

第1章 時系列と時系列解析

1.1 ファイルからのデータの読み込み

時系列解析のプログラムでは、まず時系列データをファイルから入力することが必要であるが、プログラムごとにいちいちデータを編集しなおしたり、逆にデータを変えるたびにプログラムを変更しないで済むようにデータの入力形式をできるだけ統一しておくことと便利である。本書のプログラムでは一変数時系列はサブルーチン READTS により、また多変数の時系列はサブルーチン READMD により入力するようにしている。

READTS は引数 IDEV で指定されたファイルに次のようなフォーマットに従って一変数時系列がストアされているものと仮定している。

レコード番号	フォーマット	内容
1	A72	データの名前などの情報
2	*	データ数 (N)
3以降	*	時系列データ

実際に解析に用いるデータのファイル名は OPEN 文を用いて指定すればよい。たとえば IDEV=1 の場合

```
OPEN(1,FILE='TEMP.DAT')
READ 文
CLOSE(1)
```

とすることにより、TEMP.DAT というファイルを外部装置 1 と見なしてこのファイルの内容を入力することが可能となる。その他のファイルにあるデータを解析するためには、そのデータを TEMP.DAT にコピーするか OPEN 文のファイル名を変更すればよい。

なお、サブルーチン READTS では TITLE は名前付き共通ブロックの変数となっているので、任意のサブルーチンから TITLE の情報を利用することができる。本書では、解析結果をプリントするためのサブルーチンとプロットするためのサブルーチンがそれぞれ PRXXXX および PTXXXX の

名前で与えられているが、これらのサブルーチンでは TITLE の情報を出
力して解析を行なったデータを明示するようにしている。

同様に、サブルーチン READMD は多変量時系列データの入力のためのサ
ブルーチンである。READMD は引数 IDEV で指定されたファイルに次のよ
うなフォーマットに従って多変量時系列がストアされているものと仮定
している。

c-c-1	レコード番号	フォーマット	内 容
	1	A72	データの名前などの情報
	2	*	データ数 (N), 変数の数 (L), 記述方式 (IFM)
	3	以降	* 多変量時系列データ

ただし、IFM=0 の場合には $((Y(I, J), J=1, L), I=1, N)$ の書式で、また
IFM=1 の場合には $((Y(I, J), I=1, N), J=1, L)$ の書式で多変量時系列がス
トアされているものと仮定して入力を行なう。

多変量時系列の場合には Y は 2 次元配列となるが、サブルーチン READMD
では整合配列の形式を用いている。このとき整合寸法 NMAX は READMD の
仮引数に含まれている。したがって、メインプログラムで

```
PARAMETER( NMAX= ..., LMAX= ... )
DIMENSION Y(NMAX, LMAX)
```

と宣言しておく

```
CALL READMD( IDEV, NMAX, Y, N, L )
```

とするプログラムを書いておけば、パラメータ文の NMAX と LMAX をそれ
ぞれ N と L 以上の値に変更するだけで、サブルーチン READMD は何等変
更しないでいるいろいろなサイズのデータの解析ができるようになる。

1.2 データおよび解析結果の XY プロットへの出 力

時系列解析では大量の数値が入力あるいは出力されるので、これらの
数値をプリントして眺めても、その特徴を捉えることは困難なことが多
い。このような場合、データそのものや解析結果を図示することにより
時系列に関する理解を深めることができる。この節では、データや解析

結果を図示するためのプログラムについて説明する．時系列解析の方法だけに関心がある読者はこの節は読みとばしてよい．

ここで示すプログラムは，大学の計算センターなどで広く利用されている CALCOMP のグラフィックライブラリを利用して XY プロットに作図するためのものである．大型計算機では，実際にはこのプログラムで XY プロット以外にも端末の画面に表示したりレーザープリンタに出力することができる．また，パーソナルコンピュータやワークステーションでも LASER-LIB，PLOT-PC や PLOT-WS などの市販のライブラリを利用することにより，同様にコンソールに表示したりレーザープリンタに直接出力することができる．

ここで使用している XY プロットのサブルーチンは PLOT，PLOTS，PLOTE，PLOTI，NEWPEN，SYMBOL，NUMBER の 7 種類である．これらのサブルーチンの機能は以下に示すようなものである．ワークステーションやパーソナルコンピュータのユーザでこれらのルーチンが利用できない場合には，これらの機能に相当するサブルーチンを作成する必要がある．一方，ワークステーションなどでは S などの便利なグラフィック機能を持つソフトがあるので，これらを利用することも考えられる．2 章以下のプログラムには解析結果をプロットするサブルーチンのほかに，プリンタに出力するサブルーチンを加えている．この出力先のプリンタをハードディスクなどに作られたファイルに変更して結果を保存し，それをグラフィック機能を持つソフトウェアで図示すればよい．また，以下ではプロットの単位は cm と仮定しているが，単位が mm のライブラリの場合には CALL PLOTS のあと

```
CALL FACTOR( 10.0 )
```

として 10 倍に拡大すればよい．

- PLOT(X,Y,IP) 現在点から指定した点 (X,Y) まで移動する (単位: cm) .
 - IP=±2 のとき: 線を描く
 - IP=±3 のとき: 線を描かない
 - ただし，IP<0 のときには (X,Y) を新たな座標原点とする
- PLOTS XY プロットの初期化ルーチン．
- PLOTE XY プロットの終了ルーチン．PLOT(X,Y,999) で代用する場合もある．

- PLOTI 改ページを行なうためのルーチン . PLOT(X,Y,777) で代用する場合もある .
- NEWPEN(IPEN) 作図に用いる色を指定する . 何も指定しないと 1 と仮定される . また , レーザープリンタでは線の太さに対応する .
- SYMBOL(X,Y,HEIGHT,ICHR,ANGLE,NCHR) 文字をプロットする .

X,Y	最初にプロットする文字の左下端の座標 (単位: cm)
HEIGHT	プロットする文字の高さ (単位: cm)
ICHR	プロットする文字が入っている変数名 (文字型) またはコード番号 (16 進数)
ANGLE	プロットする文字と X 軸との角度 (単位: 度)
NCHR	プロットする文字の数
- NUMBER(X,Y,HEIGHT,FNUM,ANGLE,NCHR) 数字をプロットする .

X,Y	最初にプロットする数字の左下端の座標 (単位: cm)						
HEIGHT	プロットする数字の高さ (単位: cm)						
FNUM	プロットする数字が入っている変数名 (実数型)						
ANGLE	プロットする数字と X 軸との角度 (単位: 度)						
NCHR	小数点以下の桁数 <table style="margin-left: 2em;"> <tr> <td style="font-size: 2em;">{</td> <td>= -1</td> <td>整数部のみプロットする</td> </tr> <tr> <td></td> <td>> 0</td> <td>整数部と小数点以下 NCHR 桁までプロットする</td> </tr> </table>	{	= -1	整数部のみプロットする		> 0	整数部と小数点以下 NCHR 桁までプロットする
{	= -1	整数部のみプロットする					
	> 0	整数部と小数点以下 NCHR 桁までプロットする					

プログラム 1.1 はデータをファイルから読み込み , プロットするためのプログラムである . データ読み込みのサブルーチン READTS については 1.4 節で説明した . そのほかはデータのプロットのためのサブルーチンであるが , 次章以下では計算結果の図示にも用いられる .

PTDATA はデータを XY プロッタなどにグラフィック表示するためのサブルーチンである . 本書では PT で始まる名前のサブルーチンはプロッタなどへのグラフィック出力用であることを表す . PTDATA は図 1.2 のように 4 つのサブルーチンを CALL している . はすでに説明したサブルーチンであることを表す .

PLOTS と PLOTE はすでに説明したようにプロッタの初期化と終了ルーチンである . HEADER はデータ名 , 計算プログラム名 , 計算日時 , 計算に用いられたパラメータの値などを出力する . HEADER を CALL する場合には以下の 5 つの引数を設定しておく必要がある . PNAME はプログラム名を入れた文字型変数 , NP はその文字数である . たとえば , プログラム名を PTDATA: PLOT DATA とする場合には

```
PNAME='PTDATA: PLOT DATA' , NP=17
```

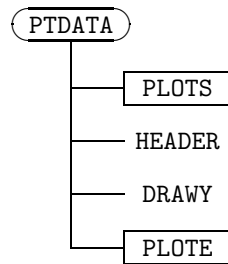


図 1.1: サブルーチン PTDATA の構造

とすればよい。M は表示するパラメータの個数，VNAME はパラメータの名前を入れた 8 バイトの文字型配列，VALUE はパラメータの値を入れた 8 バイトの実数型配列である。SIG2=1.0 と表示するためには

```
VNAME(1)='SIG2□□□□', VALUE(1)=1.0D0, M=1
```

と設定しておけばよい。

DRAWY は長さ N のデータ $Y(1), \dots, Y(N)$ を表示するためのサブルーチンである。縦軸の上限および下限は自動的に設定される。DRAWY には N と Y のほかに 8 個の引数が必要である。FNAME は図の名前を入れた文字型変数，NC はその文字数である。X0 と Y0 は図の左下端の座標 (単位: cm) で座標原点はその後 (X0, Y0) に移動する。WX と WY は図の横軸および縦軸の大きさ (単位: cm) である。IPOS はデータの表示を開始する座標を指定するパラメータで，IPOS=0 の場合は縦軸上から，IPOS=1 の場合には縦軸から WX/N cm だけ右の地点に最初のデータをプロットする。ISW はデータ表示の方法を決めるパラメータである。ISW=0 の場合にはデータを実線で結んで表示する。ISW が 1 以上の場合にはデータを棒グラフ状の縦線で表示する。ただし，ISW=1 の場合には縦線の始点は図の下端となり，ISW=2 の場合には縦座標の値が 0 となる点が始点となる。

サブルーチン HEADER と DRAWY は図 1.3 のような構造になっている。

サブルーチン DATE と CLOCK は計算した日と時間を記録するために CALL しているが，標準の FORTRAN77 の仕様にはないので計算機に応じて変更する必要がある。ただし，これらの CALL 文を削除しても計算自体には何の影響も及ぼさない。

DELX は横軸の目盛りの間隔を決めるため，MAXMIN はデータの最大値と最小値を調べ，縦軸の目盛りの上限，下限および間隔を定めるためのサブルーチンである。AXISXY は座標軸を表示するためのサブルーチンで

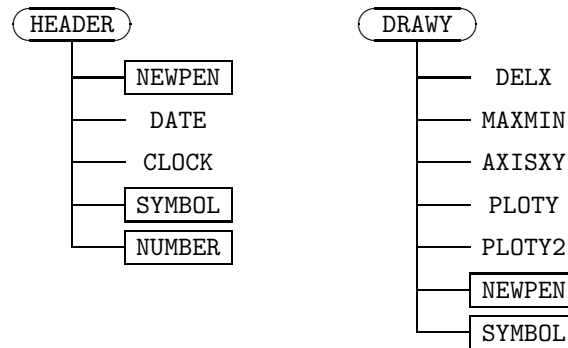


図 1.2: HEADER と DRAWY の構造

ある．入力として 14 個の引数が必要である． X と Y は図の左下端の座標を表す． WX と WY は横軸と縦軸の大きさを表す． $X0$, $X1$ および DX は横軸の目盛りの最小値, 最大値, 間隔を表す．同様に $Y0$, $Y1$, DY は縦軸の目盛りの最小値, 最大値, 間隔を指定する． DWC は目盛りの数字の大きさを指定する．単位はすべて cm である． $IBOX=0$ の場合には縦軸と横軸だけを描くが, $IBOX=1$ とすると図の右端と上端にも線を引き, 表示窓を描く． IX と IY はそれぞれ横軸と縦軸の目盛りの間に描く補助目盛りの数を指定する．

$PLOTY$ は長さ N の時系列データ $Y(1), \dots, Y(N)$ を実線でプロットするためのサブルーチンである． $YMIN$, $YMAX$, DX , DY は $AXISXY$ の場合と同様である． $IPOS$ と ISW はサブルーチン $DRAWY$ で説明したように時系列のプロット開始地点および表示方法を指定する変数である．

$PLOTY2$ は長さ N の時系列データを棒グラフ状の縦線に表示するものである．引数は $PLOTY$ と同様であるが, そのほかに縦線の始点の縦座標 $YLEVEL$ が必要となる．

$AXISXY$, $PLOTY$, $PLOTY2$ は図 1.4 のような構造になっている．

以下に示すソースリスト中でサブルーチン $DRAWY2$ は実際には $CALL$ されていないが, 2 章以降で解析結果の表示に用いられる． $DRAWY$ が座標目盛りを自動的に決定するのに対し, $DRAWY2$ は引数 $XMIN$, $XMAX$, DX , $YMIN$, $YMAX$, DY によって横軸と縦軸の目盛りの下限, 上限および間隔を任意に指定することができる．

作図のための最も基本的なサブルーチンは $AXISXY$, $PLOTY$, $PLOTY2$ の 3 つであり, これらを直接 $CALL$ することにより好みの作図ができる．しかし, 通常はこれらを組み合わせた $DRAWY$ あるいは $DRAWY2$ を使うだけ

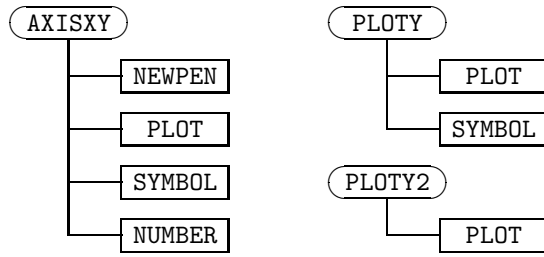


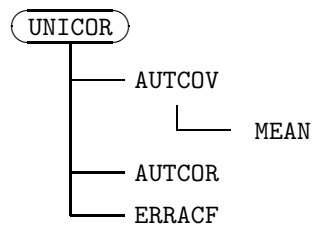
図 1.3: AXISXY, PLOTY, PLOTY2 の構造

で一応の作図は可能となる .

第2章 共分散関数

2.1 自己共分散関数の計算プログラム

プログラム 2.1 の UNICOR は一変量時系列の標本平均，標本自己共分散関数および標本自己相関関数を計算するサブルーチンである．入力としてデータ数 N ，時系列 $Y(1), \dots, Y(N)$ ，計算する最大ラグ数 $MAXLAG$ ，および異常値の判定基準 $OUTMIN$ ， $OUTMAX$ が必要である．UNICOR は内部で 3 つの新しいサブルーチンを CALL しており以下のような構造になっている．



サブルーチン AUTCOV は，まずサブルーチン MEAN を CALL して (2.11) により時系列の標本平均 $YMEAN$ を計算する．次に (2.12) により標本自己共分散関数を計算し仮引数 $C(0), \dots, C(MAXLAG)$ に書き込む．サブルーチン AUTCOV と MEAN では $OUTMIN < Y(I) < OUTMAX$ を満たすデータだけが実際の計算に使われる．すなわち，(2.11) および (2.12) において，この条件を満たす $Y(I)$ だけが加算される．また，データ数 N はこの条件を満たすデータの個数 $NSUM$ で置き換えられる．したがって， $OUTMIN$ および $OUTMAX$ に適当な値を設定することによって，異常値の影響を排除しながら計算をすることができる．ただし，標準では $OUTMIN = -10^{30}$ ， $OUTMAX = 10^{30}$ としてある．またこの機能を利用すると，欠測値がある場合にも，欠測の部分に $OUTMIN$ 以下あるいは $OUTMAX$ 以上の値を代入しておけばよい．

サブルーチン AUTCOR は AUTCOV の計算結果を利用して (2.13) により標本自己相関関数を計算し仮引数 $R(0), \dots, R(MAXLAG)$ にストアする．サブ

ルーチン ERRACF は時系列が白色雑音であると仮定したときの標本自己共分散関数と標本自己相関関数の誤差を評価し、それぞれ (CERR(I); I=0, MAXLAG) および (RERR(I); I=0, MAXLAG) に返す。

UNICOR では 2 次元配列 COV(0:MAXLAG, 4) を用意し、サブルーチン AUTCOV, AUTCOR, ERRACF を CALL するとき、

```

COV(I, 1)  ←→  C(I)
COV(I, 2)  ←→  R(I)
COV(I, 3)  ←→  CERR(I)
COV(I, 4)  ←→  RERR(I)

```

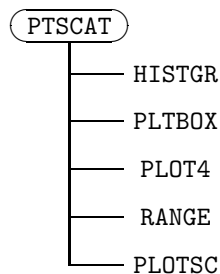
と対応させている。これにより自己共分散関数に関連した 4 つのベクトルをまとめて取り扱うことができる。

PRACOR は UNICOR によって計算された標本自己共分散関数および標本自己相関関数をプリンターへ書き出すためのサブルーチンである。一方、PTACOR は XY プロッタへの出力のためのサブルーチンである。プロッタを利用しない場合にはこの CALL 文をコメント行としておけばよい。

2.2 ヒストグラムと散布図のプログラム

プログラム 2.2 の PTSCAT は多変量時系列のヒストグラムと散布図を描くためのサブルーチンである。L 次元の時系列に対して縦 L, 横 L の正方形の対角線より上に合計 L(L+1) 個の図を描く。対角線上の図にはヒストグラムを、また対角線より上の図には横軸に Y(M, I) を縦軸に Y(M, J) をとった散布図を描く。

PTSCAT は次のような構造になっている。



HISTGR はデータ Y(I), (I=1, N) のヒストグラムを求める。HISTGR では OUTMIN<

$Y(I) < \text{OUTMAX}$ を満たすデータの中で最小値と最大値を求め、この間を NH 個の区間に等分し、各区間に何個ずつデータが入るかを数え、1次元配列 HIST にストアする。

サブルーチン PLTBOX では図の枠を描き、PLOT4 はデータ HIST(I), $I=1, NH$ を棒グラフで表示する。また、RANGE はデータ $Y(I)$, ($I=1, N$) の最小値と最大値、YMIN, YMAX を求める。ただし、OUTMIN 以下のデータおよび OUTMAX 以上のデータは異常値として無視される。PLOTSC は $X(I)$ を横座標、 $Y(I)$ を縦座標とする点を+印で表し、散布図を描く。

2.3 相互共分散関数計算のプログラム

プログラム 2.3 の CRSCOR は多変量時系列の標本平均ベクトル、標本相互共分散関数、標本相互相関関数を計算するサブルーチンである。入力としてデータ数 N 、変数の数 L 、時系列 ($Y(I, J)$, $I=1, N$; $J=1, L$) および計算する相互共分散関数の最大ラグ数 LAG が必要である。

サブルーチン CRSCOR は内部でサブルーチン MEAN を CALL している。サブルーチン MEAN は時系列 $Y(I, J)$ の平均を計算し結果を配列 YMEAN(I) にストアして返す。CRSCOR は MEAN の結果を利用して (2.21) により標本相互共分散関数を計算し、結果を3次元配列の仮引数 ($C(K, I, J)$, $K=0, \text{LAG}$; $I=1, L$; $J=1, L$) にストアする。以上の計算ではプログラム 2.1 の UNICOR の場合と同様に、 $\text{OUTMIN}(J) < Y(I, J) < \text{OUTMAX}(J)$ を満たすデータだけが実際の計算に用いられる。次に (2.22) により標本相互相関関数を計算し、結果を3次元配列の ($R(K, I, J)$, $K=0, \text{LAG}$; $I=1, L$; $J=1, L$) にストアする。

PRMCOR は CRSCOR によって計算された標本平均ベクトル、標本共分散関数、標本相互相関関数の数値を出力するためのサブルーチンである。一方、PTMCOR はこれらをプロットに出力する。

第3章 スペクトルとピリオドグラム

3.1 ピリオドグラム計算のプログラム

サブルーチン FOURIE は $x(1), \dots, x(N)$ のフーリエ変換を行なうサブルーチンであるが, Goertzel 法による効率的な計算を行なっている. 一般に, 数列 x_0, \dots, x_{L-1} のフーリエ・コサイン変換およびフーリエ・サイン変換

$$X_c(f) = \sum_{n=0}^{L-1} x_n \cos 2\pi n f, \quad X_s(f) = \sum_{n=0}^{L-1} x_n \sin 2\pi n f$$

を定義通りに計算すると, それぞれ L 回の積和演算と L 回の三角関数の評価が必要となる. 三角関数の加法定理を利用した Goertzel 法を用いると三角関数の評価は $\cos 2\pi f$ と $\sin 2\pi f$ の 2 回だけでよい. これは $a_0 = 0$, $a_1 = 1$ とし, $n = 2, \dots, L-1$ について $a_n = 2a_{n-1} \cos 2\pi f - a_{n-2}$ によって a_n を求めておけば

$$\sin 2\pi n f = a_n \sin 2\pi f, \quad \cos 2\pi n f = a_n \cos 2\pi f - a_{n-1}$$

によって $\sin 2\pi n f$ および $\cos 2\pi n f$ が求められることを利用したものである.

第4章 モデリング

4.1 密度関数を与えるプログラム

本節では、図 4.1 や図 4.2 で用いられた密度関数を定義するプログラムを示す。これらのサブルーチンやファンクションは 14 章の非ガウス型モデルや 15 章のシミュレーションでもノイズの分布として利用される。

プログラム 4.1 のサブルーチン DENSTY は仮引数 DIST で指定された密度関数 $f(x)$ について $f(x_1), \dots, f(x_k)$ の値を計算し、仮引数 F(1), \dots , F(k) にストアして返す。ただし $x_1 = XMIN$, $x_k = XMAX$ で途中の点 x_i は

$$x_i = x_1 + \frac{i-1}{k-1}(x_k - x_1), \quad i = 1, \dots, k \quad (4.1)$$

により等間隔に割り当てられる。密度関数 $f(x)$ には通常、平均や分散などのパラメータが必要であるが、これらの値は必要な個数だけ次元配列 PARAM に入れておけばよい。また仮引数 DIST は密度関数を指定するファンクション名なので DENSTY を CALL するときの実引数として実際に使用するファンクションの名前を与えてやればよい。ただし、メインプログラムでは前もってその名前を EXTERNAL 指定しておく必要がある。

プログラム 4.1 のファンクション GAUSS は与えられた x に対し正規分布の密度関数の値

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{(x-\mu)^2}{2\sigma^2} \right\} \quad (4.2)$$

を GAUSS にストアして返す。正規分布には平均 μ と分散 σ^2 の 2 つのパラメータがあるので、PARAM(1) に μ の値、PARAM(2) に σ^2 の値を代入しておく必要がある。プログラム 4.1 にはこのほか、以下のように図 4.1 や図 4.2 で用いた分布が与えられているので、プログラムの実行時に変数 MODEL の値を以下の表のように指定することにより、これらの分布を選択できる。また、その他の密度関数が必要な場合には、ファンクション USERF を適当に修正し、MODEL=0 と指定すればよい。現在のプログラムでは USERF には例として両側指数分布の密度関数を与えられている。

プログラム 4.1 ではこのほか、関数評価を行なう区間の両端 $XMIN$, $XMAX$ と適当な個数のパラメータ $PARAM(I)$ を与える必要がある。MODEL=0 の場合にはパラメータの個数 $NP(0)$ も必要である。

PTDIST は DENSTY によって一次元配列 F にストアされた密度関数を図示するためのサブルーチンである。一方、PRDIST は F をプリンタに出力するためのサブルーチンである。

MODEL	分布	ファンクション	パラメータ
1	正規分布	GAUSS	μ, σ^2
2	コーシー分布	CAUCHY	μ, τ^2
3	ピアソン分布	PEARSN	μ, τ^2, b
4	指数分布	EXPNTL	λ
5	χ^2 分布	CHISQR	k
6	2重指数分布	DBLEXP	なし
7	一様分布	UNIFRM	a, b
0	任意の密度関数	USERF	適宜

4.2 最尤法のためのプログラム

プログラム 4.3 は付録 A の数値的最適化のサブルーチン DAVIDN の利用法を説明するためにコーシー分布の 2 つのパラメータの最尤推定値を求める例を示したものである。

サブルーチン DAVIDN によって最尤推定値を求めるためには、まず問題に応じて評価関数とその一階微分を計算するサブルーチンを作成する必要がある。DAVIDN は与えられた関数を最小化するサブルーチンなので実際には尤度あるいは対数尤度の符号をかえたものを評価関数としておく。ここでは例として表 4.3 の計算のためのサブルーチン FFCACY を示す。まずメインプログラムで FFCACY を EXTERNAL 宣言する。次に $A(1)=0.0$, $A(2)=1.0$, $NPARAM=2$, $IPR=0$, $NDIF=0$ と設定して DAVIDN を CALL すればよい。A はパラメータベクトルの初期値、NPARAM はパラメータの次元を表す。また、NDIF=0 は一階微分を計算できることを意味している。

FFCACY はコーシーモデルの対数尤度とその微分を求めるためのサブルーチンである。入力はパラメータ数 K, パラメータを入れた 1 次元配列 $A(1), \dots, A(K)$ である。また、データ y_1, \dots, y_n およびデータ数 N は共

通ブロックの変数としてメインプログラムから直接引き渡される。コーシー分布は 2 個のパラメータを持つので、実際には $K=2$ となっており、 $A(1)$ には μ 、 $A(2)$ には τ^2 の値が入っている。FFCACY はこのパラメータに対応する対数尤度の符号をかえたものを F に、また $-\partial l / \partial \mu$ と $-\partial l / \partial \tau^2$ の値を $G(1)$ と $G(2)$ にストアする。IFG は通常は 0 としておくが、パラメータに制約条件がある場合に、その制約条件を満たさないときは 1 としておけばよい。現在は $A(2) \leq 0$ のときに 1 としている。ただし、パラメータの変数変換を行なって $A(2)$ を $\log \tau^2$ とすれば、このような制約条件は不要となる。

FFCACY では F と $G(I)$ の両方を計算したが、 $G(I)$ を解析的に求めるのが困難な場合には、 F だけを計算するサブルーチンを作成し、NDIF=1 または 2 と設定して DAVIDN を CALL すればよい。この場合には DAVIDN はサブルーチン FUNCND を CALL して $G(I)$ を数値微分によって求める。NDIF=1 の場合には片側差分、NDIF=2 の場合には両側差分による数値微分を行なう。ただし、片側差分では関数の極値をとる点でも $G(I)$ は完全に 0 とはならないので、1 回の関数計算に非常に長い時間を要する場合以外は NDIF=2 を用いるべきである。

第5章 最小二乗法

5.1 最小二乗法による回帰分析プログラム

プログラム 5.1 はハウスホルダー変換に基づく最小二乗法によって回帰モデルを推定するサブルーチン REGRES の使い方を示したものである。

1 章で示したように READTS はデータを読み込むためのサブルーチンである。サブルーチン REDUCT はハウスホルダー変換によって上三角行列を求める。REDUCT は内部で SETX を CALL してデータ行列 (5.14) を作成し、次に HUSHLD を CALL してハウスホルダー変換を行なう。ただし、サブルーチン名 SETX は仮引数であるので、モデルに対応するデータ行列 (5.14) や (5.27) を作成するためのサブルーチンを作成し、その名前を EXTERNAL 宣言して実引数として受け渡すことにより、さまざまなモデルに対応することができる。

2 次元配列 X の第 1 添字の大きさがデータ数より小さく $MJ < N$ のときには、5.4 節の方法によりデータを分割して処理する。データ数が小さく $N \leq MJ$ の場合には REDUCT を CALL する代わりに

```
CALL SETX( Y,0,N,K,MJ,0,X )
CALL HUSHLD( X,D,MJ,N,K+1 )
```

としても同一の結果が得られる。

一般に SETX は次のような行列を作成する。 $N0=0$ の場合にはデータの先頭の MJ 個の観測値だけを用いて $MJ \times (K+1)$ 行列

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1m} & y_1 \\ x_{21} & \cdots & x_{2m} & y_2 \\ \vdots & & \vdots & \vdots \\ x_{p1} & \cdots & x_{pm} & y_p \end{bmatrix} \quad (5.1)$$

を作る。ただし、 $p = MJ$ とする。一方、 $N0 > 0$ の場合には (5.27) に対応

して、既に求められている上三角行列の下に新しいデータを付加し

$$X = \begin{bmatrix} s_{11} & \cdots & s_{1m} & s_{1,m+1} \\ & \ddots & \vdots & \vdots \\ & & s_{mm} & s_{m,m+1} \\ 0 & & & s_{m+1,m+1} \\ x_{n_0+1,1} & \cdots & x_{n_0+1,m} & y_{n_0+1} \\ \vdots & & \vdots & \vdots \\ x_{p1} & \cdots & x_{pm} & y_p \end{bmatrix} \quad (5.2)$$

を作成する。ただし、 $p = MJ + NO - K - 1$ である。

プログラム 5.1 には例としてサブルーチン SETXTP を用意している。サブルーチン SETXTP は図 5.1 で用いた三角多項式回帰モデルの推定のために観測値 $\{y_n\}$ から行列 X を作るもので説明変数を $x_{n1} = 1$, $x_{n,2j} = \sin j\omega n$, $x_{n,2j+1} = \cos j\omega n$ と仮定したことに相当する。

REGRES は AIC によって次数選択を行ない、最小二乗法によって回帰係数を求めるためのサブルーチンである。一方、PRREG は REGRES の計算結果をプリンタに、PTTPL は原データと回帰多項式をプロッタ等へ出力するためのサブルーチンである。

第6章 ARMAモデルによる時系列の解析

6.1 ARMAモデルによる時系列解析のプログラム

プログラム 6.1 は一変量の ARMA モデルによる時系列解析のためのサブルーチンの利用法を示したものである。メインプログラムでは AR 次数 M , MA 次数 L , AR 係数 $A(I), I=1, M$, MA 係数 $B(I), I=1, L$ および白色雑音の分散 $SIG2$ を入力したあと 9 つのサブルーチンを CALL している。

サブルーチン IMPULS は入力として $M, L, A(I), B(I), K$ を与えると (6.8) によりインパルス応答関数を計算し $G(I), I=0, K$ にストアする。

サブルーチン ARMCOV に入力として $M, L, A(I), B(I), SIG2, K$ を与えると (6.11) により自己共分散関数 C_0, \dots, C_k を計算し一次元配列 COV にストアする。

ARCOEF は PARCOR から AR 係数を求めるためのサブルーチンである。AR 次数を M に, PARCOR を一次元配列 PAR にストアして ARCOEF を CALL すると, (6.13) により AR 係数を計算し A に結果が返される。

PARCOR は逆に AR 係数から PARCOR を求めるためのサブルーチンである。AR 次数を M に, AR 係数を一次元配列 A にストアして CALL すると, (6.15) によって PARCOR を計算し一次元配列 PAR にストアする。ただし, $L > 0$ の場合には MA モデルまたは ARMA モデルを AR モデルで近似したときの係数が必要となるが, 第 7 章で説明するサブルーチン ARYULE を用いると, ARMCOV によって求めた $COV(I)$ から PARCOR を求めることができる。

サブルーチン ARMASP は ARMA モデルのパワースペクトルを求めるためのサブルーチンである。 $M, L, A(I), B(I), SIG2$ およびスペクトルを計算する周波数の数 NF を与えると, $F=I/(2*NF), I=0, NF$ によって定まる $NF+1$ 個の周波数 F について (6.19) によってパワースペクトルを計算し, 一次元配列 $SP(I), I=0, NF$ にストアして返す。

CHROOT は AR オペレータまたは MA オペレータの特性根を求めるためのサブルーチンである。次数を K に、係数を一次元配列 C にストアして CHROOT を CALL すると、代数方程式

$$z^k - c_1 z^{k-1} - \dots - c_k = 0 \quad (6.1)$$

の根 $z = \alpha + \beta i$ を k 個求め、実部 α と虚部 β をそれぞれ ROOT(I,1) と ROOT(I,2), $I=1, K$ にストアする。したがって、CHROOT が計算する特性根は (6.21) や (6.22) の特性根ではなく、その逆数であることに注意する必要がある。また、代数方程式の根を数値計算によって正確に求めることは一般には困難なので、 K が大きな場合や重根に近い根があるときには求められた根の精度に注意する必要がある。このような場合には他の計算法も試みてみるとよい。

6.2 多変量 AR モデルによるスペクトル解析のプログラム

プログラム 6.2 は多変量 AR モデルによるスペクトル解析のためのサブルーチンの利用法を示したものである。メインプログラムでは、IDEV で指定されたファイルから M, L, E, A を入力する。 M と L は整数でそれぞれ多変量 AR モデルの次数、時系列の次元を表す。 E はノイズの共分散行列、 A は AR 係数行列である。

MARSPC は多変量 AR モデルからクロススペクトルを求めるためのサブルーチンである。整数 NF を与えると周波数 $II/(2*NF)$, $II=0, NF$ に対してクロススペクトル $P(II)$, 振幅スペクトル $AMP(II)$, 位相スペクトル $ANG(II)$ およびコヒーレンシー $COH(II)$ を計算する。さらに、累積ノイズ寄与率も計算し $FNC(II)$ にストアする。ただし、これらの量は I と J のいろいろな組合せに対して計算されるので、 P, AMP, ANG, COH, FNC は 3 次元のアレイである。 $CINV$ は複素数を成分とする $M \times M$ 行列 X の逆行列を計算するためのサブルーチンである。

PRMSPC はクロススペクトルとノイズ寄与率をプリントする。一方、PTCSPC はクロススペクトル、コヒーレンシーおよびノイズ寄与率をそれぞれ一枚の図にプロットする。

第7章 ARモデルの推定

7.1 一変量 ARモデル推定のプログラム

プログラム 7.1 は一変量 AR モデル推定のためのサブルーチンを示したものである。まず、メインプログラムでは LAG, ISW を読み込む。LAG は推定する AR モデルの最大次数である。ISW は推定法を指定するパラメータで、1 のときはユールウォーカー推定値をレビンソン法で求めるサブルーチン ARYULE, 2 のときは最小二乗法のサブルーチン REGRES, 3 以上のときは PARCOR 法のサブルーチン ARPCOR が CALL される。

ARYULE はレビンソンのアルゴリズムによる AR モデル推定のためのサブルーチンである。入力として、最大次数 MAXM, 自己共分散関数 $C(I), I=0, \text{MAXM}$, 原データのデータ数 N を与えると、レビンソンのアルゴリズムによって AR モデルのパラメータを推定し、AIC 最小のモデルの次数を MAR にストアする。また、この計算過程で求められる各次数の AR 係数, PARCOR, 分散および AIC の値がそれぞれ二次元配列 A および一次元配列 PARCOR, SIG2, AIC に与えられる。

ただし、このサブルーチンを CALL する前に UNICOR あるいは AUTCOV を CALL して自己共分散関数を求めておく必要がある。

REGRES はハウスホルダー変換によって回帰モデルの最小二乗推定値を求めるための、5 章で説明したサブルーチンである。このサブルーチンを用いて AR モデルを推定するためには、まず (7.19) の行列を作成するサブルーチン SETXAR を EXTERNAL 宣言し、5.5 節の REDUCT を CALL して (7.20) の上三角行列を求める。次にサブルーチン REGRES を CALL して (7.21) および (7.22) により AR モデルを推定する。

ARPCOR は PARCOR 法による AR モデル推定のためのサブルーチンである。ただし、ISW=1 のときは (7.32), ISW=2 のときは (7.33), ISW=3 のときは (7.34) による推定を行なう。入力として、データ数 N, 時系列 $Y(1), \dots, Y(N)$, 最大次数 K および ISW を与えると、PARCOR, MAR, SIG2 および AIC が求められる。ARPCOR を CALL するときには作業用の一次元配列 $(FE(I), I=1, N)$ と $(BE(I), I=1, N)$ を用意し引数として渡す必要が

ある．FE と BE にはそれぞれ各段階での前向きと後向きの予測誤差が一時的にストアされる．

ARMASP はすでに6章で説明した ARMA モデルのスペクトルを求めるためのサブルーチンで，4番目の引数 L を 0 として CALL すると AR モデルのスペクトルが得られる．PTAR は AR モデル推定の結果を図示するため，また PRARSP はプリンタに出力するためのサブルーチンである．

7.2 ユールウォーカー法による多変量 AR モデル推定のプログラム

プログラム 7.2 は多変量 AR モデル推定のためのサブルーチンを示したものである．MYULE はレビンソンのアルゴリズムを用いて多変量 AR モデルのユールウォーカー推定値を計算するためのサブルーチンである．入力として，変数の数 L，原データのデータ数 N，最大次数 LAG および自己共分散関数 $C(M, I, J), I=1, L; J=1, L; M=1, LAG$ を与えると多変量版のレビンソンのアルゴリズムによってパラメータを推定し，AIC 最小のモデルの次数を MMIN に，分散共分散行列を二次元配列 VMIN に，その係数を三次元配列 $AMIN(M, I, J), I=1, L; J=1, L; M=1, MMIN$ にストアする．また，この計算過程で求められる AIC の値が一次元配列 AIC に与えられる．ただし，このサブルーチンを CALL する前に 2.7 節で説明した CRSCOR を CALL して相互共分散関数を求めて三次元配列 C にストアしておく必要がある．

PRMAR は AR モデル推定の結果をプリンタに出力するためのサブルーチンである．PRMAR の出力をファイルに出力して，プログラム 6.2 の入力として用いるか，MYULE のあとにプログラム 6.2 の MARSPC を CALL することにより，推定された多変量 AR に基づくスペクトル解析を行なうことができる．

7.3 最小二乗法による多変量 AR モデル推定のプログラム

プログラム 7.3 はハウスホルダー法に基づく最小二乗法によって多変量 AR モデルを推定するためのプログラムである．MREDCT は一変量時系列の場合のサブルーチン REDUCT と同様に時系列データから上三角行列

を求めるためのサブルーチンである。まず (7.43) の行列を作成するサブルーチン SETMAR を EXTERNAL 宣言して MREDCT を CALL するとハウスホルダー変換を繰り返し適用して (7.44) の上三角行列を求める。入力として、多変量時系列 Z 、最大次数 LAG、データ数 NMK および次元 ID を与えると結果を X に返す。MARFIT はその結果を利用して (7.46) および (7.49) により AIC を計算して最適な次数をさがし (サブルーチン MAICE)、最小二乗法により係数行列を推定し (SRCOEF と MCOEF)、AIC 最小のモデルの次数を LMAX に、分散共分散行列を二次元配列 E に、その係数を三次元配列 $B(I, J, L)$ 、 $I=1, K; J=1, K; L=1, MMIN$ にストアする。PRMAR2 は推定された AR 次数、AR 係数行列、分散共分散行列などを出力するためのサブルーチンである。

第8章 局所定常 AR モデル

8.1 局所定常 AR モデル推定のプログラム

プログラム 8.1 の LSAR1 は局所定常 AR モデルをあてはめ、自動的に変化時点を検出するプログラムである。LSAR1 は NS 個の時系列 $Y(L+I)$, $I=1, NS$ ごとにサブルーチン LOCAL を CALL し、直前の時系列と併合すべきか分割すべきかを AIC を用いて判断する。

LOCAL では (8.12)~(8.21) に従って分割モデルの AIC と併合モデルの AIC を求めている。LOCAL は新しいデータを併合したかどうかの判断とともに、新しく得られた AR モデルの次数、分散および AR 係数を MF, SDF, (A(I), I=1, MF) にストアする。ARMASP はプログラム 6.1 で示したサブルーチンで、各時点での AR モデルの推定値を与えるとスペクトルの対数値を計算して (SP(I), I=0, NF) に返す。PTLSP は時系列と各定常区間で推定されたスペクトルを図示する。

8.2 区分時点を精密に推定するためのプログラム

プログラム 8.2 の LSAR2 は局所定常 AR モデルの区分時点を精密に推定するためのプログラムである。入力として、K, NO, N1, N2, NE を与える必要がある。K は各区間で推定する AR モデルの最大次数、NO と NE はモデルの推定に使うデータの最初の点と最後の点、N1 と N2 は区分時点の候補の最小値と最大値である。したがって、 $NO < N1 < N2 < NE$ を満たす必要がある。

サブルーチン UPDATE は前半のデータにあてはめられた AR モデルの AIC を求め、1 次元配列 (AIC1(I), I=1, N) にストアする。UPDATE ではまず、REDUCT を CALL することによりデータ $Y(NO+1), \dots, Y(N1)$ より (8.23) の S が求められ、2 次元配列 $(X(I, J), I=1, M+1; J=1, M+1)$ に返される。次に SETXAR と HUSHL2 が CALL されるたびに (8.27) の X_1 が 2 次元配列 $(X(I, J), I=1, M+1; J=1, M+1)$ に作られ、ハウスホルダー変換

により (8.28) の R が求められ上三角行列が更新される。

REGRES はこれらの結果を利用して AIC による次数選択を行ない最適な次数の AIC を返す。UPDATE では新しい上三角行列 R を求めるたびに REGRES を CALL して AIC の最小値を求め、その値を AIC1(J) にストアする。サブルーチン SETXAR, REDUCT, REGRES, HUSHL2 は AR モデルの推定のプログラムに用いられたものと同一である。同様に BUPDAT は後半のデータにあてはめられた AR モデルの AIC を求め (AIC2(I), I=1, M) にストアする。したがって (AICS(J), J=1, M) は (8.31) の AIC_j となり、AICMIN はその最小値、MMIN はその時点を表す。PRLSAR は変化時点とその AIC の値および (AICS(I), I=1, M) を出力する。一方、PTLSAR は AICS(I) とデータ Y(I) をプロッタに出力する。

```

02050      SUBROUTINE COPY( X,K,II,JJ,MJ1,MJ2,Y )
02060 C
02070 C    ... Make a copy of X on Y
02080 C
02090      IMPLICIT REAL*8( A-H,O-Z )
02100      DIMENSION X(MJ1,1) , Y(MJ2,1)
02110 C
02120      DO 10 I=1,K
02130         I1 = I + II
02140         I2 = I + JJ
02150         DO 10 J=1,K
02160            10 Y(I2,J) = X(I1,J)
02170 C
02180      RETURN
02190      E N D

```

第9章 状態空間モデルによる時系列の解析

9.1 カルマンフィルタの計算プログラム

プログラム 9.1 の FILTER は (9.1), (9.2) の状態空間モデルに対するカルマンフィルタのサブルーチンである。ただし, F, G, H は時刻 n に依存せず一定で, $\ell = 1$ と仮定する。したがって, H は m 次元横ベクトル, R はスカラーとなる。より一般の場合の拡張については次節で説明する。時系列 y_n を 1 次元配列 $Y(II), II=1, N$ に与え, このサブルーチンを CALL することにより, 状態ベクトル x_n の 1 期先予測, フィルタおよび対数尤度を計算することができる。FILTER では, $OUTMIN < Y(II) < OUTMAX$ を満たすデータだけが実際の計算に用いられる。したがって, $OUTMIN$ と $OUTMAX$ に適当な値を設定することにより, 観測値のうち一部を異常値とみなして排除することができる。標準では $OUTMIN = -10^{30}$, $OUTMAX = 10^{30}$ と設定している。データに欠測がある場合には, 欠測の部分に $OUTMIN$ 以下あるいは $OUTMAX$ 以上の値を代入しておけば自動的に欠測の処理が行なわれる。 x_n の 1 期先予測の平均ベクトル $x_{n|n-1}$ は $(XP(I), I=1, M)$ に, またその分散共分散行列 $V_{n|n-1}$ は $(VP(I, J), I=1, M; J=1, M)$ に求められる。同様に x_n のフィルタの平均ベクトル $x_{n|n}$ は $(XF(I), I=1, M)$ に, またその分散共分散行列 $V_{n|n}$ は $(VF(I, J), I=1, M; J=1, M)$ に求められる。ただし, これらの配列は時刻 n の経過にともなって順次上書きされて失われてしまうので, それぞれ $(XPS(I, II), VPS(I, J, II), XFS(I, II), VFS(I, J, II), I=1, M; j=1, M; II=1, N)$ という 2 次元あるいは 3 次元の配列にストアされる。予測値やフィルタの値をプロットしたり, 平滑化を行なうためにその後サブルーチン SMOOTH を CALL した場合にはこれらの計算結果が利用される。パラメータ推定やモデル選択に利用される対数尤度は (9.23) に従って計算され, FF にストアされる。FILTER を CALL するとき, ISW=0 と指定した場合には SIG2 を (9.28) により推定し, 対数尤度は (9.29) に従って計算される。

FILTER を CALL する目的が時系列モデルのパラメータ推定のためで対数尤度の値だけが必要な場合には、FILTER を CALL するとき NDIM=1 とおいておくことにより、実行時に必要なメモリを節約することができる。FILTER で必要なメモリの大半は3次元配列 VPS と VFS に用いられるが、NDIM=1 とおくことにより、無視できる程度に小さくすることができる。これにより、きわめて大量のデータあるいは高次元のモデルでメモリの制約にかかる場合にもカルマンフィルタを実行することができる。

サブルーチン FILTER を NDIM=N として CALL したあと、SMOOTH を CALL することにより x_n の平滑化を行なうことができる。平均ベクトル $x_{n|N}$ は (XS(I), I=1, M) に、またその分散共分散行列は (VS(I, J), I=1, M; J=1, M) に作られ、それぞれ2次元配列 (XSS(I, II), I=1, M; II=1, N) および3次元配列 (VSS(I, J, II), I=1, M; J=1, M; II=1, N) に保存される。平滑化を行なう場合には NDIM=N として $x_{n|n-1}$, $x_{n|n}$, $V_{n|n-1}$, $V_{n|n}$ をストアしておく必要があるが、どうしてもメモリが不足する場合には $x_{n|n}$ と $V_{n|n}$ だけを保存し、 $x_{n+1|n}$ と $V_{n+1|n}$ は (9.12) によって計算するようにプログラムを修正すればよい。

実際に FILTER と SMOOTH を実行するためには、状態空間モデルの F , G , H , Q , R を定めるサブルーチンと、初期状態 $x_{0|0}$, $V_{0|0}$ を定めるサブルーチンを CALL する必要がある。プログラム 9.1 の SETFGH と ISTATE には長期予測や補間を行なうために AR モデルの状態空間モデルを定義するためのサブルーチンを示してあるが、この二つのサブルーチンを適当に変更することにより様々なモデルのフィルタリングや平滑化を行なうことができる。

PTSTAT と PRSTAT は、欠測値の補間値や長期予測値とその推定誤差をプロッタとプリンタに出力するためのサブルーチンである。また、計算には用いなかった実測値を ○ 印で表示する。

メインプログラムはこれらのサブルーチンの利用例を示したものである。以下のパラメータを入力する必要がある。NFE は時系列を用いてフィルタリングを行なう最終時点、NPE は予測の終了時点で、 $NFE \leq NPE$, $NFE \leq N$ を満たす必要がある。M, TAU2, AR(I) は使用する AR モデルの次数、分散、自己回帰係数である。OUTMIN, OUTMAX は異常値の範囲を指定するパラメータである。NMISS は欠測区間の数、NO(I) と NN(I) はそれぞれ I 番目の欠測区間の最初の時点および欠測数である。このプログラムでは、この NMISS 個の欠測区間と NFE+1 から NPE の間の部分を平滑化により推定し表示する。

第10章 ARMAモデルの推定

10.1 ARMAモデルの最尤推定のプログラム

プログラム 10.1 は ARMA モデルのパラメータの最尤推定値を求めるためのプログラムである。AR 次数 M 、MA 次数 L 、パラメータの初期値の設定のしかたを指定する $IPARAM$ および時系列 $Y(I), I=1, N$ を入力する必要がある。 $L=0$ とすると AR モデル、 $M=0$ とすると MA モデルが得られる。 サブルーチン $SPARA1$ では AR 係数 $AR(I), I=1, M$ および MA 係数 $CMA(I), I=1, L$ の初期値を自動的に設定している。 $IPARAM=0$ の場合には、この値を使って自動的に計算を開始する。しかし、デフォルトの初期値では常により結果が得られるとは限らないので、 $IPARAM=1$ としておくことによりユーザ指定の初期値を読み込んで使うことができる。次に AR と CMA を変換し、 $M+L$ 次元のパラメータベクトル AA にストアし、サブルーチン $DAVIDN$ を CALL する。 $DAVIDN$ は外部関数 $FFARMA$ によって定義される対数尤度関数を $M+L$ 次元ベクトル AA に関して最大化し、その結果を同じ AA に返す。最後に AA の逆変換を行なって、AR 係数および MA 係数の最尤推定値が得られる。

外部関数 $FFARMA$ の中では、まず $M+L$ 次元パラメータベクトル $AA(I)$ から逆変換により AR 係数と MA 係数を求める。次に $SETAB3$ と $ISTAT3$ を CALL して ARMA モデルの状態空間モデルを定義し、初期状態を設定してから $FILTR3$ を CALL して状態空間モデルの対数尤度を計算する。9 章で用いたサブルーチン $FILTER$ を用いても同じ結果が得られるが、 $FILTR3$ では行列 F, G, H が (10.6) の形をしていることを利用しているので、少ない計算量で実行できる。また、パラメータの推定のためには平滑化を行なう必要がないので、 $V_{n|n-1}$ と $V_{n-1|n-1}$ を保存するための領域をとる必要がない。

第11章 トレンドの推定

11.1 多項式トレンド推定のプログラム

プログラム 11.1 の POLREG は多項式トレンド推定のためのプログラムである。第 5 章のサブルーチン REGRES は一般の回帰モデルに対し適用できるので、多項式トレンド推定のためには、(11.3) の行列を作成するサブルーチン SETXPL を用意するだけでよい。ただし、データ数 N が主プログラムで宣言した 2 次元配列 X の第 1 添字の大きさ MJ を越える場合には、サブルーチン REDUCT は時系列を適当な長さに分割してハウスホルダー変換を繰り返し適用する。したがって SETXPL は与えられた始点 $N0$ が 0 のときには

$$X = \begin{bmatrix} 1 & x_1 & \cdots & x_1^m & y_1 \\ \vdots & \vdots & & \vdots & \vdots \\ 1 & x_{mj} & \cdots & x_{mj}^m & y_{mj} \end{bmatrix} \quad (11.1)$$

を作り、 $N0 > 0$ のときにはすでに求められている上三角行列 S の下に新しいデータを付加し

$$X = \begin{bmatrix} s_{11} & s_{12} & \cdots & s_{1,m+1} & s_{1,m+2} \\ & s_{22} & \cdots & s_{2,m+1} & s_{2,m+2} \\ & & \ddots & \vdots & \\ & & & s_{m+1,m+1} & s_{m+1,m+2} \\ & & & & s_{m+2,m+2} \\ 1 & x_{n_0+1} & \cdots & x_{n_0+1}^m & y_{n_0+1} \\ \vdots & \vdots & & \vdots & \vdots \\ 1 & x_p & \cdots & x_p^m & y_p \end{bmatrix} \quad (11.2)$$

を作るようにする必要がある。ただし、 $p = N0 + MJ - m - 2$ である。PTPOL は多項式トレンドモデルの結果を図示するためのプログラムである。

11.2 トレンド推定のプログラム

プログラム 11.2 の TREND はトレンド推定の場合の、サブルーチン FILTER と SMOOTH の利用法を示したものである。READTS で読み込まれる時系列データのほかにトレンドの次数 M とパラメータ推定の方法を指定する IOPT を入力する必要がある。トレンドモデルには τ^2 と σ^2 の 2 つのパラメータがあるが、そのうち σ^2 は 9 章の方法によって自動的に推定される。したがって、実際には τ^2 だけを推定すればよい。そのためには 4 章の数値的最適化の方法を用いてもよいが、1 次元のパラメータを 2 桁程度の精度で求めるためには本プログラムのような簡単な離散的探索の方法で十分であろう。まず、IOPT=0 とおくと $\tau^2 = 2^{-k}$ の中で最適な値が選ばれる。次に IOPT=1 とおいてプログラムを実行すると TAU20 と DELTA が読み込まれ

$$\tau^2 = \text{TAU20} + \text{DELTA} * I, \quad I = 0, \pm 1, \dots, \pm 9$$

の範囲で最適な値を求め、その値で平滑化を行なった結果が PTRND と PRTRND によって、それぞれプロッタとプリンタに出力される。

サブルーチン SETTRN では、次数 $M=1, 2, 3$ の場合に応じて (11.18) の F, G, H および観測ノイズの分散 R を定義する。ISTATE では初期状態ベクトル $x_{0|0}$ およびその分散共分散行列 $V_{0|0}$ を定める。

第12章 季節調整モデル

12.1 季節調整のプログラム

プログラム 12.1 は一変量の時系列をトレンド成分，季節成分，定常 AR 成分，曜日効果項および観測ノイズの 5 つの成分に分解するプログラムである．M1 はトレンド成分モデルの次数 k で，通常は 1 か 2 が選択される．M2 は季節成分モデルの次数 ℓ で，通常は 1 とおく．PERIOD は季節成分の 1 周期の間に観測されるデータ数 p で月次データの場合には 12，四半期データの場合には 4 とおけばよい．また，季節成分を含まないモデルを推定するためには M2=0 とおけばよい．M3 は定常 AR 成分の次数 m_3 で 0 から 10 までのうちから選択するが，M3 として大きな値を選択すると計算時間が著しく増加する．通常は M3=0 としてまず計算し，推定されたトレンドがうねって波うつ場合には M3=2 程度のモデルをためしてみるとよい．M4 は曜日効果項の係数の数である．通常は 0 としておき，曜日効果を考慮するときに M4=6 とすればよい．

次に IPARAM=0 とすると，サブルーチン SPARAM で設定された標準値を用いて自動的にパラメータ推定を行ない，求められた最尤推定値に基づく平滑化の結果を PRSEAS および PTSEAS から出力する．なお，DAVIDN は 4 章で説明した数値的最適化のサブルーチン，SETABC と ISTAT1 は季節調整のための状態空間モデルと初期状態を設定するためのサブルーチンである．また，DAVIDN のリストは付録 A に示してある．

FILTR1 と SMOTH1 は，カルマンフィルタおよび固定区間平滑化のためのサブルーチンである．プログラム 9.1 のサブルーチン FILTER および SMOOTH を用いても計算できるが，季節調整モデルの状態ベクトルは通常 12 次元以上となるので，一般の係数行列 F を想定した FILTER や SMOOTH では計算量が多くなる．そこで，FILTR1 と SMOTH1 は行列 F が (9.10) の標準形であるものと仮定して計算を行なう．これにより，各時刻あたりの計算量は $O(m^3)$ から $O(m^2)$ のオーダーに減少する．

第13章 時変係数 AR モデル

13.1 時変分散推定のプログラム

プログラム 13.1 は時系列の時変分散を推定し、分散がほぼ一定の規格化した時系列を求めるものである。この方法は厳密には白色雑音を仮定しているが、長周期成分が卓越している場合以外は、一般の時系列に適用してもよい結果を出すことが多い。必要なサブルーチンは PRVAR を除いてプログラム 11.2 と共通である。ただし、観測ノイズの分散は既知で推定する必要が無いので、 $R = \pi^2/6$ とおき、ISW=1 として FILTER を CALL すればよい。サブルーチン PRVAR では推定されたトレンドから時変分散を計算するとともに、その平方根でもとの時系列を割って規格化された時系列を生成している。

13.2 時変係数 AR モデル推定のプログラム

プログラム 13.2 は時変係数 AR モデルを推定するためのプログラムである。入力として M, K, NOBS, IOPT, NOUT が必要である。M は自己回帰の次数、K は (13.10) の差分の階数、NOBS は何点毎に AR 係数が変化するかを表す。IOPT はパラメータ τ^2 の最適値のさがし方を指定するものでプログラム 13.1 と同様である。NOUT は AR 係数が突然変化する回数で、NOUT > 0 のときにはさらに変化時点 LOUT(I), I=1, NOUT を入力する。

初期値を設定する ISTCAR と状態空間モデルを定義する SETCAR の後に FILTR2 を CALL すると、カルマンフィルタにより対数尤度を計算する。次に対数尤度を最大とする TAU2 を求めた後 FILTR2 と SMOTH1 を CALL すると、時変係数の平滑値が得られる。

PRCAR は得られた時変係数の推定値を書き出す。一方、PTCAR は時変 AR 係数から各時刻での PARCOR を計算しその時間変化をプロットする。AR 係数ではなく PARCOR を表示するのは、PARCOR が常に -1 と 1 の間の値をとるため表示に適していることと、採用したモデルの次

数の影響を受けにくいためである。

13.3 時変スペクトル表示のプログラム

時変 AR モデルでは各時刻に対応して AR 係数や PARCOR が得られるが、係数の変化だけではそれぞれのモデルがどのような特徴を持っているかを見るのは困難である。時変スペクトルをみると各時刻でどの範囲の周波数成分が卓越しているかがよくわかる。

プログラム 13.3 は時変スペクトルを計算し表示するためのプログラムである。N は推定されたモデルの個数で、プログラム 13.2 の N/NOBS に相当する。また、M は AR モデルの次数、K は差分の階数である。SIG2 と AR 係数 $AR(I, J); I=1, M; J=1, N$ を読み込んでサブルーチン PT3DSP を CALL すると、まず PLOT3A で座標を描き、次に各時刻 $NOBS * J, (J=1, \dots, N)$ に対して ARMASP でスペクトルの対数値を計算し PLOT3B で鳥瞰図を描く。

メインプログラムへの入力 IVAR を 1 としておく、時変分散 $VAR(I), I=1, \dots, N$ を読み込み、実際には $\log(VAR(I))$ を加えた値を表示する。これは、プログラム 13.1 で規格化した時系列に対して時変 AR モデルが得られている場合に、原データの時変スペクトルを計算するためである。

第14章 非ガウス型モデル

14.1 非ガウス型平滑化のプログラム

プログラム 14.1 は、サブルーチン NGSMTH を用いて非ガウス型状態空間モデルによる平滑化を行なうプログラムである。時系列 y_n を一次元配列 (Y(II), II=1, N) に与え、NOISEV, NOISEW, TAU2, SIG2などを指定する必要がある。NOISEVはシステムノイズの型を指定するパラメータで、1のときは正規分布、2のときはピアソン分布族、3のときは両側指数分布である。また、NOISEV=0のときにはユーザが作成するファンクション USERV を用いて計算を行なう。同様に、NOISEWは観測ノイズの型を指定するパラメータで、1のときは正規分布、2のときはピアソン分布族、3のときは両側指数分布、4のときは2重指数分布である。また、NOISEW=0のときにはユーザが作成するファンクション USERW を用いて計算を行なう。NOISEV や NOISEW を 2 としてピアソン分布族を指定した場合には、分布型を指定するパラメータ BV や BW も入力する必要がある。

IDIST は INITD で指定された密度関数に従う初期分布 $p_{0|0}$ を計算し、(P(I), I=1, K) にストアする。サブルーチン NORMLZ は (14.11) の規格化定数を計算し規格化を行なって、常に密度関数の全区間での積分が正確に 1 となるようにしている。また、このサブルーチンは (14.12) によって対数尤度の計算にも利用される。TRANS は FUNCT で指定された密度関数に従うシステムノイズの離散近似を行ない、(Q(I), I=-K, K) にストアする。

NGSMTH は非ガウス型平滑化を行なうサブルーチンである。NGSMTH ではまず非ガウス型フィルタリングを行なう。1期先の予測分布と平滑化分布の数値近似された値 $p_1, \dots, p_d, f_1, \dots, f_d$ が (P(I), I=1, K), (F(I), I=1, K) に求められ対数尤度の値が FF に与えられる。ただし、FILTER の場合と同様に $\text{OUTMIN} < \text{Y(II)} < \text{OUTMAX}$ を満たす時系列だけが実際の計算に利用される。P(I) と F(I) は時間の経過とともに上書きされて失われてしまうのでプログラム 9.1 の FILTER と同様にこれらの計算結果を (PS(I, II), I=1, K; II=1, N) および (FS(I, II), I=1, K; II=1, N) にスト

アする。次に、これらを利用して各時刻での平滑化分布 $(S(I), I=1, K)$ を求め $(SS(I, II), I=1, K; II=1, N)$ にストアする。メモリが不足する場合には、プログラム 9.1 の SMOOTH の場合と同様に、 $FS(I, II)$ だけをストアしておき $PS(I, II)$ は計算をやり直して求めるようにプログラムを修正することができる。さらに、 x_n のフィルタ分布 $p(x_n|y_n)$ が不要な場合には、 $(S(I), I=1, K)$ を $(SS(I, II), I=1, K; II=1, N)$ の代わりに $(FS(I, II), I=1, K; II=1, N)$ にストアしていけばさらにメモリを節約できる。 $(LOC(II), II=1, N)$ は各時刻における分点の移動量を表しており、指定した密度関数の定義域が $[t_0, t_d]$ で $LOC(II)=L$ のとき、時刻 II における実際の定義域は $[t_0 + L * \Delta x, t_d + L * \Delta x]$ となる。これにより時間とともに平均値が移動していくようなデータでも比較的狭い範囲の $[t_0, t_d]$ に対応できることになる。NGSMTH を実行するためには XMIN と DX も引数により与えておく必要がある。XMIN と DX は密度関数の離散化を指定するパラメータで XMIN は t_0 の値、また DX は Δt を表す。PRNGSM と PTNGSM は非ガウス型平滑化の結果を出力するためのサブルーチンである。

第16章 シミュレーション

16.1 シミュレーションのプログラム

プログラム 16.1 は状態空間モデル (16.6), (16.7) によるシミュレーションのためのプログラムである。サブルーチン SETSEA は季節調整のための状態空間モデルを定義する。M1, M2, M3 はそれぞれトレンド成分モデル, 季節成分モデルおよび AR モデルの次数を表す。したがって M1=M2=0 とおくことにより AR モデルによるシミュレーションを行なうこともできる。N はシミュレーションにより生成するデータの個数, NS はそのうち実際には利用しない個数で, 後半の N-NS 個だけが表示される。これは AR モデルなどではシミュレーションの結果が初期値に依存するためである。INI は一様乱数を生成するときに利用する初期値, SIG2 は観測ノイズの分散である。M1>0 の場合にはトレンド成分モデルの分散 TAU1 と初期値が必要である。初期値は M1=1 のときは t_0 を X(1) に, M1=2 のときは t_0, t_{-1} を X(1), X(2) に入力する。M2>0 のときには季節成分モデルの分散 TAU2, 1 周期の長さ PERIOD と初期値 X(M1+I), I=1, PERIOD-1 を入力する。M3>0 のときには AR モデルの分散 TAU3, AR 係数 AR(I), I=1, M3 と初期値が必要である。サブルーチン SETSEA はこれらのパラメータに従って, 状態空間モデルを特徴づける行列 F, G, H, Q, R を決定する。SIMSSM はこれらの行列で定まる状態空間モデルに対しシミュレーションを行なう。CHOLES は分散共分散行列 X をコレスキー分解し, $X = YY^t$ を満たす下三角行列 Y を二次元配列 (Y(I, J), I=1, K; J=1, K) にストアする。RGAUSS は Marsaglia のアルゴリズムにより正規乱数を生成する外部関数である。RGAUSS で用いる一様乱数発生 of 外部関数は, 現在は $a = 314159269, m = 2^{31} - 1$ とおいた乗算型合同法による一様乱数生成の関数 RUNI が用いられている。整数演算でオーバーフローが発生するためにこのサブルーチンが利用できない場合には (16.1), (16.2) の合同法を用いる RUNIF を利用すればよい。

WHITE は, コレスキー分解された分散共分散行列を用いて (16.8) により k 次元正規乱数 (白色雑音) を生成し (Y(I), I=1, K) にストアする。PTSIM はシミュレーションの結果をプロットに, PRSIM はプリンタに出力する

ためのサブルーチンである。

SIMSSM は一般の状態空間モデルに対して適用できるので SETSEA を修正することにより、いろいろなモデルのシミュレーションを行なうことができる。とくに $L > 1$ とすると多変量の時系列を生成することができる。

16.2 非ガウス型シミュレーションのプログラム

プログラム 16.2 は非ガウス型状態空間モデルのシミュレーションのためのプログラムである。M1, M2, M3, N, NS, INI はプログラム 16.1 と同様である。

NOISEW は観測ノイズの種類, WMIN と WMAX は数値近似を行なうときの定義域の下限と上限である。同様に NOISEV, VMIN, VMAX はシステムノイズの種類と定義域を表す。NOISEW, NOISEV とともに 1 のときは正規分布, 2 のときはピアソン分布族, 3 のときは 2 重指数分布となる。また, 0 とおくと, それぞれ USERW, USERV で定義される任意の密度関数を用いてシミュレーションを行なうことができる。NGSIM では以下の手続きに従って逆関数法により, 任意の密度関数に従う乱数を生成する。

1. 与えられた数 M に対して分点 $X(1), \dots, X(M)$ を適当に定める。
2. 分布関数 $F(1), \dots, F(M)$ を求める。 $0 \leq F(1) \leq F(2) \leq \dots \leq F(M) \leq 1$ が成り立つ。ただし

$$F(I) = \int_{-\infty}^{X(I)} f(t) dt$$

3. 一様乱数 U を求める。
4. 次のようにして V を求める。
 - (a) $F(I-1) < U \leq F(I)$ を満たす I をさがす。
 - (b) $V = X(I-1) + (X(I) - X(I-1)) * U - F(I-1)F(I) - F(I-1)$

ただし, 実際の計算では (1), (2) のステップはサブルーチン DISTRI で計算しておき, NGENOIS でファンクション RNG が呼ばれるたびに (3), (4) のステップだけを実行している。また, 分布がコーシー分布, 指数分布, 2 重指数分布などの場合には NOISEV や NOISEW をそれぞれ $-1, -2, -3$ とおくと逆関数法を使わず (16.13), (16.15), (16.17) により一様乱数から直接, 効率的にこれらの分布に従う乱数を求めることができる。

付録 A 非線形最適化のアルゴリズム

A.1 非線形最適化のサブルーチン

DAVIDN は準ニュートン法によって、与えられた関数を最小とするパラメータの値を求めるためのサブルーチンである。引数により FUNCT, X, N を、また共通領域 /ccc/ により IPR と NDIF を受け渡す必要がある。FUNCT は最小化する関数 $f(x)$ を計算するサブルーチン名, X はパラメータを入れたベクトル, N はその次元である。IPR は出力の量を制御するパラメータ, NDIF は FUNCT により関数の微分 $g(x)$ を計算できる場合には 0 とおく。NDIF=1 または 2 の場合には DAVIDN はサブルーチン FUNCND を CALL して数値差分によって $g(x)$ を求めるので FUNCT は $f(x)$ の値だけを計算すればよい。NDIF=1 とすると片側差分による近似, NDIF=2 とすると両側差分による近似を行なう。サブルーチン DAVIDN の利用例はプログラム 4.3 に与えられている。