ForestFactory 0.1a

Generated by Doxygen 1.8.10

Sat Oct 17 2015 20:05:46

Contents

1	Clas	s Index			1
	1.1	Class I	ist		1
2	Clas	s Docui	mentation		3
	2.1	ffactory	::BaseClass	ifier Class Reference	3
		2.1.1	Detailed De	escription	3
		2.1.2	Member Fu	nction Documentation	3
			2.1.2.1 p	oredict(Sample &sample)	3
			2.1.2.2 p	redict(Dataset &d)	4
			2.1.2.3 to	est(Dataset &d)	5
			2.1.2.4 tr	rain(Dataset d)	5
			2.1.2.5 u	pdate(Sample &sample)	5
			2.1.2.6 u	pdate(Dataset &d)	5
	2.2	ffactory	::BaseErrorl	Measure Class Reference	5
		2.2.1	Detailed De	escription	6
		2.2.2	Member Fu	nction Documentation	6
			2.2.2.1 g	etError(Sample s)	6
			2.2.2.2 g	etError(Dataset d)	6
	2.3	ffactory	v::Dataloader	Class Reference	6
	2.4	ffactory	::Dataset Cl	ass Reference	7
		2.4.1	Member Fu	nction Documentation	7
			2.4.1.1 a	dd(Sample s)	7
	2.5	ffactory	::Prediction	Class Reference	7
		2.5.1	Detailed De	escription	8
	2.6	ffactory	::Sample Cla	ass Reference	8
		2.6.1	Detailed De	escription	8
		2.6.2	Member Fu	nction Documentation	8
			2.6.2.1 ft	ullRandom(DataType minv, DataType maxv, unsigned int seed)	8
			2.6.2.2 ft	ullWith(DataType v)	9
			2.6.2.3 rd	esize(unsigned int size)	9
		263	Friends And	d Related Function Documentation	9

iv		CONTE	NTS
	2.6.3.1	operator<<	9
Index			11

Chapter 1

Class Index

1.1 Class List

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

ffactory::BaseClassifier	Ç
ffactory::BaseErrorMeasure	Ę
ffactory::Dataloader	6
ffactory::Dataset	7
ffactory::Prediction	7
ffactory::Sample	8

2 Class Index

Chapter 2

Class Documentation

2.1 ffactory::BaseClassifier Class Reference

#include <baseClassifier.h>

Public Member Functions

- virtual double train (Dataset d)
- virtual Prediction predict (Sample &sample)
- virtual Prediction predict (Dataset &d)
- virtual double test (Dataset &d)
- virtual void update (Sample &sample)
- virtual void update (Dataset &d)
- const std::string & getName () const
- void **setName** (const std::string &name)
- int getNumClasses () const
- void setNumClasses (int numClasses)
- int getNumFeatures () const
- · void setNumFeatures (int numFeatures)
- int getNumTrainSamples () const
- void setNumTrainSamples (int numTrainSamples)

2.1.1 Detailed Description

Class with essential functions for all classification problems

2.1.2 Member Function Documentation

2.1.2.1 virtual Prediction ffactory::BaseClassifier::predict (Sample & sample) [virtual]

Predict class of one sample

Parameters

sample

Returns

Prediction

2.1.2.2 virtual Prediction ffactory::BaseClassifier::predict(Dataset & d) [virtual]

Predict class for dataset

Parameters sample Returns 2.1.2.3 virtual double ffactory::BaseClassifier::test(Dataset & d) [inline], [virtual] Test trained classifier on dataset d **Parameters** d Returns specified error measure FIXME: Error measure class must be here !// **2.1.2.4** virtual double ffactory::BaseClassifier::train (Dataset d) [virtual] Train classifier on dataset d **Parameters** d Returns Value of specified error measure on dataset d 2.1.2.5 virtual void ffactory::BaseClassifier::update (Sample & sample) [virtual] Update classifier using one new sample **Parameters** sample **2.1.2.6** virtual void ffactory::BaseClassifier::update (Dataset & d) [virtual] Update classifier using dataset **Parameters**

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

• src/classifiers/baseClassifier.h

sample

2.2 ffactory::BaseErrorMeasure Class Reference

#include <baseErrorMeasure.h>

Public Member Functions

- virtual double getError (Sample s)
- virtual double getError (Dataset d)
- const std::string & getName () const
- void setName (const std::string &name)

2.2.1 Detailed Description

Generic error measure class

2.2.2 Member Function Documentation

2.2.2.1 virtual double ffactory::BaseErrorMeasure::getError(Sample s) [virtual]

Get error on current sample s

Parameters

Returns

error measure value

2.2.2.2 virtual double ffactory::BaseErrorMeasure::getError(Dataset *d***)** [virtual]

Get error on entire dataset d

Parameters

d

Returns

error measure value

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

src/classifiers/baseErrorMeasure.h

2.3 ffactory::Dataloader Class Reference

Public Member Functions

• void load (Dataset d, const std::string &filename)

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

· src/data/dataloader.h

2.4 ffactory::Dataset Class Reference

Public Member Functions

- const DataVector & getMaxRange () const
- void setMaxRange (const DataVector &maxRange)
- const DataVector & getMinRange () const
- void setMinRange (const DataVector &minRange)
- unsigned int getNumClasses () const
- void setNumClasses (unsigned int numClasses)
- · unsigned int getNumFeatures () const
- void **setNumFeatures** (unsigned int numFeatures)
- unsigned int getNumSamples () const
- void setNumSamples (unsigned int numSamples)
- const std::vector< Sample > & getSamples () const
- void setSamples (const std::vector< Sample > &samples)
- void calculateRanges ()
- void clear ()
- void add (Sample s)
- Sample getSample (unsigned int i)
- · const std::string & getName () const
- void setName (const std::string &name)

2.4.1 Member Function Documentation

2.4.1.1 void ffactory::Dataset::add (Sample s) [inline]

Method adds new sample to dataset and recalculate ranges

Parameters

s

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

· src/data/dataset.h

2.5 ffactory::Prediction Class Reference

```
#include <prediction.h>
```

Public Member Functions

- Prediction (std::string classifName, unsigned int numFeat)
- Prediction (BaseClassifier c)
- const DataMatrix & getConfidences () const
- int getNumFeatures () const
- void setNumFeatures (int numFeatures)
- const DataVector & getPrediction () const
- DataVector **getConfidence** (unsigned int sampleIndex)
- const std::string & getClassifierName () const
- void setClassifierName (const std::string &classifierName)
- void setConfidences (const DataMatrix &confidences)

2.5.1 Detailed Description

Class represents result of prediction such as class probabilities. Obtaining of Confusion matrix, final class can be performed.

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

· src/classifiers/prediction.h

2.6 ffactory::Sample Class Reference

```
#include <sample.h>
```

Public Member Functions

- Sample (unsigned int size=0)
- · int getId () const
- · void setId (int id)
- · double getW () const
- void setW (double w)
- int getY () const
- void setY (int y)
- const DataVector & getVector () const
- void **setVector** (const DataVector &x)
- void setValue (unsigned int i, const DataType &v)
- const DataType **getValue** (unsigned int i)
- unsigned int length ()
- void fullWith (DataType v)
- void resize (unsigned int size)
- void fullRandom (DataType minv, DataType maxv, unsigned int seed)

Friends

• std::ostream & operator<< (std::ostream &stream, const Sample &s)

2.6.1 Detailed Description

Sample class supports various types of data containers including Eigen library types for storing data samples.

2.6.2 Member Function Documentation

2.6.2.1 void ffactory::Sample::fullRandom (DataType minv, DataType maxv, unsigned int seed) [inline]

Fulls all elements of sample data vector with uniformly distributed pseudo-random values with specified seed and ranges

Parameters

minv	
------	--

maxv	
seed	

2.6.2.2 void ffactory::Sample::fullWith (DataType v) [inline]

Fulls all elements of sample data vector with specified value

Parameters



2.6.2.3 void ffactory::Sample::resize (unsigned int size) [inline]

Changes size of vector. Please note than all data content will be forgotten!

Parameters

```
size
```

2.6.3 Friends And Related Function Documentation

2.6.3.1 std::ostream& operator<<(std::ostream & stream, const Sample & s) [friend]

Stream output

Parameters

stream	
matrix	

Returns

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

· src/data/sample.h

Index

```
add
     ffactory::Dataset, 7
ffactory::BaseClassifier, 3
     predict, 3
     test, 5
     train, 5
     update, 5
ffactory::BaseErrorMeasure, 5
     getError, 6
ffactory::Dataloader, 6
ffactory::Dataset, 7
     add, 7
ffactory::Prediction, 7
ffactory::Sample, 8
     fullRandom, 8
     fullWith, 9
     operator<<, 9
     resize, 9
fullRandom
     ffactory::Sample, 8
fullWith
     ffactory::Sample, 9
getError
     ffactory::BaseErrorMeasure, 6
operator<<
     ffactory::Sample, 9
predict
     ffactory::BaseClassifier, 3
resize
     ffactory::Sample, 9
test
     ffactory::BaseClassifier, 5
train
     ffactory:: Base Classifier, \, {\color{red} 5}
update
     ffactory::BaseClassifier, 5
```