# STT 301: Tidy data

*Shawn Santo*

*October 17, 2018*

# Introduction

**Learning objectives:**

- reading data into R
    - `read.csv`
- apply
- cleaning data
    - `tidyr`

# Raw data from Gapminder

Download the file `gm-le.csv` from D2L. Open the file to see how the data are organized. This should always be your first step before you read a data set into R.

Use `read.csv` to read the data into an object called `gapminder1` in R. Make sure to use `na.strings` to specify how missing data are indicated in the csv file. Investigate the data using `str` and `head`. You'll see a few issues. First, the data is in "wide" format with each column representing a year. We would prefer the data to be in "long" format. Second, the name of the variable containing the countries is `Life.expectancy`. Third, because names in R cannot start with a number, the columns after the first have an `X` prepended. We will deal with these in turn.

```
gapminder1 <- read.csv(file = "gm-le.csv", header = TRUE, na.strings = "")
dim(gapminder1)
str(gapminder1)
head(gapminder1)
```

The dimensions should be 260 x 218.

# Some summary statistics

Use `apply` to compute the mean, median, minimum, and maximum life expectancy for each year in `gapminder1`. You'll need to exclude the first column, and you also will need to tell R how to handle missing values.

# Wide to long format

Load the `tidyr` library (you may need to install this package - enter `install.packages("tidyr")` in your console). Use the `gather` function from the `tidyr` package to transform the data from wide to long format. Save the result as `gapminder`. Call the variable containing the years `year` and call the variable containing the life expectancies `lifeExp`. At this point the years will be represented as `X1804` and `X2001`, for example. We will fix that soon. Consult Section 6.4 of the text (notes) for details on the `gather` function. After the transformation, the data frame should contain 56420 observations with 3 variables.

## Additional cleaning

The name of the first variable, which contains the countries, should be changed to `country`. Use the `names` function in R to do this.

The years variable currently contains values such as `X1804` and `X2001`. We will use the `substr` and `as.integer` functions to strip off the `X` and then convert the resulting values to integer. (The `substr` function will be covered in detail later when we deal with string manipulations.)

First, use `gapminder$year <- substr(x = gapminder$year, start = 2, stop = 5)` to strip off the `X`. At that point the variable `year` should look good, i.e., should have the `X` removed, but will still be a character vector. So use `as.integer` to convert `year` to an integer vector.

# Graphical displays

Load the `ggplot2` library.

1. Create a histogram of life expectancy using all the data.
2. Create side by side box plots of life expectancy for the years 1900, 1910, 1920, 1930, 1940, 1950, 1960, 1970, 1980, and 1990. You'll probably want to use `subset` in the `data` argument of `ggplot`.
3. Create a line graph ( `geom_line` ) of the life expectancy against year for the United States. What do you notice?
4. Create (on the same set of axes) a line graph of life expectancy for the five most populous countries: China, India, United States, Indonesia, and Brazil. Each country's line should be a different color.