

Package ‘htsDegenerateR’

May 8, 2026

Type Package

Title Degenerate Hierarchical Time Series Reconciliation

Version 0.1.0

Description Takes the MinT implementation of the 'hts' <<https://cran.r-project.org/package=hts>> package and adapts it to allow degenerate hierarchical structures. Instead of the ``nodes" argument, this function takes an S matrix which is more versatile in the structures it allows. For a demo, see Steinmeister and Pauly (2024) <[doi:10.15488/17729](https://doi.org/10.15488/17729)>. The MinT algorithm is based on Wickramasuriya et al. (2019) <[doi:10.1080/01621459.2018.1448825](https://doi.org/10.1080/01621459.2018.1448825)>.

License GPL (>= 2)

Encoding UTF-8

Imports SparseM, methods, stats

Suggests forecast, testthat, knitr, rmarkdown, covr

RoxygenNote 7.3.1

NeedsCompilation no

Author Louis Steinmeister [aut, cre],
Markus Pauly [aut]

Maintainer Louis Steinmeister <louis.steinmeister@udo.edu>

Repository CRAN

Date/Publication 2024-12-10 21:10:02 UTC

Contents

accuracy.gts	2
BU	2
MinT	3
strucScaling	4

Index	6
--------------	----------

accuracy.gts

accuracy.gts

Description

accuracy.gts

Usage

accuracy.gts(fcasts, actuals)

Arguments

fcasts forecasts to be evaluated

actuals actuals to compare the forecasts against

Value

Averaged error measures across all time series in matrix form.

BU

Bottom-up reconciliation

Description

Bottom-up reconciliation

Usage

BU(fcasts, S)

Arguments

fcasts forecasts to be reconciled

S structure matrix representing the hierarchical structure of the hts

Value

reconciled forecasts

MinT	<i>Using the method of Wickramasuriya et al. (2019), this function (based on Hyndman et al.'s hts library) combines the forecasts at all levels of a hierarchical time series and works for degenerate hierarchies.</i>
------	---

Description

Using the method of Wickramasuriya et al. (2019), this function (based on Hyndman et al.'s hts library) combines the forecasts at all levels of a hierarchical time series and works for degenerate hierarchies.

Usage

```
MinT(
  fcasts,
  Smat,
  residual,
  covariance = c("shr", "sam", "custom"),
  nonnegative = FALSE,
  cov.type = "complete.obs",
  cov.matrix = NULL
)
```

Arguments

<code>fcasts</code>	a vector or a matrix (rows = horizon, columns = ts columns) of forecasts
<code>Smat</code>	a structure matrix detailing the hierarchical structure of the hts. Make sure that the order of the rows align with the order of the forecasts.
<code>residual</code>	a matrix of in-sample residuals (columns = ts columns)
<code>covariance</code>	should a shrinkage estimator or the sample estimator be used? alternatively, a custom covariance matrix can be passed (additionally requires the <code>cov.matrix</code> argument)
<code>nonnegative</code>	not implemented yet.
<code>cov.type</code>	specify how the covariance matrix should be computed (default = complete observations). Note that <code>pairwise.complete.obs</code> may not yield a positive definite matrix!
<code>cov.matrix</code>	specify in case a custom covariance matrix should be used

Value

reconciled forecasts

References

[hts: Hierarchical and Grouped Time Series]<<https://cran.r-project.org/package=hts>> [Optimal Forecast Reconciliation for Hierarchical and Grouped Time Series Through Trace Minimization]<[doi:10.1080/01621459.2018.1444444](https://doi.org/10.1080/01621459.2018.1444444)>
 [Degenerate Hierarchical Time Series Reconciliation With The Minimum Trace Algorithm in R]<[doi:10.15488/17729](https://doi.org/10.15488/17729)>

Examples

```

# Set the seed for reproducibility
set.seed(123)

# Create a sequence of 120 numbers
x <- seq(1, 120)

# Generate the columns
AA <- sin(x*pi/6) + rnorm(120, 0, 1) # Sine component with random noise
AB <- 0.05*x + rnorm(120, 0, 0.5) # Linear component
B <- cos(x*pi/6) + rnorm(120, 0, 1) # Cosine component

# Combine the columns into a matrix
matrix <- cbind(AA, AB, B)
hts = ts(matrix, frequency = 12)

# Define S matrix
S <- rbind(c(1,1,1), c(1,1,0), diag(1,3))
rownames(S) <- c("Total", "A", "AA", "AB", "B")
colnames(S) <- c("AA", "AB", "B")

# Aggregate hts on all levels
hts.complete <- ts(t(S %*% t(hts)), frequency = 12)

# Fit a model to the time series
hts.models = lapply(hts.complete, function(c.ts) forecast::ets(c.ts))

# Fit a model to the time series
hts.models = lapply(hts.complete, function(c.ts) forecast::ets(c.ts))
# Generate predictions based on this model
hts.forecasts = sapply(hts.models, function mdl) forecast::forecast(mdl, h = 1)$mean)
# Extract residuals
hts.residuals = sapply(hts.models, function(mdl) mdl$residuals)

# Compute reconciled forecasts
MinT(fcasts = hts.forecasts, Smat = S, residual = hts.residuals)

```

strucScaling

Structural Scaling reconciliation

Description

Structural Scaling reconciliation

Usage

```
strucScaling(fcasts, Smat, weights = rowSums(Smat))
```

Arguments

<code>fcsts</code>	forecasts to be reconciled
<code>Smat</code>	structure matrix representing the hierarchical structure of the hts
<code>weights</code>	use the default for structural scaling and a vector of the residual variances for variance scaling

Value

reconciled forecasts

Index

`accuracy.gts`, 2

`BU`, 2

`MinT`, 3

`strucScaling`, 4