Lab 11

Statistical Computing, 36-350

Friday November 20, 2015

Today's agenda: interfacing R with SQL database management.

Background. This lab shows how R interfaces with other programs, particularly SQL database management. We will use the package `RSQLite`, which not only provides an R interface but also installs a minimal library for database access.

Today, we will look trends in baseball team payrolls between the years 1985 and 2010. The data come from the Baseball Databank http://baseball-databank.org and is based in part on Lahman's Baseball Database. Information on the attributes in the database can be found at http://baseball1.com/files/database/readme58.txt. You will need to download the SQLite database file http://www.stat.cmu.edu/~ryantibs/statcomp/labs/baseball.db.

Important information about knitting and submitting this lab. In order to get your Rmd file to knit, you should set your working directory (using `setwd()`) to the folder that contains the "baseball.db" file that you downloaded. **Do not rename this file.** For this lab, submit **both the Rmd file and knitted HTML file** to blackboard.

You should install the R packages `DBI`, `RSQLite`. (The R package `plyr` should have already been installed, from previous labs and homeworks.)

```
library(DBI)
library(RSQLite)
library(plyr)
```

1. Set up a connection to the SQLite database stored in `baseball.db`. Then use `dbListTables()` to list the tables in the database.

2. Use use `dbReadTable()` to grab the table that contains salaries, and store it in a variable called `salaries`. Check that it is of class `data.frame`, and has 21464 rows and 5 columns. Display the first 5 rows.

3. With the `salaries` variable, compute the payroll (total of salaries) for each team in the year 2010. (*Hint*: consider first subsetting out the year 2010, then use `daply()`.) List the teams with the 3 highest payrolls, and the team with the lowest payroll (ouch!).

4. Recompute the payroll for each team in 2010, but now do this using only `dbGetQuery()` and SQL. In particular, use SELECT to create a data frame with two columns, the first giving the team names, and the second the payrolls. (*Hint*: SUM, WHERE, GROUP BY.) Display the first 5 rows of your data frame, and verify that the salaries you computed here are the same as in question 3.

5. Modify the SQL statement to compute the payroll for each team for each year from 1985 to 2010. In particular, use SELECT to create a data frame with 3 columns, the first giving the team names, the second the year, and the third the computed payrolls. (*Hint*: GROUP BY can take two arguments, separated by a comma.) Check that your resulting data frame has 738 rows and 3 columns, and display its first 5 rows.

6. For each baseball time, plot its payroll over time, using the data frame you created in question 5. Make sure the title of each plot has the baseball team's name. Save these plots as a PDF. (*Hint*: write and debug a function to do this for one team's worth of data; then use `d_ply()`.) What is the general trend?

Bonus questions. To make the plots in question 6 more sensible, one needs to adjust for inflation. The following reads in consumer price index or CPI, from the years 1985 through 2011. This has been calculated from FRED (the Federal Reserve Economic Data service), using the package `fImport`.

```
cpi = as.numeric(read.table("http://www.stat.cmu.edu/~ryantibs/statcomp/labs/cpi.txt")[,1])
```

The vector `cpi` is normalized so that 1 = \$1 in 2011. An expression like `y = x/cpi[1990-1985+1]` will convert x 1990 dollars into y 2011 dollars.

7. Plot the CPI as a function of time. Make sure that the horizontal axis is labeled with years, not the positions along the vector.
8. Calculate the inflation-adjusted payroll of each baseball team over time, using the data frame you created in question 5.
9. Plot the inflation-adjusted payroll of each team over time. Again, make sure the title of each plot has the baseball team's name, and save these plots as a PDF. How does this compare to the results in question 6?