

Package: hubEvals (via r-universe)

September 17, 2024

Title Basic tools for scoring hubverse forecasts

Version 0.0.0.9001

Description Using functionality from the scoringutils package, this software provides basic tools for scoring hubverse forecasts.

License MIT + file LICENSE

Encoding UTF-8

Imports cli, dplyr, hubUtils, purrr, rlang, scoringutils (>= 1.2.2.9000)

Remotes epiforecasts/scoringutils, hubverse-org/hubExamples, hubverse-org/hubUtils

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

URL <https://hubverse-org.github.io/hubEvals/>

Depends R (>= 2.10)

LazyData true

Config/Needs/website hubverse-org/hubStyle

Suggests hubExamples, testthat (>= 3.0.0)

Config/testthat/edition 3

Repository <https://hubverse-org.r-universe.dev>

RemoteUrl <https://github.com/hubverse-org/hubEvals>

RemoteRef HEAD

RemoteSha 9a87ce89c7dab1cbbba2dca7efb845d3ba8b069b

Contents

score_model_out	2
transform_pmf_model_out	4
transform_point_model_out	5
transform_quantile_model_out	5
Index	7

score_model_out	<i>Score model output predictions</i>
-----------------	---------------------------------------

Description

Scores model outputs with a single output_type against observed data.

Usage

```
score_model_out(
  model_out_tbl,
  target_observations,
  metrics = NULL,
  summarize = TRUE,
  by = "model_id",
  output_type_id_order = NULL
)
```

Arguments

model_out_tbl	Model output tibble with predictions
target_observations	Observed 'ground truth' data to be compared to predictions
metrics	Character vector of scoring metrics to compute. If NULL (the default), appropriate metrics are chosen automatically. See details for more.
summarize	Boolean indicator of whether summaries of forecast scores should be computed. Defaults to TRUE.
by	Character vector naming columns to summarize by. For example, specifying by = "model_id" (the default) will compute average scores for each model.
output_type_id_order	For ordinal variables in pmf format, this is a vector of levels for pmf forecasts, in increasing order of the levels. For all other output types, this is ignored.

Details

Default metrics are provided by the `scoringutils` package. You can select metrics by passing in a character vector of metric names to the `metrics` argument.

The following metrics can be selected (all are used by default) for the different output_types:

Quantile forecasts: (output_type == "quantile")

- wis
- overprediction
- underprediction
- dispersion

- bias
- interval_coverage_deviation
- ae_median
- "interval_coverage_XX": interval coverage at the "XX" level. For example, "interval_coverage_95" is the 95% interval coverage rate, which would be calculated based on quantiles at the probability levels 0.025 and 0.975.

See [scoringutils::get_metrics.forecast_quantile](#) for details.

Nominal forecasts: (output_type == "pmf" and output_type_id_order is NULL)

- log_score

(scoring for ordinal forecasts will be added in the future).

See [scoringutils::get_metrics.forecast_nominal](#) for details.

Median forecasts: (output_type == "median")

- ae_point: absolute error of the point forecast (recommended for the median, see Gneiting (2011))

See [scoringutils::get_metrics.forecast_point](#) for details.

Mean forecasts: (output_type == "mean")

- se_point: squared error of the point forecast (recommended for the mean, see Gneiting (2011))

Value

A data.table with scores

References

Making and Evaluating Point Forecasts, Gneiting, Tilmann, 2011, Journal of the American Statistical Association.

Examples

```
# compute WIS and interval coverage rates at 80% and 90% levels based on
# quantile forecasts, summarized by the mean score for each model
quantile_scores <- score_model_out(
  model_out_tbl = hubExamples::forecast_outputs |>
    dplyr::filter(.data[["output_type"]] == "quantile"),
  target_observations = hubExamples::forecast_target_observations,
  metrics = c("wis", "interval_coverage_80", "interval_coverage_90"),
  by = "model_id"
)
quantile_scores
```

```
# compute log scores based on pmf predictions for categorical targets,
# summarized by the mean score for each combination of model and location.
# Note: if the model_out_tbl had forecasts for multiple targets using a
```

```
# pmf output_type with different bins, it would be necessary to score the
# predictions for those targets separately.
pmf_scores <- score_model_out(
  model_out_tbl = hubExamples::forecast_outputs |>
    dplyr::filter(.data[["output_type"]] == "pmf"),
  target_observations = hubExamples::forecast_target_observations,
  metrics = "log_score",
  by = c("model_id", "location", "horizon")
)
head(pmf_scores)
```

transform_pmf_model_out

Transform pmf model output into a forecast object

Description

Transform pmf model output into a forecast object

Usage

```
transform_pmf_model_out(
  model_out_tbl,
  target_observations,
  output_type_id_order = NULL
)
```

Arguments

`model_out_tbl` Model output tibble with predictions

`target_observations` Observed 'ground truth' data to be compared against predictions

`output_type_id_order` For nominal variables, this should be NULL (the default). For ordinal variables, this is a vector of levels for pmf forecasts, in increasing order of the levels.

Value

forecast_quantile

`transform_point_model_out`

Transform either mean or median model output into a point forecast object:

Description

Transform either mean or median model output into a point forecast object:

Usage

```
transform_point_model_out(model_out_tbl, target_observations, output_type)
```

Arguments

`model_out_tbl` Model output tibble with predictions
`target_observations` Observed 'ground truth' data to be compared against predictions
`output_type` Forecast output type: "mean" or "median"

Details

This function transforms a model output tibble in the Hubverse format (with either "mean" or "median" output type) to a scoringutils "point" forecast object

Value

forecast_point

`transform_quantile_model_out`

Transform quantile model output into a forecast object

Description

Transform quantile model output into a forecast object

Usage

```
transform_quantile_model_out(model_out_tbl, target_observations)
```

Arguments

`model_out_tbl` Model output tibble with predictions
`target_observations` Observed 'ground truth' data to be compared against predictions

6

transform_quantile_model_out

Value

forecast_quantile

Index

score_model_out, [2](#)
scoringutils::get_metrics.forecast_nominal,
[3](#)
scoringutils::get_metrics.forecast_point,
[3](#)
scoringutils::get_metrics.forecast_quantile,
[3](#)

transform_pmf_model_out, [4](#)
transform_point_model_out, [5](#)
transform_quantile_model_out, [5](#)