Artificial Neural Network (ANN) Toolbox for Scilab

Prashant Dave

(IIT Bombay)
Introduction to ToolBox

• Developed by Ryurick M.Hristev and Updated by Allan Cornett
• Can be downloaded from the website ANN ToolBox
• Works for all Scilab versions
• Works on Linux and Windows
Introduction to Neural Networks

- Mathematical or Computational models
- Inspired by aspects of biological neural networks
- Applications are diversified
  1. Industrial process control
  2. Data validation
  3. Classification
- ANN as input layer, hidden layers and output layer
- Data has to be trained
- Different Algorithms to train the data
• Algorithms Implemented
• Algorithms Implemented
  1. Momentum with or without bias, batch or on-line
• Algorithms Implemented
  1. Momentum with or without bias, batch or on-line
  2. SuperSAB with or without bias, batch or on-line
• **Algorithms Implemented**
  1. Momentum with or without bias, batch or on-line
  2. SuperSAB with or without bias, batch or on-line
  3. Standard (vanilla) with or without bias, batch or online
• Algorithms Implemented
  1. Momentum with or without bias, batch or on-line
  2. SuperSAB with or without bias, batch or on-line
  3. Standard (vanilla) with or without bias, batch or online
  4. Conjugate gradient
• Unlimited number of layers
• Unlimited number of neurons per each layer separately
• Only layered feedforward networks are supported "directly", for others use the "hooks"
Getting started

Steps to follow for loading the ToolBox in Scilab

1. Open Scilab
Steps to follow for loading the ToolBox in Scilab

1. Open Scilab
2. Change the current directory to ToolBox folder
Getting started

Steps to follow for loading the ToolBox in Scilab

1. Open Scilab
2. Change the current directory to ToolBox folder
3. exec builder.sce
Getting started

Steps to follow for loading the ToolBox in Scilab

1. Open Scilab
2. Change the current directory to ToolBox folder
3. exec builder.sce
4. exec loader.sce
Classification Example

Objective: To calculate weights using training data and test the efficiency using test data

- Three classes of plants
  1. Setosa
  2. Virginica
  3. Versicolor
- Based on 4 attributes
  1. petal width
  2. petal length
  3. sepal width
  4. sepal length
Plot of Fisher's Iris Data
....Continued

- 2 Classes will be considered i.e. class 1 and class 3
- 2 attributes will be used i.e. petal width and petal length

![Fisher Iris's Data plot for Class 1 and Class 3](image)
• Training Data: 25 data set form class 1 and class 3
• Test Data: 10 Data set from class 1 and class 3
• Scaling of the Data between 0 and 1
• Online backpropagation with Momentum with bias
plot of training data set

Class 1

Class 3

output from training data

data point number

1.0

0.8

0.6

0.4

0.2

0.0

-0.2

-0.4
Exercise

- Repeat the same example with
  1. class 1 and class 2 data set with more than one hidden layers
  2. class 2 and class 3 data set (Is desired classification achieved?)
Conclusion

- Only one activation function
- Sometimes need to run twice as it gives error at the first place (WHY !!!)
- Still a Very effective toolbox
- Provides a range of Algorithms
Fisher Discriminant Analysis (FDA)

- Generalized code is written in Scilab
- Training data and Test data are required
- Gives FDA vectors and their weights
- Gives Class number to which the test data belongs
Projected Data on FDA vectors
Class No of Test Data

plot of Test Data

Class 1
Class 2
Class 3
Thank You.