

hw_5_R

Stephanie Pusker

2022-11-13

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr   0.3.5
## v tibble  3.1.8      v dplyr    1.0.10
## v tidyrr  1.2.1      v stringr  1.4.1
## v readr   2.1.3      v forcats 0.5.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()

library(stringr)
library(purrr)
library(broom)
library(scales)

##
## Attaching package: 'scales'
##
## The following object is masked from 'package:purrr':
## 
##     discard
## 
## The following object is masked from 'package:readr':
## 
##     col_factor
library(ggthemes)
library(sf)

## Linking to GEOS 3.10.2, GDAL 3.4.2, PROJ 8.2.1; sf_use_s2() is TRUE
library(tigris)

## To enable caching of data, set `options(tigris_use_cache = TRUE)`
## in your R script or .Rprofile.

library(lubridate)

##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
## 
##     date, intersect, setdiff, union
```

```

library(flexdashboard)
library(shiny)
library(jsonlite)

##
## Attaching package: 'jsonlite'
##
## The following object is masked from 'package:shiny':
##       validate
##
## The following object is masked from 'package:purrr':
##       flatten
library(mapproj)

## Loading required package: sp
## Checking rgeos availability: FALSE
## Please note that 'mapproj' will be retired during 2023,
## plan transition at your earliest convenience;
## some functionality will be moved to 'sp'.
##       Note: when rgeos is not available, polygon geometry      computations in mapproj depend on gpclib
##       which has a restricted licence. It is disabled by default;
##       to enable gpclib, type gpclibPermit()

library(leaflet)
library(plotly)

##
## Attaching package: 'plotly'
##
## The following object is masked from 'package:ggplot2':
##       last_plot
##
## The following object is masked from 'package:stats':
##       filter
##
## The following object is masked from 'package:graphics':
##       layout
library(DT)

##
## Attaching package: 'DT'
##
## The following objects are masked from 'package:shiny':
##       dataTableOutput, renderDataTable
library(viridis)

## Loading required package: viridisLite
##

```

```

## Attaching package: 'viridis'
##
## The following object is masked from 'package:scales':
##
##     viridis_pal
library(rgdal)

## Please note that rgdal will be retired during 2023,
## plan transition to sf/stars/terra functions using GDAL and PROJ
## at your earliest convenience.
## See https://r-spatial.org/r/2022/04/12/evolution.html and https://github.com/r-spatial/evolution
## rgdal: version: 1.6-2, (SVN revision 1183)
## Geospatial Data Abstraction Library extensions to R successfully loaded
## Loaded GDAL runtime: GDAL 3.4.2, released 2022/03/08
## Path to GDAL shared files: /Library/Frameworks/R.framework/Versions/4.2/Resources/library/rgdal/gdal
## GDAL binary built with GEOS: FALSE
## Loaded PROJ runtime: Rel. 8.2.1, January 1st, 2022, [PJ_VERSION: 821]
## Path to PROJ shared files: /Library/Frameworks/R.framework/Versions/4.2/Resources/library/rgdal/proj
## PROJ CDN enabled: FALSE
## Linking to sp version:1.5-1
## To mute warnings of possible GDAL/OSR exportToProj4() degradation,
## use options("rgdal_show_exportToProj4_warnings"="none") before loading sp or rgdal.

library(tmap)
library(cowplot)

##
## Attaching package: 'cowplot'
##
## The following object is masked from 'package:lubridate':
##
##     stamp
##
## The following object is masked from 'package:ggthemes':
##
##     theme_map

homicides <- read_csv("https://raw.githubusercontent.com/washingtonpost/data-homicides/master/homicide-"

## Rows: 52179 Columns: 12
## -- Column specification -----
## Delimiter: ","
## chr (9): uid, victim_last, victim_first, victim_race, victim_age, victim_sex...
## dbl (3): reported_date, lat, lon
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.

homicides <- homicides %>%
  unite(city_name, city, state, sep = ", ")

homicides %>%
  group_by(city_name) %>%
  summarize(total = n()) %>%
  arrange(desc(total))

```

```

slice(-51)

## # A tibble: 50 x 2
##   city_name      total
##   <chr>        <int>
## 1 Chicago, IL     5535
## 2 Philadelphia, PA    3037
## 3 Houston, TX     2942
## 4 Baltimore, MD     2827
## 5 Detroit, MI     2519
## 6 Los Angeles, CA    2257
## 7 St. Louis, MO     1677
## 8 Dallas, TX       1567
## 9 Memphis, TN      1514
## 10 New Orleans, LA    1434
## # ... with 40 more rows

dc_homicides <- homicides %>%
  filter(city_name == "Washington, DC") %>%
  mutate(reported_date = ymd(reported_date))

dc_homicides

## # A tibble: 1,345 x 11
##   uid reported~1 victi~2 victi~3 victi~4 victi~5 victi~6 city_~7    lat    lon
##   <chr> <date>    <chr>    <chr>    <chr>    <chr>    <chr>    <dbl> <dbl>
## 1 Was-0~ 2015-01-03 ROSS    RAHJI    Black    35      Male    Washin~ 38.9 -77.0
## 2 Was-0~ 2015-01-08 STEWART MARVIN  Black    49      Male    Washin~ 38.9 -77.0
## 3 Was-0~ 2015-01-10 CARIAS   MARTIN~ Hispan~ 21      Male    Washin~ 38.9 -77.0
## 4 Was-0~ 2015-01-11 NEWMAN   ANDREW  Black    17      Male    Washin~ 38.9 -77.0
## 5 Was-0~ 2015-01-11 ANDERS~ JAMES    Black    27      Male    Washin~ 38.9 -77.0
## 6 Was-0~ 2015-01-14 JONES    PHILLIP Black    17      Male    Washin~ 38.8 -77.0
## 7 Was-0~ 2015-01-19 OWENS    KEVIN    Black    22      Male    Washin~ 38.8 -77.0
## 8 Was-0~ 2015-01-25 WILLIA~ GERALD   Black    45      Male    Washin~ 38.9 -77.0
## 9 Was-0~ 2015-02-03 HOWARD   NAVONT~ Black    19      Male    Washin~ 38.9 -77.0
## 10 Was-0~ 2015-02-04 JONES    TRACEY   Black    46      Male    Washin~ 38.9 -77.0
## # ... with 1,335 more rows, 1 more variable: disposition <chr>, and abbreviated
## #   variable names 1: reported_date, 2: victim_last, 3: victim_first,
## #   4: victim_race, 5: victim_age, 6: victim_sex, 7: city_name

dc_districts <- state_legislative_districts(state = "DC", cb = TRUE, class = "sf")

## Retrieving data for the year 2020
##   |
unique(dc_homicides$victim_race)

## [1] "Black"      "Hispanic"   "White"      "Asian"      "Other"
dc_homicides %>%
  group_by(victim_race) %>%
  summarise(total = n())

## # A tibble: 5 x 2
##   victim_race total
##   <chr>        <int>

```

```

## 1 Asian          14
## 2 Black         1217
## 3 Hispanic       65
## 4 Other          7
## 5 White          42

dc_race <- dc_homicides %>%
  mutate(unsovled = disposition == c("Closed without arrest", "Open/No arrest")) %>%
  group_by(victim_race) %>%
  arrange(desc(victim_race)) %>%
  mutate(victim_race = fct_lump(victim_race)) %>%
  filter(!victim_race == c("Other", "Asian")) %>%
  ungroup() %>%
  slice(-c(43:45),
        -c(1328:1334))

## Warning in disposition == c("Closed without arrest", "Open/No arrest"): longer
## object length is not a multiple of shorter object length

## Warning in `==.default`(victim_race, c("Other", "Asian")): longer object length
## is not a multiple of shorter object length

## Warning in is.na(e1) | is.na(e2): longer object length is not a multiple of
## shorter object length

## Warning in `==.default`(victim_race, c("Other", "Asian")): longer object length
## is not a multiple of shorter object length

## Warning in is.na(e1) | is.na(e2): longer object length is not a multiple of
## shorter object length

## Warning in `==.default`(victim_race, c("Other", "Asian")): longer object length
## is not a multiple of shorter object length

## Warning in is.na(e1) | is.na(e2): longer object length is not a multiple of
## shorter object length

dc_race$unsovled <- as.integer(dc_race$unsovled == "true")

```

```

dc_race %>%
  group_by(victim_race) %>%
  summarise(total = n())

```

```

## # A tibble: 3 x 2
##   victim_race total
##   <fct>      <int>
## 1 Black        1217
## 2 Hispanic     65
## 3 White        42

```

Pick one city in the data. Create a map showing the locations of the homicides in that city, using the sf framework discussed in class. Use tigris to download boundaries for some sub-city geography (e.g., tracts, block groups, county subdivisions) to show as a layer underneath the points showing homicides. Use different facets for solved versus unsolved homicides and different colors to show the three race groups with the highest number of homicides for that city (you may find the fct_lump function from forcats useful for this).

```

EPSG <- make_EPSG()

dc_race_crs <- st_as_sf(dc_race, coords = c("lon", "lat")) %>%

```

```

st_set_crs(2248)

dc_districts_crs <- st_as_sf(dc_districts, coords = c("lon", "lat")) %>%
  st_set_crs(2248)

## Warning: st_crs<- : replacing crs does not reproject data; use st_transform for
## that

dc_plot <- ggplot() +
  geom_sf(data = dc_districts_crs) +
  geom_sf(data = dc_race_crs, aes(color = victim_race)) +
  theme_map() +
  facet_wrap(~ disposition)

ggplot() +
  geom_sf(data = dc_districts_crs) +
  geom_sf(data = dc_race_crs, aes(color = victim_race)) +
  facet_wrap(~ disposition) +
  ggtitle("Locations of Homicides in DC by Race and Disposition") +
  theme_map() +
  guides(color = guide_legend(title = "Victim Race"))

```

Locations of Homicides in DC by Race and Disposition

