Forecasting Unemployment Rate in USA

Operations research & industrial engineering | THE UNIVERSITY OF TEXAS AT AUSTIN

BY:

harshit mehta;PRateek agrawal;prateek punj

2017

**TABLE OF CONTENTS**

Objective .............................................................................................................................................................. 1

Introduction ........................................................................................................................................................ 1

Methodology ....................................................................................................................................................... 2

Stationary Model .............................................................................................................................................. 2

Stochastic trend and Seasonality ............................................................................................................... 3

Empirically checking for stochastic trend & Seasonality ................................................................ 4

Non-Stationary Model .................................................................................................................................... 4

F- criteria for deterministic part ............................................................................................................... 5

Auto correlation curve of non–stationary model ............................................................................... 6

Prediction using Stationary Model ........................................................................................................... 7

Prediction using Non-Stationary Model ................................................................................................. 7

ARMAV Model..................................................................................................................................................... 8

Prediction Using ARMAV................................................................................................................................ 8

Conclusion............................................................................................................................................................ 9

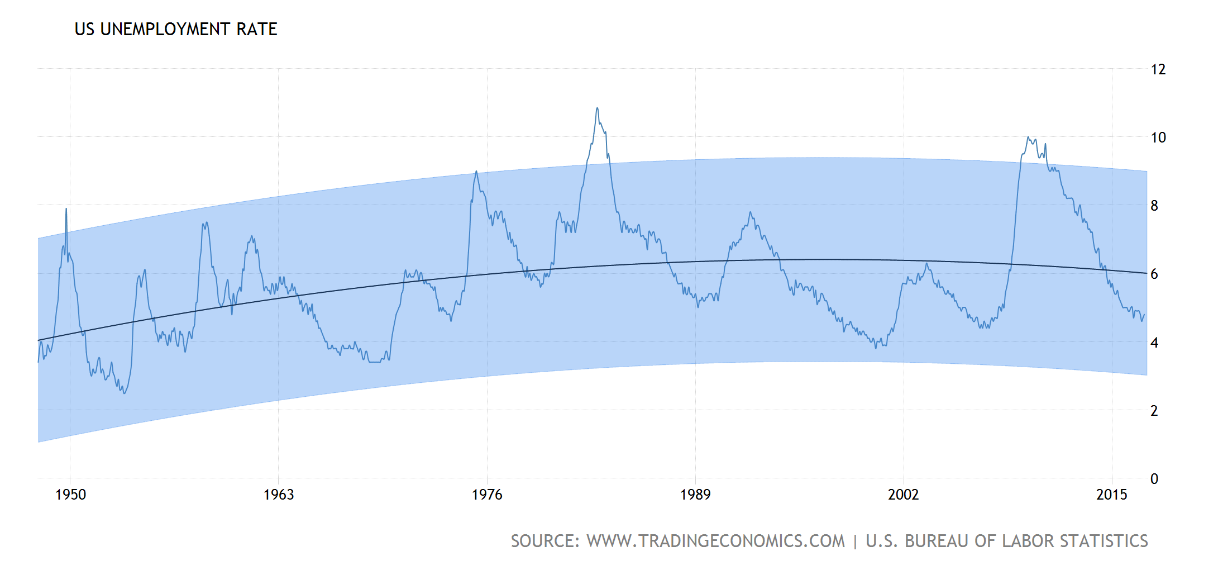
1. **Objective**

The objective of this project is to predict the unemployment rate in the USA.

1. **Introduction**

Unemployment rate has been a major talking point in America’s political scenario especially since the global financial meltdown of 2008. Even though the unemployment rate has majorly been steady in the range of 3%-6%, there have been times, like the 1970s and 1980s recession and the global financial meltdown of 2008, when it rose to a staggering 7%-10%, with a peak of 10% in October ’09.

Since then, controlled economic policies and measures have led to revival of job market. In February 2017, US economy added 235,000 jobs and have seen the unemployment rate lower down to 4.7%.



However, the major talking point still remains about the unemployment rate in terms of benefits. Though policies vary by state, unemployment benefits generally pay eligible workers up to $450 per week maximum. Benefits are generally paid by state governments, funded in large part by state and federal payroll taxes levied against employers, to workers who have become unemployed through no fault of their own. United States have allocated $36 Billion unemployment benefits in budget of fiscal year 2017. Thus, ensuring that the requisite measures are taken to keep the unemployment rates under control are necessary and forecasting plays a major role in it.

One of the greatest benefits of forecasting is that it provides the ability to manipulate the policies and other economic factors affecting the unemployment rate, so that it stays in control. Also, it helps gauge the extent of budget allocation required for the benefits.

1. **Methodology**

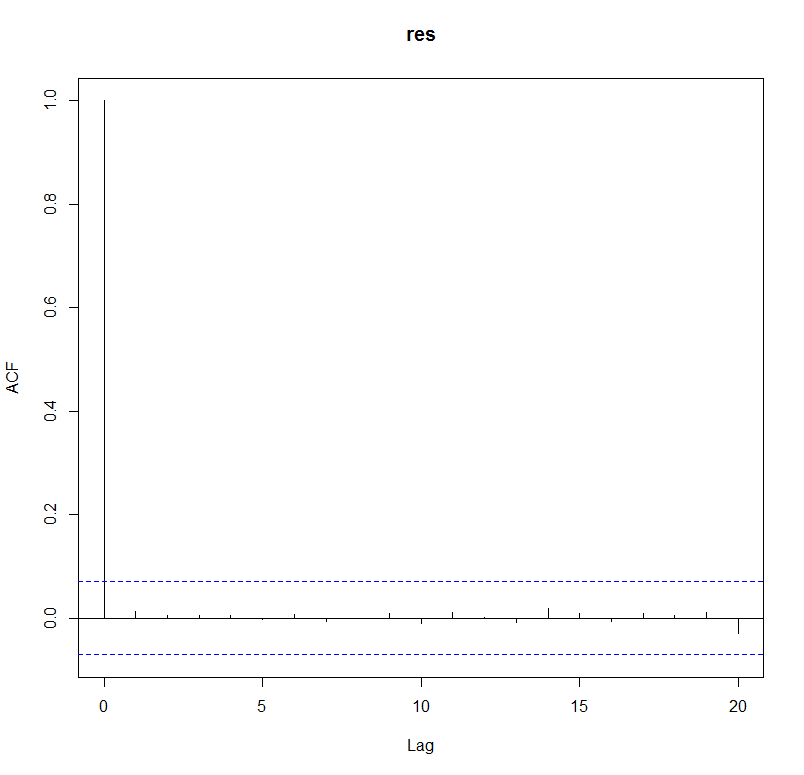
In this project, we have taken month-wise unemployment rate from January 1948 till February 2017 and have tried to build a time series model to forecast the rate.

We have used MATLAB and R to code for this project (Appendix I). We have divided the data into train and test set and then we have built a stationary model without trends and seasonality. To further evaluate the possibility of trends and seasonality, we have then built a model incorporating stochastic trends and stochastic seasonality. Also, a non-stationary time series model was developed.

Both stationary model and the non-stationary model are then used to do 20 steps ahead prediction. Further, we tried ARMAV model.

1. **Stationary Models**
   1. **Stationary Model without Trend & Seasonality**

To obtain the stationary model we first subtract the mean of the series from the entire series and then get the optimal ARMA model. We use the **“*armax”*** command in MATLAB to obtain the optimal model, which turned out to be **ARMA (22,21)** with a residual sum of error being **20.2376**. F- criteria is used for getting the optimal model. The details of the model are in Appendix II. Below mentioned is the ACF graph (Figure 1) to check for the correlations among the residuals.



**Figure 1**

One can notice that the ACF between residuals is within the required bounds. Hence, it shows that the residuals are just white noise and there is not structure within them that can be further modelled.

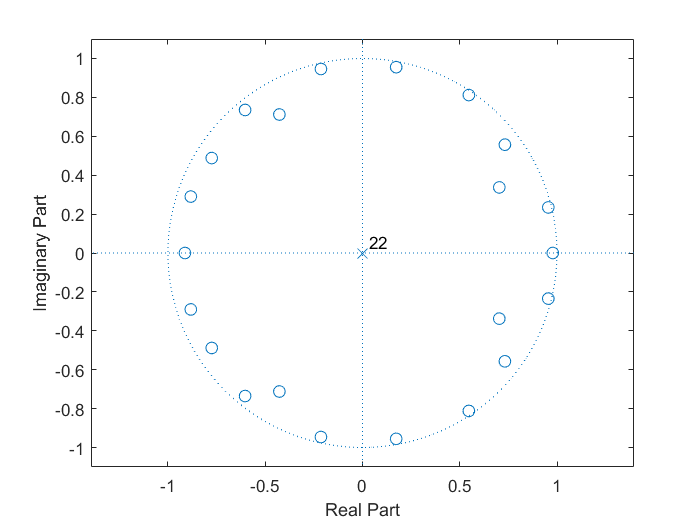
* 1. **Stochastic Trend & Seasonality**

To check for stochastic trend, we need to check for real roots of AR part of the model. If any of the real roots are close to 1 then it can result in stochastic trend. On the other hand, the complex roots of the AR part with magnitude close to 1 can cause stochastic seasonality.

From the roots of Auto Regressive part of ARMA (22,21), it was observed that none of the roots are closer to 1. Hence, theoretically, there is neither stochastic trend nor stochastic seasonality in the model.

Figure 2 represents the plot of all the roots on the unit circle. As we can see that none of the roots is close enough to the unit circle. Therefore, we can say that there is no stochastic trend or the stochastic seasonality in the time series

However, we didn’t stop just here. We empirically checked whether a stochastic trend or a stochastic seasonality model is better or not by taking into consideration the roots closest to the unit circle as the ones causing the trend and seasonality.



**Figure 2**

* + 1. **Empirical Check for Stochastic Trend**

The closest AR root to unity is 0.9787. We considered this root to be the root causing the stochastic trend. The parsimonious model would be:

**(1-B) \* Xt**

For this new series, we applied the model **ARMA (21, 21)** and calculated the residual sum of squares, which turns out to be **21.2711**.

We then performed the F-test to check the significance of drop of RSS from parsimonious model to **ARMA (22,21).** The F-value, **36.36**, turns out to be greater than the critical value (F critical is **3.84**).

Thus, we can say that the stationary model without stochastic trend is better than the parsimonious model having stochastic trend.

* + 1. **Empirical Check for Stochastic Seasonality**

The complex conjugate AR root with magnitude closest to 1 are (0.9551 + 0.2344i) and (0.9551 - 0.2344i). The time period caused by these roots turns out to be **21** **months**. The parsimonious model for the model with stochastic seasonality would be:

**(1-1.911B + B2)\*Xt**

For this series, we applied the model **ARMA (20, 21)** and calculated the residual sum of squares, which turns out to be **21.0572**.

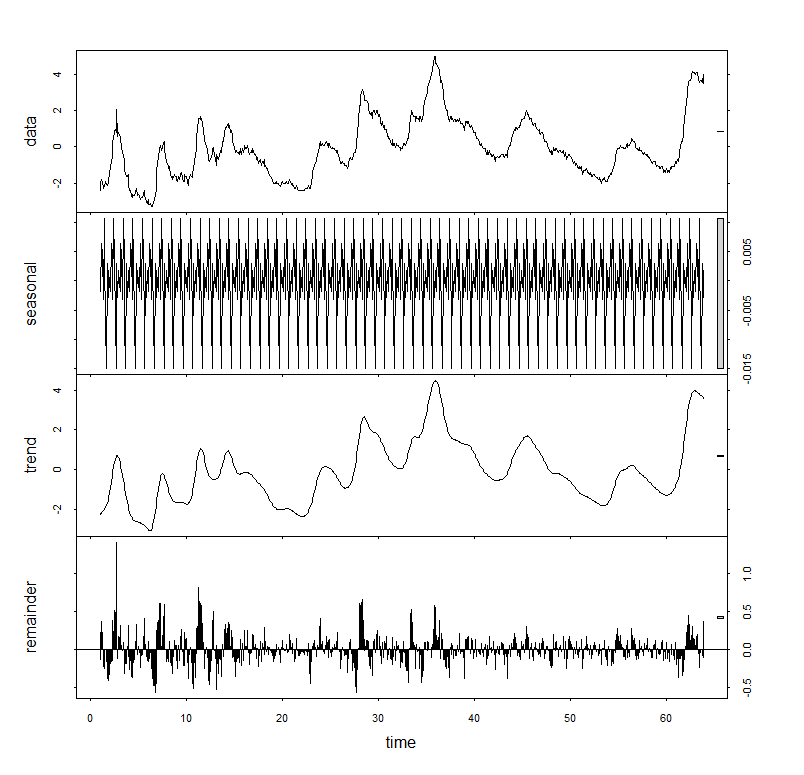
We then performed the F-test to check the significance of drop of RSS from parsimonious model to ARMA (22,21). The F-value, **14.42**, turns out to be greater than the critical value (F critical is **3**).

Thus, we can say that the stationary model without stochastic seasonality is better than the parsimonious model having stochastic seasonality.

Thus, we can say that the stationary time series model i.e. ARMA (22,21) is the better than both the parsimonious models and hence the time series doesn’t have any stochastic trend and seasonality.

1. **Non-Stationary Models**

The main methodology of building a non-stationary time series is to decide whether to fit an exponential function, harmonic or polynomial function for the deterministic part in the series. So, we decided to apply the function which is dominating in the time series. We decompose the time series into seasonality, trend and remainder. We have used ***“STL”* (Seasonality and trend by Loess Smoothing)** command in R to decompose the time series. Loess Method is explained in the (Appendix II). Figure 3 represents the graph of the different components of the decomposed time series.



**Figure 3**

As we can see that the trend seems to be deterministic part in the time series in case of non-stationarity, therefore, we considered the deterministic part to be a trend and applied different order polynomial functions of time for the deterministic part to figure out the optimal model. Using the F-Criteria, we found out that the deterministic model is of order 2.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **model** | **Sigma^2** | **Degree of Freedom** | **Residual sum of squares** | **F-value** | **F-critical** |
| 1st Order trend ( a+ b\*t) | 1.533 | 754 | 1155.882 | 29.58768 | 3.84 |
| 2nd Order trend (a+b\*t+c\*t2) | 1.477 | 753 | 1112.181 | 3.044807 | 3.84 |
| 3rd Order Trend (a+b\*t+c\*t2+d\*t3) | 1.473 | 752 | 1107.696 |  |  |

Since, The F value when we went from 2nd order polynomial to 3rd order polynomial model is not significant, hence the best model is 2nd order model. So, the deterministic part consists of 2nd order polynomial as shown below:

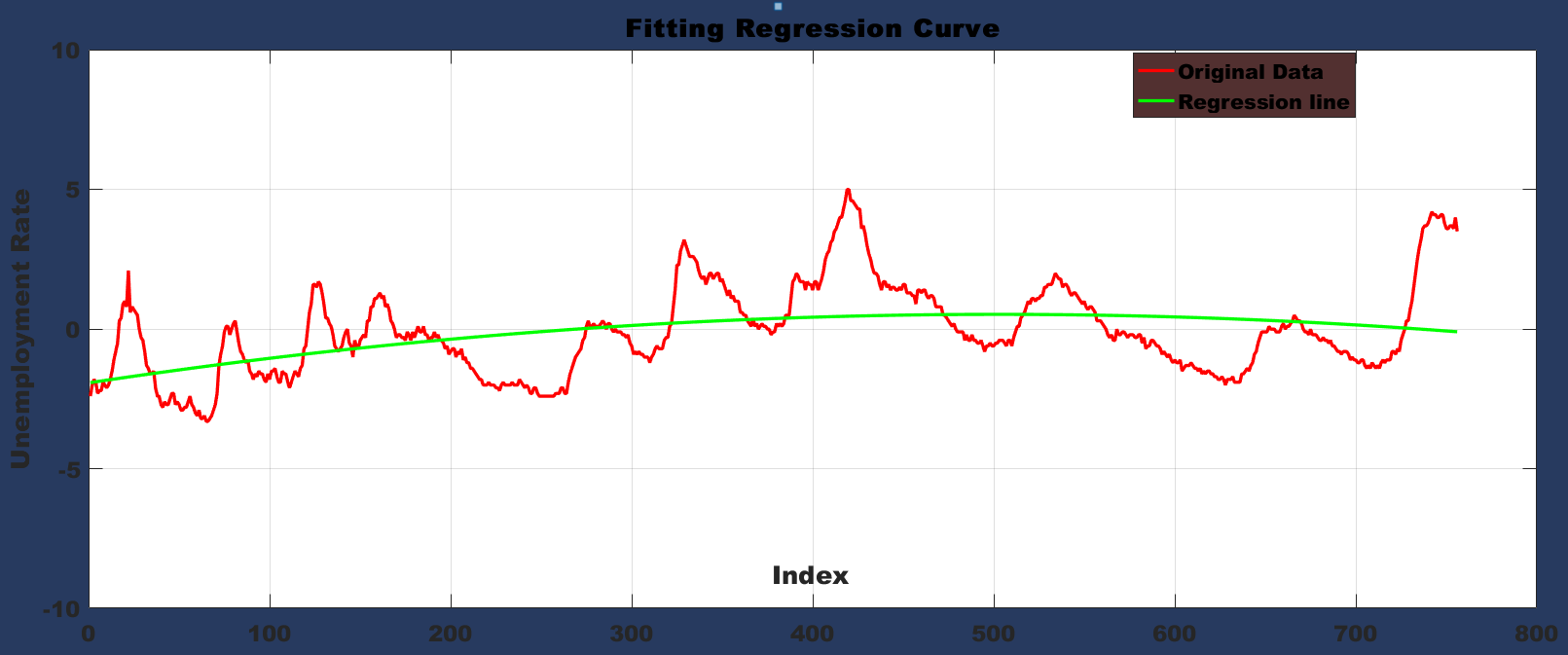
**-1.93+(9.76E-03)\*(t) –(9.71E-06)\*(t2)**

After removing the deterministic part from the time series, we are left with the remainder series, which still can be modelled further. We used the ***“armax”*** function in MATLAB to determine the best model and it turned out to be ARMA **(8, 7)**. Refer Appendix II for the details of ARMA (8, 7) model.

Thus, the final non-stationary model is as shown below:

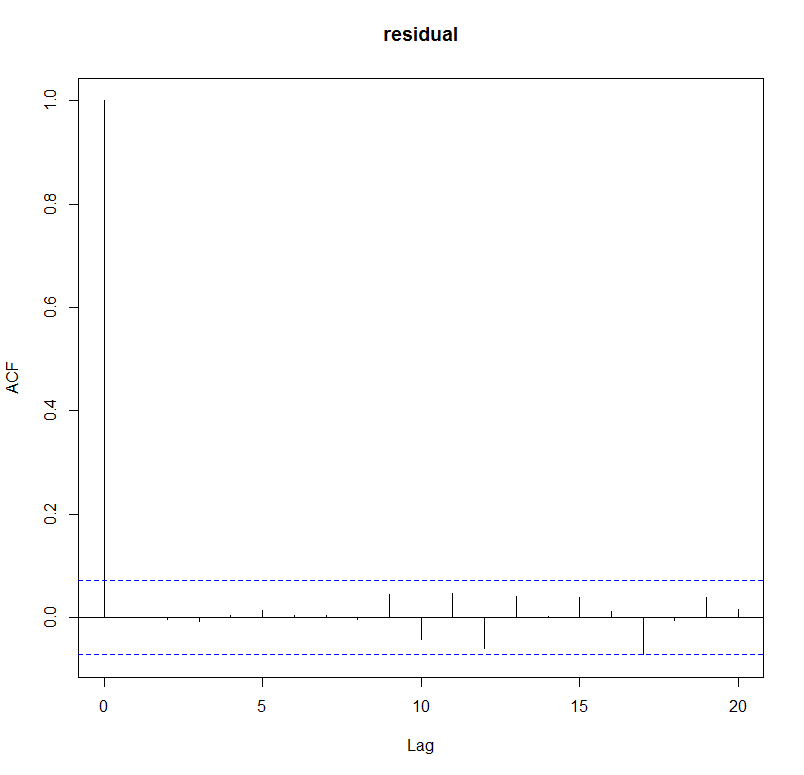
**Yt = -1.93 +9.76E-03\*t –(9.71E-06\*t2) + ARMA (8,7)**

The regression curve of 2nd order polynomial is shown below:



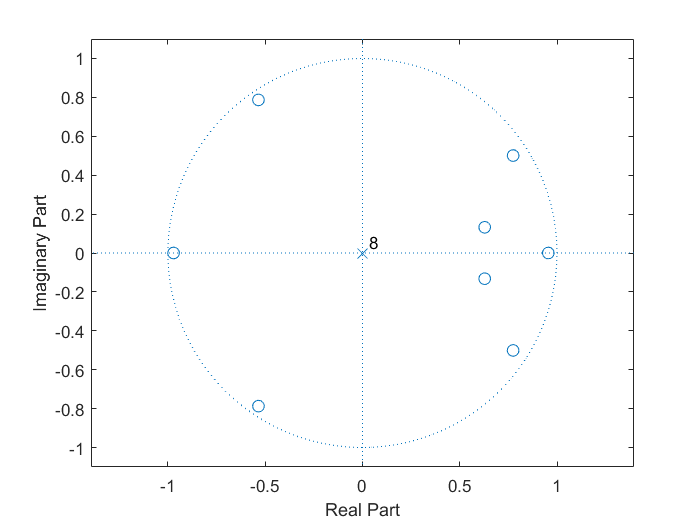
**Figure 4**

Figure 5 represents the ACF plot of the residuals of the ARMA (8, 7) model. As, we can see that the residuals are not correlated and are within the bounds, therefore, we can say that the residuals are white noise and the model is adequate.



**Figure 5**

Below shown Figure 6 represents the plot of AR roots of the ARMA (8, 7) model on the unit circle. Since the roots lie inside the unit circle, hence there is no stochastic trend and seasonality.

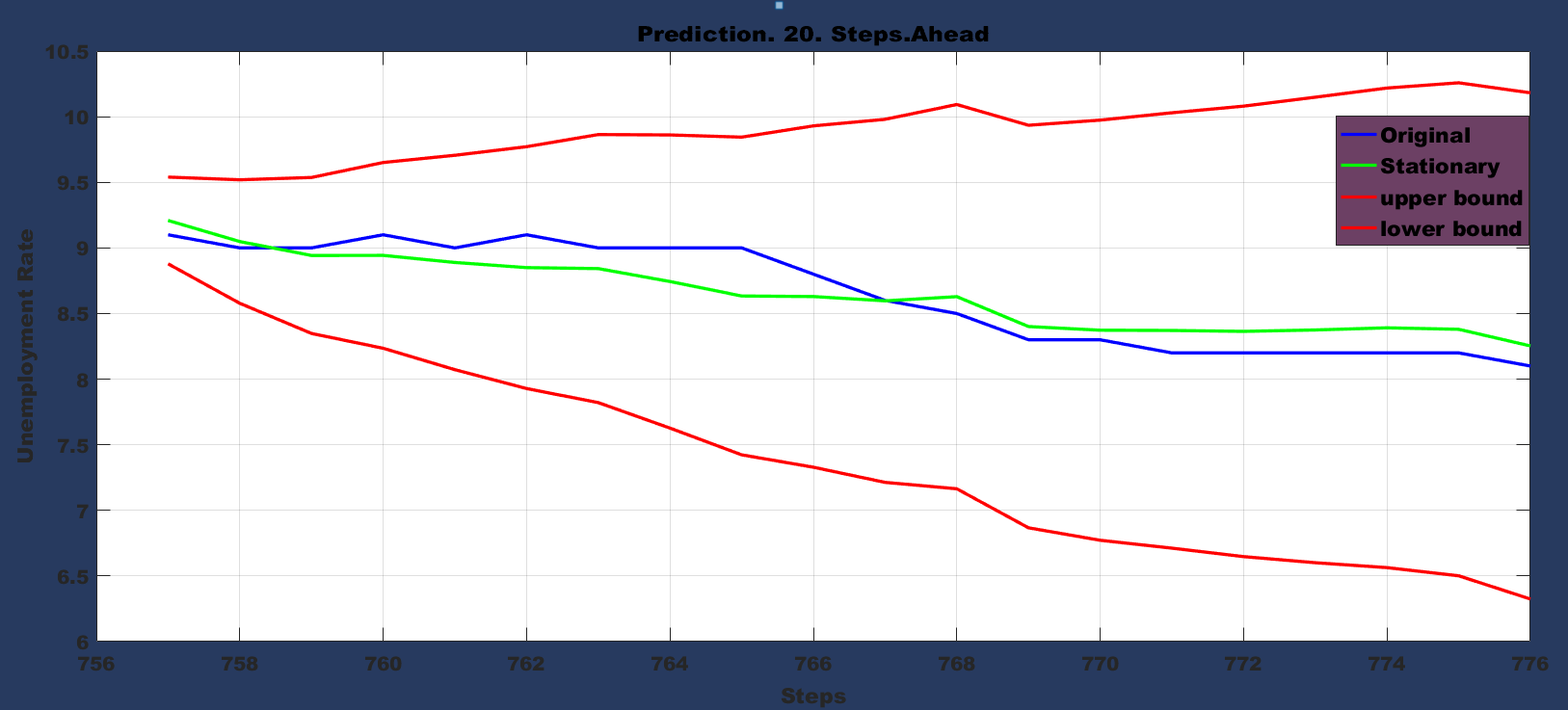


**Figure 6**

1. **Predictions**

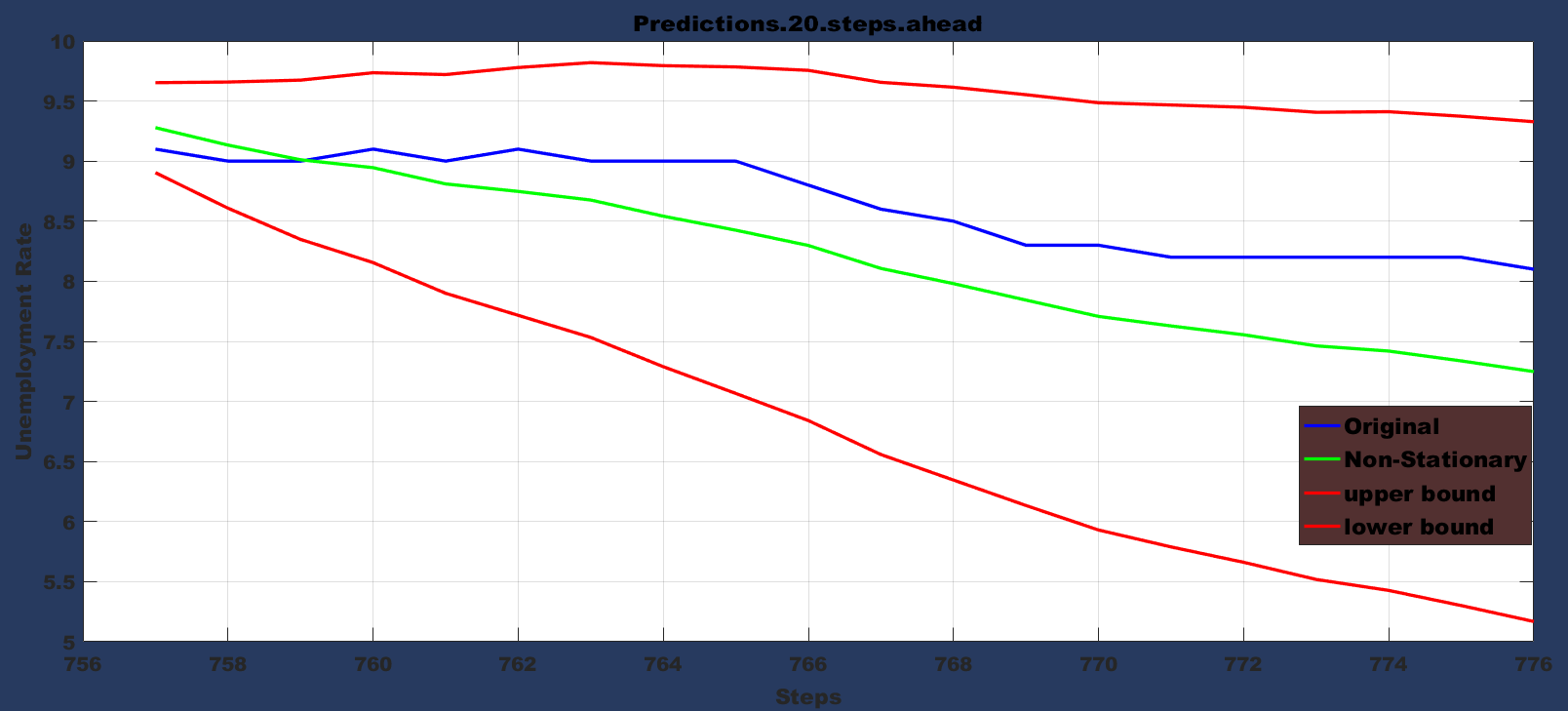
The ARMA (22, 21) stationary model and the non-stationary models were used to make 20 steps ahead prediction on the data. It was observed that the stationary model has a lower sum of squared prediction error, **0.58**, in comparison to the non-stationary model, **5.57**. Hence, the stationary model is a better model than the non-stationary model.

Below shown is the prediction plot along with 95% confidence interval (figure 7) for the stationary model.

****

**Figure 7**

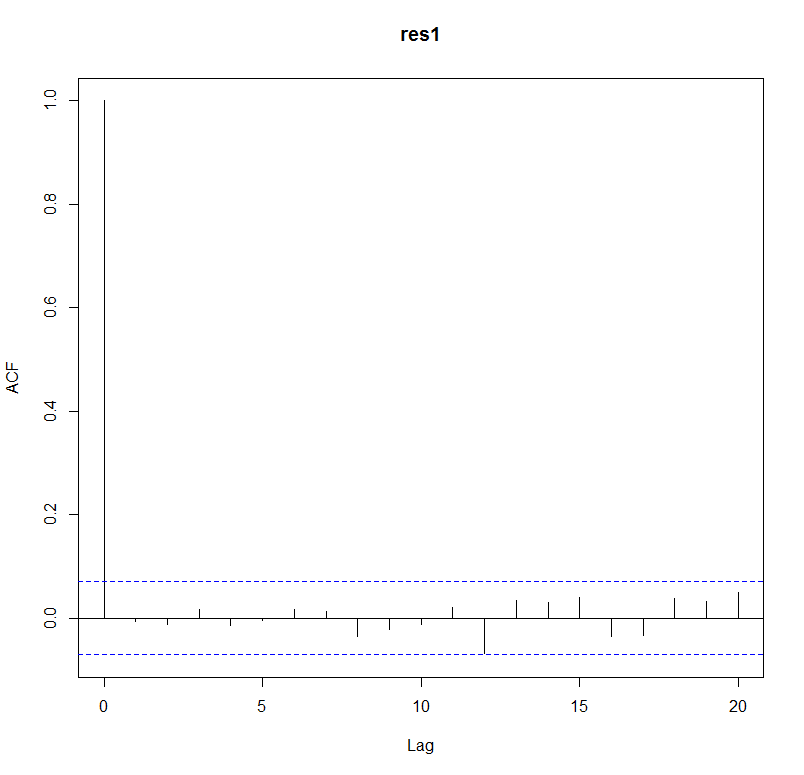
Below shown is the prediction plot along with 95% confidence interval (figure 8) for the non-stationary model.



**Figure 8**

1. **ARMAV MODEL**

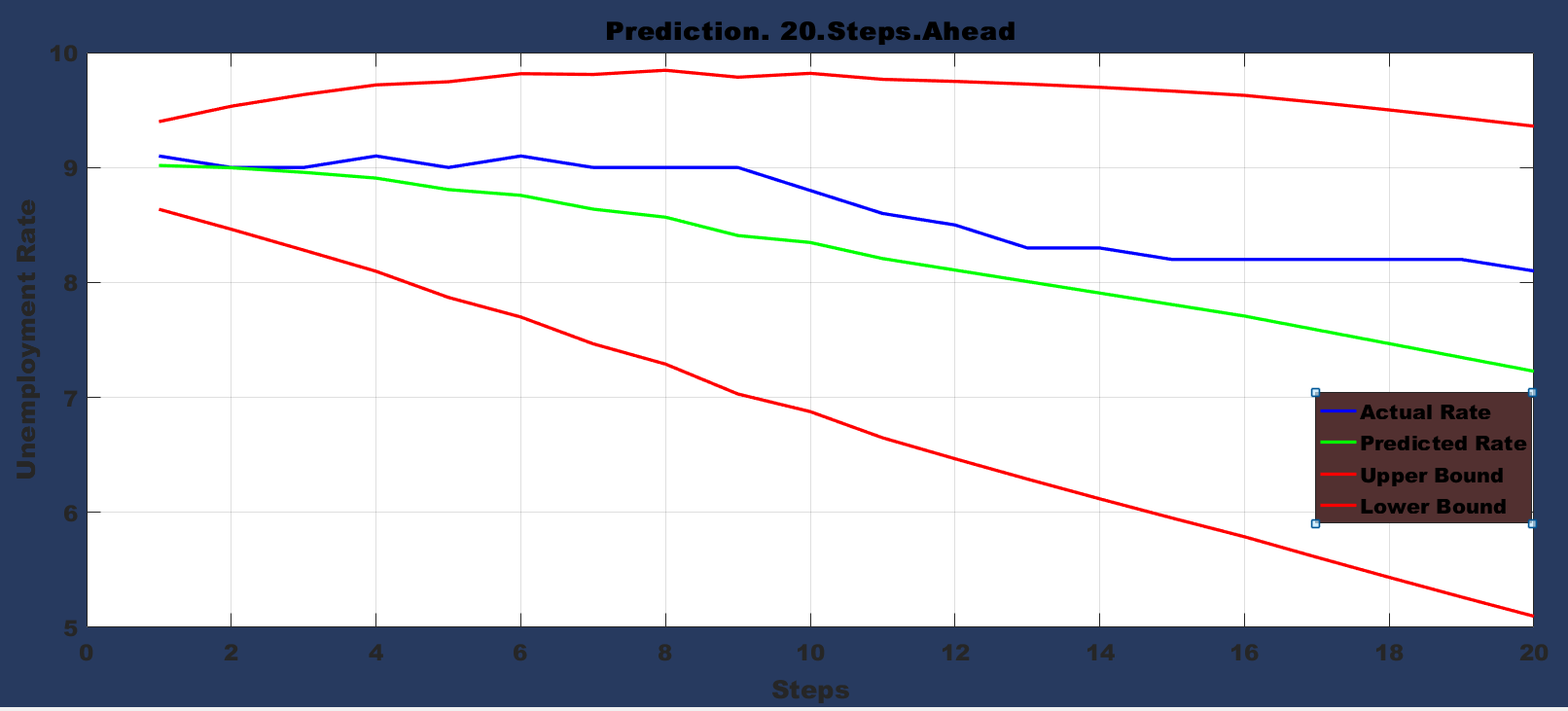
We added **“Labour Force Participation rate”** as another time series in the data set to see whether we can improve the 20 Steps ahead Prediction model. Based on F statistics, the best model is ARMAV (3,3,2). The acf curve of the residuals is shown in (Figure 9).



**Figure 9**

1. **Prediction Using ARMAV**

20 steps ahead prediction done using ARMAV model is shown in (Figure 10)



**Figure 10**

1. **Conclusion**
2. In the original data set which consists of unemployment Rate, there is no stochastic trend and seasonality and we have checked that empirically.
3. 20 Steps ahead prediction done by stationary time series and Non-Stationary time series led to prediction error of 0.58 and 5.57.
4. To improve the prediction further, we added one more time series (“Labour Force Participation Rate”) as input to our actual time series. The prediction error of ARMAV model is 4.41. Although, the prediction error is less than Non-Stationary Model but it is still higher than Stationary Model, hence we would recommend Stationary ARMA (22,21) model for the Prediction.
5. **REFERENCES**
   1. <https://www.washingtonpost.com/news/wonk/wp/2017/03/10/u-s-added-235000-jobs-in-february-unemployment-rate-drops-to-4-7-percent/?utm_term=.fcb802ad1b94>
   2. <https://en.wikipedia.org/wiki/Unemployment_benefits>
   3. <https://www.dol.gov/sites/default/files/documents/general/budget/FY2017BIB_0.pdf>

**APPENDIX-I**

**Model Code:**

**First we de-mean the data.**

N=length(value);

Cycle=1;

Data=iddata(value);

**%Initializing**

CurrentModel=armax (Data, [2 1]);

n=1;

r=resid (CurrentModel, Data);

residuals=r.y;

CurrentRSS=sum (residuals. ^2); **%residual sum of squares**

while Cycle

n=n+1;

Old Model=CurrentModel;

OldRSS=CurrentRSS;

CurrentModel=armax (Data, [2\*n 2\*n-1]);

r=resid (CurrentModel, Data);

residuals=r. y;

CurrentRSS=sum (residuals. ^2); **%residual sum of squares**

Test Ratio=((OldRSS-CurrentRSS)/4)/(CurrentRSS/(N-4\*n));

if Test Ratio<2.37

Cycle=0;

Preliminary Model=Old Model;

PreliminaryRSS=OldRSS;

end

end

AR\_Order=length (PreliminaryModel.a)-1;

MA\_Order=length (PreliminaryModel.c)-1;

%Now check if the odd valued model is good

CurrentModel=armax (Data, [AR\_Order-1 AR\_Order-2]);

r=resid (CurrentModel, Data);

residuals=r.y;

CurrentRSS=sum (residuals. ^2); %residual sum of squares

TestRatio=((CurrentRSS-PreliminaryRSS)/2)/(PreliminaryRSS/(N-(2\*AR\_Order-2)));

if Test Ratio<3

Preliminary Model=CurrentModel;

PreliminaryRSS=CurrentRSS;

end

**%Now, removing the unnecessary MA parameters**.

AR\_Order=length (PreliminaryModel.a)-1;

MA\_Order=length (PreliminaryModel.c)-1;

CurrMA=MA\_Order;

CurrentModel=Preliminary Model;

CurrentRSS=PreliminaryRSS;

if CurrMA>1

Cycle=1;

else

Cycle=0;

Model=Preliminary Model;

RSS=PreliminaryRSS;

end

while Cycle

Old Model=CurrentModel;

OldRSS=CurrentRSS;

CurrMA=CurrMA-1;

CurrentModel=armax (Data, [AR\_Order CurrMA]);

r=resid (CurrentModel, Data);

residuals=r. y;

CurrentRSS=sum (residuals. ^2); **%residual sum of squares**

NumOfParams=AR\_Order+CurrMA+1;

TestRatio=((CurrentRSS-PreliminaryRSS)/1)/(PreliminaryRSS/(N-NumOfParams));

if Test Ratio>3.84

Cycle=0;

Model=Old Model;

RSS=OldRSS;

end

end %Done

r=resid (Model, Data);

res=r. y;

**% Auto Correlation function plot in r studio**

Acf(res)

**% roots of AR\_order and MA\_order**

AR\_Poly=Model.a;

MA\_Poly=Model.c;

l=roots(AR\_Poly);

k=roots(MA\_Poly);

**% roots on unit circle**

zplane(AR\_Poly);

**%plot the Green function**

n=length(l);

for i=1: n

Down=1;

Up=polyval(MA\_Poly,l(i));

for j=1: n

if j~=i

Down=Down\*(l(i)-l(j));

end

end

C(i)=Up/Down;

end

for i=1:19

G(i)=0;

for j=1: n

G(i)=G(i)+C(j)\*l(j)^i;

end

end

G= [1 real(G)];

**% for stochastic seasonality**

**%**parsimonious model is (1-1.911B+B^2) Xt

**%** Applying ARMA (20,21) to the time series.

Pars\_model=armax (Data, [20 21]);

Pars\_r=resid (Pars\_model, Data);

Pars\_residuals=Pars\_r.y;

Pars\_RSS=sum (Pars\_residuals. ^2);

**% for stochastic trend.**

**%**parsimonious model this time is(1-B) Xt

% Applying ARMA (21,21) to the time series.

Pars\_model1=armax (Data, [21 21]);

Pars\_r1=resid (Pars\_model1, Data);

Pars\_residuals1=Pars\_r1.y;

Pars\_RSS1=sum (Pars\_residuals1. ^2);

**% for non-stationary model.**

%model isYt= -1.93+9.76E-03(t) -9.71E-06(t2) + ARMA (8,7)

**% forecasting for next 20 months**

prediction=forecast (Model, value,20) **%using stationary model**

prediction1=forecast (Non\_stationary\_model, value,20) **%using non-stationary model**.

# **For ARMAV model**

data1=iddata(value,Rate1,1);% **Rate1 is the new series**

data2=iddata(Rate1,value,1);

sys1=cell(1,25);

ct=1;

for n= 1:25

sys1{ct}=armax(data1,[n n n-1 0]);

ct=ct+1;

end

sys2=cell(1,15);

ct=1;

for n= 1:15

sys2{ct}=armax(data2,[n n n-1 0]);

ct=ct+1;

end

**% for F test of data1:**

Cycle=1;

CurrentModel=sys1{1};

j=1;

r=resid(CurrentModel,data1);

residuals=r.y;

CurrentRSS=sum(residuals.^2); %residual sum of squares

while Cycle

j=j+1;

OldModel=CurrentModel;

OldRSS=CurrentRSS;

CurrentModel=sys1{j};

r=resid(CurrentModel,data1);

residuals=r.y;

CurrentRSS=sum(residuals.^2); %residual sum of squares

TestRatio=((OldRSS-CurrentRSS)/3)/(CurrentRSS/(456-3\*j));

if TestRatio<2.60

Cycle=0;

PreliminaryModel=OldModel;

PreliminaryRSS=OldRSS;

end

end

**% For F test of data 2:**

cycle=1;

CurrentModel=sys2{1};

i=1;

r=resid(CurrentModel,data2);

residuals=r.y;

CurrentRSS=sum(residuals.^2); %residual sum of squares

while Cycle

i=i+1;

OldModel=CurrentModel;

OldRSS=CurrentRSS;

CurrentModel=sys2{i};

r=resid(CurrentModel,data2);

residuals=r.y;

CurrentRSS=sum(residuals.^2); %residual sum of squares

TestRatio=((OldRSS-CurrentRSS)/3)/(CurrentRSS/(456-3\*j));

if TestRatio<2.60

Cycle=0;

PreliminaryModel=OldModel;

PreliminaryRSS=OldRSS;

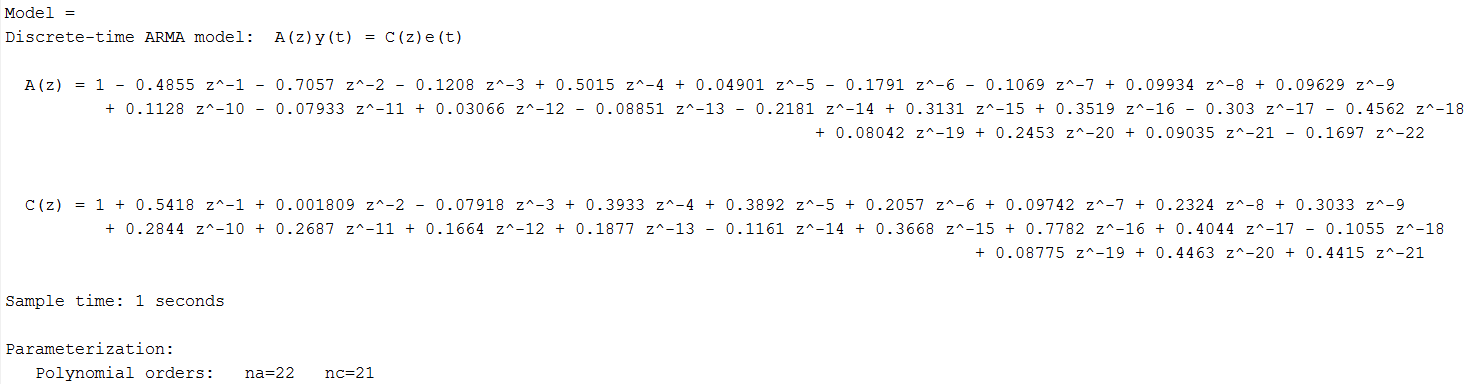
end

end

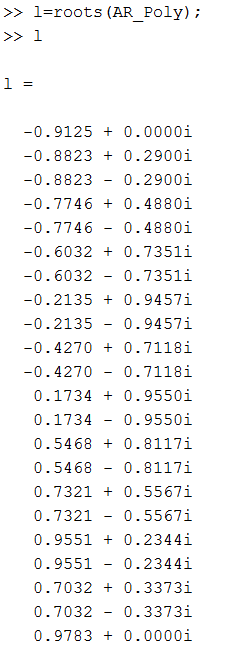
**APPENDIX-II**

**Stationary Model Details:**

**ARMA (22, 21)**

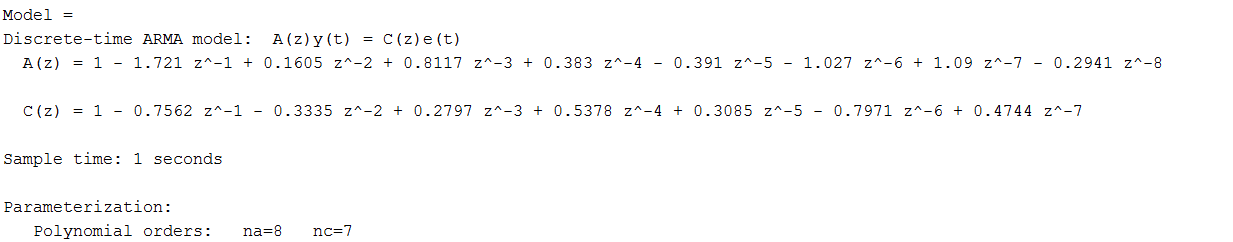
****

**AR\_Roots**

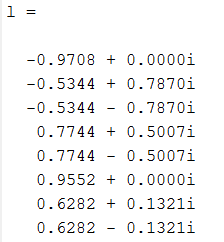
****

**Non- Stationary Model Details:**

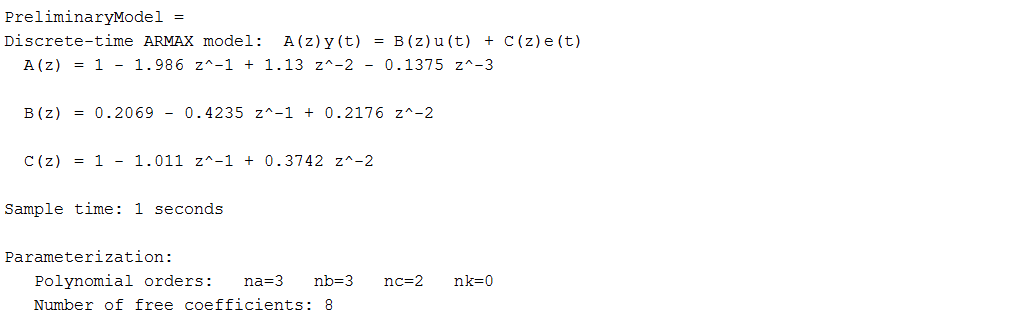
**ARMA (8, 7)**



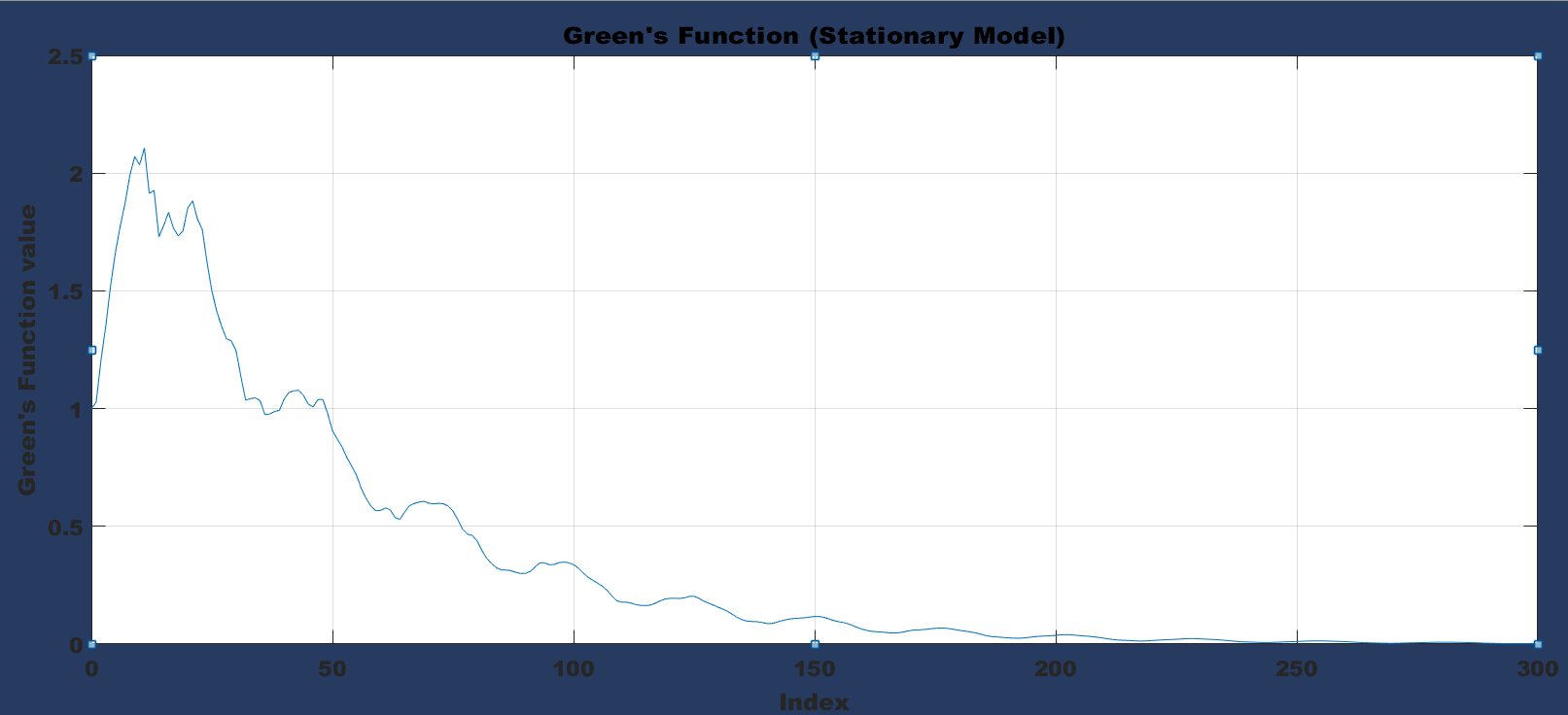
**AR\_ Roots of ARMA (8,7)**

****

**ARMAV MODEL**

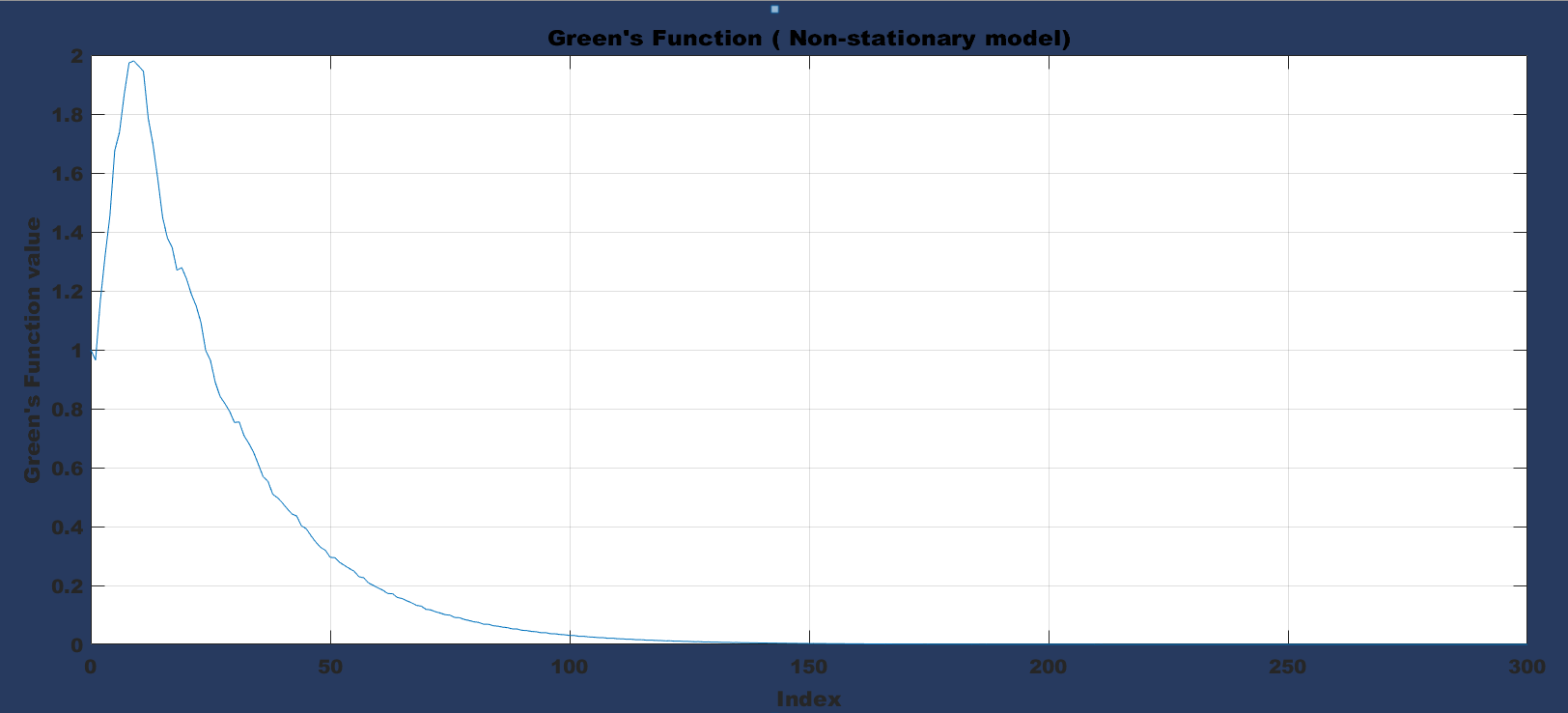
****

**Green’s Function curve (Stationary Model)**

****

**FIGURE 8**

**Green’s Function non-stationary model**

****

**FIGURE 9**

**STL (Seasonal and trend by Loess)**

**Loess** or **Lowess** stands for locally weighted scatterplot smoothing which is used to decompose the time series into seasonal, trend and irregularities components. Shown below is the decomposition of 18 instances of our time series. Command to carry out this function in r is

Decompose=Stl (time series)

Plot(Decompose)

**Working:**

1. A window of specific width is placed over the data. The wider the width, smoother the resulting loess curve is.
2. A regression line is fitted to the time series that fall within the window, the points closest to the centre of the window being weighted to have the greatest effect.
3. The weighting is reduced on those points within the window that are furthest from the regression line. The regression is re-run and weights are re calculated.
4. The loess curve is calculated by moving the window over the entire data set. Each point on the loess curve is the intersection of a regression line and a vertical line.

Since, our data is monthly so we consider the window as 12.