

# Package ‘evalHTE’

February 3, 2026

**Title** Evaluating Heterogeneous Treatment Effects

**Version** 0.1.1

**Description** Provides various statistical methods for evaluating heterogeneous treatment effects (HTE) in randomized experiments. The package includes tools to estimate uniform confidence bands for estimation of the group average treatment effect sorted by generic machine learning algorithms (GATES). It also provides the tools to identify a subgroup of individuals who are likely to benefit from a treatment the most ``exceptional responders'' or those who are harmed by it. Detailed reference in Imai and Li (2023) <[doi:10.48550/arXiv.2310.07973](https://doi.org/10.48550/arXiv.2310.07973)>.

**License** MIT + file LICENSE

**Depends** R (>= 3.50), dplyr (>= 1.0.10)

**Imports** cli, evalITR, ggplot2, ggthemes, rlang, zoo, furrr, ggdist, scales, tidyr, stats, purrr, Matrix, MASS, quadprog, caret

**Encoding** UTF-8

**RoxygenNote** 7.3.3

**Suggests** knitr, rmarkdown, future, grf, magrittr, tibble

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Michael Lingzhi Li [aut, cre],  
Kosuke Imai [aut],  
Jialu Li [ctb]

**Maintainer** Michael Lingzhi Li <[mili@hbs.edu](mailto:mili@hbs.edu)>

**Repository** CRAN

**Date/Publication** 2026-02-03 09:00:02 UTC

## Contents

compute_qoi	2
compute_qoi_user	2
consist.test	3
consistcv.test	4

estimate_hte . . . . .	5
evaluate_hte . . . . .	6
GATE . . . . .	7
GATEcv . . . . .	8
het.test . . . . .	9
hetcv.test . . . . .	10
plot.hte . . . . .	11
plot_CI . . . . .	12
plot_CI.hte . . . . .	12
print.summary.hte . . . . .	13
print.summary.test_hte . . . . .	13
summary.hte . . . . .	14
summary.test_hte . . . . .	14
test_itr . . . . .	15
URATE . . . . .	16

<b>Index</b>	<b>17</b>
--------------	-----------

---

<b>compute_qoi</b>	<i>Compute Quantities of Interest (GATE, GATEcv, URATE)</i>
--------------------	---

---

## Description

Compute Quantities of Interest (GATE, GATEcv, URATE)

## Usage

```
compute_qoi(fit_obj, algorithms)
```

## Arguments

<b>fit_obj</b>	An output object of <code>estimate_itr</code> function from the <code>evalITR</code> package.
<b>algorithms</b>	Machine learning algorithms

---

<b>compute_qoi_user</b>	<i>Compute Quantities of Interest (GATE, GATEcv, URATE) with user defined functions</i>
-------------------------	---

---

## Description

Compute Quantities of Interest (GATE, GATEcv, URATE) with user defined functions

## Usage

```
compute_qoi_user(user_hte, Tcv, Ycv, data, ngates, ...)
```

**Arguments**

user_hte	A user-defined function to estimate heterogeneous treatment effects (HTE). The function should take the data as input and return an unit-level continuous score for treatment assignment. We assume those that have score less than 0 should not have treatment. The default is NULL, which means the heterogeneous treatment effects will be estimated from the package.
Tcv	A vector of the unit-level binary treatment.
Ycv	A vector of the unit-level continuous outcome.
data	A data frame containing the variables of interest.
ngates	The number of gates to be used in the GATE function.
...	Additional arguments to be passed to the user-defined function.

---

consist.test	<i>The Consistency Test for Grouped Average Treatment Effects (GATEs) in Randomized Experiments</i>
--------------	---

---

**Description**

This function calculates statistics related to the test of treatment effect consistency across groups.

**Usage**

```
consist.test(D, tau, Y, ngates = 5, nsim = 10000)
```

**Arguments**

D	A vector of the unit-level binary treatment receipt variable for each sample.
tau	A vector of the unit-level continuous score. Conditional Average Treatment Effect is one possible measure.
Y	A vector of the outcome variable of interest for each sample.
ngates	The number of groups to separate the data into. The groups are determined by tau. Default is 5.
nsim	Number of Monte Carlo simulations used to simulate the null distributions. Default is 10000.

**Details**

The details of the methods for this design are given in Imai and Li (2022).

**Value**

A list that contains the following items:

stat	The estimated statistic for the test of consistency
pval	The p-value of the null hypothesis (that the treatment effects are consistent)

## Author(s)

Michael Lingzhi Li, Technology and Operations Management, Harvard Business School <mili@hbs.edu>, <https://www.michaellz.com/>;

## References

Imai and Li (2022). “Statistical Inference for Heterogeneous Treatment Effects Discovered by Generic Machine Learning in Randomized Experiments”,

## Examples

```
D = c(1,0,1,0,1,0,1,0)
tau = c(0,0.1,0.2,0.3,0.4,0.5,0.6,0.7)
Y = c(4,5,0,2,4,1,-4,3)
consisttestlist <- consist.test(D, tau, Y, ngates=5)
consisttestlist$stat
consisttestlist$pval
```

consistcv.test

*The Consistency Test for Grouped Average Treatment Effects (GATEs) under Cross Validation in Randomized Experiments*

## Description

This function calculates statistics related to the test of treatment effect consistency across groups under cross-validation.

## Usage

```
consistcv.test(D, tau, Y, ind, ngates = 5, nsim = 10000)
```

## Arguments

D	A vector of the unit-level binary treatment receipt variable for each sample.
tau	A vector of the unit-level continuous score. Conditional Average Treatment Effect is one possible measure.
Y	A vector of the outcome variable of interest for each sample.
ind	A vector of integers (between 1 and number of folds inclusive) indicating which testing set does each sample belong to.
ngates	The number of groups to separate the data into. The groups are determined by tau. Default is 5.
nsim	Number of Monte Carlo simulations used to simulate the null distributions. Default is 10000.

## Details

The details of the methods for this design are given in Imai and Li (2022).

**Value**

A list that contains the following items:

stat	The estimated statistic for the test of consistency under cross-validation.
pval	The p-value of the null hypothesis (that the treatment effects are consistent)

**Author(s)**

Michael Lingzhi Li, Technology and Operations Management, Harvard Business School <mili@hbs.edu>,  
<https://www.michaellz.com/>;

**References**

Imai and Li (2022). “Statistical Inference for Heterogeneous Treatment Effects Discovered by Generic Machine Learning in Randomized Experiments”,

**Examples**

```
D = c(1,0,1,0,1,0,1,0)
tau = matrix(c(0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,-0.5,-0.3,-0.1,0.1,0.3,0.5,0.7,0.9),nrow = 8, ncol = 2)
Y = c(4,5,0,2,4,1,-4,3)
ind = c(rep(1,4),rep(2,4))
consisttestlist <- consistcv.test(D,tau,Y,ind,ngates=2)
consisttestlist$stat
consisttestlist$pval
```

estimate\_hte

*Evaluate Heterogeneous Treatment Effects***Description**

Evaluate Heterogeneous Treatment Effects

**Usage**

```
estimate_hte(
  treatment,
  form,
  data,
  algorithms,
  n_folds = 5,
  split_ratio = 0,
  ngates = 5,
  preprocess = NULL,
  weights = NULL,
  trControl = caret::trainControl(method = "none"),
  tuneGrid = NULL,
  tuneLength = ifelse(trControl$method == "none", 1, 3),
```

```

user_model = NULL,
SL_library = NULL,
meta_learner = "slearner",
...
)

```

## Arguments

treatment	Treatment variable
form	a formula object that takes the form $y \sim D + x_1 + x_2 + \dots$
data	A data frame that contains the outcome $y$ and the treatment $D$ .
algorithms	List of machine learning algorithms to be used.
n_folds	Number of cross-validation folds. Default is 5.
split_ratio	Split ratio between train and test set under sample splitting. Default is 0.
ngates	The number of groups to separate the data into. The groups are determined by $\tau$ . Default is 5.
preProcess	caret parameter
weights	caret parameter
trControl	caret parameter
tuneGrid	caret parameter
tuneLength	caret parameter
user_model	A user-defined function to estimate heterogeneous treatment effects.
SL_library	A list of machine learning algorithms to be used in the super learner.
meta_learner	The type of meta-learner to use (e.g., "slearner", "tlearner"). Default is "slearner".
...	Additional arguments passed to <code>caret::train</code> .

## Value

An object of `hte` class

---

<code>evaluate_hte</code>	<i>Evaluate Heterogeneous Treatment Effects</i>
---------------------------	---

---

## Description

Evaluate Heterogeneous Treatment Effects

## Usage

`evaluate_hte(fit, ...)`

**Arguments**

fit                    Fitted model. Usually an output from estimate\_hte.  
 ...                    Additional arguments passed to the function.

**Value**

An object of hte class

---

GATE

*Estimation of the Grouped Average Treatment Effects (GATEs) in Randomized Experiments*

---

**Description**

This function estimates the Grouped Average Treatment Effects (GATEs) where the groups are determined by a continuous score. The details of the methods for this design are given in Imai and Li (2022).

**Usage**

`GATE(D, tau, Y, ngates = 5)`

**Arguments**

D                    A vector of the unit-level binary treatment receipt variable for each sample.  
 tau                A vector of the unit-level continuous score. Conditional Average Treatment Effect is one possible measure.  
 Y                    A vector of the outcome variable of interest for each sample.  
 ngates            The number of groups to separate the data into. The groups are determined by tau. Default is 5.

**Value**

A list that contains the following items:

gate                The estimated vector of GATEs of length ngates arranged in order of increasing tau.  
 sd                    The estimated vector of standard deviation of GATEs.

**Author(s)**

Michael Lingzhi Li, Technology and Operations Management, Harvard Business School <mili@hbs.edu>,  
<https://www.michaellz.com/>;

## References

Imai and Li (2022). “Statistical Inference for Heterogeneous Treatment Effects Discovered by Generic Machine Learning in Randomized Experiments”,

## Examples

```
D = c(1,0,1,0,1,0,1,0)
tau = c(0,0.1,0.2,0.3,0.4,0.5,0.6,0.7)
Y = c(4,5,0,2,4,1,-4,3)
gatelist <- GATE(D,tau,Y,ngates=5)
gatelist$gate
gatelist$sd
```

---

GATEcv

*Estimation of the Grouped Average Treatment Effects (GATEs) in Randomized Experiments Under Cross Validation*

---

## Description

This function estimates the Grouped Average Treatment Effects (GATEs) under cross-validation where the groups are determined by a continuous score. The details of the methods for this design are given in Imai and Li (2022).

## Usage

```
GATEcv(D, tau, Y, ind, ngates = 5)
```

## Arguments

D	A vector of the unit-level binary treatment receipt variable for each sample.
tau	A matrix where the $i$ th column is the unit-level continuous score for treatment assignment generated in the $i$ th fold. Conditional Average Treatment Effect is one possible measure.
Y	A vector of the outcome variable of interest for each sample.
ind	A vector of integers (between 1 and number of folds inclusive) indicating which testing set does each sample belong to.
ngates	The number of groups to separate the data into. The groups are determined by tau. Default is 5.

## Value

A list that contains the following items:

gate	The estimated vector of GATEs under cross-validation of length ngates arranged in order of increasing tau.
sd	The estimated vector of standard deviation of GATEs under cross-validation.

**Author(s)**

Michael Lingzhi Li, Technology and Operations Management, Harvard Business School <mili@hbs.edu>,  
<https://www.michaellz.com/>;

**References**

Imai and Li (2022). “Statistical Inference for Heterogeneous Treatment Effects Discovered by Generic Machine Learning in Randomized Experiments”,

**Examples**

```
D = c(1,0,1,0,1,0,1,0)
tau = matrix(c(0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,-0.5,-0.3,-0.1,0.1,0.3,0.5,0.7,0.9),nrow = 8, ncol = 2)
Y = c(4,5,0,2,4,1,-4,3)
ind = c(rep(1,4),rep(2,4))
gatelist <- GATEcv(D, tau, Y, ind, ngates = 2)
gatelist$gate
gatelist$sd
```

het.test

*The Heterogeneity Test for Grouped Average Treatment Effects (GATEs) in Randomized Experiments*

**Description**

This function calculates statistics related to the test of heterogeneous treatment effects across groups.

**Usage**

```
het.test(D, tau, Y, ngates = 5)
```

**Arguments**

D	A vector of the unit-level binary treatment receipt variable for each sample.
tau	A vector of the unit-level continuous score. Conditional Average Treatment Effect is one possible measure.
Y	A vector of the outcome variable of interest for each sample.
ngates	The number of groups to separate the data into. The groups are determined by tau. Default is 5.

**Details**

The details of the methods for this design are given in Imai and Li (2022).

**Value**

A list that contains the following items:

stat	The estimated statistic for the test of heterogeneity.
pval	The p-value of the null hypothesis (that the treatment effects are homogeneous)

**Author(s)**

Michael Lingzhi Li, Technology and Operations Management, Harvard Business School <mili@hbs.edu>,  
<https://www.michaellz.com/>;

**References**

Imai and Li (2022). “Statistical Inference for Heterogeneous Treatment Effects Discovered by Generic Machine Learning in Randomized Experiments”,

**Examples**

```
D = c(1,0,1,0,1,0,1,0)
tau = c(0,0.1,0.2,0.3,0.4,0.5,0.6,0.7)
Y = c(4,5,0,2,4,1,-4,3)
hettestlist <- het.test(D,tau,Y,ngates=5)
hettestlist$stat
hettestlist$pval
```

**hetcv.test**

*The Heterogeneity Test for Grouped Average Treatment Effects (GATEs) under Cross Validation in Randomized Experiments*

**Description**

This function calculates statistics related to the test of heterogeneous treatment effects across groups under cross-validation.

**Usage**

```
hetcv.test(D, tau, Y, ind, ngates = 5)
```

**Arguments**

D	A vector of the unit-level binary treatment receipt variable for each sample.
tau	A vector of the unit-level continuous score. Conditional Average Treatment Effect is one possible measure.
Y	A vector of the outcome variable of interest for each sample.
ind	A vector of integers (between 1 and number of folds inclusive) indicating which testing set does each sample belong to.
ngates	The number of groups to separate the data into. The groups are determined by tau. Default is 5.

## Details

The details of the methods for this design are given in Imai and Li (2022).

## Value

A list that contains the following items:

stat	The estimated statistic for the test of heterogeneity under cross-validation.
pval	The p-value of the null hypothesis (that the treatment effects are homogeneous)

## Author(s)

Michael Lingzhi Li, Technology and Operations Management, Harvard Business School <mili@hbs.edu>,  
<https://www.michaellz.com/>;

## References

Imai and Li (2022). “Statistical Inference for Heterogeneous Treatment Effects Discovered by Generic Machine Learning in Randomized Experiments”,

## Examples

```
D = c(1,0,1,0,1,0,1,0)
tau = matrix(c(0.0,1,0.2,0.3,0.4,0.5,0.6,0.7,-0.5,-0.3,-0.1,0.1,0.3,0.5,0.7,0.9),nrow = 8, ncol = 2)
Y = c(4,5,0,2,4,1,-4,3)
ind = c(rep(1,4),rep(2,4))
hettestlist <- hetcv.test(D,tau,Y,ind,ngates=2)
hettestlist$stat
hettestlist$pval
```

plot.hte *Plot the GATE estimate*

## Description

Plot the GATE estimate

## Usage

```
## S3 method for class 'hte'
plot(x, ...)
```

## Arguments

x	An table object. This is typically an output of <code>evaluate_hte()</code> function.
...	Further arguments passed to the function.

## Value

A plot of ggplot2 object.

---

**plot\_CI***Plot Confidence Intervals*

---

**Description**

A generic function to plot uniform and pointwise confidence intervals for HTE objects.

**Usage**

```
plot_CI(x, ...)
```

**Arguments**

- `x` An object for which a plot is desired.
- `...` Further arguments passed to methods.

**Value**

A ggplot2 object displaying uniform and pointwise confidence intervals for heterogeneous treatment effects.

---

**plot\_CI.hte***Plot the uniform confidence interval*

---

**Description**

Plot the uniform confidence interval

**Usage**

```
## S3 method for class 'hte'
plot_CI(x, alpha = 0.05, ...)
```

**Arguments**

- `x` An object of `evaluate_hte()` class. This is typically an output of `evaluate_hte()` function.
- `alpha` Significance level. Default is 0.05.
- `...` Further arguments passed to the function.

**Value**

A plot of ggplot2 object.

---

```
print.summary.hte      Print
```

---

**Description**

Print

**Usage**

```
## S3 method for class 'summary.hte'  
print(x, ...)
```

**Arguments**

- x An object of `summary.hte` class. This is typically an output of `summary.hte()` function.
- ... Other parameters. Currently not supported.

**Value**

No return value, called for side effects (prints summary tables to console).

---

```
print.summary.test_hte  
      Print
```

---

**Description**

Print

**Usage**

```
## S3 method for class 'summary.test_hte'  
print(x, ...)
```

**Arguments**

- x An object of `summary.test_hte` class. This is typically an output of `summary.test_hte()` function.
- ... Other parameters.

**Value**

No return value, called for side effects (prints test results to console).

---

**summary.hte***Summarize Heterogeneity and Consistency Tests*

---

**Description**

Summarize Heterogeneity and Consistency Tests

**Usage**

```
## S3 method for class 'hte'
summary(object, ...)
```

**Arguments**

object	An object of <code>evaluate_hte</code> class (typically an output of <code>evaluate_hte()</code> function).
...	Other parameters.

**Value**

An object of class `summary.hte`, which is a list containing:

**GATE** A tibble with group average treatment effect estimates, including columns: `group`, `algorithm`, `estimate`, `std.deviation`, `lower`, `upper`, `z.score`, and `p.value`.

**URATE** A tibble with uplift rate estimates for exceptional responders, including columns: `algorithm`, `estimate`, `std.deviation`, `conf.low.uniform`, `z.score`, and `p.value`. Returns `NULL` when cross-validation is used.

---

**summary.test\_hte***Summarize Heterogeneity and Consistency Tests*

---

**Description**

Summarize Heterogeneity and Consistency Tests

**Usage**

```
## S3 method for class 'test_hte'
summary(object, ...)
```

**Arguments**

object	An object of <code>test_hte</code> class (typically an output of <code>test_hte()</code> function).
...	Other parameters.

**Value**

An object of class `summary.test_hte`, which is a list containing:

**Consistency** A tibble with consistency test results, including columns: algorithm, statistic, and p.value (for sample splitting).

**Heterogeneity** A tibble with heterogeneity test results, including columns: algorithm, statistic, and p.value (for sample splitting).

**Consistency\_cv** A tibble with consistency test results for cross-validation.

**Heterogeneity\_cv** A tibble with heterogeneity test results for cross-validation.

Note: The output contains either the first two or last two elements depending on whether cross-validation was used.

---

**test\_itr** *Conduct hypothesis tests*

---

**Description**

Conduct hypothesis tests

**Usage**

```
test_itr(model, nsim = 1000, ...)
```

**Arguments**

<code>model</code>	Fitted model. Usually an output from <code>evaluate_hte</code> .
<code>nsim</code>	Number of Monte Carlo simulations used to simulate the null distributions. Default is 1000.
<code>...</code>	Further arguments passed to the function.

**Value**

An object of `test_itr` class

---

URATE	<i>This function use individualized treatment rule to identify exceptional responders. The details of the methods for this design are given in Imai and Li (2023).</i>
-------	--

---

## Description

This function use individualized treatment rule to identify exceptional responders. The details of the methods for this design are given in Imai and Li (2023).

## Usage

```
URATE(D, tau, Y)
```

## Arguments

D	A vector of the unit-level binary treatment receipt variable for each sample.
tau	A vector of the unit-level continuous score. Conditional Average Treatment Effect is one possible measure.
Y	A vector of the outcome variable of interest for each sample.

## Value

A list that contains the following items:

rate	The estimated vector of URATE of length Y.
sd	The estimated vector of standard deviation of URATE.

## Author(s)

Michael Lingzhi Li, Technology and Operations Management, Harvard Business School <mili@hbs.edu>,  
<https://www.michaellz.com/>;

## References

Imai and Li (2022). “Statistical Inference for Heterogeneous Treatment Effects Discovered by Generic Machine Learning in Randomized Experiments”,

## Examples

```
D <- c(1, 0, 1, 0, 1, 0, 1, 0)
tau <- c(0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7)
Y <- c(4, 5, 0, 2, 4, 1, -4, 3)
ratelist <- URATE(D, tau, Y)
ratelist$rate
ratelist$sd
```

# Index

- \* **band**
  - URATE, 16
- \* **evaluation**
  - consist.test, 3
  - consistcv.test, 4
  - GATE, 7
  - GATEcv, 8
  - het.test, 9
  - hetcv.test, 10
- \* **uniform**
  - URATE, 16

compute\_qoi, 2  
compute\_qoi\_user, 2  
consist.test, 3  
consistcv.test, 4

estimate\_hte, 5  
evaluate\_hte, 6

GATE, 7  
GATEcv, 8

het.test, 9  
hetcv.test, 10

plot.hte, 11  
plot\_CI, 12  
plot\_CI.hte, 12  
print.summary.hte, 13  
print.summary.test\_hte, 13

summary.hte, 14  
summary.test\_hte, 14

test\_itr, 15

URATE, 16