Package ‘liver’

March 18, 2025

Title ``Eating the Liver of Data Science''

Version 1.19

Description Provides a suite of helper functions and a collection of datasets used in the book <https://uncovering-data-science.netlify.app>. It is designed to make data science techniques accessible to individuals with minimal coding experience. Inspired by an ancient Persian idiom, the package likens this learning process to ``eating the liver of data science,'' symbolizing deep and immersive engagement with the field.

URL https://uncovering-data-science.netlify.app

Depends R (>= 3.5.0)

Imports class, ggplot2

Suggests pROC, skimr, knitr, rmarkdown, data.table, mltools, forcats

VignetteBuilder knitr

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 2025-03-18 12:00:07 UTC

Contents

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

Description

The **liver** package provides a suite of helper functions and a collection of datasets used in the book Data Science Foundations and Machine Learning Using R. Designed to make data science techniques accessible to individuals with minimal coding experience, it simplifies tasks such as data partitioning for out-of-sample testing and data transformations (z-score and min-max). Inspired by an ancient Persian idiom, the package likens this learning process to "eating the liver of data science," symbolizing deep and immersive engagement with the field. In addition to its helper functions, liver also includes a collection of datasets valuable 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>

accuracy *Average classification accuracy*

Description

Computes average classification accuracy.

Usage

```
accuracy(pred, actual, cutoff = NULL, reference = NULL)
```

Arguments

pred	a numerical vector of estimated values.
actual	a numerical vector of actual values.
cutoff	cutoff value for the case that pred is vector of probabilites.
reference	a factor of classes to be used as the true results.

Value

the computed average classification accuracy (numeric value).

Author(s)

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

See Also

conf.mat, mse, mae

Examples

```
pred   = c("no", "yes", "yes", "no", "no", "yes", "no", "no")
actual = c("yes", "no", "yes", "no", "no", "no", "yes", "yes")

accuracy(pred, actual)
```

`adult` *adult data set*

Description

the adult dataset 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 48598 rows and 15 columns (variables/features). the 15 variables are:

- `age`: age in years.
- `workclass`: a factor with 6 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 5 levels.
- `occupation`: a factor with 15 levels.
- `relationship`: a factor with 6 levels.
- `race`: a factor with 5 levels.
- `gender`: a factor with levels "Female","Male".
- `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`

A detailed description of the dataset can be found in the UCI documentation at:

`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

risk, churn, churnTel, bank, advertising, marketing, insurance, cereal, housePrice, house

Examples

```
data(adult)

str(adult)
```

---

advertising *advertising data set*

Description

the dataset is from an anonymous organisation's social media ad campaign. the advertising dataset contains 11 features and 1143 records.

Usage

```
data(advertising)
```

Format

the advertising dataset, as a data frame, contains 1143 rows and 11 columns (variables/features). the 11 variables are:

- ad.id: an unique ID for each ad.
- xyz.campaign.id: an ID associated with each ad campaign of XYZ company.
- fb.campaign.id: an ID associated with how Facebook tracks each campaign.
- age: age of the person to whom the ad is shown.
- gender: gender of the person to whim the add is shown.
- interest: a code specifying the category to which the person's interest belongs (interests are as mentioned in the person's Facebook public profile).
- impressions: the number of times the ad was shown.
- clicks: number of clicks on for that ad.
- spend: amount paid by company xyz to Facebook, to show that ad.
- conversion: total number of people who enquired about the product after seeing the ad.
- approved: total number of people who bought the product after seeing the ad.

Details

A detailed description of the dataset can be found:

https://www.kaggle.com/loveall/clicks-conversion-tracking

See Also

adult, risk, churn, churnTel, bank, marketing, insurance, cereal, housePrice, house

Examples

```
data(advertising)

str(advertising)
```

bank *Bank marketing data set*

Description

the data is related to 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

`adult`, `risk`, `churn`, `churnTel`, `advertising`, `marketing`, `insurance`, `cereal`, `housePrice`, `house`

Examples

```
data(bank)

str(bank)
```

cereal *Cereal data set*

Description

This dataset contains nutrition information for 77 breakfast cereals and includes 16 variables. the "`rating`" column is our target as a rating of the cereals (Possibly from Consumer Reports?).

Usage

```
data(cereal)
```

Format

the `cereal` dataset, as a data frame, contains 77 rows (breakfast cereals) and 16 columns (variables/features). the 16 variables are:

- `name`: Name of cereal.
- `manuf`: Manufacturer of cereal:
  - `A`: American Home Food Products;
  - `G`: General Mills;
  - `K`: Kelloggs;
  - `N`: Nabisco;
  - `P`: Post;
  - `Q`: Quaker Oats;
  - `R`: Ralston Purina;
- `type`: cold or hot.
- `calories`: calories per serving.
- `protein`: grams of protein.
- `fat`: grams of fat.
- `sodium`: milligrams of sodium.
- `fiber`: grams of dietary fiber.
- `carbo`: grams of complex carbohydrates.
- `sugars`: grams of sugars.
- `potass`: milligrams of potassium.
- `vitamins`: vitamins and minerals - 0, 25, or 100, indicating the typical percentage of FDA recommended.
- `shelf`: display shelf (1, 2, or 3, counting from the floor).
- `weight`: weight in ounces of one serving.
- `cups`: number of cups in one serving.
- `rating`: a rating of the cereals (Possibly from Consumer Reports?).

Details

the original source can be found: https://perso.telecom-paristech.fr/eagan/class/igr204/datasets

See Also

adult, risk, churn, churnTel, bank, advertising, marketing, insurance, housePrice, house

Examples

```
data(cereal)

str(cereal)
```

---

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 5000 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 5000 rows (customers) and 20 columns (variables/features). the 20 variables are:

- `state`: Categorical, for the 51 states and the District of Columbia.
- `area.code`: Categorical.
- `account.length`: count, how long account has been active.
- `voice.plan`: Categorical, yes or no, voice mail plan.
- `voice.messages`: Count, number of voice mail messages.
- `intl.plan`: Categorical, yes or no, international plan.
- `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.
- `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.
- customer.calls: Count, number of calls to customer service.
- churn: Categorical, yes or no. Indicator of whether the customer has left the company (yes or no).

References

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

See Also

adult, risk, churnTel, bank, advertising, marketing, insurance, cereal, housePrice, house

Examples

```
data(churn)

str(churn)
```

---

churnCredit *Churn dataset for Credit Card Customers*

Description

Customer *churn* occurs when customers stop doing business with a company, also known as customer attrition. the data set contains 10127 rows (customers) and 21 columns (features). the "churn" column is our target which indicate whether customer churned (left the company) or not.

Usage

```
data(churnCredit)
```

Format

the churnCredit dataset, as a data frame, contains 10127 rows (customers) and 21 columns (variables/features). the 21 variables are:

- customer.ID: Customer ID.
- gender: Whether the customer is a male or a female.
- age: Customer's Age in Years.
- educaton: Educational Qualification of the account holder (example: high school, college graduate, etc.)
- marital.status: Married, Single, Divorced, Unknown
- income: Annual Income (in Dollar). Category of the account holder (< $40K, $40K - 60K, $60K - $80K, $80K-$120K, > $120K).
- dependent.counts: Number of dependent counts.
- card.category: Type of Card (Blue, Silver, Gold, Platinum).
- months.on.book: Period of relationship with bank.
- relationship.count: Total number of products held by the customer.
- months.inactive: Number of months inactive in the last 12 months.
- contacts.count.12: Number of Contacts in the last 12 months.
- credit.limit: Credit Limit on the Credit Card.
- revolving.balance: Total Revolving Balance on the Credit Card.
- open.to.buy: Open to Buy Credit Line (Average of last 12 months).
- transaction.amount.Q4.Q1: Change in Transaction Amount (Q4 over Q1).
- transaction.amount.12: Total Transaction Amount (Last 12 months).
- transaction.count: Total Transaction Count (Last 12 months).
- transaction.change: Change in Transaction Count (Q4 over Q1).
- utilization.ratio: Average Card Utilization Ratio.
- churn: Whether the customer churned or not (yes or no).

Details

For more information related to the dataset see:

https://www.kaggle.com/sakshigoyal7/credit-card-customers

See Also

adult, risk, churn, churnTel, bank, advertising, marketing, insurance, cereal, housePrice, house

Examples

```
data(churnCredit)

str(churnCredit)
```

`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). the 21 variables are:

- `customer.ID`: Customer ID.
- `gender`: Whether the customer is a male or a female.
- `senior.citizen`: Whether the customer is a senior citizen or not (1, 0).
- `partner`: Whether the customer has a partner or not (yes, no).
- `dependent`: Whether the customer has dependents or not (yes, no).
- `tenure`: Number of months the customer has stayed with the company.
- `phone.service`: Whether the customer has a phone service or not (yes, no).
- `multiple.lines`: Whether the customer has multiple lines or not (yes, no, no phone service).
- `internet.service`: Customer's internet service provider (DSL, fiber optic, no).
- `online.security`: Whether the customer has online security or not (yes, no, no internet service).
- `online.backup`: Whether the customer has online backup or not (yes, no, no internet service).
- `device.protection`: Whether the customer has device protection or not (yes, no, no internet service).
- `tech.support`: Whether the customer has tech support or not (yes, no, no internet service).
- `streaming.TV`: Whether the customer has streaming TV or not (yes, no, no internet service).
- `streaming.movie`: Whether the customer has streaming movies or not (yes, no, no internet service).
- `contract`: the contract term of the customer (month to month, 1 year, 2 year).
- `paperless.bill`: Whether the customer has paperless billing or not (yes, no).
- `payment.method`: the customer's payment method (electronic check, mail check, bank transfer, credit card).
- `monthly.charge`: the amount charged to the customer monthly.
- `total.charges`: the total amount charged to the customer.
- `churn`: Whether the customer churned or not (yes or no).

Details

For more information related to the dataset see:

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

See Also

adult, risk, churn, bank, advertising, marketing, insurance, cereal, housePrice, house

Examples

```
data(churnTel)

str(churnTel)
```

---

conf.mat *Confusion Matrix*

Description

Create a Confusion Matrix.

Usage

```
conf.mat(pred, actual, cutoff = 0.5, reference = NULL,
         proportion = FALSE, dnn = c("Predict", "Actual"), ...)
```

Arguments

pred	a vector of estimated values.
actual	a vector of actual values.
cutoff	cutoff value for the case that pred is vector of probabilites.
reference	a factor of classes to be used as the true results.
proportion	Logical: FALSE (default) for a confusion matrix with number of cases. TRUE, for a confusion matrix with the proportion of cases.
dnn	the names to be given to the dimensions in the result (the dimnames names).
...	options to be passed to table.

Value

the results of table on pred and actual.

Author(s)

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

See Also

conf.mat.plot, accuracy

Examples

```
pred   = c("no", "yes", "yes", "no", "no", "yes", "no", "no")
actual = c("yes", "no", "yes", "no", "no", "no", "yes", "yes")

conf.mat(pred, actual)
conf.mat(pred, actual, proportion = TRUE)
```

conf.mat.plot *Plot Confusion Matrix*

Description

Plot a Confusion Matrix.

Usage

```
conf.mat.plot(pred, actual, cutoff = 0.5, reference = NULL, conf.level = 0,
              margin = 1, color = c("#ff83a8", "#83ff9b"), ...)
```

Arguments

`pred` a vector of estimated values.

`actual` a vector of actual values.

`cutoff` cutoff value for the case that `pred` is vector of probabilites.

`reference` a factor of classes to be used as the true results.

`conf.level` confidence level used for the confidence rings on the odds ratios. Must be a single nonnegative number less than 1; if set to 0 (the default), confidence rings are suppressed.

`margin` a numeric vector with the margins to equate. Must be one of 1 (the default), 2, or c(1, 2), which corresponds to standardizing the row, column, or both margins in each 2 by 2 table. Only used if std equals "margins".

`color` a vector of length 2 specifying the colors to use for the smaller and larger diagonals of each 2 by 2 table.

`...` options to be passed to `fourfoldplot`.

Author(s)

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

See Also

conf.mat

Examples

```
pred   = c("no", "yes", "yes", "no", "no", "yes", "no", "no")
actual = c("yes", "no", "yes", "no", "no", "no", "yes", "yes")

conf.mat.plot(pred, actual)
```

`corona` *Corona data set*

Description

COVID-19 Coronavirus data - daily (up to 14 December 2020).

Usage

```
data(corona)
```

Format

the `corona` dataset, as a data frame, contains 61900 rows and 12 columns (variables/features).

Details

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

https://data.europa.eu/euodp/en/data/dataset/covid-19-coronavirus-data

See Also

`churn`, `adult`, `risk`, `churnTel`, `bank`, `advertising`, `marketing`, `insurance`, `cereal`, `housePrice`, `house`

Examples

```
data(corona)

str(corona)
```

`fertilizer` *Fertilizer data set*

Description

the fertilizer dataset contains 4 features and 96 records. Results from an experiment to compare yields of a crop obtained under three different fertilizers. the target feature is *yield*.

Usage

```
 data(fertilizer)
```

See Also

`adult`, `risk`, `churn`, `churnTel`, `bank`, `advertising`, `marketing`, `insurance`, `cereal`, `housePrice`, `house`

Examples

```
data(fertilizer)

str(fertilizer)
```

`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)
```

`house` *house data set*

Description

the house dataset contains 6 features and 414 records. the target feature is *unit.price* and the remaining 5 variables are predictors.

Usage

```
data(house)
```

Format

the house dataset, as a data frame, contains 414 rows and 6 columns (variables/features). the 6 variables are:

- `house.age`: house age (numeric, in year).
- `distance.to.MRT`: distance to the nearest MRT station (numeric).
- `stores.number`: number of convenience stores (numeric).
- `latitude`: latitude (numeric).
- `longitude`: longitude (numeric).
- `unit.price`: house price of unit area (numeric).

Details

A detailed description of the dataset can be found:

`https://www.kaggle.com/quantbruce/real-estate-price-prediction`

See Also

`adult`, `risk`, `churn`, `churnTel`, `bank`, `advertising`, `marketing`, `insurance`, `cereal`, `housePrice`

Examples

```
data(house)

str(house)
```

housePrice *housePrice dataset*

Description

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

Usage

```
data(housePrice)
```

Format

the housePrice dataset, as a data frame, contains 1460 rows 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, risk, churn, churnTel, bank, advertising, marketing, insurance, cereal, house

Examples

```
data(housePrice)

str(housePrice)
```

insurance *insurance data set*

Description

the insurance dataset contains 7 features and 1338 records. the target feature is *charge* and the remaining 6 variables are predictors.

Usage

```
data(insurance)
```

Format

the insurance dataset, as a data frame, contains 1338 rows (customers) and 7 columns (variables/features). the 7 variables are:

- age: age of primary beneficiary.
- bmi: body mass index, providing an understanding of body, weights that are relatively high or low relative to height, objective index of body weight (kg / m ^ 2) using the ratio of height to weight, ideally 18.5 to 24.9.
- children: Number of children covered by health insurance / Number of dependents.
- smoker: Smoking as a factor with 2 levels, yes, no.
- gender: insurance contractor gender, female, male.
- region: the beneficiary's residential area in the US, northeast, southeast, southwest, northwest.
- charge: individual medical costs billed by health insurance.

Details

A detailed description of the dataset can be found:

https://www.kaggle.com/mirichoi0218/insurance

References

Brett Lantz (2019). Machine Learning with R: Expert techniques for predictive modeling. *Packt Publishing Ltd.*

See Also

adult, risk, churn, churnTel, bank, advertising, marketing, cereal, housePrice, house

Examples

```
data(insurance)

str(insurance)
```

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 class::knn() function of R package class. In addition, it allows normalization of the given data using the scaler function.

Usage

```
kNN(formula, train, test, k = 1, scaler = FALSE, type = "class", l = 0,
    use.all = TRUE, na.rm = FALSE)
```

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.

scaler — 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.

type — either "class" (default) for the predicted class or "prob" for model confidence values.

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.)

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.

na.rm — a logical value indicating whether NA values in x should be stripped before the computation proceeds.

Value

When type = "class" (default), a factor vector is returned, in which the doubt will be returned as NA. When type = "prob", a matrix of confidence values is returned (one column per 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, scaler

Examples

```
data(risk)

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

kNN(risk ~ income + age, train = train, test = test)
```

kNN.plot *Visualizing the Optimal Number of k*

Description

Visualizing the Optimal Number of k for k-Nearest Neighbour (kNN) algorithm based on accuracy or Mean Square Error (MSE).

Usage

```
kNN.plot(formula, train, test, k.max = 10, scaler = FALSE,
         base = "accuracy", reference = NULL, cutoff = NULL,
         type = "class", report = FALSE, set.seed = NULL, ...)
```

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.max` the maximum number of neighbors to consider can either be a single value, with a minimum of 2, or a vector representing a range of values k.

`scaler` 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.

`base` base measurement: `accuracy` (default), `error`, or `MSE` for Mean Square Error.

`reference` a factor of classes to be used as the true results.

`cutoff` cutoff value for the case that the output of knn algorithm is vector of probabilites.

`type` either `"class"` (default) for the predicted class or `"prob"` for model confidence values.

`report` a character with options `FALSE` (default) and `TRUE`. Option `TRUE` reports the values of the `base` measurement.

`set.seed` a single value, interpreted as an integer, or NULL.

`...` options to be passed to `kNN()`.

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, scaler

Examples

```
data(risk)

partition_risk <- partition(data = risk, ratio = c(0.6, 0.4))

train <- partition_risk$part1
test  <- partition_risk$part1

kNN.plot(risk ~ income + age, train = train, test = test)
kNN.plot(risk ~ income + age, train = train, test = test, base = "error")
```

mae *Mean Absolute Error (MAE)*

Description

Computes mean absolute error.

Usage

```
mae(pred, actual, weight = 1, na.rm = FALSE)
```

Arguments

pred	a numerical vector of estimated values.
actual	a numerical vector of actual 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>

See Also

mse

Examples

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

mae(pred, actual)
```

---

marketing *marketing data set*

Description

the marketing dataset contains 8 features and 40 records as 40 days that report how much we spent, how many clicks, impressions and transactions we got, whether or not a display campaign was running, as well as our revenue, click-through-rate and conversion rate. the target feature is *revenue* and the remaining 7 variables are predictors.

Usage

```
data(marketing)
```

Format

the `marketing` dataset, as a data frame, contains 40 rows and 8 columns (variables/features). the 8 variables are:

- `spend`: daily send of money on PPC (apy-per-click).
- `clicks`: number of clicks on for that ad.
- `impressions`: amount of impressions per day.
- `display`: whether or not a display campaign was running.
- `transactions`: number of transactions per day.
- `click.rate`: click-through-rate.
- `conversion.rate`: conversion rate.
- `revenue`: daily revenue.

Details

A detailed description of the dataset can be found:

https://github.com/chrisBow/marketing-regression-part-one

See Also

adult, risk, churn, churnTel, bank, advertising, insurance, cereal, housePrice, house

Examples

```
data(marketing)

str(marketing)
```

---

minmax *Min-Max scaling of numerical variables*

Description

Performs Min-Max tranformation for numerical variables.

Usage

```
minmax(x, col = "auto", min = NULL, max = NULL, na.rm = FALSE)
```

Arguments

x	a numerical vector, matrix or data.frame.
col	a character vector of column names or indices. If "auto", all numeric columns will be transformed. If "all", all columns will be transformed.
min	a numerical value or vector indicating the minimum value(s) to use for Min-Max tranformation; if NULL, the default is based on x.
max	a numerical value or vector indicating the maximum value(s) to use for Min-Max tranformation; if NULL, the default is based on 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

scaler, zscore

Examples

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

minmax(x)

minmax(x, min = 0, max = 1)
```

`mse` *Mean Squared Error (MSE)*

Description

Computes mean squared error.

Usage

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

Arguments

`pred` a numerical vector of estimated values.

`actual` a numerical vector of actual 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`>

See Also

`mae`

Examples

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

mse(pred, actual)
```

one.hot *One Hot Encoder*

Description

One-Hot-Encode unordered factor columns of a data.frame, matrix, or data.table, using the mltools::one_hot() mltools::one_hot function.

Usage

```
one.hot(data, cols = "auto", sparsifyNAs = FALSE, naCols = FALSE,
                    dropCols = TRUE, dropUnusedLevels = FALSE)
```

Arguments

data	a numerical vector, matrix, data.frame, or data.table.
cols	a character vector of column names or indices to one-hot-encode. If "auto", all unordered factor columns will be one-hot-encoded.
sparsifyNAs	a logical value indicating whether to converte NAs to 0s.
naCols	a logical value indicating whether to create a separate column for NAs.
dropCols	a logical value indicating whether to drop the original columns which are one-hot-encoded.
dropUnusedLevels	a logical value indicating whether to drop unused factor levels.

Author(s)

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

See Also

scaler

Examples

```
data(risk)
str(risk)

risk_oh <- one.hot(risk, cols = "auto")
str(risk_oh)
```

`partition` *Partition the data*

Description

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

Usage

```
partition(data, ratio = c(0.7, 0.3), set.seed = NULL)
```

Arguments

`data`	an $(n \times p)$ `matrix` or a `data.frame`.
`ratio`	a numerical vector in range of [0, 1].
`set.seed`	a single value, interpreted as an integer, or NULL.

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, ratio = c(0.7, 0.3))
```

`redWines` *Red wines data set*

Description

the redWines datasets are related to red variants of the Portuguese "Vinho Verde" wine. Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available (e.g. there is no data about grape types, wine brand, wine selling price, etc.).

the dataset can be viewed as classification or regression tasks. the classes are ordered and not balanced (e.g. there are many more normal wines than excellent or poor ones). Outlier detection algorithms could be used to detect the few excellent or poor wines. Also, we are not sure if all input variables are relevant. So it could be interesting to test feature selection methods.

Usage

```
data(redWines)
```

Format

the redWines dataset, as a data frame, contains 1599 rows and 12 columns (variables/features). the 12 variables are:

Input variables (based on physicochemical tests):

- fixed acidity
- volatile acidity
- citric acid
- residual sugar
- chlorides
- free sulfur dioxide
- total sulfur dioxide
- density
- pH
- sulphates
- alcohol

Output variable (based on sensory data)

- quality: score between 0 and 10.

Details

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

https://archive.ics.uci.edu/dataset/186/wine+quality

References

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision support systems*, 47(4), 547-553.

See Also

whiteWines, adult, risk, churn, churnTel, bank, advertising, marketing, insurance, cereal, housePrice, house

Examples

```
data(redWines)

str(redWines)
```

`risk` *Risk data set*

Description

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

Usage

```
data(risk)
```

Format

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

- `age`: age in years.
- `marital`: A factor with levels "single", "married", and "other".
- `income`: yearly income.
- `mortgage`: A factor with levels "yes" and "no".
- `nr_loans`: Number of loans that constomers have.
- `risk`: A factor with levels "good risk" and "bad risk".

See Also

`adult`, `churn`, `churnTel`, `bank`, `advertising`, `marketing`, `insurance`, `cereal`, `housePrice`, `house`

Examples

```
data(risk)

str(risk)
```

`scaler` *Feature scaling*

Description

Performs feature scaling such as Z-score and min-max scaling.

Usage

```
scaler(x, scale = c("minmax", "zscore"), col = "auto",
       par1 = NULL, par2 = NULL, na.rm = FALSE)
```

Arguments

`x` a numerical `vector`, a `matrix` or a `data.frame`.

`scale` a transfer for `x`.

`col` a character vector of column names or indices. If `"auto"`, all numeric columns will be transformed. If `"all"`, all columns will be transformed.

`par1` a numerical value or vector that for the case `scale = "minmax"` indicating the maximum value(s) and for the case `scale = "zscore"` indicating the mean value(s).

`par2` a numerical value or vector that for the case `scale = "minmax"` indicating the maximum value(s) and for the case `scale = "zscore"` indicating the sd value(s).

`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)

scaler(x, scale = "minmax")

scaler(x, scale = "zscore")
```

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)
```

skim *Skim a data frame to get useful summary statistics*

Description

skim() provides an overview of a data frame asan alternative to summary(). This function is a wrapper for the skimr::skim() function of R package skimr.

Usage

```
skim(data, hist = TRUE, ...)
```

Arguments

data a data frame or matrix.

hist Logical: TRUE (default) to report the histogram of each variable.

... columns to select for skimming. the default is to skim all columns.

Author(s)

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

See Also

summary()

Examples

```
data(risk)

skim(risk)
```

`whiteWines` *White wines data set*

Description

the whiteWines datasets are related to white variants of the Portuguese "Vinho Verde" wine. Due to privacy and logistic issues, only physicochemical (inputs) and sensory (the output) variables are available (e.g. there is no data about grape types, wine brand, wine selling price, etc.).

the dataset can be viewed as classification or regression tasks. the classes are ordered and not balanced (e.g. there are many more normal wines than excellent or poor ones). Outlier detection algorithms could be used to detect the few excellent or poor wines. Also, we are not sure if all input variables are relevant. So it could be interesting to test feature selection methods.

Usage

```
data(whiteWines)
```

Format

the `whiteWines` dataset, as a data frame, contains 4898 rows and 12 columns (variables/features). the 12 variables are:

Input variables (based on physicochemical tests):

- `fixed acidity`
- `volatile acidity`
- `citric acid`
- `residual sugar`
- `chlorides`
- `free sulfur dioxide`
- `total sulfur dioxide`
- `density`
- `pH`
- `sulphates`
- `alcohol`

Output variable (based on sensory data)

- `quality`: score between 0 and 10.

Details

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

https://archive.ics.uci.edu/dataset/186/wine+quality

References

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., and Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision support systems*, 47(4), 547-553.

See Also

redWines, adult, risk, churn, churnTel, bank, advertising, marketing, insurance, cereal, housePrice, house

Examples

```
data(whiteWines)

str(whiteWines)
```

---

zscore *Z-score scaling of numerical variables*

Description

Performs Z-score tranformation for numerical variables.

Usage

```
zscore(x, col = "auto", mean = NULL, sd = NULL, na.rm = FALSE)
```

Arguments

x	a numerical vector, matrix or data.frame.
col	a character vector of column names or indices. If "auto", all numeric columns will be transformed. If "all", all columns will be transformed.
mean	a numerical value or vector indicating the mean to use for Z-score calculation; if NULL, the default is the mean of x.
sd	a numerical value or vector indicating the standard deviation(s) to use for Z-score calculation; if NULL, the default is the standard deviation of 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

scaler, minmax

Examples

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

zscore(x)
zscore(x, mean = 1, sd = 2)
```

Index