

Assignment2

October 19, 2017

1 INF4490 - Biologically-inspired Computing

2 Furqan Tariq (furqant)

```
In [11]: %run movements.py
import mlp
```

2.1 Multilayer Perceptron

MLP with 6 hidden nodes

```
In [12]: hidden = 6

net = mlp.mlp(train, train_targets, hidden)
net.earlystopping(train, train_targets, valid, valid_targets)
net.print_confusion(test, test_targets)
```

Confusion matrix:

```
[[ 11.  0.  0.  1.  0.  0.  0.  1.]
 [ 0.  9.  0.  0.  0.  1.  0.  0.]
 [ 0.  0. 18.  0.  0.  0.  0.  0.]
 [ 0.  1.  0. 14.  1.  0.  0.  3.]
 [ 0.  0.  0.  0. 15.  0.  0.  0.]
 [ 0.  0.  0.  0.  0. 16.  0.  0.]
 [ 0.  0.  1.  2.  0.  0.  8.  0.]
 [ 0.  0.  0.  0.  0.  0.  0.  9.]]
```

Percentage Correct: 90.0900900901

MLP with 8 hidden nodes

```
In [13]: hidden = 8

net = mlp.mlp(train, train_targets, hidden)
net.earlystopping(train, train_targets, valid, valid_targets)
net.print_confusion(test, test_targets)
```

Confusion matrix:

```
[[ 11.  0.  0.  0.  0.  0.  0.  1.]
 [ 0. 10.  0.  1.  0.  0.  0.  0.]
 [ 0.  0. 18.  0.  0.  0.  0.  0.]
 [ 0.  0.  0. 15.  1.  0.  0.  1.]
 [ 0.  0.  0.  1. 15.  1.  0.  0.]
 [ 0.  0.  0.  0.  0. 16.  0.  0.]
 [ 0.  0.  1.  0.  0.  0.  8.  0.]
 [ 0.  0.  0.  0.  0.  0.  0. 11.]]
```

Percentage Correct: 93.6936936937

MLP with 12 hidden nodes

```
In [14]: hidden = 12
```

```
net = mlp.mlp(train, train_targets, hidden)
net.earlystopping(train, train_targets, valid, valid_targets)
net.print_confusion(test, test_targets)
```

Confusion matrix:

```
[[ 11.  0.  0.  0.  0.  0.  0.  1.]
 [ 0. 10.  0.  0.  0.  1.  0.  0.]
 [ 0.  0. 19.  0.  0.  0.  0.  0.]
 [ 0.  0.  0. 16.  1.  0.  0.  2.]
 [ 0.  0.  0.  1. 15.  1.  0.  0.]
 [ 0.  0.  0.  0.  0. 14.  0.  0.]
 [ 0.  0.  0.  0.  0.  1.  8.  0.]
 [ 0.  0.  0.  0.  0.  0.  0. 10.]]
```

Percentage Correct: 92.7927927928

MLP Answers

- In my findings the minimum hidden nodes required for a network to classify are 4.
- By running algorithm several time, i found that mostly predicted mistaken classes are the neighbouring classes to actual classes in confusion matrix.

2.2 K Fold Cross-validation

```
In [15]: hidden = 12
```

```
k=6
net = mlp.mlp(train, train_targets, hidden)
net.kfold(train, train_targets, valid, valid_targets, k)
```

Fold 0 Percentage Correct: 97.2972972972973

Fold 1 Percentage Correct: 100.0

Fold 2 Percentage Correct: 86.48648648648648

Fold 3 Percentage Correct: 97.2972972972973
Fold 4 Percentage Correct: 97.2972972972973
Fold 5 Percentage Correct: 97.2972972972973

Mean: 95.9459459459
Standard Dev. 4.34398682928