
```

      READ (BYOBS, BYVAR, FILE='ファイル名文' または ファイル名,
      FORMAT=BINARY or DATABANK or EXCEL or FAME or FREE or
      LOTUS or RB8 or STATA or '(フォーマット文)',
      FULL, NCOL=列数, NROW=行数, PRINT, SETSMPL,
      TYPE=CONSTANT or DIAG or GENERAL or SYMMETRI or TRIANG, UNIT= I/O ユニット番号)
      系列, または 行列, または 定数のリスト ;
      or
      READ
  
```

機能 :

READは系列, 行列, スカラーをデータストレージに読み込むのに用います。通常, データは外部ファイルから読み込まれますが, 少量のデータの場合は, (TSPをバッチモードで実行する場合は) プログラムファイルの最後にある「データセクション」に含めることができます。もし非常に大きなデータセットを繰り返し使おうと思っている場合は, それらをもっとも高速にアクセスする方法はTSPデータバンクとしておくことです。より詳細な点についてはOUTとINコマンドを参照して下さい。

READの通常の使い方は, 自由形式の外部ファイルからいくつかのデータを読み込むことで, 次のようになります。

```
READ(FILE='FOO.DAT') X Y Z;
```

引数のないREADは, TSPプログラムの中で「データセクション」部分にコントロールを移動し, ENDステートメントまでデータを読み込むのに使われます(たとえば, STOPコマンド参照)。もしデータ部分がプログラム部分から分離しているが, 同じファイル内に置きたいなら, この機能を使って下さい。

使用法 :

TSPのプログラム中でREAD ;文に出会うと, TSPはデータセクションにコントロールを移しますそしてEND ;文(あるいはファイルの最後)に出会うまでデータを読み込みます。その点で, READ文の次のTSPプログラムのステートメントにもどります。

READステートメントに続けて単純に系列名のリストをかくと, もっとも簡単なデータの入力方法になります。特にオプションが指定されていなければ, データは, READステートメントに続けて直接自由形式であると仮定されます。つまり各数値は, 一つ以上の空白によって分離されねばなりません。データの各グループは, セミコロン (;)によって分離されますが, 必ずしも必要ではありません。もし一系列以上の読み込むデータがあるならば, データの順番は, 各系列の1番目の観測値, 2番目の観測値, というふうに続きます。

自由形式では2つの特殊な機能があります。最初は欠測値を示すのにドット(.)を使うことです(SASの場合と似ています)。しかし, TSPのその他の場所(式中のような)では, ドット(.)はDOT変数として扱われるので注意して下さい。式中のこの目的のためには, @MISSあるいは@NAを用います(FRML参照)。その他の特別な機能は繰り返し数で, 整数値の繰り返し数と, (*)印と, 繰り返されるべき数値により定められます。例えば, 3*0は000と同じです。この機能は(バンド行列のように)特殊な行列で繰り返されるゼロに対して最も頻繁に使われます。そしてFORTRANの自由形式の(*)のREADにおける反復機能と似ています。

自由形式でデータを読み込む場合は、読み込まれる項目数は、系列数掛ける観測値数で、観測値数は現在の SMPL で定まっています。TSP はこれを確かめ、合わない場合はメッセージをだします。もし SETSMPL を指定しているならば、この計算により SMPL が定められます。

もし自由形式でデータを読み込んでいるならば、NOPRINT オプションを指定していない限り、データは読み込まれるとプリントされます。固定形式あるいはフォーマットなしの読み込みでは、データはプリントされません。従って、データが正しくプログラムで読み込まれたかどうかチェックしたほうがよいでしょう。

行列を読み込むためには、READ ステートメントで行列を定義するのに必要なオプションを与え、保存するときの行列の名前を定めます。次に行列を構成する数値、行の数、を自由形式で与えます。つまり、3(行数) 掛ける 2(列数) の行列ならば、(1,1),(1,2),(2,1),(2,2),(3,1),(3,2) といった順で読み込みます。

どのようなタイプの行列でもその要素の値を全て与えることにより読み込みますが、対称行列、三角行列、そして対角行列については特殊な読み込み法があります。対称行列については、下三角の要素だけ与えればよいのです。すなわち、(1,1),(2,1),(2,2),(3,1),(3,2),(3,3)、等々の要素です。FULL オプションは、完全な行列か、下三角行列かを指定します。三角行列は、上三角の部分、つまり (1,1),(1,2),(2,2),(1,3),(2,3),(3,3) 等々、の転値を与える必要があります。対角行列については、対角要素、(1,1),(2,2),(3,3)、... だけが必要です。残りの要素は自動的にゼロになります。

表計算ファイル:

TSP は表計算ファイルへとから直接に系列と行列を読むことができます。次のファイルは TSP でサポートされています。

表計算ソフト	バージョン	ファイル名拡張子	TSP のサポート
Lotus 123, Symphony	1,2	.WKS .WK1 .WRK .WR1	Read, Write
Lotus 123	3	.WK3	Read
Lotus 123/J(Japanese)	1,2	.WJ1 .WJ2 .WK2 .WT2	Read, Write
Microsoft Excel	2	.XLS	Read, Write
Microsoft Excel	3,4	.XLS	Read
Microsoft Excel	5	.XLS, .XLW	Read(usually)
Microsoft Excel	7,97	.XLS, .XLW	not supported
Quattro Pro		.WQ1	Read (PC only)
Stata(PC or unix)	2-6	.DTA	Read

[注] TSP は一般に最も初期のバージョンのファイルフォーマットを書き込みます。このフォーマットはより最近の表計算ソフトでは読み込むことが出来ます。唯一の例外は Excel のバージョン 1 で、Microsoft は何のドキュメンテーションも提供していません。

ファイルは次の例のフォーマット SML.XLS で、48:1 から 49:1 の四半期データです。

	A	B	C
1	Date	CJMTL	PMTL
2	Mar-48	183.4	#N/A
3	Jun-48	185.2	.436
4	Sep-48	192.1	.562
5	Dec-48	193.3	.507
6	Mar-49	206.9	.603

上のファイルは次のコマンドで読むことができます。

```
READ (FILE='SML.XLS') ;
```

系列はファイルのそれぞれの列から読まれます。系列名はデータ列の上のファイルの第1行目に与えてもよい(オプション)。日付は第1列に与えても構いません。多くの異なるファイル仕様は可能で、他の系列を無視していくつかの系列だけを読むこともできます。次は、TSPで読むことができる表計算ファイルを作成するいくつかの簡単なガイドを示しています。

1. 第1行目に列名を置く。この名前はTSPで有効な系列名でなければいけません(小文字でも構いません。大文字に変換されます。しかし空白や特殊文字を含んではいけません)。有効な列名は厳密にはTSPでは必要としない。ファイルに系列名がない場合は、TSPでファイルを読む場合に系列名を与えることができますが、これは面倒です。ファイルに有効でない名前がある場合は、TSPは現在付いている名前を無視してデータを行列として読み、その後でTSPの中で適当な名前を与えることができます。TSPは下の行や第1行の下シート(Lotusバージョン3)の名前を認識しません。それらは欠測値として扱われ、その下の数値は本来の列名に対するデータとして読まれます。
2. 第2行はデータでなければいけません。
3. 時系列データを読む場合には、第1列は必ず日付を入れます。これによって系列は、TSPの現在のFREQとSMPLには関係なく、確実に適切な期種と最初の日付を付けて読まれます。日付は48:1のような文、あるいは日付形式(Mar-48, 3/31/48, など)の数値でよい。十分な日付を与えるとTSPは期種を検出する(四半期と月次を区別するには5で十分)。TSPは、データが連続している(欠測値のない期/年、あるいはTSPではSMPLギャップ)とすると、この後のすべての日付を無視します。すべての系列に欠測値のある期/年があると、対応する行を空白とします。全サンプルにわたって数値データ列を作成する簡単な方法は、先ず最初にTSPでの系列C(1の定数)を含むファイルを書いて、次にCを消して手元のデータを埋めることです。以下の表は日付の列に最初の日付と期種を定義する推奨法を示す例です。

最初の日付	2番目の日付	結果となる期種
(文日付 - Lotusでは最初の文字は'^'です)		
1	2	A(現在のFREQがAのとき), その他は N.
48:	任意	A
48:5	任意	M
48:4	49:1	Q
1948:1	1948:2	M あるいは Q(その後の日付による)
48:1	49:1	A
a2:3	任意	無効
48:2	48:1	無効
(数値日付)		
12/31/48	12/31/49	A(365-366日離れた任意の日付)
12/31/48	1/31/49	M(28-31日離れた任意の日付)
12/31/48	3/31/49	Q(90-92日離れた任意の日付)
12/31/48	1/ 1/49	N(任意の日付の期間)

他の列に日付がある場合は数値として読まれます。行列を読む場合は日付の列は無視されます(すなわち、行列としては読み込みません)。

4. 欠測値は空白のセルあるいはNA, @NA, #NA, などを評価した式によって表されます。

表計算ファイルの読み込み

表計算ファイルを読み込むには、FILE='ファイル名' オプションを用います。FORMAT=LOTUS あるいは EXCEL はオプションです。ファイル名に上でリストした拡張子 (.WKS, .WK3, .XLS, など) の一つが含まれており、FORMAT がしていない場合には、TSP はファイルの最初の数バイトをチェックし、これが上でリストした表計算プログラムのどれかのバージョンであることを確認します。逆に、もし FORMAT=LOTUS あるいは EXCEL が指定されていて、ファイル名に . を含んでいなければ、.WKS あるいは .XLS がファイル名に追加されます。(列名と日付を除いて) 行列を読むには、TYPE=GEN オプションを用います。TYPE=CONSTANT は Lotus ファイルにはサポートしていません。その代わり系列が定義されます。NCOL=, NROW=, IFFULL=, UNIT=, 等のオプションは無視されますが、SETSMPL はサポートしています。

READ コマンドで引数 (系列名) がない場合、TSP はファイルで列名を探し、その名前の系列を作成します。引数を与えた場合は、TSP はファイルの列名とマッチするかどうかを試みます。ファイル上に列名がない場合は、それぞれのデータ列に対して一つの READ 引数を与える必要があります。ファイルの内容が不明な場合は、表計算で内容をチェックし、行列として読み込みます。ファイルに列名があると思っているがそれが分からない場合は、一致しないダミー名を与えて下さい。TSP はエラーメッセージをプリントし、ファイルの列名をリストします。(TYPE=GEN を使って) 行列を読み込んだら、READ に行列名の引数を与えなくても、TSP は @LOTMAT という名前の行列を作成します。

なんらかの理由で、系列が列でなく行であった場合は、ファイルを行列に読み込んで、それを転置し、その行列を UNMAKE して系列にすることができます。

最初の文字が ' ' で始まる Lotus ファイルでは、TSP はファイルの第 1 行のテキスト文字をチェックします。名前は 8 文字に切り捨てられ、大文字に変換されます。それには対応するデータ列を 1 列に並べておかなくてはなりません。最初の列に日付がある場合は、日付の列に名前は必要ありません。埋め込まれたブランク名は無視されます。

オプション :

BYOBS/NOBYOBS はデータが観測値ごとに構成されていることを示します。すべての系列に対しての最初の観測値、次にすべての系列に対しての 2 番目の観測値、等です。これは既定オプションです。

BYVAR/NOBYVAR はデータが系列ごとに構成されていることを示します。最初の系列に対してのすべての観測値、次に 2 番目の系列に対してのすべての観測値、等です。

FILE='ファイル名文' あるいは ファイル名。この指定によりデータが保存されるファイル名を定めます。もしファイル名が 8 文字以下でアルファベット以外の文字を含まないならば、引用符で囲む必要はありません。

FORMAT=BINARY or DATABANK or EXCEL or FAME or FREE or LOTUS or RB4 or RB8 or STATA or '(フォーマットテキスト文)' はデータを読み込むフォーマットを定めます。指定がない場合は自由形式と見なされ、数値は空白によって分離され別々の長さになります。各フォーマットオプションは以下の FORMAT の項でより詳しく説明します。

FORMAT=BINARY はデータがバイナリー単精度 (REAL*4) フォーマットであることを示します。このフォーマットでデータを読み込むためには、バイナリーデータは TSP プログラムのインプットファイルと混在することはできませんから、データは外部ファイル上にあることが必要です。データを読むためのこの方法は最も効率がよいですが (TSP データバンクと同じ)、あまり使いやすくはありません。これは FORMAT=RB4 とおなじです。FORMAT=RB8 は倍精度バイナリーに対するものです。

FORMAT=DATABANK データが TSP のデータバンクから読み込まれることを指定します。

FORMAT=EXCEL はエクセル表計算ファイル (FORMAT=LOTUS と類似) を読み込みます。ファイル名が .XLS となっているファイルでは、これが既定オプションです。

READ

FORMAT=FAME は FAME データベースファイルを読み込みます。これは、これらのファイルの入出力で用いられる FAME コードに対して当該機関がライセンスを有している場合に、いくつかの TSP unix バージョンでだけサポートされています。

FORMAT=LOTUS はロータス 123, エクセル, クワトロ・プロ (Quattro Pro) ワークシートファイル (ほとんどのファイルは拡張子が .WKx, ただし x は任意の文字) を読み込みます。

FORMAT=RB4 は **FORMAT=BINARY** (単精度バイナリー) と同じです。

FORMAT=RB8 は倍精度バイナリー用です。

FORMAT=STATA は STATA データファイルを読み込みます。ファイル名が ,DTA で終わっている場合はこれが既定値です。

FORMAT= 引用符に囲まれたフォーマット文の列 '(フォーマットの列)' によって、データが読み込まれるフォーマットを定めます。引用符は必ず必要で、フォートランの **FORMAT** ステートメントを、**FORMAT** という言葉を除いて括弧をつけて両側から囲む必要があります。もしフォートランの **FORMAT** ステートメントを良く御存じない場合は、このレファレンスマニュアルの **FORMAT** の項を御参照ください。

FULL/NOFULL **NOFULL** は対角行列, 対称行列, 三角行列を読み込む際に使われます。**FULL** が指定されれば完全行列が読み込まれることになります。**NOFULL** が指定されれば、(三角行列の場合は) 上三角か、(対称行列では) 下三角か、あるいは (対角行列では) 対角要素が読み込まれます。

NCOL= 行列中の列の数を指定します。これは一般の行列では必要です。

NROW= 行列中の行の数を指定します。これは一般の行列では必要です。対称行列, 三角行列, そして対角行列に関しては **NROW** か **NCOL** のどちらかを指定しないといけません。

PRINT/NOPRINT データが読み込まれてからプリントするかどうかを指定します。このオプションは自由形式の読み込みにのみ使えます。**NOPRINT** ステートメントをプログラムにに入れることにより、**READ** 部分全体について指定することもできます。

SETSMPL/NOSETSMP **SMPL** が読み込まれるデータによって定まるかどうか指定します。**SMPL** がプログラムでまだ指定されておらず、しかも他でオフされていないばあいには **SETSMPL** 標準となります。**SMPL** がプログラム中で与えられている場合には、**NOSETSMP** が標準となります。このオプションは行列の読み込みには使われません。

TYPE=GENERAL or **SYMETRIC** or **TRIANG** or **DIAG** or **CONSTANT** は行列の形式を定めます。標準の **GENERAL** は普通の正方形, あるいは矩型の行列に使われます。**SYMETRIC** は行列がその転値に等しいことを意味します。下三角行列が保存されます。**TRIANG** は行列が三角 (対角線の上はゼロ) であることを意味します。データは下三角でインプットすればよいのですが、TSP の行列プロシジャでは上三角行列と考えますので、上三角行列の形で保存します。**DIAG** は非対角要素が総てゼロである行列を意味します。対角要素のみ保存され、使われる前に拡張されます。**CONSTANT** はスカラーの読み込みに使われ、行列に関するオプションは使えません。行列の形が指定されない場合は、**GENERAL** と理解され、警告がプリントされます。

UNIT= (通常 1 から 4 と 8 から 99 までの) 整数値を指定し、変数が読み込まれる外部ファイルの入出力 **UNIT** 番号です。通常は単に **FILE=** が用いられますが、同じファイルからいくつかの **READ** コマンドで長いファイル名をタイプするのを避けるのに **UNIT=** を用いることができます。

例 :

```
READ(FILE='FOO.DAT') X Y Z;
```

自由形式で1系列をREADするには、次のようにします。

```
SMPL 1 9 ;
LOAD IMPT ; 100 106 107 120 110 116 123 133 137 ;
```

次の例はTSPインプットファイルからフォーマット付でデータを読み込みます。

```
SMPL 1 50 ;
READ (FILE='STATES.DAT',FORMAT='(F2.0,F4.0,3F4.1,F6.0/F7.0,5F4.1,F4.0)')
STATE X1-X12 ;
```

ここでSTATE.DATには次のデータが入っています。

```
01284952.326.4 1.9 4120
19055553.320.813.7 8.9 5.46677
024592 0.021.211.0 403
303168.176.819.815.5 8.29047
... and so forth ...
```

このデータが読み込まれれば、STATEとX1からX12までの系列は次の値を持ちます。

```
STATE: 1,2,.....
X1: 2849,4592,.....
X2: 52.3,0.0,.....
X3: 26.4,21.2,.....
X4: 1.9,11.0,.....
X5: 4120,403,.....
X6: 190555,3031,.....
X7: 53.3,68.1,.....
X8: 20.8,76.8,.....
X9: 13.7,19.8,.....
X10: 8.9,15.5,.....
X11: 5.4,8.2,.....
X12: 6677,9047,.....
```

次に行列の読み込み例をあげます。

```
LOAD (NROW=4,NCOL=3) COEFMAT ;
0.32 0.5 1.3
0.30 0.4 1.35
0.25 0.61 1.1
0.28 0.55 1.23
;
LOAD (NROW=2,TYPE=SYM) COVAR ;
4.64 2.3 5.1 ;
LOAD (NROW=3,TYPE=TRIANG) TMAT ;
1
2 3
4 5 6 ;

LOAD (NCOL=5,TYPE=DIAG) BAND ;
110. 140. 0. 35. 50. ;
```

これらの4つの例で保存された行列は次のようになります。

READ

COEFMAT			COVAR		TMAT			BAND				
0.32	0.5	1.3	4.64	2.3	1	2	4	110	0	0	0	0
0.30	0.4	1.35	2.3	5.1	0	3	5	0	130	0	0	0
0.25	0.61	1.1			0	0	6	0	0	0	0	0
0.28	0.55	1.23						0	0	0	35	0
								0	0	0	0	50

表計算ファイルに対する READ の例

次の SML.WKS ファイル (前出の SML.XLS ファイルのロータス版である) を用います.

	^CJMTL	^PMTL
'48:1	183.4	NA
'48:2	185.2	.436
'48:3	192.1	.562
'48:4	193.3	.507
'49:1	206.9	.603

READ(FILE='SML.WKS'); ? 系列 CJMTL と PMTL が定義される.
 ? 現在の FREQ や SMPL がないとき
 ? FREQ Q と SMPL 48:1 49:1 がセットされる

READ(FILE='SML.WKS',TYPE=GEN); ? 5x2 行列@LOTMATが作成され,
 ? その列に CJMTL と PMTL の値が入る

READ(FILE='SML.WKS') PMTL; ? PMTL だけを読み込む

次の例は, nm3.wk1 ファイル (数値日付を使ったロータスで示した) です.

04/30/57	23.2	34.5	10.9
05/31/57	23.6	35.1	11.0
	23.9	35.8	11.2
	24.0		11.5

READ(FILE='NM3.WK1') SF LA SD; ? 57:4から57:7で定義された月次系列SF,LA,SDを作成
 ? 現在の標本期間がなければ, FREQ Mの新しい標本期間
 ? 系列LAは最後の観測値が欠測値

READ(FILE='NM3.WK1') SF; ? 3つの系列が必要なので, エラーメッセージが
 ? プリントされる.

Stata (.dta) ファイルを読んで, 変数の文書をプリントし, 変数の min/max/mean をチェックします. TSP は Stata v2-6 のファイルを読むことができます. このファイルはバイナリーで, 変数名を含んでおり, 欠測値と文書を含んでいるので表計算ファイルと似ています.

READ(FILE='acq95.dta'); ? 全変数をファイルから読む. 名前は@RNMSに保存される.
 SHOW(DOC) @RNMS; ? 全変数の文書 (Stata ラベル) を示す.
 MSD(TERSE) @RNMS; ? 全変数の min/max/mean をチェックする

アウトプット :

READは自由形式オプションが用いられていて, PRINT オプションが選択されていれば, データをプリントします. 他のはあいはREADはアウトプットをしません, 系列名のもとにデータ領域に読み込まれた全てのデータを保存します. 場合によっては特定の変数をデータ保存領域に保存します.

@NOB	スカラー	1	読み込む観測値の数 (SETSMPLが有効な場合)
@RNMS	リスト	変数の数	読み込む変数名 (表計算の場合名前がない)

参考文献 :

FORTRAN マニュアル, FORTRANプログラマーのためのレファレンスマニュアル, あるいはコンピュータ・インストールガイドにはファイルフォーマットについての詳細や, またフォーマットと UNIT 番号を物理的ファイルに結び付ける方法が述べられています.