Web Scraping Box Office Data

Jonathan Che

10 April 2017

Overview

The-numbers.com has a huge amount of information about the box office sales of movies. We're interested in capturing information about when tickets are sold in order to estimate public exposure to movie violence and expected movie violence on a given day.

Dahl and DellaVigna use information on the top 50 movies from 1995-2004. The-numbers.com, though, only has daily box-office numbers for the top 10 movies of each year from mid-August 1997 onward. Thus, Dahl and DellaVigna impute sales for the remaining movies (described in their Appendix I). I will only use the top 10 movies of each year from 1998-2004 to begin with.

Method

First, I do some variable setup.

```
url1 <- "http://www.the-numbers.com/market/"
url2 <- "/summary"
years <- seq(from=1998, to=2004)
master_list <- list()
movie_sales_year_list <- list()
```

Next, I make a function that does the following: 1) Go to the "summary" page for a given year 2) Visit each of the top 10 movies for that year 3) Pull box office information for each of those movies

```
extract_year <- function(year){
  url <- str_c(url1, year, url2, sep="")
  webpage <- read_html(url)
  movie_names <- webpage %>%
    html_nodes("table:nth-child(1) b a") %>%
    html_text()
  movie_urls <- webpage %>%
    html_nodes("table:nth-child(1) b a") %>%
    html_attr("href") %>%
    str_replace("summary", "box-office")
  movie_urls <- str_c("http://www.the-numbers.com", movie_urls)
  for(i in 1:10){
    moviepage <- movie_urls[i]
    daily_sales <- moviepage %>%
      read_html() %>%
      # html_nodes(xpath=`//*[(@id = "box_office_chart") and (((count(preceding-sibling::*) + 1) = 8) a
      # html_nodes(xpath=`//*[@id="box_office_chart"]/table`) %>%
      # html_nodes("#box_office_chart:nth-child(8)") %>%
      html_nodes("#box_office_chart > table") %>%
      html_table()
    daily_sales <- daily_sales[[2]]
    daily_sales <- cbind(Title=movie_names[i], daily_sales)
    movie_sales_year_list[[i]] <- daily_sales
```

```
  }
  count <- year - years[1] + 1
  master_list[[count]] <<- rbindlist(movie_sales_year_list)
}
```

Then, I run the function on years 1998-2004.

```
for (i in years){
  extract_year(i)
  # print(str_c("Finished ", i, sep=""))
}
movie_sales <- rbindlist(master_list)
```

At this point, the data are consolidated, though they are still somewhat messy. I do some cleaning before I output everything.

```
names(movie_sales) <- c("Title", "Date", "Rank", "Gross", "Percent_Change", "Theaters", "Gross_Per_Thea

movie_sales <- movie_sales %>%
  mutate(Date = ymd(Date)) %>%
  mutate(Year = year(Date)) %>%
  mutate(Month = month(Date)) %>%
  mutate(Day = day(Date)) %>%
  mutate(Weekday = wday(Date)) %>%
  # mutate(Rank = as.numeric(Rank)) %>%
  mutate(Gross = as.numeric(str_replace_all(Gross, "[$,]",""))) %>%
  # mutate(Percent_Change = as.numeric(str_replace_all(Percent_Change, "[^-0-9]", ""))) %>%
  # mutate(Theaters = as.numeric(str_replace_all(Theaters, "[,]",""))) %>%
  # mutate(Gross_Per_Theater = as.numeric(str_replace_all(Gross_Per_Theater, "[$,]",""))) %>%
  # mutate(Cumul_Gross = as.numeric(str_replace_all(Gross, "[^0-9]",""))) %>%
  mutate(Days_Since_Release = as.numeric(str_replace_all(Days_Since_Release, "[,]", "")))
movie_sales <- movie_sales[,c("Title","Year","Month","Day","Weekday","Gross","Days_Since_Release")]
```

Finally, I export a csv and a rda file.

```
write_csv(movie_sales, path="movie_sales.csv")
saveRDS(movie_sales, "movie_sales.rds")
```