MIML Library

Damian Martinez

CONTENTS

1	Classifier	1
	l.1 mi	1
	1.2 MIMLClassifier	4
	1.3 mimlTOml	6
	1.4 mimlTOmi	8
2	Data	15
	2.1 Instance	15
	2.2 Bag	
	2.3 MIMLDataset	
	Dataset utils	
3	Report	31
	Report 3.1 Report	31
4	Fransformation	33
	4.1 mimlTOmi	33
	4.2 mimlTOml	35
In	ex	39

CHAPTER

ONE

CLASSIFIER

1.1 mi

1.1.1 APRClassifier

miml.classifier.mi.apr_classifier.	Classifier for All-Positive Bags using Axis-Aligned Pos-
APRClassifier	itive Region.

miml.classifier.mi.apr_classifier.APRClassifier

class miml.classifier.mi.apr_classifier.APRClassifier

Bases: object

Classifier for All-Positive Bags using Axis-Aligned Positive Region.

This classifier assigns a positive label to bags that contain instances within a predefined axis-parallel rectangle (APR) defined by the minimum and maximum feature values of positive instances in the training set.

Attributes

apr

 $[list[np.ndarray\ of\ shape\ (n_labels)]]$ List containing the minimum and maximum feature values defining the APR.

References

Dietterich, Thomas G., Richard H. Lathrop, and Tomás Lozano-Pérez. "Solving the multiple instance problem with axis-parallel rectangles." Artificial intelligence 89.1 (1997): 31-71.

fit(x_train: ndarray, y_train: ndarray) \rightarrow None

Fit the classifier to the training data.

x_train

[ndarray of shape (n_bags, n_instances, n_features)] Features values of bags in the training set.

y_train

[ndarray (n_bags, n_instances, n_labels)] Labels of bags in the training set.

```
predict(bag: ndarray) \rightarrow int
```

Predict the label of the bag

Parameters

bag: np.ndarray of shape(n_instances, n_features)

Features values of a bag

Returns

label: int

Predicted label of the bag

```
predict\_proba(x\_test: ndarray) \rightarrow ndarray
```

Predict probabilities of given data of having a positive label

Parameters

x_test

[np.ndarray of shape (n_bags, n_instances, n_features)] Data to predict probabilities

Returns

results: np.ndarray of shape (n_instances)

Predicted probabilities for given data

1.1.2 MIWrapperClassifier

miml.classifier.mi.mi_wrapper_classifier. MIWrapper Classifier.
MIWrapperClassifier

miml.classifier.mi.mi_wrapper_classifier.MIWrapperClassifier

class miml.classifier.mi.mi_wrapper_classifier.MIWrapperClassifier(base_classifier=DecisionTreeClassifier())

Bases: object

MIWrapper Classifier.

A simple Wrapper method for applying standard propositional learners to multi-instance data.

Attributes

base classifier

Classifier to be used

References

E. T. Frank, X. Xu (2003). Applying propositional learning algorithms to multi-instance data. Department of Computer Science, University of Waikato, Hamilton, NZ.

fit(x_{train} : ndarray, y_{train} : ndarray, weight: int = 2)

Fit the classifier to the training data.

Parameters

x train

[ndarray of shape (n_bags, n_instances, n_features)] Features values of bags in the training set.

y_train

[ndarray (n_bags, n_instances, n_labels)] Labels of bags in the training set.

weight

[int, default = 2]

The type of weight setting for each single-instance:

0: weight = 1.0 1: weight = 1.0/Total number of single-instance in the corresponding bag 2: weight = Total number of single-instance / (Total number of bags * Total number of single-instance in the corresponding bag).

 $predict(bag: ndarray) \rightarrow int$

Predict the label of the bag

Parameters

bag: np.ndarray of shape(n_instances, n_features)

Features values of a bag

1.1. mi 3

label: int

Predicted label of the bag

```
predict\_proba(x\_test: ndarray) \rightarrow ndarray
```

Predict probabilities of given data of having a positive label

Parameters

x_test

[np.ndarray of shape (n_bags, n_instances, n_features)] Data to predict probabilities

Returns

results: np.ndarray of shape (n_instances, n_labels)

Predicted probabilities for given data

1.2 MIMLClassifier

miml.classifier.miml_classifier.
MIMLClassifier

Class to represent a MIMLClassifier

1.2.1 miml.classifier.miml classifier.MIMLClassifier

class miml.classifier.miml_classifier.MIMLClassifier

Bases: ABC

Class to represent a MIMLClassifier

abstract evaluate(*dataset_test*: MIMLDataset) → ndarray

Evaluate the model on a test dataset

Parameters

dataset test

[MIMLDataset] Test dataset to evaluate the model on.

Returns results [ndarray of shape (n_bags, n_labels)] Predicted labels of dataset_test **fit**(*dataset_train*: MIMLDataset) → None Training the classifier **Parameters** dataset_train [MIMLDataset] Dataset to train the classifier abstract fit_internal($dataset_train: MIMLDataset$) $\rightarrow None$ Internal method to train the classifier **Parameters** dataset train [MIMLDataset] Dataset to train the classifier **abstract predict**(x: ndarray) \rightarrow ndarray Predict labels of given data [ndarray of shape (n, n_labels)] Data to predict their labels **Returns** results [ndarray of shape (n_bags, n_labels)] Predicted labels of data **abstract predict_bag**(bag: Bag) \rightarrow ndarray Predict labels of a given bag **Parameters** bag [Bag] Bag to predict their labels Returns

results

1.2. MIMLClassifier 5

[ndarray of shape (n_bags, n_labels)] Predicted labels of the bag

abstract predict_proba(*dataset_test:* MIMLDataset) → ndarray Predict probabilities of given dataset of having a positive label

dataset test

[MIMLDataset] Dataset to predict probabilities

Returns

results: np.ndarray of shape (n_instances, n_features)

Predicted probabilities for given dataset

1.3 mimITOmI

1.3.1 MIMLtoMLClassifier

```
miml.classifier.mimlTOml.
miml_to_ml_classifier.MIMLtoMLClassifier
```

miml.classifier.mimlTOml.miml_to_ml_classifier.MIMLtoMLClassifier

Bases: MIMLClassifier

evaluate($dataset_test$: MIMLDataset) \rightarrow ndarray

Evaluate the model on a test dataset

Parameters

dataset test

[MIMLDataset] Test dataset to evaluate the model on

Returns

results

[ndarray of shape (n_bags, n_labels)] Predicted labels of dataset_test

fit(*dataset_train*: MIMLDataset) → None

Training the classifier

dataset_train

[MIMLDataset] Dataset to train the classifier

 $fit_internal(dataset_train: MIMLDataset) \rightarrow None$

Training the classifier

Parameters

dataset train

[MIMLDataset] Dataset to train the classifier

 $predict(x: ndarray) \rightarrow ndarray$

Predict labels of given data

Parameters

 \mathbf{X}

[ndarray of shape (n_instances, n_labels)] Data to predict their labels

Returns

results

[ndarray of shape (n_instances, n_labels)] Predicted labels of data

 $predict_bag(bag: Bag) \rightarrow ndarray$

Predict labels of a given bag

Parameters

bag

[Bag] Bag to predict their labels

Returns

results

[ndarray of shape (n_labels)] Predicted labels of data

predict_proba(dataset_test: MIMLDataset) → ndarray

Predict probabilities of given dataset of having a positive label

1.3. mimITOmI 7

dataset test

[MIMLDataset] Dataset to predict probabilities

Returns

results: np.ndarray of shape (n_instances, n_features)

Predicted probabilities for given dataset

1.4 mimITOmi

1.4.1 MIMLtoMIClassifier

miml.classifier.mimlTOmi.
miml_to_mi_classifier.MIMLtoMIClassifier

Class to represent a multi-instance classifier

miml.classifier.mimlTOmi.miml to mi classifier.MIMLtoMIClassifier

class miml.classifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier(mi_classifier)

Bases: MIMLClassifier

Class to represent a multi-instance classifier

evaluate(*dataset_test*: MIMLDataset) → ndarray

Evaluate the model on a test dataset

Parameters

$dataset_test$

[MIMLDataset] Test dataset to evaluate the model on

Returns

results

[ndarray of shape (n_bags, n_labels)] Predicted labels of dataset_test

 $fit(dataset_train: MIMLDataset) \rightarrow None$

Training the classifier

dataset_train

[MIMLDataset] Dataset to train the classifier

 $abstract fit_internal(dataset_train: MIMLDataset) \rightarrow None$

Training the classifier

Parameters

dataset_train: MIMLDataset

Dataset to train the classifier

abstract predict(x: ndarray) \rightarrow ndarray

Predict labels of given data

Parameters

X

[ndarray of shape (n_instances, n_labels)] Data to predict their labels

Returns

results

[ndarray of shape (n_labels)] Predicted labels

 $predict_bag(bag: Bag) \rightarrow ndarray$

Predict labels of a given bag

Parameters

bag

[Bag] Bag to predict their labels

Returns

results

[ndarray of shape (n_labels)] Predicted labels of the bag

 $abstract\ predict_proba(\mathit{dataset_test:}\ MIMLDataset) \rightarrow ndarray$

Predict probabilities of given dataset of having a positive label

1.4. mimITOmi 9

dataset test

[MIMLDataset] Dataset to predict probabilities

Returns

results: np.ndarray of shape (n_instances, n_features)

Predicted probabilities for given dataset

1.4.2 MIMLtoMIBRClassifier

miml.classifier.mimlTOmi.
miml_to_mi_br_classifier.
MIMLtoMIBRClassifier

Class to represent a multi-instance classifier using a binary relevance transformation

miml.classifier.mimlTOmi.miml to mi br classifier.MIMLtoMIBRClassifier

class miml.classifier.mimlTOmi.miml_to_mi_br_classifier.MIMLtoMIBRClassifier(mi_classifier)

Bases: MIMLtoMIClassifier

Class to represent a multi-instance classifier using a binary relevance transformation

evaluate(*dataset_test*: MIMLDataset) → ndarray

Evaluate the model on a test dataset

Parameters

$dataset_test$

[MIMLDataset] Test dataset to evaluate the model on

Returns

results

[ndarray of shape (n_bags, n_labels)] Predicted labels of dataset_test

 $fit(dataset_train: MIMLDataset) \rightarrow None$

Training the classifier

$dataset_train$

[MIMLDataset] Dataset to train the classifier

 $fit_internal(dataset_train: MIMLDataset) \rightarrow None$

Training the classifier

Parameters

dataset_train: MIMLDataset

Dataset to train the classifier

 $predict(x: ndarray) \rightarrow ndarray$

Predict labels of given data

Parameters

 \mathbf{X}

[ndarray of shape (n_instances, n_labels)] Data to predict their labels

Returns

results

[ndarray of shape (n_labels)] Predicted labels

 $predict_bag(bag: Bag) \rightarrow ndarray$

Predict labels of a given bag

Parameters

bag

[Bag] Bag to predict their labels

Returns

results

[ndarray of shape (n_labels)] Predicted labels of the bag

predict_proba(dataset_test: MIMLDataset)

Predict probabilities of given dataset of having a positive label

Parameters

$dataset_test$

[MIMLDataset] Dataset to predict probabilities

1.4. mimlTOmi 11

results: np.ndarray of shape (n_instances, n_labels)

Predicted probabilities for given dataset

1.4.3 MIMLtoMILPClassifier

miml.classifier.mimlTOmi.
miml_to_mi_lp_classifier.
MIMLtoMILPClassifier

Class to represent a multi-instance classifier using a label powerset transformation

miml.classifier.mimlTOmi.miml_to_mi_lp_classifier.MIMLtoMILPClassifier

class miml.classifier.mimlTOmi.miml_to_mi_lp_classifier.MIMLtoMILPClassifier(mi_classifier)

Bases: MIMLtoMIClassifier

Class to represent a multi-instance classifier using a label powerset transformation

evaluate(*dataset_test*: MIMLDataset) → ndarray

Evaluate the model on a test dataset

Parameters

dataset test

[MIMLDataset] Test dataset to evaluate the model on

Returns

results

 $[ndarray\ of\ shape\ (n_bags,\ n_labels)]\ Predicted\ labels\ of\ dataset_test$

fit(*dataset_train*: MIMLDataset) → None

Training the classifier

Parameters

dataset train

[MIMLDataset] Dataset to train the classifier

fit_internal(*dataset_train*: MIMLDataset) → None

Training the classifier

Parameters

 $dataset_test$

```
dataset\_train: MIMLDataset
          Dataset to train the classifier
predict(x: ndarray) \rightarrow ndarray
     Predict labels of given data
     Parameters
     X
          [ndarray of shape (n_instances, n_features)] Data to predict their labels
     Returns
     results
          [ndarray of shape (n_labels)] Predicted labels
predict_bag(bag: Bag) \rightarrow ndarray
     Predict labels of a given bag
     Parameters
     bag
          [Bag] Bag to predict their labels
     Returns
     results
          [ndarray of shape (n_labels)] Predicted labels of the bag
predict_proba(dataset_test: MIMLDataset)
     Predict probabilities of given dataset of having a positive label
```

[MIMLDataset] Dataset to predict probabilities

1.4. mimITOmi 13

results: np.ndarray of shape (n_instances, n_features)

Predicted probabilities for given dataset

CHAPTER

TWO

DATA

2.1 Instance

miml.data.instance.Instance

Class to manage MIML Instance data representation

2.1.1 miml.data.instance.Instance

```
class miml.data.instance.Instance(values: list \mid None = None, bag=None)

Bases: object

Class to manage MIML Instance data representation

add_attribute(value=0, position=None) \rightarrow None

Add an attribute to the instance
```

Parameters

```
value
```

[float, default=0] Value for the attribute

position: int, default=NonePosition for the attribute

 $delete_attribute(position) \rightarrow None$

Delete an attribute of the instance

Parameters

position: int

Position of the attribute

 $\texttt{get_attribute}(\textit{attribute}) \rightarrow \texttt{float}$

Get value of an attribute of the instance

attribute

[int/str] Index/Name of the attribute

Returns

value

[float] Value of the attribute

$\texttt{get_attributes()} \rightarrow \texttt{ndarray}$

Get data attributes of the instance

Returns

attributes data: ndarray of shape (n_attributes)

Values of the attributes of the instance

$get_attributes_name() \rightarrow list[str]$

Get attributes name of the instance

Returns

attributes

[list[str]] Attributes name of the instance

get_features() → ndarray

Get features values of the instance

Returns

features data: ndarray of shape (n_features)

Values of the features of the instance

$\texttt{get_features_name()} \rightarrow list[str]$

Get features name of the instance

Returns

features

[list[str]] features name of the instance

$\textbf{get_labels()} \rightarrow \text{ndarray}$

Get labels values of the instance

16 Chapter 2. Data

labels data

[ndarray of shape (n_labels)] Values of the labels of the instance

$get_labels_name() \rightarrow list[str]$

Get labels name of the instance

Returns

labels

[list[str]] Labels name of the instance

${\tt get_number_attributes()} \to {\tt int}$

Get numbers of attributes of the instance

Returns

numbers of attributes: int

Numbers of attributes of the instance

get_number_features() → int

Get numbers of features of the instance

Returns

numbers of features: int

Numbers of features of the instance

${\tt get_number_labels()} \rightarrow {\tt int}$

Get numbers of labels of the instance

Returns

numbers of labels

[int] Numbers of labels of the instance

$\textbf{set_attribute}(\textit{attribute}, \textit{value}: \textit{float}) \rightarrow None$

Update value of an attribute of the instance

2.1. Instance 17

```
attribute [int/str] Index/Name of the attribute value [float] New value for the attribute set\_bag(bag) \rightarrow None Set the bag of the instance Parameters bag [Bag] Bag of the instance
```

Show instance info in table format

2.2 Bag

miml.data.bag.Bag

Class to manage MIML Bag data representation

2.2.1 miml.data.bag.Bag

 $\textbf{show_instance()} \rightarrow None$

```
class miml.data.bag.Bag(key: str)

Bases: object

Class to manage MIML Bag data representation

add_attribute(position: int, values=None) → None

Add attribute to the bag

Parameters

position
    [int] Index for the new attribute

values: array-like of shape (n_attributes)

Values for the new attribute. If not provided, new values would be zero

add_instance(instance: Instance) → None

Add instance to the bag
```

18 Chapter 2. Data

instance

[Instance] Instance to be added

$delete_attribute(position: int) \rightarrow None$

Delete attribute of the bag

Parameters

position

[int] Position of the attribute in the bag

$delete_instance(index: int) \rightarrow None$

Delete an instance of the bag

Parameters

index

[int] Index of the instance to be removed

$get_attribute(instance: int, attribute) \rightarrow float$

Get value of an attribute of the bag

Parameters

instance

[int] Index of the instance in the bag

attribute

[int/str] Index/Name of the attribute

Returns

value

[float] Value of the attribute

$\textbf{get_attributes()} \rightarrow \textbf{ndarray}$

Get attributes values of the bag

Returns

attributes data: ndarray of shape(n_instances, n_attributes)

Values of the attributes of the bag

$\texttt{get_attributes_name()} \rightarrow list[str]$

Get attributes name of the bag

2.2. Bag 19

attributes

[list[str]] Attributes name of the bag

get_features() → ndarray

Get features values of the bag

Returns

features data: ndarray of shape (n_instances, n_features)

Values of the features of the bag

$get_features_name() \rightarrow list[str]$

Get features name of the bag

Returns

features

[list[str]] Features name of the bag

$get_instance(index: int) \rightarrow Instance$

Get an Instance of the Bag

Parameters

index

[int] Index of the instance in the bag

Returns

instance

[Instance] Instance of Instance class

$get_labels() \rightarrow ndarray$

Get labels values of the bag

Returns

labels data

[ndarray of shape (n_instances, n_labels)] Values of the labels of the bag

$\texttt{get_labels_name()} \to list[str]$

Get labels name of the bag

20 Chapter 2. Data

labels

[list[str]] Labels name of the bag

$\texttt{get_number_attributes()} \rightarrow \mathsf{int}$

Get numbers of attributes of the bag

Returns

numbers of attributes: int

Numbers of attributes of the bag

${\tt get_number_features()} \to {\tt int}$

Get numbers of features of the bag

Returns

numbers of features: int

Numbers of features of the bag

$\texttt{get_number_instances}() \rightarrow \mathsf{int}$

Get numbers of instances of a bag

Returns

numbers of instances: int

Numbers of instances of a bag

${\tt get_number_labels()} \rightarrow {\tt int}$

Get numbers of labels of the bag

Returns

numbers of labels: int

Numbers of labels of the bag

 $set_attribute(instance: int, attribute, value: float) \rightarrow None$

Update value from attributes

2.2. Bag 21

instance

[int] Index of instance to be updated

attribute: int/str

Attribute name/index_instance of the bag to be updated

value: float

New value for the update

$set_dataset(dataset) \rightarrow None$

Set dataset which contains the bag

Parameters

dataset

[MIMLDataset] Dataset for the bag

show_bag($start: int = 0, end: int \mid None = None, attributes: list[<math>str$] | None = None, mode = 'table') \rightarrow None Show bag info in table format

Parameters

start

[int] Index of instance to start showing

end

[int] Index of instance to end showing

mode

[str] Mode to show the bag. Modes available are "table" and "csv" (csv format)

attributes

[list[str]] List of attributes to display. If empty, all attributes will be displayed.

2.3 MIMLDataset

miml.data.miml_dataset.MIMLDataset

Class to manage MIML data obtained from datasets

2.3.1 miml.data.miml_dataset.MIMLDataset

class miml.data.miml_dataset.MIMLDataset

Bases: object

Class to manage MIML data obtained from datasets

 $\textbf{add_attribute}(\textit{name: str, position: int} \mid \textit{None} = \textit{None, values: ndarray} \mid \textit{None} = \textit{None, feature: bool} = True) \rightarrow \textit{None}$

Add attribute to the dataset

22 Chapter 2. Data

name

[str] Name of the new attribute

position

[int, default = None] Index for the new attribute

values: ndarray of shape(n_instances)

Values for the new attribute

feature

[bool] Boolean value to determine if the attribute added is a feature or a label

$add_bag(bag: Bag) \rightarrow None$

Add a bag to the dataset

Parameters

bag

[Bag] Instance of Bag class to be added

$add_instance(bag, instance: Instance) \rightarrow None$

Add an Instance to a Bag of the dataset

Parameters

bag

[int/str] Index or key of the bag where the instance will be added

instance

[Instance] Instance of Instance class to be added

cardinality()

Computes the Cardinality as the average number of labels per pattern.

Returns

cardinality

[float] Average number of labels per pattern

$delete_attribute(position: int) \rightarrow None$

Delete attribute of the dataset

2.3. MIMLDataset 23

position

[int] Index of the attribute to be deleted

delete_bag($key_bag: str$) \rightarrow None

Delete a bag of the dataset

Parameters

key_bag

[str] Key of the bag to be deleted

$delete_instance(bag, index_instance: int) \rightarrow None$

Delete an instance of a bag of the dataset

Parameters

bag

[int/str] Index or key of the bag which contains the instance to be deleted

index_instance

[int] Index of the instance to be deleted

density()

Computes the density as the cardinality / numLabels.

Returns

density

[float] Cardinality divided by number of labels

describe()

Print statistics about the dataset

distinct()

Computes the numbers of labels combinations used in the dataset respect all the possible ones

Returns

distinct

[float] Numbers of labels combinations used in the dataset divided by all possible combinations

$\texttt{get_attribute}(bag, instance, attribute) \rightarrow \texttt{float}$

Get value of an attribute of the bag

24 Chapter 2. Data

```
bag
```

[str] Key of the bag which contains the attribute

instance

[int] Index of the instance in the bag

attribute

[int/str] Index/Name of the attribute

Returns

value

[float] Value of the attribute

get_attributes() → ndarray

Get attributes values of the dataset

Returns

attributes data: ndarray of shape (n_instances, n_attributes)

Values of the attributes of the dataset

$get_attributes_name() \rightarrow list[str]$

Get attributes name

Returns

attributes

[list[str]] Attributes name of the dataset

$\mathtt{get_bag}(bag) \to Bag$

Get data of a bag of the dataset

Parameters

bag: int/str

Index or key of the bag to be obtained

Returns

bag: Bag

Instance of Bag class

get_features() → ndarray

Get features values of the dataset

2.3. MIMLDataset 25

features: ndarray of shape (n_instances, n_features)

Values of the features of the dataset

get_features_by_bag() → ndarray

Get features values of the dataset by bag

Returns

features: ndarray of shape (n_bags, n_instances, n_features)

Values of the features of the dataset

$\texttt{get_features_name()} \rightarrow list[str]$

Get function for dataset features name

Returns

attributes

[list[str]] Attributes name of the dataset

$get_instance(bag, index_instance) \rightarrow Instance$

Get an Instance of the dataset

Parameters

bag

[int/str] Index/Key of the bag

index instance

[int] Index of the instance in the bag

Returns

instance

[Instance] Instance of Instance class

get_labels()

Get labels values of the dataset

Returns

labels: ndarray of shape (n_instances, n_labels)

Values of the labels of the dataset

get_labels_by_bag()

Get labels values of the dataset

26 Chapter 2. Data

labels

[ndarray of shape (n_bags, n_labels)] Values of the labels of the dataset

$get_labels_name() \rightarrow list[str]$

Get function for dataset labels name

Returns

labels

[list[str]] Labels name of the dataset

$\texttt{get_name()} \to str$

Get function for dataset name

Returns

name

[str] Name of the dataset

$get_number_attributes() \rightarrow int$

Get numbers of attributes of the bag

Returns

numbers of attributes: int

Numbers of attributes of the bag

${\tt get_number_bags()} \to {\rm int}$

Get numbers of bags of the dataset

Returns

numbers of bags: int

Numbers of bags of the dataset

$get_number_features() \rightarrow int$

Get numbers of attributes of the dataset

Returns

numbers of attributes: int

Numbers of attributes of the dataset

$\texttt{get_number_instances}() \rightarrow \mathsf{int}$

Get numbers of instances of the dataset

2.3. MIMLDataset 27

numbers of instances: int

Numbers of instances of the dataset

${\tt get_number_labels()} \rightarrow {\tt int}$

Get numbers of labels of the dataset

Returns

numbers of labels: int

Numbers of labels of the dataset

get_statistics()

Calculate statistics of the dataset

Returns

n instances

[int] Numbers of instances of the dataset

min_instances

[int] Number of instances in the bag with minimum number of instances

max instances

[int] Number of instances in the bag with maximum number of instances

distribution

[dict] Distribution of number of instances in bags

$save_arff(path) \rightarrow None$

Save MIMLDataset as arff file

Parameters

path

[str] Path to store the arff file

save_csv(path)

Save MIMLDataset as csv file

Parameters

path

[str] Path to store the csv file

 $set_attribute(bag, index_instance: int, attribute, value: float) \rightarrow None$

Update value from attributes

28 Chapter 2. Data

```
bag
```

[int/str] Index or key of the bag of the dataset

index_instance

[int] Index of the instance

attribute: int/str

Attribute of the dataset

value: float

New value for the update

$set_features_name(features: list[str]) \rightarrow None$

Set function for dataset features name

Parameters

features

[list[str]] List of the features name of the dataset

$set_labels_name(labels: list[str]) \rightarrow None$

Set function for dataset labels name

Parameters

labels: list[str]

List of the labels name of the dataset

```
set_name(name) \rightarrow None
```

Set function for dataset name

Parameters

name

[str] Name of the dataset

 $\textbf{show_dataset}(\textit{start: int} = 0, \textit{end: int} \mid \textit{None} = \textit{None}, \textit{attributes} = \textit{None}, \textit{mode: str} = 'table', \textit{info} = \textit{False}) \rightarrow \textit{None}$

Function to show information about the dataset

Parameters

start

[int] Index of bag to start showing

end

[int] Index of bag to end showing

attributes: List of stringAttributes to show

2.3. MIMLDataset 29

```
mode
```

[str] Mode to show the dataset. Modes available are "table" and "csv" (csv format)

info: Boolean

Show more info

split_dataset(*train_percentage: float* = 0.8, *seed*=0)

Split dataset in two parts, one for training and the other for test

Parameters

train_percentage

[float] Percentage of bags in train dataset

seed: int

Seed to generate random numbers

Returns

dataset_train

[MIMLDataset] Train dataset

dataset_test

[MIMLDataset] Test dataset

split_dataset_cv(folds: int = 4, seed=0)

CrossValidation K-Fold split of dataset

Parameters

folds

[int] Number of datasets

seed: int

Seed to generate random numbers

Returns

dataset_train

[list[MIMLDataset]] Datasets

2.4 Dataset utils

miml.data.dataset_utils

30 Chapter 2. Data

CHAPTER

THREE

REPORT

3.1 Report

miml.report.report.Report

Class to generate a report

3.1.1 miml.report.report.Report

Bases: object

Class to generate a report

 $calculate_metrics(beta: int = 0.5)$

Calculate metrics of the predicted data

Parameters

beta

[int, default = 0.5] Beta value for the fbeta_score function

 $to_csv(path: str \mid None = None, metrics: list[str] \mid None = None)$

Print/save data as csv format

Parameters

path

[str, default=None] Path to csv where the data would be stored

metrics

[list[str], default=None] List of metrics to show. If empty, show all metrics.

to_string(metrics: list[str] | None = None)

Print data as string format

metrics

[list[str], default=None] List of metrics to show. If empty, show all metrics.

32 Chapter 3. Report

CHAPTER

FOUR

TRANSFORMATION

4.1 mimITOmi

4.1.1 BinaryRelevanceTransformation

miml.transformation.mimlTOmi.
binary_relevance_transformation.
BinaryRelevanceTransformation

Class that performs a binary relevance transformation to convert a MIMLDataset class to numpy ndarray.

miml.transformation.mimlTOmi.binary relevance transformation.BinaryRelevanceTransformation

class miml.transformation.mimlTOmi.binary_relevance_transformation.
BinaryRelevanceTransformation

Bases: object

Class that performs a binary relevance transformation to convert a MIMLDataset class to numpy ndarray.

 $transform_bag(bag: Bag) \rightarrow list$

Transform miml bag to multi instance bags

Parameters

bag:

Bag to be transformed to multi-instance bag

Returns

bags

[list[Bag]] List of n_labels transformed bags

 $transform_dataset(dataset: MIMLDataset) \rightarrow list$

Transform the dataset to multi-instance datasets dividing the original dataset into n datasets with a single label, where n is the number of labels.

datasets: list

Multi instance datasets

4.1.2 LabelPowersetTransformation

miml.transformation.mimlTOmi.
label_powerset_transformation.
LabelPowersetTransformation

Class that performs a label powerset transformation.

miml.transformation.mimlTOmi.label powerset transformation.LabelPowersetTransformation

class

 $\verb|miml.transformation.mimlTOmi.label_powerset_transformation. LabelPowersetTransformation|\\$

Bases: object

Class that performs a label powerset transformation.

 $lp_to_ml_label(label: int) \rightarrow ndarray$

Transform lp label to multilabel

Parameters

label

Lp label to be transformed

Returns

labels

[np.ndarray] Multilabel labels

 $ml_to_lp_label(labels: ndarray) \rightarrow float$

Transform multilabel to lp label

Parameters

labels

[np.ndarray] Multilabel labels to be transformed

label

Lp label to be transformed

transform_bag($bag: Bag) \rightarrow Bag$

Transform miml bag to multi instance bags

Parameters

bag:

Bag to be transformed to multi-instance bag

Returns

transformed_bag

[Bag] Transformed bag

transform_dataset(dataset: MIMLDataset) → MIMLDataset

Transform the dataset to multi-instance dataset converting the labels to one label using binary to decimal codification

Returns

datasets: MIMLDataset
Multi instance dataset

4.2 mimITOml

4.2.1 ArithmeticTransformation

miml.transformation.mimlTOml.arithmetic.
ArithmeticTransformation

Class that performs an arithmetic transformation to convert a MIMLDataset class to numpy ndarray.

miml. transformation. miml TOml. arithmetic. A rithmetic Transformation

${\bf class} \ {\tt miml.transformation.mimlTOml.arithmetic}. {\bf ArithmeticTransformation}$

Bases: MIMLtoMLTransformation

Class that performs an arithmetic transformation to convert a MIMLDataset class to numpy ndarray.

transform_bag(bag: Bag)

Transform a bag to a multilabel instance

4.2. mimITOmI 35

bag

[Bag] Bag to transform

Returns

transformed_bag

[Bag] Transformed bag

$transform_dataset(dataset: MIMLDataset) \rightarrow MIMLDataset$

Transform the dataset to multilabel dataset converting each bag into a single instance being the value of each attribute the mean value of the instances in the bag.

Parameters

dataset

[MIMLDataset] Dataset to transform

Returns

transformed dataset

[MIMLDataset] Transformed dataset

4.2.2 GeometricTransformation

miml.transformation.	mimlTOml.	geometric.
GeometricTransformat.	ion	

Class that performs a geometric transformation to convert a MIMLDataset class to numpy ndarray.

miml.transformation.mimlTOml.geometric.GeometricTransformation

class miml.transformation.mimlTOml.geometric.GeometricTransformation

Bases: MIMLtoMLTransformation

Class that performs a geometric transformation to convert a MIMLDataset class to numpy ndarray.

 $transform_bag(bag: Bag) \rightarrow Bag$

Transform a bag to a multilabel instance

bag

[Bag] Bag to be transformed to multilabel instance

Returns

features

[ndarray of shape (n_features)] Numpy array with feature values

labels

[ndarray of shape (n_labels)] Numpy array with label values

$transform_dataset(dataset: MIMLDataset) \rightarrow MIMLDataset$

Transform the dataset to multilabel dataset converting each bag into a single instance being the value of each attribute the geometric center of the instances in the bag.

Parameters

dataset

[MIMLDataset] Dataset to transform

Returns

transformed dataset

[MIMLDataset] Transformed dataset

4.2.3 MinMaxTransformation

<pre>miml.transformation.mimlTOml.minmax.</pre>	Class that performs a minmax transformation to convert
MinMaxTransformation	a MIMLDataset class to numpy ndarray.

miml.transformation.mimlTOml.minmax.MinMaxTransformation

class miml.transformation.mimlTOml.minmax.MinMaxTransformation

Bases: MIMLtoMLTransformation

Class that performs a minmax transformation to convert a MIMLDataset class to numpy ndarray.

transform_bag(bag: Bag)

Transform a bag to a multilabel instance

4.2. mimlTOml 37

bag

[Bag] Bag to be transformed to multilabel instance

Returns

transformed_bag

[Bag] Transformed bag

transform_dataset(dataset: MIMLDataset)

Transform the dataset to multilabel dataset converting each bag into a single instance with the min and max value of each attribute as two new attributes.

Parameters

dataset

[MIMLDataset] Dataset to transform

Returns

transformed dataset

[MIMLDataset] Transformed dataset

4.2.4 MIMLtoMLTransformation

miml.transformation.mimlTOml.
miml_to_ml_transformation.
MIMLtoMLTransformation

Abstract class to represent a MIMLtoML Transformation

miml.transformation.mimlTOml.miml_to_ml_transformation.MIMLtoMLTransformation

class miml.transformation.mimlTOml.miml_to_ml_transformation.MIMLtoMLTransformation

Bases: ABC

Abstract class to represent a MIMLtoML Transformation

abstract transform_bag($bag: Bag) \rightarrow Bag$

abstract transform_dataset(*dataset*: MIMLDataset) → *MIMLDataset*

INDEX

A describe() (miml.data.miml_dataset.MIMLDataset add_attribute() (miml.data.bag.Bag method), 18 add_attribute() (miml.data.instance.Instance method), 15 add_attribute() (miml.data.miml_dataset.MIMLDataset method), 24 add_attribute() (miml.data.miml_dataset.MIMLDataset method), 22 add_bag() (miml.data.miml_dataset.MIMLDataset method), 23 add_instance() (miml.data.bag.Bag method), 18 describe() (miml.data.miml_dataset.MIMLDataset method), 24 distinct() (miml.data.miml_dataset.MIMLDataset method), 24 evaluate() (miml.classifier.miml_classifier.MIMLClassifier method), 4 add_instance() (miml.data.bag.Bag method), 18
add_attribute() (miml.data.instance.Instance method), 15 add_attribute() (miml.data.instance.Instance method), 24 add_attribute() (miml.data.miml_dataset.MIMLDataset method), 22 add_bag() (miml.data.miml_dataset.MIMLDataset method), 23 add_bag() (miml.data.miml_dataset.MIMLDataset method), 4
add_attribute() (miml.data.instance.Instance
add_attribute() (miml.data.miml_dataset.MIMLDataset E method), 22 add_bag() (miml.data.miml_dataset.MIMLDataset evaluate() (miml.classifier.miml_classifier.MIMLClassifier method), 23 method), 4
add_bag() (miml.data.miml_dataset.MIMLDataset evaluate() (miml.classifier.miml_classifier.MIMLClassifier method), 23 method), 4
·····//
add_instance() (miml.data.miml_dataset.MIMI.Datasetmethod), 10
method), 23 evaluate() (miml.classifier.mimlTOmi.miml_to_mi_classifier.MIMLtoM
APRClassifier (class in method), 8 miml.classifier.mi.apr_classifier), 1 evaluate() (miml.classifier.mimlTOmi.miml_to_mi_lp_classifier.MIMLto_mi_lp_cl
ArithmeticTransformation (class in method), 12 miml.transformation.mimlTOml.arithmetic), 35 evaluate() (miml.classifier.mimlTOml.miml_to_ml_classifier.MIMLtoM
method), 6
F
Bag (class in miml.data.bag), 18
BinaryRelevanceTransformation (class in fit() (miml.classifier.mi.apr_classifier.APRClassifier miml.transformation.mimlTOmi.binary_relevance_transformation), 1
fit() (miml.classifier.mi.mi_wrapper_classifier.MIWrapperClassifier method), 3
C fit() (miml.classifier.miml_classifier.MIMLClassifier
calculate_metrics() (miml.report.report.Report method), 5 method), 31 fit() (miml.classifier.mimlTOmi.miml_to_mi_br_classifier.MIMLtoMIBE
cardinality() (miml.data.miml_dataset.MIMLDataset method), 10 method), 23 method), 23 fit() (miml.classifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.mimlTOmi.miml_to_mi_classifier.miml_to_mi_classifier.mimlTOmi.miml_to_mi_classifier.miml_to
method), 23 ±1t() (miml.classifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.mimlTOmi.miml_to_mi_classifier.mimlTOmi.miml_to_mi_classifier.mimlTOmi.miml_to_mi_classifier.mimlTOmi.miml_to_mi_classifier.mimlTomi.miml_to_mi_classifier.mimlTomi.miml_to_mi_classifier.mimlTomi.miml_to_mi_classifier.mimlTomi.miml_to_mi_classifier.mimlTomi.miml_to_mi_classifier.mimlTomi.miml_to_mi_classifier.mimlTomi.miml_to_mi_classifier.mimlTomi.miml_to_mi_classifier.mimlTomi.miml_to_mi_classifier.miml_tomi.miml_tom
fit() (miml.classifier.mimlTOmi.miml_to_mi_lp_classifier.MIMLtoMILF
delete_attribute() (miml.data.bag.Bag method), 19 delete_attribute() (miml.data.bag.Bag method), 19 fit() (miml.classifier.mimlTOml.miml_to_ml_classifier.MIMLtoMLClass
method). 6
delete_attribute() (miml.data.miml_dataset.MIMLDataset_internal() (miml.classifier.miml_classifier.MIMLClassifier
delete_bag() (miml.data.miml_dataset.MIMLDataset fit_internal() (miml.classifier.mimlTOmi.miml_to_mi_br_classifier.M
delete instance() (miml data has Ras method) 19 fit_internal() (miml.classifier.mimlTOmi.miml_to_mi_classifier.MIM.
delete_instance() (miml.data.miml_dataset.MIMLDataset method), 9
delete_instance() (miml.data.miml_dataset.MIMLDataset method), 9 method), 24 method), 24 method), 24 method), 12 method), 12
density() (miml.data.miml_dataset.MIMLDataset method), 24 fit_internal() (miml.classifier.mimlTOml.miml_to_ml_classifier.MIM. method), 7

G	<pre>get_number_attributes()</pre>
GeometricTransformation (class in	method), 21
miml.transformation.mimlTOml.geometric), 36	<pre>get_number_attributes()</pre>
<pre>get_attribute() (miml.data.bag.Bag method), 19</pre>	(miml.data.instance.Instance method), 17
<pre>get_attribute()</pre>	<pre>get_number_attributes()</pre>
method), 15	(miml.data.miml_dataset.MIMLDataset
<pre>get_attribute() (miml.data.miml_dataset.MIMLDataset</pre>	method), 27
method), 24	<pre>get_number_bags() (miml.data.miml_dataset.MIMLDataset</pre>
<pre>get_attributes() (miml.data.bag.Bag method), 19</pre>	method), 27
<pre>get_attributes()</pre>	<pre>get_number_features() (miml.data.bag.Bag method),</pre>
get_attributes() (miml.data.miml_dataset.MIMLDataset	get_number_features() (miml.data.instance.Instance
method), 25	method), 17
<pre>get_attributes_name() (miml.data.bag.Bag method),</pre>	<pre>get_number_features()</pre>
19	(miml.data.miml_dataset.MIMLDataset
<pre>get_attributes_name() (miml.data.instance.Instance</pre>	method), 27
method), 16	get_number_instances() (miml.data.bag.Bag
<pre>get_attributes_name()</pre>	method), 21
(miml.data.miml_dataset.MIMLDataset	<pre>get_number_instances()</pre>
method), 25	(miml.data.miml_dataset.MIMLDataset
<pre>get_bag() (miml.data.miml_dataset.MIMLDataset</pre>	method), 27
method), 25	<pre>get_number_labels() (miml.data.bag.Bag method),</pre>
<pre>get_features() (miml.data.bag.Bag method), 20</pre>	21
<pre>get_features() (miml.data.instance.Instance method),</pre>	get_number_labels() (miml.data.instance.Instance
16	method), 17
<pre>get_features() (miml.data.miml_dataset.MIMLDataset</pre>	<pre>get_number_labels()</pre>
method), 25	method), 28
<pre>get_features_by_bag()</pre>	get_statistics() (miml.data.miml_dataset.MIMLDataset
(miml.data.miml_dataset.MIMLDataset	method), 28
method), 26	memoa), 28
<pre>get_features_name() (miml.data.bag.Bag method),</pre>	
20	Instance (class in miml.data.instance), 15
get_features_name() (miml.data.instance.Instance	instance (class in mimi.aaia.instance), 13
method), 16	
get_features_name()	LabelPowersetTransformation (class in
(miml.data.miml_dataset.MIMLDataset	`
method), 26	miml.transformation.mimlTOmi.label_powerset_transformation), 34
get_instance() (miml.data.bag.Bag method), 20	lp_to_ml_label() (miml.transformation.mimlTOmi.label_powerset_transformation.mimlTOmi.label_powe
method), 26	method), 34
get_labels() (miml.data.bag.Bag method), 20	memou), 54
get_labels() (miml.data.instance.Instance method), 16	M
get_labels() (miml.data.miml_dataset.MIMLDataset	MIMLClassifier (class in
method), 26	miml.classifier.miml_classifier), 4
get_labels_by_bag()	MIMLDataset (class in miml.data.miml_dataset), 22
(miml.data.miml_dataset.MIMLDataset	
method), 26	MIMLtoMIBRClassifier (class in miml.classifier.mimlTOmi.miml_to_mi_br_classifier),
get_labels_name() (miml.data.bag.Bag method), 20	munu.cuassijier.mimi i Omi.munu_io_mi_or_ciassijier), 10
get_labels_name() (miml.data.instance.Instance	
method), 17	MIMLtoMIClassifier (class in miml.classifier.mimlTOmi.miml_to_mi_classifier),
get_labels_name() (miml.data.miml_dataset.MIMLData	· · · · · · · · · · · · · · · · · · ·
method), 27	MIMLtoMILPClassifier (class in
get_name() (miml.data.miml_dataset.MIMLDataset	miml.classifier.mimlTOmi.miml_to_mi_lp_classifier),
method), 27	12

40 Index

```
R
 MIMLtoMLClassifier
                                                                                                                        (class
                                miml.classifier.mimlTOml.miml_to_ml_classifier) Report (class in miml.report.report), 31
MIMLtoMLTransformation
                                                                                                                                (class
                                (miml.data.miml_dataset.MIMLDataset
MinMaxTransformation
                                                                                                                            (class
                                                                                                                                                                                     in
                                                                                                                                                                                                    save_csv()
                                                                                                                                                                                                                                                             (miml.data.miml_dataset.MIMLDataset
                                 miml.transformation.mimlTOml.minmax),
                                                                                                                                                                                                                                    method), 28
                                                                                                                                                                                                    set_attribute() (miml.data.bag.Bag method), 21
MIWrapperClassifier
                                                                                                                          (class
                                                                                                                                                                                                    set_attribute()
                                                                                                                                                                                                                                                                                                 (miml.data.instance.Instance
                                miml.classifier.mi.mi_wrapper_classifier),
                                                                                                                                                                                                                                     method), 17
                                                                                                                                                                                                                     attribute() (miml.data.miml_dataset.MIMLDataset
erset_transformation.LabelPowersetTransformation
method), 28
\verb|ml_to_lp_label()| (miml.transformation.mimlTOmi.label\_power and all of the content of the co
                                 method), 34
                                                                                                                                                                                                    set_bag() (miml.data.instance.Instance method), 18
                                                                                                                                                                                                    set_dataset() (miml.data.bag.Bag method), 22
Р
                                                                                                                                                                                                    set_features_name()
predict() (miml.classifier.mi.apr_classifier.APRClassifier
                                                                                                                                                                                                                                     (miml.data.miml_dataset.MIMLDataset
                                 method), 2
                                                                                                                                                                                                                                    method), 29
predict() (miml.classifier.mi.mi_wrapper_classifier.MIWrapper_classifier.hame() (miml.data.miml_dataset.MIMLDataset
                                method), 3
                                                                                                                                                                                                                                     method), 29
\verb|predict()| (miml.classifier.miml\_classifier.MIMLClassifier_{\verb|set_name()|})|
                                                                                                                                                                                                                                                              (miml.data.miml_dataset.MIMLDataset
                                 method), 5
predict() (miml.classifier.mimlTOmi.miml_to_mi_br_classifier_MIMLto_MIBRGlassifier_Bag method), 22
                                method), 11
                                                                                                                                                                                                    show_dataset() (miml.data.miml_dataset.MIMLDataset
predict() (miml.classifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier
                                                                                                                                                                                                                                                                   (miml.data.instance.Instance
Classifier
                                 method), 9
predict() (miml.classifier.mimlTOmi.miml_to_mi_lp_classifier.MIMLtoMIL
                                method), 13
method), 13 split_dataset() (miml.data.miml_dataset.MIMLDataset predict() (miml.classifier.mimlTOml.miml_to_ml_classifier.MIMLtoMLClassifier

method), 13
                                 method), 7
                                                                                                                                                                                                    split_dataset_cv() (miml.data.miml_dataset.MIMLDataset
predict_bag() (miml.classifier.miml_classifier.MIMLClassifier
                                method), 5
predict_bag() (miml.classifier.mimlTOmi.miml_to_mi_br_classifier.MIMLtoMIBRClassifier
                                method), 11
predict_bag() (miml.classifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier.Report.Report method), 31

method) 9

to_csv() (miml.report.report.Report method), 31

method) 9
                                 method), 9
predict_bag() (miml.classifier.mimlTOmi.miml_to_mi_lp_classifier.MIMLtoM1LPClassifier

method) 13

method) 13
                                 method), 13
method), 13

transform_bag() (miml_transformation.mimlTOmi.label_powerset_transformation(), (miml_transformation.mimlTOmi.label_powerset_transformation(), 35
\verb|predict_proba()| (miml.classifier.mi.apr\_classifier.APRClassifier| bag()| (miml.transformation.mimlTOml.arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Arithmetic.Ar
                                method), 2
method), 2
transform_bag() (miml.transformation.mimlTOml.geometric.GeometricT
predict_proba() (miml.classifier.mi.mi_wrapper_classifier.MIWrapperClassifier
method), 36
\verb|predict_proba()| (miml.classifier.miml_classifier.MIMLClassifier)| transform_bag()| (miml.transformation.mimlTOml.miml_to_ml_transform_bag()| (miml.transformation.miml_to_ml_transform_bag()| (miml.transformation.miml_to_ml_transform_bag()| (miml.transformation.miml_to_ml_transform_bag()| (miml.transformation.miml_to_ml_transform_bag()| (miml.transform_bag()| (miml.tra
                                                                                                                                                                                                                                    method), 38
                                method), 5
predict_proba() (miml.classifier.mimlTOmi.miml_to_mi_br_classifier.MIMLtoMIBRClassifier

method) 11

method) 11
                                 method), 11
method), 11
transform_dataset()
predict_proba() (miml.classifier.mimlTOmi.miml_to_mi_classifier.MIMLtoMIClassifier
(miml.transformation.mimlTOmi.binary_relevance_transformatio
                                method), 9
predict_proba() (miml.classifier.mimlTOmi.miml_to_mi_lp_classifier_MIM_to_MILPClassifier_mimlTOmi.miml_to_mi_lp_classifier_MIM_to_MILPClassifier_mimlTOmi.miml_to_mi_lp_classifier_MIM_to_MILPClassifier_mimlTOmi.miml_to_mi_lp_classifier_MIM_to_MILPClassifier_mimlTOmi.miml_to_mi_lp_classifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MILPClassifier_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_to_MIM_t
predict_proba() (miml.classifier.mimlTOml.miml_to_ml_classifier.MIMLtoMf.Classifier

method) 7

method) 7
```

Index 41

method), 7

```
\label{transform_dataset} () \\ (\textit{miml.transformation.mimlTOml.arithmetic.ArithmeticTransformation} \\ \textit{method}), 36 \\ \text{transform_dataset}() \\ (\textit{miml.transformation.mimlTOml.geometric.GeometricTransformation} \\ \textit{method}), 37 \\ \text{transform_dataset}() \\ (\textit{miml.transformation.mimlTOml.miml\_to\_ml\_transformation.MIMLtoMLTransformation} \\ \textit{method}), 38 \\ \text{transform_dataset}() \\ (\textit{miml.transformation.mimlTOml.minmax.MinMaxTransformation} \\ \textit{method}), 38 \\ \end{aligned}
```

42 Index