# Package 'dataSDA'

November 6, 2026

**Type** Package

**Title** Datasets and Basic Statistics for Symbolic Data Analysis

**Version** 0.2.7.4

**Date** 2026-11-06

**Description** Provides benchmark datasets and foundational tools for Symbolic Data Analysis (SDA). The package includes functions for constructing symbolic data objects from classical data, converting among different interval-valued data formats, managing interval-valued, histogram-valued, modal-valued, and multi-valued data, and performing basic descriptive statistics. It is designed to support teaching, methodological research, and the development of SDA techniques.

**License** GPL (>= 2)

**Encoding** UTF-8

**LazyData** true

**Depends** R (>= 4.0.0)

**Suggests** testthat (>= 2.1.0), knitr, rmarkdown, ggInterval, ggplot2, MAINT.Data, e1071, symbolicDA

**VignetteBuilder** knitr

**Imports** magrittr, tidyr, dplyr, RSDA, HistDAWass, methods

**NeedsCompilation** no

**Config/roxygen2/version** 8.1.0

**Author** Po-Wei Chen [aut],
Chun-houh Chen [aut],
Han-Ming Wu [cre]

**Maintainer** Han-Ming Wu <wuhm@g.nccu.edu.tw>

**Repository** CRAN

**Date/Publication** 2026-11-06 16:30:02 UTC

# Contents

abalone.iGAP . . . 5
abalone.int . . . 6
acid_rain.int . . . 7
age_cholesterol_weight.int . . . 8
age_pyramids.hist . . . 9
aggregate_to_symbolic . . . 10
airline_flights.hist . . . 13
airline_flights2.modal . . . 14
ARRAY_to_iGAP . . . 15
ARRAY_to_MM . . . 16
ARRAY_to_RSDA . . . 16
bank_rates . . . 17
baseball.int . . . 18
bats.int . . . 19
bird.mix . . . 20
bird_color_taxonomy.hist . . . 21
bird_species.mix . . . 22
bird_species_extended.mix . . . 23
blood.hist . . . 24
blood_pressure.int . . . 25
car.int . . . 26
cardiological.int . . . 27
cars.int . . . 28
car_models.int . . . 29
census.mix . . . 30
check_zero_width_intervals . . . 31
china_climate_month.hist . . . 32
china_climate_season.hist . . . 33
china_temp.int . . . 34
china_temp_monthly.int . . . 35
cholesterol.hist . . . 36
clean_colnames . . . 37
county_income_gender.hist . . . 37
cover_types.hist . . . 38
credit_card.int . . . 39
crime.modal . . . 41
crime2.modal . . . 42
crude_oil_wti.its . . . 43
djia.its . . . 44
ecoli_routes.int . . . 45
employment.int . . . 46
energy_consumption.distr . . . 47
energy_usage.distr . . . 48
environment.mix . . . 49
euro_usd.its . . . 51
exchange_rate_returns.hist . . . 52

face.iGAP . . . 53
finance.int . . . 54
flights_detail.hist . . . 55
french_agriculture.hist . . . 56
freshwater_fish.int . . . 57
fuel_consumption.modal . . . 58
fungi.int . . . 59
genome_abundances.int . . . 60
glucose.hist . . . 61
hardwood.hist . . . 62
hdi_gender.int . . . 64
health_insurance.mix . . . 65
health_insurance2.modal . . . 66
hematocrit.hist . . . 67
hematocrit_hemoglobin.hist . . . 68
hemoglobin.hist . . . 69
hierarchy . . . 70
hierarchy.hist . . . 71
hierarchy.int . . . 72
histogram_stats . . . 73
hist_extract . . . 75
horses.int . . . 76
hospital.hist . . . 77
household_characteristics.distr . . . 78
ibovespa.its . . . 79
iGAP_to_ARRAY . . . 80
iGAP_to_MM . . . 81
iGAP_to_RSDA . . . 82
interval_distance . . . 82
interval_geometry . . . 84
interval_position . . . 86
interval_robust . . . 88
interval_shape . . . 89
interval_similarity . . . 91
interval_stats . . . 92
interval_uncertainty . . . 94
int_convert_format . . . 95
int_detect_format . . . 97
int_list_conversions . . . 98
iris.int . . . 98
irish_wind.its . . . 99
iris_species.hist . . . 101
joggers.mix . . . 102
judge1.int . . . 103
judge2.int . . . 104
judge3.int . . . 105
lackinfo.int . . . 106
lisbon_air_quality.int . . . 107

loans_by_purpose.int . . . 108
loans_by_risk.int . . . 109
loans_by_risk_quantile.int . . . 110
lung_cancer.hist . . . 111
lynne1.int . . . 112
merval.its . . . 113
MM_to_ARRAY . . . 114
MM_to_iGAP . . . 115
MM_to_RSDA . . . 116
mtcars.mix . . . 116
mushroom.int . . . 117
mushroom.int.mm . . . 119
mushroom_fuzzy.mix . . . 120
nycflights.int . . . 121
occupations.modal . . . 122
occupations2.modal . . . 123
ohtemp.int . . . 124
oils.int . . . 125
ozone.hist . . . 126
petrobras.its . . . 127
polish_cars.mix . . . 129
polish_voivodships.int . . . 130
profession.int . . . 131
prostate.int . . . 132
read_symbolic_csv . . . 133
RSDA_format . . . 134
RSDA_to_ARRAY . . . 135
RSDA_to_iGAP . . . 136
RSDA_to_MM . . . 136
search_data . . . 137
set_variable_format . . . 138
shanghai_stock.its . . . 139
simulated.hist . . . 140
soccer_bivar.int . . . 141
SODAS_to_ARRAY . . . 142
SODAS_to_iGAP . . . 143
SODAS_to_MM . . . 143
sp500.its . . . 144
state_income.hist . . . 145
synthetic_clusters.int . . . 146
teams.int . . . 147
temperature_city.int . . . 148
tennis.int . . . 149
town_services.mix . . . 150
to_all_interval_formats . . . 151
trivial_intervals.int . . . 153
uscrime.int . . . 154
utsnow.int . . . 155

veterinary.int . . . . . 156
video1.int . . . . . 157
video2.int . . . . . 158
video3.int . . . . . 159
water_flow.int . . . . . 160
weight_age.hist . . . . . 161
wine.int . . . . . 162
world_cup.int . . . . . 163
write_symbolic_csv . . . . . 164

**Index** **166**

---

`abalone.iGAP` *Abalone Dataset (iGAP Format)*

---

## Description

Interval-valued dataset of 24 units from the UCI Abalone dataset, aggregated by sex and age group. iGAP format (comma-separated interval strings). See `abalone.int` for the Min-Max column format.

## Usage

```
data(abalone.iGAP)
```

## Format

A data frame with 24 observations (e.g., F-10-12, M-4-6) and 7 character columns in iGAP format (comma-separated `"min, max"` strings):

- `Length`: Shell length range.
- `Diameter`: Shell diameter range.
- `Height`: Shell height range.
- `Whole`: Whole weight range.
- `Shucked`: Shucked weight range.
- `Viscera`: Viscera weight range.
- `Shell`: Shell weight range.

Row names encode Sex-AgeGroup (e.g., F-10-12 = Female age 10–12).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 24 |
| **Variables (p)** | 7 |
| **Subject area** | Marine biology |
| **Symbolic format** | Interval (iGAP) |
| **Analytical tasks** | Clustering, Visualization |

### Source

Derived from `MAINT.Data::AbaloneIdt`, which aggregates the UCI Abalone microdata by sex and age group. UCI Machine Learning Repository, doi:10.24432/C55C7W.

### References

Nash, W. J., Sellers, T. L., Talbot, S. R., Cawthorn, A. J., and Ford, W. B. (1994). The Population Biology of Abalone (*Haliotis* species) in Tasmania. Sea Fisheries Division, Technical Report No. 48.

UCI Machine Learning Repository. doi:10.24432/C55C7W.

### Examples

```
data(abalone.iGAP)
```

---

## `abalone.int` *Abalone Interval Dataset*

---

### Description

Interval-valued dataset of 24 units from the UCI Abalone dataset, aggregated by sex and age group. Min-Max column format (two columns per variable). See `abalone.iGAP` for the iGAP format version.

### Usage

```
data(abalone.int)
```

### Format

A data frame with 24 observations and 14 columns (7 interval variables in `_min`/`_max` pairs):

- `Length_min`, `Length_max`: Shell length range.
- `Diameter_min`, `Diameter_max`: Shell diameter range.
- `Height_min`, `Height_max`: Shell height range.
- `Whole_min`, `Whole_max`: Whole weight range.

- `Shucked_min`, `Shucked_max`: Shucked weight range.
- `Viscera_min`, `Viscera_max`: Viscera weight range.
- `Shell_min`, `Shell_max`: Shell weight range.

Row names encode Sex-AgeGroup (e.g., F-10-12 = Female age 10–12).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 24 |
| **Variables (p)** | 14 |
| **Subject area** | Marine biology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering, Visualization |

## Source

Derived from `MAINT.Data::AbaloneIdt`, which aggregates the UCI Abalone microdata by sex and age group. UCI Machine Learning Repository, doi:10.24432/C55C7W.

## References

Nash, W. J., Sellers, T. L., Talbot, S. R., Cawthorn, A. J., and Ford, W. B. (1994). The Population Biology of Abalone (*Haliotis* species) in Tasmania. Sea Fisheries Division, Technical Report No. 48.

UCI Machine Learning Repository. doi:10.24432/C55C7W.

## Examples

```
data(abalone.int)
```

---

`acid_rain.int` *Acid Rain Pollution Indices Interval Dataset*

---

## Description

Interval-valued acid rain pollution indices for sulphates and nitrates (kg/hectares) for 2 US states (Massachusetts and New York).

## Usage

```
data(acid_rain.int)
```

## Format

A data frame with 2 observations and 5 variables in Min-Max format:

- `state`: State name (character).
- `sulphate_l`, `sulphate_u`: Sulphate pollution index range (kg/hectares).
- `nitrate_l`, `nitrate_u`: Nitrate pollution index range (kg/hectares).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 2 |
| **Variables (p)** | 5 |
| **Subject area** | Environment |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard and Diday (2006), Table 2.21.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.21.

## Examples

```
data(acid_rain.int)
```

---

`age_cholesterol_weight.int`

*Age-Cholesterol-Weight Interval Dataset*

---

## Description

Interval-valued dataset of 7 age-group observations with cholesterol and weight measurements. Each observation aggregates individuals in a 10-year age band with interval ranges for cholesterol and weight.

## Usage

```
data(age_cholesterol_weight.int)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 7 observations and 4 variables:

- `Age`: Age range (years, interval).
- `Cholesterol`: Cholesterol level range (mg/dL, interval).
- `Weight`: Weight range (pounds, interval).
- `n`: Number of individuals in the age group (numeric).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 7 |
| **Variables (p)** | 4 |
| **Subject area** | Medical |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Regression |

### Source

Billard and Diday (2006).

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

### Examples

```
data(age_cholesterol_weight.int)
```

---

## `age_pyramids.hist` *World Age Pyramids Histogram-Valued Dataset (2014)*

---

### Description

Histogram-valued dataset of 229 countries with 3 population age pyramid histograms (both sexes, male, female). Each histogram has 21 age bins representing the distribution of the population across age groups.

### Usage

```
data(age_pyramids.hist)
```

### Format

A data frame with 229 observations (countries) and 3 histogram-valued variables:

- `Both.Sexes.Population`: Histogram of total population by age group.
- `Male.Population`: Histogram of male population by age group.
- `Female.Population`: Histogram of female population by age group.

Row names are country names (e.g., WORLD, Afghanistan, Albania).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 229 |
| **Variables (p)** | 3 |
| **Subject area** | Demographics |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering, Descriptive statistics |

### Source

`HistDAWass::Age_Pyramids_2014`; the upstream documentation identifies the U.S. Census Bureau as the original source.

### References

U.S. Census Bureau (2014). International population estimates and projections by age and sex.

### Examples

```
data(age_pyramids.hist)
```

---

## `aggregate_to_symbolic` *Aggregate Tabular Data to Symbolic Data*

---

### Description

Aggregate tabular numerical data (n by p) into interval-valued or histogram-valued symbolic data (K by p) based on a grouping mechanism.

### Usage

```
aggregate_to_symbolic(x, type = "int", group_by = "kmeans",
  stratify_var = NULL, K = 5, interval = "range",
  quantile_probs = c(0.05, 0.95), bins = 10, nK = NULL,
  zero_width = c("keep", "remove", "regenerate", "adjust"), epsilon = 1e-07)
```

## Arguments

| | |
|---|---|
| x | A data.frame with n rows and p columns. May contain non-numeric columns used for grouping or stratification; only numeric columns are aggregated. |
| type | Output symbolic type: "int" for interval data or "hist" for histogram data. |
| group_by | Grouping mechanism. One of:<br>"kmeans" Partition the data into K groups using k-means clustering.<br>"hclust" Partition the data into K groups using hierarchical clustering.<br>"resampling" Generate K concepts by randomly sampling nK observations with replacement, repeated K times.<br>**A column name or column index** Use the specified categorical variable to define groups. |
| stratify_var | Optional column name or index for a stratification variable. When provided, grouping and aggregation are performed independently within each level. Default is NULL. |
| K | Number of groups for clustering (group_by = "kmeans" or "hclust") or resampling (group_by = "resampling"). Ignored when group_by is a variable. Default is 5. |
| interval | Interval construction method when type = "int": "range" uses min/max; "quantile" uses quantiles given by quantile_probs. Default is "range". |
| quantile_probs | Numeric vector of length 2 giving the lower and upper quantile probabilities for interval = "quantile". Default is c(0.05, 0.95). |
| bins | Number of histogram bins when type = "hist". Default is 10. |
| nK | Number of observations to sample per group when group_by = "resampling". Default is floor(n / K). |
| zero_width | How to handle zero-width intervals (min == max) produced when type = "int". Such degenerate intervals break downstream tools that divide by interval width (e.g. ggInterval::ggInterval_indexImage()). One of:<br>"keep" (default) Leave the aggregated output unchanged; zero-width intervals are returned as-is and no action is taken. Use check_zero_width_intervals to screen the result.<br>"remove" Drop every concept (row) that contains at least one zero-width interval.<br>"regenerate" Re-run the aggregation (re-clustering or re-sampling) until no zero-width interval remains. Only effective for stochastic group_by ("kmeans", "resampling"); for deterministic grouping (a variable or "hclust") the result cannot change, so an error is raised suggesting another option.<br>"adjust" Add a small amount epsilon to the upper endpoint of each zero-width interval.<br>Ignored when type = "hist". |
| epsilon | Positive amount added to the upper endpoint of each zero-width interval when zero_width = "adjust". Default is 1e-07. |

## Details

The function aggregates classical tabular data into symbolic data by:

1. Partitioning observations into groups via `group_by` (clustering, resampling, or a categorical variable).
2. Within each group, summarizing each numeric variable as an interval (min/max or quantiles) or a histogram.

When `stratify_var` is provided, grouping and aggregation are performed within each level of the stratification variable. Label values are prefixed by the stratum name (e.g., `"setosa.cluster_1"`).

For `type = "hist"`, bin boundaries are computed from the global data range to ensure comparability across groups.

Non-numeric columns (other than those used for grouping or stratification) are silently excluded from aggregation.

## Value

- For `type = "int"`: a `symbolic_tbl` (RSDA format) with a label column followed by `symbolic_interval` columns for each numeric variable (K rows, 1 + p columns).
- For `type = "hist"`: a `MatH` object (K rows by p columns of histogram-valued data).

## Examples

```
# Group by a categorical variable -> interval data
res1 <- aggregate_to_symbolic(iris, type = "int", group_by = "Species")
res1

# K-means clustering -> interval data
res2 <- aggregate_to_symbolic(iris[, 1:4], type = "int",
                              group_by = "kmeans", K = 3)

# Quantile-based intervals
res3 <- aggregate_to_symbolic(iris[, 1:4], type = "int",
                              group_by = "kmeans", K = 3,
                              interval = "quantile",
                              quantile_probs = c(0.1, 0.9))

# Resampling -> interval data
set.seed(42)
res4 <- aggregate_to_symbolic(iris[, 1:4], type = "int",
                              group_by = "resampling", K = 5, nK = 30)

# Histogram aggregation
res5 <- aggregate_to_symbolic(iris, type = "hist",
                              group_by = "Species", bins = 5)

# Hierarchical clustering -> interval data
res6 <- aggregate_to_symbolic(iris[, 1:4], type = "int",
                              group_by = "hclust", K = 3)
```

```
# Stratified aggregation
res7 <- aggregate_to_symbolic(iris, type = "int",
                              group_by = "kmeans", K = 2,
                              stratify_var = "Species")
```

---

`airline_flights.hist` *JFK Airport Airline Flights Histogram-Valued Dataset*

---

## Description

Histogram-valued dataset of 16 airlines flying into JFK Airport. Six variables (Flight Time, Taxi In, Arrival Delay, Taxi Out, Departure Delay, Weather Delay) recorded as frequency distributions. This is the wide (flat table) format; see `airline_flights2.modal` for the modal-valued version.

## Usage

```
data(airline_flights.hist)
```

## Format

A data frame with 16 observations (Airline1–Airline16) and 17 numeric columns representing 6 histogram variables in wide format:

- `Flight Time(<120)`, `Flight Time([120, 220])`, `Flight Time(>220)`: Flight time distribution (3 bins).
- `Taxi In(<4)`, `Taxi In([4, 10])`, `Taxi In(>10)`: Taxi-in time distribution (3 bins).
- `Arrival Delay(<0)`, `Arrival Delay([0, 60])`, `Arrival Delay(>60)`: Arrival delay distribution (3 bins).
- `Taxi Out(<16)`, `Taxi Out([16, 30])`, `Taxi Out(>30)`: Taxi-out time distribution (3 bins).
- `Departure Delay(<0)`, `Departure Delay([0, 60])`, `Departure Delay(>60)`: Departure delay distribution (3 bins).
- `Weather Delay(No)`, `Weather Delay(Yes)`: Weather delay distribution (2 bins).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 16 |
| **Variables (p)** | 17 |
| **Subject area** | Transportation |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Billard and Diday (2006), Table 2.7.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.7.

## Examples

```
data(airline_flights.hist)
```

---

`airline_flights2.modal`

*JFK Airport Airline Flights Modal-Valued Dataset*

---

## Description

Modal-valued version of the airline flights dataset. See `airline_flights.hist` for the wide-format version.

## Usage

```
data(airline_flights2.modal)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 16 observations and 6 modal-valued variables:

- `FlightTime`: Modal distribution over flight time bins.
- `TaxiIn`: Modal distribution over taxi-in time bins.
- `ArrivalDelay`: Modal distribution over arrival delay bins.
- `TaxiOut`: Modal distribution over taxi-out time bins.
- `DepartureDelay`: Modal distribution over departure delay bins.
- `WeatherDelay`: Modal distribution over weather delay bins.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 16 |
| **Variables (p)** | 6 |
| **Subject area** | Transportation |
| **Symbolic format** | Modal |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Billard and Diday (2006), Table 2.7.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.7.

## Examples

```
data(airline_flights2.modal)
```

---

`ARRAY_to_iGAP` *ARRAY to iGAP*

---

## Description

Convert a 3-dimensional array `[n, p, 2]` to iGAP format (data.frame with comma-separated interval values).

## Usage

```
ARRAY_to_iGAP(data)
```

## Arguments

| | |
|---|---|
| `data` | A numeric array of dimension `[n, p, 2]` where `[,,1]` stores minima and `[,,2]` stores maxima. |

## Value

A data.frame in iGAP format with comma-separated `"min,max"` values.

## Examples

```
x <- array(NA, dim = c(4, 3, 2))
x[,,1] <- matrix(c(1,2,3,4, 5,6,7,8, 9,10,11,12), nrow = 4)
x[,,2] <- matrix(c(3,5,6,7, 8,9,10,12, 13,15,16,18), nrow = 4)
dimnames(x) <- list(paste0("obs_", 1:4), c("V1","V2","V3"), c("min","max"))
igap <- ARRAY_to_iGAP(x)
igap
```

## ARRAY_to_MM *ARRAY to MM*

### Description

Convert a 3-dimensional array [n, p, 2] to MM format (data.frame with paired _min/_max columns).

### Usage

```
ARRAY_to_MM(data)
```

### Arguments

| | |
|---|---|
| data | A numeric array of dimension [n, p, 2] where [,,1] stores minima and [,,2] stores maxima. |

### Value

A data.frame with 2p columns (paired _min/_max).

### Examples

```
x <- array(NA, dim = c(4, 3, 2))
x[,,1] <- matrix(c(1,2,3,4, 5,6,7,8, 9,10,11,12), nrow = 4)
x[,,2] <- matrix(c(3,5,6,7, 8,9,10,12, 13,15,16,18), nrow = 4)
dimnames(x) <- list(paste0("obs_", 1:4), c("V1","V2","V3"), c("min","max"))
mm <- ARRAY_to_MM(x)
mm
```

## ARRAY_to_RSDA *ARRAY to RSDA*

### Description

Convert a 3-dimensional array [n, p, 2] to RSDA format (symbolic_tbl with symbolic_interval columns).

### Usage

```
ARRAY_to_RSDA(data)
```

### Arguments

| | |
|---|---|
| data | A numeric array of dimension [n, p, 2] where [,,1] stores minima and [,,2] stores maxima. |

## Value

A symbolic_tbl with p symbolic_interval columns.

## Examples

```
x <- array(NA, dim = c(4, 3, 2))
x[,,1] <- matrix(c(1,2,3,4, 5,6,7,8, 9,10,11,12), nrow = 4)
x[,,2] <- matrix(c(3,5,6,7, 8,9,10,12, 13,15,16,18), nrow = 4)
dimnames(x) <- list(paste0("obs_", 1:4), c("V1","V2","V3"), c("min","max"))
rsda <- ARRAY_to_RSDA(x)
rsda
```

---

# bank_rates *Bank Interest Rates AR Model Symbolic Dataset*

---

## Description

Symbolic dataset of autoregressive time series models for 4 banks. Each bank is described by AR model order, parameters, and whether parameters are known.

## Usage

```
data(bank_rates)
```

## Format

A data frame with 4 observations (Bank1–Bank4) and 6 variables:

- `bank`: Bank identifier (character).
- `order`: AR model order (numeric).
- `phi1`: First AR parameter (numeric; NA if unknown).
- `phi2`: Second AR parameter (numeric; NA if order < 2 or unknown).
- `phi1_known`: Whether phi1 is known (logical).
- `phi2_known`: Whether phi2 is known (logical; NA if order < 2).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 4 |
| **Variables (p)** | 6 |
| **Subject area** | Finance |
| **Symbolic format** | Symbolic (model-valued) |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard and Diday (2006), Table 2.9.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.9.

## Examples

```
data(bank_rates)
```

---

`baseball.int` *Baseball Teams Interval Dataset*

---

## Description

Interval-valued data for 19 baseball teams with aggregated player batting statistics and a pattern variable classifying team performance.

## Usage

```
data(baseball.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 19 observations and 3 variables:

- `At_Bats`: Range of at-bats across players (interval).
- `Hits`: Range of hits across players (interval).
- `Pattern`: Team performance pattern code (character).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 19 |
| **Variables (p)** | 3 |
| **Subject area** | Sports |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Clustering |

## Source

Billard and Diday (2006).

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

## Examples

```
data(baseball.int)
```

---

`bats.int` *Bat Species Interval Dataset*

---

## Description

Interval-valued data for 21 bat species described by 4 morphological measurements. Benchmark dataset for matrix visualization.

## Usage

```
data(bats.int)
```

## Format

A data frame with 21 observations and 9 columns (4 interval variables in _l/_u Min-Max pairs, plus a label):

- `species`: Bat species name (character).
- `head_l`, `head_u`: Head length range (mm).
- `tail_l`, `tail_u`: Tail length range (mm).
- `height_l`, `height_u`: Ear height range (cm).
- `forearm_l`, `forearm_u`: Forearm length range (mm).

## Details

Used to demonstrate color coding schemes, the HCT-R2E seriation algorithm, and distance measure comparisons (Gowda-Diday, Hausdorff, City-Block, L1, L2, etc.) for interval data.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 21 |
| **Variables (p)** | 9 |
| **Subject area** | Zoology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering, Visualization |

## Source

The interval-valued bat data analyzed by Kao et al. (2014).

## References

Kao, C.-H., Nakano, J., Shieh, S.-H., Tien, Y.-J., Wu, H.-M., Yang, C.-K. and Chen, C.-H. (2014). Exploratory data analysis of interval-valued symbolic data with matrix visualization. *Computational Statistics & Data Analysis*, 79, 14–29. doi:10.1016/j.csda.2014.04.012.

## Examples

```
data(bats.int)
```

---

# bird.mix *Bird Species Mixed Symbolic Dataset*

---

## Description

Interval-valued morphological measurements for 20 bird specimens. Despite the `.mix` suffix, this dataset contains only interval-valued variables (density and size).

## Usage

```
data(bird.mix)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 20 observations and 2 variables:

- `Density`: Feather density range (interval).
- `Size`: Body size range (cm, interval).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 20 |
| **Variables (p)** | 2 |
| **Subject area** | Zoology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard and Diday (2006), Table 2.5.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.5.

## Examples

```
data(bird.mix)
```

---

`bird_color_taxonomy.hist`

*Bird Color Taxonomy Histogram Dataset*

---

## Description

Mixed symbolic dataset of 20 bird observations with histogram-valued feather density and body size, categorical tone, and distribution-valued shade (fuzzy taxonomy). From Tables 6.9 and 6.14 of Billard and Diday (2006).

## Usage

```
data(bird_color_taxonomy.hist)
```

## Format

A data frame with 20 observations and 4 variables:

- `density`: Histogram-valued feather density (up to 4 bins).
- `size`: Histogram-valued body size (2-bin).
- `tone`: Categorical tone (dark/light).
- `shade`: Distribution-valued shade (purple/red/white/yellow with fuzzy weights).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 20 |
| **Variables (p)** | 4 |
| **Subject area** | Zoology |
| **Symbolic format** | Mixed (histogram, categorical, distribution) |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Billard, L. and Diday, E. (2006), Tables 6.9 and 6.14.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Tables 6.9 and 6.14.

## Examples

```
data(bird_color_taxonomy.hist)
```

---

`bird_species.mix` *Bird Species Mixed Symbolic Dataset*

---

## Description

Symbolic data for 3 bird species (Swallow, Ostrich, Penguin) with interval-valued size, categorical flying, and categorical migration. Foundational SDA example from 600 individual bird observations.

## Usage

```
data(bird_species.mix)
```

## Format

A data frame with 3 observations (Swallow, Ostrich, Penguin) and 5 variables:

- `species`: Species name (character).
- `flying`: Flying ability (Yes/No, character).
- `size_l`, `size_u`: Size range (cm, Min-Max pair).
- `migration`: Migratory behavior (TRUE/FALSE, logical).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 3 |
| **Variables (p)** | 5 |
| **Subject area** | Zoology |
| **Symbolic format** | Mixed (interval, categorical) |
| **Analytical tasks** | Descriptive statistics |

## Source

Diday and Noirhomme-Fraiture (2008), Table 1.2, p. 6.

## References

Diday, E. and Noirhomme-Fraiture, M. (Eds.) (2008). *Symbolic Data Analysis and the SODAS Software*. Wiley. Table 1.2, p.6.

## Examples

```
data(bird_species.mix)
```

---

`bird_species_extended.mix`

*Bird Species Extended Mixed Symbolic Dataset*

---

## Description

Three bird species (Geese, Ostrich, Penguin) with interval-valued height, distribution-valued color, and categorical flying/migratory variables.

## Usage

```
data(bird_species_extended.mix)
```

## Format

A data frame with 3 observations and 6 variables:

- `species`: Species name (character).
- `flying`: Flying ability (Yes/No, character).
- `height_l`: Height lower bound (cm, numeric).
- `height_u`: Height upper bound (cm, numeric).
- `color`: Color distribution as weighted set string (e.g., "{white, 0.3; black, 0.7}").
- `migratory`: Migratory behavior (Yes/No, character).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 3 |
| **Variables (p)** | 6 |
| **Subject area** | Zoology |
| **Symbolic format** | Mixed (interval, categorical, distribution) |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard and Diday (2006), Table 2.19.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.19.

## Examples

```
data(bird_species_extended.mix)
```

---

`blood.hist` *Blood Test Histogram Dataset*

---

## Description

Histogram-valued blood test results for 14 gender-age groups (e.g., Female-20, Male-50). Each observation contains histograms for cholesterol, hemoglobin, and hematocrit, represented as multi-bin distributions.

## Usage

```
data(blood.hist)
```

## Format

A data frame with 14 observations and 3 histogram-valued variables:

- `Cholesterol`: Histogram of cholesterol levels (mg/dL).
- `Hemoglobin`: Histogram of hemoglobin levels (g/dL).
- `Hematocrit`: Histogram of hematocrit levels (%).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 14 |
| **Variables (p)** | 3 |
| **Subject area** | Medical |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Descriptive statistics, Clustering |

## Source

`HistDAWass::BLOOD`; Irpino and Verde (2015) analyze the data and attribute them to Billard and Diday (2006).

## References

Irpino, A. and Verde, R. (2015). Basic statistics for distributional symbolic variables: a new metric-based approach. *Advances in Data Analysis and Classification*, 9(2), 143–175.

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester.

## Examples

```
data(blood.hist)
```

---

# blood_pressure.int *Blood Pressure Interval Dataset*

---

## Description

Interval-valued blood pressure and pulse rate measurements for 15 patient groups.

## Usage

```
data(blood_pressure.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 15 observations and 3 interval-valued variables:

- `Pulse_Rate`: Pulse rate range (beats per minute, interval).
- `Systolic_Pressure`: Systolic blood pressure range (mmHg, interval).
- `Diastolic_Pressure`: Diastolic blood pressure range (mmHg, interval).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 15 |
| **Variables (p)** | 3 |
| **Subject area** | Medical |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Regression |

## Source

Billard and Diday (2006).

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

## Examples

```
data(blood_pressure.int)
```

---

`car.int` *Car Models Interval Dataset*

---

## Description

Interval-valued data for 8 car brands with price and performance specifications. Each brand aggregates multiple models into interval ranges.

## Usage

```
data(car.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 8 observations and 5 variables:

- `Car`: Car brand name (character).
- `Price`: Price range (thousands of currency units, interval).
- `Max_Velocity`: Maximum velocity range (km/h, interval).
- `Accn_Time`: Acceleration time range (seconds 0–100 km/h, interval).
- `Cylinder_Capacity`: Engine cylinder capacity range (cc, interval).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 8 |
| **Variables (p)** | 5 |
| **Subject area** | Automotive |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Clustering |

## Source

Billard and Diday (2006).

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

## Examples

```
data(car.int)
```

`cardiological.int` *Cardiological Examination Interval Dataset*

## Description

Interval-valued data from cardiological examinations of 44 patients. Each patient is described by 5 interval-valued physiological measurements.

## Usage

```
data(cardiological.int)
```

## Format

A data frame with 44 observations and 5 interval-valued variables:

- `pulse`: Pulse rate range (beats per minute).
- `systolic`: Systolic blood pressure range (mmHg).
- `diastolic`: Diastolic blood pressure range (mmHg).
- `arterial1`: First arterial measurement range.
- `arterial2`: Second arterial measurement range.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 44 |
| **Variables (p)** | 5 |
| **Subject area** | Medical |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Clustering |

## Source

Extracted from `RSDA::cardiologicalv2`; the RSDA documentation attributes the data to Billard and Diday (2006).

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester.

## Examples

```
data(cardiological.int)
```

`cars.int` *Cars Interval Dataset*

## Description

Interval-valued data for 27 car models classified into four classes (Utilitarian, Berlina, Sportive, Luxury), described by Price, EngineCapacity, TopSpeed and Acceleration intervals.

## Usage

```
data(cars.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 27 observations and 5 variables:

- `Price`: Price range (interval).
- `EngCap`: Engine capacity range (cc, interval).
- `TopSpeed`: Top speed range (km/h, interval).
- `Acceleration`: Acceleration range (seconds 0–100 km/h, interval).
- `class`: Car class (Utilitarian, Berlina, Sportive, Luxury; set-valued).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 27 |
| **Variables (p)** | 5 |
| **Subject area** | Automotive |
| **Symbolic format** | Interval |
| **Analytical tasks** | Classification |

## Source

`MAINT.Data::Cars`, containing interval specifications for 27 car models in four classes.

## References

Duarte Silva, A. P., Brito, P., Filzmoser, P. and Dias, J. G. (2021). MAINT.Data: Modelling and Analysing Interval Data in R. *The R Journal*, 13(2), 336–364. doi:10.32614/RJ2021074.

## Examples

```
data(cars.int)
```

`car_models.int` *Italian Car Models Interval Dataset*

## Description

Interval-valued specifications for 33 Italian car models, classified into 4 categories (Utilitaria, Berlina, Ammiraglia, Sportiva). An extended version of the classic cars interval dataset with 8 interval-valued variables including dimensions.

## Usage

```
data(car_models.int)
```

## Format

A data frame with 33 observations and 9 variables:

- `price`: Price range (currency units).
- `engine_cc`: Engine displacement range (cc).
- `top_speed`: Top speed range (km/h).
- `acceleration`: Acceleration range (seconds 0-100 km/h).
- `wheelbase`: Wheelbase range (cm).
- `length`: Length range (cm).
- `width`: Width range (cm).
- `height`: Height range (cm).
- `class`: Car category (Utilitaria, Berlina, Ammiraglia, Sportiva).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 33 |
| **Variables (p)** | 9 |
| **Subject area** | Automotive |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering, Classification |

## Source

https://github.com/Natandradesa/Kernel-Clustering-for-Interval-Data

## References

Andrade de Sa, J. N., Ferreira, M. R. P. and de Carvalho, F. A. T. (2025). Kernel clustering with automatic variable weighting for interval data. *Neurocomputing*, 650, 130849. doi:10.1016/j.neucom.2025.130849.

## Examples

```
data(car_models.int)
```

---

`census.mix` *Census Mixed Symbolic Dataset*

---

## Description

Mixed symbolic dataset of 10 census regions combining 6 different symbolic variable types: histograms (age, home value), distributions (gender, tenure), a multi-valued set (fuel), and an interval (income).

## Usage

```
data(census.mix)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 10 observations (regions) and 6 variables:

- `age`: Histogram-valued age distribution (12 age bins).
- `home_value`: Histogram-valued home value distribution (7 value bins, in $1000s).
- `gender`: Distribution over gender (male, female).
- `fuel`: Multi-valued set of fuel types used.
- `tenure`: Distribution over housing tenure (owner, renter, vacant).
- `income`: Interval-valued household income range ($1000s).

Row names are Region_1 through Region_10.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 6 |
| **Subject area** | Demographics |
| **Symbolic format** | Mixed (interval, histogram, distribution, multi-valued) |
| **Analytical tasks** | Clustering |

## Source

Billard, L. and Diday, E. (2020), Table 7-23.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 7-23.

## Examples

```
data(census.mix)
```

---

`check_zero_width_intervals` *Check for Zero-Width Intervals*

---

## Description

Checks whether interval-valued data contains any zero-width intervals (`min == max`). Such degenerate intervals break downstream tools that divide by interval width (e.g. `ggInterval::ggInterval_indexImage()`). The input may be supplied either in MM format (a `data.frame` with paired `_min`/`_max` columns) or in RSDA format (a `symbolic_tbl` with `symbolic_interval` columns).

## Usage

```
check_zero_width_intervals(data, tol = 0, warn = TRUE)
```

## Arguments

| | |
|---|---|
| `data` | Interval-valued data, in one of two accepted formats:<br>• a `data.frame` in MM format with paired `_min`/`_max` columns, or<br>• a `symbolic_tbl` (RSDA format) with `symbolic_interval` columns. Non-interval columns (e.g. set/modal variables) are ignored. |
| `tol` | Non-negative numeric tolerance. An interval is flagged when `abs(max - min) <= tol`. Defaults to `0` (exact `min == max`). |
| `warn` | Logical; if `TRUE` (default) a single warning naming the affected variables is emitted when zero-width intervals are found. |

## Value

Invisibly, a logical scalar: `TRUE` if any zero-width interval is present, otherwise `FALSE`. The returned value carries two attributes: `"flagged"`, a logical `[n, p]` matrix marking each zero-width cell (rows = concepts, columns = interval variables), and `"variables"`, the names of variables containing at least one zero-width interval.

## Examples

```
# MM format (paired _min/_max columns)
data(mushroom.int.mm)
check_zero_width_intervals(mushroom.int.mm)

# RSDA format (symbolic_tbl)
data(mushroom.int)
check_zero_width_intervals(mushroom.int)
```

`china_climate_month.hist`

*Chinese Climate Monthly Histogram Dataset*

## Description

Histogram-valued monthly climate data for 60 Chinese weather stations. Each station has 14 climate variables measured across 12 months (168 histogram columns total). Histograms are reduced to 10 decile bins from the original HistDAWass distributions.

## Usage

```
data(china_climate_month.hist)
```

## Format

A data frame with 60 observations (stations) and 168 histogram-valued variables. Variables follow the pattern `variable_Month` (e.g., `mean.temp_Jan`). The 14 climate variables are: mean pressure, mean temperature, mean max/min temperature, total precipitation, sunshine duration, mean cloud amount, mean relative humidity, snow days, dominant wind direction, mean wind speed, dominant wind frequency, extreme max/min temperature.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 60 |
| **Variables (p)** | 168 |
| **Subject area** | Climate |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering |

## Source

Derived from `HistDAWass::China_Month`; the upstream source is the Climate Databases of the People's Republic of China, 1841–1988.

## References

Carbon Dioxide Information Analysis Center. *Climate Databases of the People's Republic of China, 1841–1988*. doi:10.3334/CDIAC/CLI.TR055.

## Examples

```
data(china_climate_month.hist)
```

`china_climate_season.hist`

*Chinese Climate Seasonal Histogram Dataset*

## Description

Histogram-valued seasonal climate data for 60 Chinese weather stations. Each station has 14 climate variables measured across 4 seasons (56 histogram columns total). Histograms are reduced to 10 decile bins from the original HistDAWass distributions.

## Usage

```
data(china_climate_season.hist)
```

## Format

A data frame with 60 observations (stations) and 56 histogram-valued variables. Variables follow the pattern `variable_Season` (e.g., `mean.temp_Spring`). The 14 climate variables are: mean pressure, mean temperature, mean max/min temperature, total precipitation, sunshine duration, mean cloud amount, mean relative humidity, snow days, dominant wind direction, mean wind speed, dominant wind frequency, extreme max/min temperature.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 60 |
| **Variables (p)** | 56 |
| **Subject area** | Climate |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering |

## Source

Derived from `HistDAWass::China_Seas`; the upstream source is the Climate Databases of the People's Republic of China, 1841–1988.

## References

Carbon Dioxide Information Analysis Center. *Climate Databases of the People's Republic of China, 1841–1988*. doi:10.3334/CDIAC/CLI.TR055.

## Examples

```
data(china_climate_season.hist)
```

china_temp.int *China Meteorological Stations Quarterly Temperature Interval Dataset*

## Description

Interval-valued temperature data (Celsius) for 60 Chinese meteorological stations observed over the four quarters of years 1974 to 1988. One outlier observation (YinChuan_1982) has been discarded.

## Usage

```
data(china_temp.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 899 observations and 5 variables:

- `Q1`: Quarter 1 (Jan–Mar) temperature range (tenths of degrees Celsius, interval).
- `Q2`: Quarter 2 (Apr–Jun) temperature range (interval).
- `Q3`: Quarter 3 (Jul–Sep) temperature range (interval).
- `Q4`: Quarter 4 (Oct–Dec) temperature range (interval).
- `GeoReg`: Geographic region classification (factor).

## Details

Originates from the Long-Term Instrumental Climatic Database of the People's Republic of China. Widely used in the SDA literature for demonstrating standardization, clustering, self-organizing maps, MLE and MANOVA.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 899 |
| **Variables (p)** | 5 |
| **Subject area** | Climate |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

`MAINT.Data::ChinaTemp`, derived from the Long-Term Instrumental Climatic Database of the People's Republic of China.

## References

Brito, P. and Duarte Silva, A. P. (2012). Modelling interval data with Normal and Skew-Normal distributions. *Journal of Applied Statistics*, 39(1), 3–20. doi:10.1080/02664763.2011.575125.

## Examples

```
data(china_temp.int)
```

---

`china_temp_monthly.int`

*China Monthly Temperature Intervals (15 Stations)*

---

## Description

Interval-valued dataset of monthly temperature ranges for 15 weather stations in China. Each station has 12 monthly temperature intervals (minimum and maximum observed temperatures in degrees Celsius) and an elevation value in meters.

## Usage

```
data(china_temp_monthly.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 15 observations (weather stations) and 13 variables:

- `January`, `February`, `March`, `April`, `May`, `June`, `July`, `August`, `September`, `October`, `November`, `December`: Interval-valued monthly temperature ranges (degrees Celsius).
- `Elevation`: Station elevation above sea level (numeric, meters).

Row names are station names (e.g., BoKeTu, Hailaer, LaSa).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 15 |
| **Variables (p)** | 13 |
| **Subject area** | Climate |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

Billard, L. and Diday, E. (2020), Table 7-9.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 7-9.

## Examples

```
data(china_temp_monthly.int)
```

---

`cholesterol.hist` *Cholesterol by Gender and Age Histogram-Valued Dataset*

---

## Description

Histogram-valued cholesterol distributions for 14 gender-age groups (7 female + 7 male age groups from 20s to 80+). Each observation has a 10-bin histogram of cholesterol levels.

## Usage

```
data(cholesterol.hist)
```

## Format

A data frame with 14 observations and 3 variables:

- `gender`: Gender (Female or Male).
- `age`: Age group (20s, 30s, 40s, 50s, 60s, 70s, 80+).
- `cholesterol`: Histogram-valued cholesterol distribution.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 14 |
| **Variables (p)** | 3 |
| **Subject area** | Medical |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard, L. and Diday, E. (2006), Table 4.5.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Table 4.5.

## Examples

```
data(cholesterol.hist)
```

---

`clean_colnames` *clean_colnames*

---

## Description

This function is used to clean up variable names to conform to the RSDA format.

## Usage

```
clean_colnames(data)
```

## Arguments

| | |
|---|---|
| `data` | The conventional data. |

## Value

Data after cleaning variable names.

## Examples

```
data(mushroom.int.mm)
mushroom.clean <- clean_colnames(data = mushroom.int.mm)
```

---

`county_income_gender.hist`
*County Income by Gender Histogram-Valued Dataset*

---

## Description

Histogram-valued dataset of 12 counties with gender-stratified income histograms and sample sizes. Each county has a male income histogram, a female income histogram, and the number of respondents in each group.

## Usage

```
data(county_income_gender.hist)
```

## Format

A data frame with 12 observations (counties) and 4 variables:

- `male_income`: Histogram of male household income (4 bins from $0 to $100k).
- `female_income`: Histogram of female household income (4 bins from $0 to $100k).
- `n_males`: Number of male respondents (numeric).
- `n_females`: Number of female respondents (numeric).

Row names are County_1 through County_12.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 12 |
| **Variables (p)** | 4 |
| **Subject area** | Economics |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Billard, L. and Diday, E. (2020), Table 6-16.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 6-16.

## Examples

```
data(county_income_gender.hist)
```

---

`cover_types.hist` *Forest Cover Types Histogram-Valued Dataset*

---

## Description

Histogram-valued dataset of 7 forest cover types with 4 topographic histogram variables. Each histogram describes the distribution of a terrain feature across locations classified as that cover type.

## Usage

```
data(cover_types.hist)
```

### Format

A data frame with 7 observations (cover types) and 4 histogram-valued variables:

- `elevation`: Histogram of elevation values (meters).
- `distance_to_water`: Histogram of horizontal distance to nearest water source (meters).
- `hillshade`: Histogram of hillshade index values.
- `slope`: Histogram of slope values (degrees).

Row names are CoverType_1 through CoverType_7.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 7 |
| **Variables (p)** | 4 |
| **Subject area** | Forestry |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering, Classification |

### Source

Billard, L. and Diday, E. (2020), Table 7-21.

### References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 7-21.

### Examples

```
data(cover_types.hist)
```

---

## `credit_card.int` *Credit Card Expenses Interval Dataset*

---

### Description

Interval-valued credit card spending aggregated by person-month. Three individuals' (Jon, Tom, Leigh) monthly expenditures across five categories.

### Usage

```
data(credit_card.int)
```

## Format

A data frame with 6 observations and 11 columns (5 interval variables in _l/_u Min-Max pairs, plus a label):

- person_month: Person-month identifier (e.g., "Jon - January"; character).
- food_l, food_u: Food expenditure range (USD).
- social_l, social_u: Social expenditure range (USD).
- travel_l, travel_u: Travel expenditure range (USD).
- gas_l, gas_u: Gas expenditure range (USD).
- clothes_l, clothes_u: Clothes expenditure range (USD).

## Details

The original classical dataset (Table 2.3) records individual transactions. The symbolic version (Table 2.4) aggregates into interval-valued observations for each person-month combination.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 6 |
| **Variables (p)** | 11 |
| **Subject area** | Finance |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard and Diday (2006), Tables 2.3–2.4.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Tables 2.3-2.4.

## Examples

```
data(credit_card.int)
```

`crime.modal` *Crime Demographics Dataset*

## Description

Modal-valued dataset of 15 gangs described by probability distributions over crime type, gender, and age group. This is the wide (flat table) format; see `crime2.modal` for the modal-valued version.

## Usage

```
data(crime.modal)
```

## Format

A data frame with 15 observations (gang1–gang15) and 7 numeric columns representing 3 modal variables in wide format:

- `Crime(violent)`, `Crime(non-violent)`, `Crime(none)`: Distribution over crime types (3 bins).
- `Gender(male)`, `Gender(female)`: Distribution over gender (2 bins).
- `Age(<20)`, `Age(>=20)`: Distribution over age groups (2 bins).

## Details

Two stored distributions have unresolved inconsistencies: gang10's crime proportions are 0.18, 0.15, and 0.77 (total 1.10), and gang14's gender proportions are 0.37 and 0.64 (total 1.01). These values also occur in `crime2.modal`. The correct entries have not been verified against the original source table. They are retained without automatic normalization; rescaling would not establish the intended source values. Take these inconsistencies into account before analyses requiring unit-sum probabilities.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 15 |
| **Variables (p)** | 7 |
| **Subject area** | Criminology |
| **Symbolic format** | Modal |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Billard and Diday (2006).

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

## Examples

```
data(crime.modal)
```

---

`crime2.modal` *Crime Demographics Modal-Valued Dataset*

---

## Description

Modal-valued version of the crime demographics dataset. See `crime.modal` for the wide-format version.

## Usage

```
data(crime2.modal)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 15 observations and 3 modal-valued variables:

- `Crime`: Modal distribution over crime types (violent, non-violent, none).
- `Gender`: Modal distribution over gender (male, female).
- `Age`: Modal distribution over age groups (<20, >=20).

## Details

The unresolved gang10 crime and gang14 gender inconsistencies in `crime.modal` are also present here. Stored values are retained without automatic normalization pending verification of the original table.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 15 |
| **Variables (p)** | 3 |
| **Subject area** | Criminology |
| **Symbolic format** | Modal |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Billard and Diday (2006).

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

## Examples

```
data(crime2.modal)
```

---

`crude_oil_wti.its` *WTI Crude Oil Futures Daily High/Low Interval Time Series*

---

## Description

Daily high and low prices of WTI (West Texas Intermediate) crude oil futures from January 2, 2003 to December 30, 2011 (2261 trading days). Yang, Han, Hong and Wang (2016) provide a related interval time-series application; this object was independently downloaded from Yahoo Finance.

## Usage

```
data(crude_oil_wti.its)
```

## Format

A data frame with 2261 observations and 3 variables:

- `date`: Trading date (Date class).
- `low`: Daily low price (USD per barrel).
- `high`: Daily high price (USD per barrel).

## Details

WTI crude oil is a benchmark for oil prices in the Americas. This dataset covers a period that includes the 2003 Iraq War, the 2007–2008 oil price spike (reaching nearly USD 150/barrel), the 2008 global financial crisis, and the subsequent recovery. The wide variation in price levels and volatility regimes makes this dataset ideal for evaluating interval time series models under structural breaks.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 2261 |
| **Variables (p)** | 3 (date, low, high) |
| **Subject area** | Finance / Commodities |
| **Symbolic format** | Interval time series |
| **Analytical tasks** | Forecasting, Structural break analysis |

## Source

Yahoo Finance, continuous WTI crude-oil futures ticker `CL=F`, January 2, 2003–December 30, 2011. The Yahoo `Low` and `High` fields were retained after download via **quantmod**; the original retrieval date was not recorded.

## References

Yang, W., Han, A., Hong, Y. and Wang, S. (2016). Analysis of crisis impact on crude oil prices: A new approach with interval time series modelling. *Quantitative Finance*, **16**(12), 1917–1928. doi:10.1080/14697688.2016.1211795.

## Examples

```
data(crude_oil_wti.its)
head(crude_oil_wti.its)
plot(crude_oil_wti.its$date, crude_oil_wti.its$high, type = "l",
     col = "red", ylab = "Price (USD/barrel)", xlab = "Date",
     main = "WTI Crude Oil Daily High/Low (2003-2011)")
lines(crude_oil_wti.its$date, crude_oil_wti.its$low, col = "blue")
legend("topleft", c("High", "Low"), col = c("red", "blue"), lty = 1)
```

---

# djia.its *Dow Jones Industrial Average Daily High/Low Interval Time Series*

---

## Description

Daily high and low prices of the Dow Jones Industrial Average (DJIA) from January 2, 2004 to December 30, 2005 (504 trading days), independently downloaded from Yahoo Finance for use as an interval time-series benchmark.

## Usage

```
data(djia.its)
```

## Format

A data frame with 504 observations and 3 variables:

- `date`: Trading date (Date class).
- `low`: Daily low price of the DJIA.
- `high`: Daily high price of the DJIA.

## Details

The DJIA is a price-weighted index of 30 prominent companies listed on stock exchanges in the United States. Each observation represents a trading day with the daily low and high prices forming an interval. This interval.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 504 |
| **Variables (p)** | 3 (date, low, high) |
| **Subject area** | Finance |
| **Symbolic format** | Interval time series |
| **Analytical tasks** | Forecasting, Time series analysis |

### Source

Yahoo Finance, ticker `^DJI`, January 2, 2004–December 30, 2005. The Yahoo `Low` and `High` fields were retained after download via **quantmod**; the original retrieval date was not recorded.

### Examples

```
data(djia.its)
head(djia.its)
plot(djia.its$date, djia.its$high, type = "l", col = "red",
     ylab = "Price", xlab = "Date", main = "DJIA Daily High/Low")
lines(djia.its$date, djia.its$low, col = "blue")
legend("topleft", c("High", "Low"), col = c("red", "blue"), lty = 1)
```

---

## `ecoli_routes.int` *E. coli Transport Routes Interval Dataset*

---

### Description

Interval-valued dataset of 9 E. coli transport routes with 5 interval variables representing biochemical pathway measurements.

### Usage

```
data(ecoli_routes.int)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 9 observations (transport routes) and 5 interval-valued variables:

- `Y1` through `Y5`: Interval-valued biochemical pathway measurements.

Row names are Route_1 through Route_9.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 9 |
| **Variables (p)** | 5 |
| **Subject area** | Biology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

Billard, L. and Diday, E. (2020), Table 8-10.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 8-10.

## Examples

```
data(ecoli_routes.int)
```

---

`employment.int` *European Employment by Gender and Age Interval Dataset*

---

## Description

Interval-valued proportions for 12 sex-age population groups across employment variables (employment type, education, industry sector, occupation, marital status). Used for factorial discriminant analysis.

## Usage

```
data(employment.int)
```

## Format

A data frame with 12 observations and 20 columns (9 interval variables in _l/_u Min-Max pairs, plus a group label and class):

- `group`: Sex-age group identifier (character).
- `full_time_l`, `full_time_u`: Full-time employment proportion range.
- `part_time_l`, `part_time_u`: Part-time employment proportion range.
- `primary_studies_l`, `primary_studies_u`: Primary studies proportion range.
- `secondary_studies_l`, `secondary_studies_u`: Secondary studies proportion range.
- `uni_studies_l`, `uni_studies_u`: University studies proportion range.
- `employee_l`, `employee_u`: Employee proportion range.

- manufacturing_l, manufacturing_u: Manufacturing sector proportion range.
- construction_l, construction_u: Construction sector proportion range.
- wholesale_retail_l, wholesale_retail_u: Wholesale/retail proportion range.
- class: Group classification (numeric).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 12 |
| **Variables (p)** | 20 |
| **Subject area** | Economics |
| **Symbolic format** | Interval |
| **Analytical tasks** | Discriminant analysis, Classification |

## Source

Diday and Noirhomme-Fraiture (2008), Table 18.1.

## References

Diday, E. and Noirhomme-Fraiture, M. (Eds.) (2008). *Symbolic Data Analysis and the SODAS Software*. Wiley. Table 18.1.

## Examples

```
data(employment.int)
```

---

# energy_consumption.distr

*US Energy Consumption Distribution-Valued Dataset*

---

## Description

Distribution-valued dataset of energy consumption across US states. Each energy type described by Normal distribution parameters (mean, SD).

## Usage

```
data(energy_consumption.distr)
```

## Format

A data frame with 5 observations and 3 variables:

- type: Energy type.
- mean: Mean consumption across 50 states.
- sd: Standard deviation.

## Details

Five types: Petroleum, Natural Gas, Coal, Hydroelectric, Nuclear Power. Values are rescaled consumption from the US Census Bureau (2004).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 5 |
| **Variables (p)** | 3 |
| **Subject area** | Energy |
| **Symbolic format** | Distribution |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard and Diday (2006), Table 2.8.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.8.

## Examples

```
data(energy_consumption.distr)
```

---

`energy_usage.distr` *Energy Usage Distribution-Valued Dataset*

---

## Description

Distribution-valued dataset for 10 towns (geographic areas) with categorical probability distributions for fuel type and central heating. Each observation has two distribution-valued variables.

## Usage

```
data(energy_usage.distr)
```

## Format

A data frame with 10 observations and 2 distribution-valued variables:

- `fuel_type`: Distribution over fuel types (None, Gas, Oil, Electricity, Coal).
- `central_heating`: Distribution over central heating (No, Yes).

Row names are Town_1 through Town_10.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 2 |
| **Subject area** | Energy |
| **Symbolic format** | Distribution |
| **Analytical tasks** | Descriptive statistics |

### Source

Billard, L. and Diday, E. (2006), Table 3.7.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Table 3.7.

### Examples

```
data(energy_usage.distr)
```

---

## environment.mix *EPA Environmental Data Mixed Symbolic Dataset*

---

### Description

Mixed symbolic dataset from the US EPA with 14 state-group observations and 17 variables of mixed types: interval-valued environmental measurements and modal-valued (distributional) categorical variables.

### Usage

```
data(environment.mix)
```

### Format

A symbolic data frame (`symbolic_tbl`, of class `c("symbolic_tbl", "tbl_df", "tbl", "data.frame")`) with 14 observations and 17 variables. The first four are modal-valued (`symbolic_modal`) variables and the remaining thirteen are interval-valued (`symbolic_interval`) variables. This matches the structure of `ggInterval::Environment`.

- `URBANICITY`: Modal-valued urbanicity distribution (modal).
- `INCOMELEVEL`: Modal-valued income level distribution (modal).
- `EDUCATION`: Modal-valued education distribution (modal).
- `REGIONDEVELOPME`: Modal-valued regional development distribution (modal).

- `CONTROL`: Environmental control index range (interval).
- `SATISFY`: Satisfaction index range (interval).
- `INDIVIDUAL`: Individual concern index range (interval).
- `WELFARE`: Welfare index range (interval).
- `HUMAN`: Human impact index range (interval).
- `POLITICS`: Political concern index range (interval).
- `BURDEN`: Burden index range (interval).
- `NOISE`: Noise pollution index range (interval).
- `NATURE`: Nature preservation index range (interval).
- `SEASETC`: Seas/coastal index range (interval).
- `MULTI`: Multi-indicator range (interval).
- `WATERWASTE`: Water/waste index range (interval).
- `VEHICLE`: Vehicle emissions index range (interval).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 14 |
| **Variables (p)** | 17 |
| **Subject area** | Environment |
| **Symbolic format** | Mixed (interval, modal) |
| **Analytical tasks** | Descriptive statistics, Clustering |

## Source

Extracted from `ggInterval::Environment`. The upstream package documentation does not identify a more original data provider.

## References

Jiang, B.-S. and Wu, H.-M. (2026). *ggInterval: Visualizing Interval-Valued Data Using ggplot2*. R package. doi:10.32614/CRAN.package.ggInterval.

## Examples

```
data(environment.mix)
```

`euro_usd.its` *Euro/Dollar Exchange Rate Daily High/Low Interval Time Series*

## Description

Daily high and low values of the EUR/USD exchange rate from January 1, 2004 to December 30, 2005 (520 trading days). Inspired by the dataset used by Arroyo, Espinola and Mate (2011) for exponential smoothing methods for interval time series.

## Usage

```
data(euro_usd.its)
```

## Format

A data frame with 520 observations and 3 variables:

- `date`: Trading date (Date class).
- `low`: Daily low EUR/USD exchange rate.
- `high`: Daily high EUR/USD exchange rate.

## Details

The EUR/USD exchange rate is the most traded currency pair in the world foreign exchange market. Each observation represents a trading day with the daily low and high exchange rates (USD per EUR) forming an interval. Note: the original study by Arroyo et al. (2011) used the period 2002–2003 (519 trading days); this dataset covers 2004–2005 because Yahoo Finance historical data for this ticker is only available from late 2003 onward.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 520 |
| **Variables (p)** | 3 (date, low, high) |
| **Subject area** | Finance / Foreign Exchange |
| **Symbolic format** | Interval time series |
| **Analytical tasks** | Forecasting, Time series analysis |

## Source

Yahoo Finance, ticker `EURUSD=X`, January 1, 2004–December 30, 2005. The Yahoo `Low` and `High` fields were retained after download via **quantmod**; the original retrieval date was not recorded.

## References

Arroyo, J., Espinola, R. and Mate, C. (2011). Different approaches to forecast interval time series: A comparison in finance. *Computational Economics*, **37**(2), 169–191. doi:10.1007/s10614010-92302.

## Examples

```
data(euro_usd.its)
head(euro_usd.its)
plot(euro_usd.its$date, euro_usd.its$high, type = "l", col = "red",
     ylab = "EUR/USD", xlab = "Date",
     main = "EUR/USD Daily High/Low (2004-2005)")
lines(euro_usd.its$date, euro_usd.its$low, col = "blue")
legend("topleft", c("High", "Low"), col = c("red", "blue"), lty = 1)
```

---

# exchange_rate_returns.hist

*Exchange Rate Returns Histogram Time Series*

---

## Description

Histogram-valued time series of 108 monthly observations of daily dollar/yen exchange-rate returns. Each observation is a histogram distribution of intra-month daily returns.

## Usage

```
data(exchange_rate_returns.hist)
```

## Format

A data frame with 108 observations and 1 histogram-valued variable:

- `returns`: Histogram of daily exchange rate returns within each month.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 108 |
| **Variables (p)** | 1 |
| **Subject area** | Finance |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Time series, Descriptive statistics |

## Source

`HistDAWass::RetHTS`. The upstream package identifies the series as dollar/yen change-rate returns but does not state the original data provider.

## References

Irpino, A. *HistDAWass: Histogram-Valued Data Analysis*. R package. doi:10.32614/CRAN.package.HistDAWass.

## Examples

```
data(exchange_rate_returns.hist)
```

---

`face.iGAP` *Face Dataset (iGAP Format)*

---

## Description

Interval-valued facial measurement data for 27 face images (9 individuals x 3 replications) in iGAP format (comma-separated interval strings). Contains 6 distance measurements between facial landmarks.

## Usage

```
data(face.iGAP)
```

## Format

A data frame with 27 observations and 6 character columns in iGAP format (comma-separated `"min,max"` strings):

- `AD`: Distance AD (facial landmark pair).
- `BC`: Distance BC (facial landmark pair).
- `AH`: Distance AH (facial landmark pair).
- `DH`: Distance DH (facial landmark pair).
- `EH`: Distance EH (facial landmark pair).
- `GH`: Distance GH (facial landmark pair).

Row names encode individual and replication (e.g., FRA1, FRA2, FRA3).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 27 |
| **Variables (p)** | 6 |
| **Subject area** | Biometrics |
| **Symbolic format** | Interval (iGAP) |
| **Analytical tasks** | Classification, Visualization |

## Source

Derived from `RSDA::facedata`, which is attributed to Leroy et al. (1996), and represented here in iGAP interval-string format.

## References

Leroy, B., Chouakria, A., Herlin, I., and Diday, E. (1996). Approche geometrique et classification pour la reconnaissance de visage. In *Reconnaissance des Formes et Intelligence Artificielle (RFIA)*, 548–557.

## Examples

```
data(face.iGAP)
```

---

`finance.int` *Finance Sector Interval Dataset*

---

## Description

Interval-valued data for 14 business sectors described by job-related financial variables (job cost codes, activity codes, budgets). Used for PCA demonstrations.

## Usage

```
data(finance.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 14 observations and 7 variables:

- `Sector`: Business sector name (character).
- `Job_Cost`: Job cost range (currency units, interval).
- `Job_Code`: Job code range (interval).
- `Activity_Code`: Activity code range (interval).
- `Monthly_Cost`: Monthly cost range (currency units, interval).
- `Annual_Budget`: Annual budget range (currency units, interval).
- `n`: Number of entities in the sector (numeric).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 14 |
| **Variables (p)** | 7 |
| **Subject area** | Finance |
| **Symbolic format** | Interval |
| **Analytical tasks** | PCA |

## Source

Billard and Diday (2006), Table 5.2.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 5.2.

## Examples

```
data(finance.int)
```

---

`flights_detail.hist` *Airline Flights Detailed Histogram-Valued Dataset*

---

## Description

Histogram-valued dataset of 16 airlines with 5 flight performance histograms. Each histogram has 12 bins describing the distribution of a performance metric across flights for that airline.

## Usage

```
data(flights_detail.hist)
```

## Format

A data frame with 16 observations (airlines) and 5 histogram-valued variables:

- `airtime`: Histogram of air time (minutes).
- `taxi_in`: Histogram of taxi-in time (minutes).
- `arrival_delay`: Histogram of arrival delay (minutes).
- `taxi_out`: Histogram of taxi-out time (minutes).
- `departure_delay`: Histogram of departure delay (minutes).

Row names are Airline_1 through Airline_16.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 16 |
| **Variables (p)** | 5 |
| **Subject area** | Transportation |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering |

## Source

Billard, L. and Diday, E. (2020), Table 5-1.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 5-1.

## Examples

```
data(flights_detail.hist)
```

---

`french_agriculture.hist`

*French Agriculture Histogram-Valued Dataset*

---

## Description

Histogram-valued dataset of 22 French regions with 4 economic histogram variables related to agricultural production. Each histogram describes the distribution of farm-level values within a region.

## Usage

```
data(french_agriculture.hist)
```

## Format

A data frame with 22 observations (French regions) and 4 histogram-valued variables:

- `Y_TSC`: Histogram of total standard coefficient.
- `X_Wheat`: Histogram of wheat production.
- `X_Pig`: Histogram of pig production.
- `X_Cmilk`: Histogram of cow milk production.

Row names are French region names (e.g., Ile-de-France, Picardie).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 22 |
| **Variables (p)** | 4 |
| **Subject area** | Agriculture |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Regression, Clustering |

### Source

HistDAWass::Agronomique; provenance follows the source cited in the upstream package documentation.

### References

Verde, R., Irpino, A. and Desbois, D. (2014). How to cope with modelling and privacy concerns? A regression model and a visualization tool for aggregated data. Conference of European Statistics Stakeholders, Rome.

### Examples

```
data(french_agriculture.hist)
```

---

## freshwater_fish.int *Freshwater Fish Heavy Metal Bioaccumulation Interval Dataset*

---

### Description

Interval-valued dataset of heavy metal concentrations in organs and tissues of 12 freshwater fish species, grouped into 4 feeding categories (Carnivores, Omnivores, Detritivores, Herbivores). Contains 13 interval-valued variables measuring metal concentrations in organs and organ-to-muscle ratios.

### Usage

```
data(freshwater_fish.int)
```

### Format

A data frame with 12 observations and 14 variables:

- body_length: Body length (cm).
- body_weight: Body weight (g).
- muscle: Metal concentration in muscle tissue.
- intestine: Metal concentration in intestine.
- stomach: Metal concentration in stomach.
- gills: Metal concentration in gills.
- liver: Metal concentration in liver.
- kidney: Metal concentration in kidney.
- liver_muscle_ratio: Liver-to-muscle concentration ratio.
- kidney_muscle_ratio: Kidney-to-muscle concentration ratio.
- gills_muscle_ratio: Gills-to-muscle concentration ratio.
- intestine_muscle_ratio: Intestine-to-muscle concentration ratio.
- stomach_muscle_ratio: Stomach-to-muscle concentration ratio.
- class: Feeding category (Carnivores, Omnivores, Detritivores, Herbivores).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 12 |
| **Variables (p)** | 14 |
| **Subject area** | Biology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

### Source

https://github.com/Natandradesa/Kernel-Clustering-for-Interval-Data

### References

Andrade de Sa, J. N., Ferreira, M. R. P. and de Carvalho, F. A. T. (2025). Kernel clustering with automatic variable weighting for interval data. *Neurocomputing*, 650, 130849. doi:10.1016/j.neucom.2025.130849.

### Examples

```
data(freshwater_fish.int)
```

---

## fuel_consumption.modal

*Fuel Consumption by Region Dataset*

---

### Description

Modal-valued dataset describing fuel consumption patterns across 10 regions by proportions of heating fuel types (gas, oil, electricity, other) and per-capita expenditure.

### Usage

```
data(fuel_consumption.modal)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 10 observations and 3 variables:

- `Region`: Region identifier (character).
- `Expenditure`: Per-capita fuel expenditure (numeric).
- `Fuel_Type`: Modal distribution over fuel types (gas, oil, electric, other).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 3 |
| **Subject area** | Energy |
| **Symbolic format** | Modal |
| **Analytical tasks** | Regression |

### Source

Billard and Diday (2006), Table 3.7.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 3.7.

### Examples

```
data(fuel_consumption.modal)
```

---

## fungi.int *Fungi Morphological Measurements Interval Dataset*

---

### Description

Interval-valued morphological measurements for 55 fungi specimens from 3 genera (Amanita, Agaricus, Boletus). Contains 5 interval-valued variables describing pileus and stipe dimensions and spore characteristics.

The source data contained one `stipe_width` interval whose two recorded endpoints were reversed; we reordered the endpoints from [5, 1] to [1, 5] before analysis. No observation was deleted and no value was imputed.

### Usage

```
data(fungi.int)
```

### Format

A data frame with 55 observations and 6 variables:

- `pileus_width`: Width of the pileus (cap).
- `stipe_width`: Width of the stipe (stem).
- `stipe_thickness`: Thickness of the stipe.
- `spore_height`: Height of the spores.
- `spore_width`: Width of the spores.
- `class`: Fungus genus (Amanita, Agaricus, Boletus).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 55 |
| **Variables (p)** | 6 |
| **Subject area** | Biology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

### Source

https://github.com/Natandradesa/Kernel-Clustering-for-Interval-Data

### References

Andrade de Sa, J. N., Ferreira, M. R. P. and de Carvalho, F. A. T. (2025). Kernel clustering with automatic variable weighting for interval data. *Neurocomputing*, 650, 130849. doi:10.1016/j.neucom.2025.130849.

### Examples

```
data(fungi.int)
```

---

## genome_abundances.int *Genome Dinucleotide Abundance Intervals*

---

### Description

Interval-valued dataset of dinucleotide relative abundances for 14 genome classes. Each class aggregates multiple genomes; the intervals represent the range of observed abundance values within each class for 10 dinucleotide pairs, plus a count variable.

### Usage

```
data(genome_abundances.int)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 14 observations (genome classes) and 11 variables:

- `CG`: Interval-valued CG dinucleotide relative abundance.
- `GC`: Interval-valued GC dinucleotide relative abundance.
- `TA`: Interval-valued TA dinucleotide relative abundance.
- `AT`: Interval-valued AT dinucleotide relative abundance.
- `CC`: Interval-valued CC dinucleotide relative abundance.

- AA: Interval-valued AA dinucleotide relative abundance.
- AC: Interval-valued AC dinucleotide relative abundance.
- AG: Interval-valued AG dinucleotide relative abundance.
- CA: Interval-valued CA dinucleotide relative abundance.
- GA: Interval-valued GA dinucleotide relative abundance.
- n: Number of genomes in the class (integer).

Row names are Class_1 through Class_14.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 14 |
| **Variables (p)** | 11 |
| **Subject area** | Genomics |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Billard, L. and Diday, E. (2020), Table 3-16.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 3-16.

## Examples

```
data(genome_abundances.int)
```

---

# glucose.hist *Blood Glucose Histogram-Valued Dataset*

---

## Description

Histogram-valued dataset of 4 regions with a single histogram-valued variable describing the distribution of blood glucose measurements.

## Usage

```
data(glucose.hist)
```

## Format

A data frame with 4 observations (regions) and 1 histogram-valued variable:

- `glucose`: Histogram of blood glucose levels.

Row names are Region_1 through Region_4.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 4 |
| **Variables (p)** | 1 |
| **Subject area** | Medical |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard, L. and Diday, E. (2020), Table 4-14.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 4-14.

## Examples

```
data(glucose.hist)
```

---

`hardwood.hist` *Hardwood Tree Species Histogram-Valued Dataset*

---

## Description

Histogram-valued climate data for 5 hardwood tree species in the southeastern United States. Each observation represents a species with 4 histogram-valued climate variables.

## Usage

```
data(hardwood.hist)
```

## Format

A data frame with 5 observations and 4 histogram-valued variables:

- `ANNT`: Annual temperature histogram (degrees C).
- `JULT`: July temperature histogram (degrees C).
- `ANNP`: Annual precipitation histogram (mm).
- `MITM`: Moisture index histogram.

## Details

Since version 0.2.7.2, the final upper endpoint of `ANNT` in observation 3 is 24.4 rather than 14.4, giving the final bin `[22.70, 24.40)`. The erroneous 14.4 is also present in `RSDA::hardwoodBrito` version 3.2.5. The correction follows source-data correspondence dated 18 September 2026 reporting verification against the quantiles file supplied by a source author. The underlying microdata and the spreadsheet itself were not available for independent inspection. All other endpoints and all bin proportions are unchanged.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 5 |
| **Variables (p)** | 4 |
| **Subject area** | Forestry |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Extracted from `RSDA::hardwoodBrito`; the dataset and its distributional principal-component analysis are described by Brito and Dias (2022).

## References

Brito, P. and Dias, S. (2022). *Analysis of Distributional Data*. CRC Press.

## Examples

```
data(hardwood.hist)
```

`hdi_gender.int` *Human Development Index and Gender Indicators Interval Dataset*

## Description

Interval-valued World Bank gender indicators for 183 countries, with ordinal HDI classification. Contains interval ranges for Women, Business and the Law Index Score and proportion of seats held by women in national parliaments.

## Usage

```
data(hdi_gender.int)
```

## Format

A data frame with 183 observations and 6 variables:

- `code`: ISO 3166-1 alpha-3 country code.
- `country`: Country name.
- `hdi`: Human Development Index value (UNDP).
- `women_law_index`: Women, Business and the Law Index Score range.
- `women_parliament`: Proportion of seats held by women in national parliaments range (%).
- `hdi_category`: Ordered factor with HDI classification (Low < Medium < High < Very High).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 183 |
| **Variables (p)** | 6 |
| **Subject area** | Socioeconomics |
| **Symbolic format** | Interval |
| **Analytical tasks** | Classification |

## Source

https://github.com/aleixalcacer/OCFIVD

## References

Alcacer, A., Martinez-Garcia, M. and Epifanio, I. (2024). Ordinal classification for interval-valued data and interval-valued functional data. *Expert Systems with Applications*, 238, 122277. doi:10.1016/j.eswa.2023.122277.

## Examples

```
data(hdi_gender.int)
```

`health_insurance.mix` *Health Insurance Mixed Symbolic Dataset*

## Description

Classical (microdata) health insurance dataset of 51 individual patient records with 30 variables including demographics, clinical measurements, and diagnostic indicators. This is the raw data underlying the symbolic `health_insurance2.modal` dataset.

## Usage

```
data(health_insurance.mix)
```

## Format

A data frame with 51 observations and 30 variables (Y1–Y30):

- `Y1`: City (character).
- `Y2`: Gender (M/F, character).
- `Y3`: Age (integer).
- `Y4`: Sex (M/D, character).
- `Y5`: Marital status (S/M, character).
- `Y6`: Number of dependents (integer).
- `Y7`: Parents alive indicator (integer).
- `Y8`: Number of children (integer).
- `Y9`: Height (cm, integer).
- `Y10`: Weight (pounds, integer).
- `Y11`: Systolic blood pressure (mmHg, integer).
- `Y12`: Diastolic blood pressure (mmHg, integer).
- `Y13`: Cholesterol (mg/dL, integer).
- `Y14`: Cholesterol measure 2 (integer).
- `Y15`: Additional lab measurement (integer).
- `Y16`: Ratio measurement (numeric).
- `Y17`: Lab value (integer).
- `Y18`: Lab value (integer).
- `Y19`: Lab value (integer).
- `Y20`: Lab ratio (numeric).
- `Y21`: Additional lab value (integer).
- `Y22`: Additional lab value (integer).
- `Y23`: Blood chemistry value (numeric).

- `Y24`: Blood chemistry value (numeric).
- `Y25`: Blood chemistry value (numeric).
- `Y26`: Blood chemistry value (numeric).
- `Y27`: Blood chemistry value (numeric).
- `Y28`: Diagnostic indicator (Y/N, character).
- `Y29`: Diagnostic indicator (Y/N, character).
- `Y30`: Count variable (integer).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 51 |
| **Variables (p)** | 30 |
| **Subject area** | Medical |
| **Symbolic format** | Classical (microdata) |
| **Analytical tasks** | Descriptive statistics, Aggregation |

## Source

Billard and Diday (2006), Tables 2.1–2.2.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Tables 2.1-2.2.

## Examples

```
data(health_insurance.mix)
```

---

`health_insurance2.modal`

*Health Insurance Modal-Valued Dataset*

---

## Description

Modal-valued symbolic version of the health insurance dataset, aggregated into 6 disease-type-by-gender groups. See `health_insurance.mix` for the underlying microdata.

## Usage

```
data(health_insurance2.modal)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 6 observations and 6 variables:

- `Type Gender`: Disease type and gender label (character).
- `Age`: Modal distribution over age bins.
- `Marital Status`: Modal distribution over marital status (M, S).
- `Parents Alive`: Modal distribution over number of parents alive (0, 1, 2).
- `Weight`: Modal distribution over weight bins (pounds).
- `Cholesterol`: Modal distribution over cholesterol bins (mg/dL).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 6 |
| **Variables (p)** | 6 |
| **Subject area** | Medical |
| **Symbolic format** | Modal |
| **Analytical tasks** | Clustering, Descriptive statistics |

### Source

Billard and Diday (2006), Table 2.2b.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.2b.

### Examples

```
data(health_insurance2.modal)
```

---

## `hematocrit.hist` *Hematocrit by Gender and Age Histogram-Valued Dataset*

---

### Description

Histogram-valued hematocrit distributions for 14 gender-age groups (7 female + 7 male age groups from 20s to 80+). Each observation has a 10-bin histogram of hematocrit percentages.

### Usage

```
data(hematocrit.hist)
```

## Format

A data frame with 14 observations and 3 variables:

- `gender`: Gender (Female or Male).
- `age`: Age group (20s, 30s, 40s, 50s, 60s, 70s, 80+).
- `hematocrit`: Histogram-valued hematocrit distribution (%).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 14 |
| **Variables (p)** | 3 |
| **Subject area** | Medical |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard, L. and Diday, E. (2006), Table 4.14.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Table 4.14.

## Examples

```
data(hematocrit.hist)
```

---

# hematocrit_hemoglobin.hist

*Hematocrit and Hemoglobin Bivariate Histogram-Valued Dataset*

---

## Description

Bivariate histogram-valued dataset with 10 observations, each described by a 2-bin hematocrit histogram and a 2-bin hemoglobin histogram. Used for bivariate symbolic regression demonstrations.

## Usage

```
data(hematocrit_hemoglobin.hist)
```

### Format

A data frame with 10 observations and 2 histogram-valued variables:

- `hematocrit`: Histogram-valued hematocrit distribution (%).
- `hemoglobin`: Histogram-valued hemoglobin distribution (g/dL).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 2 |
| **Subject area** | Medical |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Regression |

### Source

Billard, L. and Diday, E. (2006), Table 6.8.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Table 6.8.

### Examples

```
data(hematocrit_hemoglobin.hist)
```

---

## `hemoglobin.hist` *Hemoglobin by Gender and Age Histogram-Valued Dataset*

---

### Description

Histogram-valued hemoglobin distributions for 14 gender-age groups (7 female + 7 male age groups from 20s to 80+). Each observation has a 10-bin histogram of hemoglobin levels (g/dL).

### Usage

```
data(hemoglobin.hist)
```

### Format

A data frame with 14 observations and 3 variables:

- `gender`: Gender (Female or Male).
- `age`: Age group (20s, 30s, 40s, 50s, 60s, 70s, 80+).
- `hemoglobin`: Histogram-valued hemoglobin distribution (g/dL).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 14 |
| **Variables (p)** | 3 |
| **Subject area** | Medical |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Descriptive statistics |

### Source

Billard, L. and Diday, E. (2006), Table 4.6.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Table 4.6.

### Examples

```
data(hemoglobin.hist)
```

---

## `hierarchy` *Hierarchy Dataset*

---

### Description

Classical (microdata) dataset of 20 observations illustrating hierarchical categorical structures with a response variable Y and hierarchical predictors X1–X5. See `hierarchy.int` for the interval-valued version.

### Usage

```
data(hierarchy)
```

### Format

A data frame with 20 observations and 6 variables:

- `Y`: Response variable (numeric).
- `X1`: Hierarchy level 1 category (a/b/c, character).
- `X2`: Hierarchy level 2 category (a1/a2, character; NA for non-a).
- `X3`: Hierarchy level 3 category (a11/a12, character; NA for non-a1).
- `X4`: Numeric predictor for group b (numeric; NA for non-b).
- `X5`: Numeric predictor for group c (numeric; NA for non-c).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 20 |
| **Variables (p)** | 6 |
| **Subject area** | Methodology |
| **Symbolic format** | Classical (microdata) |
| **Analytical tasks** | Aggregation, Descriptive statistics |

### Source

Billard and Diday (2006), Table 2.15.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.15.

### Examples

```
data(hierarchy)
```

---

## hierarchy.hist *Hierarchical Symbolic Dataset with Mixed Types*

---

### Description

Mixed symbolic dataset of 10 observations with hierarchical categorical variables, conditional histogram variables, and an interval-valued variable. From Table 6.20 of Billard and Diday (2006).

### Usage

```
data(hierarchy.hist)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 10 observations and 7 variables:

- `duration_time`: Histogram-valued duration (2-bin).
- `hierarchy_1`: Categorical hierarchy level 1 (a/b/c).
- `hierarchy_2`: Categorical hierarchy level 2 (a1/a2), conditional on hierarchy_1 = a.
- `hierarchy_3`: Categorical hierarchy level 3 (a11/a12), conditional on hierarchy_2 = a1.
- `glucose`: Histogram-valued glucose (2-bin), conditional.
- `pulse_rate`: Histogram-valued pulse rate (2-bin), conditional.
- `cholesterol`: Interval-valued cholesterol level.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 7 |
| **Subject area** | Methodology |
| **Symbolic format** | Mixed (histogram, interval, categorical) |
| **Analytical tasks** | Descriptive statistics |

### Source

Billard, L. and Diday, E. (2006), Table 6.20.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Table 6.20.

### Examples

```
data(hierarchy.hist)
```

---

## `hierarchy.int` *Hierarchy Interval Dataset*

---

### Description

Interval-valued version of the hierarchy dataset. See `hierarchy` for the classical version.

### Usage

```
data(hierarchy.int)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 20 observations and 6 variables:

- `Y`: Response variable range (interval).
- `X1`: Hierarchy level 1 category (a/b/c, character).
- `X2`: Hierarchy level 2 category (a1/a2, character; NA for non-a).
- `X3`: Hierarchy level 3 category (a11/a12, character; NA for non-a1).
- `X4`: Predictor range for group b (interval; NA for non-b).
- `X5`: Predictor range for group c (interval; NA for non-c).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 20 |
| **Variables (p)** | 6 |
| **Subject area** | Methodology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Regression |

### Source

Billard and Diday (2006), Table 2.15.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.15.

### Examples

```
data(hierarchy.int)
```

---

## histogram_stats *Statistics for Histogram Data*

---

### Description

Functions to compute the mean, variance, covariance, and correlation of histogram-valued data.

### Usage

```
hist_mean(x, var_name, method = "BG", ...)

hist_var(x, var_name, method = "BG", ...)

hist_cov(x, var_name1, var_name2, method = "BG", ...)

hist_cor(x, var_name1, var_name2, method = "BG", ...)
```

### Arguments

| | |
|---|---|
| x | histogram-valued data object. |
| var_name | the variable name or the column location. |
| method | method to calculate statistics. One of "BG" (Bertrand and Goupil, 2000; default), "BD" (Billard and Diday, 2006), "B" (Billard, 2008), or "L2W" (L2 Wasserstein). All four methods are available for all four functions. |
| ... | additional parameters. |
| var_name1 | the variable name or the column location. |
| var_name2 | the variable name or the column location. |

## Details

Four functions are provided:

- `hist_mean`: Compute the mean of histogram-valued data.
- `hist_var`: Compute the variance of histogram-valued data.
- `hist_cov`: Compute the covariance between two histogram-valued variables.
- `hist_cor`: Compute the correlation between two histogram-valued variables.

Four methods are supported for all functions:

**BG** Bertrand and Goupil (2000) method. Uses histogram bin boundaries and probabilities to compute first and second moments.

**BD** Billard and Diday (2006) method. A signed decomposition using the sign of each bin's midpoint deviation from the overall mean and a quadratic form on the bin boundaries.

**B** Billard (2008) method. Uses cross-products of deviations of the bin boundaries from the overall mean.

**L2W** L2 Wasserstein method. Uses optimal-transport (Wasserstein) distances between the quantile functions of the histogram distributions.

For the mean, BG, BD, and B return the same value because they share the same first-order moment definition; only L2W uses a different (quantile-based) mean. For variance, covariance, and correlation, all four methods generally produce different results.

For `hist_cor`, the BG, BD, and B correlations all use the Bertrand-Goupil standard deviation $S(Y)$ in the denominator, following Irpino and Verde (2015, Eqs. 30–32). Only the L2W method uses its own Wasserstein-based standard deviation in the denominator.

## Value

A numeric value or vector for `hist_mean` and `hist_var`; a single numeric value for `hist_cov` and `hist_cor`.

## Author(s)

Po-Wei Chen, Han-Ming Wu

## See Also

int_mean int_var int_cov int_cor

## Examples

```
library(HistDAWass)
x <- HistDAWass::BLOOD
hist_mean(x, var_name = "Cholesterol", method = "BG")
hist_mean(x, var_name = "Cholesterol", method = "BD")
hist_var(x, var_name = "Cholesterol", method = "BG")
hist_var(x, var_name = "Cholesterol", method = "BD")
hist_cov(x, var_name1 = "Cholesterol", var_name2 = "Hemoglobin", method = "BG")
hist_cor(x, var_name1 = "Cholesterol", var_name2 = "Hemoglobin", method = "BG")
```

`hist_extract` *Extract Histogram Endpoints and Proportions*

## Description

Convert histogram strings or numeric-bin modal data to a long data frame with one row per bin. This function also works when sourced directly, using only base R, with datasets from dataSDA 0.2.7.

## Usage

```
hist_extract(x, variables = NULL, normalize = FALSE)
```

## Arguments

| | |
|---|---|
| `x` | A character vector of histogram strings, a `symbolic_modal` column, or a data frame (including a `symbolic_tbl`) containing them. |
| `variables` | Optional character vector of column names to extract from a data frame. By default, numeric histogram columns are detected; other columns are omitted. Explicitly selected unsupported columns cause an error. |
| `normalize` | Logical; divide each histogram's proportions by their sum? Defaults to `FALSE`, preserving the stored values. |

## Details

Character histograms use the form `"{[0, 10), 0.4; [10, 20], 0.6}"`. Numeric point masses such as `"{0, 0.77; 1, 0.08; 2, 0.15}"` have equal lower and upper endpoints and both endpoints closed. Signed numbers, scientific notation, and infinite endpoints are supported. Modal columns are read from their stored `var` and `prop` components; numeric bin labels such as `"Flight Time(<120)"`, `"Flight Time([120, 220])"`, and `"Flight Time(>220)"` are supported. One-sided bins use infinite endpoints. Categorical modalities have no numeric endpoints and are omitted by automatic detection.

Stored proportions are bin masses, not densities or cumulative probabilities. Rounded source proportions need not sum exactly to one. No precision lost in the stored data can be recovered. Normalization is optional and requires a positive total. Bins remain in source order; zero-mass and zero-width bins, gaps, and overlaps are preserved. This is an extractor, not a check that bins form a non-overlapping partition. Reversed endpoints are preserved with a warning; they are not silently swapped. Malformed selected cells and nonfinite or out-of-range proportions cause an informative error. The legacy semicolon between an interval and its proportion in `joggers.mix` is accepted as an alternative to a comma.

Missing cells (`NA`, blank strings, or `NULL` modal cells) produce one row with missing bin fields. Empty input produces a zero-row result. A nonempty data frame with no detected histogram columns causes an error.

## Value

A plain data frame with columns `observation` (original row index), `concept` (the input's `concept` attribute, row names, vector names, or row numbers), `variable`, `bin` (within-cell index), numeric `lower`, `upper`, and `proportion`, logical `lower_closed` and `upper_closed`, and `label` (the stored bin label). For vector input the variable name is `"histogram"`. Use `observation` to join omitted metadata back to the result.

## Examples

```
hist_extract("{[0, 10), 0.4; [10, 20], 0.6}")
data(blood.hist)
head(hist_extract(blood.hist, variables = "Cholesterol"))
data(iris_species.hist)
iris_bins <- hist_extract(iris_species.hist)
iris_bins$species <- iris_species.hist$species[iris_bins$observation]
head(iris_bins)
data(census.mix)
unique(hist_extract(census.mix)$variable)
data(airline_flights2.modal)
head(hist_extract(airline_flights2.modal, variables = "FlightTime"))
data(lung_cancer.hist)
hist_extract(lung_cancer.hist)
```

---

# horses.int *Horse Breeds Interval Dataset*

---

## Description

Interval-valued data for 8 horse breeds (CES, CMA, PEN, TES, CEN, LES, PES, PAM) described by 6 variables: minimum/maximum weight, minimum/maximum height, cost of mares, cost of fillies.

## Usage

```
data(horses.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 8 observations and 7 variables:

- `Breed`: Horse breed code (CES, CMA, PEN, TES, CEN, LES, PES, PAM; character).
- `Minimum_Weight`: Minimum weight range (kg, interval).
- `Maximum_Weight`: Maximum weight range (kg, interval).
- `Minimum_Height`: Minimum height range (cm, interval).
- `Maximum_Height`: Maximum height range (cm, interval).
- `Mares_Cost`: Cost of mares range (currency units, interval).
- `Fillies_Cost`: Cost of fillies range (currency units, interval).

### Details

Extensively used in SDA for demonstrating divisive clustering, distance computation, hierarchy/pyramid construction, and complete objects.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 8 |
| **Variables (p)** | 7 |
| **Subject area** | Zoology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

### Source

Billard and Diday (2006), Table 7.14.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 7.14.

### Examples

```
data(horses.int)
```

---

## hospital.hist *Hospital Costs Histogram-Valued Dataset*

---

### Description

Histogram-valued cost distributions for 15 hospitals. Each observation is a hospital with a 10-bin histogram of patient costs.

### Usage

```
data(hospital.hist)
```

### Format

A data frame with 15 observations and 1 histogram-valued variable:

- `cost`: Histogram-valued cost distribution (currency units).

Row names are H1 through H15.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 15 |
| **Variables (p)** | 1 |
| **Subject area** | Healthcare |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Descriptive statistics, Clustering |

### Source

Billard, L. and Diday, E. (2006), Table 3.12.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Table 3.12.

### Examples

```
data(hospital.hist)
```

---

## household_characteristics.distr

*Household Characteristics Distribution-Valued Dataset*

---

### Description

Distribution-valued dataset of 12 counties with 3 categorical probability distribution variables describing household fuel type, number of rooms, and household income brackets.

### Usage

```
data(household_characteristics.distr)
```

### Format

A data frame with 12 observations (counties) and 3 distribution-valued variables:

- `fuel_type`: Distribution over fuel types (gas, electric, oil, wood, none).
- `rooms`: Distribution over room counts ({1,2}, {3,4,5}, {>=6}).
- `household_income`: Distribution over income brackets (<10, [10,25), [25,50), [50,75), [75,100), [100,150), [150,200), >=200).

Row names are County_1 through County_12.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 12 |
| **Variables (p)** | 3 |
| **Subject area** | Socioeconomics |
| **Symbolic format** | Distribution |
| **Analytical tasks** | Clustering, Descriptive statistics |

### Source

Billard, L. and Diday, E. (2020), Table 6-1.

### References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 6-1.

### Examples

```
data(household_characteristics.distr)
```

---

## ibovespa.its *IBOVESPA Daily High/Low Interval Time Series*

---

### Description

Daily high and low values of the Brazilian IBOVESPA stock market index from January 3, 2000 to December 28, 2012 (3216 trading days). It covers the period studied by Maciel, Ballini and Gomide (2016), but this object was independently downloaded from Yahoo Finance; the article reports Bloomberg as its data source.

### Usage

```
data(ibovespa.its)
```

### Format

A data frame with 3216 observations and 3 variables:

- `date`: Trading date (Date class).
- `low`: Daily low value of the IBOVESPA index.
- `high`: Daily high value of the IBOVESPA index.

## Details

The IBOVESPA (Indice Bovespa) is the benchmark index of the Brazilian stock exchange (B3, formerly BM&FBOVESPA). It tracks the performance of the most actively traded stocks on the Sao Paulo stock exchange. The 13-year span of this dataset covers multiple market regimes including the 2008 global financial crisis, making it suitable for evaluating forecasting models under diverse conditions.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 3216 |
| **Variables (p)** | 3 (date, low, high) |
| **Subject area** | Finance |
| **Symbolic format** | Interval time series |
| **Analytical tasks** | Forecasting, Time series analysis |

## Source

Yahoo Finance, ticker `^BVSP`, January 3, 2000–December 28, 2012. The Yahoo `Low` and `High` fields were retained after download via **quantmod**; the original retrieval date was not recorded.

## References

Maciel, L., Ballini, R. and Gomide, F. (2016). Evolving granular analytics for interval time series forecasting. *Granular Computing*, **1**(4), 213–224. doi:10.1007/s4106601600163.

## Examples

```
data(ibovespa.its)
head(ibovespa.its)
plot(ibovespa.its$date, ibovespa.its$high, type = "l", col = "red",
     ylab = "Index Value", xlab = "Date",
     main = "IBOVESPA Daily High/Low (2000-2012)")
lines(ibovespa.its$date, ibovespa.its$low, col = "blue")
legend("topleft", c("High", "Low"), col = c("red", "blue"), lty = 1)
```

---

`iGAP_to_ARRAY` *iGAP to ARRAY*

---

## Description

Convert iGAP format to a 3-dimensional array `[n, p, 2]`.

## Usage

```
iGAP_to_ARRAY(data, location = NULL)
```

### Arguments

| | |
|---|---|
| data | A data.frame in iGAP format. |
| location | Integer vector specifying which columns contain comma-separated interval values. |

### Value

A numeric array of dimension [n, p, 2] with dimnames.

### Examples

```
data(abalone.iGAP)
arr <- iGAP_to_ARRAY(abalone.iGAP, 1:7)
dim(arr)
```

---

## iGAP_to_MM *iGAP to MM*

---

### Description

To convert iGAP format to MM format.

### Usage

```
iGAP_to_MM(data, location = NULL)
```

### Arguments

| | |
|---|---|
| data | The dataframe with the iGAP format. |
| location | The location of the symbolic variable in the data. |

### Value

Return a dataframe with the MM format.

### Examples

```
data(abalone.iGAP)
abalone <- iGAP_to_MM(abalone.iGAP, 1:7)
```

## iGAP_to_RSDA *iGAP to RSDA*

### Description

To convert iGAP format interval dataframe to RSDA format (symbolic_tbl).

### Usage

```
iGAP_to_RSDA(data, location = NULL)
```

### Arguments

| | |
|---|---|
| data | The dataframe with the iGAP format. |
| location | The location of the symbolic variable in the data. |

### Value

Return a symbolic_tbl dataframe with complex-encoded interval columns.

### Examples

```
data(abalone.iGAP)
rsda <- iGAP_to_RSDA(abalone.iGAP, 1:7)
```

## interval_distance *Distance Measures for Interval Data*

### Description

Functions to compute various distance measures between interval-valued observations.

`int_dist_all` computes all available distance measures at once.

### Usage

```
int_dist(x, method = "euclidean", gamma = 0.5, q = 1, p = 2, ...)

int_dist_matrix(x, method = "euclidean", gamma = 0.5, q = 1, p = 2, ...)

int_pairwise_dist(x, var_name1, var_name2, method = "euclidean", ...)

int_dist_all(x, gamma = 0.5, q = 1)
```

## Arguments

| | |
|---|---|
| x | interval-valued data with symbolic_tbl class, or an array of dimension [n, p, 2] |
| method | distance method: "GD", "IY", "L1", "L2", "CB", "HD", "EHD", "nEHD", "snEHD", "TD", "WD", "euclidean", "hausdorff", "manhattan", "city_block", "minkowski", "wasserstein", "ichino", "de_carvalho" |
| gamma | parameter for the Ichino-Yaguchi distance, 0 <= gamma <= 0.5 (default: 0.5) |
| q | parameter for the Ichino-Yaguchi distance (Minkowski exponent) (default: 1) |
| p | power parameter for Minkowski distance (default: 2) |
| ... | additional parameters |
| var_name1 | first variable name or column location |
| var_name2 | second variable name or column location |

## Details

Available distance methods:

- GD: Gowda-Diday distance (Gowda & Diday, 1991)
- IY: Ichino-Yaguchi distance (Ichino, 1988)
- L1: L1 (midpoint Manhattan) distance
- L2: L2 (Euclidean midpoint) distance
- CB: City-Block distance (Souza & de Carvalho, 2004)
- HD: Hausdorff distance (Chavent & Lechevallier, 2002)
- EHD: Euclidean Hausdorff distance
- nEHD: Normalized Euclidean Hausdorff distance
- snEHD: Span Normalized Euclidean Hausdorff distance
- TD: Tran-Duckstein distance (Tran & Duckstein, 2002)
- WD: L2-Wasserstein distance (Verde & Irpino, 2008)
- euclidean: Euclidean distance on interval centers (same as L2)
- hausdorff: Hausdorff distance (same as HD)
- manhattan: Manhattan distance (same as L1)
- city_block: City-block distance (same as CB)
- minkowski: Minkowski distance with parameter p
- wasserstein: Wasserstein distance (same as WD)
- ichino: Ichino-Yaguchi distance (simplified version)
- de_carvalho: De Carvalho distance

## Value

A distance matrix (class 'dist') or numeric vector

## Author(s)

Han-Ming Wu

## References

Gowda, K. C., & Diday, E. (1991). Symbolic clustering using a new dissimilarity measure. *Pattern Recognition*, 24(6), 567-578.

Ichino, M. (1988). General metrics for mixed features. *Systems and Computers in Japan*, 19(2), 37-50.

Chavent, M., & Lechevallier, Y. (2002). Dynamical clustering of interval data. In *Classification, Clustering and Data Analysis* (pp. 53-60). Springer.

Tran, L., & Duckstein, L. (2002). Comparison of fuzzy numbers using a fuzzy distance measure. *Fuzzy Sets and Systems*, 130, 331-341.

Verde, R., & Irpino, A. (2008). A new interval data distance based on the Wasserstein metric.

Kao, C.-H. et al. (2014). Exploratory data analysis of interval-valued symbolic data with matrix visualization. *Computational Statistics & Data Analysis*, 79, 14–29.

## See Also

int_dist_matrix int_dist_all int_pairwise_dist

## Examples

```
# Using symbolic_tbl format
data(mushroom.int)
d1 <- int_dist(mushroom.int[, 3:4], method = "euclidean")
d2 <- int_dist(mushroom.int[, 3:4], method = "hausdorff")
d3 <- int_dist(mushroom.int[, 3:4], method = "GD")

# Using array format: 4 concepts, 3 variables
x <- array(NA, dim = c(4, 3, 2))
x[,,1] <- matrix(c(1,2,3,4, 5,6,7,8, 9,10,11,12), nrow=4)
x[,,2] <- matrix(c(3,5,6,7, 8,9,10,12, 13,15,16,18), nrow=4)
d4 <- int_dist(x, method = "snEHD")
d5 <- int_dist(x, method = "IY", gamma = 0.3)
```

---

# interval_geometry *Geometric Properties of Interval Data*

---

## Description

Functions to compute geometric characteristics of interval-valued data.

## Usage

```
int_width(x, var_name, ...)

int_radius(x, var_name, ...)

int_center(x, var_name, ...)

int_overlap(x, var_name1, var_name2, ...)

int_containment(x, var_name1, var_name2, ...)

int_midrange(x, var_name, ...)
```

## Arguments

| | |
|---|---|
| x | interval-valued data with symbolic_tbl class. |
| var_name | the variable name or the column location (multiple variables are allowed). |
| ... | additional parameters |
| var_name1 | the first variable name or column location. |
| var_name2 | the second variable name or column location. |

## Details

These functions compute basic geometric properties:

- int_width: Width of each interval (upper - lower)
- int_radius: Radius of each interval (width / 2)
- int_center: Center point of each interval ((lower + upper) / 2)
- int_overlap: Overlap measure between two interval variables
- int_containment: Check if one interval contains another
- int_midrange: Half-range of each interval ((upper - lower) / 2)

## Value

A numeric matrix or value

## Author(s)

Han-Ming Wu

## See Also

int_width int_radius int_center int_overlap

## Examples

```
data(mushroom.int)

# Calculate interval widths
int_width(mushroom.int, var_name = "Pileus.Cap.Width")
int_width(mushroom.int, var_name = 2:3)

# Calculate interval radius
int_radius(mushroom.int, var_name = c("Stipe.Length", "Stipe.Thickness"))

# Get interval centers
int_center(mushroom.int, var_name = 2:4)

# Measure overlap between two variables
int_overlap(mushroom.int, "Pileus.Cap.Width", "Stipe.Length")

# Check containment
int_containment(mushroom.int, "Pileus.Cap.Width", "Stipe.Length")

# Calculate midrange
int_midrange(mushroom.int, var_name = 2:3)
```

---

# interval_position *Position and Scale Measures for Interval Data*

---

## Description

Functions to compute position and scale statistics for interval-valued data.

## Usage

```
int_median(x, var_name, method = "CM", ...)

int_quantile(x, var_name, probs = c(0.25, 0.5, 0.75), method = "CM", ...)

int_range(x, var_name, method = "CM", ...)

int_iqr(x, var_name, method = "CM", ...)

int_mad(x, var_name, method = "CM", ...)

int_mode(x, var_name, method = "CM", breaks = 30, ...)
```

## Arguments

| | |
|---|---|
| x | interval-valued data with symbolic_tbl class. |
| var_name | the variable name or the column location (multiple variables are allowed). |
| method | methods to calculate statistics: CM (default), VM, QM, SE, FV, EJD, GQ, SPT. |

| | |
|---|---|
| ... | additional parameters |
| probs | numeric vector of probabilities with values in [0,1]. |
| breaks | number of histogram breaks for mode estimation (default: 30). |

## Details

These functions provide position and scale measures:

- int_median: Median of interval data
- int_quantile: Quantiles of interval data
- int_range: Range (max - min) of interval data
- int_iqr: Interquartile range (Q3 - Q1)
- int_mad: Median absolute deviation
- int_mode: Mode of interval data (estimated via histogram)

## Value

A numeric matrix or value

## Author(s)

Han-Ming Wu

## See Also

int_mean int_var int_median int_quantile

## Examples

```
data(mushroom.int)

# Calculate median
int_median(mushroom.int, var_name = "Pileus.Cap.Width")
int_median(mushroom.int, var_name = 2:3, method = c("CM", "EJD"))

# Calculate quantiles
int_quantile(mushroom.int, var_name = 2, probs = c(0.25, 0.5, 0.75))

# Calculate interquartile range
int_iqr(mushroom.int, var_name = c("Stipe.Length", "Stipe.Thickness"))

# Calculate range
int_range(mushroom.int, var_name = "Pileus.Cap.Width")

# Calculate MAD
int_mad(mushroom.int, var_name = 2:3, method = "CM")

# Estimate mode
int_mode(mushroom.int, var_name = "Stipe.Length", method = "CM")
```

`interval_robust` *Robust Statistics for Interval Data*

## Description

Functions to compute robust statistics for interval-valued data.

## Usage

```
int_trimmed_mean(x, var_name, trim = 0.1, method = "CM", ...)

int_winsorized_mean(x, var_name, trim = 0.1, method = "CM", ...)

int_trimmed_var(x, var_name, trim = 0.1, method = "CM", ...)

int_winsorized_var(x, var_name, trim = 0.1, method = "CM", ...)
```

## Arguments

| | |
|---|---|
| `x` | interval-valued data with symbolic_tbl class. |
| `var_name` | the variable name or the column location (multiple variables are allowed). |
| `trim` | the fraction (0 to 0.5) of observations to be trimmed from each end. |
| `method` | methods to calculate statistics: CM (default), VM, QM, SE, FV, EJD, GQ, SPT. |
| `...` | additional parameters |

## Details

These functions provide robust alternatives to standard statistics:

- `int_trimmed_mean`: Mean after trimming extreme values
- `int_winsorized_mean`: Mean after winsorizing extreme values
- `int_trimmed_var`: Variance after trimming extreme values
- `int_winsorized_var`: Variance after winsorizing extreme values

Trimming vs Winsorizing:

- Trimming: Remove extreme values
- Winsorizing: Replace extreme values with less extreme values

## Value

A numeric matrix

## Author(s)

Han-Ming Wu

## See Also

int_mean int_var int_trimmed_mean

## Examples

```
data(mushroom.int)

# Trimmed mean (10% from each end)
int_trimmed_mean(mushroom.int, var_name = "Pileus.Cap.Width", trim = 0.1)

# Winsorized mean
int_winsorized_mean(mushroom.int, var_name = 2:3, trim = 0.05, method = "CM")

# Trimmed variance
int_trimmed_var(mushroom.int, var_name = c("Stipe.Length"), trim = 0.1)
```

---

# interval_shape *Distribution Shape Measures for Interval Data*

---

## Description

Functions to compute shape statistics (skewness, kurtosis) for interval-valued data.

## Usage

```
int_skewness(x, var_name, method = "CM", ...)

int_kurtosis(x, var_name, method = "CM", ...)

int_symmetry(x, var_name, method = "CM", ...)

int_tailedness(x, var_name, method = "CM", ...)
```

## Arguments

| | |
|---|---|
| x | interval-valued data with symbolic_tbl class. |
| var_name | the variable name or the column location (multiple variables are allowed). |
| method | methods to calculate statistics: CM (default), VM, QM, SE, FV, EJD, GQ, SPT. |
| ... | additional parameters |

## Details

These functions measure distribution shape:

- int_skewness: Measure of asymmetry (skewness)
- int_kurtosis: Measure of tail heaviness (kurtosis)
- int_symmetry: Symmetry coefficient

- `int_tailedness`: Tailedness measure (alias for excess kurtosis)

Skewness interpretation:

- = 0: Symmetric distribution
- > 0: Right-skewed (positive skew)
- < 0: Left-skewed (negative skew)

Kurtosis interpretation (excess kurtosis):

- = 0: Normal distribution (mesokurtic)
- > 0: Heavy tails (leptokurtic)
- < 0: Light tails (platykurtic)

## Value

A numeric matrix

## Author(s)

Han-Ming Wu

## See Also

int_mean int_var int_skewness int_kurtosis

## Examples

```
data(mushroom.int)

# Calculate skewness
int_skewness(mushroom.int, var_name = "Pileus.Cap.Width")
int_skewness(mushroom.int, var_name = 2:3, method = c("CM", "EJD"))

# Calculate kurtosis
int_kurtosis(mushroom.int, var_name = c("Stipe.Length", "Stipe.Thickness"))

# Check symmetry
int_symmetry(mushroom.int, var_name = 2:4, method = "CM")

# Check tailedness
int_tailedness(mushroom.int, var_name = "Pileus.Cap.Width", method = "CM")
```

`interval_similarity` *Similarity Measures for Interval Data*

## Description

Functions to compute similarity measures between interval-valued observations.

## Usage

```
int_jaccard(x, var_name1, var_name2, ...)

int_dice(x, var_name1, var_name2, ...)

int_cosine(x, var_name1, var_name2, ...)

int_overlap_coefficient(x, var_name1, var_name2, ...)

int_tanimoto(x, var_name1, var_name2, ...)

int_similarity_matrix(x, method = "jaccard", ...)
```

## Arguments

| | |
|---|---|
| `x` | interval-valued data with symbolic_tbl class. |
| `var_name1` | the first variable name or column location. |
| `var_name2` | the second variable name or column location. |
| `...` | additional parameters |
| `method` | similarity method for int_similarity_matrix: "jaccard", "dice", or "overlap". |

## Details

These functions compute various similarity measures:

- `int_jaccard`: Jaccard similarity coefficient
- `int_dice`: Dice similarity coefficient
- `int_cosine`: Cosine similarity
- `int_overlap_coefficient`: Overlap coefficient
- `int_tanimoto`: Tanimoto coefficient (generalized Jaccard)
- `int_similarity_matrix`: Pairwise similarity matrix across all observations

All similarity measures range from 0 (no similarity) to 1 (perfect similarity).

## Value

A numeric matrix or value

### Author(s)

Han-Ming Wu

### See Also

int_dist int_cor int_jaccard

### Examples

```
data(mushroom.int)

# Jaccard similarity
int_jaccard(mushroom.int, "Pileus.Cap.Width", "Stipe.Length")

# Dice coefficient
int_dice(mushroom.int, 2, 3)

# Cosine similarity
int_cosine(mushroom.int,
           var_name1 = c("Pileus.Cap.Width"),
           var_name2 = c("Stipe.Length", "Stipe.Thickness"))

# Overlap coefficient
int_overlap_coefficient(mushroom.int, 2, 3:4)

# Tanimoto coefficient
int_tanimoto(mushroom.int, "Pileus.Cap.Width", "Stipe.Length")

# Similarity matrix across all observations
int_similarity_matrix(mushroom.int, method = "jaccard")
```

---

## interval_stats *Statistics for Interval Data*

---

### Description

Functions to compute the mean, variance, covariance, and correlation of interval-valued data.

### Usage

```
int_mean(x, var_name, method = "CM", ...)

int_var(x, var_name, method = "CM", ...)

int_cov(x, var_name1, var_name2, method = "CM", ...)

int_cor(x, var_name1, var_name2, method = "CM", ...)
```

## Arguments

| | |
|---|---|
| x | interval-valued data with symbolic_tbl class. |
| var_name | the variable name or the column location (multiple variables are allowed). |
| method | methods to calculate statistics: CM (default), VM, QM, SE, FV, EJD, GQ, SPT. |
| ... | additional parameters |
| var_name1 | the variable name or the column location (multiple variables are allowed). |
| var_name2 | the variable name or the column location (multiple variables are allowed). |

## Details

Available methods (applicable to all four functions):

- CM: Center Method — uses midpoints (a + b) / 2
- VM: Vertices Method — uses all 2^p vertex combinations
- QM: Quantiles Method — uses equally spaced quantile points
- SE: Set Expansion — uses endpoints only (quantiles with m = 1)
- FV: Fitted Values — uses linear regression fitted values
- EJD: Empirical Joint Distribution
- GQ: Symbolic Covariance method (Billard and Diday, 2006)
- SPT: Total Sum of Products (Billard, 2008)

## Value

A numeric matrix for int_mean and int_var (methods x variables); a named list of covariance/correlation matrices for int_cov and int_cor (one matrix per method).

## Author(s)

Han-Ming Wu

## See Also

int_mean int_var int_cov int_cor

## Examples

```
data(mushroom.int)
int_mean(mushroom.int, var_name = "Pileus.Cap.Width")
int_mean(mushroom.int, var_name = 2:3)

var_name <- c("Stipe.Length", "Stipe.Thickness")
method <- c("CM", "FV", "EJD")
int_mean(mushroom.int, var_name, method)
int_var(mushroom.int, var_name, method)

var_name1 <- "Pileus.Cap.Width"
var_name2 <- c("Stipe.Length", "Stipe.Thickness")
```

```
method <- c("CM", "VM", "EJD", "GQ", "SPT")
int_cov(mushroom.int, var_name1, var_name2, method)
int_cor(mushroom.int, var_name1, var_name2, method)
```

---

`interval_uncertainty` *Uncertainty and Variability Measures for Interval Data*

---

## Description

Functions to compute uncertainty and variability measures for interval-valued data.

## Usage

```
int_entropy(x, var_name, method = "CM", base = 2, ...)

int_cv(x, var_name, method = "CM", ...)

int_dispersion(x, var_name, method = "CM", ...)

int_imprecision(x, var_name, ...)

int_granularity(x, var_name, ...)

int_uniformity(x, var_name, ...)

int_information_content(x, var_name, method = "CM", ...)
```

## Arguments

| | |
|---|---|
| `x` | interval-valued data with symbolic_tbl class. |
| `var_name` | the variable name or the column location (multiple variables are allowed). |
| `method` | methods to calculate statistics: CM (default), VM, QM, SE, FV, EJD, GQ, SPT. |
| `base` | logarithm base for entropy calculation (default: 2) |
| `...` | additional parameters |

## Details

These functions measure uncertainty and variability:

- `int_entropy`: Shannon entropy (information content)
- `int_cv`: Coefficient of variation (CV = SD / Mean)
- `int_dispersion`: General dispersion index
- `int_imprecision`: Imprecision based on interval width
- `int_granularity`: Variability in interval sizes
- `int_uniformity`: Uniformity of interval widths (inverse of granularity)
- `int_information_content`: Normalized entropy (entropy / log2(n))

## Value

A numeric matrix or value

## Author(s)

Han-Ming Wu

## See Also

int_var int_entropy int_cv

## Examples

```
data(mushroom.int)

# Calculate entropy
int_entropy(mushroom.int, var_name = "Pileus.Cap.Width")

# Coefficient of variation
int_cv(mushroom.int, var_name = c("Stipe.Length", "Stipe.Thickness"), method = c("CM", "EJD"))

# Measure imprecision
int_imprecision(mushroom.int, var_name = c("Stipe.Length", "Stipe.Thickness"))

# Dispersion index
int_dispersion(mushroom.int, var_name = "Pileus.Cap.Width", method = "CM")

# Check data granularity
int_granularity(mushroom.int, var_name = 2:4)

# Check uniformity
int_uniformity(mushroom.int, var_name = 2:3)

# Information content
int_information_content(mushroom.int, var_name = "Stipe.Length", method = "CM")
```

---

# int_convert_format *Convert Interval Data Format*

---

## Description

Automatically detect the format of interval data and convert it to the target format.

## Usage

```
int_convert_format(x, to = "MM", from = NULL, ...)
```

## Arguments

| | |
|---|---|
| x | interval data in one of the supported formats |
| to | target format: "MM", "iGAP", "RSDA", "ARRAY", "SODAS" (default: "MM") |
| from | source format (optional): "MM", "iGAP", "RSDA", "ARRAY", "SODAS". If NULL, will auto-detect. |
| ... | additional parameters passed to specific conversion functions |

## Details

This function provides a unified interface for all interval format conversions. It automatically detects the source format (unless specified) and applies the appropriate conversion function.

Supported conversions:

- RSDA ??? MM, iGAP, ARRAY
- MM ??? iGAP, RSDA, ARRAY
- iGAP ??? MM, RSDA, ARRAY
- ARRAY ??? RSDA, MM, iGAP
- SODAS ??? MM, iGAP, ARRAY

## Value

Interval data in the target format

## Author(s)

Han-Ming Wu

## See Also

int_detect_format int_list_conversions RSDA_to_MM RSDA_to_ARRAY MM_to_RSDA MM_to_ARRAY ARRAY_to_RSDA ARRAY_to_MM ARRAY_to_iGAP iGAP_to_MM iGAP_to_RSDA iGAP_to_ARRAY MM_to_iGAP

## Examples

```
# Auto-detect and convert to MM
data(mushroom.int)
data_mm <- int_convert_format(mushroom.int, to = "MM")

# Explicitly specify source format
data(abalone.iGAP)
data_mm <- int_convert_format(abalone.iGAP, from = "iGAP", to = "MM")

# Convert MM to iGAP
data_igap <- int_convert_format(data_mm, to = "iGAP")

 # Convert multiple datasets to MM
datasets <- list(mushroom.int, abalone.int, car.int)
```

```
mm_datasets <- lapply(datasets, int_convert_format, to = "MM")

# Check what conversions are available
int_list_conversions()
```

---

`int_detect_format` *Detect Interval Data Format*

---

## Description

Automatically detect the format of interval data.

## Usage

```
int_detect_format(x)
```

## Arguments

| | |
|---|---|
| `x` | interval data in unknown format |

## Details

Detection rules:

- `RSDA`: has class "symbolic_tbl" and contains complex columns
- `MM`: data.frame with paired "_min" and "_max" columns
- `iGAP`: data.frame with columns containing comma-separated values (e.g., "1.2,3.4")
- `ARRAY`: a 3-dimensional array with `dim[3] = 2` (min/max slices)
- `SODAS`: character string ending with ".xml" (file path)
- `SDS`: alias for SODAS

## Value

A character string indicating the detected format: "RSDA", "MM", "iGAP", "ARRAY", "SODAS", or "unknown"

## Examples

```
data(mushroom.int)
int_detect_format(mushroom.int)  # Should return "RSDA"

data(abalone.iGAP)
int_detect_format(abalone.iGAP)  # Should return "iGAP"

# ARRAY format
x <- array(1:24, dim = c(4, 3, 2))
int_detect_format(x)  # Should return "ARRAY"
```

## int_list_conversions *List Available Format Conversions*

### Description

List all available format conversion functions.

### Usage

```
int_list_conversions(from = NULL, to = NULL)
```

### Arguments

| | |
|---|---|
| from | source format (optional): "RSDA", "MM", "iGAP", "ARRAY", "SODAS" |
| to | target format (optional): "RSDA", "MM", "iGAP", "ARRAY", "SODAS" |

### Value

A data.frame showing available conversions

### Examples

```
# List all conversions
int_list_conversions()

# List conversions from RSDA
int_list_conversions(from = "RSDA")

# List conversions to MM
int_list_conversions(to = "MM")
```

## iris.int *Iris Species Interval Dataset*

### Description

Interval-valued version of the classic iris dataset, aggregated from Fisher's iris data into 30 interval observations across 3 species (Setosa, Versicolor, Virginica). Each observation represents a group of flowers with ranges for sepal and petal measurements.

### Usage

```
data(iris.int)
```

## Format

A data frame with 30 observations and 5 variables:

- `sepal_length`: Sepal length range (cm).
- `sepal_width`: Sepal width range (cm).
- `petal_length`: Petal length range (cm).
- `petal_width`: Petal width range (cm).
- `class`: Species (Setosa, Versicolor, Virginica).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 30 |
| **Variables (p)** | 5 |
| **Subject area** | Botany |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

https://github.com/Natandradesa/Kernel-Clustering-for-Interval-Data

## References

Andrade de Sa, J. N., Ferreira, M. R. P. and de Carvalho, F. A. T. (2025). Kernel clustering with automatic variable weighting for interval data. *Neurocomputing*, 650, 130849. doi:10.1016/j.neucom.2025.130849.

## Examples

```
data(iris.int)
```

---

# irish_wind.its *Irish Wind Speed Monthly Interval Time Series*

---

## Description

Monthly interval-valued wind speed data at 5 meteorological stations in Ireland from January 1961 to December 1978 (216 months). For each month and station, the interval is defined as [minimum daily average wind speed, maximum daily average wind speed] across all days in that month.

## Usage

```
data(irish_wind.its)
```

## Format

A data frame with 216 observations and 11 columns (5 interval variables in _l/_u Min-Max pairs, plus a date):

- `date`: First day of the month (Date class).
- `BIR_l`, `BIR_u`: Monthly [min, max] daily wind speed at Birr (knots).
- `DUB_l`, `DUB_u`: Monthly [min, max] daily wind speed at Dublin Airport (knots).
- `KIL_l`, `KIL_u`: Monthly [min, max] daily wind speed at Kilkenny (knots).
- `SHA_l`, `SHA_u`: Monthly [min, max] daily wind speed at Shannon Airport (knots).
- `VAL_l`, `VAL_u`: Monthly [min, max] daily wind speed at Valentia Observatory (knots).

## Details

The original data contains daily average wind speeds (in knots) at 12 synoptic meteorological stations in the Republic of Ireland, collected by the Irish Meteorological Service. This is the classic Haslett and Raftery (1989) dataset, one of the most widely used benchmarks in spatial statistics. Following the approach of Teles and Brito (2015), the raw daily data is aggregated to monthly intervals for 5 selected stations: Birr (BIR), Dublin Airport (DUB), Kilkenny (KIL), Shannon Airport (SHA), and Valentia Observatory (VAL). Each monthly interval captures the range of daily wind variability within that month.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 216 |
| **Variables (p)** | 11 |
| **Subject area** | Meteorology |
| **Symbolic format** | Interval time series (multivariate) |
| **Analytical tasks** | Space-time modelling, Forecasting, Clustering |

## Source

Derived from the `wind` dataset in the **gstat** R package (originally from Haslett and Raftery, 1989). Daily data aggregated to monthly intervals.

## References

Haslett, J. and Raftery, A. E. (1989). Space-time modelling with long-memory dependence: Assessing Ireland's wind power resource. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, **38**(1), 1–50.

Teles, P. and Brito, P. (2015). Modeling interval time series with space-time processes. *Communications in Statistics – Theory and Methods*, **44**(17), 3599–3627. doi:10.1080/03610926.2013.782200.

## Examples

```
data(irish_wind.its)
head(irish_wind.its)
# Plot Valentia Observatory wind speed interval
plot(irish_wind.its$date, irish_wind.its$VAL_u, type = "l", col = "red",
     ylab = "Wind speed (knots)", xlab = "Date",
     main = "Valentia Observatory Monthly Wind Speed Interval")
lines(irish_wind.its$date, irish_wind.its$VAL_l, col = "blue")
legend("topright", c("Max", "Min"), col = c("red", "blue"), lty = 1)
```

---

# iris_species.hist *Iris Species Histogram-Valued Dataset*

---

## Description

Histogram-valued dataset of 3 iris species (Versicolor, Virginica, Setosa) with 4 histogram-valued morphological variables and a species label. Each histogram describes the distribution of measurements within a species.

## Usage

```
data(iris_species.hist)
```

## Format

A data frame with 3 observations and 5 variables:

- `species`: Species name (factor: Versicolor, Virginica, Setosa).
- `sepal_width`: Histogram-valued sepal width distribution.
- `sepal_length`: Histogram-valued sepal length distribution.
- `petal_width`: Histogram-valued petal width distribution.
- `petal_length`: Histogram-valued petal length distribution.

Row names are species names.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 3 |
| **Variables (p)** | 5 |
| **Subject area** | Botany |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Billard, L. and Diday, E. (2020), Table 4-10.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 4-10.

## Examples

```
data(iris_species.hist)
```

---

`joggers.mix` *Joggers Mixed Symbolic Dataset*

---

## Description

Mixed symbolic dataset of 10 jogger groups with one interval-valued variable (pulse rate) and one histogram-valued variable (running time distribution).

## Usage

```
data(joggers.mix)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 10 observations (jogger groups) and 2 variables:

- `pulse_rate`: Interval-valued resting pulse rate range (bpm).
- `running_time`: Histogram-valued distribution of running times (minutes).

Row names are Group_1 through Group_10.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 2 |
| **Subject area** | Sports |
| **Symbolic format** | Mixed (interval, histogram) |
| **Analytical tasks** | Clustering |

## Source

Billard, L. and Diday, E. (2020), Table 2-5.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 2-5.

## Examples

```
data(joggers.mix)
```

---

`judge1.int` *Judge 1 Interval-Valued Ratings*

---

## Description

Simulated interval-valued agricultural-product grades assigned by Judge 1 for 6 regions on 4 variables, distributed with the GPCSIV package.

## Usage

```
data(judge1.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 6 observations and 4 interval-valued variables (V1–V4).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 6 |
| **Variables (p)** | 4 |
| **Subject area** | Methodology |
| **Symbolic format** | Interval |
| **Analytical tasks** | PCA |

## Source

Simulated data distributed as `GPCSIV::Judge1`.

## References

Makosso-Kallyth, S. (2013). Analysis of m sets of symbolic interval variables. *Revue des Nouvelles Technologies de l'Information*, RNTI-E-25, 97–108.

## Examples

```
data(judge1.int)
```

`judge2.int` *Judge 2 Interval-Valued Ratings*

## Description

Simulated interval-valued agricultural-product grades assigned by Judge 2 for 6 regions on 4 variables, distributed with the GPCSIV package.

## Usage

```
data(judge2.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 6 observations and 4 interval-valued variables (V1–V4).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 6 |
| **Variables (p)** | 4 |
| **Subject area** | Methodology |
| **Symbolic format** | Interval |
| **Analytical tasks** | PCA |

## Source

Simulated data distributed as `GPCSIV::Judge2`.

## References

Makosso-Kallyth, S. (2013). Analysis of m sets of symbolic interval variables. *Revue des Nouvelles Technologies de l'Information*, RNTI-E-25, 97–108.

## Examples

```
data(judge2.int)
```

`judge3.int` *Judge 3 Interval-Valued Ratings*

## Description

Simulated interval-valued agricultural-product grades assigned by Judge 3 for 6 regions on 4 variables, distributed with the GPCSIV package.

## Usage

```
data(judge3.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 6 observations and 4 interval-valued variables (V1–V4).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 6 |
| **Variables (p)** | 4 |
| **Subject area** | Methodology |
| **Symbolic format** | Interval |
| **Analytical tasks** | PCA |

## Source

Simulated data distributed as `GPCSIV::Judge3`.

## References

Makosso-Kallyth, S. (2013). Analysis of m sets of symbolic interval variables. *Revue des Nouvelles Technologies de l'Information*, RNTI-E-25, 97–108.

## Examples

```
data(judge3.int)
```

`lackinfo.int` *Lack of Information Questionnaire Interval Dataset*

## Description

Interval-valued dataset from a lack-of-information questionnaire. Contains biographical data and responses to 5 items measuring perception of lack of information, collected via an interval-valued Likert scale.

## Usage

```
data(lackinfo.int)
```

## Format

A data frame with 50 observations and 8 variables:

- `id`: Identification number.
- `sex`: Sex of the respondent (`male` or `female`).
- `age`: Respondent's age (in years).
- `item1`: Interval-valued answer to item 1.
- `item2`: Interval-valued answer to item 2.
- `item3`: Interval-valued answer to item 3.
- `item4`: Interval-valued answer to item 4.
- `item5`: Interval-valued answer to item 5.

## Details

An educational innovation project was carried out for improving teaching-learning processes at the University of Oviedo (Spain) for the 2020/2021 academic year. A total of 50 students answered an online questionnaire about biographical data (sex and age) and their perception of lack of information by selecting the interval that best represents their level of agreement on a scale bounded between 1 (strongly disagree) and 7 (strongly agree).

The 5 items measuring perception of lack of information are:

- I1: I receive too little information from my classmates.
- I2: It is difficult to receive relevant information from my classmates.
- I3: It is difficult to receive relevant information from the teacher.
- I4: The amount of information I receive from my classmates is very low.
- I5: The amount of information I receive from the teacher is very low.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 50 |
| **Variables (p)** | 8 |
| **Subject area** | Education |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Regression |

## Source

`IntervalQuestionStat::lackinfo`; responses were collected from 50 University of Oviedo students during the 2020/2021 academic year, as documented by the upstream package.

## References

Garcia-Garcia, J. (2022). *IntervalQuestionStat: Tools to Deal with Interval-Valued Responses in Questionnaires*. R package. doi:10.32614/CRAN.package.IntervalQuestionStat.

## Examples

```
data(lackinfo.int)
```

---

`lisbon_air_quality.int`

*Lisbon Air Quality Daily Interval Dataset*

---

## Description

Interval-valued daily air quality data from the Entrecampos monitoring station in Lisbon, Portugal, covering 2019–2021 (1096 days). Each day's pollutant concentration is represented as a $[\min, \max]$ interval from hourly measurements. Missing days are imputed via linear interpolation.

## Usage

```
data(lisbon_air_quality.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 1096 observations (daily) and 8 interval-valued pollutant variables:

- `so2`: Sulphur dioxide (ug/m3).
- `pm10`: Particulate matter < 10 um (ug/m3).
- `o3`: Ozone (ug/m3).
- `no2`: Nitrogen dioxide (ug/m3).
- `co`: Carbon monoxide (ug/m3).

- pm25: Particulate matter < 2.5 um (ug/m3).
- nox: Nitrogen oxides (ug/m3).
- no: Nitric oxide (ug/m3).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 1096 |
| **Variables (p)** | 8 |
| **Subject area** | Environment |
| **Symbolic format** | Interval |
| **Analytical tasks** | Regression, Time series |

## Source

Hourly measurements from the QualAr Portuguese air-quality monitoring network, operated by the Portuguese Environment Agency, at the Entrecampos station in Lisbon, Portugal. Daily intervals were constructed from hourly measurements as described by Loureiro et al. (2025).

## References

Loureiro, C. P., Oliveira, M. R., Brito, P. and Oliveira, L. (2025). Air Quality Data Analysis with Symbolic Principal Components. In *New Frontiers in Statistics and Data Science – SPE2023*, Springer Proceedings in Mathematics and Statistics, 469, 335–348. doi:10.1007/978303168949-9_25.

## Examples

```
data(lisbon_air_quality.int)
```

---

# loans_by_purpose.int *Loans by Purpose Interval Dataset*

---

## Description

Interval-valued data for loan characteristics aggregated by their purpose. Original microdata contains 887,383 loan records from Kaggle.

## Usage

```
data(loans_by_purpose.int)
```

## Format

A data frame with 14 observations and 4 interval-valued variables:

- `ln_inc`: Natural logarithm of self-reported annual income.
- `ln_revolbal`: Natural logarithm of total credit revolving balance.
- `open_acc`: Number of open credit lines.
- `total_acc`: Total number of credit lines.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 14 |
| **Variables (p)** | 4 |
| **Subject area** | Finance |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Clustering |

## Source

Derived from `MAINT.Data::LoansbyPurpose_minmaxDt`, which aggregates the Lending Club Loan Data by loan purpose using minima and maxima. The underlying Lending Club Loan Data were distributed through the Kaggle Data Science platform; the original upstream URL is no longer active.

## References

Duarte Silva, A. P., Brito, P., Filzmoser, P. and Dias, J. G. (2021). MAINT.Data: Modelling and Analysing Interval Data in R. *The R Journal*, 13(2), 336–364. doi:10.32614/RJ2021074.

The underlying loan records were distributed as the Lending Club Loan Data on Kaggle.

## Examples

```
data(loans_by_purpose.int)
```

---

# loans_by_risk.int *Lending Club Loans by Risk Level*

---

## Description

Interval-valued dataset of 35 Lending Club loan groups classified by risk level (A through G, 5 groups each). Each group is described by 4 interval-valued financial variables.

## Usage

```
data(loans_by_risk.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 35 observations and 5 variables:

- `log_income`: Interval-valued log annual income.
- `interest_rate`: Interval-valued interest rate (%).
- `open_accounts`: Interval-valued number of open credit accounts.
- `total_accounts`: Interval-valued total number of credit accounts.
- `risk_level`: Risk grade factor (A, B, C, D, E, F, G).

Row names are A1–A5, B1–B5, ..., G1–G5.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 35 |
| **Variables (p)** | 5 |
| **Subject area** | Finance |
| **Symbolic format** | Interval |
| **Analytical tasks** | Classification, Clustering |

## Source

Derived from `MAINT.Data::LoansbyRiskLvs_minmaxDt`, based on the Lending Club Loan Data distributed through the Kaggle Data Science platform; the original upstream URL is no longer active.

## References

Duarte Silva, A. P., Brito, P., Filzmoser, P. and Dias, J. G. (2021). MAINT.Data: Modelling and Analysing Interval Data in R. *The R Journal*, 13(2), 336–364. doi:10.32614/RJ2021074.

## Examples

```
data(loans_by_risk.int)
```

---

`loans_by_risk_quantile.int`

*Lending Club Loans by Risk Level (Quantile-Based Intervals)*

---

## Description

Interval-valued dataset of 35 Lending Club loan groups stratified by risk level (A1–G5). Intervals represent the 10th to 90th percentile range of each financial variable within each risk subgrade.

### Usage

```
data(loans_by_risk_quantile.int)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 35 observations and 4 variables:

- `ln-inc`: Interval-valued log income.
- `int-rate`: Interval-valued interest rate.
- `open-acc`: Interval-valued number of open accounts.
- `total-acc`: Interval-valued total accounts.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 35 |
| **Variables (p)** | 4 |
| **Subject area** | Finance |
| **Symbolic format** | Interval |
| **Analytical tasks** | Classification, Clustering |

### Source

Derived from `MAINT.Data::LoansbyRiskLvs_qntlDt`, based on the Lending Club Loan Data distributed through the Kaggle Data Science platform; the original upstream URL is no longer active.

### References

Duarte Silva, A. P., Brito, P., Filzmoser, P. and Dias, J. G. (2021). MAINT.Data: Modelling and Analysing Interval Data in R. *The R Journal*, 13(2), 336–364. doi:10.32614/RJ2021074.

### Examples

```
data(loans_by_risk_quantile.int)
```

---

## `lung_cancer.hist` *Lung Cancer Treatments by State Histogram-Valued Dataset*

---

### Description

Histogram-valued distribution of lung cancer treatment counts for 2 US states (Massachusetts and New York).

### Usage

```
data(lung_cancer.hist)
```

## Format

A data frame with 2 observations and 2 variables:

- `state`: State name (character).
- `y30`: Histogram-valued distribution of treatment counts as a weighted set string (e.g., "{0, 0.77; 1, 0.08; 2, 0.15}").

## Metadata

| | |
|---|---|
| **Sample size (n)** | 2 |
| **Variables (p)** | 2 |
| **Subject area** | Medical |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard and Diday (2006), Table 2.20.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.20.

## Examples

```
data(lung_cancer.hist)
```

---

`lynne1.int` *Lynne1 Blood Pressure Interval Dataset*

---

## Description

Interval-valued dataset of 10 observations with pulse rate, systolic pressure, and diastolic pressure intervals.

## Usage

```
data(lynne1.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 10 observations and 4 variables:

- `concept`: Character concept label.
- `Pulse Rate`: Interval-valued pulse rate (beats/min).
- `Systolic Pressure`: Interval-valued systolic pressure (mmHg).
- `Diastolic Pressure`: Interval-valued diastolic pressure (mmHg).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 4 |
| **Subject area** | Medical |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Regression |

### Source

`RSDA::lynne1`.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester.

Original data from the RSDA R package (`lynne1` dataset).

### Examples

```
data(lynne1.int)
```

---

## `merval.its` *MERVAL Index Weekly Min/Max Interval Time Series*

---

### Description

Weekly minimum and maximum values of the Argentine MERVAL stock market index from January 4, 2016 to September 28, 2020 (248 weeks). Daily data was downloaded and aggregated to weekly intervals. This dataset matches the period used by de Carvalho and Martos (2022).

### Usage

```
data(merval.its)
```

### Format

A data frame with 248 observations and 3 variables:

- `date`: Week start date, Monday (Date class).
- `low`: Weekly minimum of daily low values.
- `high`: Weekly maximum of daily high values.

## Details

The MERVAL (Mercado de Valores de Buenos Aires) is the main stock market index of the Buenos Aires Stock Exchange. Each observation represents one week, with the weekly low computed as the minimum of daily lows and the weekly high computed as the maximum of daily highs. The date column indicates the Monday (start) of each week. This period covers the Argentine economic crisis and the early COVID-19 pandemic impact.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 248 |
| **Variables (p)** | 3 (date, low, high) |
| **Subject area** | Finance |
| **Symbolic format** | Interval time series (weekly aggregation) |
| **Analytical tasks** | Forecasting, Time series analysis |

## Source

Yahoo Finance, ticker `^MERV`, January 1, 2016–September 30, 2020. Downloaded via **quantmod** and aggregated by week using the minimum daily low and maximum daily high; the original retrieval date was not recorded.

## References

de Carvalho, F. A. T. and Martos, G. (2022). Modeling interval trendlines: Symbolic singular spectrum analysis for interval time series. *Journal of Forecasting*, **41**(1), 167–180. doi:10.1002/for.2801.

## Examples

```
data(merval.its)
head(merval.its)
plot(merval.its$date, merval.its$high, type = "l", col = "red",
     ylab = "Index Value", xlab = "Date",
     main = "MERVAL Weekly Min/Max (2016-2020)")
lines(merval.its$date, merval.its$low, col = "blue")
legend("topleft", c("High", "Low"), col = c("red", "blue"), lty = 1)
```

---

`MM_to_ARRAY` *MM to ARRAY*

---

## Description

Convert MM format (paired `_min`/`_max` columns) to a 3-dimensional array `[n, p, 2]`.

### Usage

```
MM_to_ARRAY(data)
```

### Arguments

| | |
|---|---|
| data | A data.frame in MM format with paired _min and _max columns. |

### Value

A numeric array of dimension [n, p, 2] with dimnames. Non-interval columns are excluded.

### Examples

```
data(mushroom.int)
mm <- RSDA_to_MM(mushroom.int, RSDA = FALSE)
arr <- MM_to_ARRAY(mm)
dim(arr)
```

---

## MM_to_iGAP *MM to iGAP*

---

### Description

To convert MM format to iGAP format.

### Usage

```
MM_to_iGAP(data)
```

### Arguments

| | |
|---|---|
| data | The dataframe with the MM format. |

### Value

Return a dataframe with the iGAP format.

### Examples

```
data(face.iGAP)
face <- iGAP_to_MM(face.iGAP, 1:6)
MM_to_iGAP(face)
```

## MM_to_RSDA *MM to RSDA*

### Description

To convert MM format interval dataframe to RSDA format (symbolic_tbl).

### Usage

```
MM_to_RSDA(data)
```

### Arguments

| | |
|---|---|
| data | The dataframe with the MM format (paired _min/_max columns). |

### Value

Return a symbolic_tbl dataframe with complex-encoded interval columns.

### Examples

```
data(mushroom.int)
mm <- RSDA_to_MM(mushroom.int, RSDA = FALSE)
rsda <- MM_to_RSDA(mm)
```

## mtcars.mix *Motor Trend Cars Mixed Symbolic Dataset*

### Description

Mixed symbolic dataset of 5 car groups from the mtcars data, with 7 interval-valued performance variables and 4 modal-valued categorical variables.

### Usage

```
data(mtcars.mix)
```

### Format

A symbolic data frame (symbolic_tbl) with 5 observations (car groups) and 11 variables:

- mpg: Interval-valued miles per gallon.
- cyl: Modal-valued number of cylinders.
- disp: Interval-valued displacement (cu.in.).
- hp: Interval-valued horsepower.

- `drat`: Interval-valued rear axle ratio.
- `wt`: Interval-valued weight (1000 lbs).
- `qsec`: Interval-valued quarter-mile time (seconds).
- `vs`: Modal-valued engine type (V/S).
- `am`: Modal-valued transmission type (auto/manual).
- `gear`: Modal-valued number of forward gears.
- `carb`: Modal-valued number of carburetors.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 5 |
| **Variables (p)** | 11 |
| **Subject area** | Automotive |
| **Symbolic format** | Mixed (interval, modal) |
| **Analytical tasks** | Descriptive statistics, Clustering |

## Source

ggInterval R package (`mtcars.i` dataset).

## References

Henderson, R. and Velleman, P. (1981). Building multiple regression models interactively. *Biometrics*, 37, 391–411.

Original data from the ggInterval R package (`mtcars.i` dataset).

## Examples

```
data(mtcars.mix)
```

---

# `mushroom.int` *Mushroom Species Interval Dataset*

---

## Description

Interval-valued version of the mushroom dataset. See `mushroom.int.mm`.

## Usage

```
data(mushroom.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 23 observations and 5 variables:

- `Species`: Mushroom species name (character).
- `Pileus.Cap.Width`: Pileus cap width range (cm, interval).
- `Stipe.Length`: Stipe length range (cm, interval).
- `Stipe.Thickness`: Stipe thickness range (cm, interval).
- `Edibility`: Edibility code (U = Unknown, Y = Yes, T = Toxic; character).

## Details

Contains the same corrected species names and interval endpoints as `mushroom.int.mm`. See that help page for the changes in version 0.2.7.4, including the corrected `bernardii` interval `[7, 16]`.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 23 |
| **Variables (p)** | 5 |
| **Subject area** | Biology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

Billard and Diday (2006), Table 3.3; corrected as described in `mushroom.int.mm`.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 3.3, p. 81. doi:10.1002/9780470090183.

## Examples

```
data(mushroom.int)
```

`mushroom.int.mm` *Mushroom Species Dataset (Min/Max Format)*

## Description

Interval-valued data for 23 mushroom species of the genus Agaricus with 3 morphological measurements from the Fungi of California Species.

## Usage

```
data(mushroom.int.mm)
```

## Format

A data frame with 23 observations and 8 columns representing 5 conceptual variables (species, 3 interval measurements, and edibility):

- `Species`: Mushroom species name (character).
- `Pileus.Cap.Width_min`, `Pileus.Cap.Width_max`: Minimum and maximum cap width (cm).
- `Stipe.Length_min`, `Stipe.Length_max`: Minimum and maximum stipe length (cm).
- `Stipe.Thickness_min`, `Stipe.Thickness_max`: Minimum and maximum stipe thickness (cm).
- `Edibility`: Edibility code (U = Unknown, Y = Yes, T = Toxic; character).

## Details

Classic SDA dataset used for descriptive statistics, histogram construction, and clustering of interval-valued data. Since version 0.2.7.4, the pileus cap width is `[7, 16]` for `bernardii` and `[6, 14]` for `subrutilescens`, following the MykoWeb species accounts. The former was previously `[7, 6]` here and `[6, 7]` in `mushroom.int`.

Eight original names have been corrected or updated to reflect the California species accounts: `arvenis` to `fissuratus`, `benesi` to `benesii`, `californinus` to `californicus`, `cupreo-brunneus` to `incultorum`, `diminutives` to `diminutivus`, `fuseo-fibrillosus` to `brunneofibrillosus`, `praeclaresquamosus` to `deardorffensis`, and `silvicola` to `sylvicola`. These include historical misapplications of names, not only spelling corrections. Original row order, other measurements, and edibility codes are preserved. Thanks to Diogo Pinheiro for the report and corrected data.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 23 |
| **Variables (p)** | 5 |
| **Subject area** | Biology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering, Descriptive statistics |

### Source

Billard, L. and Diday, E. (2006), Table 3.3, with corrections checked against MykoWeb's California Fungi species accounts on 6 November 2026: https://www.mykoweb.com/CAF/species_index.html.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Table 3.3, p. 81. doi:10.1002/9780470090183.

### Examples

```
data(mushroom.int.mm)
```

---

## mushroom_fuzzy.mix *Mushroom Species Fuzzy/Symbolic Dataset*

---

### Description

Extended mushroom data with fuzzy stipe thickness (Small/Average/Large), numerical stipe length, interval cap size, and categorical cap colour for two Amanita species (4 specimens).

### Usage

```
data(mushroom_fuzzy.mix)
```

### Format

A data frame with 4 observations (Mushroom1–Mushroom4) and 9 variables:

- `specimen`: Specimen identifier (character).
- `species`: Species name (character).
- `stipe_thickness`: Stipe thickness measurement (numeric, cm).
- `fuzzy_small`: Fuzzy membership degree for Small (numeric, 0–1).
- `fuzzy_average`: Fuzzy membership degree for Average (numeric, 0–1).
- `fuzzy_large`: Fuzzy membership degree for Large (numeric, 0–1).
- `stipe_length`: Stipe length (numeric, cm).
- `cap_size`: Cap size as interval string (e.g., "24 +/- 1", character).
- `cap_colour`: Cap colour (character).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 4 |
| **Variables (p)** | 9 |
| **Subject area** | Biology |
| **Symbolic format** | Fuzzy |
| **Analytical tasks** | Descriptive statistics |

### Source

Diday and Noirhomme-Fraiture (2008), Tables 1.14–1.16.

### References

Diday, E. and Noirhomme-Fraiture, M. (Eds.) (2008). *Symbolic Data Analysis and the SODAS Software*. Wiley. Tables 1.14-1.16.

### Examples

```
data(mushroom_fuzzy.mix)
```

---

## nycflights.int *New York City Flights Interval Dataset*

---

### Description

Interval-valued dataset with 142 units and four interval-valued variables from the nycflights13 package, aggregated by month and carrier.

### Usage

```
data(nycflights.int)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 142 observations and 5 variables:

- `X`: Month-carrier identifier (character).
- `dep_delay`: Departure delay range (minutes, interval).
- `arr_delay`: Arrival delay range (minutes, interval).
- `air_time`: Air time range (minutes, interval).
- `distance`: Distance range (miles, interval).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 142 |
| **Variables (p)** | 5 |
| **Subject area** | Transportation |
| **Symbolic format** | Interval |
| **Analytical tasks** | Regression, Descriptive statistics |

### Source

Derived from `MAINT.Data::FlightsIdt`. The upstream object removes incomplete rows from **nycflights13** and aggregates flights by month and carrier using the 0.05 and 0.95 quantiles.

### References

Duarte Silva, A. P., Brito, P., Filzmoser, P. and Dias, J. G. (2021). MAINT.Data: Modelling and Analysing Interval Data in R. *The R Journal*, 13(2), 336–364. doi:10.32614/RJ2021074.

### Examples

```
data(nycflights.int)
```

---

## `occupations.modal` *Occupation Salaries Dataset*

---

### Description

Modal-valued dataset of 9 occupations with gender and salary distributions. This is the wide (flat table) format; see `occupations2.modal` for the modal-valued version.

### Usage

```
data(occupations.modal)
```

### Format

A data frame with 9 observations and 11 columns:

- `Occupation`: Occupation name (character).
- `Gender(M)`, `Gender(F)`: Proportion male/female (2 bins).
- `Salary(1)` through `Salary(7)`: Salary distribution across 7 ordered bins (proportions).
- `n`: Sample size (integer).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 9 |
| **Variables (p)** | 11 |
| **Subject area** | Sociology |
| **Symbolic format** | Modal |
| **Analytical tasks** | Descriptive statistics, Clustering |

### Source

Billard and Diday (2006).

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

### Examples

```
data(occupations.modal)
```

---

## occupations2.modal *Occupation Salaries Modal-Valued Dataset*

---

### Description

Modal-valued version of the occupation salaries dataset. See `occupations.modal` for the wide-format version.

### Usage

```
data(occupations2.modal)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 9 observations and 4 variables:

- `Occupation`: Occupation name (character).
- `Gender`: Modal distribution over gender (Male, Female).
- `Salary`: Modal distribution over 7 ordered salary bins.
- `n`: Sample size (numeric).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 9 |
| **Variables (p)** | 4 |
| **Subject area** | Sociology |
| **Symbolic format** | Modal |
| **Analytical tasks** | Descriptive statistics, Clustering |

## Source

Billard and Diday (2006).

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

## Examples

```
data(occupations2.modal)
```

---

`ohtemp.int` *Ohio River Basin 30-Year Trimmed Mean Daily Temperatures Interval Dataset*

---

## Description

Interval-valued dataset of 30-year trimmed mean daily temperatures for the Ohio river basin. Intervals are defined by the mean daily maximum and minimum temperatures from January 1, 1988 to December 31, 2018.

## Usage

```
data(ohtemp.int)
```

## Format

A data frame with 161 rows and 7 variables:

- `ID`: Global Historical Climatological Network (GHCN) station identifier.
- `NAME`: GHCN station name.
- `STATE`: Two-digit state designation.
- `LATITUDE`: Latitude coordinate position.
- `LONGITUDE`: Longitude coordinate position.
- `ELEVATION`: Elevation of the measurement location (meters).
- `TEMPERATURE`: 30-year mean daily temperature (tenths of degrees Celsius).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 161 |
| **Variables (p)** | 7 |
| **Subject area** | Climate |
| **Symbolic format** | Interval |
| **Analytical tasks** | Regression, Spatial analysis |

### Source

`intkrige::ohtemp`, derived from NOAA Global Historical Climatology Network station records for January 1, 1988–December 31, 2018.

### References

Bean, B. (2025). *intkrige: A Numerical Implementation of Interval-Valued Kriging*. R package. doi:10.32614/CRAN.package.intkrige.

### Examples

```
data(ohtemp.int)
```

---

## `oils.int` *Oils and Fats Interval Dataset*

---

### Description

Classic benchmark interval-valued data for 8 oils and fats described by 4 physico-chemical properties. Originally from Ichino (1988).

### Usage

```
data(oils.int)
```

### Format

A data frame with 8 observations and 9 columns (4 interval variables in `_l`/`_u` Min-Max pairs, plus a label):

- `sample`: Oil/fat sample name (character).
- `specific_gravity_l`, `specific_gravity_u`: Specific gravity range.
- `freezing_point_l`, `freezing_point_u`: Freezing point range (degrees Celsius).
- `iodine_value_l`, `iodine_value_u`: Iodine value range.
- `saponification_value_l`, `saponification_value_u`: Saponification value range.

## Details

The 8 samples are: Linseed oil, Perilla oil, Cottonseed oil, Sesame oil, Camellia oil, Olive oil, Beef tallow, Hog fat. The expected 3-cluster structure is: {Beef tallow, Hog fat}, {Cottonseed, Sesame, Camellia, Olive}, and {Linseed, Perilla}. Widely used for comparing clustering methods and distance measures in symbolic data analysis.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 8 |
| **Variables (p)** | 9 |
| **Subject area** | Chemistry |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

Diday and Noirhomme-Fraiture (2008), Table 13.7; also distributed as `RSDA::oils`.

## References

Ichino, M. (1988). General metrics for mixed features. *Proc. IEEE Conf. Systems, Man, and Cybernetics*, pp. 494-497.

Diday, E. and Noirhomme-Fraiture, M. (Eds.) (2008). *Symbolic Data Analysis and the SODAS Software*. Wiley. Table 13.7, p.253.

## Examples

```
data(oils.int)
```

---

# ozone.hist *Ozone Air Quality Histogram-Valued Dataset*

---

## Description

Histogram-valued dataset for 84 U.S. air-quality monitoring stations with 4 weather-related histogram variables. Each histogram has 10 equal-probability (decile) bins.

## Usage

```
data(ozone.hist)
```

### Format

A data frame with 84 observations (stations) and 4 histogram-valued variables:

- `Ozone.Conc.ppb`: Histogram of ozone concentration (ppb).
- `Temperature.C`: Histogram of temperature (Celsius).
- `Solar.Radiation.WattM2`: Histogram of solar radiation (W/m^2).
- `Wind.Speed.mSec`: Histogram of wind speed (m/s).

Row names are I1 through I84.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 84 |
| **Variables (p)** | 4 |
| **Subject area** | Environment |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Regression, Clustering |

### Source

Derived from `HistDAWass::OzoneH`, which documents 84 U.S. monitoring stations and identifies EPA CASTNET/metdata as the source. The original 100 quantile bins were reduced to 10 decile bins here.

### References

U.S. Environmental Protection Agency. Clean Air Status and Trends Network (CASTNET) ozone and associated meteorological data.

### Examples

```
data(ozone.hist)
```

---

## `petrobras.its` *Petrobras Stock Daily High/Low Interval Time Series*

---

### Description

Daily high and low stock prices of Petrobras (ADR traded on NYSE) from January 3, 2005 to December 29, 2006 (503 trading days). Maia, de Carvalho and Ludermir (2008) provide related interval time-series methodology; this object is specifically the independently downloaded NYSE Petrobras ADR.

## Usage

```
data(petrobras.its)
```

## Format

A data frame with 503 observations and 3 variables:

- `date`: Trading date (Date class).
- `low`: Daily low price (USD).
- `high`: Daily high price (USD).

## Details

Petrobras (Petroleo Brasileiro S.A.) is the Brazilian multinational petroleum corporation. The ADR (American Depositary Receipt) is traded on the New York Stock Exchange under ticker PBR. Each observation represents a trading day with the daily low and high prices forming an interval. This was one of the first datasets used to demonstrate interval-valued autoregressive (iAR) models.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 503 |
| **Variables (p)** | 3 (date, low, high) |
| **Subject area** | Finance |
| **Symbolic format** | Interval time series |
| **Analytical tasks** | Forecasting, Time series analysis |

## Source

Yahoo Finance, NYSE ADR ticker `PBR`, January 3, 2005–December 29, 2006. The Yahoo `Low` and `High` fields were retained after download via **quantmod**; the original retrieval date was not recorded.

## References

Maia, A. L. S., de Carvalho, F. A. T. and Ludermir, T. B. (2008). Forecasting models for interval-valued time series. *Neurocomputing*, **71**(16–18), 3344–3352. doi:10.1016/j.neucom.2008.02.022.

## Examples

```
data(petrobras.its)
head(petrobras.its)
plot(petrobras.its$date, petrobras.its$high, type = "l", col = "red",
     ylab = "Price (USD)", xlab = "Date",
     main = "Petrobras Daily High/Low (2005-2006)")
lines(petrobras.its$date, petrobras.its$low, col = "blue")
legend("topleft", c("High", "Low"), col = c("red", "blue"), lty = 1)
```

`polish_cars.mix` *Polish Car Models Mixed Symbolic Dataset*

## Description

Mixed symbolic dataset of 30 car models sold in Poland, with 9 interval-valued technical specification variables and 3 multinomial-valued categorical variables.

## Usage

```
data(polish_cars.mix)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 30 observations and 12 variables:

- `price`: Interval-valued price (PLN).
- `body`: Multinomial body types (e.g., hatchback, sedan, combi).
- `wheelbase`: Interval-valued wheelbase (mm).
- `chassis_length`: Interval-valued chassis length (mm).
- `chassis_width`: Interval-valued chassis width (mm).
- `chassis_height`: Interval-valued chassis height (mm).
- `engine_capacity`: Multinomial engine displacement categories (litres).
- `engine_power`: Interval-valued engine power (HP).
- `maximum_speed`: Interval-valued maximum speed (km/h).
- `acceleration`: Interval-valued 0–100 km/h time (seconds).
- `fuel_type`: Multinomial fuel types (petrol, diesel, LPG).
- `fuel_consumption`: Interval-valued fuel consumption (L/100km).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 30 |
| **Variables (p)** | 12 |
| **Subject area** | Automotive |
| **Symbolic format** | Mixed (interval, multinomial) |
| **Analytical tasks** | Clustering, Descriptive statistics |

## Source

`symbolicDA::cars`; the upstream documentation states that the data were collected from authorized Polish car-dealer websites for the Polish market in 2010.

## References

Dudek, A., Pelka, M., Wilk, J. and Walesiak, M. *symbolicDA: Analysis of Symbolic Data*. R package. doi:10.32614/CRAN.package.symbolicDA.

## Examples

```
data(polish_cars.mix)
```

---

`polish_voivodships.int`

*Polish Voivodships Socio-Economic Intervals*

---

## Description

Interval-valued dataset of 18 Polish voivodships (administrative regions) with 9 socio-economic interval variables describing demographic and economic characteristics at the county (powiat) level.

## Usage

```
data(polish_voivodships.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 18 observations (voivodships) and 9 interval-valued variables:

- `V1` through `V9`: Interval-valued socio-economic indicators aggregated across counties within each voivodship.

Row names are voivodship names (e.g., Dolnoslaskie, Lubelskie).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 18 |
| **Variables (p)** | 9 |
| **Subject area** | Socioeconomics |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

`clusterSim::data_symbolic_interval_polish_voivodships`. The data were aggregated by voivodship from the Polish Central Statistical Office Local Data Bank for 2016.

## References

Walesiak, M. and Dudek, A. (2020). The choice of variable normalization method in cluster analysis. In K. S. Soliman (Ed.), *Education Excellence and Innovation Management: A 2025 Vision to Sustain Economic Development During Global Challenges*, 325–340.

## Examples

```
data(polish_voivodships.int)
```

---

`profession.int` *Profession Work Salary Time Interval Dataset*

---

## Description

Interval-valued data for 15 profession entries classified by work type (White Collar / Blue Collar). Each entry describes a specific profession with salary and working duration ranges.

## Usage

```
data(profession.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 15 observations and 4 variables:

- `Type_of_Work`: Work category (White Collar or Blue Collar, character).
- `Profession`: Profession name (character).
- `Salary`: Salary range (currency units, interval).
- `Duration`: Working duration range (hours per week, interval).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 15 |
| **Variables (p)** | 4 |
| **Subject area** | Sociology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Classification |

## Source

Billard and Diday (2006).

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

## Examples

```
data(profession.int)
```

---

`prostate.int` *Prostate Cancer Clinical Interval Dataset*

---

## Description

Interval-valued clinical measurements for 97 prostate cancer patients (training and test sets combined). Contains 9 interval-valued variables from log-transformed cancer volume, weight, age, and other clinical predictors.

## Usage

```
data(prostate.int)
```

## Format

A data frame with 97 observations and 9 interval-valued variables:

- `lcavol`: Log cancer volume range.
- `lweight`: Log prostate weight range.
- `age`: Patient age range.
- `lbph`: Log benign prostatic hyperplasia amount range.
- `svi`: Seminal vesicle invasion range.
- `lcp`: Log capsular penetration range.
- `gleason`: Gleason score range.
- `pgg45`: Percentage Gleason scores 4 or 5 range.
- `lpsa`: Log prostate specific antigen range.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 97 |
| **Variables (p)** | 9 |
| **Subject area** | Medical |
| **Symbolic format** | Interval |
| **Analytical tasks** | Regression |

## Source

Extracted from RSDA package (`int_prost_train`, `int_prost_test`).

## References

Stamey, T. et al. (1989). Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate. II. Radical prostatectomy treated patients. *J. Urology*, 141(5), 1076-1083.

## Examples

```
data(prostate.int)
```

---

`read_symbolic_csv` *Read a Symbolic Data CSV File*

---

## Description

Reads an external CSV file containing symbolic data, automatically detects whether the data is interval-valued (min/max pairs or comma-separated), histogram-valued, modal-valued, or another symbolic type, and returns an appropriate R object.

## Usage

```
read_symbolic_csv(
  file,
  sep = ",",
  header = TRUE,
  row.names = NULL,
  stringsAsFactors = FALSE,
  na.strings = c("", "NA"),
  symbolic_type = NULL,
  ...
)
```

## Arguments

| | |
|---|---|
| `file` | Path to the CSV file to read. |
| `sep` | Field separator character. Default `","`. |
| `header` | Logical; does the first row contain column names? Default `TRUE`. |
| `row.names` | Column number or character string giving row names. Passed to `read.table`. Default `NULL` (automatic). |
| `stringsAsFactors` | Logical; should character columns be converted to factors? Default `FALSE`. |
| `na.strings` | Character vector of strings to interpret as `NA`. Default `c("", "NA")`. |
| `symbolic_type` | Optional character string to override automatic type detection. One of `"interval"`, `"histogram"`, `"modal"`, or `"other"`. When `NULL` (the default) the type is detected automatically. |
| `...` | Additional arguments passed to `read.table`. |

## Details

The detection heuristic works as follows:

1. **Interval (MM)**: If the file contains paired `_min`/`_max` columns the data is returned as-is (MM format).
2. **Interval (iGAP)**: If one or more character columns contain comma-separated numeric pairs (e.g., `"1.2,3.4"`) they are expanded into `_min`/`_max` column pairs and the result is returned in MM format.
3. **Histogram / Modal**: If columns follow a `VarName(bin)` naming pattern (e.g., `Crime(violent)`) and the proportions within each variable group sum to approximately 1, the data is classified as histogram or modal. It is returned as a plain `data.frame`.
4. **Other**: If none of the above patterns match, the data is returned as a plain `data.frame`.

## Value

A `data.frame`. Interval data is returned in MM format (paired `_min`/`_max` columns). All other symbolic types are returned as plain data frames.

## See Also

`write_symbolic_csv`, `int_detect_format`, `int_convert_format`

## Examples

```
# Write then read back an interval dataset
data(mushroom.int.mm)
tmp <- tempfile(fileext = ".csv")
write_symbolic_csv(mushroom.int.mm, tmp)
df <- read_symbolic_csv(tmp)
head(df)

# Write then read back a histogram dataset
data(airline_flights.hist)
tmp2 <- tempfile(fileext = ".csv")
write_symbolic_csv(airline_flights.hist, tmp2)
df2 <- read_symbolic_csv(tmp2)
head(df2)
```

---

# RSDA_format *RSDA Format*

---

## Description

This function changes the format of the data to conform to RSDA format.

## Usage

```
RSDA_format(data, sym_type1 = NULL, location = NULL, sym_type2 = NULL, var = NULL)
```

### Arguments

| | |
|---|---|
| data | A conventional data. |
| sym_type1 | The labels I means an interval variable and $S means set variable. |
| location | The location of the sym_type in the data. |
| sym_type2 | The labels I means an interval variable and $S means set variable. |
| var | The name of the symbolic variable in the data. |

### Value

Return a dataframe with a label added to the previous column of symbolic variable.

### Examples

```
data("mushroom.int.mm")
mushroom.set <- set_variable_format(data = mushroom.int.mm, location = 8, var = "Species")
mushroom.tmp <- RSDA_format(data = mushroom.set, sym_type1 = c("I", "S"),
                            location = c(25, 31), sym_type2 = c("S", "I", "I"),
                           var = c("Species", "Stipe.Length_min", "Stipe.Thickness_min"))
```

---

## RSDA_to_ARRAY *RSDA to ARRAY*

---

### Description

Convert RSDA format (symbolic_tbl) to a 3-dimensional array `[n, p, 2]` where slice `[,,1]` contains the minima and slice `[,,2]` contains the maxima.

### Usage

```
RSDA_to_ARRAY(data)
```

### Arguments

| | |
|---|---|
| data | A symbolic_tbl with interval columns. |

### Value

A numeric array of dimension `[n, p, 2]` with dimnames. Only interval (symbolic_interval) columns are included.

### Examples

```
data(mushroom.int)
arr <- RSDA_to_ARRAY(mushroom.int)
dim(arr)  # [23, 3, 2]
```

## RSDA_to_iGAP *RSDA to iGAP*

### Description

To convert RSDA format interval dataframe to iGAP format.

### Usage

```
RSDA_to_iGAP(data)
```

### Arguments

| | |
|---|---|
| data | The RSDA format with interval dataframe. |

### Value

Return a dataframe with the iGAP format.

### Examples

```
data(mushroom.int)
RSDA_to_iGAP(mushroom.int)
```

## RSDA_to_MM *RSDA to MM*

### Description

To convert RSDA format interval dataframe to MM format.

### Usage

```
RSDA_to_MM(data, RSDA = TRUE)
```

### Arguments

| | |
|---|---|
| data | The RSDA format with interval dataframe. |
| RSDA | Whether to load the RSDA package. |

### Value

Return a dataframe with the MM format.

### Examples

```
data(mushroom.int)
RSDA_to_MM(mushroom.int, RSDA = FALSE)
```

search_data *Search Datasets*

## Description

Search and filter the dataSDA dataset catalog by metadata criteria including sample size, number of variables, subject area, symbolic format, analytical tasks, keywords, and book reference.

## Usage

```
search_data(...)
```

## Arguments

| | |
|---|---|
| ... | Filter expressions. Each argument is a comparison expression evaluated against the dataset metadata. Supported columns: |
| | `n` Sample size (numeric). Operators: `==`, `>`, `<`, `>=`, `<=`. |
| | `p` Number of variables (numeric). Operators: `==`, `>`, `<`, `>=`, `<=`. |
| | `subject` Subject area (character). Case-insensitive partial match with `==`. Areas: Agriculture, Automotive, Biology, Biometrics, Botany, Chemistry, Climate, Criminology, Demographics, Digital media, Economics, Education, Energy, Engineering, Environment, Finance, Food science, Forestry, Genomics, Healthcare, Marine biology, Medical, Methodology, Public services, Socioeconomics, Sociology, Sports, Transportation, Zoology. |
| | `type` Symbolic format (character). Exact match with `==`. Types correspond to the dataset name suffix: `"int"` (interval), `"hist"` (histogram), `"mix"` (mixed), `"distr"` (distribution), `"its"` (interval time series), `"modal"` (modal), `"iGAP"` (interval in iGAP format). |
| | `task` Analytical tasks (character). Case-insensitive partial match with `==`. Tasks: Clustering, Classification, Regression, PCA, Descriptive statistics, Discriminant analysis, Visualization, Spatial analysis, Time series, Aggregation. |
| | `tag` Keywords (character). Case-insensitive partial match with `==`. Use `tag == "all"` to list all datasets. |
| | `book` Book reference short name (character). Case-insensitive partial match with `==`. Available books: `SDA_2006` (Billard & Diday, 2006), `CMD_2020` (Billard & Diday, 2020), `SODAS_2008` (Diday & Noirhomme-Fraiture, 2008). |

## Details

For character columns (`subject`, `type`, `task`, `tag`, `book`), the `==` operator performs a case-insensitive substring match (using `grepl`). The `type` column uses short suffix-based labels that match the dataset name suffix (e.g., `type == "int"` matches all `.int` datasets).

For numeric columns (`n`, `p`), standard comparison operators are used with exact semantics.

When no arguments are provided, or when `tag == "all"` is used, all datasets are returned.

## Value

A data frame with one row per matching dataset and the following columns: `name`, `n`, `p`, `subject`, `type`, `task`, `tag`, `book`.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester.

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley.

Diday, E. and Noirhomme-Fraiture, M. (Eds.) (2008). *Symbolic Data Analysis and the SODAS Software*. Wiley.

## Examples

```
# List all datasets
search_data()

# Filter by symbolic format (suffix-based)
search_data(type == "hist")

# Filter by analytical task and size
search_data(task == "Regression", n > 10)

# Filter by book reference
search_data(book == "SDA_2006")

# Combine multiple filters
search_data(type == "int", task == "Clustering", subject == "Biology")

# Filter by size range
search_data(n >= 20, n <= 100, p < 10)
```

---

# `set_variable_format` *Set Variable Format*

---

## Description

This function changes the format of the set variables in the data to conform to the RSDA format.

## Usage

```
set_variable_format(data, location = NULL, var = NULL)
```

## Arguments

| | |
|---|---|
| `data` | A conventional data. |
| `location` | The location of the set variable in the data. |
| `var` | The name of the set variable in the data. |

### Value

Return a dataframe in which a set variable is converted to one-hot encoding.

### Examples

```
data("mushroom.int.mm")
mushroom.set <- set_variable_format(data = mushroom.int.mm, location = 8, var = "Species")
```

---

## shanghai_stock.its *Shanghai Stock Exchange Composite Index Daily High/Low Interval Time Series*

---

### Description

Daily high and low values of the Shanghai Stock Exchange Composite Index (SSE Composite) from January 2, 2019 to December 30, 2022 (970 trading days). Yang, Zhang and Wang (2026) provide a related interval autoregressive application; this object was independently downloaded from Yahoo Finance.

### Usage

```
data(shanghai_stock.its)
```

### Format

A data frame with 970 observations and 3 variables:

- `date`: Trading date (Date class).
- `low`: Daily low value of the SSE Composite Index.
- `high`: Daily high value of the SSE Composite Index.

### Details

The SSE Composite Index is the most commonly used indicator to reflect the performance of the Shanghai Stock Exchange. It tracks all stocks (A-shares and B-shares) listed on the exchange. This dataset covers a period that includes the COVID-19 pandemic and its market impacts, providing a rich testbed for evaluating interval forecasting models under extreme volatility.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 970 |
| **Variables (p)** | 3 (date, low, high) |
| **Subject area** | Finance |
| **Symbolic format** | Interval time series |
| **Analytical tasks** | Forecasting, Time series analysis |

## Source

Yahoo Finance, ticker `000001.SS`, January 2, 2019–December 30, 2022. The Yahoo `Low` and `High` fields were retained after download via **quantmod**; the original retrieval date was not recorded.

## References

Yang, K., Zhang, G. and Wang, D. (2026). On smooth transition interval autoregressive models. *Journal of Business & Economic Statistics*, **44**(2), 613–625. doi:10.1080/07350015.2025.2546455.

## Examples

```
data(shanghai_stock.its)
head(shanghai_stock.its)
plot(shanghai_stock.its$date, shanghai_stock.its$high, type = "l",
     col = "red", ylab = "Index Value", xlab = "Date",
     main = "Shanghai Composite Daily High/Low (2019-2022)")
lines(shanghai_stock.its$date, shanghai_stock.its$low, col = "blue")
legend("topleft", c("High", "Low"), col = c("red", "blue"), lty = 1)
```

---

# `simulated.hist` *Simulated Histogram-Valued Dataset*

---

## Description

Small simulated histogram-valued dataset of 5 observations with 2 histogram-valued variables. Useful for testing and demonstrating histogram-valued statistical methods.

## Usage

```
data(simulated.hist)
```

## Format

A data frame with 5 observations and 2 histogram-valued variables:

- `Y1`: Histogram-valued variable 1.
- `Y2`: Histogram-valued variable 2.

Row names are Obs_1 through Obs_5.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 5 |
| **Variables (p)** | 2 |
| **Subject area** | Methodology |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering |

## Source

Billard, L. and Diday, E. (2020), Table 7-26.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 7-26.

## Examples

```
data(simulated.hist)
```

---

`soccer_bivar.int` *French Soccer Championship Bivariate Interval Dataset*

---

## Description

Interval-valued data for 20 teams from the French premier soccer championship. Contains ranges of Weight (response), Height and Age (explanatory variables).

## Usage

```
data(soccer_bivar.int)
```

## Format

A data frame with 20 rows and 3 interval-valued variables:

- `y`: Weight (response variable, kg).
- `t1`: Height (explanatory variable, cm).
- `t2`: Age (explanatory variable, years).

## Details

Since version 0.2.7.3, `t1` is the height interval `[t1Min, t1Max]` and `t2` is the age interval `[t2Min, t2Max]` of `soccer.bivar` in **iRegression**. Earlier versions mis-paired the source columns, storing `[height min, age min]` in `t1` and `[height max, age max]` in `t2`. No values were changed; only the endpoint pairing was corrected.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 20 |
| **Variables (p)** | 3 |
| **Subject area** | Sports |
| **Symbolic format** | Interval |
| **Analytical tasks** | Regression |

## Source

French soccer interval data distributed with the **iRegression** package and analyzed by Lima Neto et al. (2011).

## References

Lima Neto, E. A., Cordeiro, G. and De Carvalho, F.A.T. (2011). Bivariate symbolic regression models for interval-valued variables. *Journal of Statistical Computation and Simulation*, 81, 1727–1744. doi:10.1080/00949655.2010.500470.

## Examples

```
data(soccer_bivar.int)
```

---

# SODAS_to_ARRAY *SODAS to ARRAY*

---

## Description

Convert SODAS format (XML file) to a 3-dimensional array `[n, p, 2]`.

## Usage

```
SODAS_to_ARRAY(XMLPath)
```

## Arguments

| | |
|---|---|
| `XMLPath` | Disk path where the SODAS `*.XML` file is. |

## Value

A numeric array of dimension `[n, p, 2]` with dimnames.

## Examples

```
## Not run:
arr <- SODAS_to_ARRAY("C:/Users/user/AppData/abalone.xml")

## End(Not run)
```

## SODAS_to_iGAP *SODAS to iGAP*

### Description

To convert SODAS format interval dataframe to the iGAP format.

### Usage

```
SODAS_to_iGAP(XMLPath)
```

### Arguments

| | |
|---|---|
| XMLPath | Disk path where the SODAS *.XML file is. |

### Value

Return a dataframe with the iGAP format.

### Examples

```
## Not run:
# Read from a SODAS XML file:
abalone <- SODAS_to_iGAP("C:/Users/user/AppData/abalone.xml")

## End(Not run)
```

## SODAS_to_MM *SODAS to MM*

### Description

To convert SODAS format interval dataframe to the MM format.

### Usage

```
SODAS_to_MM(XMLPath)
```

### Arguments

| | |
|---|---|
| XMLPath | Disk path where the SODAS *.XML file is. |

### Value

Return a dataframe with the MM format.

## Examples

```
## Not run:
# Read from a SODAS XML file:
abalone <- SODAS_to_MM("C:/Users/user/AppData/abalone.xml")

## End(Not run)
```

---

`sp500.its` *S&P 500 Daily High/Low Interval Time Series*

---

## Description

Daily high and low prices of the S&P 500 index from January 2, 2004 to December 30, 2005 (504 trading days). The independently downloaded Yahoo Finance series is provided as a benchmark for interval time-series forecasting; it is not the same sample as Arroyo, Gonzalez-Rivera and Mate (2011).

## Usage

```
data(sp500.its)
```

## Format

A data frame with 504 observations and 3 variables:

- `date`: Trading date (Date class).
- `low`: Daily low price of the S&P 500 index.
- `high`: Daily high price of the S&P 500 index.

## Details

The S&P 500 is a market-capitalization-weighted index of 500 leading publicly traded companies in the United States. Each observation represents a trading day with the daily low and high prices forming an interval. This dataset has been widely used to evaluate interval-valued autoregressive models, exponential smoothing methods for intervals, and center-and-range forecasting approaches.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 504 |
| **Variables (p)** | 3 (date, low, high) |
| **Subject area** | Finance |
| **Symbolic format** | Interval time series |
| **Analytical tasks** | Forecasting, Time series analysis |

### Source

Yahoo Finance, ticker `^GSPC`, January 2, 2004–December 30, 2005. The Yahoo `Low` and `High` fields were retained after download via **quantmod**; the original retrieval date was not recorded.

### References

Arroyo, J., Gonzalez-Rivera, G. and Mate, C. (2011). Forecasting with interval and histogram data: Some financial applications. In *Handbook of Empirical Economics and Finance*, pp. 247–280. Chapman and Hall/CRC.

### Examples

```
data(sp500.its)
head(sp500.its)
plot(sp500.its$date, sp500.its$high, type = "l", col = "red",
     ylab = "Price", xlab = "Date", main = "S&P 500 Daily High/Low")
lines(sp500.its$date, sp500.its$low, col = "blue")
legend("topleft", c("High", "Low"), col = c("red", "blue"), lty = 1)
```

---

## `state_income.hist` *State Income Histogram-Valued Dataset*

---

### Description

Histogram-valued dataset of 6 US states with 4 income distribution histograms. Each histogram describes the distribution of household income within a state.

### Usage

```
data(state_income.hist)
```

### Format

A data frame with 6 observations (states) and 4 histogram-valued variables:

- `Y1` through `Y4`: Histogram-valued income distribution variables.

Row names are State_1 through State_6.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 6 |
| **Variables (p)** | 4 |
| **Subject area** | Economics |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Clustering |

## Source

Billard, L. and Diday, E. (2020), Table 7-18.

## References

Billard, L. and Diday, E. (2020). *Clustering Methodology for Symbolic Data*. Wiley, Chichester. Table 7-18.

## Examples

```
data(state_income.hist)
```

---

# synthetic_clusters.int

*Synthetic Interval Clusters Dataset*

---

## Description

Synthetic interval-valued dataset with 125 observations in 5 groups of 25 each, described by 6 interval-valued variables and a cluster label. Designed for benchmarking interval data clustering algorithms.

## Usage

```
data(synthetic_clusters.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 125 observations and 7 variables:

- `V1` through `V6`: Six interval-valued variables.
- `class`: Cluster membership (1–5, set-valued).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 125 |
| **Variables (p)** | 7 |
| **Subject area** | Methodology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

Artificial benchmark data from `symbolicDA::data_symbolic`.

## References

Dudek, A., Pelka, M., Wilk, J. and Walesiak, M. *symbolicDA: Analysis of Symbolic Data*. R package. doi:10.32614/CRAN.package.symbolicDA.

## Examples

```
data(synthetic_clusters.int)
```

---

`teams.int` *Pickup League Teams Interval Dataset*

---

## Description

Interval-valued data for 5 teams in a local pickup league, classified by season performance. Each team is described by ranges of player age, weight, and speed.

## Usage

```
data(teams.int)
```

## Format

A data frame with 5 observations and 7 columns (3 interval variables in `_l`/`_u` Min-Max pairs, plus a label):

- `team_type`: Performance category (Very Good, Good, Average, Fair, Poor).
- `age_l`, `age_u`: Player age range (years).
- `weight_l`, `weight_u`: Player weight range (pounds).
- `speed_l`, `speed_u`: Speed range – time to run 100 yards (seconds).

## Details

The symbolic results are more informative than classical midpoint analyses: the Very Good team has homogeneous players, whereas the Poor team has players varying widely in age, weight, and speed. Used for symbolic principal component analysis.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 5 |
| **Variables (p)** | 7 |
| **Subject area** | Sports |
| **Symbolic format** | Interval |
| **Analytical tasks** | PCA |

## Source

Billard and Diday (2006), Table 2.24, p. 63.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.24, p.63.

## Examples

```
data(teams.int)
```

---

# temperature_city.int *World Cities Monthly Temperature Interval Dataset*

---

## Description

Interval-valued monthly temperatures for major cities worldwide. Benchmark dataset for comparing distance measures (Hausdorff, L2, Wasserstein) in dynamic clustering algorithms.

## Usage

```
data(temperature_city.int)
```

## Format

A data frame with 6 observations and 13 columns (6 monthly interval variables in _l/_u Min-Max pairs, plus a label). Only January through June are included:

- `city`: City name (character).
- `jan_l`, `jan_u`: January temperature range (degrees Celsius).
- `feb_l`, `feb_u`: February temperature range.
- `mar_l`, `mar_u`: March temperature range.
- `apr_l`, `apr_u`: April temperature range.
- `may_l`, `may_u`: May temperature range.
- `jun_l`, `jun_u`: June temperature range.

## Details

Expert partition into 4 classes: Class 1 (tropical/warm), Class 2 (temperate European and Asian), Class 3 (Mauritius), Class 4 (Tehran).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 6 |
| **Variables (p)** | 13 |
| **Subject area** | Climate |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

### Source

Interval-valued city-temperature data analyzed by Verde and Irpino (2008).

### References

Verde, R. and Irpino, A. (2008). A new interval data distance based on the Wasserstein metric. *Proc. COMPSTAT 2008*, pp. 705–712. doi:10.1007/9783540782469_83.

### Examples

```
data(temperature_city.int)
```

---

## tennis.int *Tennis Court Types Interval Dataset*

---

### Description

Interval-valued data for tennis players aggregated by court type (Hard, Grass, Indoor, Clay) with weight, height, and racket tension.

### Usage

```
data(tennis.int)
```

### Format

A data frame with 4 observations and 7 columns (3 interval variables in `_l`/`_u` Min-Max pairs, plus a label):

- `court_type`: Type of court (Hard, Grass, Indoor, Clay).
- `player_weight_l`, `player_weight_u`: Player weight range (kg).
- `player_height_l`, `player_height_u`: Player height range (m).
- `racket_tension_l`, `racket_tension_u`: Racket tension range.

## Details

Clustering on weight and height separates grass courts from the rest (decision rule: Weight <= 74.75 kg). When all three variables are used, clustering separates by racket tension instead.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 4 |
| **Variables (p)** | 7 |
| **Subject area** | Sports |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

Billard and Diday (2006), Table 2.25, p. 64.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 2.25, p.64.

## Examples

```
data(tennis.int)
```

---

`town_services.mix` *Town Services Concatenated Mixed Symbolic Dataset*

---

## Description

Symbolic data for 3 towns (Paris, Lyon, Toulouse) combining school and hospital databases. Contains interval-valued, multi-valued, and modal-valued variables.

## Usage

```
data(town_services.mix)
```

## Format

A data frame with 3 observations (Paris, Lyon, Toulouse) and 8 columns:

- `town`: Town name (character).
- `no_pupils_l`, `no_pupils_u`: Number of pupils range (Min-Max pair).
- `type`: School type (modal, character).
- `level`: Coded level (multi-valued, character).
- `no_beds_l`, `no_beds_u`: Number of beds range (Min-Max pair).
- `specialty`: Specialty code (multi-valued, character).

### Metadata

| | |
|---|---|
| **Sample size (n)** | 3 |
| **Variables (p)** | 8 |
| **Subject area** | Public services |
| **Symbolic format** | Mixed (interval, modal, multi-valued) |
| **Analytical tasks** | Descriptive statistics |

### Source

Diday and Noirhomme-Fraiture (2008), Table 1.21, p. 19.

### References

Diday, E. and Noirhomme-Fraiture, M. (Eds.) (2008). *Symbolic Data Analysis and the SODAS Software*. Wiley. Table 1.21, p.19.

### Examples

```
data(town_services.mix)
```

---

## to_all_interval_formats

*Convert Interval Data to All Supported Formats*

---

### Description

Convert interval data from any recognized format to all six supported interval data formats and return the results as a named list. This is useful for inspecting and comparing how the same interval data is represented across different formats.

### Usage

```
to_all_interval_formats(x, ...)
```

### Arguments

| | |
|---|---|
| `x` | Interval data in one of the supported formats: `"RSDA"`, `"MM"`, `"iGAP"`, `"ARRAY"`, `"SODAS"`, or `"SDS"`. |
| `...` | Additional arguments passed to conversion functions (e.g., `location` for iGAP input). |

## Details

Six interval data formats are supported in this package. Each format stores the same information – lower and upper bounds for every variable of every observation – but differs in its structure and origin:

**RSDA** A `symbolic_tbl` object (class `c("symbolic_tbl", "tbl_df", "tbl", "data.frame")`) where each interval variable is a complex column (`symbolic_interval`): `Re()` gives the minimum and `Im()` gives the maximum. This is the native format of the **RSDA** package (Billard & Diday, 2006; Rodriguez, 2024).

**MM (Min-Max)** A plain `data.frame` where each interval variable is represented by two numeric columns named `<var>_min` and `<var>_max`. This is a widely used general-purpose representation.

**iGAP** A `data.frame` where each interval variable is stored as a character column with comma-separated values `"min,max"`. This is the format used by the **iGAP** software (Correia, 2009).

**ARRAY** A three-dimensional numeric `array` of size `[n, p, 2]`. The first slice `[,,1]` contains all minima and the second slice `[,,2]` contains all maxima. Dimnames encode observation labels, variable names, and `c("min", "max")`. This format is convenient for matrix-based computations.

**SODAS** An XML file on disk produced by the SODAS software (Diday & Noirhomme, 2008). In R, SODAS data is referenced by its file path and read via `RSDA::SODAS.to.RSDA()`. Since SODAS is a file-based format, it cannot be generated from in-memory data.

**SDS** An alias for SODAS. Both refer to the same XML-based format.

## Value

A named list with six slots:

`RSDA` A `symbolic_tbl` with complex-encoded `symbolic_interval` columns.

`MM` A `data.frame` with paired `_min`/`_max` columns.

`iGAP` A `data.frame` with comma-separated `"min,max"` character values.

`ARRAY` A three-dimensional numeric `array` of dimension `[n, p, 2]` where `[,,1]` stores minima and `[,,2]` stores maxima.

`SODAS` `NULL` unless the input is a SODAS XML file path, in which case it stores the original path.

`SDS` `NULL` unless the input is a SODAS/SDS XML file path (alias for SODAS).

## Author(s)

Han-Ming Wu

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley.

Rodriguez, O. (2024). *RSDA: R to Symbolic Data Analysis*. R package, https://CRAN.R-project.org/package=RSDA.

Correia, M. (2009). *Interval GARCH and Aggregation of Predictions*.

Diday, E. and Noirhomme-Fraiture, M. (2008). *Symbolic Data Analysis and the SODAS Software*. Wiley.

## See Also

int_detect_format, int_convert_format, int_list_conversions

## Examples

```
data(car.int)
result <- to_all_interval_formats(car.int)
names(result)

# RSDA format (symbolic_tbl)
result$RSDA

# MM format (data.frame with _min/_max columns)
head(result$MM)

# iGAP format (data.frame with comma-separated values)
head(result$iGAP)

# ARRAY format (3D array)
dim(result$ARRAY)
result$ARRAY[1:3, , 1]  # minima
result$ARRAY[1:3, , 2]  # maxima

# SODAS/SDS slots are NULL (file-based format)
result$SODAS
result$SDS
```

---

# trivial_intervals.int *Trivial and Non-Trivial Intervals Example Dataset*

---

## Description

Simple 5x3 example illustrating different interval types: full intervals (hyperrectangles), degenerate intervals (lines), and trivial intervals (points). Used for vertices PCA demonstration.

## Usage

```
data(trivial_intervals.int)
```

## Format

A data frame with 5 observations (w1–w5) and 6 columns (3 interval variables in _l/_u Min-Max pairs):

- `y1_l`, `y1_u`: First interval variable.
- `y2_l`, `y2_u`: Second interval variable.
- `y3_l`, `y3_u`: Third interval variable.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 5 |
| **Variables (p)** | 6 |
| **Subject area** | Methodology |
| **Symbolic format** | Interval |
| **Analytical tasks** | PCA |

### Source

Billard and Diday (2006), Table 5.1, p. 146.

### References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley. Table 5.1, p.146.

### Examples

```
data(trivial_intervals.int)
```

---

## `uscrime.int` *US Crime Statistics Interval Dataset*

---

### Description

Interval-valued crime statistics for 46 US states, containing 102 interval-valued variables covering various crime types and rates. Originally from the RSDA package.

### Usage

```
data(uscrime.int)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 46 observations and 102 interval-valued variables. Key variables include:

- `fold`: Cross-validation fold assignment.
- `population`: Population range.
- `householdsize`: Household size range.

- `racepctblack`, `racePctWhite`, `racePctAsian`, `racePctHisp`: Race percentage ranges.
- `medIncome`, `medFamInc`, `perCapInc`: Income ranges.
- `PctUnemployed`, `PctEmploy`: Employment percentage ranges.
- `ViolentCrimesPerPop`: Violent crimes per population range.

Plus 90 additional interval-valued socio-economic and demographic variables.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 46 |
| **Variables (p)** | 102 |
| **Subject area** | Criminology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Regression, Clustering |

## Source

Extracted from `RSDA::uscrime_intv2`.

## References

Rodriguez, O. (2013). A generalization of Centre and Range method for fitting a linear regression model to symbolic interval data. In the proceedings of the 14th Conference of the International Federation of Classification Societies (IFCS 2013).

## Examples

```
data(uscrime.int)
```

---

# utsnow.int *Utah Snow Load Interval Dataset*

---

## Description

Interval-valued ground snow load data from 415 weather stations in Utah and surrounding states. Each observation is a station with a 50-year ground snow load interval (lower and upper bounds of the prediction interval in kPa) plus the point estimate, geographic coordinates, and elevation.

## Usage

```
data(utsnow.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 415 observations and 5 variables:

- `snow_load`: Interval-valued 50-year ground snow load (kPa).
- `point_estimate`: Numeric point estimate (kPa).
- `latitude`: Numeric latitude (degrees).
- `longitude`: Numeric longitude (degrees).
- `elevation`: Numeric elevation (meters).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 415 |
| **Variables (p)** | 5 |
| **Subject area** | Climate |
| **Symbolic format** | Interval |
| **Analytical tasks** | Regression, Spatial analysis |

## Source

`intkrige::utsnow`; the upstream documentation identifies the 415 locations as data from the 2018 Utah Snow Load Study.

## References

Bean, B., Sun, Y. and Maguire, M. (2022). Interval-valued kriging for geostatistical mapping with imprecise inputs. *International Journal of Approximate Reasoning*, 140, 31–51. doi:10.1016/j.ijar.2021.10.003.

Bean, B., Maguire, M. and Sun, Y. (2018). *Utah Snow Load Study*.

## Examples

```
data(utsnow.int)
```

---

`veterinary.int` *Veterinary Interval Dataset*

---

## Description

Interval-valued veterinary dataset of 10 animal specimens described by height and weight ranges. Includes male and female specimens of horses, bears, foxes, cats, and dogs.

## Usage

```
data(veterinary.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 10 observations and 3 variables:

- `Animal`: Animal type and sex label (e.g., HorseM, BearF; character).
- `Height`: Height range (cm, interval).
- `Weight`: Weight range (kg, interval).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 3 |
| **Subject area** | Zoology |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics, Clustering |

## Source

Distributed as `RSDA::VeterinaryData`; the RSDA documentation attributes the data to Billard and Diday (2006).

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis*. Wiley.

## Examples

```
data(veterinary.int)
```

---

# video1.int *Video Platform User Engagement Intervals (Dataset 1)*

---

## Description

Simulated interval-valued engagement metrics for 10 user groups on a video platform. Variables represent ranges of visit, watch, like, comment, and share counts.

## Usage

```
data(video1.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 10 observations and 5 interval-valued variables (V1–V5): number of visits, watches, likes, comments, and shares.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 5 |
| **Subject area** | Digital media |
| **Symbolic format** | Interval |
| **Analytical tasks** | PCA |

### Source

Simulated data distributed as `GPCSIV::video1`.

### References

Makosso-Kallyth, S. (2013). Analysis of m sets of symbolic interval variables. *Revue des Nouvelles Technologies de l'Information*, RNTI-E-25, 97–108.

### Examples

```
data(video1.int)
```

---

## `video2.int` *Video Platform User Engagement Intervals (Dataset 2)*

---

### Description

Simulated interval-valued engagement metrics for 10 user groups on a video platform. Variables represent ranges of visit, watch, like, comment, and share counts.

### Usage

```
data(video2.int)
```

### Format

A symbolic data frame (`symbolic_tbl`) with 10 observations and 5 interval-valued variables (V1–V5): number of visits, watches, likes, comments, and shares.

### Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 5 |
| **Subject area** | Digital media |
| **Symbolic format** | Interval |
| **Analytical tasks** | PCA |

## Source

Simulated data distributed as `GPCSIV::video2`.

## References

Makosso-Kallyth, S. (2013). Analysis of m sets of symbolic interval variables. *Revue des Nouvelles Technologies de l'Information*, RNTI-E-25, 97–108.

## Examples

```
data(video2.int)
```

---

# video3.int *Video Platform User Engagement Intervals (Dataset 3)*

---

## Description

Simulated interval-valued engagement metrics for 10 user groups on a video platform. Variables represent ranges of visit, watch, like, comment, and share counts.

## Usage

```
data(video3.int)
```

## Format

A symbolic data frame (`symbolic_tbl`) with 10 observations and 5 interval-valued variables (V1–V5): number of visits, watches, likes, comments, and shares.

## Metadata

| | |
|---|---|
| **Sample size (n)** | 10 |
| **Variables (p)** | 5 |
| **Subject area** | Digital media |
| **Symbolic format** | Interval |
| **Analytical tasks** | PCA |

## Source

Simulated data distributed as `GPCSIV::video3`.

## References

Makosso-Kallyth, S. (2013). Analysis of m sets of symbolic interval variables. *Revue des Nouvelles Technologies de l'Information*, RNTI-E-25, 97–108.

## Examples

```
data(video3.int)
```

---

`water_flow.int` *Water Flow Sensor Readings Interval Dataset*

---

## Description

Large interval-valued dataset of water flow sensor readings with 316 observations and 47 interval-valued feature variables (IF1-IF48, excluding IF17), classified into 2 groups. Used as a benchmark for interval data clustering with high-dimensional features.

## Usage

```
data(water_flow.int)
```

## Format

A data frame with 316 observations and 48 variables:

- `if1` through `if48` (excluding `if17`): 47 interval-valued sensor feature measurements.
- `class`: Group label (1 or 2).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 316 |
| **Variables (p)** | 48 |
| **Subject area** | Engineering |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

https://github.com/Natandradesa/Kernel-Clustering-for-Interval-Data

## References

Andrade de Sa, J. N., Ferreira, M. R. P. and de Carvalho, F. A. T. (2025). Kernel clustering with automatic variable weighting for interval data. *Neurocomputing*, 650, 130849. doi:10.1016/j.neucom.2025.130849.

## Examples

```
data(water_flow.int)
```

`weight_age.hist` *Weight by Age Group Histogram-Valued Dataset*

## Description

Histogram-valued weight distributions for 7 age groups (20s through 80s). Each observation represents an age decade with a 7-bin histogram of weight values (pounds).

## Usage

```
data(weight_age.hist)
```

## Format

A data frame with 7 observations and 1 histogram-valued variable:

- `weight`: Histogram-valued weight distribution (pounds).

Row names indicate age groups (20s, 30s, 40s, 50s, 60s, 70s, 80s).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 7 |
| **Variables (p)** | 1 |
| **Subject area** | Medical |
| **Symbolic format** | Histogram |
| **Analytical tasks** | Descriptive statistics |

## Source

Billard, L. and Diday, E. (2006), Table 3.10.

## References

Billard, L. and Diday, E. (2006). *Symbolic Data Analysis: Conceptual Statistics and Data Mining*. Wiley, Chichester. Table 3.10.

## Examples

```
data(weight_age.hist)
```

`wine.int` *Wine Chemical Properties Interval Dataset*

## Description

Interval-valued chemical and physical properties of 33 wine samples classified into 2 groups. Contains 9 interval-valued measurement variables. Used as a benchmark for interval data clustering algorithms.

## Usage

```
data(wine.int)
```

## Format

A data frame with 33 observations and 10 variables:

- `V1` through `V9`: Nine interval-valued chemical/physical property measurements.
- `class`: Wine group (1 or 2).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 33 |
| **Variables (p)** | 10 |
| **Subject area** | Food science |
| **Symbolic format** | Interval |
| **Analytical tasks** | Clustering |

## Source

https://github.com/Natandradesa/Kernel-Clustering-for-Interval-Data

## References

Andrade de Sa, J. N., Ferreira, M. R. P. and de Carvalho, F. A. T. (2025). Kernel clustering with automatic variable weighting for interval data. *Neurocomputing*, 650, 130849. doi:10.1016/j.neucom.2025.130849.

## Examples

```
data(wine.int)
```

`world_cup.int` *World Cup Soccer Teams Interval Dataset*

## Description

Interval-valued data for soccer teams grouped by World Cup qualification status (yes/no). Includes age, weight, height ranges and the covariance between weight and height.

## Usage

```
data(world_cup.int)
```

## Format

A data frame with 2 observations and 8 variables:

- `world_cup`: Qualification status (yes/no, character).
- `age_l`, `age_u`: Player age range (years).
- `weight_l`, `weight_u`: Player weight range (kg).
- `height_l`, `height_u`: Player height range (meters).
- `cov_weight_height`: Covariance between weight and height (numeric).

## Metadata

| | |
|---|---|
| **Sample size (n)** | 2 |
| **Variables (p)** | 8 |
| **Subject area** | Sports |
| **Symbolic format** | Interval |
| **Analytical tasks** | Descriptive statistics |

## Source

Diday and Noirhomme-Fraiture (2008), Table 1.9, p. 13.

## References

Diday, E. and Noirhomme-Fraiture, M. (Eds.) (2008). *Symbolic Data Analysis and the SODAS Software*. Wiley. Table 1.9, p.13.

## Examples

```
data(world_cup.int)
```

`write_symbolic_csv` *Write Symbolic Data to a CSV File*

## Description

Writes a symbolic data object (interval, histogram, modal, or any data frame) to a CSV file. Interval data stored in RSDA format (`symbolic_tbl` with complex columns) is automatically converted to MM format (paired `_min`/`_max` columns) before writing.

## Usage

```
write_symbolic_csv(
  x,
  file,
  sep = ",",
  row.names = TRUE,
  na = "NA",
  quote = TRUE,
  ...
)
```

## Arguments

| | |
|---|---|
| `x` | A `data.frame`, `symbolic_tbl`, or other tabular object containing symbolic data. |
| `file` | Path to the output CSV file. |
| `sep` | Field separator character. Default `","`. |
| `row.names` | Logical or character. If `TRUE` (the default), row names are written as the first column. |
| `na` | Character string to use for missing values. Default `"NA"`. |
| `quote` | Logical; should character and factor columns be quoted? Default `TRUE`. |
| `...` | Additional arguments passed to `write.table`. |

## Details

`write_symbolic_csv` handles every tabular symbolic type stored in **dataSDA**:

- **Interval (RSDA)**: `symbolic_tbl` objects with complex interval columns are converted to MM format before writing.
- **Interval (MM)**: Data frames with `_min`/`_max` columns are written directly.
- **Histogram / Modal / Other**: Plain data frames are written directly.

The output is a standard CSV that can be read back with `read_symbolic_csv`.

## Value

Invisibly returns the data frame that was written (after any conversion).

## See Also

read_symbolic_csv

## Examples

```
# Interval data (RSDA symbolic_tbl)
data(mushroom.int)
tmp <- tempfile(fileext = ".csv")
write_symbolic_csv(mushroom.int, tmp)
cat(readLines(tmp, n = 3), sep = "\n")

# Histogram data
data(airline_flights.hist)
tmp2 <- tempfile(fileext = ".csv")
write_symbolic_csv(airline_flights.hist, tmp2)
cat(readLines(tmp2, n = 3), sep = "\n")
```

# Index


∗ **PCA**
finance.int, 54
teams.int, 147
trivial_intervals.int, 153
∗ **agriculture**
french_agriculture.hist, 56
∗ **biology**
ecoli_routes.int, 45
∗ **categorical**
bird_species.mix, 22
bird_species_extended.mix, 23
∗ **classification**
cars.int, 28
∗ **climate**
china_temp_monthly.int, 35
∗ **clustering**
bats.int, 19
car_models.int, 29
china_temp.int, 34
freshwater_fish.int, 57
fungi.int, 59
horses.int, 76
iris.int, 98
oils.int, 125
synthetic_clusters.int, 146
temperature_city.int, 148
tennis.int, 149
water_flow.int, 160
wine.int, 162
∗ **commodities**
crude_oil_wti.its, 43
∗ **crime**
uscrime.int, 154
∗ **datasets**
abalone.iGAP, 5
abalone.int, 6
acid_rain.int, 7
age_cholesterol_weight.int, 8
age_pyramids.hist, 9
airline_flights.hist, 13
airline_flights2.modal, 14
bank_rates, 17
baseball.int, 18
bats.int, 19
bird.mix, 20
bird_color_taxonomy.hist, 21
bird_species.mix, 22
bird_species_extended.mix, 23
blood.hist, 24
blood_pressure.int, 25
car.int, 26
car_models.int, 29
cardiological.int, 27
cars.int, 28
census.mix, 30
china_climate_month.hist, 32
china_climate_season.hist, 33
china_temp.int, 34
china_temp_monthly.int, 35
cholesterol.hist, 36
county_income_gender.hist, 37
cover_types.hist, 38
credit_card.int, 39
crime.modal, 41
crime2.modal, 42
crude_oil_wti.its, 43
djia.its, 44
ecoli_routes.int, 45
employment.int, 46
energy_consumption.distr, 47
energy_usage.distr, 48
environment.mix, 49
euro_usd.its, 51
exchange_rate_returns.hist, 52
face.iGAP, 53
finance.int, 54
flights_detail.hist, 55
french_agriculture.hist, 56

freshwater_fish.int, 57
fuel_consumption.modal, 58
fungi.int, 59
genome_abundances.int, 60
glucose.hist, 61
hardwood.hist, 62
hdi_gender.int, 64
health_insurance.mix, 65
health_insurance2.modal, 66
hematocrit.hist, 67
hematocrit_hemoglobin.hist, 68
hemoglobin.hist, 69
hierarchy, 70
hierarchy.hist, 71
hierarchy.int, 72
horses.int, 76
hospital.hist, 77
household_characteristics.distr, 78
ibovespa.its, 79
iris.int, 98
iris_species.hist, 101
irish_wind.its, 99
joggers.mix, 102
judge1.int, 103
judge2.int, 104
judge3.int, 105
lackinfo.int, 106
lisbon_air_quality.int, 107
loans_by_purpose.int, 108
loans_by_risk.int, 109
loans_by_risk_quantile.int, 110
lung_cancer.hist, 111
lynne1.int, 112
merval.its, 113
mtcars.mix, 116
mushroom.int, 117
mushroom.int.mm, 119
mushroom_fuzzy.mix, 120
nycflights.int, 121
occupations.modal, 122
occupations2.modal, 123
ohtemp.int, 124
oils.int, 125
ozone.hist, 126
petrobras.its, 127
polish_cars.mix, 129
polish_voivodships.int, 130
profession.int, 131
prostate.int, 132
shanghai_stock.its, 139
simulated.hist, 140
soccer_bivar.int, 141
sp500.its, 144
state_income.hist, 145
synthetic_clusters.int, 146
teams.int, 147
temperature_city.int, 148
tennis.int, 149
town_services.mix, 150
trivial_intervals.int, 153
uscrime.int, 154
utsnow.int, 155
veterinary.int, 156
video1.int, 157
video2.int, 158
video3.int, 159
water_flow.int, 160
weight_age.hist, 161
wine.int, 162
world_cup.int, 163

∗ **demographics**
age_pyramids.hist, 9

∗ **discriminant**
employment.int, 46

∗ **distance**
temperature_city.int, 148

∗ **distribution**
bird_color_taxonomy.hist, 21
census.mix, 30
energy_consumption.distr, 47
energy_usage.distr, 48
household_characteristics.distr, 78

∗ **economics**
french_agriculture.hist, 56

∗ **environment**
ozone.hist, 126

∗ **exchange**
euro_usd.its, 51

∗ **finance**
crude_oil_wti.its, 43
djia.its, 44
euro_usd.its, 51
ibovespa.its, 79
loans_by_risk.int, 109

merval.its, 113
petrobras.its, 127
shanghai_stock.its, 139
sp500.its, 144
* **flights**
flights_detail.hist, 55
* **forestry**
cover_types.hist, 38
* **fuzzy**
mushroom_fuzzy.mix, 120
* **gender**
county_income_gender.hist, 37
* **genomics**
genome_abundances.int, 60
* **hierarchical**
hierarchy, 70
* **histogram**
age_pyramids.hist, 9
airline_flights.hist, 13
bird_color_taxonomy.hist, 21
bird_species_extended.mix, 23
blood.hist, 24
census.mix, 30
china_climate_month.hist, 32
china_climate_season.hist, 33
cholesterol.hist, 36
county_income_gender.hist, 37
cover_types.hist, 38
exchange_rate_returns.hist, 52
flights_detail.hist, 55
french_agriculture.hist, 56
glucose.hist, 61
hardwood.hist, 62
hematocrit.hist, 67
hematocrit_hemoglobin.hist, 68
hemoglobin.hist, 69
hierarchy.hist, 71
hospital.hist, 77
iris_species.hist, 101
joggers.mix, 102
lung_cancer.hist, 111
ozone.hist, 126
simulated.hist, 140
state_income.hist, 145
weight_age.hist, 161
* **household**
household_characteristics.distr, 78
* **iGAP**
abalone.iGAP, 5
face.iGAP, 53
* **income**
county_income_gender.hist, 37
state_income.hist, 145
* **interval**
abalone.iGAP, 5
abalone.int, 6
acid_rain.int, 7
age_cholesterol_weight.int, 8
baseball.int, 18
bats.int, 19
bird.mix, 20
bird_species.mix, 22
bird_species_extended.mix, 23
blood_pressure.int, 25
car.int, 26
car_models.int, 29
cardiological.int, 27
cars.int, 28
census.mix, 30
china_temp.int, 34
china_temp_monthly.int, 35
credit_card.int, 39
crude_oil_wti.its, 43
djia.its, 44
ecoli_routes.int, 45
employment.int, 46
environment.mix, 49
euro_usd.its, 51
face.iGAP, 53
finance.int, 54
freshwater_fish.int, 57
fungi.int, 59
genome_abundances.int, 60
hdi_gender.int, 64
hierarchy.hist, 71
hierarchy.int, 72
horses.int, 76
ibovespa.its, 79
iris.int, 98
irish_wind.its, 99
joggers.mix, 102
judge1.int, 103
judge2.int, 104
judge3.int, 105
lackinfo.int, 106

lisbon_air_quality.int, 107
loans_by_purpose.int, 108
loans_by_risk.int, 109
loans_by_risk_quantile.int, 110
lynne1.int, 112
merval.its, 113
mtcars.mix, 116
mushroom.int, 117
mushroom.int.mm, 119
nycflights.int, 121
ohtemp.int, 124
oils.int, 125
petrobras.its, 127
polish_cars.mix, 129
polish_voivodships.int, 130
profession.int, 131
prostate.int, 132
shanghai_stock.its, 139
soccer_bivar.int, 141
sp500.its, 144
synthetic_clusters.int, 146
teams.int, 147
temperature_city.int, 148
tennis.int, 149
town_services.mix, 150
trivial_intervals.int, 153
uscrime.int, 154
utsnow.int, 155
veterinary.int, 156
video1.int, 157
video2.int, 158
video3.int, 159
water_flow.int, 160
wine.int, 162
world_cup.int, 163

∗ **iris**
iris_species.hist, 101

∗ **medical**
cholesterol.hist, 36
glucose.hist, 61
hematocrit.hist, 67
hematocrit_hemoglobin.hist, 68
hemoglobin.hist, 69
prostate.int, 132

∗ **meteorology**
irish_wind.its, 99

∗ **mixed**
bird_color_taxonomy.hist, 21
bird_species.mix, 22
bird_species_extended.mix, 23
census.mix, 30
environment.mix, 49
health_insurance.mix, 65
hierarchy.hist, 71
joggers.mix, 102
mtcars.mix, 116
polish_cars.mix, 129
town_services.mix, 150

∗ **modal**
airline_flights2.modal, 14
crime.modal, 41
crime2.modal, 42
environment.mix, 49
fuel_consumption.modal, 58
health_insurance2.modal, 66
mtcars.mix, 116
occupations.modal, 122
occupations2.modal, 123
town_services.mix, 150

∗ **model**
bank_rates, 17

∗ **multi-valued**
town_services.mix, 150

∗ **multinomial**
polish_cars.mix, 129

∗ **ordinal**
hdi_gender.int, 64

∗ **regression**
fuel_consumption.modal, 58
hematocrit_hemoglobin.hist, 68
soccer_bivar.int, 141

∗ **simulated**
simulated.hist, 140

∗ **socioeconomic**
polish_voivodships.int, 130

∗ **spatial**
irish_wind.its, 99

∗ **symbolic**
bank_rates, 17
health_insurance.mix, 65
mushroom_fuzzy.mix, 120

∗ **synthetic**
synthetic_clusters.int, 146

∗ **temperature**
china_temp_monthly.int, 35

∗ **timeseries**

crude_oil_wti.its, 43
djia.its, 44
euro_usd.its, 51
ibovespa.its, 79
irish_wind.its, 99
merval.its, 113
petrobras.its, 127
shanghai_stock.its, 139
sp500.its, 144
∗ **visualization**
bats.int, 19
∗ **weather**
ozone.hist, 126

abalone.iGAP, 5, *6*
abalone.int, *5*, 6
acid_rain.int, 7
age_cholesterol_weight.int, 8
age_pyramids.hist, 9
aggregate_to_symbolic, 10
airline_flights.hist, 13, *14*
airline_flights2.modal, *13*, 14
ARRAY_to_iGAP, 15
ARRAY_to_MM, 16
ARRAY_to_RSDA, 16

bank_rates, 17
baseball.int, 18
bats.int, 19
bird.mix, 20
bird_color_taxonomy.hist, 21
bird_species.mix, 22
bird_species_extended.mix, 23
blood.hist, 24
blood_pressure.int, 25

car.int, 26
car_models.int, 29
cardiological.int, 27
cars.int, 28
census.mix, 30
check_zero_width_intervals, *11*, 31
china_climate_month.hist, 32
china_climate_season.hist, 33
china_temp.int, 34
china_temp_monthly.int, 35
cholesterol.hist, 36
clean_colnames, 37
county_income_gender.hist, 37
cover_types.hist, 38
credit_card.int, 39
crime.modal, 41, *42*
crime2.modal, *41*, 42
crude_oil_wti.its, 43

djia.its, 44

ecoli_routes.int, 45
employment.int, 46
energy_consumption.distr, 47
energy_usage.distr, 48
environment.mix, 49
euro_usd.its, 51
exchange_rate_returns.hist, 52

face.iGAP, 53
finance.int, 54
flights_detail.hist, 55
french_agriculture.hist, 56
freshwater_fish.int, 57
fuel_consumption.modal, 58
fungi.int, 59

genome_abundances.int, 60
glucose.hist, 61

hardwood.hist, 62
hdi_gender.int, 64
health_insurance.mix, 65, *66*
health_insurance2.modal, *65*, 66
hematocrit.hist, 67
hematocrit_hemoglobin.hist, 68
hemoglobin.hist, 69
hierarchy, 70, *72*
hierarchy.hist, 71
hierarchy.int, *70*, 72
hist_cor *(*histogram_stats*)*, 73
hist_cov *(*histogram_stats*)*, 73
hist_extract, 75
hist_mean *(*histogram_stats*)*, 73
hist_var *(*histogram_stats*)*, 73
histogram_stats, 73
horses.int, 76
hospital.hist, 77
household_characteristics.distr, 78

ibovespa.its, 79
iGAP_to_ARRAY, 80
iGAP_to_MM, 81

iGAP_to_RSDA, 82
int_center (interval_geometry), 84
int_containment (interval_geometry), 84
int_convert_format, 95, *134*, *153*
int_cor (interval_stats), 92
int_cosine (interval_similarity), 91
int_cov (interval_stats), 92
int_cv (interval_uncertainty), 94
int_detect_format, 97, *134*, *153*
int_dice (interval_similarity), 91
int_dispersion (interval_uncertainty), 94
int_dist (interval_distance), 82
int_dist_all (interval_distance), 82
int_dist_matrix (interval_distance), 82
int_entropy (interval_uncertainty), 94
int_granularity (interval_uncertainty), 94
int_imprecision (interval_uncertainty), 94
int_information_content (interval_uncertainty), 94
int_iqr (interval_position), 86
int_jaccard (interval_similarity), 91
int_kurtosis (interval_shape), 89
int_list_conversions, 98, *153*
int_mad (interval_position), 86
int_mean (interval_stats), 92
int_median (interval_position), 86
int_midrange (interval_geometry), 84
int_mode (interval_position), 86
int_overlap (interval_geometry), 84
int_overlap_coefficient (interval_similarity), 91
int_pairwise_dist (interval_distance), 82
int_quantile (interval_position), 86
int_radius (interval_geometry), 84
int_range (interval_position), 86
int_similarity_matrix (interval_similarity), 91
int_skewness (interval_shape), 89
int_symmetry (interval_shape), 89
int_tailedness (interval_shape), 89
int_tanimoto (interval_similarity), 91
int_trimmed_mean (interval_robust), 88
int_trimmed_var (interval_robust), 88
int_uniformity (interval_uncertainty), 94
int_var (interval_stats), 92
int_width (interval_geometry), 84
int_winsorized_mean (interval_robust), 88
int_winsorized_var (interval_robust), 88
interval_distance, 82
interval_geometry, 84
interval_position, 86
interval_robust, 88
interval_shape, 89
interval_similarity, 91
interval_stats, 92
interval_uncertainty, 94
iris.int, 98
iris_species.hist, 101
irish_wind.its, 99

joggers.mix, 102
judge1.int, 103
judge2.int, 104
judge3.int, 105

lackinfo.int, 106
lisbon_air_quality.int, 107
loans_by_purpose.int, 108
loans_by_risk.int, 109
loans_by_risk_quantile.int, 110
lung_cancer.hist, 111
lynne1.int, 112

MatH, *12*
merval.its, 113
MM_to_ARRAY, 114
MM_to_iGAP, 115
MM_to_RSDA, 116
mtcars.mix, 116
mushroom.int, 117
mushroom.int.mm, *117*, *118*, 119
mushroom_fuzzy.mix, 120

nycflights.int, 121

occupations.modal, 122, *123*
occupations2.modal, *122*, 123
ohtemp.int, 124
oils.int, 125
ozone.hist, 126

petrobras.its, 127

polish_cars.mix, 129
polish_voivodships.int, 130
profession.int, 131
prostate.int, 132

read.table, *133*
read_symbolic_csv, 133, *164, 165*
RSDA_format, 134
RSDA_to_ARRAY, 135
RSDA_to_iGAP, 136
RSDA_to_MM, 136

search_data, 137
set_variable_format, 138
shanghai_stock.its, 139
simulated.hist, 140
soccer_bivar.int, 141
SODAS_to_ARRAY, 142
SODAS_to_iGAP, 143
SODAS_to_MM, 143
sp500.its, 144
state_income.hist, 145
synthetic_clusters.int, 146

teams.int, 147
temperature_city.int, 148
tennis.int, 149
to_all_interval_formats, 151
town_services.mix, 150
trivial_intervals.int, 153

uscrime.int, 154
utsnow.int, 155

veterinary.int, 156
video1.int, 157
video2.int, 158
video3.int, 159

water_flow.int, 160
weight_age.hist, 161
wine.int, 162
world_cup.int, 163
write.table, *164*
write_symbolic_csv, *134*, 164