

# Package ‘aMNLFA’

November 25, 2021

**Title** Automated Moderated Nonlinear Factor Analysis Using 'M-plus'

**Version** 1.1.0

**Description** Automated generation, running, and interpretation of moderated nonlinear factor analysis models for obtaining scores from observed variables, using the method described by Gottfredson and colleagues (2019) <[doi:10.1016/j.addbeh.2018.10.031](https://doi.org/10.1016/j.addbeh.2018.10.031)>. This package creates M-plus input files which may be run iteratively to test two different types of covariate effects on items: (1) latent variable impact (both mean and variance); and (2) differential item functioning. After sequentially testing for all effects, it also creates a final model by including all significant effects after adjusting for multiple comparisons. Finally, the package creates a scoring model which uses the final values of parameter estimates to generate latent variable scores. \n\n This package generates TEMPLATES for M-plus inputs, which can and should be inspected, altered, and run by the user. In addition to being presented without warranty of any kind, the package is provided under the assumption that everyone who uses it is reading, interpreting, understanding, and altering every M-plus input and output file. There is no one right way to implement moderated nonlinear factor analysis, and this package exists solely to save users time as they generate M-plus syntax according to their own judgment.

**Depends** R (>= 3.1.0), utils

**Imports** grDevices, graphics, stats, ggplot2, MplusAutomation, reshape2, gridExtra, stringr, plyr, devtools, dplyr, stringi

**License** GPL-2

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Veronica Cole [aut, cre],  
Nisha Gottfredson [aut],  
Michael Giordano [aut],  
Isabella Stallworthy [aut],  
Meriah DeJoseph [aut],  
Robin Sifre [aut],  
Tim Janssen [ctb]

**Maintainer** Veronica Cole <[colev@wfu.edu](mailto:colev@wfu.edu)>

**Repository** CRAN

**Date/Publication** 2021-11-25 05:30:02 UTC

## R topics documented:

aMNLFA.DIFplot . . . . .	2
aMNLFA.final . . . . .	3
aMNLFA.initial . . . . .	4
aMNLFA.itemplots . . . . .	5
aMNLFA.object . . . . .	6
aMNLFA.prune . . . . .	7
aMNLFA.sample . . . . .	9
aMNLFA.scores . . . . .	10
aMNLFA.simultaneous . . . . .	11
fixPath . . . . .	12
write.inp.file . . . . .	12
xstudy . . . . .	13
<b>Index</b>	<b>15</b>

---

aMNLFA.DIFplot	<i>aMNLFA plotting function for aMNLFA.prune() results</i>
----------------	------------------------------------------------------------

---

### Description

This function gives the user a plot corresponding to loading, intercept, or threshold DIF from the aMNLFA.prune() function

### Usage

```
aMNLFA.DIFplot(diflist, diftype, log = FALSE)
```

### Arguments

diflist	The listing of results from aMNLFA.prune(), which contains the DIF tables (as well as impact tables, which aren't used here)
diftype	The type of DIF the user wants plot for. Options include "loading" (for loading DIF), "intercept" (for intercept DIF when threshold DIF is not tested), "threshold.highest" (which uses only the largest test statistic across all categories when threshold DIF is tested), and "threshold.all" (which uses the test statistic for all categories when threshold DIF is tested)
log	Logical. If TRUE, plot the y axis on a log scale. Defaults to FALSE.

### Value

No return value; generates a plot using base R.

**Examples**

```
wd <- tempdir()
first<-paste0(system.file(package='aMNLFA'),"/extdata")
the.list <- list.files(first,full.names=TRUE)
file.copy(the.list,wd,overwrite=TRUE)

ob <- aMNLFA::aMNLFA.object(dir = wd,
mrdata = xstudy,
indicators = paste0("BIN_", 1:12),
catindicators = paste0("BIN_", 1:12),
meanimpact = c("AGE", "GENDER", "STUDY"),
varimpact = c("AGE", "GENDER", "STUDY"),
measinvar = c("AGE", "GENDER", "STUDY"),
factors = c("GENDER", "STUDY"),
ID = "ID",
thresholds = FALSE)

prune.object <- aMNLFA.prune(ob)
aMNLFA.DIFplot(prune.object, "loading", log = FALSE)
```

---

aMNLFA.final

*aMNLFA simultaneous model fitting function*


---

**Description**

This function generates the simultaneous aMNLFA model from all the initial inputs.

**Usage**

```
aMNLFA.final(
  input.object,
  mchoice = "actual",
  method = "BH",
  highest.category = TRUE,
  keepmean = FALSE
)
```

**Arguments**

input.object	The aMNLFA object (created using the aMNLFA.object function) which provides instructions for the function.
mchoice	String representing the method of determining the number of tests, denoted m. Options include "actual", which uses the number of effects actually tested in the round 2 model as m, and "ibc", which uses the maximum number of all possible tests – i.e., the number of items times the number of covariates. Defaults to "actual".

method	String representing the method of adjusting for multiple comparisons. Options include "bh", which invokes Benjamini-Hochberg correction with <i>m</i> defined using the <i>mchoice</i> parameter, and "bonferroni", which invokes a Bonferroni correction with <i>m</i> defined using the <i>mchoice</i> parameter. Defaults to "bh".
highest.category	Boolean. If threshold DIF is tested, should only the category with the highest value of the test statistic be used when adjusting <i>p</i> . values? Defaults to TRUE, which corresponds to the results from "threshold.highest" in the <code>aMNLFA.prune()</code> step. If FALSE, all threshold effects will be considered, even those below the maximum value for a given item, which corresponds to the "thresholds.all" option in the <code>aMNLFA.prune()</code> step.
keepmean	Boolean. If intercept or loading DIF are present, should the corresponding mean impact effect be retained? Defaults to FALSE.

### Value

No return value. Generates a file entitled "round3calibration.inp", to be run in *Mplus*, in the directory specified in the `aMNLFA.object`.

### Examples

```
wd <- tempdir()
first <- paste0(system.file(package='aMNLFA'),"/extdata")
the.list <- list.files(first,full.names=TRUE)
file.copy(the.list,wd,overwrite=TRUE)

ob <- aMNLFA::aMNLFA.object(dir = wd,
  mrdta = xstudy,
  indicators = paste0("BIN_", 1:12),
  catindicators = paste0("BIN_", 1:12),
  meanimpact = c("AGE", "GENDER", "STUDY"),
  varimpact = c("AGE", "GENDER", "STUDY"),
  measinvar = c("AGE", "GENDER", "STUDY"),
  factors = c("GENDER", "STUDY"),
  ID = "ID",
  thresholds = FALSE)

aMNLFA.simultaneous(ob)
```

---

aMNLFA.initial

*aMNLFA initial model fitting function*

---

### Description

This function generates the initial itemwise aMNLFA models.

### Usage

```
aMNLFA.initial(input.object)
```

**Arguments**

`input.object` The aMNLFA object (created using the aMNLFA.object function) which provides instructions for the function.

**Examples**

```
wd <- tempdir()
first<-paste0(system.file(package='aMNLFA'),"/examplefiles")
the.list <- list.files(first,full.names=TRUE)
file.copy(the.list,wd,overwrite=TRUE)
ob <- aMNLFA::aMNLFA.object(dir = wd,
  mrdta = xstudy,
  indicators = paste0("BIN_", 1:12),
  catindicators = paste0("BIN_", 1:12),
  meanimpact = c("AGE", "GENDER", "STUDY"),
  varimpact = c("AGE", "GENDER", "STUDY"),
  measinvar = c("AGE", "GENDER", "STUDY"),
  factors = c("GENDER", "STUDY"),
  ID = "ID",
  thresholds = FALSE)

aMNLFA.initial(ob)
```

---

aMNLFA.itemplots

*aMNLFA item plotting function*


---

**Description**

This function generates plots of item endorsement by time, and by each covariate. This is necessary for determining which covariates to use in the MNLFA.

**Usage**

```
aMNLFA.itemplots(input.object)
```

**Arguments**

`input.object` The aMNLFA object (created using the aMNLFA.object function) which provides instructions for the function.

**Examples**

```
wd <- tempdir()
first<-paste0(system.file(package='aMNLFA'),"/examplefiles")
the.list <- list.files(first,full.names=TRUE)
file.copy(the.list,wd,overwrite=TRUE)
ob <- aMNLFA::aMNLFA.object(dir = wd,
```

```

mrdata = xstudy,
indicators = paste0("BIN_", 1:12),
catindicators = paste0("BIN_", 1:12),
meanimpact = c("AGE", "GENDER", "STUDY"),
varimpact = c("AGE", "GENDER", "STUDY"),
measinvar = c("AGE", "GENDER", "STUDY"),
factors = c("GENDER", "STUDY"),
ID = "ID",
thresholds = FALSE)

aMNLFA.itemplots(ob)

```

---

aMNLFA.object

*aMNLFA object function*


---

### Description

This function creates an aMNLFA object based on user specifications to pass to aMNLFA functions.

### Usage

```

aMNLFA.object(
  dir,
  mrdata,
  indicators = NULL,
  catindicators = NULL,
  countindicators = NULL,
  meanimpact = NULL,
  varimpact = NULL,
  measinvar = NULL,
  factors = NULL,
  time = NULL,
  auxiliary = NULL,
  ID = NULL,
  thresholds = NULL
)

```

### Arguments

<code>dir</code>	The directory in which data, inputs, and outputs are to be stored. Must be supplied.
<code>mrdata</code>	The R dataframe containing the multiple-record dataset. Must be supplied.
<code>indicators</code>	The names of all indicators (items, observed variables) in the MNLFA.
<code>catindicators</code>	The list of indicators which are categorical. Defaults to NULL.
<code>countindicators</code>	The list of indicators which are count. Defaults to NULL.

meanimpact	The list of covariates (predictors) which may generate impact on the latent variable mean. Defaults to NULL.
varimpact	The list of covariates (predictors) which may generate impact on the latent variable variance. Defaults to NULL.
measinvar	The list of covariates (predictors) which may generate DIF. Defaults to NULL.
factors	The list of covariates which are categorical. Defaults to NULL.
time	The variable which indexes time (or multiple records within a single case). If left blank, assumes single-record data. Defaults to NULL.
auxiliary	The list of variables to be considered as auxiliary (i.e., retained in the dataset but not used in the analysis). Defaults to NULL.
ID	The variable which identifies cases. Defaults to NULL.
thresholds	A Boolean operator indicating whether to test for threshold DIF.

### Examples

```
wd <- tempdir()
first<-paste0(system.file(package='aMNLFA'),"/examplefiles")
the.list <- list.files(first,full.names=TRUE)
file.copy(the.list,wd,overwrite=TRUE)

ob <- aMNLFA::aMNLFA.object(dir = wd,
  mrdata = xstudy,
  indicators = paste0("BIN_", 1:12),
  catindicators = paste0("BIN_", 1:12),
  meanimpact = c("AGE", "GENDER", "STUDY"),
  varimpact = c("AGE", "GENDER", "STUDY"),
  measinvar = c("AGE", "GENDER", "STUDY"),
  factors = c("GENDER", "STUDY"),
  ID = "ID",
  thresholds = FALSE)
```

---

aMNLFA.prune

*aMNLFA simultaneous model fitting function*


---

### Description

This function generates the simultaneous aMNLFA model from all the initial inputs.

### Usage

```
aMNLFA.prune(input.object)
```

### Arguments

input.object	The aMNLFA object (created using the aMNLFA.object function) which provides instructions for the function.
--------------	------------------------------------------------------------------------------------------------------------

**Value**

A list (entitled summary) with the following elements:

- `indicators` a list of indicators as specified by the user in the `aMNLFA.object()`
- `measinvar` a list of measurement invariance variables as specified by the user in the `aMNLFA.object()`
- `meanimpact` parameter values, standard errors, test statistics, and p. values for all mean impact effects tested in the simultaneous model
- `varimpact` parameter values, standard errors, test statistics, and p. values for all variance impact effects tested in the simultaneous model
- `loadingDIF` parameter values, standard errors, test statistics, and p. values for all loading DIF effects tested in the simultaneous model. Also includes critical values for different corrections according to the number of tests,  $m$ : Benjamini-Hochberg or Bonferroni with  $m$  defined as the actual number of tests included in the model (`BH.actual` and `bon.actual`, respectively); Benjamini-Hochberg or Bonferroni with  $m$  defined as the number of items times the number of covariates (`BH.abc` and `bon.abc`, respectively).
- `interceptDIF` If `thresholds = FALSE` in the corresponding `aMNLFA.object`: parameter values, standard errors, test statistics, and p. values for all intercept DIF effects tested in the simultaneous model. Also includes critical values for different corrections according to the number of tests,  $m$ : Benjamini-Hochberg or Bonferroni with  $m$  defined as the actual number of tests included in the model (`BH.actual` and `bon.actual`, respectively); Benjamini-Hochberg or Bonferroni with  $m$  defined as the number of items times the number of covariates (`BH.abc` and `bon.abc`, respectively).
- `tDIF_highest` If `thresholds = TRUE` in the corresponding `aMNLFA.object`: parameter values, standard errors, test statistics, and p. values for all threshold DIF effects tested in the simultaneous model, with tests performed only on the category with the largest test statistic for each item. Also includes critical values for different corrections according to the number of tests,  $m$ : Benjamini-Hochberg or Bonferroni with  $m$  defined as the actual number of tests included in the model (`BH.actual` and `bon.actual`, respectively); Benjamini-Hochberg or Bonferroni with  $m$  defined as the number of items times the number of covariates (`BH.abc` and `bon.abc`, respectively).
- `tDIF_all` If `thresholds = TRUE` in the corresponding `aMNLFA.object`: parameter values, standard errors, test statistics, and p. values for all threshold DIF effects tested in the simultaneous model, with tests performed on all categories for each item. Also includes critical values for different corrections according to the number of tests,  $m$ : Benjamini-Hochberg or Bonferroni with  $m$  defined as the actual number of tests included in the model (`BH.actual` and `bon.actual`, respectively); Benjamini-Hochberg or Bonferroni with  $m$  defined as the number of items times the number of covariates (`BH.abc` and `bon.abc`, respectively).

**Examples**

```
wd <- tempdir()
first<-paste0(system.file(package='aMNLFA'),"/extdata")
the.list <- list.files(first,full.names=TRUE)
file.copy(the.list,wd,overwrite=TRUE)

ob <- aMNLFA::aMNLFA.object(dir = wd,
mldata = xstudy,
```

```

indicators = paste0("BIN_", 1:12),
catindicators = paste0("BIN_", 1:12),
meanimpact = c("AGE", "GENDER", "STUDY"),
varimpact = c("AGE", "GENDER", "STUDY"),
measinvar = c("AGE", "GENDER", "STUDY"),
factors = c("GENDER", "STUDY"),
ID = "ID",
thresholds = FALSE)

aMNLFA.prune(ob)

```

---

aMNLFA.sample

*aMNLFA sampling function*


---

### Description

This function generates a single-record dataset using a random sample of time points from the multiple-record sample.

### Usage

```
aMNLFA.sample(input.object)
```

### Arguments

`input.object` The aMNLFA object (created using the aMNLFA.object function) which provides instructions for the function.

### Value

No return value. Generates a calibration data file in the directory specified in the aMNLFA.object.

### Examples

```

wd <- tempdir()
first<-paste0(system.file(package='aMNLFA'),"/extdata")
the.list <- list.files(first,full.names=TRUE)
file.copy(the.list,wd,overwrite=TRUE)

ob <- aMNLFA::aMNLFA.object(dir = wd,
mrdta = xstudy,
indicators = paste0("BIN_", 1:12),
catindicators = paste0("BIN_", 1:12),
meanimpact = c("AGE", "GENDER", "STUDY"),
varimpact = c("AGE", "GENDER", "STUDY"),
measinvar = c("AGE", "GENDER", "STUDY"),
factors = c("GENDER", "STUDY"),
ID = "ID",
thresholds = FALSE)

aMNLFA.sample(ob)

```

---

aMNLFA.scores	<i>aMNLFA score generating function</i>
---------------	-----------------------------------------

---

### Description

This function creates scores generated using aMNLFA. Can only be run after the final model has been fit – i.e., after the aMNLFA.final function.

### Usage

```
aMNLFA.scores(input.object)
```

### Arguments

`input.object` The aMNLFA object (created using the aMNLFA.object function) which provides instructions for the function.

### Value

No return value. Generates an INP file to be run in *Mplus* to generate scores in the directory specified in the aMNLFA.object.

### Examples

```
wd <- tempdir()
first<-paste0(system.file(package='aMNLFA'),"/extdata")
the.list <- list.files(first,full.names=TRUE)
file.copy(the.list,wd,overwrite=TRUE)

ob <- aMNLFA::aMNLFA.object(dir = wd,
  mrdta = xstudy,
  indicators = paste0("BIN_", 1:12),
  catindicators = paste0("BIN_", 1:12),
  meanimpact = c("AGE", "GENDER", "STUDY"),
  varimpact = c("AGE", "GENDER", "STUDY"),
  measinvar = c("AGE", "GENDER", "STUDY"),
  factors = c("GENDER", "STUDY"),
  ID = "ID",
  thresholds = FALSE)

aMNLFA.scores(ob)
```

---

aMNLFA.simultaneous     *aMNLFA simultaneous model fitting function*

---

## Description

This function generates the simultaneous aMNLFA model from all the initial inputs.

## Usage

```
aMNLFA.simultaneous(input.object, keepmean = FALSE)
```

## Arguments

input.object	The aMNLFA object (created using the aMNLFA.object function) which provides instructions for the function.
keepmean	Boolean. If intercept or loading DIF are present, should the corresponding mean impact effect be retained? Defaults to FALSE.

## Value

No return value. Generates a file entitled "round3calibration.inp", to be run in *Mplus*, in the directory specified in the aMNLFA.object.

## Examples

```
wd <- tempdir()
first <- paste0(system.file(package='aMNLFA'),"/extdata")
the.list <- list.files(first,full.names=TRUE)
file.copy(the.list,wd,overwrite=TRUE)

ob <- aMNLFA::aMNLFA.object(dir = wd,
  mrdta = xstudy,
  indicators = paste0("bin_", 1:12),
  catindicators = paste0("bin_", 1:12),
  meanimpact = c("AGE", "GENDER", "STUDY"),
  varimpact = c("AGE", "GENDER", "STUDY"),
  measinvar = c("AGE", "GENDER", "STUDY"),
  factors = c("GENDER", "STUDY"),
  ID = "ID",
  thresholds = FALSE)

aMNLFA.simultaneous(ob)
```

fixPath *helper function - removes the final slash at the end of a given string*

---

**Description**

This function generates the initial itemwise aMNLFA models.

**Usage**

```
fixPath(somecharacter)
```

**Arguments**

somecharacter The aMNLFA object (created using the aMNLFA.object function) which provides instructions for the function.

**Value**

outcharacter - string with slash at the end deleted

**Examples**

```
wd <- tempdir()
first<-paste0(system.file(package='aMNLFA'),"/examplefiles")
fixPath(first)
```

---

write.inp.file *helper function for writing out Mplus inputs*

---

**Description**

This function generates the initial itemwise aMNLFA models.

**Usage**

```
write.inp.file(df, outfile)
```

**Arguments**

df - the Mplus code to be written out, in a data frame  
outfile - a directory to which the Mplus code should be written

**Value**

outcharacter - string with slash at the end deleted

**Examples**

```
wd <- tempdir()
first<-paste0(system.file(package='aMNLFA'),"/examplefiles")
somestring <- as.data.frame("This is some text which would be written to an Mplus file.")
write.inp.file(somestring, first)
```

---

xstudy

*Simulated cross-study data*


---

**Description**

Data are simulated as part of a larger study (Curran et al., 2016; Curran et al., under review). Meant to simulate a dataset pooled across two studies, with 12 indicators and 3 moderators (age, gender, and study). Impact and DIF exist on the basis of these moderators.

**Usage**

```
data(xstudy)
```

**Format**

A data frame with 500 rows and 25 columns. The 25 variables are:

**ID** Unique identifier

**AGE** Age in years, centered around age 13

**GENDER** Effect-coded gender

**STUDY** Effect-coded study membership

**STUDYAGE** Interaction between age and study

**TRUEETA** True score on latent variable for each subject – not used in analysis

**STUDYETA** Interaction between study and score – not used in analysis

**ZETA** Deviation score – not used in analysis

**W** External covariate for original simulation – not used in analysis

**Z1** External outcome for original simulation – not used in analysis

**Z2** External outcome for original simulation – not used in analysis

**Z3** External outcome for original simulation – not used in analysis

**Z4** External outcome for original simulation – not used in analysis

**BIN\_1** Binary item 1

**BIN\_2** Binary item 2

**BIN\_3** Binary item 3

**BIN\_4** Binary item 4

**BIN\_5** Binary item 5

**BIN\_6** Binary item 6

- BIN\_7** Binary item 7
- BIN\_8** Binary item 8
- BIN\_9** Binary item 9
- BIN\_10** Binary item 10
- BIN\_11** Binary item 11
- BIN\_12** Binary item 12

**Source**

Curran et al., 2016 ([PubMed](#))

**References**

Curran et al., 2016 Structural Equation Modeling 23(6), 827-844. ([PubMed](#))

# Index

## \* MNLFA

- aMNLFA.DIFplot, [2](#)
- aMNLFA.final, [3](#)
- aMNLFA.initial, [4](#)
- aMNLFA.itemplots, [5](#)
- aMNLFA.object, [6](#)
- aMNLFA.prune, [7](#)
- aMNLFA.sample, [9](#)
- aMNLFA.scores, [10](#)
- aMNLFA.simultaneous, [11](#)
- fixPath, [12](#)
- write.inp.file, [12](#)

## \* datasets

- xstudy, [13](#)

- aMNLFA.DIFplot, [2](#)
- aMNLFA.final, [3](#)
- aMNLFA.initial, [4](#)
- aMNLFA.itemplots, [5](#)
- aMNLFA.object, [6](#)
- aMNLFA.prune, [7](#)
- aMNLFA.sample, [9](#)
- aMNLFA.scores, [10](#)
- aMNLFA.simultaneous, [11](#)

[fixPath, 12](#)

[write.inp.file, 12](#)

[xstudy, 13](#)