## Case Study #1: Production Forecasting

In this case, consider the data on food industry production in the U.S. (*673_case1.csv*). This dataset was extracted from www.kaggle.com and then adopted for this course. The monthly production data is given for a 20-year period, from January 1997 through December of 2016, and measured in millions of tons. The goal is to identify the best forecasting method to predict monthly food production in the U.S. in 12 months of 2017.

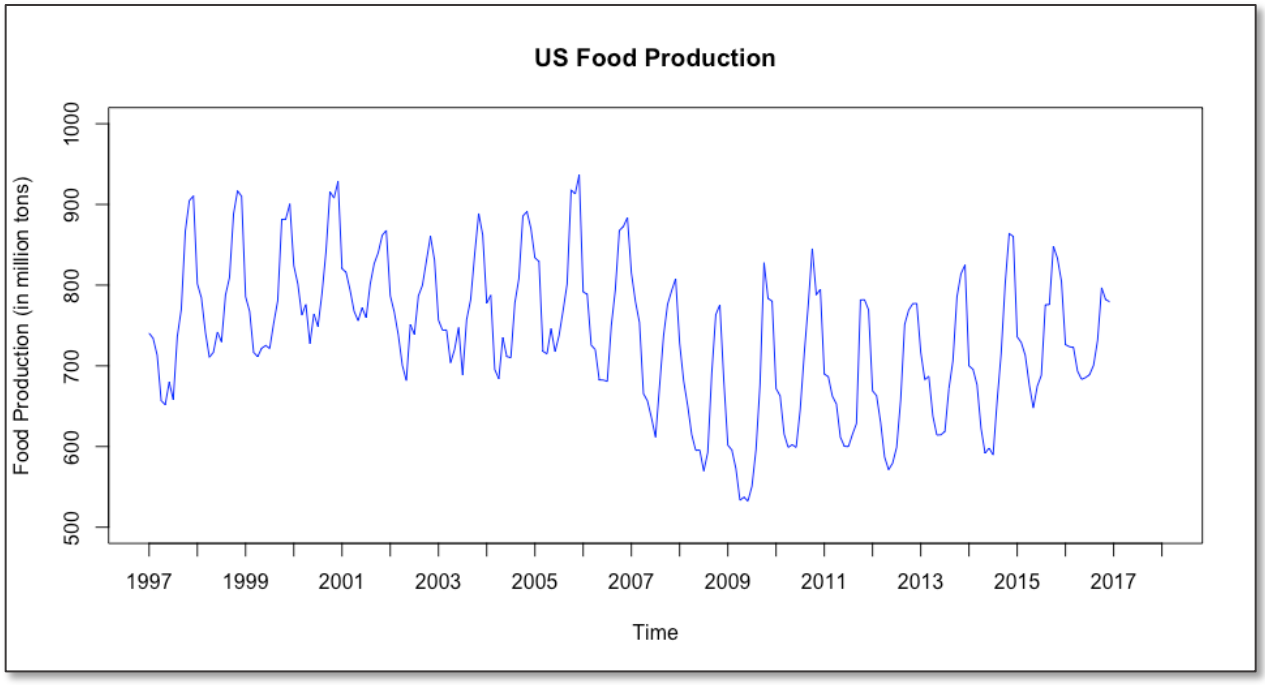
**1. Identify time series components and plot the data.**

**a. Create time series data set *prod.ts* in R using the *ts()* function.**

The given dataset contains the monthly **food production, measured in millions of tons** for a period of **20 years**, from January 1997 to December 2016 for U.S. We have used ts() function to create the time series data using the given dataset. The time series data is shown below.

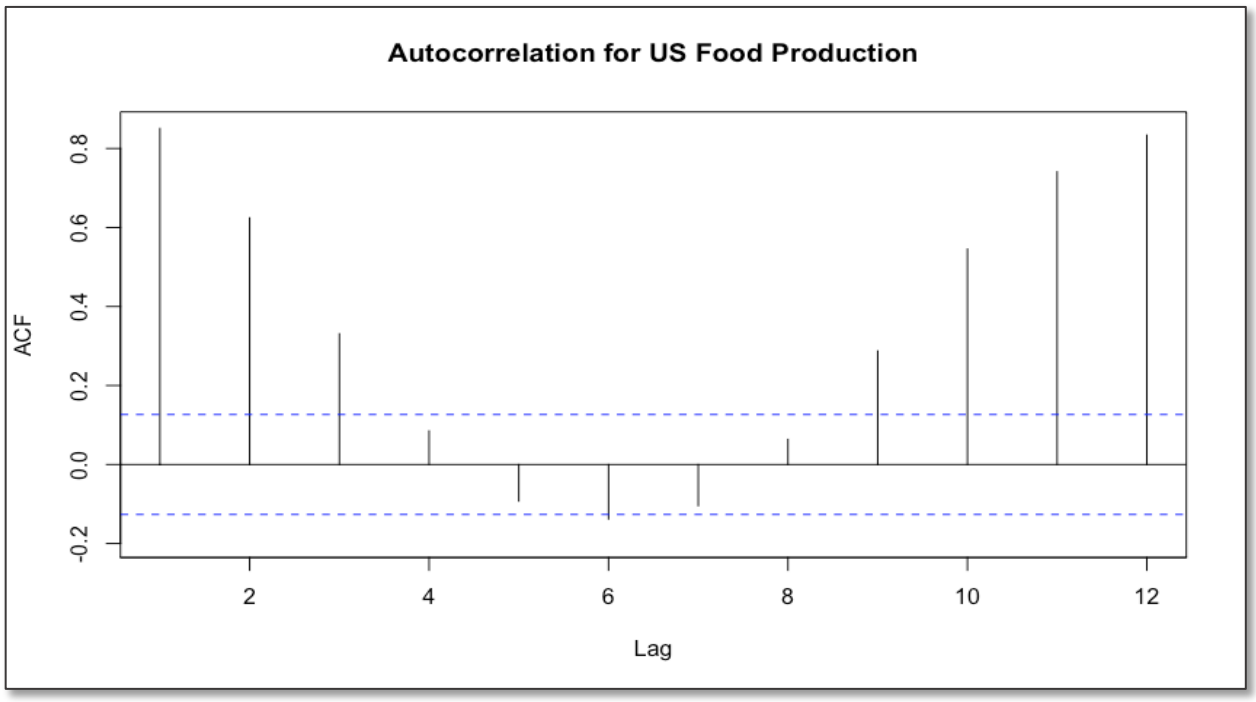
```
> prod.ts
         Jan     Feb     Mar     Apr     May     Jun     Jul     Aug     Sep     Oct     Nov     Dec
1997 740.516 733.954 713.174 657.021 651.785 679.875 658.036 736.048 769.976 866.834 904.676 910.473
1998 801.705 784.382 741.731 710.596 716.956 741.910 729.652 787.652 809.413 887.978 916.890 910.029
1999 785.656 767.660 716.863 711.317 721.790 725.126 721.630 753.023 780.746 881.167 881.934 900.691
2000 824.167 801.475 762.711 776.079 727.701 764.426 748.808 789.162 841.017 915.746 907.838 928.392
2001 820.447 815.836 793.809 767.703 755.911 771.964 759.624 801.039 826.721 840.213 861.649 867.856
2002 786.907 766.644 738.320 700.743 681.715 751.100 738.958 786.673 799.771 830.231 860.587 831.171
2003 756.549 744.155 744.411 703.828 720.270 747.396 688.421 757.335 781.872 837.912 888.812 862.243
2004 777.692 787.604 695.502 684.083 734.952 711.475 710.241 777.979 808.079 885.557 891.185 870.125
2005 833.780 829.478 717.910 714.858 746.006 717.485 737.694 768.526 801.361 918.121 912.905 936.500
2006 791.698 788.585 725.687 720.057 682.649 682.335 680.788 749.798 794.267 867.722 872.617 883.695
2007 814.822 779.764 753.819 665.585 656.417 635.411 611.685 674.351 737.287 776.275 792.868 807.964
2008 727.877 681.261 650.632 614.874 595.207 595.763 569.727 592.776 691.245 763.442 775.585 681.162
2009 601.733 595.597 572.684 533.527 537.150 532.556 550.946 596.037 677.043 827.783 783.601 780.064
2010 671.875 662.744 615.555 598.996 602.208 599.085 645.664 714.640 775.451 845.216 787.936 794.846
2011 689.838 686.433 662.529 652.783 611.522 600.324 599.902 615.050 628.545 781.536 781.954 769.441
2012 669.107 662.919 630.167 586.972 571.283 579.519 599.199 657.933 751.456 769.125 776.956 777.364
2013 716.677 682.896 686.949 638.013 614.170 614.532 618.664 671.738 706.258 785.435 814.029 824.894
2014 699.898 695.778 676.501 622.509 591.739 598.157 589.649 656.481 715.300 801.637 863.854 860.447
2015 735.949 728.953 713.015 676.449 647.782 674.915 688.269 775.777 775.733 848.340 833.438 804.789
2016 726.254 723.755 722.490 693.549 683.266 685.427 689.087 700.777 731.709 796.843 782.070 779.462
```

**b. Employ the *plot()* function to create a data plot of the historical data, provide it in your report, and explain what data patterns can be visualized in this plot.**



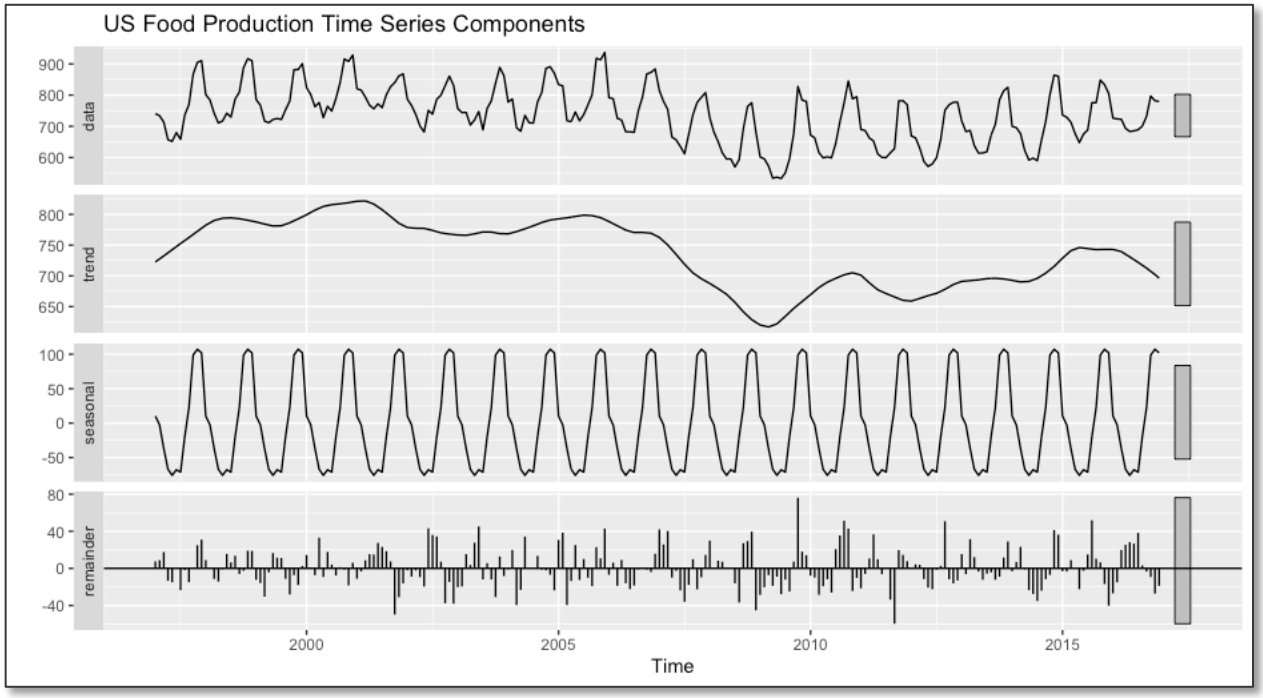

Observing the above plot showing the **historical data** for US food industry production for a 20-year period, we can infer that the trend for a period from 1997 to 2005 is almost constant, then it goes down till 2009 and from there on it increase slightly. Trend also appears to be a non-linear curve. There is one more pattern in this data, which is seasonality, as the pattern is consistent and repeats itself year after year.

**c. Apply the *Acf()* function to identify possible time series components. Provide in the report the autocorrelation chart and explain time series components existing in the historical data.**



The term autocorrelation refers to the degree of similarity between a given time series, and a lagged version of itself, over successive time intervals. We can use autocorrelation to understand strengths of the patterns of the dataset. We can observe here that the autocorrelation for 5 periods (k=4 to 8) is below the level of significance, which means that the autocorrelation is statistically insignificant for these lags. Autocorrelation coefficient is highest for a lag of 1 month and 12 months which shows **strong seasonality**. The table for autocorrelation coefficients for different values of k is shown below:

```
> data.frame(Lag, ACF)
   Lag    ACF
1    0  1.000
2    1  0.851
3    2  0.624
4    3  0.331
5    4  0.085
6    5 -0.093
7    6 -0.138
8    7 -0.104
9    8  0.064
10   9  0.288
11  10  0.546
12  11  0.741
13  12  0.834
```

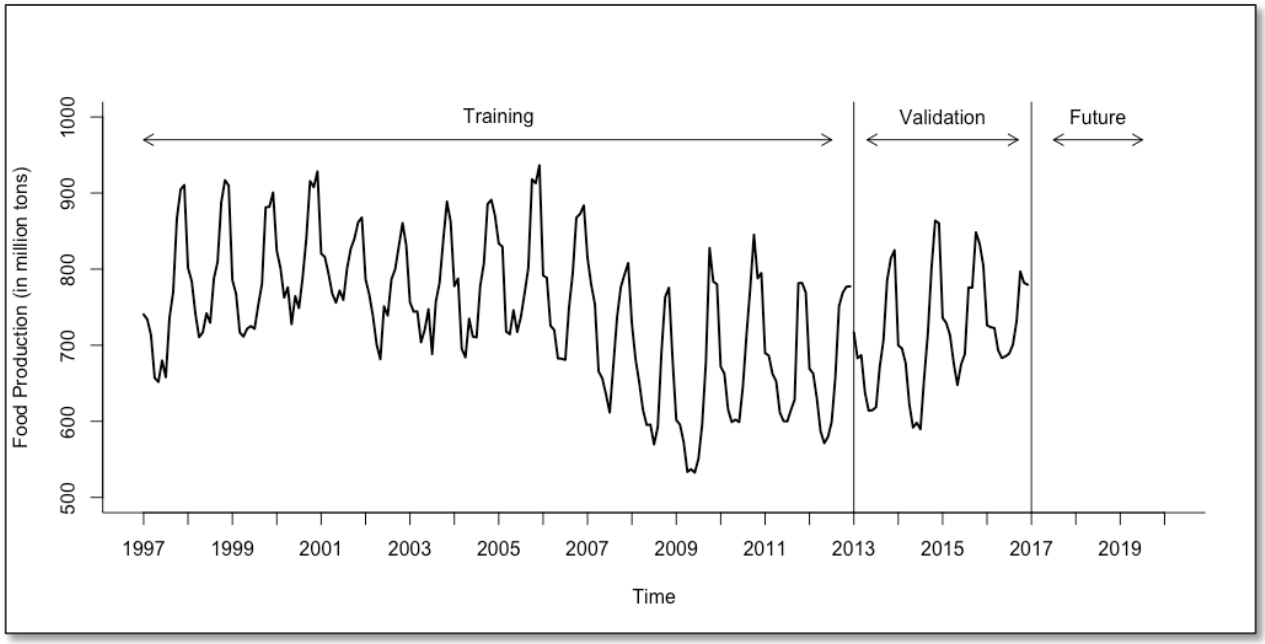


Looking at the time series components of the historical data, we can say that the **trend component is non-linear**, it is almost stagnant till 2006 and then it goes down till 2009 and from thereon it increases at a slow rate. **Seasonality is additive**, as there is no difference in peaks and valleys over the time. The remainder component consists of level and noise, it is very inconsistent in this case as it is associated with randomness.

**2. Use trailing MA for forecasting time series.**

**a. Develop data partition with the validation partition of 48 monthly periods (4 years) and training partition of 192 monthly periods (16 years).**

The dataset is partitioned in training data from January 1997 to December 2012 (192 months), and validation dataset from January 2013 to December 2016 (48 months).




**b. Use the *rollmean()* function to develop 4 trailing MAs with the window width of 2, 4, 6, and 12 for the training partition. Present the R code for these MAs in your report.**

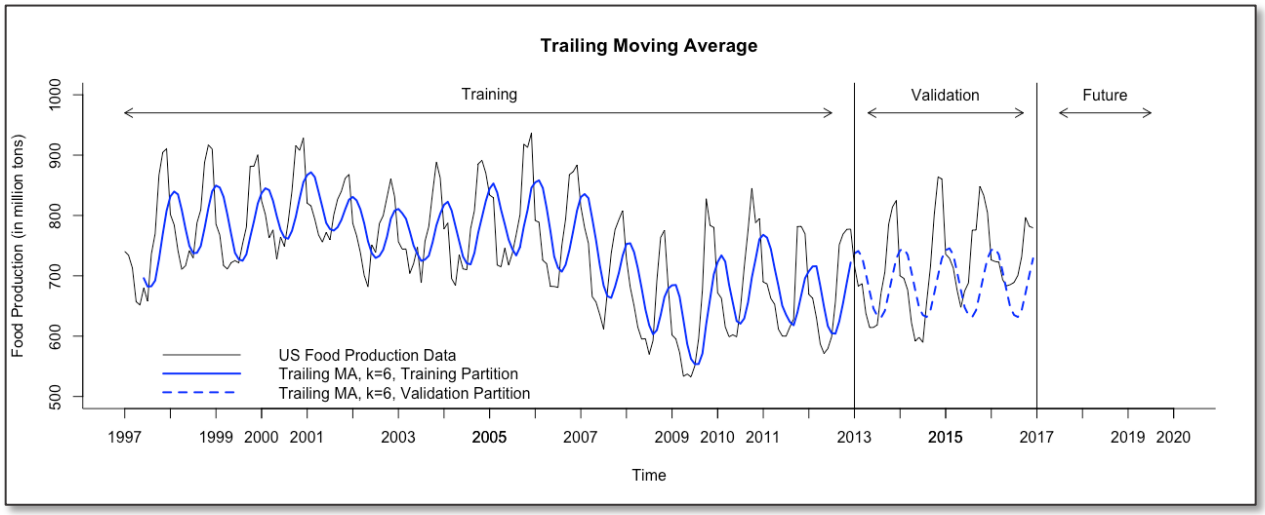
Developed trailing moving averages with k = 2, 4, 6 and 12 for training partition, using below code.

```
# # Create trailing MA with window widths (number of periods)  k = 2, 4, 6 and 12.

# In rollmean(), use argument align = "right" to calculate a trailing MA.
ma.trailing_2 <- rollmean(train.ts, k = 2, align = "right")
ma.trailing_4 <- rollmean(train.ts, k = 4, align = "right")
ma.trailing_6 <- rollmean(train.ts, k = 6, align = "right")
ma.trailing_12 <- rollmean(train.ts, k = 12, align = "right")
```

**c. Use the *forecast()* function to create a trailing MA forecast for each window width from question *2b* in the validation period, and present one of them, e.g., with window width of 4, in your report.**

The forecast for food production for the validation period is done, using trailing MAs for k = 2, 4, 6 and 12. The screenshot of the forecast for **k =6** is shown below.

```
> ma.trail_6.pred
         Point Forecast     Lo 0     Hi 0
Jan 2013       737.0483 737.0483 737.0483
Feb 2013       741.1209 741.1209 741.1209
Mar 2013       731.3749 731.3749 731.3749
Apr 2013       703.5890 703.5890 703.5890
May 2013       674.0837 674.0837 674.0837
Jun 2013       645.8985 645.8985 645.8985
Jul 2013       633.6051 633.6051 633.6051
Aug 2013       630.6496 630.6496 630.6496
Sep 2013       642.0224 642.0224 642.0224
Oct 2013       670.4772 670.4772 670.4772
Nov 2013       700.2750 700.2750 700.2750
Dec 2013       728.4208 728.4208 728.4208
Jan 2014       742.1806 742.1806 742.1806
Feb 2014       745.2268 745.2268 745.2268
```




**d. Apply the *accuracy()* function to compare accuracy of the 4 trailing MA forecasts in the validation period. Present the accuracy measures in your report, compare MAPE and RMSE of these forecasts, and identify the best trailing MA forecast.**

Accuracy Metrics for **Trailing MAs** with window **width = 2**

```
> round(accuracy(ma.trail_2.pred$mean, valid.ts), 3)
              ME    RMSE     MAE   MPE  MAPE ACF1 Theil's U
Test set 44.002 57.663 47.057 5.955 6.421 0.63     1.296
```

Accuracy Metrics for **Trailing MAs** with window **width = 4**

```
> round(accuracy(ma.trail_4.pred$mean, valid.ts), 3)
              ME    RMSE     MAE   MPE  MAPE  ACF1 Theil's U
Test set 25.421 62.367 51.503 3.055 6.971 0.733     1.363
```

Accuracy Metrics for **Trailing MAs** with window **width = 6**

```
> round(accuracy(ma.trail_6.pred$mean, valid.ts), 3)
              ME    RMSE     MAE   MPE  MAPE  ACF1 Theil's U
Test set 27.811 77.908 64.247 3.079 8.691 0.803     1.702
```

Accuracy Metrics for **Trailing MAs** with window **width = 12**

```
> round(accuracy(ma.trail_12.pred$mean, valid.ts), 3)
              ME    RMSE     MAE   MPE MAPE  ACF1 Theil's U
Test set 44.661 84.886 66.317 5.287 8.84 0.792     1.796
```

Looking at the accuracy metrics of 4 trailing MA forecasts in the validation period, we can see that the not just RMSE and MAPE is lowest for the model with window **width 2** but the ACF1 and Theil's U is also lowest for this model. Followed by the model with window width 4, 6 and the last is 12. So, we can conclude among the 4 models the trailing MA model with window width 2 performs the best on the validation data.

**3. Apply the two-level forecast with regression and trailing MA for residuals.**

**a. Develop using the *tslm()* function a regression model with linear trend and seasonality. Present the model summary in your report. Present and briefly explain the model equation in your report. Using this model, forecast food production in the validation period with the *forecast()* function. Present the forecast in your report.**

```
> summary(trend.seas)

Call:
tslm(formula = train.ts ~ trend + season)

Residuals:
     Min       1Q   Median       3Q      Max
-129.156  -23.961    1.195   36.033   96.882

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 829.96069   12.73749  65.159  < 2e-16 ***
trend        -0.81388    0.06004 -13.555  < 2e-16 ***
season2     -12.05350   16.27204  -0.741  0.45982
season3     -52.05131   16.27237  -3.199  0.00163 **
season4     -78.51756   16.27293  -4.825 2.98e-06 ***
season5     -83.04744   16.27370  -5.103 8.48e-07 ***
season6     -74.59431   16.27470  -4.583 8.56e-06 ***
season7     -79.07887   16.27592  -4.859 2.57e-06 ***
season8     -27.82456   16.27736  -1.709  0.08911 .
season9      17.75356   16.27902   1.091  0.27692
season10     94.85544   16.28090   5.826 2.58e-08 ***
season11    102.11525   16.28300   6.271 2.61e-09 ***
season12     97.55556   16.28533   5.990 1.12e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 46.02 on 179 degrees of freedom
Multiple R-squared:  0.7678,    Adjusted R-squared:  0.7522
F-statistic: 49.33 on 12 and 179 DF,  p-value: < 2.2e-16
```

Looking at the model summary of **regression model with linear trend and seasonality**. We can observe that the R-square or Coefficient of determination for this model is 77% which means that the model can explain the variability of the response data around its mean by 77%, which is quite good, it indicates how well the training data fits the model.

**Regression Model Equation is as follows:**

*yt = 829.961 – 0.814*t - 12.054*D2 - 52.051*D3 - 78.518*D4 - 83.047*D5 - 74.594*D6 - 79.079*D7 - 27.825*D8 + 17.754*D9 + 94.855*D10 + 102.115*D11 + 97.556*D12*

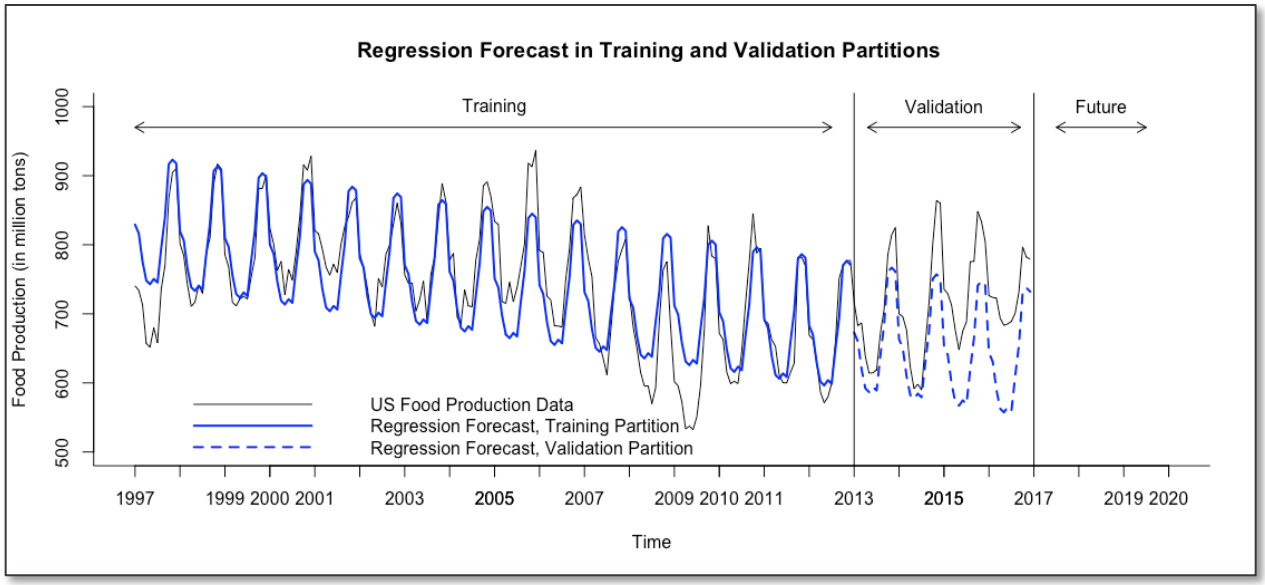
Here, yt is the US food production in million tons for a period 't'. t, on the right-hand side represents the trend and D2, D3…, D12 are dummy variables representing different seasons. So,

- D2 = 1, if it is February (Month =2), D2 = 0, if otherwise.
- D3 = 1, if it is March (Month =3), D2 = 0, if otherwise and so on.

Please note that, D1 is not defined here, as January can be defined by other 11 periods, because of the degree of freedom, we need 'n-1' dummy variables for 'n' periods.

The forecast for food production for the validation period is done, using **regression model with linear trend and seasonality**. The screenshot of the forecast is shown below.

```
> prod.lin.pred <- forecast(trend.seas, h = nValid, level = 0)
> prod.lin.pred
         Point Forecast     Lo 0     Hi 0
Jan 2013       672.8828 672.8828 672.8828
Feb 2013       660.0154 660.0154 660.0154
Mar 2013       619.2037 619.2037 619.2037
Apr 2013       591.9236 591.9236 591.9236
May 2013       586.5799 586.5799 586.5799
Jun 2013       594.2191 594.2191 594.2191
Jul 2013       588.9207 588.9207 588.9207
Aug 2013       639.3611 639.3611 639.3611
Sep 2013       684.1254 684.1254 684.1254
Oct 2013       760.4134 760.4134 760.4134
Nov 2013       766.8593 766.8593 766.8593
Dec 2013       761.4857 761.4857 761.4857
Jan 2014       663.1163 663.1163 663.1163
Feb 2014       650.2489 650.2489 650.2489
Mar 2014       609.4372 609.4372 609.4372
```

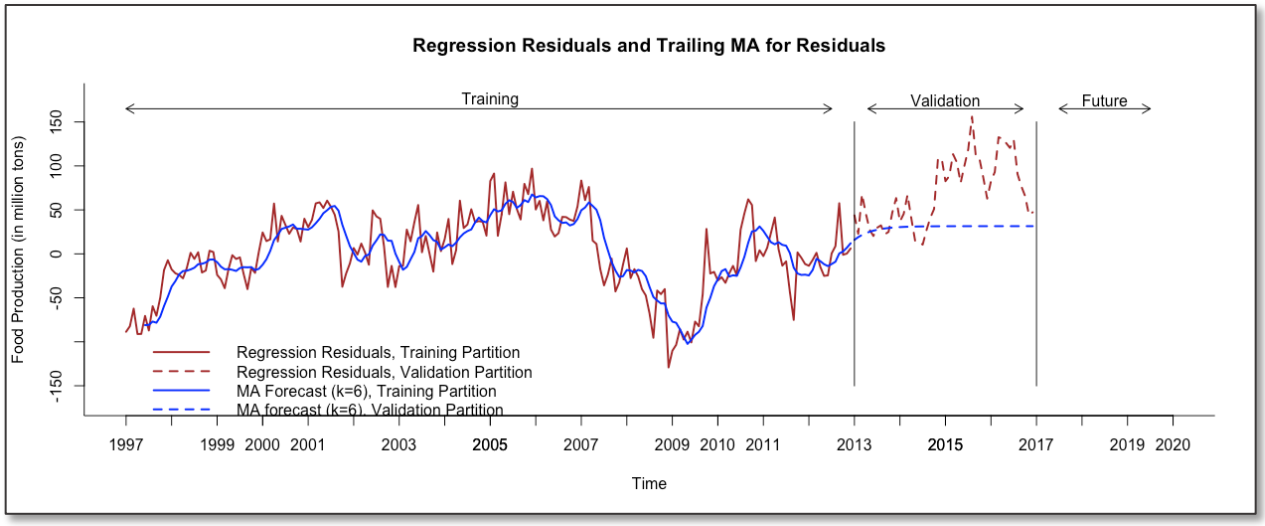



Looking at the plot we can observe that the model's prediction on validation data is underestimated, so, we should expect positive residuals.

**b. Identify regression residuals in the training period, apply a trailing MA (window width of 6) for these residuals using the *rollmean()* function, and identify trailing MA forecast of these residuals in the validation period (use the *forecast()* function).**

The residuals are the leftover between actual training data and the model that we utilized; it is basically the error term. We have smoothened the residuals using trailing MA with the window width of 6.

```
# Identify and display residuals for time series based on the regression
# (differences between actual and regression values in the same periods).
trend.seas.res <- trend.seas$residuals
trend.seas.res

# Apply trailing MA for residuals with window width k = 6.
ma.trail.res <- rollmean(trend.seas.res, k = 6, align = "right")
ma.trail.res

# Regression residuals in validation period.
trend.seas.res.valid <- valid.ts - prod.lin.pred$mean
trend.seas.res.valid

# Create residuals forecast for validation period.
ma.trail.res.pred <- forecast(ma.trail.res, h = nValid, level = 0)
ma.trail.res.pred
```




**c. Develop two-level forecast for the validation period by combining the regression forecast and trailing MA forecast for residuals. Present in your report a table that contains validation data, regression forecast, trailing MA forecast for residuals, and two-level (combined) forecast in the validation period. Apply the *accuracy()* function to compare accuracy of the regression model with linear trend and seasonality and the two-level (combined) model with regression and trailing MA for residuals. Present the accuracy measures in your report, compare MAPE and RMSE of these forecasts, and identify the best forecasting model for the validation period.**

Applied the model, two-level forecasting with regression and trailing MA for regression residuals on the validation partition and got the forecast for this period. The table below shows the actual food production for validation period, forecast for validation period using regression model, forecast for residuals using trailing MA with window width = 6 and combined forecast, which is the combination(sum) of the regression model and trailing MA forecasts.

```
> valid.df <- data.frame(valid.ts, prod.lin.pred$mean,
+                        ma.trail.res.pred$mean,
+                        fst.2level)
> names(valid.df) <- c("US Food Production", "Regression.Fst",
+                      "MA.Residuals.Fst", "Combined.Fst")
> valid.df
   US Food Production Regression.Fst MA.Residuals.Fst Combined.Fst
1             716.677       672.8828         15.87589     688.7587
2             682.896       660.0154         19.00076     679.0162
3             686.949       619.2037         21.50065     640.7044
4             638.013       591.9236         23.50056     615.4242
5             614.170       586.5799         25.10049     611.6804
6             614.532       594.2191         26.38043     620.5996
7             618.664       588.9207         27.40439     616.3251
8             671.738       639.3611         28.22355     667.5847
9             706.258       684.1254         28.87889     713.0043
10            785.435       760.4134         29.40315     789.8165
11            814.029       766.8593         29.82256     796.6819
12            824.894       761.4857         30.15809     791.6438
13            699.898       663.1163         30.42652     693.5428
14            695.778       650.2489         30.64126     680.8902
15            676.501       609.4372         30.81305     640.2503
16            622.509       582.1571         30.95048     613.1076
17            591.739       576.8134         31.06043     607.8738
18            598.157       584.4526         31.14838     615.6010
19            589.649       579.1542         31.21875     610.3729
20            656.481       629.5946         31.27504     660.8697
21            715.300       674.3589         31.32008     705.6789
22            801.637       750.6469         31.35610     782.0030
23            863.854       757.0928         31.38492     788.4777
24            860.447       751.7192         31.40798     783.1272
25            735.949       653.3498         31.42643     684.7762
```

Accuracy Metrics for **regression model** with linear trend and seasonality:

```
> round(accuracy(prod.lin.pred$mean, valid.ts), 3)
               ME    RMSE     MAE   MPE  MAPE  ACF1 Theil's U
Test set 70.074 79.917 70.074 9.665 9.665 0.835     1.782
```

Accuracy Metrics for **two-level (combined) model** with regression and trailing MA for residuals:

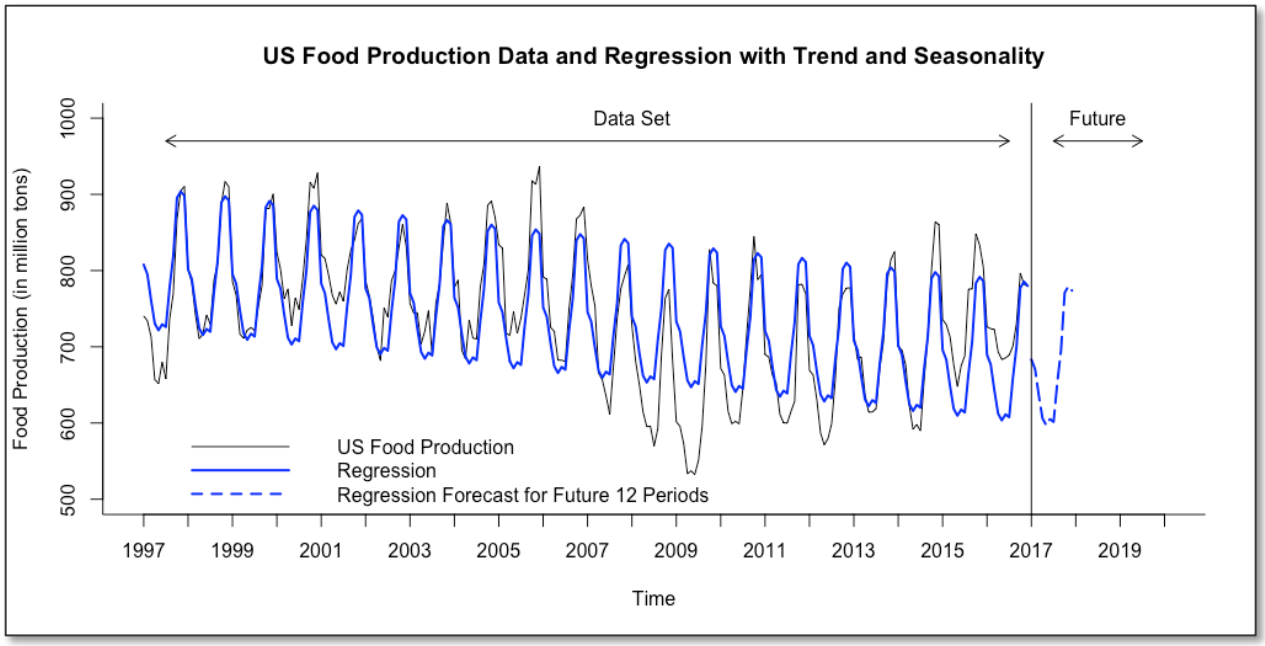
```
> round(accuracy(fst.2level, valid.ts), 3)
               ME    RMSE     MAE   MPE  MAPE  ACF1 Theil's U
Test set 40.201 54.796 43.363 5.461 5.975 0.828     1.227
```

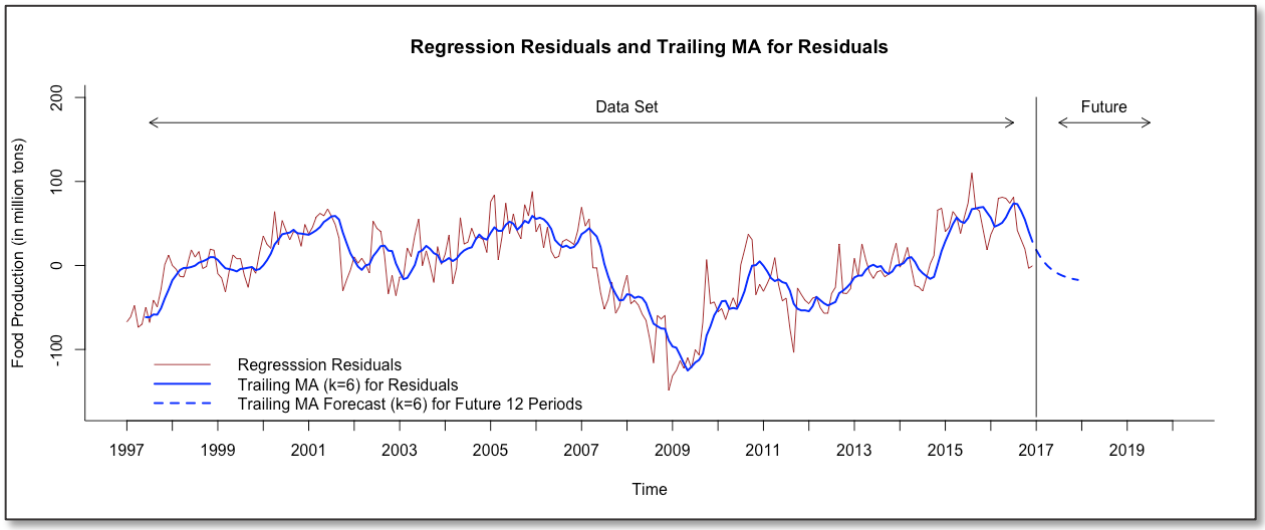
Looking at the accuracy metrics for both the models, we can see that the RMSE and MAPE for two-level (combined) model with regression and trailing MA for residuals is much lower compared to the just regression model with linear trend and seasonality. Therefore, we can conclude the two-level **(combined) model with regression and trailing MA for residuals** is best forecasting model among the given two models, for this dataset.

**d. For the entire data set, identify the regression model with linear trend and seasonality and trailing MA with the window width of 6 for the regression residuals. Use these models to forecast 12 months of 2017 and develop a two-level forecast for the 12 months as a combination of the specified forecasts. Present in your report a table that contains regression forecast, trailing MA forecast for residuals, and two-level (combined) forecast in the 12 months of 2017.**

Before attempting to forecast future values of time series, the training and validation periods must be recombined into one (entire) time series data set. The chosen model, two-level forecasting with regression and trailing MA for regression residuals, needs to be rerun on the entire (complete) data set. The table below shows the forecast for 12 months of 2017 using regression model, forecast for residuals using trailing MA with window width = 6 and combined forecast, which is the combination(sum) of the regression model and trailing MA forecasts.

```
> future12.df <- data.frame(tot.trend.seas.pred$mean, tot.ma.trail.res.pred$mean,
+                           tot.fst.2level)
> names(future12.df) <- c("Regression.Fst", "MA.Residuals.Fst", "Combined.Fst")
> future12.df
   Regression.Fst MA.Residuals.Fst Combined.Fst
1        683.3073       18.4995423     701.8068
2        670.6436       10.6902291     681.3338
3        636.3729        4.4427783     640.8156
4        606.1270       -0.5551825     605.5718
5        597.1739       -4.5535514     592.6203
6        605.0890       -7.7522466     597.3367
7        601.4821      -10.3112028     591.1709
8        652.7897      -12.3583679     640.4313
9        694.8124      -13.9961001     680.8163
10       771.0056      -15.3062858     755.6993
11       779.2191      -16.3544345     762.8647
12       773.7303      -17.1929534     756.5374
>
```



Looking at the charts above, we can say that the regression line fits the training data quite well.

**e. Develop a seasonal naïve forecast for the entire historical data set and apply the *accuracy()* function to compare accuracy of the three forecasting models: seasonal naïve forecast, regression model with trend and seasonality, and two-level (combined) model with regression and trailing MA for residuals. Present the accuracy measures in your report, compare MAPE and RMSE of these forecasts, and identify the best forecasting model for forecasting food production in 2017.**

Accuracy Metrics for **seasonal naïve forecast:**

```
> round(accuracy(prod.snaive.pred$fitted, prod.ts), 3)
             ME   RMSE    MAE    MPE  MAPE  ACF1 Theil's U
Test set -1.349 47.607 38.157 -0.386 5.368 0.748     1.024
```

Accuracy Metrics for **regression model** with linear trend and seasonality:

```
> round(accuracy(tot.trend.seas.pred$fitted, prod.ts), 3)
         ME   RMSE   MAE    MPE  MAPE  ACF1 Theil's U
Test set  0 47.325 37.74 -0.465 5.355 0.856     1.067
```

Accuracy Metrics for **two-level (combined) model** with regression and trailing MA for residuals:

```
> round(accuracy(tot.trend.seas.pred$fitted+tot.ma.trail.res, prod.ts), 3)
             ME   RMSE    MAE   MPE  MAPE  ACF1 Theil's U
Test set 0.796 24.071 18.612 0.008 2.541 0.509     0.508
```

Looking at the three-accuracy metrics, we can observe that the **two-level (combined) model** with regression and trailing MA for residuals performs exceptionally well compared to the other two models in terms of RMSE and MAPE. The ACF1 is also the least for this model compared to the other two. Now, comparing the Regression model and seasonal naïve model, they have almost equal performance on the entire dataset and have almost same RMSE and MAPE. Although Regression model is slightly better than seasonal naïve forecast in terms of RMSE and MAPE.

**4. Use advanced exponential smoothing methods.**

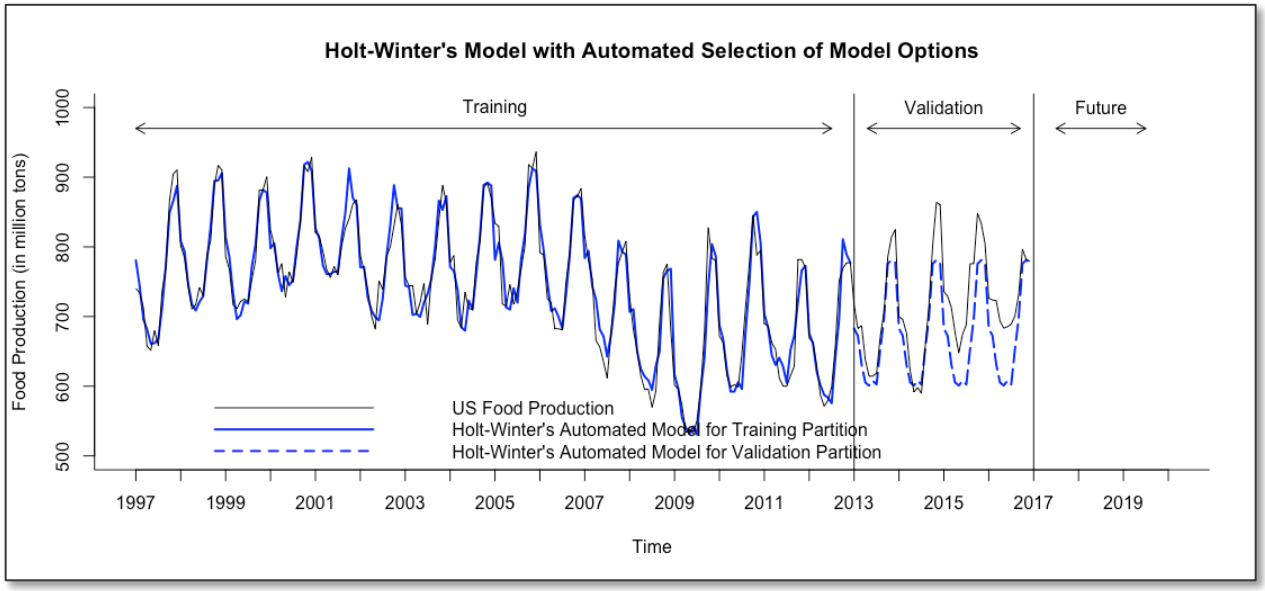
**a. For the training partition (from question *2a*), use the *ets()* function to develop a Holt-W_i_n_t_e_r_'s_ _(HW) model with automated selection of error, trend, and seasonality options, and automated selection of smoothing parameters for the training partition. Present and explain the model in your report. Use the model to forecast food production for the validation period using the *forecast()* function, and present this forecast in your report.**

```
> hw.ZZZ <- ets(train.ts, model = "ZZZ")
> hw.ZZZ
ETS(A,N,A)

Call:
 ets(y = train.ts, model = "ZZZ")

  Smoothing parameters:
    alpha = 0.6567
    gamma = 2e-04

  Initial states:
    l = 772.0586
    s = 103.3229 107.0921 101.7724 23.6764 -20.803 -71.5239
           -66.1615 -73.0104 -68.6515 -42.8149 -1.4497 8.5511

  sigma:  25.861

     AIC     AICc      BIC
2273.954 2276.681 2322.816
>
```

Holt Winters model is the most advanced exponential model, it captures the seasonal component along with the trend and level components in Holt's model. Looking at the summary of the model, we can see that the optimal model is **A, N, A model** for the training dataset; where first A means **Additive error**, second N means **No trend** and third A means **Additive seasonality**. The optimal value for smoothing constant for exponential smoothing (alpha) is 0.6567 and smoothing constant for trend estimate (beta) is 0.0002. The standard deviation of the residuals is 25.861, which shows a high variability in the residuals. The Akaike information criterion (AIC), which is an estimator of prediction error is 2273.954 for this model.

The forecast for food production for the validation period is done, using the above model. The screenshot of the forecast is shown below:

```
> hw.ZZZ.pred
         Point Forecast     Lo 0     Hi 0
Jan 2013       682.5498 682.5498 682.5498
Feb 2013       672.5418 672.5418 672.5418
Mar 2013       631.1837 631.1837 631.1837
Apr 2013       605.3440 605.3440 605.3440
May 2013       600.9852 600.9852 600.9852
Jun 2013       607.8391 607.8391 607.8391
Jul 2013       602.4765 602.4765 602.4765
Aug 2013       653.1963 653.1963 653.1963
Sep 2013       697.6769 697.6769 697.6769
Oct 2013       775.7675 775.7675 775.7675
Nov 2013       781.0935 781.0935 781.0935
Dec 2013       777.3189 777.3189 777.3189
Jan 2014       682.5498 682.5498 682.5498
Feb 2014       672.5418 672.5418 672.5418
Mar 2014       631.1837 631.1837 631.1837
Apr 2014       605.3440 605.3440 605.3440
May 2014       600.9852 600.9852 600.9852
Jun 2014       607.8391 607.8391 607.8391
Jul 2014       602.4765 602.4765 602.4765
Aug 2014       653.1963 653.1963 653.1963
Sep 2014       697.6769 697.6769 697.6769
Oct 2014       775.7675 775.7675 775.7675
Nov 2014       781.0935 781.0935 781.0935
```




Looking at the plot, we can observe that the model is **underestimating** for the later part of the validation period.

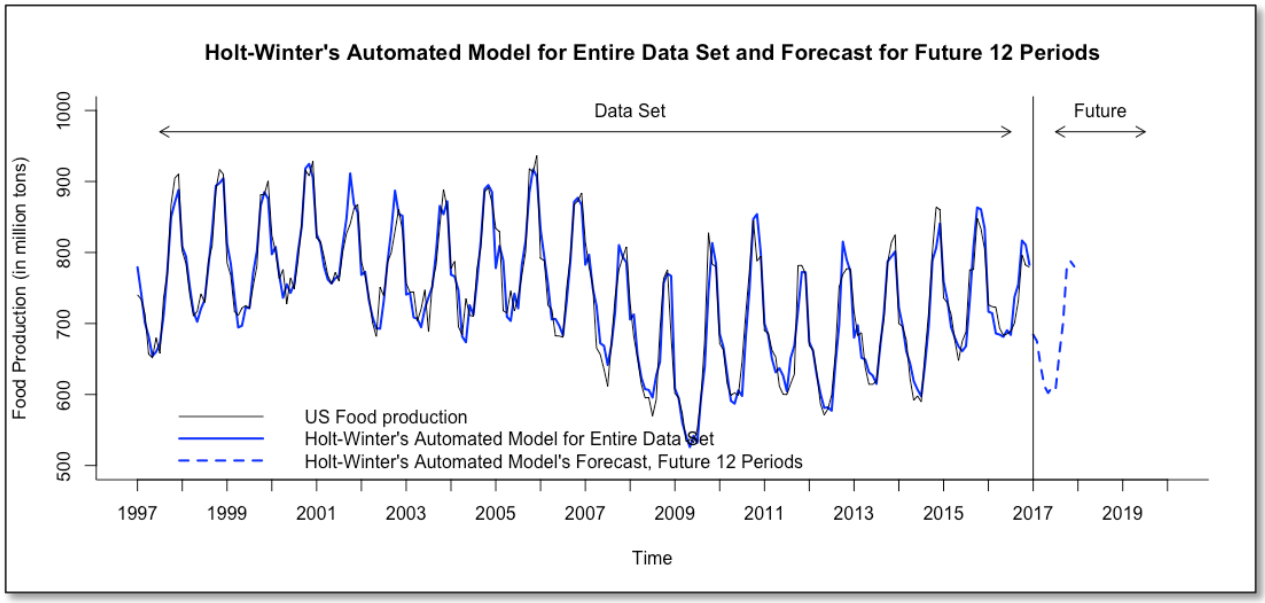
**b. To make a forecast for the 12 months of 2017, use the entire data set (no partitioning) to develop the HW model using the *ets() function* for the model with the automated selection of error, trend, and seasonality options, and automated selection of smoothing parameters. Also, use the ets() function for the model parameters (error, trend and seasonality) identified in question *4a* and automated smoothing parameters. Present and explain both models in your report. Use these HW models to forecast food production for the 12 months of 2017 using the *forecast()* function, and present both forecasts in your report.**

We've developed the Holt-Winter's using the entire dataset with the automated selection of error, trend, and seasonality options, and automated selection of smoothing parameters. Looking at the summary of the model, we can see that the optimal model is slightly different than the earlier one when we ran it on the training dataset. We got **M, N, A model** for the entire dataset; where first M means **Multiplicative error**, second N means **No trend** and third A means **Additive seasonality**. The optimal value for smoothing constant for exponential smoothing (alpha) is 0.723 and smoothing constant for trend estimate (beta) is 0.0002. The standard deviation of the residuals is 0.034, which shows a low variability in the residuals. The Akaike information criterion (AIC), which is an estimator of prediction error is 2874.911 for this model.

```
> HW.ZZZ <- ets(prod.ts, model = "ZZZ")
> HW.ZZZ
ETS(M,N,A)

Call:
 ets(y = prod.ts, model = "ZZZ")

  Smoothing parameters:
    alpha = 0.723
    gamma = 2e-04

  Initial states:
    l = 772.9382
    s = 102.4754 109.6773 101.4052 22.6539 -21.1445 -70.9292
           -67.995 -75.7587 -65.9667 -37.9767 -2.7437 6.3029

  sigma:  0.034

     AIC     AICc      BIC
2874.911 2877.054 2927.121
```

We have calculated the forecast for food production for the next 12 months, using **Holt Winter's M, N, A** model. The screenshot of the forecast is shown below:

```
> HW.ZZZ.pred <- forecast(HW.ZZZ, h = 12 , level = 0)
> HW.ZZZ.pred
         Point Forecast     Lo 0     Hi 0
Jan 2017       684.2106 684.2106 684.2106
Feb 2017       675.1422 675.1422 675.1422
Mar 2017       639.9247 639.9247 639.9247
Apr 2017       611.9238 611.9238 611.9238
May 2017       602.1433 602.1433 602.1433
Jun 2017       609.9070 609.9070 609.9070
Jul 2017       606.9691 606.9691 606.9691
Aug 2017       656.7629 656.7629 656.7629
Sep 2017       700.5485 700.5485 700.5485
Oct 2017       779.2925 779.2925 779.2925
Nov 2017       787.5742 787.5742 787.5742
Dec 2017       780.3833 780.3833 780.3833
```

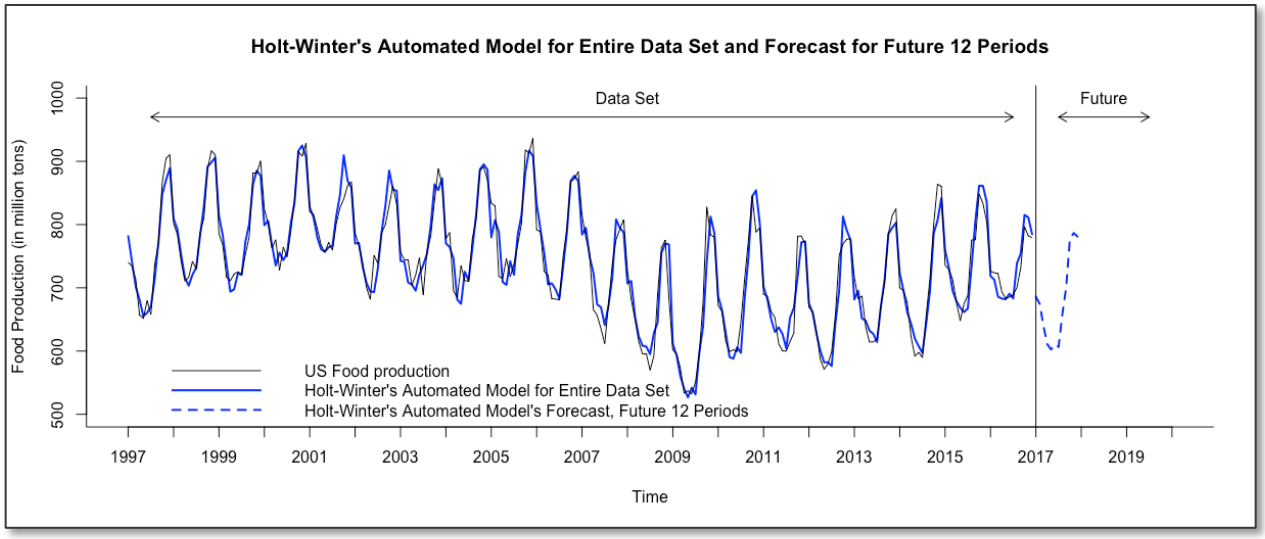



Just to see the comparison we developed the **Holt-Winter's A, N, A** model using the entire dataset with automated selection of smoothing parameters. The optimal value for smoothing constant for exponential smoothing (alpha) is 0.7113 and smoothing constant for trend estimate (beta) is 0.0001. The standard deviation of the residuals is 25.1086, which shows a high variability in the residuals. The Akaike information criterion (AIC), which is an estimator of prediction error is 2878.069 for this model. The AIC estimates the quality of each model, relative to each of the other models. So, looking at both the numbers it looks like the Holt-Winter's M, N, A model is slightly better than Holt-Winter's A, N, A for the entire dataset.

```
> HW.ANA <- ets(prod.ts, model = "ANA")
> HW.ANA # Model appears to be (A, N, A), with alpha = 0.7113 and gamma = 1e-04.
ETS(A,N,A)

Call:
 ets(y = prod.ts, model = "ANA")

  Smoothing parameters:
    alpha = 0.7113
    gamma = 1e-04

  Initial states:
    l = 773.5075
    s = 103.1196 108.7356 99.5974 22.5501 -19.7279 -71.4746
           -67.5386 -75.4483 -66.7639 -37.9505 -3.1279 8.0291

  sigma:  25.1086

     AIC     AICc      BIC
2878.069 2880.212 2930.279
```

We have calculated the forecast for food production for the next 12 months, using **Holt Winter's A, N, A** model. The screenshot of the forecast is shown below:

```
> HW.ANA.pred <- forecast(HW.ANA, h = 12 , level = 0)
> HW.ANA.pred
         Point Forecast     Lo 0     Hi 0
Jan 2017       685.9383 685.9383 685.9383
Feb 2017       674.7785 674.7785 674.7785
Mar 2017       639.9592 639.9592 639.9592
Apr 2017       611.1428 611.1428 611.1428
May 2017       602.4598 602.4598 602.4598
Jun 2017       610.3705 610.3705 610.3705
Jul 2017       606.4355 606.4355 606.4355
Aug 2017       658.1809 658.1809 658.1809
Sep 2017       700.4589 700.4589 700.4589
Oct 2017       777.5050 777.5050 777.5050
Nov 2017       786.6427 786.6427 786.6427
Dec 2017       781.0287 781.0287 781.0287
```




**c. Apply the *accuracy()* function to compare the three models: seasonal naïve forecast (applied in question *3e*) and two HW models developed in question *4b*. Present the accuracy measures in your report, compare MAPE and RMSE of these forecasts, and identify the best forecasting model.**

Accuracy Metrics for **HW model with the automated selection of error, trend, and seasonality options, and automated selection of smoothing parameters**

```
> round(accuracy(HW.ZZZ.pred$fitted, prod.ts), 3)
               ME    RMSE     MAE     MPE  MAPE  ACF1 Theil's U
Test set -0.548 24.407 18.442 -0.146 2.513 0.018     0.509
```

Accuracy Metrics for **HW ANA model:**

```
> round(accuracy(HW.ANA.pred$fitted, prod.ts), 3)
              ME    RMSE     MAE     MPE  MAPE  ACF1 Theil's U
Test set -0.56 24.365 18.354 -0.148 2.504 0.029     0.509
```

Accuracy Metrics for **seasonal naïve forecast:**

```
> round(accuracy((snaive(prod.ts))$fitted, prod.ts), 3)
               ME    RMSE     MAE     MPE  MAPE  ACF1 Theil's U
Test set -1.349 47.607 38.157 -0.386 5.368 0.748     1.024
```

Looking at the accuracy metrics for the entire dataset, we can see that we are doing much better than the Seasonal Naïve forecast for both the Holt winter's models. It's almost twice better forecast as far as **RMSE** and **MAPE** is concerned. Now comparing the two Holt-Winter's model; HW A, N, A is **marginally better** than Holt-Winter's model with the automated selection of error, trend, and seasonality options, and automated selection of smoothing parameters i.e. (M, N, A) as the RMSE and MAPE is slightly lower in Holt-Winter's A, N, A model than Holt-Winter's M, N, A. Also, it is important to note here that autocorrelation for residuals (**ACF1**) is pretty high in case of seasonal naïve forecast, which is not a good thing however in both the Holt-winter's models the ACF1 is taken care of, which is extremely important.

**d. Compare the best forecasts identified in questions *3e* and *4c*. Explain what your final choice of the forecasting model in this case will be.**

We've chosen the best model based on the performance, it was the **two-level (combined) model** with regression and trailing MA for residuals for ques 3e and **Holt-Winter's model A, N, A** model for ques 4c

Accuracy Metrics for **two-level (combined) model** with regression and trailing MA for residuals:

```
> round(accuracy(tot.trend.seas.pred$fitted+tot.ma.trail.res, prod.ts), 3)
             ME   RMSE    MAE   MPE  MAPE  ACF1 Theil's U
Test set 0.796 24.071 18.612 0.008 2.541 0.509     0.508
```

Accuracy Metrics for **HW model with the automated selection of error, trend, and seasonality options, and automated selection of smoothing parameters**

```
> round(accuracy(HW.ZZZ.pred$fitted, prod.ts), 3)
              ME   RMSE    MAE    MPE  MAPE  ACF1 Theil's U
Test set -0.548 24.407 18.442 -0.146 2.513 0.018     0.509
```

Looking at the accuracy metrics, we can observe that the RMSE is marginally better for two-level (combined) model however MAPE is slightly better in holt-Winter's A, N, A model. Now looking at the ACF1 which is autocorrelation for residuals, it is quite high in two-level (combined) model, which is taken care in holt-Winter's A, N, A model, which is extremely important. Therefore, we would recommend **Holt-winter's A, N, A model** for forecasting the future US food production for next 12 months of 2017.