Assignment 6 – R line graphs

Professor John Sokol | Due 4/12

Rstudio line graphs:

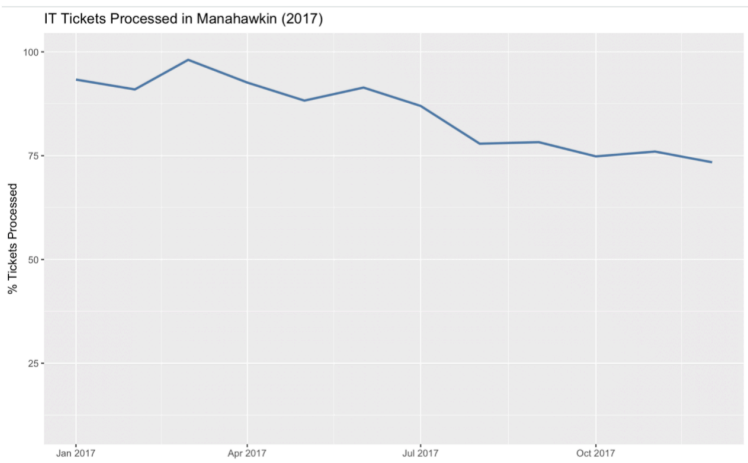
On March 27th, we introduced Rstudio as another tool in our data visualization repertoire. I introduced fundamental R coding concepts that are prerequisite understanding before any visualization can be built. Review the 3/27 lecture slides before proceeding.

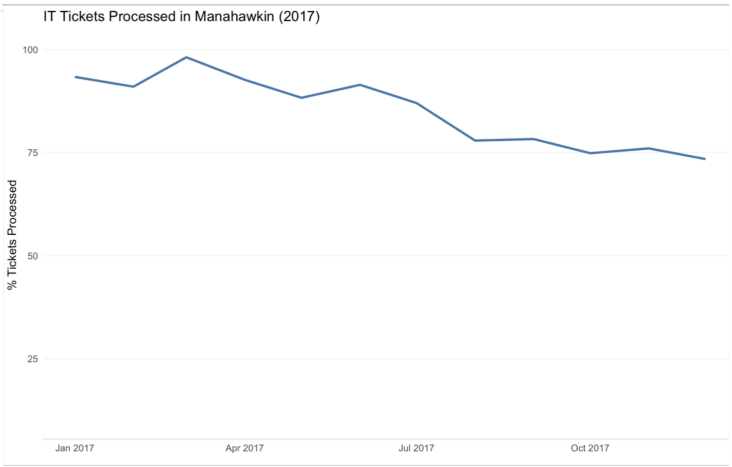
The Rstudio data visualization method we are leveraging in this class is ggplot. Although the default gray background ggplot theme is popular in the R community, the gray background does not agree with the minimalistic and data-emphasized styles taught in this course. As a result, I modified the default ggplot line graph with the following code:

```
library(ggplot2)

ggplot(data = dataset, aes(x = datafield_x_axis, y = datafield_y_axis)) + geom_line(color = "#4E79A7", size = 1) + theme(panel.background = element_blank(), axis.line.x = element_line(colour = "#DCDCDC"), panel.grid.major.y = element_line(size=.1, color="#DCDCDC"), axis.ticks = element_blank()) + ggtitle("Descriptive title") + xlab("") + ylab("y-axis label")
```

Notice the difference:





Fundamental line graph best practices that were reviewed during the Tableau line graph lecture are below for reference:

Line Graphs:
The line graph is an excellent tool to visualize change over time. Although this is feasible with a bar graph, the changing trends over time are not as easy to interpret.

Line graph formatting guide:

- Ensure data types of data fields are correctly assigned
- Remove excess white space by shrinking y-axis interval
- Employ color to pertinent categories
- Add axis labels as necessary
- Title that states a call to action
- Remove or light gray gridlines

A significant benefit of using data science grade tools such as Rstudio and Python for data visualization is the software is able to handle manipulation of large datasets that are hundreds of megabytes or even a few gigabytes in size. If you import a dataset that is gigabytes in size into Tableau, performance would take a substantial hit or Tableau can even shut itself down. This is a great transition into building a line graph of the Spotify dataset in Rstudio, which is about 216 megabytes in size. This dataset contains streaming information on popular artists such as stream count, track name, chart position, and stream by region from 1/1/2017 to 8/17/2017.

For this assignment, unzip and import the Spotify csv file into Rstudio using the 'From text (readr)' option, NOT the 'From Excel...' option as the Spotify dataset is a csv file. Change the data type of the Date column from (double) to (date). For the format string, put in %Y-%m-%d

Explore the dataset using both the head( ) and tail( ) functions.

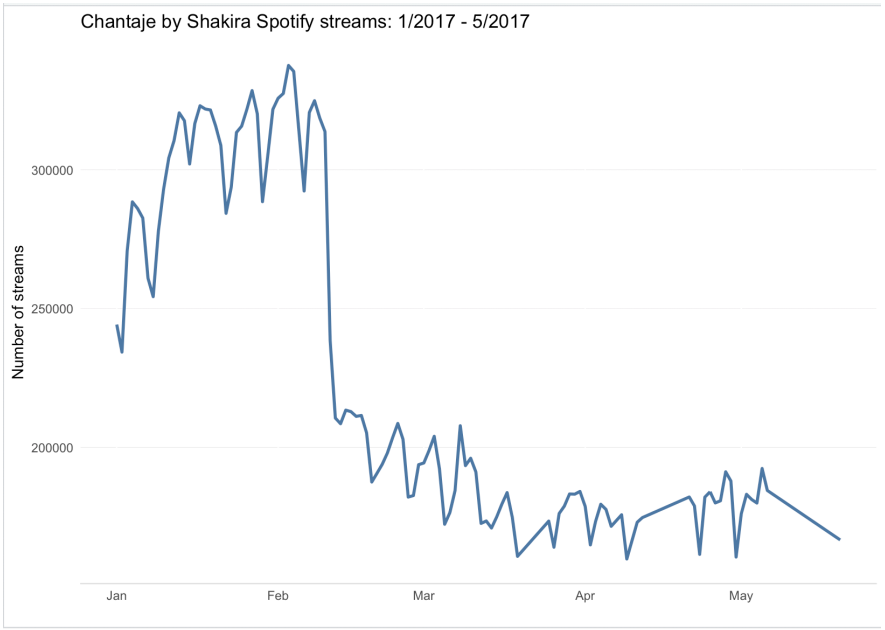
Now use the unique( ) function on region column of the dataset:

```
> unique(Spotify_data$Region)
 [1] "ar"     "at"     "au"     "be"     "bo"     "br"     "ca"     "ch"     "cl"     "co"     "cr"
[12] "cy"     "cz"     "de"     "dk"     "do"     "ec"     "ee"     "es"     "fi"     "fr"     "gb"
[23] "global" "gr"     "gt"     "hk"     "hn"     "hu"     "id"     "ie"     "is"     "it"     "jp"
[34] "lt"     "lv"     "mx"     "my"     "nl"     "no"     "nz"     "pa"     "pe"     "ph"     "pl"
[45] "pt"     "py"     "se"     "sg"     "sk"     "sv"     "tr"     "tw"     "us"     "uy"
```

This useful function returns all the unique values in the region data field; notice the “us” category.

Deliverables:

- Create an Rstudio line graph of the number of streams over time of the song Chantaje by Shakira in the United States. You will be filtering the Spotify_data dataset by Track_Name and region. The line graph should look similar to this:



- Submit a half page to one page write up of your thoughts, comments, and concerns of using Rstudio. How do you like using Rstudio compared to Tableau? Is it easier or more difficult? Do you like the idea of more freedom to change visualization detail in exchange for a more complex workflow?