Machine Learning for Image Processing

Programming Assignment 4

Name: Mohammad Al Fahim K **Roll No.:**EE17B021

1. **Aim:**
To train and test a Feedforward Neural Network for MNIST Digit Classification. The model is to be built
a) Using library functions (Keras is used here)
b) From scratch using gradient descent

2. **Brief description:**

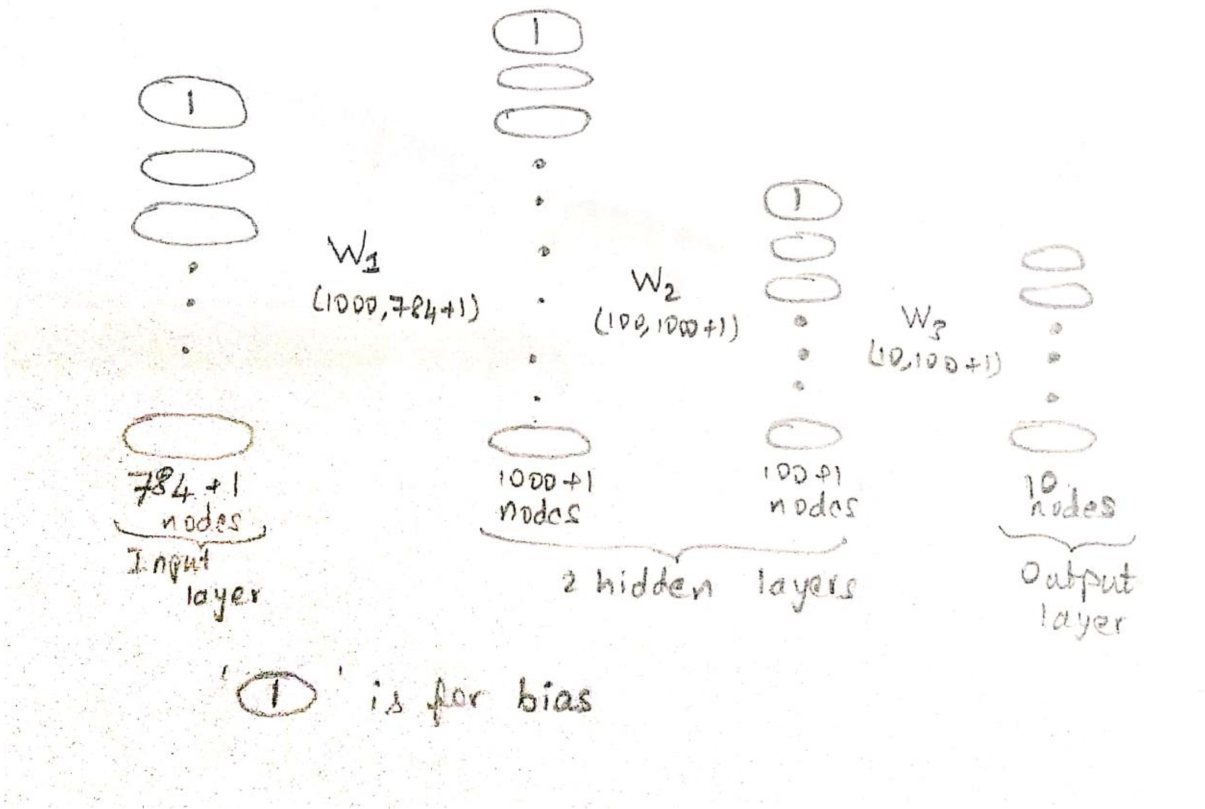

Fig. Model Architecture

The activation functions used are 'tanh' for the hidden layers and 'sigmoid' for the output layer. If 'm' represents the number of samples, the input dimensions are (785,m) and output dimensions are (10,m).

3. **Procedure and Results:**
→ The MNIST dataset is imported, split into training and test data, bias ones are added to the input features x, and the y labels are one hot encoded.
a) Using Keras:
→ Keras and other required libraries are imported
→ The model is designed following the diagram
→ The stochastic gradient descent optimizer and categorical cross entropy loss function are used to compile the model

→ The model is trained, for 10 epochs with a batch size of 100, tested and the accuracies achieved are

Training accuracy: 95.69%

Testing accuracy: 96.58%

b) From scratch:

→ The required libraries are imported

→ Functions for training, which includes forward and back-propagation, and testing, which includes only the forward propagation using the trained weights, are defined.

→ While back-propagating, the difference between the predicted and true labels are taken and back-propagated to find the weight gradients

$$\Delta W(l) = \ \delta(l+1).* \, g'\big(\Delta Net(l)\big) * x(l).T$$

where, g' = activation function's gradient

l = layer

T = transpose

δ = error

*ΔNet = W*x*

→ Using a stochastic gradient descent optimizer and mean square error as the cost function, the model is trained, for 200 epochs, tested and the accuracies achieved are

Training accuracy: 91.87%

Testing accuracy: 92.38%

	0	1	2	3	4	5	6	7	8	9
0	472	0	1	2	1	1	4	2	0	0
1	0	545	5	2	0	4	0	1	4	0
2	7	8	429	5	6	2	5	5	14	3
3	1	4	16	439	0	17	1	7	3	6
4	0	4	7	0	506	0	7	0	1	16
5	10	2	3	19	4	375	10	1	8	4
6	2	2	1	0	3	6	481	0	4	0
7	6	7	4	0	13	0	0	501	1	10
8	2	9	6	10	2	6	4	0	422	3
9	8	2	0	5	10	3	2	15	2	449

Fig. Confusion matrix of test data; rows→ true labels, columns→ predicted labels

→ The accuracy achieved from five-fold cross validation training and testing is 90.25%