Package: daltoolboxdp (via r-universe)

October 24, 2024

Title Data Pre-Processing Extensions

Version 1.0.767

Description An important aspect of data analytics is related to data management support for artificial intelligence. It is related to preparing data correctly. This package provides extensions to support data preparation in terms of both data sampling and data engineering. Overall, the package provides researchers with a comprehensive set of functionalities for data science based on experiment lines, promoting ease of use, extensibility, and integration with various tools and libraries. Information on Experiment Line is based on Ogasawara et al. (2009) <doi:10.1007/978-3-642-02279-1_20>.

License MIT + file LICENSE

URL https://github.com/cefet-rj-dal/daltoolboxdp

Encoding UTF-8

RoxygenNote 7.3.1

Imports daltoolbox, leaps, FSelector, doBy, glmnet, smotefamily

NeedsCompilation no

Author Eduardo Ogasawara [aut, ths, cre] (<https://orcid.org/0000-0002-0466-0626>), Federal Center for Technological Education of Rio de Janeiro (CEFET/RJ) [cph]

Maintainer Eduardo Ogasawara <eogasawara@ieee.org>

Repository CRAN

Date/Publication 2024-03-27 09:40:02 UTC

Contents

bal_oversampling 2
bal_subsampling 2
fs 3
fs_fss 4

fs_ig 4
fs_lasso 5
fs_relief 6

Index **7**

`bal_oversampling` *Oversampling*

Description

Oversampling balances the class distribution of a dataset by increasing the representation of the minority class in the dataset. It wraps the smotefamily library.

Usage

```
bal_oversampling(attribute)
```

Arguments

`attribute`	The class attribute to target balancing using oversampling.

Value

A `bal_oversampling` object.

Examples

```
data(iris)
mod_iris <- iris[c(1:50,51:71,101:111),]

bal <- bal_oversampling('Species')
bal <- daltoolbox::fit(bal, mod_iris)
adjust_iris <- daltoolbox::transform(bal, mod_iris)
table(adjust_iris$Species)
```

`bal_subsampling` *Subsampling*

Description

Subsampling balances the class distribution of a dataset by reducing the representation of the majority class in the dataset.

Usage

```
bal_subsampling(attribute)
```

Arguments

attribute	The class attribute to target balancing using subsampling

Value

A bal_subsampling object.

Examples

```
data(iris)
mod_iris <- iris[c(1:50,51:71,101:111),]

bal <- bal_subsampling('Species')
bal <- daltoolbox::fit(bal, mod_iris)
adjust_iris <- daltoolbox::transform(bal, mod_iris)
table(adjust_iris$Species)
```

---

fs *Feature Selection*

Description

Feature selection is a process of selecting a subset of relevant features from a larger set of features in a dataset for use in model training. The FeatureSelection class in R provides a framework for performing feature selection.

Usage

```
fs(attribute)
```

Arguments

attribute	The target variable.

Value

An instance of the FeatureSelection class.

Examples

```
#See ?fs_fss for an example of feature selection
```

fs_fss *Forward Stepwise Selection*

Description

Forward stepwise selection is a technique for feature selection in which attributes are added to a model one at a time based on their ability to improve the model's performance. It stops adding once the candidate addition does not significantly improve model adjustment. It wraps the leaps library.

Usage

```
fs_fss(attribute)
```

Arguments

attribute	The target variable.

Value

A fs_fss object.

Examples

```
data(iris)
myfeature <- daltoolbox::fit(fs_fss("Species"), iris)
data <- daltoolbox::transform(myfeature, iris)
head(data)
```

fs_ig *Information Gain*

Description

Information Gain is a feature selection technique based on information theory. It measures the information obtained for the target variable by knowing the presence or absence of a feature. It wraps the FSelector library.

Usage

```
fs_ig(attribute)
```

Arguments

attribute	The target variable.

Value

A fs_ig object.

Examples

```
data(iris)
myfeature <- daltoolbox::fit(fs_ig("Species"), iris)
data <- daltoolbox::transform(myfeature, iris)
head(data)
```

---

fs_lasso *Feature Selection using Lasso*

Description

Feature selection using Lasso regression is a technique for selecting a subset of relevant features. It wraps the glmnet library.

Usage

```
fs_lasso(attribute)
```

Arguments

attribute	The target variable.

Value

A fs_lasso object.

Examples

```
data(iris)
myfeature <- daltoolbox::fit(fs_lasso("Species"), iris)
data <- daltoolbox::transform(myfeature, iris)
head(data)
```

fs_relief *Relief*

Description

Feature selection using Relief is a technique for selecting a subset of relevant features. It calculates the relevance of a feature by considering the difference in feature values between nearest neighbors of the same and different classes. It wraps the FSelector library.

Usage

```
fs_relief(attribute)
```

Arguments

attribute	The target variable.

Value

A fs_relief object.

Examples

```
data(iris)
myfeature <- daltoolbox::fit(fs_relief("Species"), iris)
data <- daltoolbox::transform(myfeature, iris)
head(data)
```

Index

bal_oversampling, 2
bal_subsampling, 2

fs, 3
fs_fss, 4
fs_ig, 4
fs_lasso, 5
fs_relief, 6