23 Cleaning Data Video Lecture – Reading from the web

**Scraping data out of websites**

1. **Webscraping**: Programmatically extracting data from the HTML code of websites
2. useful for getting data
3. many websites have info you may want to programmatically read
4. This can be against the TOS of a website
5. Don’t be a bot about it; it can get your IP address blocked

ASDF

**Example: Google Scholar**

1. get URL <https://scholar.google.com/citations?user=HI-I6C0AAAAJ>
2. open a connection
   1. con = url("https://scholar.google.com/citations?user=HI-I6C0AAAAJ")
3. read the HTML
   1. htmlCode = readLines(con)
4. close the con
   1. close(con)
5. print out the code
   1. htmlCode

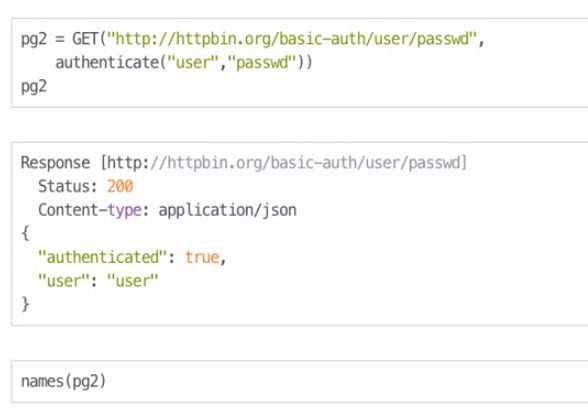
**Parsing with XML**

1. library(XML)
2. url <- “url”
3. html <- htmlTreeParse(url,useInternalNodes=T)
4. xpathSApply(html, “//[stuff]]”, xmlValue)

**GET using the httr package** \*\* seems to like me better than XML alone

1. library(httr)
2. html2 = GET(“url”)
3. content2 = content(html2, as=”text”)
4. parsedHtml = htmlParse(content2, asText=TRUE)
5. xpathSApply(parsedHtml, “//title”, xmlValues)

**Accessing Websites with Passwords**

1. replace user and passwd with the known user names and password:
2. 

**Using Handles** useful for accessing the authentication across several pages on a web page.

1. google = handle(“http://google.com”)
2. pg1 = GET(handle=google, path=/)

More Webscaping tutorials

[www.r-bloggers.com/?s=Web+Scraping](http://www.r-bloggers.com/?s=Web+Scraping)

httr help file, <http://cran.r-project.org/web/packages/httr/httr.pdf>