Assignment 1: To explore deep learning
Course: MM811

1. Description of dataset

- The wine dataset that is used for this experiment is downloaded from an open source machine learning database website http://archive.ics.uci.edu/ml/machine-learning-databases/wine/ .
- The dataset was downloaded in folder present in virtual box by using wget command in Ubuntu terminal.
- The wine.data contain 13 attributes. The first attribute is a class identifier (0, 1 ,2) i.e. the 3 different wine cultivators.(Note :I have modified the class values from 1,2,3 to 0,1,2)
- The remaining 12 attributes are names of ingredients used to make wine i.e.
  - Alcohol
  - Malic acid
  - Ash
  - Alkalinity of ash
  - Magnesium
  - Total phenols
  - Flavanoids
  - Nonflavanoid phenols
  - Proanthocyanins
  - Color intensity
  - Hue
  - OD280/OD315 of diluted wines
  - Proline

2. Description of Problem

Our task is to train the neural network to learn the structure of data so that it can predicts the label y given the input x.

3. Description of Input

- The database has 178 different instances. The input & output data is spilt into 80 percent training set, validation set and remaining 20 percent into test set.
- In wine.data the first column is output (i.e. class label) and remaining columns are inputs (i.e. attributes)
- 3 different classification architectures are used each varying in number or neurons and layers. The layers are always integers if there is a string specified then the string name is a layer type and that type of layer will be created.
- By default the layers is of type relu i.e. rectified linear units.

  - First architecture: inputs = 12, layer = 1, neurons = (6), outputs = 3. The architecture contains 1 hidden layer consisting of 6 neurons of type rectified linear units (relu).

- Second architecture: inputs = 12, layers = 2, neurons = (6, 3), outputs = 3. The architecture contains 2 hidden layers consisting of 6 neurons of type sigmoid and 3 neurons of type relu.
- Third architecture: inputs = 12, layers = 3, neurons = (4, 4, 4), outputs = 3. The architecture contains 3 hidden layer consisting of 4 neurons each of type tanh.

4. Description of Output

- For the first type training the algorithm layer wise gives accuracy of 0.827 and training the algorithm on rprop i.e. resilient back propagation gives accuracy of 0.965.
- For the first type training the algorithm layer wise gives accuracy of 0.620 and training the algorithm on rprop i.e. resilient back propagation gives accuracy of 0.758.
- For the first type training the algorithm layer wise gives accuracy of 0.931 and training the algorithm on rprop i.e. resilient back propagation gives accuracy of 0.931.
- Every time we try to run the same algorithm its accuracy changes.

5. Description of Performance

Architecture	**Tp**	**Tn**	**Fp**	**Fn**
Type 1	**12**	**9**	**0**	**7**
Type 2	**11**	**13**	**1**	**3**
Type 3	**11**	**18**	**3**	**0**

6. Improvement

As we can see in the performance table section 4 and 5 that the results generated are not as good as expected. Datamining is an iterative process and the output generated is dependent on number of factors such as input type, output type, number of layers, neuron functions, type of data used etc. If we could fine tune some parameters like weights of input, biases, number of layers and what neuron function to be used depending on type of data, better results can be obtained.

7. Summary

The experiment helped me to understand the basics of neural network. I explored different deep learning architectures as well got some knowledge of Linux environment and python programming. This exercise was very helpful in terms of understanding how to import external dataset, set up the deep learning network and train, validate and test the network on different dataset.

8. References

1. http://theanets.readthedocs.org/en/v0.5.1/creating.html
2. http://archive.ics.uci.edu/ml/machine-learning-databases/wine/
3. https://github.com/abramhindle/theanets-tutoria