Package: ssifs (via r-universe)

October 24, 2026

Type Package

Title Stochastic Search Inconsistency Factor Selection

Version 1.0.5

Author Georgios Seitidis [aut, cre] (ORCID: <https://orcid.org/0000-0003-0856-1892>), Stavros Nikolakopoulos [aut] (ORCID: <https://orcid.org/0000-0002-9769-3725>), Ioannis Ntzoufras [aut] (ORCID: <https://orcid.org/0000-0002-7615-0334>), Dimitris Mavridis [aut] (ORCID: <https://orcid.org/0000-0003-1041-4592>)

Maintainer Georgios Seitidis <g.seitidis@uoi.gr>

Description Evaluating the consistency assumption of Network Meta-Analysis both globally and locally in the Bayesian framework. Inconsistencies are located by applying Bayesian variable selection to the inconsistency factors. The implementation of the method is described by Seitidis et al. (2023) <doi:10.1002/sim.9891>.

License GPL (>= 3)

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Depends R (>= 2.10)

Imports ggplot2 (>= 3.3.6), gtools (>= 3.9.2.1), igraph (>= 1.3.1), meta (>= 8.0-1), netmeta (>= 3.0.2), plyr (>= 1.8.7), R2jags (>= 0.7.1), utils (>= 4.2.0), Rdpack (>= 2.3)

Suggests knitr, rmarkdown

VignetteBuilder knitr

URL https://github.com/georgiosseitidis/ssifs, https://georgiosseitidis.github.io/ssifs/

BugReports https://github.com/georgiosseitidis/ssifs/issues

NeedsCompilation no

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

Date/Publication 2025-06-24 15:40:02 UTC

RemoteUrl https://github.com/cran/ssifs

RemoteRef HEAD

RemoteSha 93a573e6cf7c408f2876957e31c36ec85b11eda4

Contents

Alcohol 2
smokingcessation 3
spike.slab 4
ssifs 5

Index **10**

`Alcohol` *Stochastic Search Inconsistency Factor Selection of brief alcohol interventions.*

Description

Stochastic Search Inconsistency Factor Selection for the evaluation of the consistency assumption for the network meta-analysis model.

These data are used as an example in Seitidis et al. (2021).

Format

A data frame with the following columns:

studyid	study id
treat1	treatment 1
treat2	treatment 2
m1	mean value of brief alcohol intervention in arm 1
m2	mean value of brief alcohol intervention in arm 2
n1	number of individuals in arm 1
n2	number of individuals in arm 2
sd1	standard deviation of brief alcohol intervention in arm 1
sd2	standard deviation of brief alcohol intervention in arm 2
TE	standardized mean difference of treat1 versus treat2
seTE	standard error of standardized mean difference

Source

Seitidis G, Nikolakopoulos S, Hennessy EA, Tanner-Smith EE, Mavridis D (2021): Network Meta-Analysis Techniques for Synthesizing Prevention Science Evidence *Prevention Science*, 1-10

Examples

```
data(Alcohol)

TE <- Alcohol$TE
seTE <- Alcohol$seTE
studlab <- Alcohol$studyid
treat1 <- Alcohol$treat2
treat2 <- Alcohol$treat1

# Stochastic Search Inconsistency Factor Selection using as reference treatment AO-CT and the
# design-by-treatment method for the specification of the Z matrix.

m <- ssifs(TE, seTE, treat1, treat2, studlab, ref = "AO-CT",
M = 1000, B = 100, M_pilot = 1000, B_pilot = 100)
```

`smokingcessation` *Stochastic Search Inconsistency Factor Selection of interventions for smoking cessation*

Description

Stochastic Search Inconsistency Factor Selection for the evaluation of the consistency assumption for the network meta-analysis model.

These data are used as an example in Dias et al. (2013).

Format

A data frame with the following columns:

event1	number of individuals with successful smoking cessation in arm 1
n1	number of individuals in arm 1
event2	number of individuals with successful smoking cessation in arm 2
n2	number of individuals in arm 2
event3	number of individuals with successful smoking cessation in arm 3
n3	number of individuals in arm 3
treat1	treatment 1
treat2	treatment 2
treat3	treatment 3

Source

Dias S, Welton NJ, Sutton AJ, Caldwell DM, Lu G and Ades AE (2013): Evidence Synthesis for Decision Making 4: Inconsistency in networks of evidence based on randomized controlled trials. *Medical Decision Making*, **33**, 641–56

Examples

```
data(smokingcessation)

# Transform data from arm-based format to contrast-based format

smokingcessation$id <- 1:dim(smokingcessation)[1]
smoking.pair <- meta::pairwise(
  treat = list(treat1, treat2, treat3),
  event = list(event1, event2, event3),
  n = list(n1, n2, n3),
  studlab = id,
  data = smokingcessation,
  sm = "OR"
)

TE <- smoking.pair$TE
seTE <- smoking.pair$seTE
studlab <- smoking.pair$studlab
treat1 <- smoking.pair$treat1
treat2 <- smoking.pair$treat2

# Stochastic Search Inconsistency Factor Selection using as reference treatment A and the
# design-by-treatment method for the specification of the Z matrix.

m <- ssifs(TE, seTE, treat1, treat2, studlab, ref = "A",
M = 1000, B = 100, M_pilot = 1000, B_pilot = 100)
```

---

spike.slab *Inconsistency Factors' Spike and Slab*

Description

The function visualizes the inconsistency factor's effect when the inconsistency factor is included in the Network Meta-Analysis (NMA) model and when is not.

Usage

```
spike.slab(x)
```

Arguments

x	An object of class `ssifs`.

Details

The function creates two density plots for each inconsistency factor based on the inconsistency factors' effects, which are obtained from the `ssifs` model. The former visualizes the effect when the inconsistency factor is included in the NMA model (spike), while the latter when is not (slab).

A good mixing of the SSIFS model indicates that the spike has high density for values close to zero whereas the slab is flatter.

Value

An object of class ggplot.

Examples

```
data(Alcohol)

TE <- Alcohol$TE
seTE <- Alcohol$seTE
studlab <- Alcohol$studyid
treat1 <- Alcohol$treat2
treat2 <- Alcohol$treat1

# Stochastic Search Inconsistency Factor Selection using intervention AO-CT as reference.
m <- ssifs(TE, seTE, treat1, treat2, studlab, ref = "AO-CT",
M = 500, B = 100, M_pilot = 300, B_pilot = 100)
spike.slab(m)
```

---

ssifs *Stochastic Search Inconsistency Factor Selection*

Description

Stochastic Search Inconsistency Factor Selection evaluates the consistency assumption of Network Meta-Analysis in the Bayesian framework, by treating the inconsistency detection as a variable selection problem. The consistency assumption is evaluated locally, but also globally.

Usage

```
ssifs(
  TE,
  seTE,
  treat1,
  treat2,
  studlab,
  ref,
  method = "DBT",
  rpcons = TRUE,
  pcons = 0.5,
  zellner = TRUE,
  c = 3,
  psi = NULL,
```

```
  digits = 4,
  M = 50000,
  B = 10000,
  n_thin = 1,
  n_chains = 2,
  M_pilot = 10000,
  B_pilot = 2000,
  n_thin_pilot = 1,
  n_chains_pilot = 1
)
```

Arguments

TE	Estimate of treatment effect (e.g. log odds ratio, mean difference, or log hazard ratio).
seTE	Standard error of the treatment estimate.
treat1	Label/Number of the first treatment.
treat2	Label/Number of the second treatment.
studlab	Study labels.
ref	Reference treatment.
method	Method used for the specification of the inconsistency factors; Possible choices are: • "LuAdes" for the Lu & Ades model (Lu & Ades, 2006) • "DBT" for the design-by-treatment method (Higgins et al., 2012) • "Jackson" for the random-effects implementation of the design-by-treatment model (Jackson et al., 2014)
rpcons	"logical". If TRUE, an informative beta distribution Beta(157, 44) is used for the probability to have a consistent network.
pcons	Probability to have a consistent network.
zellner	"logical". If TRUE, Zellner g-prior is used for the dependency between the inconsistency factors. If FALSE, inconsistency factors assumed independent.
c	Tuning parameter.
psi	Tuning parameter.
digits	Digits of the exported results.
M	Number of NMA MCMC iterations.
B	Burn-in period of the NMA MCMC run.
n_thin	Thinning interval of the NMA MCMC run.
n_chains	Number of parallel chains for the NMA MCMC run.
M_pilot	Number of pilot MCMC iterations.
B_pilot	Burn-in period of the pilot MCMC run.
n_thin_pilot	Thinning interval of the pilot MCMC run.
n_chains_pilot	Number of parallel chains for the pilot MCMC run.

Details

Stochastic Search Inconsistency Factor Selection (SSIFS) is the extension of Stochastic Search Variable Selection (SSVS) (George & McCulloch, 1993) for identifying inconsistencies in Network Meta-Analysis (NMA).

SSIFS (Seitidis et al., 2023), is a two-step method in which the inconsistency factors are specified in the first step, and in the second step, SSVS is performed on the inconsistency factors. The method used for the specification of the inconsistency factors, is controlled by the argument `method`. Among the choices that may be considered are the Lu and Ades model (Lu & Ades, 2006), the design-by-treatment model (Higgins et al., 2012), and the random-effects implementation of the design-by-treatment model (Jackson et al., 2014).

After specifying the inconsistency factors, the random-effects NMA model is implemented in the Bayesian framework using the `R2jags` package. An uninformative normal is assumed for the prior distribution of the treatment effects, while for the heterogeneity parameter tau, an uninformative half-normal is assumed. The function provides the MCMC run of the NMA model (item `MCMC_run`), whereby the user can check the convergence of the MCMC run.

SSIFS by default assumes that inconsistency factors are dependent by using a Zellner g-prior to describe this dependency (`zellner = TRUE`). Parameter g in the Zellner g-prior is specified using the unit information criterion (Kass & Wasserman, 1995), which is translated in SSIFS to the total number of observed comparisons that are included in the network. By setting the argument `zellner = FALSE`, inconsistency factors are assumed independent. Regarding the inclusion probabilities, the function by default assumes an informative Beta distribution (Beta(157, 44)) for the probability to have a consistent network (`rpcons = TRUE`). In the case where `rpcons = FALSE`, this probability is assumed fixed and equal to 0.5 (`pcons = 0.5`). The user can modify this probability through the argument `pcons`.

Tuning parameters in SSIFS are specified by the arguments `c` and `psi`. They should be specified in a way that, when an inconsistency factor is included in the NMA model, the corresponding coefficient lies in an area close to zero, and far away from this area when it is not included in the NMA model. Regarding the argument `c`, values between 10 and 100 usually perform well in most cases. Argument `psi` can be obtained either from a pilot MCMC run of the NMA model, as the standard deviation of the inconsistency factors (`psi = NULL`), or can be set fixed a-priory by the analyst.

In order to evaluate the consistency assumption, we can examine

- the posterior inclusion probabilities of the inconsistency factors (item `Posterior_inclusion_probabilities`)
- the posterior model probabilities (item `Posterior_Odds`)
- the posterior model odds (item `Posterior_Odds`)
- the Bayes factor of the consistent NMA model over the inconsistent NMA models (item `Bayes_Factor`)

A posterior inclusion probability above 0.5 indicates inconsistency. Also, an inconsistent NMA model with large posterior model probability suggests the presence of inconsistency. Item `Bayes_Factor` provides a global test for testing the consistency assumption, by calculating the Bayes factor of the consistent NMA model (model without inconsistency factors) over the rest inconsistent NMA models that were observed in the MCMC run. An estimate above 1 favors the consistent NMA model. For the calculation of the Bayes factor, the inconsistent NMA models are treated as a single model and the corresponding posterior model probabilities are summed.

Value

A list containing the following components:

`MCMC_run` An object of class `rjags` containing the MCMC run of the NMA model.

`Bayes_Factor` Bayes factor of the consistent NMA model over the inconsistent NMA model.

`Posterior_inclusion_probabilities`
A `data.frame` containing the posterior inclusion probabilities of the inconsistency factors. Columns `Comparison` and `Design` denote in which comparisons inconsistency factors are added. When argument `method = "LuAdes"`, column `Design` is `NA`, because only loop inconsistencies are accounted. `PIP` denotes the estimated posterior inclusion probability, `b` the estimated median effect of the inconsistency factors, `b.lb` and `b.ub` the lower and upper bounds of the inconsistency factors' effect estimates, respectively.

`Posterior_Odds` A `data.frame` containing the model posterior odds. Column `IFs` denotes in which comparisons inconsistency factors are added, `Freq` the number of times the model is observed in the MCMC run, `f(m|y)` the posterior model probability and `PO_IFCONS` the posterior model odds of the consistent NMA model (denoted as `NO IFs`) over the corresponding inconsistent NMA model.

`Summary` A `data.frame` containing the summary estimates of the MCMC run of the NMA model.

`Z_matrix` A `data.frame` containing in the first 3 columns the treatment comparisons used for the Z matrix and the Z matrix in the rest columns.

`disconnected_studies`
A vector with the studies that were excluded in order to have a connected network.

`n_subnetworks` Number of sub-networks.

`subnetworks` A list with the sub-networks of the original NMA data.

Note

The function uses the random-effects inverse-variance NMA model, and assumes common heterogeneity between different treatment comparisons and no correlation between different studies. Additionally, when the network is disconnected, the function keeps only those studies that belong to the largest sub-network.

In a multi-arm study with T comparisons, T-1 are required for the NMA model since the rest are obtained as a linear combination. The function automatically excludes the unnecessary comparisons, while maintaining the basic comparisons (if possible) and comparisons in which an inconsistency factor has been added. Therefore, all possible comparisons of multi-arm studies must be provided by the user.

For the names of the inconsistency factor, treatments are separated by the symbol `" ; "`. For example, if an inconsistency factor is added in the comparison between treatments A and B, the inconsistency factor name will be `A ; B`. In the case where `method = "DBT"` or `method = "Jackson"`, inconsistency factors' names for multi-arm designs are denoted as *treatment.comparison_design*. Thus, if an inconsistency factor is added in the comparison between treatments A and B of the ABC design, the inconsistency factor name will be `A ; B_ABC`.

In extremely large networks, the number of paths between two nodes may be exponentially high. If your network is lattice-like, you may run out of memory during the specification of the inconsistency factors with the Lu and Ades model.

References

George, E. I., & McCulloch, R. E. (1993): Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, **88**(423), 881-889.

Seitidis, G., Nikolakopoulos, S., Ntzoufras, I., & Mavridis, D. (2023): Inconsistency identification in network meta-analysis via stochastic search variable selection. *Statistics in Medicine*, **42**(26), 4850–4866.

Lu, G., & Ades, A. E. (2006): Assessing evidence inconsistency in mixed treatment comparisons. *Journal of the American Statistical Association*, **101**(474), 447-459.

Higgins, J. P. T., Jackson, D., Barrett, J. K., Lu, G., Ades, A. E., & White, I. R. (2012): Consistency and inconsistency in network meta-analysis: concepts and models for multi-arm studies. *Research synthesis methods*, **3**(2), 98-110.

Jackson, D., Barrett, J. K., Rice, S., White, I. R., & Higgins, J. P. (2014): A design-by-treatment interaction model for network meta-analysis with random inconsistency effects. *Statistics in medicine*, **33**(21), 3639-3654.

Kass, R. E., & Wasserman, L. (1995): A Reference Bayesian Test for Nested Hypotheses and its Relationship to the Schwarz Criterion. *Journal of the American Statistical Association*, **90**(431), 928–934.

Examples

```
data(Alcohol)

TE <- Alcohol$TE
seTE <- Alcohol$seTE
studlab <- Alcohol$studyid
treat1 <- Alcohol$treat2
treat2 <- Alcohol$treat1

# Stochastic Search Inconsistency Factor Selection using intervention AO-CT as reference.
m <- ssifs(TE, seTE, treat1, treat2, studlab, ref = "AO-CT", M=1000, B=100)
```

Index

Alcohol, 2

smokingcessation, 3
spike.slab, 4
ssifs, 5