Forecasting: Exam assignment

Chenxin XIE | IÉSEG School of Management

Exercise 1:

Prepare data

- Load the data and transfer to time series:

```
data <- read_excel("DataSets2020.xlsx", sheet="Turnover")

turnover <- ts(data$Turnover, frequency=12, start = c(2000, 1), end = c(2020, 1))
```

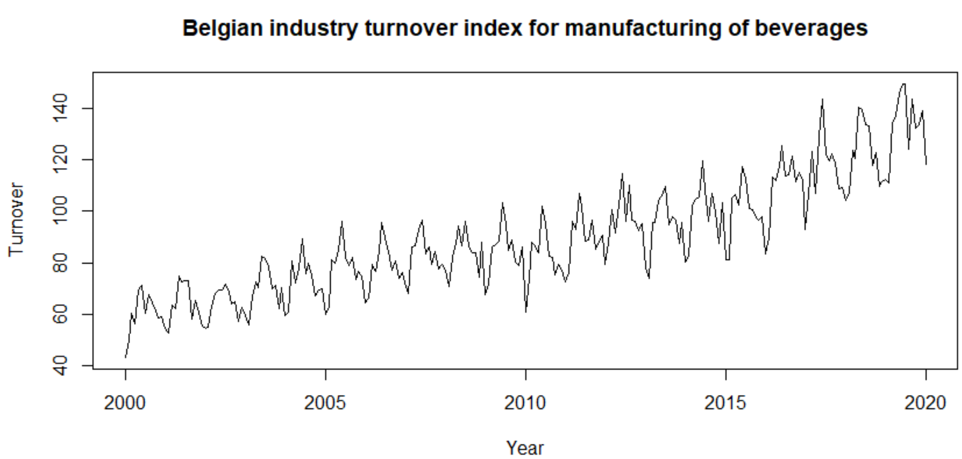
- Split train and test as required:

```
train <- window(turnover, start = c(2000, 1), end = c(2015, 12))

test <- window(turnover, start = c(2016, 1), end = c(2020, 1))
```

Question 1: Explore the data using relevant graphs, and discuss the properties of the data.

After transfer the data to time series, we want to have an overview of the time series:

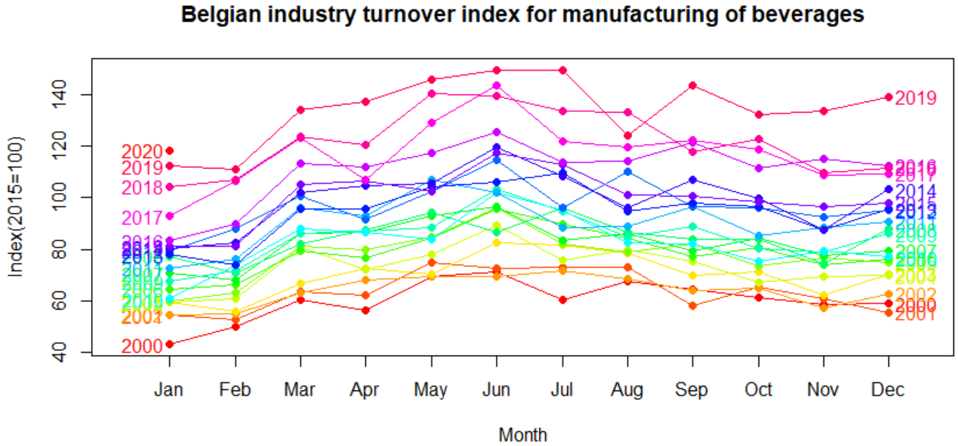
```
plot(turnover, main = 'Belgian industry turnover index for manufacturing of beverages',
xlab ='Year')
```



We can observe a seasonal pattern and an increasing trending pattern from the plot of this time series.

Since we have seen a seasonal pattern in the time series, we would want to know about the seasonal pattern for this time series:

```
seasonplot(turnover, year.labels=TRUE, year.labels.left=TRUE,
    main="Belgian industry turnover index for manufacturing of beverages",
    ylab="Index(2015=100)",col=rainbow(20), pch=19)
```

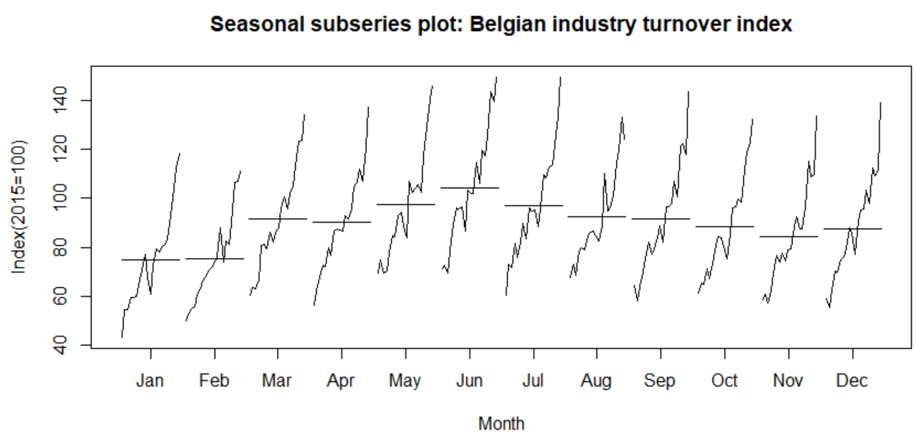


From the seasonal plot, we can observe the seasonal pattern in each year one on top of each other because of the increasing trending and we can also see that the variation increasing over time. The fluctuation is larger of the earlier years and later years than those in between.

Because the data is up till Jan 2020, so the plot for 2020 is only a dot on January.

Then we can look deeper into the seasonal pattern in each month:

```
monthplot(turnover, ylab="Index(2015=100)", xlab="Month", xaxt="n",
    main="Seasonal subseries plot: Belgian industry turnover index", type="l")
axis(1, at=1:12, labels=month.abb, cex=0.8)
```

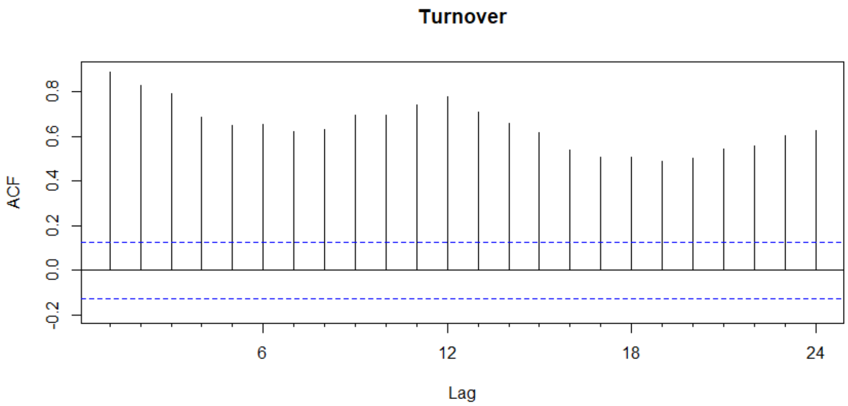


From this month plot, data of each month is being grouped and plotted as a time series, with the horizontal lines of mean level. We can observe that the mean of each month has gone up and down, where June hits the highest mean. Otherwise, we can see an increasing trend over time for all these months.

Question 2: Discuss whether a transformation of the data would be useful. If so, select the most appropriate transformation.

Frist we will look at the autocorrelation of this time series:

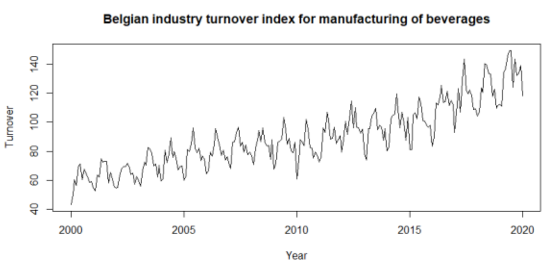
Acf(turnover)

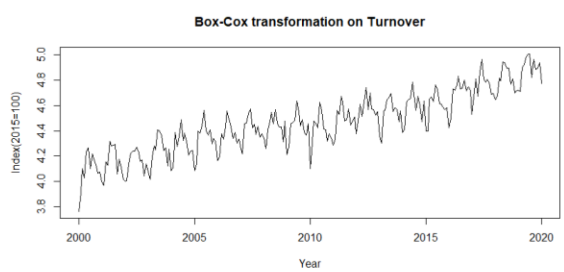


We can observe the ACF peaks at lag 1, 12, 24 and wave between seasonal lags, which means it is a seasonal monthly time series. As we look at the blue lines indicate the significant of P-value to 0, and all the spikes above the line. It means the autocorrelations are all significant. Thus, it shows there is a trend over time in this time series. Otherwise, we can tell this is not a white noise series, means that this data can be used to forecast future values.

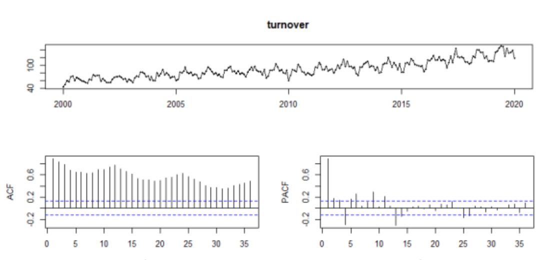
Then we can look at the variation of seasonality over time. From the plot of the data, we can observe a mild variation of seasonality over time. To adjust the data with Box-Cox transformation by setting lambda equals to 0, which means taking the logarithm on the data, we can see the shape of the data plotted below:

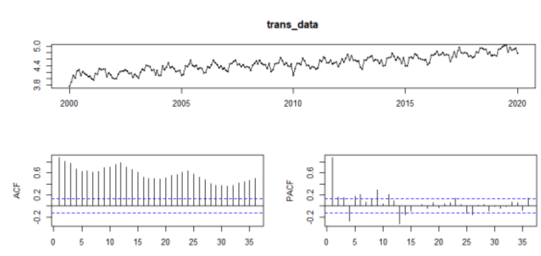
trans_data <- BoxCox(turnover, lambda = 0)





From the two plots, we can see after the Box-Cox transformation, the variation of seasonality looks smaller and the trend seems a bit smoother.





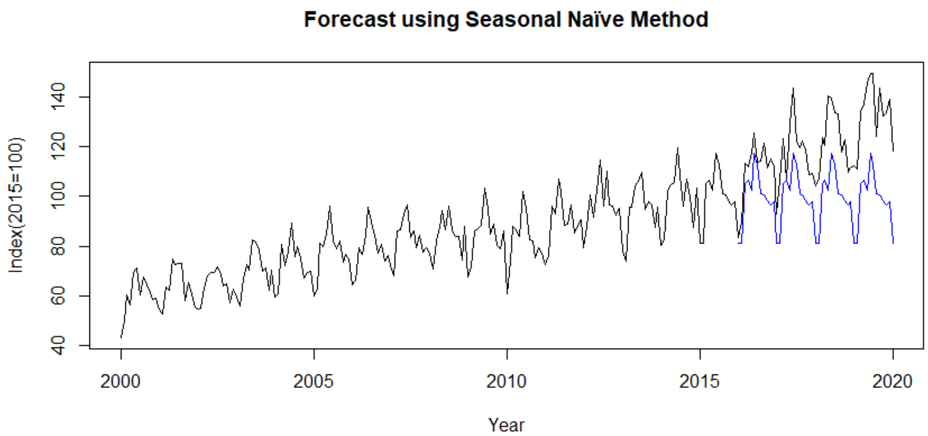
If we look at the autocorrelation of the data and transformed data, the autocorrelation does not change.

It is not always useful to do a Box-Cox transformation, it depends on the data and the model we use.

Question 3: Create forecasts using the seasonal naive method. Check the residual diagnostics and the forecast accuracy.

First we fit the seasonal naïve on the train data

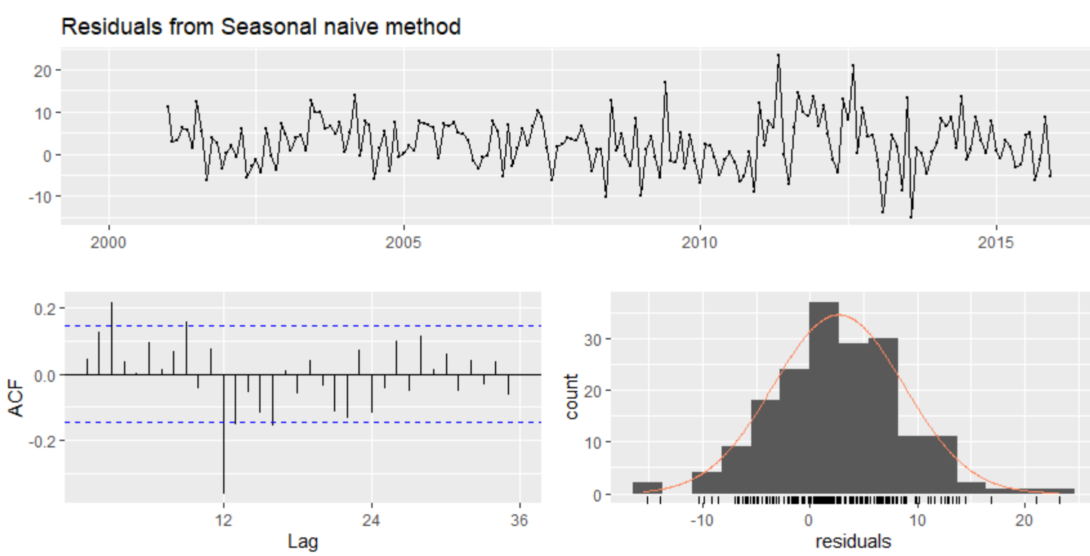
```
sn_f <- snaive(train, h = h)

plot(turnover, main = 'Forecast using Seasonal Naïve Method',
   ylab="Index(2015=100)", xlab="Year")

lines(sn_f$mean,col=4)
```



We can see the forecast of train data using seasonal naïve method compared with the test data. We can see the blue line is the fitted value and we can see the differences of the fitted values and the true values are obvious. We need to check the accuracy and the residuals to see how far are they from the true values.

First, we check the residuals of the fitted values:

```
checkresiduals(sn_f)
```



```
        Ljung-Box test

data:  Residuals from Seasonal naive method
Q* = 70.988, df = 24, p-value = 1.548e-06

Model df: 0.   Total lags used: 24
```

From the plot of the residuals from the seasonal naïve method, we can see the residuals fluctuate up and down around 0.

From the ACF of the residuals, we can see except for spikes at lag 3 and 12, the residuals look like white noise. It means there is no significant correlation between the residuals.

As we look at the distribution of the residuals, it looks pretty close to a normal distribution.

Then we look at the result of Ljung-Box test, the p-value is below 0.05, which means that we will reject the null hypotheses that this is a white noise procedure. Therefore, there are correlation between the residuals. It means that with this model we did not capture the complete data generating procedure, and there are some information from the residuals that did not be captured in this model that would be reflected on the forecast.

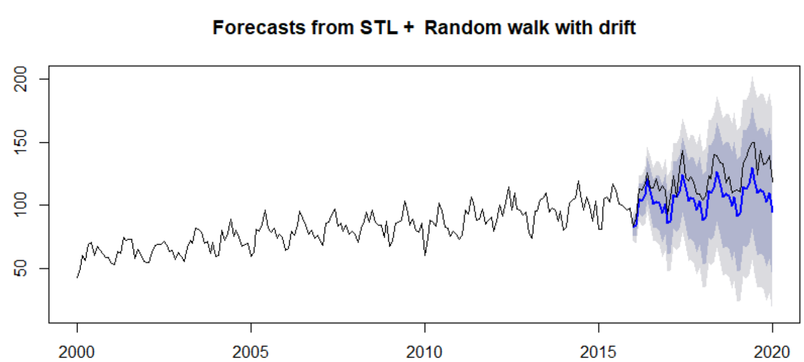
Now we check the accuracy:

```
> accuracy(sn_f,test)
                    ME      RMSE       MAE      MPE     MAPE     MASE       ACF1 Theil's U
Training set  2.659611  6.612143  5.158722  3.09471  6.23105 1.000000 0.04369396        NA
Test set     21.338367 24.051192 21.338367 17.11021 17.11021 4.136367 0.52611586  1.926643
```

If we look at the RMSE, MAE and MASE, the accuracy of the test set is not good enough, but we do not have other accuracy of other models to compare.

Question 4: Use an STL decomposition to forecast the turnover index. Use the appropriate underlying methods to do so. Check the residual diagnostics and the forecast accuracy.

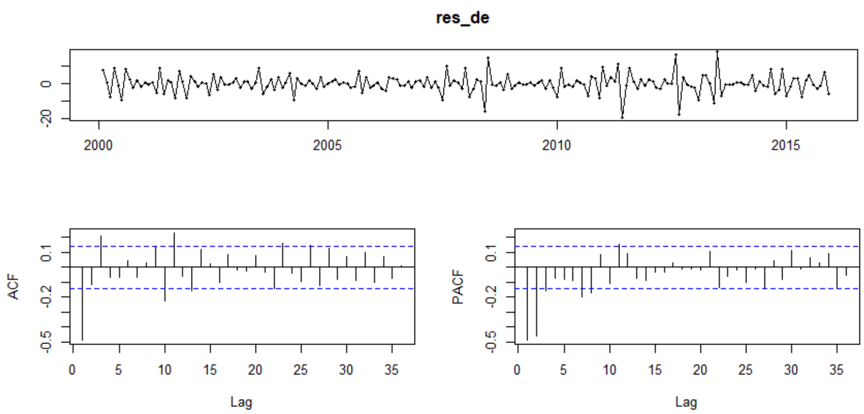
```
stl_de <- stl(train, s.window = 5, robust=TRUE)

fcast_de <- forecast(stl_de, method="rwdrift", h=h)

plot(fcast_de)

lines(turnover)
```



After we use STL to decompose the train data, then forecast with random walk with drift. The blue line and grey area are the fitted values. The darker grey area is the 80% forecast intervals, where should contain 80% of future observations and we expect 20% of the future observations outside that area. The lighter grey and boarder area is the 95% forecast intervals, where should contain 95% of future observations and we expect 5% of the future observations outside that area.

Then we have a look at the residuals:

```
res_de <- residuals(fcast_de)

tsdisplay(res_de)

Box.test(res_de, lag=10, fitdf=0, type="Lj")
```




```
        Box-Ljung test

data:  res_de
X-squared = 74.678, df = 10, p-value = 5.496e-12
```

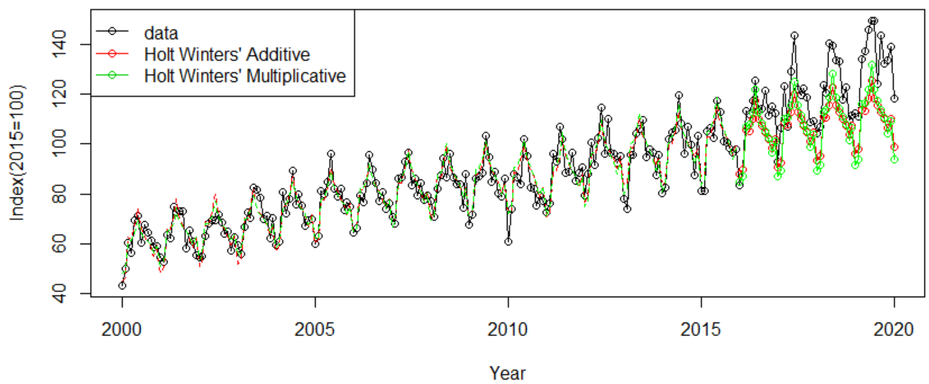
From the plot of the residuals, we can see the residuals fluctuate up and down around 0. From the ACF, we can see except for some spikes, the residuals look like white noise. It means there is no significant correlation between the residuals. But, from the Ljung-Box test, the p-value is below 0.05, means the difference is significant. Thus, we can reject the null hypotheses that this is a white noise procedure. It means that the model still need to be improved.

```
> accuracy(fcast_de,test)
                      ME      RMSE       MAE        MPE      MAPE      MASE       ACF1 Theil's U
Training set -3.42241e-15  5.187418  3.599768 -0.1294216  4.483651 0.6978023 -0.4871314        NA
Test set      1.53254e+01 17.322163 15.325395 12.2741012 12.274101 2.9707735  0.3556888  1.390659
```

From the MASE we can see the accuracy on the train data is very good, even lower than 1, and the accuracy on the test data looks not bad.

Question 5: Generate forecasts using Holt-Winters' method. Check the residual diagnostics and the forecast accuracy.

```
fcast_holtM <- hw(train, seasonal="mult", h=h)

fcast_holtA <- hw(train, seasonal="additive", h=h)
```



If we just look at the graph, we can see the green line, which is the Holt Winters' Multiplicative method, does better at the forecasting than the Holt Winters' Additive method. It means taking the variation of the seasonality into consideration in forecasting in this case can make better forecasting.

But, we need to compare the residuals and accuracy to see whether that is true.

```
res_holtM <- residuals(fcast_holtM)

LjungBox(res_holtM, lags=24, order=length(fcast_holtM$model$par))
```

```
> LjungBox(res_holtM, lags=24, order=length(fcast_holtM$model$par))
 lags statistic df      p-value
   24  93.38013  8 1.110223e-16
```

```
res_holtA <- residuals(fcast_holtA)

LjungBox(res_holtA, lags=24, order=length(fcast_holtA$model$par))
```

```
> LjungBox(res_holtA, lags=24, order=length(fcast_holtA$model$par))
 lags statistic df      p-value
   24  83.16741  8 1.121325e-14
```

Because this is a seasonal time series with seasonality length of 12, so we set lags as twice the length: 24. By setting the order, we are actually setting the degree of freedom. From the p-values for the test of both residuals are below 0.05, therefore we can reject the null hypotheses that this is a white noise procedure for both residuals. The model still need improvement.

Now we compare the accuracy between them:

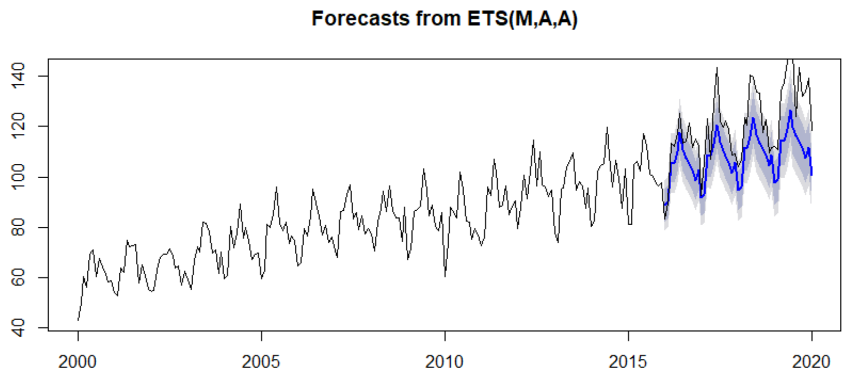
```
> accuracy(fcast_holtM, test)[,c(2,3,5,6)]
                 RMSE       MAE      MAPE      MASE
Training set  4.30480  3.333093  4.152035 0.6461082
Test set     15.46606 13.353027 10.681029 2.5884369
> accuracy(fcast_holtA, test)[,c(2,3,5,6)]
                  RMSE       MAE      MAPE      MASE
Training set  4.389998  3.549698  4.474374 0.6880964
Test set     16.077726 13.911877 10.987226 2.6967680
```

From the RMSE, MAE, MAPE and MASE, we can see the Holt Winters' Multiplicative method has lower score than the Holt Winters' Additive method, which means the Holt Winters' Multiplicative method has higher accuracy than the Holt Winters' Additive method. Thus the information of the variation of seasonality has been captured and made a better forecast on the same data.

Question 6: Generate forecasts using ETS. First, select the appropriate model(s) yourself and discuss their performance. Compare these models with the results of the automated ETS procedure. Check the residual diagnostics and the forecast accuracy for the various ETS models you have considered.

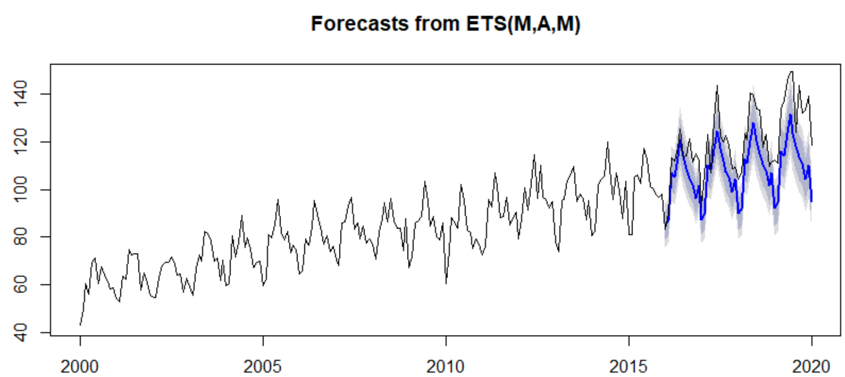
As we have known this is seasonal time series with trend, and the beta was quite low when we used the Holt Winters' Multiplicative method, which means the variation of seasonality was not that significant. Therefore, for Error, we would choose Multiplicative; for Trend, we would choose Additive; for Seasonal, we will choose Additive. Thus, the appropriate model we choose is ETS(M, A, A).

```
fc_MAA <- forecast(ets(train, model="MAA"), h=h)
```



Then we will forecast with the method automatically selected by the ETS procedure.

```
fc_ETS <- forecast(ets(train), h=h)
```

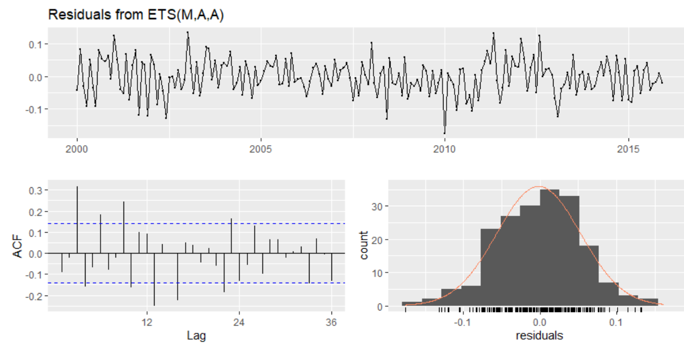



```
> ets(train)
ETS(M,A,M)

Call:
 ets(y = train)

  Smoothing parameters:
    alpha = 0.1048
    beta  = 1e-04
    gamma = 2e-04

  Initial states:
    l = 57.9718
    b = 0.2519
    s = 0.9646 0.9174 0.9758 1.0023 1.0455 1.0962
           1.1673 1.0996 1.0189 1.0376 0.8459 0.8289

  sigma:  0.0545

     AIC     AICc      BIC
1593.263 1596.780 1648.641
```

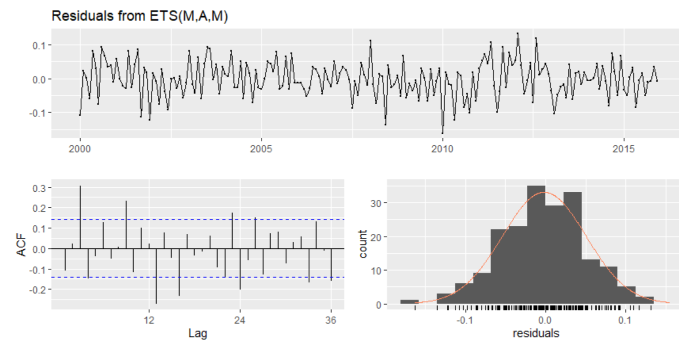
We can see that the method automatically selected by the ETS procedure is the ETS(M,A,M). The method that could get the minimum AIC_C is the Multiplicative Holt-Winters' method with multiplicative errors.

To see whether the ETS method I chose is better or the ETS automatically selected method is better, we are going to check the residuals and accuracy.

```
> checkresiduals(fc_MAA)

        Ljung-Box test

data:  Residuals from ETS(M,A,A)
Q* = 101.02, df = 8, p-value < 2.2e-16

Model df: 16.   Total lags used: 24
```




```
> checkresiduals(fc_ETS)

        Ljung-Box test

data:  Residuals from ETS(M,A,M)
Q* = 98.699, df = 8, p-value < 2.2e-16

Model df: 16.   Total lags used: 24
```



From the results of Ljung-Box test for the two methods, the p-values are both smaller than 0.05, which means we can reject the null hypotheses that this is a white noise procedure for both residuals. Thus, the residuals for ETS (M,A,A) and ETS (M,A,M) are not white noises. It means there is still some correlation need to be captured and the model needs to be improved.

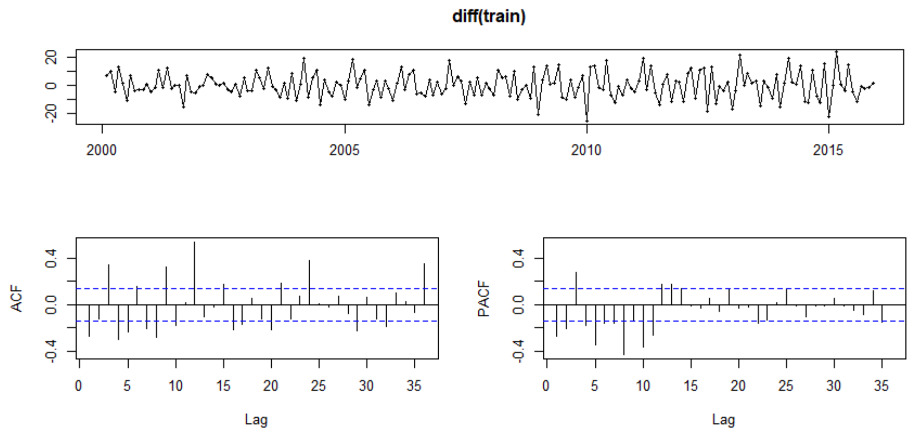
```
> accuracy(fc_MAA, test)[,c(2,3,5,6)]
                  RMSE      MAE      MAPE      MASE
Training set  4.395102  3.53572  4.464581 0.6853868
Test set     15.317092 13.16739 10.385782 2.5524517
> accuracy(fc_ETS, test)[,c(2,3,5,6)]
                  RMSE       MAE      MAPE      MASE
Training set  4.288583  3.367403  4.193484 0.6527591
Test set     15.164648 13.060222 10.444933 2.5316777
```

From the RMSE, MAE, MAPE and MASE, the accuracies for the two methods are very similar. But we can see that the ETS (M,A,M) chose by the ets() performs slightly better on than EST(M,A, A) that I chose.

Question 7: Generate forecasts using ARIMA. First, select the appropriate model(s) yourself and discuss their performance. Compare these models with the results of the auto.arima procedure. Check the residual diagnostics and the forecast accuracy for the ARIMA models you have considered.

First, this time series is not a stationary series, so we will taking 1 differencing. Then we will have a look at the autocorrelation to decide the number of terms of auto regression and terms of moving average.

```
tsdisplay(diff(train))
```



By looking at the spikes in ACF and PACF, there are 3-4 spikes that are much longer. At the autocorrelation we can observe quick drop after each high spikes.

We will try 3 methods first: an auto ETS method, a non-seasonal ARIMA(0,1,0) with drift and a seasonal ARIMA(1,0,0)(0,1,3)$_{12}$ with drift. Then we will use the auto ARIMA method that selects the ARIMA with the smallest AIC$_C$.

```
fc_ets <- forecast(ets(train, lambda=0), h=h)

 ARIMA_1 <- Arima(train, lambda=0, order=c(0,1,0), include.drift=TRUE)

ARIMA_2 <- Arima(train, lambda=0, order=c(1,0,0), seasonal=c(0,1,3),
include.drift=TRUE)

ARIMA_auto <- auto.arima(train, lambda = 0, allowdrift = TRUE)
```

```
> summary(ARIMA_auto)
Series: train
ARIMA(1,0,1)(2,1,2)[12] with drift
Box Cox transformation: lambda= 0

Coefficients:
         ar1      ma1    sar1     sar2    sma1    sma2   drift
      0.9867  -0.8182  0.6514  -0.3823  -1.546  0.7263  0.0031
s.e.  0.0146   0.0467  0.1360   0.0911   0.162  0.1558  0.0009

sigma^2 estimated as 0.002921:  log likelihood=261.19
AIC=-506.37   AICc=-505.53   BIC=-480.83

Training set error measures:
                   ME     RMSE     MAE       MPE     MAPE      MASE        ACF1
Training set -0.19898 4.238087 3.21202 -0.374753 3.913788 0.6226386 -0.1100893
```

The auto selected ARIMA is ARIMA(1,0,1)(2,1,2)$_{12}$ with drift.

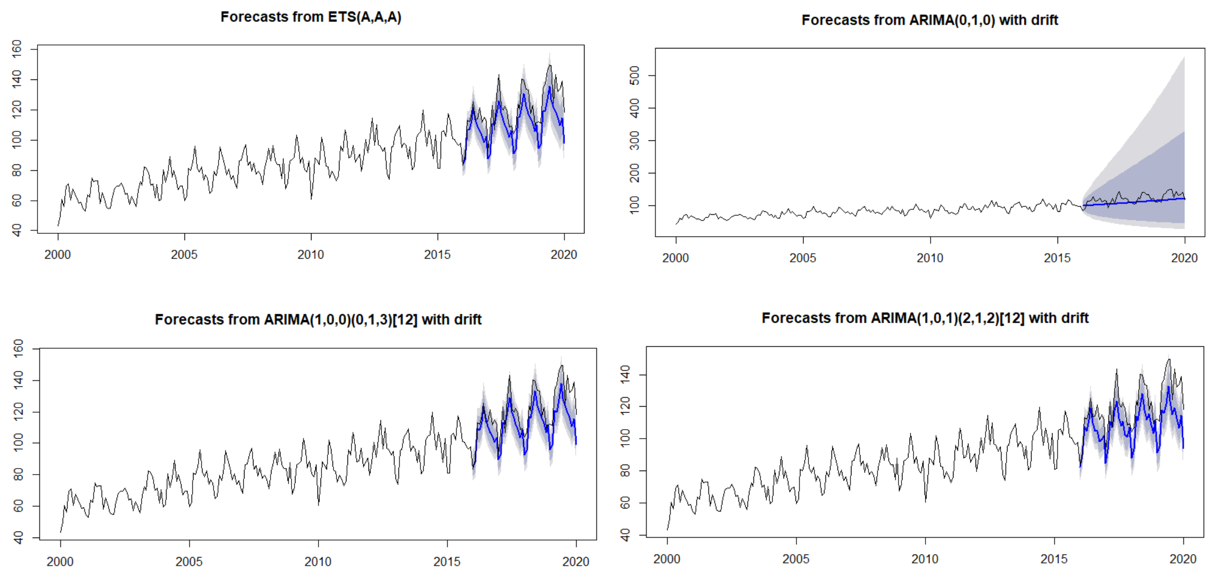
The minimum AIC$_C$ is -505.53.

Because we can not compare the AIC$_C$ between ETS and ARIMA, so we are going compare it with the seasonal ARIMA(1,0,0)(0,1,3)$_{12}$ with drift we chose.

```
Forecast method: ARIMA(1,0,0)(0,1,3)[12] with drift

Model Information:
Series: train
ARIMA(1,0,0)(0,1,3)[12] with drift
Box Cox transformation: lambda= 0

Coefficients:
         ar1     sma1     sma2    sma3   drift
      0.1329  -0.7688  -0.2506  0.0194  0.0028
s.e.  0.0785   0.1557   0.1035  0.1026  0.0001

sigma^2 estimated as 0.003386:  log likelihood=244.28
AIC=-476.55   AICc=-476.07   BIC=-457.39

Error measures:
                    ME     RMSE      MAE       MPE     MAPE      MASE        ACF1
Training set 0.4417251 4.567583 3.512183 0.4994677 4.275679 0.6808243 -0.04844832
```

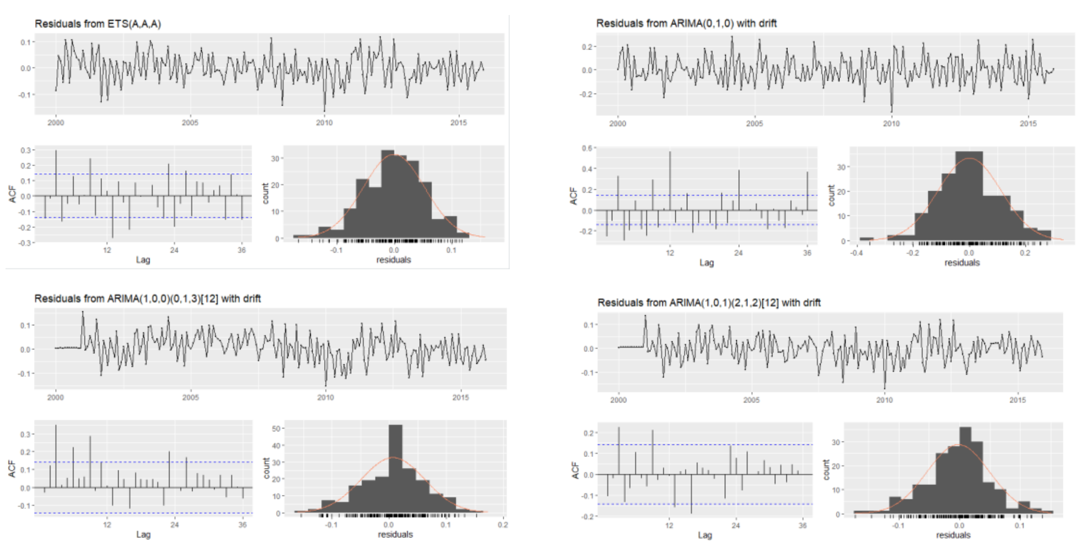
The AIC$_C$ of the is ARIMA(1,0,0)(0,1,3)$_{12}$ is -476.07. The auto ARIMA got a lower AIC$_C$.

Then we are going to compare the forecasting result:



We can see these methods are making good forecast.

We are going to check the residuals:



```
> checkresiduals(fc_ets)

        Ljung-Box test

data:  Residuals from ETS(A,A,A)
Q* = 104.6, df = 8, p-value < 2.2e-16

Model df: 16.   Total lags used: 24
```

```
> checkresiduals(fc_A1)

        Ljung-Box test

data:  Residuals from ARIMA(0,1,0) with drift
Q* = 247.32, df = 23, p-value < 2.2e-16

Model df: 1.   Total lags used: 24
```

```
> checkresiduals(fc_A2)

        Ljung-Box test

data:  Residuals from ARIMA(1,0,0)(0,1,3)[12] with drift
Q* = 81.301, df = 19, p-value = 1.108e-09

Model df: 5.   Total lags used: 24
```

```
> checkresiduals(fc_AUTO)

        Ljung-Box test

data:  Residuals from ARIMA(1,0,1)(2,1,2)[12] with drift
Q* = 51.67, df = 17, p-value = 2.317e-05

Model df: 7.   Total lags used: 24
```

From the result of checking the residuals of these four methods, all the p-values are all smaller than 0.05, therefore, we reject the null hypotheses that this is a white noise procedure for all residuals. It means all these models need to be improved.

Finally, we are going to look at the accuracy:

```
> accuracy(fc_ets, test)[,c(2,3,5,6)]
                 RMSE      MAE     MAPE      MASE
Training set  4.28433  3.39688 4.237747 0.6584731
Test set     12.94274 11.04910 8.848813 2.1418282

> accuracy(fc_A1, test)[,c(2,3,5,6)]
                  RMSE       MAE      MAPE     MASE
Training set  9.081055  7.143836  8.809198 1.384807
Test set     16.884013 14.115244 11.184654 2.736190

> accuracy(fc_A2, test)[,c(2,3,5,6)]
                  RMSE      MAE     MAPE      MASE
Training set  4.567583 3.512183 4.275679 0.6808243
Test set     11.607418 9.554709 7.630307 1.8521465

> accuracy(fc_AUTO, test)[,c(2,3,5,6)]
                  RMSE      MAE      MAPE      MASE
Training set  4.238087  3.21202  3.913788 0.6226386
Test set     14.812742 12.84807 10.263672 2.4905527
```

Frist, we compare the accuracy between the 3 methods we chose. We can see ARIMA(0,1,0) with drift got the worst accuracy. ARIMA(1,0,0)(0,1,3)$_{12}$ with drift got the best accuracy among them. ETS(A,A,A) is just slightly less accurate than the seasonal ARIMA with drift.

Then we compare the auto ARIMA with them, and we can see the auto ETS and did not get a better accuracy than the ARIMA we set. The auto ETS got a better accuracy than the auto ARIMA.

Question 8: Compare the different models in terms of residual diagnostics, model fit, and forecast accuracy. Analyze your results and select your final model.

In this exercise, we have used seasonal naïve method, random walk with drift after STL decomposition, Holt-Winters' method (Multiplicative and Additive), ETS(M,A,A), auto ETS, auto ETS with log transformation, non-seasonal ARIMA with drift, seasonal ARIMA with drift, and auto ARIMA.

To compare the performance of these methods with residual diagnostics and accuracy:

	Method	Ljung-Box Test		Accuracy				
		df	p-value		RMSE	MAE	MAPE	MASE
1	seasonal naïve	24	1.548e-06	Train	6.612143	5.158722	6.23105	1.000000
				Test	24.051192	21.338367	17.11021	4.136367
2	random walk with drift after STL decomposition	23	2.202e-13	Train	5.187418	3.599768	4.483651	0.6978023
				Test	17.322163	15.325395	12.274101	2.9707735
3	Holt-Winters' Multiplicative	8	< 2.2e-16	Train	4.30480	3.333093	4.152035	0.6461082
				Test	15.46606	13.353027	10.681029	2.5884369
4	Holt-Winters' Additive	8	1.121e-14	Train	4.389998	3.549698	4.474374	0.6880964
				Test	16.077726	13.911877	10.987226	2.6967680
5	ETS(M,A,A)	8	< 2.2e-16	Train	4.395102	3.53572	4.464581	0.6853868
				Test	15.317092	13.16739	10.385782	2.5524517
6	auto ETS : ETS(M,A,M)	8	< 2.2e-16	Train	4.288583	3.367403	4.193484	0.6527591
				Test	15.164648	13.060222	10.444933	2.5316777
7	auto ETS with log transformation: ETS(A,A,A)	8	< 2.2e-16	Train	4.28433	3.39688	4.237747	0.6584731
				Test	12.94274	11.04910	8.848813	2.1418282
8	ARIMA(0,1,0) with drift	23	< 2.2e-16	Train	9.081055	7.143836	8.809198	1.384807
				Test	16.884013	14.115244	11.184654	2.736190
9	ARIMA(1,0,0)(0,1,3)$_{12}$ with drift	19	1.108e-09	Train	4.567583	3.512183	4.275679	0.6808243
				Test	**11.607418**	**9.554709**	**7.630307**	**1.8521465**
10	auto ARIMA: ARIMA(1,0,1)(2,1,2)12 with drift	17	2.317e-05	**Train**	**4.238087**	**3.21202**	**3.913788**	**0.6226386**
				Test	14.812742	12.84807	10.263672	2.4905527

For the comparison on the residual diagnostics, we can refer to the results from the Ljung-Box Test. The results from all the methods we have fitted, did not give the desirable p-value. Because after fitting the model, the residuals should be independent, which means there is no information that has not been captured yet. But, looks like all the results showed the residuals are dependent.

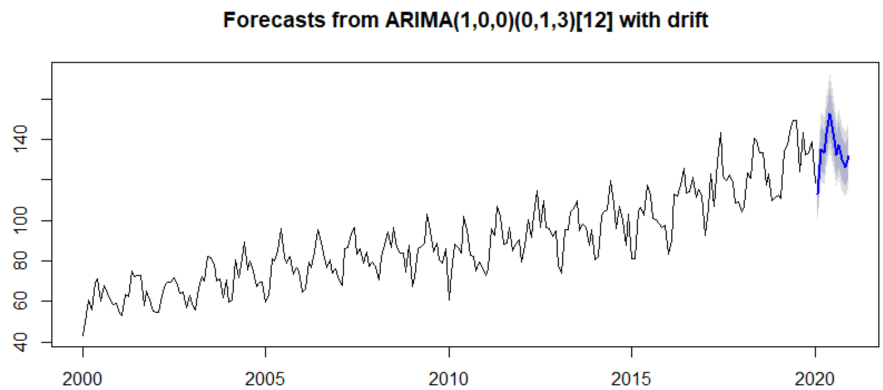
For the comparison on the accuracy, we are going to check the performance on the Train data, auto ARIMA method got the best accuracy on the Train dataset. But, when we check the performance on the Test dataset, its accuracy is not as good as the other ARIMA method that has worse accuracy on the Training dataset. It means this auto ARIMA method might have a problem of overfitting. The 9th method, ARIMA(1,0,0)(0,1,3)$_{12}$ with drift has best accuracy among all the methods have fitted.

Finally, we decided that the 9th method, ARIMA(1,0,0)(0,1,3)$_{12}$ with drift would be our final model.

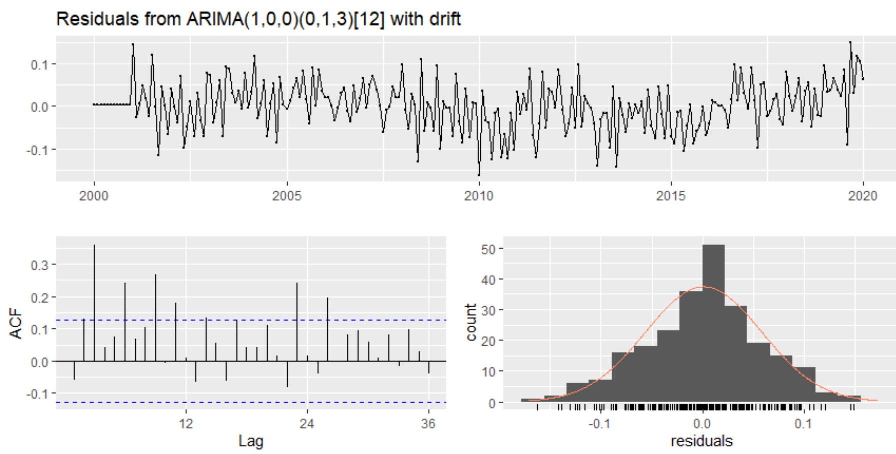
Question 9: Generate out of sample forecasts up to December 2020, based on the complete time series. Discuss your results.

We are going to use the final model we have chosen above. To generate forecasts up to December 2020, since we know that the origin turnover contain data up till January 2020, therefore, we set h=11.

```
ARIMA_9 <- Arima(turnover, lambda=0,order=c(1,0,0), seasonal=c(0,1,3),
include.drift=TRUE)

fc_A9 <- forecast(ARIMA_9, h=11)
```



checkresiduals(fc_A9)




```
        Ljung-Box test

data:  Residuals from ARIMA(1,0,0)(0,1,3)[12] with drift
Q* = 116.78, df = 19, p-value = 4.441e-16

Model df: 5.   Total lags used: 24
```

From the graph of prediction, it looks reasonable.

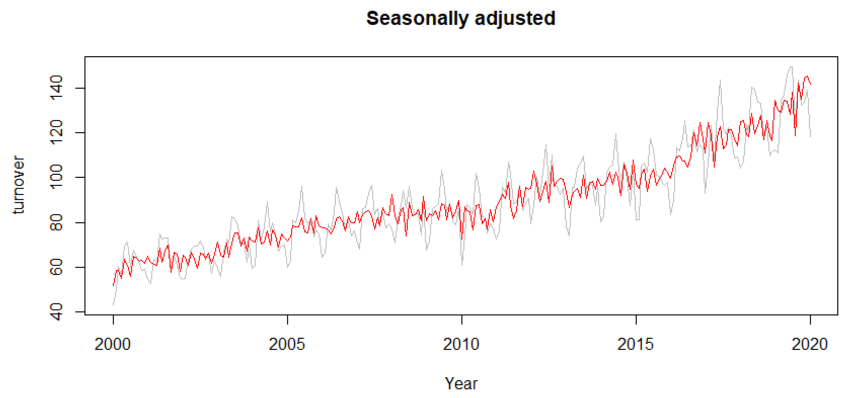
Since we do not have any data available to check the accuracy, we can check the residuals.

From the autocorrelation, the residuals do not look that with noise. The distribution of the residuals could be regard as a normal distribution. Then we can look at the result of the Ljung-Box Test, the p-value is much smaller than 0.05, showing dependence on each other.

Question 10: In addition, generate the seasonally adjusted time series for the Turnover data. Estimate an (auto-) ARIMA model and an (auto-) ETS model for this non-seasonal time series. Compare the forecast accuracy and residual diagnostics of both models, and select the final model for the seasonally adjusted series. Generate out of sample forecasts up to December 2020, based on the complete time series. Discuss your results.

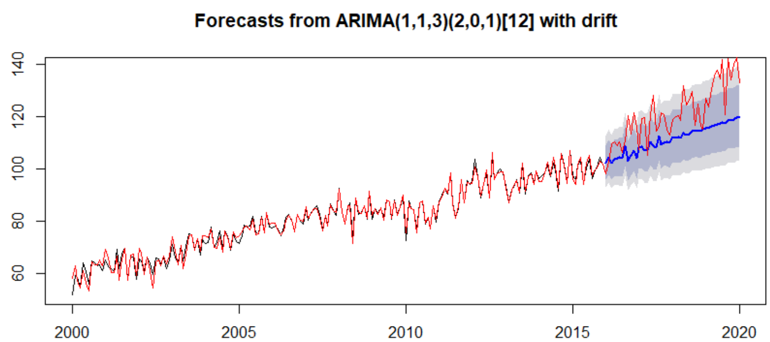
First, we generate the seasonally adjusted time series for the Turnover data:

```
turnover_ns <- seasadj(decompose(turnover, type="multiplicative"))
```



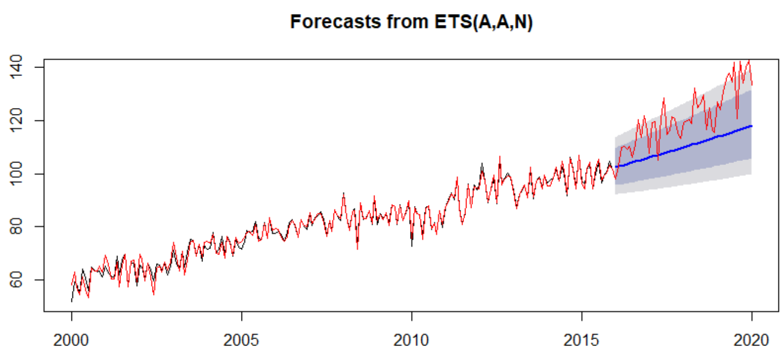
Auto ARIMA:

```
ns_ARIMA_auto <- auto.arima(train_ns, lambda = 0, allowdrift = TRUE)

fc_AA_ns <- forecast(ns_ARIMA_auto, h=h)
```



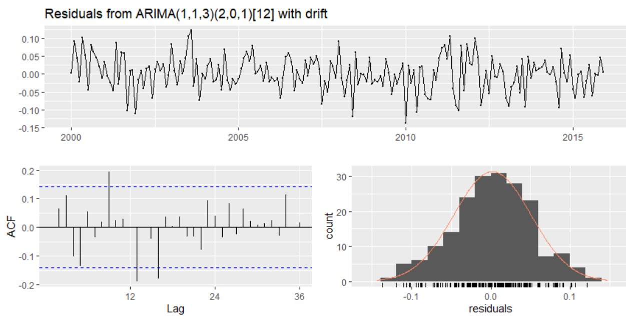
Auto ETS:

```
fc_AE_ns <- forecast(ets(train_ns, lambda=0), h=h)
```

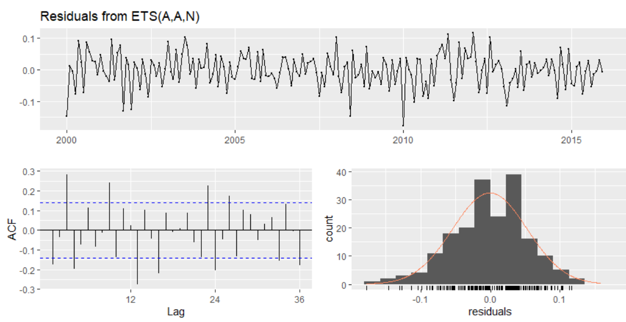


Auto ARIMA suggested ARIMA(1,1,3)(2,0,1)[12] with drift, which is a seasonal ARIMA. It might be the seasonal adjustment does not work well with this time series. Auto ETS suggested ETS(A,A,N).

From the graphs of the forecasts of the two methods, they both look not very accurate.




```
        Ljung-Box test

data:  Residuals from ARIMA(1,1,3)(2,0,1)[12] with drift
Q* = 37.06, df = 16, p-value = 0.002057

Model df: 8.   Total lags used: 24
```




```
        Ljung-Box test

data:  Residuals from ETS(A,A,N)
Q* = 111.02, df = 20, p-value = 1.277e-14

Model df: 4.   Total lags used: 24
```

From the Ljung-Box test, p-values are both smaller than 0.05, and showing dependence on each other.

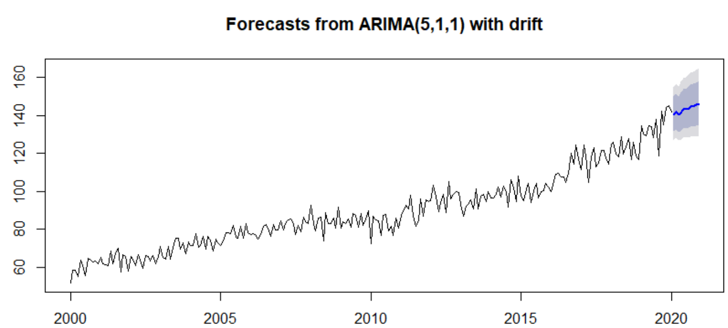
```
> accuracy(fc_AA_ns, test)[,c(2,3,5,6)]
                  RMSE       MAE      MAPE      MASE
Training set  3.941302  3.087519  3.825479 0.5988828
Test set     15.781383 13.056640 10.413300 2.5325829

> accuracy(fc_AE_ns, test)[,c(2,3,5,6)]
                  RMSE       MAE      MAPE      MASE
Training set  4.280711  3.388989  4.238444 0.6573587
Test set     16.735907 13.640148 10.787890 2.6457653
```

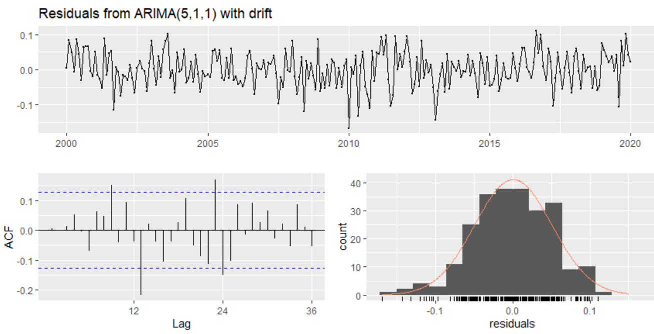
In terms of accuracy, the auto ARIMA method made more accurate forecast than the auto ETS.

Therefore, we choose auto ARIMA for this seasonally adjusted series.

Now we use the final model to make prediction till December 2020:

```
ARIMA_10 <- auto.arima(turnover_ns, lambda = 0, allowdrift = TRUE)

fc_A10 <- forecast(ARIMA_10, h=11)
```






```
        Ljung-Box test

data:  Residuals from ARIMA(5,1,1) with drift
Q* = 51.158, df = 17, p-value = 2.787e-05

Model df: 7.   Total lags used: 24
```

This time auto ARIMA suggest ARIMA(5,1,1) with drift, which is a non-seasonal ARIMA.

From the plot of the forecast, we can see the fitted values have caught the trending information and look not bad.

In terms of residual diagnostics, p-value from the Ljung-Box test is smaller than 0.05, which means we can reject the null hypotheses that this is a white noise procedure, showing dependence on each other.

Exercise 2:

1. Load the data and transform to time series:

Data source: https://finance.yahoo.com/quote/BABA/history?p=BABA

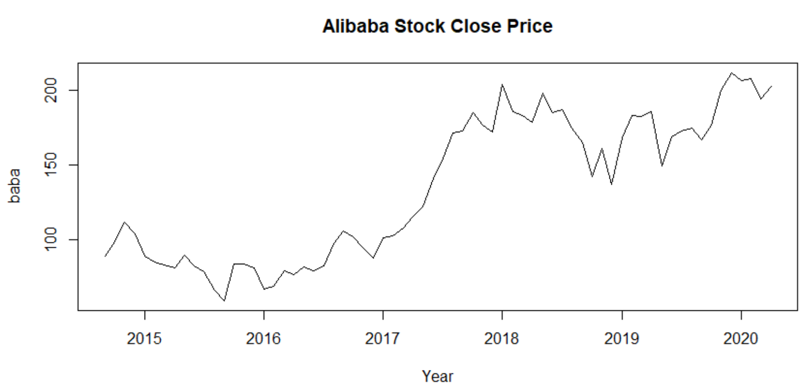
This data is a monthly record of close price of BABA on the stock market on Yahoo Finance from September 2014 to April 2020.

We use R to read in the data and transfer the data to time series.

```
baba <- ts(data2$Close, frequency=12, start = c(2014, 9), end = c(2020, 4))
```

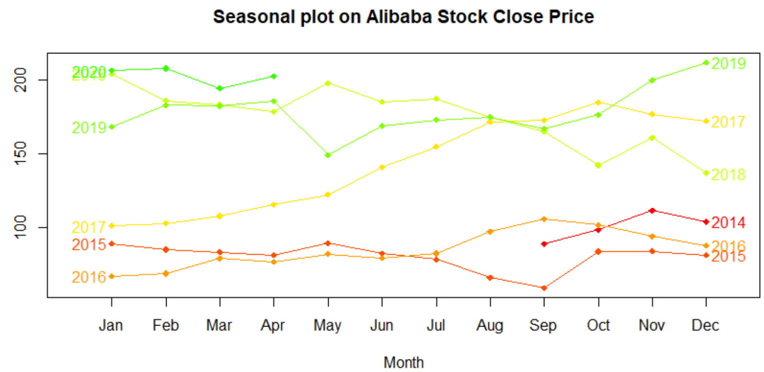
2. Plot the time series and observe through some graphs:

Overview of the time series:



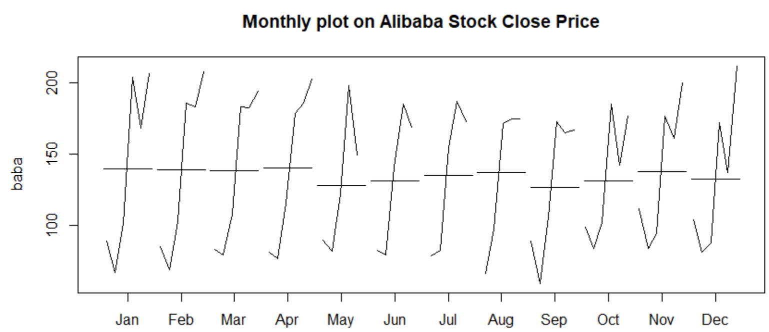
We can see this time series does not have a seasonal pattern but there is a increasing trending pattern.

Seasonal plot:



In the seasonal plot, we can see this time series does not show a certain seasonal pattern.

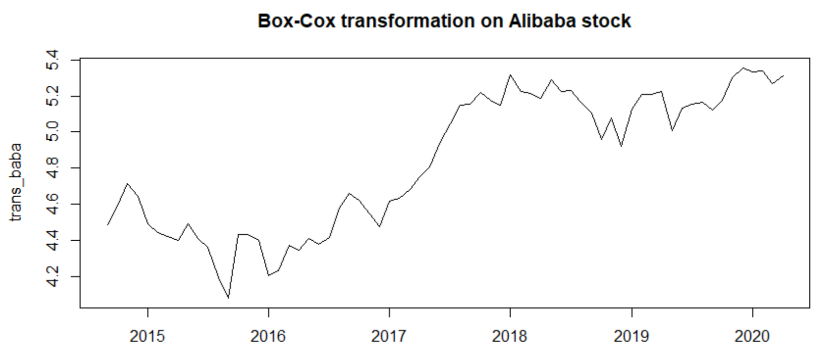
Month plot:



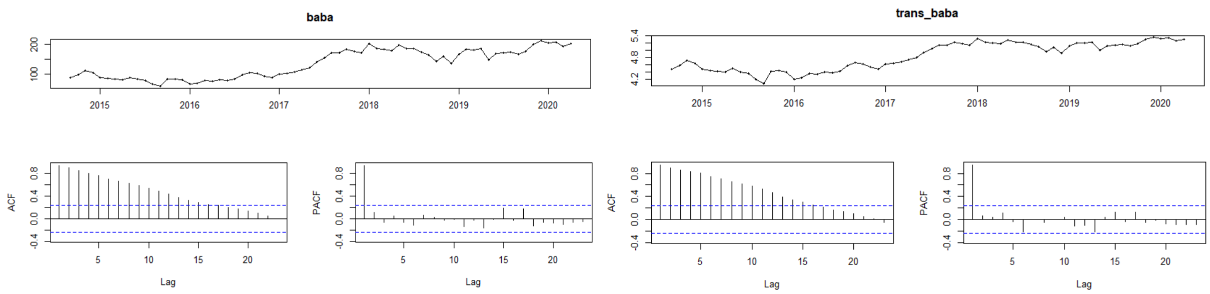
From this month plot, we can see the mean level between each month are very stable, and all of them show an increasing trend pattern.

3. Data transformation if necessary

Since we have observed that this time series does not have seasonal pattern but an increasing trending pattern, we can try the using Box-Cox transformation on this time series.

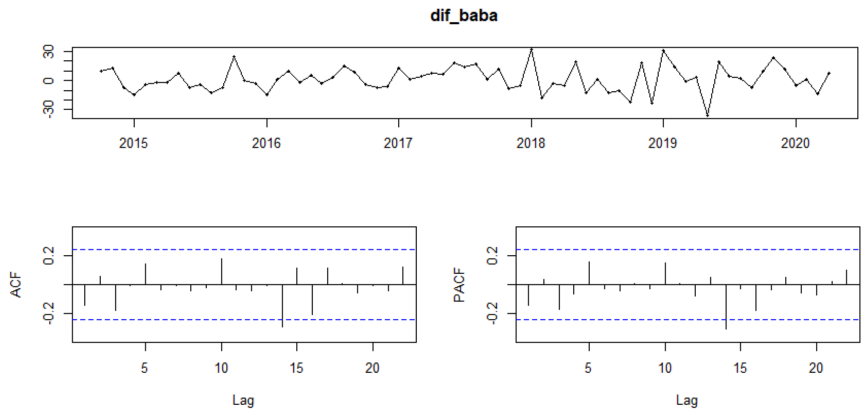


After taking the logarithm on the data, the plot still shows no seasonal but an increasing trending pattern.



From the autocorrelation of the data, we can see a decreasing correlation over lags, and the partial correlation have a spike at the first lag then the rest are all in the insignificant line, it indicates that we should take a first differencing on the data.

Now we will try to take the first difference and have a look:



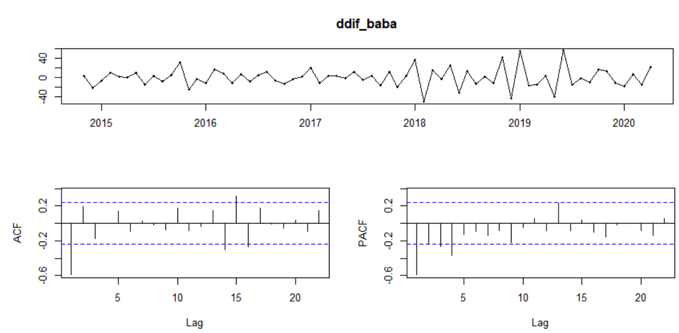
As we look at the autocorrelation, it looks like a white noise. Moreover, we can see the spike quickly drop.

Then we can perform an Augmented Dickey-Fuller test to see if it is stationary.

```
        Augmented Dickey-Fuller Test

data:  dif_baba
Dickey-Fuller = -3.2848, Lag order = 4, p-value = 0.08176
alternative hypothesis: stationary
```

We can see that p-value of the Augmented Dickey-Fuller test is larger than 0.01. It means the first differencing failed to make the data stationary.

Then we take a second differencing and perform an Augmented Dickey-Fuller test again.



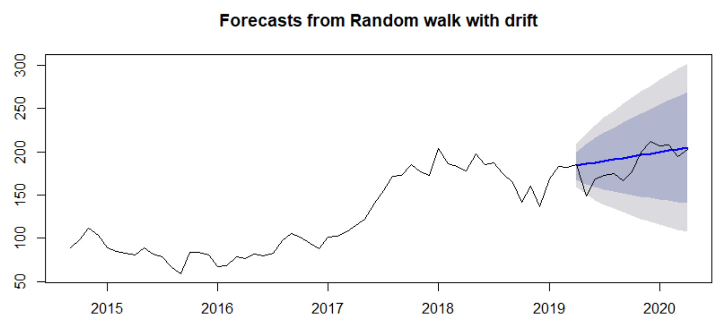
```
        Augmented Dickey-Fuller Test

data:  ddif_baba
Dickey-Fuller = -6.1962, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary

Warning message:
In adf.test(ddif_baba) : p-value smaller than printed p-value
```

After taking the second differencing, the p-value of the Augmented Dickey-Fuller test is now smaller than 0.01. Therefore, we can reject the null hypothesis that this data is non-stationary and non-seasonal. Thus, taking the second differencing has make it a stationary time series.

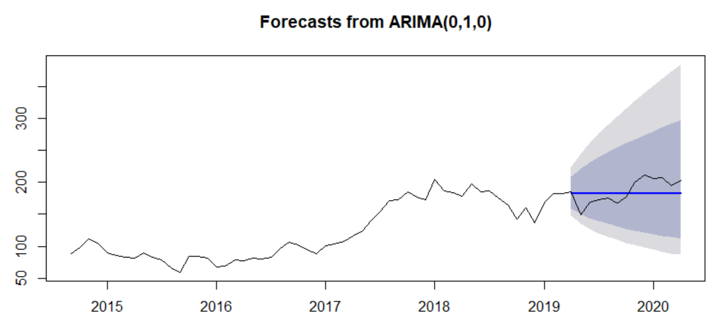
4. Choose methods and fit the data

We have observe this is a non-seasonal time series with an increasing trending pattern.We decide to try the random walk with drift, auto ARIMA and auto ETS:

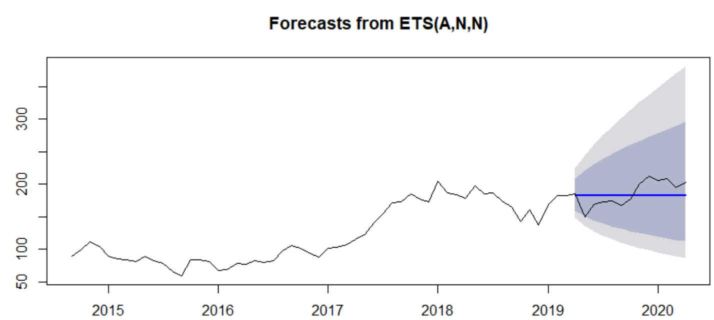
Random walk with drift:



Auto ARIMA:



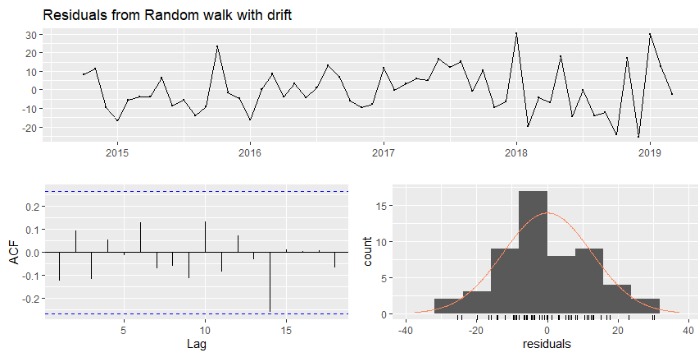
Auto ETS:



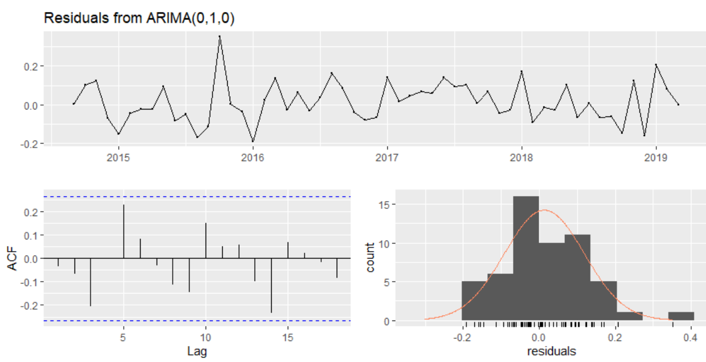
We can see that auto ARIMA suggested ARIMA(0,1,0) for this time series, which is equivalent to random walk.

From the plots of the three methods, they all made reasonable forecasts.

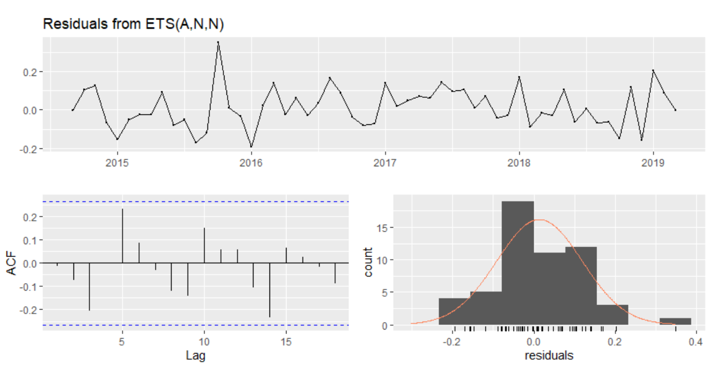
5. Check the performance of the methods




```
        Ljung-Box test

data:  Residuals from Random walk with drift
Q* = 6.6586, df = 10, p-value = 0.7572

Model df: 1.   Total lags used: 11
```




```
        Ljung-Box test

data:  Residuals from ARIMA(0,1,0)
Q* = 10.687, df = 11, p-value = 0.4699

Model df: 0.   Total lags used: 11
```




```
        Ljung-Box test

data:  Residuals from ETS(A,N,N)
Q* = 10.907, df = 9, p-value = 0.2822

Model df: 2.   Total lags used: 11
```

By checking the residuals correlation, we can see the residuals from these three methods are all look like white noise and the distribution look more or less a normal distribution.

Refer to the results of Ljung-Box test, all the p-values are above 0.01, therefore, we can not reject the null hypotheses that this is a white noise procedure for all residuals. It means that the residuals are independent from each other. That is what we want the residuals to be.

Since the results of the residual diagnostic for these three methods are satisfying, we are now going to check the accuracy:

```
> accuracy(fc_baba_rwd, test)[,c(2,3,5,6)]
                 RMSE       MAE     MAPE      MASE
Training set 12.37771  9.982345 8.443669 0.2693783
Test set     16.58614 13.438719 7.763680 0.3626502

> accuracy(fc_baba_aa, test)[,c(2,3,5,6)]
                 RMSE       MAE     MAPE      MASE
Training set 12.38446  9.719959 8.121317 0.2622977
Test set     18.96569 16.666924 8.982741 0.4497648

> accuracy(fc_baba_ets, test)[,c(2,3,5,6)]
                 RMSE       MAE     MAPE      MASE
Training set 12.36090  9.736325 8.144745 0.2627393
Test set     18.96474 16.666544 8.982776 0.4497546
```

First we can look at the accuracy on train dataset: auto ARIMA performed the best on the train dataset;

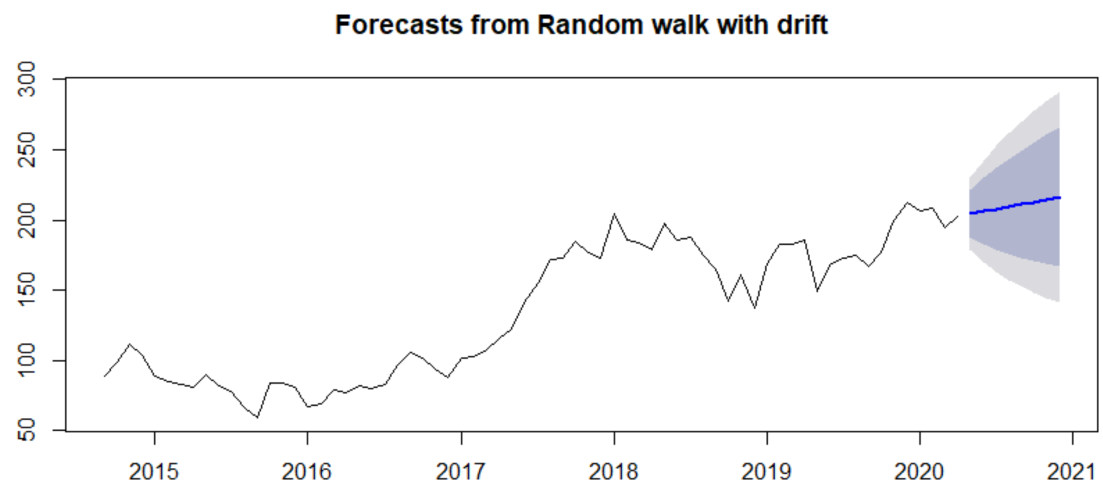
Then we look at the accuracy on the test dataset: random walk with drift performed the best on the test dataset.

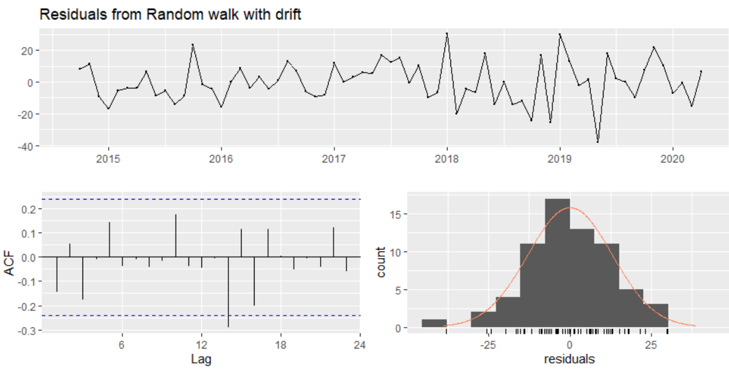
Overall, the performance on train and test between auto ARIMA and auto ETS are very close. Random walk with drift performed relatively good on the train dataset and make the most accurate forecast on the test dataset.

6. Choose the final method and make forecast

Since we want to have the most accurate forecast, we choose random walk with drift as our final model.

Now we are going to use our final model to make a forecast on the close price on stock BABA till December 2020:






```
        Ljung-Box test

data:  Residuals from Random walk with drift
Q* = 15.78, df = 13, p-value = 0.2612

Model df: 1.   Total lags used: 14
```

We can see the forecast results from the plot, the blue line is the fitted value, the darker grey area is the 80% forecast interval and the lighter and broader area is the 95% forecast interval. We expect 20% of the future value outside the darker grey area and 5% of the future value outside the lighter grey area.

In terms of residual diagnostics, from the plot of the autocorrelation between the residuals, it looks like a white noise. From the Ljung-Box test, the p-value is larger than 0.01, which means can not reject the null hypotheses that this is a white noise procedure. It means that there is no information that left out in the model. We would say this is a reasonable forecast.