Business Forecasting Midterm - Aman Mahajan (RUID:182001805)

Import Data

```
library(readr)
IPG3113N_Spring18 <- read_csv("/Users/amanmahajan/Desktop/Business Forecasting/IPG311
3N_Spring18.csv")
```

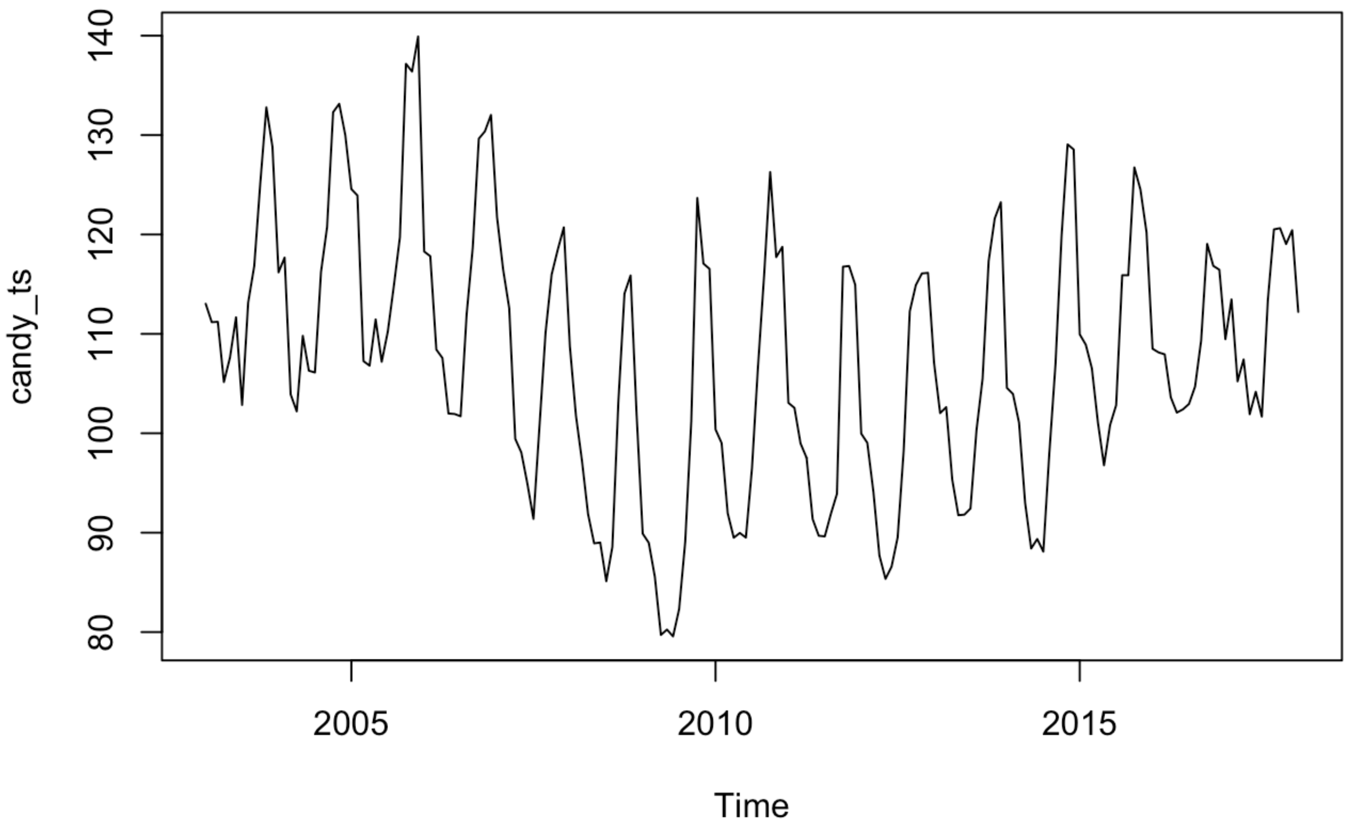
```
## Parsed with column specification:
## cols(
##   DATE = col_character(),
##   IPG3113N = col_double()
## )
```

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

Plot and Inference

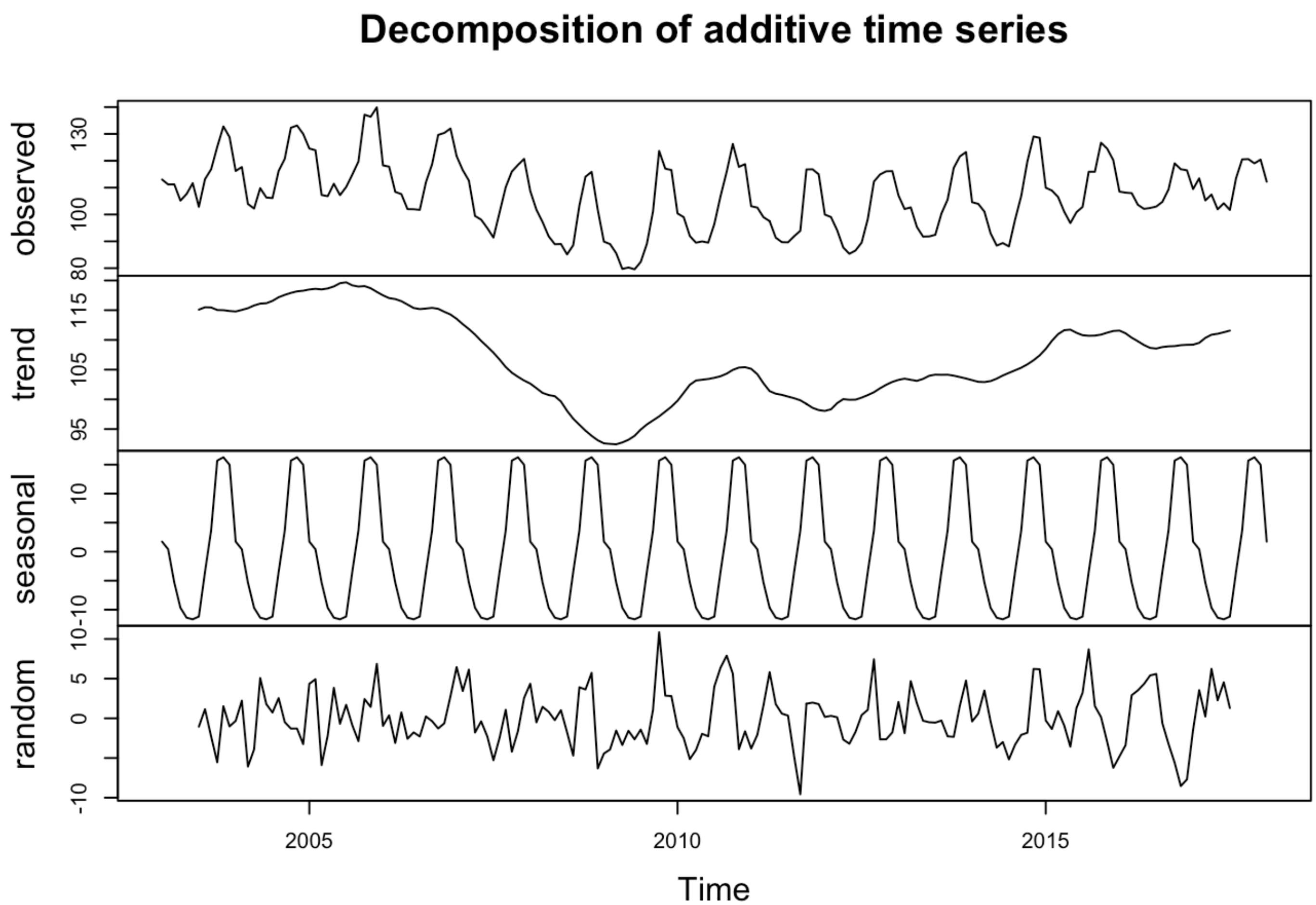
1. Show a time series plot

```
plot(candy_ts)
```



2. Please summaries your observations of the times series plot

```
candy_ts_d = decompose(candy_ts)
plot(candy_ts_d)
```



From the above plot, we see that the data is having Seasonality but no Trend. The plot is showing Cyclic Behavior.

Central Tendency

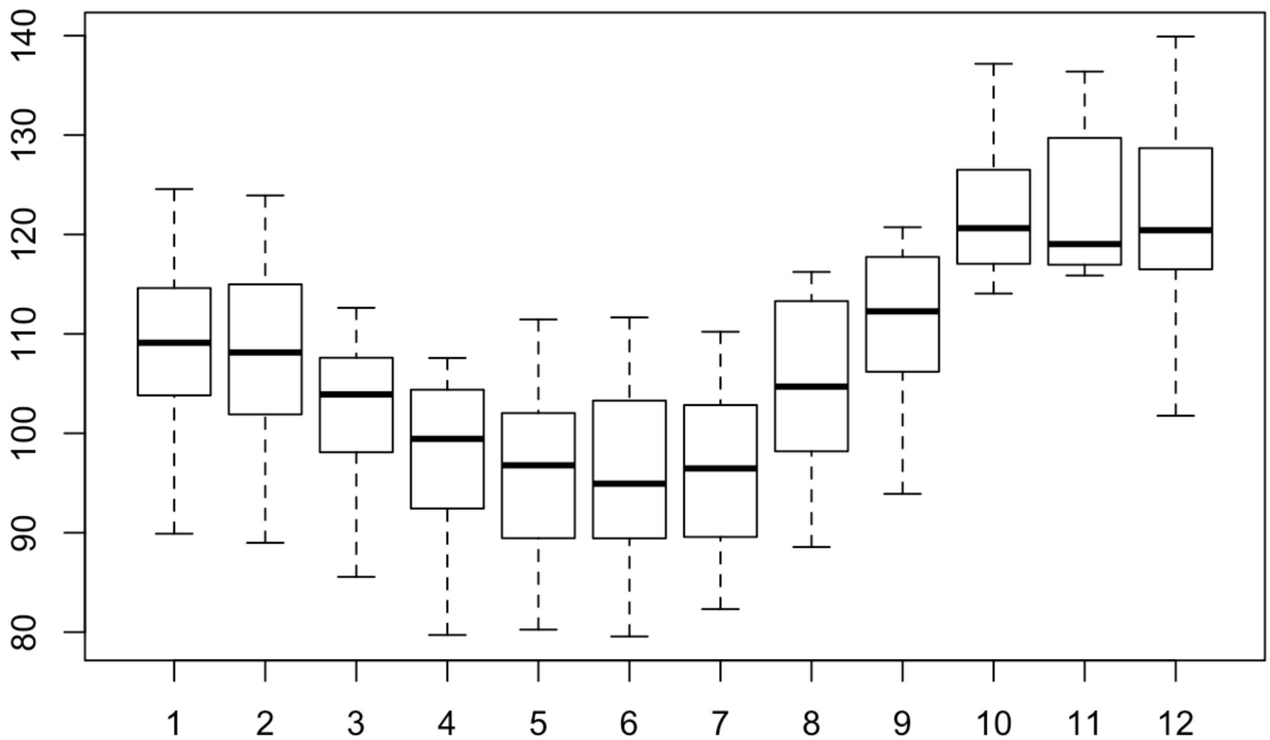
1. What are the min, max, mean, median, 1st and 3rd Quartile values of the times series?

```
summary(candy_ts)
```

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

2. Show the box plot

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



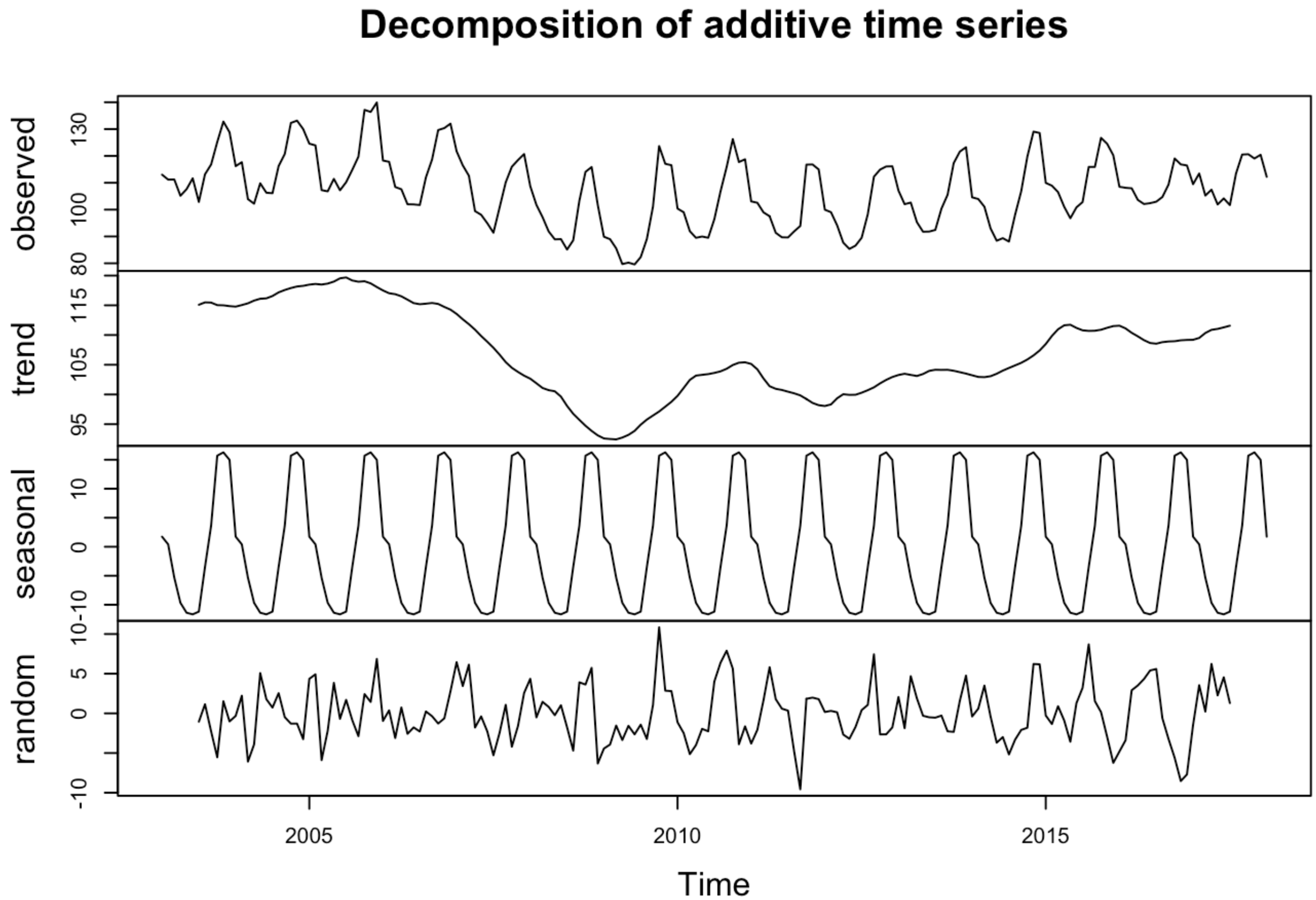
3. Can you summarize your observation about the time series from the summary stats and box plot?

Answer: The average monthly candy production is 107.19. The maximum and minimum values are 139.92 and 79.57 respectively. The box plot shows that production starts increasing from 2nd quarter and the increase goes on to 3rd quarter. Since, we have the values of 1st and 3rd quartile, we can also find the interquartile range.

Decomposition

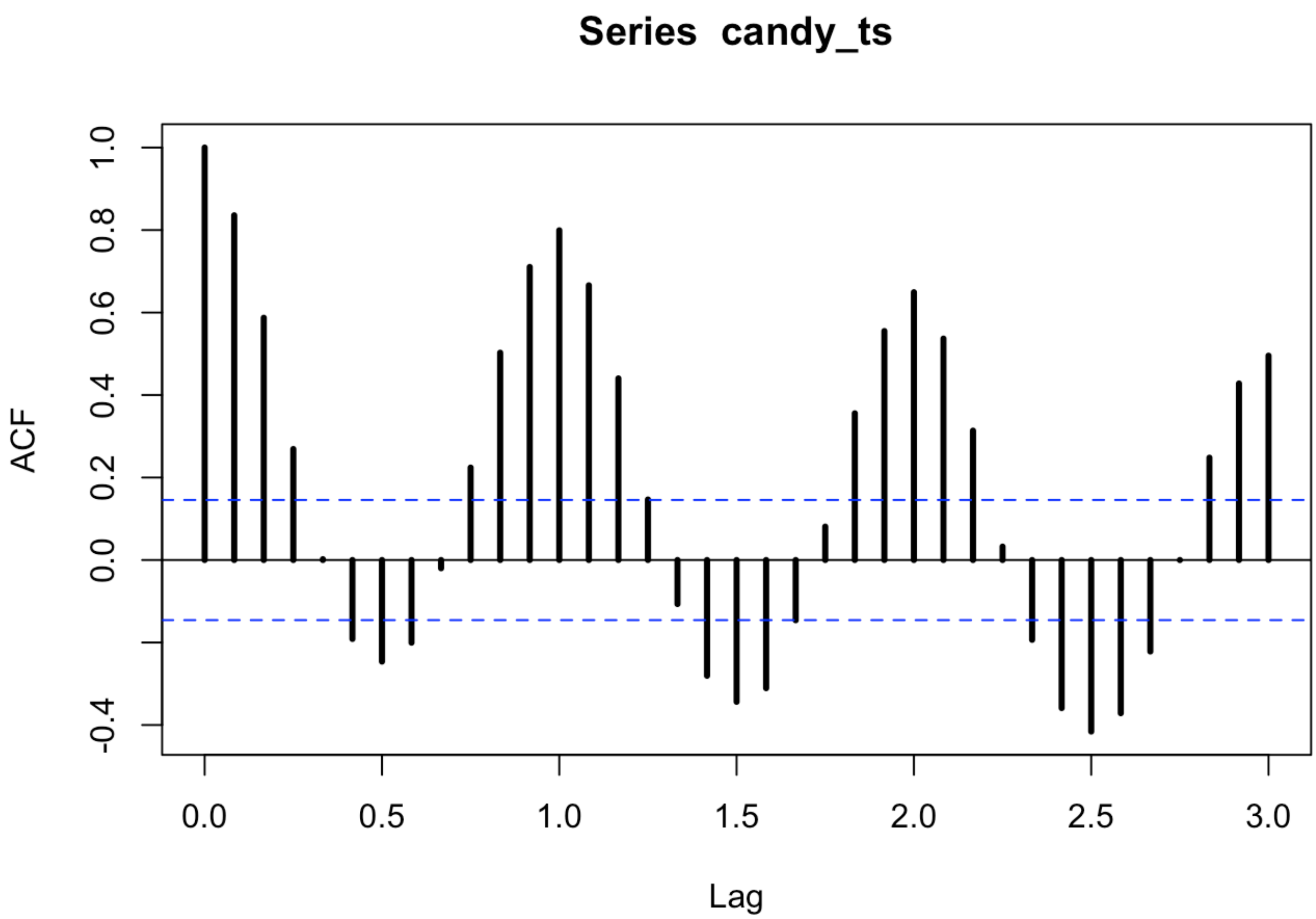
1. Plot the decomposition of the time series.

```
plot(candy_ts_d)
```



2. Is the times series seasonal?

```
acf(candy_ts,lag=36,lwd=3)
```



Answer: Yes, we can see a seasonal trend through the above plot.

3. Is the decomposition additive or multiplicative?

```
library(forecast)
ets(candy_ts)
```

```
## 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
```

Answer: Seasonality is Additive.

4. If seasonal, what are the values of the seasonal monthly indices?

```
smi=candy_ts_d$seasonal
smi
```

```
##             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
```

5. For which month is the value of time series high and for which month is it low?

Maximum value of Time Series:

```
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
max=time(as.zoo(smi))[which.max(smi)]
max
```

```
## [1] "Nov 2003"
```

-> Value of time series is maximum for month of November.

Minimum Value of Time Series:

```
min=time(as.zoo(smi))[which.min(smi)]
min
```

```
## [1] "Jun 2003"
```

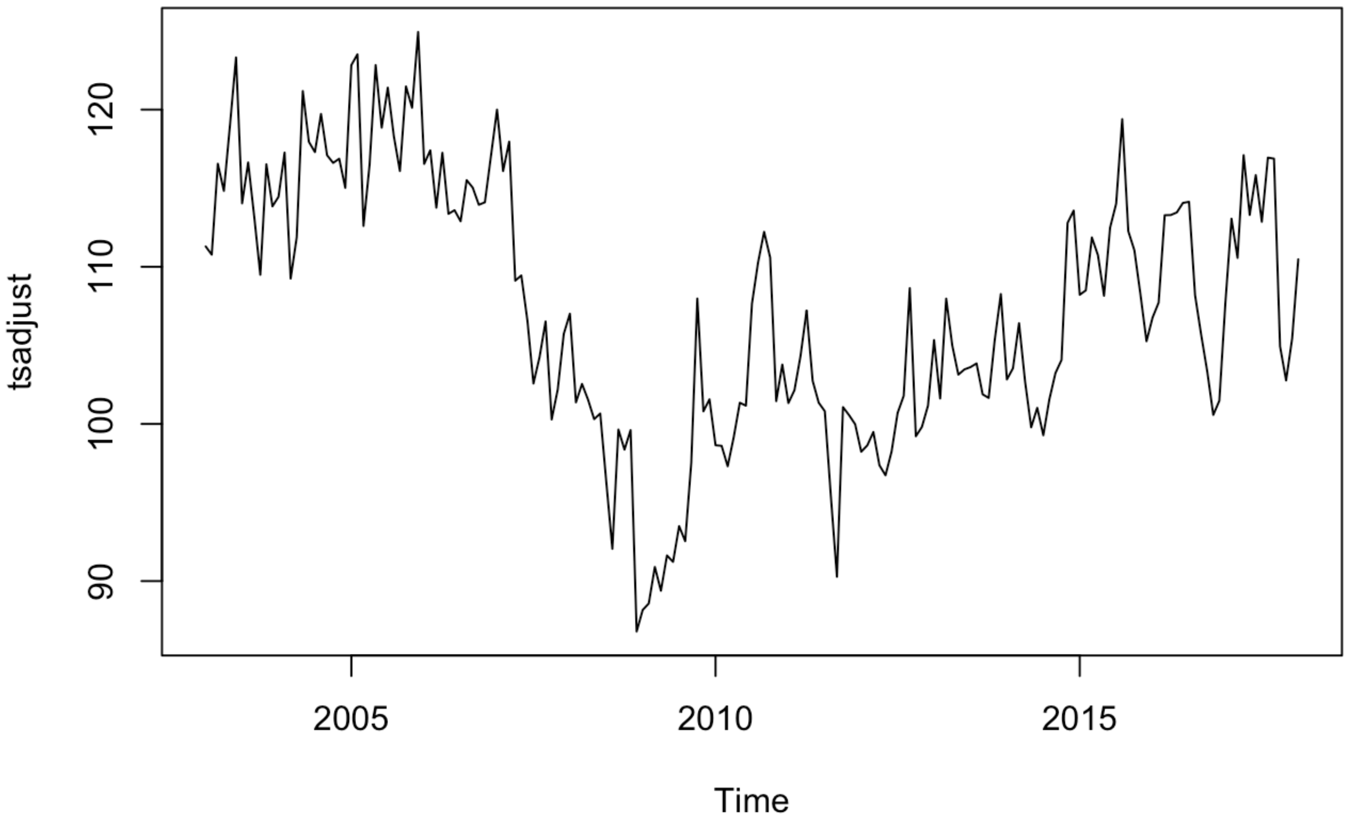
-> Value of time series is minimum for month of June.

6. Can you think of the reason behind the value being high in those months and low in those months?

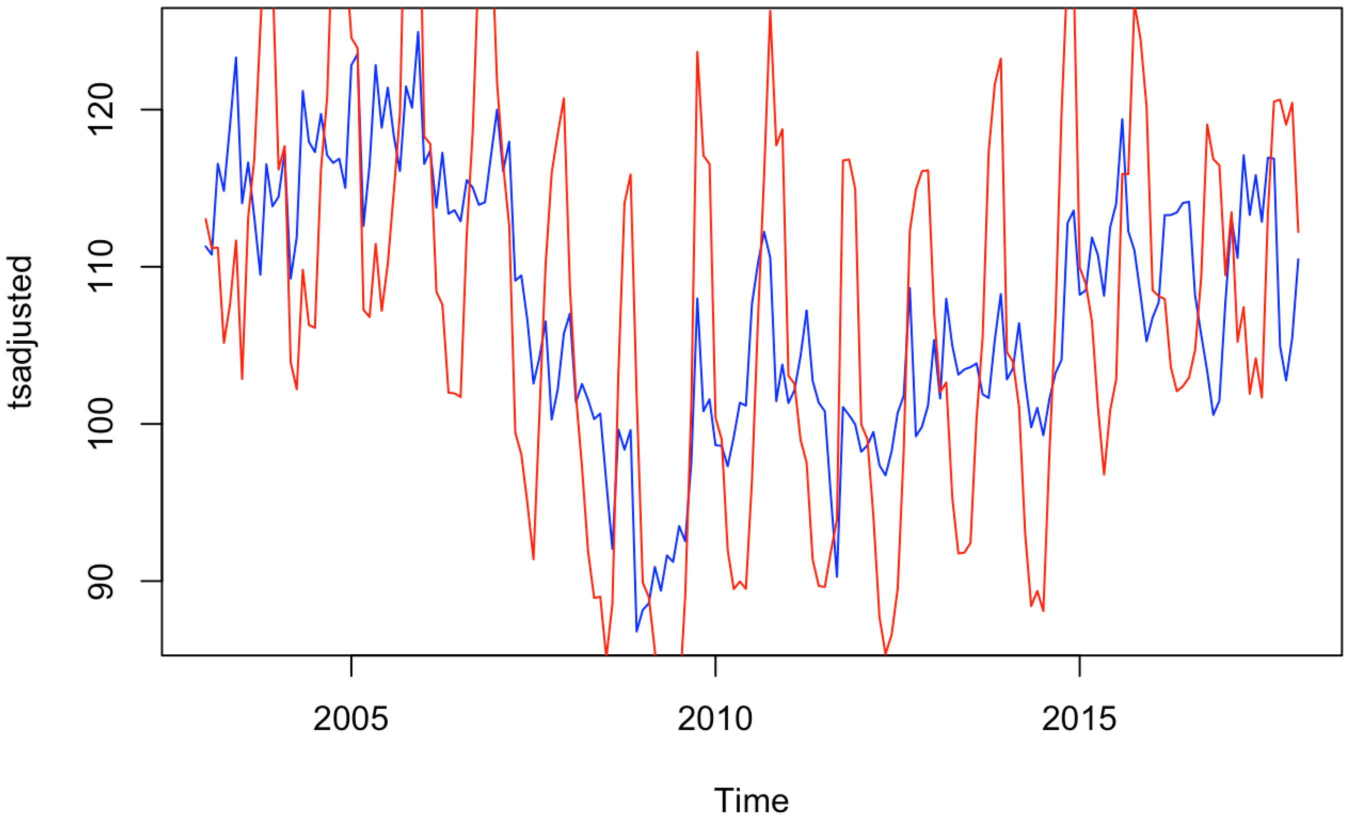
Answer: Value of time series is high during the month of November since it is the festive season and Halloween is during the same month on which day, the consumption of candies would be maximum by the consumers, so thereby, production is also maximum. On the other hand, there is low consumption during June as people would not be buying bulk quantities of candies now since the festive season has passed. Also, people tend do loose weight during the summers, that could be another reason for low consumption.

7. 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?

```
library(forecast)
tsadjust<- seasadj(candy_ts_d)
plot(tsadjust)
```




```
tsadjusted = seasadj(candy_ts_d)
plot(tsadjusted, col='blue')
lines(candy_ts, col='red')
```

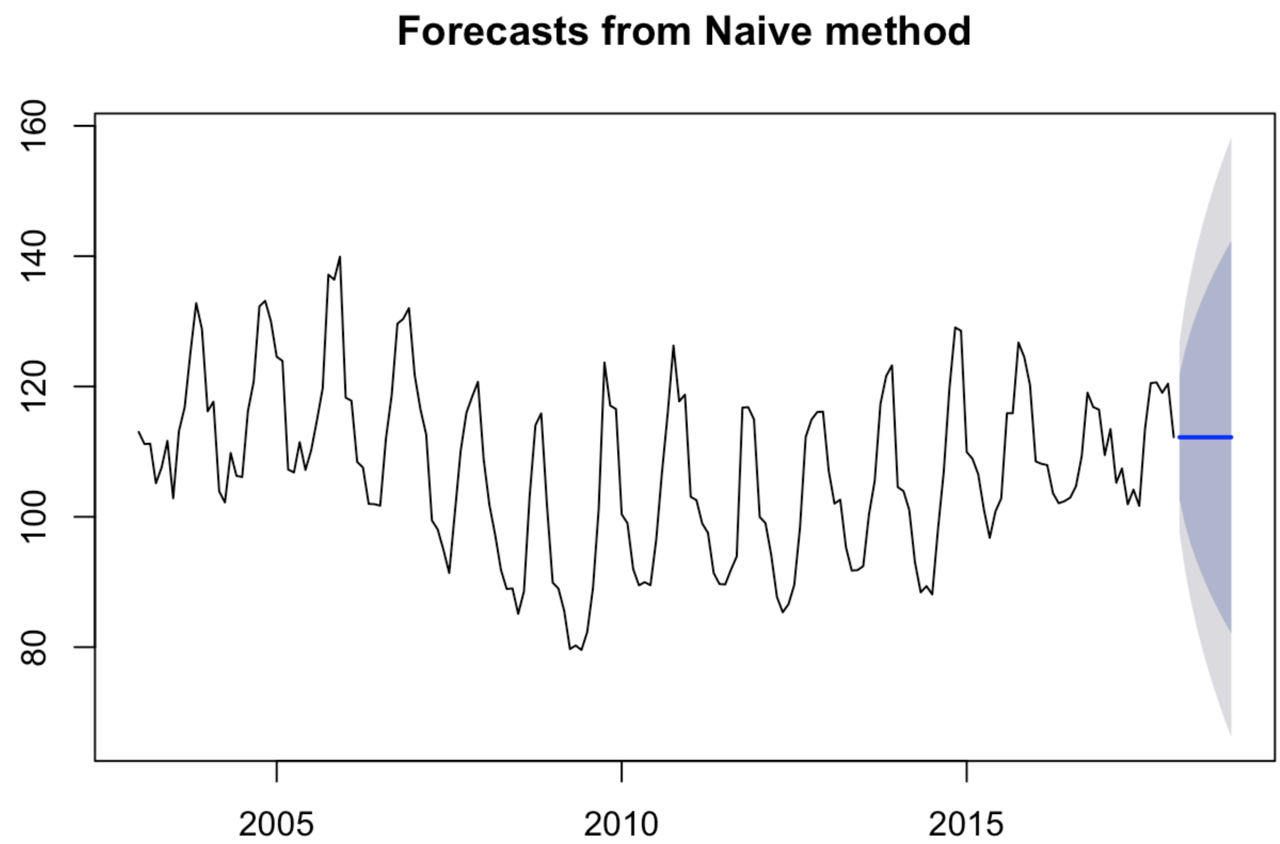


Answer: Yes, there are large fluctuations.

Naive Method

1. Output

```
library(forecast)
nm = naive(candy_ts)
plot(nm)
```

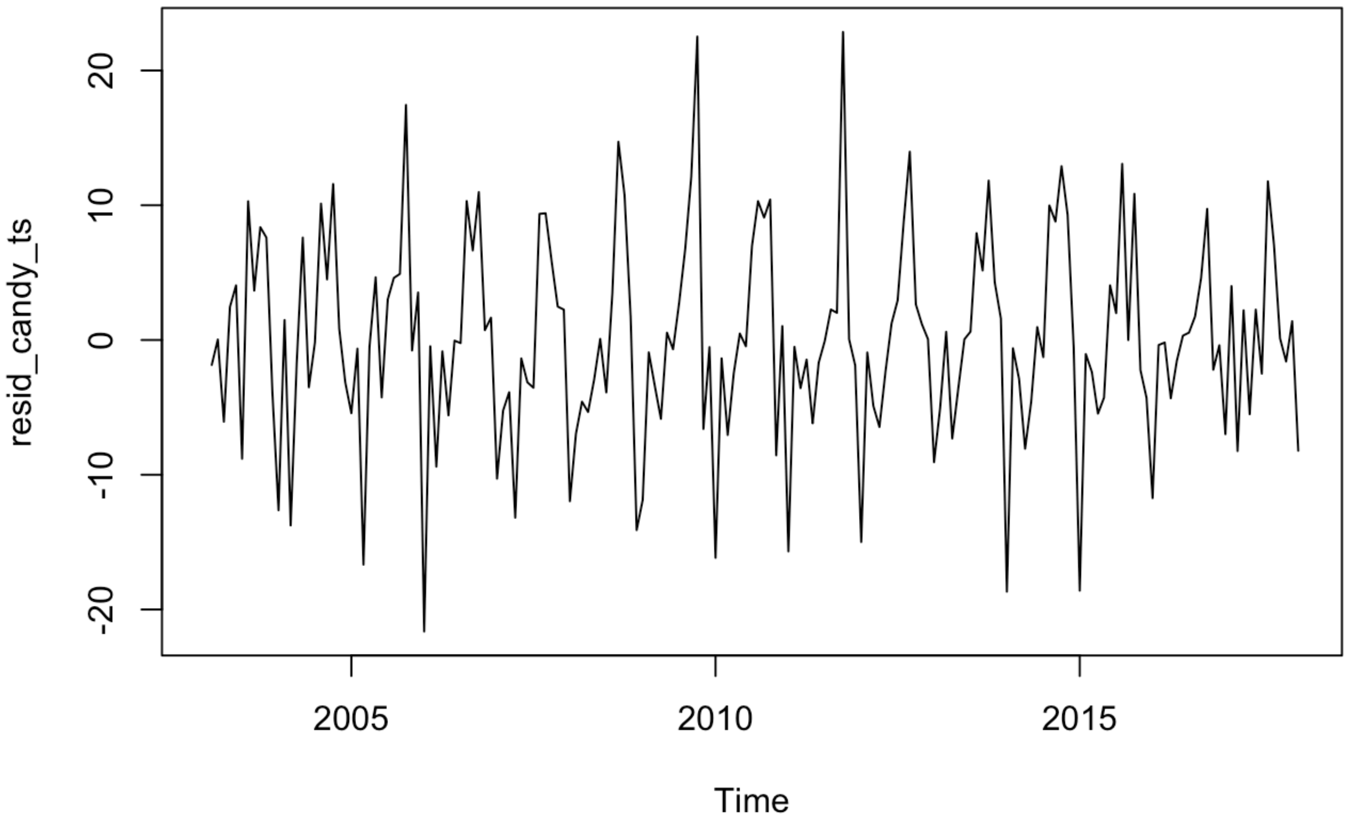


2. Perform Residual Analysis for this technique.

```
resid_candy_ts<- resid(nm)
```

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

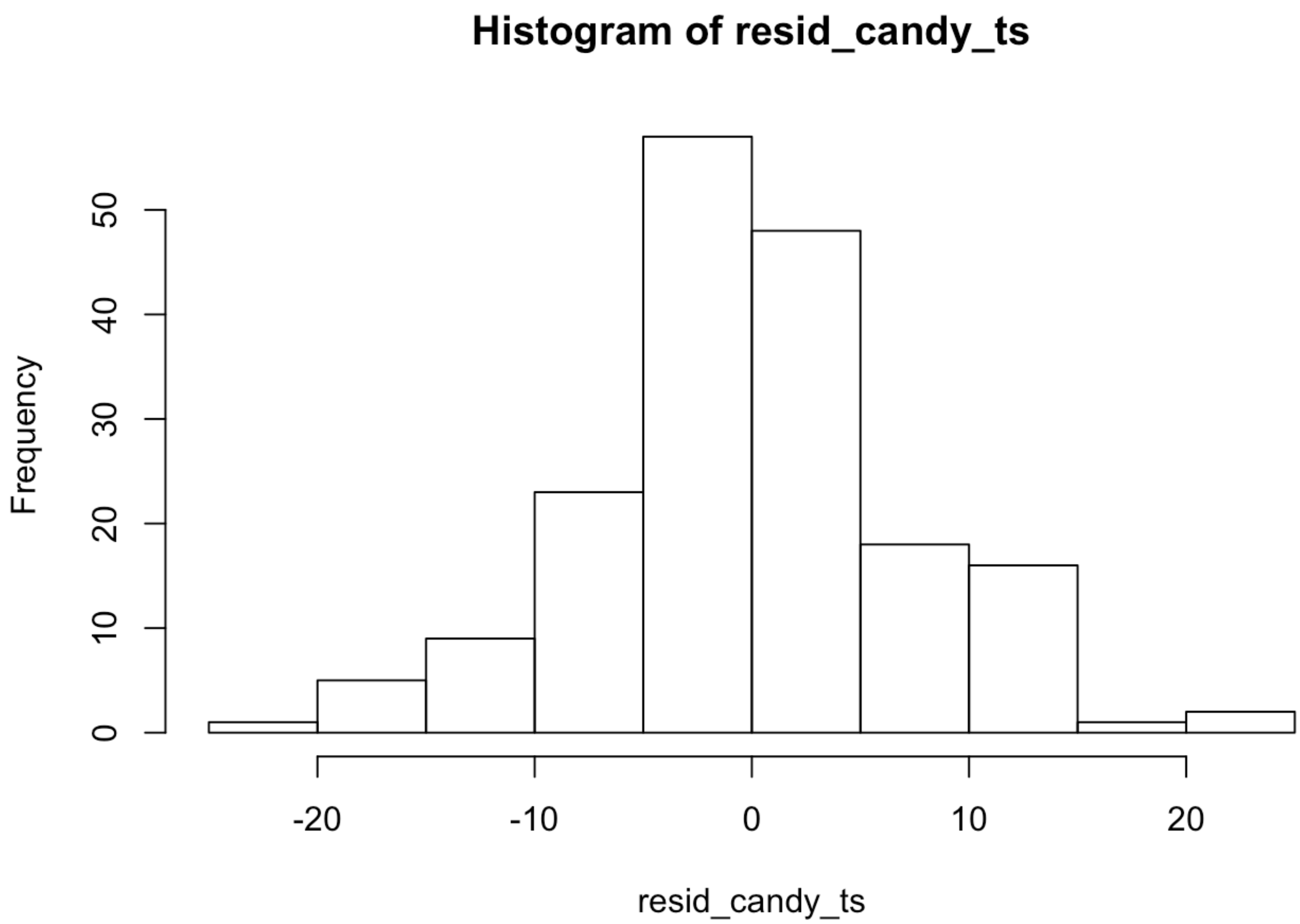
```
plot(resid_candy_ts)
```



Answer: The residual plot is more or less constant apart from occasional downward spike in 2006 and upward spikes in 2009 and 2012. So there are not many outliers in our data set.

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

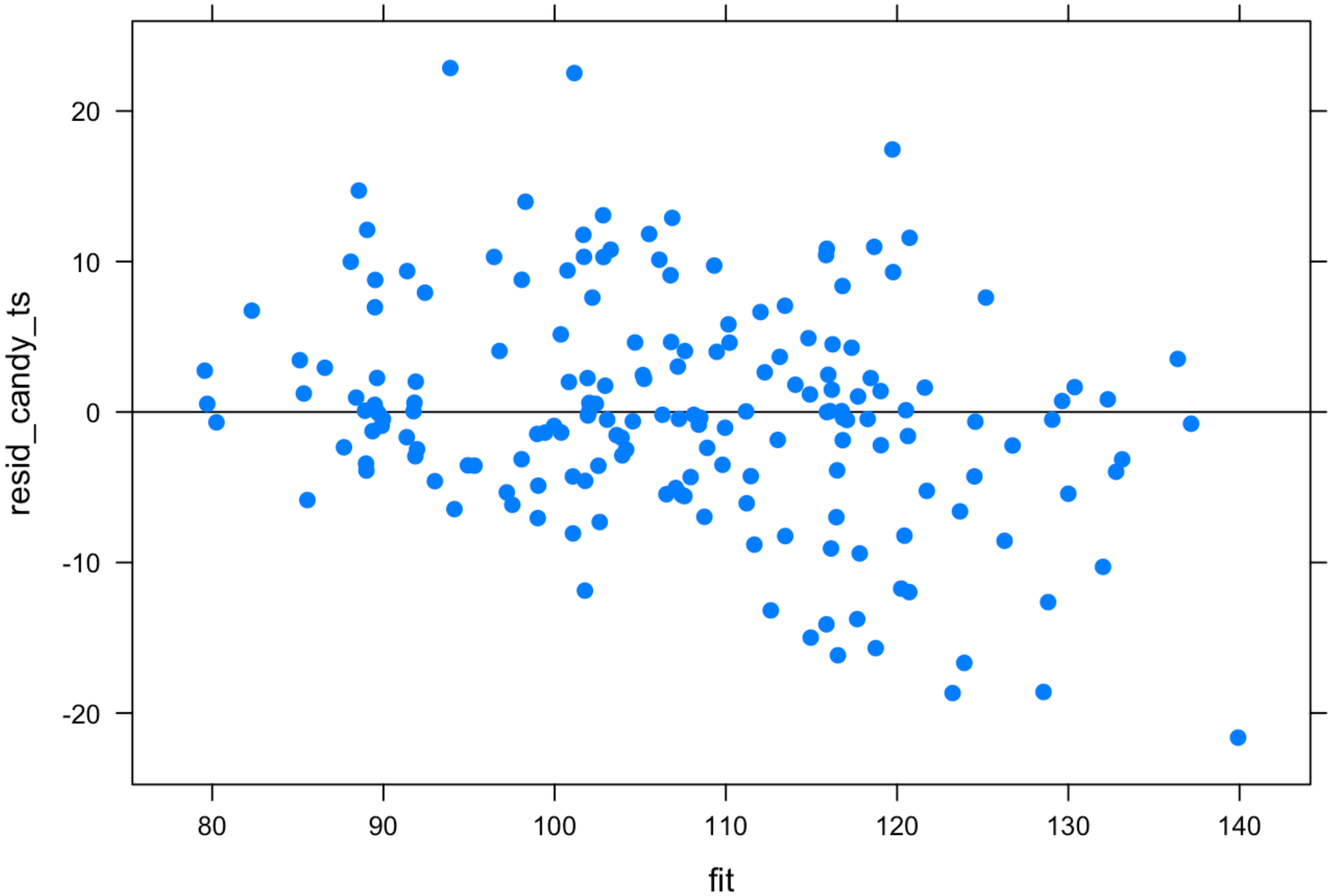
```
hist(resid_candy_ts)
```



Answer: The histogram plot of residuals suggests that the residuals follow a Normal Distribustion Curve.

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

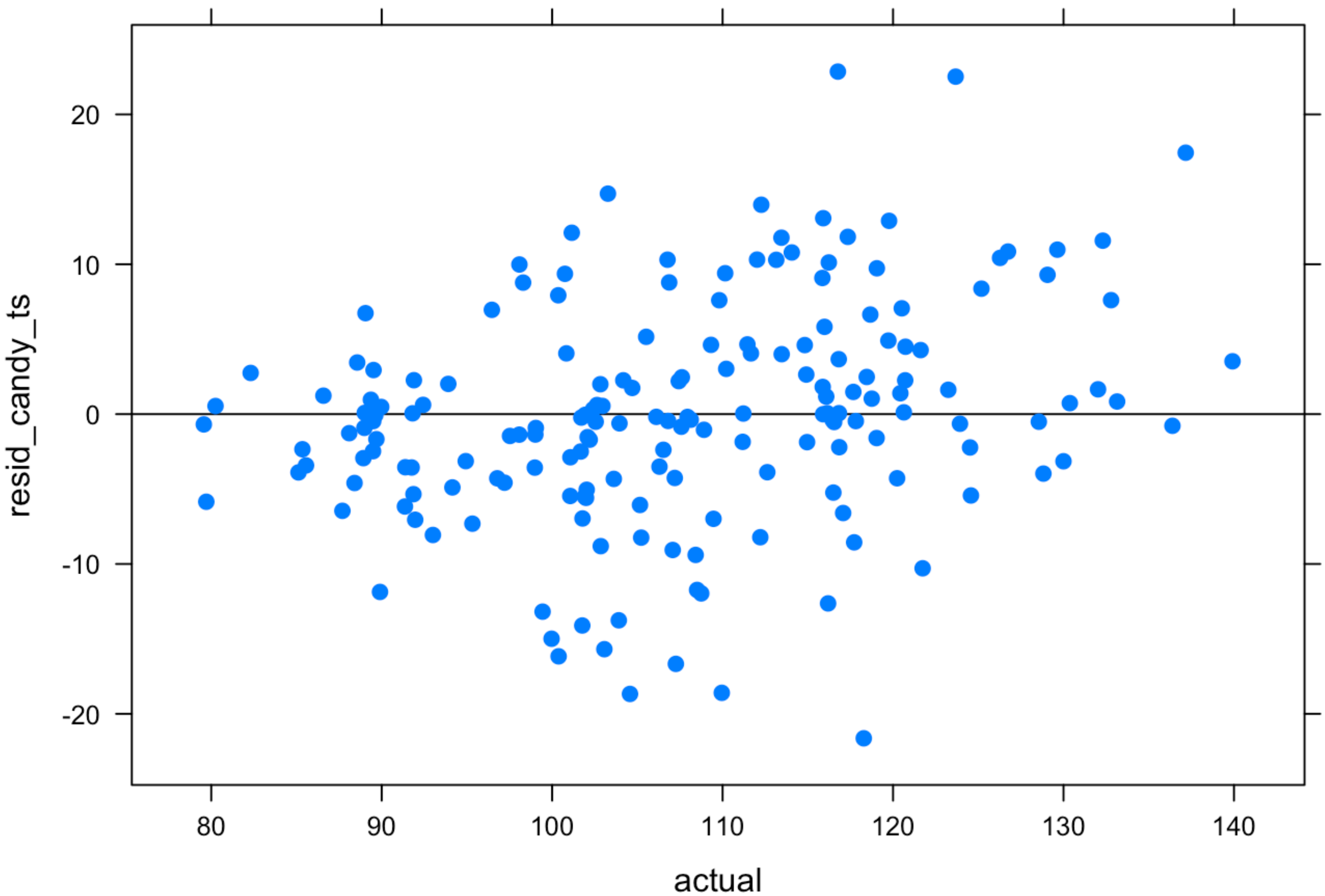
```
library(lattice)
library(fit.models)
fit <- fitted(nm)
xyplot(resid_candy_ts~fit, pch=16, cex = 1, abline=0)
```



Answer: The plot indicates no pattern between residuals and fitted values. Thus, there is no heteroscedasticity in the residuals which means the data has equal variations.

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

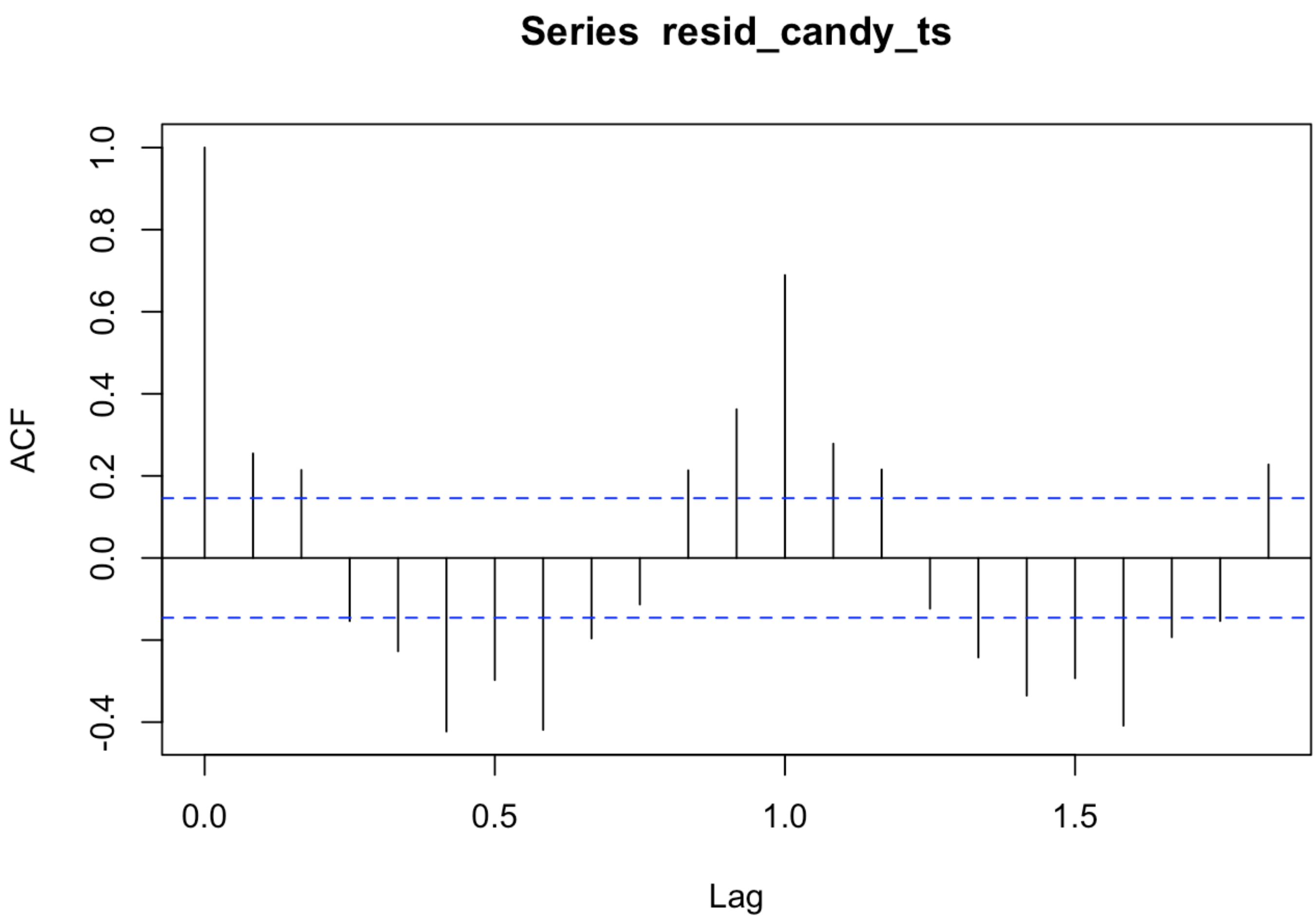
```
library(lattice)
actual <- nm$x
xyplot(resid_candy_ts~actual, pch=16, cex = 1, abline=0)
```



Answer: The plot indicates no pattern between residuals and fitted values. Thus, there is no heteroscedasticity in the residuals which means the data has equal variations.

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

```
acf(resid_candy_ts, na.action = na.pass)
```



Answer: Spikes shows the values of Autocorrelation with each lags. The Plot shows the values of ACF from -1 to +1. We can see a pattern among the lags which is repeating periodically.

3. Print the 5 measures of accuracy for this forecasting technique

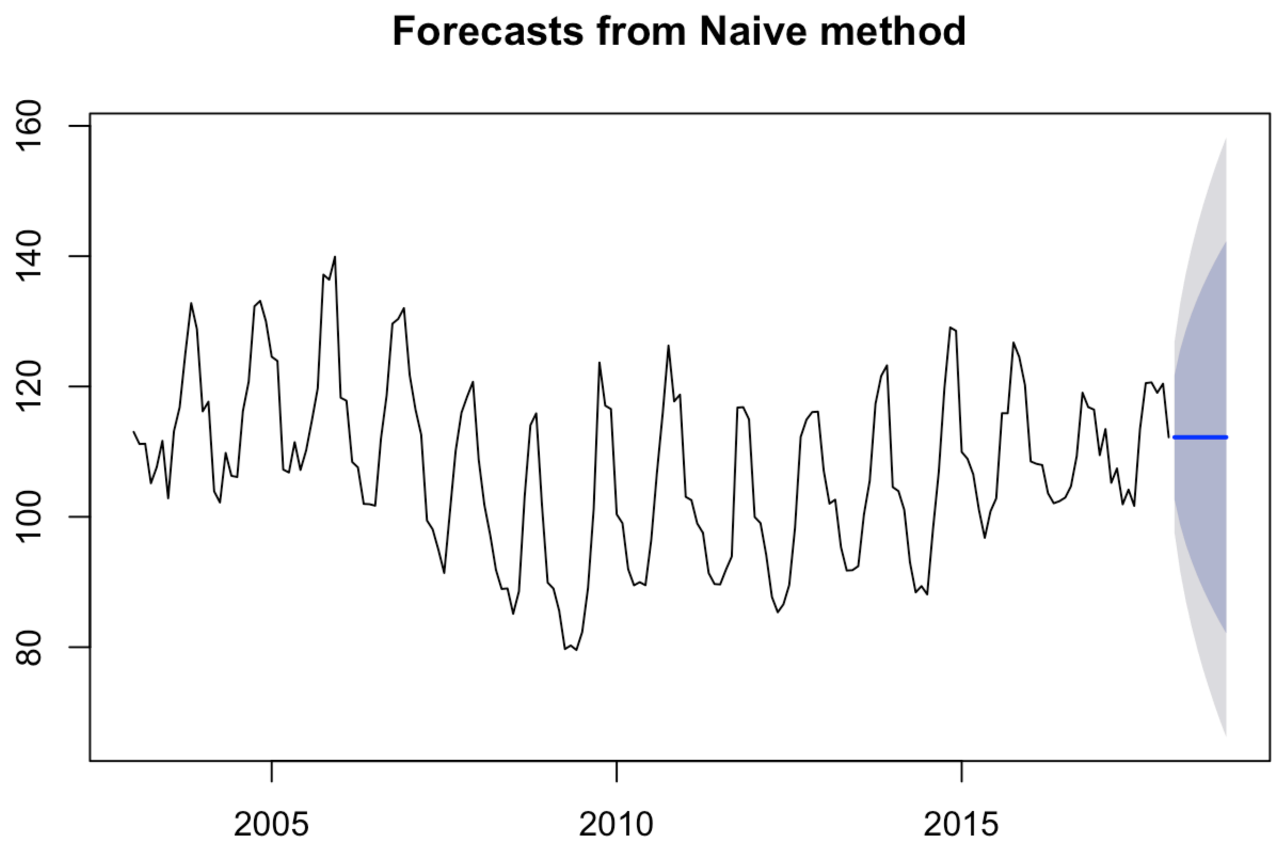
```
accuracy_nm = accuracy(nm)
accuracy_nm
```

```
##                        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
```

4. Forecast

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

```
nm = naive(candy_ts)
plot(nm)
```



5. Summarize this forecasting technique

```
summary(nm)
```

```
##
## Forecast method: Naive method
##
## Model Information:
## Call: naive(y = candy_ts)
##
## 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
```

q. How good is the accuracy?

Answer: The MASE and MAPE value are not too high which indicates the accuracy is good.

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

Answer: The value of time series in one year will be 112.2117

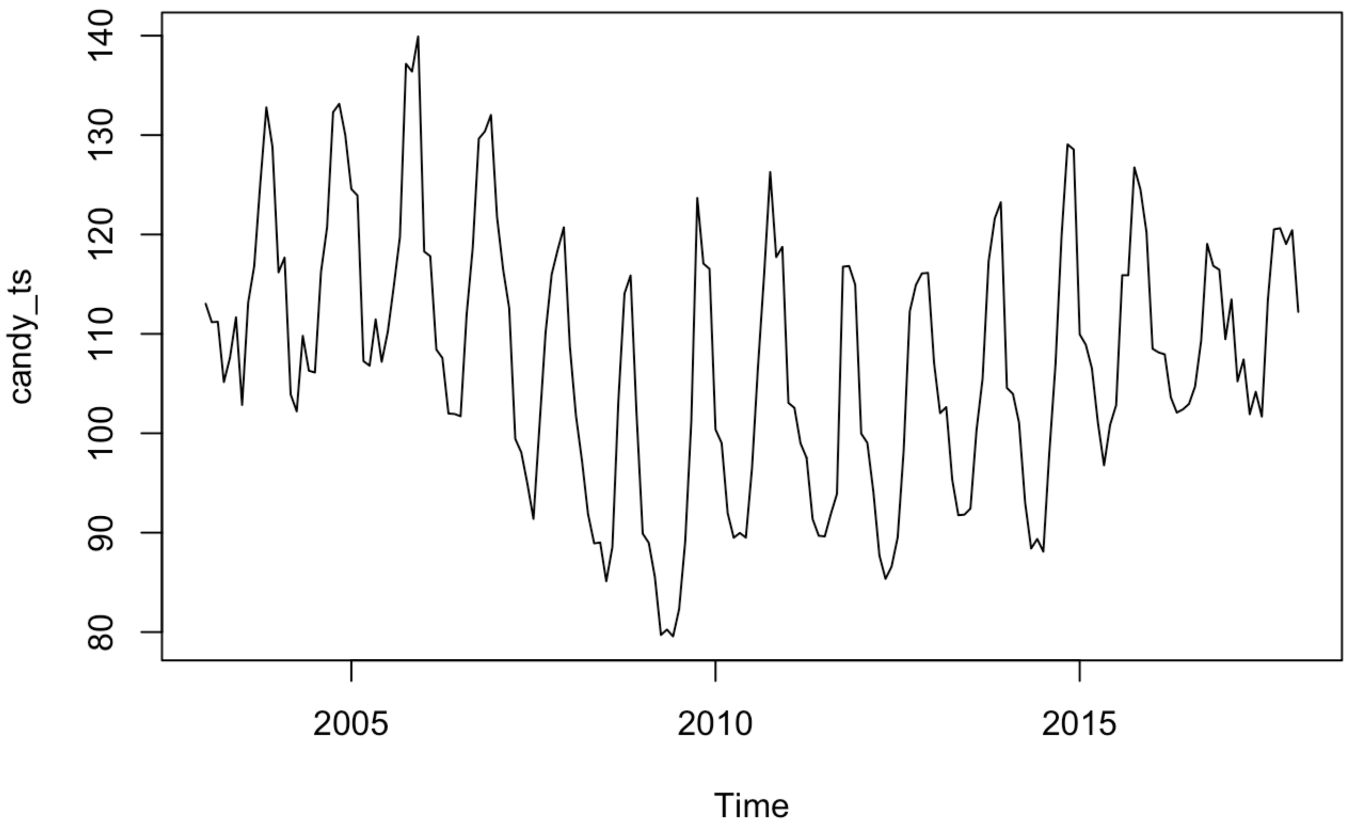
c. Other Observation

Answer: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

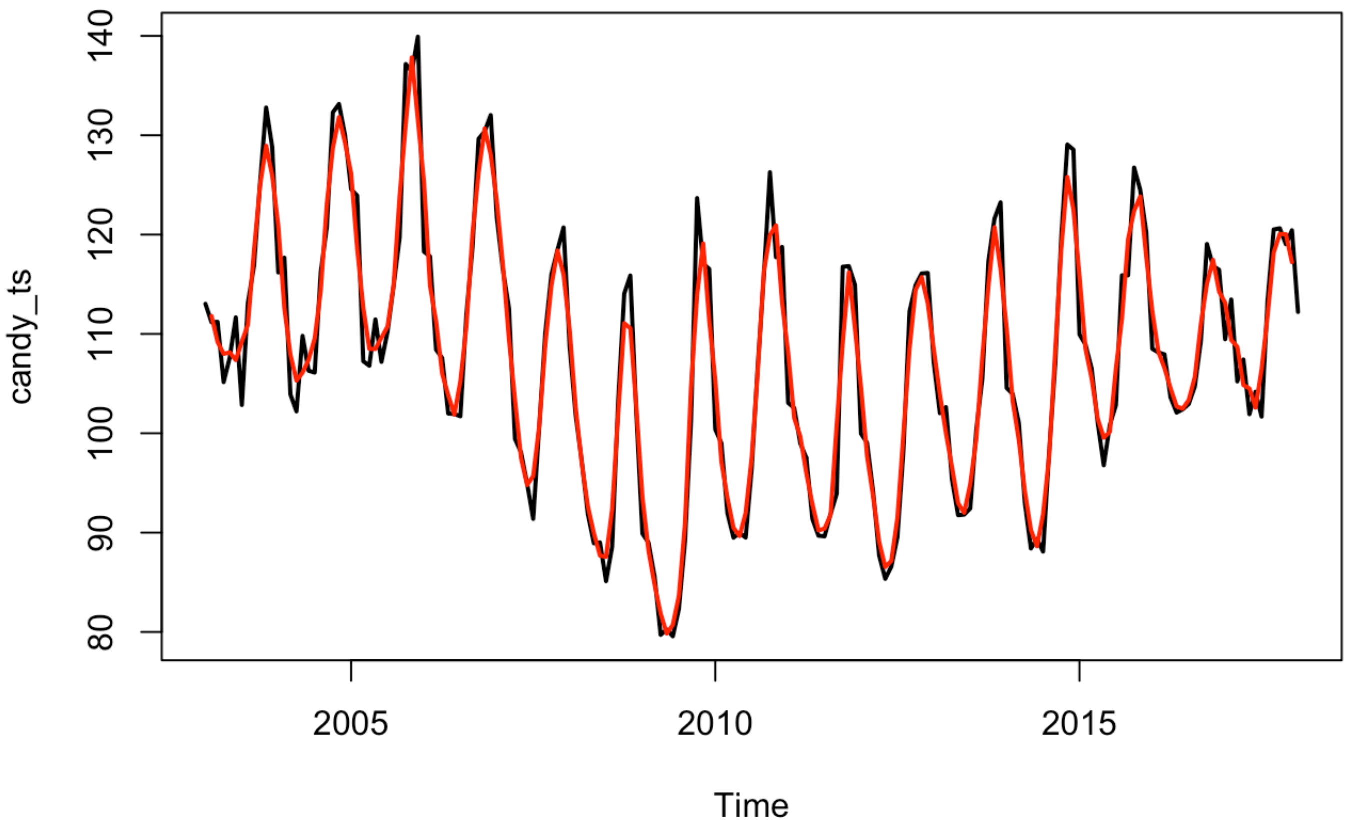
1. Plot the graph for time series.

```
plot(candy_ts)
```



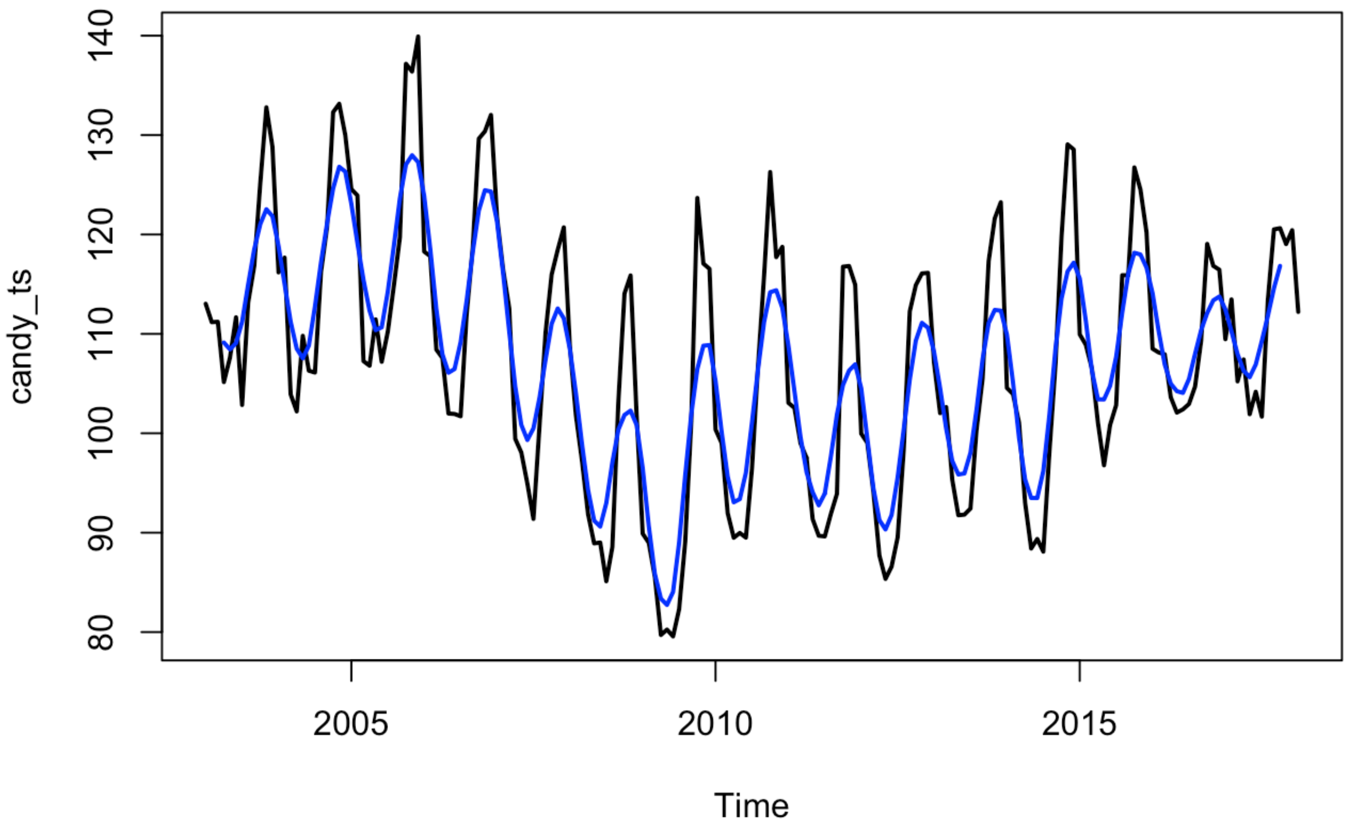
2. Show the Simple Moving average of order 3 on the plot above in Red

```
MA3_forecast <- ma(candy_ts,order=3)
plot(candy_ts, lwd=2)
lines(MA3_forecast,col="Red", lwd=2)
```



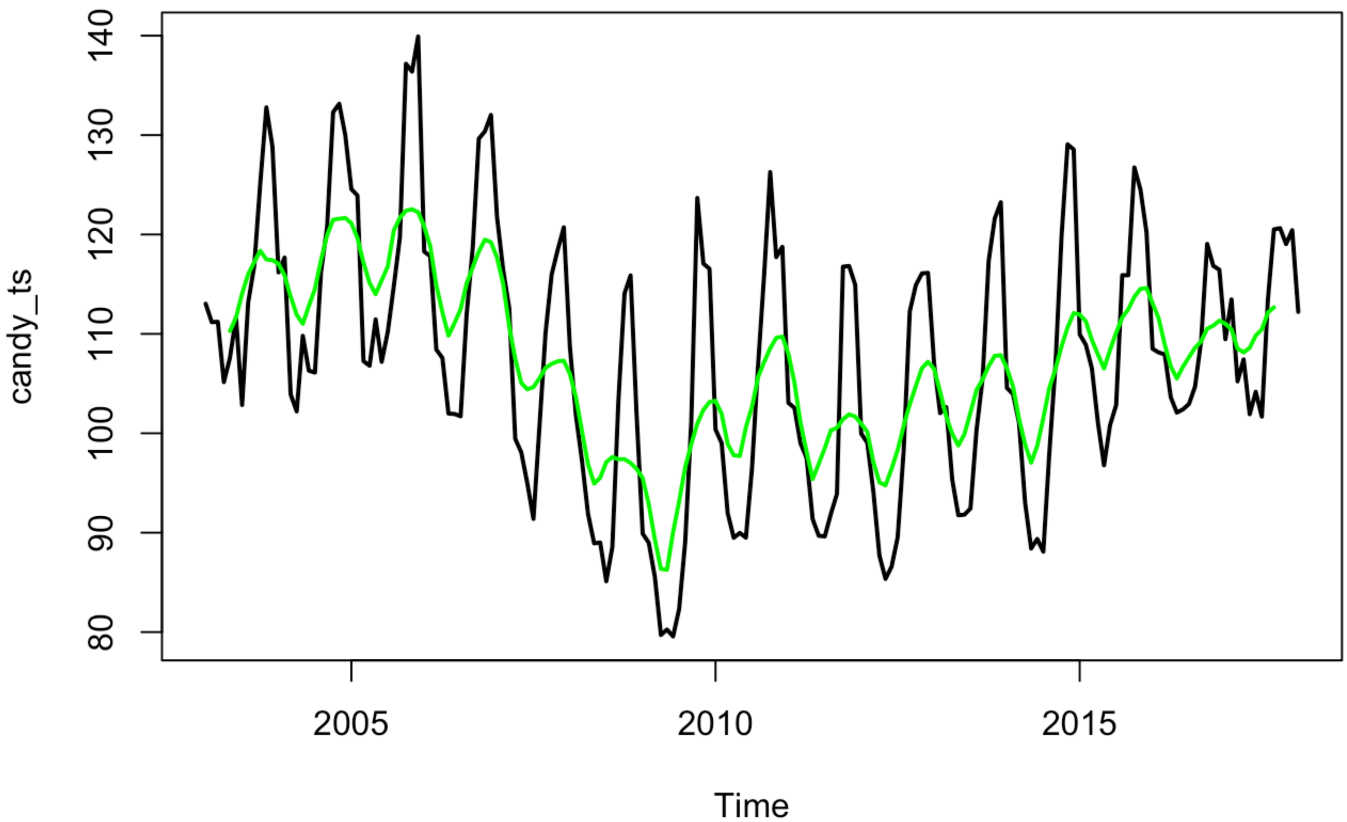
3. Show the Simple Moving average of order 6 on the plot above in Blue

```
MA6_forecast <- ma(candy_ts,order=6)
plot(candy_ts, lwd=2)
lines(MA6_forecast,col="Blue", lwd=2)
```



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

```
MA9_forecast <- ma(candy_ts,order=9)
plot(candy_ts, lwd=2)
lines(MA9_forecast,col="Green", lwd=2)
```



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

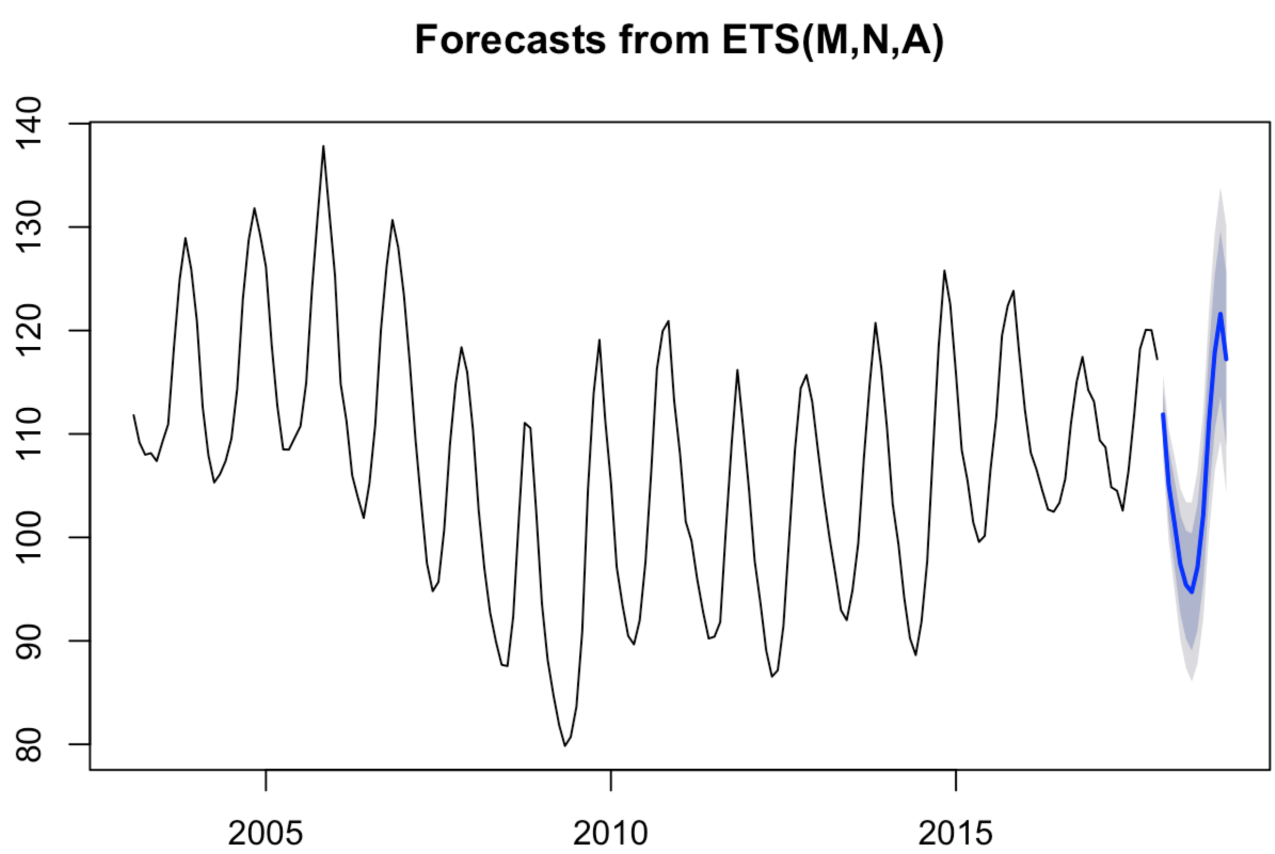
```
forecast_next12 <- forecast(MA3_forecast, h=12)
```

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

```
forecast_next12
```

```
##          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(forecast_next12)
```



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

As the average order goes up, the model does not fit properly as it is not following the data closely. For the good prediction the value of the order should be minimum.

Simple Smoothing

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

```
library(forecast)
simple_smooth = ets(candy_ts)
simple_smooth
```

```
## 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
```

a. What is the value of alpha? What does that value signify?

The value of alpha is 0.7504. Alpha gives us the level of the time series data. The value is quite high that exclaims that the level is based on most recent time series data.

b. What is the value of initial state?

```
simple_smooth$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
```

c. What is the value of sigma? What does the sigma signify?

The sigma value is 0.0361 tells us the standard deviation of residuals.

2. Perform Residual Analysis for this technique.

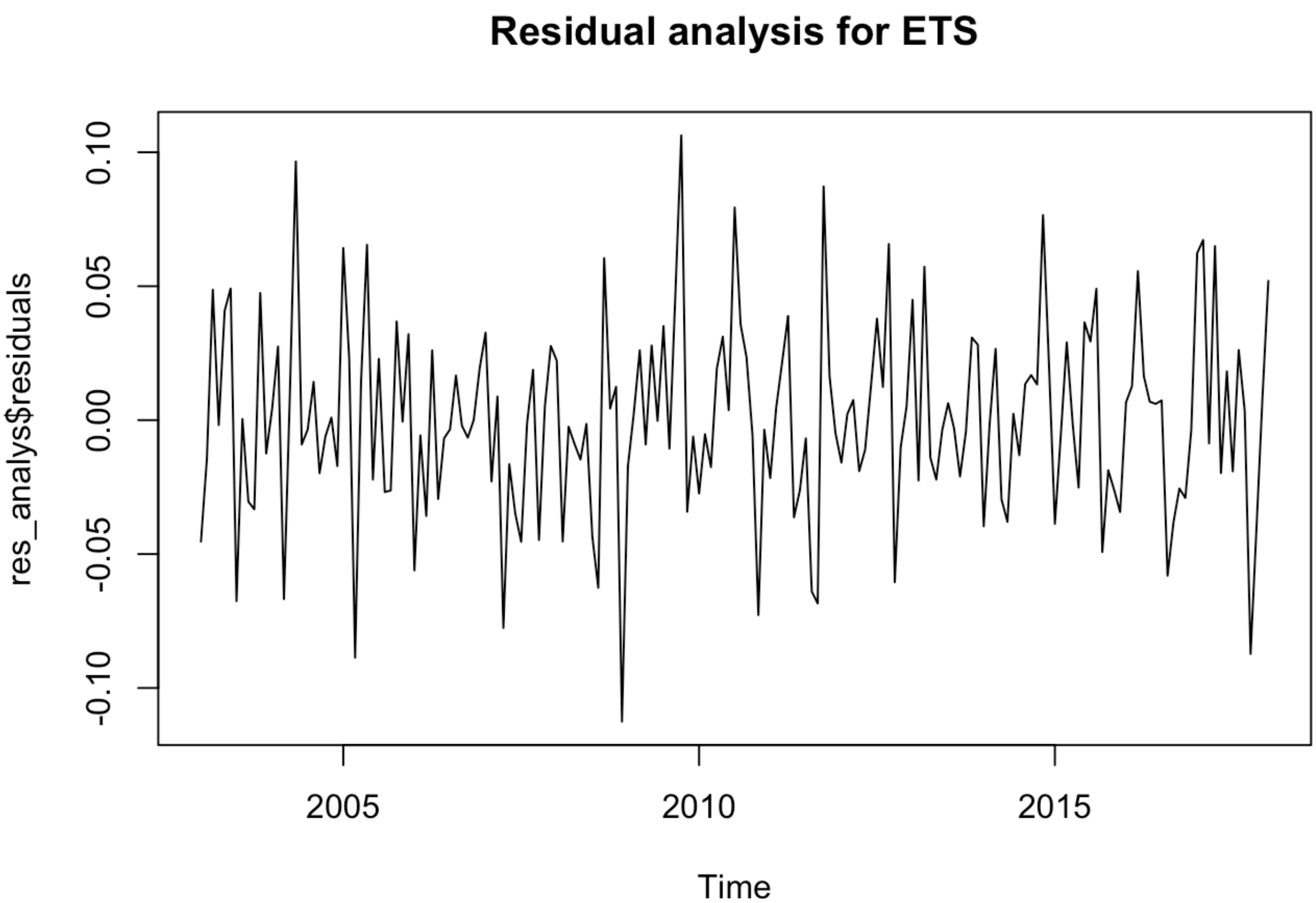
```
res_analys <- ets(candy_ts)
res_analys$residuals
```

```
##               Jan           Feb           Mar           Apr           May
## 2003 -0.0453348995 -0.0145829815  0.0487113230 -0.0018575926  0.0408699252
## 2004  0.0041519836  0.0274765466 -0.0667942836  0.0100424838  0.0965800223
## 2005  0.0642561506  0.0229486754 -0.0886689698  0.0144063765  0.0654680785
## 2006 -0.0561087051 -0.0056894688 -0.0357961221  0.0260425651 -0.0294224109
## 2007  0.0326784102 -0.0229039303  0.0088035964 -0.0775640103 -0.0164536012
## 2008  0.0222225403 -0.0452896444 -0.0024727558 -0.0089096699 -0.0147245439
## 2009 -0.0171755045  0.0026807145  0.0260938564 -0.0090788033  0.0278513491
## 2010 -0.0273884497 -0.0052554186 -0.0175325142  0.0192869291  0.0311482926
## 2011 -0.0216316135  0.0045931407  0.0215204258  0.0387825620 -0.0362911155
## 2012 -0.0158634035  0.0022197870  0.0075006719 -0.0190385081 -0.0110416640
## 2013  0.0449128757 -0.0225365415  0.0572535253 -0.0139369916 -0.0221936868
## 2014 -0.0396361788 -0.0014043310  0.0266322967 -0.0295748656 -0.0379636953
## 2015 -0.0387079542 -0.0055167894  0.0290397398 -0.0013288999 -0.0251861208
## 2016  0.0066730175  0.0126616250  0.0556073796  0.0164211270  0.0068827903
## 2017  0.0623695784  0.0672457178 -0.0086649009  0.0649552133 -0.0197718140
## 2018  0.0520085174
##               Jun           Jul           Aug           Sep           Oct
## 2003  0.0491119261 -0.0675943793  0.0004138258 -0.0304448362 -0.0332781307
## 2004 -0.0090788474 -0.0035294786  0.0142904559 -0.0198048697 -0.0060087426
## 2005 -0.0221728842  0.0228674146 -0.0268475625 -0.0262676254  0.0368239807
## 2006 -0.0067872757 -0.0034689915  0.0166383779 -0.0020452828 -0.0065421604
## 2007 -0.0348328033 -0.0454280228 -0.0011095811  0.0187794278 -0.0447679311
## 2008 -0.0014224829 -0.0434936247 -0.0625751137  0.0604945882  0.0043199844
## 2009 -0.0002739976  0.0351295913 -0.0106092103  0.0467556279  0.1062969766
## 2010  0.0037438827  0.0793723748  0.0357697369  0.0234910940 -0.0053542969
## 2011 -0.0261618099 -0.0068063349 -0.0639551667 -0.0684237002  0.0872421734
## 2012  0.0130440524  0.0378955062  0.0123940527  0.0657536268 -0.0604977105
## 2013 -0.0037306395  0.0063307010 -0.0031064578 -0.0210394568 -0.0042716102
## 2014  0.0023524439 -0.0130967670  0.0134954819  0.0168101295  0.0133117458
## 2015  0.0365005994  0.0293438981  0.0490408003 -0.0492300845 -0.0187128247
## 2016  0.0060679943  0.0073480900 -0.0580629471 -0.0381433305 -0.0255225828
## 2017  0.0182022745 -0.0191110870  0.0261575611  0.0036422022 -0.0872988107
## 2018
##               Nov           Dec
## 2003  0.0474692809 -0.0124494565
## 2004  0.0009689372 -0.0171617578
## 2005 -0.0005549428  0.0320942120
## 2006  0.0000481820  0.0193074536
## 2007  0.0051781883  0.0276814936
## 2008  0.0124417307 -0.1125622171
## 2009 -0.0342153533 -0.0061273983
## 2010 -0.0727679440 -0.0035984058
## 2011  0.0164535721 -0.0047932482
## 2012 -0.0101151917  0.0052711836
## 2013  0.0308232348  0.0281555526
```

```
## 2014  0.0765554761  0.0208509292
## 2015 -0.0261298038 -0.0342912893
## 2016 -0.0290553797 -0.0035734703
## 2017 -0.0402188318  0.0083055800
## 2018
```

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

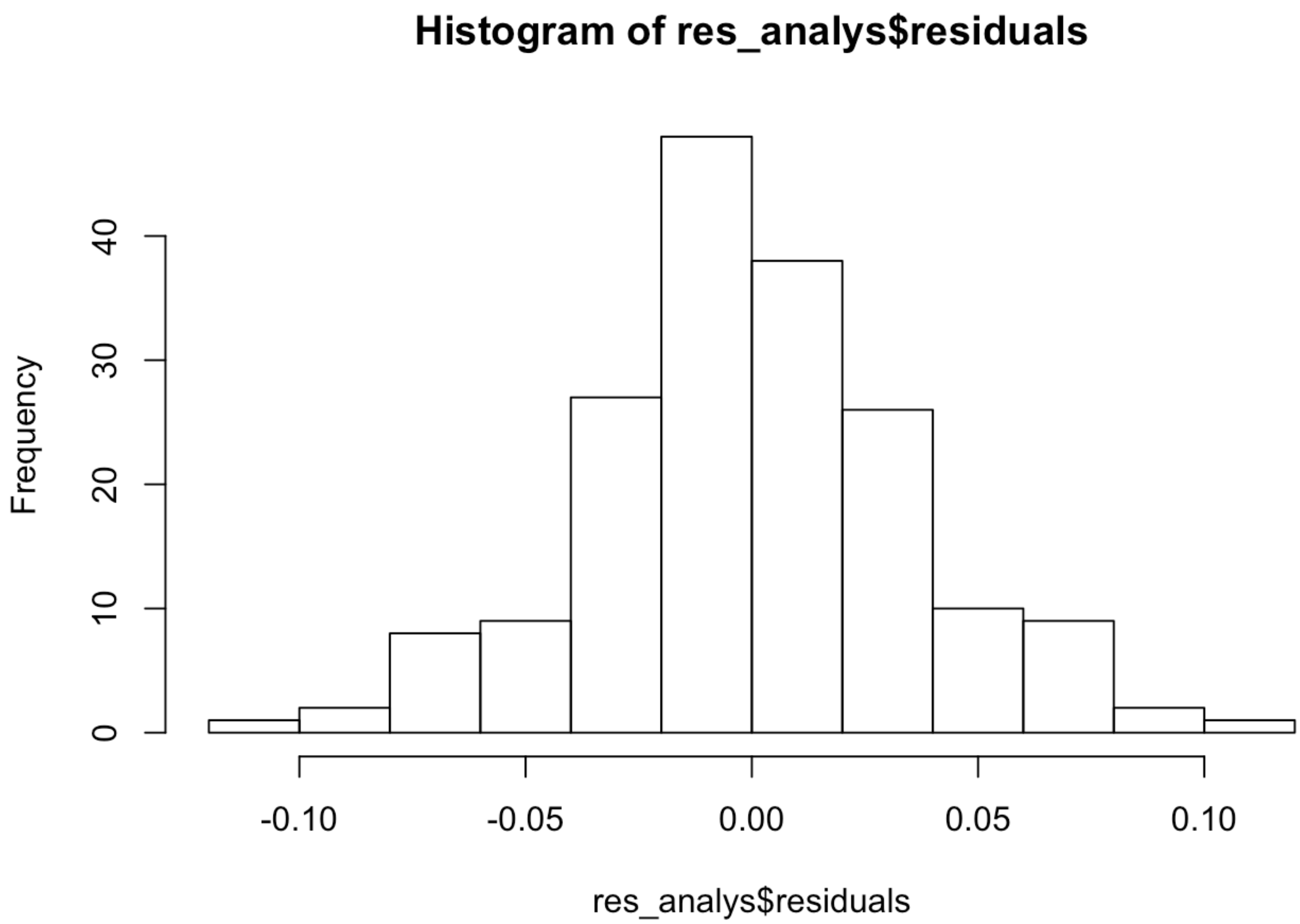
```
plot(res_analys$residuals, main="Residual analysis for ETS")
```



Answer: The residual plot indicates that variation in the residuals is not much different from that of the previous years. There are occasional upward and downward spikes which could be the outliers.

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

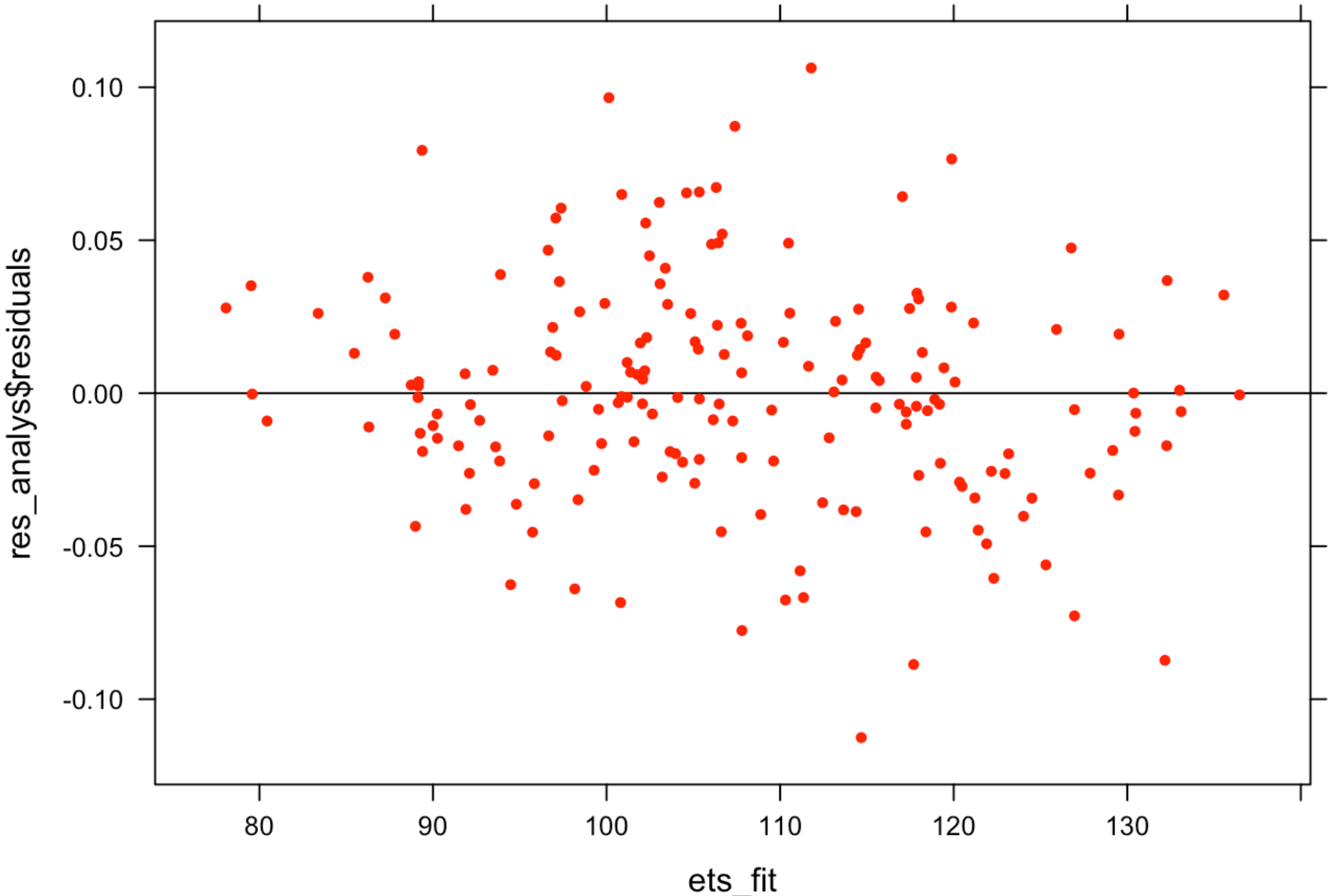
```
hist(res_analys$residuals)
```



Answer: The histogram forms a bell curve that suggest that the residuals are normally distributed.

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

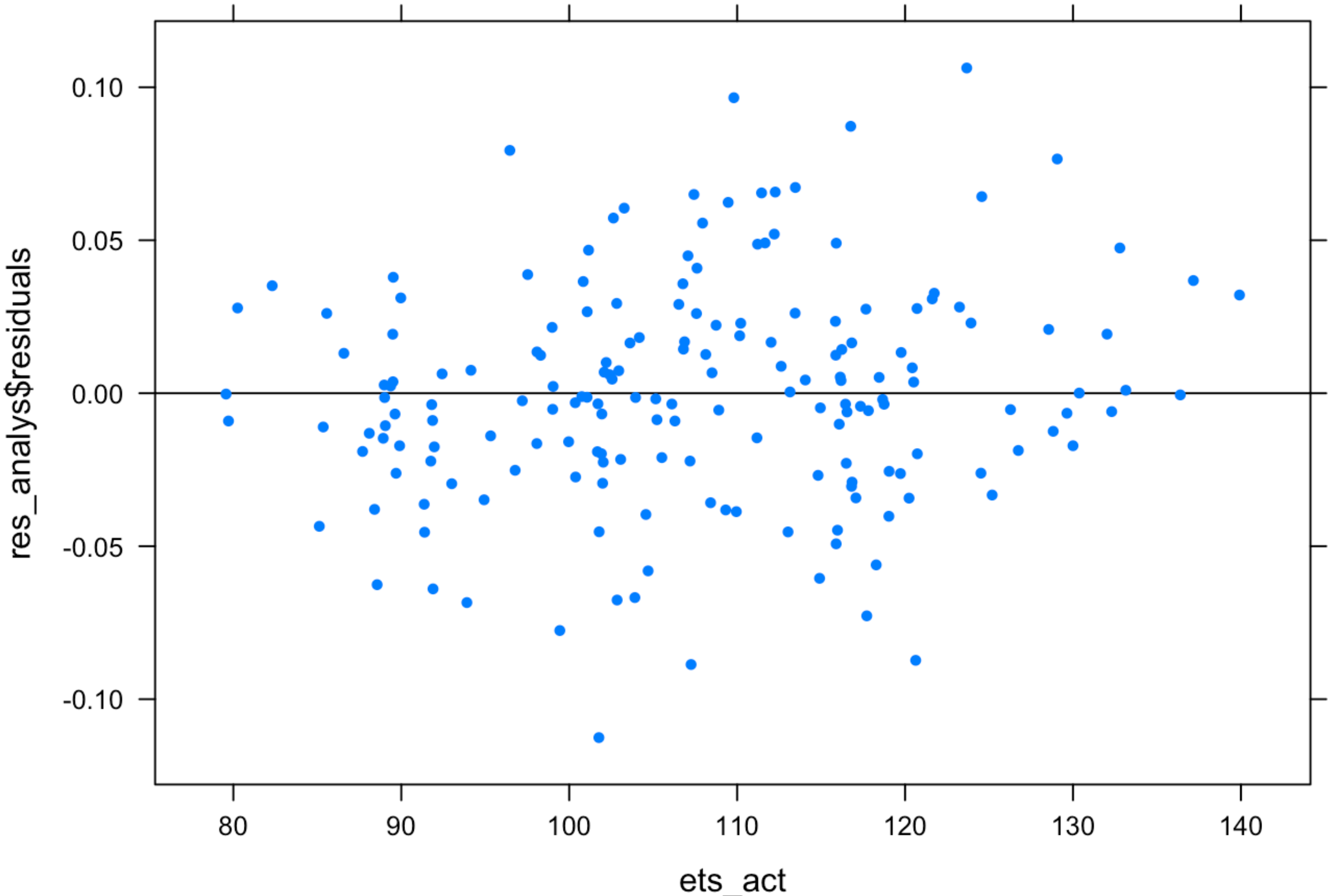
```
library(lattice)
ets_fit = fitted(res_analys)
xyplot(res_analys$residuals~ets_fit, pch=20, col="red", abline=0)
```



Answer: The plot indicates no pattern between residuals and fitted values. Thus, there is no heteroscedasticity in the residuals which means the data has equal variations.

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

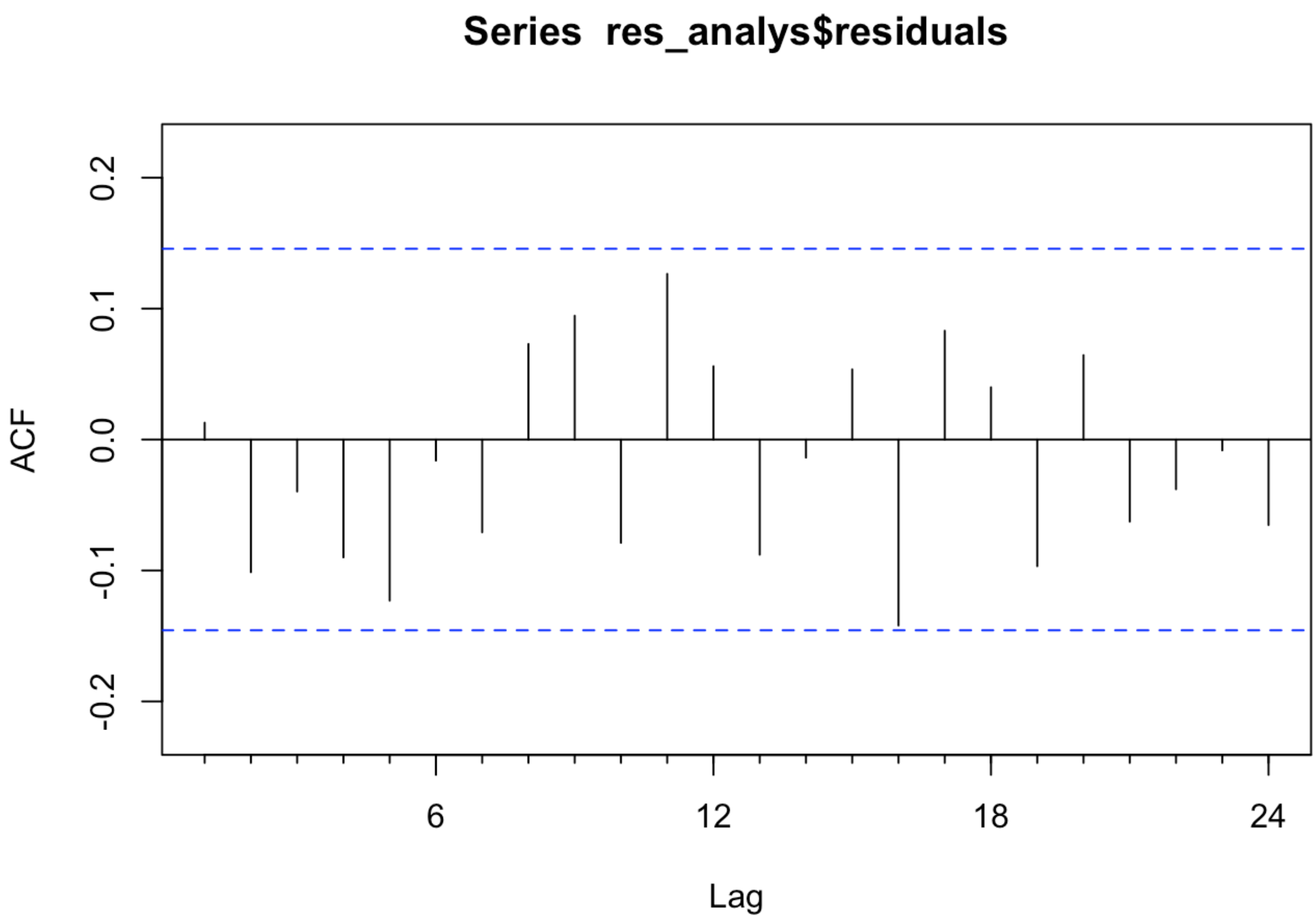
```
ets_act = res_analys$x
xyplot(res_analys$residuals~ets_act, pch=20, abline=0)
```



The plot indicates no pattern between residuals and fitted values. Thus, there is no heteroscedasticity in the residuals which means the data has equal variations.

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

```
Acf(res_analys$residuals)
```



Answer: Spikes shows the values of Autocorrelation with each lags. We can observe that amplitude of each spike is in the blue segment and are highly correlated. Hence Autocorrelation is insignificant.

f. Print the 5 measures of accuracy for this forecasting technique

```
accuracy_ss=accuracy(res_analys)
accuracy_ss
```

```
##                         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
```

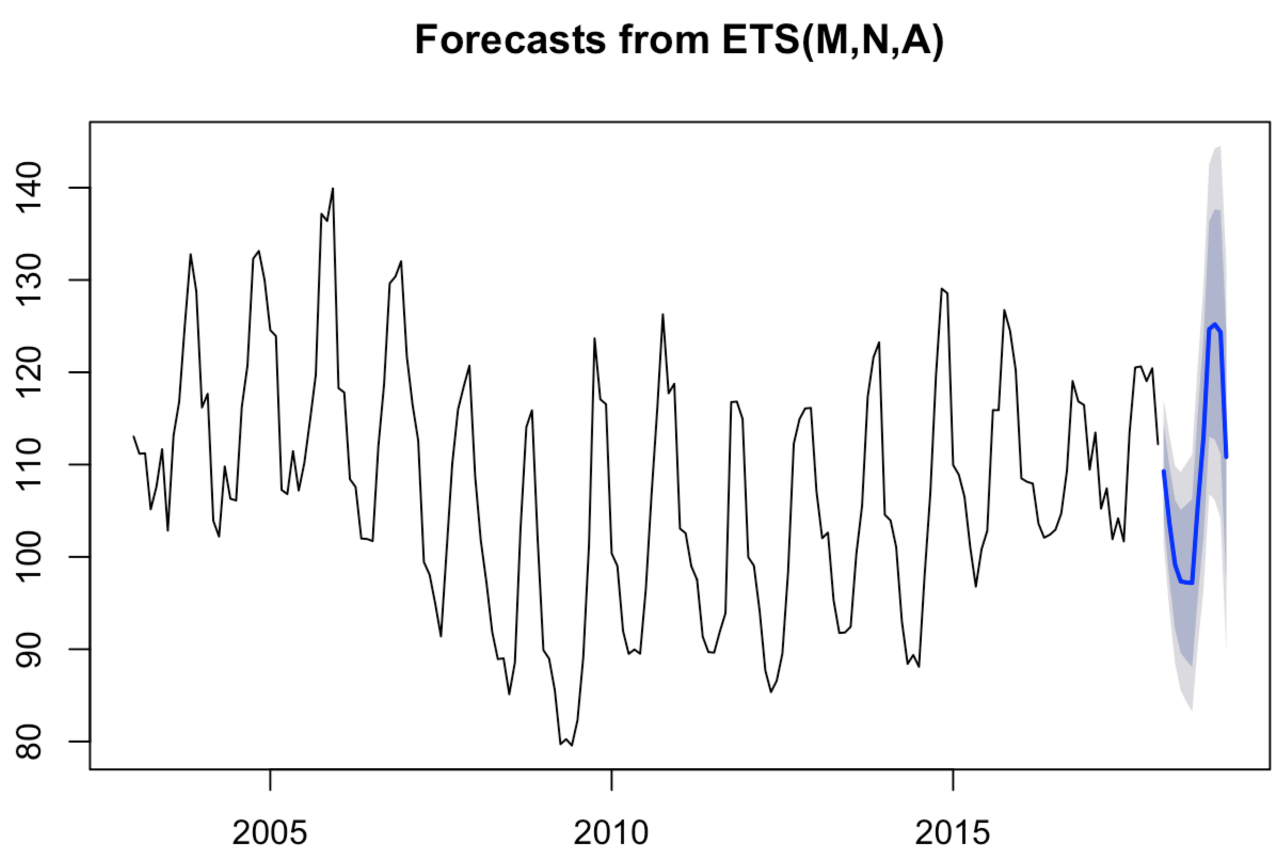
4. Forecast

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

```
forecast_ss <- forecast.ets(res_analys, h=12)
forecast_ss
```

```
##          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_ss)
```



5. Summarize this forecasting technique

```
summary(forecast_ss)
```

```
##
## 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
```

a. How good is the accuracy?

Answer: The MASE and MAPE value are not high which shows that the accuracy is good and also, better as compared to Naive method.

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

Answer: The value of time series in one year will be 110.82761

c. Other observation.

Answer: The RMSE, MAE, MPE, MAPE, MASE values for simple smoothing are low as compared to Naive Method which shows accuracy is better as compared to the Naive Method.

Holt Winters

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

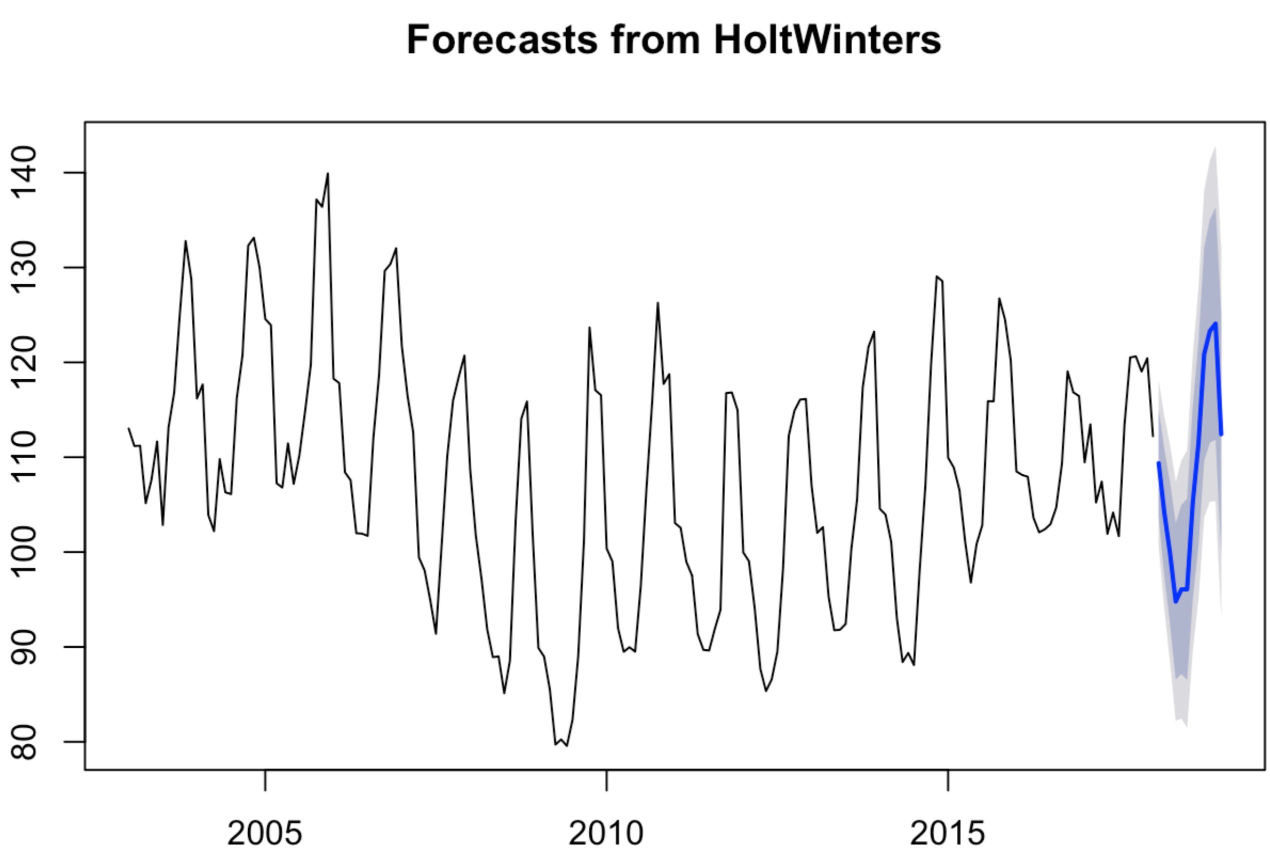
```
library(forecast)
holt = HoltWinters(candy_ts)
holt
```

```
## Holt-Winters exponential smoothing with trend and additive seasonal component.
##
## 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
```

```
forecast_holt <- forecast(holt, h = 12)
forecast_holt
```

```
##          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(forecast_holt)
```



a. What is the value of alpha? What does that value signify?

The alpha value is 0.6058406 which means that future prediction depend upon the recent observations.

b. What is the value of beta? What does that value signify?

The beta value is 0. It explains that trend component of the time series is set equal to its initial state and has not been updated.

c. What is the value of gamma? What does that value signify?

The gamma value is 0.6033215 which means the seasonality in the time series data. Since the value is quite high, therefore recent observations are weighted heavily.

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

```
holt$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
```

The initial states for the level, trend and seasonality can be seen as a,b,s1 values.

e. What is the value of sigma? What does the sigma signify?

```
sd(complete.cases(residuals(forecast_holt)))
```

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

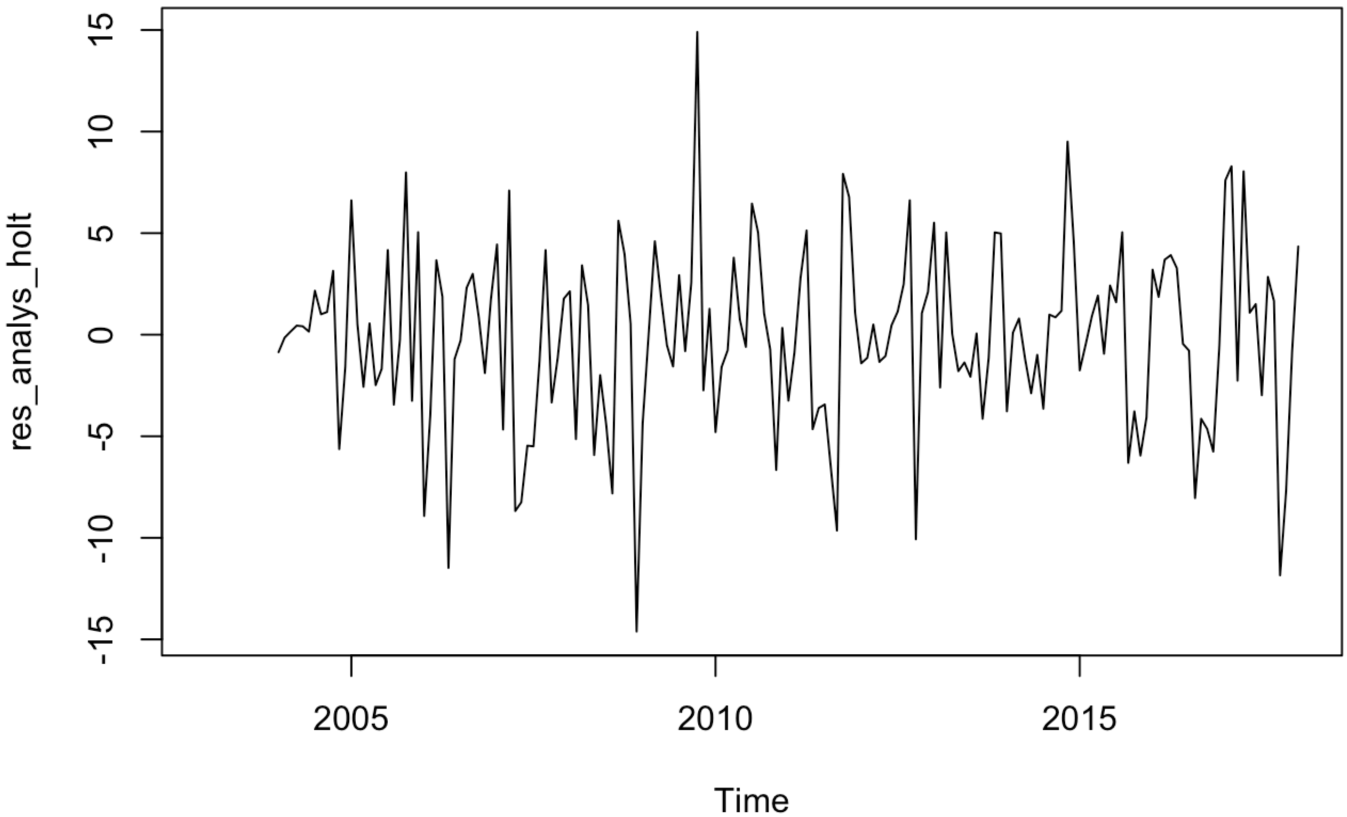
Value of Sigma is 0.249493 which is the standard deviation of residuals.

2. Perform Residual Analysis for this technique.

```
res_analys_holt = residuals(forecast_holt)
```

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

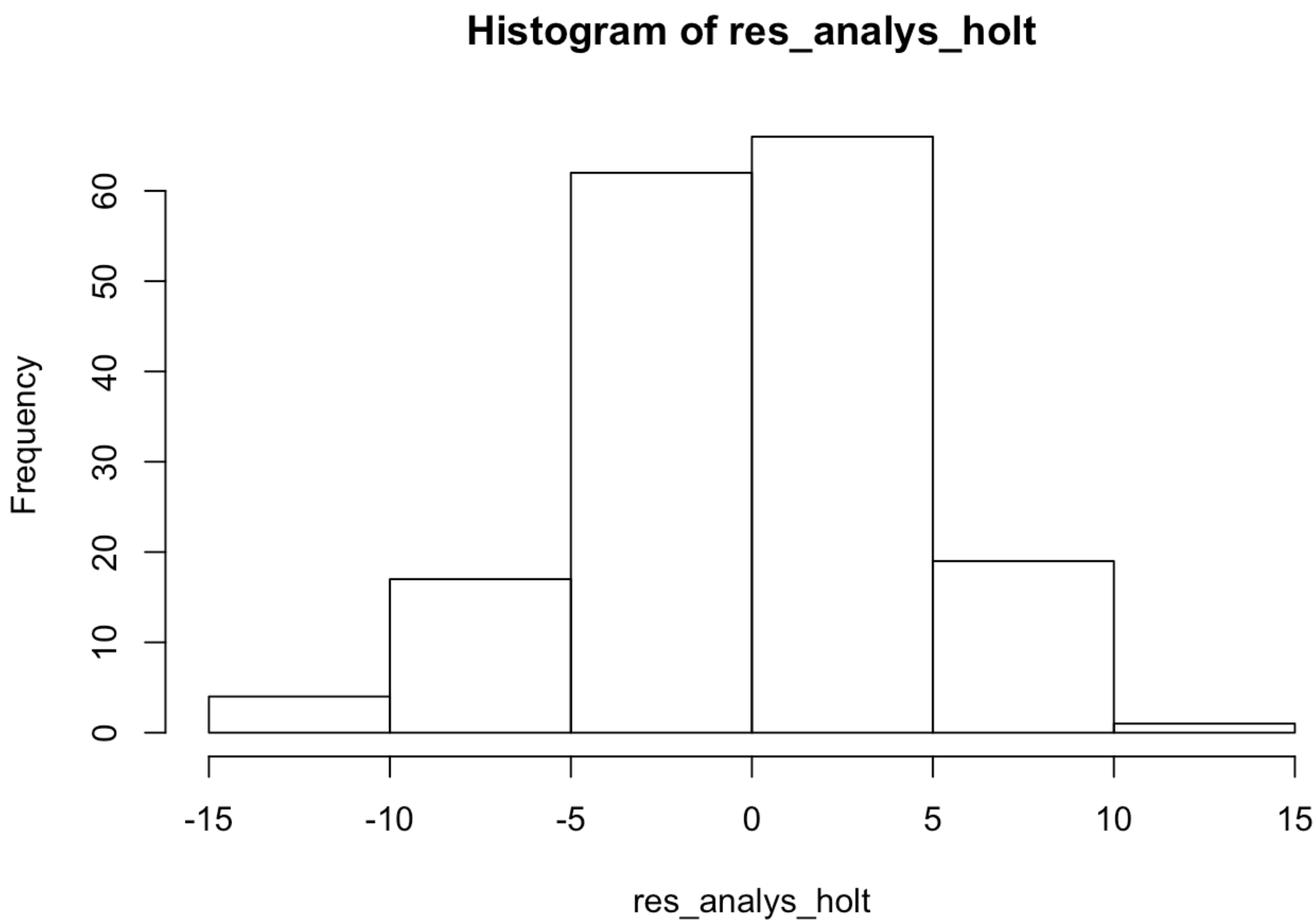
```
plot(res_analys_holt)
```



Answer: The residual plot indicates that variation in the residuals in future is not much different from that of the previous years. It comes out to be constant with occasional spikes.

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

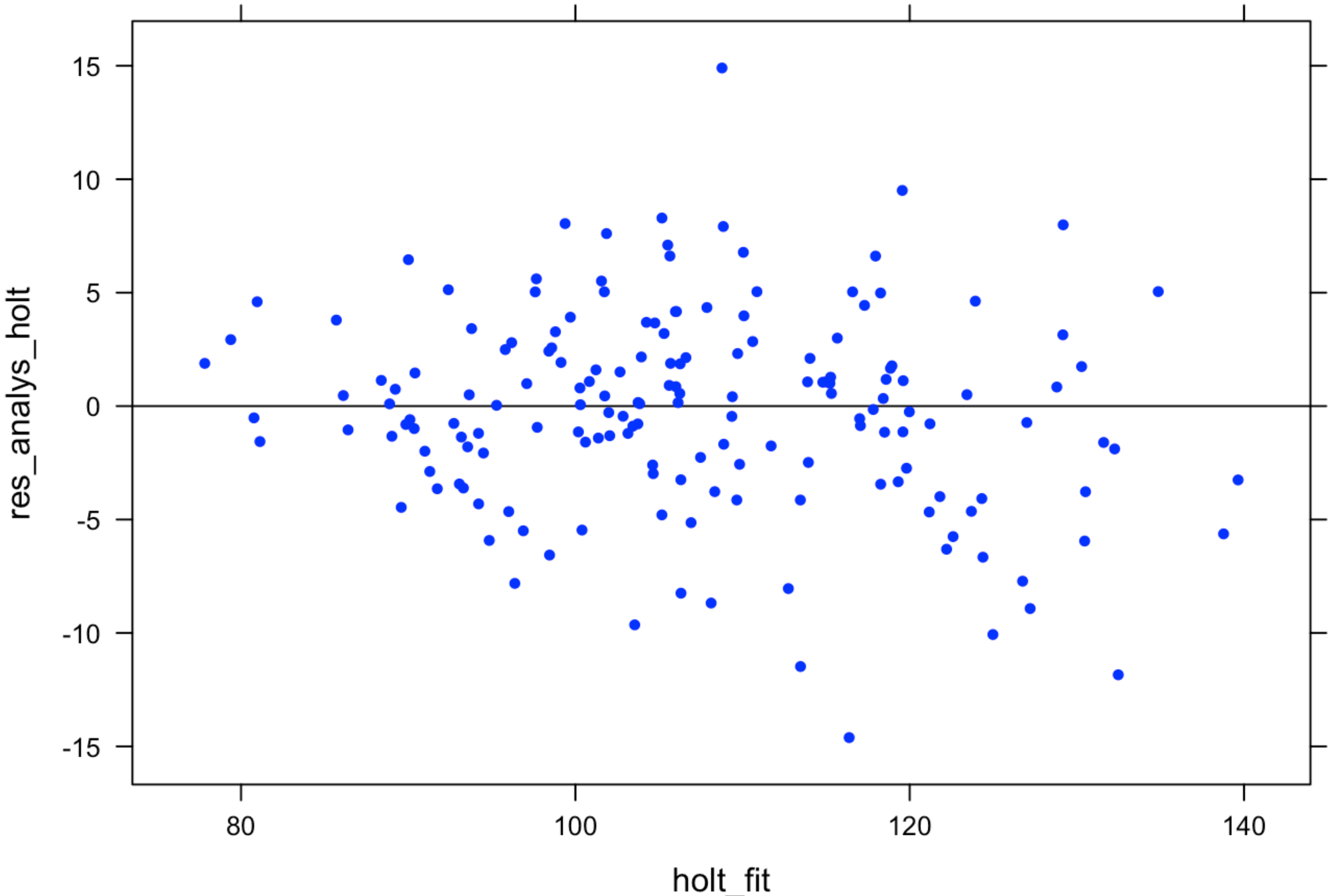
```
hist(res_analys_holt)
```



The histogram plot forms a bell curve that suggest that residuals are normally distributed.

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

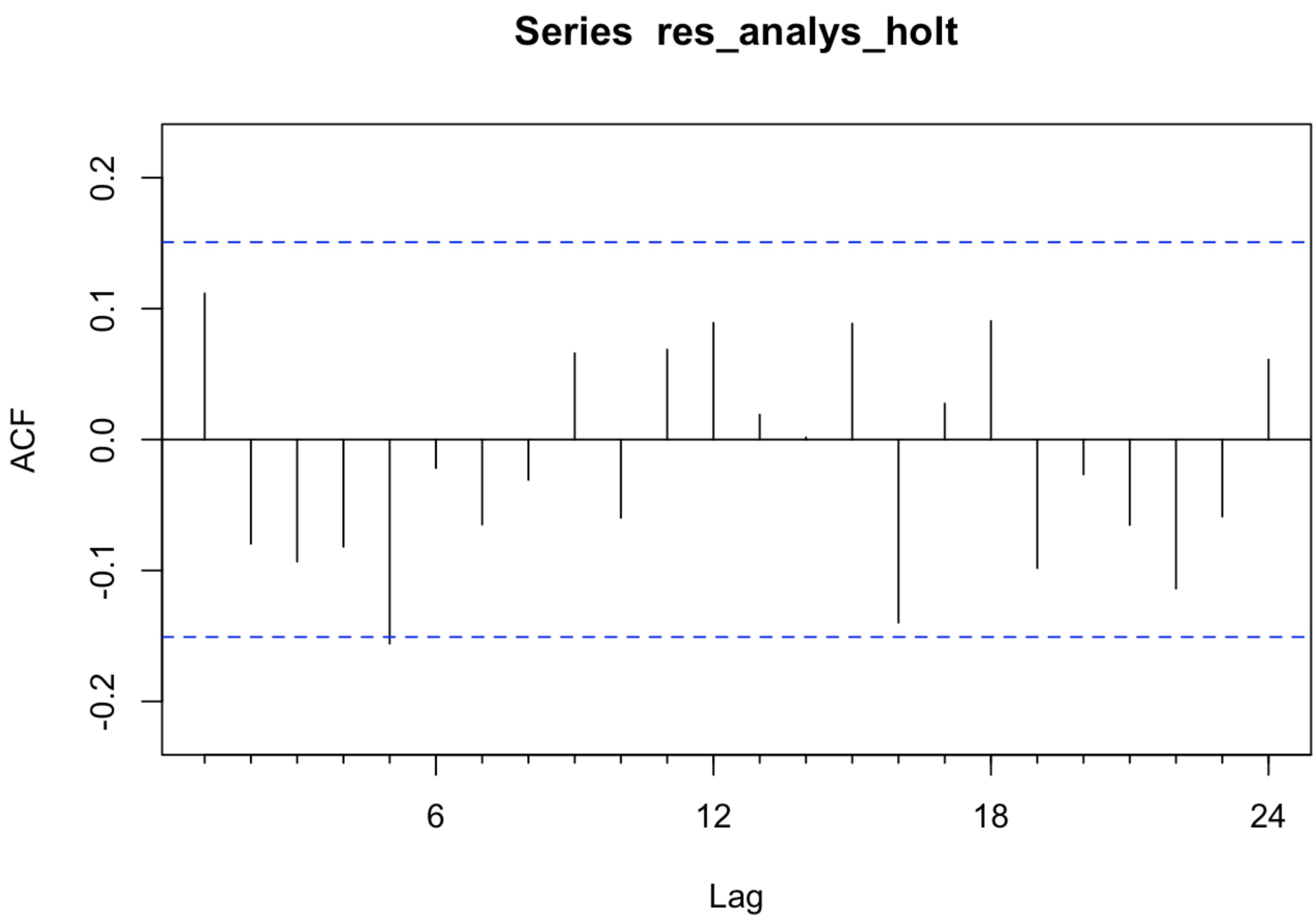
```
holt_fit <- fitted(forecast_holt)
xyplot(res_analys_holt ~ holt_fit, pch=20, col="blue", abline=0)
```



The plot indicates no pattern between residuals and fitted values. Thus, there is no heteroscedasticity in the residuals which means the data has equal variations.

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

```
Acf(res_analys_holt)
```



There is a significant lag at 5 in the downward direction. Other spikes shows the values of Autocorrelation with each lags. We can observe that amplitude of each spike is in the blue segment and are highly correlated. Hence Autocorrelation is insignificant.

3. Print the 5 measures of accuracy for this forecasting technique

```
accuracy_holt <- accuracy(forecast_holt)
accuracy_holt
```

```
##                       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
```

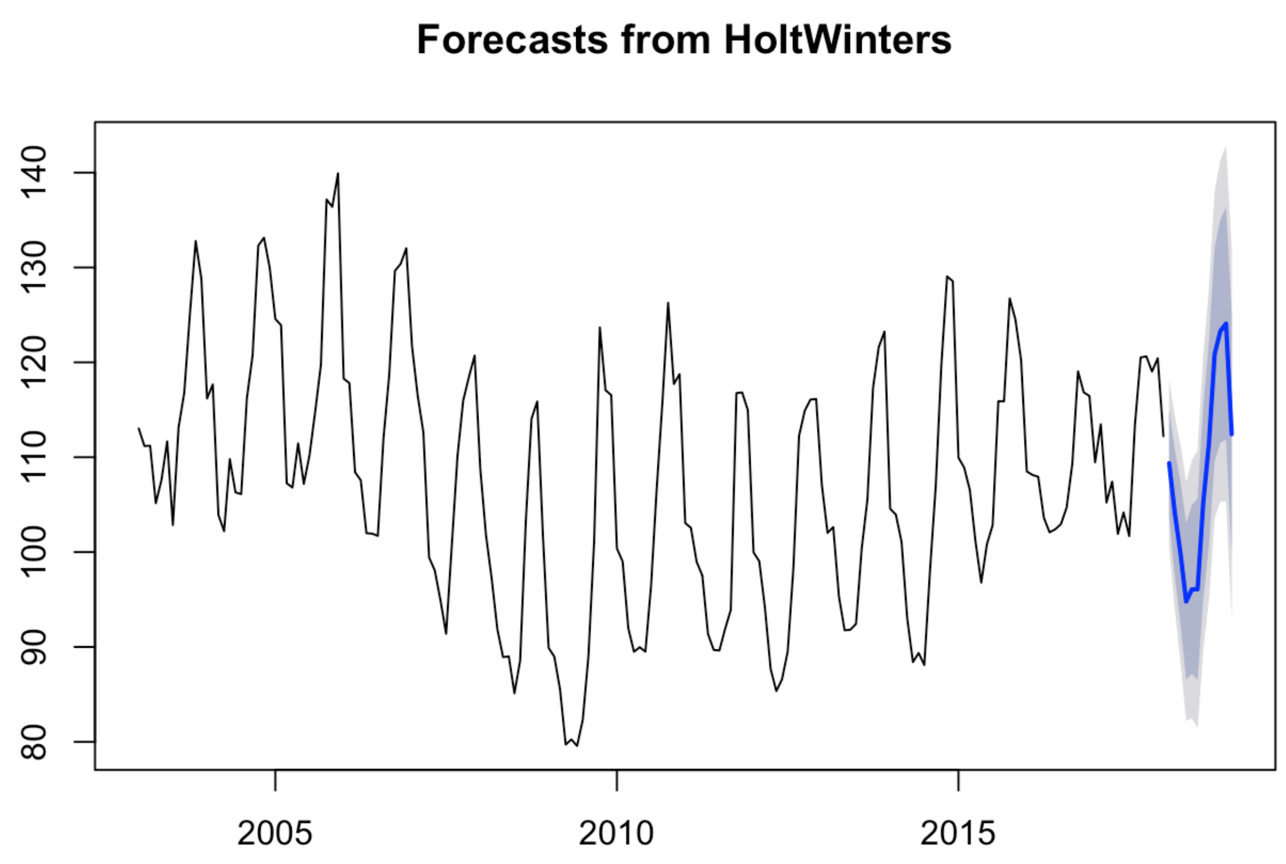
4. Forecast

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

```
forecast_holt
```

```
##          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(forecast_holt)
```



5. Summarize this forecasting technique

```
summary(forecast_holt)
```

```
##
```

```
## Forecast method: HoltWinters
##
## Model Information:
## Holt-Winters exponential smoothing with trend and additive seasonal component.
##
## 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
```

a. How good is the accuracy?

The MAPE and MASE values are low which means the accuracy is high.

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

The predicted value of time series in one year will be 112.42775

Accuracy Summary

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

```
accuracy_summary = rbind(accuracy_nm, accuracy_ss, accuracy_holt)
rownames(accuracy_summary) <- c("Naive Method", "ETS", "Holt Winter")
accuracy_summary
```

```
##                        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
```

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

Naive Method

1. It is the simplest forecasting method.
2. Naive forecasts are often used as a benchmark when assessing the accuracy of a set of forecasts.
3. Not a good option for complex data.

Simple Moving Average

1. A very basic time series smoothing method.
2. It is used when recent observations influence more than the previous observations.
3. It does not handle trend or seasonality well

Simple Smoothing

1. This method is suitable for forecasting data with no trend or seasonal pattern. The main aim is to estimate the current level.

2. It is appropriate for data with no predictable upward or downward trend.
3. Useful for forecasting short term trends

Holt Winters

1. It is used when forecast data points in a series, when the series is “seasonal”, i.e. repetitive over some period.
2. Also Known as Triple Exponential Smoothing.
3. Works on additive method and multiplicative method.

Summary of Accuracy Parameters:

ME: Mean Error : -0.004547778

lowest = Naive Method and highest = Holt winter

RMSE: Root Mean Squared Error: 3.961930

lowest = ETS and highest = Naive

MAE: Mean Absolute Error: 2.971197

lowest = ETS and highest = Naive

MPE: Mean Percentage Error: 0.1133162

MAPE: Mean Absolute Percentage Error: 2.749518

lowest = ETS, highest = Naive

MASE: Mean Absolute Scaled Error: 0.4899657

lowest = ETS, highest = Naive

ACF1: Autocorrelation of errors at lag 1: 0.0011844

lowest = ETS, highest = Naive

Therefore, as per the above accuracy measures, smoothing method ETS looks the best forecast method and Naive looks like the worst forecsating model.

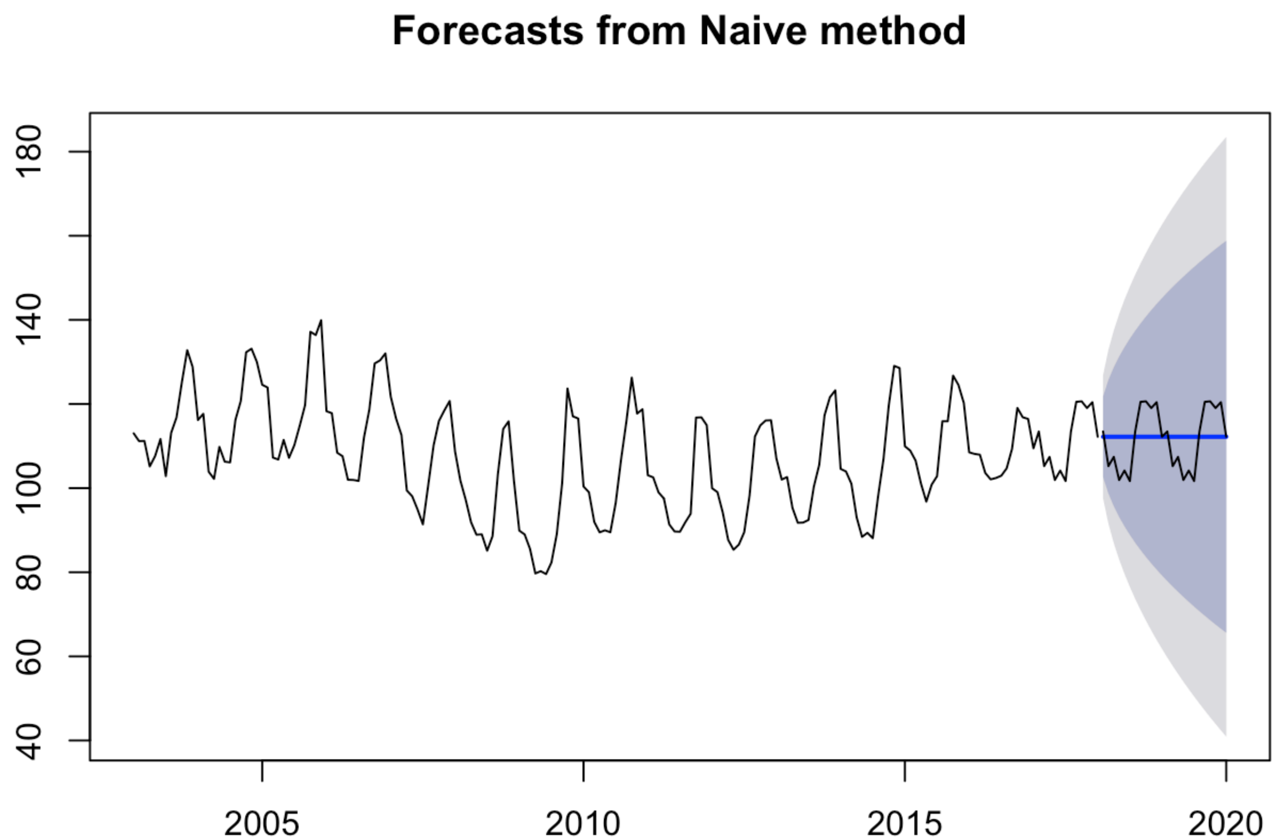
Conclusion

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

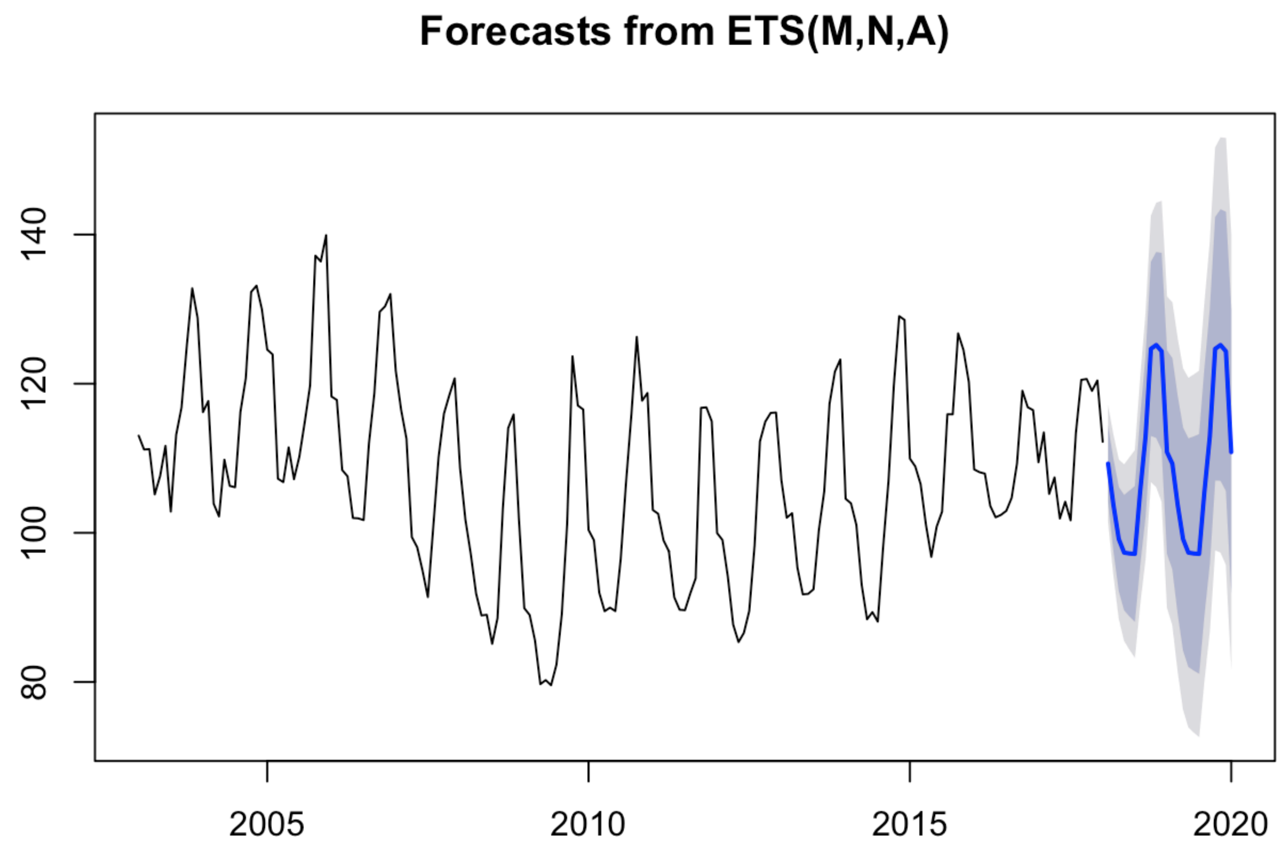
Answer: The ETS proves to be the best forecasting model for the time series data of candies over a period of time and Naive proves to be the worst with highest MAPE and MASE values.

2. 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?

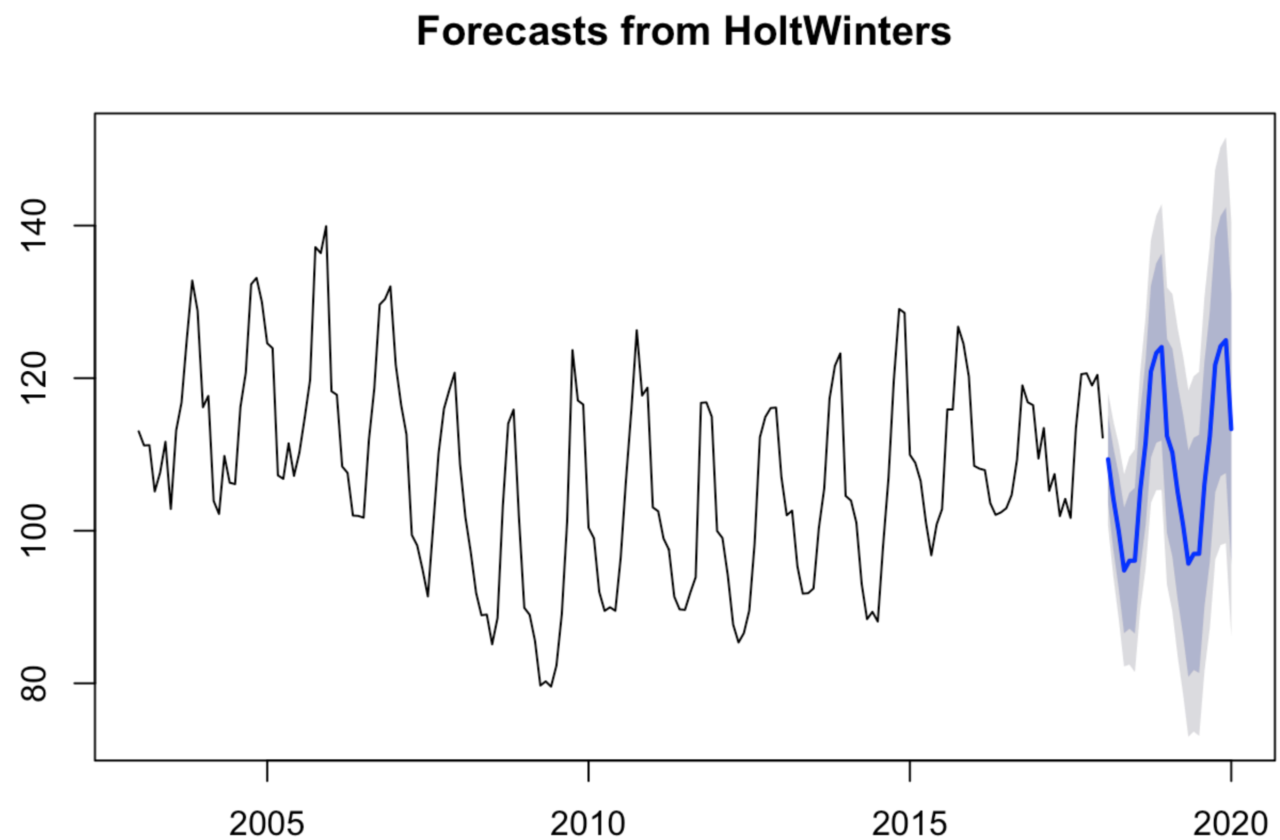
```
#Naive forecast method for 2 years
naive_forecast24 <- naive(candy_ts, 24)
#Seasonal Naive
snaive_forecast24 <- snaive(candy_ts, 24)
#naive_forecast
plot(naive_forecast24)
lines(snaive_forecast24$mean,col="black")
```




```
#Simple Moving Averages Forecast
MA1_forecast <- ma(candy_ts,order=1)
next_forecast24 <- forecast(MA1_forecast, h=24)
plot(next_forecast24)
```




```r
#Holt Winters
holt_forecast24 <- forecast(holt, h = 24)
plot(holt_forecast24)
```




```
next_forecast24
```

```
##          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
```

The moving average model shows the value at the end of 2 years in January 2020 is 110.82761 which is similar to the value in the year 2018 and 2019.

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

Answer: The Rankings are as follows:

1. Best model= Simple Smoothing
2. Better Model= Holt Winter Method
3. Worst Model= Naive Method