Population Time Series from the US Census Bureau

By: Jackson Stroup, Jack Michalowski, and Liam OConnor

MA 362 10:00 AM

11/30/2020

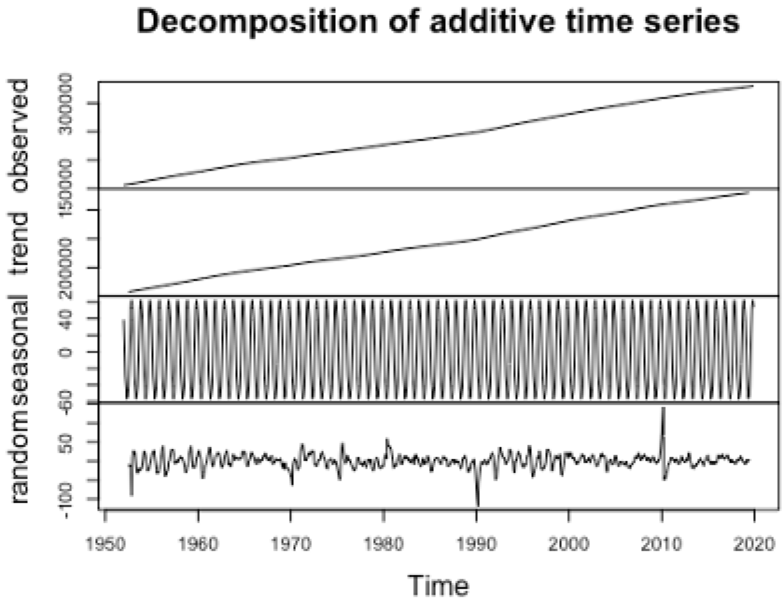
Table of Contents

Dataset Background/Description:

This data set we found is from the US Census Bureau. The data set is in time series as it shows the population from a census from 1952-2019. For our dataset we wanted to find something unique! This is a dataset from kaggle, representing the U.S. Census Bureau hosted by the Federal Reserve Economic Database (FRED). With this data, we will ultimately be able to decompose it, as well as use the data to perform different components of time series such as exponential smoothing, Holt Winters and TS regression approaches for forecasting for another year. In addition we will measure and compare the accuracy of forecasting using MAPE, RMSE, and MAD. Finally, at the end we will determine if our time series is indeed stationary or not! For our dataset we used all but the last cycle for our train data set, and used the last cycle as our test data set.

Decomposing the Time Series

First, let's create and decompose the time series to test for the different parts.

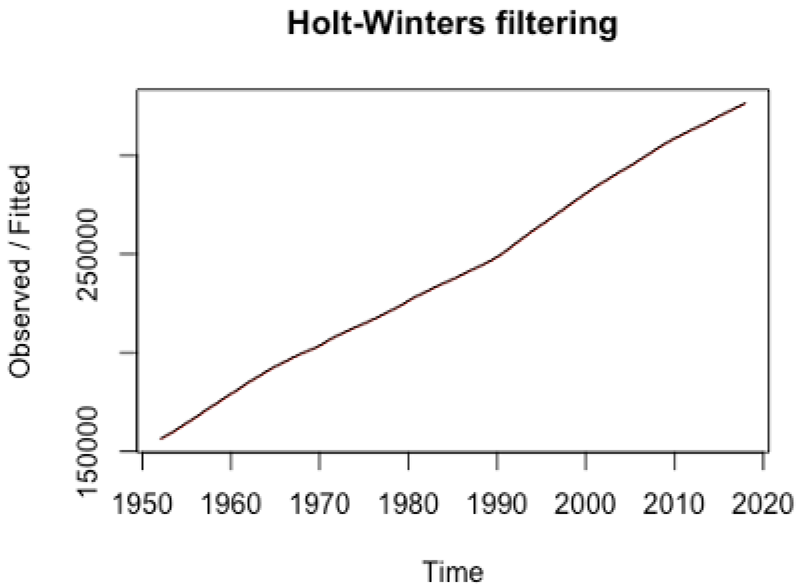


Shown above is the decomposition of the time series. From the graph, it is clear that there is an increasing secular trend. Based on the decomposition, it is also clear that there is some seasonality in the model.

Forecasting the model

Forecasting using Exponential Smoothing

Next, let's apply exponential smoothing to the model, and forecast for the next year. For the exponential smoothing constant, we will use 0.70.



Above is the output. It is important to note that this output does not take into account seasonality in the dataset. Now, let forecast the model using Exponential smoothing.

	Point Forecast <dbl>	Lo 80 <dbl>	Hi 80 <dbl>	Lo 95 <dbl>	Hi 95 <dbl>
Feb 2019	328402.2	328325.5	328479.0	328284.8	328519.7
Mar 2019	328402.2	328308.5	328496.0	328258.9	328545.6
Apr 2019	328402.2	328294.2	328510.3	328237.0	328567.5
May 2019	328402.2	328281.6	328522.9	328217.7	328586.8
Jun 2019	328402.2	328270.1	328534.4	328200.2	328604.3
Jul 2019	328402.2	328259.6	328544.9	328184.1	328620.4
Aug 2019	328402.2	328249.8	328554.7	328169.1	328635.4
Sep 2019	328402.2	328240.6	328563.9	328155.0	328649.4
Oct 2019	328402.2	328231.9	328572.6	328141.7	328662.8
Nov 2019	328402.2	328223.6	328580.9	328129.1	328675.4

Above is the forecasting estimate for selected months of 2019. It is important to note once again that every value will be the same, since seasonality is not accounted for.

HoltWinters Forecasting

Next, let's forecast for 2019 using HoltWinters exponential smoothing. This will account for seasonality in the model.

	Point Forecast <dbl>	Lo 80 <dbl>	Hi 80 <dbl>	Lo 95 <dbl>	Hi 95 <dbl>
Feb 2019	328053.1	327761.6	328344.6	327607.3	328498.9
Mar 2019	327948.6	327525.9	328371.3	327302.1	328595.0
Apr 2019	328074.3	327486.7	328661.8	327175.7	328972.9
May 2019	328508.6	327730.5	329286.7	327318.6	329698.7
Jun 2019	329259.4	328269.3	330249.4	327745.2	330773.6
Jul 2019	330248.4	329027.6	331469.2	328381.4	332115.5
Aug 2019	331335.5	329867.0	332804.1	329089.6	333581.5
Sep 2019	332342.1	330610.1	334074.1	329693.3	334991.0
Oct 2019	333121.5	331111.3	335131.7	330047.2	336195.8
Nov 2019	333570.5	331268.2	335872.7	330049.5	337091.5

Above is the forecasting estimate for selected months of 2019. Note that each value is different, since seasonality is accounted for in this model. It looks like there will be a cyclical effect for the first few months of 2019, and then beginning in April, the model will begin increasing each month.

Time series Regression Forecasting

Next, let's create a regression model for the time series. Since seasonality is evident in the model, we will include it as well.

```
Call:
lm(formula = value ~ t + as.factor(month), data = popm)

Coefficients:
       (Intercept)                  t   as.factor(month)2   as.factor(month)3
as.factor(month)4
         156421.17             214.13              -31.68              -66.19
-79.46
 as.factor(month)5   as.factor(month)6   as.factor(month)7   as.factor(month)8
as.factor(month)9
            -93.95             -89.23              -80.03              -51.76
-16.19
as.factor(month)10  as.factor(month)11  as.factor(month)12
             16.10              24.61               15.37
```

Above is the regression model output from R. Let's put it in a chart to pretty it up.

Intercept	t	Febuary	March	April	May	
156421.17	214.13	-31.68	-66.19	-79.46	-95.95	
June	July	August	September	October	November	December
-89.23	-80.03	-51.76	-16.19	16.1	24.61	15.37

It is important to note that January is the base case, therefore it is not included in the regression output. Let's predict for 2019 using Time series regression.

January	Febuary	March	April	May	June
328797.7	328979.9	329159.6	329360.4	329560.1	329778.9
July	August	September	October	November	December
330002.3	330244.7	330494.4	330740.8	330963.4	331168.3

Above are the predictions for 2019. Note the change from month to month due to seasonality.

Measuring Forecasting Accuracy

Measuring Forecasting Accuracy for Exponential Smoothing

```
[1] 2348.082
[1] 0.7123806
[1] 1684.869
```

Shown above is the R output for the MAD, MAPE, and RMSE calculations to measure the forecasting accuracy for exponential smoothing. MAD = 2348.082, MAPE = 0.7123806, RMSE = 1684.869.

Measuring Forecasting Accuracy for Holt Winters

```
[1] 3615.196
[1] 1.096557
[1] 2903.739
```

Shown above is the R output for the MAD, MAPE, and RMSE calculations to measure the forecasting accuracy for Holt Winters. MAD = 3615.196, MAPE = 1.096557, RMSE = 2903.739.

Measuring Forecasting Accuracy for Time Series Regression

```
[1] 1764.604
[1] 0.5355129
[1] 1058.975
```

Shown above is the R output for the MAD, MAPE, and RMSE calculations to measure the forecasting accuracy for Holt Winters. MAD = 1764.604, MAPE = 0.5355129, RMSE = 1058.975.

Comparing the Forecasting Accuracy Measurements

Based on the outputs, the most accurate forecasting model appears to be the Time Series Regression forecast, since it has the lowest MAD and RMSE.

Testing for a Stationary Time Series

```
	Augmented Dickey-Fuller Test

data:  TSpop
Dickey-Fuller = -2.2811, Lag order = 9, p-value = 0.4593
alternative hypothesis: stationary
```

An Augmented Dickey-Fuller test was run on the time series data in R. As we can see from the R output, the p-value of 0.4593 is greater than 0.05, so we fail to reject the null hypothesis. Hence, the data do not provide evidence to say that the time series is stationary.