Package ‘lambdaTS’

July 22, 2025

Type Package

Title Variational Seq2Seq Model with Lambda Transformer for Time Series Analysis

Version 1.1

Author Giancarlo Vercellino

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

Description
Time series analysis based on lambda transformer and variational seq2seq, built on 'Torch'.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Depends R (>= 3.6)

Imports car, purrr, abind, ggplot2, readr, stringr, lubridate, narray, fANCOVA, imputeTS, modeest, scales, tictoc, bizdays, torch

NeedsCompilation no

Repository CRAN

Date/Publication 2022-02-20 20:00:02 UTC

Contents

bitcoin_gold_oil 2
lambdaTS 2

Index **6**

bitcoin_gold_oil *bitcoin_gold_oil data set*

Description

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

Usage

```
bitcoin_gold_oil
```

Format

A data frame with 18 columns and 1827 rows.

Source

Yahoo Finance

lambdaTS *lambdaTS: Variational Seq2Seq Lambda Transformer Model for Time Series Analysis*

Description

Time series analysis based on Lambda Transformer and Variational Seq2Seq, built on 'Torch'.

Usage

```
lambdaTS(
  data,
  target,
  future,
  past = future,
  ci = 0.8,
  deriv = 1,
  yjt = TRUE,
  shift = 0,
  smoother = FALSE,
  k_embed = 30,
  r_proj = ceiling(k_embed/3) + 1,
  n_heads = 1,
  n_bases = 1,
  activ = "linear",
  loss_metric = "elbo",
  optim = "adam",
```

```
    epochs = 30,
    lr = 0.01,
    patience = epochs,
    verbose = TRUE,
    sample_n = 100,
    seed = 42,
    dev = "cpu",
    starting_date = NULL,
    dbreak = NULL,
    days_off = NULL,
    min_set = future,
    holdout = 0.5,
    batch_size = 30
)
```

Arguments

data	A data frame with ts on columns and possibly a date column (not mandatory)
target	String. Time series names to be jointly analyzed within the seq2seq model
future	Positive integer. The future dimension with number of time-steps to be predicted
past	Positive integer. The past dimension with number of time-steps in the past used for the prediction. Default: future
ci	Confidence interval. Default: 0.8
deriv	Positive integer. Number of differentiation operations to perform on the original series. 0 = no change; 1: one diff; 2: two diff, and so on.
yjt	Logical. Performing Yeo-Johnson Transformation on data is always advisable, especially when dealing with different ts at different scales. Default: TRUE
shift	Vector of positive integers. Allow for target variables to shift ahead of time. Zero means no shift. Length must be equal to the number of targets. Default: 0.
smoother	Logical. Perform optimal smooting using standard loess. Default: FALSE
k_embed	Positive integer. Number of Time2Vec embedding dimensions. Minimum value is 2. Default: 30
r_proj	Positive integer. Number of dimensions for the reduction space (to reduce quadratic complexity). Must be largely less than k_embed size. Default: ceiling(k_embed/3) + 1
n_heads	Positive integer. Number of heads for the attention mechanism. Computationally expensive, use with care. Default: 1
n_bases	Positive integer. Number of normal curves to build on each parameter. Computationally expensive, use with care. Default: 1
activ	String. The activation function for the linear transformation of the attention matrix into the future sequence. Implemented options are: "linear", "leaky_relu", "celu", "elu", "gelu", "selu", "softplus", "bent", "snake", "softmax", "softmin", "softsign", "sigmoid", "tanh", "tanhshrink", "swish", "hardtanh", "mish". Default: "linear".

`loss_metric`	String. Loss function for the variational model. Two options: "elbo" or "crps". Default: "crps".
`optim`	String. Optimization methods available are: "adadelta", "adagrad", "rmsprop", "rprop", "sgd", "asgd", "adam". Default: "adam".
`epochs`	Positive integer. Default: 30.
`lr`	Positive numeric. Learning rate. Default: 0.01.
`patience`	Positive integer. Waiting time (in epochs) before evaluating the overfit performance. Default: epochs.
`verbose`	Logical. Default: TRUE
`sample_n`	Positive integer. Number of samples from the variational model to evalute the mean forecast values. Computationally expensive, use with care. Default: 100.
`seed`	Random seed. Default: 42.
`dev`	String. Torch implementation of computational platform: "cpu" or "cuda" (gpu). Default: "cpu".
`starting_date`	Date. Initial date to assign temporal values to the series. Default: NULL (progressive numbers).
`dbreak`	String. Minimum time marker for x-axis, in liberal form: i.e., "3 months", "1 week", "20 days". Default: NULL.
`days_off`	String. Weekdays to exclude (i.e., c("saturday", "sunday")). Default: NULL.
`min_set`	Positive integer. Minimun number for validation set in case of automatic resize of past dimension. Default: future.
`holdout`	Positive numeric. Percentage of time series for holdout validation. Default: 0.5.
`batch_size`	Positive integer. Default: 30.

Value

This function returns a list including:

- prediction: a table with quantile predictions, mean and std for each ts
- history: plot of loss during the training process for the joint-transformed ts
- plot: graph with history and prediction for each ts
- learning_error: errors for the joint-ts in the transformed scale (rmse, mae, mdae, mpe, mape, smape, rrse, rae)
- feature_errors: errors for each ts in the original scale (rmse, mae, mdae, mpe, mape, smape, rrse, rae)
- pred_stats: for each predicted time feature, IQR to range, KL-divergence, risk ratio, upside probability, averaged across time-points and compared at the terminal points.
- time_log

Author(s)

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

Examples

```
## Not run:
lambdaTS(bitcoin_gold_oil, c("gold_close", "oil_Close"), 30, deriv = 1)

## End(Not run)
```

Index

∗ **datasets**
    bitcoin_gold_oil, 2

bitcoin_gold_oil, 2

lambdaTS, 2
lambdaTS-package (lambdaTS), 2