Package: phacking (via r-universe)

October 29, 2026

Title Sensitivity Analysis for p-Hacking in Meta-Analyses

Version 0.2.1

Description Fits right-truncated meta-analysis (RTMA), a bias correction for the joint effects of p-hacking (i.e., manipulation of results within studies to obtain significant, positive estimates) and traditional publication bias (i.e., the selective publication of studies with significant, positive results) in meta-analyses [see Mathur MB (2022). ``Sensitivity analysis for p-hacking in meta-analyses." <doi:10.31219/osf.io/ezjsx>.]. Unlike publication bias alone, p-hacking that favors significant, positive results (termed ``affirmative") can distort the distribution of affirmative results. To bias-correct results from affirmative studies would require strong assumptions on the exact nature of p-hacking. In contrast, joint p-hacking and publication bias do not distort the distribution of published nonaffirmative results when there is stringent p-hacking (e.g., investigators who hack always eventually obtain an affirmative result) or when there is stringent publication bias (e.g., nonaffirmative results from hacked studies are never published). This means that any published nonaffirmative results are from unhacked studies. Under these assumptions, RTMA involves analyzing only the published nonaffirmative results to essentially impute the full underlying distribution of all results prior to selection due to p-hacking and/or publication bias. The package also provides diagnostic plots described in Mathur (2022).

License MIT + file LICENSE

URL https://github.com/mathurlabstanford/phacking, https://mathurlabstanford.github.io/phacking/

BugReports https://github.com/mathurlabstanford/phacking/issues

Encoding UTF-8

RoxygenNote 7.2.3

RdMacros Rdpack

Biarch true

Depends R (>= 4.1.0)

Imports dplyr, ggplot2, metabias, metafor, methods, purrr, rlang, stats, stats4, truncnorm, Rcpp (>= 0.12.0), RcppParallel (>= 5.0.1), Rdpack, rstan (>= 2.18.1), rstantools (>= 2.2.0)

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), StanHeaders (>= 2.18.0)

SystemRequirements GNU make

LazyData true

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation yes

Author Peter Solymos [cre, ctb] (<https://orcid.org/0000-0001-7337-1740>), Maya Mathur [aut], Mika Braginsky [aut]

Maintainer Peter Solymos <peter@analythium.io>

Config/pak/sysreqs make

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

Date/Publication 2023-07-17 15:31:37 UTC

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

RemoteRef HEAD

RemoteSha bd4474f66b2d5972f8483cf2acd828e9dfbf6ad3

Contents

money_priming_meta 2
phacking_meta 3
rtma_cdf 5
rtma_qqplot 5
z_density 6

Index **8**

`money_priming_meta` *Meta-analysis of money priming studies*

Description

Dataset from a meta-analysis of experimental studies on the effect of money primes on a variety of psychological and behavioral outcomes, in which some studies were preregistered (Lodder et al. 2019).

Usage

```
money_priming_meta
```

Format

A data frame with 287 rows and 4 variables:

study Code identifying the study

yi Point estimate on the Hedges' *g* scale

vi Variance of point estimate

zi Z-score

preregistered Logical indicating whether study was preregistered

References

Lodder P, Ong HH, Grasman RPPP, Wicherts JM (2019). "A comprehensive meta-analysis of money priming." *Journal of Experimental Psychology: General*, **148**(4), 688.

Lodder P, Ong HH, Grasman RPPP, Wicherts JM (2020). "A comprehensive meta-analysis of money priming." OSF. https://osf.io/dhp63.

---

phacking_meta *Right-truncated meta-analysis*

Description

Fits right-truncated meta-analysis (RTMA), a bias correction for the joint effects of p-hacking (i.e., manipulation of results within studies to obtain significant, positive estimates) and traditional publication bias (i.e., the selective publication of studies with significant, positive results) in meta-analyses. This method analyzes only nonaffirmative studies (i.e., those with significant, positive estimates). You can pass all studies in the meta-analysis or only the nonaffirmative ones; if the former, the function will still analyze only the nonaffirmative ones.

Usage

```
phacking_meta(
  yi,
  vi,
  sei,
  favor_positive = TRUE,
  alpha_select = 0.05,
  ci_level = 0.95,
  stan_control = list(adapt_delta = 0.98, max_treedepth = 20),
  parallelize = TRUE
)
```

Arguments

yi	A vector of point estimates to be meta-analyzed.
vi	A vector of estimated variances (i.e., squared standard errors) for the point estimates.
sei	A vector of estimated standard errors for the point estimates. (Only one of vi or sei needs to be specified).
favor_positive	TRUE if publication bias are assumed to favor significant positive estimates; FALSE if assumed to favor significant negative estimates.
alpha_select	Alpha level at which an estimate's probability of being favored by publication bias is assumed to change (i.e., the threshold at which study investigators, journal editors, etc., consider an estimate to be significant).
ci_level	Confidence interval level (as proportion) for the corrected point estimate. (The alpha level for inference on the corrected point estimate will be calculated from ci_level.)
stan_control	List passed to rstan::sampling() as the control argument.
parallelize	Logical indicating whether to parallelize sampling.

Value

An object of class metabias::metabias(), a list containing:

data A tibble with one row per study and the columns yi, vi, sei, affirm.

values A list with the elements favor_positive, alpha_select, ci_level, tcrit, k, k_affirmative, k_nonaffirmative, optim_converged. optim_converged indicates whether the optimization to find the posterior mode converged.

stats A tibble with two rows and the columns param, mode, median, mean, se, ci_lower, ci_upper, n_eff, r_hat. We recommend reporting the mode for the point estimate; median and mean represent posterior medians and means respectively.

fit A stanfit object (the result of fitting the RTMA model).

References

Mathur MB (2022). "Sensitivity analysis for p-hacking in meta-analyses." doi:10.31219/osf.io/ezjsx.

Examples

```
# passing all studies, though only nonaffirmative ones will be analyzed
money_priming_rtma <- phacking_meta(money_priming_meta$yi, money_priming_meta$vi,
                                    parallelize = FALSE)
```

rtma_cdf — *Compute theoretical and empirical CDFs for a right-truncated meta-analysis*

Description

Compute theoretical and empirical CDFs for a right-truncated meta-analysis

Usage

```
rtma_cdf(rtma)
```

Arguments

rtma	Output of phacking_meta().

Value

A tibble with the columns yi (effect sizes), cdfi (their fitted CDF) and ecdfi (their empirical CDF).

A data frame with the CDF derived from a metabias object.

References

Mathur MB (2022). "Sensitivity analysis for p-hacking in meta-analyses." doi:10.31219/osf.io/ezjsx.

Examples

```
money_priming_rtma <- phacking_meta(money_priming_meta$yi,
                                    money_priming_meta$vi,
                                    parallelize = FALSE)
rtma_cdf(money_priming_rtma)
```

rtma_qqplot — *Diagnostic quantile-quantile plot for a right-truncated meta-analysis*

Description

To assess the fit of right-truncated meta-analysis and possible violations of its distributional assumptions, plots the fitted cumulative distribution function (CDF) of the published nonaffirmative estimates versus their empirical CDF. If the points do not adhere fairly closely to a 45-degree line, the right-truncated meta-analysis may not fit adequately.

Usage

```
rtma_qqplot(rtma)
```

Arguments

rtma	Output of phacking_meta().

Value

A ggplot2 object representing quantile-quantile plot.

Examples

```
money_priming_rtma <- phacking_meta(money_priming_meta$yi,
                                    money_priming_meta$vi,
                                    parallelize = FALSE)
rtma_qqplot(money_priming_rtma)
```

---

z_density *Z-score density plot*

Description

Plots the Z-scores of all published point estimates. When p-hacking favors affirmative estimates over nonaffirmative estimates, as our methods and others assume, Z-scores may disproportionately concentrate just above the critical value (e.g., 1.96). Importantly, the presence of p-hacking does not *guarantee* a concentration of Z-scores just above the critical value, so it is prudent to proceed with the fitting RTMA even if no such concentration is apparent. In contrast, if Z-scores also concentrate just *below* the critical value, or if they also concentrate below the sign-reversed critical value (e.g., -1.96), this could indicate forms of p-hacking that violate the assumptions of RTMA.

Usage

```
z_density(yi, vi, sei, alpha_select = 0.05, crit_color = "red")
```

Arguments

yi	A vector of point estimates to be meta-analyzed.
vi	A vector of estimated variances (i.e., squared standard errors) for the point estimates.
sei	A vector of estimated standard errors for the point estimates. (Only one of vi or sei needs to be specified).
alpha_select	Alpha level at which an estimate's probability of being favored by publication bias is assumed to change (i.e., the threshold at which study investigators, journal editors, etc., consider an estimate to be significant).
crit_color	Color for line and text are critical z-score.

Value

A ggplot2 object representing a Z-score density plot.

Examples

```
z_density(money_priming_meta$yi, money_priming_meta$vi)
```

Index

∗ **datasets**
money_priming_meta, 2

metabias::metabias(), *4*
money_priming_meta, 2

phacking_meta, 3
phacking_meta(), *5, 6*

rstan::sampling(), *4*
rtma_cdf, 5
rtma_qqplot, 5

z_density, 6