Homework 3 Solutions

Ian Cranfield SID [redacted]

Due 5/5/2020

```
library(dynlm)
library(pastecs)
library(openxlsx)
library(tidyverse)
library(data.table)
library(scales)
library(TSstudio)
library("readxl")
library(ggplot2)
require(cowplot)
library(psych)
library(corrplot)
library(gtable)
library(timeDate)
library(PerformanceAnalytics)
library(fpp2)

library(seasonal)
library(lubridate)
library(dplyr)
library(forecast)
library(timeSeries)
library(tseries)
library(strucchange)
```

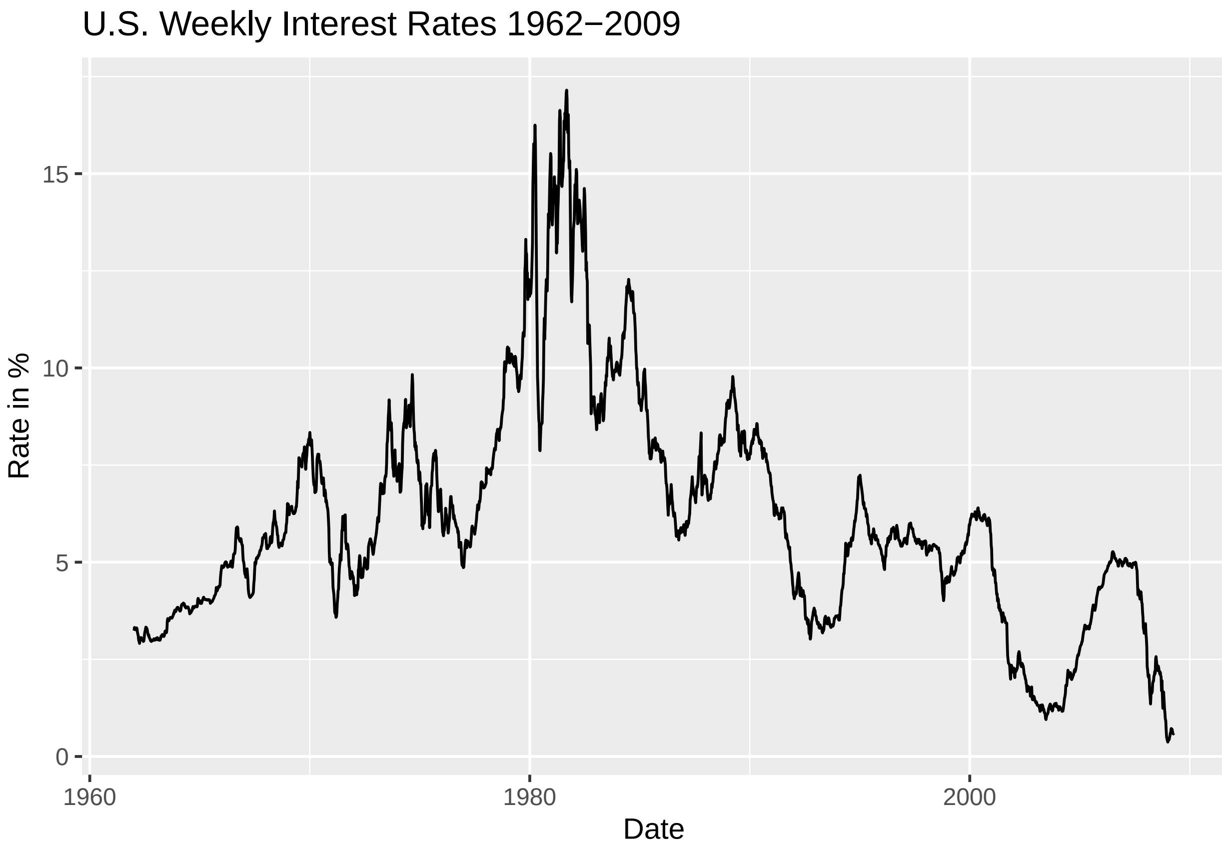
---

1. The file w-gs1yr.txt contains the U.S. weekly interest rates (in percentages) from January 5, 1962, to April 10, 2009. For this assignment you will fit an appropriate ARMA(p,q) model and make a 24-steps-ahead forecast.

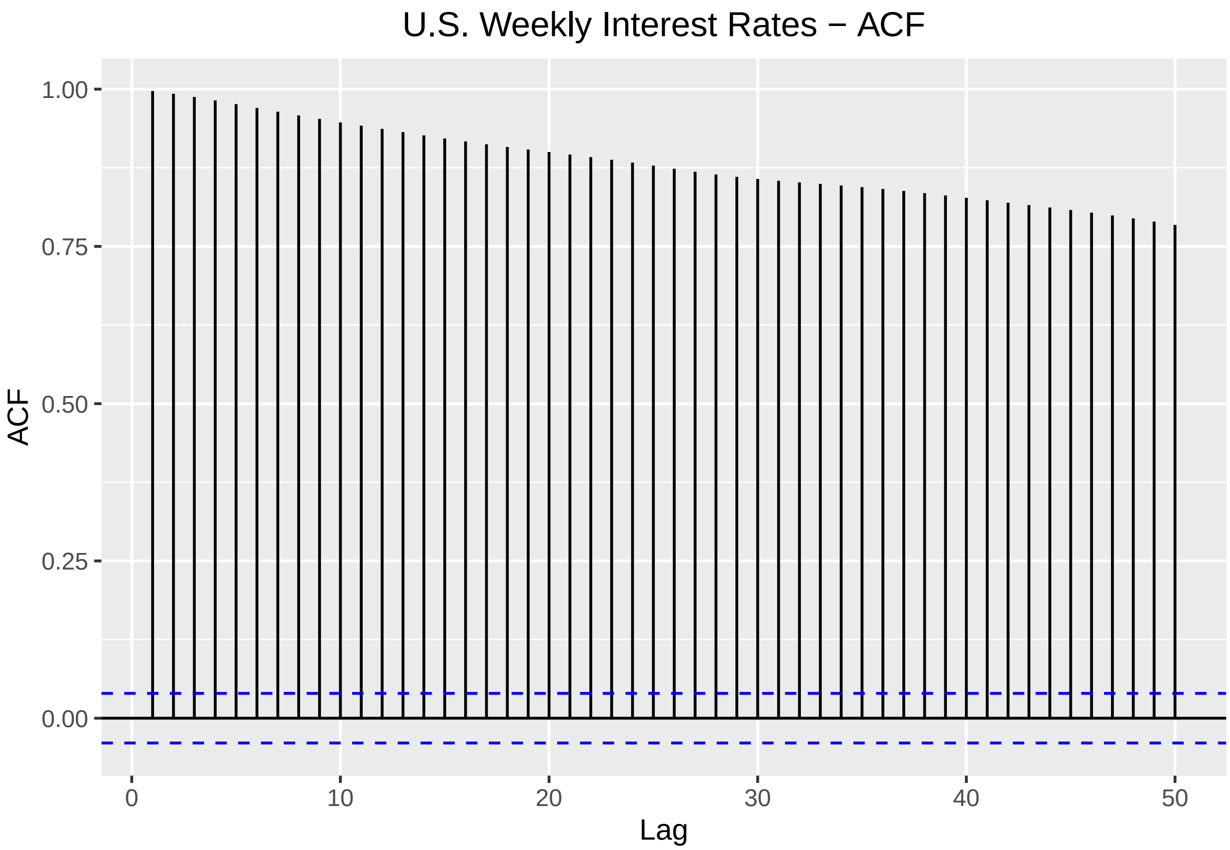
(a) Show a plot of the data, along with the respective ACF and PACF functions. Discuss the plots.

```
int_rate <- read.table(file = "/Users/icranfield/Desktop/Econ 144/Week 5/Homework 3/w-gs1yr.txt", heade
  date = ymd(paste(year, mon, day, sep="-")),
  growth = log(rate) - log(lag(rate))
  )  %>%
  dplyr::select(date, rate, growth)

rate_ts <- ts(int_rate$rate, start = decimal_date(ymd("1962-01-05")) , frequency = 52)

ggplot(int_rate, aes(date, rate)) +
```

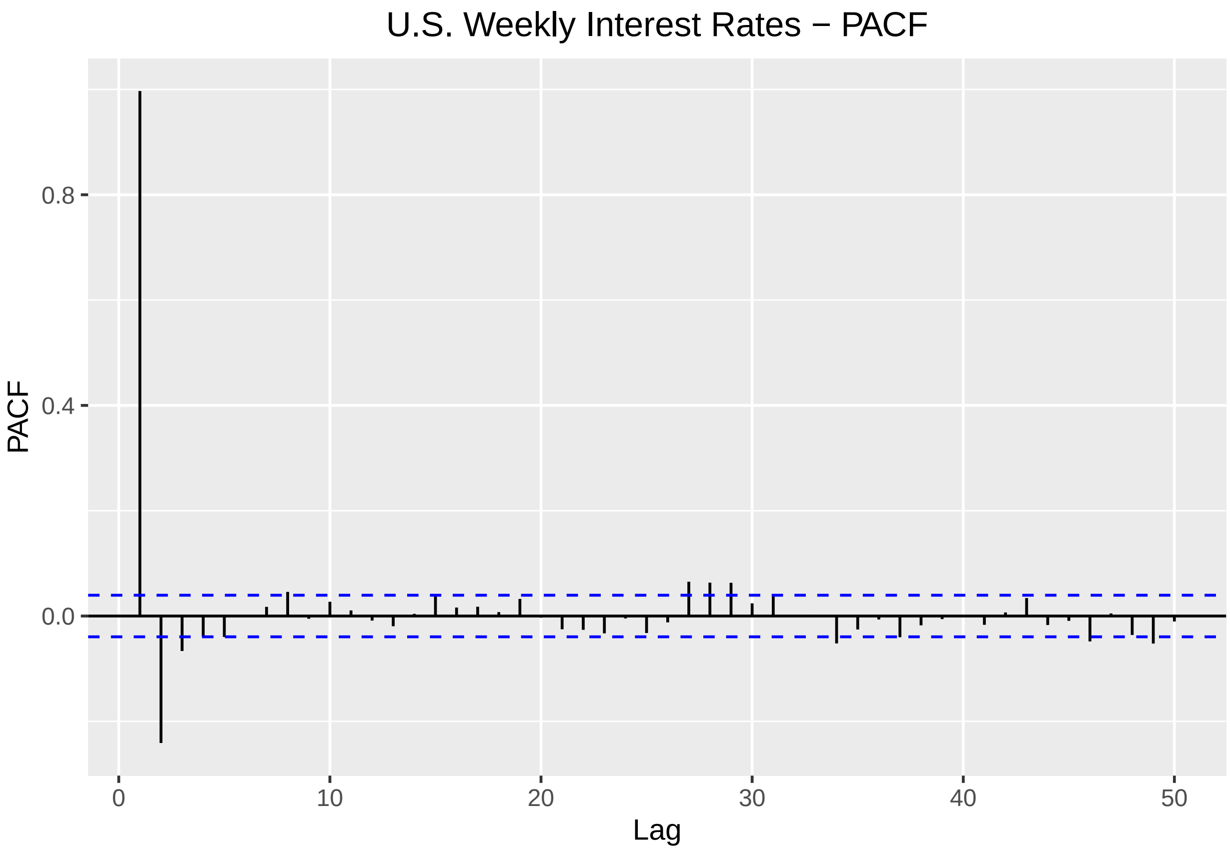
```
  geom_line() +
  ggtitle("U.S. Weekly Interest Rates 1962-2009") +
  xlab("Date") + ylab("Rate in %")
```




```
par(mfrow=c(2,1))
## ACF, slow decay to zero
ggAcf(rate_ts, lag.max = 50) + ggtitle("U.S. Weekly Interest Rates - ACF") + theme(plot.title = element_
```




```
##PACF, show strong at 1,2,3?, AR(2) or AR(3)
ggPacf(rate_ts, lag.max = 50) + ggtitle("U.S. Weekly Interest Rates - PACF") + theme(plot.title = elemen
```



Looking at the time series plot we can see persistent behavior. with highest percentages in the early eighties. The ACF plot shows a very slow decayto zero. The PACF plot has signficance at lags of 1, 2, and 3, but the third band is weaker than the first two. I would suggest an AR(2) or AR(3) model.

b. Based on your discussion in (a), fit 3 different ARMA(p,q) models, and show the fits over the original data. Discuss your results, and select one model as your preferred choice.

```r
# arma 1
arma1 <- arma(rate_ts, order = c(1,1)) # AR(1) + MA(1)
summary(arma1)
```

```
##
## Call:
## arma(x = rate_ts, order = c(1, 1))
##
## Model:
## ARMA(1,1)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -1.6083723 -0.0566853  0.0008677  0.0613990  1.4225219
##
## Coefficient(s):
##            Estimate  Std. Error  t value Pr(>|t|)
## ar1        0.997396    0.001589  627.658   <2e-16 ***
## ma1        0.295612    0.017174   17.213   <2e-16 ***
## intercept  0.014819    0.010758    1.377    0.168
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Fit:
## sigma^2 estimated as 0.03221,  Conditional Sum-of-Squares = 79.4,  AIC = -1468.08
```

```r
#arma 2
arma2 <- arma(rate_ts, order = c(2,0))
summary(arma2)
```

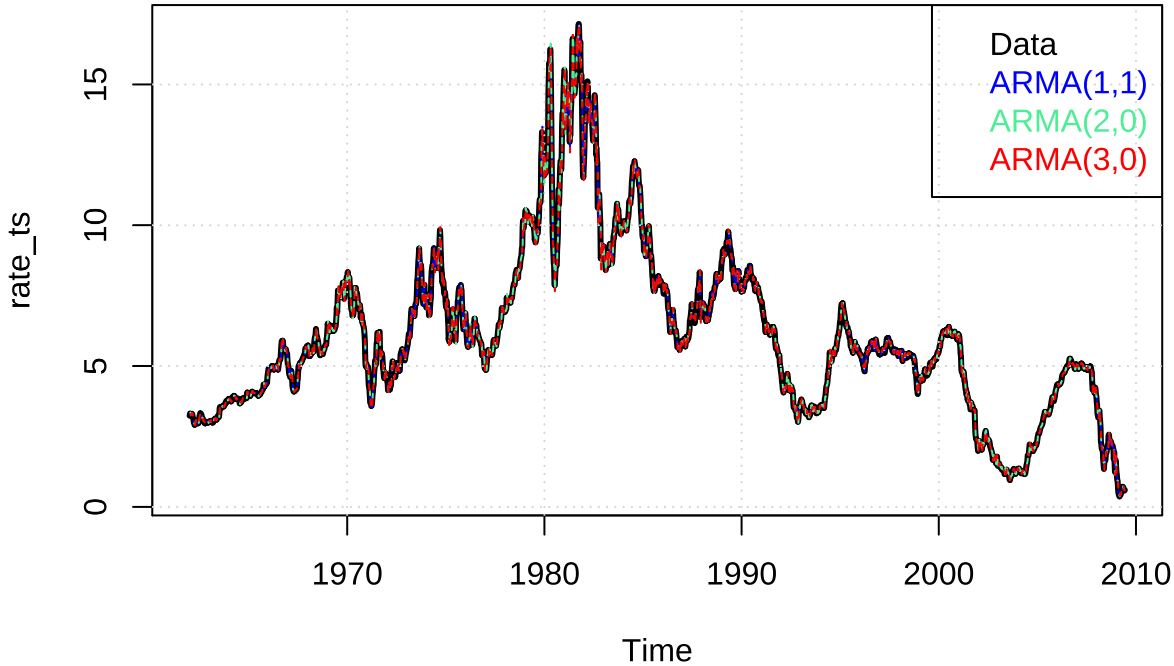
```
##
## Call:
## arma(x = rate_ts, order = c(2, 0))
##
## Model:
## ARMA(2,0)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -1.568880 -0.058626  0.001247  0.062502  1.420031
##
## Coefficient(s):
##            Estimate  Std. Error  t value Pr(>|t|)
## ar1        1.343575    0.018890   71.126   <2e-16 ***
## ar2       -0.346006    0.018900  -18.307   <2e-16 ***
## intercept  0.014101    0.008221    1.715   0.0863 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Fit:
## sigma^2 estimated as 0.03154,  Conditional Sum-of-Squares = 77.71,  AIC = -1520.3
```

```r
#arma 3
arma3 <- arma(rate_ts, order = c(3,0))
summary(arma3)
```

```
##
## Call:
## arma(x = rate_ts, order = c(3, 0))
##
## Model:
## ARMA(3,0)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -1.588599 -0.059804  0.001335  0.061573  1.429731
##
## Coefficient(s):
##            Estimate  Std. Error  t value Pr(>|t|)
## ar1        1.323516    0.020099   65.851  < 2e-16 ***
## ar2       -0.267843    0.032938   -8.132 4.44e-16 ***
## ar3       -0.058291    0.020108   -2.899  0.00375 **
## intercept  0.015300    0.008219    1.862  0.06266 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Fit:
```

```
## sigma^2 estimated as 0.03144,  Conditional Sum-of-Squares = 77.44,  AIC = -1525.81
```

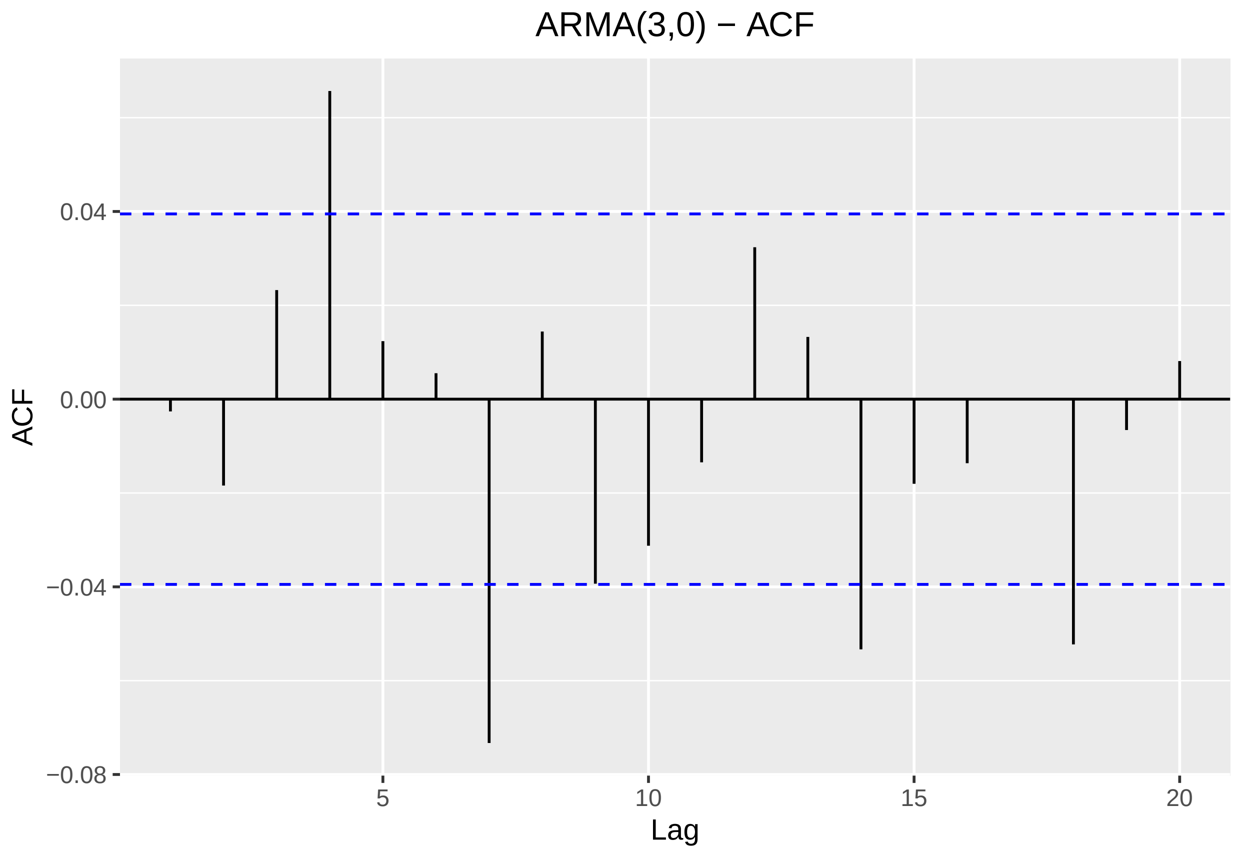
```r
plot(rate_ts, lwd = 3)
grid()
lines(arma1$fitted.values,col="blue",lwd=1, lty = 2)
lines(arma2$fitted.values,col="seagreen2",lwd=1, lty =2)
lines(arma3$fitted.values,col="red",lwd=1,lty=2)
legend("topright",legend=c("Data","ARMA(1,1)","ARMA(2,0)","ARMA(3,0)"), text.col=c("black","blue","seag
```



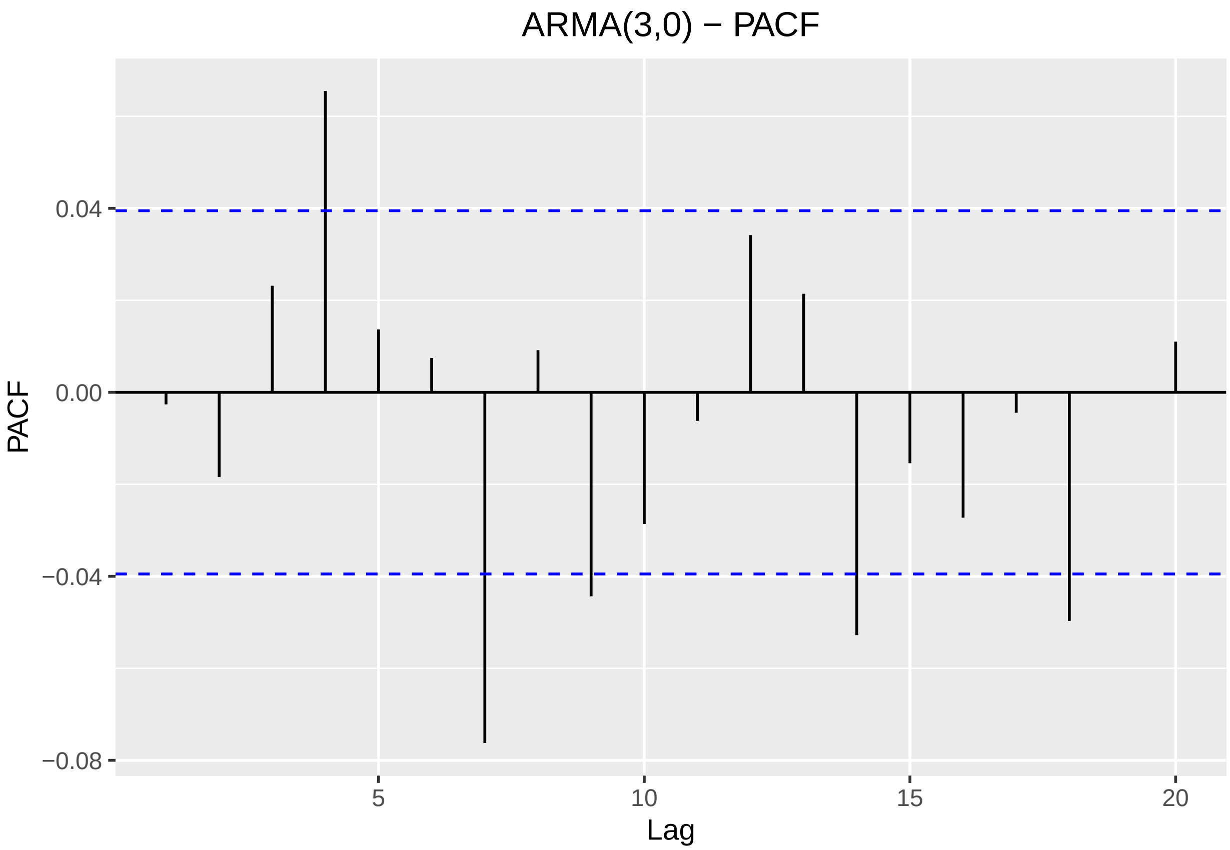
I fit 3 different ARMA models: ARMA (1,1), ARMA(2,0), and ARMA (3,0). It's hard to tell from plotting each ARMA graph, but I will go with ARMA (3,0) as my preferred choice because it has the lowest AIC score and signficant ar coefficients.

c. Now that you fit an ARMA(p,q) model to the data, plot the ACF and PACF of the residuals from your preferred fit model, and discuss your results.

```r
# Now that you fit an ARMA(p,q) model to the data, plot the ACF and PACF of the residuals from your pre

ggAcf(arma3$residuals[4:length(arma3$residuals)], lag.max = 20, type = "correlation") + ggtitle("ARMA(3
```




```
ggPacf(arma3$residuals, lag.max = 20) + ggtitle("ARMA(3,0) - PACF") + theme(plot.title = element_text(h
```



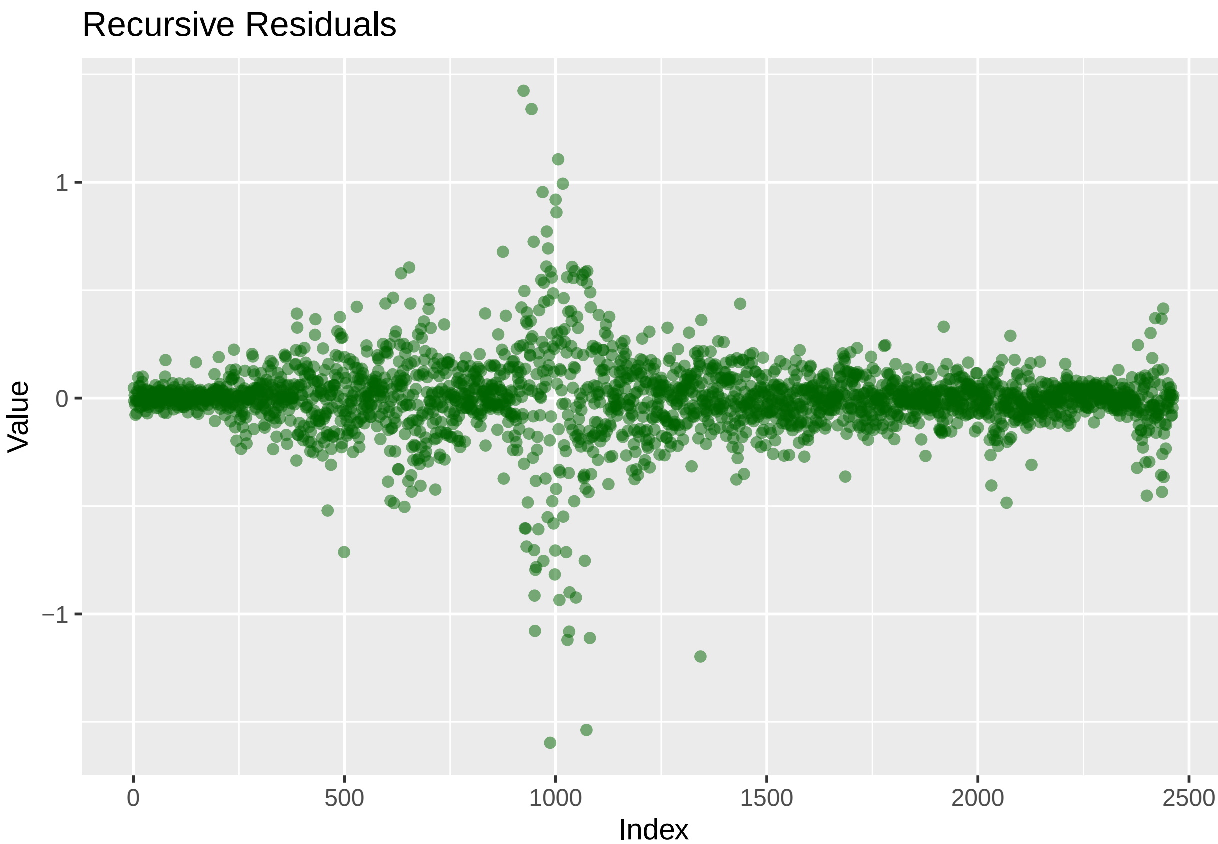
The spikes for ACF are at 4, 7, 9, 14, 18. Can't tell if this resembles white noise.

The spikes for PACF has significant spikes at the same lags as the ACF, but looking at the range of the y-axis the spikes have low correlation values.

Can not conclude it resembles white noise, but the ARMA(3,0) process did a good job of capturing the patterns.

d. Compute and plot the recursive residuals from your best fit model. Interpret the plot.

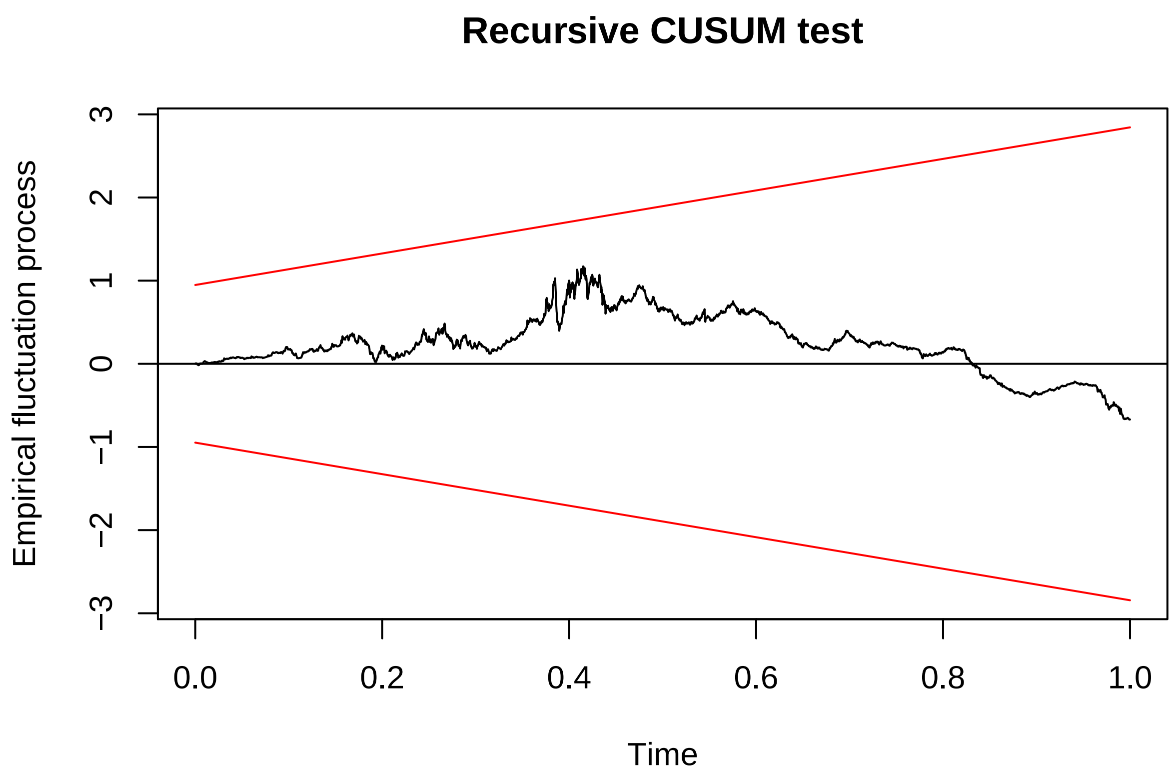
```
# (d) Compute and plot the recursive residuals from your best fit model. Interpret the plot.

recursives <- as.data.frame(recresid(arma3$residuals~1)) %>% setnames("residuals")

ggplot(recursives, aes(x=seq(nrow(recursives)), y= residuals)) +
  geom_point(color='dark green', alpha = 0.5) +
  ggtitle('Recursive Residuals') +
  xlab('Index') +
  ylab('Value')
```



Densely clustered around 0 that suggests normality. Around index 1000 which corresponds to the time period of 1980 and 1985, there is greater deviation from zero. The early eighties showed higher volatility (seen in the plot), so seeing the residuals demonstrate greater deviation is expected. Our ARMA(3,0) model did not do the best.

e. Compute and plot the CUSUM from your best fit model. Interpret the plot.

```
# (e) Compute and plot the CUSUM from your best fit model. Interpret the plot.

plot(efp(arma3$residuals~1, type = "Rec-CUSUM"))
```



The CUSUM is within the red bands so we don't see any significant breaks in the model. There is a spike around time 0.4, but it is not significant enough to break the model. Still has persistence behavior that can be captured better.

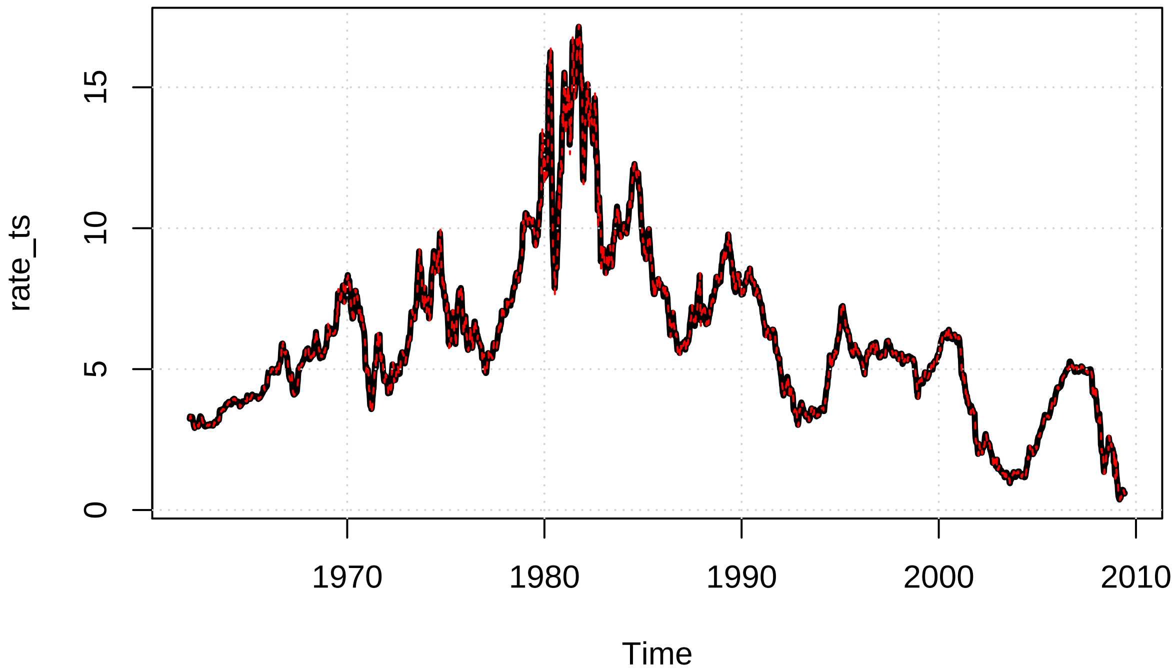
f. Compute the best fit model according to 'R' and compare it against your model. Discuss these results.

```
# (f) Compute the best fit model according to 'R' and compare it against your model. Discuss these resu

best_fit <- auto.arima(rate_ts)
summary(best_fit)
```

```
## Series: rate_ts
## ARIMA(1,1,2)
##
## Coefficients:
##          ar1      ma1      ma2
##       0.6284  -0.3065  -0.0527
## s.e.  0.0642   0.0675   0.0299
##
## sigma^2 estimated as 0.03143:  log likelihood=768.32
## AIC=-1528.65   AICc=-1528.63   BIC=-1505.41
##
## Training set error measures:
##                         ME      RMSE       MAE         MPE     MAPE       MASE
## Training set -0.0006210918 0.1771539 0.1046884 -0.05084758 1.820023 0.07539674
##                       ACF1
## Training set -0.0002601686
```

```
plot(rate_ts, lwd = 3)
grid()
```

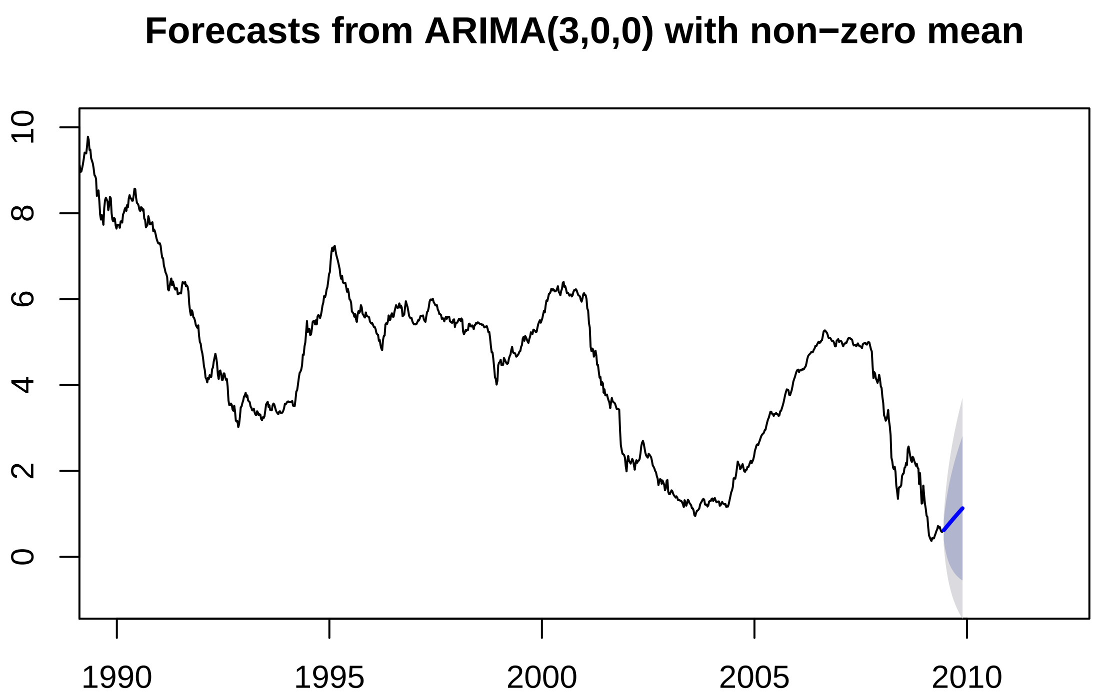
```r
lines(best_fit$fitted,col="red",lwd=1, lty = 2)
```



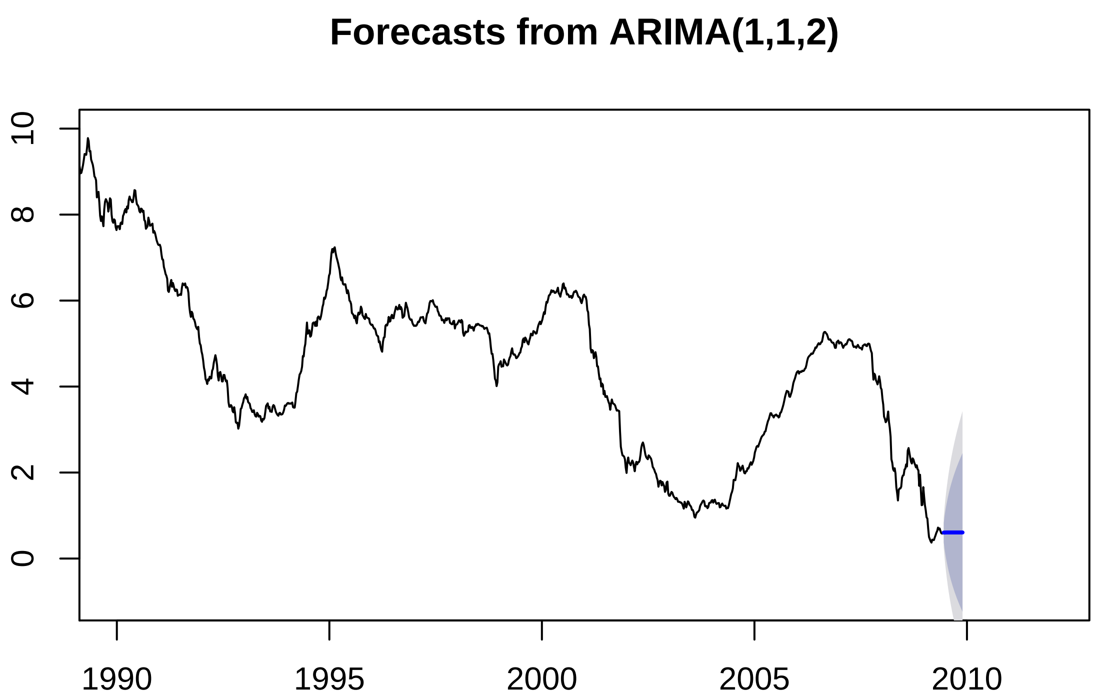
R's best fit model is an ARIMA(1,1,2). So this model has AR process of 1 and a MA process of 2. This model has a better AIC score than my arma(3,0) model.

g. Using your best fit model as well as 'R's' best fit model, compute the respective 24-steps-ahead forecast, and compare your results.

```r
# (g) Using your best fit model as well as 'R's' best fit model, compute the respective 24- steps-ahead

# my best fit model
forecast_arma3 <- Arima(rate_ts,order=c(3,0,0))

plot(forecast(forecast_arma3,h=24), xlim = c(1990,2012), ylim = c(-1,10))
```




```r
# R's best fit model

plot(forecast(best_fit, h =24), xlim = c(1990,2012), ylim = c(-1,10) )
```



My best fit model shows an upward trend and R's best fit model shows a relatively flat forecast. It is hard to tell which forecast is better, but visually and looking at the past behavior I would think the arma(3,0) forecast is better. But if R chose the ARIMA(1,1,2) as the best fit for the data, I would expect this to be the better forecast.

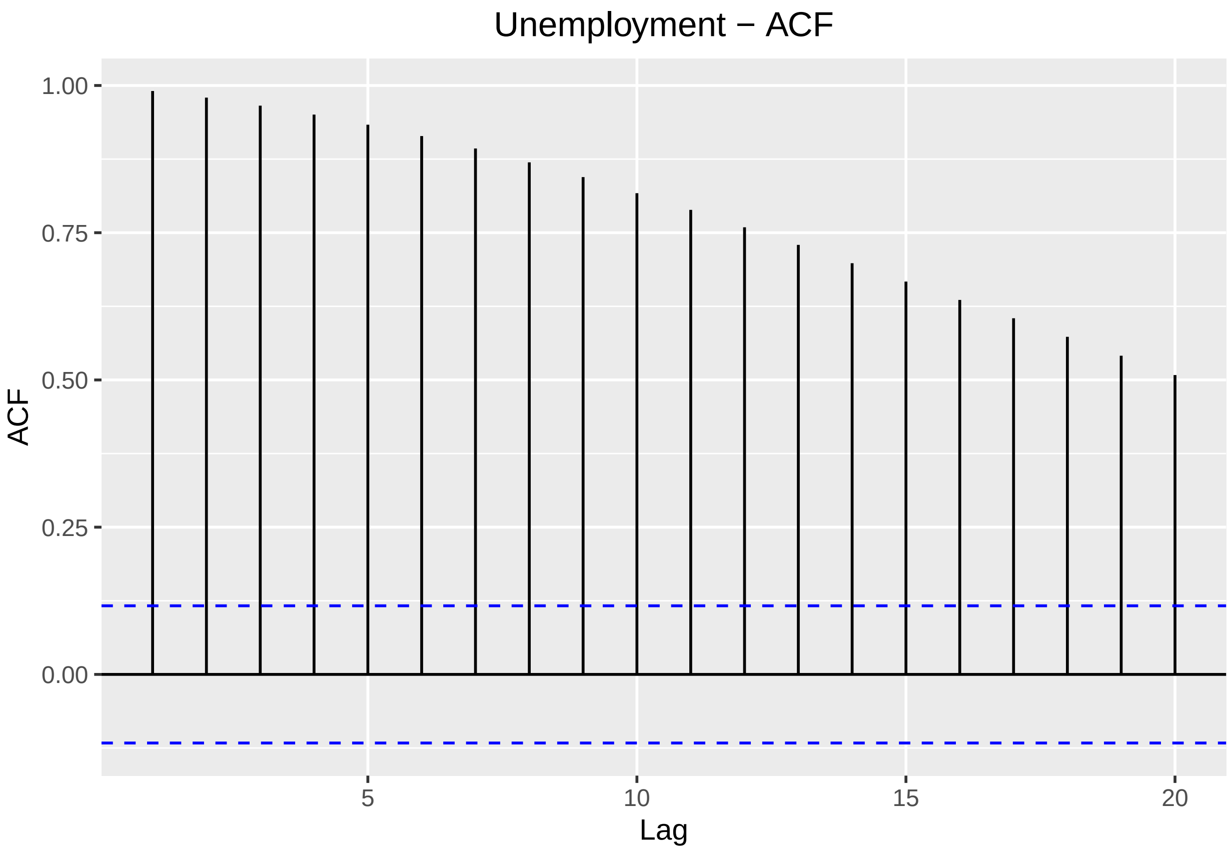
Problem 7.2:

```
getwd()
```

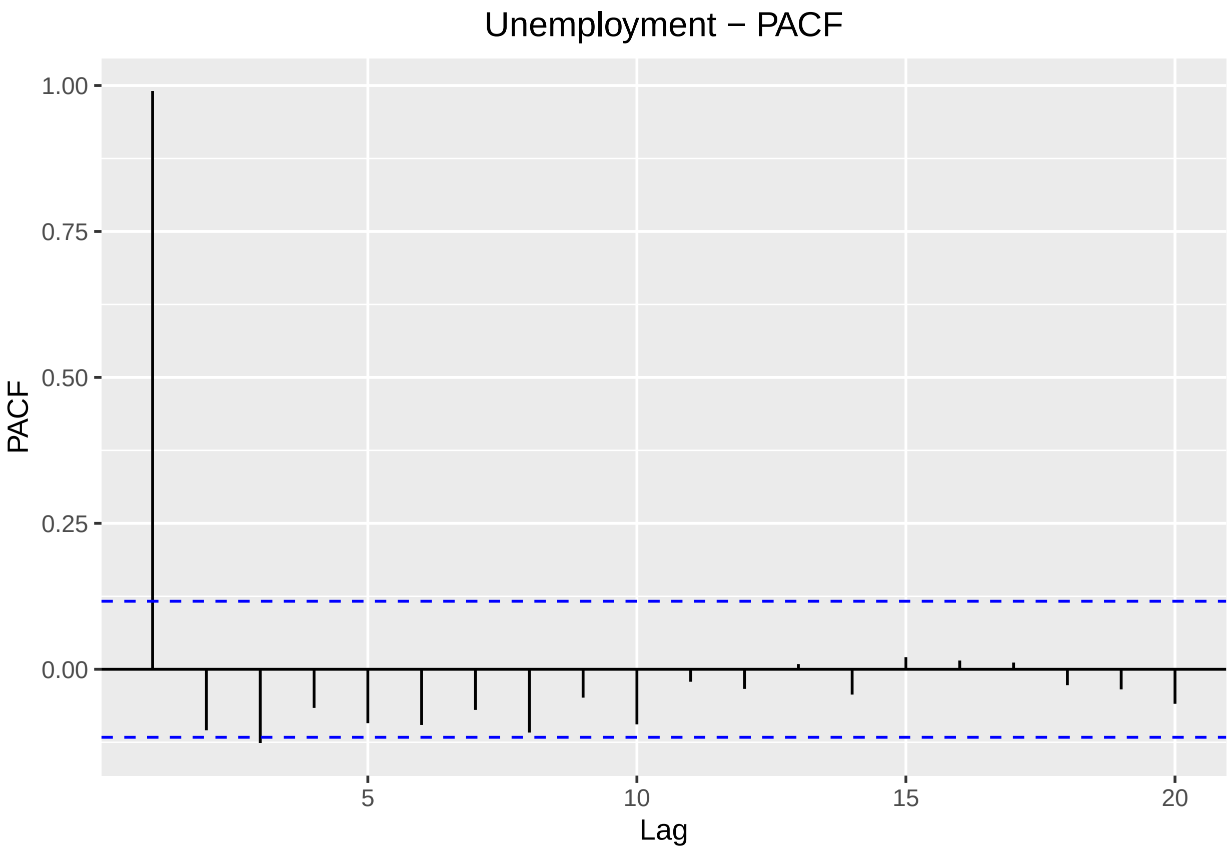
```
## [1] "/Users/icranfield/Desktop/Econ 144/Week 5/Homework 3"
```

```
unemploy_persons <- read_excel("/Users/icranfield/Desktop/Econ 144/Week 5/Homework 3/Chapter7_Exercises_
unemploy_persons <- setnames(unemploy_persons, c('date','unemployed'))
ts_unemployed <- ts(unemploy_persons$unemployed, start = c(1989,1), frequency = 12)


autoplot(ts_unemployed) + ggtitle('Unemployed Persons', '1989-2012') +
  ylab('Unemployed (thousands)') +
  xlab('Date')
```


```
ggAcf(ts_unemployed, lag.max = 20) + ggtitle("Unemployment - ACF") + theme(plot.title = element_text(hju
```




```
ggPacf(ts_unemployed, lag.max = 20) + ggtitle("Unemployment - PACF") + theme(plot.title = element_text(
```

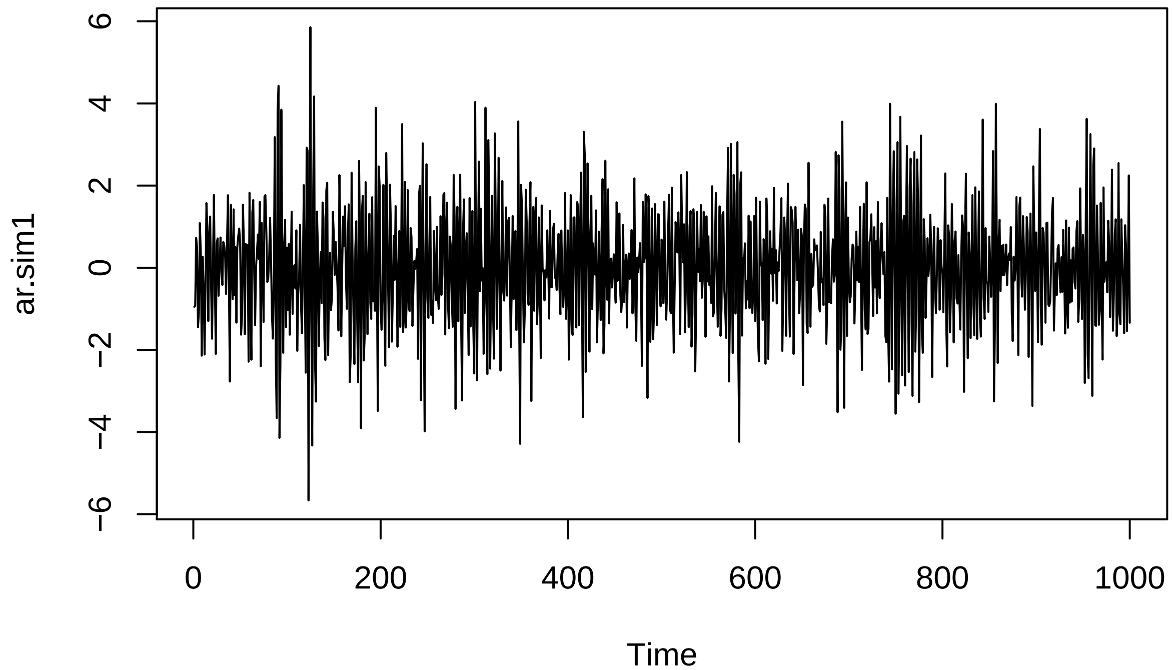


The ACF and PACF show significant spikes at lag 1 with high persistence. Looking at the PACF it suggests an AR(1) process and the ACF slowly decays to zero. Looking at the time series, it does not show a covariance stationary process.

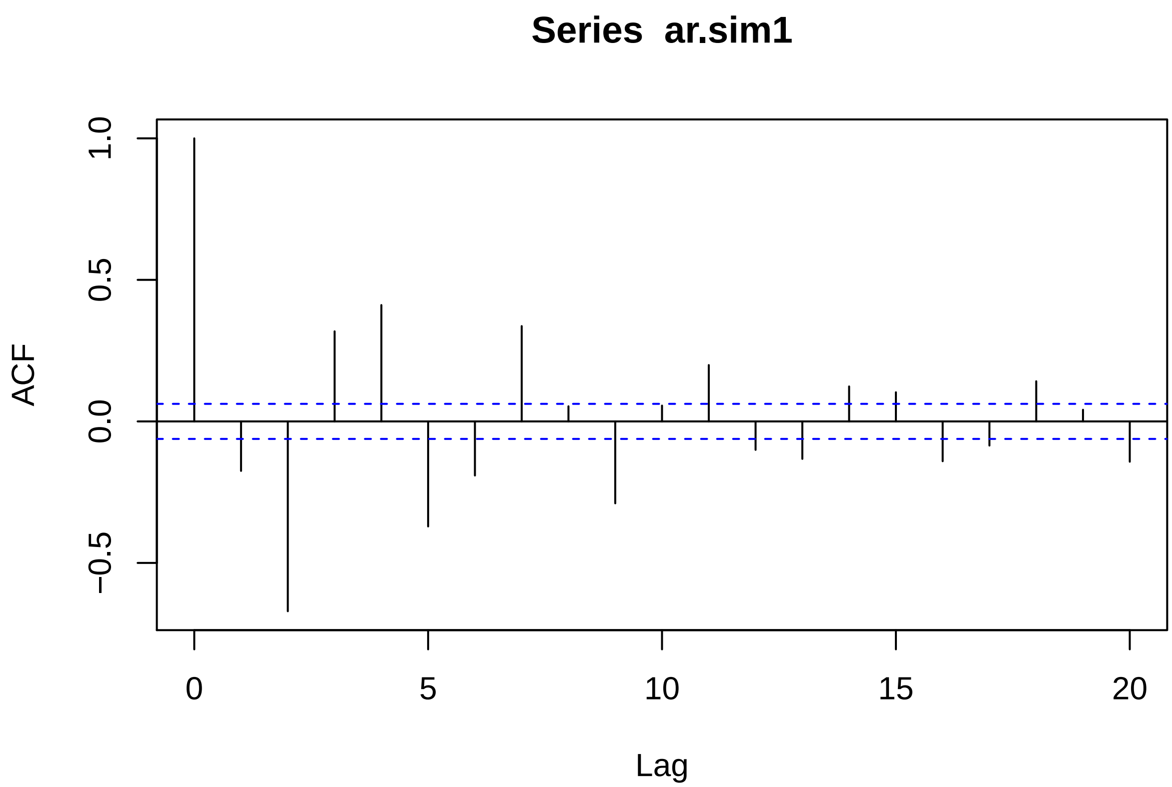
Problem 7.5:

Simulate the following AR(2) processes: Yt = 1 + 0.3Yt-1 + 0.7Yt-2 +Et Yt = 1 - 0.3Yt-1 - 0.7Yt-2 +Et for Et → N(0, 1) . Comment on their differences: Contrast their time series and their autocorrelation functions. Comment on the covariance-stationary properties of both processes.

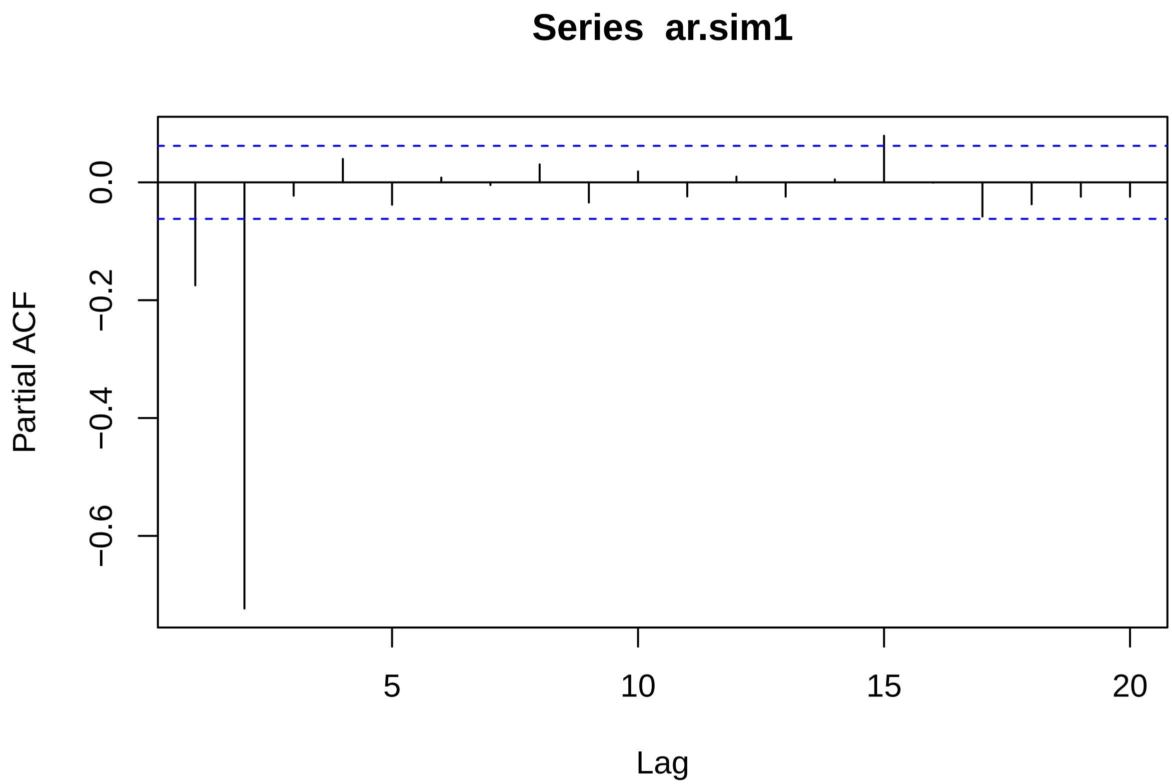
```r
# a.
# ts(arima.sim(model=list(ar=c(.3,.7)),n=100) )

#b. AR(2)
ar.sim1 <- arima.sim(model = list(ar = c(-.3, - 0.7)), n = 1000)


plot(ar.sim1)
```




```r
acf(ar.sim1, lag.max = 20)
```



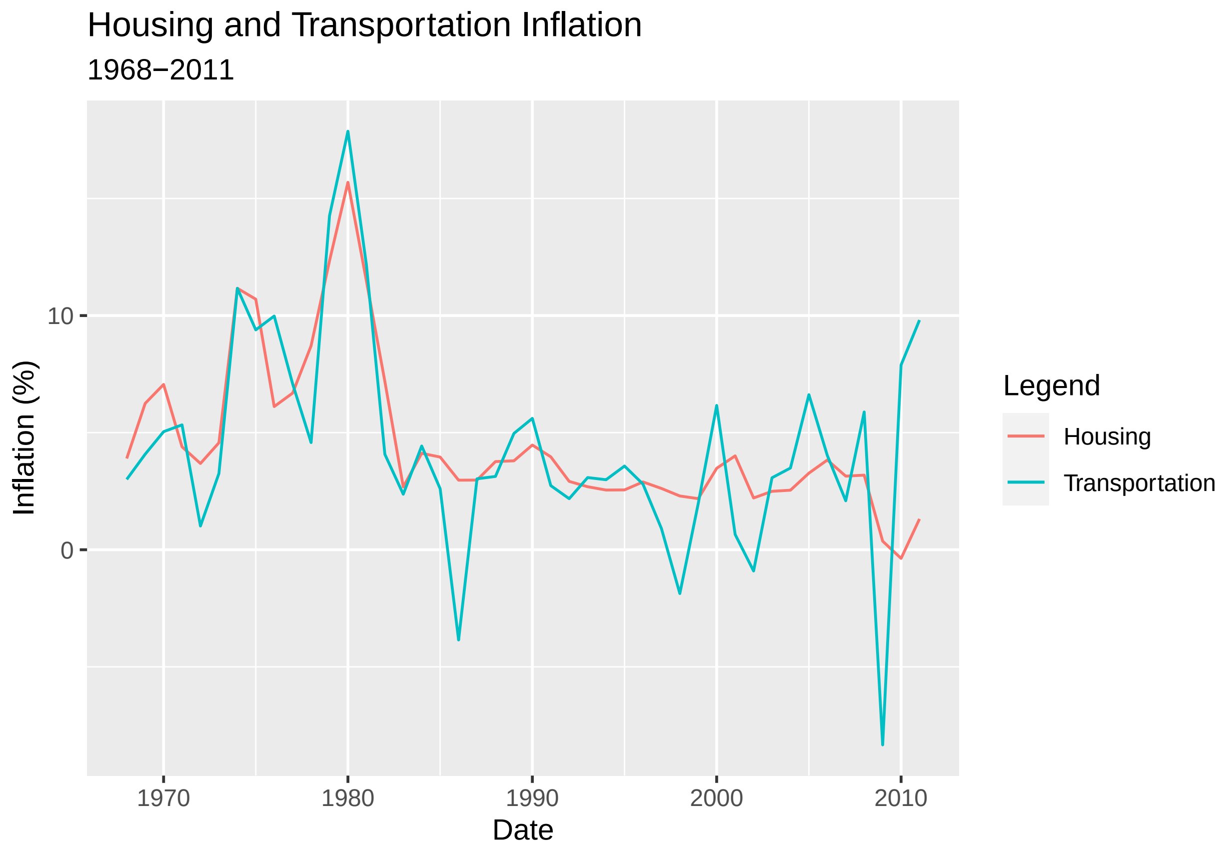

```r
pacf(ar.sim1, lag.max = 20)
```



The first equation is not able to be plotted or simulated because it is not stationary because the coefficients add up to 1. The second equation satisfies both the necessary and sufficient conditions for covariance-stationarity. The ACF plot decays to zero in aan alternating fashion while the PACF has significant spikes at lag 1 and 2.

Problem 7.6:

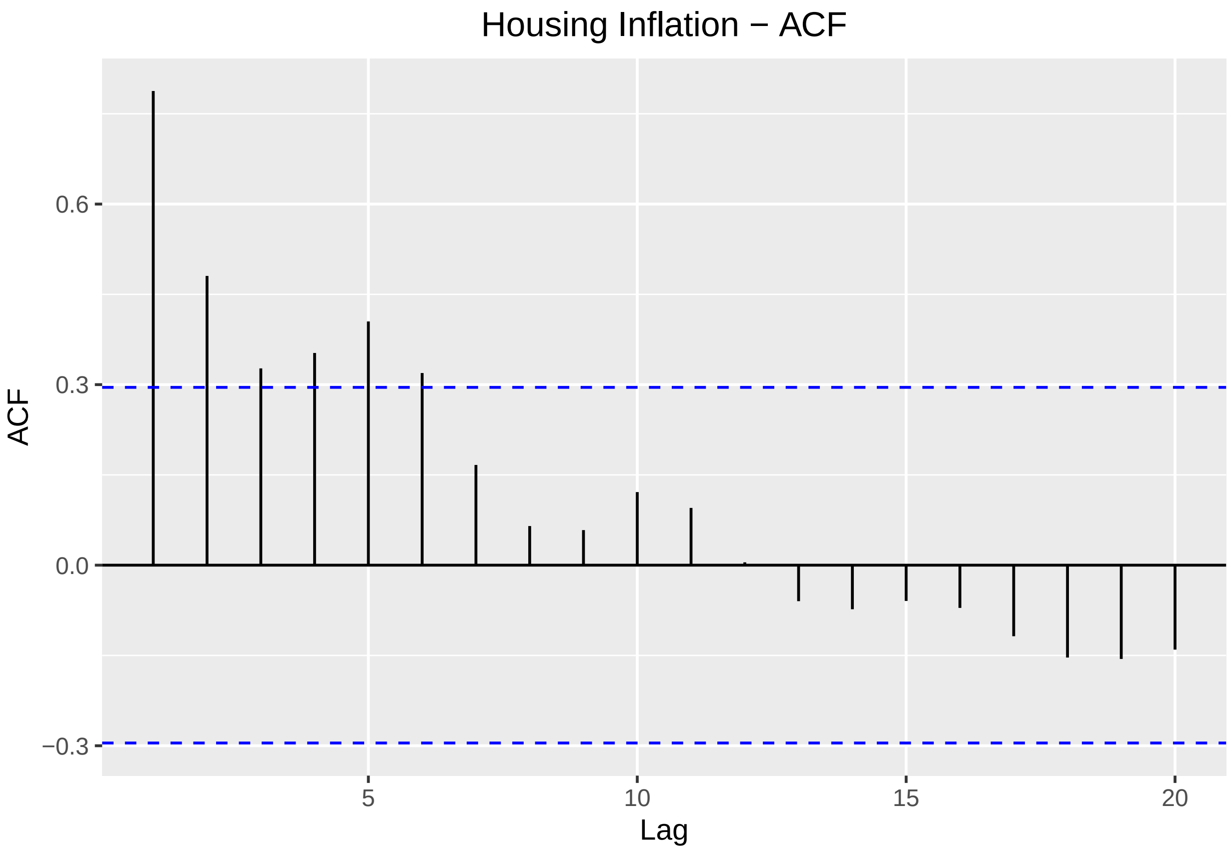
```
cpi_bls <- read_excel("/Users/icranfield/Desktop/Econ 144/Week 5/Homework 3/Chapter7_Exercises_Data.xls


ggplot(cpi_bls, aes(x = date)) +
  geom_line(aes(y=housing_inflation, color = 'Housing')) +
  geom_line(aes(y=transport_inflation, color = 'Transportation')) +
  ggtitle("Housing and Transportation Inflation", subtitle = "1968-2011") +
  ylab("Inflation (%)") + xlab("Date") + labs(colour = "Legend")
```



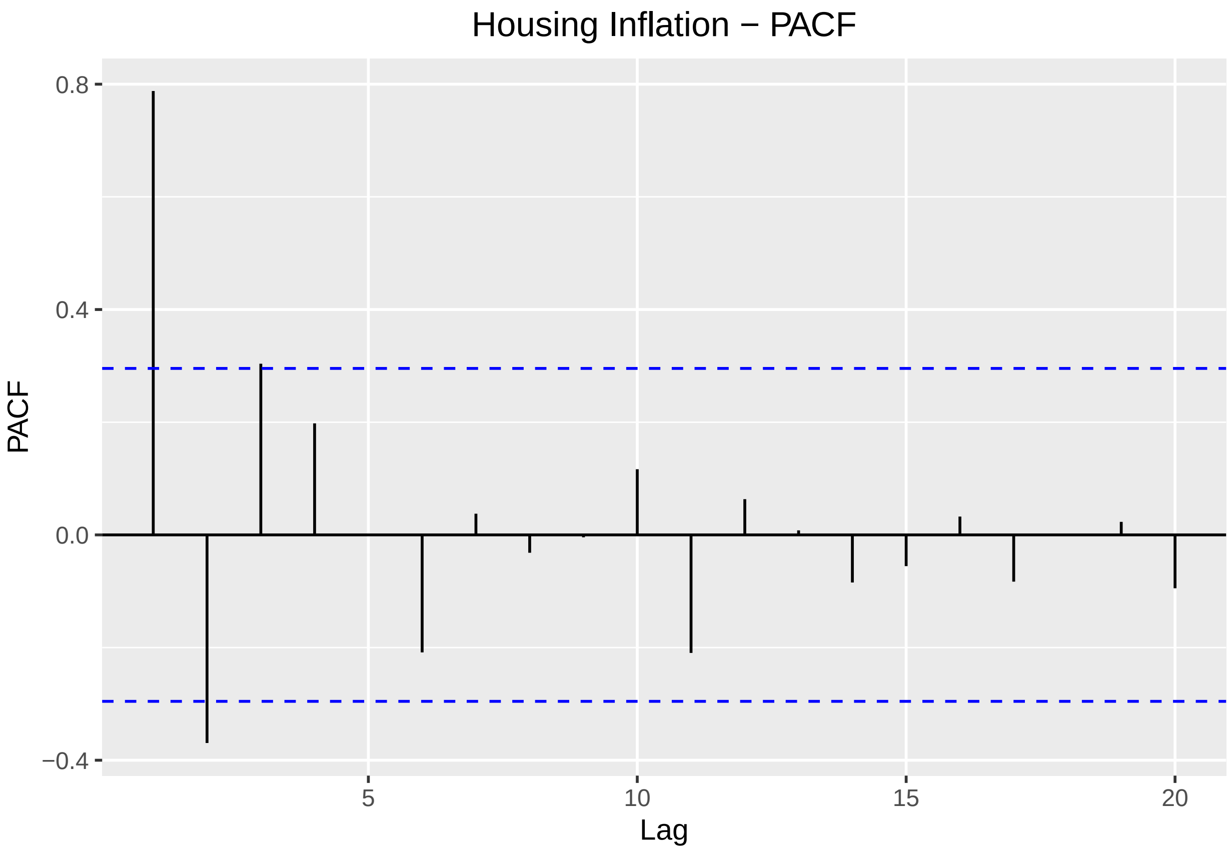
The time series plot shows the inflation rates for both Housing and Transportation from 1968-2011. Both have strong persistence. And high values of inflationin the 1970's and early 1980's.

Housing Inflation ACF and PACF

```
ggAcf(cpi_bls$housing_inflation, lag.max = 20) + ggtitle("Housing Inflation - ACF") + theme(plot.title =
```



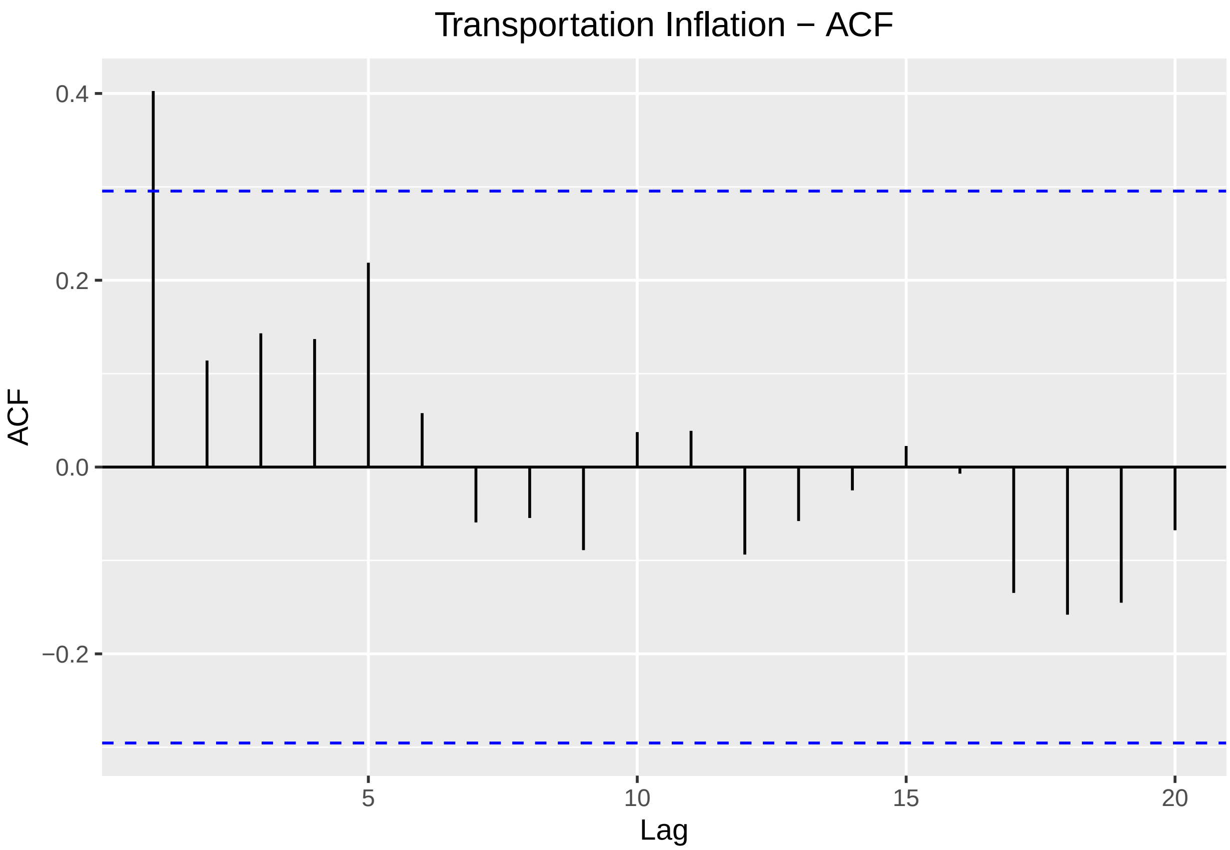

```
ggPacf(cpi_bls$housing_inflation, lag.max = 20) + ggtitle("Housing Inflation - PACF") + theme(plot.titl
```



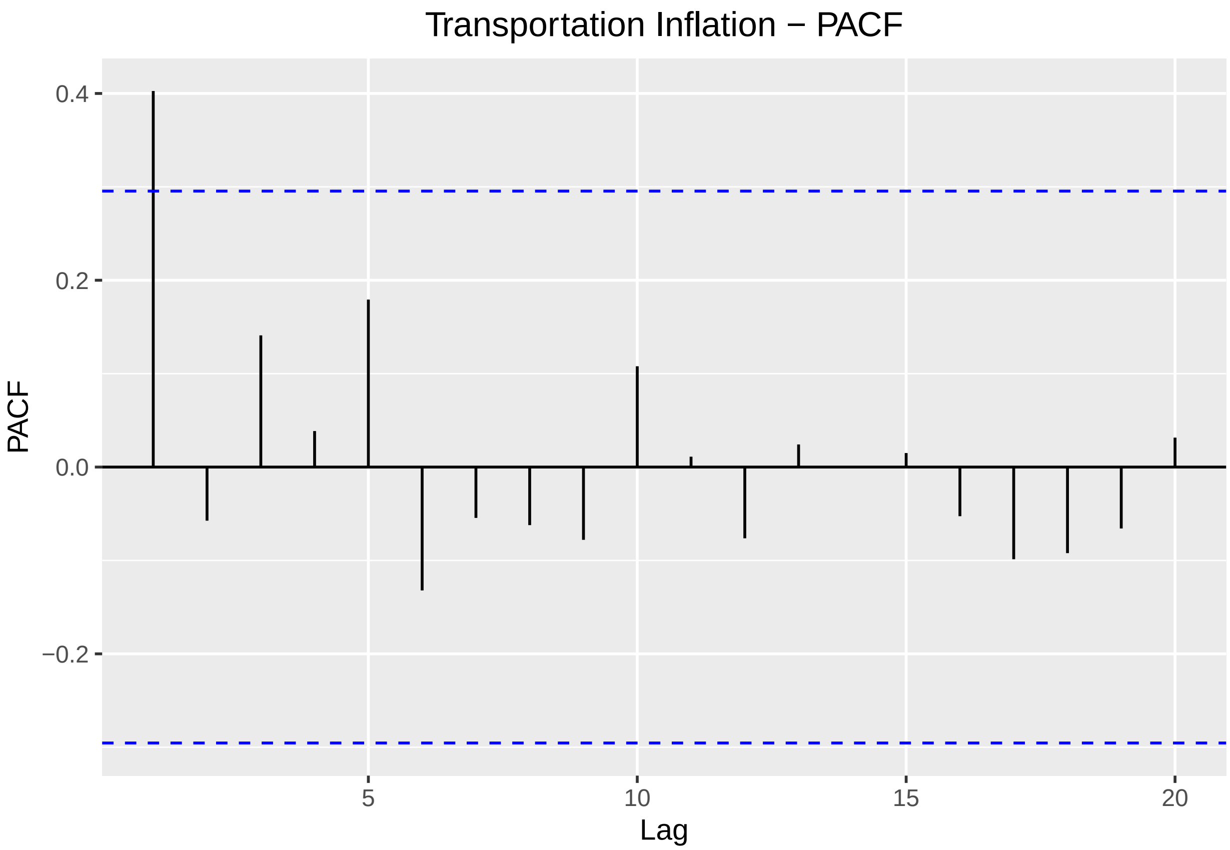
ACF decays to zero at lag 6. The PACF suggests that it is a AR(2) process. There are signicant spikes at 1 and 2. At lag 3 it shows another significant spike but we can test this.

Transport Inflation ACF and PACF

```
ggAcf(cpi_bls$transport_inflation, lag.max = 20) + ggtitle("Transportation Inflation - ACF") + theme(pl
```



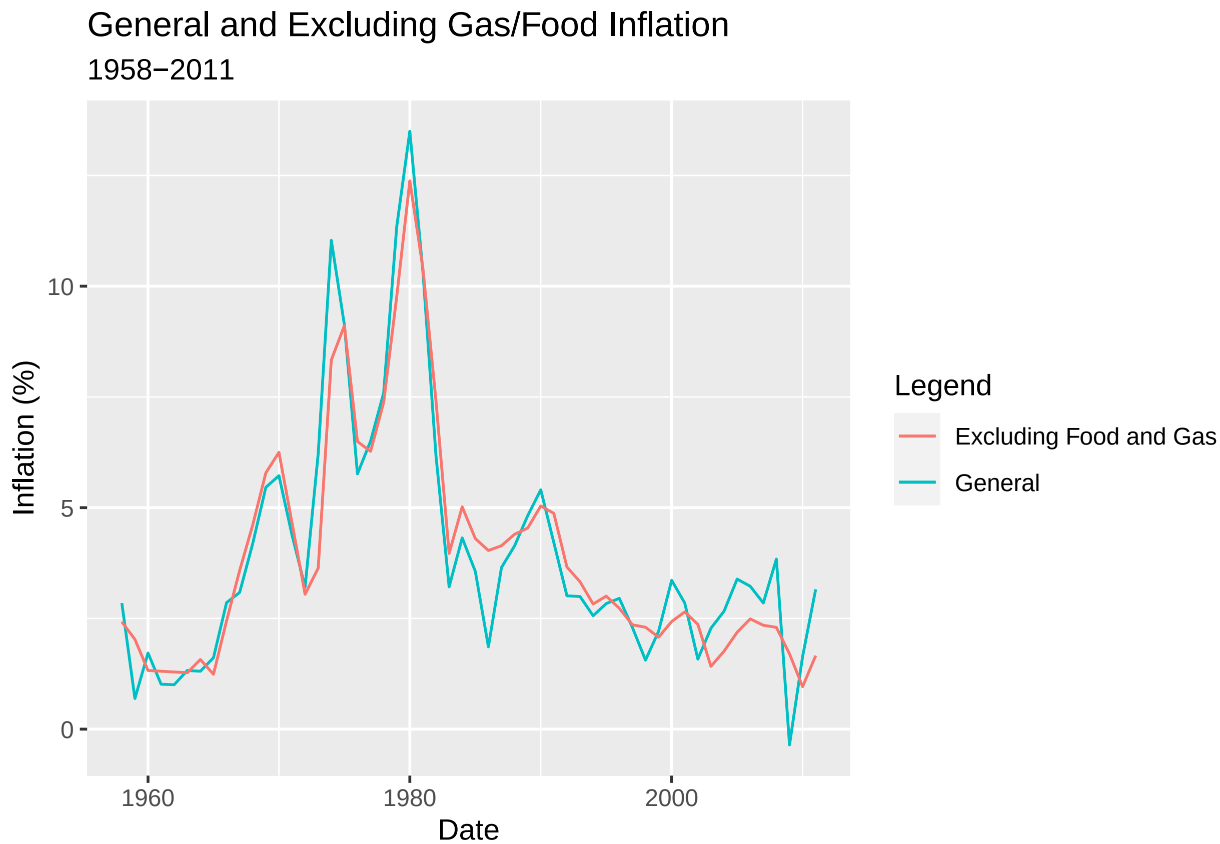

```
ggPacf(cpi_bls$transport_inflation, lag.max = 20) + ggtitle("Transportation Inflation - PACF") + theme(p
```



ACF decays to zero and the PACF has a significant spike at 1 so we can expect it to be an AR(1) process.

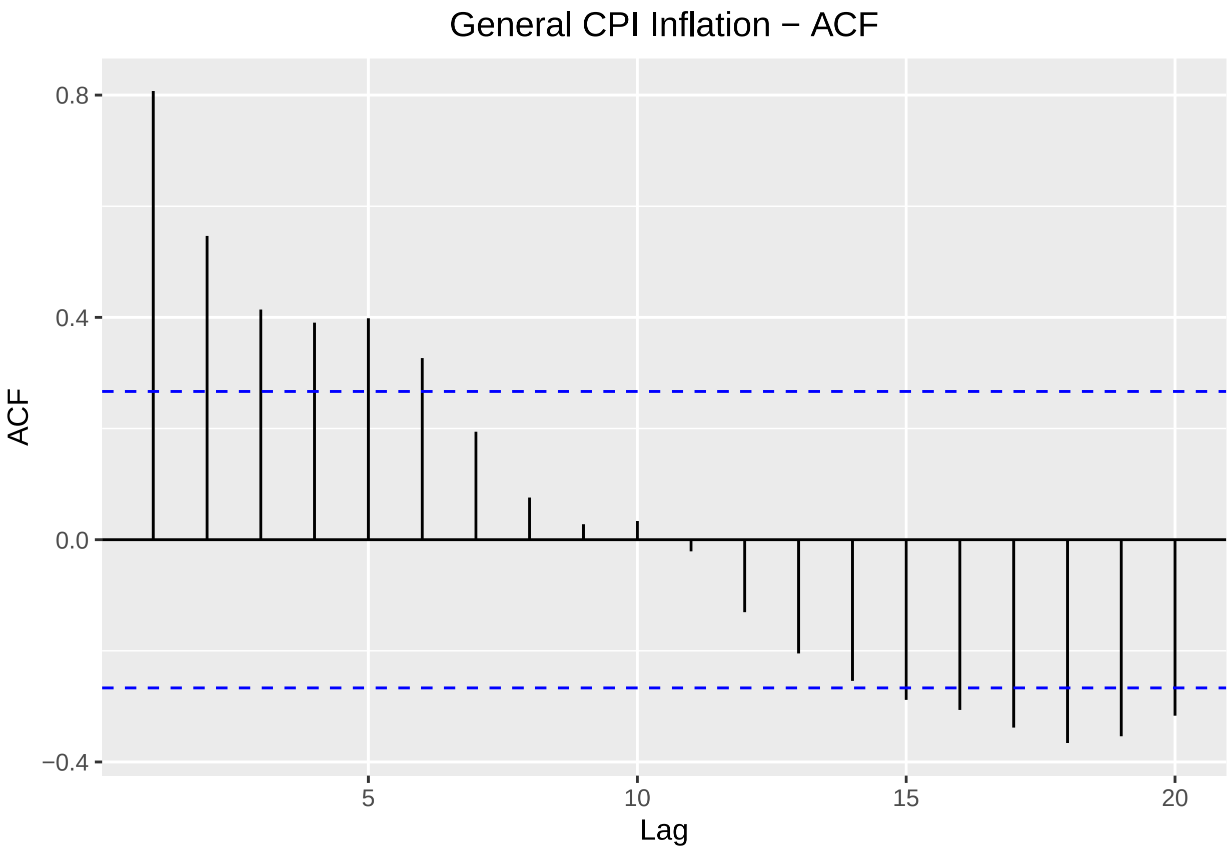
Problem 7.8:

```
cpi_general <- read_excel("/Users/icranfield/Desktop/Econ 144/Week 5/Homework 3/Chapter7_Exercises_Data


ggplot(cpi_general, aes(x = date)) +
  geom_line(aes(y=allitem_inflation, color = 'General')) +
  geom_line(aes(y=lessfood_inflation, color = 'Excluding Food and Gas')) +
  ggtitle("General and Excluding Gas/Food Inflation", subtitle = "1958-2011") +
  ylab("Inflation (%)") + xlab("Date") + labs(colour = "Legend")
```

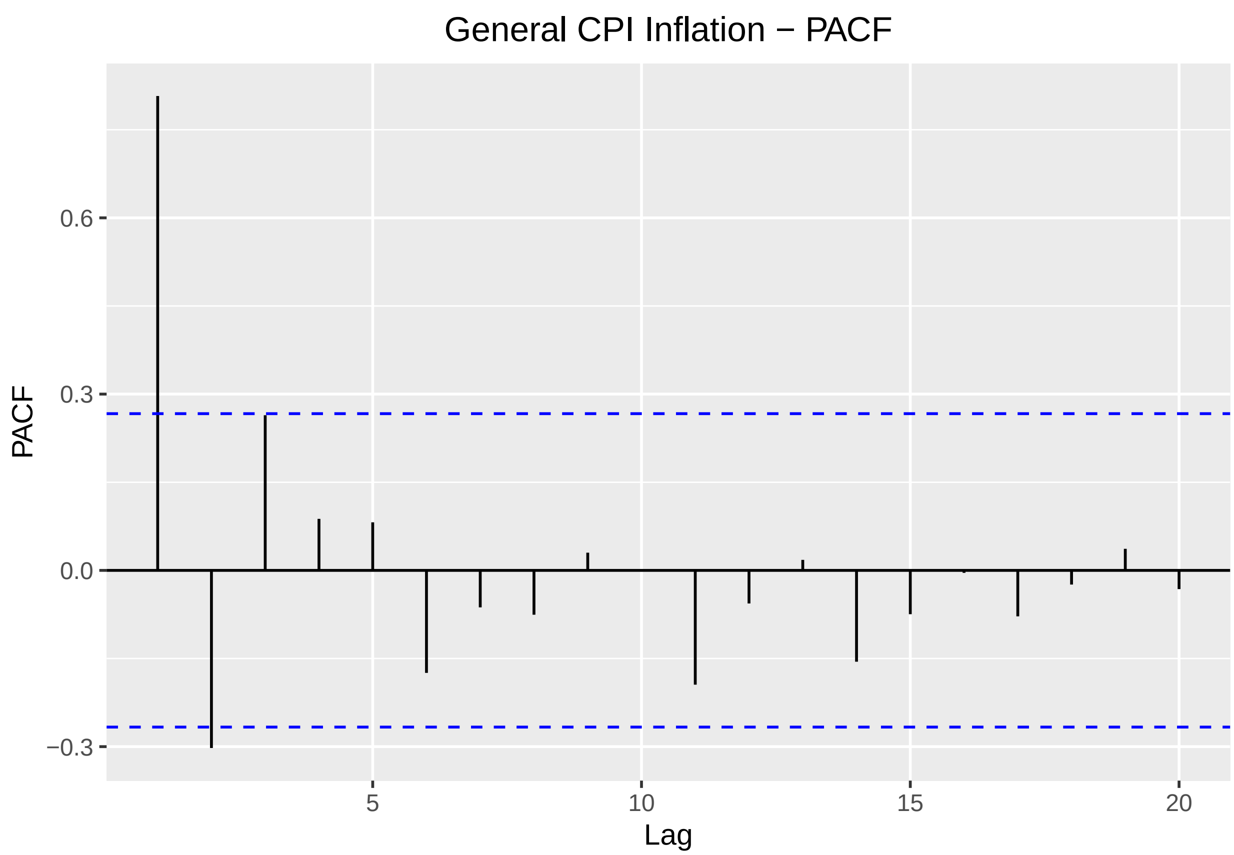


The time series plot of both inflation show strong persistence and more volatility in the Inflation Rates for General CPI.

```
## acf
ggAcf(cpi_general$allitem_inflation, lag.max = 20) + ggtitle("General CPI Inflation - ACF") + theme(plo
```

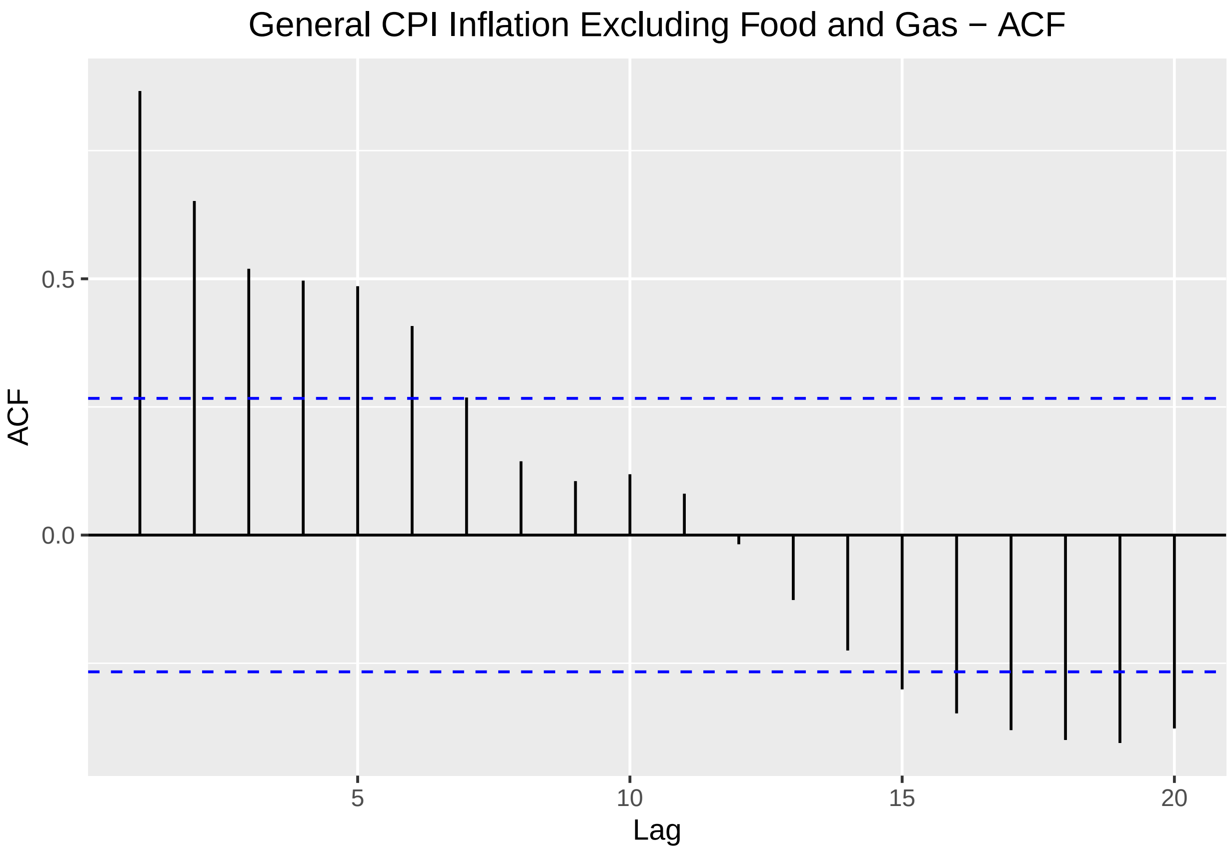



```r
##pacf
ggPacf(cpi_general$allitem_inflation, lag.max = 20) + ggtitle("General CPI Inflation - PACF") + theme(pl
```

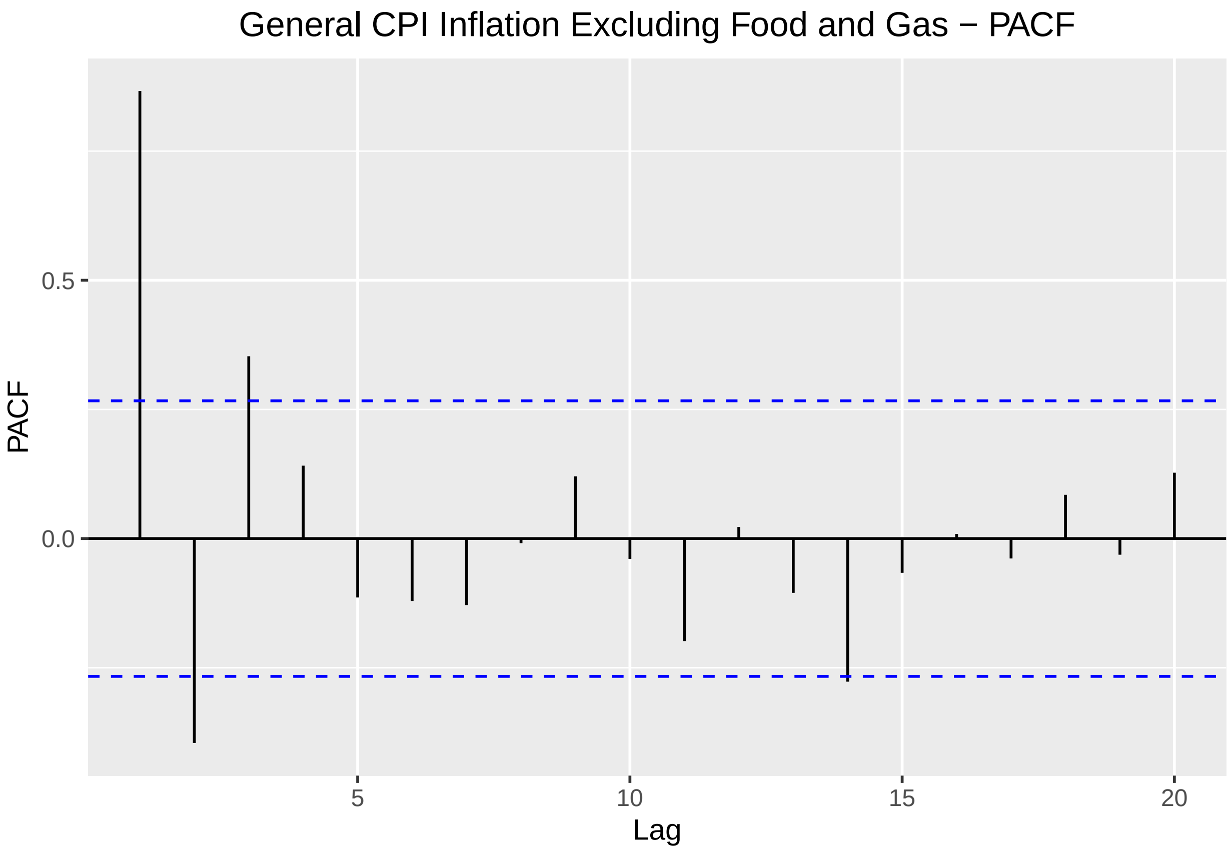


The ACF plot of General CPI inflation decays to zero and the PACF plot has significant spikes at 1,2,3 suggesting a AR(3) process.

```
## acf
ggAcf(cpi_general$lessfood_inflation, lag.max = 20) + ggtitle("General CPI Inflation Excluding Food and
```

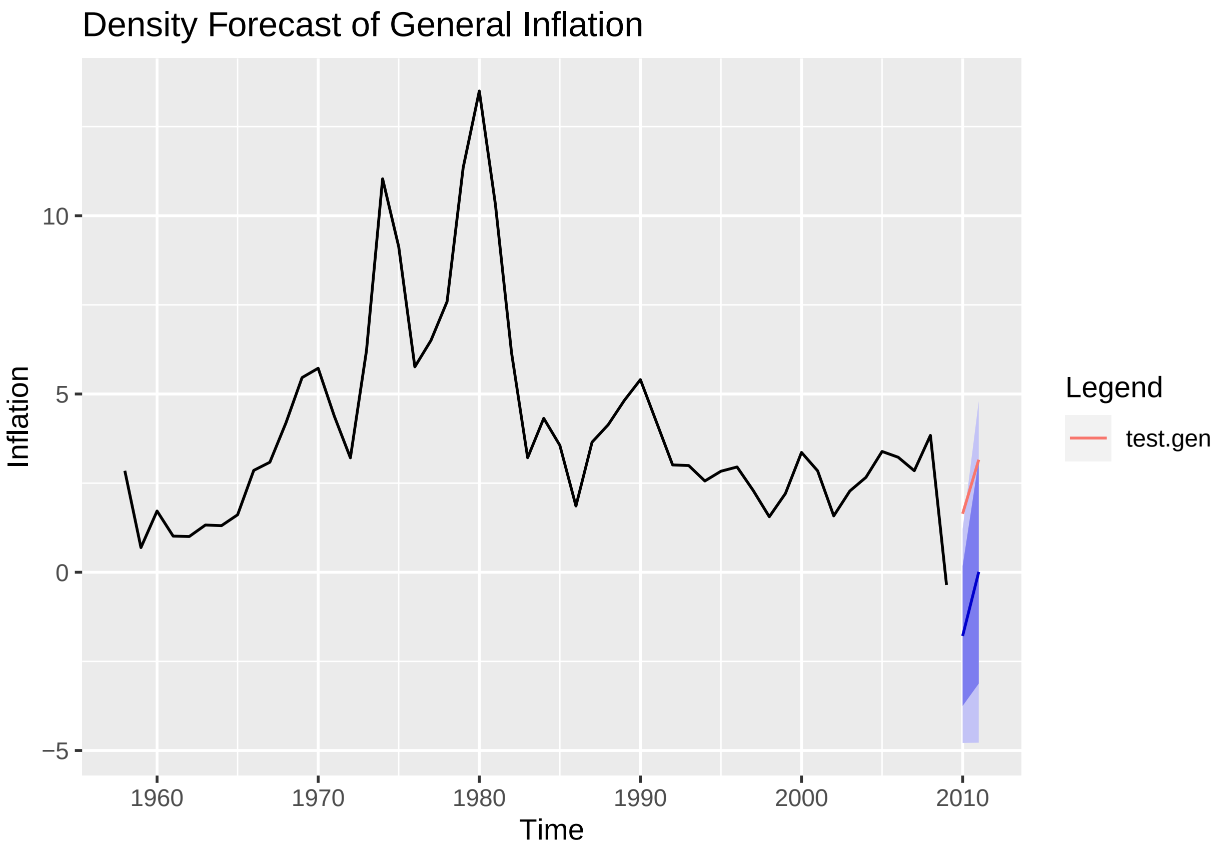



```
##pacf
ggPacf(cpi_general$lessfood_inflation, lag.max = 20) + ggtitle("General CPI Inflation Excluding Food and
```



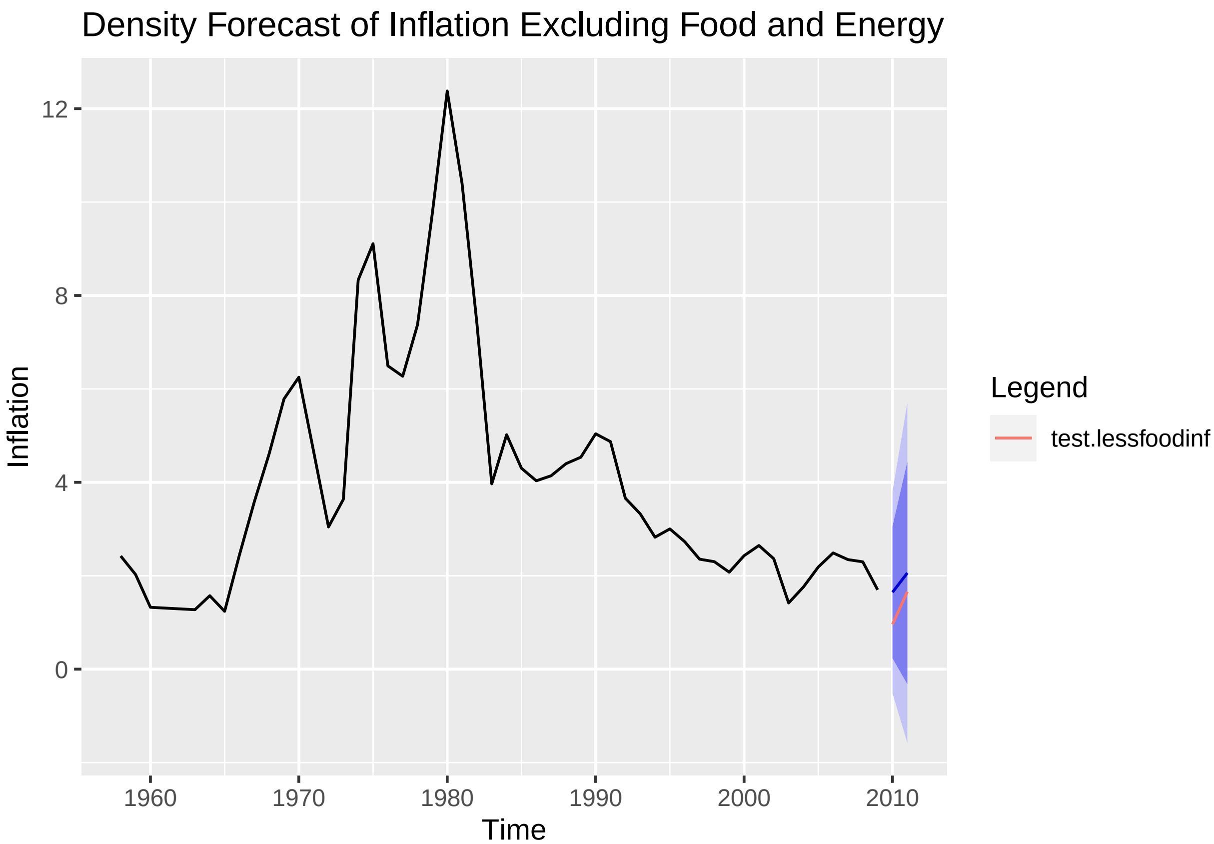
The ACF and PACF plot of the General CPI Inflation excluding Food and Gas is similar to the one of just general inflation. It shows an AR(3) process.

```
# make it a ts
cpi.gen <- ts(cpi_general$allitem_inflation, start = c(1958,1), frequency = 1)

# fit an ARUMA model to the data
general.train <- Arima(cpi_general$allitem_inflation,order=c(3,0,0))

# create training observation
training.gen <- subset(cpi.gen, end = length(cpi.gen) - 2)

# observations we will test
test.gen <- subset(cpi.gen, start=length(cpi.gen)-1)

# arima model of the training data
gen.train <- Arima(training.gen, order=c(3,0,0))

#forecast
gen.train %>%
  forecast(h=2) %>%
  autoplot() + autolayer(test.gen)  + ggtitle("Density Forecast of General Inflation") + ylab("Inflatio
```




```
arma5 <- arma(cpi_general$lessfood_inflation, order = c(3,0))

cpi.lessfood <- ts(cpi_general$lessfood_inflation, start = c(1958,1), frequency = 1)


lessfood.train <- Arima(cpi_general$lessfood_inflation,order=c(3,0,0))

training.lessfoodinf <- subset(cpi.lessfood, end = length(cpi.lessfood) - 2)

test.lessfoodinf <- subset(cpi.lessfood, start=length(cpi.lessfood)-1)

lessfood.train <- Arima(training.lessfoodinf, order=c(3,0,0))

lessfood.train %>%
  forecast(h=2) %>%
  autoplot() + autolayer(test.lessfoodinf, legend = FALSE) + ggtitle("Density Forecast of Inflation Excl
```

```
## Warning: Ignoring unknown parameters: legend
```



The forecast for the inflation excluding gas and food provides a more accurate prediction seen in the plot. The exclusion of gas in the inflation series provides a stronger prediction base. The forecast of general inflation is less accurate and difficult to predict because of the volatility in the time series.

Problem 8.6:

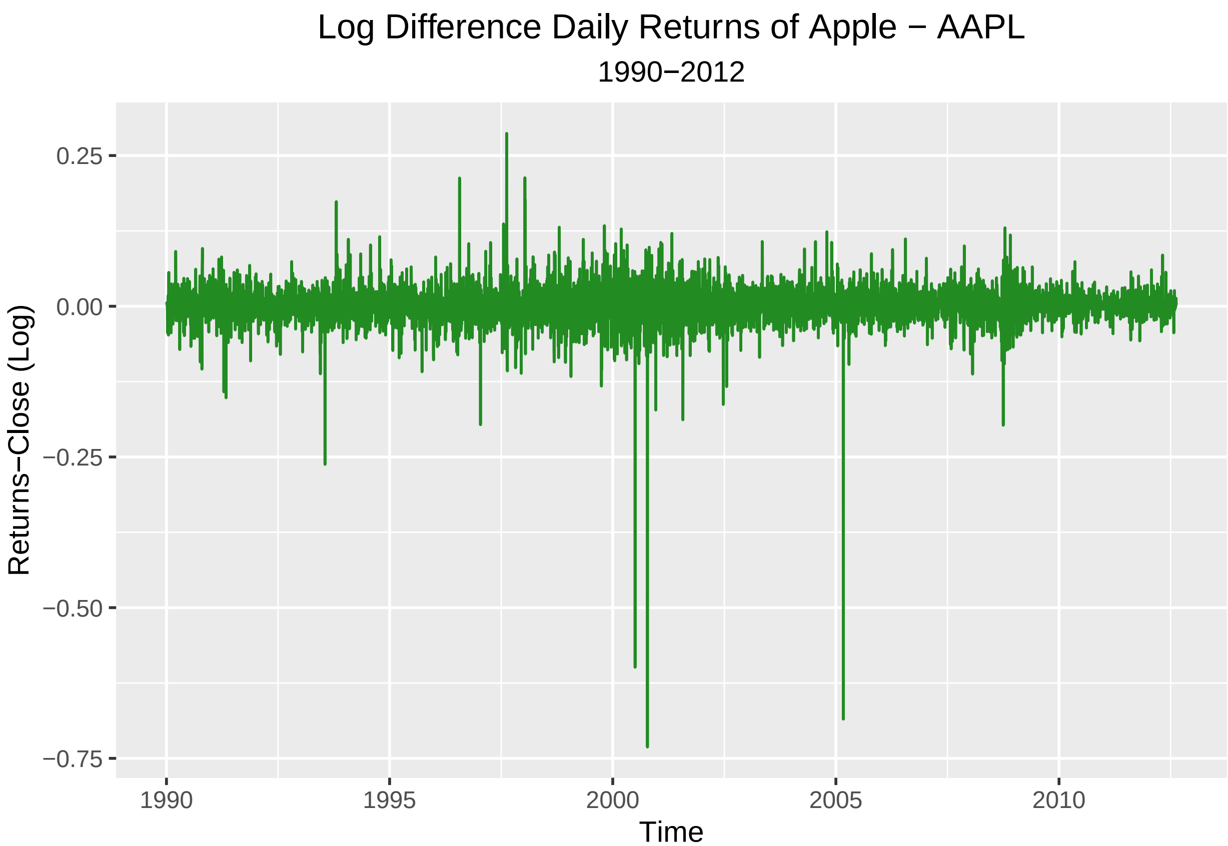
Download favorite stock (AAPL) at the daily and monthly frequencies. Compute the time series of returns and for each frequency build a density forecast.

```
AAPL_daily <- read_excel("/Users/icranfield/Desktop/Econ 144/Week 5/Homework 3/Chapter8_Exercises_Data.


AAPL_monthly <- read_excel("/Users/icranfield/Desktop/Econ 144/Week 5/Homework 3/Chapter8_Exercises_Dat

AAPL_daily = AAPL_daily %>% map_df(rev) #to reorder data
AAPL_monthly = AAPL_monthly %>% map_df(rev)
```

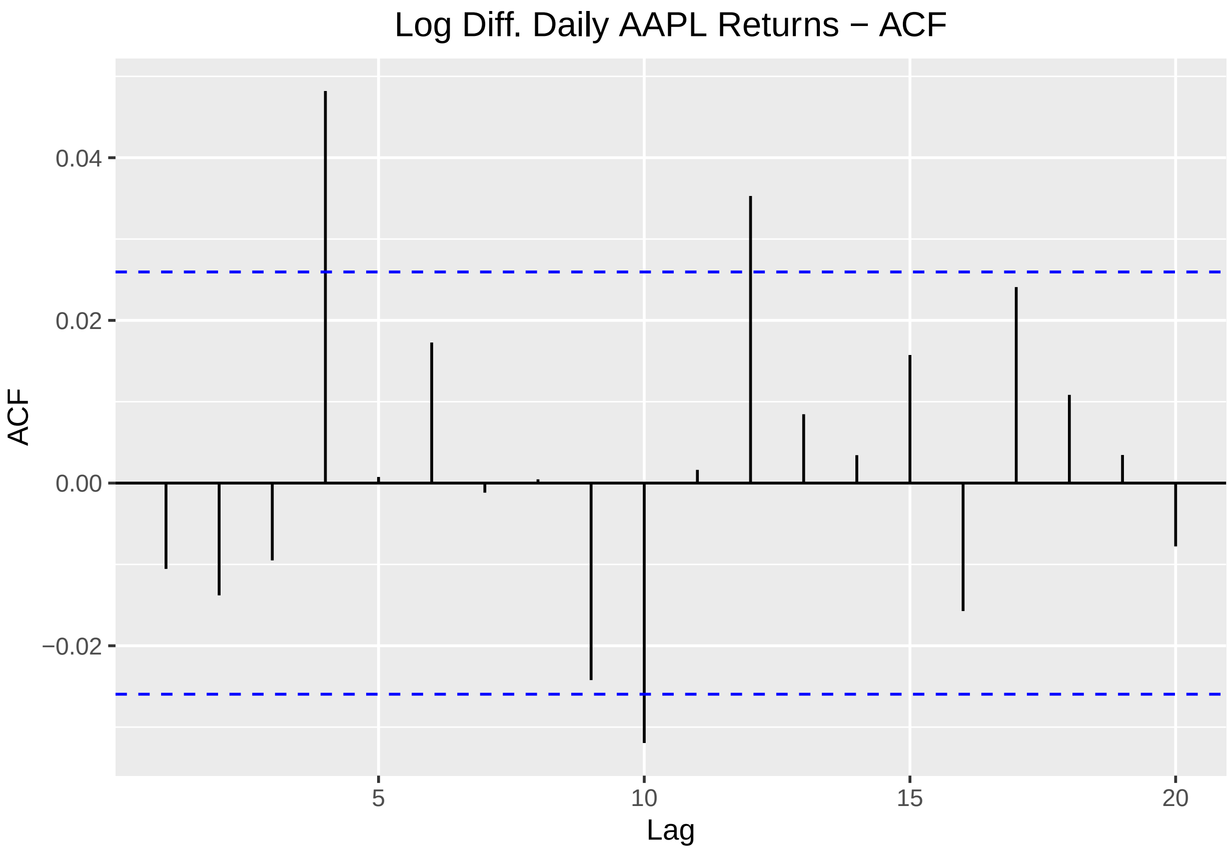
Daily returns AAPL - Time Series and ACF/PACF

```
dailyts <- ts(AAPL_daily$Close,  start = c(1990, 01), frequency = 252)

# find the percentage changes by taking log difference

daily_change <- diff(log(dailyts))

# plot ts
```

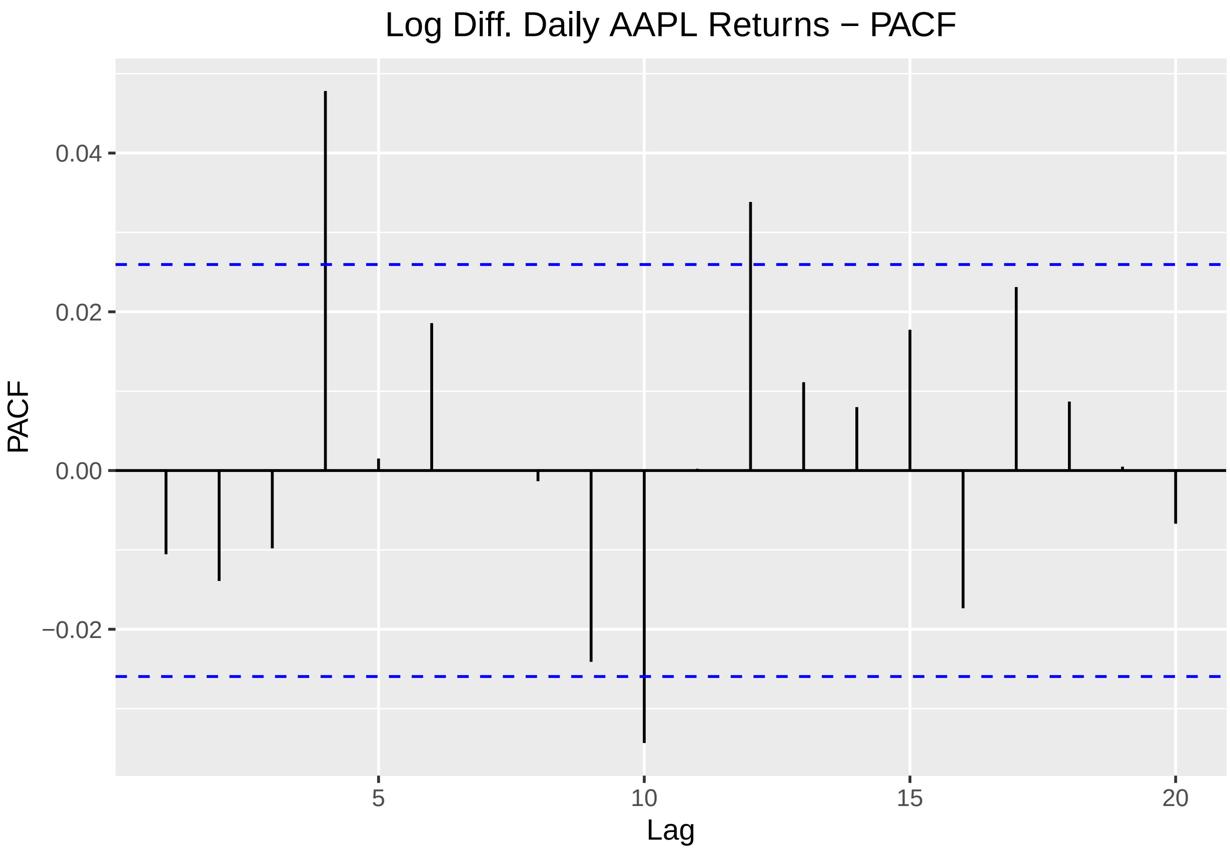
```r
#autoplot(dailyts, color = "Forest Green") + ggtitle("Daily Returns of Apple - AAPL", "1990-2012") + th

# plot the log difference
autoplot(daily_change, color = "Forest Green") + ggtitle("Log Difference Daily Returns of Apple - AAPL"
```




```r
### ACF

ggAcf(daily_change, lag.max = 20) + ggtitle("Log Diff. Daily AAPL Returns - ACF") + theme(plot.title = c
```



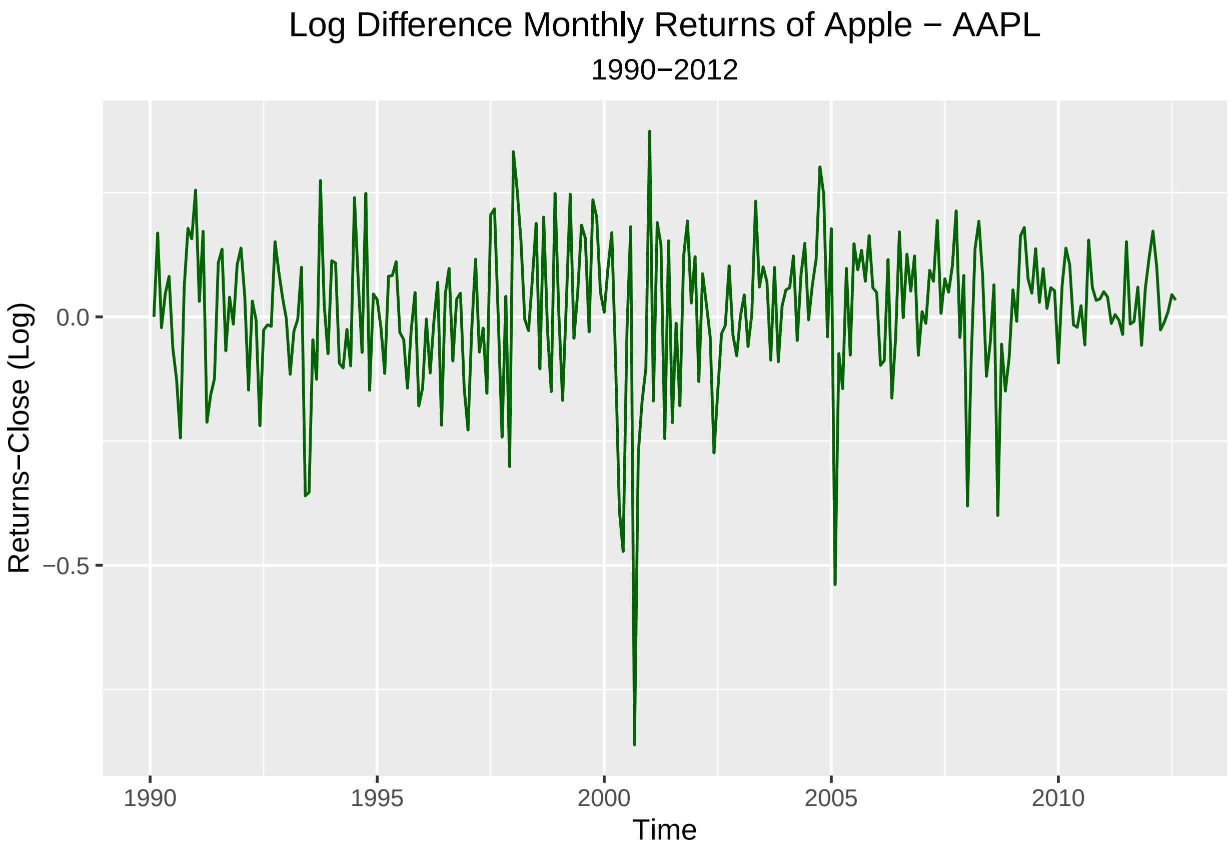

```
## PACF
ggPacf(daily_change, lag.max = 20) + ggtitle("Log Diff. Daily AAPL Returns - PACF") + theme(plot.title =
```



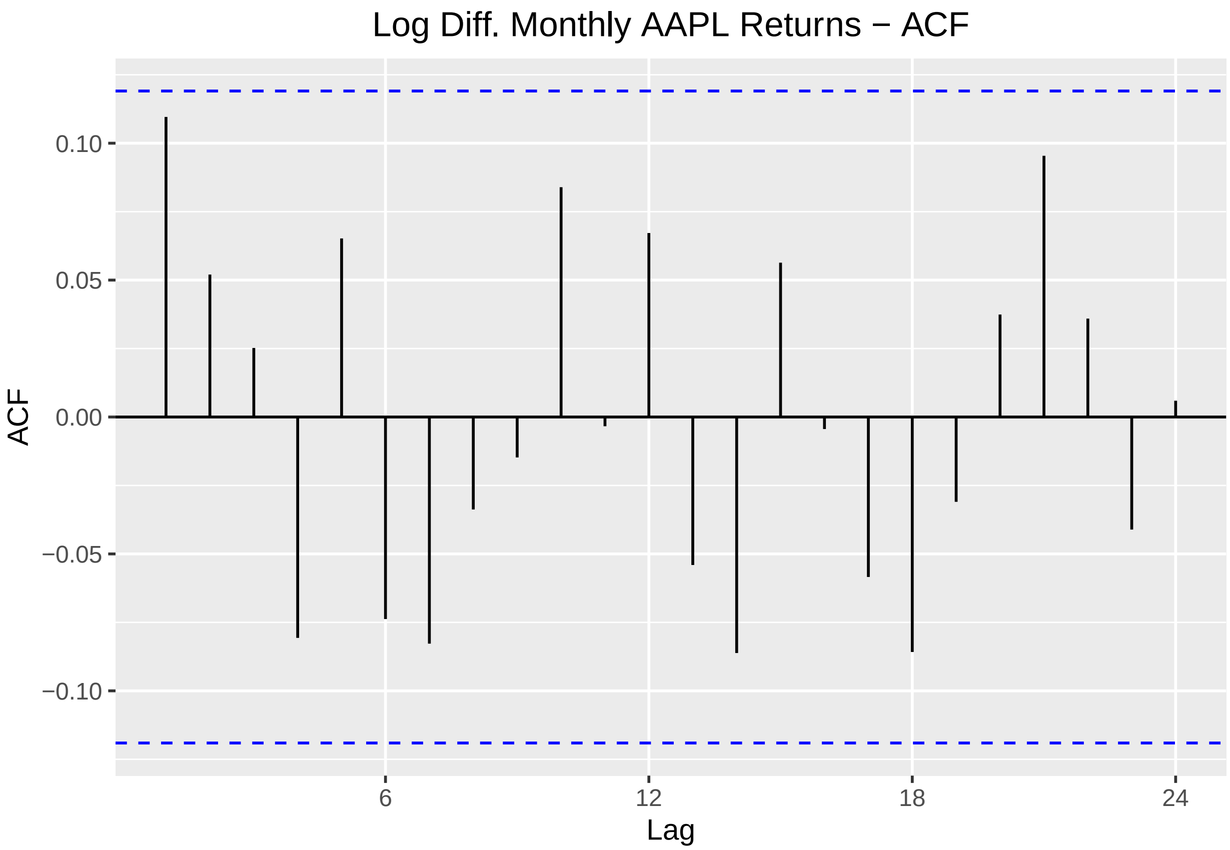
For the daily returns of AAPL, I plotted both the log difference returns to create a stationary process. It looks like it has high volatility with few but large positive and negative returns. I then took the ACF and PACF. I will focus on the log difference plots. We can also suggest that the mean is close to zero from the log difference. There are significant lags at 4, 10, and 12.

Monthly returns AAPL - Time Series and ACF/PACF

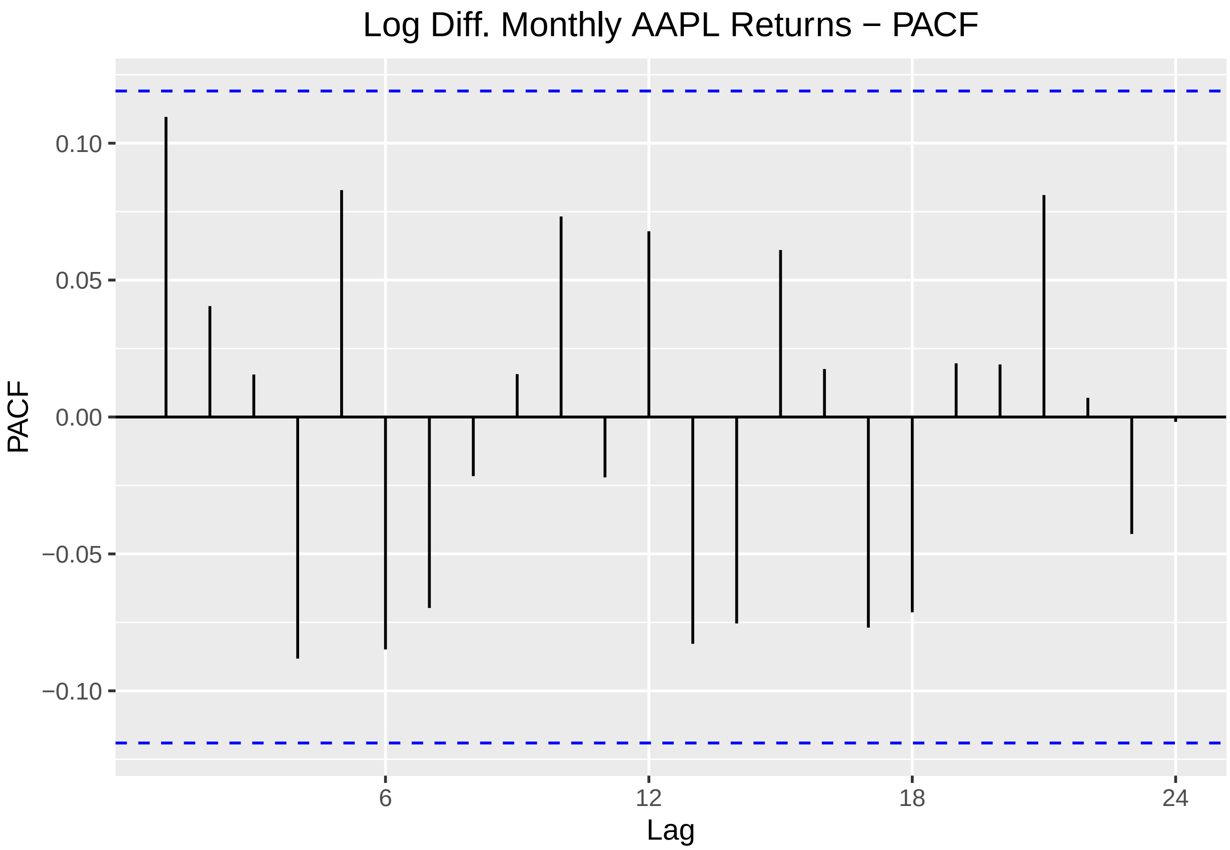
```
monthlyts <- ts(AAPL_monthly$Close,  start = c(1990, 01), frequency = 12)

monthly_change <- diff(log(monthlyts))

autoplot(monthly_change, color = "Dark Green") + ggtitle("Log Difference Monthly Returns of Apple - AAPI
```




```
### ACF

ggAcf(monthly_change, lag.max = 24) + ggtitle("Log Diff. Monthly AAPL Returns - ACF") + theme(plot.titl
```




```r
### PACF

ggPacf(monthly_change) + ggtitle("Log Diff. Monthly AAPL Returns - PACF") + theme(plot.title = element_t
```

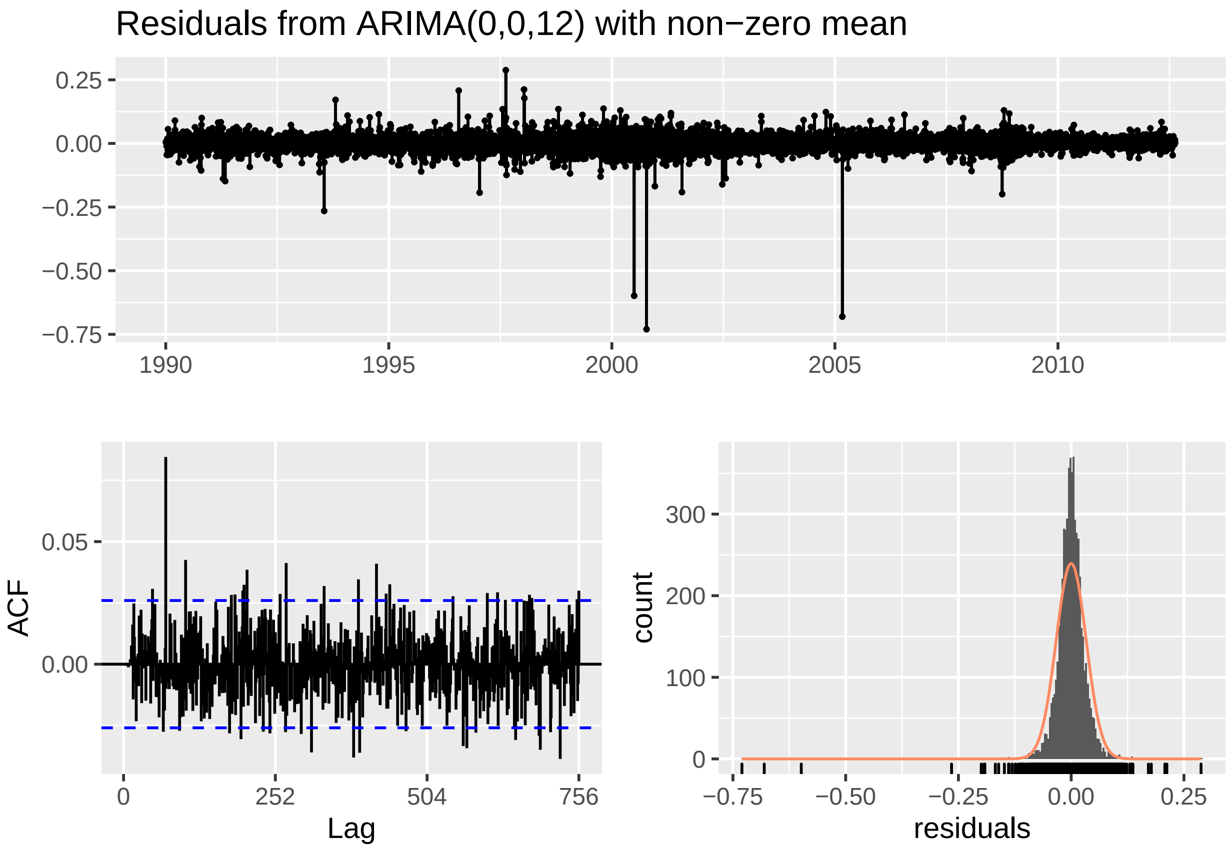


The log difference time series plot is covariance stationary of monthly returns for Apple. It looks like it has high volatility with few but large positive and negative returns. The ACF and PACF of monthly returns indicate white noise with no significant lags. Because it is white noise, providng a forecast will be difficult to do.

```r
MA12 = Arima(daily_change, order = c(0,0,12))
summary(MA12)
```

```
## Series: daily_change
## ARIMA(0,0,12) with non-zero mean
##
## Coefficients:
##           ma1      ma2      ma3     ma4     ma5     ma6      ma7     ma8
##       -0.0119  -0.0132  -0.0082  0.0509  0.0015  0.0173  -0.0013  0.0005
## s.e.   0.0132   0.0132   0.0133  0.0132  0.0133  0.0133   0.0133  0.0135
##           ma9     ma10     ma11    ma12    mean
##       -0.0235  -0.0319  -0.0017  0.0358  5e-04
## s.e.   0.0131   0.0134   0.0133  0.0132  4e-04
##
## sigma^2 estimated as 0.001114:  log likelihood=11301.92
## AIC=-22575.83   AICc=-22575.76   BIC=-22482.75
##
## Training set error measures:
##                        ME       RMSE        MAE MPE MAPE      MASE         ACF1
## Training set 2.909444e-07 0.03333861 0.02142436 NaN  Inf 0.6808821 0.0002352948
```

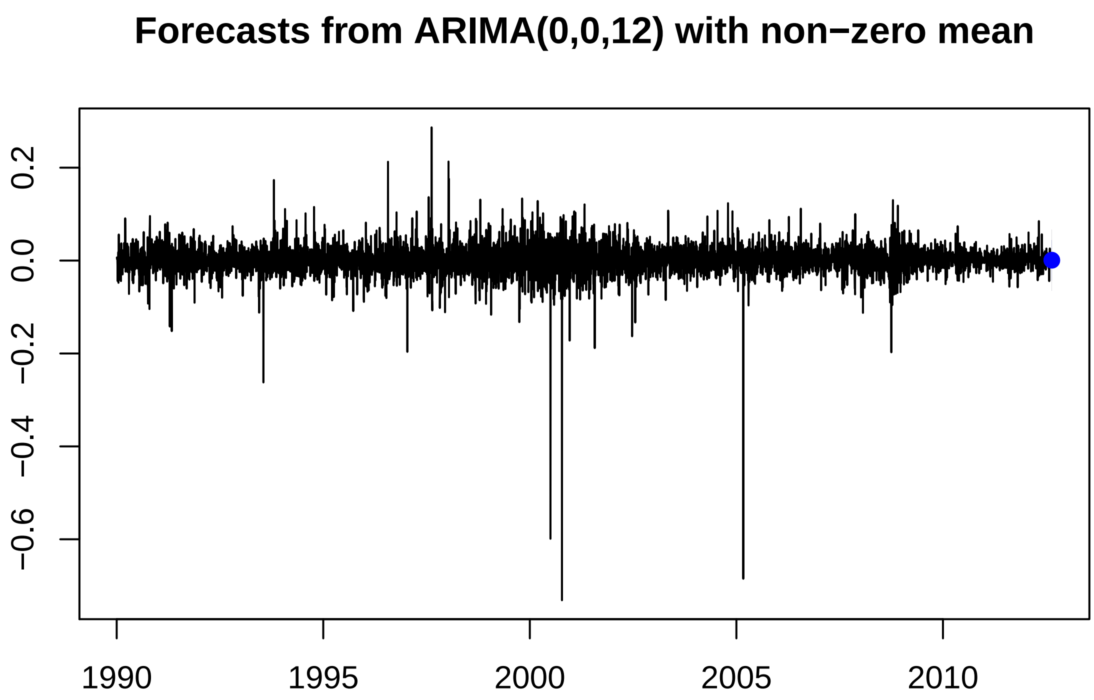
```r
checkresiduals(MA12)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,0,12) with non-zero mean
## Q* = 592.03, df = 491, p-value = 0.001155
##
## Model df: 13.   Total lags used: 504
```

Use an arima function of MA(12) to fit the model to create a density forecast. It has also cleaned it up creating a white noise process.

```r
plot(forecast(MA12,h=3))
```

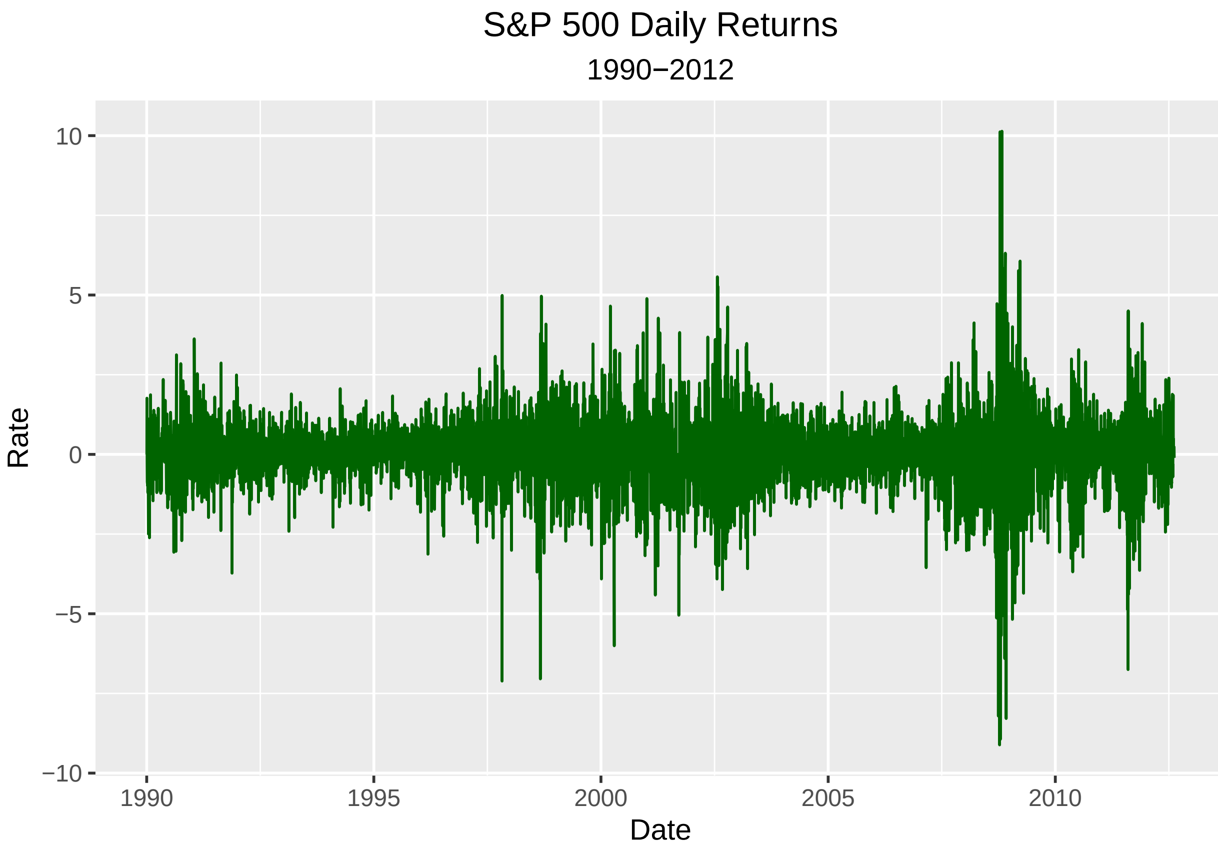


Problem 8.7:

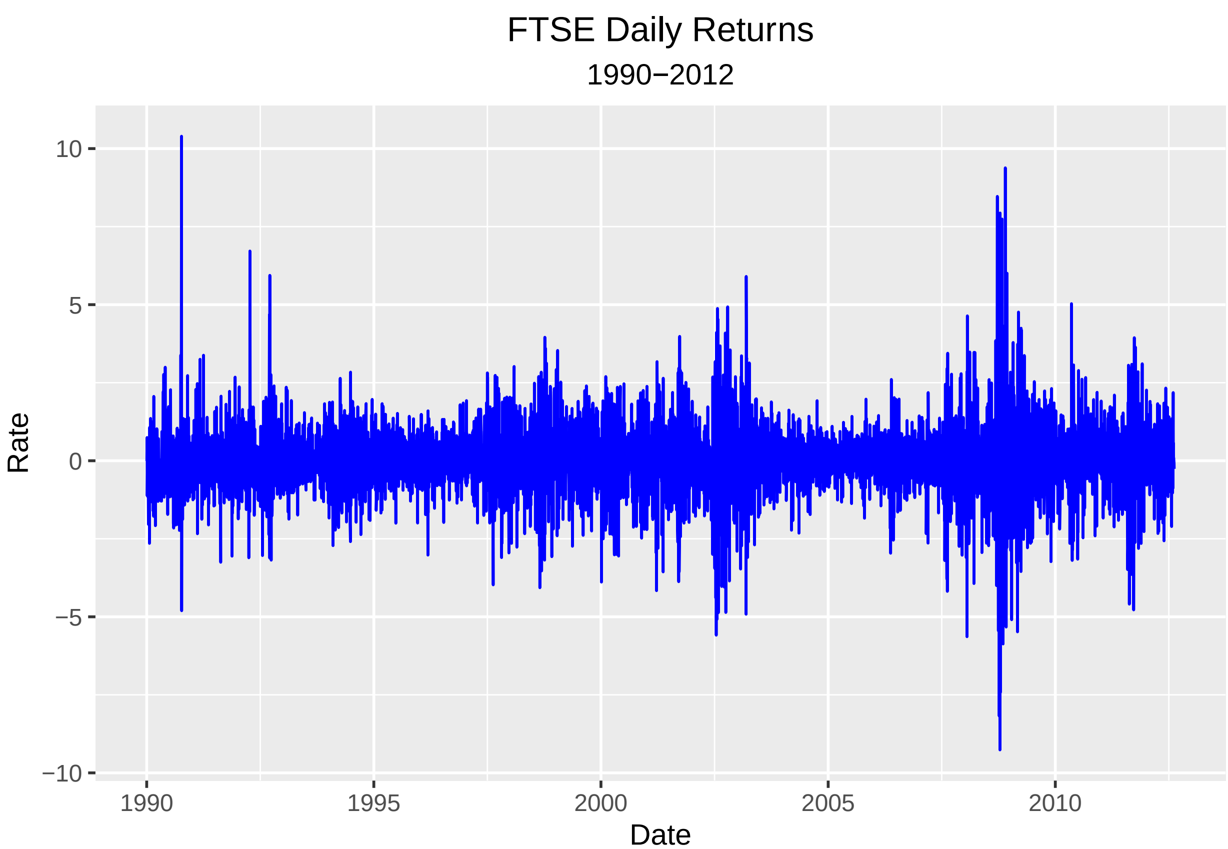
When the US stock market opens in New York, the European markets have already been in session for several hours. Does the activity in the European markets have predictive content for the US market? Download the British stock index (FTSE) and the SP500 index at the daily frequency. Examine whether FTSE returns can help forecast SP500 returns.

```
stocks <- read.xlsx("Chapter8_exercises_data.xlsx", sheet=8, detectDates=TRUE) %>%
  set_names("date", "SP", "FTSE", "SP500_Returns", "FTSE_Returns") %>%
  replace(is.na(.), 0) %>%
  mutate(date = mdy(date)) %>%
  dplyr::select(date, SP500_Returns, FTSE_Returns)
```

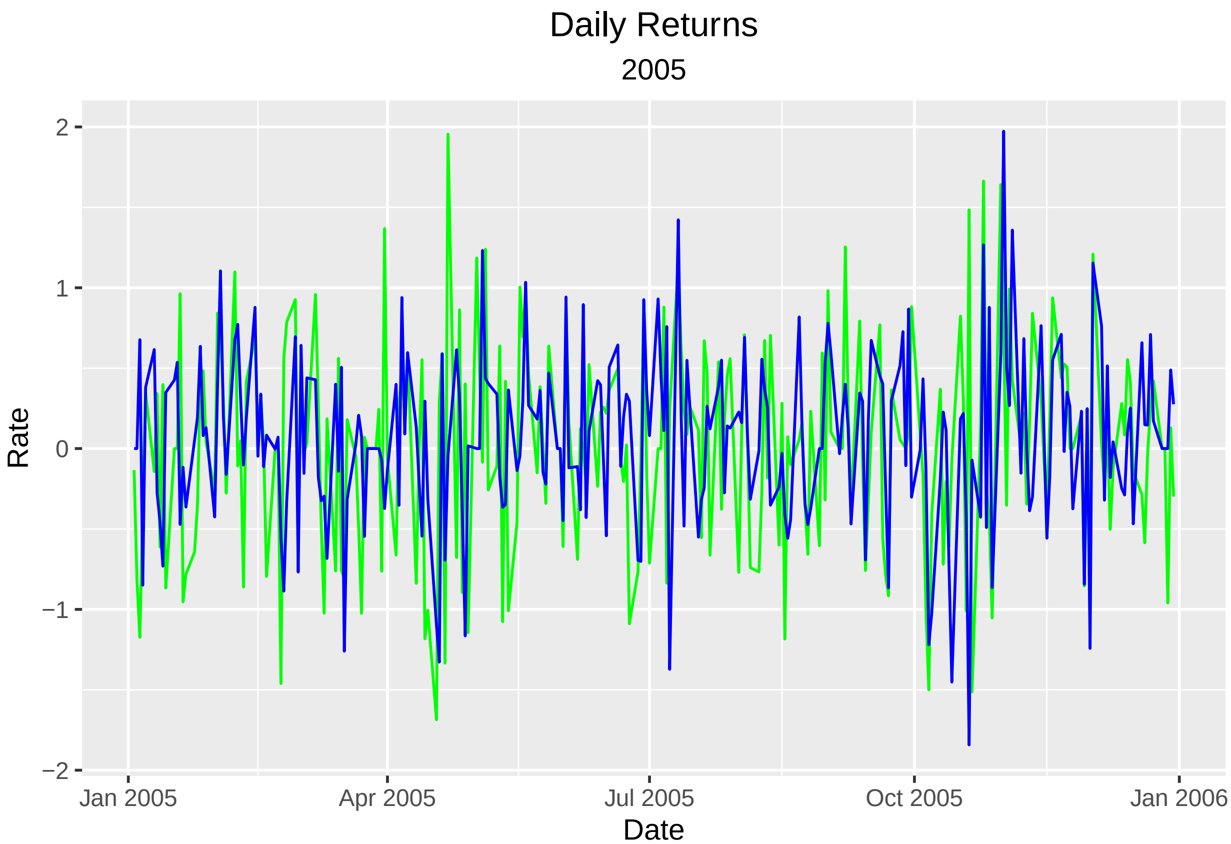
```
ggplot(stocks, aes(x=date, y=SP500_Returns)) +
  geom_line(color='dark green', na.rm=TRUE) +
  ggtitle("S&P 500 Daily Returns", "1990-2012") + theme(plot.title = element_text(hjust = 0.5), plot.su
  xlab("Date") +
  ylab("Rate")
```




```r
ggplot(stocks, aes(x=date, y=FTSE_Returns)) +
  geom_line(color='blue', na.rm=TRUE) +
  ggtitle("FTSE Daily Returns", "1990-2012") + theme(plot.title = element_text(hjust = 0.5), plot.subti
  xlab("Date") +
  ylab("Rate")
```




```
# Subset to look closer at relationship
year <-  2005

subset <- stocks %>%
  dplyr::filter(year(date) == year)

ggplot(subset, aes(x=date, group=1)) +
  geom_line(aes(y=SP500_Returns), color='green', na.rm=TRUE) +
  geom_line(aes(y=FTSE_Returns), color='blue', na.rm=TRUE) +
  ggtitle("Daily Returns", year) + theme(plot.title = element_text(hjust = 0.5), plot.subtitle = element
  xlab("Date") +
  ylab("Rate")
```

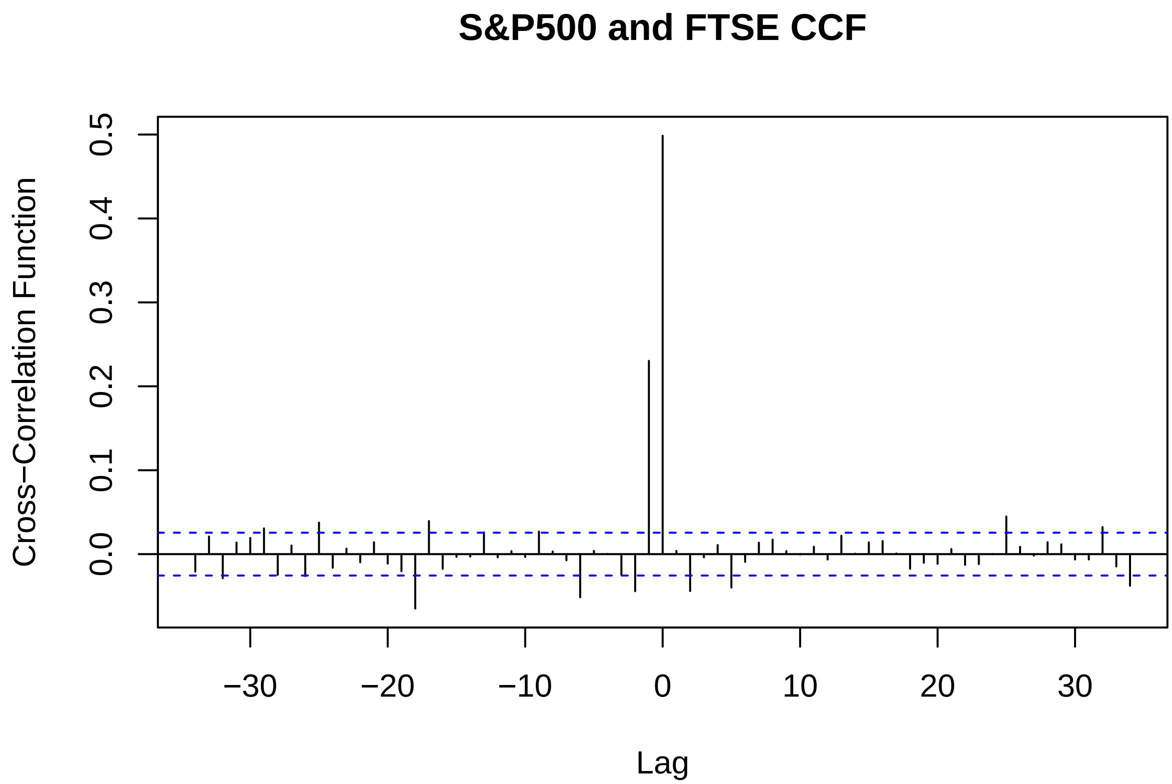


It is hard to tell if one global index affects the other by just looking at this plot.

```r
sp500 <- ts(stocks$SP500_Returns)
ftse <- ts(stocks$FTSE_Returns)

# S&P500 Correlation with FTSE
cor(sp500, ftse)
```

```
## [1] 0.4986401
```

```r
ccf(sp500, ftse, ylab="Cross-Correlation Function", main="S&P500 and FTSE CCF")
```



From the correlation coefficient we see a positive correlation between the two stock returns. And looking at the cross correlation function we see significant spikes around Lag -1,-2 and some oscillating behavior. We can further improve the model and perform analysis through granger tests and Var tests but we will learn this in Week 7 material.