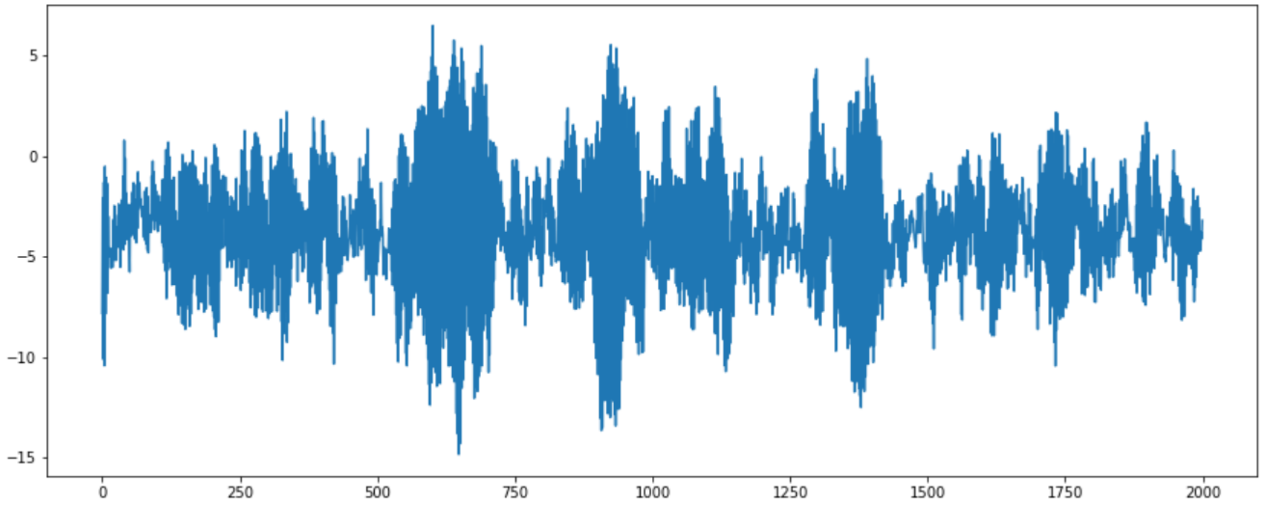
Forecasting Assignment

1) Check for Stationarity [trend (shift of mean), variable variance, seasonality]



We observe that the mean is constant throughout and there is no seasonality. But the variance is variable.

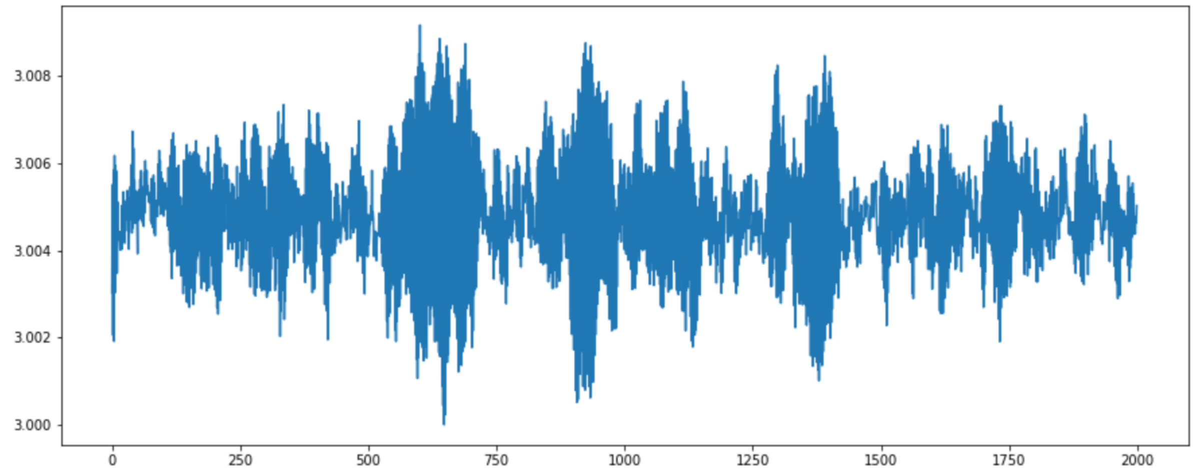
Hence, to remove the variable variance, first shifting all the values by the minimum value observed in order to make the values positive. Since adding the minimum value will make the smallest value equal to zero and if I take log transform, then the value will be undefined. Hence, adding a constant value of 1000 to all the values, so that after taking the log transformation gives minimum possible deviations between the values.

Hence, **transformed values = $\log_{10}$[(original values + min. value) + 1000]**

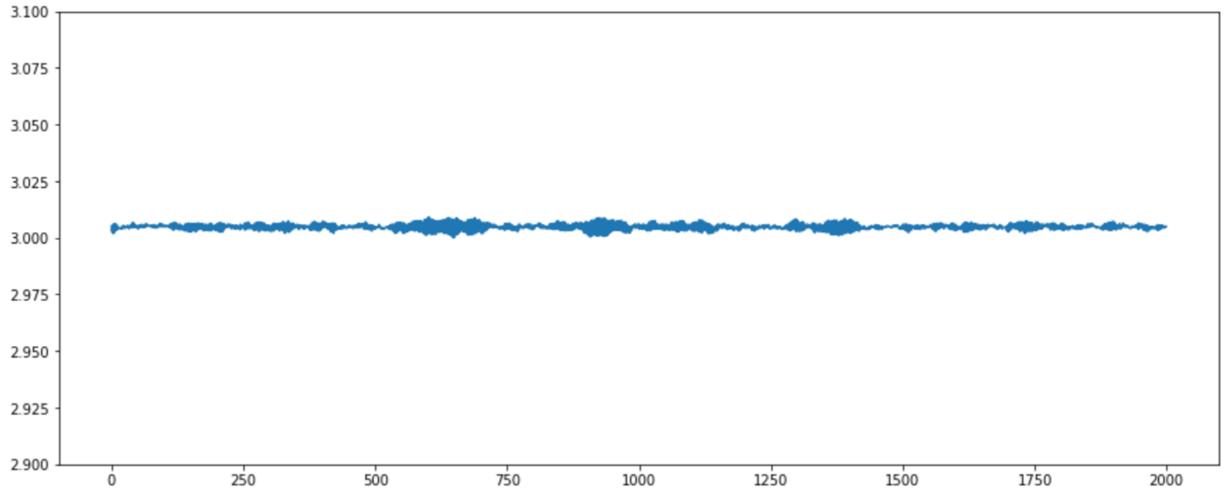
Here, min. value refers to the minimum value among the original set of values.

Plot of the transformed Values:

The maximum variance from the mean value is restricted in the interval [3.005, 3.009]. Hence, the overall change in variance is restricted in the interval [3.005, 3.009].



The above graph is rescaled and shown below (variability in variance is very small):



After applying the above mentioned transformation, the series is converted to a stationary series (constant mean, almost constant variance, no seasonality).

2)

a) This part is answered in the code section.

b) The differences between the original and the predicted values have been answered as a part of the code section.

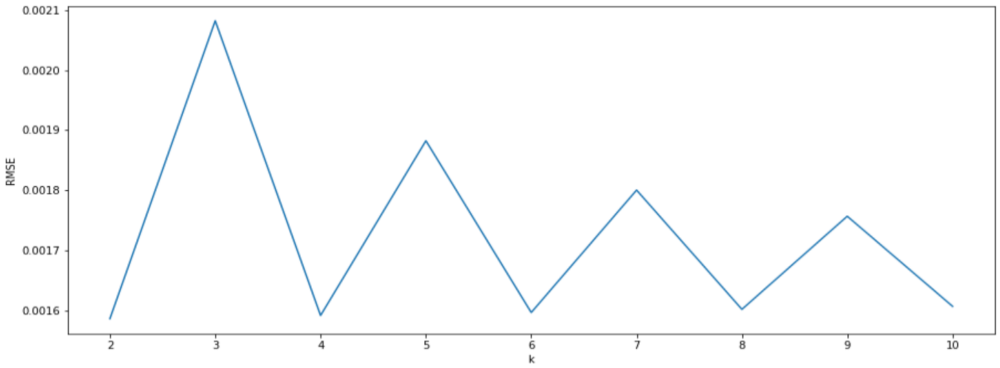
Minimum RMSE obtained for k = 2

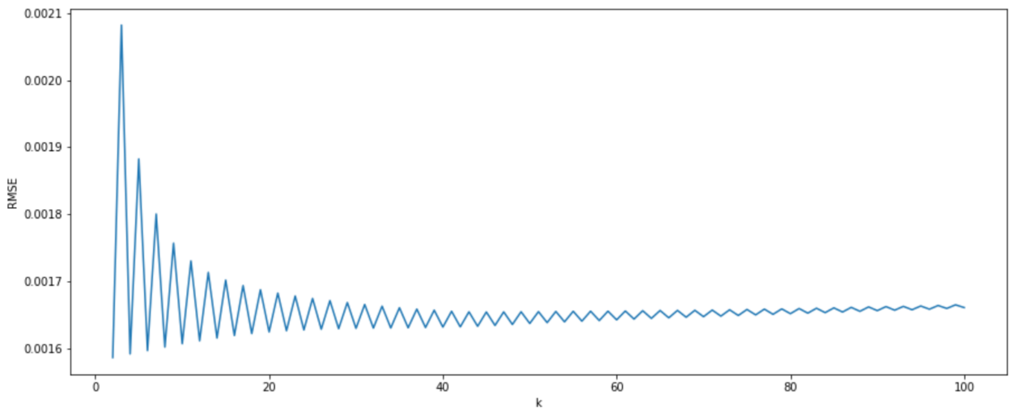
RMSE (for k = 2) = 0.0015858571532809472

c) This part is also as a part of the code. Below are the values of RMSE for k in 2 to 10

```
Value of k:  2

Value of RMSE:  0.0015858571532809472

Value of k:  3

Value of RMSE:  0.0020821808809422812

Value of k:  4

Value of RMSE:  0.0015912701579573613

Value of k:  5

Value of RMSE:  0.001882308111326717

Value of k:  6

Value of RMSE:  0.0015961846425864303

Value of k:  7

Value of RMSE:  0.0018001188561949848

Value of k:  8

Value of RMSE:  0.0016014913019771883

Value of k:  9

Value of RMSE:  0.0017565474029988457

Value of k:  10

Value of RMSE:  0.0016064744465964986
```

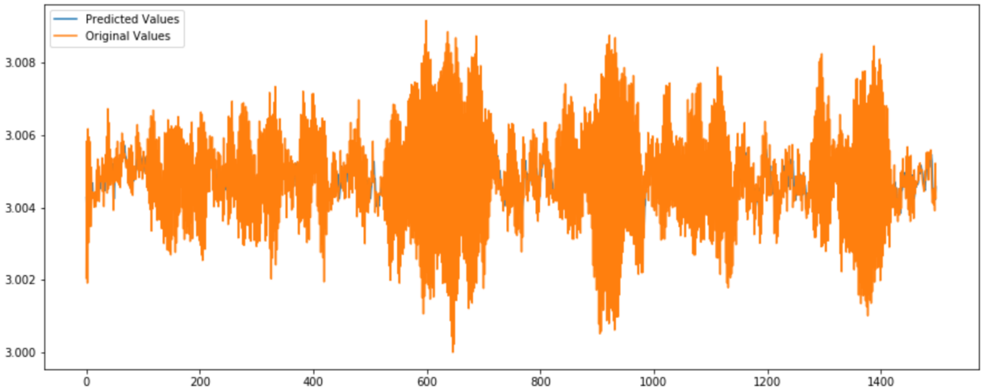
d) RMSE vs k plot:





k=2 based on the lowest value of RMSE (0.0015858571532809472)

Below is the plot of Original Values vs Predicted Values for k = 2



e) The above graphs show that the predicted values closely follow the original values with some amount of lag given by k (above for k = 2). As the value of k increases, there is a smoothing effect over a number of previous observations and hence the value of the RMSE decreases continuously.

3)

a) This part is answered in the code section.

b & c) The differences between the original and the predicted values have been answered as a part of the code section.

RMSE values for different values of a are as follows:

For a = 0.1, RMSE: 0.17796484981505029

For a = 0.2, RMSE: 0.12928571305718564

For a = 0.3, RMSE: 0.10862273819980812

For a = 0.4, RMSE: 0.0969677005892778

For a = 0.5, RMSE: 0.08957834654887102

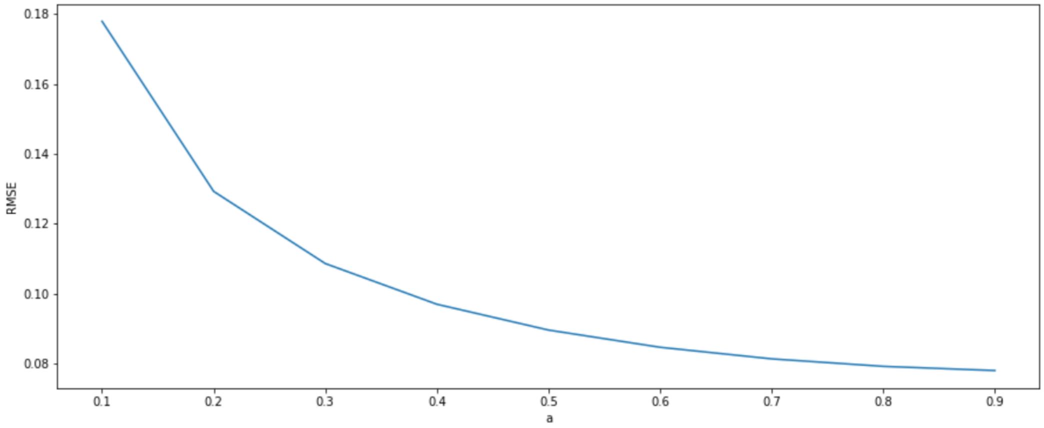
For a = 0.6, RMSE: 0.08464757053775081

For a = 0.7, RMSE: 0.08133113988192546

For a = 0.8, RMSE: 0.07918955825284825

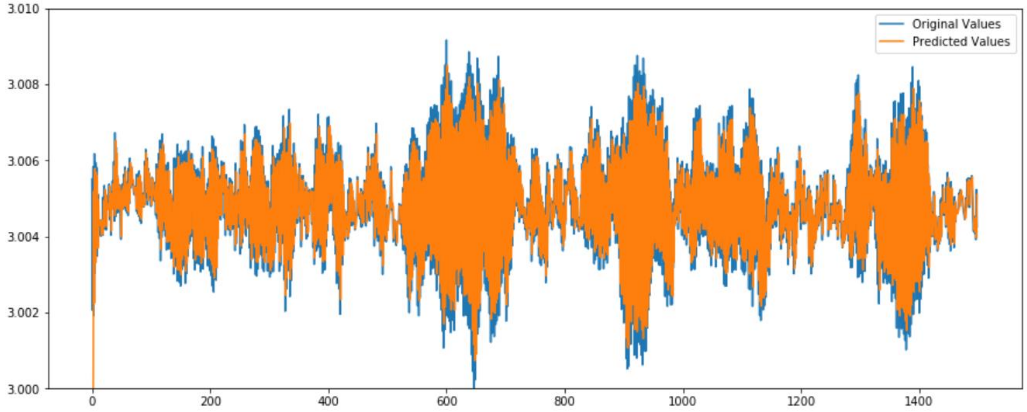
For a = 0.9, RMSE: 0.07798562437742139

d) a = 0.9 has the lowest RMSE = 0.07798562437742139

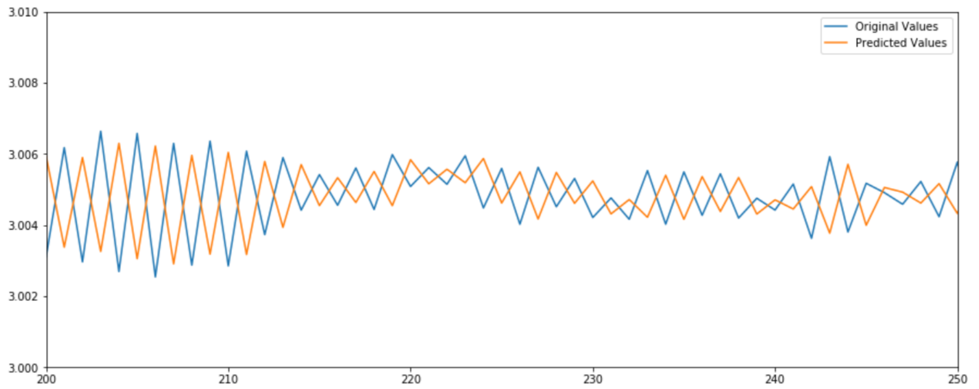


Above is the graph of a vs RMSE where a varies from 0.1 to 0.9

e) For a=0.9, below is the graph of predicted vs original values



Below graph shows taking only 50 observations:

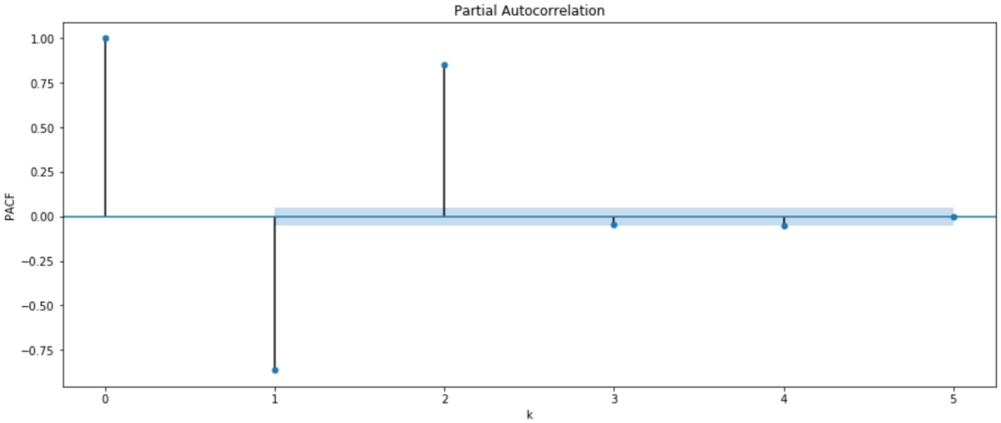


The model is quite accurate because the predicted values closely follow the observed values.

f) Above graph of RMSE vs a indicates that the error decreases with the increase in the value of a ie. the coefficient of the immediate preceding term. This is because there is a greater chance that in real world, the next value is similar to the previous value with a slight increase or decrease. For example, taking the case of recording mean temperatures of all the months. There is a high possibility that the mean temperature of the month of January has values close to the month of December or February compared to those of the month of June or July. Hence, more weight to the immediate preceding value will reduce the error to the maximum.

4)

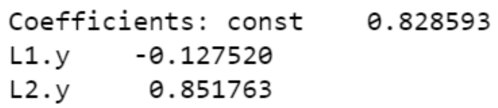
a) Partial Autocorrelation Function vs k



Order p of the AR(p) model = to the lag prior to getting the zero coefficient.

For k >= 3, value of PACF < 0.15, hence selecting p = 2

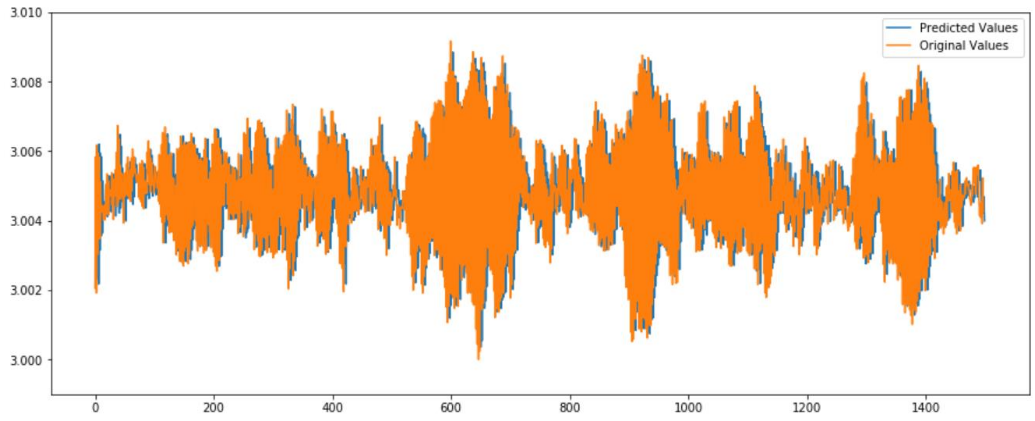
b) Parameters of the AR(p) model provided as a part of the code. Below is the screenshot of the estimated parameters:



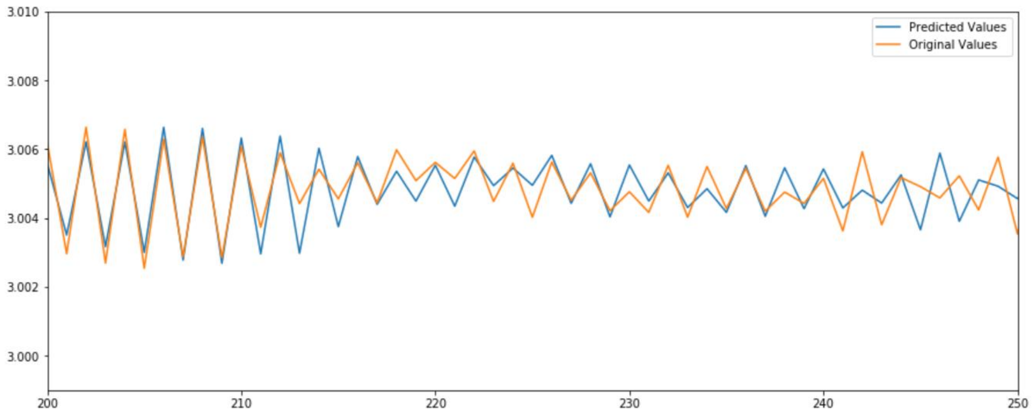
```
Coefficients: const    0.828593
L1.y    -0.127520
L2.y     0.851763
```

RMSE Value: 1.5618014452735913

Plot of Predicted values against the original values:

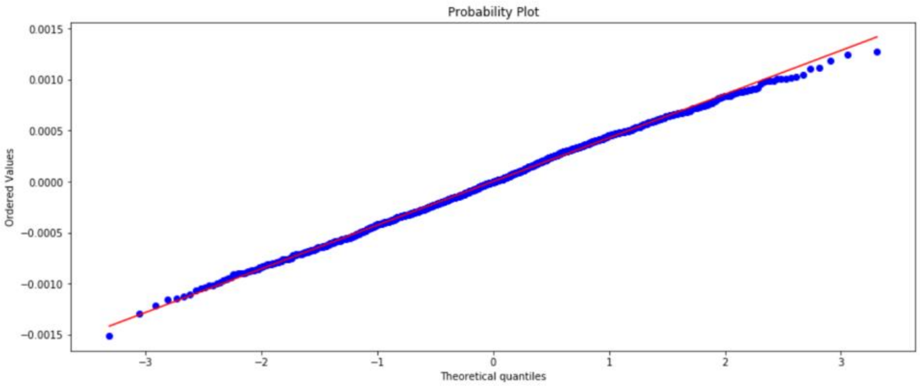


Zooming in to a few values:

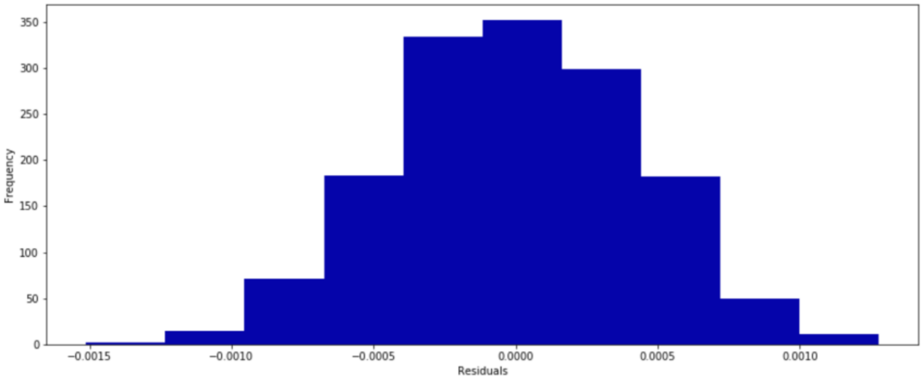


c)

c.a) QQ Plot of the pdf of the residuals vs N(0, s^2)



Histogram Plot of the Residuals



Performing the Chi Square Test on the residuals gives the following results:

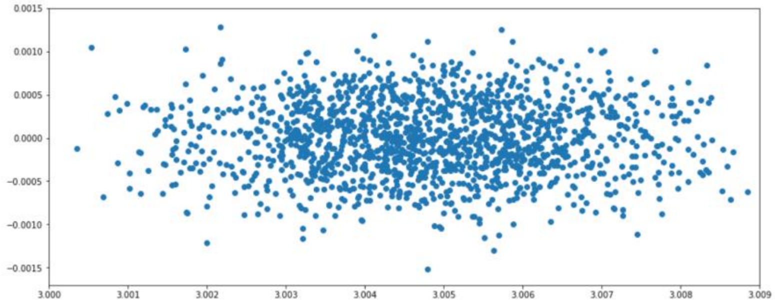
Chi-Square Obtained: 14.60347129506008

Chi-Square Critical: 15.50731305586545

p-value: 0.06733006574232758

Since the obtained p-value > 0.05, we conclude that the residuals follow a normal distribution (also evident from the histogram plot)

c.b) Scatter Plot of the Residuals



d) The predicted values very closely follow the original observations and hence this model gives the minimum RMSE. Moreover, the residuals are Normally Distributed with no correlation trends.

5)

On testing different models on the test data, below are the observations:

Simple Moving Average Model:

Train Error: 0.0015858571532809472

Test RMSE: 0.001029777660909734

Exponential Smoothing Model:

Train RMSE: 0.07798562437742139

Test RMSE: 0.13510516096092845

AR(p) Model:

Train Error: 0.0004275190336966159

Test Error: 0.0010178027404789552

Based on the RMSE values of the Test Data, I will choose the AR(p) model with p = 2 since it has the lowest RMSE on the test data. Also, it has a very low RMSE on the training data.

Simple Moving Average Model is also a good model in this case because of its low Test Error and very similar Test and Train errors indicating that the model in general is a good fit and not an overfit (ie. it doesn’t result in unexpectedly higher errors on the Test dataset).