R Notebook

```
data<- read.csv("AirPassengers.csv", header = TRUE)

install.packages("tseries") #time series package to access the time forecasting
```

```
## Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/3.6'
## (as 'lib' is unspecified)
```

```
library(tseries)
```

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

```
start(AirPassengers)
```

```
## [1] 1949    1
```

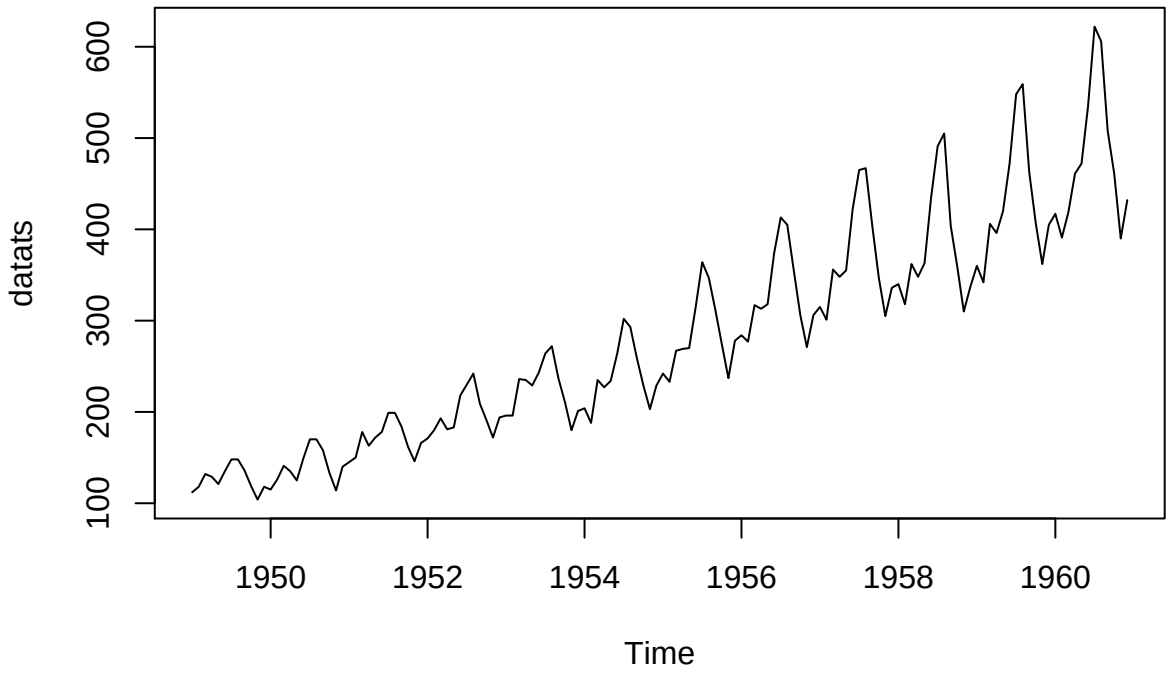
```
as.ts(AirPassengers)
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1949 112 118 132 129 121 135 148 148 136 119 104 118
## 1950 115 126 141 135 125 149 170 170 158 133 114 140
## 1951 145 150 178 163 172 178 199 199 184 162 146 166
## 1952 171 180 193 181 183 218 230 242 209 191 172 194
## 1953 196 196 236 235 229 243 264 272 237 211 180 201
## 1954 204 188 235 227 234 264 302 293 259 229 203 229
## 1955 242 233 267 269 270 315 364 347 312 274 237 278
## 1956 284 277 317 313 318 374 413 405 355 306 271 306
## 1957 315 301 356 348 355 422 465 467 404 347 305 336
## 1958 340 318 362 348 363 435 491 505 404 359 310 337
## 1959 360 342 406 396 420 472 548 559 463 407 362 405
## 1960 417 391 419 461 472 535 622 606 508 461 390 432
```

```
class(AirPassengers)
```

```
## [1] "ts"
```

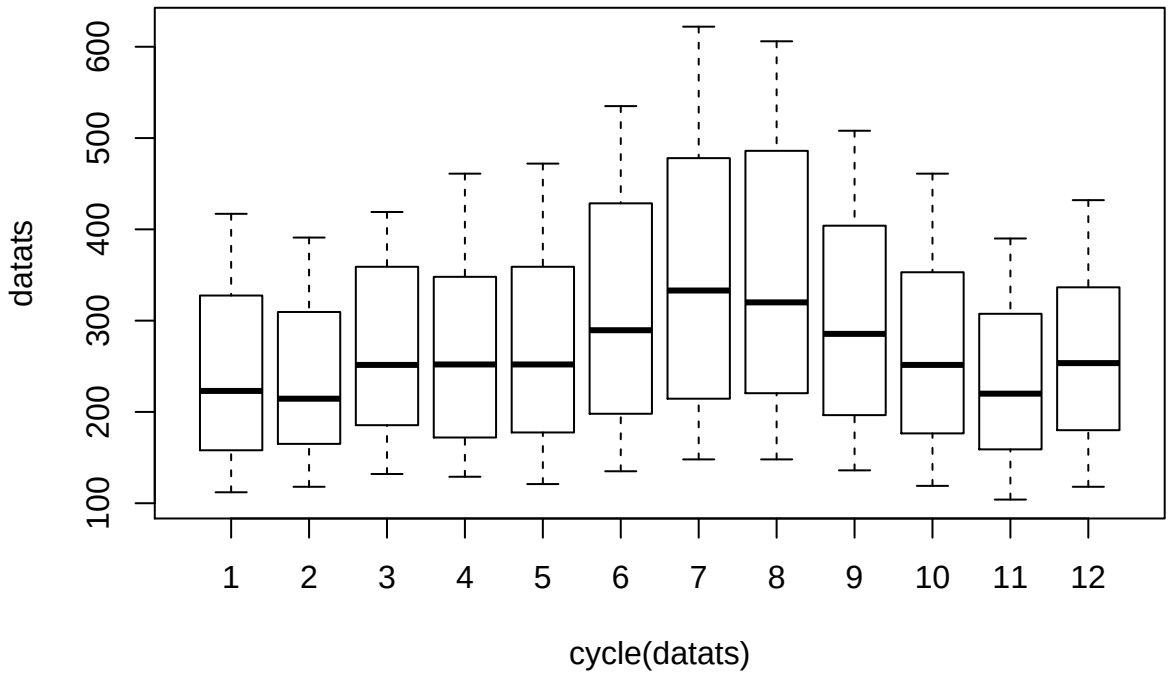
```
datats<- as.ts(AirPassengers) #as.ts is used to convert/access the timeserie.
plot(datats) #plots the graph
```



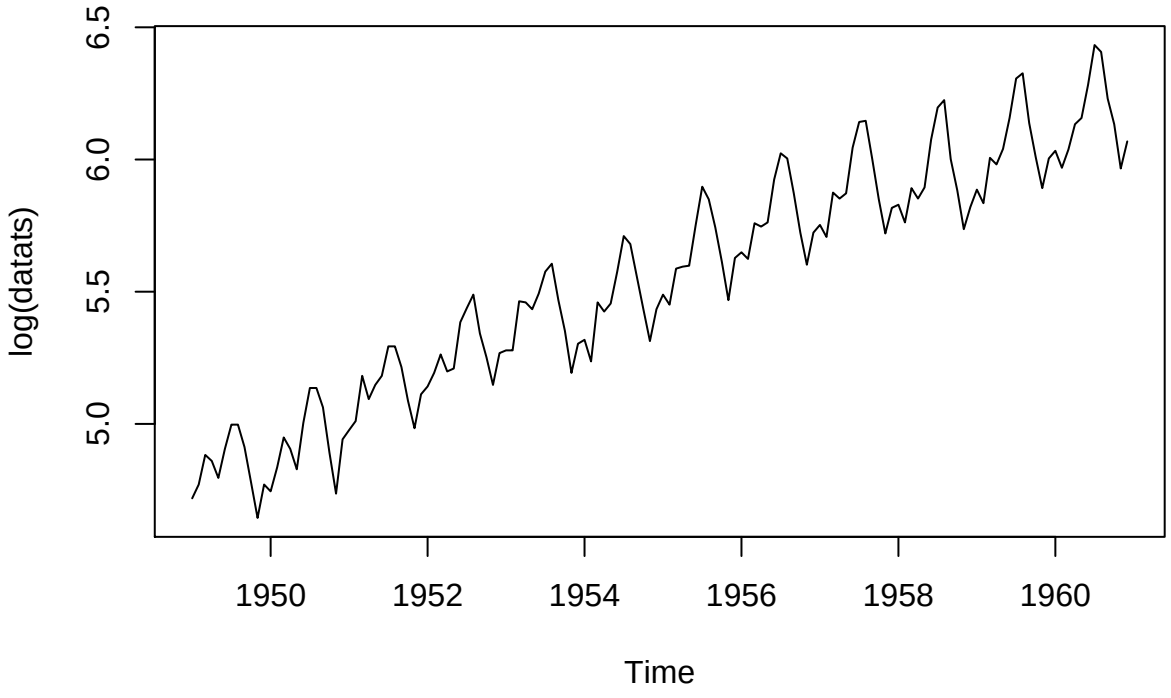

```
abline = (reg = lm(datats~time(datats))) #plots the mean line over the graph.

cycle(datats)
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1949   1   2   3   4   5   6   7   8   9  10  11  12
## 1950   1   2   3   4   5   6   7   8   9  10  11  12
## 1951   1   2   3   4   5   6   7   8   9  10  11  12
## 1952   1   2   3   4   5   6   7   8   9  10  11  12
## 1953   1   2   3   4   5   6   7   8   9  10  11  12
## 1954   1   2   3   4   5   6   7   8   9  10  11  12
## 1955   1   2   3   4   5   6   7   8   9  10  11  12
## 1956   1   2   3   4   5   6   7   8   9  10  11  12
## 1957   1   2   3   4   5   6   7   8   9  10  11  12
## 1958   1   2   3   4   5   6   7   8   9  10  11  12
## 1959   1   2   3   4   5   6   7   8   9  10  11  12
## 1960   1   2   3   4   5   6   7   8   9  10  11  12
```

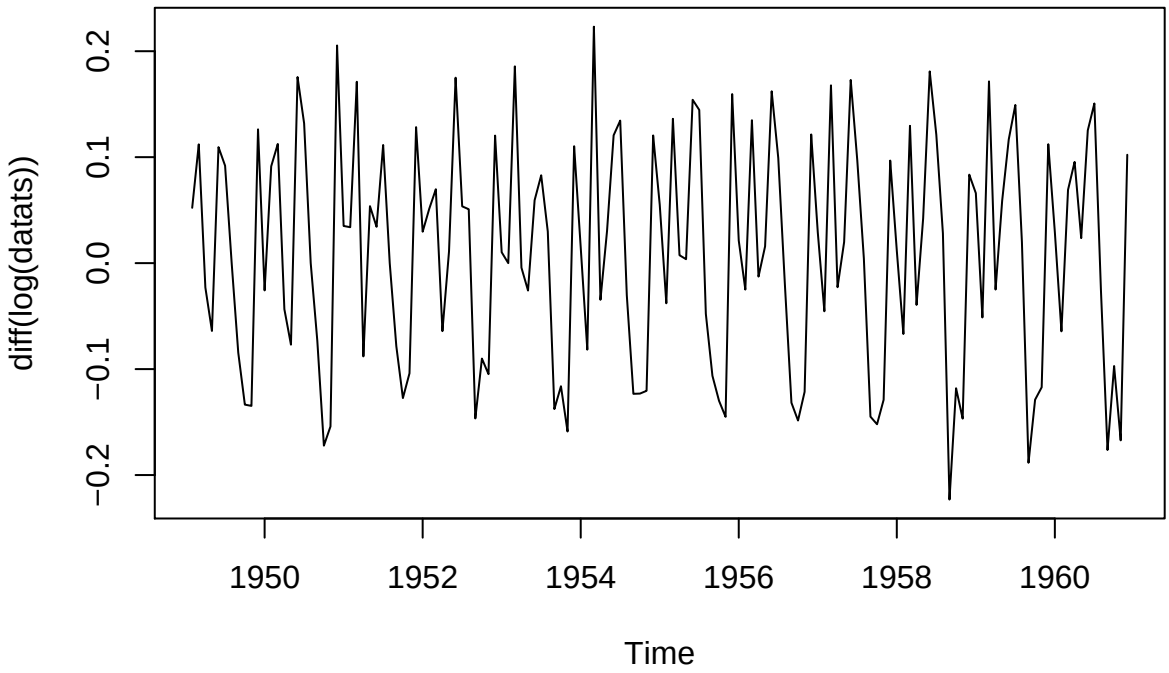
```
boxplot(datats~cycle(datats))
```



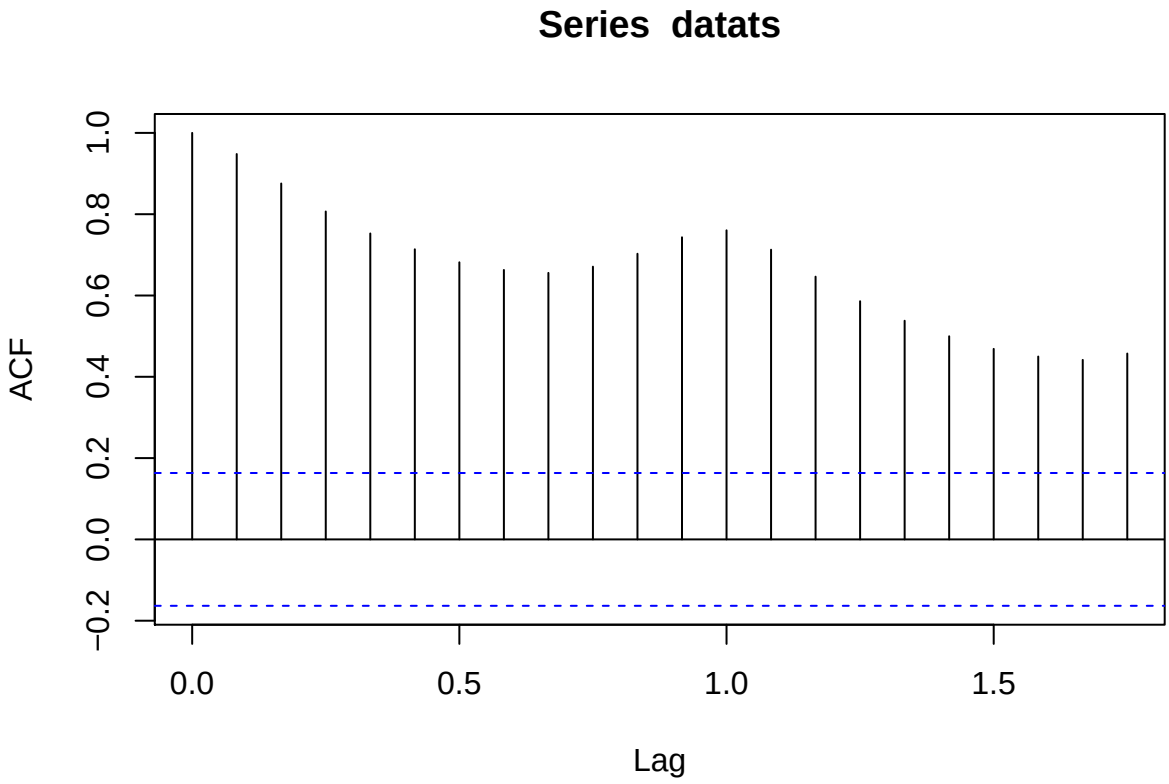

```
#to remove varience we convert it to the log by using log function
plot(log(datats))
```



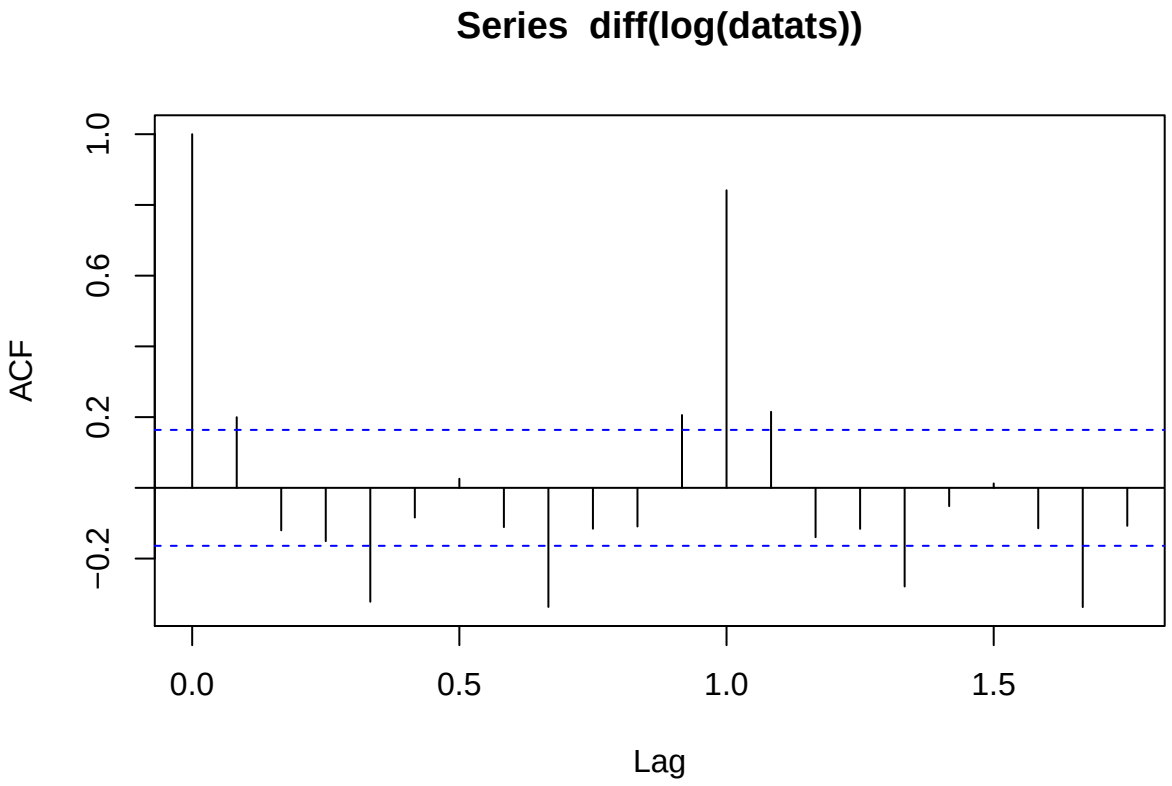

```
#we make the values stationary so that the model can be applied over by differentiating the grapg.
plot(diff(log(datats)))
```



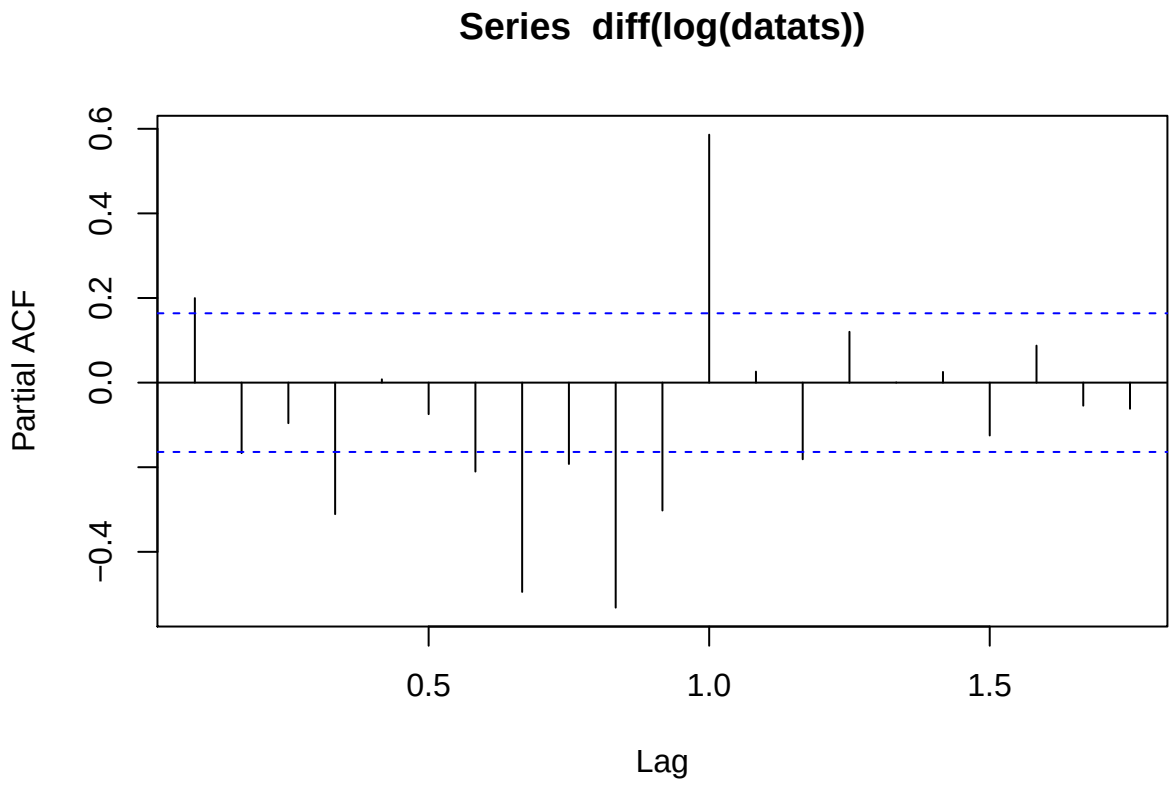

```
#To apply ARIMA we need to have
#q = Moving Average
#d = Auto regression
#p = pvalue


acf (datats)
```



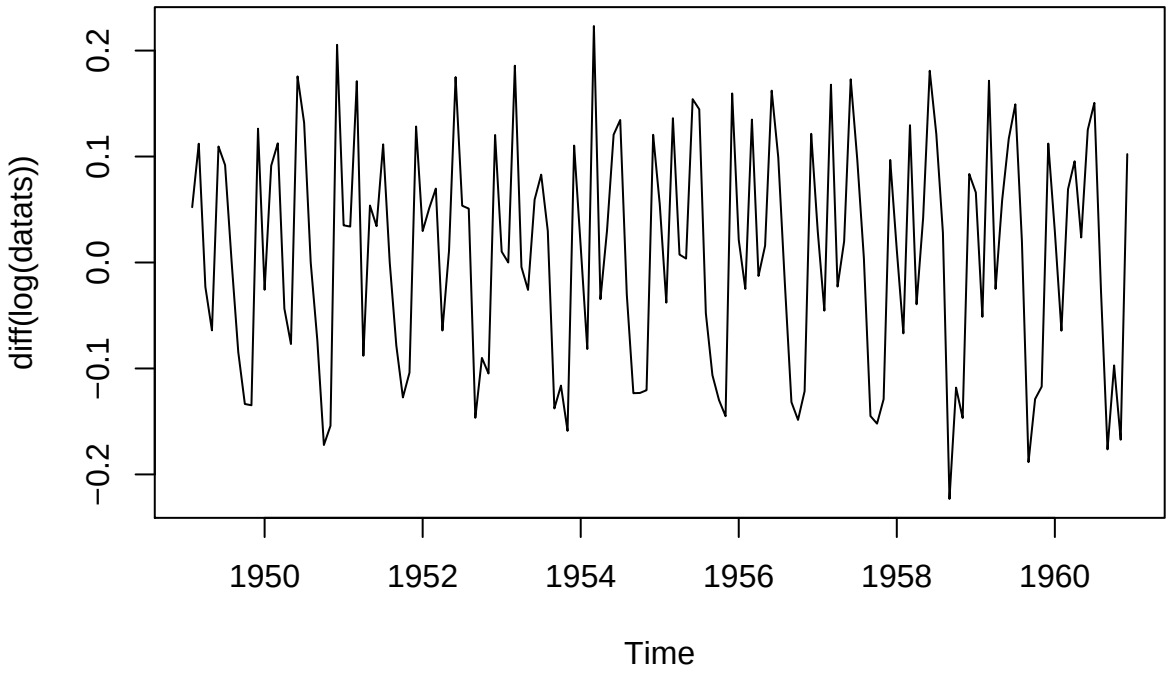
```r
acf (diff(log(datats)))# q value moving average
```



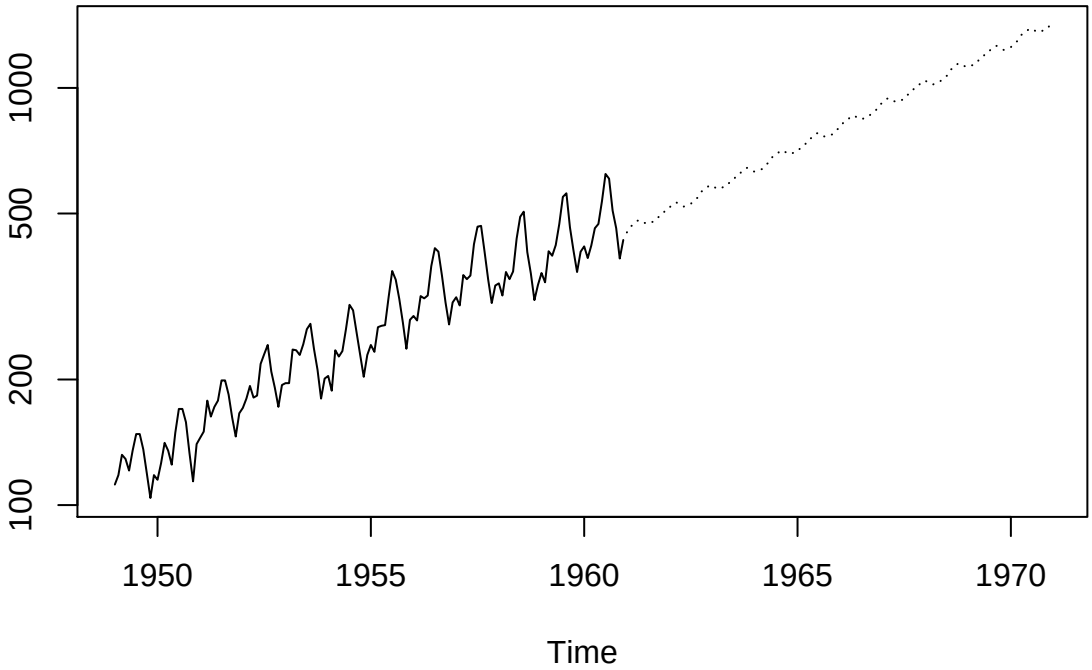

```r
pacf(diff(log(datats))) #p auto regression value
```



```
plot (diff(log(datats)))
```




```
#Building our model now,
fit<- arima(log(datats), c(0,1,1),seasonal = list(order=c(0,1,1), period = 10))
pred <- predict(fit, n.ahead = 10*12)

pred1 <- 2.718^pred$pred
ts.plot(datats, 2.718^pred$pred, log = "y", lty = c(1,3))
```




```
datawide = ts(datats, frequency = 12, start = c(1949,1), end = c(1959,12 ))
```

```r
#prediction
fit = arima(log(datawide), c(0,1,1), seasonal = list(order= c(0,1,1), period = 10))

pred <- predict(fit, n.ahead = 10*12)

pred1 <- 2.718^pred$pred

data1 <- head(pred1,12)

predicted1 = round(data1, digits = 1)
```