Observations

In all the implementations done above, I have changed the parameters and plotted or analysed the output.

For Q1.

PARAMETERS:

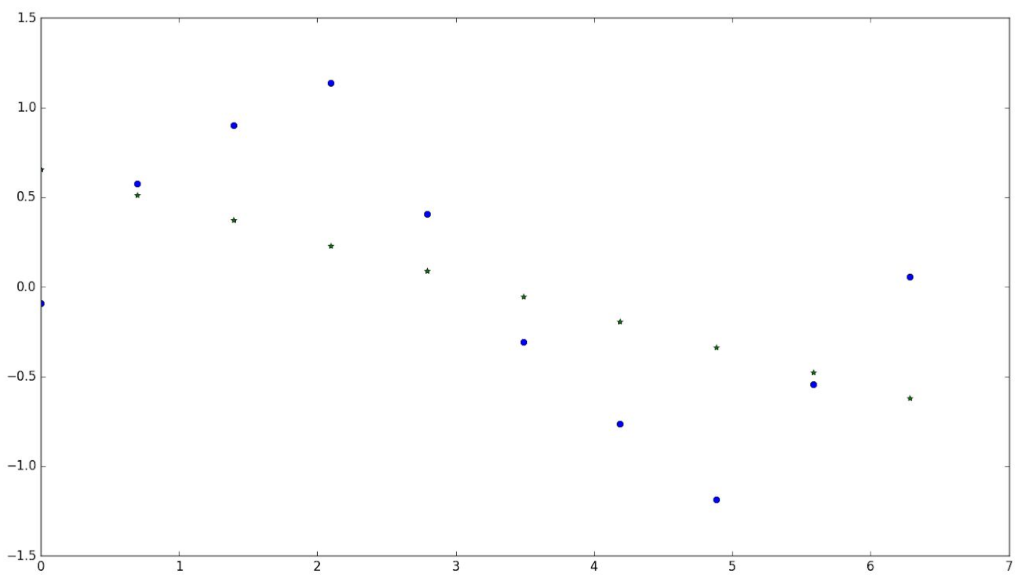
1. Noise variance which is being added to labels

Observations:

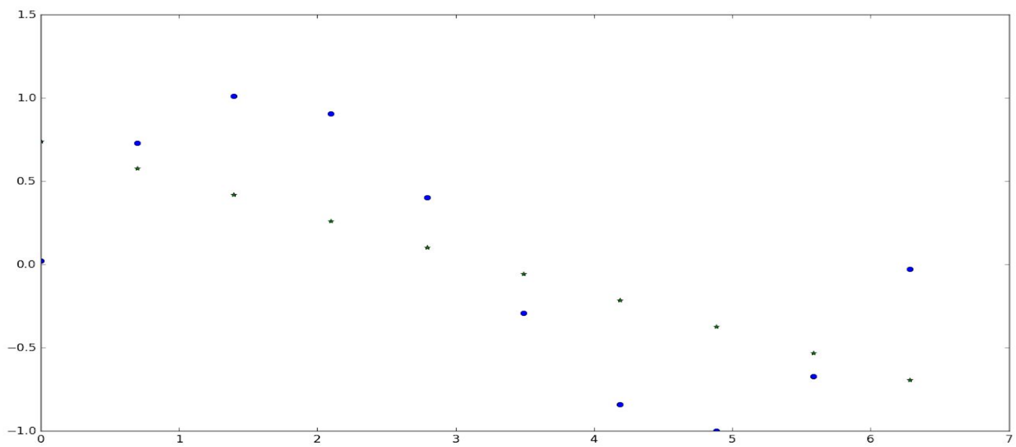
1) For additive noise to the labels y_train, If we increase the standard deviation of noise from .05 to .1 the error or cost function value increases.

   For .05 standard deviation error is 2.68656248
   For 0.1 standard deviation error is 3.33779392

Below are the train_y(dots[blue]) and y_predicted(star[green])



FOR STANDARD DEVIATION OF NOISE :0.1



FOR STANDARD DEVIATION OF NOISE :0.05

2) As the Number of training samples increases, Sum of squared error/N reduces and stagnates after a certain point

For 10000 samples Sum of squared error/N is 0.20170

For 100000 samples Sum of squared error/N is 0.2002725

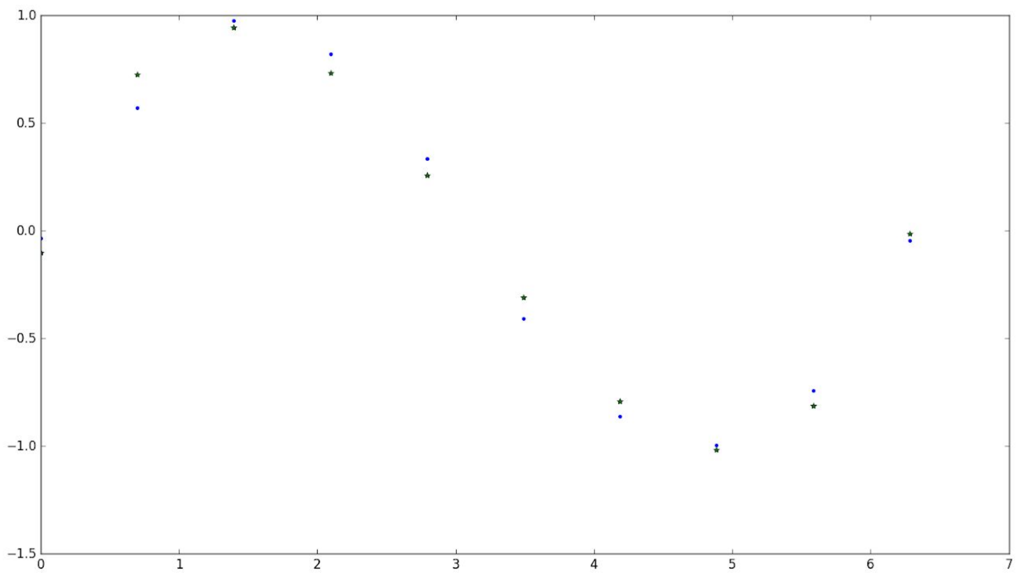
For Q2

PARAMETERS

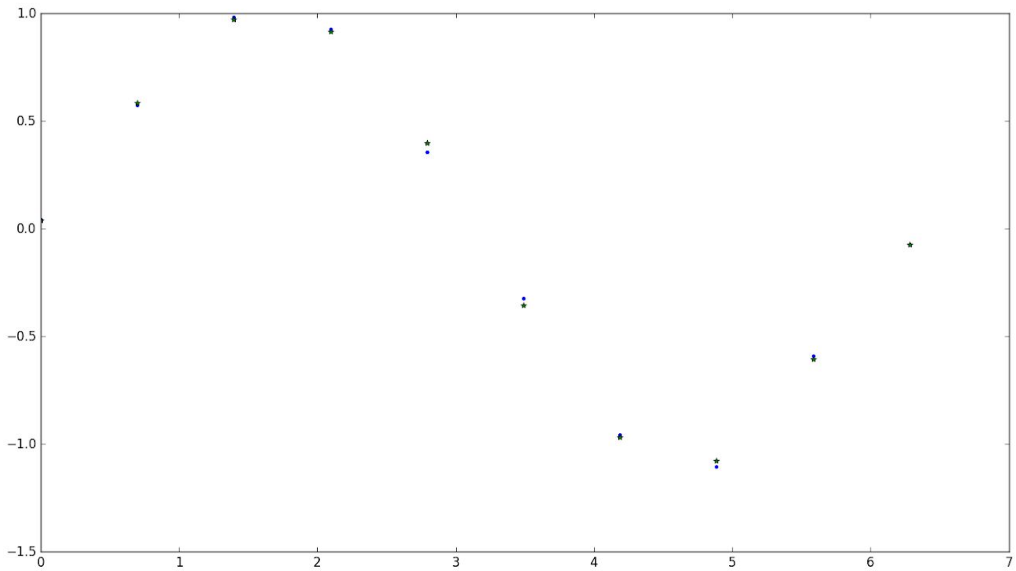
1. Noise variance which is being added to labels
2. Degree of the polynomial basis function

Observations:

1) The best order of polynomial basis found for 10 training samples found was 6-8
2) For the same order 3, On increasing training samples sum of squared error increases ie For N=10 SSE is 0.06466353 and For N=10 SSE is 0.82999
3) For additive noise to the labels y_train, If we increase the standard deviation of noise from .05 to .1 the error or cost function value increases.



Order 3, Y_train vs Y_predicted



Order 6, Y_train vs Y_predicted

For Q3

PARAMETERS

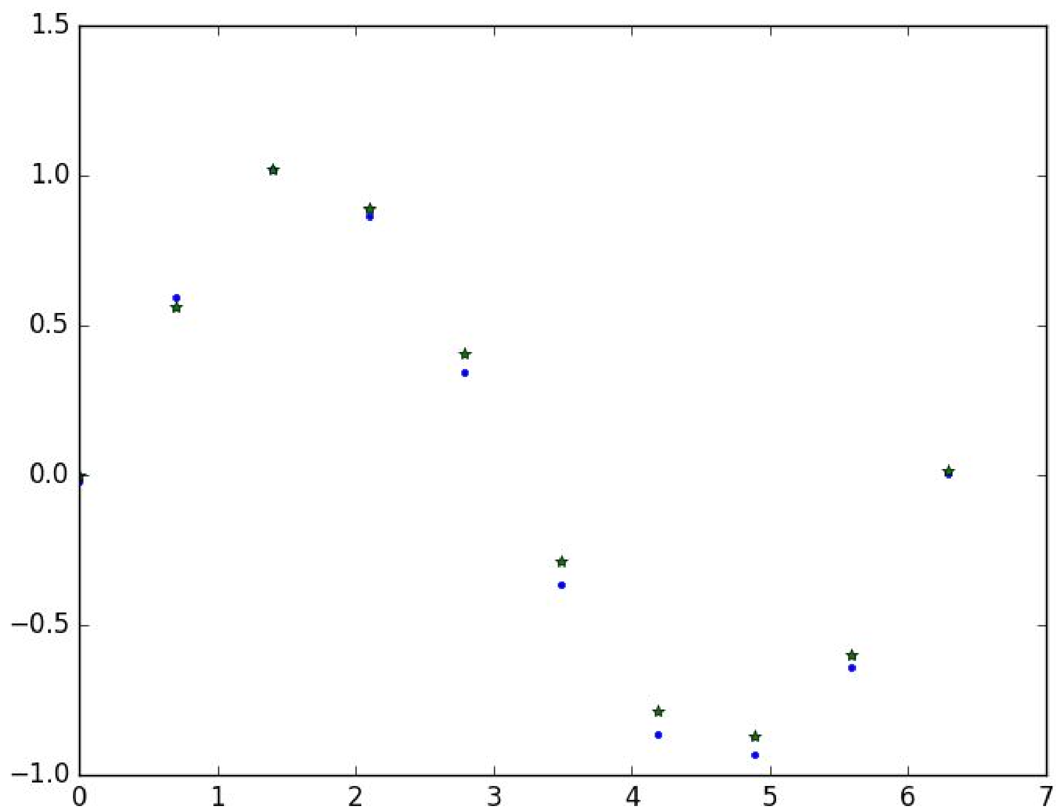
1. Clear difference between regularisation and Non regularised plot
2. Changing lambda and plot

Observations:

There has been an overfitting without L2 regularisation

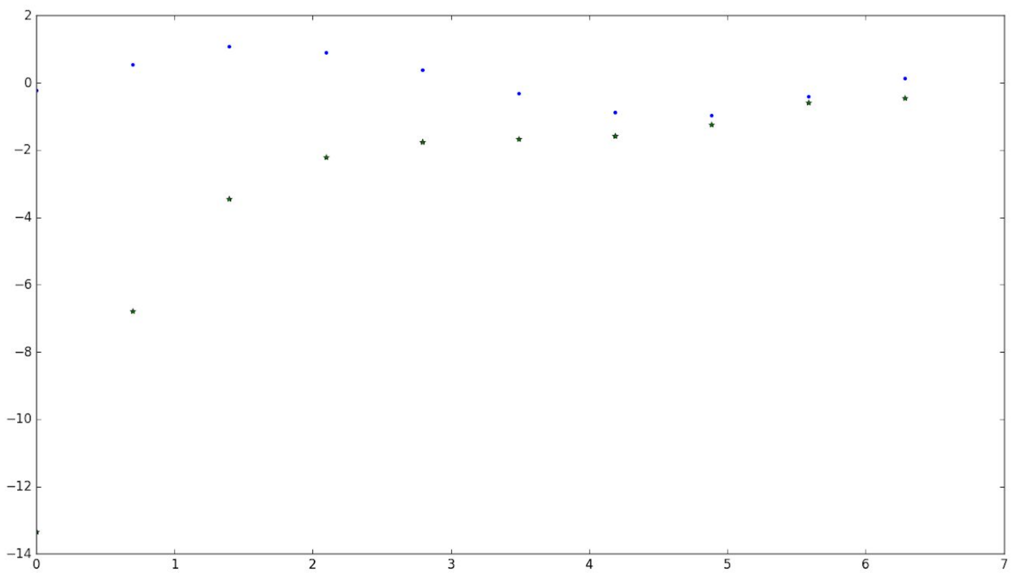
With Regularisation:

Lagrangian is .01, order is 15



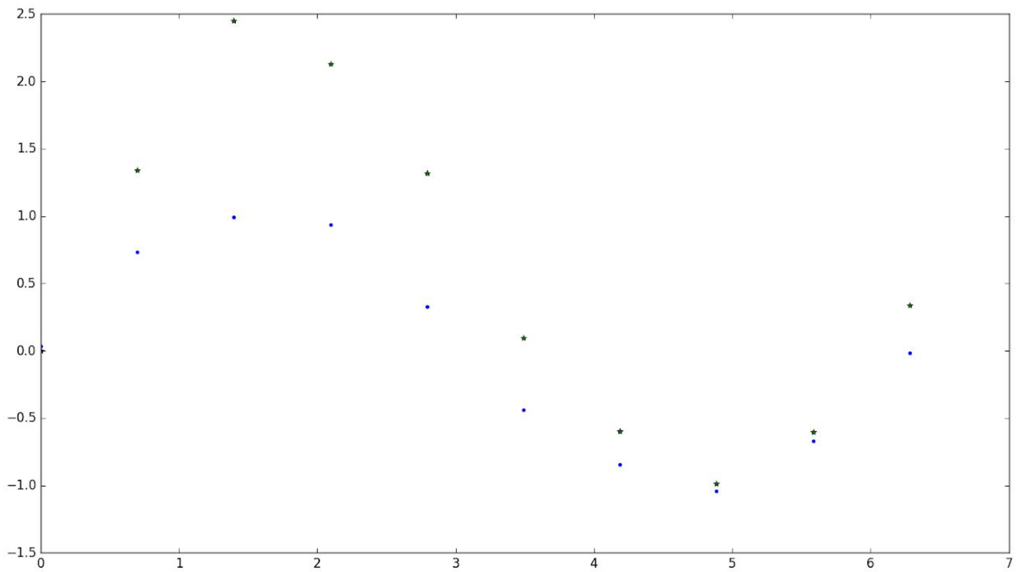
Without Regularisation:

order is 15



With Regularisation:

Lagrangian is 1, order is 15



Clearly lambda=.01 is the correct fit (N=10)

For Q4

PARAMETERS

1. Noise variance which is being added to labels
2. Observed the best variance fit from the cost function

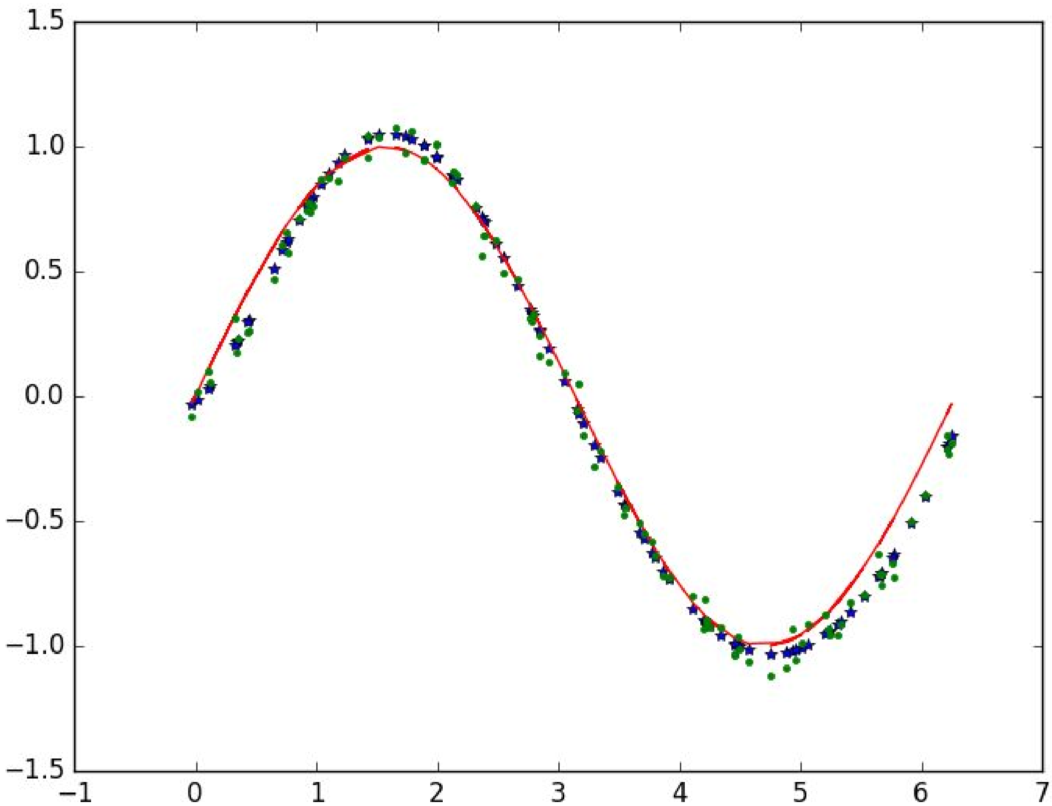
Observations:

1. For a given x,w we can derive the best variance of p(y|x,w,$\boldsymbol{\beta}$) gaussian distribution that can maximise the likelihood function p(y|x,w,$\boldsymbol{\beta}$)

   That variance has been given as user input by varying the variance we can observe the variations in variance of error ($\hat{y}$-y)

2. On addition of more standard deviation to noise adding to the training labels, error increases

3. Done for linear regression with polynomial basis function too



Red line -true estimates
*-predicted
.-estimated from mean and variance

For Q5

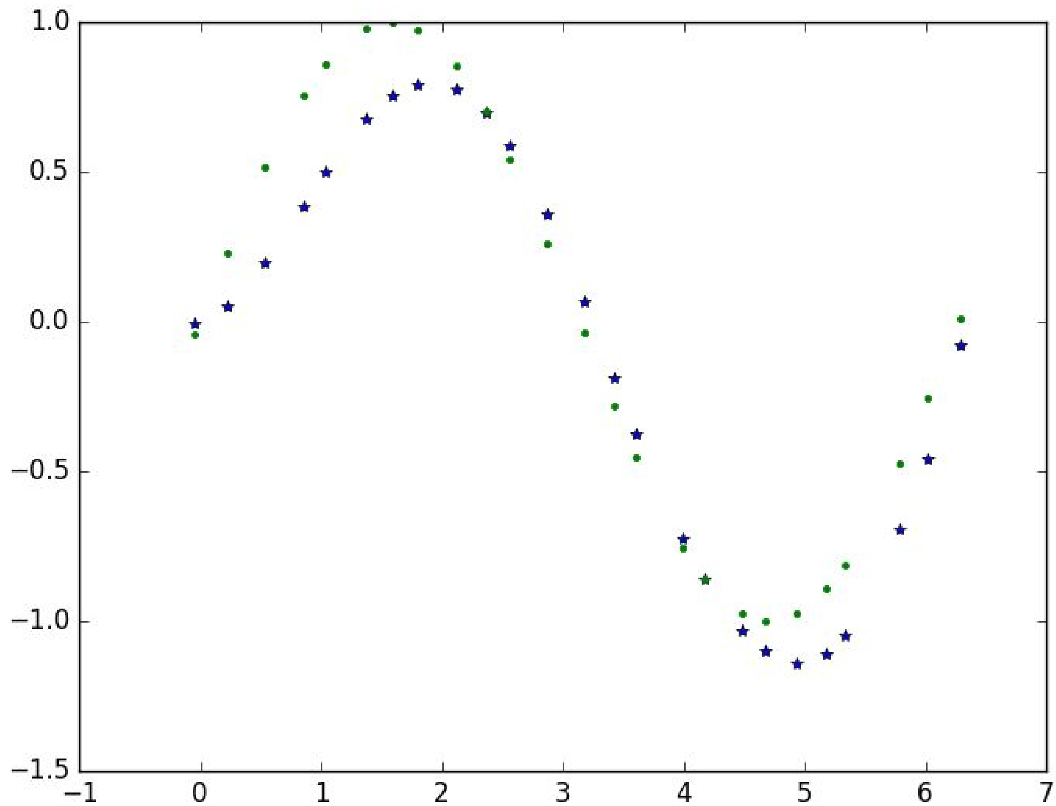
PARAMETERS

1. Varying α and σ we can attain the best fit for the polynomial
2. This is similar to varying lambda in problem 3

Observations:

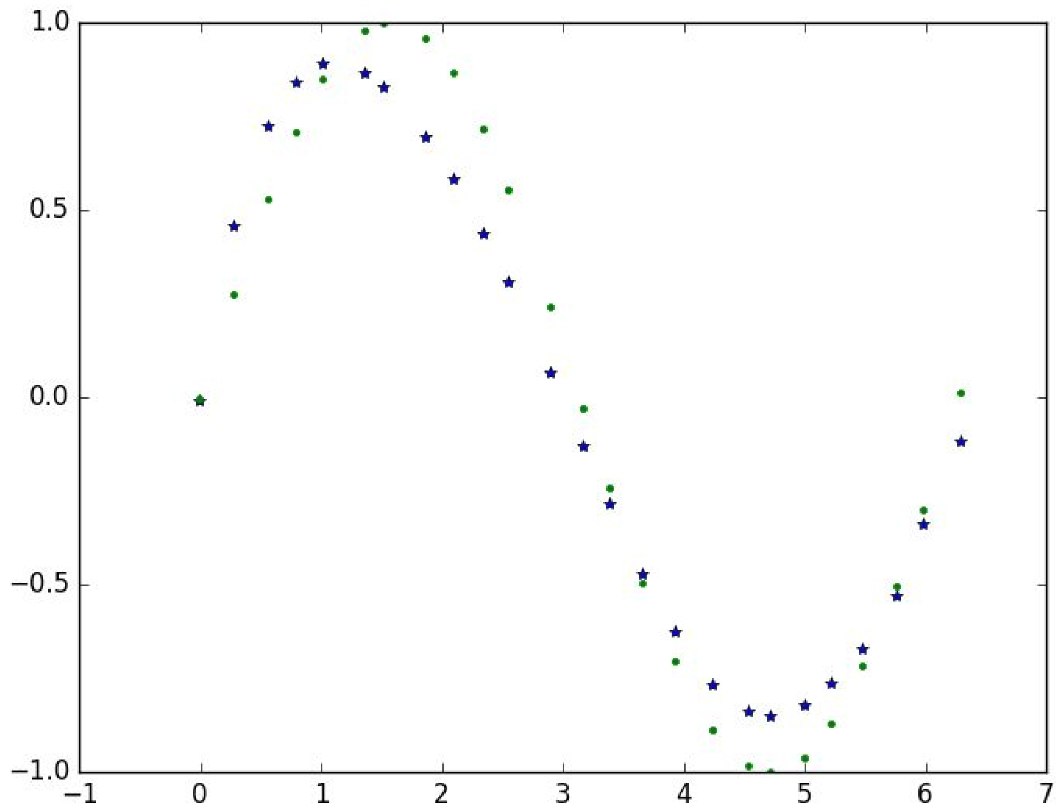
For N=25, order 15 , α **10 and** σ 1

Y_train vs Y_predicted



For N=25, order 15 , α **10 and** σ 2

Y_train vs Y_predicted



This way we can play around with parameters and find the best possible fit