



Getting Data (Part 2)

Jeffrey Leek, Assistant Professor of Biostatistics
Johns Hopkins Bloomberg School of Public Health

Interacting more directly with files

- file - open a connection to a text file
- url - open a connection to a url
- gzfile - open a connection to a .gz file
- bzfile - open a connection to a .bz2 file
- *?connections* for more information
- **Remember to close connections**

readLines() - local file

- readLines - a function to read lines of text from a connection
- Important parameters: *con*, *n*, *encoding*

```
con <- file("../data/cameras.csv", "r")  
cameraData <- read.csv(con)  
close(con)
```

readLines() - local file

```
head(cameraData)
```

	address	direction	street	crossStreet
1	S CATON AVE & BENSON AVE	N/B	Caton Ave	Benson Ave
2	S CATON AVE & BENSON AVE	S/B	Caton Ave	Benson Ave
3	WILKENS AVE & PINE HEIGHTS AVE	E/B	Wilkins Ave	Pine Heights
4	THE ALAMEDA & E 33RD ST	S/B	The Alameda	33rd St
5	E 33RD ST & THE ALAMEDA	E/B	E 33rd	The Alameda
6				
1	Caton Ave & Benson Ave (39.2693779962, -76.6688185297)			
2	Caton Ave & Benson Ave (39.2693157898, -76.6689698176)			
3	Wilkins Ave & Pine Heights (39.2720252302, -76.676960806)			
4	The Alameda & 33rd St (39.3285013141, -76.5953545714)			
5	E 33rd & The Alameda (39.3283410623, -76.5953594625)			
6	Erdman & Macon St (39.3068045671, -76.5593167803)			

readLines() - from the web

```
con <- url("http://simplystatistics.org", "r")
simplyStats <- readLines(con)
close(con)
head(simplyStats)
```

```
[1] "<!DOCTYPE html>"
[2] "<html lang=\"en-US\">"
[3] "<head>"
[4] "<meta charset=\"UTF-8\" />"
[5] "<title>Simply Statistics</title>"
[6] "<link rel=\"profile\" href=\"http://gmpg.org/xfn/11\" />"
```

Reading JSON files {RJSONIO}

You may need to run `install.packages("RJSONIO")` if the RJSONIO package is not already installed

```
library(RJSONIO)
fileUrl <- "https://data.baltimorecity.gov/api/views/dz54-2aru/rows.json?accessType=DOWNLOAD"
download.file(fileUrl, destfile = "./data/camera.json", method = "curl")
con = file("./data/camera.json")
jsonCamera = fromJSON(con)
close(con)
```

Reading JSON files {RJSONIO}

```
head(jsonCamera)
```

```
$meta
$meta$view
$meta$view$id
[1] "dz54-2aru"

$meta$view$name
[1] "Baltimore Fixed Speed Cameras"

$meta$view$attribution
[1] "Department of Transportation"

$meta$view$attributionLink
[1] "http://www.baltimorecity.gov/Government/AgenciesDepartments/Transportation/SpeedMonitoringLocation"

$meta$view$averageRating
[1] 0

$meta$view$category
[1] "Transportation"

$meta$view$createdAt
```

7/16

Writing data - write.table()

- The opposite of read.table
- Important parameters: *x*, *file*, *quote*, *sep*, *row.names*, *col.names*

```
cameraData <- read.csv("./data/cameras.csv")
tmpData <- cameraData[,-1]
write.table(tmpData, file="./data/camerasModified.csv", sep=",")
cameraData2 <- read.csv("./data/camerasModified.csv")
```


Writing data - write.table()

```
head(cameraData2)
```

	direction	street	crossStreet	intersection
1	N/B	Caton Ave	Benson Ave	Caton Ave & Benson Ave
2	S/B	Caton Ave	Benson Ave	Caton Ave & Benson Ave
3	E/B	Wilkins Ave	Pine Heights	Wilkins Ave & Pine Heights
4	S/B	The Alameda	33rd St	The Alameda & 33rd St
5	E/B	E 33rd	The Alameda	E 33rd & The Alameda
6				
1	(39.2693779962, -76.6688185297)			
2	(39.2693157898, -76.6689698176)			
3	(39.2720252302, -76.676960806)			
4	(39.3285013141, -76.5953545714)			
5	(39.3283410623, -76.5953594625)			
6	(39.3068045671, -76.5593167803)			

Writing data - save(), save.image()

- save is used to save R objects
- Important parameters: *list of objects, file*
- save.image saves everything in your working directory

```
cameraData <- read.csv("./data/cameras.csv")  
tmpData <- cameraData[,-1]  
save(tmpData,cameraData,file="./data/cameras.rda")
```

Reading saved data - load()

- Opposite of save()
- Important parameters: *file*

```
# Remove everything from the workspace  
rm(list=ls())  
ls()
```

```
character(0)
```

```
# Load data  
load("./data/cameras.rda")  
ls()
```

```
[1] "cameraData" "tmpData"
```

paste() and paste0()

- These functions are for pasting character strings together.
- Important parameters: *list of text strings, sep*
- paste0() is the same as paste but with *sep=""*
- Great for looping over files
- See also [file.path](#)

```
for(i in 1:5){  
  fileName = paste0("./data",i,".csv")  
  print(fileName)  
}
```

```
[1] "./data1.csv"  
[1] "./data2.csv"  
[1] "./data3.csv"  
[1] "./data4.csv"  
[1] "./data5.csv"
```

Getting data off webpages

Jeff Leek Edit
 Assistant Professor of Biostatistics, Johns Hopkins Bloomberg School of Public Health Edit
[Statistics](#) - [Computing](#) - [Genomics](#) - [Personalized Medicine](#) - [Scientific Communication](#) Edit
 Verified email at [jhsph.edu](mailto:jtleek@jhsph.edu) Edit
 My profile is public Edit [Link](#) [Homepage](#) Edit

Citation indices

	All	Since 2008
Citations	1285	1146
h-index	10	10
i10-index	11	11

Citations to my articles

Select: **All**, **None** Actions Show: **20** 1-20 [Next >](#)

Title / Author	Cited by	Year
<input type="checkbox"/> Significance analysis of time course microarray experiments JD Storey, W Xiao, JT Leek, RG Tompkins, RW Davis Proceedings of the National Academy of Sciences of the United States of ...	338	2005
<input type="checkbox"/> Capturing heterogeneity in gene expression studies by surrogate variable analysis JT Leek, JD Storey PLoS Genetics 3 (9), e161	171	2007
<input type="checkbox"/> EDGE: extraction and analysis of differential gene expression JT Leek, E Monsen, AR Dabney, JD Storey Bioinformatics 22 (4), 507-508	140	2006
<input type="checkbox"/> Tackling the widespread and critical impact of batch effects in high-throughput data JT Leek, RB Scharpf, HC Bravo, D Simcha, B Langmead, WE Johnson, D Geman, K ... Nature Reviews Genetics 11 (10), 733-739	133	2010
<input type="checkbox"/> The optimal discovery procedure for large-scale significance testing, with applications to comparative microarray experiments JD Storey, JY Dai, JT Leek UW Biostatistics Working Paper Series, 260	107	2005
<input type="checkbox"/> Systems-level dynamic analyses of fate change in murine embryonic stem		

Google scholar

Search Authors

My Citations - Help

Follow this author

5 Followers

[Follow new articles](#)
[Follow new citations](#)

Add co-authors

John D. Storey	Add <small>✕</small>
Rafael A. Irizarry	Add <small>✕</small>
Ben Langmead	Add <small>✕</small>
Hector Corrada Br...	Add <small>✕</small>
wenzhong xiao	Add <small>✕</small>
W. Evan Johnson	Add <small>✕</small>
Alexander Lachm...	Add <small>✕</small>
Olga Troyanskaya	Add <small>✕</small>
Avi Ma'ayan	Add <small>✕</small>
Edoardo M Airolidi	Add <small>✕</small>

[View all co-authors](#)

Co-authors

No co-authors

Name

Email

☐ Inviting co-author

[Send invitation](#)

<http://scholar.google.com/citations?user=HI-I6C0AAAAJ&hl=en>

Getting data off webpages

```
library(XML)
con = url("http://scholar.google.com/citations?user=HI-I6C0AAAAJ&hl=en")
htmlCode = readLines(con)
close(con)
htmlCode
```

```
[1] "<!DOCTYPE html><html><head><title>Jeff Leek - Google Scholar Citations</title><meta name=\"robots\"</meta></head></html>"
```

Getting data off webpages

```
url <- "http://scholar.google.com/citations?user=HI-I6C0AAAAJ&hl=en"
html3 <- htmlTreeParse(url, useInternalNodes=T)

xpathSApply(html3, "//title", xmlValue)
```

```
[1] "Jeff Leek - Google Scholar Citations"
```

```
xpathSApply(html3, "//td[@id='col-citedby']", xmlValue)
```

```
[1] "Cited by" "388"      "215"      "194"      "167"      "119"
[7] "116"      "113"      "92"       "76"       "26"       "18"
[13] "18"       "16"       "13"       "11"       "10"       "9"
[19] "8"        "6"        "4"
```

Further resources

- Packages:
 - [httr](#) - for working with http connections
 - [RMySQL](#) - for interfacing with MySQL
 - [bigmemory](#) - for handling data larger than RAM
 - [RHadoop](#) - for interfacing R and Hadoop (by [Revolution Analytics](#))
 - [foreign](#) - for getting data into R from SAS, SPSS, Octave, etc.
- Reading/writing R videos [Part 1](#), [Part 2](#)