Package 'radiant.multivariate'

March 10, 2025

Type Package

Title Multivariate Menu for Radiant: Business Analytics using R and Shiny

Version 1.6.7

Date 2025-3-9

Description The Radiant Multivariate menu includes interfaces for perceptual mapping, factor analysis, cluster analysis, and conjoint analysis. The application extends the functionality in 'radiant.data'.

Depends R (>= 4.3.0), radiant.data (>= 1.6.6)

Imports radiant.model (>= 1.6.6), shiny (>= 1.8.1), dplyr (>= 1.0.7), rlang (>= 0.4.10), ggplot2 (>= 2.2.1), scales (>= 0.4.0), magrittr (>= 1.5), psych (>= 1.8.4), GPArotation (>= 2014.11-1), car (>= 2.1.1), MASS (>= 7.3), import (>= 1.1.0), ggrepel (>= 0.8), lubridate (>= 1.7.4), polycor (>= 0.7.10), gower (>= 0.2.1), clustMixType (>= 0.2.1), patchwork (>= 1.0.0)

Suggests testthat (>= 2.0.0), pkgdown (>= 1.1.0)

URL https://github.com/radiant-rstats/radiant.multivariate/, https://radiant-rstats.github.io/radiant.multivariate/, https://radiant-rstats.github.io/docs/

BugReports https://github.com/radiant-rstats/radiant.multivariate/issues/

License AGPL-3 | file LICENSE

LazyData true

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Vincent Nijs [aut, cre]

Maintainer Vincent Nijs <radiant@rady.ucsd.edu>

Repository CRAN

Date/Publication 2025-03-10 18:40:02 UTC

Contents

carpet *Carpet cleaners*

Description

Carpet cleaners

Usage

```
data(carpet)
```

Format

A data frame with 18 rows and 5 variables

Details

Rankings reflect the evaluation of 18 alternative carpet cleaners by one respondent. Description provided in attr(carpet," description")

city *City distances*

Description

City distances

Usage

```
data(city)
```

Format

A data frame with 45 rows and 3 variables

Details

Distance in miles between nine cities in the USA. The dataset is used to illustrate multi-dimensional scaling (MDS). Description provided in attr(city, "description")

city2 *City distances 2*

Description

City distances 2

Usage

```
data(city2)
```

Format

A data frame with 78 rows and 3 variables

Details

Distance in miles between 12 cities in the USA. The dataset is used to illustrate multi-dimensional scaling (MDS). Description provided in attr(city2, "description")

clean_loadings *Sort and clean loadings*

Description

Sort and clean loadings

Usage

```
clean_loadings(floadings, cutoff = 0, fsort = FALSE, dec = 8, repl = NA)
```

Arguments

floadings	Data frame with loadings
cutoff	Show only loadings with (absolute) values above cutoff (default = 0)
fsort	Sort factor loadings
dec	Number of decimals to show
repl	Replace loadings below the cutoff by NA (or "")

Details

See https://radiant-rstats.github.io/docs/multivariate/full_factor.html for an example in Radiant

Examples

```
result <- full_factor(shopping, "v1:v6", nr_fact = 2)
clean_loadings(result$floadings, fsort = TRUE, cutoff = .5, dec = 2)
```

---

computer *Perceptions of computer (re)sellers*

Description

Perceptions of computer (re)sellers

Usage

```
data(computer)
```

Format

A data frame with 5 rows and 8 variables

Details

Perceptions of computer (re)sellers. The dataset is used to illustrate perceptual maps. Description provided in attr(computer, "description")

---

conjoint *Conjoint analysis*

Description

Conjoint analysis

Usage

```
conjoint(
  dataset,
  rvar,
  evar,
  int = "",
  by = "none",
  reverse = FALSE,
  data_filter = "",
  envir = parent.frame()
)
```

Arguments

dataset	Dataset
rvar	The response variable (e.g., profile ratings)
evar	Explanatory variables in the regression
int	Interaction terms to include in the model
by	Variable to group data by before analysis (e.g., a respondent id)
reverse	Reverse the values of the response variable (‘rvar‘)
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")
envir	Environment to extract data from

Details

See https://radiant-rstats.github.io/docs/multivariate/conjoint.html for an example in Radiant

Value

A list with all variables defined in the function as an object of class conjoint

See Also

summary.conjoint to summarize results

plot.conjoint to plot results

Examples

```
conjoint(mp3, rvar = "Rating", evar = "Memory:Shape") %>% str()
```

---

full_factor *Factor analysis (PCA)*

Description

Factor analysis (PCA)

Usage

```
full_factor(
  dataset,
  vars,
  method = "PCA",
  hcor = FALSE,
  nr_fact = 1,
  rotation = "varimax",
  data_filter = "",
  envir = parent.frame()
)
```

Arguments

dataset	Dataset
vars	Variables to include in the analysis
method	Factor extraction method to use
hcor	Use polycor::hetcor to calculate the correlation matrix
nr_fact	Number of factors to extract
rotation	Apply varimax rotation or no rotation ("varimax" or "none")
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")
envir	Environment to extract data from

Details

See https://radiant-rstats.github.io/docs/multivariate/full_factor.html for an example in Radiant

Value

A list with all variables defined in the function as an object of class full_factor

See Also

summary.full_factor to summarize results

plot.full_factor to plot results

Examples

```
full_factor(shopping, "v1:v6") %>% str()
```

`hclus` *Hierarchical cluster analysis*

Description

Hierarchical cluster analysis

Usage

```
hclus(
  dataset,
  vars,
  labels = "none",
  distance = "sq.euclidian",
  method = "ward.D",
  max_cases = 5000,
  standardize = TRUE,
  data_filter = "",
  envir = parent.frame()
)
```

Arguments

`dataset`	Dataset
`vars`	Vector of variables to include in the analysis
`labels`	A vector of labels for the leaves of the tree
`distance`	Distance
`method`	Method
`max_cases`	Maximum number of cases allowed (default is 1000). Set to avoid long-running analysis in the radiant web-interface
`standardize`	Standardized data (TRUE or FALSE)
`data_filter`	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")
`envir`	Environment to extract data from

Details

See https://radiant-rstats.github.io/docs/multivariate/hclus.html for an example in Radiant

Value

A list of all variables used in hclus as an object of class hclus

See Also

summary.hclus to summarize results

plot.hclus to plot results

Examples

```
hclus(shopping, vars = "v1:v6") %>% str()
```

---

kclus *K-clustering*

Description

K-clustering

Usage

```
kclus(
  dataset,
  vars,
  fun = "kmeans",
  hc_init = TRUE,
  distance = "sq.euclidian",
  method = "ward.D",
  seed = 1234,
  nr_clus = 2,
  standardize = TRUE,
  lambda = NULL,
  data_filter = "",
  envir = parent.frame()
)
```

Arguments

dataset	Dataset
vars	Vector of variables to include in the analysis
fun	Use either "kmeans" or "kproto" for clustering
hc_init	Use centers from hclus as the starting point
distance	Distance for hclus
method	Method for hclus
seed	Random see to use for k-clustering if hc_init is FALSE
nr_clus	Number of clusters to extract
standardize	Standardize data (TRUE or FALSE)

lambda	Parameter > 0 to trade off between Euclidean distance of numeric variables and simple matching coefficient between categorical variables. Also a vector of variable specific factors is possible where the order must correspond to the order of the variables in the data. In this case all variables' distances will be multiplied by their corresponding lambda value.
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")
envir	Environment to extract data from

Details

See https://radiant-rstats.github.io/docs/multivariate/kclus.html for an example in Radiant

Value

A list of all variables used in kclus as an object of class kclus

See Also

summary.kclus to summarize results

plot.kclus to plot results

store.kclus to add cluster membership to the selected dataset

Examples

```
kclus(shopping, c("v1:v6"), nr_clus = 3) %>% str()
```

---

mds *(Dis)similarity based brand maps (MDS)*

Description

(Dis)similarity based brand maps (MDS)

Usage

```
mds(
  dataset,
  id1,
  id2,
  dis,
  method = "metric",
  nr_dim = 2,
  seed = 1234,
  data_filter = "",
  envir = parent.frame()
)
```

Arguments

dataset	Dataset
id1	A character variable or factor with unique entries
id2	A character variable or factor with unique entries
dis	A numeric measure of brand dissimilarity
method	Apply metric or non-metric MDS
nr_dim	Number of dimensions
seed	Random seed
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")
envir	Environment to extract data from

Details

See https://radiant-rstats.github.io/docs/multivariate/mds.html for an example in Radiant

Value

A list of all variables defined in the function as an object of class mds

See Also

summary.mds to summarize results

plot.mds to plot results

Examples

```
mds(city, "from", "to", "distance") %>% str()
mds(diamonds, "clarity", "cut", "price") %>% str()
```

---

movie *Conjoint data for Movie theaters*

Description

Conjoint data for Movie theaters

Usage

```
data(movie)
```

Format

A data frame with 18 rows and 6 variables

Details

Rankings reflect the evaluation of 18 alternative movie theaters by one respondent. Description provided in attr(movie, "description")

---

mp3 *Conjoint data for MP3 players*

Description

Conjoint data for MP3 players

Usage

```
data(mp3)
```

Format

A data frame with 18 rows and 6 variables

Details

Ratings reflect the evaluation of 18 alternative MP3 players by one respondent. Description provided in attr(mp3, "description")

---

plot.conjoint *Plot method for the conjoint function*

Description

Plot method for the conjoint function

Usage

```
## S3 method for class 'conjoint'
plot(
  x,
  plots = "pw",
  show = "",
  scale_plot = FALSE,
  shiny = FALSE,
  custom = FALSE,
  ...
)
```

Arguments

x	Return value from conjoint
plots	Show either the part-worth ("pw") or importance-weights ("iw") plot
show	Level in by variable to analyze (e.g., a specific respondent)
scale_plot	Scale the axes of the part-worth plots to the same range
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and https://ggplot2.tidyverse.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/conjoint.html for an example in Radiant

See Also

conjoint to generate results

summary.conjoint to summarize results

Examples

```
result <- conjoint(mp3, rvar = "Rating", evar = "Memory:Shape")
plot(result, scale_plot = TRUE)
plot(result, plots = "iw")
```

---

plot.full_factor *Plot method for the full_factor function*

Description

Plot method for the full_factor function

Usage

```
## S3 method for class 'full_factor'
plot(x, plots = "attr", shiny = FALSE, custom = FALSE, ...)
```

Arguments

x	Return value from full_factor
plots	Include attribute ("attr"), respondents ("resp") or both in the plot
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and https://ggplot2.tidyverse.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/full_factor.html for an example in Radiant

See Also

full_factor to calculate results

plot.full_factor to plot results

Examples

```
result <- full_factor(shopping, "v1:v6", nr_fact = 2)
plot(result)
```

---

plot.hclus *Plot method for the hclus function*

Description

Plot method for the hclus function

Usage

```
## S3 method for class 'hclus'
plot(
  x,
  plots = c("scree", "change"),
  cutoff = 0.05,
  shiny = FALSE,
  custom = FALSE,
  ...
)
```

Arguments

x	Return value from hclus
plots	Plots to return. "change" shows the percentage change in within-cluster heterogeneity as respondents are grouped into different number of clusters, "dendro" shows the dendrogram, "scree" shows a scree plot of within-cluster heterogeneity
cutoff	For large datasets plots can take time to render and become hard to interpret. By selection a cutoff point (e.g., 0.05 percent) the initial steps in hierarchical cluster analysis are removed from the plot
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and https://ggplot2.tidyverse.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/hclus.html for an example in Radiant

See Also

hclus to generate results

summary.hclus to summarize results

Examples

```
result <- hclus(shopping, vars = c("v1:v6"))
plot(result, plots = c("change", "scree"), cutoff = .05)
plot(result, plots = "dendro", cutoff = 0)
```

---

plot.kclus *Plot method for kclus*

Description

Plot method for kclus

Usage

```
## S3 method for class 'kclus'
plot(x, plots = "density", shiny = FALSE, custom = FALSE, ...)
```

Arguments

x	Return value from kclus
plots	One of "density", "bar", or "scatter")
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and https://ggplot2.tidyverse.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/kclus.html for an example in Radiant

See Also

kclus to generate results

summary.kclus to summarize results

store.kclus to add cluster membership to the selected dataset

Examples

```
result <- kclus(shopping, vars = "v1:v6", nr_clus = 3)
plot(result)
```

---

plot.mds *Plot method for the mds function*

Description

Plot method for the mds function

Usage

```
## S3 method for class 'mds'
plot(x, rev_dim = NULL, fontsz = 5, shiny = FALSE, custom = FALSE, ...)
```

Arguments

x	Return value from mds
rev_dim	Flip the axes in plots
fontsz	Font size to use in plots
shiny	Did the function call originate inside a shiny app

custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and https://ggplot2.tidyverse.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/mds.html for an example in Radiant

See Also

mds to calculate results

summary.mds to plot results

Examples

```
result <- mds(city, "from", "to", "distance")
plot(result, fontsz = 7)
plot(result, rev_dim = 1:2)
```

---

plot.pre_factor *Plot method for the pre_factor function*

Description

Plot method for the pre_factor function

Usage

```
## S3 method for class 'pre_factor'
plot(
  x,
  plots = c("scree", "change"),
  cutoff = 0.2,
  shiny = FALSE,
  custom = FALSE,
  ...
)
```

Arguments

x	Return value from pre_factor
plots	Plots to return. "change" shows the change in eigenvalues as variables are grouped into different number of factors, "scree" shows a scree plot of eigenvalues
cutoff	For large datasets plots can take time to render and become hard to interpret. By selection a cutoff point (e.g., eigenvalues of .8 or higher) factors with the least explanatory power are removed from the plot
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and https://ggplot2.tidyverse.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/pre_factor.html for an example in Radiant

See Also

pre_factor to calculate results

summary.pre_factor to summarize results

Examples

```
result <- pre_factor(shopping, "v1:v6")
plot(result, plots = c("change", "scree"), cutoff = .05)
```

---

plot.prmap *Plot method for the prmap function*

Description

Plot method for the prmap function

Usage

```
## S3 method for class 'prmap'
plot(
  x,
  plots = "",
  scaling = 2,
  fontsz = 5,
```

```
  seed = 1234,
  shiny = FALSE,
  custom = FALSE,
  ...
)
```

Arguments

x	Return value from prmap
plots	Components to include in the plot ("brand", "attr"). If data on preferences is available use "pref" to add preference arrows to the plot
scaling	Arrow scaling in the brand map
fontsz	Font size to use in plots
seed	Random seed
shiny	Did the function call originate inside a shiny app
custom	Logical (TRUE, FALSE) to indicate if ggplot object (or list of ggplot objects) should be returned. This option can be used to customize plots (e.g., add a title, change x and y labels, etc.). See examples and https://ggplot2.tidyverse.org/ for options.
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/prmap.html for an example in Radiant

See Also

prmap to calculate results

summary.prmap to plot results

Examples

```
result <- prmap(computer, brand = "brand", attr = "high_end:business")
plot(result, plots = "brand")
plot(result, plots = c("brand", "attr"))
plot(result, scaling = 1, plots = c("brand", "attr"))
prmap(
  retailers,
  brand = "retailer",
  attr = "good_value:cluttered",
  pref = c("segment1", "segment2")
) %>% plot(plots = c("brand", "attr", "pref"))
```

predict.conjoint *Predict method for the conjoint function*

Description

Predict method for the conjoint function

Usage

```
## S3 method for class 'conjoint'
predict(
  object,
  pred_data = NULL,
  pred_cmd = "",
  conf_lev = 0.95,
  se = FALSE,
  interval = "confidence",
  dec = 3,
  envir = parent.frame(),
  ...
)
```

Arguments

object	Return value from conjoint
pred_data	Provide the dataframe to generate predictions. The dataset must contain all columns used in the estimation
pred_cmd	Command used to generate data for prediction
conf_lev	Confidence level used to estimate confidence intervals (.95 is the default)
se	Logical that indicates if prediction standard errors should be calculated (default = FALSE)
interval	Type of interval calculation ("confidence" or "prediction"). Set to "none" if se is FALSE
dec	Number of decimals to show
envir	Environment to extract data from
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/conjoint.html for an example in Radiant

See Also

conjoint to generate the result

summary.conjoint to summarize results

plot.conjoint to plot results

Examples

```
result <- conjoint(mp3, rvar = "Rating", evar = "Memory:Shape")
predict(result, pred_data = mp3)
```

---

predict_conjoint_by *Predict method for the conjoint function when a by variables is used*

Description

Predict method for the conjoint function when a by variables is used

Usage

```
predict_conjoint_by(
  object,
  pfun,
  pred_data = NULL,
  pred_cmd = "",
  conf_lev = 0.95,
  se = FALSE,
  dec = 3,
  envir = parent.frame(),
  ...
)
```

Arguments

object	Return value from conjoint
pfun	Function to use for prediction
pred_data	Name of the dataset to use for prediction
pred_cmd	Command used to generate data for prediction
conf_lev	Confidence level used to estimate confidence intervals (.95 is the default)
se	Logical that indicates if prediction standard errors should be calculated (default = FALSE)
dec	Number of decimals to show
envir	Environment to extract data from
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/conjoint.html for an example in Radiant

See Also

conjoint to generate the result

summary.conjoint to summarize results

plot.conjoint to plot results

---

pre_factor *Evaluate if data are appropriate for PCA / Factor analysis*

Description

Evaluate if data are appropriate for PCA / Factor analysis

Usage

```
pre_factor(
  dataset,
  vars,
  hcor = FALSE,
  data_filter = "",
  envir = parent.frame()
)
```

Arguments

dataset	Dataset
vars	Variables to include in the analysis
hcor	Use polycor::hetcor to calculate the correlation matrix
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")
envir	Environment to extract data from

Details

See https://radiant-rstats.github.io/docs/multivariate/pre_factor.html for an example in Radiant

Value

A list with all variables defined in the function as an object of class pre_factor

See Also

summary.pre_factor to summarize results

plot.pre_factor to plot results

Examples

```
pre_factor(shopping, "v1:v6") %>% str()
```

---

print.conjoint.predict *Print method for predict.conjoint*

Description

Print method for predict.conjoint

Usage

```
## S3 method for class 'conjoint.predict'
print(x, ..., n = 20)
```

Arguments

x	Return value from prediction method
...	further arguments passed to or from other methods
n	Number of lines of prediction results to print. Use -1 to print all lines

---

prmap *Attribute based brand maps*

Description

Attribute based brand maps

Usage

```
prmap(
  dataset,
  brand,
  attr,
  pref = "",
  nr_dim = 2,
  hcor = FALSE,
  data_filter = "",
  envir = parent.frame()
)
```

Arguments

dataset	Dataset
brand	A character variable with brand names
attr	Names of numeric variables
pref	Names of numeric brand preference measures
nr_dim	Number of dimensions
hcor	Use polycor::hetcor to calculate the correlation matrix
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")
envir	Environment to extract data from

Details

See https://radiant-rstats.github.io/docs/multivariate/prmap.html for an example in Radiant

Value

A list of all variables defined in the function as an object of class prmap

See Also

summary.prmap to summarize results

plot.prmap to plot results

Examples

```
prmap(computer, brand = "brand", attr = "high_end:business") %>% str()
```

---

radiant.multivariate *radiant.multivariate*

Description

Launch radiant.multivariate in the default web browser

Usage

```
radiant.multivariate(state, ...)
```

Arguments

state	Path to state file to load
...	additional arguments to pass to shiny::runApp (e.g, port = 8080)

Details

See https://radiant-rstats.github.io/docs/ for documentation and tutorials

Examples

```
## Not run:
radiant.multivariate()

## End(Not run)
```

`radiant.multivariate_viewer` *Launch radiant.multivariate in the Rstudio viewer*

Description

Launch radiant.multivariate in the Rstudio viewer

Usage

```
radiant.multivariate_viewer(state, ...)
```

Arguments

`state`	Path to state file to load
`...`	additional arguments to pass to shiny::runApp (e.g, port = 8080)

Details

See https://radiant-rstats.github.io/docs/ for documentation and tutorials

Examples

```
## Not run:
radiant.multivariate_viewer()

## End(Not run)
```

radiant.multivariate_window *Launch radiant.multivariate in an Rstudio window*

Description

Launch radiant.multivariate in an Rstudio window

Usage

```
radiant.multivariate_window(state, ...)
```

Arguments

state	Path to state file to load
...	additional arguments to pass to shiny::runApp (e.g, port = 8080)

Details

See https://radiant-rstats.github.io/docs/ for documentation and tutorials

Examples

```
## Not run:
radiant.multivariate_window()

## End(Not run)
```

retailers *Perceptions of retailers*

Description

Perceptions of retailers

Usage

```
data(retailers)
```

Format

A data frame with 6 rows and 10 variables

Details

Consumer evaluations for a set of retailers in the Chicago area on 7 attributes. The dataset is used to illustrate perceptual maps. Description provided in attr(retailers, "description")

shopping *Shopping attitudes*

Description

Shopping attitudes

Usage

```
data(shopping)
```

Format

A data frame with 20 rows and 7 variables

Details

Attitudinal data on shopping for 20 consumers. Description provided in attr(shopping, "description")

store.conjoint *Store method for the Multivariate > Conjoint tab*

Description

Store method for the Multivariate > Conjoint tab

Usage

```
## S3 method for class 'conjoint'
store(dataset, object, name, ...)
```

Arguments

dataset	Dataset
object	Return value from conjoint
name	Variable name(s) assigned to predicted values
...	further arguments passed to or from other methods

Details

Store data frame with PWs or IWs in Radiant r_data list if available

store.conjoint.predict *Store predicted values generated in predict.conjoint*

Description

Store predicted values generated in predict.conjoint

Usage

```
## S3 method for class 'conjoint.predict'
store(dataset, object, name = "prediction", ...)
```

Arguments

dataset	Dataset to add predictions to
object	Return value from model predict function
name	Variable name(s) assigned to predicted values
...	Additional arguments

Details

See https://radiant-rstats.github.io/docs/multivariate/conjoint.html for an example in Radiant

Examples

```
conjoint(mp3, rvar = "Rating", evar = "Memory:Shape") %>%
  predict(mp3) %>%
  store(mp3, ., name = "pred_pref")
```

store.full_factor *Store factor scores to active dataset*

Description

Store factor scores to active dataset

Usage

```
## S3 method for class 'full_factor'
store(dataset, object, name = "", ...)
```

Arguments

dataset	Dataset to append to factor scores to
object	Return value from full_factor
name	Name of factor score variables
...	Additional arguments

Details

See https://radiant-rstats.github.io/docs/multivariate/full_factor.html for an example in Radiant

See Also

full_factor to generate results

summary.full_factor to summarize results

plot.full_factor to plot results

Examples

```
full_factor(shopping, "v1:v6", nr_fact = 3) %>%
  store(shopping, .) %>%
  head()
```

---

store.hclus *Add a cluster membership variable to the active dataset*

Description

Add a cluster membership variable to the active dataset

Usage

```
## S3 method for class 'hclus'
store(dataset, object, nr_clus = 2, name = "", ...)
```

Arguments

dataset	Dataset to append to cluster membership variable to
object	Return value from hclus
nr_clus	Number of clusters to extract
name	Name of cluster membership variable
...	Additional arguments

Details

See https://radiant-rstats.github.io/docs/multivariate/hclus.html for an example in Radiant

See Also

hclus to generate results

summary.hclus to summarize results

plot.hclus to plot results

Examples

```
hclus(shopping, vars = "v1:v6") %>%
  store(shopping, ., nr_clus = 3) %>%
  head()
```

---

store.kclus *Add a cluster membership variable to the active dataset*

Description

Add a cluster membership variable to the active dataset

Usage

```
## S3 method for class 'kclus'
store(dataset, object, name = "", ...)
```

Arguments

dataset	Dataset to append to cluster membership variable to
object	Return value from kclus
name	Name of cluster membership variable
...	Additional arguments

Details

See https://radiant-rstats.github.io/docs/multivariate/kclus.html for an example in Radiant

See Also

kclus to generate results

summary.kclus to summarize results

plot.kclus to plot results

Examples

```
kclus(shopping, vars = "v1:v6", nr_clus = 3) %>%
  store(shopping, .) %>%
  head()
```

`summary.conjoint` *Summary method for the conjoint function*

Description

Summary method for the conjoint function

Usage

```
## S3 method for class 'conjoint'
summary(object, show = "", mc_diag = FALSE, additional = FALSE, dec = 3, ...)
```

Arguments

`object`	Return value from `conjoint`
`show`	Level in by variable to analyze (e.g., a specific respondent)
`mc_diag`	Shows multicollinearity diagnostics.
`additional`	Show additional regression results
`dec`	Number of decimals to show
`...`	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/conjoint.html for an example in Radiant

See Also

`conjoint` to generate results

`plot.conjoint` to plot results

Examples

```
result <- conjoint(mp3, rvar = "Rating", evar = "Memory:Shape")
summary(result, mc_diag = TRUE)
```

summary.full_factor *Summary method for the full_factor function*

Description

Summary method for the full_factor function

Usage

```
## S3 method for class 'full_factor'
summary(object, cutoff = 0, fsort = FALSE, dec = 2, ...)
```

Arguments

object	Return value from full_factor
cutoff	Show only loadings with (absolute) values above cutoff (default = 0)
fsort	Sort factor loadings
dec	Number of decimals to show
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/full_factor.html for an example in Radiant

See Also

full_factor to calculate results

plot.full_factor to plot results

Examples

```
result <- full_factor(shopping, "v1:v6", nr_fact = 2)
summary(result)
summary(result, cutoff = .5, fsort = TRUE)
```

summary.hclus *Summary method for the hclus function*

Description

Summary method for the hclus function

Usage

```
## S3 method for class 'hclus'
summary(object, ...)
```

Arguments

object	Return value from hclus
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/hclus.html for an example in Radiant

See Also

hclus to generate results

plot.hclus to plot results

Examples

```
result <- hclus(shopping, vars = c("v1:v6"))
summary(result)
```

summary.kclus *Summary method for kclus*

Description

Summary method for kclus

Usage

```
## S3 method for class 'kclus'
summary(object, dec = 2, ...)
```

Arguments

object	Return value from kclus
dec	Number of decimals to show
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/kclus.html for an example in Radiant

See Also

kclus to generate results

plot.kclus to plot results

store.kclus to add cluster membership to the selected dataset

Examples

```
result <- kclus(shopping, vars = "v1:v6", nr_clus = 3)
summary(result)
```

---

summary.mds *Summary method for the mds function*

Description

Summary method for the mds function

Usage

```
## S3 method for class 'mds'
summary(object, dec = 2, ...)
```

Arguments

object	Return value from mds
dec	Rounding to use for output (default = 2). +1 used for stress measure
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/mds.html for an example in Radiant

See Also

mds to calculate results

plot.mds to plot results

Examples

```
result <- mds(city, "from", "to", "distance")
summary(result, dec = 1)
```

---

summary.pre_factor *Summary method for the pre_factor function*

Description

Summary method for the pre_factor function

Usage

```
## S3 method for class 'pre_factor'
summary(object, dec = 2, ...)
```

Arguments

object	Return value from pre_factor
dec	Rounding to use for output
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/pre_factor.html for an example in Radiant

See Also

pre_factor to calculate results

plot.pre_factor to plot results

Examples

```
result <- pre_factor(shopping, "v1:v6")
summary(result)
pre_factor(computer, "high_end:business") %>% summary()
```

summary.prmap *Summary method for the prmap function*

Description

Summary method for the prmap function

Usage

```
## S3 method for class 'prmap'
summary(object, cutoff = 0, dec = 2, ...)
```

Arguments

object	Return value from prmap
cutoff	Show only loadings with (absolute) values above cutoff (default = 0)
dec	Rounding to use for output
...	further arguments passed to or from other methods

Details

See https://radiant-rstats.github.io/docs/multivariate/prmap.html for an example in Radiant

See Also

prmap to calculate results

plot.prmap to plot results

Examples

```
result <- prmap(computer, brand = "brand", attr = "high_end:business")
summary(result)
summary(result, cutoff = .3)
prmap(
  computer,
  brand = "brand", attr = "high_end:dated",
  pref = c("innovative", "business")
) %>% summary()
```

the_table *Function to calculate the PW and IW table for conjoint*

Description

Function to calculate the PW and IW table for conjoint

Usage

```
the_table(model, dataset, evar)
```

Arguments

model	Tidied model results (broom) output from conjoint passed on by summary.conjoint
dataset	Conjoint data
evar	Explanatory variables used in the conjoint regression

Details

See https://radiant-rstats.github.io/docs/multivariate/conjoint.html for an example in Radiant

See Also

conjoint to generate results

summary.conjoint to summarize results

plot.conjoint to plot results

Examples

```
result <- conjoint(mp3, rvar = "Rating", evar = "Memory:Shape")
the_table(tidy(result$model_list[[1]][["model"]]), result$dataset, result$evar)
```

toothpaste *Toothpaste attitudes*

Description

Toothpaste attitudes

Usage

```
data(toothpaste)
```

Format

A data frame with 60 rows and 10 variables

Details

Attitudinal data on toothpaste for 60 consumers. Description provided in attr(toothpaste, "description")

`tpbrands` *Toothpaste brands*

Description

Toothpaste brands

Usage

```
data(tpbrands)
```

Format

A data frame with 45 rows and 4 variables

Details

Perceived (dis)similarity of a set of toothpaste brands. The dataset is used to illustrate multidimensional scaling (MDS). Description provided in attr(tpbrands, "description")

Index