A New Method for Modeling Univariate Time Series with a Hybrid Approach Using STL Decomposition

Mertcan Yükselsin
The Department of Statistics
Ankara, Turkey
mertcan.yukselsin@metu.edu.tr

***Abstract*— This paper introduces a new univariate time series modeling approach for the data Unemployment Rate starting from 1980 to 2019. The new method uses nine different models while constructing the final model. Throughout the analyses, R-Studio is used.**

Keywords—statistics, univariate time series analysis, decomposition, modeling, forecasting, decomposition

I. Introduction

Modeling uncertainty is a certain element (Perling, et al., 2021). In this perspective, throughout time, many modeling strategies are constructed to forecast the future and still, many companies and researchers are trying to understand such time series' behaviors. Therefore, in this paper, a new modeling approach is introduced for the dataset coming from U.S. Bureau of Labor Statistics and is about the unemployment rate of the labor force in percentage in U.S. It is a seasonally adjusted monthly data starting from 1980 to 2019 December. Yet, when the dataset is investigated, the seasonal effects are still present in the dataset.

In this paper, while the new method is being introduced, mainly R programming language is used, and thanks to Rob Hyndman, one of its famous package called 'forecast' is used.

The new approach we present is a combination of nine different models and their combination. It is not surprising that people are trying to understand the time series separating its decompositions and then focusing on maybe two or three parts manually. However, we automatically use different models with this approach and let the data decide to fit the best model for its own past observations with different models.

A. Abbreviations and Acronyms

Before starting to talk about the new technique which can be referred to as the 'New Hybrid Approach,' defining such abbreviations and acronyms would be beneficial for the reader. The following abbreviations and acronyms for the accuracy metrics and the definitions come from Hyndman, R.J. and Athanasopoulos, G. (2018) 'Forecasting: Principles and Practice' book.

STL: Seasonal Trend Decomposition using Loess

ARIMA: Autoregressive Integrated Moving Average

ETS: Exponential Smoothing State Space Model

TBATS: Exponential Smoothing State Space Model with Box-Cox Transformation, ARMA errors, Trend and Seasonal Components

NNETAR: Neural Network Models

ARFIMA: Autoregressive Fractionally Integrated Moving Average

STLM: STL Modeling

Holt: Exponential Smoothing Forecast

ME: Mean Error

RMSE: Root Mean Squared Error

MAE: Mean Absolute Error

MPE: Mean Percentage Error

MAPE: Mean Absolute Percentage Error

MASE: Mean Absolute Scaled Error

ACF1: Autocorrelation of Errors at Lag 1

Theil's U: Uncertainty Coefficient

II. Application of the Process

In this part, the process of the new modeling approach is discussed using a flowchart.

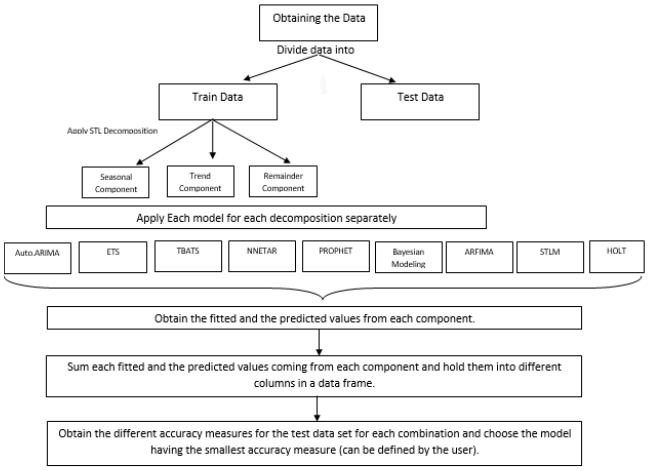

Figure 1: *Flowchart for the Process*

The process goes as follows; firstly, the data is obtained and then it is divided into two parts, namely test and train datasets. After that, train data is decomposed using STL technique with s.window as 'periodic'. After that, for each component, fit nine different models. Then, because the STL decomposition technique is working straight-forward, for each combination (there are actually 9 different models are fitted for each component) add the possible fitted and predicted values for the test dataset. For example, one would be like this: fitting ARIMA model for the seasonal component coming from auto.arima function, fitting ETS model for the trend component and TBATS model for the remainder component. Similarly, each model is fitted for each component and then the original values are obtained for adding them up for each combination. In the end, for the prediction on test dataset, we would have $9^3=729$ combination values. After that, we compare the accuracy results with the test dataset and one can pick the best model based on one or more accuracy metrics. Of course, it shouldn't be forgetten that we directly used the automated models for each modeling case.

III. Data Preparation

Assuming that the flowchart is valid, the data is first to read.

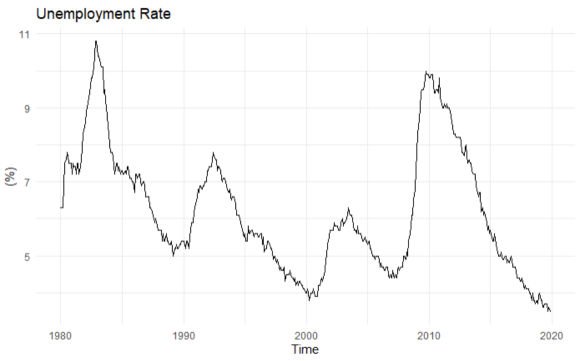

Figure 2: *Time Series Plot of Unemployment Rate*

When Figure 2 is investigated, it is clearly seen that although the data is assumed to be seasonally adjusted, we have some seasonal behavior over time. Also, it can be seen that the data is probably not stationary since there is a clear trend overtime again. In traditional methods, firstly, data should be made stationary to consider a statistical time series model. However, since the paper is mainly focused on introducing the new method, those steps are skipped but they were investigated in another paper more in detail.

Now that we have some idea about this time series' behavior let's divide the data into train and test dataset as it is the initial step for this approach to test the best model. That's why, test data contain observations starting from 2019, which means it has 12 observations in the test data and 468 observations in the train data.

IV. Modeling

In this part, firstly, using our flowchart again, we used STL Decomposition technique, and its corresponding outputs are given below.

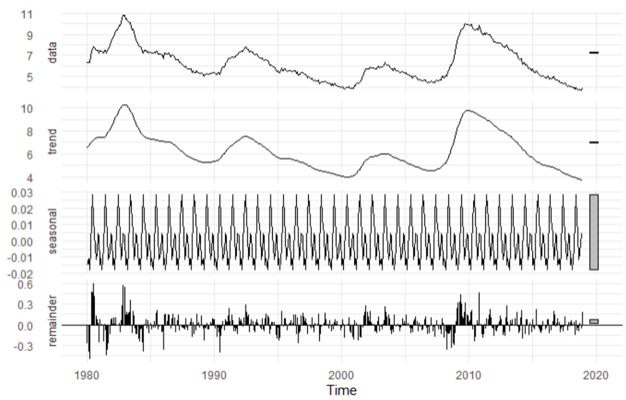

Figure 3: *STL Decomposition of the Train Data*

It is seen that after using STL Decomposition, we would have three components, namely Trend, Seasonal, and Remainder. Starting from this part, for each element, 9 different models, are used as they have stated in the flowchart. Again, we have to note that the automated models are used in R programming, precisely forecast package while getting fitted and predicted values.

To illustrate this method, let's show the results with one example. Assume that for each component we fit models coming from auto.arima function outputs.

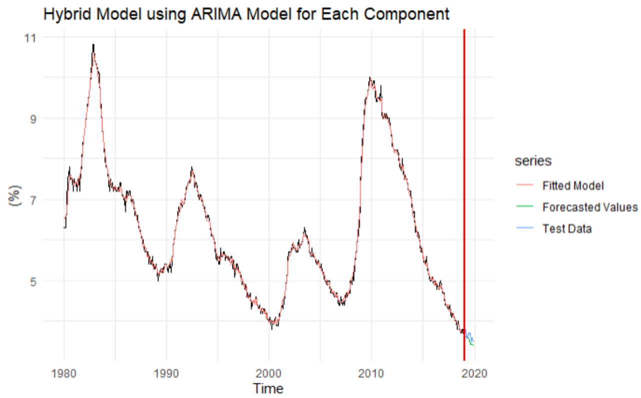

Figure 4: *Hybrid Model Time Series Plot*

To compare the results, let's fit another model for the training dataset directly as a traditional approach. Again, we used auto.arima function to fit the training data.

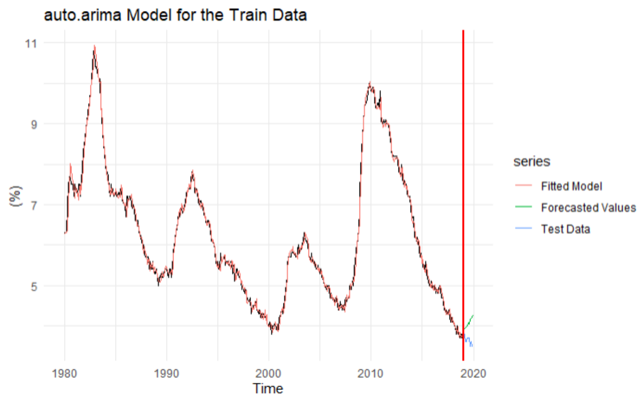

Figure 5: *auto.arima Function Time Series Plot*

Visually, we can see that the hybrid model works well compared to the auto.arima model for the train data. Moreover, let's use the accuracy metrics to show the results for the test dataset.

	ME	RMSE	MAE	MPE	MAPE	ACF1	Theil's U
Model 1	0.124	0.154	0.129	3.346	3.481	0.061	1.181
Model 2	-0.378	0.457	0.401	-10.595	11.175	0.599	4.272

Table 1: *Model Comparison Table*

The table above show the test accuracy metrics for the two models. As we can see that, overall, the first model, which is the hybrid modeling approach, works well compared to modeling train data directly.

Because the process can be exhausted to obtain models manually, another good job that this approach does is to letting the data itself to find the best model and finding the results automatically. That's why, a grid search technique is used to obtain the models one by one. For simplicity, each model name is stated numerically. For instance, 1 represents the ARIMA model, 2 represents the ETS model, and so on (as it is in the same order in the flowchart). After obtaining the 927 fitted models and predicted values for the test dataset, each model's combinations are obtained. Based on the accuracy metrics in the forecast package's accuracy function, seven different models having the smallest corresponding accuracy metrics result from suggestions are obtained as follows.

Minimum	Models	ME	RMSE	MAE	MPE	MAPE	ACF1	Theil's U
ME	S:9, T:7, R: 9	-0.947	1.011	0.947	-26.220	26.22	0.62	9.405
RMSE	S:7, T:8, R: 8	0.030	0.084	0.071	0.779	1.934	0.032	0.59
MAE	S:7, T:8, R: 8	0.030	0.084	0.071	0.779	1.934	0.032	0.59
MPE	S:9, T:7, R: 9	-0.947	1.011	0.947	-26.220	26.22	0.62	9.405
MAPE	S:7, T:8, R: 8	0.030	0.084	0.071	0.779	1.934	0.032	0.59
ACF1	S:7, T:6, R:6	0.142	0.163	0.142	3.831	8.831	-0.077	1.271
Theil's U	S:5, T:8, R:8	0.030	0.086	0.772	0.778	1.942	0.057	0.578

Table 2: *Hybrid Model Model Comparison Table*

The table is obtained using the minimum values coming from the accuracy metrics in the accuracy function. Note that S represents Seasonal, T as Trend, R as the Remainder component. The first column can also tell us that; for example, minimum RMSE for the predicted values in the test dataset can be obtained modeling Seasonal component with ARFIMA, Trend and Remainder component with (separately) STLM models. The user can also focus on the general results where the minimum criteria can be obtained mostly using the model coded as 7,8,8 (which is the same as having the minimum RMSE, MAE, MAPE).

The selected model visually represented as below.

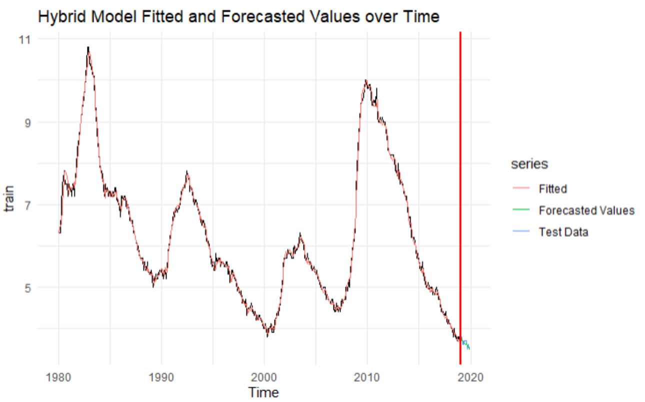

Figure 6: *The Final Best Hybrid Model Time Series Plot*

We can see that the fitted model is quite well and the test data prediction as well.

V. Discussion and Conclusion

It is known that focusing on a time series is a critical issue nowadays that it can be used to make a profit or find such catastrophic events before it has happened.

In this paper, we introduced a new model for the time series analysis. To start with the investigation, we used STL Decomposition technique to obtain three compositions. Then, we modeled each component with nine models individually, namely ARIMA, ETS, TBATS, NNETAR, Prophet, Bayesian, ARFIMA, STLM, and Holt. The aim is to obtain different combinations for each component and add them up to get the original series' fitted and predicted values. Note that their corresponding automated selected models are used for each model technique, which means we did not specify the orders for ARIMA or the parameters for ETS and so on. For future studies, maybe one can focus on composed parts individually and manually so that the researcher can be sure that the selected models are appropriate for the corresponding specified part. Also, in this study, we did not focus on AIC/BIC value criteria. One can also use those information while selecting the best model as well.

Another significant part is using how many models in the procedure. In our study, we used nine models, which result in 9^3 combinations in general. If we generalize this, using n many models will result in n^3 combinations and may take too much time to find the best model. However, this part should be tried because nowadays, many models are being constructed. In the end, this paper's aim is also introducing such a new model. Therefore, assuming that the user wants to consider different models rather than used models here, the codes are free to be adjusted.

References

U.S. Bureau of Labor Statistics, Unemployment Rate [UNRATE], retrieved from FRED, Federal Reserve Bank of St. Louis; https://fred.stlouisfed.org/series/UNRATE, December 24, 2020

Perlin, Marcelo & Mastella, Mauro & Vancin, Daniel & Ramos, Henrique. (2021). A GARCH Tutorial with R. Revista de Administração Contemporânea. 25. 10. 1590/1982-7849rac2021200088

Hyndman, R.J. and Koehler, A.B. (2006) "Another look at measures of forecast accuracy". International Journal of Forecasting, 22(4), 679-688. Hyndman, R.J. and Athanasopoulos, G. (2018) "Forecasting: principles and practice", 2nd ed., OTexts, Melbourne, Australia. Section 3.4 "Evaluating forecast accuracy". https://otexts.com/fpp2/accuracy.html