Machine Learning Project 3

Dhanashree Solanke
Department of Computer Science
University at Buffalo
Buffalo, NY 14214
ddsolank@buffalo.edu

Abstract

We take up the problem of implementing the classification problem for handwritten digits. We implement four different models for the same: Logistic Regression, Random Forest, SVM and Neural Network. We then create a ensemble classifier for these models and compare the accuracies, strengths and weaknesses of each model.

1 Datasets

We will use the MNIST database for training our models. For testing, we'll use part of MNIST and USPS dataset. MNIST and USPS are large databases for handwritten digits.

2 Logistic Regression

Multinomial Regression(Softmax) is a generalization of logistic regression that we can use for multi-class classification.
In softmax regression (SMR), we replace the sigmoid logistic function by the so-called softmax function φ:

$$P(y = j \mid z^{(i)}) = \phi_{softmax}(z^{(i)}) = \frac{e^{z^{(i)}}}{\sum_{j=0}^{k} e^{z_k^{(i)}}},$$

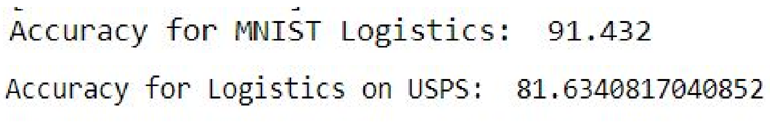
where we define the net input z as

$$z = w_0 x_0 + w_1 x_1 + \ldots + w_m x_m = \sum_{l=0}^{m} w_l x_l = \mathbf{w}^T \mathbf{x}.$$

2.1 Model

We use Softmax Regression for our Logistic Model. We update the weights using the above gradient descent and softmax as our activation function We train our model using

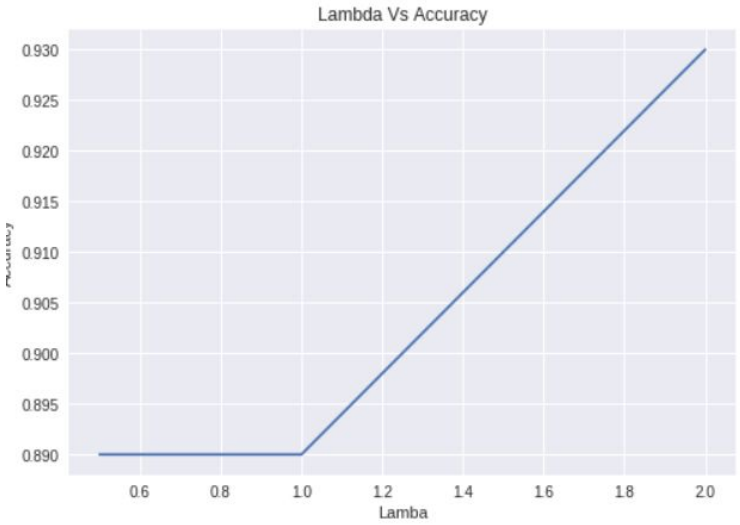
MNIST and test using both MNIST and USPS.



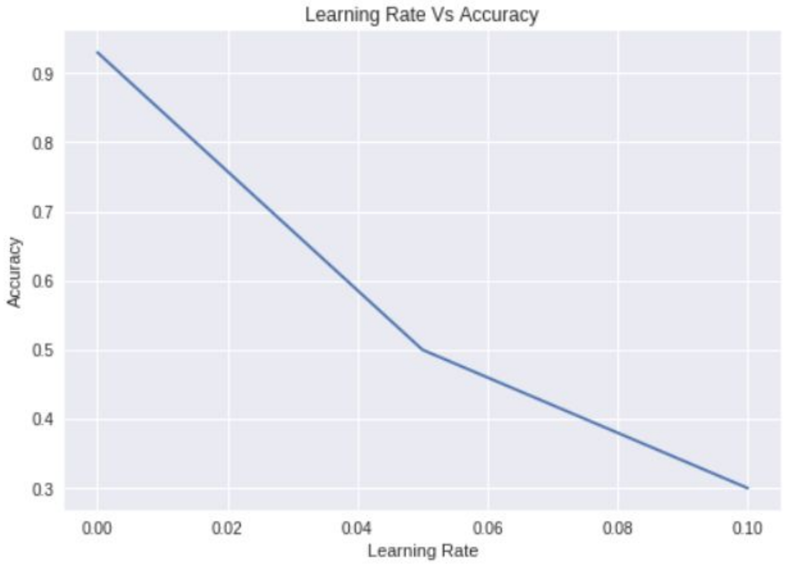
```
Accuracy for MNIST Logistics:  91.432
Accuracy for Logistics on USPS:  81.6340817040852
```

2.2 Changing the Hyper Parameters

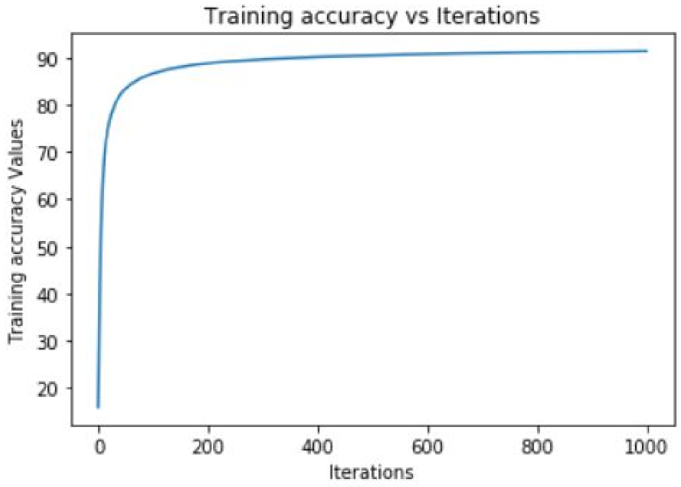
- Changing Lambda Values



- Changing Learning Rate



- Accuracy Over Iterations



The best accuracy is obtained at iterations = 1000, lambda =2 and learning rate = 0.000012

3 Neural Networks

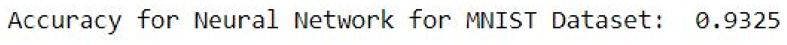
Neural nets are a means of doing machine learning, in which a computer learns to perform some task by analyzing training examples. Usually, the examples have been hand-labeled in advance.

Modeled loosely on the human brain, a neural net consists of thousands or even millions of simple processing nodes that are densely interconnected. Neural nets are organized into layers of nodes, and they're "feed-forward," meaning that data moves through them in only one direction. An individual node might be connected to several nodes in the layer beneath it, from which it receives data, and several nodes in the layer above it, to which it sends data.

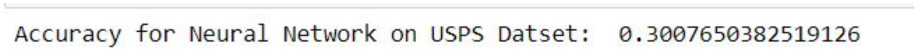
3.1 Model

We have developed our model using Keras package. It is a simple single hidden layer model. We have used MNIST dataset to train our model. For our first layer, we have 32 nodes and sigmoid as our activation function. The next layer has 10 hidden nodes where activation function is softmax. We have used sgd as our optimizer, cross entropy as our loss function.

We have tried tuning the hyper-parameters to obtain the best accuracy for MNIST and USPS which is as follows:



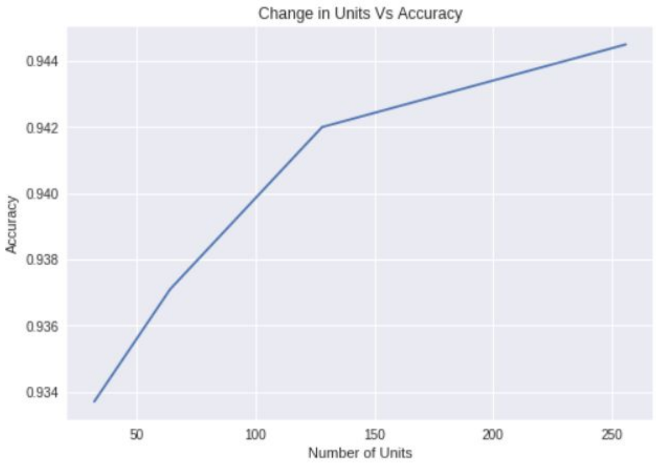
```
Accuracy for Neural Network for MNIST Dataset:  0.9325
```



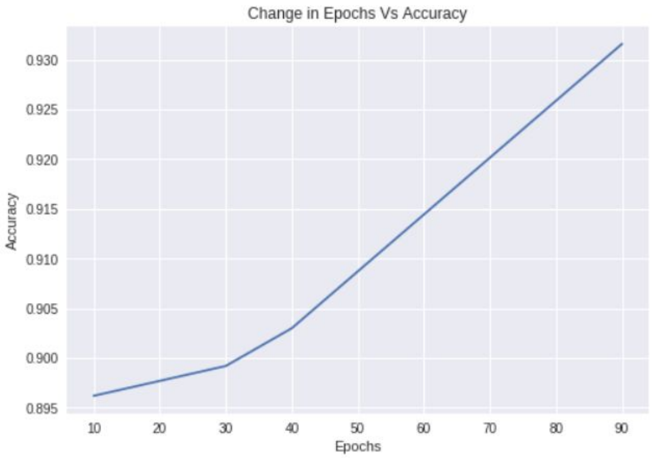
```
Accuracy for Neural Network on USPS Datset:  0.3007650382519126
```

3.1 Changing the Hyper Parameters for MNIST Dataset

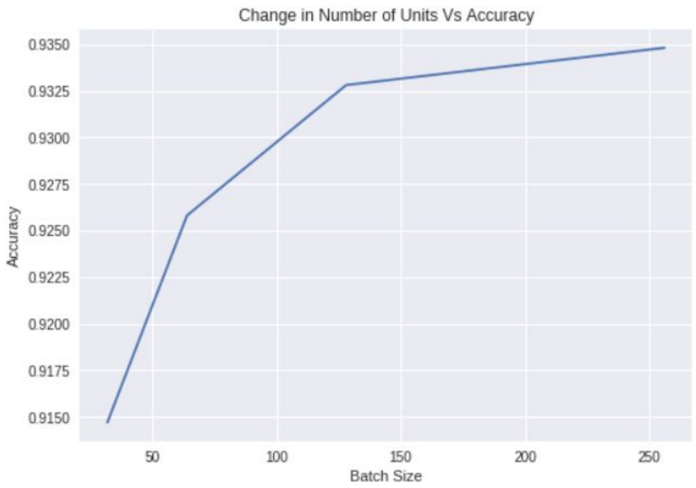
- Changing the unit size



- Changing the epochs



- Changing the batch size



The best accuracy is obtained when batch size = 256, epochs = 90 and units = 32.

4 Random Forest

Random forest classifier creates a set of decision trees from randomly selected subset of training set. It then aggregates the votes from different decision trees to decide the final class of the test object.

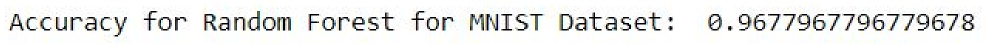
Simply, Suppose training set is given as : [X1, X2, X3, X4] with corresponding labels as [L1, L2, L3, L4], random forest may create three decision trees taking input of subset for example,

[X1, X2, X3]
[X1, X2, X4]
[X2, X3, X4]

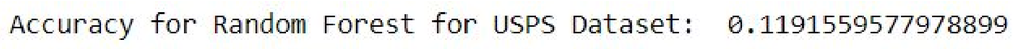
So finally, it predicts based on the majority of votes from each of the decision trees made.

4.1 Model

We create our model using sklearn RandomForestClassifier method with estimators set to 60. We train it using MNIST data and test it using MNIST and USPS data. Accuracy for both the datasets is as follows:



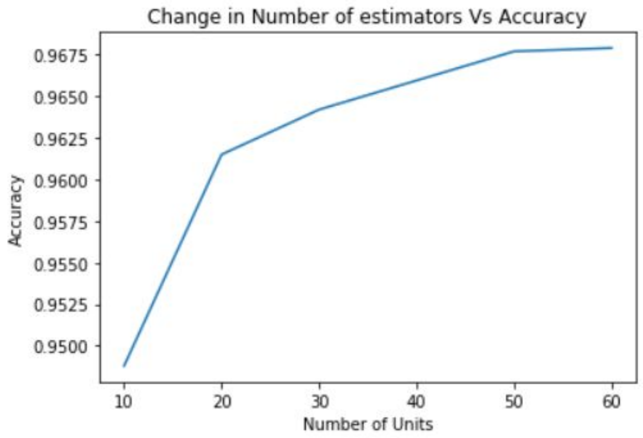
```
Accuracy for Random Forest for MNIST Dataset:  0.9677967796779678
```



```
Accuracy for Random Forest for USPS Dataset:  0.1191559577978899
```

4.2 Changing the Hyper Parameters

- Changing the estimators

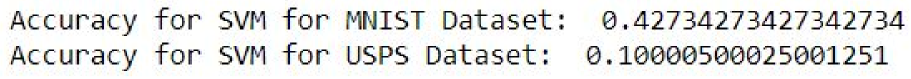


5 Support Vector Machine

A Support Vector Machine (SVM) is a discriminative classifier defined by a separating hyperplane. In other words, given labeled training data, the algorithm outputs an optimal hyperplane which categorizes new examples.

5.1 Model

We build our model using the sklearn svm class. We train our model using MNIST dataset and test using MNIST and USPS dataset. Now, as SVM takes too long to fit 60,000 training examples, we have reduced the training samples to 20,000. Here, in order to select the best model we have to take into consideration the accuracy as well as the time taken to run the model. As our final model, we have set our kernel to linear using default values.

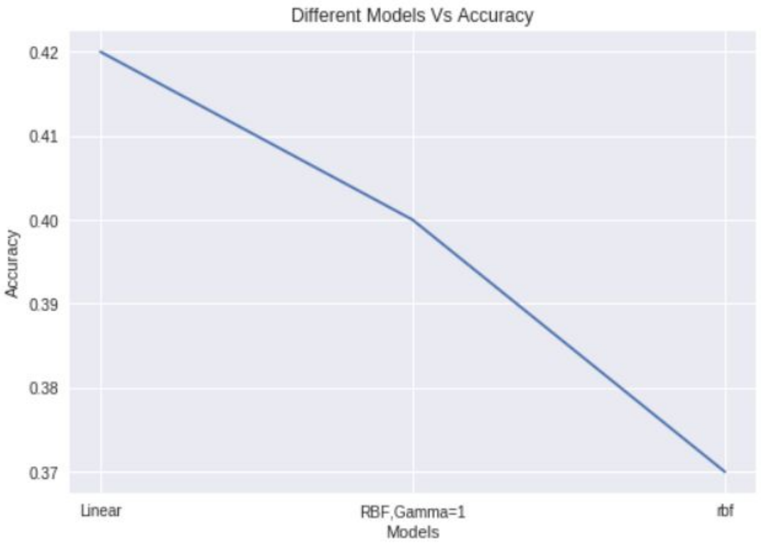


```
Accuracy for SVM for MNIST Dataset:  0.42734273427342734
Accuracy for SVM for USPS Dataset:  0.10000500025001251
```

5.2 Changing Hyper Parameters

We have tried three types of selection of model here.
First, kernel = linear and all default values
Second, kernel = rbf and gamma = 1
Third, kernel = rbf and all default values.

Below is the graph for accuracies of all these models



6 No Free Lunch Theorem

The No Free Lunch theorem states that no optimization technique is best for the generic case and all special cases. We can have a technique that is better for the generic case, but it will lose to certain techniques for specific categories. Likewise, a technique for a given category will not be comparable to some techniques for generic cases.

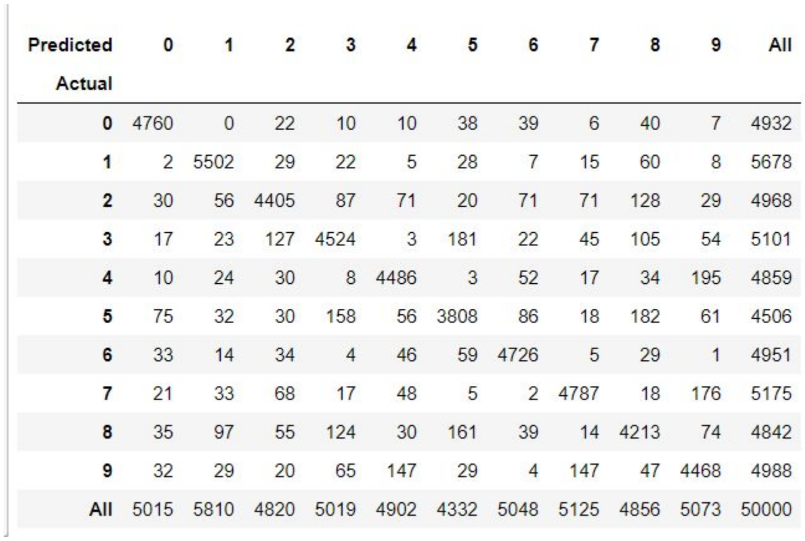
For Logistic Model, we can see that the model has 91% has accuracy for MNIST and 81% for USPS. For Neural Network, model has 93% for MNIST and 30% for USPS. For SVM, model has 42% for MNIST and 10% for USPS. For Random Forest, model has 96% and 12% for USPS.

From above results, we can see that RF performs the best on MNIST but fails to perform good on USPS. But if we observe Logistic the model performs fairly well on both the datasets with lower accuracy than RF. Same, goes for Neural Nets and SVM. Thus, No Free Lunch Theorem is proved in our problem statement. Thus, for deciding the best model for our case, we have a trade off between accuracy of MNIST and USPS.

7 Confusion Matrix Comparison

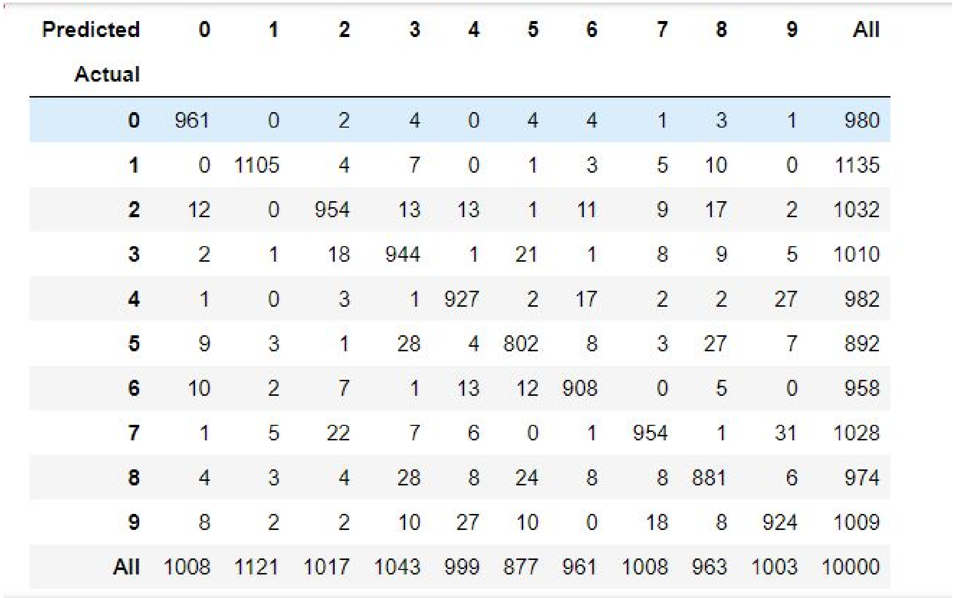
From the matrices below, the number of training examples are shown in last row and last column. Out of the total examples, the diagonal element represent the number of examples the model classified as correct. And non-diagonal matrix represent the errors in classification.

- Logistic Regression



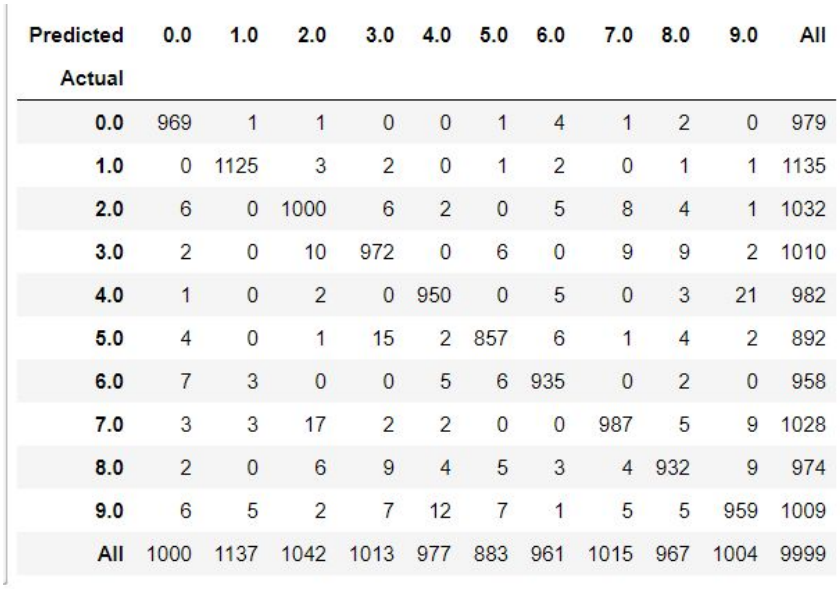
Predicted	0	1	2	3	4	5	6	7	8	9	All
Actual											
0	4760	0	22	10	10	38	39	6	40	7	4932
1	2	5502	29	22	5	28	7	15	60	8	5678
2	30	56	4405	87	71	20	71	71	128	29	4968
3	17	23	127	4524	3	181	22	45	105	54	5101
4	10	24	30	8	4486	3	52	17	34	195	4859
5	75	32	30	158	56	3808	86	18	182	61	4506
6	33	14	34	4	46	59	4726	5	29	1	4951
7	21	33	68	17	48	5	2	4787	18	176	5175
8	35	97	55	124	30	161	39	14	4213	74	4842
9	32	29	20	65	147	29	4	147	47	4468	4988
All	5015	5810	4820	5019	4902	4332	5048	5125	4856	5073	50000

- Neural Network



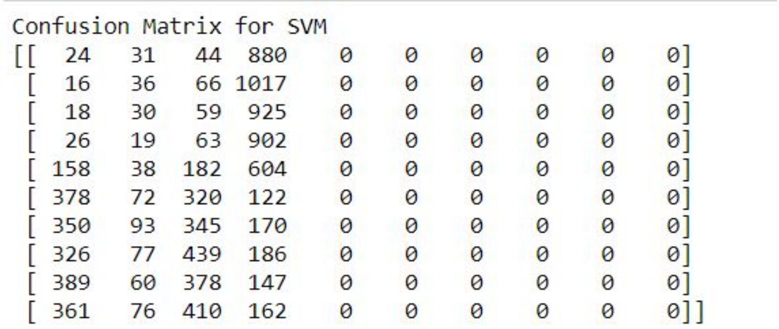
Predicted	0	1	2	3	4	5	6	7	8	9	All
Actual											
0	961	0	2	4	0	4	4	1	3	1	980
1	0	1105	4	7	0	1	3	5	10	0	1135
2	12	0	954	13	13	1	11	9	17	2	1032
3	2	1	18	944	1	21	1	8	9	5	1010
4	1	0	3	1	927	2	17	2	2	27	982
5	9	3	1	28	4	802	8	3	27	7	892
6	10	2	7	1	13	12	908	0	5	0	958
7	1	5	22	7	6	0	1	954	1	31	1028
8	4	3	4	28	8	24	8	8	881	6	974
9	8	2	2	10	27	10	0	18	8	924	1009
All	1008	1121	1017	1043	999	877	961	1008	963	1003	10000

- Random Forest

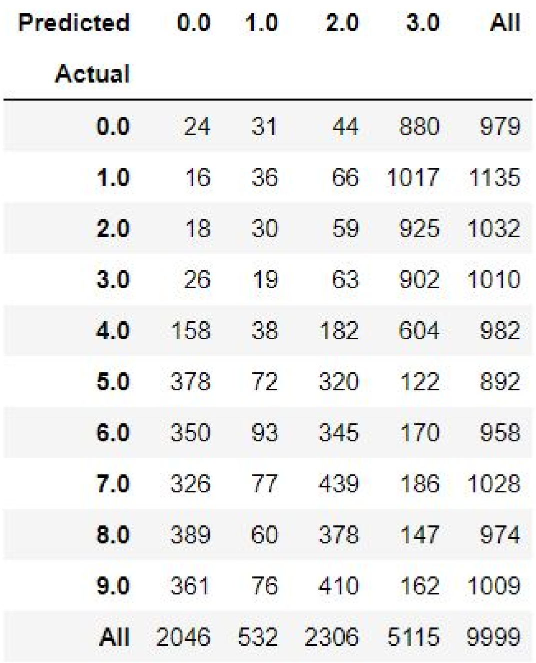


Predicted	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	All
Actual											
0.0	969	1	1	0	0	1	4	1	2	0	979
1.0	0	1125	3	2	0	1	2	0	1	1	1135
2.0	6	0	1000	6	2	0	5	8	4	1	1032
3.0	2	0	10	972	0	6	0	9	9	2	1010
4.0	1	0	2	0	950	0	5	0	3	21	982
5.0	4	0	1	15	2	857	6	1	4	2	892
6.0	7	3	0	0	5	6	935	0	2	0	958
7.0	3	3	17	2	2	0	0	987	5	9	1028
8.0	2	0	6	9	4	5	3	4	932	9	974
9.0	6	5	2	7	12	7	1	5	5	959	1009
All	1000	1137	1042	1013	977	883	961	1015	967	1004	9999

- SVM

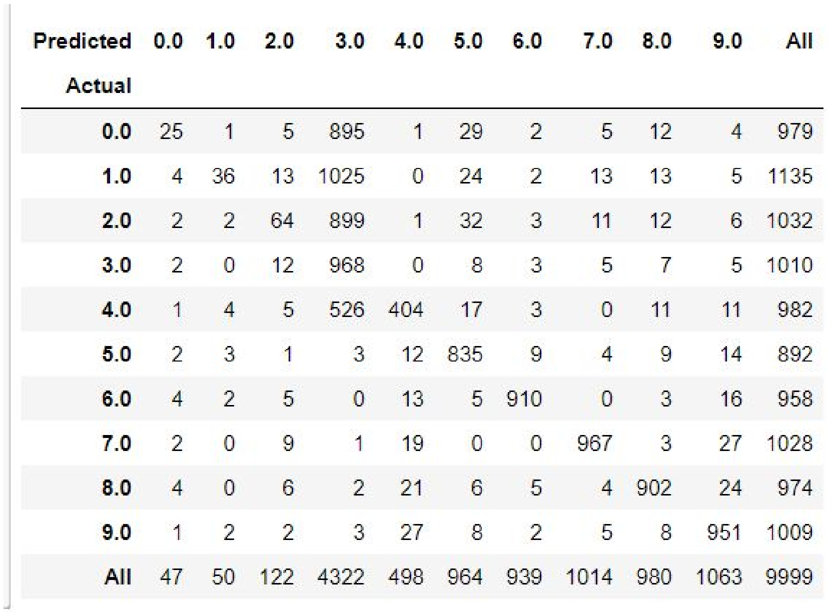


```
Confusion Matrix for SVM
[[  24   31   44  880    0    0    0    0    0    0]
 [  16   36   66 1017    0    0    0    0    0    0]
 [  18   30   59  925    0    0    0    0    0    0]
 [  26   19   63  902    0    0    0    0    0    0]
 [ 158   38  182  604    0    0    0    0    0    0]
 [ 378   72  320  122    0    0    0    0    0    0]
 [ 350   93  345  170    0    0    0    0    0    0]
 [ 326   77  439  186    0    0    0    0    0    0]
 [ 389   60  378  147    0    0    0    0    0    0]
 [ 361   76  410  162    0    0    0    0    0    0]]
```



Predicted	0.0	1.0	2.0	3.0	All
Actual					
0.0	24	31	44	880	979
1.0	16	36	66	1017	1135
2.0	18	30	59	925	1032
3.0	26	19	63	902	1010
4.0	158	38	182	604	982
5.0	378	72	320	122	892
6.0	350	93	345	170	958
7.0	326	77	439	186	1028
8.0	389	60	378	147	974
9.0	361	76	410	162	1009
All	2046	532	2306	5115	9999

- Ensemble Classifier



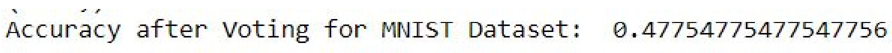
Predicted	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	All
Actual											
0.0	25	1	5	895	1	29	2	5	12	4	979
1.0	4	36	13	1025	0	24	2	13	13	5	1135
2.0	2	2	64	899	1	32	3	11	12	6	1032
3.0	2	0	12	968	0	8	3	5	7	5	1010
4.0	1	4	5	526	404	17	3	0	11	11	982
5.0	2	3	1	3	12	835	9	4	9	14	892
6.0	4	2	5	0	13	5	910	0	3	16	958
7.0	2	0	9	1	19	0	0	967	3	27	1028
8.0	4	0	6	2	21	6	5	4	902	24	974
9.0	1	2	2	3	27	8	2	5	8	951	1009
All	47	50	122	4322	498	964	939	1014	980	1063	9999

So, we can conclude that RF is the best performing model as it has maximum number of correctly classified documents.

8 Ensemble Classifier

Ensemble learning helps improve machine learning results by combining several models. Ensemble methods are meta-algorithms that combine several machine learning techniques into one predictive model in order to increase the overall performance.

In our case, to combine our 4 above models, we have used Majority Voting as a measure to obtain the final classification output. Majority rule is a decision rule that selects alternatives which have a majority, that is, more than half the votes. So, while implementing our model for majority voting, we compare the predicted target values by all individual model and output the value which is mode of our data.



```
Accuracy after Voting for MNIST Dataset:  0.47754775477547756
```

As we observe, the accuracy for Majority Voting is lesser than the individual model for Neural Network, Logistic regression and Random Forest. But it is higher when compared to the SVM model.

9 Conclusion

Thus, we have developed 4 individual models for handwritten digit recognition: Logistic Regression, SVM, Random Forest and Neural Network. We then have successfully combined all these models into one single Ensemble Classifier using Majority Voting.