

株式会社ショーワ著

版 発行

# 目次

## 第 1 章

# Graphviz

Graphviz は、人や物のネットワーク、フローチャート、系統樹などに適したグラフ描画ツールです。Windows、Mac、Linux など多くの OS で利用できます。

まずシンプルな例から見てみましょう。

次の図は夏目漱石『坊ちゃん』の人間関係をグラフにしたものです。

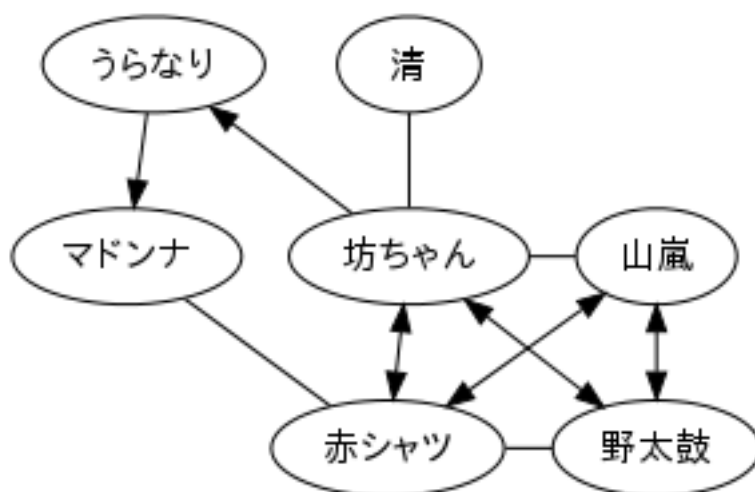


図 1.1 『坊ちゃん』の人間関係図

Graphviz のデフォルト設定では、この図のようにそれぞれの事物（**ノード**と呼びます）を楕円で描画し、ノードとノードを線（**エッジ**と呼びます）で結びます。デフォルトではエッジは片方向の矢印で表現されますが、ここでは少し工夫して、人物の敵対関係を双方向の矢印で、また一方的関係を片方向の矢印で示してみました。

もう少し複雑な例を見てみましょう。図 1.2 は Graphviz の公式サイトにあげられているものです。

ここではノードの形が楕円から円に変わっています。二重の円になっているものもあります。またノードだけでなく、それぞれのエッジ（矢印）にもラベルが付いています。この図はコンピュータサイエンスの理論をグラフ化したものですが、テキストデータを用意して Graphviz のコマンドで処理すると、このようなグラフを瞬時に生成することができます。

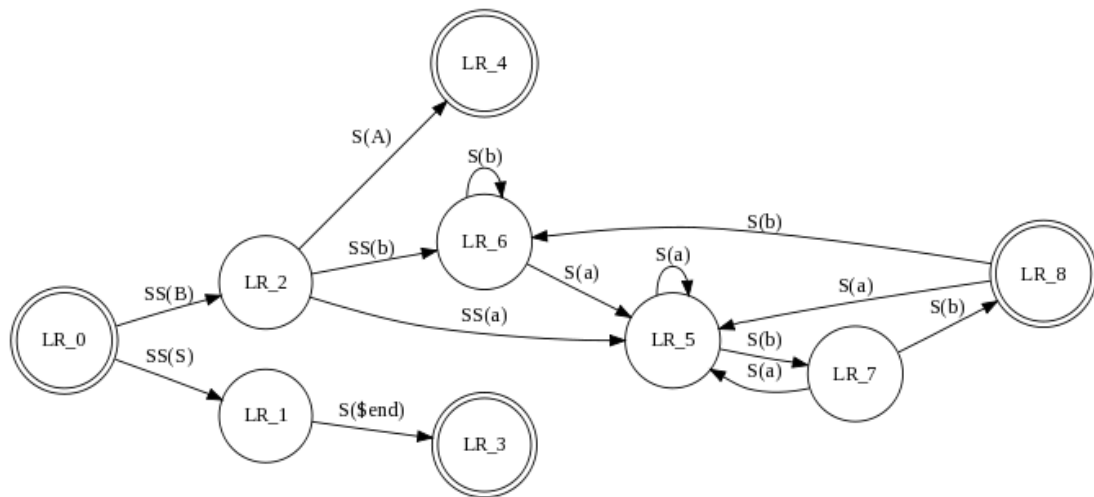


図 1.2 Finite Automaton (C) AT&amp;T

Graphviz のデータは「DOT 言語」という簡易なプログラムで記述します。

くわしくは下記のチュートリアルなどを見ていただくとして、一例だけあげておくと、DOT 言語の書式は次のようなものになります。

```

digraph {
  rankdir=LR;
  node [shape = doublecircle]; LR_0 LR_3 LR_4 LR_8;
  node [shape = circle];
  LR_0 -> LR_2 [ label = "SS(B)" ];
  LR_0 -> LR_1 [ label = "SS(S)" ];
  LR_1 -> LR_3 [ label = "S($end)" ];
  LR_2 -> LR_6 [ label = "SS(b)" ];
  LR_2 -> LR_5 [ label = "SS(a)" ];
  LR_2 -> LR_4 [ label = "S(A)" ];
  LR_5 -> LR_7 [ label = "S(b)" ];
  LR_5 -> LR_5 [ label = "S(a)" ];
  LR_6 -> LR_6 [ label = "S(b)" ];
  LR_6 -> LR_5 [ label = "S(a)" ];
  LR_7 -> LR_8 [ label = "S(b)" ];
  LR_7 -> LR_5 [ label = "S(a)" ];
  LR_8 -> LR_6 [ label = "S(b)" ];
  LR_8 -> LR_5 [ label = "S(a)" ];
}

```

このプログラムをたとえば fsm.dot という名前でセーブし、端末ウィンドウから次のように指示して得られたのが、上の 図 1.2 のグラフです。

```
dot -Tpng fsm.dot -o fsm.png
```

Graphviz では、このように簡単なプログラムで体裁の整ったグラフを作ることができます。ここで使った **dot** コマンドは、Graphviz が提供しているコマンドのうちの代表的なものです。ほかにもコマンドが用意されていて、それぞれ特色の異なるグラフが生成されます。

## Graphviz の参考資料

Graphviz はアメリカの AT&T 研究所で開発されたものですが、最近は日本でも利用が増え、日本語の情報も充実しつつあります。

まずこのあたりからご覧になると、入手方法や使い方を知ることができます。

- [Graphviz チュートリアル](#)
- [Graphviz による有向グラフの自動生成](#)
- [dot を使ったグラフ描画 \(PDF\)](#)  
完全なマニュアルは公式サイトにあります。
- [Graphviz - Graph Visualization Software](#)

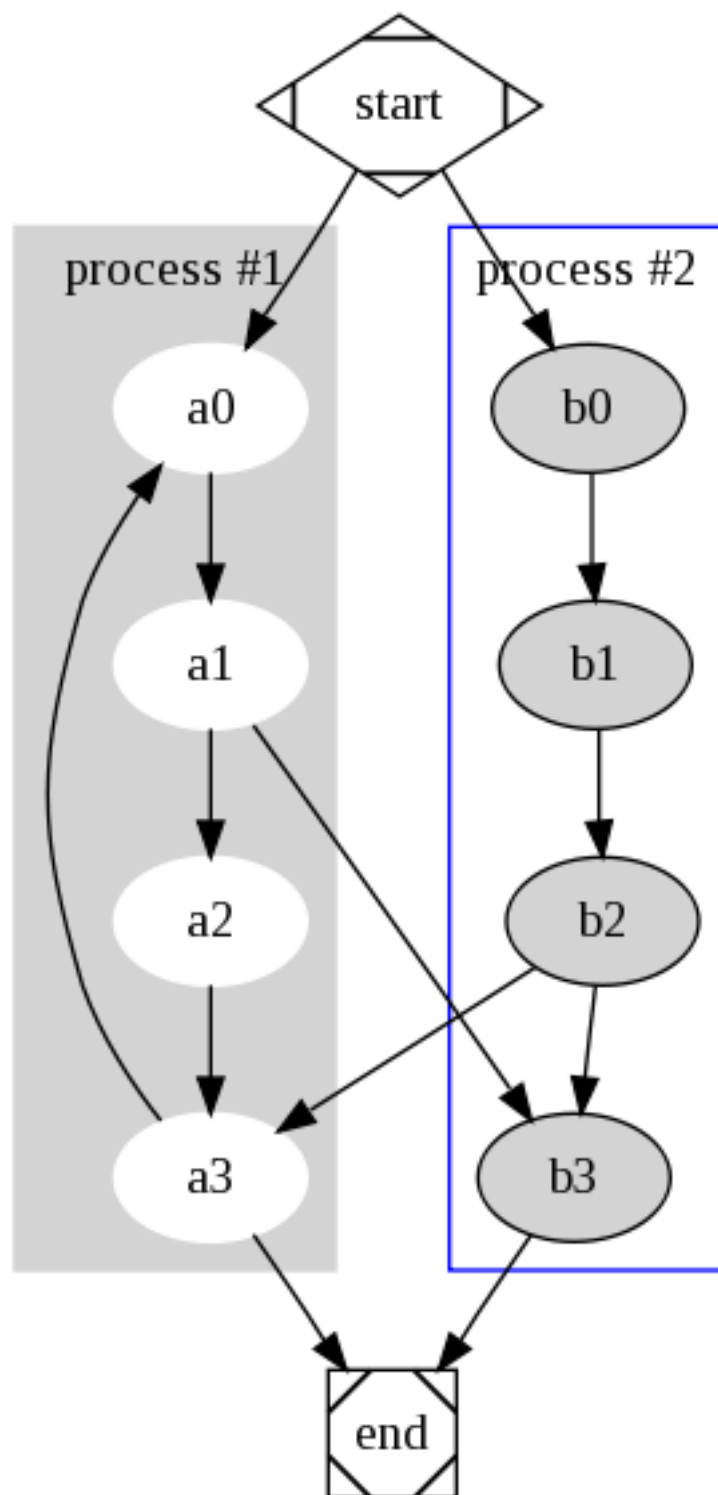
## Graphviz の高度な機能

Graphviz のパワーと表現力のうかがえる例を二、三あげておきます。

次のグラフはフローチャート風の模擬図です。

かなり手間のかかった図のように感じられますが、公式サイトにある [プログラム](#) はシンプルなものです。

Graphviz の基本をざっと学ぶだけでこの程度のグラフは容易に描けるようになります。



これまでに見た例は、ノードの数にして 10 件前後のコンパクトなグラフでしたが、Graphviz の

扱える件数に制限はありません。

次の図は Twitter の投稿から抽出した交友関係（言及関係）をグラフにしたものです。左はカナダの人気歌手ジャスティン・ビーバーをめぐる言及関係、右はアメリカの政治運動ティーパーティーを巡る言及関係を示していますが、それぞれ 3000 件のデータを処理しています。

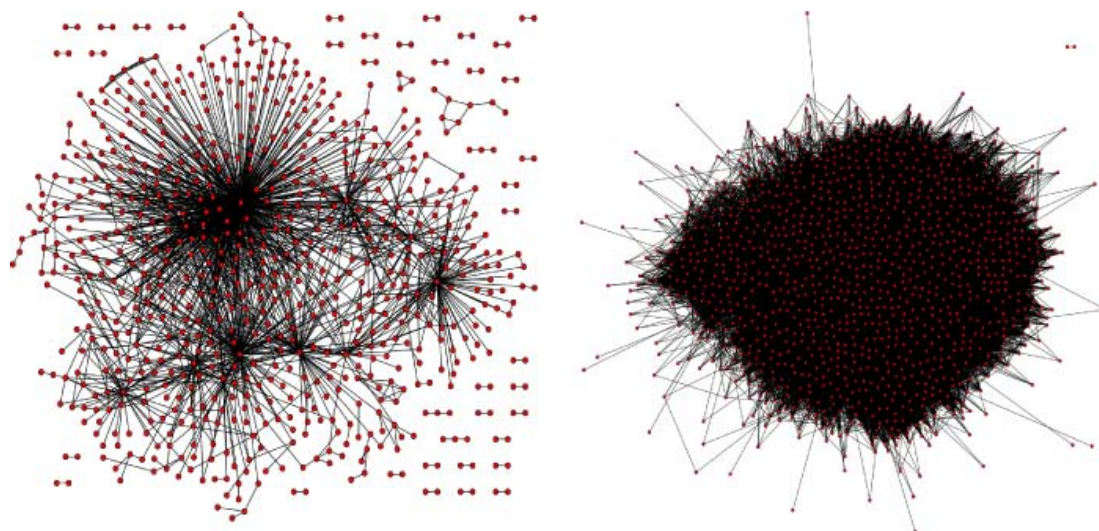


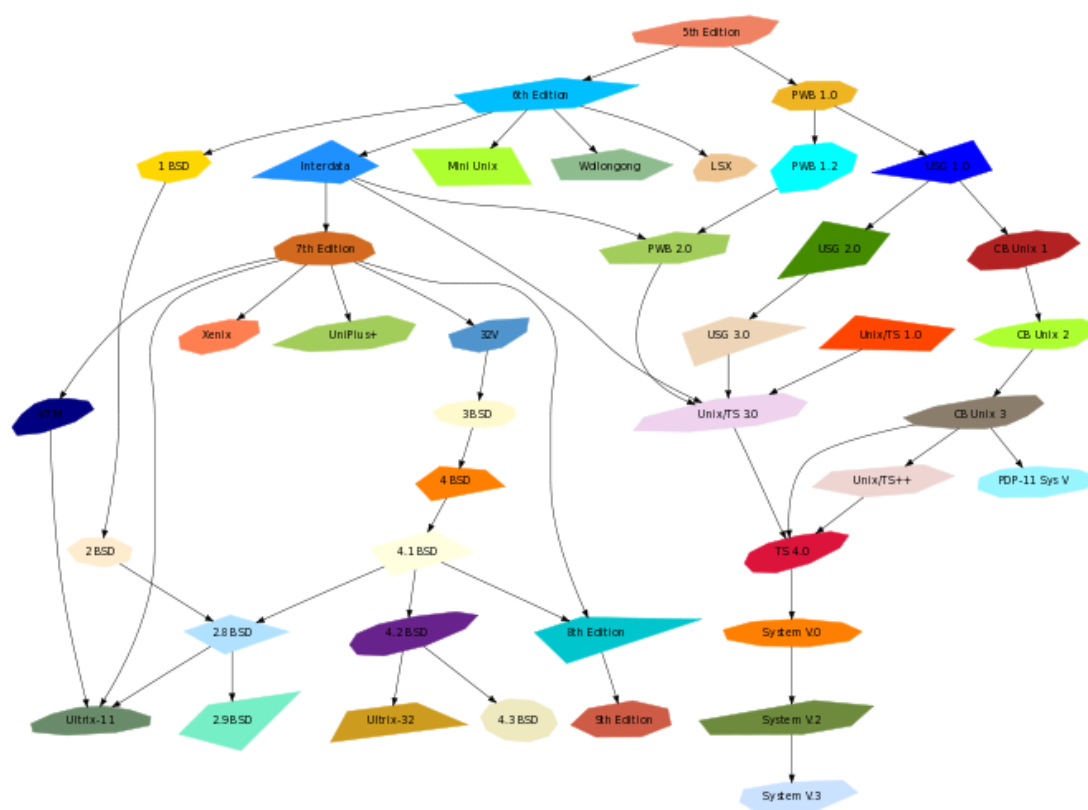
図 1.4 Twitter のツイートをもとに作成された人間関係 (C) "Mining the Social Web", O'Reilly Media

もう一つ紹介しておきます。

次図は UNIX オペレーティングシステムの発展を示す系統樹ですが、それぞれのノードを別の色、別の形に描きわけて美しく仕上げています。かなり面倒な指定を重ねてあるかのようにですが、**プログラムの説明**によると、ノードの色も形もランダムに発生させた値が割り当てられているとのこと

です。

手法はともかく、Graphviz の高い表現力を示す例となっています。



☒ 1.5 Object Oriented Graphs (C) Stephen North



## 第 2 章

# graph-easy

graph-easy はアスキーアートのグラフを簡単に生成できるツールです。  
Perl のインストールされた環境で動きますが、Perl プログラミングの知識は必要ありません。  
使い方も簡単です。たとえば次のようなテキストを用意し、

リスト 2.1: travel.txt

```
[ Bonn ] -- car --> [ Berlin ], [ Ulm ]
```

これを保存して、次のコマンドで処理すると、

```
graph-easy travel.txt --output=travel.ascii
```

次の内容のテキストファイル travel.ascii が生成されます。

リスト 2.2: travel.ascii

```
+-----+   car   +-----+
|  Bonn  | -----> | Ulm  |
+-----+         +-----+
      |
      | car
      v
+-----+
| Berlin |
+-----+
```

リスト 2.1 のような簡単なテキストを リスト 2.2 のように展開してくれるのが graph-easy というツールです。  
後述のように、このアスキーアートをさらにグラフィカルな画像に変換することもできます。

## graph-easy の参考資料

graph-easy のチュートリアル資料はあまりないのですが、このあたりを参考にすればすぐ使えるようになるでしょう。

- [本を読む graph-easy でテキストのハコ図をらくらく生成](#)

こちらは後述の ditaa というツールとあわせた記事です。

- [ditaa で簡単に作図する方法 « Stop Making Sense](#)

公式サイトマニュアルはこちらです。

- [Graph::Easy - search.cpan.org](#)

## 他ツールとの連携

graph-easy はアスキーアートだけでなく、HTML、PNG、GIF、SVG、DOT、その他の形式のファイルを出力します。どの出力が可能かは環境によりますが、たとえば先のコマンドの出力先を

```
graph-easy travel.txt --output=travel.dot
```

に変えると、次の内容のテキストファイル travel.dot が出力されます。

リスト 2.3: travel.dot

```
digraph GRAPH_0 {  
  
    // Generated by Graph::Easy 0.71 at Sun Apr 21 16:59:12 2013  
  
    edge [ arrowhead=open ];  
    graph [ rankdir=LR ];  
    node [  
        fontsize=11,  
        fillcolor=white,  
        style=filled,  
        shape=box ];  
  
    Bonn -> Ulm [ color="#000000", fontcolor="#000000", label=car ]  
    Bonn -> Berlin [ color="#000000", fontcolor="#000000", label=car ]  
}
```

これは前項で見た ?? の dot コマンドで処理できる形式のプログラムです。そこでコマンドラインから、

```
$ dot -Tpng travel.dot -o travel.png
```

と指示すると、次の画像 travel.png が生成されます。

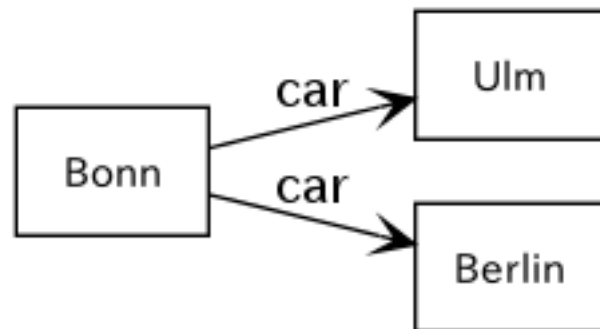


図 2.1 graph-easy の出力を Graphviz の dot コマンドで画像化

また、ditaa というツールを使うと次の画像ができます。

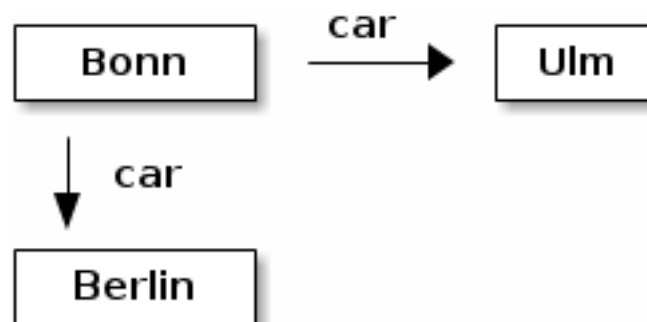


図 2.2 graph-easy の出力を ditaa で画像化

ditaa は Java プログラムの動く環境で使えます。  
公式サイトはこちらです。

- [ditaa](#)

## 第 3 章

# Google Chart Tools

Google Chart Tools は Google がオンラインで提供しているグラフ描画サービスです。JavaScript の基本的な知識が必要ですが、環境としてはウェブブラウザさえあればいいので、手軽に使うことができます。

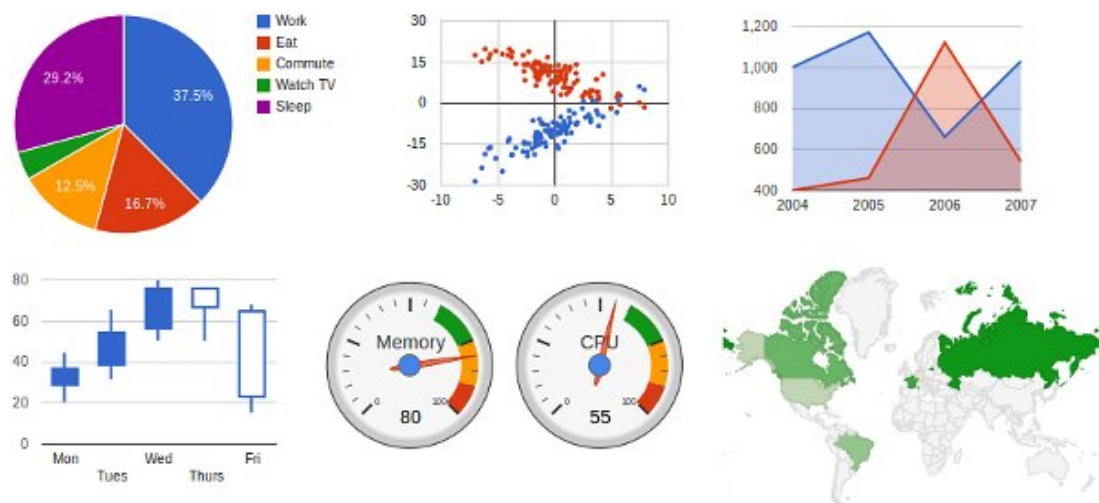


図 3.1 Google Chart Tools サンプル

公式サイトサンプルギャラリーはこちらです。

[Charts Gallery - Google Chart Tools — Google Developers](#)

これらの基本形から、色、サイズ、データなどを入れ替えて多様なグラフが作れます。

### Google Chart Tools の参考資料

こちらのチュートリアルビデオがわかりやすくできています。見るだけで Chart Tools の概要をつかむことができます。

- [Google Chart Tools 入門 - プログラミングならドットインストール](#)

次のサイトに充実した解説記事があります（現在 13 本）。

これだけ読めば Google Chart Tools が使いこなせるようになる内容です。

### - Google Chart Tools | プログラマーズ雑記帳

公式サイトはこちらです。

### Google Chart Tools — Google Developers

先に紹介した ?? や ?? と異なり、Google Chart Tools は市販の参考書も豊富です。ただし、Chart Tools に絞ったものは少なく、おおかたは Google の各種サービスを紹介する中で Chart Tools に触れています。

## インタラクティブなグラフ

Google Chart Tools はウェブページでの利用が本来の目的です。

次の円グラフの上にマウスカーソルを重ねてみましょう。グラフの形が一部変わって、パーセンテージのほかにも情報が表示されます。

ウェブで利用すると、このようなインタラクティブな表現が可能になります。また、時々刻々変化するデータをリアルタイムで表示することなどもできます。

印刷などの目的で静止画として使うと、Google Chart Tools の表現力の一部は失われますが、手軽に使えるメリットは大きく、知っていて損のないツールです。

同じく Google が提供しているスプレッドシートとの連携も可能です。

## 第 4 章

### D3.js

D3.js はきわめて表現力の高いグラフ描画ツールです。  
 ?? と同様、JavaScript ライブラリーとして提供されていますが、Chart Tools がオンライン限定のサービスであるのに対し、D3.js はオフラインでも利用できます。

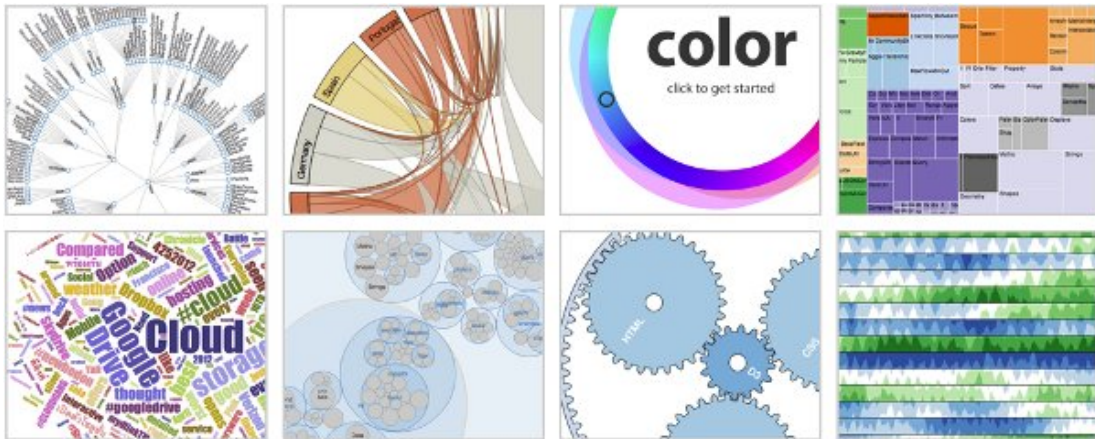


図 4.1 D3.js サンプル

公式サイトギャラリーにたくさんの例があげられています。

- [Gallery](#) ・ [mbostock/d3 Wiki](#) ・ [GitHub](#)

#### D3.js の参考資料

日本語で読める D3.js の資料はまだ多くありませんが、公式文書の日本語化が進んでいます。

- [D3.js - 日本語ドキュメント](#)

D3.js の基本をシンプルにまとめたチュートリアルです。

- [D3 チュートリアル | スコット・マレイ | alignedleft](#)

本領はウェブですが

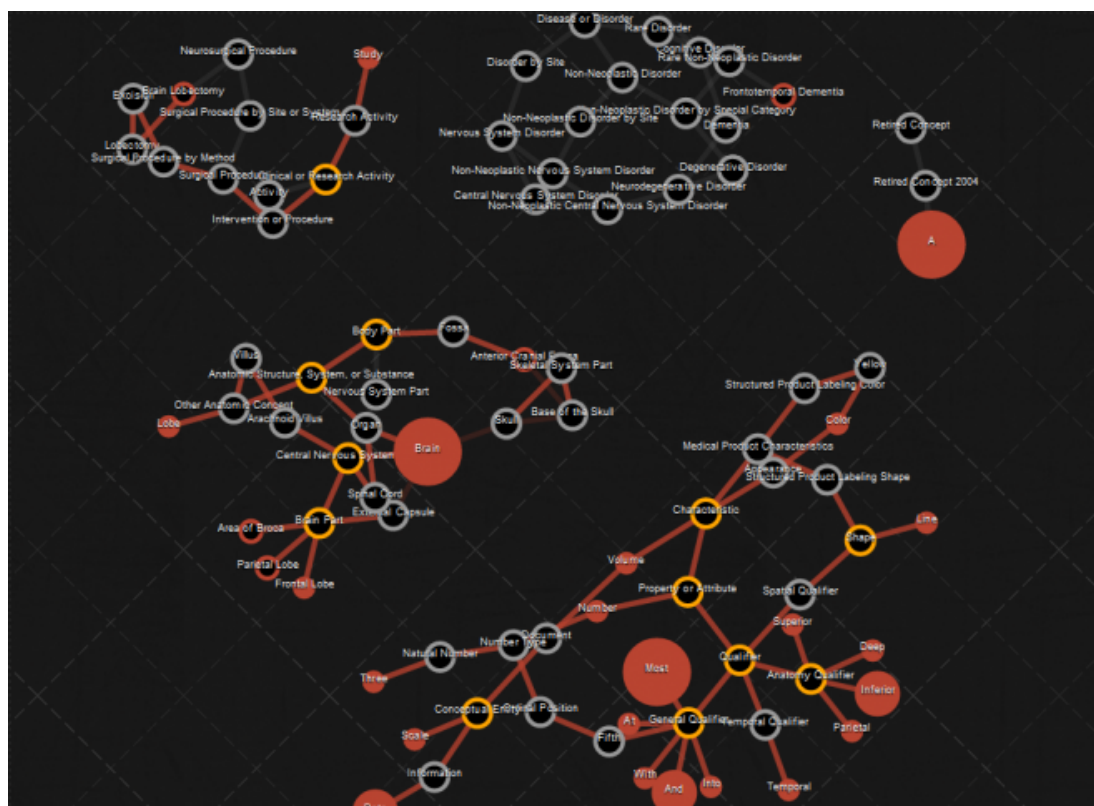


図 4.2 Force Directed Graph

D3.js は、たんに見栄えのするグラフを描くだけでなく、ウェブ上でインタラクティブなグラフやリアルタイムの情報のグラフ化を可能にするツールです。

たとえば上の画像はそれ自体センスよくまとまっていますが、次のリンク先では同じグラフがアニメーションになっています。マウスの操作に追従していろいろ変化するので試してみましょう (canvas 要素対応のブラウザが必要です)。

- [D3 DiGraph Test](#)

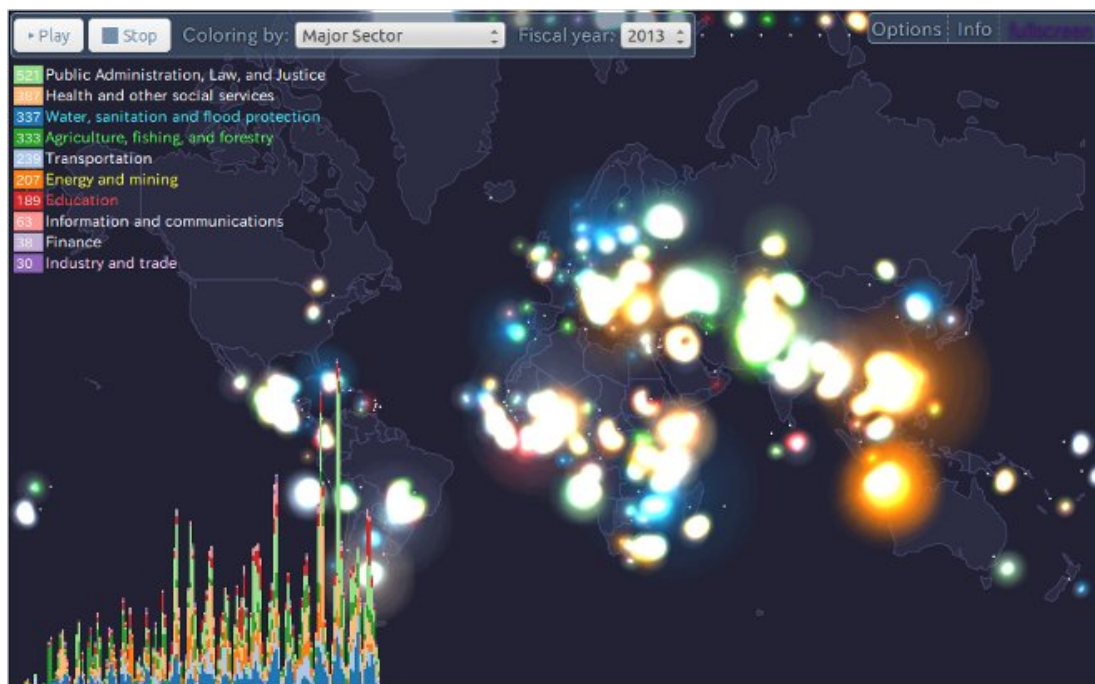


図 4.3 World Bank Global Development Sprint

この図 4.3 もアニメーションのスクリーンショットです。

下記サイトで実物が見られます。膨大なデータを読み込んでいるらしく、アニメーションの開始まで多少時間がかかります。

- [World Bank Global Development Sprint | visualizing.org](http://World Bank Global Development Sprint | visualizing.org)

このように D3.js はウェブで本領を発揮するツールですが、印刷素材としても D3.js の生成する画像は捨てがたいものがあります。

D3.js を使いこなすには、HTML、CSS、JavaScript、SVG などの基本知識が必要で、入門のハードルは低くはありません。しかし、統計類や調査・研究資料などを扱う仕事では強力な武器となるツールです。



## 第 5 章

### R (R 言語)

グラフ制作ツールの多くは、数学、物理、工学、統計など研究・調査のツールとして開発され各方面で早くから使われてきました。

それらのうち、ここでは R (R 言語) を紹介します。これまでに紹介したツールと比較して、R には (1) 対話的に使うツールである、(2) プログラムやデータのライブラリが豊富、(3) 参考書・参考資料も多い、などの特色があります。

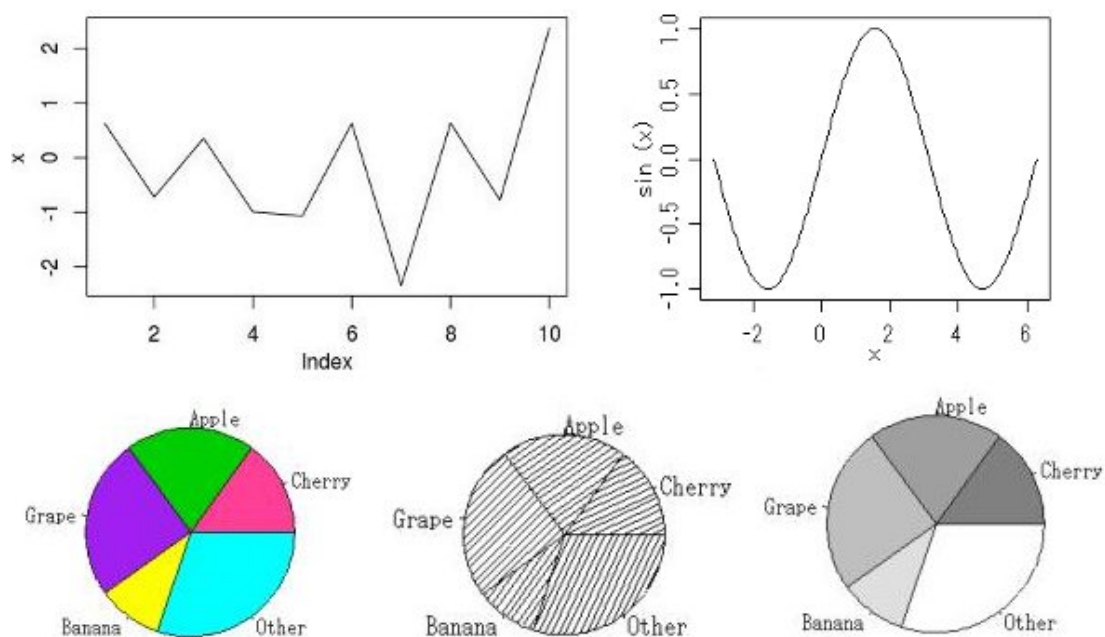


図 5.1 R の基本的なグラフ例

R は対話的なツールです。R を起動すると「>」の形のプロンプトが出てユーザーに入力を促すので、それにしたがって 1 行ずつ対話的に指示を出して行きます。たとえば次の 2 行の指示から生成されたのが 図 5.1 の折れ線グラフです。

```
> x <- rnorm(10)
> plot(x, type="l")
```

## 豊富な関連リソース

折れ線グラフ、棒グラフ、円グラフなど、シンプルなグラフが R の基本で、グラフィカルな美しさという点では ?? などに及びません。

しかし、サポートされている機能と表現の幅は広く、R のリソースライブラリである **CRAN** (Perl の CPAN に当たるものです) には 4500 件を超えるプログラムが登録されています。

レッスンに使えるデータセットも豊富に用意されています。レッスン用といっても架空のデータではなく、たとえば 図 5.2 はタイタニック号の乗客の性別・船室等級別のデータから生成されたグラフ、図 5.3 はニュージーランドの火山 Maunga Whau の地形情報から生成されたグラフで、これらのデータは標準配布のセットに含まれていたり、CRAN から入手できます。

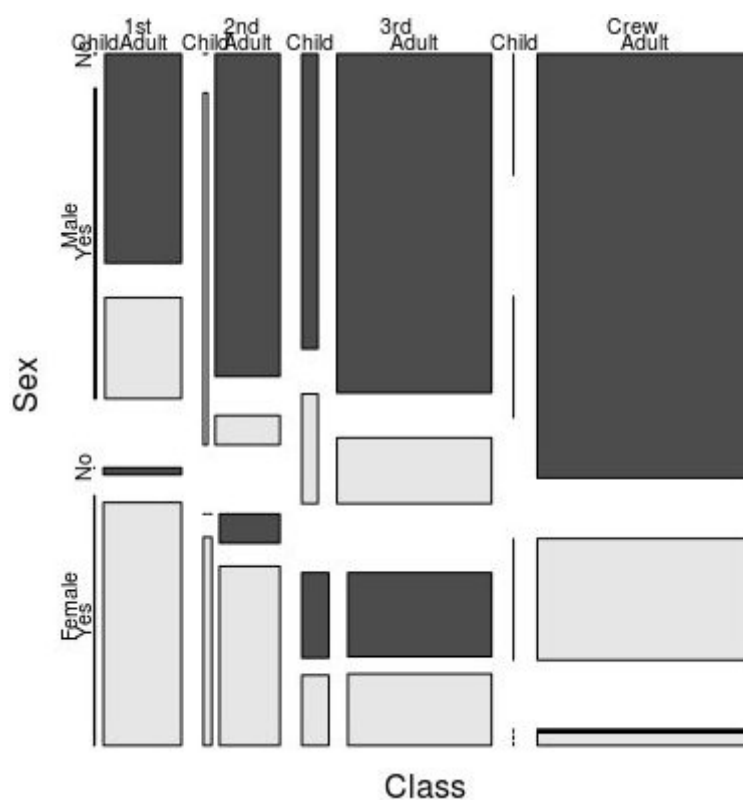


図 5.2 タイタニック号の乗員・乗客構成

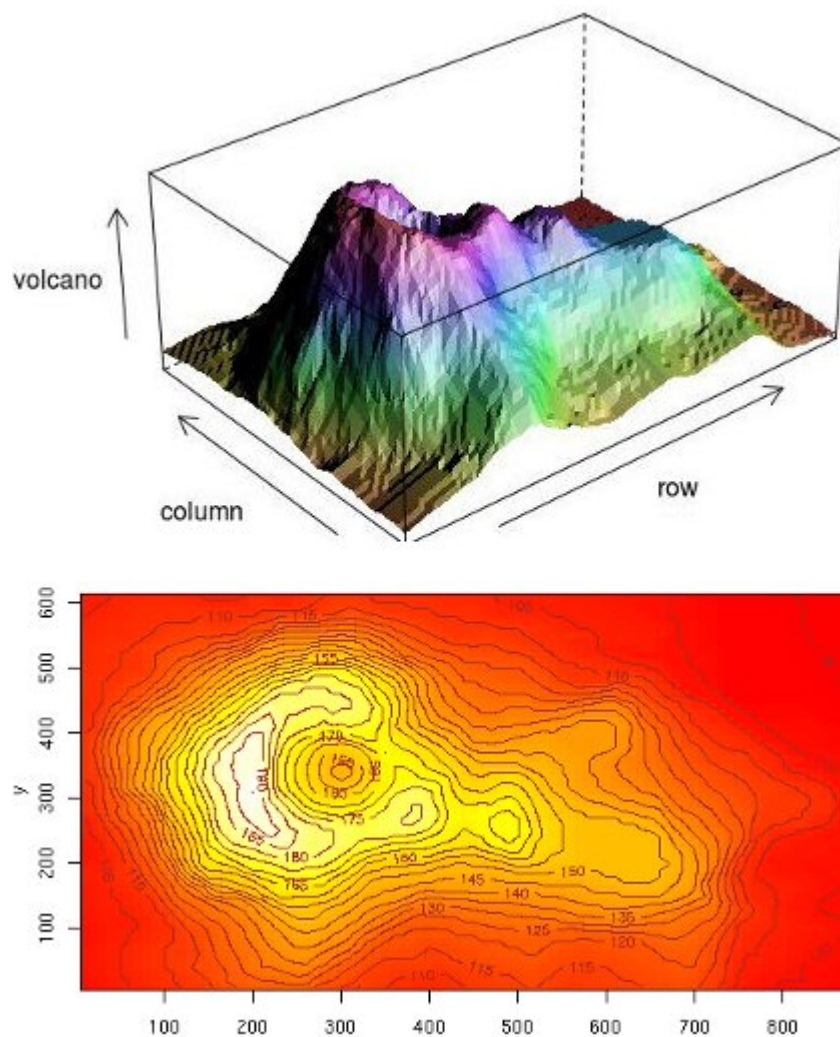


図 5.3 Maunga Whau 火山の地形

R のリソースは CRAN だけでなく、研究者や機関の手で大量のものが各方面に蓄積されています。次の日本地図は群馬大学の[サイト](#)にあるデータとプログラムから作られたものです。都道府県や市町村の境界データが公開されており、これをもとに白地図や塗り分けた地図を作ることができます。R のリソースは多岐にわたり、社会、経済などから、自然言語処理、分子生物学まで、さまざまな分野の小プログラムやデータセットが入手可能です。

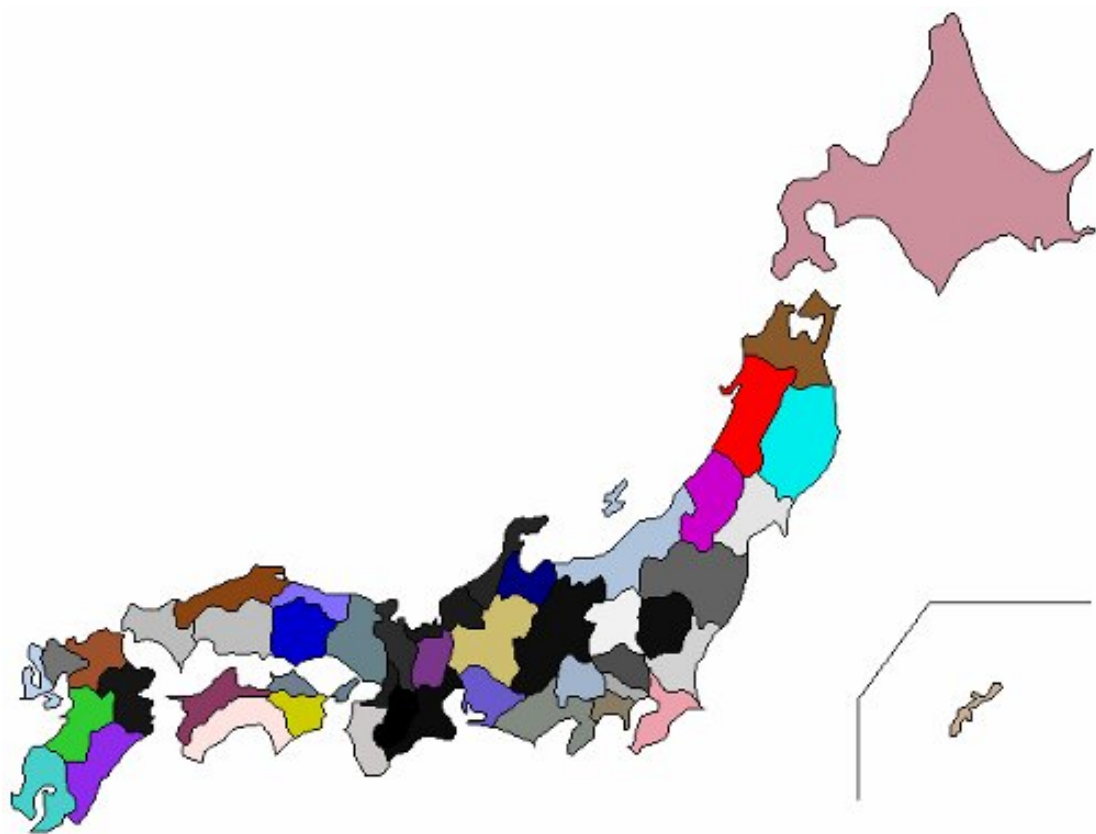


図 5.4 日本地図

## R の参考資料

参考資料も質量ともに豊富です。

ネット上では次の2つのチュートリアルをおさえておけば十分でしょう。入門から高度な使いこなしまでカバーしています。

- [R-Tips](#)
- [JIN'S PAGE](#)

R は参考書籍も早くから出ていて、最近も新刊の発行が続いています。大型書店でないと実物を目にできないのが難点ですが、オンライン書店などで検索すると多数ヒットします。

---

著 者 株式会社ショーワ  
発行所

---