Back Page Epi Lab | 10-2014 #6 | Connecting to a database - RODBC

Connecting to a database in R

So far in our back page epi labs focussed on R we have used prepared datasets for analysis. This is however quite an artificial situation as most often our field data is sitting in already made databases. For sure you can query out your info in a database, export it to a CSV file and import into R but there is a way to skip this step and query your database (be it SQL or MSAccess) directly and this is what this lab introduces. This lab is really aimed at those of you that are familiar with databases and can make the required test access database. If this is not the case then maybe give this lab a skip or alternatively contact us so that we can get you started with getting a access database going.

Epi Lab color code

Software/Packages/Add-ins required

Software/Packages/Add-ins recommended

Description text

R code to copy/paste into console

R code to copy/paste into console that needs adjustment to your personal workspace

Websites where you can download requirements

Lab #6 requirements

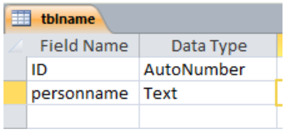
- R - http://cran.r-project.org/bin/windows/base/
- R Studio - www.rstudio.com/ide/download/desktop
- Internet connection
- Microsoft Access installed
- RODBC package (install function in text below)

Preparatory work

There is some preparatory work to do prior to this lab

Create a Microsoft access (2007 + version - so *.accdb and not *.mdb) database on your computer in a folder of your choice - just make sure that you know where to find it later - mine was on my Desktop.

Call the database "test.accdb", open it and create a table called `tblname` with fields as illustrated below



tblname

Field Name	Data Type
ID	AutoNumber
personname	Text

Add a few lines of data into your table, including one `personname` of "`John`" and close the database

The code

```
#Install and load the "RODBC" Package
install.packages("RODBC")
library("RODBC")
#The name of this package is RODBC which is how R utilises and ODBC connection: ODBC means "Open DataBase Connectivity" which is a universal way for programs to link to databases by creating a "tunnel" to the data and pulling the data through
#create a environment variable called acccesloc and put the string where the created database is residing on your computer - note that yours may differ and note the direction of the FORWARD SLASHES
accessloc<-'C:/Documents and Settings/johng/Desktop/test.accdb'
#create a variable which is a channel to the database - we call it channel1 for want of a better name - note the function used is specific to Access2007 and therefore Microsoft Access databases with an *.accdb extension.
channel1<-odbcConnectAccess2007(accessloc)
# we now work through a few function of RODBC to show you how it works. note that the connection is live, so any SQL command you give will work, #including deleting data and database tables and databases!
#lets see what tables we have in the database
sqlTables(channel1)
#note that there you will have at least 13 tables, all but one are system tables that you wont be aware off, but you should see tblname in the last row of your result.
# lets fetch just that table
sqlFetch(channel1,"tblname")
#you should see your rows of data which you entered, including the row with “John” in it
#lets do a basic SQL query now on the data
sqlQuery(channel1,query = "SELECT ID, personname FROM tblname WHERE personname='John'")
#remember you can create a data frame from your query - try
queryresult<-sqlQuery(channel1,query = "SELECT ID, personname FROM tblname WHERE personname='John'")
queryresult
#just to show you to be careful: lets delete the row with John in the personname field
sqlQuery(channel1,query = "DELETE FROM tblname WHERE personname='John'")
#so now try query for "John" - you should get no result!
sqlQuery(channel1,query = "SELECT ID, personname FROM tblname WHERE personname='John'")
#to close all connections (channels) run the following
odbcCloseAll()
```

Continued on next page

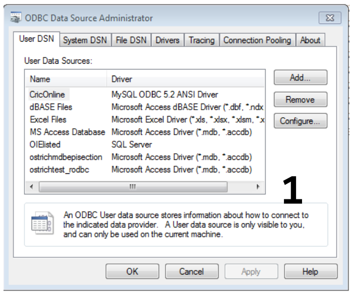
Back Page Epi Lab | 10-2014 #6 | Connecting to a database - RODBC

some more Preparatory work

Now we are going to do the same thing except we use an already made 'channel' on your computer. This is a way of connecting to a database using an already made user DSN specific to your computer. A DSN is a Data Source Name and is used with ODBC, and the application on your computer is one place where this is used - it is essentially a channel registered with your computer. It contains at least the following information:

the `name` of the data source
the `location` of the data source
the `name of a database driver` which can access the data source (so MS Access or MySQL or SQLServer etc.)
a `user ID` for data access (if required)
a `user password` for data access (if required)

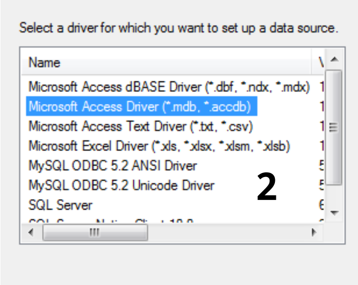
Create a DSN on your computer:
Go to `Control Panel` → (you may need to go to `System and Security` depending on your computer → `Administrative Tools` → `Data Sources` on your computer (fig 1 below)
You will see there are User DSN's, System DSN's and File DSN's - read the description in the bottom section of the window for a definition of each
Click on `ADD` in the `UserDSN` Tab (fig 1 below)
Select the `Microsoft Access Driver (*.mdb, *.accdb)`(fig 2 below) - if this DRIVER is not on your list you'll have to go online and download it
Name your new data source "`rodbc_dsntest`" and click on `SELECT` and go and find your test.accdb database that you made (fig 3 below)



User Data Sources:

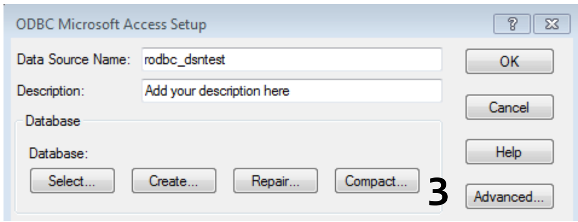
Name	Driver
CricOnline	MySQL ODBC 5.2 ANSI Driver
dBASE Files	Microsoft Access dBASE Driver (*.dbf, *.ndx
Excel Files	Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.x
MS Access Database	Microsoft Access Driver (*.mdb, *.accdb)
OIElisted	SQL Server
ostrichmdbepisection	Microsoft Access Driver (*.mdb, *.accdb)
ostrichtest_rodbc	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.



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

Name
Microsoft Access dBASE Driver (*.dbf, *.ndx, *.mdx)
Microsoft Access Driver (*.mdb, *.accdb)
Microsoft Access Text Driver (*.txt, *.csv)
Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.xlsb)
MySQL ODBC 5.2 ANSI Driver
MySQL ODBC 5.2 Unicode Driver
SQL Server



Data Source Name: rodbc_dsntest
Description: Add your description here
Database:

Click `OK` - you'll now see your new DSN listed under the User DSN's

back to The code

```
#to view all the DSN's on your machine (so similar to your DSN list you have just seen) run:
odbcDataSources()
#you should see you new DSN listed at the bottom
#similar to the first example we make a channel ("channel2" this time except this time we use the DSN and not the direct ODBCConnect method
channel2<-odbcConnect(dsn = "rodbc_dsntest")
#again fetch a table from the database - note "John should not be in your list since you deleted it!
sqlFetch(channel2,"tblname")
#just for fun lets put "John" back into the table
sqlQuery(channel2,query = "INSERT INTO tblname (personname) VALUES ('John')")
Now look at your data again and it "John" should be there
sqlFetch(channel2,"tblname")
#Remember to close your connections
odbcCloseAll()
```

Some notes

In future we can look at some more complex SQL expressions to start querying your databases

There is an `odbcConnectAccess` function in RODBC which directly connects to *.mdb access databases. However, it however only works in 32 bit windows environments, so if you work in 64 bit environments you will get an error when trying to do this method. The best way to connect then is to use the DSN method as shown above

The nice thing about using the second method is that the DSN you create can be used in other programs, so for example if you have a web server SQL database - you can create a DSN to it and using that DSN link to R or link to a program like access and see your linked SQL tables in the Access environment

There are a few other methods and functions in RODBC - have a look at their help file, but we'll be certain to use some more of them in future

References

R Core Team (2014). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/

Brian Ripley and Michael Lapsley (2013). RODBC: ODBC Database Access. R package version 1.3-10. http://CRAN.R-project.org/package=RODBC