

# 最尤推定システム MLE-SYS の研究開発

九州東海大学 工学部 道 家 暎 幸  
創価大学 工学部 浅 野 長 一 郎

(1989 年 6 月 受付)

## 1. 緒 言

現象構造を示す数学モデルでの未知母数を推定する際、最尤推定法 (MLE) は一般的な方法として頻繁に適用される。最尤法による母数推定は、単純な構造モデルのとき最尤解が陽に得られるので問題はない。しかし、実際の現象はそれほど単純でなく、母数推定は適切な初期値を与え反復収束法によって数値解を求めることになる。例えば、生体機能の現象研究では、その構造モデルが複雑で、最尤解を陽に得ることは稀である。最尤解は一致性や漸近有効性等の好ましい性質をもつが、一般に非線形連立尤度方程式の場合、反復収束法による計算は局所解や不適解が出現することも少なくない。したがって、最尤法の利用には数値解法の選択と初期値の設定が極めて重要になる。この種の算法プログラムの作成は複雑かつ労力的であり、信頼性・高速性と融通性をもって広範に利用できるソフトウェアが望まれる。

本システムは、所与のモデルの未知母数の最尤推定をパソコン上で有効に行うために、諸種の独創的機能を与えている。これまでに、非線形問題の解法システムとして NON-LIN (Upjohn Co., Kalamazoo, Michigan, U.S.A.)・SALS (Oyanagi and Nakagawa (1981), 中川・小柳 (1987)) 等があり、また単に最尤推定用サブルーチンを集めた MAXLIK (Kaplan and Elston (1972)) がある。しかし、これらは元来大型計算機の利用を前提とし、データマネージメント・操作・会話・システム誘導・初期値設定・解法の中途変更など制御の利用の機能面に不便がある。

本 MLE-SYS システム<sup>(註)</sup>は、パソコン利用を前提に、統計学や数値解法の知識を有しない利用者にも最尤解を有効に得られるように開発された。このシステムは、数値解法を選択とその中途変更・初期値の設定と初期値の変更等に種々の独創的な制御的機能を持ち、日常の利用を目的としたものである。

## 2. 基本的設計思想と特徴

統計的モデルのあてはめでは最尤法が多く適用される。しかし、最尤推定システムの作成には、アルゴリズムの複雑さと、プログラミングの労力が軽視できない。本システムは、これらを軽減する汎用システムとして開発され、非線形最適化問題での求解の知識と経験が加味されている。したがって、本システムの設計では、次の事柄を重要視した。

---

(註) 本システムは、統計数理研究所の 2 年間にわたる「生物医学データに関する推測と解析アルゴリズムの研究」(統計数理研究所共同研究(61-共研-54, 62-共研-70))に関連し、パソコン(PC-9801, J-3100, PC/AT)の上で実行される汎用システムとして研究開発された (Douke and Asano (1988))。

### (1) データとの直結性

観測データは一定書式のケースデータとし、フロッピー・ディスクの利用を前提にした。利用者がデータファイル名を入力するだけでデータは自動的に読み込まれる。

### (2) 尤度関数の簡便入力

尤度関数または対数尤度関数をキーボードから直接入力する。複雑な尤度関数は部分的置換方式でも入力できる。また、導関数を必要とするときは、キーボードからの直接入力、または数値微分の利用も可能である。

### (3) 利用操作の簡易化

コンピュータ知識に乏しい利用者をも想定し、コンパイルやリンク等の操作はコンピュータによる自動化とし、また会話的に利用するため、メニューによる指示に应答し、処理の簡易化を実現する。

### (4) 数値解法の充実

登録した数値解法は、各々固有の特徴と有用性をもっている。しかし、必要に応じて毎回プログラミングするのは面倒である。これから、各種の高品質な数値解法のプログラムを豊富に準備しておく必要がある。

### (5) 解法の補完的複合運用

最適解法について、全ての目的関数と制約条件を通じて全般的に優位な解法は存在しない。事実、所与の尤度関数に対し、適切な解法を一意的に選択することは容易でない。すなわち特定の一解法だけ利用するのではなく、最適解探索のある段階で他の解法に変更できる解法の段階的利用によって効率化をはかる必要がある。

### (6) 収束問題への対処

当該モデルに選択した解法を実行しても、解が収束しなかったり、局所解で停止する場合があります。これらは、初期値（初期ベクトル）の不適切さ、多数母数の同時推定、母数に対する制約条件の存在、尤度関数（曲面）の複雑さ、一意解の存在性や漸近有効性などに起因し、また収束判定条件や最大反復回数なども原因となると考えられる。このような際に、適切な制御的処理が必要となる。

### (7) 新研究成果や解法の簡易追加性

統計解析・数値解法や情報処理について、新しい研究成果を容易に追加できる必要がある。つまり昨今までの成果を日常に蓄積・改良・削除し、知識の集積をシステムに反映させ、一層の効率的な利用を可能にして将来への展開に備える必要がある。

## 3. ソフトウェア操作面での機能設計

基本的設計思想を実現するため、本システムで開発された諸機能について説明する。

### (1) 入出力機能

解法の選択・母数に関する制約条件・収束判定基準の設定には、それぞれメニュー表示によって入力が指示される。また入力に内容的説明を要する際には随時コメントを付し、入力事項の理解を容易にする。

### (2) 初期値と解法の中途変更機能

ある一つの解法を選択し計算を実行しても最適解に到達しない場合、母数の制約条件を満たさず計算が停止する場合、また同じ解法で初期値や停止基準を変えて計算を希望する場合、さらに他の解法に適用を切り換える場合など、計算条件の変更後も継続して最適化計算を実行す

る機能を有する。

(3) 制約条件に対するチェック機能  
種々の現象モデルでは、母数に値域的制約があったり母数間に制約式が存在する場合も少なくない。このような制約条件を取り込んで反復計算ごとにチェックして、推定の妥当性を確認する機能をもつ。

(4) 等高線マップ表示機能

最尤推定を行う際、尤度関数曲面の様相を知ることは非線形最適化問題の求解に不可欠である。反復計算が最大反復回数や不適解等のため停止した際、直ちに曲面の複雑な多峰性を想定するが、実は曲面自体に比較的単調に斜行する峡谷をもつことがある。尤度関数が多くの母数を有する場合や関数形が複雑な場合は、直接に高次元尤度関数曲面に関する考察ができない。本システムでは指定された2母数に関し2次元母数空間での等高線マップで尤度関数曲面の形状を把握する。この等高線マップは初期値を探索的に設定したり、計算過程での探索の妥当性を確認する際、診断的に利用される。とくに、初期値の設定は非線形最適化問題を解く上で重要である。そこで、初めに各母数に関して制約条件内の比較的広い範囲を指定し等高線マップの図示により最高値を視察し、その近隣の部分的拡大をズームアップ機能で確認し初期値を決定し得る。また、計算過程でも必要に応じ、等高線マップを表示し、尤度関数曲面の様相を把握する。マップは、前回の関数値と今回の関数値および現環境での最高値(表示#)の位置が示され、探索方向や適用解法の妥当性も検討できる。また、収束値が局所最大値か大域的 maximum 値かの様相も視覚的に把握できる。また、モニターTVの1画面には、6枚までのマップの同時表示が可能で、各母数の相互の関係を曲面形状で超立体的に洞察で

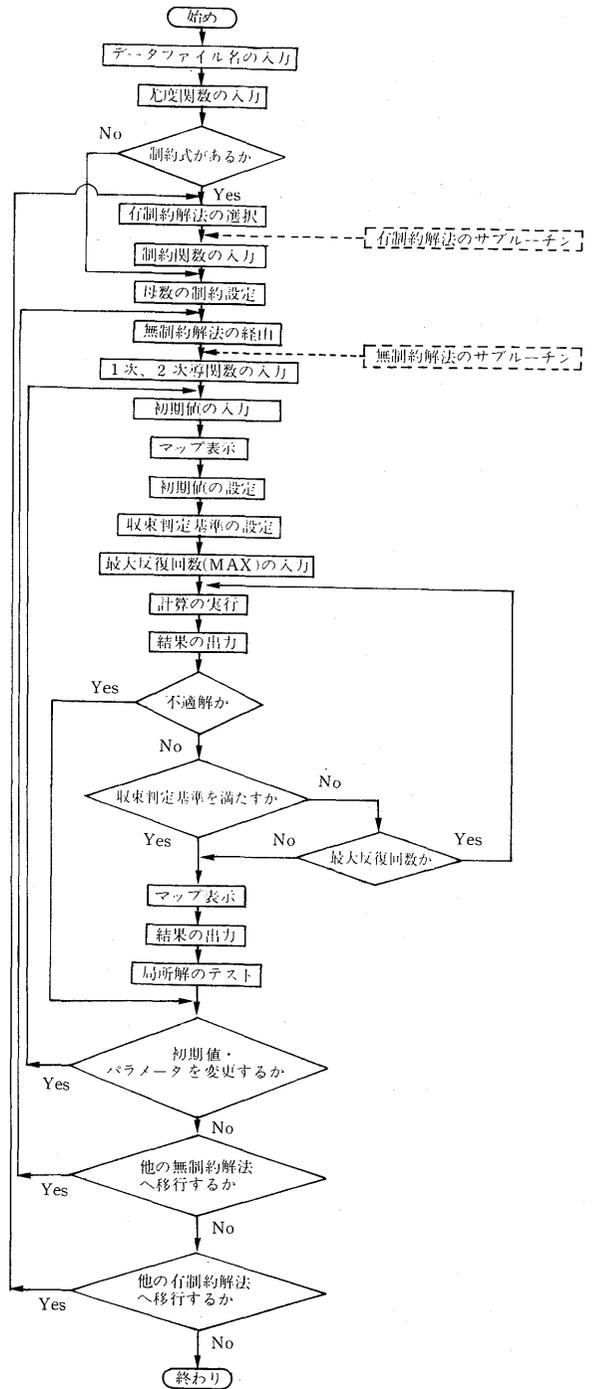


Fig. 1. System flowchart.

き、同時に反復探索の方向の変化も把握できる。

#### 4. システム構成

本システムでは、データファイル名と尤度関数を入力し、必要に応じ制約関数を入力するだけで、自動的にコンパイルされ、オブジェクトファイルが作成される。次いで、予め準備されている最適化計算の実行オブジェクトファイルと自動的にリンクし、実行ファイルが構成される。このシステムの流れを Fig. 1 に示している。採択する解法によって必要な1次や2次の導関数は、キーボードから直接に入力できるが、数値微分による自動計算も可能で、選択指定すればよい。なお、本システムの詳細な処理手順は Appendix に示している。

#### 5. 数値解法

非線形数値解法の選択は、初期値の設定とともに重要な問題である。最適化解法は、尤度関数の形状・微分可能性・制約条件の有無などにより分類される。本システムでは、多くの有用な最適化解法を収録している。それを利用する際、解法とその要約的特徴を表示したメニューから適当と思われる解法を採択し、また計算過程でも解法を制御的に変更して採用することも可能である。現段階では、17種の無制約最適化数値解法と5種の有制約最適化数値解法が要約的特徴とともに収録してあり、解法メニュー中のコード番号を指示するだけで適用される。Table 1 は、現時点における非線形最適化解法を一覧表で示している。表中で無制約最適化解法での○印は1階微係数または2階微係数を必要とする解法を示し、有制約最適化解法での○

Table 1. 非線形最適化解法.

無制約最適化解法							
No.	Method	Derivative		No.	Method	Derivative	
		1st	2nd			1st	2nd
A-1	Random			A-9	McCormick	○	
A-2	Steepest descent	○		A-10	Powell		
A-3	Newton-Raphson	○	○	A-11	Fletcher-Reeves	○	
A-4	N-R with linear search	○	○	A-12	Hestenes-Stiefel-Daniel	○	○
A-5	Levenberg-Marquardt	○	○	A-13	Sorenson-Wolfe	○	
A-6	Fletcher-Powell	○		A-14	Polak-Ribiere-Polyak	○	
A-7	Broyden-Fletcher-Shanno	○		A-15	Cyclic Coordinate		
A-8	Pearson	○		A-16	Hooke-Jeeves		
				A-17	Rosenbrock		

#### 有制約最適化解法

No.	Method	Inequality	Equality
B-1	Exterior point	○	○
B-2	Interior point	○	
B-3	Multiplier	○	○
B-4	Projection	○	
B-5	Sequential quadratic programming	○	○

印は不等式制約付きまたは等式制約付きの問題に適用できる解法を示している。

### 5.1 ランダム法 (Table 1 の A-1)

無制約数値解法のうちで事前情報を全く使わない直接的な探索方法である。実際、目的関数の性質に依存せず大域的に解を探す方法であるが、この解法は母数が多いと必要な計算回数が増加する欠点があり、この方法を適用することは稀である。しかし、どの成書にも記載されている基礎的方法として本システムにも含めている。

### 5.2 最急降下法 (Table 1 の A-2)

無制約数値解法の中で周知な降下法は、目的関数  $f(\theta)$  を最小にする  $n$  次元母数ベクトル  $\theta \in R^n$  を求める典型的なアルゴリズムである。この最急降下法は、点  $\theta^k$  における方向ベクトル  $d^k$  を

$$d^k = -g(\theta^k), \quad g(\theta^k) = \partial f(\theta) / \partial \theta \big|_{\theta = \theta^k}$$

とする。最急降下法は関数曲面に対して 1 次近似で降下方向を決めるので、勾配が大きい曲面で良い方向を与えるが、2 次以上の項が無視できなければ収束が悪くなる。この解法は、一般の教科書で周知であるが、効率的でないことが知られている。

### 5.3 Newton-Raphson 法

この方法は、一般の非線形関数が最適点の近傍において 2 次関数で近似できるとして、広範に日常的に適用される。いま、 $f(\theta)$  を  $\theta^k$  の近傍で 2 次近似すると

$$f(x) \doteq f(\theta^k) + g(\theta^k)^t (\theta - \theta^k) + (\theta - \theta^k)^t H(\theta^k) (\theta - \theta^k) / 2$$

である。ここに

$$g(\theta^k) = \partial f(\theta) / \partial \theta \big|_{\theta = \theta^k}$$

$$H(\theta^k) = \partial^2 f(\theta) / \partial \theta \partial \theta^t \big|_{\theta = \theta^k}$$

である。このとき  $H(\theta^k)$  を正定値とし、上式の右辺を  $\theta$  で微分しゼロとおくとき、Newton-Raphson 法 (Table 1 の A-3) の更新公式

$$\theta^{k+1} = \theta^k - H(\theta^k)^{-1} g(\theta^k)$$

が得られる。さらに、方向ベクトル  $d = -H(\theta^k)^{-1} g(\theta^k)$  を用いて 1 次元探索を行うアルゴリズム (Table 1 の A-4) が多く利用される。これらの解法でヘシアンヘシアンの正定値性がしばしば問題となる。

### 5.4 修正 Newton 法

Newton-Raphson 法は 2 階微係数を利用する効率的な方法であるが、ヘシアンが解析的な形で得られない場合やヘシアンが正定値でない場合には適用できない。Levenberg-Marquardt 法 (Table 1 の A-5) は、上記のヘシアンに対角行列を付加して正定値性を保たせる方法である。

### 5.5 準 Newton 法

準 Newton 法は、ヘシアンヘシアンの逆行列に近似する正定値対称行列によって降下方向を逐次生成し、反復の更新とともにこの近似を逐次修正していく。次に、その代表的なものを示す。

(i) Fletcher-Powell 法 (Table 1 の A-6)

$$H^{k+1} = H^k + \delta^k (\delta^k)^t / (\delta^k)^t \gamma^k - H^k \gamma^k (\gamma^k)^t H^k / (\gamma^k)^t H^k \gamma^k$$

(ii) Broyden-Fletcher-Shanno 法 (Table 1 の A-7)

$$H^{k+1} = [I - \delta^k (\gamma^k)^t / (\delta^k)^t \gamma^k] H^k [I - \gamma^k (\delta^k)^t / (\delta^k)^t \gamma^k] + \delta^k (\delta^k)^t / (\delta^k)^t \gamma^k$$

(iii) Pearson 法 (Table 1 の A-8)

$$H^{k+1} = H^k + (\delta^k - H^k \gamma^k) (\gamma^k)^t H^k / (\gamma^k)^t H^k \gamma^k$$

(iv) McCormick 法 (Table 1 の A-9)

$$H^{k+1} = H^k + (\delta^k - H^k \gamma^k) (\delta^k)^t / (\delta^k)^t \gamma^k$$

ここに  $\delta^k = \theta^{k+1} - \theta^k$ ,  $\gamma^k = g(\theta^{k+1}) - g(\theta^k)$  としている。

## 5.6 共役方向法

共役方向法は2次関数の係数行列に関し互いに共役な方向へ直接探索を行えば、高々  $n$  回の反復で  $n$  次元空間の最小点に到達するという性質に基づく。とくに Powell 法 (Table 1 の A-10) は尤度関数の微係数を使用しない方法の中で最も有効な方法の一つである。次に Powell 法のアルゴリズムを示す。

段階 1: 1 次独立なベクトル  $d_0^0, \dots, d_{n-1}^0 \in R^n$ , 出発点  $\theta_0^0 \in R^n$  を与え  $k=0$  とする。

段階 2:  $\theta_0^k = \theta_0^k$  とおいてこれらのベクトルの方向に沿って 1 次元探索を  $n$  回行う。

$$\theta_{i+1}^k = \min_{a_i} f(\theta_i^k + a_i d_i^k), \quad i=0, 1, \dots, n-1$$

段階 3:  $d_n^k = \theta_n^k - \theta_0^k$  として,  $\theta_{n+1}^k = \min_{a_n} f(\theta_n^k + a_n d_n^k)$ 。

段階 4:  $\theta_0^{k+1} = \theta_{n+1}^k$ 。

段階 5:  $\|\theta_0^{k+1} - \theta_0^k\| \leq \varepsilon$  ならば停止, 否なら段階 6 へ。

段階 6:  $d_i^{k+1} = d_{i+1}^k$ ,  $i=0, 1, \dots, n-2$ ,  $d_{n+1}^{k+1} = \theta_n^k - \theta_0^k$ ,  $k=k+1$  として段階 2 へ行く。

また、微係数を利用して共役方向を生成する共役勾配法は、次の関係式から方向ベクトルを与える。

$$d^k = -g(\theta^k)^t + \beta^k d^{k-1}, \quad \beta^k \in R^1$$

ここに、 $\beta^k$  の公式として次のものがある。

(i) Fletcher-Reeves の公式 (Table 1 の A-11)

$$\beta^k = \|g(\theta^k)\|^2 / \|g(\theta^{k-1})\|^2$$

(ii) Hestenes-Stiefel-Daniel の公式 (Table 1 の A-12)

$$\beta^k = g(\theta^k) H(\theta^k) d^{k-1} / (d^{k-1})^t H(\theta^k) d^{k-1}$$

(iii) Sorenson-Wolfe の公式 (Table 1 の A-13)

$$\beta^k = g(\theta^k) \gamma^{k-1} / (d^{k-1})^t \gamma^{k-1}$$

(iv) Polak-Ribiere-Polyak の公式 (Table 1 の A-14)

$$\beta^k = g(\theta^k) \gamma^{k-1} / \|g(\theta^{k-1})\|^2$$

ここで  $\gamma^k = g(\theta^{k+1}) - g(\theta^k)$  としている。

## 5.7 直接探索法

直接探索法は導関数が存在しないとか関数の形が非常に複雑な場合に利用されるが、導関数を利用する解法より効率が劣ると言われている。本システムでは次の解法を登録している。

(i) 座標方向逐次探索法 (Table 1 の A-15)

この方法は座標軸方向の直線探索を繰り返す方法であり、目的関数  $f(\theta)$  が 2 次関数で、等高線が球面であるか主軸が座標軸に平行な楕円面をなしている場合、このアルゴリズムは有効である。次に、そのアルゴリズムを示す。

段階 1:  $d_i = e_i, i = 1, 2, \dots, n, \theta_0^i \in R^n, \varepsilon > 0$  を与え  $k=0$  とする。ここに  $e_i$  は軸方向単位ベクトルである。

段階 2:  $\theta_i^k = \theta_0^k$  として次の 1 次元探索を行う。

$$\theta_{i+1}^k = \min_{a_i} f(\theta_i^k + a_i d_i^k), \quad i = 1, 2, \dots, n$$

段階 3:  $\theta_0^{k+1} = \theta_{n+1}^k$ 。

段階 4:  $\|\theta_0^{k+1} - \theta_0^k\| \leq \varepsilon$  ならば停止、そうでなければ  $k = k+1$  として段階 2 に戻る。

(ii) Hooke-Jeeves 法 (Table 1 の A-16)

この方法は探索移動とパターン移動の 2 段階から成り、探索移動では現在の点の近くで目的関数の減少方向を調べ、パターン移動ではその方向に移動する手順を反復する。

(探索移動)

点  $\theta_i^k$  から  $i = 1, 2, \dots, n$  に関して、 $f(\theta_i^k + a_i^k d_i) < f(\theta_i^k)$  であれば、探索移動は成功で  $\theta_{i+1}^k = \theta_i^k + a_i^k d_i$  とする。そうでなければ、 $a_i^k = -a_i^k$  として同じ手続きを行う。両方とも失敗ならば  $\theta_{i+1}^k = \theta_i^k$  とする。

(パターン移動)

現在の点 (基点)  $\theta_0^k$  から  $\theta_0^k$  と  $\theta_0^{k-1}$  を結ぶ方向に同じ距離だけ移動する。すなわち、 $\theta^k = \theta_0^k + (\theta_0^k - \theta_0^{k-1})$  とする。パターン探索法のアルゴリズムは次の通りである。

段階 1:  $d_i = e_i, a_i^0 > 0, i = 1, 2, \dots, n, \theta_0^0 \in R^n, \varepsilon > 0$  を与え  $k=0$  とする。ここに  $e_i$  は軸方向単位ベクトルである。

段階 2:  $\theta_i^k = \theta_0^k$  として探索移動を行う。全ての方向への探索が失敗したときは、 $a_i^k, i = 1, 2, \dots, n$  の絶対値を小さくして探索移動を反復する。 $\max |a_i^k| \leq \varepsilon$  ならば停止。移動が成功ならば  $\theta_0^{k+1} = \theta_{n+1}^k$  とし、 $k = k+1$  として次の段階へ進む。

段階 3:  $\theta_0^k$  から  $\theta^k$  にパターン移動をして、 $\theta_i^k = \theta^k$  から探索移動を行って  $\theta_0^{k+1} = \theta_{n+1}^k$  とする。

段階 4:  $f(\theta_0^{k+1}) < f(\theta_0^k)$  ならば  $k = k+1$  として段階 3 に進む。そうでなければ段階 2 に戻る。

(iii) Rosenbrock 法 (Table 1 の A-17)

この方法はパターン探索法と同様に、前段階で用いた方向ベクトルの組によって探索移動するが、その際に探索方向は Gram-Schmidt の手順によって生成された直交ベクトルを使う。

いま基点  $\theta_i^k$  において, 正規直交ベクトル  $d_1^k, d_2^k, \dots, d_n^k \in R^n$  と対応するステップ幅  $\delta_i, i=1, 2, \dots, n$  が与えられているとして, 点  $\theta_i$  からある方向  $d_i^k$  に沿っての探索は  $f(\theta_i + \delta_i d_i^k) \leq f(\theta_i)$  を満たすとき成功したと言い, それ以外のとき失敗したと言う. 次に Rosenbrock のアルゴリズムを示す.

段階 1:  $d_i = e_i, \delta_i \neq 0, i=1, 2, \dots, n, \theta_i^0 \in R^n, \varepsilon > 0, \mu \in (-1, 0), \lambda > 1$  を与え  $k=0$  とする. ここに  $e_i$ : 軸方向単位ベクトルである.

段階 2:  $\theta_1 = \theta_1^k$ .

段階 3:  $d_i^k, i=1, 2, \dots, n$  に沿って全ての探索が成功なら  $\theta_{i+1} = \theta_i + \delta_i d_i^k, \delta_i = \lambda \delta_i$ , 失敗なら  $\theta_{i+1} = \theta_i, \delta_i = \mu \delta_i$  とする.

段階 4: 全ての方向に対しても成功で, それに続く失敗が探索の途中で出れば,  $\theta_i^{k+1} = \theta_{n+1}$  において段階 5 へ行く. そうでなければ  $\theta_1 = \theta_{n+1}$  において段階 3 へ進む.

段階 5:  $\|\theta_i^{k+1} - \theta_i^k\| \leq \varepsilon$  ならば停止.

段階 6:  $\alpha_i \in R^1, i=1, 2, \dots, n$  を  $\theta_i^{k+1} - \theta_i^k = \sum_{i=1}^n \alpha_i d_i^k$  で, また  $s_m \in R^n, m=1, 2, \dots, n$  を  $s_m = \sum_{i=1}^m \alpha_i d_i^k$  で与える.

段階 7:  $s_1, s_2, \dots, s_n$  に Gram-Schmidt の直交化を行う. すなわち

$$p_1 = s_1$$

$$p_i = s_i - \sum_{r=1}^{i-1} (s_i, p_r) p_r, \quad i=2, 3, \dots, n$$

$$d_i^{k+1} = p_i / \|p_i\|, \quad i=1, 2, \dots, n$$

として  $d_1^{k+1}, \dots, d_n^{k+1}$  を計算し,  $k=k+1$  として段階 2 に行く.

## 6. 有制約最適化解法

現在, 5種の有制約最適化解法を登録している. その中で代表的な変換法は, 与えられた問題を無制約最適化に変換するため, 本システムで登録した無制約最適化法を直接使用できるので都合がよい.

### 6.1 外点法 (ペナルティ法, Table 1 の B-1)

目的関数を  $f(\theta)$ , 制約関数を  $g(\theta)$  および  $h(\theta)$  において, 制約条件  $g_i(\theta) \leq 0, i=1, 2, \dots, m$  および  $h_j(\theta) = 0, j=1, 2, \dots, r$  の下で  $f(\theta)$  の最小化を考えると, 制約を満足しないとき非常に大きなペナルティ  $r^k P(\theta)$  を目的関数に付け加え, 次の拡大目的関数により無制約の最適化を行う.

$$F(\theta, r^k) = f(\theta) + r^k P(\theta),$$

$$P(\theta) = \sum_{i=1}^m (\max[0, g_i(\theta)]) + \sum_{i=1}^r |h_i(\theta)|, \quad \alpha, \beta \geq 1,$$

ここで  $r^k$  は  $r^{k+1} > r^k (k=1, 2, \dots)$  とする.

### 6.2 内点法 (バリア法, Table 1 の B-2)

上述の制約条件で, 不等式制約のみ適用し, 内点ペナルティ関数  $B(\theta)$  を目的関数に付け加

え、次の拡大目的関数  $F(\theta, r^k)$  の無制約最適化を行う。

$$F(\theta, r^k) = f(\theta) + B(\theta)/r^k,$$

$$B(\theta) = \sum_{i=1}^m 1/(-g_i(\theta)), \quad \alpha > 0,$$

ここで  $r^{k+1} > r^k$  ( $k=1, 2, \dots$ ) とする。この内点法は制約集合の内点から出発し境界を越えない解を求める。

### 6.3 乗数法 (Table 1 の B-3)

内点法と外点法は  $r^k$  の値が大きくなると、拡大目的関数の最小化の数値計算が困難になる。乗数法は、その欠点を改良し内点法と外点法の長所を保持している。乗数法はラグランジュ関数にペナルティ項を付け加えた拡大ラグランジュ関数

$$L(\theta, \lambda, \mu) = f(\theta) + \sum_{i=1}^m \lambda_i g_i(\theta) + \sum_{i=1}^r \mu_i h_i(\theta) + r_1 \sum_{i=1}^m g_i^2(\theta) + r_2 \sum_{i=1}^r h_i^2(\theta)$$

を定義する。ここで、 $r_1, r_2$  は正の母数、ラグランジュ乗数  $\lambda^k = (\lambda_1^k, \lambda_2^k, \dots, \lambda_m^k)$  で、 $\mu^k = (\mu_1^k, \mu_2^k, \dots, \mu_r^k)$  は停留条件から得られる関係式によって更新する。

### 6.4 勾配射影法 (Table 1 の B-4)

制約条件  $a_i \theta \leq b_i$ ,  $i=1, 2, \dots, m$  の下で目的関数  $f(x)$  を最小化するとき、制約集合の境界面へ目的関数の勾配  $\nabla f(\theta)$  を射影して、降下方向の条件  $\nabla f(\theta)d < 0$  を満たす方向ベクトル  $d$  を次のように決め探索を行う。

点  $\theta^k$  において  $r$  個の活性な制約式、 $a_i \theta^k = b_i$ ,  $i=1, 2, \dots, r$  があるとき、これに対応した制約式の係数行列を  $A_i^k = [a_{i1}, a_{i2}, \dots, a_{ir}]$  とおく。この  $A_i$  による部分空間  $N = \{d \mid A_i d = 0\}$  へ  $-\nabla f(\theta)$  を射影するとき、その射影行列は  $P = I - A_i^k (A_i^k A_i^k)^{-1} A_i^k$  であり、方向ベクトルは  $d = -P \nabla f(\theta)$  で与えられる。この方向に沿って直接探索し、もし  $d=0$  となれば  $s = -\nabla f(\theta) A_i^k (A_i^k A_i^k)^{-1}$  とおくと  $\nabla f(\theta) + s A_i^k = 0$  が成立している。ここで  $s$  の成分が全て非負ならば  $\theta^k$  において Kuhn-Tucker 条件が成立し、解が得られる。 $s$  の中に負の成分があれば、その中の絶対値の最大なものに対応する制約式を  $A_i$  から取り除いて再び  $P$  の計算を行う。

### 6.5 逐次2次計画法 (Table 1 の B-5)

6.1 節でラグランジュ関数は  $L(\theta, \lambda, \mu) = f(\theta) + \lambda g(\theta) + \mu h(\theta)$  であり、最適性の必要条件を与える Kuhn-Tucker 条件は、次のようになる。

$$\nabla L(\theta, \lambda, \mu) = \nabla f(\theta) + \lambda \nabla g(\theta) + \mu \nabla h(\theta) = 0$$

ここに

$$\lambda_i g_i(\theta) = 0, \quad i=1, 2, \dots, m$$

$$h(\theta) = 0, \quad g(\theta) \leq 0, \quad \lambda \geq 0$$

である。このとき、等式部分より成る  $(\theta^0, \lambda^0, \mu^0)$  に関する連立非線形方程式を  $(\theta^0, \lambda^0, \mu^0)$  の近傍の点  $(\theta^k, \lambda^k, \mu^k)$  において線形近似すれば、

$$(6.1) \quad \begin{aligned} \nabla^2 L(\theta^k, \lambda^k, \mu^k)(\theta - \theta^k) + \lambda \nabla g(\theta^k) + \mu \nabla h(\theta^k) &= -\nabla f(\theta^k) \\ \lambda_i^k \nabla g_i(\theta^k)(\theta - \theta^k) + \lambda_i g_i(\theta^k) &= 0, \quad i=1, 2, \dots, m \\ \nabla h(\theta^k)(\theta - \theta^k) &= -h(\theta^k) \end{aligned}$$

となる。いま、点  $\theta^k$  の近傍で 6.1 節の問題の目的関数の 2 次近似式のヘシアン  $\nabla^2 f(\theta^k)$  を行列  $B^k$  で近似して得られる 2 次の目的関数を、制約式の線形近似式の下で最小化する 2 次計画問題を導入する。すなわち

$$(6.2) \quad \begin{array}{ll} \text{minimize} & \nabla f(\theta^k)d + d^T B^k d / 2 \\ \text{制約関数} & g(\theta^k) + \nabla g(\theta^k)d \leq 0 \\ & h(\theta^k) + \nabla h(\theta^k)d = 0 \end{array}$$

とする。ここでラグランジュ関数を次のように定義する。

$$L(d, u, v) = \nabla f(\theta^k)d + d^T B^k d / 2 + u g(\theta^k) + u \nabla g(\theta^k)d + v h(\theta^k) + v \nabla h(\theta^k)d$$

この最適解を  $d^k$  とし、対応するラグランジュ乗数を  $u^k, v^k$  とすれば、Kuhn-Tucker 条件

$$\begin{aligned} \nabla f(\theta^k) + B^k d + u^k \nabla g(\theta^k) + v^k \nabla h(\theta^k) &= 0 \\ u_i^k g_i(\theta^k) + u_i^k \nabla g_i(\theta^k) d^k &= 0, \quad i = 1, 2, \dots, m \\ h(\theta^k) + \nabla h(\theta^k) d^k &= 0 \\ g(\theta^k) + \nabla g(\theta^k) d^k &\leq 0, \quad u^k \geq 0 \end{aligned}$$

が成立する。ここで式 (6.1) と式 (6.2) を比較して

$$d^k = \theta^{k+1} - \theta^k, \quad u^k = \lambda^{k+1}, \quad v^k = \mu^{k+1}$$

と考えれば、 $B^k$  は (6.1) の  $\nabla L^2(\theta^k, \lambda^k, \mu^k)$  の役割を果たしていることが判る。したがって、 $B^k$  はラグランジュ関数のヘシアン  $\nabla^2 L(\theta^k, \lambda^k, \mu^k)$  の近似行列となる。 $B^k$  の更新は

$$(6.3) \quad \delta^k = \theta^{k+1} - \theta^k$$

$$(6.4) \quad \gamma^k = \nabla L^T(\theta^{k+1}, u^k, v^k) - \nabla L^T(\theta^k, u^k, v^k)$$

とおき Fletcher-Powell 公式を適用することである。ここで、 $\nabla L(\theta, u, v) = \nabla f(\theta) + u \nabla g(\theta) + v \nabla h(\theta)$  とする。次に逐次 2 次計画のアルゴリズムを示す。

段階 1: 初期点  $\theta^1$  と初期正定対称行列  $B^1$  を選び、 $k=1$  とする。

段階 2: 点  $\theta^k$  における 2 次計画問題

$$\begin{array}{ll} \text{minimize} & \nabla f(\theta^k)d + d^T B^k d / 2 \\ \text{制約関数} & g(\theta^k) + \nabla g(\theta^k)d \leq 0 \\ & h(\theta^k) + \nabla h(\theta^k)d = 0 \end{array}$$

を解き、最適解  $d^k$  と対応するラグランジュ乗数  $u^k, v^k$  を求める。

段階 3:  $\min_a F(\theta^k + a d^k)$  より、 $\theta^{k+1} = \theta^k + a d^k$  とおく。

段階 4: 現在の点  $\theta^{k+1}$  において停止基準を満たせば停止。満たさなければ、式 (6.3)、式 (6.4) より  $p^k, q^k$  を求め、Fletcher-Powell 法より  $B^k$  を  $B^{k+1}$  に更新し、 $k=k+1$  として段階 2 へ戻る。

## 7. 局所解のテスト

上記の非線形最適化解法で反復計算が終了したとき、その収束点が局所解か否かを Box-Wilson 法 (Douke and Asano (1988)) により調査する。尤度関数曲面に多項式を近似して、そのモデルの検定を行う。また最急上昇法によって一層尤度を大きくする方向に移動しながら局所解を見つける。次に、その手順を簡単に示す。

段階 1: 収束点を初期ベクトル  $x^0 = (x_1^0, x_2^0, \dots, x_n^0)$  とし,  $k=1$ .

段階 2: 点  $x^k$  のまわりに  $m=2^n$  個の点を作り, 各々の点の尤度を計算する.

$$\begin{aligned} x_1^k &= (x_1^k - d_1, x_2^k - d_2, \dots, x_n^k - d_n), & x_2^k &= (x_1^k + d_1, x_2^k - d_2, \dots, x_n^k - d_n) \\ x_3^k &= (x_1^k - d_1, x_2^k + d_2, \dots, x_n^k - d_n), & x_4^k &= (x_1^k + d_1, x_2^k + d_2, \dots, x_n^k - d_n) \\ & & & \vdots \\ x_{m-1}^k &= (x_1^k - d_1, x_2^k + d_2, \dots, x_n^k + d_n), & x_m^k &= (x_1^k + d_1, x_2^k + d_2, \dots, x_n^k + d_n) \end{aligned}$$

次いで, 次の新しくコード化した変数に変換する.

$$X_i = ((x_i \pm d) - x_i) / d_i, \quad i=1, 2, \dots, n$$

さらに, Box-Wilson の複合計画を利用し,  $a=2^{n/2}$  とし次の点を追加し各点の尤度を計算する.

$$(\pm a, \pm a, \dots, \pm a)$$

ここで, 点  $x^k$  の近似関数として, 次の 2 次多項式モデルで近似する.

$$\begin{aligned} y &= b_0 + \sum_{i=1}^n b_i X_i + \sum_{i=1}^n b_{ii} X_i^2 + 2 \sum_{i=1}^{n-1} \sum_{i=i+1}^n b_{ii} X_i X_{i+1} \\ &= b_0 + bX' + XB X' \end{aligned}$$

係数  $b_0, b_1, \dots, b_n, b_{11}, b_{12}, \dots, b_{n-1, n}$  は, 最小自乗法によって求め, モデル式の係数に関する仮説検定を行う. 帰無仮説  $H_0: b_{ii} = 0, i, j=1, 2, \dots, n$  が採択されれば, 次の段階 3 へ進む.  $H_0$  が棄却されれば,  $X^* = -B^{-1}b'/2$  を計算する. さらに,  $x^* = x^k + dX^*$  により当初の変数で局所解を示し終了する.

段階 3: 点  $x^k$  での近似として, 1 次式モデルを適用する.

$$y = b_0 + \sum_{i=1}^n b_i X_i$$

ここで, 係数  $b_0, b_1, \dots, b_n$  は最小自乗法で求め, モデル式に関する係数の仮説検定を行う. もし,  $H_0: b_i = 0, i=1, 2, \dots, n$  が棄却されれば, 段階 4 へ進む. もし採択されれば, モデル式は定数として終了する.

段階 4: モデル式の係数をもとに, 次の方向ベクトルを作る.

$$s^k = (b_1 d, b_2 d, \dots, b_n d)$$

この方向ベクトルによって, 最急上昇法を利用し, モデル式と尤度関数曲面の差が大きくなれば, 段階 2 へ行く.

## 8. 数値例

本システムの実行例として, Weibull 分布の母数推定を行う. 数値計算法として, 最急降下法, Newton-Raphson 法, Fletcher-Powell 法, Powell の共役方向法, 座標方向逐次探索法の 5 種の無制約最適化法, および不等式制約条件を付けて内点法, 勾配射影法, 逐次 2 次計画法の 3 種の有制約最適化法を適用し, 有用性を検討する. この際のパソコンは NEC PC-9801 VM2 (640KB) である. すなわち, Weibull 分布の確率密度関数

$$(8.1) \quad f(y) = b(y-c)^{b-1}/a \cdot \exp(-(y-c)^b/a)$$



Table 2. Steepest descent method.

反復回数	$\theta_1$	$\theta_2$	L.F.
0	.8000	.8000	.183441D-46
1	1.0125	1.2709	.644976D-45
2	1.0264	1.2317	.299862D-44
3	1.0360	1.2007	.767337D-44
4	1.0427	1.1760	.138013D-43
5	1.0884	.9778	.144160D-43
6	1.0082	1.1625	.146281D-43
7	1.0157	1.1432	.214467D-43
8	1.0213	1.1279	.271978D-43
9	1.0254	1.1157	.315578D-43
10	1.0285	1.1060	.346542D-43
11	1.0308	1.0982	.367672D-43
12	1.0325	1.0919	.381748D-43
13	1.0337	1.0869	.390993D-43
14	1.0423	1.0466	.391171D-43
15	1.0406	1.0504	.397421D-43
16	1.0391	1.0534	.401455D-43
17	1.0380	1.0558	.404044D-43
18	1.0370	1.0578	.405702D-43
19	1.0362	1.0593	.406761D-43
20	1.0356	1.0605	.407436D-43
21	1.0350	1.0614	.407868D-43
22	1.0346	1.0621	.408143D-43

(CPU time : 5 minutes)

Table 3. Newton-Raphson method.

反復回数	$\theta_1$	$\theta_2$	L.F.
0	.8000	.8000	.183441D-46
1	.8696	.9017	.228422D-44
2	.9277	.9708	.155361D-43
3	.9702	1.0132	.304140D-43
4	.9976	1.0375	.375822D-43
5	1.0137	1.0506	.399548D-43
6	1.0225	1.0575	.406261D-43
7	1.0271	1.0610	.408037D-43
8	1.0295	1.0628	.408494D-43
9	1.0307	1.0637	.408609D-43
10	1.0313	1.0642	.408638D-43

(CPU time : 24 minutes)

Table 4. Fletcher-Powell method.

反復回数	$\theta_1$	$\theta_2$	L.F.
0	.8000	.8000	.183441D-46
1	1.0125	1.2709	.644976D-45
2	1.0125	1.2709	.644976D-45
3	1.1828	1.2546	.243359D-44
4	1.0264	1.2317	.299862D-44
5	.9760	1.2039	.382164D-44
10	1.0884	.9778	.144160D-43
20	1.0285	1.1060	.346542D-43
30	1.0406	1.0504	.397421D-43
40	1.0356	1.0605	.407436D-43
42	1.0350	1.0614	.407868D-43

(CPU time: 20 minutes)

Table 5. Powell method.

反復回数	$\theta_1$	$\theta_2$	L.F.
0	.8000	.8000	.183441D-46
1	.9400	1.0400	.275371D-43
2	.9582	1.0722	.298518D-43
3	.9584	1.0725	.298502D-43

(CPU time: 12 minutes)

Table 6. Cyclic coordinate method.

反復回数	$\theta_1$	$\theta_2$	L.F.
0	.8000	.8000	.183441D-46
1	.9400	1.0400	.275371D-43
2	1.0200	1.0600	.406128D-43
3	1.0300	1.0610	.408205D-43
4	1.0301	1.0620	.408411D-43
5	1.0302	1.0630	.408544D-43
6	1.0312	1.0640	.408632D-43
7	1.0313	1.0650	.408633D-43
8	1.0323	1.0660	.408585D-43
9	1.0324	1.0670	.408455D-43
10	1.0334	1.0680	.408271D-43
20	1.0389	1.0780	.402750D-43
30	1.0489	1.0880	.390435D-43
40	1.0589	1.0980	.371739D-43
50	1.0689	1.1080	.347758D-43
60	1.0789	1.1180	.319767D-43
70	1.0889	1.1280	.289113D-43
80	1.0989	1.1380	.257117D-43
90	1.1089	1.1480	.224994D-43
100	1.1189	1.1580	.193789D-43

(CPU time: 15 hours 25 minutes)

Table 7. Interior point method.

反復回数	$\theta_1$	$\theta_2$	L.F.
0	.8000	.8000	.142864D-46
1	.8212	.8471	.111684D-45
2	.8401	.8833	.163200D-45
3	.8570	.9118	.816534D-49
4	.8723	.9346	.109766D-134

(CPU time : 11 minutes)

Table 8. S.Q.P. method.

反復回数	$\theta_1$	$\theta_2$	L.F.
0	.8000	.8000	.183441D-46
1	.8659	.9460	.529264D-44
2	.8659	.9463	.530451D-44

(CPU time : 4 minutes)

Table 9. Gradient projection method.

反復回数	$\theta_1$	$\theta_2$	L.F.
0	.8000	.8000	.183441D-46
1	.8622	.9378	.446988D-44
2	.8585	.9415	.448435D-44
3	.8586	.9414	.448435D-44

(CPU time : 8 minutes)

において、 $c=0$  とし、準尺度母数  $a$  と形状母数  $b$  を最尤推定する。データとしては、 $a=1.0$ 、 $b=1.0$  を与えて、100 個の Weibull 乱数を使用した。Table 2 は、 $a$  と  $b$  の推定量を  $\theta_1$  と  $\theta_2$  とし、初期値を  $\theta_1=0.8$ 、 $\theta_2=0.8$  としたときの最急降下法による推定結果を示し、22 回の反復後、収束した様子を示している。収束判定基準は今回と前回の推定ベクトル差の内積の平方をとって、 $E=10^{-3}$  とした。Fig. 2 は等高線マップによって最急降下法による反復 10 回までの探索方向を示し、矢印のように次第に最尤推定点に接近していく様子が判る。Fig. 3 は、その一部を詳細に調べる機能として、11 回から 22 回までの探索方向をズームアップして示している。

Table 3 は、Newton-Raphson 法による計算結果を示し、10 回の反復で収束判定基準を満たした。この結果はヘシアンを正定値に保ち、最急降下法より少ない反復回数で収束している。このことは、等高線マップから知れるように、尤度関数曲面が単峰で、Newton-Raphson 法の 2 次近似性に適合するためである。しかし、CPU 時間は最急降下法の 5 分に対し、Newton-Raphson 法では 24 分かかり、これはヘシアンの逆行列の計算量が多いためと思われる。

Table 4 では、Fletcher-Powell 法が 42 回で計算を停止し、最急降下法や Newton-Raphson 法より多くの反復を要しているが、推定値はほとんど同等である。Table 5 より Powell 法は 3 回の反復で収束し、他の解法に比べ速く収束している。しかし、この解法での尤度の値は他の解法に比べ低い値を得ていることから、この解法は必ずしも適当でないことが知れる。

直接探索法の中では座標方向逐次探索法を適用した。Table 6 はその解法による結果を示す

が、7回目まで尤度が増加し8回目以後は減少する傾向を示し、100回の反復計算の後に収束判定基準を満たさず計算を停止し、100回までの時間が15時間25分であった。また、Rosenbrock法も同様の結果を得た。これらの解法は初期値の設定が大きく影響すると思われる。

有制約最適化解法については、 $\theta(1)+\theta(2)\leq 1.8$ の不等式制約条件について、無制約最適化計算の場合と同じ、Weibull乱数を使用し最適化計算を実行した。Table 7は、内点法に最急降下法を利用した計算結果を示している。3回目の反復でペナルティ関数の影響が現れ、4回目で尤度が $10^{-134}$ になったため計算を停止した。Fig. 4は内点法による4回までの探索方向を等高線マップで示している。

他方、逐次2次計画法は反復回数2回で収束判定基準を満たし収束した。逐次2次計画法は、Fig. 5からも判るように単峰の尤度関数曲面における制約領域内での探索問題であるため、その前提となる目的関数の2次近似、制約式の1次近似によく適合している。また、勾配射影法についても3回の反復で収束し、その有効性が確かめられた。これらの試算から、CPU time また尤度を比較したとき、本問題においては、逐次2次計画法、勾配射影法が内点法に比して優位にあることが確かめられた。

## 9. 結 言

本 MLE-SYS システムは、パーソナル・コンピュータを使用した会話型のシステムで、観測データに基づく母数の最尤推定を合理的かつ容易に実行できる。本システムの特徴を以下に示す。(1) 多くの数値解法を収録し、それらの中から自由に解法を選択・変更できるなど、複数解法の相互補完的利用を可能にした。(2) 等高線マップの図示機能は、関数曲面の形状を把握し、初期値の選択や反復計算過程の吟味に役立つ。(3) 尤度関数または対数尤度関数は直接入力するが、解法を選択や母数の設定は番号や記号のみの入力での操作の簡易化を計っている。(4) 数値計算の実行に際し、利用者はその都度メニューやコメントの表示に従い入力するだけで最尤解を得ることができる。

このように本システムは多くの有用な機能を備えている。本システムの積極的な利用によって、経験や知識をシステムに導入し、エキスパートシステムとして、広範な領域での最尤推定の適用を可能にすることが今後の課題である。

## Appendix

本システムの構成と実行手順を明確にするために、その処理手順を示す。

### (1) 起動

本研究に用いたパソコンは NEC PC-9801 VM2 で、他にも TOSHIBA J-3100, IBM PC/AT 等の利用も可能で、主記憶容量 640 KB, ハード・ディスク不要, 演算コプロセッサの対応も可能である。

初めに MS-DOS (V3.1) および MS-FORTRAN (V4.0) を含んだ 5.25" 2HD フロッピー・ディスク (A ディスク) を用意する。次に本 MLE-SYS システムが入ったフロッピー・ディスク (B ディスク) の中から MLE.BAT を A ディスクにコピーする。A ドライブに A ディスクを B ドライブに B ディスクを入れ、MS-DOS を立ち上げる。次に 'MLE' を入力すると本システムが起動する。

Table 10.

$n$ (ケースの数)	$m$ (変数の数)
	変数 1, 変数 2, ..., 変数 $m$
ケース 1	Data matrix $D(i, j)$
ケース 2	
⋮	
ケース $n$	

(2) データファイル名の入力

---

DATA FILE NAME (A10)? --- KEY IN.

---

が表示されたら、データファイル名 XXXX.DAT を入力する。ここに、XXXX.DAT は Table 10 の形式に規定されている。

(3) 母数の個数の入力 (現在は最大 33)

---

NUMBER OF PARAMETERS TO BE ESTIMATED  
--- KEY IN.

---

が表示されたら、推定する母数の数を入力する。但し、最大 33 個までとする。

(4) 尤度関数 (対数尤度関数) の入力

尤度関数が、乗法モデルか対数モデルかの選択を行う。尤度関数  $L(X; D) = \prod_{i=1}^n F(X, D)$  あるいは、対数尤度関数  $L(X; D) = \sum_{i=1}^n \log F(X, D)$  に対して  $F(X, D)$  の式を FORTRAN 書式で入力する。式が複雑な場合は  $F(X, D)$  の中で代替入力方式が可能である。

---

F(X, D)=? (key in by FORTRAN format)  
(DLOG(X(2)\* A)-D(I, 1)\* A)  
NUMBER OF REPLACEMENT? --- KEY IN. 1  
REPLACEMENT (1) --- KEY IN.  
A=D(I, 1)\* \*(X(2)-1.0)/X(1)

---

(5) 制約条件の個数の設定

---

NUMBER OF CONSTRAINT FUNCTIONS --- KEY IN.

---

と表示されたら制約関数の数を入力する。もし制約関数がなければ 0 を入力し、手順 (7) へ移行する。

(6) 有制約解法のメニュー表示と選択

有制約最適化解法の一覧表がメニュー表示された後、

---

SELECT CODE No. OF METHOD --- KEY IN.

---

と表示され、解法の番号を入力する。次に

---

TYPE OF CONSTRAINT :

1.  $G(X, D) \leq 0$     2.  $G(X, D) = 0$

$G(X, D)$ : CONSTRAINT FUNCTIONS

SELECT CODE No.

---

と表示されたら制約式のタイプを選択する。

---

$G(X, D)$ : ? (key in by FORTRAN format)

---

と表示されたら制約関数を FORTRAN 書式で入力する。

(7) 母数の制約設定

母数の制約は、全ての母数について順次、次のように表示される。

---

RANGE OF X(1) :

《M E N U》

- |                         |                          |
|-------------------------|--------------------------|
| 1. NO RESTRICTION       | 2. $X(1) \leq A$         |
| 3. $X(1) \geq A$        | 4. $-1 \leq X(1) \leq 1$ |
| 5. $0 \leq X(1) \leq 1$ | 6. $A \leq X(1) \leq B$  |

A, B: CONSTANT TO BE KEYED IN

PLEASE SELECT ONE!

---

各母数について、適する番号で入力する。2, 3, 6 を選択した場合、A, B の値を入力する。

---

VALUE OF A IN "X(1)>A" --- KEY IN.

---

と表示されるので、A の値を入力する。また1以外については、等号を含むか否か、

---

NEED OF "=" ? (Y/N) --- KEY IN.

---

と表示されるので、必要なら 'Y', 不必要なら 'N' を入力する。

(8) 無制約解法のメニュー表示と選択

無制約解法の一覧表が表示される。

---

ONCE MORE? (Y/N) --- KEY IN.

---

の表示に対して一覧表を再度見るかどうかを 'Y' または 'N' で入力する。

---

PLEASE SELECT CODE No. OF METHOD!  
 --- KEY IN.

---

と表示されるので、解法の番号を入力する。

(9) 導関数の入力

選択された解法によっては、1次偏導関数、2次偏導関数の入力が必要である。各々の解法について自動的に次のように表示され、直接導関数を入力するか数値微分を使用するかを選択する要求がなされる。

---

PLEASE SELECT NEXT No. !  
 1. DO YOU WANT TO KEY IN 1-ST ORDER  
 DERIVATIVES?  
 2. DO YOU WANT TO USE NUMERICAL  
 DIFFERENTIATION OF 1-ST DERIVATIVE?

---

もし '1' を入力したなら次のように表示される。

(1 次偏導関数)

---

DF(X(i)): 1-ST ORDER DERIVATIVES OF  
 F(X, D) BY X(i).  
 X(i)... PARAMETER, D(i, k)... DATA  
 PLEASE KEY IN.  
 DF(X(1))=? (key in by FORTRAN format)  
 DF(X(2))=?  
 :

---

(2 次偏導関数)

---

DDF(X(i), X(j)): 2-ND ORDER DERIVATIVES OF  
 F(X, D) BY X(i), X(j)  
 X(i)... PARAMETER, D(j, k)... DATA  
 DDF(X(1), X(1))=? (key in by FORTRAN format)  
 :

---

入力方法は尤度関数入力の場合と同様である。

(10) 初期値の入力

次のように表示された後に、母数の初期値を順に入力する。

---

INITIAL VALUE OF X(1) --- KEY IN.  
 INITIAL VALUE OF X(2) --- KEY IN.  
 :

---

## (11) 等高線マップの表示

---

NEED OF CONTOUR MAPS? (Y/N) --- KEY IN.

---

が表示され、等高線マップを求めるときは‘Y’を、そうでなければ‘N’を入力する。‘Y’の入力に対しては、

---

NUMBER OF CONTOUR MAPS (MAX: 6) --- KEY IN.

---

と表示された後、1画面に必要なマップ数を入力する。

---

PLEASE SELECT TWO PARAMETERS --- KEY IN.

---

と表示されるので、等高線マップを表示させたい二つの母数の番号を入力する。

---

NUMBER OF COLUMNS AT X(1).

NUMBER OF ROWS AT X(2).

---

と表示された後、第1番目のマップを構成する。数字の列の数と行の数を順に入力する。

---

MIN. AND. MAX. VALUES OF X(1)(F7.3) --- KEY IN.

MIN. AND. MAX. VALUES OF X(2)(F7.3) --- KEY IN.

---

と表示された後、マップ表示したい範囲の最小値、最大値を入力する。

---

DO YOU WANT TO ZOOM? (Y/N) --- KEY IN.

---

と表示されるので、マップの一部をズームアップしたいときには‘Y’を入力する。

## (12) 初期値の変更

---

DO YOU WANT TO CHANGE INITIAL VALUE?

(Y/N) --- KEY IN.

---

と表示されるので、必要なら‘Y’を入力すると手順(10)に戻る。また必要ならば他の母数の組合せについても等高線マップ表示が可能である。

## (13) 収束判定基準の設定

次のような収束判定基準の一覧表が表示される。

---

《M E N U》 (CRITERION OF CONVERGENCE)

1. (distance  $X_{<i+1>}$  and  $X_{<i>}$ )

$E1 > \text{NORM}(X_{<i+1>} - X_{<i>})$

2. (relative distance)

$E2 > \text{NORM}(X_{<i+1>} - X_{<i>}) / \text{NORM}(X_{<i+1>})$

3. (difference  $F(X<i+1>)$  and  $F(X<i>)$ )  
E3>ABS( $F(X<i+1>)-F(X<i>)$ )
4. (relative distance)  
E4>ABS(( $F(X<i+1>)-F(X<i>)$ )/ $F(X<i+1>)$ )
5. E5>NORM(FIRST ORDER DERIVATIVE OF  $F(X)$ )

NORM( $A$ )はベクトル  $A$  の内積の平方根, ABS( $B$ )は  $B$  の絶対値,  $X<i>$ は  $X$  の  $i$  回目の反復値を表す.

PLEASE SELECT!

の表示に対して, 収束判定基準の番号を入力する. 次に,

VALUE OF E --- KEY IN.

と表示されるので, 収束判定基準の許容値を入力する.

(14) 最大反復回数を入力

MAXIMUM ITERATION NUMBER --- KEY IN.

の表示に対して, 停止基準とする反復計算の最大回数を入力する.

(15) 中間結果の出力

計算結果は反復ごとに, 現在用いている解法, 反復回数, 尤度関数の値, 母数の推定値について出力される.

NEED OF CONTOUR MAPS? (Y/N).

と表示されるので, その際の等高線マップが必要ならば 'Y' を入力すると手順 (11) が繰り返される.

(16) 反復回数の追加

最大反復回数に到達したときは,

DO YOU WANT TO CONTINUE? (Y/N) --- KEY IN.

と表示されるので, 入力 'Y' に対して

ADDITIONAL ITERATION NUMBER  
--- KEY IN.

と表示された後に, 追加する反復回数を入力する.

(17) 最終結果の出力

停止基準, 制約条件に抵触した場合, 解が収束したと判断された場合, 最適化計算は終了し, これまでの反復の結果が表示される.

## (18) 初期値の変更

---

DO YOU WANT TO CHANGE INITIAL VALUE? (Y/N)  
 --- KEY IN.

---

と表示されるので、現在の解法で使われている初期値の変更をしたいときは‘Y’を入力する。

(19) 他の無制約解法への移行  
 制約関数がない場合は、

---

PLEASE SELECT NEXT No.!

1. CHANGE TO ANOTHER UNCONSTRAINED METHOD.
2. JOB END.

---

と表示されるので、現在の無制約最適化解法から他の無制約解法へ移行するか、作業を終了するかを選択する。制約関数がある場合は、

---

CHANGE TO ANOTHER UNCONSTRAINED METHOD?  
 (Y/N) --- KEY IN.

---

と表示されるので、他の無制約最適化解法へ移行するには‘Y’を入力し手順(8)へ戻す。

## (20) 有制約解法への移行と終了

---

PLEASE SELECT NEXT No.!

1. CHANGE TO ANOTHER CONSTRAINED METHOD.
2. JOB END.

---

当初使用している有制約解法から他の有制約解法に移行するときは、‘1’を入力して手順(6)へ戻す。また、作業の終了時は‘2’を入力する。

## 参 考 文 献

- Douke, H. and Asano, Ch. (1988). A developmental study on an MLE system, *Proceedings of 5th Japan and Korea Joint Conference of Statistics*, 140-143.
- Kaplan, E.B. and Elston, R.C. (1972). A subroutine package for maximum likelihood estimation, Institute of Statistics Mimeo Series, No. 823.
- 中川 徹, 小柳義夫 (1987). 『最小二乗法による実験データ解析』, 東京大学出版会, 東京.
- Oyanagi, Y. and Nakagawa, T. (1981). SALS User's Manual, SALS Group.

---

この MLE-SYS に関する共同開発をご希望の方は、下記の場所にご連絡下さい。

連絡先 〒 862 熊本市大江町渡鹿 223

九州東海大学工学部情報システム工学科 道家暎幸

## Developmental Study of a New MLE System

Hideyuki Douke

(Faculty of Engineering, Kyushu Tokai University)

Chooichiro Asano

(Faculty of Engineering, Soka University)

The proposed system in the present paper is to estimate parameters on the basis of the method of maximum likelihood by the use of a personal computer. In the fields of physical science, the mathematical models are usually complicated and the ordinary likelihood equations to estimate parameters involved are simultaneously nonlinear. For the general use, the system provides several features to solve real difficulties in view of optimization in numerical analysis.

---

Key words : Maximum likelihood estimation system, controllable choice of methods of numerical analysis, interactive use of menus and messages.