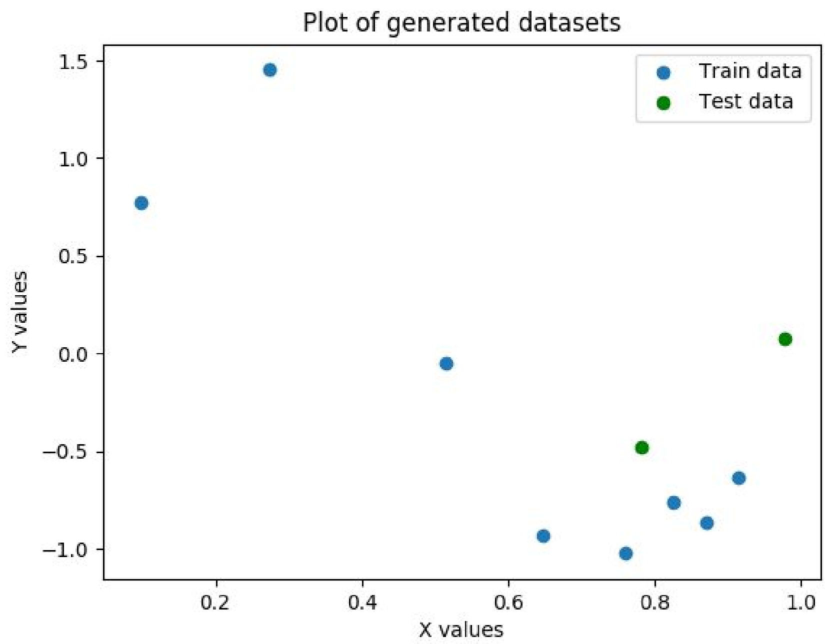
Assignment 1 Report

Name: Sai Saketh Aluru
Roll No. 16CS30030

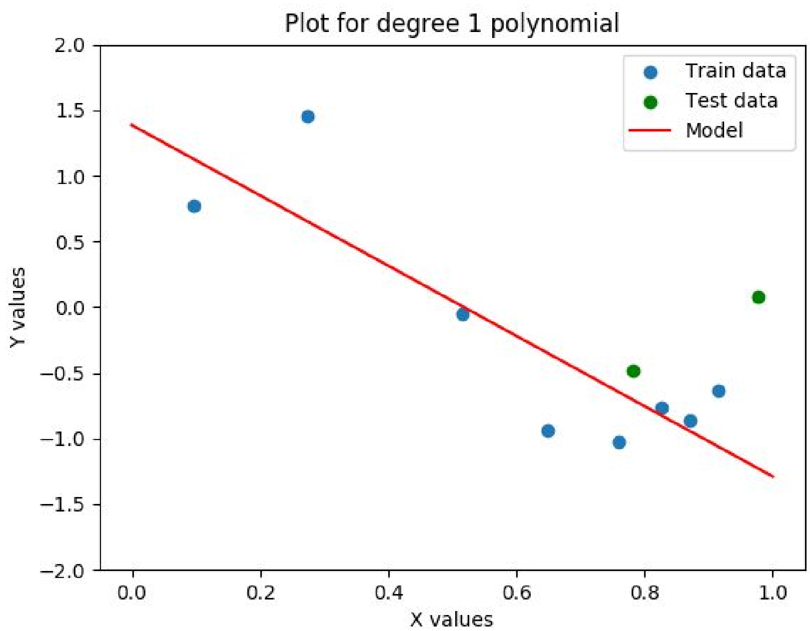
Part - 1 and 2:

Training 10 data points with polynomials of degrees 1 to 9 and plotting the generated dataset and the learnt models.



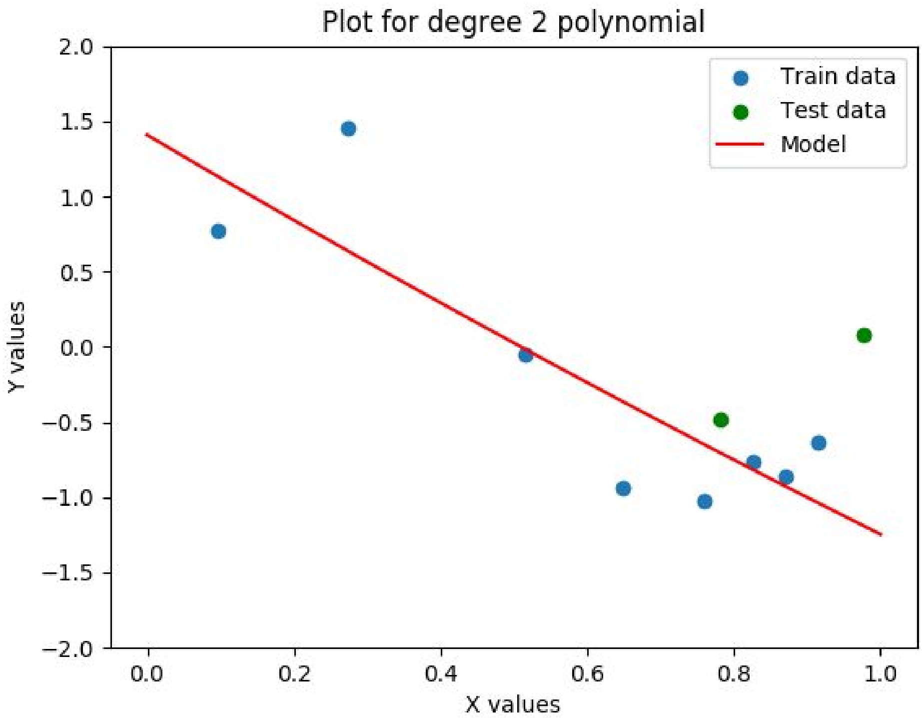
Degree - 1:

Polynomial Term	Coefficient Leant
1	1.38509479
x	-2.67420267



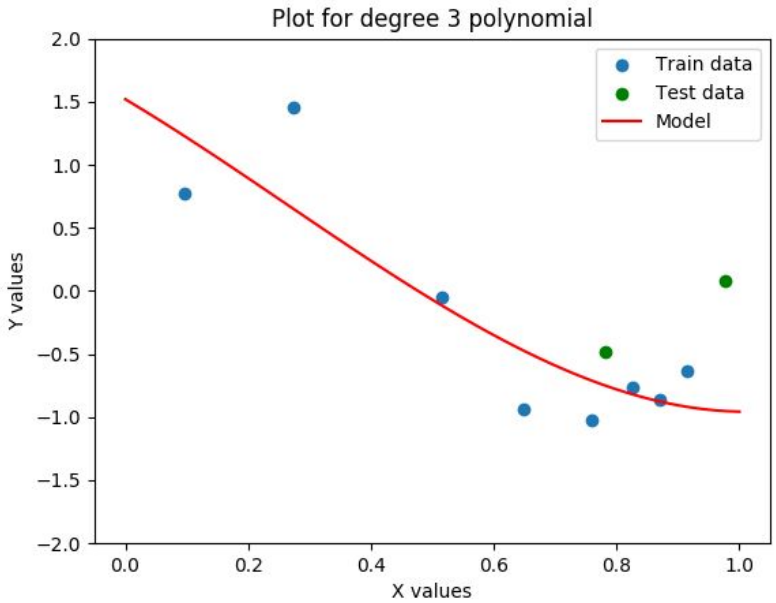
Degree - 2:

Polynomial Term	Coefficient Learnt
1	1.40793433
x	-2.87543209
x^2	0.22126281



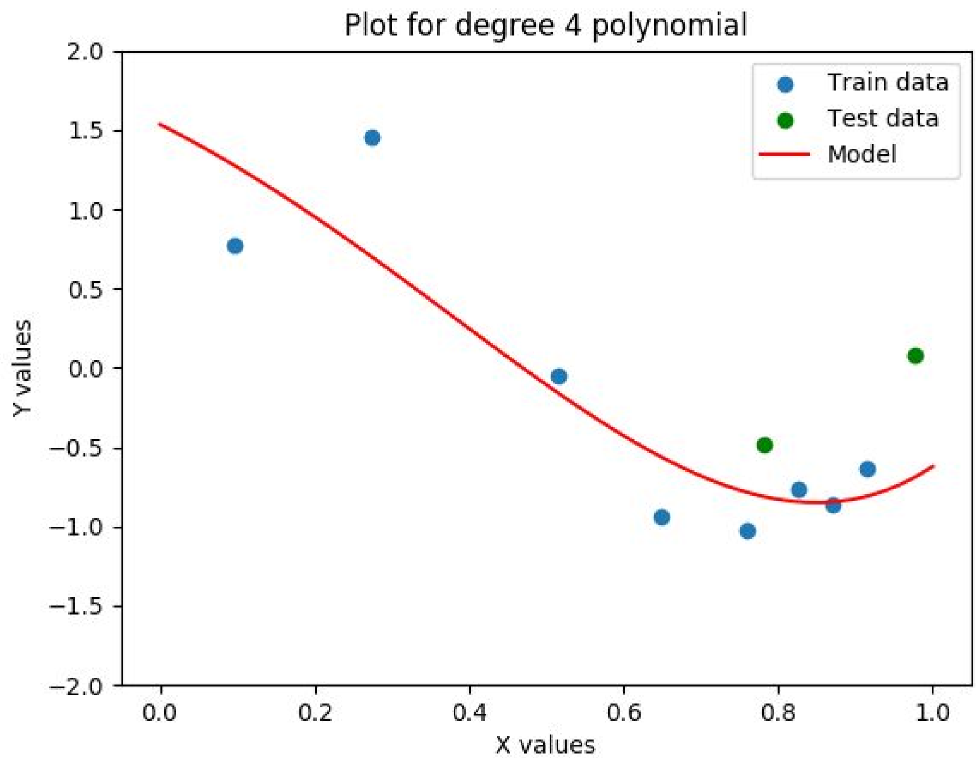
Degree - 3 :

Polynomial Term	Coefficients Learnt
1	1.51901107
x	-2.89697236
x^2	-1.54400282
x^3	1.96540465



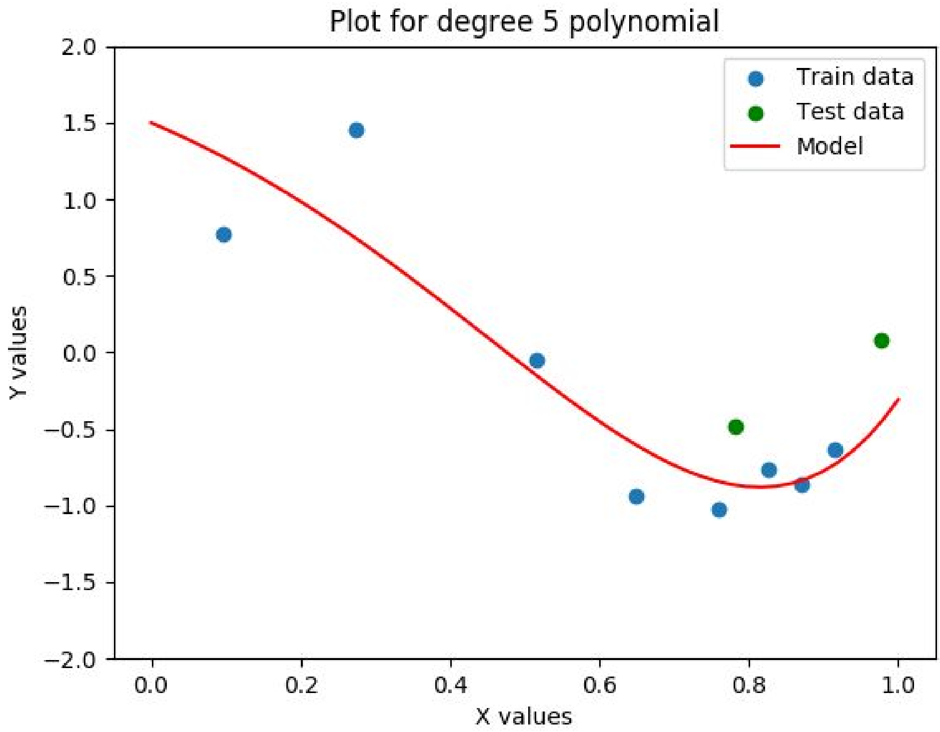
Degree 4:

Polynomial Term	Coefficients Learnt
1	1.53447152
x	-2.45946655
x^2	-2.40979411
x^3	0.34058680
x^4	2.37180618



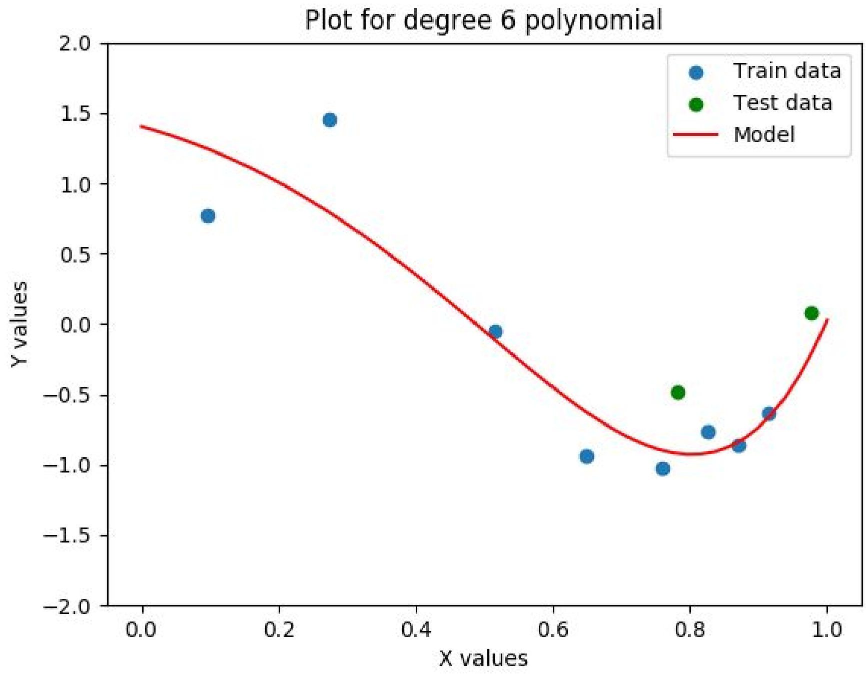
<u>Degree-5</u>

Polynomial Term	Coefficients Learnt
1	1.4977282
x	-2.06511492
x^2	-2.46097179
x^3	-0.49926925
x^4	0.63982242
x^5	2.57832187



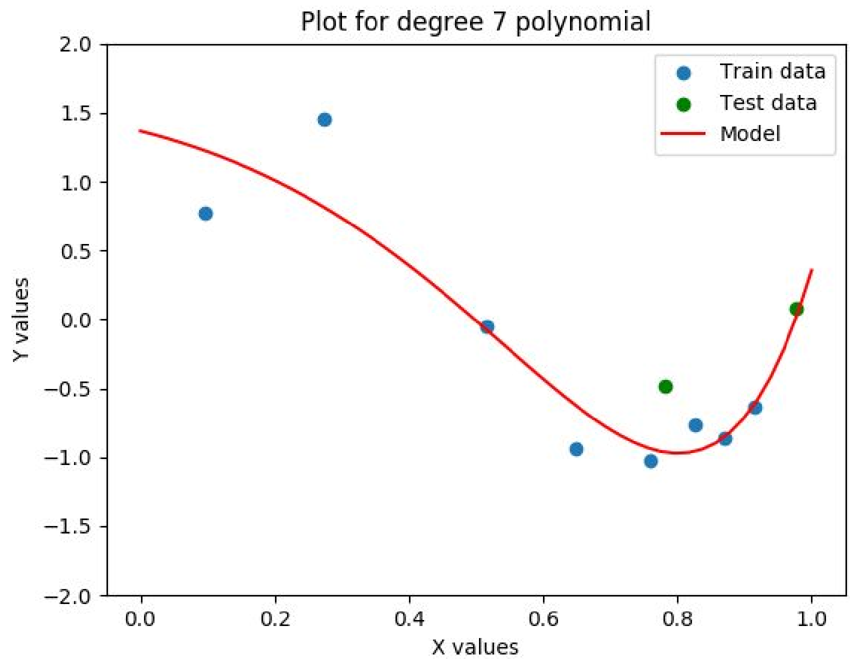
Degree - 6:

Polynomial Term	Coefficients Learnt
1	1.40411784
x	-1.36420913
x^2	-2.88179639
x^3	-1.35003796
x^4	0.57855532
x^5	1.56323346
x^6	2.07571305



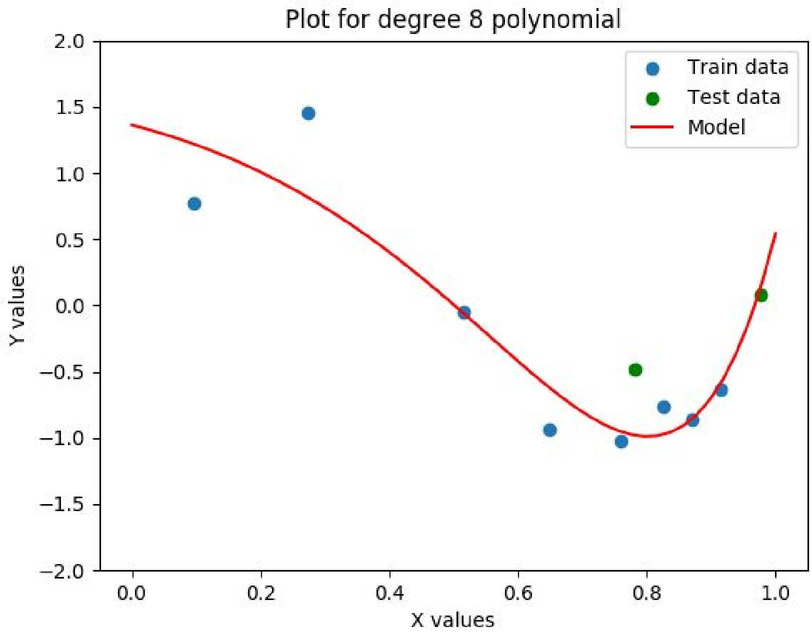
<u>Degree 7:</u>

Polynomial Term	Coefficients Learnt
1	1.36452962
x	-1.21467365
x^2	-2.53843160
x^3	-1.37228772
x^4	-0.53898502
x^5	1.08732465
x^6	1.50208276
x^7	2.06623165



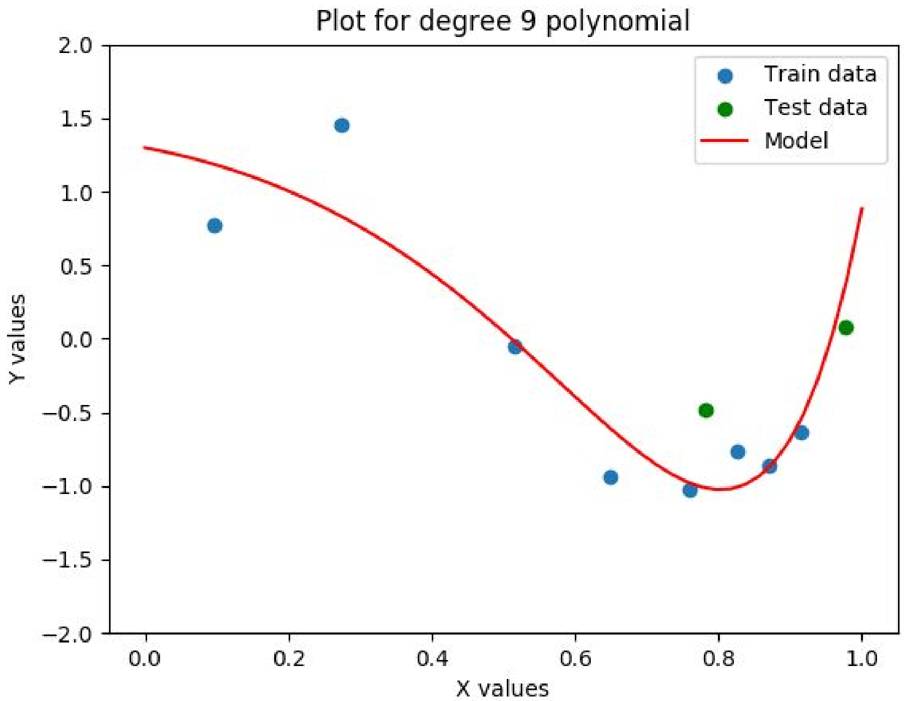
Degree 8:

Polynomial Term	Coefficients Learnt
1	1.36119954
x	-1.24677276
x^2	-2.34976219
x^3	-1.28018654
x^4	-0.76901922
x^5	1.02534432
x^6	1.01655481
x^7	1.20570275
x^8	1.57922089



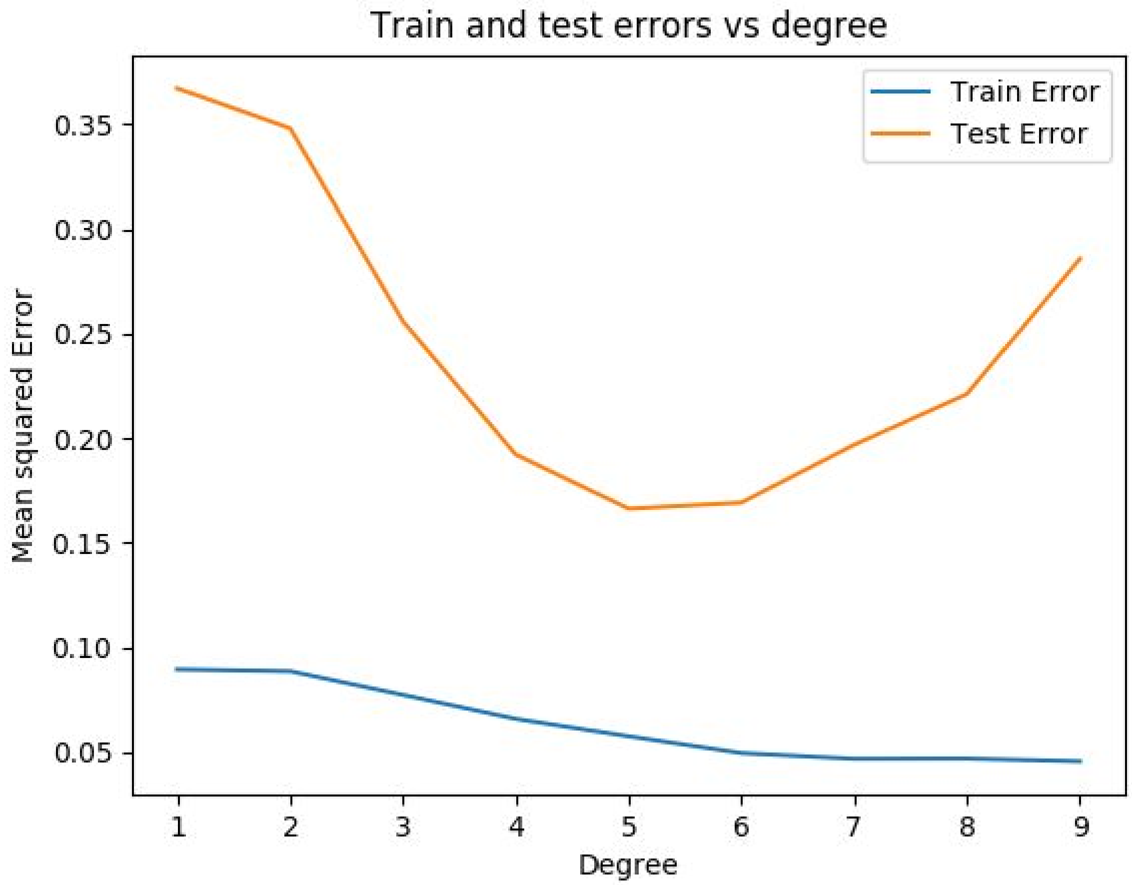
Degree 9:

Polynomial Term	Coefficients Learnt
1	1.29796089
x	-0.91918481
x^2	-2.38390929
x^3	-1.70853012
x^4	-0.31776497
x^5	0.46634811
x^6	0.41447187
x^7	1.29300087
x^8	1.25426725
x^9	1.48683865



Part - 2(b)

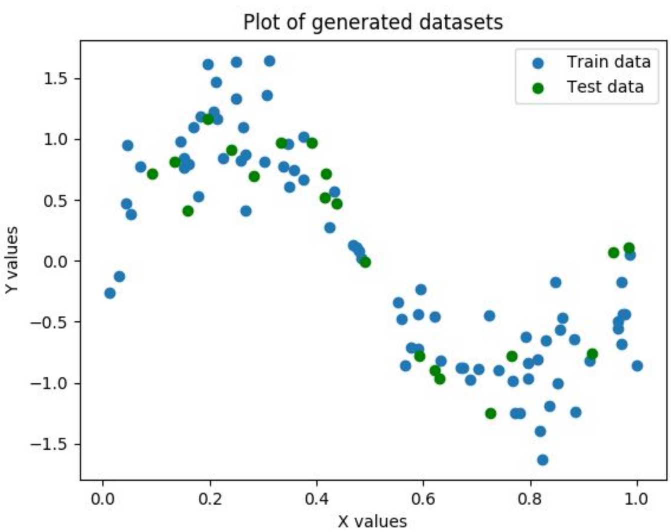
Degree	Train Error	Test Error
1	0.08970968	0.36723795
2	0.08882494	0.34809241
3	0.07750535	0.25604567
4	0.06604118	0.19235145
5	0.05767775	0.16635842
6	0.04966514	0.16928417
7	0.04691591	0.19690255
8	0.0470218	0.22115767
9	0.04573545	0.28591107



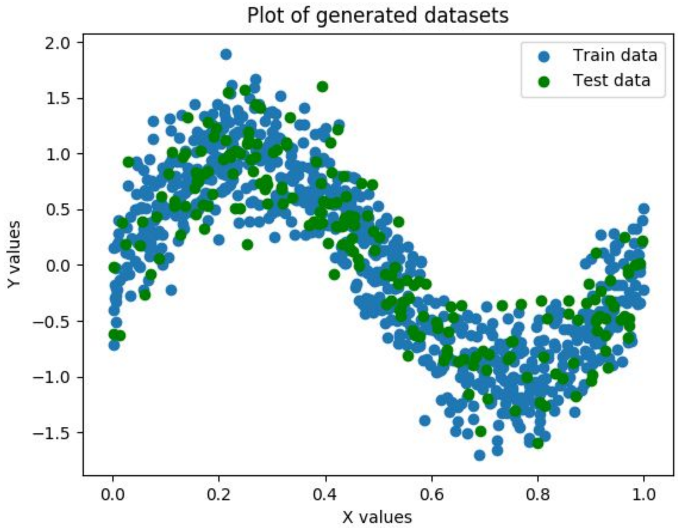
We observe that the training error is decreasing with increase in degree of the polynomial, whereas the test error is decreasing at first but increasing after degree = 5. This is expected, as with increase in degree of the polynomial, the model is starting to overfit the data. The optimal fit for this set of data points is found to be a 5th degree equation. This value is chosen based on the test errors obtained and the plot shown above

Part - 3

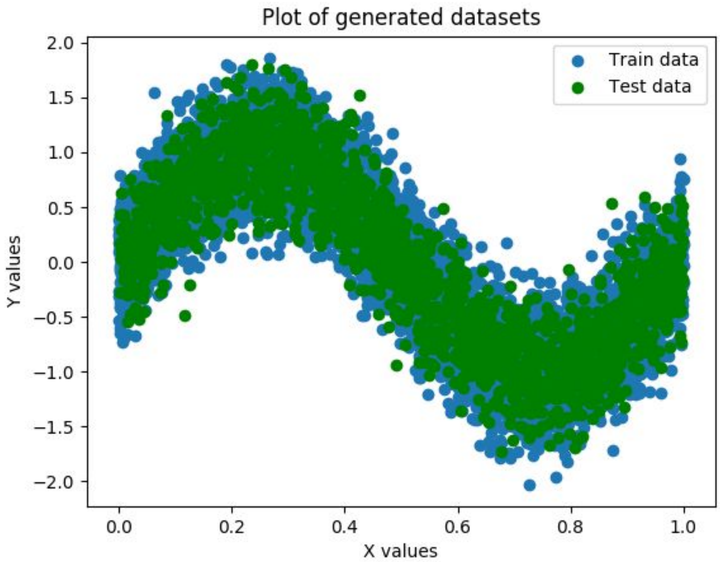
100 Points:



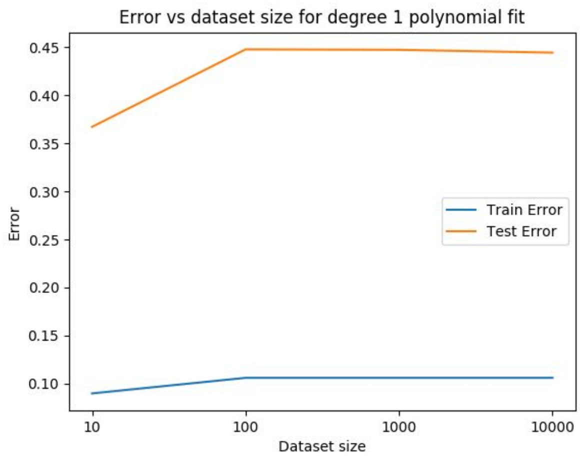
1000 points:

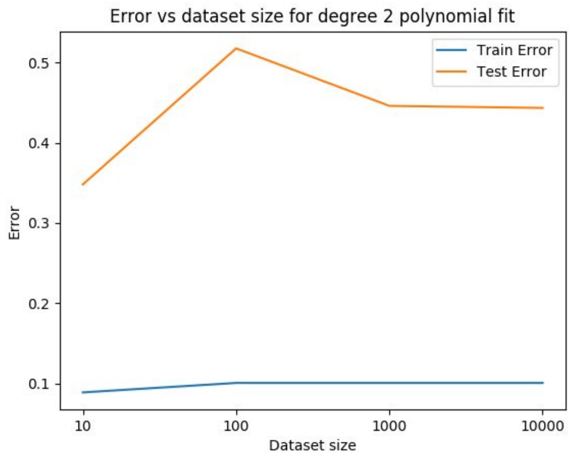


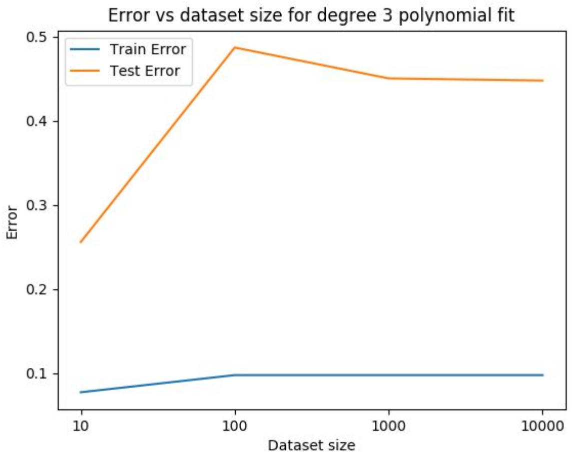
10000 points:

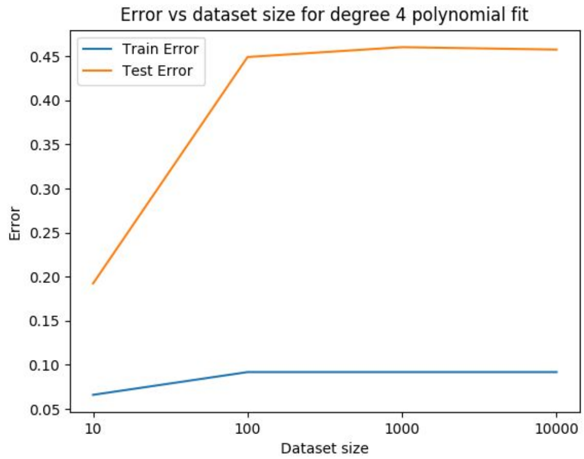


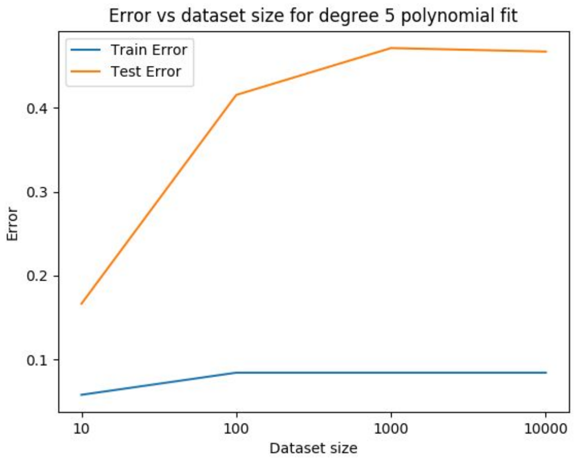
Plots for error vs dataset size for each degree polynomial

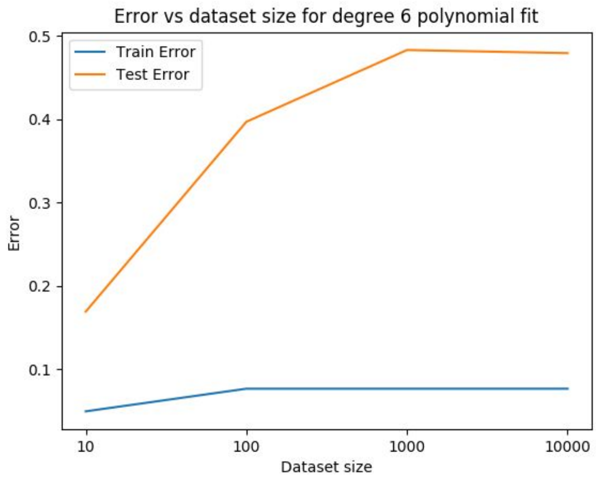


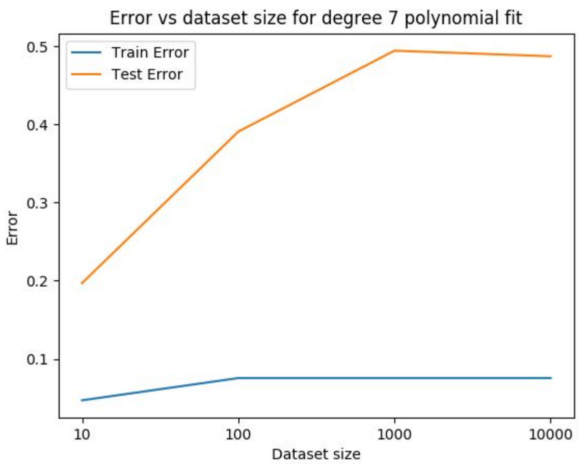


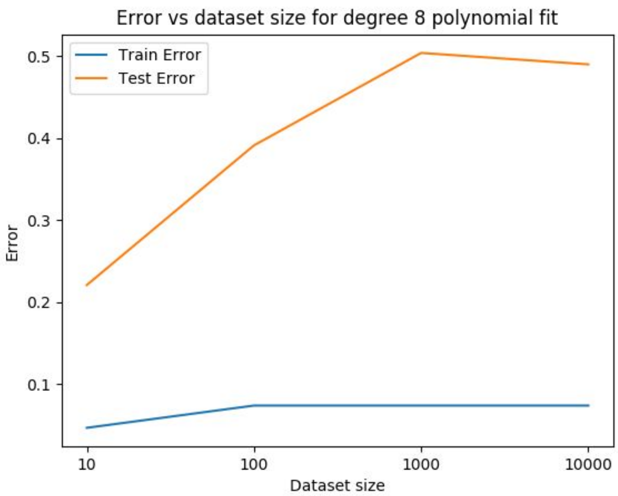

Error vs dataset size for degree 3 polynomial fit
Train Error
Test Error
Error
Dataset size
10
100
1000
10000
0.1
0.2
0.3
0.4
0.5

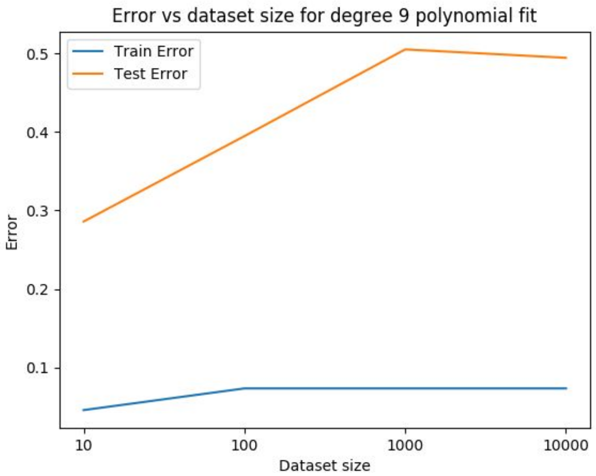

Error vs dataset size for degree 4 polynomial fit
Train Error
Test Error
Error
Dataset size
10
100
1000
10000
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
0.45


Error vs dataset size for degree 5 polynomial fit
Train Error
Test Error
Error
Dataset size
10
100
1000
10000
0.1
0.2
0.3
0.4


Error vs dataset size for degree 6 polynomial fit
Train Error
Test Error
Error
Dataset size
10
100
1000
10000
0.1
0.2
0.3
0.4
0.5


Error vs dataset size for degree 7 polynomial fit
Train Error
Test Error
Error
Dataset size
10
100
1000
10000
0.1
0.2
0.3
0.4
0.5


Error vs dataset size for degree 8 polynomial fit
Train Error
Test Error
Error
Dataset size
10
100
1000
10000
0.1
0.2
0.3
0.4
0.5

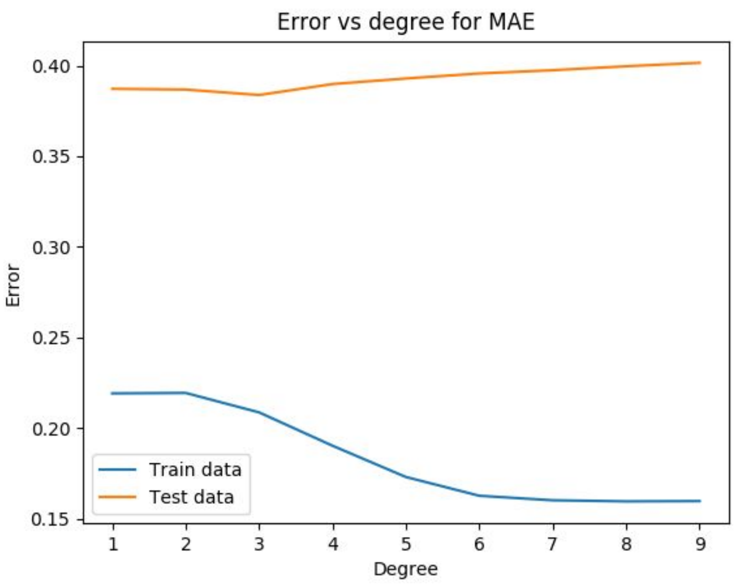


<u>Observation:</u>
For all the degree polynomials, for 10 point dataset, due to the very low number of training and test examples, the error trends seem a bit deviated from the expected trends. But in case of the other three instances (100,1000,10000), initially, for smaller degree polynomials, we observe the test error to decrease (up to degree 4) and beyond that, the test error to increase. This might be because of overfitting, as with more amount of data the amount of noise present also might increase and higher degree polynomials are trying to fit in the noise into the model and are failing to generalise more to the test data.

Part - 4:

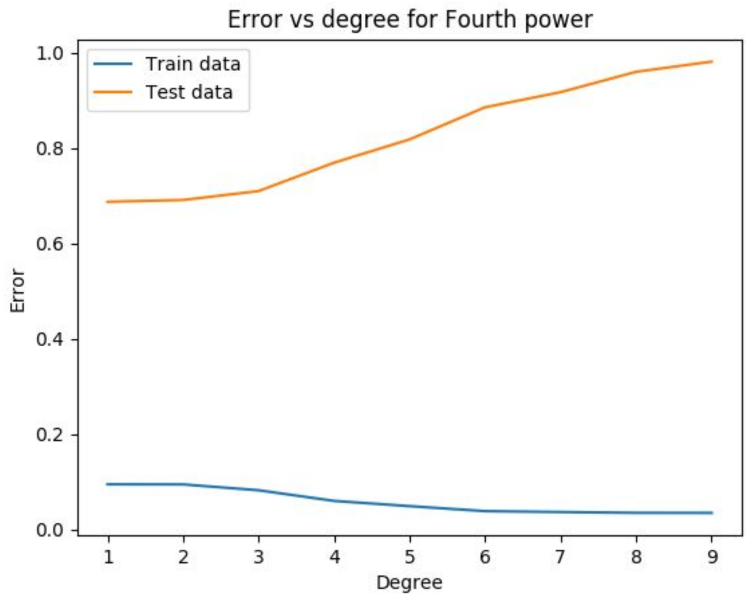
Here, I am making use of the 1000 point dataset created in part 3 for training on MAE and Fourth Power cost functions, since the dataset is sufficiently large to give a better understanding of the model and is also faster than the 10,000 point dataset.

<u>Mean Absolute Error:</u>
Here, we observe the following trend for MAE cost function.



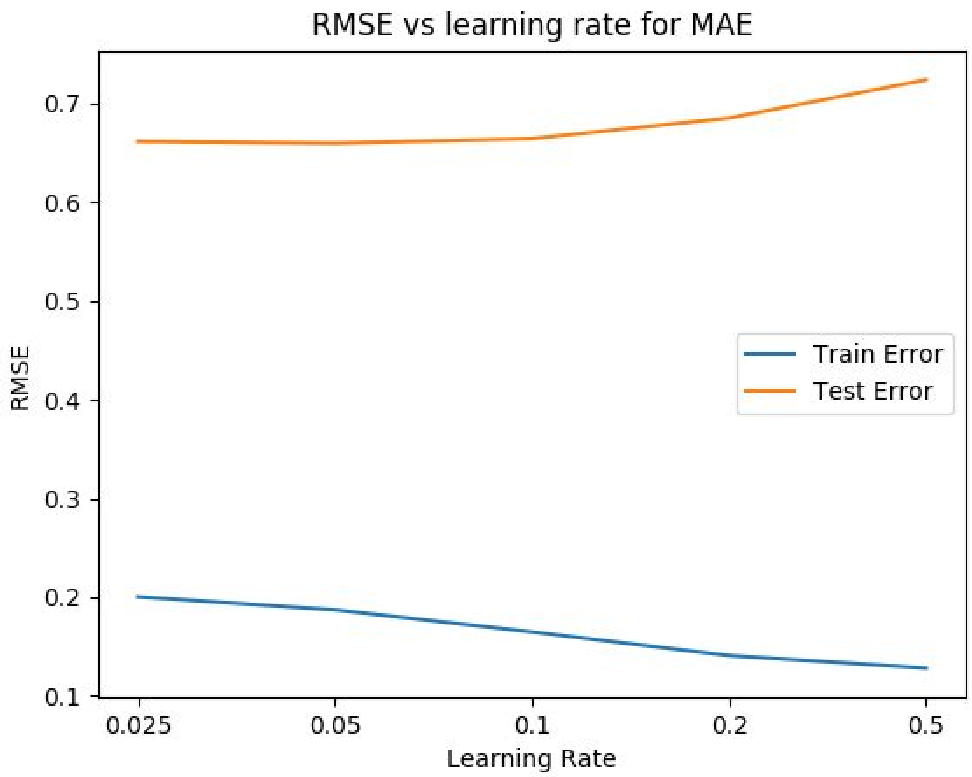
As we can see, the model start to overfit after degree 3 polynomial. Hence degree 3 is a better fit for this.

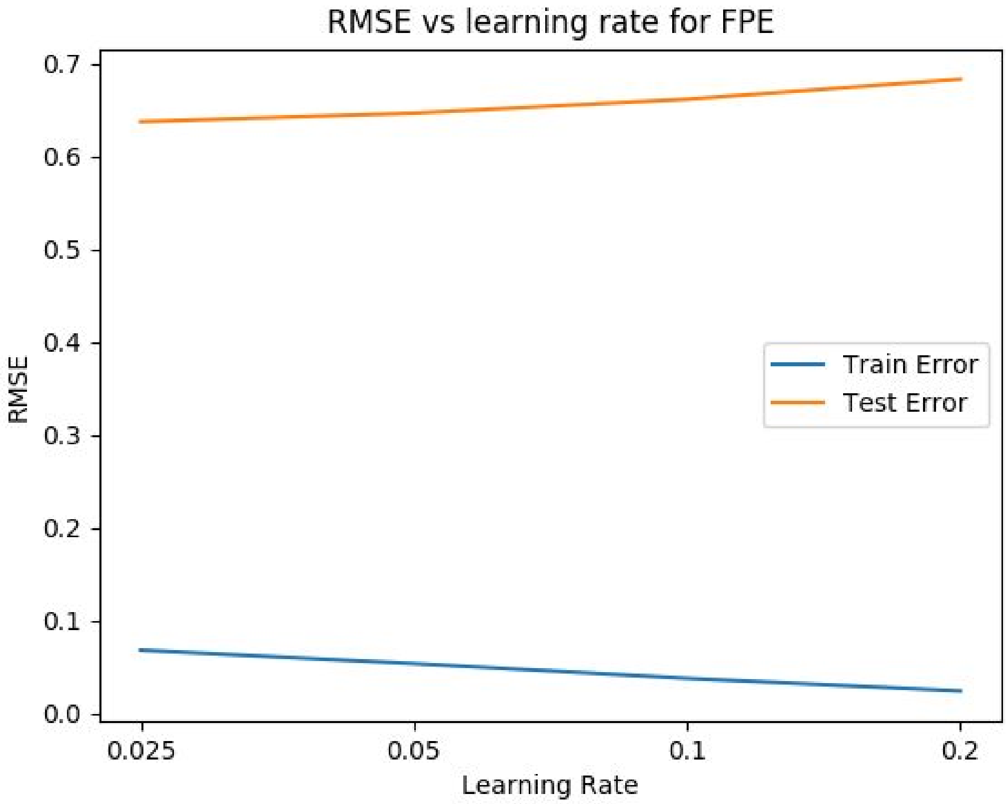
Fourth Power Error:



Here also, we see the error increasing from degree 4, which indicates overfitting of higher degree polynomials. Therefore, we choose degree 3 as the optimal value here as well.

Choosing this degree=3 as the optimal one for both MAE and Fourth power, we train the model varying the learning rate. The observed trends are





Observation:

For the mean absolute error, we observe that the minimum test error (RMSE) occurs at a learning rate of 0.05. Although the training error decreases with further increase in alpha, we observe that the test error is increasing, suggesting that the model is not able to generalise well. So for this cost function, 0.05 is optimal

For the Fourth power error, we observe that the test error is continuously increasing with an increase in alpha and the minimum occurs at alpha = 0.025. Hence here, the optimal value is 0.025

Note: We observer that there was an overflow in variable size in the last case, i.e. alpha = 0.5 This is because of the high powers (3rd and 4th-degree powers) being computed during derivative calculation and computing cost function.