

Package: icomb (via r-universe)

May 28, 2026

Title Forecasting Hierarchical Time Series Using Information Combination

Version 0.2.0

Description Implements the Information Combination (IComb) approach proposed by Nguyen, Vahid and Wickramasuriya (2025) <<https://www.monash.edu/business/ebs/research/publications/ebs/2025/wp11-2025.pdf>> for hierarchical forecast reconciliation. The method combines information from base forecasts constructed using different information sets while ensuring coherence. It is implemented using a penalized regression-based framework.

License GPL (>= 3)

Language en-US

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports cli, distributional, dplyr, fabletools (>= 0.7.0), future.apply, generics, glmnet, progressr, purrr, tsibble, vctrs

Suggests broom, fable, feasts, forecast, future, ggplot2, ggtime, hts, knitr, lubridate, plotly, rmarkdown, spelling, testthat (>= 3.0.0), tibble, tidy

VignetteBuilder knitr

Depends R (>= 4.1.0)

Config/testthat/edition 3

URL <https://shanikalw.github.io/icomb/>,
<https://github.com/ShanikaLW/icomb>

BugReports <https://github.com/ShanikaLW/icomb/issues>

Config/pak/sysreqs libicu-dev

Repository <https://shanikalw.r-universe.dev>

Date/Publication 2026-05-28 10:04:25 UTC

RemoteUrl <https://github.com/shanikalw/icomb>

RemoteRef HEAD

RemoteSha 9a54ca6468937886599c5a1ab686c1a65d4af8e1

Contents

forecast.mdl_icomb_lst	2
glance.mdl_icomb_lst	3
icomb	4
lambda_path	8

Index	10
--------------	-----------

forecast.mdl_icomb_lst

Produce coherent forecasts using the information combination method

Description

The forecast function allows you to generate coherent future predictions for hierarchical or grouped time series based on fitted models.

Usage

```
## S3 method for class 'mdl_icomb_lst'
forecast(
  object,
  new_data = NULL,
  h = NULL,
  point_forecast = list(.mean = mean),
  ...
)
```

Arguments

object	The time series models used to produce the forecasts.
new_data	A tsibble containing future information used to forecast.
h	The forecast horizon (can be used instead of new_data for regular time series with no exogenous regressors).
point_forecast	The point forecast measure(s) which should be returned in the resulting fable. Specify as a named list of functions which accept a distribution and return a vector.
...	Additional arguments for forecast model methods.

Details

The forecasts returned contain both point forecasts and their distribution. A specific forecast interval can be extracted from the distribution using the `distributional::hilo()` function. These intervals are stored in a single column using the `hilo` class, to extract the numerical upper and lower bounds you can use `fabletools::unpack_hilo()`.

Value

A fable containing the following columns:

- `.model`: The name of the model used to obtain the forecast. Taken from the column names of models in the provided mable.
- The forecast distribution. The name of this column will be the same as the dependent variable in the model(s). If multiple dependent variables exist, it will be named `.distribution`.
- Point forecasts computed from the distribution using the functions in the `point_forecast` argument. If `bootstrap = TRUE`, point forecasts are generated using bootstrapped sample paths.

Examples

```
library(fable)
library(fabletools)
library(tsibble)
library(distributional)

tourism_hts <- tourism |>
  aggregate_key(State,
                Trips = sum(Trips))

fit <- tourism_hts |>
  model(base = ETS(Trips)) |>
  reconcile(ols = min_trace(base, method = "ols"),
           icomb = icomb(base, train_size = 75))

fit |>
  forecast(bootstrap = TRUE, times = 1000) |>
  hilo(level = c(80, 95))
```

`glance.mdl_icomb_lst` *Glance a mable which contains an information combination reconciliation method*

Description

It uses the models within a mable to produce a one row summary statistics of their fits.

Usage

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

Arguments

x	A mable
...	Arguments for model methods

Value

The tibble contains the output of `glance()` for that model, with an added logical column `.included` indicating whether the corresponding node is present in the reconciliation process when the information combination method is used.

Examples

```
library(fable)
library(fabletools)
library(tsibble)
library(dplyr)

tourism_hts <- tourism |>
  aggregate_key(State,
                Trips = sum(Trips))

fit <- tourism_hts |>
  model(base = ETS(Trips)) |>
  reconcile(ols = min_trace(base, method = "ols"),
           icomb = icomb(base, train_size = 75))

fit |>
  glance()
```

 icomb

Information combination reconciliation

Description

Implements the Information Combination (IComb) method for forecast reconciliation, combining information from multiple base forecasts through a regression-based framework that can be estimated using penalized regression techniques. The penalty parameter is estimated using the rolling forecast origin cross-validation.

Usage

```
icomb(
  models,
  train_size,
  alpha = 1,
  standardize = FALSE,
  standardize_response = FALSE,
  intercept = TRUE,
  lambda = NULL,
  lambda_min_ratio = "expand",
  nlambda = 100,
  maxit = 1e+07,
  thresh = 1e-07,
  exact = TRUE
)
```

Arguments

models	A column of models in a mable.
train_size	The size of the initial training window.
alpha	The elasticnet mixing parameter, with $0 \leq \alpha \leq 1$. The penalty is defined as $(1 - \alpha)/2\ B\ _2^2 + \alpha\ B\ _2$ alpha = 1 is the group lasso penalty, and alpha = 0 is the ridge penalty.
standardize	Logical flag for x variable standardization, prior to fitting the model sequence. The coefficients are always returned on the original scale. Default is standardize = FALSE.
standardize_response	This allows the user to standardize the response variables. Default is standardize_response = FALSE.
intercept	Should intercepts be fitted (default = TRUE) or set to zero (FALSE).
lambda	A user supplied lambda sequence. Typical usage is to have the program compute its own lambda sequence based on nlambda and lambda_min_ratio. Supplying a value of lambda overrides this. Supply a decreasing sequence of lambda values.
lambda_min_ratio	The smallest value for lambda, as a fraction of lambda_max (the data derived value for which all coefficients are zero). lambda_min_ratio = "expand" (default) sets the ratio as $10^{-\lfloor \log_{10}(\lambda_{max}) \rfloor - 2}$ whereas lambda_min_ratio = "glmnet" sets the ratio to the value used in the glmnet package. If nobs < nvars, the default is 0.01, otherwise 1e-04.
nlambda	The number of lambda values. Default is 100.
maxit	Maximum number of passes over the data for all lambda values. Default is 10 ⁷ .
thresh	Convergence threshold for coordinate descent. Each inner coordinate-descent loop continues until the maximum change in the objective after any coefficient update is less than thresh times the null deviance. Defaults value is 1e-07.

exact A logical flag indicating whether to use a sequence of lambda values (from `lambda_max` to `lambda_best`) when fitting the final model on the entire dataset. The functions in the `glmnet` package are designed for efficiency by computing the entire regularization path (a sequence of lambda values) using "warm starts", which is often faster than computing a single fit. The default is `TRUE`.

Value

A 'global' model which is icomb coherent

Parallel computing

The model fitting and icomb cross-validation procedure can be parallelized using the `future` package. By specifying a `future::plan()` prior to model estimation or the reconciliation step, the computations will be carried out according to the chosen future strategy.

Progress

Progress on the cross-validation procedure can be obtained by wrapping the code with `progressr::with_progress()`. Further customization on how progress is reported can be controlled using the `progressr` package.

Note

Missing values are removed prior to applying the information combination method.

The default value of `lambda_min_ratio` may result in very small values of λ , which can increase computation time. It is therefore recommended to choose this parameter according to the specific requirements of your application.

Author(s)

Shanika L Wickramasuriya

References

Nguyen, M., Vahid, F., & Wickramasuriya, S. L. (2025). Hierarchical Forecasting: The Role of Information Combination (Working Paper No. 11/25). Department of Econometrics and Business Statistics, Monash University.

See Also

`fabletools::reconcile()`, `fabletools::aggregate_key()`

Examples

```
library(fable)
library(fabletools)
library(tsibble)
library(dplyr)
library(lubridate)
library(ggtime)
```

```
tourism_hts <- tourism |>
  aggregate_key(State,
               Trips = sum(Trips))

fit <- tourism_hts |>
  model(base = ETS(Trips)) |>
  reconcile(ols = min_trace(base, method = "ols"),
           icomb = icomb(base, train_size = 75))

fit |>
  forecast(h = "3 years")

# extracting results from icomb cross-validation
fit |>
  pull(icomb) |>
  attr("icombfit")

# Extracting probabilistic forecasts
fit |>
  forecast(h = "3 years", bootstrap = TRUE, times = 1000) |>
  filter(State == "Victoria") |>
  autoplot(filter(tourism_hts,
                 State == "Victoria", year(Quarter) > 2010))

# grouped structure
tourism_gts <- tourism |>
  aggregate_key(State * Purpose,
               Trips = sum(Trips))

fit <- tourism_gts |>
  model(base = ETS(Trips)) |>
  reconcile(ols = min_trace(base, method = "ols"),
           icomb = icomb(base, train_size = 75))

# Parallelizing cross-validation
library(future)
plan(multisession, workers = 2)

tourism_gts |>
  model(base = ETS(Trips)) |>
  reconcile(ols = min_trace(base, method = "ols"),
           icomb = icomb(base, train_size = 75)) |>
  forecast(h = "3 years")
plan(sequential)

fit <- tourism_gts |>
  model(base = ETS(Trips))

# progress
progressr::with_progress(
  fit |>
```

```
reconcile(ols = min_trace(base, method = "ols"),
          icomb = icomb(base, train_size = 35))
)
```

lambda_path

Calculate a sequence of penalty parameters

Description

Generates a decreasing sequence of regularization parameters for penalized multivariate regression by estimating the maximum lambda from the data and constructing a log-spaced path to a minimum value, with options for standardization and handling of zero-variance predictors and responses.

Usage

```
lambda_path(
  x,
  y,
  alpha = 1,
  standardize = FALSE,
  standardize_response = FALSE,
  intercept = TRUE,
  lambda_min_ratio = "expand",
  nlambda = 100
)
```

Arguments

x Input matrix, of dimension nobs x nvars; each row is an observation vector. nvars should be greater than one. In other words x should have 2 or more columns. x can be in the sparse matrix format (inherit from class "sparseMatrix" as in package Matrix).

y A matrix of quantitative responses.

alpha The elasticnet mixing parameter, with $0 \leq \alpha \leq 1$. The penalty is defined as

$$(1 - \alpha)/2 \|B\|_2^2 + \alpha \|B\|_2$$

alpha = 1 is the group lasso penalty, and alpha = 0 is the ridge penalty.

standardize Logical flag for x variable standardization, prior to fitting the model sequence. The coefficients are always returned on the original scale. Default is standardize = FALSE.

standardize_response This allows the user to standardize the response variables. Default is standardize_response = FALSE.

intercept Should intercepts be fitted (default = TRUE) or set to zero (FALSE).

lambda_min_ratio	The smallest value for lambda, as a fraction of lambda_max (the data derived value for which all coefficients are zero). lambda_min_ratio = "expand" (default) sets the ratio as $10^{-[\log_{10}(\lambda_{max})]-2}$ whereas lambda_min_ratio = "glmnet" sets the ratio to the value used in the glmnet package. If nobs < nvars, the default is 0.01, otherwise 1e-04.
nlambda	The number of lambda values. Default is 100.

Value

A sequence of penalty parameters

Author(s)

Shanika L Wickramasuriya

Examples

```
set.seed(2024)
n <- 100
p <- 50
k <- 2
x <- matrix(rnorm(n * p), ncol = p)
y <- matrix(rnorm(n * k), ncol = k)
glmnet::glmnet(x, y, family = "mgaussian", standardize = FALSE,
               standardize.response = FALSE, intercept = TRUE)$lambda
lambda_path(x, y, lambda_min_ratio = "glmnet")
```

Index

distributional::hilo(), 3

fabletools::aggregate_key(), 6

fabletools::reconcile(), 6

fabletools::unpack_hilo(), 3

forecast.mdl_icomb_lst, 2

future::plan(), 6

glance.mdl_icomb_lst, 3

icomb, 4

lambda_path, 8