Package: modelimportance (via r-universe)

July 16, 2026

Type Package

Title Measuring Contributions of Component Models to Ensemble Forecast Accuracy

Version 0.1.0

Description Provides metrics for quantifying the contribution of individual component models to the predictive accuracy of ensemble forecasts. The package implements the Leave-One-Model-Out (LOMO) and Leave-All-Subset-of-One-Model-Out (LASOMO) model importance metrics, enabling users to assess the relative importance of component models and better understand the performance of ensemble forecasting systems. Methods are described in Kim et al. (2026) <doi:10.1016/j.ijforecast.2025.12.006>.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Imports hubUtils (>= 0.4.0), dplyr (>= 1.1.4), hubEvals (>= 0.3.0), hubEnsembles (>= 0.1.9), methods (>= 4.4.3), purrr (>= 1.0.4), furrr (>= 0.3.1), future (>= 1.49.0), checkmate (>= 2.3.3), rlang (>= 1.1.6), stats (>= 4.4.3)

Depends R (>= 4.1.0)

Suggests knitr, rmarkdown, tidyr (>= 1.3.1), kableExtra (>= 1.4.0), ggplot2 (>= 4.0.1), scoringutils (>= 2.1.2), testthat (>= 3.0.0), progressr (>= 0.15.1)

Config/testthat/edition 3

VignetteBuilder knitr

URL https://github.com/mkim425/modelimportance

Config/Needs/website rmarkdown

BugReports https://github.com/mkim425/modelimportance/issues

LazyData true

NeedsCompilation no

Author Minsu Kim [aut, cre, cph] (ORCID:
<https://orcid.org/0009-0008-4637-3589>), Li Shandross [aut,
ctb] (ORCID: <https://orcid.org/0009-0008-1348-1954>), Zhian
Kamvar [ctb] (ORCID: <https://orcid.org/0000-0003-1458-7108>),
Nicholas Reich [aut] (ORCID:
<https://orcid.org/0000-0003-3503-9899>), Evan Ray [aut]

Maintainer Minsu Kim <minsu@umass.edu>

Repository https://cran.r-universe.dev

Date/Publication 2026-07-16 13:30:02 UTC

RemoteUrl https://github.com/cran/modelimportance

RemoteRef HEAD

RemoteSha fad92f5cc3ea3b63532cc010ad9635ace1f6fa2a

Contents

aggregate.model_imp_tbl 2
forecast_data_example 4
forecast_data_ma_h1 4
forecast_data_raw 5
model_importance 6
print.model_imp_tbl 9
print.summary.model_imp_tbl 9
summary.model_imp_tbl 10
target_data_example 10
target_data_ma 11
target_data_raw 12

Index **13**

`aggregate.model_imp_tbl` *Aggregate model importance scores across tasks to compute overall importance for each model*

Description

Aggregate model importance scores across tasks to compute overall importance for each model

Usage

```
## S3 method for class 'model_imp_tbl'
aggregate(
  x,
  by = "model_id",
  na_action = c("drop", "worst", "average"),
  fun = mean,
  ...
)
```

Arguments

x	An object of class model_imp_tbl.
by	A character vector with column names specifying the grouping variable(s) for summarization. Default is "model_id", which summarizes importance scores for each model across all tasks.
na_action	A character string specifying how to handle NA values generated during importance score calculation for each task, occurring when a model did not contribute to the ensemble prediction for a given task by missing its forecast submission. Three options are available: c("drop", "worst", "average"). For each specific prediction task, each option works as follows: • "drop" removes NAs. • "worst" replaces NAs with the smallest value among importance metrics of the other models. • "average" replaces NAs with the average value from the other models' importance metrics.
fun	A function used to summarize importance scores. Default is mean()
...	Additional arguments passed to the summary function fun. (e.g., fun = quantile, probs = 0.25 for a quartile summary)

Details

This method extends stats::aggregate for objects of class model_imp_tbl.

Value

A data frame with columns model_id and importance_score_<fun>, where <fun> is the name of the summary function used (e.g., importance_score_mean when fun = mean). The output is sorted in descending order of the summary importance scores.

forecast_data_example *Example forecast outputs for modelimportance article vignette*

Description

A subset of `hubExamples::forecast_outputs` filtered to quantile output type and location "25" (Massachusetts), covering the 2022/23 influenza season (from "2022-11-19" to "2023-01-07").

Usage

```
forecast_data_example
```

Format

A data frame (model_out_tbl) with 168 rows and 9 columns:

model_id Character. Unique model identifier.

reference_date Date. Date the forecast was generated.

target Character. Forecast target.

horizon Integer. Number of weeks ahead.

location Character. FIPS code for the US location.

target_end_date Date. Target date for the forecast.

output_type Character. Prediction representation type.

output_type_id Character. Identifier for the output type level.

value Numeric. Predicted value.

Source

Sourced from `hubExamples::forecast_outputs` (hubverse-org/hubExamples v1.0.0).

forecast_data_ma_h1 *Forecast outputs for Massachusetts, horizon 1, used in vignette runtime data*

Description

A subset of `hubExamples::forecast_outputs` filtered to median output type, location "25" (Massachusetts), horizon 1, and target end date 2022-12-24.

Usage

```
forecast_data_ma_h1
```

Format

A data frame (model_out_tbl) with 3 rows and 9 columns:

model_id Character. Unique model identifier.
reference_date Date. Date the forecast was generated.
target Character. Forecast target.
horizon Integer. Number of weeks ahead.
location Character. FIPS code for the US location.
target_end_date Date. Target date for the forecast.
output_type Character. Prediction representation type.
output_type_id Character. Identifier for the output type level.
value Numeric. Predicted value.

Source

Sourced from `hubExamples::forecast_outputs` (hubverse-org/hubExamples v1.0.0).

`forecast_data_raw` *Raw forecast outputs for get-started vignette*

Description

A subset of `hubExamples::forecast_outputs` filtered to median output type and two target end dates (2022-11-26, 2022-12-10).

Usage

```
forecast_data_raw
```

Format

A data frame (model_out_tbl) with 12 rows and 9 columns:

model_id Character. Unique model identifier.
reference_date Date. Date the forecast was generated.
target Character. Forecast target.
horizon Integer. Number of weeks ahead.
location Character. FIPS code for the US location.
target_end_date Date. Target date for the forecast.
output_type Character. Prediction representation type.
output_type_id Character. Identifier for the output type level.
value Numeric. Predicted value.

Source

Sourced from `hubExamples::forecast_outputs` (hubverse-org/hubExamples v1.0.0).

`model_importance` *Quantifies the contribution of ensemble component models to ensemble prediction accuracy for each prediction task.*

Description

We measure each ensemble component model's contribution to the ensemble prediction accuracy for each model task.

This function requires that one column represent the forecast date (or a date from which each forecast originates or is made in reference to) and that column be named one of `forecast_date`, `origin_date`, and `reference_date`.

For each `output_type`, the corresponding scoring rule applied to calculate the importance is as follows.

Output Type	Scoring Rule	Description
mean	se_point	evaluate using the squared error
median	ae_point	evaluate using the absolute error
quantile	wis	evaluate using the weighted interval score
pmf	log_score	evaluate using the logarithm of the probability assigned to the true outcome

Usage

```
model_importance(
  forecast_data,
  oracle_output_data,
  ensemble_fun = c("simple_ensemble", "linear_pool"),
  importance_algorithm = c("lomo", "lasomo"),
  subset_wt = c("equal", "perm_based"),
  min_log_score = -10,
  ...
)
```

Arguments

`forecast_data`	A data.frame with the predictions that is or can be coerced to a `model_out_tbl` format, which is the standard S3 class model output format defined by the 'hubverse' convention (https://docs.hubverse.io/en/latest/#). If it fails to be coerced to a `model_out_tbl` format, an error message will be returned. Only one `output_type` is allowed in the data.frame, and it must be one of the following: `mean`, `median`, `quantile`, or `pmf`.
`oracle_output_data`	Ground truth data for the variables that are used to define modeling targets. This data must follow the oracle output format. See 'Details'.
`ensemble_fun`	A character string specifying a ensemble method, either "simple_ensemble" or "linear_pool"; `c("simple_ensemble", "linear_pool")`.

- When "simple_ensemble" is specified, the ensemble is generated using the optional agg_fun function in ... (see 'Details').
- When "linear_pool" is specified, ensemble model outputs are created as a linear pool of component model outputs. This method supports only an output_type of mean, quantile, or pmf.

importance_algorithm
: A character string specifying algorithm for model importance calculation; c("lomo", "lasomo"). "lomo" stands for leave-one-model-out and "lasomo" stands for leave all subsets of models out. For "lasomo", 'furrr' and 'future' packages need to be installed for parallel execution.

subset_wt
: A character string specifying method for assigning weight to subsets when using lasomo algorithm; c("equal", "perm_based").

- "equal" assigns equal weight to all subsets.
- "perm_based" assigns weight averaged over all possible permutations as in the Shapley value. Ignored if lomo method is used. Default is "equal", if not specified.

min_log_score
: A numeric value specifying a minimum threshold for log scores for the pmf output. This threshold prevents issues with extremely low probabilities assigned to the true outcome, which would otherwise lead to undefined or negative infinite log scores. Any probability lower than this threshold will be adjusted to this minimum value. The default value is set to -10, following the CDC FluSight thresholding convention. Users may choose a different value based on their practical needs.

...
: Optional arguments passed to ensemble_fun when it is specified as "simple_ensemble". See 'Details'.

Details

The oracle_output_data is a data frame that contains the ground truth values for the variables used to define modeling targets. It is referred to as "oracle" because it is formatted as if an oracle made a perfect point prediction equal to the truth. This data must follow the oracle output format defined in the hubverse standard, which includes independent task ID columns (e.g., location, target_date), the output_type column specifying the output type of the predictions and an oracle_value column for the observed values. As in the forecast data, if the output_type is either "quantile" or "pmf", the output_type_id column is often required to provide further identifying information.

The model_out_tbl and oracle_output_data must have the same task ID columns and output_type, including output_type_id if necessary, which are used to match the predictions with the ground truth data.

Additional argument in ... is agg_fun, which is a character string name for a function specifying aggregation method of component model outputs. Default is mean, indicating that equally weighted mean is calculated across all component model outputs for each unique output_type_id. This can be median or a custom function (e.g., geometric_mean. Details can be found in https://hubverse-org.github.io/hubEnsembles/articles/hubEnsembles.html).

This function uses the furrr and future for parallelization. To enable parallel execution, please set a parallel backend, e.g., via future::plan().

Value

A `model_imp_tbl` S3 class object with columns `model_id`, `reference_date`, `output_type`, and `importance`, along with any task ID columns (e.g., `location`, `horizon`, and `target_end_date`) present in the input `forecast_data`. Note that `reference_date` is used as the name for the forecast date column, regardless of its original name in the input `forecast_data`.

Progress reporting

Optional progress bars are displayed via the `progressr` package when it is installed and the session is interactive. If `progressr` is not installed, the function will run without progress bars. To enable progress bars,

```
progressr::handlers(global = TRUE)
```

Aggregation over tasks

model importance scores calculated for individual prediction tasks can be aggregated across multiple tasks to obtain an overall importance score for each model via the `aggregate()` method for `model_imp_tbl` objects. Users can summarize the task-level scores with specified summary statistics of interest (e.g., mean, median, quantiles) for a more comprehensive view of model importance. See `aggregate.model_imp_tbl()` for more details on how to use this method.

Examples

```
library(dplyr)
forecast_data <- forecast_data_raw |>
  dplyr::filter(
    location == "25",
    horizon == 1
  )
target_data <- target_data_raw |>
  dplyr::filter(
    target_end_date %in% unique(forecast_data$target_end_date),
    location == "25"
  )
# Example with the default arguments.
model_importance(
  forecast_data = forecast_data, oracle_output_data = target_data,
  ensemble_fun = "simple_ensemble", importance_algorithm = "lomo",
  subset_wt = "equal"
)
# Example with the additional argument in `...`.
model_importance(
  forecast_data = forecast_data, oracle_output_data = target_data,
  ensemble_fun = "simple_ensemble", importance_algorithm = "lomo",
  subset_wt = "equal", agg_fun = median
)
```

print.model_imp_tbl *Print method for model importance score table*

Description

Print method for model importance score table

Usage

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

Arguments

x	An object of class model_imp_tbl.
...	Additional arguments passed to the print method.

Value

x, returned invisibly. Called for its side effect of printing the model importance score table (a model_imp_tbl object, i.e., a data frame of per-task importance scores by model) to the console.

print.summary.model_imp_tbl *Print method for summary of model importance score table*

Description

Print method for summary of model importance score table

Usage

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

Arguments

x	An object of class summary.model_imp_tbl.
...	Additional arguments passed to the print method.

Value

x, returned invisibly. Called for its side effect of printing a human-readable summary (number of models and tasks, and the top-scoring model for a subset of tasks) to the console.

summary.model_imp_tbl *Summary method for model importance score table*

Description

Summary method for model importance score table

Usage

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

Arguments

object	An object of class model_imp_tbl.
...	Additional arguments passed to the print method.

Value

A list of class summary.model_imp_tbl with four elements:

all_models A character vector of the unique model IDs present in object.

all_tasks A data frame with one row per unique prediction task (one column per task ID, e.g. location, horizon, target_end_date).

model_summary A data frame with one row per model and columns model_id, n_tasks (number of tasks the model was scored on), min_importance, max_importance, and n_NA (number of tasks with a missing importance score).

task_winners A data frame with one row per prediction task and columns for the task ID(s), top_model (the model with the highest importance score for that task), and max_score (that model's importance score).

This object is printed via print.summary.model_imp_tbl.

target_data_example *Example target data for modelimportance article vignette*

Description

A subset of hubExamples::forecast_target_ts matching the locations and target end dates in forecast_data_example.

Usage

```
target_data_example
```

Format

A data frame with 8 rows and 4 columns:

target_end_date Date. Date of the observation.

target Character. Target name.

location Character. FIPS code for the US location.

observation Numeric. Observed value.

Source

Sourced from `hubExamples::forecast_target_ts` (hubverse-org/hubExamples v1.0.0).

`target_data_ma` *Target data for Massachusetts used in vignette runtime data*

Description

A subset of `hubExamples::forecast_target_ts` for location "25" (Massachusetts) and target end date 2022-12-24.

Usage

```
target_data_ma
```

Format

A data frame with 1 row and 4 columns:

target_end_date Date. Date of the observation.

target Character. Target name.

location Character. FIPS code for the US location.

observation Numeric. Observed value.

Source

Sourced from `hubExamples::forecast_target_ts` (hubverse-org/hubExamples v1.0.0).

`target_data_raw` *Raw target data for get-started vignette*

Description

A subset of `hubExamples::forecast_target_ts` matching the locations and target end dates in `forecast_data_raw`, with the observed values column renamed to `oracle_value`.

Usage

```
target_data_raw
```

Format

A data frame with 4 rows and 4 columns:

target_end_date Date. Date of the observation.

target Character. Target name.

location Character. FIPS code for the US location.

oracle_value Numeric. Observed value.

Source

Sourced from `hubExamples::forecast_target_ts` (hubverse-org/hubExamples v1.0.0).

Index

∗ **datasets**
forecast_data_example, 4
forecast_data_ma_h1, 4
forecast_data_raw, 5
target_data_example, 10
target_data_ma, 11
target_data_raw, 12

aggregate.model_imp_tbl, 2
aggregate.model_imp_tbl(), *8*

forecast_data_example, 4
forecast_data_ma_h1, 4
forecast_data_raw, 5

model_importance, 6

print.model_imp_tbl, 9
print.summary.model_imp_tbl, 9, *10*

summary.model_imp_tbl, 10

target_data_example, 10
target_data_ma, 11
target_data_raw, 12