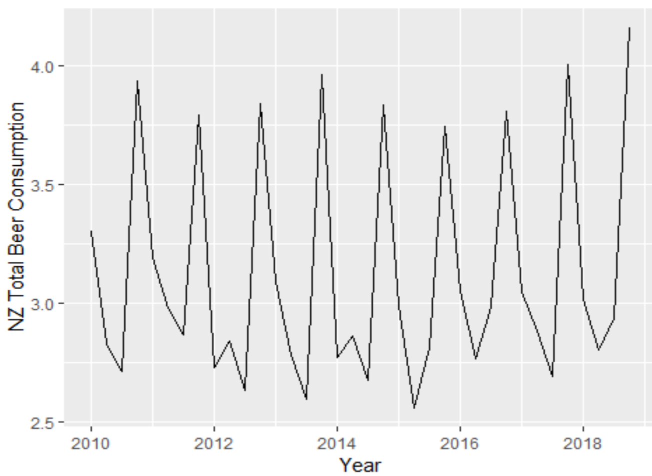
Assignment 2

YOU CHEN 14865993

1. Question 1- ETS (14 marks)

(a) Plot the series and discuss the main features of the data including stationarity (2 marks).

```
#import data and install all the packages needed for assignment
library(readxl)
library(fpp2)

data <- read_excel("../data/NZ_TotalBeer_Quarterly.xlsx",skip=1,col_typ

#plot the data
ts_beer <- ts(data[,2],start=2010,frequency = 4)
ts_beer

autoplot(ts_beer)+ylab("NZ Total Beer Consumption") + xlab("Year")
```

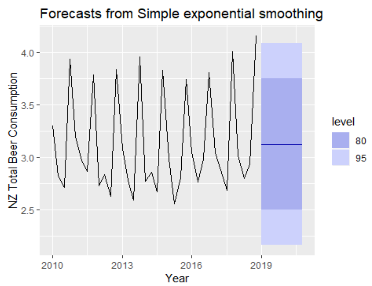


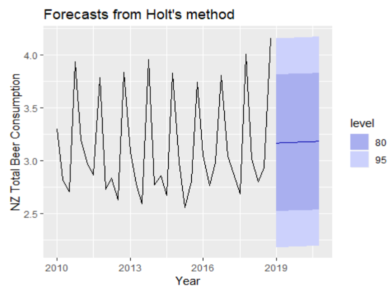
The pattern shows a strong seasonality within each year. It has a low consumption at the beginning of the year and then increase dramatically around the mid-year, and then drop down at the end of every year. Besides, there is no obvious trend and no evidence of any cyclic behaviour here.

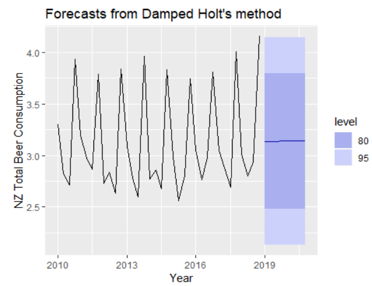
There is no stationarity here because the seasonality will affect the value of the time series at different times, and there is no predictable patterns in the long-term.

(b) Forecast the next two years using (1) simple exponential smoothing, (2) Holt's linear trend, and Holt's (3) damped trend. Plot the series and the forecasts. Merely based on this plot, discuss the adequacy of these methodologies to forecast from this series. Justify your answer (4 marks).

```
#one-step-ahead training data
#forecast two years data h=8
fc1 <- ses(ts_beer, h=8)
fc2 <- holt(ts_beer, h=8)
fc3 <- holt(ts_beer, damped = TRUE, h=8)
#plot the series and forecasts
autoplot(fc1) + ylab("NZ Total Beer Consumption") + xlab("Year")

autoplot(fc2) + ylab("NZ Total Beer Consumption") + xlab("Year")

autoplot(fc3) + ylab("NZ Total Beer Consumption") + xlab("Year")
```







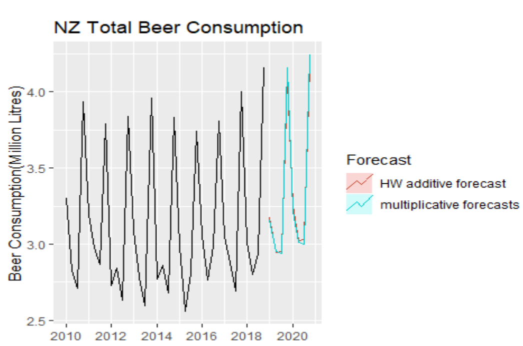
1.We can't find the seasonal or trend variation from the forecast pattern of SES method as it shows a straight flat line across next two years compared to previous fluctuation seen in the previous years. The level is in a wide range from around 2.3 to 4.2 million litres which is even larger than the fluctuation range in the previous years. Therefore, the exponential smoothing method is not suitable for forecasting data with seasonality pattern.

2.We can't find the seasonal or trend variation from the forecast pattern of Holt Linear method although it shows a very small increase on the pattern across next two years compared to SES method, but it is still difficult to get the whole idea of if there is a trend there. The level is in a wide range from around 2.3 to 4.3 million litres which is even larger than the fluctuation range in the previous years. Therefore, the Holt method is not suitable for forecasting this data as there is no trend pattern.

3.Basically the pattern is quite similar to the SES method, and there is no constant trend variation neither. The straight line can't tell the changes among these two years, and the level is in a wide range from around 2.3 to 4.3 million litres which is even larger than the fluctuation range in the previous years. There is no constant trend (increasing or decreasing), so the Holt damped method is not suitable for this data.

(c) Repeat Part (b) with Holt-Winters' seasonal methods. Discuss whether additive or multiplicative seasonality is necessary. Justify your answer (4 marks).

```
#forecast two years data h=8
fit1 <- hw(ts_beer, seasonal = "additive", h=8)
fit2 <- hw(ts_beer, seasonal = "multiplicative",h=8)
autoplot(ts_beer)+
  autolayer(fit1,series="HW additive forecast",PI=FALSE)+
  autolayer(fit2,series ="multiplicative forecasts",PI=FALSE)+
  xlab("Year")+
  ylab("Beer Consumption(Million Litres)")+
  ggtitle("NZ Total Beer Consumption")+
  guides(colour=guide_legend(title="Forecast"))
```



The additive method is preferred when the seasonal variations are roughly constant through the series, while the multiplicative method is preferred when the seasonal variations are changing proportional to the level of the series.

The original pattern shows a roughly constant seasonal variation pattern from 2010 to 2016, and then a slightly changing proportional increase from 2016 to the end of 2018.

After the plot, we can also find out that the additive and the multiplicative forecasts are quite macthed, but multiplicative method can catch the increased trend compared to the original data, so I would say these two methods are necessary due to the roughly constant variation and a slightly increased seasonal variations since 2016.

(d) Compare the mean squared error (MSE) and the mean absolute error (MAE) of the one-step-ahead and four-step-ahead forecasts from the above methods in (b)-(c). You must report your results in a Table (see, e.g., Lab-Question 3, Week 8- Monday). Comment on the adequacy of these methodologies towards forecasting. Which method appears as more accurate to forecast this time series? Does this selection depend on the number of pre-specified (steps-ahead) forecasts? Justify your answer (4 marks).

```
#one-step-ahead forecasts
e1 <- tsCV(ts_beer,ses,h=1)
e2 <- tsCV(ts_beer,holt,h=1)
e3 <- tsCV(ts_beer,holt,damped=TRUE,h=1)
e4 <- tsCV(ts_beer, hw, seasonal="additive",h=1)
e5 <- tsCV(ts_beer, hw, seasonal="multiplicative",h=1)

#MSE
mean(e1^2,na.rm = TRUE)

mean(e2^2,na.rm = TRUE)

mean(e3^2,na.rm = TRUE)

mean(e4^2,na.rm = TRUE)

mean(e5^2,na.rm = TRUE)

#MAE
mean(abs(e1),na.rm = TRUE)

mean(abs(e2),na.rm = TRUE)

mean(abs(e3),na.rm = TRUE)

mean(abs(e4),na.rm = TRUE)

mean(abs(e5),na.rm = TRUE)

#four-step-ahead forecasts
error1 <- tsCV(ts_beer,ses,h=4)
error2 <- tsCV(ts_beer,holt,h=4)
error3 <- tsCV(ts_beer,holt,damped=TRUE,h=4)
error4 <- tsCV(ts_beer,hw,seasonal="additive",h=4)
error5 <- tsCV(ts_beer,hw,seasonal="multiplicative",h=4)
#MSE
mean(error1^2,na.rm = TRUE)

mean(error2^2,na.rm = TRUE)
```

```r
mean(error3^2,na.rm = TRUE)

mean(error4^2,na.rm = TRUE)

mean(error5^2,na.rm = TRUE)

#MAE
mean(abs(error1),na.rm = TRUE)

mean(abs(error2),na.rm = TRUE)

mean(abs(error3),na.rm = TRUE)

mean(abs(error4),na.rm = TRUE)

mean(abs(error5),na.rm = TRUE)
```

ONE STEP AHEAD					
	SES	HOLT LINEAR	HOLT DAMPED	HOLT SEASONAL ADDITIVE	HOLT SEASONAL MULTIPLICATIVE
MSE	0.282	0.477	0.482	0.072	0.070
MAE	0.441	0.528	0.527	0.191	0.178
FOUR STEPS AHEAD					
MSE	0.247	0.866	0.872	0.039	0.037
MAE	0.415	0.574	0.575	0.147	0.140

According to one-step-ahead forecasts, we can discover the Holt-Winter's multiplicative seasonal model is the most suitable modelling framework because it has lowerest forecast value for MAE and MSE, which indicated the forecast error is the smallest.

Similarly, the Holt-Winter's multiplicative seasonal model has the smallest MAE and MSE for four-steps-forecast, which indicated the forecast error is the smallest. Overall, the Holt Seasonal Multiplicative model is the most ideal.

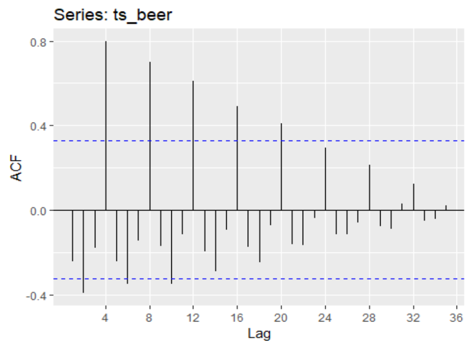
2. Stationarity (4 marks)

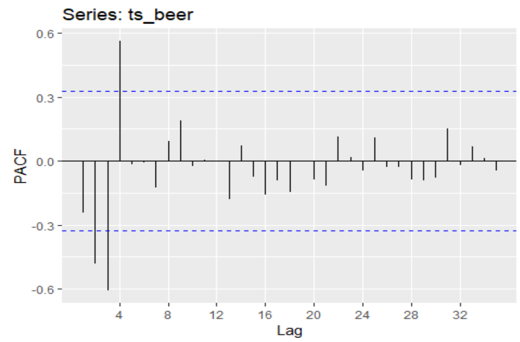
(a) Plot the autocorrelation function (ACF) and the partial ACF (PACF), and (a) discuss the stationarity of the series. Does your answer here conform with your answer in Question 1 - (a)? (b) Should the series be differenced? Justify your answer (2 marks).

```r
ggAcf(ts_beer,lag.max = 100)
ggPacf(ts_beer,lag.max = 100)
```

(a) This is signs of non-stationary. For a stationary time series, the ACF will drop to zero relatively quickly, but this ACF plot show the lags decreases slowly. Also, The ACF and PACF graphs suggest differencing of the data is needed.

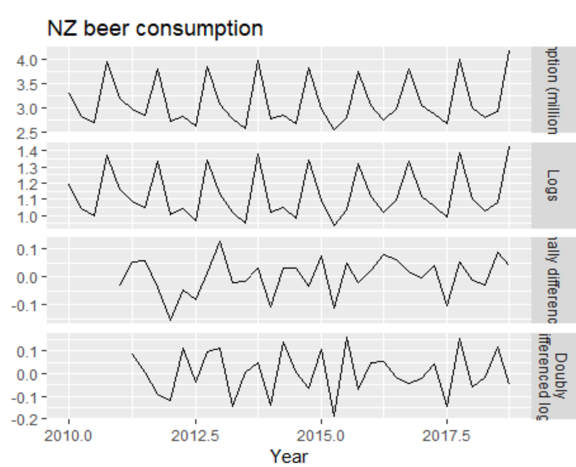
(b) The series should be differenced (because of non-stationary) to apply ARIMA models.





(b) Find an appropriate Box-Cox transformation and order of dfferencing to obtain stationary data (2 marks). Note: Justify your answer whatsoever, even if no BoxCox transformation is needed.

```
#use nsdiffs() to find an appropriate order of differencing
ts_beer%>%log()%>%nsdiffs()

## [1] 1

ts_beer%>%log()%>%diff(lag=4)%>%nsdiffs()

## [1] 0

cbind("Consumption (millions Litres)" = ts_beer,
      "Logs" = log(ts_beer),
      "Seasonally differenced logs" = diff(log(ts_beer),4),
      "Doubly\n differenced logs" = diff(diff(log(ts_beer),4),1)) %>%
  autoplot(facets=TRUE) +
    xlab("Year") + ylab("") +
    ggtitle("NZ beer consumption")
```



Yes, a log transformation is needed to remove the seasonality, and first seasonal difference are needed to obtain stationality.

Because nsdiffs() returns 1 (indicating one seasonal difference is required), we apply the nsdiffs() function to the seasonally differenced data. These functions suggest we should do a seasonal difference but don't need to do a further first difference of seasonal difference. Furthermore, this can be proved on the plot, the double differenced logs are resemble to the seasnally differenced logs, so it is not needed.

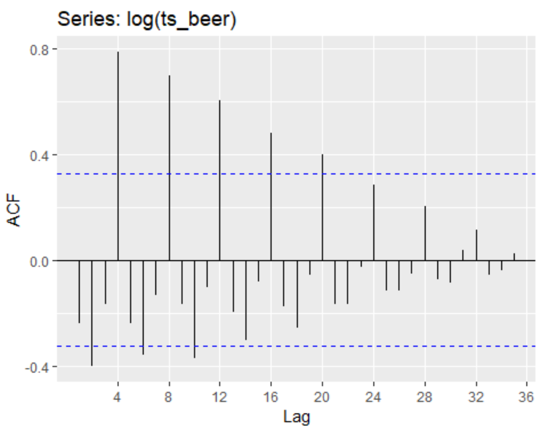
---

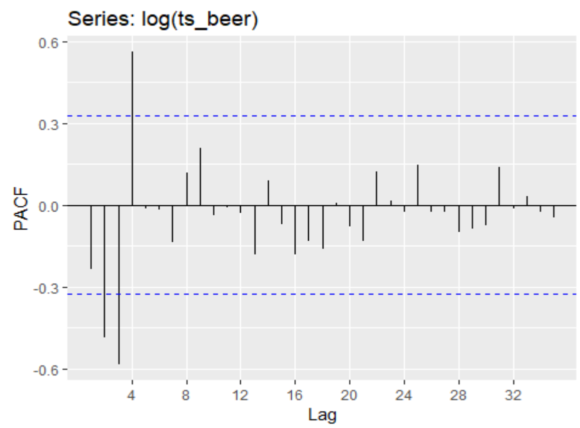
3. ARIMA (12 marks)

(a) By studying the appropriate graphs of the series in R, propose an appropriate ARIMA(p, d, q) structure to model the series. Justify your answer (1 mark).

From the previous ACF and PACF plot, it suggested to difference the data before applying ARIMRA model, and I used first seasonal difference for stablise the series. This indicated that I should use seasonal ARIMA model.However, as the question is specified to use the non season ARIMA model, so I propose a non season ARIMA model based on the ACF and PACF after log transformation.

```r
ggAcf(log(ts_beer),lag.max = 100)

ggPacf(log(ts_beer),lag.max = 100)
```





The data follow an ARIMA(0,d,q) model as the ACF and PACF plots of the log transformed data show the following patterns:

- The PACF is exponentially decaying or sinusoidal;

- There are significant spikes on lag 4, 8, 12 ,16, 20 in the ACF, while in the PACF, there is one significant spike on lag 4, and then no significant spikes thereafter.

So in this case, the ACF, PACF lead me to think an ARIMA(4,0,0) model might be appropriate. As there is no first difference, so the d=0.

(b) Should a constant be included in the model? Explain (1 mark).

A constant is not included because d=1<2

(c) Write the proposed model using backshift notation (1 mark).

Equation can be written in backshift notation as

$$(1-\phi_1 B-\cdots-\phi_p B^p) \quad (1-B)^d y_t \quad = \quad c+(1+\theta_1 B+\cdots+\theta_q B^q)\varepsilon_t$$

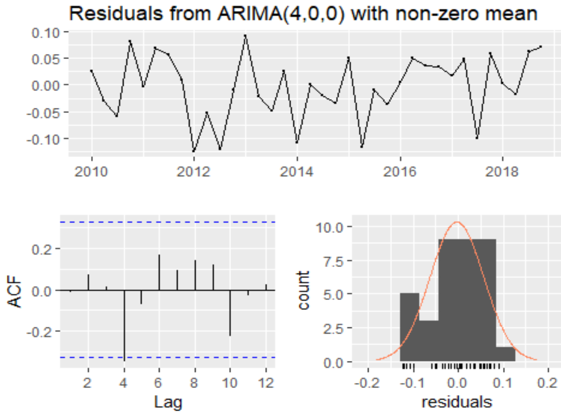
Because constant is not included, so c=0

For Arima(4,0,0), the backshift notation can be written as:

$$(1-\phi_1 B-\phi_2 B^2-\phi_3 B^3-\phi_4 B^4)y_t=\varepsilon_t$$

(d) Fit the model using R functions and examine the residuals. Is the proposed model satisfactory? Justify your answer (2 marks).

```r
(model<- Arima(BoxCox(ts_beer,lambda=0), order=c(4,0,0)))
checkresiduals(model)
```




```
Series: BoxCox(ts_beer, lambda = 0)
ARIMA(4,0,0) with non-zero mean

Coefficients:
         ar1      ar2      ar3     ar4    mean
     -0.2373  -0.2305  -0.1651  0.7207  1.1284
s.e.  0.1226   0.1187   0.1212  0.1173  0.0106

sigma^2 estimated as 0.004017:  log likelihood=47.98
AIC=-83.96   AICc=-81.07   BIC=-74.46

        Ljung-Box test

data:  Residuals from ARIMA(4,0,0) with non-zero mean
Q* = 8.2261, df = 3, p-value = 0.04156

Model df: 5.   Total lags used: 8
```

The first plot shows stationary, although it seems to have a cycle there but as the period of the cycle is uncertain, so we can say it is stationary.

The ACF shows white noise although lag 4 seems a little bit out of the blue line, but it is within the 95% level, so we can say there is white noise there.

The histogram doesn't seem ideal centered normal distributed, and the p value is small less than 0.05.

To conclude, the ARIMA(4,0,0) model can be used for forecasting, but it may also affect the coverage probability of the prediction intervals

(e) Forecast four periods ahead. Check your forecasts by hand to make sure you know how they have been calculated (2 marks). HINT: See https://otexts.com/fpp2/arima-forecasting.html .

To compute the forecast manually, we need the ARIMA(4,0,0)written in terms of the backshift operator:

$$(1-\phi_1 B-\phi_2 B^2-\phi_3 B^3-\phi_4 B^4)y_t=\varepsilon_t$$

This can be rewritten as:

$$y_t=\phi_1 y_{t-1}+\phi_2 y_{t-2}+\phi_3 y_{t-3}+\phi_4 y_{t-4}+\varepsilon_t$$

For effects of forecasting, we replace the true errors with residuals, giving place to:

$$\hat{y}_{T+h|T}=\phi_1 \hat{y}_{(T+h)-1}+\phi_2 \hat{y}_{(T+h)-2}+\phi_3 \hat{y}_{(T+h)-3}+\phi_4 \hat{y}_{(T+h)-4}+\varepsilon_{T+h}$$

When $h=1$

$$\hat{y}_{T+1|T}=\phi_1 y_T+\phi_2 y_{T-1}+\phi_3 y_{T-2}+\phi_4 y_{T-3}+\varepsilon_{T+1} \quad (\varepsilon_{T+1} \text{ replace with zero})$$

When $h=2$

$$\hat{y}_{T+2|T}=\phi_1 y_{T+1}+\phi_2 y_T+\phi_3 y_{T-1}+\phi_4 y_{T-2}+\varepsilon_{T+2} \quad (\varepsilon_{T+2} \text{ replace with zero})$$

When $h=3$

$$\hat{y}_{T+3|T}=\phi_1 y_{T+2}+\phi_2 y_{T+1}+\phi_3 y_T+\phi_4 y_{T-1}+\varepsilon_{T+3} \quad (\varepsilon_{T+3} \text{ replace with zero})$$

When $h=4$

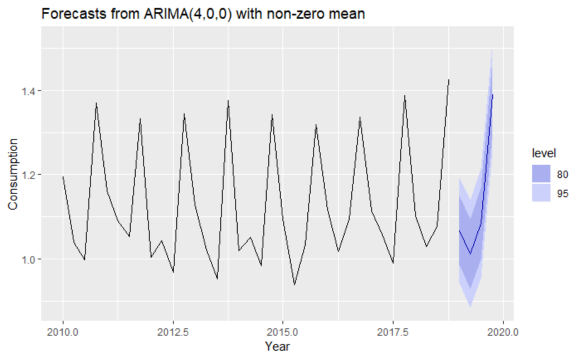
$$\hat{y}_{T+4|T}=\phi_1 y_{T+3}+\phi_2 y_{T+2}+\phi_3 y_{T+1}+\phi_4 y_T+\varepsilon_{T+4} \quad (\varepsilon_{T+4} \text{ replace with zero})$$

```r
#we use lambda=0 as we use log transformation to stablize the series above
myfit <- Arima(BoxCox(ts_beer,lambda = 0),order=c(4,0,0))
phicoef <- coefficients(myfit)
n <- length(ts_beer)
#Forecasting for h=1
(yt1 <- c(ts_beer)[n]*phicoef[1]+ts_beer[n-1]*phicoef[2]+ts_beer[n-2]*phicoef[3]+ts_beer[n-3]*phicoef[4])

fcs <- yt1
#update the vector of forecasts

#Forecasting for h=2
(yt2 <- yt1*phicoef[1]+ts_beer[n]*phicoef[2]+ts_beer[n-1]*phicoef[3]+ts_beer[n-2]*phicoef[4])

fcs <- c(fcs,yt2)

#Forecasting for h=3
(yt3 <- yt2*phicoef[1]+yt1*phicoef[2]+ts_beer[n]*phicoef[3]+ts_beer[n-1]*phicoef[4])

fcs <- c(fcs,yt3)

(yt4 <- yt3*phicoef[1]+yt2*phicoef[2]+yt1*phicoef[3]+ts_beer[n]*phicoef[4])
```

```
fcs <- c(fcs,yt4)
fcs

##         ar1        ar1        ar1        ar1
## 0.04564909 0.56384894 1.28598972 2.55389420
```

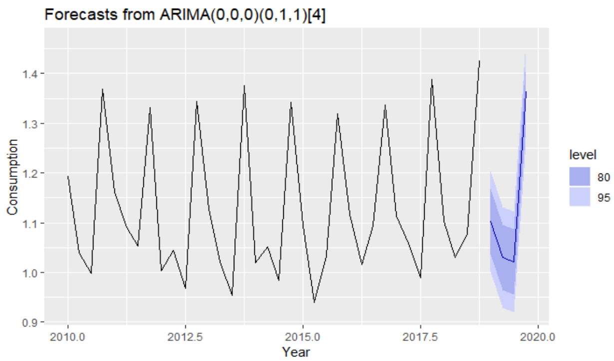
(f) Create a plot of the series with forecasts and prediction intervals for the four forecasted periods (1 mark).

```
model %>% forecast(h=4) %>% autoplot ()
```



(g) Now, let auto.arima() choose an ARIMA structure. Does auto.arima return the same model (the one you chose)? If not, which model do you think is better? Justify your answer (2 mark).

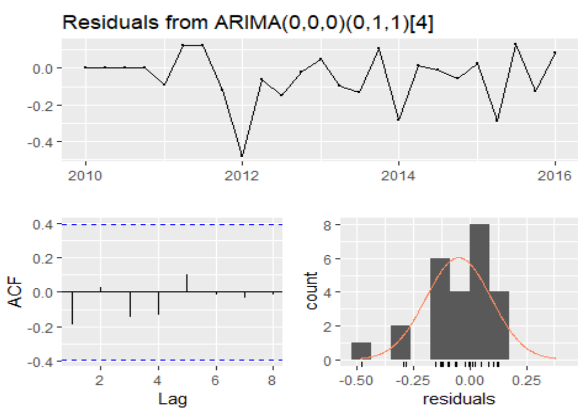
```
fit <- auto.arima(BoxCox(ts_beer,lambda=0))

fit

## Series: BoxCox(ts_beer,lambda=0)
## ARIMA(0,0,0)(0,1,1)[4]
##
## Coefficients:
##          sma1
##       -0.8671
## s.e.   0.4474
##
## sigma^2 estimated as 0.002578:  log likelihood=47.85
## AIC=-91.7   AICc=-91.28   BIC=-88.76

fit %>% forecast(h=4) %>% autoplot(include=80)+
  xlab("Year") + ylab("Consumption")
```

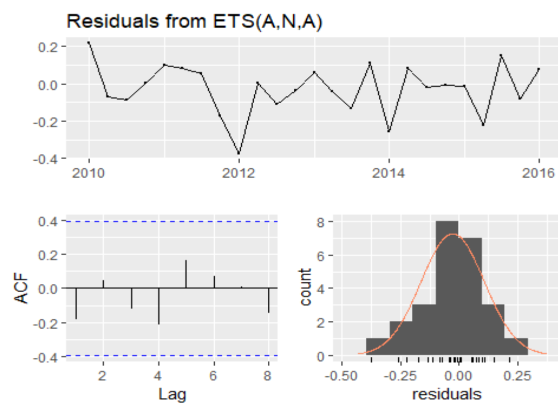


auto.arima doesn't return as the same result as me because auto.aria apply seasonal ARIMA model which I didn't use, but it turns out the AIC for ARIMA (0,0,0)(0,1,1)4 is lower than ARIMA (4,0,0) which indicates the Arima(0,0,0)(0,1,1)4 is a better model. By looking at the plot, the pattern is similar except ARIMA (0,0,0)(0,1,1)4 consider the trough will last for a few months, but they both catch the seasonal fluctuations.

(h) Which method do you think is best between ETS and ARIMA? (2 mark).

```
beerset <- ts(data[,2],start=2010,end=2016,frequency = 4)
fit.arima <- auto.arima(beerset)
checkresiduals (fit.arima)

fit.ets <- ets(beerset)
checkresiduals(fit.ets)
```





Although we can both find the stationary from the first plot, and white noise drom the ACF plots, the residuals of ETS model seems more central normal distributed than the ARIMA model.

According to the RMSE, MAE,MAPE, we can say the ARIMA model provides more accurate forecasts on the test set as it has the lower value than ETS model.

```
a1 <- fit.arima %>% forecast(h = 4*(2018-2016)) %>%
  accuracy(ts_beer)
a1[,c("RMSE","MAE","MAPE","MASE")]

##                   RMSE        MAE     MAPE      MASE
## Training set 0.1490463 0.10292982 3.550366 0.6287162
## Test set     0.1224515 0.08631531 2.715106 0.5272314

a2 <- fit.ets %>% forecast(h = 4*(2018-2016)) %>%
  accuracy(ts_beer)
a2[,c("RMSE","MAE","MAPE","MASE")]

##                   RMSE        MAE     MAPE      MASE
## Training set 0.1352571 0.10308701 3.506419 0.6296763
## Test set     0.1330039 0.08876844 2.782866 0.5422156
```