Package ‘workboots’

October 12, 2022

Title Generate Bootstrap Prediction Intervals from a 'tidymodels' Workflow

Version 0.2.0

Description Provides functions for generating bootstrap prediction intervals from a 'tidymodels' workflow. 'tidymodels' <https://www.tidymodels.org/> is a collection of packages for modeling and machine learning using 'tidyverse' <https://www.tidyverse.org/> principles. This package is not affiliated with or maintained by 'RStudio' or the 'tidymodels' maintainers.

License MIT + file LICENSE


BugReports https://github.com/markjrieke/workboots/issues

Imports assertthat, dplyr, generics, lifecycle, Metrics, purrr, rlang, rsample, stats, tibble, tidyr, vip, workflows

Encoding UTF-8

RoxygenNote 7.1.2

Suggestsforcats, ggplot2, knitr, readr, recipes, rmarkdown, scales, testthat (>= 3.0.0), tidymodels, tune, xgboost

VignetteBuilder knitr

Config/testthat/edition 3

Depends R (>= 2.10)

NeedsCompilation no

Author Mark Rieke [aut, cre]

Maintainer Mark Rieke <markjrieke@gmail.com>

Repository CRAN

Date/Publication 2022-05-16 06:40:02 UTC
**predict_boots**

*fit and predict from a workflow using many bootstrap resamples.*

**Description**

Generate a prediction interval from arbitrary model types using bootstrap resampling. `predict_boots()` generates `n` bootstrap resamples, fits a model to each resample (creating `n` models), then creates `n` predictions for each observation in `new_data`.

**Usage**

```r
predict_boots(
  workflow,
  n = 2000,
  training_data,
  new_data,
  interval = c("prediction", "confidence"),
  verbose = FALSE,
  ...
)
```

**Arguments**

- `workflow` An un-fitted workflow object.
- `n` An integer for the number of bootstrap resampled models that will be created.
- `training_data` A tibble or dataframe of data to be resampled and used for training.
- `new_data` A tibble or dataframe used to make predictions.
- `interval` One of prediction, confidence. Specifies the interval type to be generated.
- `verbose` A logical. Defaults to FALSE. If set to TRUE, prints progress of training to console.
- `...` Additional params passed to `rsample::bootstraps()`.

**Details**

Since `predict_boots()` fits a new model to each resample, the argument `workflow` must not yet be fit. Any tuned hyperparameters must be finalized prior to calling `predict_boots()`.
Value

A tibble with a column indicating the row index of each observation in `new_data` and a nested list of the model predictions for each observation.

Examples

```r
## Not run:
library(tidymodels)

# setup a workflow without fitting
wf <-
  workflow() %>%
  add_recipe(recipe(qsec ~ wt, data = mtcars)) %>%
  add_model(linear_reg())

# fit and predict 2000 bootstrap resampled models to mtcars
set.seed(123)
wf %>%
  predict_boots(n = 2000, training_data = mtcars, new_data = mtcars)
## End(Not run)
```

summarise_importance

Append a tibble of variable importances returned by `vi_boots()` with upper and lower bounds.

Description

Append a tibble of variable importances returned by `vi_boots()` with upper and lower bounds.

Usage

```r
summarise_importance(.data, interval_width = 0.95, conf = NULL)
```

Arguments

- `.data` a tibble of variable importances returned by `vi_boots()`.
- `interval_width` a value between (0, 1) specifying the interval range.
- `conf` deprecated - please use `interval_width` instead.

Details

Generates a summary of variable importances with an upper and lower interval range. Uses the `vi()` function from the `{vip}` package to compute variable importances (not all model types are supported by `vip::vi()`: please refer to `{vip}` package documentation for supported model types). Presently, the `quantile()` function from the `{stats}` package is used to determine the lower, 50th percentile, and upper interval ranges.
Examples

```r
## Not run:
library(tidymodels)

# setup a workflow without fitting
wf <-
  workflow() %>%
  add_recipe(recipe(qsec ~ wt, data = mtcars)) %>%
  add_model(linear_reg())

# evaluate variable importance from 2000 models fit to mtcars
set.seed(123)
importances <-
  wf %>%
  vi_boots(n = 2000, training_data = mtcars, new_data = mtcars)

# append with lower and upper bound importance summary columns
importances %>%
  summarise_importance(interval_width = 0.95)

## End(Not run)
```

---

**summarise_predictions**

*Append a tibble of predictions returned by predict_boots() with upper and lower bounds.*

### Description

Append a tibble of predictions returned by `predict_boots()` with upper and lower bounds.

### Usage

```r
summarise_predictions(.data, interval_width = 0.95, conf = NULL)
```

### Arguments

- `.data`:
  - A tibble of predictions returned by `predict_boots()`.
- `interval_width`:
  - A value between (0, 1) specifying the interval range.
- `conf`:
  - Deprecated - please use `interval_width` instead.

### Details

Generates a summary of predictions with a upper and lower interval range. Presently, the `quantile()` function from the `{stats}` package is used to determine the lower, 50th percentile, and upper interval ranges.
vi_boots

**Value**

Appends the tibble of predictions returned by `predict_boots()` with three new columns: `.pred_lower`, `.pred`, and `.pred_upper`.

**Examples**

```r
## Not run:
library(tidymodels)

# setup a workflow without fitting
wf <- workflow() %>%
  add_recipe(recipe(qsec ~ wt, data = mtcars)) %>%
  add_model(linear_reg())

# fit and predict 2000 bootstrap resampled models to mtcars
set.seed(123)
preds <- wf %>%
  predict_boots(n = 2000, training_data = mtcars, new_data = mtcars)

# append with prediction interval summary columns
preds %>%
  summarise_predictions(conf = 0.95)
## End(Not run)
```

---

**vi_boots** *Fit and estimate variable importance from a workflow using many bootstrap resamples.*

**Description**

Generate variable importances from a tidymodel workflow using bootstrap resampling. `vi_boots()` generates `n` bootstrap resamples, fits a model to each (creating `n` models), then creates `n` estimates of variable importance for each variable in the model.

**Usage**

`vi_boots(workflow, n = 2000, training_data, verbose = FALSE, ...)`

**Arguments**

- **workflow** *An un-fitted workflow object.*
- **n** *An integer for the number of bootstrap resampled models that will be created.*
- **training_data** *A tibble or dataframe of data to be resampled and used for training.*
- **verbose** *A logical. Defaults to FALSE. If set to TRUE, prints progress of training to console.*
- **...** *Additional params passed to `rsample::bootstraps()`.*
Details

Since `vi_boots()` fits a new model to each resample, the argument `workflow` must not yet be fit. Any tuned hyperparameters must be finalized prior to calling `vi_boots()`.

Value

A tibble with a column indicating each variable in the model and a nested list of variable importances for each variable. The shape of the list may vary by model type. For example, linear models return two nested columns: the absolute value of each variable’s importance and the sign (POS/NEG), whereas tree-based models return a single nested column of variable importance. Similarly, the number of nested rows may vary by model type as some models may not utilize every possible predictor.

Examples

```r
## Not run:
library(tidymodels)

# setup a workflow without fitting
wf <- workflow() %>%
  add_recipe(recipe(qsec ~ wt, data = mtcars)) %>%
  add_model(linear_reg())

# fit and estimate variable importance from 125 bootstrap resampled models
set.seed(123)
wf %>%
  vi_boots(n = 2000, training_data = mtcars)

## End(Not run)
```
Index

predict_boots, 2
summarise_importance, 3
summarise_predictions, 4
summarize_importance
  (summarise_importance), 3
summarize_predictions
  (summarise_predictions), 4
vi_boots, 5