- • Research
- • Software
- • Teaching

✂

11 April 2013

Dropbox & R Data

I'm always looking for ways to download data from the internet into R. Though I prefer to host and access plain-text data sets (CSV is my personal favourite) from GitHub (see my short paper on the topic) sometimes it's convenient to get data stored on Dropbox.

There has been a change in the way Dropbox URLs work and I just added some functionality to the repmis R package. So I though that I'ld write a quick post on how to directly download data from Dropbox into R.

The download method is different depending on whether or not your *plain-text* data is in a Dropbox Public folder or not.

Dropbox Public Folder

Dropbox is trying to do away with its public folders. New users need to actively create a Public folder. Regardless, sometimes you may want to download data from one. It used to be that files in Public folders were accessible through non-secure (http) URLs. It's easy to download these into R, just use the read.table command, where the URL is the file name. Dropbox recently changed Public links to be secure (https) URLs. These cannot be accessed with read.table.

Instead you need can use the source_data command from **repmis**:

```
FinURL <-"https://dl.dropbox.com/u/12581470/code/Replicability_code/Fin_Trans_Replication_Journal/Data/public.fin.msm.r

# Download data
FinRegulatorData <- repmis::source_data(FinURL,
                            sep = ",",
                            header = TRUE)
```

Non-Public Dropbox Folders

Getting data from a non-Public folder into R was a trickier. When you click on a Dropbox-based file's Share Link button you are taken to a secure URL, but not for the file itself. The Dropbox webpage you're taken to is filled with lots of other Dropbox information. I used to think that accessing a plain-text data file embedded in one of these webpages would require some tricky web scrapping. Luckily, today I ran across this blog post by Kay Cichini.

With some modifications I was able to easily create a function that could download data from non-Public Dropbox folders. The source_DropboxData command is in the most recent version of **repmis** (v0.2.4) is the result. All you need to know is the name of the file you want to download and its Dropbox key. You can find both of these things in the URL for the webpage that appears when you click on Share Link. Here is an example:

    https://www.dropbox.com/s/exh4iobbm2p5p1v/fin_research_note.csv

The file name is at the very end (`fin_research_note.csv`) and the key is the string of letters and numbers in the middle (`exh4iobbm2p5p1v`). Now we have all of the information we need for `source_DropboxData`:

```
FinDataFull <- repmis::source_DropboxData("fin_research_note.csv",
                                          "exh4iobbm2p5p1v",
                                          sep = ",",
                                          header = TRUE)
```

Posted by Christopher Gandrud at 23:05 ✏

Email ThisBlogThis!Share to TwitterShare to Facebook

Labels: Dropbox, R-project

---

3 comments:

K. Ram said...

You should also just try authenticating with Dropbox directly using my R package, rDrop.. Feedback welcome.

01:35

Christopher Gandrud said...

Yeah, I only just now saw your rDrop package (I added a shout out to it in the repmis README). I'm definitely going to start using it.

Correct me if I'm wrong, but can you use rDrop db.read.csv to access data if you don't have the user's credentials?

01:50

Bob Muenchen said...

Thanks for the helpful post! If you find that rDrop does the same two examples (i.e. without authentication), it would make a great followup article.

Cheers,
Bob

14:11

Post a Comment

Subscribe to: Post Comments (Atom)

Twitter

Tweets by @ChrisGandrud

✂

Labels

academic journals API Apple Store APSA automation Bank of Korea bank regulation banking crisis bash beamer slideshow beamer theme Berlin blogging China citations cloud computing Cloud statistics conference presentation congressional records contestation contingent liabilities Counterfeiting country names credible commitments credit crunch crony capitalism cross-country data D3 data data analysis teaching data hosting data manipulation data storage debt/GDP deck.js deck.rb democratic consolidation diorama disproportionality do-file Dodd-Frank Dropbox election maps election poster elections errors ESRB estimation uncertainty event history analysis Federal Register Federal Reserve financial regulation FOMC forecasting full guarantees functions Gallagher Index GDP growth ggplot2 github googleVis governing majorities graphing uncertainty

Greek debt crisis guarantees Haiti
higher education housing bubble
how-to Ideas IMF IMF country code
inequality inflation install.packages
Ireland Irish asset bubble Japan
javascript Jekyll-Bootstrap jquery
knitr Korea labor market inequality
lag Latent Demand LaTeX lead
lecture slides legislative violence
London School of Economics LSE
maps Markdown missing values
mistakes monetary policy
committee moral hazard museum
tours network graphs neuroscience
non-proportional hazards overseas
development aid Pandoc Pirate Party
plotting political science publishing
R R-project rankings RCurl
regulatory capture relogit repmis
reproducibility reproducible research
research Rstudio Ruby scraping SEC
signalling game simulations
slideshows slidify
source_GitHubData stagnation
standardize Stata state elections
student debt survival analysis
Sweave text analysis The Other Blog
timeline university resource transfer
variable WBI web crawler whaling
South Georgia Island Zelig

Links

- Baseline Scenario
- Felix Salmon
- FT Alphaville
- Global Policy
- Planet Money
- R-bloggers
- Statistical Modelling, Causal Inference, and Social Science
- StatsBlogs
- The Endeavor
- The Monkey Cage

R Posts Subscribe

Subscribe in a reader

Christopher Gandrud. Simple template. Powered by Blogger.