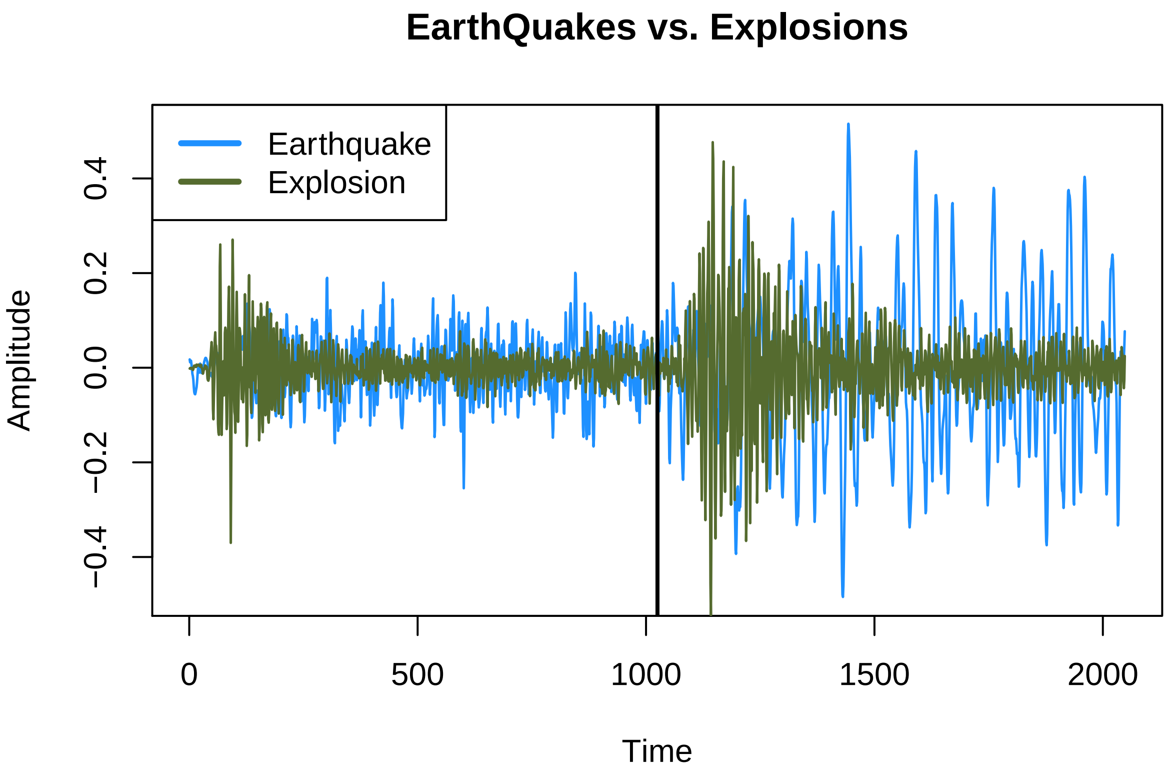
Assignment 01; STAT 626

Philip Anderson; panders2@tamu.edu

5/26/2018

Question 1.1

```
plot.ts(EQ5, col="dodgerblue"
        , lwd=1.25
        , xlab="Time"
        , ylab="Amplitude"
        , main="EarthQuakes vs. Explosions")
lines(EXP6, col="darkolivegreen", lwd=1.25)
legend("topleft"
       , c("Earthquake", "Explosion")
       , lty=c(1,1)
       , lwd=c(3,3)
       , col=c("dodgerblue", "darkolivegreen")
       )
abline(v=1025, lwd=2)
```



It appears from the plot that Earthquakes have a lower ratio of maximium amplitude in the first phase (where the black line breaks the graphic), to that of the second phase. We can check this below:

```
print("Earthquake ratio - max amplitude in phase 1 / max amplitude in phase 2")
```

```
## [1] "Earthquake ratio - max amplitude in phase 1 / max amplitude in phase 2"
```

```r
max(abs(EQ5[1:1024])) / max(abs(EQ5[1025:2048]))
```

```
## [1] 0.494083
```

```r
print("Explosion ratio - max amplitude in phase 1 / max amplitude in phase 2")
```

```
## [1] "Explosion ratio - max amplitude in phase 1 / max amplitude in phase 2"
```

```r
max(abs(EXP6[1:1024])) / max(abs(EXP6[1025:2048]))
```

```
## [1] 0.707169
```

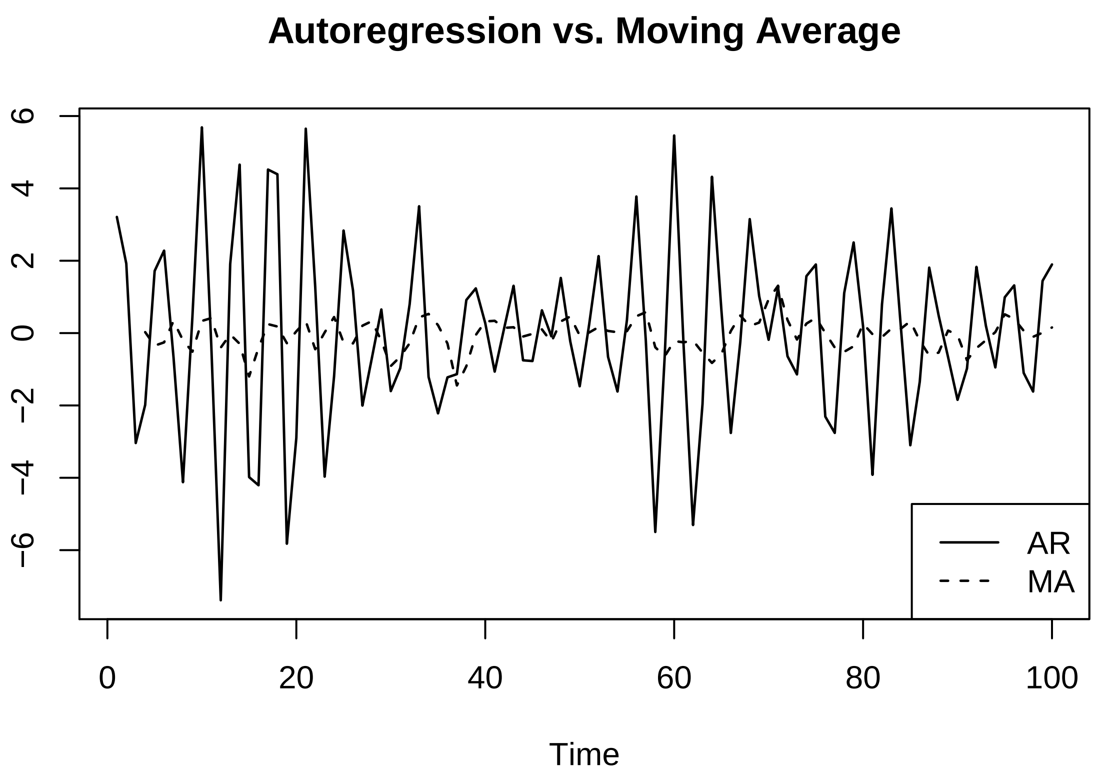
It also appears that in both phases, Earthquakes seem to be stationary (time-independent mean, time-independent variance), while Explosions do not - they have higher variance in the beginning of each phase, compared with the middle and end.

Question 1.3

Question 1.3A

```r
set.seed(1739)
# generate random draws with variance 1
w <- rnorm(125, 0, 1) # white noise
# apply the autoregressive filter and throw out first 25 records as burn-in
# note that we are looking for no lag 1 effect, just lag 2
x <- filter(w, filter=c(0, -0.9), method="recursive")[-(1:25)]
# now apply the moving average filter to the autoregression
v <- filter(x, sides=1, filter=rep(1/4, 4))

plot.ts(x, main="Autoregression vs. Moving Average", ylab="", lwd=1.25)
lines(v, lty=2, lwd=1.25)
legend("bottomright", c("AR", "MA")
       , lty=c(1,2)
       , lwd=c(1.25, 1.25)
       )
```



The moving average filter sizeably reduces the amplitude of the time series; this implies that much of that effect may be noise.

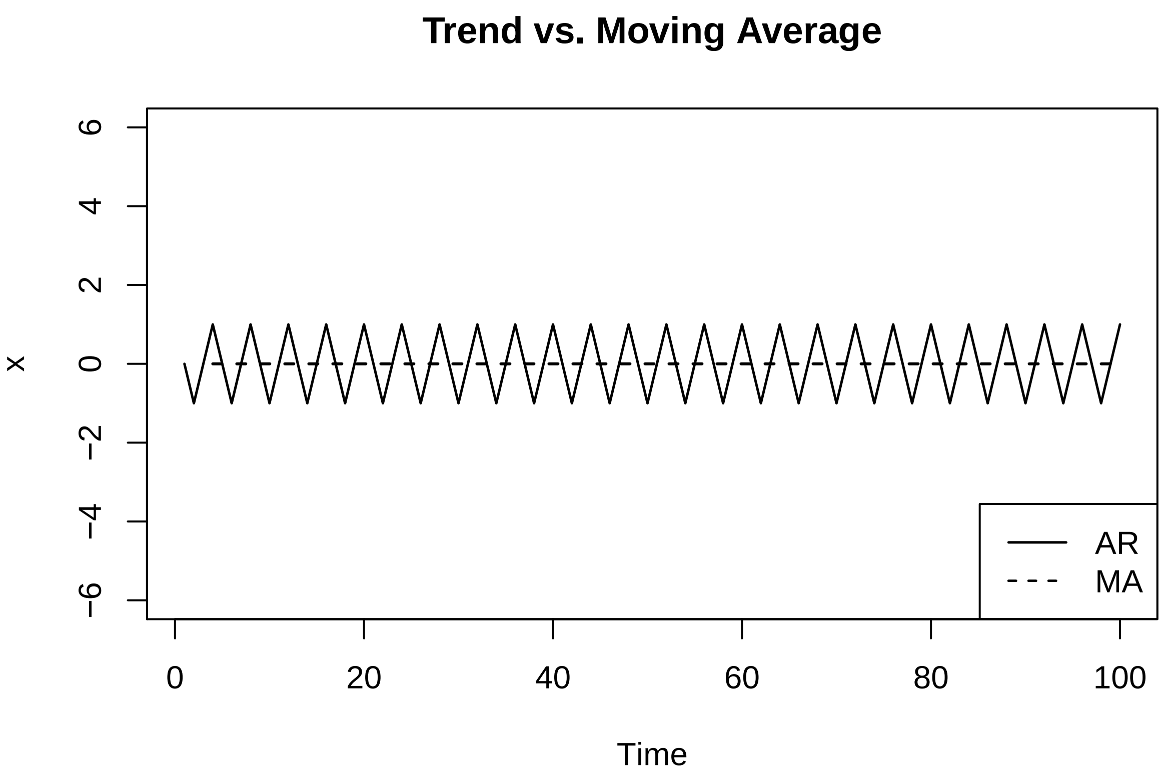
Question 1.3B

Generate a series but with no random component.

```
t <- seq(1, 100)
x <- cos(((2 * pi * t) / 4))
v <- filter(x, sides=1, filter=rep(1/4, 4))

plot.ts(x, lwd=1.25, main="Trend vs. Moving Average", ylim=c(-6, 6))
lines(v, lty=2, lwd=1.5)

legend("bottomright", c("AR", "MA")
       , lty=c(1,2)
       , lwd=c(1.25, 1.25)
       )
```



There is no noise to remove in the specified time series, so when we use the moving average filter with lag 4, a flat line is produced because the entire cycle of the trend is smoothed away.

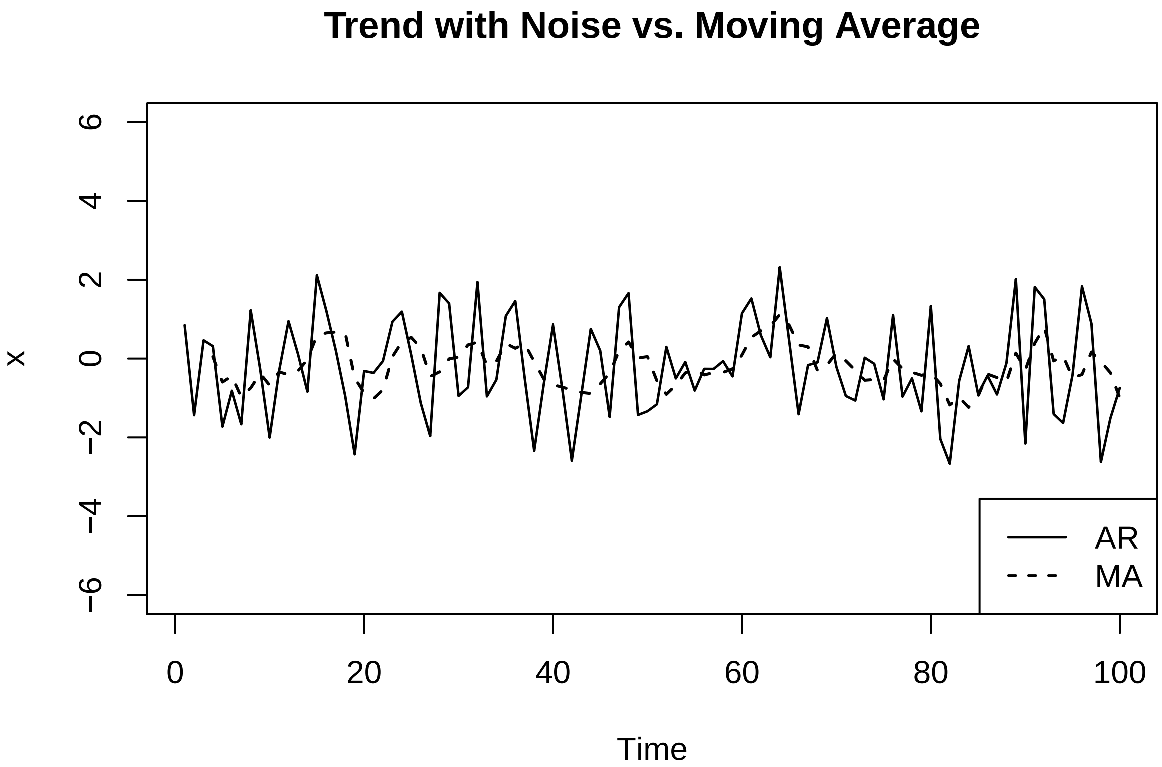
Question 3C

Repeat B but add white noise to the trend.

```
t <- seq(1, 100)
w <- rnorm(100, 0, 1)
x <- cos((2 * pi * t) / 4) + w

v <- filter(x, rep(1/4, 4), sides=1)
plot.ts(x, lwd=1.25, main="Trend with Noise vs. Moving Average", ylim=c(-6, 6))
lines(v, lty=2, lwd=1.5)

legend("bottomright", c("AR", "MA")
       , lty=c(1,2)
       , lwd=c(1.25, 1.25)
       )
```



In theory, the moving average filter should be removing the noise we added in, leaving us with the cosine wave we produced in the previous problem. It produces a line with a clear periodicity, but it is not as clean as the pure trend I produced.

Question 3D

In all cases, the moving average removes a great deal of variance from the time series. If there is not a random component to the time series, the moving average filter can remove all of the variance to the extent that it produces a flat line (3B). The filter is not perfect, however - comparing the results of 1.3.B to 1.3.C shows that the filter does not leave the pure trend by trying to remove the added noise.