Assignment 3

Kangrui Liu

2023-10-15

You may work in pairs or individually for this assignment. Make sure you join a group in Canvas if you are working in pairs. Turn in this assignment as an HTML or PDF file to ELMS. Make sure to include the R Markdown or Quarto file that was used to generate it. Include the GitHub link for the repository containing these files.

- https://github.com/krliu67/Assignment_SURV727/tree/main/a3

```
library(xml2)
library(rvest)
library(tidyverse)
library(jsonlite)
library(robotstxt)
library(RSocrata)
library(dplyr)
library(tidytext)
library(tm)
```

Web Scraping

In this assignment, your task is to scrape some information from Wikipedia. We start with the following page about Grand Boulevard, a Chicago Community Area.

https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago

The ultimate goal is to gather the table "Historical population" and convert it to a `data.frame`.

As a first step, read in the html page as an R object. Extract the tables from this object (using the `rvest` package) and save the result as a new object. Follow the instructions if there is an error. Use `str()` on this new object -- it should be a list. Try to find the position of the "Historical population" in this list since we need it in the next step.

```
link <- "https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago"
paths_allowed(link)
```

```
##  en.wikipedia.org
```

```
## [1] TRUE
```

```
url <- read_html(link)
```

```
# find other links
other_places <- html_element(url, xpath = '//*[contains(concat( " ", @class, " " ), concat( " ", "navbo
# get links of other places
place_links <- other_places %>% html_nodes("a") %>% html_attr("href")%>% data.frame()
for (i in 1:dim(place_links)[1]) {
  place_links[i,1] <- paste0("https://en.wikipedia.org",place_links[i,1], collapse = NULL)
}
names(place_links)[1] <- "link"
# get names of places
place_names <- other_places %>% html_nodes("a") %>% html_attr("title") %>% data.frame()
names(place_names)[1] <- "name"
place_names <- place_names %>% separate(name,c("place","city"),",")
# combine names and links
links <- cbind(place_links,place_names)
rm(place_links)
rm(place_names)
# combine all links
links[dim(links)[1]+1,] <- c("https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago","Grand Boulevard"

trimws(links$city)
```

```
## [1] "Chicago"                "Chicago"                "Chicago"
## [4] "Chicago"                "Chicago"                "Chicago (neighborhood)"
## [7] "Chicago"                "Chicago"
```

```
head(links)
```

```
##                                                                 link
## 1                 https://en.wikipedia.org/wiki/Armour_Square,_Chicago
## 2                      https://en.wikipedia.org/wiki/Douglas,_Chicago
## 3                      https://en.wikipedia.org/wiki/Oakland,_Chicago
## 4                   https://en.wikipedia.org/wiki/Fuller_Park,_Chicago
## 5                      https://en.wikipedia.org/wiki/Kenwood,_Chicago
## 6 https://en.wikipedia.org/wiki/Washington_Park,_Chicago_(neighborhood)
##             place                    city
## 1   Armour Square                 Chicago
## 2         Douglas                 Chicago
## 3         Oakland                 Chicago
## 4     Fuller Park                 Chicago
## 5         Kenwood                 Chicago
## 6 Washington Park  Chicago (neighborhood)
```

Extract the "Historical population" table from the list and save it as another object. You can use subsetting via `[[…]]` to extract pieces from a list. Print the result.

You will see that the table needs some additional formatting. We only want rows and columns with actual values (I called the table object `pop`).

```
# scrape data
tables <- data.frame()
for (i in 1:dim(links)[1]){
  url_temp <- read_html(links$link[i])
  html_temp <- html_element(url_temp, xpath = '//*[contains(concat( " ", @class, " " ), concat( " ", "us
```

```
  if(length(html_temp)==0){
    html_temp <- html_element(url_temp, xpath = '//*[contains(concat( " ", @class, " " ), concat( " ", 
  }
  table_temp <- html_table(html_temp)
  table_temp$City <- links$city[i]
  table_temp$Place <- links$place[i]
  table_temp <- table_temp[,-3]
  table_temp <- table_temp[-dim(table_temp)[1],]
  tables <- rbind(tables,table_temp)
}
# change column name
names(tables)[3] <- "Changes"
names(tables)[2] <- "Pop"
# remove temps
rm(url_temp)
rm(table_temp)
rm(html_temp)

head(tables)
```

```
## # A tibble: 6 x 5
##   Census Pop    Changes City       Place
##   <chr>  <chr>  <chr>   <chr>      <chr>
## 1 1930   21,450 -       " Chicago" Armour Square
## 2 1940   18,472 -13.9%  " Chicago" Armour Square
## 3 1950   23,294 26.1%   " Chicago" Armour Square
## 4 1960   15,783 -32.2%  " Chicago" Armour Square
## 5 1970   13,063 -17.2%  " Chicago" Armour Square
## 6 1980   12,475 -4.5%   " Chicago" Armour Square
```

```
# clean data, turn char to number
tables$Pop <- gsub(",",'',tables$Pop)
for (i in 1:dim(tables)[1]) {
  temp <- tables$Changes[i]
  temp <- gsub("%",'',temp)
  if(temp == "-"){
    temp = as.numeric(0.00)
  }else if(str_detect(temp,"-")== TRUE){
    temp <- gsub("-",'',temp)
    temp <- as.numeric(temp)
    temp = -1*temp
    temp <- temp/100
  }else{
    temp <- as.numeric(temp)
    temp <- temp/100
  }
  tables$Changes[i] <- temp
}
rm(temp)

tables$Changes <- as.numeric(tables$Changes)
tables$Pop <- as.numeric(tables$Pop)
head(tables)
```

```
## # A tibble: 6 x 5
##   Census   Pop Changes City       Place
##   <chr>  <dbl>   <dbl> <chr>      <chr>
## 1 1930   21450   0     " Chicago" Armour Square
## 2 1940   18472  -0.139 " Chicago" Armour Square
## 3 1950   23294   0.261 " Chicago" Armour Square
## 4 1960   15783  -0.322 " Chicago" Armour Square
## 5 1970   13063  -0.172 " Chicago" Armour Square
## 6 1980   12475  -0.045 " Chicago" Armour Square
```

Expanding to More Pages

That's it for this page. However, we may want to repeat this process for other community areas. The Wikipedia page https://en.wikipedia.org/wiki/Grand_Boulevard,/_Chicago has a section on "Places adjacent to Grand Boulevard, Chicago" at the bottom. Can you find the corresponding table in the list of tables that you created earlier? Extract this table as a new object.

Then, grab the community areas east of Grand Boulevard and save them as a character vector. Print the result.

We want to use this list to create a loop that extracts the population tables from the Wikipedia pages of these places. To make this work and build valid urls, we need to replace empty spaces in the character vector with underscores. This can be done with `gsub()`, or by hand. The resulting vector should look like this: "Oakland,_Chicago" "Kenwood,_Chicago" "Hyde_Park,_Chicago"

To prepare the loop, we also want to copy our `pop` table and rename it as `pops`. In the loop, we append this table by adding columns from the other community areas.

```
# I actually do this part in my second R block, but collect all places info.
# filter get east places
east <- other_places %>% html_table() %>% data.frame() %>% na.omit()
east <- subset(east, !apply(is.na(east) | east == "", 1, all))
east <- east[,-c(1,2)]
east <- gsub(", Chicago","",east)
east <- east %>% c("Grand Boulevard")

east_tables <- subset(tables, Place %in% east)
east_links <- subset(links, place %in% east)
```

Build a small loop to test whether you can build valid urls using the vector of places and pasting each element of it after `https://en.wikipedia.org/wiki/` in a for loop. Calling `url` shows the last url of this loop, which should be `https://en.wikipedia.org/wiki/Hyde_Park,_Chicago`.

Finally, extend the loop and add the code that is needed to grab the population tables from each page. Add columns to the original table `pops` using `cbind()`.

Scraping and Analyzing Text Data

Suppose we wanted to take the actual text from the Wikipedia pages instead of just the information in the table. Our goal in this section is to extract the text from the body of the pages, then do some basic text cleaning and analysis.

First, scrape just the text without any of the information in the margins or headers. For example, for "Grand Boulevard", the text should start with, "**Grand Boulevard** on the South Side of Chicago, Illinois, is one of

the …". Make sure all of the text is in one block by using something like the code below (I called my object `description`).

```
# get description of each place
# description <- description %>% paste(collapse = ' ')
for (i in 1:dim(east_links)[1]){
  url_temp <- read_html(east_links$link[i])
  html_temp <- html_element(url_temp, xpath = '//p[(((count(preceding-sibling::*) + 1) = 6) and parent:
  if(length(html_temp) == 0){
    html_temp <- html_element(url_temp, xpath = '//p[(((count(preceding-sibling::*) + 1) = 7) and paren
  }
  if(length(html_temp) == 0){
    html_temp <- html_element(url_temp, xpath = '//p[(((count(preceding-sibling::*) + 1) = 8) and paren
  }
  if(length(html_temp) == 0){
    html_temp <- html_element(url_temp, xpath = '//p[(((count(preceding-sibling::*) + 1) = 9) and paren
  }
  desc_temp <- html_text(html_temp)
  desc_temp <- desc_temp %>% paste(collapse = ' ')
  east_links$desc[i] <- desc_temp
}
# remove temps
rm(url_temp)
rm(desc_temp)
rm(html_temp)

trimws(east_links$desc)
```

```
## [1] "Oakland, located on the South Side of Chicago, Illinois, USA, is one of 77 officially designate
## [2] "Kenwood, one of Chicago's 77 community areas, is on the shore of Lake Michigan on the South Sid
## [3] "Hyde Park is the 41st of the 77 community areas of Chicago. It is located on the South Side, nea
## [4] "Grand Boulevard on the South Side of Chicago, Illinois, is one of the city's Community Areas. T
```

```
head(east_links)
```

```
##                                                 link           place
## 3         https://en.wikipedia.org/wiki/Oakland,_Chicago         Oakland
## 5         https://en.wikipedia.org/wiki/Kenwood,_Chicago         Kenwood
## 7       https://en.wikipedia.org/wiki/Hyde_Park,_Chicago       Hyde Park
## 8 https://en.wikipedia.org/wiki/Grand_Boulevard,_Chicago Grand Boulevard
##       city
## 3  Chicago
## 5  Chicago
## 7  Chicago
## 8  Chicago
##
## 3
## 5 Kenwood, one of Chicago's 77 community areas, is on the shore of Lake Michigan on the South Side o
## 7
## 8
```

Using a similar loop as in the last section, grab the descriptions of the various communities areas. Make a tibble with two columns: the name of the location and the text describing the location.

```
texts <- as_tibble(data.frame(east_links$place,east_links$desc))
```

Let's clean the data using `tidytext`. If you have trouble with this section, see the example shown in https://www.tidytextmining.com/tidytext.html

```
stop_words <- stopwords("en")
```

Create tokens using `unnest_tokens`. Make sure the data is in one-token-per-row format. Remove any stop words within the data. What are the most common words used overall?

```
words <- texts %>%
  unnest_tokens(word, east_links.desc)

words <- words %>%
  filter(!(word %in% stop_words))

head(texts)
```

```
## # A tibble: 4 x 2
##   east_links.place east_links.desc
##   <chr>            <chr>
## 1 Oakland          "Oakland, located on the South Side of Chicago, Illinois, US~
## 2 Kenwood          "Kenwood, one of Chicago's 77 community areas, is on the sho~
## 3 Hyde Park        "Hyde Park is the 41st of the 77 community areas of Chicago.~
## 4 Grand Boulevard  "Grand Boulevard on the South Side of Chicago, Illinois, is ~
```

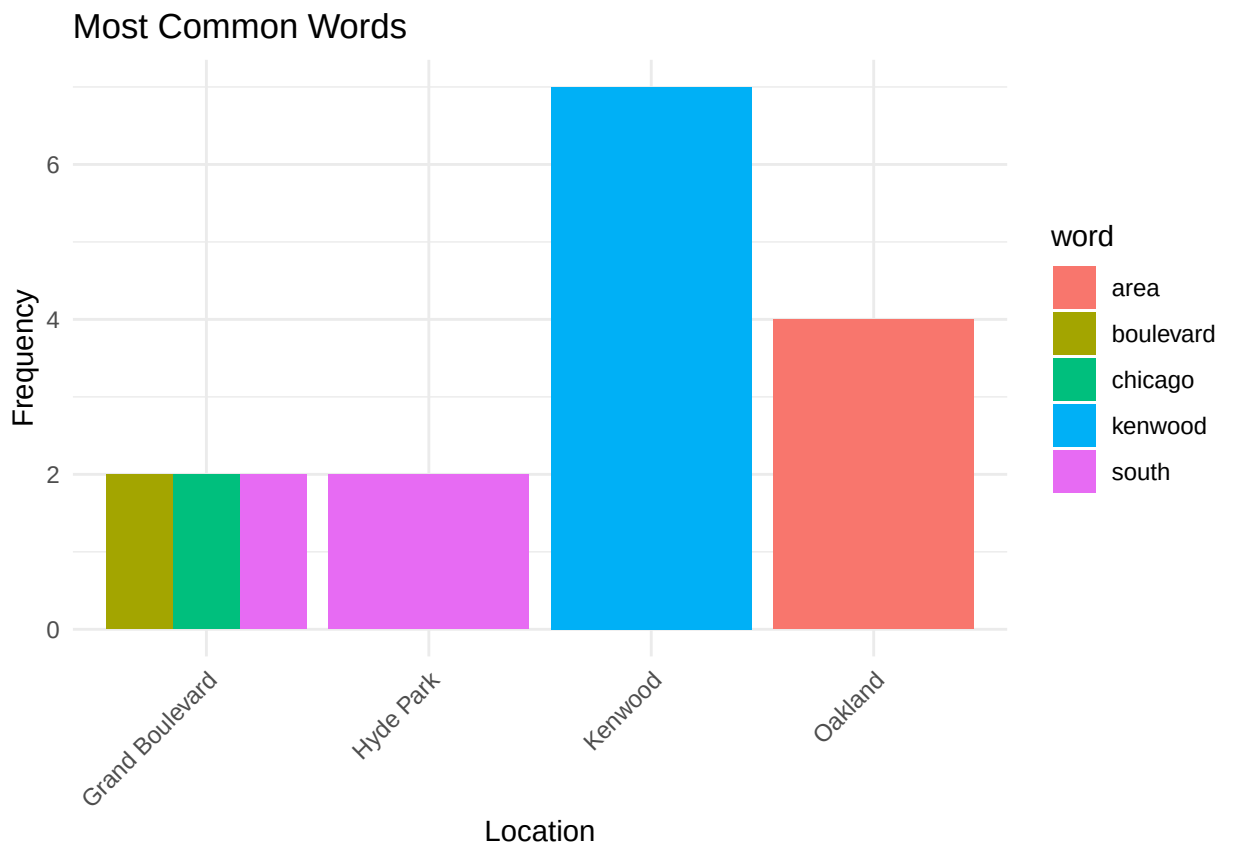
```
words %>%
  count(word, sort = TRUE)
```

```
## # A tibble: 120 x 2
##    word          n
##    <chr>     <int>
##  1 chicago       8
##  2 kenwood       7
##  3 area          6
##  4 community     6
##  5 south         6
##  6 lake          5
##  7 areas         4
##  8 one           4
##  9 park          4
## 10 shore         4
## # i 110 more rows
```

- "chicago" is the most common word used overall in east places.

Plot the most common words within each location. What are some of the similarities between the locations? What are some of the differences?

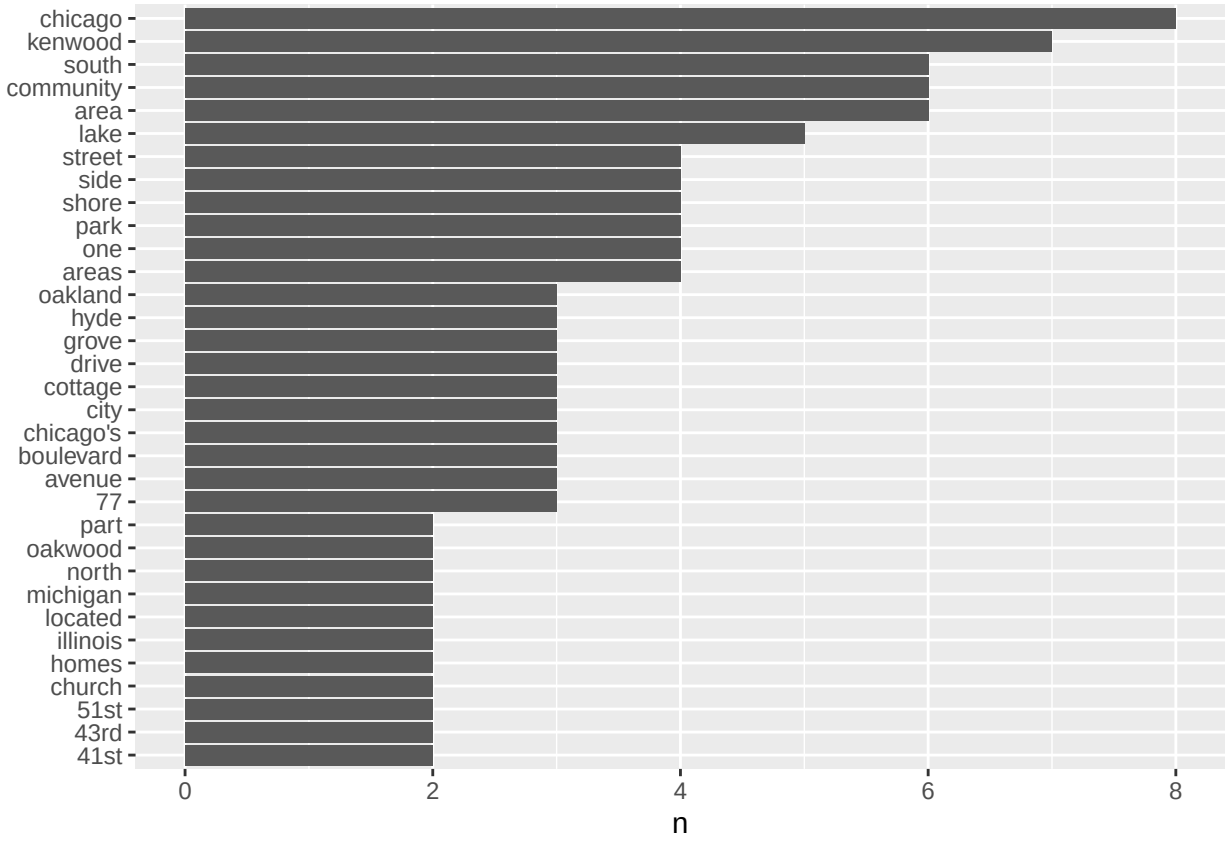
```r
words %>% table() %>% data.frame() %>%
  group_by(east_links.place) %>%
  filter(Freq == max(Freq)) %>%
  ggplot(aes(x = east_links.place, y = Freq, fill = word)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Most Common Words", x = "Location", y = "Frequency") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



- The word “kenwood” appears 7 times in Kenwood’s wiki, “area” appears 4 times in Oakland, and “south” appears 2 times in Hyde Park. However, there are 3 words appear 2 times in Grand Boulevard’s wiki and they are the most common words in this place.
- I found the similarities between each place are most of words are related to directions and places.

```r
library(ggplot2)

words %>%
  count(word, sort = TRUE) %>%
  filter(n > 1) %>%
  mutate(word = reorder(word, n)) %>%
  ggplot(aes(n, word)) +
  geom_col() +
  labs(y = NULL)
```


chicago
kenwood
south
community
area
lake
street
side
shore
park
one
areas
oakland
hyde
grove
drive
cottage
city
chicago's
boulevard
avenue
77
part
oakwood
north
michigan
located
illinois
homes
church
51st
43rd
41st
0
2
4
6
8
n