Package 'liver'

September 25, 2020

Title ``Eating the Liver of Data Science''

Version 1.2

Description Provides a collection of helper functions that make various techniques from data science more user-friendly for non-experts. In this way, our aim is to allow non-experts to become familiar with the techniques with only a minimal level of coding knowledge. Indeed, following an ancient Persian idiom, we refer to this as ``eating the liver of data science'' which could be interpreted as ``getting intimately close with data science''. Examples of procedures we include are: data partitioning for out-of-sample testing, computing Mean Squared Error (MSE) for quantifying prediction accuracy, and data transformation (z-score and min-max). Besides such helper functions, the package also includes several interesting datasets that are useful for multivariate analysis.

URL https://www.uva.nl/profile/a.mohammadi

Depends R (>= 3.5.0), class

License GPL (>= 2)

Repository CRAN

Author Reza Mohammadi [aut, cre] (<https://orcid.org/0000-0001-9538-0648>), Kevin Burke [aut]

Maintainer Reza Mohammadi <a.mohammadi@uva.nl>

NeedsCompilation no

Date/Publication 2020-09-25 16:40:04 UTC

R topics documented:

`liver-package` *liver: "Eating the Liver of Data Science"*

Description

The **liver** package provides a collection of helper functions that make various techniques from data science more user-friendly for non-experts. In this way, our aim is to allow non-experts to become familiar with the techniques with only a minimal level of coding knowledge. Indeed, following an ancient Persian idiom, we refer to this as "eating the liver of data science" which could be interpreted as "getting intimately close with data science". Examples of procedures we include are: data partitioning for out-of-sample testing, computing Mean Squared Error (MSE) for quantifying prediction accuracy, and data transformation (z-score and min-max). Besides such helper functions, the package also includes several interesting datasets that are useful for multivariate analysis.

Author(s)

Reza Mohammadi <a.mohammadi@uva.nl>
Amsterdam Business School
University of Amsterdam

Kevin Burke <kevin.burke@ul.ie>
Departement of Statistics
University of Limerick

Maintainer: Reza Mohammadi <a.mohammadi@uva.nl>

`adult` *adult data set*

Description

The adult dataset contains 15 features and 48842 records. It was collected from the US Census bureau and the primary task is to predict whether a given adult makes more than $50K a year based attributes such as education, hours of work per week, etc. The target feature is *income*, a factor with levels "<=50K" and ">50K", and the remaining 14 variables are predictors.

Usage

```
data( adult )
```

Format

The `adult` dataset, as a data frame, contains 48842 rows (customers) and 15 columns (variables/features). The 15 variables are:

- `age`: age in years.
- `workclass`: a factor with 9 levels.
- `demogweight`: the demographics to describe a person.
- `education`: a factor with 16 levels.
- `education.num`: number of years of education.
- `marital.status`: a factor with 7 levels.
- `occupation`: a factor with 15 levels.
- `relationship`: a factor with 6 levels.
- `race`: a factor with 5 levels "Female","Male".
- `gender`: a factor with 2 levels.
- `capital.gain`: capital gains.
- `capital.loss`: capital losses.
- `hours.per.week`: number of hours of work per week.
- `native.country`: a factor with 42 levels.
- `income`: yearly income as a factor with levels "<=50K" and ">50K".

Details

This dataset can be downloaded from the UCI machine learning repository:

http://www.cs.toronto.edu/~delve/data/adult/desc.html

The detailed description on the dataset can be found in the UCI documentation

http://www.cs.toronto.edu/~delve/data/adult/adultDetail.html

References

Kohavi, R. (1996). Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. *Kdd*.

See Also

`bank`, `churn`, `churnTel`, `classifyRisk`, `housePrice`

Examples

```
data( adult )

str( adult )
```

bank *Bank marketing data set*

Description

The data is related with direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, in order to access if the product (bank term deposit) would be (or not) subscribed. The classification goal is to predict if the client will subscribe a term deposit (variable deposit).

Usage

```
data( bank )
```

Format

The bank dataset, as a data frame, contains 4521 rows (customers) and 17 columns (variables/features). The 17 variables are:

Bank client data:

- age: numeric.
- job: type of job; categorical: "admin.", "unknown", "unemployed", "management", "housemaid", "entrepreneur", "student", "blue-collar, "self-employed", "retired", "technician", "services".
- marital: marital status; categorical: "married", "divorced", "single"; note: "divorced" means divorced or widowed.
- education: categorical: "secondary", "primary", "tertiary", "unknown".
- default: has credit in default?; binary: "yes","no".
- balance: average yearly balance, in euros; numeric.
- housing: has housing loan? binary: "yes", "no".
- loan: has personal loan? binary: "yes", "no".

Related with the last contact of the current campaign:

- contact: contact: contact communication type; categorical: "unknown","telephone","cellular".
- day: last contact day of the month; numeric.
- month: last contact month of year; categorical: "jan", "feb", "mar", ..., "nov", "dec".
- duration: last contact duration, in seconds; numeric.

Other attributes:

- campaign: number of contacts performed during this campaign and for this client; numeric, includes last contact.
- pdays: number of days that passed by after the client was last contacted from a previous campaign; numeric, -1 means client was not previously contacted.

- previous: number of contacts performed before this campaign and for this client; numeric.
- poutcome: outcome of the previous marketing campaign; categorical: "success", "failure", "unknown", "other".

Target variable:

- deposit: Indicator of whether the client subscribed a term deposit; binary: "yes" or "no".

Details

This dataset can be downloaded from the UCI machine learning repository:

http://archive.ics.uci.edu/ml/datasets/Bank+Marketing

References

Moro, S., Laureano, R. and Cortez, P. (2011) Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology. In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference.

See Also

churn, adult, churnTel, classifyRisk, housePrice

Examples

```
data( bank )

str( bank )
```

---

churn *Churn data set*

Description

This dataset comes from IBM Sample Data Sets. Customer *churn* occurs when customers stop doing business with a company, also known as customer attrition. The data set contains 3333 rows (customers) and 20 columns (features). The "Churn" column is our target which indicate whether customer churned (left the company) or not.

Usage

```
data( churn )
```

Format

The churn dataset, as a data frame, contains 3333 rows (customers) and 20 columns (variables/features). The 20 variables are:

- State: Categorical, for the 50 states and the District of Columbia.
- Account.Length: count, how long account has been active.
- Area.Code: Categorical.
- Int.l.Plan: Categorical, yes or no, international plan.
- VMail.Plan: Categorical, yes or no, voice mail plan.
- VMail.Message: Count, number of voice mail messages.
- Day.Mins: Continuous, minutes customer used service during the day.
- Day.Calls: Count, total number of calls during the day.
- Day.Charge: Continuous, total charge during the day.
- Eve.Mins: Continuous, minutes customer used service during the evening.
- Eve.Calls: Count, total number of calls during the evening.
- Eve.Charge: Continuous, total charge during the evening.
- Night.Mins: Continuous, minutes customer used service during the night.
- Night.Calls: Count, total number of calls during the night.
- Night.Charge: Continuous, total charge during the night.
- Intl.Mins: Continuous, minutes customer used service to make international calls.
- Intl.Calls: Count, total number of international calls.
- Intl.Charge: Continuous, total international charge.
- CustServ.Calls: Count, number of calls to customer service.
- Churn: Categorical, True or False. Indicator of whether the customer has left the company (True or False).

References

Larose, D. T. and Larose, C. D. (2014). Discovering knowledge in data: an introduction to data mining. *John Wiley & Sons*.

See Also

churnTel, bank, adult, classifyRisk, housePrice

Examples

```
data( churn )

str( churn )
```

churnTel *churnTel dataset*

Description

Customer *churn* occurs when customers stop doing business with a company, also known as customer attrition. The data set contains 7043 rows (customers) and 21 columns (features). The "Churn" column is our target which indicate whether customer churned (left the company) or not.

Usage

```
data( churnTel )
```

Format

The churnTel dataset, as a data frame, contains 7043 rows (customers) and 21 columns (variables/features).

Details

For more information related to the dataset see:

https://www.kaggle.com/blastchar/telco-customer-churn

See Also

churn, adult, bank, classifyRisk, housePrice

Examples

```
data( churnTel )

str( churnTel )
```

classifyRisk *classifyRisk data set*

Description

The classifyRisk dataset containing 6 features and 246 records. The target feature is *risk*, a factor with levels "good risk" and "bad loss" along with 5 predictors.

Usage

```
data( classifyRisk )
```

Format

The classifyRisk dataset, as a data frame, contains 246 rows (customers) and 6 columns (variables/features). The 6 variables are:

- mortgage: A factor with levels "n" and "y".
- nr_loans: Number of loans that constomers have.
- age: age in years.
- marital_status: A factor with levels "single", "married", and "other".
- income: yearly income.
- risk: A factor with levels "good risk" and "bad loss".

References

Larose, D. T. and Larose, C. D. (2014). Discovering knowledge in data: an introduction to data mining. *John Wiley & Sons*.

See Also

bank, adult, churn, churnTel, housePrice

Examples

```
data( classifyRisk )

str( classifyRisk )
```

---

find.na *find.na*

Description

Finding missing values.

Usage

```
find.na( x )
```

Arguments

x	a numerical vector, matrix or data.frame.

Value

A numeric matrix with two columns.

Author(s)

Reza Mohammadi <a.mohammadi@uva.nl> and Kevin Burke <kevin.burke@ul.ie>

Examples

```
x = c( 2.3, NA, -1.4, 0, 3.45 )

find.na( x )
```

`housePrice` *housePrice dataset*

Description

This data set contains 1460 rows (customers) and 81 columns (features). The "SalePrice" column is the target.

Usage

```
data( housePrice )
```

Format

The `housePrice` dataset, as a data frame, contains 1460 rows (customers) and 81 columns (variables/features).

Details

For more information related to the dataset see:

https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data

See Also

`adult`, `churn`, `churnTel`, `classifyRisk`

Examples

```
data( housePrice )

str( housePrice )
```

kNN *k-Nearest Neighbour Classification*

Description

kNN is used to perform k-nearest neighbour classification for test set using training set. For each row of the test set, the k nearest (based on Euclidean distance) training set vectors are found. Then, the classification is done by majority vote (ties broken at random). This function provides a formula interface to the knn function of R package class. In addition, it allows normalization of the given data using the transform function.

Usage

```
kNN( formula, train, test, k = 1, transform = FALSE, l = 0, prob = FALSE, use.all = TRUE )
```

Arguments

formula	a formula, with a response but no interaction terms. For the case of data frame, it is taken as the model frame (see model.frame).
train	data frame or matrix of train set cases.
test	data frame or matrix of test set cases.
k	number of neighbours considered.
transform	a character with options FALSE (default), "minmax", and "zscore". Option "minmax" means no transformation. This option allows the users to use normalized version of the train and test sets for the kNN aglorithm.
l	minimum vote for definite decision, otherwise doubt. (More precisely, less than k-l dissenting votes are allowed, even if k is increased by ties.)
prob	If this is true, the proportion of the votes for the winning class are returned as attribute prob.
use.all	controls handling of ties. If true, all distances equal to the kth largest are included. If false, a random selection of distances equal to the kth is chosen to use exactly k neighbours.

Value

Factor of classifications for the test set, in which the doubt will be returned as NA; basically, the return value is the same as in the knn function of R package class.

Author(s)

Reza Mohammadi <a.mohammadi@uva.nl> and Kevin Burke <kevin.burke@ul.ie>

References

Ripley, B. D. (1996) *Pattern Recognition and Neural Networks.* Cambridge.

Venables, W. N. and Ripley, B. D. (2002) *Modern Applied Statistics with S.* Fourth edition. Springer.

See Also

knn, transform

Examples

```
data( churn )

train = churn[ 1:100, ]
test  = churn[   101, ]

kNN( Churn ~ CustServ.Calls + Int.l.Plan, train = train, test = test )
```

---

minmax *Min-Max normalization*

Description

Performs Min-Max normalization of numerical variables.

Usage

```
minmax( x, na.rm = FALSE )
```

Arguments

x	a numerical vector, matrix or data.frame.
na.rm	a logical value indicating whether NA values in x should be stripped before the computation proceeds.

Value

transformed version of x.

Author(s)

Reza Mohammadi <a.mohammadi@uva.nl> and Kevin Burke <kevin.burke@ul.ie>

See Also

transform, zscore

Examples

```
x = c( 2.3, -1.4, 0, 3.45 )

minmax( x )
```

mse *Mean Squared Error (MSE)*

Description

Computes mean squared error.

Usage

```
mse( pred, true, weight = 1, na.rm = FALSE )
```

Arguments

pred	a numerical vector of estimated values.
true	a numerical vector of true values.
weight	a numerical vector of weights the same length as pred.
na.rm	a logical value indicating whether NA values in pred should be stripped before the computation proceeds.

Value

The computed mean squared error (numeric value).

Author(s)

Reza Mohammadi <a.mohammadi@uva.nl> and Kevin Burke <kevin.burke@ul.ie>

Examples

```
pred = c( 2.3, -1.4, 0, 3.45 )
true = c( 2.1, -0.9, 0, 2.99 )

mse( pred, true )
```

partition *Partition the data*

Description

Randomly partitions the data (primarly intended to split into "training" and "test" sets) according to the supplied probabilities.

Usage

```
partition( data, prob = c( 0.7, 0.3 ) )
```

Arguments

data	an ($n \times p$) matrix or a data.frame.
prob	a numerical vector in [0, 1].

Value

a list which includes the data partitions.

Author(s)

Reza Mohammadi <a.mohammadi@uva.nl> and Kevin Burke <kevin.burke@ul.ie>

Examples

```
data( iris )

partition( data = iris, prob = c( 0.7, 0.3 ) )
```

---

skewness *Skewness*

Description

Computes the skewness for each field.

Usage

```
skewness( x, na.rm = FALSE )
```

Arguments

x	a numerical vector, matrix or data.frame.
na.rm	a logical value indicating whether NA values in x should be stripped before the computation proceeds.

Value

A numeric vector of skewness values.

Author(s)

Reza Mohammadi <a.mohammadi@uva.nl> and Kevin Burke <kevin.burke@ul.ie>

Examples

```
x = c( 2.3, -1.4, 0, 3.45 )

skewness( x )
```

`transform` *Z-score normalization*

Description

Performs variable transformation such as Z-score and min-max normalization.

Usage

```
transform( x, method = c( "minmax", "zscore" ), na.rm = FALSE )
```

Arguments

`x`	a numerical `vector`, `matrix` or `data.frame`.
`method`	a method to transfer the `x`.
`na.rm`	a logical value indicating whether NA values in `x` should be stripped before the computation proceeds.

Value

transformed version of `x`.

Author(s)

Reza Mohammadi <a.mohammadi@uva.nl> and Kevin Burke <kevin.burke@ul.ie>

See Also

`zscore`, `minmax`

Examples

```
x = c( 2.3, -1.4, 0, 3.45 )

transform( x, method = "minmax" )

transform( x, method = "zscore" )
```

zscore *Z-score normalization*

Description

Performs Z-score normalization to transform numerical variables.

Usage

```
zscore( x, na.rm = FALSE )
```

Arguments

x	a numerical vector, matrix or data.frame.
na.rm	a logical value indicating whether NA values in x should be stripped before the computation proceeds.

Value

transformed version of x.

Author(s)

Reza Mohammadi <a.mohammadi@uva.nl> and Kevin Burke <kevin.burke@ul.ie>

See Also

transform, minmax

Examples

```
x = c( 2.3, -1.4, 0, 3.45 )

zscore( x )
```

Index