Package ‘GAabbreviate’

October 12, 2022

Type Package
Version 1.3
Date 2016-06-22
Title Abbreviating Items Measures using Genetic Algorithms
Description Scale abbreviation using Genetic Algorithms that maximally capture the variance in the original data.
Depends R (>= 3.0), GA (>= 3.0), psych (>= 1.4.3)
Imports stats, graphics, grDevices, utils
License GPL (>= 2)
ByteCompile true
LazyLoad yes
NeedsCompilation no
Author Luca Scrucca [aut],
Baljinder K. Sahdra [aut, cre]
Maintainer Baljinder K. Sahdra <baljinder.sahdra@acu.edu.au>
Repository CRAN
Date/Publication 2016-06-23 00:41:26

R topics documented:

GAabbreviate .......................................................... 2

Index 6
GAabbreviate is a function for abbreviating items (from questionnaire or other) measures using Genetic Algorithms (GAs).

**Description**

The GAabbreviate uses Genetic Algorithms as an optimization tool for scale abbreviation or subset selection that maximally captures the variance in the original data.

**Usage**

```r
GAabbreviate(items = NULL,
             scales = NULL,
             itemCost = 0.05,
             maxItems = 5,
             maxiter = 100,
             popSize = 50,
             ...,
             plot = FALSE,
             verbose = interactive(),
             crossVal = TRUE,
             impute = FALSE,
             pairwise = FALSE,
             minR = 0,
             sWeights = NULL,
             nSample = NULL,
             seed = NULL)
```

**Arguments**

- **items**: A matrix of subjects x item scores.
- **scales**: A matrix of subjects x scale scores.
- **itemCost**: The fitness cost of each item. This will usually need to be determined by trial and error.
- **maxItems**: The maximum number of items used to score each scale.
- **maxiter**: Number of generations of GA to run.
- **popSize**: Size of population in each generation of the GA.
- **plot**: Logical; if TRUE, plot results after every generation (this will slow down the process).
- **verbose**: Logical; by default in interactive sessions is set to TRUE, so some info during the search is displayed.
- **crossVal**: Logical; if TRUE, cross-validation will be performed. Note that if you turn this off, the predictive fit of the resulting measure will be biased by overfitting.
impute Logical; if TRUE, the mean value will be imputed for any missing item or scale scores. This is NOT recommended. Instead, you should decide how to handle missing values yourself before passing the items and scales variables.

pairwise Logical; if TRUE, the GA will use pairwise deletion to select items, i.e. some scales/items may have different NAs than others. If FALSE, the GA will crash if any NAs are passed. It’s recommended to leave this off, as NAs should really be handled separately.

minR The minimum bivariate item-scale correlation required in order to retain an item. Note that if this is set above 0, the number of items retained can be lower than the value of maxItems.

sWeights Weighting of scales. By default, all scales will have unit weighting, but if you want to emphasize some scales more heavily, pass a vector with length equal to the number of scales.

nSample For extremely large datasets, you may wish to use only a subset of observations to generate a measure. Passing any non-zero number will randomly select nSample observations to use instead of drawing on the full dataset.

seed An integer value containing the random number generator state. Set this argument to make the results exactly reproducible.

Details

The GAabbreviate uses Genetic Algorithms (GAs) as an optimization tool for shortening a large set of variables (e.g., in a lengthy battery of questionnaires) into a shorter subset that maximally captures the variance in the original data. An exhaustive search of all possible shorter forms of the original measure would be time consuming, especially for a measure with a large number of items. For a long form of length $L$ (e.g., 100 items of a self-report scale), the size of the search space is $2^L$ (1.26e+30) and forms a hypercube of $L$ dimensions. The GA uses hypercube sampling by sampling the corners of the $L$-dimensional hypercube. It optimizes the search by mimicking Darwinian evolution mechanisms (of selection, crossover, and mutation) while searching through a "landscape" of the collection of all possible fitness values to find an optimal value. This does not imply that the GA finds the "best" possible solution. Rather, the GA is highly efficient in quickly yielding a "good" and "robust" solution rated against a user-defined fitness criterion.

The GAabbreviate uses the GA package (Scrucca, 2013) to efficiently implement Yarkoni’s (2010) scale abbreviation cost function:

$$Cost = Ik + \sum_{i=1}^{s} w_i (1 - R^2_i)$$

where $I$ represents a user-specified fixed item cost, $k$ represents the number of items retained by the GA (in any given iteration), $s$ is the number of subscales in the measure, $w_i$ are the weights (by default $w_i = 1$ for any $i$) associated with each subscale (if there are any subsets to be retained), and $R^2_i$ is the amount of variance in the $i$th subscale that can be explained by a linear combination of individual item scores. Adjusting the value of $I$ low or high yields longer or shorter measures respectively. When the cost of each individual item retained in each generation outweighs the cost of a loss in explained variance, the GA yields a relatively brief measure. When the cost is low, the GA yields a relatively longer measure maximizing explained variance (Yarkoni, 2010).

Sahdra, Ciarrochi, Parker & Scrucca (2016) contains an example of how GAabbreviate can be used for item-reduction of a multidimensional scale.
Value

An object of class 'GAabbreviate' providing the following information:

- **data**: The input data.
- **settings**: The input settings.
- **results**: The results obtained.
- **best**: The cost and fit of the final solution.
- **GA**: An object of class 'GA'.
- **measure**: A list of measure values.

A summary and `plot` methods are available to inspect the results. See example section.

Author(s)

Luca Scrucca, Department of Economics, University of Perugia, Perugia, ITALY

Baljinder K. Sahdra, Institute for Positive Psychology and Education, Australian Catholic University, Strathfield, NSW, AUSTRALIA

Send inquiries to <baljinder.sahdra@acu.edu.au>.

References


See Also

ga

Examples

```r
### Example using random generated data
nsubject = 100
nitems = 15
set.seed(123)
items = matrix(sample(1:5, nsubject*nitems, replace = TRUE),
               nrow = nsubject, ncol = nitems)
scales = cbind(rowSums(items[,1:10]), rowSums(items[,11:15]))

GAA = GAabbreviate(items, scales, itemCost = 0.01, maxItems = 5,
                    popSize = 50, maxiter = 300, run = 100)
plot(GAA)
```
summary(GAA)
# more info can be retrieved using
GAA$best
GAA$measure
Index

* multivariate
  GAabbreviate, 2
* optimize
  GAabbreviate, 2
* survey
  GAabbreviate, 2

ga, 2, 4
GAabbreviate, 2