

M4 Competition

VuThanh

March 21, 2019

Import packages

```
library(M4comp2018)
library(xts)
library(astsa)
library(ggplot2)
library(forecast)
library(ggfortify)
library(fpp2)
```

Load M4 data

```
data(M4)
```

Create a data frame to summarize the M4 data structure

```
df = data.frame(matrix(ncol = 5, nrow = 100000))
colnames(df) = c("st", "n", "type", "h", "period")
df$st = unlist(Map(function(l) {as.character(l$st[[1]][1])}, M4))
df$n = unlist(Map(function(l) {c(l$n[[1]][1]) }, M4))
df$type = unlist(Map(function(l) {as.character(l$type[[1]][1])}, M4))
df$h = unlist(Map(function(l) {c(l$h[[1]][1]) }, M4))
df$period = unlist(Map(function(l) {as.character(l$period[[1]][1])}, M4))
```

```
str(df)
```

```
## 'data.frame':   100000 obs. of  5 variables:
## $ st      : chr  "D1" "D2" "D3" "D4" ...
## $ n       : int  1006 1006 130 169 156 1006 1006 999 999 674 ...
## $ type    : chr  "Macro" "Macro" "Macro" "Macro" ...
## $ h       : int  14 14 14 14 14 14 14 14 14 14 ...
## $ period  : chr  "Daily" "Daily" "Daily" "Daily" ...
```

```
table(df$period)
```

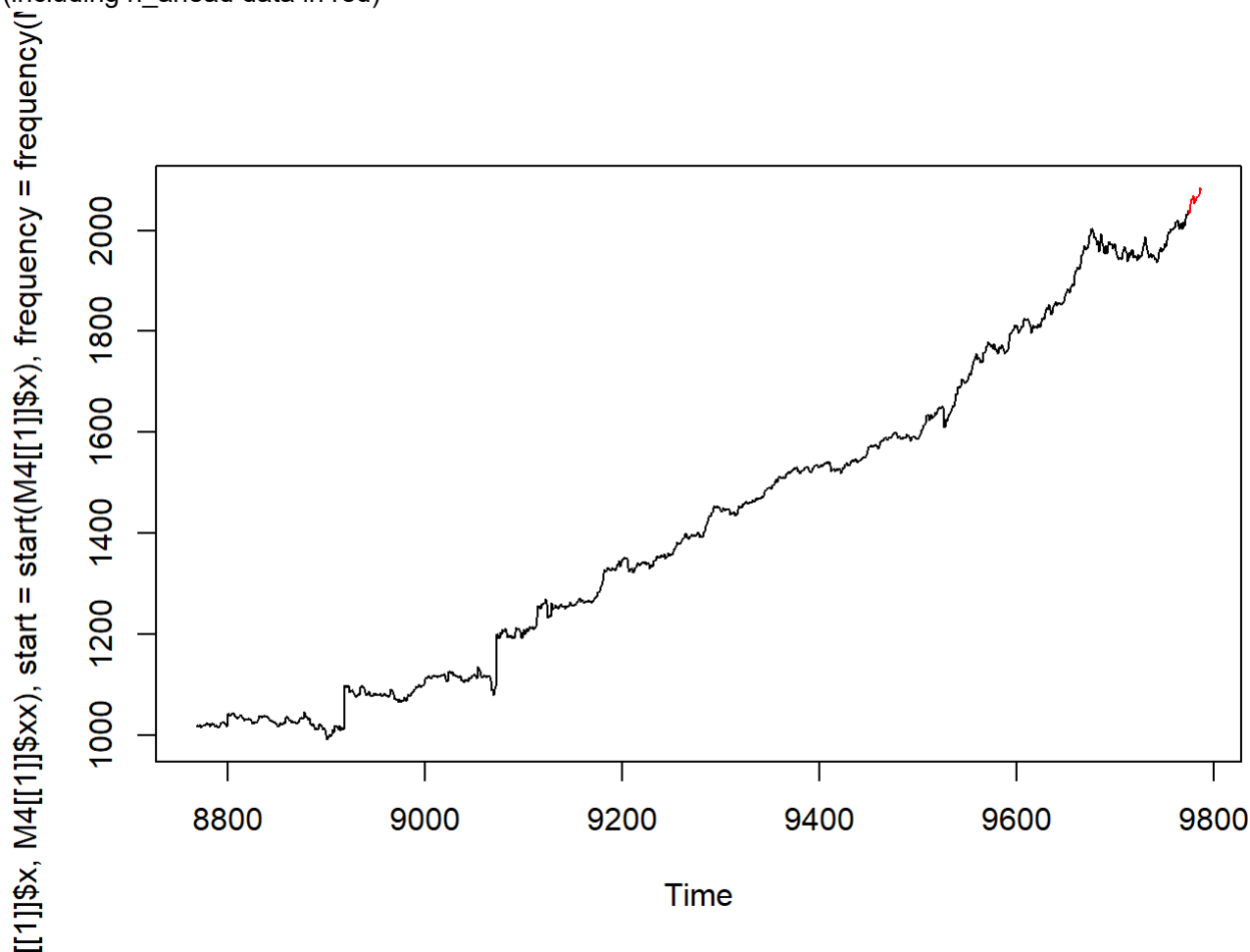
##	Daily	Hourly	Monthly	Quarterly	Weekly	Yearly
##	4227	414	48000	24000	359	23000

Extract M4 data: yearly, quarterly, monthly, daily, hourly

```
yearly_M4 = Filter(function(l) l$period == "Yearly", M4)
quarterly_M4 = Filter(function(l) l$period == "Quarterly", M4)
monthly_M4 = Filter(function(l) l$period == "Monthly", M4)
weekly_M4 = Filter(function(l) l$period == "Weekly", M4)
hourly_M4 = Filter(function(l) l$period == "Hourly", M4)
daily_M4 = Filter(function(l) l$period == "Daily", M4)
```

Plot the first sample

(including n Ahead data in red)



Extract one sample month to perform statistical models

Extract the first month

```
monthly_1_full = ts(c(monthly_M4[[1]]$x, monthly_M4[[1]]$xx),
                    start=start(monthly_M4[[1]]$x),
                    frequency = frequency(monthly_M4[[1]]$x))
monthly_1_train = ts(monthly_M4[[1]]$x,
                     start=start(monthly_M4[[1]]$x),
                     frequency = frequency(monthly_M4[[1]]$x))
monthly_1_test = ts(monthly_M4[[1]]$xx,
                    start=start(monthly_M4[[1]]$xx),
                    frequency = frequency(monthly_M4[[1]]$xx))
```

Explore the structure of the training and test set

```
head(monthly_1_train)
```

```
##      Jun  Jul  Aug  Sep  Oct  Nov
## 1976 8000 8350 8570 7700 7080 6520
```

```
str(monthly_1_train)
```

```
## Time-Series [1:469] from 1976 to 2015: 8000 8350 8570 7700 7080 6520 6070 6650 6830 5710 ...
```

```
head(monthly_1_test)
```

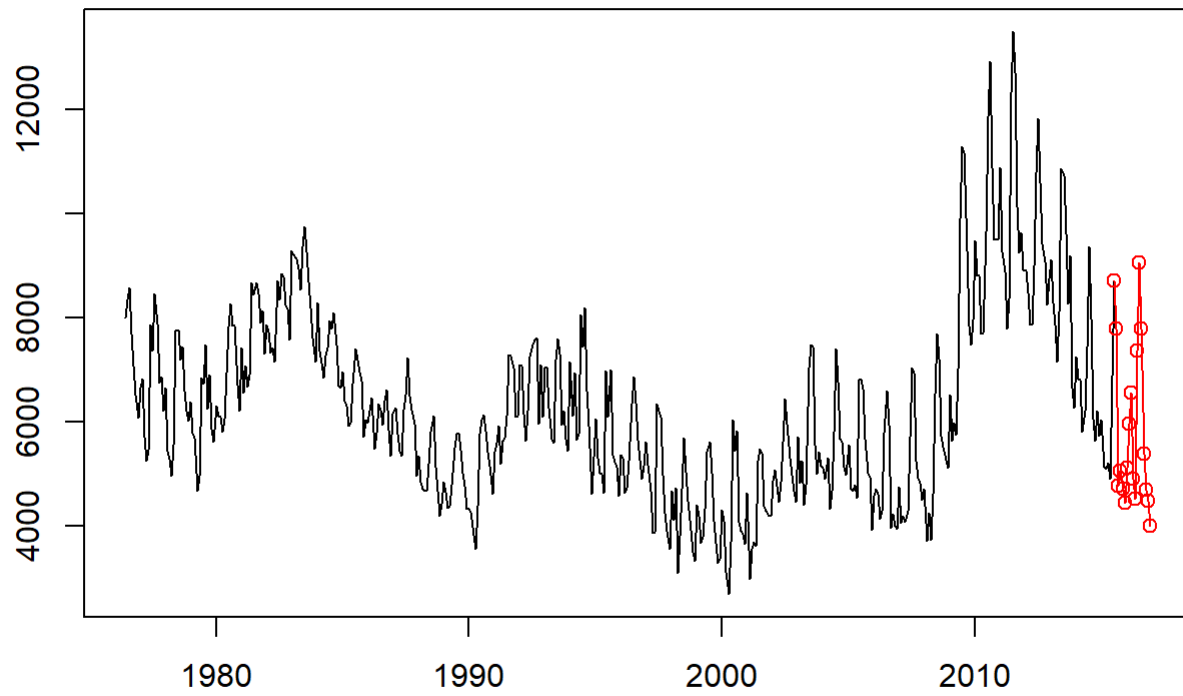
```
##      Jul  Aug  Sep  Oct  Nov  Dec
## 2015 8720 7790 4770 5060 4720 4450
```

```
str(monthly_1_test)
```

```
## Time-Series [1:18] from 2016 to 2017: 8720 7790 4770 5060 4720 4450 5120 5960 6560 4900 ...
```

plot the monthly sample: including full, training and test set

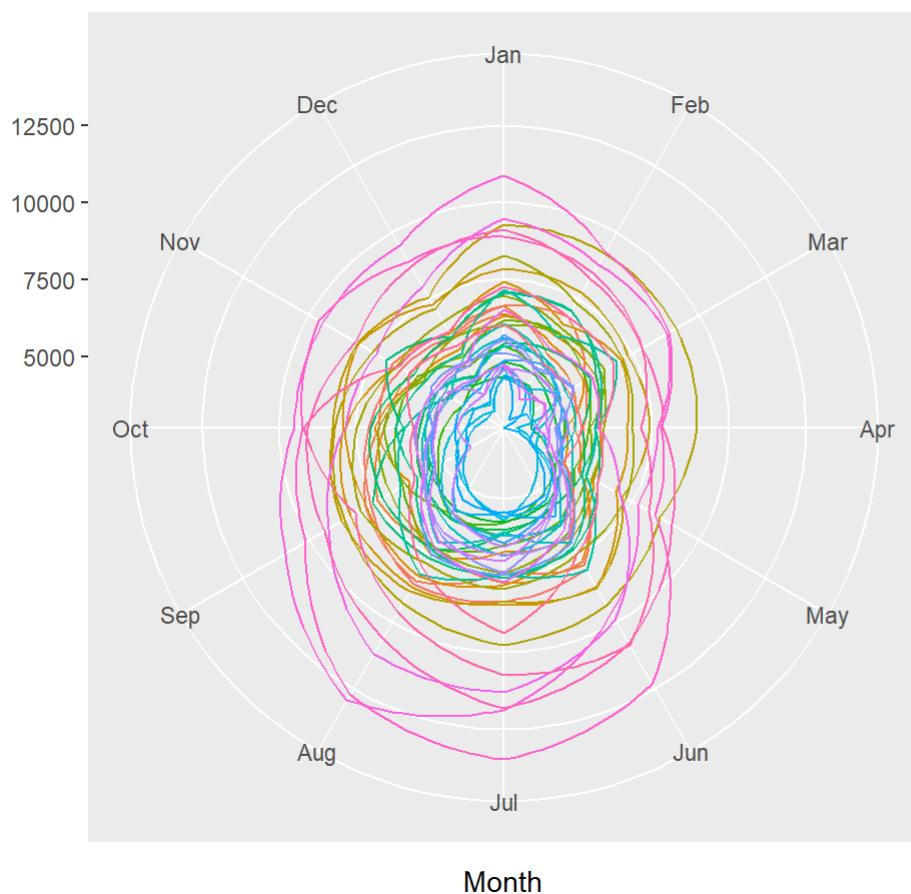
```
plot(ts(monthly_1_full,
        start = start(monthly_1_full),
        frequency = frequency(monthly_1_full)),
     type = 'l', col = 'black', ylab = '', xlab = '')
lines(monthly_1_test, col = 'red', type = 'o', xlab = '')
```



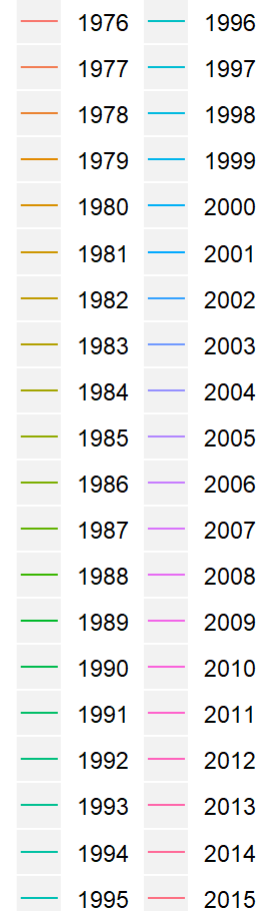
Produce a polar coordinate season plot

```
ggseasonplot(monthly_1_train, polar = T)
```

Seasonal plot: monthly_1_train

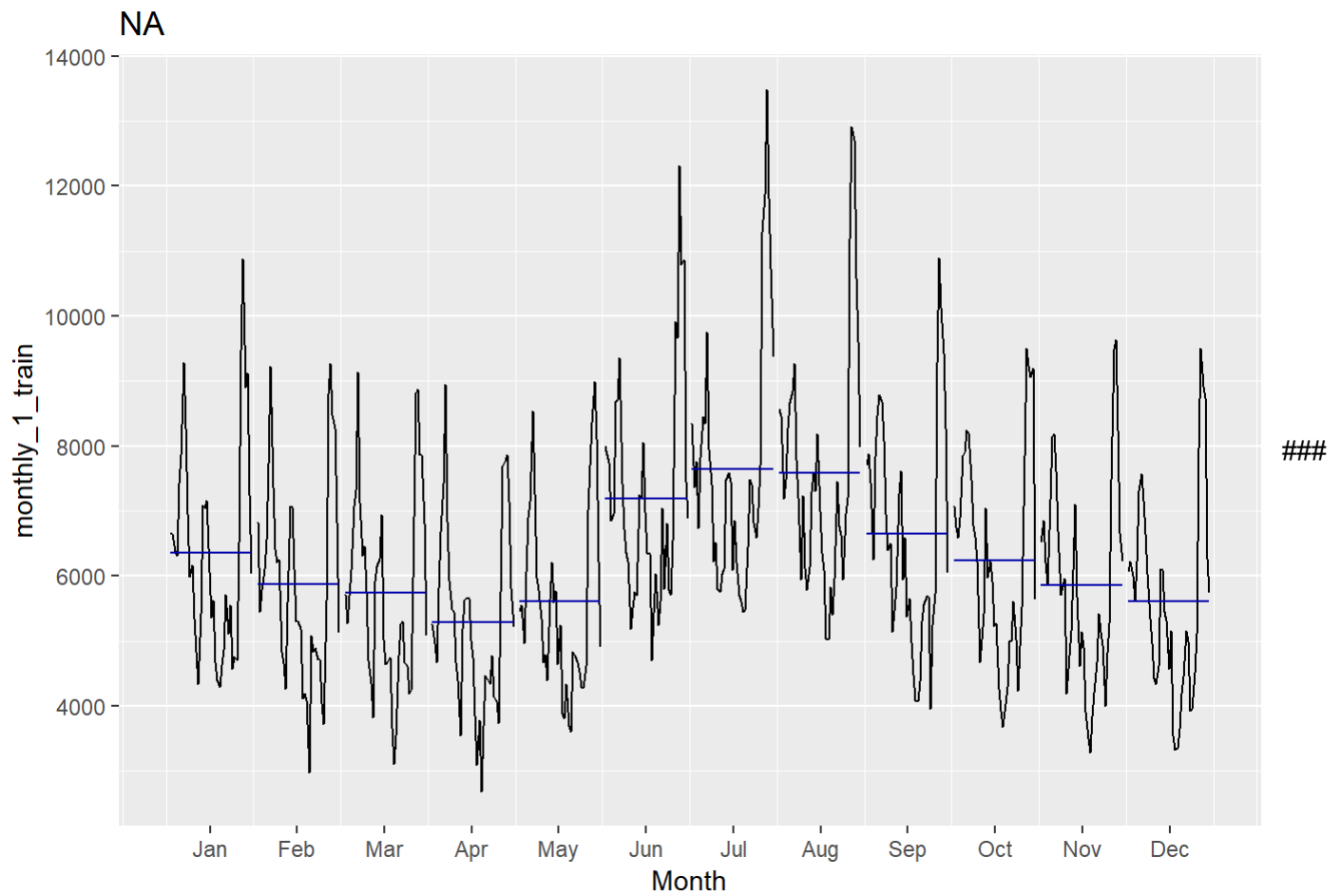


year



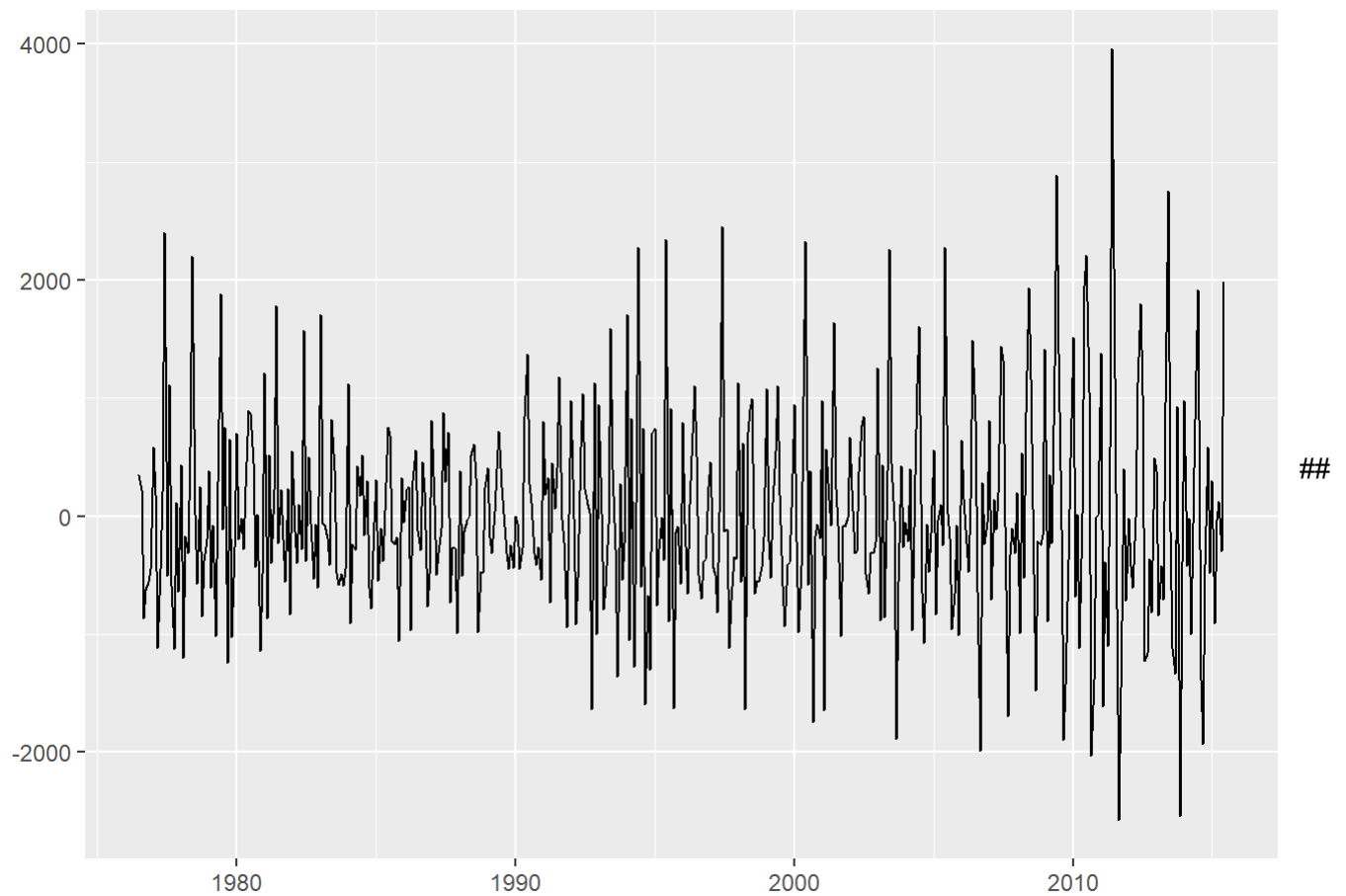
Create subseries plot that comprises mini time plots for each season

```
ggsubseriesplot(monthly_1_train)
```



Plot the monthly sample: Removing trend

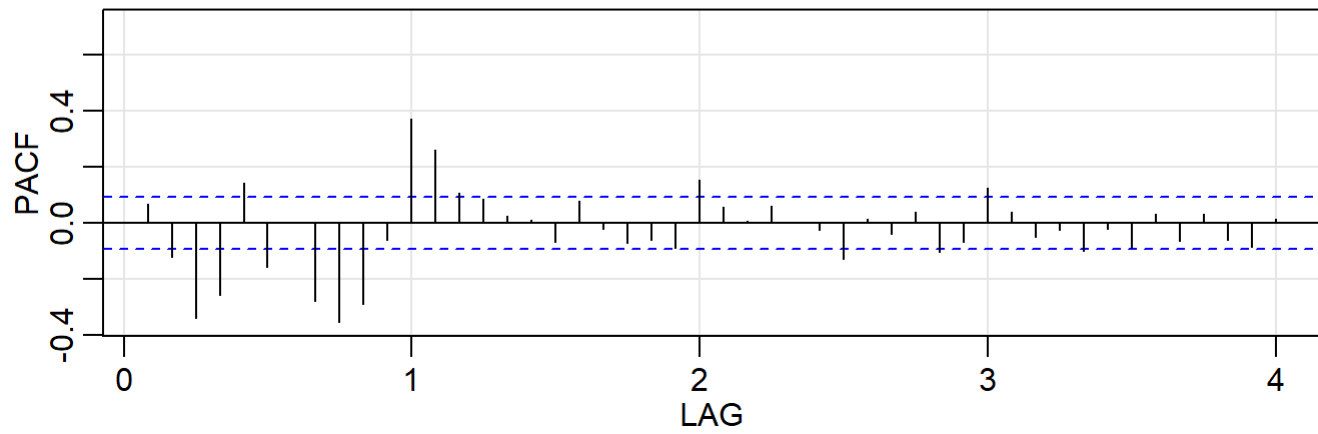
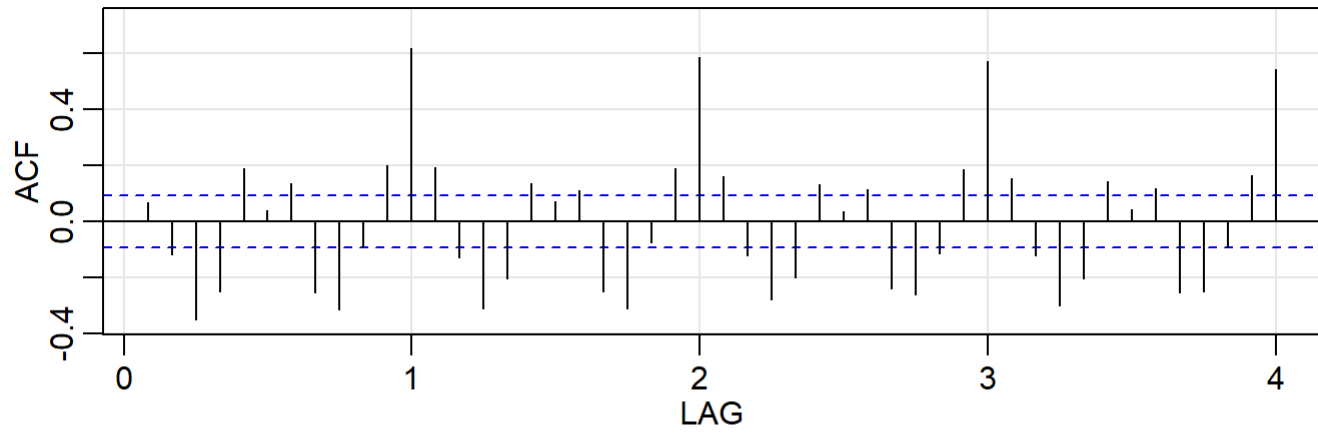
```
autoplot(diff(monthly_1_train))
```



Explore time series patterns: Trend, seasonal, or cyclic

```
acf2(diff(monthly_1_train))
```

Series: diff(monthly_1_train)



##		ACF	PACF
##	[1,]	0.07	0.07
##	[2,]	-0.12	-0.12
##	[3,]	-0.35	-0.34
##	[4,]	-0.25	-0.26
##	[5,]	0.19	0.14
##	[6,]	0.04	-0.16
##	[7,]	0.13	0.00
##	[8,]	-0.25	-0.28
##	[9,]	-0.32	-0.35
##	[10,]	-0.09	-0.29
##	[11,]	0.20	-0.06
##	[12,]	0.62	0.37
##	[13,]	0.19	0.26
##	[14,]	-0.13	0.11
##	[15,]	-0.31	0.08
##	[16,]	-0.20	0.03
##	[17,]	0.14	0.01
##	[18,]	0.07	-0.07
##	[19,]	0.11	0.08
##	[20,]	-0.25	-0.02
##	[21,]	-0.31	-0.07
##	[22,]	-0.07	-0.06
##	[23,]	0.19	-0.09
##	[24,]	0.59	0.16
##	[25,]	0.16	0.06
##	[26,]	-0.12	0.01
##	[27,]	-0.28	0.06
##	[28,]	-0.20	0.00
##	[29,]	0.13	-0.02
##	[30,]	0.04	-0.13
##	[31,]	0.11	0.01
##	[32,]	-0.24	-0.04
##	[33,]	-0.26	0.04
##	[34,]	-0.11	-0.10
##	[35,]	0.19	-0.07
##	[36,]	0.57	0.13
##	[37,]	0.15	0.04
##	[38,]	-0.12	-0.05
##	[39,]	-0.30	-0.02
##	[40,]	-0.20	-0.10
##	[41,]	0.14	-0.02
##	[42,]	0.04	-0.09
##	[43,]	0.12	0.03
##	[44,]	-0.25	-0.07
##	[45,]	-0.25	0.03
##	[46,]	-0.09	-0.06
##	[47,]	0.16	-0.08
##	[48,]	0.54	0.01