

AJAN

CALCコマンド
リファレンス

目 次

第 1 章	はじめに	4
第 2 章	機能説明	5
2.1	CALC(数学統計)コマンドの基本	5
2.2	FFT・逆 FFT	5
2.3	Xbar-R 管理図	9
2.4	異常度の学習	11
2.5	回帰直線	13
2.6	多次元を扱うコマンド	15
第 3 章	リファレンス	17
3.1	コマンド一覧	17
3.2	数学統計に関する関数・命令	19
3.2.1	CALC_FFT_EX	19
3.2.2	CALC_FFT_EX_STRUCT	20
3.2.3	CALC_IFFT_EX	21
3.2.4	CALC_IFFT_EX_STRUCT	22
3.2.5	CALC_CMPL2ABS	23
3.2.6	CALC_POLYFIT	24
3.2.7	CALC_HISTOGRAM	25
3.2.8	CALC_MODE	26
3.2.9	CALC_COVAR	27
3.2.10	CALC_CORREL	27
3.2.11	CALC_XBARR_LEARN	28
3.2.12	CALC_XBARR_JUDGE	30
3.2.13	CALC_CREATE_SINWAVE	32
3.2.14	CALC_CREATE_COMBINEWAVE	33
3.2.15	CALC_CREATE_TRIANGLEWAVE	34
3.2.16	CALC_CREATE_SQUAREWAVE	35
3.2.17	CALC_CREATE_SAWTOOTHWAVE	36
3.2.18	CALC_CREATE_FAKENOISE	37
3.3	数学統計に関するサブルーチン集	38
3.3.1	CALC_ANOMALY_LEARN	39
3.3.2	CALC_ANOMALY_SCORE	40
3.3.3	CALC_REGLINE	41
3.3.4	CALC_REGPRED	42
3.4	数学統計コマンド(追加分)	43
3.4.1	CALC_MINMAX_NORMALIZE	43
3.4.2	CALC_L1_NORMALIZE	44
3.4.3	CALC_L2_NORMALIZE	45
3.4.4	CALC_FFT_2D_EX	46


3.4.5	CALC_FFT_2D_EX_STRUCT.....	47
3.4.6	CALC_IFFT_2D_EX.....	48
3.4.7	CALC_IFFT_2D_EX_STRUCT	49
3.4.8	CALC_FFTSHIFT	50
3.4.9	CALC_FFTSHIFT_2D	51
3.4.10	CALC_FFTSHIFT_STRUCT.....	52
3.4.11	CALC_FFTSHIFT_2D_STRUCT.....	53
3.4.12	CALC_IFFTSHIFT	54
3.4.13	CALC_IFFTSHIFT_2D	55
3.4.14	CALC_IFFTSHIFT_STRUCT	56
3.4.15	CALC_IFFTSHIFT_2D_STRUCT.....	57
3.4.16	CALC_CONVOLUTION_1D	58
3.4.17	CALC_CONVOLUTION_2D	59
3.4.18	CALC_HILBERT_ENVELOPE.....	60
3.4.19	CALC_AVERAGE	61
3.4.20	CALC_VARIANCE.....	62
3.4.21	CALC_STANDARD_DEVIATION	63
3.4.22	CALC_THIRD_MOMENT	64
3.4.23	CALC_FOURTH_MOMENT	65
3.4.24	CALC_RMS	66
3.4.25	CALC_SKEWNESS	67
3.4.26	CALC_KURTOSIS.....	68
3.4.27	CALC_PEAK_VALUE	69
3.4.28	CALC_CREST_FACTOR.....	70
3.4.29	CALC_ABSOLUTE_AVERAGE	71
3.4.30	CALC_SHAPE_FACTOR.....	72
3.4.31	CALC_CLEARANCE_FACTOR	73
3.4.32	CALC_IMPULSE_INDICATOR	74
3.4.33	CALC_COVMAT	75
3.4.34	CALC_MULMAT.....	75
3.4.35	CALC_INVMAT	76
3.4.36	CALC_PSEINVMAT	76
3.4.37	CALC_MAHALANOBIS.....	77
第 4 章	サンプルプログラム	78
4.1	サンプルプログラム	78
第 5 章	索引	79
第 6 章	重要な情報	80

第1章 はじめに

本ドキュメントは、AJANのCALC(数学統計)コマンドの説明を記載しています。
CALCコマンド以外のコマンド(標準コマンド、IO制御コマンドなど)は、別マニュアルを用意しています。

本ドキュメントでは、説明で表現している表記として下記のように定義します。

- ・ コマンドの書式の説明において、[]内の引数は省略できます。
- ・ 文字の大小について
 コマンドは大文字 / 小文字のどちらでも動作します。
 変数名は大文字 / 小文字も同じものとして扱われます。
 ファイルパス / ファイル名は大文字/小文字で区別されます。

	<p>本ドキュメント記載の、AJANはIoT用プログラミング言語です。 Interface Linux System上でのみ動作可能です。</p>
---	---

第2章 機能説明

2.1 CALC(数学統計) コマンドの基本

CALC(数学統計) コマンドは、FFT(高速フーリエ変換)、共分散、異常度の学習計算など、一度に大量の計算を行う機能を提供します。

コマンドに渡す引数は、基本的に倍精度実数で扱います。

コマンドで利用可能な構造体の定義や、幾つかのサブルーチンは、CAL001.AJN にて用意されており、以下のようにプログラムの先頭でインクルードする必要があります。

```
' インクルードの例
INCLUDE "CAL001.AJN"

' CALC_CMPL 構造体は、CAL001.AJN 内で定義されています。
STRUCT CALC_CMPL A, B
B = CALC_FFT_EX_STRUCT(A)
```

2.2 FFT・逆FFT

数学統計でよく使われるものとして、FFT(高速フーリエ変換)があります。

FFTは、周波数分析など、様々な所で解析などに使われます。



FFTの意味や詳細については、数学統計関連の書籍を参考にしてください。

波形データを元に、FFT演算を行う関数として、「CALC_FFT_EX_STRUCT」, 「CALC_FFT_EX」が用意されています。

FFT演算を行ってGUIコマンドを併用し、折れ線グラフとして描画する事例を、以下に示します。

```
' FFT 演算を行って、折れ線グラフに可視化するサンプル
OPTION GUI

INCLUDE "CAL001.AJN"

' ウィンドウと、上下に折れ線グラフの描画領域を作成
GUCREATE WINDOW 1, "FFT 描画テスト", 1,, "1000,600",,
GUCREATE LINEGRAPH 1, 10,, "1000,300", "0,0",
GUCREATE LINEGRAPH 1, 11,, "1000,300", "0,300",

' 配列 A に 10Hz, 4Hz, 2Hz の合成波形データを、256 件作成
' 配列 B に 配列 A と同じ要素数の空データを作成
LIST A, B
A = CALC_CREATE_COMBINEWAVE(1.0, [ 10.0; 4; 2 ], 100.0, 256)
? A
REDIM B(UBOUND(A))

' 構造体 C の、REAL(実部)メンバに 配列 A の値を、IMAG(虚部)メンバに 配列 B の値を代入
STRUCT CALC_CMPL C, D
```

```

C. REAL = A
C. IMAG = B
' 構造体 C を引数に、FFT 演算を行い、結果を構造体 D にて受け取る
D = CALC_FFT_EX_STRUCT(C)
? D

' 構造体 D の REAL(実部)と IMAG(虚部)メンバを元に絶対値を求めて、配列 E に代入
LIST E
E = CALC_CMPL2ABS(1, D)

' 波形データ(A)と、FFT 演算結果(E)を、折れ線グラフとしてセット&描画させる
GUUPD LINEGRAPH VALUE 10, (A), (2), (2)
GUUPD LINEGRAPH VALUE 11, (E), (2), (2)
GUUPD LINEGRAPH DRAW 10
GUUPD LINEGRAPH DRAW 11

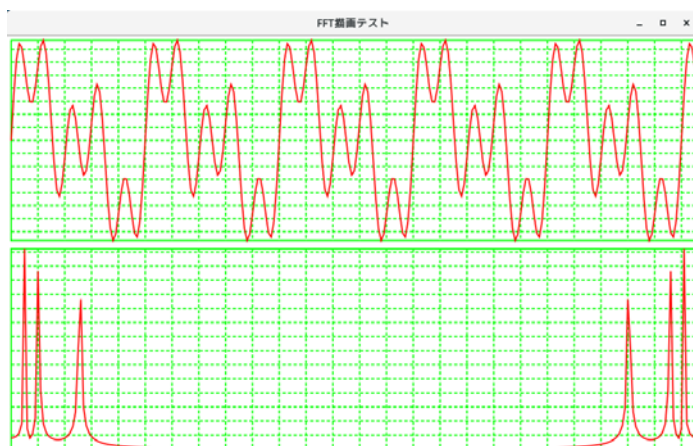
GUSHOW 1
GUMAIN LOOP

END

```

上のプログラムを実行すると、10Hz, 4Hz, 2Hzの正弦波を合成した波形データを元に、FFT演算を行い、上側に波形データ、下側に波形データをFFT演算した結果を、グラフ描画します。

上側の波形の横軸は時間で、下側の波形の横軸は周波数になります。



FFT演算結果のグラフ部分を見ると、横軸の中心（ナイキスト周波数）を境に3つほどの突起が左右対称に表示されています。

右半分は左半分の虚像で意味がないため、左半分だけ注目して、各周波数成分が特徴的にグラフに表示されている事がわかります。

プログラム中にある「CALC_CREATE_COMBINEWAVE」の、第2引数で「[10.0; 4; 2]」とある所の値を任意の値に書き換えて、再度実行してみてください。値によって、グラフの変化が目に見えます。

FFT演算の結果データを元に逆FFT演算する事で、元の波形データを求めることもできます。

逆FFT演算を行う関数として、「CALC_IFFT_EX_STRUCT」, 「CALC_IFFT_EX」が用意されています。

以下にプログラム事例を示します。

```

' FFT 演算を行って、折れ線グラフに可視化し、更に逆 FFT 演算を行うサンプル

```

OPTION GUI

```
INCLUDE "CAL001.AJN"
```

```
' ウィンドウと、上中下に折れ線グラフの描画領域を作成
```

```
GUCREATE WINDOW 1, "FFT 描画テスト", 1,, "1000,900",,
```

```
GUCREATE LINEGRAPH 1, 10,, "1000,300", "0,0",
```

```
GUCREATE LINEGRAPH 1, 11,, "1000,300", "0,300",
```

```
GUCREATE LINEGRAPH 1, 12,, "1000,300", "0,600",
```

```
' 配列 A に 10Hz, 4Hz, 2Hz の合成波形データを、256 件作成
```

```
' 配列 B に 配列 A と同じ要素数の空データを作成
```

```
LIST A, B
```

```
A = CALC_CREATE_COMBINEWAVE(1.0, [ 10.0; 4; 2 ], 100.0, 256)
```

```
? A
```

```
REDIM B(UBOUND(A))
```

```
' 構造体 C の、REAL(実部)メンバに 配列 A の値を、IMAG(虚部)メンバに 配列 B の値を代入
```

```
STRUCT CALC_CMPL C, D
```

```
C.REAL = A
```

```
C.IMAG = B
```

```
' 構造体 C を引数に、FFT 演算を行い、結果を構造体 D にて受け取る
```

```
D = CALC_FFT_EX_STRUCT(C)
```

```
? D
```

```
' [参考] 単純なデジタルフィルタ的な処理を行う部分
```

```
' FOR I=20 TO 35
```

```
' D.REAL(I) = 0
```

```
' D.IMAG(I) = 0
```

```
' D.REAL(256-I) = 0
```

```
' D.IMAG(256-I) = 0
```

```
' NEXT I
```

```
' 構造体 D の REAL(実部)と IMAG(虚部)メンバを元に絶対値を求めて、配列 E に代入
```

```
LIST E
```

```
E = CALC_CMPL2ABS(1, D)
```

```
' FFT 演算結果である構造体 D を引数に、逆 FFT 演算を行い、結果を構造体 F にて受け取る
```

```
STRUCT CALC_CMPL F
```

```
F = CALC_IFFT_EX_STRUCT(D)
```

```
' 逆 FFT 演算結果の REAL(実部)メンバを、配列 G に取り出す
```

```
LIST G
```

```
G = F.REAL
```

```
' 波形データ(A)と、FFT 演算結果(E)と、逆 FFT 演算結果の波形データ(G)を、折れ線グラフとしてセット&描画させる
```

```
GUUPD LINEGRAPH VALUE 10, (A), (2), (2)
```

```
GUUPD LINEGRAPH VALUE 11, (E), (2), (2)
```

```
GUUPD LINEGRAPH VALUE 12, (G), (2), (2)
```

```
GUUPD LINEGRAPH DRAW 10
```

```
GUUPD LINEGRAPH DRAW 11
```

```
GUUPD LINEGRAPH DRAW 12
```

```
GUSHOW 1
```

```
GUMAIN LOOP
```

```
END
```

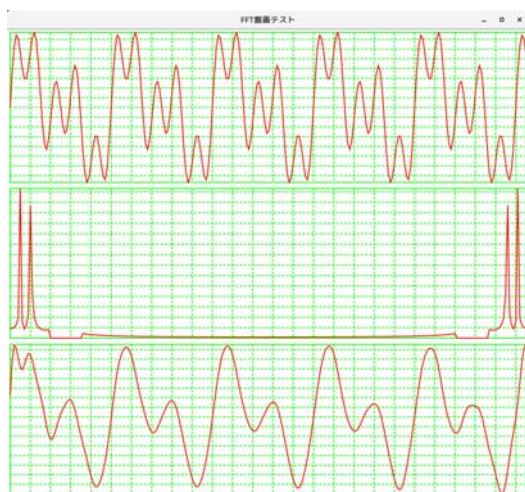
上のプログラムを実行すると、10Hz, 4Hz, 2Hzの正弦波を合成した波形データを元にFFT演算を行い、上側に波形データ、中側に波形データをFFT演算し、下側にFFT演算結果を元に逆FFT演算を行って、元の波形データを求めた結果をグラフ描画します。



「CALC_IFFT_EX_STRUCT」の逆FFT演算を行う手前で、各周波数の突起部分を消すような処理を行うと、一種のデジタルフィルタのような効果が得られます。

上のプログラムで、「[参考]」とある箇所の、「」（シングルクォーテーション）記号でコメントされている箇所の記号を削除して実行されるようにすると、その効果が確認できます。

下図に、その実行例を示します。



2.3 Xbar-R管理図

Xbar-R管理図は、工程管理や品質管理などで使われる管理図の一つです。



Xbar-R管理図の意味や詳細については、数学統計関連の書籍を参考にしてください。

AJANでは、「CALC_XBARR_LEARN」に管理図を作成する元となるデータを与えて上限値、下限値、中心値を求め、「CALC_XBARR_JUDGE」に実測したデータを与えて、範囲を超えてないか判定を行うといった使い方をします。

以下にプログラム事例を紹介します。

```
INCLUDE "CAL001.AJN"

LIST ARY
REDIM ARY(99)
FOR I=0 TO UBOUND(ARY)
    ARY(I) = RND() * 10      ' 学習データを、乱数を使って求める
NEXT I

STRUCT CALC_XBARR_ITEM INFO
INFO = CALC_XBARR_LEARN(ARY, 10)
' PRINT "学習結果:", INFO

REDIM ARY(39)
FOR I=0 TO UBOUND(ARY)
    IF I < 20 THEN
        ARY(I) = RND() * 10      ' (前半)判定データを乱数で作る
    ELSE
        ARY(I) = RND() * 20      ' (後半)判定データを乱数で作るが、係数を増やして範囲を超えやすく
    END IF
NEXT I
STRUCT CALC_XBARR_ITEM INFO2
' PRINT "判定入力:", ARY
INFO2 = CALC_XBARR_JUDGE(INFO, ARY)
' PRINT "判定結果:", INFO2

I_MAX = UBOUND(INFO2.XBAR)
PRINT "XBar 上限="; FORMAT$(INFO2.XUPCL, "###.###"); " 下限="; FORMAT$(INFO2.XLOCL, "###.###")
PRINT "R  上限="; FORMAT$(INFO2.RUPCL, "###.###"); " 下限="; FORMAT$(INFO2.RLOCL, "###.###")
FOR I=0 TO I_MAX
    PRINT "("; I; ")=XBar 平均="; FORMAT$(INFO2.XBAR(I), "###.###"); CHR$(9); "結果="; INFO2.XJUDGE(I);
    PRINT CHR$(9); "R 平均="; FORMAT$(INFO2.R(I), "###.###"); CHR$(9); "結果="; INFO2.RJUDGE(I)
NEXT I
```

元となるデータと判定に使うデータは、乱数を使用しています。

ただし、全40件のデータの内、後半部分は管理図の上限を超えやすいよう、細工を行っています。

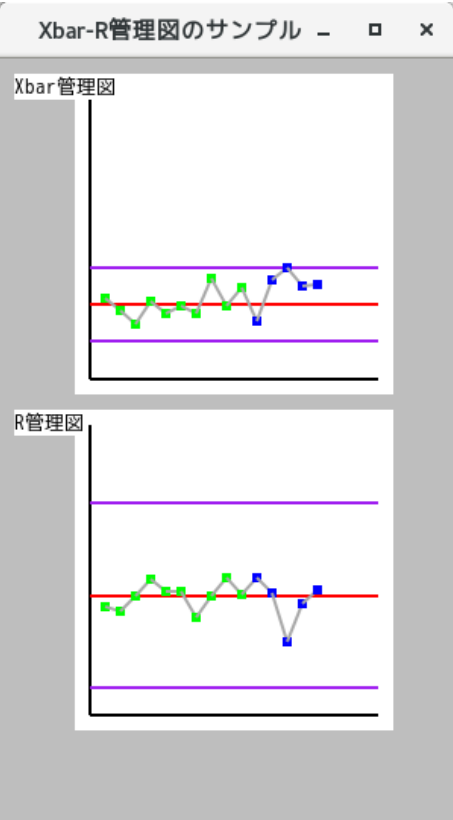
以下に、実行例を示します。

後半部分の結果が、管理図の上限を越えた結果、FALSEになっています。

```
$ ./sample
XBar 上限=7.420 下限=2.104
R      上限=15.361 下限=2.011
(0)=XBar 平均=4.260   結果=TRUE      R 平均=7.703   結果=TRUE
(1)=XBar 平均=6.980   結果=TRUE      R 平均=7.755   結果=TRUE
(2)=XBar 平均=10.521  結果=FALSE     R 平均=15.925  結果=FALSE
(3)=XBar 平均=9.549   結果=FALSE     R 平均=14.506  結果=TRUE
```

サンプルプログラム「CALC_XBARR_LEARN_JUDGE.AJN」(P. 78参照)は、Xbar-R管理図の呼び出しをグラフ化しています。

参考として、下図にサンプルプログラムを実行した例を示します。



2.4 異常度の学習

「CALC_ANOMALY_LEARN」および「CALC_ANOMALY_SCORE」は、マハラノビス距離を用いて、異常値検知を行うための関数です。

2つの関数は、CAL001.AJNにて定義されているため、「INCLUDE "CAL001.AJN"」を宣言してから、関数を呼び出す必要があります。



異常度やマハラノビス距離の意味や詳細については、数学統計関連の書籍を参考にしてください。

正常データを、「CALC_ANOMALY_LEARN」に渡すと、学習計算結果が得られます。その後、学習計算結果の平均値と分散値を引数に、新規データを「CALC_ANOMALY_SCORE」に渡すと、異常度判定が行えます。

以下にプログラム事例を紹介します。

```
INCLUDE "CAL001.AJN"

LIST ARY
ARY = [ 75; 80; 90; 95 ]      ' 学習用の正常データを準備

' 異常度学習を行う
STRUCT CALC_ANOM INFO
INFO = CALC_ANOMALY_LEARN(ARY)
PRINT "学習結果:", INFO

' 異常度判定を行うデータを準備
ARY = [ 85; 90; 105; 115 ]

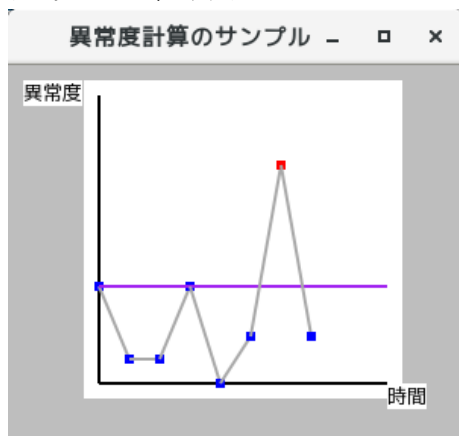
FOR I=0 TO UBOUND(ARY)
  ' 異常度判定を行う
  ANEW = CALC_ANOMALY_SCORE(INFO.MUHAT, INFO.VARHAT, ARY(I))
  PRINT "新規データの異常度=", ARY(I)
  IF ANEW < INFO.ANOM_MAX THEN
    PRINT "正常"
  ELSE
    PRINT "異常"
  END IF
NEXT I
```

以下に、実行例を示します。

```
$ ./sample
学習結果:      { MUHAT : 85, VARHAT : 62.5, ANOM_ARY : [ 1.6, 0.4, 0.4, 1.6 ], ANOM_MAX : 1.6 }
新規データの異常度=    85
正常
新規データの異常度=    90
正常
新規データの異常度=   105
異常
新規データの異常度=   115
異常
```

サンプルプログラム「CALC_ANOMALY_LEARN_SCORE.AJN」(P. 78参照)は、異常度学習と判定の呼び出しをグラフ化しています。

参考として、下図にサンプルプログラムを実行した例を示します。



2.5 回帰直線

「CALC_REGLINE」および「CALC_REGPRED」は、2つのデータ列から回帰直線を計算し予測を行うための関数です。

2つの関数は、CAL001.AJNにて定義されているため、「INCLUDE "CAL001.AJN"」を宣言してから、関数を呼び出す必要があります。



回帰直線の意味や詳細については、数学統計関連の書籍を参考にしてください。

2つのデータ列を揃えて、「CALC_REGLINE」に渡すと、回帰直線を計算して結果が得られます。その後、回帰直線結果の傾きと切片値を引数に、新規データを「CALC_REGPRED」に渡すと、予測値判定を行えます。

以下にプログラム事例を紹介します。

```
INCLUDE "CAL001.AJN"

' 計算用のデータを準備
LIST ARY1, ARY2
ARY1 = [ 83; 71; 64; 69; 69 ]
ARY2 = [ 183; 168; 171; 178; 176 ]

' 回帰直線を計算する
STRUCT CALC_REGLINE_ITEM INFO
INFO = CALC_REGLINE(ARY1, ARY2)
PRINT "計算結果:", INFO
PRINT "傾き=", INFO.A
PRINT "切片=", INFO.B

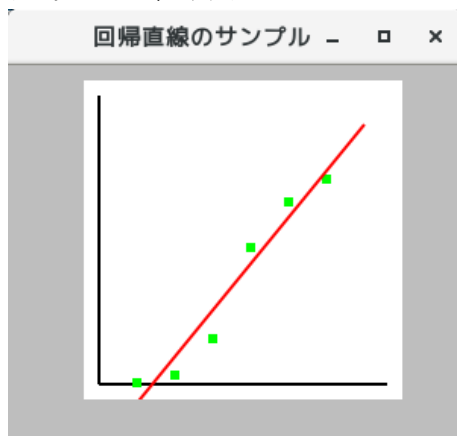
' 入力値に対する予測値を求める
Y = CALC_REGPRED(INFO.A, INFO.B, 60)
PRINT "予測値=", Y
```

以下に、実行例を示します。

```
$ ./sample
計算結果:      { MUX : 71.2, MUY : 175.2, VARX : 40.16, VARY : 27.76, SXY : 23.1599999999999, RXY :
0.693636554175354, R : 0.481131669288259, A : 0.57669322709163, B : 134.139442231076 }
傾き=    0.57669322709163
切片=   134.139442231076
予測値= 168.741035856574
```

サンプルプログラム「CALC_REGLINE_REGPRED.AJN」(P. 78参照)は、回帰直線計算と予測値の呼び出しをグラフ化しています。

参考として、下図にサンプルプログラムを実行した例を示します。



2.6 多次元を扱うコマンド

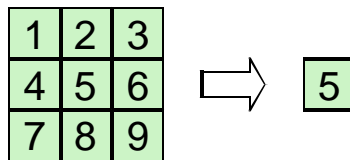


本項で紹介するコマンドは、Ver1.10より提供されました。

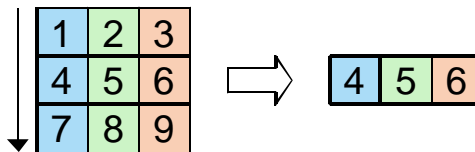
一部の数学系コマンドでは、引数に次元を指定できるようになっています。次元を指定することで、多次元配列に対してコマンドを実行する際に、計算を行うデータのまとまりを指定できるようになります。

「CALC_AVERAGE」を使って2次元配列の要素の平均値を計算してみます。

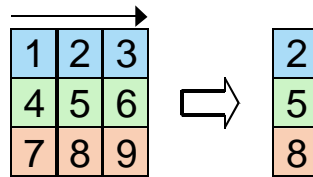
次元を指定しない場合、2次元配列の全要素の平均値を計算します。



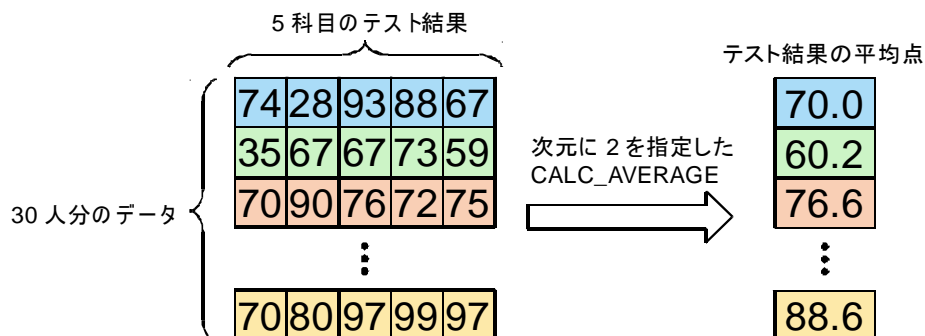
次元に1を指定した場合、2次元配列の縦方向で見た要素のまとまりで平均値を計算します。



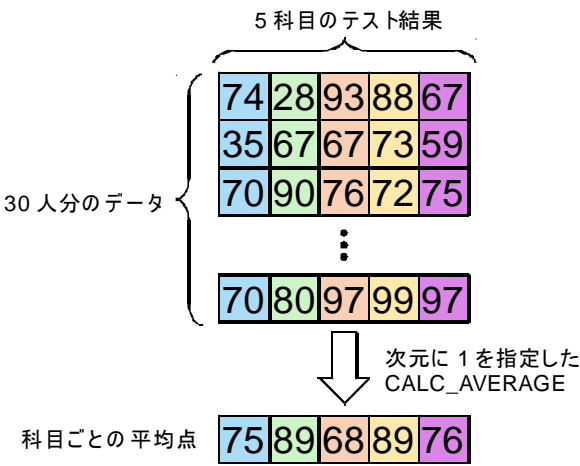
次元に2を指定した場合、2次元配列の横方向で見た要素のまとまりで平均値を計算します。



例えば5科目のテスト結果の点数が30人分あり、これらのデータが30 x 5の2次元配列に格納されているとします。この配列に対して、次元に2を指定した「CALC_AVERAGE」を実行することで、一人一人のテスト結果の平均点を計算できます。





一方で次元に1を指定した「CALC_AVERAGE」を実行することで、科目ごとの平均点を計算することもできます。



第3章 リファレンス

使用できるCALC(数学統計) コマンドの使い方について記載します。

	制限事項については、「注意」に記載しています。
	使用例は動作を保証するものではありません。 実際の使い方は各種サンプルプログラムを参照してください。

3.1 コマンド一覧

コマンド名	機能
数学統計に関する関数・命令	
CALC_FFT_EX	FFT演算を行います。(2次元配列版)
CALC_FFT_EX_STRUCT	FFT演算を行います。(構造体版)
CALC_IFFT_EX	逆FFT演算を行います。(2次元配列版)
CALC_IFFT_EX_STRUCT	逆FFT演算を行います。(構造体版)
CALC_CMPL2ABS	複素数を表す配列から絶対値を求めます。
CALC_POLYFIT	最小二乗法計算を行います。
CALC_HISTOGRAM	ヒストグラムを求めます
CALC_MODE	配列から最も頻繁に出現する値(最頻値)を求めます。
CALC_COVAR	2つの配列の共分散を求めます。
CALC_CORREL	2つの配列の相関係数を求めます。
CALC_XBARR_LEARN	Xbar-R管理図を作るための学習計算を行います。
CALC_XBARR_JUDGE	Xbar-R管理図を用いた状態判定を行います。
CALC_CREATE_SINWAVE	正弦波の波形データを作ります。
CALC_CREATE_COMBINEWAVE	正弦波を合成した波形データを作ります。
CALC_CREATE_TRIANGLEWAVE	三角波の波形データを作ります。
CALC_CREATE_SQUAREWAVE	矩形波の波形データを作ります。
CALC_CREATE_SAWTOOTHWAVE	のこぎり波の波形データを作ります。
CALC_CREATE_FAKENOISE	擬似乱数を使ったノイズの波形データを作ります。
数学統計に関するサブルーチン集	
CALC_ANOMALY_LEARN	異常度および異常度最大値の学習計算を行います。
CALC_ANOMALY_SCORE	新たに観測したデータの異常度を求めます。
CALC_REGLINE	2つのデータ列から、回帰直線を計算します。
CALC_REGPRED	回帰直線の傾き、切片値をもとに予測を行います。
数学統計コマンド(追加分)	
CALC_MINMAX_NORMALIZE	与えられた最小値と最大値を使って配列を正規化します。
CALC_L1_NORMALIZE	与えられたL1ノルムの値になるように配列を正規化します。
CALC_L2_NORMALIZE	与えられたL2ノルムの値になるように配列を正規化します。
CALC_FFT_2D_EX	2次元FFT演算を行います。(3次元配列版)
CALC_FFT_2D_EX_STRUCT	2次元FFT演算を行います。(構造体版)
CALC_IFFT_2D_EX	2次元逆FFT演算を行います。(3次元配列版)
CALC_IFFT_2D_EX_STRUCT	2次元逆FFT演算を行います。(構造体版)

コマンド名	機能
CALC_FFTSHIFT	0周波数成分が配列の中心になるように、FFT演算した配列を入れ替えます。(配列版)
CALC_FFTSHIFT_2D	0周波数成分が配列の中心になるように、FFT演算した2次元配列を入れ替えます。(配列版)
CALC_FFTSHIFT_STRUCT	0周波数成分が配列の中心になるように、FFT演算した配列を入れ替えます。(構造体版)
CALC_FFTSHIFT_2D_STRUCT	0周波数成分が配列の中心になるように、FFT演算した2次元配列を入れ替えます。(構造体版)
CALC_IFFTSHIFT	0周波数成分が配列の中心になるように入れ替えた配列を元に戻します。(配列版)
CALC_IFFTSHIFT_2D	0周波数成分が配列の中心になるように入れ替えた2次元配列を元に戻します。(配列版)
CALC_IFFTSHIFT_STRUCT	0周波数成分が配列の中心になるように入れ替えた配列を元に戻します。(構造体版)
CALC_IFFTSHIFT_2D_STRUCT	0周波数成分が配列の中心になるように入れ替えた2次元配列を元に戻します。(構造体版)
CALC_CONVOLUTION_1D	1次元配列同士の畳み込み積分を計算します。
CALC_CONVOLUTION_2D	2次元配列同士の畳み込み積分を計算します。
CALC_HILBERT_ENVELOPE	ヒルベルト変換を使用して、1次元配列で与えられた信号の包絡線を求めます。
CALC_AVERAGE	配列の平均値を求めます。
CALC_VARIANCE	配列の分散を求めます。
CALC_STANDARD_DEVIATION	配列の標準偏差を求めます。
CALC_THIRD_MOMENT	配列の3次モーメントを求めます。
CALC_FOURTH_MOMENT	配列の4次モーメントを求めます。
CALC_RMS	配列の2乗平均平方根を求めます。
CALC_SKEWNESS	与えられた入力配列の歪度を求めます。
CALC_KURTOSIS	与えられた入力配列の尖度を求めます。
CALC_PEAK_VALUE	与えられた入力配列の最大の絶対値(ピーク値)を求めます。
CALC_CREST_FACTOR	配列の波効率を求めます。
CALC_ABSOLUTE_AVERAGE	配列の絶対値の平均値を求めます。
CALC_SHAPE_FACTOR	配列の形状係数を求めます。
CALC_CLEARANCE_FACTOR	配列のクリアランス率を求めます。
CALC_IMPULSE_INDICATOR	配列のインパルスインジケータを求めます。
CALC_COVARMAT	入力情報の配列を元に、共分散行列を求めます。
CALC_MULMAT	2つの入力情報の配列同士を行列として積を求めます。
CALC_INVMAT	入力情報の配列を元に、逆行列を求めます。
CALC_PSEINVMAT	入力情報の配列を元に、擬似逆行列を求めます。
CALC_MAHALANOBIS	2つの入力情報配列の間の、マハラノビス距離を求めます。

3.2 数学統計に関する関数・命令

3.2.1 CALC_FFT_EX

関数			
機 能	FFT演算を行います。(2次元配列版)		
書 式	<(戻り値)FFT演算結果> = CALC_FFT_EX(<①入力情報の配列>)		
戻り値	戻り値	<FFT演算結果>	配列
	FFT演算した結果が得られます。 入力情報の配列と同じ形式の二次元配列が得られます。		
パラ メータ	①	<入力情報の配列>	配列
	FFT演算を行う、入力情報の入った配列を指定します。 実部と虚部を含む、実数型の $n \times 2$ 件の二次元配列を指定してください。		
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。		
使用例	<pre> ' FFT演算を行います DIM A(255, 1) CALL MAKE_WAVE() ' この呼び出しで、配列Aに波形データを生成したと仮定します DIM B(255, 1) B = CALC_FFT_EX(A) FOR I=0 TO UBOUND(B, 1) PRINT "実部="; B(I, 0) PRINT "虚部="; B(I, 1) NEXT I </pre>		

3.2.2 CALC_FFT_EX_STRUCT

関数			
機 能	FFT演算を行います。(構造体版)		
書 式	<(戻り値)FFT演算結果> = CALC_FFT_EX_STRUCT(<①入力情報の構造体>)		
戻り値	戻り値	<FFT演算結果>	構造体
	FFT演算した結果が得られます。 入力情報と同じ形式の構造体 that 得られます。		
パラ メータ	①	<入力情報の構造体>	構造体
	FFT演算を行う、入力情報の入った構造体を指定します。 以下の定義を持つ構造体を指定します。 <div>DEFINE STRUCT CALC_CMPL LIST REAL 、 実部 LIST IMAG 、 虚部 END STRUCT</div> REALとIMAGメンバは、実数型の配列です。		
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。		
使用例	' FFT演算を行います STRUCT CALC_CMPL A CALL MAKE_WAVE() ' この呼び出しで、構造体Aに波形データを生成したと仮定します STRUCT CALC_CMPL B B = CALC_FFT_EX_STRUCT(A) PRINT B.REAL ' 実部情報 PRINT B.IMAG ' 虚部情報		

3.2.3 CALC_IFFT_EX

関数			
機 能	逆FFT演算を行います。(2次元配列版)		
書 式	<(戻り値)逆FFT演算結果> = CALC_IFFT_EX(<①入力情報の配列>)		
戻り値	戻り値	<逆FFT演算結果>	配列
	逆FFT演算した結果が得られます。 入力情報の配列と同じ形式の二次元配列が得られます。		
パラ メータ	①	<入力情報の配列>	配列
	逆FFT演算を行う、入力情報の入った配列を指定します。 実部と虚部を含む、実数型の $n \times 2$ 件の2次元配列を指定してください。		
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。		
使用例	<pre> ' 逆FFT演算を行います DIM A(255, 1) CALL MAKE_WAVE() ' この呼び出しで、配列Aに波形データを生成したと仮定します DIM B(255, 1) B = CALC_FFT_EX(A) ' FFT演算 DIM C(255, 1) C = CALC_IFFT_EX(B) ' 逆FFT演算 FOR I=0 TO UBOUND(C, 1) PRINT "実部="; C(I, 0) PRINT "虚部="; C(I, 1) NEXT I </pre>		

3.2.4 CALC_IFFT_EX_STRUCT

関数			
機 能	逆FFT演算を行います。(構造体版)		
書 式	<(戻り値)逆FFT演算結果> = CALC_IFFT_EX_STRUCT(<①入力情報の構造体>)		
戻り値	戻り値	<逆FFT演算結果>	構造体
	逆FFT演算した結果が得られます。 入力情報と同じ形式の構造体 that 得られます。		
パラ メータ	①	<入力情報の構造体>	構造体
	逆FFT演算を行う、入力情報の入った構造体を指定します。 以下の定義を持つ構造体を指定します。 <div>DEFINE STRUCT CALC_CMPL LIST REAL 、実部 LIST IMAG 、虚部 END STRUCT</div> REALとIMAGメンバは、実数型の配列です。		
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。		
使用例	' 逆FFT演算を行います STRUCT CALC_CMPL A CALL MAKE_WAVE() ' この呼び出しで、構造体Aに波形データを生成したと仮定します STRUCT CALC_CMPL B B = CALC_FFT_EX_STRUCT(A) ' FFT演算 STRUCT CALC_CMPL C C = CALC_IFFT_EX_STRUCT(B) ' 逆FFT演算 PRINT C.REAL ' 実部情報 PRINT C.IMAG ' 虚部情報		

3.2.5 CALC CMPL2ABS

関数			
機 能	複素数を表す配列から絶対値を求めます。		
書 式	<(戻り値)絶対値演算結果> = CALC CMPL2ABS(<①入力モード>, <②入力情報の配列>)		
戻り値	戻り値	<絶対値演算結果>	配列
	取得モードの指定に従って、複素数→絶対値演算した結果が、実数型の配列で得られます。 式としては、 $\text{SQR}((\text{REAL} * \text{REAL}) + (\text{IMAG} * \text{IMAG}))$ と等価です。		
パラ メータ	①	< 入力モード>	数値
	複素数→絶対値演算の入力モード指定を行います。 ここで指定したモードに従って、入力情報の配列を指定してください。		
	モード値	内容	
	0	実部と虚部を含む、 $n \times 2$ 件の2次元配列を与えて下さい。	
	1	以下の定義を持つ構造体を与えてください。 <pre> DEFINE STRUCT CALC_CMPL LIST REAL 、 実部 LIST IMAG 、 虚部 END STRUCT </pre>	
	②	< 入力情報の配列>	配列／ 構造体
	複素数→絶対値演算を行う、入力情報の入った配列または構造体を指定します。 入力モードの指定と配列／構造体内容は一致させてください。 基本的に、「CALC_FFT_EX」または「CALC_FFT_EX_STRUCT」と同じデータ形式を使用します。		
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。		
使用例1	<pre> ' FFT演算結果に対して、絶対値演算を行います DIM A(255, 1) CALL MAKE_WAVE() ' この呼び出しで、配列Aに波形データを生成したと仮定します ' FFT演算を行います DIM B2(255, 1) B2 = CALC_FFT_EX(0, A) ' 絶対値演算を行います DIM B1(255) B1 = CALC_CMPL2ABS(0, B2) </pre>		

3.2.6 CALC_POLYFIT

関数			
機 能	最小二乗法計算を行います。		
書 式	<(戻り値)演算結果> = CALC_POLYFIT(<①入力情報Xの配列>, <①入力情報Yの配列>, <②次数>)		
戻り値	戻り値	<演算結果>	
	配列		
	最小二乗法演算した結果が配列で得られます。 得られる結果は、次数によって変化します。		
	次数	戻り値の要素数	得られる結果
	1	2	f(X)=aX+b (0) = a (1) = b
	2	3	f(X)=aX ² +bX+c (0) = a (1) = b (2) = c
3	4	f(X)=aX ³ +bX ² +cX+d (0) = a (1) = b (2) = c (3) = d	
⋮			
パラメータ	①	<入力情報Xの配列>, <入力情報Yの配列>	
	配列		
	最小二乗法演算を行う、入力情報の入った配列XとYを指定します。 実数型の一次元配列を指定してください。		
	②	<次数>	数値
最小二乗法演算を行う際、n次関数で近似するための次数を指定できます。 1以上の値を指定できます。			
使用例	' 配列Xと配列Yを元に、最小二乗法計算を行います。 LIST X, Y X = [1; 3; 4; 6; 7; 10] Y = [5.7; 10.4; 11.1; 19.5; 21.8; 26.2] PRINT CALC_POLYFIT(X, Y, 1)		

3.2.7 CALC_HISTOGRAM

関数			
機 能	ヒストグラムを求めます		
書 式	<(戻り値)演算結果> = CALC_HISTOGRAM(<①モード>, <②入力情報の配列>, <③階級数>, <④範囲の最小値>, <④範囲の最大値>)		
戻り値	戻り値	<演算結果>	配列
	ヒストグラム演算した結果が配列で得られます。		
パラ メータ	①	<モード>	数値
	ヒストグラムを求めた結果を返すモードを指定します。		
	モード値	得られる結果	
	0	ヒストグラムの各度の境界値と度数が、2次元配列で得られます。	
	1	ヒストグラムの度数が、1次元配列で得られます。	
	2	ヒストグラムの各度の境界値が、1次元配列で得られます。	
	②	<入力情報の配列>	配列
	ヒストグラム演算を行う、入力情報の入った配列を指定します。 実数型の1次元配列を指定してください。		
	③	<階級数>	数値
	階級の数を指定します。 1以上の値を指定できます。		
	④	<範囲の最小値>, <範囲の最大値>	数値
使用例	ヒストグラムを求める範囲の最小値から最大値を指定します。 両方同じ値を指定すると、入力情報の配列の最小値と最大値が使用されます。		
	<pre> LIST ary ary = [1; 2; 3; 1; 2] ? ary ? "mode:0="; CALC_HISTOGRAM(0, ary, 3, 1, 3) ? "mode:1="; CALC_HISTOGRAM(1, ary, 3, 1, 3) ? "mode:2="; CALC_HISTOGRAM(2, ary, 3, 1, 3) 以下、ヒストグラムを求めた、実行結果の例です。 [1, 2, 3, 1, 2] mode:0=[[1, 1.66666666666667, 2.33333333333333, 3], [2, 2, 1, 0]] mode:1=[2, 2, 1] mode:2=[1, 1.66666666666667, 2.33333333333333, 3] </pre>		

3.2.8 CALC_MODE

関数			
機 能	配列から最も頻繁に出現する値(最頻値)を求めます。		
書 式	<(戻り値)演算結果> = CALC_MODE(<①入力情報の配列>)		
戻り値	戻り値	<演算結果>	配列
	最頻値演算した結果が得られます。 最頻値が複数の場合は、配列形式で得られます。		
パラ メータ	①	<入力情報の配列>	配列
	最頻値演算を行う、入力情報の入った配列を指定します。 実数型の一次元配列を指定してください。		
使用例	‘ 可変長配列ARY中で最も頻繁に出現する値(最頻値)を求めます LIST ARY ARY = [1; 1; 2; 2; 3] PRINT CALC_MODE(ARY) ’ [1; 2] が得られます。		

3.2.9 CALC_COVAR

関数			
機 能	2つの配列の共分散を求めます。		
書 式	<(戻り値)演算結果> = CALC_COVAR(<①入力情報の配列1>, <①入力情報の配列2>)		
戻り値	戻り値	<演算結果>	数値
	共分散演算した結果が得られます。		
パラ メータ	①	<入力情報の配列1>, <入力情報の配列2>	配列
	共分散演算を行う、入力情報の入った配列を指定します。 実数型の一次元配列を指定してください。		
使用例	' ARY1とARY2の共分散値を求めます。 LIST ARY1, ARY2 ARY1 = [1; 2; 3; 5] ARY2 = [5; 4; 2; 1] PRINT CALC_COVAR(ARY1, ARY2) -2.25		

3.2.10 CALC_CORREL

関数			
機 能	2つの配列の相関係数を求めます。		
書 式	<(戻り値)演算結果> = CALC_CORREL(<①入力情報の配列1>, <①入力情報の配列2>)		
戻り値	戻り値	<演算結果>	数値
	相関係数演算した結果が得られます。		
パラ メータ	①	<入力情報の配列1>, <入力情報の配列2>	配列
	相関係数演算を行う、入力情報の入った配列を指定します。 実数型の一次元配列を指定してください。		
使用例	' ARY1とARY2の相関係数値を求めます。 LIST ARY1, ARY2 ARY1 = [1; 2; 3; 5] ARY2 = [5; 4; 2; 1] PRINT CALC_CORREL(ARY1, ARY2) -0.9621		

3.2.11 CALC_XBARR_LEARN

関数			
機能	Xbar-R管理図を作るための学習計算を行います。		
書式	<(戻り値)学習結果> = CALC_XBARR_LEARN(<①入力データの配列>, <②群の大きさ>)		
戻り値	戻り値	<学習結果>	構造体
	<p>入力データの各データを群の大きさに指定した単位に区分けして、Xbar-R管理図を作るための学習計算を行います。</p> <p>計算で得られた情報は、以下の構造体で得られます。</p> <p>(網掛けのメンバ変数が更新されます)</p> <pre> ' (L) ... CALC_XBARR_LEARN で出力 ' (J) ... CALC_XBARR_JUDGE で出力 DEFINE STRUCT CALC_XBARR_ITEM GROUP_SIZE ' (L) データの群の大きさ LIST XBAR ' (L) (J) 群の平均のデータ列 XUPCL ' (L) Xbar 管理図の上限値 XLOCL ' (L) Xbar 管理図の下限值 XCENTER ' (L) Xbar 管理図の中心線 LIST R ' (L) (J) 群の範囲のデータ列 RUPCL ' (L) R 管理図の上限値 RLOCL ' (L) R 管理図の下限值 RCENTER ' (L) R 管理図の中心線 LIST BOOL XJUDGE ' (J) Xbar 管理図の結果列 LIST BOOL RJUDGE ' (J) R 管理図の結果列 END STRUCT </pre> <p>学習結果の後、Xbar管理図とR管理図向けの値が得られます。</p> <p>Xbar管理図向けの値は、XUPCL(上限値)、XLOCL(下限値)、XCENTER(中心線)です。</p> <p>R管理図向けの値は、RUPCL(上限値)、RLOCL(下限値)、RCENTER(中心線)です。</p> <p>RLOCL(下限値)は、群の大きさが6以下の時に NaN(非数)値となります。</p> <p>(NaN値は、ISNAN関数で判定できます)</p>		
パラメータ	①	<入力データの配列>	配列
	<p>Xbar-R管理図の学習計算を行う為の入力データの入った配列を指定します。</p> <p>実数型の一次元配列を指定してください。</p> <p>入力データの配列は、100以上の要素数を与えてください。</p> <p>100未満の要素数の配列を指定すると、警告メッセージが出力されます。</p>		
	②	<群の大きさ>	数値
	<p>Xbar-R管理図の学習計算を行う際、入力データの配列に対して、群の大きさに指定した単位に分けて計算を行います。</p> <p>群の大きさは、2から10の間で指定します。</p> <p>入力データの配列の要素数に対して、割り切れる数を指定してください。割り切れない数を指定した場合は、エラーとなります。</p>		
備考	<p>構造体の定義は、CAL001.AJNにて定義されています。</p> <p>プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。</p>		
使用例	<pre> ' 正常データを元に、Xbar-R管理図の学習計算を行います。 INCLUDE "CAL001.AJN" LIST ARY REDIM ARY(99) FOR I=0 TO UBOUND(ARY) </pre>		

	<pre> ARY(I) = RND() * 10 ' 入力データとして乱数を仮に用いています。 NEXT I STRUCT CALC_XBARR_ITEM INFO INFO = CALC_XBARR_LEARN(ARY, 10) PRINT "学習結果:", INFO</pre>
--	--

3. 2. 12 CALC_XBARR_JUDGE

関数			
機 能	Xbar-R管理図を用いた状態判定を行います。		
書 式	<(戻り値)判定結果> = CALC_XBARR_JUDGE(<①学習情報>, <②入力データの配列>)		
戻り値	戻り値	<判定結果>	構造体
	入力データの各データを群の大きさで指定した単位に区分けして、Xbar-R管理図を作るための学習計算を行います。		
	計算で得られた情報は、以下の構造体で得られます。		
	(網掛けのメンバ変数が更新されます)		
	<div><div>(L) ... CALC_XBARR_LEARN で出力</div><div>(J) ... CALC_XBARR_JUDGE で出力</div><div>DEFINE STRUCT CALC_XBARR_ITEM</div><div>GROUP_SIZE (L) データの群の大きさ</div><div>LIST XBARR (L) (J) 群の平均のデータ列</div><div>XUPCL (L) Xbar 管理図の上限値</div><div>XLOCL (L) Xbar 管理図の下限値</div><div>XCENTER (L) Xbar 管理図の中心線</div><div>LIST R (L) (J) 群の範囲のデータ列</div><div>RUPCL (L) R 管理図の上限値</div><div>RLOCL (L) R 管理図の下限値</div><div>RCENTER (L) R 管理図の中心線</div><div>LIST BOOL XJUDGE (J) Xbar 管理図の結果列</div><div>LIST BOOL RJUDGE (J) R 管理図の結果列</div><div>END STRUCT</div></div>		
判定結果の後に得られる構造体の値は、引数の学習情報の値がコピーされる他、以下の値が更新されます。			
XBAR(入力データを群の大きさで区分けした際の平均値を配列にしたもの)			
R(入力データを群の大きさで区分けした際の範囲値を配列にしたもの)			
XJUDGE(XBARの各値が、上限値(XUPCL)から下限値(XLOCL)内に収まっているか否かの値を配列にしたもの)			
RJUDGE(Rの各値が、上限値(RUPCL)から下限値(RLOCL)内に収まっているか否かの値を配列にしたもの)			
GROUP_SIZE(群の大きさ)が6以下の場合、RJUDGEの判定は、上限値を越えるか否かの判定結果となります。			
パラメータ	①	<学習情報>	構造体
	Xbar-R管理図の判定計算を行う為の学習情報を与えます。		
	「CALC_XBARR_LEARN」の戻り値の、CALC_XBARR_ITEM 構造体の値を渡してください。		
	②	<入力データの配列>	数値
	Xbar-R管理図の判定計算を行う為の入力データの入った配列を指定します。		
	実数型の一次元配列を指定してください。		
入力データの要素数は、学習情報で与える CALC_XBARR_ITEM構造体の GROUP_SIZE メンバの数を群の大きさとして、割り切れる数を指定してください。			
割り切れない数を指定した場合は、エラーとなります。			
備 考	構造体の定義は、CAL001.AJNにて定義されています。		
	プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。		
使用例	' Xbar-R管理図の学習計算を行った後、新たな入力データに対して判定計算を行います。		
	INCLUDE "CAL001.AJN"		

```
LIST ARY
REDIM ARY(99)
FOR I=0 TO UBOUND(ARY)
    ARY(I) = RND() * 10 ' 入力データとして乱数を仮に用いています。
NEXT I
STRUCT CALC_XBARR_ITEM INFO
INFO = CALC_XBARR_LEARN(ARY, 10)
PRINT "学習結果:", INFO

REDIM ARY(19)
FOR I=0 TO UBOUND(ARY)
    ARY(I) = RND() * 10 ' 入力データとして乱数を仮に用いています。
NEXT I
STRUCT CALC_XBARR_ITEM INFO2
INFO2 = CALC_XBARR_JUDGE(INFO, ARY)
PRINT "判定結果:", INFO2
```

3. 2. 13 CALC_CREATE_SINWAVE

関数			
機 能	正弦波の波形データを作ります。		
書 式	<(戻り値)波形データ> = CALC_CREATE_SINWAVE(<①振幅>, <②基本周波数>, <③サンプリング周波数>, <④データ件数>)		
戻り値	戻り値	<波形データ>	配列
	正弦波の波形データを、実数の1次元配列形式で得ます。		
パラ メータ	①	<振幅>	数値
	作成する正弦波の振幅を指定します。正の実数を与えて下さい。		
	②	<基本周波数>	数値
	作成する正弦波の周波数を指定します。正の実数を与えて下さい。		
	③	<サンプリング周波数>	数値
	作成する正弦波のサンプリング周波数を指定します。正の実数を指定して下さい。		
	④	<データ件数>	数値
	作成する波形波のデータ件数を指定します。1以上の値を指定してください。		
備 考	・ 秒数換算でデータを作成したい場合、サンプリング周波数 × 秒数 の値を、データ件数に指定します。		
使用例	‘ 10Hzの正弦波の波形データを1024件作成します。 LIST A A = CALC_CREATE_SINWAVE(1.0, 10.0, 20.0, 1024)		

3.2.14 CALC_CREATE_COMBINEWAVE

関数		
機 能	正弦波を合成した波形データを作ります。	
書 式	<(戻り値)波形データ> = CALC_CREATE_COMBINEWAVE (<①振幅>, <②基本周波数の配列>, <③サンプリング周波数>, <④データ件数>)	
戻り値	戻り値	<波形データ> 配列
	正弦波を合成した波形データを、実数の1次元配列形式で得ます。	
パラ メータ	①	<振幅> 数値
	作成する合成正弦波の振幅を指定します。正の実数を与えて下さい。	
	②	<基本周波数の配列> 配列
	合成したい正弦波の周波数を、実数の1次元配列で指定します。正の実数を与えて下さい。	
	③	<サンプリング周波数> 数値
	作成する合成正弦波のサンプリング周波数を指定します。正の実数を指定して下さい。	
	④	<データ件数> 数値
	作成する合成波形波のデータ件数を指定します。1以上の値を指定してください。	
	・ 秒数換算でデータを作成したい場合、サンプリング周波数 × 秒数 の値を、データ件数に指定します。	
使用例	‘ 10Hzと4Hzと2Hzの正弦波を合成した波形データを1024件作成します。 DIM AA(2) AA = [10.0; 4; 2] LIST A A = CALC_CREATE_COMBINEWAVE(1.0, AA, 20.0, 1024)	

3. 2. 15 CALC_CREATE_TRIANGLEWAVE

関数		
機 能	三角波の波形データを作ります。	
書 式	<(戻り値)波形データ> = CALC_CREATE_TRIANGLEWAVE(<①振幅>, <②基本周波数>, <③サンプリング周波数>, <④データ件数> [, <⑤重ね合わせ数>])	
戻り値	戻り値	<波形データ> 配列
	三角波の波形データを、実数の1次元配列形式で得ます。	
パラ メータ	①	<振幅> 数値
	作成する三角波の振幅を指定します。正の実数を与えて下さい。	
	②	<基本周波数> 数値
	作成する三角波の周波数を指定します。正の実数を与えて下さい。	
	③	<サンプリング周波数> 数値
	作成する三角波のサンプリング周波数を指定します。正の実数を指定して下さい。	
	④	<データ件数> 数値
	作成する三角波のデータ件数を指定します。1以上の値を指定してください。	
	⑤	<重ね合わせ数> 数値
	三角波を作るために、指定した重ね合わせ数の正弦波を重ね合わせる事で作ります。重ね合わせ数を増やすと計算時間が延びますが、正確な三角波に近づきます。省略すると、100で計算します。	
備 考	・ 秒数換算でデータを作成したい場合、サンプリング周波数 × 秒数 の値を、データ件数に指定します。	
使用例	‘ 10Hzの三角波の波形データを1024件作成します。 LIST A A = CALC_CREATE_TRIANGLEWAVE(1.0, 10.0, 20.0, 1024)	

3.2.16 CALC_CREATE_SQUAREWAVE

関数		
機 能	矩形波の波形データを作ります。	
書 式	<(戻り値) 波形データ> = CALC_CREATE_SQUAREWAVE (<①振幅>, <②基本周波数>, <③サンプリング周波数>, <④データ件数> [, <⑤重ね合わせ数>])	
戻り値	戻り値	<波形データ> 配列
	矩形波の波形データを、実数の1次元配列形式で得ます。	
パラメータ	①	<振幅> 数値
	作成する矩形波の振幅を指定します。正の実数を与えて下さい。	
	②	<基本周波数> 数値
	作成する矩形波の周波数を指定します。正の実数を与えて下さい。	
	③	<サンプリング周波数> 数値
	作成する矩形波のサンプリング周波数を指定します。正の実数を指定して下さい。	
	④	<データ件数> 数値
	作成する矩形波のデータ件数を指定します。1以上の値を指定してください。	
	⑤	<重ね合わせ数> 数値
	矩形波を作るために、指定した重ね合わせ数の正弦波を重ね合わせる事で作ります。重ね合わせ数を増やすと計算時間が延びますが、正確な矩形波に近づきます。省略すると、100で計算します。	
備 考	・ 秒数換算でデータを作成したい場合、サンプリング周波数 × 秒数 の値を、データ件数に指定します。	
使用例	‘ 10Hzの矩形波の波形データを1024件作成します。	
	LIST A	
	A = CALC_CREATE_SQUAREWAVE(1.0, 10.0, 20.0, 1024)	

3. 2. 17 CALC_CREATE_SAWTOOTHWAVE

関数		
機 能	のこぎり波の波形データを作ります。	
書 式	<(戻り値)波形データ> = CALC_CREATE_SAWTOOTHWAVE(<①振幅>, <②基本周波数>, <③サンプリング周波数>, <④データ件数> [, <⑤重ね合わせ数>])	
戻り値	戻り値	<波形データ> 配列
	のこぎり波の波形データを、実数の1次元配列形式で得ます。	
パラ メータ	①	<振幅> 数値
	作成するのこぎり波の振幅を指定します。正の実数を与えて下さい。	
	②	<基本周波数> 数値
	作成するのこぎり波の周波数を指定します。正の実数を与えて下さい。	
	③	<サンプリング周波数> 数値
	作成するのこぎり波のサンプリング周波数を指定します。正の実数を指定して下さい。	
	④	<データ件数> 数値
	作成するのこぎり波のデータ件数を指定します。1以上の値を指定してください。	
	⑤	<重ね合わせ数> 数値
	のこぎり波を作るために、指定した重ね合わせ数の正弦波を重ね合わせる事で作ります。	
	重ね合わせ数を増やすと計算時間が延びますが、正確なのこぎり波に近づきます。省略すると、100で計算します。	
備 考	・ 秒数換算でデータを作成したい場合、サンプリング周波数 × 秒数 の値を、データ件数に指定します。	
使用例	‘ 10Hzののこぎり波の波形データを1024件作成します。 LIST A A = CALC_CREATE_SAWTOOTHWAVE(1.0, 10.0, 20.0, 1024)	

3.2.18 CALC_CREATE_FAKENOISE

関数			
機 能	擬似乱数を使ったノイズの波形データを作ります。		
書 式	<(戻り値)波形データ> = CALC_CREATE_FAKENOISE(<①データ件数>)		
戻り値	戻り値	<波形データ>	配列
	擬似乱数を使ったノイズのサンプルデータを、実数の1次元配列形式で得ます。 値は、0～1.0までです。		
パラ メータ	①	<データ件数>	数値
	作成するノイズのデータ件数を指定します。1以上の値を指定してください。		
使用例	‘ 10件分の擬似乱数を使ったノイズデータを1024件作成します。 LIST A A = CALC_CREATE_FAKENOISE (1024)		

3.3 数学統計に関するサブルーチン集

ここで紹介しているサブルーチンは、AJANコマンドを組み合わせて作ったサブルーチンです。

サブルーチンは、以下の場所に配置されています。

```
/usr/share/interface/AJANPro/include/CAL001.AJN
```

ここで紹介した命令および関数群は、このサブルーチン(CAL001.AJN)で定義&実装しています。

これらのサブルーチンを使用する際は、以下のようにプログラムの先頭にてINCLUDE命令で読み込んでから利用ください。

```
INCLUDE "CAL001.AJN"
```

3.3.1 CALC_ANOMALY_LEARN

関数			
機 能	異常度および異常度最大値の学習計算を行います。		
書 式	<(戻り値)学習結果> = CALC_ANOMALY_LEARN(<①入力データの配列>)		
戻り値	戻り値	<学習結果>	構造体
	<p>入力データの各データを正常データ(訓練データ)とし、各データの異常度をマハラノビス距離により計算します。 計算で得られた情報は、以下の構造体で得られます。</p> <pre> DEFINE STRUCT CALC_ANOM MUHAT ' 平均値 VARHAT ' 分散値 LIST ANOM_ARY ' 異常度の配列 ANOM_MAX ' 異常度の最大値 END STRUCT </pre> <p>メンバ変数の平均値(MUHAT)と分散値(VARHAT)は、「CALC_ANOMALY_SCORE」で使用します。 異常度の最大値(ANOM_MAX)は、一般的に 異常度に対して異常か否かを判定する為の しきい値に用いられます。(あるいは、この値に安全係数をかけて、しきい値に使用します)</p>		
パラメータ	①	<入力データの配列>	配列
	<p>異常度を学習する為の正常データの入った配列を指定します。 実数型の一次元配列を指定してください。</p>		
備 考	<ul style="list-style-type: none"> ・構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。 ・入力データの配列の要素数は、2以上を与えてください。 ・0除算などの事由によりエラーとなったとき、計算が未了となったメンバ値は、NaN(非数)値となるよう設定しています。(NaN値は、ISNAN関数で判定できます) 		
使用例	<pre> ' 正常データを元に、異常度の学習計算を行います。 INCLUDE "CAL001.AJN" LIST ARY ARY = [75; 80; 90; 95] ' 学習用の正常データを準備 STRUCT CALC_ANOM INFO INFO = CALC_ANOMALY_LEARN(ARY) PRINT "学習結果:", INFO </pre>		

3.3.2 CALC_ANOMALY_SCORE

関数		
機 能	新たに観測したデータの異常度を求めます。	
書 式	<(戻り値) 異常度> = CALC_ANOMALY_SCORE(<①学習時の平均値>, <②学習時の分散値>, <③新規のデータ>)	
戻り値	戻り値	<異常度> 数値
	異常度を学習した際のデータ値と、新規のデータを元に、異常度を求めます。 得られた異常度が しきい値を超えるか否かで、異常判定を行います。 しきい値は、異常度最大値または、安全係数をかけたものを一般的に使用します。	
パラ メータ	①	<学習時の平均値> 数値
	異常度を学習した際の平均値を与えます。 「CALC_ANOMALY_LEARN」の戻り値の CALC_ ANOM 構造体の MUHAT メンバ変数を与えてください。	
	②	<学習時の分散値> 数値
	異常度を学習した際の分散値を与えます。 「CALC_ANOMALY_LEARN」の戻り値の CALC_ ANOM 構造体の VARHAT メンバ変数を与えてください。	
	③	<新規のデータ> 数値
	異常度を求めたい、新たなデータを与えます。	
備 考	<ul style="list-style-type: none">・ 構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。・ 学習時の平均値と分散値は、「CALC_ANOMALY_LEARN」の戻り値を使用してください。そうでない場合、正しい異常度値が得られません。・ 学習時の分散値に、0または負数を与えないでください。	
使用例	<pre>' 異常度の学習結果を元に、新たなデータの異常度を求めて、判定をくだします。 INCLUDE "CAL001.AJN" LIST ARY ARY = [75; 80; 90; 95] ' 学習用の正常データを準備 STRUCT CALC_ANOM INFO INFO = CALC_ANOMALY_LEARN(ARY) PRINT "学習結果:", INFO ANEW = CALC_ANOMALY_SCORE(INFO.MUHAT, INFO.VARHAT, 85) PRINT "新規データの異常度=", ANEW ' この例では、異常度判定のしきい値に、学習結果の異常度最大値を そのまま使用しています。 IF ANEW < INFO.ANOM_MAX THEN PRINT "正常" ELSE PRINT "異常" END IF</pre>	

3.3.3 CALC_REGLINE

関数			
機 能	2つのデータ列から、回帰直線を計算します。		
書 式	<(戻り値)回帰直線結果> = CALC_REGLINE(<①入力データの配列1>, <①入力データの配列2>)		
戻り値	戻り値	<回帰直線結果>	構造体
	<p>2つの入力データ列について、その関係性を回帰直線 ($y = ax + b$) で与えた時、誤差が最も小さくなる 傾き (a) と切片 (b) の値を計算で見つけます。 計算で得られた情報は、以下の構造体で得られます。</p> <pre> DEFINE STRUCT CALC_REGLINE_ITEM MUX ' データ列 1 の平均値 MUY ' データ列 2 の平均値 VARX ' データ列 1 の分散値 VARY ' データ列 2 の分散値 SXY ' データ列 1 と 2 の共分散値 RXY ' データ列 1 と 2 の相関係数値 R ' 寄与率 A ' 回帰直線の傾き B ' 回帰直線の切片 END STRUCT </pre>		
パラメータ	①	<入力データの配列1>, <入力データの配列2>	配列
	回帰直線を求めるための入力データを配列で指定します。 実数型の一次元配列を指定して、同じ要素数にしてください。		
備 考	<ul style="list-style-type: none"> 構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。 入力データの配列の要素数は、2以上を与えてください。 入力データの配列の中身を全て同じ値にはできません。 分散値が0となり、その後の計算でエラーとなります。 0除算などの事由によりエラーとなったとき、計算が未了となったメンバ値は、NaN(非数)となるよう設定しています。(NaN値は、ISNAN関数で判定できます) 		
使用例	<pre> ' 2つのデータ列から、回帰直線を計算します。 INCLUDE "CAL001.AJN" LIST ARY1, ARY2 ARY1 = [83; 71; 64; 69; 69] ARY2 = [183; 168; 171; 178; 176] STRUCT CALC_REGLINE_ITEM INFO INFO = CALC_REGLINE(ARY1, ARY2) PRINT "計算結果:", INFO PRINT "傾き=", INFO.A PRINT "切片=", INFO.B </pre>		

3.3.4 CALC_REGPRED

関数		
機 能	回帰直線の傾き、切片値をもとに予測を行います。	
書 式	<(戻り値) 予測値> = CALC_REGPRED(<①回帰直線の傾き>, <②回帰直線の切片>, <③回帰直線への入力値>)	
戻り値	戻り値	<予測値> 数値
	得られた回帰直線 ($y = ax + b$) を元に、予測値 (y に相当) を得ます。	
パラ メータ	①	<回帰直線の傾き> 数値
	回帰直線の予測を求める為の傾きを与えます。 「CALC_REGLINE」の戻り値の、CALC_REGLINE_ITEM 構造体の A メンバ変数を与えてください。	
	②	<回帰直線の切片> 数値
	回帰直線の予測を求める為の切片を与えます。 「CALC_REGLINE」の戻り値の、CALC_REGLINE_ITEM 構造体の B メンバ変数を与えてください。	
	③	<回帰直線への入力値> 数値
	回帰直線の予測を求める為の入力データ (x に相当) を与えます。	
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。	
使用例	<pre>' 2つのデータ列から、回帰直線を計算した後、予測値を求めます。 INCLUDE "CAL001.AJN" LIST ARY1, ARY2 ARY1 = [83; 71; 64; 69; 69] ARY2 = [183; 168; 171; 178; 176] STRUCT CALC_REGLINE_ITEM INFO INFO = CALC_REGLINE(ARY1, ARY2) PRINT "計算結果:", INFO PRINT "傾き=", INFO.A PRINT "切片=", INFO.B Y = CALC_REGPRED(INFO.A, INFO.B, 60) PRINT "予測値=", Y</pre>	

3.4 数学統計コマンド(追加分)



本章で紹介するコマンドは、Ver1.10より提供されました。

3.4.1 CALC_MINMAX_NORMALIZE

関数			
機 能	与えられた最小値と最大値を使って配列のスケールを揃える正規化を行います。		
書 式	$\langle \text{戻り値} \rangle \text{正規化された配列} = \text{CALC_MINMAX_NORMALIZE}(\langle \text{①入力情報の配列} \rangle, \langle \text{②最小値} \rangle, \langle \text{③最大値} \rangle [, \langle \text{④次元} \rangle])$		
戻り値	戻り値	$\langle \text{正規化された配列} \rangle$	配列
	正規化された配列が得られます。 入力情報の配列と同じ形式の配列が得られます。		
パラメータ	①	$\langle \text{入力情報の配列} \rangle$	配列
	正規化する配列を指定します。 整数型および実数型である必要があります。		
	②	$\langle \text{最小値} \rangle$	数値
	正規化に使用する最小値を指定します。		
	③	$\langle \text{最大値} \rangle$	数値
	正規化に使用する最大値を指定します。		
④	$\langle \text{次元} \rangle$		数値
	正規化する配列の次元を指定します。 次元を指定しなかった場合は配列の全ての要素で正規化を行います。		
備 考	・存在しない次元を指定するとエラーが返ります。		
使用例1	’ 配列Aを最小値0, 最大値1で正規化します。 DIM A(4) A = [0, 1, 2, 3, 4] A = CALC_MINMAX_NORMALIZE(A, 0, 1) ? A [0, 0.25, 0.5, 0.75, 1]		
使用例2	’ 配列Aの1次元目を最小値0, 最大値1で正規化します。 DIM A(2, 2) A = [[0, 1, 2], [3, 4, 5], [6, 7, 8]] A = CALC_MINMAX_NORMALIZE(A, 0, 1, 1) ? A [[0, 0, 0], [0.5, 0.5, 0.5], [1, 1, 1]]		
使用例3	’ 配列Aの2次元目を最小値0, 最大値1で正規化します。 DIM A(2, 2) A = [[0, 1, 2], [3, 4, 5], [6, 7, 8]] A = CALC_MINMAX_NORMALIZE(A, 0, 1, 2) ? A [[0, 0.5, 1], [0, 0.5, 1], [0, 0.5, 1]]		

3.4.2 CALC_L1_NORMALIZE

関数			
機 能	<p>与えられたL1ノルムの値になるように配列を正規化します。 L1ノルムの計算は以下の式を使います。</p> $ x_1 + x_2 + \dots + x_n $		
書 式	<p>〈(戻り値)正規化された配列〉 = CALC_L1_NORMALIZE(〈①入力情報の配列〉, 〈②L1ノルムの値〉 [, 〈③次元〉])</p>		
戻り値	戻り値	〈正規化された配列〉	配列
	<p>与えられたL1ノルムの値になるように正規化された配列が得られます。 入力情報の配列と同じ形式の配列が得られます。</p>		
パラ メータ	①	〈入力情報の配列〉	配列
	<p>正規化する配列を指定します。 整数型および実数型である必要があります。</p>		
	②	〈L1ノルムの値〉	数値
	<p>正規化で使用するL1ノルムの値を指定します。</p>		
	③	〈次元〉	数値
	<p>正規化する配列の次元を指定します。 次元を指定しなかった場合は配列の全ての要素で正規化を行います。</p>		
備 考	<p>・ 存在しない次元を指定するとエラーが返ります。</p>		
使用例1	<p>’ 配列AのL1ノルムが2になるように正規化を行います。 DIM A(4) A = [0, 1, 2, 3, 4] A = CALC_L1_NORMALIZE(A, 2) ? A [0, 0.2, 0.4, 0.6, 0.8]</p>		
使用例2	<p>’ 配列Aの1次元目のL1ノルムが1になるように正規化を行います。 DIM A(2, 2) A = [[3, 12, 1], [5, 8, 1], [2, 0, 3]] A = CALC_L1_NORMALIZE(A, 1, 1) ? A [[0.3, 0.6, 0.2], [0.5, 0.4, 0.2], [0.2, 0, 0.6]]</p>		
使用例3	<p>’ 配列Aの2次元目のL1ノルムが1になるように正規化を行います。 DIM A(2, 2) A = [[3, 5, 2], [12, 8, 0], [1, 1, 3]] A = CALC_L1_NORMALIZE(A, 1, 2) ? A [[0.3, 0.5, 0.2], [0.6, 0.4, 0], [0.2, 0.2, 0.6]]</p>		

3.4.3 CALC_L2_NORMALIZE

関数		
機 能	<p>与えられたL2ノルムの値になるように配列を正規化します。 L2ノルムの計算は以下の式を使います。</p> $\sqrt{(x_1^2 + x_2^2 + \dots + x_n^2)}$	
書 式	<p><(戻り値)正規化された配列> = CALC_L2_NORMALIZE(<①入力情報の配列>, <②L2ノルムの値> [, <③次元>])</p>	
戻り値	戻り値	<p><正規化された配列></p> <p>与えられたL2ノルムの値になるように正規化された配列が得られます。 入力情報の配列と同じ形式の配列が得られます。</p>
パラ メータ	①	<p><入力情報の配列></p> <p>正規化する配列を指定します。 整数型および実数型である必要があります。</p>
	②	<p>< L2ノルムの値></p> <p>正規化で使用するL2ノルムの値を指定します。</p>
	③	<p><次元></p> <p>正規化する配列の次元を指定します。 次元を指定しなかった場合は配列の全ての要素で正規化を行います。</p>
備 考	<p>・存在しない次元を指定するとエラーが返ります。</p>	
使用例1	<p>’配列AのL2ノルムが2になるように正規化を行います。 DIM A(5) A = [0, 1, 2, 3, 4, 5] A = CALC_L2_NORMALIZE(A, 2) ? A [0, 0.26967994, 0.53935989, 0.80903983, 1.07871978, 1.34839972]</p>	
使用例2	<p>’配列Aの2次元目のL2ノルムが1になるように正規化を行います。 DIM A(1, 2) A = [[0, 1, 2], [3, 4, 5]] A = CALC_L2_NORMALIZE(A, 1, 2) ? A [[0, 0.447213594999916, 0.8944271899999832], [0.424264068823857, 0.565685425098476, 0.707106781373095]]</p>	

3. 4. 4 CALC_FFT_2D_EX

関数			
機 能	2次元FFT演算を行います。(3次元配列版)		
書 式	<(戻り値)2次元FFT演算結果> = CALC_FFT_2D_EX(<①入力情報の配列>)		
戻り値	戻り値	<2次元FFT演算結果>	配列
	2次元FFT演算した結果が得られます。 入力情報の配列と同じ形式の3次元配列が得られます。 得られたFFT演算の結果の中心が高周波成分、周辺が低周波成分になります。 		
パラ メータ	①	<入力情報の配列>	配列
	2次元FFT演算を行う、入力情報の入った配列を指定します。 実数型のN × M × 2件の3次元配列を指定してください。 N, Mはそれぞれ画像の縦方向の画素数と横方向の画素数です。 配列の3次元目には、0要素目に実部の値を指定し、1要素目に虚部の値を指定します。		
使用例	<pre>' 2次元FFT演算を行います。 DIM A(255, 255, 1) CALL LOAD_IMAGE() ' この呼び出しで、配列Aに画像データを生成したと仮定します。 DIM B(255, 255, 1) B = CALC_FFT_2D_EX(A) FOR I = 0 TO UBOUND(B, 1) FOR J = 0 TO UBOUND(B, 2) PRINT "実部="; B(I, J, 0) PRINT "虚部="; B(I, J, 1) NEXT J NEXT I</pre>		

3.4.5 CALC_FFT_2D_EX_STRUCT

関数		
機 能	2次元FFT演算を行います。(構造体版)	
書 式	<(戻り値)2次元FFT演算結果> = CALC_FFT_2D_EX_STRUCT(<①入力情報の構造体>)	
戻り値	<div> <div>戻り値</div> <div><2次元FFT演算結果></div> <div>構造体</div> </div> <p>2次元FFT演算した結果が得られます。 入力情報の構造体と同じ形式の構造体 that 得られます。 得られたFFT演算の結果の中心が高周波成分、周辺が低周波成分になります。</p> <p style="text-align: center;">ナイキスト周波数</p>	
パラメータ	<div> <div>①</div> <div><入力情報の構造体></div> <div>構造体</div> </div> <p>2次元FFT演算を行う、入力情報の入った構造体を指定します。 以下の定義を持つ構造体を指定します。</p> <pre> DEFINE STRUCT CALC_CMPL LIST REAL ' 実部 LIST IMAG ' 虚部 END STRUCT </pre> <p>REALとIMAGメンバはどちらも実数型のN×M件の2次元配列です。 N,Mはそれぞれ画像の縦方向の画素数と横方向の画素数です。</p>	
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。	
使用例	<pre> ' 2次元FFT演算を行います。 STRUCT CALC_CMPL A CALL LOAD_IMAGE() ' この呼び出しで、構造体Aに画像データを生成したと仮定します。 STRUCT CALC_CMPL B B = CALC_FFT_2D_EX_STRUCT(A) FOR I = 0 TO UBOUND(B.REAL, 1) FOR J = 0 TO UBOUND(B.REAL, 2) PRINT "実部=" ; B.REAL(I, J) PRINT "虚部=" ; B.IMAG(I, J) NEXT J NEXT I </pre>	

3.4.6 CALC_IFFT_2D_EX

関数			
機 能	2次元逆FFT演算を行います。(3次元配列版)		
書 式	<(戻り値) 2次元逆FFT演算結果> = CALC_IFFT_2D_EX(<①入力情報の配列>)		
戻り値	戻り値	< 2次元逆FFT演算結果>	配列
	2次元逆FFT演算した結果が得られます。 入力情報の配列と同じ形式の3次元配列が得られます。		
パラ メータ	①	<入力情報の配列>	配列
	2次元逆FFT演算を行う、入力情報の入った配列を指定します。 実数型のN × M × 2件の3次元配列を指定してください。 N, Mはそれぞれ画像の縦方向の画素数と横方向の画素数です。 配列の3次元目には、0要素目に実部の値を指定し、1要素目に虚部の値を指定します。		
使用例	<pre>' 2次元逆FFT演算を行います。 DIM A(255, 255, 1) CALL LOAD_IMAGE() ' この呼び出しで、配列Aに画像データを生成したと仮定します。 DIM B(255, 255, 1) B = CALC_FFT_2D_EX(A) ' 2次元FFT演算 DIM C(255, 255, 1) C = CALC_IFFT_2D_EX(B) ' 2次元逆FFT演算 FOR I = 0 TO UBOUND(B, 1) FOR J = 0 TO UBOUND(B, 2) PRINT "実部=" ; C(I, J, 0) PRINT "虚部=" ; C(I, J, 1) NEXT J NEXT I</pre>		

3.4.7 CALC_IFFT_2D_EX_STRUCT

関数		
機 能	2次元逆FFT演算を行います。(構造体版)	
書 式	<(戻り値)2次元逆FFT演算結果> = CALC_IFFT_2D_EX_STRUCT(<①入力情報の構造体>)	
戻り値	戻り値	<2次元逆FFT演算結果> 構造体
	2次元逆FFT演算した結果が得られます。 入力情報の構造体と同じ形式の構造体 that 得られます。	
パラメータ	①	<入力情報の構造体> 構造体
	2次元逆FFT演算を行う、入力情報の入った構造体を指定します。 以下の定義を持つ構造体を指定します。 <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre> DEFINE STRUCT CALC_CMPL LIST REAL ' 実部 LIST IMAG ' 虚部 END STRUCT </pre> </div> REALとIMAGメンバはどちらも実数型のN×M件の2次元配列です。 N, Mはそれぞれ画像の縦方向の画素数と横方向の画素数です。	
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。	
使用例	<pre> ' 2次元逆FFT演算を行います。 STRUCT CALC_CMPL A CALL LOAD_IMAGE() ' この呼び出しで、構造体Aに画像データを生成したと仮定します。 STRUCT CALC_CMPL B, C B = CALC_FFT_2D_EX_STRUCT(A) ' 2次元FFT演算 C = CALC_IFFT_2D_EX_STRUCT(B) ' 2次元逆FFT演算 FOR I = 0 TO UBOUND(C.REAL, 1) FOR J = 0 TO UBOUND(C.REAL, 2) PRINT "実部=" ; C.REAL(I, J) PRINT "虚部=" ; C.IMAG(I, J) NEXT J NEXT I </pre>	

3.4.8 CALC_FFTSHIFT

関数			
機 能	0周波数成分が配列の中心になるように、FFT演算した配列を入れ替えます。(配列版)		
書 式	<(戻り値) 入れ替えた配列> = CALC_FFTSHIFT(<①入力情報の配列> [, <②複素数フラグ>])		
戻り値	戻り値	<入れ替えた配列>	配列
	FFT演算した配列を、0周波数成分が配列の中心になるように入れ替えた配列が得られます。入力情報の配列と同じ形式の配列が得られます。		
パラ メータ	①	<入力情報の配列>	配列
	FFT演算した配列を指定します。 実数型の配列を指定します。		
	②	<複素数フラグ>	数値
	入力情報の配列を複素数として扱うかを指定します。 TRUE: 複素数として扱います。入力情報の配列はN × 2件の2次元配列になります。(デフォルト) FALSE: 実数として扱います。入力情報の配列はN件の1次元配列になります。		
使用例	DIM A(4) A = [1, 2, 3, 4, 5] DIM B(4) B = CALC_FFTSHIFT(A, FALSE) ? B [4, 5, 1, 2, 3]		

3.4.9 CALC_FFTSHIFT_2D

関数			
機 能	0周波数成分が配列の中心になるように、FFT演算した2次元配列を入れ替えます。(配列版)		
書 式	<(戻り値)入れ替えた配列> = CALC_FFTSHIFT_2D(<①入力情報の配列> [, <②複素数フラグ>])		
戻り値	戻り値	<入れ替えた配列>	配列
	FFT演算した2次元配列を、0周波数成分が配列の中心になるように入れ替えた2次元配列が得られます。 入力情報の配列と同じ形式の配列が得られます。		
パラ メータ	①	<入力情報の配列>	配列
	FFT演算した配列を指定します。 実数型の配列を指定します。		
	②	<複素数フラグ>	数値
	入力情報の配列を複素数として扱うかを指定します。 TRUE: 複素数として扱います。入力情報の配列はN × M × 2件の3次元配列になります。 (デフォルト) FALSE: 実数として扱います。入力情報の配列はN × M件の2次元配列になります。		
使用例	DIM A(2, 2) A = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] DIM B(2, 2) B = CALC_FFTSHIFT_2D(A, FALSE) ? B [[9, 7, 8], [3, 1, 2], [6, 4, 5]]		

3. 4. 10 CALC_FFTSHIFT_STRUCT

関数			
機 能	0周波数成分が配列の中心になるように、FFT演算した配列を入れ替えます。(構造体版)		
書 式	<(戻り値)入れ替えた構造体> = CALC_FFTSHIFT_STRUCT(<①入力情報の構造体>)		
戻り値	戻り値	<入れ替えた構造体>	構造体
	0周波数成分が配列の中心になるように入れ替えた配列を持つ構造体が得られます。 入力情報と同じ形式の構造体 that 得られます。		
パラ メータ	①	<入力情報の構造体>	構造体
	FFT演算した配列をメンバに持つ構造体を指定します。 以下の定義を持つ構造体を指定します。 <div>DEFINE STRUCT CALC_CMPL LIST REAL '実部 LIST IMAG '虚部 END STRUCT</div> REALとIMAGメンバは、実数型のN件の配列です。		
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。		
使用例	STRUCT CALC_CMPL A A.REAL = [1, 2, 3, 4, 5] A.IMAG = [6, 7, 8, 9, 10] STRUCT CALC_CMPL B B = CALC_FFTSHIFT_STRUCT(A) ? B.REAL [4, 5, 1, 2, 3] ? B.IMAG [9, 10, 6, 7, 8]		

3. 4. 11 CALC_FFTSHIFT_2D_STRUCT

関数			
機 能	0周波数成分が配列の中心になるように、FFT演算した2次元配列を入れ替えます。(構造体版)		
書 式	<(戻り値)入れ替えた構造体> = CALC_FFTSHIFT_2D_STRUCT(<①入力情報の構造体>)		
戻り値	戻り値	<入れ替えた構造体>	構造体
	0周波数成分が配列の中心になるように入れ替えた2次元配列を持つ構造体が得られます。 入力情報と同じ形式の構造体 that 得られます。		
パラ メータ	①	<入力情報の構造体>	構造体
	FFT演算した2次元配列をメンバに持つ構造体を指定します。 以下の定義を持つ構造体を指定します。 <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <pre> DEFINE STRUCT CALC_CMPL LIST REAL '実部 LIST IMAG '虚部 END STRUCT </pre> </div> REALとIMAGメンバはどちらも実数型のN×M件の2次元配列です。 N, Mはそれぞれ画像の縦方向の画素数と横方向の画素数です。		
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。		
使用例	<pre> STRUCT CALC_CMPL A DIM DATA(2, 2) DATA = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] A.REAL = DATA DATA = [[10, 11, 12], [13, 14, 15], [16, 17, 18]] A.IMAG = DATA STRUCT CALC_CMPL B B = CALC_FFTSHIFT_2D_STRUCT(A) ? B.REAL [[9, 7, 8], [3, 1, 2], [6, 4, 5]] ? B.IMAG [[18, 16, 17], [12, 10, 11], [15, 13, 14]] </pre>		

3. 4. 12 CALC_IFFTSHIFT

関数			
機 能	CALC_FFTSHIFTの逆変換です。 0周波数成分が配列の中心になるように入れ替えた配列を元に戻します。(配列版)		
書 式	<(戻り値)入れ替えた配列> = CALC_IFFTSHIFT(<①入力情報の配列> [, <②複素数フラグ>])		
戻り値	戻り値	<入れ替えた配列>	配列
	0周波数成分が配列の外側になるように、入力情報の配列を入れ替えた配列が得られます。 入力情報の配列と同じ形式の配列が得られます。		
パラ メータ	①	<入力情報の配列>	配列
	0周波数成分が配列の中心にある、FFT演算した配列を指定します。 実数型の配列を指定します。		
	②	<複素数フラグ>	数値
	入力情報の配列を複素数として扱うかを指定します。 TRUE: 複素数として扱います。入力情報の配列はN × 2件の2次元配列になります。(デフォルト) FALSE: 実数として扱います。入力情報の配列はN件の1次元配列になります。		
使用例	DIM A(4) A = [1, 2, 3, 4, 5] DIM B(4) B = CALC_IFFTSHIFT(A, FALSE) ? B [3, 4, 5, 1, 2]		

3.4.13 CALC_IFFTSHIFT_2D

関数			
機 能	CALC_FFTSHIFT_2Dの逆変換です。 0周波数成分が配列の中心になるように入れ替えた2次元配列を元に戻します。(配列版)		
書 式	<(戻り値)入れ替えた配列> = CALC_IFFTSHIFT_2D(<①入力情報の配列> [, <②複素数フラグ>])		
戻り値	戻り値	<入れ替えた配列>	配列
	0周波数成分が2次元配列の外側になるように、入力情報の配列を入れ替えた配列が得られます。 入力情報の配列と同じ形式の配列が得られます。		
パラ メータ	①	<入力情報の配列>	配列
	0周波数成分が配列の中心にある、FFT演算した配列を指定します。 実数型の配列を指定します。		
	②	<複素数フラグ>	数値
	入力情報の配列を複素数として扱うかを指定します。 TRUE: 複素数として扱います。入力情報の配列はN × N × 2件の3次元配列になります。 (デフォルト) FALSE: 実数として扱います。入力情報の配列はN × N件の2次元配列になります。		
使用例	DIM A(2, 2) A = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] DIM B(2, 2) B = CALC_IFFTSHIFT_2D(A, FALSE) ? B [[5, 6, 4], [8, 9, 7], [2, 3, 1]]		

3. 4. 14 CALC_IFFTSHIFT_STRUCT

関数			
機 能	CALC_FFTSHIFT_STRUCTの逆変換です。 0周波数成分が配列の中心になるように入れ替えた配列を元に戻します。(構造体版)		
書 式	<(戻り値)入れ替えた構造体> = CALC_IFFTSHIFT_STRUCT(<①入力情報の構造体>)		
戻り値	戻り値	<入れ替えた構造体>	構造体
	0周波数成分が配列の外側になるように、入力情報の配列を入れ替えた配列を持つ構造体 が得られます。 入力情報と同じ形式の構造体 that 得られます。		
パラ メータ	①	<入力情報の構造体>	構造体
	0周波数成分が配列の中心にある、FFT演算した配列をメンバに持つ構造体を指定します。 以下の定義を持つ構造体を指定します。 <div>DEFINE STRUCT CALC_CMPL LIST REAL '実部 LIST IMAG '虚部 END STRUCT</div> REALとIMAGメンバは、実数型のN件の配列です。		
備 考	構造体の定義は、CAL001. AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001. AJN"」を記述してください。		
使用例	STRUCT CALC_CMPL A A. REAL = [1, 2, 3, 4, 5] A. IMAG = [6, 7, 8, 9, 10] STRUCT CALC_CMPL B B = CALC_IFFTSHIFT_STRUCT(A) ? B. REAL [3, 4, 5, 1, 2] ? B. IMAG [8, 9, 10, 6, 7]		

3. 4. 15 CALC_IFFTSHIFT_2D_STRUCT

関数			
機 能	CALC_IFFTSHIFT_2D_STRUCTの逆変換です。 0周波数成分が配列の中心になるように入れ替えた2次元配列を元に戻します。(構造体版)		
書 式	<(戻り値)入れ替えた構造体> = CALC_IFFTSHIFT_2D_STRUCT (<①入力情報の構造体>)		
戻り値	戻り値	<入れ替えた構造体>	構造体
	0周波数成分が配列の外側になるように入れ替えた2次元配列を持つ構造体 that 得られます。 入力情報と同じ形式の構造体 that 得られます。		
パラメータ	①	<入力情報の構造体>	構造体
	0周波数成分が配列の中心にある、FFT演算した2次元配列をメンバに持つ構造体を指定します。 以下の定義を持つ構造体を指定します。		
	<pre> DEFINE STRUCT CALC_CMPL LIST REAL '実部 LIST IMAG '虚部 END STRUCT </pre>		
	REALとIMAGメンバはどちらも実数型のN×Mの2次元配列です。 N,Mはそれぞれ画像の縦方向の画素数と横方向の画素数です。		
備 考	構造体の定義は、CAL001.AJNにて定義されています。 プログラムの先頭で「INCLUDE "CAL001.AJN"」を記述してください。		
使用例	<pre> STRUCT CALC_CMPL A DIM DATA(2, 2) DATA = [[1, 2, 3], [4, 5, 6], [7, 8, 9]] A. REAL = DATA DATA = [[10, 11, 12], [13, 14, 15], [16, 17, 18]] A. IMAG = DATA STRUCT CALC_CMPL B B = CALC_IFFTSHIFT_2D_STRUCT(A) ? B. REAL [[5, 6, 4], [8, 9, 7], [2, 3, 1]] ? B. IMAG [[14, 15, 13], [17, 18, 16], [11, 12, 10]] </pre>		

3. 4. 16 CALC_CONVOLUTION_1D

関数			
機 能	1次元配列同士の畳み込み積分を計算します。		
書 式	<(戻り値)演算結果> = CALC_CONVOLUTION_1D(<①1次元配列1>, <①1次元配列2>)		
戻り値	戻り値	<演算結果>	配列
	1次元配列同士の畳み込み積分の演算結果が得られます。 演算結果の配列の長さは、<1次元配列1>, <1次元配列2>をそれぞれA, Bとおくと、 ABS (LDIM(A) - LDIM(B)) + 1 になります。 以下の式を利用します。 $(a * b)[n] = \sum_{m=-\infty}^{\infty} a[m]b[n - m]$ ただし、添字が配列の範囲内になる計算のみ行います。 (添字が配列の範囲外になる要素を0で補間するような処理は行いません。)		
パラ メータ	①	<1次元配列1>, <1次元配列2>	配列
	畳み込み積分を計算する配列を指定します。 整数型および実数型である必要があります。		
使用例	LIST A, B A = [1, 2, 3, 4, 5] B = [0.2, 0.8]		
	LIST C C = CALC_CONVOLUTION_1D(A, B) ? C [1.2, 2.2, 3.2, 4.2] 配列Aと配列Bを畳み込み積分した結果が得られます。		

3. 4. 17 CALC_CONVOLUTION_2D

関数			
機 能	2次元配列同士の畳み込み積分を計算します。		
書 式	<(戻り値)演算結果> = CALC_CONVOLUTION_2D(<①2次元配列1>, <①2次元配列2>)		
戻り値	戻り値	<演算結果>	配列
	<p>2次元配列同士の畳み込み積分の演算結果が得られます。</p> <p>演算結果の配列の大きさは、<2次元配列1>, <2次元配列2>をそれぞれA, Bとおくと、</p> <p>1次元目: $\text{ABS}(\text{LDIM}(A, 1) - \text{LDIM}(B, 1)) + 1$</p> <p>2次元目: $\text{ABS}(\text{LDIM}(A, 2) - \text{LDIM}(B, 2)) + 1$</p> <p>になります。</p> <p>以下の式を利用します。</p> $c[i, j] = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} a[n, m] b[i-n, j-m]$ <p>ただし、添字が配列の範囲内になる式の計算のみ行います。 (0で補間するような処理は行いません。)</p>		
パラ メータ	①	<2次元配列1>, <2次元配列2>	配列
	<p>畳み込み積分を計算する配列を指定します。</p> <p>整数型および実数型である必要があります。</p>		
使用例	<p>DIM A(2, 2)</p> <p>DIM B(1, 1)</p> <p>A = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]</p> <p>B = [[0.1, 0.2], [0.3, 0.4]]</p> <p>DIM C(2, 2)</p> <p>C = CALC_CONVOLUTION_2D(A, B)</p> <p>? C</p> <p>[[2.3, 3.3], [5.3, 6.3]]</p>		

3. 4. 18 CALC_HILBERT_ENVELOPE

関数			
機 能	ヒルベルト変換を使用して、1次元配列で与えられた信号の包絡線を求めます。		
書 式	<(戻り値)演算結果> = CALC_HILBERT_ENVELOPE (<①1次元配列>)		
戻り値	戻り値	<演算結果>	配列
	1次元配列で与えられた信号の包絡線が得られます。		
パラ メータ	①	<1次元配列>	配列
	包絡線を求める1次元配列の信号を指定します。		
使用例	LIST A CALL MAKE_WAVE() ’ この呼び出しで、配列Aに波形データを生成したと仮定します。 LIST B B = CALC_HILBERT_ENVELOPE(A) ’ 配列Aで与えられる信号の包絡線を得ます。		

3. 4. 19 CALC_AVERAGE

関数		
機 能	配列の平均値を求めます。	
書 式	<(戻り値)平均値> = CALC_AVERAGE(<①入力配列> [, <②次元>])	
戻り値	戻り値	<平均値>
	数値、配列	
パラ メータ	求めた平均値が得られます。	
	次元を指定しなかった場合、単一の数値が得られます。	
	次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。	
	①	<入力配列>
	配列	
	平均値を求める配列を指定します。配列の次元数に制限はありません。	
	②	<次元>
	数値	
	平均値を求める配列の次元を指定します。	
	次元を指定しなかった場合、配列の全ての要素で平均値を計算します。	
備 考	・ 存在しない次元を指定するとエラーが返ります。	
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_AVE = CALC_AVERAGE(X) ? X_AVE 3	

3. 4. 20 CALC_VARIANCE

関数			
機 能	配列の分散を求めます。		
書 式	<(戻り値)分散> = CALC_VARIANCE(<①入力配列> [, <②次元>])		
戻り値	戻り値	<分散>	数値、配列
	求めた分散が得られます。 次元を指定しなかった場合、単一の数値が得られます。 次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。		
パラ メータ	①	<入力配列>	配列
	分散を求める配列を指定します。配列の次元数に制限はありません。		
	②	<次元>	数値
	分散を求める配列の次元を指定します。 次元を指定しなかった場合、配列の全ての要素で分散を計算します。		
	・存在しない次元を指定するとエラーが返ります。		
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_VAR = CALC_VARIANCE(X) ? X_ VAR 2		

3. 4. 21 CALC_STANDARD_DEVIATION

関数			
機 能	配列の標準偏差を求めます。		
書 式	<(戻り値)標準偏差> = CALC_STANDARD_DEVIATION(<①入力配列> [, <②次元>])		
戻り値	戻り値	<標準偏差>	数値、配列
	<p>求めた標準偏差が得られます。</p> <p>次元を指定しなかった場合、単一の数値が得られます。</p> <p>次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。</p>		
パラ メータ	①	<入力配列>	配列
	標準偏差を求める配列を指定します。配列の次元数に制限はありません。		
	②	<次元>	数値
使用例	標準偏差を求める配列の次元を指定します。		
	次元を指定しなかった場合、配列の全ての要素で標準偏差を計算します。		
備 考	・存在しない次元を指定するとエラーが返ります。		
使用例	DIM X(4)		
	X = [1, 2, 3, 4, 5]		
	X_STDEV = CALC_STANDARD_DEVIATION(X)		
	? X_STDEV		
	1.4142135623731		

3. 4. 22 CALC_THIRD_MOMENT

関数		
機 能	配列の3次モーメントを求めます。	
書 式	<(戻り値)3次モーメント> = CALC_THIRD_MOMENT(<①入力配列> [, <②次元>])	
戻り値	戻り値	<3次モーメント>
	数値、配列	
パラ メータ	求めた3次モーメントが得られます。	
	次元を指定しなかった場合、単一の数値が得られます。	
	次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。	
	①	<入力配列>
パラ メータ	配列	
	3次モーメントを求める配列を指定します。配列の次元数に制限はありません。	
	②	<次元>
パラ メータ	数値	
	3次モーメントを求める配列の次元を指定します。	
備 考	次元を指定しなかった場合、配列の全ての要素で3次モーメントを計算します。	
	・ 存在しない次元を指定するとエラーが返ります。	
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_MU_3 = CALC_THIRD_MOMENT(X) ? X_MU_3 0	

3. 4. 23 CALC_FOURTH_MOMENT

関数		
機 能	配列の4次モーメントを求めます。	
書 式	<(戻り値)4次モーメント> = CALC_FOURTH_MOMENT (<①入力配列> [, <②次元>])	
戻り値	戻り値	<4次モーメント>
	数値、配列	
パラ メータ	求めた4次モーメントが得られます。	
	次元を指定しなかった場合、単一の数値が得られます。	
	次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。	
	①	<入力配列>
パラ メータ	配列	
	4次モーメントを求める配列を指定します。配列の次元数に制限はありません。	
	②	<次元>
パラ メータ	数値	
	4次モーメントを求める配列の次元を指定します。	
備 考	次元を指定しなかった場合、配列の全ての要素で4次モーメントを計算します。	
	・存在しない次元を指定するとエラーが返ります。	
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_MU_4 = CALC_FOURTH_MOMENT(X) ? X_MU_4 6.8	

3. 4. 24 CALC_RMS

関数		
機 能	配列の2乗平均平方根を求めます。	
書 式	<(戻り値)2乗平均平方根> = CALC_RMS(<①入力配列> [, <②次元>])	
戻り値	戻り値	<2乗平均平方根> 数値、配列
	求めた2乗平均平方根が得られます。 次元を指定しなかった場合、単一の数値が得られます。 次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。	
パラ メータ	①	<入力配列> 配列
	2乗平均平方根を求める配列を指定します。配列の次元数に制限はありません。	
	②	<次元> 数値
	2乗平均平方根を求める配列の次元を指定します。 次元を指定しなかった場合、配列の全ての要素で2乗平均平方根を計算します。	
備 考	・ 存在しない次元を指定するとエラーが返ります。	
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_RMS = CALC_RMS(X) ? X_RMS 3.3166247903554	

3. 4. 25 CALC_SKEWNESS

関数		
機 能	与えられた入力配列の歪度を求めます。 歪度とは、与えられたデータ列の分布が正規分布からどれだけ歪んでいるかを示す統計量で、左右対称正を示す指標のことです。	
書 式	<(戻り値)歪度> = CALC_SKEWNESS(<①入力配列> [, <②次元>])	
戻り値	戻り値	<歪度>
	数値、配列	
パラ メータ	求めた歪度が得られます。	
	次元を指定しなかった場合、単一の数値が得られます。	
	次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。	
	①	<入力配列>
	歪度を求める配列を指定します。配列の次元数に制限はありません。	
	②	<次元>
	歪度を求める配列の次元を指定します。	
	次元を指定しなかった場合、配列の全ての要素で歪度を計算します。	
備 考	・ 存在しない次元を指定するとエラーが返ります。	
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_SKEWNESS = CALC_SKEWNESS(X) ? X_SKEWNESS 0	

3. 4. 26 CALC_KURTOSIS

関数			
機 能	与えられた入力配列の尖度を求めます。 尖度とは、与えられたデータ列の分布が正規分布からどれだけ尖っているかを示す統計量で、山の尖り度と裾の広がり度を示す指標のことです。		
書 式	<(戻り値)尖度> = CALC_KURTOSIS(<①入力配列> [, <②次元>])		
戻り値	戻り値	<尖度>	数値、配列
	求めた尖度が得られます。 次元を指定しなかった場合、単一の数値が得られます。 次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。		
パラ メータ	①	<入力配列>	配列
	尖度を求める配列を指定します。配列の次元数に制限はありません。		
	②	<次元>	数値
	尖度を求める配列の次元を指定します。 次元を指定しなかった場合、配列の全ての要素で尖度を計算します。		
備 考	・存在しない次元を指定するとエラーが返ります。		
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_KURTOSIS = CALC_KURTOSIS(X) ? X_KURTOSIS 1.7		

3. 4. 27 CALC_PEAK_VALUE

関数			
機 能	与えられた入力配列の最大の絶対値(ピーク値)を求めます。		
書 式	<(戻り値)ピーク値> = CALC_PEAK_VALUE(<①入力配列> [, <②次元>])		
戻り値	戻り値	<ピーク値>	数値、配列
	<p>求めたピーク値が得られます。</p> <p>次元を指定しなかった場合、単一の数値が得られます。</p> <p>次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。</p>		
パラ メータ	①	<入力配列>	配列
	ピーク値を求める配列を指定します。配列の次元数に制限はありません。		
	②	<次元>	数値
使用例	<p>ピーク値を求める配列の次元を指定します。</p> <p>次元を指定しなかった場合、配列の全ての要素でピーク値を計算します。</p>		
	<p>・存在しない次元を指定するとエラーが返ります。</p>		
使用例	DIM X(3)		
	X = [2, 5, -8, 3]		
	X_PEAK = CALC_PEAK_VALUE(X)		
	? X_PEAK		
使用例	8		

3. 4. 28 CALC_CREST_FACTOR

関数			
機 能	配列の波効率を求めます。 与えられたデータ列のピーク値をそのデータ列の実効値で割った値を計算します。		
書 式	<(戻り値)波効率> = CALC_CREST_FACTOR(<①入力配列> [, <②次元>])		
戻り値	戻り値	<波効率>	数値、配列
	求めた波効率が得られます。 次元を指定しなかった場合、単一の数値が得られます。 次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。		
パラ メータ	①	<入力配列>	配列
	波効率を求める配列を指定します。配列の次元数に制限はありません。		
	②	<次元>	数値
	波効率を求める配列の次元を指定します。 次元を指定しなかった場合、配列の全ての要素で波効率を計算します。		
備 考	・ 存在しない次元を指定するとエラーが返ります。		
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_CREST_FACTOR = CALC_CREST_FACTOR(X) ? X_CREST_FACTOR 1. 50755672288882		

3. 4. 29 CALC_ABSOLUTE_AVERAGE

関数			
機 能	配列の絶対値の平均値を求めます。		
書 式	<(戻り値)絶対値の平均値> = CALC_ABSOLUTE_AVERAGE (<①入力配列> [, <②次元>])		
戻り値	戻り値	<絶対値の平均値>	数値、配列
	<p>求めた絶対値の平均値が得られます。</p> <p>次元を指定しなかった場合、単一の数値が得られます。</p> <p>次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。</p>		
パラ メータ	①	<入力配列>	配列
	絶対値の平均値を求める配列を指定します。配列の次元数に制限はありません。		
	②	<次元>	数値
	絶対値の平均値を求める配列の次元を指定します。		
	次元を指定しなかった場合、配列の全ての要素で絶対値の平均値を計算します。		
備 考	・ 存在しない次元を指定するとエラーが返ります。		
使用例	DIM X(4) X = [-1, 2, -4, 3, 8] X_ABS_AVE = CALC_ABSOLUTE_AVERAGE(X) ? X_ABS_AVE 3.6		

3. 4. 30 CALC_SHAPE_FACTOR

関数		
機 能	配列の形状係数を求めます。 与えられたデータ列の実効値をそのデータ列の絶対値の平均で割った値を計算します。	
書 式	<(戻り値)形状係数> = CALC_SHAPE_FACTOR(<①入力配列> [, <②次元>])	
戻り値	戻り値	<形状係数>
	数値、配列	
パラ メータ	求めた形状係数が得られます。	
	次元を指定しなかった場合、単一の数値が得られます。	
	次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。	
	①	<入力配列>
	形状係数を求める配列を指定します。配列の次元数に制限はありません。	
	②	<次元>
	形状係数を求める配列の次元を指定します。	
	次元を指定しなかった場合、配列の全ての要素で形状係数を計算します。	
備 考	・ 存在しない次元を指定するとエラーが返ります。	
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_SHAPE_FACTOR = CALC_SHAPE_FACTOR(X) ? X_SHAPE_FACTOR 1.10554159678513	

3.4.31 CALC_CLEARANCE_FACTOR

関数		
機 能	配列のクリアランス率を求めます。 与えられたデータ列のピーク値を絶対値の平均の2乗で割った値を計算します。	
書 式	<(戻り値)クリアランス率> = CALC_CLEARANCE_FACTOR(<①入力配列> [, <②次元>])	
戻り値	戻り値	<クリアランス率>
	数値、配列	
パラ メータ	求めたクリアランス率が得られます。	
	次元を指定しなかった場合、単一の数値が得られます。	
	次元を指定した場合、入力配列の次元数 - 1の次元数の配列が得られます。	
	①	<入力配列>
	クリアランス率を求める配列を指定します。配列の次元数に制限はありません。	
	②	<次元>
	クリアランス率を求める配列の次元を指定します。	
	次元を指定しなかった場合、配列の全ての要素でクリアランス率を計算します。	
備 考	・ 存在しない次元を指定するとエラーが返ります。	
使用例	DIM X(4) X = [1, 2, 3, 4, 5] X_CLEARANCE_FACTOR = CALC_CLEARANCE_FACTOR(X) ? X_CLEARANCE_FACTOR 0.5555555555555556	

3. 4. 32 CALC_IMPULSE_INDICATOR

関数		
機 能	配列のインパルスインジケータを求めます。 与えられたデータ列のピーク値をそのデータ列の平均レベルで割った値を計算します。	
書 式	〈(戻り値)インパルスインジケータ〉 = CALC_IMPULSE_INDICATOR(〈①入力配列〉, 〈②平均レベル〉 [, 〈③次元〉])	
戻り値	戻り値	

3. 4. 33 CALC_COVARMAT

関数			
機 能	入力情報の配列を元に、共分散行列を求めます。		
書 式	<(戻り値)演算結果> = CALC_COVARMAT(<①入力情報の配列> [, <②追加の入力情報の配列>])		
戻り値	戻り値	<演算結果>	数値
	入力情報の配列を元に、共分散行列の演算結果が得られます。		
パラメータ	①	<入力情報の配列>	配列
	共分散行列演算を行う、入力情報の入った配列を指定します。 実数型の1次元または2次元の配列を指定してください。		
	②	<追加の入力情報の配列>	配列
備考	入力情報の配列に対して追加したい配列を指定します。 実数型の配列を指定してください。		
	<ul style="list-style-type: none"> ・演算時、2次元配列を行列として扱います。 ・本関数は、下位ライブラリとして numpy の cov を使用しています。 		
使用例	<pre>' ARY1とARY2の共分散行列値を求めます。 LIST ARY1, ARY2 ARY1 = [1; 2; 3; 5] ARY2 = [5; 4; 2; 1] PRINT CALC_COVARMAT(ARY1, ARY2) ' 以下のような結果が得られます ' [[1.666..., -2.333...], [-2.333..., 3.333...]]</pre>		

3. 4. 34 CALC_MULMAT

関数			
機 能	2つの入力情報の配列同士を行列として積を求めます。		
書 式	<(戻り値)演算結果> = CALC_MULMAT(<①入力情報の配列1>, <①入力情報の配列2>)		
戻り値	戻り値	<演算結果>	数値
	入力情報の配列同士を行列として積を求めます。		
パラメータ	①	<入力情報の配列1>, <入力情報の配列2>	配列
	行列演算を行う、入力情報の入った配列を指定します。 実数型の配列を指定してください。		
備考	<ul style="list-style-type: none"> ・演算時、2次元配列を行列として扱います。 ・本関数は、下位ライブラリとして numpy の matmul を使用しています。 		
使用例	<pre>' ARY1とARY2の積を求めます。 DIM ARY1(1, 1), ARY2(1, 1) ARY1 = [[2; 0], [1; 2]] ARY2 = [[4; 1], [2; 2]] PRINT CALC_MULMAT(ARY1, ARY2) ' 以下のような結果が得られます ' [[8, 2], [8, 5]]</pre>		

3. 4. 35 CALC_INVMAT

関数			
機 能	入力情報の配列を元に、逆行列を求めます。		
書 式	<(戻り値)演算結果> = CALC_INVMAT(<①入力情報の配列>)		
戻り値	戻り値	<演算結果>	数値
	入力情報の配列を元に、逆行列の演算結果が得られます。		
パラ メータ	①	<入力情報の配列>	配列
	逆行列演算を行う、入力情報の入った配列を指定します。 実数型の2次元の配列を指定してください。		
備 考	<ul style="list-style-type: none"> ・演算時、2次元配列を行列として扱います。 ・正則行列以外を逆行列にしたい場合「CALC_PSEINVMAT」が利用できます。 ・本関数は、下位ライブラリとして numpy の inv を使用しています。 		
使用例	<pre>' ARYの逆行列値を求めます。 DIM ARY(1,1) ARY = [[1; 2], [3; 4]] PRINT CALC_INVMAT(ARY) ' 以下のような結果が得られます ' [[-2, 1], [1.5, -0.5]]</pre>		

3. 4. 36 CALC_PSEINVMAT

関数			
機 能	入力情報の配列を元に、擬似逆行列を求めます。		
書 式	<(戻り値)演算結果> = CALC_PSEINVMAT(<①入力情報の配列>)		
戻り値	戻り値	<演算結果>	数値
	入力情報の配列を元に、(ムーア・ペンローズ)擬似逆行列の演算結果が得られます。		
パラ メータ	①	<入力情報の配列>	配列
	擬似逆行列演算を行う、入力情報の入った配列を指定します。 実数型の2次元の配列を指定してください。		
備 考	<ul style="list-style-type: none"> ・演算時、2次元配列を行列として扱います。 ・正則行列を逆行列にしたい場合「CALC_INVMAT」が利用できます。 ・本関数は、下位ライブラリとして numpy の pinv を使用しています。 		
使用例	<pre>' ARYの擬似逆行列値を求めます。 DIM ARY(2,1) ARY = [[1; 2], [3; 4], [5; 6]] PRINT CALC_PSEINVMAT(ARY) ' 以下のような結果が得られます ' [[-1.333..., -0.333..., 0.666...], [1.0833..., 0.333..., -0.4166...]]</pre>		

3. 4. 37 CALC_MAHALANOBIS

関数			
機 能	2つの入力情報配列の間の、マハラノビス距離を求めます。		
書 式	<(戻り値)演算結果> = CALC_MAHALANOBIS(<①入力情報の配列1>, <①入力情報の配列2>, <②共分散行列の逆行列の値>)		
戻り値	戻り値	<演算結果>	数値
	入力情報の配列を元に、共分散行列の演算結果が得られます。		
パラ メータ	①	<入力情報の配列1>, <入力情報の配列2>	配列
	マハラノビス距離を求める、入力情報の入った配列を2つ指定します。 実数型の1次元配列を指定してください。		
	②	<共分散行列の逆行列の値>	配列
	マハラノビス距離を求める際に用いる引数として、共分散行列の逆行列を指定します。 実数型の2次元配列を指定してください。		
備 考	<ul style="list-style-type: none"> ・演算時、2次元配列を行列として扱います。 ・本関数は、下位ライブラリとして scipy の mahalanobis を使用しています。 		
使用例	<pre> ' ARY1とARY2のマハラノビス距離を求めます。 DIM IV(2, 2) IV = [[1; 0.5; 0.5], [0.5; 1; 0.5], [0.5; 0.5; 1]] LIST ARY1, ARY2 ARY1 = [2; 0; 0] ARY2 = [0; 1; 0] PRINT CALC_MAHALANOBIS(ARY1, ARY2, IV) ' 以下のような結果が得られます ' 1.73205...</pre>		

第4章 サンプルプログラム

AJANのサンプルプログラムについて記載します。

サンプルプログラムは「/usr/share/interface/AJANPro/samples/CAL/」に格納されています。

AJAN統合開発環境を起動すると、左ペインのエクスプローラウィンドウ内の「Samples/CAL/」に、ファイルが取り込まれて配置されます。

4.1 サンプルプログラム

#	ファイル名	内容
CALC		
1	CALC_FFT_IFFT_EX_STRUCT. AJN	FFT / 逆 FFT 演算のサンプルプログラムです。 波形データを元に、CALC_FFT_EX_STRUCT 関数で FFT 演算し、 CALC_IFFT_EX_STRUCT 関数で逆 FFT 演算します。 演算結果を GUI 機能を使って波形グラフ表示します。
2	CALC_ANOMALY_LEARN_SCORE. AJN	異常度の学習および状態判定のサンプルプログラムです。 CALC_ANOMALY_LEARN 関数で、異常度および異常度最大値の 学習を行い、CALC_ANOMALY_SCORE 関数で、新たに観測した データの異常度の計算を行います。 GUI コマンドと組み合わせて、簡単なグラフプロットを行っ ています。
3	CALC_REGLINE_REGPRED. AJN	回帰直線および予測を行うサンプルプログラムです。 CALC_REGLINE 関数で、2つのデータ列から回帰直線となる傾 きと切片を求めます。 CALC_REGPRED 関数で、CALC_REGLINE 関数で求めた傾きと切 片から予測値を求めるます。 GUI コマンドと組み合わせて、簡単なグラフプロットを行っ ています。
4	CALC_XBARR_LEARN_JUDGE. AJN	Xbar-R 管理図の学習および状態判定のサンプルプログラム です。 CALC_XBARR_LEARN 関数で、Xbar-R 管理図用の学習を行いま す。 CALC_XBARR_JUDGE 関数で、CALC_XBARR_LEARN 関数で求めた 学習値を参考に、判定処理を行います。 GUI コマンドと組み合わせて、簡単なグラフプロットを行いま す。

第5章 索引

C		CALC_IFFT_2D_EX	48
CALC_CMPL2ABS	23	CALC_IFFT_2D_EX_STRUCT	49
CALC_ABSOLUTE_AVERAGE	71	CALC_IFFT_EX	21
CALC_ANOMALY_LEARN	39	CALC_IFFT_EX_STRUCT	22
CALC_ANOMALY_SCORE	40	CALC_IFFTSHIFT	54
CALC_AVERAGE	61	CALC_IFFTSHIFT_2D	55
CALC_CLEARANCE_FACTOR	73	CALC_IFFTSHIFT_2D_STRUCT	57
CALC_CONVOLUTION_1D	58	CALC_IFFTSHIFT_STRUCT	56
CALC_CONVOLUTION_2D	59	CALC_IMPULSE_INDICATOR	74
CALC_CORREL	27	CALC_INVMAT	76
CALC_COVAR	27	CALC_KURTOSIS	68
CALC_COVARMAT	75	CALC_L1_NORMALIZE	44
CALC_CREATE_COMBINEWAVE	33	CALC_L2_NORMALIZE	45
CALC_CREATE_FAKENOISE	37	CALC_MAHALANOBIS	77
CALC_CREATE_SAWTOOTHWAVE	36	CALC_MINMAX_NORMALIZE	43
CALC_CREATE_SINWAVE	32	CALC_MODE	26
CALC_CREATE_SQUAREWAVE	35	CALC_MULMAT	75
CALC_CREATE_TRIANGLEWAVE	34	CALC_PEAK_VALUE	69
CALC_CREST_FACTOR	70	CALC_POLYFIT	24
CALC_FFT_2D_EX	46	CALC_PSEINVMAT	76
CALC_FFT_2D_EX_STRUCT	47	CALC_REGLINE	41
CALC_FFT_EX	19	CALC_REGPRED	42
CALC_FFT_EX_STRUCT	20	CALC_RMS	66
CALC_FFTSHIFT	50	CALC_SHAPE_FACTOR	72
CALC_FFTSHIFT_2D	51	CALC_SKEWNESS	67
CALC_FFTSHIFT_2D_STRUCT	53	CALC_STANDARD_DEVIATION	63
CALC_FFTSHIFT_STRUCT	52	CALC_THIRD_MOMENT	64
CALC_FOURTH_MOMENT	65	CALC_VARIANCE	62
CALC_HILBERT_ENVELOPE	60	CALC_XBARR_JUDGE	30
CALC_HISTOGRAM	25	CALC_XBARR_LEARN	28

第6章 重要な情報

保証の内容と制限

弊社は本ドキュメントに含まれるソースプログラムの実行が中断しないこと、またはその実行に誤りが無いことを保証していません。

本製品の品質や使用に起因する、性能に起因するいかなるリスクも使用者が負うものとします。

弊社はドキュメント内の情報の正確さに万全を期しています。万一、誤記または誤植などがあつた場合、弊社は予告無く改訂する場合があります。ドキュメントまたはドキュメント内の情報に起因するいかなる損害に対しても弊社は責任を負いません。

ドキュメント内の図や表は説明のためであり、ユーザ個別の応用事例により変化する場合があります。

著作権、知的所有権

弊社は本製品に含まれるおよび本製品に対する権利や知的所有権を保持しています。

本製品はコンピュータ ソフトウェア、映像/音声(例えば図、文章、写真など)を含んでいます。

医療機器/器具への適用における注意

弊社の製品は人命に関わるような状況下で使用される機器に用いられる事を目的として設計、製造された物では有りません。

弊社の製品は人体の検査などに使用するに適する信頼性を確保する事を意図された部品や検査機器と共に設計された物では有りません。

医療機器、治療器具などの本製品の適用により、製品の故障、ユーザ、設計者の過失などにより、損傷/損害を引き起こす場合が有ります。

複製の禁止

弊社の許可なく、本ドキュメントの全て、または一部に関わらず、複製、改変などを行うことはできません。

責任の制限

弊社は、弊社または再販売者の予見の有無にかかわらず発生したいかなる特別損害、偶発的損害、間接的な損害、重大な損害について、責任を負いません。

本製品(ハードウェア、ソフトウェア)のシステム組み込み、使用、ならびに本製品から得られる結果に関する一切のリスクについては、本製品の使用者に帰属するものとします。

本製品に含まれる不都合、あるいは本製品の供給(納期遅延)、性能もしくは使用に起因する付帯的損害もしくは間接的損害に対して、弊社に全面的に責がある場合でも、弊社はその製品に対する改良(有償サービスの利用)、代品交換までとし、製品の予防交換並びに、代金減額等、金銭面での賠償の責任は負わないものとします。

本製品は、日本国内仕様です。

商標/登録商標

本書に掲載されている会社名、製品名は、それぞれ各社の商標または登録商標です。

改訂履歴

Ver.	年 月	改 訂 内 容
0.90	2019年10月	新規作成
1.00	2022年1月	最新情報に更新
1.10	2023年3月	「数学統計コマンド(追加分)」を追加。

このマニュアルは、製品の改良その他により将来予告なく改訂しますので、予めご了承ください。