My Project

Generated by Doxygen 1.8.6

Thu Mar 10 2016 07:43:39

Contents

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

Class Index

2.1 Class List

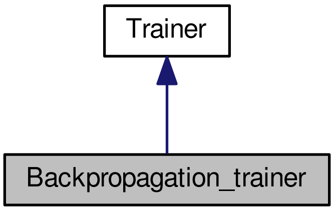
Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

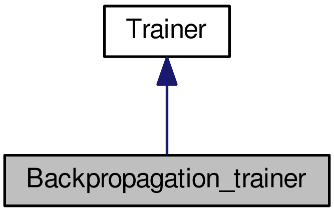
Class Documentation

3.1 Backpropagation_trainer Class Reference

Inheritance diagram for Backpropagation_trainer:



Collaboration diagram for Backpropagation_trainer:



Public Member Functions

- void **train** (Data_set data_set, NeuralNet &net)
- void **train** (Data_set data_set, NeuralNet &net, mat &results_cost_and_score_evolution)

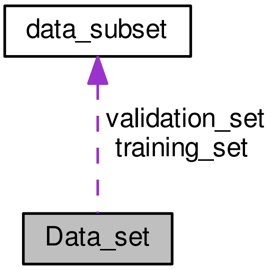
Additional Inherited Members

The documentation for this class was generated from the following files:

- backpropagation_trainer.h
- backpropagation_trainer.cpp

3.2 Data_set Class Reference

Collaboration diagram for Data_set:



Public Member Functions

- **Data_set** (unsigned int data_set_index)
- **Data_set** (string full_path)
- void **select_data_set** (unsigned int chosen_data_set_index, string &data_set_filename, string &octave_variable_name_performances_VS_nb_epochs, string &octave_variable_name_cost_training_set_size, string &octave_variable_name_cost_validation_set_size, string &octave_variable_name_scores_pop_size, string &result_filename)
- void **set_data_set** (unsigned int chosen_data_set_index, string &data_set_filename)
- string **get_data_set_info** (mat data_set)
- void **subdivide_data_cross_validation** (unsigned int index_validation_fold, unsigned int nb_folds)

Public Attributes

- mat **data**
- data_subset **training_set**
- data_subset **validation_set**

The documentation for this class was generated from the following files:

- data_set.h
- data_set.cpp

3.3 data_subset Struct Reference

Public Attributes

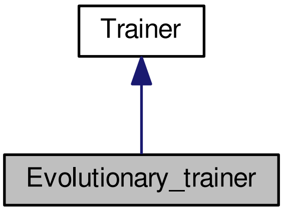
- mat **X**
- mat **Y**

The documentation for this struct was generated from the following file:

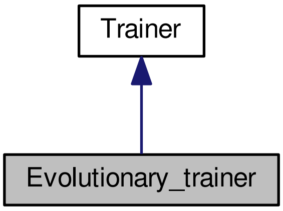
- data_set.h

3.4 Evolutionary_trainer Class Reference

Inheritance diagram for Evolutionary_trainer:



Collaboration diagram for Evolutionary_trainer:



Public Member Functions

- void **train** (Data_set data_set, NeuralNet &net)
- void **train** (Data_set data_set, NeuralNet &net, mat &results_score_evolution)
- void **train_weights** (data_subset training_set, data_subset validation_set, NeuralNet &net, unsigned int nb_epochs, mat &results_score_evolution)
- NeuralNet **train_topology_plus_weights** (Data_set data_set, net_topology max_topo, mat &results_score_evolution)

- NeuralNet **cross_validation_training** (Data_set data_set, net_topology min_topo, net_topology max_topo, mat &results_score_evolution, double &avrg_score)
- NeuralNet evolve_through_generations (data_subset training_set, net_topology min_topo, net_topology max_topo, unsigned int nb_epochs, mat &results_cost_and_score_evolution, unsigned int index_cross_-validation_section)
- void **generate_random_population** (unsigned int pop_size, net_topology max_topo)
- NeuralNet **get_best_model** (vector< NeuralNet > pop)
- NeuralNet **get_best_model** (vector< vec > genome_pop)
- vec **get_genome** (NeuralNet n, net_topology largest_topology)
- NeuralNet **generate_net** (vec genome)
- unsigned int **get_genome_length** (net_topology t)
- unsigned int **get_population_size** ()
- mat **get_population_scores** (data_subset d)
- vector< NeuralNet > **get_population** ()
- void **set_population** (vector< NeuralNet > pop)
- void **insert_individual** (NeuralNet indiv)

Additional Inherited Members

3.4.1 Member Function Documentation

3.4.1.1 NeuralNet Evolutionary_trainer::evolve_through_generations (data_subset *training_set,* net_topology *min_topo,* net_topology *max_topo,* unsigned int *nb_epochs,* mat & *results_cost_and_score_evolution,* unsigned int *index_cross_validation_section*)

RUNNING: Differential Evolution

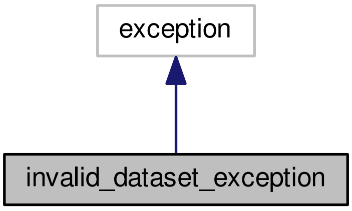
TERMINATION CRITERION: If all generations were achieved OR if the GA has already converged

The documentation for this class was generated from the following files:

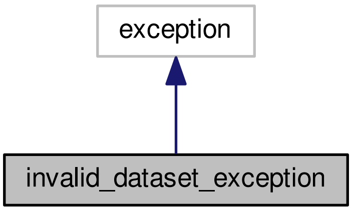
- evolutionary_trainer.h
- evolutionary_trainer.cpp

3.5 invalid_dataset_exception Class Reference

Inheritance diagram for invalid_dataset_exception:



Collaboration diagram for invalid_dataset_exception:



The documentation for this class was generated from the following file:

- data_set.h

3.6 Net_benchmark Class Reference

Public Member Functions

- void **run_benchmark** (unsigned int nb_rep)
- void **train_topology** (NeuralNet &net)
- void **set_topology** (net_topology t)
- void **compute_perfs_test_validation** (double &model_score_training_set, double &model_prediction_accuracy_training_set, double &model_score_validation_set, double &model_prediction_accuracy_validation_set)

The documentation for this class was generated from the following files:

- net_benchmark.h
- net_benchmark.cpp

3.7 net_topology Struct Reference

Public Member Functions

- unsigned int **get_total_nb_weights** ()
- string **to_string** ()

Public Attributes

- unsigned int **nb_input_units**
- unsigned int **nb_units_per_hidden_layer**
- unsigned int **nb_output_units**
- unsigned int **nb_hidden_layers**

The documentation for this struct was generated from the following file:

- neuralnet.h

3.8 NeuralNet Class Reference

The NeuralNet class.

`#include <neuralnet.h>`

Public Member Functions

- **NeuralNet** (net_topology t)
- mat forward_propagate (mat X)

  forward_propagate
- mat forward_propagate (mat X, vector< mat > &Zs, vector< mat > &As)

  forward_propagate
- vector< mat > reshape_weights ()

  reshape_weights
- void **save_net** (ofstream &model_file)
- unsigned int **get_total_nb_weights** ()
- vec **generate_random_model** ()
- void **print_topology** ()
- vec **get_params** ()
- void **set_params** (vec p)
- net_topology **get_topology** ()
- void **set_topology** (net_topology t)
- double get_accuracy (data_subset d)

  get_accuracy
- double get_f1_score (data_subset d)

  get_f1_score
- double get_matthews_correlation_coefficient (data_subset d)

  get_matthews_correlation_coefficient
- void **print_topology** (net_topology t)
- bool operator< (NeuralNet &n)

  operator < (comparator-function for sorting by highest score)

3.8.1 Detailed Description

The NeuralNet class.

3.8.2 Member Function Documentation

3.8.2.1 mat NeuralNet::forward_propagate (mat *X*)

forward_propagate

Parameters

X	Input data as matrix, whether it contains a single row or several. (Must fit the number of input neurons)

Returns

The predictions made by the net on the input data X

3.8.2.2 mat NeuralNet::forward_propagate (mat *X*, vector< mat > & *Zs*, vector< mat > & *As*)

forward_propagate

Parameters

X	Input data as matrix, whether it contains a single row or several. (Must fit the number of input neurons)
Zs	vector of matrices for the summed weights∗inputs(not yet been through sigmoid) to be returned by reference
As	vector of matrices for the activations (outputs) of the neurons to be returned by reference

Returns

The predictions made by the net on the input data X Also returns (by reference) the updated state of the vectors Z and A

3.8.2.3 double NeuralNet::get_accuracy (data_subset *d*)

get_accuracy

Parameters

d	data portion used for accuracy calculation

Returns

returns percentage representing how often the model correctly predicts <Y> on the data-set <X>

3.8.2.4 double NeuralNet::get_f1_score (data_subset *d*)

get_f1_score

Parameters

d	

Returns

(score function) returns an indication of the quality of the model [0, 1] The F1 Score function calculates the *precision* and *recall* of the model. This function is used as fitness function by the Differential Evolution algorithm.

3.8.2.5 double NeuralNet::get_matthews_correlation_coefficient (data_subset *d*)

get_matthews_correlation_coefficient

Parameters

d	

Returns

(score function) returns an indication of the quality of the model [-1, +1]

3.8.2.6 bool NeuralNet::operator< (NeuralNet & *n*) [inline]

operator < (comparator-function for sorting by highest score)

Parameters

n	

Returns

true if the provided net is less fit than this net

3.8.2.7 vector< mat > NeuralNet::reshape_weights ( )

reshape_weights

Returns

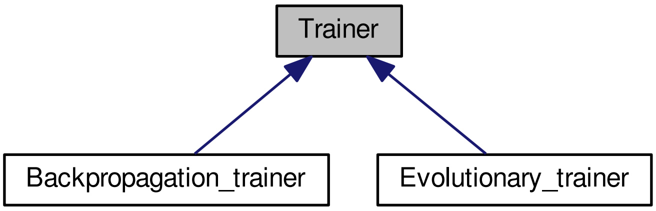
returns vector of Theta (weights) matrices of the neural network e.g. if net has 1 input layer, 1 hidden layer, 1 output layer reshape_weights() will return a vector of the two weight matrices respectively Theta[0] and Theta[1]

The documentation for this class was generated from the following files:

- neuralnet.h
- neuralnet.cpp

3.9 Trainer Class Reference

Inheritance diagram for Trainer:



Public Member Functions

- virtual void **train** (Data_set data_set, NeuralNet &net)=0
- virtual void **train** (Data_set data_set, NeuralNet &net, mat &results_cost_and_score_evolution)=0
- unsigned int **get_nb_epochs** ()
- void **set_nb_epochs** (unsigned int e)

Protected Member Functions

- unsigned int **generate_random_integer_between_range** (unsigned int min, unsigned int max)

Protected Attributes

- unsigned int **nb_epochs**

The documentation for this class was generated from the following files:

- trainer.h
- trainer.cpp