Package 'echos'

June 23, 2025

Type Package

Title Echo State Networks for Time Series Modeling and Forecasting

Version 1.0.2

Description Provides a lightweight implementation of functions and methods for fast and fully automatic time series modeling and forecasting using Echo State Networks (ESNs).

License GPL-3

URL https://github.com/ahaeusser/echos, https://ahaeusser.github.io/echos/

BugReports https://github.com/ahaeusser/echos/issues

Depends R (>= 4.0.0), fabletools (>= 0.3.0)

Imports Rcpp (>= 1.0.3), RcppArmadillo, tsibble, dplyr, tidyr, rlang, distributional

LinkingTo Rcpp, RcppArmadillo

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Suggests knitr, rmarkdown, tidyverse, fable, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation yes

Author Alexander Häußer [aut, cre, cph] (ORCID: <https://orcid.org/0009-0000-5419-8479>)

Maintainer Alexander Häußer <alexander-haeusser@gmx.de>

Repository CRAN

Date/Publication 2025-06-23 10:20:10 UTC

Contents

`ESN` *Train an Echo State Network*

Description

Train an Echo State Network (ESN) to a univariate time series. The function automatically manages data pre-processing, reservoir generation (i.e., internal states) and model estimation and selection. The function is a wrapper for `train_esn()` and intended to be used in combination with `fabletools::model()`.

Usage

```
ESN(formula, ...)
```

Arguments

`formula`	Model specification (currently not in use).
`...`	Further arguments passed to `train_esn()`.

Value

An object of class `ESN`.

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value))
```

`filter_esn` *Filter ESN models*

Description

Filter an object of class `mdl_df` ("mable") to include ESN models only, i.e., other models like ARIMA or ETS are excluded from the mable.

Usage

```
filter_esn(object)
```

Arguments

`object`	An object of class `mdl_df`, containing an ESN model.

Value

An object of class `mdl_df` in long-format.

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value)) %>%
filter_esn()
```

fitted.ESN *Extract fitted values from a trained ESN*

Description

Extract fitted values from a trained ESN as `tsibble`.

Usage

```
## S3 method for class 'ESN'
fitted(object, ...)
```

Arguments

`object`	An object of class `mdl_df`, containing an ESN model.
`...`	Currently not in use.

Value

Fitted values extracted from the object.

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value)) %>%
fitted()
```

forecast.ESN *Forecast an Echo State Network*

Description

Forecast an Echo State Network (ESN) from a trained model via recursive forecasting. Forecast intervals are generated by simulating future sample path based on a moving block bootstrap of the residuals and estimating the quantiles from the simulations. The function is a wrapper for `forecast_esn()` and intended to be used in combination with `fabletools::model()`.

Usage

```
## S3 method for class 'ESN'
forecast(
  object,
  new_data,
  normal = TRUE,
  n_sim = 200,
  specials = NULL,
  xreg = NULL,
  ...
)
```

Arguments

object	An object of class mdl_df, containing an ESN model.
new_data	Forecast horizon (n-step ahead forecast).
normal	Logical value. If TRUE, dist_normal() is used, otherwise dist_sample().
n_sim	Integer value. The number of future sample path generated during simulation.
specials	Currently not in use.
xreg	A tsibble containing exogenous variables.
...	Currently not in use.

Value

An object of class fbl_ts ("fable").

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value)) %>%
forecast(h = 18)
```

---

forecast_esn *Forecast an Echo State Network*

Description

Forecast an Echo State Network (ESN) from a trained model via recursive forecasting. Forecast intervals are generated by simulating future sample path based on a moving block bootstrap of the residuals and estimating the quantiles from the simulations.

Usage

```
forecast_esn(
  object,
  n_ahead = 18,
  levels = c(80, 95),
  n_sim = 100,
  n_seed = 42
)
```

Arguments

object	An object of class esn. The result of a call to train_esn().
n_ahead	Integer value. The number of periods for forecasting (i.e. forecast horizon).
levels	Integer vector. The levels of the forecast intervals, e.g., 80% and 95%.
n_sim	Integer value. The number of future sample path generated during simulation.
n_seed	Integer value. The seed for the random number generator (for reproducibility).

Value

A list containing:

- point: Numeric vector containing the point forecasts.
- interval: Numeric matrix containing the forecast intervals.
- sim: Numeric matrix containing the simulated future sample path.
- levels: Integer vector. The levels of the forecast intervals.
- actual: Numeric vector containing the actual values.
- fitted: Numeric vector containing the fitted values.
- n_ahead: Integer value. The number of periods for forecasting (forecast horizon).
- model_spec: Character value. The model specification as string.

Examples

```
xdata <- as.numeric(AirPassengers)
xmodel <- train_esn(y = xdata)
xfcst <- forecast_esn(xmodel, n_ahead = 12)
plot(xfcst)
```

glance.ESN *Summary of trained models during random search*

Description

Return summary statistics from trained ESN models during random search as `tibble`.

- `model`: Model identifier.
- `loglik`: Log-likelihood.
- `nobs`: Number of observations.
- `df`: Effective degrees of freedom.
- `lambda`: Regularization parameter.
- `aic`: Akaike Information Criterion.
- `aicc`: Corrected Akaike Information Criterion.
- `bic`: Bayesian Information Criterion.
- `hqc`: Hannan-Quinn Information Criterion.
- `mse`: Mean Squared Error.
- `mae`: Mean Absolute Error.

Usage

```
## S3 method for class 'ESN'
glance(x, ...)
```

Arguments

`x`	An object of class `mdl_df`, containing an ESN model.
`...`	Currently not in use.

Value

Summary statistics extracted from the object.

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value)) %>%
glance()
```

is.esn *Checks if object is of class "esn"*

Description

Returns TRUE if the object is of class "esn".

Usage

```
is.esn(object)
```

Arguments

object	object to be tested.

Value

Logical value. If TRUE, the object is of class "esn".

Examples

```
xdata <- as.numeric(AirPassengers)
xmodel <- train_esn(y = xdata)
is.esn(xmodel)
```

is.forecast_esn *Checks if object is of class "forecast_esn"*

Description

Returns TRUE if the object is of class "forecast_esn".

Usage

```
is.forecast_esn(object)
```

Arguments

object	object to be tested.

Value

Logical value. If TRUE, the object is of class "forecast_esn".

Examples

```
xdata <- as.numeric(AirPassengers)
xmodel <- train_esn(y = xdata)
xfcst <- forecast_esn(xmodel, n_ahead = 12)
is.forecast_esn(xfcst)
```

`m4_data` *M4 dataset*

Description

`tsibble` with six monthly time series from the M4 Forecasting Competition. The datasets contains the following time series:

- M21655 (Demographic), 1995 Jan - 2015 Mar
- M21683 (Demographic), 2000 Jan - 2023 Apr
- M2717 (Macro), 1996 Jan - 2016 Nov
- M28597 (Industry), 1996 Jan - 2016 Dec
- M42529 (Finance), 2001 Jan - 2009 Apr
- M4813 (Macro), 1994 Apr - 2006 May

Usage

```
data(m4_data)
```

Format

A time series object of class `tsibble` with 1.152 rows and 4 columns:

- `series`: Unique identifier as `character` (key variable).
- `category`: Category (e.g., Demographic, Macro) as `factor`.
- `index`: Date as `yearmonth` (index variable).
- `value`: Value as `numeric` (measurement variable).

Source

M4 Forecasting Competition

Examples

```
data(m4_data)
```

model_sum.ESN *Model specification of a trained ESN model*

Description

Provides a compact overview of the model specification in the format `ESN({n_states, alpha, rho}, {n_models, df})`.

Usage

```
## S3 method for class 'ESN'
model_sum(x)
```

Arguments

x	An object of class `mdl_df`, containing an ESN model.

Value

Model summary extracted from the object.

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value))
```

plot.esn *Plot internal states of a trained ESN model*

Description

Plot internal states (i.e., the reservoir) of a trained ESN model as line chart.

Usage

```
## S3 method for class 'esn'
plot(x, ...)
```

Arguments

x	An object of class `esn`. The result of a call to `train_esn()`.
...	Further arguments passed to `matplot()`.

Value

Line chart of internal states.

Examples

```
xdata <- as.numeric(AirPassengers)
xmodel <- train_esn(y = xdata)
plot(xmodel)
```

`plot.forecast_esn` *Plot forecasts of a trained ESN model*

Description

Plot point forecasts and forecast intervals, actual values of a trained ESN model. Optionally, test data (out-of-sample) and fitted values can be added to the plot.

Usage

```
## S3 method for class 'forecast_esn'
plot(x, test = NULL, fitted = TRUE, interval = TRUE, n_obs = NULL, ...)
```

Arguments

`x`	An object of class `forecast_esn`. The result of a call to `forecast_esn()`.
`test`	Numeric vector. Test data, i.e., out-of-sample actual values.
`fitted`	Logical value. If `TRUE`, fitted values are added.
`interval`	Logical value. If `TRUE`, forecast intervals are added.
`n_obs`	Integer value. If `NULL`, all in-sample values are shown, otherwise only the last `n_obs`.
`...`	Currently not in use.

Value

Line chart of point forecast and actual values.

Examples

```
xdata <- as.numeric(AirPassengers)
xmodel <- train_esn(y = xdata)
xfcst <- forecast_esn(xmodel, n_ahead = 12)
plot(xfcst)
```

print.esn *Print model specification of the trained ESN model*

Description

Provides a compact overview of the model specification in the format `ESN({n_states, alpha, rho}, {n_models, df})`.

Usage

```
## S3 method for class 'esn'
print(x, ...)
```

Arguments

`x`	An object of class `esn`. The result of a call to `train_esn()`.
`...`	Currently not in use.

Value

Print specification of the trained ESN model.

Examples

```
xdata <- as.numeric(AirPassengers)
xmodel <- train_esn(y = xdata)
print(xmodel)
```

report.ESN *Provide a detailed summary of the trained ESN model*

Description

Provide a detailed summary of the trained ESN model. The function is a wrapper for `summary.esn()`.

Usage

```
## S3 method for class 'ESN'
report(object, ...)
```

Arguments

`object`	An object of class `mdl_df`, containing an ESN model.
`...`	Currently not in use.

Value

Print detailed model summary.

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value)) %>%
report()
```

`reservoir` *Return the reservoir from a trained ESN as tibble*

Description

Return the reservoir (internal states) from a trained ESN as tibble. The function works only for models of class `ESN`.

Usage

```
reservoir(object)
```

Arguments

`object`	An object of class `mdl_df`, containing an ESN model.

Value

A tibble containing the reservoir (internal states).

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value)) %>%
reservoir()
```

residuals.ESN *Extract residuals from a trained ESN*

Description

Extract residuals from a trained ESN as `tsibble`.

Usage

```
## S3 method for class 'ESN'
residuals(object, ...)
```

Arguments

`object`	An object of class `mdl_df`, containing an ESN model.
`...`	Currently not in use.

Value

Residuals extracted from the object.

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value)) %>%
residuals()
```

run_reservoir *Run reservoir*

Description

Run reservoir creates the internal states for the ESN.

Arguments

`input`	Numeric matrix containing the input features
`win`	Numeric matrix. The input weight matrix.
`wres`	Numeric matrix. The reservoir weight matrix.
`alpha`	Numeric value. The leakage rate (smoothing parameter).

Value

states train Numeric matrix with the internal states.

summary.esn *Provide a detailed summary of the trained ESN model*

Description

Provide a detailed summary of the trained ESN model.

Usage

```
## S3 method for class 'esn'
summary(object, ...)
```

Arguments

object	An object of class esn. The result of a call to train_esn().
...	Currently not in use.

Value

Print detailed model summary.

Examples

```
xdata <- as.numeric(AirPassengers)
xmodel <- train_esn(y = xdata)
summary(xmodel)
```

synthetic_data *Synthetic data*

Description

tibble with ten synthetic time series. The dataset contains the following time series:

- Square Wave
- Sawtooth Wave
- Harmonic Wave
- Harmonic Wave w/ Trend
- Amplitude Modulated Wave
- Frequency Modulated Wave
- AR(1) Process
- MA(2) Process
- White Noise Process
- Random Walk Process

Usage

```
data(synthetic_data)
```

Format

An object of class `tibble` with 2.000 rows and 3 columns:

- `variable`: Unique identifier as `character` (key variable).
- `index`: Index as `integer` (index variable).
- `value`: Value as `numeric` (measurement variable).

Examples

```
data(synthetic_data)
```

---

tidy.ESN *Estimated coefficients*

Description

Return the estimated coefficients from a trained ESN as `tibble`.

Usage

```
## S3 method for class 'ESN'
tidy(x, ...)
```

Arguments

`x`	An object of class `mdl_df`, containing an ESN model.
`...`	Currently not in use.

Value

Coefficients extracted from the object.

Examples

```
library(tsibble)
library(fable)
AirPassengers %>%
as_tsibble() %>%
model("ESN" = ESN(value)) %>%
tidy()
```

`train_esn` *Train an Echo State Network*

Description

Train an Echo State Network (ESN) to a univariate time series. The function automatically manages data pre-processing, reservoir generation (i.e., internal states) and model estimation and selection.

Usage

```
train_esn(
  y,
  lags = 1,
  inf_crit = "bic",
  n_diff = NULL,
  n_states = NULL,
  n_models = NULL,
  n_initial = NULL,
  n_seed = 42,
  alpha = 1,
  rho = 1,
  density = 0.5,
  lambda = c(1e-04, 2),
  scale_win = 0.5,
  scale_wres = 0.5,
  scale_inputs = c(-0.5, 0.5)
)
```

Arguments

`y`	Numeric vector containing the response variable.
`lags`	Integer vector with the lag(s) associated with the input variable.
`inf_crit`	Character value. The information criterion used for variable selection `inf_crit = c("aic", "aicc", "bic", "hqc")`.
`n_diff`	Integer vector. The nth-differences of the response variable.
`n_states`	Integer value. The number of internal states per reservoir.
`n_models`	Integer value. The maximum number of (random) models to train for model selection.
`n_initial`	Integer value. The number of observations of internal states for initial drop out (throw-off).
`n_seed`	Integer value. The seed for the random number generator (for reproducibility).
`alpha`	Numeric value. The leakage rate (smoothing parameter) applied to the reservoir.
`rho`	Numeric value. The spectral radius for scaling the reservoir weight matrix.
`density`	Numeric value. The connectivity of the reservoir weight matrix (dense or sparse).

lambda	Numeric vector. Lower and upper bound of lambda sequence for ridge regression.
scale_win	Numeric value. The lower and upper bound of the uniform distribution for scaling the input weight matrix.
scale_wres	Numeric value. The lower and upper bound of the uniform distribution for scaling the reservoir weight matrix.
scale_inputs	Numeric vector. The lower and upper bound for scaling the time series data.

Value

A list containing:

- actual: Numeric vector containing the actual values.
- fitted: Numeric vector containing the fitted values.
- resid: Numeric vector containing the residuals.
- states_train: Numeric matrix containing the internal states.
- method: A list containing several objects and meta information of the trained ESN (weight matrices, hyperparameters, model metrics, etc.).

Examples

```
xdata <- as.numeric(AirPassengers)
xmodel <- train_esn(y = xdata)
summary(xmodel)
```

Index