

# Package: qrensemble (via r-universe)

November 6, 2024

**Title** Forecast ensembles using Quantile Regression Average (QRA)

**Version** 0.1.3

**Description** Performs quantile regression average

**Depends** R (>= 4.1.0)

**Imports** checkmate, data.table, purrr, quantgen, scoringutils

**Suggests** knitr, rmarkdown, roxyglobals, testthat (>= 3.0.0)

**Remotes** ryantibs/quantgen/quantgen

**License** MIT + file LICENSE.md

**Encoding** UTF-8

**LazyData** true

**Roxygen** list(markdown = TRUE, roclets = c("`collate", "`namespace",  
`rd", "`roxyglobals::global\_roclet"))

**RoxygenNote** 7.3.2

**VignetteBuilder** knitr

**Config/roxyglobals/filename** globals.R

**Config/roxyglobals/unique** FALSE

**Config/testthat/edition** 3

**Config/pak/sysreqs** libglpk-dev

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

**RemoteUrl** <https://github.com/epiforecasts/qrensemble>

**RemoteRef** v0.1.3

**RemoteSha** ee18f01f0c2c4e3741059356a190fe7ae36c30e1

## Contents

gra . . . . .	2
<b>Index</b>	<b>4</b>

---

gra	<i>Quantile Regression Average Calculates a quantile regression average for forecasts.</i>
-----	--

---

## Description

Quantile Regression Average Calculates a quantile regression average for forecasts.

## Usage

```
gra(
  forecast,
  target,
  group = c(),
  model = "Quantile Regression Average",
  per_quantile_weights = FALSE,
  enforce_normalisation = TRUE,
  intercept = FALSE,
  noncross = TRUE,
  ...
)
```

## Arguments

forecast	a data.table representing forecast; this is expected to have been created using <a href="#">scoringutils::as_forecast_quantile()</a>
target	the target for which to create the quantile regression average. This should be given as a vector of form <code>column = target</code> , where <code>target</code> is the value of <code>column</code> that represents the target. Note that the column named here cannot be a grouping variable.
group	any columns which to group a vector of character vectors (e.g., "horizon", "geography_scale", etc.) indicating columns in the forecasts and data data frames; by default, will not group anything, i.e. create one ensemble model
model	the name of the model to return; default: "Quantile Regression Average"
per_quantile_weights	logical; whether to estimate weights per quantile
enforce_normalisation	logical; whether to enforce quantiles
intercept	logical; whether to estimate and intercept
noncross	logical; whether to enforce non-crossing of quantiles
...	passed to <a href="#">quantgen::predict.quantile_ensemble()</a> ; of particular interest might be setting <code>iso = TRUE</code> for isotonic regression

**Value**

a data.table representing the forecasts forecast, but with model set to the value of the 'model parameter. This will be in the forecast format produced by [scoringutils::as\\_forecast\\_quantile\(\)](#)

**Examples**

```
library("scoringutils")
example_quantile |>
  as_forecast_quantile() |>
  qra(
    group = c("target_type", "location", "location_name"),
    target = c(target_end_date = "2021-07-24")
  )
```

# Index

qra, [2](#)  
quantgen::predict.quantile\_ensemble(),  
[2](#)  
scoringutils::as\_forecast\_quantile(),  
[2](#), [3](#)