IMT 573: Problem Set 3 - Working With Data II

Miloni Desai

Due: Tuesday, October 22, 2019

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. Be sure to include well-documented (e.g. commented) code chucks, figures, and clearly written text chunk explanations as necessary. Any figures should be clearly labeled and appropriately referenced within the text. Be sure that each visualization adds value to your written explanation; avoid redundancy – you do not need four different visualizations of the same pattern.
4. Collaboration on problem sets is fun and useful, and we encourage it, 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.
5. All materials and resources that you use (with the exception of lecture slides) must be appropriately referenced within your assignment.
6. Remember partial credit will be awarded for each question for which a serious attempt at finding an answer has been shown. 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. (Note: I am also using the `include=FALSE` option here to not include this code in the PDF, but you need to remove this or change it to `TRUE` if you want to include the code chunk.)
7. 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_YourLastName_YourFirstName.pdf`, and submit the PDF file on Canvas.

Setup:

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

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

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.

```
#Import Crime Dataset
crime_data <- read_csv("imt573/Data/crime_data.csv")
```

```
## Parsed with column specification:
## cols(
##   `Report Number` = col_double(),
##   `Occurred Date` = col_character(),
##   `Occurred Time` = col_double(),
##   `Reported Date` = col_character(),
##   `Reported Time` = col_double(),
##   `Crime Subcategory` = col_character(),
##   `Primary Offense Description` = col_character(),
##   Precinct = col_character(),
##   Sector = col_character(),
##   Beat = col_character(),
##   Neighborhood = col_character()
## )
```

```
#View(crime_data)
```

```
#Inspecting the dataset
names(crime_data)
```

```
##  [1] "Report Number"               "Occurred Date"
##  [3] "Occurred Time"               "Reported Date"
##  [5] "Reported Time"               "Crime Subcategory"
##  [7] "Primary Offense Description" "Precinct"
##  [9] "Sector"                      "Beat"
## [11] "Neighborhood"
```

```
summary (crime_data)
```

```
##  Report Number       Occurred Date      Occurred Time  Reported Date
##  Min.   :2.008e+08   Length:523591      Min.   :   0   Length:523591
##  1st Qu.:2.008e+13   Class :character   1st Qu.: 900   Class :character
##  Median :2.012e+13   Mode  :character   Median :1500   Mode  :character
##  Mean   :1.635e+13                      Mean   :1359
##  3rd Qu.:2.016e+13                      3rd Qu.:1920
##  Max.   :2.019e+13                      Max.   :2359
##                                         NA's   :2
```

```
##  Reported Time  Crime Subcategory  Primary Offense Description
##  Min.   :   0   Length:523591      Length:523591
##  1st Qu.: 950   Class :character   Class :character
##  Median :1407   Mode  :character   Mode  :character
##  Mean   :1353
##  3rd Qu.:1817
##  Max.   :2359
##  NA's   :2
##    Precinct            Sector              Beat
##  Length:523591      Length:523591      Length:523591
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  Neighborhood
##  Length:523591
##  Class :character
##  Mode  :character
##
##
##
##
```

```
dim(crime_data)
```

```
## [1] 523591     11
```

```
str(crime_data)
```

```
## Classes 'spec_tbl_df', 'tbl_df', 'tbl' and '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              : num  900 1 1600 2029 2000 ...
##  $ Reported Date              : chr  "12/16/1975" "01/31/1976" "02/09/1979" "08/22/1981" ...
##  $ Reported Time              : num  1500 2359 1430 2030 435 ...
##  $ 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" NA "G" "S" ...
##  $ Beat                       : chr  "R3" NA "G2" "S2" ...
##  $ Neighborhood               : chr  "LAKEWOOD/SEWARD PARK" "UNKNOWN" "CENTRAL AREA/SQUIRE PARK" "BRI
##  - attr(*, "spec")=
##   .. cols(
##   ..   `Report Number` = col_double(),
##   ..   `Occurred Date` = col_character(),
##   ..   `Occurred Time` = col_double(),
##   ..   `Reported Date` = col_character(),
##   ..   `Reported Time` = col_double(),
##   ..   `Crime Subcategory` = col_character(),
##   ..   `Primary Offense Description` = col_character(),
##   ..   Precinct = col_character(),
##   ..   Sector = col_character(),
##   ..   Beat = col_character(),
```

```
##   ..   Neighborhood = col_character()
##   .. )
```

#The data has reports that have the time and the date of the #crime incident, a short description of the crime, #the neighbourhood and other location details.‘ #The data set has 523591 cases reported

(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?

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

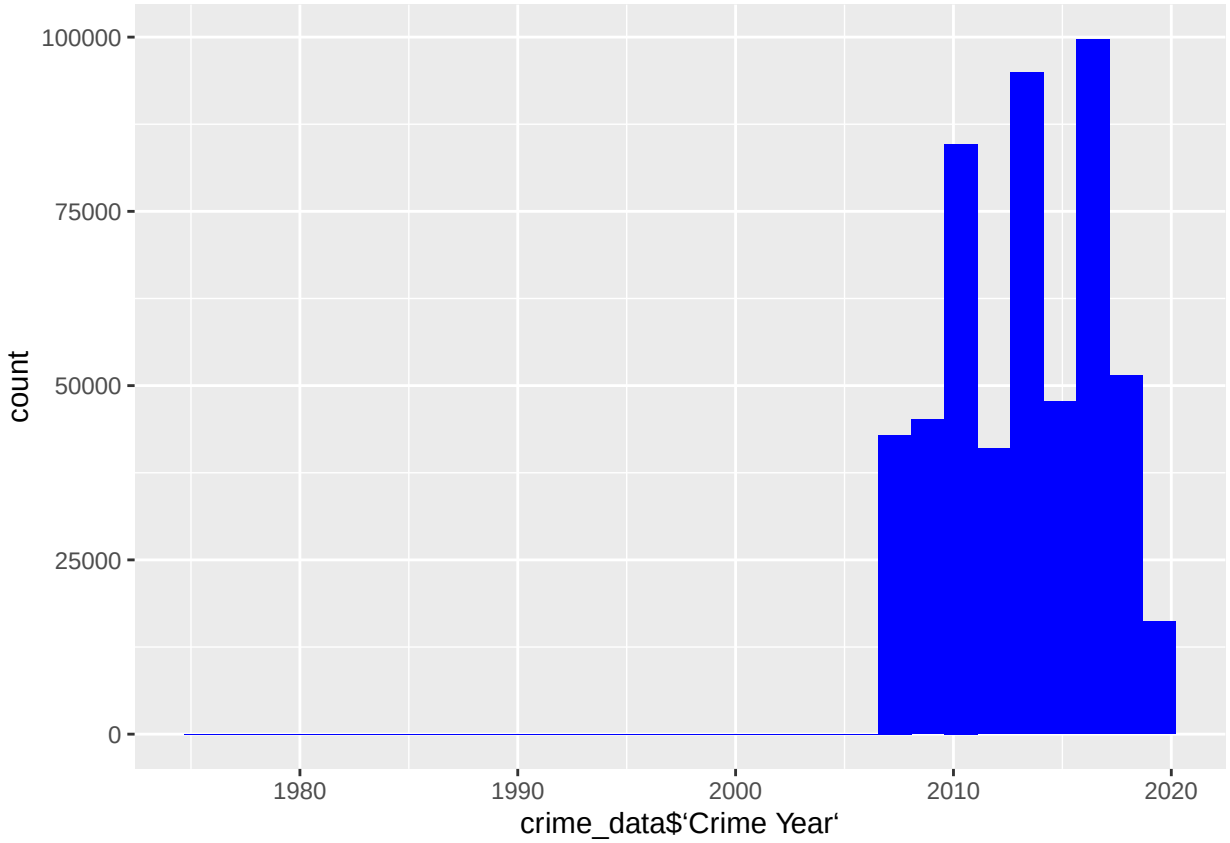
```
x<-year(as.Date(crime_data$`Reported Date`, format = "%m/%d/%Y"))
crime_data["Crime Year"] <- x
min(crime_data$`Crime Year`)
```

```
## [1] 1975
```

```
max(crime_data$`Crime Year`)
```

```
## [1] 2019
```

```
ggplot(data = crime_data, aes(crime_data$`Crime Year`)) +
geom_histogram(bins = 30,fill="blue")
```



#The earliest crime was committed in 1975 #Most crimes are reportedly between 2008-2019

```
x<-crime_data %>%
  group_by(crime_data$‘Crime Year‘) %>%
  summarize(n())
```

#With each year the crimes seem to be increasing #with 2018 having the highest no of crimes

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.

```
df <- crime_data[crime_data$‘Crime Year‘ > 2011, ]
#Hence we now have a new data frame that has only crimes
#committed after 2011 (aka starting 2012)
dim(df)
```

```
## [1] 350828     12
```

(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?

```
x1<-df %>%
  group_by(df$Beat) %>%
  summarize(n())
```

#Police Beat is the time and area that a police patrols #Beats are listed for most observations in the dataset, only #a few have null values. #However there are missing beats in the dataset, where for some #the

frequency is low (indicating missing values), in the table #x1 we can see that there are over 2000 missing values for Beats

(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."

```
Beats_Dataset <- read_csv("imt573/Data/police_beat_and_precinct_centerpoints.csv")
```

```
## Parsed with column specification:
## cols(
##   Name = col_character(),
##   `Location 1` = col_character(),
##   Latitude = col_double(),
##   Longitude = col_double()
## )
```

```
#View(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?

```
common <- intersect(df$Beat,Beats_Dataset$Name)
y<-sort(common)
#View(y)
```

#As we can see that Crime Dataset has 59 Beats, but only 53 are #common with the Beats Dataset.We know that there are 6 police beats not present #in the Beats Dataset. #The 6 missing police beats are #CTY, DET , K , S, SS, WS. #The freqency is of these are CTY (1), DET(7), K(1),S(4),SS(1) #and WS(1).

Would you say that these comprise a large number of the observations in the Crime Dataset or are they rather infrequent? #They are rather infrequent

Do you think removing them would drastically alter the scope of the Crime Dataset? #There are more missing values than these police beats. Removing #them will not alter the scope of this dataset at all, as in comparison #to the number of observations, this is less than a percent of it.

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?

```
df%>%
      group_by("Beat") %>%
      filter(n() > 10)
```

```
## # A tibble: 350,828 x 13
## # Groups:   "Beat" [1]
##    `Report Number` `Occurred Date` `Occurred Time` `Reported Date`
##              <dbl> <chr>                     <dbl> <chr>
##  1  20120000100012 04/02/2012                 2040 04/03/2012
##  2  20120000100035 04/02/2012                 2100 04/02/2012
##  3  20120000100055 04/02/2012                 1930 04/02/2012
##  4  20120000100062 04/02/2012                 2144 04/02/2012
##  5  20120000100092 04/02/2012                 2218 04/02/2012
##  6  20120000100102 04/02/2012                 2229 04/02/2012
##  7  20120000100105 04/02/2012                 2230 04/02/2012
```

```
##  8  20120000100109 04/02/2012                2015 04/02/2012
##  9  20120000100120 04/02/2012                2256 04/02/2012
## 10  20120000100125 04/02/2012                2255 04/02/2012
## # ... with 350,818 more rows, and 9 more variables: 'Reported Time' <dbl>,
## #   'Crime Subcategory' <chr>, 'Primary Offense Description' <chr>,
## #   Precinct <chr>, Sector <chr>, Beat <chr>, Neighborhood <chr>, 'Crime
## #   Year' <dbl>, '"Beat"' <chr>
```

```
df_new <- na.omit(df)
#View(df_new)
```

#Based on the new filtering criterias we get a new #dataframe called df_new which contains all the rows #having no null values, years after 2011 and police beats #frequency greater than 10. #This dataframe has a total of 348,684 enteries.

(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)

```
Beats_Dataset['Census_Tract']<- 0

for( i in 1:57) {
n  <- call_geolocator_latlon(Beats_Dataset$Latitude[i],Beats_Dataset$Longitude[i])
Beats_Dataset$Census_Tract[i]<- n
}
#View(Beats_Dataset)
```

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) #Had we firdt joined the two datasets and then found the census tract, #while the advantage of that would have been time, the disadvantage of the #same would have been that we would be doing extra computation for data #that is not needed. Thus this is the more efficient way to do the same.

(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?

```
library(tidyr)
new_df<-extract(Beats_Dataset, Census_Tract, into = c("State", "County","Tracts","Blocks"), "(.{2})(.{3
#View(new_df)
```

#The code is as expected, both for WA and Seattle.

(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.

```
Beats_Dataset_Final<-extract(Beats_Dataset, Census_Tract, into = c("11_digit_code"),  "(.{11})", remove=
#View(Beats_Dataset_Final)
```

(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.

```
census_data <- read_csv("imt573/Data/census_edu_data.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   GEO.id = col_character(),
##   `GEO.display-label` = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
#View(census_data)
```

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. #That column is already in the dataset as GEO.id2

(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?

```
Beats_Census<- merge(Beats_Dataset_Final, census_data, by.x = "11_digit_code",by.y = "GEO.id2")
#View(Beats_Census)
#They all have census data as there are 57 records in the new dataset.
```

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?

```
Crime_Beats_Census<- merge(crime_data, Beats_Census, by.x = "Beat",by.y = "Name")
dim(Crime_Beats_Census)
```

```
## [1] 520261     44
```

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