**Problem Statement:**

Quarterly beer sales data has been provided in the [beer.csvView in a new window](https://olympus.greatlearning.in/courses/4744/files/408672/download?wrap=1)files.

Part A)

Using the Holt-Winters method, model the data and predict the sales for the next 2 years. Your submission should contain the complete modeling steps with explanations. Include pictures and R-code where applicable.

Part B)

Using the ARIMA method, model the data and predict the sales for the next 2 years. Your submissions should contain the complete modeling steps with explanations. Include pictures and R-code where applicable.

**Dataset**: [beer.csvView in a new window](https://olympus.greatlearning.in/courses/4744/files/408673/download?wrap=1)

There are 72 observations and one variable names OzBeer. It is quarterly data. So, 72 observations mean 72/4=18 years data. (Assume the most recent 18 years.) OzBeer variable signifies the sales revenue in a thousand dollars.

**Group Members**

1. Akshay Surve
2. Aditya Godse
3. Amay Saxena

**Answer**

**1] Exploratory Data Analysis**

setwd("C:/Users/DELL/Desktop/Akshay/Group Assignments/Group Assignment 6 TSF")

getwd()

beer <- read.csv("C:/Users/DELL/Desktop/Akshay/Group Assignments/Group Assignment 6 TSF/beer.csv", sep="")

summary(beer)

OzBeer

Min. :212.8

1st Qu.:272.6

Median :317.5

Mean :329.9

3rd Qu.:379.7

Max. :525.0

str(beer)

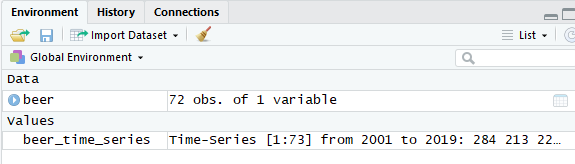
'data.frame': 72 obs. of 1 variable:

$ OzBeer: num 284 213 227 308 262 ...

Since as mentioned in the problem statement that it is quarterly data with 72 observations Total of 18 years of data is available. Assuming these 18 years to be recent i.e. considering starting year 2001 Converting dataset into Time Series in order to predict outcomes.

beer\_time\_series <- ts(beer, start = 2001, frequency = 4, end = 2019 )

frequency = 4 since data is for quarters

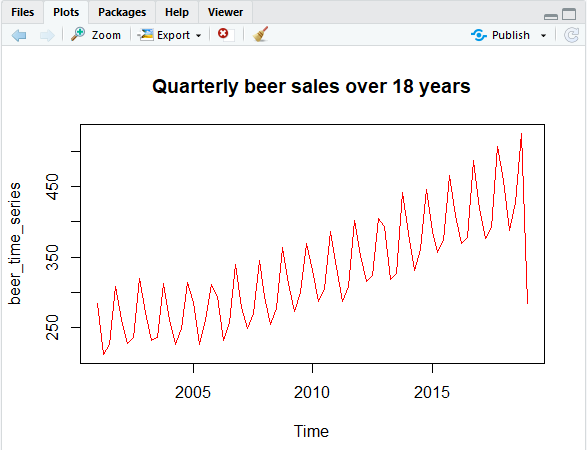


str(beer\_time\_series)

Time-Series [1:73] from 2001 to 2019: 284 213 227 308 262 ...

**Plotting beer time series**

plot(beer\_time\_series, main = "Quarterly beer sales over 18 years", col = "red")



From plot it can be seen that seasonality of beer sales is almost consistent over a period of 18 years. Inter quarter/year fluctuations are almost similar across the years but there is seasonality in the data. There is slightly increasing trend in the time series which goes on increasing Year over year also there are no outliers in the Time series. There is consistency in the time series in terms of pattern. Overall upward positive trend shows increase in sales with time. There is seasonality in the data which shows Regular pattern and increases slowly and constantly. There is variation in the data, as pattern changes from quarter to quarter.

cycle(beer\_time\_series)

Qtr1 Qtr2 Qtr3 Qtr4

2001 1 2 3 4

2002 1 2 3 4

2003 1 2 3 4

2004 1 2 3 4

2005 1 2 3 4

2006 1 2 3 4

2007 1 2 3 4

2008 1 2 3 4

2009 1 2 3 4

2010 1 2 3 4

2011 1 2 3 4

2012 1 2 3 4

2013 1 2 3 4

2014 1 2 3 4

2015 1 2 3 4

2016 1 2 3 4

2017 1 2 3 4

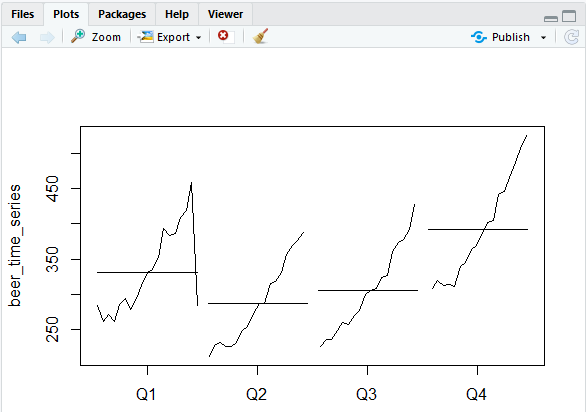
2018 1 2 3 4

2019 1

**Subseries plot**

This is required for checking the seasonality in the time series. It takes seasonality component out and plot data for every season of the series. Also checks behavior of data for each quarter.

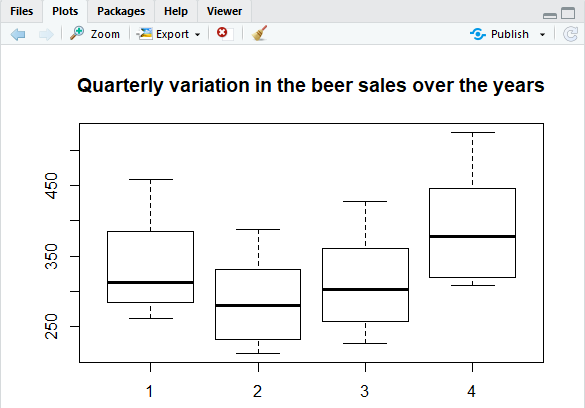
monthplot(beer\_time\_series)



From seasonality plot it can be seen that for quarter 2, 3 and 4 seasonality is almost similar also the beer sales were low at the start of every quarter. Beer sales drops at the end of quarter 1.

boxplot(beer\_time\_series~cycle(beer\_time\_series),

main = "Quarterly variation in the beer sales over the years")



Boxplot shows that at the end of each year there are higher sales. Middle quarters shows lower sales compared to quarter 1 and quarter 4. Further quarters according to sales can be arranged in descending order.

They are: Quarter 4 > Quarter 1 > Quarter 3 > Quarter 2.

**Following libraries are going to be used for further visualization:**

library(tseries)

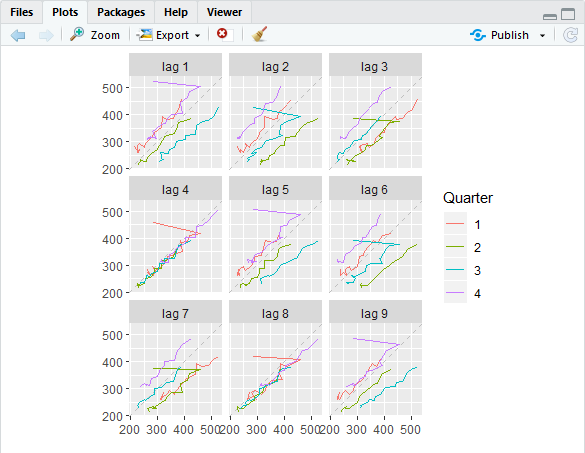
library(ggplot2)

library(fpp2)

library(TSA)

**Plotting a lag plot**

gglagplot(beer\_time\_series)



From lag plot it can be seen that,

Lag plot 1, lag plot 5 and lag plot 9 shows similar pattern.

Lag plot 4 and lag plot 8 shows similar pattern.

Lag plot 3 and lag plot 7 shows similar pattern.

Lag plot 2 and lag plot 6 exhibit similar pattern.

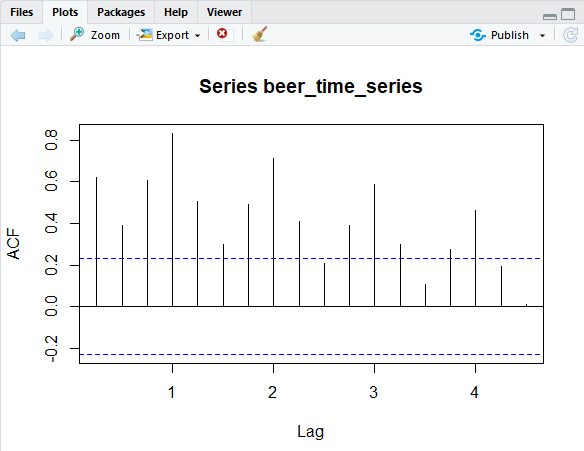
This further proves the presence of seasonality in the time series

**Auto Correlation**

Lags are very useful in time series analysis as calculated above, because of a phenomenon called autocorrelation, which is a tendency for the values within a time series to be correlated with previous copies of itself.

* Auto correlation function

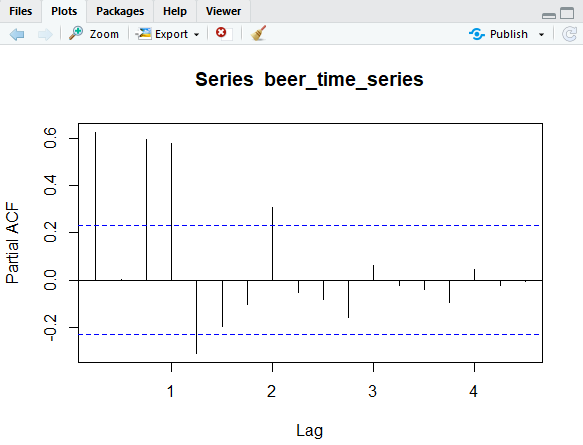
acf(beer\_time\_series)



Form above graph it can be seen that,

The values within a time series are correlated with previous copies of themselves and it decreases with time, also with advancing time the effect of past observations decreases. The non-seasonality in time series reflects an AR model.

* Partial Autocorrelation Function



Auto correlation outside purple band indicate significant dependency of series on a series with lag 2. It also indicates that after lag 2, the autocorrelation decreases but it progresses to zero after it.

Also it helps us identify that 2 post period have to be included in auto regression model.

There seems presence of both AR(1) and MA(2) model and even presence of some error.

**Checking If Model Is Stationary or Non Stationary**

Only Stationary model can be forecasted in the ARIMA model. So if our model isn't stationary we have to make it stationary.

Augmented dickey fuller test can identify whether model is stationary or not.

library(tseries)

adf.test(beer\_time\_series)

Augmented Dickey-Fuller Test

data: beer\_time\_series

Dickey-Fuller = -2.2708, Lag order = 4, p-value = 0.4651

alternative hypothesis: stationary

P value in this case is much higher than our considered level of significance which is 0.05

H0: time series is non stationary

H1: time series is stationary

The p-value is greater than 0.05. We cannot reject the null hypothesis.

**DECTECTING Variance of Stability and Seasonality**

var\_beer <- diff(beer\_time\_series,lag=4)

for checking if data is stationary

adf.test(var\_beer)

Augmented Dickey-Fuller Test

data: var\_beer

Dickey-Fuller = -3.1977, Lag order = 4, p-value = 0.09544

alternative hypothesis: stationary

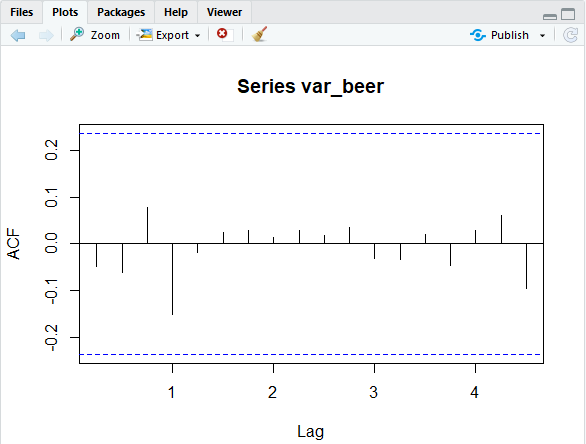
BoxCox.lambda(var\_beer)

[1] 1.154986

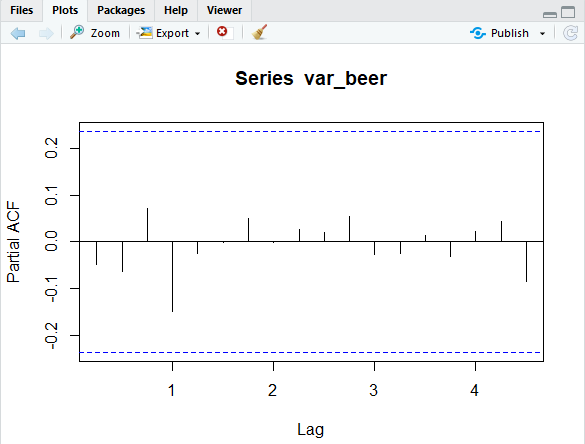
The “optimal value” in this case, which gives best approximation of a normal distribution curve is 1.154986.

**Checking Autocorrelation function and partial Autocorrelation function again on var\_beer**.

acf(var\_beer)



pacf(var\_beer)



From these correlation plot it can be seen that correlation of time series with previous copies of themselves is within purple line indicating independency of series on a series with lag 2.

**Analysing Trend and Seasonality**

Considering moving average for constructing trend in beer sales.

beer\_trend <- ma(beer\_time\_series,order = 4)

beer\_trend

Qtr1 Qtr2 Qtr3 Qtr4

2001 NA NA 255.3250 254.4125

2002 257.4500 260.1000 262.8375 264.6875

2003 265.4125 264.6500 262.4625 260.4000

2004 261.2625 262.9875 266.1875 269.2375

2005 270.5125 271.4625 272.1750 274.0125

2006 274.3750 277.4500 278.9750 279.1750

2007 282.9000 285.2875 287.9375 290.3875

2008 291.9625 295.1375 299.8000 304.5125

2009 309.6000 313.1875 316.1250 320.1750

2010 322.7750 325.5750 328.2000 328.7750

2011 329.1000 331.4250 335.6500 341.3625

2012 346.9500 349.3375 354.6750 360.0500

2013 360.6625 365.6125 369.0625 369.4125

2014 375.3000 380.0500 380.9375 384.5750

2015 389.3000 393.3625 398.7750 403.2250

2016 405.4250 408.6500 412.4500 414.5125

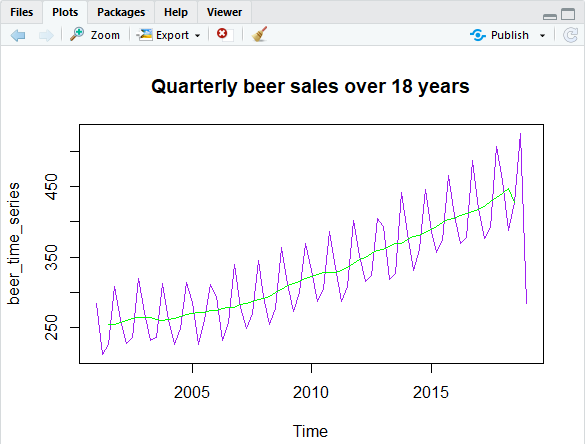
2017 417.1500 421.3125 428.6000 434.8375

2018 440.4375 447.0625 427.6750 NA

2019 NA

plot(beer\_time\_series, main = "Quarterly beer sales over 18 years", col = "purple")

lines(beer\_trend ,col = "green")



In above plot, Purple line indicate actual time series while green line indicate trend in beer sales.

* Now, in order to stabilize the time series, remove trend from the series. To remove trend from the original time series it must be subtracted.

beer\_wo\_trend <- beer\_time\_series - beer\_trend

beer\_wo\_trend

Qtr1 Qtr2 Qtr3 Qtr4

2001 NA NA -28.4250 53.9875

2002 4.5500 -32.2000 -26.7375 55.7125

2003 6.4875 -31.8500 -25.4625 53.0000

2004 0.1375 -36.1875 -16.2875 45.0625

2005 15.5875 -44.9625 -11.7750 37.3875

2006 20.3250 -44.8500 -21.7750 60.0250

2007 -3.8000 -35.4875 -18.1375 55.3125

2008 1.8375 -40.4375 -22.3000 58.8875

2009 3.8000 -40.3875 -16.0250 49.3250

2010 8.0250 -37.7750 -22.3000 57.3250

2011 6.1000 -43.4250 -27.3500 60.9375

2012 5.8500 -33.2375 -29.7750 44.7500

2013 32.3375 -46.7125 -42.0625 72.8875

2014 7.8000 -48.4500 -19.5375 61.3250

2015 -2.7000 -36.1625 -25.1750 62.9750

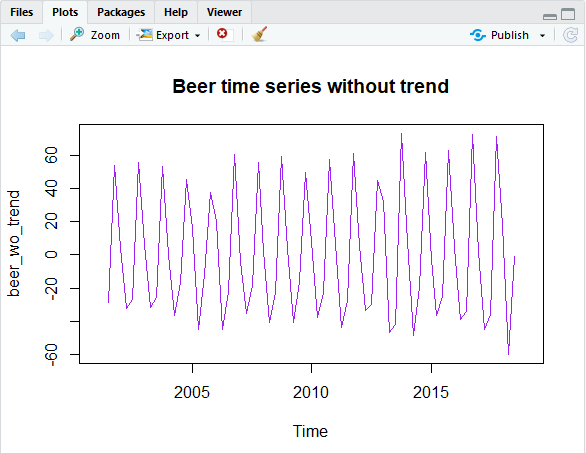
2016 4.1750 -38.8500 -33.8500 72.4875

2017 2.0500 -44.6125 -35.8000 71.2625

2018 17.9625 -59.6625 -0.7750 NA

2019 NA

plot(beer\_wo\_trend, col = "purple", main = "Beer time series without trend")



After removing trend from the rime series, seasonality appears to have consistency in it.

For years up to 2005 there is little change in seasonality while for middle years seasonality appears to be constant. And for recent years it further shows slight movement.

* Now creating matrix for beer sales without trend,

beer\_matrix <- t(matrix(data = beer\_wo\_trend, nrow = 4))

beer\_matrix

[,1] [,2] [,3] [,4]

[1,] NA NA -28.4250 53.9875

[2,] 4.5500 -32.2000 -26.7375 55.7125

[3,] 6.4875 -31.8500 -25.4625 53.0000

[4,] 0.1375 -36.1875 -16.2875 45.0625

[5,] 15.5875 -44.9625 -11.7750 37.3875

[6,] 20.3250 -44.8500 -21.7750 60.0250

[7,] -3.8000 -35.4875 -18.1375 55.3125

[8,] 1.8375 -40.4375 -22.3000 58.8875

[9,] 3.8000 -40.3875 -16.0250 49.3250

[10,] 8.0250 -37.7750 -22.3000 57.3250

[11,] 6.1000 -43.4250 -27.3500 60.9375

[12,] 5.8500 -33.2375 -29.7750 44.7500

[13,] 32.3375 -46.7125 -42.0625 72.8875

[14,] 7.8000 -48.4500 -19.5375 61.3250

[15,] -2.7000 -36.1625 -25.1750 62.9750

[16,] 4.1750 -38.8500 -33.8500 72.4875

[17,] 2.0500 -44.6125 -35.8000 71.2625

[18,] 17.9625 -59.6625 -0.7750 NA

[19,] NA NA NA -28.4250

* seasonality in beer consumption

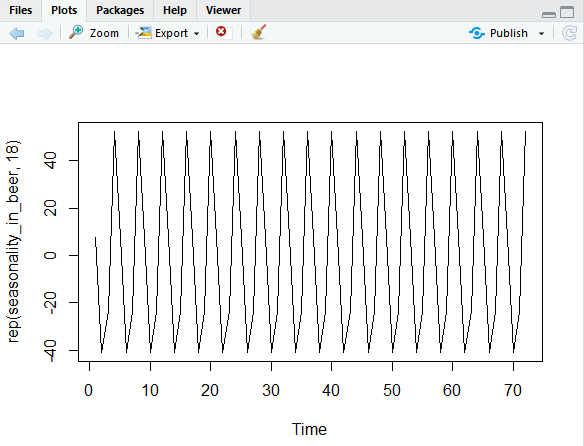
seasonality\_in\_beer= colMeans(beer\_matrix, na.rm = TRUE)

seasonality\_in\_beer

[1] 7.677941 -40.897059 -23.530556 52.456944

Plotting seasonality,

plot.ts(rep(seasonality\_in\_beer,18))



In this case from the seasonality plot, it can be said that seasonality is constant over time.

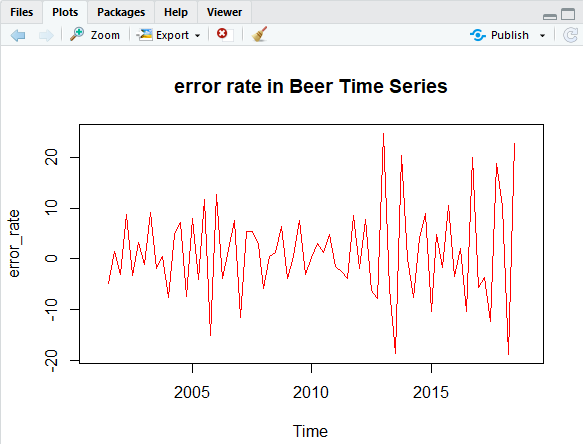
Indicating presence of additive seasonality.

* now calculating the white noise in beer time series,

error\_rate <- beer\_wo\_trend - seasonality\_in\_beer

plot(error\_rate, col = "red", main = "error rate in Beer Time Series")

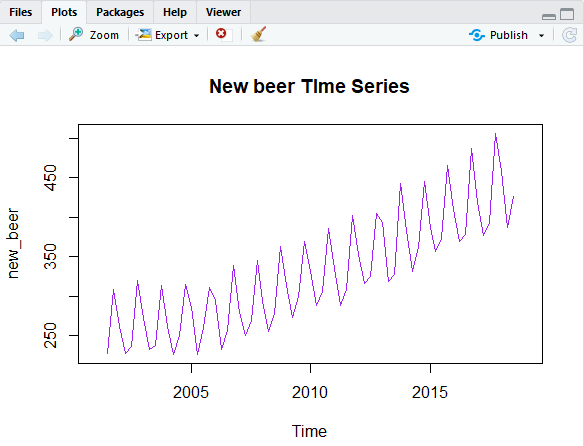
White Noise here indicates, stationary time series, i.e. series with no autocorrelation.



Now using the calculated trend, seasonality and error rate form a new time series

new\_beer <- beer\_trend + seasonality\_in\_beer + error\_rate

plot(new\_beer, col = "purple", main = "New beer TIme Series")



Time series = Trend + Seasonality + Error Rate.

**Decomposing beer time series**

It is not possible to estimate the movement of the trend because trend movement is impacted by the seasonality.

beer\_Dec <- stl(beer\_time\_series,s.window = 'p')

beer\_Dec

stl(x = beer\_time\_series, s.window = "p")

Components

seasonal trend remainder

2001 Q1 3.628637 269.7765 10.9949070

2001 Q2 -42.037953 263.1656 -8.3276141

2001 Q3 -23.275960 257.5891 -7.4131096

2001 Q4 61.685313 253.7271 -7.0124218

2002 Q1 3.628637 257.6288 0.7425673

2002 Q2 -42.037953 260.8179 9.1200147

2002 Q3 -23.275960 262.7039 -3.3279424

2002 Q4 61.685313 264.0706 -5.3559157

2003 Q1 3.628637 265.8148 2.4566050

2003 Q2 -42.037953 265.4941 9.3438805

2003 Q3 -23.275960 262.3639 -2.0879047

2003 Q4 61.685313 259.4894 -7.7747033

2004 Q1 3.628637 260.8477 -3.0763602

2004 Q2 -42.037953 263.7764 5.0616027

2004 Q3 -23.275960 266.1809 6.9950399

2004 Q4 61.685313 268.7576 -16.1428645

2005 Q1 3.628637 270.4347 12.0366566

2005 Q2 -42.037953 272.5149 -3.9769324

2005 Q3 -23.275960 271.7070 11.9689113

2005 Q4 61.685313 273.4022 -23.6874862

2006 Q1 3.628637 274.4548 16.6165866

2006 Q2 -42.037953 278.1422 -3.5042032

2006 Q3 -23.275960 278.7793 1.6966096

2006 Q4 61.685313 278.8953 -1.3806513

2007 Q1 3.628637 282.4536 -6.9821874

2007 Q2 -42.037953 285.7433 6.0946153

2007 Q3 -23.275960 288.3834 4.6926011

2007 Q4 61.685313 289.9721 -5.9574510

2008 Q1 3.628637 291.6281 -1.4567584

2008 Q2 -42.037953 295.3146 1.4233727

2008 Q3 -23.275960 299.8204 0.9556094

2008 Q4 61.685313 304.3547 -2.6400536

2009 Q1 3.628637 309.4659 0.3054306

2009 Q2 -42.037953 313.7122 1.1257213

2009 Q3 -23.275960 316.2158 7.1601534

2009 Q4 61.685313 319.6734 -11.8586954

2010 Q1 3.628637 322.7131 4.4582560

2010 Q2 -42.037953 326.2384 3.5995300

2010 Q3 -23.275960 328.1641 1.0118959

2010 Q4 61.685313 328.5755 -4.1608507

2011 Q1 3.628637 329.1042 2.4672015

2011 Q2 -42.037953 331.3210 -1.2830936

2011 Q3 -23.275960 335.2591 -3.6831006

2011 Q4 61.685313 341.1861 -0.5714432

2012 Q1 3.628637 347.4164 1.7549649

2012 Q2 -42.037953 350.0285 8.1094057

2012 Q3 -23.275960 353.6538 -5.4778214

2012 Q4 61.685313 359.5583 -16.4436086

2013 Q1 3.628637 362.1885 27.1828490

2013 Q2 -42.037953 365.6798 -4.7418086

2013 Q3 -23.275960 367.6207 -17.3447447

2013 Q4 61.685313 369.8764 10.7383331

2014 Q1 3.628637 375.8685 3.6028181

2014 Q2 -42.037953 379.7199 -6.0819162

2014 Q3 -23.275960 381.0364 3.6395609

2014 Q4 61.685313 384.4591 -0.2443773

2015 Q1 3.628637 389.0224 -6.0510812

2015 Q2 -42.037953 393.4867 5.7512378

2015 Q3 -23.275960 398.9588 -2.0828377

2015 Q4 61.685313 403.1344 1.3802394

2016 Q1 3.628637 405.7822 0.1891218

2016 Q2 -42.037953 408.3955 3.4424929

2016 Q3 -23.275960 412.2194 -10.3434754

2016 Q4 61.685313 414.8008 10.5138477

2017 Q1 3.628637 417.5615 -1.9901578

2017 Q2 -42.037953 420.4707 -1.7327787

2017 Q3 -23.275960 427.8452 -11.7692466

2017 Q4 61.685313 435.7368 8.6779039

2018 Q1 3.628637 441.2896 13.4818032

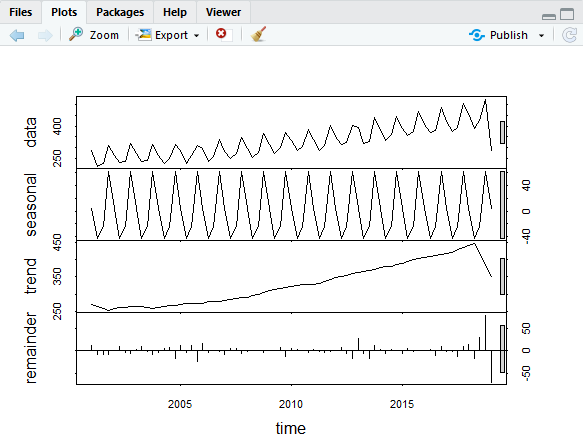
2018 Q2 -42.037953 446.1380 -16.7000929

2018 Q3 -23.275960 420.3173 29.8587069

2018 Q4 61.685313 385.9874 77.3273096

2019 Q1 3.628637 349.5828 -68.8114435

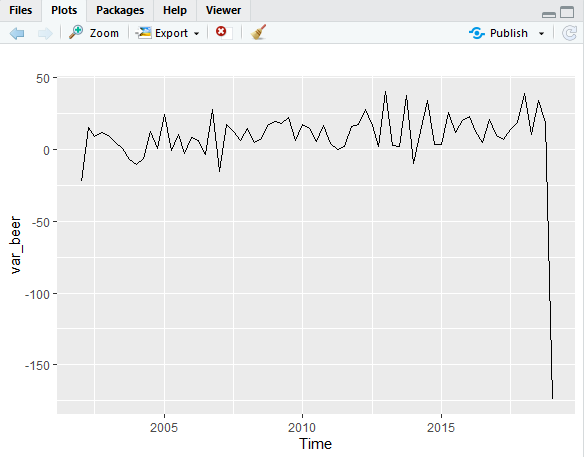
plot(beer\_Dec)



from plot it can be seen that, scale for trend is bigger, indicating volatility in the trend as compared to constant seasonality with time trend goes on increasing error rates can be calculated in recent years, because as projecting in future the error decreases with older time series data.

**2]Holt Winters Method**

autoplot(var\_beer)



Plot exhibits both trend and seasonality with upward movement, also the model variation in the model is consistent year over year there is consistency in beer sales.

Since both seasonality and trend are present, holt-winter model can be applied here.

holt\_winter <- hw(beer\_time\_series,seasonal = "additive")

summary(holt\_winter)

Forecast method: Holt-Winters' additive method

Model Information:

Holt-Winters' additive method

Call:

hw(y = beer\_time\_series, seasonal = "additive")

Smoothing parameters:

alpha = 0.0436

beta = 0.0234

gamma = 1e-04

Initial states:

l = 258.441

b = 1.0453

s = 57.5941 -23.3313 -37.9635 3.7006

sigma: 24.1468

AIC AICc BIC

787.6165 790.4736 808.2306

Error measures:

ME RMSE MAE MPE MAPE MASE ACF1

Training set -0.1894768 22.78531 11.43097 -0.2476881 3.649031 0.7204395 -0.07767902

Forecasts:

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

2019 Q2 404.5985 373.6531 435.5439 357.2716 451.9254

2019 Q3 419.9567 388.9419 450.9714 372.5237 467.3896

2019 Q4 501.6079 470.4674 532.7484 453.9827 549.2331

2020 Q1 448.4274 417.0888 479.7660 400.4992 496.3557

2020 Q2 407.4879 375.8632 439.1126 359.1222 455.8536

2020 Q3 422.8461 390.8343 454.8579 373.8882 471.8039

2020 Q4 504.4973 471.9844 537.0102 454.7731 554.2215

2021 Q1 451.3168 418.1783 484.4553 400.6358 501.9978

From summary it can be seen that slope values are quite lower. Therefore, change in the slope will be gradual.

**Predicting for next two year**

library(forecast)

forecast(holt\_winter,h = 8)

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

2019 Q2 404.5985 373.6531 435.5439 357.2716 451.9254

2019 Q3 419.9567 388.9419 450.9714 372.5237 467.3896

2019 Q4 501.6079 470.4674 532.7484 453.9827 549.2331

2020 Q1 448.4274 417.0888 479.7660 400.4992 496.3557

2020 Q2 407.4879 375.8632 439.1126 359.1222 455.8536

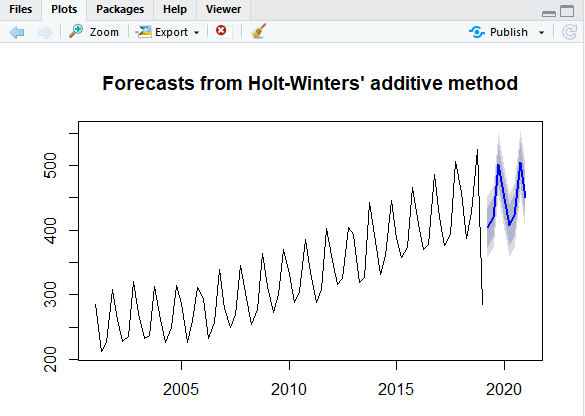
2020 Q3 422.8461 390.8343 454.8579 373.8882 471.8039

2020 Q4 504.4973 471.9844 537.0102 454.7731 554.2215

2021 Q1 451.3168 418.1783 484.4553 400.6358 501.9978

since forecast is for next 2 years, considering 8 quarters

plot(forecast(holt\_winter,h = 8))

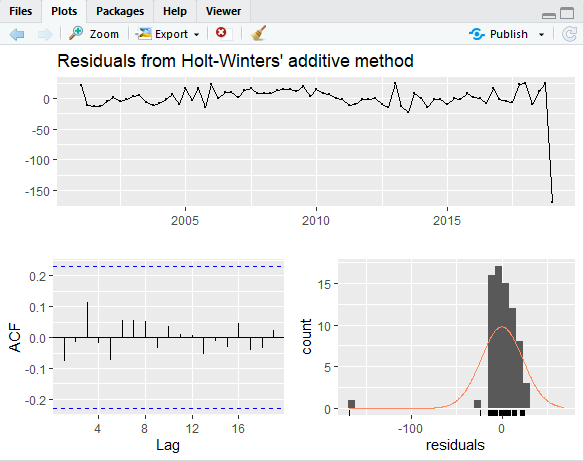


It can be seen that, beer sales will be slightly lower in the next two years compared to the previous year’s sales of around 550 thousand dollars.

Dark blue colored shaded area indicate 80% confidence interval.

Light blue colored shaded area indicate 95% confidence interval.

checkresiduals(holt\_winter)



P value greater than significance level indicate that, there will be no effect of past and present data on the future forecast.

**3]ARIMA Model**

* Beer sale data doesn't requires differentiating of order, hence d = 0
* Beer data has AR(1) model indicating p = 1
* It contain MA terms, hence q = 2
* In notation, ARIMA(1,0,2)

arima\_model = arima((beer\_time\_series), c(1,0, 2),seasonal = list(order = c(0, 1, 2), period = 4, include.constant=FALSE))

arima\_model

Call:

arima(x = (beer\_time\_series), order = c(1, 0, 2), seasonal = list(order = c(0,

1, 2), period = 4, include.constant = FALSE))

Coefficients:

ar1 ma1 ma2 sma1 sma2

0.9421 -1.1161 0.4079 -1.1885 0.8452

s.e. 0.0643 0.2827 0.1893 0.1808 0.2796

sigma^2 estimated as 442.5: log likelihood = -313.91, aic = 637.82

**AIC for the model**

AIC(arima\_model)

[1] 639.8218

**BIC for the model**

BIC(arima\_model)

[1] 653.2264

Lesser the value of AIC, more preferable is the ARIMA model. therefore, as our model is having least value of AIC, ARIMA(1,0,2)x(0,1,2)[4] is highly preferable.

**Automated ARIMA model**

auto.arima model itself takes appropriate value of (p,d,q).

Also it helps in identifying the model with low value of AIC.

ar\_model <- auto.arima(beer\_time\_series,stepwise = FALSE)

ar\_model

Series: beer\_time\_series

ARIMA(1,0,2)(0,1,2)[4]

Coefficients:

ar1 ma1 ma2 sma1 sma2

0.9421 -1.1161 0.4079 -1.1885 0.8452

s.e. 0.0643 0.2827 0.1893 0.1808 0.2796

sigma^2 estimated as 477.1: log likelihood=-313.91

AIC=639.82 AICc=641.18 BIC=653.23

forecast(ar\_model,h=8)

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

2019 Q2 443.7922 415.6131 471.9714 400.6959 486.8885

2019 Q3 395.0421 366.4495 423.6348 351.3135 438.7708

2019 Q4 506.9835 477.5767 536.3904 462.0096 551.9574

2020 Q1 456.4573 426.3544 486.5602 410.4189 502.4958

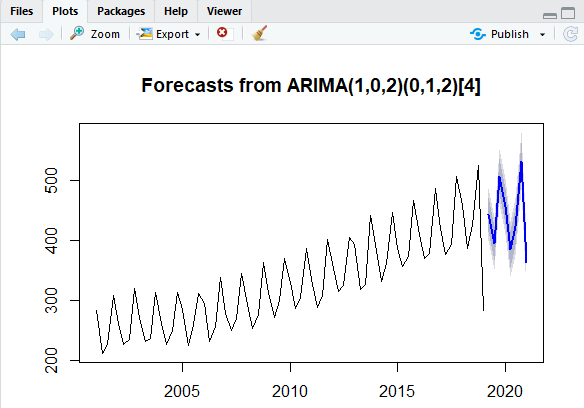
2020 Q2 385.8492 355.7326 415.9657 339.7899 431.9085

2020 Q3 422.7934 391.9582 453.6286 375.6350 469.9517

2020 Q4 531.8264 500.7090 562.9438 484.2365 579.4164

2021 Q1 363.5498 332.2223 394.8772 315.6385 411.4610

plot(forecast(ar\_model,h=8))



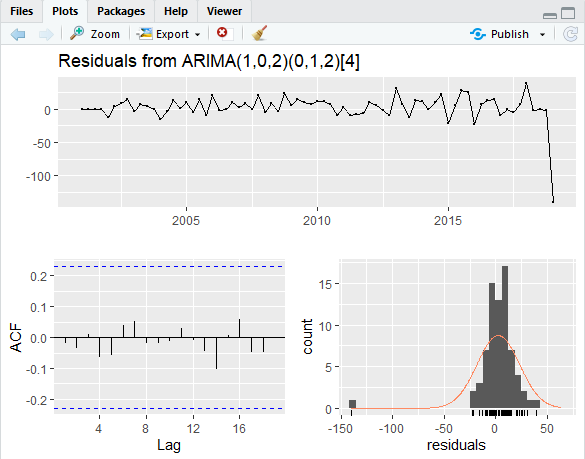
As compared to winter Holt’s model, ARIMA model gives slight better forecast.

In case of winter Holt’s model forecast plot, for two consecutive forecasted years, the beer sales almost remain constant.

But in case of ARIMA model, beer sale for two consecutive forecasted years, beer sales have increasing movement.

checking residuals

checkresiduals(ar\_model)



P value greater than significance level indicate that, there will be no effect of past and present data on the future forecast.

**4]Setting up forecast function for Winter Holt’s and ARIMA model to find best fit model.**

wh <- function(x,h)

{

forecast(ets(x),h=h)

}

ar <- function(x,h)

{

forecast(auto.arima(x),h=h)

}

**Calculate Cross Validation errors for Winter Holt’s (l1) and ARIMA (l2),**

l1 <- tsCV(beer\_time\_series,wh,h=1)

l2 <- tsCV(beer\_time\_series,ar,h=1)

**Mean Square Error of each model,**

For winter holt's model,

mean(l1^2, na.rm=TRUE)

[1] 959.2144

## for ARIMA model,

mean(l2^2, na.rm=TRUE)

[1] 807.5177

From the above values, the model with minimum mean square error is considered to be "Optimal Model"

Since ARIMA model gives minimum Mean Square Error value between the two models.

ARIMA model is selected.

**Interpretation**

* As compared to winter Holt’s model, ARIMA model gives slight better forecast.
* In case of winter Holt’s model forecast plot, for two consecutive forecasted years, the beer sales almost remain constant.
* But in case of ARIMA model, beer sale for two consecutive forecasted years, beer sales have increasing movement.
* As compared to the previous data of 18 years, Quarter 4 beer sales also goes down which affects the rest of the quarters too. Again in 2021, quarter 4 beer consumption will increase but quarter2 will again go down beyond earlier one.

**Conclusion**

* Overall there seems to be slight decrease in beer consumption for next 2 years.
* Quarter4 beer sales slightly goes down in 2020 but maintains consistency in beer sales in 2021.
* By 2020 and 2021, Quarter 2 beer sales will have lowest beer sales compared to previous 18 years.
* Quarter3 maintains its consistency of beer sales in upcoming years. But there are chances that low performance of Quarter 2 might affect the Quarter 3 performance.
* Quarter1 beer sales might get some influence from the Quarter4 beer sales.