Package: MetaStudies (via r-universe)

September 6, 2024

Type Package

Title Shiny app and function for Meta studies following Andrews and Kasy (2019).

Version 0.1.0

Author Maximilian Kasy, Sebastian Kranz

Maintainer Sebastian Kranz <sebastian.kranz@uni-ulm.de>

Description This package mainly consists of Maximilan Kasy's code of his shiny app for MetaStudies on publication bias. I added some stuff, rewrote it a bit and put everything into an R package. The original code is here: https://github.com/maxkasy/MetaStudiesApp The code is based on Andrews and Kasy (2019). In particular, I added code to run misspecification tests that will be explained in Kranz and Pütz (2021). References: - Andrews, Isaiah and Maximilian Kasy. 2019. "Identification of and correction for publication bias." American Economic Review 109 (8): 2766-94. - Kranz, Sebastian and Peter Pütz. 2021 "Rounding and other pitfalls in meta-studies on p-hacking and publication bias. A comment on Brodeur et al. (2020)", working paper.

License Attribution-NonCommercial-ShareAlike 4.0 https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode

Encoding UTF-8

Depends ggplot2, reshape, shiny, dplyr

Suggests parallel

LazyData true

RoxygenNote 7.1.1

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

RemoteUrl https://github.com/skranz/MetaStudies

RemoteRef main

RemoteSha c0715f99b9aa4ecdc7a4e0ef5b0e7834349dfbd1

Contents

`bootstrap_specification_tests` *Call metastudy_X_sigma_cors repeatedly with new bootstrap samples to compute standard errors for correlations using the inverse probability weighting approach*

Description

Note that this procedure can take very long to run.

Usage

```
bootstrap_specification_tests(X, sigma, B = 10, ..., num.cores = 1)
```

Arguments

`X`	vector of parameters
`sigma`	vector of standard errors
`B`	number of bootstrap replications
`...`	parameters passed to `metastudies_estimation`
`num.cores`	number of cores used for parallel computation using `mclapply`.

`estimates_plot` *Plot*

Description

Plot

Usage

```
estimates_plot(
  ms,
  X = ms$X,
  sigma = ms$sigma,
  cutoffs = ms$cutoffs,
  symmetric = ms$symmetric,
  model = ms$model
)
```

MetaStudiesApp *Create and run the MetaStudies shiny app*

Description

Create and run the MetaStudies shiny app

Usage

```
MetaStudiesApp(csv.file = NULL, show.cor = TRUE, ...)
```

Arguments

csv.file	if not NULL a file that will be initially used by the app
show.cor	if TRUE add a tabPanel with correlations between estimates and standard errors to get better insights whether the independence assumption of Andrews & Kasy (2019) may be violated in the given data set.
...	additional parameters passed to shinyApp

---

metastudies_estimation *Perform Andrews and Kasy (2019) estimation*

Description

Perform Andrews and Kasy (2019) estimation

Usage

```
metastudies_estimation(
  X,
  sigma,
  cutoffs = c(1.96),
  symmetric = FALSE,
  model = "normal",
  eval.max = 10^5,
  iter.max = 10^5,
  abs.tol = 10^(-8),
  stepsize = 10^(-6)
)
```

Arguments

X	vector of reported coefficients
sigma	vector of reported stanard errors
cutoffs	significance thresholds that define intervals of different publication probabilities, e.g. c(1.645, 1.96, 2.576).
symmetric	if TRUE assume that positive and negative z-statistics are symmetrically distributed.
model	either "normal" or "t". The assumed functional form of the distribution absent publication bias.
eval.max, iter.max, abs.tol	Control parameters for [stats::nlminb].

---

metastudy_X_sigma_cors

Computes correlations to test the independece assumption between estimate and standard error of Andrews and Kasy (2019)

Description

A crucial assumption of Andrews and Kasy (2019) is that in the unobserved latent distribution without publication error the estimate and its standard error are statistically independent from each other.

Usage

```
metastudy_X_sigma_cors(ms)
```

Arguments

ms	An object returned from the function metastudies_estimation

Details

While the latent distribution cannot be observed, this function computes some correlations that may indicate problems with respect to this assumption. We use an inverse probability weighting approach. More precisely, it weights inversely with the estimated publication probabilities to recover the correlation in the unobserved latent distribution. (This approach was suggested in an email by Isaiah Andrews and implemented by Sebastian Kranz).

To compute standard errrors via bootstrap (very time consuming), call the function `bootstrap_specification_tests`.

Value

a data frame with correlations, confidence intervals and also relevent results from a linear regression of standard errors on estimates.

Index