

Package ‘plotBart’

May 27, 2022

Type Package

Title Diagnostic and Plotting Functions to Supplement 'bartCause'

Version 0.1.7

Description

Functions to assist in diagnostics and plotting during the causal inference modeling process.
Supplements the 'bartCause' package.

License MIT + file LICENSE

URL <https://priism-center.github.io/plotBart/>,
<https://github.com/priism-center/plotBart>

BugReports <https://github.com/priism-center/plotBart/issues>

Encoding UTF-8

LazyData true

RoxygenNote 7.2.0

Depends R (>= 2.10), bartCause (>= 1.0.4), ggplot2 (>= 3.3.2)

Imports dplyr (>= 1.0.5), tidyr (>= 1.1.3), rpart (>= 4.1.15), stats
(>= 3.6.2), ggdendro (>= 0.1.22)

Suggests testthat, vdiffr, arm (>= 1.10.1), covr, knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Joseph Marlo [aut, cre],
George Perrett [aut]

Maintainer Joseph Marlo <jpm770@nyu.edu>

Repository CRAN

Date/Publication 2022-05-27 07:50:06 UTC

R topics documented:

lalonge	2
plot_balance	3
plot_CATE	3
plot_common_support	5
plot_ICATE	6
plot_moderator_c_loess	7
plot_moderator_c_pd	8
plot_moderator_d_density	9
plot_moderator_d_linrange	10
plot_moderator_search	11
plot_overlap_pScores	12
plot_overlap_vars	13
plot_PATE	14
plot_SATE	15
plot_trace	16
plot_waterfall	17
Index	19

lalonge	<i>Lalonge dataset</i>
---------	------------------------

Description

Lalonge dataset

Usage

lalonge

Format

An object of class `data.frame` with 445 rows and 12 columns.

Source

<https://CRAN.R-project.org/package=arm>

plot_balance	<i>Plot the balance</i>
--------------	-------------------------

Description

Visualize balance of variables between treatment and control groups. Balance plot reflects balance in standardized units.

Usage

```
plot_balance(.data, treatment, confounders)
```

Arguments

.data	dataframe
treatment	the column denoted treatment. Must be binary.
confounders	character list of column names denoting the X columns of interest

Value

ggplot object

Author(s)

Joseph Marlo

Examples

```
data(lalonde)
plot_balance(lalonde, 'treat', c('re78', 'age', 'educ')) + labs(title = 'My new title')
```

plot_CATE	<i>Plot the histogram or density of the Conditional Average Treatment Effect</i>
-----------	--

Description

Plot the conditional average treatment effect (CATE) of a 'bartCause' model. The conditional average treatment effect is derived from taking the difference between predictions for each individual under the control condition and under the treatment condition averaged over the population. Means of the CATE distribution will resemble SATE and PATE but the CATE distribution accounts for more uncertainty than SATE and less uncertainty than PATE.

Usage

```
plot_CATE(
  .model,
  type = c("histogram", "density"),
  ci_80 = FALSE,
  ci_95 = FALSE,
  reference = NULL,
  .mean = FALSE,
  .median = FALSE
)
```

Arguments

<code>.model</code>	a model produced by <code>'bartCause::bartc()'</code>
<code>type</code>	histogram or density
<code>ci_80</code>	TRUE/FALSE. Show the 80% credible interval?
<code>ci_95</code>	TRUE/FALSE. Show the 95% credible interval?
<code>reference</code>	numeric. Show a vertical reference line at this value
<code>.mean</code>	TRUE/FALSE. Show the mean reference line
<code>.median</code>	TRUE/FALSE. Show the median reference line

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hispanic', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSup.rule = 'none'
)
plot_CATE(model_results)
```

plot_common_support *Plot common support based on the standard deviation rule, chi squared rule, or both*

Description

Plot common support based on the standard deviation rule, chi squared rule, or both.

Usage

```
plot_common_support(.model, rule = c("both", "sd", "chi"))
```

Arguments

`.model` a model produced by `'bartCause::bartc()'`
`rule` one of `c('both', 'sd', 'chi')` denoting which rule to use to identify lack of support

Details

Sufficient overlap/common support is an assumption of causal inference. BART models use the uncertainty of counterfactual uncertainty. When the posterior distribution of an individual's counterfactual prediction extends beyond a specified cut-point, that point likely has insufficient common support. 'bartCause' model offer the option to automatically remove points without common support from analyses, however, this must be specified during model fitting. Cut-points are determined through one of two rules: the standard deviation (sd) or chi-squared (chi). Under the standard deviation rule, a point has weak common support if its posterior distribution of the counterfactual deviation is greater than the maximum posterior of the observed predictions with 1 standard deviation of the distribution of standard deviations for each individual's predicted outcome under the observed assignment. Under the chi-squared rule, a point is discarded if the variance between its counterfactual prediction over observed prediction are statistically different under a chi-squared distribution with 1 degree of freedom. For more details on discard rules see Hill and Su 2013.

When called this plot will show how many points would have been removed under the standard deviation and chi-squared rules. This plot should be used as a diagnostic for 'bartCause' models fit without a common-support rule.

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

References

Hill, J., & Su, Y. S. (2013). Assessing lack of common support in causal inference using Bayesian nonparametrics: Implications for evaluating the effect of breastfeeding on children's cognitive outcomes. *The Annals of Applied Statistics*, 1386-1420.

Examples

```

data(lalonde)
confounders <- c('age', 'educ', 'black', 'hisp', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSuprule = 'none'
)
plot_common_support(model_results)

```

plot_ICATE

Plot Individual Conditional Average Treatment effects

Description

Plots a histogram of Individual Conditional Average Treatment effects (ICATE). ICATEs are the difference in each individual's predicted outcome under the treatment and predicted outcome under the control averaged over the individual. Plots of ICATEs are useful to identify potential heterogeneous treatment effects between different individuals. ICATE plots can be grouped by discrete variables.

Usage

```
plot_ICATE(.model, .group_by = NULL, n_bins = 30, .alpha = 0.7)
```

Arguments

<code>.model</code>	a model produced by <code>'bartCause::bartc()'</code>
<code>.group_by</code>	a grouping variable as a vector
<code>n_bins</code>	number of bins
<code>.alpha</code>	transparency of histograms

Value

ggplot object

Author(s)

George Perrett

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hispanic', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSup.rule = 'none'
)
plot_ICATE(model_results, lalonde$married)
```

plot_moderator_c_loess

LOESS plot of a continuous moderating variable

Description

Plot the LOESS prediction of ICATEs by a continuous covariate. This is an alternative to partial dependency plots to assess treatment effect heterogeneity by a continuous covariate. See Carnegie, Dorie and Hill 2019.

Usage

```
plot_moderator_c_loess(.model, moderator, line_color = "blue")
```

Arguments

.model	a model produced by 'bartCause::bartc()'
moderator	the moderator as a vector
line_color	the color of the loess line

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

References

Carnegie, N., Dorie, V., & Hill, J. L. (2019). Examining treatment effect heterogeneity using BART. *Observational Studies*, 5(2), 52-70.

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hisp', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSuprule = 'none'
)
plot_moderator_c_loess(model_results, lalonde$age)
```

plot_moderator_c_pd *Partial dependency plot of a continuous moderating variable*

Description

Plot a partial dependency plot with a continuous covariate from a 'bartCause' model. Identify treatment effect variation predicted across levels of a continuous variable.

Usage

```
plot_moderator_c_pd(.model, moderator, n_bins = NULL)
```

Arguments

.model	a model produced by 'bartCause::bartc()'
moderator	the moderator as a vector
n_bins	number of bins to cut the moderator with. Defaults to the lesser of 15 and number of distinct levels of the moderator

Details

Partial dependency plots are one way to evaluate heterogeneous treatment effects that vary by values of a continuous covariate. For more information on partial dependency plots from BART causal inference models see Green and Kern 2012.

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

References

Green, D. P., & Kern, H. L. (2012). Modeling heterogeneous treatment effects in survey experiments with Bayesian additive regression trees. *Public opinion quarterly*, 76(3), 491-511.

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hisp', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSuprule = 'none',
  keepTrees = TRUE
)
plot_moderator_c_pd(model_results, lalonde$age)
```

plot_moderator_d_density

Plot the Conditional Average Treatment Effect conditional on a discrete moderator

Description

Plot the Conditional Average Treatment Effect split by a discrete moderating variable. This plot will provide a visual test of moderation by discrete variables.

Usage

```
plot_moderator_d_density(
  .model,
  moderator,
  .alpha = 0.7,
  facet = FALSE,
  .ncol = 1
)
```

Arguments

.model	a model produced by 'bartCause::bartc()'
moderator	the moderator as a vector
.alpha	transparency value [0, 1]
facet	TRUE/FALSE. Create panel plots of each moderator level?
.ncol	number of columns to use when faceting

Value

ggplot object

Author(s)

George Perrett

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hisp', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSuprule = 'none'
)
plot_moderator_d_density(model_results, lalonde$educ)
```

plot_moderator_d_linerange

Plot the posterior interval of the Conditional Average Treatment Effect grouped by a discrete variable

Description

Plots the range of the Conditional Average Treatment Effect grouped by a discrete variable. This is analogous to plot_moderator_d_density but is preferable for moderators with many categories. Rather than plotting the full density, the posterior range is shown.

Usage

```
plot_moderator_d_linerange(.model, moderator, .alpha = 0.7, horizontal = FALSE)
```

Arguments

.model	a model produced by ‘bartCause::bartc()’
moderator	the moderator as a vector
.alpha	transparency value [0, 1]
horizontal	flip the plot horizontal?

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hispanic', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSuprule = 'none'
)
plot_moderator_d_linerange(model_results, lalonde$educ)
```

plot_moderator_search *Plot a single regression tree of covariates on ICATEs*

Description

Plot a single regression tree for exploratory heterogeneous effects. Fit single regression tree on bartc() ICATEs to produce variable importance plot. This plot is useful for identifying potential moderating variables. Tree depth may be set to depths 1, 2 or 3. Terminal nodes signal the Conditional Average Treatment effect within levels of moderation variables. Trees with different values across terminal nodes suggest strong treatment effect moderation.

Usage

```
plot_moderator_search(.model, max_depth = c(2, 1, 3))
```

Arguments

.model	a model produced by 'bartCause::bartc()'
max_depth	one of c(1, 2, 3). Maximum number of node levels within the tree. 2 is recommended

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

Examples

```

data(lalonde)
confounders <- c('age', 'educ', 'black', 'hispanic', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSuprule = 'none'
)
plot_moderator_search(model_results)

```

plot_overlap_pScores *Plot the overlap via propensity score method*

Description

Plot histograms showing the overlap between propensity scores by treatment status.

Usage

```

plot_overlap_pScores(
  .data,
  treatment,
  response,
  confounders,
  plot_type = c("histogram", "density"),
  pscores = NULL,
  ...
)

```

Arguments

.data	dataframe
treatment	character. Name of the treatment column within .data
response	character. Name of the response column within .data
confounders	character list of column names denoting confounders within .data
plot_type	the plot type, one of c('Histogram', 'Density')
pscores	propensity scores. If not provided, then propensity scores will be calculated using BART
...	additional arguments passed to 'bartCause::bartc' propensity score calculation

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

See Also

[plot_overlap_vars](#)

Examples

```
data(lalonde)
plot_overlap_pScores(
  .data = lalonde,
  treatment = 'treat',
  response = 're78',
  confounders = c('age', 'educ'),
  plot_type = 'histogram',
  pcores = NULL,
  seed = 44
)
```

plot_overlap_vars *Plot the overlap of variables*

Description

Plot histograms showing the overlap between variables by treatment status.

Usage

```
plot_overlap_vars(
  .data,
  treatment,
  confounders,
  plot_type = c("histogram", "density")
)
```

Arguments

.data	dataframe
treatment	character. Name of the treatment column within .data
confounders	character list of column names denoting confounders within .data
plot_type	the plot type, one of c('histogram', 'density'). Defaults to 'histogram'

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

See Also[plot_overlap_pScores](#)**Examples**

```
data(lalonde)
plot_overlap_vars(
  .data = lalonde,
  treatment = 'treat',
  confounders = c('age', 'educ'),
  plot_type = 'Histogram'
)
```

plot_PATE

Plot histogram or density of Population Average Treatment Effect

Description

Plot shows the Population Average Treatment Effect which is derived from the posterior predictive distribution of the difference between $y|z = 1, X$ and $y|z = 0, X$. Mean of PATE will resemble CATE and SATE but PATE will account for more uncertainty and is recommended for informing inferences on the average treatment effect.

Usage

```
plot_PATE(
  .model,
  type = c("histogram", "density"),
  ci_80 = FALSE,
  ci_95 = FALSE,
  reference = NULL,
  .mean = FALSE,
  .median = FALSE
)
```

Arguments

.model	a model produced by 'bartCause::bartc()'
type	histogram or density
ci_80	TRUE/FALSE. Show the 80% credible interval?
ci_95	TRUE/FALSE. Show the 95% credible interval?
reference	numeric. Show a vertical reference line at this value
.mean	TRUE/FALSE. Show the mean reference line
.median	TRUE/FALSE. Show the median reference line

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hispanic', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSup.rule = 'none'
)
plot_PATE(model_results)
```

plot_SATE

Plot histogram or density of Sample Average Treatment Effects

Description

Plot a histogram or density of the Sample Average Treatment Effect (SATE). The Sample Average Treatment Effect is derived from taking the difference of each individual's observed outcome and a predicted counterfactual outcome from a BART model averaged over the population. The mean of SATE will resemble means of CATE and PATE but will account for the least uncertainty.

Usage

```
plot_SATE(
  .model,
  type = c("histogram", "density"),
  ci_80 = FALSE,
  ci_95 = FALSE,
  reference = NULL,
  .mean = FALSE,
  .median = FALSE
)
```

Arguments

.model	a model produced by <code>'bartCause::bartc()'</code>
type	histogram or density
ci_80	TRUE/FALSE. Show the 80% credible interval?
ci_95	TRUE/FALSE. Show the 95% credible interval?
reference	numeric. Show a vertical reference line at this x-axis value
.mean	TRUE/FALSE. Show the mean reference line
.median	TRUE/FALSE. Show the median reference line

Value

ggplot object

Author(s)

George Perrett, Joseph Marlo

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hispanic', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSup.rule = 'none'
)
plot_SATE(model_results)
```

plot_trace

Trace plot the estimands of a 'bartCause::bartc()' model

Description

Returns a ggplot of the estimated effect over each iteration of the model fit. This is used to visually assess the convergence of Markov chain Monte Carlo (MCMC) sampling. Chains should be well mixed such that no single color is notably separate from others.

Usage

```
plot_trace(.model)
```


Arguments

`.model` a model produced by `'bartCause::bartc()'`

Value

ggplot object

Author(s)

Joseph Marlo, George Perrett

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hisp', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSup.rule = 'none'
)
plot_trace(.model = model_results)
```

plot_waterfall *Plot a waterfall of the ICATEs*

Description

Plots the point and posterior intervals of each individual's ICATE ordered by the ICATE or a continuous variable. Points can be colored by a discrete variable. Waterfall plots are a useful visual diagnostic of possible treatment effect heterogeneity. A flat line implies little treatment effect heterogeneity while a steeper curve implies that the treatment effect varies across individuals in the sample. Ordering points by a continuous variable or coloring points by a discrete variable can be helpful to identify potential moderators of the treatment effect.

Usage

```
plot_waterfall(
  .model,
  descending = TRUE,
  .order = NULL,
  .color = NULL,
  .alpha = 0.5
)
```

Arguments

<code>.model</code>	a model produced by <code>'bartCause::bartc()'</code>
<code>descending</code>	order the ICATEs by value?
<code>.order</code>	a vector representing a custom order
<code>.color</code>	a vector representing colors
<code>.alpha</code>	transparency value [0, 1]

Value

ggplot object

Author(s)

George Perrett

Examples

```
data(lalonde)
confounders <- c('age', 'educ', 'black', 'hisp', 'married', 'nodegr')
model_results <- bartCause::bartc(
  response = lalonde[['re78']],
  treatment = lalonde[['treat']],
  confounders = as.matrix(lalonde[, confounders]),
  estimand = 'ate',
  commonSuprule = 'none'
)
plot_waterfall(model_results)
```

Index

* datasets

lalonde, [2](#)

lalonde, [2](#)

plot_balance, [3](#)

plot_CATE, [3](#)

plot_common_support, [5](#)

plot_ICATE, [6](#)

plot_moderator_c_loess, [7](#)

plot_moderator_c_pd, [8](#)

plot_moderator_d_density, [9](#)

plot_moderator_d_linerange, [10](#)

plot_moderator_search, [11](#)

plot_overlap_pScores, [12](#), [14](#)

plot_overlap_vars, [13](#), [13](#)

plot_PATE, [14](#)

plot_SATE, [15](#)

plot_trace, [16](#)

plot_waterfall, [17](#)