Package 'iai'

July 18, 2019

Type Package

Title Interface to 'Interpretable AI' Modules

Version 1.0.0

Description An interface to the algorithms of 'Interpretable AI' <https://www.interpretable.ai> from the R programming language. 'Interpretable AI' provides various modules, including 'Optimal Trees' for classification, regression, prescription and survival analysis, 'Optimal Imputation' for missing data imputation and outlier detection, and 'Optimal Feature Selection' for exact sparse regression. The 'iai' package is an open-source project. The 'Interpretable AI' software modules are proprietary products, but free academic and evaluation licenses are available.

URL `https://www.interpretable.ai`

SystemRequirements Julia (>= 1.0) and Interpretable AI System Image (>= 1.0.0)

License MIT + file LICENSE

Imports JuliaCall, stringr, rlang

RoxygenNote 6.1.1

Suggests testthat, covr

NeedsCompilation no

Author Jack Dunn [aut, cre], Interpretable AI LLC [cph]

Maintainer Jack Dunn <jack@interpretable.ai>

Repository CRAN

Date/Publication 2019-07-18 06:36:08 UTC

R topics documented:

`apply` *apply*

Description

Return the leaf index in a tree model into which each point in the features falls

Usage

```
apply(lnr, X)
```

Arguments

`lnr`	The learner or grid to query.
`X`	The features of the data.

Details

Julia Equivalent: `IAI.apply`[1]

[1] `https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.apply`

Examples

```
## Not run: iai::apply(lnr, X)
```

`apply_nodes` *apply_nodes*

Description

Return the indices of the points in the features that fall into each node of a trained tree model

Usage

```
apply_nodes(lnr, X)
```

Arguments

`lnr`	The learner or grid to query.
`X`	The features of the data.

Details

Julia Equivalent: `IAI.apply_nodes`[2]

Examples

```
## Not run: iai::apply_nodes(lnr, X)
```

`as.mixeddata` *as.mixeddata*

Description

Convert a vector of values to IAI mixed data format

Usage

```
as.mixeddata(values, categorical_levels, ordinal_levels = c())
```

[2] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.apply_nodes

Arguments

`values`	The vector of values to convert
`categorical_levels`	The values in `values` to treat as categoric levels
`ordinal_levels`	(optional) The values in `values` to treat as ordinal levels, in the order supplied

Details

Julia Equivalent: `IAI.make_mixed_data`[3]

Examples

```
df <- iris
set.seed(1)
df$mixed <- rnorm(150)
df$mixed[1:5] <- NA  # Insert some missing values
df$mixed[6:10] <- "Not graded"
iai::iai_setup()
df$mixed <- iai::as.mixeddata(df$mixed, c("Not graded"))
```

`clone` *clone*

Description

Return an unfitted copy of a learner with the same parameters

Usage

```
clone(lnr)
```

Arguments

`lnr`	The learner to copy.

Details

Julia Equivalent: `IAI.clone`[4]

Examples

```
## Not run: new_lnr <- iai::clone(lnr)
```

[3] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.make_mixed_data
[4] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.clone

decision_path *decision_path*

Description

Return a matrix where entry `(i, j)` is true if the `i`th point in the features passes through the `j`th node in a trained tree model.

Usage

```
decision_path(lnr, X)
```

Arguments

`lnr`	The learner or grid to query.
`X`	The features of the data.

Details

Julia Equivalent: `IAI.decision_path`[5]

Examples

```
## Not run: iai::decision_path(lnr, X)
```

delete_rich_output_param *delete_rich_output_param*

Description

Delete a global rich output parameter

Usage

```
delete_rich_output_param(key)
```

Arguments

`key`	The parameter to delete.

[5] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.decision_path

Details

Julia Equivalent: `IAI.delete_rich_output_param!`[6]

Examples

```
## Not run: iai::delete_rich_output_param("simple_layout")
```

`fit` *fit*

Description

Fits a model to the training data

Usage

```
fit(lnr, X, ...)
```

Arguments

`lnr`	The learner or grid to fit.
`X`	The features of the data.
`...`	Other parameters, including zero or more target vectors as required by the problem type. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.fit!`[7]

Examples

```
X <- iris[, 1:4]
y <- iris$Species
iai::iai_setup()
grid <- iai::grid_search(
    iai::optimal_tree_classifier(max_depth = 1),
)
iai::fit(grid, X, y)
```

[6] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.delete_rich_output_param!

[7] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.fit!

fit_cv *fit_cv*

Description

Fits a grid search to the training data with cross-validation

Usage

```
fit_cv(grid, X, ...)
```

Arguments

grid	The grid to fit.
X	The features of the data.
...	Other parameters, including zero or more target vectors as required by the problem type. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.fit_cv!`[8]

Examples

```
X <- iris[, 1:4]
y <- iris$Species
iai::iai_setup()
grid <- iai::grid_search(
    iai::optimal_tree_classifier(max_depth = 1),
)
iai::fit_cv(grid, X, y)
```

fit_transform *fit_transform*

Description

Fit an imputation model using the given features and impute the missing values in these features

Usage

```
fit_transform(lnr, X, ...)
```

[8] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.fit_cv!

Arguments

`lnr`	The learner or grid to use for imputation
`X`	The features of the data.
`...`	Refer to the Julia documentation for available parameters.

Details

Similar to calling `fit` followed by `transform`

Julia Equivalent: `IAI.fit_transform!`[9]

Examples

```
X <- iris
X[1, 1] <- NA
iai::iai_setup()
grid <- iai::grid_search(
    iai::imputation_learner(),
    method = c("opt_knn", "opt_tree"),
)
iai::fit_transform(grid, X)
```

`fit_transform_cv` *fit_transform_cv*

Description

Train a grid using cross-validation with features and impute all missing values in these features

Usage

```
fit_transform_cv(grid, X, ...)
```

Arguments

`grid`	The grid to use for imputation
`X`	The features of the data.
`...`	Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.fit_transform_cv!`[10]

[9] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.fit_transform!
[10] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.fit_transform_cv!

Examples

```
X <- iris
X[1, 1] <- NA
iai::iai_setup()
grid <- iai::grid_search(
    iai::imputation_learner(),
    method = c("opt_knn", "opt_tree"),
)
iai::fit_transform_cv(grid, X)
```

---

get_best_params *get_best_params*

Description

Return the best parameter combination from a grid

Usage

```
get_best_params(grid)
```

Arguments

grid	The grid search to query.

Details

Julia Equivalent: `IAI.get_best_params`[11]

Examples

```
## Not run: iai::get_best_params(grid)
```

[11] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.get_best_params

`get_classification_label` *get_classification_label*

Description

Return the predicted label at a node of a tree

Usage

```
get_classification_label(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_classification_label`[12]

Examples

```
## Not run: iai::get_classification_label(lnr, 1)
```

`get_classification_proba` *get_classification_proba*

Description

Return the predicted probabilities of class membership at a node of a tree

Usage

```
get_classification_proba(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

[12] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_classification_label

Details

Julia Equivalent: `IAI.get_classification_proba`[13]

Examples

```
## Not run: iai::get_classification_proba(lnr, 1)
```

`get_depth` *get_depth*

Description

Get the depth of a node of a tree

Usage

```
get_depth(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_depth`[14]

Examples

```
## Not run: iai::get_depth(lnr, 1)
```

[13] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_classification_proba

[14] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_depth

get_grid_results *get_grid_results*

Description

Return a summary of the results from the grid search

Usage

```
get_grid_results(grid)
```

Arguments

grid	The grid search to query.

Details

Julia Equivalent: `IAI.get_grid_results`[15]

Examples

```
## Not run: iai::get_grid_results(grid)
```

get_learner *get_learner*

Description

Return the fitted learner using the best parameter combination from a grid

Usage

```
get_learner(grid)
```

Arguments

grid	The grid to query.

Details

Julia Equivalent: `IAI.get_learner`[16]

Examples

```
## Not run: lnr <- iai::get_learner(grid)
```

[15] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.get_grid_results
[16] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.get_learner

get_lower_child *get_lower_child*

Description

Get the index of the lower child at a split node of a tree

Usage

```
get_lower_child(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_lower_child`[17]

Examples

```
## Not run: iai::get_lower_child(lnr, 1)
```

get_num_nodes *get_num_nodes*

Description

Return the number of nodes in a trained learner

Usage

```
get_num_nodes(lnr)
```

Arguments

lnr	The learner or grid to query.

Details

Julia Equivalent: `IAI.get_num_nodes`[18]

[17] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_lower_child
[18] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_num_nodes

Examples

```
## Not run: iai::get_num_nodes(lnr)
```

`get_num_samples` *get_num_samples*

Description

Get the number of training points contained in a node of a tree

Usage

```
get_num_samples(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_num_samples`[19]

Examples

```
## Not run: iai::get_num_samples(lnr, 1)
```

`get_params` *get_params*

Description

Return the value of all parameters on a learner

Usage

```
get_params(lnr)
```

Arguments

`lnr`	The learner to query.

[19] `https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_num_samples`

Details

Julia Equivalent: `IAI.get_params`[20]

Examples

```
## Not run: iai::get_params(lnr)
```

`get_parent` *get_parent*

Description

Get the index of the parent node at a node of a tree

Usage

```
get_parent(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_parent`[21]

Examples

```
## Not run: iai::get_parent(lnr, 2)
```

[20] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.get_params
[21] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_parent

get_prescription_treatment_rank
get_prescription_treatment_rank

Description

Return the treatments ordered from most effective to least effective at a node of a tree

Usage

```
get_prescription_treatment_rank(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_prescription_treatment_rank`[22]

Examples

```
## Not run: iai::get_prescription_treatment_rank(lnr, 1)
```

get_regression_constant
get_regression_constant

Description

Return the constant term in the regression prediction at a node of a tree

Usage

```
get_regression_constant(lnr, node_index, ...)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.
...	If a prescription problem, the treatment to query.

[22] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_prescription_treatment_rank

Details

Julia Equivalent: `IAI.get_regression_constant`[23] (for regression or prescription tree learners as appropriate)

Examples

```
## Not run:
iai::get_regression_constant(lnr, 1)
iai::get_regression_constant(lnr, 1, "A")

## End(Not run)
```

`get_regression_weights` *get_regression_weights*

Description

Return the weights for each feature in the regression prediction at a node of a tree

Usage

```
get_regression_weights(lnr, node_index, ...)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.
`...`	If a prescription problem, the treatment to query.

Details

Julia Equivalent: `IAI.get_regression_weights`[24] (for regression or prescription tree learners as appropriate)

Examples

```
## Not run:
iai::get_regression_weights(lnr, 1)
iai::get_regression_weights(lnr, 1, "A")

## End(Not run)
```

[23] https://docs.interpretable.ai/IAITrees/stable/reference/
[24] https://docs.interpretable.ai/IAITrees/stable/reference/

`get_rich_output_params` *get_rich_output_params*

Description

Return the current global rich output parameter settings

Usage

```
get_rich_output_params()
```

Details

Julia Equivalent: `IAI.get_rich_output_params`[25]

Examples

```
## Not run: iai::get_rich_output_params()
```

`get_split_categories` *get_split_categories*

Description

Return the categoric/ordinal information used in the split at a node of a tree

Usage

```
get_split_categories(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_split_categories`[26]

Examples

```
## Not run: iai::get_split_categories(lnr, 1)
```

[25] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.get_rich_output_params
[26] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_split_categories

get_split_feature *get_split_feature*

Description

Return the feature used in the split at a node of a tree

Usage

```
get_split_feature(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_split_feature`[27]

Examples

```
## Not run: iai::get_split_feature(lnr, 1)
```

get_split_threshold *get_split_threshold*

Description

Return the threshold used in the split at a node of a tree

Usage

```
get_split_threshold(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

[27] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_split_feature

Details

Julia Equivalent: `IAI.get_split_threshold`[28]

Examples

```
## Not run: iai::get_split_threshold(lnr, 1)
```

`get_split_weights` *get_split_weights*

Description

Return the weights for numeric and categoric features used in the hyperplane split at a node of a tree

Usage

```
get_split_weights(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_split_weights`[29]

Examples

```
## Not run: iai::get_split_weights(lnr, 1)
```

[28] `https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_split_threshold`
[29] `https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_split_weights`

get_survival_curve *get_survival_curve*

Description

Return the survival curve at a node of a tree

Usage

```
get_survival_curve(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_survival_curve`[30]

Examples

```
## Not run: iai::get_survival_curve(lnr, 1)
```

get_survival_curve_data *get_survival_curve_data*

Description

Extract the underlying data from a survival curve (as returned by `predict` or `get_survival_curve`)

Usage

```
get_survival_curve_data(curve)
```

Arguments

curve	The curve to query.

[30] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_survival_curve

Details

The data is returned as a list with two keys: `times` containing the time for each breakpoint on the curve, and `coefs` containing the probability for each breakpoint on the curve.

Julia Equivalent: `IAI.get_survival_curve_data`[31]

Examples

```
## Not run: iai::get_survival_curve_data(curve)
```

`get_upper_child` *get_upper_child*

Description

Get the index of the upper child at a split node of a tree

Usage

```
get_upper_child(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

Details

Julia Equivalent: `IAI.get_upper_child`[32]

Examples

```
## Not run: iai::get_upper_child(lnr, 1)
```

[31] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.get_survival_curve_data
[32] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.get_upper_child

grid_search *grid_search*

Description

Controls grid search over parameter combinations

Usage

```
grid_search(lnr, ...)
```

Arguments

lnr	The learner to use when validating.
...	The parameters to validate over.

Details

Julia Equivalent: `IAI.GridSearch`[33]

Examples

```
iai::iai_setup()
grid <- iai::grid_search(
    iai::optimal_tree_classifier(
        random_seed = 1,
    ),
    max_depth = 1:5,
)
```

iai_setup *iai_setup*

Description

Initialize Julia and the IAI package. This needs to be done in every R session before calling 'iai' functions

Usage

```
iai_setup(...)
```

Arguments

...	All parameters are passed through to `JuliaCall::julia_setup`[34]

[33] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.GridSearch
[34] https://www.rdocumentation.org/packages/JuliaCall/topics/julia_setup

Examples

```
## Not run: iai::iai_setup()
```

---

imputation_learner *imputation_learner*

Description

Generic learner for imputing missing values

Usage

```
imputation_learner(method = "opt_knn", ...)
```

Arguments

`method`	(optional) Specifies the imputation method to use.
`...`	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.ImputationLearner`[35]

Examples

```
## Not run: lnr <- iai::imputation_learner(method = "opt_tree")
```

---

impute *impute*

Description

Impute missing values using either a specified method or through validation

Usage

```
impute(X, ...)
```

Arguments

`X`	The dataframe in which to impute missing values.
`...`	Refer to the Julia documentation for available parameters.

[35] https://docs.interpretable.ai/OptImpute/stable/reference/#IAI.ImputationLearner

Details

Julia Equivalent: `IAI.impute`[36]

Examples

```
X <- iris
X[1, 1] <- NA
iai::iai_setup()
iai::impute(X)
```

`impute_cv` *impute_cv*

Description

Impute missing values using cross validation

Usage

```
impute_cv(X, ...)
```

Arguments

`X`	The dataframe in which to impute missing values.
`...`	Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.impute_cv`[37]

Examples

```
X <- iris
X[1, 1] <- NA
iai::iai_setup()
iai::impute_cv(X, list(method = c("opt_knn", "opt_tree")))
```

[36] https://docs.interpretable.ai/OptImpute/stable/reference/#IAI.impute
[37] https://docs.interpretable.ai/OptImpute/stable/reference/#IAI.impute_cv

is_categoric_split *is_categoric_split*

Description

Check if a node of a tree applies a categoric split

Usage

```
is_categoric_split(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

Details

Julia Equivalent: `IAI.is_categoric_split`[38]

Examples

```
## Not run: iai::is_categoric_split(lnr, 1)
```

is_hyperplane_split *is_hyperplane_split*

Description

Check if a node of a tree applies a hyperplane split

Usage

```
is_hyperplane_split(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

[38] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.is_categoric_split

Details

Julia Equivalent: `IAI.is_hyperplane_split`[39]

Examples

```
## Not run: iai::is_hyperplane_split(lnr, 1)
```

`is_leaf` *is_leaf*

Description

Check if a node of a tree is a leaf

Usage

```
is_leaf(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

Details

Julia Equivalent: `IAI.is_leaf`[40]

Examples

```
## Not run: iai::is_leaf(lnr, 1)
```

[39] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.is_hyperplane_split
[40] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.is_leaf

is_mixed_ordinal_split *is_mixed_ordinal_split*

Description

Check if a node of a tree applies a mixed ordinal/categoric split

Usage

```
is_mixed_ordinal_split(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

Details

Julia Equivalent: `IAI.is_mixed_ordinal_split`[41]

Examples

```
## Not run: iai::is_mixed_ordinal_split(lnr, 1)
```

is_mixed_parallel_split *is_mixed_parallel_split*

Description

Check if a node of a tree applies a mixed parallel/categoric split

Usage

```
is_mixed_parallel_split(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

[41] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.is_mixed_ordinal_split

Details

Julia Equivalent: `IAI.is_mixed_parallel_split`[42]

Examples

```
## Not run: iai::is_mixed_parallel_split(lnr, 1)
```

`is_ordinal_split` *is_ordinal_split*

Description

Check if a node of a tree applies a ordinal split

Usage

```
is_ordinal_split(lnr, node_index)
```

Arguments

`lnr`	The learner or grid to query.
`node_index`	The node in the tree to query.

Details

Julia Equivalent: `IAI.is_ordinal_split`[43]

Examples

```
## Not run: iai::is_ordinal_split(lnr, 1)
```

[42] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.is_mixed_parallel_split

[43] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.is_ordinal_split

is_parallel_split *is_parallel_split*

Description

Check if a node of a tree applies a parallel split

Usage

```
is_parallel_split(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

Details

Julia Equivalent: `IAI.is_parallel_split`[44]

Examples

```
## Not run: iai::is_parallel_split(lnr, 1)
```

mean_imputation_learner *mean_imputation_learner*

Description

Learner for conducting mean imputation

Usage

```
mean_imputation_learner(...)
```

Arguments

...	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.MeanImputationLearner`[45]

[44] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.is_parallel_split
[45] https://docs.interpretable.ai/OptImpute/stable/reference/#IAI.MeanImputationLearner

Examples

```
## Not run: lnr <- iai::mean_imputation_learner()
```

---

missing_goes_lower *missing_goes_lower*

Description

Check if points with missing values go to the lower child at a split node of of a tree

Usage

```
missing_goes_lower(lnr, node_index)
```

Arguments

lnr	The learner or grid to query.
node_index	The node in the tree to query.

Details

Julia Equivalent: `IAI.missing_goes_lower`[46]

Examples

```
## Not run: iai::missing_goes_lower(lnr, 1)
```

---

optimal_tree_classifier *optimal_tree_classifier*

Description

Learner for training Optimal Classification Trees

Usage

```
optimal_tree_classifier(...)
```

Arguments

...	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

[46] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.missing_goes_lower

Details

Julia Equivalent: `IAI.OptimalTreeClassifier`[47]

Examples

```
## Not run: lnr <- iai::optimal_tree_classifier()
```

`optimal_tree_prescription_maximizer`
optimal_tree_prescription_maximizer

Description

Learner for training Optimal Prescriptive Trees where the prescriptions should aim to maximize outcomes

Usage

```
optimal_tree_prescription_maximizer(...)
```

Arguments

`...`	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.OptimalTreePrescriptionMaximizer`[48]

Examples

```
## Not run: lnr <- iai::optimal_tree_prescription_maximizer()
```

[47] https://docs.interpretable.ai/OptimalTrees/stable/reference/#IAI.OptimalTreeClassifier

[48] https://docs.interpretable.ai/OptimalTrees/stable/reference/#IAI.OptimalTreePrescriptionMaximizer

---

`optimal_tree_prescription_minimizer`
optimal_tree_prescription_minimizer

Description

Learner for training Optimal Prescriptive Trees where the prescriptions should aim to minimize outcomes

Usage

```
optimal_tree_prescription_minimizer(...)
```

Arguments

`...`	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.OptimalTreePrescriptionMinimizer`[49]

Examples

```
## Not run: lnr <- iai::optimal_tree_prescription_minimizer()
```

`optimal_tree_regressor`
optimal_tree_regressor

Description

Learner for training Optimal Regression Trees

Usage

```
optimal_tree_regressor(...)
```

Arguments

`...`	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

[49] `https://docs.interpretable.ai/OptimalTrees/stable/reference/#IAI.OptimalTreePrescriptionMinimizer`

Details

Julia Equivalent: `IAI.OptimalTreeRegressor`[50]

Examples

```
## Not run: lnr <- iai::optimal_tree_regressor()
```

`optimal_tree_survivor` *optimal_tree_survivor*

Description

Learner for training Optimal Survival Trees

Usage

```
optimal_tree_survivor(...)
```

Arguments

`...`	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.OptimalTreeSurvivor`[51]

Examples

```
## Not run: lnr <- iai::optimal_tree_survivor()
```

[50] https://docs.interpretable.ai/OptimalTrees/stable/reference/#IAI.OptimalTreeRegressor

[51] https://docs.interpretable.ai/OptimalTrees/stable/reference/#IAI.OptimalTreeSurvivor

opt_knn_imputation_learner
opt_knn_imputation_learner

Description

Learner for conducting optimal k-NN imputation

Usage

```
opt_knn_imputation_learner(...)
```

Arguments

`...`	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.OptKNNImputationLearner`[52]

Examples

```
## Not run: lnr <- iai::opt_knn_imputation_learner()
```

opt_svm_imputation_learner
opt_svm_imputation_learner

Description

Learner for conducting optimal SVM imputation

Usage

```
opt_svm_imputation_learner(...)
```

Arguments

`...`	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

[52] https://docs.interpretable.ai/OptImpute/stable/reference/#IAI.OptKNNImputationLearner

Details

Julia Equivalent: `IAI.OptSVMImputationLearner`[53]

Examples

```
## Not run: lnr <- iai::opt_svm_imputation_learner()
```

`opt_tree_imputation_learner`
opt_tree_imputation_learner

Description

Learner for conducting optimal tree-based imputation

Usage

```
opt_tree_imputation_learner(...)
```

Arguments

`...`	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.OptTreeImputationLearner`[54]

Examples

```
## Not run: lnr <- iai::opt_tree_imputation_learner()
```

[53] https://docs.interpretable.ai/OptImpute/stable/reference/#IAI.OptSVMImputationLearner

[54] https://docs.interpretable.ai/OptImpute/stable/reference/#IAI.OptTreeImputationLearner

predict *predict*

Description

Return the predictions made by the model for each point in the features

Usage

```
predict(lnr, X)
```

Arguments

lnr	The learner or grid to use for prediction.
X	The features of the data.

Details

Julia Equivalent: `IAI.predict`[55]

Examples

```
## Not run: iai::predict(lnr, X)
```

predict_outcomes *predict_outcomes*

Description

Return the the predicted outcome for each treatment made by a model for each point in the features

Usage

```
predict_outcomes(lnr, X)
```

Arguments

lnr	The learner or grid to use for prediction.
X	The features of the data.

Details

Julia Equivalent: `IAI.predict_outcomes`[56]

[55] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.predict
[56] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.predict_outcomes

Examples

```
## Not run: iai::predict_outcomes(lnr, X)
```

`predict_proba` *predict_proba*

Description

Return the probabilities of class membership predicted by a model for each point in the features

Usage

```
predict_proba(lnr, X)
```

Arguments

`lnr`	The learner or grid to use for prediction.
`X`	The features of the data.

Details

Julia Equivalent: `IAI.predict_proba`[57]

Examples

```
## Not run: iai::predict_proba(lnr, X)
```

`print_path` *print_path*

Description

Print the decision path through the learner for each sample in the features

Usage

```
print_path(lnr, X, ...)
```

Arguments

`lnr`	The learner or grid to query.
`X`	The features of the data.
`...`	Refer to the Julia documentation for available parameters.

[57] `https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.predict_proba`

Details

Julia Equivalent: `IAI.print_path`[58]

Examples

```
## Not run:
iai::print_path(lnr, X)
iai::print_path(lnr, X, 1)

## End(Not run)
```

`rand_imputation_learner` *rand_imputation_learner*

Description

Learner for conducting random imputation

Usage

```
rand_imputation_learner(...)
```

Arguments

`...`	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.RandImputationLearner`[59]

Examples

```
## Not run: lnr <- iai::rand_imputation_learner()
```

[58] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.print_path
[59] https://docs.interpretable.ai/OptImpute/stable/reference/#IAI.RandImputationLearner

read_json *read_json*

Description

Read in a learner saved in JSON format

Usage

```
read_json(filename)
```

Arguments

filename	The location of the JSON file.

Details

Julia Equivalent: `IAI.read_json`[60]

Examples

```
## Not run: lnr <- iai::read_json("out.json")
```

reset_display_label *reset_display_label*

Description

Reset the predicted probability displayed to be that of the predicted label when visualizing a learner

Usage

```
reset_display_label(lnr)
```

Arguments

lnr	The learner to modify.

Details

Julia Equivalent: `IAI.reset_display_label!`[61]

[60] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.read_json
[61] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.reset_display_label!

Examples

```
## Not run: iai::reset_display_label(lnr)
```

`roc_curve` *roc_curve*

Description

Construct an ROC curve using a trained model on the given data

Usage

```
roc_curve(lnr, X, y)
```

Arguments

`lnr`	The learner or grid to use for prediction.
`X`	The features of the data.
`y`	The labels of the data.

Details

Julia Equivalent: `IAI.ROCCurve`[62]

Examples

```
## Not run: iai::roc_curve(lnr, X, y)
```

`score` *score*

Description

Calculate the score for a model on the given data

Usage

```
score(lnr, X, ...)
```

[62] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.ROCCurve

Arguments

`lnr`	The learner or grid to evaluate.
`X`	The features of the data.
`...`	Other parameters, including zero or more target vectors as required by the problem type. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.score`[63]

Examples

```
## Not run: iai::score(lnr, X, y)
```

`set_display_label` *set_display_label*

Description

Show the probability of a specified label when visualizing a learner

Usage

```
set_display_label(lnr, display_label)
```

Arguments

`lnr`	The learner to modify.
`display_label`	The label for which to show probabilities.

Details

Julia Equivalent: `IAI.set_display_label!`[64]

Examples

```
## Not run: iai::set_display_label(lnr, "A")
```

[63] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.score
[64] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.set_display_label!

set_julia_seed *set_julia_seed*

Description

Set the random seed in Julia

Usage

```
set_julia_seed(seed)
```

Arguments

seed	The seed to set

Details

Julia Equivalent: `Random.seed!`[65]

Examples

```
## Not run: iai::set_julia_seed(1)
```

set_params *set_params*

Description

Set all supplied parameters on a learner

Usage

```
set_params(lnr, ...)
```

Arguments

lnr	The learner to modify.
...	The parameters to set on the learner.

Details

Julia Equivalent: `IAI.set_params!`[66]

[65] https://docs.julialang.org/en/v1/stdlib/Random/index.html#Random.seed!
[66] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.set_params!

Examples

```
## Not run: iai::set_params(lnr, random_seed = 1)
```

`set_rich_output_param` *set_rich_output_param*

Description

Sets a global rich output parameter

Usage

```
set_rich_output_param(key, value)
```

Arguments

`key`	The parameter to set.
`value`	The value to set

Details

Julia Equivalent: `IAI.set_rich_output_param!`[67]

Examples

```
## Not run: iai::set_rich_output_param("simple_layout", TRUE)
```

`set_threshold` *set_threshold*

Description

For a binary classification problem, update the the predicted labels in the leaves of the learner to predict a label only if the predicted probability is at least the specified threshold.

Usage

```
set_threshold(lnr, label, threshold, ...)
```

[67] `https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.set_rich_output_param!`

Arguments

lnr	The learner or grid to modify.
label	The referenced label.
threshold	The probability threshold above which label will be be predicted.
...	Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: IAI.set_threshold![68]

Examples

```
## Not run: iai::set_threshold(lnr, "A", 0.4)
```

---

show_in_browser *show_in_browser*

Description

Show interactive visualization of an object (such as a learner or curve) in the default browser

Usage

```
show_in_browser(obj, ...)
```

Arguments

obj	The object to visualize.
...	Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: IAI.show_in_browser[69]

Examples

```
## Not run: iai::show_in_browser(lnr)
```

[68] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.set_threshold!
[69] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.show_in_browser

show_questionnaire *show_questionnaire*

Description

Show an interactive questionnaire based on a learner in default browser

Usage

```
show_questionnaire(lnr, ...)
```

Arguments

lnr	The learner to visualize.
...	Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.show_questionnaire`[70]

Examples

```
## Not run: iai::show_questionnaire(lnr)
```

single_knn_imputation_learner *single_knn_imputation_learner*

Description

Learner for conducting heuristic k-NN imputation

Usage

```
single_knn_imputation_learner(...)
```

Arguments

...	Use keyword arguments to set parameters on the resulting learner. Refer to the Julia documentation for available parameters.

[70] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.show_questionnaire

Details

Julia Equivalent: `IAI.SingleKNNImputationLearner`[71]

Examples

```
## Not run: lnr <- iai::single_knn_imputation_learner()
```

---

split_data *split_data*

Description

Split the data into training and test datasets

Usage

```
split_data(task, X, ...)
```

Arguments

`task`	The type of problem.
`X`	The features of the data.
`...`	Other parameters, including zero or more target vectors as required by the problem type. Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.split_data`[72]

Examples

```
X <- iris[, 1:4]
y <- iris$Species
iai::iai_setup()
split <- iai::split_data("classification", X, y, train_proportion = 0.75)
train_X <- split[[1]][[1]]
train_y <- split[[1]][[2]]
test_X <- split[[2]][[1]]
test_y <- split[[2]][[2]]
```

[71] https://docs.interpretable.ai/OptImpute/stable/reference/#IAI.SingleKNNImputationLearner

[72] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.split_data

transform *transform*

Description

Impute missing values in a dataframe using a fitted imputation model

Usage

```
transform(lnr, X)
```

Arguments

lnr	The learner or grid to use for imputation
X	The features of the data.

Details

Julia Equivalent: `IAI.transform`[73]

Examples

```
## Not run: iai::transform(lnr, X)
```

variable_importance *variable_importance*

Description

Generate a ranking of the variables in the learner according to their importance when training the trees

Usage

```
variable_importance(lnr)
```

Arguments

lnr	The learner or grid to query.

Details

Julia Equivalent: `IAI.variable_importance`[74]

[73] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.transform
[74] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.variable_importance

Examples

```
## Not run: iai::variable_importance(lnr)
```

`write_dot` *write_dot*

Description

Output a learner in .dot format[75]

Usage

```
write_dot(filename, lnr, ...)
```

Arguments

`filename`	Where to save the output.
`lnr`	The learner or grid to output.
`...`	Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.write_dot`[76]

Examples

```
## Not run: iai::write_dot(file.path(tempdir(), "tree.dot"), lnr)
```

`write_html` *write_html*

Description

Output a learner as an interactive browser visualization in HTML format

Usage

```
write_html(filename, lnr, ...)
```

[75] http://www.graphviz.org/content/dot-language/
[76] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.write_dot

Arguments

filename	Where to save the output.
lnr	The learner or grid to output.
...	Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.write_html`[77]

Examples

```
## Not run: iai::write_html(file.path(tempdir(), "tree.html"), lnr)
```

---

write_json *write_json*

Description

Output a learner in JSON format

Usage

```
write_json(filename, lnr, ...)
```

Arguments

filename	Where to save the output.
lnr	The learner to output.
...	Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.write_json`[78]

Examples

```
## Not run: iai::write_json(file.path(tempdir(), "out.json"), lnr)
```

[77] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.write_html
[78] https://docs.interpretable.ai/IAIBase/stable/reference/#IAI.write_json

write_png *write_png*

Description

Output a learner as a PNG image

Usage

```
write_png(filename, lnr, ...)
```

Arguments

filename	Where to save the output.
lnr	The learner or grid to output.
...	Refer to the Julia documentation for available parameters.

Details

Julia Equivalent: `IAI.write_png`[79]

Examples

```
## Not run: iai::write_png(file.path(tempdir(), "tree.png"), lnr)
```

write_questionnaire *write_questionnaire*

Description

Output a learner as an interactive questionnaire in HTML format

Usage

```
write_questionnaire(filename, lnr, ...)
```

Arguments

filename	Where to save the output.
lnr	The learner or grid to output.
...	Refer to the Julia documentation for available parameters.

[79] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.write_png

Details

Julia Equivalent: `IAI.write_questionnaire`[80]

Examples

```
## Not run: iai::write_questionnaire(file.path(tempdir(), "questionnaire.html"), lnr)
```

[80] https://docs.interpretable.ai/IAITrees/stable/reference/#IAI.write_questionnaire