



第25回 マセマティカでグラフを描こう

前回は、超高級電卓のマセマティカ (Mathematica) を使って、基本操作から連立方程式 (つるかめ算) を解くところまでやってみた。今回はマセマティカでいろいろなグラフを描いてみよう。先月号を読んでいないひと人や復習したい人は、スタプロのバックナンバーを読んでね。

キットの情報は
KoKa Shop!へ定期購読者特典割引あり



ゲーム 工作 プログラミング

ラズベリーパイ Raspberry Pi 大活用

監修・原案／青山学院大学客員教授 阿部和広 構成・文／塩野祐樹

ジブン専用パソコン特設サイト
prog.kodomonokagaku.com/jibun



マセマティカの起動

スタートメニューの「プログラミング」から「Mathematica」を選んで、マセマティカを起動しよう。起動したら、ノートブックのウィンドウをクリックして手前に出す。詳しい手順は、前回のバックナンバーを見てね。

一次関数と二次関数ってナニ?

今回は、「関数」のグラフを描くんだけど、これは中学校で習う内容だ。そこで、小学生のみんなのために、まずは関数について説明しよう。

関数を簡単にいうと、「2つの変数x、yがあり、xの値を決めるとき、yはxの関数である」となる。あんまり簡単じゃない?

では、例を挙げて説明しよう。例えば「 $y=2x+5$ 」という式があるとする。xに3を入れると、右辺は「 $2 \times 3 + 5$ 」となり、yは11になるね。つまり、xの値を決めたことで、yの値も決まったというわけ。これが関数だ。

「一次」や「二次」というのは、xを何回掛けているかを表している。さっきの式は「 $2x+5$ 」だったので「一次関数」。もし「 $2x^2+5$ 」なら、xを2回掛けている(自乗(二乗)している)ので「二次関数」になる。関数についてはこのくらいにして、さっそくグラフを描いてみよう。

一次関数のグラフ

では、手始めに一次関数のグラフを描いてみよう。最初は超簡単に、こんな式はどうだろう。

$$y = x$$

これは、yの値はいつもxと同じになる、という意味だ。

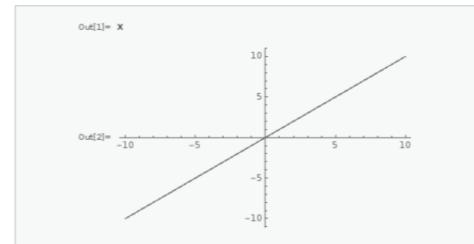
マセマティカでは、グラフを描くために「Plot」という命令を使う。Plotは「描画する」という意味。右上のように入力(In)しよう。大文字と小文字も正しく入力してね。改行するときは「Enter」キーを押そう。ただし「Shift」キーと一緒に押すと実行しちゃうので注意。今回は「In」の表示を省略しているよ。

$y = x$
Plot[y, {x, -10, 10}]

前回やったように、1行目の「=」は、比較ではなく代入の意味だね。

2行目の「Plot[]」の中にある「y, {x, -10, 10}」は、xの範囲を-10から10までとしたときのyのグラフを描くことを表している。「{}」で囲んでいるのは、複数の要素を持つリストだからだ。式を入力したら「Shift」キーを押しながら「Enter」キーを押して実行しよう(グラフ①)。

グラフ①



「Out[1]」は、最初の式の「 $y = x$ 」の結果、「Out[2]」が2番目の式のPlotの結果のグラフだ。グラフの横軸が「x」で、縦軸が「y」。xとyの関係が、右肩上がりの直線で表されているね。例えばxが5のとき、yも5になっているよね。このグラフから、yの値がいつもxの値と同じであることが読み取れる。

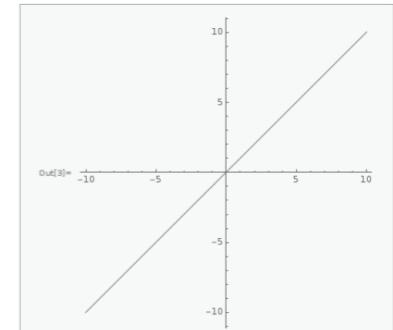
ただ、このグラフは縦横の比率が黄金比(1:1.61803...)になっているので、xとyの関係がちょっとわかりにくい。これを変えるには、「AspectRatio」というオプションを使うよ。関数は前と同じなので、「 $y = x$ 」は入力しなくても丈夫だ。

Plot[y, {x, -10, 10}, AspectRatio → Automatic]

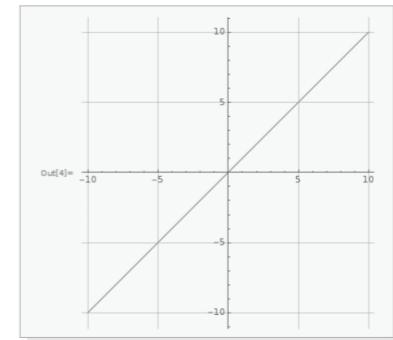
この式を入力するときは、前と違うところだけを変えたり、すでに入力してある式をコピー&ペーストしたりすると楽だよ。「→」は、「-」「>」を順番に押せばOK。その後の「Automatic」は「自動」という意味だ。この式を実行したのがグラフ②だ。

さっきより見やすくなったね。これに目盛線がつくと、もっと見やすそうだ。目盛線を表示するオプションは「GridLines」だよ(グラフ③)。

グラフ②



グラフ③



Plot[y, {x, -10, 10}, AspectRatio → Automatic, GridLines → Automatic]

追加した部分

では、次にyの増え方を変えてみよう。グラフの傾きが変わるぞ。

$$y = -1/2 x$$

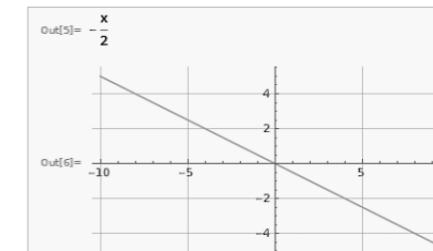
-1/2がグラフの傾きを表しているよ。この式がどんなグラフになるか想像できるかな?

追加した部分

Plot[y, {x, -10, 10}, AspectRatio → Automatic, GridLines → Automatic]

右肩下がりで、傾きがさっきの半分(1/2)になったね(グラフ④)。例えば、xが4のとき、yは-2になっている。今度は、このグラフを上に(y軸の+方向に)動かしてみよう。

グラフ④



$$y = -1/2x + 4$$

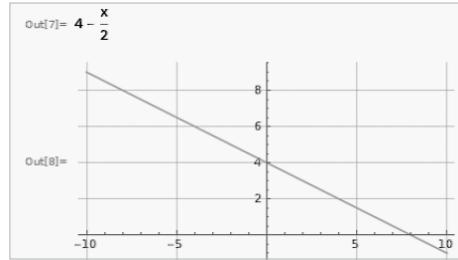
ここで直線が上に「4」動くはず。この「4」のことを切片といふんだ(正確にはy切片)。

実行してみよう(グラフ⑤)。グラフを読むと、 x が0のとき、 y は4になっていることがわかるね。

追加した部分

```
y = -1/2 x + 4  
Plot[y, {x, -10, 10}, AspectRatio → Automatic, GridLines → Automatic]
```

グラフ⑤



二次関数のグラフ

同じようにして、二次関数のグラフも描ける。

$$y = x^2$$

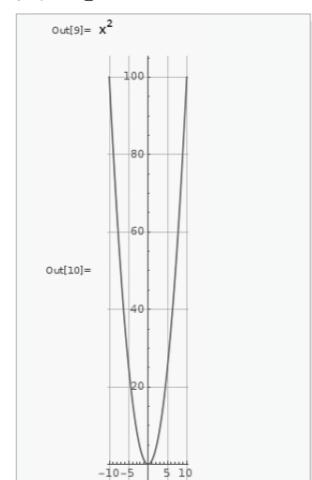
この式は「 $y = x \times x$ 」と同じ意味だ。自乗(二乗)は「 \wedge 」で表すから、入力する式はこうなる。

変更した部分

```
y = x^2  
Plot[y, {x, -10, 10}, AspectRatio → Automatic, GridLines → Automatic]
```

実行すると、ものすごく細長いグラフになったぞ!(グラフ⑥)こんな形の線のことを放物線というんだ。上下を逆さまにすると、物を投げたときの位置の変化を表すような線になるね。

グラフ⑥

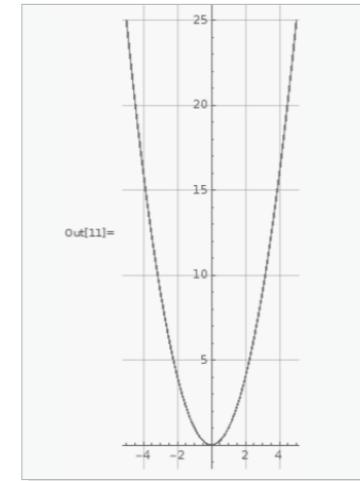


ちょっと細長すぎるので、 x の範囲を-5から5までに変えておこう(グラフ⑦)。

変更した部分

```
Plot[y, {x, -5, 5}, AspectRatio → Automatic, GridLines → Automatic]
```

グラフ⑦



複数のグラフの交点を求める

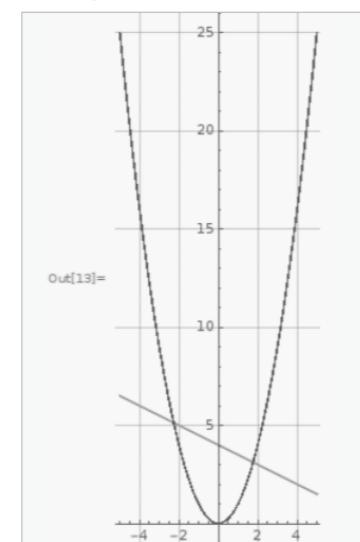
このグラフに、さっきの一次関数のグラフを重ねて表示することもできる。

どちらにも変数の y を使ってるので、区別するために1つを「 $y0$ 」という名前に変えたよ(グラフ⑧)。

変更した部分

```
y0 = -1/2 x + 4  
Plot[{y0, y}, {x, -5, 5}, AspectRatio → Automatic, GridLines → Automatic]
```

グラフ⑧



ところで、この2つのグラフがぶつかっているところ、つまり交点の座標(点の位置を指定する数の組)を調べてみたい。でも、目で見てもよくわからないね。こんなときも、マセマティカは自動で答えを出してくれるんだ。

まず、変数 y をクリアしておこう。

```
Clear[y]
```

交点の座標を求めるための式は下のようになる。「NSolve」は前回使った「N」と「Solve」を組み合わせたもので、解いた結果を数値で返してくれるんだ。

```
NSolve[{y = -1/2 x + 4, y = x^2}, {x, y}]
```

ここでは、「=」じゃなくて「==」を使っていることに注意しよう。これは、代入じゃなくて、式の左辺と右辺が等しいという意味だったね。実行するとこうなるよ。

```
Out[15]= {{x → 1.76556, y → 3.11722}, {x → -2.26556, y → 5.13278}}
```

グラフ⑧を見ると、この値が交点の座標を表していることがわかる。ところで、これって前回のつるかめ算の連立方程式にちょっと似てると思わない? 式が2つで、求めたい変数も2つ。実は、グラフの交点を求めるということは、連立方程式の解を求めるのと同じことなんだ。ということは、つるかめ算もグラフで表せる、ということだ。

つるかめ算をグラフで表す

前回の問題は、ツルとカメが合わせて10匹、足の数の合計が28本だったとき、ツルとカメのそれぞれの数だったね。

入力した式と答えは、こうだった。

```
In[16]:= Solve[{tsuru + kame == 10, 2 tsuru + 4 kame == 28}, {tsuru, kame}]  
Out[16]= {{tsuru → 6, kame → 4}}
```

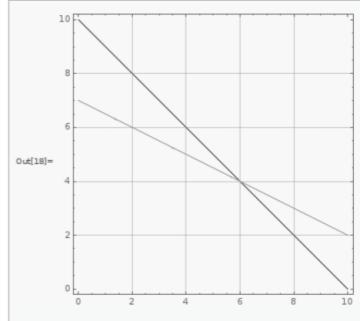
この式をグラフで表すためには「ContourPlot」という命令を使うんだ。「Contour」は「等高線」という意味で、この式を満たす点の集まりを線として

```
ContourPlot[{tsuru + kame == 10, 2 tsuru + 4 kame == 28}, {tsuru, 0, 10}, {kame, 0, 10}, GridLines → Automatic]
```

描いてくれる。

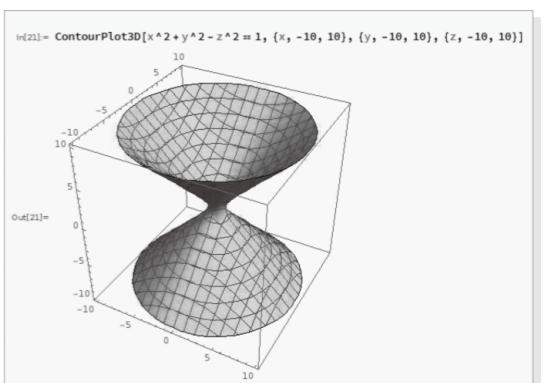
この式を実行したのがグラフ⑨。このグラフは、横軸がtsuruで、縦軸がkameなので、tsuruが6、kameが4のところで、グラフが交わっているね。

グラフ⑨



3Dグラフを描く

PlotやContourPlotには、3D版のPlot3DやContourPlot3Dもある。詳しい説明は省略するけど、例えば、こんなかっこいいグラフも描けるよ。



ここまで入力したノートブックは、後で呼び出せるように「File」メニューの「Save」で保存しておこう。

今回は、マセマティカでグラフを描き、グラフの交点を求ることと、連立方程式の関係を見てきた。より詳しい情報は、マセマティカを起動したときに表示される、情報へのリンクが表示されたウィンドウから調べられるので、ぜひ見てみよう。