Intro to Data Science - Lab 6

Week 6 - Using ggplot to Build Complex Data Displays

```
# Enter your name here: Hongdi Li

#install.packages('imputeTS')
#libary(imputeTS)
```

Please include nice comments.

Instructions:

Run the necessary code on your own instance of R-Studio.

Attribution statement: (choose only one and delete the rest)

```
# 1. I did this lab assignment by myself, with help from the book and the
professor.
```

Creating graphical displays of data is an essential skill for all data scientists because so much of what data scientists do involves communicating with other people about data. The **ggplot2** package developed by data scientist Hadley Wickham (https://en.wikipedia.org/wiki/Hadley_Wickham) provides excellent power and flexibility for graphically displaying data. Whole books have been written about **ggplot2** (e.g.: https://www.springer.com/gp/book/9780387981413), so we will only be able to scratch the surface, but we will master the basic grammar that you need in order to use this package. For this lab we will also use a dataset that comes delivered in R, called ** economics **. It contains 574 snapshots of U.S. economic conditions between 1967 and 2014.
Here are two lines of starter code:

```
MyPlot <- ggplot(economics, aes(x=date),inherit.aes = FALSE)
myPlot <- MyPlot + geom_line(aes(y=psavert))
```

1. Run these two lines of code below. What happens? How do you actually invoke the plot
   (i.e., how do you get it to draw in the plot window)?

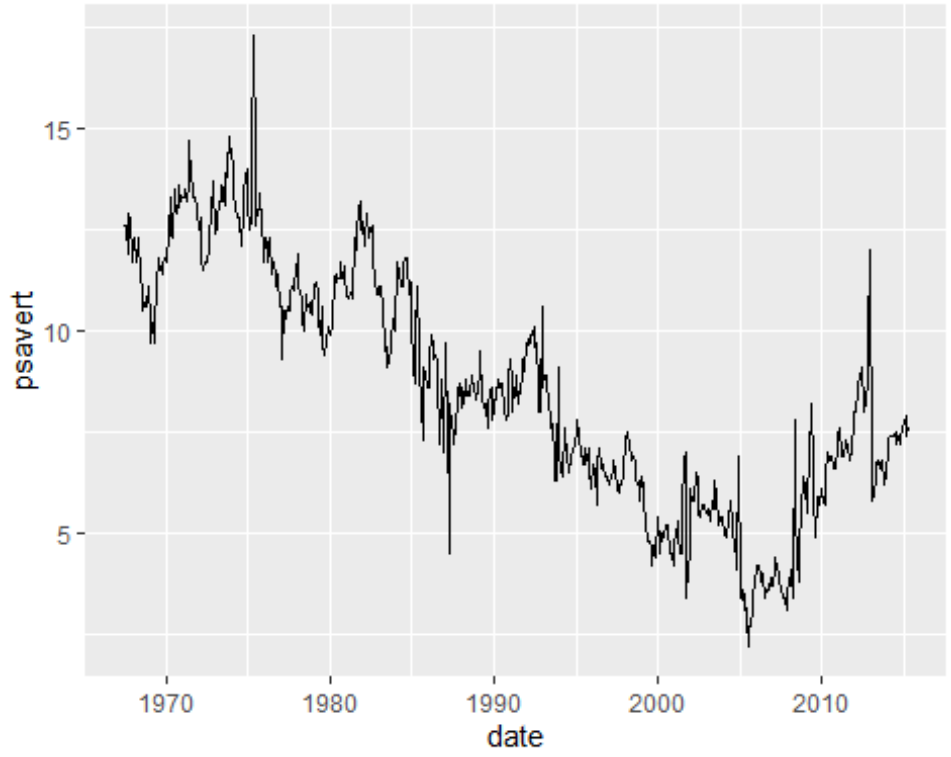
```
library(ggplot2)
MyPlot <- ggplot(economics, aes(x=date),inherit.aes = FALSE)
myPlot <- MyPlot + geom_line(aes(y=psavert))
#will create a graph and saved in myPlot.
#First, step get the x-axis with the date, and saved in MyPlot
#Then, add y-axis and draw the graph based on Myplo
#Get the new graph and saved in myPlot
```

2. Run `help("economics")` to find out the meaning of the **psavert** variable.

```
help("economics")

## starting httpd help server ... done

# which means the personal saving sate
```

3. Examine the plot to estimate when the personal savings rate reached its maximum value. Also examine the plot to estimate when the personal savings rate reached its minimum value.

```
myPlot
```



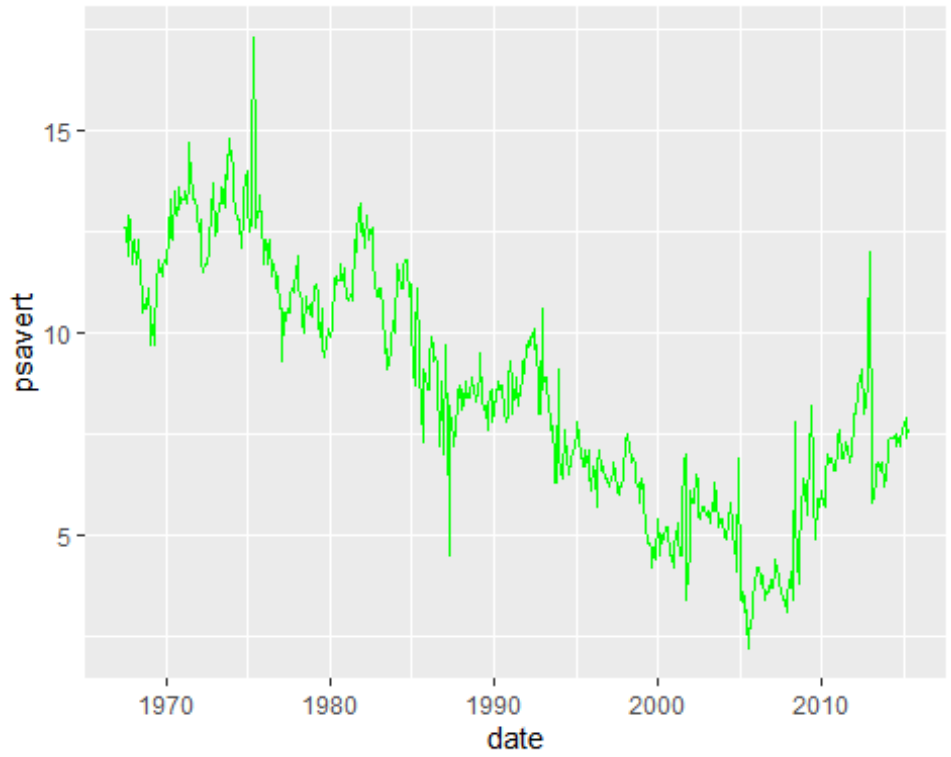

```
#around 1975 reach the maximum, 2005 reach the minimum
```

4. Use **which.max( )** and **which.min( )** to verify your guesses from problem 3.

```
economics$date[which.max(economics$psavert)]

## [1] "1975-05-01"

economics$date[which.min(economics$psavert)]

## [1] "2005-07-01"

#Almost same the 1975-05-01 reach max
# 2005-07-01 reach the minimum
```

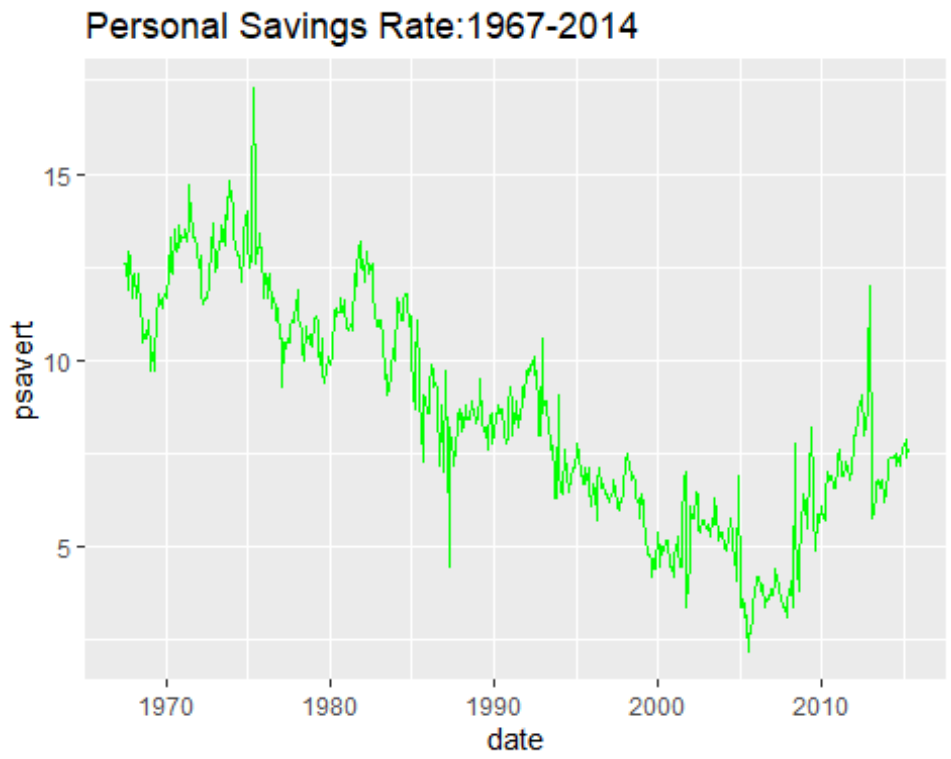
5. Change the color of the plot line to green. **Hint:** Changing a line to a constant color happens in the specification of the **geometry**.

```
myPlot <- MyPlot + geom_line(aes(y=psavert),color='green')
myPlot
```



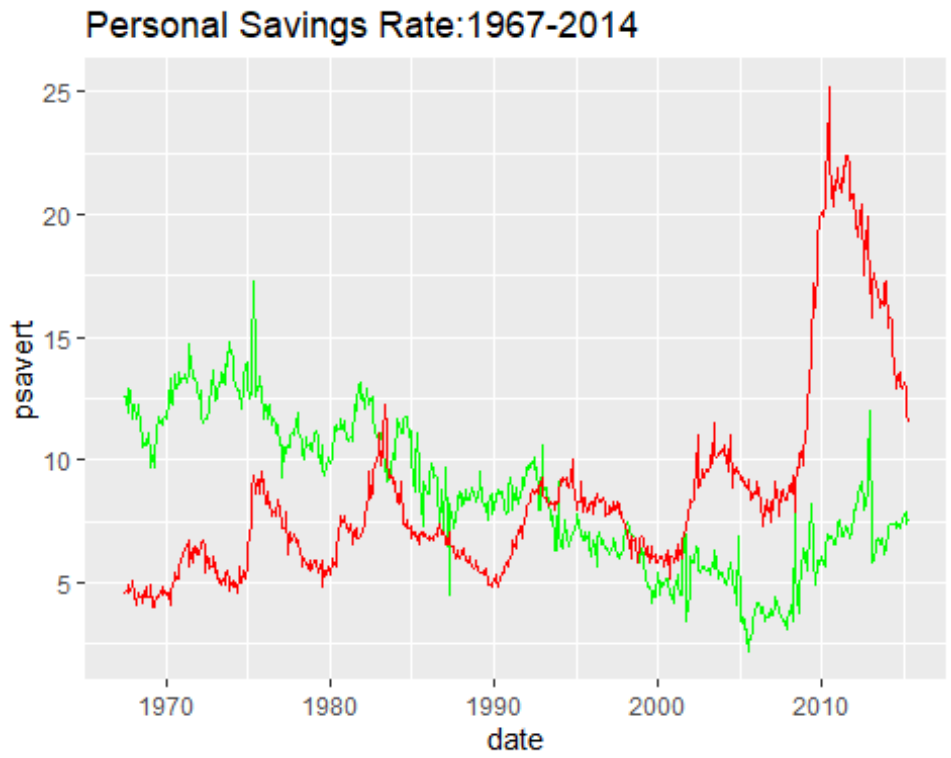
6. Add a title to the plot with the **ggtitle("Put title here")** sub-command. The title **"Personal Savings Rate: 1967-2014"** would be a good choice.

```
myPlot <- MyPlot + geom_line(aes(y=psavert),color='green')+ggtitle("Personal
Savings Rate:  1967-2014")
myPlot
```



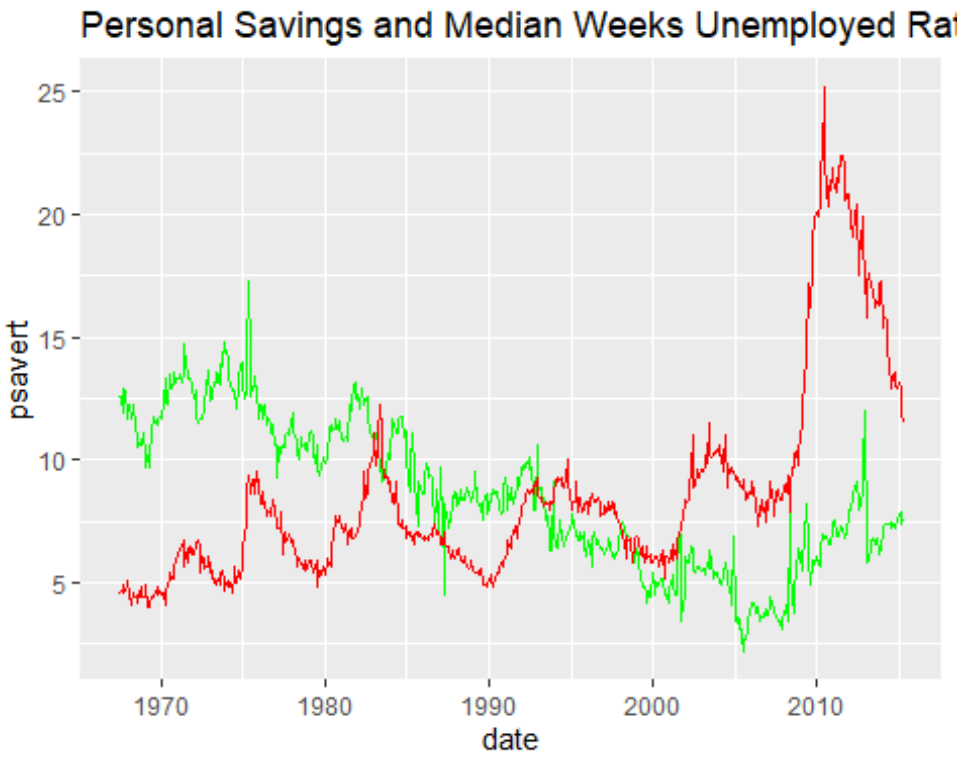
7. Add another data series to your plot to show the variable **uempmed** as a red line.

```
new_plot <- myPlot + geom_line(aes(y=uempmed),color="red")
new_plot
```



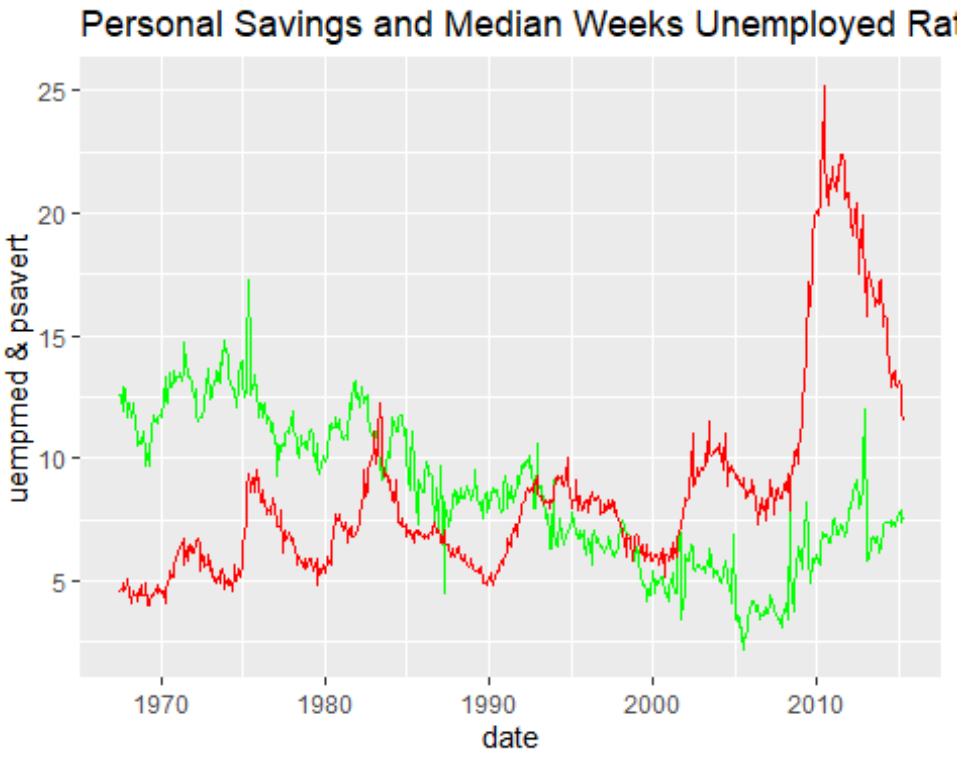
8. Change the title of the plot to mention both variables.

```
myPlot <- MyPlot + geom_line(aes(y=psavert),color='green')+ggtitle("Personal
Savings and Median Weeks Unemployed Rate:  1967-2014")
new_plot <- myPlot + geom_line(aes(y=uempmed),color="red")
new_plot
```



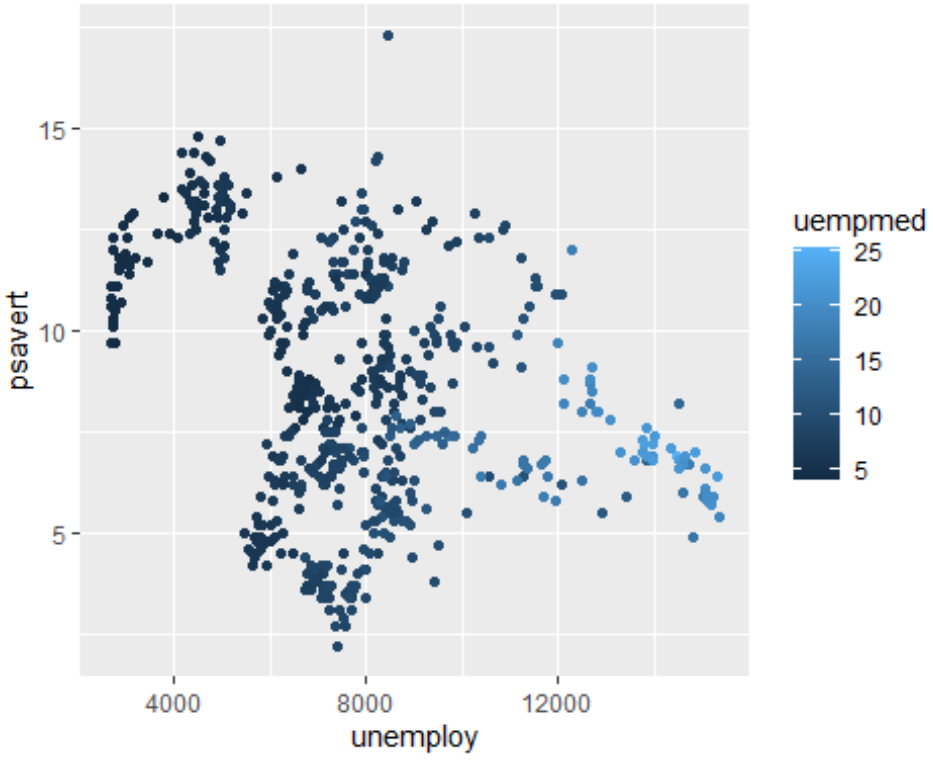
9. You can modify the axis labels in a ggplot with **ylab( )** and **xlab( )** subcommands. Change the axis labeling as needed to account for plotting both **psavert** and **uempmed** in the same window.

```
New<-new_plot+ylab("uempmed & psavert")
New
```



10. Create one last plot, creating a scatter plot, having the **unemploy** on the x-axis, **psavert** on the yaxis. Color each point based on the **uempmed**.

```
ggplot(data=economics,aes(x=unemploy,y=psavert,color=uempmed))+geom_point()
```



11. Interpret (using comments in R) what you see in this last graph.

```
#I can see a scatter plot with the relation between
#unemploy and psavert and they colored based on the uempmed variable
```