M a t h e m a t i c a
コマンドリファレンス

吉田 誠一 (S. Yoshida)

ver. 4.0 （1994-11-14)

参考文献

[1] *Mathematica* 数式処理入門, 小池慎一,1990, 技術評論社

[2] *Mathematica* 微積分入門

目次

基本関数

一般的なオプション

AccuracyGoal	–> 結果の桁数（def： 6）
WorkingPrecision	–> 内部演算での精度

定数

Pi	π
E	e
I	i
Infinity	∞
GoldenRatio	黄金分割比 $\frac{1+\sqrt{5}}{2}$
True	真
False	偽

数値関数

N$[a]$	a の近似値
N$[a, n]$	a を n 桁で表示
Precision$[a]$	桁数
Accuracy$[a]$	桁数（小数点以下）
Round$[a]$	a に最も近い整数
Floor$[a]$	a を越えない最大の整数
Ceiling$[a]$	a より小さくない最小の整数
Sign$[a]$	符号
Abs$[a]$	絶対値
Max$[a_1, a_2, \ldots]$	最大値
Min$[a_1, a_2, \ldots]$	最小値
Rationalize$[a, dx]$	小数 a を精度 dx で有理化（分数化）、 dx は省略可能

整数関数

$n!$	階乗
$n!!$	$n(n-2)(n-4)\cdots$
Prime$[k]$	k 番目の素数
Quotient$[m,n]$	$\frac{m}{n}$ の整数部
GCD$[n_1,n_2,\ldots]$	最大公約数
LCM$[n_1,n_2,\ldots]$	最小公倍数
FactorInteger$[n]$	n の素因数分解のリスト
Divisors$[n]$	n の約数のリスト
DivisorSigma$[k,n]$	n の約数の k 乗の和
Mod$[k,n]$	$k \bmod n$
PowerMod$[a,b,n]$	$a^b \bmod n$
ExtendedGCD$[m,n]$	不定方程式を解くのに使う
	$\{g,\{r,s\}\}$ s.t. $g=(m,n)$, $g=rm+sn$
EulerPhi$[n]$	オイラーの ϕ 関数、 $1 \le m \le n$に対して$(m,n)=1$となるmの個数
Binomial$[n,m]$	二項係数 ${}_n\mathrm{C}_m = \frac{n!}{m!(n-m)!}$
Multinomial$[n_1,n_2,\ldots]$	多項係数 $\frac{(n_1+n_2+\cdots)!}{n_1!n_2!\cdots}$
PartitionsP$[n]$	分割数 $p(n)$

n をいくつかの正の整数の和で表した時のパターン数
例： PartitionsP[5] $\rightarrow$ 7 （5,4+1,3+2,3+1+1,2+2+1,2+1+1+1,1+1+1+1+1）

PartitionsQ$[n]$	分割数 $q(n)$

n をいくつかの正の整数の和で表した時のパターン数。和の中に同じ数が複数回現れてはならない。
例： PartitionsP[5] $\rightarrow$ 3 （5,4+1,3+2）

複素関数

Re$[z]$	実部
Im$[z]$	虚部
Conjugate$[z]$	共役

Abs[z]	絶対値
Arg[z]	偏角（$z = \|z\|e^{i\phi}$ なる ϕ）

真理関数

Positive[a]	$a > 0$ なら True
NonNegative[a]	$a \geq 0$ なら True
Negative[a]	$a < 0$ なら True
NumberQ[a]	a が数値なら True
PrimeQ[a]	a が素数なら True
IntegerQ[a]	a が整数なら True
OddQ[a]	a が奇数なら True
EvenQ[a]	a が偶数なら True
PolynomialQ[式, 変数]	式が変数の多項式なら True
MemberQ[リスト, a]	a がリストの要素なら True
FreeQ[リスト, a]	a がリスト中になければ True
VectorQ[リスト]	ベクトルなら True
MatrixQ[リスト]	行列なら True

論理関数

&&,||,!,==,!=,<,>	C 言語と同じ
Xor[a, b]	a xor b
Implies[a, b]	$!a\|\|b$

乱数関数

Random[ ]	区間 [0,1] の乱数
Random[Real, max]	[0,max]
Random[型, {min, max}]	型は Real,Complex,Integer
Random[型, {min, max}, n]	n 桁の精度の乱数
Random[Integer]	(1,0) を $\frac{1}{2}$ の確率で発生

SeedRandom[ ]	日付、時刻によるシード
SeedRandom[n]	整数 n によるシード

表現関数

Simplify[式]	表示を簡単にする
Short[式, n]	n 行で短縮表示（n は省略可能）
TableForm[リスト]	リストを縦に表示
MatrixForm[M]	行列表現
NumberForm[a]	数値表示のフォーマット

オプション

DigitBlock	–> 数のひとかたまりの大きさ（def：全桁）
Numberseparator	–> ブロック間の区切り文字（def：,）
Numberpoint	–> 小数点（def：.）
ExponentStep	–> 指数形式の小数点以上の桁数（def： 1）

EngineeringForm[a]	工学的表現（小数点以上が 3 桁）

数式関数

式 [[n]]	第 n 項
Length[式]	項数
Coefficient[式, 項]	係数
CoefficientList[式, 変数のリスト]	係数のリスト

CoefficientList[f,{x,y}] $\rightarrow$ $\mathbf{C}$ s.t. $f = \mathbf{x}\mathbf{C}\mathbf{y}^t$
($\mathbf{x} = (1, x, x^2, \ldots)$, $\mathbf{y} = (1, y, y^2, \ldots)$)

Variables[式]	式中の変数のリスト
Exponent[式, 変数]	変数に関する最大次数
Collect[式, 変数]	式を変数について並べ変え
Factor[式]	因数分解
Expand[式]	展開

Decompose[式, 変数]	変数についての多項式を、多項式の多項式として表わす
PolynomialQuotient[式$_1$, 式$_2$, 変数]	変数に関して 式$_1$÷式 $_2$ を行い、商を返す
PolynomialReminder[式$_1$, 式$_2$, 変数]	剰余を返す
ExpandPower[式]	$\sqrt{\ }$ の中身を正として展開
TrigExpand[式]	三角関数のまとめ（$\sin^2 x + \cos^2 x = 1$ など）
TrigReduce[式]	三角関数の整理
Fit[座標リスト, 項リスト, 変数]	近似式を求める

例： Fit[{{0,0},{1,2},{10,19}},{1,x},x] → その 3 点をよく表す 1 次近似式を求める

方程式

Solve[式, 変数]	方程式を解く
Reduce[式, 変数]	方程式を解く（場合分けあり）
NRoots[方程式, 変数]	方程式の近似解を求める
LinearSolve[係数行列, 解ベクトル]	連立一次方程式を解く

例： LinearSolve[{{1,2},{3,4}},{5,6}] → $\begin{cases} x + 2y = 5 \\ 3x + 4y = 6 \end{cases}$

FindRoot[方程式, {変数, 初期値}]	ニュートン法で方程式の近似解を求める（反復は 15 回まで）
FixedPoint[f, x_1, n]	$f(x) = x$ の解を $x = x_1$ から不動点反復法で求める
	（反復回数の上限 n は省略可能）
FixedPointList[f, x_1, n]	不動点反復法による解を反復回数ごとのリストで返す

有理関数（分数関数）

Together[式]	通分してまとめる
Apart[式]	分母を最小にするような有理関数に分解
Numerator[式]	分子を取り出す

Denominator[式]	分母を取り出す
ExpandNumerator[式]	分子を展開
ExpandDenominator[式]	分母を展開
ExpandAll[式]	分子、分母共展開

リスト、集合

リスト [[n]]	n[1]番目の要素
リスト [[m, n]]	m 番目の要素の中の n 番目の要素
リスト [[$\{m, n\}$]]	m 番目から n 番目の要素
リスト [[Range[m,n]]]	m 番目と n 番目の間の要素
First[リスト]	先頭の要素
Last[リスト]	末尾の要素
Take[リスト, $\{m, n\}$]	リスト [[$\{m$,$n\}$]] と同じ
Prepend[リスト, a]	先頭に a を追加
Append[リスト, a]	末尾に a を追加
Insert[リスト, a, n]	n 番目に a を挿入
Rest[リスト]	先頭の要素を除去
Drop[リスト, n]	先頭から n 個の要素を除去
Length[リスト]	要素数
Count[リスト, a]	リスト中に a がいくつあるか
Position[リスト, a]	リスト中の a の位置
Range[リスト]	リスト中の最大値と最小値の差
Reverse[リスト]	逆順に並べる
Sort[リスト]	ソート
RotateLeft[リスト, n]	n 個左回転
RotateRight[リスト, n]	n 個右回転
Join[リスト$_1$, リスト$_2$]	リストの結合
Union[リスト$_1$, リスト$_2$]	和集合

[1] $-n$ とすると、末尾から数えて n 番目になる。

Intersection[リスト$_1$, リスト$_2$]	共通集合
Complement[リスト$_1$, リスト$_2$]	差集合、 リスト$_1$ の要素で リスト$_2$ に含まれないもの
Dimensions[リスト]	次元数
Partition[リスト, n]	n 個の要素から成るサブリストのリストにする
Flatten[リスト]	１次元のリストにする
Flatten[リスト, n]	$n+1$ 次元のリストにする
Select[リスト, 条件]	条件を満たす要素を取り出す
	条件の例： 0 以外 $\rightarrow$ #[2]!=0&[3]
Range[始め, 終り, 間隔]	連続する要素のリストの作成、間隔は省略可能
	例： Range[x,$x+3$] $\rightarrow$ { x , $1+x$, $2+x$, $3+x$ }
Map[関数, リスト]	各要素に関数を作用させる
Apply[関数, リスト]	各要素を関数でつなぐ
	例： Apply[Plus,{a,b,c}] $\rightarrow a+b+c$

ベクトル、行列

リスト. リスト	内積、行列の積
IdentityMatrix[n]	$n \times n$ の単位行列 E
DiagonalMatrix[リスト]	リストを対角成分とする対角行列
Transpose[M]	転置行列
Inverse[M]	逆行列
Det[M]	行列式
Minors[M, n]	$n \times n$ 部分行列の行列式のリスト
Array[変数, n]	n 次元ベクトルを作る
Array[変数, {m, n}]	$m \times n$ 行列を作る
Table[式, {n}]	式を n 個コピーしたリストを作る

[2] 引数
[3] ピュア関数参照

Table[式, $\{i, i_{\min}, i_{\max}, i_{\rm step}\}, \{\}, \ldots$]
$i = i_{\min}$ から $i_{\max}$ まで $i_{\rm step}$ ごとに変化させていき、その時の式の値を並べてリストを作る
例： Table[n^ 2,{n,4}] → {1,4,9,16}

LatticeReduce[$\{v_1, v_2, \ldots\}$] 整数ベクトル v_i の集合に対する既約基底を返す

数学関数

Plus,Substract,Times,Divide,Power　+ , -, *, / , ^

Sqrt[a]　$\sqrt{a}$

Exp[z]　e^z

Log[z]　$\log_e z$

Log[b, z]　$\log_b z$

Sin[z],Cos[z],Tan[z],Csc[z],Sec[z],Cot[z]　三角関数

Arc ~　$\cdots$ 逆三角関数
~ h　$\cdots$ ハイポボリック関数
Arc ~ h　$\cdots$ 逆ハイポボリック関数

ArcTan[x, y]　$x + yi$ の偏角

微分

$f'[x]$　偏微分（２階以上も可）

D[f, x[4]]　$\frac{\partial}{\partial x} f$

D[$f, \{x, n\}$]　$\frac{\partial^n}{\partial x^n} f$

D[$f, x_1, x_2, \ldots$]　$\frac{\partial}{\partial x_1} \frac{\partial}{\partial x_2} \cdots f$

Dt[f, x[5]]　$\frac{d}{dx} f$

Dt[f]　df

Dt[$f, \{x, n\}$]　$\frac{d^n}{dx^n} f$

Dt[$f, x_1, x_2, \ldots$]　$\frac{d}{dx_1} \frac{d}{dx_2} \cdots f$

Derivative[n][f][x]　D[f,{x,n}] と同じ

[4] 偏微分では、x_i に依存している変数は NonConstants->{ 変数 } で指定
[5] 全微分では、x_i に依存していない変数は Constants->{ 変数 } で指定

$\mathrm{DSolve}[DE, f[x], x]$ 微分方程式を解く

$\mathrm{DSolve}[\{DE, f[0] == a\}, f[x], x]$ 初期値を指定

$\mathrm{NDSolve}[\{DE, f[0] == a\}, f[x], \{x, x_{\min}, x_{\max}\}]$ DE を数値的に解く
さらに $\mathrm{Plot}[\mathrm{Evaluate}[f[x]\ /.\ \%],\{x, x_{\min}, x_{\max}\}];$ とすると解のグラフを描画

積分

$\mathrm{Integrate}[f, x]$ 不定積分 $\int f\,dx$

$\mathrm{Integrate}[f, \{x, x_{\min}, x_{\max}\}, \{y, y_{\min}, y_{\max}\}]$ 定積分 $\int_{x_{\min}}^{x_{\max}} \int_{y_{\min}}^{y_{\max}} f\,dy\,dx$

$\mathrm{NIntegrate}[f, \{x, x_{\min}, x_{\max}\}, \{y, y_{\min}, y_{\max}\}]$ 数値積分

$\mathrm{NIntegrate}[f, \{x, x_{\min}, x_1, x_2, \ldots, x_{\max}\}]$ 数値積分の分割（特異点がある場合等）

$$\left(\int_{x_{\min}}^{x_1} + \int_{x_1}^{x_2} + \cdots + \int^{x_{\max}}\right) f\,dx$$

オプション

MinRecursion –> 積分区間のサブ区間を求める最小の回数（def： 1）

MaxRecursion –> 積分区間のサブ区間を求める最大の回数（def： 6）

SingularityDepth –> 特異点でのサブ区間を求める回数（def： 3）

Points –> 第 1 回目の分割数（def： Automatic）

極限, 級数

$\mathrm{Limit}[f, x- > x_0]$ $x \to x_0$ の時の f の極限

$\mathrm{Series}[f, \{x, x_0, n\}]$ f の $x = x_0$ のまわりでの $(x - x_0)^n$ までの展開

$\mathrm{Series}[f, \{x, x_0, n_x\}, \{y, y_0, n_y\}]$ 先に y について展開

InverseSeries[級数展開された式] 逆関数の級数展開を返す

Normal[式] ハイヤーオーダーの項 0[] を除去

数列

$\mathrm{Sum}[f,\{i,i_{\min},i_{\max},i_{\mathrm{step}}\},\{j,j_{\min},j_{\max},j_{\mathrm{step}}\},\ldots]$ 和 $\sum_{i=i_{\min}}^{i_{\max}}\sum_{j=j_{\min}}^{j_{\max}}\cdots f$

$\mathrm{Product}[f,\{i,i_{\min},i_{\max},i_{\mathrm{step}}\},\{j,j_{\min},j_{\max}\},\ldots]$ 積 $\prod_{i=i_{\min}}^{i_{\max}}\prod_{j=j_{\min}}^{j_{\max}}\cdots f$

$\mathrm{NSum}[f,\{i,i_{\min},i_{\max},i_{\mathrm{step}}\}]$ 数値和

オプション

ExtraTerms -> 外挿に用いる最大の点（def： 11）

Method -> Integrate または Fit （def： Automatic）

Integrate：オイラー・マクローリン法

Fit：ウィンのイプシロン法

Terms -> 外挿前に用いられる項数（def： 15）

$\mathrm{NProduct}[f,\{i,i_{\min},i_{\max},i_{\mathrm{step}}\}]$ 数値積

$\mathrm{ArithmeticGeometricMean}[a,b]$ 算術幾何平均関数

a,b を初期値として、 $a_{n+1}=\frac{a_n+b_n}{2}$, $b_{n+1}=\sqrt{a_nb_n}$ を順次求め、 $a_n=b_n$ なる値を返す

ピュア関数

Function[引数,式]

例： $\mathrm{Function}[x,ax+b]\rightarrow f[x\_]{:=}ax+b$

$\mathrm{Function}[\{x,y\},x+y]\rightarrow f[x\_,y\_]{:=}x+y$

Function[引数,式][代入値]

例： $\mathrm{Function}[x,ax+b][p+q]\rightarrow a(p+q)+b$

式 & Function の省略形

式 &[代入値] Function の省略形

引数

#	引数が 1 つの時
#n	第 n 引数
##	引数列
##n	n 番目からの引数列

例： {Log[#],#^2}&[2] → {Log[2],4}

高等関数

ラプラス変換（$Calculus`LaplaceTransform`$）

LaplaceTransform[式, t, s]	ラプラス変換
InverseLaplaceTransform[式, s, t]	逆ラプラス変換

積分関数

Gamma[z]	ガンマ関数 $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}dt$
	z が正整数なら、$\Gamma(z) = (z-1)!$
Gamma[$a, 0, z$]	第 1 種不完全ガンマ関数 $\int_0^z t^{a-1}e^{-t}dt$
Gamma[z, a]	第 2 種不完全ガンマ関数 $\int_a^\infty t^{z-1}e^{-t}dt$
Gamma[z, a_0, a_1]	一般不完全ガンマ関数 $\int_{a_0}^{a_1} t^{z-1}e^{-t}dt$
Beta[x, y]	ベータ関数 $B(x,y) = \frac{\Gamma(x)\Gamma(y)}{\Gamma(x+y)} = \int_0^1 t^{x-1}(1-t)^{y-1}dt$
Beta[a, x, y]	不完全ベータ関数 $B_a(x,y) = \int_0^a t^{x-1}(1-t)^{y-1}dt$
Pochhammer[a, n]	ポックハマー関数 $(a)_n = \frac{\Gamma(a+n)}{\Gamma a}$
PolyGamma[z]	ディガンマ関数 $\Psi(z) = \frac{d}{dz}\log\Gamma(z) = \frac{\Gamma'(z)}{\Gamma(z)}$
PolyGamma[n, z]	ポリガンマ関数 $\Psi^{(n)}(z) = \frac{d^n}{dz^n}\Psi(z)$
ExpIntegralEi[z]	指数積分関数 $\mathrm{E}_i(z) = -\int_z^\infty \frac{e^{-t}}{t}dt = \int_{-\infty}^{-z}\frac{e^{-t}}{t}dt = \int_{-\infty}^{-z}$
ExpIntegralE[n, z]	指数積分関数 $\mathrm{E}_n(z) = \int_1^\infty \frac{e^{-zt}}{t^n}dt$
LogIntegral[z]	対数積分関数 $\mathrm{l}_i(z) = \int_0^\infty \frac{dt}{\log t}$
Erf[z]	誤差関数 $\mathrm{erf}(z) = \frac{2}{\pi}\int_0^z e^{-t^2}dt$
Erf[z_0, z_1]	誤差関数 $\frac{2}{\sqrt{\pi}}S_{z_0}^{z_1}e^{-t^2}dt$

級数関数

Zeta[s]	リーマンのツェータ関数 $\zeta(s) = \sum_{k=1}^\infty \frac{1}{k^s}$
Zeta[s, a]	一般のツェータ関数 $\zeta(s,a) = \sum_{k=0}^\infty \frac{1}{(k+a)^s}$
Poly[n, z]	ポリ対数関数 $\mathrm{Li}_n(z) = \sum_{k=1}^\infty \frac{z^k}{k^n}$
LerchPhi[z, s, a]	レルヒ関数 $\Phi(z,s,a) = \sum_{k=0}^\infty \frac{z^k}{(a+k)^n}$

BernoulliB$[n, x]$	ベルヌーイ多項式 $\mathrm{B}_n(x)$	$\frac{te^{xt}}{e^t-1} = \sum_{n=0}^{\infty} \mathrm{B}_n(x)\frac{t^n}{n!}$
Bernoulli$[n]$	ベルヌーイ数 $\mathrm{B}_n = \mathrm{B}_n(0)$	
EulerE$[n, x]$	オイラー多項式 $\mathrm{E}_n(x)$	$\frac{2e^{xt}}{e^t+1} = \sum_{n=0}^{\infty} \mathrm{E}_n(x)\frac{t^n}{n!}$
EulerE$[n]$	オイラー数 $\mathrm{E}_n = 2^n\mathrm{E}_n(\frac{1}{2})$	
StirlingS1$[n, m]$	第 1 種のスターリング数 $\mathrm{S}_n^{(m)}$	$x(x-1)\cdots(x-n+1) = \sum_{m=0}^{n} \mathrm{S}_n^{(m)}x^m$
StirlingS2$[n, m]$	第 2 種のスターリング数 $\mathrm{S}_n^{(m)}$	$x^n = \sum_{m=0}^{n} \mathrm{S}_n^{(m)}x(x-1)\cdots(x-m+1)$

特殊関数

AiryAi$[z]$	エヤリ関数 $\mathrm{Ai}(z)$	微分方程式 $y'' - xy = 0$ の解
BesselJ$[n, z]$	第 1 種のベッセル関数 $\mathrm{J}_n(z)$	
BesselY$[n, z]$	第 2 種のベッセル関数 $\mathrm{Y}_n(z)$	
BesselI$[n, z]$	第 1 種の変形ベッセル関数 $\mathrm{I}_n(z)$	
BesselK$[n, z]$	第 2 種の変形ベッセル関数 $\mathrm{K}_n(z)$	
LegendreP$[n, z]$	第 1 種のルジャンドル関数 $\mathrm{P}_n(z)$	
LegendreP$[n, m, z]$	第 1 種のルジャンドル随伴関数 $\mathrm{P}_n^m(z)$	
LegendreQ$[n, z]$	第 2 種のルジャンドル関数 $\mathrm{Q}_n(z)$	
LegendreQ$[n, m, z]$	第 2 種のルジャンドル随伴関数 $\mathrm{Q}_n^m(z)$	

グラフィックス

―――――― グラフィック関数 ――――――

総合関数

Show[オブジェクト] グラフィックスオブジェクトを描画

複数指定した場合は結合して描画、オプションの変更も可能。

InputForm[オブジェクト] オブジェクトの内容表示

２Ｄグラフ関数

$\mathrm{Plot}[f, \{x, x_{\min}, x_{\max}\}]$ グラフ描画

$\mathrm{Plot}[\{f_1, f_2, \ldots\}, \{x, x_{\min}, x_{\max}\}]$ 複数のグラフ描画

$\mathrm{PointPlot}[\{f_1, f_2, \ldots\}, \{x, x_{\min}, x_{\max}, x_{\mathrm{step}}\}]$ グラフを点描画

$\mathrm{ParametricPlot}[\{\{f_x(t), f_y(t)\}, \{g_x(t), g_y(t)\}, \ldots\}, \{t, t_{\min}, t_{\max}\}]$

パラメータ曲線を描画

$\mathrm{PointParametricPlot}[\{\{f_x(t), f_y(t)\}, \{g_x(t), g_y(t)\}, \ldots\}, \{t, t_{\min}, t_{\max}, t_{\mathrm{step}}\}]$

パラメータ曲線を点描画

$\mathrm{ListPlot}[\{y_1, y_2, \ldots\}]$ 点描画（$x = 1, 2, \ldots$）

$\mathrm{ListPlot}[\{\{x_1, y_1\}, \{x_2, y_2\}, \ldots\}]$ 点描画

Graphics[{プリミティブ}] オブジェクトを直接作る

３Ｄグラフ関数

$\mathrm{Plot3D}[f, \{x, x_{\min}, x_{\max}\}, \{y, y_{\min}, y_{\max}\}]$ グラフ描画

Plot3D[{f, 濃度関数}, $\{x, x_{\min}, x_{\max}\}, \{y, y_{\min}, y_{\max}\}$] 濃度変化を関数で指定

$\mathrm{PointPlot3D}[f, \{x, x_{\min}, x_{\max}, x_{\mathrm{step}}\}, \{y, y_{\min}, y_{\max}, y_{\mathrm{step}}\}]$

グラフを点描画

$\mathrm{ParametricPlot3D}[\{f_x(u,v), f_y(u,v), f_z(u,v)\}, \{u, u_{\min}, u_{\max}\}, \{v, v_{\min}, v_{\max}\}]$

パラメータ曲面を描画

PointParametricPlot3D[$\{f_x(u,v), f_y(u,v), f_z(u,v)\}, \{u, u_{\min}, u_{\max}, u_{\mathrm{step}}\}, \{v, v_{\min}, v_{\max}, v_{\mathrm{step}}\}$]

パラメータ曲面を点描画

ListPlot3D[データリスト] 点描画

データリストは $m \times n$ 行列で、各要素は z 座標を表す。（$x = 1 \sim m$, $y = 1 \sim n$）

ListPlot3D[データリスト, 濃度リスト] 濃度リストは $(m-1)\times(n-1)$ 行列

Graphics3D[{プリミティブ}] オブジェクトを直接作る

等高線グラフ関数

ContourPlot[$f, \{x, x_{\min}, x_{\max}\}, \{y, y_{\min}, y_{\max}\}$] 等高線グラフ

ListContourPlot[データリスト] データリストは n 次正方行列

濃淡グラフ関数

DensityPlot[$f, \{x, x_{\min}, x_{\max}\}, \{y, y_{\min}, y_{\max}\}$] 濃淡グラフ

ListDensityPlot[データリスト] データリストは n 次正方行列

アニメ-ション

Animate[式, $\{i, i_{\min}, i_{\max}, i_{\mathrm{step}}\}$] アニメ-ション

――――――― オプション ―――――――

２Ｄグラフのオプション

PlotRange −> 範囲（def： Automatic）

$\{y_{\min}, y_{\max}\}$： y の範囲

$\{\{x_{\min}, x_{\max}\}, \{y_{\min}, y_{\max}\}\}$： x, y の範囲

All：すべての点をプロット

PlotLabel −> ”ラベル”（def： None）

Framed –> フレームを描くか (def : False)

AspectRatio –> 縦軸 / 横軸比 (def : Automatic)

Axes –> 軸 (def : Automatic)

$\{x, y\}$: (x, y) を原点として軸を描く

None : 軸を描かない

AxesLabel –> {"x ラベル","y ラベル"} (def : None)

Ticks –> 目盛 (def : Automatic)

None : 両軸ともに目盛を描かない

$\{x_{\mathrm{ticks}}, y_{\mathrm{ticks}}\}$: 目盛を打つ位置をリストで指定

PlotStyle –> $\{\{\mathrm{style}_1\}, \{\mathrm{style}_2\}, \ldots\}$、各曲線のスタイルを順に指定

GrayLevel[x] : 線の濃度 ($0 \leq x \leq 1$)

Thickness[t] : 線の太さ (t は数値)

Dashing[$\{d_1, d_2\}$] : d_1, d_2 間隔で点線を描く

PlotPoints[6] –> 関数をサンプルする最小の点の数 (def : 25)

$\{n_x, n_y\}$: x 軸、y 軸のサンプル数を指定

Maxbend[6] –> 隣り合うセグメント間の最大の曲げ角 (def : 10.)

Plotdivision[6] –> セグメント間のサブ分割の最大値 (def : 20.)

PlotJoined –> ListPlot 関数で、点と点を結ぶか (def : False)

3Dグラフのオプション

AmbientLight –> GrayLevel[x]、地の濃度 ($0 \leq x \leq 1$、def : $x = 0$)

BoxRatios –> $\{x, y, z\}$、グラフを囲むボックスの各方向の比 (def : $\{1, 1, 0.4\}$)

Boxed –> ボックスを描くか (def : True)

ClipFill –> {bottom,top}、グラフが上下でクリップされた場合、下を bottom、上を top で描く。

[6] Show 関数の中では使用できない。

それぞれ GrayLevel で指定する。

Automatic：シェーディングする（デフォルト）

None：シェーディングしない

HiddenSurface -> 隠線処理をするか（def： True）

Lighting -> 擬似照明をあてるか（def： False）

LightSources -> $\{\{\{x_1, y_1, z_1\}, i_1\}, \{\{x_2, y_2, z_2\}, i_2\}, \ldots\}$

点 $\{x_j, y_j, z_j\}$ に強さ i_j の擬似光源を置く

Mesh -> メッシュを描くか（def： True）

PlotPoints -> 格子数（def： 15）

n：$(n-1)\times(n-1)$ のメッシュ

$\{n, m\}$：$(n-1)\times(m-1)$ のメッシュ

PlotRange -> 範囲 (def： Automatic）

All：すべての点をプロット

$\{\min, \max\}$：高さの範囲を指定

Shading -> 各メッシュに影をつけるか（def： True）

ViewPoint -> $\{x, y, z\}$、視点（def： $\{1.3, -2.4, 2.\}$）

等高線グラフのオプション

AspectRatio -> 縦横比（def： 1）

Axes -> x-y 軸（def： Automatic）

None：軸を描かない

AxesLabel -> {”x ラベル”,”y ラベル”}（def： None）

ContourLevels -> 等高線のレベルの数（def： 10）

ContourSpacing -> 等高線の間隔

Framed -> フレームを描くか（def： True）

PlotLabel -> ”ラベル”（def： None）

PlotRange -> 範囲（def： Automatic）

$\{\min, \max\}$：範囲

PlotPoints -> 分割数（def： 15）

Ticks	–> 目盛 (def : Automatic)	
	None : 目盛を描かない	
	リスト : 目盛を打つ位置	

濃淡グラフのオプション

AspectRatio	–> 縦横比 (def : 1)
Axes	–> x-y 軸 (def : Automatic)
	None : 軸を描かない
AxesLabel	–> {"x ラベル","y ラベル"} (def : None)
Framed	–> フレームを描くか (def : True)
Mesh	–> メッシュを描くか (def : True)
PlotLabel	–> "ラベル" (def : None)
PlotPoints	–> n、 n×n のメッシュを描く (def : 15)
PlotRange	–> 範囲 (def : Automatic)
	{min, max} : min 以下なら GrayLavel[0]、 max 以上なら GrayLevel[1] とする
Ticks	–> 目盛 (def : Automatic)
	None : 目盛を描かない
	リスト : 目盛を打つ位置

———————— プリミティブ ————————

Graphics 関数のプリミティブ

Point[$\{x, y\}$]	点 $\{x, y\}$ に 1 点をプロット
Line[$\{\{x_1, y_1\}, \{x_2, y_2\}, \ldots\}$]	各点を順に線で結ぶ
Rectangle[$\{x_{\min}, y_{\min}\}, \{x_{\max}, y_{\max}\}$]	長方形を描き、塗り潰す
Polygon[$\{\{x_1, y_1\}, \{x_2, y_2\}, \ldots\}$]	順に結んで多角形を描き、塗り潰す
CellArray[array, $\{\{x_{\min}, y_{\min}\}, \{x_{\max}, y_{\max}\}\}$]	長方形の中を濃度レベルの記入された array にとも

	なうメッシュで埋める
Text[テキスト, $\{x, y\}$, オフセット]	テキストを書く（スペースを含むのなら ”” で囲む）

オフセット

$\{0,0\}$	センタリング
$\{-1,0\}$	左詰め
$\{1,0\}$	右詰め
$\{0,1\}$	上
$\{0,-1\}$	下

GrayLevel[i]	濃度（黒 $= 0 \leq i \leq 1 =$ 白）
PointSize[s]	点の大きさ（グラフの幅に対する比率）
Thickness[t]	線の大きさ（グラフの幅に対する比率）
Dashing[$\{d_1, d_2, \ldots\}$]	破線を描く（d_i はグラフの幅に対する比率）

Graphics3D 関数のプリミティブ

Point[$\{x, y, z\}$]	点 $\{x, y, z\}$ に 1 点をプロット
Line[$\{\{x_1, y_1, z_1\}, \{x_2, y_2, z_2\}, \ldots\}$]	各点を順に線で結ぶ
Polygon[$\{\{x_1, y_1, z_1\}, \{x_2, y_2, z_2\}, \ldots\}$]	順に結んで多角形を描く。原則として、３点ずつの三角形の集まったものとして描く。

GrayLevel[i]	濃度（黒 $= 0 \leq i \leq 1 =$ 白）
PointSize[s]	点の大きさ（グラフの幅に対する比率）
Thickness[t]	線の大きさ（グラフの幅に対する比率）
FaceForm[p_f, p_b]	Polygon の前面を p_f、後面を p_b とする（p_f, p_b は GrayLevel で与える）
EdgeForm[p]	エッジの属性（p は GrayLevel または Thickness で与える）

プログラミング

プログラミング関数

Block[{変数リスト}, 式]	ブロック化（式：セミコロン；で区切られたコマンド列）

変数リスト

$\{\}$	ローカル変数を使用しない
$\{x_1, x_2, \ldots\}$	ローカル変数のリスト
$\{x_1 =$ 初期値$, \ldots\}$	初期化する場合

Do[式, $\{n\}$]	n 回反復
Do[式, $\{i, i_{\min}, i_{\max}\}, \{\}, \ldots$]	カウンタを使用
	Return,Break での脱出不可、関数値を返さない
For[初期式, 条件, インクリメント, 式]	反復

初期式	最初に実行する式（例： i=0）
条件	True の間反復
インクリメント	i++; j+=2 等

While[条件, 式]	反復
Return[関数値]	関数値を返す（省略可）
Break[]	一番内側の For,While から抜け出す
Continue[]	それ以降をとばして For,While の先頭に戻る
Nest[式, 引数, n]	引数に式を n 回繰り返し実行

Catch[~ Throw[関数値] ~]
Throw を実行した時点でその Catch ブロックは終了となる。 Catch の返す値は Throw で指定した値

Goto[ラベル]	分岐
Label[ラベル]	ラベルを設定
If[条件, 式$_1$, 式$_2$, 式$_3$]	条件が True なら 式$_1$、 False なら 式$_2$、どちらでもなければ

	式$_3$ を実行（式$_3$ は省略可）
Which[条件$_1$, 式$_1$, 条件$_2$, 式$_2$, ...]	条件$_i$ が True なら 式$_i$ を実行
Switch[式, 値$_1$, 式$_1$, 値$_2$, 式$_2$, ...]	式の値が 値$_i$ なら 式$_i$ を実行
Print[式]	式を表示

ファイル

パッケージ

<< パッケージファイル名（—.m）　ロード

パッケージの書式

BeginPackage["コンテクスト名 ‘ "]　· · · パッケージの始め

関数名::usage=　· · · オンラインヘルプ

"メッセージ"

関数定義

EndPackage[]　· · · パッケージの終り

ファイル入出力

式 >> ファイル名　式の内容をセーブ、ファイル名は "" でくくってもよい

グラフィックスオブジェクトもセーブ可

Put[式$_1$, 式$_2$, · · · , ファイル名]　同上

式 >>> ファイル名　式の内容をファイルに追加

PutAppend[式$_1$, 式$_2$, · · · , ファイル名]　同上

<< ファイル名　ロード、そのまま *Mathematica* の入力となる

複数の式がある時は最後のものだけが有効

Get[ファイル名]　同上

!! ファイル名　ファイルの全内容表示、ロードはしない

OpenWrite["ファイル名"]　セーブ

オプション

FormatType　–> CForm：C 言語のフォーマット

FortranForm： Fortran のフォーマット

InputForm： *Mathematica* 入力フォーマット（デフォルト）

OutputForm：出力フォーマット

TexForm： $\TeX$ フォーマット

TextForm：テキストフォーマット

PageHeight –> １ページの行数（def： 22）

PageWidth –> １行のカラム数（def： 78）

OpenRead["ファイル名"]	ファイルをオープンして、先頭を指示
OpenAppend["ファイル名"]	追加、 OpenWrite と同じオプションを指定できる
Close["ファイル名"]	クローズ
Write["ファイル名", 式$_1$, 式$_2$, . . .]	（オープンされていなければオープンして）書き込み クローズはしない
Read["ファイル名"]	１個の式を読み出す
Read["ファイル名", タイプ]	１個のオブジェクトを指定されたタイプのデータとして読む

タイプ

Byte	１個のバイトで整数として返す
Character	１個の文字、長さ１の文字列
Number	整数、または指数形式の数
String	１個の文字列
Real	指数形式の数
Expression	*Mathematica* の式（デフォルト）

Read["ファイル名", {タイプ$_1$, タイプ$_2$, . . .}]	リストで示されたタイプの要素を１つの グループとしてリストにまとめながら読み出す
ReadList["ファイル名", タイプ]	読んだデータをリストの形式で戻す

例： tmp.m は 1.2 4.4 5.7 6.6 9.2 8.2 から成る
ReadList["tmp.m",{String,String}] → {{1.2,4.4},{5.7,6.6},{9.2,8.1}}

名前、定義

―――――――― 定義、代入 ――――――――

名前の定義

名前	変数の宣言
Remove[名前]	名前の除去
名前 = 式	変数の代入
式$_1$:=式$_2$	関数定義（例： $f[x\_]{:=}x\hat{}2$）
式 /. 変数 −> 代入値	一時的代入
式 /. { 変数 −> 代入値,...}	一時的代入
	例： $a+b$ /. $\{a{-}{>}1, b{-}{>}2\} \rightarrow 3$
名前 /: 式$_1$=式$_2$	タグセット
名前 /: 式$_1$:=式$_2$	タグセット

例： a /: $f[a]=0 \rightarrow a$ は $f[a]=0$ という性質を持つ

式$_1$ ^= 式$_2$	アップセット、 式$_1$ が現れると 式$_2$ に置き換える
式$_1$ ^:= 式$_2$	アップセット
Clear[変数$_1$, 変数$_2$, ...]	変数クリア
名前 =.	内容クリア、 Clear[変数$_1$, 変数$_2$, ...] と同じ
名前 = 式 /; 条件	条件付き定義
式$_1$:=式$_2$ /; 条件	条件付き定義
コンテクスト ‘名前	正式名称（コンテクストも指定）
名前$_1$::=名前$_2$	別名定義、 名前$_1$ が現れると 名前$_2$ に置き換える
”別名 ”::=.	別名取消
Alias[”別名 ”]	別名の内容表示、引数を省略すると全別名を表示

公式

公式名 ={ 公式, 公式,...}　　公式の定義

公式の書式：　関数 :> 書き換えられる形

例：　c={a_. Cos[x_]+a_. Cos[y_] :> 2aCos[$x/2+y/2$] Cos[$x/2-y/2$]}

式 //. 公式名　　公式による書き換え

例：　Cos[2Pi/5]+Cos[3Pi/5] //. $c \to 0$

関数の評価

Evaluate[関数名]　　関数に名前を付けて、それを用いてグラフを描く時等に使う。

例：　f={Sin[x],Exp[x]}

Plot[Evaluate[f],{x,-1,1}];

名前の定義内容参照関数

Definition[名前$_1$, 名前$_2$, ...]	定義された値と属性表示
FullDefinition[名前$_1$, 名前$_2$, ...]	すべての定義を表示
Save["ファイル名", 名前$_1$, 名前$_2$, ...]	定義をファイルにアペンド

———————— 関数定義の引数 ————————

引数パターンの表記法

_	任意の式
x_	x という名前の束縛変数
x_h	ヘッダ h 型の引数 x
__	１個以上の式の列
x_	名前が x である列
x_h	ヘッダ h 型の列 x
x___	式の列（長さ 0 も可）
x_:v	引数が省略された時、デフォルトとして v をとる
x_.	引数が省略された時、デフォルトとして 0（加減算）や 1（乗除算、べき乗）になる

例：　f[x_,n_.]:=x^n $\to$ f[2]=2^1=2

ヘッダ

Integer	整数
Rational	有理数
Real	実数
Complex	複素数
List	リスト
String	”ストリング”
Symbol	シンボル
Equal	等式

引数パターンの制約

x: パターン　　名前 x のパターンを指定

例： $f[x{:}a\_\hat{}n\_]{:=}n\ \mathrm{Log}[a]$, $f[x\_]{:=}\mathrm{Log}[x]$ $\rightarrow$ $f[2\hat{}3]{=}3\ \mathrm{Log}[2]$, $f[8]{=}\mathrm{Log}[8]$

パターン ? 条件　　条件が True なら有効

例： $f[x\_?\mathrm{Positive}]{:=}\mathrm{Sqrt}[x]\rightarrow$ $f[2]{=}\mathrm{Sqrt}[2]$, $f[\text{-}1]{=}f[\text{-}1]$

パターン..　　1 個以上のパターンの列

パターン...　　パターンの列（長さ 0 も可）

———————— アトリビュート ————————

アトリビュート関数

Attributes[名前]	アトリビュートを返す
Attributes[名前]	={ アトリビュート,...} で設定
	={} で空にする
SetAttributes[名前, アトリビュート]	セット
ClearAttributes[名前, アトリビュート]	クリア
Unprotect[名前]	Protect 属性の解除

アトリビュート

Orderless	引数の順を問わない
Flat	$f[f[a,b],c] \to f[a,b,c]$ とする
OneIdentity	$f[x], \cdots$ 等がパターンマッチングに対しては x に等価
Listable	$f[\{a,b\}] \to \{f[a],f[b]\}$
Constant	微分すると 0
Protected	名前の内容の変更不可
Locked	属性変更不可
ReadProtected	読み出し不可
HoldFirst	先頭の引数は評価されない
HoldRest	２番目以降の引数は評価されない
HoldAll	すべての引数は評価されない

ヘルプ、デバッグ

ヘルプ

? 名前	オンラインヘルプ
Names["名前"]	名前に一致するものの一覧

名前におけるワイルドカード

*	0 個以上の文字
**	0 個以上の文字、又は $
@	1 個以上の小文字

デバッグ

On[関数]	関数の実行をトレース、引数を省略すると全関数
Off[関数]	トレース終了、引数を省略すると全関数

個人的に追加した関数（$user`$）

数値関数、整数関数

PrimeList[n]	n 以下の素数のリスト
Fibonacci[n]	フィボナッチ数列
BinomialList[n]	$(1+x)^n$ の係数のリスト
PascalTriangle[n]	パスカルの三角形

数式関数

CompleteSquare[f, x]	2 次関数 $f(x)$ を平方完成する
Average[$f, \{a, b\}$]	$f(x)$ の区間 $[a, b]$ における平均値
Tangent[f, a][x]	$f(x)$ の、$x = a$ における接線の方程式
Normal[f, a][x]	$f(x)$ の、$x = a$ における法線の方程式
Curvature[f][a]	$f(x)$ の $x = a$ における曲率
RadiusofCurvature[f][a]	$f(x)$ の $x = a$ における曲率半径
CenterofCurvature[f][a]	$f(x)$ の $x = a$ における曲率中心の座標
Canonical[$\{x_1, x_2, \ldots\}, M$]	変数 $x_1, x_2, \ldots$、係数行列 M の 2 次形式の標準形を求める
FourierExpand[$f, t, n, \{t_{\min}, t_{\max}\}$]	$f(t)$ を区間 $[t_{\min}, t_{\max}]$ について n 回フーリエ級数展開

リスト、ベクトル、行列

RegularPartition[$\{a, b\}, n$]	区間 $[a, b]$ を n 等分し、リストを返す
RandomPartition[$\{a, b\}, n$]	区間 $[a, b]$ を n 個に分割（ランダム）
Refine[リスト, n]	昇べきに並んだリストに n 個の要素をランダムに追加
Norm[リスト]	リスト中の隣あった要素の差の最大を返す
Changes[リスト]	リスト中の要素が何回変わるか

$\mathrm{VectorAngle}[v_1, v_2]$	ベクトルのなす角
$\mathrm{Cross3D}[v_1, v_2]$	3次元ベクトルの外積
$\mathrm{GramSchmidt}[\{v_1, v_2, \ldots\}]$	グラム・シュミットの正規直交化

図形関数

$\mathrm{Ex}[n, r]$	半径 r の円の外接 n 角形の面積
$\mathrm{In}[n, r]$	半径 r の円の内接 n 角形の面積

グラフ関数

$\mathrm{TangentPlot}[f, a, \{x_1, x_2\}]$	$f(x)$ の、 $x = a$ における接線のグラフ
$\mathrm{NormalPlot}[f, a, \{x_1, x_2\}]$	$f(x)$ の、 $x = a$ における法線のグラフ
$\mathrm{TangentFamily}[f, \{x_{\mathrm{min}}, x_{\mathrm{max}}, x_{\mathrm{step}}\}]$	$f(x)$ の接線の族のグラフ
$\mathrm{NormalFamily}[f, \{x_{\mathrm{min}}, x_{\mathrm{max}}, x_{\mathrm{step}}\}, \{y_{\mathrm{min}}, y_{\mathrm{max}}\}]$	$f(x)$ の法線の族のグラフ
$\mathrm{PlotCurvature}[f, \{x_{\mathrm{min}}, x_{\mathrm{max}}\}]$	曲率のグラフ
$\mathrm{PlotOsculatingCircle}[f, a]$	$f(x)$ の $x = a$ における曲率円のグラフ
$\mathrm{PlotEvolute}[f, \{x_{\mathrm{min}}, x_{\mathrm{max}}\}]$	縮閉線（法線の包絡線）のグラフ
$\mathrm{TraceFixedPoint}[f, x_0, \{x_{\mathrm{min}}, x_{\mathrm{max}}\}][n]$	不動点を探していく軌跡（ n 回反復）

索引

A

B