Google Motion Charts with R:
Package Vignette

Markus Gesmann
markus.gesmann@gmail.com

October 3, 2010

Abstract

The `GoogleMotionChart` package provides an interface between R and the Google Visualisation API. The purpose of the package is to allow the user to transform data stored in a R `data.frame` structure into a Google Motion Chart. The output is html code, which when run on a web server, generates a dynamic flash based bubble chart to explore several indicators over time.

1 Introduction

The standard motion chart is essentially an animated bubble chart, which allows the user to explore several indicators over time. The idea of motion chart was popularised by Hans Rosling at a Ted talk [TED06] about social and economic developments in the world, see also [Gap10].

2 Google Motion Chart API

The Google Motion Chart is part of Google Visualisation API [Goo10a]. Charts are rendered within a browser using Adobe Flash. The charting data can either be embedded into the html file or read dynamically. Key to Google Visualisation API is that the data is structured in a DataTable [Goo10c], and this is where this package helps, as it uses the functionality of the RJSONIO package [Lan10] and wraps it into a Google Motion chart. The data has to have at least four columns with subject name (`idvar`), time (`timevar`) and two columns of numeric values. Further columns are optional. The Google Public Data Explorer [Goo10b] shows several examples of motion charts and how they can help to analyse data. Figure 1 shows the graphical user interface of a motion chart.
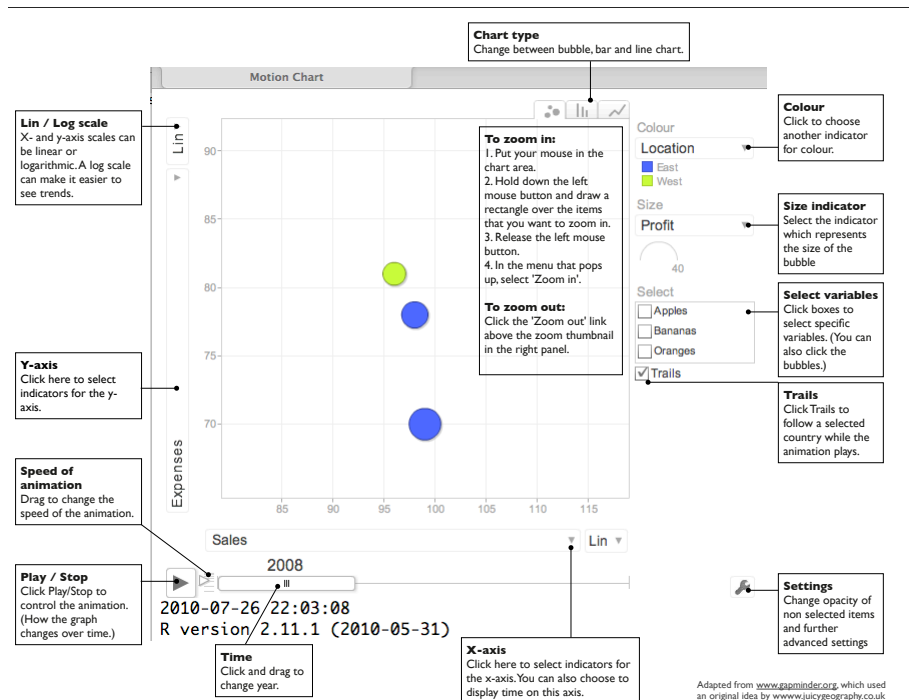
Figure 1: Overview of a Google motion chart. Screenshot of the output of `MotionChartPage(Fruits, idvar='Fruit', timevar='Year', file='FruitAnalysis.rsp')`

3 Example

The package comes with an example data set called `Fruits` and `Exports`.

```
R> require(GoogleMotionChart)
R> data(Fruits)
R> Fruits
```

	Fruit	Year	Location	Sales	Expenses	Profit	Date
1	Apples	2008	West	98	78	20	2008-12-31
2	Oranges	2008	East	96	81	15	2008-12-31
3	Bananas	2008	West	99	70	29	2008-12-31
4	Apples	2008	East	106	84	22	2008-12-31
5	Oranges	2008	West	117	85	32	2008-12-31
6	Bananas	2008	East	85	76	9	2008-12-31
7	Apples	2009	West	111	79	32	2009-12-31
8	Oranges	2009	East	93	80	13	2009-12-31

```
9  Bananas 2009     West    101       69      32 2009-12-31
10   Apples 2009     East    105       66      39 2009-12-31
11 Oranges 2009     West    113       80      33 2009-12-31
12 Bananas 2009     East     94       78      16 2009-12-31
13   Apples 2010     West     89       76      13 2010-12-31
14 Oranges 2010     East     98       91       7 2010-12-31
15 Bananas 2010     West     97       87      10 2010-12-31
16   Apples 2010     East    111       72      39 2010-12-31
17 Oranges 2010     West    103       79      24 2010-12-31
18 Bananas 2010     East     81       71      10 2010-12-31
```

The example data shows sales information by Fruit and Year (also Date to illustrate how to handle this data type). Suppose we would like to generate the html output required for a motion chart, than we can use the function `MotionChart`.

R> MotionChart(Fruits[,-7], "Fruit", "Year") ## exclude Date column

```
<script type="text/javascript" src="http://www.google.com/jsapi"></script>
<script type="text/javascript">
  google.load("visualization", "1", { packages:["motionchart"] });
  google.setOnLoadCallback(drawChart);
  function drawChart() {
    var data = new google.visualization.DataTable();
    data.addRows(18);
    data.addColumn('string', 'Fruit');
    data.addColumn('number', 'Year');
    data.addColumn('string', 'Location');
    data.addColumn('number', 'Sales');
    data.addColumn('number', 'Expenses');
    data.addColumn('number', 'Profit');
    data.setValue(0, 0, 'Apples');
    data.setValue(1, 0, 'Oranges');
    data.setValue(2, 0, 'Bananas');
    data.setValue(3, 0, 'Apples');
    data.setValue(4, 0, 'Oranges');
    data.setValue(5, 0, 'Bananas');
    data.setValue(6, 0, 'Apples');
    data.setValue(7, 0, 'Oranges');
    data.setValue(8, 0, 'Bananas');
    data.setValue(9, 0, 'Apples');
    data.setValue(10, 0, 'Oranges');
    data.setValue(11, 0, 'Bananas');
    data.setValue(12, 0, 'Apples');
    data.setValue(13, 0, 'Oranges');
    data.setValue(14, 0, 'Bananas');
    data.setValue(15, 0, 'Apples');
```

```
data.setValue(16, 0, 'Oranges');
data.setValue(17, 0, 'Bananas');
data.setValue(0, 1, 2008);
data.setValue(1, 1, 2008);
data.setValue(2, 1, 2008);
data.setValue(3, 1, 2008);
data.setValue(4, 1, 2008);
data.setValue(5, 1, 2008);
data.setValue(6, 1, 2009);
data.setValue(7, 1, 2009);
data.setValue(8, 1, 2009);
data.setValue(9, 1, 2009);
data.setValue(10, 1, 2009);
data.setValue(11, 1, 2009);
data.setValue(12, 1, 2010);
data.setValue(13, 1, 2010);
data.setValue(14, 1, 2010);
data.setValue(15, 1, 2010);
data.setValue(16, 1, 2010);
data.setValue(17, 1, 2010);
data.setValue(0, 2, 'West');
data.setValue(1, 2, 'East');
data.setValue(2, 2, 'West');
data.setValue(3, 2, 'East');
data.setValue(4, 2, 'West');
data.setValue(5, 2, 'East');
data.setValue(6, 2, 'West');
data.setValue(7, 2, 'East');
data.setValue(8, 2, 'West');
data.setValue(9, 2, 'East');
data.setValue(10, 2, 'West');
data.setValue(11, 2, 'East');
data.setValue(12, 2, 'West');
data.setValue(13, 2, 'East');
data.setValue(14, 2, 'West');
data.setValue(15, 2, 'East');
data.setValue(16, 2, 'West');
data.setValue(17, 2, 'East');
data.setValue(0, 3,  98);
data.setValue(1, 3,  96);
data.setValue(2, 3,  99);
data.setValue(3, 3, 106);
data.setValue(4, 3, 117);
data.setValue(5, 3,  85);
data.setValue(6, 3, 111);
data.setValue(7, 3,  93);
```

```
data.setValue(8, 3, 101);
data.setValue(9, 3, 105);
data.setValue(10, 3, 113);
data.setValue(11, 3,  94);
data.setValue(12, 3,  89);
data.setValue(13, 3,  98);
data.setValue(14, 3,  97);
data.setValue(15, 3, 111);
data.setValue(16, 3, 103);
data.setValue(17, 3,  81);
data.setValue(0, 4, 78);
data.setValue(1, 4, 81);
data.setValue(2, 4, 70);
data.setValue(3, 4, 84);
data.setValue(4, 4, 85);
data.setValue(5, 4, 76);
data.setValue(6, 4, 79);
data.setValue(7, 4, 80);
data.setValue(8, 4, 69);
data.setValue(9, 4, 66);
data.setValue(10, 4, 80);
data.setValue(11, 4, 78);
data.setValue(12, 4, 76);
data.setValue(13, 4, 91);
data.setValue(14, 4, 87);
data.setValue(15, 4, 72);
data.setValue(16, 4, 79);
data.setValue(17, 4, 71);
data.setValue(0, 5, 20);
data.setValue(1, 5, 15);
data.setValue(2, 5, 29);
data.setValue(3, 5, 22);
data.setValue(4, 5, 32);
data.setValue(5, 5,  9);
data.setValue(6, 5, 32);
data.setValue(7, 5, 13);
data.setValue(8, 5, 32);
data.setValue(9, 5, 39);
data.setValue(10, 5, 33);
data.setValue(11, 5, 16);
data.setValue(12, 5, 13);
data.setValue(13, 5,  7);
data.setValue(14, 5, 10);
data.setValue(15, 5, 39);
data.setValue(16, 5, 24);
data.setValue(17, 5, 10);
```

```
        var chart = new google.visualization.MotionChart(document.getElementById('chart_div'
        chart.draw(data, {width: 600, height: 500});", "         }
    </script>
    <div id="chart_div" style="width: 600px; height: 500px;"></div>
[1] ""
```

Now we can also create a motion chart using the date colum. Please notice that

```
R> MotionChart(Fruits[,-2], "Fruit", "Date") ## exclude Year column
```

```
    <script type="text/javascript" src="http://www.google.com/jsapi"></script>
    <script type="text/javascript">
      google.load("visualization", "1", { packages:["motionchart"] });
      google.setOnLoadCallback(drawChart);
      function drawChart() {
        var data = new google.visualization.DataTable();
        data.addRows(18);
        data.addColumn('string', 'Fruit');
        data.addColumn('date', 'Date');
        data.addColumn('string', 'Location');
        data.addColumn('number', 'Sales');
        data.addColumn('number', 'Expenses');
        data.addColumn('number', 'Profit');
        data.setValue(0, 0, 'Apples');
        data.setValue(1, 0, 'Oranges');
        data.setValue(2, 0, 'Bananas');
        data.setValue(3, 0, 'Apples');
        data.setValue(4, 0, 'Oranges');
        data.setValue(5, 0, 'Bananas');
        data.setValue(6, 0, 'Apples');
        data.setValue(7, 0, 'Oranges');
        data.setValue(8, 0, 'Bananas');
        data.setValue(9, 0, 'Apples');
        data.setValue(10, 0, 'Oranges');
        data.setValue(11, 0, 'Bananas');
        data.setValue(12, 0, 'Apples');
        data.setValue(13, 0, 'Oranges');
        data.setValue(14, 0, 'Bananas');
        data.setValue(15, 0, 'Apples');
        data.setValue(16, 0, 'Oranges');
        data.setValue(17, 0, 'Bananas');
        data.setValue(0, 1, new Date('2008/12/31'));
        data.setValue(1, 1, new Date('2008/12/31'));
        data.setValue(2, 1, new Date('2008/12/31'));
        data.setValue(3, 1, new Date('2008/12/31'));
        data.setValue(4, 1, new Date('2008/12/31'));
        data.setValue(5, 1, new Date('2008/12/31'));
```

```
data.setValue(6, 1, new Date('2009/12/31'));
data.setValue(7, 1, new Date('2009/12/31'));
data.setValue(8, 1, new Date('2009/12/31'));
data.setValue(9, 1, new Date('2009/12/31'));
data.setValue(10, 1, new Date('2009/12/31'));
data.setValue(11, 1, new Date('2009/12/31'));
data.setValue(12, 1, new Date('2010/12/31'));
data.setValue(13, 1, new Date('2010/12/31'));
data.setValue(14, 1, new Date('2010/12/31'));
data.setValue(15, 1, new Date('2010/12/31'));
data.setValue(16, 1, new Date('2010/12/31'));
data.setValue(17, 1, new Date('2010/12/31'));
data.setValue(0, 2, 'West');
data.setValue(1, 2, 'East');
data.setValue(2, 2, 'West');
data.setValue(3, 2, 'East');
data.setValue(4, 2, 'West');
data.setValue(5, 2, 'East');
data.setValue(6, 2, 'West');
data.setValue(7, 2, 'East');
data.setValue(8, 2, 'West');
data.setValue(9, 2, 'East');
data.setValue(10, 2, 'West');
data.setValue(11, 2, 'East');
data.setValue(12, 2, 'West');
data.setValue(13, 2, 'East');
data.setValue(14, 2, 'West');
data.setValue(15, 2, 'East');
data.setValue(16, 2, 'West');
data.setValue(17, 2, 'East');
data.setValue(0, 3,  98);
data.setValue(1, 3,  96);
data.setValue(2, 3,  99);
data.setValue(3, 3, 106);
data.setValue(4, 3, 117);
data.setValue(5, 3,  85);
data.setValue(6, 3, 111);
data.setValue(7, 3,  93);
data.setValue(8, 3, 101);
data.setValue(9, 3, 105);
data.setValue(10, 3, 113);
data.setValue(11, 3,  94);
data.setValue(12, 3,  89);
data.setValue(13, 3,  98);
data.setValue(14, 3,  97);
data.setValue(15, 3, 111);
```

```
        data.setValue(16, 3, 103);
        data.setValue(17, 3,  81);
        data.setValue(0, 4, 78);
        data.setValue(1, 4, 81);
        data.setValue(2, 4, 70);
        data.setValue(3, 4, 84);
        data.setValue(4, 4, 85);
        data.setValue(5, 4, 76);
        data.setValue(6, 4, 79);
        data.setValue(7, 4, 80);
        data.setValue(8, 4, 69);
        data.setValue(9, 4, 66);
        data.setValue(10, 4, 80);
        data.setValue(11, 4, 78);
        data.setValue(12, 4, 76);
        data.setValue(13, 4, 91);
        data.setValue(14, 4, 87);
        data.setValue(15, 4, 72);
        data.setValue(16, 4, 79);
        data.setValue(17, 4, 71);
        data.setValue(0, 5, 20);
        data.setValue(1, 5, 15);
        data.setValue(2, 5, 29);
        data.setValue(3, 5, 22);
        data.setValue(4, 5, 32);
        data.setValue(5, 5,  9);
        data.setValue(6, 5, 32);
        data.setValue(7, 5, 13);
        data.setValue(8, 5, 32);
        data.setValue(9, 5, 39);
        data.setValue(10, 5, 33);
        data.setValue(11, 5, 16);
        data.setValue(12, 5, 13);
        data.setValue(13, 5,  7);
        data.setValue(14, 5, 10);
        data.setValue(15, 5, 39);
        data.setValue(16, 5, 24);
        data.setValue(17, 5, 10);
        var chart = new google.visualization.MotionChart(document.getElementById('chart_div'
        chart.draw(data, {width: 600, height: 500});", "         }
    </script>
    <div id="chart_div" style="width: 600px; height: 500px;"></div>
[1] ""
```

The function `MotionChartPage` would top and tail the above output with some
html tags to create a valid standalone page to be viewed. The page can be viewed

immediately using the web server provided by the `R.rsp` package [Ben09].

Further examples are available via demos. See `demo('GoogleMotionChart')` for a list of demos files.

References

[Ben09]   Henrik Bengtsson. *R.rsp: R Server Pages*, 2009. R package version 0.3.6.

[Gap10]   Gapminder, http://www.gapminder.org. 2010.

[Goo10a] Google Inc., http://code.google.com/apis/visualization/documentation/gallery/motionchart.html. *Google Motion Chart API*, 2010.

[Goo10b] Google Inc., http://www.google.com/publicdata/home. *Google Public Data Explorer*, 2010.

[Goo10c] Google Inc, http://code.google.com/apis/visualization/documentation/reference.html. *Google Visualisation Reference*, 2010.

[Lan10]   Duncan Temple Lang. *RJSONIO: Serialize R objects to JSON, JavaScript Object Notation*, 2010. R package version 0.3-0.

[TED06]   TED Talk, http://www.ted.com/talks/hans_rosling_shows_the_best_stats_you_ve_ever_seen.html. *Hans Rosling shows the best stats you've ever seen*, 2006.