Package 'tidyLPA'

September 14, 2018

Type Package

Title Easily Carry Out Latent Profile Analysis

Version 0.2.0

Description An interface to the 'mclust' package to easily carry out latent profile analysis (``LPA''). Provides functionality to estimate commonly-specified models. Follows a tidy approach, in that output is in the form of a data frame that can subsequently be computed on. Also has functions to interface to the commercial 'MPlus' software via the 'MplusAutomation' package.

License MIT + file LICENSE

URL https://jrosen48.github.io/tidyLPA/

BugReports https://github.com/jrosen48/tidyLPA/issues

Depends R (>= 2.10)

Imports dplyr, forcats, ggplot2, magrittr, mclust, purrr, readr, rlang, stringr, tibble, tidyr

Suggests covr, devtools, knitr, MplusAutomation, parallel, rmarkdown, roxygen2, testthat

VignetteBuilder knitr

Encoding UTF-8

LazyData true

RoxygenNote 6.1.0

NeedsCompilation no

Author Joshua M Rosenberg [aut, cre], Jennifer A Schmidt [ctb], Patrick N Beymer [ctb], Daniel Anderson [ctb], Caspar van Lissa [ctb]

Maintainer Joshua M Rosenberg <jmichaelrosenberg@gmail.com>

Repository CRAN

Date/Publication 2018-09-13 22:40:03 UTC

R topics documented:

`bootstrap_lrt` *Bootstrap the likelihood-ratio test statistic for mixture components*

Description

Bootstrap the likelihood-ratio test statistic for mixture components

Usage

```
bootstrap_lrt(df, ..., n_profiles, variances = "fixed",
  covariances = "zero")
```

Arguments

`df`	data.frame with two or more columns with continuous variables
`...`	unquoted variable names separated by commas
`n_profiles`	the number of profiles (or mixture components) to be estimated
`variances`	how the variable variances are estimated; defaults to "equal" (to be constant across profiles); other option is "varying" (to be varying across profiles)
`covariances`	how the variable covariances are estimated; defaults to "zero" (to not be estimated, i.e. for the covariance matrix to be diagonal); other options are "varying" (to be varying across profiles) and "equal" (to be constant across profiles)

Details

Bootstrap the p-values for the likelihood-ratio test statistic for the number of mixture components for an mclust model.

Examples

```
## Not run:
d <- pisaUSA15
d <- dplyr::sample_n(d, 200)
bootstrap_lrt(d,
              broad_interest, enjoyment, self_efficacy)

## End(Not run)
```

`compare_solutions` *Explore BIC for various models and numbers of profiles*

Description

Explore BIC for various models and numbers of profiles

Usage

```
compare_solutions(df, ..., n_profiles_range = 1:9,
  models = list(c("equal", "zero"), c("varying", "zero"), c("equal",
  "equal"), c("varying", "varying")), center_raw_data = FALSE,
  scale_raw_data = FALSE, statistic = "BIC", return_table = FALSE,
  prior_control = F)
```

Arguments

`df`	data.frame with two or more columns with continuous variables
`...`	unquoted variable names separated by commas
`n_profiles_range`	a vector with the range of the number of mixture components to explore; defaults to 1 through 9 (1:9)
`models`	which models to include as a list of vectors; for each vector, the first value represents how the variances are estimated and the second value represents how the covariances are estimated; defaults to list(c("equal", "zero"), c("varying", "zero"), c("equal", "equal"), c("varying", "varying"))
`center_raw_data`	logical for whether to center (M = 1) the raw data (before clustering); defaults to FALSE
`scale_raw_data`	logical for whether to scale (SD = 1) the raw data (before clustering); defaults to FALSE
`statistic`	what statistic to plot; BIC or ICL are presently available as options
`return_table`	logical (TRUE or FALSE) for whether to return a table of the output instead of a plot; defaults to FALSE
`prior_control`	whether to include a regularizing prior; defaults to false

Details

Explore the BIC values of a range of models in terms of a) the structure of the residual covariance matrix and b) the number of mixture components (or profiles)

Value

a ggplot2 plot of the BIC values for the explored models

Examples

```
compare_solutions(iris, Sepal.Length, Sepal.Width, Petal.Length, Petal.Width)
```

---

compare_solutions_mplus

Explore fit statistics various models and numbers of profiles using MPlus (requires purchasing and installing MPlus to use)

Description

Explore fit statistics various models and numbers of profiles using MPlus (requires purchasing and installing MPlus to use)

Usage

```
compare_solutions_mplus(df, ..., n_profiles_min = 2,
  n_profiles_max = 10, models = list(c("equal", "zero"), c("varying",
  "zero"), c("equal", "equal"), c("varying", "varying")), starts = c(100,
  10), cluster_ID = NULL, m_iterations = 500, st_iterations = 20,
  convergence_criterion = 1e-06, save_models = FALSE,
  return_table = TRUE, n_processors = 1, return_stats_df = TRUE,
  include_VLMR = TRUE, include_BLRT = FALSE)
```

Arguments

df	data.frame with two or more columns with continuous variables
...	unquoted variable names separated by commas
n_profiles_min	lower bound of the number of profiles to explore; defaults to 2
n_profiles_max	upper bound of the number of profiles to explore; defaults to 10
models	which models to include as a list of vectors; for each vector, the first value represents how the variances are estimated and the second value represents how the covariances are estimated; defaults to list(c("equal", "zero"), c("varying", "zero"), c("equal", "equal"), c("varying", "varying"))
starts	number of initial stage starts and number of final stage optimizations; defaults to c(20, 4); can be set to be more conservative to c(500, 50)
cluster_ID	clustering variable to use as part of MPlus 'type is complex' command

m_iterations	number of iterations for the EM algorithm; defaults to 500
st_iterations	the number of initial stage iterations; defaults to 10; can be set more to be more conservative to 50
convergence_criterion	convergence criterion for the Quasi-Newton algorithm for continuous outcomes; defaults to 1E-6 (.000001); can be set more conservatively to 1E-7 (.0000001)
save_models	whether to save the models as rds files
return_table	logical (TRUE or FALSE) for whether to return a table of the output instead of a plot; defaults to TRUE
n_processors	= 1
return_stats_df	whether to return a list of fit statistics for the solutions explored; defaults to TRUE
include_VLMR	whether to include the Vu-Lo-Mendell-Rubin likelihood-ratio test; defaults to TRUE
include_BLRT	whether to include the bootstrapped LRT; defaults to FALSE because of the time this takes to run

Details

Explore the BIC values of a range of Mplus models in terms of a) the structure of the residual covariance matrix and b) the number of mixture components (or profiles)

Value

a list with a data.frame with the BIC values and a list with all of the model output; if save_models is the name of an rds file (i.e., "out.rds"), then the model output will be written with that filename and only the data.frame will be returned

Examples

```
## Not run:
compare_solutions_mplus(iris, Sepal.Length, Sepal.Width, Petal.Length, Petal.Width,
n_profiles_max = 3)

## End(Not run)
```

---

estimate_profiles *Estimate parameters for profiles for a specific solution*

Description

Estimate parameters for profiles for a specific solution

Usage

```
estimate_profiles(df, ..., n_profiles, variances = "equal",
  covariances = "zero", to_return = "tibble", model = NULL,
  center_raw_data = FALSE, scale_raw_data = FALSE,
  return_posterior_probs = TRUE, return_orig_df = FALSE,
  prior_control = FALSE, print_which_stats = "some")
```

Arguments

df	data.frame with two or more columns with continuous variables
...	unquoted variable names separated by commas
n_profiles	the number of profiles (or mixture components) to be estimated
variances	how the variable variances are estimated; defaults to "equal" (to be constant across profiles); other option is "varying" (to be varying across profiles)
covariances	how the variable covariances are estimated; defaults to "zero" (to not be estimated, i.e. for the covariance matrix to be diagonal); other options are "varying" (to be varying across profiles) and "equal" (to be constant across profiles)
to_return	character string for either "tibble" (or "data.frame") or "mclust" if "tibble" is selected, then data with a column for profiles is returned; if "mclust" is selected, then output of class mclust is returned
model	which model to estimate (DEPRECATED; use variances and covariances instead)
center_raw_data	logical for whether to center (M = 1) the raw data (before clustering); defaults to FALSE
scale_raw_data	logical for whether to scale (SD = 1) the raw data (before clustering); defaults to FALSE
return_posterior_probs	TRUE or FALSE (only applicable if to_return == "tibble"); whether to include posterior probabilities in addition to the posterior profile classification; defaults to TRUE
return_orig_df	TRUE or FALSE (if TRUE, then the entire data.frame is returned; if FALSE, then only the variables used in the model are returned)
prior_control	whether to include a regularizing prior; defaults to false
print_which_stats	if set to "some", prints (as a message) the log-likelihood, BIC, and entropy; if set to "all", prints (as a message) all information criteria and other statistics about the model; if set to any other values, then nothing is printed

Details

Creates profiles (or estimates of the mixture components) for a specific mclust model in terms of the specific number of mixture components and the structure of the residual covariance matrix

Value

either a tibble or a ggplot2 plot of the BIC values for the explored models

Examples

```
estimate_profiles(iris,
    Sepal.Length, Sepal.Width, Petal.Length, Petal.Width,
    n_profiles = 3)
```

`estimate_profiles_mplus` *Estimate parameters for profiles for a specific solution (requires purchasing and installing MPlus to use)*

Description

Estimate parameters for profiles for a specific solution (requires purchasing and installing MPlus to use)

Usage

```
estimate_profiles_mplus(df, ..., n_profiles, idvar = NULL,
  data_filename = "d.dat", script_filename = "i.inp",
  output_filename = "i.out", savedata_filename = "d-mod.dat",
  variances = "equal", covariances = "zero", model = NULL,
  starts = c(100, 10), m_iterations = 500, st_iterations = 20,
  convergence_criterion = 1e-06, remove_tmp_files = TRUE,
  print_input_file = FALSE, return_save_data = TRUE, optseed = NULL,
  n_processors = 1, cluster_ID = NULL, include_VLMR = TRUE,
  include_BLRT = FALSE, return_all_stats = FALSE)
```

Arguments

`df`	data.frame with two or more columns with continuous variables
`...`	unquoted variable names separated by commas
`n_profiles`	the number of profiles (or mixture components) to be estimated
`idvar`	optional name of the column to be used as the ID variable (should be supplied as a string). Defaults to `NULL`, in which case row numbers will be used. Note the ID can be numeric or string, but must be unique.
`data_filename`	name of data file to prepare; defaults to d.dat
`script_filename`	name of script to prepare; defaults to i.inp
`output_filename`	name of the output; defaults to o.out

savedata_filename
: name of the output for the save data (with the original data conditional probabilities); defaults to o-mod.out

variances
: how the variable variances are estimated; defaults to "equal" (to be constant across profiles); other option is "varying" (to be varying across profiles)

covariances
: how the variable covariances are estimated; defaults to "zero" (to not be estimated, i.e. for the covariance matrix to be diagonal); other options are "varying" (to be varying across profiles) and "equal" (to be constant across profiles)

model
: which model to estimate (DEPRECATED; use variances and covariances instead)

starts
: number of initial stage starts and number of final stage optimizations; defaults to c(20, 4); can be set to be more conservative to c(500, 50)

m_iterations
: number of iterations for the EM algorithm; defaults to 500

st_iterations
: the number of initial stage iterations; defaults to 10; can be set more to be more conservative to 50

convergence_criterion
: convergence criterion for the Quasi-Newton algorithm for continuous outcomes; defaults to 1E-6 (.000001); can be set more conservatively to 1E-7 (.0000001)

remove_tmp_files
: whether to remove data, script, and output files; defaults to TRUE

print_input_file
: whether to print the input file to the console

return_save_data
: whether to return the save data (with the original data and the posterior probabilities for the classes and the class assignment) as a data.frame along with the MPlus output; defaults to TRUE

optseed
: random seed for analysis

n_processors
: = 1

cluster_ID
: clustering variable (i.e., if data are from students clustered into distinct classrooms) to be used as cluster variables as part of the type = complex option

include_VLMR
: whether to include the Vu-Lo-Mendell-Rubin likelihood-ratio test; defaults to TRUE

include_BLRT
: whether to include the bootstrapped LRT; defaults to FALSE because of the time this takes to run

return_all_stats
: defaults to FALSE; if TRUE, returns as a one-row data frame all of the statistics returned from compare_solutions_mplus()

Details

Creates an mplus model (.inp) and associated data file (.dat)

Value

either a tibble or a ggplot2 plot of the BIC values for the explored models

Examples

```
## Not run:
m <- estimate_profiles_mplus(iris,
                             Sepal.Length, Sepal.Width, Petal.Length, Petal.Width,
                             n_profiles = 2)

## End(Not run)
```

`extract_LL_mplus` *Extract log-likelihoods from models fit with estimate_profiles_mplus()*

Description

Extract log-likelihoods from models fit with estimate_profiles_mplus()

Usage

```
extract_LL_mplus(output_filename = "i.out")
```

Arguments

`output_filename`
: name of output_filename from estimate_profiles_mplus()

Details

Extract log-likelihoods associated with solutions from random starts from estimate_profiles_mplus(). Note that return_tmp_files = FALSE must be added to estimate_profiles_mplus() for this function to work.

Value

a tibble or a ggplot2 plot of the BIC values for the explored modelswith the log-likelihood, random start seed, and the number of the iteration

Examples

```
## Not run:
m1 <- estimate_profiles_mplus(iris,
                             Sepal.Length, Sepal.Width, Petal.Length, Petal.Width,
                             n_profiles = 2,
                             remove_tmp_files = FALSE)
extract_LL_mplus()

## End(Not run)
```

`pisaUSA15` *student questionnaire data with four variables from the 2015 PISA for students in the United States*

Description

student questionnaire data with four variables from the 2015 PISA for students in the United States

Usage

```
pisaUSA15
```

Format

Data frame with columns #'

broad_interest composite measure of students' self reported broad interest

enjoyment composite measure of students' self reported enjoyment

instrumental_mot composite measure of students' self reported instrumental motivation

self_efficacy composite measure of students' self reported self efficacy ...

Source

http://www.oecd.org/pisa/data/

`plot_profiles` *Plot variable means and variances by profile*

Description

Plot variable means and variances by profile

Usage

```
plot_profiles(x, to_center = F, to_scale = F, plot_what = "tibble",
  plot_error_bars = TRUE, plot_rawdata = TRUE, ci = 0.95)
```

Arguments

x	output from estimate_profiles()
to_center	whether to center the data before plotting
to_scale	whether to scale the data before plotting
plot_what	whether to plot tibble or mclust output from estimate_profiles(); defaults to tibble
plot_error_bars	whether to plot error bars (representing the 95 percent confidence interval for the mean of each variable)
plot_rawdata	whether to plot raw data; defaults to TRUE
ci	confidence interval to plot (defaults to 0.95)

Details

Plot the variable means and variances for data frame output from estimate_profiles()

Plot the variable means and variances for data frame output from estimate_profiles(). When plot_what is set to 'mclust', the errorbars represent non-parametric confidence intervals, obtained using bootstrapping (100 samples). Note that 100 samples might be adequate for plotting, but is low for inference. If the number of participants per class is highly unbalanced, then weighted likelihood bootstrapping is used to ensure that each case is represented in the bootstrap samples (see O'Hagan, Murphy, Scrucca, and Gormley, 2015).

Examples

```
m3 <- estimate_profiles(iris,
    Sepal.Length, Sepal.Width, Petal.Length, Petal.Width,
    n_profiles = 3)
plot_profiles(m3)

m3 <- estimate_profiles(iris,
    Sepal.Length, Sepal.Width, Petal.Length, Petal.Width,
    n_profiles = 3, to_return = "mclust")
plot_profiles(m3, plot_what = "mclust")
```

---

plot_profiles_mplus *Plot variable means and their confidence intervals by profile for models estimated with MPlus (requires purchasing and installing MPlus to use)*

Description

Plot variable means and their confidence intervals by profile for models estimated with MPlus (requires purchasing and installing MPlus to use)

Usage

```
plot_profiles_mplus(mplus_data, to_center = T, to_scale = T)
```

Arguments

mplus_data	output from estimate_profiles_mplus() with return_savedata = T specified
to_center	whether to center the data before plotting
to_scale	whether to scale the data before plotting

Details

Plot the variable means and variances for data frame output from estimate_profiles_mclust()

---

tidyLPA *tidyLPA: Functionality to carry out Latent Profile Analysis in R*

Description

Latent Profile Analysis (LPA) is a statistical modeling approach for estimating distinct profiles, or groups, of variables. In the social sciences and in educational research, these profiles could represent, for example, how different youth experience dimensions of being engaged (i.e., cognitively, behaviorally, and affectively) at the same time.

Details

tidyLPA provides the functionality to carry out LPA in R. In particular, tidyLPA provides functionality to specify different models that determine whether and how different parameters (i.e., means, variances, and covariances) are estimated and to specify (and compare solutions for) the number of profiles to estimate.

Index