Pick four - comparing trends in population over time

## Purpose

The purpose of this report is to compare the population trends for four countries of your choosing. In addition, this serves as an example of literate programming. Literate programming is a way to document how you performed your analysis. It serves as a guide to others (and your future self) how to reproduce your work.

## Required Packages

library(here)

## here() starts at /Users/claudiuforgaci/Projects/repro-r-workshop-livecode

library(ggplot2)

## Data

Always add as many details as possible about your data including where it came from, how it was processed, licensing, and where it can be accessed.

* Gapminder data [available here](http://www.gapminder.org/data/). [Gapminder data is licensed CC-BY 3.0](https://docs.google.com/document/pub?id=1POd-pBMc5vDXAmxrpGjPLaCSDSWuxX6FLQgq5DhlUhM#h.ul2gu2-uwathz).
* Processed data via [@jennybc](https://github.com/jennybc), [R package available here](https://github.com/jennybc/gapminder). The [data-raw](https://github.com/jennybc/gapminder/tree/master/data-raw) sub-directory reveals the journey from Gapminder.org’s Excel workbooks to increasingly clean and tidy data.

**Read in data**: To read in the data, make sure this file is in the same directory/folder as the gapminderDataFiveYear.tsv file.

path <- here("data", "gapminderDataFiveYear.tsv")  
gapMinder <- read.delim(path)  
  
### Check data   
head(gapMinder, n = 10L) #First 10 lines of dataset

## country year pop continent lifeExp gdpPercap  
## 1 Afghanistan 1952 8425333 Asia 28.801 779.4453  
## 2 Afghanistan 1957 9240934 Asia 30.332 820.8530  
## 3 Afghanistan 1962 10267083 Asia 31.997 853.1007  
## 4 Afghanistan 1967 11537966 Asia 34.020 836.1971  
## 5 Afghanistan 1972 13079460 Asia 36.088 739.9811  
## 6 Afghanistan 1977 14880372 Asia 38.438 786.1134  
## 7 Afghanistan 1982 12881816 Asia 39.854 978.0114  
## 8 Afghanistan 1987 13867957 Asia 40.822 852.3959  
## 9 Afghanistan 1992 16317921 Asia 41.674 649.3414  
## 10 Afghanistan 1997 22227415 Asia 41.763 635.3414

dim(gapMinder) #number of rows and columns in data set

## [1] 1704 6

You can see what countries are available by looking at the how many unique categories are in the country column of the gapMinder dataset.

levels(factor(gapMinder$country))

### Pick Four Countries

Now pick four countries that you are interested in. Just replace with the countries name below.

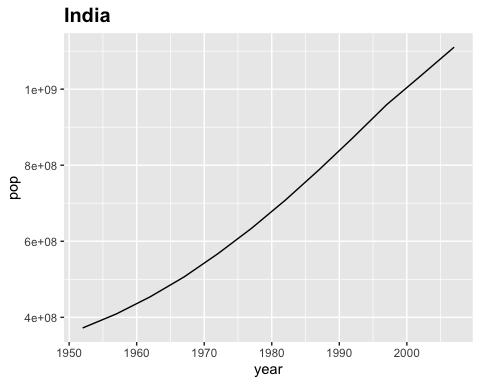
countryName1 <- "India"  
countryName2 <- "United States"  
countryName3 <- "Nigeria"  
countryName4 <- "Germany"

## Individual countries

### Country One

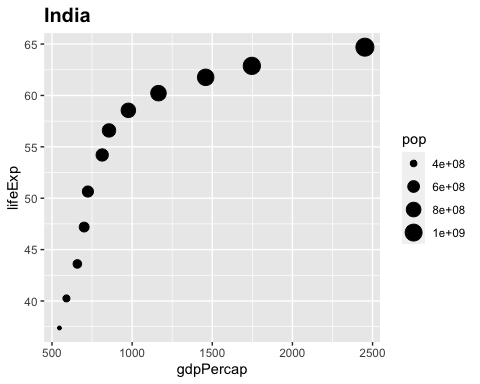
We want to look at how population changes over time for the first country.

country1 <- subset(gapMinder, country == countryName1)  
  
ggplot(country1, aes(year, pop)) +   
 geom\_path() +  
 ggtitle(countryName1) +  
 theme(plot.title = element\_text(size = 15, face = "bold"))



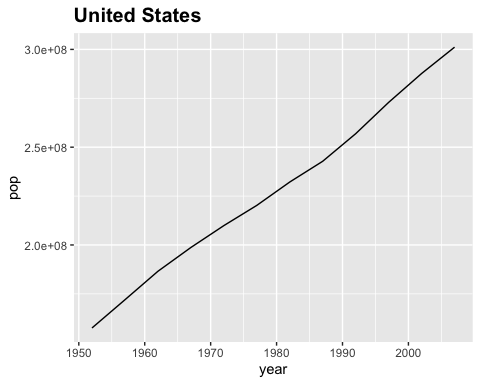
This second graph is looking at the correlation between life expectancy (lifeExp) and GDP per person (gdpPercap). The size of the circles on the plot represents total population.

ggplot(country1, aes(gdpPercap, lifeExp, size = pop)) +   
 geom\_point() +  
 ggtitle(countryName1) +  
 theme(plot.title = element\_text(size = 15, face = "bold"))

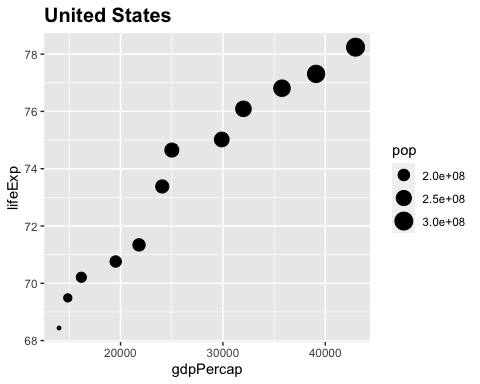


### Country Two

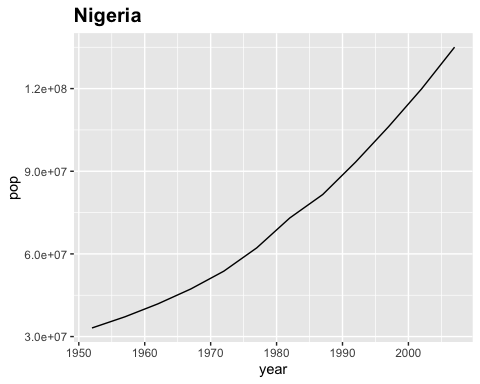
We will do this for each country. Since the code is very similar, we will omit viewing it below by adding the named parameter echo=FALSE (TRUE is the default):



**Notes**: In a real report you can add information about the results of the analysis you are performing. That way your code, analysis, questions, and results are all in one place.

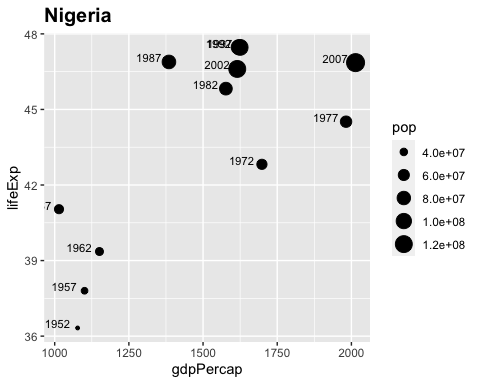


### Country Three

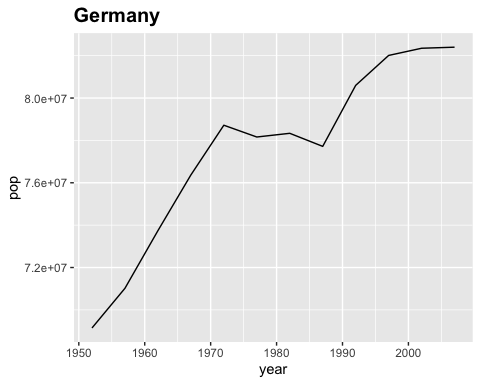


**Notes** Maybe a country has an unusual distribution and we want to label the graph with the year. We added label = year to the first line of the code below. To display the text we also added the geom\_text(hjust = 1.3, vjust = 0, size = 3) option.

ggplot(country3, aes(gdpPercap, lifeExp, size = pop, label = year)) +   
 geom\_point() +  
 geom\_text(hjust = 1.3, vjust = 0, size = 3) +  
 ggtitle(countryName3) +  
 theme(plot.title = element\_text(size = 15, face = "bold"))

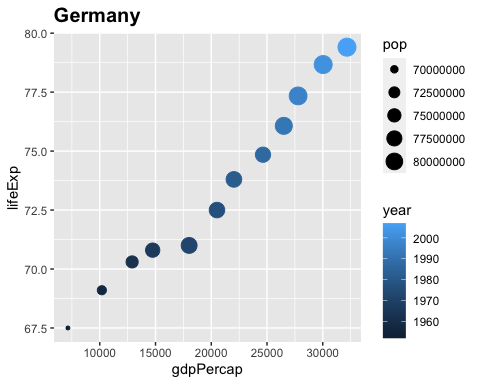


### Country Four



**Notes**: Or we can try out labeling the year by adding color.

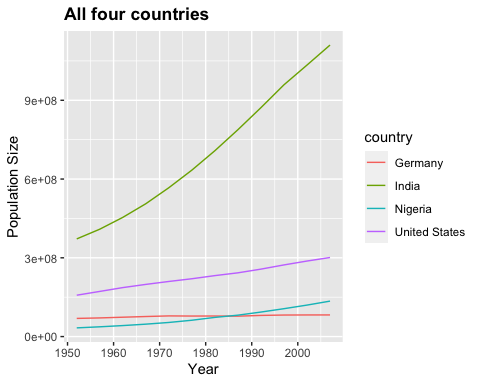
ggplot(country4, aes(gdpPercap, lifeExp, size = pop, color = year)) +   
 geom\_point() +  
 ggtitle(countryName4) +  
 theme(plot.title = element\_text(size=15, face = "bold"))



## All four countries

Let’s add all four countries together and to see how they compare.

#Add subsetted data together  
allCountries <- rbind(country1, country2, country3, country4)  
  
#Notice the code for this is similar to when we are just looking at one country  
#just with the added color option  
ggplot(allCountries, aes(year, pop, color=country)) +   
 geom\_path() +  
 xlab("Year") + ylab("Population Size") +  
 ggtitle("All four countries") +  
 theme(plot.title = element\_text(lineheight=.8, face = "bold"))

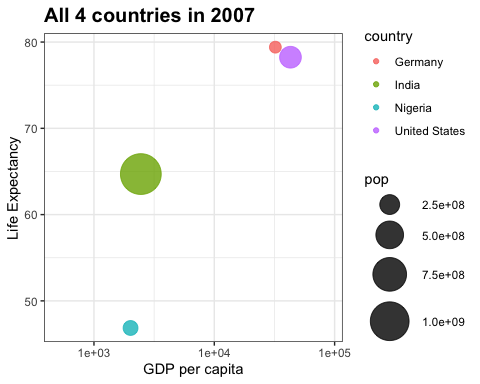


What about what is occurring in a particular year? You can change the year by changing the code in the year == 2007 section. To look at what years are possible use allCountries$year.

allCountries$year

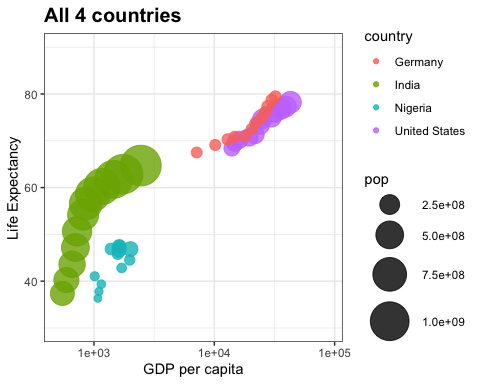
## [1] 1952 1957 1962 1967 1972 1977 1982 1987 1992 1997 2002 2007 1952 1957 1962  
## [16] 1967 1972 1977 1982 1987 1992 1997 2002 2007 1952 1957 1962 1967 1972 1977  
## [31] 1982 1987 1992 1997 2002 2007 1952 1957 1962 1967 1972 1977 1982 1987 1992  
## [46] 1997 2002 2007

yr <- 2007  
ggplot(subset(allCountries, year == yr),  
 aes(x = gdpPercap, y = lifeExp, color = country, size = pop)) +   
 scale\_x\_log10(limits = c(500, 90000)) +   
 geom\_point(alpha = 0.8) +   
 scale\_size\_area(max\_size = 14) +  
 theme\_bw() + # black grid on white background  
 xlab("GDP per capita") + ylab("Life Expectancy") +  
 ggtitle(paste("All 4 countries in", yr)) +  
 theme(plot.title = element\_text(size = 15, face = "bold"))



You can plot all the years at once also!

ggplot(allCountries,  
 aes(x = gdpPercap, y = lifeExp, color = country, size = pop)) +   
 scale\_x\_log10(limits = c(500, 90000)) +  
 ylim(c(30, 90)) +  
 geom\_point(alpha = 0.8) +   
 scale\_size\_area(max\_size = 14) +  
 theme\_bw() + # black grid on white background  
 xlab("GDP per capita") + ylab("Life Expectancy") +  
 ggtitle("All 4 countries") +  
 theme(plot.title = element\_text(size = 15, face = "bold"))



## Conclusions

In a real report you can add conclusions about your analysis or future plans for the project. The best part is that if you want to change something in your report you don’t have to redo every step. You can just make the change and re-print the report.