CSE 574 Machine Learning

Project 1 Report

Siddharth Srivastava

UB#: 50097583

Problem Statement:

Regression Using Microsoft Lector 4.0:

We are given the query-url pair datasets from Microsoft LETOR 4.0 to train a regression model to predict the document relevancy labels given query features. The dataset contains the relevance of some web pages labeled by humans for a set of queries. The relevance values are 0, 1, 2 with higher value indicate higher relevance. We need to implement the appropriate regression model, train, validate and test your model.

Dataset LECTOR 4.0:

Microsoft LETOR4.0 dataset which is for web search ranking, it contains queries and urls represented by IDs, and feature vectors extracted from query-url pairs along With relevance judgment labels.

Learning Model:

Input data contain 46 feathers in each input vector and have one associated target value. The target is to compute a function y(x,w) so that we can compute the relevance labels (target values) with the help of input feathers.

The Function we have to compute

$$y(\boldsymbol{x}, \boldsymbol{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\boldsymbol{x}) = \boldsymbol{\phi}(\boldsymbol{x})\boldsymbol{w}$$

Here :

X: D dimensional input vector from LECTOR 4.0

M : Model complexity.

$\Phi(\mathbf{x})$ - Gaussian radial function

Gaussian Funtion:

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

Here μ_j is the mean and govern the location of basis function in input sequence.
And **'s'** governs the spatial scale.
Error: All the error we are calculating as

$$E_{RMS}(\mathbf{w}) = \sqrt{\frac{2E_D(\mathbf{w})}{N}}$$

Regularised Least Square:

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

The coefficient λ governs the relative importance of regularization term with respect to the squared error.

Training Algorithm:

1. Segregate training set, Validation set and Testing set.
2. Build the Gaussian radial function by setting the M=2.
3. We will get μ_j by diving the dataset in M-1 sets and then taking column wise mean and 's' can be taken standard deviation of those sets.
4. Test the RMS error by changing the value of M and compute the M where minimum value of Error occurs.
5. Compute design matrix by setting M as M found in previous step.
6. We then compute :

   $$\mathbf{w}^* = (\boldsymbol{\Phi}^{\mathbf{T}}\boldsymbol{\Phi} + \lambda\mathbf{I})^{-1}\boldsymbol{\Phi}^{\mathbf{T}}\mathbf{t}$$

   by adding a λ = 0.1 for regularised least square.
7. Change the values of λ to compute the minimum error.
   From the training set we got the Model complexity , W and λ.

Validation Algorithm:

1. Using the W , μ_j and 's' from the training set, we compute the RMS error in the validation set also:

$$E_{RMS}(\mathbf{w}) = \sqrt{\frac{2E_D(\mathbf{w})}{N}}$$

2. Repeating steps 2- 7 in training set to minimize the error

Testing Algorithm:

1. Using this model try to predict the target valued in Test Set.

Experimental Evaluation:

CSE 574 Machine Learning

I have taken the training set as a Matrix of feathers of dimension**: matLearing[6048*46].**

Validation set as a Matrix of feathers of dimension**: matValid[1522*46].**

Testing set as a Matrix of feathers of dimension**: matTest[6706*46].**

1. Intaially started with M=2 and no regulariser.
2. Changed M and found the M with minimum error on training set asM=9.
3. For Over fitting of curves, started with the λ =0.1 and M=9 on the validation set, changed the λ as [0.2, 0.5,1, 1.5….10] and found error min at λ = 2.

Result (On testing Set):

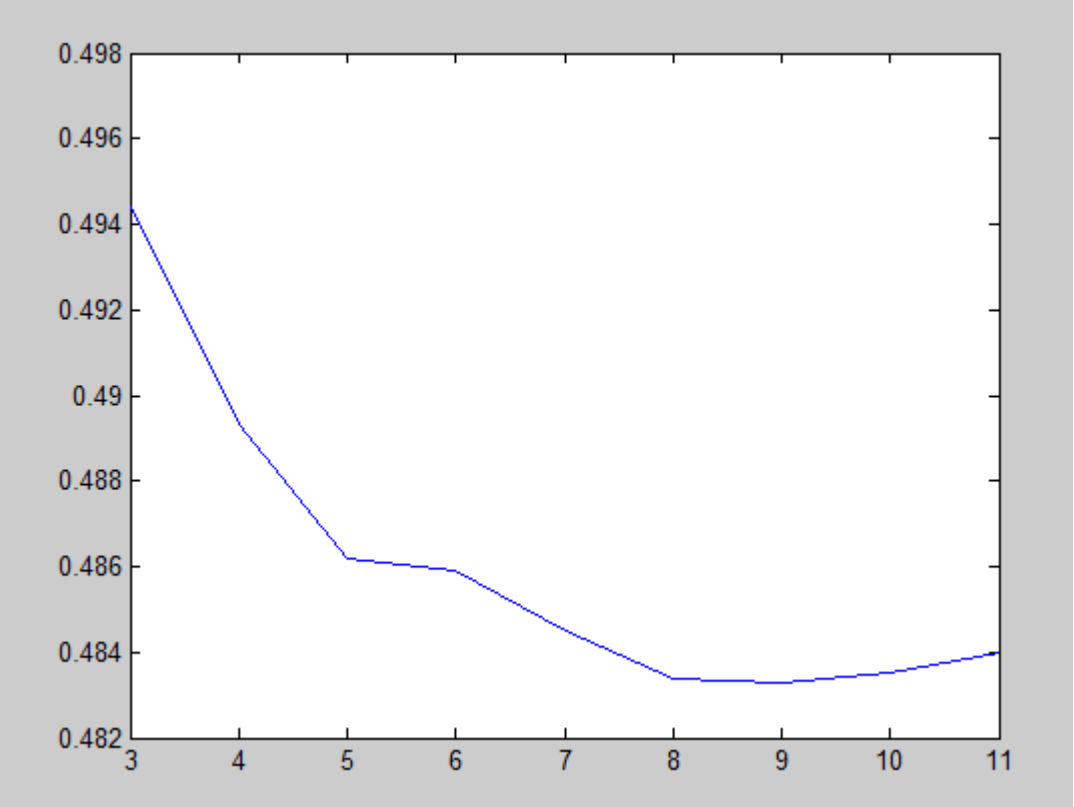
Optimum M= 9.

λ = 2.

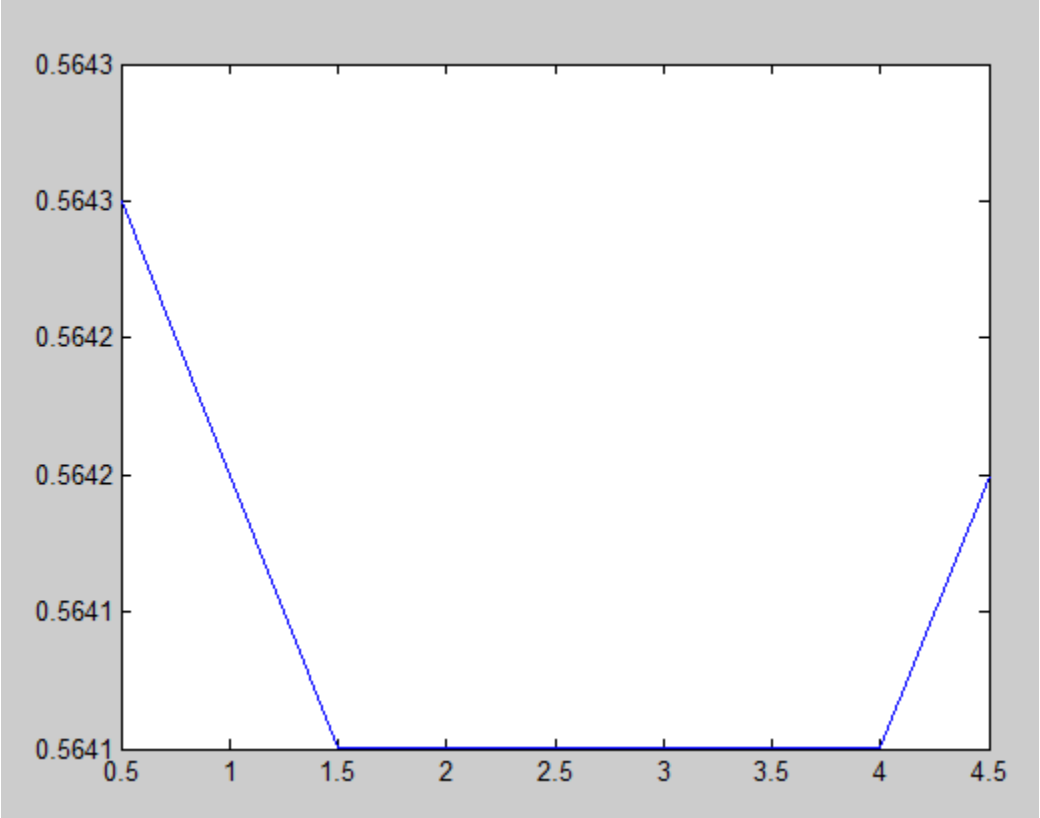
Error= 0.5241

Graphs:

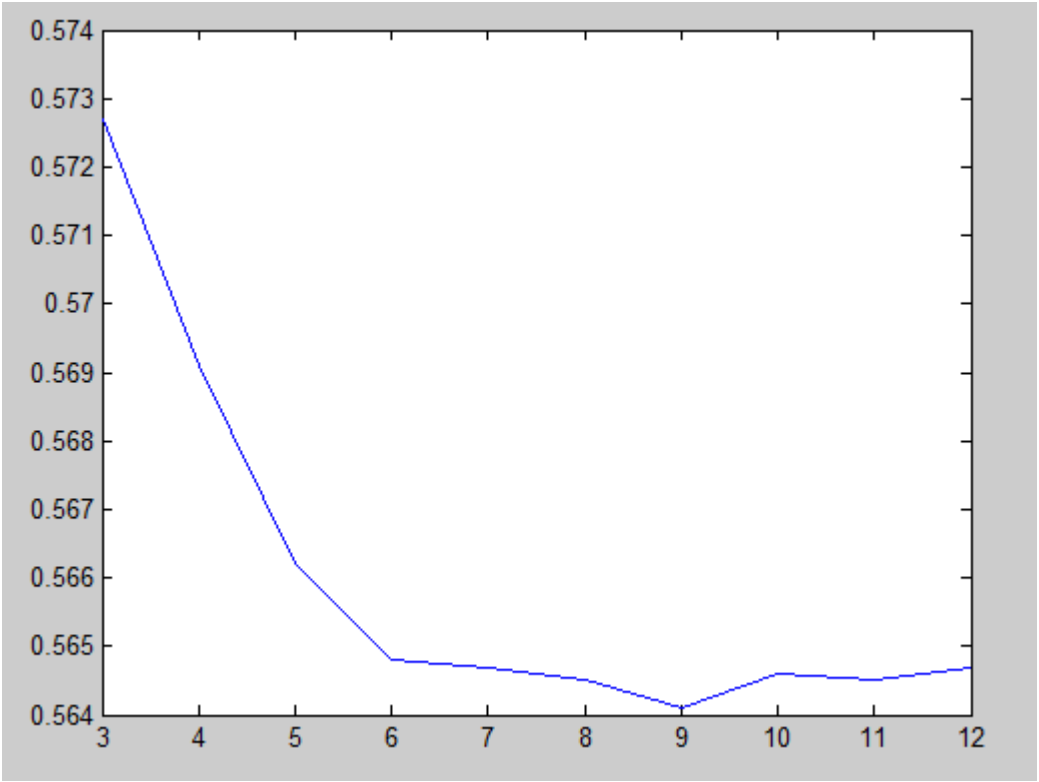
1. Error on training set with changing M : RMS Error vs M (model complexity)



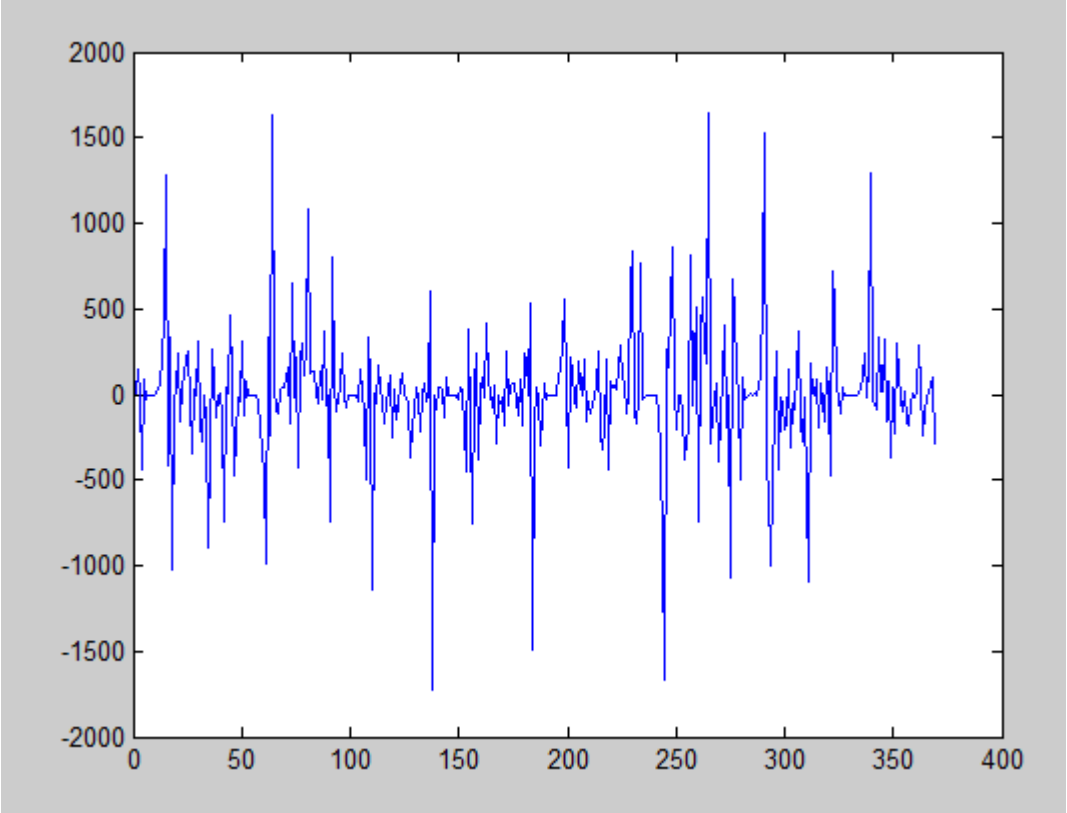
2. Error on Validation set with M=9 and lambda λ varying: RMS error vs λ



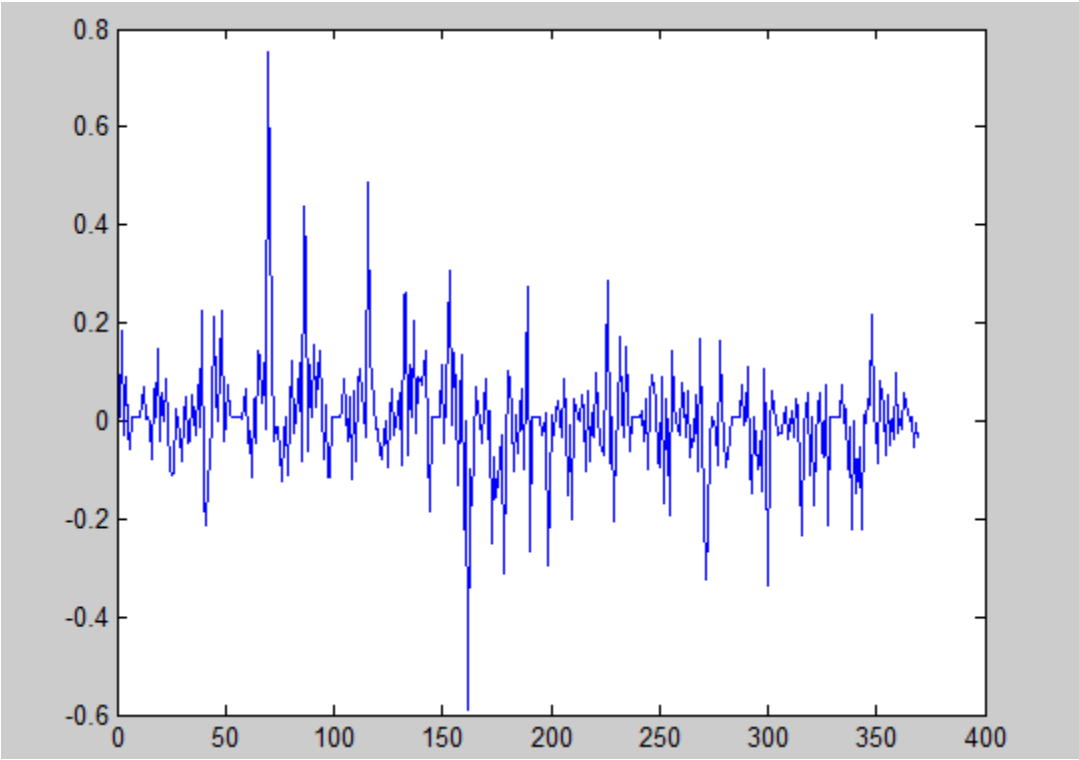
3. Error on Validation set with λ =2 and M varying : RMS Error vs M:



4. W with no Regulariser: λ =0:



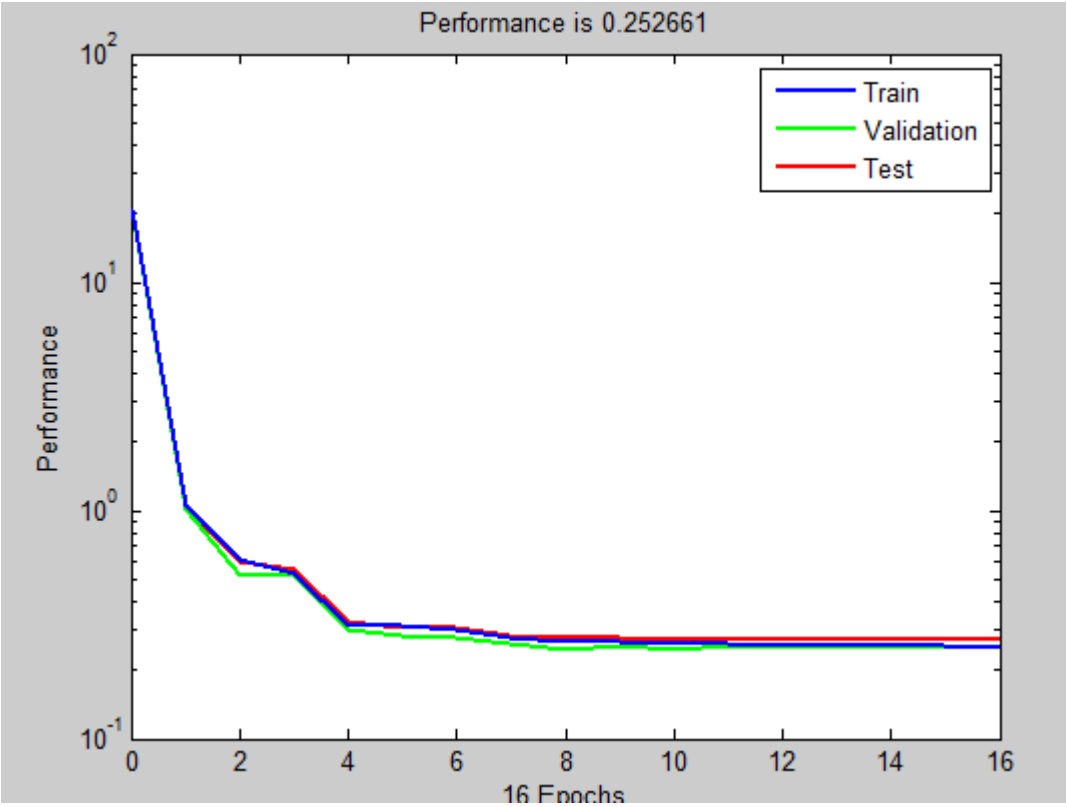
5: Error with λ =2 on validation set:



Comparison with Neural Network Model:

RMS error Neural network: **0.521**

RMS error Regression**: 0.5241**



Conclusion :

Regression Model performance was tested with M=9 and λ =2 with Neural Network Model and it error found in both model is quite comparable.