I did this assignment in collaboration with Nahuel Simon Halaager

Due to lack of time, we weren’t able to finish the entire assignment, but we made through a big deal of assignment two.

Please find a tutorial with instructions on web scraping data on US police killings in this repository at <https://github.com/adivea/KilledbyPolice2020.git> . Clone it and depending on your familiarity with R, either

2.3) produce data visualisations that shed light on another interesting aspect of the police killing data:

First and foremost, we chose to work with the police killings. Therefore, we followed the instructions and downloaded the R packages. We then used the given URL <https://killedbypolice.net/kbp2020> , <https://killedbypolice.net/kbp> and <https://killedbypolice.net/kbp2018> to access the data in R.

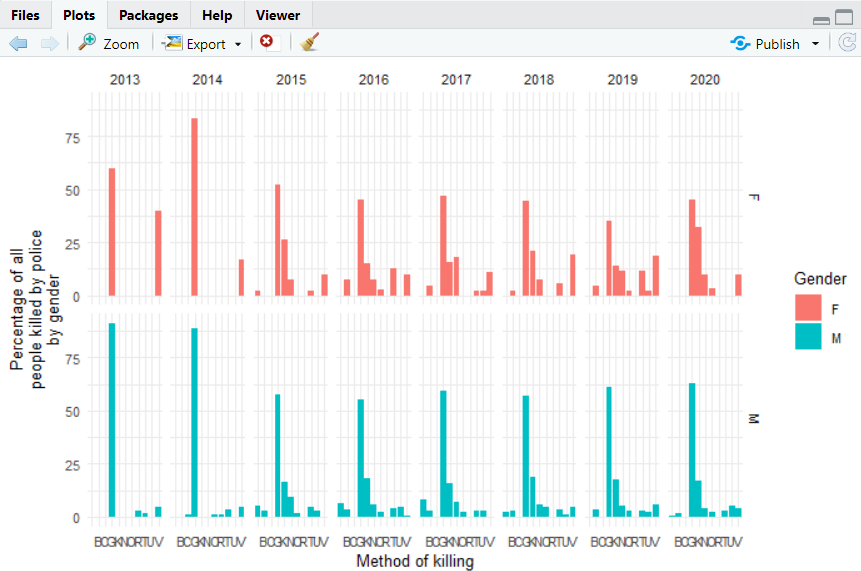
Firstly we created a plot that showed the age difference of male and female victims of police killing in the period 2013-2020:



Afterwards we created a plot that showed age from people killed by the police from multiple races from the time period 2013-2020. B = black people, W= white people, L = lantino people.



lastly we created a graph shows how the victims of police murders were killed, this includes both male and female.



And this where we got stuck, throughout three hours of work, we were unable to continue further with the assignment.

download.file(rvest)

install.packages("rvest")

install.packages("dplyr")

install.packages("tidyr")

install.packages("stringr")

install.packages("janitor")

library(rvest)

library(dplyr)

library(tidyr)

library(stringr)

library(janitor)

url <- "https://killedbypolice.net/kbp2020"

url\_html <- read\_html(url)

whole\_table <- url\_html %>%

html\_nodes("table") %>%

html\_table() #str(whole\_table) turns out to be a list

library(tidyverse)

head(whole\_table)

new\_table <- do.call(cbind,unlist(whole\_table, recursive = FALSE))

head(new\_table) # ok, looks good, too bad it took 2 hours

scrape\_police\_kill <- function(website){

url <- read\_html(website)

annual\_table <- url %>%

html\_nodes("table") %>%

html\_table() # result is a list

annual\_table <- do.call(cbind,unlist(annual\_table, recursive = FALSE))

}

table2018 <- scrape\_police\_kill("https://killedbypolice.net/kbp2018")

table2018 %>%

head()

mastertable=NULL

for (year in 2013:2020){ # here we create a loop to iterate over the years

print(year)

url <- "https://killedbypolice.net/kbp" # the annual URLs end with "kbp2017" ,etc.

website <- paste0(url,year) # here we bind the year to the website to form the URL

annual\_table <- scrape\_police\_kill(website) # here we apply the function

mastertable <- rbind(mastertable, annual\_table) # we add the scraped results from the given year to our master dataset

}

head(mastertable,2)

tail(mastertable)

mastertable <- as\_tibble(mastertable)

str(mastertable)

data <- mastertable %>%

mutate(Age = as.numeric(Age)) %>%

rename(Method = "\*")

mastertable$Date[c(70:80, 160:170)]

tail(unique(mastertable$Date))

library(lubridate)

data <- data %>%

mutate(Date =

case\_when(

grepl("201[34]",Date) ~ mdy(Date),

# convert dates that contain 2013 or 2014 into mdy format

!grepl("201[34]",Date)~ ymd(Date)))

# convert all other dates ymd format

data <- data %>%

mutate(Year = year(Date)) # I am creating a new column Year from the Date for later plots

tail(data$Year)

class(data$Date)

length(which(is.na(data$Date)))

write.csv(data,"data/policekillings202010.csv")

install.packages("ggridges")

install.packages("statebins")

library(ggplot2)

library(ggridges)

#graff for forskel på døde i forhold til mand/kvinde

data %>%

filter(Gender %in% c("F", "M", "T")) %>%

filter(!is.na(Year)) %>%

ggplot(aes(x = Age,

y = factor(Year),

fill = Gender)) +

geom\_density\_ridges(alpha = 0.5,

scale = 0.9) +

theme\_ridges(font\_size = 10) +

scale\_x\_continuous(breaks = seq(0, 100, 10),

labels = seq(0, 100, 10)) +

xlab("Age at death (years)") +

ylab("Year") +

theme(axis.title = element\_text(size = 14))

#Graph for døde Sorte overfor hvide

data %>%

filter(Race %in% c("B", "W", "L")) %>%

filter(!is.na(Year)) %>%

ggplot(aes(x = Age,

y = factor(Year),

fill = Race)) +

geom\_density\_ridges(alpha = 0.6,

scale = 0.9) +

theme\_ridges(font\_size = 10) +

scale\_x\_continuous(breaks = seq(0, 100, 10),

labels = seq(0, 100, 10)) +

xlab("Age at death (years)") +

ylab("Year") +

theme(axis.title = element\_text(size = 14))

#graf for måden de er dræbt på

data %>%

filter(!is.na(Year)) %>%

filter(Method != "NA") %>%

filter(Gender %in% c("M", "F", NA)) %>%

group\_by(Year,

Gender,

Method) %>%

tally() %>%

mutate(perc = n / sum(n) \* 100) %>%

ggplot(aes(Method,

perc,

fill = Gender)) +

geom\_col() +

facet\_grid(Gender~Year) +

theme\_minimal(base\_size = 10) +

xlab("Method of killing") +

ylab("Percentage of all\npeople killed by police\nby gender")

#install.packages(c("statebins", "viridis"))

library(statebins) # using GitHub version

library(viridis)

state\_abb <- data\_frame(state\_name = state.name,

state\_abb = state.abb)

url2 <- "https://www2.census.gov/programs-surveys/popest/tables/2010-2016/state/totals/nst-est2016-01.xlsx"

state\_populations <- readr::read\_csv("data/nst-est2016-01.csv")

write\_csv(data, "data/nst-est2016-01.csv")

# clean it a little

state\_populations <-

state\_populations %>%

mutate(state\_name = gsub("\\.", "", X\_\_1)) %>%

left\_join(state\_abb)

# compute deaths by state and as deaths per 1000 people in each state

by\_state16 <- data %>%

filter(Year == 2016) %>%

group\_by(State) %>%

tally() %>%

left\_join(state\_abb, by = c('State' = 'state\_abb')) %>%

filter(!is.na(state\_name)) %>%

left\_join(state\_populations) %>%

mutate(per\_n\_people = (n / `2016`) \* 1000000)