# ANN & R Programming

Jaishankar Hebballi(13030142017)

# Overview

## Character Recognition

Many Optical Character recognition softwares (OCR) exist. Their basic functionality is to be able to parse and guess the characters that may be handwritten or typed. But my scope is not as big. In this project a very basic character recognition is implemented. Two learning rules are implemented to guess the correct output. Two kinds of recognition is done in this project.

- Pattern Classification

## Character Recognition

Many Optical Character recognition softwares (OCR) exist. Their basic functionality is to be able to parse and guess the characters that may be handwritten or typed. But my scope is not as big. In this project a very basic character recognition is implemented. Two learning rules are implemented to guess the correct output. Two kinds of recognition is done in this project.

- Pattern Classification
- Pattern Association

# Overview

- `RSNNS`

All these packages can be installed from the repository using `install.packages(package_name)`

- `RSNNS`
- `pixmap`

All these packages can be installed from the repository using `install.packages(package_name)`

- `RSNNS`
- `pixmap`
- `gWidgets`

All these packages can be installed from the repository using `install.packages(package_name)`

- `RSNNS`
- `pixmap`
- `gWidgets`
- `gWidgets2`

All these packages can be installed from the repository using `install.packages(package_name)`

- `RSNNS`
- `pixmap`
- `gWidgets`
- `gWidgets2`
- `gWidgetsRGtk`

All these packages can be installed from the repository using `install.packages(package_name)`

- `RSNNS`
- `pixmap`
- `gWidgets`
- `gWidgets2`
- `gWidgetsRGtk`
- `gWidgets2RGtk2`

All these packages can be installed from the repository using `install.packages(package_name)`

- `RSNNS`
- `pixmap`
- `gWidgets`
- `gWidgets2`
- `gWidgetsRGtk`
- `gWidgets2RGtk2`
- `stringr`

All these packages can be installed from the repository using `install.packages(package_name)`

- `RSNNS`
- `pixmap`
- `gWidgets`
- `gWidgets2`
- `gWidgetsRGtk`
- `gWidgets2RGtk2`
- `stringr`
- `cairo`

All these packages can be installed from the repository using `install.packages(package_name)`

# Overview

## My Goal

Very Basic Pattern Recognition

- Accept images from the user

## My Goal

Very Basic Pattern Recognition

- Accept images from the user
- Train the network with a few known input images.

## My Goal

Very Basic Pattern Recognition

- Accept images from the user
- Train the network with a few known input images.
- Make the Network guess the unkown input.

# Overview

## Encode Input

- From the user get an image of 9*9 pixels.

## Encode Input

- From the user get an image of 9*9 pixels.
- The Image can contain any alphabet in Capital letters

## Encode Input

- From the user get an image of 9*9 pixels.
- The Image can contain any alphabet in Capital letters
- Transform the input into a matrix of bi-polar numbers which will be fed into the network.

# Overview

## Decode Output

Pattern Classification

- The network will output a sequence of binary numbers

Pattern Association

## Decode Output

Pattern Classification

- The network will output a sequence of binary numbers
- Decode this number into an ASCII character

Pattern Association

## Decode Output

Pattern Classification

- The network will output a sequence of binary numbers
- Decode this number into an ASCII character
- Display the correct character to the user

Pattern Association

## Decode Output

Pattern Classification

- The network will output a sequence of binary numbers
- Decode this number into an ASCII character
- Display the correct character to the user

Pattern Association

- The network will output a sequence of binary numbers

## Decode Output

Pattern Classification

- The network will output a sequence of binary numbers
- Decode this number into an ASCII character
- Display the correct character to the user

Pattern Association

- The network will output a sequence of binary numbers
- Decode this number into an pixmap image

## Decode Output

Pattern Classification

- The network will output a sequence of binary numbers
- Decode this number into an ASCII character
- Display the correct character to the user

Pattern Association

- The network will output a sequence of binary numbers
- Decode this number into an pixmap image
- Plot or draw this image on the window

# Overview

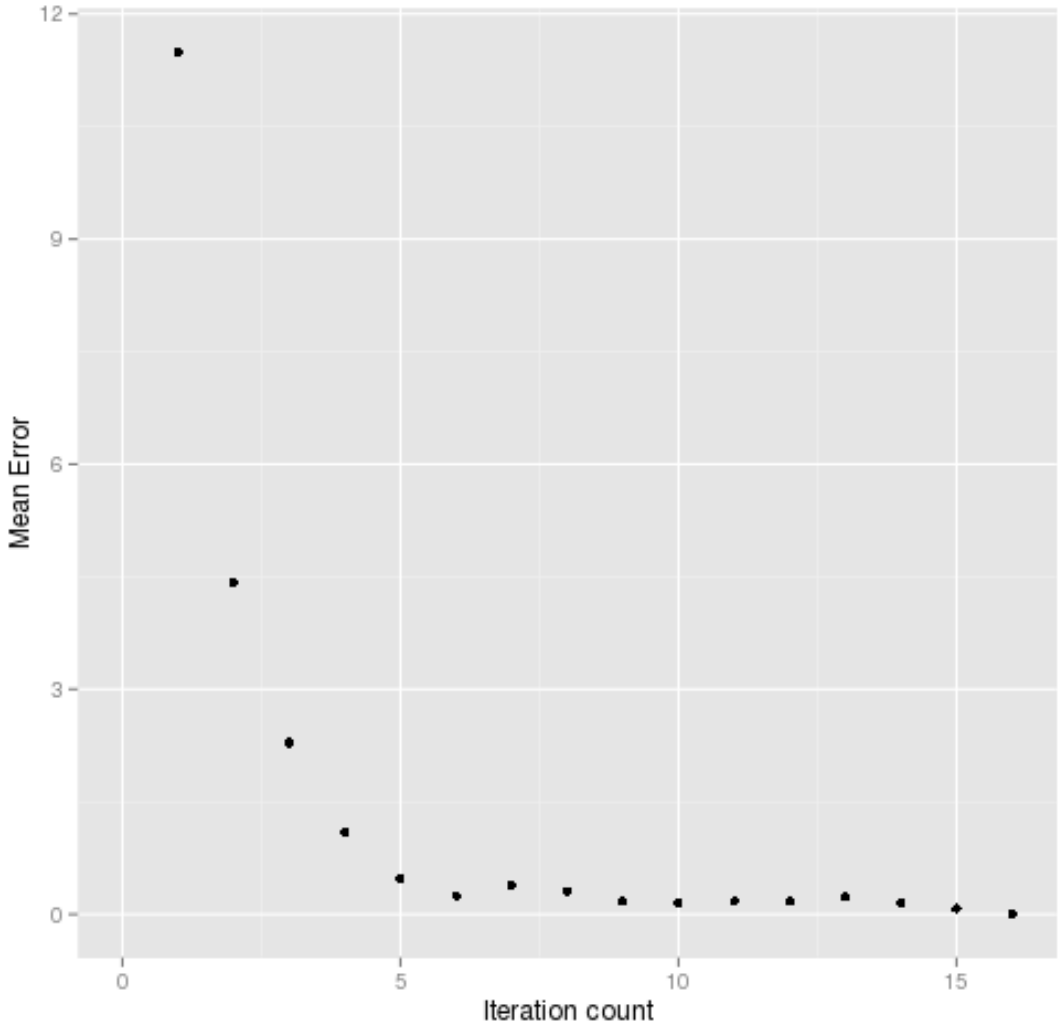
- Hebbian Learning Rule
- Perceptron Rule
- Delta Learning Rule
- Backpropagation

# Overview

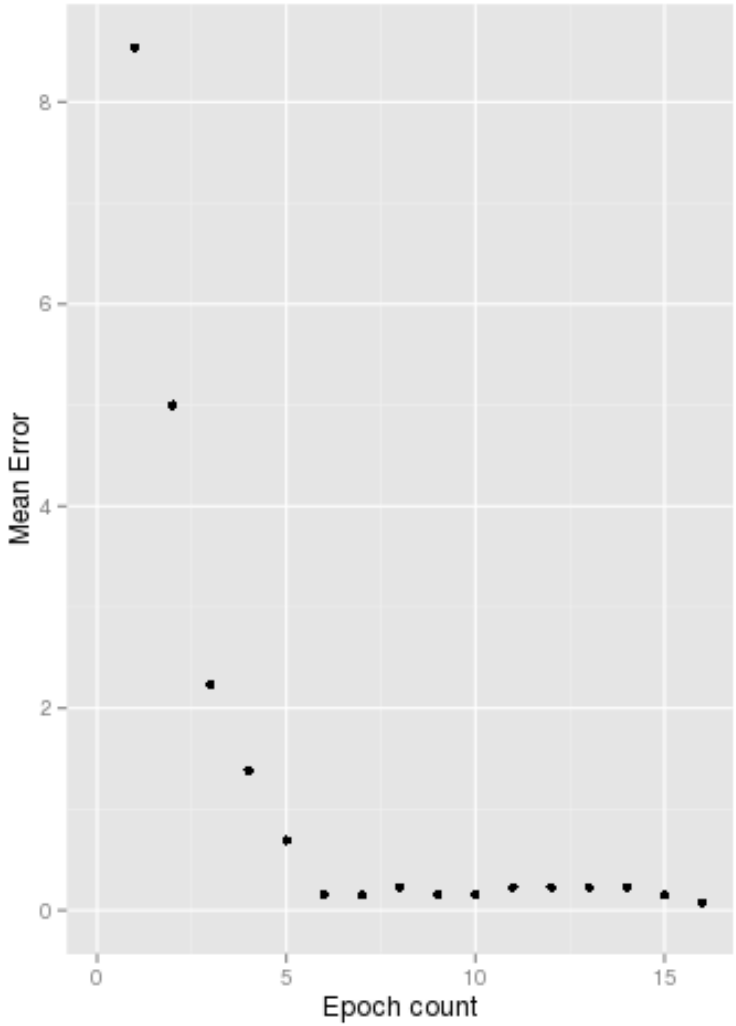
**Error Graphs** Mean Error against epoch times. The graphs tell us how the learning has taken place.

- MLP Backpropagation (classification)
- My Perceptron Implementation
- Delta Learning
- MLP Backpropagation (association)

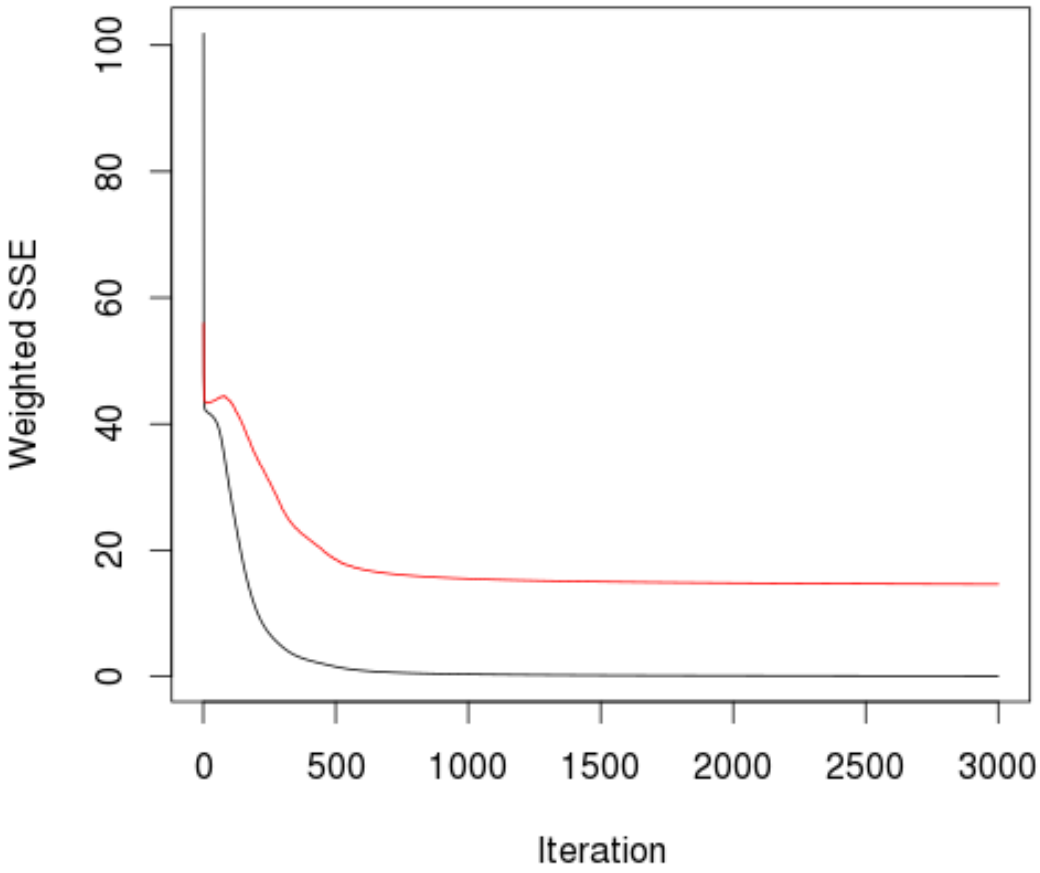
# My Perceptron Error Reduction Curve



# Delta Rule Error Reduction Curve

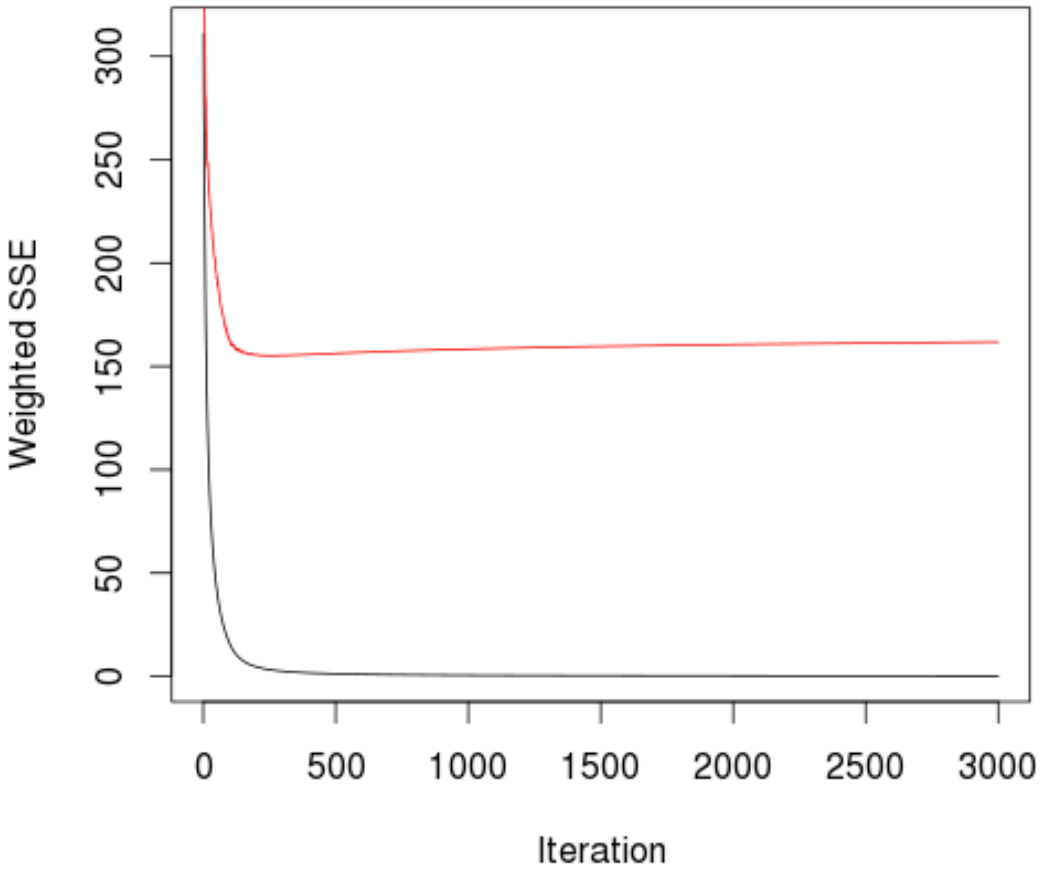


# Multi-Layer Perceptron (Classification) Error Reduction Curve




RSNNS Package network

# Multi-Layer Perceptron (Association) Error Reduction Curve




RSNNS Package network

# Overview

Performance Times

- MLP Backpropagation (classification) (6.650 sec)
- My Perceptron Implementation (3.155 sec)
- Delta Learning (2.817 sec)
- Hebbian Rule (association) (0.023 sec)
- MLP Backpropagation (association) (8.685 sec)

# Overview

# References



FundamentalsofNeuralNetworks(LaureneFausett)


NeuralNetworksinRUsingtheStuttgartNeuralNetworkSimulator: RSNNS


Packagesmanualpages


IntroductiontoNeuralNetworksusingMATLAB ...

# Thank you

Thank You!