

## HW 2

MFE 407: Empirical Methods in Finance

Professor Lochstoer

Group 6

Students: Xiahao Wang, Juan Manuel Ferreyra Maspero, Xinyue Zhu,  
Yichu Li, Mu Lin

### Problem 1 AR(p) Process:

1.(a)

```
phi0 <- 0
phi1 <- 1.1
phi2 <- -0.25

size <- 10000
w <- rep(0, size)

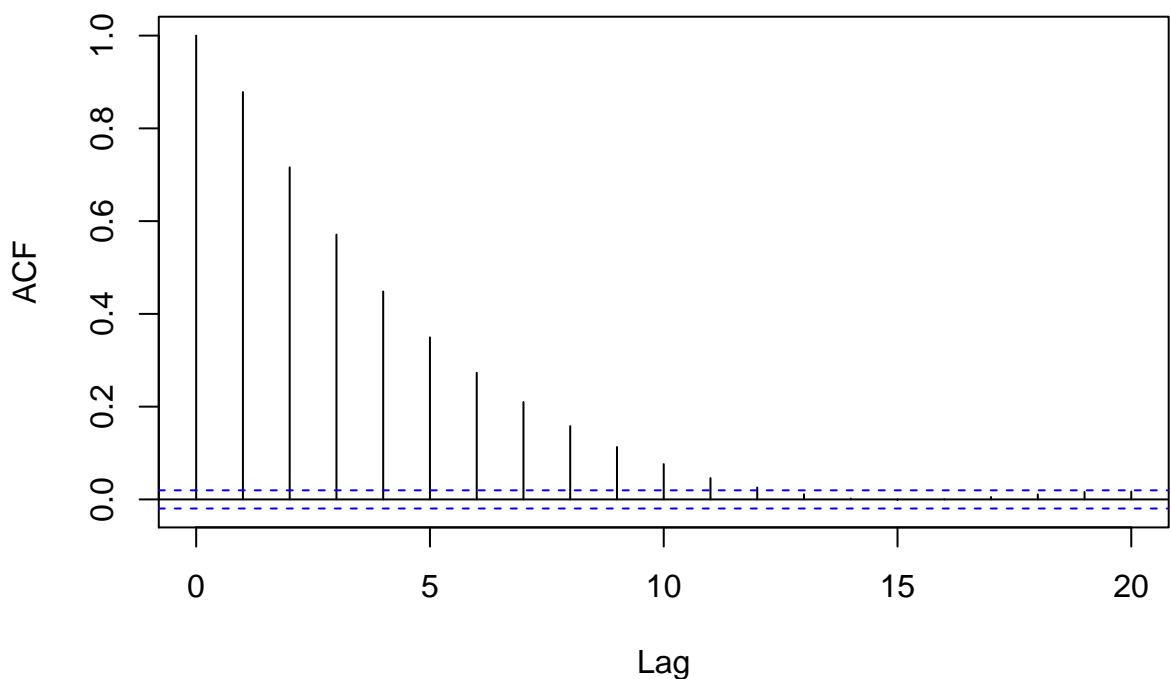
w[1] <- 0.5
w[2] <- 0.1

epsilon <- rnorm(size, 0,1)

for(i in 3:size) {
  w[i] = phi0 + phi1 * w[i-1] + phi2 * w[i-2] + epsilon[i]
}

acf(w, lag.max = 20)
```

**Series w**



1.(b)

```
x1 <- (phi1 + sqrt(phi1^2 + 4 * phi2))/(-2 * phi2)
x2 <- (phi1 - sqrt(phi1^2 + 4 * phi2))/(-2 * phi2)

w1 <- 1/x1
w2 <- 1/x2
w1
```

```
## [1] 0.3208712
```

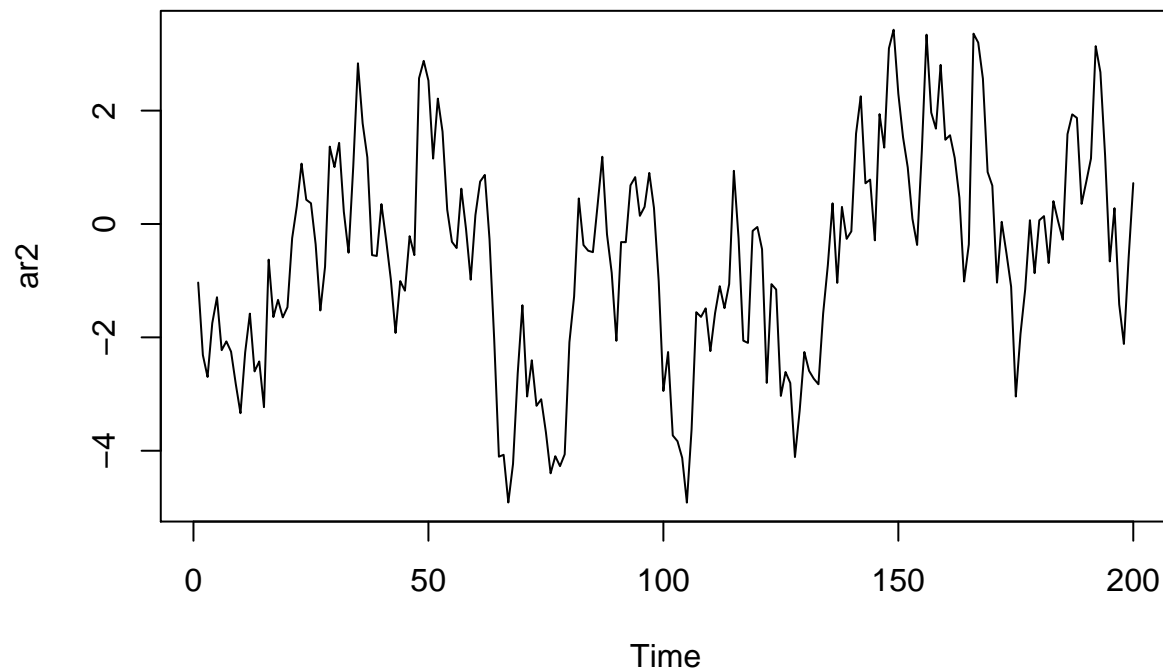
```
w2
```

```
## [1] 0.7791288
```

The two roots are real, so the process is stationary. Also based on the plot, the series converges

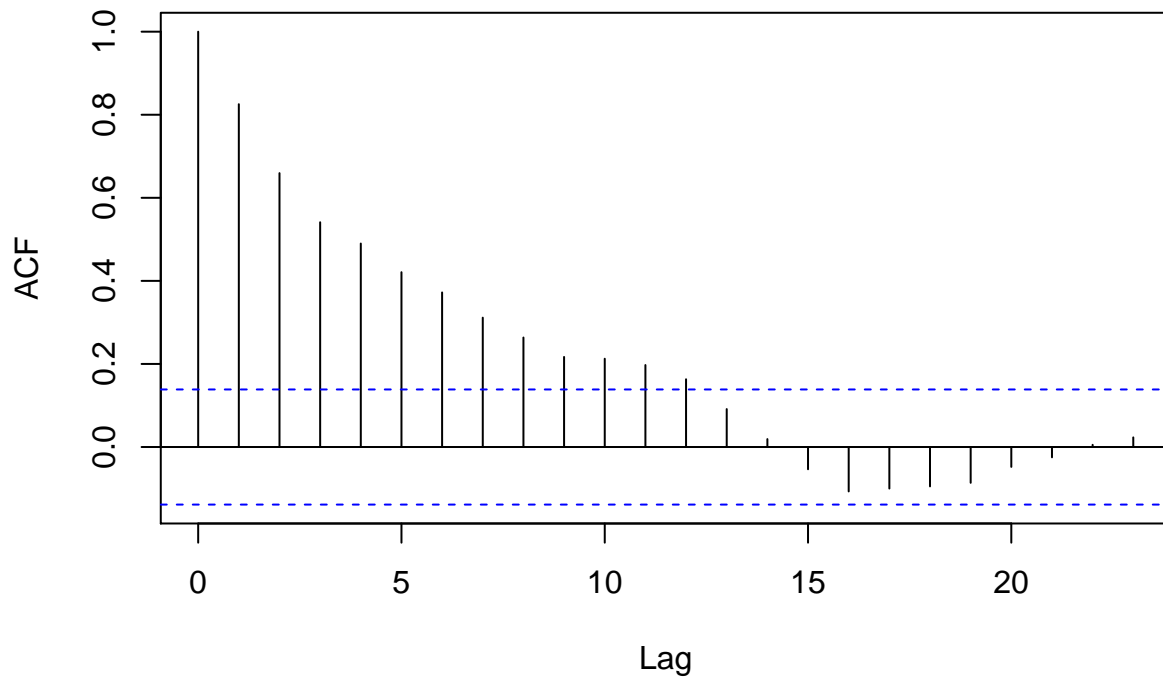
Also plot the graph and acf:

```
ar2 <- arima.sim(model = list(order = c(2,0,0), ar=c(1.1,-0.25)), n = 200)
plot(ar2)
```



```
acf(ar2)
```

## Series ar2



shows that it is stationary.

(c) After expanding the series:  $\frac{\partial X_t}{\partial \varepsilon} = \phi_1^6 + 5\phi_1^4\phi_2 + 6\phi_1^2\phi_2^2 + \phi_2^3$

```
multiplier1 <- phi1^6 + 5 * phi1^4 * phi2 + 6 * phi1^2 * phi2^2 + phi2^3
cat("the multiplier is", multiplier1)
```

```
## the multiplier is 0.379561
```

(d)

```
phi1 <- 0.9
phi2 <- 0.8
x1 <- (phi1 + sqrt(phi1^2 + 4 * phi2)) / (-2 * phi2)
x2 <- (phi1 - sqrt(phi1^2 + 4 * phi2)) / (-2 * phi2)
multiplier2 <- phi1^6 + 5 * phi1^4 * phi2 + 6 * phi1^2 * phi2^2 + phi2^3
cat("The multiplier is", multiplier2)
```

```
## The multiplier is 6.778241
```

```
cat("The two roots are", 1/x1, 1/x2)
```

```
## The two roots are -0.5512492 1.451249
```

One of the roots are bigger than 1, hence the process is not stationary.

## Problem 2

1.

```
library(lubridate)
```

```
##
```

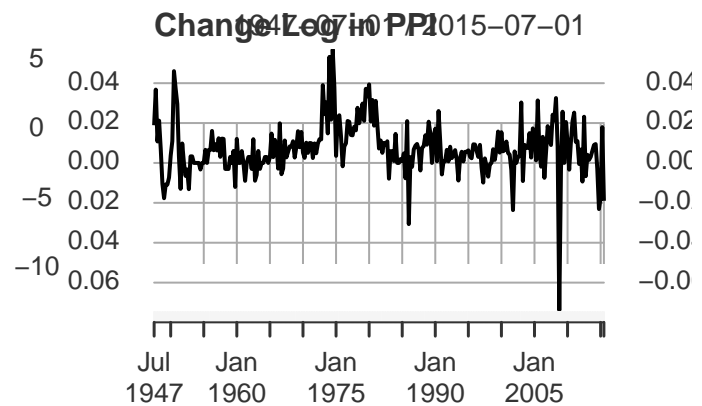
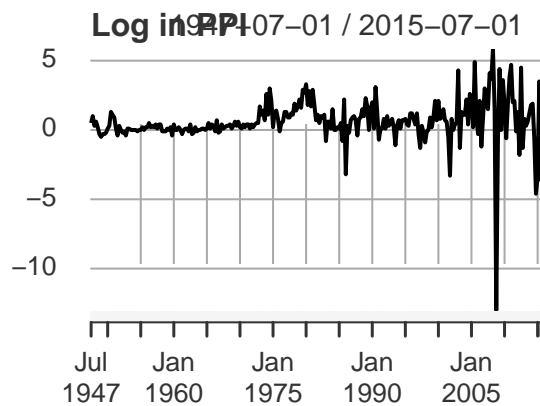
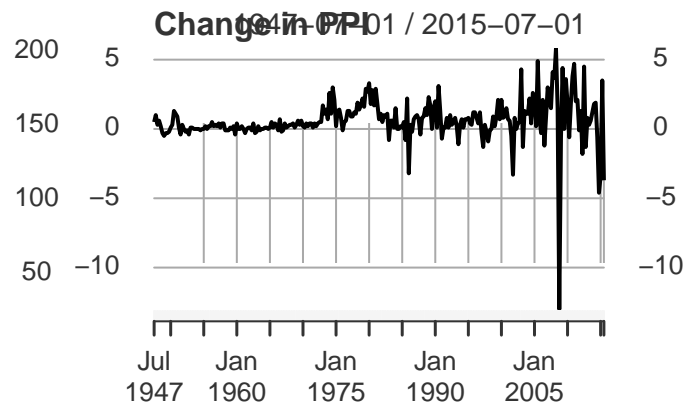
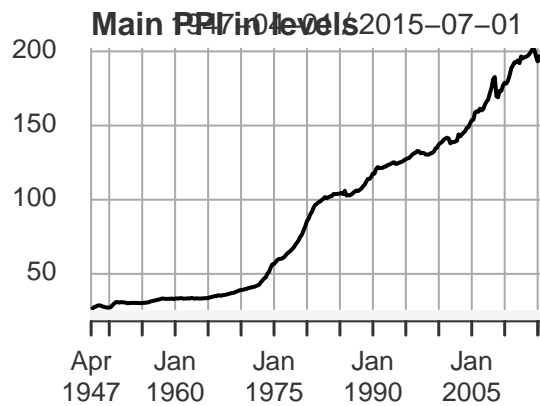
```

## Attaching package: 'lubridate'
## The following object is masked from 'package:base':
##
##      date
library(xts)

## Loading required package: zoo
##
## Attaching package: 'zoo'
## The following objects are masked from 'package:base':
##
##      as.Date, as.Date.numeric
library(forecast)

## Warning: package 'forecast' was built under R version 3.5.2
par(mfrow=c(2,2))
ppi_raw <- read.csv("PPIFGS.csv")
ppi_xts <- xts(x=as.double(ppi_raw$VALUE), ymd(ppi_raw$DATE))
#2.a
plot(ppi_xts, main = "Main PPI in levels")
#2.b
ppi_diff <- diff(ppi_xts)[-1]
plot(ppi_diff, main = "Change in PPI")
#2.c
ppi_log <- log(ppi_xts)
plot(ppi_diff, main = "Log in PPI")
#2.d
ppi_diff_log <- diff(ppi_log)[-1]
plot(ppi_diff_log, main = "Change Log in PPI")

```

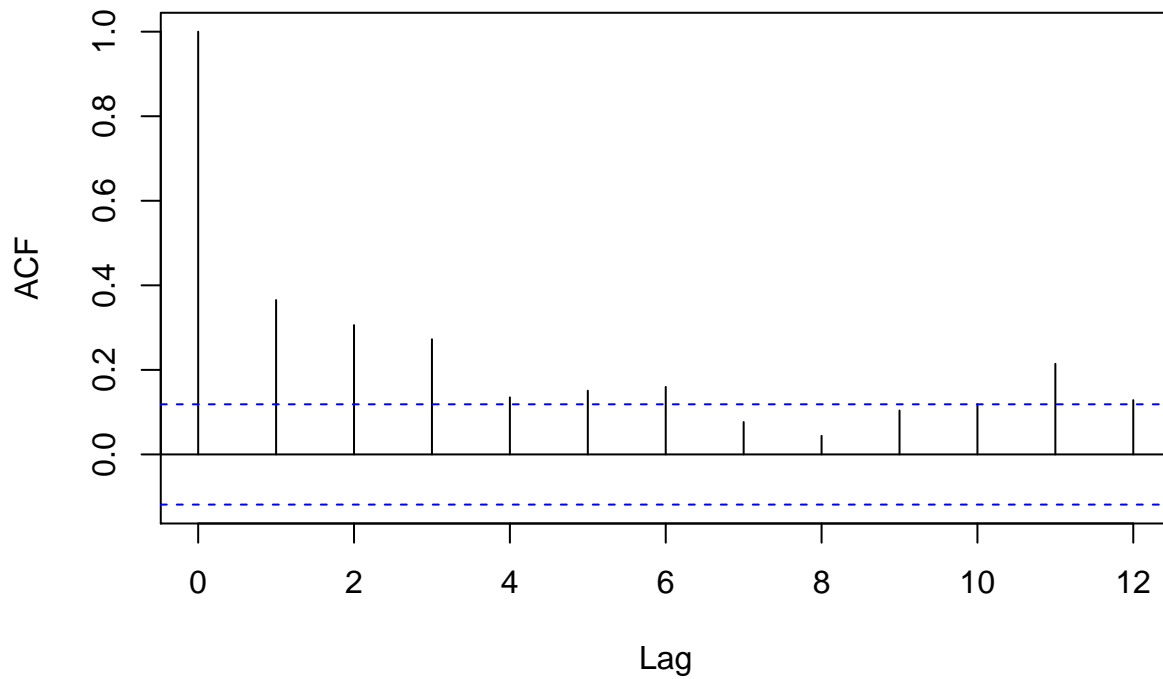


2. Change Log in PPI 2.(d) is covariance stationary. 2(a) has no constant mean 2(b), 2(c) don't have constant variance.

3.

```
acf(ppi_diff_log, lag.max = 12)
```

### Series ppi\_diff\_log

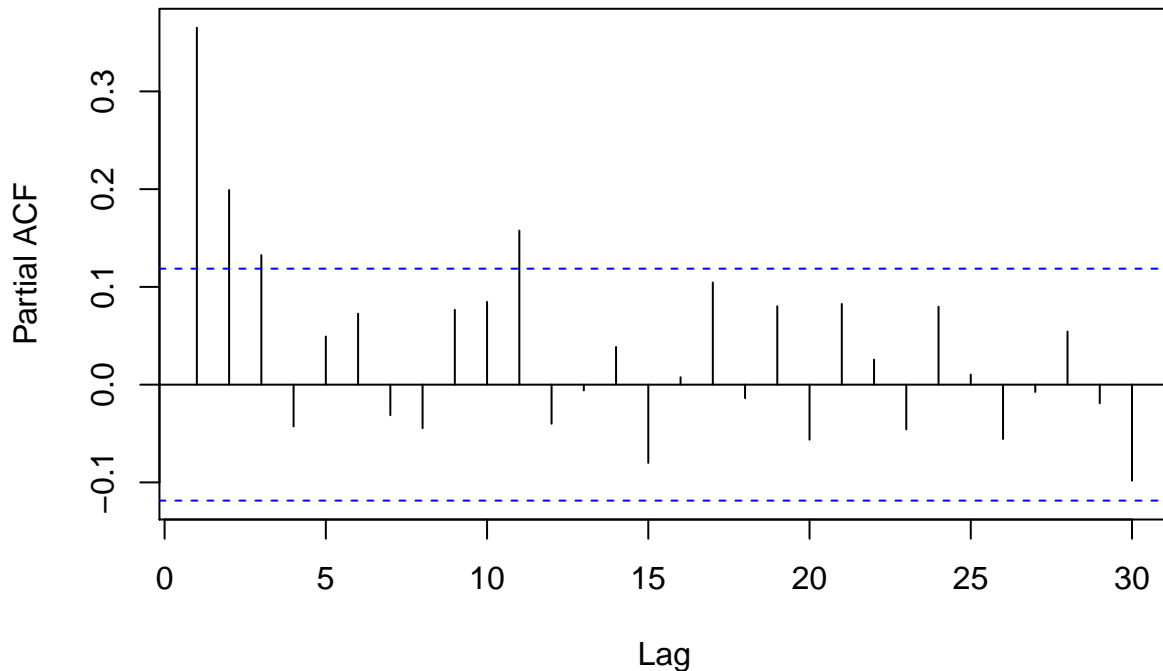


The ACF of changes of log(PPI) converges as lags increase and starts to diminish below one standard deviation (s.d) when lag is greater than 4. There might exist some seasonality in prior periods as we observe ACF greater than 1 s.d for lag = 11.

4.

```
pacf(ppi_diff_log, lag.max = 30)
```

## Series ppi\_diff\_log



We tried to plot PACF with 30 lags, and identify that there is no seasonality in the data. On the PACF plot, it is still significant after 3 lags and also significant on 11th lag. We would like to try different models (AR(3), AR(1:3,11)) to see which one fits better.

5. Choose AR(3) and AR with lag 1,2,3 and 11

```
ar3 <- arima(ppi_diff_log, order=c(3,0,0))
ar12311<- arima(ppi_diff_log, order=c(11,0,0),
                fixed=c(NA,NA,NA,0,0,0,0,0,0,0,NA,NA),
                transform.pars = FALSE)
cat("Coefficient for Ar3 is:\n")
```

```
## Coefficient for Ar3 is:
```

```
ar3$coef
```

```
##          ar1          ar2          ar3  intercept
## 0.268731607 0.160715955 0.140186620 0.007322805
```

```
cat("#####\n")
```

```
## #####
```

```
cat("Coefficient for AR with lag on (1,2,3,11) is:\n")
```

```
## Coefficient for AR with lag on (1,2,3,11) is:
```

```
ar12311$coef
```

```
##          ar1          ar2          ar3          ar4          ar5          ar6
## 0.252303148 0.146834382 0.139532331 0.000000000 0.000000000 0.000000000
##          ar7          ar8          ar9          ar10          ar11  intercept
## 0.000000000 0.000000000 0.000000000 0.000000000 0.174621530 0.007050635
```

```

cat("#####\n")

## #####

cat("Standard error for Ar3 is:\n")

## Standard error for Ar3 is:
sqrt(ar3$sigma2)

## [1] 0.0117843

cat("#####\n")

## #####

cat("Standard error for AR with lag on (1,2,3,11) is:\n")

## Standard error for AR with lag on (1,2,3,11) is:
sqrt(ar12311$sigma2)

## [1] 0.01157444

ar3roots <-polyroot(c(1,-ar3$coef[1:3]))
ar12311roots <-polyroot(c(1,-ar12311$coef[1:11]))
cat("#####\n")

## #####

cat("Mod for Ar3 is:\n")

## Mod for Ar3 is:
Mod(1/ar3roots)

## [1] 0.7409665 0.4349647 0.4349647

cat("Mod for AR with lag on (1,2,3,11) is:\n")

## Mod for AR with lag on (1,2,3,11) is:
Mod(1/ar12311roots)

## [1] 0.8336369 0.8502784 0.8502784 0.8688015 0.8428316 0.8428316 0.9304305
## [8] 0.8688015 0.8346560 0.8336369 0.8346560

```

As the mod of the characteristic roots are smaller than 1, so we conclude that these models are stationary

5.b Plot both AR plots and ACF to show that residual is white noise and has no autocorrelation

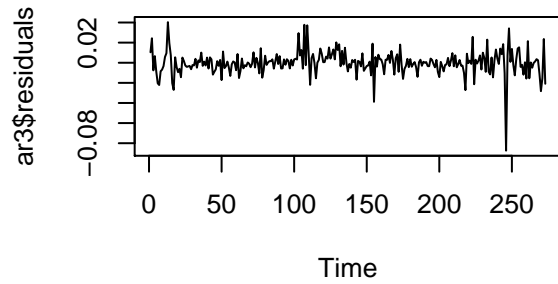
```

par(mfrow=c(2,2))
plot(ar3$residuals, main="AR(3) Residual plot")
acf(ar3$residuals, main="ACF AR(3) Residual plot")
plot(ar12311$residuals, main="AR(1:3,11) Residual plot")
acf(ar12311$residuals, main="ACF AR(1:3,11) Residual plot")

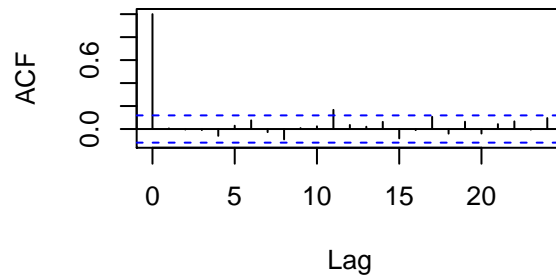
```



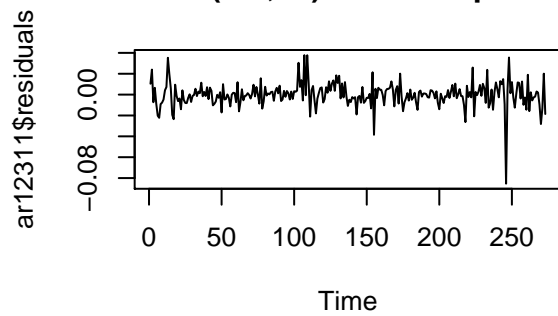
**AR(3) Residual plot**



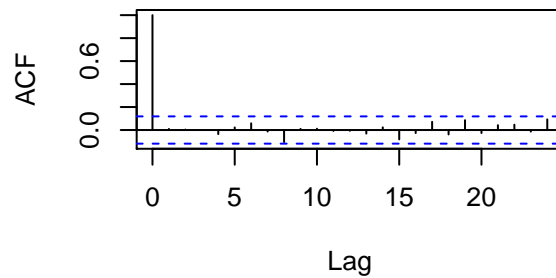
**ACF AR(3) Residual plot**



**AR(1:3,11) Residual plot**



**ACF AR(1:3,11) Residual plot**



5.c

```
# Test Normality
Box.test(ar3$residuals, lag=8, type='Ljung')

##
## Box-Ljung test
##
## data: ar3$residuals
## X-squared = 5.3458, df = 8, p-value = 0.7201

Box.test(ar3$residuals, lag=12, type='Ljung')

##
## Box-Ljung test
##
## data: ar3$residuals
## X-squared = 13.829, df = 12, p-value = 0.3118

AIC(ar3)

## [1] -1639.746

BIC(ar3)

## [1] -1621.699

Box.test(ar12311$residuals, lag=8, type='Ljung')

##
## Box-Ljung test
##
## data: ar12311$residuals
```

```
## X-squared = 4.6285, df = 8, p-value = 0.7964
Box.test(ar12311$residuals, lag=12, type='Ljung')
```

```
##
## Box-Ljung test
##
## data: ar12311$residuals
## X-squared = 4.7273, df = 12, p-value = 0.9665
```

```
AIC(ar12311)
```

```
## [1] -1647.148
```

```
BIC(ar12311)
```

```
## [1] -1625.491
```

Choose AR with lag on 1,2,3,11 because of lower AIC & BIC and p-value greater than 5% significance level.

6. Fit AR(3) and AR model with lag 1,2,3 and 11

```
ppi_xts2005 <- ppi_xts["1947-04-01/2005-12-31"]
ppi_diff_log_2005 <- diff(log(ppi_xts2005))[-1]
num <- length(ppi_diff_log) - length(ppi_diff_log_2005)

traindata <- ppi_diff_log["2005-12-31/2015"]

# fit for the models
ar3_2005<- arima(ppi_diff_log_2005, order=c(3,0,0))
ar12311_2005<- arima(ppi_diff_log_2005, order=c(11,0,0),
                    fixed=c(NA,NA,NA,0,0,0,0,0,0,0,NA,NA),transform.pars = FALSE)
```

```
fitar3 <- forecast(ar3_2005, h=39)
```

```
fitar12311 <-forecast(ar12311_2005, h=num)
```

```
e_ar3 <- sum((as.double(traindata) - fitar3$mean)^2) /num
e_ar12311 <- sum((as.double(traindata) - fitar12311$mean)^2) /num
cat("#####\n")
```

```
## #####
```

```
cat("MSPE AR(3) is:",e_ar3,"\n")
```

```
## MSPE AR(3) is: 0.000338503
```

```
cat("MSPE AR(1,2,3,11) is:",e_ar12311,"\n")
```

```
## MSPE AR(1,2,3,11) is: 0.0003393177
```

Simulate Random Walk (39 steps)

```
e_rw <- rep(0,10000)
```

```
for(j in 1: 10000){
  result <- rep(0,num+1)
  result[1] <- last(ppi_xts2005)
  for(i in 2:40){
    result[i] = result[i-1] + rnorm(1,0,1)
  }
}
```

```
rw <- diff(log(result))
e_rw[j] <- sum((as.double(traindata) - rw)^2) / num
}

cat("Average mean of MSPE of 10000 random walk is:", mean(e_rw), "\n")
```

```
## Average mean of MSPE of 10000 random walk is: 0.000386051
```

Hence our model is definitely better than random walk.