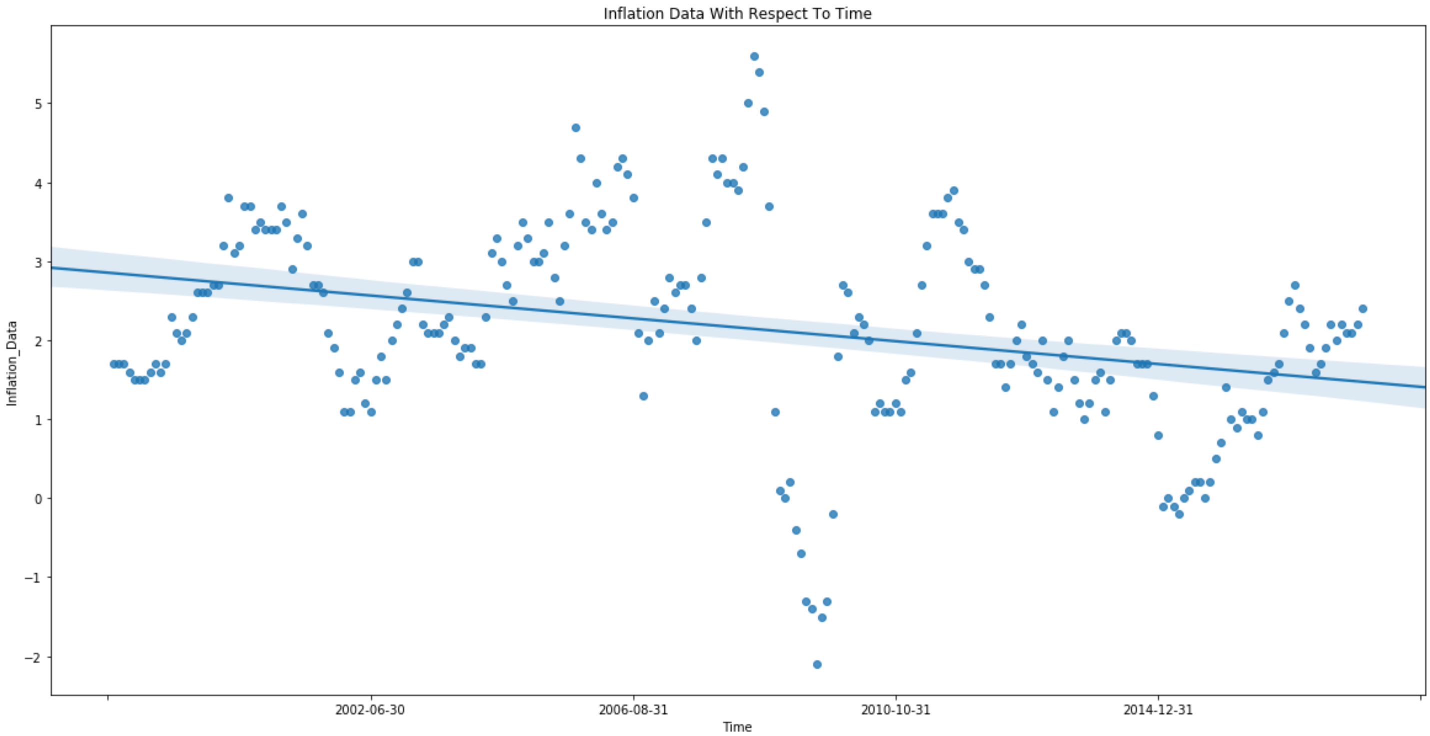
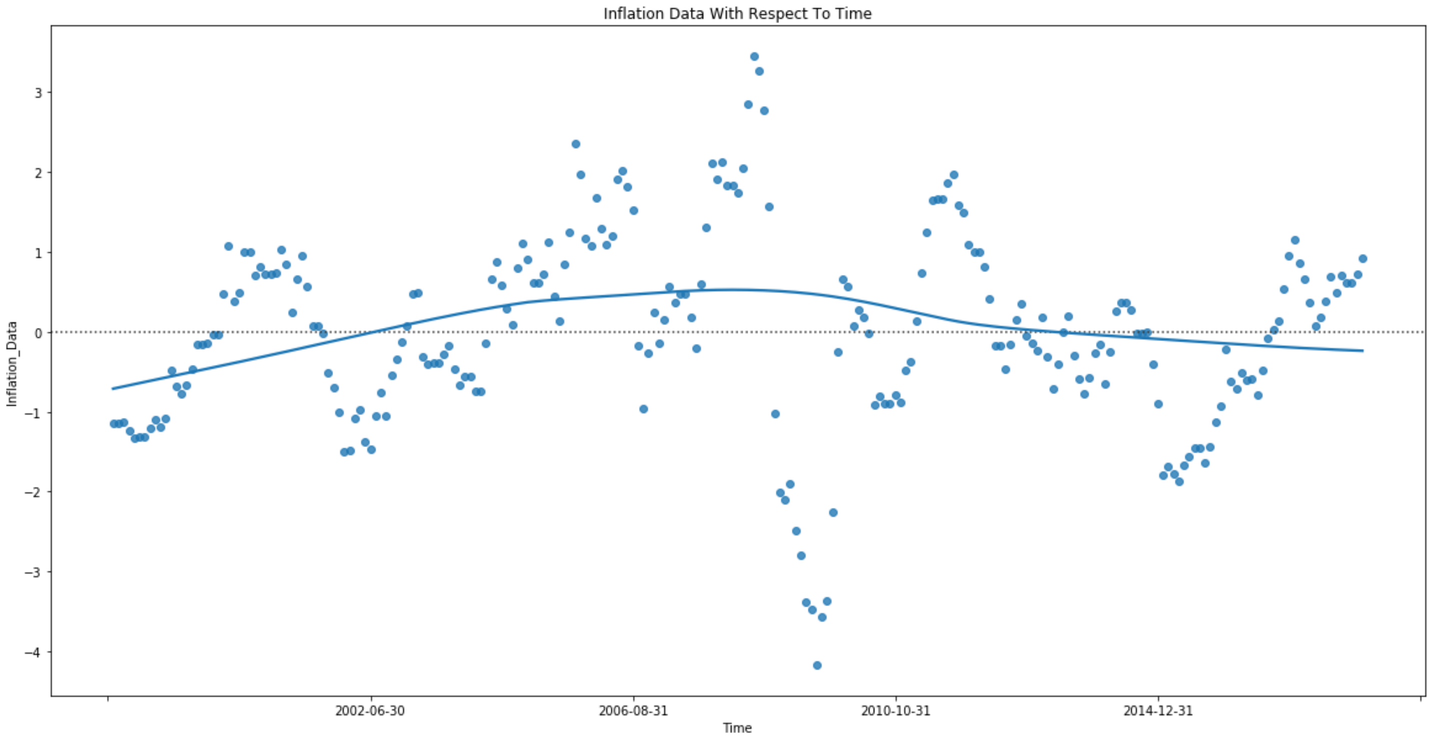
FRE 6351 Final Project
Written by Po-Hsuan (Michael) Lin

Data set selection:

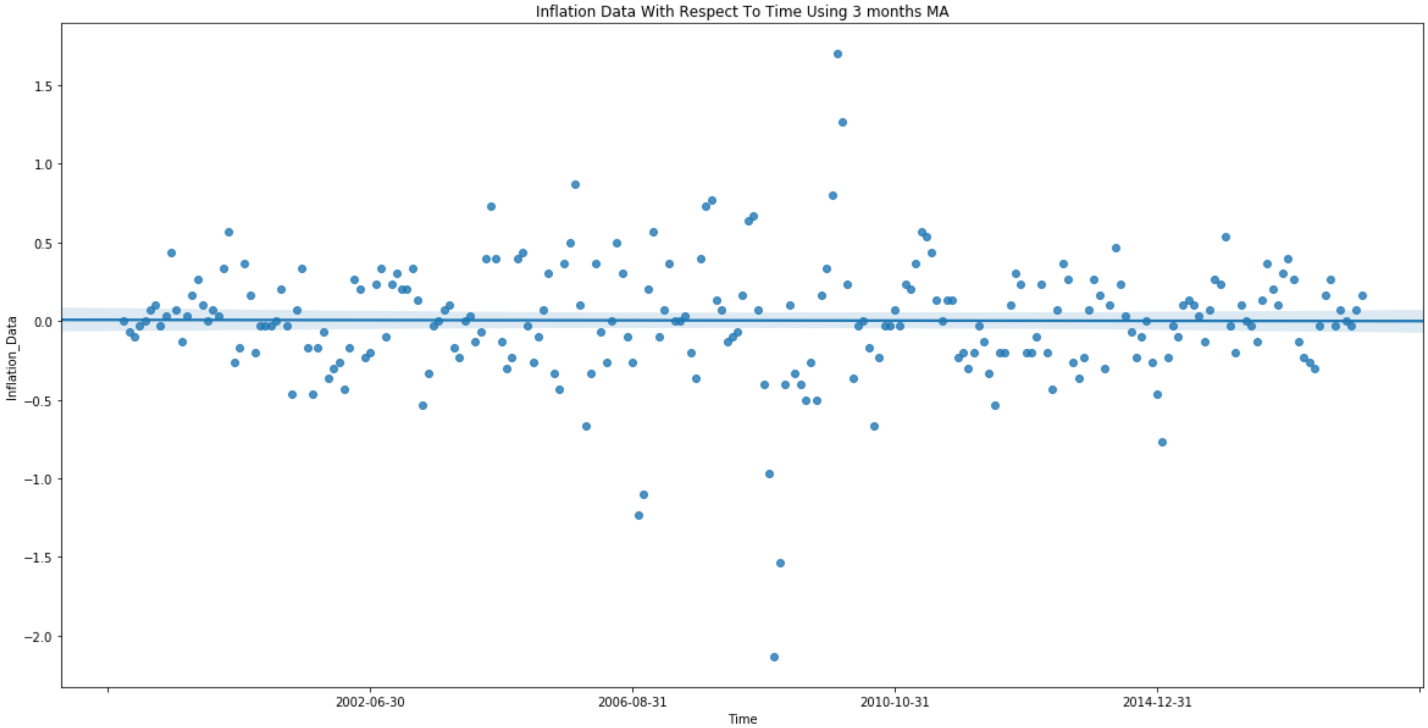
Inflation Data US CPI Urban Consumers (CPI YOY)
05/31/1998 ~ 03/31/2018

Inflation data with respect to time:



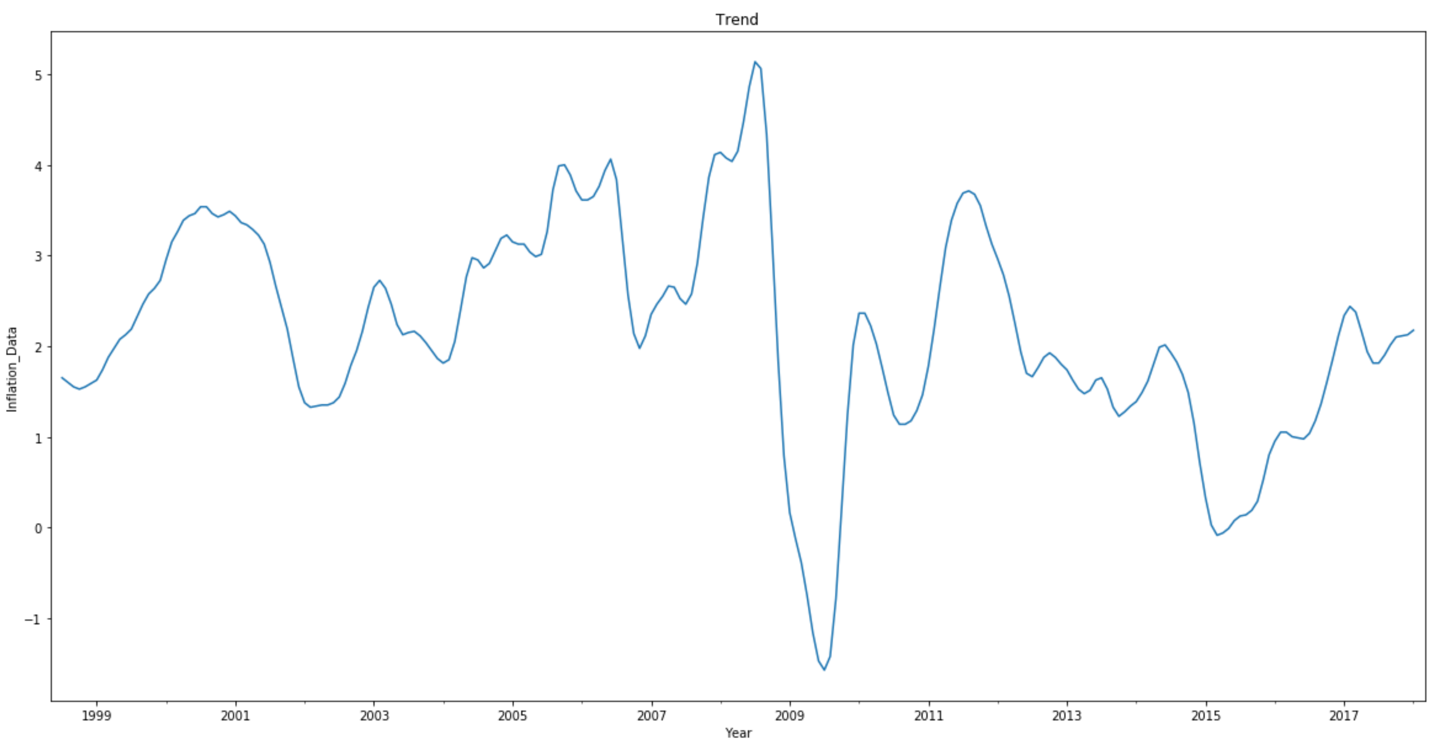


Manually detrend using 3 months moving average

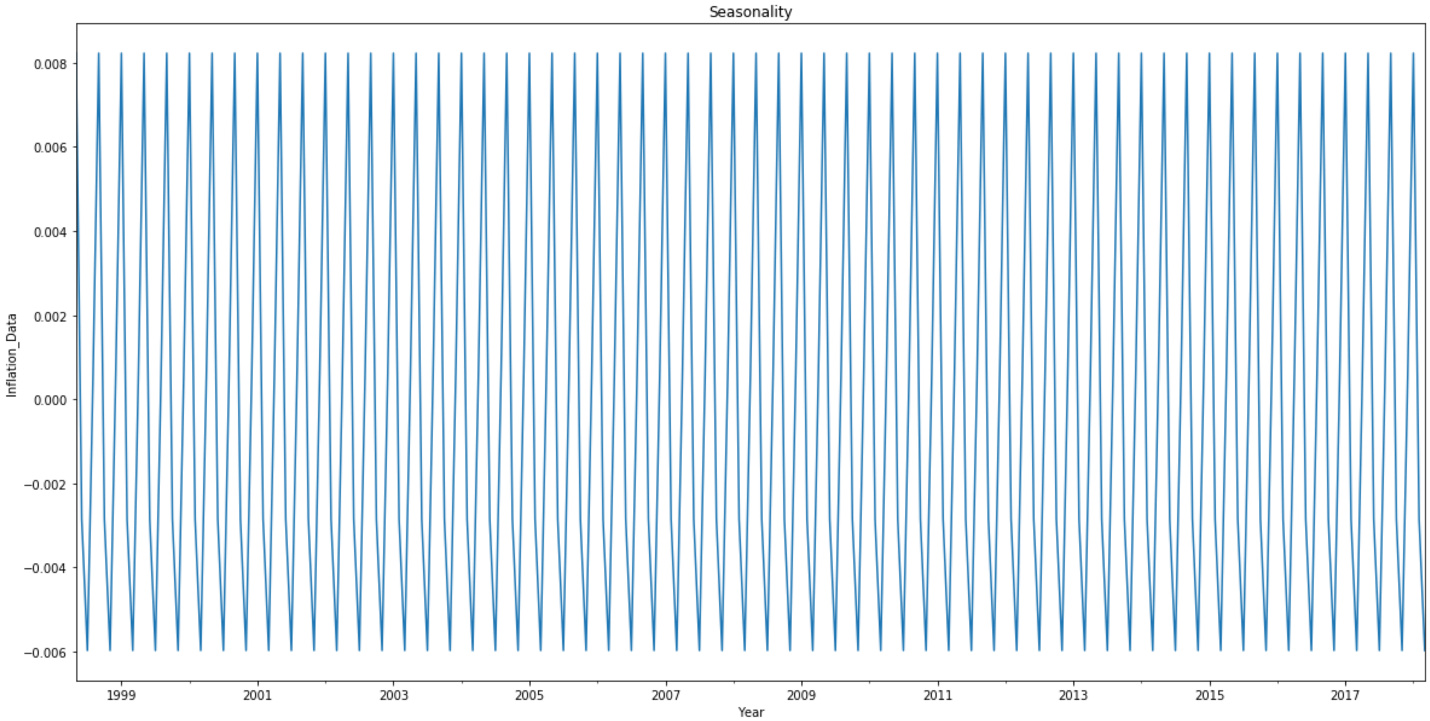


Detrend using built-in decomposition function

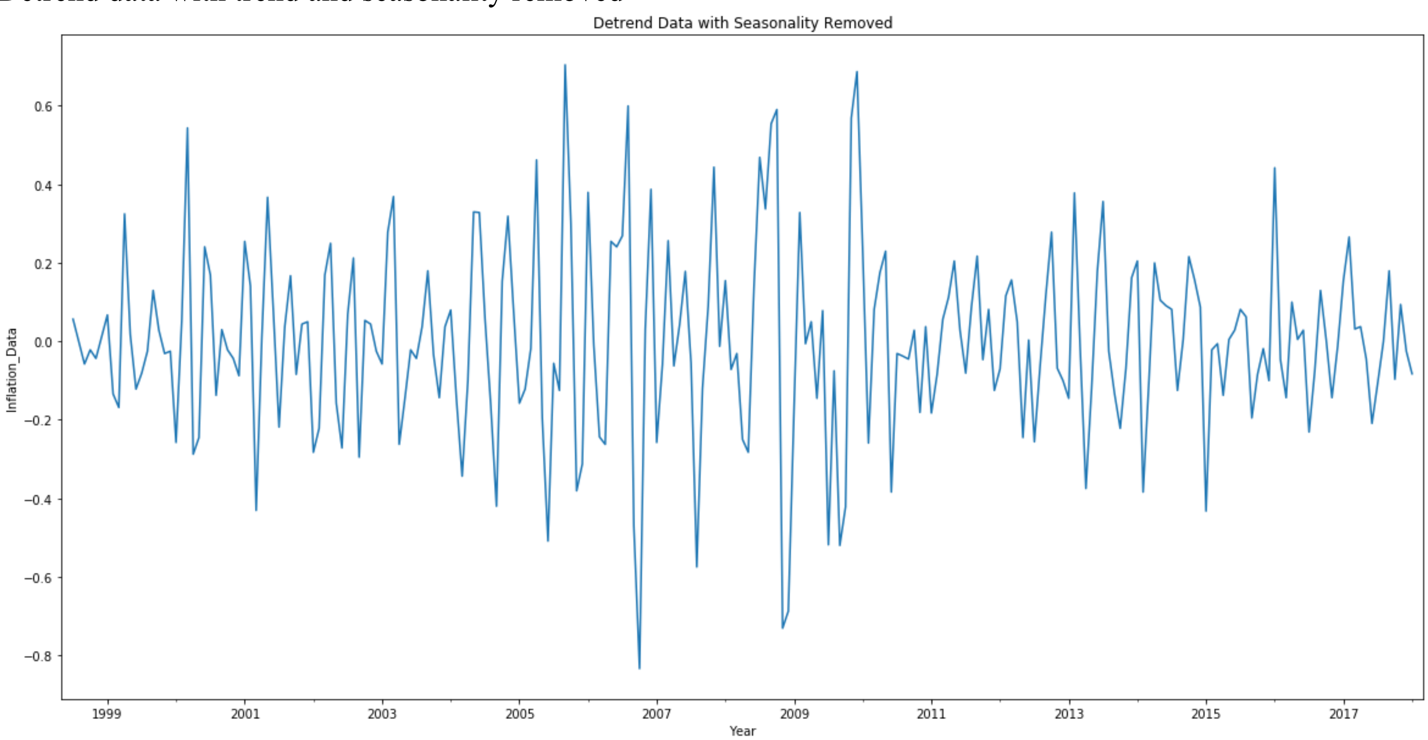
Trend



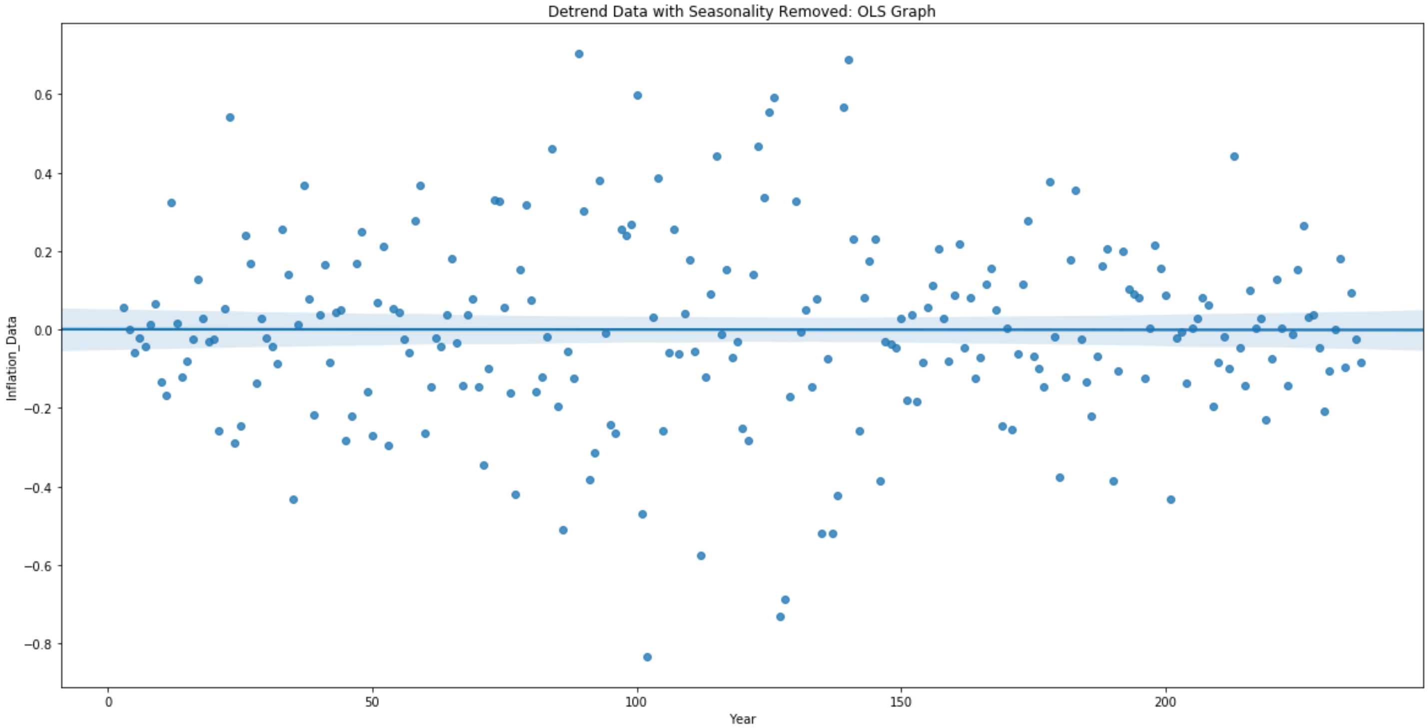
Seasonality

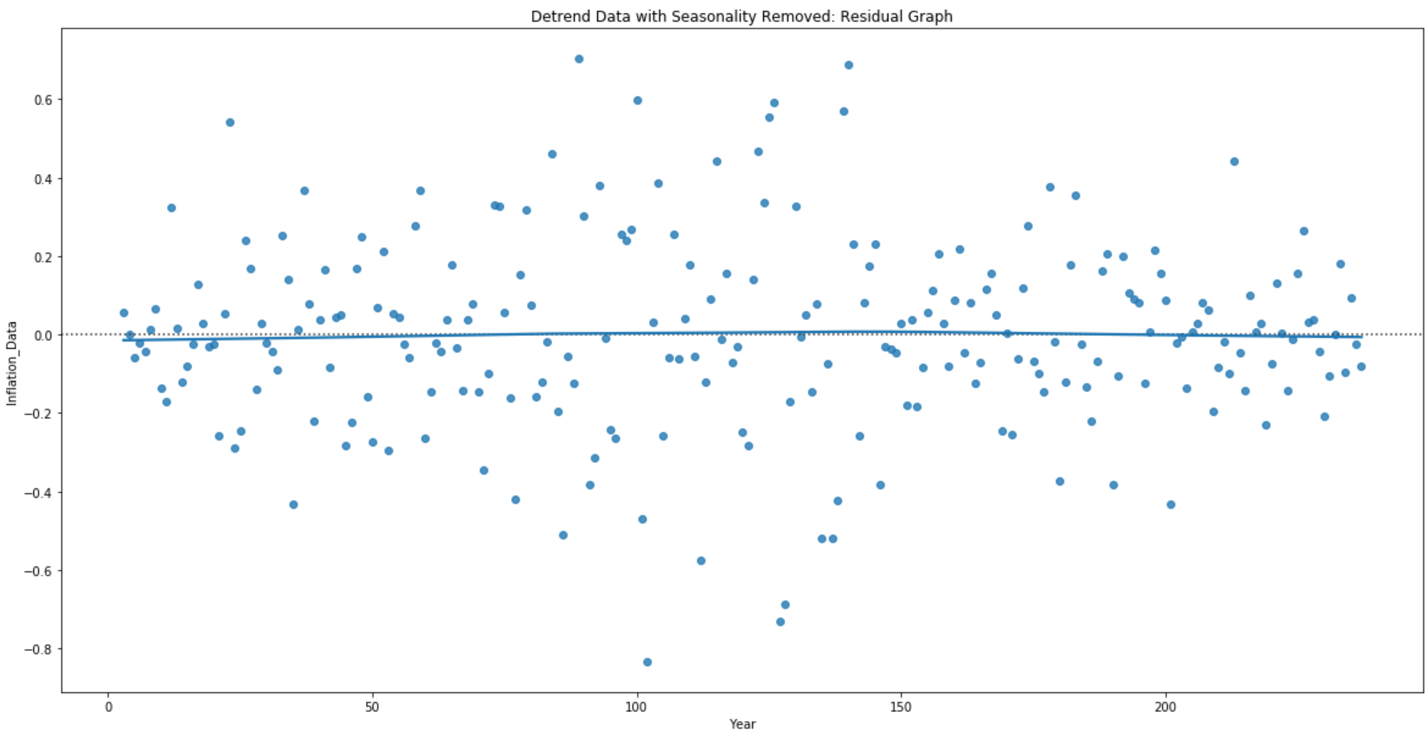


Detrend data with trend and seasonality removed



Graph detrend inflation data and residuals with respect to time





Check stationary using Dickey-Fuller Test

```
Results of Augmented Dickey-Fuller Test:
Augmented Dickey-Fuller Statistic:  -9
p-value:0.000000
Crititcal Values:
        1%:-3.4603
        5%:-2.8747
        10%:-2.5738
```

1% chance of being not stationary corresponds to a Dickey-Fuller value of -3.4603. We have a Dickey-Fuller value of -9 which implies that our model is >99% stationary; <1% probability of it not being stationary.

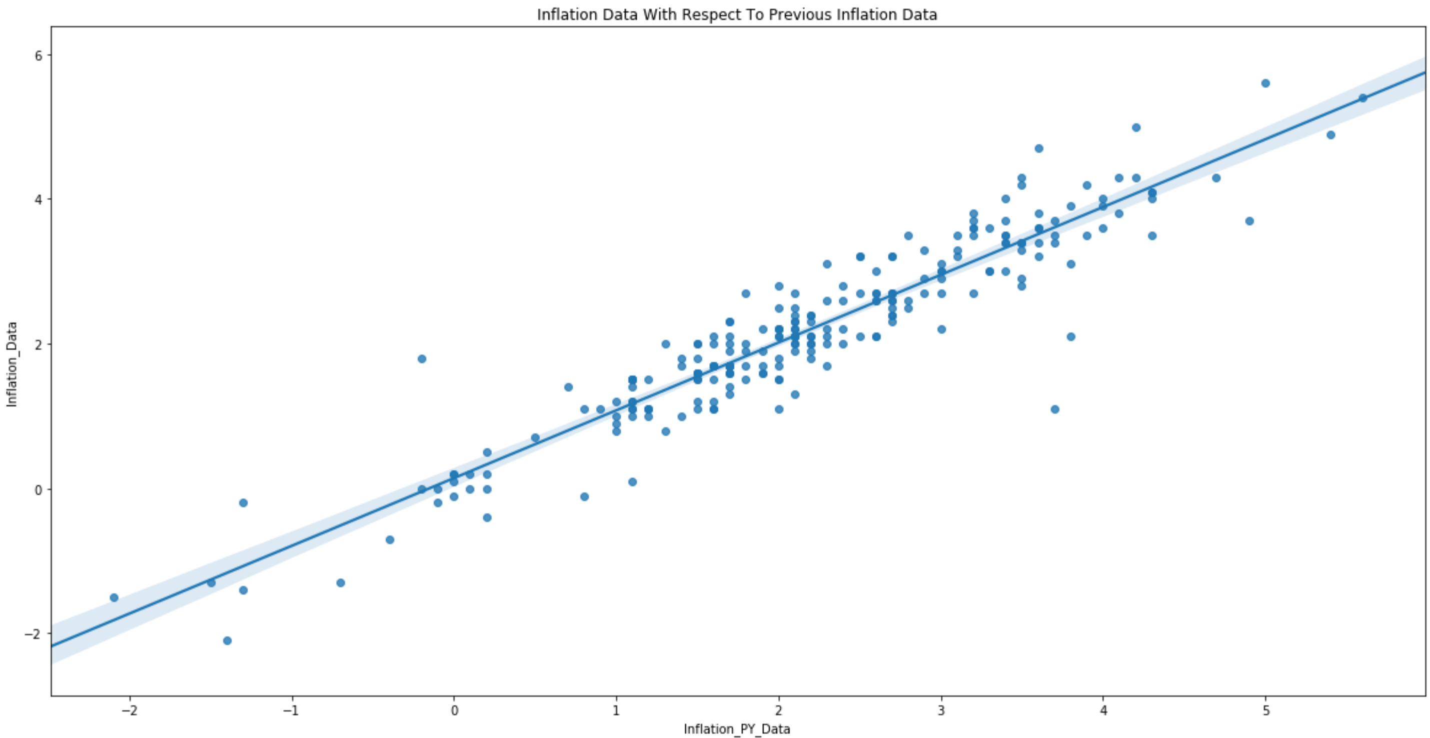
Model selection:

We remove the first 35 data points since these data points are dated back in 1980s and early 1990s which has a lower significance to the overall model and might skew our forecast. Then, we separate our data into train and test sets. Train sets have the first 150 values while test sets have the remaining 50 values.

I. AR(1,q) model:

Intuition:

While playing with the original data, we discover that the inflation data has a strong linear relationship with previous year's inflation data. This makes a lot of sense from the economics standpoint. Economics condition and business cycle tends to be slow to react, hence it has a strong correlation with last year's correlation.



This leads to our first model which is a ARMA(1,q) model. To find the proper number of moving average terms, we fit the train data into ARMA(1,q) model; q from 0 to 10. We use AIC to select the model as AIC combines the fit of the model with the simplicity of the model. If we simply uses the fit of the model as the criterion, then it will most likely be ARMA(1,10). The model will be overly rigid and not optimal for forecasting.

The smaller the AIC, the better the model in terms of fit and simplicity. Therefore, we choose ARMA(1,2) as the optimal model.

ARMA	(1,0)	(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)	(1,8)	(1,9)	(1,10)
AIC	33.24	23.76	15.11	23.91	22.34	22.75	22.86	30.59	24.57	39.26	37.26

To determine the coefficients of the ARMA model, we use the following algorithms:
ARIMA default: conditional sum-of-squares to find starting values, then maximum likelihood

ML	ARMA(1,2)
phi	0.4262
theta 1	-0.4525
theta 2	-0.5475
AIC	15.12
Sigma Squared	0.04821

Hannan-Rissanen Algorithm

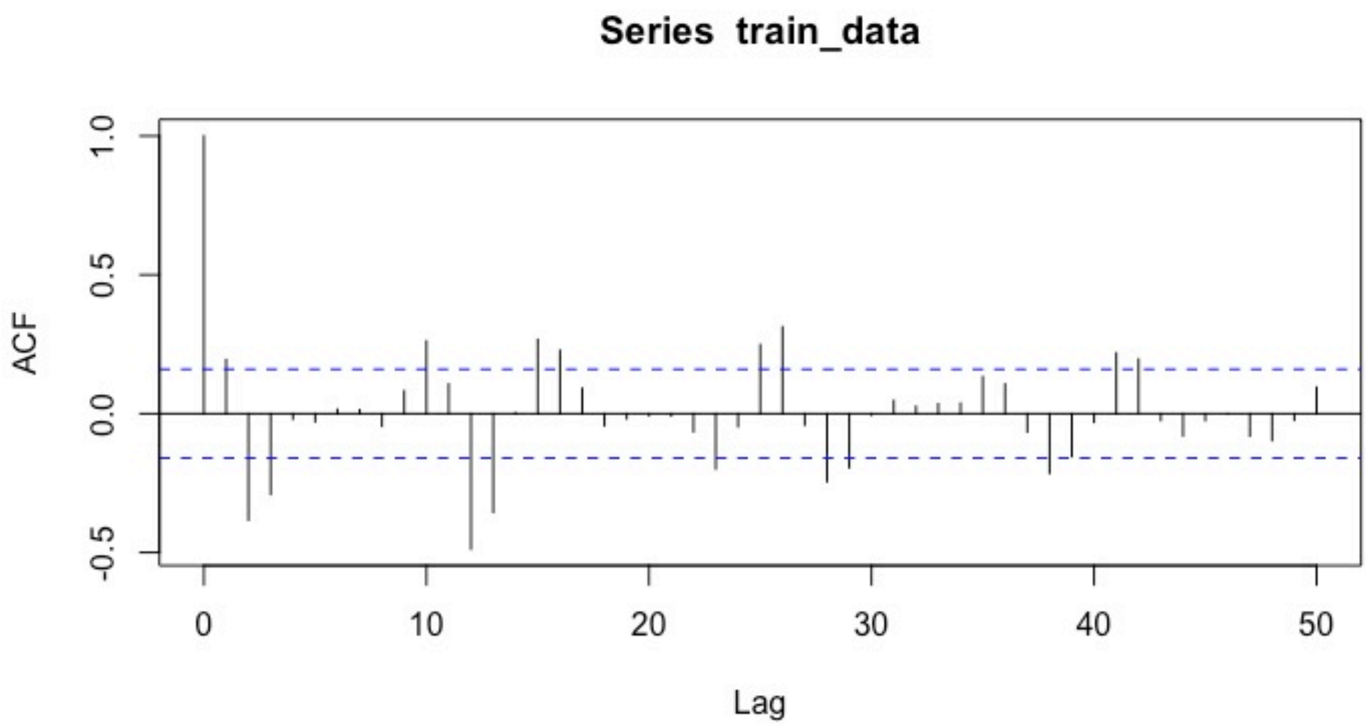
HR	ARMA(1,2)
phi	0.3513
theta 1	-0.2801
theta 2	-0.5373
AICc	6.96
Sigma Squared	0.05257

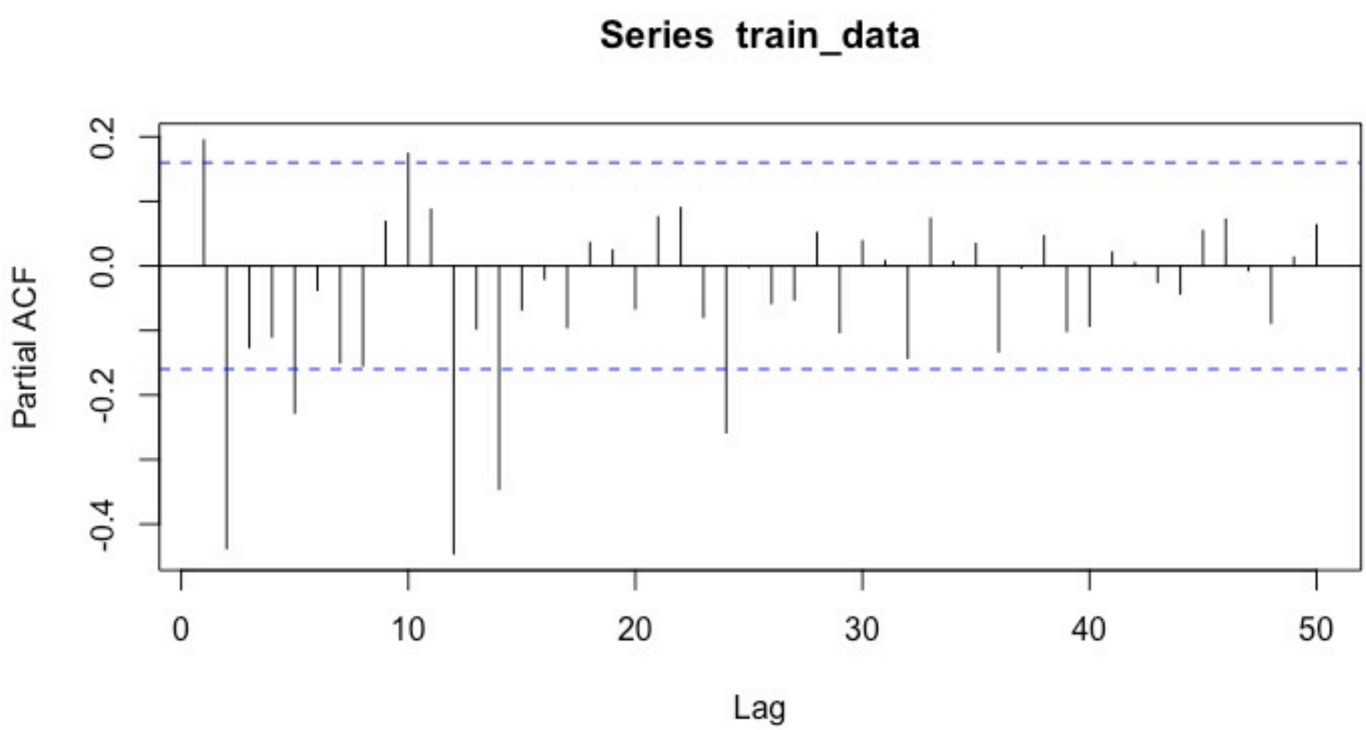
Innovations Algorithm with recursion = 17

$$\psi_1 = \theta_1 + \phi_1$$
$$\psi_2 = \theta_2 + \phi_1\psi_1$$
$$\psi_3 = \phi_1\psi_2$$

Innovations	MA(0,3)	ARMA(1,2)
phi	-	0.5832
theta 1	0.06719	-0.5160
theta 2	-0.5478	-0.5870
theta 3	-0.3195	-
AICc	16.91	-
Sigma Squared	0.04909	-

II. ARMA(p,q) model:
We can examine the ACF and PACF function of the train data set. Both graphs don't have distinct cut off that points to either AR(p) or MA(q) models. With both ACF and PACF tailing off, it points to an ARMA model. For this model, we decide to use the autofit function.





The autofit function with a limit of [0,5] on both p and q yields ARMA(2,5).

To determine the coefficients of the ARMA model, we use the following algorithms:
Default Autofit: conditional sum-of-squares to find starting values, then maximum likelihood

ML	ARMA(2,5)
phi 1	-1.3631
phi 2	-0.8023
theta 1	1.5842
theta 2	0.5171
theta 3	-1.2266
theta 4	-1.2991
theta 5	-0.5752
AIC	25.86
Sigma Squared	0.04115

Hannan-Rissanen Algorithm

HR	ARMA(2,5)
phi 1	0.4625
phi 2	-0.4677
theta 1	-0.3953
theta 2	-0.0707
theta 3	-0.08918
theta 4	0.007831
theta 5	-0.2576
AICc	8.95
Sigma Squared	0.04880

III. Pure MA model:
We run arima model fit with p in range of 0 to 5 and q in range 0 to 10 and select the one with the smallest AIC. The model that has the smallest AIC is ARMA(0,2).

AIC(p,q)	0	1	2	3	4	5	6	7	8	9	10
0	37.02	23.78	1.38	-24.02	-22.24	-20.80	-24.76	-22.86	-20.97	-27.39	-34.82
1	33.24	23.76	-15.12	-23.91	-22.34	-22.75	-22.86	-30.59	-24.57	-39.26	-37.26
2	3.38	-15.69	-18.57	-22.68	-20.91	-30.72	-29.04	-29.04	-32.15	-36.62	-45.68
3	3.02	-18.79	-17.09	-23.69	-32.38	-31.61	-29.75	-27.81	-32.37	-40.27	-44.45
4	3.18	-17.38	-21.66	-28.62	-24.28	-29.79	-39.23	-40.62	-40.68	-46.46	-44.66
5	-2.75	-16.10	-15.30	-26.89	-29.39	-25.93	-31.76	-29.51	-35.80	-44.67	-43.29

To determine the coefficients of the ARMA model, we use the following algorithms:
ARIMA default: conditional sum-of-squares to find starting values, then maximum likelihood

ML	MA(2)
theta 1	-0.2635
theta 2	-0.7365
AIC	1.3807
Sigma Squared	0.05429

Innovations Algorithm with recursion = 17

IA	MA(2)
theta 1	0.06720
theta 2	-0.5478
AICc	11.0811
Sigma Squared	0.06020

IV. Pure AR model: Relatively unrealistic and overfitted case

Similar to the procedure done for ARMA(1,q) model, we can compare the AIC of AR(p) models.

ARMA	(0,0)	(1,0)	(2,0)	(3,0)	(4,0)	(5,0)	(6,0)	(7,0)	(8,0)	(9,0)	(10,0)
AIC	37.02	33.24	3.380	3.019	3.183	2.754	0.912	2.260	4.103	2.795	5.646

To determine the coefficients of the ARMA model, we use the following algorithms:

ARIMA default: conditional sum-of-squares to find starting values, then maximum likelihood

ML	AR(6)
phi 1	0.1774
phi 2	-0.4708
phi 3	-0.2085
phi 4	-0.07640
phi 5	-0.2200
phi 6	-0.03233
AIC	0.9124
Sigma Squared	0.05202

Yule-Walker Algorithm

YW	AR(6)
phi 1	0.1774
phi 2	-0.4721
phi 3	-0.2110
phi 4	-0.07938
phi 5	-0.2210
phi 6	-0.03748
AICc	2.1017
Sigma Squared	0.05202

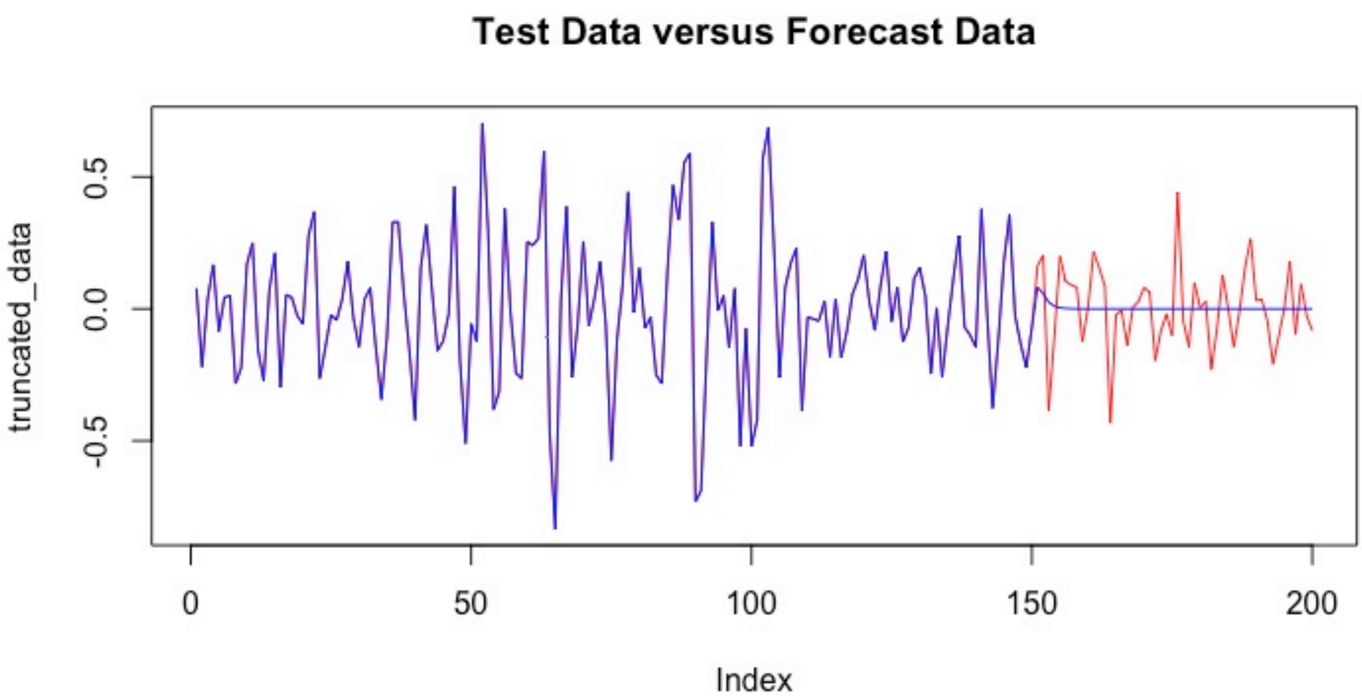
Burg Algorithm

B	AR(6)
phi 1	0.1777
phi 2	-0.4752
phi 3	-0.2115
phi 4	-0.07864
phi 5	-0.2245
phi 6	-0.03306
AICc	2.0994
Sigma Squared	0.05202

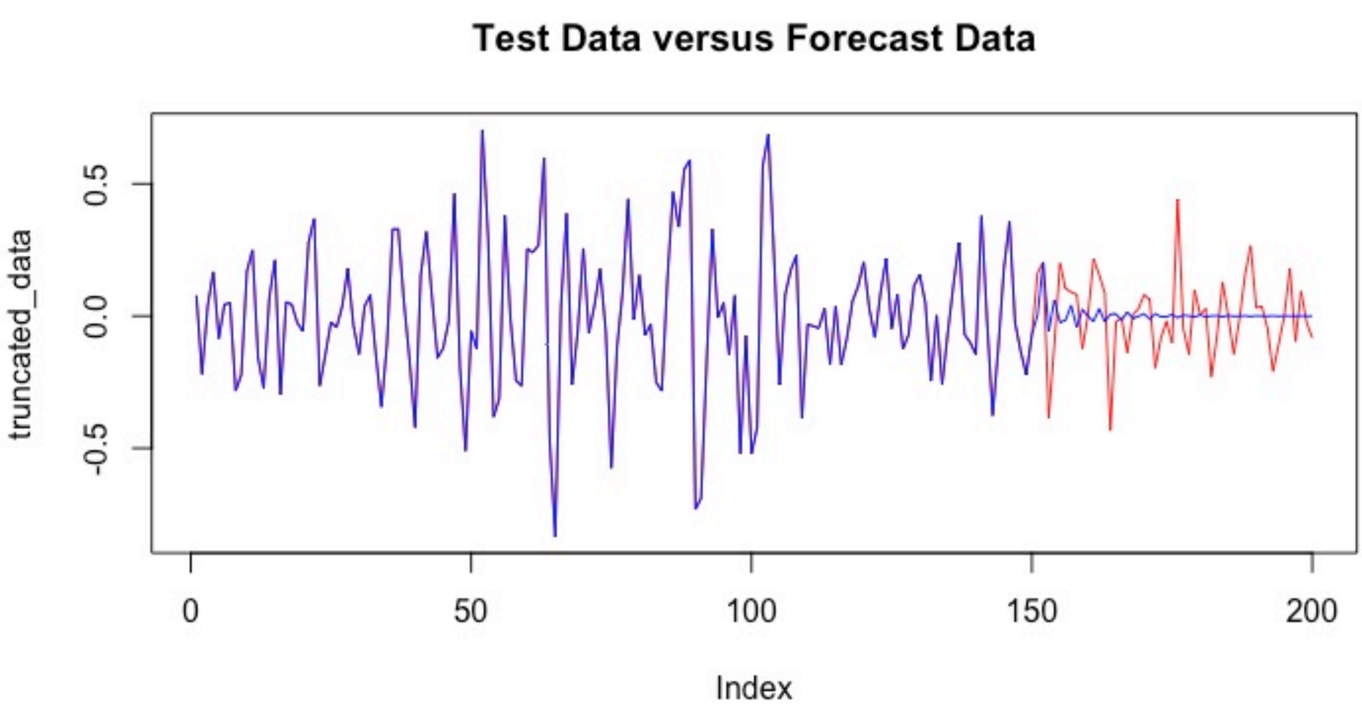
Forecasting with Seasonality Removed:

For forecasting, since we separate the data sets into two and we fit the train portion to four different models: ARMA(1,2), ARMA(2,5), MA(2) and AR(6). Using the model, we can forecast fifty data points. Due to the lack of seasonality, the forecast quickly levels out.

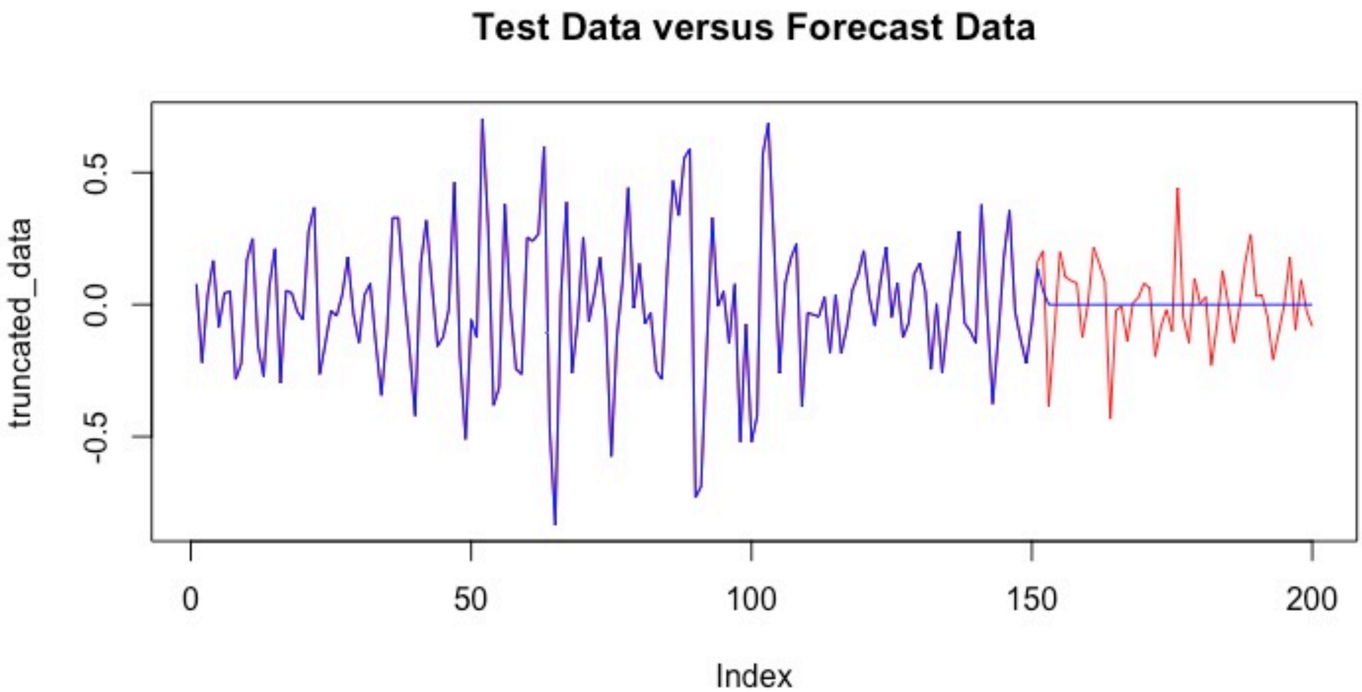
ARMA(1,2) Model:



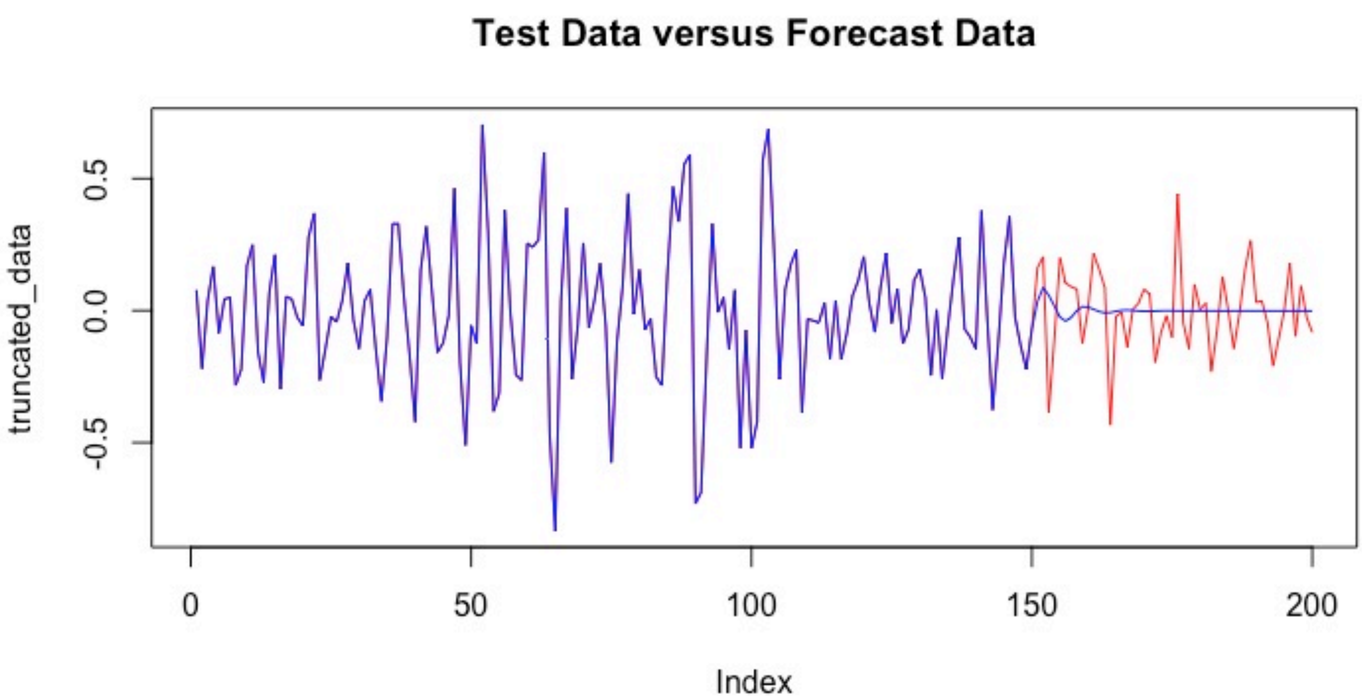
ARMA(2,5) Model:



MA(2) Model:



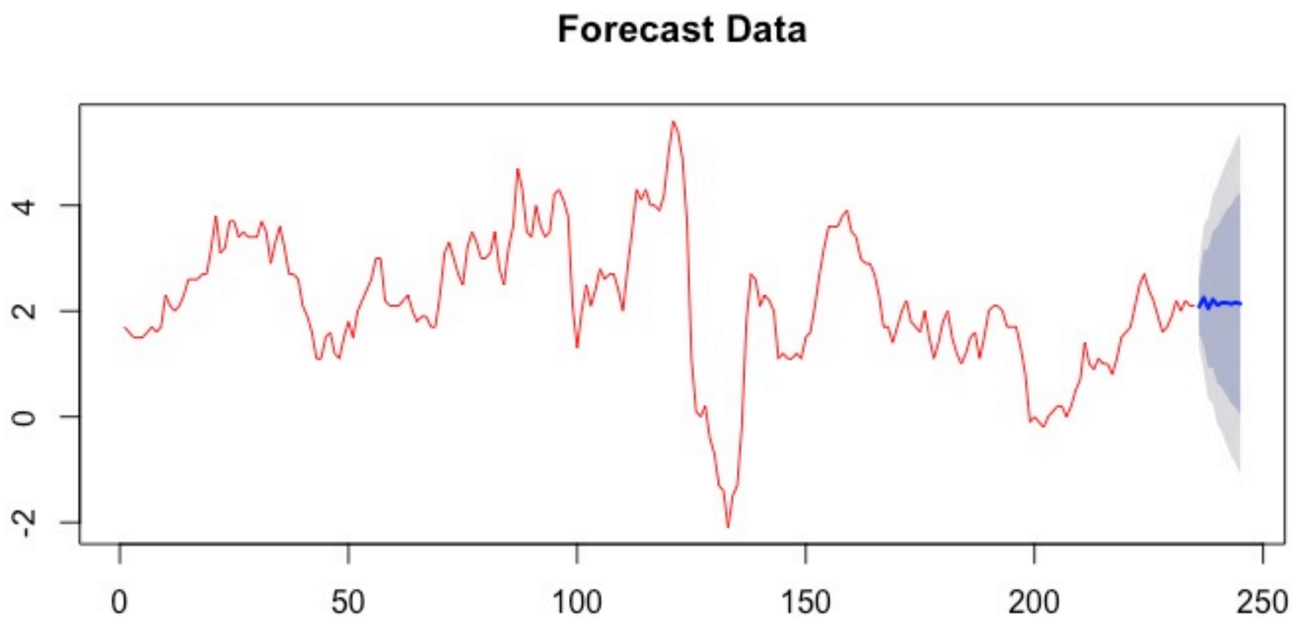
AR(6) Model:



Forecasting with Seasonality:

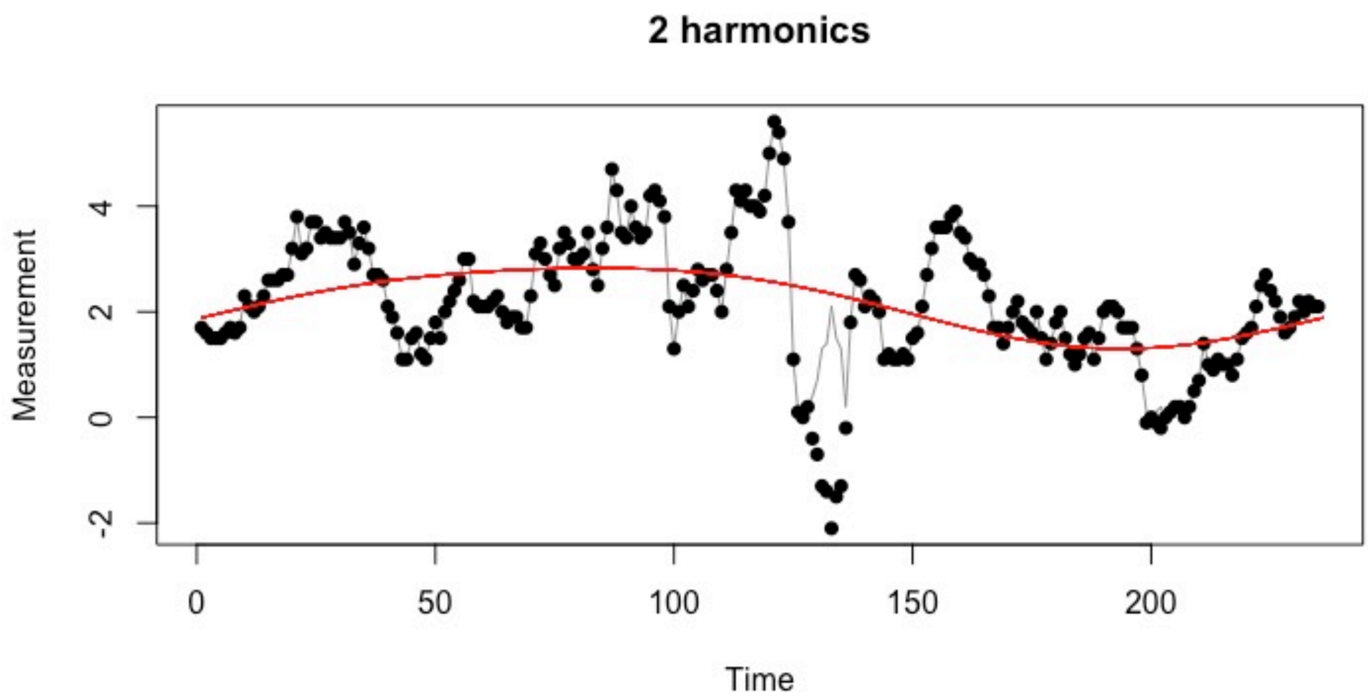
To forecast with the original seasonality, we need to look at the data before detrending. Then using auto.arima, we get a ARIMA(2,1,2) model with the following coefficents.

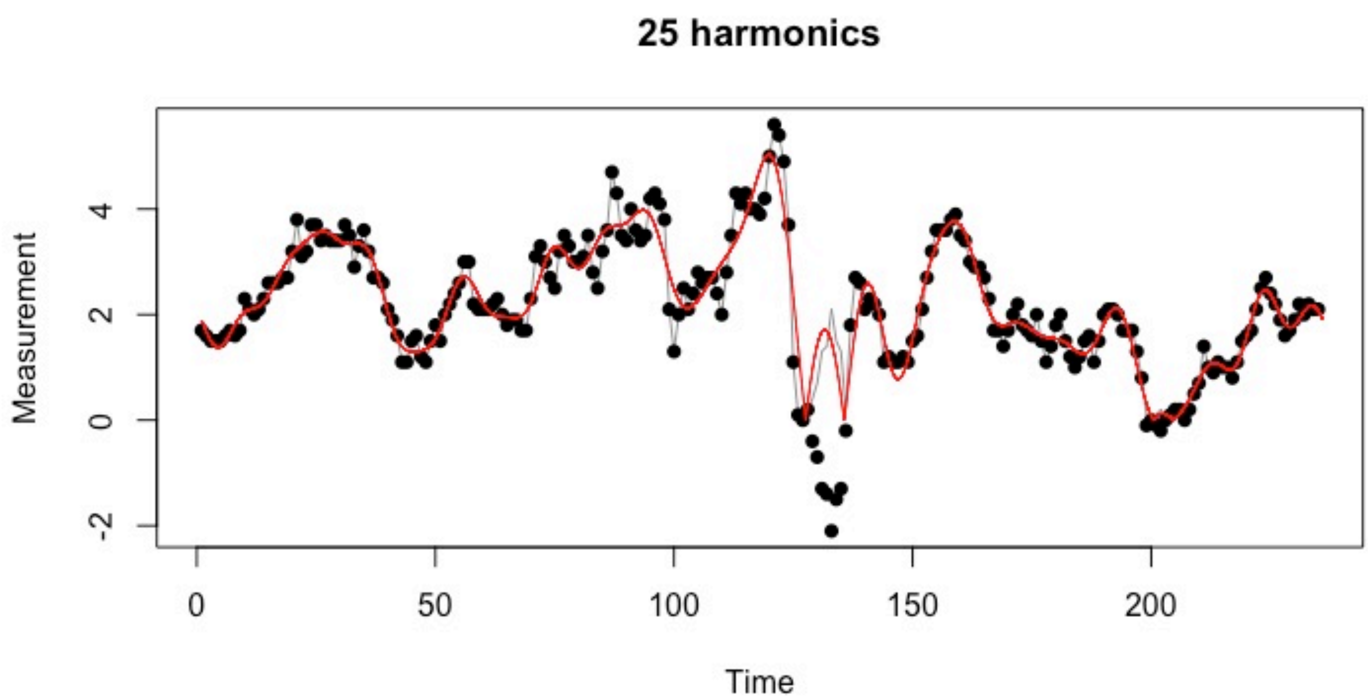
Auto ARIMA	ARMA(2,2)
phi 1	-1.2545
phi 2	-0.5205
theta 1	1.6792
theta 2	0.9494
AIC	245.13
AICc	245.39
BIC	262.41
Sigma Squared	0.161



Fourier Analysis:

To utilize Fourier Transform, we can't have negative values in the time series which will result in a mirror image in the positive realm as shown below due to the use of Mod(z) in R. Mod(z) only provides positive values. Changing to Re(z) solves the problem.





To enhance the accuracy of the harmonics to the original graph, we can shift the original value up 2%. This results in a better and much more accurate representation of the inflation data.

Since we are computing the inverse FFT on a sequence with Hermitian symmetry, you should expect a real-valued result. There will be some small imaginary numbers but these are negligible compared to the real portion, therefore we can use Re(z) to extract the values.