

# Package: xpect (via r-universe)

March 25, 2025

**Type** Package

**Title** Probabilistic Time Series Forecasting with XGBoost and Conformal Inference

**Version** 1.0

**Maintainer** Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

**Description** Implements a probabilistic approach to time series forecasting combining XGBoost regression with conformal inference methods. The package provides functionality for generating predictive distributions, evaluating uncertainty, and optimizing hyperparameters using Bayesian, coarse-to-fine, or random search strategies.

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Imports** normalp (>= 0.7.2), glogis (>= 1.0-2), gld (>= 2.6.6), edfun (>= 0.2.0), purrr (>= 1.0.1), ald (>= 1.3.1), evd (>= 2.3-6.1), GeneralizedHyperbolic (>= 0.8-6), cubature (>= 2.1.0), furrr (>= 0.3.1), future (>= 1.33.0), xgboost (>= 1.7.5.1), rBayesianOptimization (>= 1.2.0), lubridate (>= 1.9.2), ggplot2 (>= 3.5.1), scales (>= 1.3.0)

**URL** [https://rpubs.com/giancarlo\\_vercellino/xpect](https://rpubs.com/giancarlo_vercellino/xpect)

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Depends** R (>= 2.10)

**NeedsCompilation** no

**Author** Giancarlo Vercellino [aut, cre, cph]

**Date/Publication** 2025-03-24 11:30:01 UTC

**Config/pak/sysreqs** make

**Repository** <https://pigan.r-universe.dev>

**RemoteUrl** <https://github.com/cran/xpect>

**RemoteRef** HEAD

**RemoteSha** 2d7086e0bddb2c11328644ef890e5f1d699132f8

## Contents

xpect . . . . .	2
<b>Index</b>	<b>6</b>

---

xpect	<i>xpect</i>
-------	--------------

---

## Description

This function implements probabilistic time series forecasting by combining gradient-boosted regression (XGBoost) with conformal inference techniques. It produces predictive distributions capturing uncertainty and optimizes hyper parameters through Bayesian, coarse-to-fine, or random search methods. The approach leverages historical observations from predictor series to estimate the future values of a specified target series. Users can customize the forecasting model extensively by setting parameters for model complexity, regularization, and conformal calibration.

Implements a probabilistic approach to time series forecasting combining XGBoost regression with conformal inference methods. The package provides functionality for generating predictive distributions, evaluating uncertainty, and optimizing hyper parameters using Bayesian, coarse-to-fine, or random search strategies.

## Usage

```
xpect(
  predictors,
  target,
  future,
  past = 1L,
  coverage = 0.5,
  max_depth = 3L,
  eta = 0.1,
  gamma = 0,
  alpha = 0,
  lambda = 1,
  subsample = 0.8,
  colsample_bytree = 0.8,
  search = "none",
  calib_rate = 0.5,
  n_sim = 1000,
  nrounds = 200,
  n_samples = 10,
  n_exploration = 10,
  n_phases = 3,
```

```

    top_k = 3,
    seed = 42
)

```

### Arguments

predictors	A data frame containing multiple time series predictors and the target series to forecast.
target	Character string specifying the name of the target series to forecast within the predictors dataset.
future	Integer specifying the number of future time steps to forecast.
past	Integer or numeric vector specifying past observations used as input features. Single value sets fixed value (default: 1). NULL sets standard range (1L-30L), while two values define custom range.
coverage	Numeric or numeric vector for fraction of total variance preserved during SVD. Single value sets fixed value (default: 0.5). NULL sets standard range (0.05-0.95), while two values define custom range.
max_depth	Integer or numeric vector for max depth of XGBoost trees. Single value sets fixed value (default: 3). NULL sets standard range (3L-10L), while two values define custom range.
eta	Numeric or numeric vector for learning rate in XGBoost. Single value sets fixed value (default: 0.1). NULL sets standard range (0.01-0.3), while two values define custom range.
gamma	Numeric or numeric vector for minimum loss reduction to split a leaf node. Single value sets fixed value (default: 0). NULL sets standard range (0-5), while two values define custom range.
alpha	Numeric or numeric vector for L1 regularization strength. Single value sets fixed value (default: 0). NULL sets standard range (0-1), while two values define custom range.
lambda	Numeric or numeric vector for L2 regularization strength. Single value sets fixed value (default: 1). NULL sets standard range (0-1), while two values define custom range.
subsample	Numeric or numeric vector (0-1) for instance subsampling ratio per tree. Single value sets fixed value (default: 0.8). NULL sets standard range (0-1), while two values define custom range.
colsample_bytree	Numeric or numeric vector (0-1) for column subsampling ratio per tree. Single value sets fixed value (default: 0.8). NULL sets standard range (0-1), while two values define custom range.
search	Character string specifying the hyper parameter search method to employ. Options include: "none" (default), "random_search", "bayesian", "coarse_to_fine".
calib_rate	Numeric fraction (default: 0.5) of observations allocated for conformal calibration, influencing the uncertainty estimation.
n_sim	Integer (default: 1000) determining the number of simulated calibration error samples used during conformal inference.

nrounds	Integer (default: 200) specifying the maximum number of boosting iterations allowed during model training.
n_samples	Integer specifying the number of parameter configurations evaluated during random search or initial Bayesian sampling.
n_exploration	Integer specifying the number of exploratory evaluations during Bayesian optimization to balance exploration-exploitation.
n_phases	Integer specifying how many iterative refinement phases are performed in coarse-to-fine optimization.
top_k	Integer (default: 3) indicating how many top-performing parameter configurations are retained in each coarse-to-fine optimization iteration.
seed	Integer setting the random seed for reproducibility.

### Value

A list containing:

**history** A data frame logging each evaluated hyperparameter configuration and its associated cross-entropy performance against the selected benchmark.

**best\_model** The optimal forecasting model, including probability density functions (pdf), cumulative distribution functions (cdf), inverse cumulative distribution functions (icdf), and random sampling functions (sampler) for each point in the forecasted horizon.

**best\_params** A named vector detailing the selected hyper parameters of the best-performing forecasting model.

**plot** A visualization displaying the optimal forecasts alongside confidence bands derived from conformal intervals, facilitating intuitive uncertainty interpretation.

**time\_log** Duration tracking the computational time required for the complete optimization and model-building process.

### Author(s)

Giancarlo Vercellino <giancarlo.vercellino@gmail.com>

**Maintainer:** Giancarlo Vercellino <giancarlo.vercellino@gmail.com> [copyright holder]

### See Also

Useful links:

- [https://rpubs.com/giancarlo\\_vercellino/xpect](https://rpubs.com/giancarlo_vercellino/xpect)

### Examples

```
dummy_data <- data.frame(target_series = cumsum(rnorm(100)), predictor1 = cumsum(rnorm(100)))

result <- xpect(predictors = dummy_data,
                target = "target_series",
                future = 3,
                past = c(5L, 20L),#CUSTOM RANGE
```

```
coverage = 0.9,  
max_depth = c(3L, 8L),#CUSTOM RANGE  
eta = c(0.01, 0.05),  
gamma = NULL,#STANDARD RANGE  
alpha = NULL,#STANDARD RANGE  
lambda = NULL,#STANDARD RANGE  
subsample = 0.8,  
colsample_bytree = 0.8,  
search = "random_search",  
n_samples = 3,  
seed = 123)
```

# Index

xpect, [2](#)

xpect-package (xpect), [2](#)