Project 2: Learning to Rank using Linear Regression

Anonymous Author(s)
Affiliation
Address
email

Abstract

The goal of this project is to solve the handwriting comparison task in forensics. We formulate this as a problem of linear regression where we map a set of input features x to a real-valued scalar target y(x,w).

Our task is to find similarity between the handwritten samples of the known and the questioned writer by using linear regression models and Neural Network.

Each instance in the CEDAR "AND" training data consists of set of input features for each hand-written "AND" sample. The features are obtained from two different sources:

(a) Human Observed features: Features entered by human document examiners manually.

(b) GSC features: Features extracted using Gradient Structural Concavity (GSC) algorithm.

The target values are scalars that can take two values {1:same writer, 0:different writers}. Although the training target values are discrete we use regression models to obtain real values which is more useful for finding similarity.

We have two objectives Three this project:

1. Train a Linear Regression model on both Human read feature data-set and GSC features data-set using Concatenation and Subtraction technique.

2. Train a Logistic Regression model on both Human read feature data-set and GSC features data-set using Concatenation and Subtraction technique.

3. Train a Neural Network model (Keras) on both Human read feature data-set and GSC features data-set using Concatenation and Subtraction technique.

1 Types of Datasets:

Based on feature extraction process, we have provided two datasets:

1.1 Human Observed Dataset:

The Human Observed dataset shows only the cursive samples in the data set, where for each image the features are entered by the human document examiner. There are 9 distinct features for each image in human observed dataset.

Submitted to 32nd Conference on Neural Information Processing Systems (NIPS 2018). Do not distribute.

Both concatenation and substraction has been performed based on the image id. After performing concatenation based on the image id, each sample will have 18 featue values and the dataset will look like the following image.

img_id_A	img_id_B	f_{A1}	f_{A2}	f_{A3}	f_{A4}	f_{A5}	f_{A6}	f_{A7}	f_{A8}	f_{A9}	f_{B1}	f_{B2}	f_{B3}	f_{B4}	f_{B5}	f_{B6}	f_{B7}	f_{B8}	f_{B9}	t
1121a_num1	1121b_num2	2	1	1	3	2	2	0	1	2	2	1	1	0	2	2	0	3	2	1
1121a_num1	1386b_num1	2	1	1	3	2	2	0	1	2	3	1	1	0	2	2	0	1	2	0

Figure 1: Neural Network

1.2 GSC Dataset:

Gradient Structural Concavity algorithm generates 512 features for an input handwritten "AND"image. The dataset is named as "GSC-Features-Data". Similar to the Human observed dataset.

Both concatenation and substraction has been performed based on the image id. After performing concatenation based on the image id, each sample will have 1024 featue values and the dataset will look like the following image.

img_id_A	img_id_B	f_{A1}	f_{A2}	f_{A3}	f_{A4}	f_{A5}	f_{A6}	. . .	f_{A512}	f_{B1}	f_{B2}	f_{B3}	f_{B4}	f_{B5}	f_{B6}	. . .	f_{B512}	t
1121a_num1	1121b_num2	0	1	1	0	1	0	. . .	0	0	1	1	0	0	1	. . .	1	1
1121a_num1	1386b_num1	0	1	1	0	1	0	. . .	0	1	1	1	0	1	0	. . .	0	0

Figure 2: Neural Network

1.3 Data Cleansing:

The GSC dataset contains several such features which contains all zero values (or all same values) which in terms generate a determinant of 0. These makes the dataset "non-inversable".

For this reason, after performing concatenation/subtraction and removing the img_id columns and target column, all the features containing all 0 values are removed from the GSC dataset.

2 E_RMS Graphs:

The following diagrams describes the E_RMS values for each iteration plotted in a line graph:

2.1 Linear Regression on Human observed dataset:

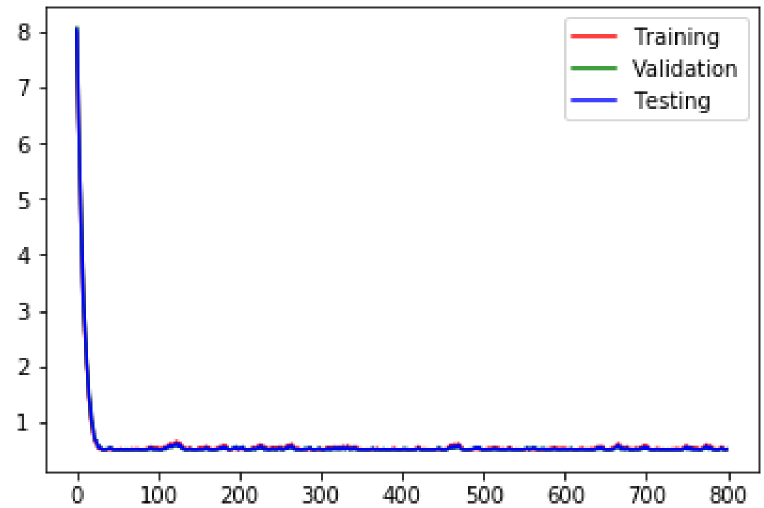

Figure 3: Concatenation Technique

E_rms Training = 0.49908
E_rms Validation = 0.49628
E_rms Testing = 0.49789

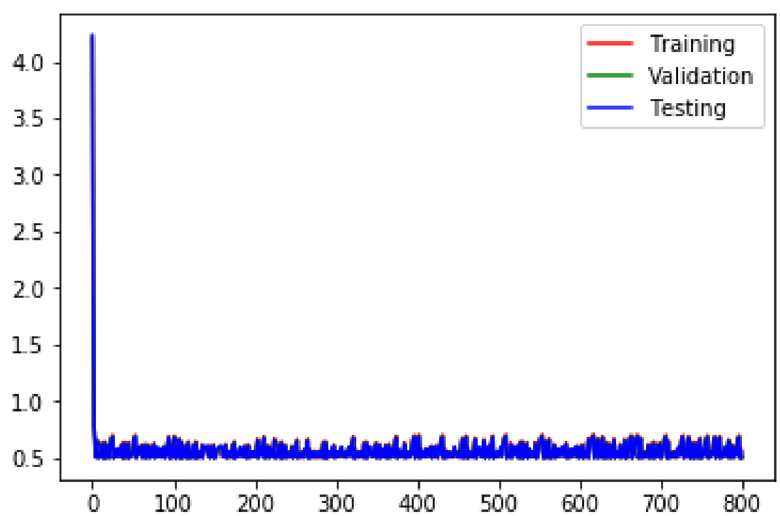

Figure 4: Subtraction Technique

E_rms Training = 0.49993
E_rms Validation = 0.49704
E_rms Testing = 0.49898

2.2 Logistic Regression on GSC dataset:

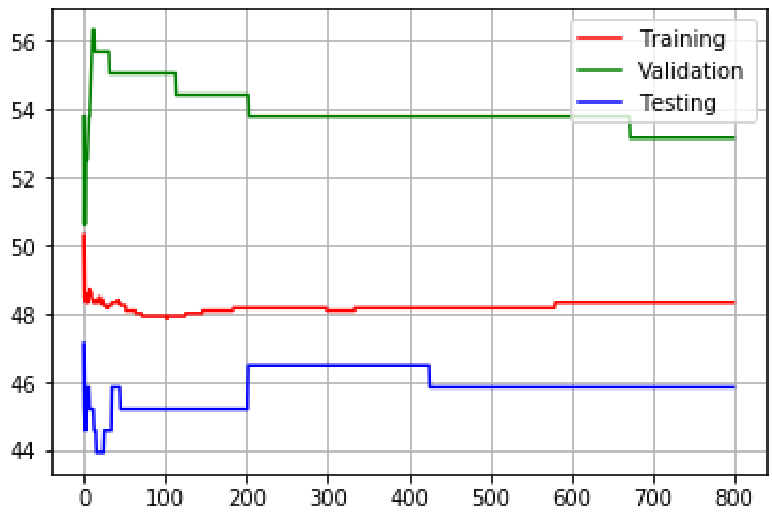

Figure 5: Concatenation Technique

Training_accuracy: 50.00
Validation_accuracy: 51.89873
Testing_accuracy: 47.7707

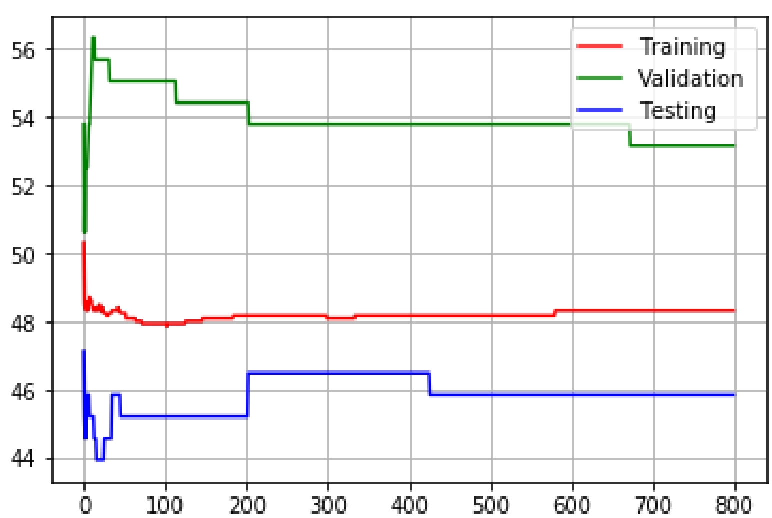

Figure 6: Subtraction Technique

Training_accuracy: 50.31596
Validation_accuracy: 56.32911
Testing_accuracy: 47.13376

2.3 Neural Network Model on Human observed dataset:

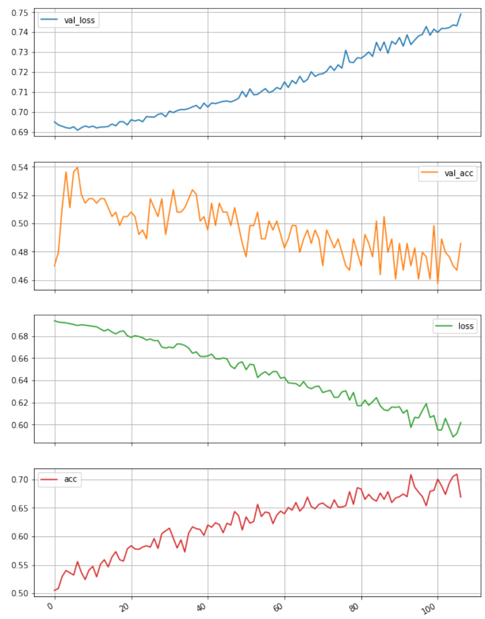

Figure 7: Concatenation Technique

Training_accuracy: 0.5238
Validation_accuracy: 0.4858
Testing_accuracy: 0.4902

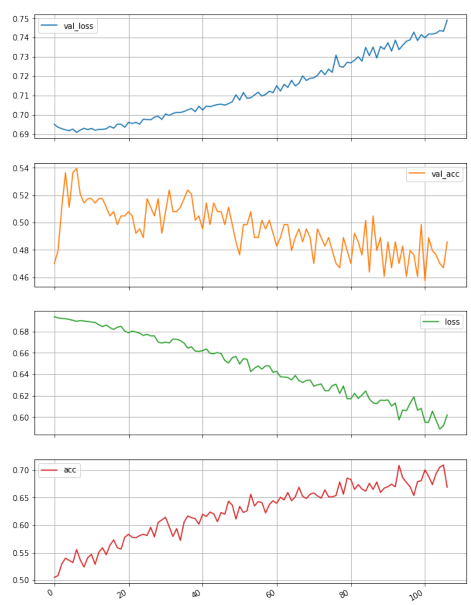

Figure 8: Subtraction Technique

Training_accuracy: 0.5127
Validation_accuracy: 0.4700
Testing_accuracy: 0.4895

2.4 Linear Regression on GSC dataset:

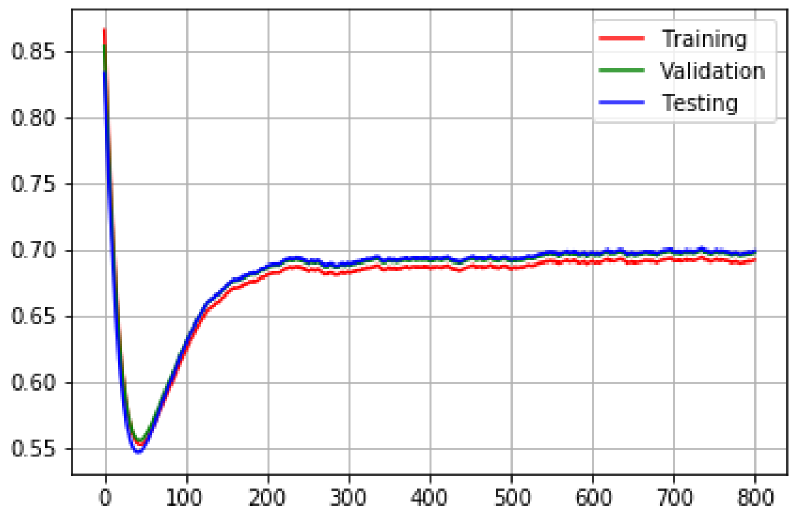

Figure 9: Concatenation Technique

E_rms Training = 0.55253
E_rms Validation = 0.55582
E_rms Testing = 0.54671

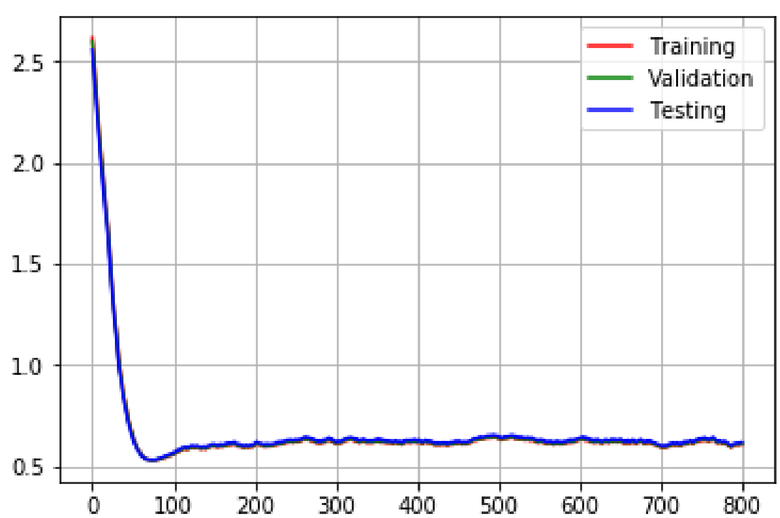

Figure 10: Subtraction Technique

E_rms Training = 0.5284
E_rms Validation = 0.52922
E_rms Testing = 0.52997

2.5 Logistic Regression on GSC dataset:

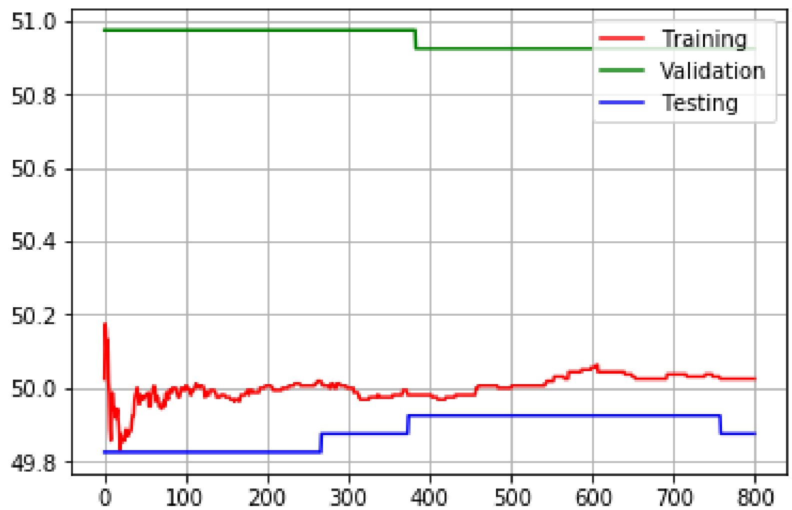

Figure 11: Concatenation Technique

Training_accuracy: 50.14375
Validation_accuracy: 50.07504
Testing_accuracy: 48.77439

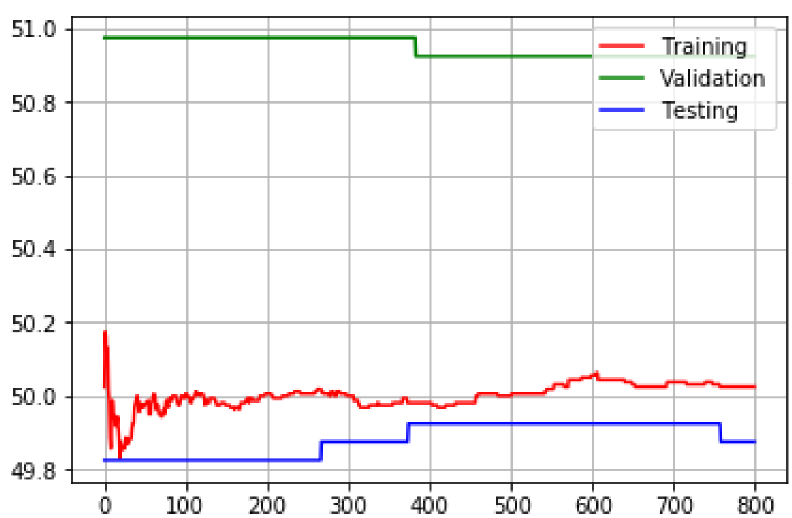

Figure 12: Subtraction Technique

Training_accuracy: 50.175
Validation_accuracy: 50.97549
Testing_accuracy: 49.92496

2.6 Neural Network on GSC dataset:

Figure 13: Concatenation Technique

Training_accuracy: 0.9982
Validation_accuracy: 0.8875
Testing_accuracy: 0.8528

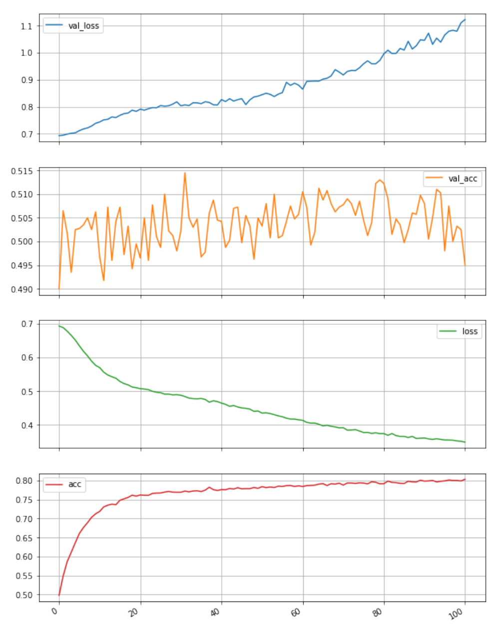

Figure 14: Subtraction Technique

Training_accuracy: 0.8875
Validation_accuracy: 0.4950
Testing_accuracy: 0.5249

3 references

https://www.coursera.org/learn/machine-learning

Bishop - Pattern Recognition And Machine Learning - Springer 2006
https://en.wikipedia.org/wiki/Machine$_{l}earning$