Learning To Rank using Linear Regression

Kiran Prabhakar

October 11, 2018

Abstract

The purpose of this project is to perform the linear regression operation on LeToR data set by using both closed form solution and stochastic gradient descent (SGD) approaches.

1 Introduction

The LeToR data set consists of input values which are real valued vectors and has three scalar target values 0,1,2. As the training target values are discrete, we use linear regression to obtain a model for the given data set.

2 Definitions

2.1 Linear Regression

Linear Regression is a predictive modeling technique in which a model is derived from existing input and output values. Using the model, the outcomes of feature inputs can be predicted. The number of features on which the input data is relied will play a key role to in constructing the model.

2.2 Gaussian radial basis function

The value of radial basis function depends on the distance from the center (C) or origin, it is represented by ϕ. A The radial basis function satisfies the below property.

$$\phi(x) = ||x - c||$$

$||x||$ is called as the L2 Norm of x, which is usually the Euclidean distance. The basis function used in the project is Gaussian radial basis function, which is represented by the equation.

$$\phi_j(x) = exp\Bigg(-\frac{1}{2}(x-\mu_j)^T\Sigma_j^{-1}(x-\mu_j)\Bigg)$$

Here μ_j is the center of the basis function and Σ_j represents the variance or spread of the basis function

2.3 Regression with Basis function

In a linear regression, the problem of not knowing the form of exact relationship between the input vectors and the target, can be resolved by using linear basis functions. When a input vector is passed to a basis function, it outputs a scalar quantity corresponding to the value of the vector.

The standard linear regression equation with basis function is as below

$$Y = w_0 + w_1\phi_1(x_1) + w_2\phi_2(x_2) + ... + w_p\phi_p(x_p)$$

where ϕ is the basis function and the $w_0, w_1..., w_p$ are weights of p different features.

Two methodologies used for linear regression in the project are:

1. Closed-form solution.
2. Stochastic Gradient Descent (SGD).

2.4 Probability density function (PDF)

Probability distribution provides the probabilities of occurrences of different possible outcomes in an experiment. PDF provides the relative likely hood for occurrence of a value in the distribution. It is expressed as below

$$\int_a^b f(x)dx = Pr[a < X < b]$$

where a and b provides the bounds of the data items in the distribution.

2.5 Regularization

Regularization is a technique which helps to avoid the problem of over-fitting and obtains better generalization. It is also refereed as unlearning technique. A regularization constant (λ) is used to perform regularization.

If the training data contains less inputs and less features, then the model will exactly fit to the available data set and causes the problem of over-fitting. Regularizers like weight decay regularizer can be used to circumvent this issue.

Regularization also reduces the variance in the model without increasing the bias. For a design matrix ϕ and target (t), the regularized weights are obtained using below equation.

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

2.6 Root Mean Square Error (ERMS)

ERMS is the square root over sum of the squared differences between the observed(t_n) and predicted(t) outputs. While performing regression, the error calculated using ERMS is used to update the weights. The error or cost function for a data set is calculated using the equation

$$E_D(w) = \frac{1}{2}\sum_{n=1}^{N}\Big\{t_n - w^T\phi(x_n)\Big\}^2$$

After adding regularizer on the $E_D(w)$, we get the regularized error as (W^*), which we use to calculate the ERMS using the the below equation.

$$E_{RMS} = \sqrt{2E(w^*)/N_v}$$

N_v is the number of data points in the data set

2.7 Clustering

Clustering is a technique of grouping the data points in such a way that the points which are similar will fall under same group. In the current project we use the most popular and efficient **K-Means** clustering to cluster the 'N' data points in to 'M' clusters. The process of clustering the data is done as below

1. Initially divide the entire data into 'M' sets and the centers of each group are the vectors of same length as each data point.
2. Then find the distance between each point and each group center and classify the a point into the group if the distance is closest.
3. After classifying a set of points in to group, the new group center is obtained by calculating the mean of all the vectors in the group
4. The above steps are repeated until the group centers do not change much, which indicates that the requied means and clusters are obtained.

In the current project all the above operations are performed using the **KMeans** functionality provided in the **scikit-learn** library.

2.8 Variance

Variance tells about the spread of the data set. To find the variance, we find the sum of all squared differences between the mean and each data item and then divide the total sum by the number of data points.

$$\sigma = \frac{\sum_{n=1}^{N}(x_n - \mu)^2}{N}$$

where σ is the variance of the set, μ is the mean and N is the total number of data elements in the set

2.9 Co-Variance

Unlike variance, which calculates the spread of entire data set, co-variance will measure the correlation between the features of two random variables with in the data set. If there is no correlation between the features, the co-variance between the data points is considered to be zero.

In the current project as each feature is independent of other the co-variance among the values in the same column(feature) is calculated, which are the values present in the diagonal of the co-variance matrix and other values are considered as zero. The co-variance matrix is obtained as below, where σ refers to the variance corresponding to particular feature.

$$\Sigma = \begin{pmatrix} \sigma_{\mathbf{1}}^2 & & & \\ & \sigma_2^2 & & \\ & & \ddots & \\ & & & \sigma_D^2 \end{pmatrix}$$

3 Data Pre-processing

To perform the linear regression on the LETOR data set, we initially process the text file which has the raw data into two .csv files

1. Querylevelnorm_X which contains the 69623 data items, where each data point is the combination of 46 features
2. Querylevelnorm_t has the relevance label for each data point.

we ignore other information related to docid, qid and prob as they are irrelevant for our regression operation.

All the data points in the columns 5, 6, 7, 8, 9 of the input file has 0 values for the corresponding features. So, we ignore these columns while reading the data from the .csv file. This will reduce some unnecessary computing while performing multiplication, inversion and co-variance calculation.

We split the given data in to the below percentages for training, validation and testing

- Training - 80 percent
- Validation - 10 percent
- Testing - 10 percent

4 Linear Regression Using Closed form solution

The objective of the closed form solution in linear regression is obtain and leverage a closed form equation for calculating the weights directly.

Unlike batch regression or SGD, using the closed form solution, the entire data set is taken at once and the regression is performed using matrix operation on the data set.

The key operations performed in regression is to obtain the weights for the model by reducing the cost function. The cost function is obtained by calculating the difference between the target vector(y) and the calculated output for the inputs ($\mathbf{X}$). The weights are represented using ($\mathbf{W}$) matrix.

Considering the inputs, wights and the output as matrices, the cost function can be represented in matrix terms as below.

$$E(\mathbf{W}) = \frac{1}{2}(\mathbf{XW} - y)^T(\mathbf{XW} - y)$$

To minimize the cost function we take the derivative of the function with respect to $\mathbf{W}$ and equate it to zero

$$\nabla_w E(\mathbf{W}) = \frac{1}{2}\nabla_w(\mathbf{XW} - y)^T(\mathbf{XW} - y)$$

The derivative of the above equation is obtained as

$$\nabla_w E(\mathbf{W}) = \mathbf{X}^T\mathbf{XW} - \mathbf{X}^T y$$

By equating the above equation to zero we get

$$\mathbf{X}^T\mathbf{XW} - \mathbf{X}^T y = 0$$

$$\mathbf{X}^T\mathbf{XW} = \mathbf{X}^T y$$

$$\mathbf{W} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T y$$

The above obtained equation is called closed form solution for the weight matrix $\mathbf{W}$

In the current project as the inputs are vectors, Gaussian radial basis function is used to convert input vectors to scalar equivalents and then the design matrix is constructed as below.

$$\mathbf{\Phi} = \begin{bmatrix} \phi_0(\mathbf{x}_1) & \phi_1(\mathbf{x}_1) & \phi_2(\mathbf{x}_1) & \cdots & \phi_{M-1}(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2) & \phi_1(\mathbf{x}_2) & \phi_2(\mathbf{x}_2) & \cdots & \phi_{M-1}(\mathbf{x}_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_0(\mathbf{x}_N) & \phi_1(\mathbf{x}_N) & \phi_2(\mathbf{x}_N) & \cdots & \phi_{M-1}(\mathbf{x}_N) \end{bmatrix}$$

The dimensions of the Φ matrix depends on the input data (training or validation or testing) and the number of clusters(M) the input data is divided into. In the current project the training data size is 55698 and number of clusters (M = 60). So the dimensions of training Φ matrix is 55698 X M.

The Φ matrix is also calculated for validation and testing data which is used for predicting the validation and testing accuracy and E_{RMS}

The above derived closed form solution is altered using the Φ as below for calculating the weights

$$\mathbf{W} = (\Phi_T \Phi)^{-1} \Phi_T y$$

The above obtained equation is called the he Moore-Penrose pseudo-inverse of the matrix Φ

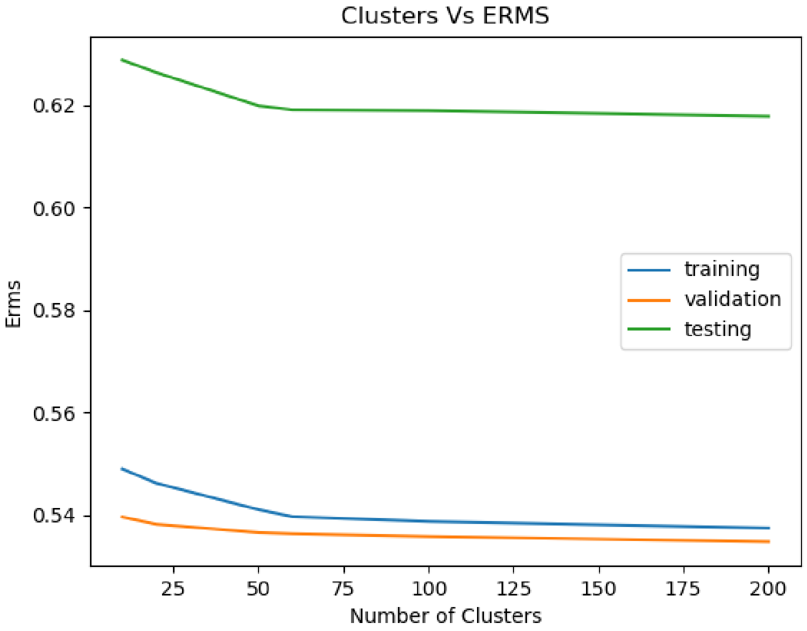
After introducing the regularization constant to avoid over fitting and the new regularized weight matrix is calculated using the below equation.

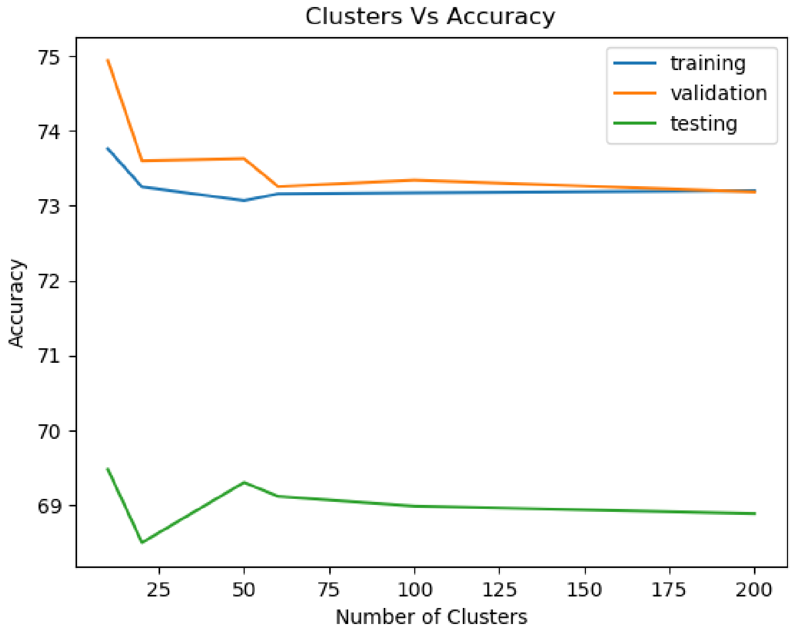
$$\mathbf{W}^* = (\lambda I + \phi^T \phi)^{-1} \phi^T t$$

By using the $\mathbf{W}^*$, the error value $(E(\mathbf{W}^*))$ is calculated, which is used for calculating E_{RMS} and accuracy. Accuracy is defined as the fraction of the data points that exactly matches with target.

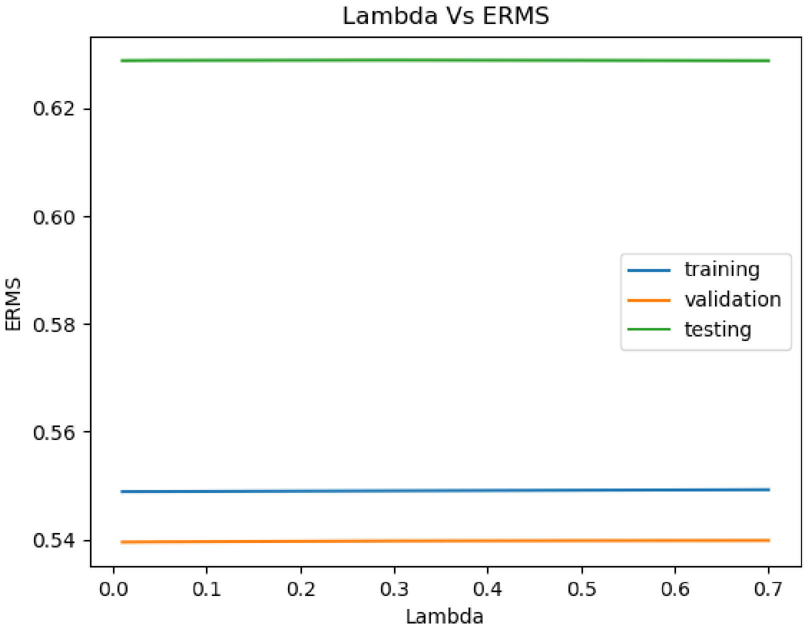
After the closed form solution execution completes, the program will out put the accuracy and E_{RMS} for training, validation and testing data.

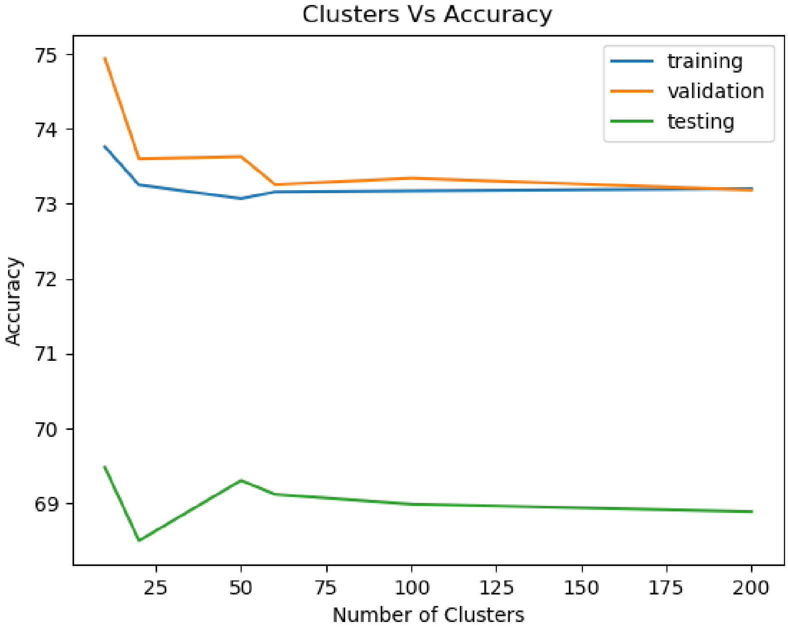
The below graph shows the E_{RMS} and accuracy obtained when the number of clusters (M) is from the array [10, 20, 50, 60, 100, 200] and regularization constant is 0.03





From the above graph it is evident that the E_{RMS} value starts decreasing initially and almost remains constant for values of 'M' greater than 60. So the ideal value for M can be considered as 60
Similarly, the below graph shows the E_{RMS} and accuracy obtained when the regularization constant is from the array [0.01, 0.03, 0.05, 0.3, 0.5, 0.7]





5 Stochastic Gradient Descent (SGD)

Stochastic Gradient Descent is a methodology for performing the linear regression where the weight update happens every time after each input is processed. The word 'Stochastic' means the inputs are selected randomly instead of groups or in the order they appear in the training set.

5.1 Gradient Descent

It is an optimization technique where we minimize the function by moving down in the direction of the decreasing slope of the function. This iterative process is carried out until we reach the optimum value. In SGD, we apply this technique to minimize the cost function and update the weights accordingly. The equation for updating weights is as below.

$$w^{\tau} = w^{\tau+1} + \Delta w^{\tau}$$

Here Δw^{τ} is called the weight update, which is done after processing each data item. The equation of for Δw^{τ} is as below

$$\Delta w^{\tau} = -\eta^{\tau} \nabla E$$

Where ∇E is the derivative of error function and η is the learning rate.

5.2 Learning Rate

In the technique of gradient descent, we use learning rate to determine how fast or how slow the minimizing the loss function will take place. If the learning rate is too less, it takes lot of time to converge to optimum. If it is too high, then the model might skip the global optimum. In the current project we represent it with λ and we take its value as 0.01.

5.3 Regression with SGD

The main purpose of regression using SGD is to update the weights continuously for every iteration of data set by minimizing the loss function.

The loss function is calculated using the below equation provided in the section (2.6 - ERMS) for (E_D). While calculating the error, the regularization is performed by considering the weight decay regularizer (E_W) and we get the regularized error function as below

$$E = E_D + \lambda E_W$$

By applying the derivative on the above equation we get the below equations

$$\nabla E = \nabla E_D + \lambda \nabla E_W$$

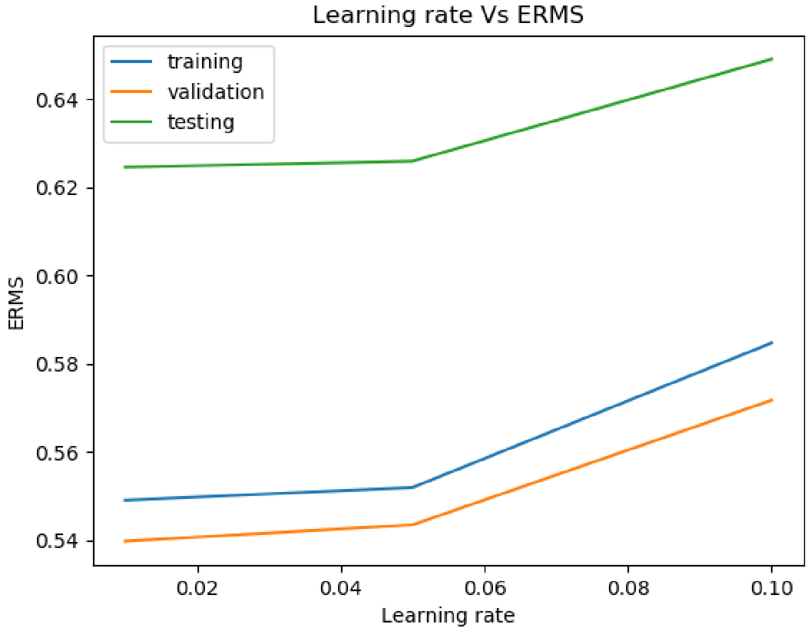
here

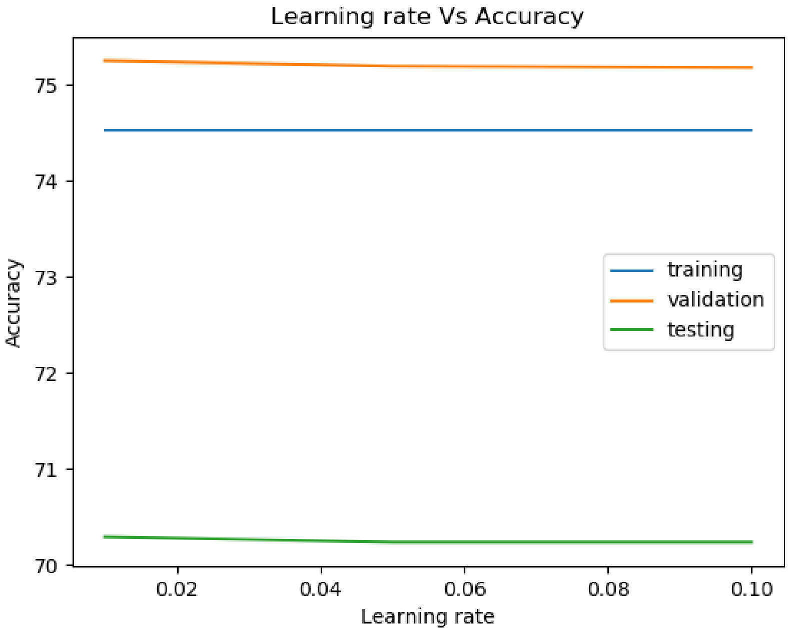
$$\nabla E_D = -\big(t_n - \mathbf{w}^{(\tau)T}\phi(x_n)\big)\phi(x_n)$$

$$\nabla E_W = \mathbf{w}^{(\tau)}$$

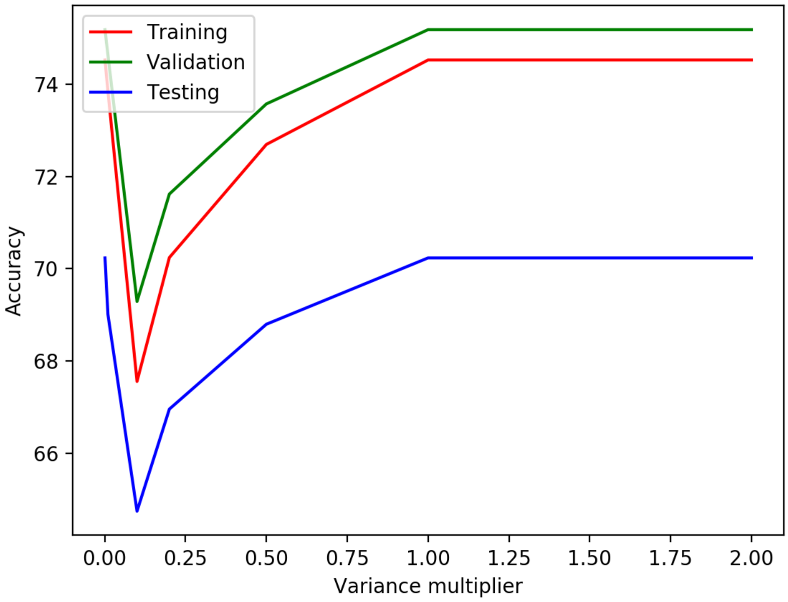
In the current project the above operations are performed for 400 data points and the accuracy and E_{RMS} is calculated for testing, training and validation data sets.

The below graphs shows the E_{RMS} and accuracy obtained when the number of clusters is 60 and the values for learning rate is from the array [0.01, 0.05, 0.1]





Another hyper parameter that can be considered is variance multiplier, which is the value multiplied to each row in the Big Sigma matrix. The below is the graph shows the accuracy when the variance multiplier is taken from the array [0.001, 0.01, 0.1, 0.2, 0.5, 1, 2].



6 Conclusion

The two approaches taken for linear regression in this project are closed form solution and Stochastic Gradient Descent (SGD). Below are the observations that can be noticed in the two approaches.

- Using Closed form we take the entire training data for creating the model, where as in SGD, we take few data items from the training set and perform continuous update of weights for generating the model
- As closed form solution involves lot of matrix operations, which are costlier, it may perform relatively slower when compared with SGD

Thus by analyzing and prepossessing the data set any of the one of the two approaches can be used for performing linear regression to generate a model for the training data set.

References

[1] https://towardsdatascience.com

[2] https://medium.com/data-science-group-iitr

[3] https://chemicalstatistician.wordpress.com