

教育用計算機の利用を想定した  
3次元データの簡便な可視化手法の提案—gnuplotの利用

野口克行\*<sup>1</sup>

A method of visualization of three-dimensional data by “gnuplot”  
by  
Katsuyuki Noguchi\*<sup>1</sup>

Abstract

We have proposed a simple and easy method plotting three dimensional data which are frequently appeared in space and Earth science fields by using “gnuplot”, which is a free and one of the popular software for plotting data and is usually installed on most of the educational computer systems in universities and institutes. As the data should be reformed into appropriate formats before the input for gnuplot, we describe how to reform the data in this paper. We show a few examples of the applications of the method proposed.

Keywords: three-dimensional data, visualization, plot, gnuplot

概要

宇宙・地球科学において、3次元データを可視化する機会が頻繁にある。このような場合、第1変数と第2変数で2次元平面を取って第3変数の値を等高線（コンター）や色で表すことが多い。3次元データを可視化可能なソフトウェアは数多く存在すると思われるが、無償かつ初心者でも容易に操作できるものとなると多くはない。本論文では、ほぼ全ての教育用計算機にインストールされている（または容易にインストールが可能）と思われる gnuplot を利用し、できるだけ簡便に3次元データを可視化できるような手法を提案する。具体的には、gnuplot の入力に適うようなデータフォーマットの変換方法を説明する。応用例としては地球および惑星大気科学分野のデータを挙げるが、本論文で示すプロット手法の原理自体は自然科学の幅広い分野のデータに応用できるものである。

キーワード: 3次元データ 可視化 プロット gnuplot

1 はじめに

近年、ハードウェア・ソフトウェア両面から、宇宙・地球観測分野のデータの質と量が劇的に向上しつつあり、膨大なデータの蓄積が進んでいる。一方で、そのようなデータを詳細に解析する際には、人間が実際にデータをプロットして可視化することが必要である場合が多い。そのため、大量のデータに対応するには、データの可視化の負担をできるだけ小さくし、効率よく、安易かつ低コストでプロット

---

\* 平成 27 年 12 月 17 日受付 (Received December 17, 2015)

<sup>1</sup> 奈良女子大学大学院自然科学系 (Faculty of Science, Nara Women's University)

できる環境の構築が望ましい。

大学等の教育現場においては、UNIX (Linux) をベースとした計算機環境が整備されていることが多く、そのような計算機環境においてデフォルトで動作する可視化システムがあれば、高価なワークステーションやライセンス料の高いソフトウェアを用いること無しに学生が気軽にそのようなデータに初等学年から親しむことが可能になる。

gnuplot は、そのような条件に合致するデータプロット用ソフトウェアと言える。ユーザとのインタラクティブなコマンド入力システムを提供する一方で、スクリプト化にも対応しており、初等利用者・高度利用者共に満足し得るソフトウェアの一つである。インストールに関しても、多くの Linux ディストリビューションにおいてパッケージ化されているため簡便であり、また Windows や Mac OS でも利用可能である。マニュアルは、全て gnuplot の Web ページ (<http://www.gnuplot.info/>) から利用可能である<sup>1)</sup>。

一般的に、3次元データのプロットは2次元データとは異なり工夫が必要である。例えば、ある高度面における気温の経度・緯度分布をプロットする場合は、第1変数(x)を経度、第2変数(y)を緯度、第3変数(z)を気温とし、x-y平面上にzの値を色や等高線などを用いて表現することが多い。本論文では、「色(あるいは白黒の濃淡等)による塗りつぶしを用いて、3次元データを2次元面上に表現すること」を(3次元データの)可視化と呼ぶことにする。このような図のプロットは、データの格子化や欠損値の対応等が必要となり、初学者には困難なことが多い。本論文では、gnuplot マニュアル<sup>[1]</sup>を参考とし、3次元データの可視化をできるだけ簡易かつ元のデータに忠実になるようにgnuplotで行なう手法を示す。なお、ここで示す手法は、著者の所属する奈良女子大学において学生の計算機演習などで実際に教えているものである。

## 2 gnuplot による簡易な3次元データ可視化の手法

gnuplot では、pm3d という機能を用いて、以下のようなコマンド列を打つことで3次元データファイル(例としてファイル名を data.dat とする)をプロットすることが可能である。

```
gnuplot> set pm3d
gnuplot> set view map
gnuplot> splot "data.dat"
```

gnuplot による3次元データの描画原理を図1に示す。gnuplot は、与えられたデータセットにおいて隣接する4点を探し出し、4点の平均値等(後述)に相当する色を用いて、4点を結ぶ矩形状に塗りつぶすように描画を実行する。3点では描かれない。この仕組みを利用して、扱いやすさと正確さのバランスの取れた描画をユーザの望む形で実現することを考える。

まず、単純にこの gnuplot の描画機能を利用することを考える。最低で4点揃っていれば良い訳であるから、例えば、座標が等間隔で並ぶ格子状のデータを入力すれば、gnuplot は各格子点のデータを4点ずつ用いながら平均し、その平均値に対応する色で各点を結ぶ正方形・長方形状に塗りつぶす。これは、gnuplot に自動的に各点を補間させているイメージである。そこで、この方法を「gnuplot の自動補間機能を用いる方法」と呼ぶことにする。

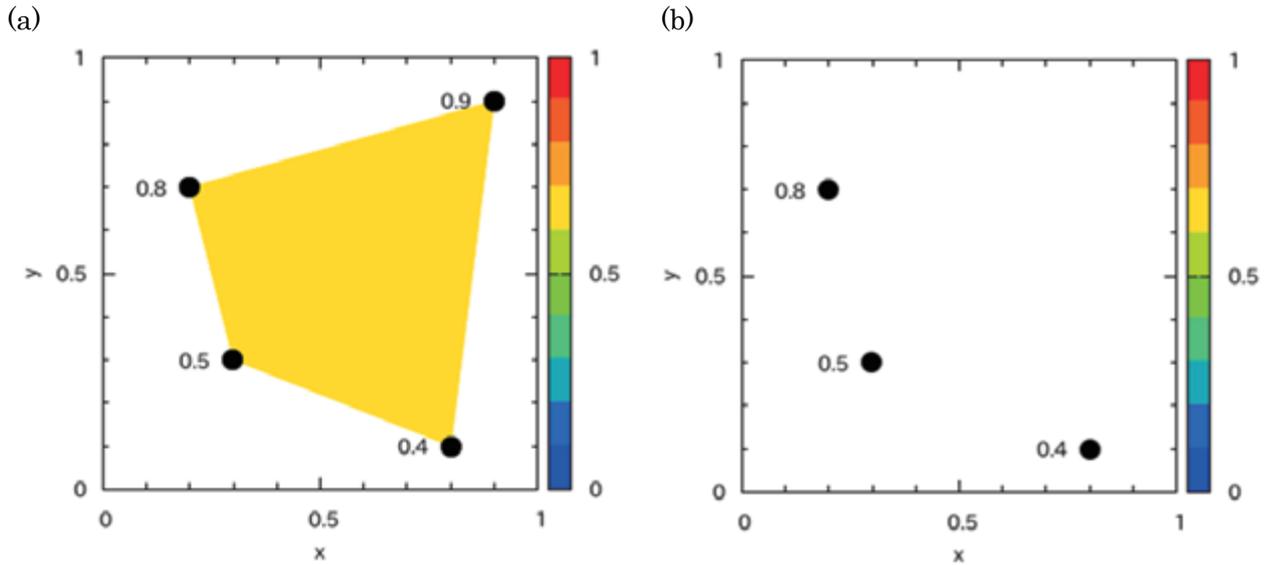


図1 gnuplotによる3次元データ描画の原理。x-y平面上に各点をプロットし、zの値を色で表現する。図中の数字は、各点のzの値である。(a)4点の場合。z値の平均は0.65  $(=(0.4+0.5+0.8+0.9)/4)$ となるが、これは黄色に相当する。この色で4点を結ぶ矩形上に塗りつぶされる。(b)3点の場合(このとき、色の塗りつぶしは実行されない)。

一方で、gnuplotに勝手に補間させるのではなく、あるデータ値を持つ領域に対して、その位置においてそのデータの値を厳密に反映した色で描画したい場合も存在するであろう。例えば、データ点が1点しかない場合、上述のgnuplotの補間機能を用いる方法ではプロットがなされない。このような場合は、例えばその点の周囲を囲むように4点取り、その4点を結んでできる矩形領域をzの値に対応する色で塗りつぶせば良いことになる。これは、データ点を一点ずつ有限の大きさにピクセル化することに相当する。そこで、この方法を「データ点をピクセル化する方法」と呼ぶことにする。

## 2.1 gnuplotの自動補間機能を用いる場合

gnuplotの自動補間機能を用いて3次元データを可視化する場合には、事前にデータを整形することが必要となる。具体的には、第1変数(x)についてまずソートし、次に第2変数(y)についてソートをする。その際、xが次の値に変わる前に空行を一行入れる。例として、付録1に示すようなデータセットをプロットすることを考える。このデータを整形して図2に示すようにソート・空行挿入を施してからgnuplotに読み込ませると、gnuplotは図3に示すように隣接する4点のz値を単純平均(デフォルトの場合、corners2colorオプションで変更可能)してから描画する。このようなデータ整形は、Linuxの各種コマンド(sortやawk等)を用いればそれほど手間を掛けずに実現が可能であるので、格子状データを簡易的にプロットするには実用的な方法である。ただし、次に述べるような注意が必要である。

まず、データ点数が少ない場合やデータ値の変動が激しい場合は、gnuplotによって補間された際に実際のデータの様子とは大きく異なったイメージの図が描画される可能性がある。図3の例においても、実際のデータにはカラーバーの最小値(0:濃紺)と最大値(1:赤)に相当する値が含まれているが、それらに対応する色はプロットには現れていない。データ値の変動が激しい場合には、値が平滑化され、

(a)	(b)
x y z	x y z
0.5 0.5 0.05	0.5 0.5 0.05
0.5 1.5 0.25	0.5 1.5 0.25
0.5 2.5 0.45	0.5 2.5 0.45
0.5 3.5 0.65	0.5 3.5 0.65
1.5 0.5 0.15	
1.5 1.5 0.35	1.5 0.5 0.15
1.5 2.5 0.55	1.5 1.5 0.35
1.5 3.5 0.75	1.5 2.5 0.55
2.5 0.5 0.25	1.5 3.5 0.75
...	
...	2.5 0.5 0.25
...	...

図 2 格子データの整形の例。(a) 元のデータ、(b) 整形後のデータ。第 1 変数 (x)、第 2 変数 (y) の順にソートし、x の値が変わる際には空行を 1 行入れる。

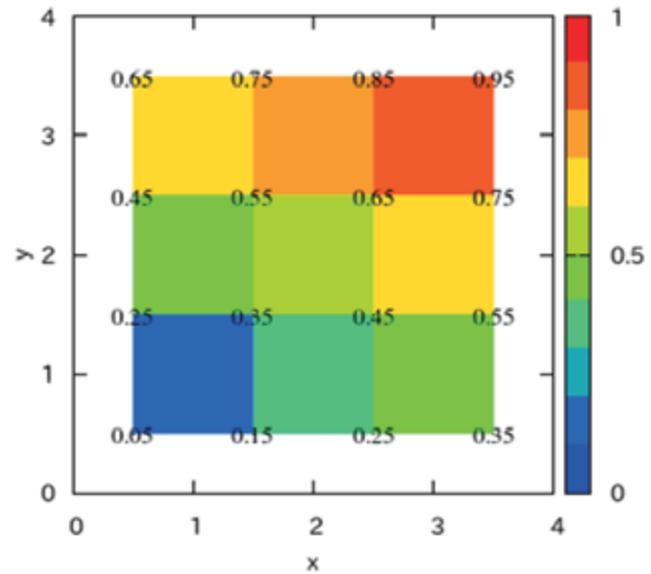


図 3 gnuplot の補間機能を用いたプロットの例。図 2(b)のデータを用いている。図中の数字は、各点の第 3 変数 (z) の値である。

元データの変動が十分に表現されない可能性がある。なお、先述の `corners2color` オプションを用いると 4 点の最大値・最小値・どれか 1 点の値・メジアン値などを表示することが可能であるが、4 点全ての情報を適切に表現しきれないという意味では同じである。

また、データに欠損がある際にうまく対応できない場合があるという問題が挙げられる。データの欠損には二通りあり、ある値で欠損値が定義されている場合 (gnuplot は、任意の文字列を欠損値として定義し、この値をプロットの対象から除外することが可能である) と、座標ごと欠落している場合とが考えられる。例として、付録 1 に示す欠損のあるデータをプロットした結果を図 4 に示す。図 4(a)のように欠損値が定義されている場合は、その点の周辺では矩形を作らずに色で塗りつぶされない。図 4(b)のように座標ごとデータから欠落している場合は、gnuplot はその周囲にある点を変則的に 4 点ずつ結ぶことで色を塗りつぶそうとする。これが自動的に補間された結果となるが、このようなプロットが必ずしもユーザの望むことかどうかは難しいところであろう。欠損値が定義されている場合でも、たった 1 点の欠損値に対して 4 つのピクセル分のデータがプロットされないことになるため、欠損が多い場合には多くのデータがプロットされないことにもなり得る。

したがって、gnuplot の自動補間機能を用いる際には、データが格子状に揃っており、データ点数がある程度多くて、値の変動が激しくなく、欠損値も無いかごく少ないことが望ましいといえる。特に、座標ごと欠落している場合は自動的に不規則に塗りつぶされる可能性があるため、注意が必要である。

## 2.2 データ点をピクセル化する方法

前節の gnuplot の自動補間機能を用いる方法では、必ずしも元データを忠実に描画することができない場合があり、またデータ欠損があるとユーザが想定しないような図が得られる恐れがあった。そこで、

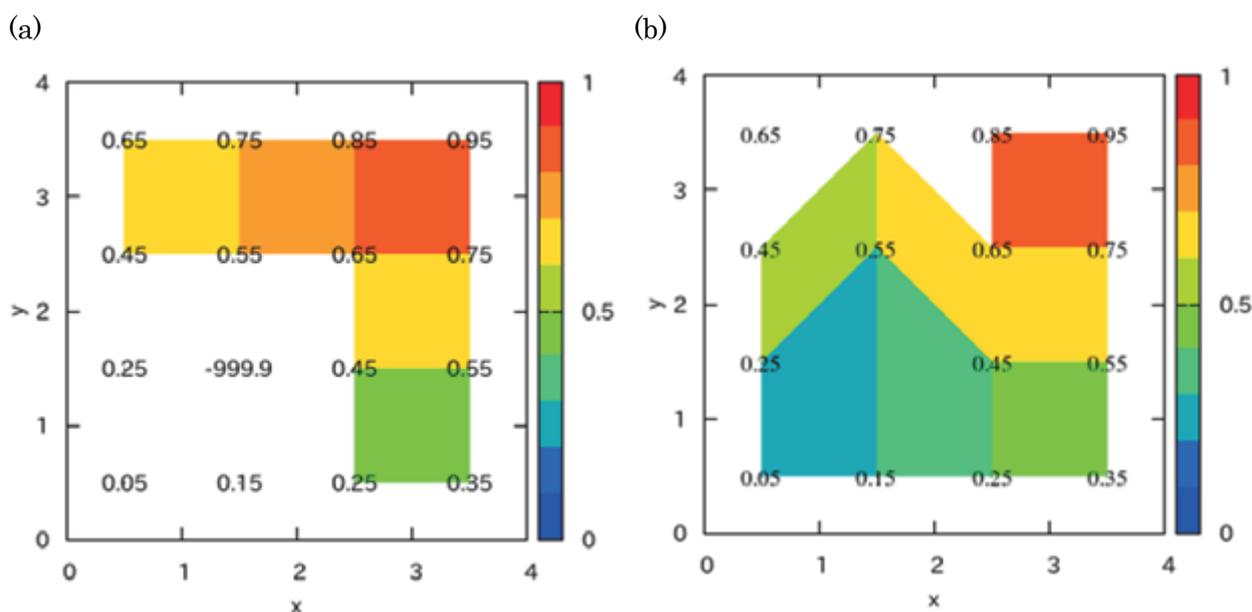


図4 gnuplot の補間機能を用いた場合の、欠損があるデータのプロットの例。(a) 欠損値が定義されている場合 (例では-999.9)。(b)座標自体がデータに存在しない場合。図中の数字は、各点の第3変数(z)の値である。(a)と(b)のいずれにおいても、座標(1.5,1.5)のデータが欠損になっている(付録1のデータを参照)。

データに忠実に描画することを考える。前述したように、ある1点のデータに対し、その点を囲むような形でピクセルを描けばよい。図5に、このようなデータ変換の方法の例を示す。この例では、ユーザが予めピクセルの二辺の長さ(LxとLy)を固定値として与えて、元のデータ点をその大きさのピクセルに変換するようにしている。データ1点当たり4点のデータを生成し、ピクセルの下側の2点を最初に記述した後に1行の空行を入れ、次にピクセルの上側の2点を記述して、2行の空行を入れる。もし、ピクセルの辺の長さを個々の点毎に変えたければ、データ列をもう2列増やし、各データ点にLxとLyの値を与えればよい。

このようなデータ変換の後にgnuplotに入力すると、図6のようなプロット結果が得られる。この結果を図3と比較すると、ピクセル化したほうがより元のデータに忠実にプロットされていることが分かる。

ここまでは格子状に揃っているデータを例に挙げてきたが、データをピクセル化する手法を応用すれば、必ずしも格子状ではなく不均質なピクセル形状を持つデータでもプロットが可能である。4つの点のx・y座標情報とその点に対するz値があれば、上述の方法に準じる形で2点ごとに空行を1行ないし2行加えることで、そのピクセルを忠実にプロットすることが可能である。図7に例を示す。この例では、人工衛星による大気汚染物質(二酸化窒素)の観測結果をプロットしている。元のデータはセンサの視野の四隅に相当する4点の経度・緯度情報とその視野内での二酸化窒素量を含んでおり、図5に準じる形でデータ変換をすることでセンサ視野ごとの二酸化窒素量をプロットすることに成功している。なお、付録2に、地図(海岸線)をgnuplotでプロットする手法を示した。地図をプロットするにはGMTが有名であるが、海岸線のデータさえ入手できれば、gnuplotでも比較的容易に地図をプロットすることが可能である。

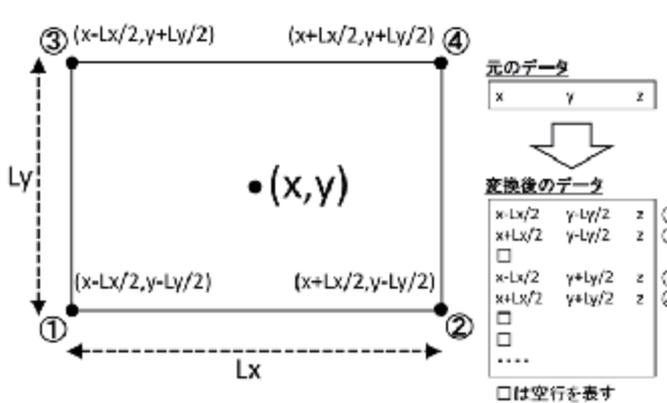


図5 gnuplot でプロットするためのデータ変換（ピクセル化）の例。元のデータ点  $(x, y, z)$  を中心として、辺の長さが  $Lx$  と  $Ly$  である長方形を作るようにする。元のデータ 1 点当たり計 4 点を記述することになるが、最初の二つの点の組（①と②）の次には一行空行を入れ、次の二つの点の組（③と④）の後には、二行空行を入れる。

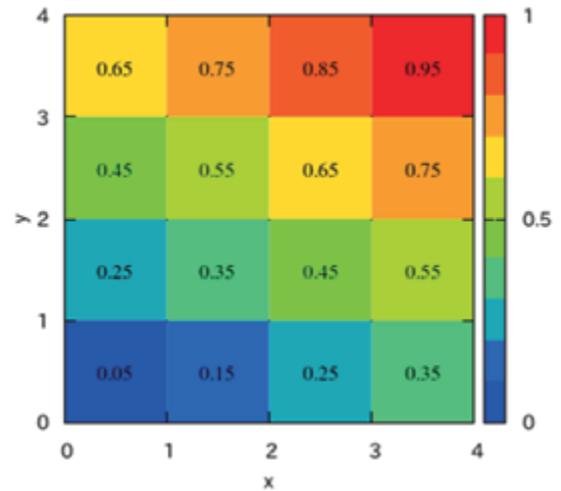


図6 データ点をピクセル化した場合のプロットの例。付録1のデータを用いている。図中の数字は、各点の第3変数 ( $z$ ) の値である。

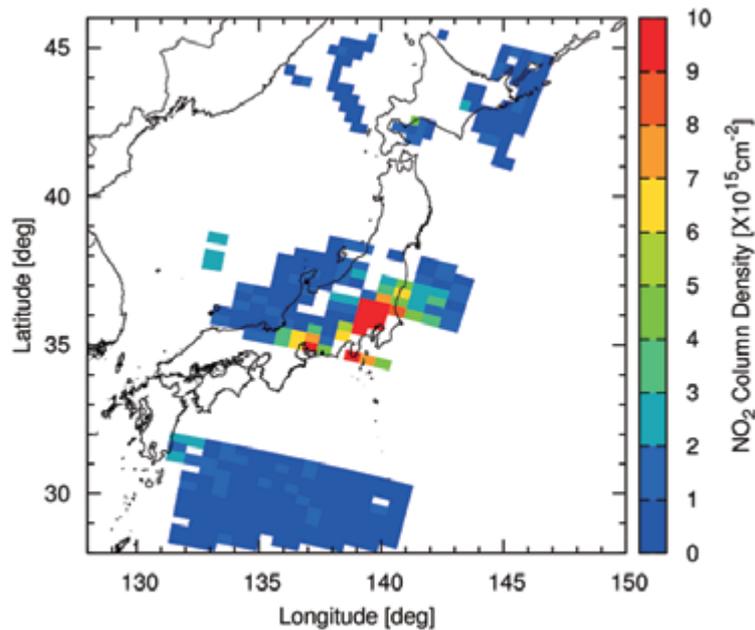


図7 格子状ではない不均質なピクセル形状をもつデータを gnuplot でプロットした例。データを取得したセンサの視野が緯度経度の格子に対して傾いており、大きさもそれぞれで異なっている。欧州宇宙機関の地球観測衛星 EINVISAT に搭載された SCIAMACHY センサによる二酸化窒素気柱量の観測結果（2009年2月7日）。地図のプロット方法は付録2を参照のこと。

## 2.3 画像データとしてプロットする場合との比較

これまでに述べた方法の他に、`gnuplot` では `image` オプションを用いて 3 次元データを画像データとしてそのまま値を表示する機能がある。

```
gnuplot> plot "data.dat" with image
```

この場合、データは格子状に揃っていることが必要であり、また欠損値もうまく処理することができない（つまり、付録 1 に示した 3 つのデータのうち、最初のものしかプロットできない）。また、図 7 のような不均質なピクセルサイズのデータもプロットすることはできない。そのため、数値シミュレーション結果などデータが格子状に揃っていてかつ欠損値がない場合を除けば、画像データとしてプロットするのは難しいといえる。

## 3 まとめ

本稿では、`gnuplot` を用いて 3 次元データを可視化する手法を提案した。簡易的に行なうには、`gnuplot` の自動補間機能を利用する。3 列の値が入ったデータファイルを用意し、第 1 変数、第 2 変数の順でソートし、第 1 変数の値が変わる毎に空行を入れる。これにより、`gnuplot` は隣接する 4 点ごとに矩形を作って第 3 変数の値を平均等した上で矩形をその値に相当する色で塗りつぶす。ただし、この方法は格子状データであること、データ数が比較的多いこと、欠損値が少ないことなどが必要である。

元のデータに対してより忠実にプロットするには、データ点をピクセル化する変換を実施してから `gnuplot` に入力する。もし、格子状のデータであれば、ピクセルとなる長方形（または正方形）の中心にデータ点があるように 4 点の座標を計算し、1 点あたり 4 行のデータとする。ただし、最初の 2 行の後に空行を 1 行入れ、最後の 2 行の後には空行を 2 行入れる。この方法を応用すれば、必ずしも格子状のデータではない不均質なピクセルサイズのデータに対しても、ユーザが 4 点の座標を与えることにより 1 ピクセルずつ忠実に描くことが可能になる。

今後ますます増大する地球・宇宙観測データに対して、容易かつ低コストで可視化を行なえる事はデータ解析を効率化する上で不可欠である。初学者にも比較的扱いやすいプロット用のソフトウェアを採用し、プロットの方法を分かりやすくかつ実践的に解説することで、データ解析の発展に寄与していきたいと考えている。

## 4 謝辞

図 7 の SCIAMACHY データは、ドイツ・ブレーメン大学の Andreas Richter 博士から提供頂いた。

## 参考文献

- 1) Thomas Williams and Colin Kelley, “gnuplot 4.4 An Interactive Plotting Program”, [http://www.gnuplot.info/docs\\_4.4/gnuplot-ja.pdf](http://www.gnuplot.info/docs_4.4/gnuplot-ja.pdf), 2011.

## 付録1 本文で用いたデータ

格子状データ	欠損値定義のあるデータ	一部座標が欠落しているデータ
2.5 2.5 0.65	3.5 1.5 0.55	2.5 0.5 0.25
1.5 0.5 0.15	0.5 0.5 0.05	0.5 0.5 0.05
2.5 3.5 0.85	1.5 1.5 -999.9	3.5 2.5 0.75
3.5 1.5 0.55	0.5 3.5 0.65	0.5 2.5 0.45
0.5 0.5 0.05	3.5 2.5 0.75	1.5 0.5 0.15
1.5 1.5 0.35	0.5 1.5 0.25	3.5 3.5 0.95
0.5 1.5 0.25	1.5 3.5 0.75	1.5 3.5 0.75
3.5 3.5 0.95	2.5 0.5 0.25	2.5 1.5 0.45
0.5 3.5 0.65	2.5 2.5 0.65	1.5 2.5 0.55
1.5 2.5 0.55	0.5 2.5 0.45	0.5 1.5 0.25
0.5 2.5 0.45	1.5 2.5 0.55	2.5 3.5 0.85
1.5 3.5 0.75	2.5 3.5 0.85	3.5 0.5 0.35
2.5 1.5 0.45	1.5 0.5 0.15	2.5 2.5 0.65
3.5 0.5 0.35	3.5 0.5 0.35	0.5 3.5 0.65
3.5 2.5 0.75	2.5 1.5 0.45	3.5 1.5 0.55
2.5 0.5 0.25	3.5 3.5 0.95	

## 付録 2 gnuplot の 3 次元データプロット時における地図（海岸線）のプロット

gnuplot で地図（海岸線）をプロットするには、まず海岸線のデータが必要となる。例えば、米国の NOAA の Web ページ等からダウンロードが可能である。データには、緯度・経度点が含まれていればよい。実際のプロットの際には、gnuplot のコマンドプロンプトにおいて以下のようなコマンドを打つとよい。

```
gnuplot> set pm3d
gnuplot> set view map
gnuplot> splot "data.dat", "coast.dat" using 1:2:0 with line title ""
```

この例において、「data.dat」というのが 3 次元プロットを行ないたいデータファイル名、「coast.dat」が海岸線データファイル名である。海岸線データには、以下に示すように第 1 変数に経度、第 2 変数に緯度を格納し、島等の閉曲線を描くためには、閉曲線と閉曲線のデータの間に行空行を入れる。

```
114.221346    10.21988
114.221346    10.223401
. . .
114.221639    10.219587
114.221346    10.21988

114.242174    10.176757
114.242174    10.178224
. . .
114.242467    10.17617
114.242174    10.176757

114.338688    10.738237
. . .
```