

ベクトルの演算

- ベクトルは「リスト」：成分をコンマ (,) 区切りで中括弧 { } で囲む.
- 和は「+」、差は「-」
- スカラー倍は「*」
- 内積はピリオド「.」、長さ（ノルム）は「`Norm[list]`」
つまり「`Norm[list]`」は「`Sqrt[list.list]`」と同じ？
- 空間ベクトルの外積は「`Cross[list,list]`」

関数のプロット (平面内のグラフ)

- 陽関数 : $y = f(x)$

`Plot[f(x), {x, “x の最小値”, “x の最大値”}]`

- 陰関数 : $f(x, y) = 0$

`ContourPlot[f(x,y)==0, {x, xmin, xmax}, {y, ymin, ymax}]`

- パラメータ表示 : $(x, y) = (x(t), y(t))$

`ParametricPlot[{x(t), y(t)}, {t, tmin, tmax}]`

等号「=」

- 「=」は「割り当て」または「代入」

- 方程式の等号を表す場合は「==」

例) 連立1次方程式の解を求める

```
Solve[{a*x+b*y==e, c*x+d*y==f}, {x, y}]
```

- 引数付き関数の定義（割り当て）は「:=」を用いる。

例) 2次関数 $f(x) = ax^2 + bx + c$ を定義するには

```
f[x_] := a*x^2 + b*x + c
```

関数のプロット (空間内のグラフ)

- 陽関数 : $z = f(x, y)$

```
Plot3D[f(x, y), {x, xmin, xmax}, {y, ymin, ymax}]
```

- 陰関数 : $f(x, y, z) = 0$

```
ContourPlot3D[f(x, y, z)==0,  
{x, xmin, xmax}, {y, ymin, ymax}, {z, zmin, zmax}]
```

- パラメータ表示 (変数が1つ) : $(x, y, z) = (x(t), y(t), z(t))$

```
ParametricPlot3D[{x(t), y(t), z(t)}, {t, tmin, tmax}]
```

- パラメータ表示 (変数が2つ) : $(x, y, z) = (x(s, t), y(s, t), z(s, t))$

```
ParametricPlot3D[{x(s, t), y(s, t), z(s, t)},  
{s, smin, smax}, {t, tmin, tmax}]
```

複数の図形を同一平面また空間内に描画する

- Show コマンド

```
Show[(プロットコマンド1) , (プロットコマンド2) , ...]
```

- 例)

```
Show[  
  Plot[f(x) , {x, a, b}] ,  
  ParametricPlot[{x(t) , y(t)} , {t, c, d}]  
]
```

複数の図形を同一平面また空間内に描画する

- 同じプロットコマンドで、描画範囲が同じ場合

例) 以下の2つは同じ出力

- Show[

```
ParametricPlot[{x(t), y(t)}, {t, a, b}],
```

```
ParametricPlot[{u(t), v(t)}, {t, a, b}],
```

```
]
```

- ParametricPlot[{{x(t), y(t)}, {u(t), v(t)}}, {t, a, b}]