JavaScript is disabled on your browser.

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Tree](http://docs.google.com/package-tree.html)
* [Index](http://docs.google.com/index-all.html)
* [Help](http://docs.google.com/help-doc.html)
* [Prev Class](http://docs.google.com/org/opencv/ml/RTrees.html)
* [Next Class](http://docs.google.com/org/opencv/ml/SVM.html)
* [Frames](http://docs.google.com/index.html?org/opencv/ml/StatModel.html)
* [No Frames](http://docs.google.com/StatModel.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* [Field](#3znysh7) |
* Constr |
* [Method](#2et92p0)
* Detail:
* [Field](#1t3h5sf) |
* Constr |
* [Method](#26in1rg)

org.opencv.ml

## Class StatModel

* java.lang.Object
  + [org.opencv.core.Algorithm](http://docs.google.com/org/opencv/core/Algorithm.html)
    - org.opencv.ml.StatModel
* Direct Known Subclasses: [ANN\_MLP](http://docs.google.com/org/opencv/ml/ANN_MLP.html), [DTrees](http://docs.google.com/org/opencv/ml/DTrees.html), [EM](http://docs.google.com/org/opencv/ml/EM.html), [KNearest](http://docs.google.com/org/opencv/ml/KNearest.html), [LogisticRegression](http://docs.google.com/org/opencv/ml/LogisticRegression.html), [NormalBayesClassifier](http://docs.google.com/org/opencv/ml/NormalBayesClassifier.html), [SVM](http://docs.google.com/org/opencv/ml/SVM.html), [SVMSGD](http://docs.google.com/org/opencv/ml/SVMSGD.html)  
    
  public class StatModel  
  extends [Algorithm](http://docs.google.com/org/opencv/core/Algorithm.html)  
  Base class for statistical models in OpenCV ML.

### Field SummaryFields

| Modifier and Type | Field and Description |
| --- | --- |
| static int | [**COMPRESSED\_INPUT**](http://docs.google.com/org/opencv/ml/StatModel.html#COMPRESSED_INPUT) |
| static int | [**PREPROCESSED\_INPUT**](http://docs.google.com/org/opencv/ml/StatModel.html#PREPROCESSED_INPUT) |
| static int | [**RAW\_OUTPUT**](http://docs.google.com/org/opencv/ml/StatModel.html#RAW_OUTPUT) |
| static int | [**UPDATE\_MODEL**](http://docs.google.com/org/opencv/ml/StatModel.html#UPDATE_MODEL) |

### Method SummaryMethods

| Modifier and Type | Method and Description |
| --- | --- |
| static [StatModel](http://docs.google.com/org/opencv/ml/StatModel.html) | [**\_\_fromPtr\_\_**](http://docs.google.com/org/opencv/ml/StatModel.html#__fromPtr__(long))(long addr) |
| float | [**calcError**](http://docs.google.com/org/opencv/ml/StatModel.html#calcError(org.opencv.ml.TrainData,%20boolean,%20org.opencv.core.Mat))([TrainData](http://docs.google.com/org/opencv/ml/TrainData.html) data, boolean test, [Mat](http://docs.google.com/org/opencv/core/Mat.html) resp) Computes error on the training or test dataset |
| boolean | [**empty**](http://docs.google.com/org/opencv/ml/StatModel.html#empty())() Returns true if the Algorithm is empty (e.g. |
| int | [**getVarCount**](http://docs.google.com/org/opencv/ml/StatModel.html#getVarCount())() Returns the number of variables in training samples |
| boolean | [**isClassifier**](http://docs.google.com/org/opencv/ml/StatModel.html#isClassifier())() Returns true if the model is classifier |
| boolean | [**isTrained**](http://docs.google.com/org/opencv/ml/StatModel.html#isTrained())() Returns true if the model is trained |
| float | [**predict**](http://docs.google.com/org/opencv/ml/StatModel.html#predict(org.opencv.core.Mat))([Mat](http://docs.google.com/org/opencv/core/Mat.html) samples) Predicts response(s) for the provided sample(s) |
| float | [**predict**](http://docs.google.com/org/opencv/ml/StatModel.html#predict(org.opencv.core.Mat,%20org.opencv.core.Mat))([Mat](http://docs.google.com/org/opencv/core/Mat.html) samples, [Mat](http://docs.google.com/org/opencv/core/Mat.html) results) Predicts response(s) for the provided sample(s) |
| float | [**predict**](http://docs.google.com/org/opencv/ml/StatModel.html#predict(org.opencv.core.Mat,%20org.opencv.core.Mat,%20int))([Mat](http://docs.google.com/org/opencv/core/Mat.html) samples, [Mat](http://docs.google.com/org/opencv/core/Mat.html) results, int flags) Predicts response(s) for the provided sample(s) |
| boolean | [**train**](http://docs.google.com/org/opencv/ml/StatModel.html#train(org.opencv.core.Mat,%20int,%20org.opencv.core.Mat))([Mat](http://docs.google.com/org/opencv/core/Mat.html) samples, int layout, [Mat](http://docs.google.com/org/opencv/core/Mat.html) responses) Trains the statistical model |
| boolean | [**train**](http://docs.google.com/org/opencv/ml/StatModel.html#train(org.opencv.ml.TrainData))([TrainData](http://docs.google.com/org/opencv/ml/TrainData.html) trainData) Trains the statistical model |
| boolean | [**train**](http://docs.google.com/org/opencv/ml/StatModel.html#train(org.opencv.ml.TrainData,%20int))([TrainData](http://docs.google.com/org/opencv/ml/TrainData.html) trainData, int flags) Trains the statistical model |

### Methods inherited from class org.opencv.core.[**Algorithm**](http://docs.google.com/org/opencv/core/Algorithm.html)[clear](http://docs.google.com/org/opencv/core/Algorithm.html#clear()), [getDefaultName](http://docs.google.com/org/opencv/core/Algorithm.html#getDefaultName()), [getNativeObjAddr](http://docs.google.com/org/opencv/core/Algorithm.html#getNativeObjAddr()), [save](http://docs.google.com/org/opencv/core/Algorithm.html#save(java.lang.String))

### Methods inherited from class java.lang.Objectequals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Field Detail

#### COMPRESSED\_INPUT public static final int COMPRESSED\_INPUTSee Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.ml.StatModel.COMPRESSED_INPUT)

#### PREPROCESSED\_INPUT public static final int PREPROCESSED\_INPUTSee Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.ml.StatModel.PREPROCESSED_INPUT)

#### RAW\_OUTPUT public static final int RAW\_OUTPUTSee Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.ml.StatModel.RAW_OUTPUT)

#### UPDATE\_MODEL public static final int UPDATE\_MODELSee Also:[Constant Field Values](http://docs.google.com/constant-values.html#org.opencv.ml.StatModel.UPDATE_MODEL)

### Method Detail

#### \_\_fromPtr\_\_ public static [StatModel](http://docs.google.com/org/opencv/ml/StatModel.html) \_\_fromPtr\_\_(long addr)

#### calcError public float calcError([TrainData](http://docs.google.com/org/opencv/ml/TrainData.html) data, boolean test, [Mat](http://docs.google.com/org/opencv/core/Mat.html) resp) Computes error on the training or test datasetParameters:data - the training datatest - if true, the error is computed over the test subset of the data, otherwise it's computed over the training subset of the data. Please note that if you loaded a completely different dataset to evaluate already trained classifier, you will probably want not to set the test subset at all with TrainData::setTrainTestSplitRatio and specify test=false, so that the error is computed for the whole new set. Yes, this sounds a bit confusing.resp - the optional output responses. The method uses StatModel::predict to compute the error. For regression models the error is computed as RMS, for classifiers - as a percent of missclassified samples (0%-100%). Returns:automatically generated

#### empty public boolean empty() **Description copied from class:**[**Algorithm**](http://docs.google.com/org/opencv/core/Algorithm.html#empty()) Returns true if the Algorithm is empty (e.g. in the very beginning or after unsuccessful read**Overrides:** [empty](http://docs.google.com/org/opencv/core/Algorithm.html#empty()) in class [Algorithm](http://docs.google.com/org/opencv/core/Algorithm.html) Returns:automatically generated

#### getVarCount public int getVarCount() Returns the number of variables in training samplesReturns:automatically generated

#### isClassifier public boolean isClassifier() Returns true if the model is classifierReturns:automatically generated

#### isTrained public boolean isTrained() Returns true if the model is trainedReturns:automatically generated

#### predict public float predict([Mat](http://docs.google.com/org/opencv/core/Mat.html) samples) Predicts response(s) for the provided sample(s)Parameters:samples - The input samples, floating-point matrix Returns:automatically generated

#### predict public float predict([Mat](http://docs.google.com/org/opencv/core/Mat.html) samples, [Mat](http://docs.google.com/org/opencv/core/Mat.html) results) Predicts response(s) for the provided sample(s)Parameters:samples - The input samples, floating-point matrixresults - The optional output matrix of results. Returns:automatically generated

#### predict public float predict([Mat](http://docs.google.com/org/opencv/core/Mat.html) samples, [Mat](http://docs.google.com/org/opencv/core/Mat.html) results, int flags) Predicts response(s) for the provided sample(s)Parameters:samples - The input samples, floating-point matrixresults - The optional output matrix of results.flags - The optional flags, model-dependent. See cv::ml::StatModel::Flags. Returns:automatically generated

#### train public boolean train([Mat](http://docs.google.com/org/opencv/core/Mat.html) samples, int layout, [Mat](http://docs.google.com/org/opencv/core/Mat.html) responses) Trains the statistical modelParameters:samples - training sampleslayout - See ml::SampleTypes.responses - vector of responses associated with the training samples. Returns:automatically generated

#### train public boolean train([TrainData](http://docs.google.com/org/opencv/ml/TrainData.html) trainData) Trains the statistical modelParameters:trainData - training data that can be loaded from file using TrainData::loadFromCSV or created with TrainData::create. new training samples, not completely overwritten (such as NormalBayesClassifier or ANN\_MLP). Returns:automatically generated

#### train public boolean train([TrainData](http://docs.google.com/org/opencv/ml/TrainData.html) trainData, int flags) Trains the statistical modelParameters:trainData - training data that can be loaded from file using TrainData::loadFromCSV or created with TrainData::create.flags - optional flags, depending on the model. Some of the models can be updated with the new training samples, not completely overwritten (such as NormalBayesClassifier or ANN\_MLP). Returns:automatically generated

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Tree](http://docs.google.com/package-tree.html)
* [Index](http://docs.google.com/index-all.html)
* [Help](http://docs.google.com/help-doc.html)
* [Prev Class](http://docs.google.com/org/opencv/ml/RTrees.html)
* [Next Class](http://docs.google.com/org/opencv/ml/SVM.html)
* [Frames](http://docs.google.com/index.html?org/opencv/ml/StatModel.html)
* [No Frames](http://docs.google.com/StatModel.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* [Field](#3znysh7) |
* Constr |
* [Method](#2et92p0)
* Detail:
* [Field](#1t3h5sf) |
* Constr |
* [Method](#26in1rg)

Generated on 2021-04-02 03:15:03 / OpenCV 3.4.14