Package 'ROOT'

February 17, 2026

Title ROOT (Rashomon Set of Optimal Trees)

Version 0.0.0.9000

Description ROOT (Rashomon Set of Optimal Trees) is a general framework for globally optimizing user-specified objective functionals over interpretable binary weight functions represented as sparse decision trees. It searches over candidate trees to construct a Rashomon set of near-optimal solutions and derives a summary tree highlighting stable patterns in the optimized weights. ROOT includes a built-in generalizability mode for identifying subgroups in trial settings for transportability analyses.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Suggests mlbench,
testthat (>= 3.0.0),
knitr,
rmarkdown,
ragg

Config/testthat/edition 3

Imports MASS,
rpart,
gbm,
stats,
withr,
rpart.plot

VignetteBuilder knitr

Depends R (>= 3.5)

LazyData true

Contents

characterizing_underrep *Characterize subgroups (wrapper around ROOT) via binary weight*

Description

A convenience wrapper around ROOT() for under-representation analyses. It takes a single data set and calls ROOT() either in generalizability_path mode (when generalizability_path = TRUE) to identify insufficiently represented subgroups in trial data in comparison to the population of interest or in general optimization mode (generalizability_path = FALSE) to identify the binary weight for each subgroup.

Usage

```
characterizing_underrep(
  data,
  global_objective_fn = NULL,
  generalizability_path = FALSE,
  leaf_proba = 0.25,
  seed = 123,
  num_trees = 10,
  vote_threshold = 2/3,
  explore_proba = 0.05,
  feature_est = "Ridge",
  feature_est_args = list(),
  top_k_trees = FALSE,
  k = 10,
  cutoff = "baseline",
  verbose = FALSE
)
```

Arguments

data
: A data.frame containing covariates and, in generalizability_path mode, also columns Y, Tr, and S.

global_objective_fn
: function with signature function(D) -> numeric scoring the entire state and minimized by ROOT. If NULL, a default variance-based objective is used (see objective_default()).

generalizability_path
: Logical. If TRUE, calls ROOT() with generalizability_path = TRUE and expects columns Y, Tr, and S in data. If FALSE, calls ROOT() in general optimization mode. Default FALSE.

leaf_proba
: A numeric(1) tuning parameter that increases the chance a node stops splitting by selecting a synthetic "leaf" feature. Internally, the probability of choosing "leaf" is leaf_proba / (1 + leaf_proba) (assuming the covariate probabilities sum to 1). Default 0.25.

seed
: Random seed for reproducibility.

num_trees
: Number of trees to grow.

vote_threshold	Majority vote threshold used for w_opt.
explore_proba	Exploration probability in tree growth.
feature_est	Either "Ridge", "GBM", or a custom feature importance function.
feature_est_args	List of extra arguments passed to feature_est when it is a function.
top_k_trees	Logical; if TRUE, uses top k trees by objective, otherwise a cutoff rule.
k	Number of trees when top_k_trees = TRUE.
cutoff	Numeric or "baseline" Rashomon cutoff.
verbose	Logical; if TRUE, prints progress/estimands from ROOT().

Details

When generalizability_path = TRUE, data must contain standardized columns:

- Y: outcome,
- Tr: treatment indicator (0/1),
- S: sample indicator (1 = trial, 0 = target).

Value

A characterizing_underrep S3 object (a list) with:

root	The ROOT object returned by ROOT().
combined	The input data (for continuity with prior API).
leaf_summary	Data frame of terminal node rules and labels, or NULL.

References

Parikh H, Ross RK, Stuart E, Rudolph KE (2025). "Who Are We Missing?: A Principled Approach to Characterizing the Underrepresented Population." *Journal of the American Statistical Association*. doi:10.1080/01621459.2025.2495319

Examples

```
## Not run:
char.output = characterizing_underrep(diabetes_data,generalizability_path = TRUE, seed = 123)

## End(Not run)
```

diabetes_data *Simulated diabetes dataset for examples*

Description

A toy dataset for illustrating ROOT examples and tests.

Usage

```
data(diabetes_data)
```

Format

A `data.frame` with one row per individual and the columns:

Age45 Indicator in `0/1`: age >= 45.

DietYes Indicator in `0/1`: on a diet program.

Race_Black Indicator in `0/1`: race is Black.

S Sample indicator in `0/1`: `1` means RCT or source, `0` means target.

Sex_Male Indicator in `0/1`: male.

Tr Treatment assignment in `0/1`.

Y Observed outcome (`numeric` or `0/1`).

Abbreviations

RCT means randomized clinical trial. ATE means Average Treatment Effect.

plot.characterizing_underrep

Plot Under represented Population Characterization

Description

Visualizes the decision tree derived from the `ROOT` analysis. Highlights which subgroups are represented where `w = 1` versus underrepresented where `w = 0` in generalization mode, or simply `w(x)` in `{0,1}` in general optimization mode.

Usage

```
## S3 method for class 'characterizing_underrep'
plot(
  x,
 main = "Subgroup Characterization from Final Characterized Tree from Rashomon Set",
  cex.main = 1.2,
  ...
)
```

Arguments

`x`	A `characterizing_underrep` S3 object with `x$root$f` present as an `rpart` object for the summary or characterization tree.
`main`	Character string for the plot title. Default is `"Underrepresented Population Characterization"`.
`cex.main`	Numeric scaling factor for the title text size. Default is `1.2`.
`...`	Additional arguments passed to `rpart.plot::prp()`.

Value

`NULL`. The plot is drawn to the active graphics device.

Examples

```
## Not run:
char.output = characterizing_underrep(diabetes_data, generalizability_path = TRUE, seed = 123)
plot(char.output)
plot(char.output, main = "My Custom Title", cex.main = 1.5)

## End(Not run)
```

`plot.ROOT` *Plot the ROOT summary tree*

Description

Visualizes the decision tree that characterizes the weighted subgroup (the weight function $w(d)$ in `{0,1}`) identified by `ROOT()`, using `rpart.plot::prp()`.

Usage

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

Arguments

`x` A `"ROOT"` S3 object returned by `ROOT()` with `x$f` an `rpart` object representing the summary / characterization tree.

`...` Additional arguments passed to `rpart.plot::prp()`.

Value

No return value; the plot is drawn to the active graphics device.

Examples

```
## Not run:
ROOT.output = ROOT(diabetes_data,generalizability_path = TRUE, seed = 123)
plot(ROOT.output)

## End(Not run)
```

`ROOT` *Ensemble of weighted trees for general optimization and Rashomon selection*

Description

Builds multiple weighted trees, then identifies a "Rashomon set" of top-performing trees and aggregates their weight assignments by majority vote.

Usage

```
ROOT(
  data,
  global_objective_fn = NULL,
  generalizability_path = FALSE,
  leaf_proba = 0.25,
  seed = NULL,
  num_trees = 10,
  vote_threshold = 2/3,
  explore_proba = 0.05,
  feature_est = "Ridge",
  feature_est_args = list(),
  top_k_trees = FALSE,
  k = 10,
  cutoff = "baseline",
  verbose = FALSE
)
```

Arguments

`data` A data.frame containing the dataset.

In *general optimization* mode (`generalizability_path = FALSE`), `data` can be any set of covariates and auxiliary columns. The user supplies a `global_objective_fn` that takes a data frame with a column `w` and returns a scalar loss.

In *generalizability_path* mode (`generalizability_path = TRUE`), `data` must contain columns `"Y"` (outcome), `"Tr"` (treatment indicator, 0/1), and `"S"` (sample indicator, 1 = trial, 0 = target). ROOT internally constructs transportability scores and, if no custom objective is given, uses a default variance-based loss.

`global_objective_fn`

A function with signature `function(D) -> numeric` scoring the entire state and minimized by ROOT. If NULL, a default variance-based objective is used (see `objective_default()`).

`generalizability_path`

`Logical(1)`. If TRUE, use the built-in transportability objective based on (Y, Tr, S). If FALSE, treat `data` as arbitrary and rely on `global_objective_fn`. Default FALSE.

`leaf_proba` A numeric tuning parameter that increases the chance a node stops splitting by selecting a synthetic `"leaf"` feature. Internally, the probability of choosing `"leaf"` is `leaf_proba / (1 + leaf_proba)` (assuming the covariate probabilities sum to 1). Default `0.25`.

`seed` An optional numeric seed for reproducibility.

`num_trees` An integer number of trees to grow. Default 10.

`vote_threshold` A numeric in (0.5, 1] giving the majority vote threshold for final `w = 1`. Default 2/3.

`explore_proba` A numeric giving the exploration probability at leaves. Default 0.05.

`feature_est` Either a character(1) in c("Ridge", "GBM") or a function(X, y, ...) returning a named nonnegative numeric vector of importances with names matching columns of X. Used only to bias which covariates are chosen for splitting. If it fails, ROOT falls back to uniform feature sampling with a warning.

`feature_est_args`
: A list of additional arguments passed to a user supplied `feature_est` function.

`top_k_trees`
: `Logical(1)`. If TRUE, select top `k` trees by objective; otherwise use `cutoff`. Default FALSE.

`k`
: An integer giving the number of top trees when `top_k_trees = TRUE`. Default 10.

`cutoff`
: A numeric or `"baseline"`. Used as the Rashomon cutoff when `top_k_trees = FALSE`. `"baseline"` uses the objective at w = 1 (all weights equal to 1).

`verbose`
: `Logical(1)`. If TRUE, prints unweighted and (when available) weighted estimates and their standard errors in generalizability_path mode.

Details

The function is framed as a general functional optimization routine: given data D_n and a loss $L(w, D_n)$, ROOT searches over interpretable tree-based weight functions $w(d)$ in `{0,1}`.

Value

An object of class "ROOT" (a list) with elements:

- D_rash: data frame with Rashomon-set votes and w_opt.
- D_forest: data frame with forest-level working columns.
- w_forest: list of per-tree results from split_node().
- rashomon_set: indices of selected trees.
- global_objective_fn: the objective function used.
- f: summary classifier (e.g., rpart tree) or NULL.
- testing_data: data frame aligned to rows used to compute scores.
- estimate: (only if generalizability_path = TRUE) list with unweighted and weighted estimands, standard errors (SEs), and a note about the SE.
- generalizability_path: logical flag.

References

Parikh H, Ross RK, Stuart E, Rudolph KE (2025). "Who Are We Missing?: A Principled Approach to Characterizing the Underrepresented Population." *Journal of the American Statistical Association*. doi:10.1080/01621459.2025.2495319

Examples

```
## Not run:
ROOT.output = ROOT(diabetes_data,generalizability_path = TRUE, seed = 123)

## End(Not run)
```

summary.characterizing_underrep

Summarize a characterizing_underrep fit

Description

Summarizes the ROOT summary which includes unweighted and (when in generalization mode) weighted estimates with standard errors, as reported by summary.ROOT(). Provides a brief overview of terminal rules from the annotated summary tree when available.

Usage

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

Arguments

object A characterizing_underrep S3 object. Expected components include root which is a ROOT object (summarized by summary.ROOT()) and may contain f which is an rpart object for the summary tree, and leaf_summary which is a data.frame with one row per terminal node and may include a rule column of type character.

... Currently unused. Included for S3 compatibility.

Details

Delegates core statistics and estimands to summary(object$root). Previews up to ten terminal rules when a summary tree exists.

Value

object returned invisibly. Printed output is a human readable summary.

Abbreviations

ATE means Average Treatment Effect. RCT means Randomized Controlled Trial. SE means Standard Error. TATE means Transported ATE. WTATE means Weighted TATE. WATE means Weighted ATE. PATE means Population ATE.

Examples

```
## Not run:
char.output = characterizing_underrep(diabetes_data,generalizability_path = TRUE, seed = 123)
summary(char.output)

## End(Not run)
```

summary.ROOT *Summarize a ROOT fit*

Description

Provides a human-readable summary of a ROOT object, including:

1. the summary characterization tree f,
2. the first few rows of testing_data,
3. the global_objective_fn used during optimization, and
4. in generalizability mode (generalizability_path = TRUE), the unweighted and weighted estimands with their standard errors and an explanatory note for the weighted standard error (SE).

Usage

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

Arguments

object	A "ROOT" S3 object returned by ROOT().
...	Currently unused and included for S3 compatibility.

Details

When generalizability_path = TRUE, the unweighted estimand corresponds to a SATE-type quantity and the weighted estimand to a WTATE-type quantity for the transported target population. When generalizability_path = FALSE, ROOT is used for general functional optimization and no causal labels are imposed; the summary focuses on the tree and diagnostics.

Value

object returned invisibly. Printed output is for inspection.

Diagnostics

The summary also reports:

- the number of trees grown,
- the size of the Rashomon set,
- the percentage of observations with ensemble vote w_opt == 1.

Examples

```
## Not run:
ROOT.output = ROOT(diabetes_data,generalizability_path = TRUE, seed = 123)
summary(ROOT.output)

## End(Not run)
```

Index