CS6011: Kernel Methods for Pattern Analysis **Even 2015**

Assignment I Report

5/02/2015

Charu Chauhan, CS14M058
Devi G, CS14M011
Ditty Mathew, CS13D018

1 Objective

The objective of this assignment is function approximation using given datasets. The datasets given are univariate and bivariate data. The goal is to perform the following models.

1. Polynomial curve fitting for Dataset 1 with different sizes, different model complexities and different values of regularization parameter.
2. Linear model for regression using Gaussian basis functions for Dataset 2 with different model complexities and different values of regularization parameter.

2 Polynomial curve fitting for Dataset 1

Given function, $f(x) = cos^2(2\pi x)$. Its plot has shown in Figure 1.

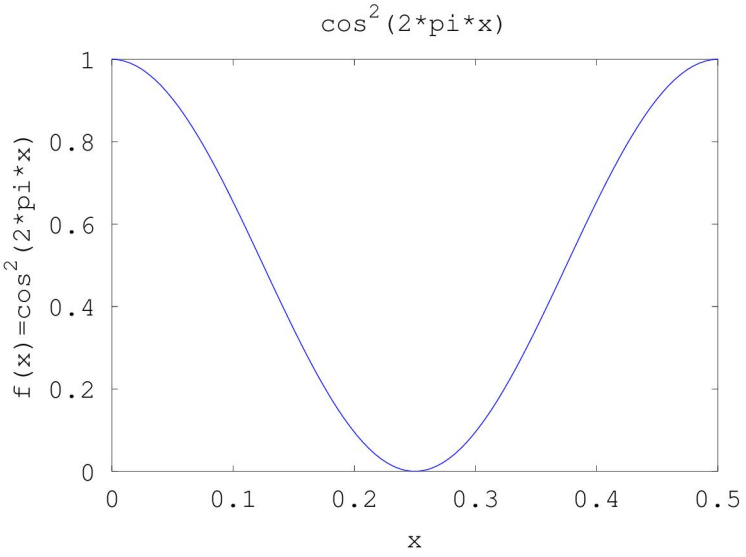

Figure 1: Underlying Function

2.1 Dataset generation

The dataset was generated for different sizes using $f(x)$ for $x\epsilon(0, 0.5)$ along with gaussian noise having mean zero, and standard deviation 0.1.

2.2 Curve fitting without regularization

The dataset of sample size 10 is regressed for different model complexities. Figure 2 to Figure 5 corresponds to the plot for model complexities 0,1,3,6,9 respectively. Table 1 shows the values of the coefficients $w*$ obtained for polynomials of different degrees.

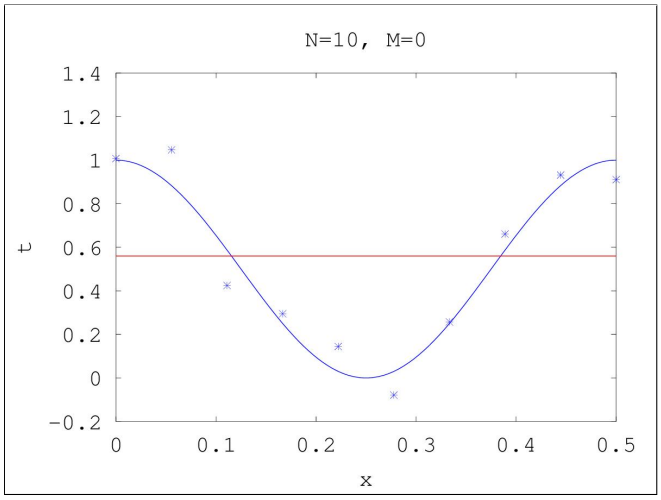

Figure 2: Curve fitting with N=10, M=0

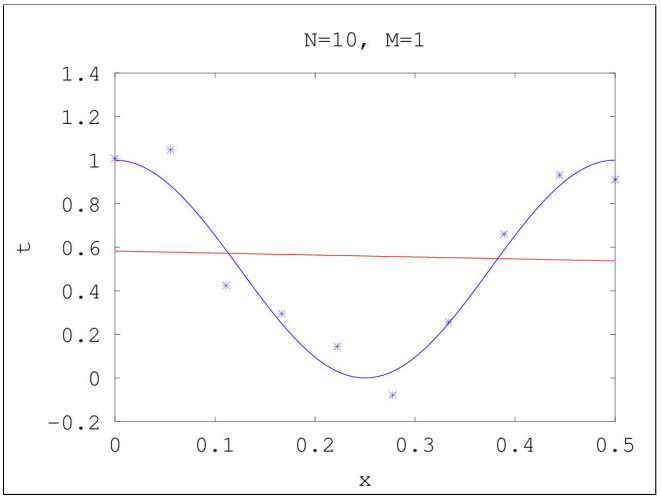

Figure 3: Curve fitting with N=10, M=1

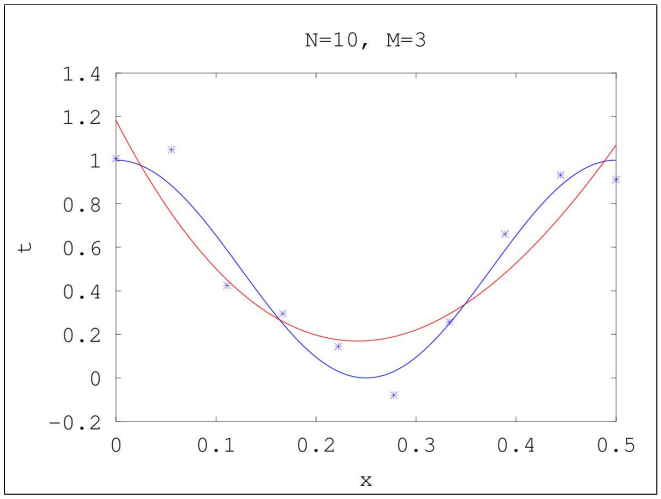

Figure 4: Curve fitting with N=10, M=3

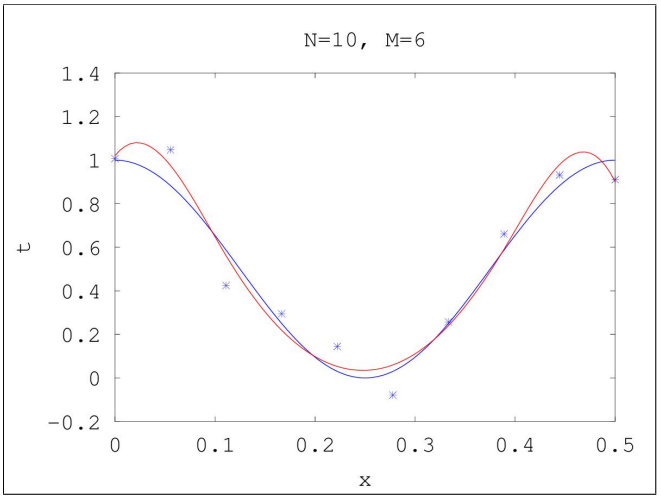

Figure 5: Curve fitting with N=10, M=6

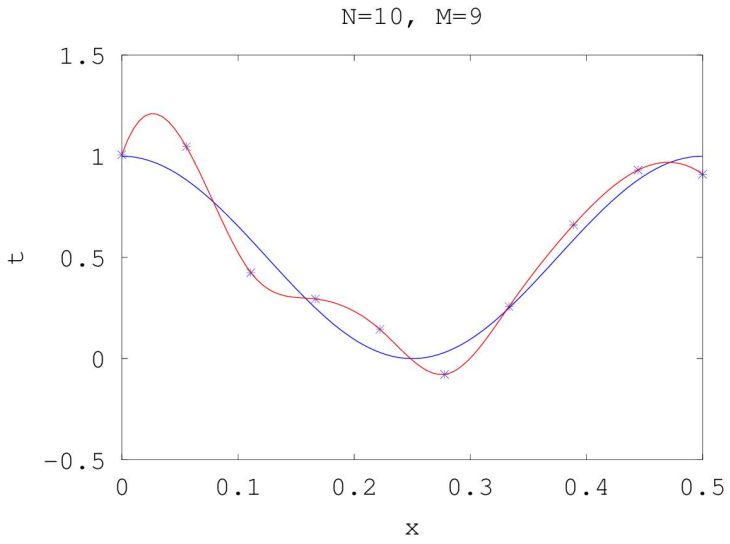

Figure 6: Curve fitting with N=10, M=9

Observations

- When M is less, curves are less complex and underfits the training data.
- As M increases, the model's approximation of the training data increases.
- At M=6, polynomial looks like best fit to the undelying function.
- At M=9, the model overfits the training data.

The values of weights w for $N = 10$ and M varies between 0 and 9 are shown in *Table* 1.

	M=0	M=1	M=3	M=6	M=9
w_0*	0.55987	0.582542	1.1822	1.0202	1.0068e+00
w_1*		-0.090682	-8.8389	7.8639	-8.7896e+00
w_2*			21.1171	-227.0515	1.0245e+03
w_3*			-7.7791	1527.8295	-2.9106e+04
w_4*				1527.8295	3.4771e+05
w_5*				-4955.9347	-2.1994e+06
w_6*				8288.8842	7.9586e+06
w_7*				-5603.1637	-1.6543e+07
w_8*					1.8378e+07
w_9*					-8.4572e+06

Table 1: Table of the coefficients w* for polynomials of various order

Observations

- As M increases, the magnitude of the coefficients gets larger.
- At M= 9, large positive and negative values of the coefficients makes the curve fits exactly to the training data.

In the next experiment, we examined the behavior of a model, as the size of the dataset is varied. We experimented with N=15 and N=100. The plots are shown in Figure 7 and 8.

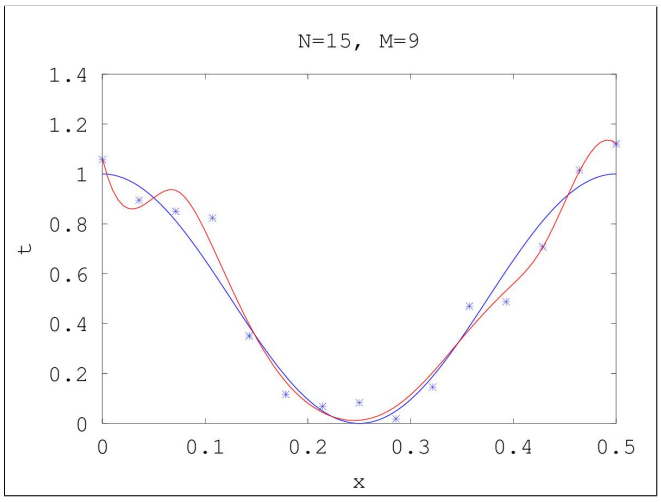

Figure 7: Curve fitting with N=15, M=9

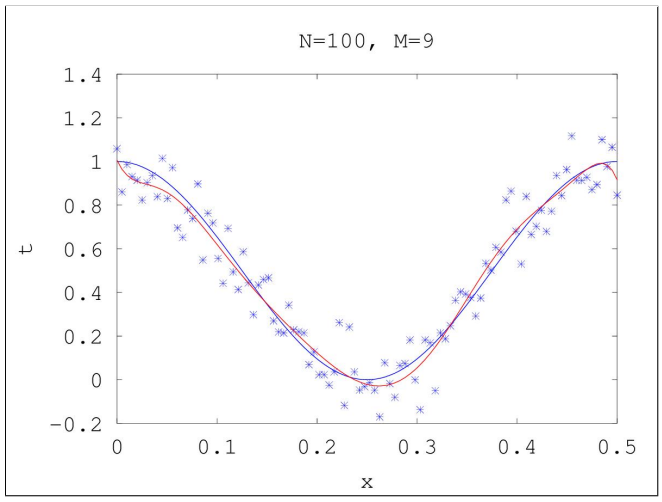

Figure 8: Curve fitting with N=100, M=9

Observations

- For the same model complexity(M=9), over-fitting problem has reduced as the number of samples increases.
- At N=100,the underlying function is approximated closely without over-fitting.

2.3 Curve fitting with Regularization

Regularization is a technique to control over-fitting.Over fitting occurs when the size of the training data is less than ten times the number of parameters to be estimated. In curve fitting with regularization, the error function is defined as follows,

$$\hat{E}(w) = \frac{1}{2}\sum_{n=1}^{N}(y(x_n, w) - t_n)^2 + \frac{\lambda}{2}\|w\|^2 \tag{1}$$

λ is the regularization parameter. For this experiment, we chose the sample size as 10 and the model complexity as 9 to study about the effect λ in overcoming over-fitting. At $\lambda = 0$, the plot is same as without regularization. In Figure 9, $ln\,\lambda = -18$ which takes λ as a value in between 0 and 1. Figure 10 corresponds to $\lambda = 1$

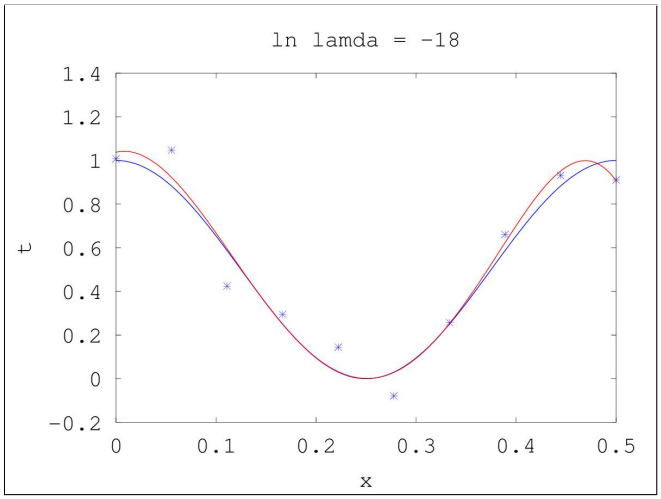

Figure 9: Plot for N=10, M=9, ln λ=-18

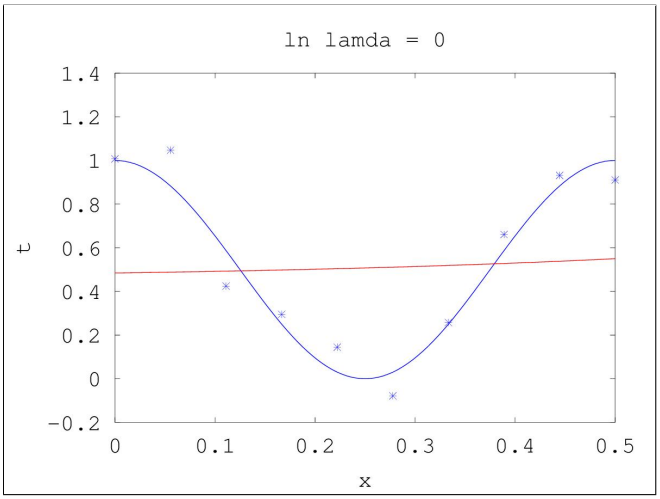

Figure 10: Plot for N=10, M=9, ln λ=0

Observations

- As λ increases,the coefficients of w decreases and we get a smoother fit.
- At $ln\,\lambda = -18$, over-fitting has suppressed.
- For small values of λ, the modeled output will over-fit the given data(At $\lambda = 0$, performance is equivalent to M=9 without regularization)
- For large values of λ, the modeled output will under-fit the given data(At $\lambda = 1$, performance is equivalent to M=1 without regularization).

The values of weights w for $N = 10$, $M = 9$ and $ln\,\lambda = [-\infty, -18, 0]$ are shown in $Table\,2$.

	$ln\,\lambda = -\infty$	$ln\,\lambda = -18$	$ln\,\lambda = 0$
w_0*	1.0068e+00	1.0377	4.8445e-01
w_1*	-8.7896e+00	1.2121	6.5134e-02
w_2*	1.0245e+03	-72.3511	9.0709e-02
w_3*	-2.9106e+04	247.9577	6.0141e-02
w_4*	3.4771e+05	-149.0152	3.3113e-02
w_5*	-2.1994e+06	-95.2612	1.6983e-02
w_6*	7.9586e+06	-57.6996	8.4408e-03
w_7*	-1.6543e+07	-41.7644	4.1346e-03
w_8*	1.8378e+07	-26.0852	2.0122e-03
w_9*	-8.4572e+06	-12.0865	9.7708e-04

Table 2: Table of the coefficients w* for polynomials of various order

Observations

- As λ increases, weights are getting smaller

2.4 Root Mean Square Error(RMSE)

RMSE is the measure of the difference in the target output and model output. This measure allows us to compare different sizes of data sets.

$$E_{RMS} = \sqrt{\frac{\sum_{n=1}^{N} (y(x_n, w) - t_n)^2}{N}} \tag{2}$$

Plots of the training, validation and test set RMS errors are shown, for various values of M, in *Figure* 11 and various values of λ, in *Figure* 12.

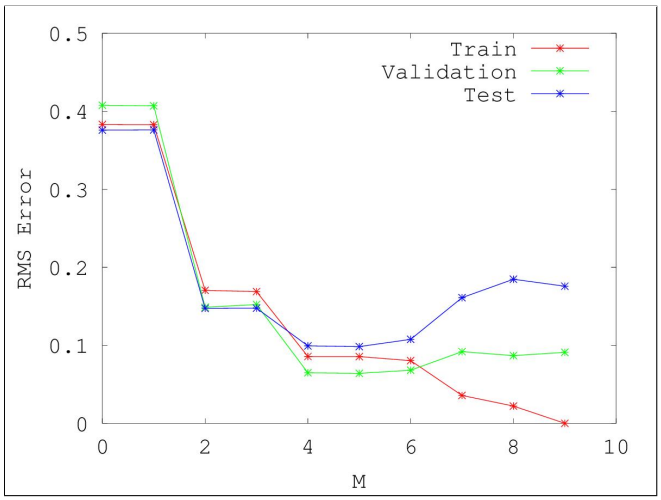

Figure 11: RMS Error on various values of M

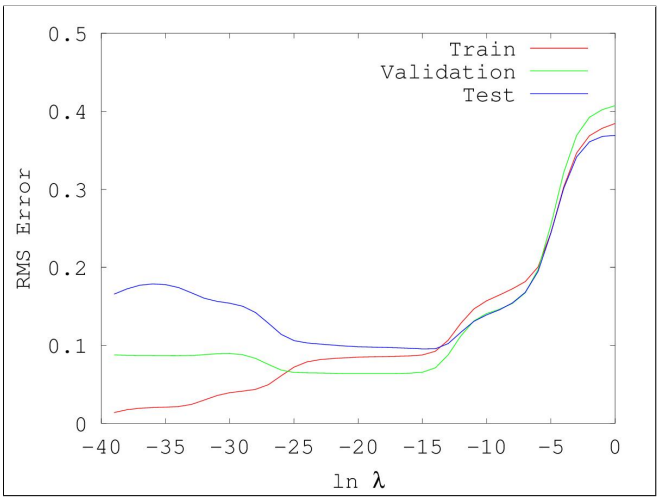

Figure 12: RMS Error versus ln λ for M=9

Observations

- In *Figure* 11,the train, validation and test error is decreasing until M=6. After M=6, test error is increasing and training error is decreasing. This indicates that model tends to overfit the training data after M=6.
- At $M = 9$, training error is zero, but test error and validation error is there.
- $M = 6$ is the preferred model complexity for this sample data because, at this point training and validation error is less.
- In *Figure* 12, we can observe that regularization reduces the problem of over-fitting for $M = 9$.
- For small values of λ training error is almost close to zero.
- As λ increases train, validation and test error also increases. This indicates that over-fitting is reducing by increasing λ.
- For large values of λ, all the errors are high.

2.5 Scatter plot with target output on x-axis and model output on y axis

Scatter plot with target output vs model output for training data ,validation data and test data are shown in *Figure* 13, 15 and 17 respectively, for different model complexities. *Figure* 14, 16 and 18 corresponds to training data ,validation data and test data respectively for various values of λ

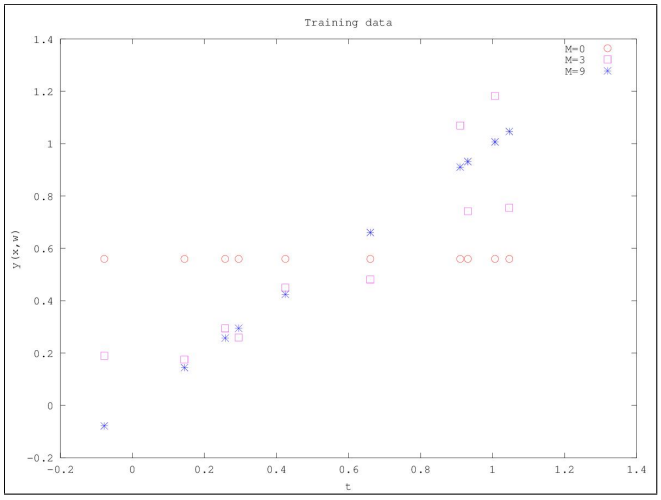

Figure 13: Scatter plot for various values of M in training data

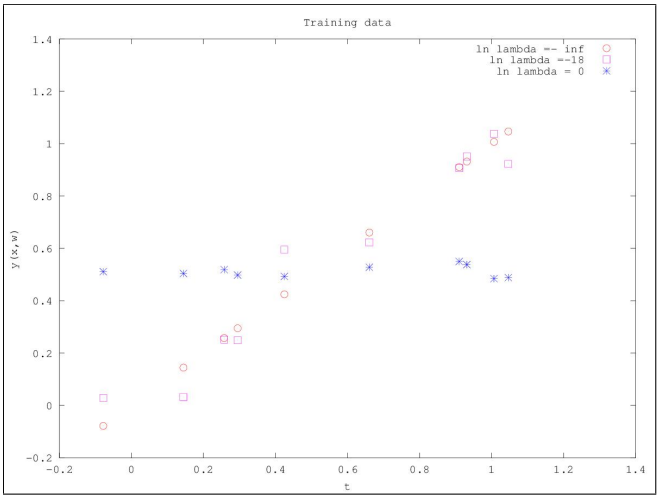

Figure 14: Scatter plot for various values of λ in training data

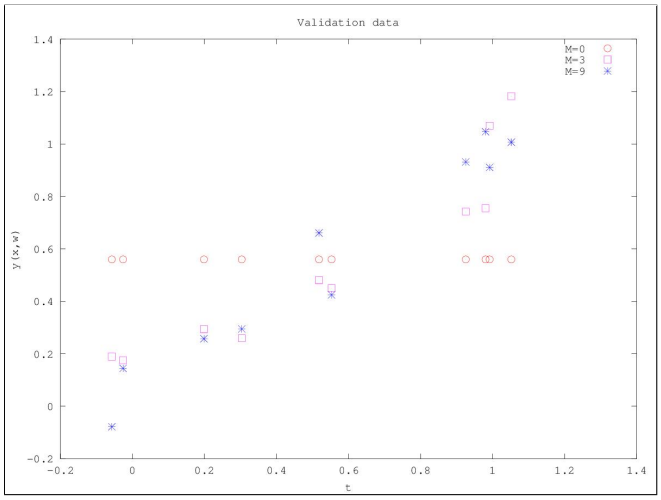

Figure 15: Scatter plot for various values of M in validation data

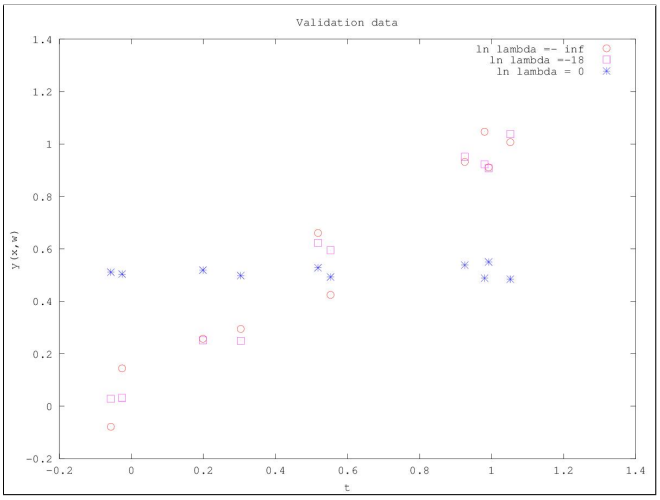

Figure 16: Scatter plot for various values of λ in validation data

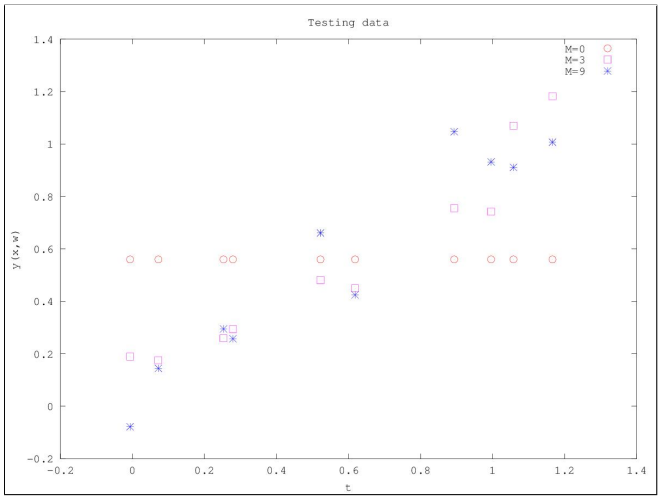

Figure 17: Scatter plot for various values of M in test data

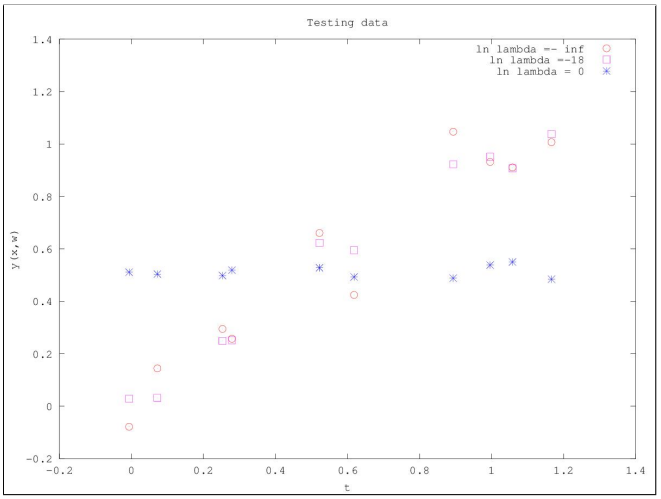

Figure 18: Scatter plot for various values of λ in test data

Observations

- For training data, as M increases, points are coming close to the line $y(x, w) = t$ from $y(x, w) = c(c\ 0.4)$. Same thing happens when λ decreases. Points along $y(x, w) = c(c\ 0.4)$ indicates under-fitting and $y(x, w) = t$ indicates over-fitting.
- For validation data and test data, points are not exactly in line $y(x, w) = t$ for $M = 9$ and also for $ln\lambda = -\infty$ when compared to training data.

3 Linear Regression Model using Gaussian Basis Function

The linear regression can be performed by using a model that is a linear combination of input variables.

$$y(x, w) = w_{\mathrm{o}} + w_1 x_1 + ... + w_{\mathrm{D}} x_{\mathrm{D}} \tag{3}$$

The model given by equation 3 is also known as linear regression.This model has advantage of simplicity but it has serious limitations.To overcome the shortcomings by using a model which uses linear combination of fixed non linear functions known as the basis function.The models take the form

$$y(x, w) = w_{\mathrm{o}} + \sum_{j=1}^{M-1} w_j \phi_j(x) \tag{4}$$

The fixed non linear functions that are used to model the regression problem are known as the basis functions.There are various basis functions that can be used like gaussian basis function,logistic sigmoidal funtion,'tanh' function.The task at hand was to model the given bivariate data using gaussian basis function.

3.1 Dataset

The given dataset contains 2000 samples from a bivariate distribution. The visualization of the training dataset is as shown in Figure 19.

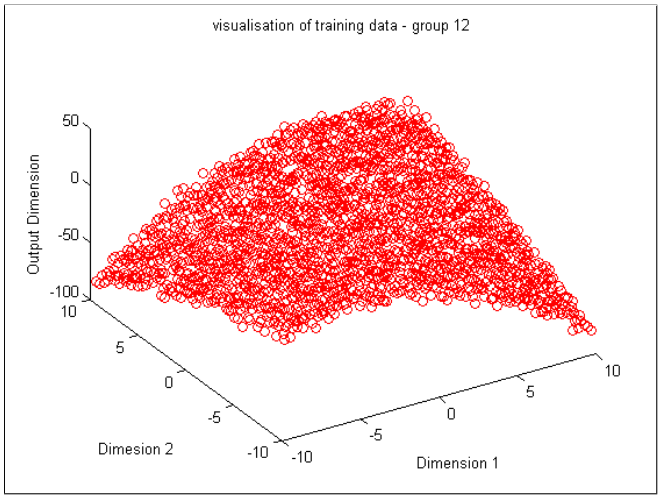

Figure 19: Bivariate dataset

3.2 Linear Gaussian Basis Functions

The linear gaussian basis function is given by Equation 5.

$$\phi_j(x) = exp\{\frac{(x-\mu_j)^2}{2S}\} \tag{5}$$

Where μ governs the value of the basis function in input space and S governs the spatial scale of the basis function.

3.3 Overview of the methodology used to freeze the model complexity and parameters

The following steps are followed to freeze the model complexity and parameters.

- For large value of N check the value of E_{rms} for various values of M on training data. Choose a value of M that performs well on the training data and validation data.(The value of N is kept large when λ is 0, because by thumb rule if samples are of order of 10p where p are the number of model parameters to be estimated then chances of over-fitting are less).
- For the chosen value of M, tune the performance of the model by varying λ over the validation data and choose the value of λ that gives minimum E_{rms}.
- Report the performance on test data.

3.4 Plot of target output and model output for training data for different model parameters

We are experimenting on the training data of N =2000 samples for different model complexities and different regularization parameter values. The main objective of this experiment is to analyse the performance of the model for these parameters.

Figure 20-23 shows plots for various values of M. The figure from 24-27 represents the plots when λ is varied.

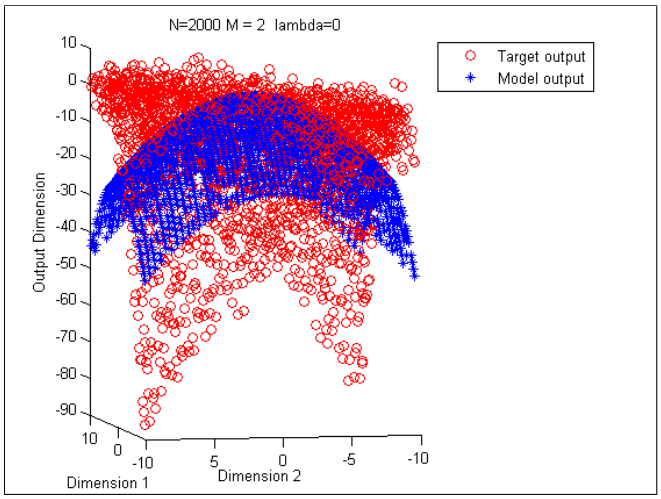

Figure 20: Model output when M=2 and $\lambda = 0$

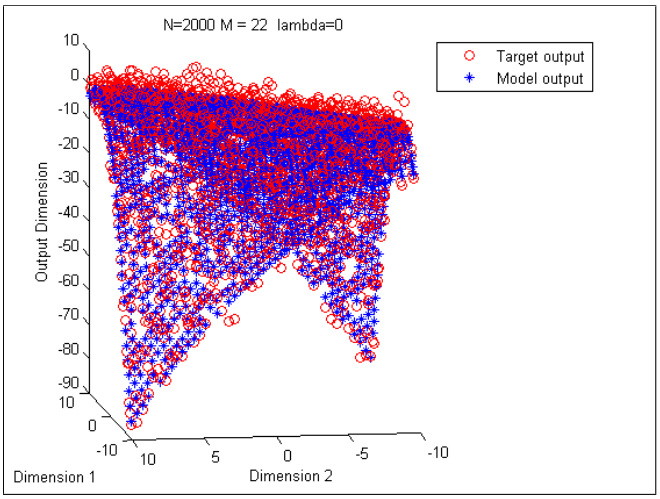

Figure 21: Model output when M=22 and$\lambda = 0$

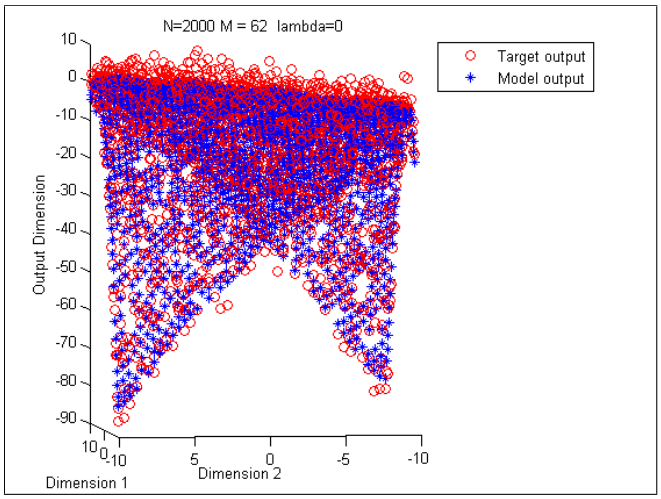

Figure 22: Model output when M=62 and $\lambda = 0$

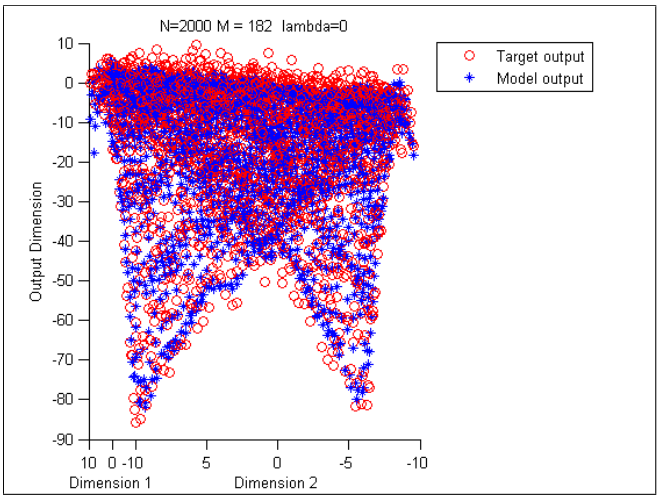

Figure 23: Model output when M=182 and $\lambda = 0$

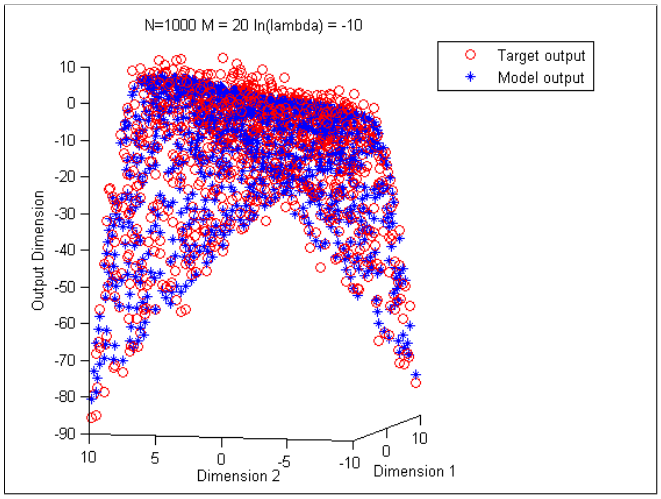

Figure 24: Model output with M=20 and λ = -10

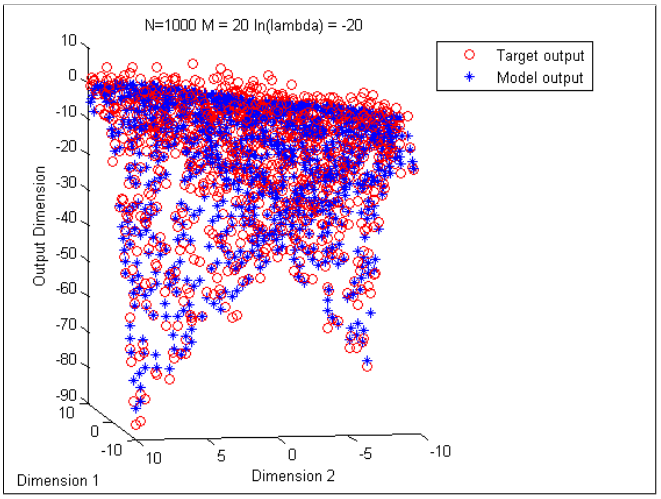

Figure 25: Model output with M=20 and λ = -20

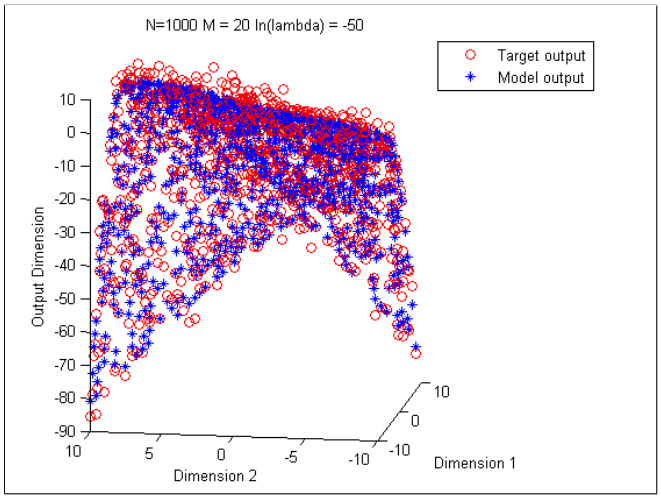

Figure 26: Model output with M=20 and λ = -50

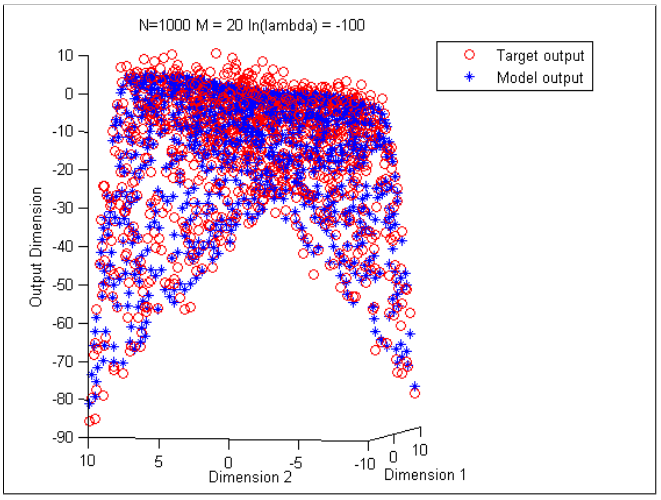

Figure 27: Model output with M=20 and λ = -100

Observations

- As M increases the model output follows closely the training data.
- As we increase the value of λ the over fitting decreases and it generalizes better.
- At M=20 and $ln\lambda$=-10, the best plot is obtained.

3.5 Plot for RMSE for different model parameters

The figure 28-31 shows the plot of RMSE Vs. lambda for various values of M.Figure 32-35 shows the plots of RMSE Vs M for different values of λ.

Observations

- For a given value of M,a very high value of λ gives large RMS error due to under fitting.As we reduce the value of λ the RMS error drops.The value of λ should be optimally chosen so that the model neither over fits(λ is small) nor under fits(λ is large).
- For a fixed λ and N as we increase the model complexity M the RMS error decreases till some value of M and increases there after.This is a generic trend for all the four plots of RMSE Vs M.

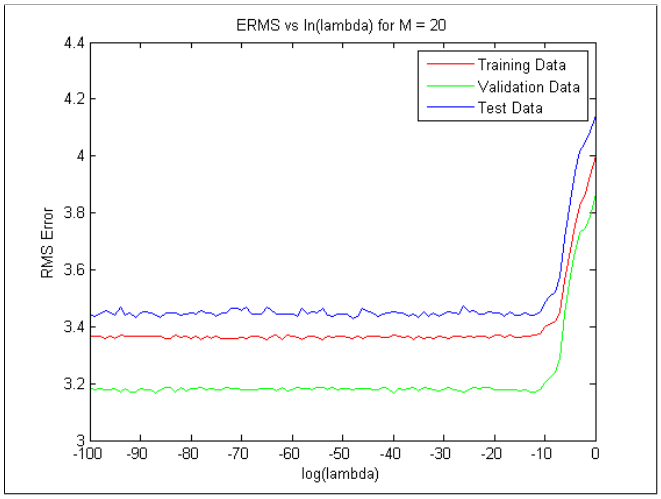

Figure 28: Plot of RMSE Vs. lambda M=20 N=2000

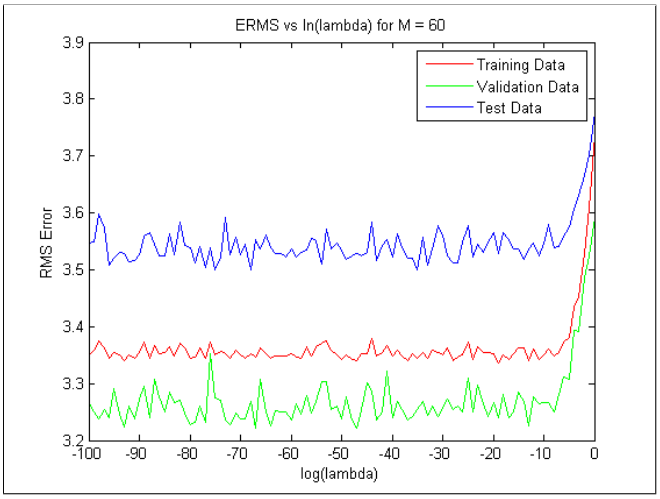

Figure 29: Plot of RMSE Vs. lambda M=60 N=2000

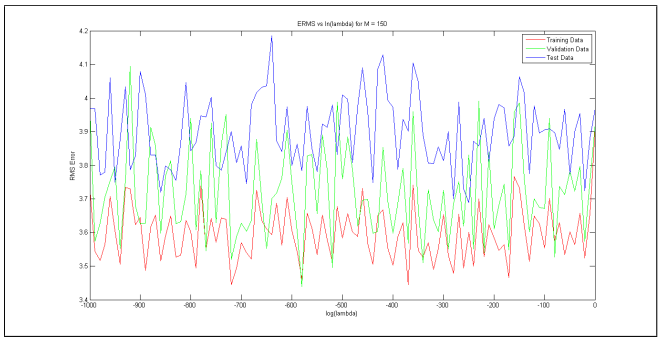

Figure 30: Plot of RMSE Vs. lambda M=150 N=2000

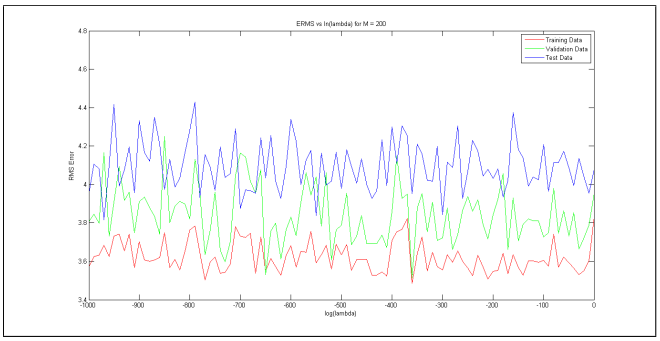

Figure 31: Plot of RMSE Vs. lambda M=200 N=2000

3.6 Scatter plots with target value on x-axis and model output on y-axis

Scatter plots of target value Vs model output have the general property that if the plot follows the line x=y closely then the model is good for the given data set.Please refer to the figure 36 for the plots of target value Vs Model output for various values of M and λ.The plots along horizontal axis are for same value of M and the plots along the vertical axis are for the same value of λ.

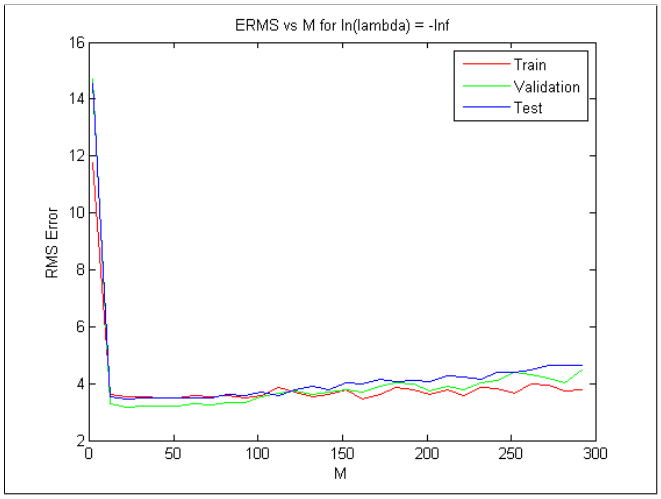

Figure 32: Plot of RMSE Vs M for logλ= -inf

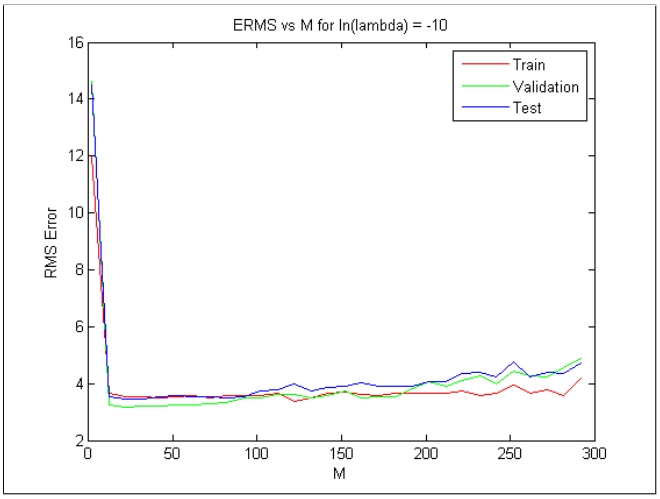

Figure 33: Plot of RMSE Vs M for logλ= -10

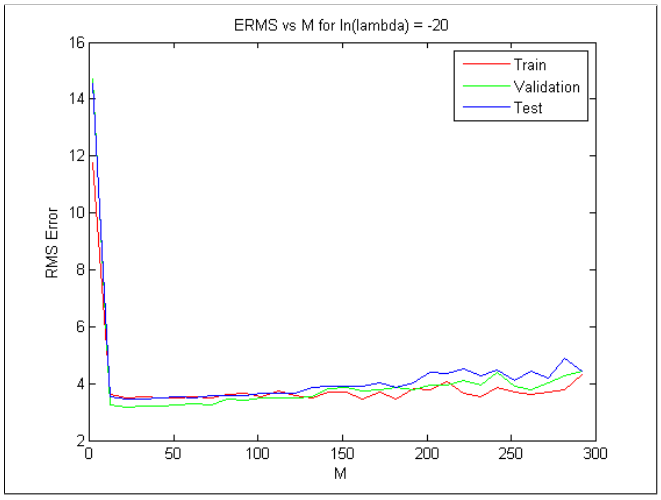

Figure 34: Plot of RMSE Vs M for logλ= -20

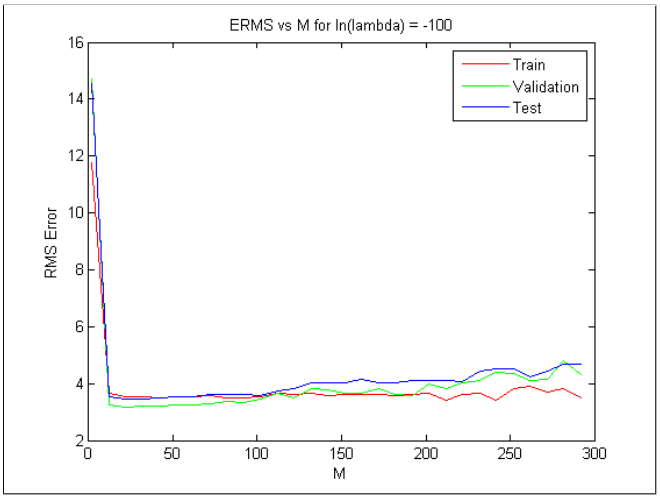

Figure 35: Plot of RMSE Vs M for logλ= -100

Observations

- We can observe when the value of λ is large then the model tries too hard to generalize and eventually ends up performing very badly on the test data. The scatter plot is highly spread out.That is the reason why the plots in the first column are more spread out than the subsequent columns. As the value of the parameter λ decreases, the performance on the test data improves.The scatter plots become more and more compact and close to the line x=y.
- The spread of the scatter plot also decreases when the model complexity M increases.

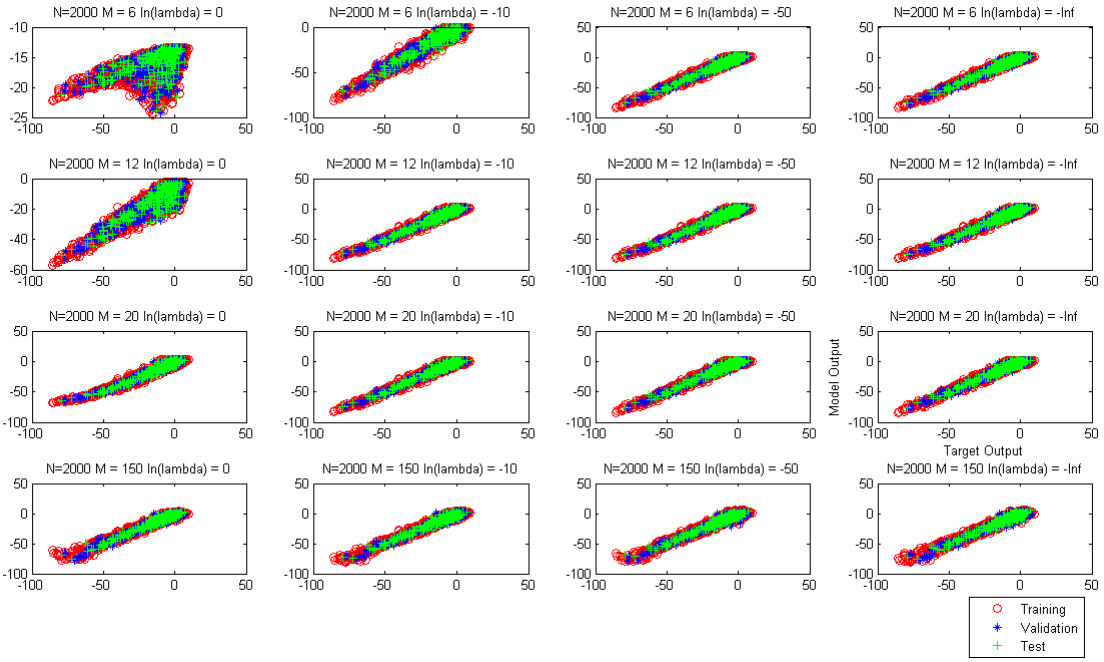

Figure 36: Plot of target value Vs model output for various values of M and λ

3.7 The relation between M and the parameter S

- M represents the number of basis functions that are being used to generate the model for the given data set.
- S in the basis function determines the spatial scale of that particular basis function.
- We refer to average cluster radius as s.

We observed while experimenting that the RMS error shows very inconsistent behavior when M is low say 6 and the spatial scale factor is s^2.This is because we are using a small number of basis functions and there spread is also small.This resulted in the design matrix having lots of 0 and 1.Therefore we chose M as 20 and spread as $3s^2$.That gave us a smooth curve of RMS error with various values of λ and the spikes in the plot were largely removed.Please refer to the figure 37-40 for the plots of RMS error with various values of M.

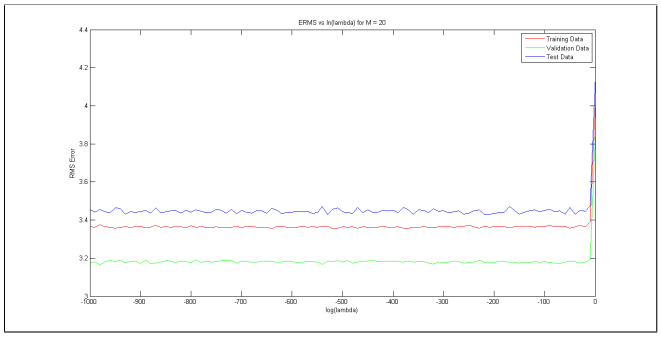

Figure 37: RMS error with S=$3s^2$ and M = 20

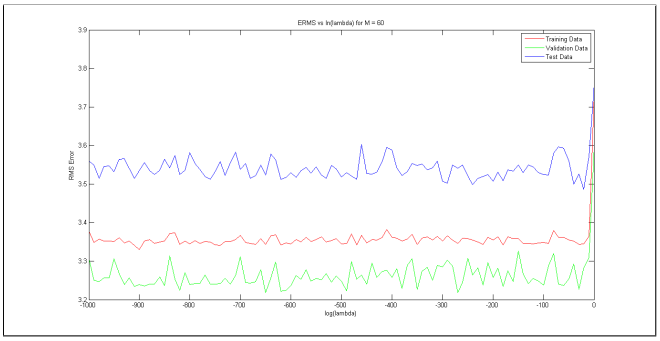

Figure 38: RMS error with S=$3s^2$ and M = 60

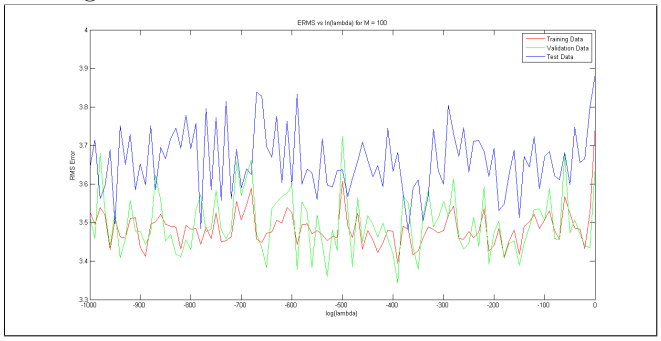

Figure 39: RMS error with S=$3s^2$ and M = 100

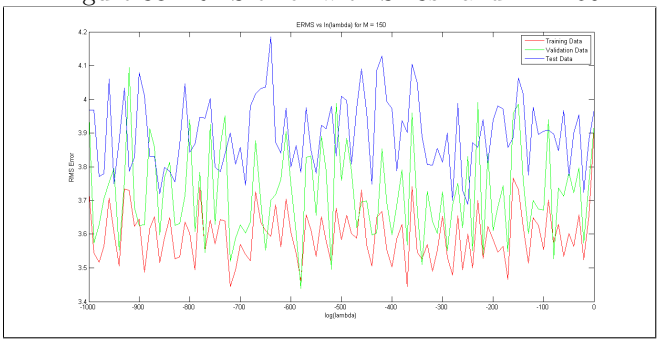

Figure 40: RMS error with S=$3s^2$ and M = 150

4 Conclusion

We conclude from the observations that M=6 gives the best polynomial curve fit for Dataset 1.For Dataset 2 M=20 and logλ = -10 gives the best linear model of regression when gaussian basis functions are used.