Package: uncertR (via r-universe)

November 3, 2026

Type Package

Title Bootstrap Prediction Uncertainty and Its Biological Drivers in Gene-Expression Models

Version 0.1.3

Description Fits bootstrap ensembles of penalized regression models on bulk gene-expression data, derives per-sample uncertainty scores from out-of-bag predictions, and tests whether technical covariates explain that uncertainty. The bootstrap method is described by Efron and Tibshirani (1993, ISBN:9780412042317), and elastic-net regression follows Friedman, Hastie, and Tibshirani (2010) <doi:10.18637/jss.v033.i01>. The optional immunedeconv package, used for immune-cell deconvolution, is available from GitHub <https://github.com/omnideconv/immunedeconv>.

Encoding UTF-8

Depends R (>= 4.2.0)

License MIT + file LICENSE

Imports glmnet, future.apply, splines, stats, graphics

Suggests testthat (>= 3.0.0), SummarizedExperiment, knitr, rmarkdown, immunedeconv

VignetteBuilder knitr

Config/roxygen2/version 8.1.0

NeedsCompilation no

Author Tanay Belapurkar [aut, cre]

Maintainer Tanay Belapurkar <btanay041@gmail.com>

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

Date/Publication 2026-11-03 15:24:05 UTC

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

RemoteRef HEAD

RemoteSha f5ba096a75d4b8b3f9826276353eec796f36921e

Contents

boot_uncertainty 2
clr_transform 3
collapse_rare_levels 3
tumor_covariates 4
uncertainty_drivers 4
uncertainty_score 5
validate_uncertainty 6

Index **7**

`boot_uncertainty` *Bootstrap ensemble with out-of-bag predictions*

Description

Resamples samples with replacement `B` times, fits an elastic-net model on each bootstrap sample, and predicts only the samples that were *not* drawn (out-of-bag, OOB). Every sample therefore accumulates a distribution of honest (non-in-sample) predictions.

Usage

```
boot_uncertainty(
  expr,
  y,
  B = 200,
  alpha = 0.5,
  lambda_mode = c("global", "per_boot"),
  nfolds = 5,
  seed = NULL
)
```

Arguments

`expr`	Genes x samples matrix (bulk expression, e.g. log2 TPM), or a `SummarizedExperiment` (first assay is used).
`y`	Numeric outcome vector (e.g. log IC50 or AUC), one per sample.
`B`	Number of bootstrap replicates.
`alpha`	glmnet mixing parameter (1 = lasso, 0 = ridge).
`lambda_mode`	`"global"` selects lambda once by CV on all data (fast; slight optimism because OOB samples informed lambda). `"per_boot"` runs inner CV in every replicate (slower, cleaner).
`nfolds`	Folds for cross-validation.
`seed`	Optional integer for reproducibility.

Value

An object of class `boot_unc`.

`clr_transform` *Centered log-ratio transform for compositional fractions*

Description

Cell-type fractions sum to a constant, so use CLR before regression.

Usage

```
clr_transform(fractions, pseudo = 0.001)
```

Arguments

`fractions`	Samples x cell types matrix of non-negative fractions.
`pseudo`	Pseudocount replacing zeros.

Value

A numeric matrix with the same dimensions as `fractions`, containing centered log-ratio-transformed values. Each row is centered by subtracting the mean of its log-transformed components.

`collapse_rare_levels` *Collapse rare factor levels*

Description

Tissue/lineage labels often have many tiny groups that make the driver model unstable. Levels with fewer than `min_n` samples become `"Other"`.

Usage

```
collapse_rare_levels(x, min_n = 15)
```

Arguments

`x`	Character or factor vector.
`min_n`	Minimum group size.

Value

A factor in which levels represented by fewer than `min_n` observations are replaced with `"Other"`. Levels meeting the threshold are retained.

tumor_covariates *Assemble covariates that may drive uncertainty*

Description

Combines technical covariates computed from `expr`, optional deconvolution scores (patient tumors only; not meaningful for cell lines), and any user-supplied covariates (e.g. tissue lineage, growth rate, mutation status).

Usage

```
tumor_covariates(expr, methods = NULL, extra = NULL, technical = TRUE)
```

Arguments

`expr`	Genes x samples matrix (HGNC symbols, TPM-like scale for deconvolution).
`methods`	Character vector of `immunedeconv` methods (e.g. `"estimate"`, `"quantiseq"`); `NULL` to skip.
`extra`	Optional data frame of extra covariates; rownames = samples.
`technical`	If `TRUE`, add log library size and number of detected genes. Only meaningful for raw/TPM RNA-seq counts; set `FALSE` for RMA-normalized microarray data such as GDSC.

Value

Data frame, one row per sample.

uncertainty_drivers *Which covariates explain prediction uncertainty?*

Description

Fits a joint linear model of (log) uncertainty on covariates and reports the partial R^2 of each covariate with a permutation p-value and BH FDR. Optionally removes the dependence of uncertainty on the predicted value first, since uncertainty is often mechanically tied to how extreme a prediction is.

Usage

```
uncertainty_drivers(
  unc,
  covariates,
  adjust_for = c("prediction", "none"),
  log_transform = TRUE,
  n_perm = 500,
  seed = NULL
)
```

Arguments

unc	An unc_scores object.
covariates	Data frame of covariates, rownames = sample IDs.
adjust_for	"prediction" residualizes on a spline of the mean OOB prediction; "none" skips this.
log_transform	Model log(uncertainty) (recommended; positive, skewed).
n_perm	Number of permutations.
seed	Optional seed.

Value

An object of class unc_drivers.

---

uncertainty_score *Per-sample uncertainty score from OOB predictions*

Description

Per-sample uncertainty score from OOB predictions

Usage

```
uncertainty_score(
  fit,
  metric = c("ci_width", "sd", "iqr"),
  level = 0.95,
  min_oob = 10
)
```

Arguments

fit	A boot_unc object.
metric	"sd", "ci_width" (percentile interval), or "iqr".
level	Interval level for "ci_width".
min_oob	Samples with fewer OOB predictions get NA.

Value

An object of class unc_scores holding a data frame with pred_mean, uncertainty, y, abs_error, and n_oob.

`validate_uncertainty` *Does the uncertainty score track real error?*

Description

Run this *before* interpreting drivers. If uncertainty is not associated with absolute out-of-bag error, explaining it is not meaningful.

Usage

```
validate_uncertainty(unc)
```

Arguments

unc	An `unc_scores` object.

Value

List with Spearman correlation and mean error by uncertainty quartile.

Index

boot_uncertainty, 2

clr_transform, 3
collapse_rare_levels, 3

tumor_covariates, 4

uncertainty_drivers, 4
uncertainty_score, 5

validate_uncertainty, 6