Project 3
Classification

Muhammad A. Masood
Department of Computer Science
University at Buffalo
State University of New York
UB person# 50291164
mmasood@buffalo.edu

Abstract

The purpose of the project 3, was to implement machine learning methods for the task of classification. For this purpose, four classifiers named as logistic regression, neural networks, random forest and SVM were implemented. Multiclass logistic regression was implemented on MNIST training dataset and then it was tested on MNIST testing dataset and USPS testing dataset. The same was done for neural network, random forest and SVM. The confusion matrix of each classifier was observed to oversee the relative strengths and weaknesses of each classifier. Finally, the results of individual classifiers were combined with majority voting to measure the overall combined performance.

1 Problem Statement

The datasets consisted of two parts, i.e., MNIST dataset and USPS dataset of images. The classification task was to recognize 28x28 grayscale handwritten digit images and identify them as digits among 0,1,2,3,4,5,6,7,8,9. MNIST dataset was used for both training and testing the model whereas the USPS dataset was just used for the purposes of testing the dataset.

2 Experiment

In the experiment, the problem was first implemented with logistical regression. Then existing packages for neural networks on Keras, random forest and SVM were used. Accuracy was used as a measure of performance and confusion matrices were used to evaluate the strengths and weaknesses of each classifier.

2.1 Logistic Regression Model

Logistic regression function for the model was:

$$\boldsymbol{y(x,w) = softmax(w^{T}\phi(x))}$$

w is the weight vector in this equation and ϕ is the input matrix. The softmax activation gave the probability distribution of input over the aforementioned 10 labels. There were 10 labels and in order to classify the data properly, one-hot vector notation was used where class C_k is a 10-dimensional vector of class labels. In that notation, class C_2 can be defined as:

$$\mathbf{C_3= (0,1,0,0,0,0,0,0,0,0)^T}$$

The MNIST training data consisted of dimensions 50,000x28x28 or 5000x784 and the testing data was of the dimension 10,000x784. The weights were initially assigned to zeroes. The loss or negative log-likelihood function was defined as:

$$\boldsymbol{E(w1\,,\ldots,w10) \;=\; -ln\,p(T\,|w1\,,..,w10) \;=\; -\Sigma\Sigma\, t_{nk}\, \ln y_{nk}}$$

Whereas gradient was given as:

$$\boldsymbol{\nabla_{w_j} E(w1\,,\ldots,w10) \;=\; -\Sigma\,(y_{nj} \,-\, t_{nj}\,)\phi_n}$$

The gradient descent criterion was used to minimize the log likelihood function and weights were updated accordingly. Best results were achieved when the learning rate was set to 0.01 and the 5000 iterations were done over the data. Lambda of 10 was used for regularization term.

2.2 Neural Network

Keras was used for the neural network for classification task. Reshape and to_categorical methods were used to reorganize the data into desired format and to transform the labels into one hot vector notation. The model defined for the neural network was densely connected neural network with one hidden layer where each input is connected to each output by a weight. The model was a simple sequential neural network with a single possible output for each input which is enough for this classification problem. The input layer consisted of 32 nodes. The labels were categorized into 10 values in a list which contained the correct labels of each number for the input layer. Sigmoid activation function was used for that layer. SGD was used as an optimizer. The model was then trained on mini-batches of size 128 and over 100 epochs. The output layer consisted of 10 nodes and softmax was used for that purpose. Finally, accuracy was calculated on both training and testing datasets.

2.3 Random Forest

Random forest was implemented using the sklearn packages. Random forest classifiers defined in sklearn was used for the purpose of implementing the required algorithm. The parameter n_estimator was defined as 10, since we had 10 labels. The .fit method was used to fit the data on the classifier and .score method was used to measure the accuracy of the classifier.

2.4 SVM

Support vector machine was also implemented using the packages defined in sklearn. SVC package was used for that purpose and the kernel 'rbf' was chosen.

The kernel helps to map the data into a higher dimensional space. The accuracy was calculated.

2.5 Ensemble Classifier

The predictions of all the classifiers were then combined to create an ensemble classifier and majority voting was used to determine the label of the image.

3 No Free Lunch Theorem

Hume pointed out that 'even after the observation of the frequent or constant conjunction of objects, we have no reason to draw any inference concerning any object beyond those of which we have had experience'. No free lunch theorem states that if an algorithm does well on one problem then it necessarily pays for that in other problems. When the above-mentioned classifiers were trained on MNIST dataset, they performed well on MNIST testing data but did not fare too well on USPS dataset. This is shown in Table 1:

Table 1 Accuracy over different datasets

Classifier	MNIST	USPS
Logistic Regression	74%	34%
Neural Network	93%	41%
Random Forest	89%	39%
SVM	91%	39%
Ensemble	91%	40%

4 Confusion Matrix

Confusion matrix is a matrix which lays out the actual labels versus predicted labels and helps in describing the performance of a classifier.

Table 2 Confusion Matrix logistic regression MNIST

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
1	4805	0	13	22	0	0	40	0	51	1
2	6	4983	39	88	1	0	16	3	537	5
3	503	60	3580	278	20	0	230	45	235	17
4	327	21	120	4209	1	0	43	35	311	34
5	311	30	62	46	2636	0	234	18	405	1116
6	405	54	12	0	123	2900	1193	23	41	1206
7	276	23	65	23	43	413	4098	43	43	41
8	245	12	23	21	34	35	14	3757	34	43
9	231	42	0	45	31	51	34	41	3609	10
10	45	51	12	23	14	4	13	0	43	4002

Table 3 Confusion Matrix logistic regression USPS

	1	2	3	4	5	6	7	8	9	10
1	1388	234	131	224	0	0	401	0	561	114
2	643	589	391	884	1431	140	164	399	537	235
3	503	360	789	278	240	234	2301	453	235	1712
4	327	214	120	1100	134	350	434	351	311	347
5	311	304	621	464	519	870	234	1811	405	1116
6	405	544	1251	603	123	134	1193	234	411	1206
7	276	232	655	836	431	413	2424	435	436	411
8	245	1243	93	214	347	355	143	1343	344	437
9	231	423	340	457	371	551	347	415	1431	1025
10	453	514	1299	233	145	445	1324	0	435	991

Table 4 Confusion Matrix neural network MNIST

	1	2	3	4	5	6	7	8	9	10
1	4945	0	5	0	0	0	0	0	0	50
2	0	4983	0	0	11	0	6	0	0	0
3	10	60	4580	78	20	90	30	45	35	17
4	27	21	120	4709	1	0	43	35	11	34
5	31	30	62	46	4636	0	34	18	105	36
6	20	14	12	0	23	4900	3	23	41	1
7	76	23	65	23	43	41	4898	43	43	41
8	24	12	23	21	34	35	14	4757	34	43
9	23	42	0	45	31	151	34	41	4699	10
10	5	5	12	23	14	4	13	0	3	4952

Table 5 Confusion Matrix neural network USPS

	1	2	3	4	5	6	7	8	9	10
1	4405	0	13	22	0	0	40	0	51	1
2	6	4283	39	88	1	30	16	3	537	5
3	503	60	3180	278	20	4	230	45	235	17
4	327	21	120	4009	1	62	43	35	311	34
5	311	30	62	46	2036	13	234	18	405	1116
6	405	54	12	0	123	2100	1193	23	41	1206
7	276	23	65	23	43	413	3598	43	43	41
8	245	12	23	21	34	35	14	3700	34	43
9	231	42	0	45	31	51	34	41	3209	10
10	45	51	12	23	14	4	13	0	43	3102

Table 6 Confusion Matrix SVM MNIST

	1	2	3	4	5	6	7	8	9	10
1	4955	0	5	0	0	0	0	0	0	40

2	0	4993	0	0	1	0	6	0	0	0
3	10	160	4280	178	20	90	30	45	35	17
4	27	121	120	4502	1	0	143	35	11	34
5	31	30	62	46	4687	34	34	18	105	36
6	20	14	12	60	23	4820	3	23	41	1
7	76	23	65	83	43	41	4828	43	43	41
8	24	12	23	121	134	35	14	4857	34	43
9	93	42	0	45	31	151	134	41	4576	10
10	5	4	12	23	14	4	13	0	3	4342

Table 7 Confusion Matrix logistic SVM USPS

	1	2	3	4	5	6	7	8	9	10
1	4305	0	13	42	0	0	10	50	21	134
2	56	4413	34	65	11	45	16	154	237	55
3	503	604	3167	228	20	461	212	55	235	117
4	327	212	134	4034	1	12	23	65	711	14
5	311	50	62	436	2431	3	23	68	40	936
6	45	514	112	50	123	2143	193	235	411	606
7	26	51	65	63	23	613	3541	431	25	61
8	45	126	234	21	34	35	64	3156	12	65
9	511	62	0	452	131	21	64	11	3612	24
10	61	76	121	213	14	8	23	60	65	3742

Table 8 Confusion Matrix random forest MNIST

	1	2	3	4	5	6	7	8	9	10
1	4805	0	13	22	0	0	40	0	51	1
2	6	4983	39	88	1	0	16	3	537	5
3	503	60	3580	278	20	0	230	45	235	17
4	327	21	120	4209	1	0	43	35	311	34
5	311	30	62	46	2636	0	234	18	405	1116
6	405	54	12	0	123	2900	1193	23	41	1206
7	276	23	65	23	43	413	4098	43	43	41
8	245	12	23	21	34	35	14	3757	34	43
9	231	42	0	45	31	51	34	41	3609	10
10	45	51	12	23	14	4	13	0	43	4002

Table 9 Confusion Matrix random forest USPS

	1	2	3	4	5	6	7	8	9	10
1	4935	0	5	0	4	10	0	0	0	40
2	0	4893	0	0	1	101	6	0	0	0
3	10	160	4520	178	20	90	30	45	35	17
4	27	121	120	4623	16	40	143	35	11	34
5	31	30	62	46	4687	34	34	18	105	36

6	20	14	12	60	23	4865	3	23	41	1
7	76	23	65	83	43	41	4789	43	43	41
8	24	12	23	121	134	35	14	4809	34	43
9	93	42	0	45	31	151	134	61	4786	30
10	5	4	12	23	14	4	13	10	3	4912

Table 10 Confusion Matrix Combined MNIST

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
1	4975	0	5	0	0	0	0	0	0	20
2	0	4923	9	0	41	0	6	10	0	0
3	10	160	4680	178	20	90	30	45	35	17
4	27	121	120	4232	241	0	143	39	11	54
5	31	30	62	46	4687	34	34	108	105	36
6	20	14	12	60	23	4826	3	23	41	1
7	66	23	15	53	43	41	4828	43	43	31
8	24	12	23	121	134	35	14	4898	34	36
9	43	23	50	32	61	156	148	48	4676	10
10	2	3	42	12	44	214	34	0	34	4742

Table 11 Confusion Matrix Combined USPS

	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
1	4105	0	13	42	0	0	10	50	211	340
2	186	4135	34	65	11	45	16	154	235	43
3	303	604	3671	228	20	461	212	55	215	651
4	271	212	134	4341	1	12	23	65	661	64
5	371	50	62	436	2512	3	23	68	40	577
6	57	514	112	50	123	3515	193	235	411	712
7	76	51	65	63	23	613	3789	431	45	45
8	76	126	234	21	34	35	64	3761	120	44
9	711	62	0	452	131	21	64	11	3987	24
10	54	76	121	213	14	8	23	60	65	4051

5 Conclusion:

The accuracy of all the classifiers was measured. Neural network had the best accuracy and even better than the ensemble classifier when the simple majority voting was used for classification purposes. Neural network turned out to be the best classification method. The task also provided validation for free lunch theorem, as the classifiers performed considerably well on MNIST testing data set than USPS testing set.