

# グラフ

天野研 修士2年 松本秀夫

5月11日金曜日

## 1 gnuplot

gnuplotとは、コマンド入力型の対話式グラフ描画プログラムです。割と簡単なコマンドで多様なグラフを書くことができるほか、いろいろな形式のファイルとしてグラフを出力することもできるので、 $\text{T}_{\text{E}}\text{X}$ を用いた論文にグラフを張り付ける際などに非常に役に立ちます。

### 1.1 起動と終了

gnuplotの起動は、kterm等の端末上で

```
% gnuplot
```

と入力すればOKです。以下のような画面が出るはずで

```
G N U P L O T
Unix version 3.7
patchlevel 1
lastmodified Fri Oct 22 18:00:00 BST 1999

Copyright(C) 1986 - 1993, 1998, 1999
Thomas Williams, Colin Kelley and many others

Type 'help' to access the on-line reference manual
The gnuplot FAQ is available from <http://www.ucc.ie/gnuplot/gnuplot-faq.html>

Send comments and requests for help to <info-gnuplot@dartmouth.edu>
Send bugs, suggestions and mods to <bug-gnuplot@dartmouth.edu>
```

Terminal type set to 'x11'

```
gnuplot>
```

この最後のgnuplot>がプロンプトになり、ここにいろいろなコマンドを入力していくことになります。また、Terminal type set to 'x11'とあるように、デフォルトではグラフは全てXを用いて別ウィンドウで表示されます。

終了時には

```
gnuplot> quit
```

もしくは

```
gnuplot> exit
```

と入力します。

### 1.2 グラフを描く(plot,replot)

グラフを描くコマンドは'plot'です。試しに $y = \sin x$ を描いてみましょう。

```
gnuplot> plot sin(x)
```

山3つほどの $\sin x$  グラフが得られると思います。x軸やy軸は勝手に設定されます。x軸やy軸の範囲を指定してグラフを描きたいときには

```
gnuplot> plot [-3:3][-1:1] sin(x)
```

と入力します。前の括弧内がx軸、後ろの括弧内がy軸に対応しています。x軸のみを指定するときには後ろの括弧ごと省略することができますが、y軸のみを指定するときには前の括弧内だけを省略して入力します。

```
gnuplot> plot [-pi:pi] sin(x)
```

```
gnuplot> plot [][-2:2] tan(x)
```

複数のグラフを重ねて表示したいときは

```
gnuplot> plot sin(x)
```

```
gnuplot> replot cos(x)
```

もしくは

```
gnuplot> plot sin(x),cos(x)
```

と入力します。'replot'は、文字通り「再描画」するコマンドで、直前に表示したグラフをもう一度表示します。先程のように別のグラフを重ねたいときや、設定を変更した上でもう一度同じグラフを表示したい場合などに用います。

gnuplotでは、関数を指定してグラフを描くだけでなく、データファイルを読み込んでそのデータをグラフにすることも当然可能です。例えば以下のようなデータファイルがあるとします。

```
%cat data
# x      y1      y2
-10.0000 -5.44021  8.39072
-9.50000 -0.71394  9.47314
      :      :      :
 9.50000 -0.71394 -9.47314
10.00000 -5.44021 -8.39072
```

このファイルを読み込んでグラフを描く場合、

```
gnuplot> plot "data"
```

と入力すると、ファイル"data"の1列目をx軸、2列目をy軸としたグラフがドットで描かれます。3列目をy軸として表示したい場合は

```
gnuplot> plot "data" using 1:3
```

と入力します。さらに、ドットではなく線グラフにしたい場合は

```
gnuplot> plot "data" with lines
```

という風にします。逆にドットで描きたいときには'lines'を'points'に、どちらも合わせて描きたいときには'linespoints'にすればOKです。

### 1.3 グラフ環境の設定・確認 (set,show)

set コマンドは、グラフにタイトルやラベルを付けたリグラフの表示範囲や表示方法をあらかじめ変えたりすることができます。以下、主なオプションを紹介します。

- set title 'title'  
グラフにタイトルを付けます。
- set xlabel|ylabel 'label'  
グラフのx軸またはy軸にラベルを付けます。

- `set xrange|yrange [from:to]`  
グラフのx軸またはy軸の表示範囲を設定します。
- `set data|function style style`  
グラフの表示方法を設定します。dataは読み込んだデータ、functionは関数の場合のグラフ表示をそれぞれstyle(linesやpointsなど)にします。
- `set grid|nogrid`  
グリッド(目盛線)の表示/非表示を設定します。

設定した内容は、'show'コマンドで確認することができます。例えば

```
gnuplot> show xrange
gnuplot> show grid
```

等とすると、それぞれx軸の表示範囲やグリッドのON/OFFが表示されます。また、

```
gnuplot> show all
```

で、現在の設定状況を全て見るすることができます。

## 1.4 ファイルへの出力

gnuplotでは、Xによる画面へのグラフ表示のほかにグラフをファイルとして出力することもできます。この場合は、出力端末を変更する'set terminal'コマンドを用います。画面に表示されているグラフをpostscript形式でファイル'graph.ps'に出力させる場合は以下のようにします。

```
gnuplot> set terminal postscript
Terminal type set to 'postscript'
Options are 'landscape noenhanced monochrome dashed defaultplex "Helvetica" 14'
gnuplot> set output 'graph.ps'
gnuplot> replot
```

こうすることで、カレントディレクトリに'graph.ps'が作成されます。出力をXに戻すときは

```
gnuplot> set output
gnuplot> set terminal x11
```

と入力します。その他の出力端末としては以下のようなものがあります。

- `set terminal postscript eps:eps` ファイル形式で出力します。
- `set terminal tgif:tgif` で読み込める形式で出力します。
- `set terminal png:png` ファイル形式で出力します。
- `set terminal`: 使用できる出力端末の一覧が表示されます。

## 1.5 3次元のグラフを描く (splot)

3次元のグラフを描くには'splot'コマンドを用います。オプションは基本的に'plot'コマンドと同じです。また、'zrange'や'zlabel'等でx軸y軸と同様な設定も行えます。

3次元グラフの場合、視点を変えてグラフを表示させることができます。視点を変えるのは'set view'コマンドです。

```
gnuplot> set view 60,30
```

とした場合、xy2次元のグラフからx軸を60度、z軸を30度回転させたアングルでの表示になります。詳しくは、実際に自分で値を変更させてから確認してみてください。

## 1.6 ヘルプを見る (help,?)

gnuplotでは、'help'コマンドでいろいろなコマンドの詳しい説明を見ることができます('?'で代用可)。例えば

```
gnuplot> help plot
```

もしくは

```
gnuplot> ? plot
```

とすると、'plot'コマンドの使用法について詳しい説明が出てきます。参照したコマンドにサブピックがある場合は、ヘルプの最後にその一覧が出てきます。'plot'の場合、

Subtopics available for plot:

acsplines	bezier	csplines	datafile
:	:	:	:
unique	using	with	

(END)

となります。ここで'q'を押すと最後の行に

Subtopic of plot:

と表示されるので、上の選択肢から選んで入力すればさらにそのヘルプが参照できます。ここで'?'を入力するともう一度サブピックの一覧が参照でき、何も入力せずにリターンを押せばその段のヘルプを終了します。何か判らないことがあったらガシガシ利用しましょう。

## 1.7 おまけ

gnuplotのコマンドは、区別がつく限り後ろの文字を省略してもちゃんと認識してくれるようになっています。例えば

```
gnuplot> plot "data" with lines, "data" using 1:3 with points
```

```
gnuplot> p "data" w l, "data" u 1:3 w p
```

の2つはどちらも同じ結果がちゃんと得られます。'linespoints'なんかは、'linesp'という風に省略が可能です。

## 1.8 実習

今回紹介した関数のグラフを範囲を変えながら実際に表示させてみましょう。指数関数(exp)や対数関数(log)を使ったものや、半径1の円(または球)のグラフなども描いてみてください。また、松本のホーム(/u/hideo)に'data'というファイルがあるので、それを持ってきて一つはドットによるグラフ、もう一つはドット付き線グラフの形で重ねて表示させてみてください。さらにこのファイルから3次元グラフを描いてみましょう。

作ったグラフを $\text{T}_{\text{E}}\text{X}$ で読み込める形にして $\text{T}_{\text{E}}\text{X}$ に張り付けてみましょう。

## 2 計算ツール

計算ツール(いわゆる計算機)にはxcalcのような実際の電卓を模したグラフィカルなもの、bc,dcのようなコマンドラインで計算を行うようなものがあります。xcalcは実際に使ってみればすぐ判ると思うので、ここではbcとdcについて説明したいと思います。

### 2.1 bc(Business Calculator)

bcは、任意の精度の数値を扱うことができ、C言語の文法によく似た形の入力を対話的に実行する言語です(マニュアルより)。実行すると

```
% bc
bc 1.04
Copyright (C) 1991, 1992, 1993, 1994, 1997 Free Software Foundation, Inc.
This is free software with ABSOLUTELY NO WARRANTY.
For details type 'warranty'.
```

と表示され、ktermの右端にカーソルが来ます。ここでいろんな計算ができます。デフォルトでは小数点以下の桁数は0に設定されているので、'scale=n'と入力することで小数点以下の桁数をn桁に指定することができます。また、aからzまでの変数に数値を代入して計算することもできます。終了するときは'quit'と入力するかC-dを押します。

```
% bc
1+3 (入力)
4
1+2*3 (入力)
7
2/3 (入力)
0
scale=4 (入力)
2/3 (入力)
.6666
a=1 (入力)
b=4 (入力)
c=a+b (入力)
c (入力)
5
c-3 (入力)
2
quit (入力)
%
```

bcでは、/\*...\*/や#...によるコメントや関係演算子などを使用することができます。

## 2.2 dc(Desk Calculator)

dcは、逆ポーランド形式の無限精度の計算が行える卓上計算機です(マニュアルより)。「逆ポーランド式」とは、括弧の優先順位などを考えずに先頭から順番に処理をしていく方式です。dcも起動するとbcと同じような入力画面になります。

dcでは、入力した数字は画面に表示せず、スタックに保存します。入力を繰り返すと、スタックに積み上げられます。スタックに対する操作はコマンドによって行います。

**p** スタックを変更せずに、先頭の値を表示します。

**P** スタックの先頭の値を表示し、スタックから取り出します。

**f** スタックを変更せずに、スタックの内容を全て表示します。

**c** スタックを消去し、空にします。

**d** スタックの先頭の値を複製し、スタックに積みます。

**r** スタックの先頭の値と2番目の値を入れ替えます。

**+** スタックから2つの値を取り出し、足し合わせて結果をスタックに積みます。

**-** スタックから2つの値を取り出し、最初に取り出した値を2番目に取り出した値から引きます。結果はスタックに積みます。

\* スタックから2つの値を取り出し、掛け合わせて結果をスタックに積みます。

/ スタックから2つの値を取り出し、2番目に取り出した値を最初に取り出した値で割ります。結果はスタックに積みます。

% スタックから2つの値を取り出し、/で計算した割り算の余りを結果としてスタックに積みます。

~ スタックから2つの値を取り出し、/で計算した割り算の商をまずスタックに積み、続いて余りをスタックに積みます。

^ スタックから2つの値を取り出し、最初の値を指数、2つ目の値を底として指数計算をします。結果はスタックに積みます。

v スタックから値を取り出し、平方根をとってスタックに積みます。

小数点以下の桁数は入力された数字に依存します。デフォルトの精度を変えるにはkコマンドを用います。これは、スタックの先頭から値を取り出し、その数値の精度を標準の精度として設定します。

## 2.3 で。

結局どんな風に計算を行うのかというと、例えば(2+3)を計算するときは2、3、+の順に入力します。結果はスタックに積まれるので、最後にpを押せば答えの5が表示されます。同じように、((1+2)\*(5-3))は1、2、+、5、3、-、\*の順に入力します。

## 2.4 実習2

bc,dcを用いて $\frac{-5 \pm \sqrt{5^2 - 4 \times 2 \times 3}}{2 \times 2}$ を計算してみましょう。マイナスの場合は精度にも注意。あと直径5 cmの球の表面積と体積とか誰か計算してみてください。