Package ‘proteus’

March 8, 2023

Type Package
Title Multiform Seq2Seq Model for Time-Feature Analysis
Version 1.1.0
Author Giancarlo Vercellino
Maintainer Giancarlo Vercellino <giancarlo.vercellino@gmail.com>
Description Seq2seq time-feature analysis based on variational model, with a wide range of distributions available for the latent variable.
License GPL-3
Encoding UTF-8
LazyData true
RoxygenNote 7.2.1
Depends R (>= 3.6)
Imports purrr (>= 0.3.4), abind (>= 1.4-5), ggplot2 (>= 3.3.3),
ggthemes (>= 0.2.4), readr (>= 1.4.0), stringr (>= 1.4.0),
lubridate (>= 1.7.9.2), narray (>= 0.4.1), fANCOVA (>= 0.6-1),
imputeTS (>= 3.1), modeest (>= 2.4.0), scales (>= 1.1.1),
tictoc (>= 1.0.1), torch (>= 0.3.0), actuar (>= 3.1-1), VGAM
(>= 1.1-5), moments (>= 0.14), dplyr (>= 1.0.2), greybox (>=
1.0.7)
URL https://rpubs.com/giancarlo_vercellino/proteus
Suggests testthat (>= 3.0.0)
Config/testthat/edition 3
NeedsCompilation no
Repository CRAN
Date/Publication 2023-03-08 09:20:05 UTC

R topics documented:

amzn_aapl_fb .................................................. 2
proteus .......................................................... 2
proteus_random_search ....................................... 5
amzn_aapl_fb data set

Description
A data frame with the close prices for Amazon, Google and Facebook.

Usage
amzn_aapl_fb

Format
A data frame with 4 columns and 1798 rows.

Source
Yahoo Finance

proteus

Description
Proteus is a Sequence-to-Sequence Variational Model designed for time-feature analysis, leveraging a wide range of distributions for improved accuracy. Unlike traditional methods that rely solely on the normal distribution, Proteus uses various latent models to better capture and predict complex processes. To achieve this, Proteus employs a neural network architecture that estimates the shape, location, and scale parameters of the chosen distribution. This approach transforms past sequence data into future sequence parameters, improving the model’s prediction capabilities. Proteus also assesses the accuracy of its predictions by estimating the error of measurement and calculating the confidence interval. By utilizing a range of distributions and advanced modeling techniques, Proteus provides a more accurate and comprehensive approach to time-feature analysis.

Usage
proteus(
    data,
    target,
    future,
    past,
    ci = 0.8,
    smoother = FALSE,
    t_embed = 30,
    activ = "linear",
)
```
proteus

nodes = 32, 
distr = "normal", 
optim = "adam", 
loss_metric = "crps", 
epochs = 30, 
lr = 0.01, 
patience = 10, 
latent_sample = 100, 
verbose = TRUE, 
stride = 1, 
dates = NULL, 
rolling_blocks = FALSE, 
n_blocks = 4, 
block_minset = 30, 
error_scale = "naive", 
error_benchmark = "naive", 
batch_size = 30, 
omit = FALSE, 
seed = 42 )
```

**Arguments**

- **data**
  A data frame with time features on columns and possibly a date column (not mandatory)

- **target**
  Vector of strings. Names of the time features to be jointly analyzed

- **future**
  Positive integer. The future dimension with number of time-steps to be predicted

- **past**
  Positive integer. Length of past sequences

- **ci**
  Positive numeric. Confidence interval. Default: 0.8

- **smoother**
  Logical. Perform optimal smoothing using standard loess for each time feature. Default: FALSE

- **t_embed**
  Positive integer. Number of embedding for the temporal dimension. Minimum value is equal to 2. Default: 30.

- **activ**
  String. Activation function to be used by the forward network. Implemented functions are: "linear", "mish", "swish", "leaky_relu", "celu", "elu", "gelu", "selu", "bent", "softmax", "softmin", "softsign", "softplus", "sigmoid", "tanh". Default: "linear".

- **nodes**

- **distr**
  String. Distribution to be used by variational model. Implemented distributions are: "normal", "genbeta", "gav", "gpd", "genray", "cauchy", "exp", "logis", "chisq", "gumbl", "laplace", "lognorm". Default: "normal".

- **optim**
  String. Optimization method. Implemented methods are: "adadelta", "adagrad", "rmsprop", "rprop", "sgd", "asgd", "adam".

- **loss_metric**
  String. Loss function for the variational model. Three options: "elbo", "crps", "score". Default: "crps".

lr Positive numeric. Learning rate. Default: 0.01.


latent_sample Positive integer. Number of samples to draw from the latent variables. Default: 100.

verbose Logical. Default: TRUE


dates String. Label of feature where dates are located. Default: NULL (progressive numbering).

rolling_blocks Logical. Option for incremental or rolling window. Default: FALSE.


block_minset Positive integer. Minimum number of sequence to create a block. Default: 3.

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".

batch_size Positive integer. Default: 30.

omit Logical. Flag to TRUE to remove missing values, otherwise all gaps, both in dates and values, will be filled with kalman filter. Default: FALSE.

seed Random seed. Default: 42.

**Value**

This function returns a list including:

- `model_descr`: brief model description (number of tensors and parameters)
- `prediction`: a table with quantile predictions, mean, std, mode, skewness and kurtosis for each time feature (and other metrics, such as iqr_to_range, above_to_below_range, upside_prob, divergence).
- `plot`: graph with history and prediction for each time feature
- `feature_errors`: train and test error for each time feature (me, mae, mse, rmsse, mpe, mape, rmae, rrmse, rame, mase, smse, sce)
- `history`: average cross-validation loss across blocks
- `time_log`: computation time.

**Author(s)**

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

References

https://rpubs.com/giancarlo_vercellino/proteus

Description

proteus_random_search is a function for fine-tuning using random search on the hyper-parameter space of proteus (predefined or custom).

Usage

proteus_random_search(
  n_samp,
  data,
  target,
  future,
  past = NULL,
  ci = 0.8,
  smoother = FALSE,
  t_embed = NULL,
  activ = NULL,
  nodes = NULL,
  distr = NULL,
  optim = NULL,
  loss_metric = "crps",
  epochs = 30,
  lr = NULL,
  patience = 10,
  latent_sample = 100,
  verbose = TRUE,
  stride = NULL,
  dates = NULL,
  rolling_blocks = FALSE,
  n_blocks = 4,
  block_minset = 10,
  error_scale = "naive",
  error_benchmark = "naive",
  batch_size = 30,
  seed = 42,
  omit = FALSE,
  keep = FALSE
)
Arguments

n_samp: Positive integer. Number of models to be randomly generated sampling the hyper-parameter space.
data: A data frame with time features on columns and possibly a date column (not mandatory).
target: Vector of strings. Names of the time features to be jointly analyzed.
future: Positive integer. The future dimension with number of time-steps to be predicted.
smoother: Logical. Perform optimal smoothing using standard loess for each time feature. Default: FALSE.
t_embed: Positive integer. Number of embedding for the temporal dimension. Minimum value is equal to 2. Default: NULL (search range 2:30).
 loss_metric: String. Loss function for the variational model. Three options: "elbo", "crps", "score". Default: "crps".
 lr: Positive numeric. Learning rate. Default: NULL (search range 0.001:0.1).
 latent_sample: Positive integer. Number of samples to draw from the latent variables. Default: 100.
 verbose: Logical. Default: TRUE
 stride: Positive integer. Number of shifting positions for sequence generation. Default: NULL (search range 1:10).
 dates: String. Label of feature where dates are located. Default: NULL (progressive numbering).
 rolling_blocks: Logical. Option for incremental or rolling window. Default: FALSE.
proteus_random_search

- **block_minset**: Positive integer. Minimum number of sequence to create a block. Default: 3.
- **error_scale**: String. Scale for the scaled error metrics (for continuous variables). 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 (for continuous variables). Two options: "naive" (sequential extension of last value) or "average" (mean value of true sequence). Default: "naive".
- **batch_size**: Positive integer. Default: 30.
- **seed**: Random seed. Default: 42.
- **omit**: Logical. Flag to TRUE to remove missing values, otherwise all gaps, both in dates and values, will be filled with kalman filter. Default: FALSE.
- **keep**: Logical. Flag to TRUE to keep all the explored models. Default: FALSE.

**Value**

This function returns a list including:

- random_search: summary of the sampled hyper-parameters and average error metrics.
- best: best model according to overall ranking on all average error metrics (for negative metrics, absolute value is considered).
- all_models: list with all the model generated (if keep flagged to TRUE).
- time_log: computation time.

**Author(s)**

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

**References**

https://rpubs.com/giancarlo_vercellino/proteus
Index

* datasets
  amzn_aapl_fb, 2
amzn_aapl_fb, 2
proteus, 2
proteus_random_search, 5