Time Series Analysis Assignment

Instructors : Prof. Subir Kumar Bhandari and Monitirtha Dey
Interdisciplinary Statistics and Research Unit, Indian Statistical Institute, Kolkata

November 15, 2022

Debarshi Chakraborty
Roll no : MD2105
Master of Statistics, 2nd year
Applied Statistics Specialization
Email : maharajtheboss@gmail.com

Contents

1 Introduction

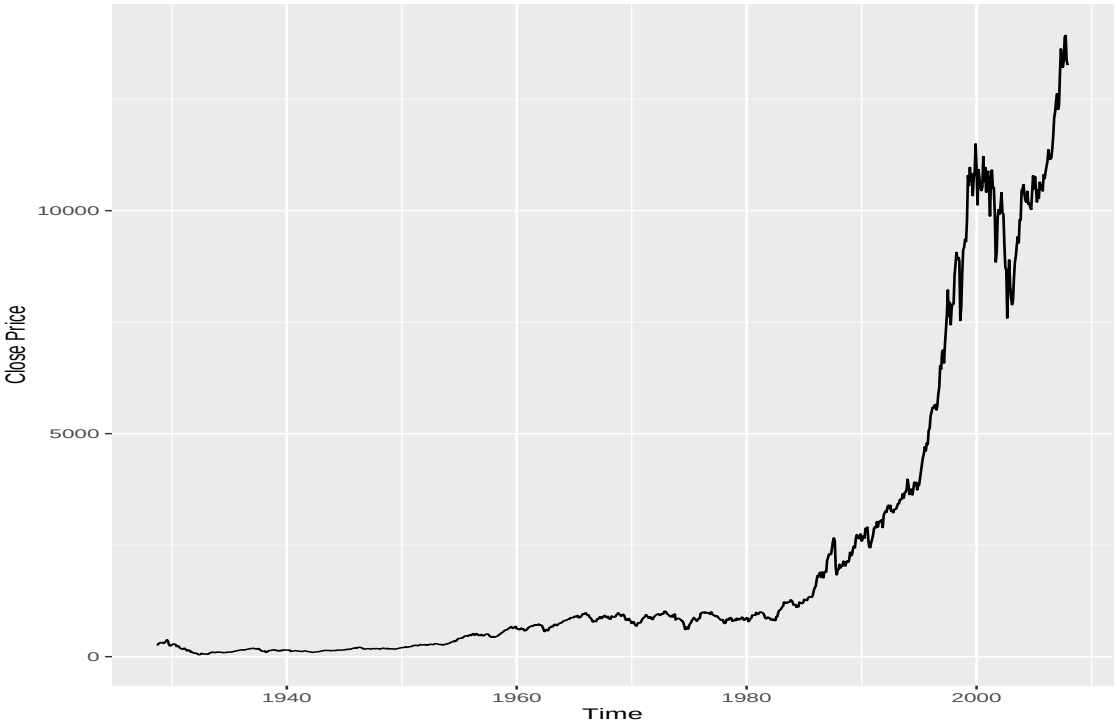
In this project, we work with a univariate time series data. We try two methods for modeling, in the first attempt our goal is to separate out the deterministic component (trend and seasonality in our case) and the random component, then model the random component using ARMA. The next method involves directly modeling the original time series using SARIMA. After modeling with ARMA nd SARIMA, we check the residuals using autocorrelation plots, partial autocorrelation plots and some Portmanteau tests.

2 Data Description

2.1 Source of data

We have a dataset available as **djiclose** inbuilt in the R package **fpp2,** also in **expsmooth**. It has two colums "close" and "pcreturn". We are to work with the **close** column only. It represents the closing values of Dow Jones Index on the first day of each month. Data is available from October 1928 to December 2007. Seems like we have sufficient data to analyze. Let us take a look at the data at first.

2.2 Plot of raw data



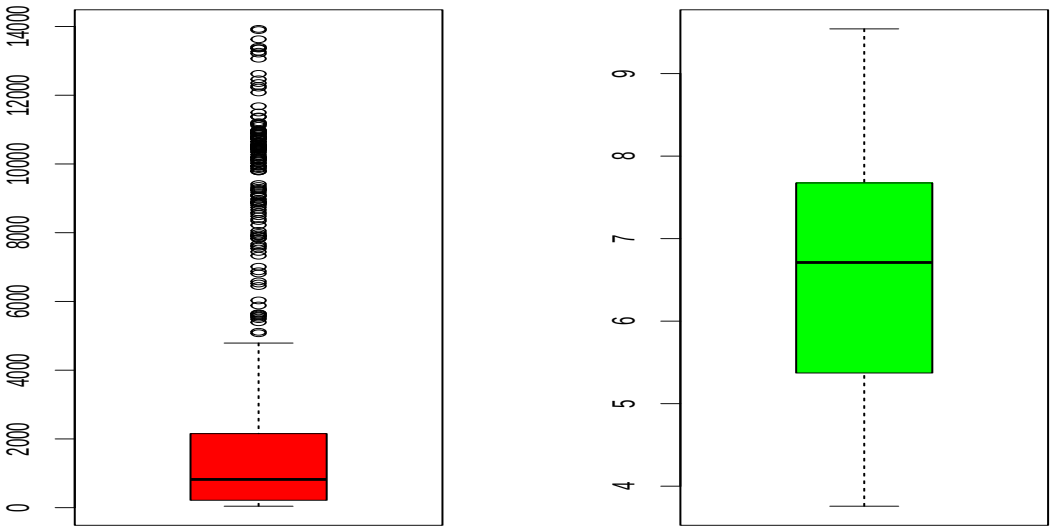
3 Trend and Seasonality

3.1 Transforming the data and choice of model

At first, it is very clear that since monthly data is provided, thus both trend and seasonal patterns are present. But, before going into modeling directly, we should observe the plot closely. A rough inspection gives an idea that the seasonal fluctuations increase as the trend goes upward rapidly, especially around the last 2 decades. Hence the trend and seasonal components are not independent of each other. So we will prefer a multiplicative model over an additive model in our analysis. In other words, we can take logarithm of our original data and fit an additive model to that.

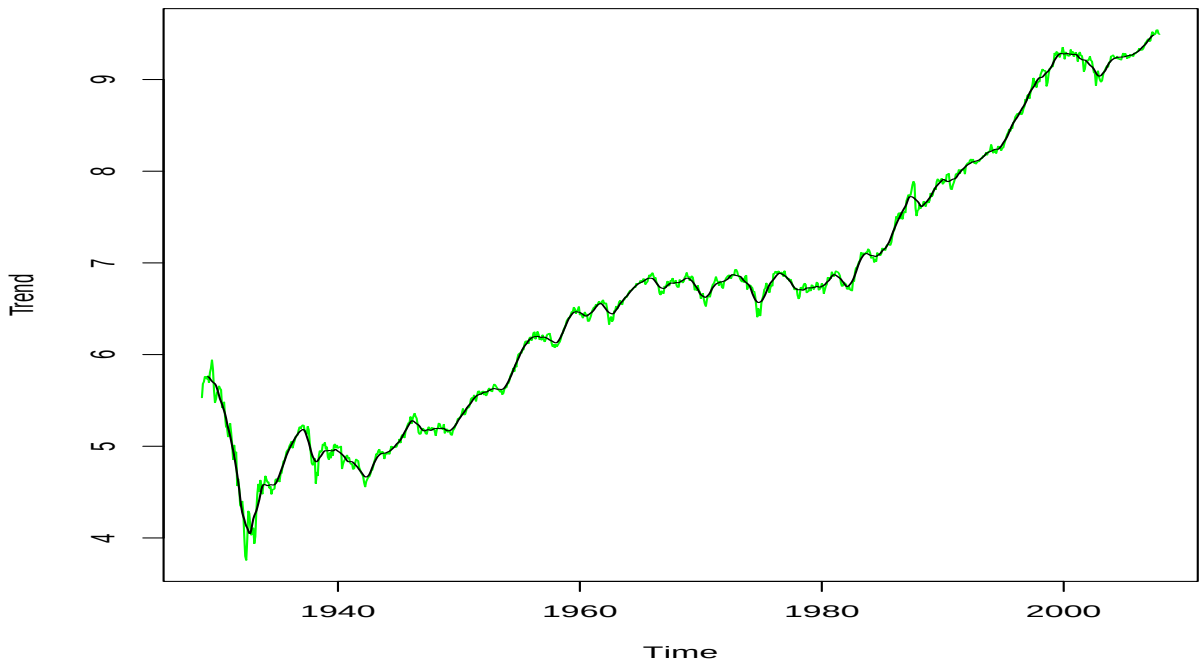
Also, let us see if there are any unusual observations. We take a look at one of the simplest EDA tools - the Boxplot. We note that there are a lot of observations outside the outer fence i.e. the data has high variance. Thus,

it can create problems if we work with the raw data. But, since we are doing a log transformation, hopefully this problem will be taken care of, the variance will be reduced automatically after taking the logarithm. We juxtapose the two boxplots to see it.



3.2 Estimating different components of the time series

Now, we can proceed to detrend and deseasonalize the data. The **decompose()** function in R makes our lives easier here. It estimates the trend, seasonal components and random fluctuations. Random fluctuations are calculated simply by subtracting the trend and seasonal components from the original data (log transformed data in our case). Looking at the estimated trend curve will be much more insightful than looking at the values, since method of moving averages is used here to estimate the trend.

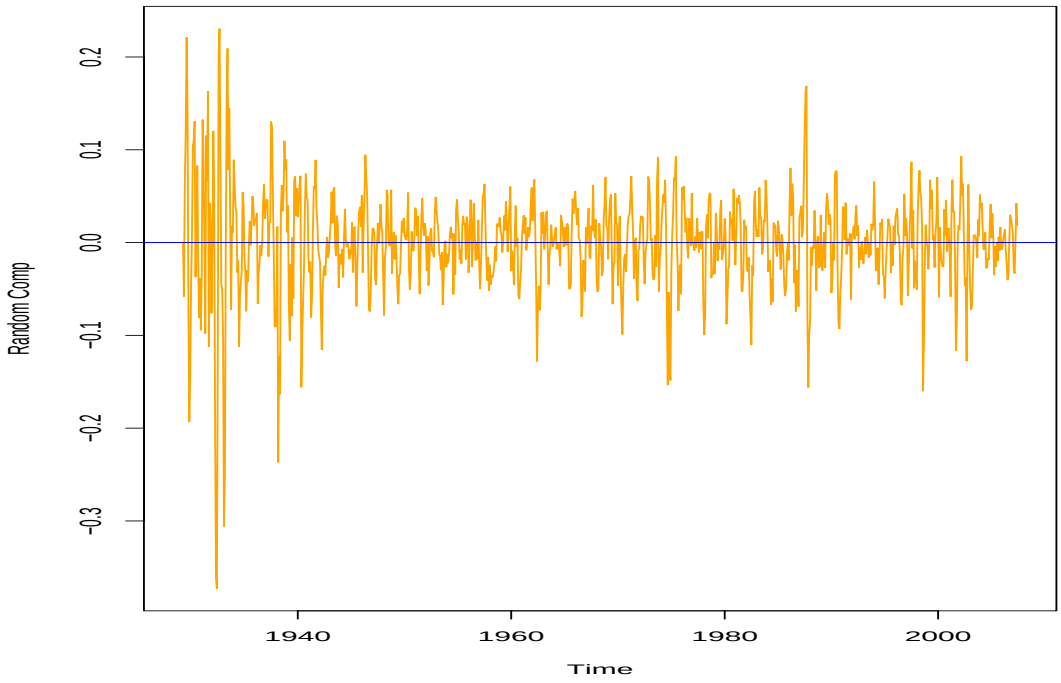


We have plotted the trend line on top of the actual (log transformed) time series to show how good the fit

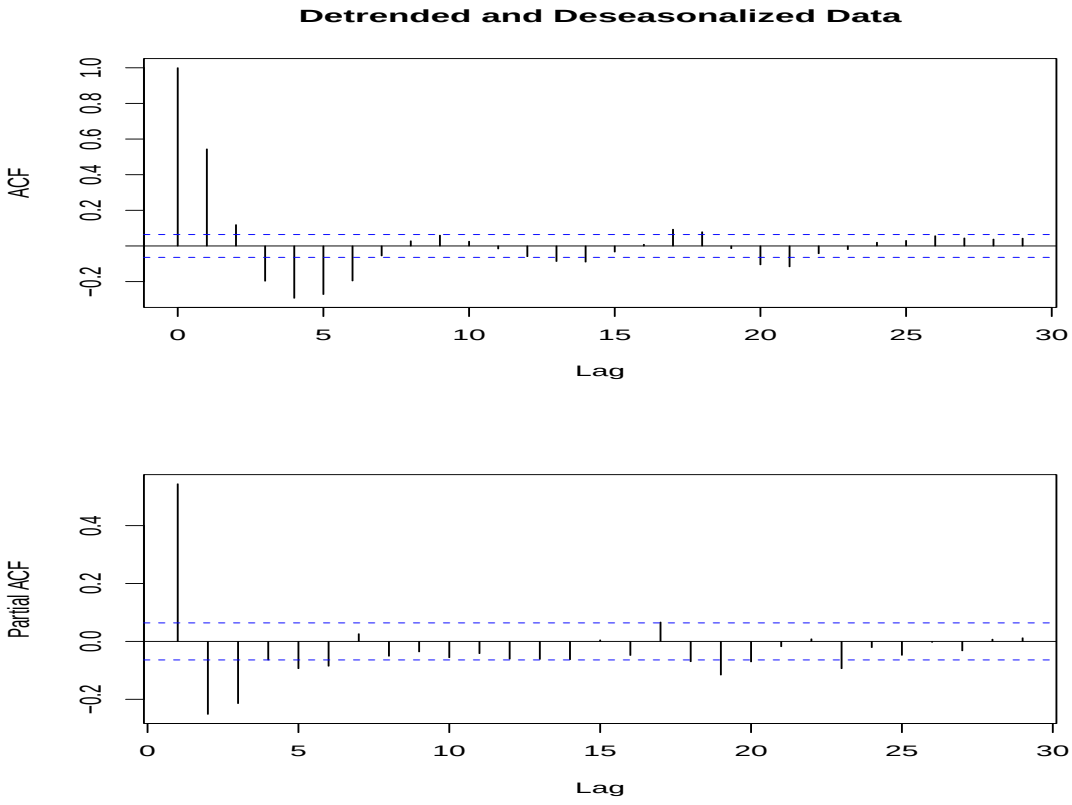
is.The Graph of seasonal component is not that useful, so we omit it in the report, it will be there in the code.

3.3 Visual Inspection of the Random Component

Now, let us examine the detrended and deseasonalized data.



Roughly it can be seen that the random component fluctuates symetrically around zero. We visualize the ACF and PACF plot to get a better idea.



Note that, the number of observations has decreased from 951 to 939 since 6 points in the beginning and the end each are lost due to the estimation using moving averages.

3.4 Guessing the values of ARMA parameters

From the plots, we can see that the autocorrelation plot cuts off at lag 6, thus ARMA(0,6) or MA(6) should be a good guess. On the other hand, the partial autocorrelation plot roughly cuts off at lag 3, thus ARMA(3,0) or AR(3) may also be a good choice.

4 ARMA Modeling

4.1 Choosing potential models

We can use the **auto.arima()** function in R to select the best model automatically according to some information criterion provided by us such as the Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC).We can get the optimal ARMA model or fit any ARMA (p,q) model just be keeping the argument "d" of ARIMA(p,d,q) fixed at 0. Here we will consider 4 choices of (p,q). The two choices are what we proposed by our intuition , one model automatically selected using stepwise selection method and another one selected among all possible models fixing the maximum value of p and q as 6.For the automatic selection, we have used both AIC and BIC as criterion, both yield same results in each case. The stepwise selection gives ARMA(5,0) as the best model while selection among all possible models give ARMA(2,1).
The estimates of the coefficients of the four different models are given by :

Model	MA(6)	AR(3)	ARMA(5,0)	ARMA(2,1)
Coefficient 1	0.5373	0.6262	0.6056	(AR1)1.4747
Coefficient 2	0.0889	-0.1043	-0.1269	(AR2)-0.6114
Coefficient 3	-0.3245	-0.2149	-0.1857	(MA1)-0.9887
Coefficient 4	-0.4376	-	-0.0058	-
Coefficient 5	-0.3642	-	-0.0962	-
Coefficient 6	-0.2568	-	-	-
Intercept	0	0	0	-

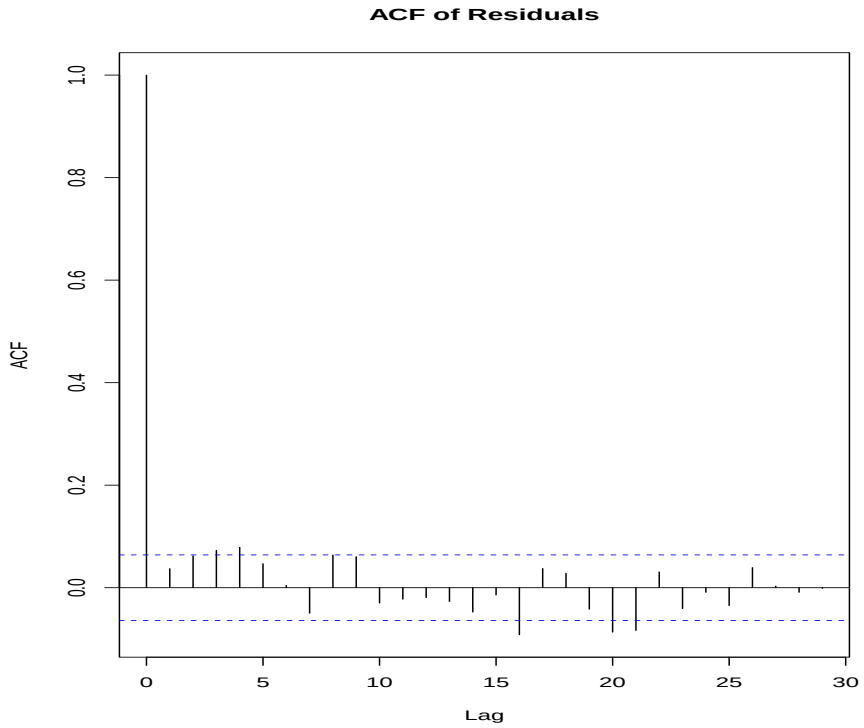
4.2 Comparing Models

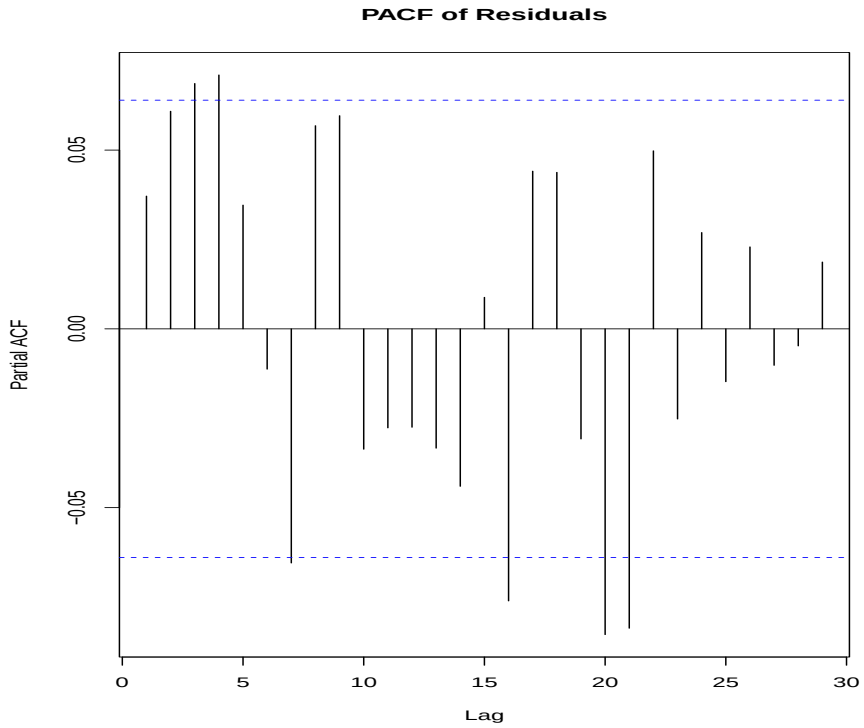
After fitting, we do Portmanteau Tests (Box- Pierce and Ljung Box Tests) and Rank Correlation Test on the residuals to check for white noise. the results are summarized below :

Model	AIC	BIC	Box Pierce p value	Ljung Box p value	Rank Corr p value
MA(6)	-3302.468	-3263.71	0.255	0.2543	0.8183
AR(3)	-3274.359	-3250.135	0.7009	0.7004	0.4442
ARMA(5,0)	-3282.761	-3248.847	0.8478	0.8476	0.3993
ARMA(2,1)	-3311.061	-3291.682	0.02571	0.02547	0.4309

Although the ARMA(2,1) model is automatically selected among all possible models due to it's least AIC, BIC values; but the residuals we get after fitting the model are not satisfying the Portmanteau Tests for white noise. Moreover, the AIC, BIC values of all four models are low enough, those values of ARMA(2,1) are not substantially less than the others to give it a preference over the other three. Since, the residuals produce by the other three models are not showing significant deviance from being white noise in terms of Portmanteau tests and rank correlation test, hence we choose the model with least AIC nd BIC amongst them. hence, our final model is the MA(6) or ARMA (0,6) model.

Since the p-values of the tests are already reported, we show the ACF and PACF plots of our chosen model.





The ACF and PACF plots closely resemble the pattern of those of a white noise.

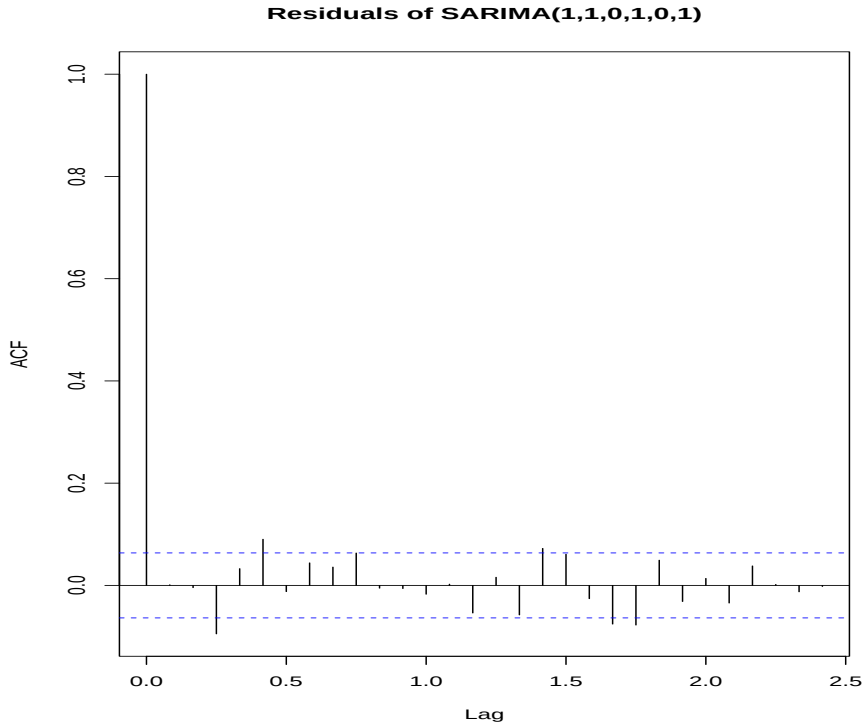
5 SARIMA Modeling

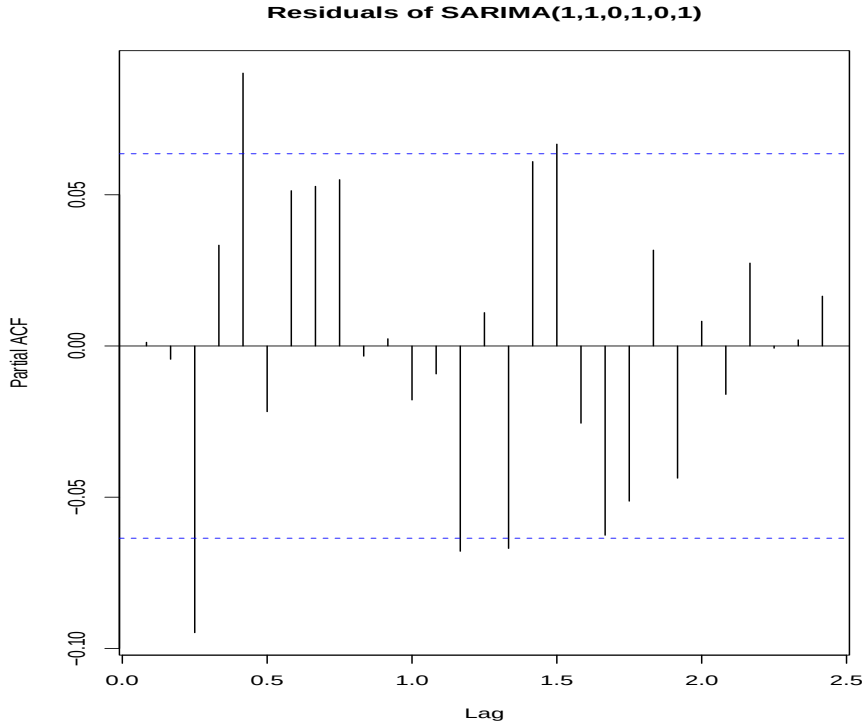
Now, we will fit a SARIMA model on our original data. Since we have worked with the log transformed data throughout the project, we will use that one only i.e. we fit an appropriate SARIMA model on the log transformed data without the trend and seasonality component removed. Here also we use the **auto.arima()** function which selects the best model using some information criterion provided by us.
We get **SARIMA(p=2,d=1,q=2,P=2,D=0,Q=2)** as our optimal model if we use AIC as our information criterion and **SARIMA(p=1,d=1,q=0,P=1,D=0,Q=1)** if we use BIC as our information criterion.

When we test the residuals for white noise obtained from both the models, the results come out to be:

Model	Box Pierce p value	Ljung Box p value	Rank Corr p value
SARIMA(2,1,2,2,0,2)	0.7585	0.7581	0.8605
SARIMA(1,1,0,1,0,1)	0.9695	0.9694	0.7104

The test results suggest that both of these models will perform good enough, so we choose the second one i.e. SARIMA(1,1,0,1,0,1) because it is much simpler than the other. the estimated model coefficients are given by : **AR(1) = 0.0692 , Seasonal AR(1)= 0.8772 , Seasonal MA(1) = -0.8464**.





The patterns of ACF and PACF roughly resemble that of a white noise.

6 Acknowledgements

I express my sincere gratitude to Professor Subir Kumar Bhandari for giving us the opportunity to have some hands on experience on the concepts we have learnt in our time series analysis course. Special thanks to Monitirtha Da for his constant guidance and help, without which the project would not have been completed as smoothly as it was.