DSC640 WEEK 3 & 4

Cindy Herrera

12/19/2019

DSC640 WEEKS 3 & 4

EXERCISE 2.2

Datasets - Exercise 2.2 you can download them directly from this link: https://content.bellevue.edu/cst/dsc/640/datasets/ex2-2.zip

Exercise Goal:

You need to submit 3 line charts and 3 step charts using Tableau or PowerBI, Python and R using the data below (or your own datasets). You can also submit using D3, though not required. You can choose which library to use in Python or R, documentation is provided to help you decide and as you start to play around in the libraries, you will decide which you prefer.

CINDY HERRERA

Decemeber 19, 2019

```
library(readxl)
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(plotly)
```

```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(treemap)
library(formattable)
```

```
##
## Attaching package: 'formattable'

## The following object is masked from 'package:plotly':
##
##     style
```

```
library(date)
library(scales)
```

```
##
## Attaching package: 'scales'

## The following objects are masked from 'package:formattable':
##
##     comma, percent, scientific
```

```
library(ggthemes)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following object is masked from 'package:base':
##
##     date
```

Import/read data file

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    50 obs. of  2 variables:
##  $ Year      : num  1960 1961 1962 1963 1964 ...
##  $ Population: num  3028654024 3068356747 3121963107 3187471383 3253112403 ...
```

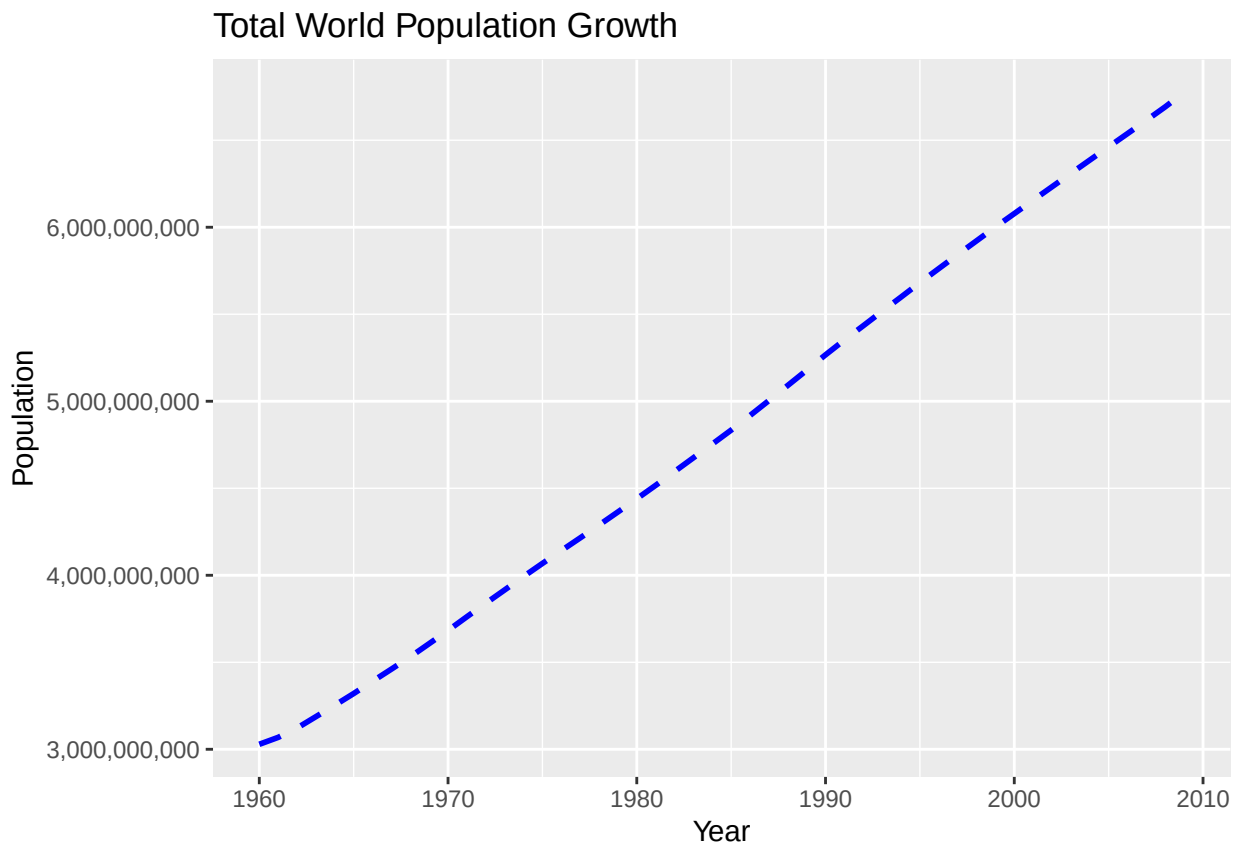
As we can see above our year and population are automatically formatted as numbers

```
summary(wp)
```

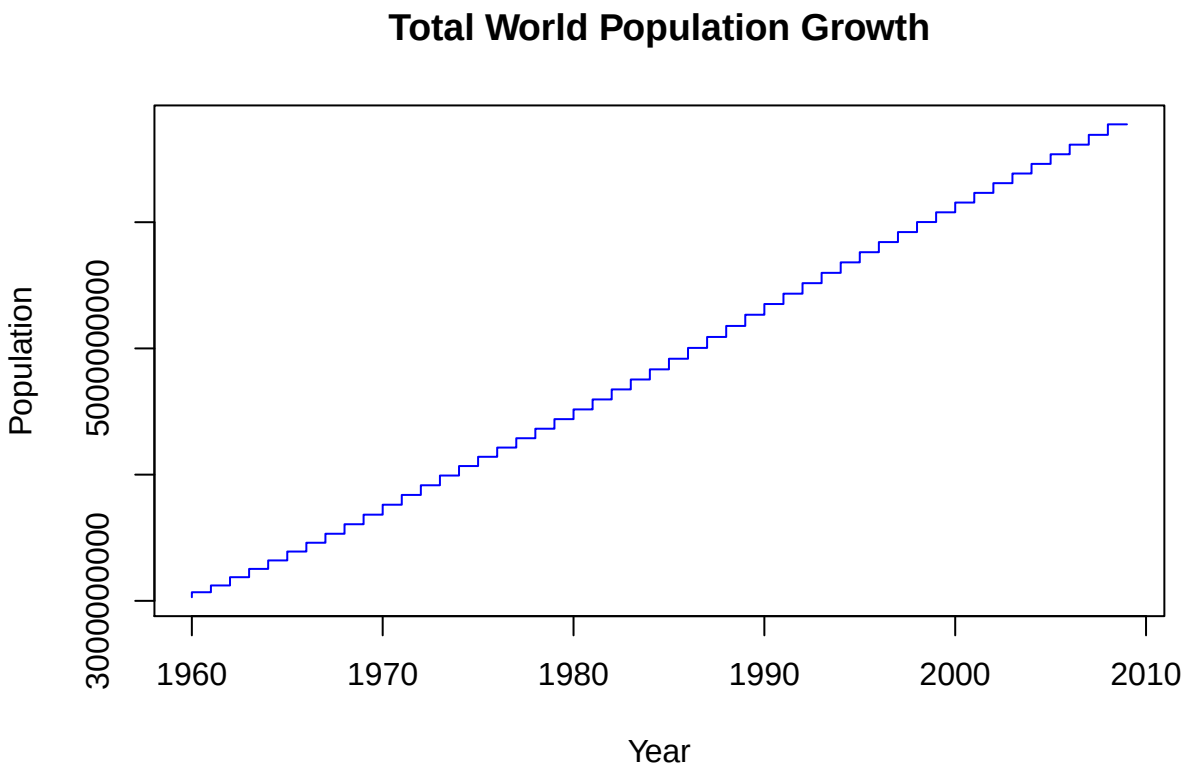
```
##       Year        Population
##  Min.   :1960   Min.   :3028654024
##  1st Qu.:1972   1st Qu.:3858296454
##  Median :1984   Median :4794042253
##  Mean   :1984   Mean   :4837313338
##  3rd Qu.:1997   3rd Qu.:5822497913
##  Max.   :2009   Max.   :6775235741
```

We can view from above the min/max year and population for the dataset

```
wp %>%
  ggplot(aes(x = Year, y = Population)) +
  geom_line(linetype = "dashed", size = 1, colour = "blue") +
  ggtitle("Total World Population Growth") +
  scale_y_continuous(labels = function(y) format(y, big.mark = ",",
                                                 scientific = FALSE))
```




```
plot(wp$Year, wp$Population, type = "S",
     col = "blue",
     main = "Total World Population Growth",
     xlab = "Year",
     ylab = "Population",
     )
```


Total World Population Growth
Population
3000000000
5000000000
1960
1970
1980
1990
2000
2010
Year