Project 2: Learning to Rank using Linear Regression

Name	UBIT name	Person No
Anant Gupta	anantram	50249127
Parikshit Deshmukh	pdeshmuk	50247649

23rd October, 2017

Problem Statement:

In this project we are using machine learning to solve a famous problem of information retrieval i.e. Learning to Rank (LeToR) problem. We implement the linear regression model on two given data set. The learn to Rank (LeToR) data set has 69623 query-document pairs (rows) along with 46 features for each row. We also implement the model on the given synthetic data set which has 20000 entries. We first fit the linear regression model to each data set using the closed-form solution and then by using Stochastic Gradient Descent (SGD) Solution.
After fitting the model we find the minimum difference of the ERMS value between the training and the validation data set and then report the test error.

We perform the below 4 steps to complete the task on both LeToR Data Set and Synthetic Data Set:

STEP #1
Data Partition

Here we partition both the data set. First, the LeToR data set is imported using 'genfromtext' function from the library 'NUMPY'. Then the loaded data is partitioned as per the below ratio:

Training Data Set – 80%

Validation Data Set – 10%

Test Data Set – 10%

Below table shows the exact distribution of the data:

Set	Start	End
Training Set	0	floor(0.8 * #records)
Validation Set	floor(0.8 * #records)+1	floor(0.8 * #records) + floor(0.1* #records
Testing Set	floor(0.8 * #records) + floor(0.1* #records) +1	End of File

STEP #2
Find the Closed form Solution

To find the closed form solution for both the data set we perform the below steps:
1) Find the clusters for the given data.

2) Find the spreads for the given data.

3) Compute the design matrix.

4) Start with a random value for the hyper parameters and then keep updating as we do the hyper parameter tuning (in step 4).

5) Find the Closed form solution using the model and the hyper parameters.

6) Compute the ERMS values for the training, validation and test data set.

STEP #3
Find the Stochastic Gradient Solution (Using Early Stopping Algorithm)

1) Design matrix that has been computed earlier.

2) A functions for determining the model parameters has been defined. This function uses early stopping algorithm where the program stops when it hits the minimum validation error after the patience parameter is overridden.

3) Using the model parameters, we predict the target values of the test data.

4) Find the ERMS values using the predicted target values from the above model.

Parameters used for Early Stopping: patience = 10, validation Steps = 10, learning_rate=0.1, epochs=100

STEP #4

Perform Hyper Parameter Tuning

1) Parameter Tuning for both Closed Form and SGS is done and the values are updated.

2) Parameter Tuning is done by finding the min value of the difference between ERMS values of Training and Validation data sets. When the values for hyper parameters are obtained, ERMS for the test data is determined and is reported.

Hyper Parameters

We use the validation data set to perform hyper tuning of parameters. Finally, we take the below values for the various hyper parameters after performing the tuning.

For LeToR Data Set (Closed):

Hyper Parameters	Symbol	Value
Number of clusters	M/K	14
Regularization Term	λ	0.01
Learning Rate	$\eta(\tau)$	0.1

For LeToR Data Set (SGD):

Hyper Parameters	Symbol	Value
Number of clusters	M/K	12
Regularization Term	λ	0.01
Learning Rate	$\eta(\tau)$	0.1
Parameters Used for Early Stopping		
Patience	P	10
Validation Steps	V steps	10
Epoch	e	100

For Synthetic Data Set (Closed):

Hyper Parameters	**Symbol**	**Value**
Number of clusters	M/K	18
Regularization Term	λ	0.01
Learning Rate	$\eta(\tau)$	0.1

For Synthetic Data Set (SGD):

Hyper Parameters	**Symbol**	**Value**
Number of clusters	M/K	10
Regularization Term	λ	0.01
Learning Rate	$\eta(\tau)$	0.1
Parameters used for Early Stopping		
Patience	p	10
Epoch	epoch	100
Validation Steps	Val steps	10

ERMS Values:

As described in the steps above we computed the ERMS values as below:

For LeToR Data Set

DATA SET	Closed Form Solution	Stochastic Gradient Solution
Training	0.5708	0.569
Validation	0.5679	0.577
Test	0.5633	0.5673

For Synthetic Data Set

DATA SET	Closed Form Solution	Stochastic Gradient Solution
Training	O.72	0.77
Validation	0.73	0.79073
Test	0.74	0.7940

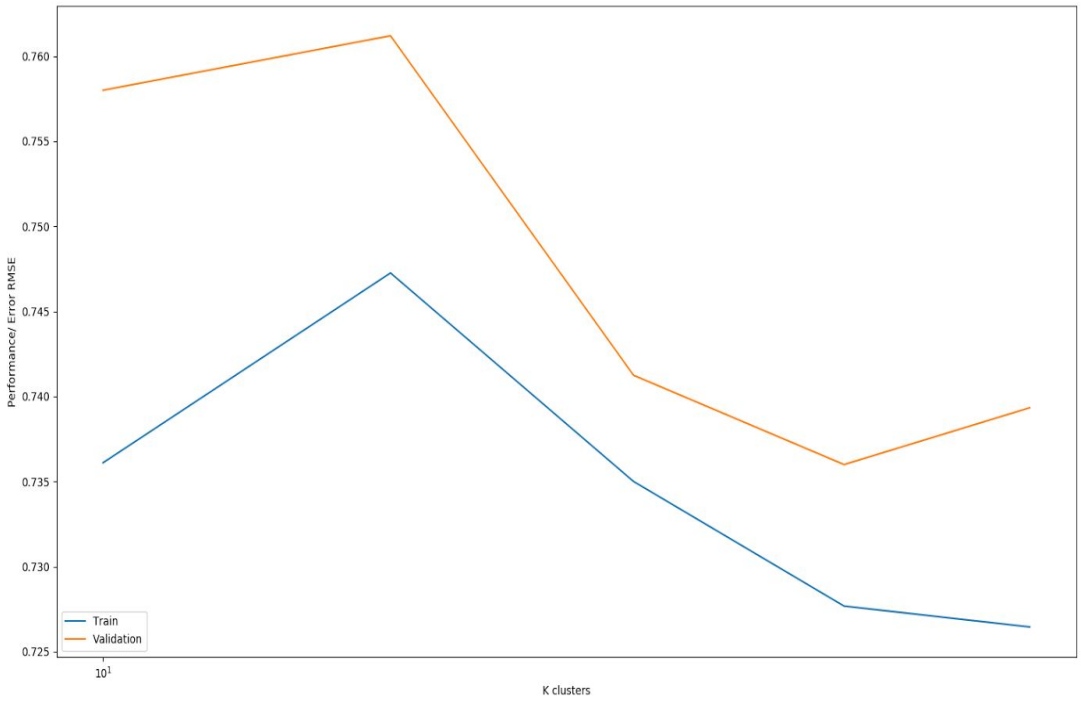
Printed out results from code:

```
personNumber: 50247649
erms_val_close_syn 0.7351207694405604
erms_tr_close_syn 0.7258052639249162
erms_test_close_syn 0.7463548141553978
KClose_syn 18
LambdaCloseSyn 0.01
erms_val_sgd_Syn 0.7907331989948544
erms_train_sgd_Syn 0.7797079616439568
erms_test_sgd_syn 0.7940543388466381
KSGD_syn 10
Lambda SGD_syn 0.31
erms_val_close_letor 0.5679089679521918
erms_tr_close_letor 0.5708503755929183
erms_test_close_letor 0.563320041561073
KClose_syn 14
LambdaClose_letor 0.01
^[^Terms_val_sgd_letor 0.5681804473998012
erms_train_sgd_letor 0.5714685929325389
erms_test_sgd_letor 0.5639527017881113
KSGD_letor 14
Lambda SGD_letor 0.01
```

```
erms_val_sgd_letor 0.5773629969727536
erms_train_sgd_letor 0.5696317424445312
erms_test_sgd_letor 0.5673284267930686
KSGD_letor 12
Lambda SGD_letor 0.01
```

Train and Validation Error Comparison:

For fixed lambda =0.01



For Lambda = 0.01 to 1