UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

COS 314: Artificial Intelligence

Project 2: Neural Networks

Maree, Armand

120 178 00

Department of Computer Science, University of Pretoria

May 8, 2016

Contents

1 Code and Compilation

The Neural Network (NN) was coded in Java[1] and is located in the subdirectory called "NeuralNetwork". Each of the experiments is located in this directory as "Experiment1", "Experiment2" and "Experiment3". In order to compile and run a project simply run the command "make all" in the terminal of a Linux machine. This will compile, run and clean the directory.

2 Stopping Conditions

The NN is stopped as soon as the generalisation accuracy reaches 99% or when the NN starts over fitting. An epoch maximum was specified at 1000 for experiment 1 and 2. This was to allow the NN to to reach the best possible values it can, but if one has to wait ages for the result then it is useless. Experiment 3, however, was allowed to run until 3000 epochs to allow it more time since it is more complex.

3 Experiment 1

3.1 Usage

Once the NN has started executing (see section 1) the user will be prompted (in the terminal) to type a specific letter. This will be the letter that the NN attempts to learn the pattern for. Statistics will be printed to the screen after each epoch[2]. Similar statistics will be written to the a file called "NNStats.txt".

3.2 Results

Table 1 shows the average results over 30 sessions for each of the 16 test cases that was run on the NN for the letter 'A'. The columns are described as follows: ϵ (epoch of solution), HU[3] (number of hidden units), η (learning rate) , α (momentum), TE[4] (training error), GE (generalisation error).

As seen from table 1, 15 hidden units (HU), a learning rate (η) of 0.3 and a momentum (α) of 0.3 are the optimal values for this NN. This gives the fastest result and also the best accuracy.

In figure 1 the change in the accuracy can be seen as the epochs increase.

[1] Oracle Java was used to test the program and no guarantee can be made for any Java distributions.

[2] Processing the training set once.

[3] Number of neurons in the single hidden layer.

[4] Percentage incorrect classifications in the training set.

ϵ	HU	η	α	TE (%)	GE (%)
7.2	5	0.1	0.1	1.1	0.9
3.5	5	0.2	0.2	1.1	0.9
2.3	5	0.3	0.3	1.1	0.8
1.8	5	0.4	0.4	1.1	1.0
6.5	10	0.1	0.1	1.1	0.9
3.1	10	0.2	0.2	1.1	0.9
2.1	10	0.3	0.3	1.1	0.8
1.5	10	0.4	0.4	1.1	0.9
6.8	15	0.1	0.1	1.1	0.9
3.3	15	0.2	0.2	1.1	0.9
1.9	15	0.3	0.3	1.1	0.8
1.3	15	0.4	0.4	1.2	0.9
6.6	20	0.1	0.1	1.1	0.9
3.2	20	0.2	0.2	1.1	0.9
2.1	20	0.3	0.3	1.1	0.9
1.5	20	0.4	0.4	1.1	0.9

Table 1: Experiment 1 test results

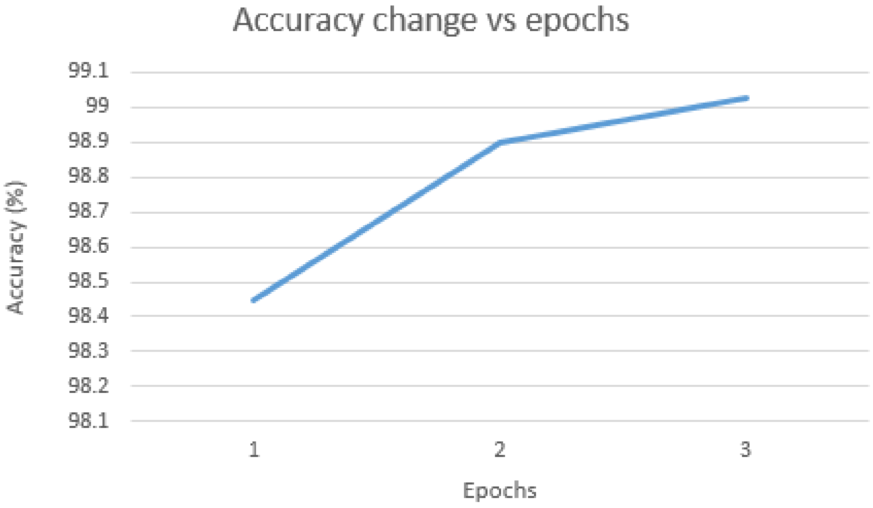

Figure 1: Experiment 1 accuracy over epoch.

4 Experiment 2

4.1 Usage

Once the program has started executing (see section 1) the NN will immediately start learning to identify the all vowels. Statistics will be printed to the screen after each epoch. Similar statistics will be written to the a file called "NNStats.txt".

4.2 Results

Table 2 shows the average results over 30 sessions for each of the 16 test cases that was run on the NN for all vowels. The columns are described as follows: ϵ (epoch of solution), HU (number of hidden units), η (learning rate) , α (momentum), TE (training error), GE (generalisation error).

ϵ	HU	η	α	TE (%)	GE (%)
85.2	5	0.1	0.1	22.3	25.0
59.6	5	0.2	0.2	14.2	16.7
41.0	5	0.3	0.3	13.9	16.5
19.5	5	0.4	0.4	15.4	18.8
258.0	10	0.1	0.1	11.2	13.2
138.2	10	0.2	0.2	8.7	10.2
73.9	10	0.3	0.3	9.3	11.5
44.2	10	0.4	0.4	10.4	13.8
454.1	15	0.1	0.1	6.1	7.2
187.8	15	0.2	0.2	6.5	8.0
76.9	15	0.3	0.3	8.1	10.9
42.6	15	0.4	0.4	10.0	13.9
440.0	20	0.1	0.1	5.1	7.0
252.5	20	0.2	0.2	5.2	7.4
102.6	20	0.3	0.3	7.0	10.1
42.7	20	0.4	0.4	9.2	13.3

Table 2: Experiment 2 test results

As seen from the table 2, 20 hidden units (HU), a learning rate (η) of 0.2 and a momentum (α) of 0.2 are the optimal values for this NN. It provides the second best results (the difference between first place and second place is nearly neglectable) in a reasonable amount of time.

In figure 2 the change of the incorrect classifications can be seen as the epochs increase.

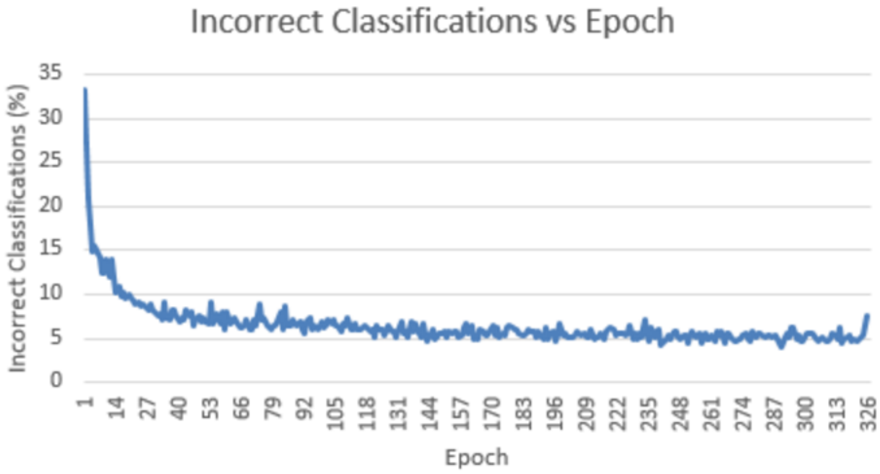

Figure 2: Experiment 2 incorrect classifications over epoch.

5 Experiment 3

5.1 Usage

Once the program has started executing (see section 1) the NN will immediately start learning to identify the all alphabet characters. Statistics will be printed to the screen after each epoch. Similar statistics will be written to the a file called "NNStats.txt".

5.2 Results

Table 3 shows the average results over 30 sessions for each of the 16 test cases that was run on the NN for all vowels. The columns are described as follows: ϵ (epoch of solution), HU (number of hidden units), η (learning rate) , α (momentum), TE (training error), GE (generalisation error).

Due to the bad nature of these results I have decided to run another few tests to see if a better result could be obtained. These results are listed in table 4.

As seen from the table 4, 40 hidden units (HU), a learning rate (η) of 0.1 and a momentum (α) of 0.1 are the optimal values for this NN if accuracy is the priority. If speed is the priority, a learning rate and momentum of 0.3 should be applied to 40 hidden units.

ϵ	HU	η	α	TE (%)	GE (%)
538.2	5	0.1	0.1	66.3	66.7
204.4	5	0.2	0.2	68.1	69.4
108.3	5	0.3	0.3	67.5	69.3
66.8	5	0.4	0.4	68.5	70.0
460.9	10	0.1	0.1	41.0	41.5
213.8	10	0.2	0.2	41.0	42.3
110.7	10	0.3	0.3	41.5	43.1
71.0	10	0.4	0.4	42.4	44.3
689.9	15	0.1	0.1	31.9	32.4
240.9	15	0.2	0.2	33.4	34.8
150.0	15	0.3	0.3	33.5	34.8
79.2	15	0.4	0.4	35.0	37.0
661.2	20	0.1	0.1	27.9	28.3
303.9	20	0.2	0.2	28.8	29.3
180.2	20	0.3	0.3	28.9	30.3
103.9	20	0.4	0.4	29.8	31.3

Table 3: Experiment 3 test results

ϵ	HU	η	α	TE (%)	GE (%)
1384.9	35	0.1	0.1	18.3	19.0
548.2	35	0.2	0.2	19.8	20.6
312	35	0.3	0.3	21.2	22.3
189.1	35	0.4	0.4	22.4	23.9
2598	40	0.1	0.1	15.1	15.5
594.3	40	0.2	0.2	18.3	18.8
431	40	0.3	0.3	19.0	19.5
318.3	40	0.4	0.4	20.2	21.2

Table 4: Experiment 3 test results

In figure 3 the change of the incorrect classifications can be seen as the epochs increase for the latter case in the previous paragraph.

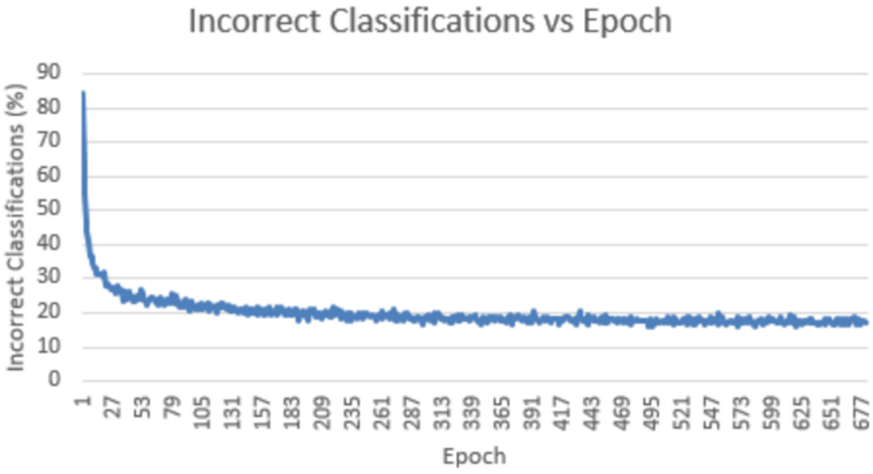

Figure 3: Experiment 3 incorrect classifications over epoch.