Data Visualization Analysis with R

(v. 2.0)

Oscar Torres-Reyna

otorres@princeton.edu

June 2019

http://www.princeton.edu/~otorres/

Data visualization helps reduce the mental stress of extracting meaning from data and plays an important role at all stages of data analysis when exploring data, making inferences, and presenting results

While different fields of study have developed their own way of visualizing data, the common goal across all types of visuals for data analysis is to find meaningful patterns through trends, relationships, or distribution.

See the following site

https://www.r-graph-gallery.com/

The data on the right shows a snapshot of the unemployment rate and presidential approval in the United States from 1948 to 2019.
In table form it is hard to figure something out of the data, except for the fact that it represents indicators measured over time, making it an ideal candidate for a time series graph.

date	unemp	approve	president	party
1/1/1948	3.4	50	Truman	Democrat
2/1/1948	3.8	*NA*	Truman	Democrat
3/1/1948	4.0	*NA*	Truman	Democrat
4/1/1948	3.9	36	Truman	Democrat
5/1/1948	3.5	39	Truman	Democrat
6/1/1948	3.6	39	Truman	Democrat
7/1/1948	3.6	*NA*	Truman	Democrat
8/1/1948	3.9	*NA*	Truman	Democrat
9/1/1948	3.8	*NA*	Truman	Democrat
10/1/1948	3.7	*NA*	Truman	Democrat
11/1/1948	3.8	*NA*	Truman	Democrat
12/1/1948	4.0	*NA*	Truman	Democrat
1/1/1949	4.3	69	Truman	Democrat
2/1/1949	4.7	*NA*	Truman	Democrat

We will be using RStudio, please see the following document for some introduction to its interface

https://dss.princeton.edu/training/RStudio101.pdf

We shall start by activating some of the R packages we will need for this document:

```
library(zoo)
library(ggplot2)
library(stargazer)
```

If the package is not available, you need to install it, type

```
install.packages("name of package")
```

The data is in *.csv format, we can use the `read.csv()` function import it into R:

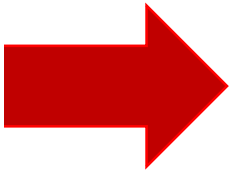
```
mydata = read.csv("http://www.princeton.edu/~otorres/mydataviz.csv",
                  header = TRUE, stringsAsFactors = FALSE)
```

1) unemp = Unemployment from the Bureau of Labor Statistics:

2) approve = Own estimation of monthly averages using presidential approval data from ROPER center. Note that data is publicly available at *The American Presidency Project* at the following site:
https://www.presidency.ucsb.edu/statistics/data/presidential-job-approval

```
mydata$date = as.Date(mydata$date,"%m/%d/%Y")

mydata = mydata[ order(mydata$date),]

mydata$month = as.yearmon(mydata$date)
```

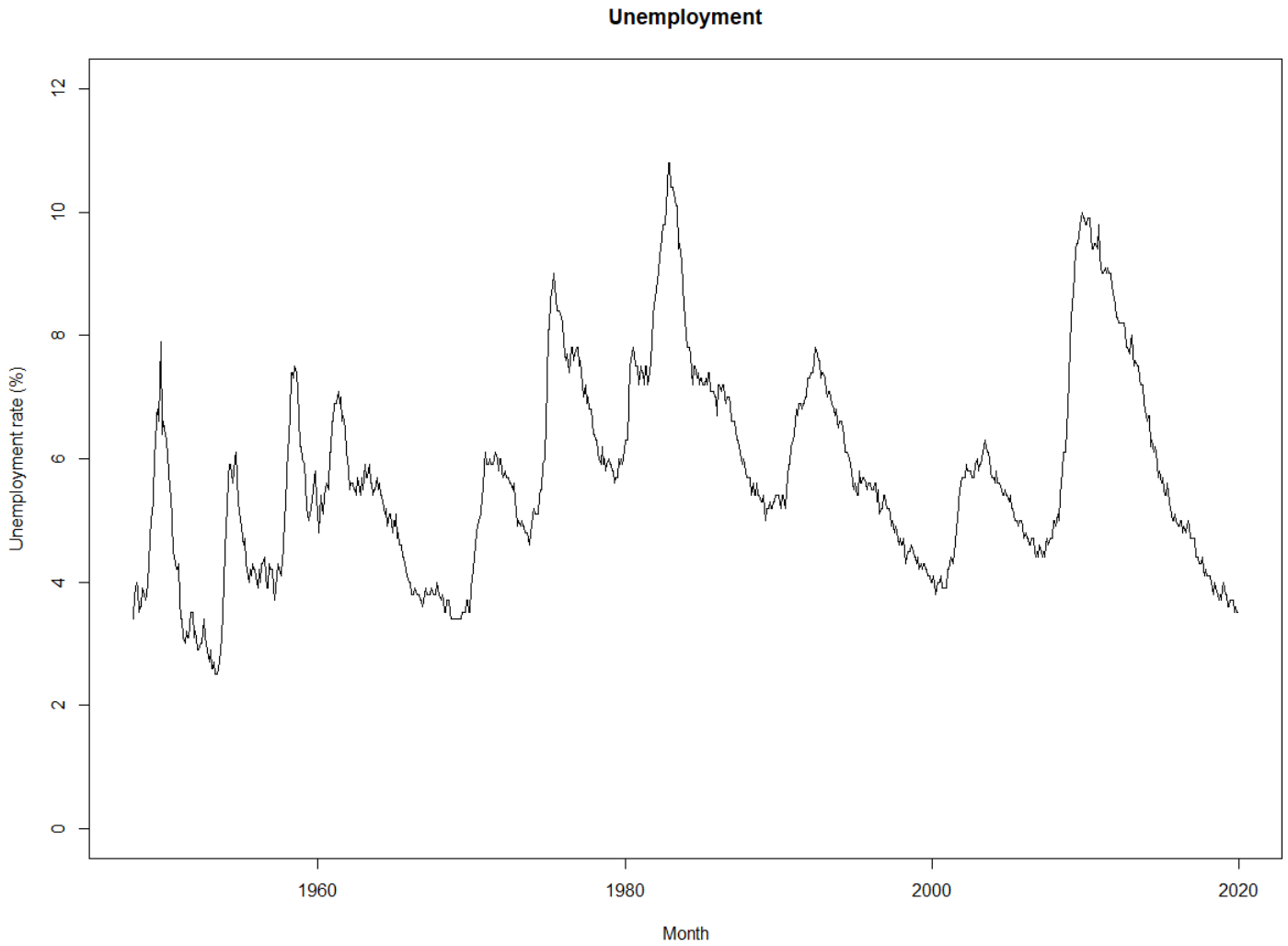
date	unemp	approve	president	party
1/1/1948	3.4	50	Truman	Democrat
2/1/1948	3.8	NA	Truman	Democrat
3/1/1948	4.0	NA	Truman	Democrat
4/1/1948	3.9	36	Truman	Democrat
5/1/1948	3.5	39	Truman	Democrat
6/1/1948	3.6	39	Truman	Democrat
7/1/1948	3.6	NA	Truman	Democrat
8/1/1948	3.9	NA	Truman	Democrat
9/1/1948	3.8	NA	Truman	Democrat
10/1/1948	3.7	NA	Truman	Democrat
11/1/1948	3.8	NA	Truman	Democrat
12/1/1948	4.0	NA	Truman	Democrat
1/1/1949	4.3	69	Truman	Democrat
2/1/1949	4.7	NA	Truman	Democrat



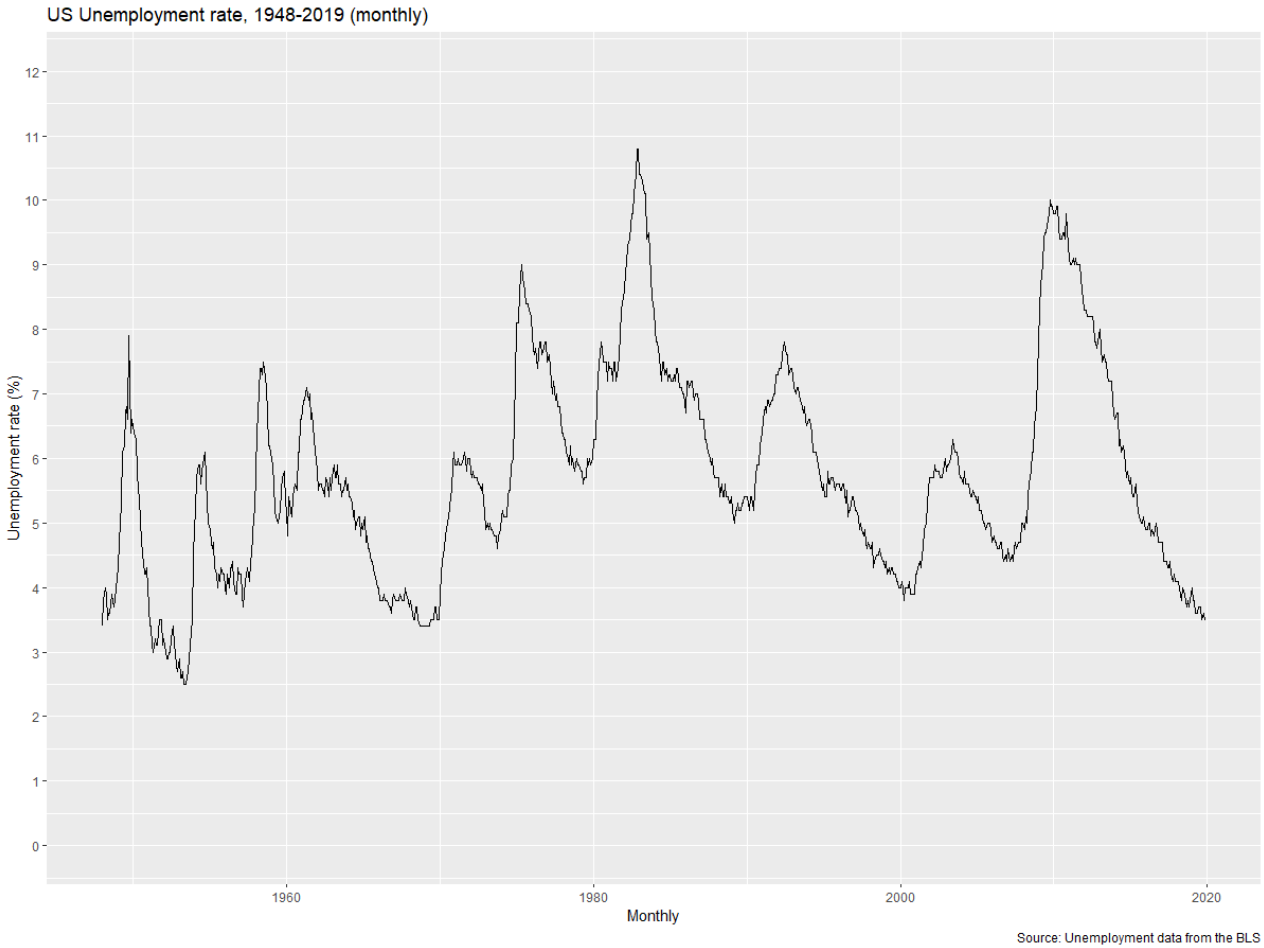
date	unemp	approve	president	party	month
1948-01-01	3.4	50	Truman	Democrat	Jan 1948
1948-02-01	3.8	NA	Truman	Democrat	Feb 1948
1948-03-01	4.0	NA	Truman	Democrat	Mar 1948
1948-04-01	3.9	36	Truman	Democrat	Apr 1948
1948-05-01	3.5	39	Truman	Democrat	May 1948
1948-06-01	3.6	39	Truman	Democrat	Jun 1948
1948-07-01	3.6	NA	Truman	Democrat	Jul 1948
1948-08-01	3.9	NA	Truman	Democrat	Aug 1948
1948-09-01	3.8	NA	Truman	Democrat	Sep 1948
1948-10-01	3.7	NA	Truman	Democrat	Oct 1948
1948-11-01	3.8	NA	Truman	Democrat	Nov 1948
1948-12-01	4.0	NA	Truman	Democrat	Dec 1948
1949-01-01	4.3	69	Truman	Democrat	Jan 1949
1949-02-01	4.7	NA	Truman	Democrat	Feb 1949

See: https://dss.princeton.edu/training/TimeSeriesR101.pdf

```
# Plotting unemployment data using base R function plot()
plot(mydata$month, mydata$unemp,
     type = "l", main="Unemployment", ylim=c(0,12),
     xlab="Month", ylab="Unemployment rate (%)")
```



```
# Plotting unemployment data using ggplot2
ggplot(data=mydata, aes(x=month, y=unemp)) +
  geom_line() +
  labs(title = "US Unemployment rate, 1948-2019 (monthly)",
       y = "Unemployment rate (%)",
       x = "Monthly",
       caption = "Source: Unemployment data from the BLS") +
  scale_y_continuous(limits = c(0, 12), breaks = seq(0,12))
```



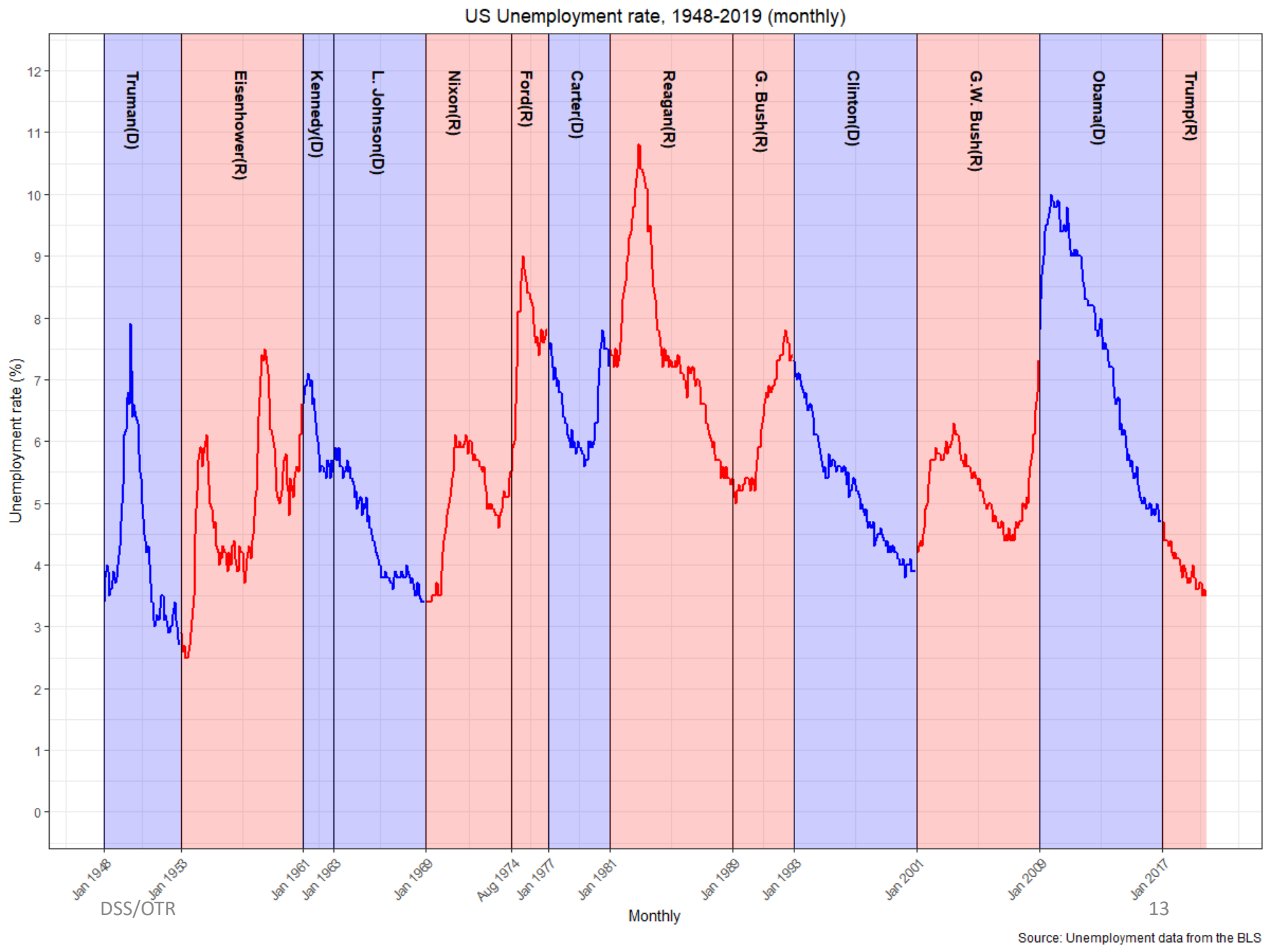
We can add analytic component to the visual by incorporating context to the trends. In the next slides we will see the unemployment trends by presidential terms.

```
# Mid-point date per administration

terms = data.frame(month = as.yearmon(c("Jan 1950","Jan 1957","Jan 1962","Jan 1966","Jan 1971",
                                        "Sep 1975","Jan 1979","Jan 1985","Jan 1991","Jan 1997",
                                        "Jan 2005","Jan 2013","Jan 2019")),
                   y = 12,
                   name = c("Truman(D)","Eisenhower(R)","Kennedy(D)","L. Johnson(D)","Nixon(R)",
                            "Ford(R)","Carter(D)","Reagan(R)","G. Bush(R)","Clinton(D)",
                            "G.W. Bush(R)","Obama(D)","Trump(R)"))


# Begin date

start = c("Jan 1948","Jan 1953","Jan 1961","Jan 1963",
              "Jan 1969","Aug 1974","Jan 1977","Jan 1981",
              "Jan 1989","Jan 1993","Jan 2001","Jan 2009","Jan 2017")


# End date

end = c("Jan 1953","Jan 1961","Jan 1963","Jan 1969",
               "Aug 1974","Jan 1977","Jan 1981","Jan 1989",
               "Jan 1993","Jan 2001","Jan 2009","Jan 2017","Dec 2019")

# Order

order = c("blue","red","blue","blue","red","red","blue","red","red","blue","red","blue","red")
```

```
mydata$unempd = ifelse(mydata$party=="Democrat",mydata$unemp, NA)
mydata$unempr = ifelse(mydata$party=="Republican",mydata$unemp, NA)
```

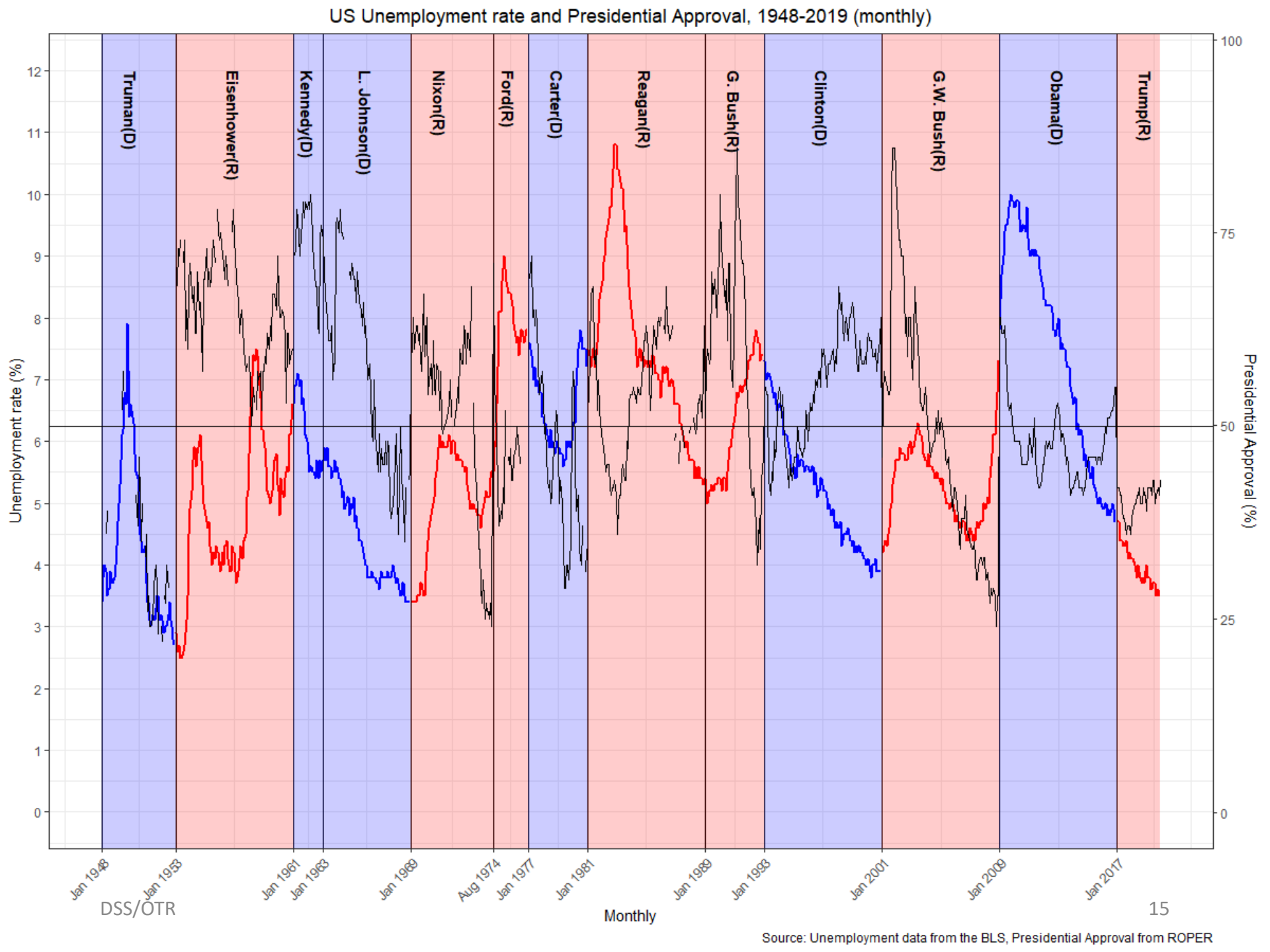
```
ggplot(data=mydata, aes(x=month)) +
  geom_line(data = mydata, aes(y = unempd), color = "blue", size = 1) +
  geom_line(data = mydata, aes(y = unempr), color = "red", size = 1) +
  theme_bw() +
  labs(title = "US Unemployment rate, 1948-2019 (monthly)",
       y = "Unemployment rate (%)",
       x = "Monthly",
       caption = "Source: Unemployment data from the BLS") +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits = c(0, 12), breaks = seq(0,12)) +
  scale_x_yearmon(breaks = as.yearmon(start)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  geom_vline(xintercept = as.yearmon(start),
             linetype = "solid",
             size = 0.5,
             color = "black") +
  annotate("rect",
           xmin = as.yearmon(start),
           xmax = as.yearmon(end),
           ymin = -Inf, ymax = Inf,
           fill = order,
           alpha = 0.2) +
  geom_text(data = terms, aes(x=month, y = y, label=name), angle = 270, hjust = 0, fontface= "bold")
```

See next page


US Unemployment rate, 1948-2019 (monthly)
Truman(D)
Eisenhower(R)
Kennedy(D)
L. Johnson(D)
Nixon(R)
Ford(R)
Carter(D)
Reagan(R)
G. Bush(R)
Clinton(D)
G.W. Bush(R)
Obama(D)
Trump(R)
Unemployment rate (%)
0
1
2
3
4
5
6
7
8
9
10
11
12
Jan 1948
Jan 1953
Jan 1961
Jan 1963
Jan 1969
Aug 1974
Jan 1977
Jan 1981
Jan 1989
Jan 1993
Jan 2001
Jan 2009
Jan 2017
Monthly
Source: Unemployment data from the BLS

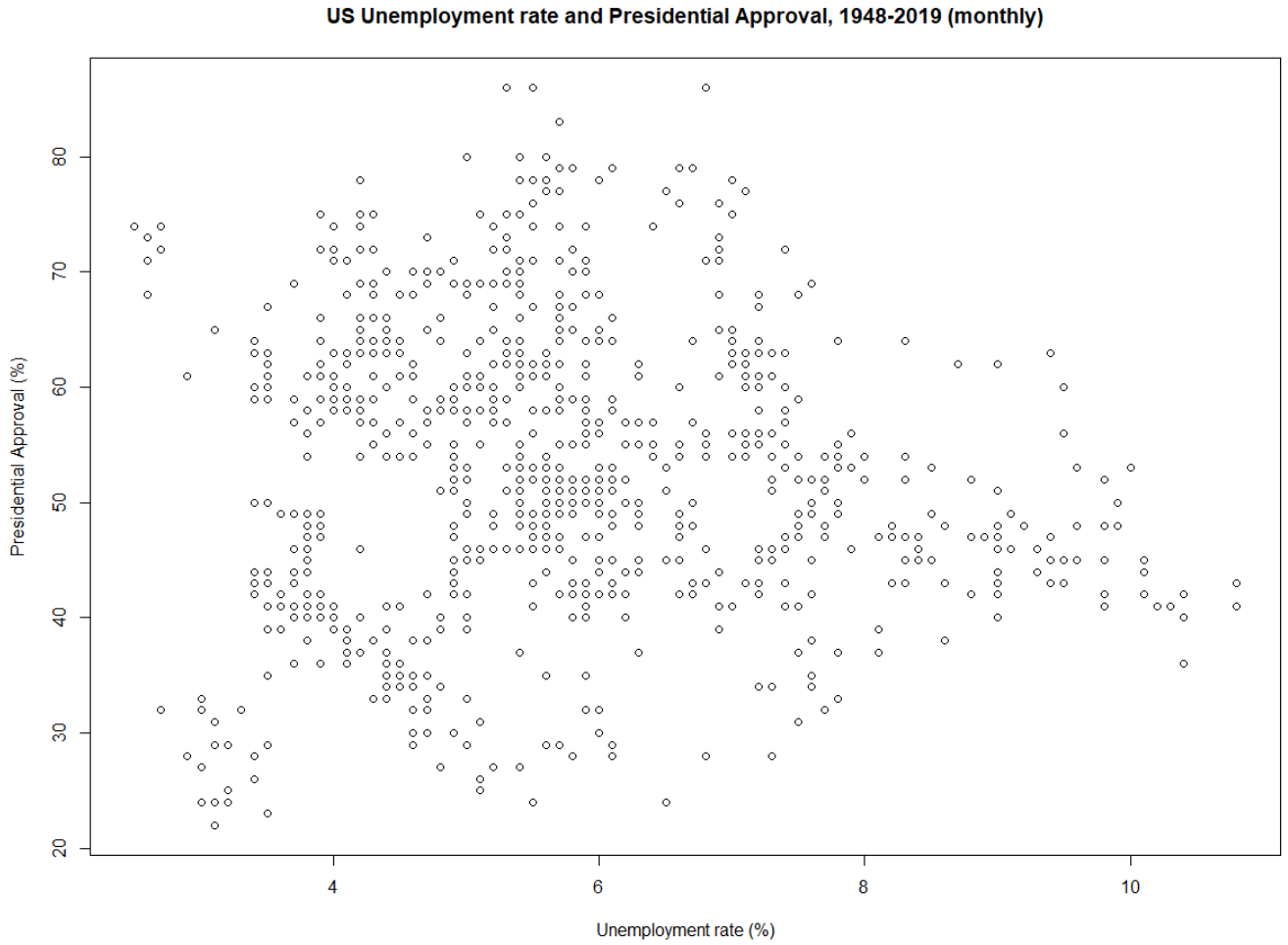
Adding presidential approval
See next page

```
ggplot(data=mydata, aes(x=month)) +
  geom_line(data = mydata, aes(y = unempd), color = "blue", size = 1) +
  geom_line(data = mydata, aes(y = unempr), color = "red", size = 1) +
  theme_bw() +
  labs(title = "US Unemployment rate and Presidential Approval, 1948-2019 (monthly)",
       y = "Unemployment rate (%)",
       x = "Monthly",
       caption = "Source: Unemployment data from the BLS, Presidential Approval from ROPER") +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_y_continuous(limits = c(0, 12), breaks = seq(0,12)) +
  scale_x_yearmon(breaks = as.yearmon(start)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  geom_vline(xintercept = as.yearmon(start),
             linetype = "solid",
             size = 0.5,
             color = "black") +
  annotate("rect",
           xmin = as.yearmon(start),
           xmax = as.yearmon(end),
           ymin = -Inf, ymax = Inf,
           fill = order,
           alpha = 0.2) +
  geom_hline(yintercept = 50/7.997222) +
  geom_text(data = terms, aes(x=month, y = y, label=name), angle = 270, hjust = 0, fontface= "bold") +
  geom_line(data = mydata, aes(y = approve/7.997222)) +
  scale_y_continuous(sec.axis = sec_axis(~.*7.997222,(name = "Presidential Approval (%)")),
                     limits = c(0, 12), breaks = seq(0,12))
```


US Unemployment rate and Presidential Approval, 1948-2019 (monthly)
Truman(D)
Eisenhower(R)
Kennedy(D)
L. Johnson(D)
Nixon(R)
Ford(R)
Carter(D)
Reagan(R)
G. Bush(R)
Clinton(D)
G.W. Bush(R)
Obama(D)
Trump(R)
Unemployment rate (%)
Presidential Approval (%)
Jan 1948
Jan 1953
Jan 1961
Jan 1963
Jan 1969
Aug 1974
Jan 1977
Jan 1981
Jan 1989
Jan 1993
Jan 2001
Jan 2009
Jan 2017
Monthly
Source: Unemployment data from the BLS, Presidential Approval from ROPER

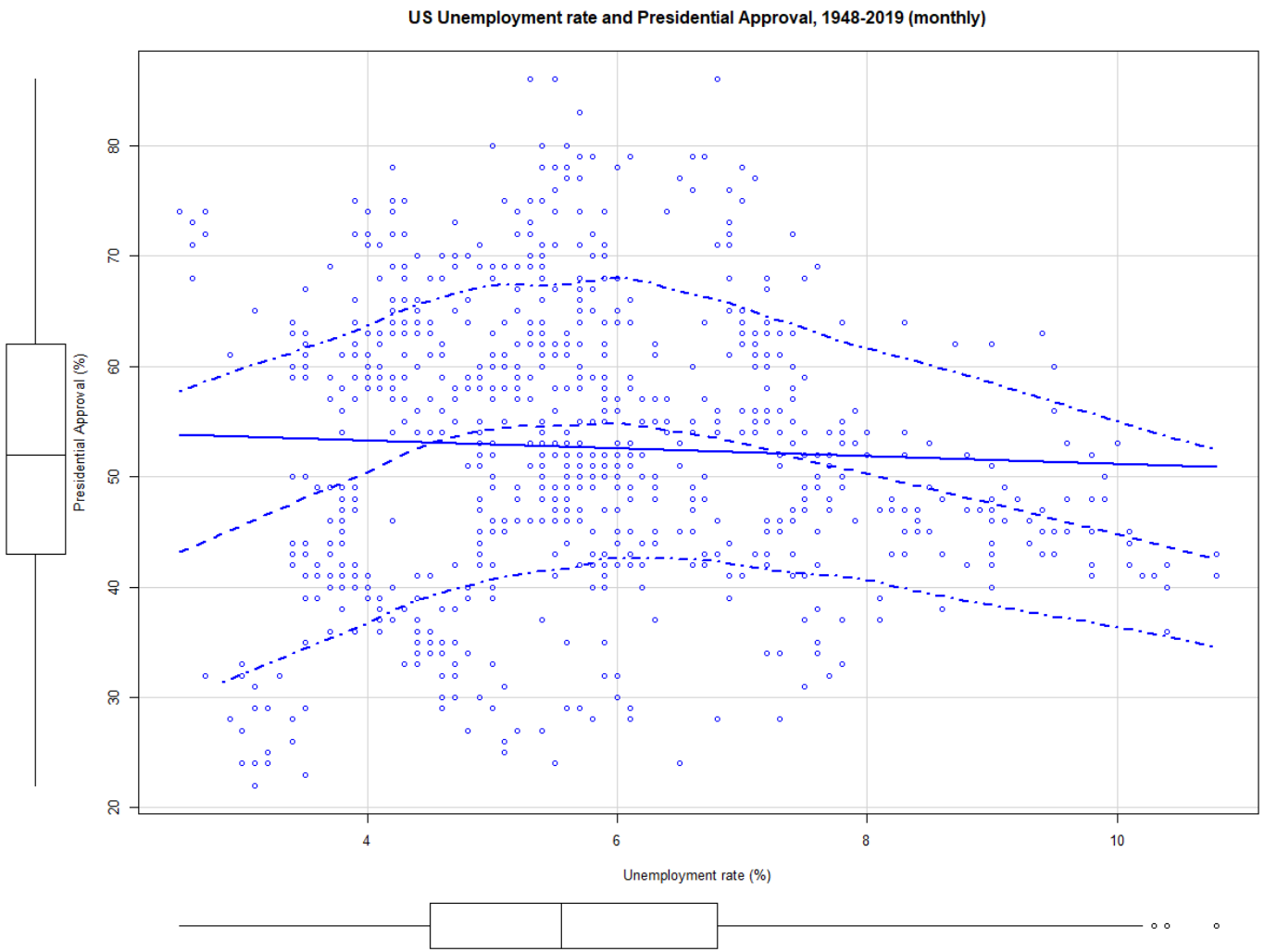
The previous visual does not provide a clear idea of the relationship between unemployment rates and presidential approval. Scatterplots are ideal plots to find relationships between variables. The following code produces a scatterplot using base R:

```
plot(mydata$unemp, mydata$approve,
     main = "US Unemployment rate and Presidential Approval, 1948-2019 (monthly)",
     xlab = "Unemployment rate (%)", ylab = "Presidential Approval (%)")
```



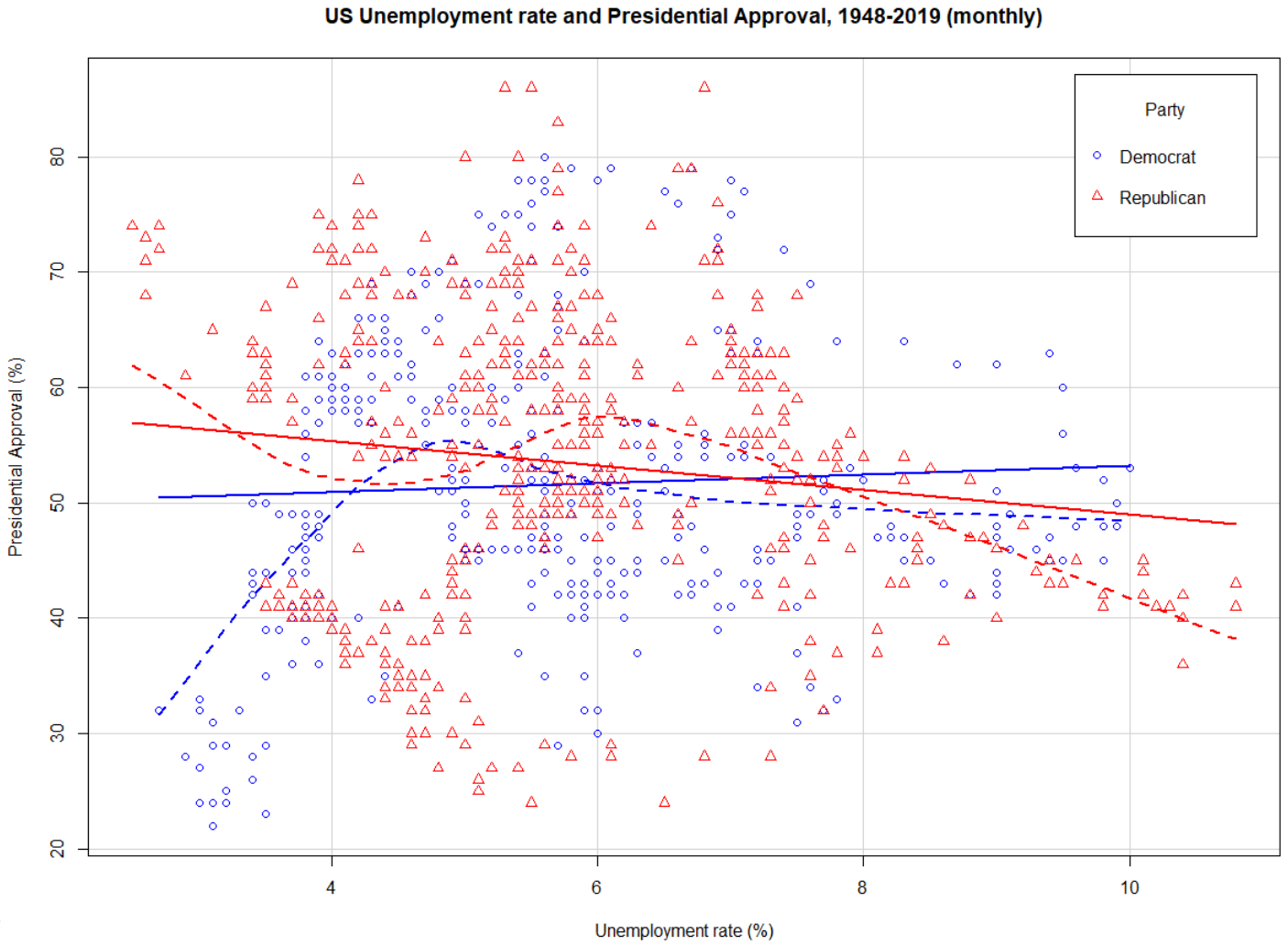
The `car` package provides an informative scatterplot including a linear and loess fit with boxplots

```
scatterplot(approve ~ unemp, data = mydata,
            main = "US Unemployment rate and Presidential Approval, 1948-2019 (monthly)",
            xlab = "Unemployment rate (%)", ylab = "Presidential Approval (%)")
```



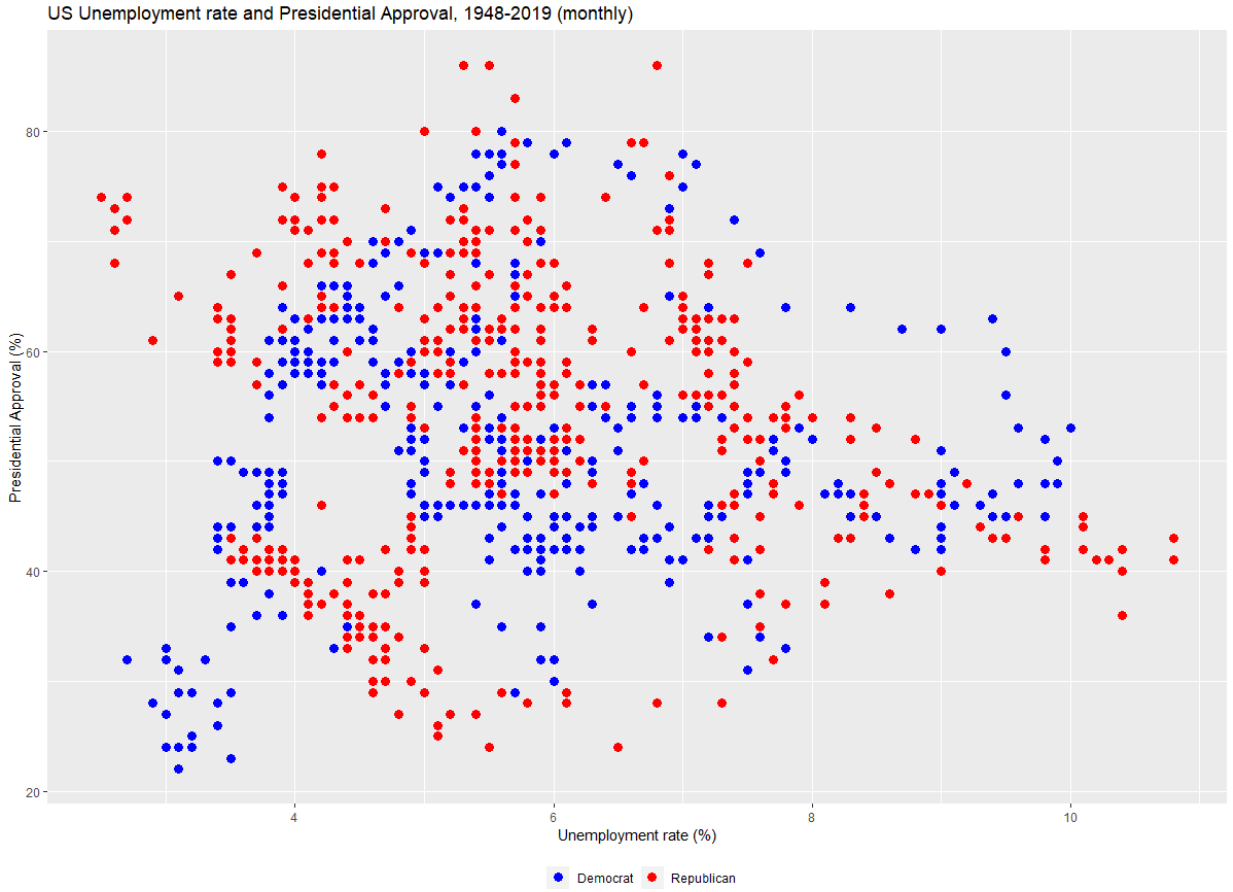
Per group

```
scatterplot(approve ~ unemp|party, data = mydata,
            main = "US Unemployment rate and Presidential Approval, 1948-2019 (monthly)",
            xlab = "Unemployment rate (%)", ylab = "Presidential Approval (%)",
            col = c("blue","red"),
            legend = c(title="Party", coords="topright"))
```



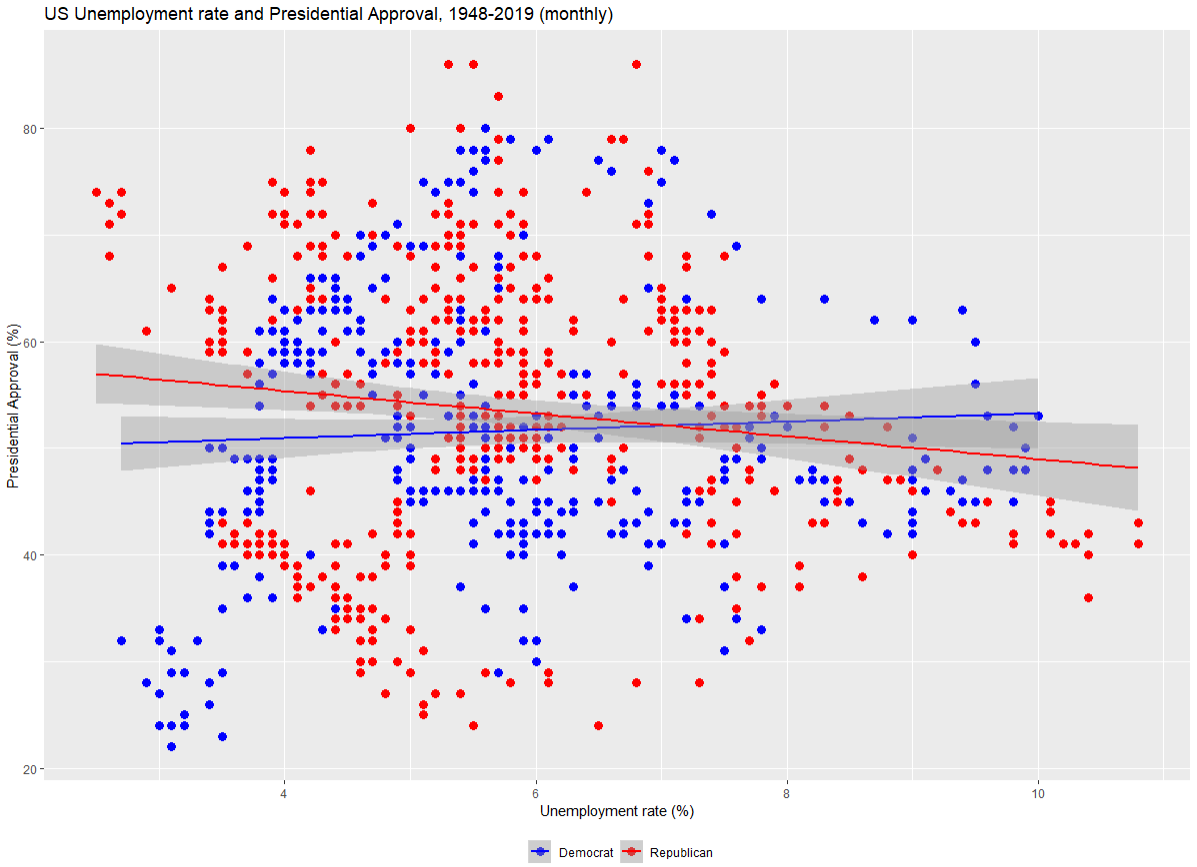
Scatterplots using `ggplot()`

```
ggplot(data = mydata, aes(x=unemp, y=approve, group=factor(party), color = factor(party))) +
  geom_point(size = 3) +
  scale_color_manual(name="",values=c('blue', 'red')) +
  theme(legend.position = "bottom") +
  labs(title = "US Unemployment rate and Presidential Approval, 1948-2019 (monthly)",
       x = "Unemployment rate (%)",
       y = "Presidential Approval (%)",
       caption = "Source: Unemployment data from the BLS, Presidential Approval from ROPER")
```



Adding a linear fit per party

```
ggplot(data = mydata, aes(x=unemp, y=approve, group=factor(party), color = factor(party))) +
  geom_point(size = 3) +
  scale_color_manual(name="",values=c('blue', 'red')) +
  theme(legend.position = "bottom") +
  labs(title = "US Unemployment rate and Presidential Approval, 1948-2019 (monthly)",
       x = "Unemployment rate (%)",
       y = "Presidential Approval (%)",
       caption = "Source: Unemployment data from the BLS, Presidential Approval from ROPER") +
  stat_smooth(method=lm)
```



We can estimate the equation of the linear fit per party by using the `lm()` function and produce a nice presentation using `stargazer()`.

```
reg1 = lm(approve ~ unemp, data = subset(mydata, party=="Democrat"))


reg2 = lm(approve ~ unemp, data = subset(mydata, party=="Republican"))


stargazer(reg1, reg2, type = "html", out="reg.html",
          dep.var.labels=c("Presidential Approval"),
          model.numbers = FALSE,
          column.labels =c("Democrats", "Republicans"),
          covariate.labels=c("Unemployment rate"))
```

The file reg.html can be opened with Word, it is saved in the working directory.

See: https://dss.princeton.edu/training/Regression101R.pdf and https://dss.princeton.edu/training/NiceOutputR.pdf

See next page

	Dependent variable:	
	Presidential Approval	
	Democrats	Republicans
Unemployment rate	0.387	-1.066***
	(0.373)	(0.389)
Constant	49.389***	59.654***
	(2.233)	(2.327)
Observations	365	445
R^2	0.003	0.017
Adjusted R^2	0.0002	0.014
Residual Std. Error	12.169 (df = 363)	12.979 (df = 443)
F Statistic	1.080 (df = 1; 363)	7.486*** (df = 1; 443)
Note:		*p**p***p<0.01

References

- Data and Statistical Services tutorials: https://dss.princeton.edu/training/
- UCLA: https://stats.idre.ucla.edu/
- Stackoverflow: https://stackoverflow.com/questions/tagged/r
- StackExchange: https://stats.stackexchange.com/questions/tagged/r
- Google: https://www.google.com/
- Data visualization with –`ggplot2`: https://ggplot2.tidyverse.org/reference/