Table of Contents
Sample wiki page with R code and chart generated

```
1. require(gvisVis)
2. M <- gvisMotionChart(Fruits, "Fruit", "Year", options = list(width = 550,
3.   height = 450))
4. print(M, "chart")
```

Simple syntax highlighted & preview

Note: Remember that this is only a preview, and has not yet been saved!

1. Text output

This code:

```
(r()) 1:10(r)
```

Produces:

```
1 2 3 4 5 6 7 8 9 10
```
Escaping Wiki syntax

_Parsing Wiki Syntax_

Simple Interface: list runs/datasets

<table>
<thead>
<tr>
<th>Summary</th>
<th>Description</th>
<th>From user</th>
<th>Dataset file</th>
<th>Minimum value for axis X</th>
<th>Maximum value for axis X</th>
<th>LastModif</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample dataset</td>
<td>This dataset was created as part of the sample data for r_test.</td>
<td>admin</td>
<td>1</td>
<td>10</td>
<td>30</td>
<td>2013-08-30 17:37</td>
</tr>
<tr>
<td>We are working on this dataset</td>
<td>This will soon be changed</td>
<td>admin</td>
<td>21</td>
<td>30</td>
<td>100</td>
<td>2012-05-11 16:57</td>
</tr>
<tr>
<td>A really old dataset</td>
<td>This dataset is outdated.</td>
<td>admin</td>
<td>100</td>
<td>110</td>
<td>2012-05-11 16:57</td>
<td></td>
</tr>
</tbody>
</table>
Simple interface: Results for one run/dataset

Results

Values for X:
min: 1
max: 10

Those are the results:

Results from 1*10: 10
No attachment to display in this raw dataset

Graph with xmin 1 & xmax 10 and y=x^2

Simple templates for custom output
Flexible databases in Trackers to hold run parameters

Optional pop up helpers to edit plugin calls based on GUI

1. Basic image

R Code:

```r
x <- seq(1:10)
y <- x * x
plot(x, y)
```

RR (R syntax also)

- Same as PluginR, but allowing the execution of potentially dangerous commands once the admin has validated
- echo
- `if(require(Cairo)){
  install.packages("Cairo", repos="http://ftp.heanet.ie/mirrors/cran.r-project.org")
}
LoadAndSave
- Load a previous R user session (.RData, if any) for the same wiki page so that R object will be used while you
  (wiki pages with sameId), the R session data (.RData) will be shared for the same Id and
  ```
Custom maps with GoogleVis

```r
G5 <- gvisGeoMap(CiudadPopular, locationvar="Ciudad", numvar="Popular",
                  options=list(region="ES", height=350,
                                 dataMode="markers",
                                 colors=[0xFF8747, 0xFFB581, 0x060000]))

# plot(G5)
print(G5, "chart")
```

Spanish city popularity after UseR!2013 ;-)}
Embedded plot.ly charts

Fun with the Lognormal distribution

Embedded plot.ly charts: Heatmaps

x: Wednesday
y: Afternoon
z: 50
Custom output for higher control on figure results (pdf)

```r
device.height = convertHeight(sum(g[["heights"]]), "in", valueOnly=TRUE)
dev.pdf("test.str.pdf", height = device.height)
grid.draw(g)
invisible(dev.off())
```

Mobile display mode when needed

---

**rCharts**

*rCharts* is an R package to create, customize and publish interactive javascript visualizations from R using a familiar lattice style plotting interface. It has been created by [Ramanath Vaidyanathan](http://ramanathvaidyanathan.com). See more here: [http://rcharts.io/](http://rcharts.io/)

Below you will find a series of examples of nice charts using *rcharts* and the corresponding javascript library used in each case.

**Page contents:**

- Introduction
- Examples
- Credits
- License
rCharts Interactive figures: NYT 512 Paths to White House

Obama has 106 ways to win  83% of paths

Romney has 18 ways to win  14% of paths

rCharts: show data on hover & control vars. displayed

Toggle display of variables by clicking on them in legend
rCharts: Easy creation of georeferenced custom maps

```r
map3 <- Leaflet$new()
map3$setView(c(51.505, -0.09), zoom = 13)
map3$marker(c(51.5, -0.09), bindPopup = "Hi, I am a popup")
map3$marker(c(51.495, -0.063), bindPopup = "Hi, I am another popup")
map3$print("chart3")
map3$save("map3.html")
```

rCharts: Interactive magnification of figure regions

```r
n2 <- nPlot(Sepal.Length ~ Sepal.Width, data = sepal, type = "scatterChart",
            group = "Species")
n2$xAxis(axisLabel = "Sepal.Width") # add x axis label
n2$yAxis(axisLabel = "Sepal.Length")
print("nvd3Scatter")

n2$save("n2.html")
```
rCharts: Select time range on X and vars on Y

move slider ends on X axis to filter on new time frame and toggle variables clicking on legend

Clickme: Interactive filtering charts by point names

Groups  Show one
- A (168)
- B (165)
- C (167)
Clickme: highlight data points with partial filter match

Show names (500)

Groups
- Noise (279)
- Significant (221)

Animation in time-based charts

Violent Crime Rate in Decade 1961-1970

CrimeType: Low | Medium | High

Map of the United States with crime rate categories
Ecoengine: distribution maps based on database records

Ecoengine: Photo viewer based on remote ecological data