Package: naive (via r-universe)

August 25, 2024

Type Package

Title Empirical Extrapolation of Time Feature Patterns

Version 1.2.3

Description An application for the empirical extrapolation of time features selecting and summarizing the most relevant patterns in time sequences.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Depends R (>= 4.1)

Imports purrr (>= 1.0.1), ggplot2 (>= 3.4.2), readr (>= 2.1.4), lubridate (>= 1.9.2), imputeTS (>= 3.3), fANCOVA (>= 0.6-1), scales (>= 1.2.1), tictoc (>= 1.2), modeest (>= 2.4.0), moments (>= 0.14.1), greybox (>= 1.0.8), Rfast (>= 2.0.7), fastDummies (>= 1.6.3), entropy (>= 1.3.1), philentropy (>= 0.7.0)

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

NeedsCompilation no

Author Giancarlo Vercellino [aut, cre, cph]

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

Repository CRAN

Date/Publication 2023-06-20 14:30:04 UTC

Contents

naive *naive*

Description

Empirical Extrapolation of Time Feature Pattern

Usage

```
naive(
  df,
  seq_len = NULL,
  ci = 0.8,
  smoother = FALSE,
  cover = NULL,
  stride = NULL,
  method = NULL,
  location = NULL,
  n_windows = 10,
  n_samp = 30,
  dates = NULL,
  error_scale = "naive",
  error_benchmark = "naive",
  seed = 42
)
```

Arguments

df	A data frame with time features on columns (all numerics or all categories, but not both). In case of missing values, automatic missing imputation through kalman filter will be performed.
seq_len	Positive integer. Time-step number of the forecasting sequence. Default: NULL (random selection within boundaries).
ci	Confidence interval for prediction. Default: 0.8
smoother	Logical. Flag to TRUE for loess smoothing (only for numeric series). Default: FALSE.
cover	Positive numeric. The quantile cover around the location parameter (between 0 and 1). Default: NULL (random selection within boundaries).
stride	Positive integer. Shift between subsequent sequences. Default: NULL (random selection within boundaries).
method	String. Distance method using during the comparison of time sequences. Possible options are: "euclidean", "manhattan", "minkowski". Default: NULL (random selection).
location	String. Statistic used to center the cover parameter. Possible options are: "mean", "mode" (parzen method), "median". Default: NULL (random selection).

n_windows	Positive integer. Number of validation windows to test prediction error. Default: 10.
n_samp	Positive integer. Number of sample selected during random search. Default: 30.
dates	Date. Vector with dates for time features.
error_scale	String. Scale for the scaled error metrics. Two options: "naive" (average of naive one-step absolute error for the historical series) or "deviation" (standard error of the historical series). Default: "naive".
error_benchmark	String. Benchmark for the relative error metrics. Two options: "naive" (sequential extension of last value) or "average" (mean value of true sequence). Default: "naive".
seed	Positive integer. Random seed. Default: 42.

Value

This function returns a list including:

- exploration: collection of all the models explored with random search
- history: a table with the explored models' hyper-parameters and validation errors
- best_model: best combination resulting from the average prediction score across different ranks and features, including:
  - quant_preds: min, max, q25, q50, q75, quantiles at selected ci, mean, sd, mode, skewness, kurtosis, IQR to range, above to below median range, upside probability and divergence for each point fo predicted sequences
  - errors: testing errors for each time feature averaged across validation windows
  - plots: standard plot with confidence interval for each time feature
- time_log

Author(s)

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

Maintainer: Giancarlo Vercellino <giancarlo.vercellino@gmail.com> [copyright holder]

See Also

Useful links:

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

Examples

```
{
naive(time_features[, 2:3, drop = FALSE], seq_len = 30, n_samp = 1, n_windows = 5)
}
```

`time_features` *time features example: IBM, AAPL, AMZN, GOOGL and MSFT Close Prices*

Description

A data frame with with daily with daily prices for some Big Tech Companies since March 2017.

Usage

```
time_features
```

Format

A data frame with 6 columns and 1336 rows.

Source

finance.yahoo.com

Index