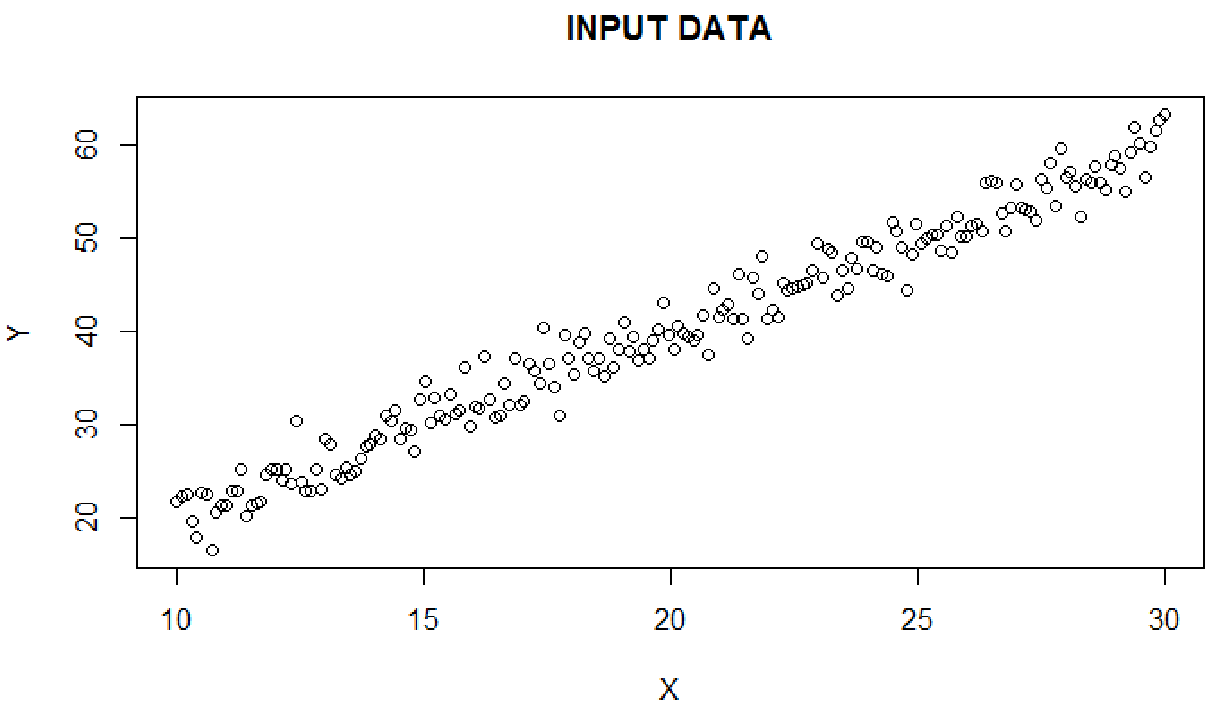
Name: Nirmal kumar Ravi

Id: A20320832

Data-Set Used

svar-set1.dat

Data Plot



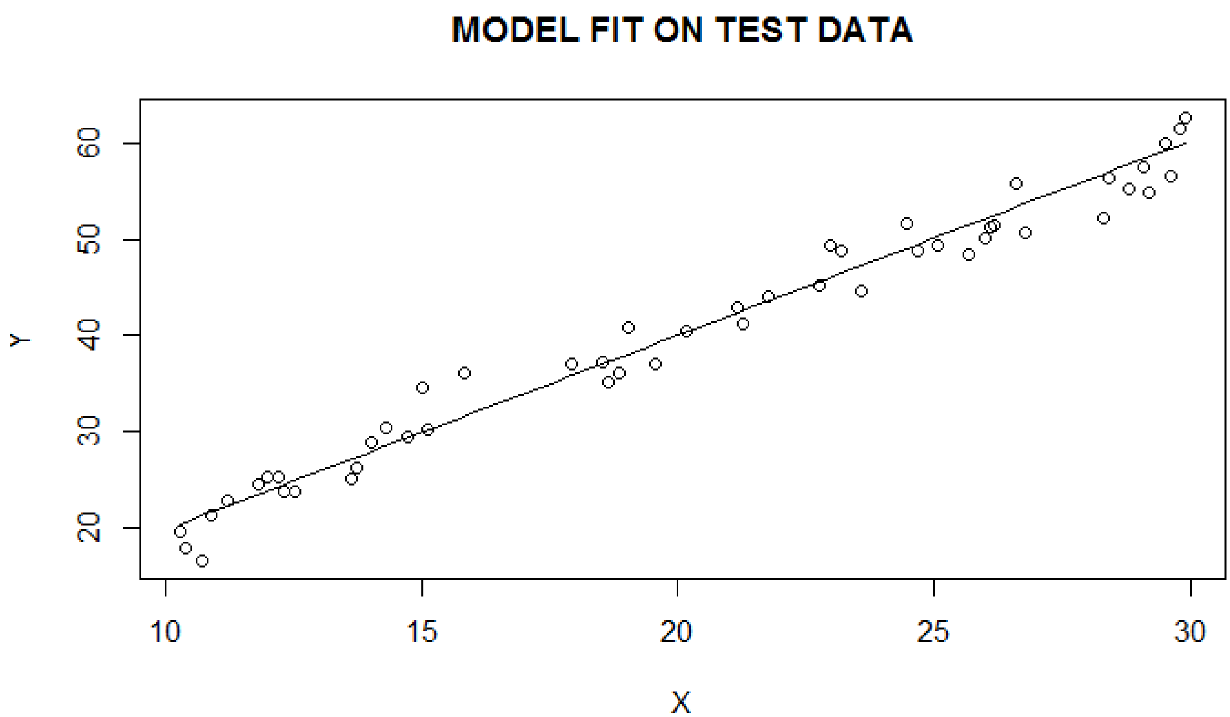
The data looks fairly linear. Let's fit linear model to it.

I have split the input data to train (75%) and test (25%) and trained the model using linear regression

Coefficients of our model fit:

0.2662876 1.9893976

Model on test Data



As we can see from the above graph our model fits well on test data

Mean squared error	
Testing set	4.63139
Training Set	4.106837

Training set error will always be less than testing set. As we use training set to build our model

Let’s try with higher order polynomials

Polynomial	Error on test set	Error on Train Set
1	4.63139	4.106837
2	4.930847	4.072704
3	4.938525	4.028656

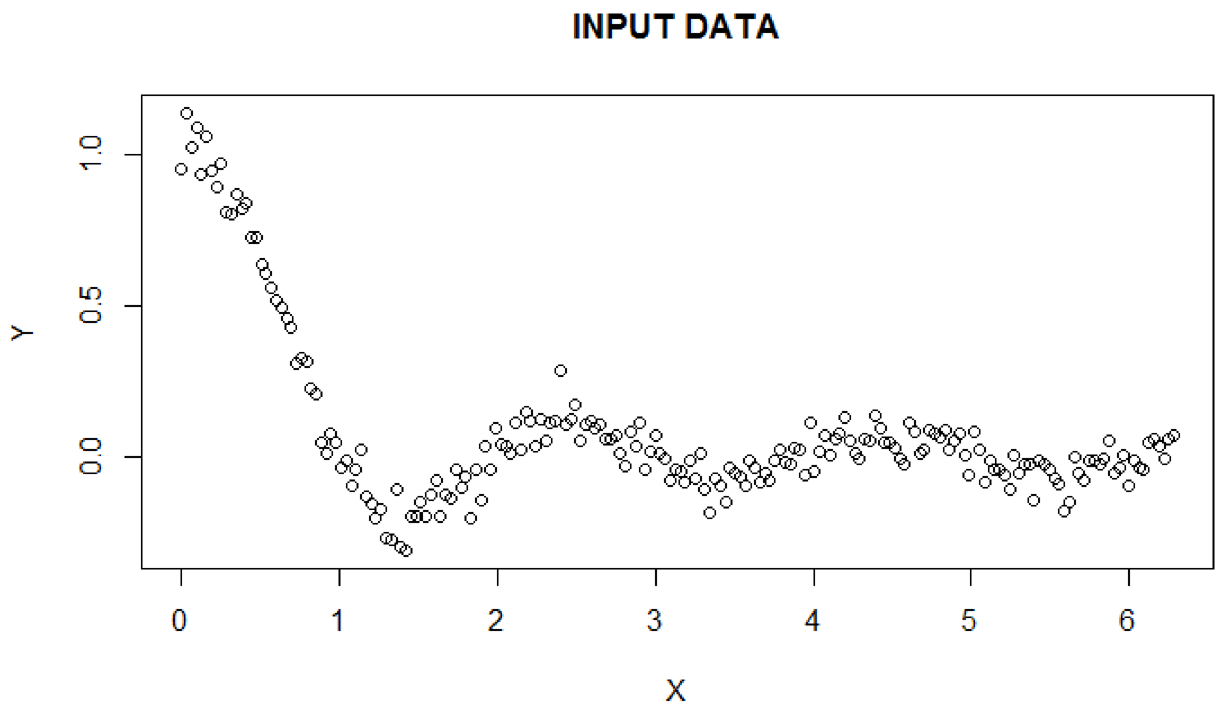
From the above table as we go for higher order polynomial, the error on train set decreases but error on test set seems to increase. Over fitting happens for higher order polynomial.

So **Linear model is good fit**

Data-Set Used

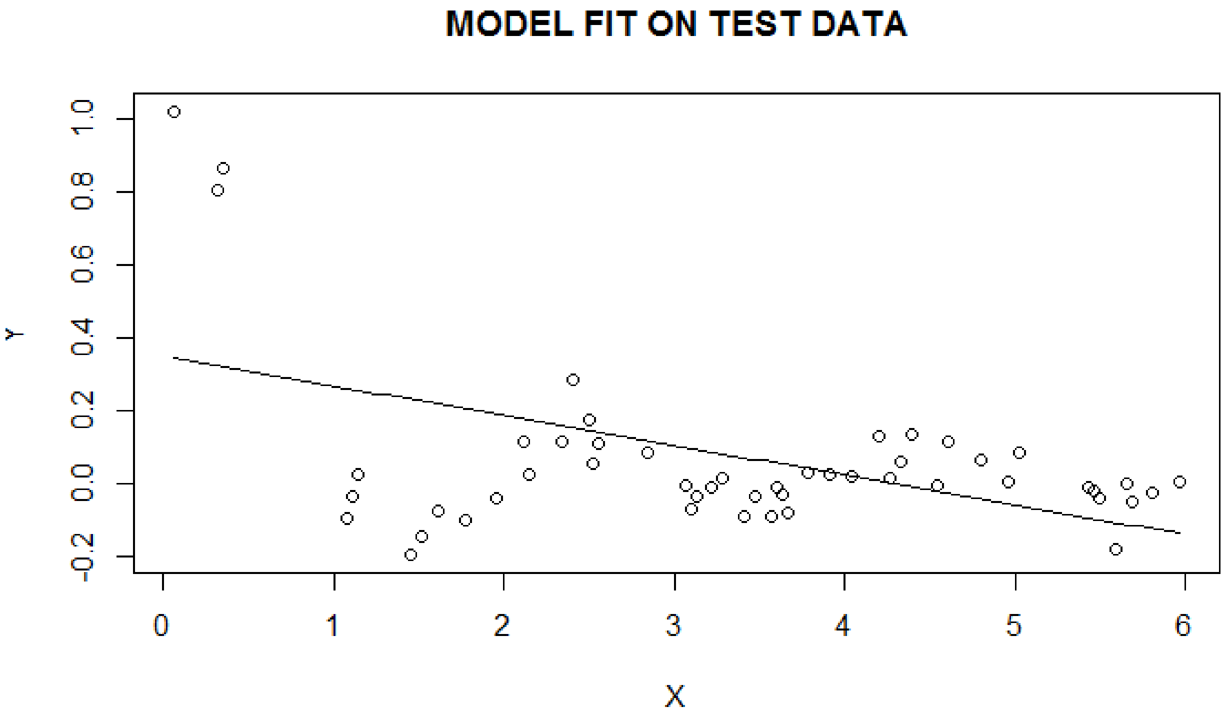
svar-set2.dat

Data Plot



From the above graph we can see that the data is non-linear. We would be requiring higher order polynomial to fit our data

Let us start with linear model



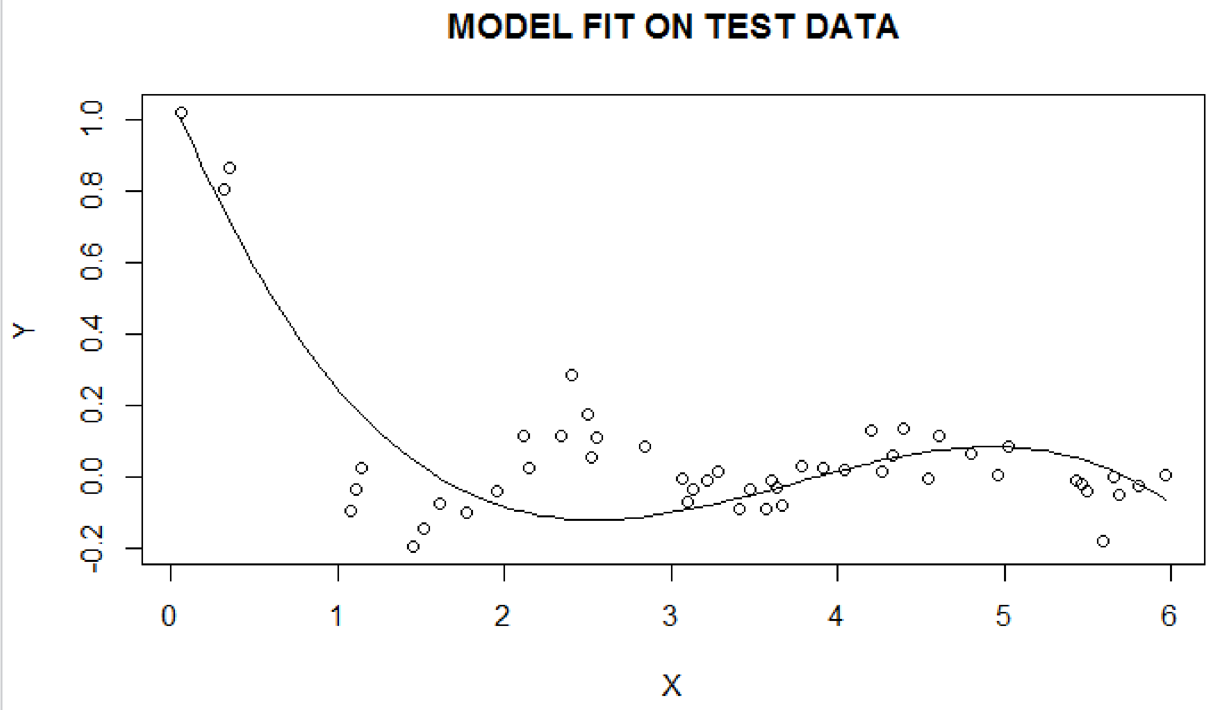
From the above graph we can see linear model does not fit our data well.

Let's try non-linear some non-linear models

Plot with polynomial two

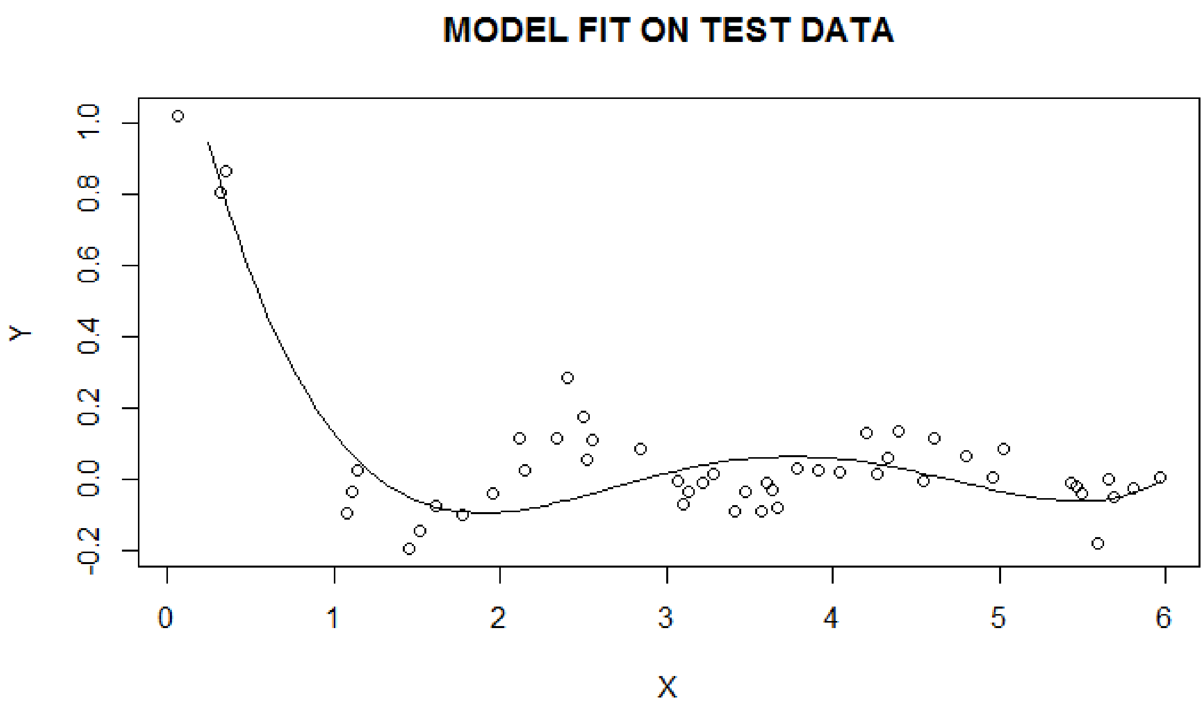
This seems okay. But let’s try with order 3

Plot with polynomial three

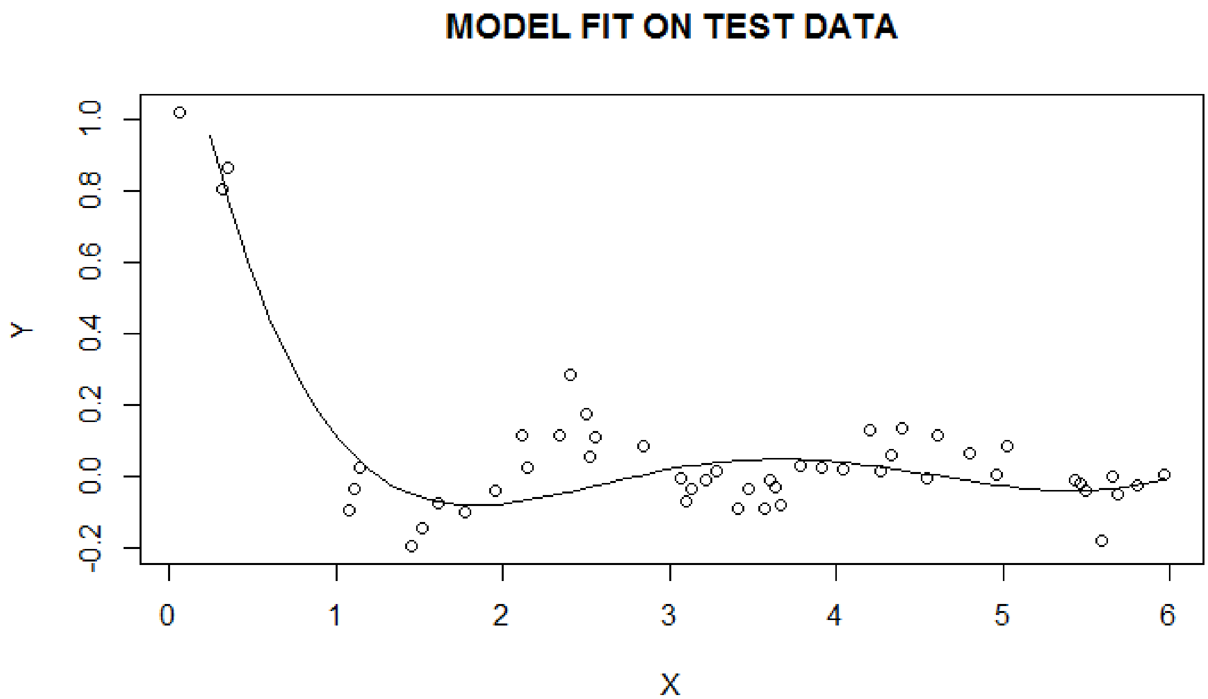


This seems to be the best fit. Let’s try with higher order four.

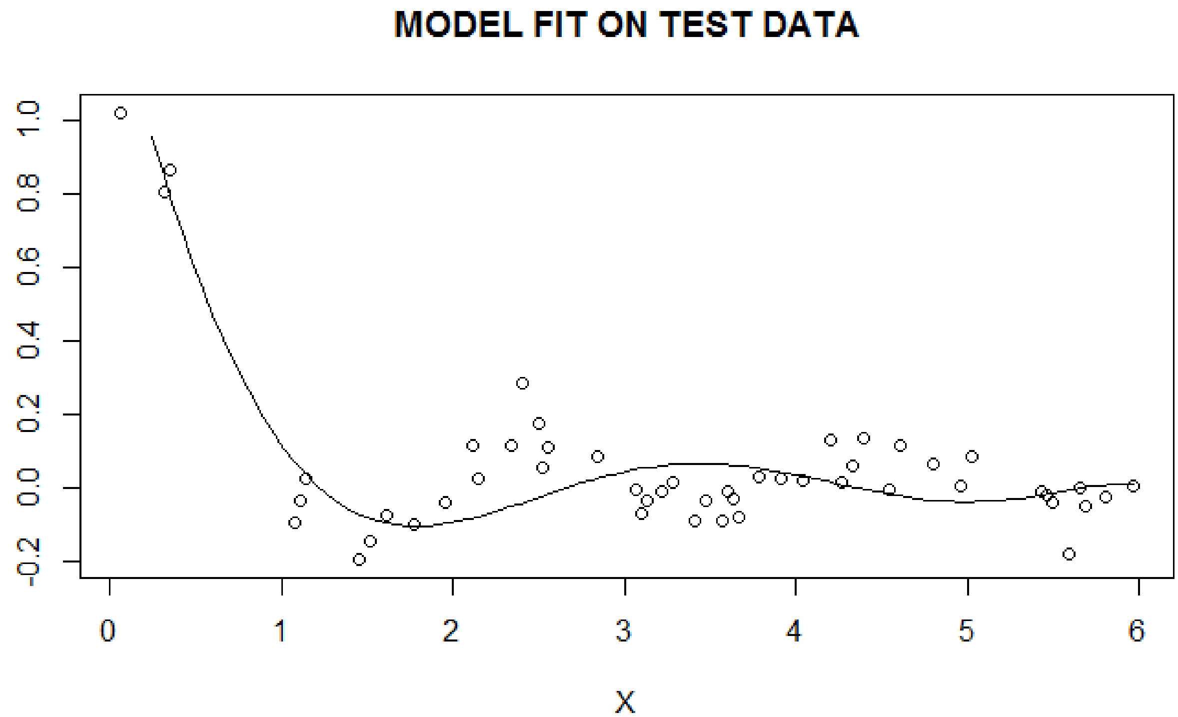
Plot with polynomial four


MODEL FIT ON TEST DATA
Y
-0.2
0.0
0.2
0.4
0.6
0.8
1.0
X
0
1
2
3
4
5
6

Plot with polynomial five



Plot with polynomial six



Definitely from the above graph it is clear that our model over fits the data.

Let's do **10 fold cross validation** to decide among polynomial 3, 4, 5

The mean squared error is low for **model with degree 4**. So we conclude this is the **best model**

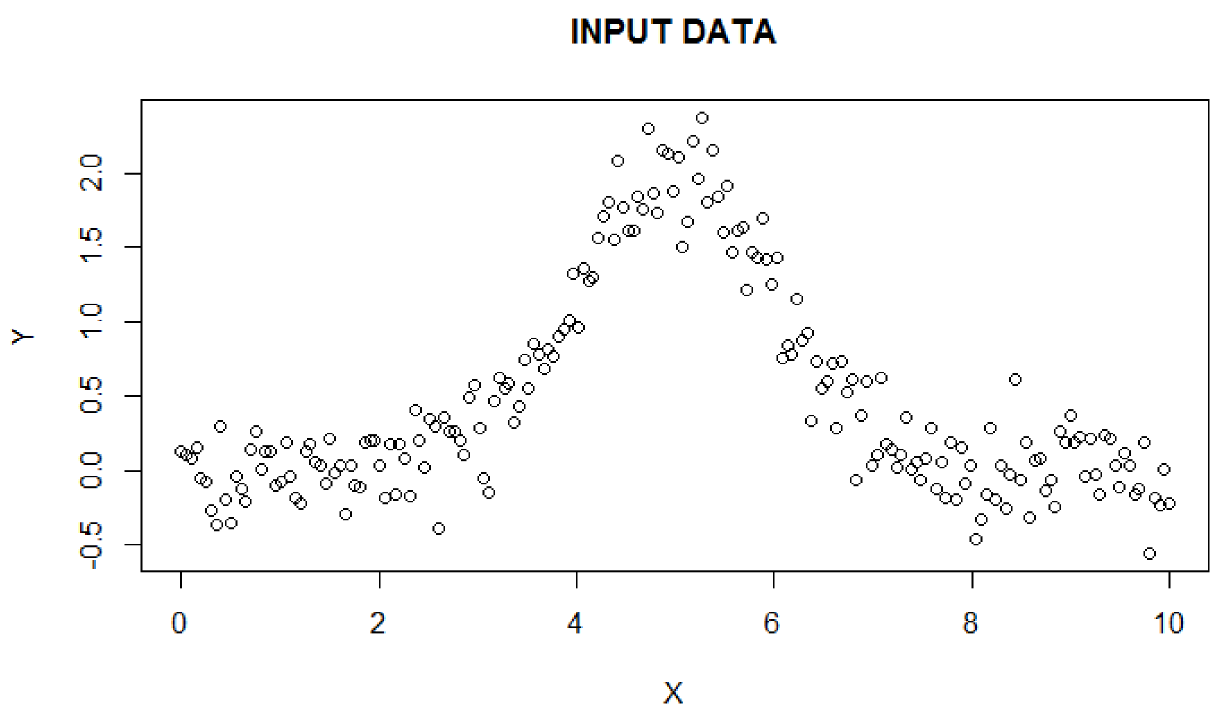
Polynomial	Error on test set	Error on Train Set
1	0.04382677	0.06488278
2	0.03304288	0.0407704
3	0.01845931	0.02117343
4	0.01167638	0.01147739
5	0.01075711	0.01127793
6	0.01150745	0.01084436

From the above table the error on test set starts increasing at polynomial six. To decide upon 3, 4, 5 we used 10-fold cross validation and **model with degree four seems to be the best**

Data-Set Used

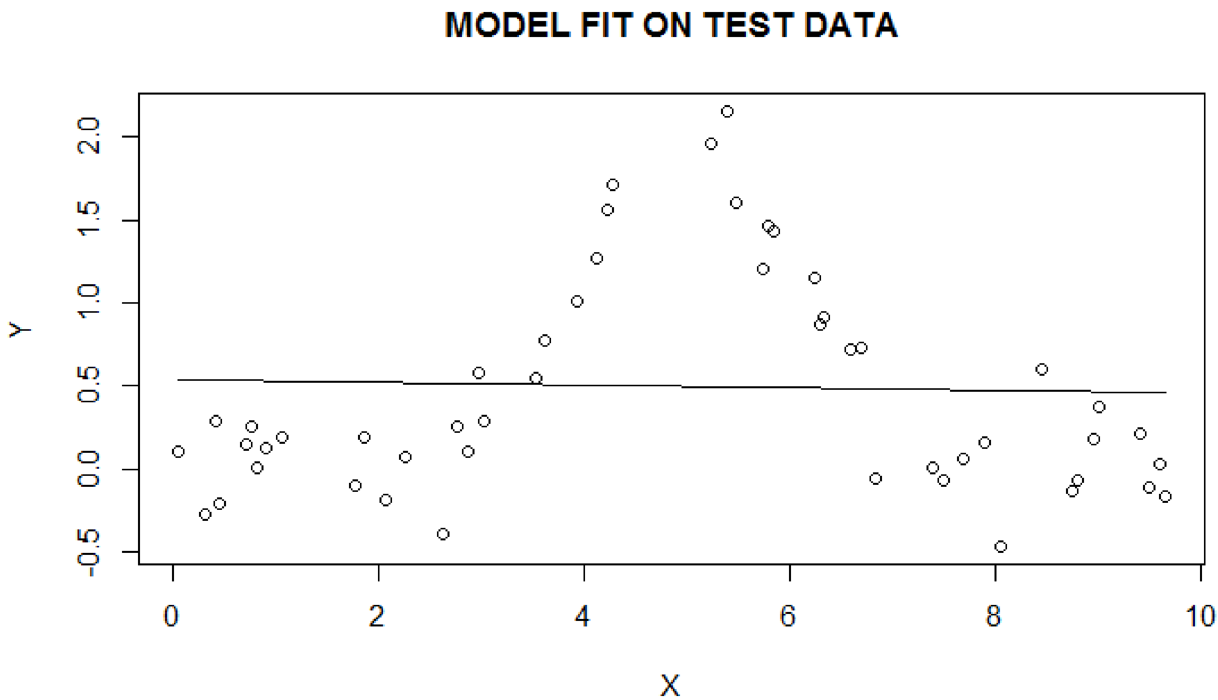
svar-set3.dat

Data Plot



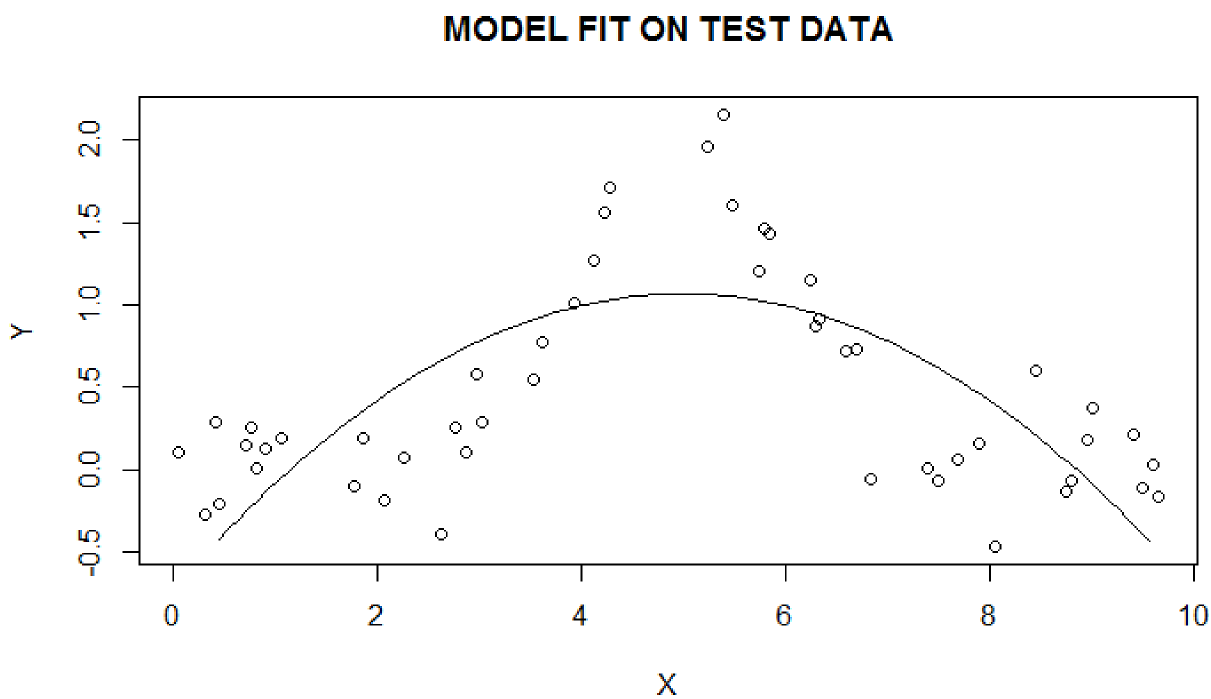
From the graph we can see there is a big bump in the middle. Polynomial with order two would be the good fit

Let’s try linear model first



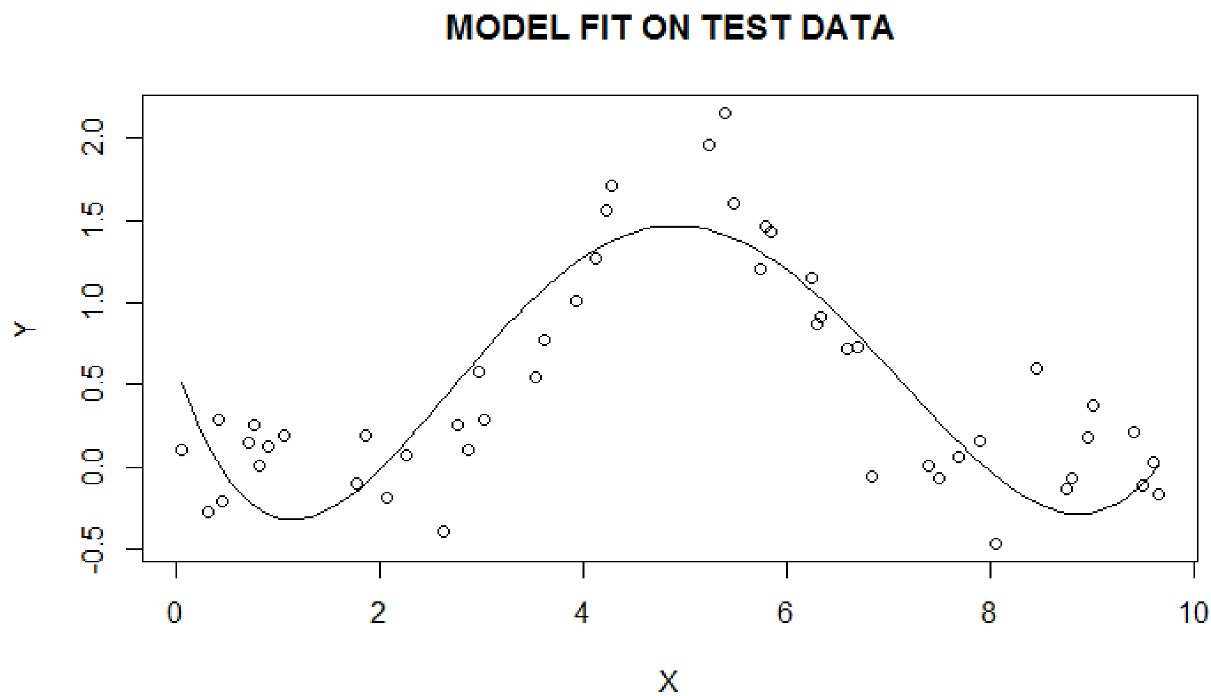
As expected the model does not fit the data well. As the dataset is non-linear

Let’s try with polynomial of order 2



This fits the graph fairly well.

Let us try with polynomial with order 5



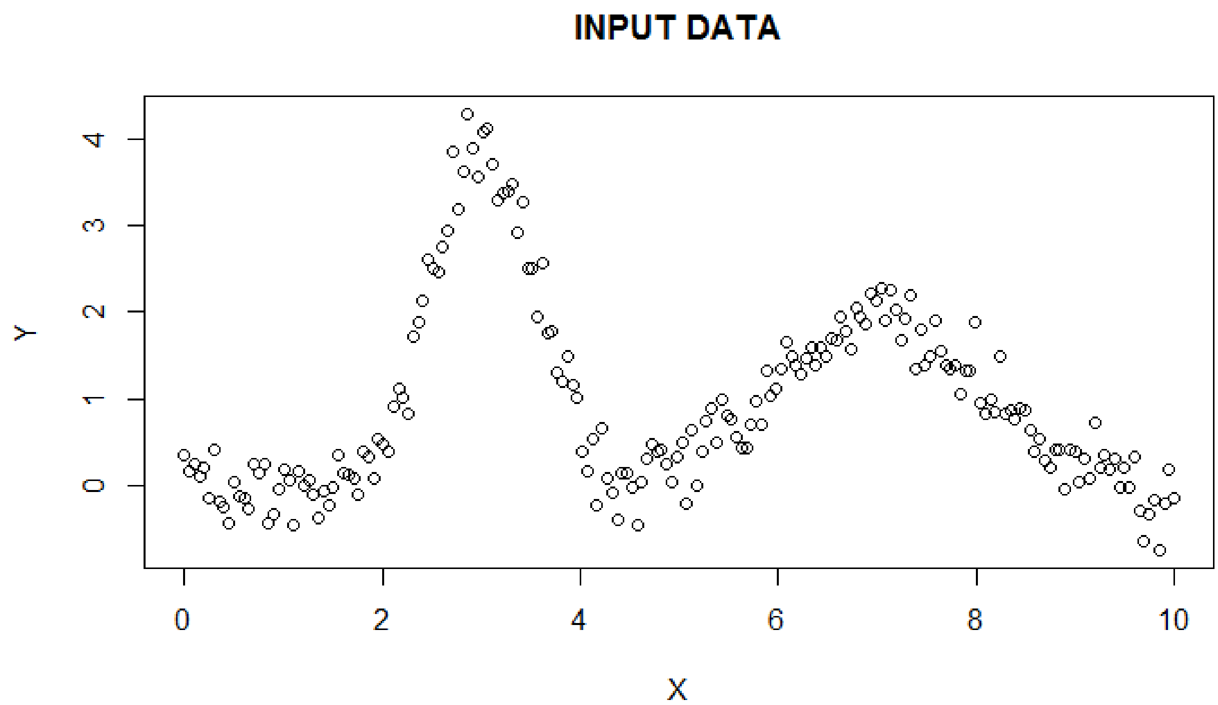
This is clearly **over-fitting the data**

So the best fit for **model for this data is model with polynomial 2 or 3**

Data-Set Used

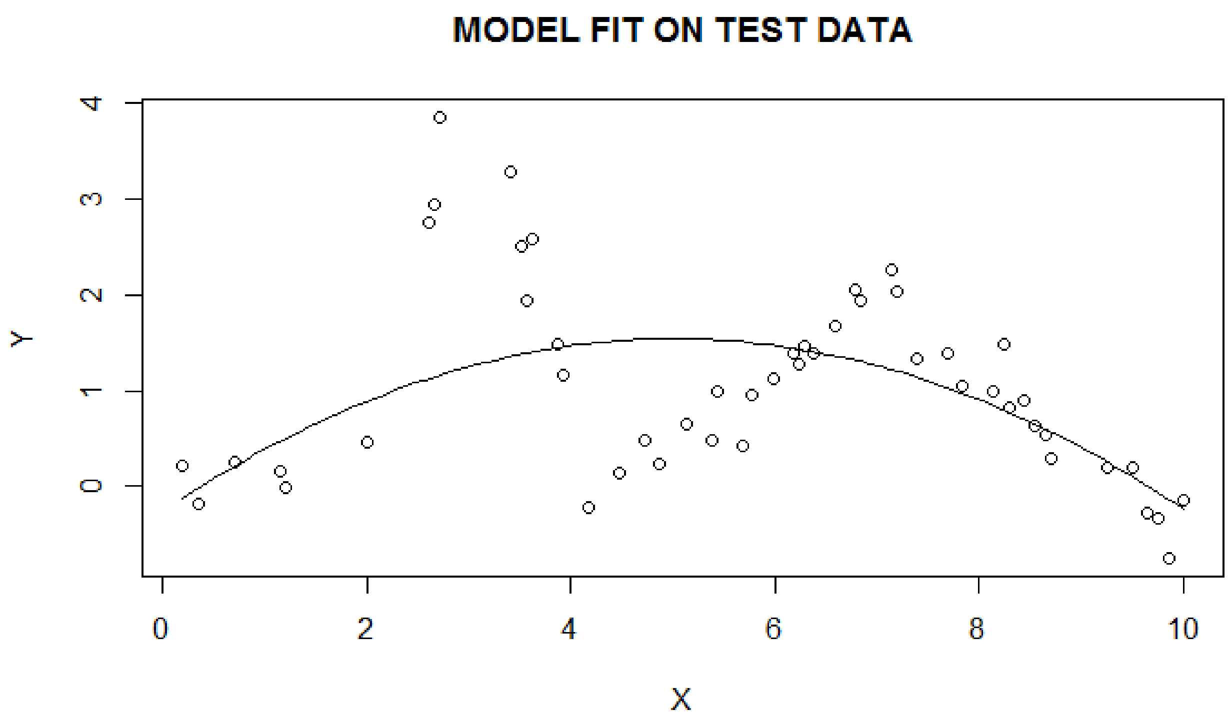
svar-set4.dat

Data Plot



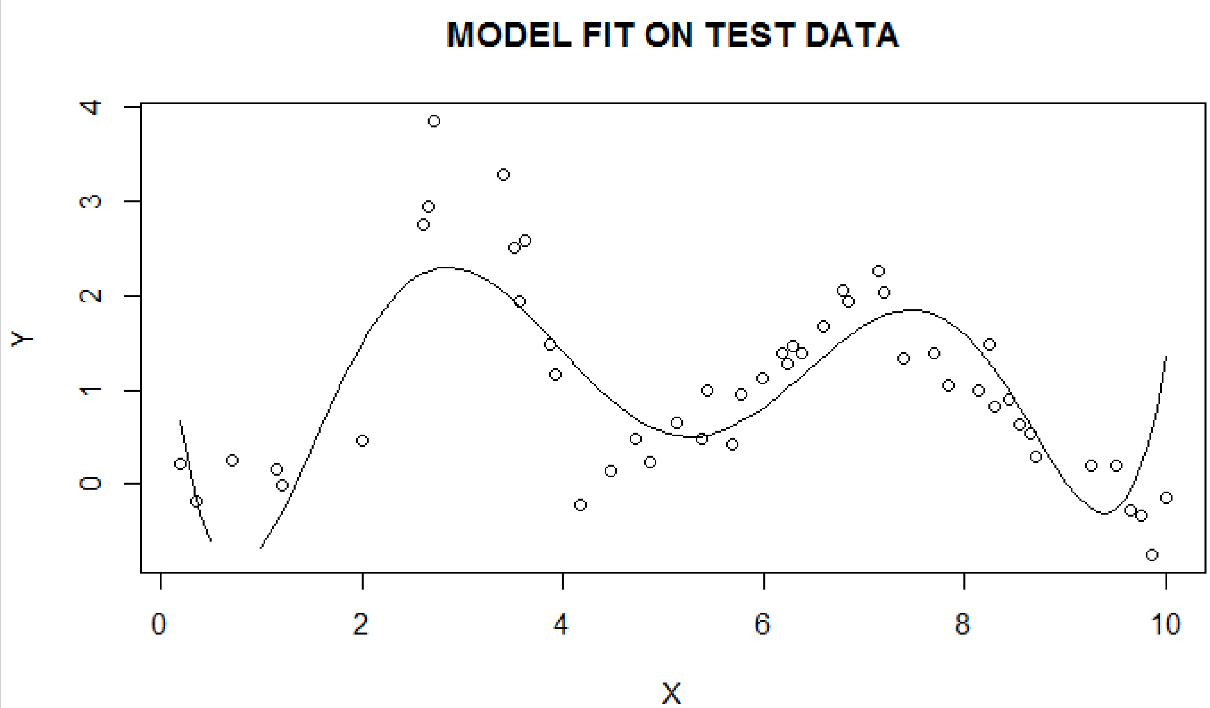
This data has got three bumps so polynomial with 3 or 4 would be the good fit

As we clearly know the **data is non-linear let's start from polynomial of order 2**



As we can see from the above plot the model does not fit the data well and the **MSE is also high**

Let's try with **polynomial of order 6.** This should reduce the MSE on training set and increase the error on test set

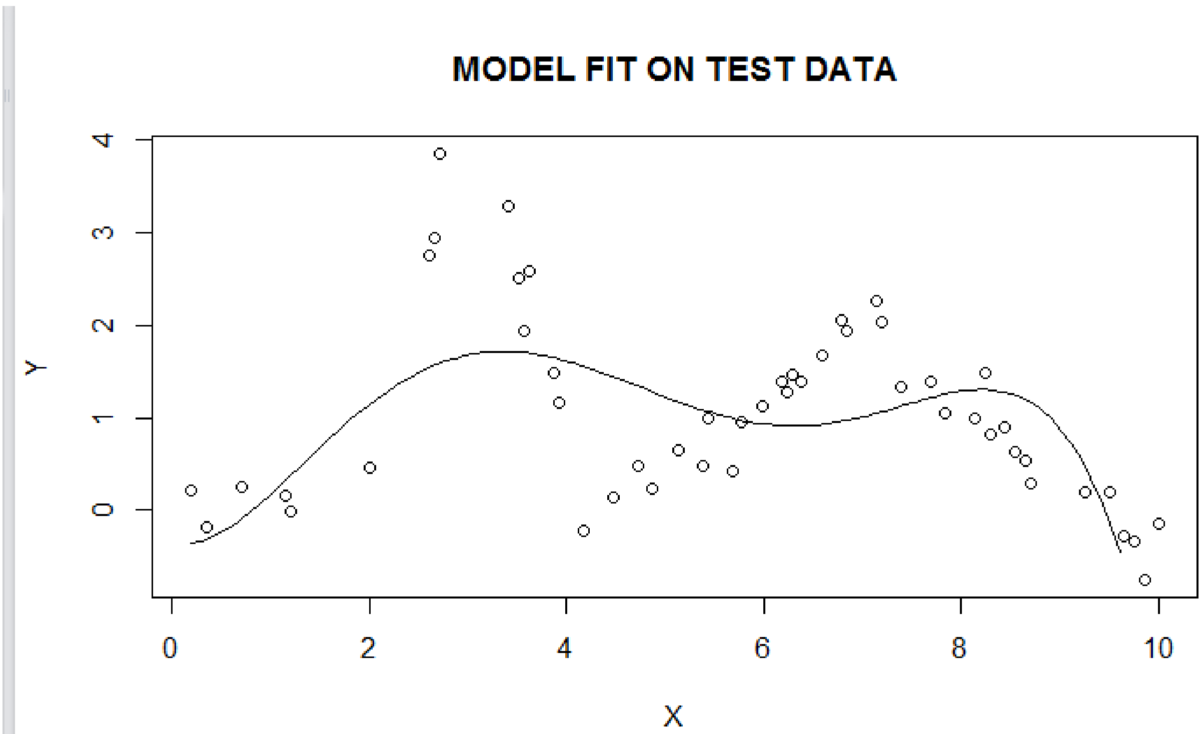


As expected it **over-fits** the data. So our answer is somewhere between 3 to 5

Let's try **10 fold cross validation to choose the best model.**

After 10-fold cross validation we found **that polynomial with order 5 is the best fit for this data**

Model fit with polynomial 5



Let's take a linear model and polynomial model **reduce the amount of training data to see what happens**

Linear model dataset	svar-set1.dat
Polynomial model dataset	svar-set2.dat

- In both data set as we reduce the amount of training data, the MSE **starts increasing**.
- It **increases in higher rate** in polynomial model than in linear model