MOTION CHART GUIDE

Prepared By

Varun Pidyar

Deepak Goyal

DA-IICT

Introduction

googleVis is an R package providing an interface between R and Google Charts. The functions of the package allow the user to visualise data with the Google Chart Tools without uploading their data to Google.

The output of googleVis functions is html code that contains the data and references to JavaScript functions hosted by Google. To view the output a browser with Flash and Internet connection is required, the actual chart is rendered in the browser.

Installtion

- First download R software from the link given below for 32/64 windows operating system. It is an open source software.

  http://cran.r-project.org/bin/windows/base/

- After the download install the software into your system and run R software. Open R console in the software and type the following commands to download googleVis library packages. Choose mirror for your country while downloading.

```
install.packages('googleVis')
```

- After the successfully installation of googleVis library to use the library in your project type the following command.

```
library('googleVis')
```

  A message similar to below one will appear on the screen

```
Welcome to googleVis version 0.5.8
```

```
Please read the Google API Terms of Use
before you start using the package:
https://developers.google.com/terms/

Note, the plot method of googleVis will by default use
the standard browser to display its output.

See the googleVis package vignettes for more details,
or visit http://github.com/mages/googleVis.

To suppress this message use:
suppressPackageStartupMessages(library(googleVis))
```

- After this you can try to run a preinstall motion chart example on the R software to check your software functionality. For this your system must have a installed browser(preferably Mozilla) which will show output of the R script. Try to run the following code in your R software and see if it works.

```
 Motion=gvisMotionChart(Fruits,
                        idvar="Fruit",
                       timevar="Year")
plot(Motion)
```

- Now in order to right your own R script for motion chart it can be done in two ways.

  1) Directly enter fields and value of data and plot motion chart

     ➢ First can be done easily by writing few lines of code directly within R console with data values and fields value .The sample code is given below.

```
library(googleVis)
gender=c("male","male","female","female")
location=c("east","west","east","west")
 year=c(2005,2008,2005,2007)
 age=c(20,50,15,30)
```

```
weight=c(40,80,25,50)
df=data.frame(gender,age,weight,location,year)
m<-gvisMotionChart(df,"gender","year")
plot(m)
```

2) Use a server database by connecting to server and then plotting motion chart of the required fields.

This requires a server(localhost or live) where you can put your data in tables and then by connecting to server using database driver for the particular type of database. We have discussed here connection to mysql database hosted on wampserver.

Download the wampserver for 32/64 bit computer system from the given link below and install it in your computer.

http://www.wampserver.com/en/

For installation of wampserver on your computer the video link might be helpful.
https://www.youtube.com/watch?v=kVc_9vAO7oI

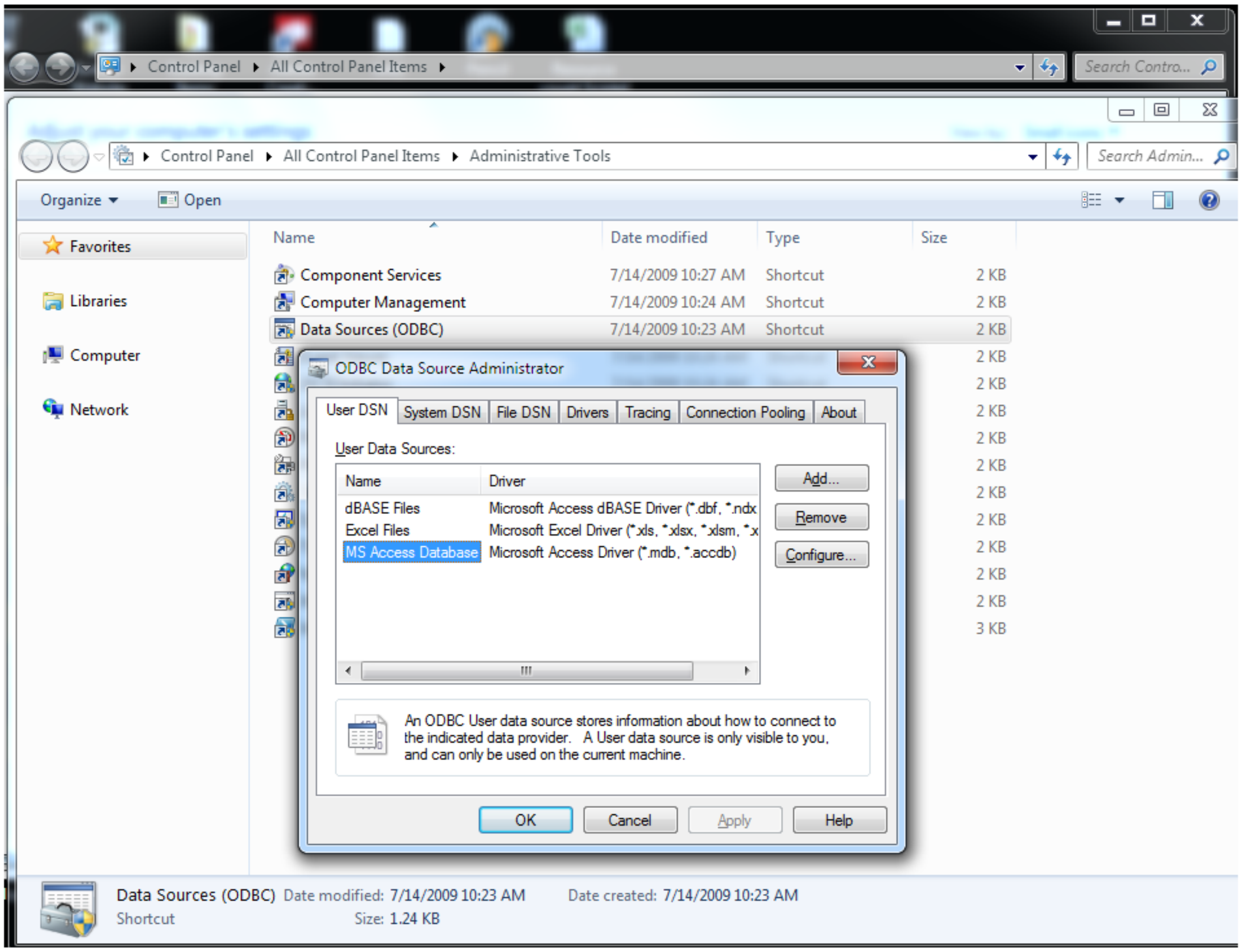
Create tables of My Sql data which you want to analyze before connecting to database using ODBC Drivers.

For connection to my sql database on the server your computer system must have installed mysql odbc database connector which can be download from the given link below.

Step 1:
Download the ODBC Driver from the site https://dev.mysql.com/downloads/connector/odbc/ to make sure if you have the driven if you don’t have one. I have downloaded mysql-connector-odbc-5.3.2-winx64.msi as My operating and System is 64-bit. Please proceed with the installation of the same.

Step 2:

Goto Control Panel->Administrative Tools->ODBC



Name	Date modified	Type	Size
Component Services	7/14/2009 10:27 AM	Shortcut	2 KB
Computer Management	7/14/2009 10:24 AM	Shortcut	2 KB
Data Sources (ODBC)	7/14/2009 10:23 AM	Shortcut	2 KB

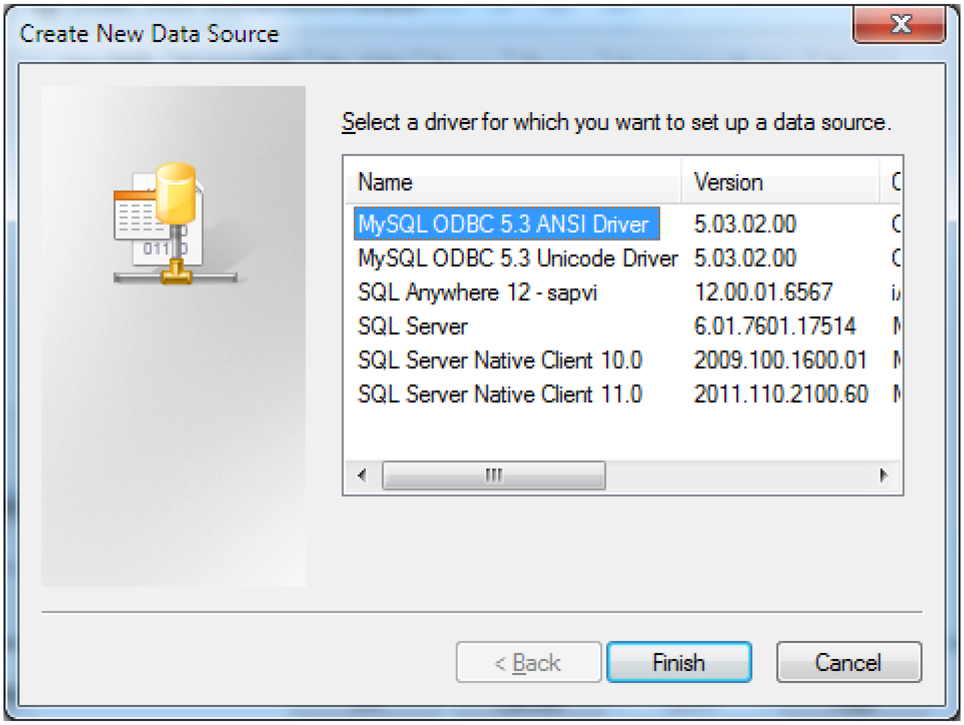
User Data Sources:

Name	Driver
dBASE Files	Microsoft Access dBASE Driver (*.dbf, *.ndx
Excel Files	Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.x
MS Access Database	Microsoft Access Driver (*.mdb, *.accdb)

An ODBC User data source stores information about how to connect to the indicated data provider. A User data source is only visible to you, and can only be used on the current machine.

Data Sources (ODBC) Date modified: 7/14/2009 10:23 AM Date created: 7/14/2009 10:23 AM
Shortcut Size: 1.24 KB

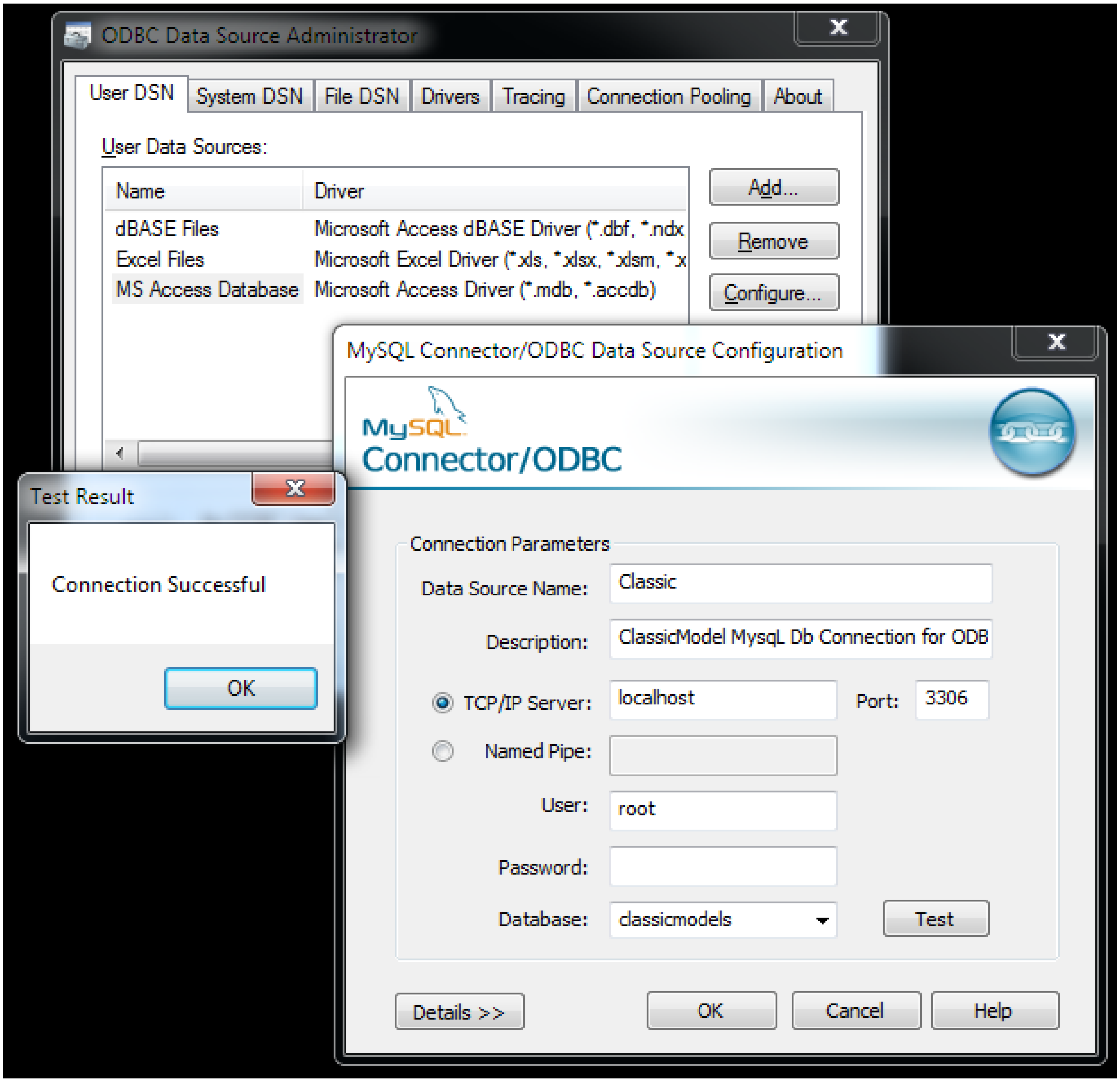
Step 3: Click Add to add a new ODBC Setup for MySQL



Select a driver for which you want to set up a data source.

Name	Version
MySQL ODBC 5.3 ANSI Driver	5.03.02.00
MySQL ODBC 5.3 Unicode Driver	5.03.02.00
SQL Anywhere 12 - sapvi	12.00.01.6567
SQL Server	6.01.7601.17514
SQL Server Native Client 10.0	2009.100.1600.01
SQL Server Native Client 11.0	2011.110.2100.60

Step 4: Once you click finish you will get the below given screen where in you need to feed the IP Address/Hostname of the MySQL Server, Username and Password credentials and database and click on the Test button to make sure you are able to connect to the database without any problems. Once the Test is successful click OK to added to the List of ODBC connections.



User Data Sources:

Name	Driver
dBASE Files	Microsoft Access dBASE Driver (*.dbf, *.ndx
Excel Files	Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.x
MS Access Database	Microsoft Access Driver (*.mdb, *.accdb)

Connection Successful

Data Source Name: Classic

Description: ClassicModel MysqL Db Connection for ODB

● TCP/IP Server: localhost Port: 3306

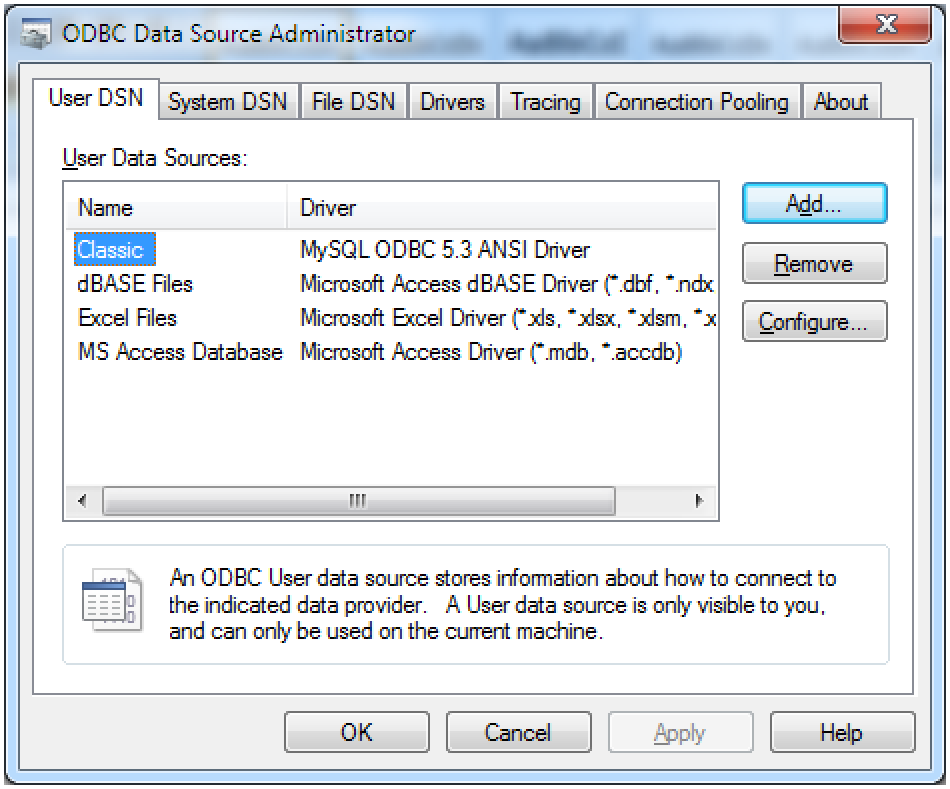
○ Named Pipe:

User: root

Password:

Database: classicmodels

After the MySQL ODBC connection is added:



User Data Sources:

Name	Driver
Classic	MySQL ODBC 5.3 ANSI Driver
dBASE Files	Microsoft Access dBASE Driver (*.dbf, *.ndx
Excel Files	Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.x
MS Access Database	Microsoft Access Driver (*.mdb, *.accdb)

An ODBC User data source stores information about how to connect to the indicated data provider. A User data source is only visible to you, and can only be used on the current machine.

Now we will move to R to invoke this Datasource and try to access any one of the table in R

Step 5: Now load the library RODBC using the following command on the R console, if its not getting loaded you can look at the instructions in the link to install the RODBC Library.

```
chooseCRANmirror()
```

The above command launches a large list of available mirrors. Select the one you want (mine is "India") and hit "OK".

Now for installing the RODBC package, enter the command:

```
utils:::menuInstallPkgs()
```

Scroll down and select "RODBC", finish by clicking "OK". The following should now show up in the console:

```
trying URL 'http://cran-mirror.cs.uu.nl/bin/windows/contrib/2.15/RODBC_1.3-
6.zip'
Content type 'application/zip' length 759673 bytes (741 Kb)
opened URL
downloaded 741 Kb

package �RODBC� successfully unpacked and MD5 sums checked

The downloaded binary packages are in
C:\Documents and Settings\Iwan Luijks\Local
Settings\Temp\RtmpmW37k7\downloaded_packages
```

Now you have RODBC installed. Let's configure a MySQL data source we can use.

- Now for the using RODBC library go back to the R console and type the following commands.

```
library(RODBC)
ch <- odbcConnect("THE_DSN_YOU_ENTERED_BEFORE")
sqlTables(ch)
odbcClose(ch)
```

Where these functions above do the following (line-by-line):

1. load the RODBC library
2. connect to your Data Source using the DSN you specified before and assign that connection to the variable "ch"
3. list all tables
4. close the connection you setup using odbcConnect()

That's actually all.. Now go analyse your data! ☺

One of the sample code using wampserver and my sql database is given below you can also create the same for yourself by following the below commands and above steps.

Before running the below command make sure your localhost or server is already started otherwise it will show an error.

In place of ‘cricket’ used below, use your own data source name.

```
library(RODBC)

db<-odbcConnect("cricket")

sqlTables(db)

p1<-sqlQuery(db,"select p.gender from person p")

p2<-sqlQuery(db,"select p.year from person p")

p3<-sqlQuery(db,"select p.age from person p")

p4<-sqlQuery(db,"select p.weight from person p")

p5<-sqlQuery(db,"select p.location from person p")

df1<-data.frame(p1,p2,p3,p4,p5)

library(googleVis)

M<-gvisMotionChart(df1,"gender","year")

plot(M)
```

Save your File

The R file can be easily stored in any position of the by entering below commands in R which convert the R script to html file and store at the given location. The command is following.

```
        print(M1, file='d:/delete/fruit.html')
```

where M1 is object of gVisMotionChart function() in your code .A sample is shown below.

```
M1 <- gvisMotionChart(Fruits, idvar="Fruit", timevar="Year")
plot(M1)
print(M1, file='d:/delete/fruit.html')
```

To see the out put without loading from R again and again the file should be load from the a server .This can be easily done by saving file in your localhost directory or by putting it on a live server.Also you can use show your localhost file to any other user on internet by tunneling your local ip address from the site given below.

https://forwardhq.com/signup