Package: FAfA (via r-universe)

October 29, 2026

Title Factor Analysis for All

Version 1.4

Date 2026-09-29

Description Provides a comprehensive Shiny-based graphical user interface for conducting a wide range of factor analysis procedures. 'FAfA' (Factor Analysis for All) guides users through data uploading, assumption checking (descriptives, collinearity, multivariate normality, outliers), data wrangling (variable exclusion, data splitting), factor retention analysis (e.g., Parallel Analysis, Hull method, EGA), Exploratory Factor Analysis (EFA) with various rotation and extraction methods, internal split-sample EFA replication analysis, Confirmatory Factor Analysis (CFA) for model testing, Reliability Analysis (e.g., Cronbach's Alpha, McDonald's Omega), Measurement Invariance testing across groups, and item weighting techniques. The application leverages established R packages such as 'lavaan' and 'psych' to perform these analyses, offering an accessible platform for researchers and students. Results are presented in user-friendly tables and plots, with options for downloading outputs. Analysis projects can be saved and restored, and reproducible R, HTML, and PDF workflow reports can be generated.

License AGPL-3

Copyright See file inst/COPYRIGHTS.

Depends R (>= 4.1.0)

URL https://github.com/AFarukKILIC/FAfA

BugReports https://github.com/AFarukKILIC/FAfA/issues

Imports Amelia, EFA.MRFA, EGAnet (>= 2.4.1), ItemRest, bsicons, bslib, ggplot2, golem, grDevices, graphics, haven, lavaan, mice, missForest, mvnormalTest, naniar, psych, qgraph, readxl, semPlot, shiny, shinycssloaders, stats, tools, utils

Suggests flextable, officer, shinytest2, spelling, testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

Config/roxygen2/version 8.1.0

Config/pak/sysreqs cmake libglpk-dev make libgsl0-dev libicu-dev libjpeg-dev libpng-dev libuv1-dev libxml2-dev libssl-dev libx11-dev zlib1g-dev

Repository https://afarukkilic.r-universe.dev

Date/Publication 2026-09-29 19:28:26 UTC

RemoteUrl https://github.com/afarukkilic/fafa

RemoteRef HEAD

RemoteSha e92c4a3c9a149a9ae60894e0d24cbe90ee87f408

Contents

about_server . . . 3
assumptions_server . . . 3
cfa_server . . . 4
cfa_ui . . . 5
data_selection_server . . . 5
efa_replication_analysis . . . 6
efa_replication_server . . . 7
efa_server_analysis . . . 7
efa_server_fac_ret . . . 8
efa_server_report . . . 8
inv_server . . . 9
item_weighting_server . . . 9
mod_itemrest_server . . . 10
mod_itemrest_ui . . . 10
mod_missing_server . . . 11
mod_missing_ui . . . 11
op_null_or . . . 12
reliability_server . . . 12
run_app . . . 12
wrangling_server_ex_var . . . 13

Index **14**

about_server *About Server Module*

Description

About Server Module

Usage

```
about_server(id)
```

Arguments

id	Module namespace ID.

assumptions_server *Assumptions Server Logic*

Description

Assumptions Server Logic

Usage

```
assumptions_server(id, data, error_recorder = NULL, language = NULL)
```

Arguments

id	Module namespace ID.
data	Input data (reactive).
error_recorder	Optional function used for anonymous diagnostics.
language	Optional reactive interface language.

`cfa_server` *Confirmatory Factor Analysis (CFA) Server Module*

Description

Confirmatory Factor Analysis (CFA) Server Module

Usage

```
cfa_server(
  id,
  data,
  factor_dictionary = NULL,
  error_recorder = NULL,
  language = NULL
)
```

Arguments

`id`	Module namespace ID.
`data`	Reactive containing the input dataset.
`factor_dictionary`	Shared reactive value containing factor-to-item mappings.
`error_recorder`	Optional function used for anonymous diagnostics.
`language`	Optional reactive interface language.

Details

The Dynamic Fit Index workflow called by this module is adapted from version 1.1.0 of the `dynamic` R package under the GNU Affero General Public License, version 3 (AGPL-3). The integration was rewritten for FAfA's existing lavaan analysis flow rather than copied verbatim.

References

McNeish, D., & Wolf, M. G. (2023). Dynamic fit index cutoffs for confirmatory factor analysis models. *Psychological Methods, 28*(1), 61-88. doi:10.1037/met0000425

cfa_ui *Confirmatory Factor Analysis (CFA) UI Module*

Description

Confirmatory Factor Analysis (CFA) UI Module

Usage

```
cfa_ui(id)
```

Arguments

id	Module namespace ID.

data_selection_server *Data Selection Server Logic*

Description

Data Selection Server Logic

Usage

```
data_selection_server(id, data, language = NULL)
```

Arguments

id	Module namespace ID.
data	Input data (reactive).
language	Reactive or character interface language ("en" or "tr").

`efa_replication_analysis` *Conduct Internal Replication Analysis in EFA*

Description

Randomly splits a data set into two halves, fits the same exploratory factor model in each half, aligns factor labels and signs, and compares item-level structure and loading magnitudes.

Usage

```
efa_replication_analysis(
  data,
  nfactors = 1L,
  rotate = "oblimin",
  fm = "minres",
  cor = "poly",
  seed = 2026L,
  volatility_cutoff = 0.04
)
```

Arguments

`data`	A data frame containing numeric analysis variables.
`nfactors`	Fixed number of factors extracted in both samples.
`rotate`	Rotation method passed to `psych::fa()`.
`fm`	Extraction method passed to `psych::fa()`.
`cor`	Correlation type: `"cor"` or `"poly"`.
`seed`	Non-negative integer used for the random split.
`volatility_cutoff`	Squared loading-difference threshold. Osborne and Fitzpatrick (2012) suggest `.04`, corresponding to an absolute loading difference of `.20`.

Value

A list containing both fitted models, aligned loading comparisons, factor alignment information, and replication decisions.

References

Osborne, J. W., & Fitzpatrick, D. C. (2012). Replication analysis in exploratory factor analysis: What it is and why it makes your analysis better. *Practical Assessment, Research, and Evaluation, 17*, Article 15. doi:10.7275/h0bd4d11

efa_replication_server *EFA Replication Analysis Server Module*

Description

Runs the internal split-sample replication procedure described by Osborne and Fitzpatrick (2012) and reports item-level structural and loading-magnitude comparisons.

Usage

```
efa_replication_server(id, data, error_recorder = NULL, language = NULL)
```

Arguments

id	Module namespace ID.
data	Input data (reactive).
error_recorder	Optional function used for anonymous diagnostics.
language	Optional reactive interface language.

References

Osborne, J. W., & Fitzpatrick, D. C. (2012). Replication analysis in exploratory factor analysis: What it is and why it makes your analysis better. *Practical Assessment, Research, and Evaluation, 17*, Article 15. doi:10.7275/h0bd4d11

efa_server_analysis *EFA Analysis Server Module*

Description

EFA Analysis Server Module

Usage

```
efa_server_analysis(id, data, error_recorder = NULL)
```

Arguments

id	Module namespace ID.
data	Input data (reactive).
error_recorder	Optional function used for anonymous diagnostics.

efa_server_fac_ret *EFA Factor Retention Server Module*

Description

EFA Factor Retention Server Module

Usage

```
efa_server_fac_ret(id, data, error_recorder = NULL)
```

Arguments

id	Module namespace ID.
data	Input data (reactive).
error_recorder	Optional function used for anonymous diagnostics.

efa_server_report *EFA Reporting Server Module*

Description

EFA Reporting Server Module

Usage

```
efa_server_report(
  id,
  data,
  efa_output_reactive,
  efa_settings_reactive,
  language = NULL
)
```

Arguments

id	Module namespace ID.
data	Input data (reactive).
efa_output_reactive	Reactive containing the EFA results.
efa_settings_reactive	Reactive containing the EFA settings.
language	Optional reactive interface language.

inv_server *Measurement Invariance Server Module*

Description

Measurement Invariance Server Module

Usage

```
inv_server(id, data, error_recorder = NULL, language = NULL)
```

Arguments

id	Module namespace ID.
data	Input data (reactive)
error_recorder	Optional function used for anonymous diagnostics.
language	Optional reactive interface language.

item_weighting_server *Item Weighting Server Module*

Description

Item Weighting Server Module

Usage

```
item_weighting_server(id, data, error_recorder = NULL)
```

Arguments

id	Module namespace ID.
data	Input data (reactive)
error_recorder	Optional function used for anonymous diagnostics.

mod_itemrest_server *ItemRest Analysis Server Module*

Description

ItemRest Analysis Server Module

Usage

```
mod_itemrest_server(id, data, error_recorder = NULL, language = NULL)
```

Arguments

id	Module namespace ID.
data	Reactive containing the input dataset.
error_recorder	Optional function used for anonymous diagnostics.
language	Optional reactive interface language.

mod_itemrest_ui *ItemRest Analysis UI Module*

Description

ItemRest Analysis UI Module

Usage

```
mod_itemrest_ui(id)
```

Arguments

id	Module namespace ID.

mod_missing_server *Missing Value Handling Server Module*

Description

Missing Value Handling Server Module

Usage

```
mod_missing_server(
  id,
  data,
  project_state = NULL,
  restore_state = NULL,
  error_recorder = NULL,
  language = NULL
)
```

Arguments

id	Module namespace ID.
data	Reactive containing the input dataset.
project_state	Optional shared project-state object.
restore_state	Optional reactive used while loading a project.
error_recorder	Optional function used for anonymous diagnostics.
language	Optional reactive interface language.

Value

Reactive containing the processed (imputed) dataset.

mod_missing_ui *Missing Value Handling UI Module*

Description

Missing Value Handling UI Module

Usage

```
mod_missing_ui(id)
```

Arguments

id	Module namespace ID.

op_null_or *Default Value Operator*

Description

Helper for default NULL values (coalesce)

Usage

```
x %||% y
```

Arguments

x	Left hand side
y	Right hand side

reliability_server *Reliability Analysis Server Module*

Description

Reliability Analysis Server Module

Usage

```
reliability_server(id, data, factor_dictionary = NULL, error_recorder = NULL)
```

Arguments

id	Module namespace ID.
data	Input data (reactive)
factor_dictionary	Shared reactive value containing factor-to-item mappings.
error_recorder	Optional function used for anonymous diagnostics.

run_app *Run the Shiny Application*

Description

This function launches the Shiny application. It uses default options for `shinyApp` and golem.

Usage

```
run_app()
```

wrangling_server_ex_var *Wrangling Server Modules*

Description

Wrangling Server Modules

Usage

```
wrangling_server_ex_var(
  id,
  data,
  project_state = NULL,
  restore_state = NULL,
  error_recorder = NULL
)
```

Arguments

id	Module namespace ID.
data	Input data (reactive)
project_state	Optional shared project-state object.
restore_state	Optional reactive used while loading a project.
error_recorder	Optional function used for anonymous diagnostics.

Index

about_server, 3
assumptions_server, 3

cfa_server, 4
cfa_ui, 5

data_selection_server, 5

efa_replication_analysis, 6
efa_replication_server, 7
efa_server_analysis, 7
efa_server_fac_ret, 8
efa_server_report, 8

inv_server, 9
item_weighting_server, 9

mod_itemrest_server, 10
mod_itemrest_ui, 10
mod_missing_server, 11
mod_missing_ui, 11

op_null_or, 12

psych::fa(), *6*

reliability_server, 12
run_app, 12

wrangling_server_ex_var, 13