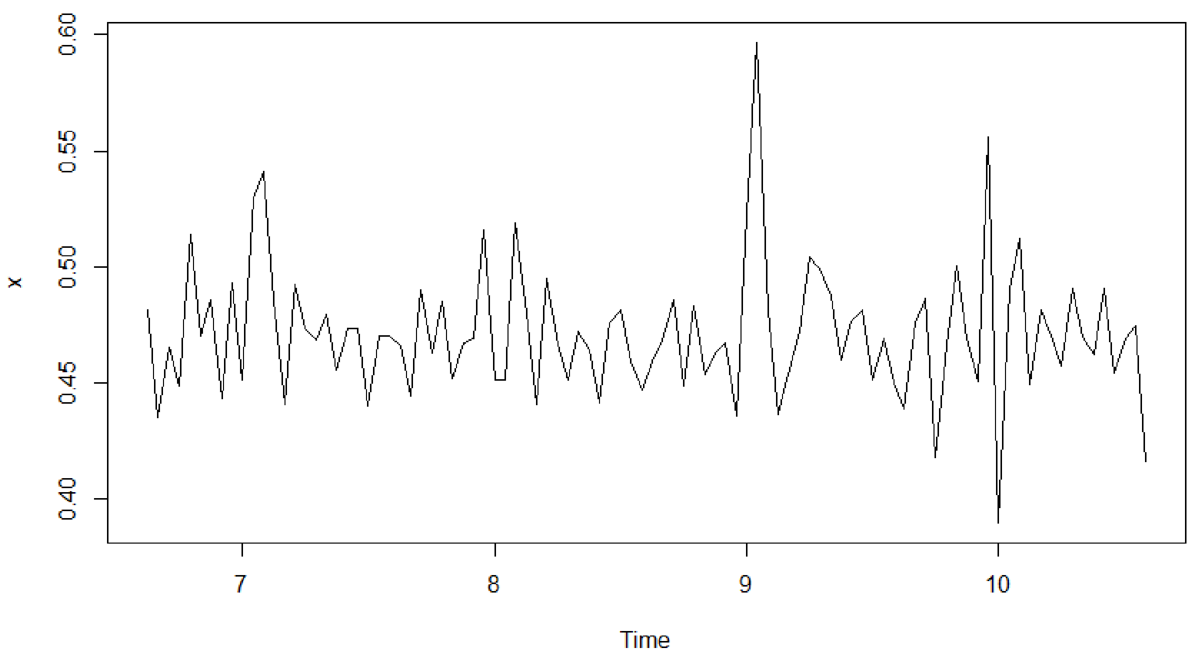
Assignment 4: Time Series Analysis (100 points)
Student Name: Abdul Rahman Gulam
Fall 2019

Purpose: To perform time series analysis to predict future customer sentiment on Twitter
Description: The objective of this study is to find the right time series model to make predictions about the future customer sentiment in Twitter network of AT&T. The data contains the average hourly sentiment of tweets customers sent to AT&T's Twitter handle.

Instructions: Please follow these steps:

1. In Canvas, navigate to Assignments and then Assignment4
2. Download and save the data set ATT_Twitter.csv
3. Read the file: data <- fread("ATT_Twitter.csv", sep=",", header=T, strip.white = T, na.strings = c("NA","NaN","","?"))
4. Use packages "forecast", "timeSeries", and "rugarch" to answer the following questions:
   4.1. **(5 points)** Paste the plot of the time series in the space below:

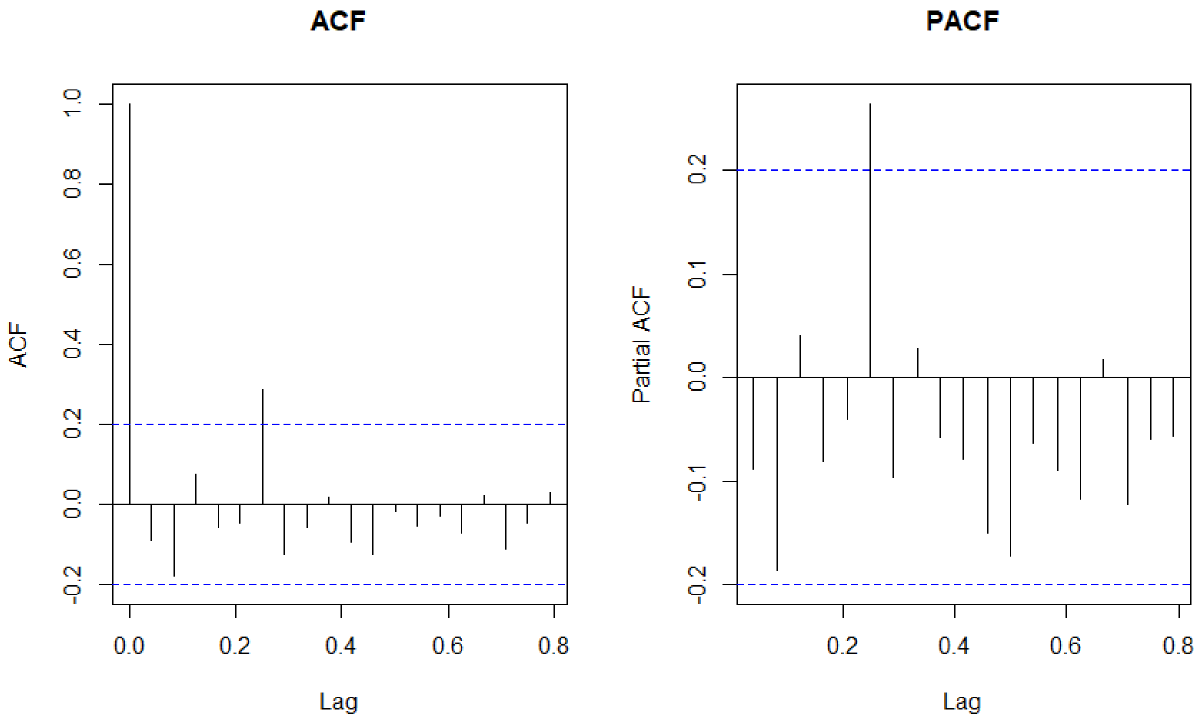


   4.2. **(10 points)** Determine if this time series is a random walk process?

I used Phillips-Perron Unit root test to determine if the time series in random walk process. Here Null Hypothesis is that the variable contains a unit root, and the alternative is that the variable was generated by a stationary process.

Since the P value is 0.01, which is less than significant level of 0.05, we reject the null hypothesis. Thus, the given timeseries is not random walk.

4.3. **(20 points)** Use ACF and PACF to determine if the times series has any MA or AR process. If so, what is (are) the order(s)? Please paste the ACF and PACF plots in the space below:



Auto Correlation Function (ACF) plot: It is used to find the Moving Average (MA) Process. Here we can see that; the correlation becomes insignificant after the second lag. Therefore we suggest that the process is MA(2).
Partial Auto Correlation Function (PACF) plot: It is used to find the Autoregressive (AR) Process. Here we can see that; the correlation becomes insignificant after the first lag. Therefore we suggest that the process is AR(1).
Order: MA(2), AR(1)

4.4. **(5 points)** Use auto.arima function to determine the best model ARIMA model for this data. What is the best order?

The Best model:ARIMA(0,0,0) with non-zero mean

```
Training set 7.170229e-17 0.02880406 0.0203803 -0.3576226 4.283708 0.
> ARIMAfit = auto.arima(x, approximation=FALSE,trace=TRUE)

 ARIMA(2,0,2)(1,0,1)[24] with non-zero mean : Inf
 ARIMA(0,0,0)            with non-zero mean : -404.5046
 ARIMA(1,0,0)(1,0,0)[24] with non-zero mean : -401.0344
 ARIMA(0,0,1)(0,0,1)[24] with non-zero mean : -401.4722
 ARIMA(0,0,0)            with zero mean     : 130.5425
 ARIMA(0,0,0)(1,0,0)[24] with non-zero mean : -402.4306
 ARIMA(0,0,0)(0,0,1)[24] with non-zero mean : -402.4105
 ARIMA(0,0,0)(1,0,1)[24] with non-zero mean : Inf
 ARIMA(1,0,0)            with non-zero mean : -403.1535
 ARIMA(0,0,1)            with non-zero mean : -403.5914
 ARIMA(1,0,1)            with non-zero mean : -402.6517

 Best model: ARIMA(0,0,0)            with non-zero mean

> plot(pca, type = "l")
```

```
> summary(ARIMAfit)
Series: x
ARIMA(0,0,0) with non-zero mean

Coefficients:
        mean
      0.4716
s.e.  0.0029

sigma^2 estimated as 0.0008384:  log likelihood=204.32
AIC=-404.63   AICc=-404.5   BIC=-399.5

Training set error measures:
                       ME       RMSE       MAE        MPE     MAPE     MASE        ACF1
Training set 7.170229e-17 0.02880406 0.0203803 -0.3576226 4.283708 0.793385 -0.08863149
> plot(pca, type = "l")
```

5. Use “rugarch” package to build ARCH and GARCH models. Use “sGARCH” in the specification.
   5.1. **(5 points)** First use armaOrder(0,0). What are the values of AIC and BIC?

```
Information Criteria
------------------------------------

Akaike       -4.1770
Bayes        -4.0702
Shibata      -4.1803
```

AIC: -4.1770 BIC:-4.0702

   5.2. **(5 points)** Now use armaOrder(0,1). What are the values of AIC and BIC?

```
Information Criteria
------------------------------------

Akaike       -4.1680
Bayes        -4.0345
Shibata      -4.1731
Hannan-Quinn -4.1141
```

AIC: -4.1680 BIC:-4.0345

6. Use “rugarch” package to build ARCH and GARCH models. Use “apARCH” in the specification.
   6.1. **(5 points)** First use armaOrder(0,0). What are the values of AIC and BIC?

```
Information Criteria
------------------------------------

Akaike       -4.1487
Bayes        -3.9884
Shibata      -4.1559
Hannan-Quinn -4.0839

Weighted Ljung-Box Test on Standardized Residuals
```

AIC: -4.1487 BIC:--3.9884

   6.2. **(5 points)** Now use armaOrder(0,1). What are the values of AIC and BIC?

```
Information Criteria
------------------------------------

Akaike       -4.2497
Bayes        -4.0627
Shibata      -4.2594
Hannan-Quinn -4.1741
```

AIC: -4.2497 BIC:-4.0627

   6.3. **(10 points)** Using scholarly articles, explain what is the main difference between apARCH (asymmetric power ARCH) and simple GARCH models?

In some financial time, series, large negative returns appear to increase volatility more than do positive returns of the same magnitude. This is called the leverage effect. Standard GARCH models cannot model the leverage effect because they model σt as a function of past values of at2- whether the past values of at are positive or negative is not taken into account. The problem here is that the square function x2 is symmetric in x. The solution is to replace the square function with a flexible class of nonnegative functions that include asymmetric functions. The

APARCH (asymmetric power ARCH) models does this. They also offer more flexibility than GARCH models by modeling σtδ, where δ > 0 is another parameter.

Source:

1. **https://faculty.washington.edu/ezivot/econ589/ch18-garch.pdf**
2.

7. **(30 points)** Based on your analysis in 5 and 6, which model specification you would choose for predicting the future values? How the values of AIC and BIC influence your decision?

		AIC	BIC
sGARCH	armaOrder(0,0)	-4.177	-4.0702
	armaOrder(0,1)	-4.168	-4.0345
apARCH	armaOrder(0,0)	-4.148	-3.9884
	armaOrder(0,1)	-4.2497	-4.0627

Models that have low AIC and BIC are better ones. ***Thus, apGARCH model with order (0,1) has a lesser AIC and BIC values and is preferred*.**

AIC and BIC are useful to evaluate the closeness of fit and number of parameters estimated. The main difference between AIC and BIC is that AIC does not penalize the number of parameters as strongly as BIC does. The only difference between AIC and BIC in practice is the size of the penalty whereas BIC penalizes model complexity more heavily.

CODE FOR ASSIGNMENT 4

```
library(forecast)
library(timeSeries)
library(rugarch)

getwd()
setwd( "C:/Users/agulammo/Documents")
data <- read.csv("ATT_Twitter.csv")
colnames(data)
nrow(data)
summary(data)
head(data,n=1)
tail(data,n=1)
str(data)
x = ts(data[,2], start=c(06,16), end=c(10,15), frequency=24)
plot(x)


PP.test(x)
par(mfrow = c(1,2))
acf(x,main='ATT_Twitter.csv')
pacf(x,main='ATT_Twitter.csv')

library(forecast)
ARIMAfit = auto.arima(z, approximation=FALSE,trace=TRUE)
summary(ARIMAfit)
pred = predict(ARIMAfit, n.ahead = 8760)
pred
pred

par(mfrow = c(1,1))
plot(x, type='l', xlim=c(6,12), ylim=c(0,1), xlab='Month', ylab='Tweet Sentiment')
lines(10^(pred$pred),col='blue')
lines(10^(pred$pred+2*pred$se),col='orange')
lines(10^(pred$pred-2*pred$se),col='orange')



library(quantmod)
library(lattice)
library(timeSeries)
library(rugarch)
library(parallel)
```

```
spec1=ugarchspec(variance.model=list(model="sGARCH"),
          mean.model=list(armaOrder=c(0,0)))
fit1=ugarchfit(data=x,spec=spec1)
show(fit1)

spec2=ugarchspec(variance.model=list(model="sGARCH"),
          mean.model=list(armaOrder=c(0,1)))
fit2=ugarchfit(data=x,spec=spec2)
show(fit2)

spec3=ugarchspec(variance.model=list(model="apARCH"),
          mean.model=list(armaOrder=c(0,0)))
fit3=ugarchfit(data=x,spec=spec3, solver='hybrid')
show(fit3)

spec4=ugarchspec(variance.model=list(model="apARCH"),
          mean.model=list(armaOrder=c(0,1)))
fit4=ugarchfit(data=x,spec=spec4, solver='hybrid')
show(fit4)

fit1_pred <- ugarchforecast(fit1, data = x, n.ahead = 10)
head(sigma(fit1_pred))
head(fitted(fit1_pred))
plot(fit1_pred,which=1)
```