Assignment 1 codes Documentation

The images in this document are plotted without adding noise to input variables
In the code noise was added later and corrected

For question 1

- Assuming linear input data from 1 - 2*pi, trying two cases for plotting the output values for the testing data. One, considering Bias term, two, not considering the Bias term. Assuming the output to be of the form 1 + sin(x)

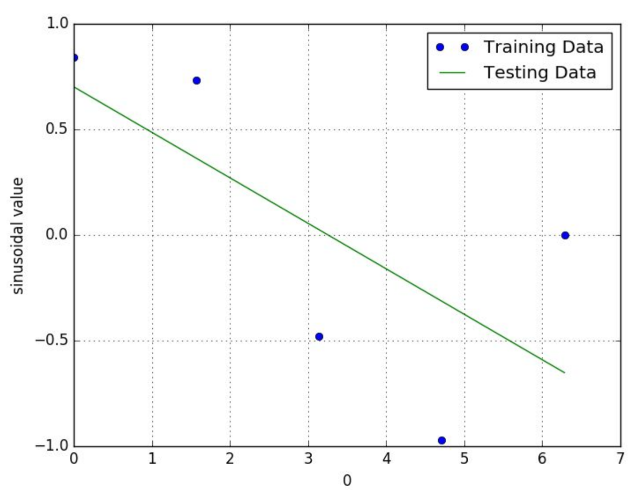
- The input matrix dimension : - (N * 2) , N- no. of Test cases
  2 - no. of attributes (first being Bias term)

  The output Vector Y dimension : - N*1

  Parameter matrix W = ((X'X)^-1)X'Y

- Since the function is assumed to be linear, what we get is a straight line

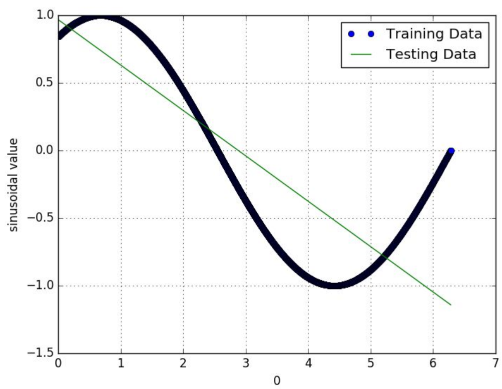
  Training with samples < 10 (Very less)



After testing with any number of examples the slope of the line was initially low.
The estimation of sinusoidal function at t = 0 is not exactly 1

But when training the data set with a large number of examples

The estimation was almost accurately 1 at t = 0 and the line saturates at that position even after increasing the training set to 5000



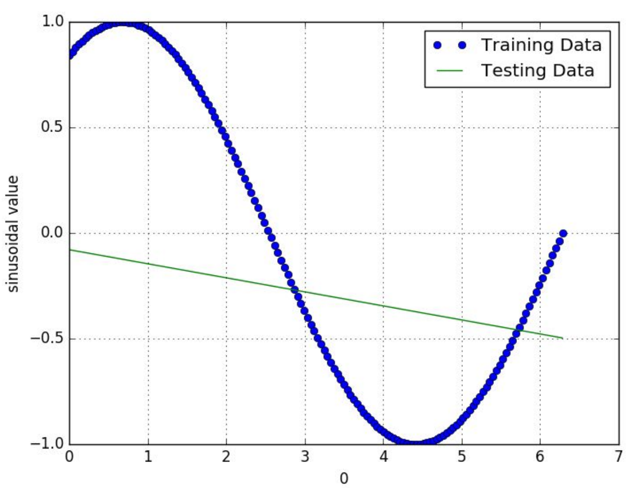
The line seems to be trying to minimize the error at all the points.

The error being the square of the difference of the estimated value and actual value

when the sum of the errors at each point came to a minimum, The parameters got fixed and the line saturated at that position

Case 2 : Not considering the bias term

The slope of the line changes same as the previous case but the line is below X-axis



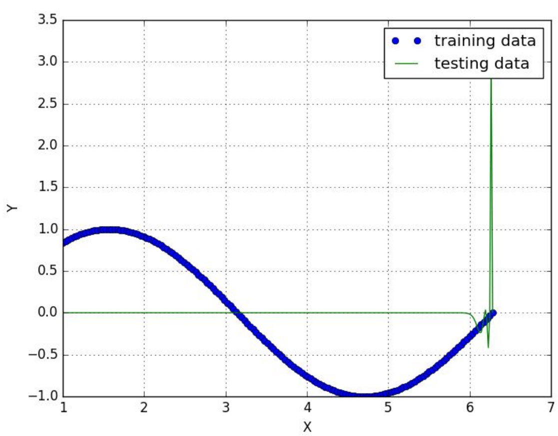
For question 2

Using Polynomial Basis Function

a) For case that the the degree of basis function > number of training samples

The parameters are so adjusted that the function seems to overshoot for the testing data depending on how many parameters more there are than the training samples

#training samples = 150
Degree of basis function = 300
#testing samples = 300 For the below image

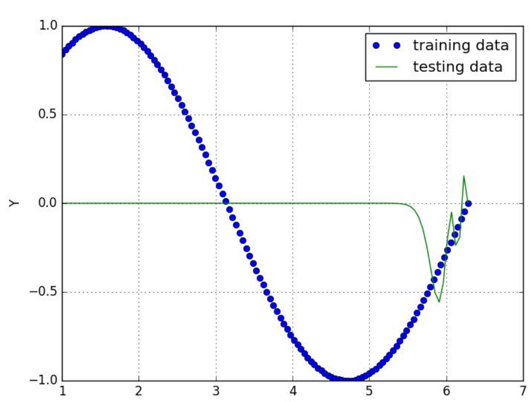


This is also the case for Degree > 15

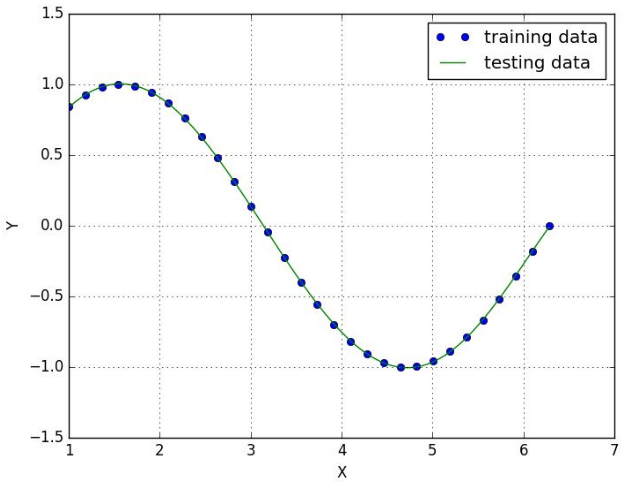
For large degrees of polynomial, the function seems to be behaving abnormally and the estimation is not accurate

And when we go on increasing the degree of the polynomial, the function estimation overshoots as shown in the above image

Case B : When the training samples < degree of polynomial



This image if for
#training samples = 120
Degree of basis function = 130
#testing samples = 100



The estimation is accurate for 4 < degree of polynomial < 9

This image if for
#training samples = 30
Degree of basis function = 6
#testing samples = 140

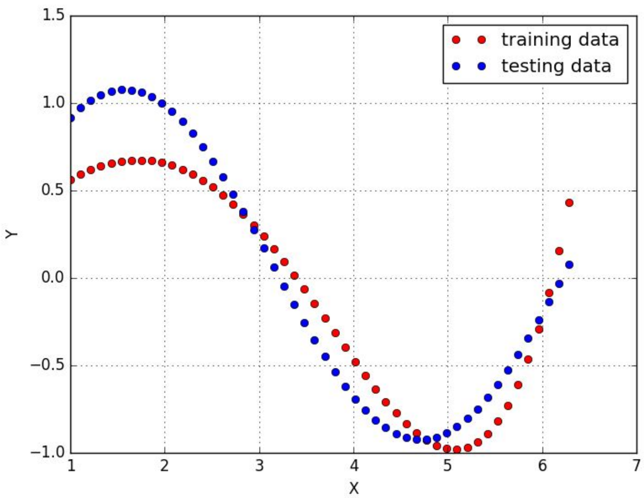
<u>For question 3</u>

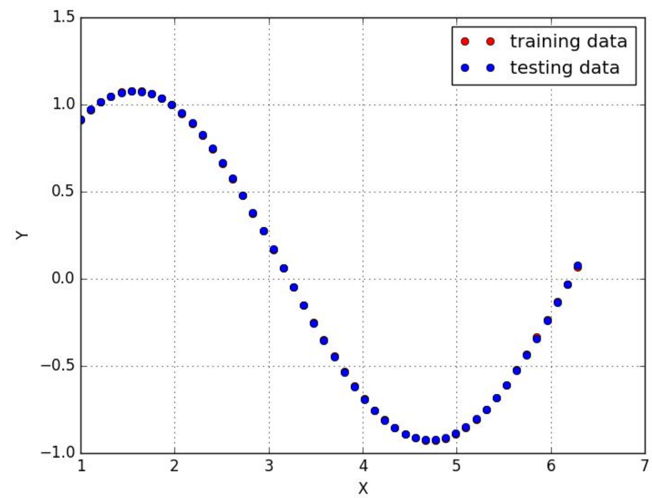
a) In this we use the L2 regularization method, where the entire problem is similar to using linear regression in polynomial basis function, but there is one more parameter in addition to that.

   The parameter ('Lambda' (say 'L')) is called the regularization parameter

- This parameter is used to reduce or prevent overfitting of parameters w.r.t the training data.

    - When the curve seems to be overfitting, this parameter alters the output values for the test inputs
    - Increasing the value of the regularization parameter decreases the overfitting only to a certain extent. If the value increases beyond the curve again overshoots at some point and the estimation would be completely inaccurate
    - For small values of the parameter, the curve seems exactly same and the curve overfits for some particular degree values of the polynomial function





- The difference with and without the regularization parameter can be seen in the above images

For 1st image

No. of training samples = 50
Degree of polynomial = 6
Values of parameter = 16
No. of testing samples = 50

For 2nd image

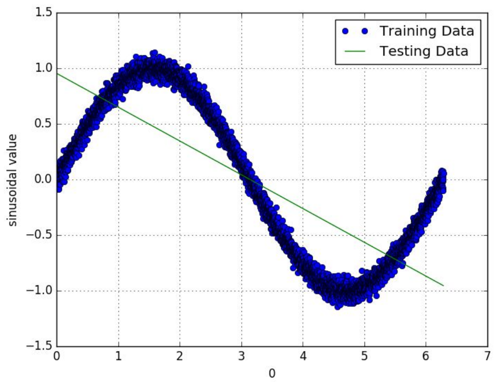
No. of training samples = 50
Degree of polynomial = 0
Values of parameter = 16
No. of testing samples = 50

<u>For question 4</u>

- Gaussian noise is added to the training labels conditioned on inputs and labels
  Each label is considered to be following a gaussian distribution with some standard deviation

- After maximizing the max likelihood, the values that we get for the parameter matrix will be the same as those as we get in minimizing squared error

- For the output of test data, we will get the function similar to that when we did not consider normal distribution over the variables.



Observation

The output will also have same distribution whose variance is near to that of the training data

_The above image is for the inputs

No. of training samples = 5000
No. test samples = 500

<u>For question 5</u>

Here we assumed that our Variable vector follows gaussian distribution conditioned on inputs and weights. But the weights themselves follow a gaussian distribution.

- We have seen in theoretical assignment that maximizing a-posterior distribution is same as applying L2 norm, but the regularization parameter is now changed as the ratio of variance of Labels and variance of parameter vector

- By changing the value of the variances of both of those distributions is like changing the value of the regularization parameter.

- Here, The entire thing is same, except we are adjusting the value of the regularization parameter by changing the values of variances of both the distributions

- Particular values of variances reduce the overfitting of curve and if the ratio is large, it also disturbs the correct estimation for new data

-