

計算結果の可視化

計算結果を分かりやすいグラフや画像の形に表すことを**可視化 (visualization)**という。

GNUplot によるグラフの作成

GNUplot は (x, y) データなどをグラフに表示するソフトウェアである。Excel を使っても同じことができるが、GNUplot は Linux における標準的なソフトである。ここでは GNUplot で、sine.dat や mysine.dat をグラフに表示する作業を行う。

1. グラフの画面への表示

sine.dat などが出来ていることを確認する。

```
ls -l
```

と入力。

gnuplot を起動する

```
gnuplot
```

と入力。プロンプトが“gnuplot>”に変わるのを確認する。以下はほんのさわりであるので、gnuplot でどんな作図が出来るのかをインターネットで検索して調べてみよう。

“sine.dat”のグラフを表示させる

```
gnuplot> plot "sine.dat"
```

これで画面にグラフが表示される。3カラム以上ある場合はどのカラムを x 軸、 y 軸とするか選択する。

```
gnuplot> plot "mysine.dat" using 1:2 title "internal func"
```

```
gnuplot> replot "mysine.dat" using 1:3 title "Taylor Ser."
```

replot コマンドはグラフを重ね書きするコマンドである。

レポートに図を貼り付ける場合は、ウィンドウをキャプチャーする。

2. グラフのポストスクリプトファイルへの保存

以下は、グラフのみを印刷する場合の手順である。この授業では行う必要がない。

画像の出力先をファイルに指定する。GNUplot では直接ポストスクリプトへ出力できる。

```
set output "graph1.ps"
```

```
set terminal postscript
```

と入力すると出力先が画面からポストスクリプトファイルに変更される。この後、上の `plot` コマンドを実行する。

`gnuplot` を終了する。

```
quit
```

と入力。

3. ポストスクリプトファイルの PDF への変換

出力のファイルができていないか確認する。

```
ls -l
```

出力のポストスクリプトファイル `graph1.ps` が出来ているか確かめる。

```
ps2pdf graph1.ps
```

で PDF に変換したファイルが `graph1.pdf` に保存される。

4. 出力されたファイルをプリントする

```
lpr graph1.ps
```

と入力する。`lpr` はポストスクリプトファイルをプリンターに送るコマンド。

プログラムなどのテキストファイルの出力には `a2ps` を使う。例えば、

```
a2ps sine.f
```

とする。

Gnuplot のコマンド

`plot "datafile"` : 点でグラフのプロット

`plot "datafile" with line` : 線でグラフをプロット

`plot "datafile" using 1:2` : データの1コラム目を x 軸、2コラム目を y 軸としてグラフを
プロット

`plot "datafile" using 1:3 title "example"` : データの1コラム目を x 軸、3コラム目を y
軸としてグラフをプロット、右上の注釈に example と名前を付ける。

`replot "datafile2"` : 別のファイルのグラフを重ねてプロットする

`set title "Graph1"` : グラフに Graph1 とタイトルを付ける

`xlabel "axis1"` : x 軸に axis1 とラベルを付ける

`set yrange [-1.0:2.0]` : y 軸の範囲を-1.0 から 2.0 までに指定する

`set xtics 1.0` : x 軸に 1.0 ごとに目盛りを付け、値を入れる。

`set mxtics 5` : 目盛りの間を 5 等分して数値のない小目盛りを入れる。

`set terminal postscript` : グラフの出力先をポストスクリプトファイルに指定する

`set output "graph1.ps"` : グラフの出力先のファイル名を graph1.ps に指定する

`save "graph1.plt"` : Gnuplot で作成したグラフを保存する