**Exploring Multivariate Data in R with the googleVis Package[[1]](#footnote-1)**

Using the googleVis package in R, we can create dynamic interactive displays and charts that will revolutionize how you display and interpret your data. Every object created in R through the googleVis package is displayed locally in a browser window for viewing.

Motion Charts:

To create the motion charts made famous by Hans Rosling, you first need to set up your data frame. The data should contain one column as the identifying variable, one column for the time variable, and at least two columns of values for different variables. As an example, your data should look something like this:



This data set is available in its entirety as a .csv file called “MinnesotaData.csv”.

Once you have your data set up this way, the next step is to load the data into R and create the motion chart. The R commands for these operations follow as an example.

> library(googleVis)

> county<- read.csv("MinnesotaData.csv", header=TRUE)

> attach(county)

> countyMotion<- gvisMotionChart(county, idvar='County', timevar='Year')

> plot(countyMotion)

With these selections for the idvar and timevar, the motion chart will display the county names as the bubbles and move through time following the years from 1900-2010.

After plotting the motion chart, a browser will open and display the motion chart. Displayed here is a static motion chart as an example of how they look. These motion charts allow you to do many things and really be in control of how you want to view the data.



To account for the large differences in counties in relation to urban and rural counties, it is easier to view the data in the log-log scale. To change the display to the log-log scale, click on the lin arrow and select log for both the x- and y- axes.





You can change the variables that are displayed on the x- and y- axes by clicking on the axis and selecting the variable you would like to be shown.



You can also choose a variable to be displayed in the color of the bubbles and also the size of the bubbles.





Once you have selected the variables you wish to view, click the arrow button in the bottom left hand corner to watch your data move through time. You can stop the movement at any time and can also use the slider to move to a specific time.

By moving your cursor over the bubbles, you will be able to see the name of the identifying variable, in this example, the county name. You can also select a county by selecting them from the list on the right hand side. You can select as many counties as you wish.



If you play the motion chart with counties selected, you will be able to see the path they take through time.

Notice the three tabs in the upper right hand corner of the motion chart. There are three options for viewing the data: the classic bubble plot, a bar graph, and a time line. Click on the tab to view the different kinds of displays.

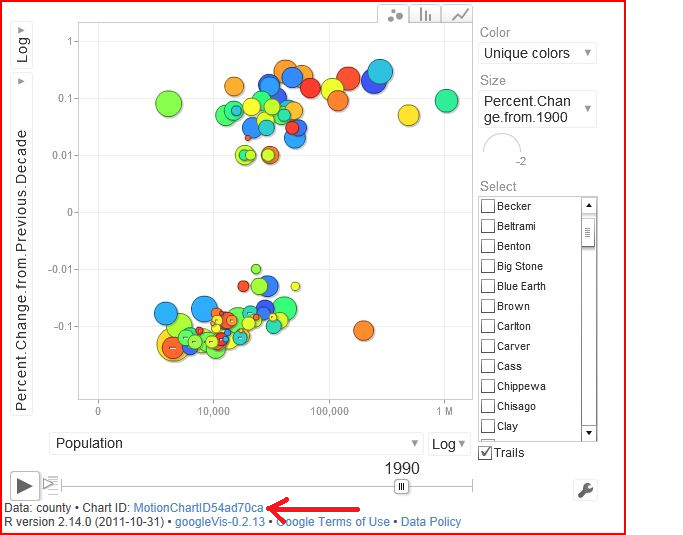
Below is an example of the bar graph. This graph functions in the same way as the bubble chart: you have your choice of variables for the axes, color, and size, and you can also select certain counties again.



The third tab shows a time line of the data. While this display does not have a play button, you can still see the changes in each county through time by moving your cursor over the display or by selecting counties from the list.



To be able to publish the gvisMotionChart to a webpage other than your local browser you need to do the following. Click on the link for the Chart ID at the bottom of the motion chart.



This will take you to the HTML code for the visualization.



Select all the text in this box and copy it. Then paste it into a text editor and save the file as a .txt file. This text file is all you need for publishing the gvisMotionChart to an external webpage.

Geo Map:

You can create interactive maps in R through the googleVis package that have the same features and abilities as the familiar Google maps.

Again, the first step is to set up your data. For the geo map, you need to have two very important columns: one with the latitude and longitude for the locations and one with a tip that will be displayed on the final map. Your data should look something like this:



This data set is available in its entirety as a .csv file called “DP1 County 2010.csv”.

The neat thing about creating the geo map in R is that you can display your data directly on the map. We can choose what data will be displayed on the map by entering this information in the Tip column of our data set. We can enter as many variables as we wish. We also need to specify the latitude and longitude for the locations.

A fast way to create the Tip column in an Excel workbook is to use the concatenate formula. In this example, in cell C2 we would type the following:

=concatenate(A2,”<BR>”,”2010 Population=”,D2)

This formula can then be copied down across all rows. [Note: You will need to cut this column and paste it as values because R will not accept a data set with Excel formulas in it.]

Now, load the data into R and create the geo map.

> library(googleVis)

> geo<- read.csv('DP1 County 2010.csv', header=TRUE)

> attach(geo)

> geoMap<- gvisMap(geo, 'LatLong','Tip',options=list(showTip=TRUE,showLine=FALSE,enableScrollWheel=TRUE,useMapTypeControl=TRUE))

> plot(geoMap)

The plot command will open a browser window and show the geo map.



This map has the same usability features as the familiar Google maps. You can change the display of the map from a satellite image with or without labels to a map of the terrain. You can zoom in and out of the map with either the buttons on the left hand side of the map or by using the scroll wheel on your mouse.

To view the data from the Tip column, simply click on a balloon.



You can get a live image by moving the figure in the upper left hand corner and placing it on the map.



To publish the gvisGeoMap to an external webpage, click the link for the Chart ID, copy all the text in the box and paste it into a text editor. Save the file as a .txt file. This text file is all you need to publish the gvisGeoMap to an external webpage.

Table:

The googleVis package allows you to create very nice looking tables. To do this, simply create the table in Excel and load the data into R. [Note: Any breaks between rows or columns that you want in the final table need to be left blank in the Excel sheet.]

Here is an example of the data in Excel:



This data set is available in its entirety as a .csv file called “Stevens County DP1.csv”.

Load the data into R and create the table.

> library(googleVis)

> stevens<- read.csv("Stevens County DP1.csv",header=TRUE)

> attach(stevens)

> tbl<- gvisTable(stevens)

> plot(tbl)

This opens a browser window showing the table. You can scroll up and down the table and rows become highlighted for easier reading by moving your cursor over them.



To publish the gvisTable to an external webpage, click the link for the Chart ID, copy all the text in the box and paste it into a text editor. Save the file as a .txt file. This text file is all you need to publish the gvisTable to an external webpage.

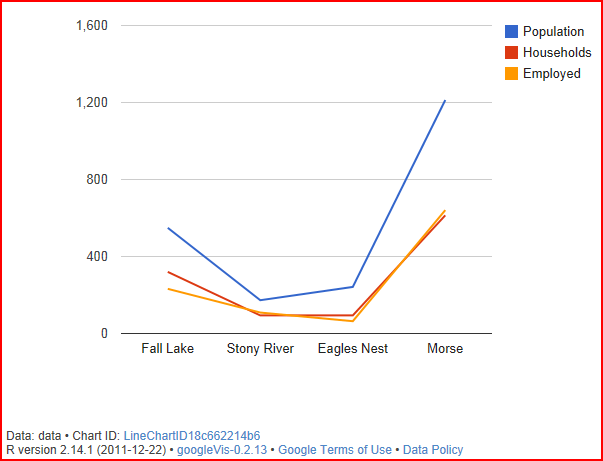
gvisLineChart:

> data<- read.csv("gvisData.csv",header=TRUE)

> attach(data)

> LineChart<- gvisLineChart(data)

> plot(LineChart)



To publish the gvisLineChart to an external webpage, click the link for the Chart ID, copy all the text in the box and paste it into a text editor. Save the file as a .txt file. This text file is all you need to publish the gvisLineChart to an external webpage.

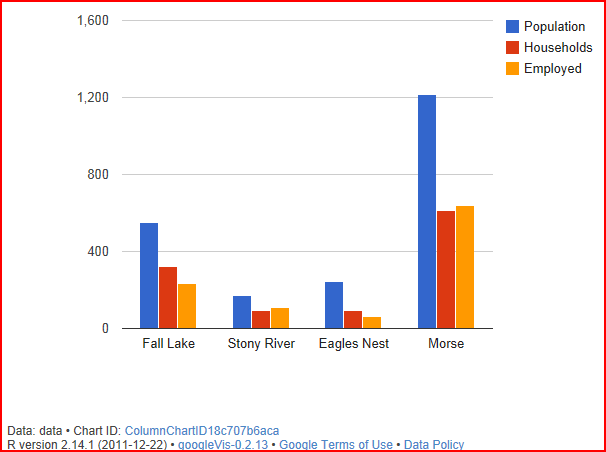
gvisColumnChart:

> data<- read.csv("gvisData.csv",header=TRUE)

> attach(data)

> ColumnChart<- gvisColumnChart(data)

> plot(ColumnChart)



To publish the gvisColumnChart to an external webpage, click the link for the Chart ID, copy all the text in the box and paste it into a text editor. Save the file as a .txt file. This text file is all you need to publish the gvisColumnChart to an external webpage.

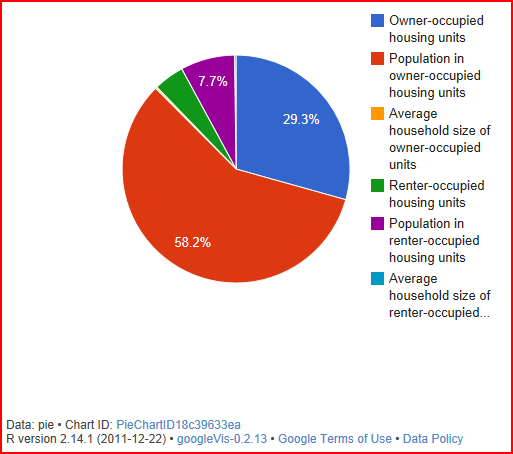
gvisPieChart:

> pie<- read.csv("gvisPieChart.csv",header=TRUE)

> attach(pie)

> pieChart<- gvisPieChart(pie)

> plot(pieChart)



To publish the gvisPieChart to an external webpage, click the link for the Chart ID, copy all the text in the box and paste it into a text editor. Save the file as a .txt file. This text file is all you need to publish the gvisPieChart to an external webpage.

Conclusion:

The googleVis package allows you to not only be able to produce visually appealing and interactive displays, but also a simple way to embed these displays in a webpage. The package can produce many other displays and tables not discussed here and the possibilities for this package and its uses are endless.

Reference:

Markus Gesmann and Diego de Castillo (2011). googleVis: Interface between R and the Google Visualisation API. R package version 0.2.8.

http://CRAN.R-project.org/package=googleVis

1. R statistical computing handout for using the googleVis package prepared for Multivariate Statistical Analysis course by Jessica Orth [↑](#footnote-ref-1)