Data visualization

Graphs for one categorical & one continuous variable

Jenine Harris
Brown School

foundations in public health
biostatistics

Import and clean the data

```
# import nhanes data and open tidyverse
nhanes.2012 <- read.csv(file = "~/Box/teaching/Teaching/Fall2020/data/nh
library(package = "tidyverse")

# clean gun use variable
nhanes.2012.clean <- nhanes.2012  %>%
  mutate(AUQ300 = na_if(x = AUQ300, y = 7)) %>%
  mutate(AUQ300 = recode_factor(.x = AUQ300,
                                `1` = 'Yes',
                                `2` = 'No')) %>%
  rename(gun.use = AUQ300)

#check data cleaning
summary(object = nhanes.2012.clean$gun.use)
```

```
##  Yes   No NA's
## 1613 3061 4690
```

Bar graphs for one categorical and one continuous variable

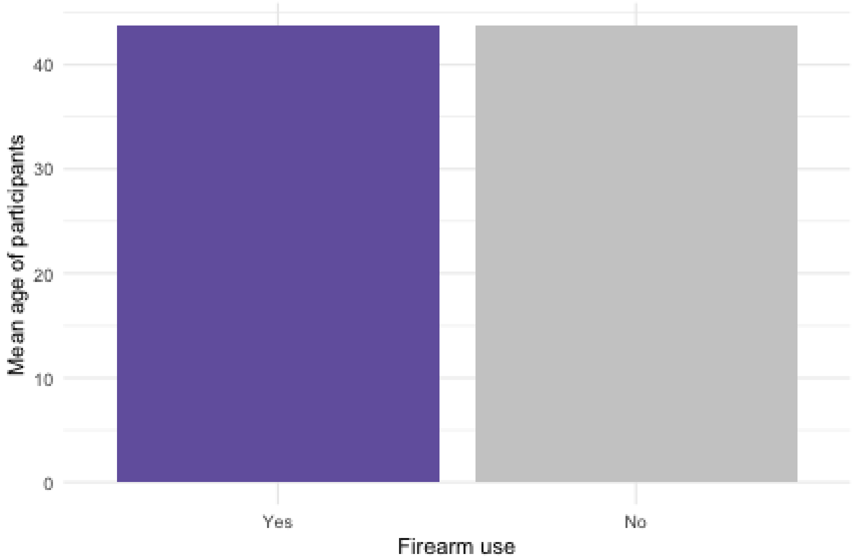
- Bar graphs can also be useful to examine how continuous measures differ across groups.
- For example, the NHANES data includes a measure of age in years.
- Do firearm users tend to be younger or older than those who do not use firearms.
- Age is measured in years which is not *truly continuous* since partial years are not included, but the underlying concept is a continuous one, with age spanning across a continuum of years rather than being broken up into categories.

Data management

- Age in years is measured along a continuum while firearm use is categorical with two categories.
- A bar graph could show two bars for gun use (Yes and No) with the height of each bar based on the mean or median age of gun users or gun non-users.
- Summary statistics in a bar graph by adding `stat = "summary"` to the `geom_bar()` layer.
- Once summary is specified, the layer also needs to know which summary statistic to use.
    - Adding `fun = mean` will result in the mean of the `y =` variable from the aesthetics, which, in this case, is `RIDAGEYR` for age.

```r
# bar graph with means for bar height
bar.gun.use.age <- nhanes.2012.clean %>%
  drop_na(gun.use) %>%
  ggplot(aes(x = gun.use, y = RIDAGEYR)) +
  geom_bar(aes(fill = gun.use), stat = "summary", fun = mean) +
  theme_minimal() +
  labs(x = "Firearm use", y = "Mean age of participants") +
  scale_fill_manual(values = c("#7463AC", "gray80"),
                    guide = FALSE)
bar.gun.use.age
```

Examining the bar plot



- There is not much of a difference in the mean age of those who have used a firearm and those who have not used a firearm.

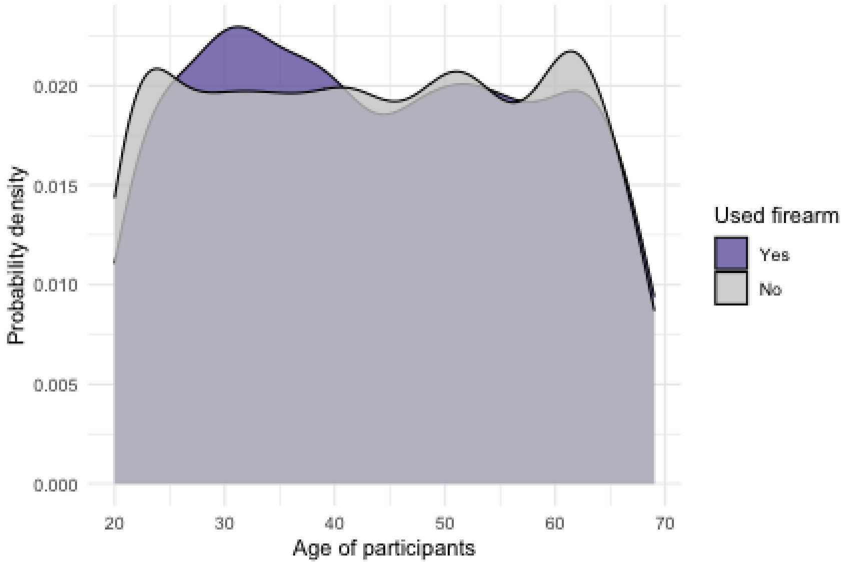
- Both groups were just under 45 years old as a mean.

Was the mean the best statistic?

- The mean is only useful when the data are normally distributed.
- Check the distribution of age for people who do and do not use firearms with a density plot.

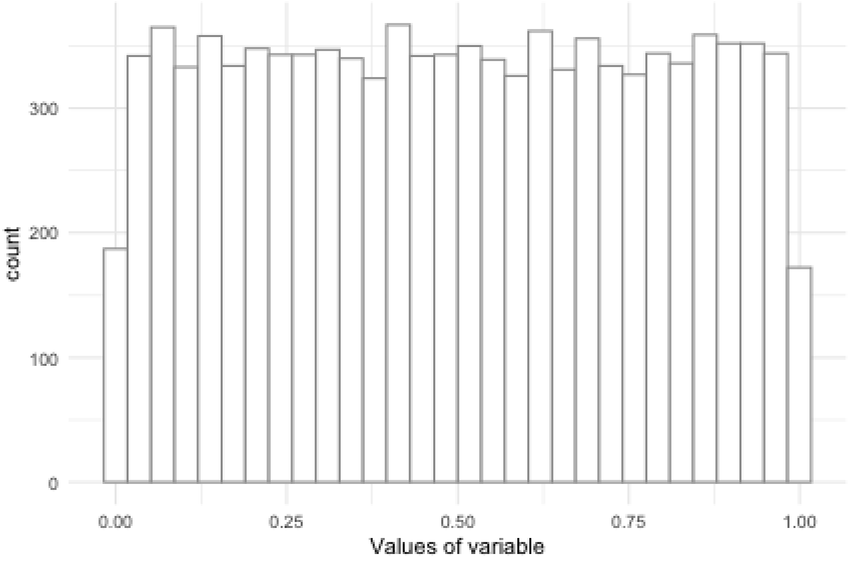
```
# density plots of age by firearm use category
dens.gun.use.age <- nhanes.2012.clean %>%
  drop_na(gun.use) %>%
  ggplot(aes(x = RIDAGEYR)) +
  geom_density(aes(fill = gun.use), alpha = .8) +
  theme_minimal() +
  labs(x = "Age of participants", y = "Probability density") +
  scale_fill_manual(values = c("#7463AC", "gray80"),
                    name = "Used firearm")
dens.gun.use.age
```

Check the distribution



The uniform distribution

- The distributions definitely did not look normal, nor skewed.
- This graph looks more like a **uniform distribution** than any of the other options.
- A perfect uniform distribution has the same frequency for each value of the variable.
- Essentially, it looks like a rectangle.



Adding the median to the plot

- Since the distribution was not normally distributed, Leslie suggested they use the median instead of the mean.

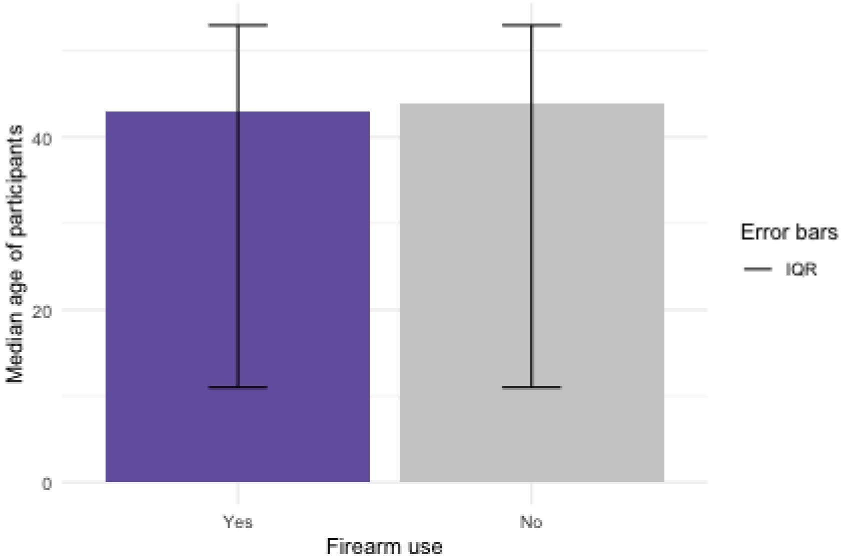
```
# bar graph with median for bar height
bar.gun.use.age.md <- nhanes.2012.clean %>%
  drop_na(gun.use) %>%
  ggplot(aes(x = gun.use, y = RIDAGEYR)) +
  geom_bar(aes(fill = gun.use), stat = "summary", fun = median) +
  theme_minimal() +
  labs(x = "Firearm use", y = "Median age of participants") +
  scale_fill_manual(values = c("#7463AC", "gray80"),
                    guide = FALSE)
bar.gun.use.age.md
```

Adding measures of spread to the plot

- Measures of central tendency tend to be reported with measures of spread.
- Spread can be added to a plot using `geom_errorbar()` layer to add standard deviations or IQR.

```
# bar graph with median for bar height and error bars
gun.use.age.md.err <- nhanes.2012.clean %>%
  drop_na(gun.use) %>%
  group_by(gun.use) %>%                                          # speci
  summarize(central = median(RIDAGEYR),
            lowIQR = quantile(nhanes.2012.clean$RIDAGEYR)[2],
            hiIQR = quantile(nhanes.2012.clean$RIDAGEYR)[4]) %>%
  ggplot(aes(x = gun.use, y = central)) +                        # use c
  geom_col(aes(fill = gun.use)) +
  geom_errorbar(aes(ymin = lowIQR,                               # lower
                    ymax = hiIQR,                                # upper
                    linetype = "IQR"),
                width = .2) +                                    # width
  theme_minimal() +
  labs(x = "Firearm use", y = "Median age of participants") +
  scale_fill_manual(values = c("#7463AC", "gray80"),
                       guide = FALSE) +
  scale_linetype_manual(values = 1, name = "Error bars")
gun.use.age.md.err
```

Examine the plot



Examine another data source

- Use FBI data to determine if there is a difference in the mean number of gun homicides per year by gun type.
- Import and summarize the **fbi_deaths_2016_ch3.csv** data set.

```
# import FBI data
fbi.deaths <- read.csv(file = "/Users/harrisj/Box/teaching/Teaching/Fall

# review the data
summary(object = fbi.deaths)
```

```
##       X                 X2012              X2013              X2014
##  Length:18          Min.   :    8.00   Min.   :    2.00   Min.   :    7.0
##  Class  :character  1st Qu.:   87.75   1st Qu.:   87.25   1st Qu.:   75.5
##  Mode   :character  Median :  304.00   Median :  296.50   Median :  261.0
##                     Mean   : 1926.28   Mean   : 1831.11   Mean   : 1825.1
##                     3rd Qu.: 1403.50   3rd Qu.: 1330.00   3rd Qu.: 1414.2
##                     Max.   :12888.00   Max.   :12253.00   Max.   :12270.0
##      X2015              X2016
##  Min.   :    1.00   Min.   :    1.0
##  1st Qu.:   87.75   1st Qu.:  100.2
##  Median :  265.00   Median :  318.0
##  Mean   : 2071.00   Mean   : 2285.8
##  3rd Qu.: 1410.00   3rd Qu.: 1428.8
##  Max.   :13750.00   Max.   :15070.0
```

Make a long data set from a wide data set

- Each year is a variable in this data frame and each observation was a type of weapon.
- Change the data from wide to long with long data having a variable called `year` specifying the year.

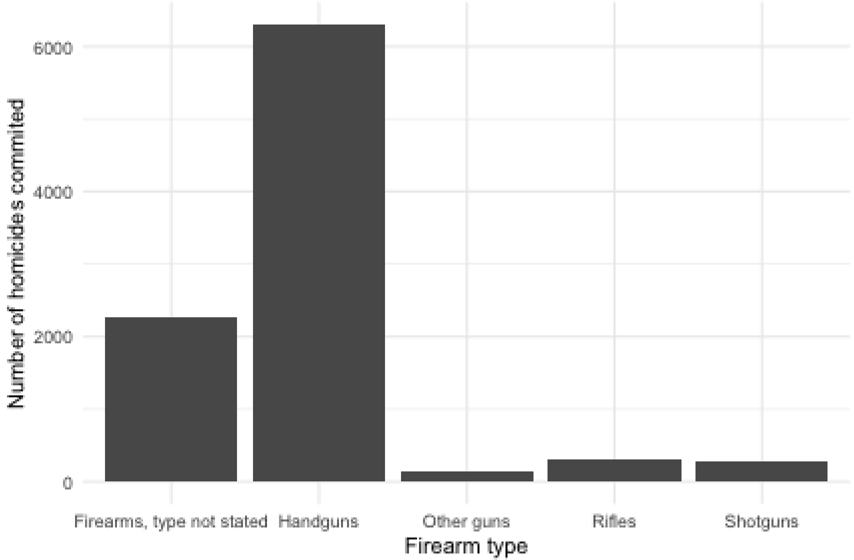
```
# make a long data frame
fbi.deaths.cleaned <- fbi.deaths %>%
  slice(3:7) %>%                                                  # sele
  gather(key = year, value = number, X2012,
         X2013, X2014, X2015, X2016) %>%                          # turn
  mutate(year, year = substr(x = year, start = 2, stop = 5)) %>%  # remo
  rename(weapons = X)

summary(object = fbi.deaths.cleaned)
```

```
##    weapons              year               number    
##  Length:25          Length:25          Min.   :  93  
##  Class :character   Class :character   1st Qu.: 258  
##  Mode  :character   Mode  :character   Median : 308  
##                                        Mean   :1858  
##                                        3rd Qu.:2502  
##                                        Max.   :7105  
```

Bar plot

```
# plot number of homicides by gun type
bar.homicide.gun <- fbi.deaths.cleaned %>%
  ggplot(aes(x = weapons, y = number)) +
  geom_bar(stat = "summary", fun = mean) +
  theme_minimal() +
  labs(x = "Firearm type", y = "Number of homicides commited")
bar.homicide.gun
```

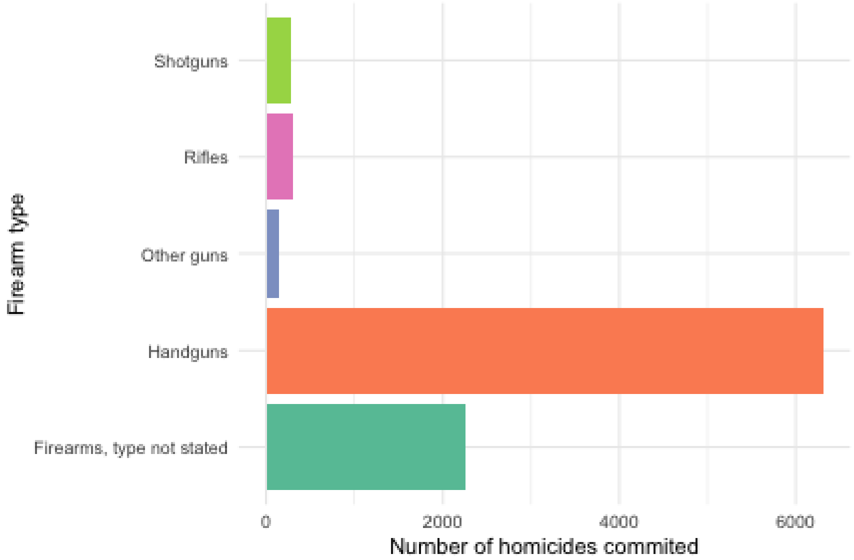


Flipping the bars

- It might be easier to read this bar graph if it were flipped since some of the bar labels are complicated.
- Flip the coordinates by adding a `coord_flip()` layer.
- Add some color to the bars using `scale_fill_brewer()`, which has a number of built in color schemes (including many that are colorblind friendly) that are directly from the Color Brewer 2.0 website.
- Use the *Set2* palette by adding `palette = "Set2"`.

```r
# flip the coordinates for better reading
# removed unnecessary legend
bar.homicide.gun <- fbi.deaths.cleaned %>%
  ggplot(aes(x = weapons, y = number)) +
  geom_bar(aes(fill = weapons), stat = "summary", fun = mean) +
  theme_minimal() +
  labs(x = "Firearm type", y = "Number of homicides commited") +
  coord_flip() +
  scale_fill_brewer(palette = "Set2", guide = FALSE)
bar.homicide.gun
```

Examine the flipped plot



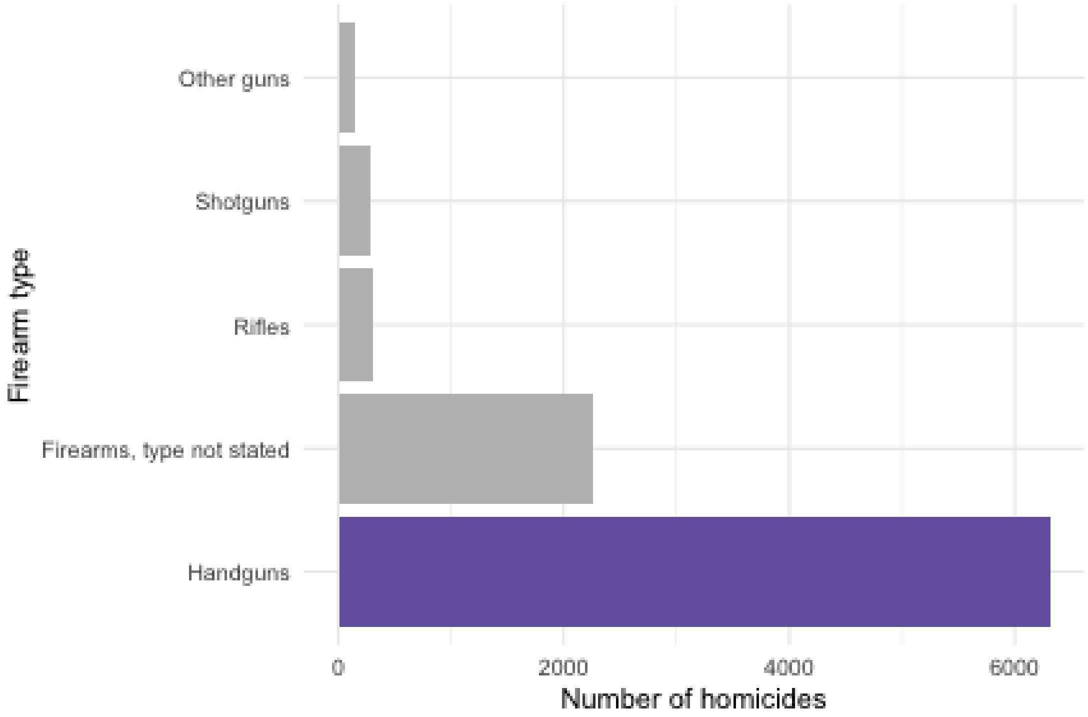
Using order and color to highlight a group

- Try changing the order of the bars so that the bars were in order by length.
- Use `reorder()` to order the bars from largest to smallest by the value of the number variable.
  - Type `reorder()` in the `aes()` as part of the `x =` argument.
  - Then, within the parentheses, add the variable to be put in order and the variable that should be used to decide the order, like this: `reorder(weapons, -number)`.
  - This means the factor `weapons` will be placed in order based on the numeric `number` variable.
- Pick colors for specific bars to highlight a category.

Code for using order and color

```
# highlight handguns using color
bar.homicide.gun <- fbi.deaths.cleaned %>%
  ggplot(aes(x = reorder(x = weapons, X = -number), y = number)) +
  geom_bar(aes(fill = weapons), stat = "summary", fun = mean) +
  theme_minimal() +
  labs(x = "Firearm type", y = "Number of homicides") +
  coord_flip() +
  scale_fill_manual(values = c("Handguns" = "#7463AC",
                               "Firearms, type not stated" = "gray",
                               "Rifles" = "gray",
                               "Shotguns" = "gray",
                               "Other guns" = "gray"), guide=FALSE)
bar.homicide.gun
```

Examine the graph

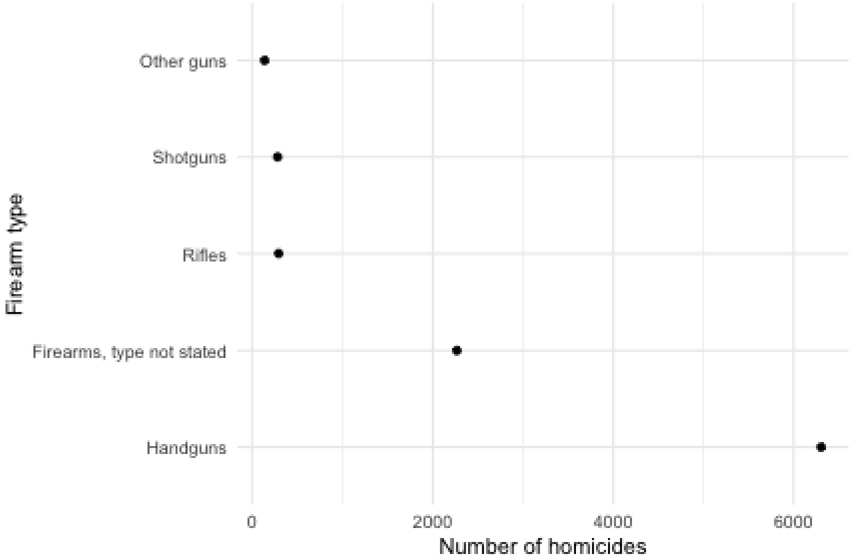


Make a point graph

- The same data could be displayed with a single point rather than a bar by changing the `geom_bar()` layer to a `geom_point()` layer.

```
# gun deaths by gun type
# highlight handguns using color
point.homicide.gun <- fbi.deaths.cleaned %>%
  ggplot(aes(x = reorder(x = weapons, X = -number), y = number)) +
  geom_point(aes(fill = weapons), stat = "summary", fun = mean) +
  theme_minimal() +
  labs(x = "Firearm type", y = "Number of homicides") +
  coord_flip() +
  scale_fill_manual(values = c("Handguns" = "#7463AC",
                               "Firearms, type not stated" = "gray",
                               "Rifles" = "gray",
                               "Shotguns" = "gray",
                               "Other guns" = "gray"), guide=FALSE)
point.homicide.gun
```

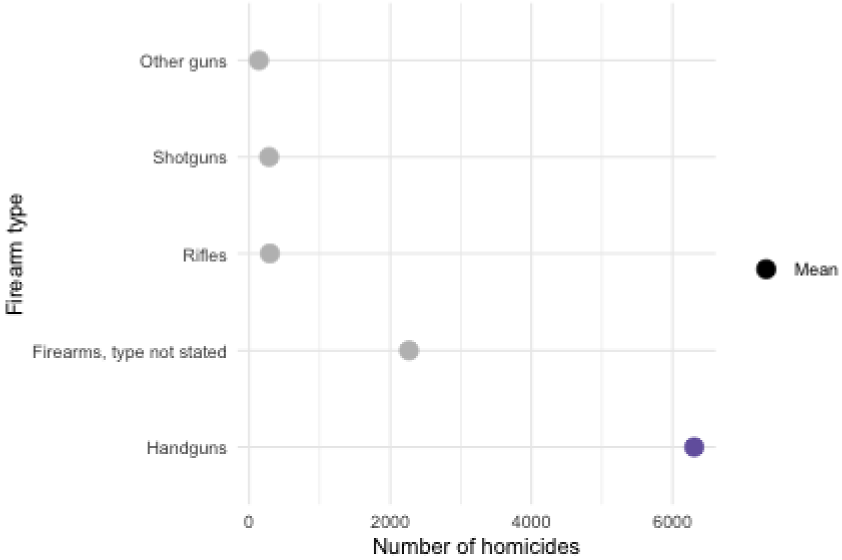
Interpret the point graph



Add color to the point graph

```
# change fill to color add size to geom_point
point.homicide.gun <- fbi.deaths.cleaned %>%
  ggplot(aes(x = reorder(x = weapons, X = -number), y = number)) +
  geom_point(aes(color = weapons, size = "Mean"), stat = "summary", fun
  theme_minimal() +
  labs(x = "Firearm type", y = "Number of homicides") +
  coord_flip() +
  scale_color_manual(values = c("Handguns" = "#7463AC",
                               "Firearms, type not stated" = "gray",
                               "Rifles" = "gray",
                               "Shotguns" = "gray",
                               "Other guns" = "gray"), guide=FALSE) +
  scale_size_manual(values = 4, name = "")
point.homicide.gun
```

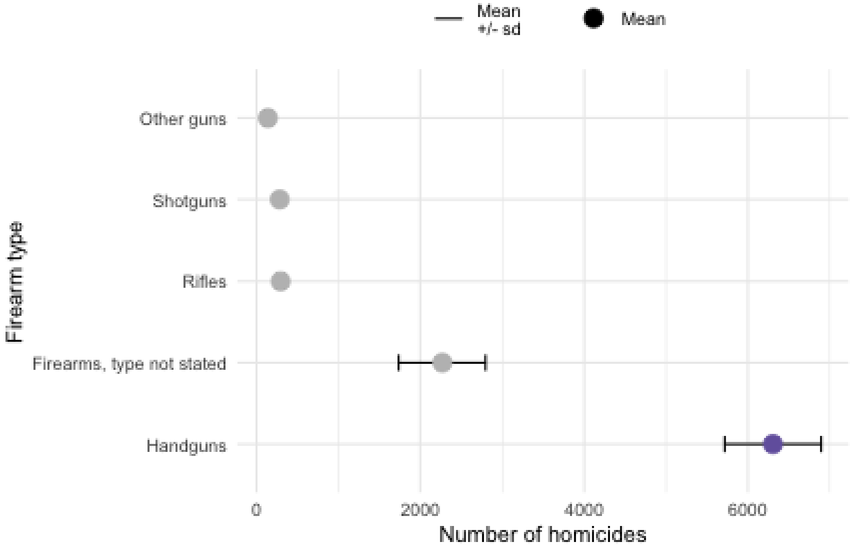
Interpret the point graph



Adding error bars to the point graph

```
# add error bars
point.homicide.gun <- fbi.deaths.cleaned %>%
  group_by(weapons) %>%
  summarize(central = mean(x = number),
            spread = sd(x = number)) %>%
  ggplot(aes(x = reorder(x = weapons, X = -central),
             y = central)) +
  geom_errorbar(aes(ymin = central-spread,
                    ymax = central+spread,
                    linetype = "Mean\n+/- sd"),
                width = .2) +
  geom_point(aes(color = weapons, size = "Mean"), stat = "identity") +
  theme_minimal() +
  labs(x = "Firearm type",
       y = "Number of homicides") +
  coord_flip() +
  scale_color_manual(values = c("Handguns" = "#7463AC",
                               "Firearms, type not stated" = "gray",
                               "Rifles" = "gray",
                               "Shotguns" = "gray",
                               "Other guns" = "gray"), guide=FALSE) +
  scale_linetype_manual(values = 1, name = "") +
  scale_size_manual(values = 4, name = "") +
  theme(legend.position = "top")
```

Interpreting the graph



Boxplots

- Change the `geom_` layer to make another option, a boxplot.

```r
# change to boxplot
box.homicide.gun <- fbi.deaths.cleaned %>%
  ggplot(aes(x = reorder(x = weapons, X = -number),
             y = number)) +
  geom_boxplot(aes(color = weapons)) +
  theme_minimal() +
  labs(x = "Firearm type", y = "Number of homicides") +
  coord_flip() +
  scale_color_manual(values = c("Handguns" = "#7463AC",
                               "Firearms, type not stated" = "gray",
                               "Rifles" = "gray",
                               "Shotguns" = "gray",
                               "Other guns" = "gray"), guide=FALSE)
box.homicide.gun
```

Using color with boxplots

- Boxplot color is specified with `fill =` in order to fill the boxplots instead of outlining them.

```r
# fix color for boxplots
box.homicide.gun <- fbi.deaths.cleaned %>%
  ggplot(aes(x = reorder(x = weapons, X = -number),
             y = number)) +
  geom_boxplot(aes(fill = weapons)) +
  theme_minimal() +
  labs(x = "Firearm type", y = "Number of homicides") +
  coord_flip() +
  scale_fill_manual(values = c("Handguns" = "#7463AC",
                               "Firearms, type not stated" = "gray",
                               "Rifles" = "gray",
                               "Shotguns" = "gray",
                               "Other guns" = "gray"), guide=FALSE)
box.homicide.gun
```

Add data points to boxplot

- To see the underlying data with a boxplot, add a `geom_jitter()` layer to the `ggplot()`.
- Use `alpha = .8` to make the color a little less bright so that it was easier to see the data points.

```r
# fix color for boxplots
box.homicide.gun <- fbi.deaths.cleaned %>%
  ggplot(aes(x = reorder(x = weapons, X = -number),
             y = number)) +
  geom_boxplot(aes(fill = weapons), alpha = .8) +
  geom_jitter() +
  theme_minimal() +
  labs(x = "Firearm type", y = "Number of homicides") +
  coord_flip() +
  scale_fill_manual(values = c("Handguns" = "#7463AC",
                               "Firearms, type not stated" = "gray",
                               "Rifles" = "gray",
                               "Shotguns" = "gray",
                               "Other guns" = "gray"), guide=FALSE)
box.homicide.gun
```

Violin plots

- Violin plots are somewhere between boxplots and density plots, typically used to look at the distribution of continuous data within categories.

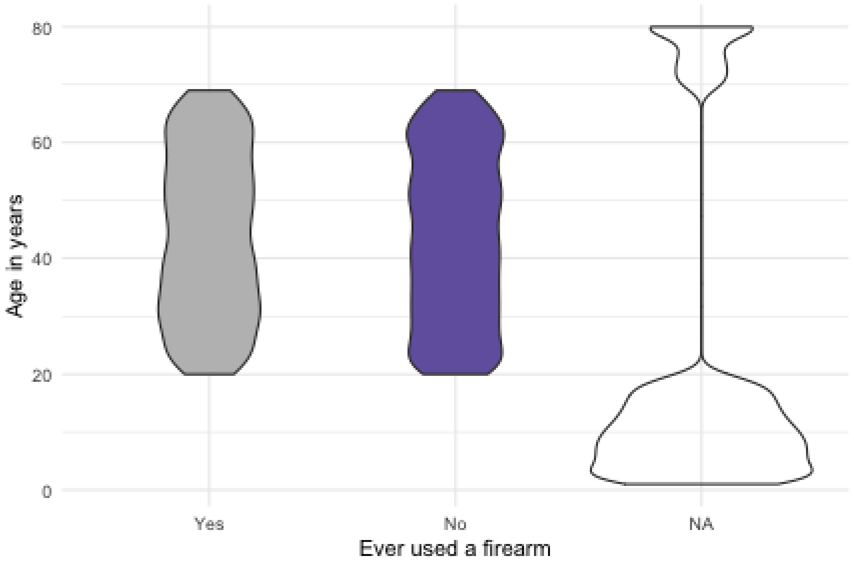
- From the code above, remove the `geom_jitter`, and change `geom_boxplot` to `geom_violin`.

```
# violin plot
violin.homicide.gun <- fbi.deaths.cleaned %>%
  ggplot(aes(x = reorder(x = weapons, X = -number),
             y = number)) +
  geom_violin(aes(fill = weapons)) +
  theme_minimal() +
  labs(x = "Firearm type", y = "Number of homicides") +
  coord_flip() +
  scale_fill_manual(values = c('gray', "#7463AC", 'gray',
                               'gray', 'gray'), guide=FALSE)
violin.homicide.gun
```

Another violin plot

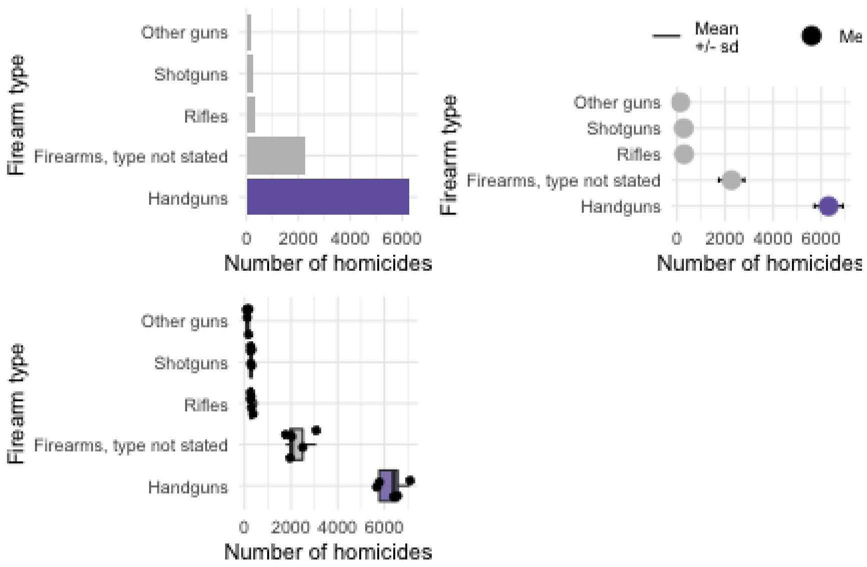
- Use the `nhanes.2012.clean` data from above to look at whether the distributions of age are the same for gun users as a better example of a violin plot.

```
# violin plot of age by sex for NHANES
nhanes.2012.clean %>%
  ggplot(aes(x = gun.use, y = RIDAGEYR)) +
  geom_violin(aes(fill = gun.use)) +
  scale_fill_manual(values = c("gray", "#7463AC"), guide = FALSE) +
  labs(y = "Age in years", x = "Ever used a firearm") +
  theme_minimal()
```



See several plots together

```
# plot all three options together
gridExtra::grid.arrange(bar.homicide.gun,
                        point.homicide.gun,
                        box.homicide.gun,
                        ncol = 2)
```


Fiream type
Other guns
Shotguns
Rifles
Firearms, type not stated
Handguns
Number of homicides
Mean +/- sd
Me
Firearm type
Other guns
Shotguns
Rifles
Firearms, type not stated
Handguns
Number of homicides
Fiream type
Other guns
Shotguns
Rifles
Firearms, type not stated
Handguns
Number of homicides