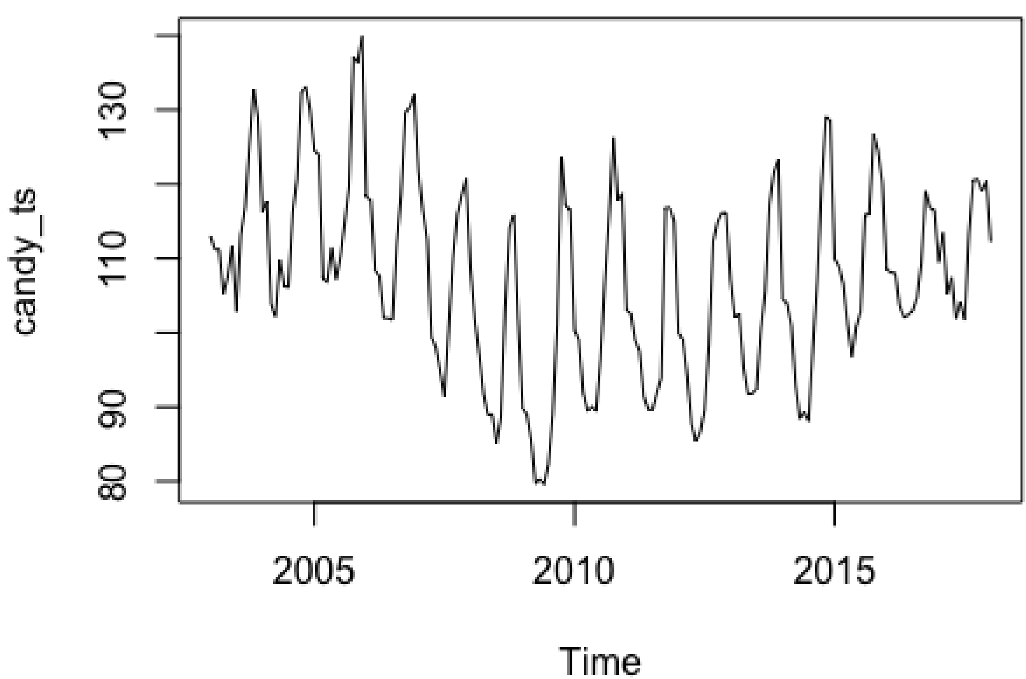
# Business Forecasting :Candy Production Forecast

#Import Data

```
library(readr)
    IPG3113N_Spring18 <- read_csv("IPG3113N_Spring18.csv")

## Parsed with column specification:
## cols(
##   DATE = col_character(),
##   IPG3113N = col_double()
## )

    candy_ts <- ts(IPG3113N_Spring18$IPG3113N,frequency = 12,start=c(2003,1))
```

Ploting Time series:

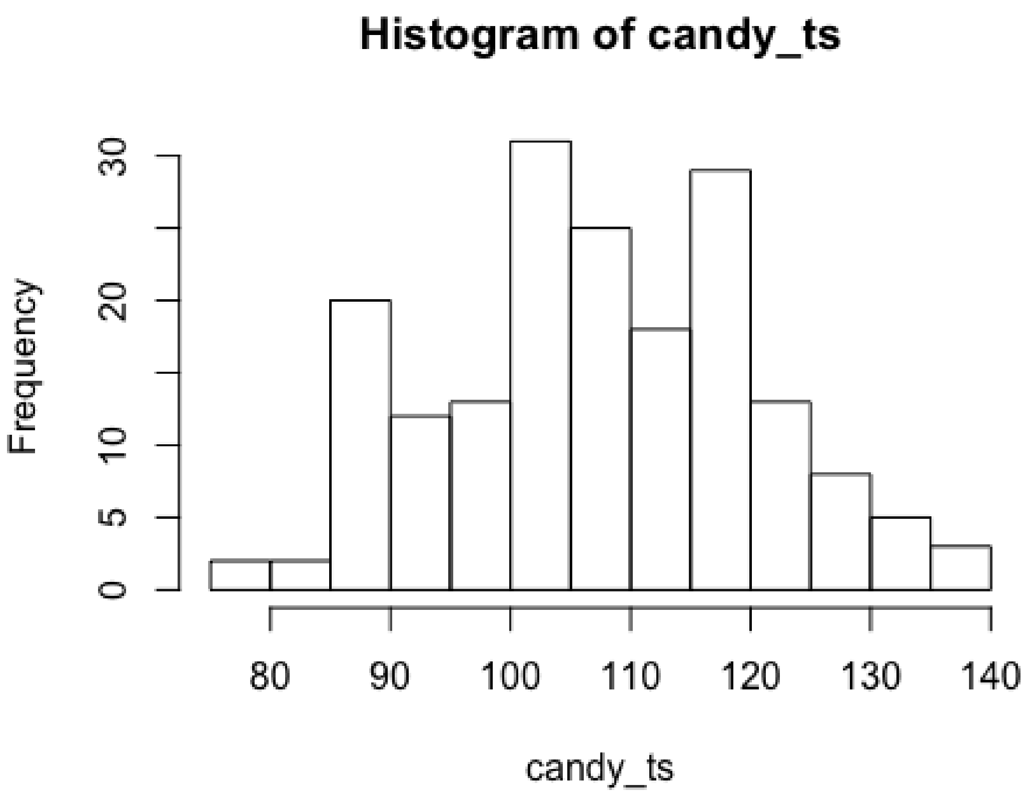
```
    plot(candy_ts)
```



• **Observation**:The time series has strong seeasonality and which is driving the time series however there is no trend in the series as it dropped and then heading up. We can observe that candy production is first decrease until 2010 due to the incfluence of the trend the later we can observe that it started increasing with time but seasonality stayed somewhat constaint around the time.

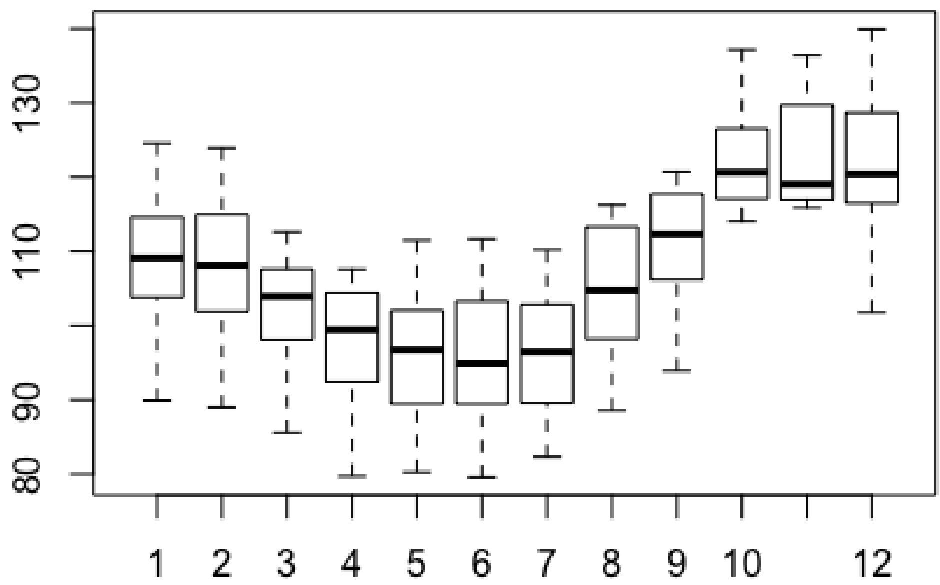
• **The min, max, mean, median, 1st and 3rd Quartile values of the times series?**

```
summary(candy_ts)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   79.57   99.02  107.19  107.45  116.76  139.92

hist(candy_ts)
```




• **box plot.**

```
boxplot(candy_ts~cycle(candy_ts))
```




• **Observation from the Box Plot:**

->From the Histogram the data looks right skewed.

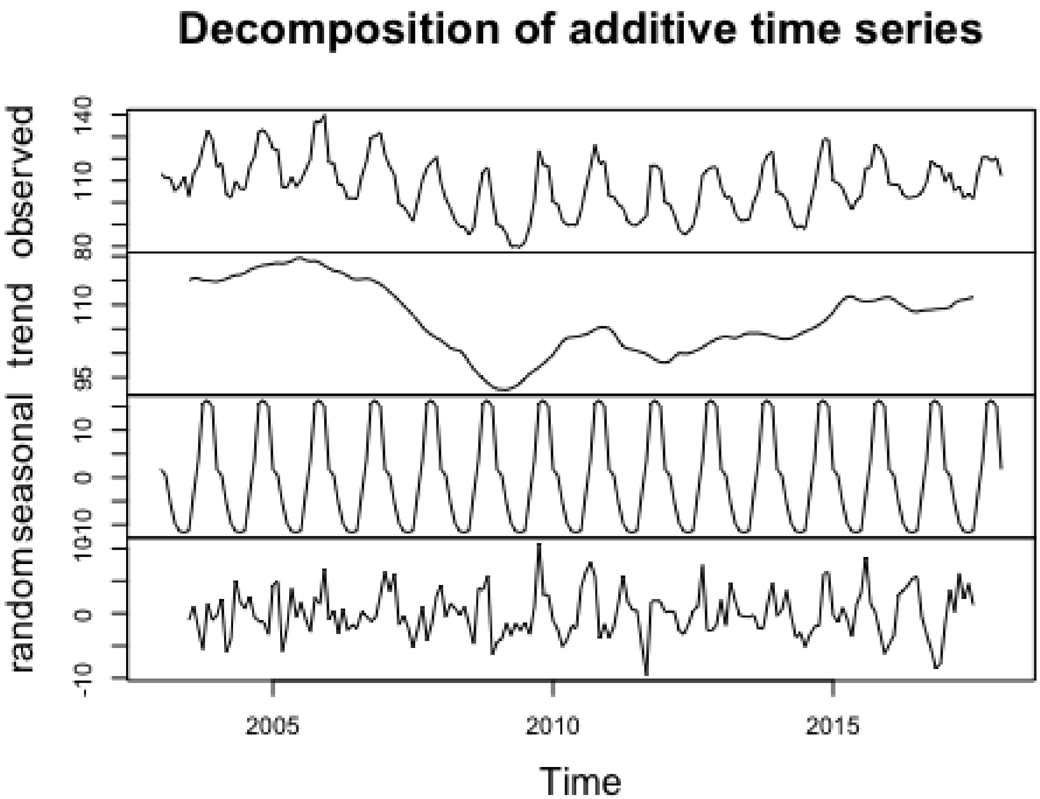
->From the Summary, we got information on minimum, maximum and mean of the candy production in given time frame.

->From the boxplot, we can observe that Nov has the maximum production which implies the maximum production during the Nov month. It is becuase of the christmas season and inventories should get filled before the season.Same way we can observe June has the minimum production, the main reason is the summer time when people prefer Icecreams and cold drinks over the choclates. Hence, Summer months has the minimum production whereas the winter terms has larger production due to the festivel season and also the preferences because people prefer less cold items in winter season.

**Decomposition**

•**The decomposition of the time series.**

```
candy_ts_decom <- decompose(candy_ts)

plot(candy_ts_decom)
```




**• Is the times series seasonal?**

-> Yes, it is seasonal and major component and it contributes 25 units of variance of approx 41 percent of the time series.

**• Is the decomposition additive or multiplicative?**

```
candy_ts_decom$type

## [1] "additive"
```

**• If seasonal, what are the values of the seasonal monthly indices?**

```
candy_ts_decom$seasonal

##              Jan         Feb         Mar         Apr         May
## 2003   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2004   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2005   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2006   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2007   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2008   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2009   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2010   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
```

```
## 2011   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2012   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2013   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2014   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2015   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2016   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2017   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
## 2018   1.7367141
##              Jun         Jul         Aug         Sep         Oct
## 2003 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2004 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2005 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2006 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2007 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2008 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2009 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2010 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2011 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2012 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2013 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2014 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2015 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2016 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2017 -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## 2018
##              Nov         Dec
## 2003  16.2695507  14.9767867
## 2004  16.2695507  14.9767867
## 2005  16.2695507  14.9767867
## 2006  16.2695507  14.9767867
## 2007  16.2695507  14.9767867
## 2008  16.2695507  14.9767867
## 2009  16.2695507  14.9767867
## 2010  16.2695507  14.9767867
## 2011  16.2695507  14.9767867
## 2012  16.2695507  14.9767867
## 2013  16.2695507  14.9767867
## 2014  16.2695507  14.9767867
## 2015  16.2695507  14.9767867
## 2016  16.2695507  14.9767867
## 2017  16.2695507  14.9767867
## 2018
```

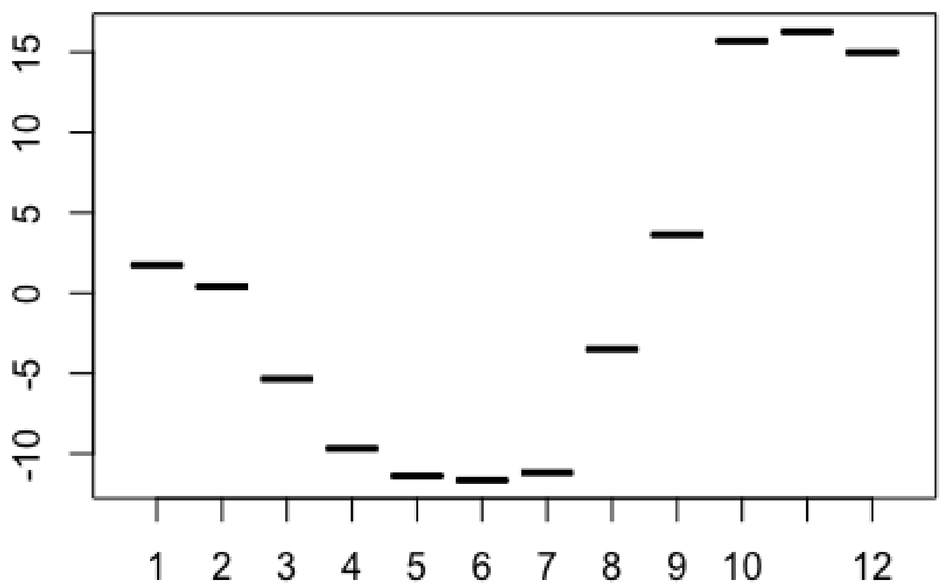
```
candy_ts_decom$figure
```

```
##  [1]   1.7367141   0.4089563  -5.3388684  -9.6736722 -11.3775282
##  [6] -11.6560576 -11.1830346  -3.4903600   3.6323090  15.6952043
## [11]  16.2695507  14.9767867
```

-> Showing the the seasonal monthly indices

• Months which have the value of time series high and low!

```
boxplot(candy_ts_decom$seasonal~cycle(candy_ts_decom$seasonal))
```



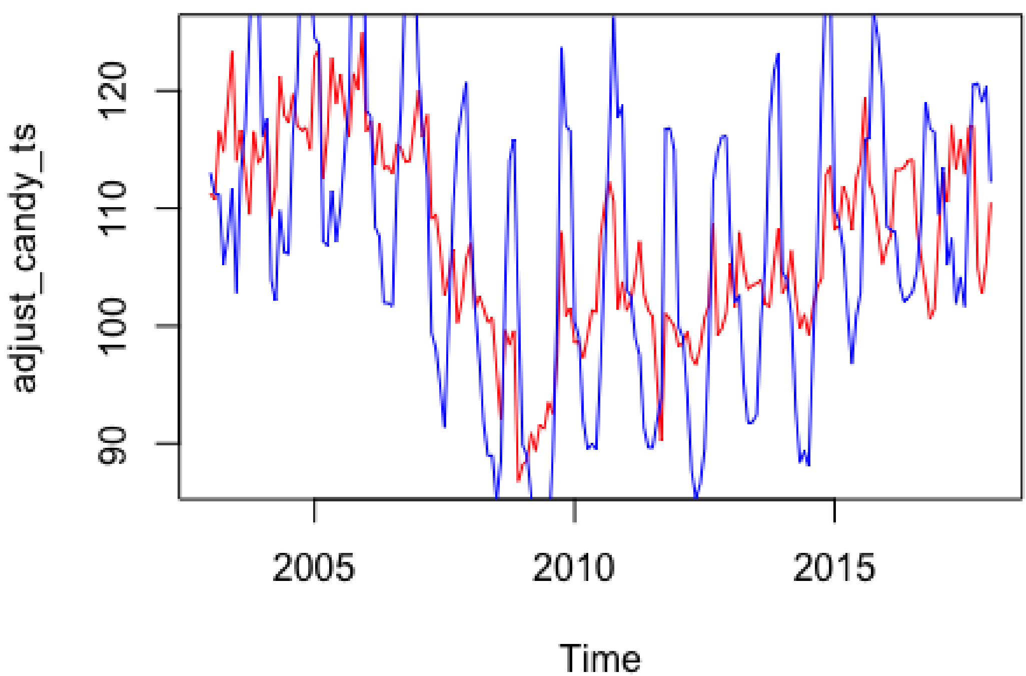

Inference: -> From the above figure, we can see that Nov has the maximum production whereas June has the minimum production.

• The reason behind the value being high in those months and low in those months?

-> we can observe that Nov has the maximum production which implies the maximum production during the Nov month. It is becuase of the christmas season and inventories should get filled before the season.Same way we can observe June has the minimum production, the main reason is the summer time when people prefer Icecreams and cold drinks over the choclates. Hence, Summer months has the minimum production whereas the winter terms has larger production due to the festivel season and also the preferences because people prefer less cold items in winter season.

• Show the plot for time series adjusted for seasonality. Overlay this with the line for actual time series? Does seasonality have big fluctuations to the value of time series?

```
adjust_candy_ts = candy_ts - candy_ts_decom$seasonal
plot(adjust_candy_ts, col='red')
lines(candy_ts, col='blue')
```



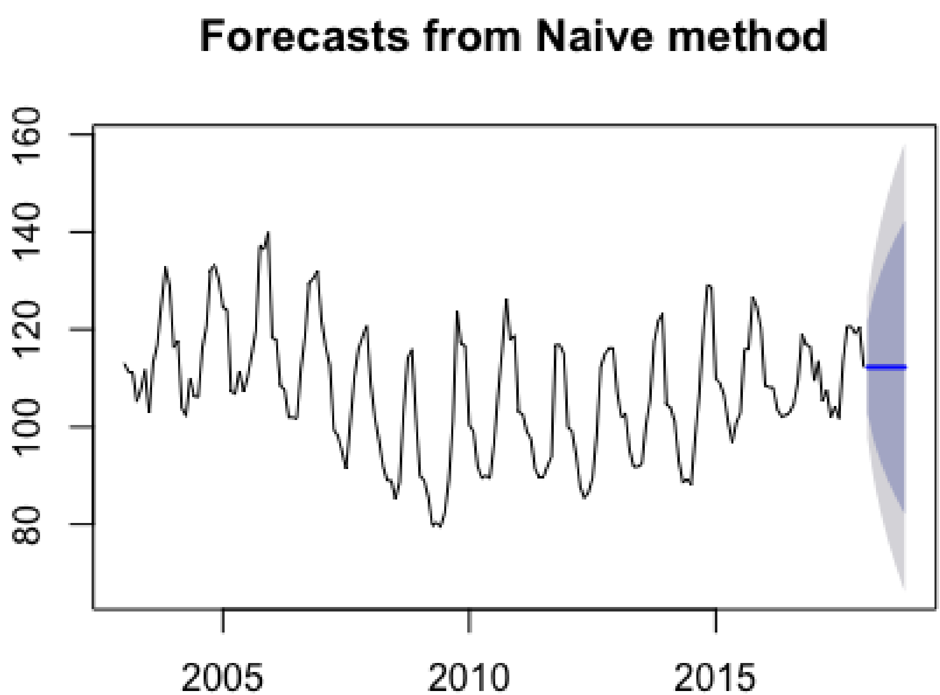

Inference->Yes, Indeed the seasonality has big fluctuations to the actual time series.

**Naïve Method**

• Output

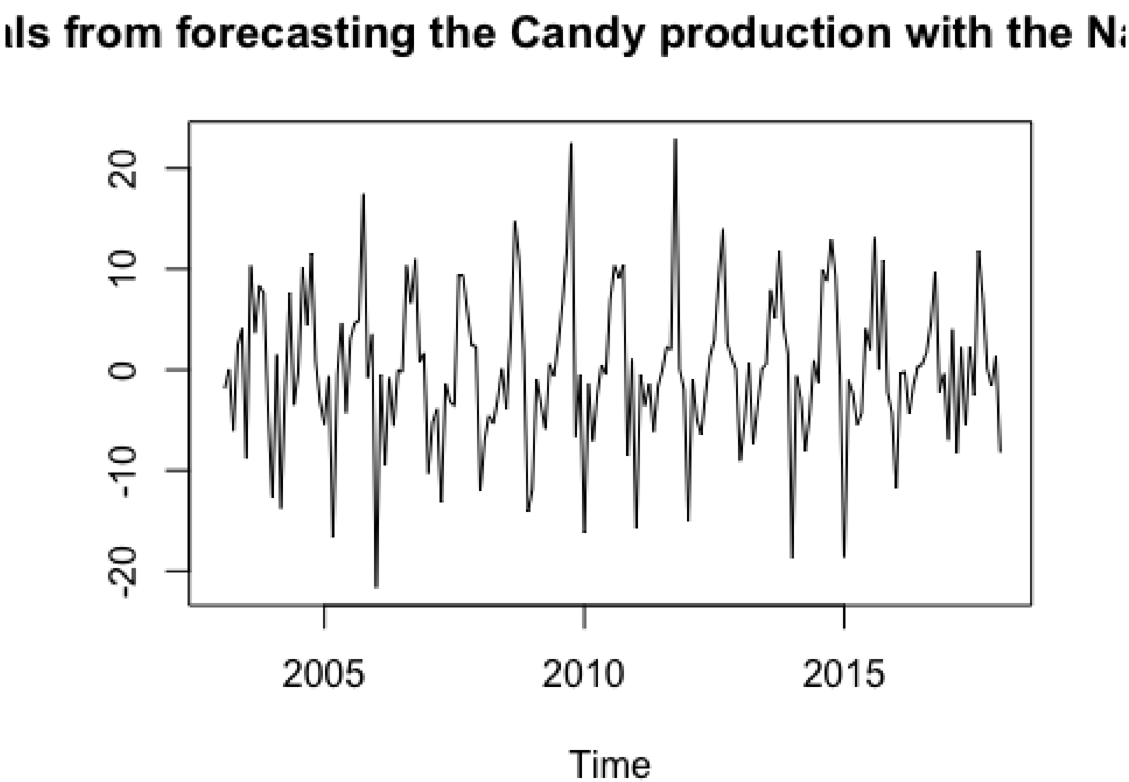
```
library(forecast)

## Warning: package 'forecast' was built under R version 3.4.2

## Warning in as.POSIXlt.POSIXct(Sys.time()): unknown timezone 'zone/tz/2018c
.
## 1.0/zoneinfo/America/New_York'

naive_forecast <- naive(candy_ts)
naive_forecast

##          Point Forecast     Lo 80    Hi 80    Lo 95    Hi 95
## Feb 2018       112.2117 102.69944 121.7240 97.66395 126.7595
```

```
## Mar 2018       112.2117  98.75933 125.6641 91.63807 132.7853
## Apr 2018       112.2117  95.73598 128.6874 87.01426 137.4091
## May 2018       112.2117  93.18717 131.2362 83.11620 141.3072
## Jun 2018       112.2117  90.94163 133.4818 79.68194 144.7415
## Jul 2018       112.2117  88.91151 135.5119 76.57713 147.8463
## Aug 2018       112.2117  87.04462 137.3788 73.72197 150.7014
## Sep 2018       112.2117  85.30696 139.1164 71.06445 153.3590
## Oct 2018       112.2117  83.67491 140.7485 68.56845 155.8550
## Nov 2018       112.2117  82.13128 142.2921 66.20767 158.2157

plot(naive_forecast)
```




• Residual Analysis for this technique and the plot of residuals, inference from the plot indicate?

```
plot(naive_forecast$residuals,main="Residuals from forecasting the Candy prod
uction with the Naïve method",
     ylab="", xlab="Time")
```



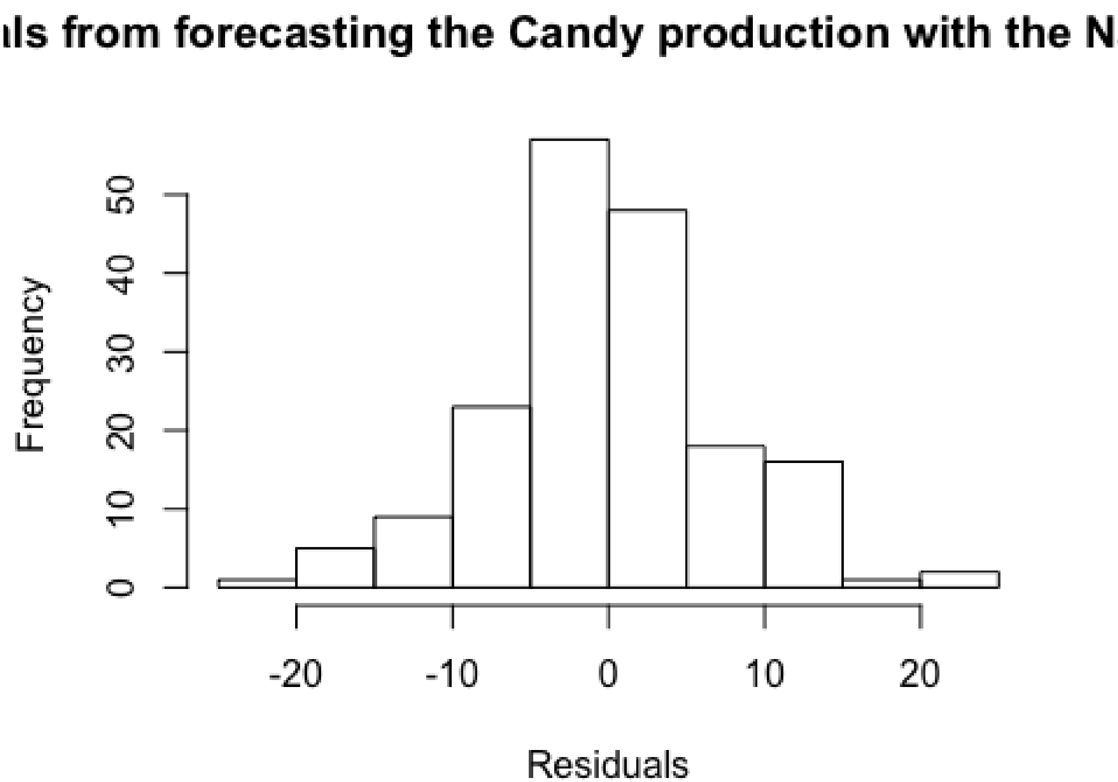

Inference:

->From the time plot, we can see that the variation of the residuals stayed constant through out the time frame.

-> We can a downward spike after 2005 and 2 upward spike near 2010, It could be due to some unprecedented circumstances. Which boosted the production near 2010 .

**o Histogram plot of residuals. What does the plot indicate?**

```
hist(naive_forecast$residuals,main='Residuals from forecasting the Candy prod
uction with the Naïve method', xlab = 'Residuals')
```




Inference:

->The histogram of the residuals shows the distribution of the residuals for all observations.The model fits the data well, the residuals are random with a mean of 0 and the histogram is symmatric about the mean.Hence, it is normally distributed which implies model fits well!

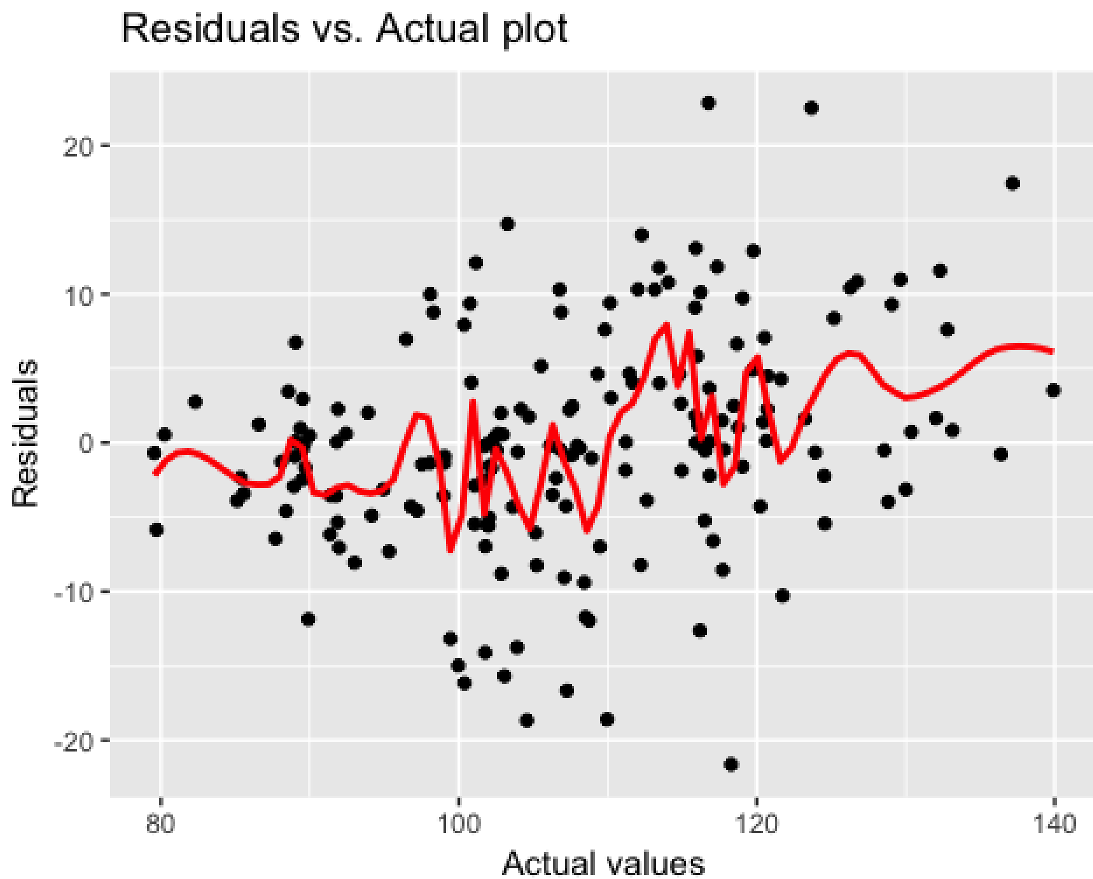
**o fitted values vs. residuals.**

```
library(ggplot2)
qplot(y = naive_forecast$residuals, x = naive_forecast$fitted,
      ylab = "Residuals", xlab = "Fitted values",
      main = " Residuals vs. Fitted plot") +
  stat_smooth(method = "loess", span = 0.1, colour = I("red"), se = FALSE)

## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.
## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.

## Warning: Removed 1 rows containing non-finite values (stat_smooth).

## Warning: Removed 1 rows containing missing values (geom_point).
```


->In the above plot of Residuals VS Fitted values shows residuals has no pattern and they are randomly distributed among themself.Hence the model fits well.

**A plot of actual values vs. residuals**

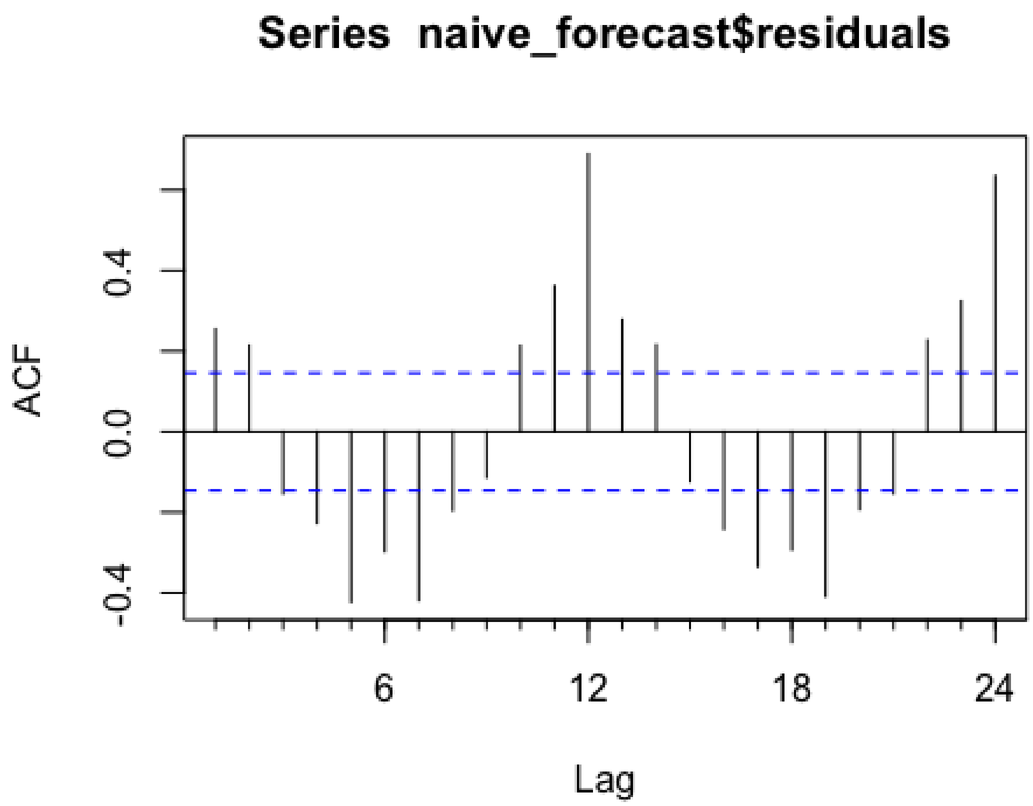
```
library(ggplot2)
qplot(y = naive_forecast$residuals, x = candy_ts,
      ylab = "Residuals", xlab = "Actual values",
      main = " Residuals vs. Actual plot") +
  stat_smooth(method = "loess", span = 0.1, colour = I("red"), se = FALSE)

## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.
## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.

## Warning: Removed 1 rows containing non-finite values (stat_smooth).

## Warning: Removed 1 rows containing missing values (geom_point).
```




->In the above figure, we observe that the Residuals are spreaded sporadiclly and they dont have any pattern among them.

An ACF plot of the residuals? What does this plot indicate?

```
Acf(naive_forecast$residuals)
```




->In the ACF, we can conclude below mentioned points:

->There is a significant Autocorrelation which is positive and negative both.

->It has positive autocorrelation with lags 1,2,11,12,13,22,23,24 whereas negative with lags 5,6,7,16,17,18,19.

->Rest lags correlations are insignificant as they are close to zero.

-> We can see a pattern among the lags which is repeating periodically.

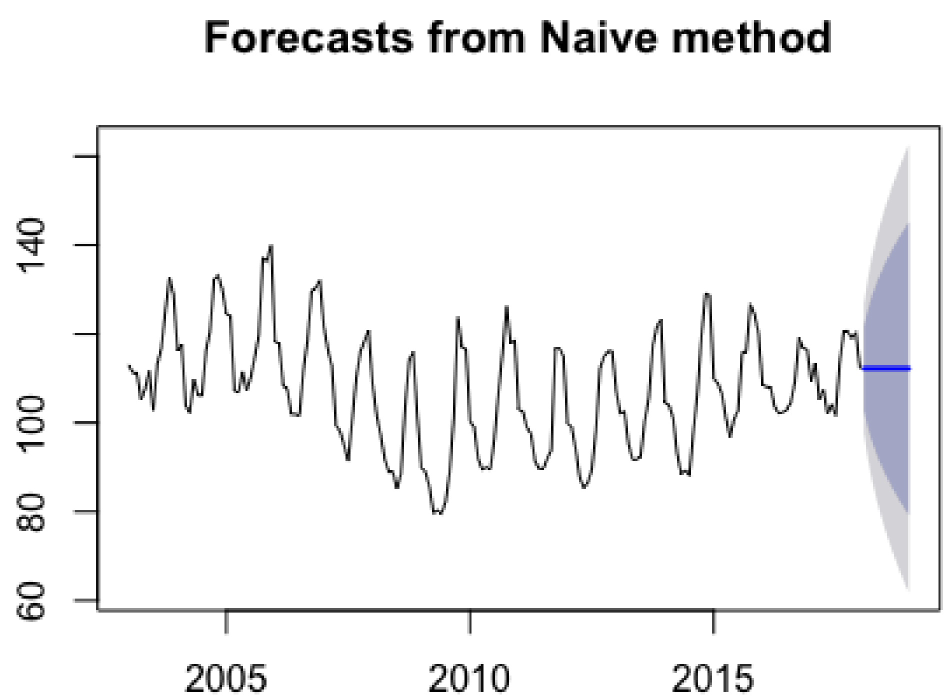
- **The 5 measures of accuracy for this forecasting technique:**

```
naive_accuracy <- accuracy(naive_forecast)
naive_accuracy

##                         ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -0.004547778 7.422458 5.470242 -0.2333585 5.057813 0.9020712
##                   ACF1
## Training set 0.2547176
```

• Forecast

o Time series value for next year. Show table and plot

```
naive_forecast <- naive(candy_ts,12)
naive_forecast

##          Point Forecast     Lo 80    Hi 80    Lo 95    Hi 95
## Feb 2018       112.2117 102.69944 121.7240 97.66395 126.7595
## Mar 2018       112.2117  98.75933 125.6641 91.63807 132.7853
## Apr 2018       112.2117  95.73598 128.6874 87.01426 137.4091
## May 2018       112.2117  93.18717 131.2362 83.11620 141.3072
## Jun 2018       112.2117  90.94163 133.4818 79.68194 144.7415
## Jul 2018       112.2117  88.91151 135.5119 76.57713 147.8463
## Aug 2018       112.2117  87.04462 137.3788 73.72197 150.7014
## Sep 2018       112.2117  85.30696 139.1164 71.06445 153.3590
## Oct 2018       112.2117  83.67491 140.7485 68.56845 155.8550
## Nov 2018       112.2117  82.13128 142.2921 66.20767 158.2157
## Dec 2018       112.2117  80.66309 143.7603 63.96227 160.4611
## Jan 2019       112.2117  79.26025 145.1631 61.81681 162.6066

plot(naive_forecast)
```



• Summarize this forecasting technique

```
summary(naive_forecast)

## 
## Forecast method: Naive method
## 
## Model Information:
## Call: naive(y = candy_ts, h = 12) 
## 
## Residual sd: 7.4432 
## 
## Error measures:
##                        ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -0.004547778 7.422458 5.470242 -0.2333585 5.057813 0.9020712
##                   ACF1
## Training set 0.2547176
## 
## Forecasts:
##          Point Forecast     Lo 80    Hi 80    Lo 95    Hi 95
## Feb 2018       112.2117 102.69944 121.7240 97.66395 126.7595
## Mar 2018       112.2117  98.75933 125.6641 91.63807 132.7853
## Apr 2018       112.2117  95.73598 128.6874 87.01426 137.4091
## May 2018       112.2117  93.18717 131.2362 83.11620 141.3072
## Jun 2018       112.2117  90.94163 133.4818 79.68194 144.7415
## Jul 2018       112.2117  88.91151 135.5119 76.57713 147.8463
## Aug 2018       112.2117  87.04462 137.3788 73.72197 150.7014
## Sep 2018       112.2117  85.30696 139.1164 71.06445 153.3590
## Oct 2018       112.2117  83.67491 140.7485 68.56845 155.8550
## Nov 2018       112.2117  82.13128 142.2921 66.20767 158.2157
## Dec 2018       112.2117  80.66309 143.7603 63.96227 160.4611
## Jan 2019       112.2117  79.26025 145.1631 61.81681 162.6066
```

o How good is the accuracy?

->Error measures are not that high but it could perform well. We can use other model to get better predictions.

o What does it predict the value of time series will be in one year?

->It has Point Forecasted 112.2117 for whole year but highs and lows for the 80 and 95 percent increases while going ahead in time, which can be observed in above table.

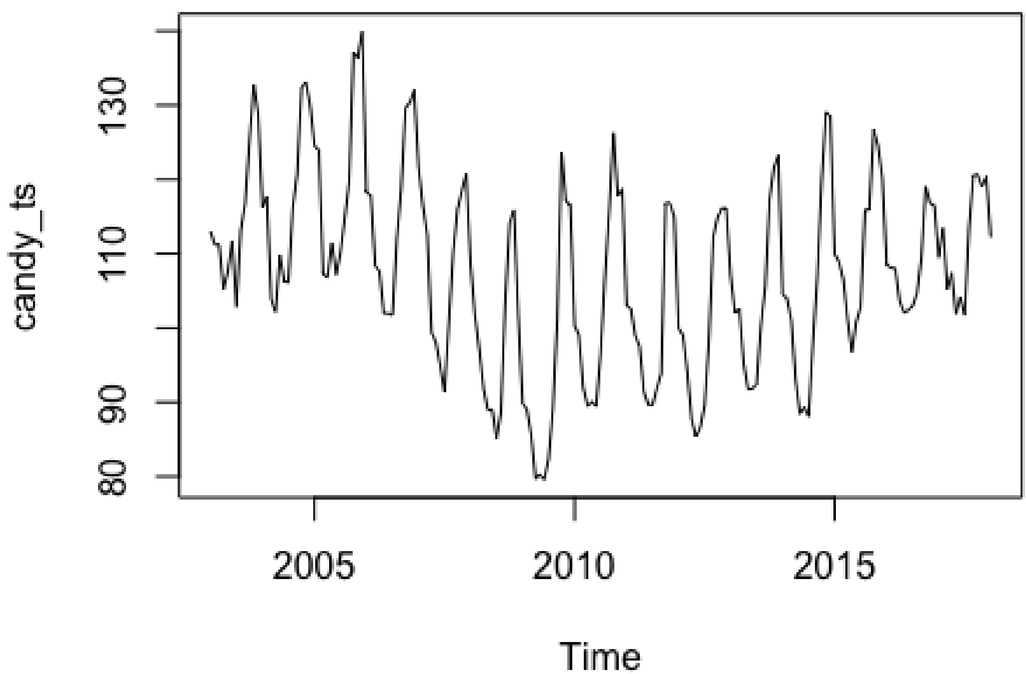
o Other observation

->I believe, WIth point Forecast for prediction for over an year. It would not be a great idea to predict far in the future.

**Simple Moving Averages**

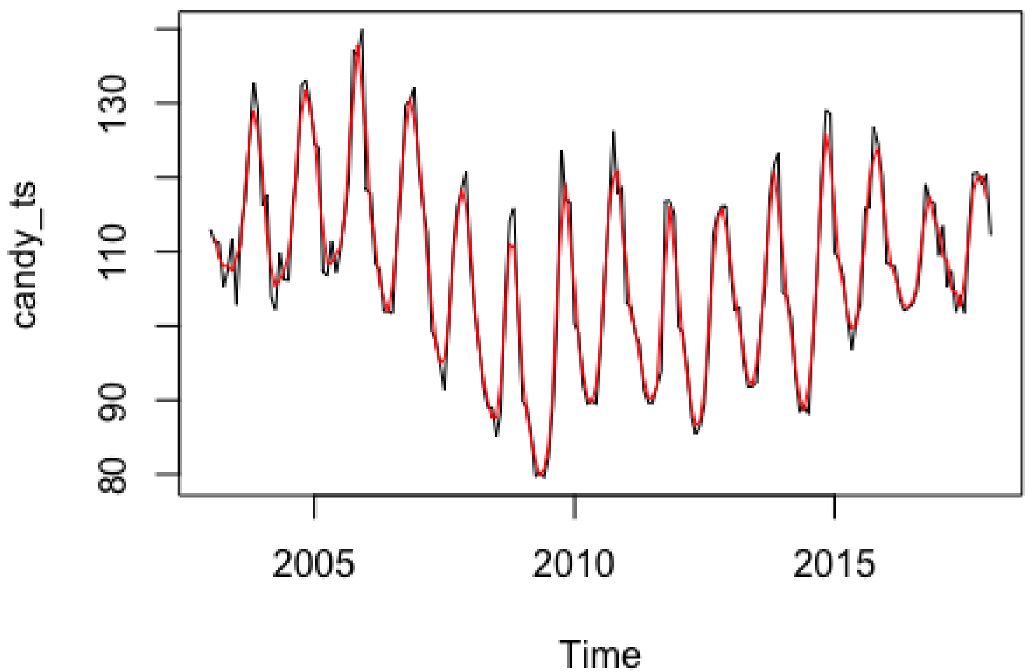
- Plot the graph for time series.

```
plot(candy_ts)
```



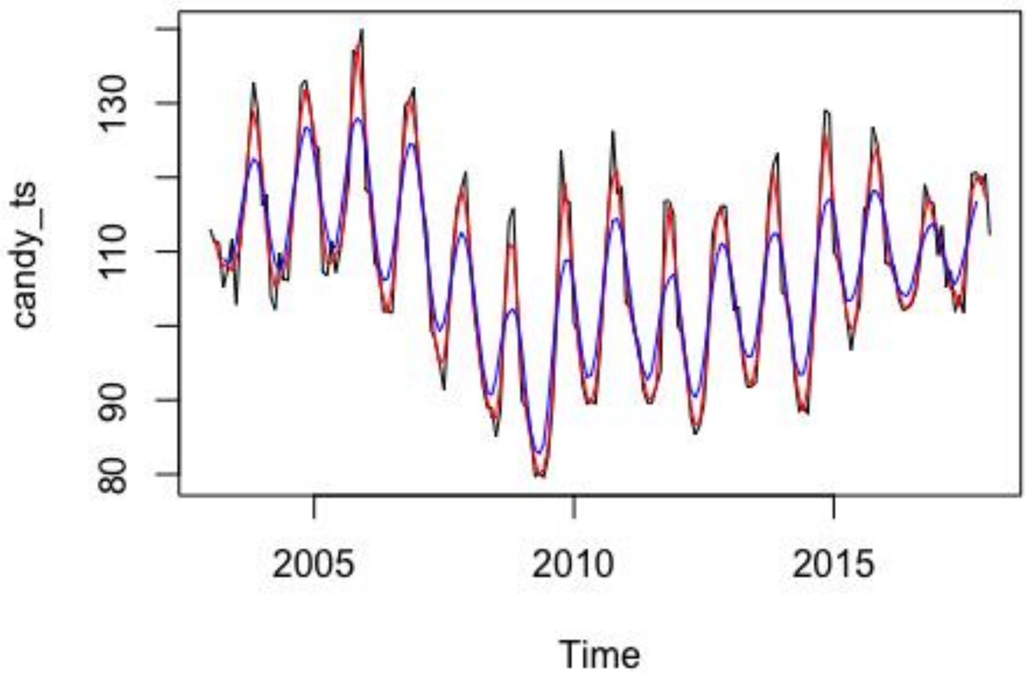

- Show the Simple Moving average of order 3 on the plot above in Red

```
MA3_forecast <- ma(candy_ts,order=3)
plot(candy_ts)
lines(MA3_forecast, col='Red')
```



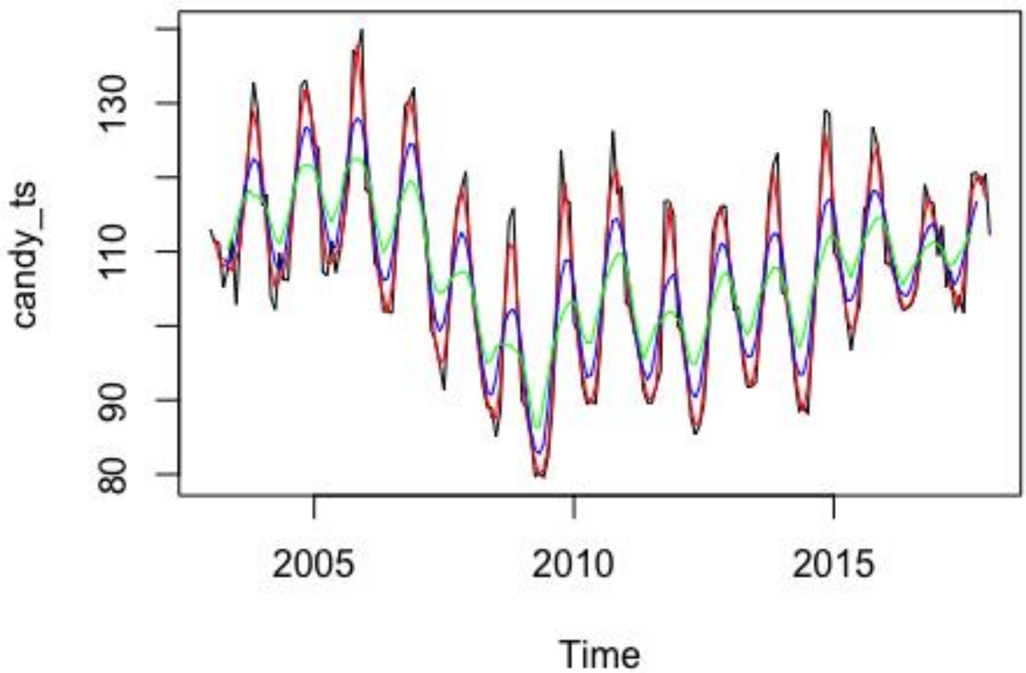

• Show the Simple Moving average of order 6 on the plot above in Blue

```
MA6_forecast <- ma(candy_ts,order=6)
plot(candy_ts)
lines(MA3_forecast, col='Red')
lines(MA6_forecast, col='Blue')
```




• Show the Simple Moving average of order 9 on the plot above in Green

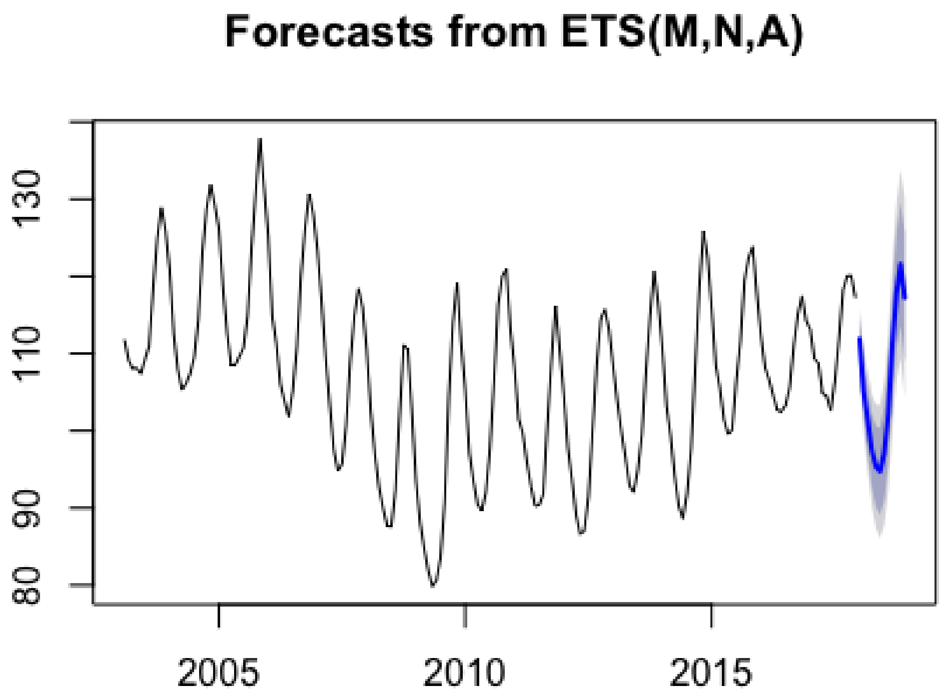
```
MA9_forecast <- ma(candy_ts,order=9)
plot(candy_ts)
lines(MA3_forecast, col='Red')
lines(MA6_forecast, col='Blue')
lines(MA9_forecast, col='Green')
```




• (Bonus) show the forecast of next 12 months using one of the simple average order that you feel works best for time series

```
ma_forcast= forecast(object=MA3_forecast, h= 12 )

## Warning in ets(object, lambda = lambda, allow.multiplicative.trend =
## allow.multiplicative.trend, : Missing values encountered. Using longest
## contiguous portion of time series

ma_forcast

##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Jan 2018      111.88195 109.33042 114.4335 107.97972 115.7842
## Feb 2018      105.06161 101.56138 108.5618  99.70846 110.4148
## Mar 2018      101.36627  97.17125 105.5613  94.95053 107.7820
## Apr 2018       97.43879  92.69110 102.1865  90.17782 104.6998
## May 2018       95.40960  90.18648 100.6327  87.42153 103.3977
## Jun 2018       94.72172  89.06884 100.3746  86.07639 103.3670
## Jul 2018       97.09840  91.02653 103.1703  87.81228 106.3845
## Aug 2018      102.14584  95.64164 108.6500  92.19852 112.0932
## Sep 2018      111.14842 104.16687 118.1300 100.47106 121.8258
## Oct 2018      117.88131 110.39912 125.3635 106.43829 129.3243
## Nov 2018      121.59878 113.61821 129.5794 109.39355 133.8040
## Dec 2018      117.22552 108.80790 125.6431 104.35188 130.0992
```

```
plot(ma_forcast)
```




->I choose MA of order 3 for the forecast because it overlaps best in all of the orders used for the prediction here. Hence, It makes the better predictions as compare to other

• What are your observations of the plot as the moving average order goes up?

-> As the order goes up in moving average. It starts approching towards the mean of whole forecast. which can be observed in the above plot of order 9 which is much near to the mean of time series whereas of order 3 is overlapping best among of all. Simple Smoothing:

• Perform a simple smoothing forecast for next 12 months for the time series.

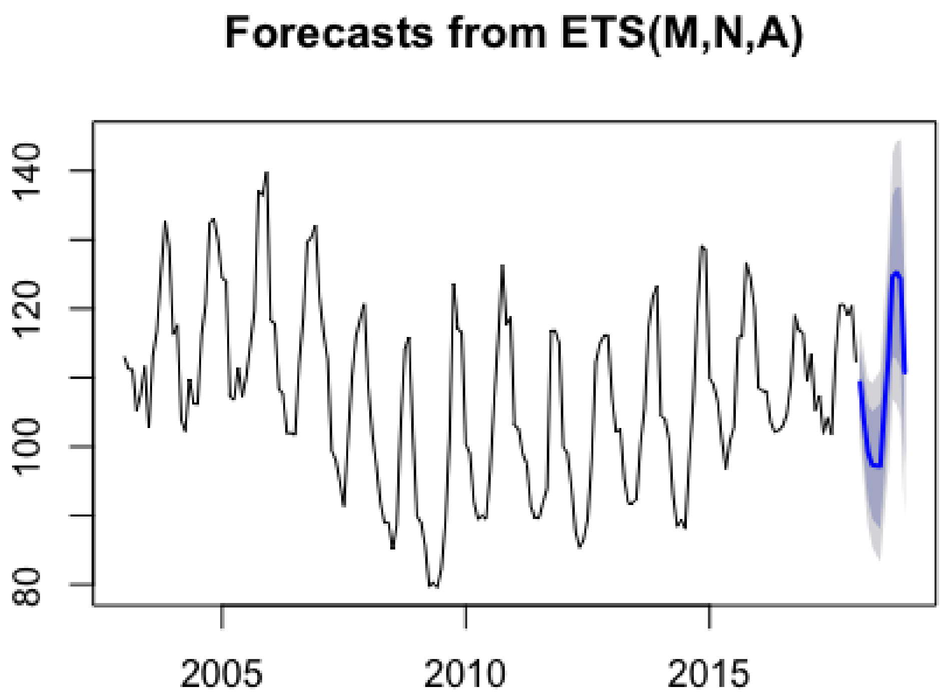
```
ets_candy<-ets(candy_ts)
ets_candy

## ETS(M,N,A)
##
## Call:
##  ets(y = candy_ts)
##
##   Smoothing parameters:
##     alpha = 0.7504
##     gamma = 1e-04
```

```
##
##   Initial states:
##     l = 116.5249
##     s=15.3902 16.2337 15.7225 3.9562 -3.3893 -11.7773
##            -11.7272 -11.6073 -9.7897 -5.2116 0.3267 1.8729
##
##   sigma:  0.0361
##
##      AIC     AICc      BIC
## 1459.573 1462.482 1507.551
```

```
forecast_ets_candy <- forecast.ets(ets_candy, h=12)
forecast_ets_candy
```

```
##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Feb 2018      109.28137 104.21953 114.3432 101.53994 117.0228
## Mar 2018      103.74316  97.61639 109.8699  94.37307 113.1133
## Apr 2018       99.16507  92.19556 106.1346  88.50612 109.8240
## May 2018       97.34774  89.61964 105.0758  85.52863 109.1668
## Jun 2018       97.22752  88.79154 105.6635  84.32580 110.1292
## Jul 2018       97.17709  88.08769 106.2665  83.27605 111.0781
## Aug 2018      105.56456  95.67847 115.4507  90.44509 120.6840
## Sep 2018      112.91012 102.19973 123.6205  96.52999 129.2903
## Oct 2018      124.67594 113.00564 136.3462 106.82776 142.5241
## Nov 2018      125.18774 112.72384 137.6516 106.12585 144.2496
## Dec 2018      124.34403 111.15523 137.5328 104.17350 144.5146
## Jan 2019      110.82761  97.19179 124.4634  89.97343 131.6818
```

```
plot(forecast_ets_candy)
```




o What is the value of alpha? What does that value signify?

->alpha = 0.7504

The value of alpha is high, which means the model is giving more weight to the recent values than to the past value.The value of alpha lies between 0 to 1.

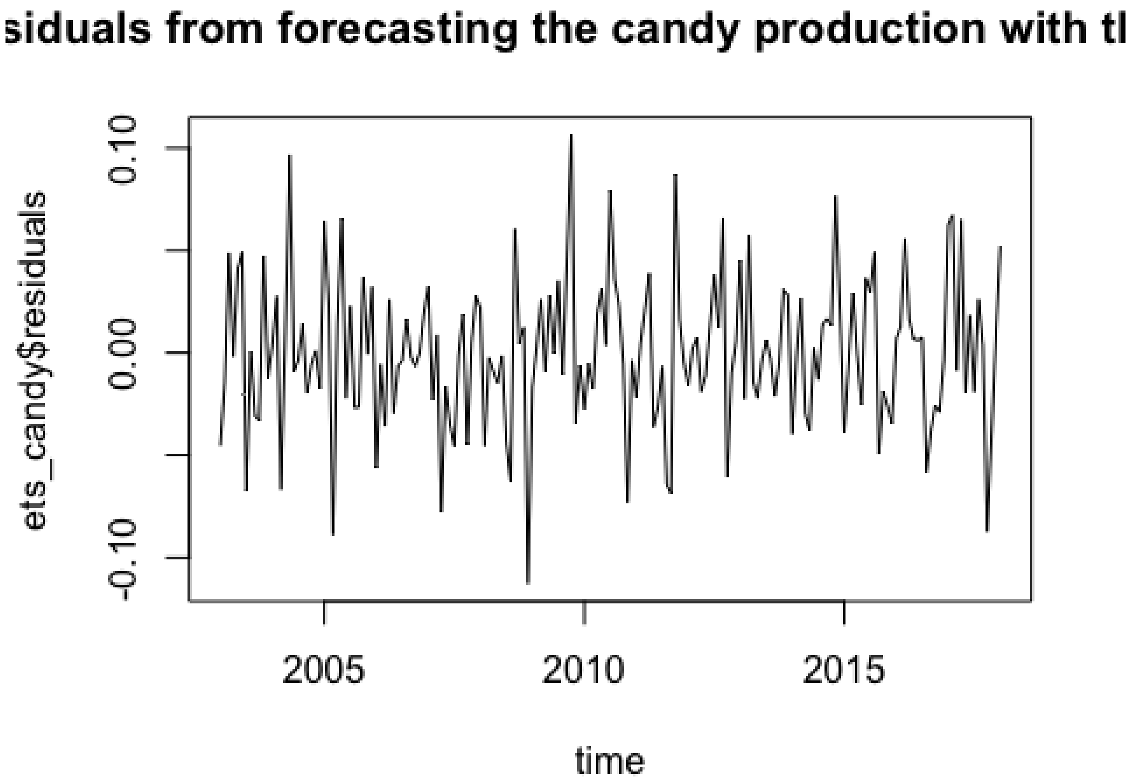
o What is the value of initial state?

```
ets_candy$initstate

##           l          s1          s2          s3          s4          s5
## 116.5249074  15.3902240  16.2337226  15.7225175   3.9562479  -3.3892758
##          s6          s7          s8          s9         s10         s11
## -11.7773018 -11.7271911 -11.6073372  -9.7896608  -5.2116327   0.3267399
##         s12
##   1.8729475
```

o What is the value of sigma? What does the sigma signify?

• Perform Residual Analysis for this technique. o Do a plot of residuals. What does the plot indicate?

```
plot(ets_candy$residuals,xlab = "time", main="Residuals from forecasting the
candy production with the SSM")
```

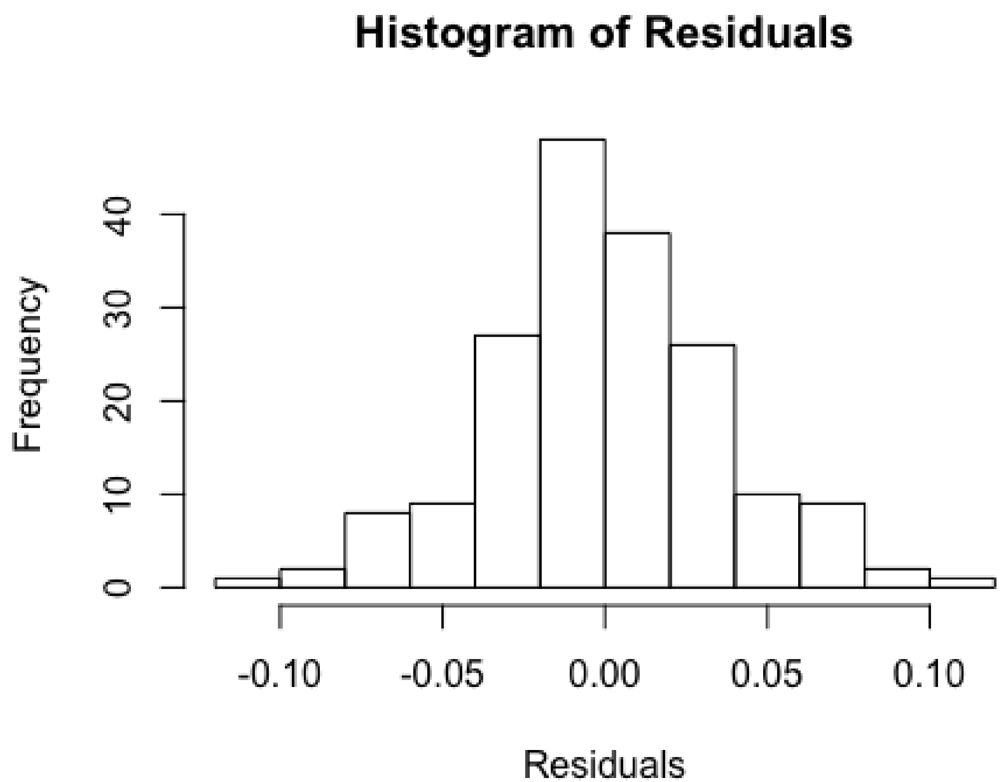



-> By observing the above graph of residuals,we can see that residuals stays same accross the historical data.Hence the it can be considered constant over the time.

-> There are few spikes observed in the residues(2005,,2008,2017), which can be due to some special event occurance in the country which impacted the production of candy.

o Do a Histogram plot of residuals. What does the plot indicate?

```
hist(ets_candy$residuals, xlab = "Residuals", main="Histogram of Residuals")
```

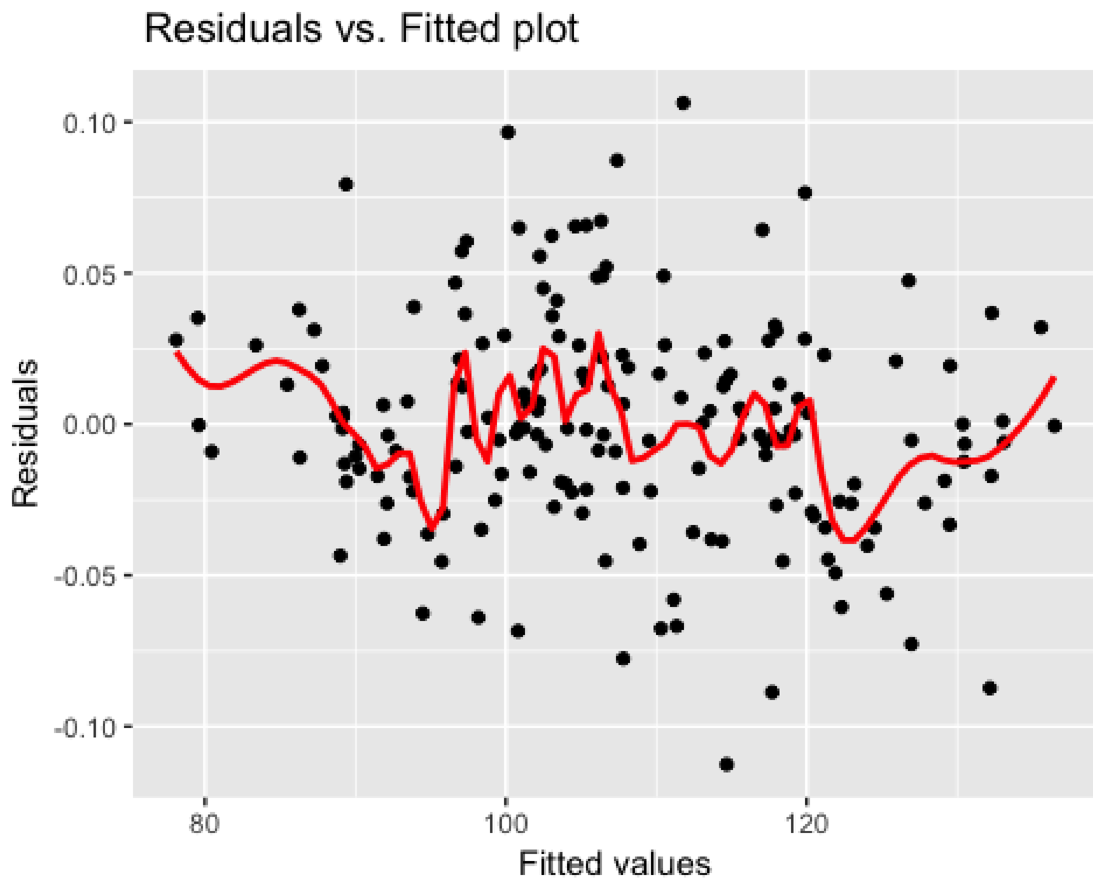



->The histogram of the residuals shows the distribution of the residuals for all observations.The model fits the data well, the residuals are random with a mean of 0 and the histogram is symmatric about the mean.Hence, it is normally distributed which implies model fits well!

o Do a plot of fitted values vs. residuals. What does the plot indicate?

```
library(ggplot2)
qplot(y = ets_candy$residuals, x = forecast_ets_candy$fitted,
      ylab = "Residuals", xlab = "Fitted values",
      main = " Residuals vs. Fitted plot") +
  stat_smooth(method = "loess", span = 0.1, colour = I("red"), se = FALSE)
```

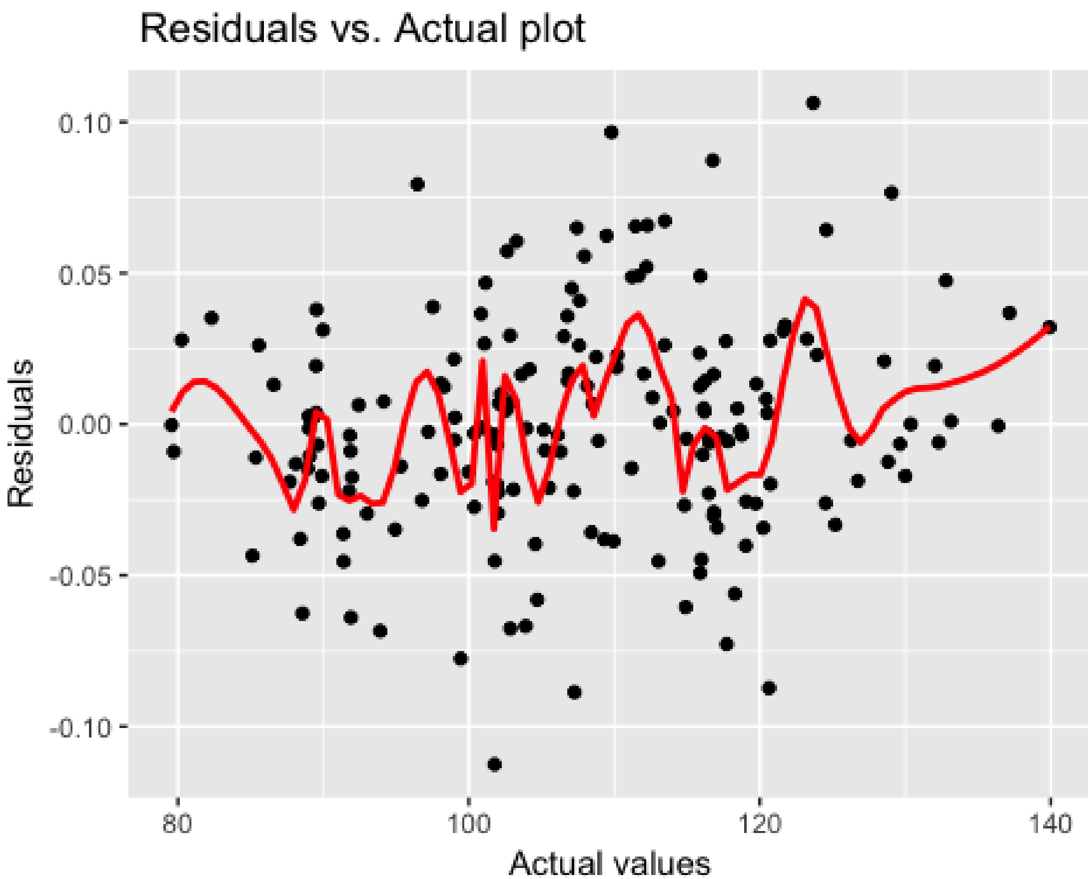
```
## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.
## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.
```




->In the above plot of Residuals VS Fitted values shows residuals has no pattern and they are randomly distributed among themself.Hence the model fits well.

o Do a plot of actual values vs. residuals. What does the plot indicate?

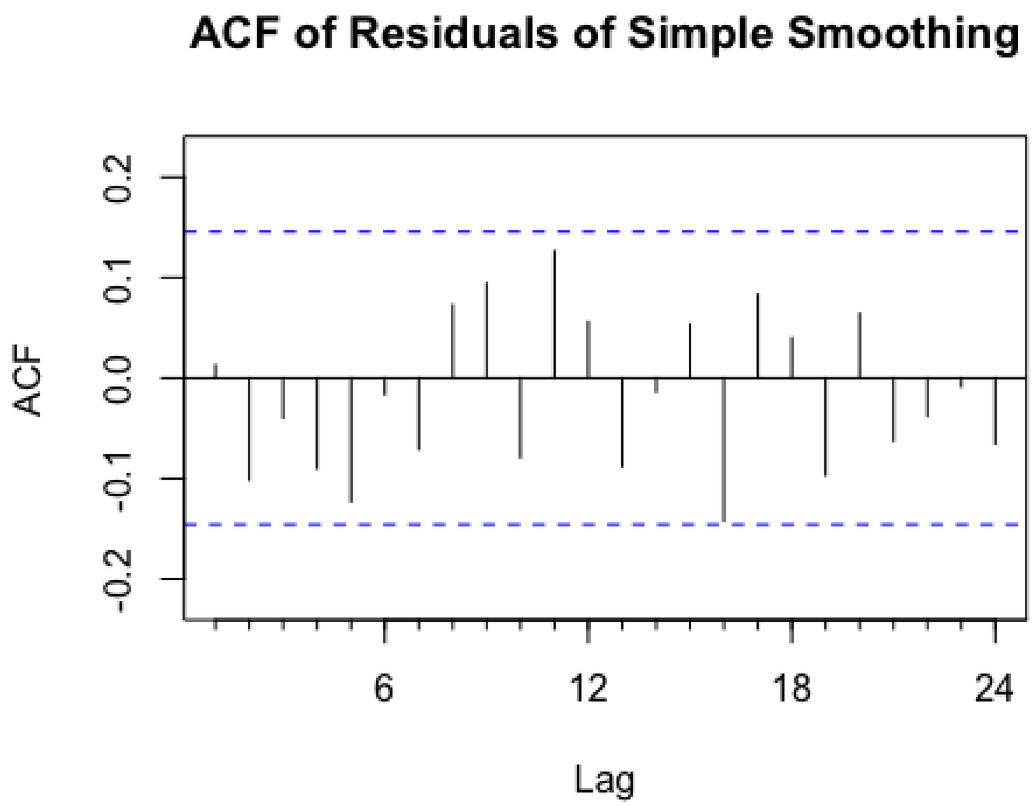
```
library(ggplot2)
qplot(y = ets_candy$residuals, x = candy_ts,
      ylab = "Residuals", xlab = "Actual values",
      main = " Residuals vs. Actual plot") +
  stat_smooth(method = "loess", span = 0.1, colour = I("red"), se = FALSE)

## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.
## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.
```




->In the above figure, we observe that the Residuals are spreaded sporadiclly and they dont have any pattern among them.

o Do an ACF plot of the residuals? What does this plot indicate?

```
Acf(ets_candy$residuals, main = "ACF of Residuals of Simple Smoothing")
```




->Spikes shows the values of Autocorrelation with each lags. We can observe that amplitude of each spike is in the blue segment which implies they are insignificant.Hence Autocorrelation is insignificant.

- Print the 5 measures of accuracy for this forecasting technique

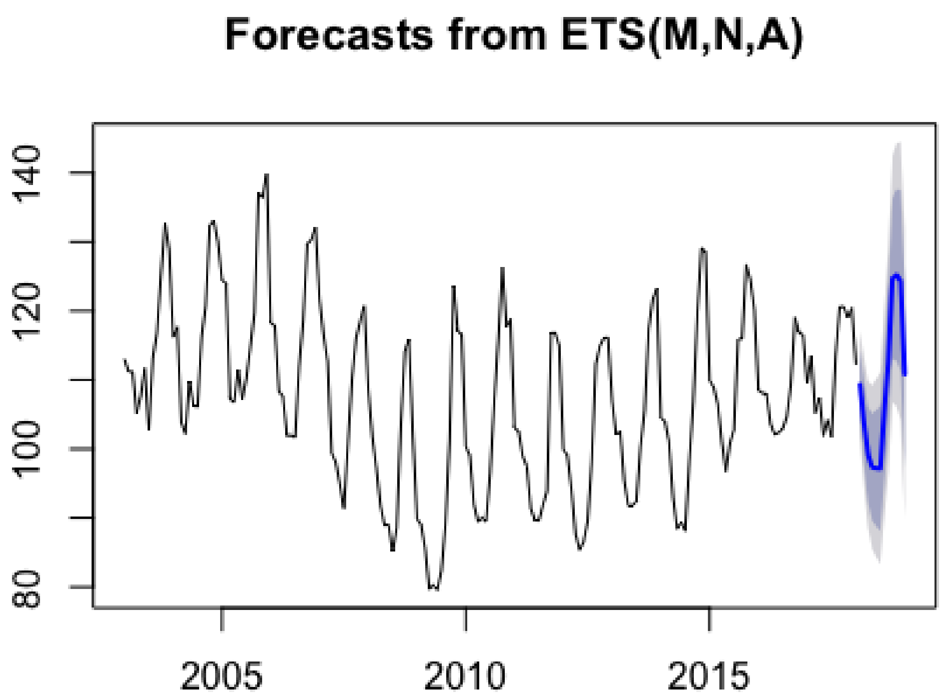
```
accuracy_ets <- accuracy(ets_candy)
accuracy_ets

##                       ME    RMSE      MAE        MPE     MAPE      MASE
## Training set -0.05573914 3.96193 2.971197 -0.1133162 2.749518 0.4899657
##                   ACF1
## Training set 0.0011844
```

- Forecast o Time series value for next year. Show table and plot

```
forecast_ets_candy

##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Feb 2018      109.28137 104.21953 114.3432 101.53994 117.0228
## Mar 2018      103.74316  97.61639 109.8699  94.37307 113.1133
## Apr 2018       99.16507  92.19556 106.1346  88.50612 109.8240
## May 2018       97.34774  89.61964 105.0758  85.52863 109.1668
## Jun 2018       97.22752  88.79154 105.6635  84.32580 110.1292
```

```
## Jul 2018       97.17709  88.08769 106.2665  83.27605 111.0781
## Aug 2018      105.56456  95.67847 115.4507  90.44509 120.6840
## Sep 2018      112.91012 102.19973 123.6205  96.52999 129.2903
## Oct 2018      124.67594 113.00564 136.3462 106.82776 142.5241
## Nov 2018      125.18774 112.72384 137.6516 106.12585 144.2496
## Dec 2018      124.34403 111.15523 137.5328 104.17350 144.5146
## Jan 2019      110.82761  97.19179 124.4634  89.97343 131.6818
```

```
plot(forecast_ets_candy)
```




- Summarize this forecasting technique

```
summary(forecast_ets_candy)
```

```
##
## Forecast method: ETS(M,N,A)
##
## Model Information:
## ETS(M,N,A)
##
## Call:
##  ets(y = candy_ts)
##
##   Smoothing parameters:
```

```
##     alpha = 0.7504
##     gamma = 1e-04
##
##   Initial states:
##     l = 116.5249
##     s=15.3902 16.2337 15.7225 3.9562 -3.3893 -11.7773
##            -11.7272 -11.6073 -9.7897 -5.2116 0.3267 1.8729
##
##   sigma:  0.0361
##
##      AIC     AICc      BIC
## 1459.573 1462.482 1507.551
##
## Error measures:
##                       ME    RMSE      MAE        MPE     MAPE      MASE
## Training set -0.05573914 3.96193 2.971197 -0.1133162 2.749518 0.4899657
##                   ACF1
## Training set 0.0011844
##
## Forecasts:
##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Feb 2018      109.28137 104.21953 114.3432 101.53994 117.0228
## Mar 2018      103.74316  97.61639 109.8699  94.37307 113.1133
## Apr 2018       99.16507  92.19556 106.1346  88.50612 109.8240
## May 2018       97.34774  89.61964 105.0758  85.52863 109.1668
## Jun 2018       97.22752  88.79154 105.6635  84.32580 110.1292
## Jul 2018       97.17709  88.08769 106.2665  83.27605 111.0781
## Aug 2018      105.56456  95.67847 115.4507  90.44509 120.6840
## Sep 2018      112.91012 102.19973 123.6205  96.52999 129.2903
## Oct 2018      124.67594 113.00564 136.3462 106.82776 142.5241
## Nov 2018      125.18774 112.72384 137.6516 106.12585 144.2496
## Dec 2018      124.34403 111.15523 137.5328 104.17350 144.5146
## Jan 2019      110.82761  97.19179 124.4634  89.97343 131.6818
```

o How good is the accuracy?

-> Accuracy of the model is good, which is better then simple smoothing.

o What does it predict the value of time series will be in one year?

->In one year the value of production would be 110.82761 with 95 percent confidence interval, It would be 89.97 low to 131.68 high.

o Other observation During the residual analysis, we observed that they are normally distributed.

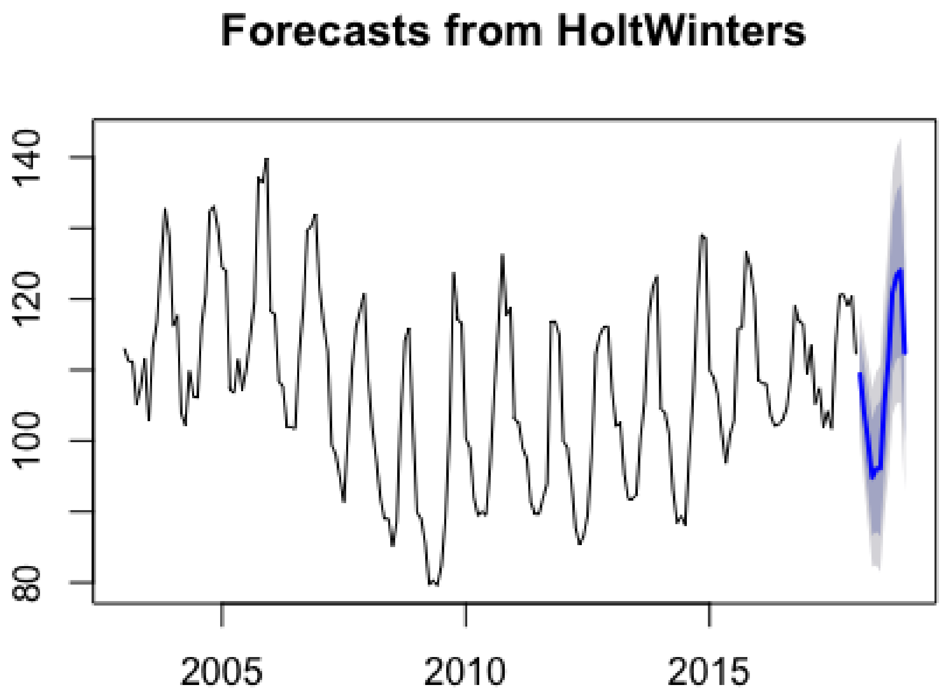
**Holt-Winters**

• Perform Holt-Winters forecast for next 12 months for the time series.

```
HW_candy_ts <- HoltWinters(candy_ts)

HW_candy_ts

## Holt-Winters exponential smoothing with trend and additive seasonal compon
ent.
##
## Call:
## HoltWinters(x = candy_ts)
##
## Smoothing parameters:
##  alpha: 0.6058406
##  beta : 0
##  gamma: 0.6033215
##
## Coefficients:
##             [,1]
## a   108.28086742
## b     0.07459764
## s1    1.01477173
## s2   -4.28108430
## s3   -8.63739788
## s4  -13.78779419
## s5  -12.58529699
## s6  -12.65078438
## s7   -3.58622669
## s8    2.57698313
## s9   11.90956775
## s10  14.26863348
## s11  14.97629420
## s12   3.25171168

HW_candy_ts_forecast <- forecast(HW_candy_ts, h= 12)
HW_candy_ts_forecast

##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Feb 2018      109.37024 103.70645 115.0340 100.70822 118.0323
## Mar 2018      104.14898  97.52684 110.7711  94.02130 114.2767
## Apr 2018       99.86726  92.40892 107.3256  88.46071 111.2738
## May 2018       94.79146  86.58165 103.0013  82.23564 107.3473
## Jun 2018       96.06856  87.17051 104.9666  82.46017 109.6769
## Jul 2018       96.07767  86.54093 105.6144  81.49248 110.6629
## Aug 2018      105.21682  95.08156 115.3521  89.71627 120.7174
## Sep 2018      111.45463 100.75427 122.1550  95.08984 127.8194
## Oct 2018      120.86181 109.62473 132.0989 103.67618 138.0474
## Nov 2018      123.29548 111.54617 135.0448 105.32647 141.2645
```

```
## Dec 2018       124.07774 111.83762 136.3178 105.35810 142.7974
## Jan 2019       112.42775  99.71577 125.1397  92.98645 131.8691
```

```r
plot(HW_candy_ts_forecast)
```




o What is the value of alpha? What does that value signify? alpha: 0.6058406

The value of alpha is high, which means the model is giving more weight to the recent values than to the past value.The value of alpha lies between 0 to 1.

o What is the value of beta? What does that value signify?

beta : 0 The value of beta , tells us the weights given to slope of the trend component but here it is zero which means it is given an average of all at it is not weighted for the recent points.

o What is the value of gamma? What does that value signify?

gamma: 0.6033215 Gamma represents the weights given to the seasonality component of the time time series. Here it is high which means it gives weight to seasonality component of recent points then the old ones.

o What is the value of initial states for the level, trend and seasonality? What do these values signify?

```r
HW_candy_ts$coefficients
```

```
##            a            b           s1           s2           s3 
## 108.28086742   0.07459764   1.01477173  -4.28108430  -8.63739788 
##           s4           s5           s6           s7           s8 
## -13.78779419 -12.58529699 -12.65078438  -3.58622669   2.57698313 
##           s9          s10          s11          s12 
##  11.90956775  14.26863348  14.97629420   3.25171168
```

->a,b,s1 contain the initial estimated values for the level, trend and seasonal components respectively.

o What is the value of sigma? What does the sigma signify?

```r
sd(complete.cases(HW_candy_ts_forecast$residuals))
```

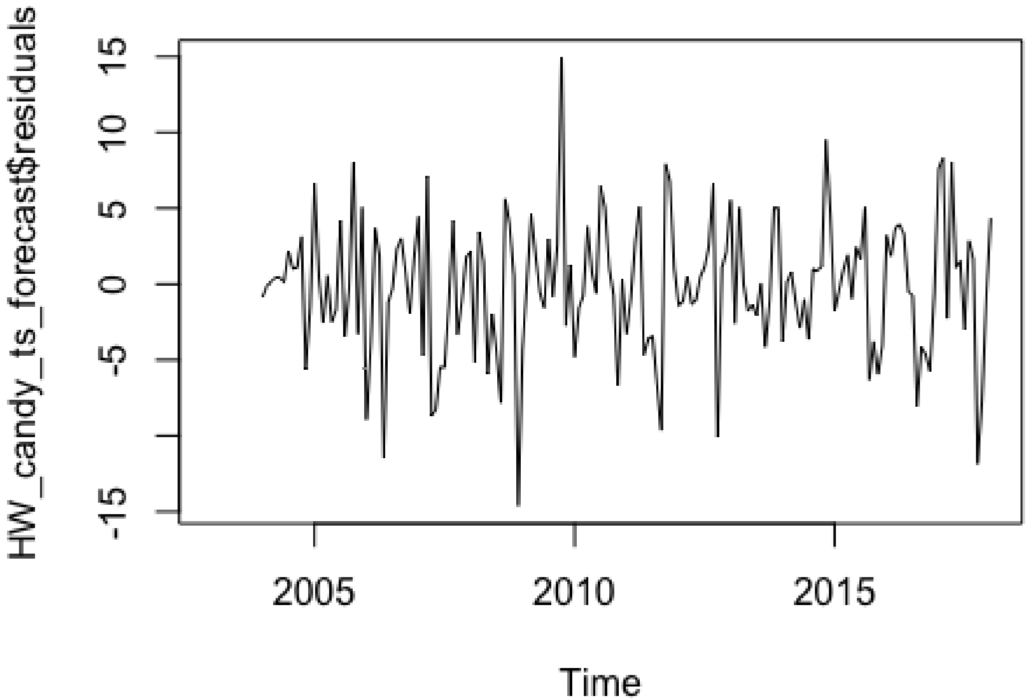
```
## [1] 0.249493
```

->Above we calculated the Standard deviation of residuals which means it is small and implies goodness of the model.

• Perform Residual Analysis for this technique.

o Do a plot of residuals. What does the plot indicate?

```r
plot(HW_candy_ts_forecast$residuals)
```

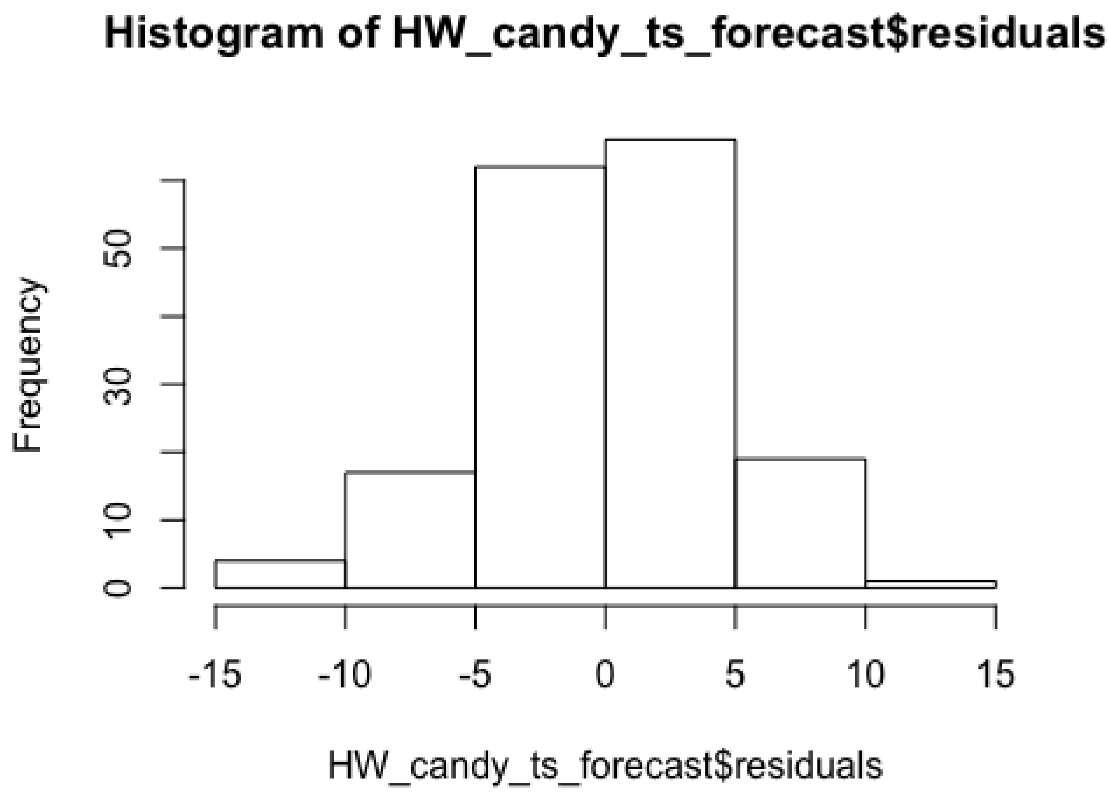



->-> By observing the above graph of residuals,we can see that residuals stays same accross the historical data.Hence the it can be considered constant over the time.

-> There are few spikes observed in the residues(2008,2010,2013,2017), which can be due to some special event occurance in the country which impacted the production of candy.

o Do a Histogram plot of residuals. What does the plot indicate?

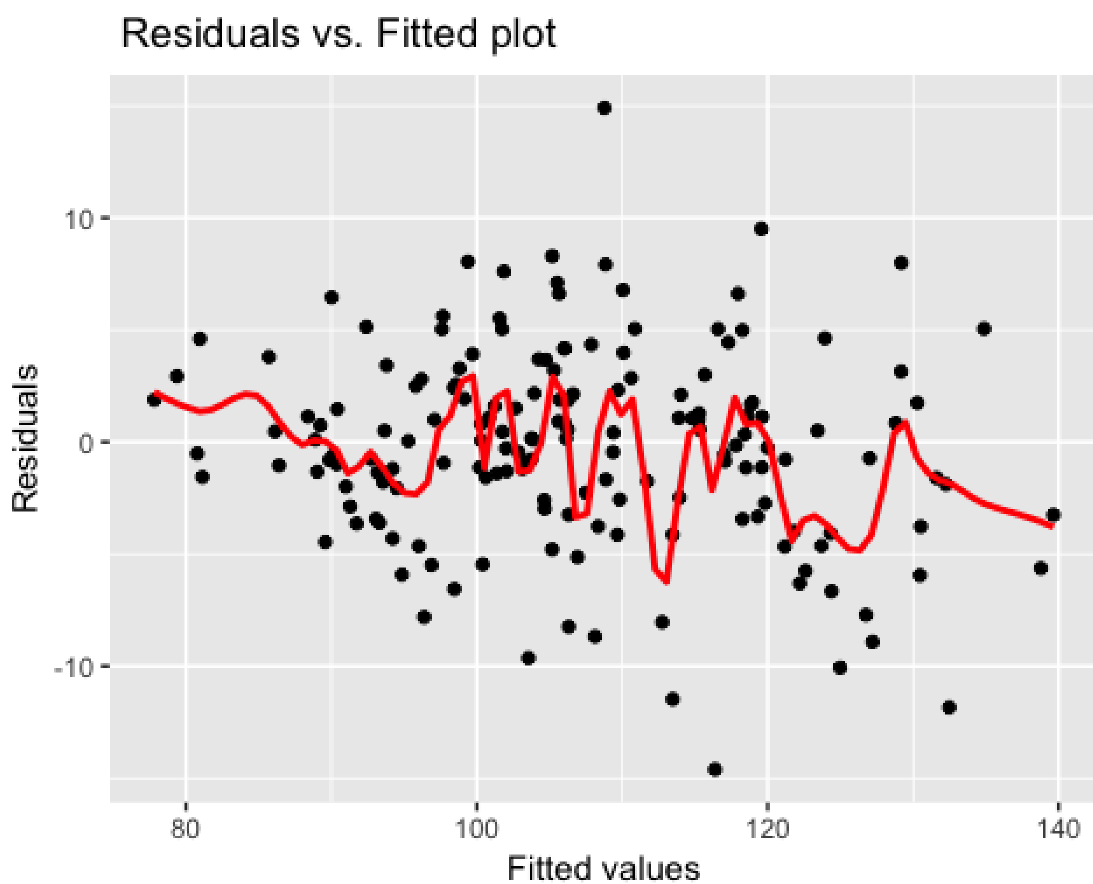
```
hist(HW_candy_ts_forecast$residuals)
```




->The histogram plot of the residuals suggests that the residuals can be considered to follow a normal distribution.

o Do a plot of fitted values vs. residuals. What does the plot indicate?

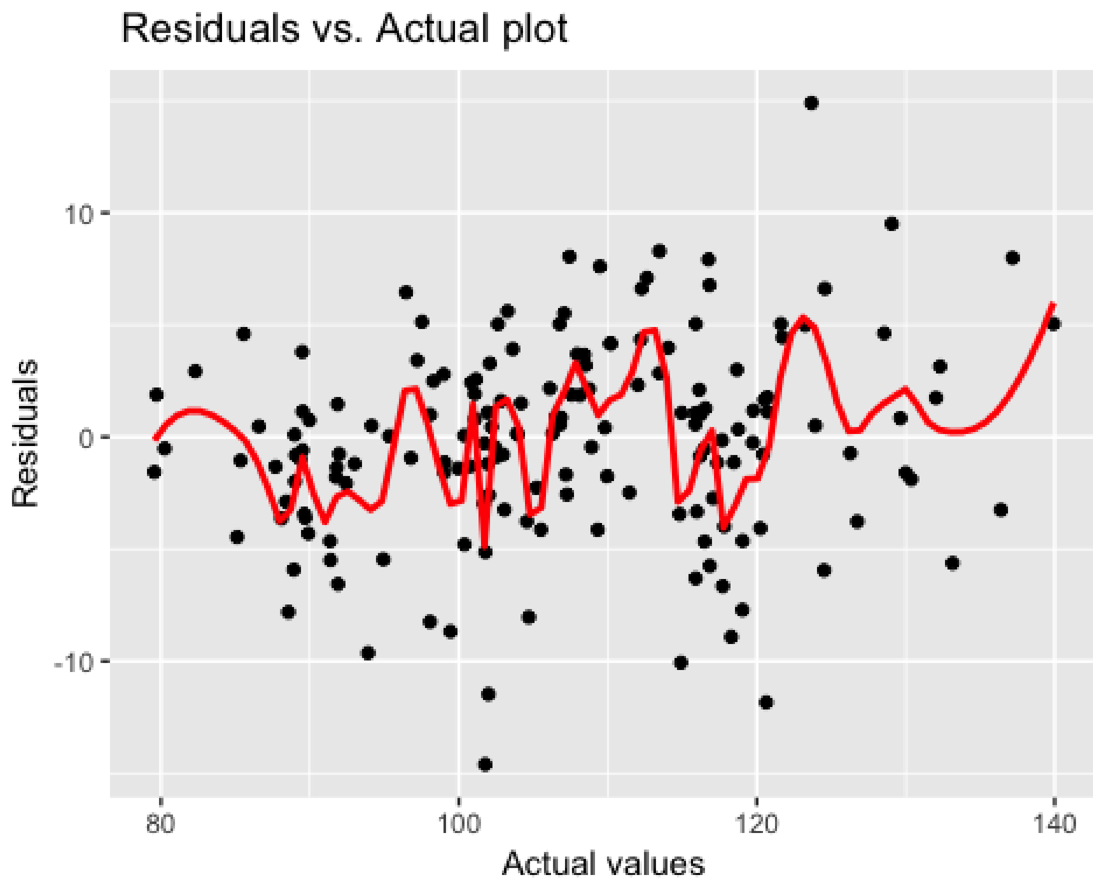
```
library(ggplot2)
qplot(y = HW_candy_ts_forecast$residuals, x = HW_candy_ts_forecast$fitted,
      ylab = "Residuals", xlab = "Fitted values",
      main = " Residuals vs. Fitted plot") +
  stat_smooth(method = "loess", span = 0.1, colour = I("red"), se = FALSE)

## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.
## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.

## Warning: Removed 12 rows containing non-finite values (stat_smooth).

## Warning: Removed 12 rows containing missing values (geom_point).
```




->In the above plot of Residuals VS Fitted values shows residuals has no pattern and they are randomly distributed among themself.There is no correlation among the residues.

o Do a plot of actual values vs. residuals. What does the plot indicate?

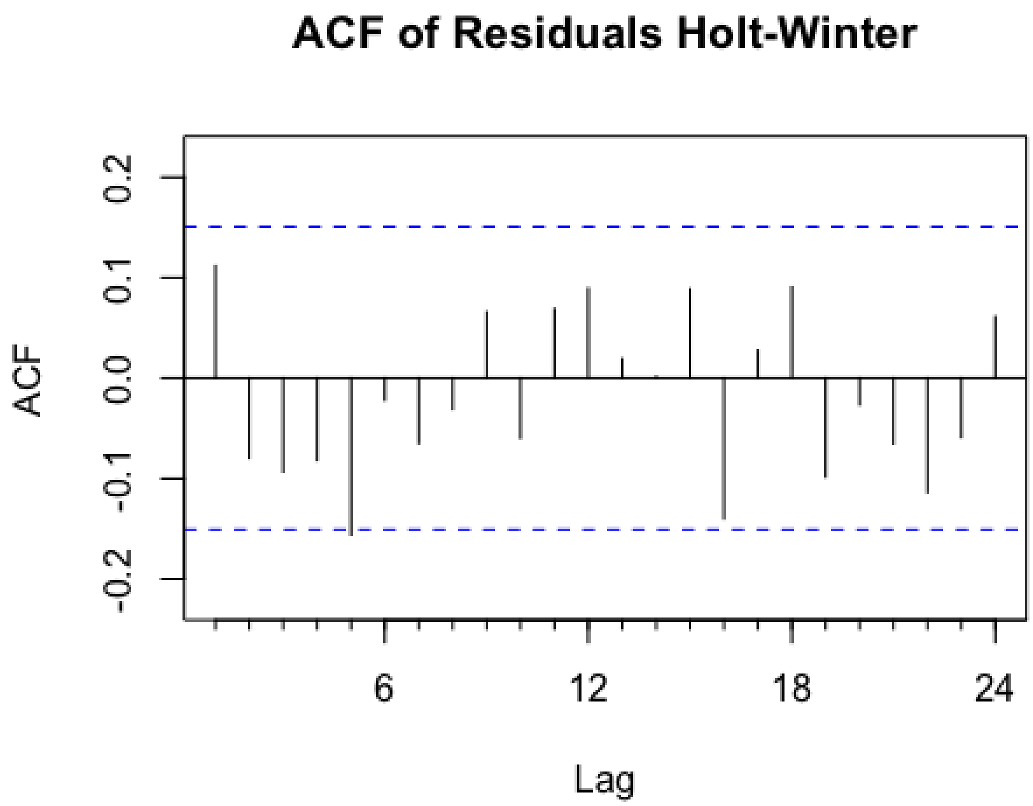
```
library(ggplot2)
qplot(y = HW_candy_ts_forecast$residuals, x = candy_ts,
      ylab = "Residuals", xlab = "Actual values",
      main = " Residuals vs. Actual plot") +
  stat_smooth(method = "loess", span = 0.1, colour = I("red"), se = FALSE)

## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.
## Don't know how to automatically pick scale for object of type ts. Defaulti
ng to continuous.

## Warning: Removed 12 rows containing non-finite values (stat_smooth).

## Warning: Removed 12 rows containing missing values (geom_point).
```




->In the above figure, we observe that the Residuals are spreaded sporadiclly and they dont have any pattern among them.There is little correlation among the residuals.

o Do an ACF plot of the residuals? What does this plot indicate?

```
Acf(HW_candy_ts_forecast$residuals, main = "ACF of Residuals Holt-Winter")
```




->Spikes shows the values of Autocorrelation with each lags. We can observe that amplitude of each spike is in the blue segment which implies they are insignificant.Hence the Autocorrelation is insignificant.

• Print the 5 measures of accuracy for this forecasting technique

```
accuracy_HW <- accuracy(HW_candy_ts_forecast)
accuracy_HW
```

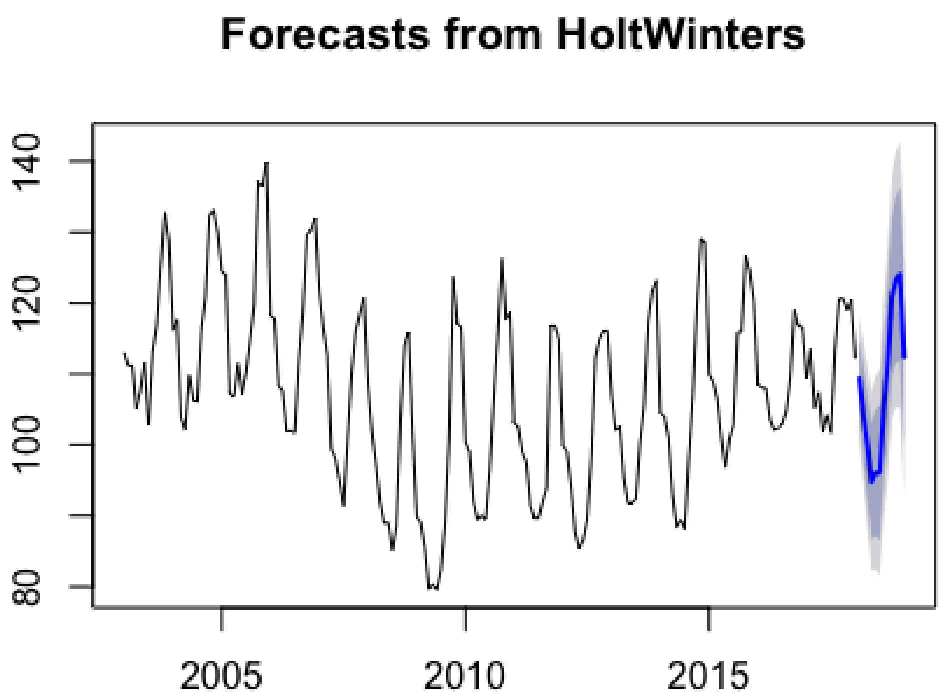
```
##                      ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -0.1873801 4.410365 3.349646 -0.2713261 3.124352 0.5523739
##                   ACF1
## Training set 0.1115922
```

• Forecast o Time series value for next year. Show table and plot

```
HW_candy_ts_forecast
```

```
##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Feb 2018      109.37024 103.70645 115.0340 100.70822 118.0323
## Mar 2018      104.14898  97.52684 110.7711  94.02130 114.2767
## Apr 2018       99.86726  92.40892 107.3256  88.46071 111.2738
## May 2018       94.79146  86.58165 103.0013  82.23564 107.3473
## Jun 2018       96.06856  87.17051 104.9666  82.46017 109.6769
```

```
## Jul 2018       96.07767  86.54093 105.6144  81.49248 110.6629
## Aug 2018      105.21682  95.08156 115.3521  89.71627 120.7174
## Sep 2018      111.45463 100.75427 122.1550  95.08984 127.8194
## Oct 2018      120.86181 109.62473 132.0989 103.67618 138.0474
## Nov 2018      123.29548 111.54617 135.0448 105.32647 141.2645
## Dec 2018      124.07774 111.83762 136.3178 105.35810 142.7974
## Jan 2019      112.42775  99.71577 125.1397  92.98645 131.8691

plot(HW_candy_ts_forecast)
```




• Summarize this forecasting technique

```
summary(HW_candy_ts_forecast)

##
## Forecast method: HoltWinters
##
## Model Information:
## Holt-Winters exponential smoothing with trend and additive seasonal compon
ent.
##
## Call:
## HoltWinters(x = candy_ts)
##
```

```
## Smoothing parameters:
##  alpha: 0.6058406
##  beta : 0
##  gamma: 0.6033215
##
## Coefficients:
##             [,1]
## a   108.28086742
## b     0.07459764
## s1    1.01477173
## s2   -4.28108430
## s3   -8.63739788
## s4  -13.78779419
## s5  -12.58529699
## s6  -12.65078438
## s7   -3.58622669
## s8    2.57698313
## s9   11.90956775
## s10  14.26863348
## s11  14.97629420
## s12   3.25171168
##
## Error measures:
##                      ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -0.1873801 4.410365 3.349646 -0.2713261 3.124352 0.5523739
##                   ACF1
## Training set 0.1115922
##
## Forecasts:
##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Feb 2018      109.37024 103.70645 115.0340 100.70822 118.0323
## Mar 2018      104.14898  97.52684 110.7711  94.02130 114.2767
## Apr 2018       99.86726  92.40892 107.3256  88.46071 111.2738
## May 2018       94.79146  86.58165 103.0013  82.23564 107.3473
## Jun 2018       96.06856  87.17051 104.9666  82.46017 109.6769
## Jul 2018       96.07767  86.54093 105.6144  81.49248 110.6629
## Aug 2018      105.21682  95.08156 115.3521  89.71627 120.7174
## Sep 2018      111.45463 100.75427 122.1550  95.08984 127.8194
## Oct 2018      120.86181 109.62473 132.0989 103.67618 138.0474
## Nov 2018      123.29548 111.54617 135.0448 105.32647 141.2645
## Dec 2018      124.07774 111.83762 136.3178 105.35810 142.7974
## Jan 2019      112.42775  99.71577 125.1397  92.98645 131.8691
```

o How good is the accuracy?

->From the above eorror measures, we can see that its accuracy is decent enough better then Naive method but not good as the ETS.

o What does it predict the value of time series will be in one year? 112.42775 with 95 percent confidence interval, It would be 92.986 low and 11.8691 high.

o Other observation During the residual analysis, we observed that they are normally distributed.

Accuracy Summary

• Show a table of all the forecast method above with their accuracy measures.

```
final_accuracy <- rbind(naive_accuracy,accuracy_ets,accuracy_HW)

rownames(final_accuracy) <- c("Naive Method", "ETS", "Holt-Winter")

final_accuracy
```

```
##                        ME     RMSE      MAE        MPE     MAPE      MASE
## Naive Method -0.004547778 7.422458 5.470242 -0.2333585 5.057813 0.9020712
## ETS          -0.055739135 3.961930 2.971197 -0.1133162 2.749518 0.4899657
## Holt-Winter  -0.187380106 4.410365 3.349646 -0.2713261 3.124352 0.5523739
##                   ACF1
## Naive Method 0.2547176
## ETS          0.0011844
## Holt-Winter  0.1115922
```

**• Separately define each forecast method and why it is useful. Show the best and worst forecast method for each of the accuracy measures.**

-> Naive Forecast: Naïve 1 forecasts are often used as a benchmark when assessing the accuracy of a set of forecasts. A ratio is obtained to show the upper bound of a forecasting method's accuracy relative to naïve 1 forecasts when the mean squared error is used to measure accuracy.It is know as no change forecast which has been observed while forecasting above.

->Simple Moving Average:

It is the weighted average of the previous n data. It is used when recent observations influence more than the previous observations. As new data comes in , newest value is added and oldest value is dropped. Equal weights are assigned to each observation which is not considering seasonality and trend of the time series. -> Simple Smoothing:

When forecaster believes more-recent observations are likely to contain more information, this is the technique to use. This method is suitable for forecasting data with no trend or seasonal pattern. The main aim is to estimate the current level. The level estimate is then used to forecast future values. Since the most recent period's forecast was created based on the previous period's demand and the previous period's forecast, which was based on the demand for the period before that and the forecast for the period before that.

->Holt Winters: Holt Winters has levels which are level, trend ans seasonality. Hence it is called Triple Exponential Smoothing There is additive method and multiplicative method. It is used when forecast data points in a series, provided that the series is "seasonal", i.e. repetitive over some period.

We can select anyone of the method based on the business needs.

Best model forecast method for each of the accuracy measures:

ME: Mean Error : -0.004547778 - lowest -> Naive Method

RMSE: Root Mean Squared Error: 3.961930 -> (Penalizes large errors) lowest -> ETS

MAE: Mean Absolute Error: 2.971197 -> lowest -> ETS

MPE: Mean Percentage Error: -0.1133162 -> closest to zero -> ETS

MAPE: Mean Absolute Percentage Error: 2.749518 -> lowest -> ETS

MASE: Mean Absolute Scaled Error: 0.4899657 -> lowest -> ETS

ACF1: Autocorrelation of errors at lag 1: 0.0011844 -> lowest -> ETS

Worst model based on the accuracy measures:

ME: Mean Error : -0.187380106 - High -> Holt-Winter.

RMSE: Root Mean Squared Error: 7.422458 -> (Penalizes large errors) High -> Naive Method

MAE: Mean Absolute Error: 5.470242 -> high -> Naive Method

MPE: Mean Percentage Error: -0.2713261 - high -> Holt-Winter

MAPE: Mean Absolute Percentage Error: 5.057813 -> high -> Naive Method

MASE: Mean Absolute Scaled Error: 0.9020712 -> High -> Naive Method

ACF1: Autocorrelation of errors at lag 1: 0.2547176 -> High -> Naive Method

Conclusion

• Summarize your analysis of time series value over the time-period.

-> ETS came out as the best model for the forecasting among all.It is due to the less error rate in accuracy measurement.

• Based on your analysis and forecast above, do you think the value of the time series will increase, decrease or stay flat over the next year? How about next 2 years?

-> Due to the seasonality the forecast will follow the same pattern. hence it will increase than decrease in the due course.

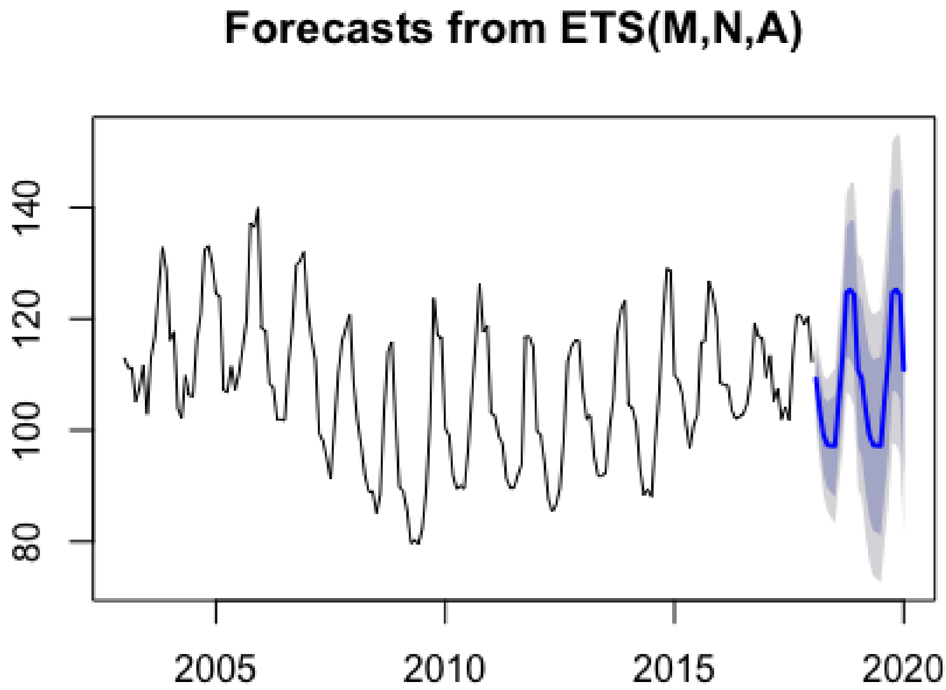
```
ets_candy<-ets(candy_ts)
ets_candy

## ETS(M,N,A)
##
## Call:
```

```
##  ets(y = candy_ts)
##
##   Smoothing parameters:
##     alpha = 0.7504
##     gamma = 1e-04
##
##   Initial states:
##     l = 116.5249
##     s=15.3902 16.2337 15.7225 3.9562 -3.3893 -11.7773
##            -11.7272 -11.6073 -9.7897 -5.2116 0.3267 1.8729
##
##   sigma:  0.0361
##
##      AIC     AICc      BIC
## 1459.573 1462.482 1507.551
```

```
forecast_ets_candy <- forecast.ets(ets_candy, h=24)
forecast_ets_candy
```

```
##          Point Forecast     Lo 80    Hi 80     Lo 95    Hi 95
## Feb 2018      109.28137 104.21953 114.3432 101.53994 117.0228
## Mar 2018      103.74316  97.61639 109.8699  94.37307 113.1133
## Apr 2018       99.16507  92.19556 106.1346  88.50612 109.8240
## May 2018       97.34774  89.61964 105.0758  85.52863 109.1668
## Jun 2018       97.22752  88.79154 105.6635  84.32580 110.1292
## Jul 2018       97.17709  88.08769 106.2665  83.27605 111.0781
## Aug 2018      105.56456  95.67847 115.4507  90.44509 120.6840
## Sep 2018      112.91012 102.19973 123.6205  96.52999 129.2903
## Oct 2018      124.67594 113.00564 136.3462 106.82776 142.5241
## Nov 2018      125.18774 112.72384 137.6516 106.12585 144.2496
## Dec 2018      124.34403 111.15523 137.5328 104.17350 144.5146
## Jan 2019      110.82761  97.19179 124.4634  89.97343 131.6818
## Feb 2019      109.28137  95.13287 123.4299  87.64310 130.9196
## Mar 2019      103.74316  89.17528 118.3111  81.46350 126.0228
## Apr 2019       99.16507  84.21889 114.1113  76.30685 122.0233
## May 2019       97.34774  82.02891 112.6666  73.91962 120.7759
## Jun 2019       97.22752  81.53557 112.9195  73.22875 121.2263
## Jul 2019       97.17709  81.12037 113.2338  72.62046 121.7337
## Aug 2019      105.56456  89.03988 122.0892  80.29224 130.8369
## Sep 2019      112.91012  95.87566 129.9446  86.85816 138.9621
## Oct 2019      124.67594 107.01849 142.3334  97.67121 151.6807
## Nov 2019      125.18774 106.99248 143.3830  97.36049 153.0150
## Dec 2019      124.34403 105.64132 143.0467  95.74071 152.9473
## Jan 2020      110.82761  91.80364 129.8516  81.73295 139.9223
```

```
plot(forecast_ets_candy)
```




The value after 2 years would be point forecast of 110.827 but with 95 percent confidence interval it could have high of 139.92 and low of 81.73.

• Rank forecasting methods that best forecast for this time series based on historical values.

Following are the Forecasting Method Ranks: ->Simple Smoothing (ETS) ->Holt-Winter ->Naive Method