

# Package ‘segen’

October 14, 2022

**Type** Package

**Title** Sequence Generalization Through Similarity Network

**Version** 1.1.0

**Author** Giancarlo Vercellino

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

**Description** Proposes an application for sequence prediction generalizing the similarity within the network of previous sequences.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**Depends** R (>= 3.6)

**Imports** purrr (>= 0.3.4), ggplot2 (>= 3.3.5), readr (>= 2.1.2),  
lubridate (>= 1.7.10), imputeTS (>= 3.2), fANCOVA (>= 0.6-1),  
scales (>= 1.1.1), tictoc (>= 1.0.1), modeest (>= 2.4.0),  
moments (>= 0.14), greybox (>= 1.0.1), philentropy (>= 0.5.0),  
entropy (>= 1.3.1), Rfast (>= 2.0.6), narray (>= 0.4.1.1),  
fastDummies (>= 1.6.3)

**URL** [https://rpubs.com/giancarlo\\_vercellino/segen](https://rpubs.com/giancarlo_vercellino/segen)

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2022-08-15 19:30:02 UTC

## R topics documented:

segen	2
time_features	4
<b>Index</b>	<b>5</b>

---

 segen

*segen*


---

## Description

Sequence Generalization Through Similarity Network

## Usage

```
segen(
  df,
  seq_len = NULL,
  similarity = NULL,
  dist_method = NULL,
  rescale = NULL,
  smoother = FALSE,
  ci = 0.8,
  error_scale = "naive",
  error_benchmark = "naive",
  n_windows = 10,
  n_samp = 30,
  dates = NULL,
  seed = 42
)
```

## Arguments

<code>df</code>	A data frame with time features on columns. They could be numeric variables or categorical, but not both.
<code>seq_len</code>	Positive integer. Time-step number of the forecasting sequence. Default: NULL (automatic selection between 2 and max limit).
<code>similarity</code>	Positive numeric. Degree of similarity between two sequences, based on quantile conversion of distance. Default: NULL (automatic selection between 0.01, maximal difference, and 0.99, minimal difference).
<code>dist_method</code>	String. Method for calculating distance among sequences. Available options are: "euclidean", "manhattan", "maximum", "minkowski". Default: NULL (random search).
<code>rescale</code>	Logical. Flag to TRUE for min-max scaling of distances. Default: NULL (random search).
<code>smoother</code>	Logical. Flag to TRUE for loess smoothing. Default: FALSE.
<code>ci</code>	Confidence interval for prediction. Default: 0.8
<code>error_scale</code>	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".
n_windows	Positive integer. Number of validation windows to test prediction error. Default: 10.
n_samp	Positive integer. Number of samples for random search. Default: 30.
dates	Date. Vector with dates for time features.
seed	Positive integer. Random seed. Default: 42.

### Value

This function returns a list including:

- exploration: list of all not-null models, complete with predictions and error metrics
- history: a table with the sampled models, hyper-parameters, validation errors
- best\_model: results for the best selected model according to the weighted average rank, including:
  - predictions: for continuous variables, min, max, q25, q50, q75, quantiles at selected ci, mean, sd, mode, skewness, kurtosis, IQR to range, risk ratio, upside probability and divergence for each point fo predicted sequences; for factor variables, min, max, q25, q50, q75, quantiles at selected ci, proportions, difformity (deviation of proportions normalized over the maximum possible deviation), entropy, upgrade probability and divergence for each point fo predicted sequences
  - testing\_errors: testing errors for each time feature for the best selected model (for continuous variables: me, mae, mse, rmsse, mpe, mape, rmae, rrmse, rame, mase, smse, sce, gmrae; for factor variables: czekanowski, tanimoto, cosine, hassebrook, jaccard, dice, canberra, gower, lorentzian, clark)
  - plots: standard plots with confidence interval for each time feature
- time\_log

### Author(s)

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

### See Also

Useful links:

- [https://rpubs.com/giancarlo\\_vercellino/segen](https://rpubs.com/giancarlo_vercellino/segen)

### Examples

```
segen(time_features[, 1, drop = FALSE], seq_len = 30, similarity = 0.7, n_windows = 3, n_samp = 1)
```

---

`time_features`*time features example: IBM and Microsoft Close Prices*

---

**Description**

A data frame with with daily with daily prices for IBM and Microsoft since April 2020

**Usage**

```
time_features
```

**Format**

A data frame with 2 columns and 1324 rows.

**Source**

[finance.yahoo.com](https://finance.yahoo.com)

# Index

\* **datasets**

time\_features, [4](#)

segen, [2](#)

segen-package (segen), [2](#)

time\_features, [4](#)