



HOKKAIDO UNIVERSITY

Title	漁獲物の体長組成データにもとづくマルチコホート解析法の開発と推定精度の評価
Author(s)	松石, 隆
Description	東京大学大学院農学生命科学研究科 博士(農学)学位申請論文
Degree Grantor	University of Tokyo
Degree Name	博士(農学)
Issue Date	1996-03-15
Doc URL	https://hdl.handle.net/2115/39355
Type	journal article
File Information	Matsuishi1996.pdf



漁獲物の体長組成データにもとづく
マルチコホート解析法の開発と
推定精度の評価

Development and Accuracy Assessment of
a Method of Multi Cohort Analysis
based on Length Composition Data
of Catch

松石 隆

北海道大学水産学部

東京大学大学院農学生命科学研究科
博士（農学）学位申請論文

1996年1月19日	提出・受理
1996年2月16日	審査・合格
1996年3月15日付	学位授与

目次

1	緒言	1
1.1	体長別資源量推定の必要性	1
1.2	本研究の目的	2
1.3	概要	3
2	先行研究	5
2.1	Age-Length Key	5
2.2	体長組成分解に関する研究	5
2.3	シングルコホート解析	6
2.4	マルチコホート解析	6
2.5	体長組成データによるコホート解析-Jonesの方法	10
2.6	体長組成データを用いたマッコウクジラの資源動態解析	10
3	体長組成データによるマルチコホート解析	11
3.1	概要	11
3.2	必要とするデータ	11
3.3	行列 P の推定	11
3.3.1	方法1	12
3.3.2	方法2	12
3.4	個体数推定法	13
4	妥当性の検討	15
4.1	目的	15
4.2	模擬データの作り方	16
4.2.1	パラメータの設定	16
4.2.2	模擬データ作成の手順	17
4.3	数値計算例	18
4.3.1	パラメータ	18
4.3.2	精度の評価の仕方	21
4.4	結果	21
4.4.1	誤差なしの基本データについて	21

4.4.2	A系列のデータについて	21
4.4.3	B系列のデータについて	28
5	従来の方法との比較	34
5.1	目的	34
5.2	方法	34
5.3	パラメータ	35
5.4	結果	35
6	事例解析 1: 秋田県南部におけるアワビ漁業	41
6.1	漁業の概要	41
6.2	解析に用いたデータ	44
6.3	解析方法	44
6.4	資源量推定結果	50
7	事例解析 2: 北海道噴火湾沿岸におけるスケトウダラ漁業	56
7.1	漁業の概要	56
7.2	解析に用いたデータ	59
7.3	解析方法	60
7.4	資源量推定結果	63
8	まとめ	72
	摘要	75
	謝辞	79
	参考文献	80
	付録	86
A	A系列のシミュレーション結果	87
B	B系列のシミュレーション結果	96
C	LPA プログラムリスト	106
D	LPA 精度評価プログラムリスト	112

第 1 章

緒言

1.1 体長別資源量推定の必要性

外国 200 海里水域や公海における漁業にたいする規制が強化され、水産業をめぐる状況は厳しくなっている。これに対応し、周辺水域の水産資源を維持・増大させるとともにこれを安定的な漁業生産につなげていく「管理型漁業」の推進が、重要になってきている。これに伴い、漁場別漁獲情報や生物データの収集・調査船による資源量推定などさまざまな調査とそのデータの科学的解析が行われている。しかし、その評価精度やこれにもとづく資源管理方策の実効性については、まだ確立した評価を得るには至っていない¹⁾。

資源量推定は、今まで多くの魚種でコホート解析²⁾の方法をはじめとする VPA によって行われてきている^{3, 4, 5, 6, 7)}。これらの方法は、年齢別の漁獲個体数、最終年齢での漁獲死亡係数、自然死亡係数があれば、各年級群の年齢別の資源尾数を過去にさかのぼって計算できる。また、これらの方法は扱いが難しい努力量のデータを使わずに資源量を推定することができる。しかし、年齢別の漁獲尾数を推定するためには、多くの個体の年齢査定をする必要がある。

年齢形質が明らかな場合は資源量推定が可能であるが、年齢形質が明らかではない魚種も少なくない。また、過去のデータの解析をするときには体長は測定されていても年齢査定をしていない場合も多い。

このような場合は、従来は体長組成を正規分布分解などによって年齢組成に変換する方法が用いられた。そして、これに関して、最尤法などを用いた高度な解析法が開発されている^{8, 9, 10, 11, 12, 13, 14, 15)}。成長がはやい魚の場合は、体長組成から年齢組成を推定する方法によって比較的精度良く年齢組成が推定できるが、寿命が長く高齢での成長が遅い魚種や、体長による漁獲選択がある場合などは、体長組成から年齢組成を推定する際に多くの誤差が混入している。

さらに、推定された年齢別資源量から、たとえば若齢魚の漁獲制限をするべきである、というような資源管理の方策を導く場合、漁獲規制は、主に漁獲対象魚の体長による漁具の選択性を利用することが多い。たとえば、魚類では網目制限、貝類・甲殻類では体長制限がこれにあたる。このような漁獲規制の実効性を検討するために必要なのは、年齢別資

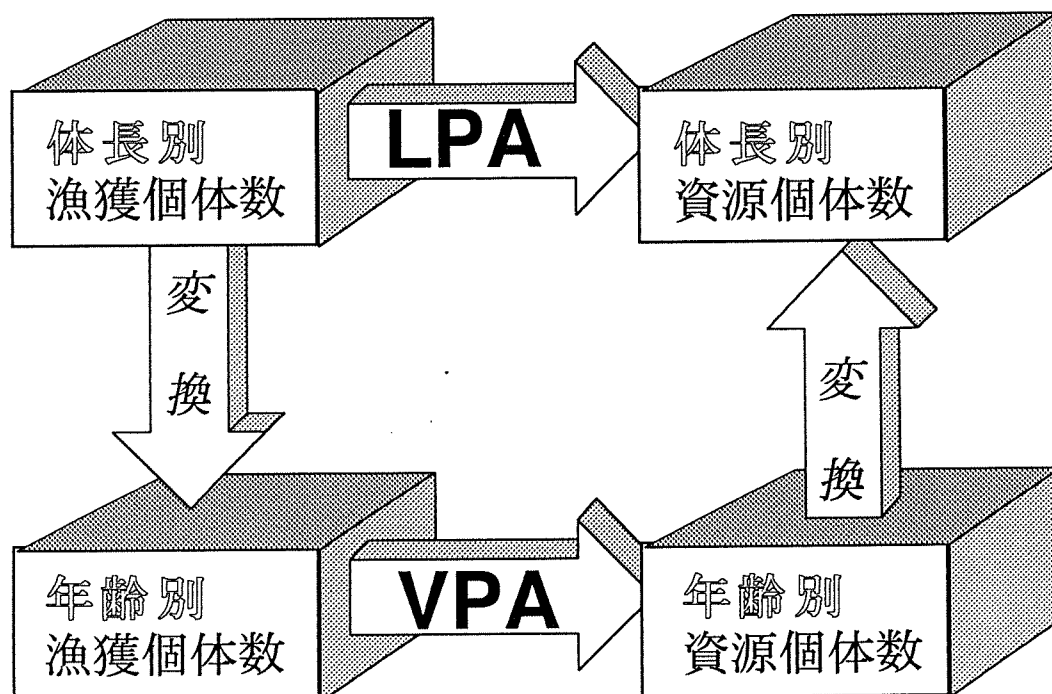


Fig. 1.1: 体長別漁獲個体数から体長別資源個体数を推定する時の手順。VPA を用いる場合は体長-年齢間の変換を2回行わなければならないが、本研究で開発したLPA (Length based Population Analysis) を用いると直接推定できる。

源量ではなく体長別の資源量である。

年齢査定が十分にできない魚に対して体長別資源量を推定するためには、データを体長でとって、これを年齢に変換し、コホート解析にかけて年齢別資源量を推定して、さらにこれを体長に変換して資源管理するという手順をふむことになり、体長-年齢間の変換を2回行っていることになる。そしてそのたびに誤差が入ってきている。

そこで、とった体長組成データを年齢に変換せずに「体長によるコホート解析」にかけ、この結果得られた「体長別資源量」から資源管理を考えれば、2回の変換で生じた誤差を省いてより精度良く推定できるはずである (Fig. 1.1)。

1.2 本研究の目的

第2章で述べるように、体長によるコホート解析は既に Jones¹⁶⁾が提案しているが、この方法は成長の個体差を考慮に入れていない。また、複数年級群間の整合性も考慮していな

い。白木原¹⁷⁾は、マッコウクジラの体長組成をもとに、再生産関係を用いて将来の資源動態の予測を行っている。この一部は体長組成を用いたマルチコホートに類似する方法になっているが、再生産関係を用いているため、出産頭数が少ない哺乳類では可能でも、初期減耗の大きい魚類への応用は不可能である。

本研究では、毎年の体長組成を、漁獲物の体長組成と若干の年齢査定データから、整合性をもって推定できる、「体長によるマルチコホート解析法」を開発した。

また、この方法の妥当性や精度、従来の方法との性能の比較をシミュレーションデータをもちいて行なった。

また、実際に、秋田県南部におけるアワビ漁業のデータと、北海道噴火湾沿岸におけるスケトウダラ漁業のデータに適用した。

1.3 概要

第2章では、体長組成を年齢組成に変換する方法、年齢組成によるコホート解析法、および体長組成によるコホート解析の今までの試みを概説する。

第3章では、本研究で開発した、体長によるマルチコホート解析 (Length based Population Analysis / LPA) を説明する。この方法は、第2章で述べられたマルチコホート解析を体長組成で用いるように改変したものである。

第4章では、LPAの推定精度を、シミュレーションによって検討し、LPAの際に与えるパラメータが結果へ与える影響の感度解析を行う。

第5章では、第2章に述べられた、体長組成を年齢組成に変換する方法と、年齢組成によるコホート解析を用いて得られる従来の推定値と、LPAの推定値の比較を行う。

第6章では、秋田県南部におけるアワビ漁業のデータへLPAを適用する。アワビは、殻から成長の履歴がわかるので、体長組成の遷移を直接推定することが可能である。

第7章では、北海道噴火湾沿岸におけるスケトウダラ漁業のデータへLPAを適用する。なおスケトウダラの成長の履歴は推定が難しいので、体長組成の推移を年齢別平均体長とその分散から推定する。

第8章では、以上の議論を総括し、本法の適用可能性と今後の改良点について述べる。

付録に、本文で紹介できなかったシミュレーションの結果とLPAのプログラムのソースプログラムを付す。

なお、本論文に用いる主な記号とその意味をTable 1.1に記した。

Table 1.1: 論文中に使用した主な記号とその意味

記号	意味
A	計算する最大年齢
a	年齢
C	漁獲個体数
d	体長階級幅
F	漁獲係数
F_T	最終年齢における漁獲係数
f	完全加入漁獲係数
I	体長階級数
i	体長階級
i_n	$j+1$ 年の体長階級
i_p	j 年の体長階級
i_r	完全加入体長階級 i_r
J	計算する年数
j	年
K	von Bertalanffy 成長式のパラメータ
L_∞	von Bertalanffy 成長式のパラメータ
ℓ	体長
M	自然死亡係数
N	資源個体数
P	$j+1$ 年と j 年の体長組成の関係を表す行列 P
p	その体長階級(年齢)に含まれる個体の割合
R	加入個体数
s	選択率(ある体長階級を1とした相対値)
s_a	年齢 a の個体の体長の標準偏差
α_ϵ	成長曲線のまわりの変動のパラメータ
β_ϵ	成長曲線のまわりの変動のパラメータ
CV	変動係数
SS	残差平方和

第 2 章

先行研究

ここでは、従来の年齢によるコホート解析と、その前処理として用いられる体長組成を年齢組成に変換する方法についてレビューし、その方法と特徴を述べる。まず、体長別漁獲個体数を年齢別漁獲個体数に変換する方法として、Age-Length Key と体長組成分解について述べる。その後、年齢別漁獲個体数から年齢別資源個体数を推定する方法として、シングルコホート解析、マルチコホート解析について述べる。また、体長別漁獲個体数から直接、体長別資源個体数を推定する方法として、Jones の方法とマッコウクジラの資源動態解析について述べる。

2.1 Age-Length Key

Age-Length Key 法は、各体長階級に属する個体の年齢組成を求め、これを用いて体長組成を年齢組成に変換する方法である。Fridriksson¹⁸⁾がタラの資源研究にはじめて導入し、その後、年齢査定が可能な魚種の年齢別漁獲量を推定する方法として広く用いられている。

Tanaka¹⁹⁾および田中²⁰⁾は、この方法の推定精度を検討している。その計算によると、Age-Length Key を用いる場合と、漁獲物を体長層別に分けずに年齢査定を行って年齢組成を得る場合との精度の比は、体長層別で年齢がどの程度分けられるかによって左右されるが、一例では同じ個体数の年齢査定で、標準誤差を 50% 減少させることが可能であるとしている。

Age-Length Key を作成する場合、複数年の標本を用いると、偏りが生じることが知られている。これに対し Kimura and Chikuni⁹⁾と真子・松宮²¹⁾は、EM アルゴリズムによる Age-Length Key の推定法を提案しており、これはともに Hasselblad²²⁾の、方法の展開となっている²³⁾。

2.2 体長組成分解に関する研究

Age-Length Key は、少数の個体でも、年齢査定をする必要がある。しかし、全く年齢形質が得られない魚種も少なくない。このような場合は、体長組成だけから年齢組成を推定しなければならない。

もし同一年級群の体長が正規分布をすると仮定できれば、体長組成は複数の正規分布の和として表される。したがって、多峰型を示す体長組成をいくつかの正規分布に分解すれば、その面積比から年齢組成が得られると考えられる。

Buchanan-Wollaston and Hodgson²⁴⁾は、正規分布の密度関数の対数を取ると、放物線になることに注目し、モード付近の3点から放物線を推定する方法を考えた。Cassie⁸⁾は、正規確率紙を用いて図的に多峰型を正規分布に分解した。田中²⁵⁾は、図的に放物線のあてはめを行った。さらに、岡²⁶⁾やBhattacharya²⁷⁾は、放物線の微係数が直線になることを利用して、直線のあてはめに持ち込んだ。

その後、コンピュータの普及によって、最尤法等によるパラメータ推定が可能になり、Hasselblad²²⁾が、はじめて最尤法によってこの問題を定式化した。Macdonald and Pitcher²⁸⁾は、この手法を漁業データへ応用した。近年は、ELEFAN²⁹⁾、MIXNOR³⁰⁾、LEFRAN³¹⁾、MULTIFAN¹⁴⁾、といった計算ソフトが普及し、広く用いられている。

Martin and Cook¹⁵⁾は、Age-Length Key による情報と、体長組成の情報を同時に用いて最尤法により年齢組成を推定する方法を開発した。

2.3 シングルコホート解析

シングルコホート解析とは、単一年級群だけに注目し、その年齢群の個体数が年齢を追うごとにどのように変化してきたかを推定するものである。必要なデータは、年齢別漁獲量 C_a 、自然死亡係数 M 、最終年齢での漁獲死亡係数 (Terminal F) F_T である。

基本的な式は、以下の2本である。

$$N_{a+1} = N_a e^{-(M+F_a)} \quad (2.1)$$

$$C_a = N_a \frac{F_a}{M + F_a} (1 - e^{-(M+F_a)}) \quad (2.2)$$

ここで、 N_a は、対象としている年齢群が a 歳になった時の個体数、 F_a は対象となっている年齢群が a 歳の間の漁獲係数である。

求めたい値は $N_1, N_2, \dots, N_T; F_1, F_2, \dots, F_{T-1}$ の $2T-1$ 個である。それに対して方程式も $2T-1$ 本あるので、原理としては未知数は決定できる。

計算の過程で C_a と N_a から F_a を求める必要が生じるが、これは解析的には解けない。この過程を回避するために、Pope²⁾の方法など、さまざまな近似式が提案されている^{5, 6)}。

2.4 マルチコホート解析

マルチコホート解析とは、複数年齢群の解析を一度に行うコホート解析である。漁獲係数を年齢に依存する項と年に依存する項に分解するので、Separable VPA とよばれる。

各年齢群は、上述のようにシングルコホート解析によって解析できる。それぞれの年齢群を独立に計算するため、年齢毎の漁獲選択性が年によらずほぼ一定であるはずであって

も、そのような結果が出ないことが多い。また、各年級ごとに F_T を与える必要が生じる。従来は $F_{j,T}$ を試行錯誤的に動かして、整合性のとれた推定値を得ようとしていた。

Pope and Shepherd³²⁾はこの試行錯誤を、最適化アルゴリズムを使って自動的に求める方法を開発した。これによって、今まで恣意的に与えられていた $F_{j,T}$ なども、データから求めることができるようになった。

アルゴリズム

漁具の選択性が年ごとに変わらないことを利用して、以下のように定式化した。各年 j ($j = 1, \dots, J$) の完全加入漁獲係数 (fully exploited fishing mortality) を f_j , 年齢 a ($a = 1, \dots, A$) ごとの選択率を s_a とし、年 j の年齢 a の漁獲係数 $F_{j,a}$ を、

$$F_{j,a} = f_j \times s_a \quad (2.3)$$

とする。

このように漁獲係数を年に依存する f と、年齢に依存する s に分解することによって、 $J \times A$ 個あった未知変数 $F_{j,a}$ は、 $J + A$ 個の未知変数 s_j, f_a で表せることになった。

以下では、 s_j, f_a と M の $J + A + 1$ 個の未知変数を決定するために、 $J + A + 1$ 本の方程式を提示する

同じ年級群の1歳違う魚の漁獲個体数の比の対数を D で表す

$$D_{j,a} = \ln[C_{j+1,a+1}/C_{j,a}] \quad (2.4)$$

$C_{j,a}, C_{j+1,a+1}$ とも、観測されているので、 $D_{j,a}$ は与えられる。

一方、(2.3) の f を用いて、漁獲個体数を表すと、

$$\hat{C}_{j,a} = N_{j,a}(1 - e^{-Z_{j,a}}) \frac{F_{j,a}}{Z_{j,a}} \quad (2.5)$$

である。ここで $Z_{j,a} = F_{j,a} + M$ である。

同様に、

$$\hat{C}_{j+1,a+1} = N_{j+1,a+1}(1 - e^{-Z_{j+1,a+1}}) \frac{F_{j+1,a+1}}{Z_{j+1,a+1}} \quad (2.6)$$

である。また、個体数の変化は

$$N_{j+1,a+1} = N_{j,a}e^{-Z_{j,a}} \quad (2.7)$$

として表される。(2.4) に相当する推定値は (2.6) と (2.5) の商なので、(2.7) に注意して計算すると、

$$\hat{D}_{j,a} = \ln\left\{F_{j+1,a+1} \frac{Z_{j,a}(1 - e^{-Z_{j+1,a+1}})e^{-Z_{j,a}}}{F_{j,a}Z_{j+1,a+1}(1 - e^{-Z_{j,a}})}\right\} \quad (2.8)$$

となる。 $\hat{D}_{j,a}$ は N に依存せず、 $f_j, f_{j+1}, s_a, s_{a+1}, M$ の関数になっている。

実測値と理論値の差を $R_{j,a}$ と書く。

$$R_{j,a} = \hat{D}_{j,a} - D_{j,a} \quad (2.9)$$

もし、現実には $J + A + 1$ 個の未知変数 s, f, M が見つければ、 $\hat{D}$ は D のまわりにばらつくことになり、年ごとまたは年齢ごとの平均値が、推定値と観測値で等しくなることが期待される。そのとき、

$$\sum_{a=1}^A (\hat{D}_{j,a} - D_{j,a}) = \sum_{a=1}^A R_{j,a} = 0 \quad (2.10)$$

これが、 $j = 1, \dots, J - 1$ で成り立つ。

$$\sum_{j=1}^J (\hat{D}_{j,a} - D_{j,a}) = \sum_{j=1}^J R_{j,a} = 0 \quad (2.11)$$

これが、 $a = 1, \dots, A - 1$ で成り立つ。

また、ある年齢（加入が終了した時） a_r において、

$$s_{a_r} = 1 \quad (2.12)$$

とする。

以上で、未知数 $y + a + 1$ に対して、 $y + a - 2$ 本の方程式がたったことになる。 f_J, s_A, M を与えると、未知数の数だけの方程式がそろい、未知数が全部決定できることになる。

方程式の解法

1. まず

$$\begin{aligned} f_j &= f_J \quad (j = 1, 2, \dots, J - 1) \\ s_a &= 1 \quad (a = 1, 2, \dots, A - 1). \end{aligned}$$

と置く。

2. これを用いて $\hat{D}$ を (2.8) によって計算し、 $R_{j,a}$ を (2.9) によって計算する。

3. この残差 R を用いて、 f, s を改善する。改善された f, s を $\tilde{f}, \tilde{s}$ と書くと、

$$\tilde{f}_j = f_j \exp\left[\sum_{a=1}^{J-1} R_{j,a}/A\right] \quad (j = 1, \dots, J - 1) \quad (2.13)$$

$$\tilde{s}_a = s_a \exp\left[\sum_{j=1}^{A-1} R_{j,a}/J\right] \quad (a = 1, \dots, A - 1) \quad (2.14)$$

4. $s_{a_r} = 1$ となるように、

$$\tilde{s}_a \leftarrow \tilde{s}_a / \tilde{s}_{a_r} \quad (2.15)$$

と置き直す。

5. 2～4の操作を繰り返すと、次第に R が小さくなって来る。普通10回から30回、この操作を繰り返すと $\tilde{f}, \tilde{s}$ はほとんど変化しなくなる。この時の $\tilde{f}, \tilde{s}$ が推定値 $\hat{f}, \hat{s}$ であり、推定漁獲係数 $\hat{F}$ は、

$$\hat{F}_{j,a} = \hat{f}_j \times \hat{s}_a \quad (2.16)$$

である。

以上で推定された、 $\hat{F}_{j,a}$ を用いて推定個体数 $\hat{N}_{j,a}$ を求める。 j 年に生まれた年級群が a 歳の時の個体数は、

$$\hat{N}_{j+a-1,a} = \hat{N}_{j,1} e^{-\sum_{k=1}^{a-1} \hat{Z}_{j+k-1,k}} \quad (2.17)$$

である。

また、 $\hat{C}$ は (2.5) で与えられているので、これらを整理して、残差平方和 SS を以下のように定義する。

$$SS = \sum_{a=1}^{A'} [\ln C_{j+a-1,a} - \ln \hat{N}_{j,1} \hat{E}_{j+a-1,a}]^2 \quad (2.18)$$

ここで A' は、コホートが計算できる最大の年齢である。すなわち、

$$A' = \min(A, J+1-j)$$

また

$$\hat{E}_{j+a-1,a} = \frac{\hat{F}_{j+a-1,a}}{\hat{Z}_{j+a-1,a}} \{1 - e^{-\hat{Z}_{j+a-1,a}}\} e^{-\sum_{k=1}^{A-1} \hat{Z}_{j+k-1,k}} \quad (2.19)$$

である。

残差平方和を最小にする値を求めるために、

$$\frac{\partial SS}{\partial \hat{N}_{j,1}} = 0 \quad (2.20)$$

を解くと、

$$\ln \hat{N}_{j,1} = \frac{1}{J} \sum_{a=1}^J [\ln C_{j+a-1,a} - \ln \hat{E}_{j+a-1,a}] \quad (2.21)$$

となる。

$\hat{N}_{1,a}$ についても同様に求まる。これが求まれば、あとは

$$\hat{N}_{j+1,a+1} = \hat{N}_{j,a} e^{-\hat{Z}_{j,a}} \quad (2.22)$$

の関係を用いて、全ての j, a で推定個体数 $\hat{N}$ が計算できる。

2.5 体長組成データによるコホート解析-Jonesの方法

Jones¹⁶⁾は、体長によるコホート解析を示している。この方法は、成長にあわせて時間間隔を調整したVPAである。

任意の時間間隔 Δt に対して、

$$N_t = N_{t+\Delta t}e^{M\Delta t} + C_te^{M\Delta t/2} \quad (2.23)$$

と書ける。ここで N_t は時刻 t におけるある年級群の個体数、 M は自然死亡係数、 C_t は、時刻 t から $t + \Delta t$ の間の漁獲量である。

Δt を、体長が ℓ_1 から ℓ_2 になるまでの時間とする。von Bertalanffyの成長曲線に従うとすると、

$$\Delta t = \frac{1}{K} \ln\left(\frac{L_\infty - \ell_1}{L_\infty - \ell_2}\right) \quad (2.24)$$

である。

漁獲される最大体長階級から、順次この計算をすると、漁獲物の体長組成からコホート解析を行うことができる。

この方法によって、成長曲線に従って成長していれば個体数推定が可能であるが、実際は成長曲線のまわりにばらついているので、実際の推定の際には精度が悪くなることが予想される。

2.6 体長組成データを用いたマッコウクジラの資源動態解析

北西太平洋のマッコウクジラについては、体長組成の推移を用いた資源量評価が行われている^{33, 34, 35, 36, 37, 38, 39)}。マッコウクジラについては、年齢査定は行われていないが、捕獲された個体の性別と体長のデータは整備されている。そこで、この研究ではマッコウクジラの再生産や成長をモデル化し、捕獲開始前の初期頭数を与えて資源頭数の推移を計算した。これと、得られている雌雄別体長別捕獲頭数にあてはめることによって、初期頭数とその後の資源頭数の推移を推定し、今後の資源の衰退を予測しようとするものである。

この方法は、体長組成データを活用して資源量推定を試みた点で、本研究の方法と目的を同じくするものである。しかし、再生産を決定論的な単純なモデルで与えている点に、本研究の方法と大きな違いが見られる。

クジラなど哺乳類の場合、1回に少数の個体しか出産せず、その後の死亡率は魚類に比べて低いため、再生産関係をモデルに入れることは、比較的容易である。しかし、このような状況は水産生物の中では特殊な状況であり、一般には安定した再生産関係は期待できない。

そのため本研究では、モデルは最高齢の個体数を決定してから順次若齢の個体数を推定する方法を用いている。こうすることによって再生産関係をモデルから排除することが可能になっている。

第 3 章

体長組成データによるマルチコホート解析

3.1 概要

まず、今回新たに開発した体長組成データによるマルチコホート解析法 (Length based Population Analysis/ LPA) の原理を説明する。

従来の年齢によるコホート解析は、「1年前は1歳若い」という明白な関係にもとづいて、最終年の資源個体数を与えてそこから1年間の漁獲個体数と自然死亡による減少分を足すことによって、前年の1歳若い年齢群の資源個体数を推定する方法を取っている。それに対して、LPAは今年の体長別資源個体数から前年の体長別資源個体数を推定する行列を作成して前年の体長組成を推定し、これに自然死亡分・漁獲死亡分を足すことによって、前年の体長別の資源個体数を推定する方法をとる。漁獲個体数データとのあてはまりがよい漁獲パターンを求めることによって、最終年の体長別資源個体数や最大体長での資源個体数をデータより求め、各年・各年級群間で整合性のとれた推定値を得ることができる。

3.2 必要とするデータ

LPAに必要なデータは以下の通りである。

- 自然死亡係数 M
- 完全加入体長階級 i_r
- 年 j 体長階級 i の漁獲量 $C_{i,j}$
- 体長の推移を表す行列 P

3.3 行列 P の推定

行列 P とは $j+1$ 年に体長階級 i_n に属している資源が、 j 年に体長階級 i_p に属していた確率を (i_p, i_n) 成分とする行列である。ここで体長階級幅は一定で d とする。また、自然死亡

や漁獲による個体の減少は、別途考慮するので、ここでは含めない。

行列 P を推定する方法について、ここでは2つの方法を提案する。方法1は、個体毎に、その個体の各齢での体長がわかる場合に適用できる。たとえば、貝殻の年輪をよんだり、耳石・鱗などの硬組織に年輪が刻まれ、年輪径と体長に一定の関係が認められる場合、標識放流などで2時点の体長がわかる場合などが、この例である。方法1が適用できるデータがとれる資源は限られるが、このデータがあると、成長の個体差が大きい場合でも精度良く推定できることが期待される。

方法2は、年齢-体長関係だけしかわからない場合に用いる。年齢査定が可能であっても、成長の履歴が追えない場合は方法2を用いる。もし、年齢査定が不可能な場合でも、体長組成を分解することによって年齢-体長関係を求めることができれば、適用することができる。しかし、成長の個体差が大きい場合は精度が悪くなることが予想される。

3.3.1 方法1

方法1は、以下の手順により直接遷移行列を求める方法である。

データとして、個体 k ($k = 1, 2, \dots, K$) の年齢 a ($a = 1, \dots, A_k$) における体長 $\ell_{k,a}$ が与えられているとする。ここで、 A_k は個体 k から得られる一番大きい体長に相当する年齢を表す。

1. 全ての個体 ($k = 1, 2, \dots, K$) について、 a 歳 ($a = 1, A_k - 1$) の時に属する体長階級 i_p ($i_p = 1, 2, \dots, I$) と $a + 1$ 歳の時に属する体長階級 i_n ($i_n = 1, 2, \dots, I$) を全ての年齢について調べる。体長 ℓ と体長階級 i の関係は

$$d \cdot (i - 1) \leq \ell < d \cdot i \quad (3.1)$$

とする。また、1歳時の体長については、 $i_p = 0$ として i_n に1歳時の体長階級をとる。

2. (i_p, i_n) の組を数え上げ、出現頻度 $f(i_p, i_n)$ とする。
3. 遷移行列の各成分 $p(i_p, i_n)$ ($i_p = 1, \dots, I, i_n = 1, \dots, I$) は

$$p(i_p, i_n) = \frac{f(i_p, i_n)}{\sum_{i_p=0}^I f(i_p, i_n)} \quad (3.2)$$

とする。なお、 $\sum_{i_p=1}^I f(i_p, i_n) = 0$ の場合は、 $p(i_p, i_n) = 0$ とする。

3.3.2 方法2

方法2は、以下の方法によって間接的に行列 P を求める方法である。データとして K 組の年齢 a_k と体長 ℓ_k ($k = 1, \dots, K$) が与えられているものとする。

1. 各齢毎の平均体長 $\bar{\ell}_a$ と標準偏差 s_a を求める。

2. $\bar{\ell}_a$ と標準偏差 s_a の直線回帰

$$\hat{s}_a = \alpha_s + \beta_s \bar{\ell}_a \quad (3.3)$$

を求める。

3. $\bar{\ell}_a$ と $\bar{\ell}_{a+1}$ の直線回帰

$$\hat{\ell}_a = \alpha_\ell + \beta_\ell \bar{\ell}_{a+1} \quad (3.4)$$

を求める。

4. 全ての体長階級 i について、体長階級の中央 $\ell_{i_p}^M$ を

$$\ell_{i_p}^M = (i_p - 0.5) \times d \quad (3.5)$$

で求める。

5. 前年の平均体長 $\bar{\ell}_{i_p}$ を

$$\bar{\ell}_{i_p} = \alpha_\ell + \beta_\ell \ell_{i_p}^M \quad (3.6)$$

として求める。

6. 前年の体長の標準偏差 s_{i_p} を

$$s_{i_p} = \alpha_s + \beta_s \bar{\ell}_{i_p} \quad (3.7)$$

として求める。

7. 体長階級 i_p の下限 $\ell_{i_p}^L$ と上限 $\ell_{i_p}^U$ を

$$\ell_{i_p}^L = d \times (i_p - 1) \quad (3.8)$$

$$\ell_{i_p}^U = d \times i_p \quad (3.9)$$

として求める。

8. 遷移行列の成分 $p(i_p, i_n)$ を、

$$p(i_p, i_n) = F(\ell_{i_p}^U) - F(\ell_{i_p}^L) \quad (3.10)$$

として求める。ここで $F(\cdot)$ は、平均 $\bar{\ell}_{i_p}$ 、標準偏差 s_{i_p} の正規分布の累積密度関数である。

3.4 個体数推定法

上述の行列 P を用いて、以下の手順で資源個体数の推定を行う。

1. 各データを読み込む。

2. 初期値として

$$\begin{aligned} f_j &= F_T \quad (j = 1, \dots, J) \\ s_i &= 1.0 \quad (i = 1, \dots, I) \end{aligned} \quad (3.11)$$

を与える。

3. 全ての i, j に対して、漁獲係数 $F_{i,j}$, 全死亡係数 $Z_{i,j}$ を以下のように計算する。

$$\hat{F}_{i,j} = f_j \times s_i \quad (3.12)$$

$$\hat{Z}_{i,j} = \hat{F}_{i,j} + M \quad (3.13)$$

4. 最終年 Y の体長別資源個体数 $\hat{N}_{i,J}$ を、漁獲個体数と漁獲係数・全死亡係数から推定する。

$$\hat{N}_{i,J} = \frac{C_{i,J} \hat{Z}_{i,J}}{(1 - e^{-\hat{Z}_{i,J}}) \hat{F}_{i,J}} \quad (3.14)$$

5. 同様に最大体長階級の資源個体数 $\hat{N}_{I,j}$ を推定する。

$$\hat{N}_{I,j} = \frac{C_{I,j} \hat{Z}_{I,j}}{(1 - e^{-\hat{Z}_{I,j}}) \hat{F}_{I,j}} \quad (3.15)$$

6. 最終年から順次、前年の体長階級別資源個体数を以下のように求める。

$$\hat{N}_{i,j} = e^{\hat{Z}_{i,j}} \sum_{i_n=1}^I p(i_n, i) \hat{N}_{i_n, j+1} \quad (3.16)$$

ここで、 $p(i_n, i)$ は行列 P の (i_n, i) 成分である。

7. 求められた各年の体長別資源個体数から、モデルから導出される漁獲個体数 $\hat{C}_{i,j}$ を以下によって計算する。

$$\hat{C}_{i,j} = \hat{N}_{i,j} (1 - e^{-\hat{Z}_{i,j}}) \frac{\hat{F}_{i,j}}{\hat{Z}_{i,j}} \quad (3.17)$$

8. モデルから導出される漁獲個体数とデータとして得られている漁獲個体数の差より、残差平方和 SS を求める。

$$SS = \sum_{j=1}^J \sum_{i=1}^I (\hat{C}_{i,j} - C_{i,j})^2 \quad (3.18)$$

9. SS が最小になるように非線形最適化を行ない、収束したときの体長別資源個体数と漁獲係数が、最終的な推定値である。

最適化法にはシンプレックス探索法⁴⁰⁾を用いた。計算は奥村⁴¹⁾のシンプレックス法最適化プログラム SMPLX を FORTRAN に書き換えたプログラムを用い、ワークステーション上で実行した。

第 4 章

妥当性の検討

4.1 目的

LPAによってどの程度の精度で資源量が推定できるのかをシミュレーションによって検討する。

精度検討の手順は以下の通りである (Fig. 4.1)。

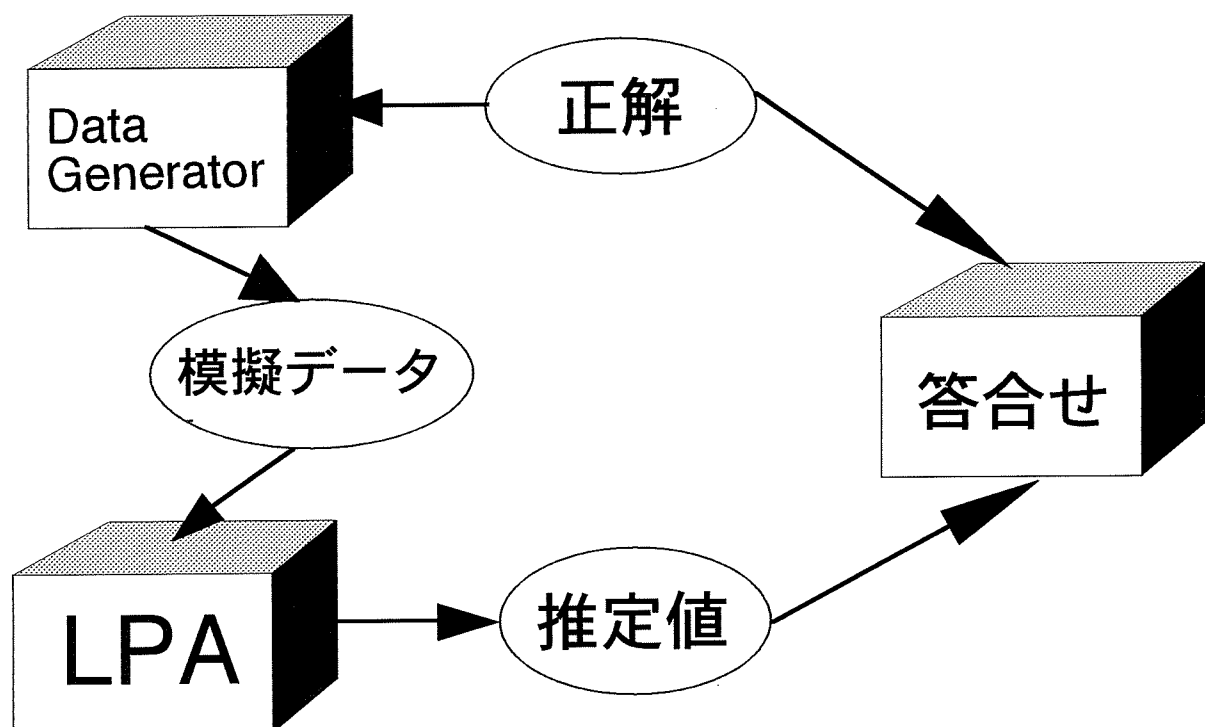


Fig. 4.1: シミュレーションによる精度評価の模式図

まず、現実的なパラメータを設定する。これを「正解」とする。次に、正解にしたがって、パラメータ推定に必要なデータを作る。これを「模擬データ」と言う。現実のデータに確率的な誤差が含まれる可能性のある部分には、正規乱数等を用いて誤差を導入する。この模擬データを推定過程にかけて、パラメータを推定する。

一つの正解に対し、いくつもの模擬データを作ることができる。それぞれの模擬データに対する推定値の平均値や分散を求め、正解と比べることによって、精度を評価する。

4.2 模擬データの作り方

4.2.1 パラメータの設定

模擬データを作成するデータジェネレータには、以下のパラメータを与える。

- 計算する年数 J
- シミュレーションで用いる仮想生物の最大寿命 A
- 平均加入量 $\bar{R}$
- 加入量の年変動の変動係数 CV_R
- 仮想生物の平均自然死亡係数 $\bar{M}$
- 自然死亡係数の個体変動の変動係数 CV_M
- 体長階級幅 d
- 体長階級数 I
- 年 j の完全加入体長における平均漁獲係数 $\bar{f}_j$
- $\bar{f}_j$ の変動係数 CV_f
- 各体長階級 i の平均選択率 $\bar{s}_i$
- 選択率の変動係数 CV_s
- 漁獲係数 $f_j \times s_i$ の変動係数 CV_F
- 成長曲線の平均パラメータ $\bar{L}_\infty, \bar{K}$
- 成長曲線のパラメータの変動係数 CV_{L_∞}, CV_K
- 成長曲線のまわりの変動のパラメータ $\alpha_\epsilon, \beta_\epsilon$
- 行列 P の計算方法 (方法1, 方法2)
- 行列 P 作成に用いる個体数 n

4.2.2 模擬データ作成の手順

模擬データ作成の手順は以下の通りである。。

まず、各パラメータを読み込む。

次に、漁獲係数 $F_{i,j}$ を以下の対数正規乱数によって与える。

$$F_{i,j} \sim LN(\tilde{F}_{i,j}, \tilde{F}_{i,j} \cdot CV_F) \quad (4.1)$$

ここで、

$$\tilde{f}_j \sim LN(\bar{f}_j, \bar{f}_j \cdot CV_f) \quad (4.2)$$

$$\tilde{s}_i \sim LN(\bar{s}_i, \bar{s}_i \cdot CV_s) \quad (4.3)$$

$$\tilde{F}_{i,j} = \tilde{f}_j \cdot \tilde{s}_i \quad (4.4)$$

である。

自然死亡係数 $M_{i,j}$ を以下によって計算する。

$$M_{i,j} \sim LN(\bar{M}, \bar{M} \cdot CV_M) \quad (4.5)$$

加入量 R_j を以下によって計算する。

$$R_i \sim N(\bar{R}, \bar{R} \cdot CV_R) \quad (4.6)$$

加入年 j ごと各個体 k ごとに成長曲線のパラメータ L_∞, K を以下により計算し、各年齢 a での体長を計算する。

$$L_{\infty,j,k} \sim N(\bar{L}_\infty, \bar{L}_\infty \cdot CV_{L_\infty}) \quad (4.7)$$

$$K_{j,k} = N(\bar{K}, \bar{K} \cdot CV_K) \quad (4.8)$$

$$(4.9)$$

成長曲線には von Bertalanffy の成長曲線を用いて、各個体の各齢での体長を以下のように与える。

$$\tilde{\ell}_{j,k,a} = L_{\infty,j,k}(1 - e^{-K_{j,k} \cdot a}) \quad (4.10)$$

$$\ell_{j,k,a} \sim N(\tilde{\ell}_{j,k,a}, \alpha_\epsilon + \beta_\epsilon \tilde{\ell}_{j,k,a}) \quad (4.11)$$

として与える。

計算開始前は、自然死亡だけによって減少していたと仮定し、各齢の個体数を $R_j e^{-M a}$ とする。死亡した個体数には、その齢以降の $\ell_{j,k,a}$ に負の値を入れて、体長階級毎に数えるときに数に入らないようにする。

計算開始年 $j = 1$ から順次、まず、体長階級 j に $\ell_{j,k,a}$ が含まれる個体数を数え上げて、漁獲前の個体数 $N_{i,j}$ を求める。

次に、漁獲死亡個体数を

$$C_{i,j} = N_{i,j}(1 - e^{-Z_{i,j}}) \frac{F_{i,j}}{Z_{i,j}} \quad (4.12)$$

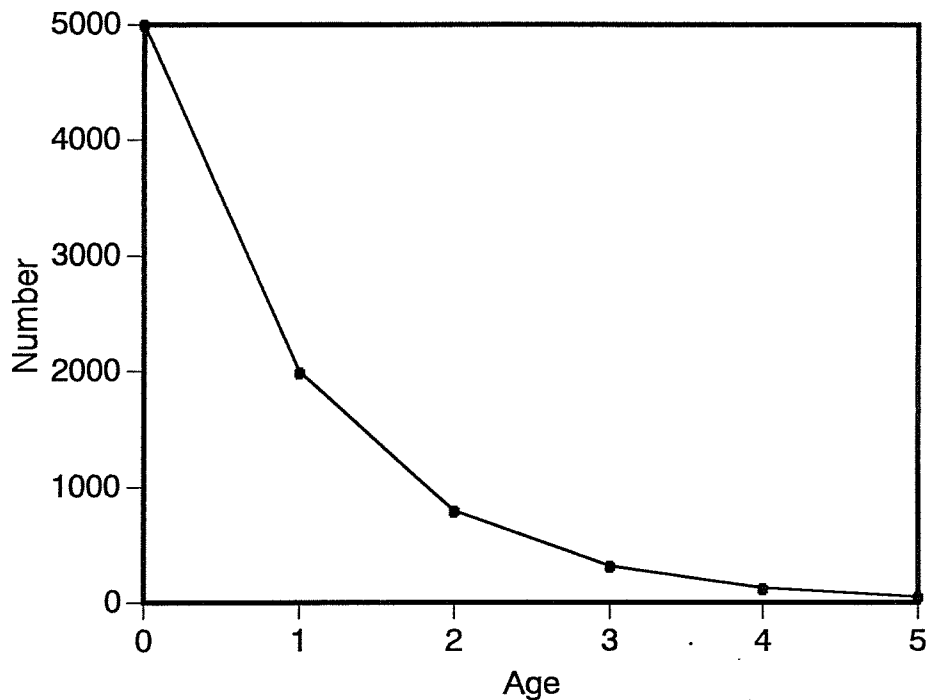


Fig. 4.2: シミュレーションに設定された個体数の推移。($F = 0.3$, $M = 0.62$)

として求める。

体長階級ごとの漁獲個体数・自然死亡個体数を、その年のその階級に含まれる年級群の比率に応じて比例配分し、死亡個体数分の l に負の値を入れる。

計算最終年まで計算したら、漁獲量 $C_{i,j}$ を出力する。また、推定過程が正しいかどうかを検証するために、真の個体数 $N_{i,j}$ 、真の漁獲係数 $F_{i,j}$ を出力する。

4.3 数値計算例

以上の方法によって得られたシミュレーションデータを LPA にかけて、推定精度を検討する。

4.3.1 パラメータ

誤差を含まない基本となるパラメータ (A 系列) は、以下のような生物と漁業形態を想定して設定した。

平均加入量 $\bar{R}$ は 5000 単位、個体数は自然死亡係数 $M = 0.62$ 、完全加入漁獲係数 $f = 0.3$ と設定した。なお、 $s = 1$ と固定し、漁業のサイズ選択性は無いものとした。最大年齢を 5 年とした。このように設定すると、5 年間で 99% の個体が死亡することになる。個体数の推移を Fig. 4.2 に示した。

生物は、von Bertalanffy の成長曲線に従って成長すると仮定した。体長の推移を Fig. 4.3 に示した。漁獲物の体長は幅 10 の体長階級に集計した。この体長組成 10 年分を用いた。

Table 4.1: データジェネレータに与えた誤差を含まないパラメータセット（A 系列）と誤差を含むパラメータセット (B 系列)

記号	A 系列	B 系列
J	10	10
A	5	5
$\bar{R}$	5000	5000
CV_R	0	0.05
$\bar{M}$	0.62	0.62
CV_M	0	0.05
d	10	10
I	10	10
$\bar{f}_j$	0.3	0.3
CV_f	0	0.05
$\bar{s}_i$	1.0	1.0
CV_s	0	0.05
CV_F	0	0.05
$\bar{L}_\infty$	130	130
$\bar{K}$	0.25	0.25
CV_{L_∞}	0	0.05
CV_K	0	0.05
α_ε	0	0
β_ε	0	0
行列の作成方法	方法 1	方法 1
n	100	100

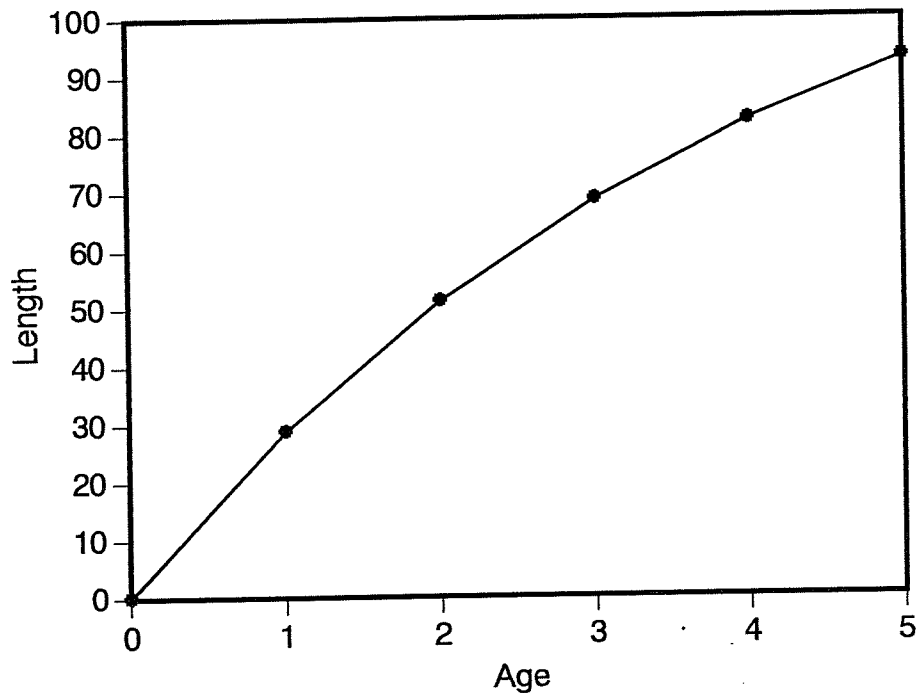


Fig. 4.3: シミュレーションに設定された個体の成長。($L_{\infty} = 130$, $K = 0.3$)

行列 P を作成する際の個体数は 100 個体とし、方法 1 によって行列 P を求めた。基本となるデータセットには、行列 P 作成の際のサンプリング誤差を除いて、誤差は導入していない。A 系列のパラメータを Table 4.1 に示した。

これに対して、 CV_R , CV_M , CV_f , CV_s , CV_F , $CV_{L_{\infty}}$, CV_K , β_e をそれぞれ、1%, 2%, 5%, 10%, 20% と変化させたもの、および α_e を 1, 2, 5, 10, 20 と変化させたもの、 n を 10, 20, 50, 100, 200 と変化させたデータセットを作成した。

データセット名は 4 文字からなり、最初の 1 文字は系列を示した。2 文字目は変化させたパラメータに対応し、 CV_R , CV_M , CV_f , CV_s , CV_F , $CV_{L_{\infty}}$, CV_K , α_e , β_e , n を変化させた物をデータセット名の 2 文字をそれぞれ、R, M, F, S, G, P, A, B, T とした。また、後ろの 2 文字は変化させた値に対応し、1(%), 2(%), 5(%), 10(%), 20(%) を、01, 02, 05, 10, 20 と表示した。たとえば、A 系列で $CV_F = 10\%$ としたデータセット名は AG10 である。

一方、多くのパラメータに誤差があらかじめ含まれているデータセット (B 系列) を作成した。このデータセットは、A 系列のパラメータと平均値は等しいが、全てのパラメータにあらかじめ CV 5% の変動を導入した。B 系列のパラメータを Table 4.1 に示した。

B 系列のデータに対しても、 CV_R , CV_M , CV_f , CV_s , CV_F , $CV_{L_{\infty}}$, CV_K , β_e をそれぞれ、1%, 2%, 5%, 10%, 20% と変化させたもの、および α_e を 1, 2, 5, 10, 20 と変化させたデータセット n を 10, 20, 50, 100, 200 と変化させたデータセットを作成した。B 系列のデータセット名も、A 系列のデータセットと同様に設定した。

4.3.2 精度の評価の仕方

データジェネレータから出力される真の個体数 $N_{i,j}$ と、LPA から得られた個体数推定値 $\hat{N}_{i,j}$ を比較して、推定精度を検討した。

データジェネレータで作成された 50 セットのデータセットに対して、LPA で個体数を推定し、その推定精度を相対誤差

$$\frac{\hat{N}_{i,j} - N_{i,j}}{N_{i,j}}$$

によって評価した。相対誤差の平均値・標準偏差・最小値、10%点、25%点、50%点、75%点、90%点および最大値を表に示した。

評価する推定値は、中間の年 $j = 5$ の体長階級 $i = 6$ を参考点とした。各シリーズ(たとえば AR01, AR02, AR05, AR10, AR20) について $j = 5, i = 6$ について評価した。また、「10」のデータ (AR10 など) については $j = 5, i = 1, \dots, 10$ および、 $i = 6, j = 1, \dots, 10$ についても評価した。

4.4 結果

4.4.1 誤差なしの基本データについて

誤差を導入しないデータセット (Table 4.1) を LPA にかけたところ、資源個体数推定値の相対誤差は 0.5% 以内と極めて小さく、誤差が無い場合には推定過程が正しいことが示された。

4.4.2 A 系列のデータについて

A 系列のデータセットの推定結果をボックスダイアグラム (Fig. 4.4~4.11) で示した。なお、第 5 年の推定結果と第 6 体長階級の推定結果については付録に付けた。

加入量 R を変化させた場合 (Fig. 4.4)、推定結果の偏りは小さく、加入量の変動が CV 10% 程度でも、推定値の偏りは最大 6% 程度であった。体長別の推定結果を見ると、体長が小さいほど、推定精度が良くなる傾向が見られた (付録参照)。年別の推定結果からは、年 j が小さいほど (すなわち年をさかのぼるほど)、精度が良くなる傾向が見られた (付録参照)。これは、 j が小さい方が、多くの漁獲データを用いることができるからであると考えられる。

自然死亡係数を変化させた場合 (Fig. 4.5)、CV 10% 以上の変動を与えると大きな偏りを持つ推定値が少数、生じることがあるが、CV 10% でも 90% 点が 0.063 であり、多くのデータセットで十分な精度の推定値が得られた。体長別の推定結果 (付録参照) には明確な傾向は見られなかった。年別推定結果 (付録参照) をみると、 j が小さいほど偏りが小さくなる傾向が見られたが、最終年 ($j = 10$) でも推定量の 50% 点は -10% 程度であった。

年ごとの漁獲係数 f を変化させた場合 (Fig. 4.6)、CV 20% の変動を与えると、10% を越える推定値の偏りが生じることがあるが、CV 20% のデータセットに対する推定値でも 90% 点が 0.114 と小さく、ほとんどのデータセットで十分な精度の推定値が得られた。また、推定

値の50%点は、CV20%のデータセットで-0.017と0に近く、平均的には偏りのない推定値が得られた。体長別の推定結果(付録参照)には明確な傾向は見られなかった。、年別推定結果(付録参照)をみると、 j が小さいほど偏りが小さくなり、精度が向上する傾向が見られたが、最終年($j = 10$)でも、推定量の50%点は-0.09程度であった。最初の年($j = 1$)では、50%点が0.045で、半数が相対誤差 ± 0.0012 以内であった。

選択率 s を変化させた場合(Fig. 4.7)、CV 20%の変動を与えると、推定値の10%点が-0.11, 90%点が0.36と推定精度が悪くなるが、CV 10%のデータセットの推定値は偏りが0.15を越える推定値はなかった。しかし、まれに推定結果が大きくなる結果が見られた。これは、あたかもある年級群だけを集中して漁獲しているかのような漁具選択性の変動が起こったためと考えられる。実際の漁業では、漁具や操業方法が年によって変化しない限りは、漁具のサイズ選択性の変動は少ないと考えられるので、この方法によって十分な推定精度が得られると考えられる。体長別の推定結果(付録参照)には体長が小さいほど精度が良くなる傾向が認められた。また、年別推定結果(付録参照)をみると、 j が小さいほど偏りが小さくなる傾向が見られたが、精度はかえって悪くなる傾向が見られた。

漁獲係数 F に直接変動を与えた場合(Fig. 4.8)、CV 20%の変動を与えると、推定値の10%点が-0.23, 90%点が0.32と推定精度が悪くなる。しかし、CV 5%の変動を与えた場合は、推定値の10%点が-0.07, 90%点が0.03と小さく、十分な推定精度が得られている。漁獲係数 F に変動を与えた場合は、 $F = f \cdot s$ (Separable仮定)という分解ができなくなり、推定過程がうまく働かなくなることが考えられる。しかし、CV 5%程度ならば問題なく推定できることを示唆している。体長別の推定結果(付録参照)には傾向が認められなかった。また、年別推定結果(付録参照)をみると、 j が大きいと著しく偏った推定値が出ることもあるが、小さいほど精度が良くなる傾向が見られた。

成長曲線のパラメータ L_{∞}, K に変動を与えた場合(Fig. 4.9)、CV 10%までの変動では、推定値の分散は大きくないが推定値の偏りが大きくなった。たとえば、CV 5%の変動を与えた場合は10%点と90%点の間は0.278であるが、50%点は-0.59と0から大きく離れた。体長別の推定結果(付録参照)をみると、第2体長階級で著しく精度が悪いことがわかる。この体長階級は、成長に関する変動を導入しない場合には、0歳から1歳になる間に通過して、漁獲物の中に入っていない階級である。ここで成長に関する変動を導入したため少数個体がこの階級に入るが、必ずしもこの階級から漁獲が無いので、その推定が困難であったためと考えられる。また、年別推定結果(付録参照)をみると、 j が大きいと著しく偏った推定値が出ることもあるが、小さいほど精度が良くなる傾向が見られた。

成長曲線のまわりに体長に依存しない変動を与えた場合(Fig. 4.10)、10を越える変動を与えると推定値が大きく偏った。しかし、年をさかのぼると、精度・偏りともに改善された(付録参照)。体長階級別の推定結果(付録参照)からは階級によって精度が悪いものが見られた。

成長曲線のまわりに体長に比例した変動を与えた場合(Fig. 4.11)も同様の推定結果を得た。この場合も、年をさかのぼると、精度・偏りともに改善された(付録参照)。体長階級別の推定結果(付録参照)からは階級によって精度が悪いものが見られた。

なお、A系列の基本データは成長に変動がないので、行列 P を作成する際に用いた個体数を変化させても、A系列の基本データを推定した結果と同様にほとんど偏りのない(0.005

以内)結果が得られた。

以上のように、成長に誤差が導入された場合、推定値の偏りが大きくなったり精度が悪くなったりすることがあるが、それ以外の場合では総じて安定した結果を得た。また、ほとんどの場合、年をさかのぼるに従って精度が向上した。

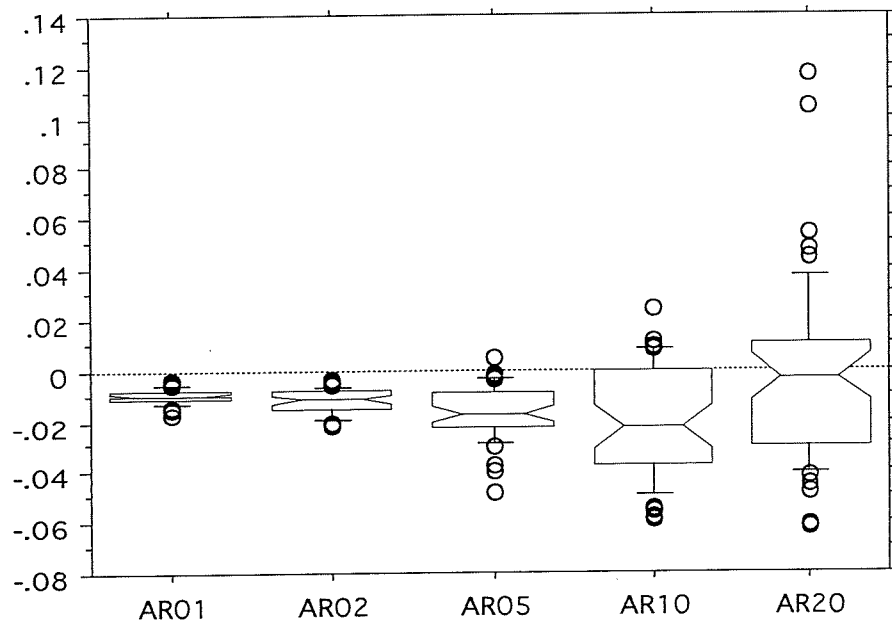


Fig. 4.4: AR系列の第5年目、第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

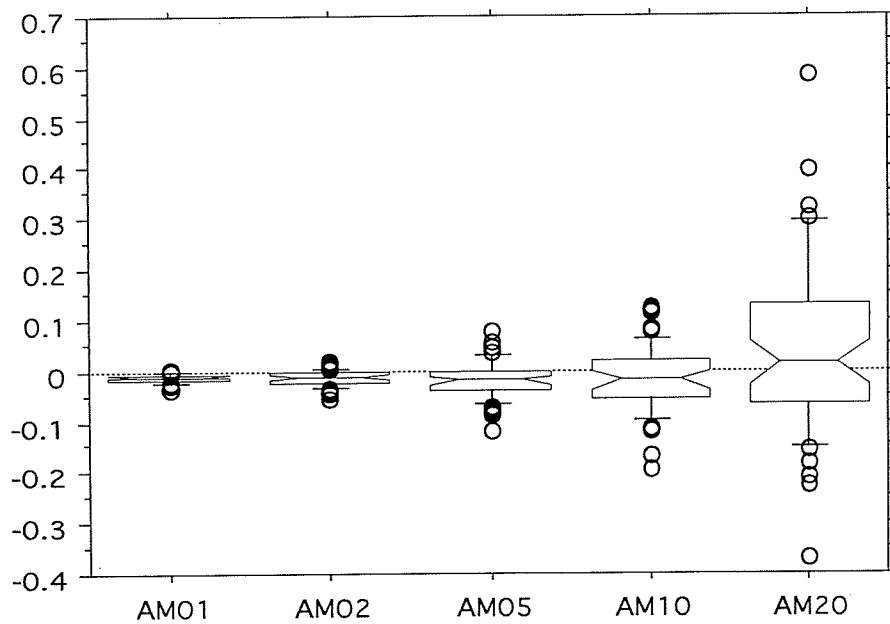


Fig. 4.5: AM系列の第5年目、第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

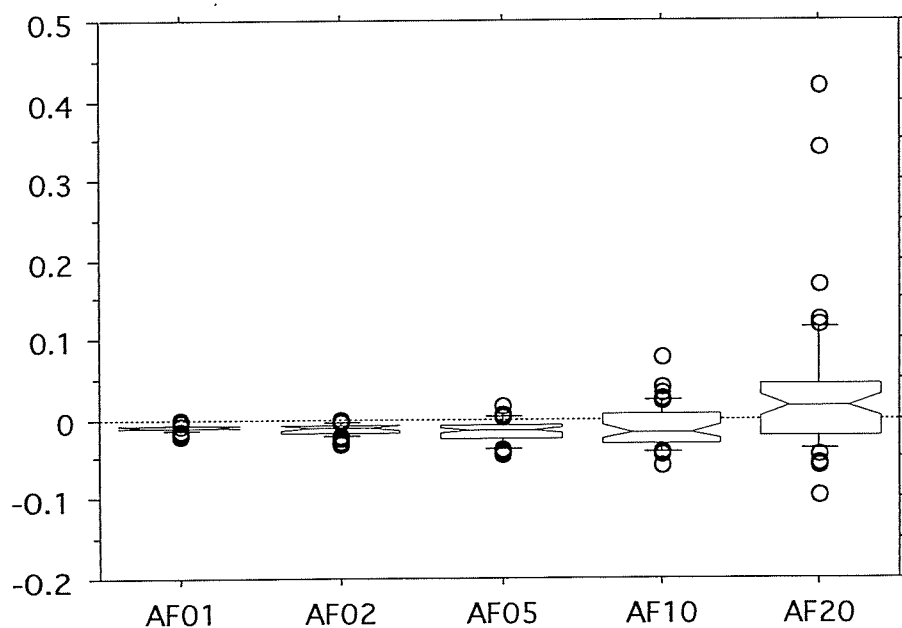


Fig. 4.6: AF系列の第5年目、第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

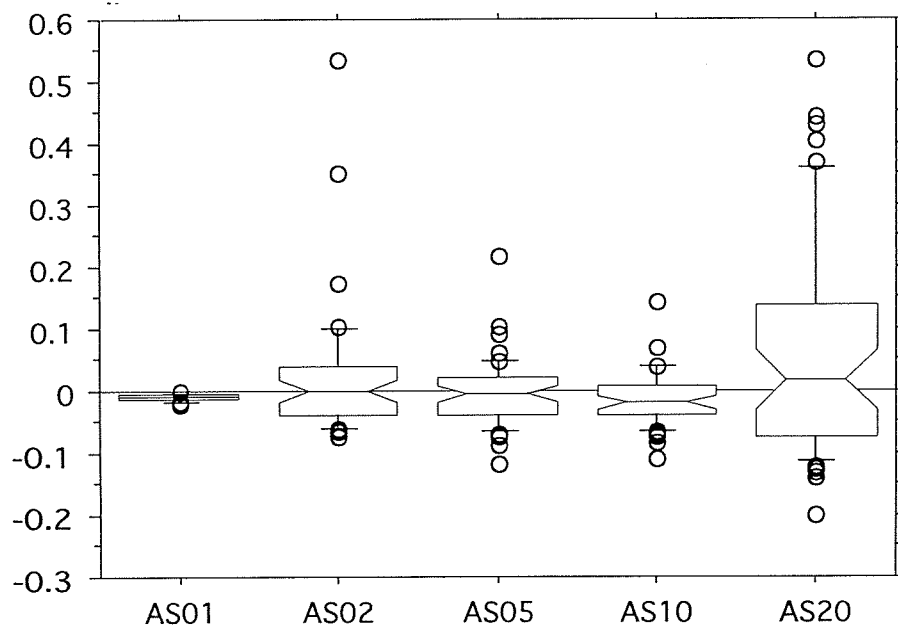


Fig. 4.7: AS系列の第5年目、第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

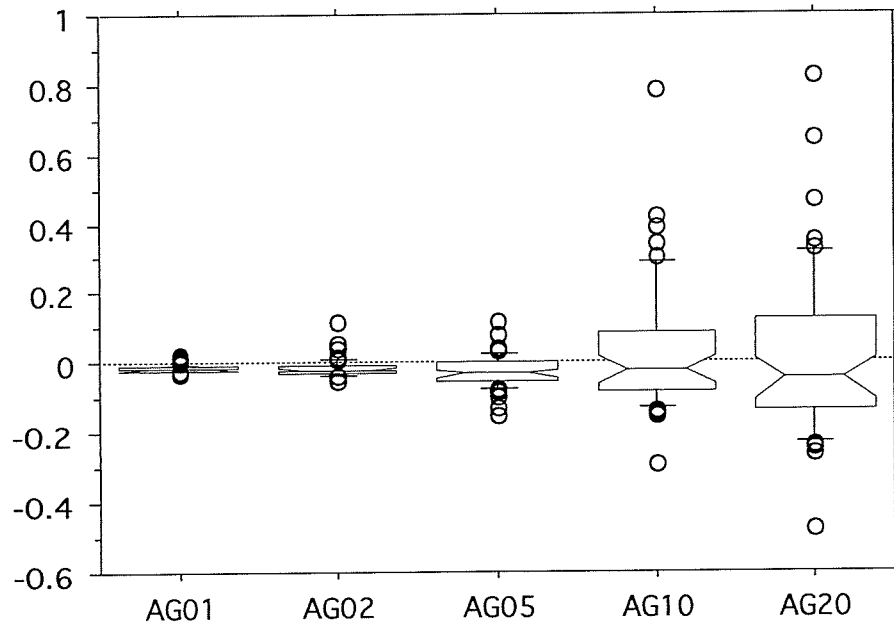


Fig. 4.8: AG 系列第 5 年目、第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

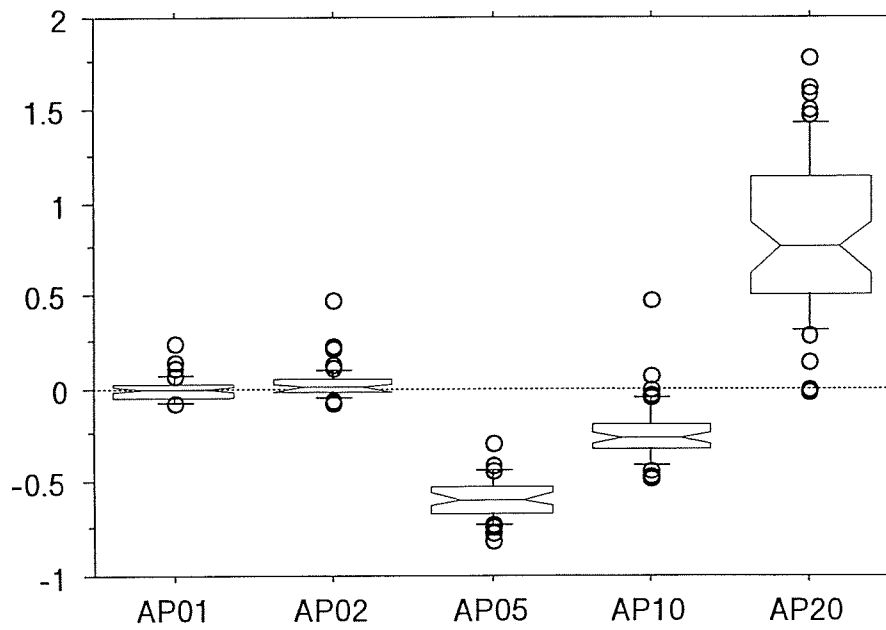


Fig. 4.9: AP 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

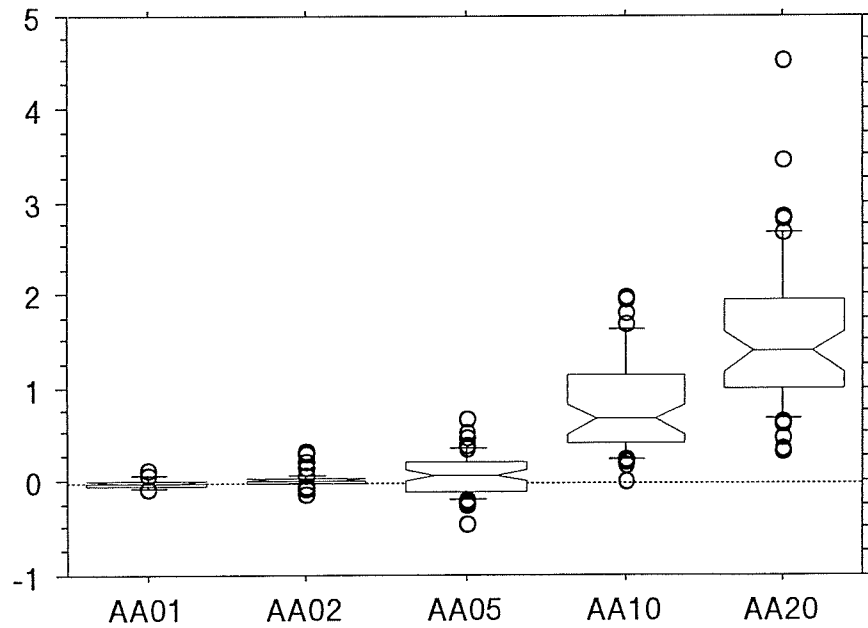


Fig. 4.10: AA 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

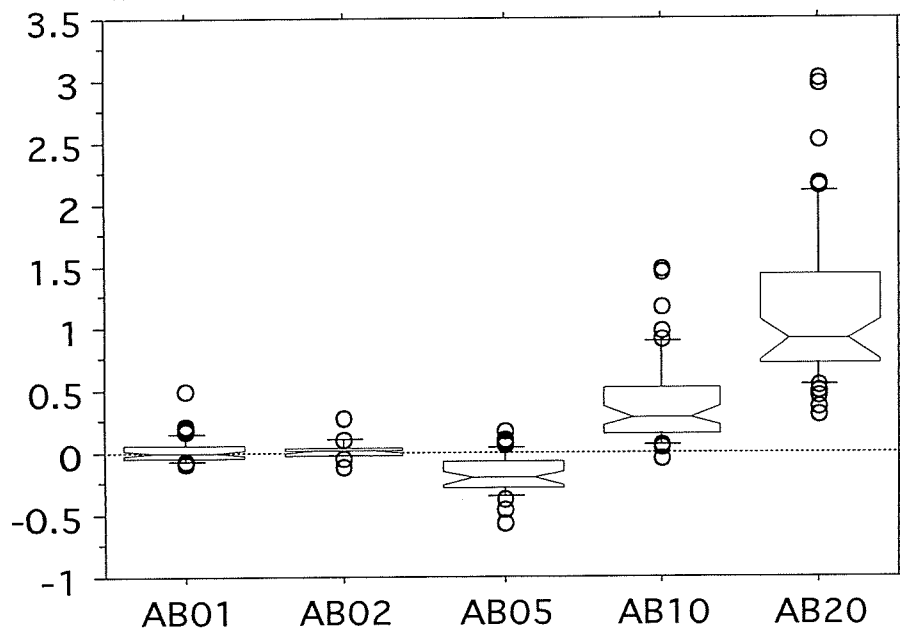


Fig. 4.11: AB 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

4.4.3 B系列のデータについて

B系列のデータセットの推定結果のボックスダイアグラムをFig. 4.12～4.20に示した。B系列は、全てのパラメータにCV 5%の変動(ただし、 $\alpha_e = 5$)があらかじめ導入されている。そのため、複合的に誤差が含まれている場合における、特定のパラメータの影響を調べることができる。

B系列の基礎となる模擬データを推定にかけたところ、平均値が-0.58と偏っていた。この偏りがどのパラメータの変動によるものなのかについても、以下の解析で検討する。

R , M , f , s , F の変動係数を変化させたBR(Fig. 4.12), BM(Fig. 4.13), BF(Fig. 4.14), BS(Fig. 4.15), BG(Fig. 4.16)については、得られる推定値の偏りや精度は同様であり、これらのパラメータが推定精度に与える影響は少ないと考えられる。

変化させているパラメータを10にした場合の年ごとの推定結果(付録参照)についても同様の推移を示し、 $j = 10$ では過大評価をしており推定値の分散も大きい、年をさかのぼるに従って偏り、精度ともに減少している。また、 $j = 6$ 以前では過小評価になっている。

変化させているパラメータを10にした場合の体長階級ごとの推定結果(付録参照)は、 $i = 11$ 以外では過小評価をしているが、 i が小さくなるにつれて、偏りは少なくなってくる傾向が見られる。

p の変動係数を変化させた場合(Fig. 4.17)は、推定値の50%点がCV 1%の時には0.02と小さいのに対し、CV 10%では-0.29, CV 20%では-0.42と急激に大きくなった。 p の変動が推定値の偏りに影響していることが示唆された。これは、成長の変動が少ない場合にはほとんど該当する個体のいない体長階級に、変動が大きい場合は確率的に個体が入り、そのような体長階級に対する行列 P の値が適切ではないことが原因であると考えられる。

年別の推定結果(付録参照)は、年をさかのぼるに従って偏り、精度とも大幅に改善されていることが示された。また、体長階級別の推定結果(付録参照)では、変動が無い場合には該当する個体のいない第2体長階級に、成長の変動の結果から少数の個体が入ったため、第2体長階級の推定結果が大きくはずれている。しかし、その他の体長階級では50%点は最大0.6偏るにすぎない。

a_e , b_e を変化させたとき(Fig. 4.18, 4.19)にも同様の傾向が見られ、成長に関する変動が推定値の偏りに影響を及ぼすことが示唆された。

行列 P を作成する際の個体数 n を10, 20, 50, 100, 200と変えた場合(標準は100)の推定結果をFig. 4.20に示した。個体数が10, 20の場合は精度も悪く、推定値も大きく偏っている。 $n = 10$ (BT01)では25%、 $n = 20$ (BT20)では10%を越える。 $i = 6, j = 5$ の推定結果が-1であったが、これは実際にはこの階級に個体数があるにもかかわらず、個体数を0と推定したことを意味している。しかし、 $n = 50$ (BT05)では精度は大幅に改善され、 $n = 100$, $n = 200$ (BT10, BT20)では、ほとんど性能に差が見られなかった。ある程度以上の個体数を用いれば、十分な推定ができることが示唆された。

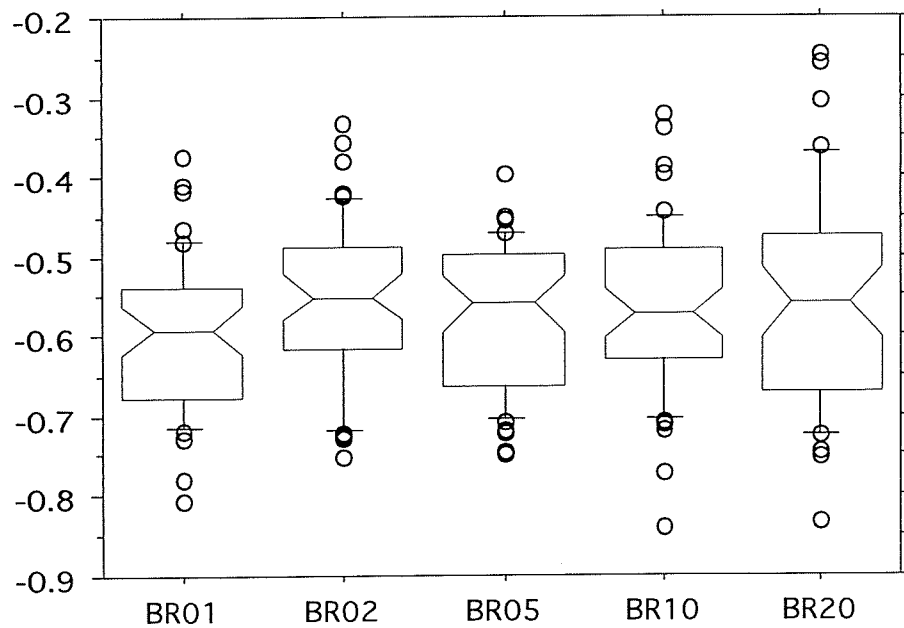


Fig. 4.12: BR 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10% に含まれる値、棒の端は両端 10% 点、箱の端は両端 25% 点、くびれの中央は 50% 点を表す。

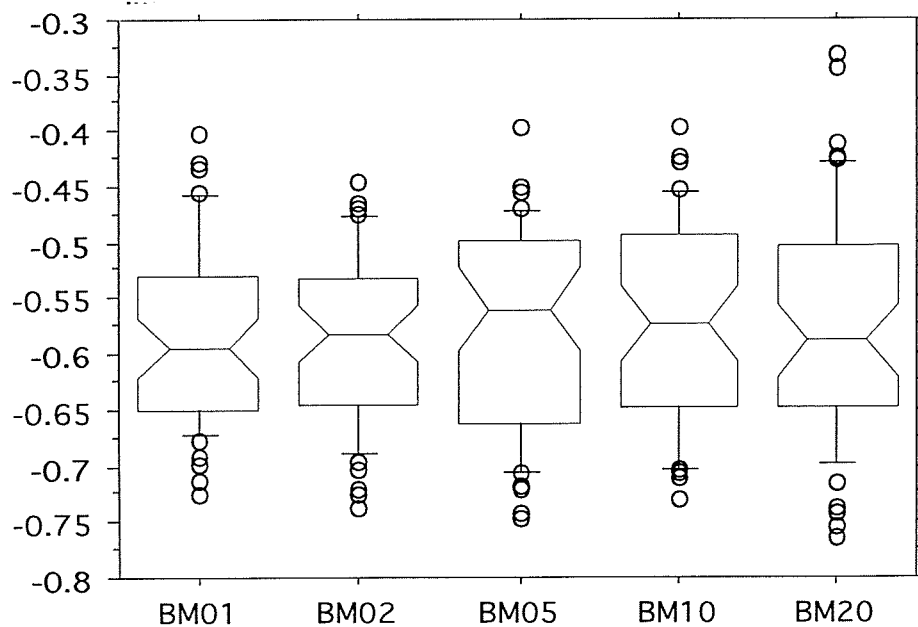


Fig. 4.13: BM 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10% に含まれる値、棒の端は両端 10% 点、箱の端は両端 25% 点、くびれの中央は 50% 点を表す。

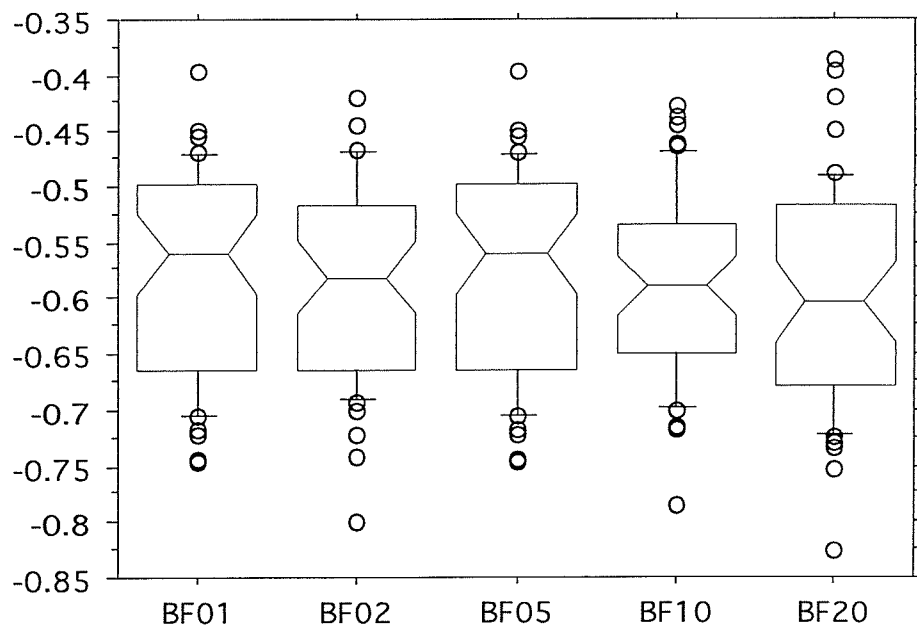


Fig. 4.14: BF 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10% に含まれる値、棒の端は両端 10% 点、箱の端は両端 25% 点、くびれの中央は 50% 点を表す。

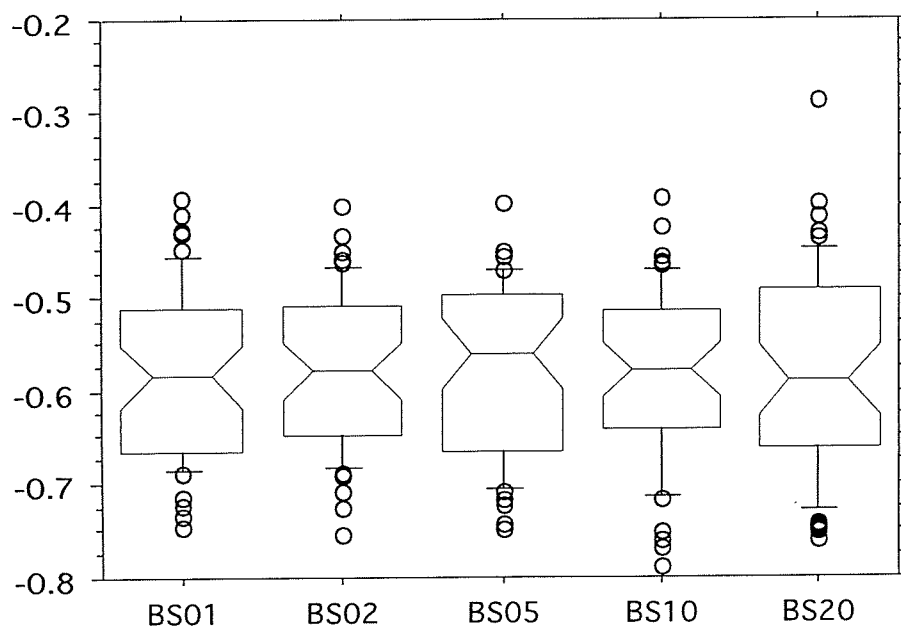


Fig. 4.15: BS 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10% に含まれる値、棒の端は両端 10% 点、箱の端は両端 25% 点、くびれの中央は 50% 点を表す。

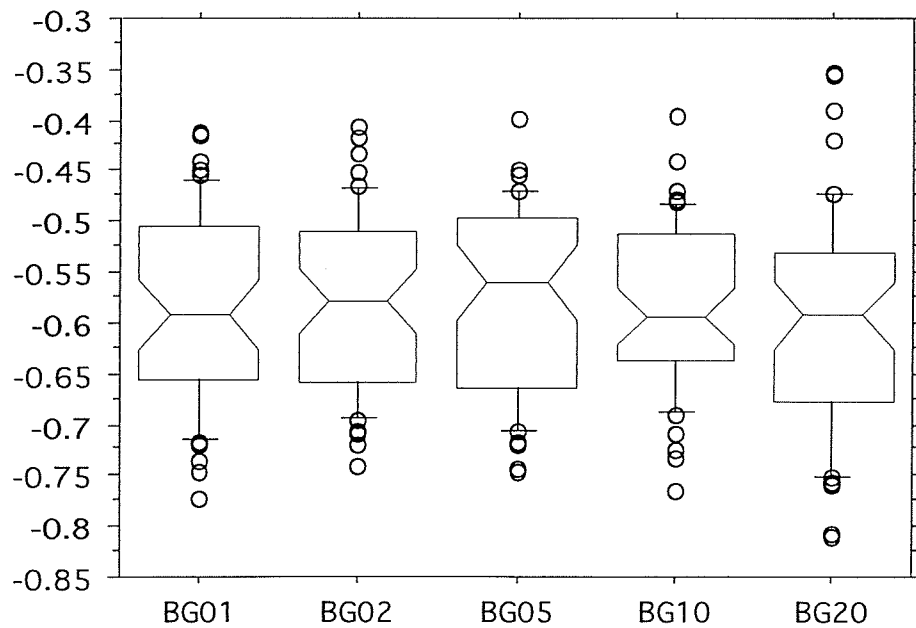


Fig. 4.16: BG 系列第 5 年目、第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

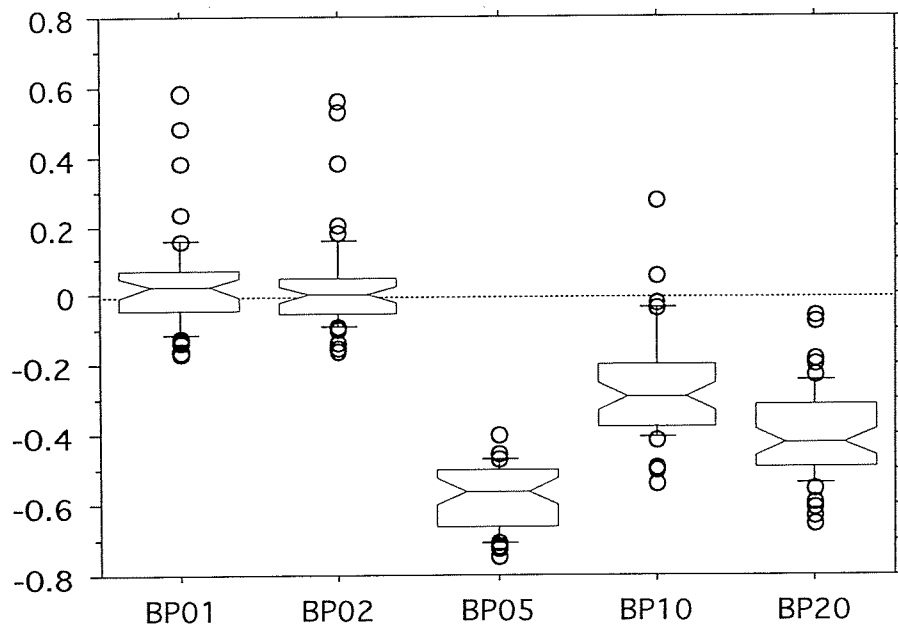


Fig. 4.17: BP 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

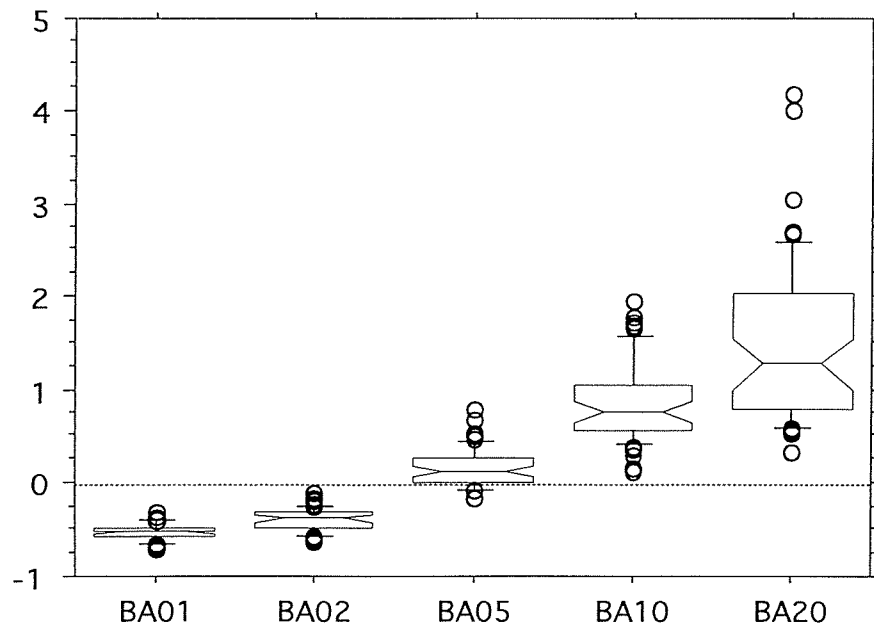


Fig. 4.18: BA 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

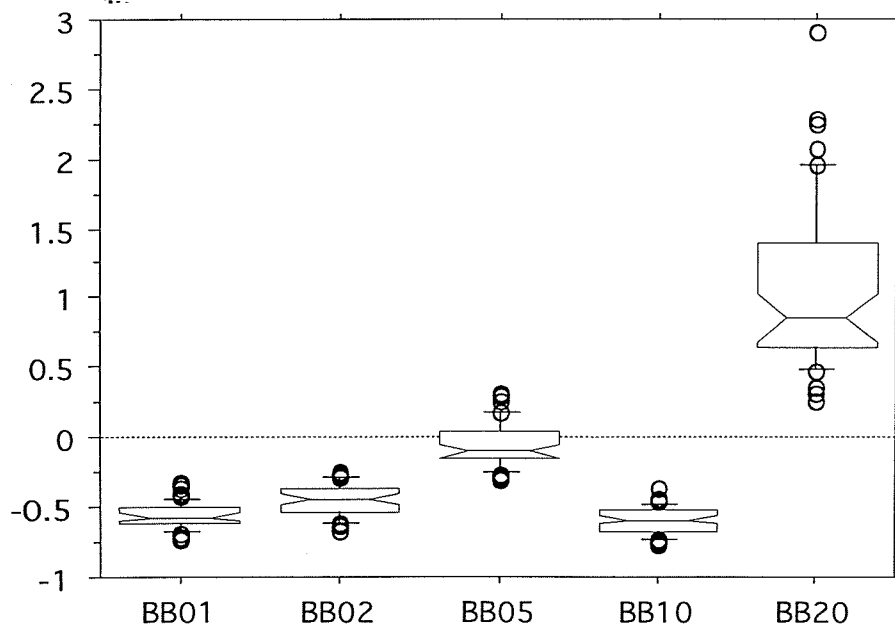


Fig. 4.19: BB 系列の第 5 年目、第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

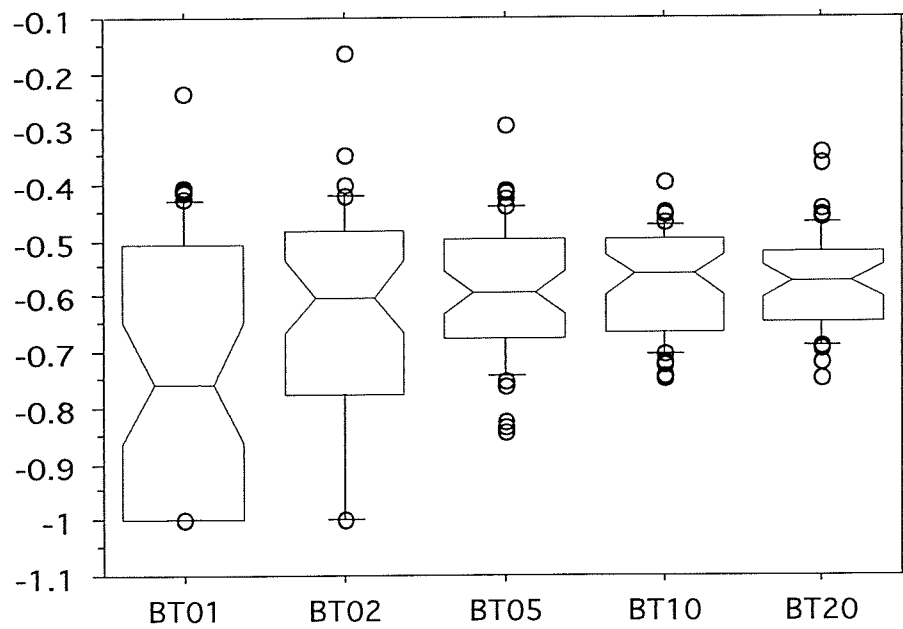


Fig. 4.20: BT系列の第5年目、第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

第 5 章

従来の方法との比較

5.1 目的

本研究では、体長別漁獲個体数から体長別資源個体数を推定した。しかし、体長別資源個体数は従来の方法を組み合わせることによっても得ることが可能である。そこでここでは、従来のコホート解析などの方法を組み合わせて使用した場合の推定精度と本研究による方法による推定精度の比較を行った。

5.2 方法

従来行われているコホート解析は、年齢別漁獲個体数から年齢別資源個体数を推定するものである。したがって、体長別漁獲個体数をデータに用いる場合、いったん年齢別漁獲個体数に換算する必要がある。この方法には年齢査定が可能な場合は Age-Length Key(ALK)、年齢査定をしていない場合は体長組成分解(LFA)が一般に用いられる。

ここでは、ALK, LFA とも FORTRAN のサブルーチンを作成し、計算に用いた。ALK のプログラムは、田中²⁰⁾をもとにして独自に FORTRAN でプログラミングした。LFA については Hasselblad²²⁾を元に赤嶺・加藤によって作成された BASIC プログラム³⁰⁾を FORTRAN に翻訳して用いた。

コホート解析には、マルチコホートを含めていくつもの小修正を施したモデルがあるが、ここでは Pope²⁾の VPA をもとにし、島本が作成した BASIC プログラム⁴²⁾を FORTRAN に翻訳して用いた。

コホート解析によって得られた年齢別資源個体数を体長別資源個体数へ変換する際は、各年齢の個体群の体長が正規分布にしたがうと仮定し、以下の式にもとづいて計算した。

$$N_{i,j}^{\ell} = \sum_{a=1}^J N_{a,j}^a \int_{i \cdot d}^{(i+1) \cdot d} f(\bar{\ell}_a, s_a^2) d\ell \quad (5.1)$$

ここで、 $N_{i,j}^{\ell}$ は、年 j 、体長階級 i の資源個体数、 $N_{a,j}^a$ は、年 j 、年齢 a の資源個体数、 d は体長階級幅、 $\bar{\ell}_a, s_a^2$ は、年齢査定や漁獲物の体長組成分解によって得られた年齢 a の個体の

Table 5.1: 精度比較に用いたデータセットのパラメータ

J	10
A	5
$\bar{R}$	5000
CV_R	0.05
$\bar{M}$	0.62
CV_M	0.05
d	10
I	10
$\bar{f}_j$	0.3
CV_f	0.05
$\bar{s}_i$	1.0
CV_s	0.05
CV_F	0.05
$\bar{L}_\infty$	130
$\bar{K}$	0.25
CV_{L_∞}	0.025
CV_K	0.025
α_ε	0
β_ε	0
行列の作成方法	方法 1
n	100

平均体長と標準偏差、 $f(\bar{\ell}_a, s_a^2)$ は、平均 $\bar{\ell}_a$ 、分散 s_a^2 の正規分布の密度関数である。

5.3 パラメータ

計算に用いた模擬データの作成には、Table 5.1のパラメータを用いた。

データジェネレータで50組のデータセットを作成し、LPA, LFA+VPA, ALK+VPA の3通りの方法で、50セットの推定値を得た。それぞれに対して、5年目および第6体長階級の推定値に対して、真の個体数との差(%)を計算し、その10%点,25%点,50%点,75%点,90%点をボックスプロットにより示した。

5.4 結果

5年目第6体長階級の結果を Fig. 5.1, Table 5.2に示した。ALK+VPAを用いた場合、50%点が0.1423、LFA+VPAを用いた場合は50%点が−0.0987と0からはずれたのに対し、LPA

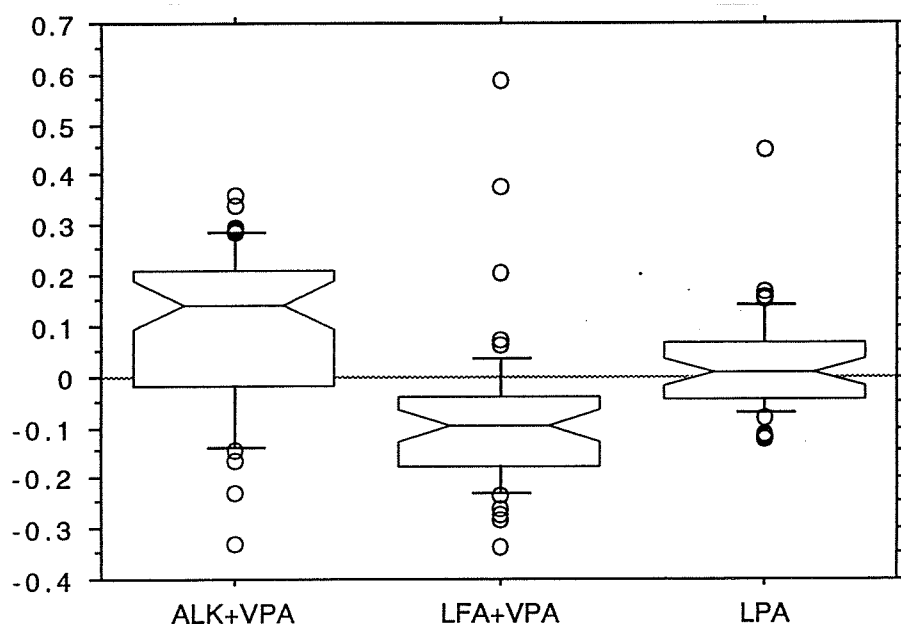


Fig. 5.1: 模擬データ 50 組に対して、ALK+VPA, LFA+VPA, LPA の 3 種類の方法で推定した体長別個体数 (5 年目第 6 体長階級) の相対誤差

を用いた場合は 50% 点が 0.0095 と、1桁小さい値を示し、従来の方法よりも偏りが小さいことが示された。

また、推定値の標準偏差も ALK+VPA が 0.1599、LFA+VPA が 0.1561 に対し、LPA は 0.0977 となり、LPA の精度が ALK+VPA, LFA+VPA よりも高いことが示された。

5 年目の、第 1 体長階級から第 10 体長階級に対する、それぞれの方法の相対誤差の分布を Fig. 5.2~5.4 に示した。ALK+VPA は体長が大きくなるに従って過大評価、LFA+VPA は過小評価をする傾向にあるのに対し、LPA は階級によっては偏りに傾向が無い。また、50% 点は ± 0.1 以内に収まっている。

第 6 体長階級の 1 年目から 10 年目に対する、それぞれの方法による推定値の相対誤差の分布を Fig. 5.5~5.7 に示した。ALK は常に 50% 点が 0.1 をこえ、過大に偏って推定されている。LFA + VPA は 1 年目から 8 年目までは 50% 点が -0.1 前後であるが、9 年目は正に転じ、10 年目は 0.7 と大きく過大評価している。LPA は 1 年目から 8 年目まで 50% 点が ± 0.06 に収まり、一番偏った 10 年目の結果でも 50% 点は -0.1485 と、他の方法より小さい。

ALK を用いる場合は、100 個体の年齢査定を行っている。年齢査定における誤差は無いものとしているので、漁獲物の年齢組成に含まれる誤差は、サンプリング誤差によるものである。ALK+VPA による推定値が偏っていて、分散も大きいのは、加入量に対する年齢査定個体数の割合が少ないことと、年齢-体長間の変換を 2 回行っているためであると考えられる。

LFA を用いる場合は、全漁獲物についての体長のデータが正確に得られていることを想

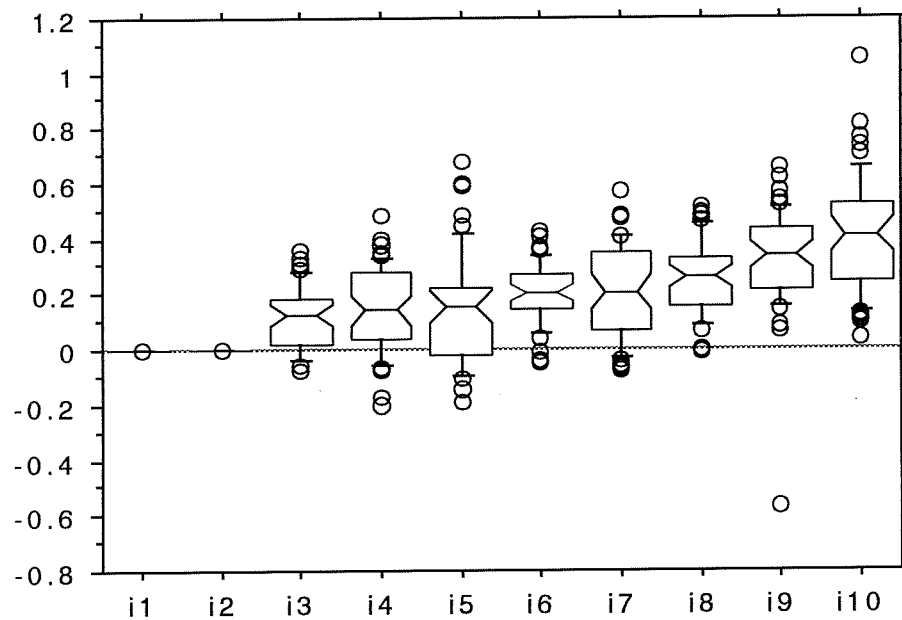


Fig. 5.2: 模擬データ 50 組に対して、ALK+VPA で推定した体長別個体数 (5 年目) の相対誤差

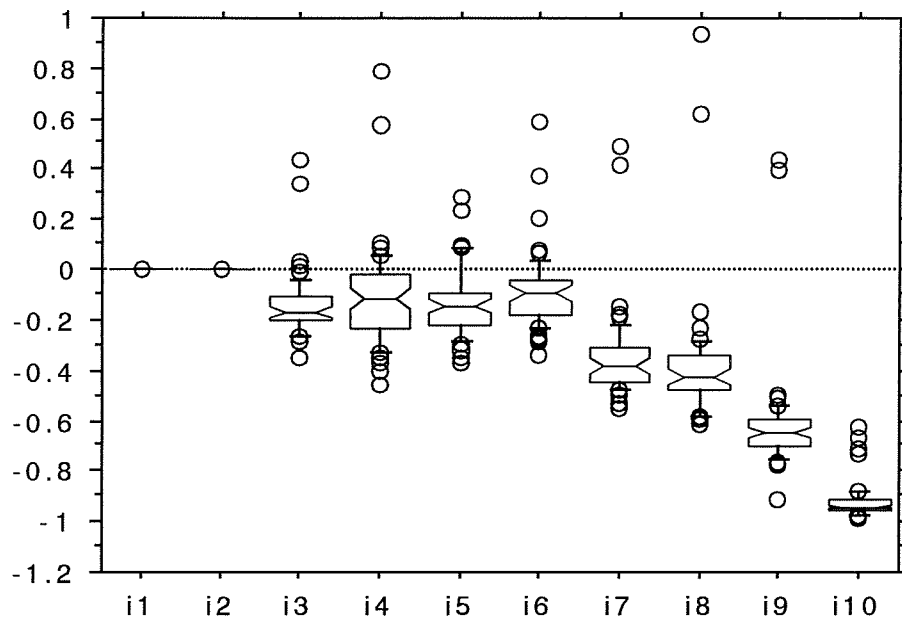


Fig. 5.3: 模擬データ 50 組に対して、LFA+VPA で推定した体長別個体数 (5 年目) の相対誤差

Table 5.2: 模擬データ 50 組に対して、LPA, ALK+VPA, LFA+VPA の 3 種類の方法で推定した体長別個体数 (5 年目第 6 体長階級) の相対誤差の諸統計値

	ALK+VPA	LFA+VPA	LPA
mean	0.1002	-0.0864	0.0246
s.d.	0.1599	0.1561	0.0977
min	-0.3322	-0.3360	-0.1240
10%	-0.1413	-0.2292	-0.0730
25%	-0.0155	-0.1793	-0.0440
50%	0.1423	-0.0987	0.0095
75%	0.2092	-0.0378	0.0670
90%	0.2839	0.0375	0.1410
max	0.3576	0.5881	0.4500

定しているので、サンプリング誤差は導入されない。しかし、体長組成を年齢組成に分解する際に大きなばらつきが生じ、推定精度が悪くなっていると考えられる。

以上のように、体長別漁獲物個体数から体長別個体数を推定する際には、Age-Length Key や体長組成分解を用いるよりも LPA を用いた方が、精度よく偏りの少ない結果を得ることができた。

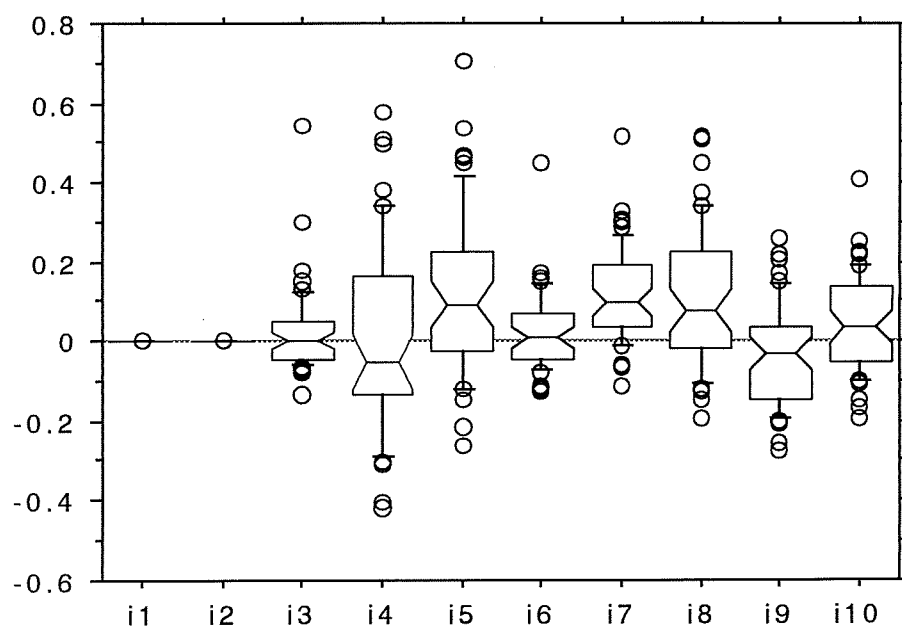


Fig. 5.4: 模擬データ 50 組に対して、LPA で推定した体長別個体数 (5 年目) の相対誤差

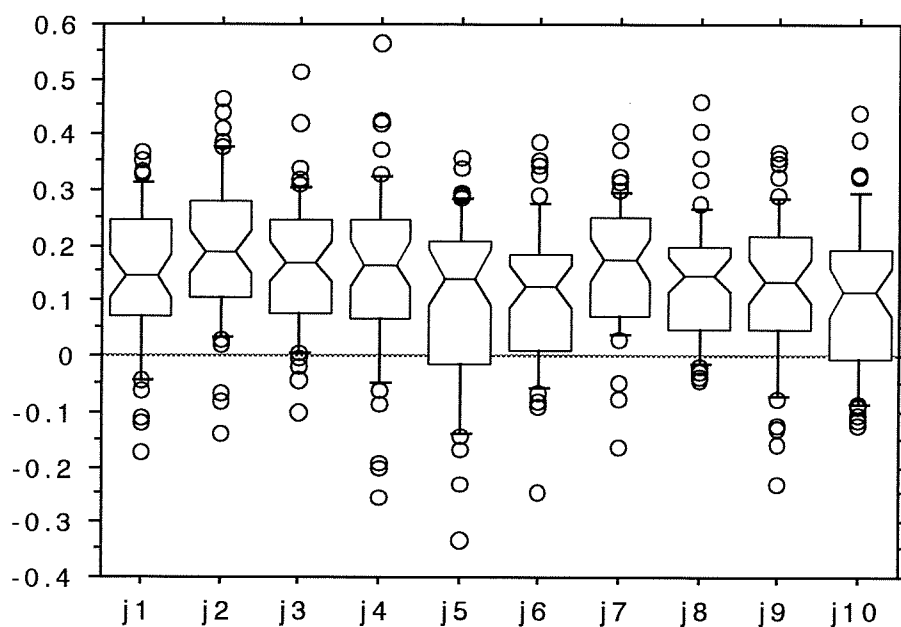


Fig. 5.5: 模擬データ 50 組に対して、ALK+VPA で推定した体長別個体数 (第 6 体長階級) の相対誤差

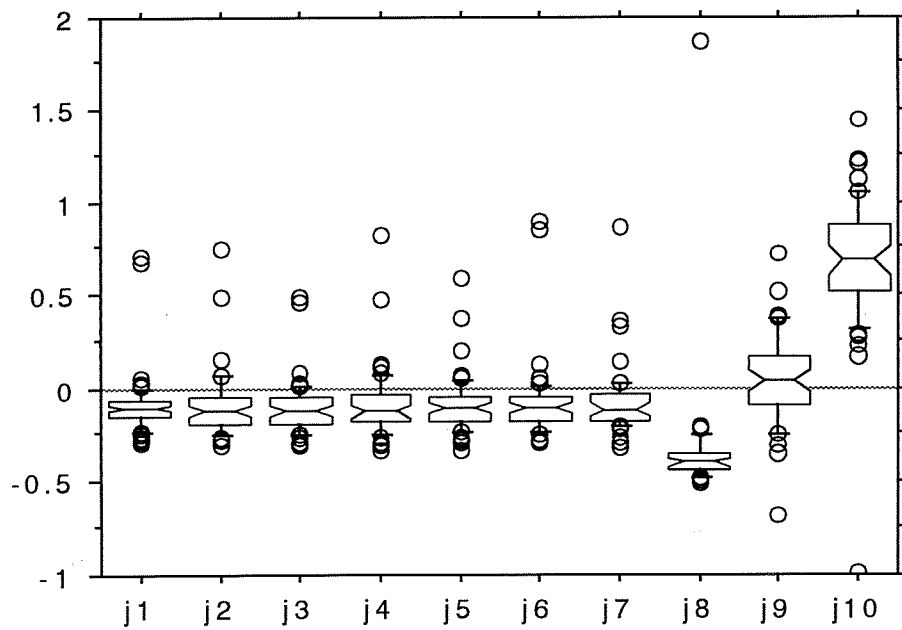


Fig. 5.6: 模擬データ 50 組に対して、LFA+VPA で推定した体長別個体数 (第 6 体長階級) の相対誤差

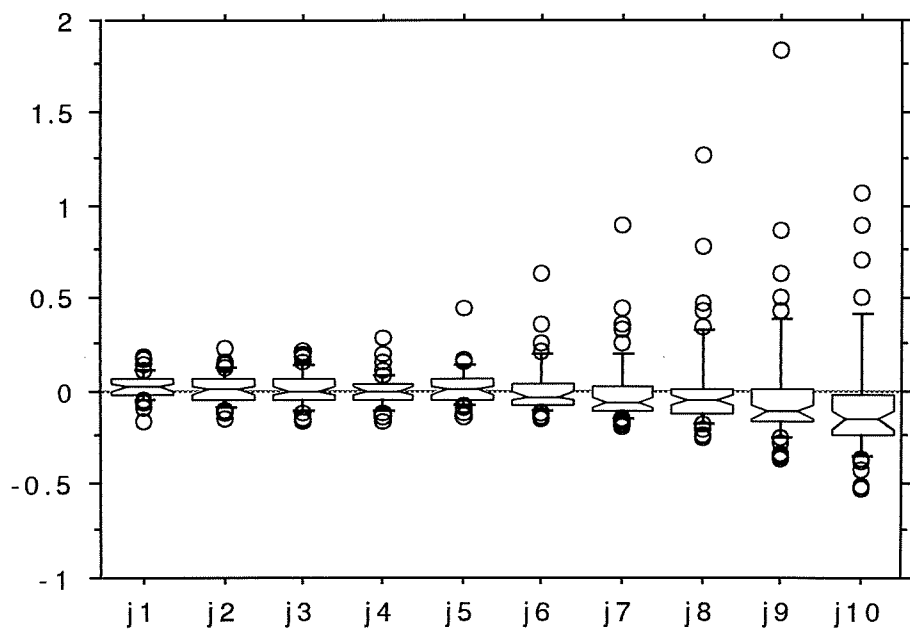


Fig. 5.7: 模擬データ 50 組に対して、LPA で推定した体長別個体数 (第 6 体長階級) の相対誤差

第 6 章

事例解析 1: 秋田県南部におけるアワビ漁業

ここでは、LPA が実際の漁業のデータに対して有効であるかどうかを示すために、2種類の漁獲統計資料を用いて体長別資源量の推定を試みる。

6.1 漁業の概要

まず、秋田県南部に位置する象潟・金浦地区 (Fig. 6.1) のアワビ漁業を扱う。

秋田県南部地区（金浦・象潟地区）は沖合を暖流が流れ、なだらかな海岸線の所々に磯が点在している。毎年7月と8月は素潜りによる磯根漁業が解禁になり、アワビ・ナマコ・サザエ・ウニ・カキ・モズク等の好漁場である。

秋田県南部地区におけるアワビの漁獲量を Fig. 6.2 に示す。1960年代より急激に漁獲量が増加し、1969・1970年には9500kgを越える漁獲量をあげた。その後1983年までは徐々に減少しながらも6~10tの範囲で安定していた。その後減少し、1985年には過去25年間の最低値である3224kgを記録したが、エゾアワビの移植・放流のためか、その後漁獲量は好転してきている。

この地区で行われるアワビ漁業は、以下のような管理規則のもとに行われる。漁期は、7月・8月の2カ月で、7月は金浦地区・象潟地区のそれぞれで地先のアワビを漁獲する、8月は隣の金浦地区と象潟地区の入り会いになり、象潟の漁業者が金浦に漁獲に来ることもある。漁獲は素もぐりにより行われ、殻長10cm以下のアワビの漁獲は禁止されている。また、年毎に1日に漁獲できる個数を規定している。

漁業者の資源管理意識は大変高く、漁業協同組合を通さない漁獲物の流通は無いと考えられ、漁獲量データの信頼性は高い。また、漁場の沿岸には人家があるため、外部からの密漁も起こりにくい。したがって、この地区の漁獲量データは信頼性が高いと考えられる。

アワビは殻を白熱電球にあてて見る方法⁴³⁾によって、年輪に相当する輪紋を読むことができ、これによってこの個体の成長の履歴を知ることができる。輪紋の判読の際には、不鮮明な輪紋を読み飛ばしてしまうため、1齢ずつずれた年齢-体長データが生じることが

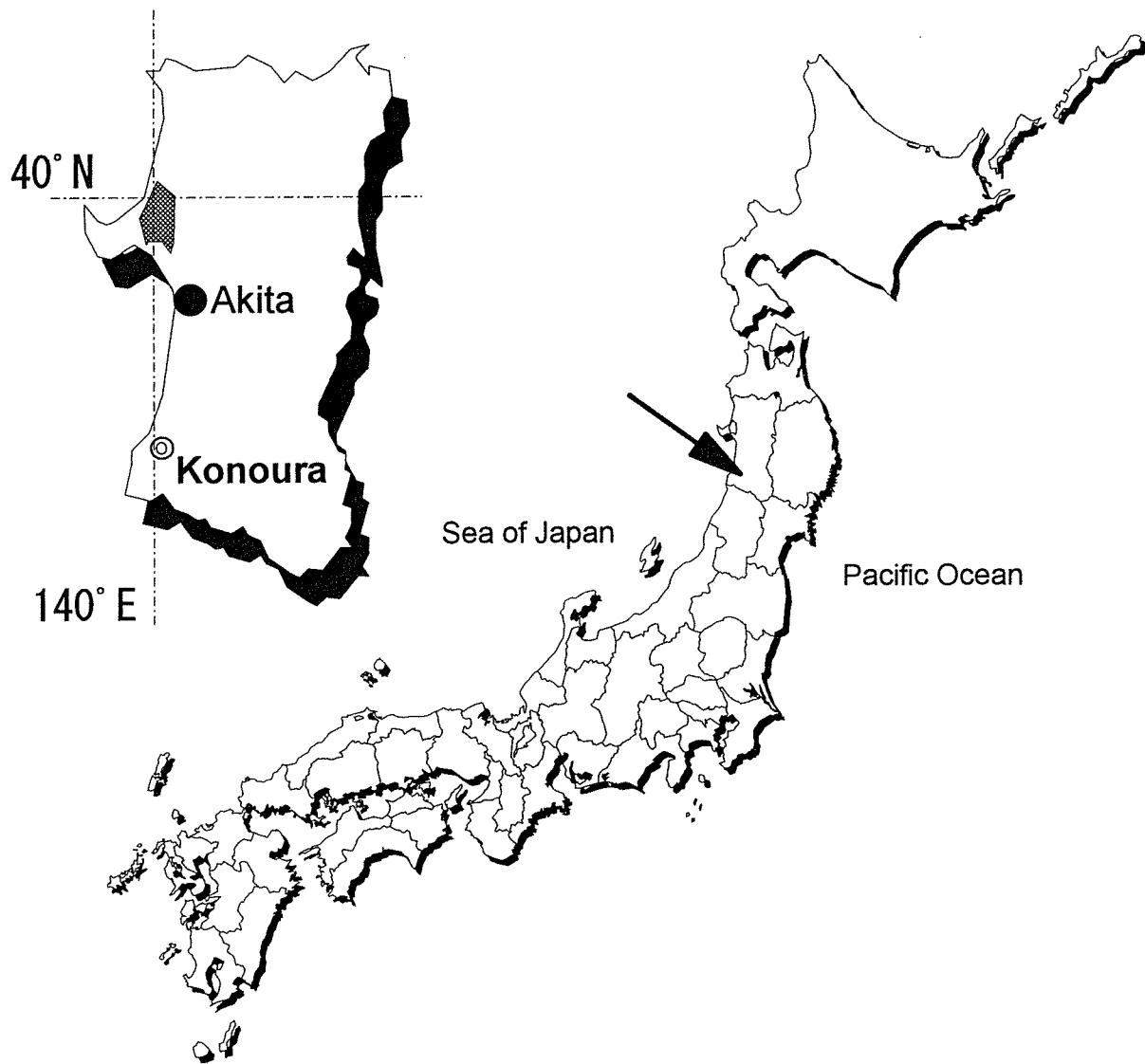


Fig. 6.1: 解析対象とする海域

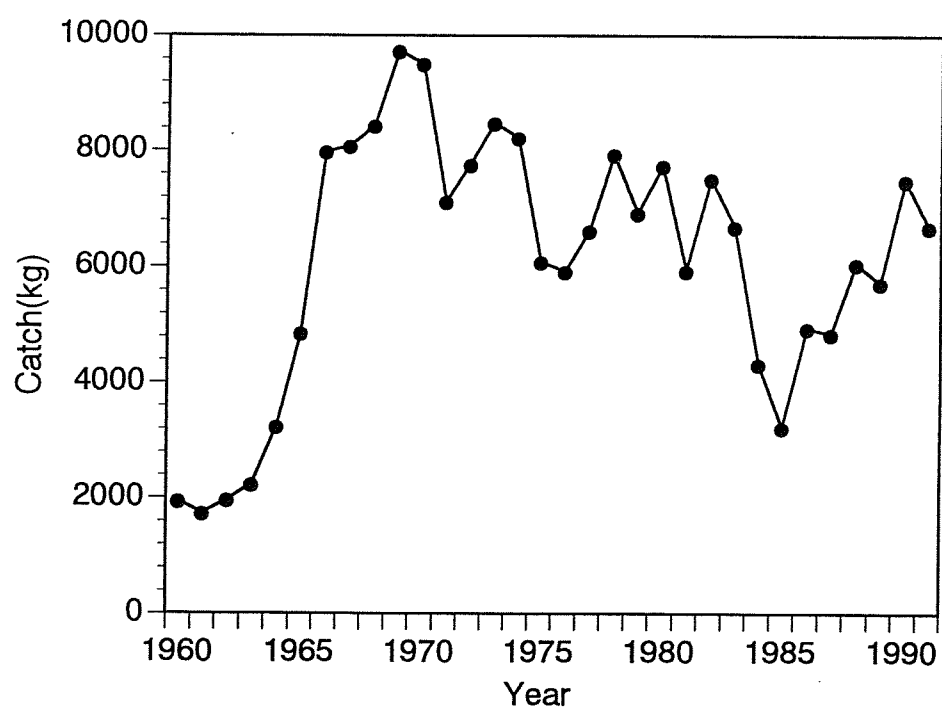


Fig. 6.2: 象潟・金浦地区のアワビの漁獲量

避けられない。松石ら⁴⁴⁾では、輪紋の読み飛ばしが一定の確率で起きると仮定したモデルを作成し、これにより、正確な成長曲線の推定を可能にしている。

松石ら⁴⁵⁾は、上記の方法によって得られた秋田県南部のアワビの年齢-殻長データを von Bertalanffy, Logistic, Gompertz, Richards の成長曲線にあてはめた。その結果、僅差ながら以下の Logistic 成長曲線が最もよくあてはまった。

$$\ell_t = \frac{105.9}{1 + e^{-0.4498(t-3.200)}} + \varepsilon \quad (6.1)$$

従来、アワビの資源量推定には DeLury 法⁴⁶⁾やこれを改良した方法による資源解析が多く行われている。たとえば、Matsumiya and Matsuishi⁴⁷⁾は、2種類の価格が異なるアワビが同時に漁獲される場合について、努力量の配分を決定する関数を導入して2種を同時に DeLury 法で資源量推定する方法を開発し、五島列島のアワビ漁業に適用した。また、平山ら⁴⁸⁾は天候などによって生じる日ごとの漁獲効率の変化を考慮に入れた解析をおこない、秋田県南部のアワビ漁業に適用した。

DeLury 法の場合は努力量が正確に把握されなければならない。秋田県南部のアワビ漁業の場合は1日の漁獲個数に制限があるために、日を努力量とすると資源量にかかわらずほぼ一定の CPUE が算出されてしまうため、資源量推定に用いることができない。Zhao *et al.*⁴⁹⁾は、年齢別漁獲個体数を体長組成から Tanaka and Tanaka³¹⁾によって推定し、これを用いたコホート解析によって秋田県南部地区のアワビの資源量推定を行っている。

ここでは体長組成をそのまま用いて、LPA によって資源量を推定する。またその結果を、Zhao *et al.* の推定値と比較する。

6.2 解析に用いたデータ

用いた漁獲物の殻長組成データは、1986年から1993年に秋田県南部漁業協同組合象潟支場に集荷されたアワビの殻長を、月に2~4回、秋田県水産振興センターが測定したものである。この地区のアワビは、天然産のものと放流種苗から育ったものに分けられるが、ここでは一括して解析した。

成長に関するデータをとる標本は、1990年7月から11月、および、1993年7月と9月に、金浦地区禁漁区において採集された356個体である。これを秋田県水産振興センターの斎藤和敬と松石が透過法により年輪に相当する輪紋の長径を測定し、各個体の各年齢での殻長を求めた。

得られた年齢-殻長関係と、その標準偏差を Fig. 6.3, Table 6.1 に示した。

6.3 解析方法

漁獲物の殻長測定の結果100mmを下回る殻長の漁獲物も少数あったが、個体数が少なく、また漁獲選択性が不安定であることが予想されることからデータから除外した。また、

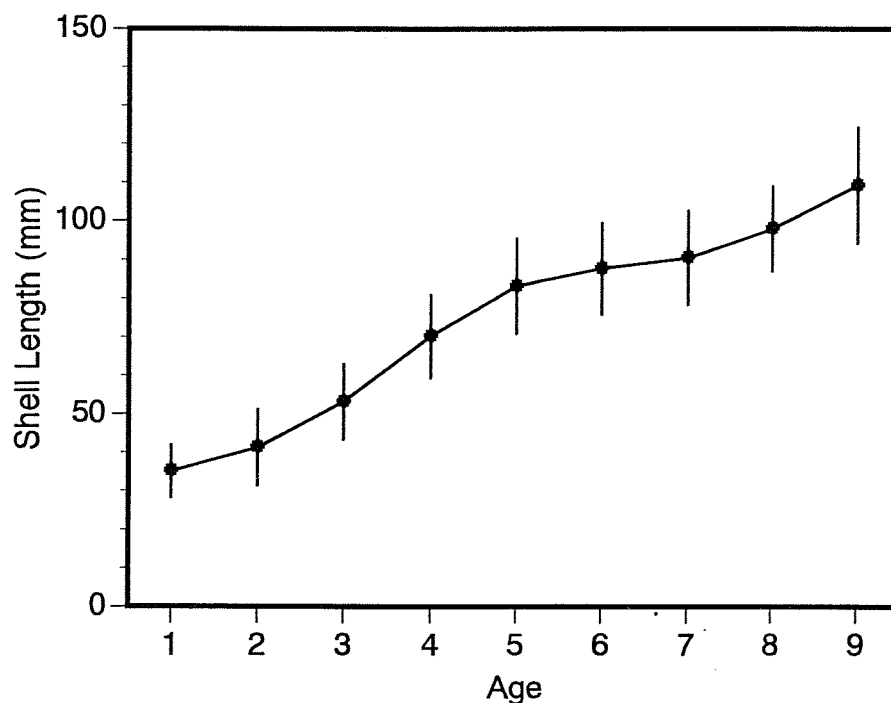


Fig. 6.3: 1990年7月から11月、および、1993年7月と9月に、金浦地区禁漁区において採集されたアワビの輪紋から得られた年齢-殻長の平均値(点)と標準偏差(縦棒)

Table 6.1: 1990年7月から11月、および、1993年7月と9月に、金浦地区禁漁区において採集されたアワビの輪紋から得られた年齢-殻長関係と、その標準偏差。

Age	mean (mm)	s.d. (mm)	N
1	35.0	6.8	110
2	41.2	9.9	195
3	53.1	9.8	238
4	70.1	10.8	232
5	83.1	12.3	180
6	87.6	11.8	84
7	90.4	12.1	32
8	97.9	10.9	12
9	109.1	15.0	6

Table 6.2: 象潟地区における体長別漁獲個体数。†のついた数値は補間により求めた。

殻長階級	平均重量 (g)	年							
		1986	1987	1988	1989	1990	1991	1992	1993
100-105	130.32	5573	4288	6543	8963	2981	4963	4278	5109
105-110	150.76	5495	6979	10211	8327	11014	11820	8146	10066
110-115	173.26	6430	4903	7276	4909	8958	8680	6098	6622
115-120	197.91	3079	3581	3974	4074	6515	4871	3504	3027
120-125	224.81	3702	2522	2629	2305	4030	3116	2048	1362
125-130	254.08	1052	1414	2018	1828	2415	1824	1138	833
130-135	285.81	1013	716	1101	576	1560	1108	683	341
135-140	320.10	312	456	245	735	773	669	410	151
140+	376.56	468	†821	573	278	1008	231	546	227
Total		27123	25680	34570	31996	39254	37282	26848	27738

ほとんどの個体は殻長 140mm 以下であった。そこで、体長階級は 100mm から 5mm ずつ 140mm までの 9 階級とした。140mm を越える個体は最後の体長階級 (140mm+) に含めた。

体長別漁獲量は、以下の手順により推定された。

まず、データとして得られている市場調査による漁獲物の殻長を、上記の体長階級に分けた。次に、得られている、体長と体重の関係式⁵⁰⁾

$$w_x = 9.209 \times 10^{-5} \ell_x^{3.059} \quad (6.2)$$

を用い、各階級の個体数に階級値中央における計算体重を乗じた。

各年でこれを合計し、漁獲量との比をとり、この比にもとづいて、個体数を引き延ばした。

殻長データが無いために、個体数が 0 となる体長階級については、以下の式にもとづいて補間した。

$$\tilde{C}_{i,j} = \frac{\sum_i C_{i,j} \sum_j C_{i,j}}{\sum_i \sum_j C_{i,j}} \quad (6.3)$$

以上の手順により求められた、体長別漁獲個体数を Table 6.2 に示した。

行列 P は、方法 1 と方法 2 の 2 通りの方法で作成した。

得られている各個体の各齢での殻長 (Table 6.1, Fig 6.3) をもとに、方法 1 によって求められた行列 P を Table 6.3 に示した。

この方法は、成長に関するさまざまな仮定を排除し、各個体の各齢での殻長を直接用いるため、標本数が多いと良い推定結果が得られることが期待される。しかし、今回用いた標本には、漁獲対象となるような大型の個体が少なかったので、行列 P を作るのに十分なデータ数が得られなかった。そこで、方法 2 による推定も行った。

Table 6.3: 象潟地区のアワビの遷移行列(方法1によるもの)

$t+1$ 年の 体長階級	t 年の体長階級								
	100	105	110	115	120	125	130	135	140
	↓	↓	↓	↓	↓	↓	↓	↓	+
	105	110	115	120	125	130	135	140	
100 - 105	0	0	0	0	0	0	0	0	0
105 - 110	0.111	0	0	0	0	0	0	0	0
110 - 115	0	0.111	0	0	0	0	0	0	0
115 - 120	0.213	0.213	0	0	0	0	0	0	0
120 - 125	0	0.25	0.25	0	0	0	0	0	0
125 - 130	0	0	1	0	0	0	0	0	0
130 - 135	0	0	0	0.5	0.5	0	0	0	0
135 - 140	0	0	0	0	0	1	0	0	0
140 +	0	0	0	0	0	0	0.5	0.5	0

方法2による行列 P を作成する際には、 $\bar{\ell}_{a+1}$ と $\bar{\ell}_a$ の直線回帰、および、 $\bar{\ell}_a$ と $\bar{s}_a$ の直線回帰を行なう必要がある。全ての個体について ℓ_{a+1} と ℓ_a の両方を得ることができる全ての齢 a についてプロットし直線回帰を行った結果、

$$\hat{\ell}_t = -7.598 + 0.903\ell_{t+1} \quad (N = 823, r = 0.92) \quad (6.4)$$

となり、有意 (t -test, $P < 0.01$) に相関したので、この関係を用いた。Fig. 6.4 に、 $\bar{\ell}_{a+1}$ と $\bar{\ell}_a$ の関係を示した。

また、 $\bar{\ell}_a$ と $\bar{s}_a$ (Table 6.2) の直線回帰を行った結果、

$$\hat{s}_a = 5.46 + 0.0753\bar{\ell}_a \quad (N = 9, r = 0.87) \quad (6.5)$$

となり、有意 (t -test, $P < 0.01$) に相関したので、この関係を用いた。Fig. 6.5 に、この関係を示した。

以上の関係をもとに、方法2によって求められた行列 P を Table 6.4 に示した。

自然死亡係数 M は、秋田県水産振興センターの推定⁵¹⁾により、0.102 とした。

推定された結果は、方法1と方法2や Zhao *et al.* の推定結果と比較して検討した。また、方法2については Retrospective analysis⁵²⁾を行い、近年の推定結果の偏りを検討した。Retrospective analysis とは、用いるデータを1年ずつ減らしてコホート解析を行い、その個体数推定結果の変動と推移を検討するものである。ここでは、1986-1993年の全てのデータを用いて推定した結果にたいして、1986-1992, 1986-1991, 1986-1990, 1986-1989年のデータを用いて推定した結果を比較した。

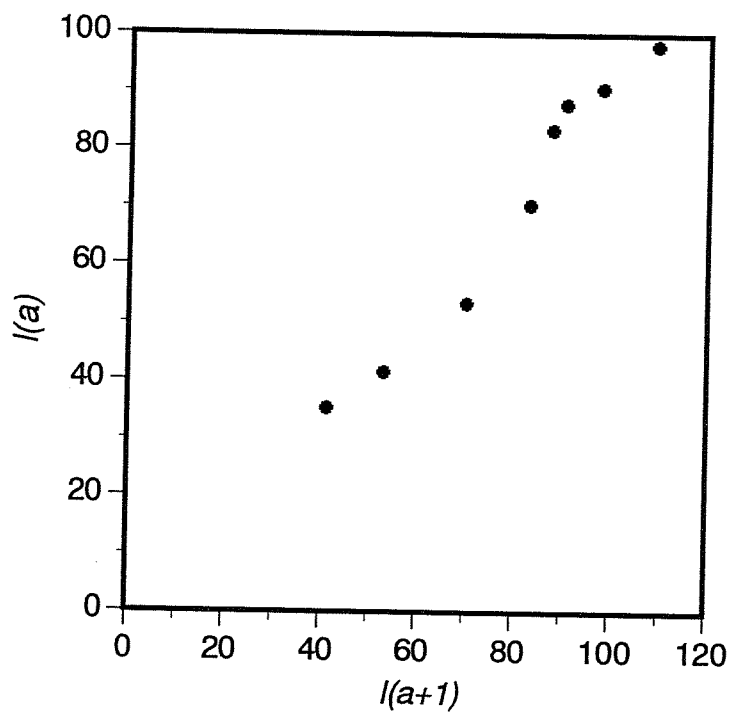


Fig. 6.4: $\bar{l}_{a+1}$ と $\bar{l}_a$ の関係。

Table 6.4: 象潟地区のアワビの遷移行列 (方法2によるもの)

$t+1$ 年の 体長階級	t 年の体長階級								
	100	105	110	115	120	125	130	135	140
	∟	∟	∟	∟	∟	∟	∟	∟	+
	105	110	115	120	125	130	135	140	
100 - 105	0.057	0.028	0.012	0.004	0.001	0	0	0	0
105 - 110	0.093	0.056	0.028	0.012	0.004	0.001	0	0	0
110 - 115	0.126	0.089	0.054	0.028	0.012	0.005	0.002	0	0
115 - 120	0.147	0.121	0.086	0.053	0.028	0.013	0.005	0.002	0.001
120 - 125	0.150	0.142	0.117	0.083	0.052	0.028	0.013	0.005	0.003
125 - 130	0.136	0.146	0.137	0.112	0.080	0.050	0.028	0.013	0.008
130 - 135	0.113	0.135	0.143	0.132	0.108	0.078	0.049	0.027	0.022
135 - 140	0.086	0.114	0.134	0.139	0.128	0.104	0.075	0.048	0.050
140 +	0.061	0.088	0.114	0.132	0.136	0.124	0.101	0.073	0.098

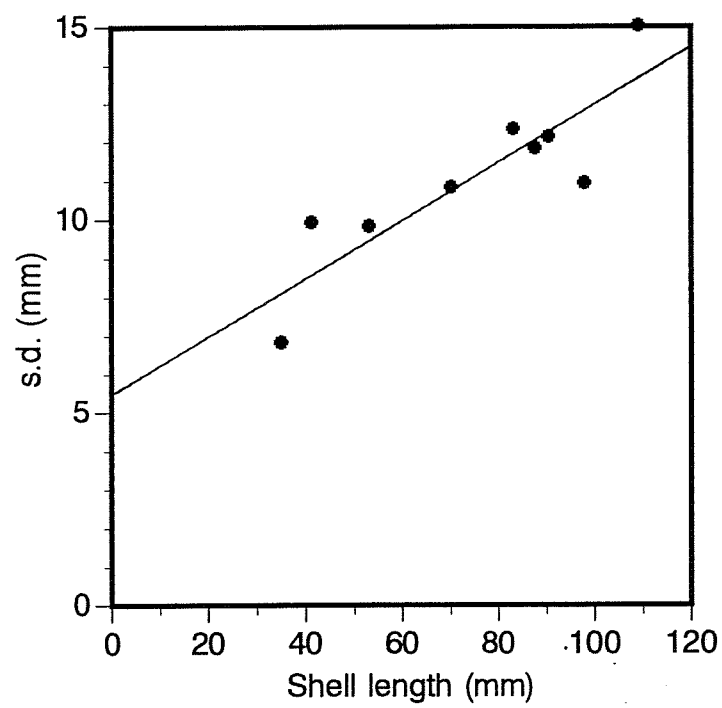


Fig. 6.5: s_a と $\bar{l}_a$ の関係。直線は $5.46 + 0.0753\bar{l}_a$

Table 6.5: 推定された秋田県南部のアワビの体長別資源個体数 $\hat{N}_{i,j}$ 。(方法1によるもの)

Size Class (mm)	Year							
	1986	1987	1988	1989	1990	1991	1992	1993
100 - 105	5140	6783	7964	8744	8923	7539	8071	7848
105 - 110	10534	9749	12327	13920	14859	12473	11063	13620
110 - 115	6612	12632	5433	12290	9691	10100	9778	8987
115 - 120	2729	2163	7635	1284	9382	3271	1463	3894
120 - 125	1971	1751	5087	1022	6291	2109	1046	1941
125 - 130	1497	1020	3292	588	2557	1020	1098	1578
130 - 135	3239	998	1384	1879	757	2382	685	509
135 - 140	1558	620	555	1125	308	887	322	444
140 +	785	1835	849	588	1506	329	900	374
Total	34069	37555	44531	41444	54277	40112	34430	39200

6.4 資源量推定結果

方法1によって推定された体長別個体数 $\hat{N}_{i,j}$ を Table 6.5 に示した。また、方法2によって推定された体長別個体数 $\hat{N}_{i,j}$ を Table 6.6 に示した。それぞれの方法で推定された $\hat{N}_{i,j}$, $\hat{f}_j$, $\hat{s}_i$ を Fig. 6.6 ~ 6.8 に示した。

推定された個体数は、方法1よりも方法2の方が多くなっている。方法1では34千個から54千個、方法2では41千個から61千個となり、どちらも1990年に最大値をとるなど、同様の傾向を示した。

完全加入漁獲係数 $\hat{f}_j$ は、方法1では1.0から2.2、方法2では0.9から1.5をとった。方法2の方が小さい値を推定した。また、方法1による推定値は大きく変動するのに対して、方法2による推定値は比較的安定していた。

選択率 $\hat{s}_i$ は、第4体長階級(115~120 mm)を1と標準化している。ここで示される選択率は、体長階級ごとの漁獲強度の指数であり、網目から逃避する率を示す「漁具の選択係数」⁵³⁾とは、定義が異なる。方法1では推定値が体長階級によって大きく異なっている。しかし、素潜り漁業の場合、漁獲選択性はほとんどないと考えられ、この推定結果は実際の漁業形態を反映していないと考えられる。方法2では110~140mmに対して1.0に近い選択性を示した。

Zhao *et al.*⁴⁹⁾の推定結果と、LPAの推定結果を Fig. 6.9 に示した。1991年を除いて、ほぼ同様な資源の増加を示している。全ての場合で、Zhao *et al.*の結果の方がLPAよりも多く推定しているが、これはZhao *et al.*では4歳以上の個体全てが漁獲対象になっていると仮定しているためと考えられる。Table 6.2に示されるように、4歳のアワビは平均殻長が70.7mmであり、全てが漁獲対象になってはいないと考えられる。

次に方法2の行列 P を用いて行った Retrospective Analysis の結果を検討する。1986-1993, 1986-1992, 1986-1991, 1986-1990, 1986-1989年の5種類のデータセットを用いて推定した殻

Table 6.6: 推定された秋田県南部のアワビの体長別資源個体数 $\hat{N}_{i,j}$ 。(方法2によるもの)

Size Class (mm)	Year							
	1986	1987	1988	1989	1990	1991	1992	1993
100 - 105	9783	11042	12283	12456	12840	10935	10355	11706
105 - 110	11484	12241	15080	14365	16351	15019	12062	16036
110 - 115	8633	9017	11264	10689	12187	11063	8221	10223
115 - 120	5688	5807	7210	6946	7704	6857	4735	4710
120 - 125	3877	3777	4729	4623	5005	4528	2783	2052
125 - 130	2413	2236	2759	2806	2871	2665	1483	1255
130 - 135	1469	1279	1548	1660	1582	1557	781	510
135 - 140	819	663	777	898	773	829	370	225
140 +	791	1505	910	481	1533	320	843	374
Total	44962	47571	56562	54928	60850	53777	41638	47095

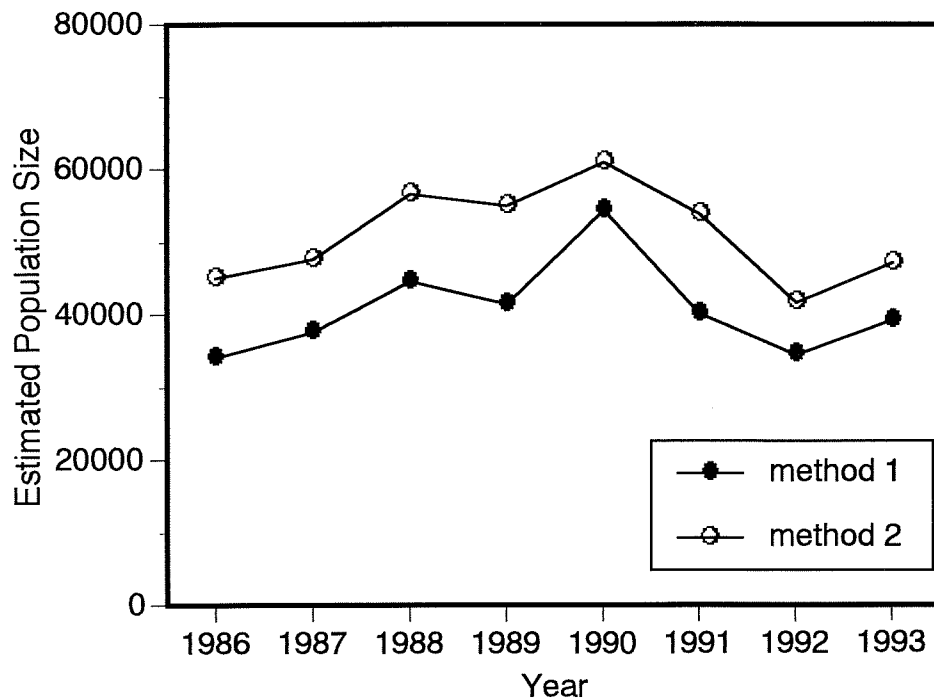


Fig. 6.6: 方法1と方法2によって推定された秋田県南部におけるアワビ資源個体数

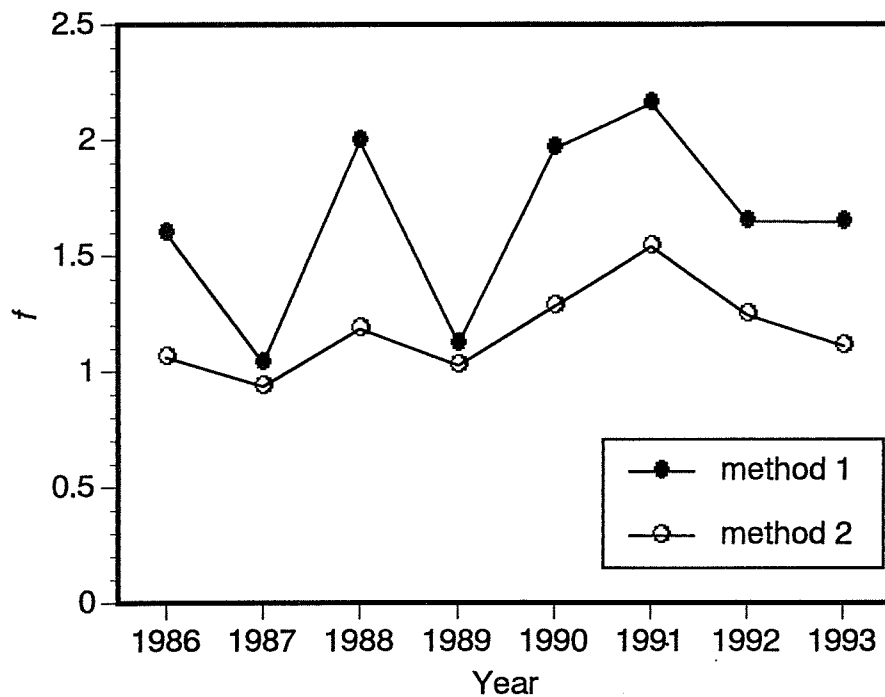


Fig. 6.7: 方法1と方法2によって推定された秋田県南部におけるアワビの完全加入漁獲係数

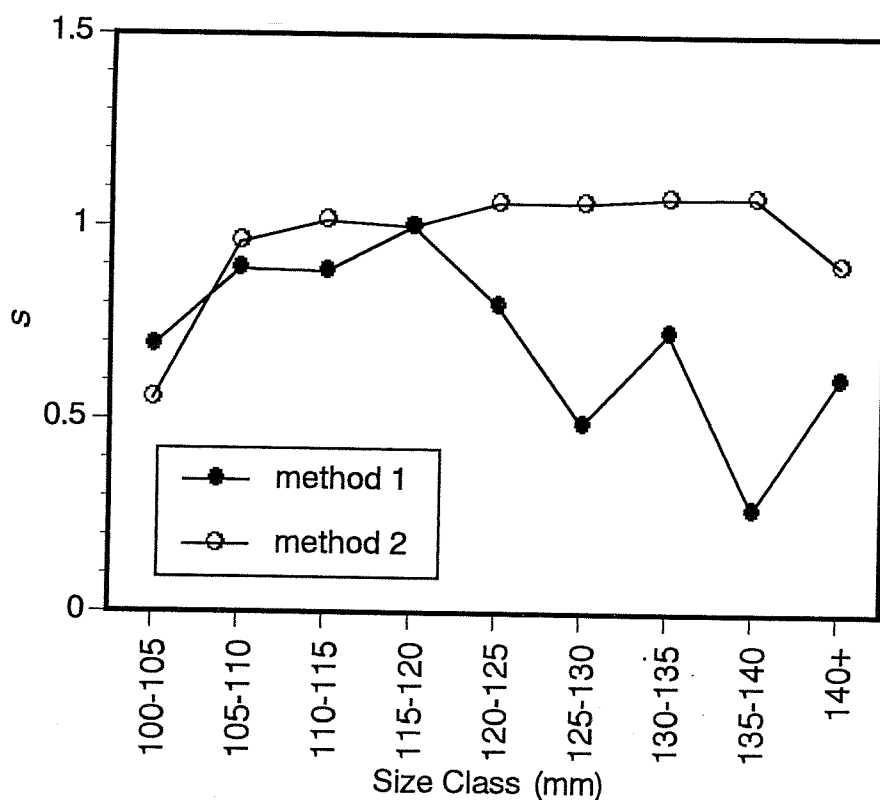


Fig. 6.8: 方法1と方法2によって推定された秋田県南部におけるアワビ漁業の選択率。115 - 120 mmの体長階級の選択率を1として標準化している。

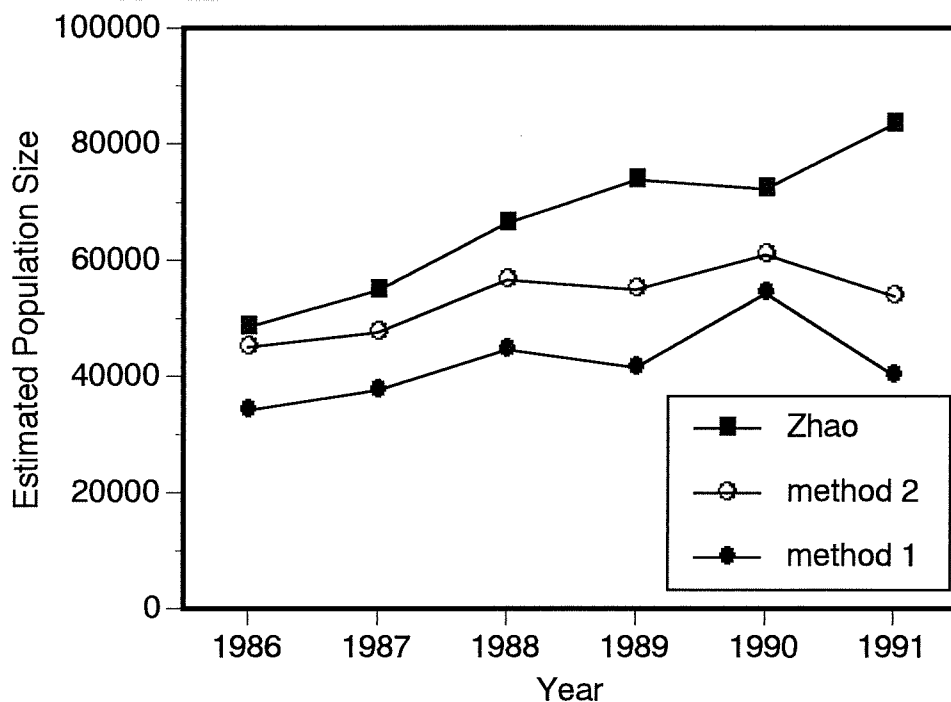


Fig. 6.9: 方法1と方法2とZhao *et al.*によって推定された秋田県南部におけるアワビ資源個体数

長100mm以上の資源個体数の合計をFig 6.10に示した。データセットによる推定値の変動は極めて少なく、近年の推定値に対しても精度が高いことが示唆される。

完全加入漁獲係数の推定値をFig 6.11に示した。1986-1992年のデータを用いた時の1992年の完全加入漁獲係数と、1986-1991年のデータを用いたときの1991年の推定値は完全加入漁獲係数は過大に推定されているようであるが、年をさかのぼるとすぐに、他のデータセットの推定値と同様な推定結果を示すようになり、年をさかのぼれば推定結果は信頼できることが示唆された。

選択率の推定値をFig 6.12に示した。体長の大きな体長階級では推定値がばらつき、特に1986-1991年のデータを用いた場合の135-140mmの推定値は大きくはずれている。しかし、130mm以下の体長階級についてはどのデータセットでも安定した推定結果を示している。

以上より、方法2を用いて行列 P を作成し、LPAによって秋田県南部におけるアワビ漁業のデータから資源個体数を推定した結果は、おおむね妥当であり、特に年をさかのぼった場合や体長の小さい体長階級に対しては、精度良く推定されていることが示唆された。

なお、今回解析した標本には大型の個体が少なく、方法1による遷移行列の信頼性が低いと考えられる。高年齢における年齢査定個体の個体数を増やすことによって精度の高い遷移行列を作成することによってより信頼性の高い推定値をえることができると考えられる。

この海域では、天然由来のアワビ(主にクロアワビ *Nordotis discus*)と放流種苗由来のアワビ(エゾアワビ *Nordotis discus hannai*)が混在しており、両者の成長に差があることが知られている⁴⁵⁾。これらを分けて解析することも、精度を上げる上で必要である。

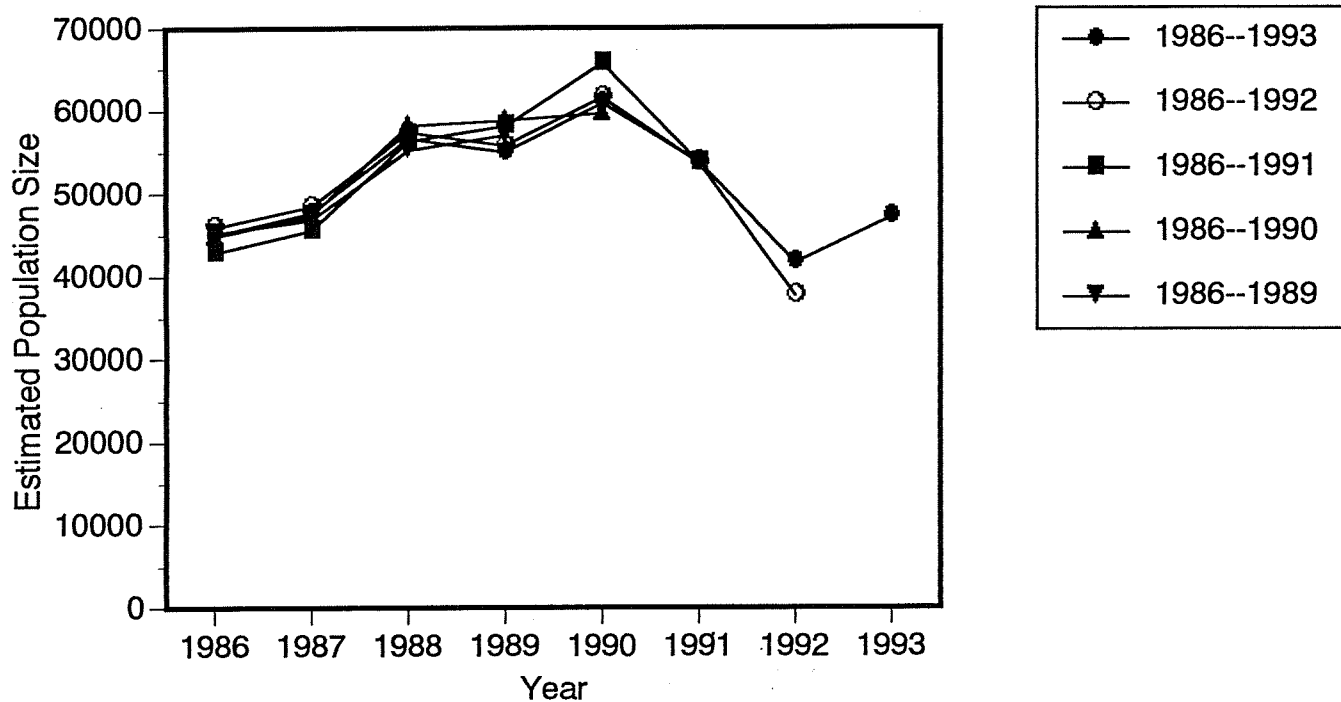


Fig. 6.10: 用いるデータの年数を変えて方法2によって推定した秋田県南部のアワビ資源個体数

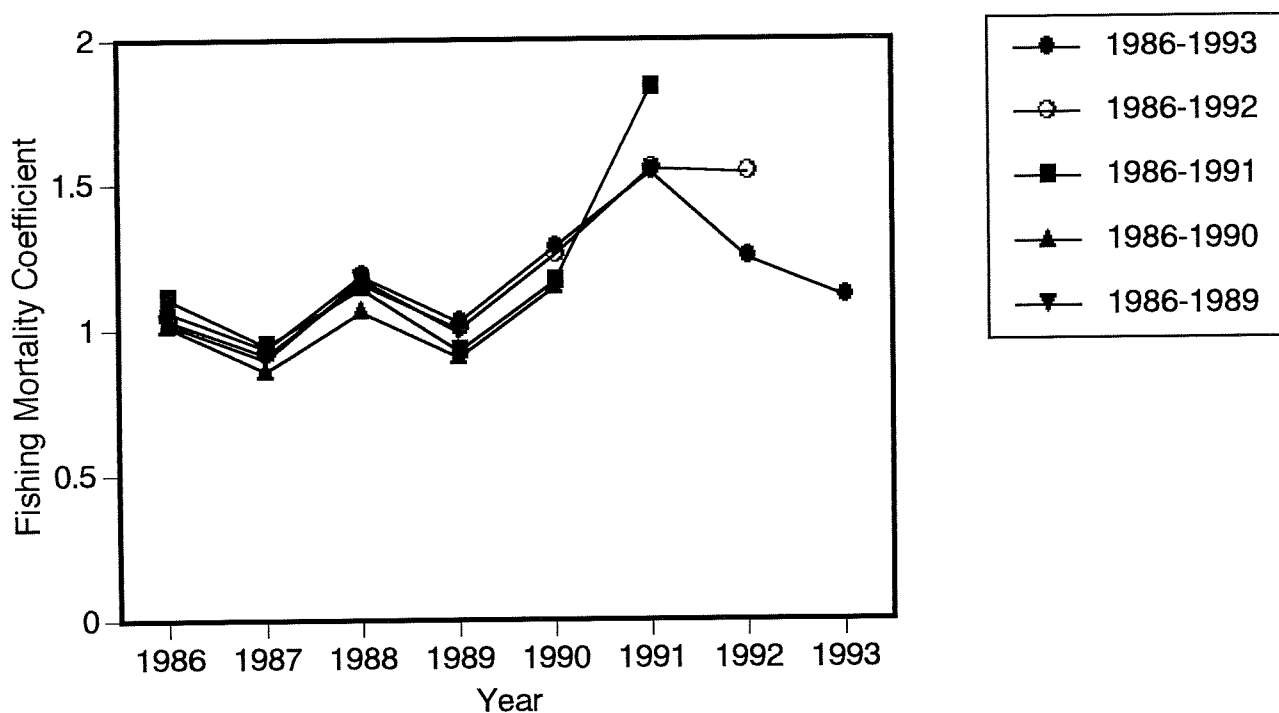


Fig. 6.11: 用いるデータの年数を変えて方法2によって推定した秋田県南部のアワビの完全加入漁獲係数

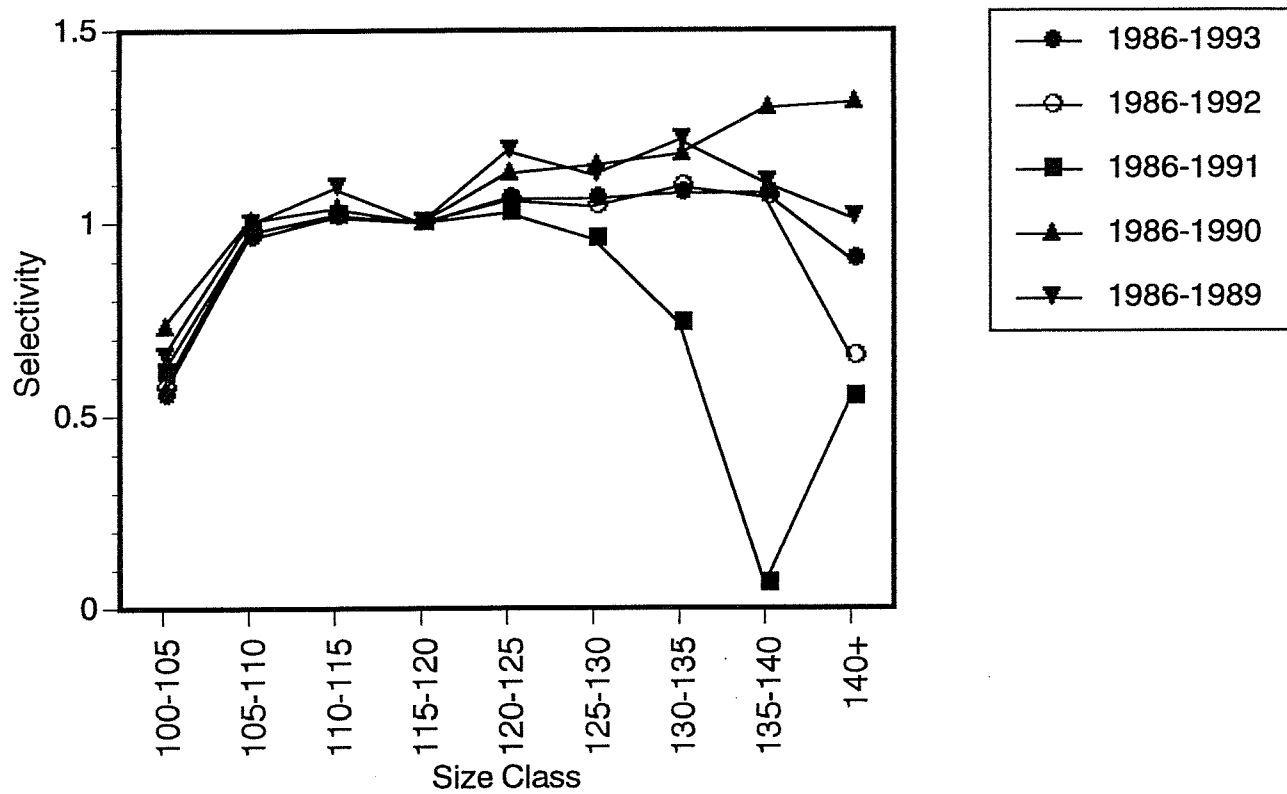


Fig. 6.12: 用いるデータの年数を変えて方法2によって推定した秋田県南部のアワビ漁業の選択率

第 7 章

事例解析 2: 北海道噴火湾沿岸におけるスケトウダラ漁業

7.1 漁業の概要

次に、北海道南西部太平洋側の噴火湾 (Fig.7.1) を主産卵場としているスケトウダラ *Theragra chalcogramma* に対する沿岸漁業 (襟裳以西、恵山以东) について解析する。

スケトウダラは漁業資源としては世界で最も大きなバイオマスを持ち、その食糧資源としての価値が認識されてきている⁵⁴⁾。日本沿岸では銚子以北の太平洋側、オホーツク海、本州以南の日本海に分布し、大きく太平洋系群・サハリンオホーツク海北見系群・北日本海系群・日本海南系群に分類される。ここで扱う太平洋系群の主産卵場は噴火湾から襟裳岬以西の沿岸である⁵⁵⁾。

この海域での産卵期は11月頃から3月である⁵⁶⁾。漁業は、この時期に産卵回遊してきたスケトウダラをねらい、刺網・底曳網・定置網で漁獲する。漁期は10月から始まり、3月に終了する。

漁具別漁獲量を Fig. 7.2 に示す。漁獲量は1966年度以降、4万トンから9万トンの間で変動をしているが、1980年度、1981年度に卓越発生が見られ、これが加入する1984年度以降、漁獲量は10万トンを越えている。その後、この卓越年級群の消滅に従って漁獲量は徐々に減り、1990年度には5万8千トンにまで減少した。漁獲対象となる魚は3歳以上、25cm以上のものである。ほとんどは刺し網によって漁獲されているが、近年は底曳網・定置網による漁獲も少しずつ増加している。しかし、その割合は1986年度では17%にすぎない。

噴火湾で産卵するスケトウダラの漁獲物の体長は、1950年代より北海道立函館水産試験場と同試験場室蘭支場によって測定されている。漁獲物の年齢査定は1980年代から行われている。

体長測定データは長期間にわたってあるが、途中で測定数が著しく少なかったり、全くデータが無い年がある。また、1987年度以降のデータは国際交渉にかかわる資料と指定されているため、詳細には公表されていない。

そこでここでは、体長組成を用いて1971年度から1986年度までの資源量をLPAにより

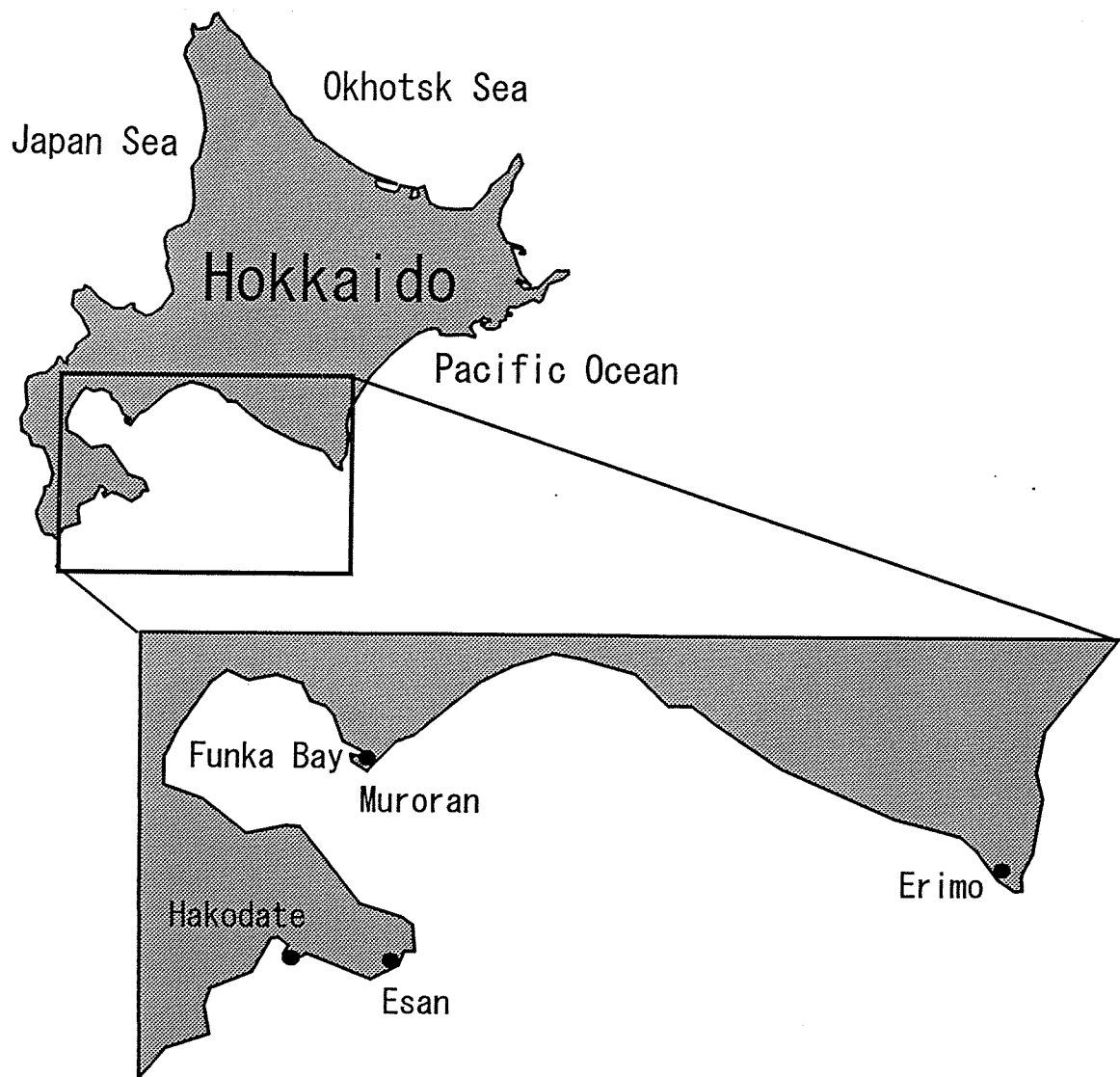


Fig. 7.1: 解析対象とする北海道噴火湾沿岸海域

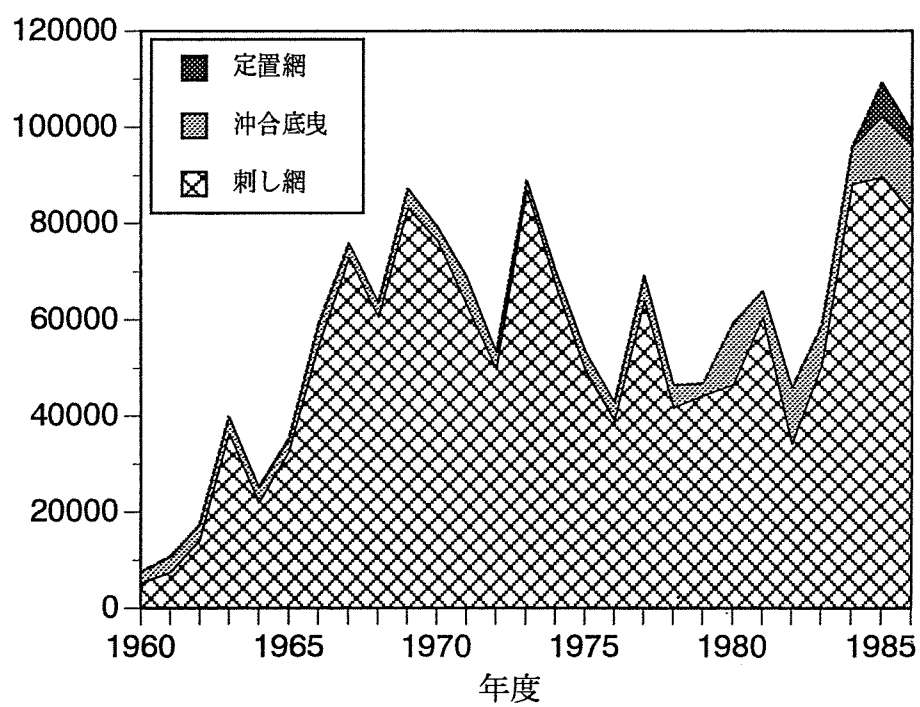


Fig. 7.2: 北海道噴火湾沿岸のスケトウダラ漁業の漁具別漁獲量の経年変化

Table 7.1: 北海道太平洋襟裳以西で漁獲されたスケトウダラの漁獲量(年度計、単位:t)

年 度	刺 網	底曳網	定置網	合 計
1971	62873	5819	0	68692
1972	49745	3518	0	53263
1973	87015	1950	0	88965
1974	67104	2931	0	70035
1975	49440	4022	0	53462
1976	39024	3808	0	42832
1977	64068	5180	0	69248
1978	41744	4683	0	46427
1979	44266	2565	0	46831
1980	46592	12796	0	59388
1981	60492	5507	0	65999
1982	34120	11710	0	45830
1983	50610	8156	0	58766
1984	88220	7634	0	95854
1985	89572	12529	7101	109202
1986	82133	13998	2970	99101

算出した。また、近年の推定結果については従来行われていたコホート解析の結果と比較検討した。

7.2 解析に用いたデータ

漁期が年をまたぐため、集計は年度毎に行った。

対象海域の漁獲量は、函館水産試験場室蘭支場によって集計されたものを用いた。用いた値を Table 7.1、Fig. 7.2に示した。

漁獲物の体長組成データは、函館水産試験場室蘭支場によって調査された漁獲物測定台帳によった。これを25から55cmまで5cm間隔の体長階級ごとに集計した。なお、55cmを越える体長をもった個体は、プラスグループ(55+)として扱った。

25cmに満たないスケトウダラも対象海域に回遊し一部漁獲されるが、完全加入しておらず、当該海域に回遊してくる割合も推定できないので、解析対象から外した。

行列 P を求める際に用いた年齢別体長は、1984年の測定台帳に記載されている4460個体から求めた。データを Table 7.2に示した。

Table 7.2: 北海道噴火湾沿岸で漁獲され、測定されたスケトウダラの年齢別体長(1984年度)

Age	mean	s.d.	N
0	69.8	27.5	273
1	170.7	19.1	586
2	238.5	22.8	201
3	332.0	26.1	417
4	374.2	16.6	1564
5	404.3	15.5	865
6	426.9	17.4	314
7	446.8	21.7	89
8	467.8	24.4	31
9	482.9	40.6	15
10	519.0	69.5	4
11	485.0		1

7.3 解析方法

体長と体重の関係は、1984年の測定結果より

$$w_x = 9.57 \times 10^{-6} \ell_x^{2.98} \quad (N = 4361, r = 0.997) \quad (7.1)$$

として求められた。

体長別漁獲個体数は、以下の手順により推定された。まず測定した体長を、5cm刻みの体長階級に分けた。なお、55cmを越える個体はプラスグループ(55+)として数え、25cmに満たない個体は計数しなかった。次に、体長-体重関係(7.1)を用い、各階級の個体数に階級値中央における計算体重を乗じた。プラスグループの階級中央値は60cmとした。年ごとに全体長階級の体重の合計を計算し、その年の漁獲量との比をとった。この比にもとづいて、個体数を引き延ばした。体長データが無いために、個体数が0となる体長階級については、以下の式にもとづいて補間した。

$$\tilde{C}_{i,j} = \frac{\sum_i C_{i,j} \sum_j C_{i,j}}{\sum_i \sum_j C_{i,j}} \quad (7.2)$$

このようにして得られた体長別漁獲個体数を Fig. 7.3 に示した。

行列 P は、1984年の測定台帳のデータをもとに、方法2によりもとめた。

$\bar{\ell}_{a+1}$ と $\bar{\ell}_a$ をプロットし直線回帰を行った (Fig. 7.4) 結果、

$$\hat{\ell}_t = -141.7 + 1.253 \ell_{t+1} \quad (N = 10, r = 0.994) \quad (7.3)$$

となり、有意 (t -test, $P < 0.01$) に相関したので、この関係を用いた。

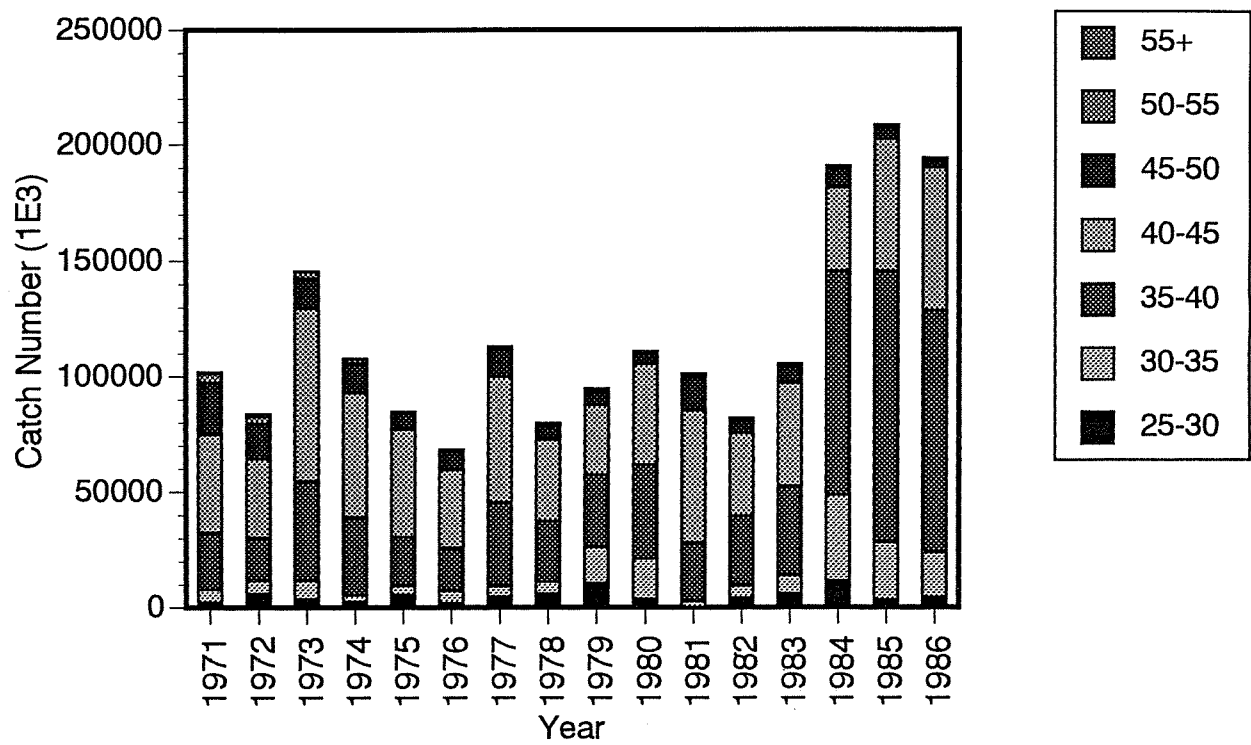


Fig. 7.3: 解析に用いた噴火湾産スケトウダラの体長階級別漁獲個体数

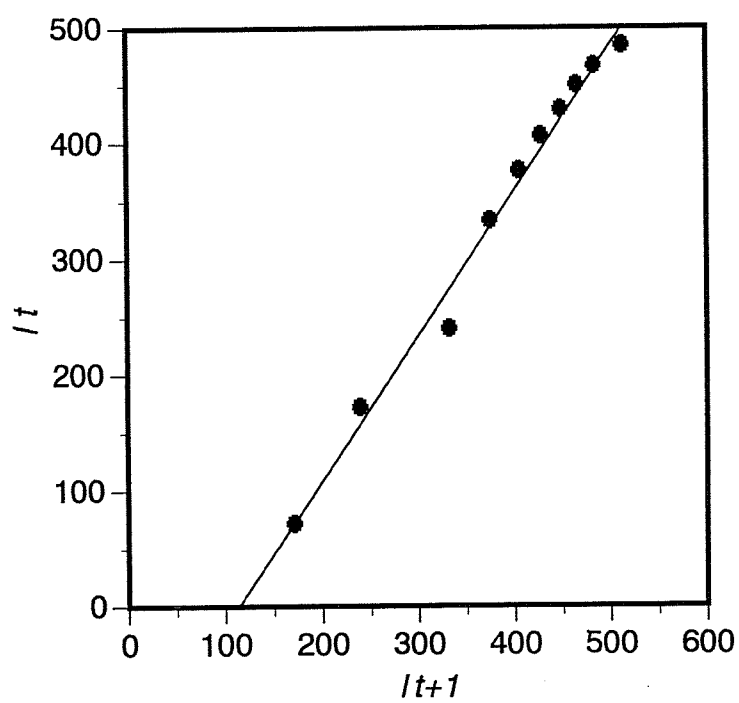


Fig. 7.4: $\bar{l}_{a+1}$ と $\bar{l}_a$ の関係。直線は $\hat{l}_t = -141.7 + 1.253l_{t+1}$

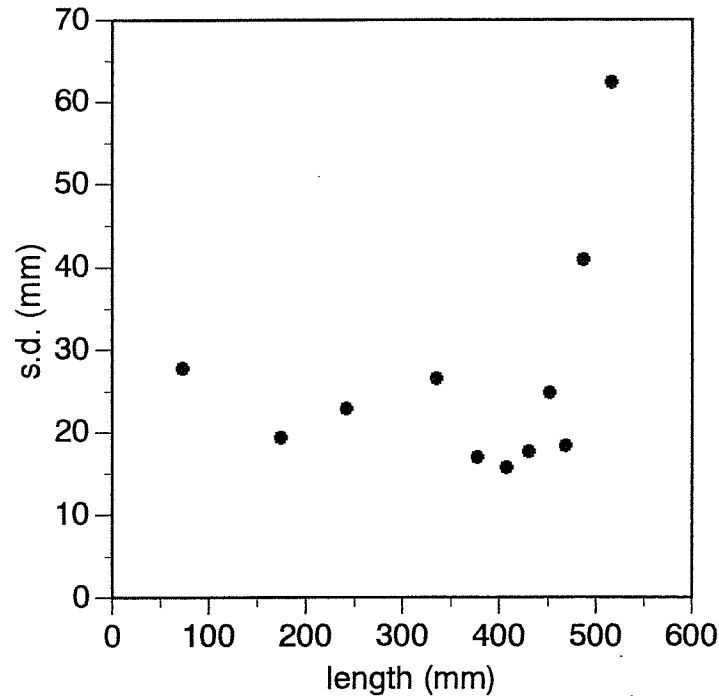


Fig. 7.5: s_a と $\bar{\ell}_a$ の関係。

また、分散の体長依存性を見るために $\bar{\ell}_a$ と $\bar{s}_a$ の直線回帰を行った結果 (Fig. 7.5)、

$$\hat{s}_a = 14.84 + 0.0324\bar{\ell}_a \quad (N = 11, r = 0.330) \quad (7.4)$$

となり、有意な相関は得られなかった (t -test, $P > 0.05$) ので、s.d. の平均値をとり、年齢や体長にかかわらず $s.d. = 20.84$ とした。

以上の結果を用いて求めた行列 P を Table 7.3 に示した。従来より行われているコホート解析⁵⁷⁾にあわせて、自然死亡係数 M は 0.3 とした。

また、1971-1986 年のデータに対して、1971-1985, 1971-1984, 1971-1983, 1971-1982, 1971-1981, 1971-1980, 1971-1979, 1971-1978 年のデータを用いて同様に推定する Retrospective analysis を行い、近年の推定値の偏りを検討した。

7.4 資源量推定結果

推定された体長別個体数 $\hat{N}_{i,j}$ を、Fig. 7.6、Table 7.4 に、完全加入漁獲係数 f_j を Fig. 7.7 に、選択率 s_i を Fig. 7.8 に示した。

Table 7.3: 1984年に測定された北海道噴火湾沿岸のスケトウダラから求められた行列 P

$t+1$ 年の 体長階級	t 年の体長階級						
	25-30	30-35	35-40	40-45	45-50	50-55	55+
25-30	0.012	0	0	0	0	0	0
30-35	0.722	0.049	0	0	0	0	0
35-40	0.089	0.764	0.147	0	0	0	0
40-45	0	0.025	0.646	0.327	0.002	0	0
45-50	0	0	0.005	0.430	0.552	0.013	0
50-55	0	0	0	0.001	0.220	0.728	0.052
55+	0	0	0	0	0	0.084	0.916

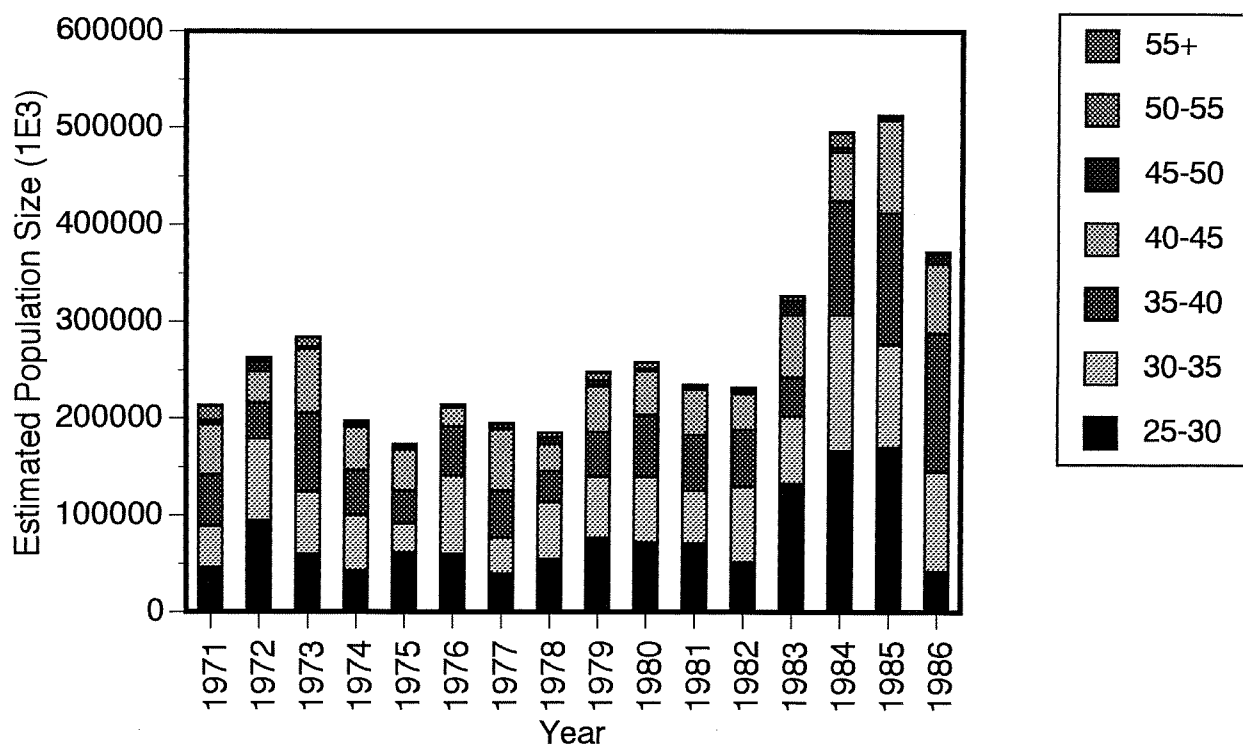


Fig. 7.6: LPA で推定された 25cm 以上の噴火湾産スケトウダラの資源個体数

Table 7.4: LPA で計算された噴火湾産スケトウダラの推定資源個体数（単位：千尾）。

Size class (cm)	Year							
	1971	1972	1973	1974	1975	1976	1977	1978
25 - 30	45810	94891	60119	42964	61189	60223	39556	54770
30 - 35	42945	83983	63558	56881	30423	80477	37142	58249
35 - 40	53261	37365	81734	47126	33546	50843	48359	32279
40 - 45	51568	31685	65829	43611	41877	19160	63015	27055
45 - 50	4172	9202	2228	4401	2407	1453	5160	7511
50 - 55	14270	3044	9595	1561	4005	1902	1305	4854
55 +	1271	2366	707	713	131	271	402	116
Total	213300	262539	283773	197260	173580	214331	194943	184836

Size class (cm)	Year							
	1979	1980	1981	1982	1983	1984	1985	1986
25 - 30	76543	71757	71293	51283	132820	166861	169779	41856
30 - 35	62633	67276	53916	78167	69470	139835	106320	103131
35 - 40	46200	64099	57783	58876	40307	117795	135725	142945
40 - 45	47208	45436	46677	37344	63842	50253	95376	71075
45 - 50	6331	3041	2186	4576	14645	4689	740	12321
50 - 55	7713	5594	2439	1851	4766	14857	3515	465
55 +	1569	1082	469	236	516	892	1198	341
Total	248200	258287	234767	232336	326368	495185	512656	372136

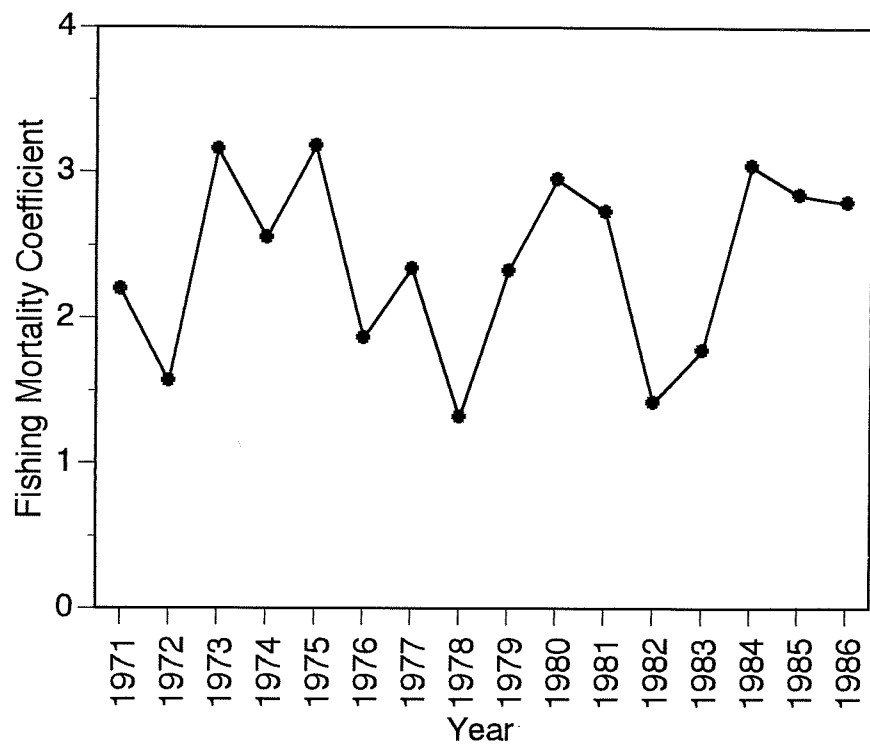


Fig. 7.7: LPAで推定された25cm以上の噴火湾産スケトウダラの完全加入漁獲係数

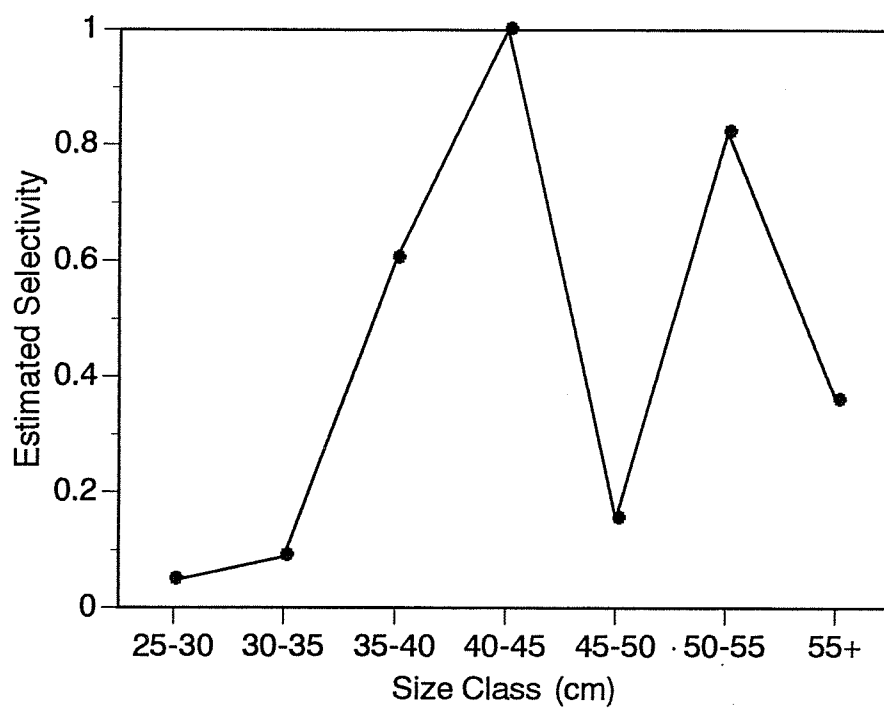


Fig. 7.8: LPA で推定された 25cm 以上の噴火湾産スケトウダラの選択率。35-40cm の体長階級を 1 として標準化している。

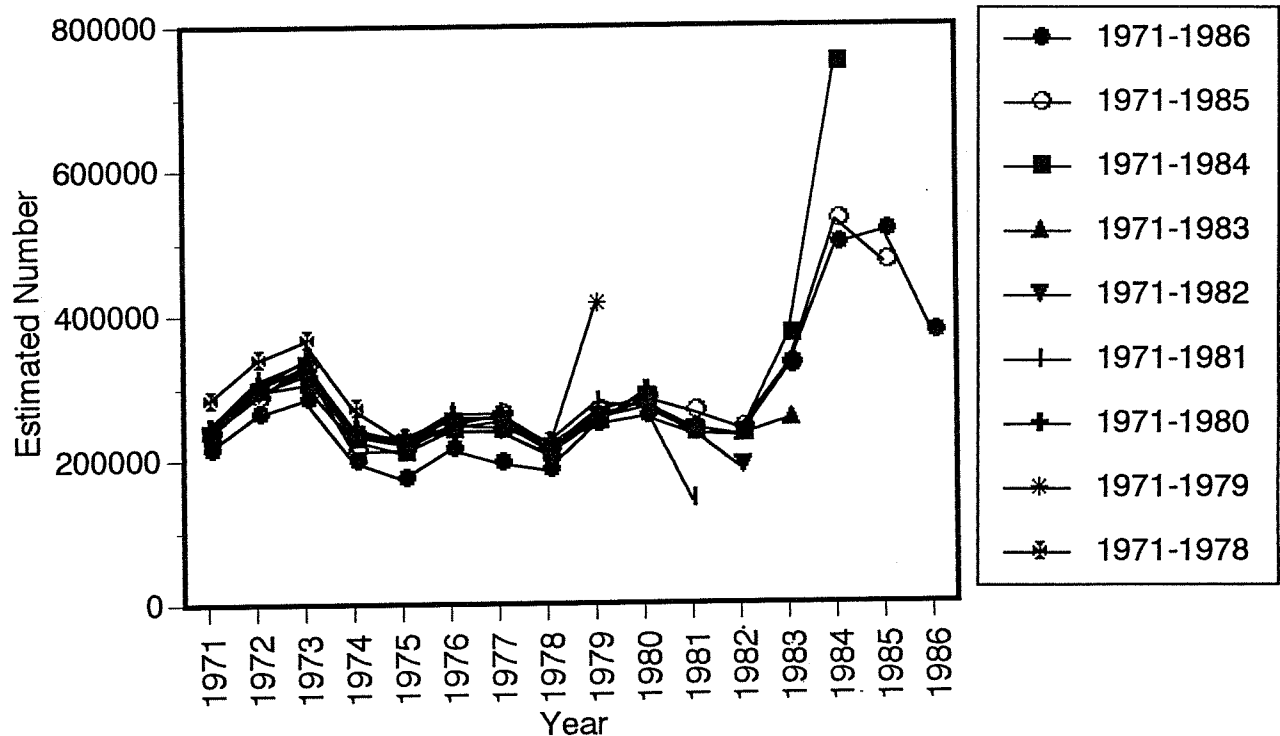


Fig. 7.9: 用いるデータの年数を変えて推定した 25cm 以上の噴火湾産スケトウダラの資源個体数

推定された来遊個体数は体長 25cm 以上の合計で、174 百万個体 (1977 年) から 513 百万個体 (1971 年) をとった。完全加入漁獲係数は 1.3 から 3.2 をとり、安定していない。また、選択率も 45-50cm の体長階級で著しく低く、残差平方和の極小点などに収束して誤った推定値が得られた恐れがある。

用いるデータを変えて個体数を推定した結果を Fig. 7.9 に示した。この結果、推定する最終年の推定値は大きくはずれることがある (たとえば 1971-1984 年のデータを用いた場合の 1984 年の推定値) が、年をさかのぼると、他のデータによって推定された推定値と近い値をとることが判る。また、推定値に疑問のあった全てのデータを使って推定した結果 (1971-1986) も、他のデータを用いたものと同様の推定結果を得ている。

用いるデータを変えて推定した完全加入漁獲係数を Fig. 7.10 に示した。多くのデータで、最後の年の推定がはずれている。しかし、年をさかのぼると、推定結果は収束することが判る。

また、用いるデータを変えて推定した選択率を Fig. 7.11 に示した。1971-1986 と 1971-1984 のデータを用いて推定した結果は、45-50cm の体長階級の選択率より 50-55cm の体長階級の選択率の方が大きくなっているが、他の全ての結果は逆になっている。刺し網の選択性から考えても、45-50cm の体長階級だけで選択率が低くなっていることは考えにくく、この 2 つのデータによる推定結果は選択率の推定を誤っていたと考えられる。しかし、資

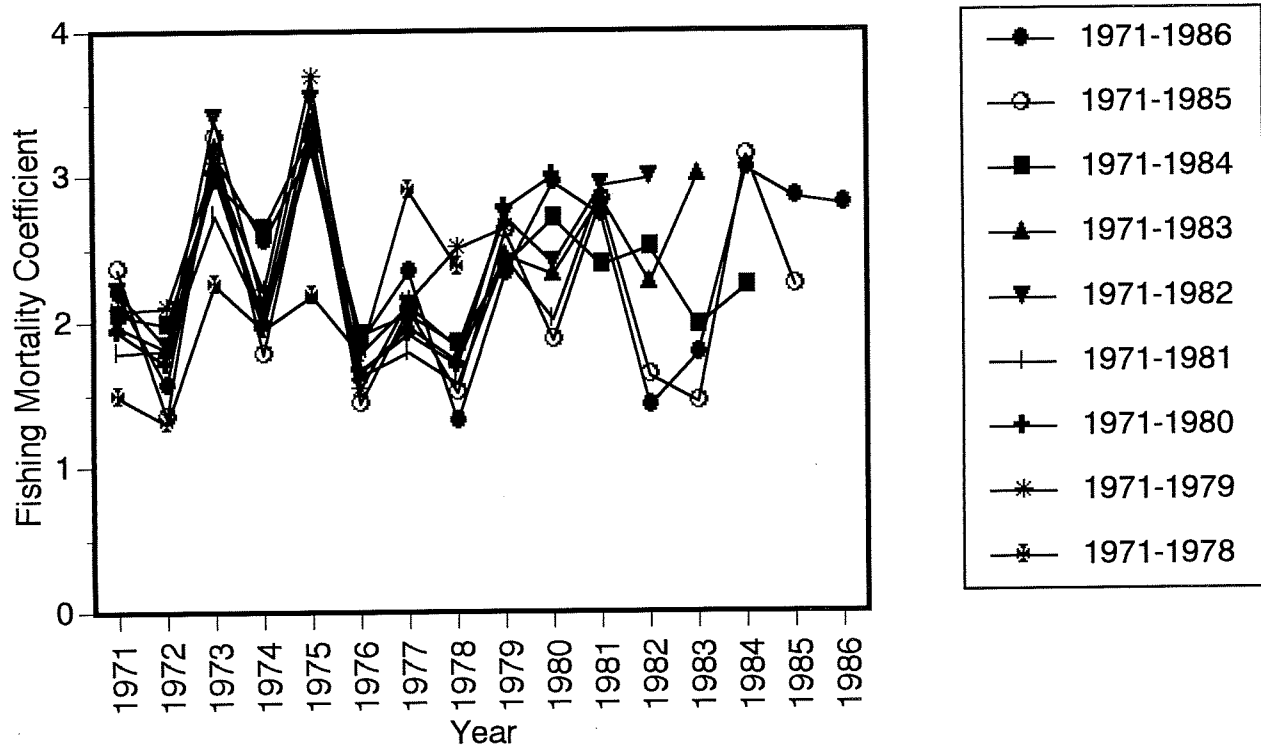


Fig. 7.10: 用いるデータの年数を変えて推定した 25cm 以上の噴火湾産スケトウダラの完全加入漁獲係数

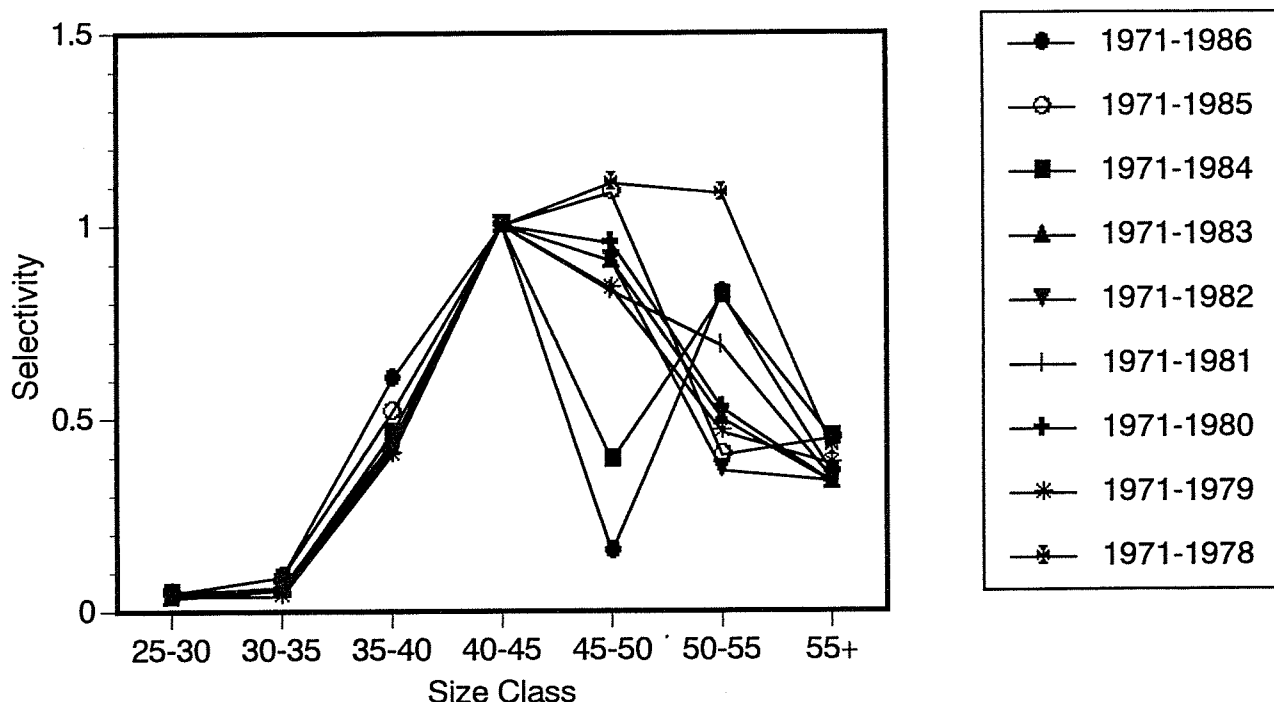


Fig. 7.11: 用いるデータの年数を変えて推定した25cm以上の噴火湾産スケトウダラの選択率

源個体数推定値にはこの誤りによる影響がほとんど及んでいない。

函館水産試験場室蘭支場が行った年齢によるコホート解析⁵⁷⁾による4歳以上の個体数推定結果とLPAによる推定結果をFig. 7.12に示した。1981年以前は両者とも極めて近い値をとっている。1982年以降は両者とも同様の傾向を示しているが、LPAの方が多く資源個体数を推定している。

なお、この水域では1980、1981年に卓越年級群が発生しているが、推定結果では1983、1984年に来遊尾数のピークを示しており、この現象をとらえている。

スケトウダラについては、アワビのように個体の成長の履歴を調べるのが困難であり、遷移行列は方法2を用いざるを得ない。また、シミュレーションで、成長に関する誤差が推定値に大きな影響を及ぼすことが示されているため、推定精度が悪くなることが懸念された。しかし、従来の方法と同様の結果が推定され、卓越年級群や漁具の選択性もほぼ正しく推定されており、このような資源に対しても適用可能であることが示された。

スケトウダラは、体長測定の際に雌雄を判別することが可能であり、また雌雄によって成長や体長-体重関係も異なっていると考えられる。また、長期間のデータを扱う場合は、その間の成長の変化が懸念される。以上の点を詳細に検討することによってより精度の高い推定値を得ることができると考えられる。

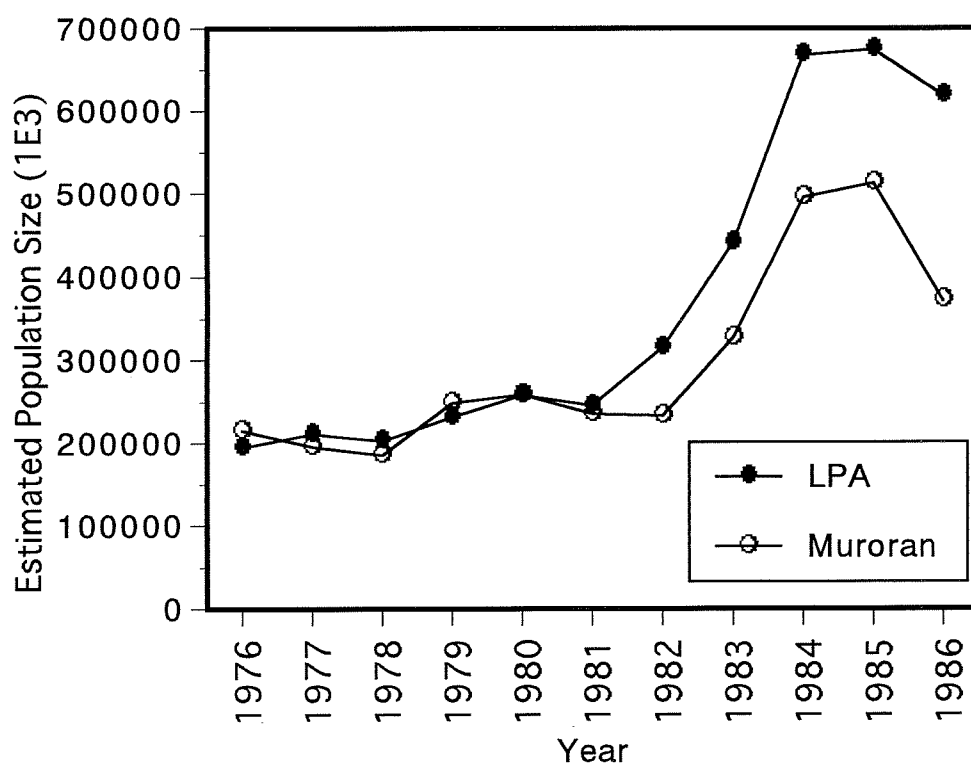


Fig. 7.12: LPAによって推定した25cm以上の噴火湾産スケトウダラの資源個体数(LPA)と函館水産試験場室蘭支場がVPAによって推定した4歳以上のスケトウダラ資源個体数(Muroran)

第 8 章

まとめ

LPA の推定精度

これまで議論してきたように、LPA によって従来の方法と遜色ない精度の資源量推定が可能であることが示された。従来の VPA と同様に最後の数年の推定結果は不安定なこともあるが、年をさかのぼると急速に推定結果が安定した。また、小さい体長階級の方がよい推定結果が得られた。

成長に関する変動が大きくなると、推定精度の低下や偏りが生じることがシミュレーションで示されたが、複合的な変動が入っていると思われる事例解析のデータに対しても妥当と考えられる結果が得られ、頑健性は高いと考えられる。

体長組成データを用いる利点

LPA は、漁獲物の体長組成が得られているが年齢査定は重点的に行われていない資源に対して、漁具の選択性を利用した漁業管理を行う際に、有効である。年齢査定は耳石や鱗の採取とその分析が必要である。これらの作業は、専門的な知識と技術を要するため試験研究機関が直接実施する必要がある。そのため、重点的に資源管理を行う必要がある資源に対しては実施できても、ほとんどの魚種に対しては、コホート解析のデータとして用いるに足るだけの標本数を年齢査定していないのが現状である。

また、年齢査定をするためには魚体を傷つけなければならないため、標本を買い取らなければならない。希少で需要の高い高価な魚種ほど資源管理が必要であるにもかかわらず、金銭的制約から高価な魚種ほど年齢査定が困難になる。

その点、体長の測定には、専門的な技術はほとんど不要である。また、迅速かつ非破壊的に大量に測定することが可能である。そのため、既に多くの魚種でデータが蓄積されており、また新たに収集することも容易である。

行列 P の作成について

体長組成の変遷を表す行列 P を作成する際には、成長に関するデータを用いる必要がある。各個体の成長の履歴を用いて行列を作成する方法(方法1)を行うには年齢査定だけでは不十分である。

たとえば、鱗を用いる場合、年齢査定を行なうためには年輪に相当する部分の本数を数えればよい。しかし、成長の履歴を知る場合には、それぞれの年輪半径を測定するとともに、鱗半径と体長関係を求めて、両者を用いて各年齢での体長を推定することになる。年輪数を計数する作業に対して年輪径を測定する作業は煩雑であり、多数の標本を処理することは困難である。また、本研究のアワビの事例解析のように、漁獲対象となっている体長階級の標本が十分に集まらなないと、行列はつくれない。

アワビやスケトウダラの事例解析では、行列 P を各年齢での平均体長とその分散から求める方法(方法2)によって作成し、これによって推定を行った結果、妥当と思われる結果が得られた。方法2では、各年齢での平均体長と分散が判ればよいので、年輪径などの測定は不要である。

全く年齢査定がされていない場合は、体長組成を正規分布に分解して平均体長と分散を推定することも可能かもしれない。しかし、この過程で誤差が導入される上、年齢組成も同時に推定されるので、LPA で解析するメリットは減少するだろう。

不安定な推定結果について

本研究で開発した LPA は、大きい体長階級と最終年の推定値が不安定であった。高齢の体長階級の推定が不安定になる原因の一つは、その階級に入る個体数が少ないことであると考えられる。したがって、ある程度不安定でも、全体の推定結果には大きな影響を与えない。

また、体長が大きくなると成長速度が遅くなるため、複数の年級群が一つの体長階級に入りその分離ができないのも原因の一つである。この場合、成長速度に応じて小さい体長階級は広く、大きい体長階級は狭くすると、各年級群の分離が良くなると考えられる。しかし、体長階級を狭くしすぎると、そこにはいる個体数は少なくなり、推定が不安定になる恐れがある。

最終年の推定結果が不安定であるのは、最終年近くに生まれた年級群にたいする漁獲量などの情報が少なく、漁獲係数の推定が困難であるためであると考えられる。本法では最終年の漁獲係数 F_T も推定したが、これを与えた方が推定結果は安定する。しかし、誤った

F_T を与えれば、推定結果は偏るので注意が必要である。

欠測年の扱いについて

本研究で用いたスケトウダラのデータには途中に欠測年があるために、1960年代の体長組成データがあるにもかかわらず推定に利用できなかった。体長組成の欠測年がある場合、漁獲量と選択率からその年の体長別漁獲量を推定して解析することも考えられる。今後検討したい。

重量組成によるコホート解析について

本研究では体長データを用いてコホート解析を行ったが、漁獲物の重量組成のデータがあり、対象魚種の重量にかんする成長がわかれば、同様にコホート解析が可能である。大型魚の場合は、市場で1個体ずつ重量が測定されているので、この伝票を用いれば特別な測定を行わずに重量階級別漁獲個体数が求められるだろう。また、体長－重量の換算が省けてより良い結果がでると考えられる。

摘要

従来、多くの種でVPAやコホート解析といった年齢別漁獲個体数から資源個体数を推定する手法が広く用いられている。一般に年齢査定には手数がかかるため、容易に測定することができる体長別漁獲個体数を用いて年齢別漁獲個体数を推定する方法がとられる。体長と年齢との関係は、魚種や漁獲年次により大きな変動を示すため、個体数を年齢別に推定する際に多くの誤差が生じる。

推定された年齢別個体数を資源管理に用いる際には、漁具の選択性などを利用することが多い。この際に必要な情報は、年齢別資源個体数ではなく、体長別資源個体数である。したがって、得られた年齢別資源個体数を再び体長別資源個体数に変換する必要がある。この過程でも誤差が生じる。直接体長別漁獲個体数から体長別資源個体数を推定する方法があれば、推定過程が単純になり、精度が向上することが期待される。

Jones(1984)は、体長階級毎の漁獲個体数からコホート解析を行なう方法を提示している。この方法は、ある体長階級を成長曲線が通過する期間ごとに漁獲量を集計して計算するものであるが、漁獲個体数を集計する期間が不定期になるため、利用できる状況は限られる。

Shirakihara(1983)はマッコウクジラの体長組成データを用いた資源動態解析を行っている。これは、再生産モデルによって加入量を推定し、これに体長組成の変遷を表す行列をかけることによって、順次、次年の体長組成を推定していくものである。この方法は加入量の推定値から計算を始めており、加入量が比較的安定している鯨類では有効であっても、加入量変動の大きい魚類等には適用が困難である。

そこで本研究では、上記の先行研究の欠点を克服して、加入量の変動の大きい種に対しても適用できるように、最近年および最大体長階級の個体数から順次年をさかのぼって体

長組成の変遷を推定することによって、体長別漁獲個体数から体長別資源個体数を推定する方法 (Length based Population Analysis / LPA) を開発した。本研究ではさらに、LPA の推定精度をシミュレーションにより検討し、従来の方法との精度の比較をおこなった。また、秋田県南部地区のアワビ漁業と北海道噴火湾岸におけるスケトウダラ漁業のデータへ、本法を適用した。

体長組成にもとづくマルチコホート解析法

本法は、従来の年齢別漁獲個体数にもとづくマルチコホート解析法を、体長別漁獲個体数データに用いることができるように改変したものである。年齢別データを用いる場合、ある年齢の資源は1年前には1歳若いという明確な関係を用いることができるが、体長別データを用いる際には、1年前に属していた体長階級は一意には定まらず、いくつかの体長階級に振り分けられる。

推定の際に必要なデータは、自然死亡係数 M 、および年 j における体長階級 i の漁獲個体数 $C_{i,j}$ 、および行列 P である。

行列 P とは、 $t+1$ 年に体長階級 k に属している個体が年 t に体長階級 j に属していた確率を (j, k) 成分とする行列である。もし、対象とする生物が漁獲時に過去の成長の履歴を追うことができる場合は、実際に上記に該当する場合の数を計数して行列 P を得ることができる。後に解析するアワビは、殻に刻まれた年輪から過去の成長の履歴が判るので、この方法を用いた。漁獲時の年齢と体長しか分からない場合には、年齢ごとの体長の分散と成長曲線から行列 P を推定する。後に解析するスケトウダラはこの方法を用いた。

推定方法は、以下の手順で行う。まず、年別漁獲係数 $\hat{f}(j)$ 、および体長別選択率 $\hat{s}(i)$ に初期値を与える。選択率は年によらず一定であると仮定し、漁獲係数 $\hat{F}_{i,j}$ を両者の積として与える。全死亡係数 $\hat{Z}_{i,j}$ は、漁獲係数 $\hat{F}_{i,j}$ と自然死亡係数 M の和として与える。最終年および、最大体長階級の個体数を、観測されている体長別漁獲量 $C_{i,j}$ と漁獲係数 $\hat{F}$ 、全死亡係数 $\hat{Z}$ によって求め、最終年より順に、前年の体長別個体数を、 $t+1$ 年と t 年の体長組成の関係を表す行列 P と全死亡係数 $\hat{Z}$ を用いて計算する。求められた各年の体長別資源個体数と $\hat{Z}$ 、 $\hat{F}$ から、モデルから導出される漁獲個体数 $\tilde{C}_{i,j}$ を計算し、これとデータとして得られている漁獲個体数の残差平方和 SS を求める。 SS が最小になるようにシンプレックス探索法によって最適化を行ない、収束したときの各パラメータ $\hat{N}_{i,j}$ 、 $\hat{F}_{i,j}$ 、 $\hat{f}(j)$ 、 $\hat{s}(i)$ が、最終的な推定値である。

シミュレーションによる妥当性の検討

実際に本法を適用する際には、たとえば自然死亡は餌環境や水温のランダムな変化に応じて年変動することが想定される。また、漁獲圧や漁具の選択性の変動も避けられないであろう。環境の変化に応じて成長率も変化することが予想される。そこで、自然死亡係数、漁獲係数、成長に関するパラメータ等に変動がある場合の体長別漁獲個体数の模擬データを出力するデータジェネレータを作成した。この模擬データをLPAにかけて体長別資源個体数を推定し、データジェネレータに与えた体長別資源個体数と比較することによって、LPAの推定量の精度と偏りを検討した。

まず、変動を全く加えない模擬データを用いて推定した結果、データジェネレータに与えた体長別資源個体数と極めて近い値がLPAによって得られ、正しく個体数を推定することが示された。年別漁獲係数や体長別選択率のランダムな変動に関しては、資源個体数は偏りなく推定された。また、自然死亡の変動に関しても大きな偏りは見られなかった。

以上の結果より、LPAは種々の変動を含む状況から発生する体長別漁獲個体数データに対して、頑健に資源個体数を推定することが示された。

従来の方法との比較

従来、体長別漁獲個体数から体長別資源個体数を推定するためには、Age Length Key (ALK) や体長組成分解 (LFA) によって体長別漁獲個体数から年齢別漁獲個体数へ変換し、これをVPAにかけて年齢別個体数を推定し、さらにこれを各齢の平均体長とその分散から体長別個体数に変換している。

従来の方法とLPAによって直接、体長別個体数を推定する場合の推定精度を比較検討するために、模擬データを用いて、ALKとVPA、LFAとVPA、およびLPAによって体長別資源個体数を推定した。

各パラメータにCV 5%の程度の変動を与えた模擬データを解析した結果、ALK + VPAによって推定された5年目第6体長階級個体数推定値の相対誤差50%点は0.142、LFA+CPAは-0.0987を示したのに対し、LPAは0.0095と絶対値が最も小さく、推定値の偏りが少ないことが示された。また、推定精度を表す相対誤差の標準偏差はLPAが0.0977と最も小さかった。

このように、LPAが従来の方法と同等以上の性能を持つことが示された。

秋田県南部のアワビ漁業への適用

実際のデータに対してこの解析が有効に働くかどうかを調べるために、秋田県南部地区のアワビの素潜り漁業データへLPAを適用した。

調査によって漁獲されたアワビ356個体の殻から年輪に相当する輪紋を判読して行列 P を作成した(方法1)。また、各年齢での平均殻長とその標準偏差からも行列 P を作成した(方法2) 秋田県水産振興センターが測定した1986年から1993年までの漁獲物の殻長組成と漁獲個体数から、殻長別漁獲個体数を求めた。自然死亡係数は秋田県水産振興センターの推定により0.102とした。

方法2によって個体数を推定した結果、10 cm以上の個体数は44千個(1987年)から68千個(1990年)と推定された。

従来のコホート解析によって得られている推定結果(Zhao *et al.*, 1993)の結果と比較すると、解析が重なっている1986~1991年について、同様な資源変動傾向が見られた。

また、用いるデータの年数を変えて推定を行ったところ、どの場合も同様な推定値が得られ、推定値の信頼性が高いことが示唆された。

噴火湾のスケトウダラ漁業への適用

北海道南西部太平洋側の噴火湾を主産卵場としているスケトウダラ *Theragra chalcogramma* の刺し網・底曳網漁業のデータについて解析した。漁獲物の体長組成は函館水産試験場室蘭支場によって測定された1971年から1986年のデータを用いた。行列 P は1984年に年齢査定された4460個体を用い、各年齢の平均体長と分散から求めた。自然死亡係数は、従来この海域のコホート解析に用いられていた0.3とした。

資源個体数は19.5億尾(1978年)から51.3億尾(1971年)と推定された。体長別選択性を示すパラメータ s は35cm以上の階級で急激に大きくなり、漁具の漁獲選択性を反映していることから、推定された結果の妥当性を裏付けている。

このように、成長の履歴が把握できない場合でも、解析が成功していることから、成長の推定が困難な種に対してもLPAが有効であることが示唆された。

まとめ

以上のように、LPAは体長別漁獲個体数から体長別資源個体数を推定する方法として有効であることが示された。年齢査定の困難な多くの魚種へ、この方法を適用して資源個体数推定を行うことが可能であると考えられる。

謝辞

本研究をとりまとめるにあたり、懇切なご指導と御校閲を賜った東京大学農学部教授 清水 誠博士に深く感謝申し上げます。研究の最初からとりまとめまで、常に懇切なご指導を賜った東京大学教養学部助教授 岸野 洋久博士に深く感謝申し上げます。校閲と有益なご教示を賜った東京大学農学部教授 上村 賢治博士、同助教授 谷内 透博士、東京大学海洋研究所助教授 青木 一郎博士に感謝します。東海大学海洋学部教授 沼知 健一博士、北海道大学水産学部教授 菅野 泰次博士、同助教授 西村 欣也博士には、温かい励ましと協力をいただきました。

秋田県南部のアワビ漁業の解析に関しては、秋田県水産振興センターの斎藤和敬氏に貴重なデータや標本の提供と懇切な助言をいただきました。スケトウダラの解析に関しては、北海道立函館水産試験場室蘭支場の上田吉幸氏と三原行雄氏に貴重なデータの提供と懇切な助言をいただきました。

また、東京大学海洋研究所教授 松宮 義晴博士、三重大学生物資源学部助教授 原田 泰志博士、水産庁遠洋水産研究所 平松 一彦博士には、研究に関して有益なコメントをいただきました。

本研究の一部は、平成6年度科学研究費補助金奨励研究(A)06760167、平成6年度・平成7年度北水協会補助事業、住友財団1994年度研究助成により行った。

参考文献

- 1) Anonymous. 平成5年度漁業の動向に関する年次報告(第129回国会(常会)提出). 1994.
- 2) J. G. Pope. An investigation of the accuracy of virtual population analysis using cohort analysis. *ICNAF Res. Bull.*, **9**: 65–74, 1972.
- 3) J. A. Gulland. *Estimation of Mortality Rates*, Vol. 3 of *Annex to Rep. Arctic Fish. Working Group, Int. Counc. Explor. Sea. C.M.* ICES, 1965.
- 4) G. I. Murphy. A solution of the catch equation. *J. Fish. Res. Bd. Can.*, **22**: 191–202, 1965.
- 5) G. T. Evans. Rational approximations to solutions of the VPA equations. *Can. J. Fish. Aquat. Sci.*, **46**: 1274–1276, 1989.
- 6) K. R. Allen and W. S. Hearn. Some procedures for use in cohort analysis and other population simulations. *Can. J. Fish. Aquat. Sci.*, **46**: 483–488, 1989.
- 7) A. D. MacCall. Virtual population analysis (VPA) equations for nonhomogeneous populations, and a family of approximations including improvements on Pope's cohort analysis. *Can. J. Fish. Aquat. Sci.*, **43**: 2406–2409, 1986.
- 8) R. M. Cassie. Some uses of probability paper in the analysis of size frequency distributions. *Aust. J. Mar. Freshw. Res.*, **5**: 513–522, 1954.
- 9) D. K. Kimura and S. Chikuni. Mixtures of empirical distributions: An iterative application of the age-length key. *Biometrics*, **43**: 23–35, 1987.
- 10) J. Schnute and D. Fournier. A new approach to length-frequency analysis: Growth structure. *Can. J. Fish. Aquat. Sci.*, **37**: 1337–1351, 1980.

- 11) M. Terceiro and J. S. Idoine. A practical assessment of the performance of Shepherd's length composition analysis (SRLCA): Application to Gulf of Maine northern shrimp *pandalus borealis* survey data. *Fishery Bulletin*, **88**: 761–773, 1990.
- 12) D. Pauly. A review of the ELEFAN system for analysis of length-frequency data in fish and aquatic invertebrates. In D. Pauly and G. R. Margan, editors, *Length Based Methods in Fisheries Research*, **13**: 7–34. ICLARM Conf. Proc., Manila, 1987.
- 13) M. Basson, A. A. Rosenberg, and J. R. Beddington. The accuracy and reliability of two new methods for estimating growth parameters from length-frequency data. *J. Cons. Int. Explor. Mer*, **44**: 277–285, 1988.
- 14) D. A. Fournier, J. R. Sibert, J. Majkowski, and J. Hampton. MULTIFAN a likelihood-based method for estimating growth parameters and age composition from multiple length frequency data sets illustrated using data for southern bluefin tuna *thunnus maccoyii*. *Can. J. Fish. Aquat. Sci.*, **47**: 301–317, 1990.
- 15) I. Martin and R. M. Cook. Combined analysis of length and age-at length data. *J. Cons. Int. Explor. Mer*, **46**: 178–186, 1990.
- 16) R. Jones. *Assessing the effects of changes in exploitation pattern using length composition data*. FAO Fisheries Technical Paper 256. FAO, Rome, 1984.
- 17) 白木原国雄. 体長組成データを用いたマッコウクジラの資源動態解析. 桜本和美, 加藤秀弘, 田中昌一 (編), 鯨類資源の研究と管理, pp. 147–158. 恒星社厚生閣, 東京, 1991.
- 18) Á. Fridriksson. On the calculation of age-distribution within a stock of cod by means of relatively few age-determinations as a key to measurements on a large scale. *Rapp. P. V. Réun. Cons. Int. Explor. Mer*, **86**: 5, 1934.
- 19) S. Tanaka. Precision of age-composition of fish estimated by double sampling method using the length for stratification. *Bull. Japan. Soc. Sci. Fish.*, **19**: 657–670, 1953.
- 20) 田中昌一. 水産資源学総論, 新水産学全集, 第8巻. 恒星社厚生閣, 東京, 1985.
- 21) 真子 渺・松宮義晴. 銘柄組成による年齢組成推定法. 西水研研報, **50**: 1–8, 1977.

- 22) V. Hasselblad. Estimation of parameters for a mixture of normal distributions. *Technometrics*, **8**: 431-444, 1966.
- 23) T. Akamine and Y. Matsumiya. Mathematical analysis of age-length key method for estimating age composition from length composition. *Bull. Japan Sea Natl. Fish. Res. Inst.*, **42**: 17-24, 1992.
- 24) H. J. Buchanan-Wollaston and W. C. Hodgson. A new method of treating frequency curves in fishery statistics, with some results. *J. Cons. Int. Explor. Mer*, **4**: 207-225, 1929.
- 25) 田中昌一. Polymodal な度数分布の一つの取扱方及びそのキダイ体長組成解析への応用. 東海区水産研究所報告, **14**: 1-13, 1956.
- 26) 岡正雄. 統計面から見た東支那海における黄鯛の生態學的研究-ii. 黄鯛 (*Taius tumifrons*) の成長に就いて. 長崎大学水産研究報告, **2**: 8-25, 1954.
- 27) C. G. Bhattacharya. A simple method of resolution of a distribution into Gaussian components. *Biometrics*, **23**: 115-135, 1967.
- 28) P. D. M. Macdonald and T. J. Pitcher. Age-groups from size-frequency data: A versatile and efficient method of analyzing distribution mixtures. *J. Fish. Res. Bd Can.*, **36**: 987-1001, 1979.
- 29) D. Pauly and N. David. ELEFAN I, a BASIC program for the objective extraction of growth parameters from length-frequency data. *Meersforsch.*, **28**: 205-211, 1981.
- 30) 赤嶺達郎, 加藤史彦. 複合正規分布のパラメータ推定. 東海区水産研究所数理統計部 (編), パソコンによる資源解析プログラム集, pp. 134-143. 東海区水産研究所, 東京, 1988.
- 31) E. Tanaka and S. Tanaka. A method for estimating age-composition from length-frequency by using stochastic growth equation. *Nippon Suisan Gakkaishi*, **56**: 1209-1218, 1990.
- 32) J. G. Pope and J. G. Shepherd. A simple method for the consistent interpretation of catch-at-age data. *J. Cons. Int. Explor. Mer*, **40**: 176-184, 1982.

- 33) J. R. Beddington and J. G. Cooke. Development of an assessment technique for male sperm whales based on the use of length data from the catches, with special reference to the north-west Pacific stock. *Rep. Int. Whal. Commn*, **31**: 747-760, 1981.
- 34) K. Shirakihara and S. Tanaka. Reconstruction of the Beddington and Cooke age-specific model and some comments on their models for sperm whales. *Rep. Int. Whal. Commn*, **32**: 243-247, 1982.
- 35) K. Shirakihara and S. Tanaka. Population estimation of the Northwest Pacific stock of sperm whales. *Rep. Int. Whal. Commn*, **32**: 249-250, 1982.
- 36) J. G. Cooke and J. R. Beddington. Further development of an assessment technique for male sperm whales based on length data from the catches. *Rep. Int. Whal. Commn*, **32**: 239-241, 1982.
- 37) K. Shirakihara and S. Tanaka. An alternative length-specific model and population assessment for the Western North Pacific sperm whales. *Rep. Int. Whal. Commn*, **33**: 287-294, 1983.
- 38) K. Shirakihara and S. Tanaka. Simulation studies on length-specific population assessment techniques for Western North Pacific sperm whales. *Rep. Int. Whal. Commn*, **34**: 259-264, 1984.
- 39) K. Shirakihara, S. Tanaka, and T. Nakano. Further simulation studies on length-specific population assessment techniques for Western North Pacific sperm whales. *Rep. Int. Whal. Commn*, **35**: 199-203, 1985.
- 40) J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, **7**: 308-313, 1965.
- 41) 奥村晴彦. パソコンによるデータ解析. 技術評論社, 東京, 1986.
- 42) 島本信夫. 複数年級群のための virtual population analysis -1. 東海区水産研究所数理統計部 (編), パソコンによる資源解析プログラム集, pp. 134-143. 東海区水産研究所, 東京, 1988.
- 43) 藤本武. アワビ類の年齢査定の一方法について. 水産増殖, **15**: 19-22, 1967.

- 44) 松石隆, 岸野洋久, 菅野泰次. 年齢の読み飛ばしを考慮した成長曲線の推定法. 日本水産学会誌, **62**: 28–31, 1996.
- 45) 松石隆, 斎藤和敬, 菅野泰次. アワビの成長曲線について. 北海道大学水産学部研究彙報, **46**: 53–62, 1995.
- 46) D. B. DeLury. On the estimation of biological populations. *Biometrics*, **3**: 145–167, 1947.
- 47) Y. Matsumiya and T. Matsuishi. Estimation of population parameters by optimizing catch effort allocation. *Res. Popul. Ecol.*, **31**: 305–310, 1989.
- 48) 平山信夫, 山田作太郎, 菊池弘, 山田潤一. Delury法の修正とアワビ採捕漁業への応用. 日本水産学会誌, **55**: 409–416, 1989.
- 49) B. Zhao, E. Tanaka, and J. Yamada. Stock assessment of abalone in Kisakata, Akita Prefecture. *Nippon Suisan Gakkaishi*, **59**: 1823–1830, 1993.
- 50) 趙柏賢. 秋田県磯根漁場におけるアワビ漁業管理に関する研究. PhD thesis, 東京水産大学, 1991.
- 51) 秋田県水産振興センター. 放流漁場高度利用技術開発事業(アワビ類), 1988.
- 52) R. Hilborn. Current and future trends in fisheries stock assessment and management. *S. Afr. J. mar. Sci.*, **12**: 975–988, 1992.
- 53) J. A. Gulland. *Fish Stock Assessment: A Manual of Basic Methods*. Wiley, New York, 1993.
- 54) 前田辰昭, 高木省吾, 亀井佳彦, 梶原善之, 目黒敏美, 中谷敏邦. スケトウダラ調査研究の歴史と問題点. 北水試研報, **42**: 1–14, 1993.
- 55) 前田辰昭, 高橋豊美, 伊地知誠, 平川英人, 上野元一. 噴火湾周辺海域におけるスケトウダラの漁場学的研究–ii産卵期. 日本水産学会誌, **42**: 1213–1221, 1976.
- 56) 前田辰昭, 中谷敏邦, 高橋豊美, 上野元一. スケトウダラ稚魚の沿岸水域での生活. 水産海洋研究会報, **34**: 81–85, 1979.

57) 北海道立函館水産試験場室蘭支場. 道南太平洋海域のスケトウダラ漁況について, 1994.

付録

付録 A

A 系列のシミュレーション結果

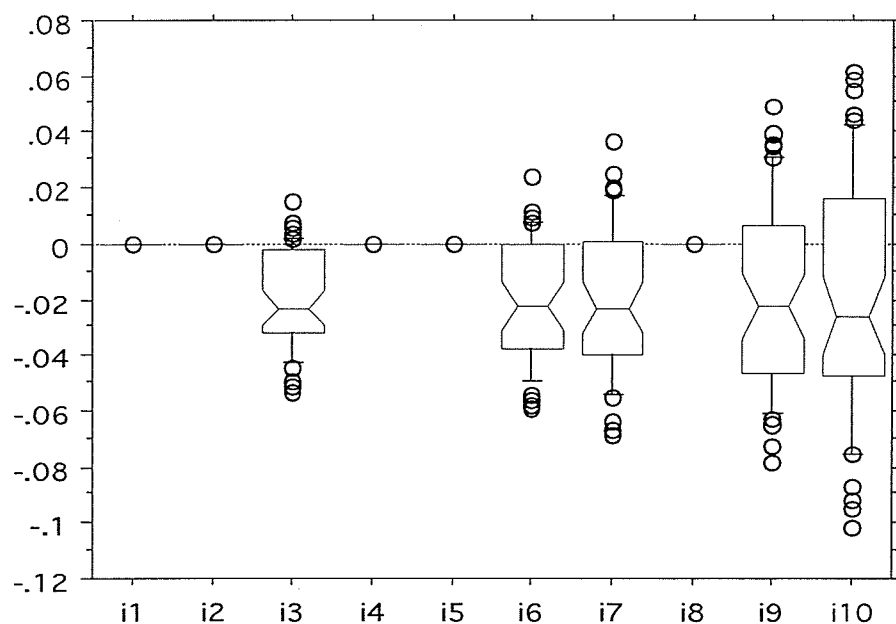


Fig. A.1: AR10の第5年目の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

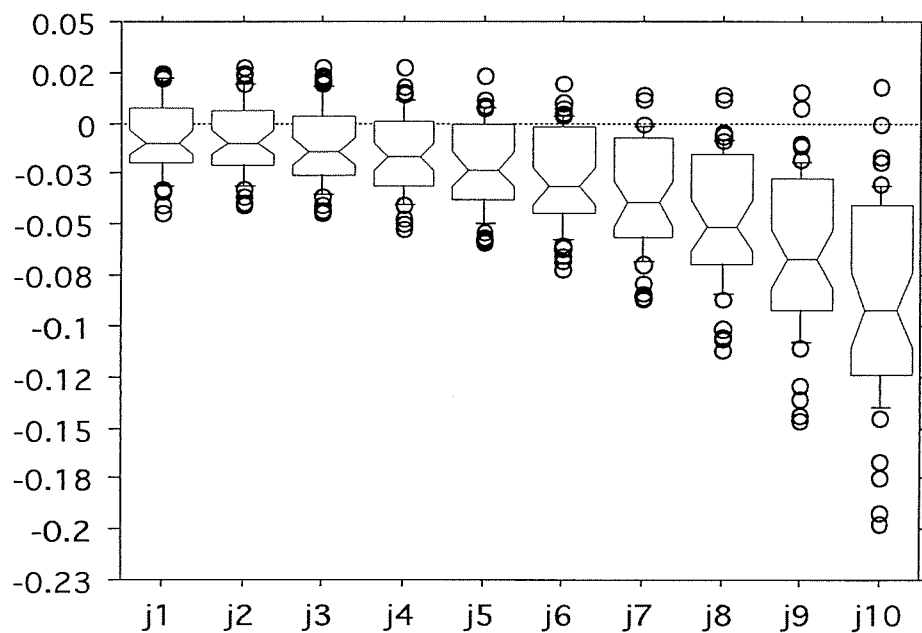


Fig. A.2: AR10の第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

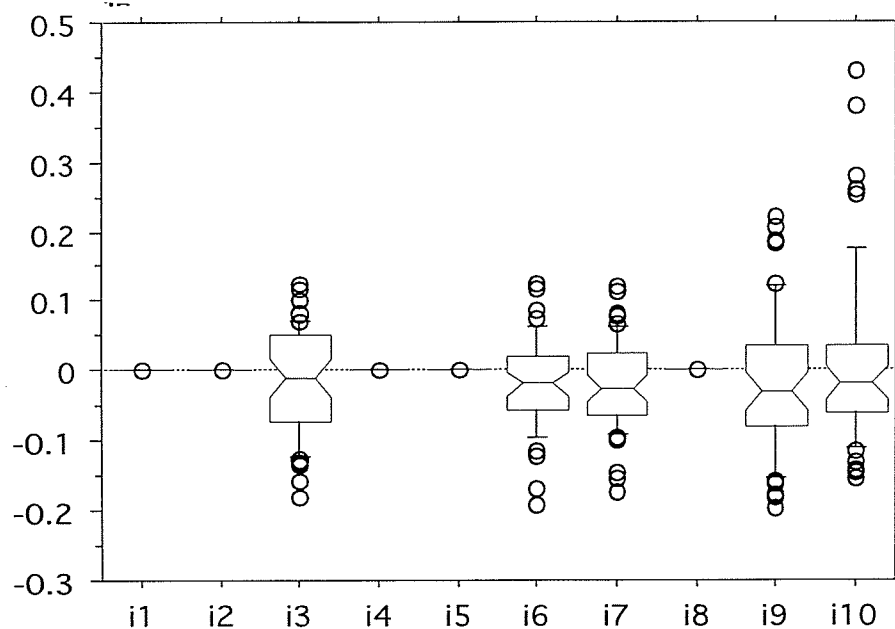


Fig. A.3: AM10の第5年目の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

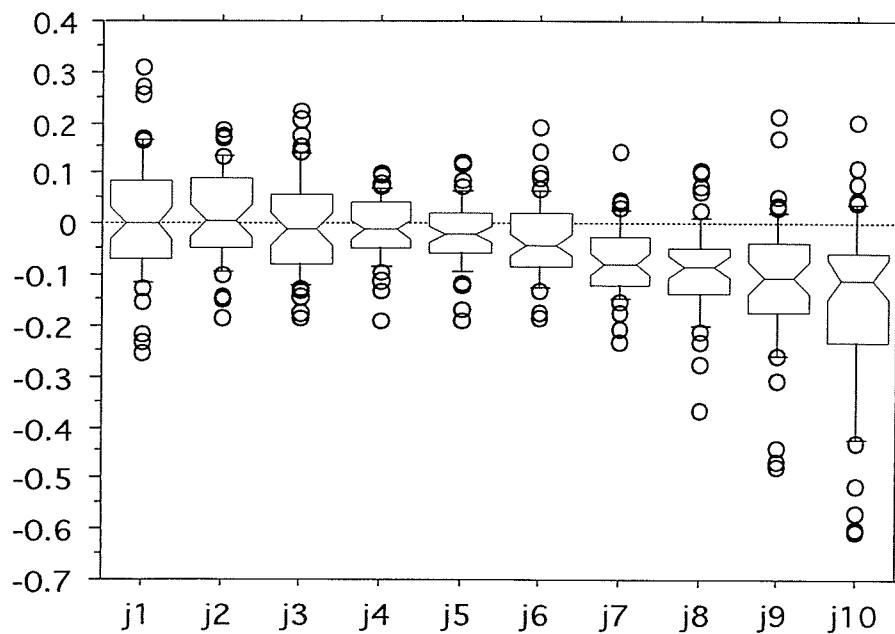


Fig. A.4: AM10 の第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

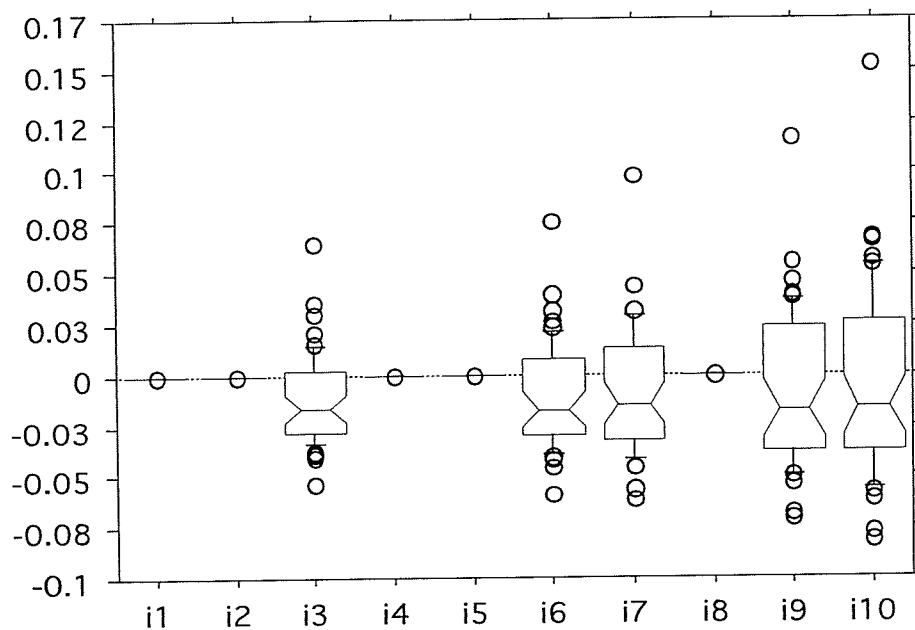


Fig. A.5: AF10 の第 5 年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

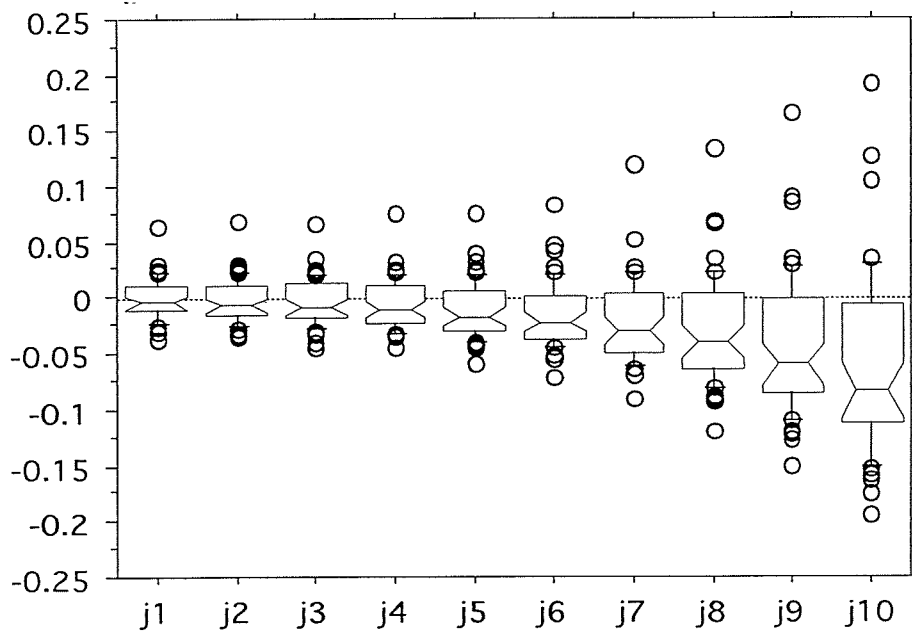


Fig. A.6: AF10の第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

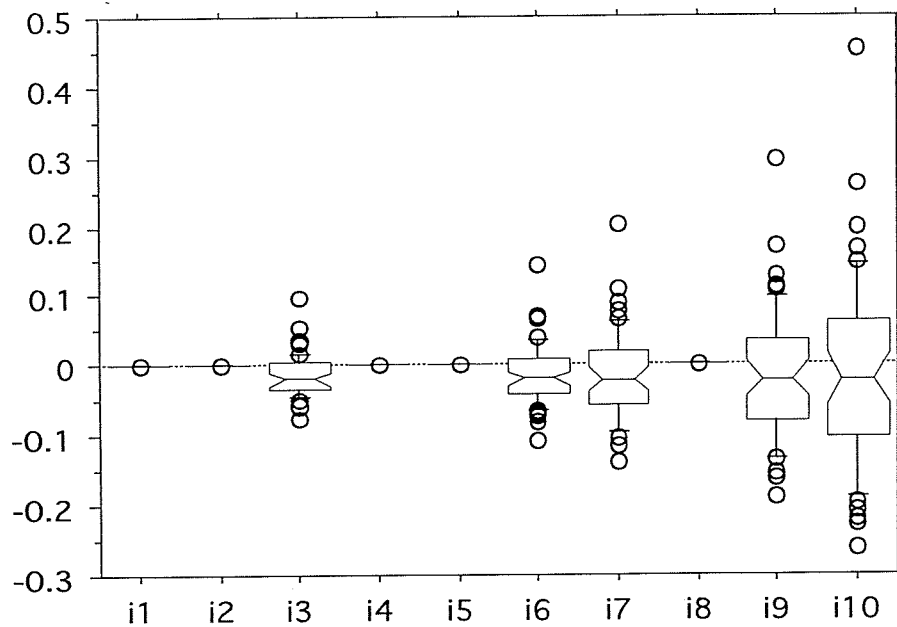


Fig. A.7: AS10の第5年目の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

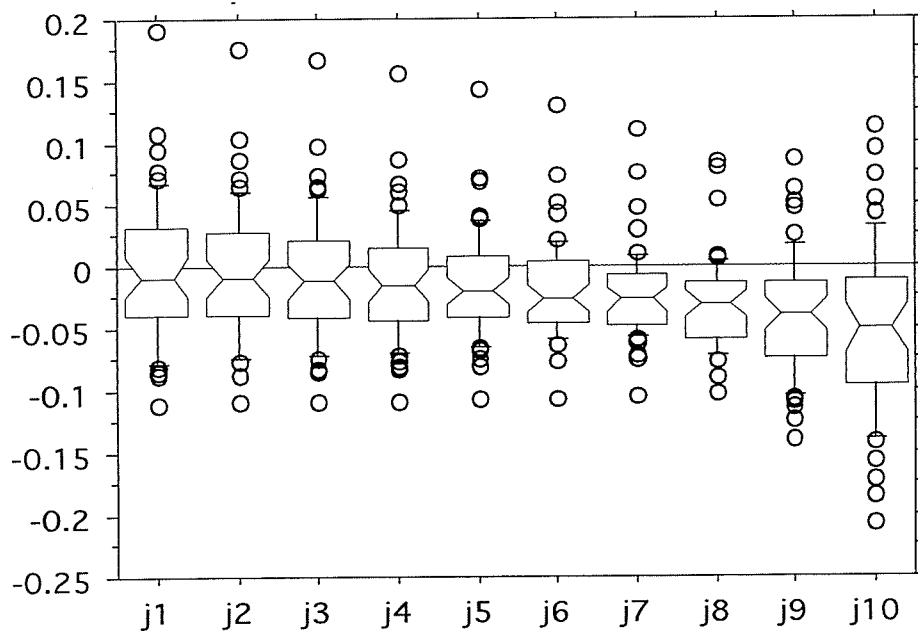


Fig. A.8: AS10の第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

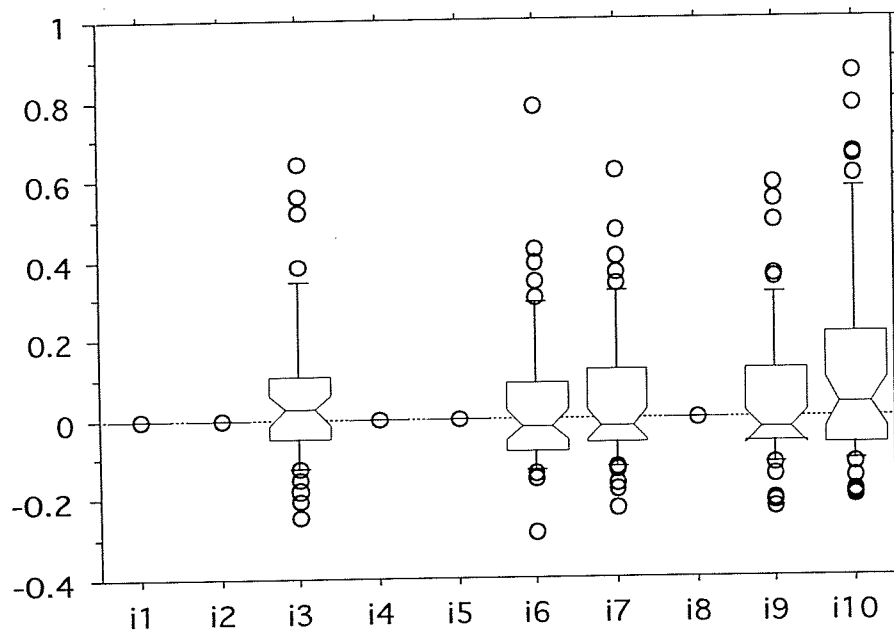


Fig. A.9: AG10第5年目、推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

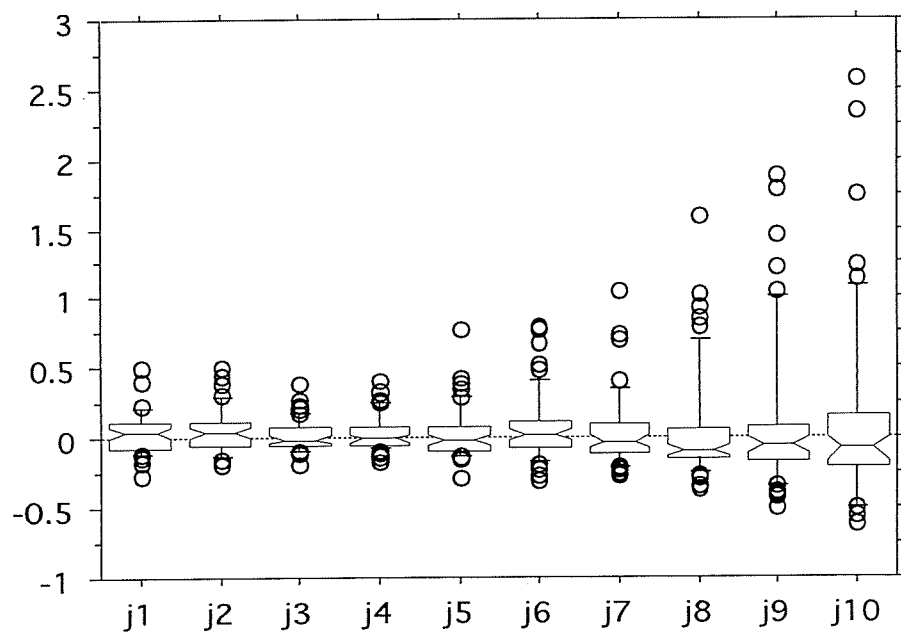


Fig. A.10: AG10 第6体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

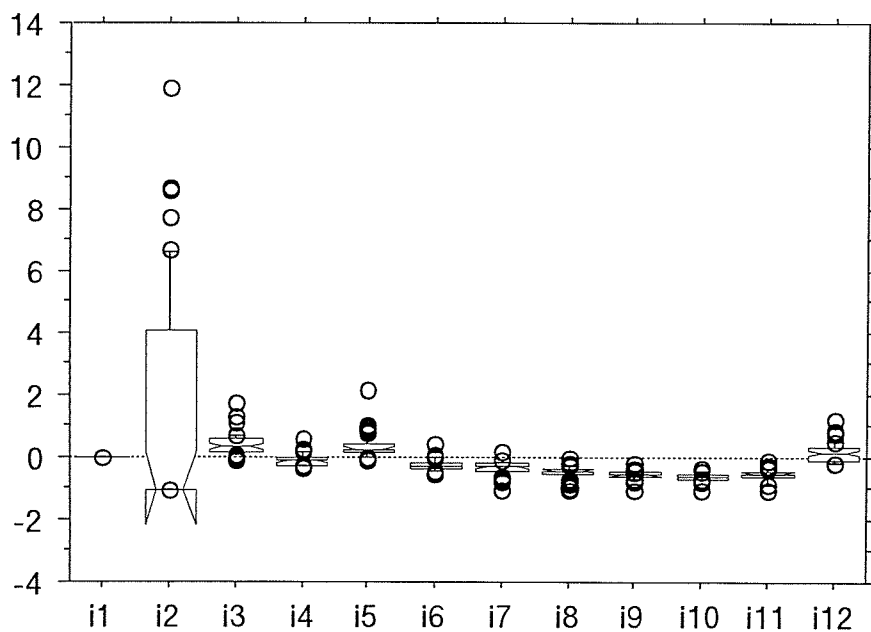


Fig. A.11: AP10 の第5年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

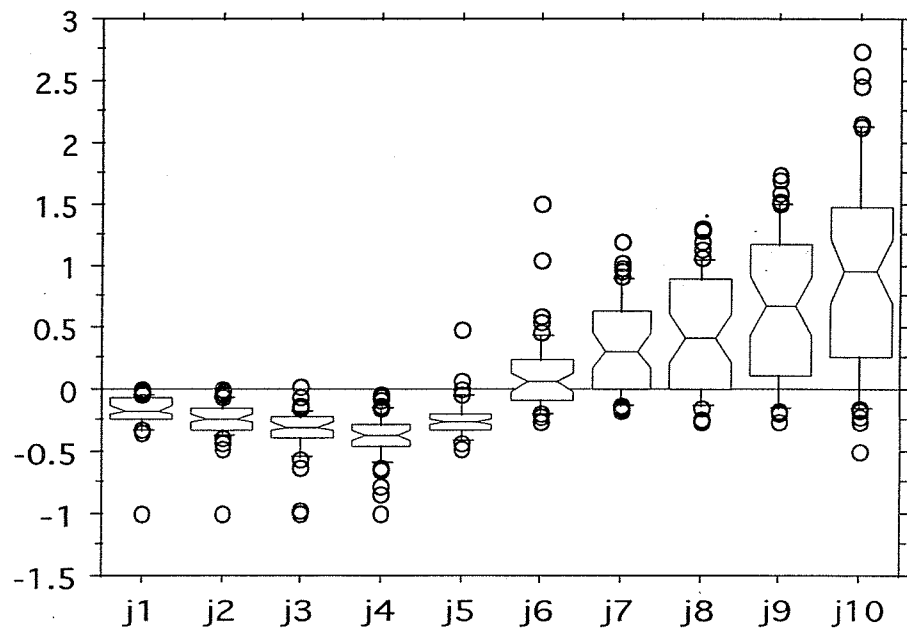


Fig. A.12: AP10 の第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

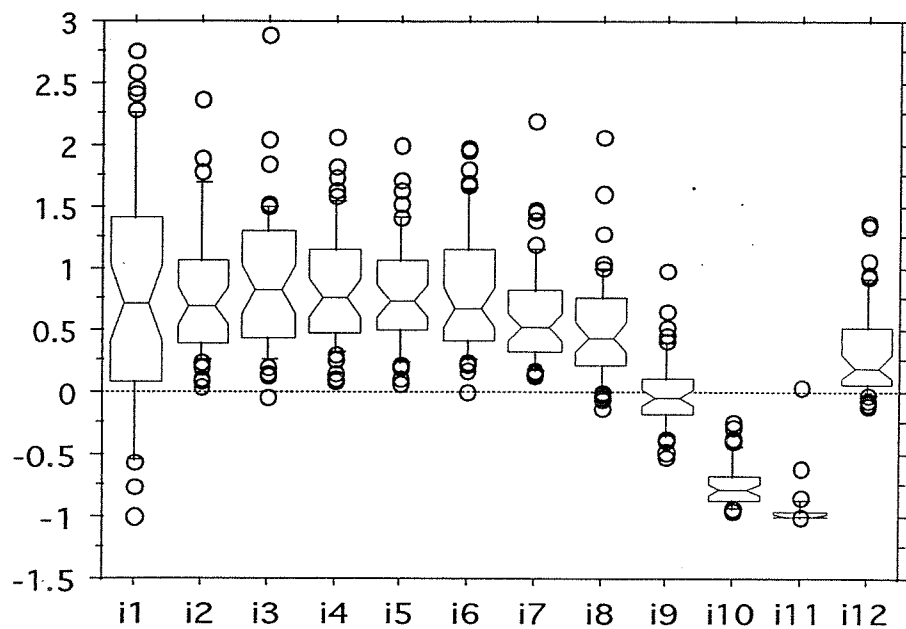


Fig. A.13: AA10 の第 5 年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

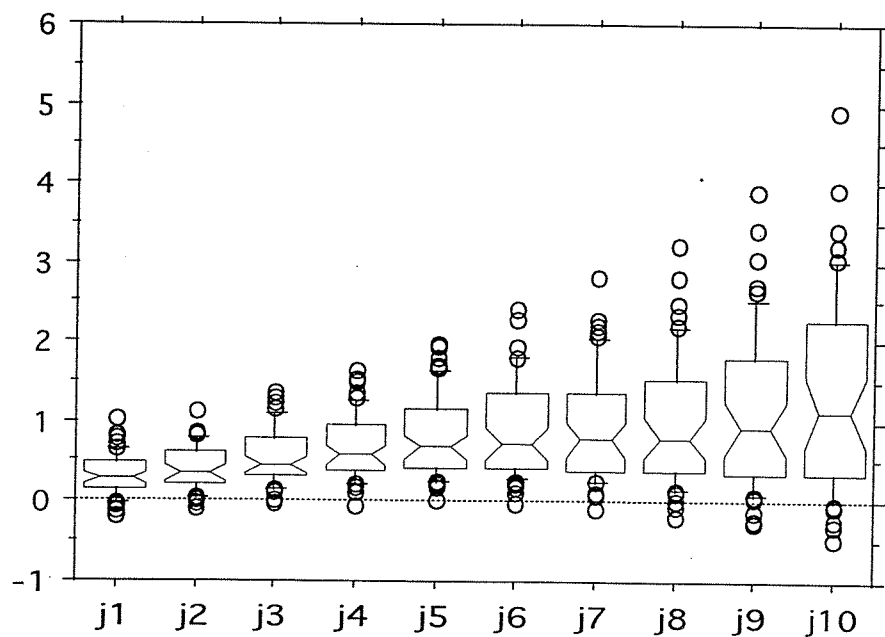


Fig. A.14: AA10 の第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

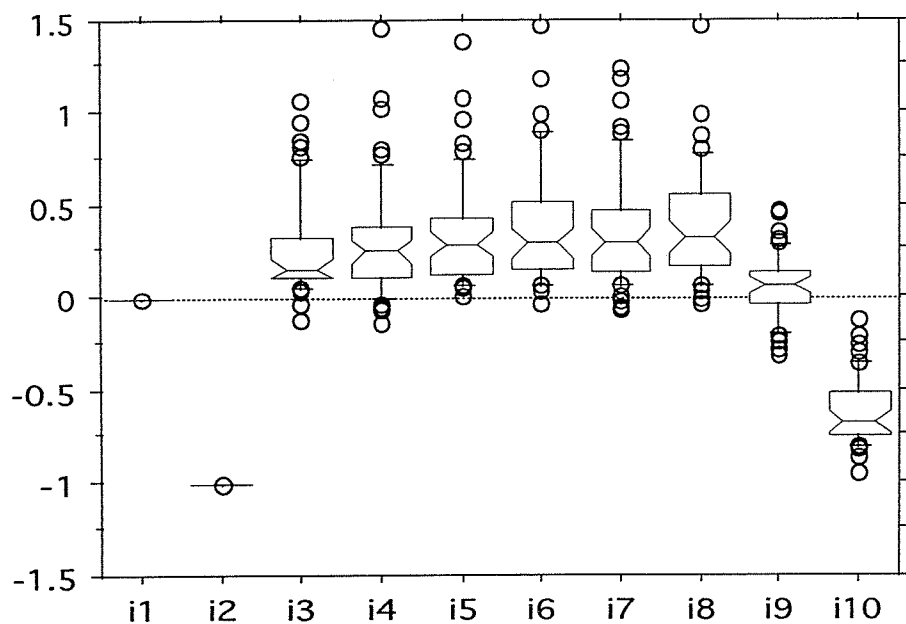


Fig. A.15: AB10 の第 5 年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

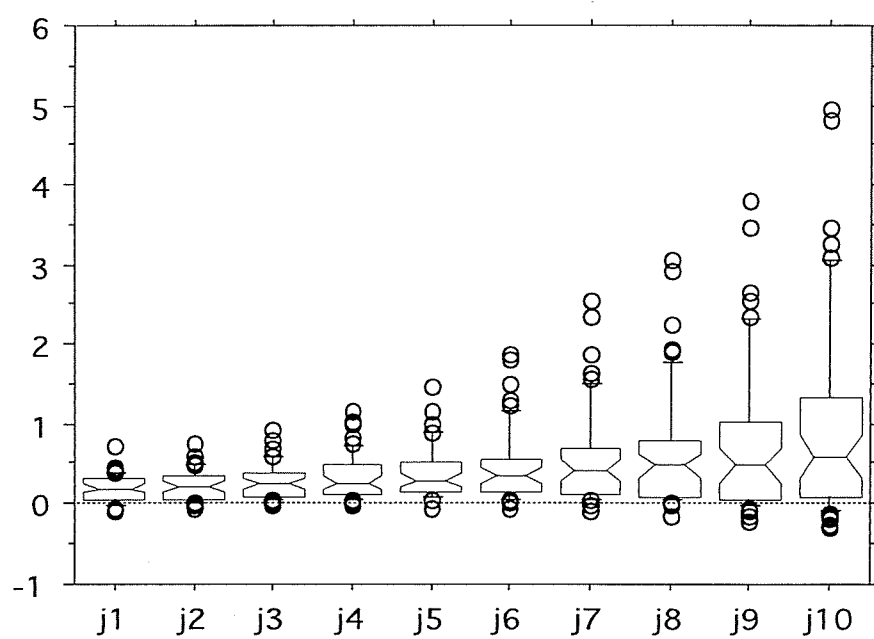


Fig. A.16: AB10 の第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

付録 B

B 系列のシミュレーション結果

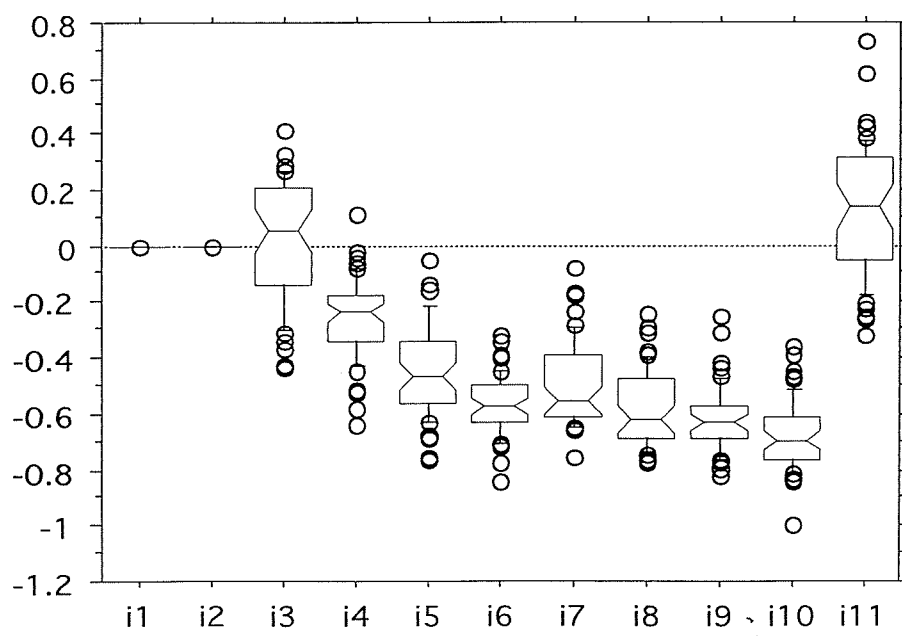


Fig. B.1: BR10 の第 5 年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

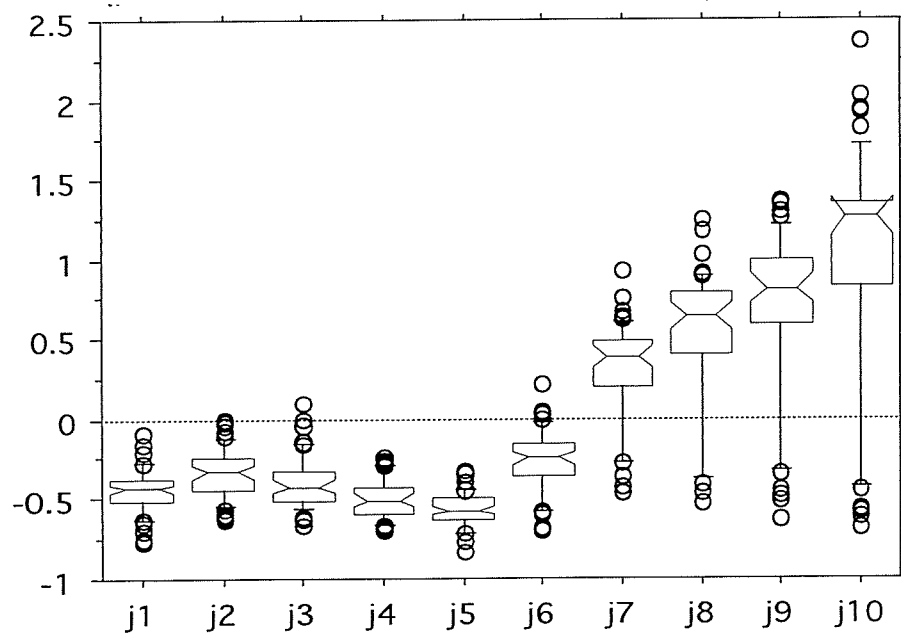


Fig. B.2: BR10 の第6体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

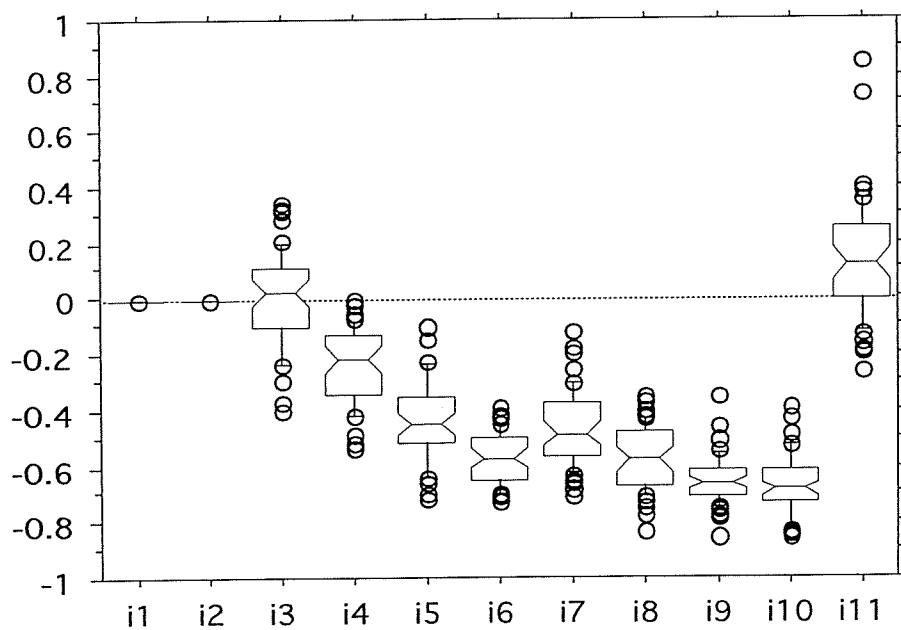


Fig. B.3: BM10 の第5年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

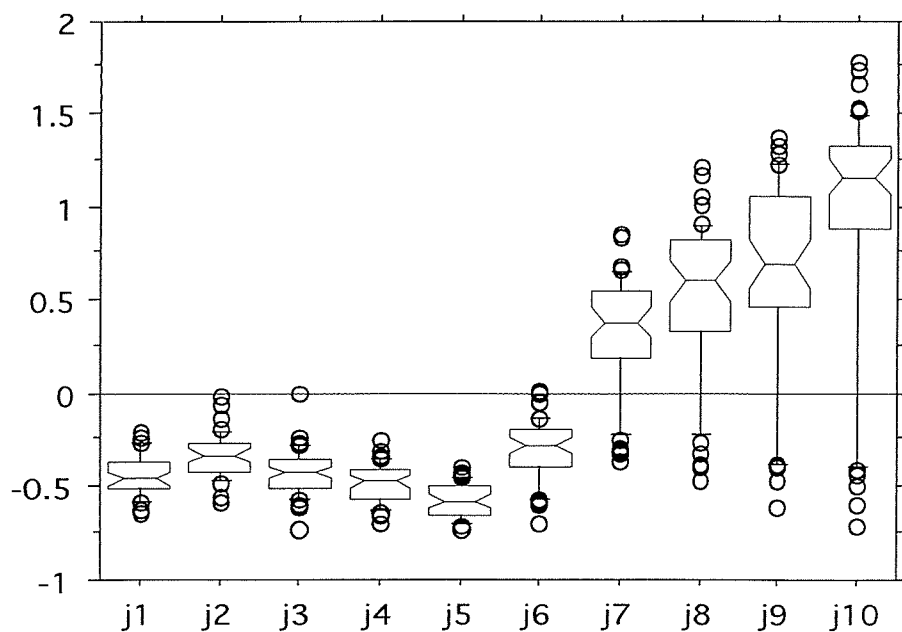


Fig. B.4: BM10 の第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

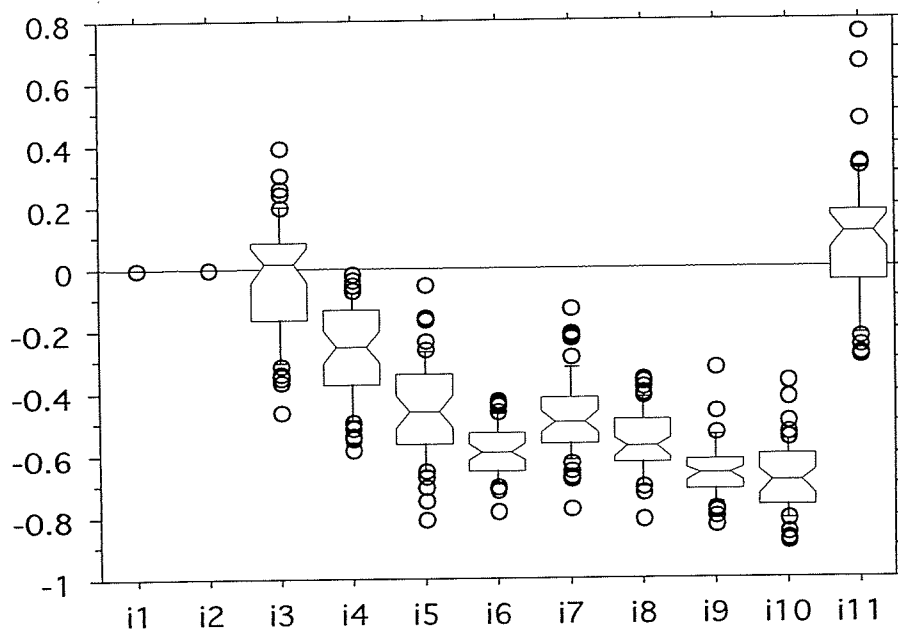


Fig. B.5: BF10 の第 5 年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

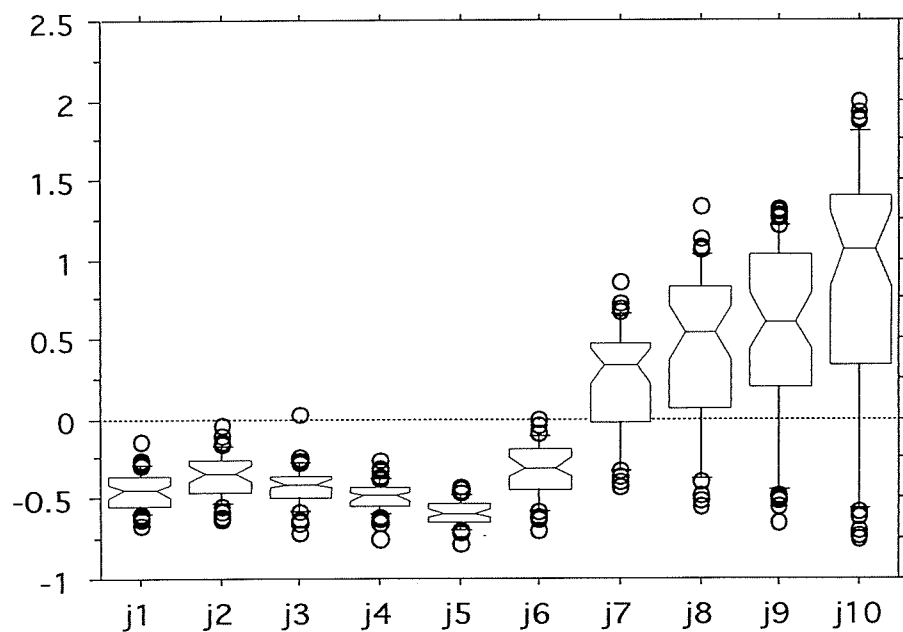


Fig. B.6: BF10 の第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

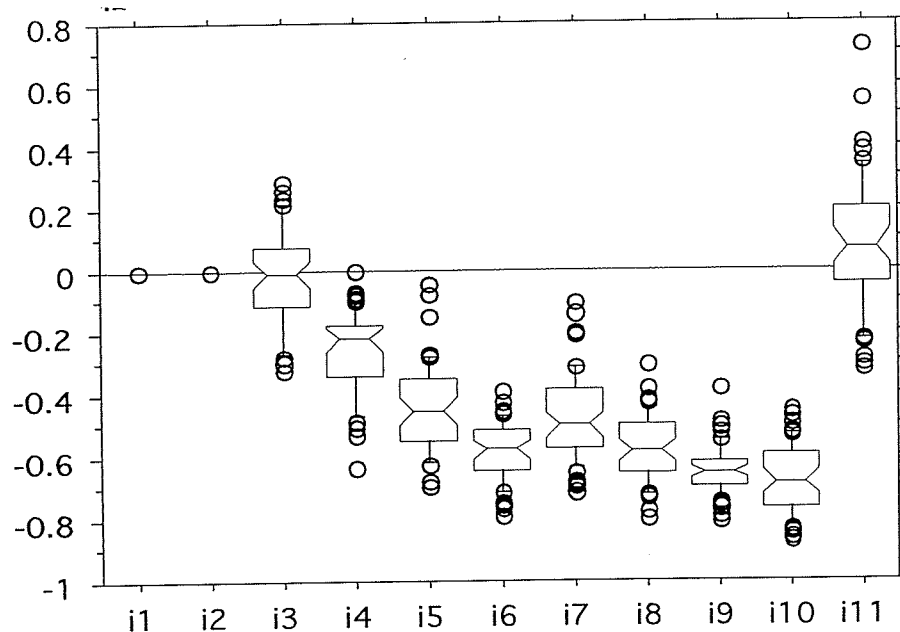


Fig. B.7: BS10 の第 5 年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

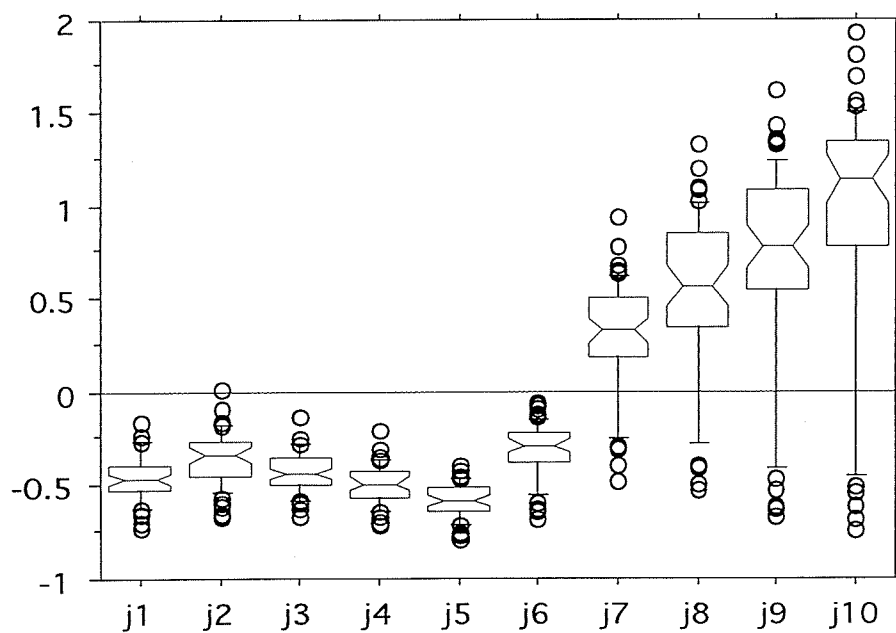


Fig. B.8: BS10の第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

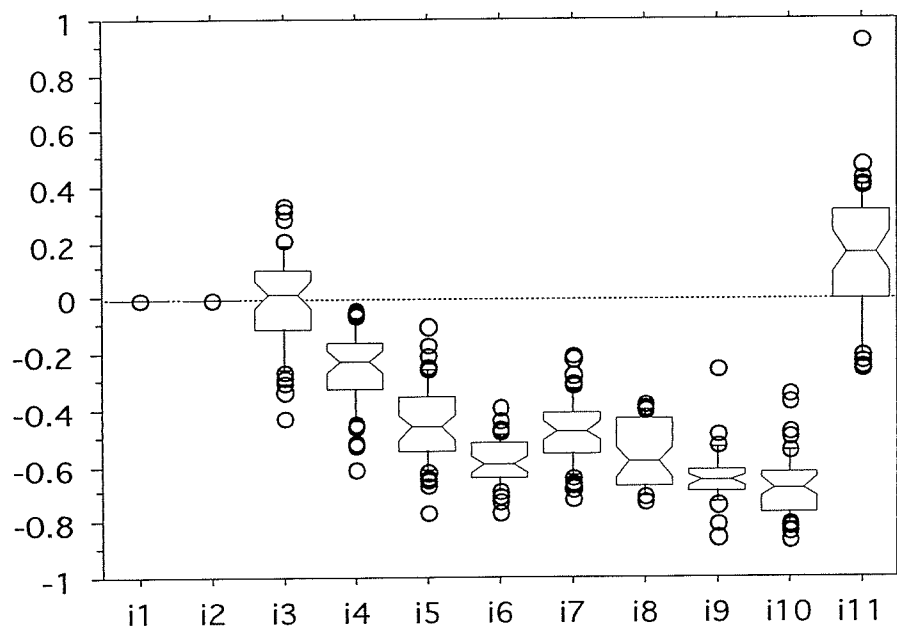


Fig. B.9: BG10第5年目、推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

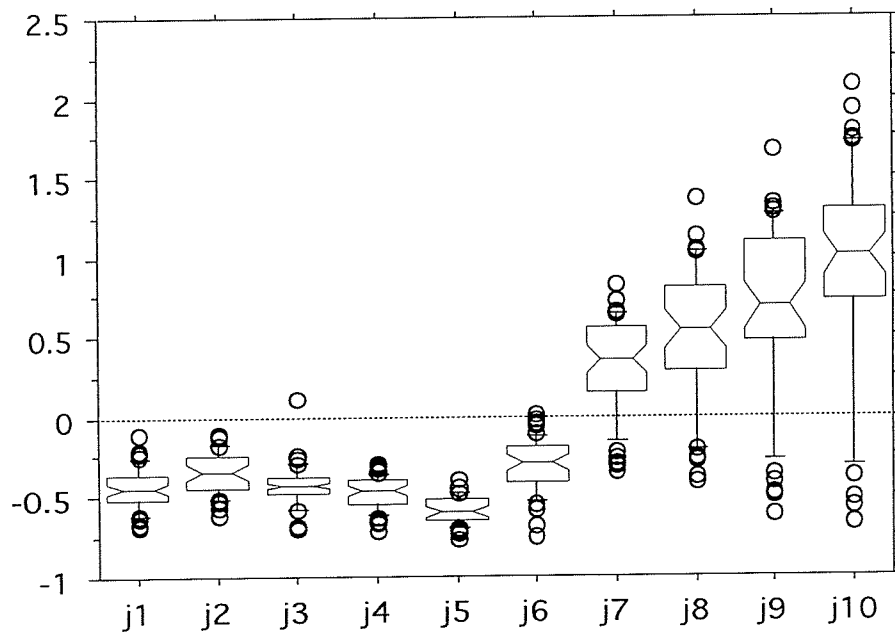


Fig. B.10: BG10 第6体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

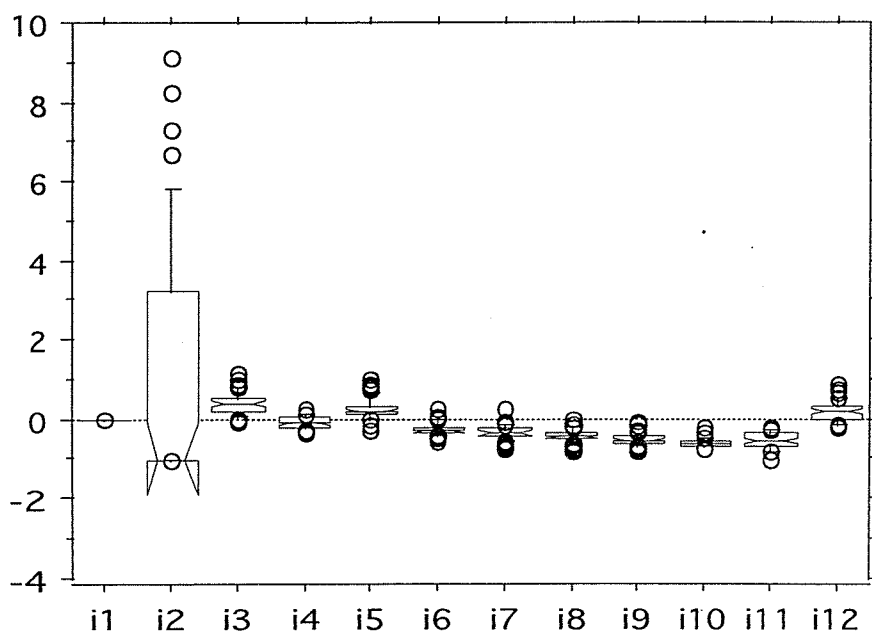


Fig. B.11: BP10 の第5年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

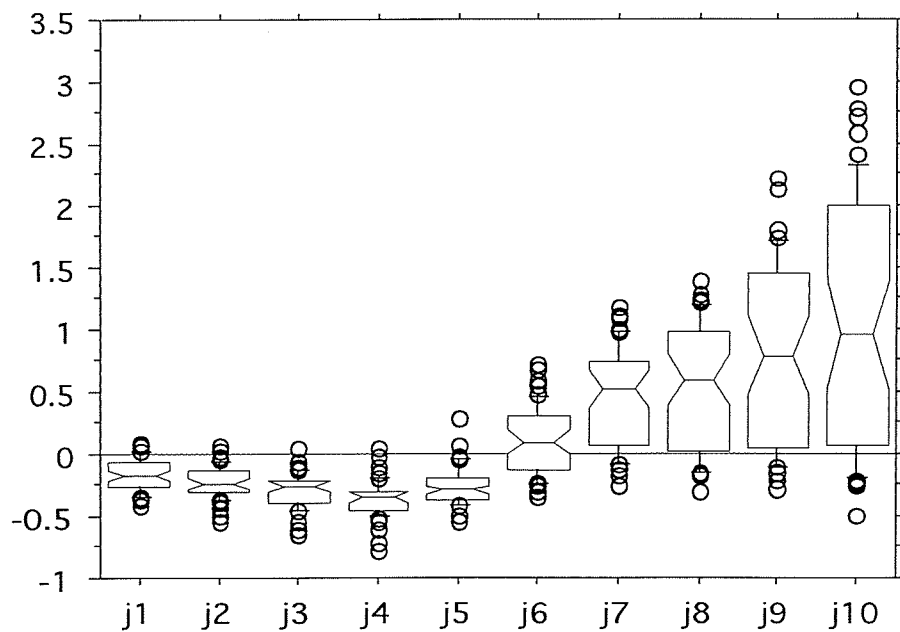


Fig. B.12: BP10の第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

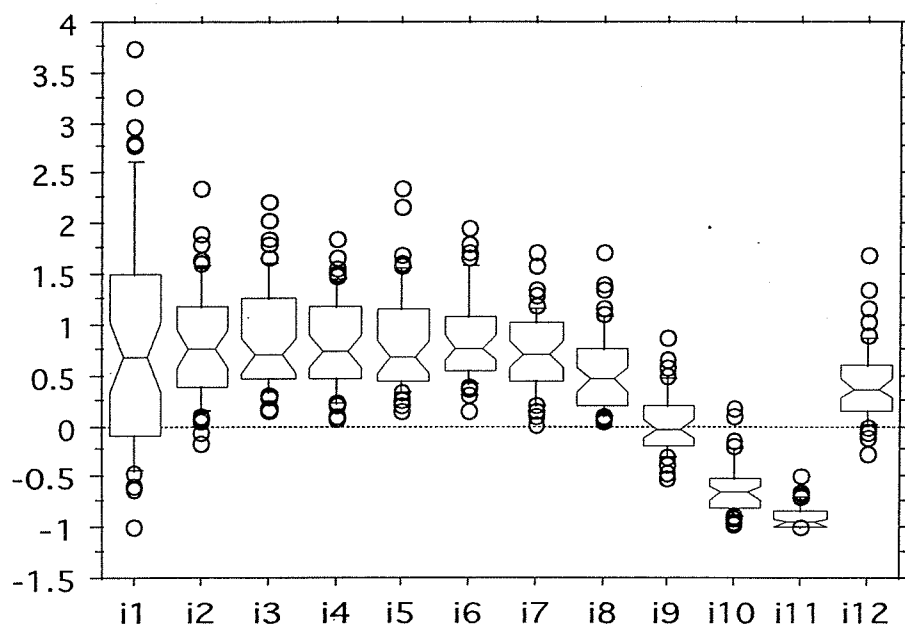


Fig. B.13: BA10の第5年目の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

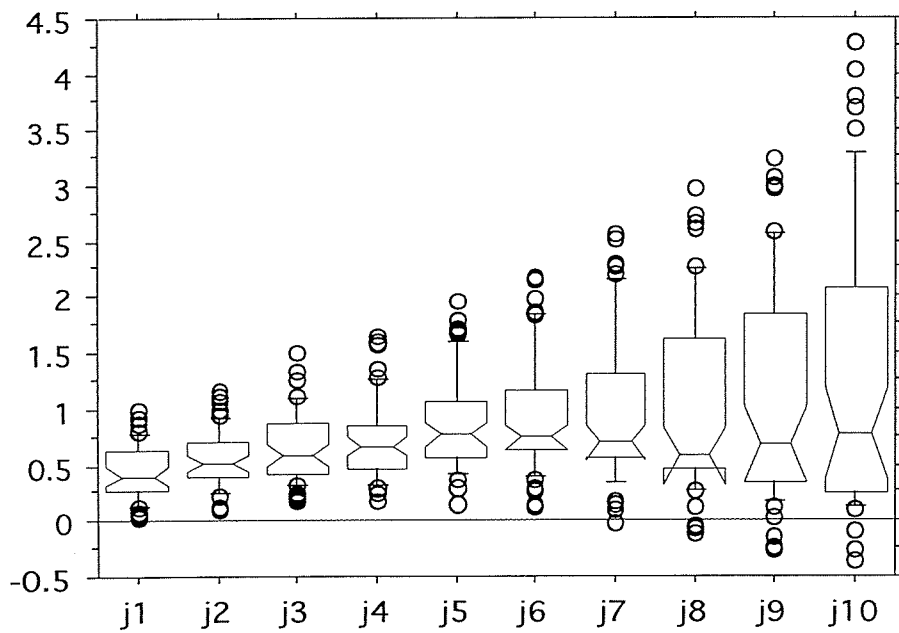


Fig. B.14: BA10の第6体長階級の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

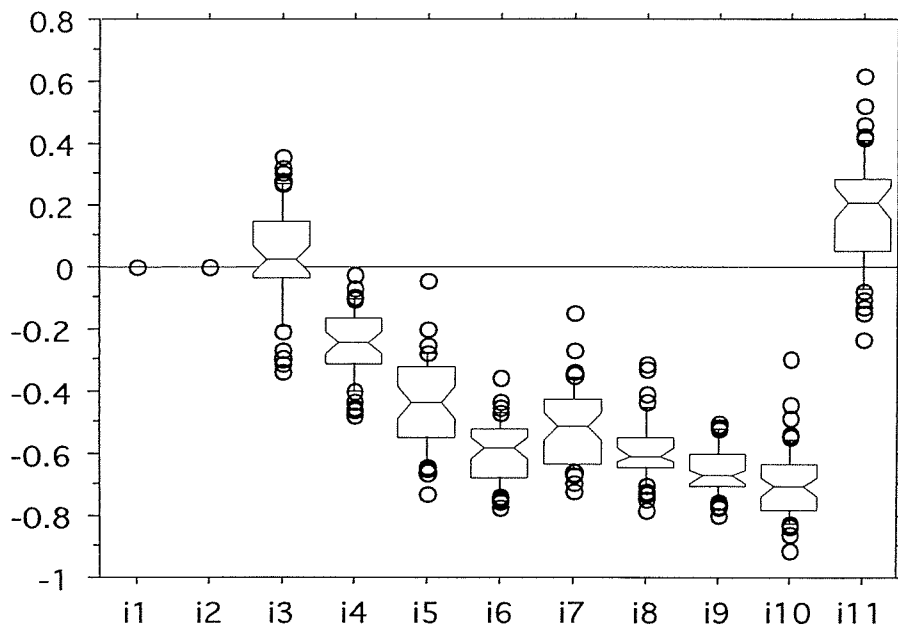


Fig. B.15: BB10の第5年目の推定結果。点は両端10%に含まれる値、棒の端は両端10%点、箱の端は両端25%点、くびれの中央は50%点を表す。

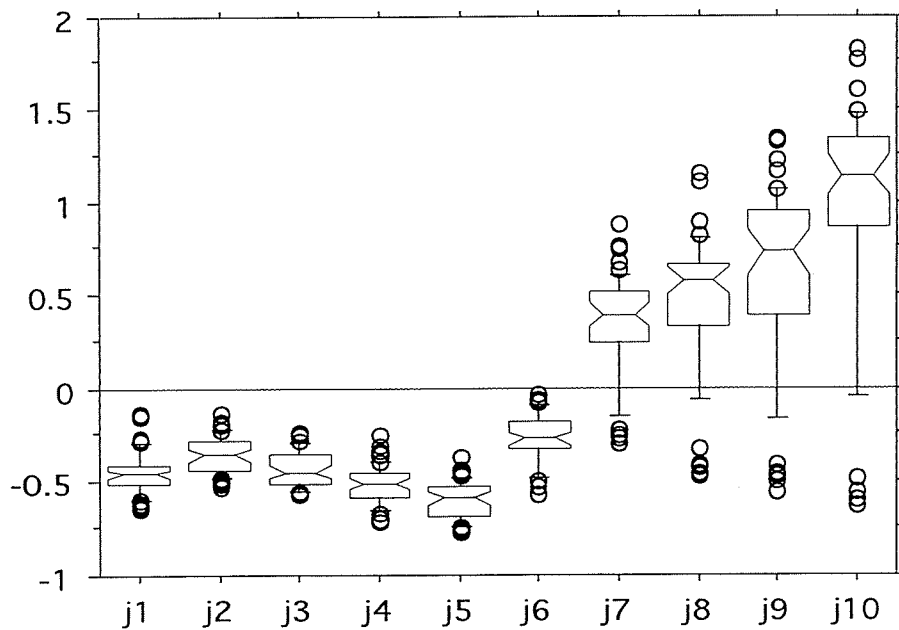


Fig. B.16: BB10 の第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

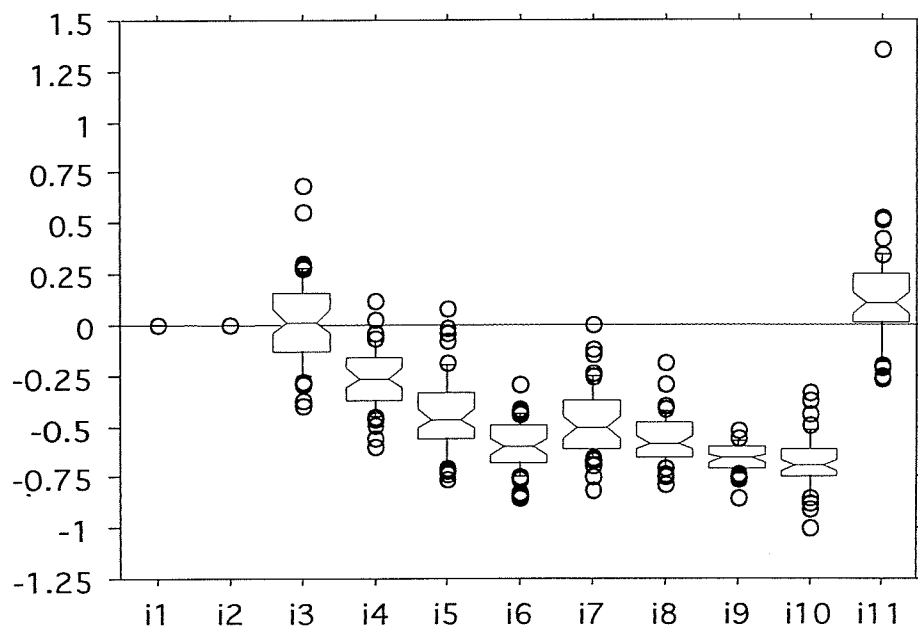


Fig. B.17: BT05 の第 5 年目の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

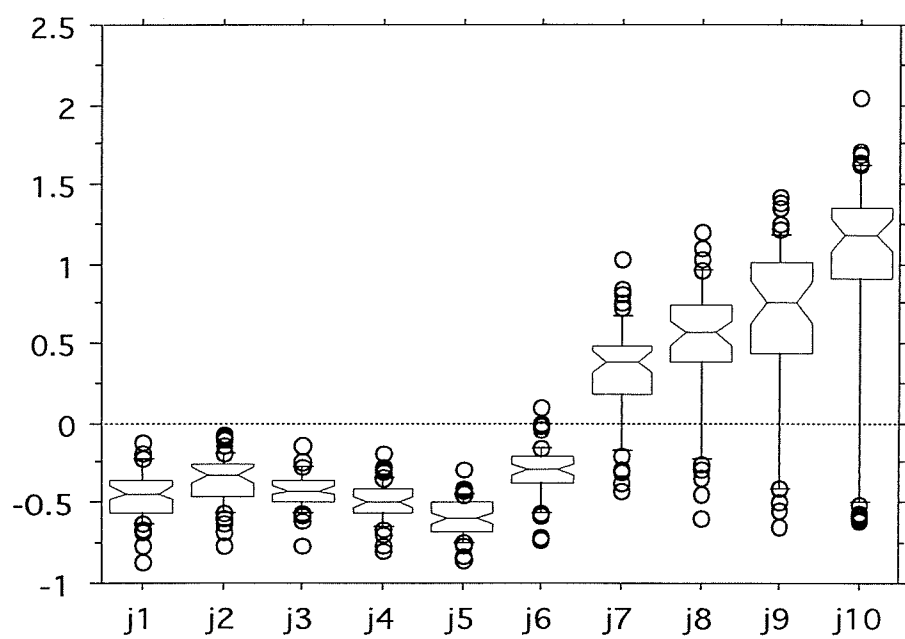


Fig. B.18: BT05 の第 6 体長階級の推定結果。点は両端 10%に含まれる値、棒の端は両端 10%点、箱の端は両端 25%点、くびれの中央は 50%点を表す。

付録 C

LPA プログラムリスト

```
c=====
      program lpa
      open (4, file = 'dat')
      open (7, file = 'log')
      open (8, file = 'rlt')
      open (9, file = 'dig')

      call input
      call multi
      stop
      end

c=====
      subroutine input
      real m, div, xref, fi, eps, ver, step, caut(20,50), trans(50,50)
      integer nyear, na, nr, ndiv, iw, idiv, iyear, idiv2
      character*20 gc(3)
      data gc/ '(von Bertalanffy)', '(Logistic)', '(Gompertz)'/

      common /blk1/nyear, na, nr
      common /blk4/m
      common /blk5/div, ndiv
      common /blk23/xref, fi
      common /blk31/eps, step
      common /blk21/caut
      common /blk22/trans

      ver = 4.0

      read (4, *) nyear, na
      read (4, *) m
      read (4, *) div, ndiv
      read (4, *) xref, fi
      read (4, *) eps, step
      do 1020 idiv = 1, ndiv
         read (4, *) (caut(iyear, idiv), iyear = 1, nyear)
1020 continue
      do 1030 idiv = 1, ndiv
         read (4, *) (trans(idiv2, idiv), idiv2 = 1, ndiv)
1030 continue
C      do 1010 iw = 7, 9, 2
      iw = 7
         write (iw, 2020) ' LPA RUNNING LOG '
         write (iw, *)
```

```

        write (iw, *) ' File: '
        write (iw, *)
        write (iw, *) ' Date: '
        write (iw, *)
        write (iw, 2080) ' Ver.: ', ver
        write (iw, *)
        write (iw, *) ' Machine: ', 'salmon'
        write (iw, 2010)
        write (iw, 2040) 'nyear : ', nyear
        write (iw, 2040) 'na      : ', na
        write (iw, 2050) 'm       : ', m
        write (iw, 2050) 'div     : ', div
        write (iw, 2040) 'ndiv    : ', ndiv
        write (iw, 2050) 'xref    : ', xref
        write (iw, 2050) 'fi      : ', fi
        write (iw, 2050) 'eps     : ', eps
        write (iw, 2050) 'step    : ', step
        write (iw, 2010)
1010 continue
C
C
2010 format (' ', 72('-'))
2020 format (' ', 26('*'), A, 26('*'))
2030 format (' ', A, 50(F5.3, 1X))
2040 format (' ', A, I10)
2050 format (' ', A, F10.3)
2060 format (' ', A, I10, ' ', A)
2070 format (' ', A, 10(F7.3, 1X))
2080 format (' ', A, F5.2)

        return
        end
=====
c=====
      double precision function func(param, m)

      real div, trans(50,50), s(50), f(20), z(100,100), fi
&      , xref, eps, ff(100, 100)
&      , n(20, 50), sum1, sum2, ch(20, 50), d(20, 50), nd(50)
&      , mort, caut(20, 50), step, fl, fu, sl, su
      real*8 param(50)
      integer ndiv, nyear, i, j, k, na, nr
&      , iflag, ierror, iterx, ip, m

      common /blk1/ nyear, na, nr
      common /blk4/ mort
      common /blk5/ div, ndiv
      common /blk21/ caut
      common /blk22/ trans
      common /blk23/ xref, fi
      common /blk25/ f, s, ch, n
      common /blk31/ eps, step
      common /blk32/ iflag, ierror, iterx
      common /blk40/ d

      fl = 1E-1
      fu = 5.0
      sl = 0.1
      su = 3.0

      do 1000 ip = 1, nyear - 1
        if (param(ip) .lt. fl) param(ip) = fl
        if (param(ip) .gt. fu) param(ip) = fu
        f(ip) = param(ip)
1000 continue

```



```

do 1010 ip = nyear, nyear+ndiv - 2
  if (param(ip) .lt. sl) param(ip) = sl
  if (param(ip) .gt. su) param(ip) = su
  s(ip - nyear + 1) = param(ip)
1010 continue
f(nyear) = fi
s(ndiv) = 1

do 1040 i = 1, nyear
  do 1050 j = 1, ndiv
    ff(i, j) = f(i) * s(j)
    z(i, j) = ff(i, j) + mort
1050 continue
1040 continue

do 1052 j = 1, ndiv
  n(nyear, j) = caut(nyear, j) * z(nyear, j)
  & / (1.0 - exp(-z(nyear, j))) / ff(nyear, j)
1052 continue
do 1054 i = 1, nyear
  n(i, ndiv) = caut(i, ndiv) * z(i, ndiv)
  & / ff(i, ndiv) / (1.0 - exp(-z(i, ndiv)))
1054 continue
do 1060 i = nyear-1, 1, -1
  sum1 = 0.0
  do 1062 j = 1, ndiv
    sum1 = sum1 + n(i+1, j)
1062 continue

do 1070 j = 1, ndiv-1
  nd(j) = 0.0
  do 1080 k = 1, ndiv
    nd(j) = nd(j) + trans(j, k) * n(i+1, k)
1080 continue
1070 continue
nd(ndiv) = n(i, ndiv)

sum2 = 0.0
do 1072 j = 1, ndiv - 1
  sum2 = sum2 + nd(j)
1072 continue
do 1074 j = 1, ndiv - 1
  n(i, j) = exp(z(i, j)) * nd(j)
C * sum2 / sum1
1074 continue
C write (6, *) sum1/sum2
1060 continue

do 1090 i = 1, nyear
  do 1100 j = 1, ndiv
    ch(i, j) = n(i, j) * (1.0 - exp(-z(i, j)))
    & * ff(i, j) / z(i, j)
1100 continue
1090 continue

func = 0.0
do 1112 i = 1, nyear
  do 1122 j = 1, ndiv
    d(i, j) = caut(i, j) - ch(i, j)
    d(i, j) = (caut(i, j) - ch(i, j)) / caut(i, j)
    func = func + d(i, j) * d(i, j)
1122 continue
1112 continue

return

```

```

end

c=====
      subroutine subsimp
      &      (m, initial, stepsize, tolerance,
      &      optparam, optfunc, iteration, iflag)
c Translated by T. Matsuishi(Univ. Hokkaido) from BASIC program in
c Paso-kon niyoru deta akiseki nyuumon by Okumura (1986)
c
c maximum number of parameters 50
c
c make real function func(parameter, m)
c      argument parameter(50) : value of parameters
c      m : number of parameters
c      func : value of object function
c
      real initial(50), stepsize(50), tolerance(50), optparam(50)
      &      , optfunc
      real*8 , vertex(51,50)
      &      , criterion(51)
      &      , s, v1, v2, f, v, vmax, vmin, vsum, p
      &      , parameter(50)
      double precision func
      integer m, j, kworst, kbest, iteration, flag, k, itermax
      &      , idebug, iflag
      external func

      itermax = 50000
      idebug = 0
      iflag = 0
c initial point
      s = sqrt(0.5)
      v1 = s * (sqrt(m+1.0) - 1.0) / dble(m)
      do 1100 j = 1, m
         vertex(1, j) = initial(j)
1100 continue
      do 1130 j = 1, m
         v2 = vertex(1, j) + v1 * stepsize(j)
         do 1150 k = 2, m+1
            vertex(k, j) = v2
1150 continue
         vertex(j+1, j) = v2 + s * stepsize(j)
1130 continue
c calculate objective function
      do 1200 k = 1, m+1
         do 1210 j = 1, m
            parameter(j) = vertex(k, j)
1210 continue
         f = func(parameter, m)
         criterion(k) = f
1200 continue

         do 1280 iteration = 1, itermax
            kworst = 1
            kbest = 1
            do 1310 k = 2, m+1
               if (criterion(k) .gt. criterion(kworst)) kworst = k
               if (criterion(k) .lt. criterion(kbest)) kbest = k
1310 continue
            if (idebug .ne. 0) then
               write (6, *)
               write (6, *) 'iteration: ', iteration
               write (6, *) 'criterion: ', criterion(kbest)

```

```

        write (6, *) 'parameters: '
        do 1390 j = 1, m
            write (6, *) j, vertex(kbest, j)
1390      continue
        endif
        flag = 0
        do 1430 j = 1, m
            vmax = vertex(1, j)
            vmin = vmax
            do 1450 k = 2, m+1
                v = vertex(k, j)
                if (v .gt. vmax) vmax = v
                if (v .lt. vmin) vmin = v
1450          continue
            if (vmax - vmin .gt. tolerance(j)) flag = 1
1430      continue
        if(flag .eq. 0) goto 1930
        do 1530 j = 1, m
            vsum = 0
            do 1550 k = 1, m+1
                if (k .ne. kworst) vsum = vsum + vertex(k, j)
1550          continue
            parameter(j) = 2.0 * vsum / m - vertex(kworst, j)
1530      continue
        f = func(parameter, m)
        if (f .lt. criterion(kbest)) then
            do 1630 j = 1, m
                p = 1.5 * parameter(j) - 0.5 * vertex(kworst, j)
                vertex(kworst, j) = parameter(j)
                parameter(j) = p
1630          continue
            criterion(kworst) = f
            f = func(parameter, m)
            if (f .lt. criterion(kworst))
                &      call replace(vertex, parameter, criterion, f, m, kworst)
            else if (f .lt. criterion(kworst)) then
                call replace (vertex, parameter, criterion, f, m, kworst)
            else
                do 1760 j = 1, m
                    parameter(j) = 0.75 * vertex(kworst, j)
                    &      + 0.25 * parameter(j)
1760          continue
                f = func(parameter, m)
                if (f .lt. criterion(kworst)) then
                    call replace (vertex, parameter, criterion, f, m, kworst)
                    goto 1910
                endif
                do 1810 k = 1, m+1
                    if (k .ne. kbest) then
                        do 1830 j = 1, m
                            parameter(j) = 0.5
                            &      * (vertex(k, j) + vertex(kbest, j))
                            vertex(k, j) = parameter(j)
1830          continue
                        f = func(parameter, m)
                        criterion(k) = f
                    endif
                continue
1810          continue
        endif
1910      continue
1280  continue
c      error
        iflag = 1

```

```

c      exit
1930 continue
      do 1932 j = 1, m
          optparam(j) = vertex(kbest, j)
1932 continue
      optfunc = criterion(kbest)

      return
      end

c=====
      subroutine replace(vertex, parameter, criterion, f, m, kworst)
      real*8 vertex(51, 50), parameter(50), criterion(51), f
      integer j, m, kworst

      do 2030 j = 1, m
          vertex(kworst, j) = parameter(j)
2030 continue
      criterion(kworst) = f

      return
      end

c=====

```

付録 D

LPA 精度評価プログラムリスト

```
C=====
C  MAIN PROGRAM

      program seido
      integer iter, iterm, iflag, ierror, ie, iem, iterx, ied, iedf
      common /blk28/ iterm, iter
      common /blk32/ iflag, ierror, iterx

      open (4, file = 'dat')
      open (7, file = 'log')
      open (9, file = 'inf')
      open (10, file= 'dig')

      ierror = 0
      ied = 0
      iedf = 0

3010 continue
      call input(ied)

      do 1010 iter = 1, iterm
        write (6, *) ' ITERATION : ', iter
        write (7, *) ' ITERATION : ', iter
        write (6, *) '      =      NDG'
        call fset
        call grow
        call trans
        call catch(ied, iedf)
        if (iedf .ne. 0) goto 3010
        write (6, *) '      ==      MUL'
        call multi
        write (6, *) '      ===     EVALU'
        call evalu
1010 continue

      if (ierror .eq. 0) then
        iterx = iterm
      else
        do 1020 ie = 1, 100
          iterx = iter
          iem = ierror
          ierror = 0
          do 1030 iter = iterx, iterx+iem-1
            write (6, *) ' ITERATION : ', iter
            write (7, *) ' ITERATION : ', iter
            write (6, *) '      =      NDG'
            call fset
            call grow
            call trans
            call catch(ied, iedf)
```

```

        write (6, *) ' == MUL'
        call multi
        write (6, *) ' === EVALU'
        call evalu
1030    continue
        if (ierror .eq. 0) goto 3020
1020    continue
endif
3020 continue
    call mse
    write (6, *) char(07)
    close (4)
    close (7)
    close (9)
    close (10)

    stop
    end

```

```

c=====
      subroutine input(ied)
      real cvnr, f(20), cvf, s(50), cvs, p(3), cvp(3), erra, errb
&      , m, cvm, div, l(30,12000,10), cvfm, xref, fi, eps, ver
&      , step
      integer igr, itr, ntr, i, j, k, nyear, na, nr, ndiv
&      , iter, iterm, ifm, ism, in, ismethod, ied
      character*20 gc(3)
      data gc/ '(von Bertalanffy)', '(Logistic)', '(Gompertz)'/

      common /blk1/nyear, na, nr
      common /blk2/l
      common /blk3/igr
      common /blk4/cvnr, m, cvm
      common /blk5/div, ndiv
      common /blk7/p, cvp, erra, errb
      common /blk8/f, s, cvf, cvs, cvfm
      common /blk11/itr, ntr
      common /blk23/xref, fi
      common /blk28/ iterm, iter
      common /blk31/ eps, step
      common /blk50/ ismethod

      ver = 6.2

      if (ied .ne. 0) then
        close (4)
        close (7)
        close (8)
        close (9)
        close (10)
        open (4, file = 'dat')
        open (7, file = 'log')
        open (8, file = 'rlt')
        open (9, file = 'inf')
        open (10, file= 'dig')
      endif

      write (9, 2020) ' SEIDO RUNNING LOG '
      write (9, *)
      write (9, *) ' File: '
      write (9, *)
      write (9, *) ' Date: '

```

```

write (9, *)
write (9, 2080) ' Ver.: ', ver
write (9, *)
write (9, *) ' Machine: ', 'salmon'
write (9, 2010)
read (4, *) iterm
write (9, 2040) 'iterm : ', iterm
read (4, *) nyear, na
write (9, 2040) 'nyear : ', nyear
write (9, 2040) 'na : ', na
read (4, *) nr, cvnr
write (9, 2040) 'nr : ', nr
write (9, 2050) ' cvnr : ', cvnr
read (4, *) m, cvm
write (9, 2050) 'm : ', m
write (9, 2050) ' cvm : ', cvm

read (4, *) div, ndiv
if (ied .ne. 0) ndiv = ndiv - ied
write (9, 2050) 'div : ', div
write (9, 2040) 'ndiv : ', ndiv
read (4, *) (f(i), i=1, nyear)

ifm = int((nyear-1)/10) + 1
if (ifm .eq. 1) then
  write (9, 2030) 'f : ', (f(i), i = 1, nyear)
else if (ifm .eq. 2) then
  write (9, 2030) 'f : ', (f(i), i = 1, 10)
  write (9, 2030) ' : ', (f(i), i = 11, nyear)
else
  write (9, 2030) 'f : ', (f(i), i = 1, 10)
  do 1020 in = 2, ifm-1
    write (9, 2030) ' : ', (f(i), i = (in-1)*10+1, in*10)
1020   continue
  write (9, 2030) ' : ', (f(i), i = (ifm-1)*10+1, nyear)
endif
read (4, *) (s(j), j=1, ndiv)

ism = int((ndiv-1)/10) + 1
if (ism .eq. 1) then
  write (9, 2030) 's : ', (s(j), j=1, ndiv)
else if (ism .eq. 2) then
  write (9, 2030) 's : ', (s(j), j=1, 10)
  write (9, 2030) ' : ', (s(j), j=11, ndiv)
else
  write (9, 2030) 's : ', (s(i), i = 1, 10)
  do 1030 in = 2, ism-1
    write (9, 2030) ' : ', (s(i), i = (in-1)*10+1, in*10)
1030   continue
  write (9, 2030) ' : ', (s(i), i = (ism-1)*10+1, ndiv)
endif
read (4, *) cvf, cvs, cvfm
write (9, 2050) ' cvf : ', cvf
write (9, 2050) ' cvs : ', cvs
write (9, 2050) ' cvfm : ', cvfm
read (4, *) igr

write (9, 2060) 'igr : ', igr, gc(igr)
read (4, *) (p(k), k=1, 3)
write (9, 2070) 'p : ', (p(k), k=1, 3)
read (4, *) (cvp(k), k=1, 3)
write (9, 2070) ' cv : ', (cvp(k), k=1, 3)
read (4, *) erra, errb
write (9, 2050) ' erra : ', erra

```

```

write (9, 2050) ' errb : ', errb
write (9, *)
read (4, *) itr, ntr

write (9, 2040) 'itr : ', itr
write (9, 2040) 'ntr : ', ntr
read (4, *) xref, fi
write (9, 2050) 'xref : ', xref
write (9, 2050) 'fi : ', fi
read (4, *) eps, step
write (9, 2050) 'eps : ', eps
write (9, 2050) 'step : ', step
write (9, 2010)

C
write (7, 2020) ' SEIDO RUNNING LOG '
write (7, *)
write (7, *) ' Date: '
write (7, *)
write (7, *) ' Ver.: ', ver
write (7, *)
write (7, *) ' Machine: ', 'salmon'
write (7, 2010)
write (7, 2040) 'iterm : ', iterm
write (7, 2040) 'nyear : ', nyear
write (7, 2040) 'na : ', na
write (7, 2040) 'nr : ', nr
write (7, 2050) ' cvnr : ', cvnr
write (7, 2050) 'm : ', m
write (7, 2050) ' cvm : ', cvm
write (7, 2050) 'div : ', div
write (7, 2040) 'ndiv : ', ndiv
ifm = int(nyear /10) + 1
if (ifm .eq. 1) then
  write (7, 2030) 'f : ', (f(i), i = 1, nyear)
else if (ifm .eq. 2) then
  write (7, 2030) 'f : ', (f(i), i = 1, 10)
  write (7, 2030) ' : ', (f(i), i = 11, nyear)
else
  write (7, 2030) 'f : ', (f(i), i = 1, 10)
  do 1025 in = 2, ifm-1
    write (7, 2030) ' : ', (f(i), i = (in-1)*10+1, in*10)
1025   continue
  write (7, 2030) ' : ', (f(i), i = (ifm-1)*10+1, nyear)
endif
if (ism .eq. 1) then
  write (7, 2030) 's : ', (s(j), j=1, ndiv)
else if (ism .eq. 2) then
  write (7, 2030) 's : ', (s(j), j=1, 10)
  write (7, 2030) ' : ', (s(j), j=11, ndiv)
else
  write (7, 2030) 's : ', (s(i), i = 1, 10)
  do 1035 in = 2, ism-1
    write (7, 2030) ' : ', (s(i), i = (in-1)*10+1, in*10)
1035   continue
  write (7, 2030) ' : ', (s(i), i = (ism-1)*10+1, ndiv)
endif
write (7, 2050) ' cvf : ', cvf
write (7, 2050) ' cvs : ', cvs
write (7, 2050) ' cvfm : ', cvfm
write (7, 2060) 'igr : ', igr, gc(igr)
write (7, 2070) 'p : ', (p(k), k=1, 3)
write (7, 2070) ' cv : ', (cvp(k), k=1, 3)
write (7, 2050) ' erra : ', erra

```



```

write (7, 2050) ' errb : ', errb
write (7, *)
write (7, 2040) 'itr  : ', itr
write (7, 2040) 'ntr  : ', ntr
write (7, 2050) 'xref : ', xref
write (7, 2050) 'fi   : ', fi
write (7, 2050) 'eps  : ', eps
write (7, 2010)

```

C

```

2010 format (' ', 72('-'))
2020 format (' ', 26('*'), A, 26('*'))
2030 format (' ', A, 50(F5.3, 1X))
2040 format (' ', A, I10)
2050 format (' ', A, F10.3)
2060 format (' ', A, I10, ' ', A)
2070 format (' ', A, 10(F7.3, 1X))
2080 format (' ', A, F5.2)

return
end

```

c=====

```

subroutine fset

real f(20), cvf, s(50), cvs, fm(20,50), rnor, cvfm
&      , fr(20), sr(50), div, cvnr, m, cvm, mm(20,50)
&      , lnor, lucky
integer y, ndiv, nyear, na, nr, id, yy, rec(30), i
external rnor, lnor

common /blk1/ nyear, na, nr
common /blk4/ cvnr, m, cvm
common /blk5/ div, ndiv
common /blk8/ f, s, cvf, cvs, cvfm
common /blk9/ fm, mm
common /blk10/rec

do 1000 i = 1, 777
  lucky = rnor(0)
1000 continue

  do 1010 y = 1, nyear
    fr(y) = lnor(f(y), cvf)
    if (fr(y) .lt. 0.0) fr(y) = 0.0
1010 continue

    do 1020 id = 1, ndiv
      sr(id) = lnor(s(id), cvs)
      if (sr(id) .lt. 0.0) sr(id) = 0.0
1020 continue

    write (7, *) 'Real Fishing Mortality'
    do 1040 id = 1, ndiv
      do 1030 y = 1, nyear
        fm(y, id) = lnor(fr(y)*sr(id), cvfm)
        if (fm(y, id) .lt. 0.0) fm(y, id) = 0.0
1030      continue
        write (7, 2010) (fm(y, id), y = 1, nyear)
2010      format (' ', 20(f5.3,1x))
1040      continue

    write (7, *) 'Real Natural Mortality'
    do 1140 id = 1, ndiv
      do 1130 y = 1, nyear

```

```

        mm(y, id) = lnor(m, cvm)
        if (mm(y, id) .lt. 0) mm(y, id) = 0.0
1130    continue
        write (7, 2010) (mm(y, id), y = 1, nyear)
1140 continue
        do 1210 yy = 1, nyear+na
            rec(yy) = int(real(nr) * (1.0 + rnor(0)*cvnr) + 0.5)
1210 continue
        return
    end

```

```

c=====
    subroutine grow
    real l(30,12000,10), p(3), cvp(3), erra, errb, pp(3), e
&      , rnor, ll, fm(20,50), mm(20,50), lnor
    integer igr, a, nr, yy, i, nyear, na, rec(30)
    external rnor, lnor

    common /blk1/nyear, na, nr
    common /blk2/l
    common /blk3/igr
    common /blk7/p, cvp, erra, errb
    common /blk9/fm, mm
    common /blk10/rec

    if (nr .gt. 12000) then
        write (6, *) 'Warning: nr <- 12000)'
        nr = 12000
    endif

    do 1000 yy = 1, nyear+na
        do 1010 i = 1, rec(yy)
            if (igr .eq. 1) then
                pp(1) = p(1) + cvp(1) * p(1) * rnor(0)
                pp(2) = lnor(p(2), cvp(2))
                pp(3) = 0.0
                do 1020 a = 1, na
                    ll = pp(1) * (1.0 - exp(-pp(2) * (a - pp(3))))
                    e = (erra + errb * ll) * rnor(0)
                    l(yy,i,a) = ll + e
1020                continue
            endif
            if (igr .eq. 2) write (6, *) 'modify grow.f'
            if (igr .eq. 3) write (6, *) 'modify grow.f'
1010        continue
1000    continue

2010    format (' ', i2,1x,10(F5.1,1X), ': ', F5.1, 1X, F5.3)

    return
    end

```

```

c=====
    subroutine trans
    integer itr, ntr
    common /blk11/itr, ntr

    if (itr .eq. 1) call trans1
    if (itr .eq. 2) call trans2

    return
    end

```

```

c=====
      subroutine trans1
      real  l(30,12000,10), div, tr0(0:50,0:50)
      &      , trans(50,50)
      integer itr, ntr, idc, ndiv, rec(30), nyear, na, nr
      &      , nt(0:50,0:50), ns(0:50), yy, idn, i, a

      common /blk1/ nyear, na, nr
      common /blk2/ l
      common /blk5/ div, ndiv
      common /blk11/itr, ntr
      common /blk10/rec
      common /blk22/ trans

      if (rec(1) .lt. ntr) then
        write (7, *) 'warning: ntr <- ', rec(1)
        ntr = rec(1)
      endif
      yy = 1

      do 1000 idc = 0, ndiv
        ns (idc) = 0
        do 1005 idn = 0, ndiv
          nt (idc, idn) = 0
1005      continue
1000      continue

      do 1020 i = 1, ntr
        do 1030 a = 1, na-1
          idc = int(l(yy, i, a) /div) + 1
          idn = int(l(yy, i, a+1) /div) + 1
          if (idc .gt. ndiv) idc = ndiv+1
          if (idn .gt. ndiv) idn = ndiv+1
          if (idn .lt. idc) idn = idc
          nt (idc, idn) = nt (idc, idn) + 1
          ns (idn) = ns(idn) + 1
1030      continue
1020      continue

        do 1032 i = 1, ntr
          idn = int(l(yy, i, 1) / div) + 1
          nt (0, idn) = nt (0, idn) + 1
          ns (idn) = ns(idn) + 1
1032      continue

        do 1040 idn = 0, ndiv
          if (ns(idn) .ne. 0) then
            do 1050 idc = 0, ndiv
              tr0(idn, idc) = real(nt(idc, idn)) / real(ns(idn))
1050          continue
            else
              do 1052 idc = 1, ndiv
                tr0(idn, idc) = 0.0
1052          continue
            endif
1040      continue

        write (7, *) 'Transition Matrix : itr = ', itr
        do 1060 idn = 1, ndiv
          write (7, 2010) (tr0(idn, idc), idc = 1, ndiv)
2010      format (' ', 50(F5.3,1x))
          do 1062 idc = 1, ndiv
            trans(idn, idc) = tr0(idn, idc)
1062      continue
1060      continue

      return

```

end

```
c=====
subroutine trans2

real l(30,12000,10), s(10), ss(10), avg(10), std(10), sx
& , sy, s1, s2, s3, a, b, r, x(10), y(10), bl, al, rl
& , ml, div, trans(50,50), caut(20, 50)
& , f, low, upper, pml, psd, var, ssqrt
integer itr, ntr, ia, na, rec(30), nyear, i, id, idp, nr, ndiv
& , iscaut, cd(50), k(50), kk
external f

common /blk1/ nyear, na, nr
common /blk2/ l
common /blk5/ div, ndiv
common /blk11/ itr, ntr
common /blk10/ rec
common /blk21/ caut
common /blk22/ trans

if (ntr .gt. rec(1)) then
  write (7, *) 'warning : ntr <- ', rec(1)
  ntr = rec(1)
endif

iscaut = 0
do 1000 id = 1, ndiv
  iscaut = iscaut + caut(1, id)
1000 continue

do 1001 id = 1, ndiv
  cd(id) = int(caut(1, id) * real(ntr) / real(iscaut))
1001 continue

do 1010 ia = 1, na
  s(ia) = 0.0
  ss(ia) = 0.0
  k(ia) = 0
1010 continue
do 1020 i = 1, ntr
  do 1022 ia = 1, na
    id = int(l(1, i, ia) / div)
    if (id .gt. 1) then
      cd(id) = cd(id) - 1
      if (cd(id) .ge. 0) then
        k(ia) = k(ia) + 1
        s(ia) = s(ia) + l(1, i, ia)
        ss(ia) = ss(ia) + l(1, i, ia) * l(1, i, ia)
      endif
    endif
  endif
1022 continue
1020 continue

sx = 0.0
sy = 0.0
kk = 0
do 1030 ia = 1, na
  if (k(ia) .gt. 0) then
    kk = kk + 1
    avg(ia) = s(ia) / real(k(ia))
    sx = sx + avg(ia)
    var = ss(ia) / real(k(ia)) - avg(ia) * avg(ia)
    std(ia) = ssqrt(var)
    sy = sy + std(ia)
  endif
1030 continue
```

```

        else
            avg(ia) = 0.0
            std(ia) = 0.0
        endif
1030 continue

    sx = sx / real(kk)
    sy = sy / real(kk)
    s1 = 0.0
    s2 = 0.0
    s3 = 0.0
    do 1040 ia = 1, na
        if (avg(ia) .gt. 0) then
            s1 = s1 + (avg(ia) - sx) * (std(ia) - sy)
            s2 = s2 + (avg(ia) - sx) * (avg(ia) - sx)
            s3 = s3 + (std(ia) - sy) * (std(ia) - sy)
        endif
1040 continue
        if (s2 .ne. 0) then
            b = s1 / s2
        else
            b = 0
        endif
        a = sy - b * sx
        r = s1 / sqrt(s2 * s3)

        write (7, *) 'Standard Deviation - Length regression'
        write (7, *) ' N : ', na
        write (7, *) ' a : ', a
        write (7, *) ' b : ', b
        write (7, *) ' r : ', r

        sx = 0
        sy = 0

        do 1050 ia = 1, na-1
            x(ia) = avg(ia)
            sx = sx + x(ia)
            y(ia) = avg(ia+1)
            sy = sy + y(ia)
1050 continue
        sx = sx / real(na - 1)
        sy = sy / real(na - 1)

        s1 = 0.0
        s2 = 0.0
        s3 = 0.0
        do 1060 ia = 1, na-1
            s1 = s1 + (x(ia) - sx) * (y(ia) - sy)
            s2 = s2 + (x(ia) - sx) * (x(ia) - sx)
            s3 = s3 + (y(ia) - sy) * (y(ia) - sy)
1060 continue
        bl = s1 / s2
        al = sy - bl * sx
        rl = s1 / sqrt(s2 * s3)

        write (7, *) 'Lentgh(t) - Length(t+1) regression'
        write (7, *) ' N : ', na-1
        write (7, *) ' a : ', al
        write (7, *) ' b : ', bl
        write (7, *) ' r : ', rl
        write (7, *) ' L^: ', al / (1.0 - bl)
        write (7, *) ' K^: ', -log(bl)

        do 1070 id = 1, ndiv
            ml = (id - 0.5) * div
            pml = real(al + bl * ml)
            psd = real(a + b * pml)

```

```

        do 1080 idp = 1, ndiv
            low = real(div * (idp - 1))
            upper = real(div * idp)
            trans(idp, id) = real(f(low, upper, pml, psd))
1080     continue
1070 continue

        write (7, *) 'Transition Matrix : itr = ', itr
        do 1090 id = 1, ndiv
            write (7, 2010) (trans(idp, id), idp = 1, ndiv)
2010     format (' ', 50(F5.3,1x))
1090 continue

        return
    end

```

```

c=====
    subroutine catch(ied, iedf)

        real cvnr, m, cvm, mm(20, 50)
&        , fm(20, 50), z, upper, low, ll, rnor, div
&        , l(30, 12000, 10), caut(20, 50)
        integer yy, a, id, y, nyear, na, ndiv, nr, rec(30)
&        , mort(20, 50), cautc(30, 10, 50)
&        , mortc(30, 10, 50), ccn, mmn, i, aa
&        , nn(20, 50), nc(30, 10, 50), n, ied, iedf
        external rnor

        common /blk1/ nyear, na, nr
        common /blk2/ l
        common /blk4/ cvnr, m, cvm
        common /blk5/ div, ndiv
        common /blk6/ nn, nc
        common /blk9/ fm, mm
        common /blk10/ rec
        common /blk21/ caut

        do 1225 yy = 1, na-1
            do 1230 a = na-yy, 1, -1
                z = fm(1, 1) + mm(1, 1)
                n = int(real(rec(yy)) * exp(-z*a)+0.5)
                do 1240 i = n+1, rec(yy)
                    do 1250 aa = a, na
                        l(yy, i, aa) = -3333.
1250                 continue
1240             continue
1230         continue
1225     continue

        do 1050 y = 1, nyear
            call count(y)
            do 1060 id = 1, ndiv
                z = fm(y, id) + mm(y, id)
                if (z .ne. 0) then
                    caut(y, id) = nn(y, id) * (1.0 - exp(-z))
&                    * fm(y, id) / z
                else
                    caut(y, id) = 0.0
                endif
                if (z .ne. 0) then
                    mort(y, id) = int(nn(y, id) * (1.0 - exp(-z))
&                    * mm(y, id) / z + 0.5)
                else
                    mort(y, id) = 0.0

```

```

endif
do 1070 a = 1, na
  yy = y + na - a
  if (nn(y, id) .ne. 0) then
    cautc(yy, a, id) = int(
      caut(y, id) * real(nc(yy, a, id))
      / real(nn(y, id))
      + 0.5)
    mortc(yy, a, id) = int(
      real(mort(y, id)) * real(nc(yy, a, id))
      / real(nn(y, id))
      + 0.5)
  else
    cautc(yy, a, id) = 0
    mortc(yy, a, id) = 0
  endif
1070 continue
1060 continue

do 1080 a = 1, na
  yy = y + na - a
  do 1090 id = 1, ndiv
    ccn = cautc(yy, a, id)
    mmn = mortc(yy, a, id)
    low = div * real(id - 1)
    upper = div * real(id)
    if (id .eq. ndiv) upper = low * 1000
    do 1100 i = 1, rec(yy)
      ll = l(yy, i, a)
      if (ll .ge. low .and. ll .lt. upper
        & .and. ccn .gt. 0) then
        do 1110 aa = a + 1, na
          l(yy, i, aa) = -2222
1110 continue
          ccn = ccn - 1
          goto 1100
        endif
        if (ll .ge. low .and. ll .lt. upper
          & .and. mmn .gt. 0) then
          do 1120 aa = a + 1, na
            l(yy, i, aa) = -1111
1120 continue
            mmn = mmn - 1
            goto 1100
          endif
        endif
      continue
    1090 continue
  1080 continue
1050 continue
1040 continue

write (7, *) 'Real Catch Number in size'
do 1130 id = 1, ndiv
  write (7, 2020) (int(caut(y, id)), y=1, nyear)
2020 format (' ', 20(i5,1x))
1130 continue

write (7, *) 'Real Number dead by Natural in size'
do 1140 id = 1, ndiv
  write (7, 2020) (mort(y, id), y=1, nyear)
1140 continue

write (7, *) 'Real Population in size'
do 1150 id = 1, ndiv
  write (7, 2020) (nn(y, id), y = 1, nyear)

```

```

1150 continue
      iedf = 0
      if (caut(nyear, ndiv) .eq. 0) then
        write (6, *) 'ERROR: caut(nyear, ndiv) = 0 in catch.f'
        ied = ied + 1
        iedf = 1
      endif
      return
      end

```

```

c=====
      subroutine count(y)
      real    div, l(30,12000,10), ll, low, upper
      integer ndiv, nn(20,50), nyear, na, nr, y, id, a, yy, i
      &      , nc(30,10,50), rec(30)
      common /blk1/nyear, na, nr
      common /blk2/l
      common /blk5/div, ndiv
      common /blk6/nn, nc
      common /blk10/ rec

      do 1005 id = 1, ndiv
        nn(y, id) = 0
1005 continue

      do 1007 a = 1, na
        yy = y + na - a
        do 1008 id = 1, ndiv
          nc(yy,a,id) = 0
1008 continue
1007 continue

      do 1020 a = 1, na
        yy = y + na - a
        do 1030 i = 1, rec(yy)
          do 1040 id = 1, ndiv
            low = div * real(id - 1)
            upper = div * real(id)
            if (id .eq. ndiv) upper = low * 1000.
            ll = l(yy, i, a)
            if (ll .ge. low .and. ll .lt. upper) then
              nn(y, id) = nn(y, id) + 1
              nc(yy,a,id) = nc(yy,a,id) + 1
            endif
1040 continue
1030 continue
1020 continue

      return
      end

```

```

c=====
      subroutine multi
      real div, trans(50,50), s(50), f(20), fi
      &      , xref, eps, param(50)
      &      , n(20, 50), ch(20, 50), d(20, 50)
      &      , caut(20, 50), cd(50), si(50)
      &      , init(50), step(50), tol(50), ll
      &      , alpha, stepa
      integer ndiv, nyear, i, j, iter, na, nr
      &      , iflag, ierror, iterx, ip, m

```



```

&      , ismethod
common /blk1/ nyear, na, nr
common /blk5/ div, ndiv
common /blk21/ caut
common /blk22/ trans
common /blk23/ xref, fi
common /blk25/ f, s, ch, n
common /blk31/ eps, stepa
common /blk32/ iflag, ierror, iterx
common /blk40/ d

open (18, file='multi.log')

iflag = 0
ismethod = 1
2010 format (' ', 100F6.3)
2020 format (' ', 100f6.0)

C      Initial Value
m = ndiv + nyear - 1

do 1032 ip = 1, nyear - 1
  init(ip) = fi * caut(ip, ndiv) / caut(nyear, ndiv)
1032 continue

alpha = caut(nyear, ndiv-1) / caut(nyear, ndiv)
if (caut(nyear, ndiv-1) * caut(nyear, ndiv) .eq. 0) then
  write (6, *) 'ERROR : C(xref) or C(xref-1) = 0 in mul'
  stop 100
endif
cd(ndiv) = caut(nyear, ndiv)
do 1110 j = ndiv-1, 1, -1
  cd(j) = cd(j+1) * alpha
1110 continue

do 1130 j = 1, ndiv
  si(j) = caut(nyear, j) / cd(j)
1130 continue
do 1034 ip = nyear, nyear + ndiv - 2
  if (ismethod .eq. 0) then
    init(ip) = si(ip - nyear)
  else
    init(ip) = 1.0
  endif
1034 continue
init(nyear+ndiv-1) = fi

write (18, *) 'init', (init(ip) , ip= 1, m)

do 1036 ip = 1, m
  step(ip) = stepa
1036 continue
do 1038 ip = 1, m
  tol(ip) = eps
1038 continue

c      calculation

call subsimp(m, init, step, tol, param, ll, iter, iflag)
write (6, *) '          criterion met', iter, ll

c      output result

write (18, *) 'n'
do 1210 j = 1, ndiv

```

```

        write (18, 2020) (n(i, j), i = 1, nyear)
1210 continue
        write (18, *) 'f'
        write (18, 2010) (f(i), i = 1, nyear)
        write (18, *) 's'
        write (18, 2010) (s(j), j = 1, ndiv)

        write (18, *) 'F'
        do 1220 j = 1, ndiv
            write (18, 2010) (f(i) * s(j), i = 1, nyear)
1220 continue
        write (18, *) 'd'
        do 1230 j = 1, ndiv
            write (18, 2020) (d(i,j), i = 1, nyear)
1230 continue
        return
        end

```

```

c=====
      subroutine evalu
      real fe(20), fo(20), se(50), so(50), ffe(20, 50)
&      , ffo(20, 50), ce(20, 50), co(20, 50), ne(20, 50)
&      , div, cvf, cvs, cvfm, mm(20, 50)
&      , foi(20, 200), foj(50, 200)
&      , noi(20, 200), noj(50, 200)
&      , fei(20, 200), fej(50, 200)
&      , nei(20, 200), nej(50, 200)
&      , coi(20, 200), coj(50, 200)
&      , cei(20, 200), cej(50, 200)
&      , fmse(20,50), nmse(20,50), cmse(20,50)

      integer nyear, ndiv, i, j, na, nr, no(20, 50), nc(30, 10, 50)
&      , iter, iterm, iflag, ierror, iterx

      common /blk1/ nyear, na, nr
      common /blk5/ div, ndiv
      common /blk6/ no, nc
      common /blk8/ fo, so, cvf, cvs, cvfm
      common /blk21/ co
      common /blk9/ ffo, mm
      common /blk25/ fe, se, ce, ne
      common /blk28/ iterm, iter
      common /blk29/ foi, foj, coi, coj, noi, noj
&      , fei, fej, cei, cej, nei, nej
&      , fmse, cmse, nmse
      common /blk32/ iflag, ierror, iterx

      if (iter .eq. 1) then
        do 1001 i = 1, nyear
          do 1002 j = 1, ndiv
            fmse(i, j) = 0.0
            cmse(i, j) = 0.0
            nmse(i, j) = 0.0
1002          continue
1001        continue
      endif

      if(iflag .eq. 1) then
        ierror = ierror + 1
        goto 3010
      endif

```

```

do 1000 i = 1, nyear
  do 1010 j = 1, ndiv
    ffe(i, j) = fe(i) * se(j)
1010  continue
1000  continue

  do 1020 i = 1, nyear
    do 1030 j = 1, ndiv
      fmse(i, j) = fmse(i, j) + (ffe(i, j) - ffo(i, j))**2
1030  continue
1020  continue

  write (7, *) 'ff'
  write (7, 2010) 'ffo(5,j)', (ffo(5, j), j = 1, ndiv)
  write (7, 2010) 'ffe(5,j)', (ffe(5, j), j = 1, ndiv)
  write (7, 2010) 'ffo(i,6)', (ffo(i, 6), i = 1, nyear)
  write (7, 2010) 'ffe(i,6)', (ffe(i, 6), i = 1, nyear)
2010 format (' ', a, 50(f6.4, 2x))
  do 1050 i = 1, nyear
    do 1060 j = 1, ndiv
      cmse(i, j) = cmse(i, j) + (ce(i, j) - co(i, j))**2
1060  continue
1050  continue

  write (7, *) 'catch'
  write (7, 2020) 'co(5,j)', (co(5, j), j = 1, ndiv)
  write (7, 2020) 'ce(5,j)', (ce(5, j), j = 1, ndiv)
  write (7, 2020) 'co(i,6)', (co(i, 6), i = 1, nyear)
  write (7, 2020) 'ce(i,6)', (ce(i, 6), i = 1, nyear)
2020 format (' ', a, 50(f6.0, 1x))

  do 1080 i = 1, nyear
    do 1090 j = 1, ndiv
      nmse(i, j) = nmse(i, j) + (ne(i, j) - real(no(i, j)))**2
1090  continue
1080  continue

  write (7, *) 'population'
  write (7, 2020) 'no(5,j)', (real(no(5, j)), j = 1, ndiv)
  write (7, 2020) 'ne(5,j)', (ne(5, j), j = 1, ndiv)
  write (7, 2020) 'no(i,6)', (real(no(i, 6)), i = 1, nyear)
  write (7, 2020) 'ne(i,6)', (ne(i, 6), i = 1, nyear)
3010 continue
  if (iflag .eq. 0) then
    do 1210 i = 1, nyear
      foi(i, iter) = ffo(i, 6)
      fei(i, iter) = ffe(i, 6)
      coi(i, iter) = co(i, 6)
      cei(i, iter) = ce(i, 6)
      noi(i, iter) = no(i, 6)
      nei(i, iter) = ne(i, 6)
1210  continue
      do 1220 j = 1, ndiv
        foj(j, iter) = ffo(5, j)
        fej(j, iter) = ffe(5, j)
        coj(j, iter) = co(5, j)
        cej(j, iter) = ce(5, j)
        noj(j, iter) = no(5, j)
        nej(j, iter) = ne(5, j)
1220  continue
      else
        do 1212 i = 1, nyear
          foi(i, iter) = -1

```

```

        fei(i, iter) = -1
        coi(i, iter) = -1
        cei(i, iter) = -1
        noi(i, iter) = -1
        nei(i, iter) = -1
1212    continue
        do 1222 j = 1, ndiv
            foj(j, iter) = -1
            fej(j, iter) = -1
            coj(j, iter) = -1
            cej(j, iter) = -1
            noj(j, iter) = -1
            nej(j, iter) = -1
1222    continue
        endif
        return
    end

```

```

c=====
      subroutine mse
        real foi(20,200), fei(20,200), foj(50,200), fej(50,200)
&      , coi(20,200), cei(20,200), coj(50,200), cej(50,200)
&      , noi(20,200), nei(20,200), noj(50,200), nej(50,200)
&      , div, ssqrt, rei(20, 200), rej(20, 200)
        real foix(20,200), feix(20,200), fojx(50,200), fejx(50,200)
&      , coix(20,200), ceix(20,200), cojx(50,200), cejx(50,200)
&      , noix(20,200), neix(20,200), nojx(50,200), nejx(50,200)
&      , feia(20), feja(50), ceia(20), ceja(50), neia(20), nej(50)
        real foia(20), foja(50), coia(20), coja(50), noia(20), noja(50)
&      , feisd(20), feisd(50), ceisd(20), cejsd(50)
&      , neisd(20), nejsd(50)
        real foisd(20), fojsd(50)
&      , coisd(20), cojsd(50)
&      , noisd(20), nojsd(50)
&      , fmse(20, 50), cmse(20, 50), nmse(20, 50)

        integer iterm, iter, i, j, nyear, na, nr, ndiv
&      , ierror, iflag, iterx

        external ssqrt

        common /blk1/ nyear, na, nr
        common /blk5/ div, ndiv
        common /blk28/ iterm, iter
        common /blk29/ foi, foj, coi, coj, noi, noj
&      , fei, fej, cei, cej, nei, nej
&      , fmse, cmse, nmse
        common /blk32/ iflag, ierror, iterx

        open (11, file='rltyd')
        open (12, file='rlty')
        open (13, file='rltd')

        call rep20(fei, feix, nyear, iterx)
        call rep20(cei, ceix, nyear, iterx)
        call rep20(nei, neix, nyear, iterx)
        call sd20(feix, feia, feisd, nyear, iterx)
        call sd20(ceix, ceia, ceisd, nyear, iterx)
        call sd20(neix, neia, neisd, nyear, iterx)
        call rep20(foi, foix, nyear, iterx)
        call rep20(coi, coix, nyear, iterx)

```

```

call rep20(noi, noix, nyear, iterx)
call sd20(foix, foia, foisd, nyear, iterx)
call sd20(coix, coia, coisd, nyear, iterx)
call sd20(noix, noia, noisd, nyear, iterx)

call rep50(fej, feyx, ndiv, iterx)
call rep50(cej, cejx, ndiv, iterx)
call rep50(nej, nejx, ndiv, iterx)
call sd50(feyx, feja, fejsd, ndiv, iterx)
call sd50(cejx, ceja, cejsd, ndiv, iterx)
call sd50(nejx, neja, nejsd, ndiv, iterx)
call rep50(foj, fojx, ndiv, iterx)
call rep50(coj, cojx, ndiv, iterx)
call rep50(noj, nojx, ndiv, iterx)
call sd50(fojx, foja, fojsd, ndiv, iterx)
call sd50(cojx, coja, cojsd, ndiv, iterx)
call sd50(nojx, noja, nojsd, ndiv, iterx)

do 1044 i = 1, nyear
  do 1046 j = 1, ndiv
    fmse(i, j) = ssqrt(fmse(i, j) / real(iterm))
    cmse(i, j) = ssqrt(cmse(i, j) / real(iterm))
    nmse(i, j) = ssqrt(nmse(i, j) / real(iterm))
1046   continue
1044  continue
  do 1052 iter = 1, iterx
    do 1054 i = 1, nyear
      if (noi(i, iter) .ne. 0) then
        rei(i, iter) = (nei(i, iter) - noi(i, iter))/noi(i, iter)
      else
        rei(i, iter) = 0
      endif
1054   continue
1052  continue

    do 1086 iter = 1, iterx
      write (11, 2070) rei(5, iter)
1086  continue

    do 1087 iter = 1, iterx
      write (12, 2070) (rei(i, iter), i = 1, nyear)
1087  continue

    do 1056 iter = 1, iterx
      do 1058 j = 1, ndiv
        if (noj(j, iter) .ne. 0) then
          rej(j, iter) = (nej(j, iter) - noj(j, iter))/noj(j, iter)
        else
          rej(j, iter) = 0.
        endif
      continue
1058   continue
1056  continue
    do 1088 iter = 1, iterx
      write (13, 2070) (rej(j, iter), j = 1, ndiv)
1088  continue

    write (10, 2050) ' RESULT OF ESTIMATION '
    write (10, *)
    write (10, *) 'F(i,6) observed'
    write (10, 2020)'mean', (foia(i), i=1, nyear)
    write (10, 2020)'s.d.', (foisd(i), i=1, nyear)
    write (10, *) 'F(i,6) estimated'
    write (10, 2020)'mean', (feia(i), i=1, nyear)
    write (10, 2020)'s.d.', (feisd(i), i=1, nyear)

```

```

write (10, *)
write (10, *) 'C(i,6) observed'
write (10, 2040)'mean', (int(coia(i)), i=1, nyear)
write (10, 2040)'s.d.', (int(coisd(i)), i=1, nyear)
write (10, *) 'C(i,6) estimated'
write (10, 2040)'mean', (int(ceia(i)), i=1, nyear)
write (10, 2040)'s.d.', (int(ceisd(i)), i=1, nyear)
write (10, *)
write (10, *) 'N(i,6) observed'
write (10, 2040)'mean', (int(noia(i)), i=1, nyear)
write (10, 2040)'s.d.', (int(noisd(i)), i=1, nyear)
write (10, *) 'N(i,6) estimated'
write (10, 2040)'mean', (int(neia(i)), i=1, nyear)
write (10, 2040)'s.d.', (int(neisd(i)), i=1, nyear)
write (10, *)
write (10, *) 'F(5,j) observed'
write (10, 2020)'mean', (foja(j), j=1, ndiv)
write (10, 2020)'s.d.', (fojsd(j), j=1, ndiv)
write (10, *) 'F(5,j) estimated'
write (10, 2020)'mean', (feja(j), j=1, ndiv)
write (10, 2020)'s.d.', (fejsd(j), j=1, ndiv)
write (10, *)
write (10, *) 'C(5,j) observed'
write (10, 2040)'mean', (int(coja(j)), j=1, ndiv)
write (10, 2040)'s.d.', (int(cojsd(j)), j=1, ndiv)
write (10, *) 'C(5,j) estimated'
write (10, 2040)'mean', (int(ceja(j)), j=1, ndiv)
write (10, 2040)'s.d.', (int(cejisd(j)), j=1, ndiv)
write (10, *)
write (10, *) 'N(5,j) observed'
write (10, 2040)'mean', (int(noja(j)), j=1, ndiv)
write (10, 2040)'s.d.', (int(nojsd(j)), j=1, ndiv)
write (10, *) 'N(5,j) estimated'
write (10, 2040)'mean', (int(neja(j)), j=1, ndiv)
write (10, 2040)'s.d.', (int(nejsd(j)), j=1, ndiv)

write (10, *)
write (10, *) char(12)

write (10, *) 'RMSE'
write (10, *) ' F'
do 1100 j = 1, ndiv
  write (10, 2060) (fmse(i, j), i = 1, nyear)
1100 continue
write (10, *)
write (10, *) ' C'
do 1110 j = 1, ndiv
  write (10, 2060) (cmse(i, j), i = 1, nyear)
1110 continue
write (10, *)
write (10, *) ' N'
do 1120 j = 1, ndiv
  write (10, 2060) (nmse(i, j), i = 1, nyear)
1120 continue

2010 format (' ', i4, 2x, 50(f6.4, 2x))
2020 format (' ', A4, 2x, 50(f6.4, 2x))
2030 format (' ', i4, 2x, 20(i6, 2x))
2040 format (' ', A4, 2x, 20(i6, 2x))
2050 format (' ', 25('*'), A, 25('*'))
2060 format (' ', 50(1PE9.2, 2x))

```

```
2070 format (' ', 50(f6.3, 1x))
```

```
close (11)  
close (12)  
close (13)
```

```
return  
end
```

```
C=====
```

```
subroutine sd20(x, avg, sd, imax, jmax)
```

```
real x(20, 200), sum, sum2, var, sd(20), avg(20), ssqrt  
integer i, j, imax, jmax  
external ssqrt
```

```
do 1010 i = 1, imax
```

```
sum = 0.0
```

```
sum2 = 0.0
```

```
do 1020 j = 1, jmax
```

```
sum = sum + x(i, j)
```

```
sum2 = sum2 + x(i, j) * x(i, j)
```

```
1020 continue
```

```
avg(i) = sum / real(jmax)
```

```
var = sum2 / real(jmax) - avg(i) * avg(i)
```

```
sd(i) = ssqrt(var)
```

```
1010 continue
```

```
return
```

```
end
```

```
C=====
```

```
subroutine sd50(x, avg, sd, imax, jmax)
```

```
real x(50, 200), sum, sum2, var, sd(50), avg(50), ssqrt  
integer i, j, imax, jmax  
external ssqrt
```

```
do 1010 i = 1, imax
```

```
sum = 0.0
```

```
sum2 = 0.0
```

```
do 1020 j = 1, jmax
```

```
sum = sum + x(i, j)
```

```
sum2 = sum2 + x(i, j) * x(i, j)
```

```
1020 continue
```

```
avg(i) = sum / real(jmax)
```

```
var = sum2 / real(jmax) - avg(i) * avg(i)
```

```
sd(i) = ssqrt(var)
```

```
1010 continue
```

```
return
```

```
end
```

```
C=====
```

```
subroutine rep20 (x, y, imax, jmax)
```

```
real x(20, 200), y(20, 200)  
integer i, j, imax, jmax, jc
```

```
do 1010 i = 1, imax
```

```
jc = 1
```

```
do 1020 j = 1, jmax
```

```
if (x(i, j) .gt. 0) then
```

```
y(i, jc) = x(i, j)
```

```
jc = jc + 1
```

```
endif
```

```
1020 continue
```

```
1010 continue
```

```
return
```

```

      end
C=====
      subroutine rep50 (x, y, imax, jmax)
      real x(50, 200), y(50, 200)
      integer i, j, imax, jmax, jc
      do 1010 i = 1, imax
        jc = 1
        do 1020 j = 1, jmax
          if (x(i, j) .gt. 0) then
            y(i, jc) = x(i, j)
            jc = jc + 1
          endif
        1020 continue
      1010 continue

      return
      end
C=====
      real function ssqrt(x)
      real x
      if (x .gt. 0.0) then
        ssqrt = sqrt(x)
      else
        ssqrt = 0.0
      endif

      return
      end
C=====
      double precision function func(param, m)
      real div, trans(50,50), s(50), f(20), z(100,100), fi
      &      , xref, eps, ff(100, 100)
      &      , n(20, 50), sum1, sum2, ch(20, 50), d(20, 50), nd(50)
      &      , cvnr, mort, cvm, caut(20, 50)
      &      , step, fl, fu, sl, su
      real*8 param(50)
      integer ndiv, nyear, i, j, k, na, nr
      &      , iflag, ierror, iterx, ip, m

      common /blk1/ nyear, na, nr
      common /blk4/ cvnr, mort, cvm
      common /blk5/ div, ndiv
      common /blk21/ caut
      common /blk22/ trans
      common /blk23/ xref, fi
      common /blk25/ f, s, ch, n
      common /blk31/ eps, step
      common /blk32/ iflag, ierror, iterx
      common /blk40/ d

      fl = 0.1
      fu = 0.4
      sl = 0.5
      su = 1.5

      do 1000 ip = 1, nyear - 1
        if (param(ip) .lt. fl) param(ip) = fl
        if (param(ip) .gt. fu) param(ip) = fu
        f(ip) = param(ip)
      1000 continue
      do 1010 ip = nyear, nyear + ndiv - 2
        if (param(ip) .lt. sl) param(ip) = sl
        if (param(ip) .gt. su) param(ip) = su

```



```

      s(ip - nyear + 1) = param(ip)
1010 continue
      s(ndiv) = 1.0
      f(nyear) = param(nyear + ndiv - 1)

      do 1040 i = 1, nyear
        do 1050 j = 1, ndiv
          ff(i, j) = f(i) * s(j)
          if (caut(i, j) .eq. 0) ff(i, j) = 0.0
          z(i, j) = ff(i, j) + mort
1050   continue
1040 continue

        do 1052 j = 1, ndiv
          if (caut(nyear, j) .ne. 0) then
            n(nyear, j) = caut(nyear, j) * z(nyear, j)
            & / (1.0 - exp(-z(nyear, j))) / ff(nyear, j)
          else
            n(nyear, j) = 0
          endif
1052 continue
        do 1054 i = 1, nyear
          if (caut(i, ndiv) .ne. 0) then
            n(i, ndiv) = caut(i, ndiv) * z(i, ndiv)
            & / ff(i, ndiv) / (1.0 - exp(-z(i, ndiv)))
          else
            n(i, ndiv) = 0.0
          endif
1054 continue
        do 1060 i = nyear-1, 1, -1
          sum1 = 0.0
          do 1062 j = 1, ndiv
            sum1 = sum1 + n(i+1, j)
1062   continue

          do 1070 j = 1, ndiv-1
            nd(j) = 0.0
            do 1080 k = 1, ndiv
              nd(j) = nd(j) + trans(k, j) * n(i+1, k)
1080   continue
1070   continue
            nd(ndiv) = n(i, ndiv)

            sum2 = 0.0
            do 1072 j = 1, ndiv - 1
              sum2 = sum2 + nd(j)
1072   continue
            do 1074 j = 1, ndiv - 1
              if (caut(i, j) .ne. 0) then
                n(i, j) = exp(z(i, j)) * nd(j)
              else
                n(i, j) = 0.0
              endif
1074   continue
1060 continue

          do 1090 i = 1, nyear
            do 1100 j = 1, ndiv
              ch(i, j) = n(i, j) * (1.0 - exp(-z(i, j)))
              & * ff(i, j) / z(i, j)
1100   continue
1090 continue

          func = 0.0
          do 1112 i = 1, nyear
            do 1122 j = 1, ndiv
              d(i, j) = caut(i, j) - ch(i, j)

```

```

                func = func + d(i, j) * d(i, j)
1122         continue
1112         continue

        return
        end

        real function lnor(a, c)
        real mu, sig, rnor, ssqrt, a, s, c

        external ssqrt, rnor

        s = a * c
        if (s .eq. 0.0) then
            lnor = a
        else
            sig=ssqrt(log((s / a)**2 + 1))
            mu = log(a / sqrt((s / a)**2 + 1))
            lnor = exp(mu + sig * rnor(0))
        endif

        return
        end

```

```

=====
        real function f(x1, x2, m, s)
        real x1, x2, f1, f2, m, s, z1, z2, r2,
C         rp, zd, erf

        r2 = sqrt(2.0)
        rp = 2.0

        z1 = (x1 - m) / r2 / s
        z2 = (x2 - m) / r2 / s
        if (z1 .gt. z2) then
            zd = z2
            z2 = z1
            z1 = zd
        endif

        f1 = erf(abs(z1))/rp
        f2 = erf(abs(z2))/rp

        if(z1 .ge. 0 .and. z2 .ge. 0) f = f2 - f1
        if(z1 .lt. 0 .and. z2 .ge. 0) f = f1 + f2
        if(z1 .lt. 0 .and. z2 .lt. 0) f = f1 - f2
2010 format (' ', 7(E10.5, 1X))

        return
        end

```

```

=====
        subroutine subsimp
        &         (m, initial, stepsize, tolerance,
        &         optparam, optfunc, iteration, iflag)

c Translated by T. Matsuishi(Univ. Hokkaido) from BASIC program in
c Paso-kon niyoru deta akiseki nyuumon by Okumura (1986)
c
c maximum number of parameters    50
c
c make real function func(parameter, m)
c     argument parameter(50) : value of parameters
c         m                  : number of parameters
c         func                : value of object function
c

```

```

    real initial(50), stepsize(50), tolerance(50), optparam(50)
    & , optfunc
    real*8 , vertex(51,50)
    & , criterion(51)
    & , s, v1, v2, f, v, vmax, vmin, vsum, p
    & , parameter(50)
    double precision func
    integer m, j, kworst, kbest, iteration, flag, k, itermax
    & , idebug, iflag
    external func

    itermax = 50000
    idebug = 0
    iflag = 0

c   initial point
    s = sqrt(0.5)
    v1 = s * (sqrt(m+1.0) - 1.0) / dble(m)
    do 1100 j = 1, m
        vertex(1, j) = initial(j)
1100 continue
    do 1130 j = 1, m
        v2 = vertex(1, j) + v1 * stepsize(j)
        do 1150 k = 2, m+1
            vertex(k, j) = v2
1150     continue
        vertex(j+1, j) = v2 + s * stepsize(j)
1130 continue

c   calculate objective function
    do 1200 k = 1, m+1
        do 1210 j = 1, m
            parameter(j) = vertex(k, j)
1210     continue
        f = func(parameter, m)
        criterion(k) = f
1200 continue

    do 1280 iteration = 1, itermax
        kworst = 1
        kbest = 1
        do 1310 k = 2, m+1
            if (criterion(k) .gt. criterion(kworst)) kworst = k
            if (criterion(k) .lt. criterion(kbest)) kbest = k
1310     continue
        if (idebug .ne. 0) then
            write (7, *)
            write (7, *) 'iteration: ', iteration
            write (7, *) 'criterion: ', criterion(kbest)
            write (7, *) 'parameters: '
            do 1390 j = 1, m
                write (7, *) j, vertex(kbest, j)
1390         continue
        endif

        flag = 0
        do 1430 j = 1, m
            vmax = vertex(1, j)
            vmin = vmax
            do 1450 k = 2, m+1
                v = vertex(k, j)
                if (v .gt. vmax) vmax = v
                if (v .lt. vmin) vmin = v
1450         continue
            if (vmax - vmin .gt. tolerance(j)) flag = 1
1430     continue

```

```

        if(flag .eq. 0) goto 1930
        do 1530 j = 1, m
            vsum = 0
            do 1550 k = 1, m+1
                if (k .ne. kworst) vsum = vsum + vertex(k, j)
1550            continue
                parameter(j) = 2.0 * vsum / m - vertex(kworst, j)
1530        continue
        f = func(parameter, m)
        if (f .lt. criterion(kbest)) then
            do 1630 j = 1, m
                p = 1.5 * parameter(j) - 0.5 * vertex(kworst, j)
                vertex(kworst, j) = parameter(j)
                parameter(j) = p
1630            continue
            criterion(kworst) = f
            f = func(parameter, m)
            if (f .lt. criterion(kworst))
                &         call replace(vertex, parameter, criterion, f, m, kworst)
        else if (f .lt. criterion(kworst)) then
            call replace (vertex, parameter, criterion, f, m, kworst)
        else
            do 1760 j = 1, m
                parameter(j) = 0.75 * vertex(kworst, j)
                &         + 0.25 * parameter(j)
1760            continue
            f = func(parameter, m)
            if (f .lt. criterion(kworst)) then
                call replace (vertex, parameter, criterion, f, m, kworst)
                goto 1910
            endif
            do 1810 k = 1, m+1
                if (k .ne. kbest) then
                    do 1830 j = 1, m
                        parameter(j) = 0.5
                        &         * (vertex(k, j) + vertex(kbest, j))
                        vertex(k, j) = parameter(j)
1830                    continue
                    f = func(parameter, m)
                    criterion(k) = f
                endif
            continue
1810        endif
1910        continue
1280    continue
c      error
      iflag = 1
c
c      exit
1930    continue
        do 1932 j = 1, m
            optparam(j) = vertex(kbest, j)
1932    continue
        optfunc = criterion(kbest)

        return
        end

c=====
      subroutine replace(vertex, parameter, criterion, f, m, kworst)
        real*8 vertex(51, 50), parameter(50), criterion(51), f
        integer j, m, kworst
        do 2030 j = 1, m

```

```
        vertex(kworst, j) = parameter(j)
2030 continue
    criterion(kworst) = f
    return
end
```

c=====