Package ‘mhtopt’

July 29, 2026

Title Optimal Multiple Hypothesis Testing Corrections

Version 1.0.0

Description Implements the optimal multiple hypothesis testing correction from Viviano, Wuthrich, and Niehaus (2026) <doi:10.48550/arXiv.2104.13367>. Derives the optimal per-test significance level from the economic incentives of research production, providing a correction that lies between Bonferroni (too conservative) and unadjusted (too permissive). Supports two cost models: a Linear one calibrated to United States Food and Drug Administration (FDA) clinical-trial costs, and a Cobb-Douglas one calibrated to Abdul Latif Jameel Poverty Action Lab (J-PAL) project costs. Reports optimal, Bonferroni, Holm, Benjamini-Hochberg (BH), and unadjusted results side by side.

License MIT + file LICENSE

URL https://github.com/dviviano/mhtopt

BugReports https://github.com/dviviano/mhtopt/issues

Encoding UTF-8

Language en-US

RoxygenNote 7.3.2

Imports stats

Suggests testthat (>= 3.0.0), haven, fixest, estimatr, knitr, rmarkdown

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Author Davide Viviano [aut],
Kaspar Wuthrich [aut],
Paul Niehaus [aut],
Erick Rosas Lopez [cre, ctb]

Maintainer Erick Rosas Lopez <erosaslpez@ucsd.edu>

Repository CRAN

Date/Publication 2026-07-29 16:40:03 UTC

Contents

`mht_cost_estimate` *Estimate Cost Function Parameters for MHT Adjustment*

Description

Estimates the parameters of the research cost function from data on project costs, number of treatment arms, and sample sizes. Implements the estimation approach from Section 6 and Appendix A of Viviano, Wuthrich, and Niehaus (2026).

Usage

```
mht_cost_estimate(
  cost,
  arms,
  sample_size,
  alpha_bar,
  model = c("cobbdouglas", "linear_share"),
  controls = NULL,
  robust = TRUE
)
```

Arguments

`cost`	Numeric vector. Total project/trial costs.
`arms`	Numeric vector. Number of treatment arms in each study.
`sample_size`	Numeric vector. Sample size (total or per arm) in each study.
`alpha_bar`	Numeric. Benchmark single-hypothesis test size.
`model`	Character. `"cobbdouglas"` (default) estimates $\log(C) = \text{const} + \beta \log \lvert J\rvert + \iota \log(n)$ via OLS, as in Table 2 of the paper. `"linear_share"` estimates $C = c_f + c_v \cdot \lvert J\rvert \cdot n$ via OLS to decompose fixed and variable costs, then computes the fixed-cost share $c_f/\bar{C}$.
`controls`	Optional data frame of control variables to include in the regression. Columns will be added as covariates.
`robust`	Logical. Use heteroskedasticity-robust (HC1) standard errors. Default `TRUE`.

Value

A list of class "`mht_cost_estimate`" containing:

model Model type used

fit The fitted `lm` object

alpha_bar Benchmark alpha

beta, iota Estimated parameters (Cobb-Douglas)

beta_se, iota_se Standard errors

p_beta0, p_beta1 P-values for H0: beta=0 and beta=1

p_iota0, p_iota1 P-values for H0: iota=0 and iota=1

c_f, c_v Estimated fixed and variable costs (Linear)

cf_share Estimated fixed cost share (Linear)

References

Viviano, D., Wuthrich, K., and Niehaus, P. (2026). "A Model of Multiple Hypothesis Testing." arXiv:2104.13367v10. https://arxiv.org/abs/2104.13367

Examples

```
# Simulate data mimicking J-PAL cost structure
set.seed(42)
n <- 300
arms <- sample(1:5, n, replace = TRUE)
sample_size <- sample(500:5000, n, replace = TRUE)
cost <- exp(10 + 0.2 * log(arms) + 0.15 * log(sample_size) + rnorm(n, 0, 0.4))

# Estimate Cobb-Douglas cost function
est <- mht_cost_estimate(cost, arms, sample_size, alpha_bar = 0.05)
print(est)

# Use estimated parameters to compute critical values
mht_critical(J = 5, alpha_bar = 0.05, model = "cobbdouglas",
             beta = est$beta, iota = est$iota)
```

`mht_critical` *Compute Optimal MHT Critical Values*

Description

Compute optimal critical values for multiple hypothesis testing based on Proposition 4.1 of Viviano, Wuthrich, and Niehaus (2026). The optimal per-test significance level is $\alpha(J, \Sigma) = C(J, \Sigma)/(b \cdot \bar{\omega}(J))$.

Usage

```
mht_critical(
  J,
  alpha_bar,
  model = c("linear", "cobbdouglas"),
  cf_share = 0.46,
  J_bar = 3,
  nm_ratio = 1,
  beta = 0.13,
  iota = 0.075
)
```

Arguments

`J`	Integer or `Inf`. Number of hypotheses being tested. Use `Inf` to obtain the limiting value as $\|J\| \to \infty$.
`alpha_bar`	Numeric. Benchmark single-hypothesis test size (e.g., 0.05).
`model`	Character. Cost model: `"linear"` (default) for the Linear model (Equation 27) or `"cobbdouglas"` for the Cobb-Douglas model (Appendix A).
`cf_share`	Numeric. Fixed cost share for the Linear model. Default 0.46, based on Sertkaya et al. (2016).
`J_bar`	Numeric. Average number of subgroups across trials (Linear model). Default 3, based on Pocock et al. (2002).
`nm_ratio`	Numeric. Ratio of per-arm sample size to benchmark ($\bar{n}/\bar{m}$). Default 1.0.
`beta`	Numeric. Elasticity of cost with respect to number of arms (Cobb-Douglas model). Default 0.13 (J-PAL estimate).
`iota`	Numeric. Elasticity of cost with respect to sample size (Cobb-Douglas model). Default 0.075 (J-PAL estimate).

Value

A list of class `"mht_critical"` containing:

alpha_opt Optimal per-test significance level

t_star Optimal z-threshold (critical value for one-sided test)

alpha_bonf Bonferroni significance level ($\bar{\alpha}/|J|$); 0 when J = Inf

t_bonf Bonferroni z-threshold

alpha_sidak Sidak significance level $1 - (1 - \bar{\alpha})^{1/|J|}$; 0 when J = Inf

t_sidak Sidak z-threshold

alpha_bar Benchmark alpha

J Number of hypotheses (may be Inf)

nm_ratio Sample size ratio used

model Cost model used

References

Viviano, D., Wuthrich, K., and Niehaus, P. (2026). "A Model of Multiple Hypothesis Testing." arXiv:2104.13367v10. https://arxiv.org/abs/2104.13367

Examples

```
# Linear calibration: 5 hypotheses, benchmark alpha = 0.05
mht_critical(J = 5, alpha_bar = 0.05)

# Limiting case J -> Inf
mht_critical(J = Inf, alpha_bar = 0.025)

# Cobb-Douglas (J-PAL calibration)
mht_critical(J = 5, alpha_bar = 0.05, model = "cobbdouglas",
             beta = 0.13, iota = 0.075)

# Linear model with larger sample
mht_critical(J = 3, alpha_bar = 0.025, nm_ratio = 1.5)
```

`mht_est` *Apply MHT Testing Directly to a Fitted Model (Postestimation)*

Description

Extracts coefficient estimates, standard errors, and p-values from a fitted regression object and applies the optimal MHT adjustment from Proposition 4.1 of Viviano, Wuthrich, and Niehaus (2026). This is the standard research workflow interface: run a regression, then call `mht_est()` on the result.

Usage

```
mht_est(
  fit,
  vars = NULL,
  alpha_bar,
  onesided = TRUE,
  model = c("linear", "cobbdouglas"),
  cf_share = 0.46,
  J_bar = 3,
  nm_ratio = 1,
  mbar = NULL,
  beta = 0.13,
  iota = 0.075
)
```

Arguments

`fit`	A fitted model object. See Details for supported classes.
`vars`	Character vector of coefficient names to test. If `NULL`, all coefficients except `"(Intercept)"` are tested. Names must match the rownames of `coef(summary(fit))` exactly.
`alpha_bar`	Numeric. Benchmark single-hypothesis significance level (e.g., 0.05).
`onesided`	Logical. If `TRUE` (default), two-sided p-values from the model are converted to one-sided (positive direction). See Details.
`model`	Character. Cost model: `"linear"` (default) or `"cobbdouglas"`. Passed to `mht_critical`.
`cf_share`	Numeric. Fixed cost share (Linear model). Default 0.46.
`J_bar`	Numeric. Average number of subgroups (Linear model). Default 3.
`nm_ratio`	Numeric. Sample size ratio n_bar/m_bar. Default 1.0. Overridden by `mbar` if supplied.
`mbar`	Numeric or `NULL`. Benchmark per-arm sample size. If supplied, `nm_ratio` is computed as `nobs(fit) / J / mbar`, where J is the number of hypotheses tested.
`beta`	Numeric. Arms elasticity (Cobb-Douglas). Default 0.13.
`iota`	Numeric. Sample size elasticity (Cobb-Douglas). Default 0.075.

Details

Supported model classes:

- `lm`, `glm` (base R)
- `lm_robust`, `iv_robust` (estimatr package)
- `fixest` class objects, e.g. `feols()`, `fepois()` (fixest package)

Any other class with a `coef(summary(fit))` method returning a standard 4-column matrix (Estimate, Std. Error, t/z value, Pr(>|t|)) will also work.

Value

A data frame of class `c("mht_est", "data.frame")` with columns:

term Coefficient name

estimate Point estimate

std_error Standard error

t_stat t- or z-statistic

p_value p-value used for testing (one-sided if `onesided=TRUE`)

reject_optimal Logical. Rejection under optimal model-based procedure

reject_bonferroni Logical. Rejection under Bonferroni

reject_holm Logical. Rejection under Holm step-down

reject_bh Logical. Rejection under Benjamini-Hochberg (FDR)

reject_unadjusted Logical. Rejection without adjustment

Attributes `alpha_opt`, `alpha_bonf`, `alpha_bar`, `J`, and `model` are attached.

One-sided vs. two-sided p-values

Most estimation functions return two-sided p-values. By default (`onesided = TRUE`), this function converts to one-sided p-values in the positive direction. This is theoretically grounded in **Remark 6** of the paper: under Assumptions 2–4 (linearity, welfare additivity, and normality), one-sided t-tests are *globally optimal*. The key assumption is that the *status quo* (the baseline policy) remains in place whenever the researcher does not report a positive finding. Negative treatment effects therefore never lead to rejection (welfare gains require positive effects).

Two-sided tests (`onesided = FALSE`) are appropriate when there is uncertainty about the policymaker's default action — i.e., the policymaker may implement a treatment even in the absence of a positive recommendation. **Appendix B.2** (Proposition 9) of the paper shows that in this extended model, two-sided t-tests with the *same* critical threshold $t^* = \Phi^{-1}(1 - \alpha^*)$ are maximin optimal.

Conversion when `onesided = TRUE`:

- If t-statistic >= 0: `p_one = p_two / 2`
- If t-statistic < 0: `p_one = 1 - p_two / 2`

A negative effect is never rejected (p_one > 0.5 for t < 0).

References

Viviano, D., Wuthrich, K., and Niehaus, P. (2026). "A Model of Multiple Hypothesis Testing." arXiv:2104.13367v10. https://arxiv.org/abs/2104.13367

See Also

`mht_test` for direct p-value input, `mht_critical` for computing critical values only.

Examples

```
# Basic usage with lm()
data(mtcars)
fit <- lm(mpg ~ cyl + hp + wt + am, data = mtcars)
mht_est(fit, vars = c("cyl", "hp", "wt", "am"), alpha_bar = 0.05)

# With Cobb-Douglas model
mht_est(fit, vars = c("cyl", "hp", "wt", "am"), alpha_bar = 0.05,
               model = "cobbdouglas", beta = 0.13, iota = 0.075)

# Two-sided test (e.g., for a two-sided hypothesis)
mht_est(fit, vars = c("cyl", "hp", "wt"), alpha_bar = 0.05,
               onesided = FALSE)

# With estimatr::lm_robust(), when that suggested package is available
if (requireNamespace("estimatr", quietly = TRUE)) {
  fit_r <- estimatr::lm_robust(mpg ~ cyl + hp + wt, data = mtcars)
  print(mht_est(fit_r, vars = c("cyl", "hp", "wt"), alpha_bar = 0.05))
}

# With fixest::feols(), when that suggested package is available
if (requireNamespace("fixest", quietly = TRUE)) {
```

```
  fit_fe <- fixest::feols(mpg ~ cyl + hp + wt | gear, data = mtcars)
  print(mht_est(fit_fe, vars = c("cyl", "hp", "wt"), alpha_bar = 0.05))
}
```

---

mht_table *Generate Table of Optimal Critical Values*

Description

Generates a table of optimal critical values for a range of hypothesis counts and sample size ratios, reproducing Table 1 in Viviano, Wuthrich, and Niehaus (2026) (linear calibration) or the J-PAL Cobb-Douglas calibration.

Usage

```
mht_table(
  alpha_bar = c(0.025, 0.05, 0.1, 0.15),
  J_range = c(1:9, Inf),
  nm_ratios = c(0.5, 1, 1.5, 2),
  sidak_bars = c(0.025, 0.05),
  model = c("linear", "cobbdouglas"),
  ...
)
```

Arguments

`alpha_bar`	Numeric vector. Benchmark single-hypothesis sizes for the optimal columns. Default `c(0.025, 0.05, 0.1, 0.15)`.
`J_range`	Numeric vector. Hypothesis counts (rows). May include `Inf` for the limiting $\|J\| \to \infty$ row. Default `c(1:9, Inf)`.
`nm_ratios`	Numeric vector. Sample size ratios $\bar{n}/\bar{m}$. Default `c(0.5, 1.0, 1.5, 2.0)`.
`sidak_bars`	Numeric vector. $\bar{\alpha}$ values for Sidak benchmark columns, appended after the optimal columns. Set to `NULL` to suppress. Default `c(0.025, 0.05)`, matching Table 1 of the paper.
`model`	Character. Cost model: `"linear"` or `"cobbdouglas"`.
`...`	Additional arguments passed to `mht_critical`.

Details

The default arguments reproduce Table 1 of the paper exactly: rows for $|J| = 1, \ldots, 9, \infty$, four $\bar{\alpha}$ columns at $\bar{n}/\bar{m} = 100\%$, single-column groups at 50\ (all with $\bar{\alpha} = 0.025$), and Sidak benchmark columns for $\bar{\alpha} \in \{0.025, 0.05\}$.

Value

A data frame (class "mht_table") with columns:

- J: hypothesis count (may include Inf)
- a<ab>_nm<nm>: optimal alpha for each (alpha_bar, nm_ratio) pair
- sidak_<ab>: Sidak level for each sidak_bars value

References

Viviano, D., Wuthrich, K., and Niehaus, P. (2026). "A Model of Multiple Hypothesis Testing." arXiv:2104.13367v10. https://arxiv.org/abs/2104.13367

Examples

```
# Reproduce Table 1 exactly (v10)
mht_table()

# Cobb-Douglas with J-PAL parameters, no Sidak columns
mht_table(model = "cobbdouglas", beta = 0.13, iota = 0.075, sidak_bars = NULL)

# Custom range, single alpha
mht_table(alpha_bar = 0.05, J_range = 1:20, nm_ratios = 1.0, sidak_bars = NULL)
```

---

mht_test *Perform Hypothesis Tests with Optimal MHT Adjustment*

Description

Given p-values or test statistics, applies the model-optimal MHT adjustment from Proposition 4.1 of Viviano, Wuthrich, and Niehaus (2026) and returns rejection decisions compared across multiple procedures.

Usage

```
mht_test(
  p = NULL,
  z = NULL,
  alpha_bar,
  model = c("linear", "cobbdouglas"),
  cf_share = 0.46,
  J_bar = 3,
  nm_ratio = 1,
  beta = 0.13,
  iota = 0.075
)
```

Arguments

`p`	Numeric vector. One-sided p-values. Provide either `p` or `z`.
`z`	Numeric vector. Z-statistics. If provided, p-values are computed as `1 - pnorm(z)`.
`alpha_bar`	Numeric. Benchmark single-hypothesis test size.
`model`	Character. Cost model: `"linear"` (default) or `"cobbdouglas"`.
`cf_share`	Numeric. Fixed cost share (Linear model). Default 0.46.
`J_bar`	Numeric. Average number of subgroups (Linear model). Default 3.
`nm_ratio`	Numeric. Sample size ratio. Default 1.0.
`beta`	Numeric. Arms elasticity (Cobb-Douglas). Default 0.13.
`iota`	Numeric. Sample size elasticity (Cobb-Douglas). Default 0.075.

Value

A data frame of class `"mht_test"` with columns:

p_value Input p-values

reject_optimal Rejection under the optimal model-based procedure

reject_bonferroni Rejection under Bonferroni correction

reject_holm Rejection under Holm's step-down procedure

reject_bh Rejection under Benjamini-Hochberg (FDR) control

reject_unadjusted Rejection without adjustment

The object also carries attributes `alpha_opt`, `alpha_bonf`, `alpha_bar`, `J`, and `model`.

References

Viviano, D., Wuthrich, K., and Niehaus, P. (2026). "A Model of Multiple Hypothesis Testing." arXiv:2104.13367v10. https://arxiv.org/abs/2104.13367

Examples

```
# Test with p-values
pvals <- c(0.003, 0.015, 0.030, 0.048, 0.120, 0.500)
result <- mht_test(p = pvals, alpha_bar = 0.05)
print(result)

# Test with z-statistics
zstats <- qnorm(1 - pvals)
result2 <- mht_test(z = zstats, alpha_bar = 0.05)

# Cobb-Douglas model
result3 <- mht_test(p = pvals, alpha_bar = 0.05, model = "cobbdouglas")
```

Index