**Business Problem**

Quarterly beer sales data has been provided in the beer.csv files. Our Aim is “Using the Winter-Holts methods and ARIMA, model the data and predict for the next 2 years”.

**Description**

There are 72 observations and one variable names OzBeer. It is quarterly data. So, 72 observations mean 72/4=18 years data. OzBeer variable contains the sales information. Here the data period is not known. Assume the data is for the past 18 years.

**Time Series Analysis**

Anything that is observed sequentially over time is a time series.

**Dataset**

data.frame': 72 observations of 1 variable:

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

Class of the dataset is dataframe

**summary(beer_timeseries)**

```
    OzBeer
 Min.   :212.8
 1st Qu.:272.6
 Median :317.5
 Mean   :329.9
 3rd Qu.:379.7
 Max.   :525.0
```

**Convert the data into time series one**

The dataset does not have any column stating year and month of data collection. Starting year been assumed as 2003. This is not shared as part of assignment.

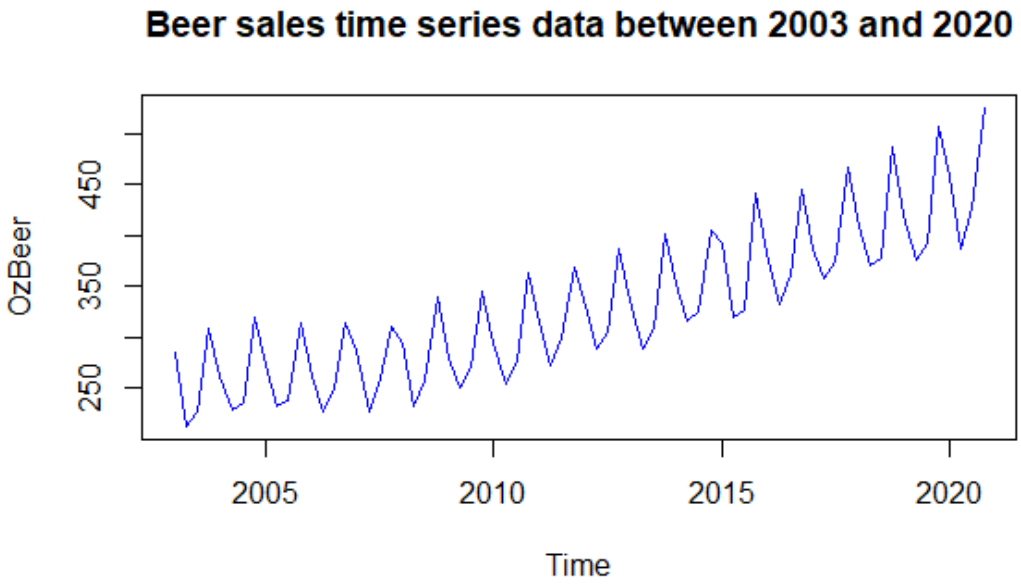
```
start(beer_timeseries)
[1] 2003    1
> end(beer_timeseries)
[1] 2020    4
```

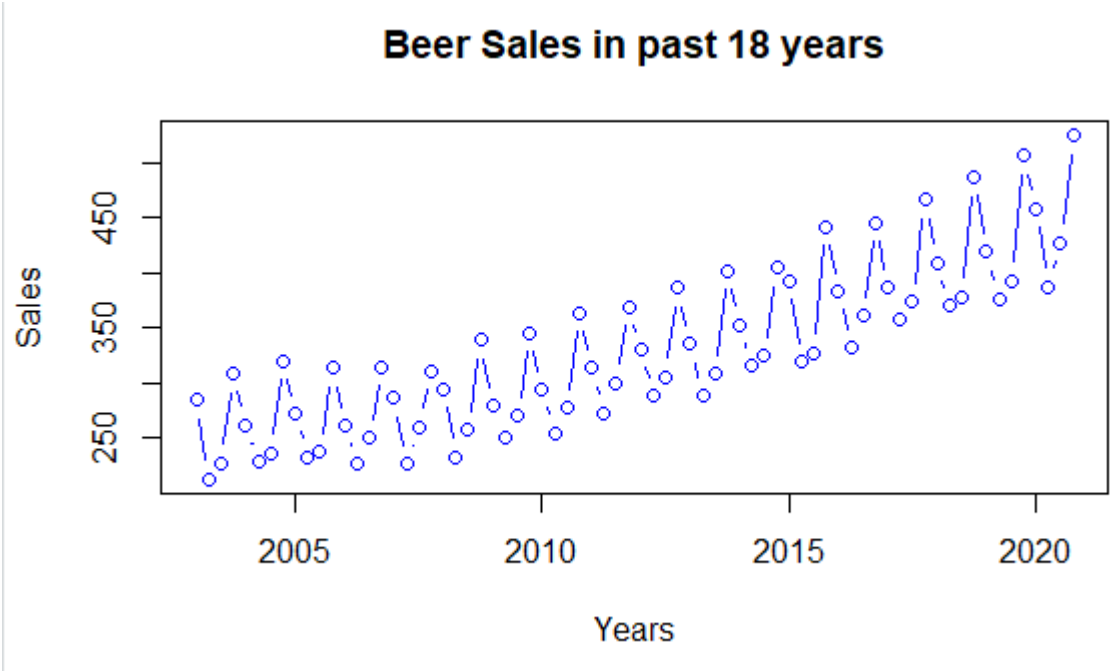
**Importing libraries**

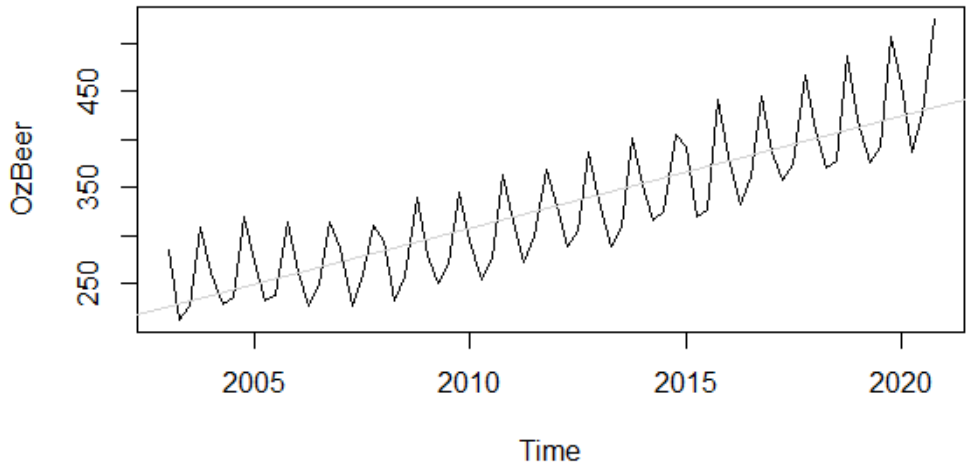
```
library('ggplot2') # visualization
library('ggthemes') # visualization
library('scales') # visualization
library('forecast')
library('TSA')
library('tseries')
library('caret')
library('TeachingDemos')
library('astsa')
library(fpp2)
library(stats)
library('xts')
```

**1. Read the data as a time series object in R. Plot the data.**

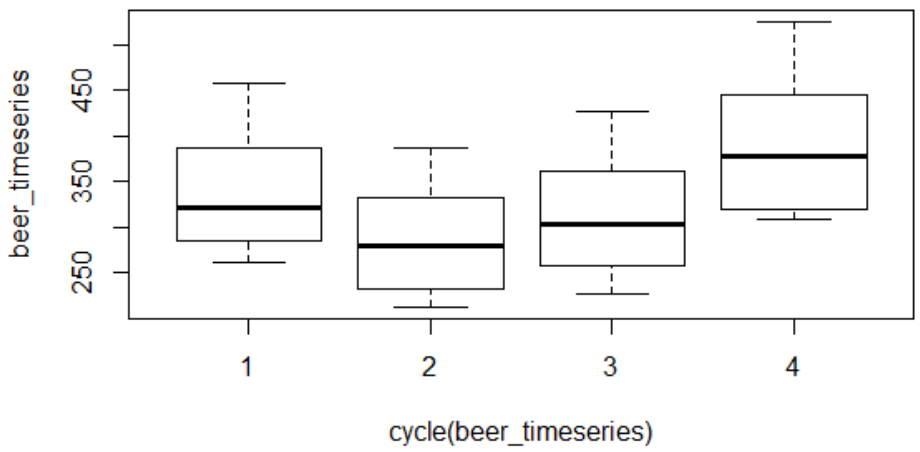
**Visualize the time series**



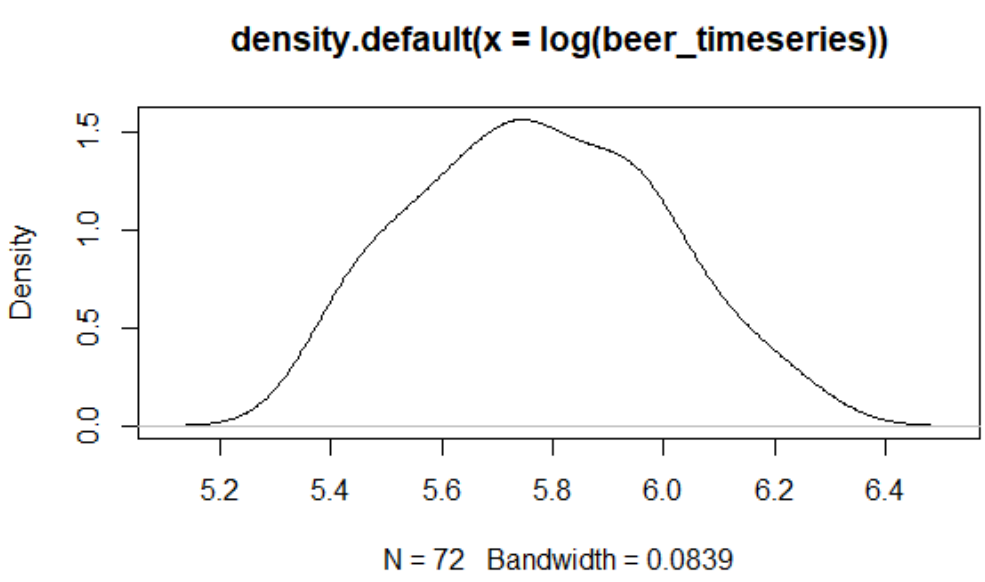


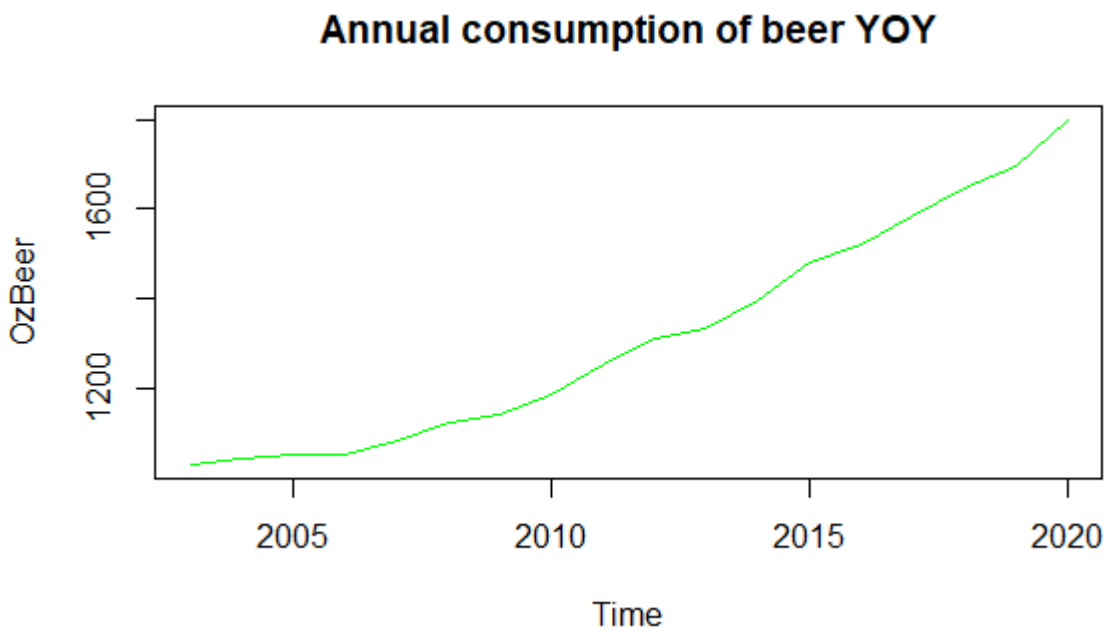




Data is plotted with time series with abline to understand if the data has any pattern. The abline is a simple linear regression line. Most time series patterns can be described in terms of two basic classes of components: trend and seasonality. The former represents a general systematic linear. The latter may have a formally similar nature however, it repeats itself in systematic intervals over time.


beer_timeseries
250
350
450
1
2
3
4
cycle(beer_timeseries)


Histogram of beer_timeseries
Frequency
0
5
10
15
200
250
300
350
400
450
500
550
beer_timeseries


density.default(x = log(beer_timeseries))
Density
0.0
0.5
1.0
1.5
5.2
5.4
5.6
5.8
6.0
6.2
6.4
N = 72 Bandwidth = 0.0839




**2. What are your observations? What components of the Time Series are present in this data?**

Key Observations are: – From the graphs, we see that there is a linear trend and seasonality in the time series. The year-on-year trend clearly shows that the Beer sales have been increasing without fail which shows rising prosperity and greater availability of beer with time. Cycle affect seems to be there in Beer sales. Annual Beer sales also confirm the increasing trend.

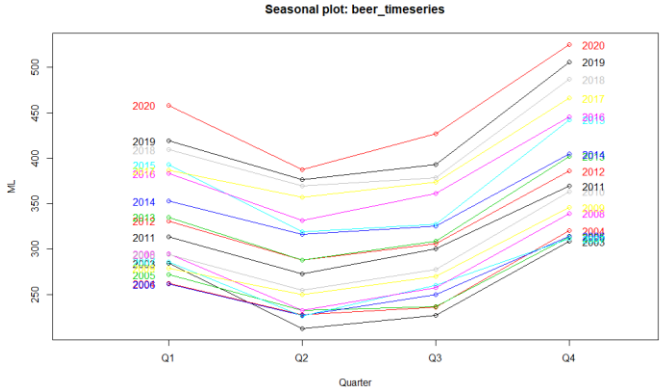
Again, it seems that this time series could probably be described using an additive model, as the seasonal fluctuations (The magnitude of the seasonal component) are roughly constant in size over time and do not seem to depend on the level of the time series, and the random fluctuations also seem to be roughly constant in size over time.

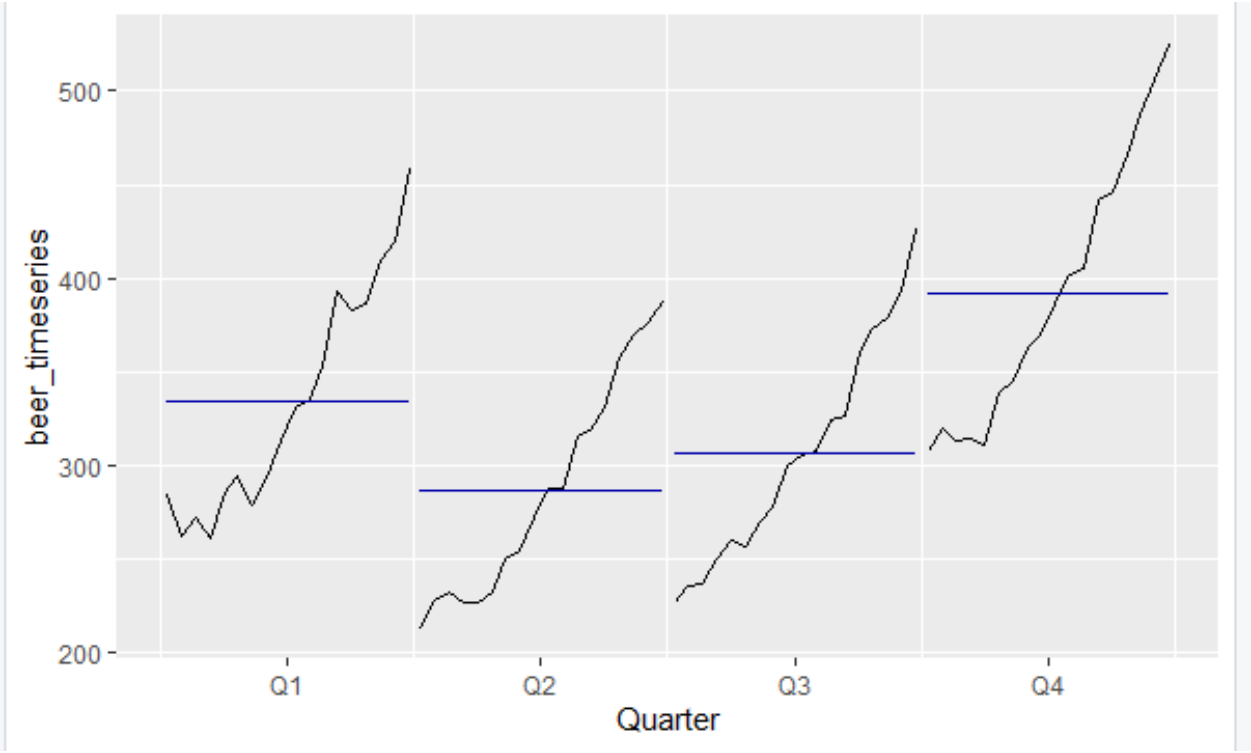
The seasonality is getting repeated for every 4 quarters and is almost constant.

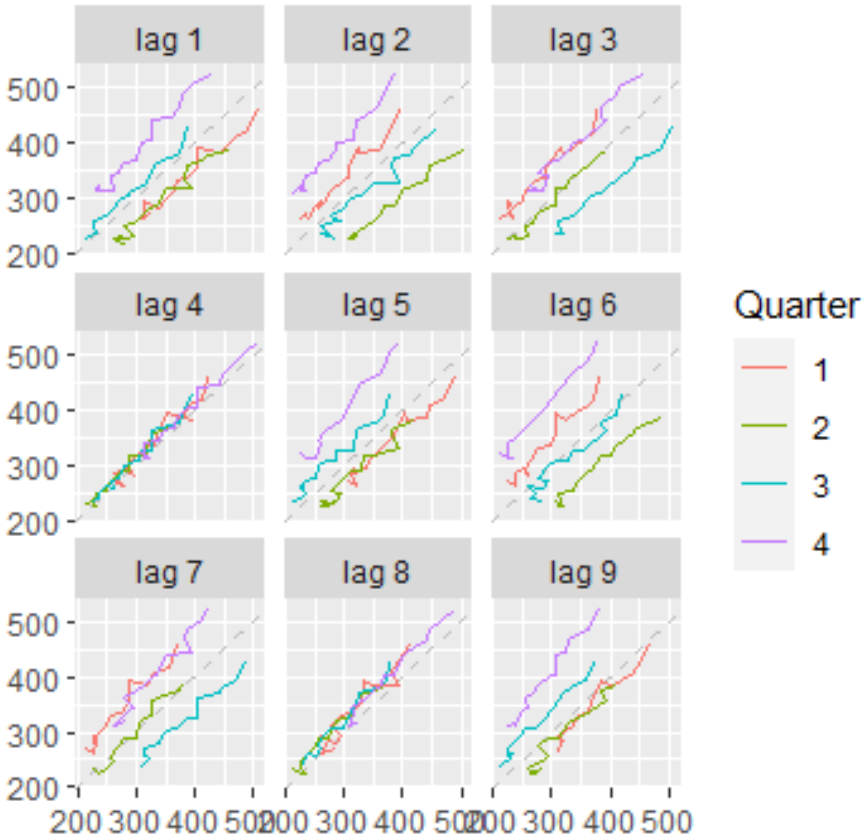
Frequency is identified as 4.

**3. Comment on the periodicity and stationarity of the time series. Use plots to explain your observations. If the Time Series is not stationary, then stationaries it.**

- Periodicity: Quarterly periodicity from 2003 Q1 to 2020 Q4



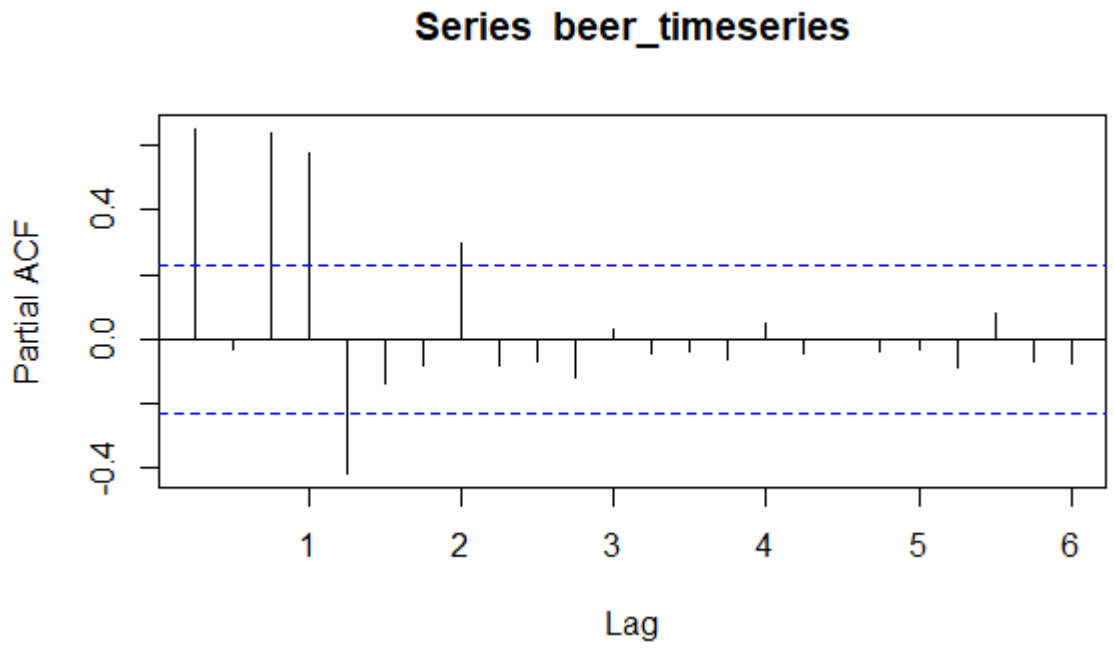

beer_timeseries
500
400
300
200
Q1
Q2
Q3
Q4
Quarter


lag 1
lag 2
lag 3
lag 4
lag 5
lag 6
lag 7
lag 8
lag 9
500
400
300
200
200 300 400 500
Quarter
1
2
3
4

**To check whether beer_ts series is stationary or non-stationary.**

**Autocorrelation Analysis**

ACF and PACF plots


Autocorrelation is the correlation of a Time Series with lags of itself. It shows if the previous states (lagged observations) of the time series has an influence on the current state. In the autocorrelation

chart, if the autocorrelation crosses the dashed blue line, it means that specific lag is significantly correlated with current series. It is used commonly to determine if the time series is stationary or not. A stationary time series will have the autocorrelation fall to zero quickly but for a non-stationary series it drops gradually.

Partial Autocorrelation and PACF plot

Partial Autocorrelation is the correlation of the time series with a lag of itself, with the linear dependence of all the lags between them removed.

**Augmented Dickey-Fuller Test**

data:  beer_timeseries

Dickey-Fuller = -0.73068, Lag order = 4, p-value = 0.9633

alternative hypothesis: stationary

As pvalue is greater than .05, we would reject the alternative hypothesis and accept the null hypothesis that the series is non-stationary.

**Shapiro-Wilk normality test**

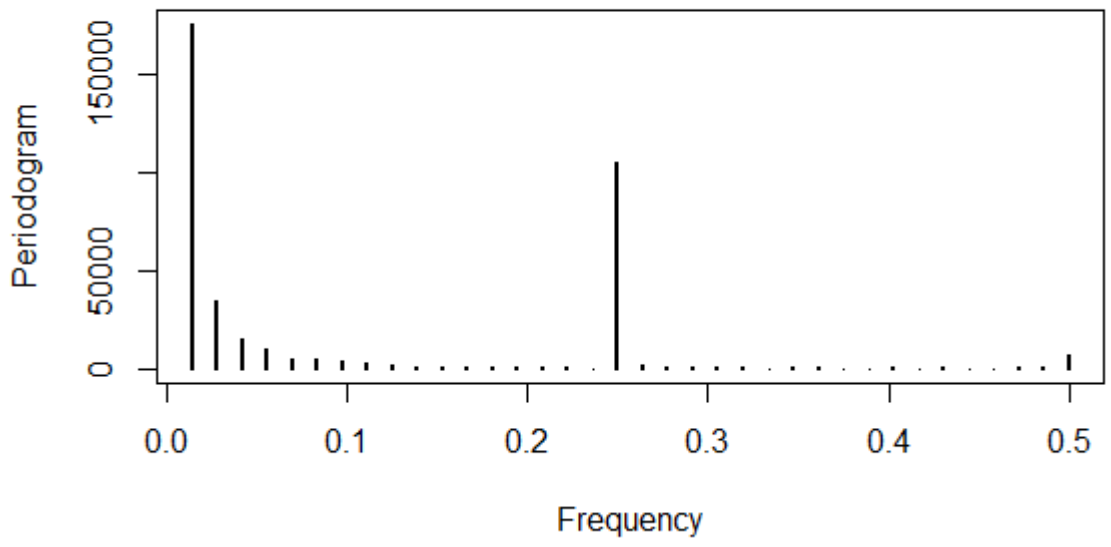
data:  beer_timeseries

W = 0.96352, p-value = 0.03499

pvalue is less than alpha. Hence, accept alternative hypothesis that the series is normally distributed.

From the test and plots it is found that the time series is non stationary.

Step 2:  Time Series Decomposition

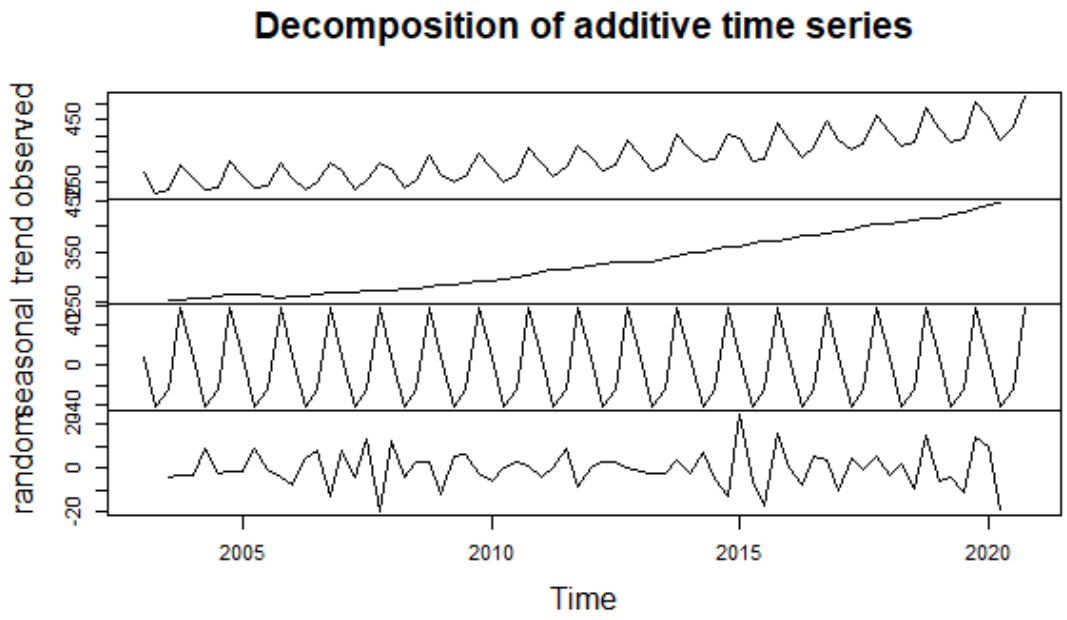


To perform the decomposition, it is vital to use a moving window of the exact size of the seasonality. Therefore, to decompose a time series we need to know the seasonality period: weekly, monthly, etc Using a “periodogram” in R we detect the seasonality.

The signal is the strongest in the beginning and then it is strongest at a frequency of 0.25. As we know Time =1/frequency Time =1/0.25 = 4 The relationship between periodicity and frequency is given by Periodicity = 1/Frequency Periodicity = 1/.25 = 4

If we check the graph, the beer production clearly follows an annual seasonality and records data quarterly i.e., 4 times in a year. This gives a periodicity of 4 quarters. Use a moving average window of 4.

**Decomposing the model into ‘Level’, ‘Trend’ and ‘Seasonality’**




This seasonal time series consists of a trend component, a seasonal component, and an irregular component. Decomposing the time series means separating the time series into these three components: that is, estimating these three components

Here Seasonality is constant. Trend is more important than seasonality.

The plot shows the observed series, the smoothed trend line, the seasonal pattern and the random part of the series. Note that the seasonal pattern is a regularly repeating pattern. The elements of figure are the effects for the four quarters.

attributes(decompose_beer_data)

$names

[1] "x"  "seasonal" "trend"  "random"  "figure"  "type"

$class

[1] "decomposed.ts"

**Stationarize the Series**

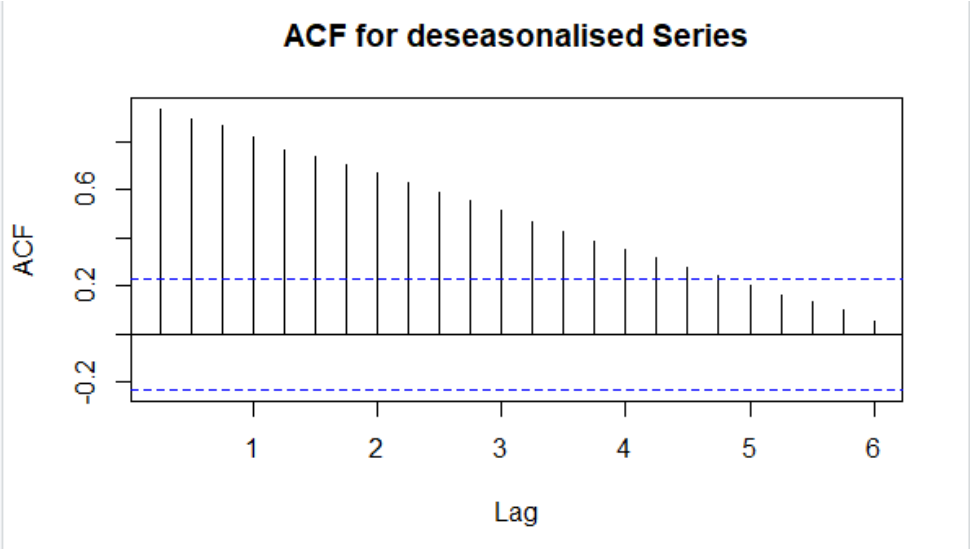
Here the components add together to make the time series. If we have an increasing trend, we will still see roughly the same size peaks and troughs throughout the time series. Note that the peak height of the graph in the seasonal component is almost constant and hence our assumption of additive model is validated Additive: xt = Trend + Seasonal + Random

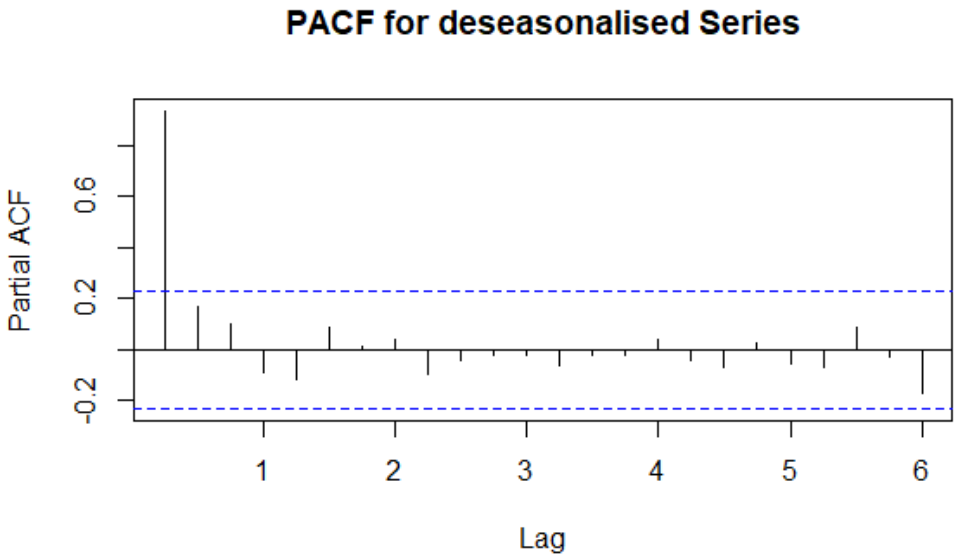
#We will adjust/ remove seasonality and check

**Augmented Dickey-Fuller Test**

data:  deseasonal_beer

Dickey-Fuller = -0.79513, Lag order = 4, p-value = 0.9577

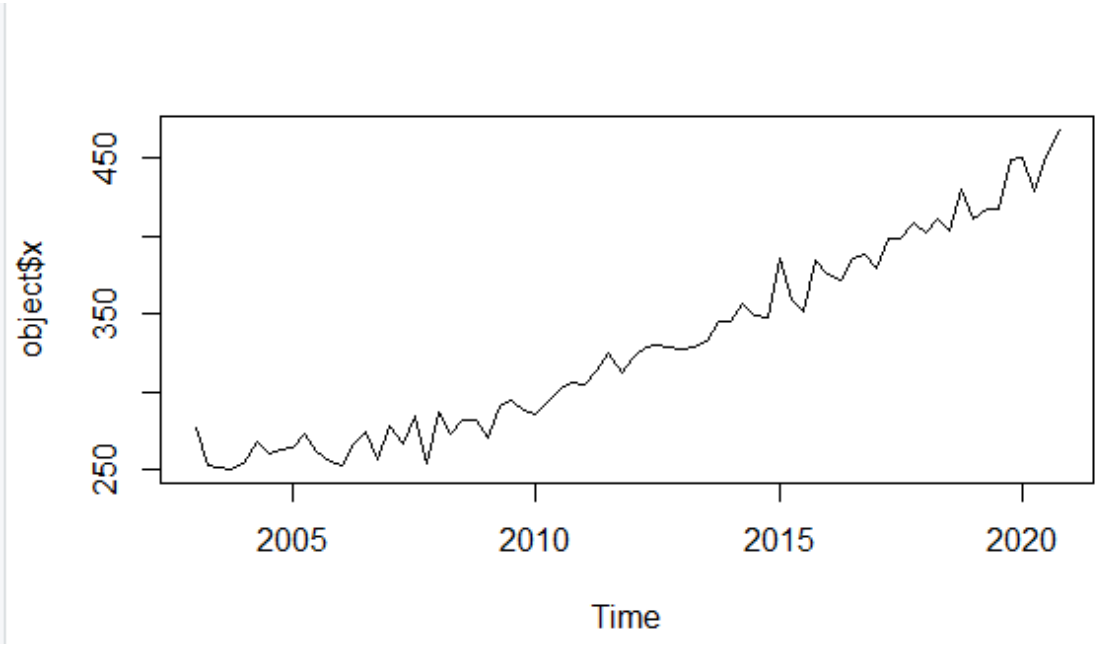



Using the combine results of both the tests we conclude that the given time series is not stationary.

**Seasonality adjusted**

We can seasonally adjust the time series by estimating the seasonal component and subtracting the estimated seasonal component from the original time series. We can see that the seasonal variation has been removed from the seasonally adjusted time series. The seasonally adjusted time series now just contains the trend component and an irregular component. Data has become much smoother after removing the seasonality.




**Removing Trends to achieve stationary**

Removing the previously calculated trend from the time series will result into a new time series clearly exposing the seasonality.

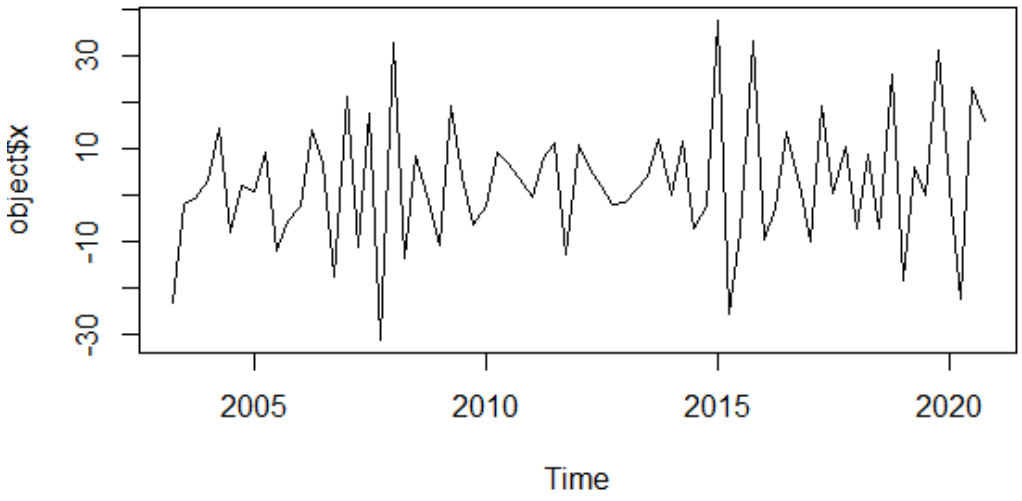
**Generating Stationary Timeseries**

Differencing the time series data

Differencing a series d times will make it I(d) series stationary. In addition differencing once will remove any linear trend.

R has a built in function that allows us to determine how many times we have to conduct trend and seasonal differencing nsdiffs:-estimates the number of seasonal differences.

Remove the seasonal part of time series from the detrended time series




**Augmented Dickey-Fuller Test**

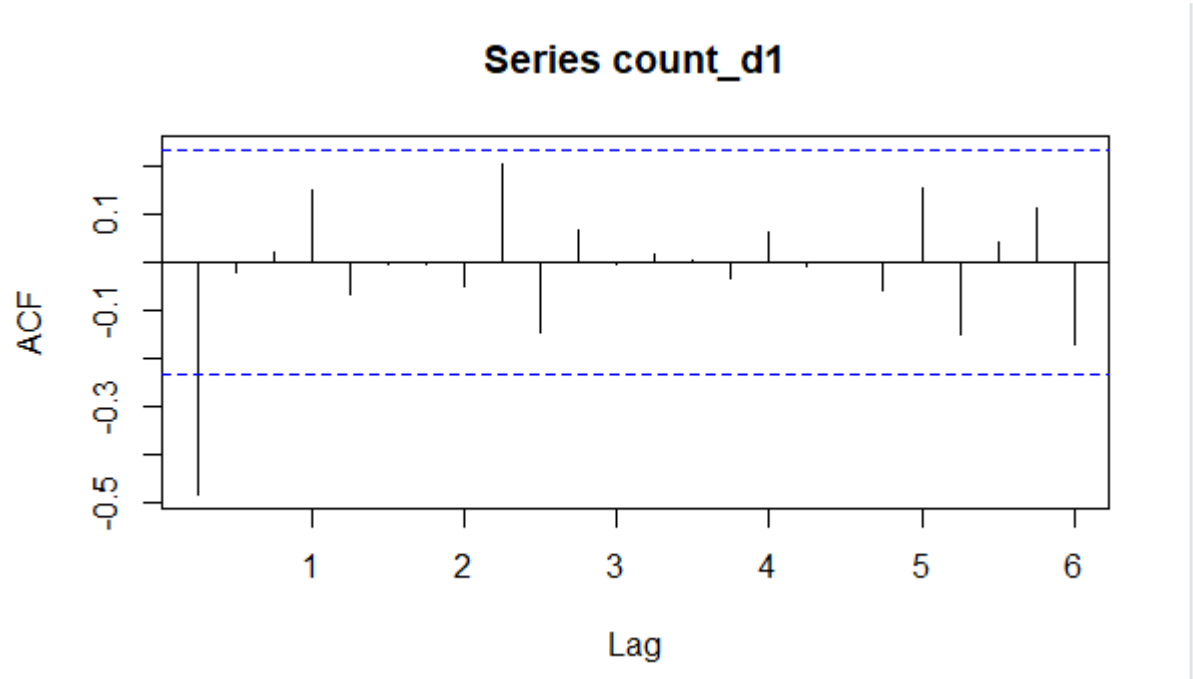
data:  count_d1

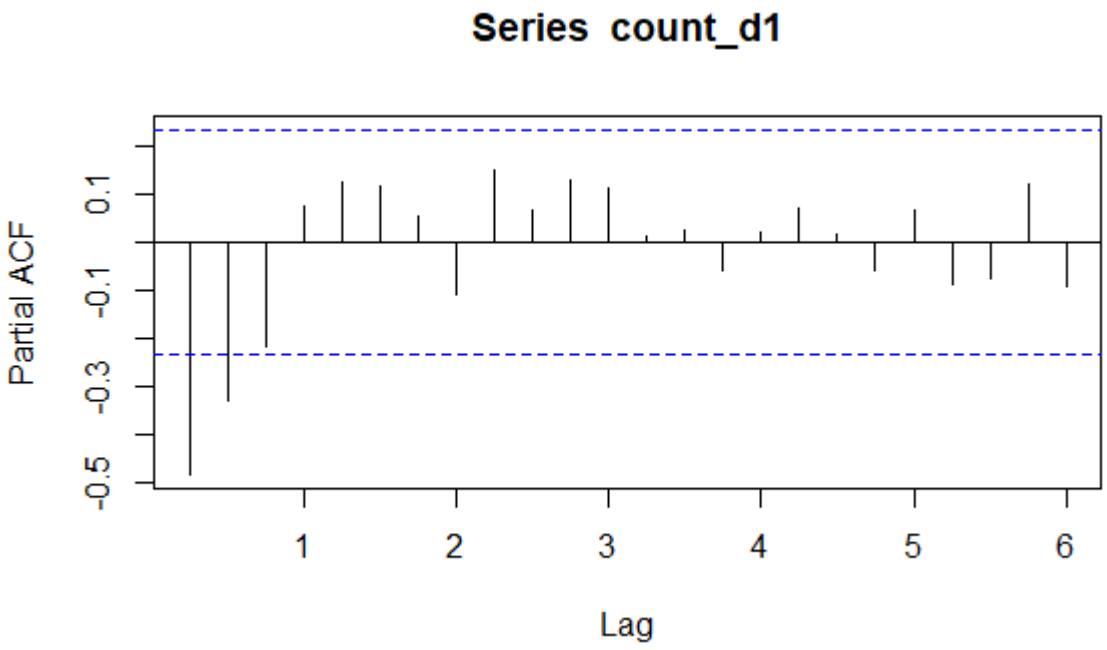
Dickey-Fuller = -5.8466, Lag order = 4, p-value = 0.01

alternative hypothesis: stationary

#P- Value is 0.01 <0.05 so reject the null Hypothesis.

#Null Hypothesis is Data is non Stationary, that means Data is stationary






Plot and adf test are confirming that the time series is stationary now

Examining Remaining Random Noise- The previous steps have already extracted most of the data from the original time series, leaving behind only "random" noise. The additive formula is "Time series = Seasonal + Trend + Random", which means "Random = Time series – Seasonal – Trend"

Dataset is split into training and test dataset

4. **Using the Winter-Holts methods and model the data and predict for the next 2 years. Your submission should contain the complete modelling steps with explanations. Include pictures and R-code where applicable.**

   As there is trend and seasonality in the data, we can use HOLT WINTER's Method to forecast the time series data.

Step 1: Identifying an Additive Timeseries

Holt-Winters method is an exponential smoothing approach for handling SEASONAL data.

In our case, the beer data is time series with constant seasonal variations so we will use the **Additive Holt-Winters method**

**summary(winter_model1)**

ETS(A,A,A)

Call:

ets(y = beerTStrain, model = "AAA")

Smoothing parameters:

alpha = 0.0442

beta  = 0.0441

gamma = 2e-04

Initial states:

l = 258.5383

b = -0.0577

s = 54.4219 -24.0089 -40.7183 10.3053

sigma:  9.8525

AIC AICc BIC

491.0121 494.9251 509.2403

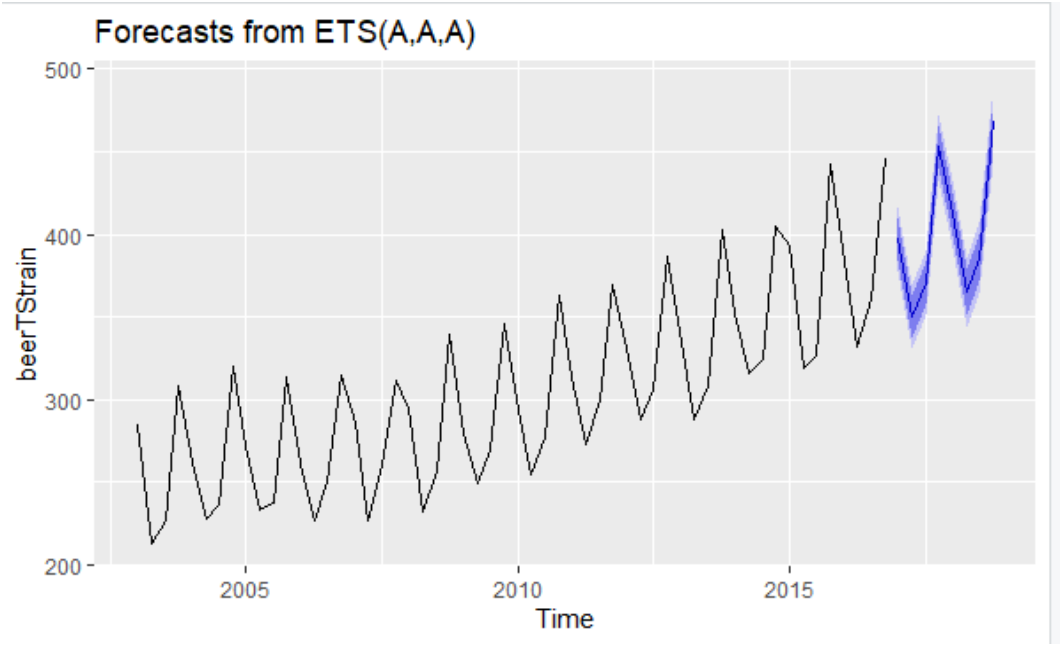
Training set error measures:

| | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 |
|---|---|---|---|---|---|---|---|
| Training set | 1.545655 | 9.12161 | 7.495372 | 0.5140247 | 2.517315 | 0.6041844 | -0.129779 |

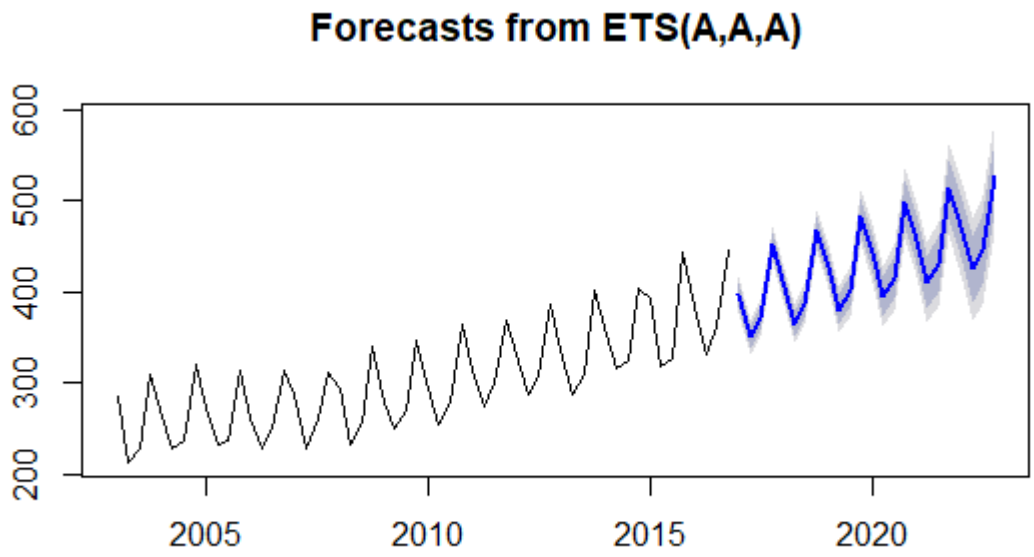
**accuracy(fcast_ets1,beerTStest)**

| | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 |
|---|---|---|---|---|---|---|---|
| Training set | 1.545655 | 9.12161 | 7.495372 | 0.5140247 | 2.517315 | 0.6041844 | -0.1297790 |
| Test set | 4.801830 | 12.89103 | 10.804528 | 0.9053668 | 2.435448 | 0.8709277 | -0.1617099 |

| | Theil's U |
|---|---|
| Training set | NA |
| Test set | 0.1984544 |




The plot shows the forecasted values, the forecasted values are blurred this indicates that the forecasted values will vary within the blur part.




**Check accuracy on test data**

accuracy(fcast_ets1,beerTStest)

| | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 |
|---|---|---|---|---|---|---|---|
| Training set | 1.545655 | 9.12161 | 7.495372 | 0.5140247 | 2.517315 | 0.6041844 | -0.1297790 |
| Test set | 4.801830 | 12.89103 | 10.804528 | 0.9053668 | 2.435448 | 0.8709277 | -0.1617099 |

Theil's U

Training set NA

Test set 0.1984544

Running Holt Winter's additive method by taking logarithm on the timeseries data:

summary(winter_logmodelAdd)

ETS(A,A,A)

Call:

ets(y = log(beerTStrain), model = "AAA")

Smoothing parameters:

alpha = 0.0503

beta = 0.0503

gamma = 0.0014

Initial states:

l = 5.5459

b = 0.0025

s = 0.1749 -0.0735 -0.1416 0.0402

sigma: 0.0333

AIC AICc BIC

-146.3697 -142.4566 -128.1415

Training set error measures:

ME RMSE MAE MPE MAPE MASE

Training set 0.00266642 0.03079933 0.02618507 0.04609786 0.4602777 0.6420862

ACF1

Training set -0.1080065

From the above results, we can find the overall smoothing parmater (alpha), trend smoothing parameter(beta) and seasonal smoothing parameter(gamma). The initial values of level, trend and seasonality is interpreted.

The AIC for additive model is -146.37.

accuracy(fcast_logets,log(beerTStest))

ME RMSE MAE MPE MAPE MASE

Training set 0.002666420 0.03079933 0.02618507 0.04609786 0.4602777 0.6420862

Test set 0.007988705 0.02879981 0.02329314 0.13667277 0.3891611 0.5711728

ACF1 Theil's U

Training set -0.1080065 NA

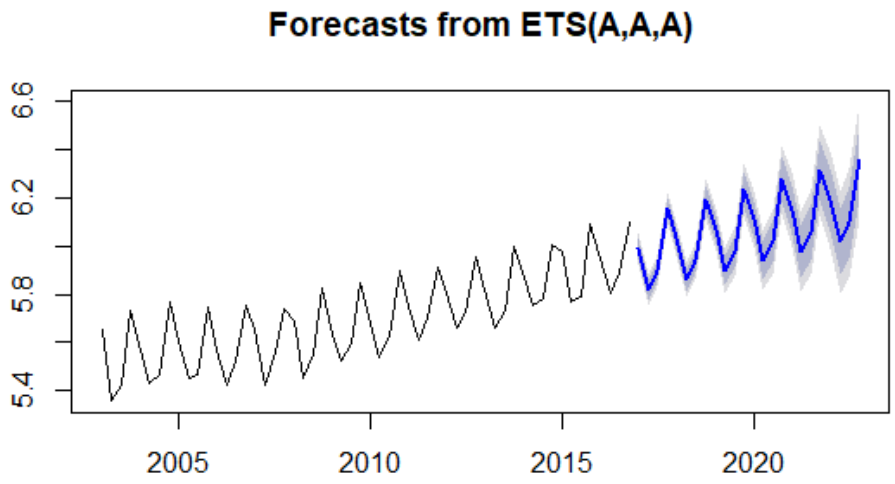
Test set -0.2582614 0.1896008

MAPE value is .486 for train data and .467 for test data.

#The model is valid and accurate

Forecasting for the next 2 years

Using forecast method, we have predicted below.The forecasted values will show the point forecast values and the range with low and high values as 80% and 95% confidence intervals, respectively.



Plot of Holt-Winter prediction has shaded and blue portion. The blue lines show forecasts for the next two years. Notice how the forecasts have captured the seasonal pattern seen in the historical data and replicated it for the next two years. These prediction intervals are a useful way of displaying the uncertainty in forecasts.

**Analysis of Residuals**

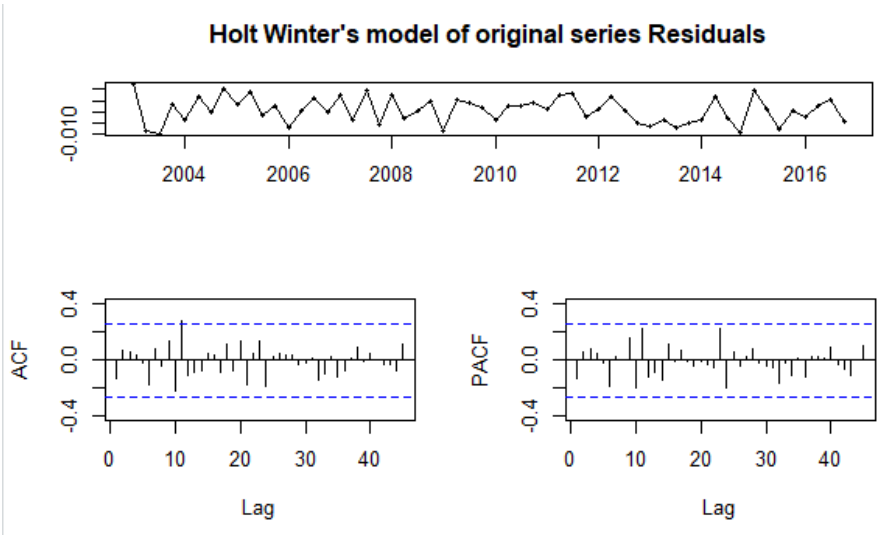
Ljung-Box test is carried out on residuals to see that after fitting the model what remains is the residuals. Null Hypothesis: Data is independently distributed Alternative Hypothesis: Data shows serial correlation

**Box-Pierce test**

data: winter_logmodel$residuals

X-squared = 0.96605, df = 1, p-value = 0.3257

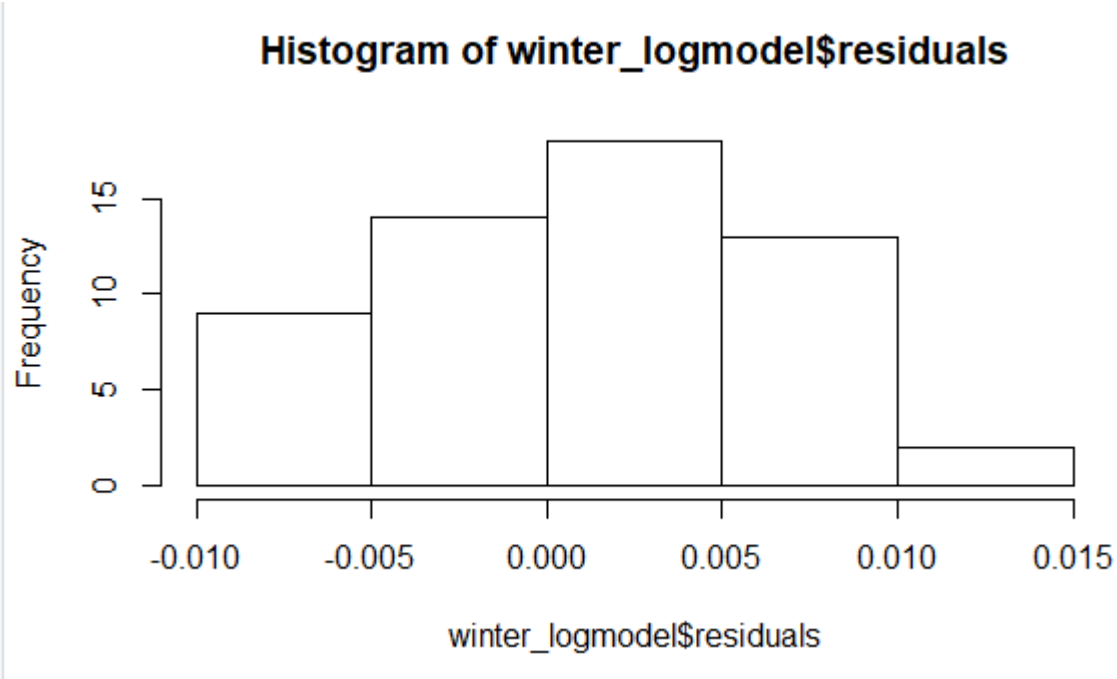
P value is >0.05 so do not reject the null hypothesis, henceforth "Data is independently distributed"

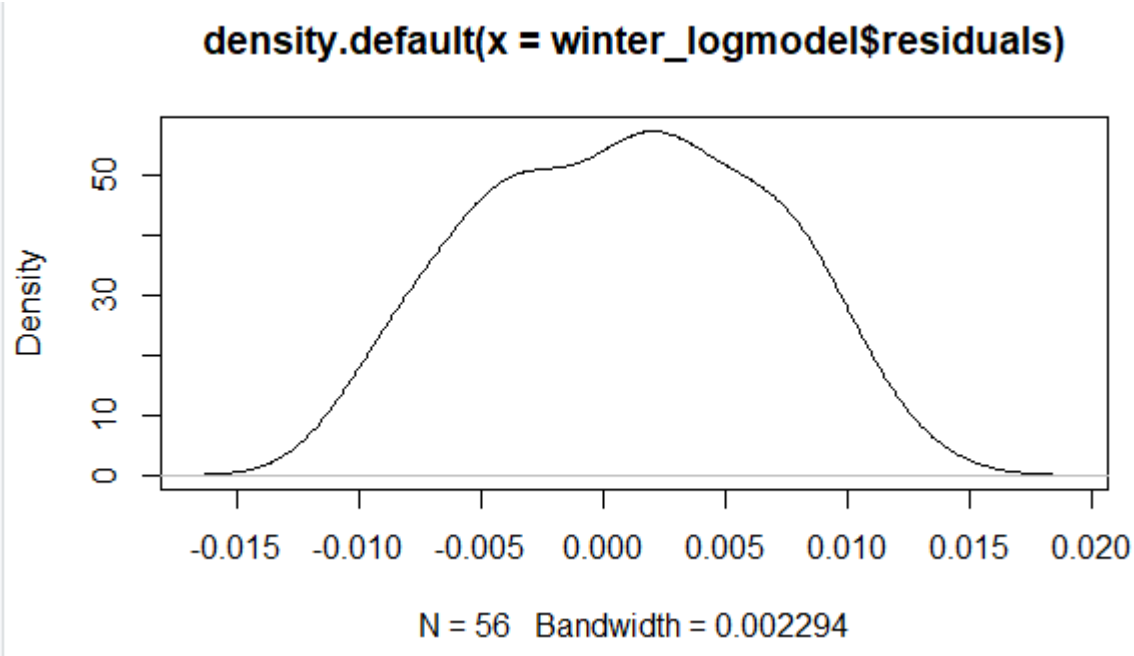



Residuals variance is less constant

As the p value is less than 0.05. We reject the null hypothesis hence number of autocorrelation coefficients are different from zero.

It is quite clear than mean of Residuals is approximately zero and fluctuating over mean of zero.




**Shapiro-Wilk normality test**

data: winter_logmodel$residuals

W = 0.97385, p-value = 0.2623

pvalue is greater than alpha. Hence residuals are normally distributed

The model is valid and accurate.

5. **Using the ARIMA method model the data and predict for the next 2 years. Your submissions should contain the complete modelling steps with explanations. Include pictures and R-code where applicable. (HINT - Use auto.arima() to find the optimum parameters as a basis of building the arima model. auto.arima(Train_data (refers to the train data set), seasonal=TRUE if seasonality is present in the data, FALSE is seasonality is not present.)**

ARIMA(P,d,q): p and q determines the Auto Regressive coefficient and Error terms/ MA Process coefficient in the model which are estimated using Auto Correlation and Partial Autocorrelation functions/plot. It is here that modelling become an "art" when choosing values of p and q

Next step is to find the right parameters to be used in the ARIMA model. We already know that the 'd' component is 1 as we need 1 difference to make the series stationary. We do this using the Correlation plots.

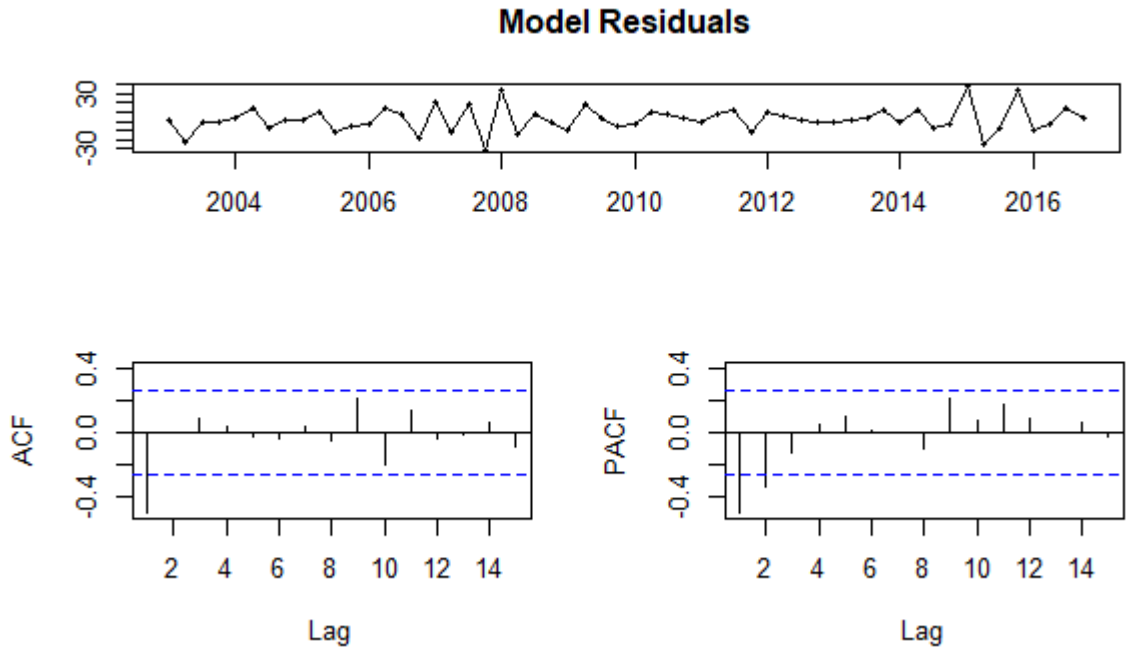
summary(beerArima)

Call:

arima(x = beer_seasadjtrain, order = c(0, 1, 0))

sigma^2 estimated as 180.8: log likelihood = -220.98, aic = 441.95

Using ARIMA Model for prediction



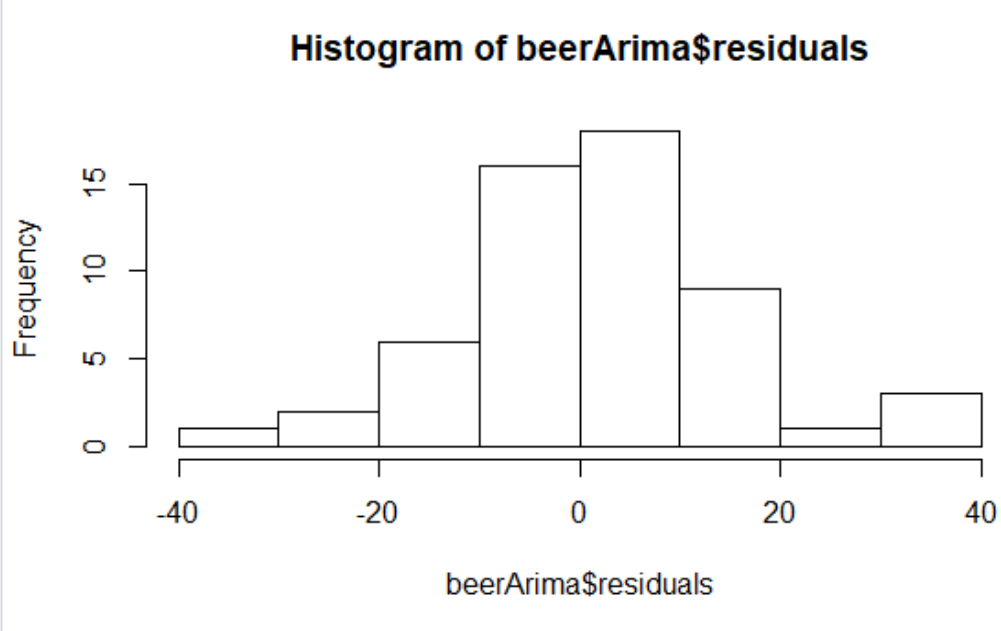

In the ACF and PACF chart if the autocorrelation crosses the dashed blue line then it means that specific lag is significantly related with current series. residuals variance is not constant here.

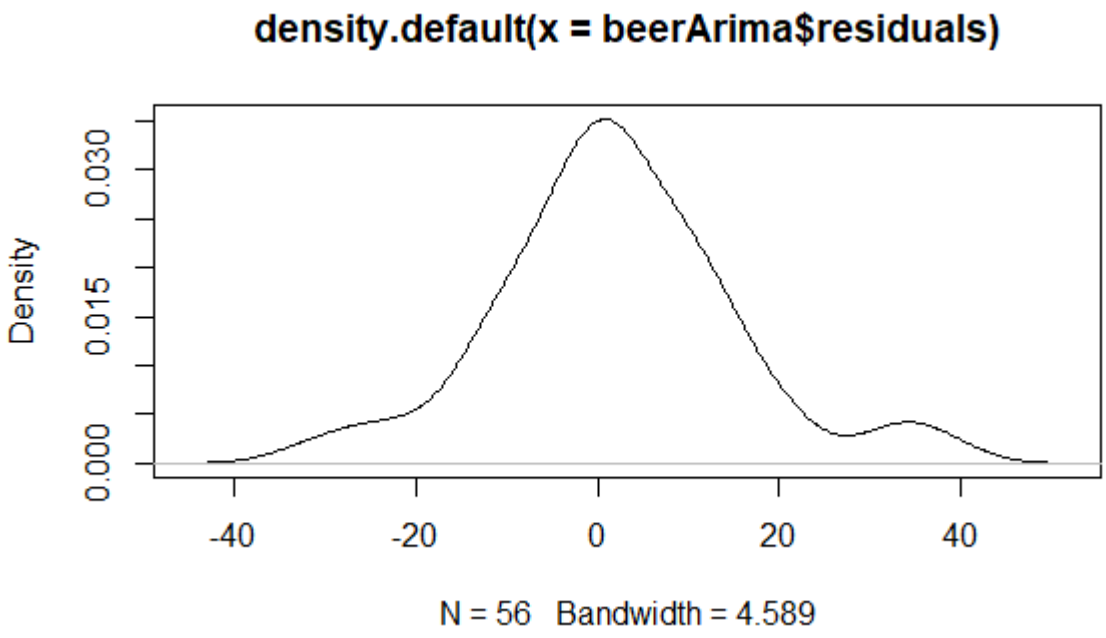
**Box-Pierce test**

data:  beerArima$residuals

X-squared = 14.297, df = 1, p-value = 0.0001561

Box-Ljung (used for significant evidence for non-zero correlations) test shows that p-value is < 0.05. So, we reject the null hypothesis. Our null hypothesis is “There is little evidence of non-zero autocorrelation in the forecast errors.” Residuals are not independent. That is there is a problem of auto correlation



**Shapiro-Wilk normality test**

data:  beerArima$residuals

W = 0.97457, p-value = 0.282

Further, the residuals are normally distributed, thus we can probably conclude this model as an adequate representation of the data.

This is not a valid model and accuracy is not acceptable

We will use auto.arima which will run iteratively to find out optimal values of p,d and q.

Series: beerTStrain

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

Coefficients:

| | ma1 | ma2 | sma1 |
|---|---|---|---|
| | -1.0667 | 0.3407 | -0.6511 |
| s.e. | 0.1349 | 0.1243 | 0.1308 |

sigma^2 estimated as 112.2:  log likelihood=-192.9

AIC=393.79   AICc=394.66   BIC=401.52

Series: beerTStrain

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

Coefficients:

| | ma1 | ma2 | sma1 |
|---|---|---|---|
| | -1.0667 | 0.3407 | -0.6511 |
| s.e. | 0.1349 | 0.1243 | 0.1308 |

sigma^2 estimated as 112.2:  log likelihood=-192.9

AIC=393.79   AICc=394.66   BIC=401.52

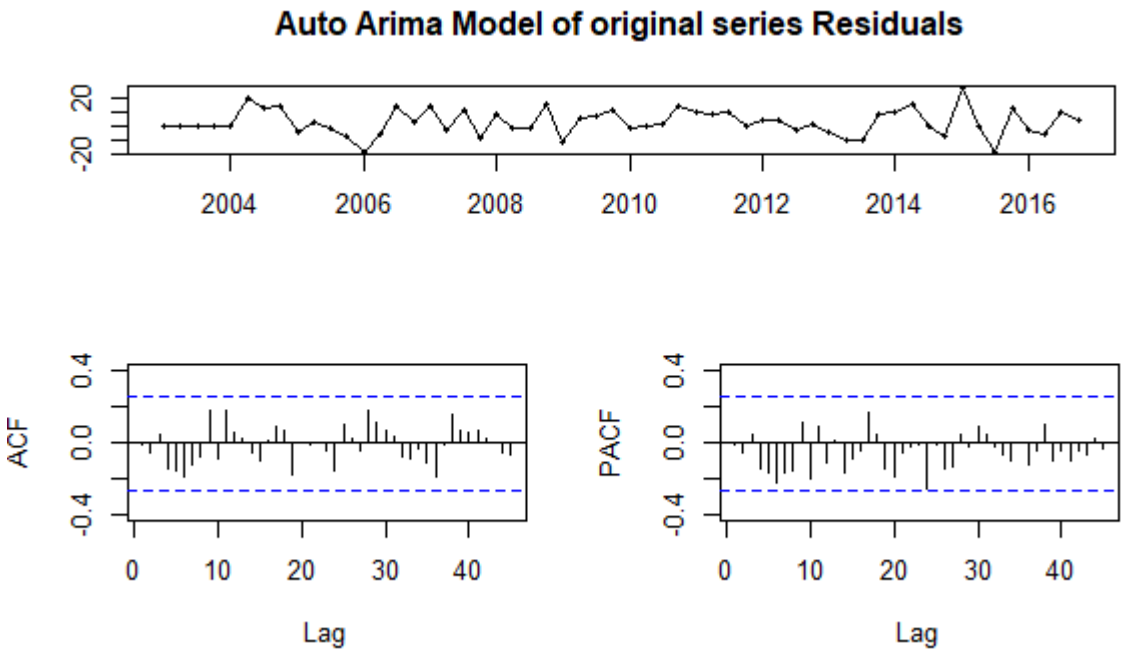
Training set error measures:

| | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 |
|---|---|---|---|---|---|---|---|
| Training set | 2.679271 | 9.807664 | 7.586202 | 0.8311679 | 2.503148 | 0.611506 | -0.01138104 |

Best model will be selected based on AIC value and our optimal model is Best model: ARIMA(0,1,2)(0,1,1)[4]

**Final Model and Residual value**

We will now use ARIMA(0,1,2)(0,1,1)[4] to generate final model. We will plot the residuals and test the residuals also.




Residual plot shows while the mean is roughly constant, the variance does not seem to be obviously constant, but it is close enough.

**Evaluating Residuals**

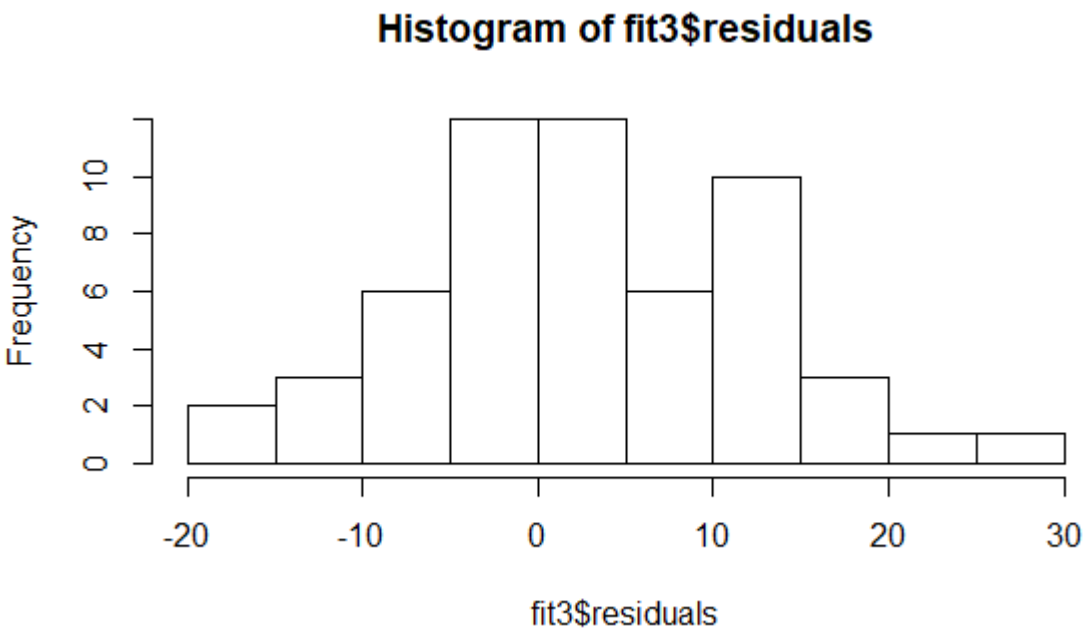
Ljung-Box test is carried out on residuals to see that after fitting the model what remains is the residuals. Null Hypothesis: Data is independently distributed Alternative Hypothesis: Data shows serial correlation

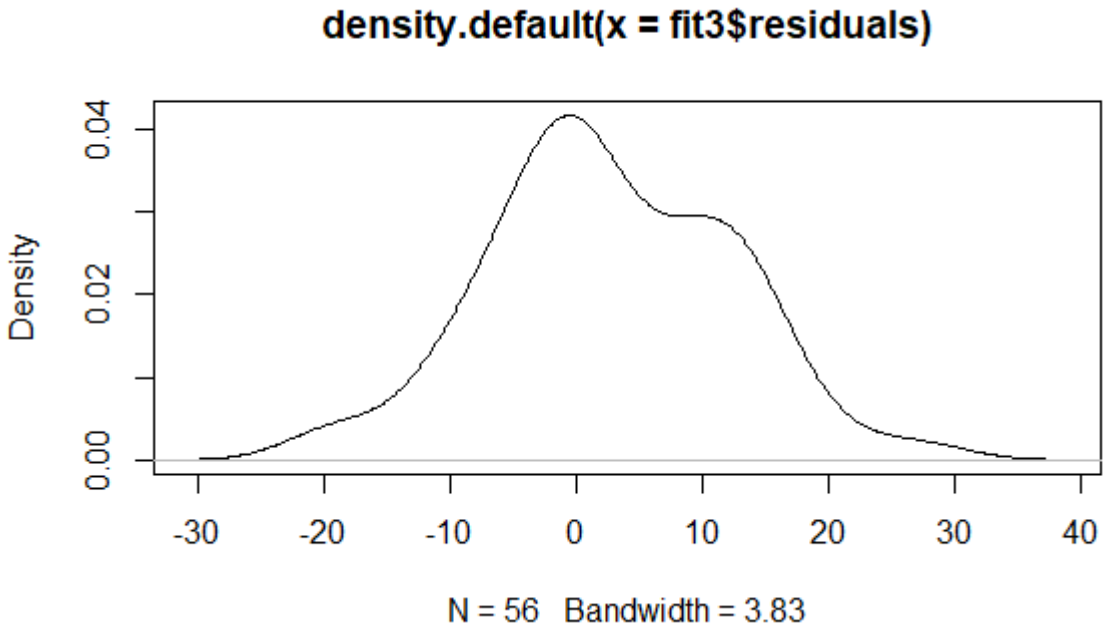
Box-Pierce test

data: fit3$residuals

X-squared = 0.0072536, df = 1, p-value = 0.9321

pvalue is greater than alpha. Residuals are independent.




**Shapiro-Wilk normality test**

data: fit3$residuals

W = 0.98528, p-value = 0.7246

pvalue is greater than alpha. Hence residuals are normally distributed

#This is a valid model

**Fitting ARIMA model**

Based on the information from the above plots, an ARIMA model is fit to the data. With different p and q values, the model is built and the best one with the lowest AIC value is selected.

Auto arima on log series

Series: log(beerTStrain)

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

Coefficients:

| | ar1 | ma1 | sma1 |
|---|---|---|---|
| | -0.4055 | -0.6794 | -0.6393 |
| s.e. | 0.1674 | 0.1298 | 0.1386 |

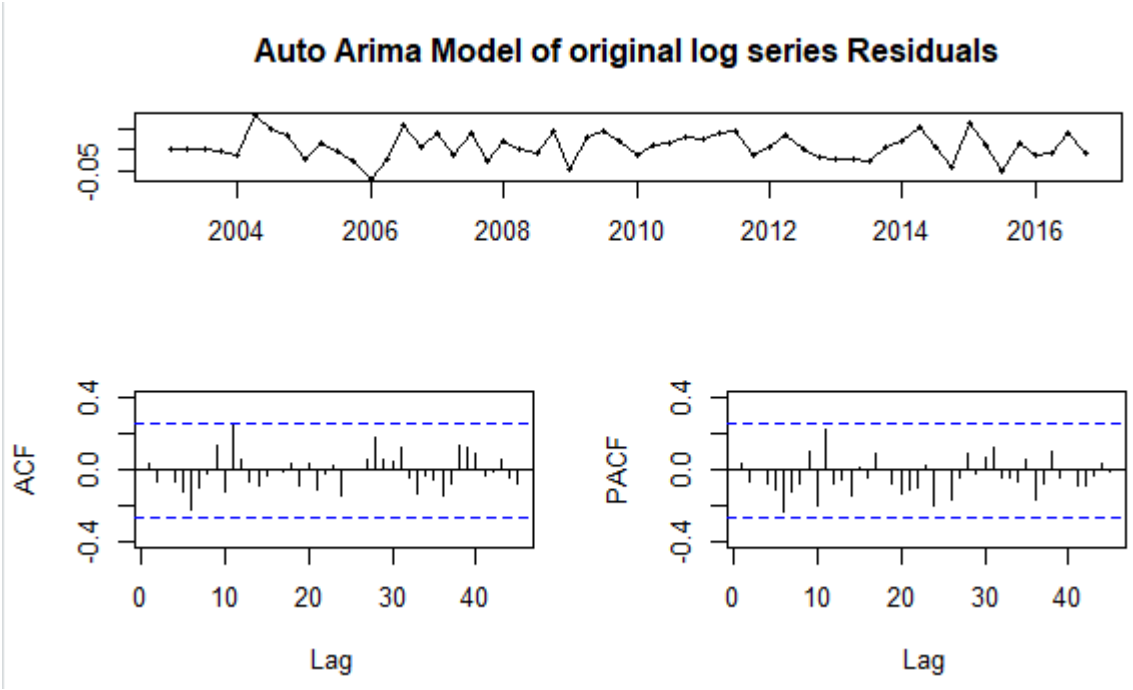
sigma^2 estimated as 0.001212: log likelihood=98.67

AIC=-189.33 AICc=-188.46 BIC=-181.6

Training set error measures:

| | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 |
|---|---|---|---|---|---|---|---|
| Training set | 0.007916231 | 0.03223623 | 0.02559781 | 0.1402197 | 0.4498529 | 0.627686 | 0.02987286 |

Checking for AIC values, the least AIC value will be considered as the best fit model.

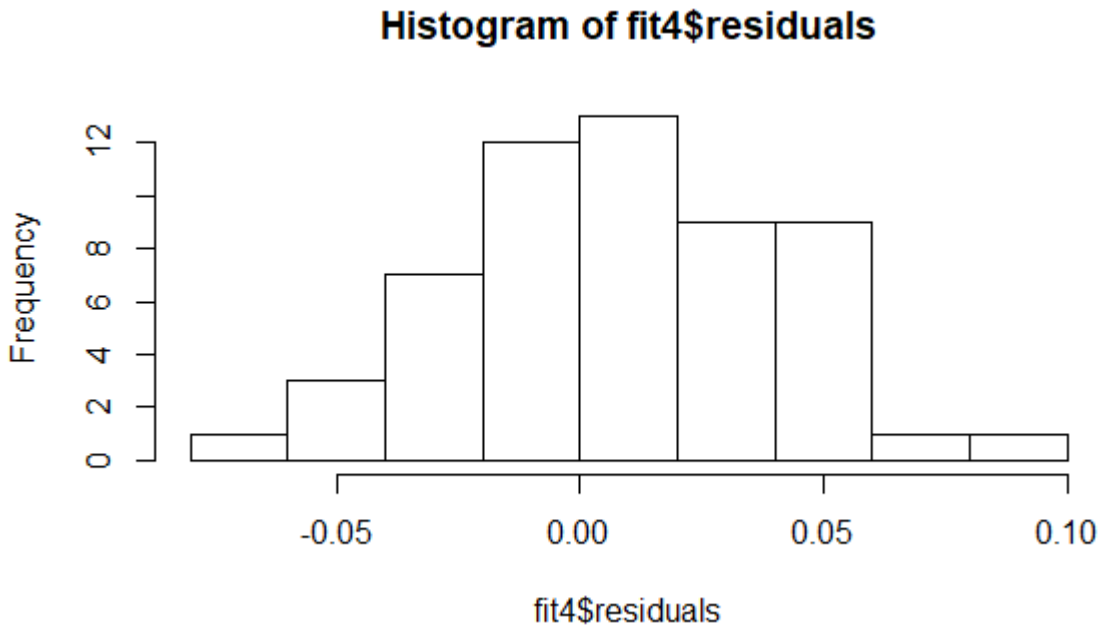


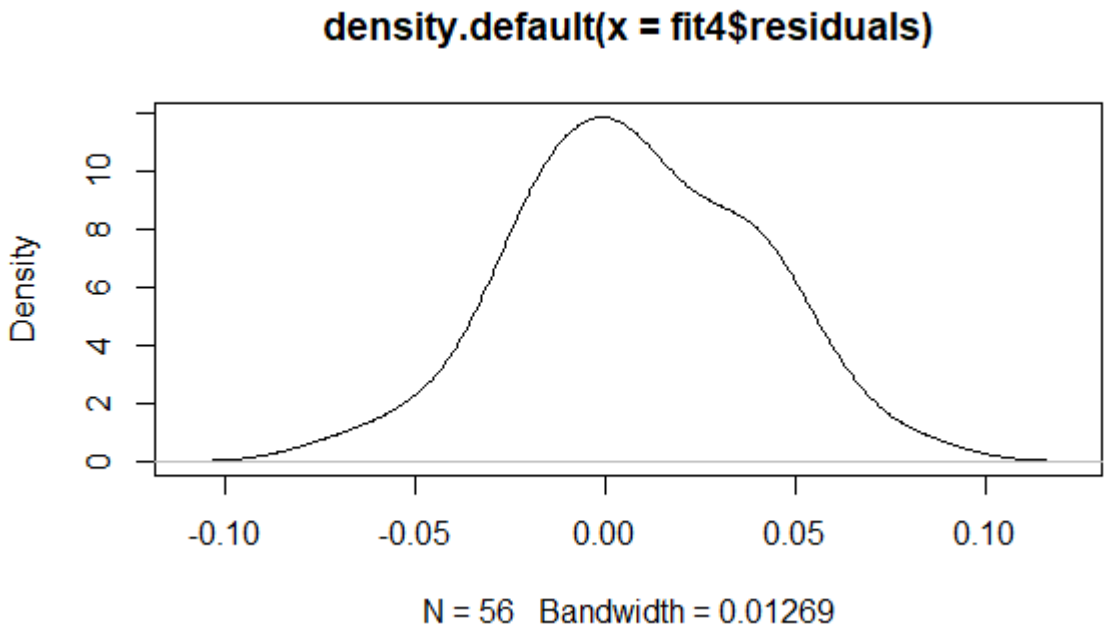

**Box-Pierce test**

data: fit4$residuals

X-squared = 0.049974, df = 1, p-value = 0.8231

pvalue is greater than alpha. Residuals are independent.





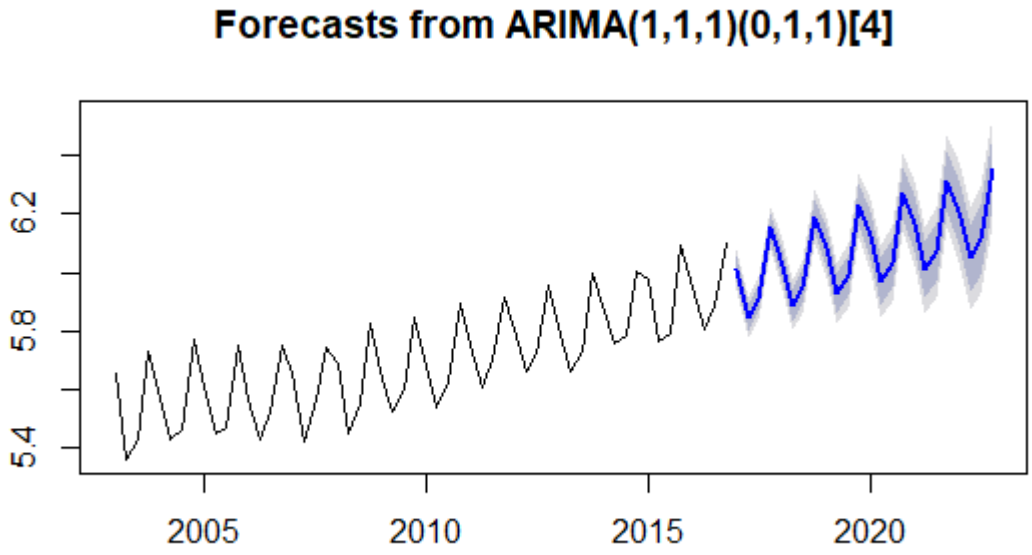

**Shapiro-Wilk normality test**

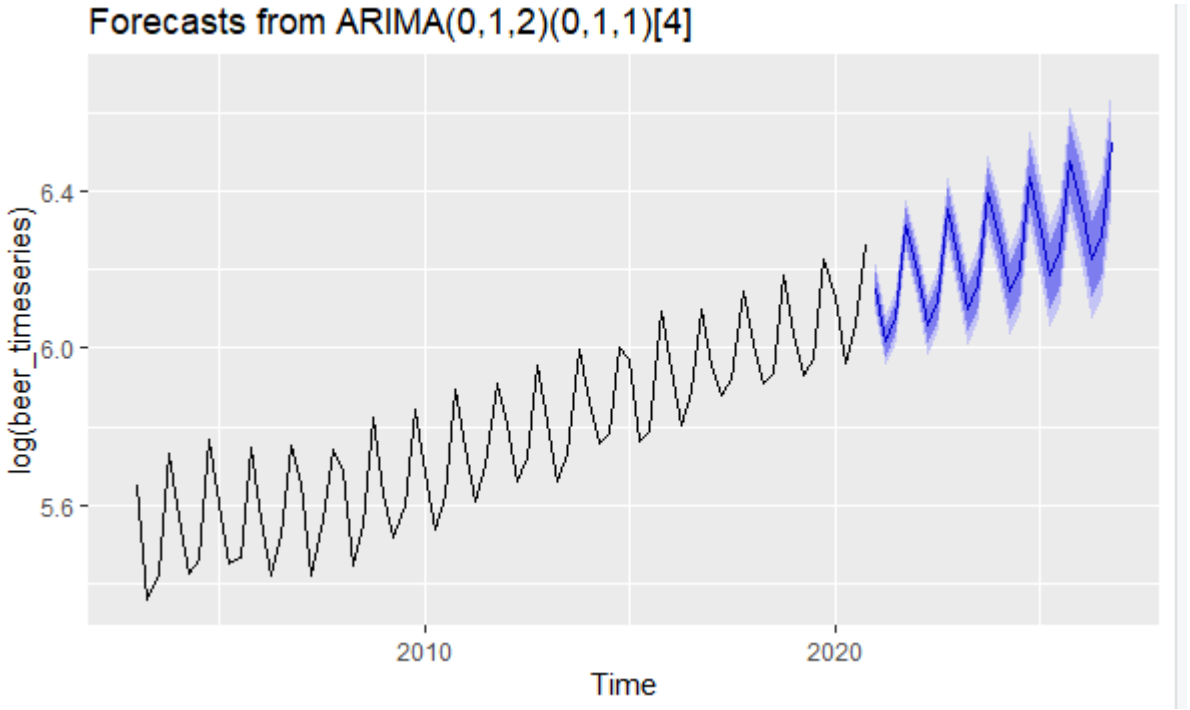
data: fit4$residuals

W = 0.99348, p-value = 0.9905

pvalue is greater than alpha. Hence residuals are normally distributed. This is a valid model.

**Forecasting**






The blue lines show forecasts for the next two years. Notice how the forecasts have captured the seasonal pattern seen in the historical data and replicated it for the next two years. The dark shaded region shows 80% prediction intervals. That is, each future value is expected to lie in the dark shaded region with a probability of 80%. The light shaded region shows 95% prediction intervals. These prediction intervals are a useful way of displaying the uncertainty in forecasts. In this case the forecasts are expected to be accurate, and hence the prediction intervals are quite narrow.

**Evaluating Forecast Accuracy**

Here, we have split the data into training set and testing set we computed accuracy on testing data. The mean predicted value from the seasonal naive method is 458

| | ME | RMSE | MAE | MPE | MAPE | MASE |
|---|---|---|---|---|---|---|
| Training set | 0.007916231 | 0.03223623 | 0.02559781 | 0.14021974 | 0.4498529 | 0.6276860 |
| Test set | - 0.004988432 | 0.02751270 | 0.02255514 | -0.08101203 | 0.37 88178 | 0.5530762 |

| | ACF1 | Theil's U |
|---|---|---|
| Training set | 0.02987286 | NA |
| Test set | -0.38464401 | 0.143591 |

MAPE value is .4499 on train data and .3788 on test data.

#The model is valid and more accurate

**Overall Analysis**

->there is not much difference in the predicted sales of beer from both the methods, the values are quite closer which shows that the predicted values are significant and reliable.

-> After comparing all the accuracy value ME, MPE, MAPE and MASE we concluded that ARIMA model is more reliable in terms of forecasted accuracy

->Beer market is seasonal as people consume more beer in summer as compare to winter because beer drinkers look for a range of new qualities in their brews as well as considered the lighter drink and refreshing too.

->From the forecasted plot it is clearly visible that the beer sales is the highest every year in the month of October and is the lowest in the months of April/July or it is the highest during Q4(Oct-Dec) and the lowest during the Q2(Apr-Jun) and Q3(July-Sep) the months with low temperatures.

->Which is well supported by the fact that the weather of Australia where October to December are the hottest months of the year and beer sales is supposed to be the highest during summer.

->Also, we can predict that weather plays an important role in beer sales.

.