Project 2 : Handwriting Recognition using Predictive Models

Vandana Prasad Gaallee/50289877

November 2018

1 Introduction

Project 2 is identifying solving the Learning to Rank (LeToR) problem using machine learning approach. This project model is trained over LETOR dataset of 69623 query-document pair where each pair is represented using 46 feature vector. This LETOR problem is solved by both linear regression using closed form solution and the Learning to Rank (LeToR) problem and accuracy is calculated at each case.

2 Data Extraction

This problem is implemented using "AND" images samples extracted from CEDAR Letter dataset. The predictive models are built using tow datasets.

Human Observed Dataset : Human Observed Dataset have a total of 18 features for a pair of handwritten "AND" sample (9 features for each sample). The total dataset has equal number of sample and different writer pairs which is divided in to training, testing and validation set.

GSC Dataset using Feature Engineering : GSC Dataset have a total of 1024 features for a pair of handwritten "AND" sample (512 features for each sample). The total dataset is divided in to training, testing and validation set.

The following pre-processing steps are performed on the dataset.

1. The features of image pairs are concatenated for both Human-Observed(9+9) and GSC dataset(512+512).
2. The absolute value of subtraction of the each image pair features is computed for both the datasets.

3 Linear Regression

Linear regression using Basis Function Models is implemented to learn the handwriting similarity. The design matrix is generated using Basis function and weights are computed using error backpropagation.

3.1 Results Obtained

1. **Human Observed Dataset : Feature Concatenation Output**
   M = 40
   learning rate = 0.5
   Lambda = 0.05
   Accuracy Training = 50.15798
   Accuracy Validation = 50.0
   Accuracy Testing = 48.40764
   E-rms Training = 0.69612
   E-rms Validation = 0.70055
   E-rms Testing = 0.70731

2. **Human Observed Dataset : Feature Subtraction Output**
   M = 40
   learning rate 0.5
   Lambda 0.05
   Accuracy Training = 51.4218
   Accuracy Validation = 50.63291
   Accuracy Testing = 49.68153
   E-rms Training = 0.59776
   E-rms Validation = 0.60364
   E-rms Testing = 0.61116

3. **GSC Dataset : Feature Concatenation Output**
   Accuracy Training = 49.87768
   Accuracy Validation = 49.59458
   Accuracy Testing = 49.39186
   Erms Training = 0.70797
   Erms Validation = 0.70997
   Erms Testing = 0.71139

4. **GSC Dataset : Feature Subtraction Output**
   Accuracy Training = 49.87768
   Accuracy Validation = 49.59458
   Accuracy Testing = 49.39186
   Erms Training = 0.70797
   Erms Validation = 0.70997
   Erms Testing = 0.71139

3.2 Results on changing HyperParameters

1. **Case 1 : Experimenting with HyperParameter Lamda** Lambda is the regularization parameter added to error function which controls the

value of weights and helps in obtaining better generalized model. Below is the graph representing the change in RMS error with respect to lamda

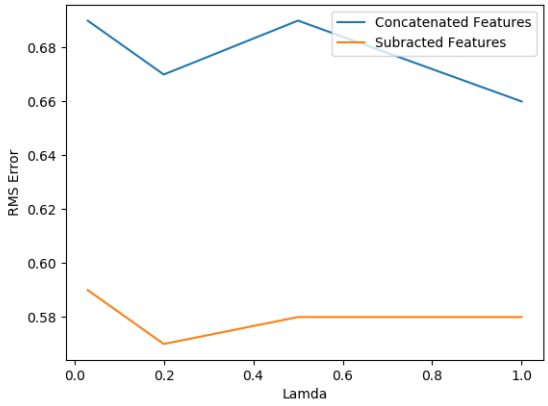

Figure 1: RMS Error with respect to lamda

2. **Case 2 : Experimenting with HyperParameter Learning rate** Learning rate controls how the weights are adjusted with respect to loss gradient. A higher learning rate would converge and train the model quickly, Below graph represents the change in accuracy with respect to learning rate.

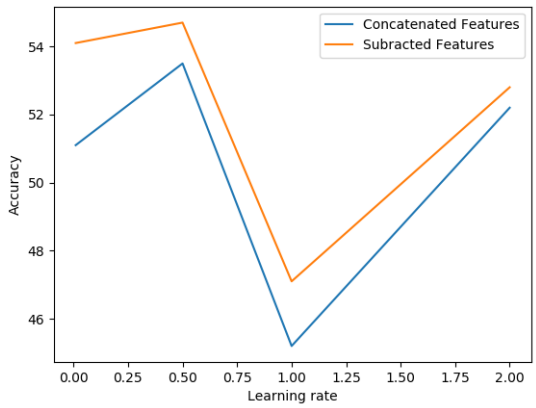

Figure 2: Accuracy with respect to Learning rate

3. **Case 3 : Experimenting with HyperParameter M** M is the number of basis functions used to compute the phi matrix. Observations are

recorded for each M value and graph is displayed for Concatenation and subtraction dataset.

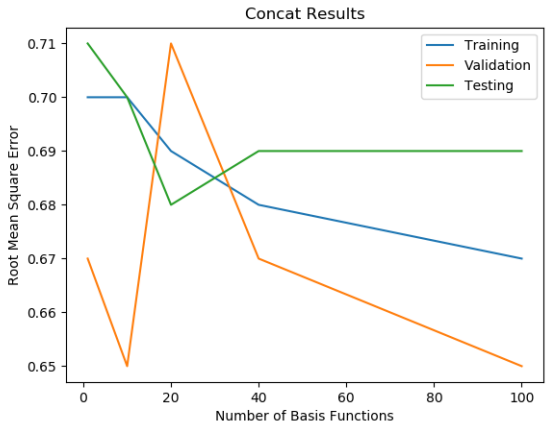

Figure 3: RMS Error with respect to Number of Basis Functions

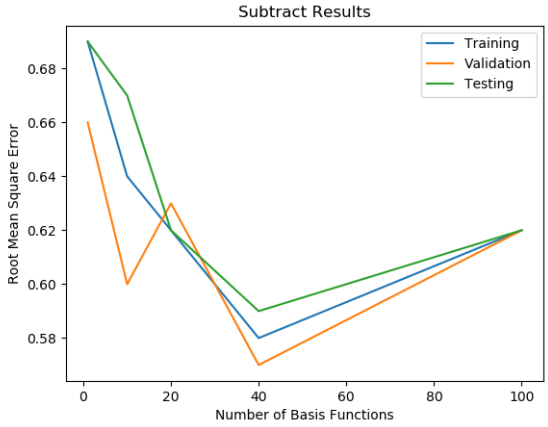

Figure 4: RMS Error with respect to Number of Basis Functions

4 Logistic Regression

Logistic Regression is a Machine Learning classification algorithm that is used to predict the ouput given a input. It is implemented using Gradient Descent to train the model using a group of hyperparameters on each of the 4 input datasets. It is given by formula

$$y(x, w) = \sigma(w^T X) \tag{1}$$

4.1 Results Obtained

1. **Human Observed Dataset : Feature Concatenation Output**
   Accuracy Training = 50.15798
   Accuracy Validation = 54.43038
   Accuracy Testing = 55.41401
   Erms Training = 0.69405
   Erms Validation = 0.67418
   Erms Testing = 0.65584

2. **Human Observed Dataset : Feature Subtraction Output**
   Accuracy Training = 53.5545
   Accuracy Validation = 56.96203
   Accuracy Testing = 62.42038
   Erms Training = 0.49945
   Erms Validation = 0.49928
   Erms Testing = 0.49606

3. **GSC Dataset : Feature Concatenation Output**
   Accuracy Training = 50.25775
   Accuracy Validation = 50.09786
   Accuracy Testing = 50.60115
   Erms Training = 0.70528
   Erms Validation = 0.70641
   Erms Testing = 0.70272

4. **GSC Dataset : Feature Subtraction Output**
   Accuracy Training = 50.25775
   Accuracy Validation = 50.45435
   Accuracy Testing = 51.11841
   Erms Training = 0.66081
   Erms Validation = 0.65812
   Erms Testing = 0.65461

4.2 Results on changing HyperParameters

1. **Case 1 : Experimenting with HyperParameter Lamda** Below is the graph representing the change in RMS error with respect to lamda.

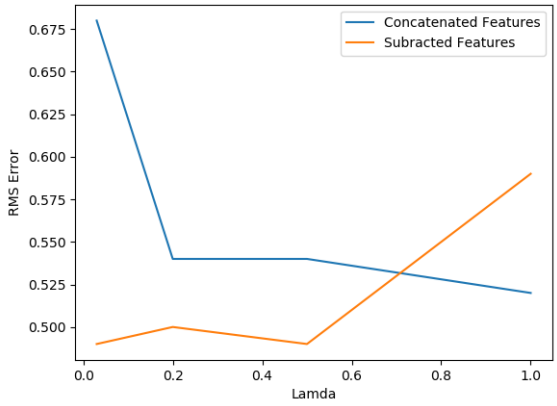

Figure 5: RMS Error with respect to lamda

2. **Case 2 : Experimenting with HyperParameter Learning rate** Below graph represents the change in accuracy with respect to learning rate.

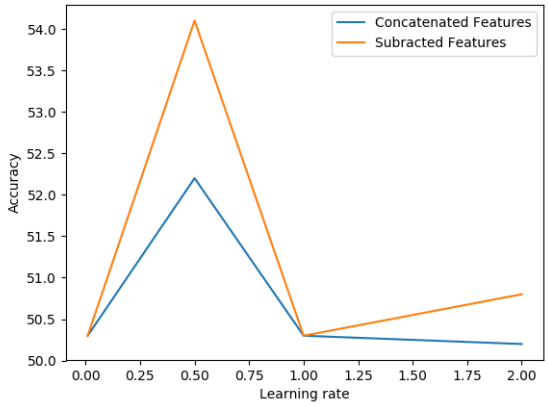

Figure 6: Accuracy with respect to Learning rate

5 Neural Networks

5.1 Results Obtained

1. **Human Observed Dataset : Feature Concatenation Output**
   Errors: 155 Correct :160
   Testing Accuracy: 50.79365079365079

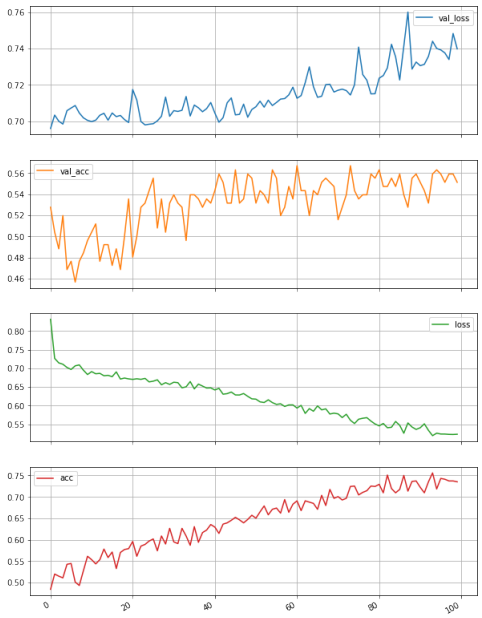

Figure 7: Accuracy and Loss for Concatenated Features

2. **Human Observed Dataset : Feature Subtraction Output**
   Errors: 168 Correct :147
   Testing Accuracy: 46.666666666666664

3. **GSC Dataset : Feature Concatenation Output** Errors: 1448 Correct :12858
   Testing Accuracy: 89.87837271075072

4. **GSC Dataset : Feature Subtraction Output**
   Errors: 2973 Correct :11333
   Testing Accuracy: 79.21850971620299

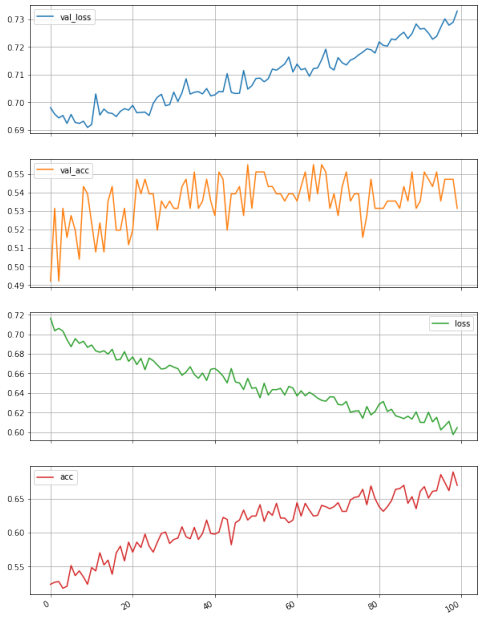

Figure 8: Accuracy and Loss for Subtracted Features

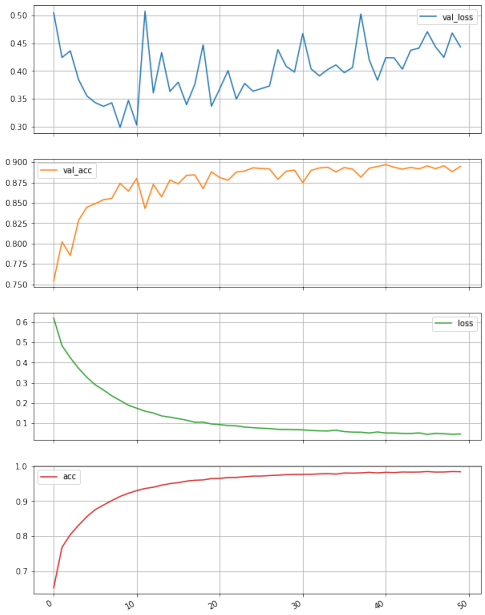

Figure 9: Accuracy and Loss for Concatenated Features

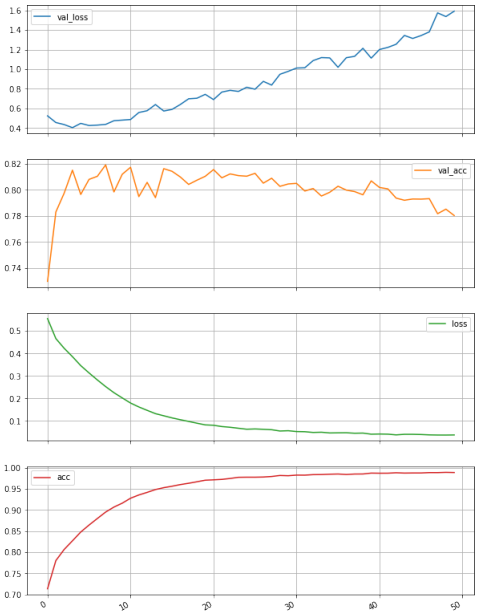

Figure 10: Accuracy and Loss for Subtracted Features

6 Comparison of Predictive Models

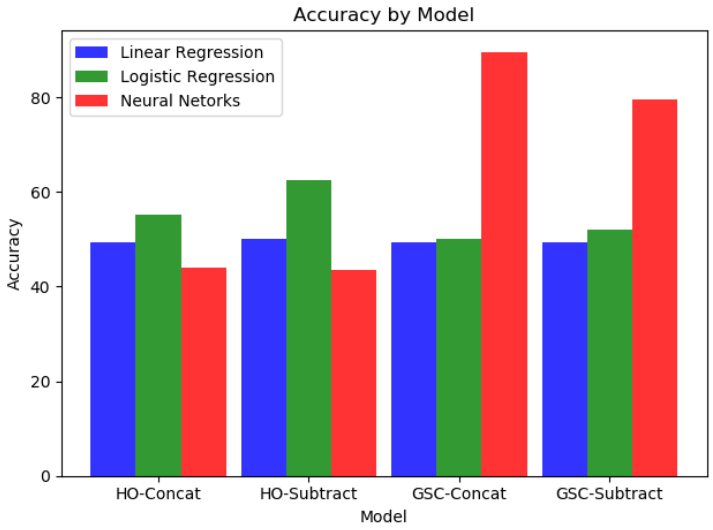

Figure 11: Accuracy with respect to each Model

6.1 Inference

From the above comparison graph, the following conclusions are made

1. Logistic Regression gave a slightly better performance than Linear regression for all four datasets.
2. Logistic Regression on subracted features performed better than concatenated dataset.
3. High accuracy for GSC dataset is obtained using neural networks

7 Conclusion

In this project, predictive models are implemented over all four datasets to measure the similarity between handwritten texts. Comparisons are made on the accuracy obtained from different models for the four datasets

References

Pattern Recognition and Machine Learning by Christopher M. Bishop