

# Package: l1rotation (via r-universe)

May 19, 2026

**Title** Identify Loading Vectors under Sparsity in Factor Models

**Version** 1.0.2

**Description** Simplify the loading matrix in factor models using the l1 criterion as proposed in Freyaldenhoven (2025) [doi:10.21799/frbp.wp.2020.25](https://doi.org/10.21799/frbp.wp.2020.25). Given a data matrix, find the rotation of the loading matrix with the smallest l1-norm and/or test for the presence of local factors with main function `local_factors()`.

**License** MIT + file LICENSE

**Depends** R (>= 3.5)

**Imports** cli, doParallel, dplyr, foreach, ggplot2, magrittr, matrixStats, pracma, scales, stats

**Suggests** knitr, quarto, rmarkdown, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Encoding** UTF-8

**LazyData** true

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**URL** <https://kobleary.github.io/l1rotation/>,  
<https://github.com/SimonFreyaldenhoven/l1rotation>

**BugReports** <https://github.com/SimonFreyaldenhoven/l1rotation/issues>

**VignetteBuilder** knitr

**Config/Needs/website** quarto, rmarkdown

**Repository** <https://simonfreyaldenhoven.r-universe.dev>

**Date/Publication** 2025-08-21 17:34:24 UTC

**RemoteUrl** <https://github.com/simonfreyaldenhoven/l1rotation>

**RemoteRef** HEAD

**RemoteSha** 3cdf09b0d34f06f3b549cd8a3b9d730d5e6bd7a0

## Contents

example_data . . . . .	2
find_local_factors . . . . .	2
local_factors . . . . .	3
test_local_factors . . . . .	5

<b>Index</b>	<b>6</b>
--------------	----------

---

example_data	<i>Example data with two factors from the replication files of Freyaldenhoven (2025).</i>
--------------	---

---

### Description

Example data with two factors from the replication files of Freyaldenhoven (2025).

### Usage

example\_data

### Format

example\_data:  
A matrix with 224 rows and 207 columns.

### Source

Dataset available as a .mat file can be found under local\_factors.zip at <https://simonfreyaldenhoven.github.io/software/>

---

find_local_factors	<i>Find the rotation of the loading matrix with the smallest <math>l_1</math>-norm, as in <code>local_factors()</code>, with additional flexibility.</i>
--------------------	--

---

### Description

Find the most sparse rotation of an orthonormal basis of the loading space of a  $t$  by  $n$  matrix  $X$ . Additional flexibility with the `initial_loadings` argument allows the user to specify any orthonormal basis rather than defaulting to PCA.

### Usage

find\_local\_factors(X, r, initial\_loadings, parallel = FALSE, n\_cores = NULL)

**Arguments**

<code>X</code>	A (usually standardized) $t$ by $n$ matrix of observations.
<code>r</code>	An integer denoting the number of factors in $X$ .
<code>initial_loadings</code>	Matrix that represents an orthonormal basis of the loading space. If not supplied, PCA is used by default in this function and also in <code>local_factors</code> .
<code>parallel</code>	A logical denoting whether the algorithm should be run in parallel.
<code>n_cores</code>	An integer denoting how many cores should be used, if <code>parallel == TRUE</code> .

**Value**

Returns a list with the following components:

- `initial_loadings` Principal Component estimate of the loading matrix (if not supplied).
- `rotated_loadings` Matrix that is the rotation of the loading matrix that produces the smallest  $l_1$ -norm.
- `rotation_diagnostics` A list containing 3 components:
  - `R` Rotation matrix that when used to rotate `initial_loadings` produces the smallest  $l_1$ -norm.
  - `l1_norm` Vector of length  $r$  containing the value of the  $l_1$  norm each solution generates.
  - `sol_frequency` Vector of length  $r$  containing the frequency in the initial grid of each solution.

**Examples**

```
# Minimal example with 2 factors, where X is a 224 by 207 matrix
r <- 2
M <- nrow(example_data)
n <- ncol(example_data)

# Compute PCA estimates
basis <- svd(example_data / sqrt(M), nu = M, nv = n)
initial_loadings <- sqrt(n) * basis$v[, 1:r]

# Find minimum rotation using orthonormal basis initial_loadings
rotation_result <- find_local_factors(X = example_data, r = r, initial_loadings = initial_loadings)
```

---

<code>local_factors</code>	<i>Check whether local factors are present and find the rotation of the loading matrix with the smallest <math>l_1</math>-norm.</i>
----------------------------	---

---

**Description**

`local_factors` tests whether local factors are present and returns both the Principal Component estimate of the loadings and the rotation of the loadings with the smallest  $l_1$ -norm. It also produces graphical illustrations of the results.

**Usage**

```
local_factors(X, r, parallel = FALSE, n_cores = NULL)
```

**Arguments**

<code>X</code>	A (usually standardized) $t$ by $n$ matrix of observations.
<code>r</code>	An integer denoting the number of factors in $X$ .
<code>parallel</code>	A logical denoting whether the algorithm should be run in parallel.
<code>n_cores</code>	An integer denoting how many cores should be used, if <code>parallel == TRUE</code> .

**Value**

Returns a list with the following components:

- `has_local_factors` A logical equal to `TRUE` if local factors are present.
- `initial_loadings` Principal component estimate of the loading matrix.
- `rotated_loadings` Matrix that is the rotation of the loading matrix that produces the smallest l1-norm.
- `rotation_diagnostics` A list containing 3 components:
  - `R` Rotation matrix that when used to rotate `initial_loadings` produces the smallest l1-norm.
  - `l1_norm` Vector of length `r` containing the value of the l1 norm each solution generates.
  - `sol_frequency` Vector of length `r` containing the frequency in the initial grid of each solution.
- `pc_plot` Tile plot of the Principal Component estimate of the loading matrix.
- `rotated_plot` Tile plot of the l1-rotation of the loading matrix estimate.
- `small_loadings_plot` Plot of the number of small loadings for each column of the l1-rotation of the loading matrix estimate.

**Examples**

```
# Minimal example with 2 factors, where X is a 224 by 207 matrix
lf <- local_factors(X = example_data, r = 2)

# Visualize Principal Component estimate of the loadings
lf$pc_plot

# Visualize l1-rotation loadings
lf$pc_rotated_plot
```

---

test_local_factors	<i>Test for the presence of local factors, as in <code>local_factors()</code>, with additional flexibility.</i>
--------------------	---

---

### Description

Test for the presence of local factors, as in `local_factors()`, with additional flexibility.

### Usage

```
test_local_factors(X, r, loadings = NULL)
```

### Arguments

X	A (usually standardized) t by n matrix of observations.
r	An integer denoting the number of factors in X.
loadings	(optional) Matrix that represents a sparse basis of the loading space.

### Value

Returns a list with the following components:

- `has_local_factors` Logical equal to TRUE if local factors are present.
- `n_small` Integer denoting the number of small loadings in sparse rotation.
- `gamma_n` Integer denoting the critical value to compare `n_small` to.
- `h_n` Number denoting the cutoff used to determine which loadings are small.
- `loadings` Matrix that is the rotation of the loadings that produces the smallest l1-norm (if not supplied).

### Examples

```
# Minimal example with 2 factors, where X is a 224 by 207 matrix
r <- 2
M <- nrow(example_data)
n <- ncol(example_data)

# Find minimum rotation
rotation_result <- find_local_factors(X = example_data, r)

# Test if sparse basis has local factors
test_result <- test_local_factors(
  X = example_data,
  r = r,
  loadings = rotation_result$rotated_loadings
)

test_result$has_local_factors
```

# Index

## \* datasets

example\_data, 2

example\_data, 2

find\_local\_factors, 2

local\_factors, 3

local\_factors(), 2, 5

test\_local\_factors, 5