PROJECT REPORT

Project 2

CSE 574:

Introduction to Machine Learning

Group members:

1. pbisht2 - 50247429

2. rakeshsi - 50249135

3. karanhor - 50249274

# 1. Introduction

We have to design a linear regression model to solve the Learning to Rank (LeToR) problem. We used two ways to perform linear regression namely, closed form solution and stochastic gradient descent.

# 2. Dataset

We have two datasets:

1. LeToR data: 46 features, 69623 rows
2. Synthetic data: 10 features, 20000 rows

We split the data sets into **training 80%, validation 10%** and **testing 10%** proportions.

# 3. Tuning of Parameters

- *M* – number of basis functions
  Here, our basis functions are our Gaussian radial basis functions.
  We find the centers and spreads of these functions using K-means clustering. We find the optimal number of clusters by iterating through values for 'k' = 2 to 25, and calculating the validation error. The 'k' with the minimum validation error becomes our model parameter.

- $\mu_j$ – Centers of basis function. For our data, they are center of the clusters.

- Sigma j – Spreads of the basis function. The spread of basis function is calculated using all the data points inside the particular cluster.

- λ – Regularization constant. We have taken different values of λ and calculated our error.

- η – Learning rate. We took two values for learning rate while calculating our SGD solution.

# 4. Implementation

We split the data sets into **training 80%, validation 10%** and **testing 10%** proportions. We have calculated centers and spreads of the basis functions using K means algorithm.

- For different values of 'k' number of clusters, we have calculated our validation error. The 'k' with minimum validation error is taken as our optimal clusters.
- Using the design matrix, output data and lamda λ (regularization constant) we calculated weights using **closed form solution** and **stochastic gradient descent** models.
- We used regularization to avoid overfitting the model. Regularization is nothing but adding a penalty term to the objective function and control the model complexity using that penalty term. Regularization helps to choose a model with less complexity, so that model is better at predicting real world unseen data.
- We then apply our model on the test data to get the value for Erms.

## 4.a. Implementation of Early Stop

- We have implemented Early Stop in our SGD_sol function.
    1. We have taken the value of patience = 10.
    2. During training, we calculate the validation error after every $5^{th}$ epoch using the intermediate weights.
    3. If the value of validation error increases compared to the previous value, we decrement patience by 1, else we reset p=10 and continue.
    4. If the value of validation error increases consecutively after 10 checks, we stop training.
    5. We then use the weights the network had in that previous step as the result of the training run.
- We took two values for learning rate while calculating our SGD solution. For 0.1, our SGD_sol stopped after 75 epochs. For 0.01, our SGD_sol stopped after 475 epochs.

# 5. Results

## A. Lambda = 0.01

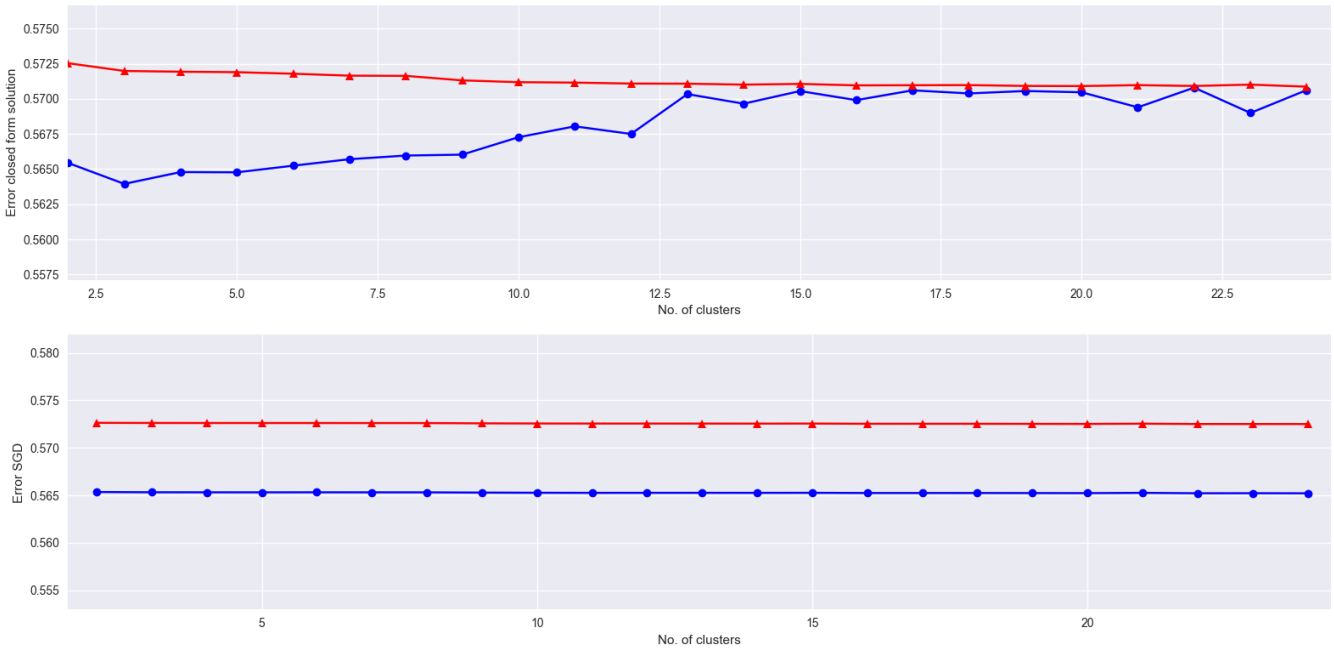


Figure: LETOR training error(red) vs. validation error(blue) for closed form(upper) SGD (lower)

We can see that the training error decreases with increase in number of clusters (M) and validation error increases. Hence, our model begins to overfit as the number of clusters increase.

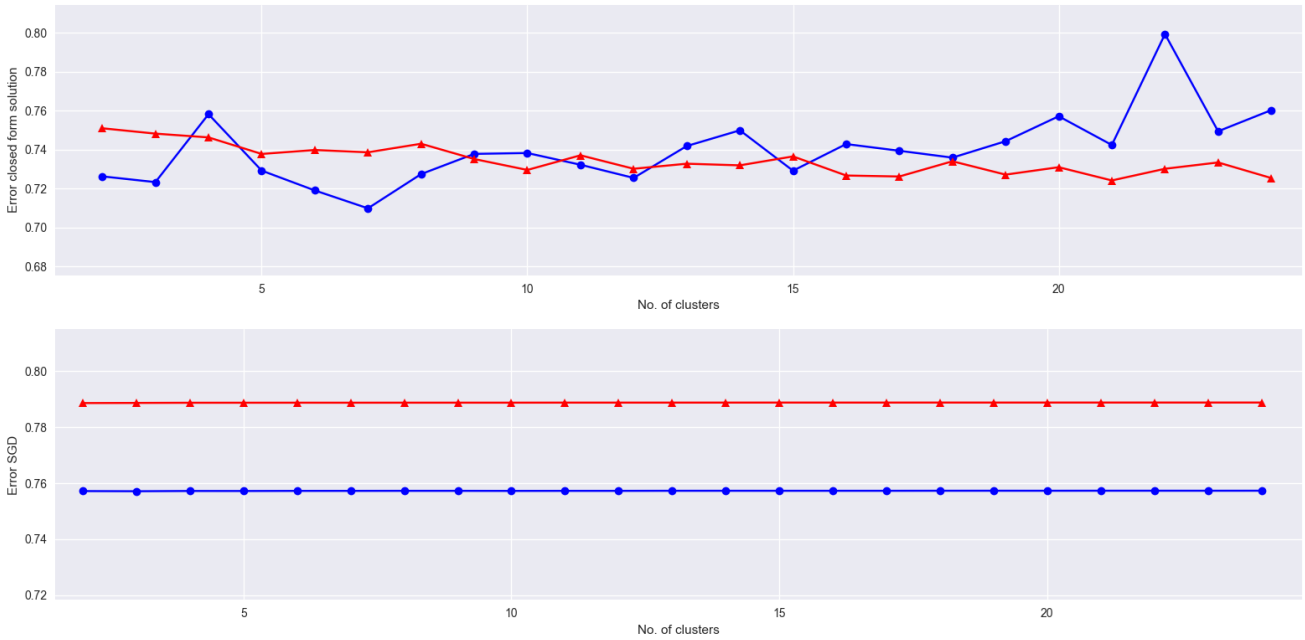


Figure: SYNTHETIC data training error(red) vs. validation error(blue) for closed form(upper) SGD (lower)

For closed form there is large variation in values with increase in number of clusters.

SGD the variations between the errors remain constant.

**Complete Output with Lambda = 0.01**

```
--------------------------
--------------------------
Optimal Number of Clusters for LETOR Data using Closed form solution =
3
Erms on LETOR Test Data Set using Closed Form Solution =
0.5652977605273489
Optimal Number of Clusters for LETOR Data using SGD =  24
Erms on LETOR Test Data Set using SGD =  0.565598591661354
Optimal Number of Clusters for Synthetic Data using Closed form Solution
=  7
Erms on Synthetic Test Data Set using Closed Form Solution =
0.7428178640453662
Optimal Number of Clusters for Synthetic Data using SGD =  3
Erms on Synthetic Test Data Set using SGD =  0.7948324867708371
```

## B. Lambda = 0.1

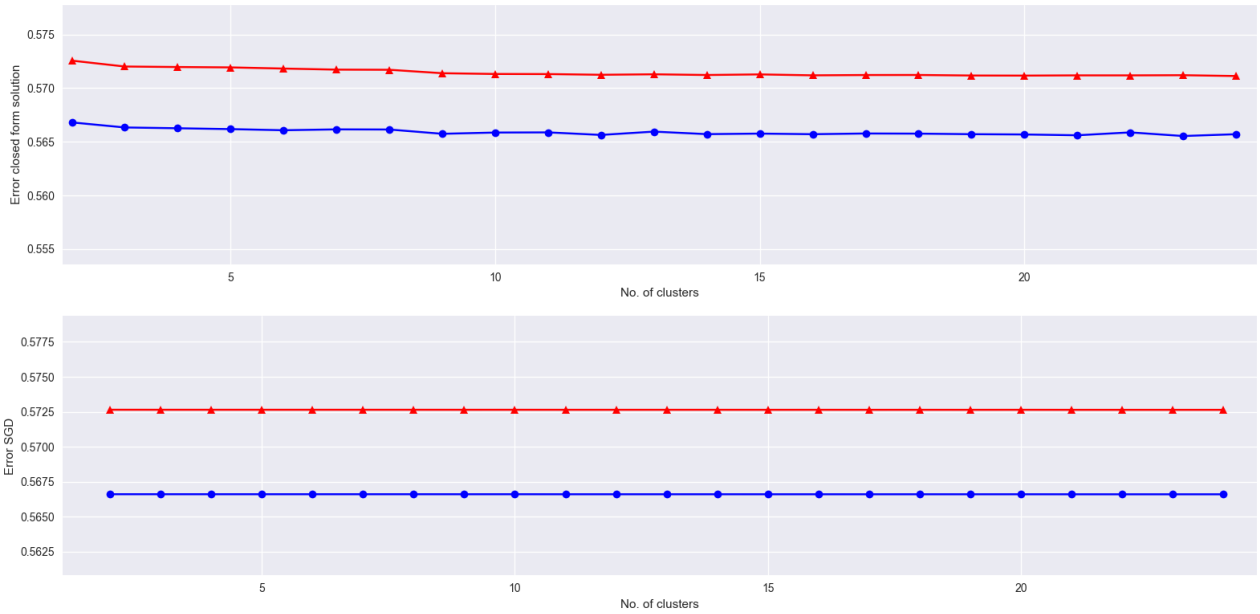


Figure: LETOR training error(red) vs. validation error(blue) for closed form(upper), SGD (lower)

We can see that both training and validation error decrease slightly with increase in number of clusters (M).

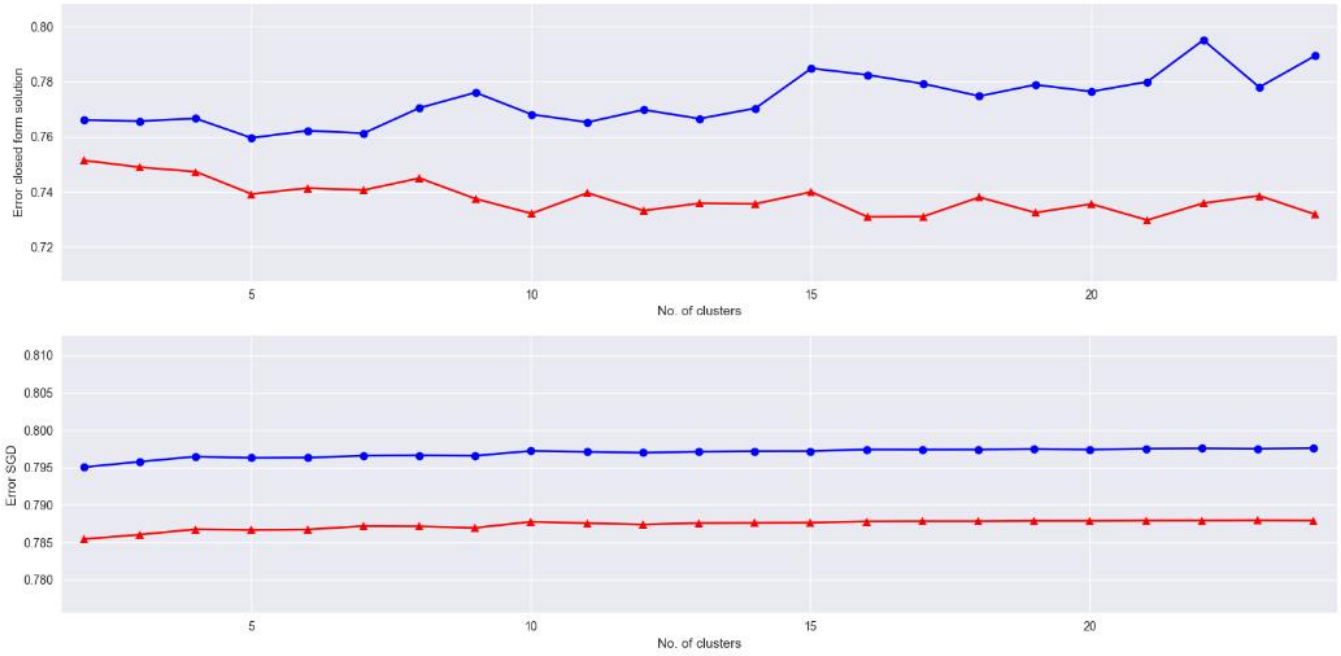


Figure: SYNTHETIC data training error(red) vs. validation error(blue) for closed form(upper), SGD (lower)

We can see that the training error decreases with increase in number of clusters (M) and validation error increases for our closed form solution. Hence, our model begins to overfit as the number of clusters increase.

**Complete Output with Lambda = 0.1**

```
-------------------------
-------------------------
Optimal Number of Clusters for LETOR Data using Closed form solution
=  23
Erms on LETOR Test Data Set using Closed Form Solution =
0.5642205688491557
Optimal Number of Clusters for LETOR Data using SGD =  24
Erms on LETOR Test Data Set using SGD =  0.5647038372947882
Optimal Number of Clusters for Synthetic Data using Closed form
Solution =  5
Erms on Synthetic Test Data Set using Closed Form Solution =
0.7570341368863621
Optimal Number of Clusters for Synthetic Data using SGD =  2
Erms on Synthetic Test Data Set using SGD =  0.7842299985645576
```