Extracting Data from Text and Geocoding: Office-involved Shootings (OIS) in Philadelphia

Pauline I. Alvarado

Fall 2018

This coding exercise was done in University of Pennsylvania's Criminal Justice Data Science Course taught by Dr. Gregory Ridgeway. Code has been modified from course examples to fit personal learning style. Data consists of PPD Service Area Shapefiles and information scraped from the PPD Officer-involved Shooting Website.

Key skills: Web and document scraping, regular expressions, OpenStreetMap and ArcGIS geocoders, JSON, simple features, stack/match/merge data

Scraping OIS Data from PPD Webpage

Load packages and grab raw html

```
# Packages
  library(lubridate)
  library(tm)
  library(pdftools)
  library(jsonlite)
  library(ggmap)
  library(sf)

# Grab raw html
  ppd_website <- scan("http://phillypolice.com/ois", what = "", sep = "\n")
  ppd_website <- gsub("\t", "", ppd_website)
  head(ppd_website)
```

```
## [1] "<!DOCTYPE html>"
## [2] "<html>"
## [3] "<head>"
## [4] "<title>Officer Involved Shootings | Philadelphia Police Department</title>"
## [5] "<meta name=\"viewport\" content=\"width=1020px, initial-scale=0.5\">"
## [6] "<meta http-equiv=\"Content-Type\" content=\"text/html;charset=utf-8\">"
```

Create dataframe

Grab 2013-2018 data

Make sure to view a sample row to determine how to clean up the data. Each table row starts with <tr-id="yyyy-####"]. The next row is the URL that contains PDF with detailed information. Date and address are in the 3rd and 4th row. respectively.

```
  a <- ppd_website
  i <- grep("id=\"2018-2954", a)
  a[i + 0:9]
```

```
##  [1] "<tr id=\"2018-2954\">"
##  [2] "<td><a href=\"/assets/crime-maps-stats/officer-involved-shootings/18-01.pdf\" class=\"fancybox\
##  [3] "<td>01/13/2018</td> "
##  [4] "<td>2800 Block of Kensington Avenue</td> "
##  [5] "<td>Wounded</td> "
##  [6] "<td>Yes</td> "
##  [7] "<td>No</td> "
##  [8] "<td>Pending</td> "
##  [9] "<td>Pending</td> "
## [10] "</tr>"
```

Extract, clean, and add 2013-2018 data to ois dataframe

```
  i <- grep("id=\"201(3|4|5|6|7|8)", a)

  ois <- data.frame(id = gsub("<[^>]*>", "", a[i+1]),
                    date = gsub("<[^>]*>", "", a[i+2]),
                    location = gsub("<[^>]*>", "", a[i+3]),
                    url = gsub("<td><a href=\"(.*)\" class=.*", "\\1", a[i+1]),
                    stringsAsFactors = FALSE)
```

Extract, clean, and add 2007-2012 data to OIS dataframe

Table for these years provide no incident date. Incident location is 2 elements after the

tag.

```
  i <- grep("id=\"20(07|08|09|10|11|12)", a)

  temp <- data.frame(id = gsub("<[^>]*>", "", a[i+1]),
                     date = NA,
                     location = gsub("<[^>]*>", "", a[i+2]),
                     url = gsub("<td><a href=\"(.*)\" class=.*", "\\1", a[i+1]),
                     stringsAsFactors = FALSE)
```

Stack "ois" and "temp" data

```
  ois <- rbind(ois, temp)
  ois$id <- gsub(" ", "", ois$id) # clear out extra spaces
  ois$url <- paste0("http://www.phillypolice.com", ois$url) # add full URL
  ois[1:5,]
```

```
##      id        date                          location
## 1 18-01 01/13/2018  2800 Block of Kensington Avenue
## 2 18-02 01/29/2018      1300 Block of Bigler Street
## 3 18-08 04/18/2018     3100 block of N. 33rd Street
## 4 18-12 06/08/2018     1400 block of Lardner Street
## 5 18-16 08/06/2018        4800 block of Knox Street
##                                                                                                  url
## 1    http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/18-01.pdf
## 2 http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/OIS18-02.pdf
## 3 http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/OIS18-08.pdf
## 4    http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/18-12.pdf
## 5    http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/18-16.pdf
```

Consistent "id" variable format

```
  # View rows that do not follow the ##-## convention
    grep("^[^-]*$", ois$id, value = TRUE)
```

```
## [1] "1822" "1730" "1630" ""     ""     ""
```

```
  # Add hypens
    ois$id[ois$id == "1630"] <- "16-30"
    ois$id[ois$id == "1730"] <- "17-30"
    ois$id[ois$id == "1822"] <- "18-22"
```

View rows with missing id

```
  subset(ois, id == "")
```

```
##     id        date                       location
## 81     04/22/2014  5100 block of Willows Avenue
## 279           <NA>           "B" and Ontario St
## 286           <NA>            "A" and Louden St
##                                    url
## 81  http://www.phillypolice.com/assets/
## 279 http://www.phillypolice.com/assets/
## 286 http://www.phillypolice.com/assets/
```

Willows Avenue OIS

```
  # View rows above and below
    i <- grep("5100 block of Willows Avenue", ois$location)
    ois[(i-1):(i+1),]
```

```
##       id        date                        location
## 80 14-12 03/25/2014  100 block of W. Louden Street
## 81       04/22/2014   5100 block of Willows Avenue
## 82 14-16 04/26/2014     5400 block of Media Street
##                                                                                  url
## 80 http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/14-12.pdf
## 81                                                 http://www.phillypolice.com/assets/
## 82 http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/14-16.pdf
```

```
  # Check if PDFs are available for 14-13, 14-14, 14-15
    a <- try(scan("http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/14-13
```

```
## Error in file(file, "r") :
##   cannot open the connection to 'http://www.phillypolice.com/assets/crime-maps-stats/officer-involve
```

```
    a <- try(scan("http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/14-14
```

```
## Error in file(file, "r") :
##   cannot open the connection to 'http://www.phillypolice.com/assets/crime-maps-stats/officer-involve
```

```
    a <- try(scan("http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/14-15
```

```
## Error in scan("http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/14-15.p
##   scan() expected 'a real', got '%PDF-1.3'
```

```
  # Add 14-15 and URL to Willows Avenue OIS
    ois$id[1+which(ois$id=="14-12")] <- "14-15"
    ois$url[ois$id=="14-15"] <- "http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-sh
```

View "B St and Ontario St" and "A St and Louden St"

```
  # View rows above and below
    i <- which(ois$id == "")
    ois[c((i[1]-1):(i[1]+1), (i[2]-1):(i[2]+1)), 1:3]
```

```
##        id date                    location
## 278 10-60 <NA> 1200 block of W. Harold St
## 279       <NA>         "B" and Ontario St
## 280 10-67 <NA>    5200 block of Marlow St
## 285 10-74 <NA>           3300 Brighton St
## 286       <NA>          "A" and Louden St
## 287 10-78 <NA>   2900 block of Island Ave
```

```
  # After checking possible PDFs, add missing ids, remove smart quotes, and fix URL
    ois$id[1+which(ois$id=="10-60")] <- "10-65"
    ois$location[ois$id=="10-65"] <- "B and Ontario St"
    ois$url[ois$id=="10-65"] <- "http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-sh

    ois$id[1+which(ois$id=="10-74")] <- "10-76"
    ois$location[ois$id=="10-76"] <- "A and Louden St"
    ois$url[ois$id=="10-76"] <- "http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-sh
```

Clean up characters or extra spaces

```
  # location column
    ois$location <- gsub("&quot;", "", ois$location)
    ois$location <- gsub("&amp;", "", ois$location)
    ois$location <- gsub("^ *| *$", "", ois$location)

  # html
    ois$url <- gsub("&amp;", "&", ois$url)
    ois$url <- gsub(" ","%20",ois$url) # need %20 to indicate space
```

Reformat dates

```
  ois$date <- mdy(ois$date)
```

View working ois dataframe

```
  head(ois)
```

```
##      id       date                         location
## 1 18-01 2018-01-13 2800 Block of Kensington Avenue
## 2 18-02 2018-01-29     1300 Block of Bigler Street
## 3 18-08 2018-04-18    3100 block of N. 33rd Street
## 4 18-12 2018-06-08    1400 block of Lardner Street
## 5 18-16 2018-08-06       4800 block of Knox Street
## 6 18-17 2018-08-09     2000 block of Snyder Avenue
##                                                                                  url
## 1    http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/18-01.pdf
## 2 http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/OIS18-02.pdf
## 3 http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/OIS18-08.pdf
## 4    http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/18-12.pdf
## 5    http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/18-16.pdf
## 6    http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/18-17.pdf
```

Extracting Data from PDF Files

View and isolate date from pdf

```
  # Raw text
    pdf_file_0701 <- "http://www.phillypolice.com/assets/crime-maps-stats/officer-involved-shootings/20
    pdf_text_0701 <- pdf_text(pdf_file_0701)
    pdf_text_0701
```

```
## [1] "PS# 07-01\n1/1/07\nOn 1/1/07, at approximately 12:02 A.M., uniformed officers were traveling ea
```

```
  # Separate lines
    a <- strsplit(pdf_text_0701, split = "\n")[[1]] # used only \n for mac can be a combo of \r\n
    a
```

```
##  [1] "PS# 07-01"
##  [2] "1/1/07"
##  [3] "On 1/1/07, at approximately 12:02 A.M., uniformed officers were traveling east on"
##  [4] "Wingohocking Street from 18th Street when they heard multiple gunshots coming from the 4400"
##  [5] "block of N.17th Street. When approaching the intersection of 17th and Wingohocking Street, the"
##  [6] "officers observed a male discharging an assault rifle into the air on the 4400 block of N. 17t"
##  [7] "Street."
##  [8] "The officers exited their patrol vehicle, identified themselves as police officers, and ordere"
##  [9] "offender to put the weapon down. The offender then turned and pointed the weapon in the"
## [10] "officers' direction. One of the officers discharged his weapon at the offender. The offender"
## [11] "dropped the weapon and fled into a residence on the 4400 block of N. 17th Street where he was"
## [12] "apprehended after a brief struggle."
## [13] "A .223 caliber rifle loaded with four live rounds was recovered at the scene. There were no"
## [14] "reported injuries resulting from the police discharge. Two additional apprehensions were made "
## [15] "the scene for assault on police and related offenses."
```

```
  # Apply mdy() to a
    a <- mdy(a)
```

For-loops to extract text from pdfs

```
ois$text <- NA

for(i in 1:nrow(ois))
{
  a <- pdf_text(ois$url[i])
  a <- paste(a, collapse = "\n") # collapse into one page
  ois$text[i] <- a
}
```

For-loops to extract dates and add to missing date fields

```
for(i in which(is.na(ois$date)))
{
  a <- strsplit(ois$text[i], split = "\n")[[1]]
  a <- sort(mdy(a))[1]

  ois$date[i] <- as.character(a)
}
```

Check date column

```
  # Find any missing fields
    sum(is.na(ymd(ois$date)))
```

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

```
  # Reconcile id number and year (e.g. id starting with 07 only have dates in 2007)
    aggregate(year(date) ~ substr(id,1,2), data = ois, FUN = unique)
```

```
##    substr(id, 1, 2) year(date)
## 1                07       2007
## 2                08       2008
## 3                09       2009
## 4                10       2010
## 5                11       2011
## 6                12       2012
## 7                13       2013
## 8                14       2014
## 9                15       2015
## 10               16       2016
## 11               17       2017
## 12               18       2018
```

Geocoding OIS Locations

Automate the reformatting of URLs (

Open Street Maps Function

```
geocode_osm <- function(address)
{
  a <- gsub(" +", "\\%20", address)
  a <- paste0("http://nominatim.openstreetmap.org/search/", a, "?format=json&addressdetails=0&limit=1
  return( fromJSON(a) )
}
```

###ArcGIS Function

```
geocode_arcgis <- function(address)
{
  a <- gsub(" +", "\\%20", address)
  a <- paste0("https://geocode.arcgis.com/arcgis/rest/services/World/GeocodeServer/findAddressCandida
  return( fromJSON(a) )
}
```

Clean up addresses

View OISs with missing locations

```
i <- which(ois$location %in% c("", "withheld", "Withheld"))
ois$text[i]
```

```
##  [1] "PS#1618\n5/31/16\nOn Tuesday, May 31, 2016, at approximately 1:12 PM, an off-
duty officer,\nin civilian attire, arrived home at his residence. Upon entering the front\ndoor, the of
duty detective\napprehended the other offender near Brighton and Hawthorne Streets.\nThere were no repo
##  [2] "PS# 1626\n9/05/16\nOn Monday, September 5, 2016, at approximately 6:28 P.M., an off-
duty\nofficer, in plainclothes, became involved in a verbal and physical altercation\nwith his son at t
Torresdale Hospital for treatment.\nThe officer's firearm, a .40 caliber semiautomatic pistol, loaded w
##  [3] "PS#10-06\n1/20/10\nOn 1/20/10, at approximately 7:36 P.M., plainclothes officers traveling in a
##  [4] "PS# 09-25\n3/21/09\nOn 3/21/09, at approximately 10:17 P.M., an officer, on-duty and in uniform
##  [5] "PS# 09-27\n4/7/09\nOn 4/7/09, at approximately 6:52 P.M., uniformed officers responded to a Po
##  [6] "\t\r  \nPS# 09-76\n10/25/09\nOn 10/25/09, at approximately 2:15 A.M., a uniformed officer obse
##  [7] "PS# 08-06\n1/11/08\nOn 1/11/08, between approximately 8:50 PM and 9:20 PM, plainclothes office
##  [8] "PS# 08-18\n2/25/08\nOn 02/25/08, at approximately 11:06 AM, uniform officers responded to a rad
##  [9] "PS# 08-30\n4/2/08\nOn 4/2/08, at approximately 8:32 P.M., uniformed officers responded to a bu
## [10] "PS# 08-35\n5/1/08\nOn 5/1/08, at approximately 12:54 P.M., uniformed bicycle officers recogniz
## [11] "PS# 08-40\n6/13/08\nOn 6-13-08, at approximately 8:42 P.M., plainclothes officers received inf
## [12] "PS# 08-60\n10/7/08\nOn 10/7/08, at approximately 2:46 P.M., uniformed officers responded to a
## [13] "PS #08-70\n12/5/08\nOn 2/5/08, at approximately 4:25 P.M., a call was received at Police Radio
## [14] "PS# 08-74\n12/27/08\nOn12/27/08, at approximately 11:14 A.M., uniformed officers responded to a
## [15] "PS# 07-27\n4/21/07\nOn 4/21/07, at approximately 12:57 A.M., a uniformed officer stopped severa
```

Add or correct missing locations

```
ois$location[ois$id=="16-18"] <- "3200 block of Wellington Street"
ois$location[ois$id=="10-06"] <- "Howard and Grange Street"
ois$location[ois$id=="08-06"] <- "200 block of Clapier Street"
ois$location[ois$id=="08-18"] <- "900 block of E. Slocum Street"
ois$location[ois$id=="08-30"] <- "700 block of W. Rockland Street"
ois$location[ois$id=="08-40"] <- "5400 Jefferson Street"
ois$location[ois$id=="08-60"] <- "3000 Memphis Street"
ois$location[ois$id=="08-70"] <- "1300 block of S. 29th Street"
ois$location[ois$id=="08-74"] <- "5600 block of N. Mascher Street"
```

Other type of address corrections

```
# Insert "and" between cross streets
  ois$location[ois$id=="07-19"] <- "51st and Arch" # needs "and"

# Shooting that didn't involve PPD officers
  ois$text[ois$id=="17-08"] # shooting didn't involve PPD officers
```

```
## [1] "OIS# 1708 (March 29, 2017)\nOn Wednesday, March 29, 2017, at approximately 5:39 PM, two uniform
```

```
  ois <- subset(ois, id != "17-08")

# Assign address with "blocks" at the midpoint, e.g. change "5400 block of Erdick St" to "5450 Erdick
  a <- gsub("00 block( of)?", "50", a, ignore.case=TRUE)
  a <- gsub("unit bl(oc)?k( of)?", "50", a, ignore.case=TRUE)

# Alleys
  a <- gsub("Rear Alley of |near ", "", a, ignore.case = TRUE)

# Add city and state to improve geocoding accuracy
  a <- paste0(ois$location, ", Philadelphia, PA")
```

Geocode locations

Run addresses through OpenStreetMap geocoder

```
ois_gc <- vector("list", nrow(ois)) #list to store geocoding results
for(i in 1:nrow(ois))
{
  ois_gc[[i]] <- geocode_osm(a[i])
  if(length(ois_gc[[i]]) == 0)
  {
    cat("Could not geocode address #", i, ":", a[i], "/n")
  }
}
```

```
## Could not geocode address # 1 : 2800 Block of Kensington Avenue, Philadelphia, PA /nCould not geocod
```

Run failed addresses through ArcGIS geocoder

```
for (i in which(sapply(ois_gc, length) == 0))
{
  ois_gc[[i]] <- geocode_arcgis(a[i])
}
```

Reconcile formatting of OSM and ArcGIS results to extract longitude and latitude

```
# Function to extract longitude and latitude
  b <- lapply(ois_gc, function(x)
  {
    if(is.null(x$candidates)) #OSM
    {
      a <- data.frame(lon = as.numeric(x$lon),
                      lat = as.numeric(x$lat),
                      score = as.numeric(x$importance),
                      loctype = paste(x$class, x$type, sep = ":"),
                      method = "osm",
```

```
                        addressGeo = x$display_name,
                        stringsAsFactors = FALSE)
      } else # ArcGIS
      {
        a <- data.frame(lon = x$candidates$location[1, "x"],
                        lat = x$candidates$locatio[1, "y"],
                        score = x$candidates$score[1],
                        loctype = x$candidates$attributes$Addr_type[1],
                        method = "arcgis",
                        addressGeo = x$candidates$attributes$Match_addr[1],
                        stringsAsFactors = FALSE)
      }
      return(a)
    })

  # Combine OSM and ArcGIS results
    ois_gc <- do.call(rbind, b)

  # Add a column containing the original address
    ois_gc <- cbind(ois_gc, addressOrig = a)

  # View dataframe
    head(ois_gc)
```

```
##         lon      lat score       loctype method
## 1 -75.12253 39.99151   100 StreetAddress arcgis
## 2 -75.16997 39.91379   100  PointAddress arcgis
## 3 -75.18385 40.00434   100  PointAddress arcgis
## 4 -75.08054 40.03605   100 StreetAddress arcgis
## 5 -75.16408 40.02508   100 StreetAddress arcgis
## 6 -75.17930 39.92555   100 StreetAddress arcgis
##                                                addressGeo
## 1 2800 Kensington Ave, Philadelphia, Pennsylvania, 19134
## 2      1300 Bigler St, Philadelphia, Pennsylvania, 19148
## 3      3100 N 33rd St, Philadelphia, Pennsylvania, 19132
## 4     1400 Lardner St, Philadelphia, Pennsylvania, 19149
## 5        4800 Knox St, Philadelphia, Pennsylvania, 19144
## 6     2000 Snyder Ave, Philadelphia, Pennsylvania, 19145
##                                        addressOrig
## 1 2800 Block of Kensington Avenue, Philadelphia, PA
## 2     1300 Block of Bigler Street, Philadelphia, PA
## 3    3100 block of N. 33rd Street, Philadelphia, PA
## 4    1400 block of Lardner Street, Philadelphia, PA
## 5       4800 block of Knox Street, Philadelphia, PA
## 6     2000 block of Snyder Avenue, Philadelphia, PA
```

Double-check the longitude and latitude

Check all coordinates and examine outliers

```
  # View all longitude and latitude
    stem(ois_gc$lat)
```

```
##
##   The decimal point is at the |
##
##   31 | 1
##   31 |
##   32 |
##   32 |
##   33 |
##   33 |
##   34 |
##   34 |
##   35 |
##   35 |
##   36 |
##   36 |
##   37 |
##   37 |
##   38 |
##   38 |
##   39 |
##   39 | 99999999999999999999999999999999999999999999999999999999999999999999999999
##   40 | 000000000000000000000000000000000000000000000000000000000000000000000000000+308
```

```
    stem(ois_gc$lon)
```

```
##
##   The decimal point is at the |
##
##   -85 | 1
##   -84 |
##   -83 |
##   -82 |
##   -81 |
##   -80 |
##   -79 |
##   -78 |
##   -77 |
##   -76 |
##   -75 | 333222222222222222222222222222222222222222222222222222222222222222222222222+379
```

```
  # View outliers
    subset(ois_gc, lon < -80) # Philadelphia, MS
```

```
##           lon      lat score             loctype method
## 378 -85.08885 31.14714  0.41 highway:residential    osm
##                                                                                  addressGeo
## 378 Jefferson Street, Philadelphia, Gordon, Houston County, Alabama, United States of America
##                                 addressOrig
## 378 5400 Jefferson Street, Philadelphia, PA
```

Run outlier coordinates through ArcGIS geocoder

```
i <- which(ois_gc$lon < -80)
b <- lapply(ois_gc$addressOrig[i], geocode_arcgis)

b <- lapply(b, function(x)
{
  data.frame(lon = x$candidates$location[1, "x"],
             lat = x$candidates$locatio[1, "y"],
             score = x$candidates$score[1],
             loctype = x$candidates$attributes$Addr_type[1],
             method = "arcgis",
             addressGeo = x$candidates$attributes$Match_addr[1],
             stringsAsFactors = FALSE)
})

b <- do.call(rbind, b)

ois_gc[i, names(b)] <- b
ois_gc[i,]
```

```
##           lon      lat score       loctype method
## 378 -75.23095 39.97907  99.5 StreetAddress arcgis
##                                                 addressGeo
## 378 5400 W Jefferson St, Philadelphia, Pennsylvania, 19131
##                                 addressOrig
## 378 5400 Jefferson Street, Philadelphia, PA
```

Geocoding issues with location type

Location type should be specific locations (office, shop, building, station, house, PointAddress, StreetAddress, StreetInt). Highway, neighborhood, city, Locality, and StreetName are too general.

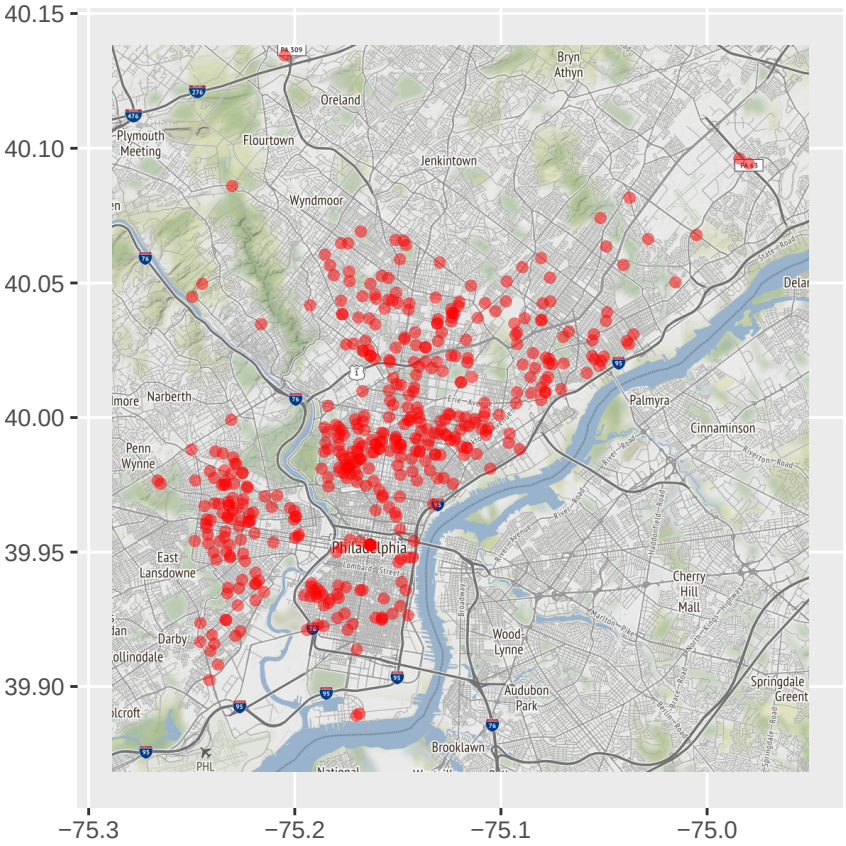
```
sort(table(ois_gc$loctype))
```

```
##
##             amenity:bar          building:house      building:warehouse
##                       1                       1                       1
##           building:yes         highway:service           highway:trunk
##                       1                       1                       1
##       office:government     place:neighbourhood        shop:convenience
##                       1                       1                       1
##        StreetAddressExt        highway:tertiary         railway:station
##                       1                       2                       2
##       highway:secondary                Locality boundary:administrative
##                       3                       3                       6
##         highway:primary     highway:residential              StreetName
##                       7                      11                      29
##               StreetInt           PointAddress             place:house
##                      38                      91                     111
##           StreetAddress
##                     147
```

Map Officer Involved Shootings in Philadelphia

```
# Place all essential information in one data frame
  ois$lon <- ois_gc$lon
  ois$lat <- ois_gc$lat

# Get map of Philly
  map_philly <- get_map(c(-75.288486, 39.868285, -74.950965, 40.138251), source = "stamen")
  ggmap(map_philly, extent = "normal") +
    geom_point(aes(x = lon, y = lat), data = ois,
               color = "red",
               alpha = 0.5) +
    theme(axis.title.x = element_blank(),
          axis.title.y = element_blank())
```



Working with Shapefiles and Coordinate Systems

Plot OIS Data on the Philly Police Service Area (PSA) Maps

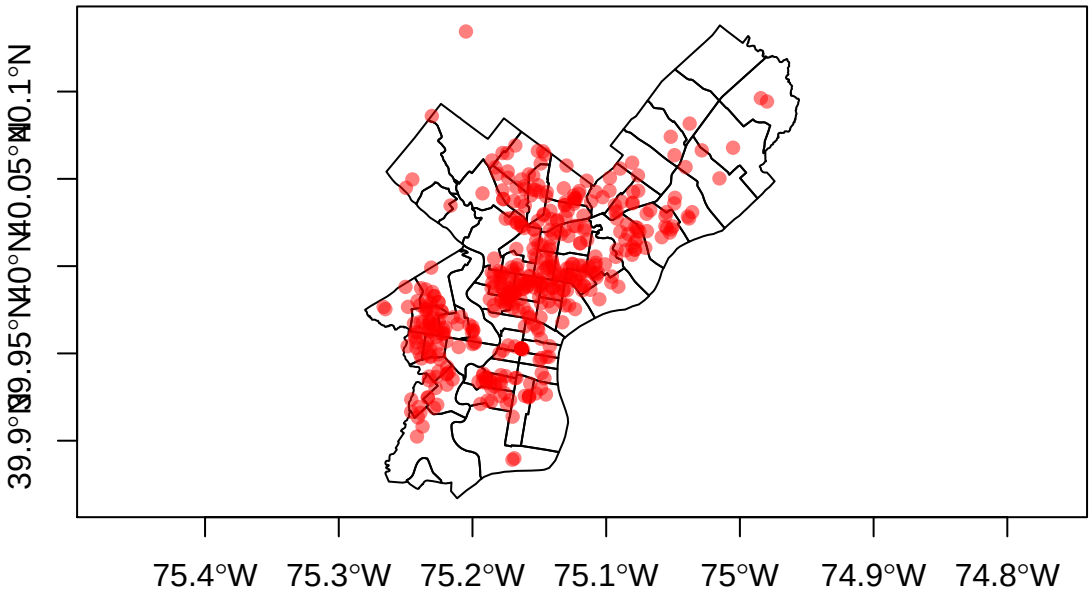
```
# Load shapefile and check map projection
  map_ppd <- st_read("Boundaries_PSA/Boundaries_PSA.shp")
```

```
## Reading layer `Boundaries_PSA' from data source `/Users/leng/Dropbox (Personal)/Programming/OG Github
## Simple feature collection with 66 features and 10 fields
## geometry type:  POLYGON
## dimension:      XY
## bbox:           xmin: -75.28031 ymin: 39.86701 xmax: -74.95575 ymax: 40.13793
## epsg (SRID):    4326
## proj4string:    +proj=longlat +datum=WGS84 +no_defs
```

```
    st_crs(map_ppd)
```

```
## Coordinate Reference System:
##   EPSG: 4326
##   proj4string: "+proj=longlat +datum=WGS84 +no_defs"
```

```
  # Plot
    plot(st_geometry(map_ppd), axes = TRUE)
    points(lat~lon, data = ois, col = rgb(1,0,0,0.5), pch = 16)
```



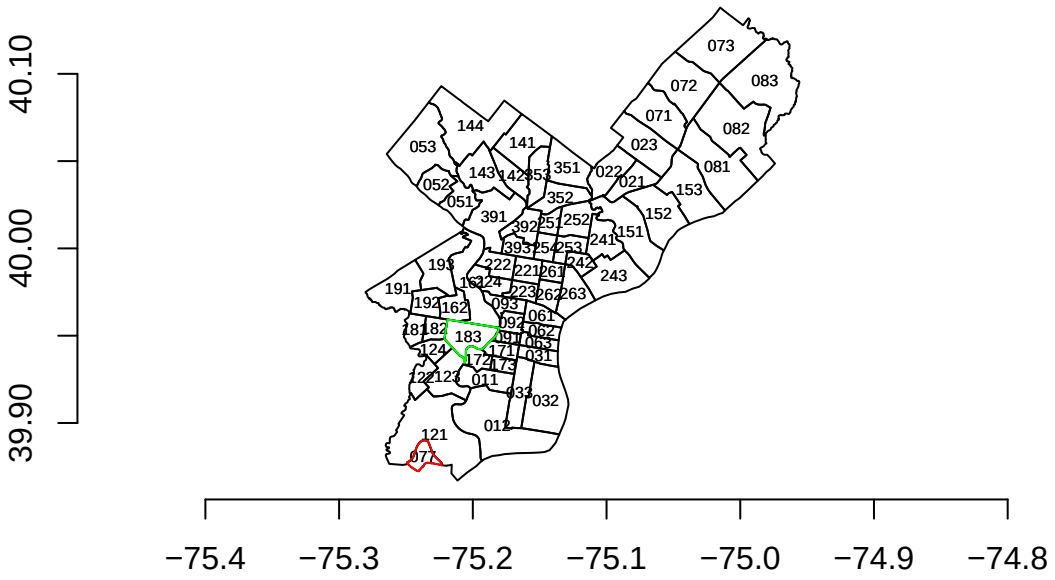
Exercise: Adding visual features to a map

Add PSA number labels & highlight PSA areas

```
# Plot PSA map
  plot(st_geometry(map_ppd))
  axis(side =1) # x-axis
  axis(side = 2) # y-axis

# Add number labels (center aligned within the polygon)
  a <- st_coordinates(st_centroid(st_geometry(map_ppd))) # extract coordinates from polygon
  text(a[,1], a[,2], map_ppd$PSA_NUM, cex = 0.5)
```

```
# Highlight PSA 77 & 183
  text(a[,1], a[,2], map_ppd$PSA_NUM, cex = 0.5)
  plot(st_geometry(subset(map_ppd, PSA_NUM == "077")),
       add = TRUE, border = "red")
  plot(st_geometry(subset(map_ppd, PSA_NUM == "183")),
       add = TRUE, border = "green")
```

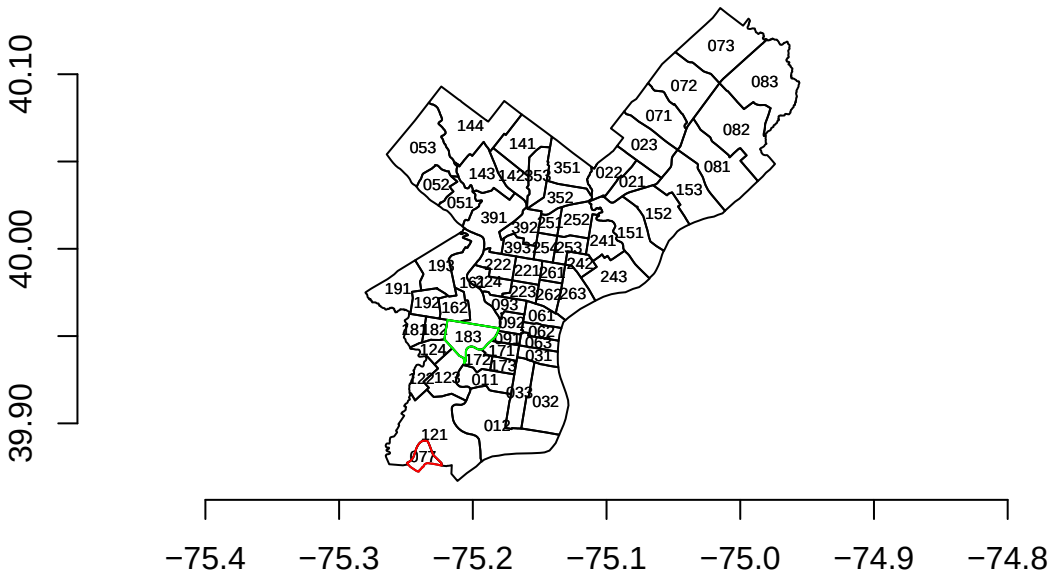


Add PSA number labels & highlight PSA areas

```
# Plot PSA map
  plot(st_geometry(map_ppd))
  axis(side =1) # x-axis
  axis(side = 2) # y-axis

# Add number labels (center aligned within the polygon)
  a <- st_coordinates(st_centroid(st_geometry(map_ppd))) # extract coordinates from polygon
  text(a[,1], a[,2], map_ppd$PSA_NUM, cex = 0.5)


# Highlight PSA 77 & 183
  text(a[,1], a[,2], map_ppd$PSA_NUM, cex = 0.5)
  plot(st_geometry(subset(map_ppd, PSA_NUM == "077")),
       add = TRUE, border = "red")
  plot(st_geometry(subset(map_ppd, PSA_NUM == "183")),
       add = TRUE, border = "green")
```



Exercise: Spatial joins & highlight areas with the most OIS

```
  # Indicate coordinate system for latitude and longitude values
    ois <- st_as_sf(ois,
                    coords = c("lon", "lat"),
                    crs = st_crs("+init=epsg:4326")) #  World Geodetic System 1984 (used in GPS)

  # Match each row in "ois" to each PSA polygon
    lookup_psa <- st_join(ois, map_ppd[,"PSA_NUM"])
    lookup_psa[1:3, c("id", "date", "location", "PSA_NUM", "geometry")]
```

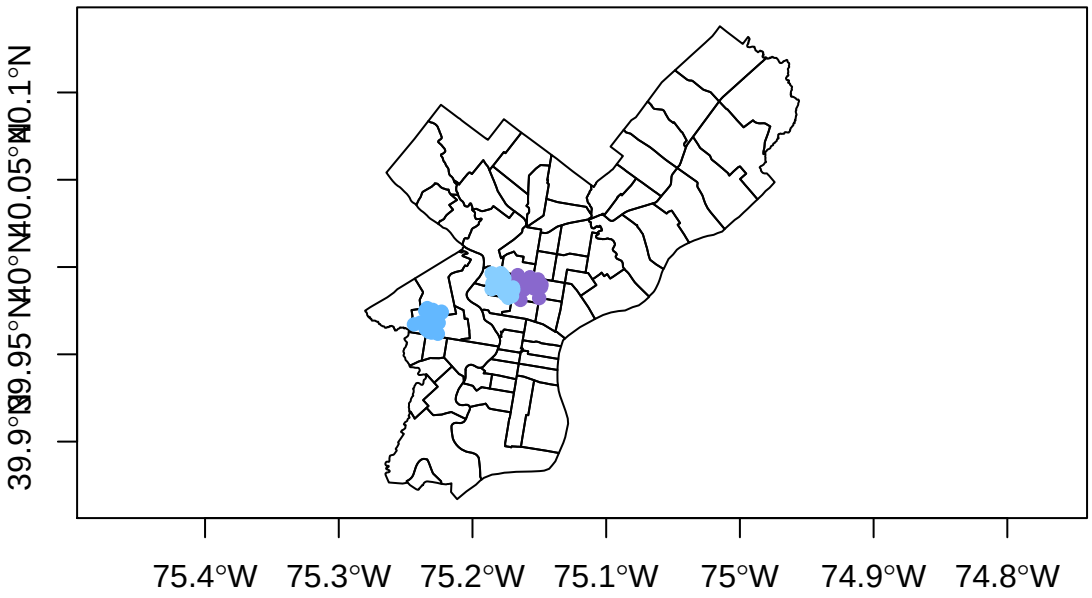
```
## Simple feature collection with 3 features and 4 fields
## geometry type:  POINT
## dimension:      XY
## bbox:           xmin: -75.18385 ymin: 39.91379 xmax: -75.12253 ymax: 40.00434
## epsg (SRID):    4326
## proj4string:    +proj=longlat +datum=WGS84 +no_defs
##      id       date                          location PSA_NUM
## 1 18-01 2018-01-13 2800 Block of Kensington Avenue     242
## 2 18-02 2018-01-29     1300 Block of Bigler Street     033
## 3 18-08 2018-04-18    3100 block of N. 33rd Street     391
##                     geometry
## 1 POINT (-75.12253 39.99151)
## 2 POINT (-75.16997 39.91379)
## 3 POINT (-75.18385 40.00434)
```

```
  # Find top 3 PSAs with the most OIs
    a <- rev(sort(table(lookup_psa$PSA)))
    a
```

```
##
## 221 192 222 151 352 253 224 141 351 254 242 241 182 391 252 172 152 123 261 193
```

```
##  26  22  21  18  13  13  13  13  11  11  11  11  11  10  10  10  10  10   9   9
## 181 162 392 353 092 021 393 263 251 223 191 173 142 033 011 161 153 022 243 121
##   9   9   8   8   8   8   7   7   7   7   7   7   7   7   7   6   6   6   5   5
## 061 262 124 122 063 183 143 081 032 012 144 091 083 071 062 053 031 023 082 072
##   5   4   4   4   4   3   3   3   3   3   2   2   2   2   2   2   2   2   1   1
## 051 171 093 077 073 052
##   1   0   0   0   0   0
```

```r
# Plot
  plot(st_geometry(map_ppd), axes = TRUE)
  i <- which(lookup_psa$PSA_NUM == names(a)[1])
  plot(st_geometry(lookup_psa[i,]), add = TRUE, col = "mediumpurple3", pch = 16)
  i <- which(lookup_psa$PSA_NUM == names(a)[2])
  plot(st_geometry(lookup_psa[i,]), add = TRUE, col = "steelblue1", pch = 16)
  i <- which(lookup_psa$PSA_NUM == names(a)[3])
  plot(st_geometry(lookup_psa[i,]), add = TRUE, col = "skyblue1", pch = 16)
```



Exercise: Tabulate OIS count by PSA and add color to the map

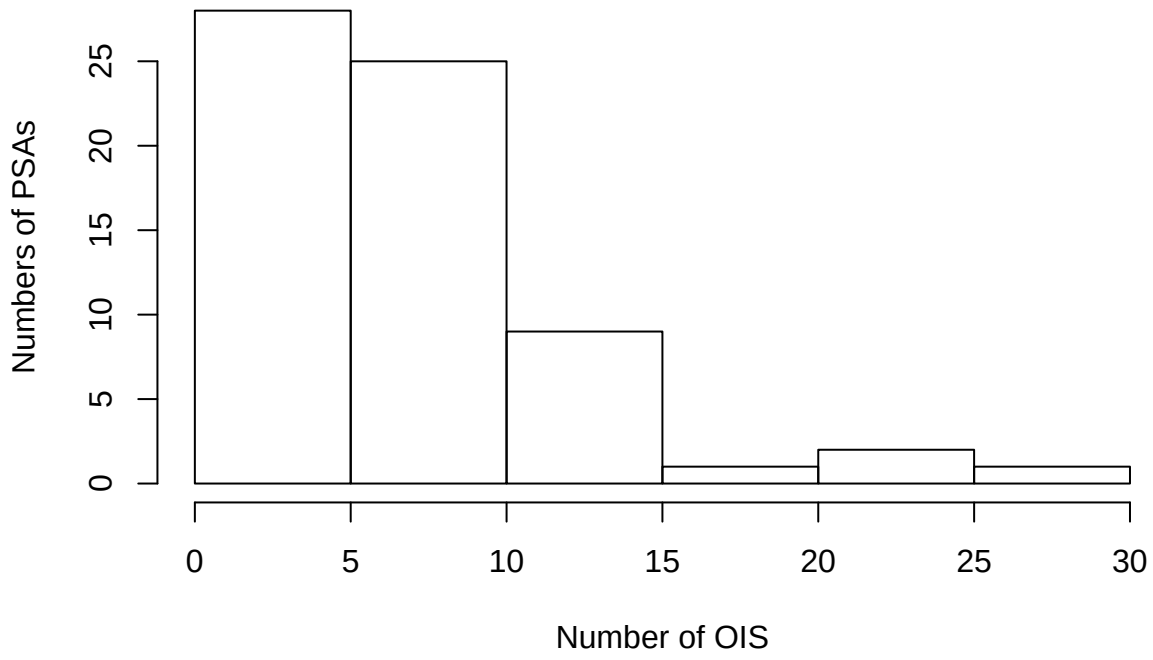
```r
# How many shootings in each PSA?
  a <- table(lookup_psa$PSA_NUM)

# Merge shooting counts into "map_ppd" data
  i <- match(map_ppd$PSA_NUM, names(a))
  map_ppd$nShoot <- a[i]
  map_ppd[1:3,]
```

```
## Simple feature collection with 3 features and 11 fields
## geometry type:  POLYGON
## dimension:      XY
## bbox:           xmin: -75.24925 ymin: 39.87239 xmax: -75.13535 ymax: 39.93435
## epsg (SRID):    4326
```

```
## proj4string:    +proj=longlat +datum=WGS84 +no_defs
##   OBJECTID AREA PERIMETER PSACOV_ PSACOV_ID ID DISTRICT__ PSA_NUM    OLD_SECTOR
## 1        1 <NA>      <NA>    <NA>      <NA>  1       <NA>     077            A
## 2        2 <NA>      <NA>    <NA>      <NA>  2       <NA>     011 A,B,C,D,E,F,J
## 3        3 <NA>      <NA>    <NA>      <NA>  3       <NA>     012 G,H,I,K,L,M,N
##   DESCRIPT                       geometry nShoot
## 1     <NA> POLYGON ((-75.2338 39.88977...      0
## 2     <NA> POLYGON ((-75.19724 39.9294...      7
## 3     <NA> POLYGON ((-75.17305 39.9105...      3
```

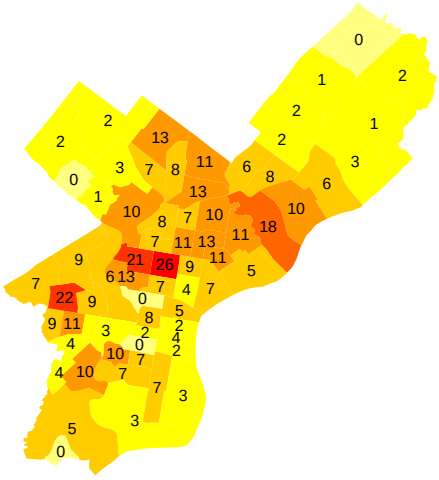
```
# Historgram to view distribution of counts
  hist(a, xlab = "Number of OIS", ylab = "Numbers of PSAs", main = "")
```




```
# OIS counts into categories
  a <- cut(map_ppd$nShoot,
           breaks = c(0,1,5,10,15,20,25,30),
           right = FALSE)

# Heat color for the map
  col <- rev(heat.colors(7,1))

# Assign colors to each PSA & add PSA labels
  plot(st_geometry(map_ppd), col = col[as.numeric(a)], border = NA)
  a <- st_coordinates((st_centroid(map_ppd)))
  text(a[,1], a[,2], map_ppd$nShoot, cex = 0.5)
```


0
1
2
2
2
1
2
13
2
3
7
8
11
6
8
3
0
1
13
6
10
8
7
10
10
18
7
11
13
11
9
21
26
9
11
5
6
13
7
4
7
7
22
9
0
5
8
2
9
11
3
2
4
4
0
2
10
7
4
10
7
7
3
3
5
0