

Package: hubPredEvalsData (via r-universe)

June 3, 2026

Title Data Creation for Hub Prediction Evaluations via predevals

Version 1.1.0

Description This package generates data objects with scores for predictions in a hub, to be displayed on a dashboard using predevals.

License MIT + file LICENSE

Imports cli, dplyr (>= 1.1.0), hubData, hubEvals, hubUtils, jsonlite, jsonvalidate, methods, purrr, rlang, scoringutils (>= 2.2.0), yaml

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

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Suggests testthat (>= 3.0.0), withr

Config/testthat/edition 3

URL <https://github.com/hubverse-org/hubPredEvalsData>

BugReports <https://github.com/hubverse-org/hubPredEvalsData/issues>

Config/Needs/website hubverse-org/hubStyle

Config/pak/sysreqs cmake git make libicu-dev libuv1-dev libssl-dev libnode-dev libx11-dev

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

Date/Publication 2026-06-03 13:28:30 UTC

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

RemoteRef v1.1.0

RemoteSha 1d4a7f6e0c3a9d0bc853a3825b743badec172360

Contents

generate_eval_data	2
generate_predevals_options	3
read_predevals_config	4

generate_eval_data	<i>Generate evaluation data for a hub</i>
--------------------	---

Description

Scores each target configured in the predevals config against its oracle data and writes wide-format score tables to disk.

Usage

```
generate_eval_data(hub_path, config_path, out_path, oracle_output = NULL)
```

Arguments

hub_path	A path to the hub.
config_path	A path to a yaml file that specifies the configuration options for the evaluation.
out_path	The directory to write the evaluation data to.
oracle_output	Optional data frame of oracle output to use for the evaluation. When NULL (the default), oracle output is discovered from hub_path via <code>hubData::connect_target_oracle_output()</code> . Supplying a pre-loaded data frame remains supported for back-compat with callers that load oracle data themselves.

Output

For each (target, eval_set, disaggregation) requested in the config, a scores.csv is written under out_path/<target_id>/<eval_set_name>[/<by>]/ in wide format, with one row per model per disaggregation key.

When a target has a configured transform, transformed-scale metrics appear as <metric>__<label>-suffixed columns (e.g. wis__log) alongside the natural-scale columns. Setting append: false emits only the suffixed columns. Transform-invariant metrics (interval_coverage_<n> and bias, whose values are unchanged by any monotonic transform) always appear only under their natural-scale name regardless of append.

For the full configuration schema, see the JSON Schema files installed with the package under `system.file("schema", package = "hubPredEvalsData")`.

 generate_predevals_options

Generate the contents of the predevals-options.json file

Description

Assembles the contents of the predevals-options.json file that initialises the predevals dashboard. The dashboard reads this file on load to build its menus and labelling: which targets and evaluation sets a user can select, which metrics can be displayed for each target, which task-id variables the scores can be disaggregated by, and the human-readable text for task-id values. It is the dashboard's index of what can be shown; the score values themselves come from the per-target scores.csv files written by [generate_eval_data\(\)](#).

Usage

```
generate_predevals_options(hub_path, config_path)
```

Arguments

hub_path	A path to the hub.
config_path	A path to a yaml file that specifies the configuration options for the evaluation.

Details

The result is the validated predevals config with each entry of targets augmented with the metric and transform metadata the dashboard needs to populate its metric selector:

- **metrics**: the metric columns present in the target's scores.csv, in column order. A `<metric>_scaled_relative_ski` entry is spliced in before each metric listed in `relative_metrics`. When a transform applies, the transformed-scale `<metric>__<label>` columns are appended, or replace the natural-scale columns when `append: false`. Transform-invariant metrics (`interval_coverage_<n>`, `bias`) are unchanged by any monotonic transform and so always appear only under their natural-scale name. This is the list the dashboard iterates to build its metric selector.
- **transform**: the resolved transform for the target (with `transform_defaults` inheritance applied), as a list with `fun`, `label`, `append` and a composed human-readable description the dashboard can show to explain the transformed scale to the user. The key is absent when no transform applies, including targets that opt out with `transform: false` and targets whose available output types are all non-transformable (where the transform is skipped at scoring time).

The config is read via [read_predevals_config\(\)](#), so the config schema and transform validation run as a side effect. The caller decides where to write the result.

Value

The predevals config as a list, with each entry of targets augmented as described above.

read_predevals_config *Load and validate a predevals config file*

Description

Load and validate a predevals config file

Usage

```
read_predevals_config(hub_path, config_path)
```

Arguments

hub_path	A path to the hub.
config_path	A path to a yaml file that specifies the configuration options for the evaluation.

Value

A list of configuration options for the evaluation.

Index

`generate_eval_data`, [2](#)

`generate_eval_data()`, [3](#)

`generate_predevals_options`, [3](#)

`hubData::connect_target_oracle_output()`,
[2](#)

`read_predevals_config`, [4](#)

`read_predevals_config()`, [3](#)