

Boosting with Husky

Generated by Doxygen 1.8.12

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

husky	??
husky::mllib	??

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

husky::mllib::Classifier	??
husky::mllib::LogisticRegression	??
husky::mllib::LogitBoost	??
husky::mllib::RealAdaBoost	??
husky::mllib::DataPreprocessor	??
husky::mllib::MaxAbsScaler	??
husky::mllib::Estimator	??
husky::mllib::LinearRegression	??
husky::mllib::LinearRegression_SGD	??
husky::mllib::SimpleLinearRegression	??
husky::mllib::Instance	??
husky::mllib::Instances	??
PObject	??
husky::mllib::Prediction	??
husky::mllib::PseudoObject	??

Chapter 3

Class Index

3.1 Class List

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

husky::mllib::Classifier	??
husky::mllib::DataPreprocessor	??
husky::mllib::Estimator	??
husky::mllib::Instance	??
husky::mllib::Instances	??
husky::mllib::LinearRegression	??
husky::mllib::LinearRegression_SGD	??
husky::mllib::LogisticRegression	??
husky::mllib::LogitBoost	??
husky::mllib::MaxAbsScaler	??
PIObject	??
husky::mllib::Prediction	??
husky::mllib::PseudoObject	??
husky::mllib::RealAdaBoost	??
husky::mllib::SimpleLinearRegression	??

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Classifier.hpp	??
DataPreprocessor.hpp	??
DataReader.cpp	??
DataReader.hpp	??
Estimator.hpp	??
Instances.hpp	??
LinearRegression.cpp	??
LinearRegression.hpp	??
LinearRegression_SGD.cpp	??
LinearRegression_SGD.hpp	??
LogisticRegression.cpp	??
LogisticRegression.hpp	??
LogitBoost.cpp	??
LogitBoost.hpp	??
MaxAbsScaler.cpp	??
MaxAbsScaler.hpp	??
RealAdaBoost.cpp	??
RealAdaBoost.hpp	??
SimpleLinearRegression.cpp	??
SimpleLinearRegression.hpp	??
testcompile.cpp	??
testcompile.hpp	??
Utility.hpp	??

Chapter 5

Namespace Documentation

5.1 husky Namespace Reference

Namespaces

- [mllib](#)

5.2 husky::mllib Namespace Reference

Classes

- class [Classifier](#)
- class [DataPreprocessor](#)
- class [Estimator](#)
- class [Instance](#)
- class [Instances](#)
- class [LinearRegression](#)
- class [LinearRegression_SGD](#)
- class [LogisticRegression](#)
- class [LogitBoost](#)
- class [MaxAbsScaler](#)
- class [Prediction](#)
- class [PseudoObject](#)
- class [RealAdaBoost](#)
- class [SimpleLinearRegression](#)

Enumerations

- enum [LABEL_TYPE](#) { [LABEL_TYPE::NO_LABEL](#) =0, [LABEL_TYPE::Y](#) =1, [LABEL_TYPE::CLASS](#) =2 }
- enum [MODE](#) { [MODE::GLOBAL](#) =0, [MODE::LOCAL](#) =1 }

Functions

- void [svReader](#) ([Instances](#) &instances, std::string filepath, boost::char_separator< char > delimiter, [LABEL_TYPE](#) label_type)
- void [max_abs_vec](#) ([vec_double](#) &va, const [vec_double](#) &vb)

5.2.1 Enumeration Type Documentation

5.2.1.1 LABEL_TYPE

```
enum husky::mllib::LABEL_TYPE [strong]
```

Enumerator

NO_LABEL	
Y	
CLASS	

5.2.1.2 MODE

```
enum husky::mllib::MODE [strong]
```

Enumerator

GLOBAL	
LOCAL	

5.2.2 Function Documentation

5.2.2.1 max_abs_vec()

```
void husky::mllib::max_abs_vec (
    vec_double & va,
    const vec_double & vb )
```

5.2.2.2 svReader()

```
void husky::mllib::svReader (
    Instances & instances,
    std::string filepath,
    boost::char_separator< char > delimiter,
    LABEL_TYPE label_type )
```

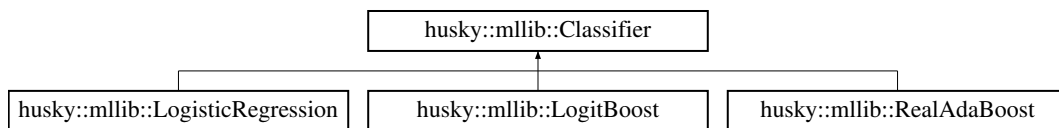
Chapter 6

Class Documentation

6.1 husky::mllib::Classifier Class Reference

```
#include <Classifier.hpp>
```

Inheritance diagram for `husky::mllib::Classifier`:



Public Member Functions

- virtual void `fit` (const [Instances](#) &instances)=0
- virtual void `fit` (const [Instances](#) &instances, std::string instance_weight_name)=0
- virtual `AttrList< Instance, Prediction > & predict` (const [Instances](#) &instances, std::string prediction_name="prediction")=0
- virtual `Classifier * clone` (int seed=0)=0

6.1.1 Member Function Documentation

6.1.1.1 clone()

```
virtual Classifier* husky::mllib::Classifier::clone (  
    int seed = 0 ) [pure virtual]
```

The clone function will only copy hyperparameters but not estimated parameter. e.g. parameters used to train model, including learning rate, etc.

Implemented in [husky::mllib::LogisticRegression](#), [husky::mllib::LogitBoost](#), and [husky::mllib::RealAdaBoost](#).

6.1.1.2 fit() [1/2]

```
virtual void husky::mllib::Classifier::fit (
    const Instances & instances ) [pure virtual]
```

Train a classifier with instances.

Implemented in [husky::mllib::LogisticRegression](#), [husky::mllib::LogitBoost](#), and [husky::mllib::RealAdaBoost](#).

6.1.1.3 fit() [2/2]

```
virtual void husky::mllib::Classifier::fit (
    const Instances & instances,
    std::string instance_weight_name ) [pure virtual]
```

Train a classifier with weighted instances.

Implemented in [husky::mllib::LogisticRegression](#), [husky::mllib::LogitBoost](#), and [husky::mllib::RealAdaBoost](#).

6.1.1.4 predict()

```
virtual AttrList<Instance, Prediction>& husky::mllib::Classifier::predict (
    const Instances & instances,
    std::string prediction_name = "prediction" ) [pure virtual]
```

Do prediction, and results are stored in a AttrList with name being prediction_name

Implemented in [husky::mllib::LogisticRegression](#), [husky::mllib::LogitBoost](#), and [husky::mllib::RealAdaBoost](#).

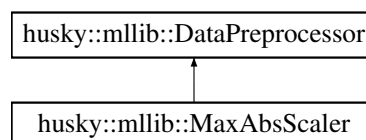
The documentation for this class was generated from the following file:

- [Classifier.hpp](#)

6.2 husky::mllib::DataPreprocessor Class Reference

```
#include <DataPreprocessor.hpp>
```

Inheritance diagram for husky::mllib::DataPreprocessor:

**Public Member Functions**

- [DataPreprocessor](#) ()
- virtual void [fit](#) (const [Instances](#) &instances)=0
- virtual void [fit_transform](#) ([Instances](#) &instances)=0
- virtual void [transform](#) ([Instances](#) &instances)=0
- virtual void [inverse_transfrom](#) ([Instances](#) &instances)=0

6.2.1 Detailed Description

This is the base class of [DataPreprocessor](#)

6.2.2 Constructor & Destructor Documentation

6.2.2.1 DataPreprocessor()

```
husky::mllib::DataPreprocessor::DataPreprocessor ( ) [inline]
```

6.2.3 Member Function Documentation

6.2.3.1 fit()

```
virtual void husky::mllib::DataPreprocessor::fit (
    const Instances & instances ) [pure virtual]
```

Fit model into data

Implemented in [husky::mllib::MaxAbsScaler](#).

6.2.3.2 fit_transform()

```
virtual void husky::mllib::DataPreprocessor::fit_transform (
    Instances & instances ) [pure virtual]
```

Fit model and transform data

Implemented in [husky::mllib::MaxAbsScaler](#).

6.2.3.3 inverse_transform()

```
virtual void husky::mllib::DataPreprocessor::inverse_transform (
    Instances & instances ) [pure virtual]
```

Inverse the transform

Implemented in [husky::mllib::MaxAbsScaler](#).

6.2.3.4 transform()

```
virtual void husky::mllib::DataPreprocessor::transform (
    Instances & instances ) [pure virtual]
```

Transform data

Implemented in [husky::mllib::MaxAbsScaler](#).

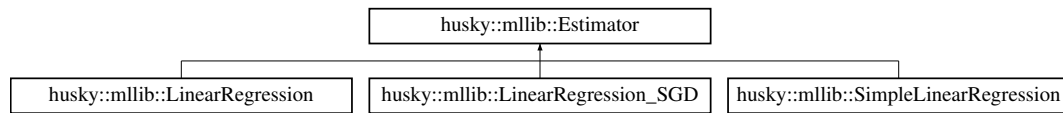
The documentation for this class was generated from the following file:

- [DataPreprocessor.hpp](#)

6.3 husky::mllib::Estimator Class Reference

```
#include <Estimator.hpp>
```

Inheritance diagram for husky::mllib::Estimator:



Public Member Functions

- virtual void [fit](#) (const [Instances](#) &instances)=0
- virtual void [fit](#) (const [Instances](#) &instances, std::string instance_weight_name)
- virtual AttrList< [Instance](#), double > & [predict](#) (const [Instances](#) &instances, std::string prediction_name="prediction")=0
- virtual [Estimator](#) * [clone](#) (int seed=0)=0

6.3.1 Member Function Documentation

6.3.1.1 clone()

```
virtual Estimator* husky::mllib::Estimator::clone (
    int seed = 0 ) [pure virtual]
```

Implemented in [husky::mllib::LinearRegression_SGD](#), [husky::mllib::LinearRegression](#), and [husky::mllib::SimpleLinearRegression](#).

6.3.1.2 fit() [1/2]

```
virtual void husky::mllib::Estimator::fit (
    const Instances & instances ) [pure virtual]
```

Implemented in [husky::mllib::LinearRegression_SGD](#), [husky::mllib::LinearRegression](#), and [husky::mllib::SimpleLinearRegression](#).

6.3.1.3 fit() [2/2]

```
virtual void husky::mllib::Estimator::fit (
    const Instances & instances,
    std::string instance_weight_name ) [inline], [virtual]
```

Reimplemented in [husky::mllib::LinearRegression_SGD](#), [husky::mllib::LinearRegression](#), and [husky::mllib::SimpleLinearRegression](#).

6.3.1.4 predict()

```
virtual AttrList<Instance, double>& husky::mllib::Estimator::predict (
    const Instances & instances,
    std::string prediction_name = "prediction" ) [pure virtual]
```

Implemented in [husky::mllib::LinearRegression_SGD](#), [husky::mllib::LinearRegression](#), and [husky::mllib::SimpleLinearRegression](#).

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

- [Estimator.hpp](#)

6.4 husky::mllib::Instance Class Reference

```
#include <Instances.hpp>
```

Public Types

- using [KeyT](#) = std::string

Public Member Functions

- [Instance](#) ()
- [Instance](#) (const [KeyT](#) &k)
- virtual [KeyT](#) const & id () const
- [Instance](#) & operator= (const [Instance](#) &instance)

Public Attributes

- [KeyT](#) key
- [vec_double](#) X

Friends

- husky::BinStream & operator<< (husky::BinStream &stream, const [Instance](#) &u)
- husky::BinStream & operator>> (husky::BinStream &stream, [Instance](#) &u)

6.4.1 Member Typedef Documentation

6.4.1.1 KeyT

```
using husky::mllib::Instance::KeyT = std::string
```

6.4.2 Constructor & Destructor Documentation

6.4.2.1 Instance() [1/2]

```
husky::mllib::Instance::Instance ( ) [inline]
```

6.4.2.2 Instance() [2/2]

```
husky::mllib::Instance::Instance (
    const KeyT & k ) [inline], [explicit]
```

6.4.3 Member Function Documentation

6.4.3.1 id()

```
virtual KeyT const& husky::mllib::Instance::id ( ) const [inline], [virtual]
```

6.4.3.2 operator=()

```
Instance& husky::mllib::Instance::operator= (
    const Instance & instance ) [inline]
```

6.4.4 Friends And Related Function Documentation

6.4.4.1 operator<<

```
husky::BinStream& operator<< (
    husky::BinStream & stream,
    const Instance & u ) [friend]
```

6.4.4.2 operator>>

```
husky::BinStream& operator>> (
    husky::BinStream & stream,
    Instance & u ) [friend]
```

6.4.5 Member Data Documentation

6.4.5.1 key

```
KeyT husky::mllib::Instance::key
```

6.4.5.2 X

```
vec_double husky::mllib::Instance::X
```

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

- [Instances.hpp](#)

6.5 husky::mllib::Instances Class Reference

```
#include <Instances.hpp>
```

Public Member Functions

- [Instances](#) ()
- `size_t` [add](#) (const [Instance](#) &instance)
- `void` [globalize](#) ()
- `template<typename AttrT >`
 `auto &` [createAttrlist](#) (std::string name) const
- `template<typename AttrT >`
 `auto &` [getAttrlist](#) (std::string name) const
- `void` [deleteAttrlist](#) (std::string name) const
- `void` [set_y](#) (const [Instance](#) &instance, double y)
- `double` [get_y](#) (const [Instance](#) &instance) const
- `void` [set_class](#) (const [Instance](#) &instance, int label)
- `int` [get_class](#) (const [Instance](#) &instance) const
- `auto &` [enumerator](#) () const
- [Instances](#) & [operator=](#) (const [Instances](#) &instances)

Public Attributes

- `int` [numAttributes](#)
- `int` [numInstances](#)
- `int` [numClasses](#)

6.5.1 Constructor & Destructor Documentation

6.5.1.1 Instances()

```
husky::mllib::Instances::Instances ( ) [inline]
```

6.5.2 Member Function Documentation

6.5.2.1 add()

```
size_t husky::mllib::Instances::add (
    const Instance & instance ) [inline]
```

6.5.2.2 createAttrlist()

```
template<typename AttrT >
auto& husky::mllib::Instances::createAttrlist (
    std::string name ) const [inline]
```

6.5.2.3 deleteAttrlist()

```
void husky::mllib::Instances::deleteAttrlist (
    std::string name ) const [inline]
```

6.5.2.4 enumerator()

```
auto& husky::mllib::Instances::enumerator ( ) const [inline]
```

6.5.2.5 get_class()

```
int husky::mllib::Instances::get_class (
    const Instance & instance ) const [inline]
```

6.5.2.6 get_y()

```
double husky::mllib::Instances::get_y (
    const Instance & instance ) const [inline]
```

6.5.2.7 getAttrlist()

```
template<typename AttrT >
auto& husky::mllib::Instances::getAttrlist (
    std::string name ) const [inline]
```

6.5.2.8 globalize()

```
void husky::mllib::Instances::globalize ( ) [inline]
```

6.5.2.9 operator=()

```
Instances& husky::mllib::Instances::operator= (
    const Instances & instances ) [inline]
```

6.5.2.10 set_class()

```
void husky::mllib::Instances::set_class (
    const Instance & instance,
    int label ) [inline]
```

6.5.2.11 set_y()

```
void husky::mllib::Instances::set_y (
    const Instance & instance,
    double y ) [inline]
```

6.5.3 Member Data Documentation

6.5.3.1 numAttributes

```
int husky::mllib::Instances::numAttributes
```

6.5.3.2 numClasses

```
int husky::mllib::Instances::numClasses
```

6.5.3.3 numInstances

```
int husky::mllib::Instances::numInstances
```

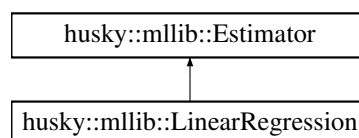
The documentation for this class was generated from the following file:

- [Instances.hpp](#)

6.6 husky::mllib::LinearRegression Class Reference

```
#include <LinearRegression.hpp>
```

Inheritance diagram for husky::mllib::LinearRegression:



Public Member Functions

- [LinearRegression](#) ()
- void [fit](#) (const [mllib::Instances](#) &original_instances)
- void [fit](#) (const [Instances](#) &instances, std::string instance_weight_name)
- AttrList< [Instance](#), double > & [predict](#) (const [mllib::Instances](#) &instances, std::string prediction_name="prediction")
- [Estimator](#) * [clone](#) (int seed=0)
- [vec_double](#) [get_parameters](#) ()

6.6.1 Constructor & Destructor Documentation

6.6.1.1 LinearRegression()

```
husky::mllib::LinearRegression::LinearRegression ( ) [inline]
```

6.6.2 Member Function Documentation

6.6.2.1 clone()

```
Estimator* husky::mllib::LinearRegression::clone (
    int seed = 0 ) [inline], [virtual]
```

Implements [husky::mllib::Estimator](#).

6.6.2.2 fit() [1/2]

```
void husky::mllib::LinearRegression::fit (
    const mllib::Instances & original_instances ) [virtual]
```

Implements [husky::mllib::Estimator](#).

6.6.2.3 fit() [2/2]

```
void husky::mllib::LinearRegression::fit (
    const Instances & instances,
    std::string instance_weight_name ) [virtual]
```

Reimplemented from [husky::mllib::Estimator](#).

6.6.2.4 get_parameters()

```
vec_double husky::mllib::LinearRegression::get_parameters ( ) [inline]
```

6.6.2.5 predict()

```
AttrList< Instance, double > & husky::mllib::LinearRegression::predict (
    const mllib::Instances & instances,
    std::string prediction_name = "prediction" ) [virtual]
```

Implements [husky::mllib::Estimator](#).

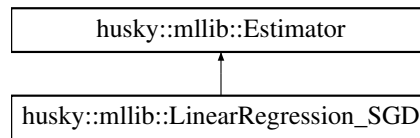
The documentation for this class was generated from the following files:

- [LinearRegression.hpp](#)
- [LinearRegression.cpp](#)

6.7 husky::mllib::LinearRegression_SGD Class Reference

```
#include <LinearRegression_SGD.hpp>
```

Inheritance diagram for husky::mllib::LinearRegression_SGD:



Public Member Functions

- [LinearRegression_SGD](#) ()
- [LinearRegression_SGD](#) (int m_iter, double e0, double al, double trival_im, double max_trival_im)
- void [fit](#) (const [Instances](#) &instances)
- void [fit](#) (const [Instances](#) &original_instances, std::string instance_weight_name)
- AttrList< [Instance](#), double > & [predict](#) (const [Instances](#) &instances, std::string prediction_name="prediction")
- [Estimator](#) * [clone](#) (int seed=0)
- [vec_double](#) [get_parameters](#) ()

6.7.1 Constructor & Destructor Documentation

6.7.1.1 [LinearRegression_SGD\(\)](#) [1/2]

```
husky::mllib::LinearRegression_SGD::LinearRegression_SGD ( ) [inline]
```

6.7.1.2 [LinearRegression_SGD\(\)](#) [2/2]

```
husky::mllib::LinearRegression_SGD::LinearRegression_SGD (
    int m_iter,
    double e0,
    double al,
    double trival_im,
    double max_trival_im ) [inline]
```

6.7.2 Member Function Documentation

6.7.2.1 [clone\(\)](#)

```
Estimator* husky::mllib::LinearRegression_SGD::clone (
    int seed = 0 ) [inline], [virtual]
```

Implements [husky::mllib::Estimator](#).

6.7.2.2 `fit()` [1/2]

```
void husky::mllib::LinearRegression_SGD::fit (
    const Instances & instances ) [virtual]
```

Implements [husky::mllib::Estimator](#).

6.7.2.3 `fit()` [2/2]

```
void husky::mllib::LinearRegression_SGD::fit (
    const Instances & original_instances,
    std::string instance_weight_name ) [virtual]
```

Reimplemented from [husky::mllib::Estimator](#).

6.7.2.4 `get_parameters()`

```
vec_double husky::mllib::LinearRegression_SGD::get_parameters ( ) [inline]
```

6.7.2.5 `predict()`

```
AttrList< Instance, double > & husky::mllib::LinearRegression_SGD::predict (
    const Instances & instances,
    std::string prediction_name = "prediction" ) [virtual]
```

Implements [husky::mllib::Estimator](#).

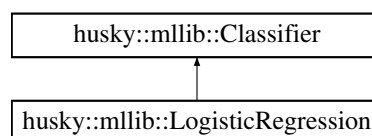
The documentation for this class was generated from the following files:

- [LinearRegression_SGD.hpp](#)
- [LinearRegression_SGD.cpp](#)

6.8 `husky::mllib::LogisticRegression` Class Reference

```
#include <LogisticRegression.hpp>
```

Inheritance diagram for `husky::mllib::LogisticRegression`:



Public Member Functions

- [LogisticRegression](#) ()
- [LogisticRegression](#) (int m_iter, double e0, double al, double trival_im, double max_trival_im, [MODE](#) m, int classNum)
- void [fit](#) (const [Instances](#) &instances)
- void [local_fit](#) (const [Instances](#) &instances)
- void [global_fit](#) (const [Instances](#) &instances)
- void [local_fit](#) (const [Instances](#) &instances, std::string instance_weight_name)
- void [global_fit](#) (const [Instances](#) &instances, std::string instance_weight_name)
- void [fit](#) (const [Instances](#) &instances, std::string instance_weight_name)
- [AttrList](#)< [Instance](#), [Prediction](#) > & [predict](#) (const [Instances](#) &instances, std::string prediction_name="prediction")
- [Classifier](#) * [clone](#) (int seed=0)
- [matrix_double](#) [get_parameters](#) ()

6.8.1 Constructor & Destructor Documentation

6.8.1.1 [LogisticRegression](#)() [1/2]

```
husky::mllib::LogisticRegression::LogisticRegression ( ) [inline]
```

6.8.1.2 [LogisticRegression](#)() [2/2]

```
husky::mllib::LogisticRegression::LogisticRegression (
    int m_iter,
    double e0,
    double al,
    double trival_im,
    double max_trival_im,
    MODE m,
    int classNum ) [inline]
```

6.8.2 Member Function Documentation

6.8.2.1 [clone](#)()

```
Classifier* husky::mllib::LogisticRegression::clone (
    int seed = 0 ) [inline], [virtual]
```

The clone function will only copy hyperparameters but not estimated parameter. e.g. parameters used to train model, including learning rate, etc.

Implements [husky::mllib::Classifier](#).

6.8.2.2 [fit](#)() [1/2]

```
void husky::mllib::LogisticRegression::fit (
    const Instances & instances ) [virtual]
```

Train a classifier with instances.

Implements [husky::mllib::Classifier](#).

6.8.2.3 fit() [2/2]

```
void husky::mllib::LogisticRegression::fit (
    const Instances & instances,
    std::string instance_weight_name ) [virtual]
```

Train a classifier with weighted instances.

Implements [husky::mllib::Classifier](#).

6.8.2.4 get_parameters()

```
matrix_double husky::mllib::LogisticRegression::get_parameters ( ) [inline]
```

6.8.2.5 global_fit() [1/2]

```
void husky::mllib::LogisticRegression::global_fit (
    const Instances & instances )
```

6.8.2.6 global_fit() [2/2]

```
void husky::mllib::LogisticRegression::global_fit (
    const Instances & instances,
    std::string instance_weight_name )
```

6.8.2.7 local_fit() [1/2]

```
void husky::mllib::LogisticRegression::local_fit (
    const Instances & instances )
```

6.8.2.8 local_fit() [2/2]

```
void husky::mllib::LogisticRegression::local_fit (
    const Instances & instances,
    std::string instance_weight_name )
```

6.8.2.9 predict()

```
AttrList< Instance, Prediction > & husky::mllib::LogisticRegression::predict (
    const Instances & instances,
    std::string prediction_name = "prediction" ) [virtual]
```

Do prediction, and results are stored in a AttrList with name being prediction_name

Implements [husky::mllib::Classifier](#).

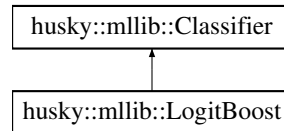
The documentation for this class was generated from the following files:

- [LogisticRegression.hpp](#)
- [LogisticRegression.cpp](#)

6.9 husky::mllib::LogitBoost Class Reference

```
#include <LogitBoost.hpp>
```

Inheritance diagram for husky::mllib::LogitBoost:



Public Member Functions

- [LogitBoost](#) ([Estimator](#) *baselearner, double maxIter, double heuristicStop)
- void [fit](#) (const [mllib::Instances](#) &original_instances)
- void [fit](#) (const [Instances](#) &instances, std::string instance_weight_name)
- [AttrList](#)< [Instance](#), [Prediction](#) > & [predict](#) (const [Instances](#) &instances, std::string prediction_name="prediction")
- [Classifier](#) * [clone](#) (int seed=0)
- [Estimator](#) * [get_baselearner](#) (int j, int m)

6.9.1 Constructor & Destructor Documentation

6.9.1.1 LogitBoost()

```

husky::mllib::LogitBoost::LogitBoost (
    Estimator * baselearner,
    double maxIter,
    double heuristicStop ) [inline]

```

6.9.2 Member Function Documentation

6.9.2.1 clone()

```

Classifier* husky::mllib::LogitBoost::clone (
    int seed = 0 ) [inline], [virtual]

```

The clone function will only copy hyperparameters but not estimated parameter. e.g. parameters used to train model, including learning rate, etc.

Implements [husky::mllib::Classifier](#).

6.9.2.2 fit() [1/2]

```

void husky::mllib::LogitBoost::fit (
    const mllib::Instances & instances ) [virtual]

```

Train a classifier with instances.

Implements [husky::mllib::Classifier](#).

6.9.2.3 fit() [2/2]

```
void husky::mllib::LogitBoost::fit (
    const Instances & instances,
    std::string instance_weight_name ) [virtual]
```

Train a classifier with weighted instances.

Implements [husky::mllib::Classifier](#).

6.9.2.4 get_baselearner()

```
Estimator* husky::mllib::LogitBoost::get_baselearner (
    int j,
    int m ) [inline]
```

6.9.2.5 predict()

```
AttrList< Instance, Prediction > & husky::mllib::LogitBoost::predict (
    const Instances & instances,
    std::string prediction_name = "prediction" ) [virtual]
```

Do prediction, and results are stored in a AttrList with name being prediction_name

Implements [husky::mllib::Classifier](#).

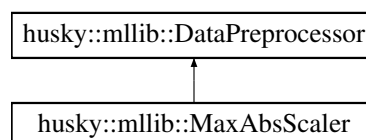
The documentation for this class was generated from the following files:

- [LogitBoost.hpp](#)
- [LogitBoost.cpp](#)

6.10 husky::mllib::MaxAbsScaler Class Reference

```
#include <MaxAbsScaler.hpp>
```

Inheritance diagram for husky::mllib::MaxAbsScaler:



Public Member Functions

- [MaxAbsScaler](#) ()
- [vec_double](#) & [get_params](#) ()
- void [set_params](#) ([vec_double](#) params)
- void [fit](#) (const [Instances](#) &instances)
- void [fit_transform](#) ([Instances](#) &instances)
- void [transform](#) ([Instances](#) &instances)
- void [inverse_transform](#) ([Instances](#) &instances)

6.10.1 Constructor & Destructor Documentation

6.10.1.1 MaxAbsScaler()

```
husky::mllib::MaxAbsScaler::MaxAbsScaler ( ) [inline]
```

6.10.2 Member Function Documentation

6.10.2.1 fit()

```
void husky::mllib::MaxAbsScaler::fit (
    const Instances & instances ) [virtual]
```

Fit model into data

Implements [husky::mllib::DataPreprocessor](#).

6.10.2.2 fit_transform()

```
void husky::mllib::MaxAbsScaler::fit_transform (
    Instances & instances ) [virtual]
```

Fit model and transform data

Implements [husky::mllib::DataPreprocessor](#).

6.10.2.3 get_params()

```
vec_double& husky::mllib::MaxAbsScaler::get_params ( ) [inline]
```

6.10.2.4 inverse_transform()

```
void husky::mllib::MaxAbsScaler::inverse_transform (
    Instances & instances ) [virtual]
```

Inverse the transform

Implements [husky::mllib::DataPreprocessor](#).

6.10.2.5 set_params()

```
void husky::mllib::MaxAbsScaler::set_params (
    vec_double params ) [inline]
```

6.10.2.6 transform()

```
void husky::mllib::MaxAbsScaler::transform (
    Instances & instances ) [virtual]
```

Transform data

Implements [husky::mllib::DataPreprocessor](#).

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

- [MaxAbsScaler.hpp](#)
- [MaxAbsScaler.cpp](#)

6.11 PObject Class Reference

```
#include <testcompile.hpp>
```

Public Types

- typedef int [KeyT](#)

Public Member Functions

- [PObject](#) ([KeyT](#) key)
- const int & [id](#) () const

Public Attributes

- int [key](#)

6.11.1 Member Typedef Documentation

6.11.1.1 KeyT

```
typedef int PObject::KeyT
```

6.11.2 Constructor & Destructor Documentation

6.11.2.1 PObject()

```
PObject::PObject (
    KeyT key ) [explicit]
```


6.11.3 Member Function Documentation

6.11.3.1 id()

```
const int & PObject::id ( ) const
```

6.11.4 Member Data Documentation

6.11.4.1 key

```
int PObject::key
```

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

- [testcompile.hpp](#)
- [testcompile.cpp](#)

6.12 husky::mllib::Prediction Class Reference

```
#include <Classifier.hpp>
```

Public Member Functions

- [Prediction](#) ()
- [Prediction](#) (int l, [vec_double](#) p)
- [Prediction](#) & [operator=](#) (const [Prediction](#) &p)

Public Attributes

- int [label](#)
- [vec_double](#) [proba](#)

Friends

- husky::BinStream & [operator<<](#) (husky::BinStream &stream, const [Prediction](#) &p)
- husky::BinStream & [operator>>](#) (husky::BinStream &stream, [Prediction](#) &p)

6.12.1 Constructor & Destructor Documentation

6.12.1.1 Prediction() [1/2]

```
husky::mllib::Prediction::Prediction ( ) [inline]
```

6.12.1.2 Prediction() [2/2]

```
husky::mllib::Prediction::Prediction (
    int l,
    vec_double p ) [inline], [explicit]
```

6.12.2 Member Function Documentation

6.12.2.1 operator=()

```
Prediction& husky::mllib::Prediction::operator= (
    const Prediction & p ) [inline]
```

6.12.3 Friends And Related Function Documentation

6.12.3.1 operator<<

```
husky::BinStream& operator<< (
    husky::BinStream & stream,
    const Prediction & p ) [friend]
```

6.12.3.2 operator>>

```
husky::BinStream& operator>> (
    husky::BinStream & stream,
    Prediction & p ) [friend]
```

6.12.4 Member Data Documentation

6.12.4.1 label

```
int husky::mllib::Prediction::label
```

6.12.4.2 proba

```
vec_double husky::mllib::Prediction::proba
```

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

- [Classifier.hpp](#)

6.13 husky::mllib::PseudoObject Class Reference

```
#include <LinearRegression.hpp>
```

Public Types

- typedef int [KeyT](#)

Public Member Functions

- [PseudoObject](#) ()
- [PseudoObject](#) ([KeyT](#) key)
- const int & [id](#) () const

Public Attributes

- int [key](#)

Friends

- husky::BinStream & [operator<<](#) (husky::BinStream &stream, const [PseudoObject](#) &u)
- husky::BinStream & [operator>>](#) (husky::BinStream &stream, [PseudoObject](#) &u)

6.13.1 Member Typedef Documentation

6.13.1.1 KeyT

```
typedef int husky::mllib::PseudoObject::KeyT
```

6.13.2 Constructor & Destructor Documentation

6.13.2.1 PseudoObject() [1/2]

```
husky::mllib::PseudoObject::PseudoObject ( ) [inline]
```

6.13.2.2 PseudoObject() [2/2]

```
husky::mllib::PseudoObject::PseudoObject (
    KeyT key ) [inline], [explicit]
```

6.13.3 Member Function Documentation

6.13.3.1 id()

```
const int& husky::mllib::PseudoObject::id ( ) const [inline]
```

6.13.4 Friends And Related Function Documentation

6.13.4.1 operator<<

```
husky::BinStream& operator<< (
    husky::BinStream & stream,
    const PseudoObject & u ) [friend]
```

6.13.4.2 operator>>

```
husky::BinStream& operator>> (
    husky::BinStream & stream,
    PseudoObject & u ) [friend]
```

6.13.5 Member Data Documentation

6.13.5.1 key

```
int husky::mllib::PseudoObject::key
```

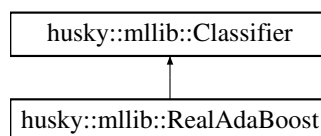
The documentation for this class was generated from the following file:

- [LinearRegression.hpp](#)

6.14 husky::mllib::RealAdaBoost Class Reference

```
#include <RealAdaBoost.hpp>
```

Inheritance diagram for husky::mllib::RealAdaBoost:



Public Member Functions

- [RealAdaBoost](#) ()
- [RealAdaBoost](#) ([Classifier](#) *bM, int mi)
- void [fit](#) (const [Instances](#) &instances)
- void [fit](#) (const [Instances](#) &instances, std::string instance_weight_name)
- [AttrList](#)< [Instance](#), [Prediction](#) > & [predict](#) (const [Instances](#) &instances, std::string prediction_name="prediction")
- [Classifier](#) * [clone](#) (int seed=0)
- [~RealAdaBoost](#) ()
- void [clear_community](#) ()

6.14.1 Constructor & Destructor Documentation

6.14.1.1 RealAdaBoost() [1/2]

```
husky::mllib::RealAdaBoost::RealAdaBoost ( ) [inline]
```

6.14.1.2 RealAdaBoost() [2/2]

```
husky::mllib::RealAdaBoost::RealAdaBoost (
    Classifier * bM,
    int mi ) [inline], [explicit]
```

6.14.1.3 ~RealAdaBoost()

```
husky::mllib::RealAdaBoost::~~RealAdaBoost ( ) [inline]
```

6.14.2 Member Function Documentation

6.14.2.1 clear_community()

```
void husky::mllib::RealAdaBoost::clear_community ( ) [inline]
```

6.14.2.2 clone()

```
Classifier* husky::mllib::RealAdaBoost::clone (
    int seed = 0 ) [inline], [virtual]
```

The clone function will only copy hyperparameters but not estimated parameter. e.g. parameters used to train model, including learning rate, etc.

Implements [husky::mllib::Classifier](#).

6.14.2.3 fit() [1/2]

```
void husky::mllib::RealAdaBoost::fit (
    const Instances & instances ) [virtual]
```

Train a classifier with instances.

Implements [husky::mllib::Classifier](#).

6.14.2.4 fit() [2/2]

```
void husky::mllib::RealAdaBoost::fit (
    const Instances & instances,
    std::string instance_weight_name ) [virtual]
```

Train a classifier with weighted instances.

Implements [husky::mllib::Classifier](#).

6.14.2.5 predict()

```
AttrList< Instance, Prediction > & husky::mllib::RealAdaBoost::predict (
    const Instances & instances,
    std::string prediction_name = "prediction" ) [virtual]
```

Do prediction, and results are stored in a AttrList with name being prediction_name

Implements [husky::mllib::Classifier](#).

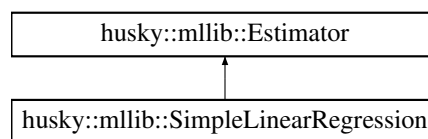
The documentation for this class was generated from the following files:

- [RealAdaBoost.hpp](#)
- [RealAdaBoost.cpp](#)

6.15 husky::mllib::SimpleLinearRegression Class Reference

```
#include <SimpleLinearRegression.hpp>
```

Inheritance diagram for husky::mllib::SimpleLinearRegression:



Public Member Functions

- [SimpleLinearRegression](#) ()
- void [fit](#) (const [mllib::Instances](#) &original_instances)
- void [fit](#) (const [mllib::Instances](#) &original_instances, std::string weight_name)
- AttrList< [Instance](#), double > & [predict](#) (const [mllib::Instances](#) &instances, std::string prediction_name="prediction")
- [Estimator](#) * [clone](#) (int seed=0)
- double [get_slope](#) ()
- double [get_intercept](#) ()
- int [get_selected](#) ()

6.15.1 Constructor & Destructor Documentation

6.15.1.1 SimpleLinearRegression()

```
husky::mllib::SimpleLinearRegression::SimpleLinearRegression ( ) [inline]
```

6.15.2 Member Function Documentation

6.15.2.1 clone()

```
Estimator* husky::mllib::SimpleLinearRegression::clone (
    int seed = 0 ) [inline], [virtual]
```

Implements [husky::mllib::Estimator](#).

6.15.2.2 fit() [1/2]

```
void husky::mllib::SimpleLinearRegression::fit (
    const mllib::Instances & original_instances ) [virtual]
```

Implements [husky::mllib::Estimator](#).

6.15.2.3 fit() [2/2]

```
void husky::mllib::SimpleLinearRegression::fit (
    const mllib::Instances & original_instances,
    std::string weight_name ) [virtual]
```

Reimplemented from [husky::mllib::Estimator](#).

6.15.2.4 get_intercept()

```
double husky::mllib::SimpleLinearRegression::get_intercept ( ) [inline]
```

6.15.2.5 get_selected()

```
int husky::mllib::SimpleLinearRegression::get_selected ( ) [inline]
```

6.15.2.6 get_slope()

```
double husky::mllib::SimpleLinearRegression::get_slope ( ) [inline]
```

6.15.2.7 predict()

```
AttrList< Instance, double > & husky::mllib::SimpleLinearRegression::predict (
    const mllib::Instances & instances,
    std::string prediction_name = "prediction" ) [virtual]
```

Implements [husky::mllib::Estimator](#).

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

- [SimpleLinearRegression.hpp](#)
- [SimpleLinearRegression.cpp](#)

Chapter 7

File Documentation

7.1 Classifier.hpp File Reference

```
#include "mllib/Instances.hpp"
```

Classes

- class [husky::mllib::Prediction](#)
- class [husky::mllib::Classifier](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

7.2 DataPreprocessor.hpp File Reference

```
#include "mllib/Instances.hpp"
```

Classes

- class [husky::mllib::DataPreprocessor](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

7.3 DataReader.cpp File Reference

```
#include "mllib/DataReader.hpp"
```

Namespaces

- [husky](#)
- [husky::mllib](#)

Functions

- void [husky::mllib::svReader](#) (Instances &instances, std::string filepath, boost::char_separator< char > delimiter, LABEL_TYPE label_type)

7.4 DataReader.hpp File Reference

```
#include <string>
#include <stdexcept>
#include "mllib/Instances.hpp"
#include "io/input/line_inputformat.hpp"
#include "boost/tokenizer.hpp"
#include "lib/aggregator_factory.hpp"
```

Namespaces

- [husky](#)
- [husky::mllib](#)

Enumerations

- enum [husky::mllib::LABEL_TYPE](#) { [husky::mllib::LABEL_TYPE::NO_LABEL](#) =0, [husky::mllib::LABEL_TYPE::E::Y](#) =1, [husky::mllib::LABEL_TYPE::CLASS](#) =2 }

Functions

- void [husky::mllib::svReader](#) (Instances &instances, std::string filepath, boost::char_separator< char > delimiter, LABEL_TYPE label_type)

7.5 Estimator.hpp File Reference

```
#include <mllib/Instances.hpp>
```

Classes

- class [husky::mllib::Estimator](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

7.6 Instances.hpp File Reference

```
#include <vector>
#include <string>
#include <core/engine.hpp>
```

Classes

- class [husky::mllib::Instance](#)
- class [husky::mllib::Instances](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

Typedefs

- typedef std::vector< double > [vec_double](#)

7.6.1 Typedef Documentation

7.6.1.1 vec_double

```
typedef std::vector<double> vec\_double
```

7.7 LinearRegression.cpp File Reference

```
#include "mllib/LinearRegression.hpp"
#include "mllib/Utility.hpp"
```

Namespaces

- [husky](#)
- [husky::mllib](#)

7.8 LinearRegression.hpp File Reference

```
#include "mllib/Instances.hpp"
#include "mllib/Estimator.hpp"
#include "lib/aggregator_factory.hpp"
```

Classes

- class [husky::mllib::PseudoObject](#)
- class [husky::mllib::LinearRegression](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

7.9 LinearRegression_SGD.cpp File Reference

```
#include "LinearRegression_SGD.hpp"
```

Namespaces

- [husky](#)
- [husky::mllib](#)

7.10 LinearRegression_SGD.hpp File Reference

```
#include <limits>
#include "mllib/Utility.hpp"
#include "mllib/Instances.hpp"
#include "mllib/Estimator.hpp"
#include "lib/aggregator_factory.hpp"
```

Classes

- class [husky::mllib::LinearRegression_SGD](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

7.11 LogisticRegression.cpp File Reference

```
#include "LogisticRegression.hpp"
```

Namespaces

- [husky](#)
- [husky::mllib](#)

7.12 LogisticRegression.hpp File Reference

```
#include <limits>
#include "mllib/Utility.hpp"
#include "mllib/Instances.hpp"
#include "lib/aggregator_factory.hpp"
#include "mllib/Classifier.hpp"
```

Classes

- class [husky::mllib::LogisticRegression](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

Enumerations

- enum [husky::mllib::MODE](#) { [husky::mllib::MODE::GLOBAL](#) =0, [husky::mllib::MODE::LOCAL](#) =1 }

7.13 LogitBoost.cpp File Reference

```
#include "mllib/LogitBoost.hpp"
```

Namespaces

- [husky](#)
- [husky::mllib](#)

7.14 LogitBoost.hpp File Reference

```
#include "mllib/Instances.hpp"
#include "mllib/Classifier.hpp"
#include "mllib/Utility.hpp"
#include "mllib/Estimator.hpp"
#include "lib/aggregator_factory.hpp"
#include <math.h>
#include <float.h>
```

Classes

- class [husky::mllib::LogitBoost](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

7.15 MaxAbsScaler.cpp File Reference

```
#include <algorithm>
#include "mllib/MaxAbsScaler.hpp"
```

Namespaces

- [husky](#)
- [husky::mllib](#)

Functions

- void [husky::mllib::max_abs_vec](#) ([vec_double](#) &va, const [vec_double](#) &vb)

7.16 MaxAbsScaler.hpp File Reference

```
#include "mllib/Utility.hpp"
#include "mllib/DataPreprocessor.hpp"
#include "lib/aggregator_factory.hpp"
```

Classes

- class [husky::mllib::MaxAbsScaler](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

7.17 RealAdaBoost.cpp File Reference

```
#include "mllib/RealAdaBoost.hpp"
```

Namespaces

- [husky](#)
- [husky::mllib](#)

7.18 RealAdaBoost.hpp File Reference

```
#include "mllib/Utility.hpp"  
#include "mllib/Instances.hpp"  
#include "mllib/Classifier.hpp"  
#include "lib/aggregator_factory.hpp"
```

Classes

- class [husky::mllib::RealAdaBoost](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

7.19 SimpleLinearRegression.cpp File Reference

```
#include "mllib/SimpleLinearRegression.hpp"  
#include "mllib/Utility.hpp"
```

Namespaces

- [husky](#)
- [husky::mllib](#)

7.20 SimpleLinearRegression.hpp File Reference

```
#include "mllib/Instances.hpp"
#include "mllib/Estimator.hpp"
#include <float.h>
#include "lib/aggregator_factory.hpp"
```

Classes

- class [husky::mllib::SimpleLinearRegression](#)

Namespaces

- [husky](#)
- [husky::mllib](#)

7.21 testcompile.cpp File Reference

```
#include "mllib/testcompile.hpp"
```

Functions

- void [pi](#) ()

7.21.1 Function Documentation

7.21.1.1 [pi\(\)](#)

```
void pi ( )
```

7.22 testcompile.hpp File Reference

```
#include <random>
#include <string>
#include <vector>
#include "core/engine.hpp"
```


Classes

- class [PObject](#)

Functions

- void [pi](#) ()

7.22.1 Function Documentation

7.22.1.1 [pi\(\)](#)

```
void pi ( )
```

7.23 Utility.hpp File Reference

```
#include <cstdint>
#include <string>
#include <vector>
#include <ctime>
#include <cmath>
#include <cstdlib>
#include <sstream>
#include <limits>
#include <boost/numeric/ublas/vector.hpp>
#include <boost/numeric/ublas/vector_proxy.hpp>
#include <boost/numeric/ublas/matrix.hpp>
#include <boost/numeric/ublas/triangular.hpp>
#include <boost/numeric/ublas/lu.hpp>
#include <boost/numeric/ublas/io.hpp>
```

Typedefs

- typedef std::vector< double > [vec_double](#)
- typedef std::vector< std::vector< double > > [matrix](#)
- typedef std::vector< std::vector< double > > [matrix_double](#)

Functions

- `std::string vec_to_str` (const `vec_double` &v)
- `std::string matrix_to_str` (`matrix_double` &m)
- `matrix_double & operator+=` (`matrix_double` &m1, const `matrix_double` &m2)
- `matrix_double & operator/=` (`matrix_double` &m, const double coe)
- `matrix_double & operator*=` (`matrix_double` &m, const double coe)
- `matrix_double operator*` (const `matrix_double` &m, const double coe)
- `double sum` (const `vec_double` &v)
- `double operator*` (const `vec_double` &va, const `vec_double` &vb)
- `vec_double & operator+=` (`vec_double` &va, const `vec_double` &vb)
- `vec_double operator+` (const `vec_double` &va, const `vec_double` &vb)
- `vec_double & operator-=` (`vec_double` &va, const `vec_double` &vb)
- `vec_double & operator*=` (`vec_double` &va, const double &c)
- `vec_double operator*` (const double &c, const `vec_double` &va)
- `vec_double & operator/=` (`vec_double` &va, const double &c)
- `vec_double & operator/=` (`vec_double` &a, const `vec_double` &b)
- `vec_double & operator*=` (`vec_double` &a, const `vec_double` &b)
- `template<class T >`
`bool InvertMatrix` (const `ublas::matrix< T >` &input, `ublas::matrix< T >` &inverse)
- `bool MatrixInversion` (`matrix` &input)
- `void MatrixVectormultiplication` (const `matrix` &A, const `vec_double` &B, `vec_double` &output)

7.23.1 Typedef Documentation

7.23.1.1 matrix

```
typedef std::vector<std::vector<double> > matrix
```

7.23.1.2 matrix_double

```
typedef std::vector<std::vector<double> > matrix_double
```

7.23.1.3 vec_double

```
typedef std::vector<double> vec_double
```

7.23.2 Function Documentation

7.23.2.1 InvertMatrix()

```
template<class T >
bool InvertMatrix (
    const ublas::matrix< T > & input,
    ublas::matrix< T > & inverse )
```

7.23.2.2 matrix_to_str()

```
std::string matrix_to_str (
    matrix_double & m ) [inline]
```

7.23.2.3 MatrixInversion()

```
bool MatrixInversion (
    matrix & input ) [inline]
```

7.23.2.4 MatrixVectormultiplication()

```
void MatrixVectormultiplication (
    const matrix & A,
    const vec_double & B,
    vec_double & output ) [inline]
```

7.23.2.5 operator*() [1/3]

```
matrix_double operator* (
    const matrix_double & m,
    const double coe ) [inline]
```

7.23.2.6 operator*() [2/3]

```
double operator* (
    const vec_double & va,
    const vec_double & vb ) [inline]
```

7.23.2.7 operator*() [3/3]

```
vec_double operator* (
    const double & c,
    const vec_double & va ) [inline]
```

7.23.2.8 operator*=() [1/3]

```
matrix_double& operator*= (
    matrix_double & m,
    const double coe ) [inline]
```

7.23.2.9 operator*=() [2/3]

```
vec_double& operator*= (
    vec_double & va,
    const double & c ) [inline]
```

7.23.2.10 operator*=() [3/3]

```
vec_double& operator*= (
    vec_double & a,
    const vec_double & b ) [inline]
```

7.23.2.11 operator+()

```
vec_double operator+ (
    const vec_double & va,
    const vec_double & vb ) [inline]
```

7.23.2.12 operator+=() [1/2]

```
matrix_double& operator+= (
    matrix_double & m1,
    const matrix_double & m2 ) [inline]
```

7.23.2.13 operator+=() [2/2]

```
vec_double& operator+= (
    vec_double & va,
    const vec_double & vb ) [inline]
```

7.23.2.14 operator-=()

```
vec_double& operator-= (
    vec_double & va,
    const vec_double & vb ) [inline]
```

7.23.2.15 operator/=() [1/3]

```
matrix_double& operator/= (
    matrix_double & m,
    const double coe ) [inline]
```

7.23.2.16 operator/=() [2/3]

```
vec_double& operator/= (
    vec_double & va,
    const double & c ) [inline]
```

7.23.2.17 operator/=() [3/3]

```
vec_double& operator/= (
    vec_double & a,
    const vec_double & b ) [inline]
```

7.23.2.18 sum()

```
double sum (  
    const vec_double & v ) [inline]
```

7.23.2.19 vec_to_str()

```
std::string vec_to_str (  
    const vec_double & v ) [inline]
```

