Data visualization

Graphs for two continuous variables

Jenine Harris
Brown School

Import and summarize the data

- The data are from the US Bureau of Alcohol, Tobacco, Firearms, and Explosives and quantify the number of guns of each type manufactured in the US each year 1990-2015
- Variables include:
    - Year: year data collected
    - Pistols: number of pistols manufactured
    - Revolvers: number of revolvers manufactured
    - Rifles: number of rifles manufactured
    - Shotguns: number of shotguns manufactured
    - Total.firearms: total number of firearms manufactured (sum of four types)

```
# bring in the data
guns.manu <- read.csv(file = "/Users/harrisj/Box/teaching/Teaching/Fall2
summary(object = guns.manu)
```

```
##       Year          Pistols           Revolvers          Rifles
##  Min.   :1990   Min.   : 677434   Min.   :274399   Min.   : 883482
##  1st Qu.:1996   1st Qu.: 989508   1st Qu.:338616   1st Qu.:1321474
##  Median :2002   Median :1297072   Median :464440   Median :1470890
##  Mean   :2002   Mean   :1693216   Mean   :476020   Mean   :1796195
##  3rd Qu.:2009   3rd Qu.:2071096   3rd Qu.:561637   3rd Qu.:1810749
##  Max.   :2015   Max.   :4441726   Max.   :885259   Max.   :3979568
##     Shotguns       Total.firearms
##  Min.   : 630663   Min.   : 2962002
##  1st Qu.: 735563   1st Qu.: 3585090
```

Data management: Turning wide data to long data

- Each firearm type is included as a different variable.
- Instead gun type could be a factor variable with each type of gun as a category of the factor.
    - The restructured data would have 5 entries per year, one each for the four firearm types and the total firearms.
- Use `gather()` to make this wide data into long data.

```
# open the tidyverse
library(package = "tidyverse")

# make wide data long
guns.manu.cleaned <- guns.manu %>%
  gather(key = gun.type, value = num.guns, Pistols,
         Revolvers, Rifles, Shotguns, Total.firearms) %>%
  mutate(gun.type = as.factor(gun.type))
```

Check the new data shape

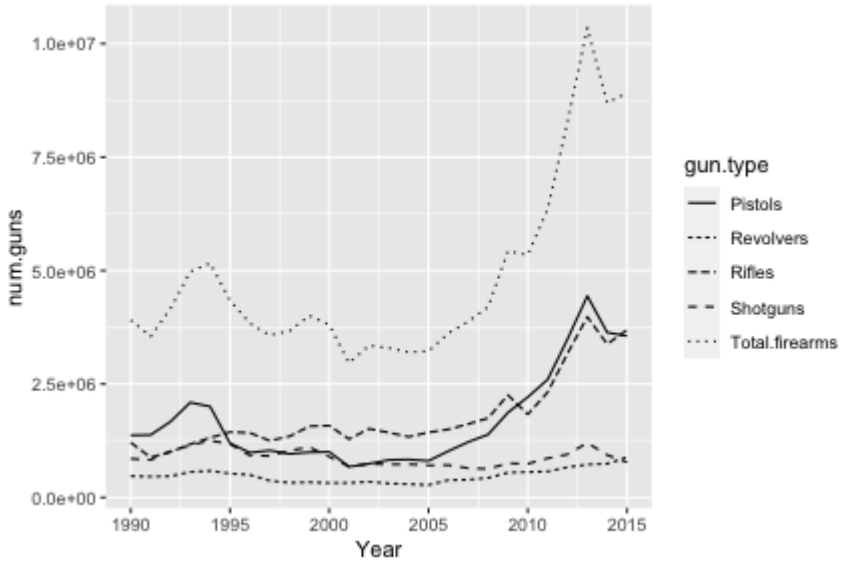
```
# check the data
summary(object = guns.manu.cleaned)
```

```
##       Year                 gun.type     num.guns
##  Min.   :1990   Pistols        :26   Min.   :  274399
##  1st Qu.:1996   Revolvers      :26   1st Qu.:  741792
##  Median :2002   Rifles         :26   Median : 1199178
##  Mean   :2002   Shotguns       :26   Mean   : 1939577
##  3rd Qu.:2009   Total.firearms:26   3rd Qu.: 3119839
##  Max.   :2015                       Max.   :10349648
```

Line graphs

- Create a line graph by piping the new data frame into `ggplot()` command with `geom_line()`.
- To use a different line for each gun type, add `linetype = gun.type` to the `aes()`.

```
# plot it
line.gun.manu <- guns.manu.cleaned %>%
  ggplot(aes(x = Year, y = num.guns)) +
  geom_line(aes(linetype = gun.type))
line.gun.manu
```



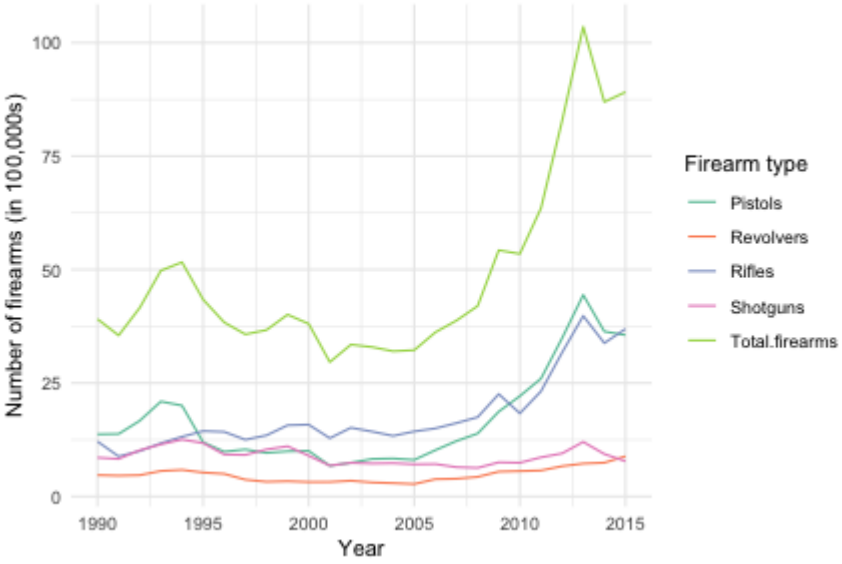
Format the graph for clarity

A list of things to change:

- convert the scientific notation on the y-axis to regular numbers
- add a theme to get rid of the gray background
- make better labels for the axes and legend
- add color to the lines to help differentiate between gun types

Code for formatting changes

```
# update the y-axis, theme, line color, labels
line.gun.manu <- guns.manu.cleaned %>%
  ggplot(aes(x = Year, y = num.guns/100000)) +
  geom_line(aes(color = gun.type)) +
  theme_minimal() +
  labs(y = "Number of firearms (in 100,000s)") +
  scale_color_brewer(palette = "Set2",
                     name = "Firearm type")
line.gun.manu
```



Combine categories

- Handguns are the most common type of gun used in homicides.
- Pistols and revolvers are both types of handguns.
- Combine the pistols and revolvers into a single type of gun before converting from wide to long.

```
# make a handguns category that is pistols + revolvers
# remove pistols and revolvers from graph
guns.manu.cleaned <- guns.manu %>%
  mutate(Handguns = Pistols + Revolvers) %>%
  gather(key = gun.type, value = num.guns, Pistols, Revolvers,
         Rifles, Shotguns, Total.firearms, Handguns) %>%
  mutate(gun.type, gun.type = as.factor(gun.type)) %>%
  filter(gun.type != "Pistols" & gun.type != "Revolvers")
```

Check the data

```
# data with combined handgun category
summary(object = guns.manu.cleaned)
```

```
##       Year                  gun.type     num.guns
##  Min.   :1990   Pistols        :26   Min.   :  274399
##  1st Qu.:1996   Revolvers      :26   1st Qu.:  741792
##  Median :2002   Rifles         :26   Median : 1199178
##  Mean   :2002   Shotguns       :26   Mean   : 1939577
##  3rd Qu.:2009   Total.firearms:26   3rd Qu.: 3119839
##  Max.   :2015                       Max.   :10349648
```

Drop levels to get rid of empty groups

```
# make a handguns category that is pistols + revolvers
# remove pistols and revolvers from graph, droplevels
guns.manu.cleaned <- guns.manu %>%
  mutate(Handguns = Pistols + Revolvers) %>%
  gather(key = gun.type, value = num.guns, Pistols, Revolvers,
         Rifles, Shotguns, Total.firearms, Handguns) %>%
  mutate(gun.type, gun.type = as.factor(gun.type)) %>%
  filter(gun.type != "Pistols" & gun.type != "Revolvers") %>%
  droplevels()

# data with combined handgun category
summary(object = guns.manu.cleaned)
```

```
##       Year                gun.type     num.guns
##  Min.   :1990   Handguns       :26   Min.   :  630663
##  1st Qu.:1996   Rifles         :26   1st Qu.: 1102768
##  Median :2002   Shotguns       :26   Median : 1542610
##  Mean   :2002   Total.firearms:26   Mean   : 2424471
##  3rd Qu.:2009                       3rd Qu.: 3555676
##  Max.   :2015                       Max.   :10349648
```

Update the graph with data & thick lines

```
# update the line graph with new data and thicker lines
line.gun.manu <- guns.manu.cleaned %>%
  ggplot(aes(x = Year, y = num.guns/100000)) +
  geom_line(aes(color = gun.type), size = 1) +
  theme_minimal() +
  labs(y = "Number of firearms (in 100,000s)") +
  scale_color_brewer(palette = "Set2",
                     name = "Firearm type")
line.gun.manu
```

Scatterplots

- A **scatterplot** is also useful to show the relationship between two continuous variables.
- In a scatterplot, instead of connecting data points to form a line, one dot is used to represent each data point.
- There are situations where a *line graph* is more useful than a *scatterplot*:
  - (1) when the graph is showing change over time, and
  - (2) when there is not a lot of variation in the data.
- Relationships where there is no measure of time and data that include a lot of variation are better shown with a scatterplot.

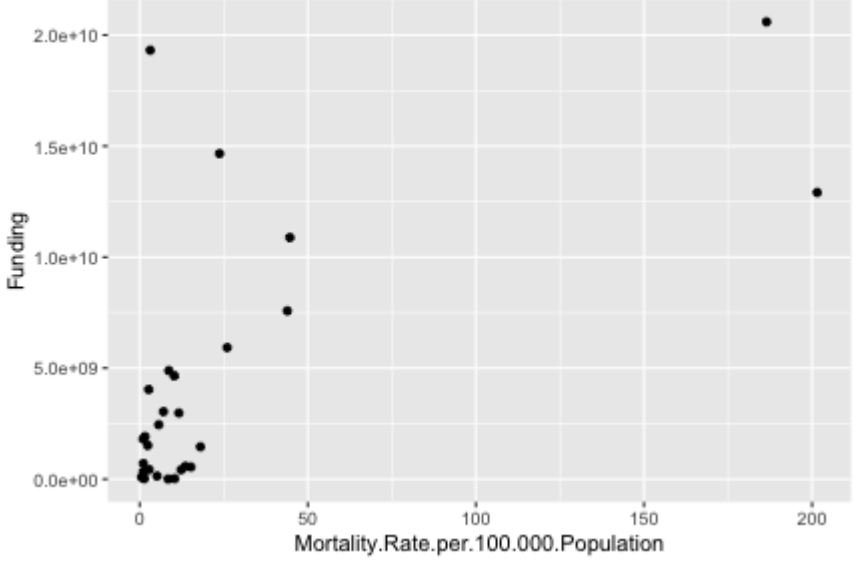
Create a scatterplot for gun manufacturing

```
# use scatterplot instead of line
scatter.gun.manu <- guns.manu.cleaned %>%
  ggplot(aes(x = Year, y = num.guns/100000)) +
  geom_point(aes(color = gun.type)) +
  theme_minimal() +
  labs(y = "Number of firearms (in 100,000s)") +
  scale_color_brewer(palette = "Set2",
                     name = "Firearm type")
scatter.gun.manu
```

Funding for research and mortality

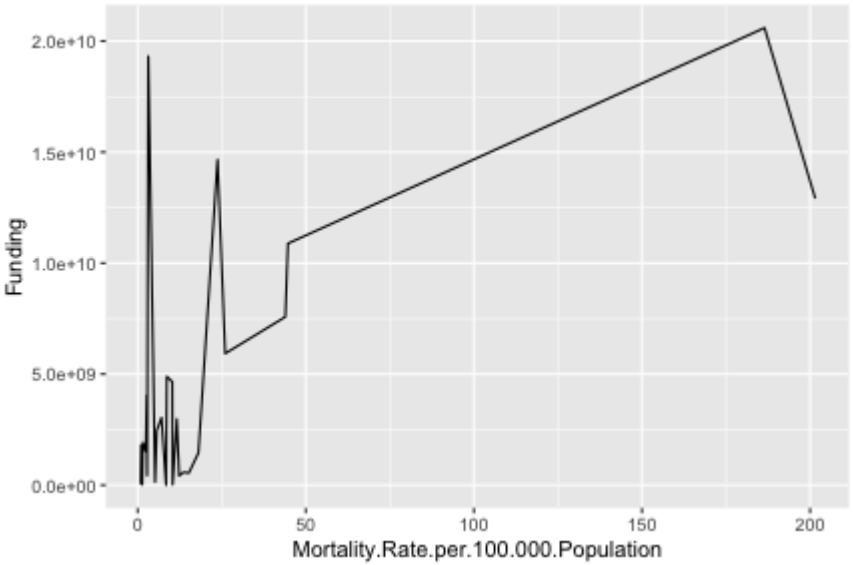
```
# bring in the data
research.funding <- read.csv(file = "/Users/harrisj/Box/teaching/Teachin

# scatterplot of gun research by funding
scatter.gun.funding <- research.funding %>%
  ggplot(aes(x = Mortality.Rate.per.100.000.Population, y = Funding))+
    geom_point()
scatter.gun.funding
```



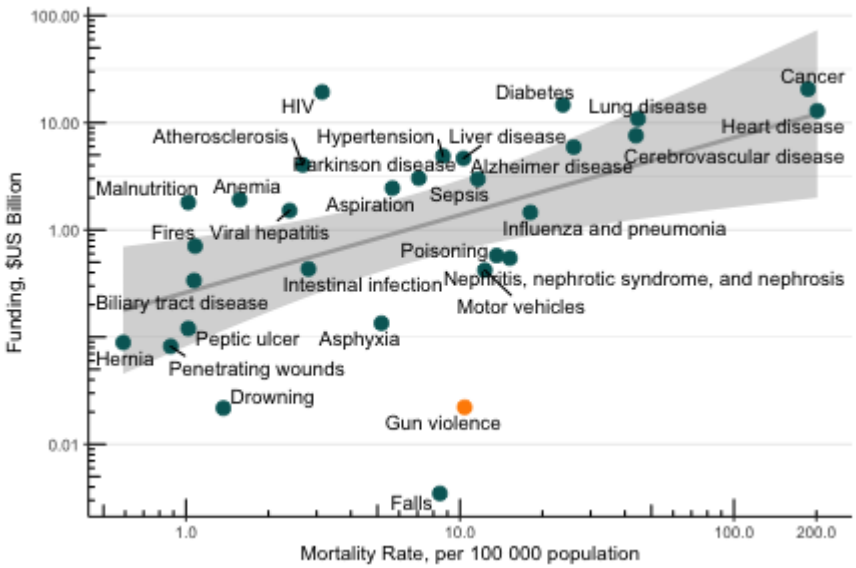
What would a line graph look like?

```
# Line graph of gun research by funding
scatter.gun.funding <- research.funding %>%
  ggplot(aes(x = Mortality.Rate.per.100.000.Population, y = Funding))+
    geom_line()
scatter.gun.funding
```



Formatting the scatterplot

- An examination of the original figure these data came from showed a different scale on the x and y axes (Source: Stark DE, Shah NH. Funding and publication of research on gun violence and other leading causes of death. Jama. 2017 Jan 3;317(1):84-5.)

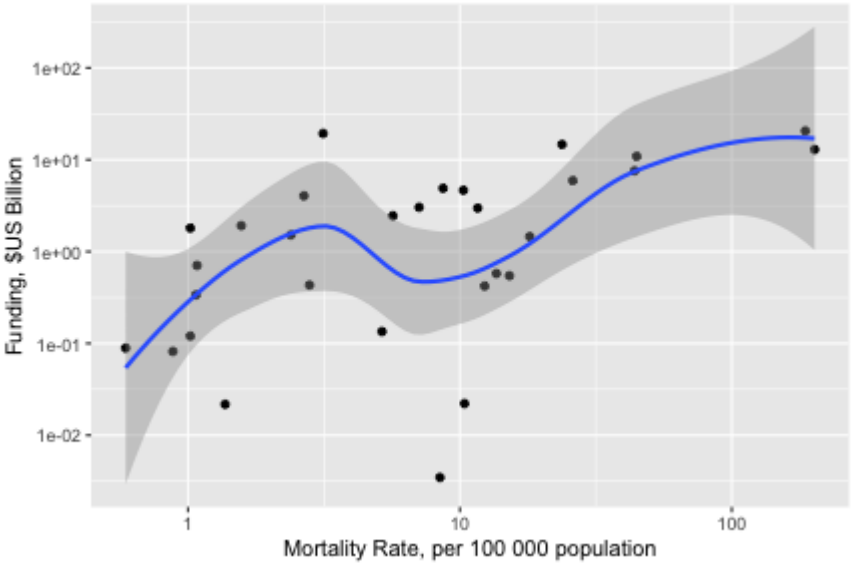


Changing the axis scale & adding trend lines

- Add layers for scaling with `scale_x_log10()` and `scale_y_log10()` for the axes and a layer with `stat_smooth()` for a smooth line through the dots.
- Change the axes values to match the figure, the y-axis layer appears to be in billions, so the funding variable should be divided by billions to make this true.

```
# scatterplot of gun research by funding
# with axes showing a natural log scale
scatter.gun.funding <- research.funding %>%
  ggplot(aes(x = Mortality.Rate.per.100.000.Population, y = Funding/1000
  geom_point() +
  stat_smooth() +
  scale_x_log10() +
  scale_y_log10() +
  labs(y = "Funding, $US Billion",
       x = "Mortality Rate, per 100 000 population")
scatter.gun.funding
```

Check the graph



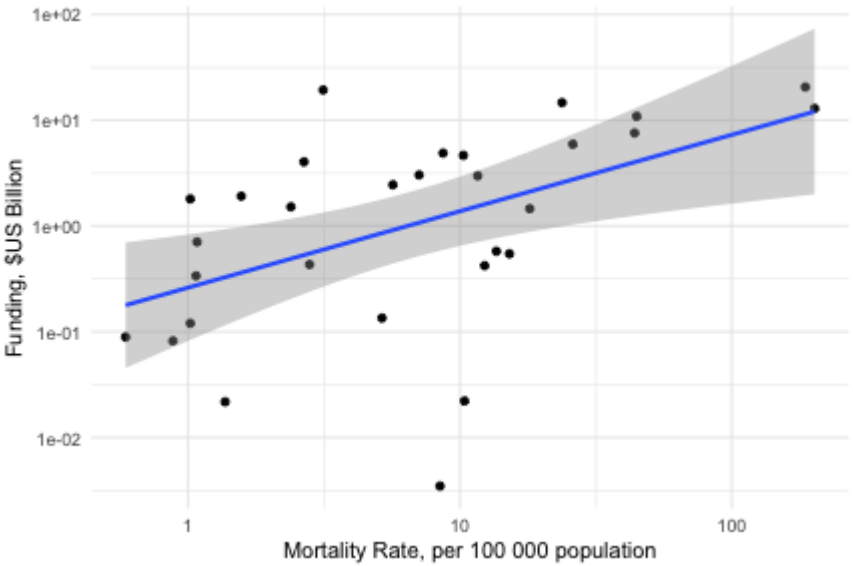
Fix the smoothing line

- The line does not look right, use the `method = lm` or linear model option to add a straight line with the `stat_smooth()` command.

```
#scatterplot of gun research by funding
#with axes showing a natural log scale
scatter.gun.funding <- research.funding %>%
  ggplot(aes(x = Mortality.Rate.per.100.000.Population, y = Funding/1000
  geom_point() +
  stat_smooth(method = "lm") +
  scale_x_log10() +
  scale_y_log10() +
  labs(y = "Funding, $US Billion",
       x = "Mortality Rate, per 100 000 population") +
  theme_minimal()
scatter.gun.funding
```

Check the plot

```
## `geom_smooth()` using formula 'y ~ x'
```

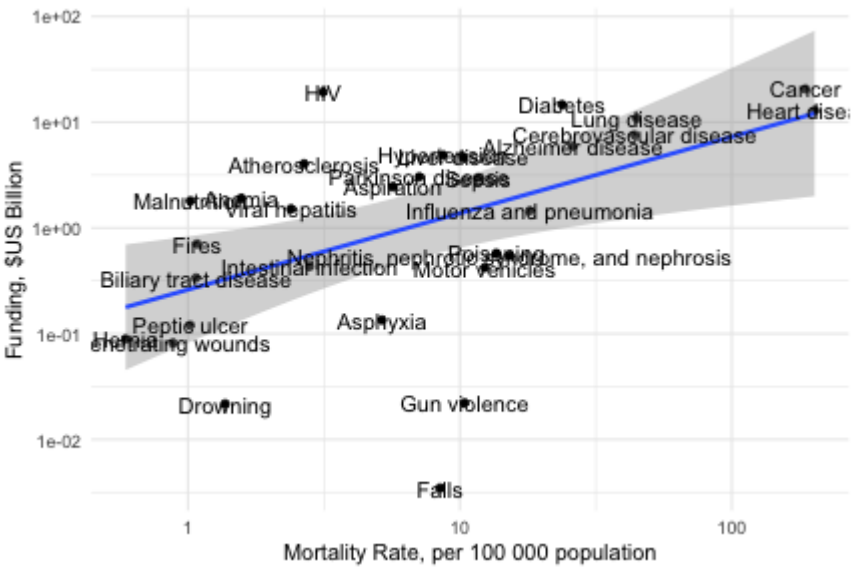


Add labels to the points

- Use a geom_text layer to add labels.

```
# fancy graph
scatter.gun.funding.lab <- research.funding %>%
  ggplot(aes(x = Mortality.Rate.per.100.000.Population, y = Funding/1000
  geom_point() +
  stat_smooth(method = "lm") +
  scale_x_log10() +
  scale_y_log10() +
  labs(y = "Funding, $US Billion",
       x = "Mortality Rate, per 100 000 population") +
  theme_minimal() +
  geom_text(aes(label = Cause.of.Death))
scatter.gun.funding.lab
```

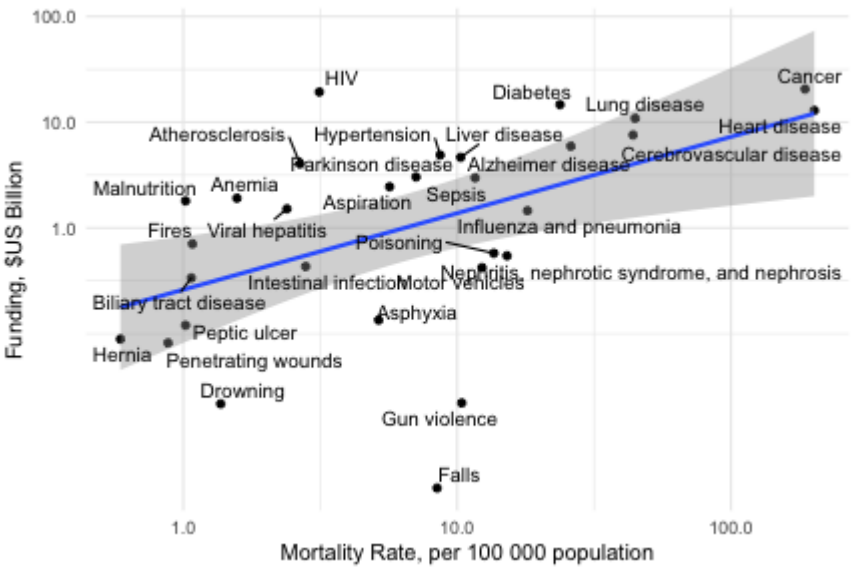
Check point labels



Fix the axis labels some more

```
# fancy graph with better labels
scatter.gun.funding.lab <- research.funding %>%
  ggplot(aes(x = Mortality.Rate.per.100.000.Population,
             y = Funding/1000000000)) +
  geom_point() +
  stat_smooth(method = "lm") +
  scale_x_log10(breaks = c(1,10,100), labels = comma) +
  scale_y_log10(breaks = c(1,10,100), labels = comma) +
  labs(y = "Funding, $US Billion",
       x = "Mortality Rate, per 100 000 population") +
  theme_minimal() +
  ggrepel::geom_text_repel(aes(label = Cause.of.Death), size = 3.5)
scatter.gun.funding.lab
```

Check the axis labels



Review all the plots together

```
# show graph types
gridExtra::grid.arrange(line.gun.manu,
                        scatter.gun.funding,
                        nrow = 2)
```

```
## `geom_smooth()` using formula 'y ~ x'
```