IMT 573: Problem Set 3 - Working With Data II

Xinyi Yang

Due: Tuesday, October 27, 2020

Collaborators:

Instructions: Before beginning this assignment, please ensure you have access to R and RStudio; this can be on your own personal computer or on the IMT 573 R Studio Server.

1. Download the `problemset3.Rmd` file from Canvas or save a copy to your local directory on RStudio Server. Open `problemset3.Rmd` in RStudio and supply your solutions to the assignment by editing `problemset3.Rmd`.
2. Replace the "Insert Your Name Here" text in the `author:` field with your own full name. Any collaborators must be listed on the top of your assignment.
3. All materials and resources that you use (with the exception of lecture slides) must be appropriately referenced within your assignment. In particular, note that Stack Overflow is licenses as Creative Commons (CC-BY-SA). This means you have to attribute any code you refer from SO.
4. Partial credit will be awarded for each question for which a serious attempt at finding an answer has been shown. But please **DO NOT** submit pages and pages of hard-to-read code and attempts that is impossible to grade. That is, avoid redundancy. Remember that one of the key goals of a data scientist is to produce coherent reports that others can easily follow. Students are *strongly* encouraged to attempt each question and to document their reasoning process even if they cannot find the correct answer. If you would like to include R code to show this process, but it does not run without errors you can do so with the `eval=FALSE` option as follows:

```
a + b # these object dont' exist
# if you run this on its own it with give an error
```

6. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit PDF`, rename the knitted PDF file to `ps3_ourLastName_YourFirstName.pdf`, and submit the PDF file on Canvas.
7. Collaboration is often fun and useful, but each student must turn in an individual write-up in their own words as well as code/work that is their own. Regardless of whether you work with others, what you turn in must be your own work; this includes code and interpretation of results. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.

Setup: In this problem set you will need, at minimum, the following R packages.

```
# Load standard libraries
library('dplyr')
library('tidyr')
library('censusr')
library('stringr')
```

[reference][https://github.com/yihui/knitr-examples/blob/master/077-wrap-output.Rmd]

```
library(knitr)
hook_output = knit_hooks$get('output')
knit_hooks$set(output = function(x, options) {
  # this hook is used only when the linewidth option is not NULL
  if (!is.null(n <- options$linewidth)) {
    x = knitr:::split_lines(x)
    # any lines wider than n should be wrapped
    if (any(nchar(x) > n)) x = strwrap(x, width = n)
    x = paste(x, collapse = '\n')
  }
  hook_output(x, options)
})
```

Problem 1: Joining Census Data to Police Reports In this problem set, we will be joining disparate sets of data - namely: Seattle police crime data, information on Seattle police beats, and education attainment from the US Census. Our goal is to build a dataset where we can examine questions around crimes in Seattle and the educational attainment of people living in the areas in which the crime occurred; this requires data to be combined from these two individual sources.

As a general rule, be sure to keep copies of the original dataset(s) as you work through cleaning (remember data provenance!).

(a) Importing and Inspecting Crime Data Load the Seattle crime data from the provided `crime_data.csv` data file. You can find more information on the data here: https://data.seattle.gov/Public-Safety/Crime-Data/4fs7-3vj5. This dataset is constantly refreshed online so we will be using the provided csv file for consistency. We will call this dataset the "Crime Dataset." Perform a basic inspection of the Crime Dataset and discuss what you find.

```
crime_data<-read.csv("crime_data.csv")
# make a copy of the original dataset
crime <- crime_data
#have a peek of the data
str(crime)
```

```
## 'data.frame':    523591 obs. of  11 variables:
##  $ Report.Number              : num  1.98e+12 1.98e+12 1.98e+12 1.98e+13 1.98e+12 ...
##  $ Occurred.Date              : chr  "12/16/1975" "01/01/1976" "01/28/1979" "08/22/1981" ...
##  $ Occurred.Time              : int  900 1 1600 2029 2000 155 2213 0 1130 NA ...
##  $ Reported.Date              : chr  "12/16/1975" "01/31/1976" "02/09/1979" "08/22/1981" ...
##  $ Reported.Time              : int  1500 2359 1430 2030 435 155 2213 844 1700 NA ...
##  $ Crime.Subcategory          : chr  "BURGLARY-RESIDENTIAL" "SEX OFFENSE-OTHER" "CAR PROWL" "HOMICIDE
##  $ Primary.Offense.Description: chr  "BURGLARY-FORCE-RES" "SEXOFF-INDECENT LIBERTIES" "THEFT-CARPROWL
##  $ Precinct                   : chr  "SOUTH" "UNKNOWN" "EAST" "SOUTH" ...
##  $ Sector                     : chr  "R" "" "G" "S" ...
##  $ Beat                       : chr  "R3" "" "G2" "S2" ...
##  $ Neighborhood               : chr  "LAKEWOOD/SEWARD PARK" "UNKNOWN" "CENTRAL AREA/SQUIRE PARK" "BRI
```

(b) Looking at Years That Crimes Were Committed Let's start by looking at the years in which crimes were committed. What is the earliest year in the dataset? Are there any distinct trends with the annual number of crimes committed in the dataset?

Subset the data to only include crimes that were committed after 2011 (remember good practices of data provenance!). Going forward, we will use this data subset.

```r
crime <- crime %>% separate(Occurred.Date,c("occur_day","occur_month","occur_year"))
```

```
## Warning: Expected 3 pieces. Missing pieces filled with `NA` in 2 rows [10, 123].
```

```r
sort(unique(crime$occur_year))
```

```
##  [1] "1908" "1964" "1973" "1974" "1975" "1976" "1977" "1978" "1979" "1980"
## [11] "1981" "1985" "1986" "1987" "1988" "1989" "1990" "1991" "1993" "1994"
## [21] "1995" "1996" "1997" "1998" "1999" "2000" "2001" "2002" "2003" "2004"
## [31] "2005" "2006" "2007" "2008" "2009" "2010" "2011" "2012" "2013" "2014"
## [41] "2015" "2016" "2017" "2018" "2019"
```

```r
paste("The earliest year in the dataset when crimes were committed in 1908.")
```

```
## [1] "The earliest year in the dataset when crimes were committed in 1908."
```

```r
# subset data to crimes committed after 2011
crime_new <- crime %>% filter(occur_year>2011)
```

(c) Looking at Frequency of Beats What is a Police Beat? How frequently are the beats in the Crime Dataset listed? Are there any anomolies with how frequently some of the beats are listed? Are there missing beats?

```r
# make the beats factors to easier summarise the frequencies
crime_new <- crime_new %>% mutate(beat_fct=as.factor(Beat))
length(unique(crime_new$beat_fct))
```

```
## [1] 60
```

```r
table(crime_new$beat_fct)
```

```
##
##          B1    B2    B3    C1    C2    C3   CTY    D1    D2    D3   DET    E1
##  2054  7954  9253  8846  5694  4789  4726     1  8066  7491  6530     7  7459
##    E2    E3    F1    F2    F3    G1    G2    G3    J1    J2    J3     K    K1
## 10200  7032  4332  6429  5361  3257  5259  4327  5668  6585  7203     1  6611
##    K2    K3    L1    L2    L3    M1    M2    M3     N    N1    N2    N3    O1
##  6560 11611  5823 10049  5710  9883 10210  9723     1  5303  7409  7517  4523
##    O2    O3    Q1    Q2    Q3    R1    R2    R3     S    S1    S2    S3    SS
##  2894  3239  5647  8159  9249  6080  7448  6909     4  4819  5139  6027     1
##    U1    U2    U3     W    W1    W2    W3    WS
## 10157  8866  9019     3  5135  6514  5286     1
```

According to the description listed on the Seattle government webpage, a Police Beat represents designated police sector boundary where offense(s) occurred. The frequencies of the beats are listed in the table above. There are 2054 missing beats. According to the definition of beats[beats][https://www.seattle.gov/police/information-and-data/tweets-by-beat], there should be 51 sectors, but we have 60 unique beats here. There are many sectors in the table that only have several reports compared to other sectors that have thousands of reports. I suppose these abnormal sectors were input mistakenly.

(d) Importing Police Beat Data and Filtering on Frequency Load the data on Seattle police beats provided in `police_beat_and_precinct_centerpoints.csv`. You can find additional information on the data here: (https://data.seattle.gov/Land-Base/Police-Beat-and-Precinct-Centerpoints/4khs-fz35). We will call this dataset the "Beats Dataset."

Does the Crime Dataset include police beats that are not present in the Beats Dataset? If so, how many and with what frequency do they occur? Would you say that these comprise a large number of the observations

in the Crime Dataset or are they rather infrequent? Do you think removing them would drastically alter the scope of the Crime Dataset?

Let's remove all instances in the Crime Dataset that have beats which occur fewer than 10 times across the Crime Dataset. Also remove any observations with missing beats. After only keeping years of interest and filtering based on frequency of the beat, how many observations do we now have in the Crime Dataset?

```
# load the beats dataset
beats_data <- read.csv('police_beat_and_precinct_centerpoints.csv')

# find police beats in crime dataset that don't have a match in the Beats dataset
# anti join two dataset with different column names
diff <- anti_join(crime_new,beats_data, by=c("Beat"="Name"))
# count the frequencies
diff %>% count(Beat,sort = TRUE)
```

```
##   Beat    n
## 1      2054
## 2  DET    7
## 3    S    4
## 4  CTY    1
## 5    K    1
## 6   SS    1
## 7   WS    1
```

The missing value occupies a large number of the observations in the crime dataset. The rest of them are rather infrequent. And it won't alter the scope of the crime datasset to remove these entries.

```
# remove na and blanks, reference: https://stackoverflow.com/questions/9126840/delete-rows-with-blank-v
crime_new_clean <- crime_new[!(is.na(crime_new$Beat) | crime_new$Beat==""), ]

# reference: https://r.789695.n4.nabble.com/Delete-observations-with-a-frequency-lt-x-td3081226.html
crime_new_clean <- crime_new_clean [!table(crime_new_clean$beat_fct)[crime_new_clean$beat_fct] < 10,]

# check observations
dim(crime_new_clean)
```

```
## [1] 347980     14
```

```
paste(" After only keeping years of interest and filtering based on frequency of the beat, we have", nr
```

```
## [1] " After only keeping years of interest and filtering based on frequency of the beat, we have 347
```

(e) Importing and Inspecting Police Beat Data To join the Beat Dataset to census data, we must have census tract information. Use the `censusr` package to extract the 15-digit census tract for each police beat using the corresponding latitude and longitude. Do this using each of the police beats listed in the Beats Dataset. Do not use a for-loop for this but instead rely on R functions (e.g. the 'apply' family of functions). Add a column to the Beat Dataset that contains the 15-digit census tract for the each beat. (HINT: you may find `censusr`'s `call_geolocator_latlon` function useful)

We will eventually join the Beats Dataset to the Crime Dataset. We could have joined the two and then found the census tracts for each beat. Would there have been a particular advantage/disadvantage to doing this join first and then finding census tracts? If so, what is it? (NOTE: you do not need to write any code to answer this)

```
# use call_geolocator_latlon function and apply along axis function to extract 15-digit census tract
beats_data <- beats_data %>%
  mutate(geo_code = apply(beats_data, 1,function(x) call_geolocator_latlon(x[3],x[4])))
```

I don't think it's a good choice to join first and then find census tracts. Because the crime dataset is large and it takes longer runtime. Even there are more duplicate data to process in the joined dataset than the beats data only.

(f) Extracting FIPS Codes Once we have the 15-digit census codes, we will break down the code based on information of interest. You can find more information on what these 15 digits represent here: https://transition.fcc.gov/form477/Geo/more_about_census_blocks.pdf.

First, create a column that contains the state code for each beat in the Beats Dataset. Then create a column that contains the county code for each beat. Find the FIPS codes for WA State and King County (the county of Seattle) online. Are the extracted state and county codes what you would expect them to be? Why or why not?

```
# extract state code and county code
beats_data <- beats_data %>%
  mutate(state = str_sub(geo_code,1,2),
         county = str_sub(geo_code,3,5))
```

RESPONSE: I found that WA state code is 53, and code of King county is 033. They are the same with the extracted state and county codes. So I think the resources matched with each other and there's no error so far.

(g) Extracting 11-digit Codes The census data uses an 11-digit code that consists of the state, county, and tract code. It does not include the block code. To join the census data to the Beats Dataset, we must have this code for each of the beats. Extract the 11-digit code for each of the beats in the Beats Dataset. The 11 digits consist of the 2 state digits, 3 county digits, and 6 tract digits. Add a column with the 11-digit code for each beat.

```
# extract the 11-digit code
beats_data <- beats_data %>%
  mutate(eleven_digit = str_sub(geo_code,1,11))
```

(h) Extracting 11-digit Codes From Census Now, we will examine census data provided om `census_edu_data.csv`. The data includes counts of education attainment across different census tracts. Note how this data is in a 'wide' format and how it can be converted to a 'long' format. For now, we will work with it as is.

The census data contains a `GEO.id` column. Among other things, this variable encodes the 11-digit code that we had extracted above for each of the police beats. Specifically, when we look at the characters after the characters "US" for values of `GEO.id`, we see encodings for state, county, and tract, which should align with the beats we had above. Extract the 11-digit code from the `GEO.id` column. Add a column to the census data with the 11-digit code for each census observation.

```
# load census data
census <- read.csv('census_edu_data.csv')
# make a copy of the original dataset
census_edu_data <- census
# extract the 11-digit code
census <- census %>% mutate(eleven_digit = str_sub(GEO.id,10))
```

(i) Join Datasets Join the census data with the Beat Dataset using the 11-digit codes as keys. Be sure that you do not lose any of the police beats when doing this join (i.e. your output dataframe should have the same number of rows as the cleaned Beats Dataset - use the correct join). Are there any police beats that do not have any associated census data? If so, how many?

```r
# join census data and beats
beats_census <- beats_data %>%
  left_join(census)
```

```
## Joining, by = "eleven_digit"
```

```r
# check the dimension of the new dataset
dim(beats_census)
```

```
## [1] 57 36
```

```r
# check if there is any data not associated
sum(is.na(beats_census$GEO.id))
```

```
## [1] 0
```

RESPONSE: All police beats have associated census data.

Then, join the Crime Dataset to our joined beat/census data. We can do this using the police beat name. Again, be sure you do not lose any observations from the Crime Dataset. What is the final dimensions of the joined dataset?

Once everything is joined, save the final dataset for future use.

```r
# join the crime dataset and census_beats dataset
crime_beats <- crime_new_clean %>%
  left_join(beats_census,by=c("Beat"="Name"))

# check the size of the new dataset
dim(crime_beats)
```

```
## [1] 347980     49
```

```r
# save the final dataset

filename <- file.path("C:\\Users\\yang_\\OneDrive - UW\\573\\problem sets", "crime_beats_census.csv")
write.csv(x=crime_beats,file = filename)
```