Notes - Class 1

Simon G. Brauer

Tuesday, October 06, 2015

Class 1: October 6th, 2015

Websites for the class

http://socviz.github.io/soc880/

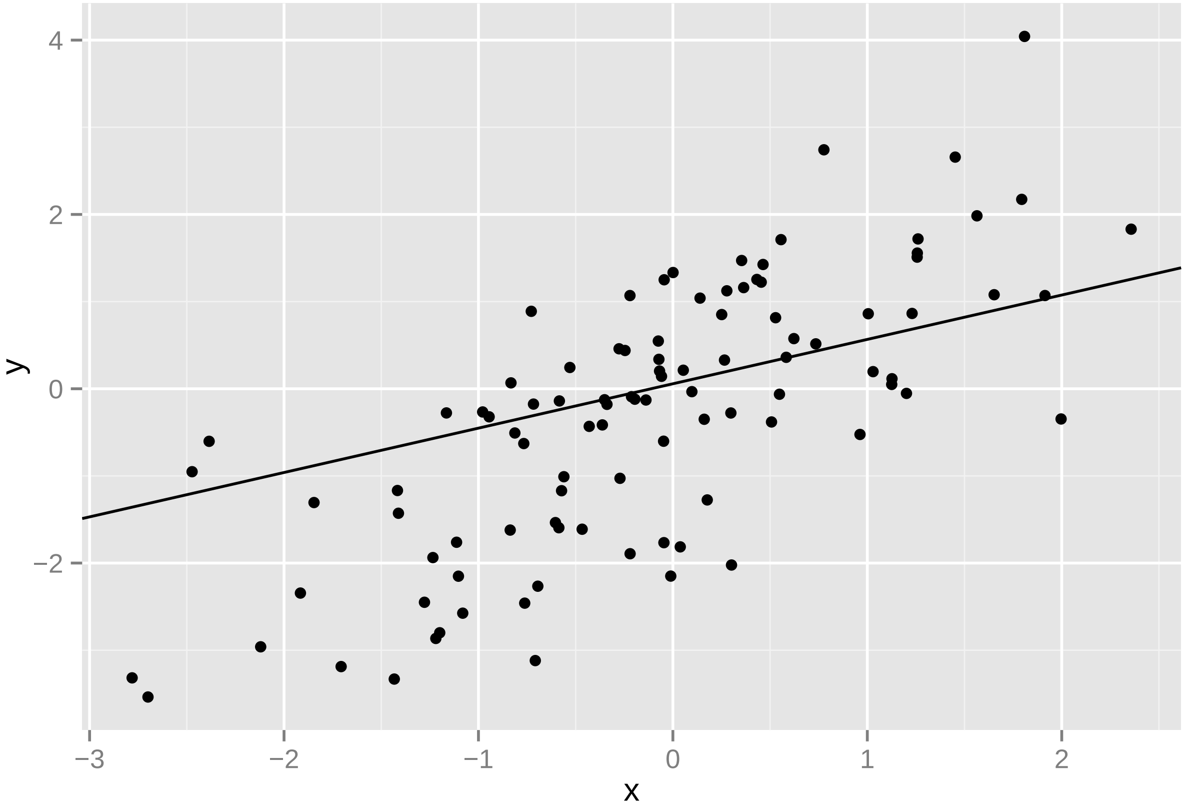
http://github.com/socviz/soc880

First sample code (sort of)

```
data <- c(1, 1, 4, 1, 1, 4, 1)
data
```

```
## [1] 1 1 4 1 1 4 1
```

```
x <- rnorm(n = 100)
y <- x + rnorm(n = 100)
```



Details on R

R has a working directory

```
getwd()
```

```
## [1] "C:/Users/Simon/OneDrive/Documents/Github/Data-Visualization"
```

How Kieran organizes his working directory…

1. data folder
2. doc folder
3. figures folder

- r file in root

everything in R is an object…

```
(my_numbers <- c(1, 2, 3, 1, 3,5, 25))
```

```
## [1]  1  2  3  1  3  5 25
```

```
summary(my_numbers)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   1.500   3.000   5.714   4.000  25.000
```

every object has a class

```
class(my_numbers)
```

```
## [1] "numeric"
```

```
#even functions have classes
class(summary)
```

```
## [1] "function"
```

Objects can hold the output of functions

```
(my_summary <- summary(my_numbers))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   1.500   3.000   5.714   4.000  25.000
```

```
class(my_summary)
```

```
## [1] "summaryDefault" "table"
```

Examples in R

```
my_numbers * 2
```

```
## [1]  2  4  6  2  6 10 50
```

```
table(my_numbers)
```

```
## my_numbers
##  1  2  3  5 25
##  2  1  2  1  1
```

```
sd(my_numbers)
```

```
## [1] 8.616153
```

you can get information in different ways

```
str(my_numbers)
```

```
##  num [1:7] 1 2 3 1 3 5 25
```

```
str(my_summary)
```

```
## Classes 'summaryDefault', 'table'  Named num [1:6] 1 1.5 3 5.71 4 ...
##   ..- attr(*, "names")= chr [1:6] "Min." "1st Qu." "Median" "Mean" ...
```

```
str(summary)
```

```
## function (object, ...)
```

why are charts sometimes bad?

*Pure aesthetics: presentation is fine but ugly

*Figure shows you less than it should or could

*Figure shows more than it should

*Figure can be misleading or misrepresent the data, accidentally or purposefully

Consider the actual amount of information you're presenting and conside simplifying to emphasizeit (instead the background, coloring, etc.)

Look at William Cleveland's work on presenting data

His thoughts:

- It's difficult to compare without common baselines
- People misjudge areas like crazy
- Acute angles are underestimated, obtuse are overestimated
- You can't distinguish data points if they are overlapping
- Colors are good for distinguishing categories but not for continuous variables
- Difficult at interpreting curves because our perception is influenced by horizontal space

Let's get some data...

```
library(devtools)
```

```
## Warning: package 'devtools' was built under R version 3.2.2

## WARNING: Rtools is required to build R packages, but is not currently installed.
##
## Please download and install Rtools 3.3 from http://cran.r-project.org/bin/windows/Rtools/ and then ru
```

```
gapminder.url <- "https://raw.githubusercontent.com/socviz/soc880/master/data/gapminder.csv"

## What type of object is this?
class(gapminder.url)
```

```
## [1] "character"
```

```
data <- read.csv(url(gapminder.url))
class(data)
```

```
## [1] "data.frame"
```

```
str(data)
```

```
## 'data.frame':    1704 obs. of  6 variables:
##  $ country  : Factor w/ 142 levels "Afghanistan",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ continent: Factor w/ 5 levels "Africa","Americas",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ year     : int  1952 1957 1962 1967 1972 1977 1982 1987 1992 1997 ...
##  $ lifeExp  : num  43.1 45.7 48.3 51.4 54.5 ...
##  $ pop      : num  9279525 10270856 11000948 12760499 14760787 ...
##  $ gdpPercap: num  2449 3014 2551 3247 4183 ...
```

```
head(data)
```

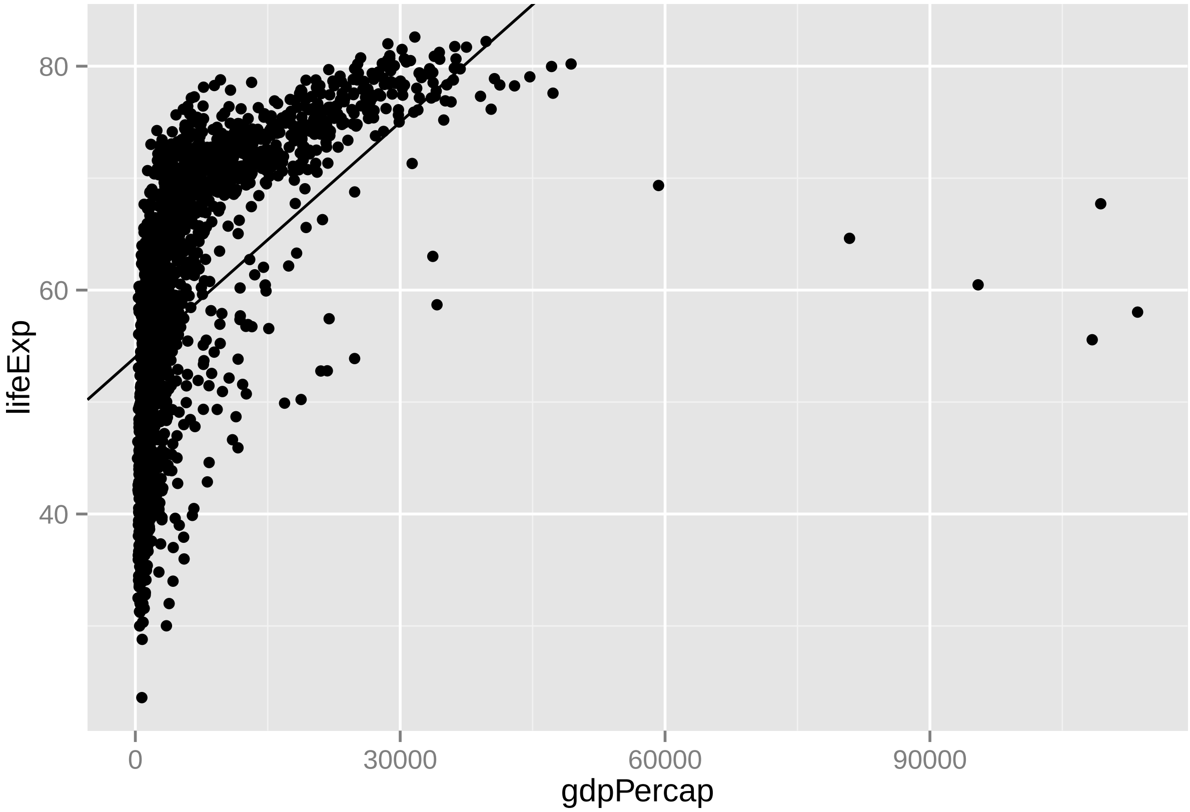
```
##   country continent year lifeExp      pop gdpPercap
## 1 Algeria    Africa 1952  43.077  9279525  2449.008
## 2 Algeria    Africa 1957  45.685 10270856  3013.976
## 3 Algeria    Africa 1962  48.303 11000948  2550.817
## 4 Algeria    Africa 1967  51.407 12760499  3246.992
## 5 Algeria    Africa 1972  54.518 14760787  4182.664
## 6 Algeria    Africa 1977  58.014 17152804  4910.417
```

let's plot it!

```r
p <- ggplot(data = data, aes(x = gdpPercap, y = lifeExp))
lm(data = data, formula = lifeExp ~ gdpPercap)
```

```
##
## Call:
## lm(formula = lifeExp ~ gdpPercap, data = data)
##
## Coefficients:
## (Intercept)    gdpPercap
##   5.396e+01    7.649e-04
```

```r
p + geom_point() + geom_abline(intercept = 54, slope = 0.0007)
```

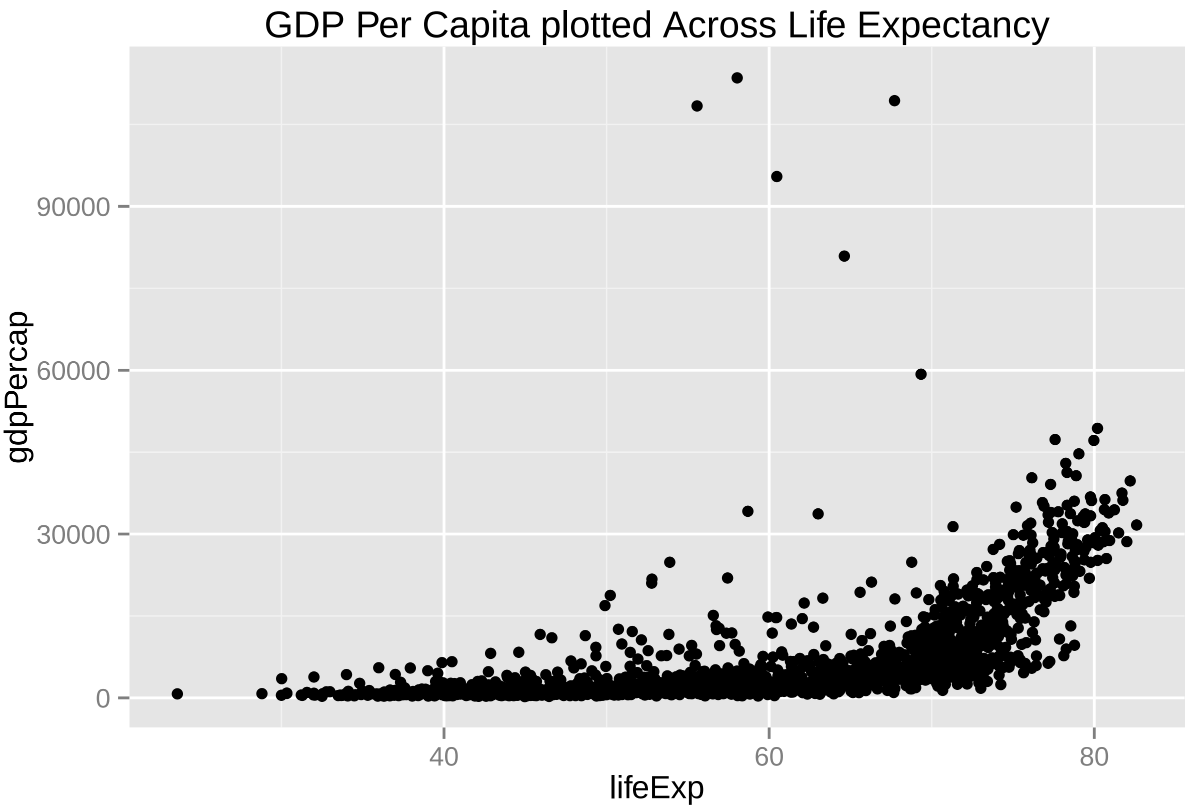


Homework for next time

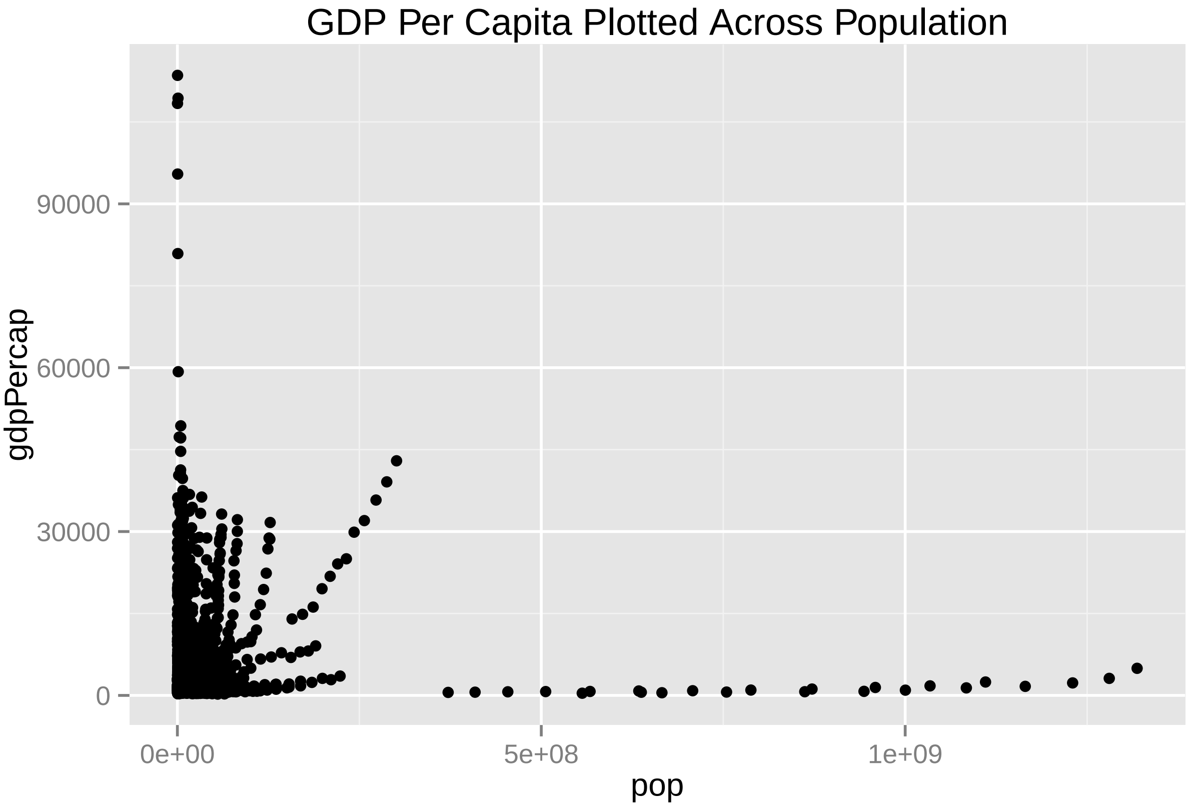
1. Create an RMarkdown file for your work
2. Look again at the data
3. Put lifeExp on the x-axis and gdpPercap on the y-axis
4. plot pop on the x-axis and gdpPercap on the y-axis
5. Plot year on the x-axis and any continuous variable on the y-axis

Homework

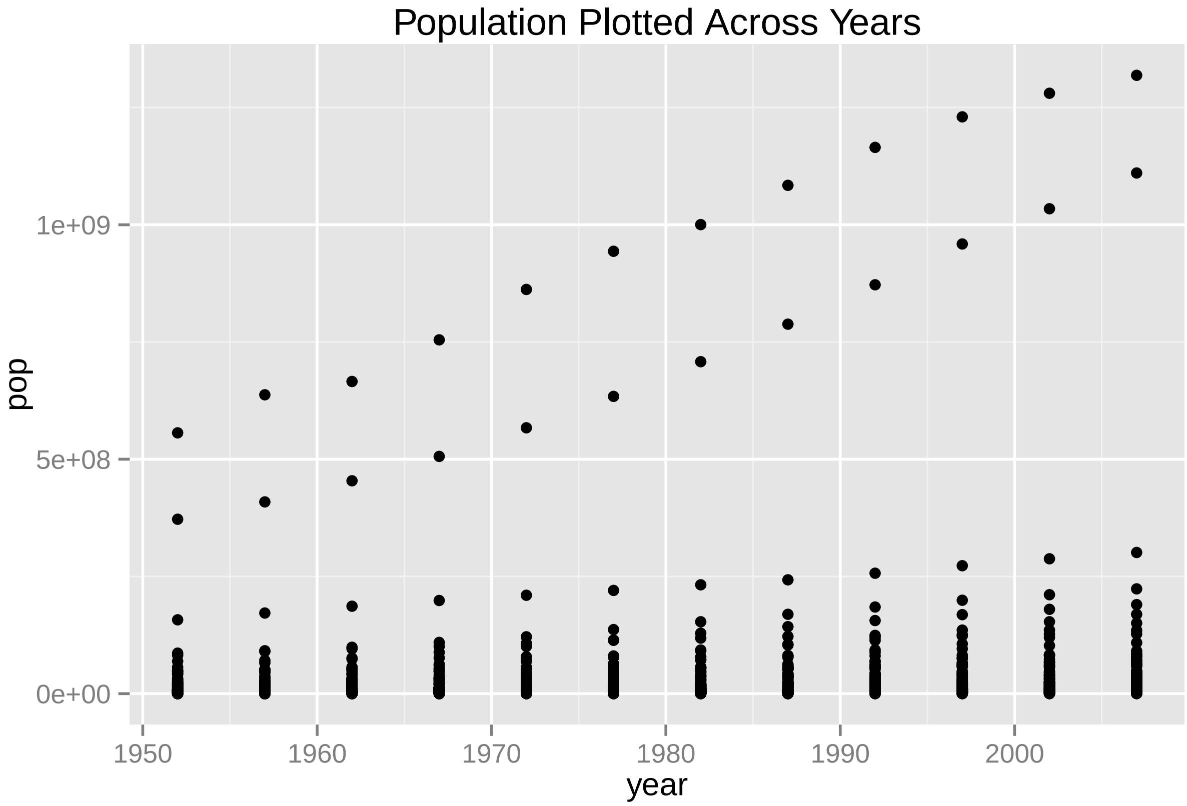
```
##Put lifeEXP on the x-axis and gdpPercap on the y-axis
ggplot(data = data, aes(x = lifeExp, y = gdpPercap)) + geom_point() + labs(title = "GDP Per Capita plot
```



```
##Put pop on the x-axis and gdpPercap on the y-axis
ggplot(data = data, aes(x = pop, y = gdpPercap)) + geom_point() + labs(title = "GDP Per Capita Plotted
```



```
##Plot year on the x-axis and any continuous variable on the y-axis
ggplot(data = data, aes(x = year, y = pop)) + geom_point() + labs(title = "Population Plotted Across Yea
```


Population Plotted Across Years
pop
1e+09
5e+08
0e+00
1950
1960
1970
1980
1990
2000
year