

# Package ‘radiant.multivariate’

March 5, 2019

**Type** Package

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

**Version** 0.9.9

**Date** 2019-3-4

**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 (>= 3.4.0), radiant.data (>= 0.9.7)

**Imports** radiant.model (>= 0.9.7), shiny (>= 1.2.0), dplyr (>= 0.8.0), rlang (>= 0.3.1), ggplot2 (>= 2.2.1), gridExtra (>= 2.0.0), 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), Gmedian (>= 1.2.3), ggrepel (>= 0.8)

**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** 6.1.1

**NeedsCompilation** no

**Author** Vincent Nijs [aut, cre]

**Maintainer** Vincent Nijs <[radiant@rady.ucsd.edu](mailto:radiant@rady.ucsd.edu)>

**Repository** CRAN

**Date/Publication** 2019-03-05 15:20:03 UTC

**R topics documented:**

|                                 |    |
|---------------------------------|----|
| carpet                          | 3  |
| city                            | 3  |
| city2                           | 4  |
| clean_loadings                  | 4  |
| computer                        | 5  |
| conjoint                        | 5  |
| full_factor                     | 6  |
| hclus                           | 7  |
| kclus                           | 8  |
| mds                             | 9  |
| movie                           | 10 |
| mp3                             | 10 |
| plot.conjoint                   | 11 |
| plot.full_factor                | 12 |
| plot.hclus                      | 13 |
| plot.kclus                      | 14 |
| plot.mds                        | 15 |
| plot.pre_factor                 | 16 |
| plot.prmmap                     | 17 |
| predict.conjoint                | 18 |
| predict_conjoint_by             | 19 |
| pre_factor                      | 20 |
| print.conjoint.predict          | 21 |
| prmap                           | 21 |
| radiant.multivariate            | 22 |
| radiant.multivariate-deprecated | 23 |
| radiant.multivariate_viewer     | 23 |
| radiant.multivariate_window     | 24 |
| retailers                       | 24 |
| shopping                        | 25 |
| store.conjoint                  | 25 |
| store.conjoint.predict          | 26 |
| store.full_factor               | 26 |
| store.kclus                     | 27 |
| summary.conjoint                | 28 |
| summary.full_factor             | 29 |
| summary.hclus                   | 30 |
| summary.kclus                   | 30 |
| summary.mds                     | 31 |
| summary.pre_factor              | 32 |
| summary.prmmap                  | 33 |
| the_table                       | 34 |
| toothpaste                      | 34 |
| tpbrands                        | 35 |

---

|        |                        |
|--------|------------------------|
| carpet | <i>Carpet cleaners</i> |
|--------|------------------------|

---

**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 | <i>City distances</i> |
|------|-----------------------|

---

**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 | <i>City distances 2</i> |
|-------|-------------------------|

---

**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 | <i>Sort and clean loadings</i> |
|----------------|--------------------------------|

---

**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](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$flodings, fsort = TRUE, cutoff = .5, dec = 2)
```

---

|          |  |
|----------|--|
| computer | <i>Perceptions of computer (re)sellers</i> |
|----------|--|

---

**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 | <i>Conjoint analysis</i> |
|----------|--------------------------|

---

**Description**

Conjoint analysis

**Usage**

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

**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") |

**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", nr_fact = 1,
  rotation = "varimax", data_filter = "")
```

**Arguments**

|             |  |
|-------------|--|
| dataset     | Dataset  |
| vars        | Variables to include in the analysis   |
| method      | Factor extraction method to use  |
| 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") |

**Details**

See [https://radiant-rstats.github.io/docs/multivariate/full\\_factor.html](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 | <i>Hierarchical cluster analysis</i> |
|-------|--------------------------------------|

---

**Description**

Hierarchical cluster analysis

**Usage**

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

**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                   |
| 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") |

**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 = "mean", hc_init = TRUE,
       distance = "sq.euclidian", method = "ward.D", seed = 1234,
       nr_clus = 2, data_filter = "")
```

**Arguments**

|             |  |
|-------------|--|
| dataset     | Dataset  |
| vars        | Vector of variables to include in the analysis   |
| fun         | Function to use: "mean" or "median"  |
| 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  |
| 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") |

**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 = "")
```

**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") |

**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 | <i>Conjoint data for Movie theaters</i> |
|-------|---|

---

**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 | <i>Conjoint data for MP3 players</i> |
|-----|--------------------------------------|

---

**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 | <i>Plot method for the conjoint function</i> |
|---------------|--|

---

**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 <a href="#">conjoint</a>  |
| 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 <a href="http://docs.ggplot2.org">http://docs.ggplot2.org</a> 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 <a href="#">full_factor</a>   |
| 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 <a href="http://docs.ggplot2.org">http://docs.ggplot2.org</a> for options. |
| ...    | further arguments passed to or from other methods   |

## Details

See [https://radiant-rstats.github.io/docs/multivariate/full\\_factor.html](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 <a href="#">hclus</a>   |
| 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 <a href="http://docs.ggplot2.org">http://docs.ggplot2.org</a> 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 <a href="#">kclus</a>   |
| 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 <a href="http://docs.ggplot2.org">http://docs.ggplot2.org</a> 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 | <i>Plot method for the mds function</i> |
|----------|---|

---

## 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 <a href="#">mds</a>   |
| 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 <a href="http://docs.ggplot2.org">http://docs.ggplot2.org</a> 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 <a href="#">pre_factor</a>  |
| 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 <a href="http://docs.ggplot2.org">http://docs.ggplot2.org</a> for options. |
| ...    | further arguments passed to or from other methods   |

### Details

See [https://radiant-rstats.github.io/docs/multivariate/pre\\_factor.html](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.prmmap | <i>Plot method for the prmap function</i> |
|-------------|---|

---

### 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 <a href="#">prmap</a>   |
| 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 <a href="http://docs.ggplot2.org">http://docs.ggplot2.org</a> 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.prmmap](#) 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,
  ...)

```

## Arguments

|           |  |
|-----------|--|
| object    | Return value from <a href="#">conjoint</a>   |
| 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   |
| ...       | 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, ...)
```

### Arguments

|           |   |
|-----------|---|
| object    | Return value from <a href="#">conjoint</a>  |
| 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  |
| ...       | 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 | <i>Evaluate if data are appropriate for PCA / Factor analysis</i> |
|------------|---|

---

### Description

Evaluate if data are appropriate for PCA / Factor analysis

### Usage

```
pre_factor(dataset, vars, data_filter = "")
```

### Arguments

|             |  |
|-------------|--|
| dataset     | Dataset  |
| vars        | Variables to include in the analysis   |
| 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") |

### Details

See [https://radiant-rstats.github.io/docs/multivariate/pre\\_factor.html](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, data_filter = "")
```

**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   |
| 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") |

**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.prmmap` to summarize results

`plot.prmmap` to plot results

**Examples**

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

---

`radiant.multivariate` *radiant.multivariate*

---

**Description**

`radiant.multivariate`

Launch `radiant.multivariate` in the default 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-deprecated`*Deprecated function(s) in the radiant.multivariate package*

---

**Description**

These functions are provided for compatibility with previous versions of radiant. They will eventually be removed.

**Usage**`pmap(...)`**Arguments**

... Parameters to be passed to the updated functions

**Details**

- Replace `pmap` by [prmap](#)

---

`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**

|                    |   |
|--------------------|---|
| <code>state</code> | Path to state file to load  |
| <code>...</code>   | additional arguments to pass to <code>shiny::runApp</code> (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 | <i>Shopping attitudes</i> |
|----------|---------------------------|

---

**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 | <i>Store method for the Multivariate &gt; Conjoint tab</i> |
|----------------|--|

---

**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 <code>full_factor</code> |
| name    | Name of factor score variables             |
| ...     | Additional arguments                       |

**Details**

See [https://radiant-rstats.github.io/docs/multivariate/full\\_factor.html](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.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 <code>kclus</code>                |
| 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 <a href="#">conjoint</a>                    |
| 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 <a href="#">full_factor</a>                        |
| 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](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 <a href="#">hclus</a>           |
| ...    | 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 <a href="#">kclus</a>           |
| 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 <a href="#">mds</a>                                |
| 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 | <i>Summary method for the pre_factor function</i> |
|--------------------|---|

---

**Description**

Summary method for the pre\_factor function

**Usage**

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

**Arguments**

|        |   |
|--------|---|
| object | Return value from <a href="#">pre_factor</a>      |
| 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](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.pmap`*Summary method for the pmap function*

---

## Description

Summary method for the pmap function

## Usage

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

## Arguments

|                     |  |
|---------------------|--|
| <code>object</code> | Return value from <a href="#">pmap</a>                               |
| <code>cutoff</code> | Show only loadings with (absolute) values above cutoff (default = 0) |
| <code>dec</code>    | Rounding to use for output   |
| <code>...</code>    | further arguments passed to or from other methods                    |

## Details

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

## See Also

[pmap](#) to calculate results  
[plot.pmap](#) to plot results

## Examples

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

---

|           |   |
|-----------|---|
| the_table | <i>Function to calculate the PW and IW table for conjoint</i> |
|-----------|---|

---

**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 <a href="#">conjoint</a> passed on by <code>summary.conjoint</code> |
| 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 | <i>Toothpaste attitudes</i> |
|------------|-----------------------------|

---

**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 multi-dimensional scaling (MDS). Description provided in attr(tpbrands, "description")

# Index

## \*Topic **datasets**

- carpet, 3
- city, 3
- city2, 4
- computer, 5
- movie, 10
- mp3, 10
- retailers, 24
- shopping, 25
- toothpaste, 34
- tpbrands, 35

carpet, 3

city, 3

city2, 4

clean\_loadings, 4

computer, 5

conjoint, 5, 11, 18, 19, 28, 29, 34

full\_factor, 6, 12, 27, 29

hclus, 7, 13, 30

kclus, 8, 14, 27, 28, 31

mds, 9, 15, 31, 32

movie, 10

mp3, 10

plot.conjoint, 6, 11, 18, 19, 29, 34

plot.full\_factor, 7, 12, 12, 27, 29

plot.hclus, 8, 13, 30

plot.kclus, 9, 14, 28, 31

plot.mds, 10, 15, 32

plot.pre\_factor, 16, 20, 32

plot.prmmap, 17, 22, 33

prmap (radiant.multivariate-deprecated), 23

pre\_factor, 16, 20, 32

predict.conjoint, 18

predict\_conjoint\_by, 19

print.conjoint.predict, 21

prmap, 17, 21, 23, 33

radiant.multivariate, 22

radiant.multivariate-deprecated, 23

radiant.multivariate-deprecated-package (radiant.multivariate-deprecated), 23

radiant.multivariate-package (radiant.multivariate), 22

radiant.multivariate\_viewer, 23

radiant.multivariate\_window, 24

retailers, 24

shopping, 25

store.conjoint, 25

store.conjoint.predict, 26

store.full\_factor, 26

store.kclus, 9, 14, 27, 31

summary.conjoint, 6, 11, 18, 19, 28, 34

summary.full\_factor, 7, 27, 29

summary.hclus, 8, 13, 30

summary.kclus, 9, 14, 28, 30

summary.mds, 10, 15, 31

summary.pre\_factor, 16, 20, 32

summary.prmmap, 17, 22, 33

the\_table, 34

toothpaste, 34

tpbrands, 35