1 R

1.

```
library(itsmr)
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:itsmr':
##
##     deaths
```

```
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 4.0.4
```

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

```
##
## Attaching package: 'tseries'
```

```
## The following object is masked from 'package:itsmr':
##
##     arma
```

```
library(nortest)
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.0.4
```

```
##
## Attaching package: 'forecast'
```

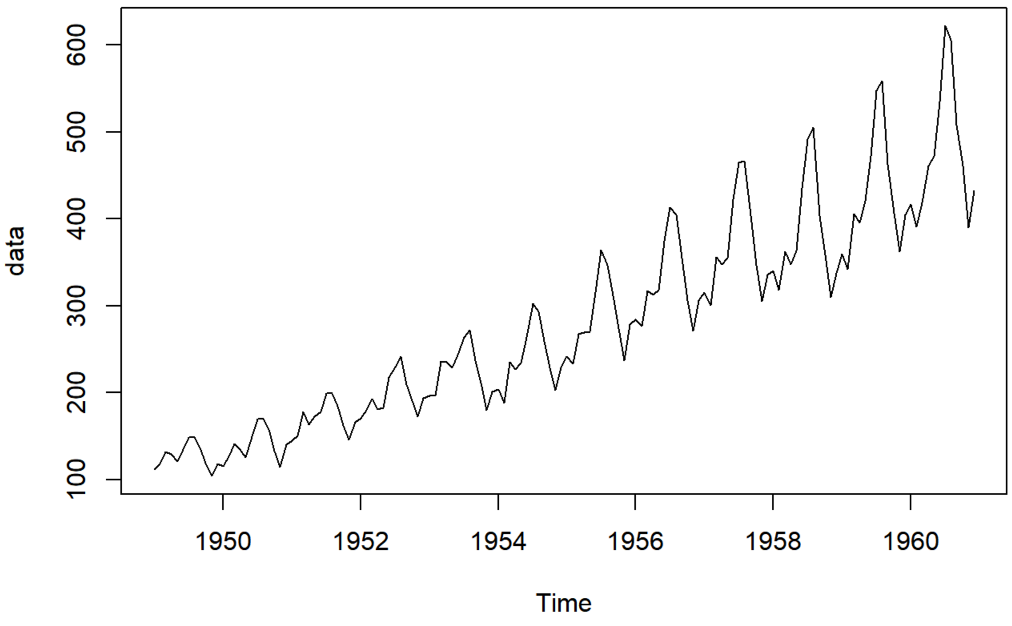
```
## The following object is masked from 'package:itsmr':
##
##     forecast
```

2.

AirPassengers (1949 - 1960) 3 .

- (trend) ; 1949
- (heteroscedasticity) ; .
- (seasonality) ; 12 .

```
data = AirPassengers
plot(data)
```

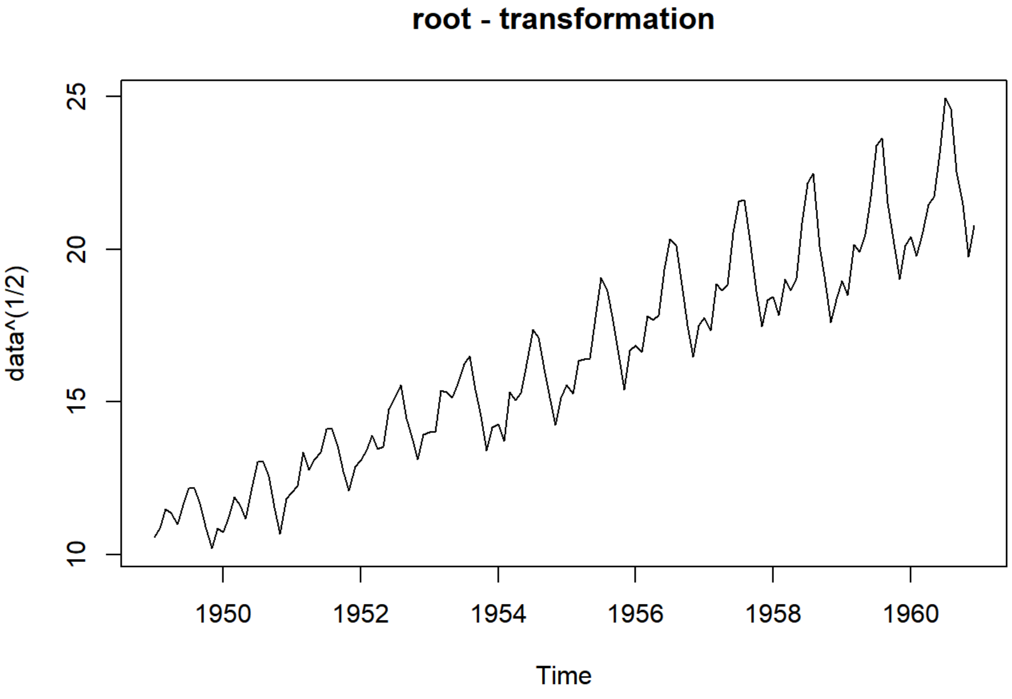


3.

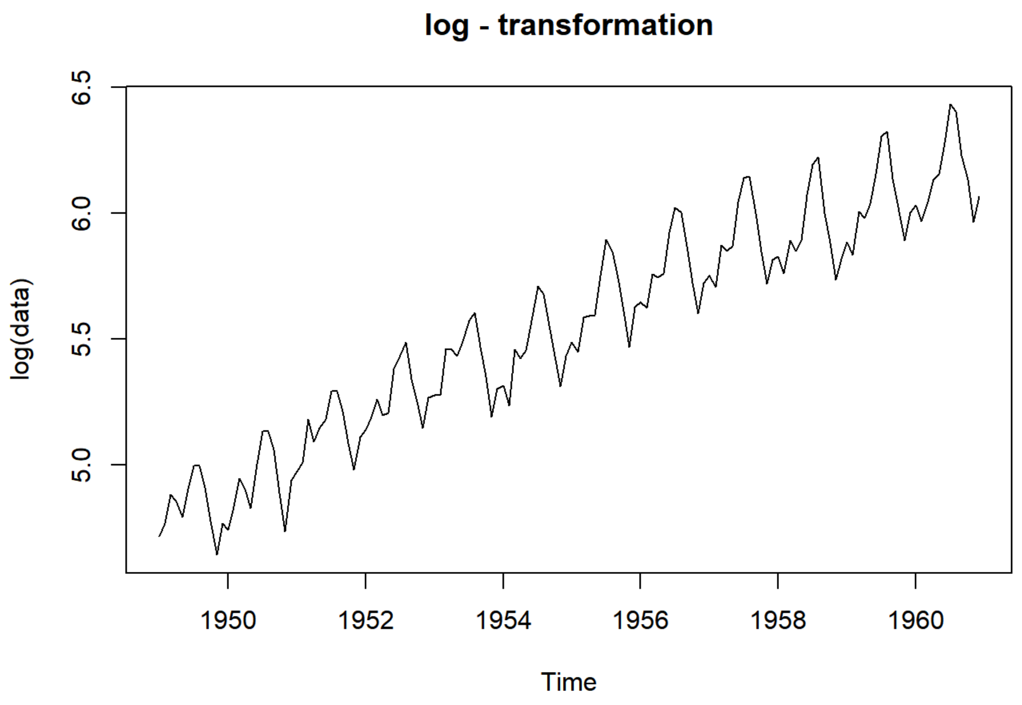
, , Box-Cox .

, .

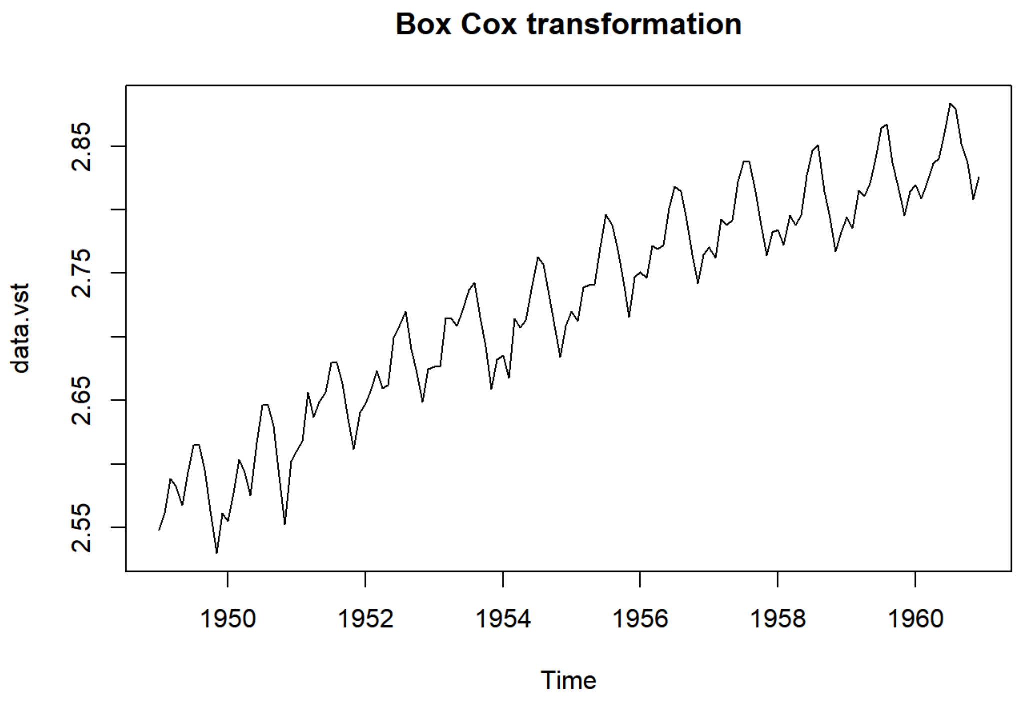
```
plot(data^(1/2)); title('root - transformation')
```



```
plot(log(data)); title('log - transformation')
```



```
lambda <- BoxCox.lambda(data)
data.vst<-BoxCox(data,lambda)
plot(data.vst);title("Box Cox transformation")
```

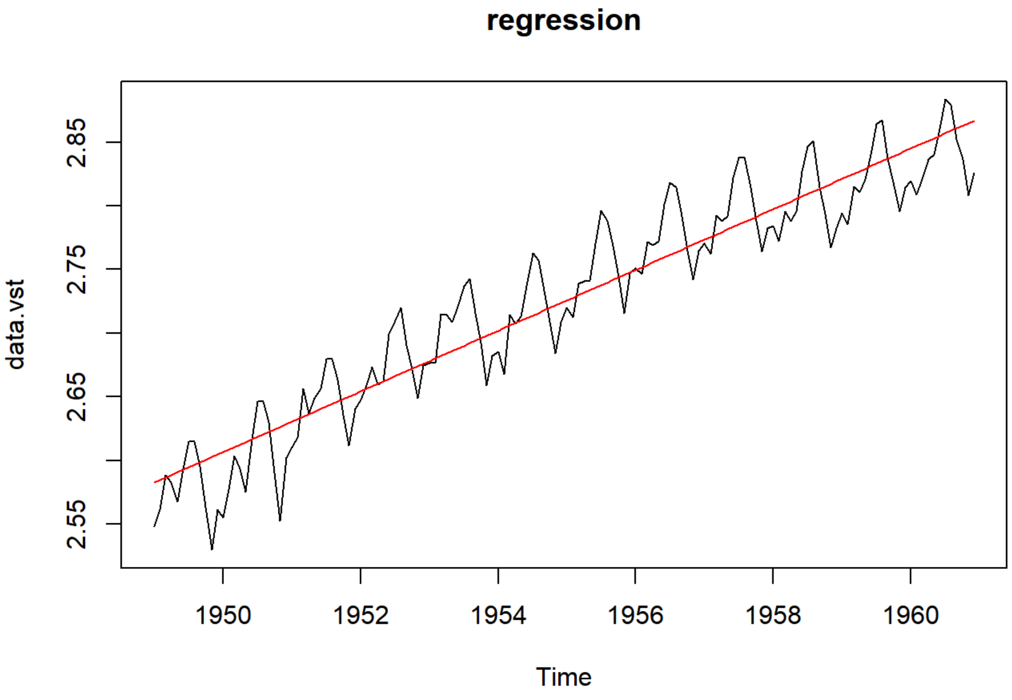


4.

.

1 .

```
n = length(data.vst)
x = seq(1,n,1)
Tt = lm(data.vst~1+x)

plot.ts(data.vst)
title('regression')
xa = as.vector(time(data.vst))
lines(xa, Tt$fitted.values,col='red')
```



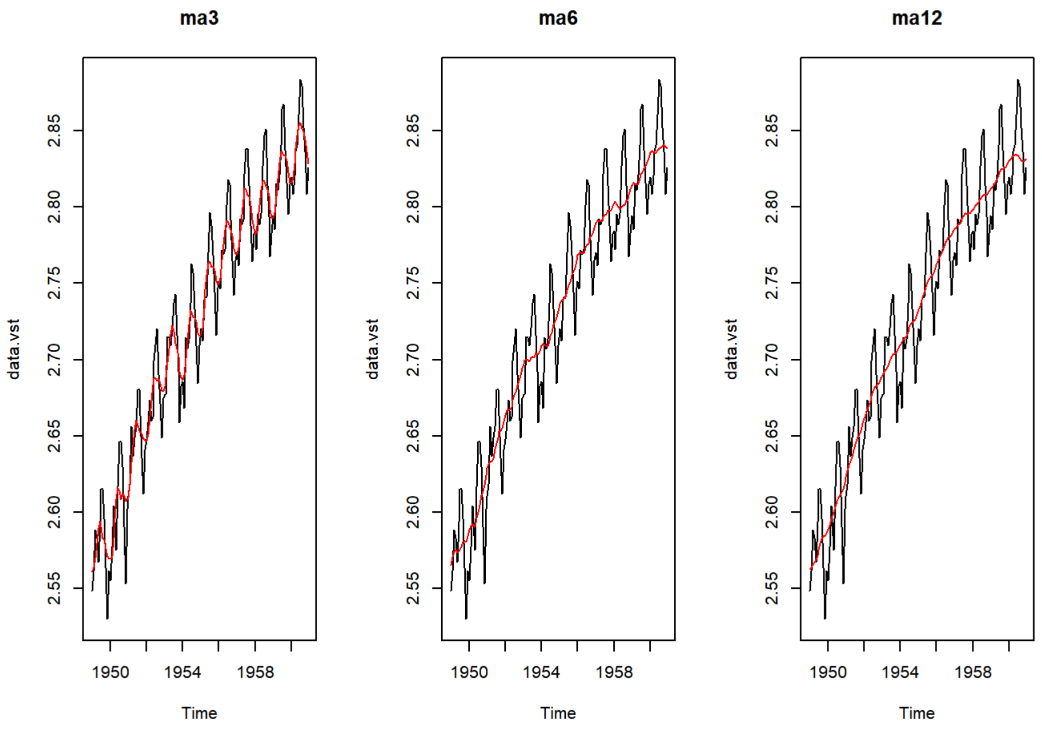
```
dtrend <- data.vst - Tt$fitted.values
```

5.

.

`smooth.ma` , .

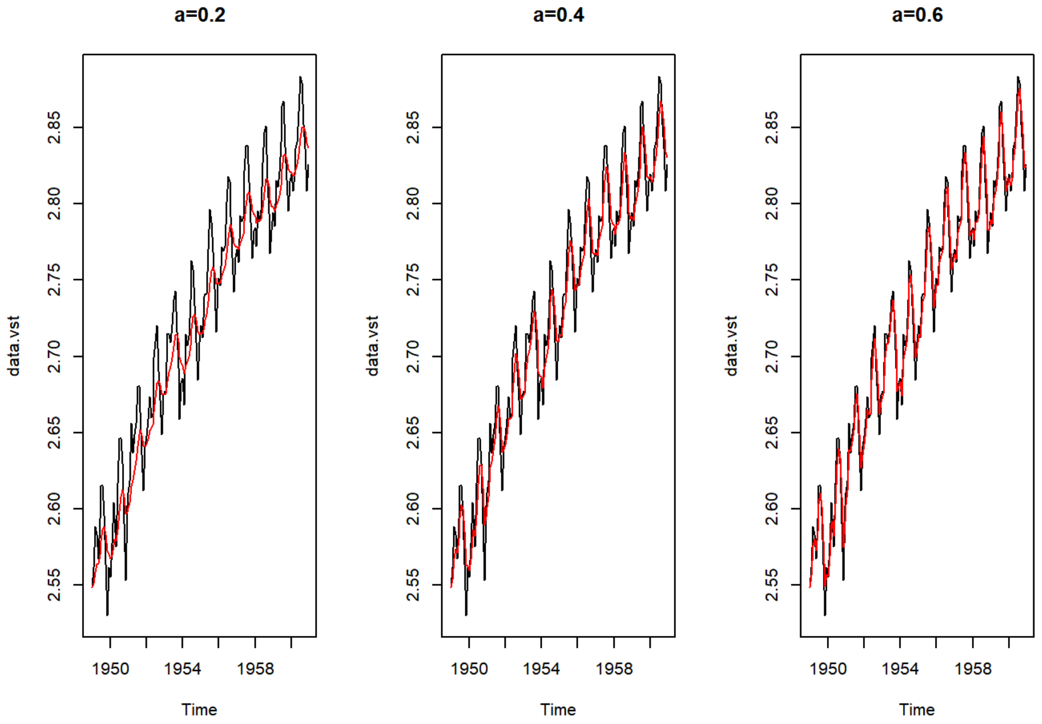
```
ma3 = smooth.ma(data.vst, q=3)
ma6 = smooth.ma(data.vst, q=6)
ma12 = smooth.ma(data.vst, q=12)

par(mfrow = c(1,3))
plot.ts(data.vst)
title('ma3')
lines(xa,ma3,col = 'red')

plot.ts(data.vst)
title('ma6')
lines(xa,ma6,col = 'red')

plot.ts(data.vst)
title('ma12')
lines(xa,ma12,col = 'red')
```



- q .

  `smooth.exp` .

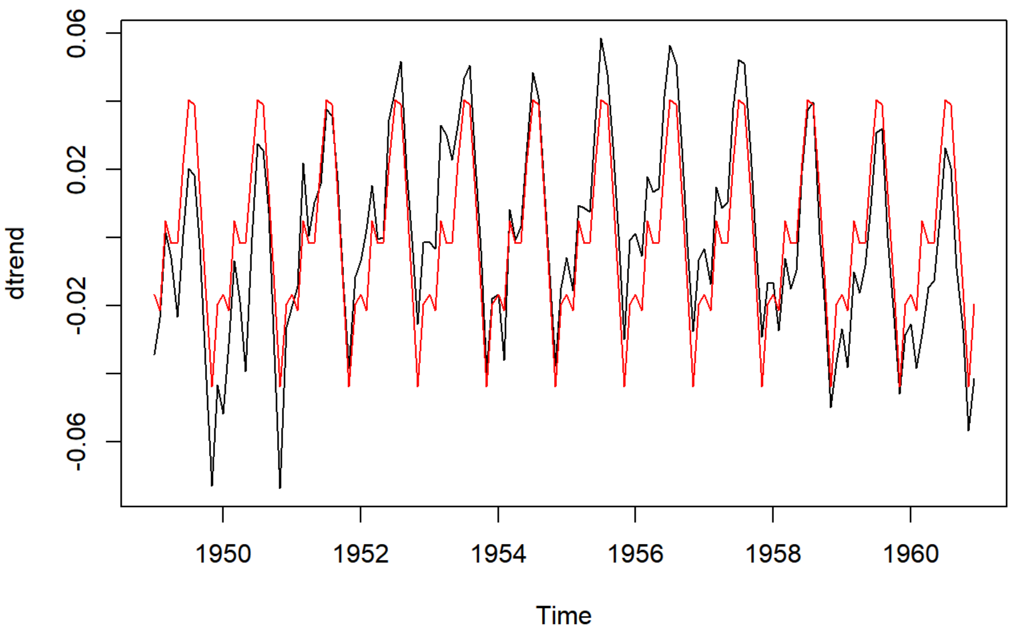
```
ex2 = smooth.exp(data.vst,.2)
ex4 = smooth.exp(data.vst,.4)
ex6 = smooth.exp(data.vst,.6)

par(mfrow = c(1,3))

plot.ts(data.vst)
lines(xa,ex2,col = 'red');title("a=0.2")

plot.ts(data.vst)
lines(xa,ex4,col = 'red');title("a=0.4")

plot.ts(data.vst)
lines(xa,ex6,col = 'red');title("a=0.6")
```



- a .

  `season` .

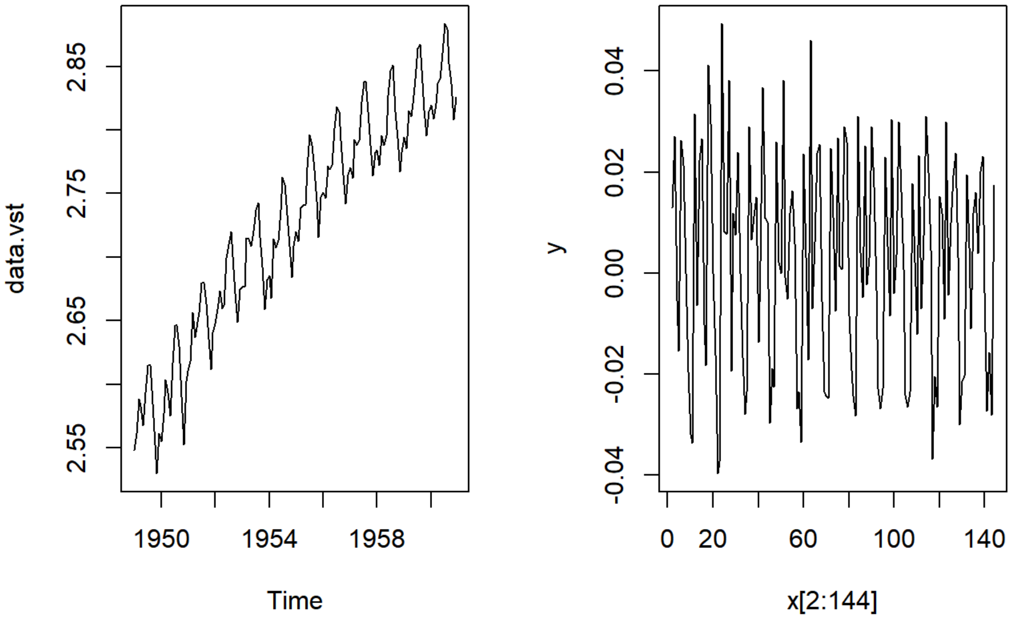
```
season.avg = season(dtrend, d=12)
plot.ts(dtrend)
lines(xa,season.avg,col = 'red')
```



6.

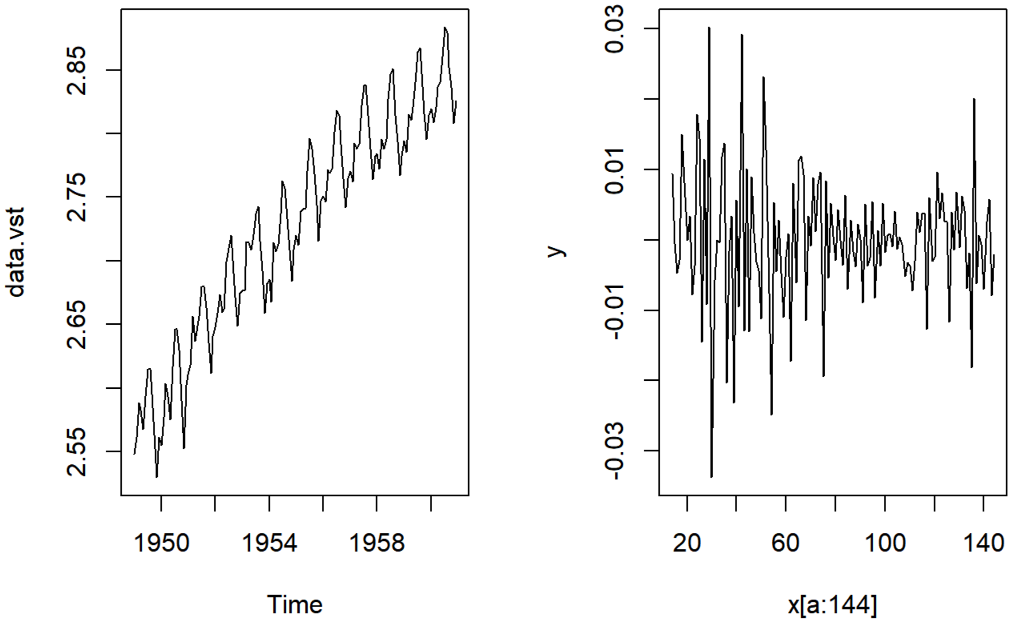
. `diff` .

```
y = diff(data.vst)

par(mfrow = c(1,2))
plot.ts(data.vst)
plot(x[2:144],y,type = 'l')
```



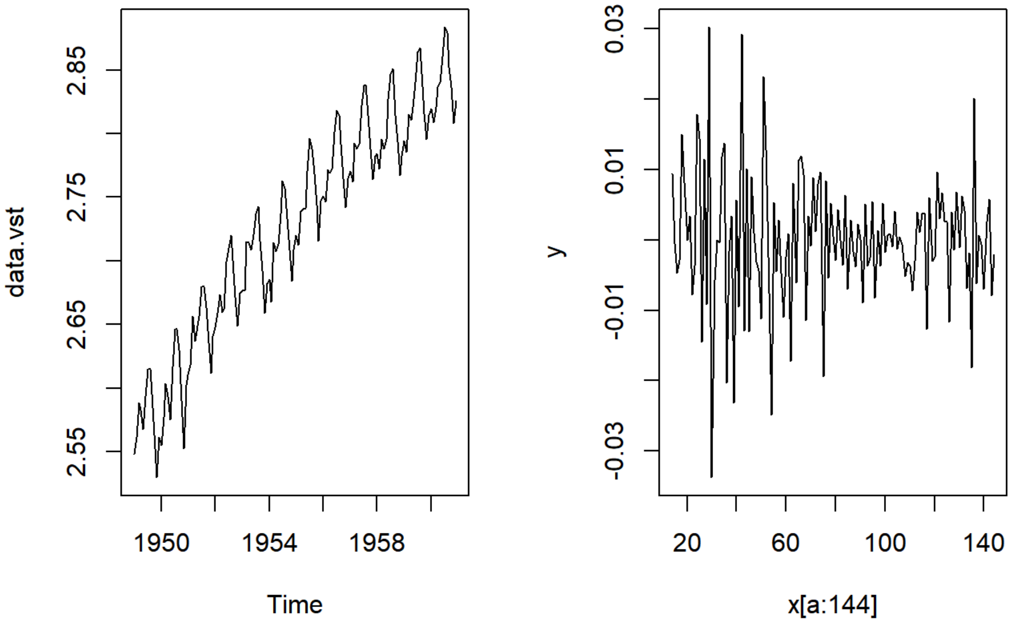
`diff` `lag` .

```
diff12 = diff(data.vst, lag = 12)
y = diff(diff12)
par(mfrow = c(1,2))
plot.ts(data.vst)
a <- 145-length(y)

plot(x[a:144], y,type = 'l')
```



1

```
diff12 = diff(data.vst, lag = 12)
y = diff(diff12)
a <- 145-length(y)

par(mfrow = c(1,2))
plot.ts(data.vst)
plot(x[a:144],y,type = 'l')
```

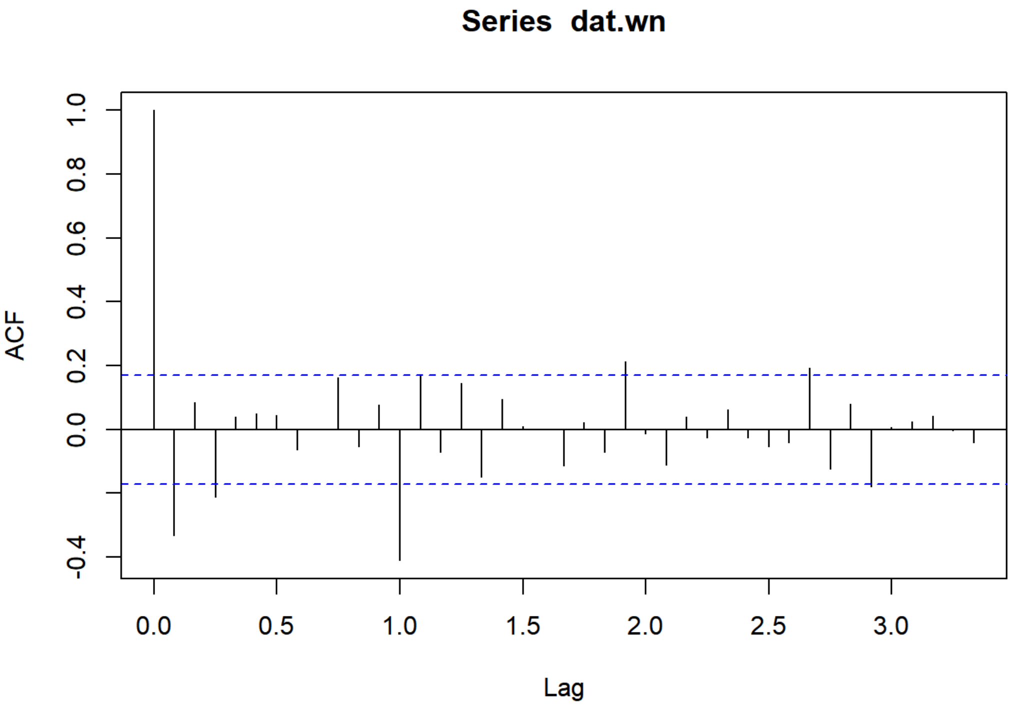


7.

ACF

ACF

```
dat.wn <- y
acf(dat.wn,lag.max = 40)
```

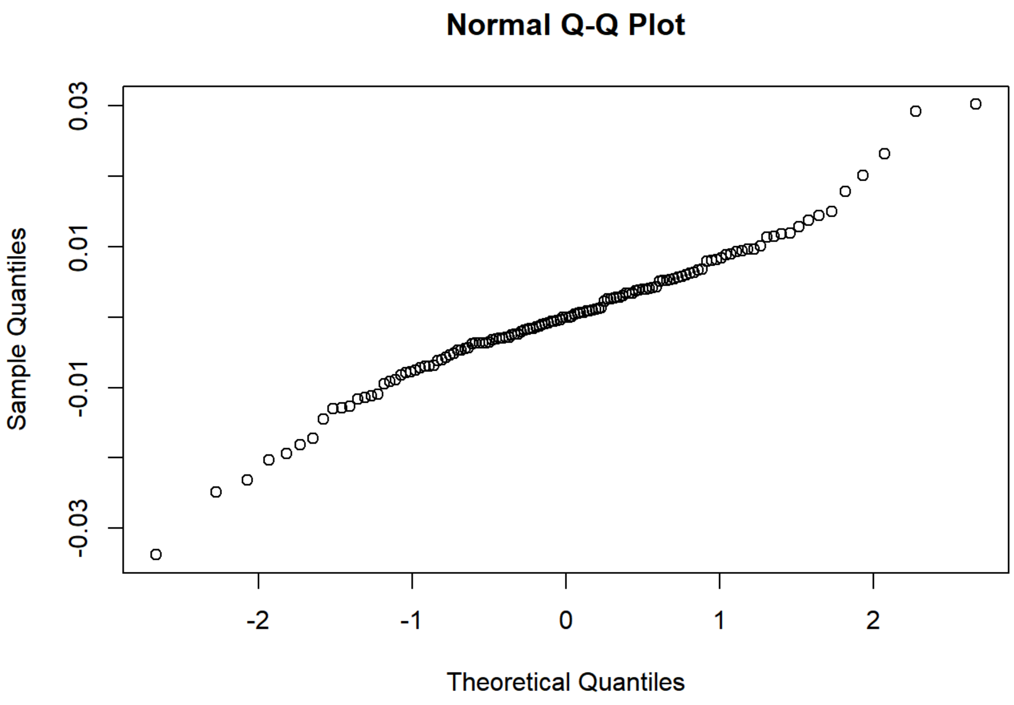


, .

```
Box.test(dat.wn,lag=10,type="Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  dat.wn
## X-squared = 27.347, df = 10, p-value = 0.002294
```

```
qqnorm(dat.wn)
```



```
jarque.bera.test(dat.wn)
```

```
##
##  Jarque Bera Test
##
## data:  dat.wn
## X-squared = 17.246, df = 2, p-value = 0.0001799
```

adf

- : .
- : .

```
adf.test(dat.wn)
```

```
## Warning in adf.test(dat.wn): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  dat.wn
## Dickey-Fuller = -5.2679, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

kpss

- : .
- : .

```
kpss.test(dat.wn)
```

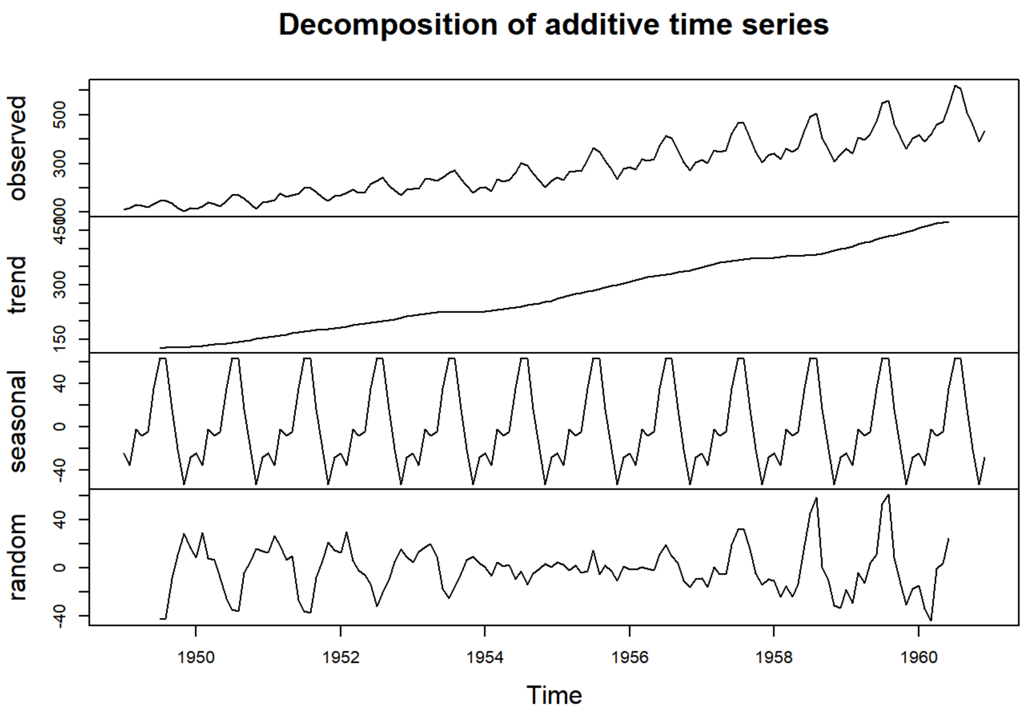
```
## Warning in kpss.test(dat.wn): p-value greater than printed p-value
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  dat.wn
## KPSS Level = 0.086789, Truncation lag parameter = 4, p-value = 0.1
```

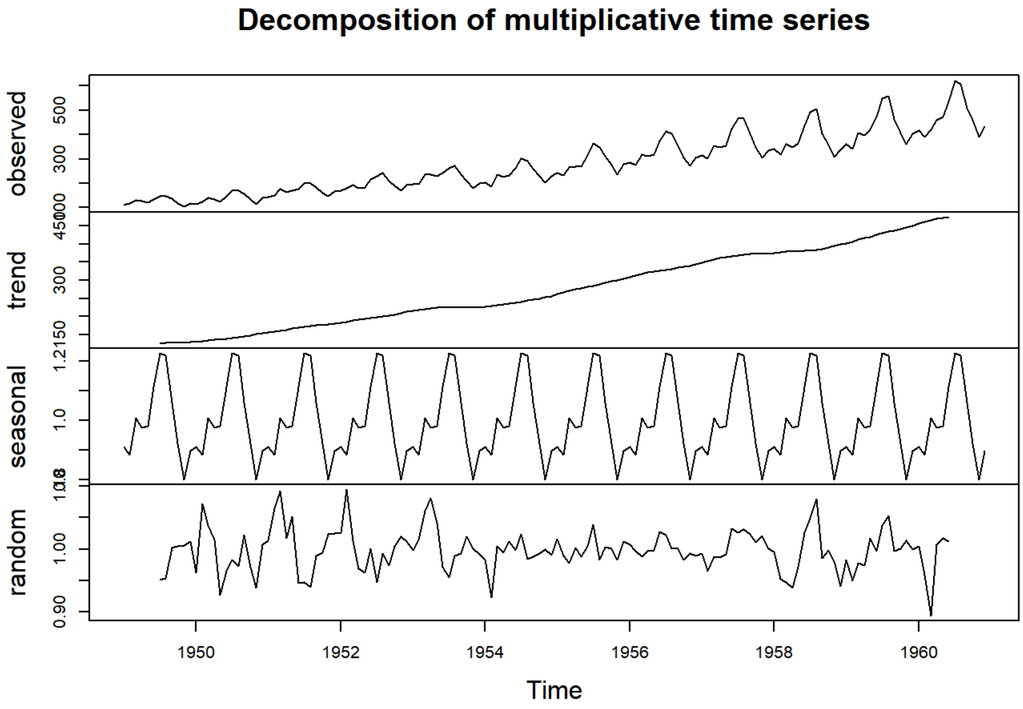
8.Classical Deomposition

`decompose` , , , .

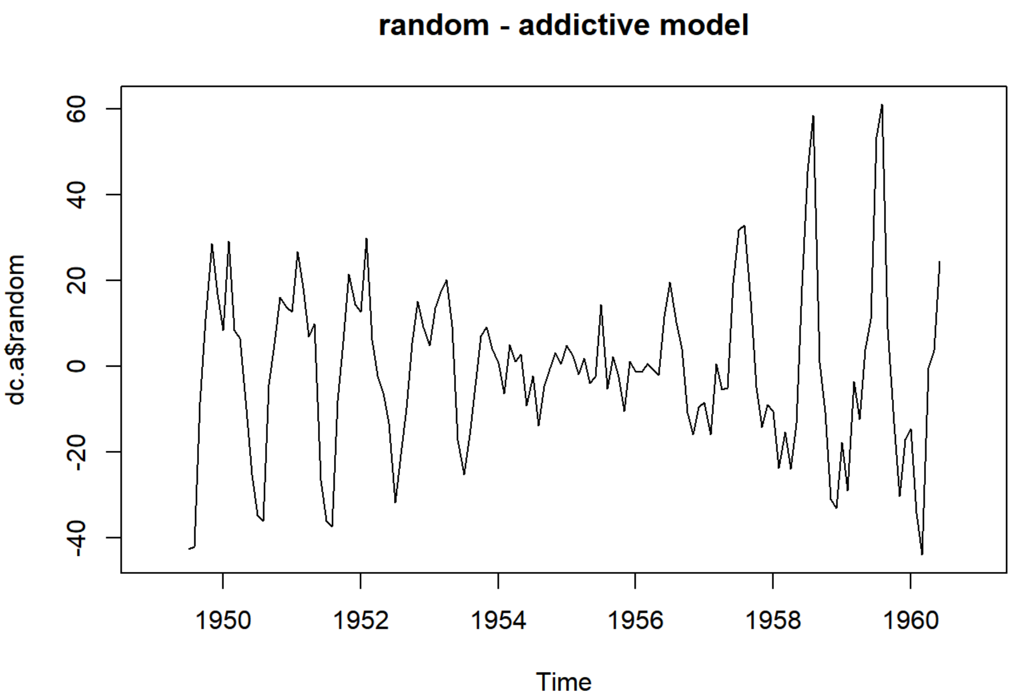
```
plot(decompose(data))
```



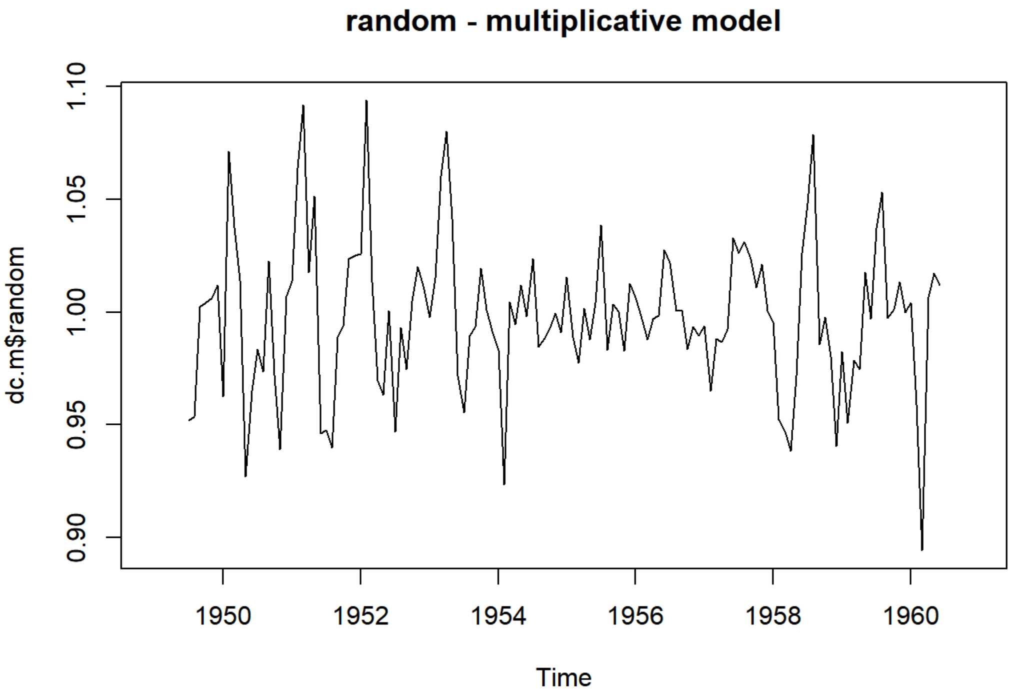
```
plot(decompose(data, type="multiplicative"))
```



```
dc.m <- decompose(data, type="multiplicative")
dc.a <- decompose(data)
plot(dc.a$random);title("random - addictive model")
```

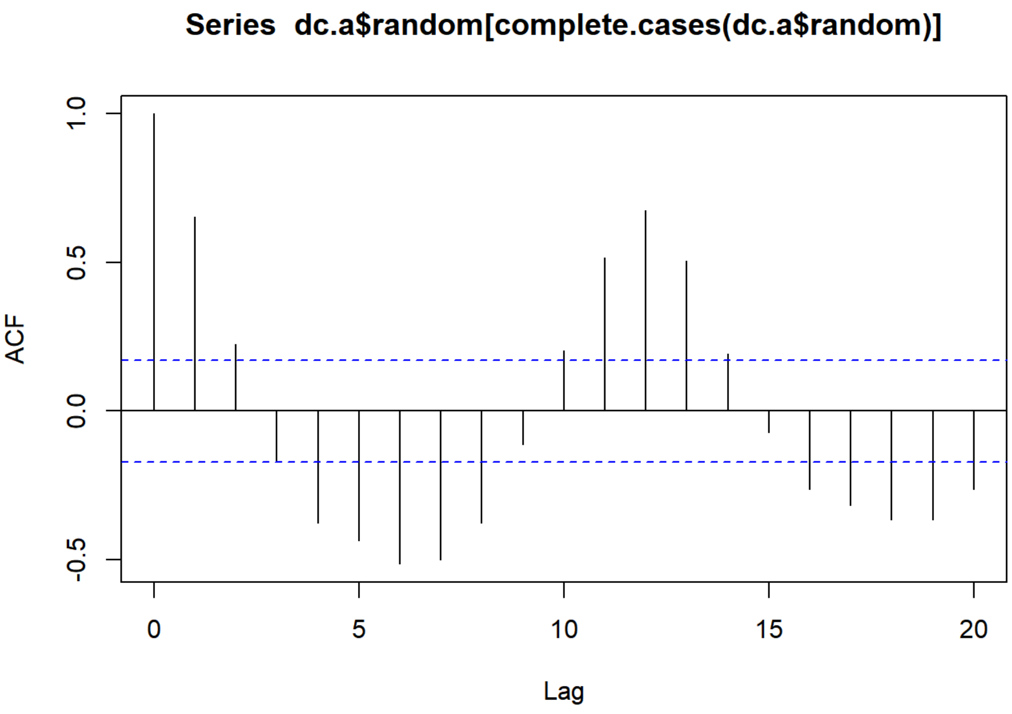


```
plot(dc.m$random);title("random - multiplicative model")
```

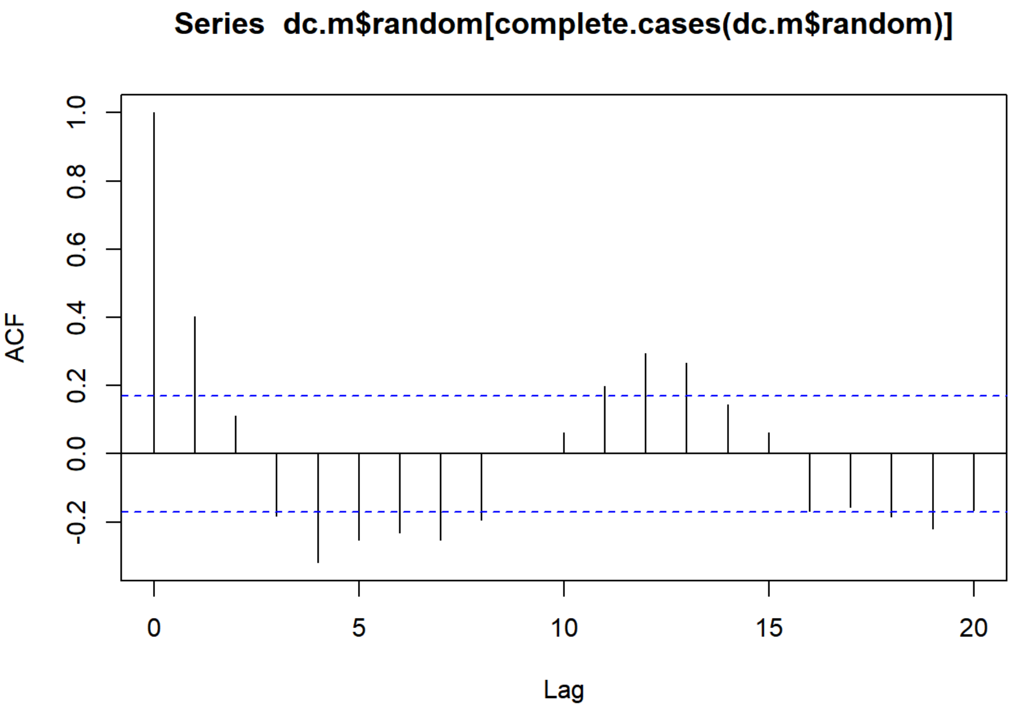


acf

```
acf(dc.a$random[complete.cases(dc.a$random)],lag.max = 20)
```



```
acf(dc.m$random[complete.cases(dc.m$random)],lag.max = 20)
```



- !!
- 2 ~!