Intro to Data Science - Lab 6

Copyright 2022, Jeffrey Stanton and Jeffrey Saltz Please do not post online.

Week 6 - Using ggplot to Build Complex Data Displays

```
# Enter your name here: Caribel Florentino
```

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 (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 (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))
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)?

```
#install.packages('imputeTS')
#install.packages("ggmap")
library(ggmap)
```

```
## Loading required package: ggplot2
```

```
## ℹ Google's Terms of Service: <https://mapsplatform.google.com>
##   Stadia Maps' Terms of Service: <https://stadiamaps.com/terms-of-service/>
##   OpenStreetMap's Tile Usage Policy: <https://operations.osmfoundation.org/policies/t
iles/>
## ℹ Please cite ggmap if you use it! Use `citation("ggmap")` for details.
```
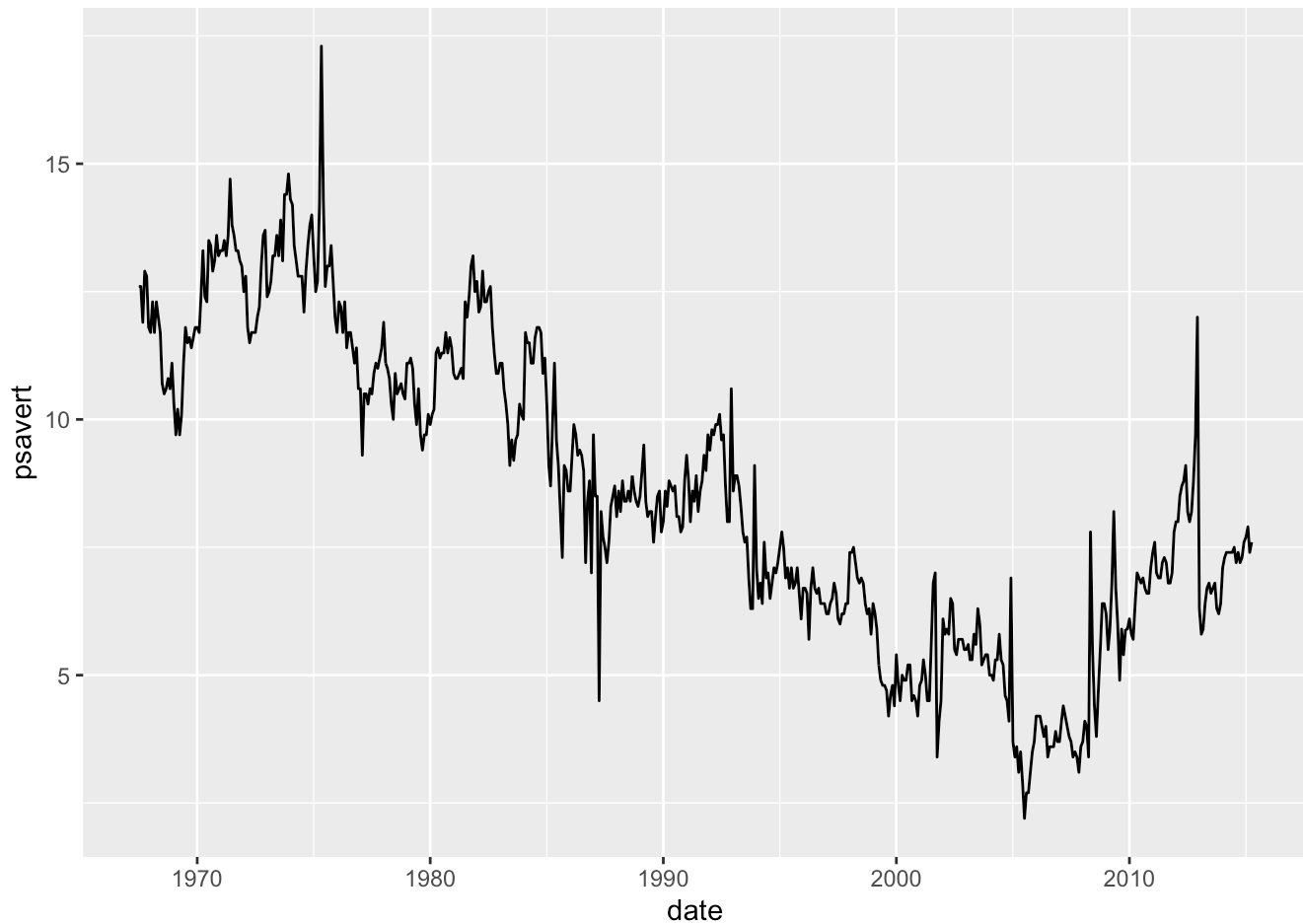
```
library(imputeTS)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

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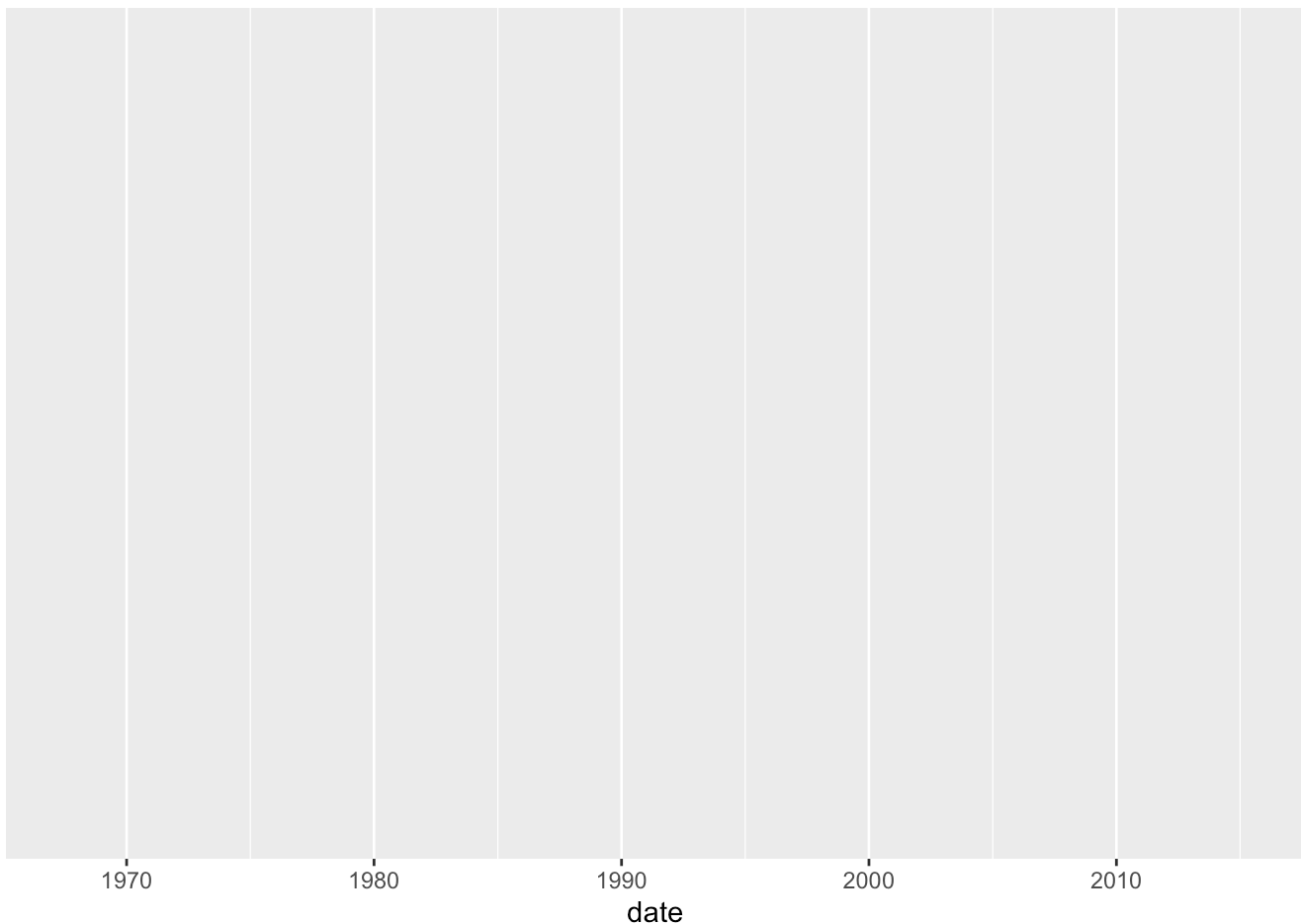


```
myPlot
```

```
# geom_line which is a  basic geometriy invoked the plot because after it was added, the
plot was added.
```

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

```
help("economics")
#psavert is the personal savings rate
```

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.

```
#max value 1976
#min value 2006
```

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

```
economics$date[which.max(economics$psavert)]
```
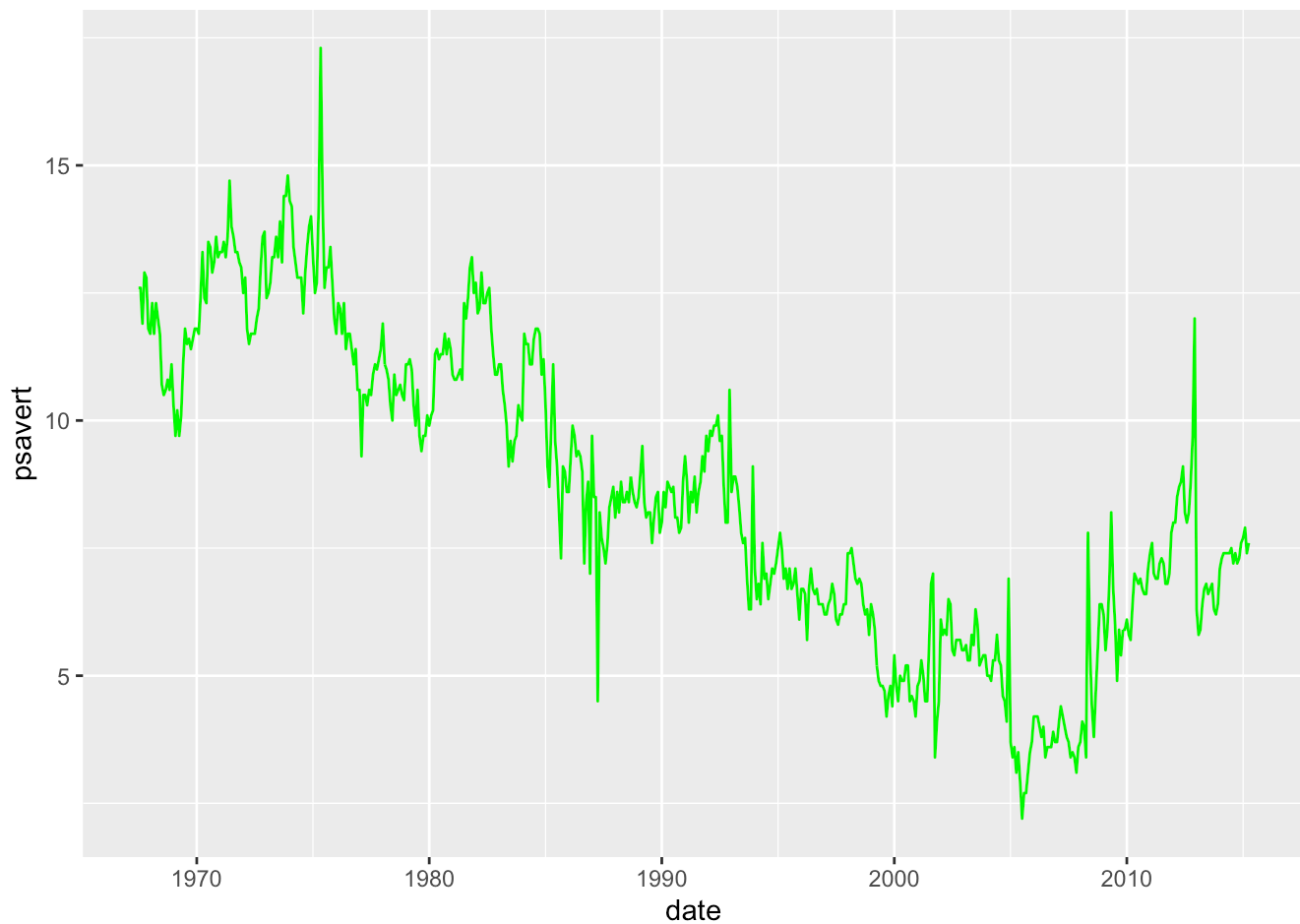
```
## [1] "1975-05-01"
```

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

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

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 + geom_line(aes(y=psavert), color ="green")
```


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 + geom_line(aes(y=psavert), color ="green") + ggtitle("Personal Savings Rate: 196
7-2014")
```
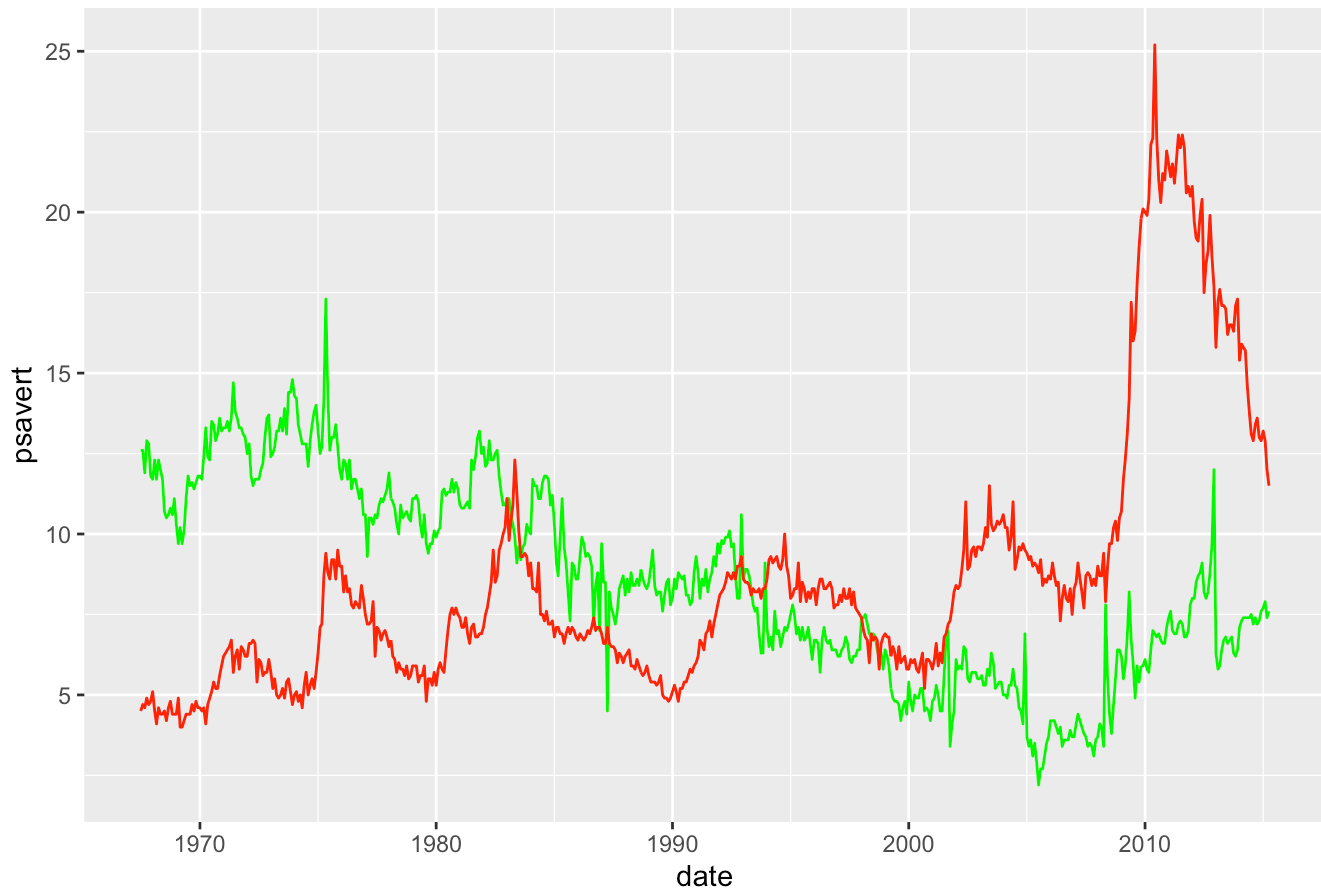
Personal Savings Rate: 1967-2014
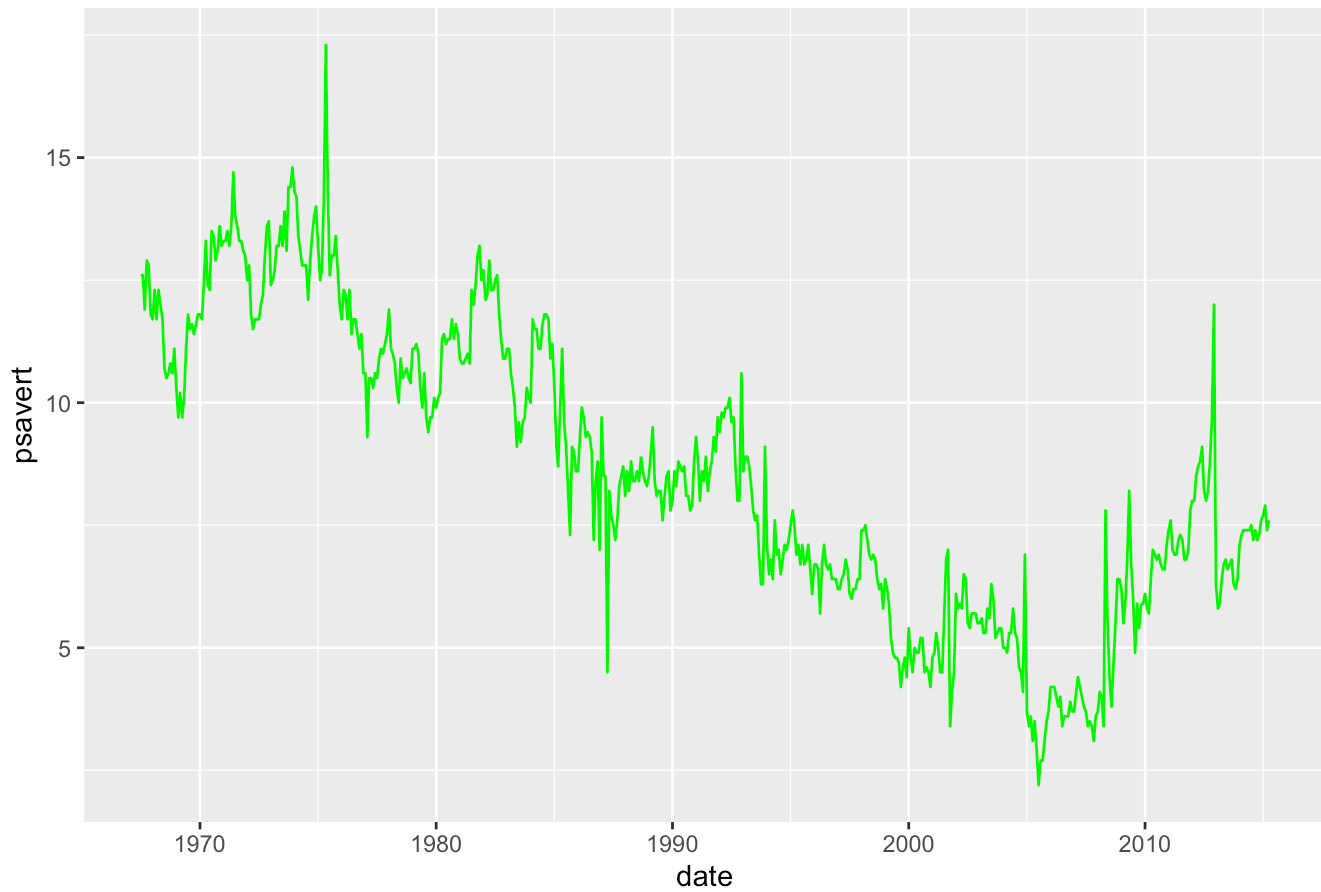
7. Add another data series to your plot to show the variable **uempmed** as a red line.

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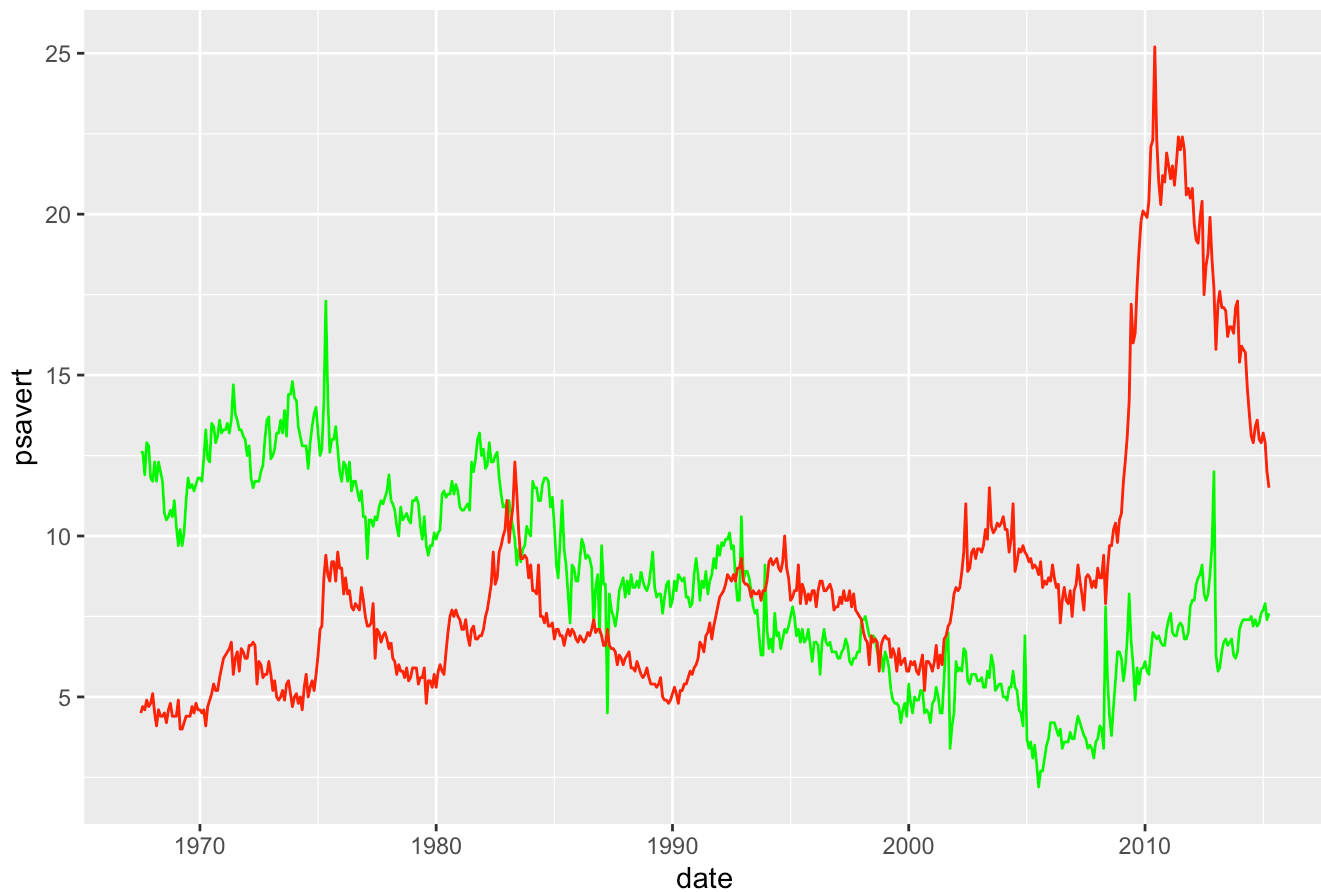
Personal Savings Rate: 1967-2014

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

```
MyPlot + geom_line(aes(y=psavert), color ="green") + geom_line(aes(y=uempmed), color =
"red") + ggtitle("Personal Savings Rate and Unemployment Rate: 1967-2014")
```

Personal Savings Rate and Unemployment Rate: 1967-2014

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.

```
MyPlot + geom_line(aes(y=psavert), color ="green") + geom_line(aes(y=uempmed), color =
"red") + ggtitle("Personal Savings Rate and Unemployment Rate: 1967-2014") + ylab("rat
e") + xlab("year")
```

Personal Savings Rate and Unemployment Rate: 1967-2014
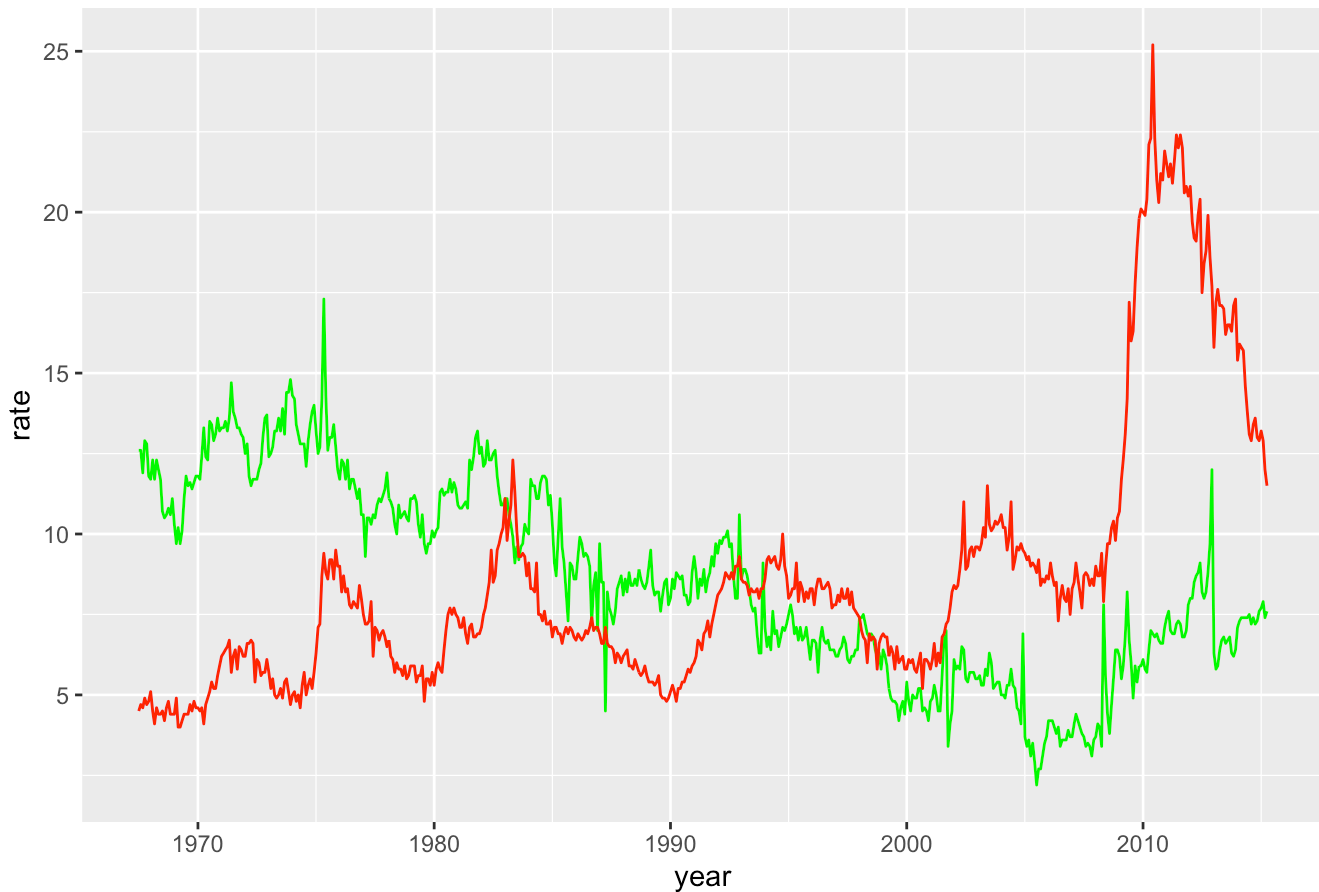
10. Create one last plot, creating a scatter plot, having the **unemploy** on the x-axis, **psavert** on the yaxis. Color each point based on the **uempmed**.

```
MyPlot + geom_point(aes(y=psavert, x=unemploy, color = uempmed)) + ylab("savings rate")
+ xlab("unemployment rate") + ggtitle("Personal Savings Rate: 1967–2014")
```

Personal Savings Rate: 1967-2014
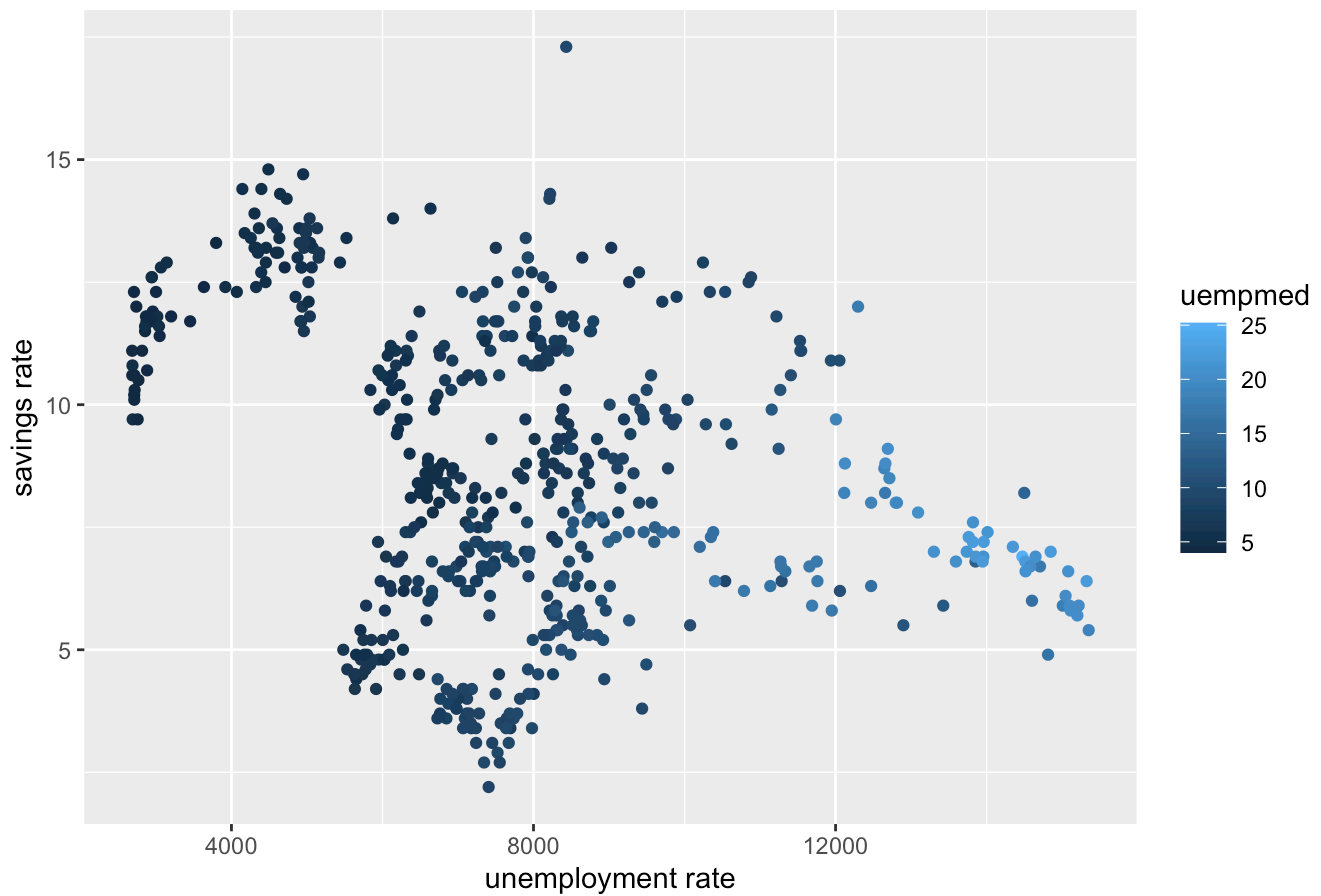
11. Interpret (using comments in R) what you see in this last graph.

```
#Higher unemployment rates and lower savings rates, increases the number of weeks that s
omeone is unemployed. In addition, lower unemployment rates and less time being unemploy
ed, increases their savings rate.
```