Crime on Women

Joana Lame

2024-11-20

```
{r setup, include=FALSE} knitr::opts_chunk$set(echo = TRUE)
```

A prominent perception around crime in LA City is that crime on women does not affect specific cohorts; on the contrary, it crosses geographic and income boundaries. To confirm whether crime on women is indeed geographically uniform in LA City, we compare crime rates against women across the 21 LA areas.

Because there is a considerable number of areas, to be able to view and derive immediate information from the data at a glance, the data will be displayed in maps with color gradients.

As a baseline for comparison, the distribution across areas of the crime rate against the entire population will be used.

Install the necessary packages for data treatment and plotting:

```
install.packages("readr")
install.packages("ggplot2")
install.packages("sf")
install.packages("dplyr")
install.packages("stringr")
library(readr)
library(ggplot2)
library(sf)
library(dplyr)
library(stringr)
```

Import the LA City crime data:

```
Crime_Data_from_2020_to_Present <- read_csv("Crime_Data_from_2020_to_Present.csv")
```

Understand what the area divisions are:

```
Areas <- Crime_Data_from_2020_to_Present %>% distinct(AREA, ‘AREA NAME‘)
```

Download and read an LA City shapefile with the same divisions as our main data:

```
shapefile_path <- "C:/Users/joana/OneDrive/Desktop/LAPD_Division_1980236667069515482/LAPD_Divisions.shp
```

Ensure the file exists and is not in zip form:

```
file.exists(shapefile_path)
st_drivers()

shapefile <- st_read(shapefile_path)
```

Create a dataframe with area code, area name and sum of total crimes per area. Total crime includes all crime types, both violent and nonviolent.

```
total_crimes <- Crime_Data_from_2020_to_Present %>%
  count(AREA, name = "Total_crime") %>%  #derive total crimes from original data
  right_join(Areas, by = c("AREA" = "AREA"))
```

Note: In case of future use of the data, besides having an understanding of what the LA City areas are, there is no need to create the dataframe Areas. total_crimes can be directly created using only CRime_Data_from_2020_to_Present

```
total_crimes
```

Join the crime data with the shapefile (for the purpose of mapping crime, this is not necessary, but might be needed for ease of use in future research):

```
LA_crime_data <- shapefile %>%
  left_join(total_crimes, by = c("APREC" = "AREA NAME"))
```

Wrap the labels (referring to the names of the areas) at 10 characters to fit within the respective area on the map:

```
shapefile$label_wrapped <- str_wrap(shapefile$APREC, width = 10)
```

Compute centroids of the divisions (polygons). This will be useful to locate the labels of area names at the center of each area on the map:

```
shapefile_centroids <- st_centroid(shapefile)
```

Add centroid coordinates to the shapefile data:

```
shapefile$centroid_x <- st_coordinates(shapefile_centroids)[, 1]
shapefile$centroid_y <- st_coordinates(shapefile_centroids)[, 2]
```

Plot the map of “Total Crimes”:

```
ggplot(data = shapefile) +
  geom_sf(aes(fill = total_crimes$Total_crime), color = "white", lwd = 0.1) +
  # borders between areas are white
  geom_sf_text(aes(x = centroid_x, y = centroid_y, label = label_wrapped),
             size = 1.5,
             color = "black") +    # color of labels (area name) is black
  scale_fill_gradient(low = "lightyellow", high = "purple", name = "Crime Rate")
  + theme_minimal() +
  theme(
    legend.position = c(0, 0),  # position the legend at the bottom left
    legend.justification = c(-0.2, -0.2), #justify the legend to the bottom left
    legend.title = element_text(size = 8),  # smaller legend title than default
    legend.text = element_text(size = 6),   # smaller legend text than default
    legend.key.size = unit(0.3, "cm"),      # smaller legend keys than default
    legend.key.height = unit(0.3, "cm"),    # adjust height of the legend key
    axis.title = element_blank(),     # remove axis titles to avoid overcrowding
    axis.text = element_blank(),           # remove axis text
    axis.ticks = element_blank(),          # remove axis ticks
    plot.margin = margin(0, 0, 0, 0)      #adjust margins for more space
    panel.grid = element_blank()           # remove gridlines
  ) +
  coord_sf(expand = FALSE)  # remove unnecessary padding around the map
```

After creating the map for our baseline (total crime across LA City areas), let's focus solely on crime against women, and map the total number of crimes against women per area.

Select only crimes against females:

```
female_victims <- Crime_Data_from_2020_to_Present %>%
  filter(Crime_Data_from_2020_to_Present$‘Vict Sex‘ == "F")
```

For future analysis of the same data, a similar command to that of deriving total_crime can be used. The following code is another alternative:

```
area_sums_f <- numeric(21)  #Create an empty vector to store the results
area_codes_f <- sprintf("%02d", 1:21) #Create a vector for area names.

for (i in 1:21) {   # loop through each area code from "01" to "21"
  area_code <- sprintf("%02d", i)  # area code as "01", "02", ..., "21"

  area_sums_f[i] <- sum(female_victims$AREA == area_code)
}   # calculate sum of the rows where AREA matches the current area code

area_df_f <- data.frame(Area = area_codes_f, Crime_on_women = area_sums_f)

crimes_against_women <- merge(Areas, area_df_f, by.x = "AREA", by.y = "Area",
all.x = TRUE)
```

Create the map with a color gradient for crime rate:

```
ggplot(data = shapefile) +
  geom_sf(aes(fill = crimes_against_women$Crime_on_women), color = "white",
  lwd = 0.1) +
  geom_sf_text(aes(x = centroid_x, y = centroid_y, label = label_wrapped),
               size = 1.5,
               color = "black") +
  scale_fill_gradient(low = "lightyellow", high = "purple", name = "Crime Rate")
  + theme_minimal() +
  theme(
    legend.position = c(0, 0),  # position the legend at the bottom left
    legend.justification = c(-0.2, -0.2),  # move the legend
    legend.title = element_text(size = 8),  # smaller legend title
    legend.text = element_text(size = 6),   # smaller legend text
    legend.key.size = unit(0.3, "cm"),      # smaller legend keys
    legend.key.height = unit(0.3, "cm"),    # adjust height of the legend key
    axis.title = element_blank(),           # remove axis titles
    axis.text = element_blank(),           # remove axis text
    axis.ticks = element_blank(),          # remove axis ticks
    plot.margin = margin(0, 0, 0, 0),      # adjust margins for more space
    panel.grid = element_blank()           # remove gridlines
  ) +
  coord_sf(expand = FALSE)  # remove unnecessary padding around the map
```

Seeing the number of crimes on the entire LA population and on women is useful in drawing comparisons, but it does not give the entire picture because the number of women inhabitants is different from the number of the entire population. For a more standardized comparison, let’s also look at what percentage of total victims are women in each LA City area.

Create a dataframe with number of total crimes, number of crimes on women, and percentage of crime that is on women:

```
LA_crime <- crimes_against_women %>%
  full_join(total_crimes) %>%
  mutate(’Female Victims (%)’ = round(100*Crime_on_women/Total_crime, 2))
        # add new column derived from a simple mathematical operation
        #  between two existing columns
```

Join this new crime data with the shapefile (not necessary for now, but if in the future data needs to be transferred and researcher would like to have all information on one file):

```
LA_crime_data <- shapefile %>%
  left_join(LA_crime, by = c("APREC" = "AREA NAME"))
```

Create the map with a color gradient for crime rate:

```
ggplot(data = shapefile) +
  geom_sf(aes(fill = LA_crime$‘Female Victims (%)‘), color = "white", lwd = 0.1)
  +
  geom_sf_text(aes(x = centroid_x, y = centroid_y, label = label_wrapped),
               size = 1.5,
               color = "black") +
  scale_fill_gradient(low = "lightyellow", high = "purple",
  name = "Female Victims (%)") +
  theme_minimal() +
  theme(
    legend.position = c(0, 0),  # position the legend at the bottom left
    legend.justification = c(-0.2, -0.1),  # move the legend as desired
    legend.title = element_text(size = 8),  # smaller legend title
    legend.text = element_text(size = 6),   # smaller legend text
    legend.key.size = unit(0.3, "cm"),      # smaller legend keys
    legend.key.height = unit(0.3, "cm"),    # adjust height of the legend key
    axis.title = element_blank(),           # remove axis titles
    axis.text = element_blank(),           # remove axis text
    axis.ticks = element_blank(),          # remove axis ticks
    plot.margin = margin(0, 0, 0, 0),      # adjust margins for more space
    panel.grid = element_blank()           # remove gridlines
  ) +
  coord_sf(expand = FALSE)  # remove unnecessary padding around the map
```