SKS104\_Midterm

Skasko\_Stephen

10/11/2023

library(readxl)  
library(plotly)

## Loading required package: ggplot2

##   
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':  
##   
## last\_plot

## The following object is masked from 'package:stats':  
##   
## filter

## The following object is masked from 'package:graphics':  
##   
## layout

library(fpp)

## Loading required package: forecast

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

## Loading required package: fma

## Loading required package: expsmooth

## Loading required package: lmtest

## Loading required package: zoo

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

## Loading required package: tseries

library(astsa)

##   
## Attaching package: 'astsa'

## The following object is masked from 'package:fpp':  
##   
## oil

## The following objects are masked from 'package:fma':  
##   
## chicken, sales

## The following object is masked from 'package:forecast':  
##   
## gas

library(latex2exp)

##   
## Attaching package: 'latex2exp'

## The following object is masked from 'package:plotly':  
##   
## TeX

library(ggplot2)  
library(forecast)  
library(Ecdat)

## Loading required package: Ecfun

##   
## Attaching package: 'Ecfun'

## The following object is masked from 'package:forecast':  
##   
## BoxCox

## The following object is masked from 'package:base':  
##   
## sign

##   
## Attaching package: 'Ecdat'

## The following object is masked from 'package:datasets':  
##   
## Orange

library(TSstudio)  
library(stats)  
library(TSA)

## Registered S3 methods overwritten by 'TSA':  
## method from   
## fitted.Arima forecast  
## plot.Arima forecast

##   
## Attaching package: 'TSA'

## The following objects are masked from 'package:stats':  
##   
## acf, arima

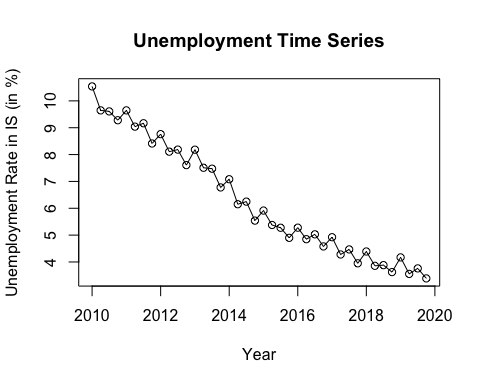
## The following object is masked from 'package:utils':  
##   
## tar

#1

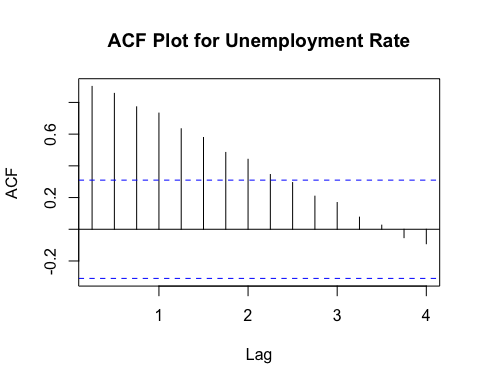
unemployment = read.csv("Unemployment.csv")  
  
# a)  
a = ts(unemployment$Unemployment.Rate, start = 2010, f = 4)  
print(a)

## Qtr1 Qtr2 Qtr3 Qtr4  
## 2010 10.539050 9.646366 9.607654 9.279040  
## 2011 9.646167 9.040166 9.166256 8.412814  
## 2012 8.761515 8.108409 8.185104 7.607361  
## 2013 8.181824 7.509379 7.473938 6.775757  
## 2014 7.082292 6.151049 6.247847 5.540235  
## 2015 5.915210 5.377158 5.274545 4.898546  
## 2016 5.277505 4.848599 5.029617 4.578183  
## 2017 4.922869 4.278584 4.467627 3.950070  
## 2018 4.390030 3.856373 3.881915 3.623181  
## 2019 4.170182 3.554909 3.760393 3.387985

# b)  
plot(a, main = "Unemployment Time Series", xlab = "Year", ylab = "Unemployment Rate in IS (in %)", type = "o")



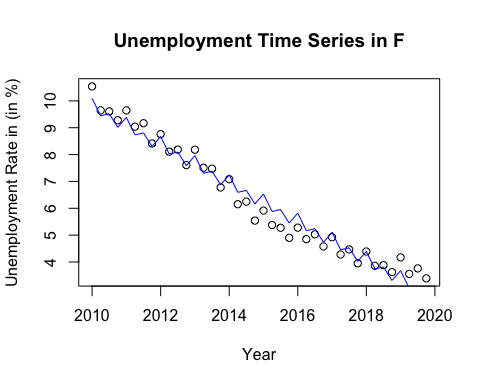
# c)   
  
# The plot displays a downward trend from 2010-2020 for unemployment rates. When looking at the plot through a quarterly lens, there tends to be some seasonality to it when you took it by the 4 seasons (Winter, Spring, Summer, Fall). There is a slight spike when hitting the Winter months then it gradually declines. Based on the plot and data we have, there may not be any cyclicity since it doesn't follow a pattern or irregularity over years and more in quarters of a year.   
  
# d)   
  
# The plot shows 4 different bars that follow quarterly for each lag. In plot 1, there seems to be similar to the previous one, meaning there is a downward trend with seasonality in the lags. With this, there tends to be consistency and not following a cyclicity pattern too.   
  
acf(a, main = "ACF Plot for Unemployment Rate")



# e)  
t = seq(1, length(time(a)), by = 1)  
s = season(a)  
  
model = lm(a ~ s + t - 1)  
summary(model)

##   
## Call:  
## lm(formula = a ~ s + t - 1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.67829 -0.18939 0.07852 0.23264 0.79252   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## s1Q 10.276841 0.157668 65.18 <2e-16 \*\*\*  
## s2Q 9.803601 0.161089 60.86 <2e-16 \*\*\*  
## s3Q 10.054317 0.164609 61.08 <2e-16 \*\*\*  
## s4Q 9.728469 0.168222 57.83 <2e-16 \*\*\*  
## t -0.178325 0.005288 -33.72 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3842 on 35 degrees of freedom  
## Multiple R-squared: 0.9971, Adjusted R-squared: 0.9967   
## F-statistic: 2393 on 5 and 35 DF, p-value: < 2.2e-16

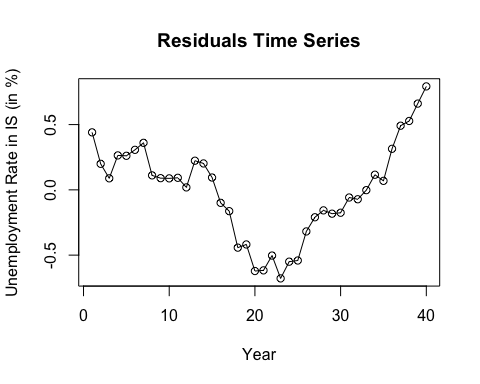
# f)   
  
# The plot has many data points in a downward trend, while the fitted line shows us how closely related the data points are. Based on the fitted model that closely relates with the data points in a consistent trend then it does imply a strong model fit for the data.   
  
plot(a,main = "Unemployment Time Series in F", xlab = "Year", ylab = "Unemployment Rate in (in %)", type = "p")  
lines(as.vector(time(a)), fitted(model), type = "l",col="blue")



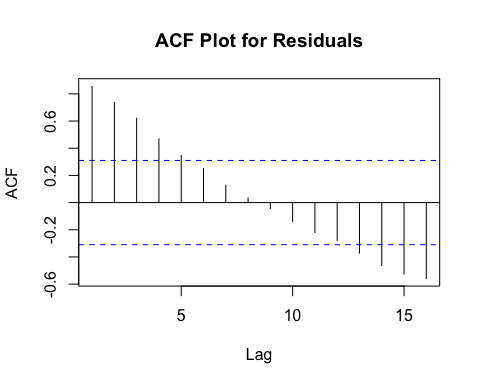
# g)   
  
# Based on the linear trend, the coefficients are in a downward constant trend, which could mean that there is a correlation between the policies implemented and years of unemployment throughout 2010-2020. Similarly, there could be a correlation between presidential candidates having good economic years like Obama from 2009-2017, then Trump from 2017-2021. Of course, this is all before COVID-19 hit the job market badly . Another way to interpret it is that overtime, the unemployment rate has decreased, potentially from the economy getting slightly better over time in the USA. With this, the coefficients are   
  
print(coef(model))

## s1Q s2Q s3Q s4Q t   
## 10.2768411 9.8036011 10.0543165 9.7284693 -0.1783251

# y = B\_0 + B\_1sqQ1 + B\_2sqQ2 + B4sqQ3 + B\_4sqQ4 + B\_5t + e   
  
# or can be shown as:  
  
# y = B\_0 + 10.27sqQ1 + 9.80sqQ2 + 10.05sqQ3 + 9.73sqQ4 - 0.18t  
  
# h)   
  
# Based on the model, the unemployment rate mostly decrease slightly when moving from the second quarter to the third quarter but it also shows the opposite with increasing slightly. Additionally, based on the model, it generally seems to increase slightly and decrease slightly by approximately 0.01 (1%) or 0.02 (2%).   
  
# i)  
  
# Based on the plot, there tends to be no seasonality since there is no trend regarding periods of seasons or quarterly. With this, there may be a trend where the percentage drops every 7 years but then there is an irregular drop and increase from 14 to 40 years. Before the irregularly, there tends to be a potential cyclicity trend that may occur. So, out of the 4, there may be trend, Cyclicity and Irregularity.   
  
r = residuals(model)  
plot(r,main = "Residuals Time Series", xlab = "Year", ylab = "Unemployment Rate in IS (in %)", type = "o")

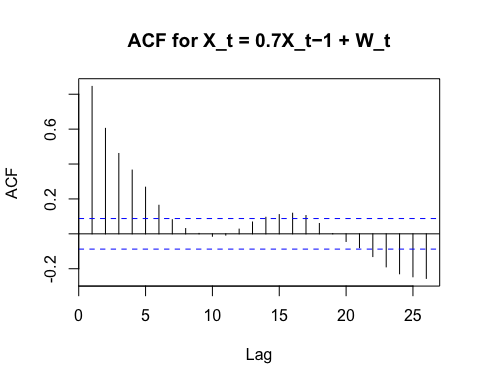


# j)  
  
# Yes, the plot does show autocorrelation by the consistency of the plot for each lag. One big indicator was lag 5 being 0.1 and lag 10 being -0.1. This could definitely indicate a cyclicity trend. Since we have a constant autocorrelation, this convey that there was an economic boom and lots of people were working in the market. Additionally, the lags can convey that employment rates correlate with previous years, meaning the policy changes or other factors done years ago.   
  
acf(r, main = "ACF Plot for Residuals")

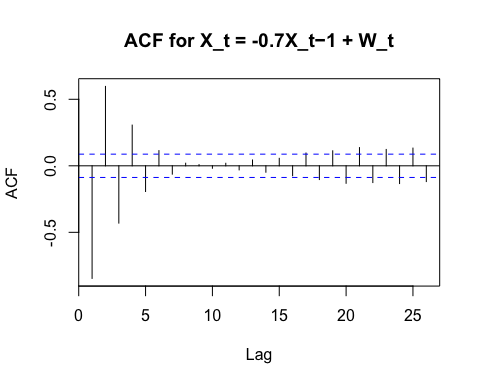


#2

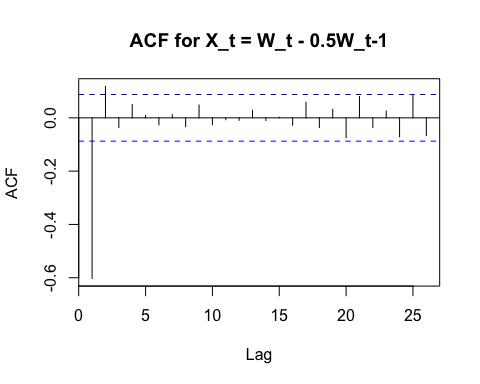
set.seed(1000)  
  
n = 502  
wt = rnorm(n, 0, sqrt(4))  
xt\_a = numeric(n)  
xt\_b = numeric(n)  
xt\_c = numeric(n)  
  
  
  
#500 observations   
for (i in 3:n) {  
 xt\_a[i] = 0.7 \* xt\_a[i-1] + wt[i]  
 xt\_b[i] = -0.7 \* xt\_b[i-1] + wt[i]  
 xt\_c[i] = wt[i] - 0.5\*wt[i-1]   
  
}  
  
  
xt\_ap = stats::filter(xt\_a, c(1, 0.7), sides = 1)  
xt\_bp = stats::filter(xt\_b, c(1, -0.7), sides = 1)  
xt\_cp = stats::filter(xt\_c, c(1, -0.5), sides = 1)  
  
  
xt\_ao = na.omit(xt\_ap)  
xt\_bo = na.omit(xt\_bp)  
xt\_co = na.omit(xt\_cp)  
  
  
acf(xt\_ao, main = "ACF for X\_t = 0.7X\_t−1 + W\_t")



acf(xt\_bo, main = "ACF for X\_t = -0.7X\_t−1 + W\_t")



acf(xt\_co, main = "ACF for X\_t = W\_t - 0.5W\_t-1")



# e)   
  
# The plots show similarities by having high drops in the first few years then shows consistently after about 10 lags. With this, the plot tends to stay around 0.1 or -0.1 after lag 10 in both plots. The key difference in the plots is one slowly decreases overtime in same direction, while the other one jumps up/down from positive to negative before getting smaller. Similarly, one graph shows more predictability compared to the other plot showing erratic jumps between positive and negative values. Lastly, it may tell us that there may be a pattern with seasonality.   
  
# f)   
  
# Main difference is not having X\_t in the formula for time and filter function uses (1, -0.5) instead of the (1, 0.7)