

# Package: sparvaride (via r-universe)

September 8, 2024

**Type** Package

**Title** Variance Identification in Sparse Factor Analysis

**Version** 0.1.0

**Description** This is an implementation of the algorithm described in Section 3 of Hosszejni and Frühwirth-Schnatter (2022) <[doi:10.48550/arXiv.2211.00671](https://doi.org/10.48550/arXiv.2211.00671)>. The algorithm is used to verify that the counting rule  $CR(r,1)$  holds for the sparsity pattern of the transpose of a factor loading matrix. As detailed in Section 2 of the same paper, if  $CR(r,1)$  holds, then the idiosyncratic variances are generically identified. If  $CR(r,1)$  does not hold, then we do not know whether the idiosyncratic variances are identified or not.

**License** GPL (>= 3)

**SystemRequirements** C++14

**Encoding** UTF-8

**BuildResaveData** best

**VignetteBuilder** knitr

**BugReports** <https://github.com/hdarjus/sparvaride/issues>

**URL** <https://hdarjus.github.io/sparvaride/>

**LazyData** true

**Depends** R (>= 4.1)

**Imports** Rcpp

**LinkingTo** RcppArmadillo, Rcpp

**RoxygenNote** 7.2.3

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

**Config/testthat/edition** 3

**Roxygen** list(markdown = TRUE)

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

**RemoteUrl** <https://github.com/hdarjus/sparvaride>

**RemoteRef** HEAD

**RemoteSha** 3f4a0e7bc91c7501976ef88cbcd3266038e6e3c2

## Contents

counting\_rule\_holds . . . . . 2

**Index** 4

counting\_rule\_holds    *Verify that the counting rule CR(r,1) holds*

### Description

This is an implementation of the algorithm described in Section 3 of Hosszejni and Fruehwirth-Schnatter (2022). The algorithm is used to verify that the counting rule CR(r,1) holds for the sparsity pattern of the transpose of a factor loading matrix. As detailed in Section 2 of the same paper, if CR(r,1) holds, then the idiosyncratic variances are generically identified. If CR(r,1) does not hold, then we do not know whether the idiosyncratic variances are identified or not.

### Usage

```
counting_rule_holds(delta)
```

### Arguments

delta            an  $m \times r$  matrix of 0s and 1s, where  $\text{delta}(i, j) == 1$  if and only if the  $i$ -th observation loads on the  $j$ -th factor

### Value

TRUE if CR(r,1) holds, FALSE otherwise

### References

Hosszejni and Fruehwirth-Schnatter (2022). "Cover It Up! Bipartite Graphs Uncover Identifiability in Sparse Factor Analysis". arXiv:2211.00671. doi:10.48550/arXiv.2211.00671

### See Also

[stats::factanal\(\)](#)

### Examples

```
# Two example matrices
cr_holds <-
  matrix(c(1, 0, 0,
           1, 0, 1,
           0, 1, 0,
           0, 1, 1,
           0, 1, 1,
           1, 1, 1,
           1, 1, 1),
```

```
7, 3, byrow = TRUE)

cr_does_not_hold <-
  matrix(c(1, 0, 0,
          0, 0, 1,
          0, 1, 0,
          0, 1, 0,
          0, 1, 0,
          1, 1, 1,
          1, 1, 1),
        7, 3, byrow = TRUE)

# Check if the counting rule holds
counting_rule_holds(cr_holds)
#> [1] TRUE
counting_rule_holds(cr_does_not_hold)
#> [1] FALSE
```

# Index

\* **factor analysis variance identification**

counting\_rule\_holds, [2](#)

\* **models**

counting\_rule\_holds, [2](#)

\* **multivariate**

counting\_rule\_holds, [2](#)

counting\_rule\_holds, [2](#)

stats::factanal(), [2](#)