PROJECT 1

LINEAR REGRESSION WITH BASIS FUNCTIONS

CSE 574 – INTRODUCTION TO MACHINE LEARNING

UNIVERSITY AT BUFFALO

Authors:

KEYUR SANJEEV JOSHI

keyurjos@buffalo.edu

ABSTRACT

OBJECTIVE

1. The objective is to learn how to map an input vector x into a target value t using the model

$$Y(x, w) = w^t \emptyset(x)$$

Methods

- Maximum Likely hood Approach
- Stochastic Gradient Descent
- Bayesian Linear Regression

INTRODUCTION

What is linear regression?

Linear regression is an approach for modeling the relationship between a scalar dependent variable 'y' and one or more explanatory variables denoted 'X'.X could be single random variable called as simple regression or X could be a vector of multiple input variables called multiple linear regression.

Linear regression means that the output is predicted using a linear combination of inputs variables or functions of input variables where these functions may be non-linear.

E.g.

$Y = w_0 + w_1 x$ is simple linear regression example which estimates the value of y as a function of x.

In more general form,

$$y(x, w) = w_0 + \sum_{j=1}^{M-1} w'_j \emptyset_j(x)$$

where ϕj(**x**) are known as basis functions and 'M' is called model complexity. A basis function is an element of a particular basis for a function space. Every continuous function in the function space can be represented as a linear combination of basis functions. Mostly non-linear functions are used for better approximation.

Typically, $\emptyset_0(x) = 1$, so that w_0 acts as a bias.To ensure better stability of our solution Gaussian functions are being used as the basis functions here.

$$\phi_j(x) = \exp\left\{-\frac{(x-\mu_j)^2}{2s^2}\right\}$$

This equation can be generalized for a multivariate input or vector input X as

$$\emptyset_j(x) = exp\left\{-\frac{((x-\mu^T)\sum^{-1}(x-\mu))}{2}\right\}$$

What is regularization?

Regularization involves adding a penalty term to the error function in order to discourage the coefficients from reaching large values. Otherwise over-fitting can occur.

APPROACH

The data set contains queries and urls represented by IDs, and feature vectors extracted from query-url pairs x along with relevance judgment labels t which are 0, 1, 2 with higher values indicating higher relevance. The entire data contains N = 69623 query-document pairs (rows) and d = 46 dimensions of features. This data is parsed and loaded as output vector and input matrix.

The key approaches to linear regression include

1. Maximum Likelihood

   Assumes observations are from a deterministic function with added Gaussian noise:

   $$t = y(\mathbf{x}, \mathbf{w}) + \epsilon \quad \text{where} \quad p(\epsilon|\beta) = \mathcal{N}(\epsilon|0, \beta^{-1})$$

   Given observed inputs, $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ and targets $\mathbf{t} = [t_1, \ldots, t_N]^{\mathrm{T}}$ the likelihood function using above approach is

   $$p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n), \beta^{-1}).$$

   Now maximizing likelihood

   $$\begin{aligned} \ln p(\mathbf{t}|\mathbf{w}, \beta) &= \sum_{n=1}^{N} \ln \mathcal{N}(t_n|\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n), \beta^{-1}) \\ &= \frac{N}{2}\ln\beta - \frac{N}{2}\ln(2\pi) - \beta E_D(\mathbf{w}) \end{aligned}$$

$$E_D(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\}^2$$

Where

$$\nabla_{\mathbf{w}} \ln p(\mathbf{t}|\mathbf{w},\beta) = \beta\sum_{n=1}^{N}\left\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\right\}\boldsymbol{\phi}(\mathbf{x}_n)^{\mathrm{T}} = \mathbf{0}.$$

Additionally considering regularization

$$E_D(\mathbf{w}) + \lambda E_W(\mathbf{w})$$

W is estimated as

$$\mathbf{w} = \left(\lambda\mathbf{I} + \boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{t}.$$

Where

$$\boldsymbol{\Phi} = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \cdots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{pmatrix}.$$

Is called as design matrix.

The data-set has been divided into training set and testing set.80% of data has been used for training the model. The mean vector set has been generated by selecting a set of dataset rows. Here the variance value is s^2 for the entire training set.

2. Stochastic Gradient Descent

This is an iterative approach for reducing mean error. The weight vectors are initialized to some default value to achieve this. Randomly an input vector is picked up from training set and using w estimate of y (x, w) is obtained the error is calculated as (t_n-y).

$$\begin{aligned} \mathbf{w}^{(\tau+1)} &= \mathbf{w}^{(\tau)} - \eta\nabla E_n \\ &= \mathbf{w}^{(\tau)} + \eta(t_n - \mathbf{w}^{(\tau)\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n))\boldsymbol{\phi}(\mathbf{x}_n). \end{aligned}$$

w has been initialized to zero in this case and η = 1 or 0.5 depending on the error.

i) Here the design for CFS is that the mean vectors are means of a set of rows from the training data set. The value of a variance of the entire data set multiplies by identity matrix is being used as the covariance matrix. Tuning was using achieved by varying M and λ.

ii)For the stochastic gradient part η has been taken as 1 and may be adjusted as the per error for better tuning of the model.

RESULTS

In this section, we discuss the results we obtained using the linear regression techniques on the given dataset.

Maximum Likely hood

The observed values of RMS Error for different values of model complexity of M are as follows:

Model Complexity	RMS Error
5	0.5643
7	0.5640
10	0.5634
12	0.5633
20	0.5633
50	0.5633

Table 1: Mean Square Error against Training set

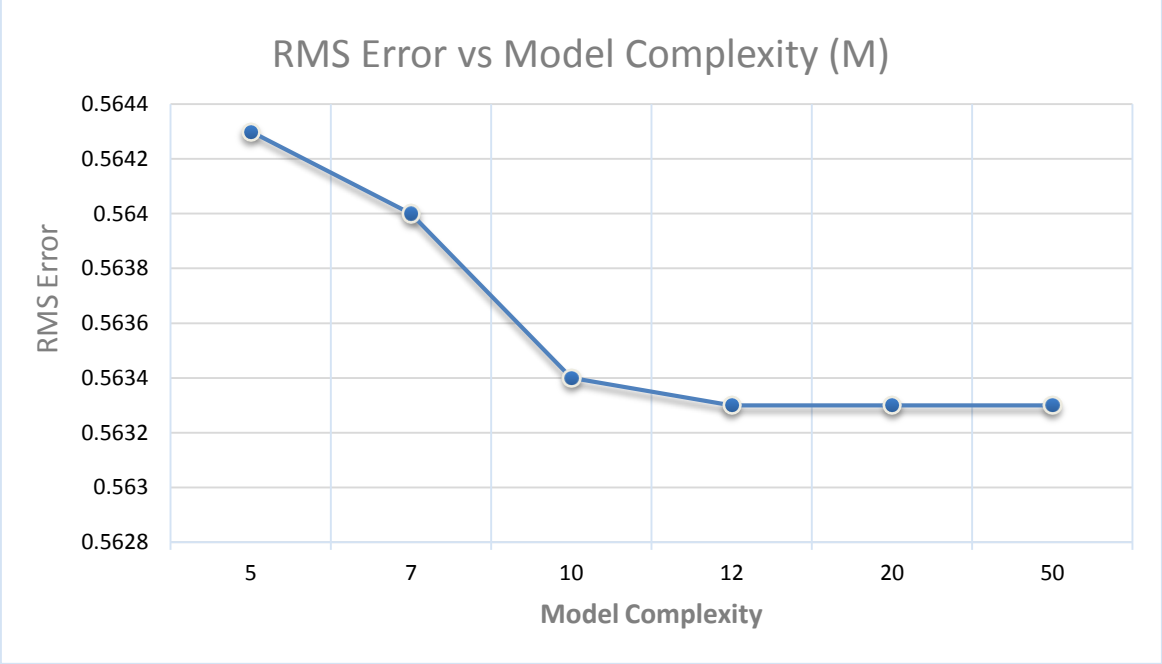

Figure 1: Plot of Model Complexity against RMS error for training set

The plot shows that RMS value of error decreases at the start and then stabilizes to a constant value of 0.5633.Thus it is seen that error rate does not decrease after a certain threshold even after increase in model complexity.

Model Complexity	RMS Error
5	0.5982
7	0.5978
10	0.5973
12	0.5971
20	0.5971
50	0.5972

Table 2: Mean Square Error against test set

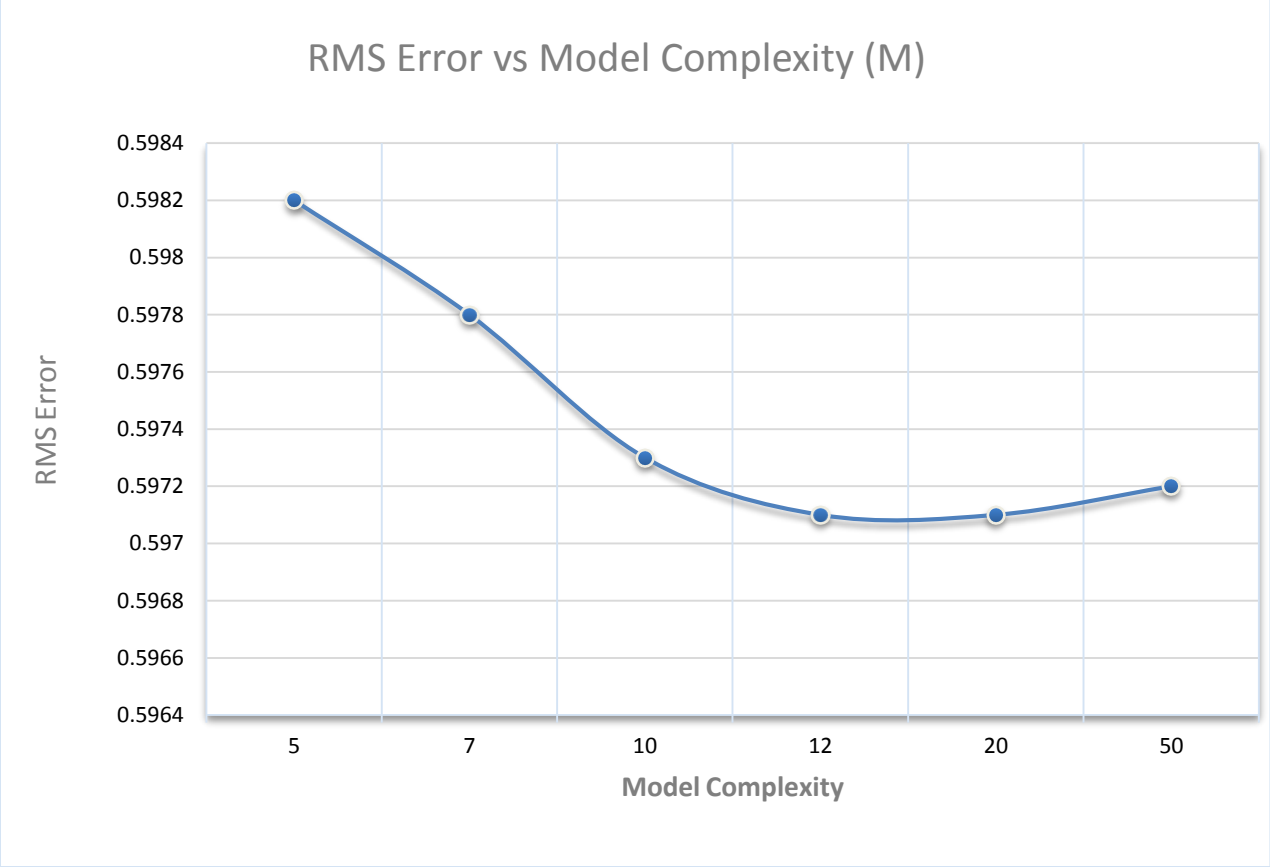

Figure 2: Plot of Model Complexity against RMS error for test set

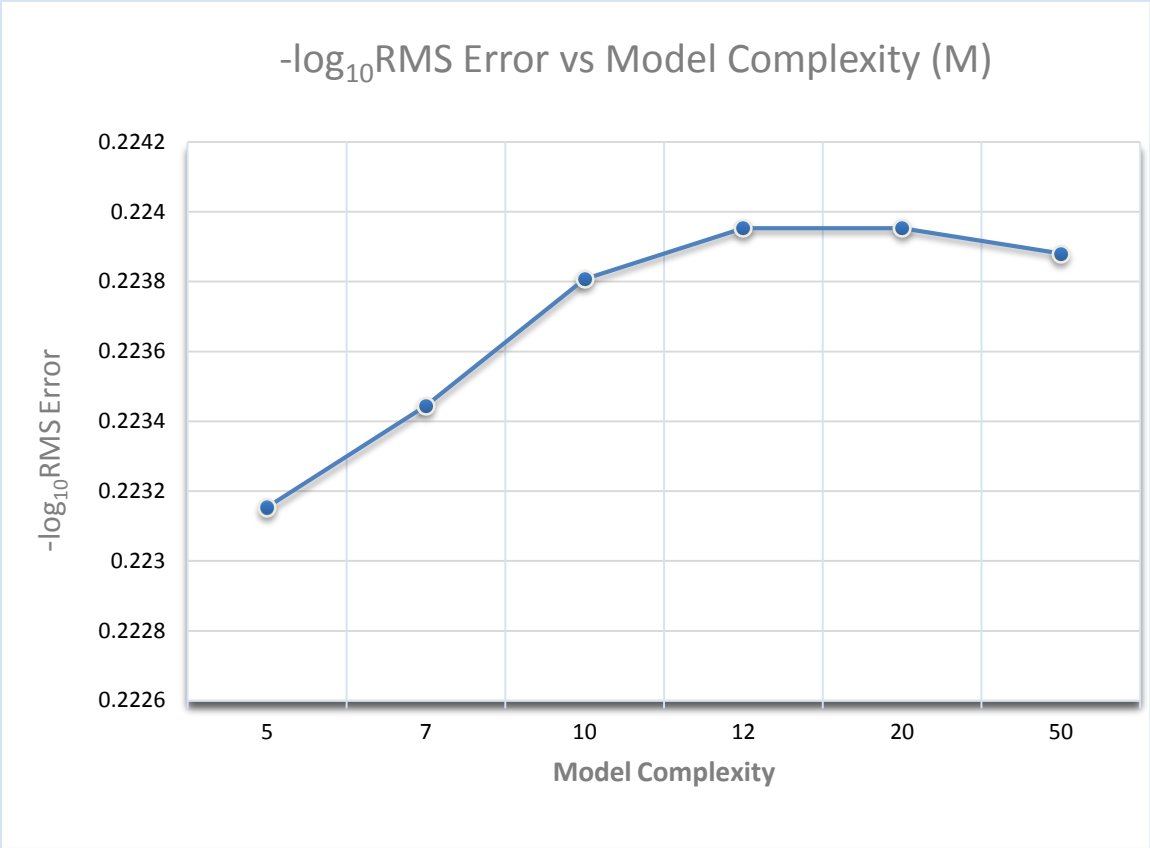

Figure 3: Plot of Model Complexity vs**.** – (**Log_{10} RMS** Error)

From the figures above, we observe that error decreases with increasing model complexity. However, the decrease is not continuous rather a slight increase is seen for higher 'M' indicating over fitting on training set. However, the semi-log plot shows a linear behavior with increasing slope in some sections.

Regularization Coefficient λ	RMS Error Test
0	0.5969
0.0005	0.5969
0.5	0.5971
1	0.5971
2	0.5973
5	0.5974

Table 3: Value of RMS Error for different λ and M=12

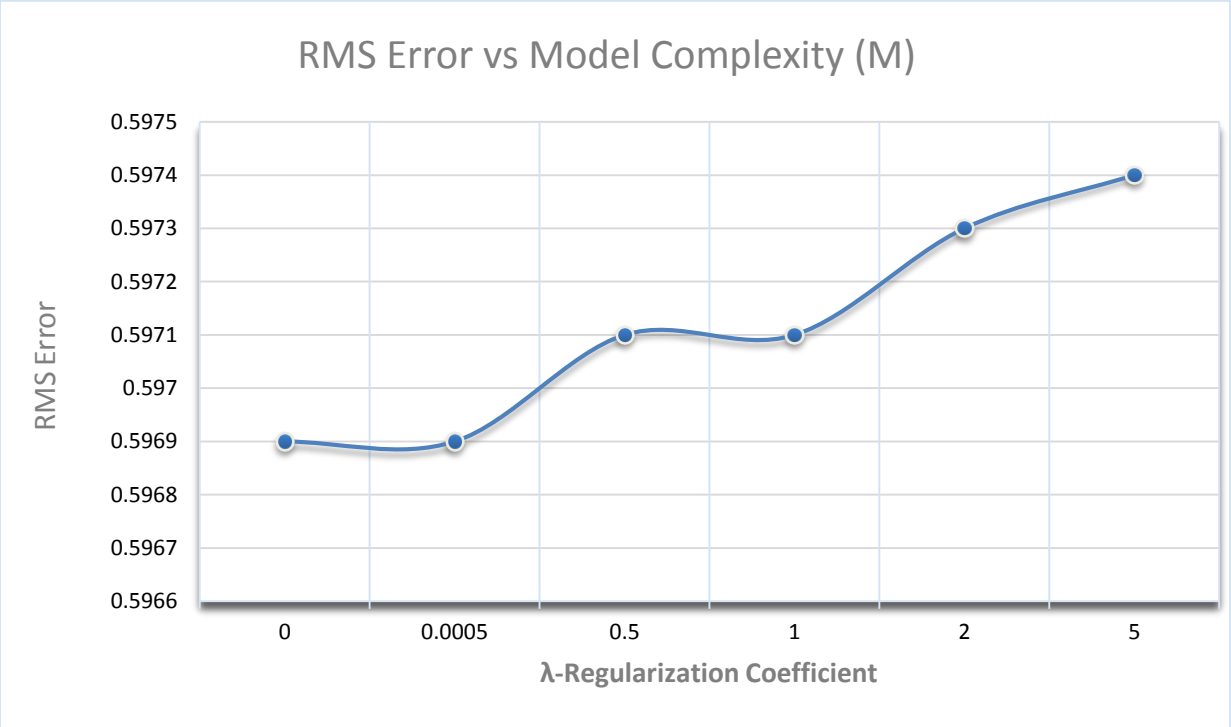

Figure 3: Plot of RMS Error vs. Model Complexity

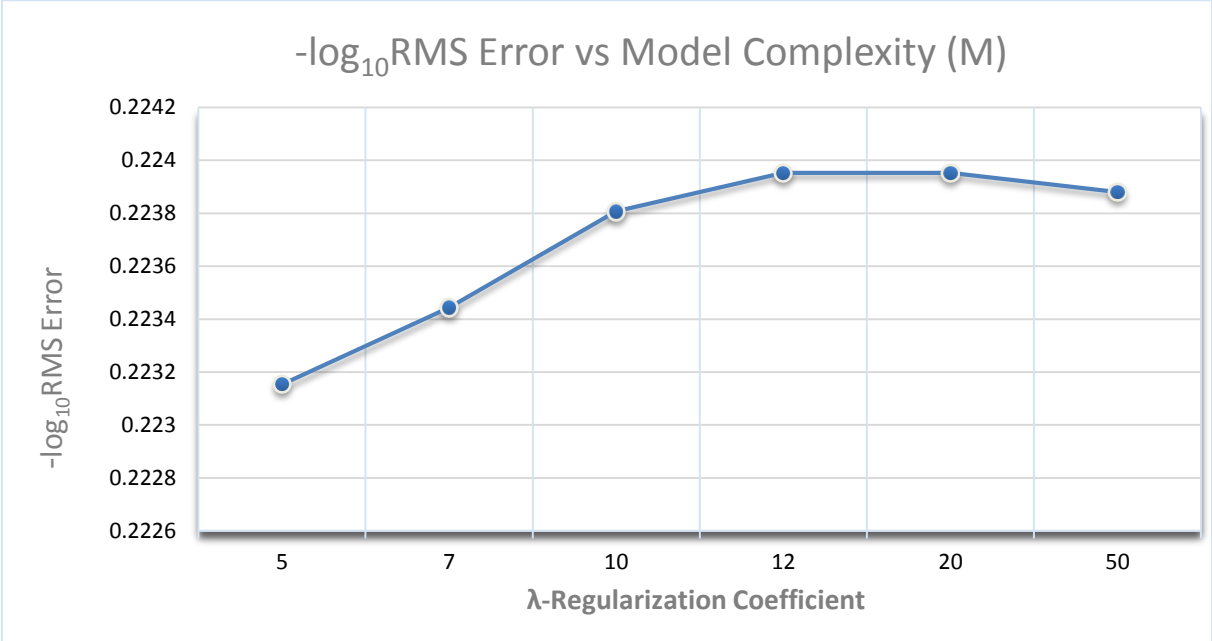

Figure 4:- Plot Log_{10} RMS Error vs. λ

The figures above RMS Error shows no real explicit relation between the model complexity and regularization coefficient .But in general and increase in E_{RMS} is seen as λ increases.

Gradient Descent

The value of RMS error was calculated for different values of model complexity M.M was varied ranging from 5 to 35 .The results are as follows.

Model Complexity	RMS Error Train	RMS Error Test
10	0.6439	0.6882
15	0.6436	0.6880
20	0.6444	0.6887
30	0.6450	0.6682
35	24.23	24.4

η_b=1

Table 3: RMS Error vs. Model Complexity

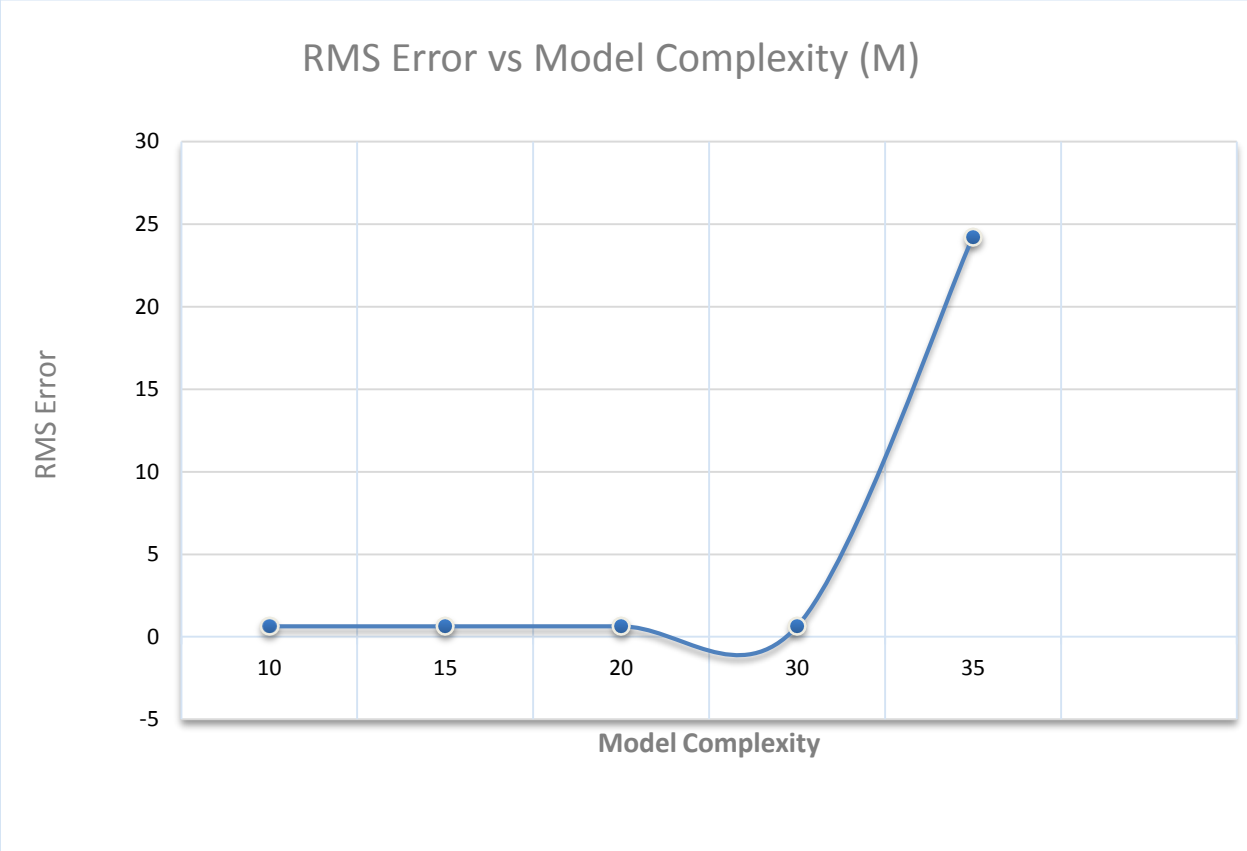

Figure 5: Plot of E_{RMS} vs. Model Complexity

The graph for stochastic gradient descent decreases slightly with increasing value of M reaching minimum once for M=15 and then again for M=34.Its observed that the error values keeps oscillating in between this range.

DISCUSSION

In this section, we discuss the advantages, challenges and possible improvements of the regression algorithm.

ADVANTAGES [1]

1) The use of Gaussians as basis functions helps in better predictions of output for given input. Being a radial function, any change in x remains local unlike a polynomial regressor.
2) Efficiently generates output with multiple parameters that allow better tuning of model.

CHALLENGES OF ALGORITHM [1]

1) Selecting set of M different μ_j for each basis function $\emptyset_j(x)$
2) Select a variance s^2
3) Selecting no of iterations/error threshold for stochastic gradient method

IMPROVING THE PREDICTION OF OUTPUT, GENERAL IDEAS [1]

1) Try different functions for obtaining the set of means
2) Change the variance value
3) Facilitate support counting of candidates

CONCLUSION

The project was implemented successfully in accordance with the requirements

REFERENCES

[1] Pattern Recognition and Machine Learning- Christopher Bishop