**Project – CSE574**

**Project Title – Linear Regression with Basis**

**Functions**

**Student Name: Harsha Hassan Somashekara**

**UBID: hhassans**

**Person Number: 50098099**

**Problem:**

The project asks to implement and evaluate several supervised machine learning algorithms to the task of linear regression. We were given the data which consists of the relevancy of the results in a search engine based on 46 features.69000+ results were given for that many queries. The project was to apply Linear Regression and train the machine to find the relevancy or find the maximum likelihood relevancy given the features as input.

**Procedure followed:**

The approach to this problem included the following steps.

**1. Dividing the Dataset**

Initially the dataset is divided into 3 parts:

Approximately 60% for training the model, 10%

for validation and 30% for testing.

**2. Choose the model variables**

These are few model variables, initialize them

to some value and write the code to fit the training model. First of all, select the type of basis function(I choose Gaussian), choose mean and variance, complexity of the model M and the regularization parameter lambda.

**3. Calculate Mean**

Mean value is required in order to find the design matrix in Gaussian Basis function. Mean is choosen based on the complexity of the Gaussian Basis function, if the complexity is M, the training dataset is divided into M-1 parts and calculated mean

for each feature in those M-1 parts.

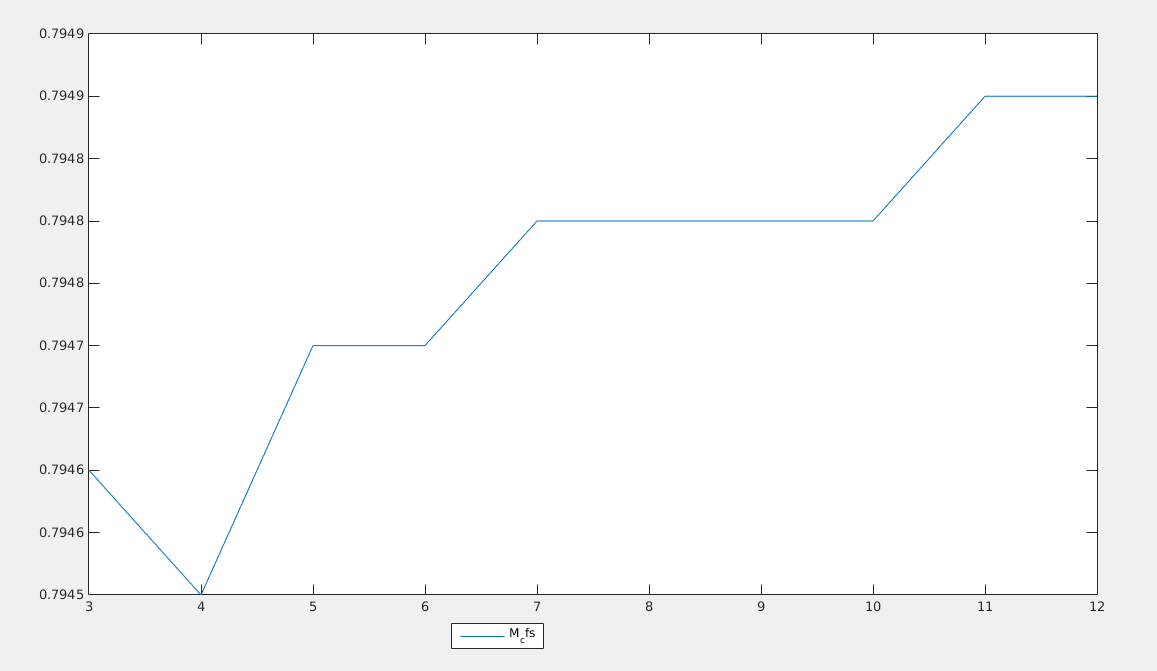
**4. Calculate Variance**

Variance value is calculated combining all the features or the values in all the columns.

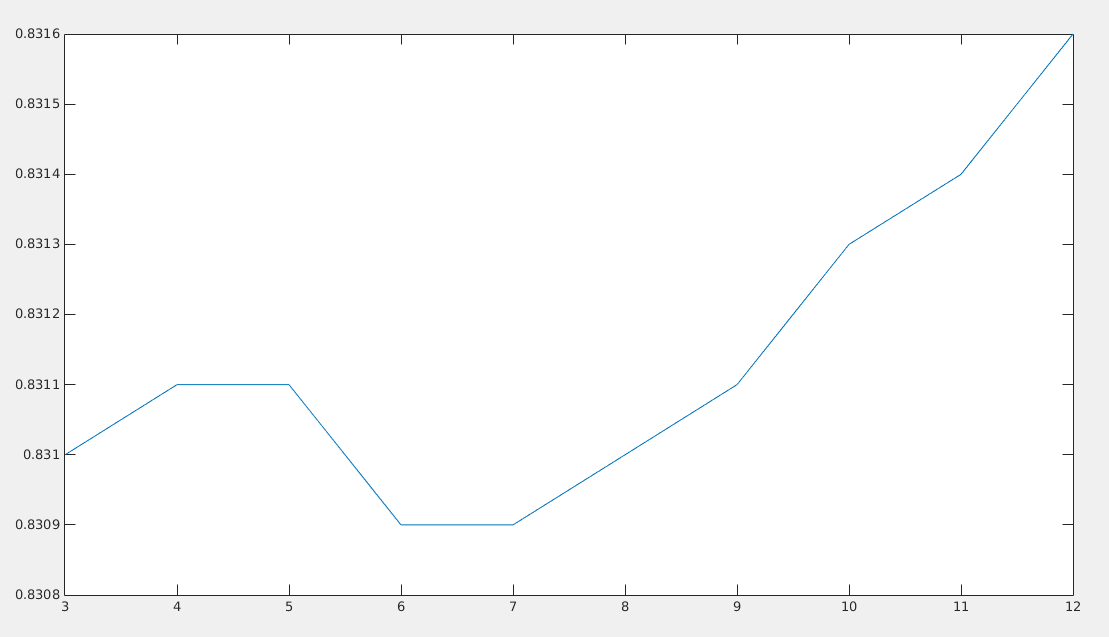
**5. Tune Model Complexity**

Iterate through certain range of model complexities M and then fine tune and select the complexity where you are getting minimum Root mean square error. We have plotted a graph of model Complexity M versus ERMS, We can get the point where we are getting the minimum Erms value and can fix that point as the Model complexity and then tweet with other parameters.

**mVersusErms\_CFS**



**mVersusErms\_GD**



**6. Calculating design matrix phiMatrix and weights**

The design matrix is a N \* M matrix where N is the length of the sample training data and M is the model complexity. The first column values are set to 1.

After that the weights are calculated by using the formula:

weightMatrix = pinv(transpose(phiMatrix)\*phiMatrix+lambda\*eye(M))\*transpose(phiMatrix)\*relevanceMatrixTraining;

where, phiMatrix is design matrix

lambda is the regularization parameter to avoid

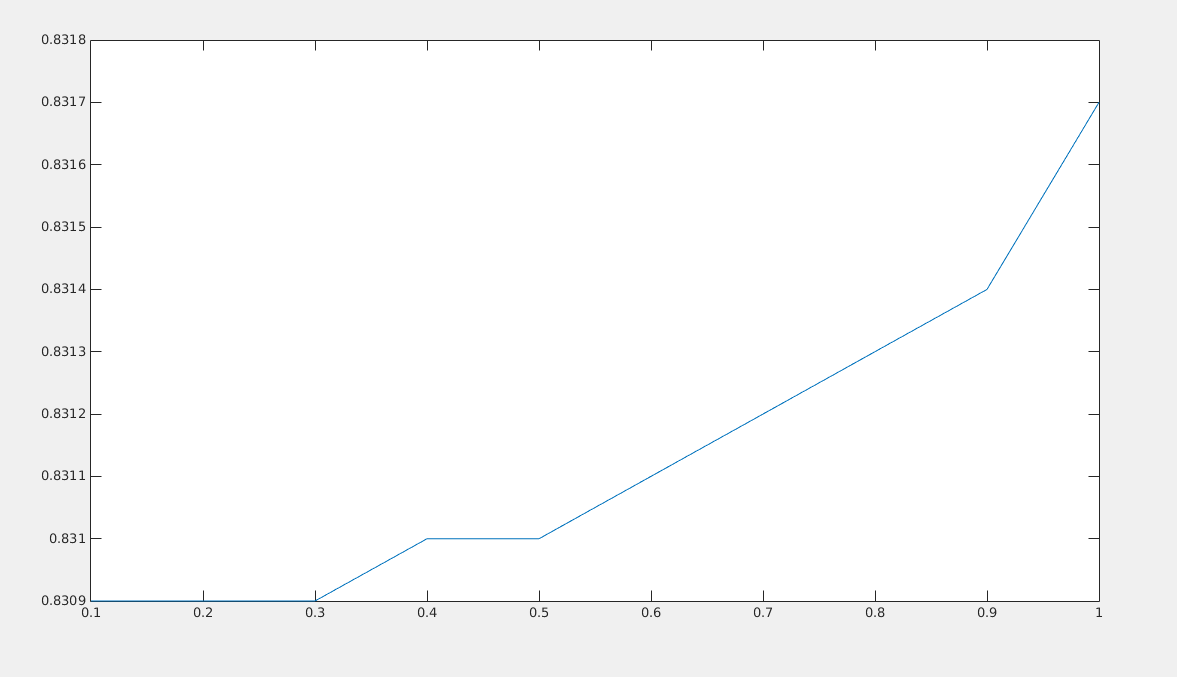
over fitting

relevanceMarixTraining is the target value or

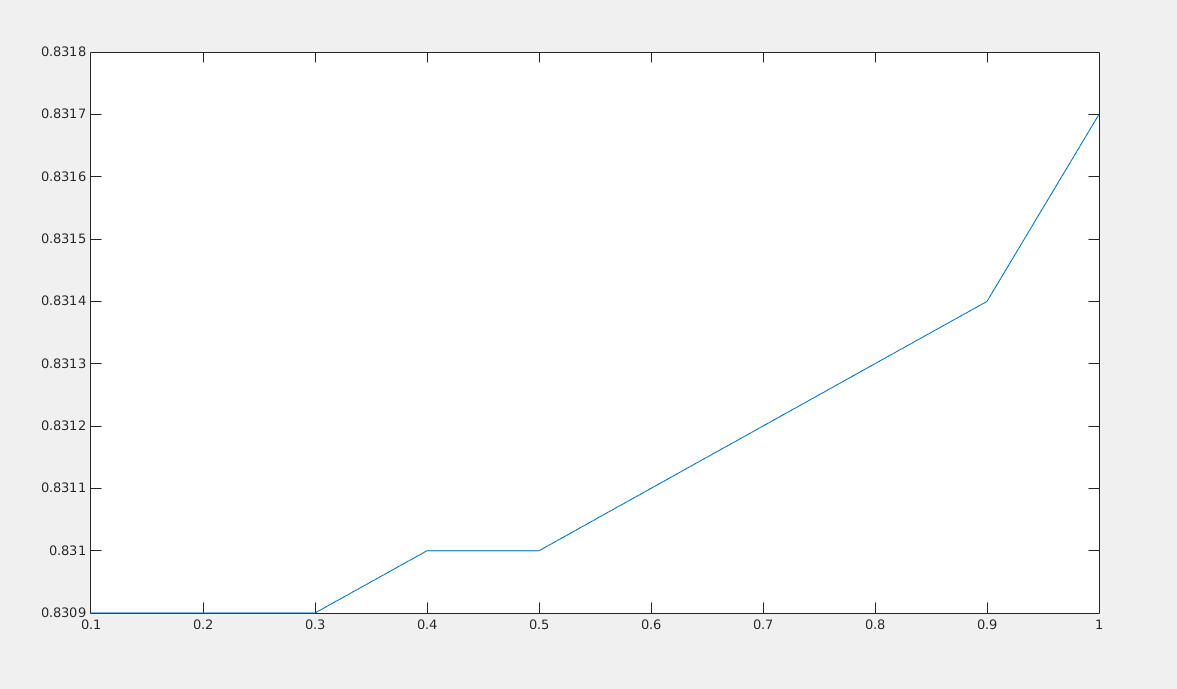
relevance values for the search query in our case.

**7. Tune the Regularization parameter lambda**

**LambdaVersusErms\_CFS**



**lambdaVersusErms\_GD**



Now again find the Erms value for a range of

different values of lambda by keeping the weight matrix constant.

Select the lambda value whichever gives the minimum Erms value.

Model is tuned according to two parameters M, the model complexity and lambda, the regularization parameter

I have chosen the final model complexity and lambda according by checking the effective Erms value. Point at which the Erms value is minimum is chosen as the final model complexity and run with the test data to check the correctness of the model and it is verfied with some degree of accuracy.