



GSLetterNeo Vol.113

2017年12月



数値データで地図を作成する

松原 伸人

matubara@sra.co.jp

◆ 時系列データの変化を見る

時系列データの変化を見る場合、データ系列の数値に基づいて表示位置が決まるようにして表示位置の変化を見る方法と、データ系列の表示位置を固定し数値の大きさの変化を見る方法が挙げられます。

縦軸と横軸に異なる2種類のデータの軸を設定する散布図のようなものは、表示位置が変化する方法です。

横軸にデータ系列を並べて縦軸にデータの大きさをとる棒グラフや折れ線グラフは、表示位置を固定した方法です。

散布図のような位置が変化するダイアグラムは、項目がどのあたりに集まっているかが見やすいと思います。

要素の位置が固定のダイアグラムは、個々の項目の変化を見比べやすいと思います。

散布図の場合、同じ位置にある項目が重なって分からなくなる、棒グラフや折れ線グラフの場合、項目数が多いと全体を表示しきれなくなったり個々の項目が小さくて見えなくなったりする、という問題があります。

◆ Treemap

Treemap は数値または配列を要素とする配列の、各要素の数値または配列の合計値を面積比として、2次元平面を配列の要素群で矩形領域に分割して表示方法です。

配列は数値のグループとして表され、要素ごとの数値の大きさと、グループごとの数値の大きさを比べられるというものです。

「Vol.107 大量データの関係性の変化をインタラクティブに見るツールの紹介」で紹介したプロトタイプシステムは、Treemap を用いてデータを2次元平面上に配置していました。

記事の中では、2008年8月から2017年5月までに発行されたGSLetterNeoの記事の中に使われている名詞をデータ項目とし、名詞が最初に登場した年月(掲載号)をグループとして、掲載号順の配列を作成したデータを用いて、Treemapにより名詞の地図を作成しました。

この時は、全データ項目の数値は全て1としました。各号の面積は新たに登場した名詞の数を表します。

◆ 例: CodeDepot

例として2017年12月1日にオープンソースで公開されたCodeDepotのコードを見てみます。

CodeDepot はコード検索システムで、プログラムを解析して、関数の位置や呼び出し、コードの構造の類似などによる検索がWebブラウザ上で行えるソフトウェアです。

GitHub からダウンロードできます。

[CodedepotOSS/CodeDepot](https://github.com/CodeDepotOSS/CodeDepot)

CodeDepot をインストールして、CodeDepot プロジェクトを登録してみました。

CodeDepot は、プロジェクトごとにプログラム群を登録

し解析するようになっていて、各プロジェクトの中で使われている言語ごとのファイル数と行数を算出してくれます。[図 1]



図 1 CodeDepot に CodeDepot を登録した結果

言語ごとのファイル数のデータから TreemapComponentCanvas を用いて地図を作成してみます。

◆地図の作成

TreemapComponentCanvas は、Treemap を用い、JavaScript の Object を要素とする配列から地図を作成するプロトタイプです。

Treemap のプログラムは、imranghory/treemap-squared を使用しています。

value 属性に数値、values 属性に数値または配列の

配列を記述し、ファイル数の昇順にソートして treemap を作成するデータは次のように書きます。

```
let tree = [
  {
    language: 'text',
    value: 156
  }, {
    language: 'sql',
    value: 1
  }, {
    language: 'jsp',
    value: 39
  }, {
    language: 'java',
    value: 247
  }, {
    language: 'C',
    value: 1
  }
].sort(function (i1, i2) {
  return i1.value - i2.value;
});
```

TreemapComponentCanvas は HTML の Canvas を用いて 2 次元描画で表示を実現しています。

これを実行するサンプルコードが treemap-codedepot-files.html です。

実行すると次のスクリーンショットのようになります。[図 2]

ただし、描画領域の縦横幅の比率やデータの順序により描画結果は変わります。

上記 URL には、次の説明のようなプログラムが書いてあります。

Treemap の個々の要素の描画方法をデータに合わ

せて変えられるようにするために、`TreemapItemDrawer` を作ってあります。

`TreemapItemDrawer` は、`Treemap` によって得られた矩形エリアを画面上に描画するプログラムです。

描画方法は `TreemapItemDrawer` を継承して `drawItemComponentOn` 関数を上書き実装します。

`drawItemComponentOn` は、数値データを持つ要素を描画する際に呼ばれる関数です。関数が呼ばれる際に `Canvas` の `Graphic` オブジェクトと要素オブジェクトを受け取ります。

要素オブジェクトには、数値を表す `value` 属性と、数値に対応する `treemap` で計算した矩形領域が、`rect` 属性に入っています。

`drawValuesComponentOn` は、数値または配列からなる配列データを持つ要素を描画する際に呼ばれる関数です。関数が呼ばれる際に `Canvas` の `Graphic` オブジェクトと要素オブジェクトを受け取ります。

この要素オブジェクトには、数値または配列の配列を表す `values` 属性と、この配列の合計値に対応する `treemap` で計算した矩形領域が、`rect` 属性に入っています。

1 個目の例の中では、`Drawer` という名前で `TreemapItemDrawer` を継承したクラスを作り、`TreemapItemDrawer.prototype.drawItemComponentOn` だけ実装しています。

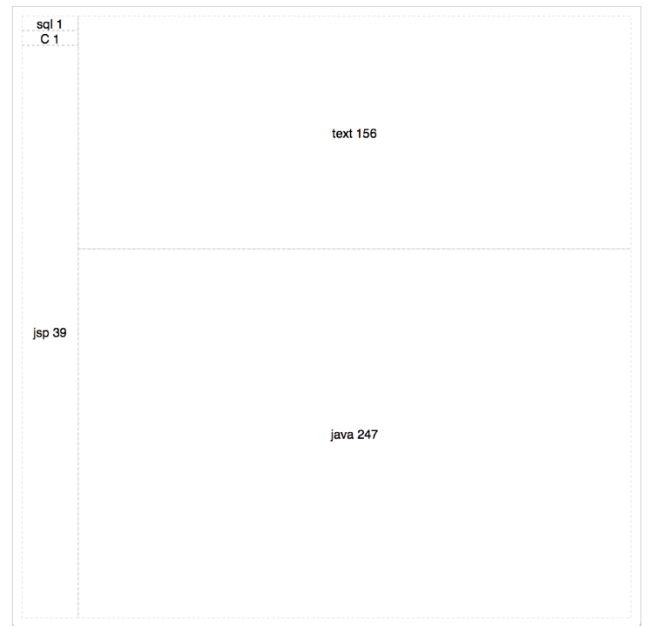


図 2 言語ごとのファイル数による `treemap` のスクリーンショット

次の例は、ファイル拡張子ごとにファイルのリストを作り、ファイル毎に行数を数えて `csv` ファイルに記録し、作成した `csv` ファイルを読んで `treemap` で表示します。

次のようなコマンドで、ファイル拡張子の一覧を得られました。

```
% find . -type f | perl -ne 'print $1 if m/\.[^\.\V]+)$/'|sort -u
```

```
c, css, html, java, js, jsp, md, properties, sh, sql, txt, xml
```

拡張子ごとにファイル行数を数えて `csv` ファイルに記録します。例えば `xml` の場合、次のようなコマンドで出来ました。

```
% find . -name "*.xml" -exec wc {} ";" | sort -k1r | awk '\nBEGIN {print "path,lines.words,chars"}\n{print $4 " ", $1 " ", $2 " ", $3}' > xml.csv
```

拡張子ごとにファイル行数を集計した 12 個の csv ファイルを読み込んで Treemap で地図を作成します。

treemap-codedepot-lines.html

図 3 はサンプルプログラムを実行したスクリーンショットです。

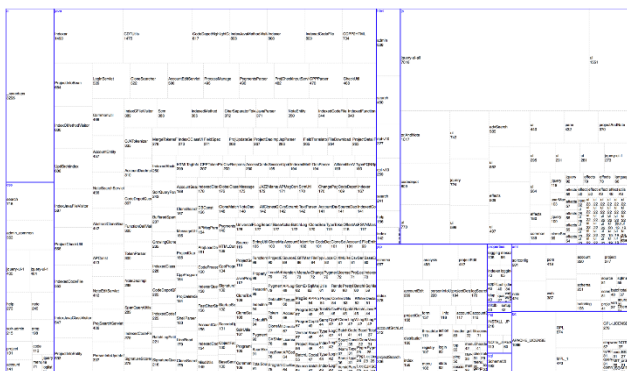


図 3 ファイルごとの行数による treemap のスクリーンショット

このサンプルプログラムでは、拡張子ごとにファイルを

グループ分けしているので、drawValuesComponent On を実装しています。

青い矩形がグループを表していて、この例だと拡張子ごとの行数の合計を面積で表しています。

項目を表す方法を工夫しないと、ファイル名が長すぎて表示できないような問題はありますが、どのファイルの種類がどれくらいの行数あるか、全体を見やすいと思います。

TreemapComponent は、ズーム機能を試作しています。

マウスをズームしたい位置に合わせて上にクリックする(マウスホイールを上を回す)と、マウスのある位置のグループを画面全体に表示するように拡大します。さらにマウスを合わせて上にクリックするとグループ内のマウスがある位置の要素を大きく表示します。

下にクリックするとズームアウトします。

GSLetterNeo Vol.113

2017年12月20日発行

発行者●株式会社 SRA 先端技術研究所

編集者●土屋正人

夢を。



株式会社SRA

〒171-8513 東京都豊島区南池袋 2-32-8

夢を。Yawaraka Innovation
やわらかいのべーしょん