Assignment #3

*Submit your* ***printed*** *solution at the start of class on Tuesday, October 15.*

Short answers, please. *Explain your answer concisely*. *If you refer to a plot in your answer, include that plot as part of your answer.*  Do not include extraneous plots that you do not refer to in your narrative. “Significance” implies statistical significance. Presume necessary conditions for inference hold unless the question addresses these.

The data for the first five questions is in the oddly-named file “MRTSSM4453USN.csv” (in the Canvas data folder). This filename is given by FRED to total monthly retail sales of beer, wine, and liquor stores (in millions of dollars). (Google that if you’re curious.) Read the data from this file and define time series as follows. We will truncate the series at the end of 2023 for convenience working with STL. [You need to have installed the lubridate R package for this assignment.]

library(lubridate)  
 Data <- read.csv("[ your path ]MRTSSM4453USN.csv")  
 sales <- ts(Data$MRTSSM4453USN, start=1992,end=2023+11/12,frequency=12)  
 dates <- lubridate::ymd(Data$DATE[1:length(sales)])

1. Explain why one should use a multiplicative decomposition rather than an additive decomposition for the sales time series. Include the diagnostic plot of a median polish of the sales time series in your answer.
2. Use the STL function in R to perform a decomposition of log(sales).   
    sales.stl <- stl(log(sales), s.window=9)  
   Using this decomposition, define a seasonally adjusted version of the *sales* time series (actual sales, not on a log scale). Show a sequence plot of your seasonally adjusted time series in a plot with the original raw time series (on the same axes in different colors, with a plot legend identifying which is which).
3. Based on the estimated trend from STL, how did Covid affect the long-term growth of sales of these products? [ This analysis is most straightforward on the log scale. ]
4. Does the seasonal pattern change over time? In particular, is the seasonal pattern identified by STL different pre-Covid (2018-2019) and post-Covid (say, 2022-2023).
5. Does the remainder from the STL decomposition contain further seasonal information? Use the following commands to create time series that count the number of days and the number of weekend days in a month.

count\_weekdays <- function(from, to) {   
 sum(!wday(seq(from, to, "days")) %in% c(1,7)) }

next.date <- c(dates[-1], ymd("2024-01-01"))-1  
 wdays <- mapply(count\_weekdays, dates, next.date)  
 ndays <- lubridate::days\_in\_month(dates)  
Use these variables in a regression model (a choice for the model follows) to determine whether STL has left seasonal information in the remainder. Covid produces an artifact in the remainder time series, so use data only through 2018.   
 remain <- window(sales.stl$time.series[,'remainder'], end=2018.99)  
 n <- length(remain)  
 wdays <- wdays[1:n]  
 ndays <- ndays[1:n]  
 month <- factor(rep(month.abb, n/12), levels=month.abb)

regr <- dynlm(remain ~ month + ndays + wdays)

The file “a3\_series.csv” (in the Canvas assignments folder) defines three time series that are used in questions 6, 7, and 8. All have length *n* = 150. Define these time series as follows.

Data <- read.csv("[your path]a3\_series.csv")  
ts.6 <- ts(Data[,2])  
ts.7 <- ts(Data[,3])  
ts.8 <- ts(Data[,4])

1. Identify an ARMA model for the time series ts.6.   
   (a) Identify a model from a visual inspection of the estimated ACF and PACF.  
   (b) Following the procedure described in class that uses the function `fit\_models`  
    (Lecture 12), use AIC and BIC to identify models for this time series.   
   (c) Do the chosen models in parts (a) and (b) agree with each other? If the results   
    disagree, what model would you recommend? Why?
2. Repeat the analysis of Q6, but with the time series ts.7.
3. Repeat the analysis of Q6, but with the time series ts.8.
4. Use the R program arima.sim to generate a Gaussian realization of length *n*=800 of the AR(2) process defined by  
    Xt = 1.6 Xt-1 – 0.80 Xt-2 + wt  
   Set the random seed to 54 (i.e., set.seed(54)) and the white noise variance to 1.  
   (a) Confirm that the ACF and PACF for your simulated time series match those of an   
    AR(2) process and that the estimated coefficients when fitting this model are   
    close to the parameters of the process (fit the model using sarima):  
    sarima(xt, p=2, d=0, q=0)  
   (b) Reset the seed to 62 (set.seed(62)), then generate an independent sequence of   
    standard Gaussian white noise of the same length as your simulated time series.   
    Add this white noise to your realization of Xt.   
    yt <- xt + rnorm(length(xt))

What ARMA process describes the resulting time series?

1. Textbook exercise 4.3, parts (a) and (b).