

Package ‘sense’

September 8, 2026

Type Package

Title Automatic Stacked Ensemble for Regression Tasks

Version 1.2.0

Description Stacked ensembles for regression tasks using the 'mlr3' framework, internal preprocessing and out-of-fold stacking, and hyperparameter tuning using grid or random search. Supports numeric and categorical predictors.

License GPL-3

Encoding UTF-8

LazyData true

Depends R (>= 4.1)

Imports mlr3 (>= 0.12.0), mlr3learners (>= 0.5.0), paradox (>= 1.0.0), mlr3tuning (>= 0.8.0), bbotk (>= 0.3.2), data.table (>= 1.14.0), R6, stats, graphics, utils

Suggests testthat (>= 3.0.0), mlr3filters (>= 0.4.2), knitr, xgboost (>= 1.4.1.1), rpart (>= 4.1-15), ranger (>= 0.13.1), kknn (>= 1.3.1), glmnet (>= 4.1-2), e1071 (>= 1.7-8), FSelectorRcpp (>= 0.3.8), care (>= 1.1.10), praznik (>= 8.0.0), lme4 (>= 1.1-27.1)

VignetteBuilder knitr

Config/testthat/edition 3

URL <https://mlr3.mlr-org.com/>

Config/roxygen2/version 8.1.0

RoxygenNote 7.3.3

NeedsCompilation no

Author Giancarlo Vercellino [aut, cre]

Maintainer Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

Repository CRAN

Date/Publication 2026-09-08 05:20:02 UTC

Contents

benchmark	2
plot.sense_pipeline	2
sense	3
Index	6

benchmark	<i>benchmark data set</i>
-----------	---------------------------

Description

A data frame for regression task generated with mlbench friedman1.

Usage

benchmark

Format

A data frame with 11 columns and 150 rows.

Source

mlbench, friedman1

plot.sense_pipeline	<i>Plot the sense preprocessing and stacking architecture</i>
---------------------	---

Description

Plot the sense preprocessing and stacking architecture

Usage

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

Arguments

- x A pipeline returned in the ‘plot’ element of [sense()].
- ... Additional arguments passed to ‘graphics::plot.default()’.

Value

The pipeline, invisibly.

sense	sense
-------	-------

Description

Stacked ensembles for regression using mlr3 and internal preprocessing.

Usage

```
sense(  
  df,  
  target_feat,  
  benchmarking = "all",  
  super = "avg",  
  algos = c("glmnet", "ranger", "xgboost", "rpart", "kknn", "svm"),  
  sampling_rate = 1,  
  metric = "mae",  
  collapse_char_to = 10,  
  num_preproc = "scale",  
  fct_preproc = "one-hot",  
  impute_num = "sample",  
  missing_fusion = FALSE,  
  inner = "holdout",  
  outer = "holdout",  
  folds = 3,  
  repeats = 3,  
  ratio = 0.5,  
  selected_filter = "information_gain",  
  selected_n_feats = NULL,  
  tuning = "random_search",  
  budget = 30,  
  resolution = 5,  
  n_evals = 30,  
  minute_time = 10,  
  patience = 0.3,  
  min_improve = 0.01,  
  java_mem = 64,  
  decimals = 2,  
  seed = 42  
)
```

Arguments

df	A data frame with features and target.
target_feat	String. Name of the numeric feature for the regression task.
benchmarking	Positive integer. Number of base learners to stack. Default: "all".

super	String. Super learner of choice among the available learners. Default: "avg".
algos	String vector. Available learners are: "glmnet", "ranger", "xgboost", "rpart", "kkn", "svm".
sampling_rate	Positive numeric. Sampling rate before applying the stacked ensemble. Default: 1.
metric	String. Evaluation metric for outer and inner cross-validation. Default: "mae".
collapse_char_to	Positive integer. Conversion of characters to factors with predefined maximum number of levels. Default: 10.
num_preproc	Numeric preprocessing: "scale", "range" (zero to one), or "nop".
fct_preproc	String. Options for factor pre-processing: "encodeimpact", "encodelmer", "one-hot", "treatment", "poly", "sum", "helmert". Default: "one-hot".
impute_num	String. Options for missing imputation in case of numeric: "sample" or "hist". Default: "sample". For factor the default mode is Out-Of-Range.
missing_fusion	Logical. Add missingness indicators before imputation.
inner	String. Cross-validation inner cycle: "holdout", "cv", "repeated_cv", "subsampling". Default: "holdout".
outer	String. Cross-validation outer cycle: "holdout", "cv", "repeated_cv", "subsampling". Default: "holdout".
folds	Positive integer. Number of repetitions used in "cv" and "repeated_cv". Default: 3.
repeats	Positive integer. Number of repetitions used in "subsampling" and "repeated_cv". Default: 3.
ratio	Positive numeric. Percentage value for "holdout" and "subsampling". Default: 0.5.
selected_filter	String. Filters available for regression tasks: "carscore", "cmim", "correlation", "find_correlation", "information_gain", "relief", "variance". Default: "information_gain".
selected_n_feats	Positive integer. Number of features to select through the chosen filter. Default: NULL.
tuning	String. Available options are "random_search" and "grid_search". Default: "random_search".
budget	Positive integer. Maximum random-search trials, also capped by n_evals.
resolution	Positive integer. Grid resolution for each hyper-parameter. Default: 5.
n_evals	Positive integer. Number of evaluation for termination. Default: 30.
minute_time	Positive integer. Maximum run time before termination. Default: 10.
patience	Positive numeric. Percentage of stagnating evaluations before termination. Default: 0.3.
min_improve	Positive numeric. Minimum error improvement required before termination. Default: 0.01.
java_mem	Retained for backwards compatibility; unused. No Java options are changed.
decimals	Positive integer. Decimal format of prediction. Default: 2.
seed	Positive integer. Default: 42.

Details

Preprocessing and three-fold stacking are implemented internally without `mlr3pipelines`. Each stacking fold learns its own preprocessing state, including supervised encodings and filtering. Benchmark selection is repeated inside each training split. Outer resampling evaluates the complete tuning procedure. Numeric imputation samples from observed training values or their histogram; an entirely missing numeric column uses zero and is removed if constant. Unseen categories use a zero contrast or impact, or the fitted intercept for `encodelmer`. Character levels are collapsed using training frequencies only. Correlation and variance filters are internal. Other filters require the optional `mlr3filters` package and its method-specific backend. Learner backends and `lme4` (only for `encodelmer`) must be installed when requested. Percentage and relative errors retain ordinary R division semantics: zero denominators can produce `Inf` or `NaN`. `test_metrics` uses rounded predictions; `model_error` uses unrounded `mlr3` predictions. Random state is restored after fitting. Imputation at prediction time samples using the caller's random state.

Value

This function returns a list including:

- `benchmark_error`: out-of-fold comparison of base learners from the final fit, or `NULL` when benchmarking is "all".
- `resampled_model`: `mlr3` standard description of the analytic pipeline.
- `plot`: a `sense_pipeline` object; use `plot(result$plot)` for a base R diagram.
- `selected_n_feats`: selected encoded feature names and scores from the first final base model; each base model has its own fitted preprocessing state.
- `model_error`: error measure for outer cycle of cross-validation.
- `testing_frame`: data set used for calculating the test metrics.
- `test_metrics`: metrics reported are `mse`, `rmse`, `mae`, `mape`, `mdae`, `rae`, `rse`, `rrse`, `smape`.
- `model_predict`: prediction function using the ensemble refitted on all supplied data.
- `time_log`: elapsed computation time as a `difftime` in seconds.

Author(s)

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

Maintainer: Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

See Also

Useful links:

- <https://mlr3.ml-org.com/>

Examples

```
## Not run:
sense(benchmark, "y", algos = c("glmnet", "rpart"))

## End(Not run)
```

Index

* **datasets**

benchmark, [2](#)

benchmark, [2](#)

plot.sense_pipeline, [2](#)

sense, [3](#)

sense-package (sense), [3](#)