CS584 – MACHINE LEARNING

FALL 2016

Handwritten Digits Recognition

Group Members: Vinod Rao ,

Ajay Ramesh

Table of Contents

Task....2
Dataset....2
  Data source....2
  Target variable....2
  Features....2
  Data size....2
Preprocessing....2
Visualization....2
  Target....2
  Features....3
Evaluation....4
  Performance Measure....4
  Classifiers....4
  Evaluation Strategy....4
  Performance Results....5
  Top Features....5
  Discussion....5
Interesting/Unexpected Results....5
Contributions of Each Group Member....6
Conclusion....6
References....6

Handwritten Digits Recognition

Group Members: Vinod Rao (A20369838), Ajay Ramesh (A20384062)

Task

To classify handwritten black & white images of digits from 0 to 9 by implementing a supervised machine learning method that uses a set of images already transformed into vector of pixels (0-255) each with a specified label of particular digit to get trained itself. This is very much useful to scan correct zip code written of postal letters to classify which state they belong to and many such other applications.

Dataset

Training dataset has 42,000 rows and 785 columns (last column for the label of image).Each row represents a vector of 784 pixels considering each image is of 28 pixels in height and 28 pixels in width.There are two additional sets of test dataset , one is having 28,000 rows other has 10,000 rows.

Data source

We downloaded preprocessed training and two test sets from MNSIT web portal.Neither we collected any additional data nor we manually labelled any data.

Target variable

The target variable has been denoted as ‘label’ whose values are from ‘0’ through ‘9’.

Features

Each image is a matrix of size 28 pixels X 28 pixels.Each pixel column denotes a feature of the image and its named as pixelY, where Y is an integer from 0 to 783.To locate this pixel on the image , whereas , Y = i*28 + j while i,j are integers from 0 to 27, that lies on the co-ordinate (i,j) of 28 x 28 matrix.Each pixel indicates intensity of lightness or darkness , which varies from 0 to 255 (higher value means darker).Hence total features are 784.

Data size

Training dataset has 42,000 instances, whereas, general test and random prediction test sets have 28,000 and 10,000 instances respectively.

Preprocessing

We did not do any preprocessing.

Visualization

Target

Frequencies of each class :

Digit 0 (Class) has 4132 records

Digit 1 (Class) has 4684 records

Digit 2 (Class) has 4177 records

Digit 3 (Class) has 4351 records

Digit 4 (Class) has 4072 records

Digit 5 (Class) has 3795 records

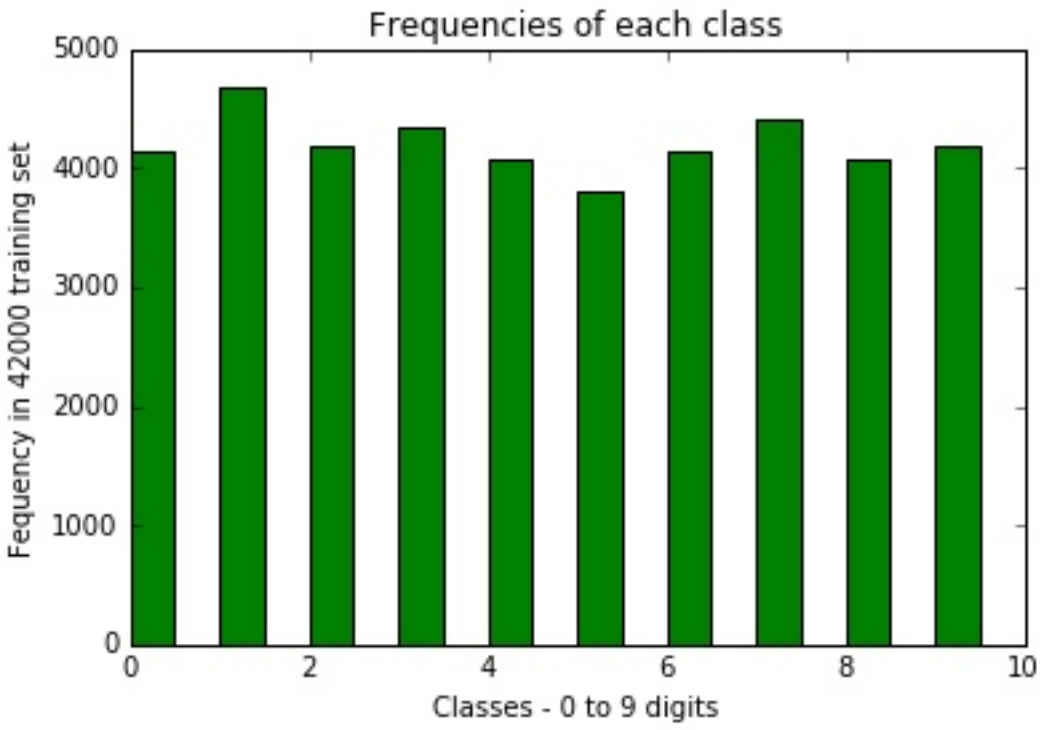
Digit 6 (Class) has 4137 records

Digit 7 (Class) has 4401 records

Digit 8 (Class) has 4063 records

Digit 9 (Class) has 4188 records>

Histogram of the target variable :



Features

Since the data is taken from MNSIT, where they already preprocessed raw (black & white) images of digits considering each of them as a matrix of size 28 pixels X 28 pixels.Each pixel column denotes a feature of the image and its named as pixelY, where Y is an integer from 0 to 783.To locate this pixel on the image , whereas , Y = i*28 + j while i,j are integers from 0

to 27, that lies on the co-ordinate (i,j) of 28 x 28 matrix.Each pixel indicates intensity of lightness or darkness , which varies from 0 to 255 (higher value means darker).

Evaluation

Performance Measure

We used accuracy, precision, recall ,confusion matrix and F1 scoring as performance measures to select best model.We used these measures because for a classification problem accuracy prediction ratio alone is not enough to make decision.Moreover, precision,recall are used to know positive predictive value and true positive rate respectively.We performed 10-fold cross validation to get accuracy on unseen data.

Classifiers

We used two models to classify handwritten digits :

- MLP Classifier :

Parameters like Activation Function , number of hidden layers , number of units in each layer have been tested with different values.Based on the perceptron theory, the learning in neural network depends on the activation function used by neurons and their back propagation process across different hidden layers with various number of units.Each setting will result into different performance metrics.Hence we tried different settings to get to know the setting (Activation function = 'relu', number of hidden layers = 10 , number of units = 100) which resulted the best performance.

Activation function used :

relu

tanh

logistic

identity

Hidden layers used:

1 hidden layer with 100 units

3 hidden layer with 100 units in each layer

10 hidden layer with 100 units in each layer

10 hidden layer with 10 units in each layer

10 hidden layer with 50 units in each layer

- Random Forest Classifier :

No settings were changed apart from the values of number of estimators=100 and number of jobs=2.

Evaluation Strategy

We performed 10-fold cross validation to get accuracy on unseen data.

Performance Results

Model	Parameters	Performance
Baseline	Majority class	Accuracy
MLP Classifer	HLU=(10,50), AF=relu	95.98%
	HLU=(3,100), AF=logistic	93.66%
	HLU=(10,100), AF=tanh	91.61%
	HLU=(1,100), AF=relu	94.44%
	HLU=(10,100), AF=relu	96.66%
Random Forest Classifier	n_estimators=100,n-jobs=2	98.88%
	majority class is digit 1 in both cases	…
Note: HLU = Hidden Layer Unit (# of layers, # of units) , AF = Activation Function		

Top Features

It's difficult to find important features for the given input vector of the pixel values of the images of various digits.

Discussion

Random Forest Classifier is best suited model among the experimented models to classify handwritten images of the 10 digits.We have got accuracy result by calling cross validation score function for RandomForestClassifier as '0.984' .

We had expected MLP Classifier to perform well, because it's based on artificial neural network.But it seemed not to be the best one.We have got accuracy results by calling cross validation score function as well for MLPClassifier on various inputs of the parameters (# of hidden layers , # of units in each of them, Activation Functions) .Which could be listed as :

0.944 , 0.966, 0.959, 0.920, 0.946, 0.907, 0.891, 0.920, 0.889, 0.919, 0.913, 0.900, 0.783, 0.929, 0.942, 0.108, 0.109, 0.112, 0.937

We concluded that there was tradeoff among the parameters (# of hidden layers , # of units in each of them, Activation Functions), and we could figure out only best accuracy up to 0.966.A sequence of experiments need to be carried out by selecting a number of combinations of these parameters to finalize best settings.

Interesting/Unexpected Results

We have tried third model 'SVM', but waited endlessly for the training process to finish.So, we did not proceed with this model.While using MLP Classifier with activation function as 'relu' ,as number of hidden layers (each with 100 units) was increased beyond 10 ,then the accuracy was reduced all along.Similar case was observed when number of hidden layers was decreased below 3.

Contributions of Each Group Member

Contributions by Vinod Rao :

1) Random Forest Classifier has been written including evaluation and prediction statistics.

2) Documentation has been done across the Jupyter notebooks.

3) Models have been compared based on their performances and selected the best one.

Contributions by Ajay Ramesh :

1) Collected training and test datasets from MNIST portal.

2) MLPClassifier has been written including evaluation and prediction statistics.

Conclusion

We learnt the classification process starting from the input image of digit till it's label was known.

We did this by converting black & white image of digits into the vector of its pixels stored in a matrix of 28 x 28 format.We are excited to continue this for colored image by considering it's pixels in terms of (R,G,V) values. We could try 'neuralnet' and convolutional neural network libraries to classify handwritten digit images to judge the best model.

References

http://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html -> (MLP Classifier)

http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html -> (Random Forest Classifier)

http://yann.lecun.com/exdb/mnist/ -> (MNSIT)

https://drive.google.com/drive/folders/0B6tUGc-vSrq0bkFuZ1ZPUV9yeWs -> (For Datasets used in this project)