# Week 13 Programming Assignment: Iteration and Conditional Plot

## Instructor: Chien-Hao Fu

### 5/16/2019

**Due: 5/22/2019**

**Data Source**

Population data downloaded from Kaiser Family Foundation, which is based on analysis of the Census Bureau's American Community Survey (ACS). Accessed 5/12/2019 from [https://www.kff.org/other/state-indicator/distribution-by-age/](https://www.kff.org/other/state-indicator/distribution-by-age/)

**Preparation**

1. Load the package `tidyverse`

2. Install the package: `carData`. We will use the data set `UN98` from the package

3. Download the six data files (`us_percent_age_data_2013.csv~-2017.csv`)

4. Take a moment to examine the data files

**Part 1**

Objective: Practice `for` loop and `map` function

1. Import the data frame from `carData` using the following code:

```
indicators <- carData::UN98 %>% select(-region)
```

2. Examine the data frame `indicators`

3. Write a `for` loop to calculate the average of each variable

   - Set the option `na.rm` so missing values are removed from calculation
   - Save the output in the object: `ind.m`

4. Use a `map_dbl` function to replicate your results in Step 3

**Part 2**

Objectives

1. Import, clean, and merge data using a `for` loop
2. Draw a faceted plot

Key Steps

1. Import the six data files using a `for` loop and store them in a list (e.g., `pop`)

   - The following columns should be removed from the imported data frame: `Footnotes` and `Total`
   - You should also remove observations that are not actual observations
   - You should add a variable (e.g., `year`) to indicate the year for the observation
   - Tip: you may want to test your codes by first importing a file and *tidy* it. Then modify the code to be used in the `for` loop

2. Merge the 6 data frames (in the list `pop`) into a single one

3. Draw a faceted plot in which `Location` is used as the grouping variable

   - You will need to choose appropriate number of rows or columns so the faceted plots are readable
   - You may also want to add the following codes to make the axis ticks more legible

```
+  theme(axis.text.x = element_text(angle = 90, vjust = 0.5))
```

4. Export the plot with `ggsave()` and choose appropriate `width` and `length`

5. Submit the plot and your source codes for Part 1 and 2 to the course website