TIME SERIES Assignment

The attached data shows monthly demand of two different types of consumable items in a certain store from January 2002 to July 2017. The ultimate objective of this exercise is to predict sales for the period August 2017 to December 2018.

1. Read the data as time series objects in R. Plot the data. What are the major features you notice in the series? How do the two series differ?
2. Any missing value in the series? How do you tackle missing values in the series?
3. Before a formal extraction of time series components is done, can you check for seasonal changes in the data for the two series separately? Particularly whether there are more variability in a season compared to the others, whether seasonal variations are changing across years etc. Compare the behavior of the two series.
4. Decompose each series to extract trend and seasonality, if there are any. Which seasonality is more appropriate – additive or multiplicative? Explain the seasonal indices. In which month(s) do you see higher sales and which month(s) you see lower sales? Any difference in the nature of demand of the two items?
5. Can you extract the residuals for the two decomposition exercises? Comment on the pattern of the residuals?
6. Before the final forecast is undertaken one would like to compare a few models. Use the last 21 months as hold-out sample fit a suitable exponential smoothing model to the rest of the data and calculate MAPE. What are the values of α, β and γ? What role do they play in the modeling? For the same hold-out period compare forecast by decomposition and compute MAPE. Which model gives smaller MAPE? Give a comparison for the two demands.
7. Use the ‘best’ model obtained from above to forecast demand for the period August 2017 to December 2018 for both items. Provide forecasted values as well as their upper and lower confidence limits. If you are the store manager what decisions would you make after looking at the demand of the two items over years?

1. Setting Working directory and reading the data,

> setwd("D:/Time Series Forecasting/Assignment 2")

> library(readxl)

> DemandAB = read\_excel("DemandAB.xlsx")

Taking the summary of the data,

> summary(DemandAB)

Year Month Item A Item B

Min. :2002 Min. : 1.000 Min. :1954 Min. :1153

1st Qu.:2005 1st Qu.: 3.000 1st Qu.:2743 1st Qu.:2358

Median :2009 Median : 6.000 Median :3114 Median :2869

Mean :2009 Mean : 6.406 Mean :3251 Mean :2957

3rd Qu.:2013 3rd Qu.: 9.000 3rd Qu.:3729 3rd Qu.:3460

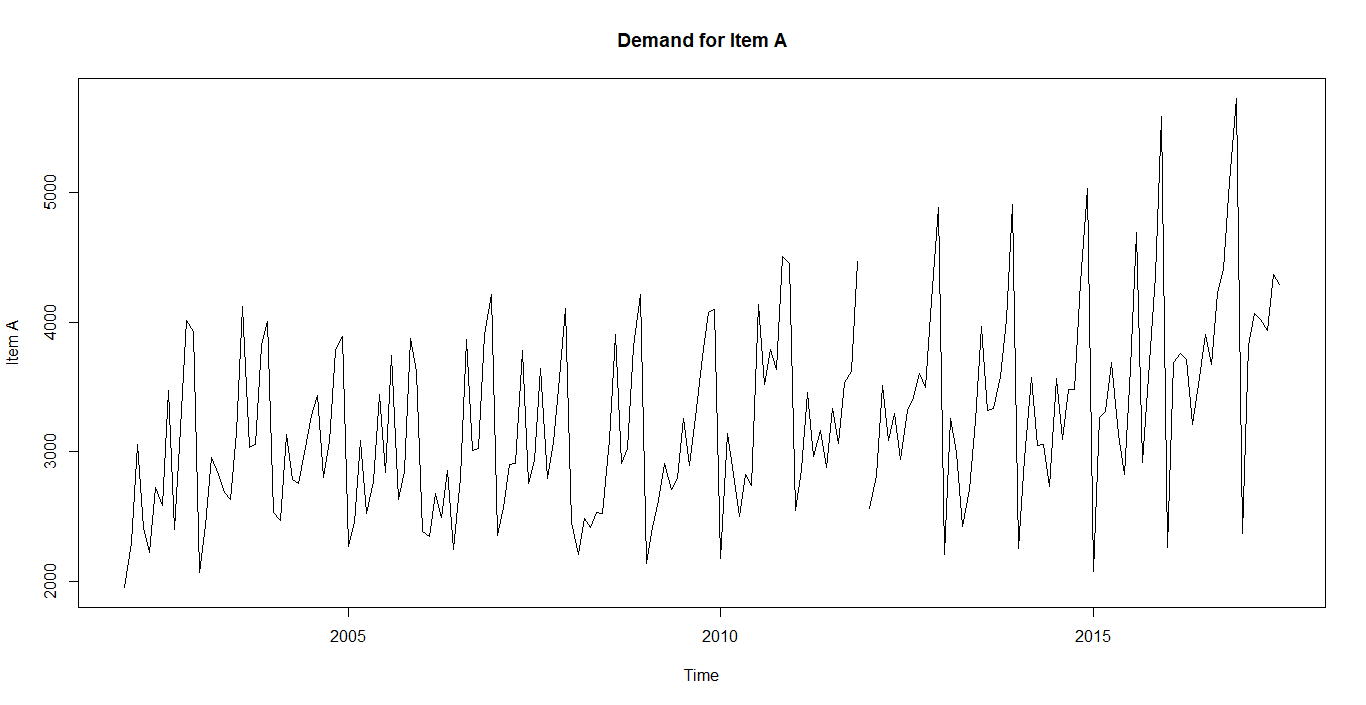
Max. :2017 Max. :12.000 Max. :5725 Max. :5618

NA's :1 NA's :1

Saving data in time series format,

> DemandA <- ts(DemandAB[,3], start=c(2002,1), frequency=12)

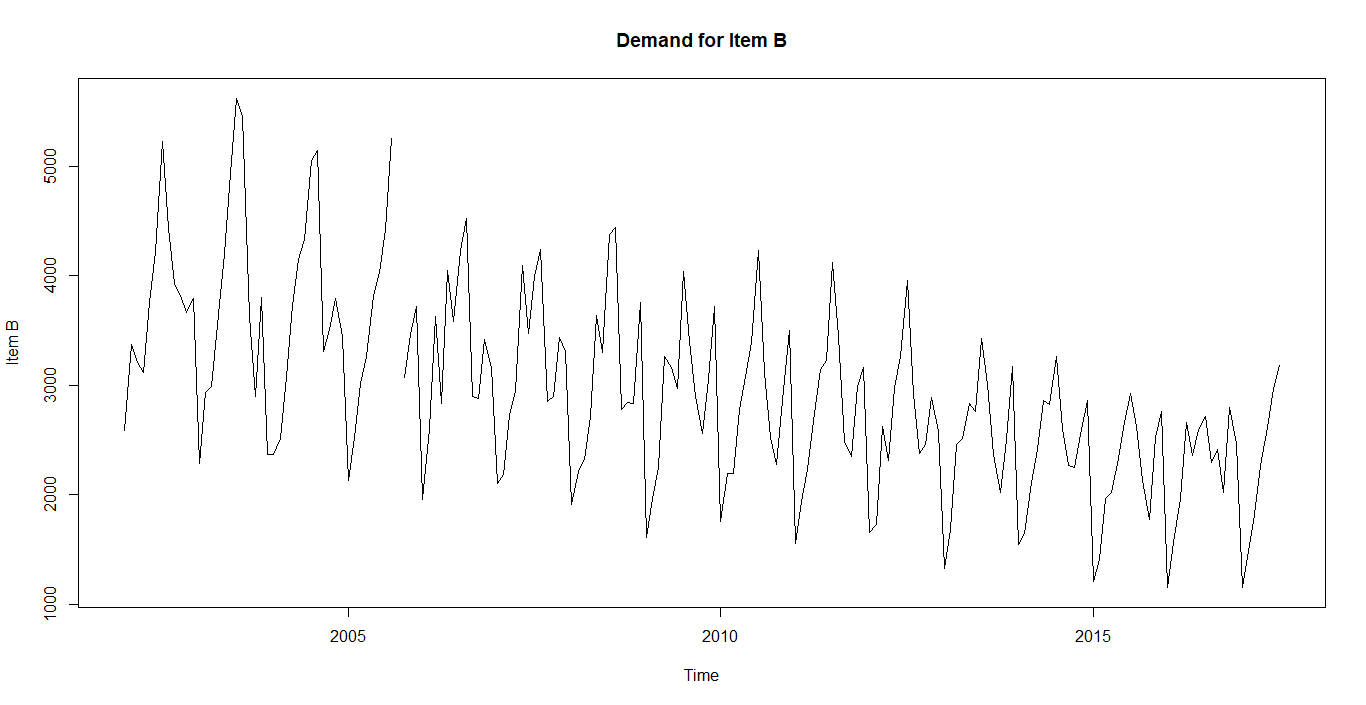
> plot(DemandA, main = "Demand for Item A")



> DemandB <- ts(DemandAB[,4], start=c(2002,1), frequency=12)

> plot(DemandB,main = "Demand for Item B")

Here item A is showing a seasonality but not sure whether there is a trend or not. It is showing properties of addictive series till 2010, but after 2010 it is showing more variance and showing some positive trend. Hope item A can be considered as multiplicative series.



Item B is showing both seasonality and negative trend. Here the Item B follows multiplicative series.

2. There are missing values in both Item A data and Item B data.

Imputing the missing data in Item A with average of all Decembers,

Average of December = 5219,

> DemandAB$`Item A`[is.na(DemandAB$`Item A`)] = 5219

Imputing missing values of Item B with average of adjacent values for item B,

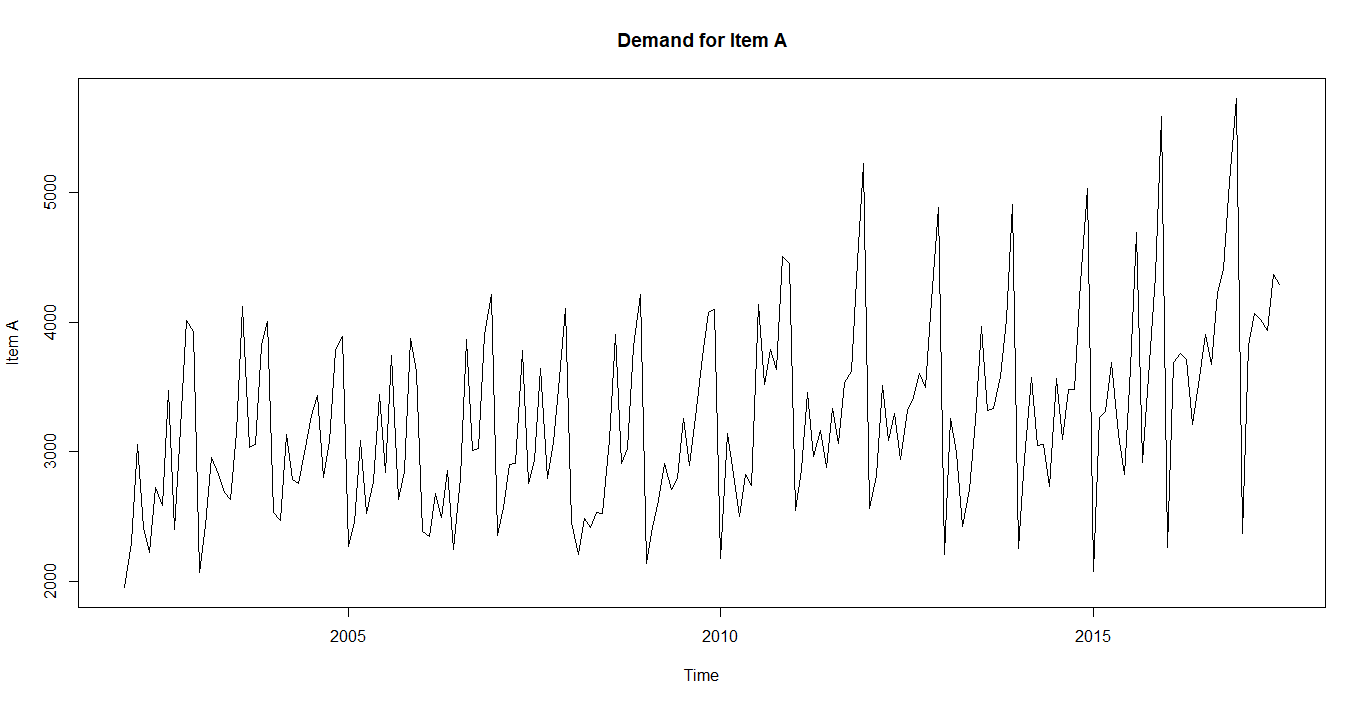
> library(zoo)

> DemandAB$`Item B`=na.approx(DemandAB$`Item B`)

Time series plot after imputing missing values,

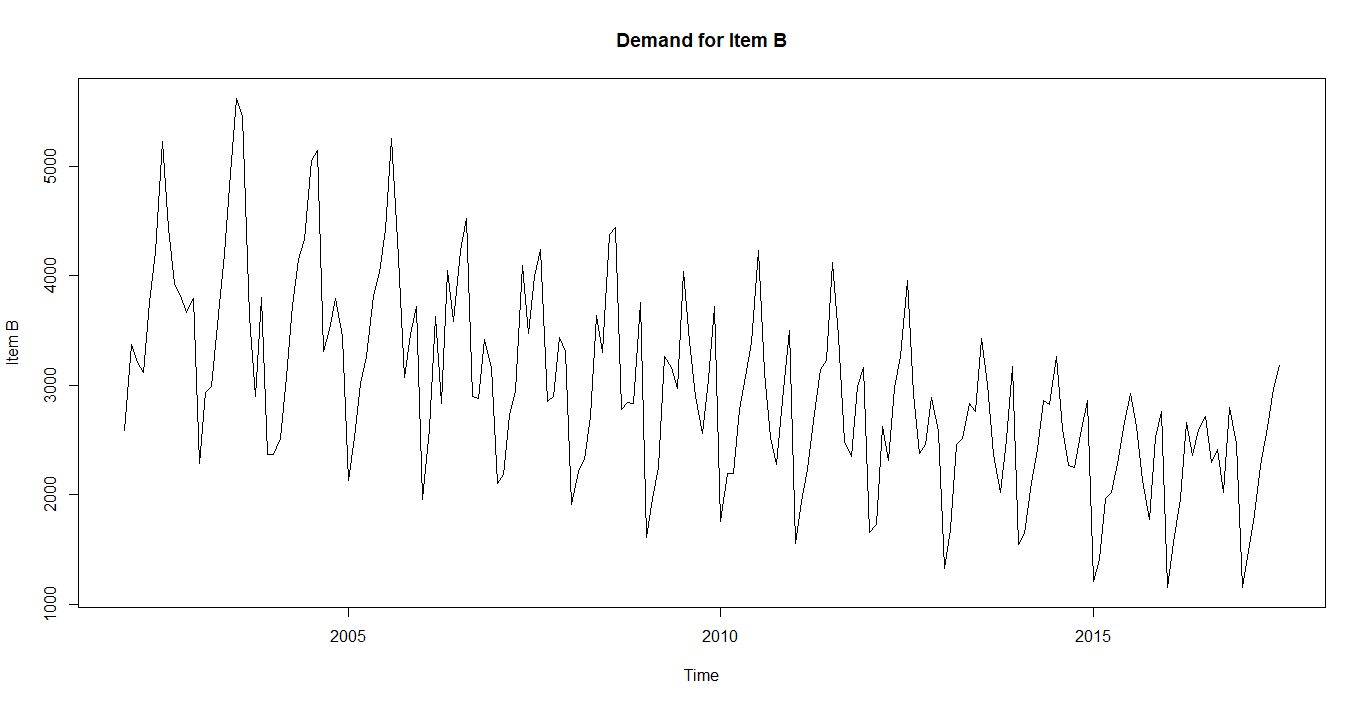
> DemandA <- ts(DemandAB[,3], start=c(2002,1), frequency=12)

> plot(DemandA, main = "Demand for Item A")



> DemandB <- ts(DemandAB[,4], start=c(2002,1), frequency=12)

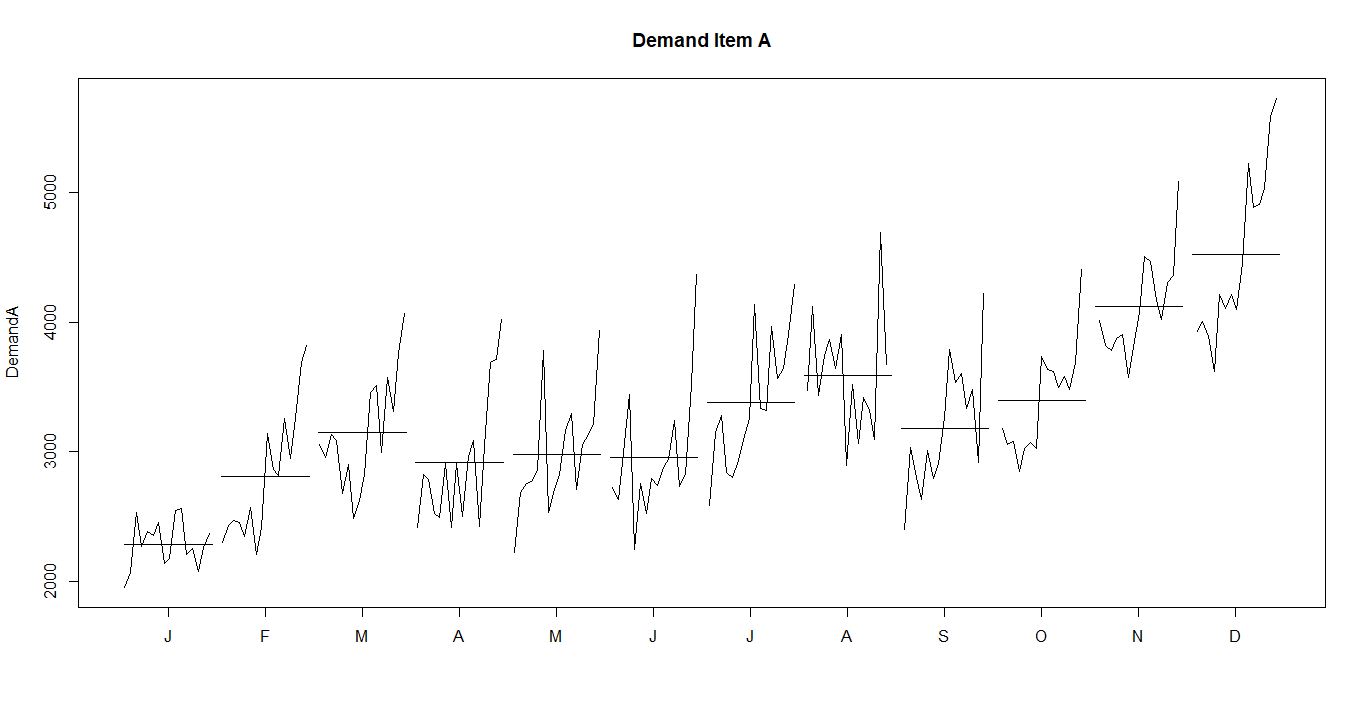
> plot(DemandB,main = "Demand for Item B")



3. Checking for seasonal changes

**Item A**

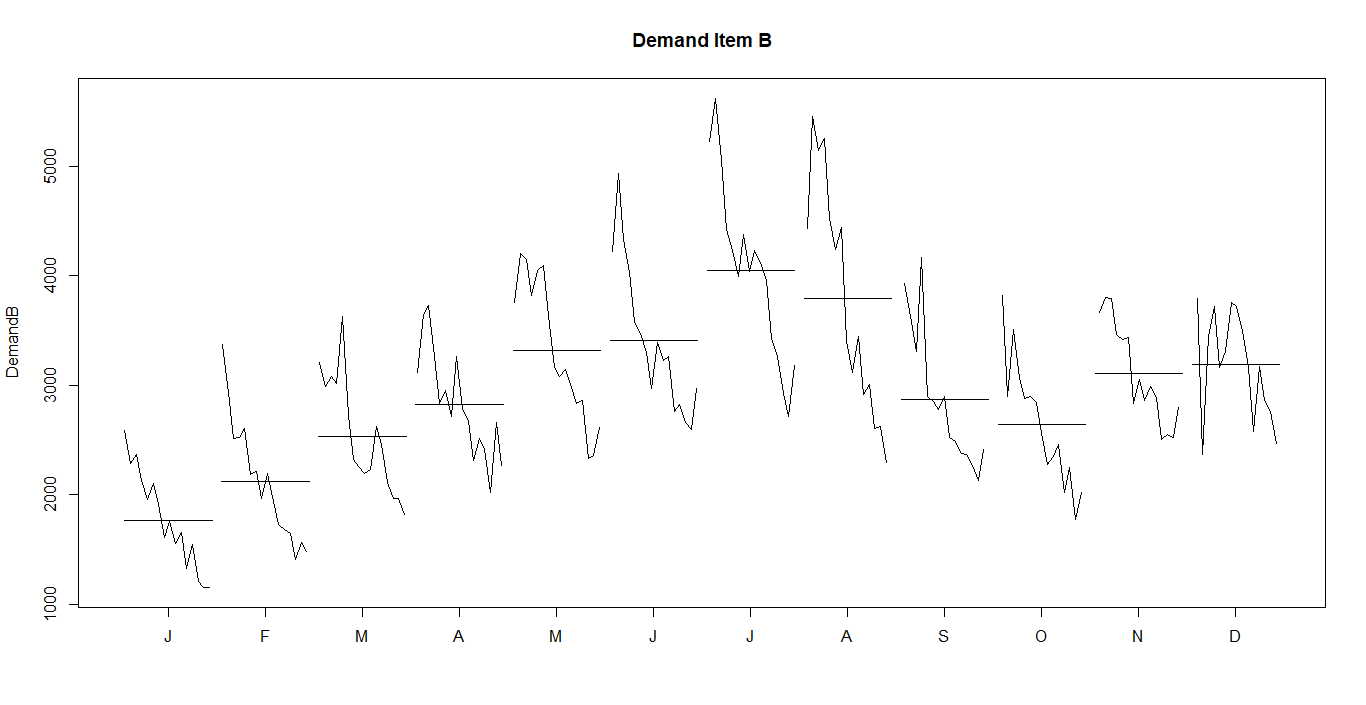
> monthplot(DemandA, main = "Demand Item A")



Here the average values for months February to August is almost same. But in November and December it increases and reaches the peak. Also the graph is showing some variances in most of the months. It showing highest variation in December. Also it has some overall increasing demand .

**Item B**

> monthplot(DemandB, main = "Demand Item B")



Here the averages are increasing from January and reaches the peak at june and again reduces. Here in june, july and august the demand for B is high. But it have a high variance in all the months. From the graph it is clear that the demand for B is reducing over years.

Checking moving averages with different periods,

**Item A**

> DemandA3 = ma(DemandA, order = 3)

> DemandA9 = ma(DemandA, order = 9)

> DemandA15 = ma(DemandA, order = 15)

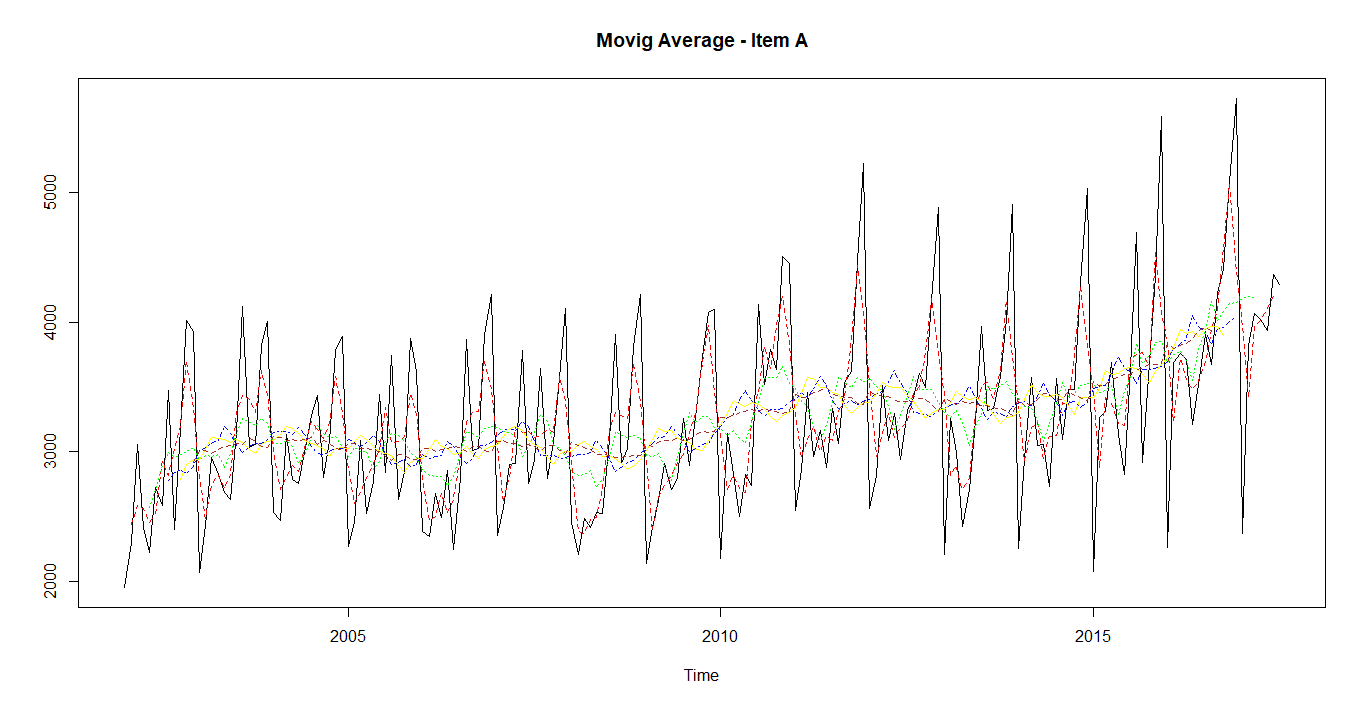
> DemandA19 = ma(DemandA, order = 19)

> DemandA23 = ma(DemandA, order = 23)

> ts.plot(DemandA,DemandA3,DemandA9,DemandA15,DemandA19,DemandA23,lty=c(1:4),

+ col=c('black','red','green','blue','yellow','brown'),

+ main = "Movig Average - Item A")



Selecting 23 as window size since it looks optimal compared to others.

**Item B**

> DemandB3 = ma(DemandB, order = 3)

> DemandB9 = ma(DemandB, order = 9)

> DemandB15 = ma(DemandB, order = 15)

> DemandB19 = ma(DemandB, order = 19)

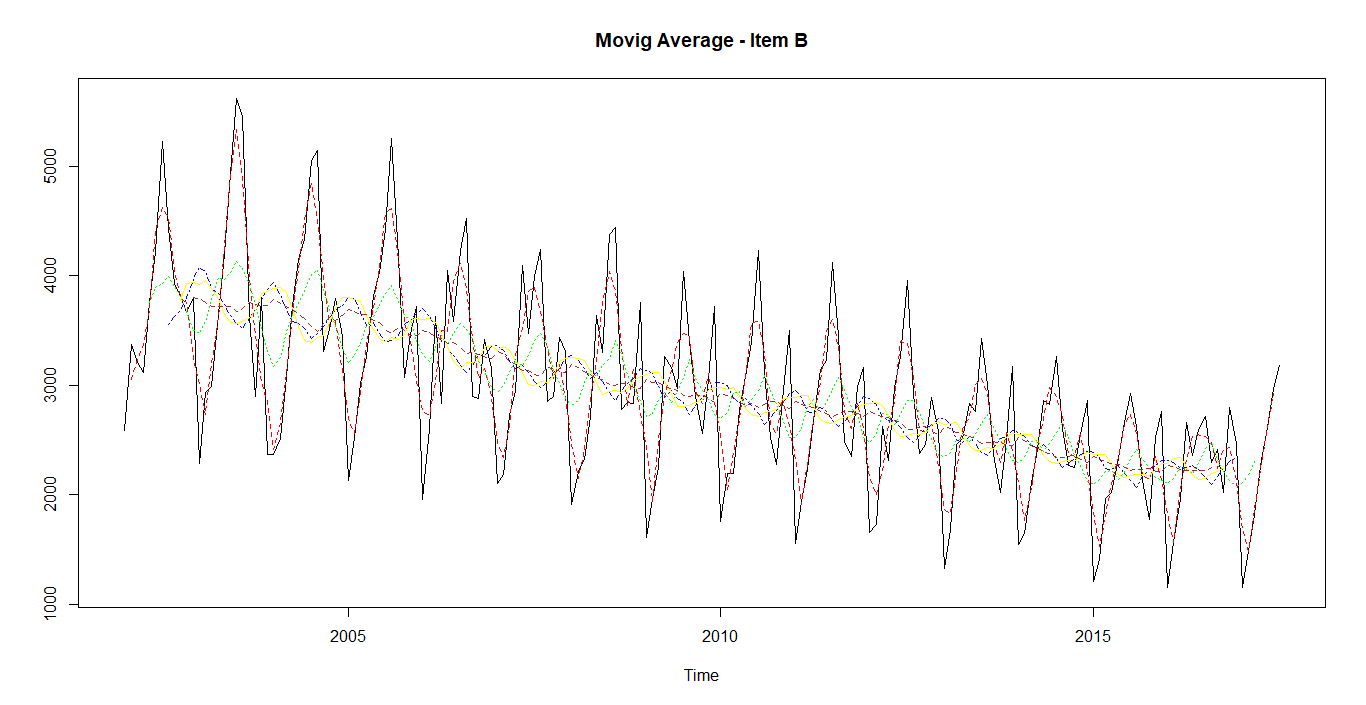
> DemandB23 = ma(DemandB, order = 23)

>

> ts.plot(DemandB,DemandB3,DemandB9,DemandB15,DemandB19,DemandB23,lty=c(1:4),

+ col=c('black','red','green','blue','yellow','brown'),

+ main = "Movig Average - Item B")



Selecting 19 as window size since it looks optimal compared to others.

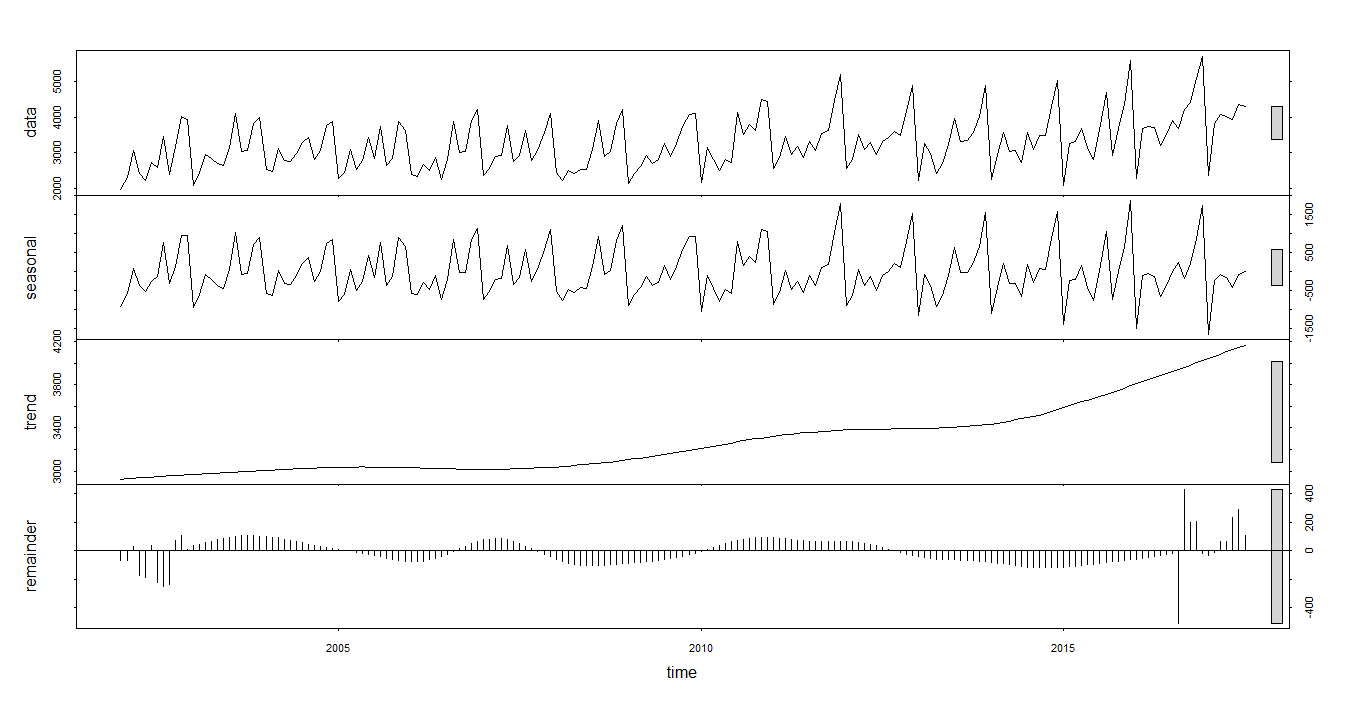
4. **Item A**

Here it is not sure whether Demand of A is addictive or multiplicative series, trying to decompose A in both addictive and multiplicative ways.

Considering A as Addictive Series,

> AAdDec2<-stl(DemandA[,1], s.window=2)

> plot(AAdDec2)



From the addictive decomposition plot we can see that seasonality component

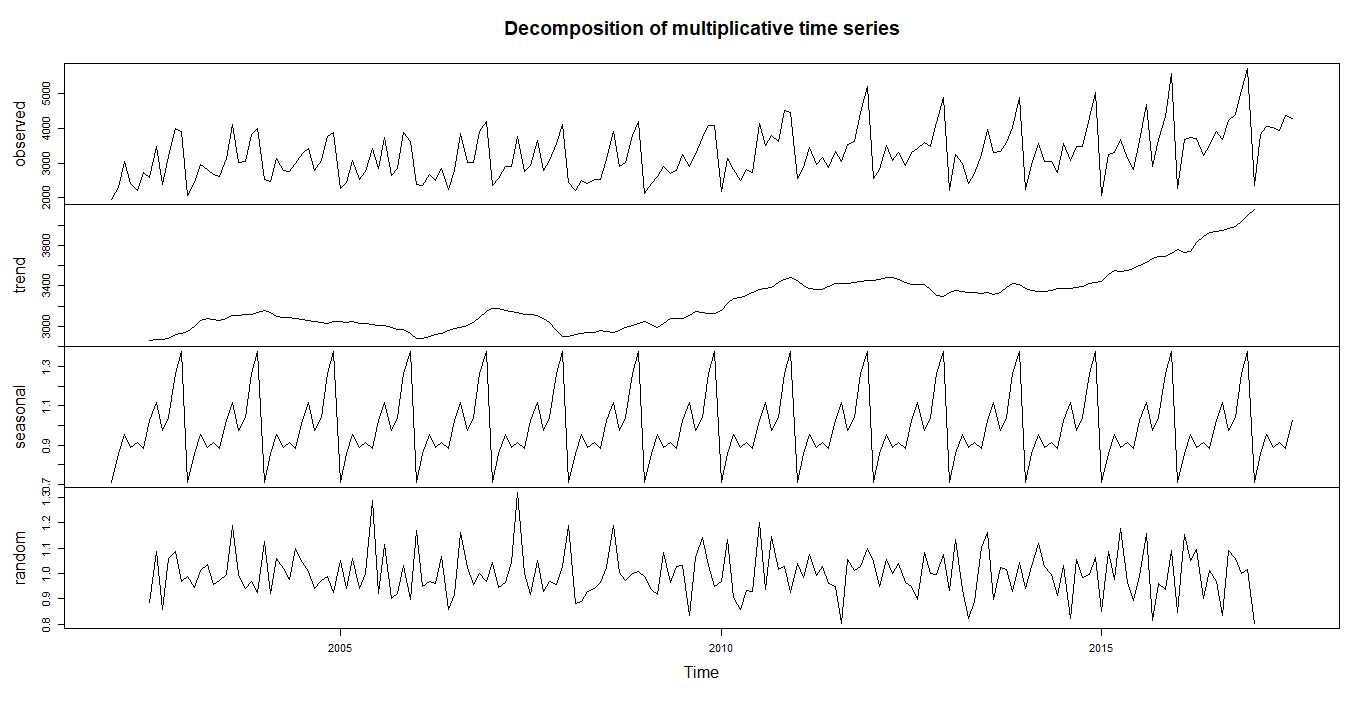
have more importance than trend and reminder is much less compared to trend

and seasonality.

Now considering A as multiplicative Series,

> ADec<-decompose(DemandA, type = "m")

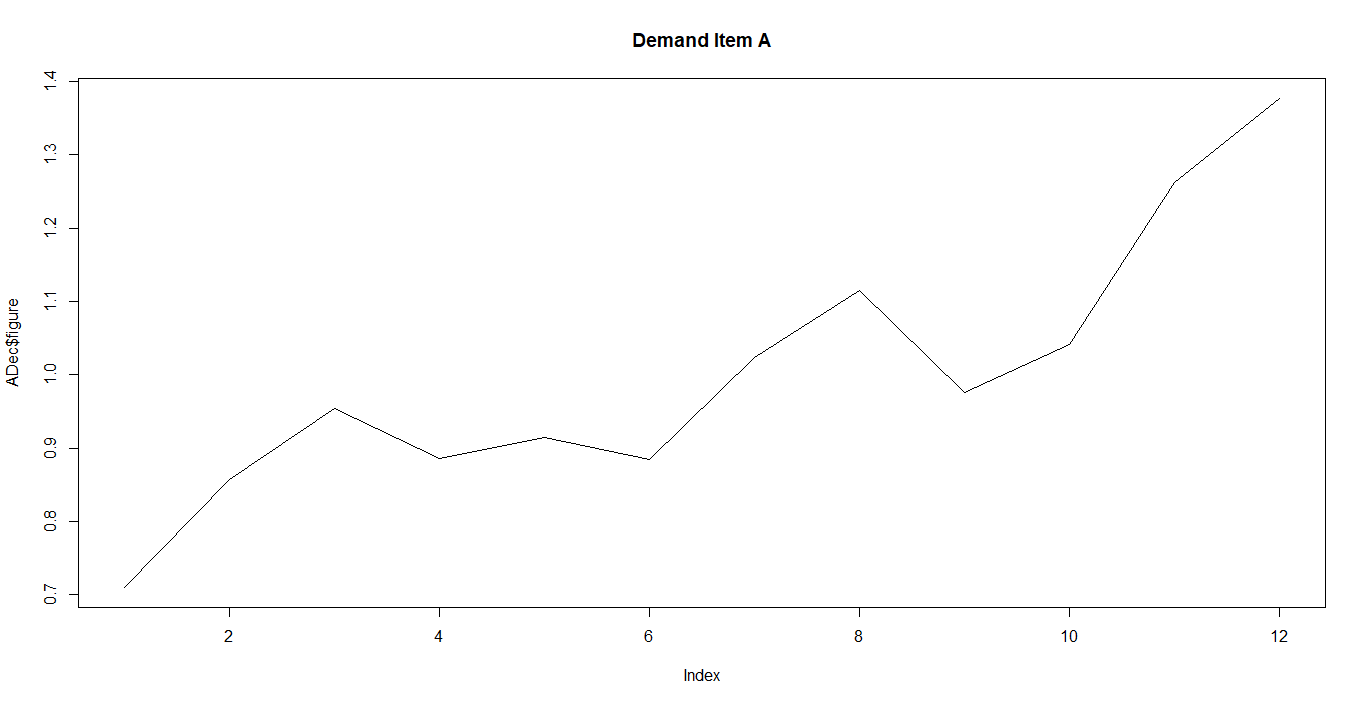
> plot(ADec)



Here multiplicative model is also showing good model. In this, a better seasonality is extracted. Here trend is dominating since it is considered as a multiplicative model.

Checking the seasonality

> plot(ADec$figure, type = "l",main = "Demand Item A")



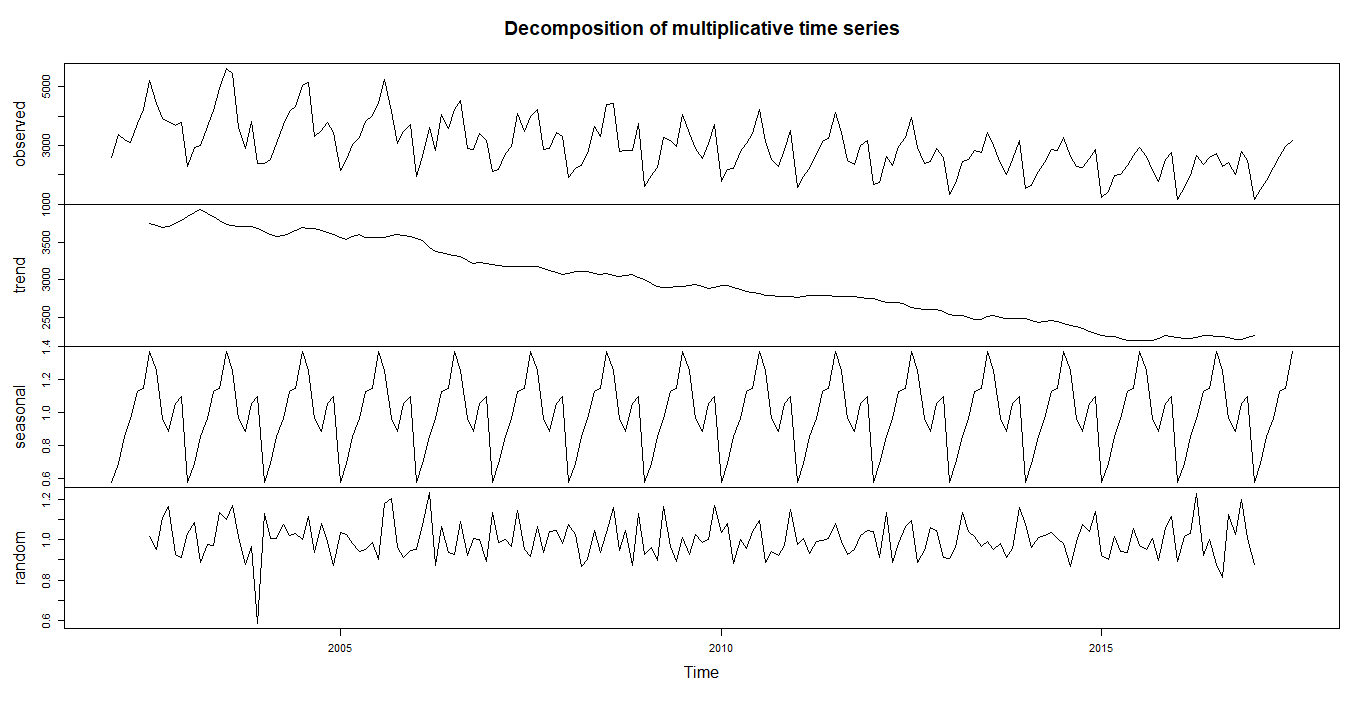
For item A, average sales is highest in November and December of every year and it is lowest in January. Here the average sales gradually increases from January to December even though there is dip in months April, June and September. Also the demand of item A slightly increasing over years from 2010.

**Item B**

Here demand of item is multiplicative. So taking multiplicative decomposition model.

> BDec<-decompose(DemandB, type = "m")

> plot(BDec)



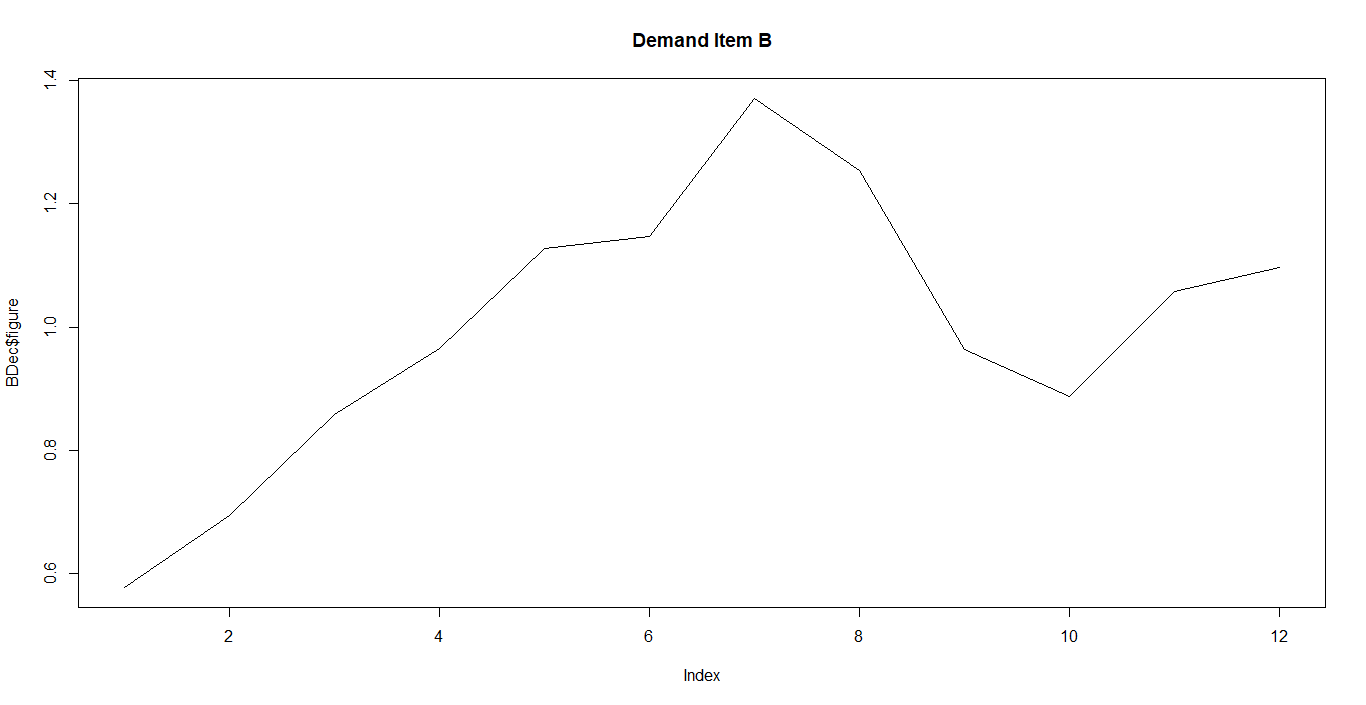
The demand of B is showing a very good negative trend and a good seasonality.

> BDec$figure

[1] 0.5775388 0.6938067 0.8583482 0.9656827 1.1273059 1.1472868 1.3712496 1.2537145 0.9628808 0.8877766

[11] 1.0572445 1.0971649

> plot(BDec$figure, type = "l", main = "Demand Item B")



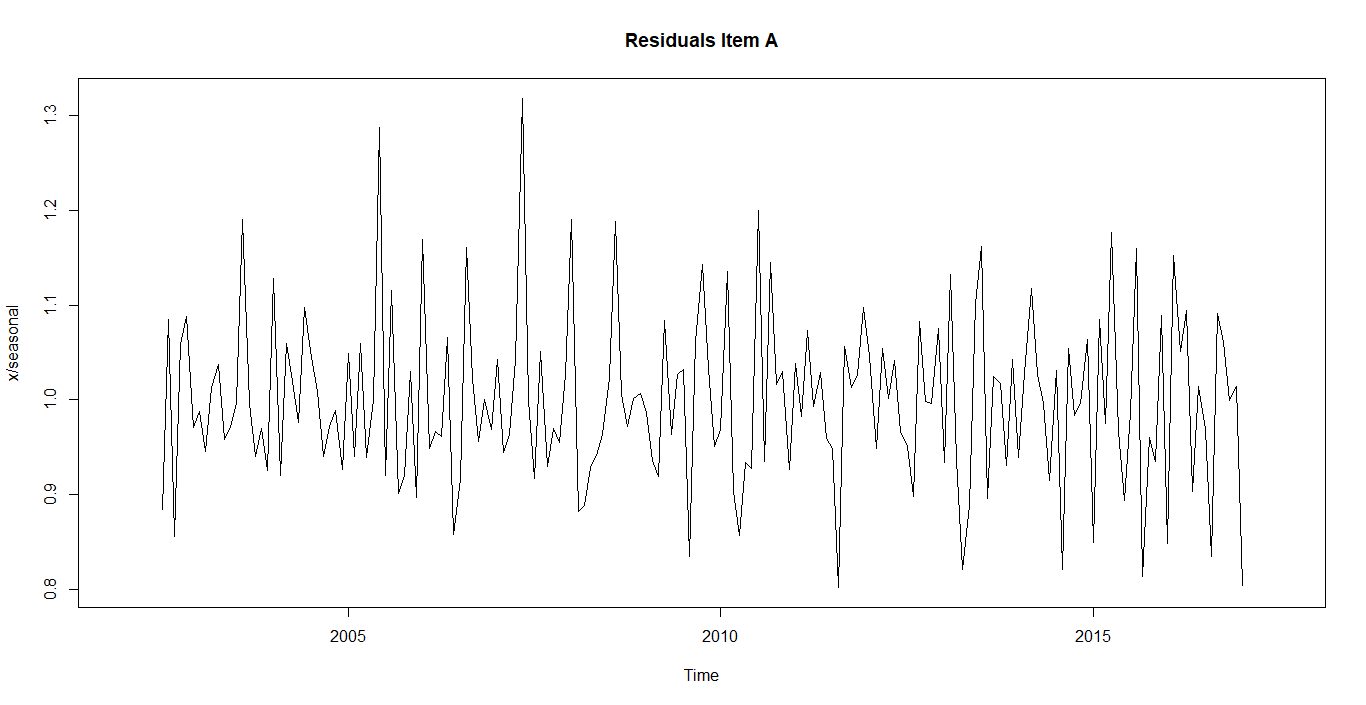
From the above graph it is clear that average sales increases from January and reaches its peak at July and then decreases up to October and again raises in December. Here sales is lowest in January. For item A also sales was lowest in January. But for item B, the demand is gradually decreasing over years, from the trend graph it is clear. That is when comparing the demand of A & B we can see that the demand of item is slightly increasing in each year and demand of B gradually decreasing.

5.

Checking the residuals for both decompositions

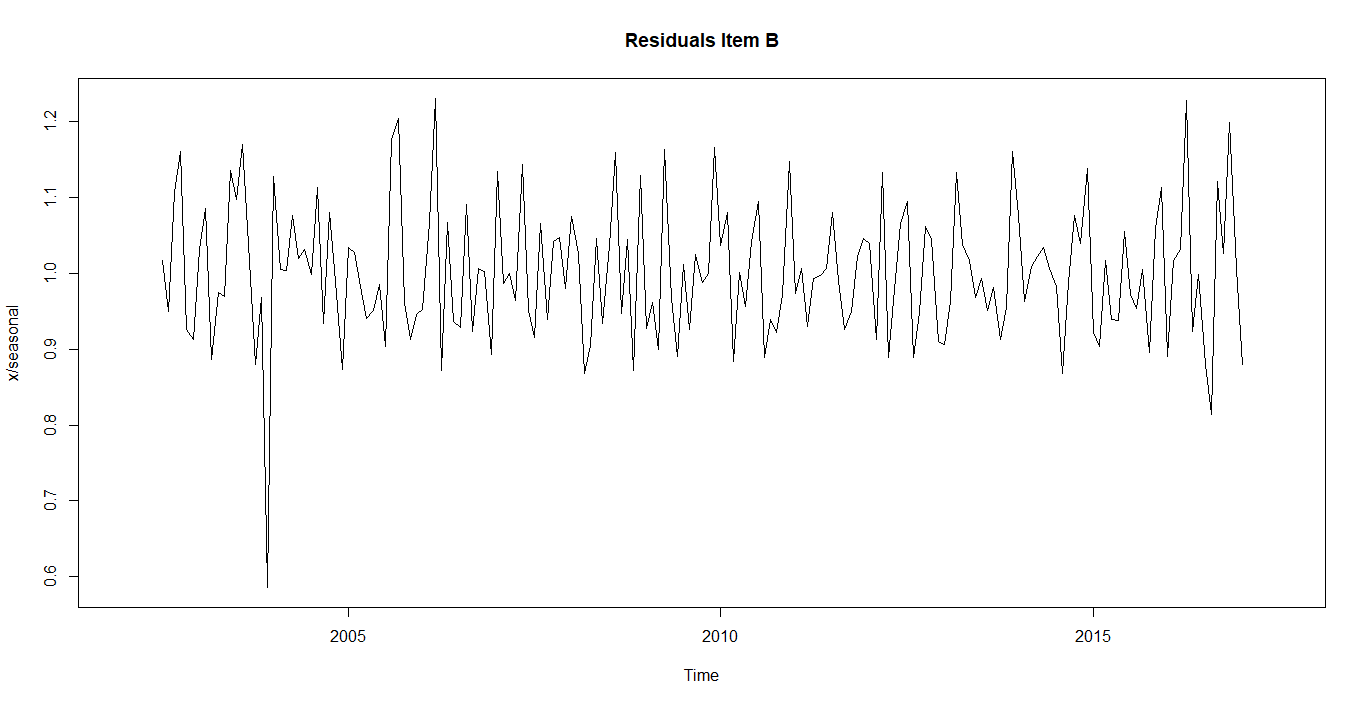
**Item A**

> plot(remainder(ADec), main = "Residuals Item A")



Here the residuals doesn’t follow any particular pattern. It suddenly increases and decreases without any pattern.

> plot(remainder(BDec), main = "Residuals Item B")



Residuals of B also not showing any pattern and it doesn’t have much contribution in the entire time series model.

**6. Model**

**Item A**

Dividing a time series into train and test data,

> ATrain <- window(DemandA, start=c(2002,1), end=c(2015,10), frequency=12)

> ATest <- window(DemandA, start=c(2015,11), frequency=12)

Made last 21 months data for testing purpose.

**Considering A as multiplicative**

**Applying holt winters method,**

> ATrain.fc = hw(ATrain, seasonal = 'm', h=21)

> ATrain.fc

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

Nov 2015 4448.660 3853.461 5043.858 3538.383 5358.937

Dec 2015 4782.008 4137.991 5426.026 3797.068 5766.948

Jan 2016 2572.349 2223.542 2921.157 2038.894 3105.805

Feb 2016 2981.604 2574.423 3388.785 2358.874 3604.334

Mar 2016 3382.737 2917.368 3848.107 2671.016 4094.458

Apr 2016 3143.802 2708.004 3579.599 2477.307 3810.297

May 2016 3248.357 2794.519 3702.194 2554.272 3942.441

Jun 2016 3135.528 2693.899 3577.157 2460.115 3810.942

Jul 2016 3626.514 3111.466 4141.563 2838.815 4414.213

Aug 2016 3982.101 3411.699 4552.504 3109.746 4854.457

Sep 2016 3430.307 2934.620 3925.994 2672.219 4188.395

Oct 2016 3719.759 3177.399 4262.119 2890.291 4549.227

Nov 2016 4514.485 3850.170 5178.800 3498.503 5530.467

Dec 2016 4852.679 4131.869 5573.488 3750.296 5955.062

Jan 2017 2610.318 2218.857 3001.779 2011.630 3209.006

Feb 2017 3025.559 2567.380 3483.739 2324.834 3726.285

Mar 2017 3432.545 2907.547 3957.542 2629.631 4235.459

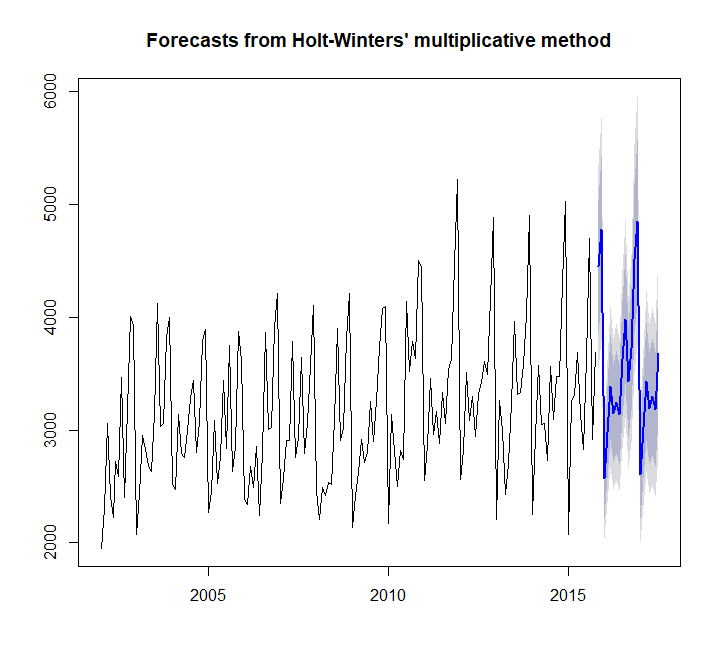
Apr 2017 3190.034 2697.179 3682.890 2436.277 3943.792

May 2017 3296.068 2781.584 3810.553 2509.232 4082.904

Jun 2017 3181.526 2679.729 3683.324 2414.093 3948.960

Jul 2017 3679.651 3093.136 4266.165 2782.654 4576.647

> plot(ATrain.fc)



> ATrain.fc$model

Holt-Winters' multiplicative method

Call:

hw(y = ATrain, h = 21, seasonal = "m")

Smoothing parameters:

alpha = 0.1114

beta = 0.0032

gamma = 1e-04

Initial states:

l = 2979.4887

b = 17.4149

s = 1.359 1.2658 1.0443 0.9642 1.1206 1.0218

0.8845 0.9175 0.8891 0.9578 0.8453 0.7301

sigma: 0.1044

AIC AICc BIC

2789.076 2793.211 2841.980

**Smoothing Parameters got from the iteration are,**

alpha = 0.1114

beta = 0.0032

gamma = 1e-04

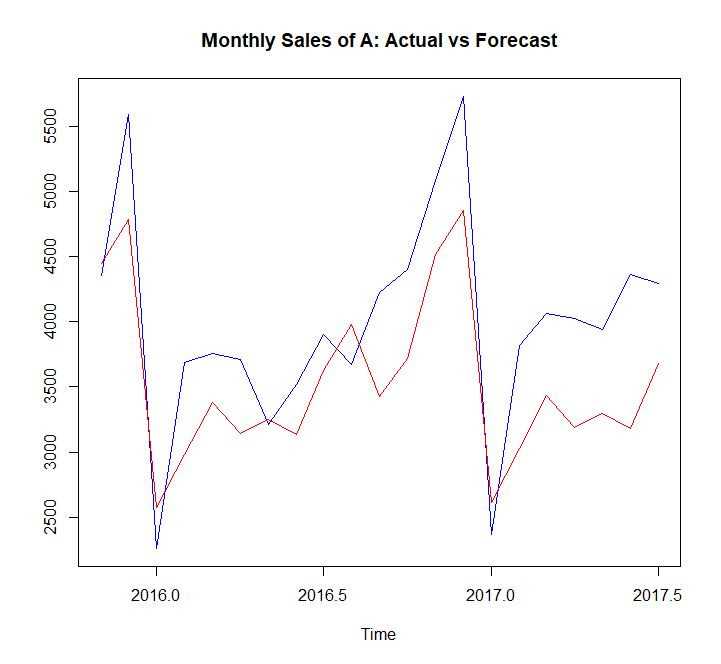
What role do Smoothing Parameters play in the modeling?

Here alpha specifies how to smooth the level component, beta specifies how to smooth the trend component and gamma specifies how to smooth the seasonal component. So these smoothing parameters decides how to smooth each components of the model.

**Checking MAPE value,**

> Vec<- cbind(ATest,ATrain.fc$mean)

> ts.plot(Vec, col=c("blue", "red"), main="Monthly Sales of A: Actual vs Forecast")



Here the actual and forecasted data are showing little difference. The forecasted data seems like bit biased to downward. May be adjustment of smoothing parameters are required

> MAPE <- mean(abs(Vec[,1]-Vec[,2])/Vec[,1])

> MAPE

[1] 0.1369681

Here we are getting MAPE as 13.69%

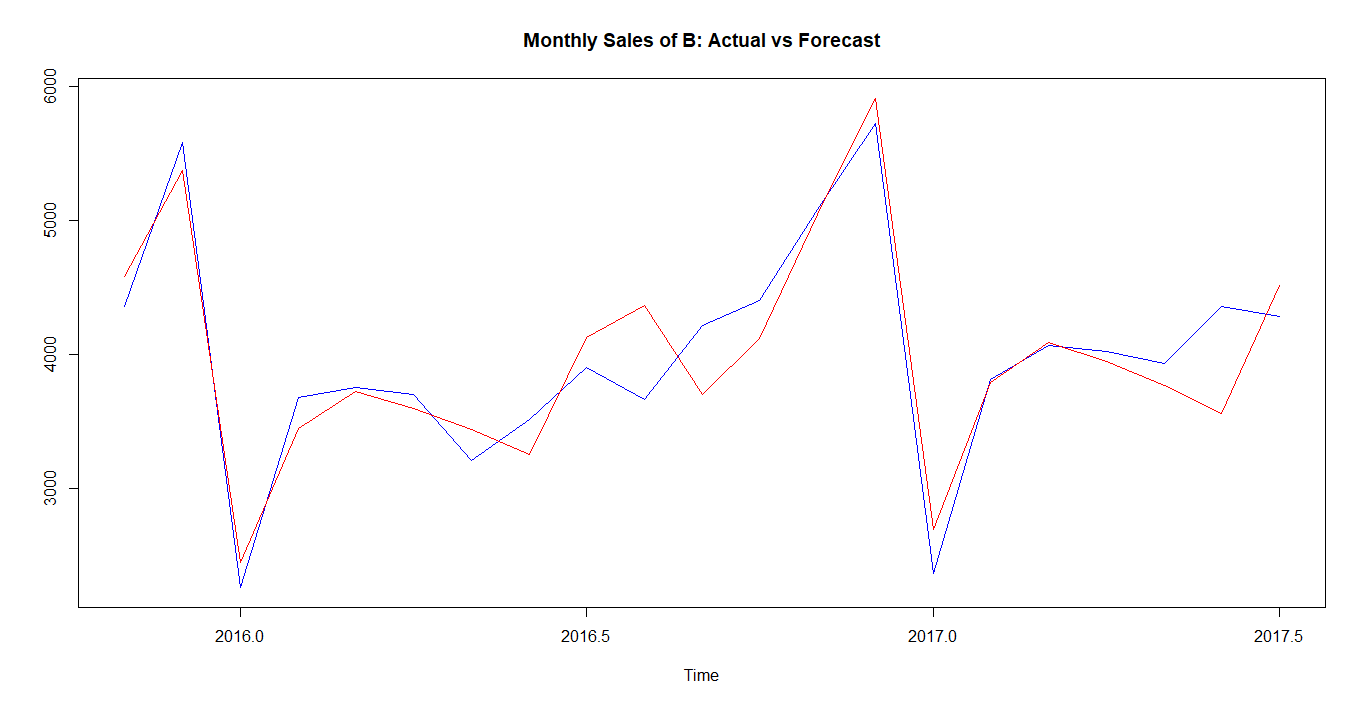
**Need to adjust the smoothing parameters to get a better model,**

Remodeling the train data with alpha=0.05,beta=0.03,gamma=0.4,

> ATrain.fc1 = hw(ATrain, seasonal = 'm',h=21,alpha=0.05,beta=0.03,gamma=0.4)

> Vec1<- cbind(ATest,ATrain.fc1$mean)

> ts.plot(Vec1, col=c("blue", "red"), main="Monthly Sales of B: Actual vs Forecast")



> MAPE1 <- mean(abs(Vec1[,1]-Vec1[,2])/Vec1[,1])

> MAPE1

[1] 0.06389478

Now MAPE ha came down to 6.3% also the forecasted values are almost in line with the actual values, which is very good compared to the previous model.

**Considering A as addictive,**

|  |
| --- |
| > ATrain.fca = hw(ATrain, seasonal = 'a',h=21)  > ATrain.fca  Point Forecast Lo 80 Hi 80 Lo 95 Hi 95  Nov 2015 4350.111 3931.587 4768.635 3710.034 4990.188  Dec 2015 4677.277 4258.040 5096.514 4036.109 5318.445  Jan 2016 2619.046 2199.095 3038.998 1976.786 3261.307  Feb 2016 3033.014 2612.347 3453.681 2389.659 3676.369  Mar 2016 3365.298 2943.914 3786.682 2720.847 4009.749  Apr 2016 3139.434 2717.332 3561.536 2493.884 3784.983  May 2016 3282.637 2859.815 3705.458 2635.987 3929.286  Jun 2016 3177.294 2753.753 3600.836 2529.543 3825.046  Jul 2016 3623.661 3199.398 4047.925 2974.806 4272.516  Aug 2016 3883.883 3458.897 4308.870 3233.923 4533.844  Sep 2016 3498.994 3073.284 3924.705 2847.926 4150.063  Oct 2016 3669.496 3243.060 4095.932 3017.318 4321.673  Nov 2016 4392.124 3964.959 4819.289 3738.832 5045.416  Dec 2016 4719.290 4291.397 5147.183 4064.885 5373.695  Jan 2017 2661.059 2232.438 3089.681 2005.539 3316.579  Feb 2017 3075.027 2645.675 3504.379 2418.390 3731.664  Mar 2017 3407.311 2977.228 3837.394 2749.556 4065.066  Apr 2017 3181.447 2750.631 3612.262 2522.571 3840.322  May 2017 3324.650 2893.100 3756.199 2664.652 3984.647  Jun 2017 3219.307 2787.023 3651.592 2558.186 3880.429  Jul 2017 3665.674 3232.654 4098.695 3003.427 4327.922  > plot(ATrain.fca)    > ATrain.fca$model  Holt-Winters' additive method  Call:  hw(y = ATrain, h = 21, seasonal = "a")  Smoothing parameters:  alpha = 0.0583  beta = 1e-04  gamma = 1e-04  Initial states:  l = 2952.9218  b = 3.5342  s = 1166.348 842.6795 123.5558 -43.3763 344.8795 88.2216  -354.6246 -245.7344 -385.4842 -156.1634 -484.9029 -895.3982  sigma: 326.5761  AIC AICc BIC  2787.602 2791.737 2840.505  **Smoothing parameters we got,**  alpha = 0.0583  beta = 1e-04  gamma = 1e-04  **Checking MAPE,**  > Vec2<- cbind(ATest,ATrain.fca$mean)  > ts.plot(Vec2, col=c("blue", "red"), main="Monthly Sales of A: Actual vs Forecast")    > MAPE2 <- mean(abs(Vec2[,1]-Vec2[,2])/Vec2[,1])  > MAPE2  [1] 0.1384765  MAPE is 13.84% which is high also the forecasted values looks biased as same as  what we found in the previous model, So need to adjust smoothing parameters.  **Adjusting alpha, beta, gamma to get a better model,**  > ATrain.fca1 = hw(ATrain, seasonal = 'a',h=21,alpha=0.055,beta=0.035,gamma=0.4)  > Vec4<- cbind(ATest,ATrain.fca1$mean)  > ts.plot(Vec4, col=c("blue", "red"), main="Monthly Sales of A: Actual vs Forecast")    > MAPE4 <- mean(abs(Vec4[,1]-Vec4[,2])/Vec4[,1])  > MAPE4  [1] 0.06815259  After adjusting the smoothing parameters the model improved , but it doesn’t look as good as the model  what we got in multiplicative model. So we will select the multiplicative model to forecast. |
|  |
|  |

**Item B**

Dividing a time series into train and test data,

> BTrain <- window(DemandB, start=c(2002,1), end=c(2015,10), frequency=12)

> BTest <- window(DemandB, start=c(2015,11), frequency=12)

Made last 21 months data for testing purpose.

**Applying holt winters method,**

> BTrain.fc = hw(BTrain, seasonal = 'm',h=21)

> BTrain.fc

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

Nov 2015 2315.306 2017.767 2612.845 1860.2598 2770.352

Dec 2015 2345.123 2043.672 2646.573 1884.0941 2806.151

Jan 2016 1309.081 1140.756 1477.405 1051.6508 1566.511

Feb 2016 1594.748 1389.623 1799.874 1281.0364 1908.460

Mar 2016 1877.245 1635.693 2118.798 1507.8225 2246.668

Apr 2016 2047.756 1784.153 2311.359 1644.6098 2450.902

May 2016 2446.959 2131.822 2762.097 1964.9978 2928.921

Jun 2016 2487.709 2167.160 2808.259 1997.4709 2977.947

Jul 2016 3008.529 2620.653 3396.405 2415.3233 3601.735

Aug 2016 2802.819 2441.242 3164.396 2249.8350 3355.803

Sep 2016 2066.804 1799.998 2333.609 1658.7592 2474.848

Oct 2016 1908.396 1661.860 2154.932 1531.3514 2285.440

Nov 2016 2194.605 1910.870 2478.341 1760.6695 2628.541

Dec 2016 2222.334 1934.769 2509.900 1782.5407 2662.128

Jan 2017 1240.238 1079.606 1400.870 994.5726 1485.904

Feb 2017 1510.514 1314.683 1706.345 1211.0159 1810.012

Mar 2017 1777.651 1546.941 2008.360 1424.8112 2130.490

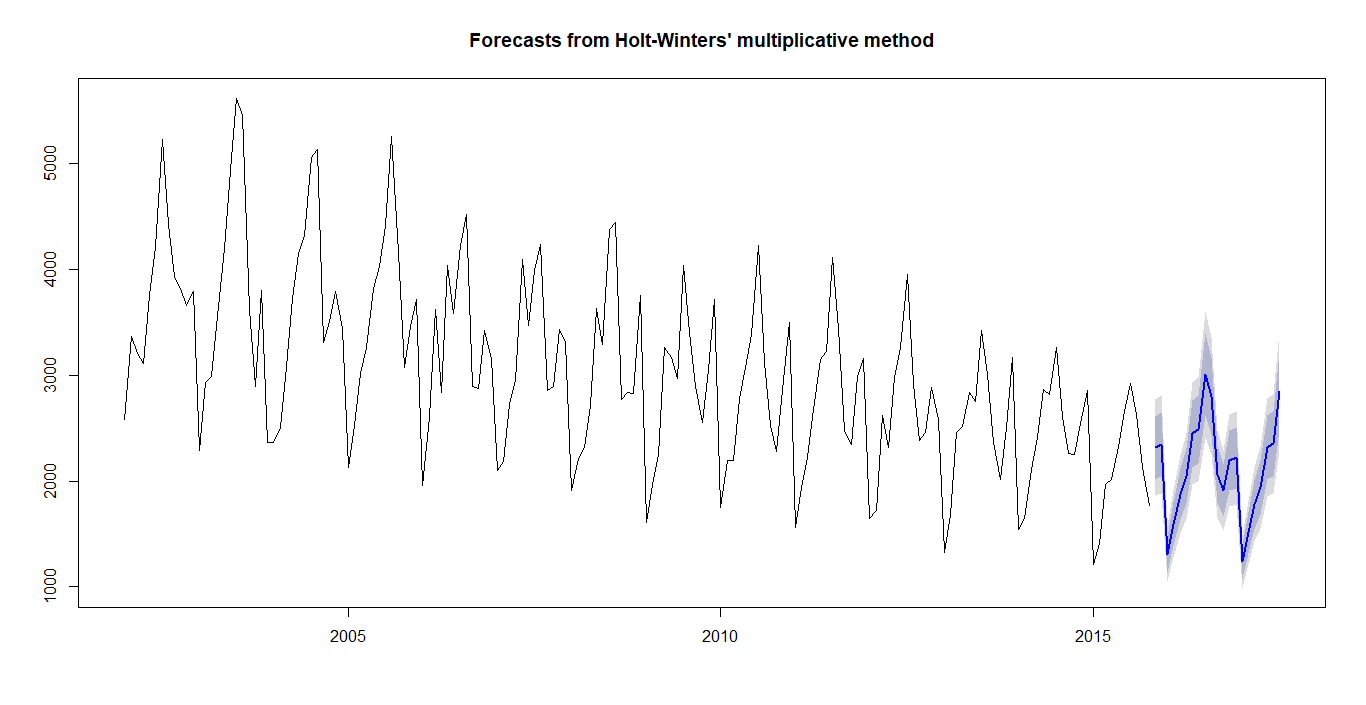
Apr 2017 1938.633 1686.744 2190.522 1553.4020 2323.864

May 2017 2315.981 2014.696 2617.267 1855.2054 2776.758

Jun 2017 2353.953 2047.330 2660.577 1885.0129 2822.894

Jul 2017 2846.042 2474.805 3217.280 2278.2842 3413.801

> plot(BTrain.fc)



> BTrain.fc$model

Holt-Winters' multiplicative method

Call:

hw(y = BTrain, h = 21, seasonal = "m")

Smoothing parameters:

alpha = 0.0215

beta = 0.0013

gamma = 1e-04

Initial states:

l = 4062.3407

b = -12.6974

s = 1.0511 1.0333 0.8945 0.9643 1.3019 1.3911

1.1451 1.1213 0.9342 0.8526 0.7212 0.5894

sigma: 0.1003

AIC AICc BIC

2751.943 2756.078 2804.847

**Smoothing parameters we got are:**

alpha = 0.0215

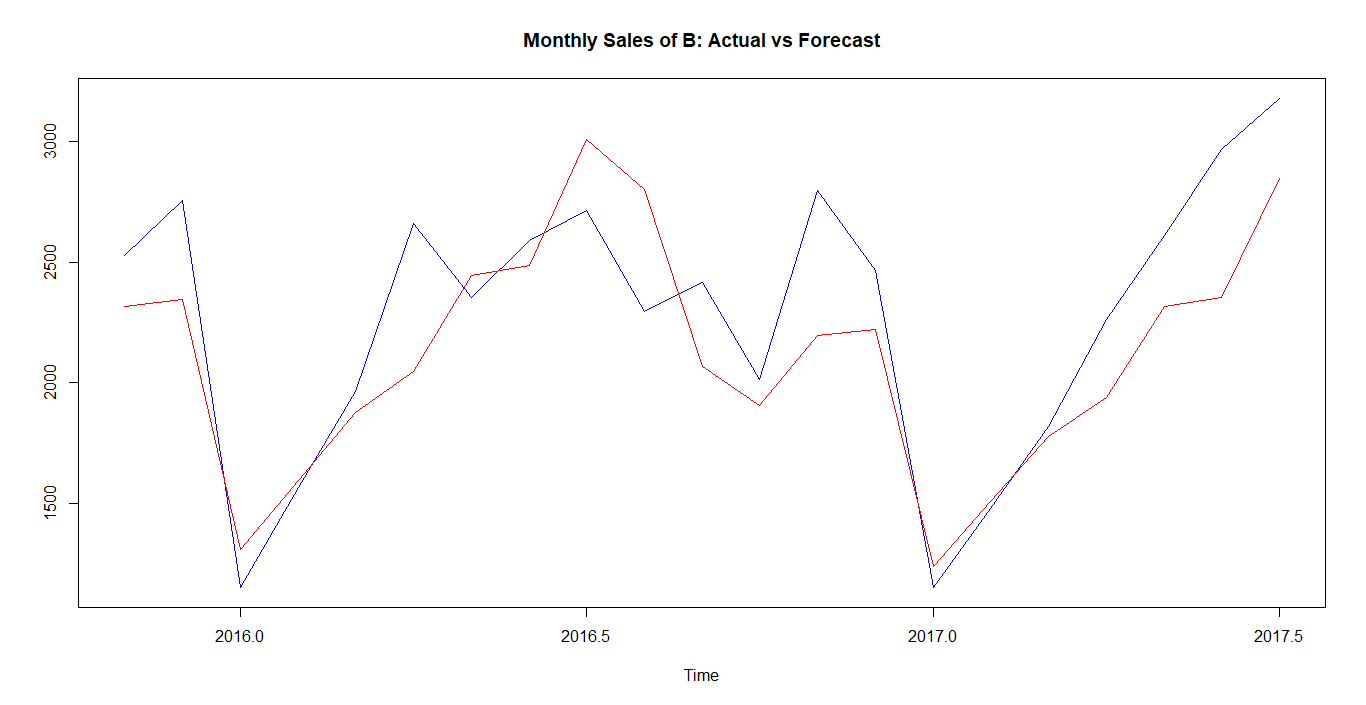
beta = 0.0013

gamma = 1e-04

**Checking MAPE value,**

> Vec5<- cbind(BTest,BTrain.fc$mean)

> ts.plot(Vec5, col=c("blue", "red"), main="Monthly Sales of B: Actual vs Forecast")



Here the actual and forecasted data are showing little difference. May be adjustment of smoothing parameters are required

> MAPE5 <- mean(abs(Vec5[,1]-Vec5[,2])/Vec5[,1])

> MAPE5

[1] 0.1079037

Here we are getting MAPE as 10.79%

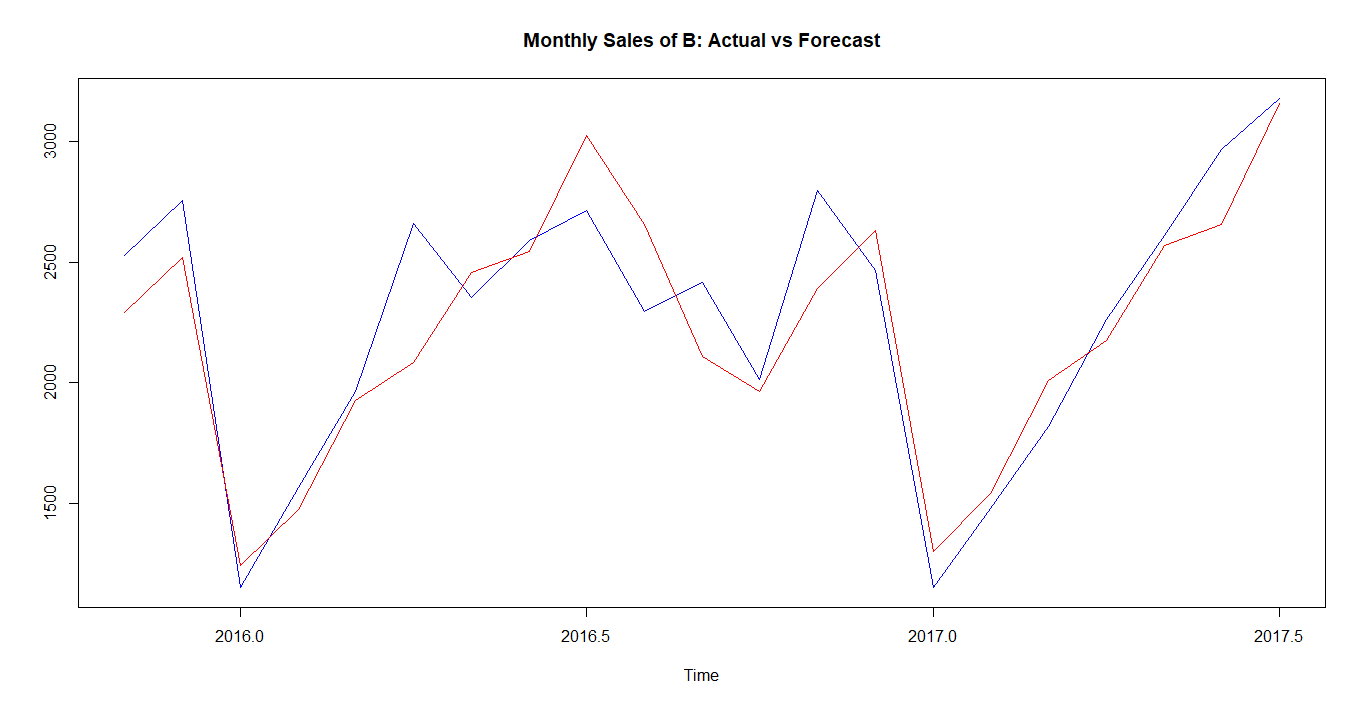
**Need to adjust the smoothing parameters to get a better model,**

Remodeling the train data with alpha=0.2,beta=0.12,gamma=0.18,

> BTrain.fc1 = hw(BTrain, seasonal = 'm',h=21,alpha=0.2,beta=0.12,gamma=0.18)

> Vec6<- cbind(BTest,BTrain.fc1$mean)

> ts.plot(Vec6, col=c("blue", "red"), main="Monthly Sales of B: Actual vs Forecast")



> MAPE6 <- mean(abs(Vec6[,1]-Vec6[,2])/Vec6[,1])

> MAPE6

[1] 0.08072476

Now MAPE ha came down to 8.07% also the forecasted values are almost in line with the actual values, which is very good compared to the previous model.

Here the model for item A demand gives MAPE of 6.3% and model for item B demand gives MAPE of 8.07%. Comparing the MAPE s of two models, model for item B is giving better performance. But comparing the actual v/s forecasted graphs of A and B we can find that, forecasted values of B are more equally distributed around the actual value.

**6. Forecasting**

**Item A**

> DemandA.fc = hw(DemandA, seasonal= 'm',h=17,alpha=0.05,beta=0.03,gamma=0.4)

> DemandA.fc

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

Aug 2017 4538.476 3871.595 5205.358 3518.569 5558.384

Sep 2017 4384.340 3738.071 5030.608 3395.958 5372.722

Oct 2017 4752.955 4048.296 5457.614 3675.272 5830.638

Nov 2017 5619.143 4778.535 6459.752 4333.543 6904.743

Dec 2017 6642.563 5636.162 7648.963 5103.406 8181.719

Jan 2018 2846.173 2407.704 3284.643 2175.592 3516.754

Feb 2018 4408.783 3715.323 5102.243 3348.228 5469.339

Mar 2018 4683.409 3928.236 5438.583 3528.471 5838.348

Apr 2018 4611.589 3846.375 5376.803 3441.295 5781.883

May 2018 4362.573 3615.019 5110.126 3219.289 5505.857

Jun 2018 4579.932 3766.987 5392.877 3336.639 5823.224

Jul 2018 5039.350 4110.361 5968.339 3618.584 6460.116

Aug 2018 5213.541 4088.647 6338.434 3493.164 6933.917

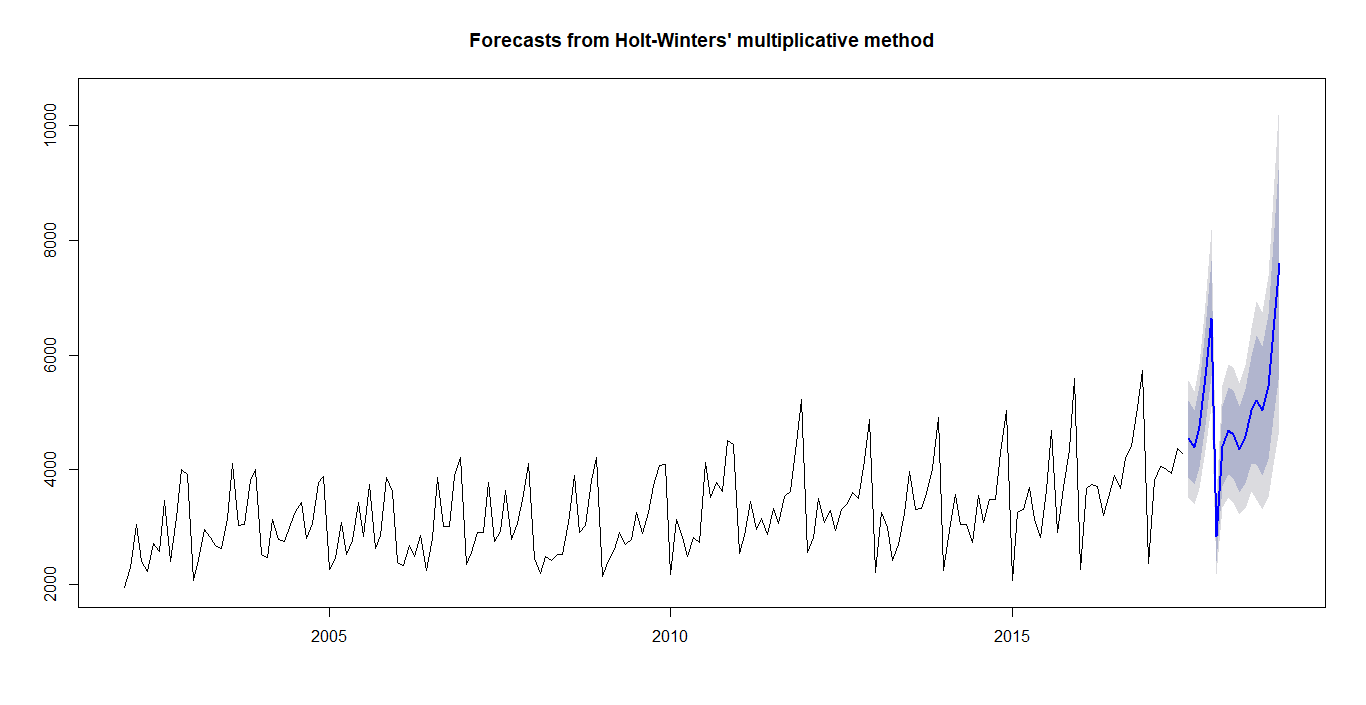
Sep 2018 5028.721 3906.999 6150.444 3313.195 6744.248

Oct 2018 5443.306 4186.107 6700.505 3520.586 7366.025

Nov 2018 6425.830 4887.276 7964.384 4072.815 8778.845

Dec 2018 7585.233 5700.728 9469.737 4703.132 10467.334

> plot(DemandA.fc)



While checking the forecasted values of Demand of A we can find that, the forecasting from Aug 2017 to Jan 2018 gives a very good confidence interval.

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

Aug 2017 4538.476 3871.595 5205.358 3518.569 5558.384

Sep 2017 4384.340 3738.071 5030.608 3395.958 5372.722

Oct 2017 4752.955 4048.296 5457.614 3675.272 5830.638

Nov 2017 5619.143 4778.535 6459.752 4333.543 6904.743

Dec 2017 6642.563 5636.162 7648.963 5103.406 8181.719

Jan 2018 2846.173 2407.704 3284.643 2175.592 3516.754

Also by analyzing the forecasted values we can find that the sales of item A is likely to go up in Nov and Dec 2017 and in January 2018 it is going to drop down. So as a store manager, I will plan to keep more stock of item A in Nov and December. Since the item A sales is likely to go very low in January, can plan for giving some offer for item A.

**Item B**

> DemandB.fc = hw(DemandB, seasonal = 'm',h=17,lpha=0.2,beta=0.12,gamma=0.18)

> DemandB.fc

Point Forecast Lo 80 Hi 80 Lo 95 Hi 95

Aug 2017 2919.130 2452.8293 3385.431 2205.9846 3632.275

Sep 2017 2558.179 2102.3330 3014.026 1861.0226 3255.336

Oct 2017 2339.374 1865.9319 2812.815 1615.3070 3063.440

Nov 2017 3014.888 2317.0537 3712.723 1947.6423 4082.134

Dec 2017 3156.533 2321.3647 3991.702 1879.2532 4433.813

Jan 2018 1517.976 1061.0360 1974.916 819.1466 2216.805

Feb 2018 1927.710 1271.8972 2583.523 924.7310 2930.689

Mar 2018 2522.102 1559.3959 3484.809 1049.7698 3994.435

Apr 2018 2957.232 1699.7756 4214.689 1034.1181 4880.347

May 2018 3306.095 1750.6372 4861.553 927.2274 5684.963

Jun 2018 3560.312 1718.4473 5402.176 743.4233 6377.200

Jul 2018 4091.485 1777.2127 6405.757 552.1111 7630.858

Aug 2018 3708.680 1384.4544 6032.905 154.0838 7263.276

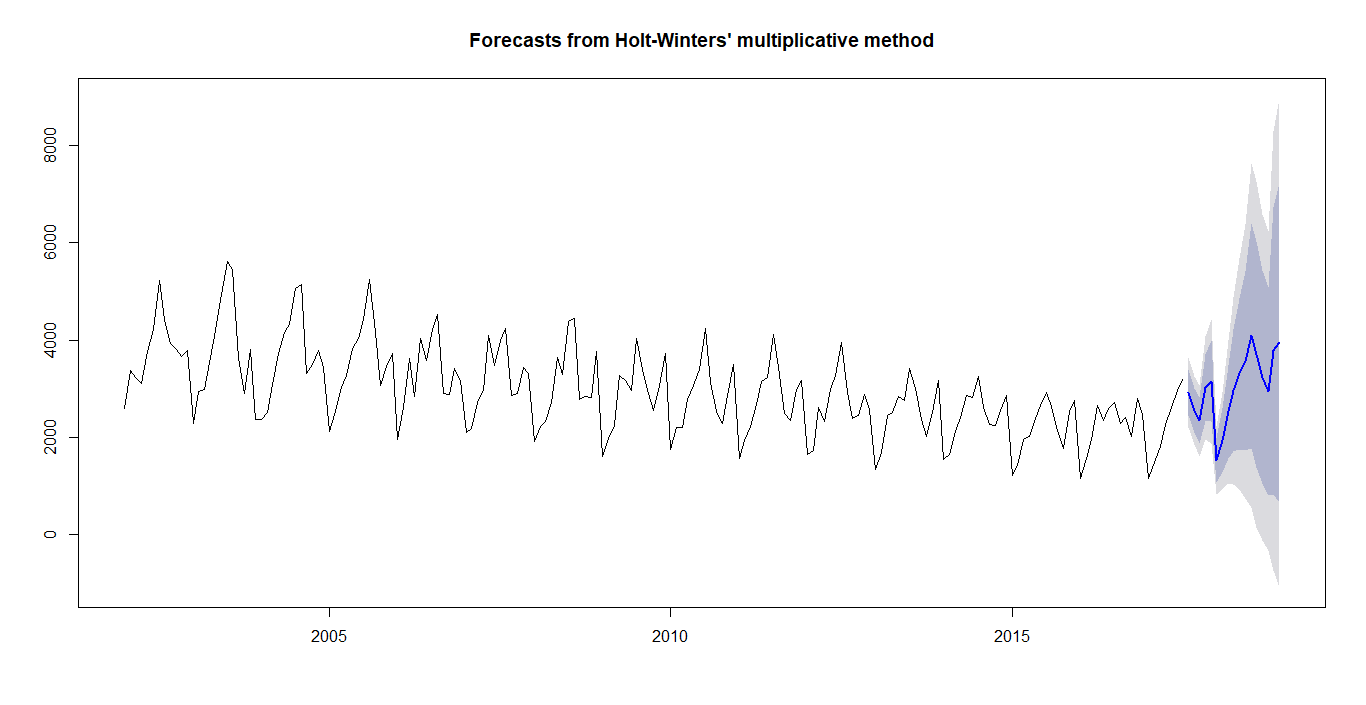
Sep 2018 3235.412 1046.0323 5424.792 -112.9555 6583.780

Oct 2018 2945.818 801.1157 5090.521 -334.2212 6225.858

Nov 2018 3780.559 829.0198 6732.097 -733.4305 8294.548

Dec 2018 3942.215 651.8408 7232.589 -1089.9782 8974.408

> plot(DemandB.fc)



From the forecasted data we can find that, the sales of item B is likely to go down in Jan 2018, so can plan for giving some offer for item B in this month. After January sales may go up as per forecasting, so have to stock keep more stock of item B after January.