varSelection function

Simone Del Sarto

21/7/2019

This function allows for the item selection based on two steps:

1. preliminary selection (non model-based)
2. IRT selection (model-based)

Preparation

```
source("varSelection.R")
load("dati_UNHCR.RData")
```

where "dati_UNHCR.RData" is an object containing the data for which the item selection has to be applied.

Suppose the `household` object contained the dataset at issue (UNHCR Mauritania household dataset)

```
nrow(household); ncol(household)
```

```
## [1] 12747
```

```
## [1] 223
```

First, select the food sections (from 4 to 10): columns from 20 to 109

```
hh <- household[, 20:109] #90 variables
```

This will be the first argument of function `varSelection`.

Arguments

```
## function (data, miss_thr = 2/3, gamma_thr = 0.7, beta_thr = c(-3,
##     3), m = 10, crit_thr = 0.95, Theta_range = c(-15, 15), qsel = 0.25,
##     ...)
## NULL
```

- `data`: dataset (dichotomous variable must be 0/1 coded)
- `miss_thr`: threshold for missing values (default 2/3)
- `gamma_thr`: threshold for item discrimination (default 0.7)
- `beta_thr`: threshold for beta parameters (default [-3,3])
- `m`: max no. of categories for discrete variables, otherwise it is considered as continuous
- `crit_thr`: threshold for "critical items"; observed response rate concentrated in a certain category (default 0.95)
- `Theta_range`: latent trait range (standard normal distributed) for evaluating the item/test information (default [-15,15])
- `qsel`: quantile for the selection based on item information proportion on the whole test information (default 0.25 = 1st quartile)
- `...`: further (if needed) arguments for 'mirt' function within IRT selection

```
out <- varSelection(hh, verbose=FALSE)
```

```
## Loading required package: stats4
```

```
## Loading required package: lattice
## *****************Preliminary selection*****************
## - Missing proportion threshold 0.6666667
## - Variables with number of response categories greater than 10 are removed
## ......
## End preliminary selection
## -  49 variables out of 90 are retained
## -  22 are "critical" items
##
## *****************IRT selection*****************
## End IRT selection
```

The output is a list of two objects:

```r
str(out$prel)
```

```
## 'data.frame':    90 obs. of  9 variables:
##  $ ID       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ name     : chr  "s4q1" "s4q2" "s4q3" "s4q4" ...
##  $ miss_prop: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ ncat     : int  2 2 2 2 2 2 2 2 2 2 ...
##  $ min      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ max      : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ mean     : num  0.997 0.185 0.152 0.265 0.814 ...
##  $ cont     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ miss     : num  0 0 0 0 0 0 0 0 0 0 ...
```

It contains the summary of all the variables in the initial dataset

- `ID`: ID of the variables
- `name`: name of the variables
- `miss_prop`: observed proportion of missing values
- `ncat`: number of categories (observed)
- `min`: (observed) minimum values
- `max`: (observed) maximum values
- `mean`: observed average values
- `cont`: whether the variables are considered continuous (`ncat` > `m`), then discarded
- `miss`: whether the missing values are beyond the threshold (`miss_thr`), then discarded

```r
str(out$final)
```

```
## 'data.frame':    49 obs. of  24 variables:
##  $ ID       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ name     : chr  "s4q1" "s4q2" "s4q3" "s4q4" ...
##  $ miss_prop: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ ncat     : int  2 2 2 2 2 2 2 2 2 2 ...
##  $ min      : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ max      : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ mean     : num  0.997 0.185 0.152 0.265 0.814 ...
##  $ cont     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ miss     : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ crit_item: num  1 0 0 0 0 1 0 1 1 1 ...
##  $ gamma    : num  1.027 1.211 1.297 1.201 0.918 ...
##  $ beta1    : num  -6.05 1.54 1.71 1.07 -1.87 ...
##  $ beta2    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ beta3    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ beta4    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ beta5    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ beta6    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ beta7    : num  NA NA NA NA NA NA NA NA NA NA ...
##  $ prop_info: num  1.4 1.65 1.77 1.64 1.25 ...
##  $ gamma_sel: chr  "keep" "keep" "keep" "keep" ...
##  $ beta_sel : chr  "drop" "keep" "keep" "keep" ...
##  $ info_sel : chr  "drop" "drop" "drop" "drop" ...
##  $ info_selQ: chr  "keep" "keep" "keep" "keep" ...
##  $ final_sel: chr  "drop" "keep" "keep" "keep" ...
```

It contains the summary of the variables retained after the preliminary selection. In addition to the above summary, it has further columns:

- `crit_item`: whether the items are considered as "critical" (observed response rate for a certain category > `crit_thr`)
- `gamma`: estimated discrimination parameters
- `beta1`, `beta2`, ...: estimated difficulty (cut-off) parameters (as many as the number of response categories minus 1)
- `prop_info`: proportion (%) of item information over the whole test information
- `gamma_sel`: whether the item is kept or dropped according to the discrimination criterion (≥ `gamma_thr`)
- `beta_sel`: whether the item is kept or dropped according to the beta criterion (the (median) beta parameter outside interval in `beta_thr`)
- `info_sel`: whether the item is kept or dropped according to the first criterion based on item information proportion ($\geq (1/J) \times 100$, where J is the number of items)
- `info_selQ`: whether the item is kept or dropped according to the second criterion based on item information proportion (item information proportion ≥ `qsel`-quantile of the information proportion distribution)
- `final_sel`: final selection – whether the item is kept or dropped (at least two criteria out of four)