Package: iBART (via r-universe)

October 10, 2024

Title Iterative Bayesian Additive Regression Trees Descriptor Selection Method

Version 1.0.0

Maintainer Shengbin Ye <sy53@rice.edu>

Description A statistical method based on Bayesian Additive Regression Trees with Global Standard Error Permutation Test (BART-G.SE) for descriptor selection and symbolic regression. It finds the symbolic formula of the regression function y=f(x) as described in Ye, Senftle, and Li (2023) <arXiv:2110.10195>.

URL https://github.com/mattsheng/iBART

BugReports https://github.com/mattsheng/iBART/issues

License GPL (>= 3)

Encoding UTF-8

LazyData true

VignetteBuilder knitr

RoxygenNote 7.2.3

Depends R (>= 4.0.0)

Imports bartMachine (>= 1.2.6), glmnet (>= 4.1-1), foreach, stats

Suggests knitr, rmarkdown, ggplot2, ggpubr

SystemRequirements Java (>= 8.0)

NeedsCompilation no

Author Shengbin Ye [aut, cre, cph] (<https://orcid.org/0000-0001-8767-2595>), Meng Li [aut]

Repository CRAN

Date/Publication 2023-11-14 17:40:02 UTC

Contents

`catalysis` *Single-Atom Catalysis Data*

Description

Single-Atom Catalysis Data

Usage

```
catalysis
```

Format

A list with 4 objects:

X Primary feature matrix: physical properties of transition metals and oxide supports

y Reponse variable: binding energy of metal/oxide pairs

head Column names of X

unit Unit of columns of X

`generate_unit` *A helper function to generate unit for iBART input*

Description

A helper function to generate unit for iBART input

Usage

```
generate_unit(unit, dimension)
```

Arguments

`unit` A vector of unit of the primary features. For example, unit <- c("cm", "s"). Then the unit of $x1$ is centimeter and the unit of $x2$ is second.

`dimension` A vector of dimension of the units. For example, unit <- c("cm", "s") and dimension <- c(2, 1) mean that the unit of $x1$ is square centimeter and the unit of $x2$ is second.

Value

A list that contains unit and dimension information.

`iBART` *iBART descriptor selection*

Description

Finds a symbolic formula for the regression function $y = f(X)$ using (y, X) as inputs.

Usage

```
iBART(
  X = NULL,
  y = NULL,
  head = NULL,
  unit = NULL,
  BART_var_sel_method = "global_se",
  num_trees = 20,
  num_burn_in = 10000,
  num_iterations_after_burn_in = 5000,
  num_reps_for_avg = 10,
  num_permute_samples = 50,
  type.measure = "deviance",
  nfolds = 10,
  nlambda = 100,
  relax = FALSE,
  gamma = c(0, 0.25, 0.5, 0.75, 1),
  opt = c("binary", "unary", "binary"),
  sin_cos = FALSE,
  apply_pos_opt_on_neg_x = TRUE,
  hold = 0,
  pre_screen = TRUE,
  corr_screen = TRUE,
  out_sample = FALSE,
  train_idx = NULL,
  train_ratio = 1,
  Lzero = TRUE,
  parallel = FALSE,
  K = ifelse(Lzero, 5, 0),
  aic = FALSE,
  standardize = TRUE,
  writeLog = FALSE,
  verbose = TRUE,
  count = NULL,
  seed = NULL
)
```

Arguments

`X` Input matrix of primary features X.

`y` Response variable y.

`head` Optional: name of primary features.

`unit` Optional: units and their respective dimensions of primary features. This is used to perform dimension analysis for generated descriptors to avoid generating unphyiscal descriptors, such as $size + size^2$. See `generate_dimension()` for details.

`BART_var_sel_method`
Variable selection criterion used in BART. Three options are available: (1) "global_se", (2) "global_max", (3) "local". The default is "global_se". See `var_selection_by_permute` in `R` package `bartMachine` for more detail.

`num_trees` BART parameter: number of trees to be grown in the sum-of-trees model. If you want different values for each iteration of BART, input a vector of length equal to number of iterations. Default is `num_trees = 20`.

`num_burn_in` BART parameter: number of MCMC samples to be discarded as “burn-in". If you want different values for each iteration of BART, input a vector of length equal to number of iterations. Default is `num_burn_in = 10000`.

`num_iterations_after_burn_in`
BART parameter: number of MCMC samples to draw from the posterior distribution of $hatf(x)$. If you want different values for each iteration of BART, input a vector of length equal to number of iterations. Default is `num_iterations_after_burn_in = 5000`.

`num_reps_for_avg`
BART parameter: number of replicates to over over to for the BART model’s variable inclusion proportions. If you want different values for each iteration of BART, input a vector of length equal to number of iterations. Default is `num_reps_for_avg = 10`.

`num_permute_samples`
BART parameter: number of permutations of the response to be made to generate the “null” permutation distribution. If you want different values for each iteration of BART, input a vector of length equal to number of iterations. Default is `num_permute_samples = 50`.

`type.measure` `glmnet` parameter: loss to use for cross-validation. The default is `type.measure="deviance"`, which uses squared-error for Gaussian models (a.k.a `type.measure="mse" there`). `type.measure="mae"` (mean absolute error) can be used also.

`nfolds` glmnet parameter: number of folds - default is 10. Smallest value allowable is nfolds=3.

`nlambda` glmnet parameter: the number of lambda values - default is 100.

`relax` glmnet parameter: If TRUE, then CV is done with respect to the mixing parameter gamma as well as lambda. Default is relax=FALSE.

`gamma` glmnet parameter: the values of the parameter for mixing the relaxed fit with the regularized fit, between 0 and 1; default is gamma = c(0, 0.25, 0.5, 0.75, 1)

`opt` A vector of operation order. For example, opt = c("unary", "binary", "unary") will apply unary operators, then binary operators, then unary operators. Available operator sets are "unary", "binary", and "all", where "all" is the union of "unary" and "binary".

`sin_cos` Logical flag for using $sin(\pi * x)$ and $cos(\pi * x)$ to generate descriptors. This is useful if you think there is periodic relationship between predictors and response. Default is sin_cos = FALSE.

`apply_pos_opt_on_neg_x`
Logical flag for applying non-negative-valued operators, such as $\sqrt{x}$ and $log(x)$, when some values of x is negative. If apply_pos_opt_on_neg_x == TRUE, apply absolute value operator first then non-negative-valued operator, i.e. generate $\sqrt{|x|}$ and $log(|x|)$ instead. Default is apply_pos_opt_on_neg_x = TRUE.

`hold` Number of iterations to hold. This allows iBART to run consecutive operator transformations before screening. Note hold = 0 is equivalent to no skipping of variable selection in each iBART iterations. It should be less than iter.

`pre_screen` Logical flag for pre-screening the primary features X using BART. Only selected primary features will be used to generate descriptors. Note that pre_screen = FALSE is equivalent to hold = 1.

`corr_screen` Logical flag for screening out primary features that are independet of the response variable y.

`out_sample` Logical flag for out of sample assessment. Default is out_sample = FALSE.

`train_idx` Numerical vector storing the row indices for training data. Please set out_sample = TRUE if you supplied train_idx.

`train_ratio` Proportion of data used to train model. Value must be between (0,1]. This is only needed when out_sample = TRUE and train_idx == NULL. Default is train_ratio = 1.

`Lzero` Logical flag for L-zero variable selection. Default is Lzero = TRUE.

`parallel` Logical flag for parallel L-zero variable selection. Default is parallel = FALSE.

`K` If Lzero == TRUE, K sets the maximum number of descriptors to be selected.

`aic` If Lzero == TRUE, logical flag for selecting best number of descriptors using AIC. Possible number of descriptors are $1 \leq k \leq K$.

`standardize` Logical flag for data standardization prior to model fitting in BART and LASSO. Default is standardize = TRUE.

`writeLog` Logical flag for writing log file to working directory. The log file will contain information such as the descriptors selected by iBART, RMSE of the linear model build on the selected descriptors, etc. Default is writeLog = FALSE.

`verbose` Logical flag for printing progress to console. Default is `verbose = TRUE`.

`count` Internal parameter. Default is `count = NULL`.

`seed` Optional: sets the seed in both R and Java. Default is `seed = NULL` which does not set the seed in R nor Java.

Value

A list of iBART output.

`iBART_model` The LASSO output of the last iteration of iBART. The predictors with non-zero coefficient are called the iBART selected descriptors.

`X_selected` The numerical values of the iBART selected descriptors.

`descriptor_names` The names of the iBART selected descriptors.

`coefficients` Coefficients of the iBART model. The first element is an intercept.

`X_train` The training matrix used in the last iteration.

`X_test` The testing matrix used in the last iteration.

`iBART_gen_size` The number of descriptors generated by iBART in each iteration.

`iBART_sel_size` The number of descriptors selected by iBART in each iteration.

`iBART_in_sample_RMSE` In sample RMSE of the LASSO model.

`iBART_out_sample_RMSE` Out of sample RMSE of the LASSO model if `out_sample == TRUE`.

`Lzero_models` The l_0-penalized regression models fitted on the iBART selected descriptors for $1 \leq k \leq K$.

`Lzero_names` The name of the best kD descriptors selected by the l_0-penalized regression model for $1 \leq k \leq K$.

`Lzero_in_sample_RMSE` In sample RMSE of the l_0-penalized regression model for $1 \leq k \leq K$.

`Lzero_out_sample_RMSE` Out of sample RMSE of the l_0-penalized regression model for $1 \leq k \leq K$ if `out_sample == TRUE`.

`Lzero_AIC_model` The best l_0-penalized regression model selected by AIC.

`Lzero_AIC_names` The best kD descriptors where $1 \leq k \leq K$ is chosen via AIC.

`Lzero_AIC_in_sample_RMSE` In sample RMSE of the best l_0-penalized regression models chosen by AIC.

`Lzero_AIC_out_sample_RMSE` Out of sample RMSE of the best l_0-penalized regression models chosen by AIC if `out_sample == TRUE`.

`runtime` Runtime in second.

Author(s)

Shengbin Ye

References

Ye, S., Senftle, T.P., and Li, M. (2023) *Operator-induced structural variable selection for identifying materials genes*, `https://arxiv.org/abs/2110.10195`.

`iBART_real_data` *iBART Real Data Result*

Description

iBART result in the real data vignette

Usage

```
iBART_real_data
```

Format

A list of iBART outputs

iBART_model A cv.glmnet object storing the iBART selected model ...

`iBART_sim` *iBART Simulation Result*

Description

iBART result in the simulation vignette

Usage

```
iBART_sim
```

Format

A list of iBART outputs

iBART_model A cv.glmnet object storing the iBART selected model ...

`k_var_model` *Best subset selection for linear regression*

Description

Best subset selection for linear regression

Usage

```
k_var_model(
  X_train,
  y_train,
  X_test = NULL,
  y_test = NULL,
  k = 1,
  parallel = FALSE
)
```

Arguments

`X_train` The design matrix used during training.

`y_train` The response variable used during training.

`X_test` The design matrix used during testing. Default is `X_test = NULL` and full data will be used to train the best subset linear regression model.

`y_test` The response variable used during testing. Default is `y_test = NULL` and full data will be used to train the best subset linear regression model.

`k` The maximum number of predictors allowed in the model. For example, `k = 5` will produce the best model 5 predictors.

`parallel` Logical flag for parallelization. Default is `parallel = FALSE`.

Value

A list of outputs.

`models` An `lm` object storing the best k-predictor linear model.

`names` The variable name of the best k predictors.

`rmse_in` In-sample RMSE of the model.

`rmse_out` Out-of-sample RMSE of the model.

Index