Package: audrex (via r-universe)

February 22, 2025

Type Package

Title Automatic Dynamic Regression using Extreme Gradient Boosting

Version 2.0.1

Author Giancarlo Vercellino

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

Description Dynamic regression for time series using Extreme Gradient Boosting with hyper-parameter tuning via Bayesian Optimization or Random Search.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Depends R (>= 4.1)

Imports rBayesianOptimization (>= 1.2.0), xgboost (>= 1.4.1.1), purrr (>= 0.3.4), ggplot2 (>= 3.3.5), readr (>= 2.1.2), stringr (>= 1.4.0), lubridate (>= 1.7.10), narray (>= 0.4.1.1), fANCOVA (>= 0.6-1), imputeTS (>= 3.2), scales (>= 1.1.1), tictoc (>= 1.0.1), modeest (>= 2.4.0), moments (>= 0.14), Metrics (>= 0.1.4), parallel (>= 4.1.1), utils (>= 4.1.1), stats (>= 4.1.1)

URL https://rpubs.com/giancarlo_vercellino/audrex

NeedsCompilation no

Date/Publication 2022-03-23 10:10:14 UTC

Config/pak/sysreqs make libicu-dev libjpeg-dev libpng-dev libxml2-dev libssl-dev libx11-dev

Repository https://pigian.r-universe.dev

RemoteUrl https://github.com/cran/audrex

RemoteRef HEAD

RemoteSha 7ae3287a012d4183bd0406d60097a8ccfc698899

Contents

---

audrex *audrex: Automatic Dynamic Regression using Extreme Gradient Boosting*

Description

Dynamic regression for time series using Extreme Gradient Boosting with hyper-parameter tuning via Bayesian Optimization or Random Search.

Usage

```
audrex(
  data,
  n_sample = 10,
  n_search = 5,
  smoother = FALSE,
  seq_len = NULL,
  diff_threshold = 0.001,
  booster = "gbtree",
  norm = NULL,
  n_dim = NULL,
  ci = 0.8,
  min_set = 30,
  max_depth = NULL,
  eta = NULL,
  gamma = NULL,
  min_child_weight = NULL,
  subsample = NULL,
  colsample_bytree = NULL,
  lambda = NULL,
  alpha = NULL,
  n_windows = 3,
  patience = 0.1,
  nrounds = 100,
  dates = NULL,
  acq = "ucb",
  kappa = 2.576,
  eps = 0,
```

```
  kernel = list(type = "exponential", power = 2),
  seed = 42
)
```

Arguments

data	A data frame with time features on columns.
n_sample	Positive integer. Number of samples for the Bayesian Optimization. Default: 10.
n_search	Positive integer. Number of search steps for the Bayesian Optimization. When the parameter is set to 0, optimization is shifted to Random Search. Default: 5,
smoother	Logical. Perform optimal smoothing using standard loess. Default: FALSE
seq_len	Positive integer. Number of time-steps to be predicted. Default: NULL (automatic selection)
diff_threshold	Positive numeric. Minimum F-test threshold for differentiating each time feature (keep it low). Default: 0.001.
booster	String. Optimization methods available are: "gbtree", "gblinear". Default: "gbtree".
norm	Logical. Boolean flag to apply Yeo-Johson normalization. Default: NULL (automatic selection from random search or bayesian search).
n_dim	Positive integer. Projection of time features in a lower dimensional space with n_dim features. The default value (NULL) sets automatically the values in c(1, n features).
ci	Confidence interval. Default: 0.8.
min_set	Positive integer. Minimun number for validation set in case of automatic resize of past dimension. Default: 30.
max_depth	Positive integer. Look to xgboost documentation for description. A vector with one or two positive integer for the search boundaries. The default value (NULL) sets automatically the values in c(1, 8).
eta	Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric between (0, 1] for the search boundaries. The default value (NULL) sets automatically the values in c(0, 1).
gamma	Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric for the search boundaries. The default value (NULL) sets automatically the values in c(0, 100).
min_child_weight	Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric for the search boundaries. The default value (NULL) sets automatically the values in c(0, 100).
subsample	Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric between (0, 1] for the search boundaries. The default value (NULL) sets automatically the values in c(0, 1).

`colsample_bytree`
: Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric between (0, 1] for the search boundaries. The default value (NULL) sets automatically the values in c(0, 1).

`lambda`
: Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric for the search boundaries. The default value (NULL) sets automatically the values in c(0, 100).

`alpha`
: Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric for the search boundaries. The default value (NULL) sets automatically the values in c(0, 100).

`n_windows`
: Positive integer. Number of (expanding) windows for cross-validation. Default: 3.

`patience`
: Positive numeric. Percentage of waiting rounds without improvement before xgboost stops. Default: 0.1

`nrounds`
: Positive numeric. Number of round for the extreme boosting machine. Look to xgboost for description. Default: 100.

`dates`
: Date. Vector of dates for the time series. Default: NULL (progressive numbers).

`acq`
: String. Parameter for Bayesian Optimization. For reference see rBayesianOptimization documentation. Default: "ucb".

`kappa`
: Positive numeric. Parameter for Bayesian Optimization. For reference see rBayesianOptimization documentation. Default: 2.576.

`eps`
: Positive numeric. Parameter for Bayesian Optimization. For reference see rBayesianOptimization documentation. Default: 0.

`kernel`
: List. Parameter for Bayesian Optimization. For reference see rBayesianOptimization documentation. Default: list(type = "exponential", power = 2).

`seed`
: Random seed. Default: 42.

Value

This function returns a list including:

- history: a table with the models from bayesian (n_sample + n_search) or random search (n_sample), their hyper-parameters and optimization metric, the weighted average rank
- models: a list with the details for each model in history
- best_model: results for the best selected model according to the weighted average rank, including:
  - predictions: min, max, q25, q50, q75, quantile at selected ci, mean, sd, skewness and kurtosis for each time feature
  - joint_error: max sequence error for the differentiated time features (max_rmse, max_mae, max_mdae, max_mape, max_mase, max_rae, max_rse, max_rrse, both for training and testing)
  - serie_errors: sequence error for the differentiated time features averaged across testing windows (rmse, mae, mdae, mape, mase, rae, rse, rrse, both for training and testing)
  - pred_stats: for each predicted time feature, IQR to range, divergence, risk ratio, upside probability, averaged across prediction time-points and at the terminal points

  - plots: a plot for each predicted time feature with highlighted median and confidence intervals
- time_log

Author(s)

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

See Also

Useful links:

- https://rpubs.com/giancarlo_vercellino/audrex

Examples

```
audrex(covid_in_europe[, 2:5], n_samp = 3, n_search = 2, seq_len = 10) ### BAYESIAN OPTIMIZATION
audrex(covid_in_europe[, 2:5], n_samp = 5, n_search = 0, seq_len = 10) ### RANDOM SEARCH
```

---

bitcoin_gold_oil *bitcoin_gold_oil data set*

Description

A data frame with different time series (prices and volumes) for bitcoin, gold and oil.

A data frame with different time series (prices and volumes) for bitcoin, gold and oil.

Usage

```
bitcoin_gold_oil
```

```
bitcoin_gold_oil
```

Format

A data frame with 18 columns and 1827 rows.

A data frame with 18 columns and 1827 rows.

Source

Yahoo Finance

Yahoo Finance

climate_anomalies *climate_anomalies data set*

Description

A data frame with different two time series on global mean temperature anomalies (GMTA) and global mean sea level (GMTA).

Usage

```
climate_anomalies
```

Format

A data frame with 2 columns and 266 rows.

Source

Datahub.io, Climate-change collection

covid_in_europe *covid_in_europe data set*

Description

A data frame with with daily and cumulative cases of Covid infections and deaths in Europe since March 2021.

A data frame with with daily and cumulative cases of Covid infections and deaths in Europe since March 2021.

Usage

```
covid_in_europe
```

```
covid_in_europe
```

Format

A data frame with 5 columns and 163 rows.

A data frame with 5 columns and 163 rows.

Source

www.ecdc.europa.eu

www.ecdc.europa.eu

engine *support functions for audrex*

Description

support functions for audrex

Usage

```
engine(
  predictors,
  target,
  booster,
  max_depth,
  eta,
  gamma,
  min_child_weight,
  subsample,
  colsample_bytree,
  lambda,
  alpha,
  n_windows,
  patience,
  nrounds
)
```

Arguments

predictors	A data frame with predictors on columns.
target	A numeric vector with target variable.
booster	String. Optimization methods available are: "gbtree", "gblinear". Default: "gbtree".
max_depth	Positive integer. Look to xgboost documentation for description. A vector with one or two positive integer for the search boundaries. The default value (NULL) sets automatically the values in c(1, 8).
eta	Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric between (0, 1] for the search boundaries. The default value (NULL) sets automatically the values in c(0, 1).
gamma	Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric for the search boundaries. The default value (NULL) sets automatically the values in c(0, 100).
min_child_weight	Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric for the search boundaries. The default value (NULL) sets automatically the values in c(0, 100).

`subsample` Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric between (0, 1] for the search boundaries. The default value (NULL) sets automatically the values in c(0, 1).

`colsample_bytree` Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric between (0, 1] for the search boundaries. The default value (NULL) sets automatically the values in c(0, 1).

`lambda` Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric for the search boundaries. The default value (NULL) sets automatically the values in c(0, 100).

`alpha` Positive numeric. Look to xgboost documentation for description. A vector with one or two positive numeric for the search boundaries. The default value (NULL) sets automatically the values in c(0, 100).

`n_windows` Positive integer. Number of (expanding) windows for cross-validation. Default: 3.

`patience` Positive numeric. Percentage of waiting rounds without improvement before xgboost stops. Default: 0.1

`nrounds` Positive numeric. Number of round for the extreme boosting machine. Look to xgboost for description. Default: 100.

Author(s)

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

Index