U.S. Police Killings from 2015

Kemberli Jennings

2023-10-15

In 2015, The Washington Post started keeping a dataset of all the police killings in the United States since the police agencies weren't very diligent in keeping the dataset up to date.

The first iteration of the Washington Post's Police Killings data didn't include the 'explanatory' columns the newest dataset has. The latest iteration now looks at more unique variables that might add new insights to an individual ending up on this list who isn't in a policing entity. The single dataset used earlier is now two datasets. I could have used Excel for such a small dataset but used SQL to merge the tables.

The following is the SQL used to bring the two .csv tables together for the second iteration. I've since learned there's a way to eliminate going to an outside source to do SQL when using R, but there needed to be a hookup for BigQuery. This created file comes out of BigQuery as a .csv, and I used R for data cleaning instead of cleaning in an Excel file due to how the Excel file comes from BigQuery.

SELECT k.date, k.threat_type, k.flee_status, k.armed_with, k.city, k.state, k.name, k.age, k.gender, k.race, k.was_mental_illness_related, k.body_camera, k.agency_ids, a.agency_name, a.type FROM my-project-052823.PoliceKillings.PoliceKillings k JOIN my-project-052823.PoliceAgencies.PoliceAgencies a ON k.agency_ids = a.id AND k.state = a.state order by k.date;

Next is my R code. I was excited about this dataset, believing I'd find something 'new' about this problem. All of the usual comments/beliefs are out there and are well known. Plus, The Washington Post does an excellent job of putting all the known variables attached to some statistics in a visualization for everyone interested. I wasn't trying to repeat what they had done.

```
install.packages("janitor")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## package 'janitor' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\User\AppData\Local\Temp\RtmpaOgjXo\downloaded_packages
```

```
library(janitor)
```

```
##
## Attaching package: 'janitor'

## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```r
install.packages("here")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## package 'here' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\User\AppData\Local\Temp\RtmpaOgjXo\downloaded_packages
```

```r
library(here)
```

```
## here() starts at C:/Users/User/Documents/R
```

```r
install.packages("skimr")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## package 'skimr' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\User\AppData\Local\Temp\RtmpaOgjXo\downloaded_packages
```

```r
library(skimr)
install.packages("tidyverse")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## package 'tidyverse' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\User\AppData\Local\Temp\RtmpaOgjXo\downloaded_packages
```

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.3     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
install.packages("dplyr")
```

```
## Warning: package 'dplyr' is in use and will not be installed
```

```r
library(dplyr)

install.packages("readr")
```

```
## Warning: package 'readr' is in use and will not be installed
```

```r
library(readr)
#install.packages("readxl")
#library(readxl)

#for date manipulations
install.packages("lubridate")
```

```
## Warning: package 'lubridate' is in use and will not be installed
```

```r
library(lubridate)
install.packages("parsedate")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## package 'parsedate' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\User\AppData\Local\Temp\RtmpaOgjXo\downloaded_packages
```

```r
library(parsedate)
```

```
##
## Attaching package: 'parsedate'
##
## The following object is masked from 'package:readr':
##
##     parse_date
```

```r
install.packages("zoo")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## package 'zoo' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\User\AppData\Local\Temp\RtmpaOgjXo\downloaded_packages
```

```r
library(zoo)
```

```
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
#needed to install ggpubr for correlation test
install.packages("ggpubr")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## package 'ggpubr' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\User\AppData\Local\Temp\RtmpaOgjXo\downloaded_packages
```

```r
library(ggpubr)

#loading vcd package for correlation between non-numeric variables
install.packages("libgfortran")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## Warning: package 'libgfortran' is not available for this version of R
##
## A version of this package for your version of R might be available elsewhere,
## see the ideas at
## https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```r
install.packages("libquadmath")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)

## Warning: package 'libquadmath' is not available for this version of R
##
## A version of this package for your version of R might be available elsewhere,
## see the ideas at
## https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```r
install.packages("vcd")
```

```
## Installing package into 'C:/Users/User/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

```
## package 'vcd' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\User\AppData\Local\Temp\RtmpaOgjXo\downloaded_packages
```

```
library(vcd)
```

```
## Loading required package: grid
```

```
killings_df <- read_csv("SQLQuery_PoliceKillings.csv")
```

```
## Rows: 500 Columns: 15
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (11): date, threat_type, flee_status, armed_with, city, state, name, gen...
## dbl  (2): age, agency_ids
## lgl  (2): was_mental_illness_related, body_camera
##
## 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.
```

```
#checking for blank rows
x <- subset(killings_df, !complete.cases(killings_df))

#filling in the blanks
killings_df[killings_df==''] <- "undisclosed"

#changing yyyy-mm-dd to yyyy-mm
my_date <- format(as.Date(killings_df$date, "%d/%m/%Y"), "%Y-%m")

#add column called 'my_date'
killings_df$my_date <- my_date
```

I'm working on the second iteration

Since there are 3 more columns, I'd need to write a function to run the assocstats through the dataset's variables without my human interaction. But, using the old dataset, I came up with pairs that I thought would interact together and give that new insight into these killings, which is expressed in the upcoming programming.

```
#preparing to look for correlation between non-numeric variables
# Create a 2x2 contingency table
mytable_armrace <- table(killings_df$armed_with, killings_df$race)
mytable_fleerace <- table(killings_df$flee_status, killings_df$race)
mytable_staterace <- table(killings_df$state, killings_df$race)
mytable_camerapolice <- table(killings_df$agency_name, killings_df$body_camera)
mytable_racecamera <- table(killings_df$race, killings_df$body_camera)
mytable_racecity <- table(killings_df$race, killings_df$city)
glimpse(mytable_racecity)
```

```
##  'table' int [1:6, 1:393] 0 0 1 0 0 1 0 0 0 0 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:6] "A" "B" "H" "N" ...
##   ..$ : chr [1:393] "Albuquerque" "Allentown" "Aloha" "Alpine" ...
```

```r
mytable_cameracity <- table(killings_df$body_camera, killings_df$city)
mytable_policecity <- table(killings_df$agency_name, killings_df$city)

# Calculate stats associated with character vectors
assocstats(mytable_armrace)
```

```
##                     X^2 df P(> X^2)
## Likelihood Ratio 44.286 40  0.29562
## Pearson          46.144 40  0.23322
##
## Phi-Coefficient   : NA
## Contingency Coeff.: 0.3
## Cramer's V        : 0.141
```

```r
assocstats(mytable_fleerace)
```

```
##                     X^2 df P(> X^2)
## Likelihood Ratio 28.176 15 0.020496
## Pearson          24.535 15 0.056543
##
## Phi-Coefficient   : NA
## Contingency Coeff.: 0.218
## Cramer's V        : 0.129
```

```r
assocstats(mytable_staterace)
```

```
##                     X^2  df   P(> X^2)
## Likelihood Ratio 279.93 230 1.3596e-02
## Pearson          427.52 230 4.9849e-14
##
## Phi-Coefficient   : NA
## Contingency Coeff.: 0.682
## Cramer's V        : 0.417
```

```r
assocstats(mytable_camerapolice)
```

```
##                     X^2  df P(> X^2)
## Likelihood Ratio 234.99 369 1.000000
## Pearson          428.49 369 0.017583
##
## Phi-Coefficient   : NA
## Contingency Coeff.: 0.679
## Cramer's V        : 0.926
```

```r
assocstats(mytable_racecamera)
```

```
##                     X^2 df P(> X^2)
## Likelihood Ratio 12.836  5 0.024967
## Pearson          10.080  5 0.072994
##
```

```
## Phi-Coefficient   : NA
## Contingency Coeff.: 0.142
## Cramer's V        : 0.143
```

```
assocstats(mytable_racecity)
```

```
##                     X^2   df P(> X^2)
## Likelihood Ratio 941.21 1960        1
## Pearson             NaN 1960      NaN
##
## Phi-Coefficient   : NA
## Contingency Coeff.: NaN
## Cramer's V        : NaN
```

```
assocstats(mytable_cameracity)
```

```
##                     X^2  df P(> X^2)
## Likelihood Ratio 232.30 392  1.00000
## Pearson          422.31 392  0.14021
##
## Phi-Coefficient   : NA
## Contingency Coeff.: 0.677
## Cramer's V        : 0.919
```

```
assocstats(mytable_policecity)
```

```
##                       X^2     df P(> X^2)
## Likelihood Ratio   5642.2 144648        1
## Pearson          173414.0 144648        0
##
## Phi-Coefficient   : NA
## Contingency Coeff.: 0.999
## Cramer's V        : 0.969
```

The results of the correlation tests weren't very heartening. By using Cramer's V:

```
* police & city --  .8161699
* camera & city --  .6189389
```

The pair that showed significant correlation at 82% per Cramer's V, police and city, will have to be examined more to determine why that is so high. The surprising pair, camera and city at 62%, didn't show such a high correlation as the first pair but was surprising since 85% of the agencies do not use cameras!

This is from the old dataset. The new dataset has added fields that adds to the explanation of the story of how that individual ended up on this list. I'm looking forward to working with the new dataset to see if it will reveal something new.

The percentage of the cameras used was taken off of my Pivot Table information from the dataset.