

Package: ExtremeConformal (via r-universe)

June 25, 2026

Type Package

Title Extreme Conformal Prediction Intervals

Version 0.2.2

Description This new extreme conformal prediction framework provides informative prediction intervals at the high-confidence levels for which classical conformal methods fail. In applications with potentially high-impact events, a very high level of confidence is often required for predictions. If that level is too large relative to the amount of data used for calibration, classical conformal methods provide infinitely wide, thus, uninformative prediction intervals. Our extreme conformal procedure bridges extreme value statistics and conformal prediction to provide reliable and informative prediction intervals with high-confidence coverage, which can be constructed using any black-box extreme quantile regression method. A weighted version of the approach can account for nonstationary data. The methodology was introduced in Pasche, Lam, and Engelke (2026) <[doi:10.1007/s10687-026-00536-9](https://doi.org/10.1007/s10687-026-00536-9)>.

License GPL (>= 3)

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

Imports ExtremeCI, extRemes, ismev, stats

Config/roxygen2/version 8.0.0

URL <https://github.com/opusche/ExtremeConformal>,
<https://opusche.github.io/ExtremeConformal/>

BugReports <https://github.com/opusche/ExtremeConformal/issues>

Config/pak/sysreqs libicu-dev

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

Date/Publication 2026-06-24 12:02:11 UTC

RemoteUrl <https://github.com/opasche/extremeconformal>

RemoteRef HEAD

RemoteSha f9180e297652bbe1637a348196671eb7903e5c29

Contents

block_weighted_conformal_PI_unilat	2
conformal_PI_unilat	3
conformalize_EQR_unilat	4

Index	8
--------------	----------

block_weighted_conformal_PI_unilat
Single-sided block-weighted conformal prediction interval from conformalizer

Description

Single-sided block-weighted conformal prediction interval from conformalizer

Usage

```
block_weighted_conformal_PI_unilat(
  Q_pred,
  block_ids,
  dQ_blocks,
  return_format = c("upper_limit", "interval", "text"),
  ymin = c(-Inf),
  coverage_proba = NULL,
  coverage_alpha = NULL
)
```

Arguments

Q_pred	Vector of extreme quantile regression predictions for the test data (same length as the number of test points).
block_ids	Vector of block IDs for each test point, of the same length as Q_pred. The block IDs should be integers between 1 and the number of blocks, and correspond to the order of the dQ_blocks vector.
dQ_blocks	Vector of block-specific conformal corrections to be added to the extreme quantile regression predictions, of the same length as the number of blocks.
return_format	Format of the returned prediction interval. Either 'upper_limit' for a numerical vector of upper PI limits (default), 'interval' for a data frame with columns of lower and upper PI limits, or 'text' for a single string description of the PI (only available for single predictions).

`ymin` Lower endpoint for the response distribution (if known). Can be a single value (marginal lower endpoint), or a vector of the same length as `Q_pred` (conditional lower endpoint), or a vector of the same length as `Q_pred` (block-conditional lower endpoint). Default is `-Inf`.

`coverage_proba, coverage_alpha` (Optional) Marginal coverage probability (or level alpha) for the conformal prediction interval. Only one of `coverage_proba` or `coverage_alpha` must be provided, as `coverage_alpha = 1 - coverage_proba`. Only used for certain `return_format` options.

Value

Depending on the `return_format` argument, either a numerical vector of upper prediction interval (PI) limits, a data frame with lower and upper PI limits, and the block IDs, as columns (and optionally coverage probability and alpha), or (only available for compatibility) a single string description of the PI (for single predictions only).

`conformal_PI_unilat` *Single-sided conformal prediction interval from conformalizer*

Description

Single-sided conformal prediction interval from conformalizer

Usage

```
conformal_PI_unilat(
  Q_pred,
  unilat_conformalizer,
  return_format = c("upper_limit", "interval", "text"),
  ymin = c(-Inf),
  coverage_proba = NULL,
  coverage_alpha = NULL
)
```

Arguments

`Q_pred` Vector of extreme quantile regression predictions for the test data (same length as the number of test points).

`unilat_conformalizer` Either a conformalizer object obtained from `conformalize_EQR_unilat()`, or, directly, the conformal correction to be added to the extreme quantile regression predictions. In the latter case, either a single conformal correction value or a vector of conformal corrections of the same length as `Q_pred` is expected.

`return_format` Format of the returned prediction interval. Either `'upper_limit'` for a numerical vector of upper PI limits (default), `'interval'` for a data frame with columns of lower and upper PI limits, or `'text'` for a single string description of the PI (only available for single predictions).

`ymin` Lower endpoint for the response distribution (if known). Can be a single value (marginal lower endpoint) or a vector of the same length as `Q_pred` (conditional lower endpoint). Default is `-Inf`.

`coverage_proba, coverage_alpha` (Optional) Marginal coverage probability (or level alpha) for the conformal prediction interval. Only one of `coverage_proba` or `coverage_alpha` must be provided, as `coverage_alpha = 1 - coverage_proba`. Only used for certain `return_format` options.

Value

Depending on the `return_format` argument, either a numerical vector of upper prediction interval (PI) limits, a data frame with lower and upper PI limits as columns (and optionally coverage probability and alpha), or a single string description of the PI (for single predictions only).

Examples

```
conformal_PI_unilat(Q_pred=c(10,12), unilat_conformalizer=0.3, return_format='upper_limit')
```

```
conformalize_EQR_unilat
```

Single-sided extreme conformal prediction

Description

The main function of this package, performing single-sided extreme conformal prediction from extreme quantile regression predictions. This function computes the conformal correction score-quantile \hat{q}_α^c (or \hat{q}_α) to be added to the extreme quantile regression prediction(s) to obtain the desired PIs. It can optionally perform weighted conformal prediction for nonexchangeable data (e.g. under distribution shift or drift), by providing calibration-sample weights.

Usage

```
conformalize_EQR_unilat(
  EQR_pred,
  y_calibr,
  coverage_proba = NULL,
  coverage_alpha = NULL,
  method = c("GPD_safeprofile", "GPD_boot", "GPD_profile", "Empirical",
    "GPD_profile_naive", "GPD_delta", "GPD_max", "GPD_simple", "Hill"),
  threshold_lvls = NULL,
  alpha_correction = c("Sidak", "Bonferroni"),
  correction_prop = 0.5,
  R = 1004,
  min_obs_GPD = 10,
  profile_init_step_pos = 100,
  profile_init_step_neg = 10,
```

```

profile_tol = 0.001,
profile_steps_beyond_conf = 5,
profile_max_steps = 1000,
alpha_profile_naive = 0.01,
obs_weights = NULL,
test_weight = NULL,
return_q_list = FALSE,
verbose = 1,
.pre_sorted_scores = NULL
)

```

Arguments

EQR_pred	Vector of extreme quantile predictions for the calibration data, ideally at level coverage_proba (same length as y_calibr).
y_calibr	Vector of observed response values of the calibration data (same length as EQR_pred).
coverage_proba, coverage_alpha	Marginal coverage probability (or level alpha) for the conformal prediction interval. Only one of coverage_proba or coverage_alpha must be provided, as coverage_alpha = 1 - coverage_proba.
method	Method for constructing the conformal prediction interval. Defaults to 'GPD_safeprofile'. See the Details section for more information.
threshold_lvls	Threshold probability level for the GPD-based methods. The 'GPD_max' model instead requires a vector of threshold levels. Defaults to 0.95 (or to a sequence from 0.8 to 0.99 with the 'GPD_max' method).
alpha_correction	Confidence correction method for the CI-based extreme conformal prediction intervals used to choose α_1 and α_2 . Defaults to 'Sidak'.
correction_prop	Relative (pseudo-)proportion between the values of α_1 and α_2 , for the CI-based extreme conformal prediction intervals. Defaults to 0.5 (equal values of α_1 and α_2). Larger values increase α_1 and decrease α_2 .
R	Number of bootstrap replicates for the 'GPD_boot' method.
min_obs_GPD	Minimum number of observations above the threshold for the GPD-based methods.
profile_init_step_pos, profile_init_step_neg	Initial binary-search step size, in the positive and negative directions, for the profile-likelihood methods. See ExtremeCI::GPD_profile_CI() for more details.
profile_tol	Tolerance for the profile-likelihood search. See ExtremeCI::GPD_profile_CI() for more details.
profile_steps_beyond_conf	Number of safety steps beyond the confidence line for the profile-likelihood search. See ExtremeCI::GPD_profile_CI() for more details.
profile_max_steps	Maximum number of initial search steps for the profile-likelihood CIs. See ExtremeCI::GPD_profile_CI() for more details.

<code>alpha_profile_naive</code>	CI confidence level (i.e., the equivalent of α_2) for the 'GPD_profile_naive' method.
<code>obs_weights</code>	Optional vector of sample weights (same length as <code>y_calibr</code>), to perform weighted conformal prediction for nonexchangeable data (e.g. under distribution shift or drift).
<code>test_weight</code>	Weight of the test point for which the weighted conformal prediction is performed. Only necessary for the 'Empirical' method if <code>obs_weights</code> is provided (defaults to <code>max(obs_weights)</code>).
<code>return_q_list</code>	Boolean indicating whether to return a debug quantile list.
<code>verbose</code>	Verbose level (0 for no messages, 1 for warnings, 2 for warnings duplicated as <code>cat()</code> prints).
<code>.pre_sorted_scores</code>	(For development only.) If already computed: Vector of pre-sorted calibration nonconformity scores.

Details

The method argument specifies the conformalization method used to construct the (extreme) conformal prediction intervals (PIs). The available method options are:

- '**GPD_safeprofile**' Recommended choice for extreme conformal prediction. Tries the method 'GPD_profile' first, and falls back to 'GPD_boot' if the former suffers from numerical instability.
- '**GPD_profile**' GPD-based extreme conformalization using the profile-likelihood CI endpoint of the extreme score quantile. It captures the asymmetric uncertainty of the score-quantile best, yielding the most reliable coverage. It might sometimes overcover or suffer from numerical convergence issues.
- '**GPD_boot**' GPD-based extreme conformalization using the nonparametric bootstrap percentile CI endpoint of the extreme score quantile.
- '**GPD_delta**' GPD-based extreme conformalization using the Delta method CI endpoint of the extreme score quantile.
- '**Empirical**' The classical (non-extreme) conformalized quantile regression method, relying on the empirical quantile of the nonconformity scores. Yields infinitely wide PIs if the coverage_prob is larger than $1-1/(\text{length}(y_calibr)+1)$.
- '**GPD_profile_naive**' A naive version of 'GPD_profile', without the `alpha_correction` for multiple testing. Is likely to undercover, only use for comparison.
- '**GPD_max**' A naive GPD-based approach repeating the 'GPD_simple' method for a range of threshold levels, keeping the most conservative results. Is likely to undercover, only use for comparison.
- '**GPD_simple**' A naive GPD-based extreme conformalization using a simple extrapolated score quantile estimate instead of a CI endpoint. Is likely to undercover, only use for comparison.
- '**Hill**' (Not implemented) Extreme conformalization based on the Hill estimator from extreme value analysis.

See Pasche et al. (2026), referenced below, for the technical details of extreme conformal prediction.

Value

A named list containing the following elements.

dQ	The conformal correction \hat{q}_α^e or \hat{q}_α to be added to the extreme quantile prediction(s) to obtain the conformal PI endpoint.
coverage_proba	The marginal coverage probability for the conformal PI.
coverage_alpha	The marginal coverage alpha level for the conformal PI.
threshold_lvls	The threshold level(s) effectively used for the GPD-based methods.
method	The method used to obtain the conformal PI.
dQ_thresholds	A debugging vector of quantiles at the specified threshold levels. Only for GPD-based methods when <code>return_q_list==TRUE</code> .
sigma	The GPD scale parameter estimate. Currently only for method 'GPD_simple', when <code>return_q_list==TRUE</code> .
xi	The GPD shape parameter estimate. Currently only for method 'GPD_simple', when <code>return_q_list==TRUE</code> .

References

Pasche, O. C., Lam, H., and Engelke, S. (2026). "Extreme Conformal Prediction: Reliable Intervals for High-Impact Events." *Extremes* 29(2), 129–155. doi:[10.1007/s10687026005369](https://doi.org/10.1007/s10687026005369).

Index

`block_weighted_conformal_PI_unilat`, 2

`cat()`, 6

`conformal_PI_unilat`, 3

`conformalize_EQR_unilat`, 4

`conformalize_EQR_unilat()`, 3

`ExtremeCI::GPD_profile_CI()`, 5