Econ 217, HW 4

Swati Sharma

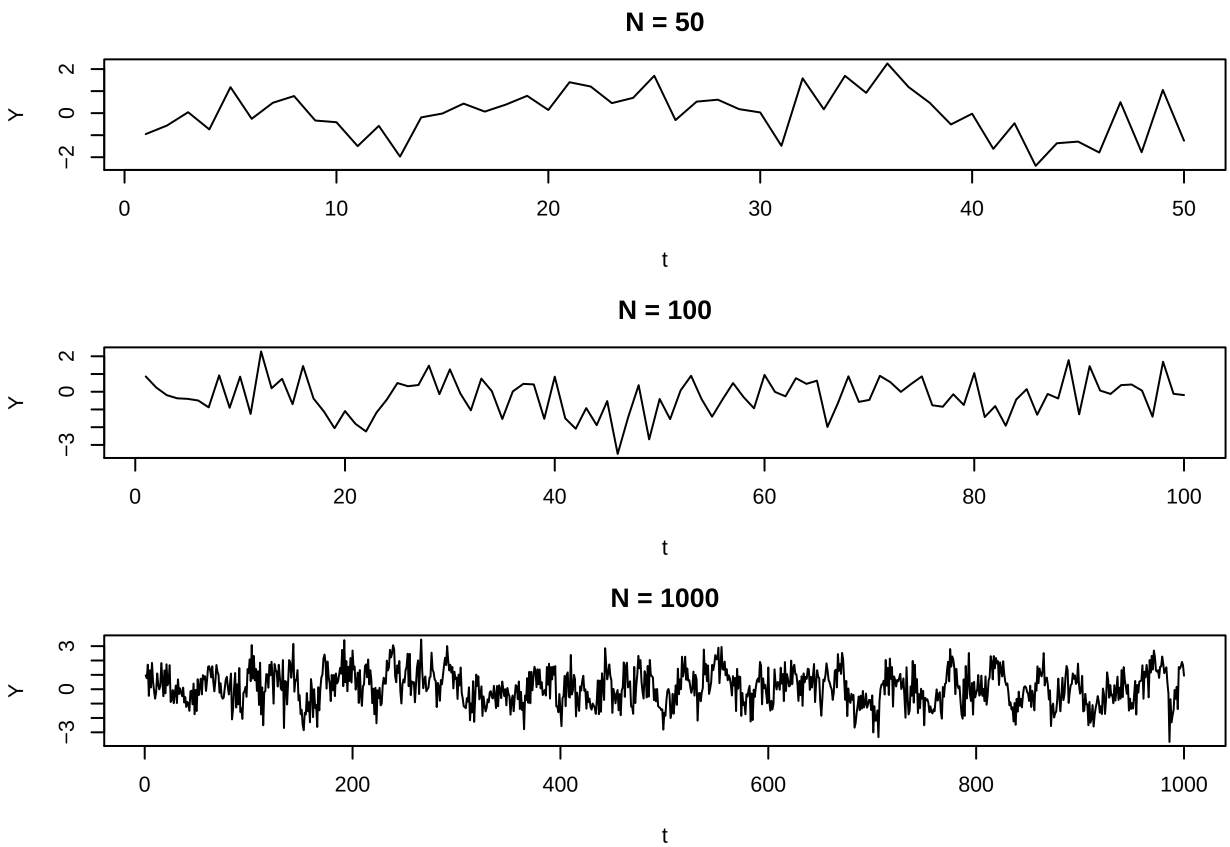
3/14/2018

Problem 1

Part a

```
ARMA<-function(n,phi,theta){
  p<-length(phi)
  q<-length(theta)
  nn<-max(p,q)
  es<-rnorm(n+nn)
  Y<-rep(0,n+nn)
  for(i in (nn+1):length(Y)){
    Y[i]<-t(phi)%*%Y[(i-p):(i-1)]+
          t(theta)%*%es[(i-q):(i-1)]+
          es[i]
  }
  Y<-Y[-(1:nn)]
  return(Y)
}

par(mfrow=c(3,1))
par(mar = c(4,4,3,1), oma = c(0,0,0,0))
plot(ARMA(50,c(.3,.1), c(0.2, 0.2, 0.1, 0.05)), type = "l",
     main = "N = 50", xlab = "t", ylab = "Y")
plot(ARMA(100,c(.3,.1), c(0.2, 0.2, 0.1, 0.05)), type = "l",
     main = "N = 100", xlab = "t", ylab = "Y")
plot(ARMA(1000,c(.3,.1), c(0.2, 0.2, 0.1, 0.05)), type = "l",
     main = "N = 1000", xlab = "t", ylab = "Y")
```



Part b

```
y1 <- ARMA(50,c(.3,.1), c(0.2, 0.2, 0.1, 0.05))
y2 <- ARMA(100,c(.3,.1), c(0.2, 0.2, 0.1, 0.05))
y3 <- ARMA(1000,c(.3,.1), c(0.2, 0.2, 0.1, 0.05))

cat("The mean of the sample for N = 50 is", mean(y1), "and the variance is", var(y1))
cat("The mean of the sample for N = 100 is", mean(y2), "and the variance is", var(y2))
cat("The mean of the sample for N = 1000 is", mean(y3), "and the variance is", var(y3))
```

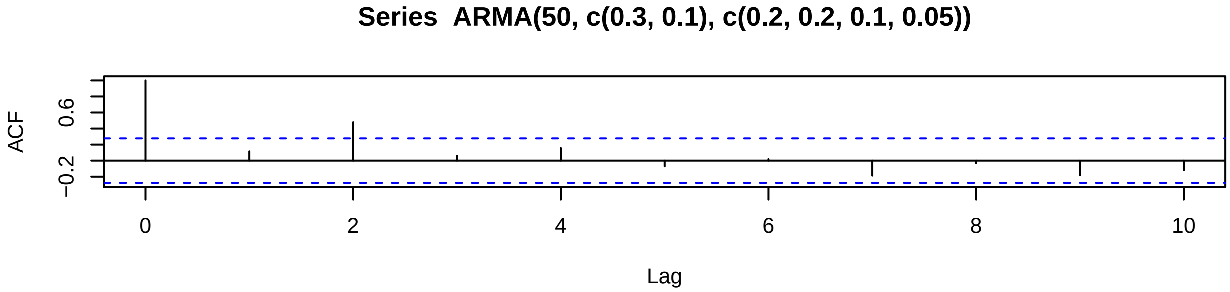
```
## The mean of the sample for N = 50 is -0.2829788 and the variance is 1.757641
## The mean of the sample for N = 100 is -0.04760877 and the variance is 1.117298
## The mean of the sample for N = 1000 is 0.03468305 and the variance is 1.496334
```

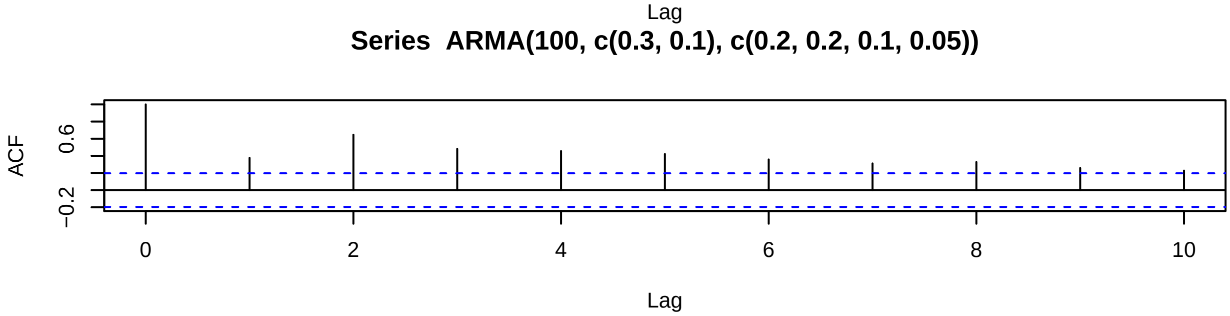
The variance of the sample should decrease with more samples while the mean should just converge to 0. However, because we are randomly adding noise (with the rnorm function) to the Y values, our mean and variance values do not always exhibit these tendencies.

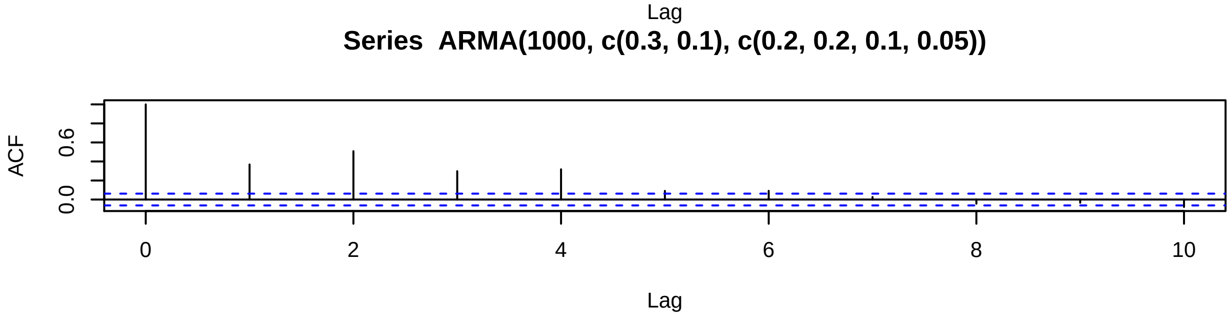
Part c

```
par(mfrow=c(3,1))
par(mar = c(4,4,3,1), oma = c(0,0,0,0))
acf(ARMA(50,c(.3,.1), c(0.2, 0.2, 0.1, 0.05)),lag.max=10,type="correlation")
```

```r
acf(ARMA(100,c(.3,.1), c(0.2, 0.2, 0.1, 0.05)),lag.max=10,type="correlation")
acf(ARMA(1000,c(.3,.1), c(0.2, 0.2, 0.1, 0.05)),lag.max=10,type="correlation")
```







Yes, we do find a difference by sample size. The larger the sample size, the less correlated the data is over time (seperated by lag length).

Part d

```r
confInt <- quantile(replicate(1000, var(ARMA(100,c(.3,.1), c(0.2, 0.2, 0.1, 0.05)))),
                    probs=c(0.05, 0.95))
print(confInt)
```

```
##        5%       95%
## 0.9684517 1.8778161
```

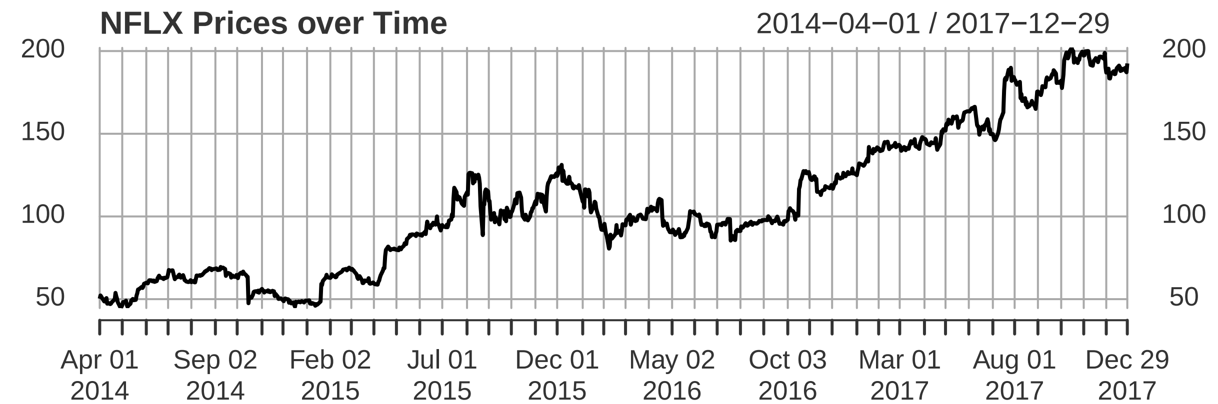
Problem 2

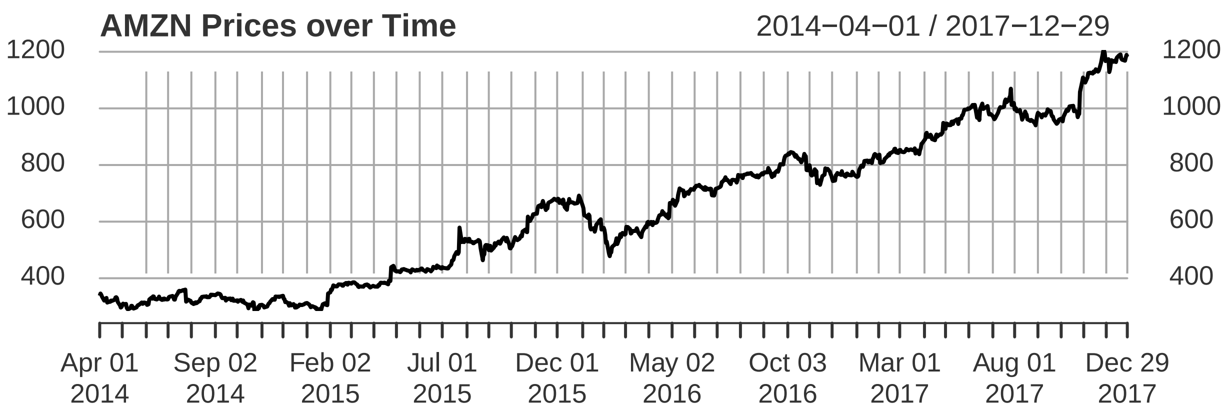
Part a

```r
library(quantmod)
getSymbols(c("NFLX", "AMZN"),from="2014-04-01",to="2018-01-01")
```

```
## [1] "NFLX" "AMZN"
```

```r
pricesN <- NFLX$NFLX.Open
pricesA <- AMZN$AMZN.Open
#rets <- dailyReturn(GOOG)
par(mfrow=c(2,1))
par(mar = c(4,4,2,1), oma = c(0,0,0,0))
plot(pricesN, type = "l", main = "NFLX Prices over Time",
     xlab = "t", ylab = "Price")
plot(pricesA, type = "l", main = "AMZN Prices over Time",
     xlab = "t", ylab = "Price")
```





Derivation of Reduced Form Autoregression:

$$N_t = \beta_0 + \beta_1 A_t + \beta_2 N_{t-1} + \beta_3 A_{t-1} + \beta_4 N_{t-2} + \beta_5 A_{t-2} + u_t$$
$$A_t = \gamma_0 + \gamma_1 N_t + \gamma_2 A_{t-1} + \gamma_3 N_{t-1} + \gamma_4 N_{t-2} + \gamma_5 A_{t-2} + e_t$$

$$N_t - \beta_1 A_t = \beta_0 + \beta_2 N_{t-1} + \beta_3 A_{t-1} + \beta_4 N_{t-2} + \beta_5 A_{t-2} + u_t$$
$$A_t - \gamma_1 N_t = \gamma_0 + \gamma_2 A_{t-1} + \gamma_3 N_{t-1} + \gamma_4 N_{t-2} + \gamma_5 A_{t-2} + e_t$$

$$N_t - \beta_1 A_t = \beta_0 + \beta_2 N_{t-1} + \beta_3 A_{t-1} + \beta_4 N_{t-2} + \beta_5 A_{t-2} + u_t$$
$$-\gamma_1 N_t + A_t = \gamma_0 + \gamma_3 N_{t-1} + \gamma_2 A_{t-1} + \gamma_4 N_{t-2} + \gamma_5 A_{t-2} + e_t$$

In matrix/vector notation the system is

$$\begin{pmatrix} 1 & -\beta_1 \\ -\gamma_1 & 1 \end{pmatrix}\begin{pmatrix} N_t \\ A_t \end{pmatrix} = \begin{pmatrix} \beta_0 \\ \gamma_0 \end{pmatrix} + \begin{pmatrix} \beta_2 & \beta_3 \\ \gamma_3 & \gamma_2 \end{pmatrix}\begin{pmatrix} N_{t-1} \\ A_{t-1} \end{pmatrix} + \begin{pmatrix} \beta_4 & \beta_5 \\ \gamma_4 & \gamma_5 \end{pmatrix}\begin{pmatrix} N_{t-2} \\ A_{t-2} \end{pmatrix} + \begin{pmatrix} u_t \\ e_t \end{pmatrix}$$

Inverting,

$$\begin{pmatrix} N_t \\ A_t \end{pmatrix} = \begin{pmatrix} 1 & -\beta_1 \\ -\gamma_1 & 1 \end{pmatrix}^{-1}\begin{pmatrix} \beta_0 \\ \gamma_0 \end{pmatrix} + \begin{pmatrix} 1 & -\beta_1 \\ -\gamma_1 & 1 \end{pmatrix}^{-1}\begin{pmatrix} \beta_2 & \beta_3 \\ \gamma_3 & \gamma_2 \end{pmatrix}\begin{pmatrix} N_{t-1} \\ A_{t-1} \end{pmatrix} + \begin{pmatrix} 1 & -\beta_1 \\ -\gamma_1 & 1 \end{pmatrix}^{-1}\begin{pmatrix} \beta_4 & \beta_5 \\ \gamma_4 & \gamma_5 \end{pmatrix}\begin{pmatrix} N_{t-2} \\ A_{t-2} \end{pmatrix} + \begin{pmatrix} 1 & -\beta_1 \\ -\gamma_1 & 1 \end{pmatrix}^{-1}\begin{pmatrix} u_t \\ e_t \end{pmatrix}$$

Given $z_t = \begin{pmatrix} N_t \\ A_t \end{pmatrix}$, $\boldsymbol{\beta_0} = \begin{pmatrix} 1 & -\beta_1 \\ -\gamma_1 & 1 \end{pmatrix}^{-1}\begin{pmatrix} \beta_0 \\ \gamma_0 \end{pmatrix}$, $\boldsymbol{\beta_1} = \begin{pmatrix} 1 & -\beta_1 \\ -\gamma_1 & 1 \end{pmatrix}^{-1}\begin{pmatrix} \beta_2 & \beta_3 \\ \gamma_3 & \gamma_2 \end{pmatrix}$, $\boldsymbol{\beta_2} = \begin{pmatrix} 1 & -\beta_1 \\ -\gamma_1 & 1 \end{pmatrix}^{-1}\begin{pmatrix} \beta_4 & \beta_5 \\ \gamma_4 & \gamma_5 \end{pmatrix}$, and $\tilde{u}_t = \begin{pmatrix} 1 & -\beta_1 \\ -\gamma_1 & 1 \end{pmatrix}^{-1}\begin{pmatrix} u_t \\ e_t \end{pmatrix}$,
the reduced form autoregression is $z_t = \boldsymbol{\beta_0} + \boldsymbol{\beta_1} z_{t-1} + \boldsymbol{\beta_2} z_{t-2} + \tilde{u}_t$

Estimation of the two-lag reduced form vector autoregression:

```r
rfvarN <- lm(pricesN~lag(pricesN,k=1)+lag(pricesA,k=1)+lag(pricesN,k=2)+lag(pricesA,k=2))
rfvarA<- lm(pricesA~lag(pricesA,k=1)+lag(pricesN,k=1)+lag(pricesN,k=2)+lag(pricesA,k=2))
summary(rfvarN)
```

```
##
## Call:
## lm(formula = pricesN ~ lag(pricesN, k = 1) + lag(pricesA, k = 1) +
##     lag(pricesN, k = 2) + lag(pricesA, k = 2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17.5912  -1.1118  -0.0853   1.1981  18.3726
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.183009   0.246145   0.744  0.45736
## lag(pricesN, k = 1)  1.016253   0.035927  28.286  < 2e-16 ***
## lag(pricesA, k = 1) -0.021695   0.008908  -2.435  0.01506 *
## lag(pricesN, k = 2) -0.026335   0.035950  -0.733  0.46402
## lag(pricesA, k = 2)  0.023368   0.008904   2.625  0.00882 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.747 on 939 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.9958, Adjusted R-squared:  0.9957
## F-statistic: 5.518e+04 on 4 and 939 DF,  p-value: < 2.2e-16
```

```r
summary(rfvarA)
```

```
##
## Call:
## lm(formula = pricesA ~ lag(pricesA, k = 1) + lag(pricesN, k = 1) +
##     lag(pricesN, k = 2) + lag(pricesA, k = 2))
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -57.930  -4.655   0.075   5.061  86.091
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.02755    0.99360   0.028    0.978
## lag(pricesA, k = 1)  0.94238    0.03596  26.207   <2e-16 ***
## lag(pricesN, k = 1)  0.07480    0.14502   0.516    0.606
## lag(pricesN, k = 2) -0.04363    0.14512  -0.301    0.764
## lag(pricesA, k = 2)  0.05378    0.03594   1.496    0.135
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.09 on 939 degrees of freedom
##   (2 observations deleted due to missingness)
## Multiple R-squared:  0.9981, Adjusted R-squared:  0.9981
## F-statistic: 1.26e+05 on 4 and 939 DF,  p-value: < 2.2e-16
```

The lag at t-1 of Netflix price has a highly statistically significant positive effect on its own price at t (very low p-value of 2e-16). The lags of Amazon's price at t-1 and t-2 also both have an effect on Netflix's price at t - these effects are still statistically significant but less so than the previous effect. At t-1 Amazon's price has a negative effect on Netflix's price at t with a p-value of 0.01506. At t-2, Amazon's price has a positive effect on Netflix's price at t with a p-value of 0.00882.

The only variable that has a statistically significant effect on Amazon's prices is itself at t-1. This is highly significant with a p-value of 2e-16. In both cases, it seems that the price of the companies at t goes up from their prices at t-1.

Part b

```r
library(MSBVAR)
y <- ts(data.frame(pricesN, pricesA))
granger.test(y,p=2)
```

```
##                          F-statistic     p-value
## AMZN.Open -> NFLX.Open   4.7024837 0.009287462
## NFLX.Open -> AMZN.Open   0.9316706 0.394258926
```

The first row in which we assess whether Amazon prices Granger cause Netflix prices shows a p-value of 0.00928, so we can reject the null of no Granger causality. In other words, the history of Amazon prices helps predict Netflix prices. The same is not true in reverse; with a very high p-value of 0.39425, we fail to reject the null that Netflix prices do not Granger cause Amazon prices. The history of Netflix prices does not help predict Amazon prices.

Part c

```r
rfVar <- reduced.form.var(y, p = 2)

resid <- rfVar[["residuals"]]
resid[944, 2] <- resid[944,2] + 60 # shock by 60 dollars in e at last date
resid <- matrix(resid[944,], 2, 1)
```

```r
B_0 <- matrix(rfVar[["intercept"]],2,1)
B_1 <- matrix(rfVar[["ar.coefs"]][, , 1], 2, 2)
B_2 <- matrix(rfVar[["ar.coefs"]][, , 2], 2, 2)

forecast <- matrix(NA, 13, 3)
for(i in 1:13){forecast[i,1] = i}

forecast[1,2:3] <- y[945,] # prices at time t-2
forecast[2,2:3] <- y[946,] # prices at time t-1

# prices at t(during shock)
forecast[3,2:3] <- B_0 + B_1%*%forecast[1,2:3] + B_2%*%forecast[2,2:3] + resid

for(h in 4:13){
  lag <- forecast[h-1,2:3]
  lag2 <- forecast[h-2,2:3]
  forecast[h,2:3] <- B_0 + B_1%*%lag + B_2%*%lag2
}

matplot(forecast[,1], forecast[,2:3], type = "l", lty = 1, col = c("black", "darkgreen"),
        main = "Price Forecast of NFLX and AMZN after Shock", xlab = "Time Periods",
        ylab = "Prices")
abline(v=3, col = "firebrick4")
legend(9.5,1000,c("NFLX Prices", "AMZN Prices", "Shock"),
       col = c("black", "darkgreen", "firebrick4"), lty = 1)
```