

Package: deep (via r-universe)

August 20, 2024

Type Package

Title A Neural Networks Framework

Version 0.1.0

Author Brian Lee Mayer

Maintainer Brian <bleemayer@gmail.com>

Description This package provides a layer oriented way of creating neural networks, the framework is intended to give the user total control of the internals of a net without much effort. Use classes like PerceptronLayer to create a layer of perceptron neurons, and specify how many you want. The package does all the tricky stuff internally leaving you focused in what you want. I wrote this package during a neural networks course to help me with the problem set.

License GPL-3 + file LICENSE

Encoding UTF-8

LazyData true

Imports methods

RoxygenNote 6.1.1

Repository <https://blmayer.r-universe.dev>

RemoteUrl <https://github.com/blmayer/deep>

RemoteRef HEAD

RemoteSha 25c618c7e430b147ddae0fddcb9500d9f2a94c76

Contents

deep	2
McCullochPitts-class	2
McCullochPittsLayer-class	3
NeuralNetwork-class	3
Perceptron-class	4
PerceptronLayer-class	4

Index	6
--------------	----------

deep *deep: A Neural Networks Framework*

Description

The deep package provides classes for layers, types of neurons and the neural network as a whole.

McCullochPitts-class *The McCullochPitts neuron class, that implements the logic of the McCullochPitts neuron model.*

Description

The McCullochPitts neuron class, that implements the logic of the McCullochPitts neuron model.

Arguments

inputs	The actual data to be fed to the neuron, this input's dimensions vary with the chosen weights dimensions.
ins	The list of vectors of inputs to the first layer in the network
outs	The list of vectors of outputs of the last layer in the network
epochs	How many rounds of training to run
tax	This is the learning rate, aka eta
maxErr	A condition to early stop the training process

Value

The computed value using the McCullochPitts model.

Vector of computed values of the same size of the last layer

Fields

ws The matrix of weights that multiply the input vector, it can be a vector, a matrix or an array.

bias The bias value.

 McCullochPittsLayer-class

The McCullochPittsLayer class, that implements a layer of McCullochPitts neurons.

Description

The McCullochPittsLayer class, that implements a layer of McCullochPitts neurons.

Arguments

input	The actual data to be fed to the layer, this input's dimention's vary with the chosen n.
ins	The list of vectors of inputs to the first layer in the network
outs	The list of vectors of outputs of the last layer in the network
epochs	How many rounds of training to run
tax	This is the learning rate, aka eta
maxErr	A contition to early stop the training process

Value

The computed value using the McCullochPittsLayer model.

Vector of computed values of the same size of the last layer

Fields

n	The number of neurons to create in the layer
dims	A vector of dimensions of the inputs to the layer
neurons	A list with the internal neurons

 NeuralNetwork-class *The main NeuralNetwork class, that holds the layers.*

Description

The main NeuralNetwork class, that holds the layers.

Fields

eta	The learning tax, representes the size of the weight adjustment between each epoch of training.
layers	This field is a list of the layers of the network, you can use subsetting to inspect them.

Perceptron-class	<i>The Perceptron neuron class, that implements the logic of the perceptron model.</i>
------------------	--

Description

The Perceptron neuron class, that implements the logic of the perceptron model.

Arguments

inputs	The actual data to be fed to the neuron, this input's dimensions vary with the chosen weights dimensions.
ins	The list of vectors of inputs to the first layer in the network
outs	The list of vectors of outputs of the last layer in the network
epochs	How many rounds of training to run
tax	This is the learning rate, aka eta
maxErr	A condition to early stop the training process

Value

The computed value using the Perceptron model.

Vector of computed values of the same size of the last layer

Fields

ws	The matrix of weights that multiply the input vector, it can be a vector, a matrix or an array.
bias	The bias value.

PerceptronLayer-class	<i>The PerceptronLayer class, that implements a layer of Perceptron neurons.</i>
-----------------------	--

Description

The PerceptronLayer class, that implements a layer of Perceptron neurons.

Arguments

input	The actual data to be fed to the layer, this input's dimensions vary with the chosen n.
ins	The list of vectors of inputs to the first layer in the network
outs	The list of vectors of outputs of the last layer in the network
epochs	How many rounds of training to run
tax	This is the learning rate, aka eta
maxErr	A condition to early stop the training process

Value

The computed value using the Perceptron model.

Vector of computed values of the same size of the last layer

Fields

n The number of neurons to create in the layer

dims A vector of dimensions of the inputs to the layer

neurons A list with the internal neurons

Index

deep, [2](#)
deep-package (deep), [2](#)

McCullochPitts (McCullochPitts-class), [2](#)
McCullochPitts-class, [2](#)
McCullochPittsLayer
 (McCullochPittsLayer-class), [3](#)
McCullochPittsLayer-class, [3](#)

neuralNet (NeuralNetwork-class), [3](#)
NeuralNetwork-class, [3](#)

perceptron (Perceptron-class), [4](#)
Perceptron-class, [4](#)
perceptronLayer
 (PerceptronLayer-class), [4](#)
PerceptronLayer-class, [4](#)