

統計学・経営科学分析システム U A S M 1

— 基本システム —

福井正康・佐藤真司・田口賢士

1 はじめに

現在、多くの統計関係のソフトウェアが世に出ているが、その中にはSAS¹⁾やSPSS²⁾等のように複数のプラットフォーム上で動作し、高い評価を得ているシステムもある。また、他方ではパーソナルコンピュータ専用で作成されたコンパクトなシステムで、高度なインターフェイスを備えたものもある。一方、経営科学にも様々な分析手法があり多くのソフトウェアが作成されているが、統計解析のシステムほど巨大で整備されたものは存在しない。その理由としてはこれらの分析手法が比較的新しいものであること、また分析手法が各分野で独立しており、統一的な体系が完全には出来上がっていないこと等が考えられる。著者らはこれらの分析手法の体系化についてソフトウェアの分野から考察する。則ち、形式的な分析手法の分類でなく、実際に動作するプログラムを通して、統計解析まで含めた種々の分析手法のデータ構造や表示形式を分類し、出来る限り統一されたインターフェイスで分析を実行するシステムの構築を試みる。この実験的なシステムを以後 U A S M (Unified Analysing System for Management) と呼ぶ。

著者らは統計解析と経営科学いずれにも利用出来るプログラムの基盤（基本システム）を構築し、その環境下で各種の分析プログラムを作成する。データ形式等について、現在確定された指針がない以上、基本システムは取り上げる

分析手法と共に段階的に改良されると考えられるので、柔軟性を重視してなるべく単純で軽いものが望ましい。しかし実際の使用にも耐えなければならないため、ある程度のインターフェイスを備えたものでなければならない。そのためにはプログラムで共通化出来る部分は出来る限り共通化し、ユーティリティとして多くの分析プログラムで活用する。さらに分析プログラムは単独のプログラムとしても実行可能にし、簡単な方法でシステムに加えることが出来るようにする。システムは第1段階から徐々に拡張して行くが、最初の段階は汎用的な統計プログラムに近い形式から出発し、その限界を見極める。

分析プログラムを作成して行く過程で、統計解析についてはなるべく既存のシステムも利用出来るようにしておく事が経済的である。ここで著者らが特に参考にした既存のシステムは統計パッケージHALBAU³⁾である。システム開発の第1段階において、システムが利用するデータはフィールドとレコードによるCSV形式の2次元テーブルとし、HALBAUのデータがそのまま利用出来るようにした。またこの形式は通常のエディターで加工がし易く、他の多くの統計解析パッケージでも利用可能である。

この論文は第1段階の基本システムについて解説している。則ち、まず2章で基本システムの概略について述べ、次に3～6章でその外部仕様と操作方法等に触れる。その後、7章でプログラムファイルの内容について紹介し、8章で代表的なデータ構造について解説する。分析プログラムについては以後の論文で述べる。

最後にこの論文で使用されている用語について注意する。プログラムの実行画面では、上述の“フィールド”が“変数”と表示される。これはデータベース等に不慣れた利用者に対する配慮と、画面上の字数の制限から決められた事である。多少実行画面とは異なるが、この論文中では説明には一貫して“フィールド”を用いる。これはプログラムで使用される“変数”と混同しないためである。また、説明の都合上ファイル名や変数名が実際のプログラムとは若干異

なっている場合があるが、これは単に文字列上の問題で本質的なものではない。

プログラムはPC-9800シリーズ上のTurbo Cで記述されており、この論文で解説される部分のソースコードは現在の段階でコメントも含めて約220KB程度である。

2 システムの概略

このシステムはメインメニューから子プロセスとして各種の分析・ユーティリティプログラムを起動させるもので、分析プログラムの増加に対して柔軟に対応出来る。メインメニューの項目は現在のバージョンでは図2-1に示す通りである。この論文で紹介する基本システムに含まれるものは、これらの項目のうちデータエディターとデータユーティリティ及び、最後の環境変更である。その他は具体的な分析プログラムの集合で、これから後に作成して行く予定のプログラム群である。この段階では、統計解析は検定を主とした基本統計と多変量解析に分け、それに現在検討中のファジィ統計を加えた。また経営科学の分野はある程度分析の種類が揃ってから細分することにした。各メニューの内容を簡単にまとめて以下に示す。

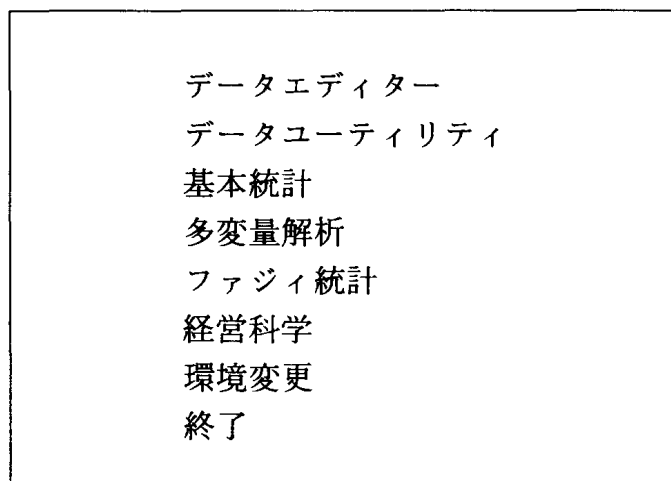


図2-1 メインメニュー

【データエディター】

このメニューはデータの入力・修正に使用されるもので、利用者はこれを選択することによって、画面上に表示される枠線の中で、各レコード毎にデータを入力したり、以前入力したデータを参照・修正する事が出来る。このメニューでは他にフィールドやレコードの消去・挿入・追加、数式を利用したデータの加工等が出来る。詳細については後述する。以後このプログラムをデータエディターと呼ぶ。後のデータ形式の拡張によって修正される可能性が高く、変更にも最も労力を要するものとなろう。

【データユーティリティ】

これはデータファイルのレコード数やフィールド数を変更するもので、あるデータファイルから他のファイルへの指定データの抽出、2つのデータファイル間でのレコードまたはフィールドの連結等が出来る。

【基本統計】

これには基本統計量の算出、度数分布、各種検定、相関分析、回帰分析等が入る予定で、殆どの市販統計パッケージの中に入っている内容である。出来るだけ多くの検定・分析についてプログラムを作成して行かなければならないが、特にノンパラメトリック検定に重点を置いて項目を設定する予定である。

【多変量解析】

これも基本統計と同様に重回帰分析、主成分分析、判別分析、クラスター分析、数量化理論等と出来るだけ多くの分析を取り扱う。これらも多くの市販統計パッケージの中に入っている分析である。

【ファジィ統計】

近年ファジィ理論の応用が盛んで統計学の分野にもその考え方が導入されている。これを著者らのプログラムに取り入れたものがファジィ統計のメニューである。しかしファジィ集合の取り扱い方には種々の方法があり、同じ分析についてさえも一意的に決められない。ここでは混乱しない程度になるべく多くの取り扱い方法を導入する。これはこのシステムで重点を置いて作成する予定のプログラム群である。

【経営科学】

このメニューでは経営科学で用いられる様々な分析方法を意欲的に取り入れて行く。但し、これらの分析ではフィールドとレコードといった統計学でよく利用されるデータ形式で表現しにくいものもあり、いかにしてデータや表示の形式を統一化するかという問題が考察の中心となる。これはこのシステムで特に重点を置いて作成する予定のプログラム群である。

【環境変更】

最後の環境変更はこのシステムの起動環境を設定するメニューである。選択出来る項目は、データファイルのディレクトリ、ワークファイルの使用ディレクトリ、欠損値の表示記号、カラーかモノクロかの起動モードである。このシステムでは欠損値の表示記号は変更可能である。デフォルトは“*”記号になっている。このシステムでは大きなデータについてはワークファイルを利用して編集・計算等を行うが、ワークファイルのドライブは十分空きがあり常に使用可能なハードディスク等に設定しておく必要がある。ここで決定されたデータは環境設定ファイルに記憶される。

3 データエディター

3-1) データファイル

このシステムのデータエディターを利用してデータファイルを作成すると、図3-1の形式のファイルが作成される。ここに“↓”記号は改行復帰を表わす。データファイルの先頭にレコード数とフィールド数が記入されている。次に各フィールドの名称が記入されている。フィールド名を設定していない場合には“*”記号が入っている。フィールド名は10バイトまでの文字列を設定することが出来る。データとしては数字だけでなく12バイトまでの文字列も利用出来る。利用者はこのデータファイルを一般のテキストエディターで参照することが出来る。このCSV形式は統計パッケージHALBAUのファイル形式に準拠しているが、HALBAUでは欠損値をデフォルトで“-99”の数字で表しているのに対して、このシステムでは“*”記号で表している。しかし、この値は環境変更のメニューを利用して変更することが出来る。

```

40,6↓
ID,身長,体重,胸囲,座高,*↓
1101001,170,56,85,85,2↓
1101003,166,55,83,89,1↓
1101010,176,65,90,92,1↓
1101008,169,72,93,93,2↓
. . . . .

```

図3-1 ファイルの構造

3-2) データエディターの構成

データエディターでのデータ編集画面の例を図3-2に示す。表示は1レコード毎であるが、1画面には最大80フィールドのデータが表示される。それ以外

データ入力		変数: ID	Record: 10	Total: 26			
001	1001	021	4	041	1	061	5
002	0	022	3	042	5	062	3
003	171	023	3	043	3	063	2
004	60	024	1	044	2	064	3
005	87	025	1	045	3	065	4
006	90	026	1	046	4	066	3
007	3	027	1	047	3	067	2
008	2	028	2	048	2	068	3
009	1	029	3	049	4	069	2
010	4	030	2	050	3	070	4
011	3	031	2	051	2	071	3
012	3	032	3	052	1	072	2
013	2	033	4	053	4	073	1
014	3	034	3	054	3	074	3
015	1	035	2	055	4	075	4
016	3	036	4	056	5	076	4
017	4	037	2	057	3	077	5
018	2	038	1	058	2	078	5
019	1	039	5	059	4	079	3
020	3	040	2	060	1	080	2

F1:メニュー1 F2:メニュー2 F3:メニュー3 ファイル[A:¥TC¥FLKUI¥RONBUN.DAT
F6:変数参照 F7:先頭リポート F8:最終リポート F9:前リポート F10:次リポート]

図 3 - 2 データ入力画面

データ入力		変数: ID	Record: 10	Total: 26			
001	1001	021	4	041	1		
002	0	022	3	042	5		
003	171	023	3	043	3		
004	60	024	1	044	2		
005	87	025	1	045	3		
006	90	026	1	046	4		
007	3	027	1	047	3		
008	2	028	2	048	2		
009	1	029	3	049	4		
010	4	030	2	050	3		
011	3	031	2	051	2		
012	3	032	3	052	1		
013	2	033	4	053	4		
014	3	034	3	054	3		
015	1	035	2	055	4		
016	3	036	4	056	5		
017	4	037	2	057	3		
018	2	038	1	058	2		
019	1	039	5	059	4		
020	3	040	2	060	1	080	2

変数参照

001: 年齢

002: 性別

003: 身長

004: 体重

005: 胸囲

006: 座高

007:

008:

009:

010:

011:

012:

013:

014:

015:

F1:メニュー1 F2:メニュー2 F3:メニュー3 ファイル[A:¥TC¥FLKUI¥RONBUN.DAT
F6:変数参照 F7:先頭リポート F8:最終リポート F9:前リポート F10:次リポート]

図 3 - 3 フィールド名の参照画面

のデータは矢印キーを利用して画面を横にスクロールさせ表示することが出来る。カーソルが次のデータに移動する毎に上部の表示バーの“変数”の部分にそのデータのフィールド名が表示される。まとめてフィールド名を参照したい場合、フィールド名の一覧はF・6キーを押すことによって図3-3のように表示される。但し、ウィンドウ内の空欄の部分はまだフィールド名が定義されていない部分である。ウィンドウ内の表示内容は矢印キーやROLLUP, ROLLDOWNキーでスクロールさせることが出来る。また画面に表示されているレコードはF・9キーとF・10キーで前後に変更することが出来る。

データエディターのメニューはF・1～F・3キーを押下することによってメニューウィンドウが開き、表示される。それぞれF・1キーはファイル、F・2キーはレコード、F・3キーはフィールドに関する操作を実行するメニューである。メニューに関する詳細は次節で述べる。

3-3) データエディターのメニュー

データエディターの入力画面で利用可能なメニューはファイルメニュー、レコードメニュー、フィールドメニューである。それぞれのメニューには図3-4で示される項目がある。

最初にファイルメニューから説明する。ファイルの読込みはディスクのデータファイルをデータエディターで編集出来るようにメモリに読み込む操作で、まず環境記憶ファイルで指定されたデータディレクトリからファイルを選択す

ファイルメニュー	レコードメニュー	フィールドメニュー
ファイル読込み	レコードジャンプ	フィールド挿入
編集終了・保存	レコード挿入	フィールド追加
編集続行・保存	レコード追加	フィールド削除
編集終了・破棄	レコード削除	フィールド名変更
		数式入力

図3-4 エディターメニュー

る。図3-5にファイルの指定画面の例を示す。利用者は直接ファイル名を入力してもよいし、画面に表示されるファイル名の一覧から選択してもよい。編集終了・保存は編集集中のデータをディスクへ保存した後に編集を終了するもので、ファイル名は保存の際に図3-5と同様の画面で指定する。編集続行・保存は編集を終了しないでデータをディスクに保存する。編集終了・破棄はデータを保存しないで、編集を終了するものである。

データ入力		変数:	Record:	Total:
<div style="border: 1px solid black; padding: 5px;"> ファイル選択 A: ¥TC¥FUKUI¥ </div>				
..¥	STAT.MAK	FUZSU2.OBJ	SHIKI.C	
GBIOS.H	GUTIL.C	EDFILE.MAK	EDFILE.C	
TEST3.DAT	TEST4.DAT	TEST2.DAT	TEST.C	
TEST6.DAT	TABAI.C	EDFILE.OBJ	MUTIL.C	
ISM.OBJ	BSTAT.MAK	UTIL2.BAK	UTIL.C	
FUZ1B.DAT	FUZSU1.EXE	TEST7.DAT	CORR.OBJ	
UTIL.BAK	STUTIL.MAK	STAT.INI	CORR.C	
CORR.MAK	HEAP.C	GTEST.C	TCCG.BAT	
GRAPHICS.LIB	FUZ3D.C	FUZ2.DAT	LCD.C	
FUZSU1.MAK	FUZSU1.MAP	STAT.OBJ	UTIL.OBJ	
F1:メニュー-1 F2:メニュー-2 F3:メニュー-3 ファイル[] F6:変数参照 F7:先頭レコード F8:最終レコード F9:前レコード F10:次レコード				

図3-5 ファイル選択画面

次にレコードメニューであるが、レコードジャンプは編集したいレコードまでジャンプする機能で、先頭レコード、最終レコード、レコード番号指定を選択してジャンプ出来る。レコード挿入は指定レコードの前に1レコード挿入するものであり、レコード追加は最終レコードの次に1レコード追加するものである。レコード削除は現在表示されているレコードを削除するものである。

最後にフィールドメニューであるが、フィールド挿入、フィールド追加、フィールド削除はレコードの場合と同様である。ただし、挿入と削除の位置はフィー

データ入力		変数: ID	Record: 10 Total: 26
001	1001	021	4
002	0	022	3
003	171	023	3
004	60	024	1
005	87	025	1
006	90	026	1
007	3	027	1
008	2	028	2
009	1	029	3
010	4	030	2
011	3	031	2
012	3	032	3
013	2	033	4
014	3	034	3
015	1	035	2
016	3	036	4
017	4	037	2
018	2	038	1
019	1	039	5
020	3	040	2
		041	1
		042	5
		043	3
		044	2
		045	3
		046	4
		047	3
		048	2
		049	4
		050	3
		051	2
		052	1
		053	4
		054	3
		055	4
		056	5
		057	3
		058	2
		059	4
		060	1
		080	2

変数名変更

001: ID
 002: 性別
 003: 身長
 004: 体重
 005: 胸囲
 006: 座高
 007:
 008:
 009:
 010:
 011:
 012:
 013:
 014:
 015:

F1:メニュー-1 F2:メニュー-2 F3:メニュー-3 ファイル[A:¥TC¥FUKUI¥RONBUN.DAT
 F6:変数参照 F7:先頭レコード F8:最終レコード F9:前レコード F10:次レコード]

図 3 - 6 フィールド名変更画面

フィールド番号とフィールド名を表示したウィンドウが開かれるので、その中から選択する。フィールド名変更は各々のフィールドに名前を付ける機能である。これもフィールド番号を表示したウィンドウが開かれて、その中で名前を入力・変更する。図 3 - 6 にフィールド名変更の画面例を示す。

数式入力はいこれまでのデータを数式を用いて加工し、新しいデータを作成するものである。数式の入力は図 3 - 7 に示す数式入力用ウィンドウの中で行われる。数式入力で使用される書式は以下の規則に従っている。使用可能な算術演算子は (), ^, *, /, +, - であり、その意味は BASIC と同様である。優先順位は 1: () 2: ^ 3: *, / 4: +, - である。各レコード毎の特定フィールドのデータを利用する場合は @24 のように @ の後にフィールド番号を付けて表わす。図 3 - 7 の数式入力例をフィールド名を用いて表わすと、(身長-100)×0.9 を計算していることになる。数式の中で使用出来る関数とその機能は図 3 - 8 にまとめる。

データ入力		変数: ID		Record: 10 Total: 26	
001	1001	021	4	041	1
002	0	022	3	042	5
003	171	023	3	043	3
004	60	024	1	044	2
005	87	025	1	045	3
006	90	026	1	046	4
007	3	027	1	047	3
008	2	028	2	048	2
009	1	029	3	049	4
010	4	030	2	050	3
0					
0					
0					
0					
0					
016	3	036	4	056	5
017	4	037	2	057	3
018	2	038	1	058	2
019	1	039	5	059	4
020	3	040	2	060	1
				080	2

数式変数選択

086:
087:
088:
089:
090:
091:
092:

数式入力: 変数[0番号] 関数[ABS, SQRT, EXP, LOG, LN, SIN, COS, TAN, ATAN]

(03-100)*0.9

098:
099:
100:

F1: 上-1 F2: 上-2 F3: 上-3 ファイル[A: ¥TC¥FLKUI¥RONBLN.DAT]
F6: 変数参照 F7: 先頭リット F8: 最終リット F9: 前リット F10: 次リット

図 3 - 7 数式入力画面

関 数	機 能
ABS(X)	X の絶対値
SQRT(X)	X の平方根
EXP(X)	e^x : e は自然対数の底
LOG(X)	$\log_{10} X$: 常用対数
LN(X)	$\log_e X$: 自然対数
SIN(X)	sin X
COS(X)	cos X
TAN(X)	tan X
ATAN(X)	$\tan^{-1} X$: tan X の逆関数

図 3 - 8 利用可能な関数

4 データユーティリティ

図2-1に示されるメインメニューのデータユーティリティはデータファイルを加工して利用し易い新しいファイルを作成するものである。これにはさらにサブメニューがあり、その項目は図4-1に示す通りである。

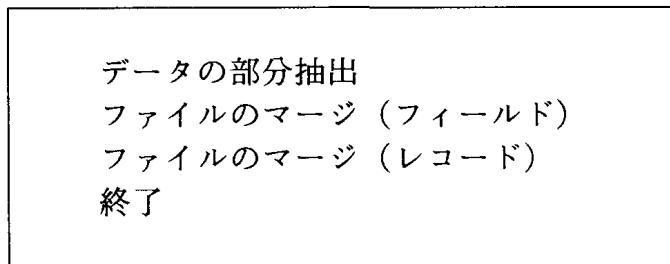


図4-1 データユーティリティメニュー

データの部分抽出は、データ中の指定されたフィールドとレコードについてのデータをファイルに出力するもので、必要なデータについてのみ抽出して、利用し易いファイルを作成するものである。フィールドに関する指定は図4-2に示すような画面を利用して行われる。フィールドの選択順は自由に設定出来る。選択にはカーソルを選択したいフィールドに重ね、リターンキーを押下する。選択の解除は選択されたフィールドで再度リターンキーを押下する。

レコードはあるフィールドについてデータの範囲を指定し、その範囲に入るデータを持つレコードだけ抽出するという方法で選択される。この範囲を指定するフィールドを制御フィールドと呼ぶことにする。制御フィールドの選択と範囲の設定用の画面例は図4-3に示す。制御フィールドの選択は図4-3の上半分の部分で行われる。選択の方法はフィールド選択の方法と同様である。制御フィールドが選択されると画面下の部分にそのフィールド番号とフィールド名が表示される。カーソル位置をF5キーによって画面の上から下に切り換えて、選択されたフィールドの番号の下に指定したい範囲を入力する。この例

データユーティリティ		出力変数指定	
001	ID	1 021	7 041
002	性別	2 022	8 042
003	身長	3 023	9 043
004	体重	4 024	10 044
005	胸囲	5 025	11 045
006	半高	6 026	12 046
007		027	13 047
008		028	14 048
009		029	15 049
010		030	16 050
011		031	17 051
012		032	18 052
013		033	19 053
014		034	20 054
015		035	21 055
016		036	22 056
017		037	23 057
018		038	24 058
019		039	25 059
020		040	26 060
			27 061
			28 062
			29 063
			30 064
			31 065
			32 066
			33 067
			34 068
			35 069
			36 070
			071
			072
			073
			074
			075
			076
			077
			078
			079
			080

ファイル[A:¥TC¥FUKUI¥RONBUN.DAT]
 分析に使用する変数を選択して下さい RET:選択・解除

図 4 - 2 フィールド選択画面

データユーティリティ		出力制御変数指定		AND/OR MODE : AND
001	ID	011	2 021	031
002	性別	1 012	022	032
003	身長	013	023	033
004	体重	014	024	034
005	胸囲	015	025	035
006	半高	016	026	036
007		017	027	037
008		018	028	038
009		019	029	039
010		020	030	040

制御変数	2	2 性別	11		
区分	1 0	-0	1 -2	-	-
			4 -5	-	-
				-	-
				-	-
				-	-
				-	-
				-	-
				-	-
				-	-
				-	-

F5:上下切換 F1:AND/OR 切換 ファイル[A:¥TC¥FUKUI¥RONBUN.DAT]
 制御に使う変数とその範囲を指定して下さい RET:選択・解除

図 4 - 3 制御フィールドの選択・範囲指定画面

ではフィールド2（フィールド名：性別）が0であり、かつフィールド11（フィールド名：未設定）は1~2または4~5であるデータのみを抽出する。2つのフィールドを選択した場合、それぞれの範囲の和集合を取るか積集合を取るかの選択は、データの部分抽出に限り、利用者がF1キーで設定出来る。図4-3の右上の“AND/OR MODE”の部分にどちらのモードが選択されているか表示されている。

プログラムの実行順序は最初に入力・出力ファイル名を尋ね、次に抽出するフィールドを選択画面から選択する。その後に制御フィールドを選択し、それぞれの制御フィールドの取る値の範囲を指定する。最後にこれらの指定に適合するデータだけがファイルに出力される。図4-4はフィールド・制御フィールド選択画面の中で利用出来るメニュー項目である。最初の実行次段階は、利用者がこれを選べばほぼ間違いなく通常の処理は出来るという簡易メニューである。図4-4中のメニューにはフィールド選択と制御フィールド選択の2種類ある。これらのメニューは、現在どちらの画面であっても再度他の選択画面に移ることが出来る機能である。これは選択を誤った場合の訂正等に利用される。実行中止は今までの処理を中止して最初のメニュー画面に戻るものである。最後の実行開始は選択されたデータのファイル出力を開始するものである。

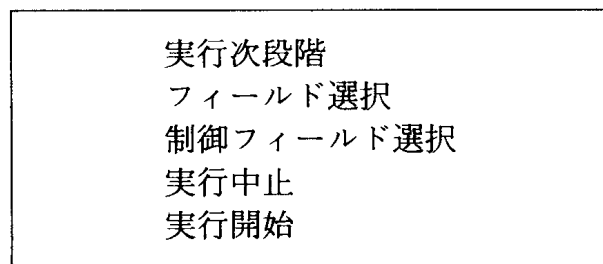


図4-4 フィールド選択メニュー

データユーティリティの図4-1におけるファイルのマージはフィールドについてのマージとレコードについてのマージの2種類ある。フィールドについ

でのマージは、レコードが同一でフィールドの異なるデータが格納されている2つのファイルを1つにまとめるものである。また、同様にレコードのマージは同一のフィールドに対する異なったレコードの格納された2つのファイルを1つにまとめるものである。いずれも最初に2つの入力ファイル名と1つの出力ファイル名を指定して実行する。

5 環境変更

これはプログラムの実行環境を決定するためのもので、現在のバージョンではデータディレクトリ・ワークディレクトリ・欠損値・起動モードが指定出来る。このシステムのデータファイルの選択は、図3-5のようにファイル名を直接入力しても、またファイル名の一覧から選択してもよい。その際、最初に表示されるディレクトリがデータディレクトリである。もちろん実行中にデータのディレクトリを変更することは容易である。これに対してワークディレクトリは編集や計算で使用するワークファイルを作成する場所を決めるもので、ワークファイル使用中はそのディスクは常に使用状態にしておかなければならないため、ハードディスク等に設定し、十分な空き容量があることが望ましい。

欠損値はデフォルトでは“*”記号になっているが12文字までの任意の文字列を設定出来る。起動モードはプログラムの利用される環境がカラーであるか、モノクロであるか指定する。モノクロの場合画面の配色や表示がカラーの場合と多少異なる。例として設定が終了した段階での画面を図5-1に示す。

設定された環境は右上メニューで変更終了を選択することにより、システム終了まで有効にすることが出来る。システムは起動時に環境設定ファイルを読んで立ち上がるが、メニューの変更保存で環境設定ファイルにデータを保存して、次回からそのデータを使用してシステムを立ち上げることが出来る。また、起動時に環境設定ファイルが見つからない場合はシステムのデフォルトの値つ

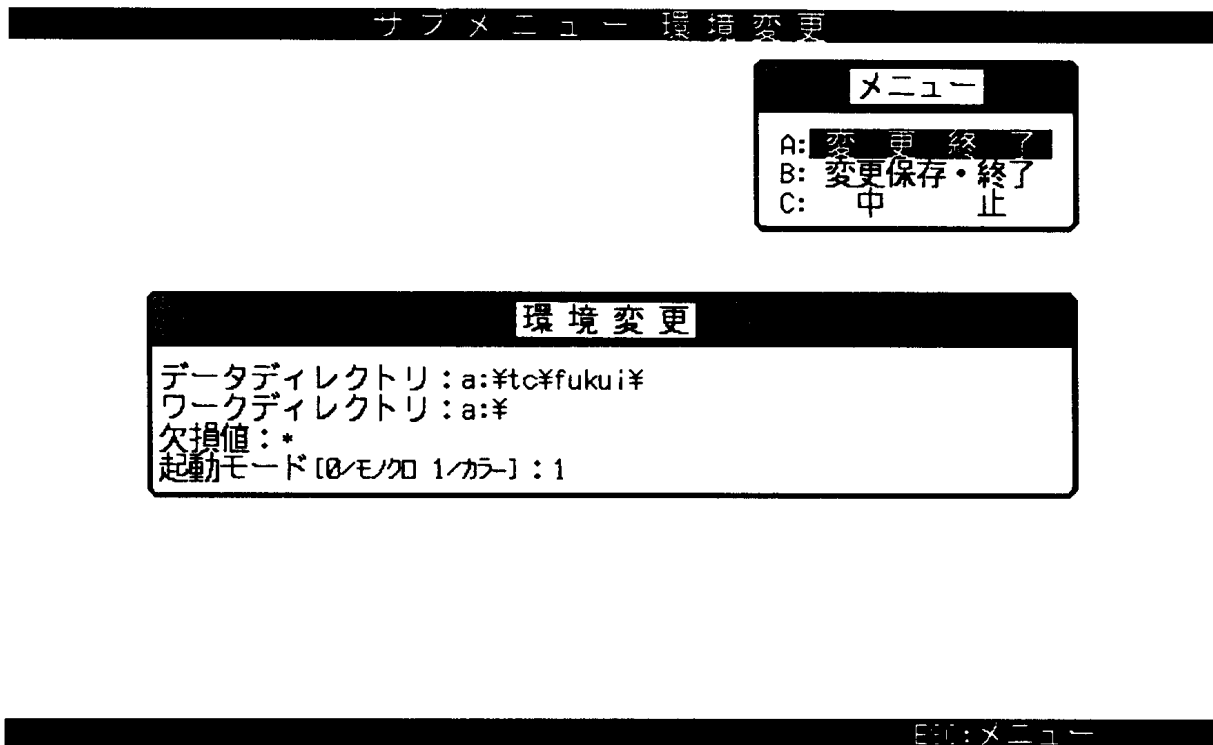


図5-1 環境変更画面

まり、データディレクトリとワークディレクトリはカレントドライブ、欠損値は“*”記号、起動モードはカラーになる。各分析プログラムは統合環境の中からでも、単独にでも起動可能であるが、単独に起動させた場合でもこの環境設定ファイルを参照する。ただし、この場合は環境の変更は出来ない。

6 分析プログラム

分析用のプログラムは図6-1に示すような統一的手法で処理される。まずデータファイルを選択し、その中で使用するフィールドを選択する。ただし、分析の種類によってフィールド選択の方法は変化する。例えば相関係数の検定では相関行列に使用するフィールドを一度に指定するが、回帰分析では目的フィールドと説明フィールドを順番に指定する。これはメニュー1の中で実行次段階

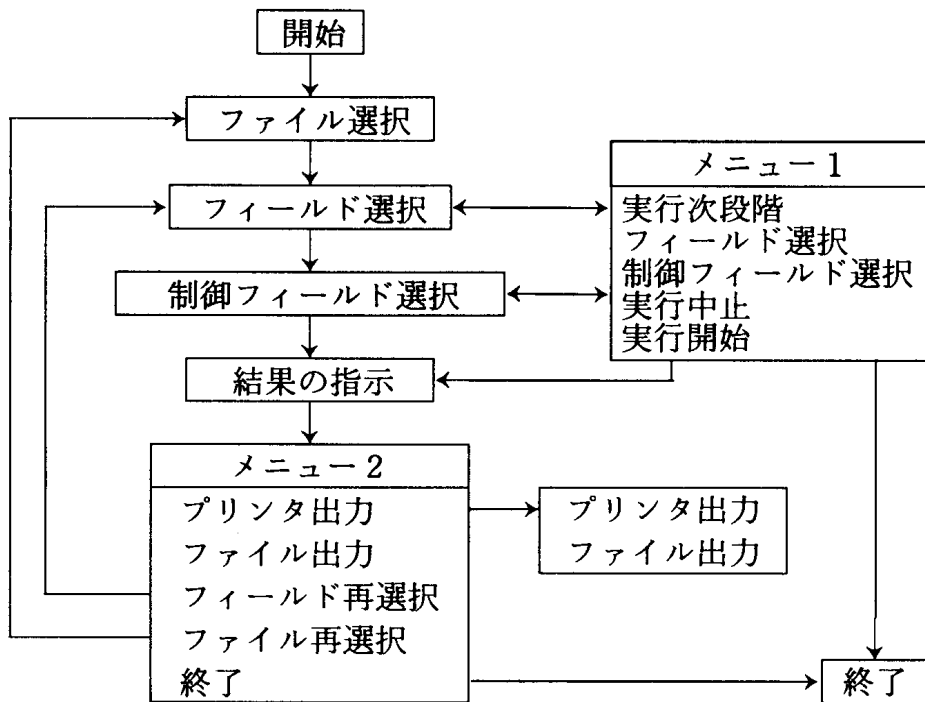


図6-1 分析プログラムの実行手順

を選択することによって順番に選択出来るようになっている。フィールド選択の後に制御フィールドの選択を行う。これは4章データユーティリティのデータの部分抽出の所でも述べたように、分析に使用するレコードをある制御フィールドの値の範囲によって設定するものである。まず、制御フィールドを選択し、データとして使用する範囲を指定するが、この操作は図4-3の画面とほぼ同じ選択・指定画面によって行われる。制御フィールドの選択可能な数は、分析の種類によって異なるが最大4フィールドまでで、個々のフィールドの値の範囲や分割数は自由に設定出来る。

範囲設定の方法を図6-2に示す例を用いて説明する。ここに示すように、フィールド1に3個、フィールド2に1個、フィールド3に2個の範囲が設定してある場合、選択されるレコードは以下の条件を満たすものである。

フィールド 1 (f1)	フィールド 2 (f2)	フィールド 3 (f3)
範囲1-1	範囲2-1	範囲3-1
範囲1-2		範囲3-2
範囲1-3		

図 6-2 制御フィールドの値指定

$\{(f1 \in \text{範囲1-1}) \text{ OR } (f1 \in \text{範囲1-2}) \text{ OR } (f1 \in \text{範囲1-3})\}$

$\text{AND } (f2 \in \text{範囲2-1}) \text{ AND } \{(f3 \in \text{範囲3-1}) \text{ OR } (f3 \in \text{範囲3-2})\}$.

ここに $f1 \in \text{範囲1-1}$ はフィールド 1 のデータが範囲1-1に含まれることを意味する。

欠損値があった場合、分析の方法や利用者の考え方に応じてその取扱方法を変えることが出来る。1つは欠損値を含むレコードを全て除去する方法で、他の1つは欠損値だけを除去する方法である。この指定はフィールドまたは制御フィールドの選択画面の中でF1キーを押すことにより変更されるが、分析の種類によっては1種類の除去方法しか選べないものもある。フィールド選択と制御フィールド選択の2つの画面についての共通のメニューが図6-1の右に書かれたメニュー1である。フィールド選択画面から制御フィールド選択画面へ移ることも、逆に制御フィールド選択画面からフィールド選択画面へ戻ることも可能である。操作に慣れない利用者は分析次段階を選択することによって無難に結果を得ることが出来る。

フィールド・制御フィールドの選択が終って分析が開始されたら、しばらくして結果が表示される。これはスクロール可能な形式で画面上に表示される。出力結果が多量になるときはワークファイルが利用される。この表示段階で利用出来るメニューが、図6-1の下に書かれたメニュー2である。この段階から結果をプリンタやファイルへ出力出来る。これらはいずれも画面に表示されたものをそのまま出力するもので、プリンタ出力で書式設定は行っていない。

使用するフィールドを変えて再度分析をやり直したい場合や、データファイ

ルを変えてもう1度やり直したい場合は、メニュー2のフィールド再選択やファイル再選択を選び再度分析を行う。

7 プログラムファイルの構成

このシステムのプログラム上の構成は図7-1のように大きく分けて3つの部分に別れる。1つは全てのプログラムから共通して利用されるユーティリティプログラム、次は全ての分析プログラムを子プロセスとして実行するための起動プログラム、最後は編集や各種の分析などを実行する実行プログラムである。

まず、共通に利用されるユーティリティプログラムから説明する。ファイルUTIL1.Cは主としてハードウェアを制御する関数^{4),5)}が多く含まれている。その1つがテキストVRAMの直接制御である。即ちウィンドウを開く際に利用するテキスト画面の退避・復活、メニュー選択時に文字列の反転等で利用するVRAMアトリビュートの変更、エディター等に利用するテキスト画面のスクロール、文字列の高速表示等、VRAMの直接制御を利用することで表示の高速化を図っている。次の機能がキー入力の制御である。ここではMS-DOSのキーコードの再定義機能を利用してファンクションキーや矢印キーのコードを変更し、プログラムから利用し易くしている。もちろんプログラム終了時には元の

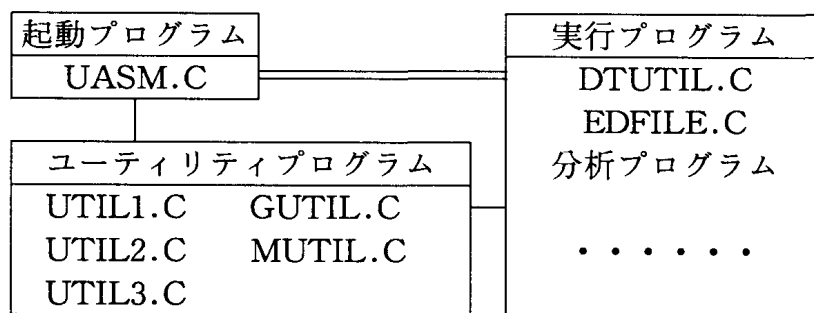


図7-1 プログラムファイルの関連

キーコードに戻しておく。画面のスクロール等には応答性の悪さを解消するために、キーボードバッファをクリアする機能も必要である。またフロッピーディスク等の入っていない場合に独自の方法で注意を促すためのハードウェアエラー処理、キーボードからの文字列入力時の簡便化のための文字列入力関数等が含まれている。

ファイルUTIL2.Cはもう少し水準が上がる。このファイルにはディレクトリとファイルの選択についての関数や、メニュー等のウィンドウを表示するための関数が含まれる。これらの関数を利用して図3-5のようなファイル選択画面を表示する関数、メッセージやエラーを表示するウィンドウやメニューを選択するウィンドウを開く関数を作成している。またデータファイルやワークファイルのデータを処理するための関数も含まれている。則ちフロッピー等が入っていないときの処理を含めた高・低水準のファイルオープン関数、区切り記号“,”で区切られたファイルからのデータの読み込み関数等である。

ファイルUTIL3.Cではさらに水準が上がり、フィールド選択画面と制御フィールド選択画面を作成する。これらの画面は全ての分析プログラムで利用される予定であるので利用頻度は非常に高い。これ以外にこのファイルでは結果表示用の関数も作成されている。結果の表示とスクロール、ワークファイルの利用、メニュー2の表示まで、分析プログラム側からは表示したい結果をメモリに記憶させ、1つの関数を呼ぶだけでよい。この部分は全体のプログラムの中でも、最も統一的に、利用し易く作られた部分である。

このシステムではグラフィック機能を用いてメニュー画面や選択画面を作成しているが、グラフィックライブラリはTurbo C標準のものを使用している。ただし、他のBASICに類似のライブラリとの互換性を高めるためにファイルGUTIL.CでTurbo C標準のグラフィックライブラリを利用してBASICライクな関数を作成している。これは多少実行時間に負荷がかかるが、BASICのコマンドに慣れたプログラマーには利用し易い。

共通に使用する関数ライブラリの最後はファイル MUTIL.C に含まれる数学関数である。現在のバージョンでは統計学で頻繁に使用される、行列式、逆行列、行列の固有値・固有ベクトル⁷⁾、また正規分布、 χ^2 分布、F分布、t分布の確率変数の値と上側確率等を求める関数⁸⁾が含まれている。さらに様々な分析が増えるに従って共通的に使用される数学関数は増えてゆく予定である。

次に起動プログラムについて述べる。ファイル UASM.C はこの基本システムの動作の主要部分でメインメニューから他の分析プログラムを子プロセスとして呼び出す役割をする。また環境の変更や登録もこのプログラム内で行われる。その際環境が変更された場合は各分析プログラムに変更内容を伝え、分析プログラム内で新しいデータディレクトリ、ワークディレクトリ、欠損値記号等が使用出来るようにする。このプログラムは全体を統括するものであるが、個々の分析プログラムはこのプログラム無しでも動作する。その際プログラムの起動環境は環境設定ファイルに従い、変更は行えない。

最後に実行プログラムに含まれるファイルについて説明する。これらのプログラムの中には各種の分析プログラムの他に、基本システムに含まれるデータエディター用のファイル EDFILE.C とデータユーティリティ用のファイル DTUTIL.C も含まれる。データエディターのプログラムではデータ構造がシステムの開発の容易さやスピードに影響する。著者らの現在のプログラムではメモリに負荷がかかり過ぎる傾向があるため今後検討しなければならない。またこのプログラムでは、大量データの処理はワークファイルを利用して行うが、ワークファイルの構造、レコードやフィールドの挿入・削除でのワークファイルの使用法等議論すべき点は多い。データエディターのメニューは大きく分けてファイルメニュー、レコードメニュー、フィールドメニューになっているが、これらの処理はこのプログラム内で行っている。

最後にファイル DTUTIL.C はデータユーティリティに関するプログラムである。データの部分抽出にファイル UTIL2.C で作成したファイル選択の関数

やファイル UTIL3.C で作成したフィールドや制御フィールド選択の関数を利用している。プログラム自体は特に複雑なものではない。

以上の8つのファイルにより基本システムは構成されるが、このシステムの主役は分析プログラムである。共通に利用される関数をいかに有効に活用して分析プログラムを作成するかがこれからの課題である。

8 データとアルゴリズム

このプログラムでは多くの構造体を定義しているが、プログラムの中で重要な役割を担う構造体について解説する。まず、分析プログラムで使用するフィールドや制御フィールドの選択方法について説明する。各フィールドには以下の型の構造体変数が対応する。

```
struct vtype {
    int sel;
    int con;
    char var[11];
};
```

この構造体変数はファイルに含まれるフィールドの数だけ、入力時に配列として動的に確保され、配列番号は(フィールド番号-1)と付与される。変数 sel はフィールド選択の状態を表わし、初期値として-1を取って、選択された順番に0から番号が付けられる。但し、フィールドの選択が回帰分析での目的変数と説明変数のように2種類以上ある場合は最初のフィールドの選択で0~998、次の選択で1000~1998というように番号が付けられてゆく。ここで998までと書いたが、分析によっては選択出来るフィールド数に上限を設け、その上限に達したらそれ以上選択出来なくする。

変数 sel と同様に、変数 con は制御フィールドの選択を表わし、初期値を

ー1として、選択された順番に番号が付けられてゆく。これは前述のように最大4フィールドまで選択可能で、選択された各制御フィールドについてはレコード抽出のためにその値の範囲を指定する。変数 var[11]は画面表示のためにフィールド名の文字列を格納するための文字配列である。

制御フィールドの設定範囲は次のリスト型構造体の中に格納される。

```
struct contype {
    char min[8];
    char max[8];
    struct contype *prev;
    struct contype *next;
};
```

制御フィールドは最大4フィールドであるから、使用範囲のデータはこのリストの4つの先頭ポインタを格納したポインタ配列によって表わされる。分割区間は何区間になるか予め分からないので、各フィールド毎にリスト形式にして、追加と削除が自由に行えるようにしている。変数 min と max はそれぞれの範囲の最小値と最大値を表わしているが、文字列データも扱えるように7桁の文字列になっている。例えばあるフィールドに“men”という文字列が入ったレコードだけを抽出する場合 min と max は共に“men”とする。データが数値の場合は、自動的に文字列データを実数に変換して使用する。

分析プログラムの計算結果を表示する際に使用するデータは、画面上の最大幅の文字列のリストであり、具体的には以下の構造である。

```
struct restype {
    char str[81];
    struct restype *prev;
    struct restype *next;
};
```

変数 `str[81]` が 1 行分の表示文字列であり（最後の 1 バイトは Null）、表示桁数が少ない場合は桁数 80 を少なくしてメモリの節約に努める。このデータを行数分リストとして繋いでスクロール画面を作成するが、リストの先頭と最後、画面表示部分の先頭と最後を常に確認しながらスクロールさせる。ただし、全体のデータがメモリに入り切らないようならば、ワークファイルを利用し、メモリ内にはワークファイルのどの部分が読み込まれているのか、画面にはその中でどの部分が表示されているのかを確認しながらスクロールさせなければならない。スクロール画面の表示機能はファイル `UTIL3.C` の中で関数化されているので、分析プログラムの中からはただ 1 つの関数を呼び出すだけでよい。

最後にデータエディターで大量のデータがメモリに入り切らない場合はワークファイルを利用するが、後に説明するレコードの挿入や削除に高速に対応出来るように `worktype` 構造体を定義する。

```
struct worktype {
    int work;
    struct worktype *prev;
    struct worktype *next;
};
```

これは挿入や削除の容易なリスト構造で、変数 `work` はワークファイル内でそのレコードが何レコード目にあるのかを与える索引変数である。則ち、レコードの挿入では `worktype` 構造体はリストのレコード番号の位置に挿入されるが、ワークファイル内では処理スピードを上げるために実際はワークファイルの最後に追加する。その際、そのデータの位置を記憶しておく変数が `work` である。削除の場合はワークファイルはそのまま `worktype` 構造体の所定のデータを削除する。ファイル等への出力の場合には `worktype` 構造体からワークファイル上のデータを読み出して出力ファイルに書き込むことになる。以上から `worktype` 構造体リストはレコード番号とワークファイル内でのデータの位置

を繋ぐ索引の役割をする。この方法はレコードの挿入や削除に対しては強力であるが、フィールドの挿入や削除については殆ど無力である。著者らはフィールドの変更はデータ入力時のレコードの増加ほど頻繁でないであろうと考え、直接ワークファイルを書き換える操作を行うようにしているが、今後検討を要する課題である。

9 終りに

ここまで、基本システムについて、概略、外部様式、内部様式と説明してきたが、プログラミングがこれで終わったわけではない。プログラムは常に発展の可能性を持っているし、また確率は低くなってもバグが含まれている可能性も残っている。それゆえ、この論文の発表後にもプログラムには多くの改良が加えられるであろう。データ構造やアルゴリズムについても同様にさらに良いものがあれば採用して行く予定である。この基本システムは平易さと簡潔さを最優先して作成されたプログラムで、現段階では基本的に際立って新しいものは含まない。また、ユーザーインターフェイスや機能の面からも他に優れたソフトウェアは多く存在する。いくつかのソフトウェアを組み合わせ利用出来るユーザーには市販のシステムで不満はないであろう。

これらの方向性に対して著者らは経営分野で利用されるソフトウェアの統合化に注目している。それ故、これまでの市販プログラムでは扱われていない新しい分析手法やユニークな分析手法も積極的に取り入れる。著者らは、これらの手法で必要なデータ形式や表示形式を考え、その形式を包含するためのシステムの拡張方針を検討する。またシステムの利用環境やデータ形式に関してだけでなく、プログラミングの立場からも統合化を推進する。則ち、処理や操作を出来るだけ大きな部品に分割し、共通の操作はこれらの高機能な部品を利用して、プログラマーが分析部分のプログラミングに集中出来るようにする。

最終的に経営分野での分析手法が個々の独立なトピックスではなく、プログラムの関連ある手法として統合化され、利用者が統一的なインターフェースで各種の問題解決に当れるシステムを作成することが目標である。

参考文献

- 1) SAS : SAS Institute Inc. の登録商標.
- 2) SPSS : SPSS Inc. の登録商標.
- 3) 高木廣文, 佐伯圭一郎, 中井里史 : 「HALBAUによる統計解析入門」, 現代数学社 (1989).
- 4) 鷹野澄, 額額一起 : 「OSシリーズ4 MS-DOS」, 共立出版 (1988).
- 5) アスキー出版局テクライト : 「PC-9800シリーズ テクニカルデータブック」, アスキー (1990).
- 6) 河西朝雄 : 「TURBO C初級プログラミング」, 技術評論社 (1989).
- 7) 篠崎寿夫, 松下祐輔 : 「工学のための応用数値計算法入門」, コロナ社 (1986).
- 8) 脇本和昌, 垂水共之, 田中豊 : 「パソコン統計解析ハンドブック I」, 共立出版 (1984).

Analyzing System UASM for Statistics and Management Science 1
— Basic System —

Summary

There exist many computer programs for statistics and management science. Especially, statistical analyzing programs are almost completed and many famous systems are used in various fields. However, analysing programs for management science are individual in each field. We hereafter consider the unification problem of analysing systems in statistics and management science, deciding the data structure and the user interface.

In this paper, we show fundamental ideas in a basic system of our sample program which is in the first step of our plan. Here, the basic system is composed of main menu, utilities and editor program. We will gradually extend our system and data structure, starting with usual two-dimensional data form used by almost programs for statistics.

We should extract constituent elements of the analyzing programs and make them subroutines for next programming. We want in the future to classify analyzing methods in terms of these elements and data structure.