Neural Networks: Backpropagation Implementation Report

Chun-Chan (Bill) Cheng
Texas A&M University
College Station, TX
aznchat@tamu.edu

1. INTRODUCTION

The purpose of this project is to implement backpropagation algorithm described in the textbook and evaluate the confidence interval of the UCI data using the ten fold cross validation.

2. THE PROGRAM

2.1 Files Turned In

```
backPropAlgorithm.py
Case/
  data.txt
  structure.txt
  Different_Test_Case/
      test cases....
```

2.2 How to Compile the Program?

Inside the zip file there is a `backPropAlgorithm.py`. In order to run the script, run `python backPropAlgorithm.py -number_of_hidden_nodes -test_file_to run.` (`e.g python backPropAlgorithm.py 3 balance.txt`). More information can be seen by typing `python backPropAlgorithm.py -help`. Not that all test files are located in `Case/`.

Below is an example of a `structure.txt`:

```
buying,maint,doors,persons,safety,class
vhigh,high,med,low
vhigh,high,med,low
2,3,4,5more
2,4,more
low,med,high
unacc,acc,good,vgood
```

In the first line of the `structure.txt` is the name of the different attributes, the line always ends with a the element class. This hierarchy is to distinguish different attributes and the classifier. The first element of the first element maps to the second line of the `structure.txt`, in other words, `vhigh,high,med,low` is the attribute values of `buying`.

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

2.3 High Level Overview of Program

The program starts with reading in the structure file and data set file. While reading the data set files, **when it finds a missing attributes it randomly selects a value and inserts it into the missing cell**. The randomly selection of attribute values to insert was chosen is due to the fact that randomly selecting them is similar is putting weight on different attributes according to their numbers in the data set.

Next, it loops through all the data in the data set, calculating which attribute should be the root. After that it recursively calculates which attribute should be the leaf while recording the attribute value to the node. The program will stop that recursion when no more attributes can be selected and move on the the next branch.

In the last step the program prints out the tree and calculates the mean and standard deviation.

2.4 Pseudo Code

Data: this text
Result: how to write algorithm with LaTeX2e
initialization;
while *not at end of this document* **do**
  read current;
  if *understand* **then**
    go to next section;
    current section becomes this one;
  else
    go back to the beginning of current section;
  end
end

Algorithm 1: How to write algorithms

2.5 Stopping Criteria, Momentum, Training Rate

In this program, the stopping criteria is it stops whenever the loop exceeds 10000 epochs or when the error is less than 0.01.

2.6 Normalization of Input and Outputs

As you can see in Section 3.6 Table 1, the Balance Scale data set is highly inaccurate. This may be due to small number of data set (625) causing the learning algorithm to decrease in accuracy. Also note that in this program, the pruning algorithm might not have been implemented correctly leading to low accuracy of all the data sets.

3. DATA SETS

3.1 Car Evaluation

This data set is composed of six attributes: buying price, maintenance price, number of doors, number of persons that can fit, size of trunk (lug_boot in database), and the safety of the car. The purpose of this data set is to evaluate if a car is good or not. The classifier are as follows: unacc, acc, good, v-good.

3.2 Balance Scale

This data set is composed of four attributes: left-weight, left-distance, right-weight, and right-distance.This data set was generated to model psychological experimental results. Each example is classified as having the balance scale tip to the right, tip to the left, or be balanced. The correct way to find the class is the greater of (left-distance * left-weight) and (right-distance * right-weight). If they are equal, it is balanced.

3.3 Congressional Voting

This data set is composed of sixteen attributes: handicapped-infants, water-project-cost-sharing, adoption of the budget resolution, physician feefreeze, el salvador aid, religious groups in schools, anti satellite test ban, aid to nicaraguan contras, mx missile, immigration, synfuels corporation cutback, education spending, superfund right to sue, crime, duty free exports, export administration act south africa.

Also, this data set includes votes for each of the U.S. House of Representatives Congressmen on the 16 key votes identified by the CQA. The CQA lists nine different types of votes: voted for, paired for, and announced for (these three simplified to yea), voted against, paired against, and announced against (these three simplified to nay), voted present, voted present to avoid conflict of interest, and did not vote or otherwise make a position known (these three simplified to an unknown disposition).

3.4 Nursery

This data set is composed of eight attributes: parents, has nurse, form, children, housing, finance, social, health. Nursery Database was derived from a hierarchical decision model originally developed to rank applications for nursery schools. It was used during several years in 1980's when there was excessive enrollment to these schools in Ljubljana, Slovenia, and the rejected applications frequently needed an objective explanation.

3.5 Hayes-Roth

This data set is composed of four attributes: hobby, age, educational level, marital status. This data set was from *Hayes-Roth, B., & Hayes-Roth, F. (1977). Concept learning and the recognition and classification of exemplars. Journal of Verbal Learning and Verbal Behavior, 16, 321-338.* This database borrowed the concepts of psychologists that were used in their laboratory experiments that aim to investigate human categorization in natural domain.

3.6 Statistics

4. PRINTING TREE

See file printed_tree.txt for whole tree.

Data Set	μ	σ	95% C.I.
Car Evalutaion	0.60	0.23	(0.45, 0.75)
Balance Scale	0.07	0.43	(-0.27, 0.33)
Congressional Voting	0.43	0.34	(0.22, 0.64)
Nursery	0.31	0.33	(0.10, 0.51)
Hayes-Roth	0.64	0.29	(0.46, 0.82)

Table 1: Statistics on Decision Tree Data Set

Data Set	μ	σ	95% C.I.
Car Evalutaion	0.89	0.023	(0.88, 0.92)
Balance Scale	0.93	0.03	(0.92, 0.96)
Congressional Voting	0.99	0.01	(0.99, 1.002)
Nursery	0.91	0.01	(0.90, 0.92)
Hayes-Roth	0.79	0.17	(0.67, 0.92)

Table 2: Statistics on Neural Network Data Set

```
safety []
    class = unacc [low, unacc]
    persons [med]
        class = unacc [med, 2, unacc]
        maint [med, 4]
            buying [med, 4, vhigh]

Tree continues to grow ....
```

If it is the last leaf on the tree, there will be class = sign indicating which classifier it represents. The elements in the brackets are the attribute values that it takes in to get to the node. For example, in the second line `class = [low, unacc]`, this represents `safety = low` which leads to the class `unacc`.

5. CONCLUSION

In this project we have implemented a decision tree using the ID3 algorithm and tested the decision tree accuracy with UCI repository data sets. Even though the implementation was not perfect due to the fact that the pruning was not fully implemented.

Out of the five data sets, the car evaluation data sets was the best data set that fitted the decision tree and the balanced scale data set was the worst fit for the decision tree implementation.