
MATRIX 行列 = 行列方程式 ;

機能 :

MATRIXは行列代数式の処理を行います。行列の演算はMATに続く行列方程式で行われます。これらの方程式は、現在のSMPLの制御の下で作用しません。そしてMAT方程式の結果は行列の形で保存される点を除いて、ちょうどGENRで行われる変数変換と同じになります。MATプロシジャは行列演算での行と列が一致しているかどうかをチェックし、指定した演算ができない場合はエラーメッセージをプリントします。その場合、しばしばその行列をプリントすると演算が行われなかった理由がはっきりします。

使用法 :

TSP方程式で使うことができる通常の演算と関数は、すべてMATコマンドでも使うことができますが、それらは要素ごとに行います(それらが二項オペレータなら行列が一致していることが必要である)。これには1つの重要な例外があり、それは乗算オペレータ $*$ です。単純化のために、このオペレータは通常の行列乗算を表し、要素ごとの乗算(アダマール積)はオペレータ $\%$ で表します。

行列オペレータを記述する場合に、次の記号を使って演算の入出力を表します。

s	=	スカラーあるいは添字付変数
i	=	整数スカラー
m	=	任意の行列(スカラーの場合は 1×1 行列とする)
qm	=	平方行列($N \times N$)
sm	=	対称行列, 正定符号行列と仮定
dm	=	対角行列, 正定符号行列と仮定
tm	=	上三角行列, 正定符号行列と仮定
v	=	列ベクトル, $N \times 1$

次は、GENRでも使われる通常のオペレータに加えて、MATコマンドで利用できる記号オペレータです。演算は必ず必要な演算で次元があってなければいけません。TSPは次元をチェックし、満たされていない場合には計算を行いません。

m	=	$m * m$	行列積
m	=	$m * s$	スカラー積(あるいは $s * m$)
m	=	m'	行列転置
m	=	$m' m$	行列積のある行列転置
m	=	qm''	逆行列
m	=	$qm'' m$	行列積のある逆行列
m	=	$n \# m$	クロネッカー積($\otimes$)
m	=	$m \% m$	アダマール積(要素ごとの積)

TSPがMATコマンドを処理するとき、2重計算を除くことによって計算時間を大幅に節約できる場合があります。これらの状況には交差積オペレーション(対称行列になる)や2次形式の計算($A * B * A'$ の式)が含まれますがこれだけではありません。これはこれらの式の引数がそれ自体複雑である場合に生じます。した

MATRIX

がって、行列方程式に現れる場合はいつも同じように注意して、そのような複雑な引数を表す必要があります。たとえば、次の関数はそのインプットとして行列を、アウトプットとしてスカラーを作成します。これらの関数は、スカラーが許される MAT ステートメントのどこで用いても構いません。スカラーは 1×1 行列であることを忘れないようにして下さい。

s	$=$	$\text{DET}(qm)$	行列式 ($\geq 1.E-37$ のときゼロで切り捨て)
s	$=$	$\text{LOGDET}(qm)$	(正の) 行列式の対数, 切り捨てせず
s	$=$	$\text{TR}(qm)$	トレース (対角要素の和)
s	$=$	$\text{MIN}(m)$	最小値の要素
s	$=$	$\text{MAX}(m)$	最大値の要素
s	$=$	$\text{SUM}(m)$	要素の和
i	$=$	$\text{NROW}(m)$	行の数
i	$=$	$\text{NCOL}(m)$	列の数
i	$=$	$\text{RANK}(m)$	階数 (線形独立の行または列の数)

次の関数は行列から行列に対するものです：すなわち行列を作り、何らかの計算を行い、別の行列をアウトプットとして作成します。これらの関数も MAT 方程式のどこでも使えます。

tm	$=$	$\text{CHOL}(sm)$	コレスキー分解 (行列平方根)
sm	$=$	$\text{YINV}(sm)$	ハウスマン検定に対する $\text{CHOL}()$ による正準定符号逆行列
dm	$=$	$\text{IDENT}(i)$	次数が i の単位行列の生成
v	$=$	$\text{EIGVAL}(qm)$	qm の固有値ベクトルの計算. qm が対称の正準定符号でなければ固有値の虚数部分は @EIGVALI として保存.
qm	$=$	$\text{EIGVEC}(qm)$	固有ベクトル (列) の行列の計算. qm が対称の正準定符号でなければ固有ベクトルの虚数部分は @EIGVECI として保存.
v	$=$	$\text{VEC}(m)$	m のすべての要素のベクトルを列ごとに生成
v	$=$	$\text{VECH}(m)$	m のすべての一意の要素のベクトルを列ごとに生成 $qm: N \times N$ 要素 $sm, tm: N * (N + 1) / 2$ 要素 $dm: N$ 要素
dm	$=$	$\text{DIAG}(m)$	行列から対角行列を生成 qm, sm, tm : インプット行列から対角行列をとる v : ベクトルを対角行列に変換 s : 違反, 対角線に s をおいた対角行列を作成するのに $s * \text{IDENT}(i)$ を使用
sm	$=$	$\text{SYM}(qm)$	対称行列を平方行列から作成 (e.g. 上三角要素は無視)
m	$=$	$\text{GEN}(qm)$	一般行列を対称な対角行列から作成
系列	$=$	$\text{SER}(v)$	ベクトルから系列を作成 ($\text{UNMAKE } v$ 系列 と同じ)

例:

```
MAT B=(X'X)"X'Y;      は OLS 回帰係数になります (正確な計算ではありません)
```

次の例は、回帰の後推定された係数の分散共分散の Eicker-White 推定値を計算します。

```
OLSQ Y C X;  
MMAKE XMAT C X;  
MAT XXI=(XMAT'XMAT)";  
MAT VCOV=XXI*XMAT'DIAG(@RES*@RES')*XMAT*XXI;
```

アウトプット:

MATRIX は何もプリントアウトはありません。通常、1 つの行列がデータ領域に保存されます。