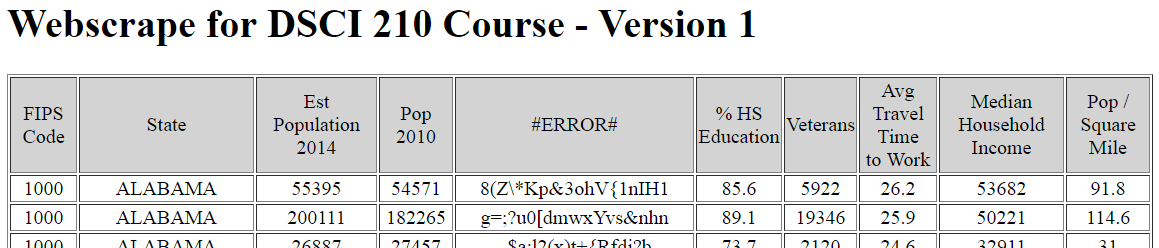
R: Web Scraping

Web scrapping in R is accomplished through the **rvest** package.

library(rvest)

Example #1

Website: <http://course1.winona.edu/cmalone/dsci210/Exams/USQuickFacts_Webscrape1.html>



HTML Snipit



The data to be scraped sits inside a HTML table, thus the function html\_table() should be used to put data into a useable R format.

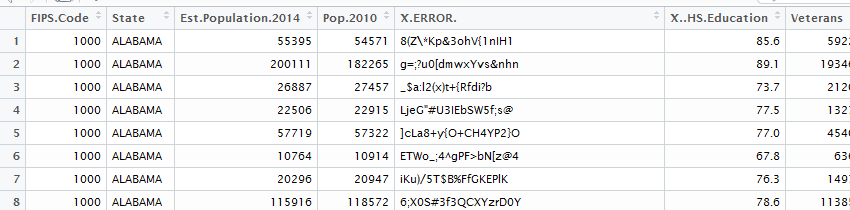
url <- "http://course1.winona.edu/cmalone/dsci210/Exams/USQuickFacts\_Webscrape1.html"

page <- read\_html(url)

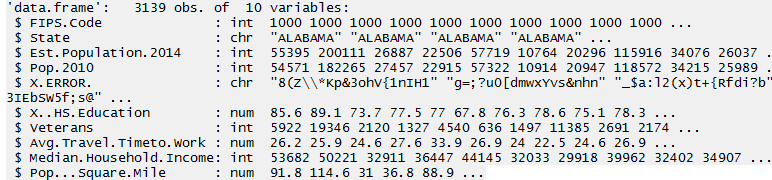
temptable <- html\_table(page, header=TRUE)

df<-as.data.frame(temptable)

Data put into a data.frame in R



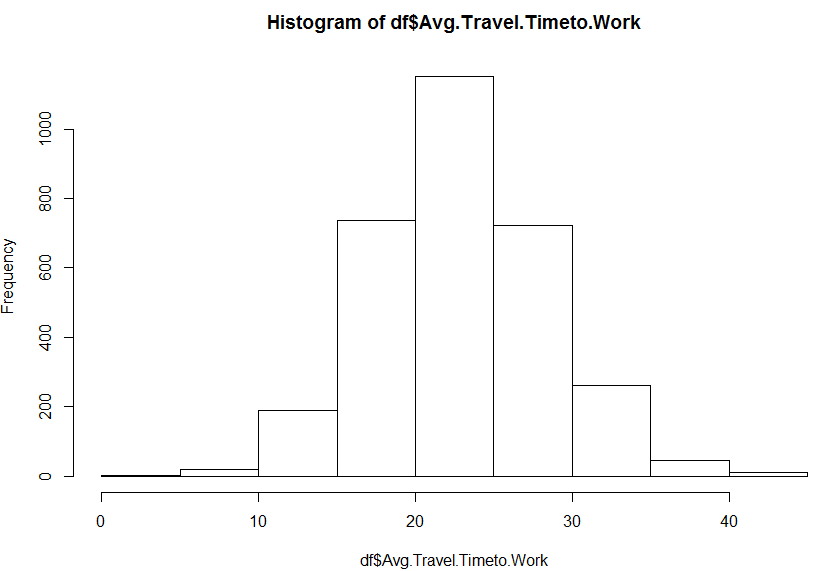
> str(df)



Getting summaries for Avg. Travel Time to Work

summary(df$Avg.Travel.Timeto.Work)  
hist(df$Avg.Travel.Timeto.Work)

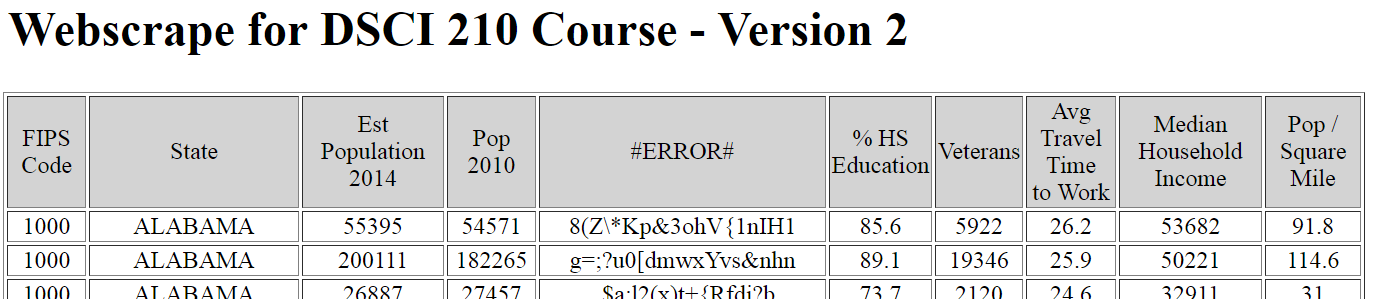




Example #2

Consider a slightly modified version of the HTML table read in above.

Website: <http://course1.winona.edu/cmalone/dsci210/Exams/USQuickFacts_Webscrape2.html>



HTML Snipit



R code to read in this version of webpage. Note the [[2]] is pulling off the second table instance on this webpage.

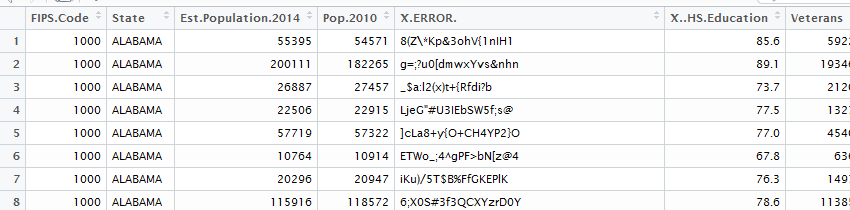
url <- "http://course1.winona.edu/cmalone/dsci210/Exams/USQuickFacts\_Webscrape2.html"

page <- read\_html(url)

temptable <- html\_table(page, header=TRUE)[[2]]

df<-as.data.frame(temptable)

This data as read in my R and put into a data.frame in R



Example #3

For this example, we will consider a more typical webpage. For illustrative purposes, we will use the Social Security Admirations website. Information regarding the Number of Social Security Card Holders over time will be scraped.

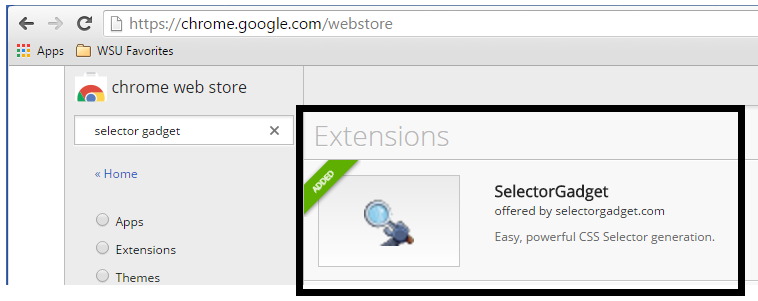
Website: <https://www.ssa.gov/oact/babynames/numberUSbirths.html>



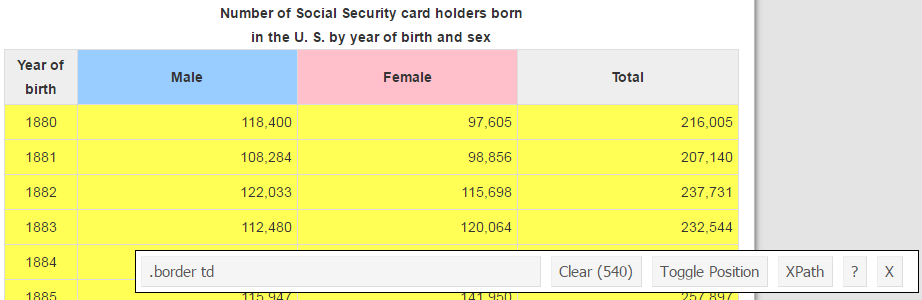
HTML Snipit



The Chrome Extension: SelectorGadget can be used to help identify which information from this website should be gathered. The SelectorGradget extension for Chrome must be downloaded first. To use the extension, click the icon near the upper-right corner. Click on the content to be gathered. You may need to deselect content that is not needed.



Using the SelectorGadget inside Chrome.



R code to scrape this data

url<-"https://www.ssa.gov/oact/babynames/numberUSbirths.html"

page <- read\_html(url)

temp<-html\_text(html\_nodes(page, ".border td"))

df<-as.data.frame(temp)

Structure of data.frame is not correct. The Year, Male, Female, Total are stacked vertically.



Data processing steps. Note: gsub() function is necessary to remove comma in counts before converting to numeric.

rowid<-rep(c(1,2,3,4),135)

Year<-as.numeric(as.character(df$temp[rowid==1]))

Males<-as.numeric(gsub(",","",df$temp[rowid==2]))

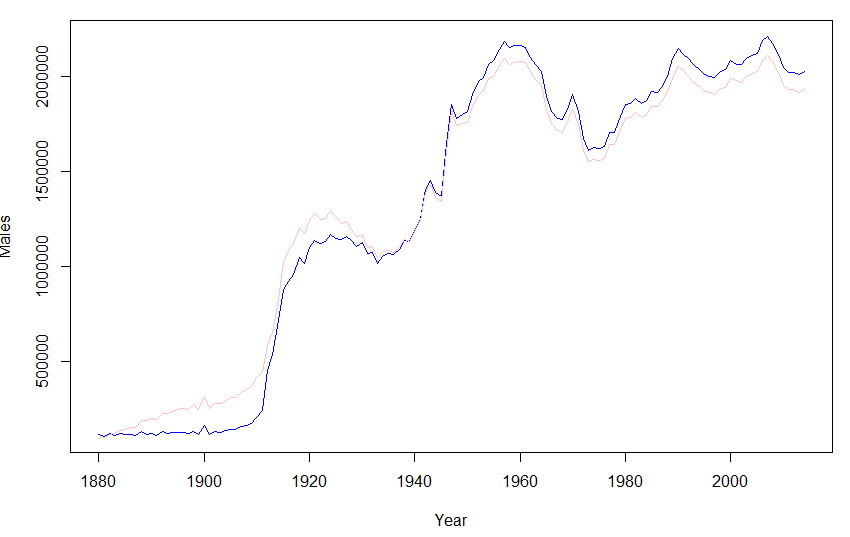
Females<-as.numeric(gsub(",","",df$temp[rowid==3]))

Getting a simple plot to compare trends over time between Males and Females.

plot(Year,Males,type='n') #Points are not being plotted here.

lines(Year,Males,col="blue")

lines(Year,Females,col='pink')



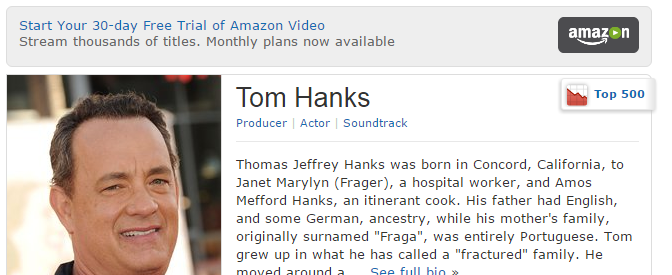
Example #4

This last example will require a two-step process to gather the information needed. The goal here is to compare the movie ratings of two celebrities, Tom Hanks and Julia Roberts.

Step 1: Using the IMDB database, identify the movies for each person.

Step 2: Once the appropriate movies have been identified, scrape the movie ratings from the IMDB   
 database

IMBD webpage for Tom Hanks: <http://www.imdb.com/name/nm0000158/>



The movies for which Tom Hanks has been an actor are displayed near the bottom of this webpage.



#Code to scrape movies for Tom Hanks

hanks<- read\_html("http://www.imdb.com/name/nm0000158/")

hanks\_links <- hanks %>%

html\_nodes("#filmo-head-actor+ .filmo-category-section b a") %>%

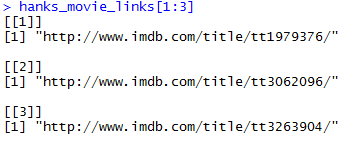
html\_attr("href")

hanks\_movies<-substr(hanks\_links,8,16)

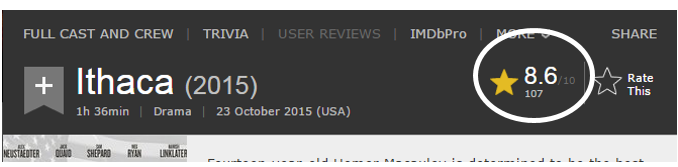
hanks\_movie\_links<-lapply(hanks\_movies,function(x) paste("http://www.imdb.com/title/",x,"/",sep=""))

Note: This code uses piping. This is common in unix type environments and can be used in R. The output is automatically pushed into the next statement.

The code above produces the web addresses of each movie for which Tom Hanks is an actor.



The next step is to obtain the movie rating from each movie. Some movies listed do not have a rating. Thus, an if() statement is used to skip over such movies.



The following will scrape the movie rating for all Tom Hanks movies.

number\_hanks<-length(hanks\_movie\_links)

hanks\_ratings<-rep(NA,number\_hanks)

for(i in 1:number\_hanks){

Sys.sleep(0.5)

temp <- read\_html(hanks\_movie\_links[[i]])

temp2 <- html\_nodes(temp,".ratingValue strong span")

temp3 <- html\_text(temp2)

if(length(temp3)>0){

hanks\_ratings[i] <- as.numeric(temp3)

}

}

The same process must be followed for getting the movie ratings for Julia Roberts.

Note: The html\_nodes() function has a change as the css element is identified as actress instead of actor. This can be seen on the fourth line below.

#Getting information on Julia Roberts

roberts<- read\_html("http://www.imdb.com/name/nm0000210/")

roberts\_links <- roberts %>%

html\_nodes("#filmo-head-actress+ .filmo-category-section b a") %>%

html\_attr("href")

roberts\_movies<-substr(roberts\_links,8,16)

roberts\_movie\_links<-lapply(roberts\_movies,function(x) paste("http://www.imdb.com/title/",x,"/",sep=""))

number\_roberts<-length(roberts\_movie\_links)

roberts\_ratings<-rep(NA,number\_roberts)

for(i in 1:number\_roberts){

Sys.sleep(0.5)

temp <- read\_html(roberts\_movie\_links[[i]])

temp2 <- html\_nodes(temp,".ratingValue strong span")

temp3 <- html\_text(temp2)

if(length(temp3)>0){

roberts\_ratings[i] <- as.numeric(temp3)

}

}

The following code is used to prepare the data for plotting. The main issue with creating this plot is that the number of movies is not the same for each. Thus, to evaluate trends from early in their career to later, the vector of shorter length, Roberts in this case, need to be shifted. A variety of plotting options were used here as well.

#Shifting Roberts ratings to align with Hanks, i.e. start of career

#How much shift is necessary

shiftamount<-length(hanks\_ratings) - length(roberts\_ratings)

#Shifting the data for Roberts

roberts\_ratings<-c(rep(NA,shift\_amount),roberts\_ratings)

#Plotting of ratings over time

plot(hanks\_ratings, xlab="", ylab="Rating", ylim=c(0,10),axes=F, type="n")

lines(hanks\_ratings, col="blue")

axis(1, at=c(0,80), labels=c("Most Recent", "Start of Career"))

axis(2)

lines(roberts\_ratings, col="red")

#Add legend to plot

legend("bottomright", c("Hanks","Roberts"),lty=c(1,1), col=c("blue","red"))

#Mean differnce in ratings

difference<-hanks\_ratings - roberts\_ratings

mean(difference, na.rm=T)

title(paste("Average of (Hanks - Roberts) is",round(mean(difference,na.rm=T),2)))

The final plot that can be used to compare the movie rating between Tom Hanks and Julia Roberts.

