090    PRINT " N(I,1)= "
8100    FOR I=1 TO N9 :  PRINT BNIJ2(I), : NEXT I
8110    PRINT : PRINT
8120    PRINT " N(1,J)= "
8130    FOR J=2 TO M9 :  PRINT BNIJ3(J), : NEXT J
8140    PRINT : PRINT
8150    PRINT " F(I)=  "
8160    FOR I=1 TO N9 :  PRINT FI2(I), : NEXT I
8170    PRINT : PRINT
8180    PRINT " S(J)= "
8190    FOR J=1 TO M9 :  PRINT SJ2(J), : NEXT J
8200    PRINT : PRINT
8210    PRINT "  M  =";BMM2
8220    PRINT
8230   RETURN
10000   '
10010  DATA 100 , 0.01 , 2    :' number of iterations , lambda , nu
10020  '
10030  '   Example data of catch
10040  '
10050  DATA 10 , 5                              :' n , m
10060  DATA   44,  101,   82,   71,   47        :' C0(1,1) ----- C0(1,m)
10070  DATA   43,   80,   76,   65,   35
10080  DATA   98,  173,  134,  130,   71
10090  DATA   42,  171,  120,   95,   57
10100  DATA   43,  146,  234,  164,   81
10110  DATA   74,   60,   78,  126,   53
10120  DATA  148,  204,   66,   86,   87
10130  DATA   40,   99,   55,   18,   15
10140  DATA   29,  199,  205,  120,   26
10150  DATA   77,   90,  249,  261,   96        :' C0(n,1) ----- C0(n,m)
10160  '
10170  '   Example initial value of parameter
10180  '
10190  DATA 1000,1000,1000,1000,1000,1000,1000,1000,1000,1000  :' N(i,1)
10200  DATA 500,500,500,500                                    :' N(1,j)
10210  DATA 1,1,1,1,1,1,1,1,1,1                                :' f(i)
10220  DATA .2,.2,.2,.2,.2                                     :' s(j)
10230  DATA .3                                                 :' M
```

```
20000  *SUMUP1
20010   '
20020   '  Division of C(i,j)
20030   '                              Diagonal part
20040    FOR K=1 TO N9-M9+1
20050       FOR J=1 TO M9
20060          I=K+J-1
20070          GOSUB *SUMUP2
20080          GOSUB *SUMUP5
20090       NEXT J
20100    NEXT K
20110    '                             Lower left part
20120    FOR K=N9-M9+2 TO N9
20130       FOR J=1 TO N9-K+1
20140          I=K+J-1
20150          GOSUB *SUMUP2
20160          GOSUB *SUMUP5
20170       NEXT J
20180    NEXT K
20190    '                             Upper right part
20200    FOR L=2 TO M9
20210       FOR I=1 TO M9-L+1
20220          J=I+L-1
20230          GOSUB *SUMUP2
20240          GOSUB *SUMUP5
20250       NEXT I
20260    NEXT L
20270   RETURN
21000  *SUMUP2
21010   '
21020   '  Calculation of differential
21030   '
21040    FOR II1=1 TO NP : DCEST(II1)=0 : NEXT II1
21050    GOSUB *SUMUPN2
21060    GOSUB *SUMUPF2
21070    GOSUB *SUMUPS2
21080    GOSUB *SUMUPM2
21090   RETURN
22000  *SUMUPN2      :'  dC/dN
22010    IR2=K
22020    IF I<J THEN IR2=N9+L-1
22030    IF I=1 THEN DNIJ(IR2)=1 : GOTO *PARTN2
22040    IF J=1 THEN DNIJ(IR2)=1 : GOTO *PARTN2
22050    DNIJ(IR2)=DNIJ(IR2)*FNH1(I-1,J-1)
22060   *PARTN2
22070    DCEST(IR2)=DNIJ(IR2)*FNG1(I,J)
22080   RETURN
23000  *SUMUPF2      :'  dC/df
23010    IR1=I-J+1
23020    IF I<J THEN IR1=1
23030    FOR IP1=IR1 TO I
23040       IP2=I-IP1 : IP3=NP2+IP1
23050       IIP2=IP2+1 : IF IIP2>3 THEN IIP2=3
23060       ON IIP2 GOSUB *PARTF1 , *PARTF2 , *PARTF3
23070    NEXT IP1
23080   RETURN
23090       *PARTF1
23100         DCEST(IP3)=BNIJ(I,J)*FNGF1(I,J)
23110        RETURN
23120       *PARTF2
23130         Q1=BNIJ(I-1,J-1)*FNHF1(I-1,J-1)
23140         DNIJ(IP3)=Q1
23150         DCEST(IP3)=Q1*FNG1(I,J)
23160        RETURN
23170       *PARTF3
23180         Q2=DNIJ(IP3)*FNH1(I-1,J-1)
23190         DNIJ(IP3)=Q2
23200         DCEST(IP3)=Q2*FNG1(I,J)
23210        RETURN
```

```
24000 *SUMUPS2     :'  dC/ds
24010   IR1=1
24020   IF I<J THEN IR1=J-I+1
24030   FOR IP1=IR1 TO J
24040      IP2=J-IP1 : IP3=NP3+IP1
24050      IIP2=IP2+1 : IF IIP2>3 THEN IIP2=3
24060      ON IIP2 GOSUB *PARTS1 , *PARTS2 , *PARTS3
24070   NEXT IP1
24080  RETURN
24090      *PARTS1
24100        DCEST(IP3)=BNIJ(I,J)*FNGS1(I,J)
24110       RETURN
24120      *PARTS2
24130        Q1=BNIJ(I-1,J-1)*FNHS1(I-1,J-1)
24140        DNIJ(IP3)=Q1
24150        DCEST(IP3)=Q1*FNG1(I,J)
24160       RETURN
24170      *PARTS3
24180        Q2=DNIJ(IP3)*FNH1(I-1,J-1)
24190        DNIJ(IP3)=Q2
24200        DCEST(IP3)=Q2*FNG1(I,J)
24210       RETURN
25000 *SUMUPM2     :'  dC/dM
25010   IF I=1 THEN DNIJ(NP)=0 : GOTO *PARTM2
25020   IF J=1 THEN DNIJ(NP)=0 : GOTO *PARTM2
25030   DNIJ(NP)=DNIJ(NP)*FNH1(I-1,J-1)+BNIJ(I-1,J-1)*FNHM1(I-1,J-1)
25040  *PARTM2
25050   DCEST(NP)=DNIJ(NP)*FNG1(I,J)+BNIJ(I,J)*FNGM1(I,J)
25060  RETURN
26000 *SUMUP5
26010  '
26020  '  Calculation of HESSIAN and GVECTOR
26030  '
26040   D1=DD1(I,J)
26050   FOR II1=1 TO NP
26060      GVECTOR(II1)=GVECTOR(II1)+D1*DCEST(II1)
26070      FOR JJ1=II1 TO NP
26080         HESSIAN(II1,JJ1)=HESSIAN(II1,JJ1)+DCEST(II1)*DCEST(JJ1)
26090      NEXT JJ1
26100   NEXT II1
26110  RETURN
27000 *SHUSEI9
27010  '
27020  '  Standardization of f and s
27030  '
27040  TOTALS=0
27050  FOR J=1 TO M9 : TOTALS=TOTALS+SJ2(J) : NEXT J
27060  FOR J=1 TO M9 : SJ2(J)=SJ2(J)/TOTALS : NEXT J
27070  FOR I=1 TO N9 : FI2(I)=FI2(I)*TOTALS : NEXT I
27080  RETURN
30000 '
30010 ' Notation of variables
30020 '
30030 ' NIT        : number of iteration
30040 ' CLAMBDA    : lambda
30050 ' CNU        : nu
30060 ' N9         : n                  = number of year
30070 ' M9         : m                  = nember of age
30080 ' COBS       : C0(i,j)            = data of catch
30090 ' CEST       : C(i,j)             = estimation of catch
30100 ' DD1        : C(i,j)-C0(i,j)
30110 ' BNIJ       : new N(i,j)
30120 ' FI         : new f(i)
30130 ' SJ         : new s(j)
30140 ' BMM        : new M
30150 ' BNIJ2      : old N(i,1)
30160 ' BNIJ3      : old N(1,j)
30170 ' FI2        : old f(i)
30180 ' SJ2        : old s(j)
30190 ' BMM2       : old M
```

```
30200 ' DNIJ       : dN/d(parameter)
30210 ' DCEST      : dC/d(parameter)
30220 ' HESSIAN    : H                  = hessian matrix
30230 ' GVECTOR    : g                  = gradient vector
30240 ' SCALE      : S                  = scaling matrix
30250 ' PDELTA     : delta(parameter)
30260 ' Y1         : old Y
30270 ' Y2         : new Y
30280 ' YDELTA     : delta Y
31000 ' FNG1       : g
31010 ' FNH1       : h
31020 ' FNGF1      : dg/df
31030 ' FNGS1      : dg/ds
31040 ' FNGM1      : dg/dM
31050 ' FNHF1      : dh/df
31060 ' FNHS1      : dh/ds
31070 ' FNHM1      : dh/dM
```