

Package: RQK (via r-universe)

October 24, 2026

Type Package

Title Rational Quadratic Kriging

Version 1.0.0

Description Fits the rational quadratic kriging models of Song and Joseph (2026) <[doi:10.48550/arXiv.2610.18077](https://doi.org/10.48550/arXiv.2610.18077)> for deterministic and noisy responses using a rational quadratic kernel with anisotropic factor weights. Provides leave-one-out tuning, fast 'C++' prediction routines, and conformal prediction intervals. This work is supported by a U.S. NSF grant DMS-2310637.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.3.0)

Imports Rcpp (>= 1.0.14), nloptr

LinkingTo Rcpp, RcppArmadillo, nloptr

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

NeedsCompilation yes

Config/roxygen2/version 8.1.0

Author Difan Song [aut, cre], V. Roshan Joseph [aut]

Maintainer Difan Song <dfsong96@gmail.com>

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

Date/Publication 2026-10-24 11:17:32 UTC

RemoteUrl <https://github.com/cran/RQK>

RemoteRef HEAD

RemoteSha a81ff9d2569fd32c70c28fea9525482fa3ededa1

Contents

initial_lengthscale	2
predict.RQK	3
RQK	5

Index**8**

initial_lengthscales *Obtain an initial length scale and factor weights*

Description

Computes a starting point for RQK fitting, including the kernel length scale and the anisotropic factor weights over the p input coordinates.

Usage

```
initial_lengthscales(
  X,
  y,
  theta = NULL,
  theta_fit = TRUE,
  noise = FALSE,
  q = NULL
)
```

Arguments

X	Numeric input matrix with n rows and p columns.
y	Numeric response vector of length n .
θ	Optional positive kernel length scale. Must be supplied when $\theta_{\text{fit}} = \text{FALSE}$.
θ_{fit}	Logical; if TRUE, optimize θ while constructing the initial point.
noise	Logical; if TRUE, use the noisy-case default for q .
q	Optional positive shape parameter of the rational quadratic kernel. If NULL, the implementation uses $(p + 1) / 2$ for noiseless fits and $p / 2$ for noisy fits, where $p = \text{ncol}(X)$.

Value

A list with elements:

- θ : initial kernel length scale on the original scale.
- $\log\theta$: the same value on the log scale.
- q : kernel shape parameter used for the initialization.
- u : unconstrained optimization variables for the factor weights.
- w : factor weights on the simplex.
- par : full optimization vector returned by the initializer.
- $\log\theta_{\text{lb}}$, $\log\theta_{\text{ub}}$: log-scale bounds used for θ .

References

Song, D., and Joseph, V. R. (2026). Rational Quadratic Kriging. arXiv preprint arXiv:2610.18077. [doi:10.48550/arXiv.2610.18077](https://doi.org/10.48550/arXiv.2610.18077)

Examples

```
borehole <- function(xx) {
  lower <- c(0.05, 100, 63070, 990, 63.1, 700, 1120, 9855)
  upper <- c(0.15, 50000, 115600, 1110, 116, 820, 1680, 12045)
  xx <- xx * (upper - lower) + lower
  rw <- xx[1]
  r <- xx[2]
  Tu <- xx[3]
  Hu <- xx[4]
  Tl <- xx[5]
  Hl <- xx[6]
  L <- xx[7]
  Kw <- xx[8]
  frac1 <- 2 * pi * Tu * (Hu - Hl)
  frac2a <- 2 * L * Tu / (log(r / rw) * rw^2 * Kw)
  frac2b <- Tu / Tl
  frac2 <- log(r / rw) * (1 + frac2a + frac2b)
  frac1 / frac2
}

set.seed(1)
X_train <- matrix(runif(200 * 8), ncol = 8)
y_train <- apply(X_train, 1, borehole)

init <- initial_lengthscales(X_train, y_train, noise = FALSE)
init$theta
head(init$w)
```

predict.RQK

Predict from a fitted RQK model

Description

Computes point predictions and, unless disabled, conformal prediction intervals from an object returned by `RQK()`.

Usage

```
## S3 method for class 'RQK'
predict(object, newdata, alpha = 0.05, interval_type = 4, ...)
```

Arguments

object	An object returned by <code>RQK()</code> .
newdata	Numeric matrix or data frame of new inputs with p columns.
alpha	Miscoverage tuning parameter in $(0, 1)$.
interval_type	Integer in $1:4$ selecting the construction of the prediction interval, or "none" to return point predictions without interval components. 1 is the smooth symmetric score-quantile interval. 2 is the conformal set obtained by taking the union of possibly disjoint intervals. 3 is the convex hull of type 2. 4 is a conservative interval guaranteed to be a superset of type 3.
...	Ignored. Included for generic compatibility.

Details

If object was fitted with `scale = TRUE`, newdata is scaled to $[0, 1]$ using the training minima and maxima saved in the fitted object before predictions are computed.

Value

A list with:

- mean: point predictions. If `interval_type = "none"`, this is the only returned component. Otherwise the list also contains:
- lower, upper: numeric prediction bounds for interval types 1, 3, and 4. For type 2 they are `NA_real_` because the prediction set may be disjoint.
- intervals: a list with one element per prediction point. Each element is a two-column matrix with columns lower and upper. Type 2 may contain multiple rows, whereas types 1, 3, and 4 contain a single row.

References

Song, D., and Joseph, V. R. (2026). Rational Quadratic Kriging. arXiv preprint arXiv:2610.18077. [doi:10.48550/arXiv.2610.18077](https://doi.org/10.48550/arXiv.2610.18077)

Examples

```
borehole <- function(xx) {
  lower <- c(0.05, 100, 63070, 990, 63.1, 700, 1120, 9855)
  upper <- c(0.15, 50000, 115600, 1110, 116, 820, 1680, 12045)
  xx <- xx * (upper - lower) + lower
  rw <- xx[1]
  r <- xx[2]
  Tu <- xx[3]
  Hu <- xx[4]
  Tl <- xx[5]
  Hl <- xx[6]
  L <- xx[7]
  Kw <- xx[8]
  frac1 <- 2 * pi * Tu * (Hu - Hl)
```

```

    frac2a <- 2 * L * Tu / (log(r / rw) * rw^2 * Kw)
    frac2b <- Tu / Tl
    frac2 <- log(r / rw) * (1 + frac2a + frac2b)
    frac1 / frac2
  }

  set.seed(1)
  X_train <- matrix(runif(200 * 8), ncol = 8)
  y_train <- apply(X_train, 1, borehole)
  fit <- RQK(X_train, y_train, theta = 0.2, theta_fit = FALSE)

  X_test <- matrix(runif(1000 * 8), ncol = 8)
  pred <- predict(fit, X_test, alpha = 0.1)
  head(pred$mean)

```

RQK

Fit a rational quadratic kriging model

Description

Fits the rational quadratic kriging (RQK) model using compiled C++ routines for factor-weight estimation, leave-one-out tuning, and conformal prediction.

The implementation is based on **RcppArmadillo**. It is recommended to use a high performance BLAS library for optimized speed.

Usage

```

RQK(
  X,
  y,
  theta = NULL,
  theta_fit = TRUE,
  noise = FALSE,
  nugget = NULL,
  q = NULL,
  theta_bound = NULL,
  scale = FALSE
)

```

Arguments

<code>X</code>	Numeric input matrix with <code>n</code> rows and <code>p</code> columns.
<code>y</code>	Numeric response vector of length <code>n</code> .
<code>theta</code>	Optional positive kernel length scale. Must be supplied when <code>theta_fit = FALSE</code> . The supplied value is used as the initial value when <code>theta_fit = TRUE</code> .
<code>theta_fit</code>	Logical; if <code>TRUE</code> , optimize <code>theta</code> by leave-one-out loss.

noise	Logical; if TRUE, fit the noisy RQK variant with an estimated nugget term.
nugget	Optional nugget term. If supplied, it is kept fixed in both noiseless and noisy fits. If NULL, noiseless fits use 1e-8, whereas noisy fits estimate the nugget by line search and local optimization.
q	Optional positive shape parameter of the rational quadratic kernel. If NULL, the implementation uses $(p + 1) / 2$ for noiseless fits and $p / 2$ for noisy fits, where $p = \text{ncol}(X)$.
theta_bound	Optional numeric vector <code>c(theta_lb, theta_ub)</code> giving positive lower and upper bounds for theta.
scale	Logical; if TRUE, each input variable is scaled to $[0, 1]$ before fitting. The training minima and maxima are saved in the returned object and reused by <code>predict.RQK()</code> . This is recommended when the input data are not already scaled to $[0, 1]$.

Details

The model uses a rational quadratic kernel on a p -dimensional input space:

$$R(\mathbf{x}_i, \mathbf{x}_j) = \left(1 + \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_w^2}{\theta^2} \right)^{-q} + \text{nugget} \mathbb{I}\{\mathbf{x}_i = \mathbf{x}_j\}$$

and estimates anisotropic factor weights over the p coordinates. The returned fit stores both user-facing quantities such as `theta`, `q`, and `w`, and cached matrices used by `predict.RQK()` to evaluate point predictions and conformal prediction intervals. When `scale = TRUE`, fitting is performed on the scaled inputs and the returned `X_train` cache is also on the scaled input scale.

Value

An object of class "RQK" with elements including:

- `w`: estimated nonnegative factor weights summing to one.
- `theta`: fitted kernel length scale.
- `nugget`: fixed or estimated nugget term.
- `q`: kernel shape parameter.
- `L00_pred`: leave-one-out point predictions for the training responses.
- `scores`: training conformity scores used by `predict.RQK()` to construct interval types 1–4.
- `X_train`, `y_train`: training inputs and responses.
- `scale`: logical flag indicating whether the inputs were scaled.
- `scale_min`, `scale_max`: training minima and maxima for each input variable, present when `scale = TRUE`.
- additional cached matrices used internally by `predict.RQK()`.

References

Song, D., and Joseph, V. R. (2026). Rational Quadratic Kriging. arXiv preprint arXiv:2610.18077. doi:10.48550/arXiv.2610.18077

Examples

```
borehole <- function(xx) {  
  lower <- c(0.05, 100, 63070, 990, 63.1, 700, 1120, 9855)  
  upper <- c(0.15, 50000, 115600, 1110, 116, 820, 1680, 12045)  
  xx <- xx * (upper - lower) + lower  
  rw <- xx[1]  
  r <- xx[2]  
  Tu <- xx[3]  
  Hu <- xx[4]  
  Tl <- xx[5]  
  Hl <- xx[6]  
  L <- xx[7]  
  Kw <- xx[8]  
  frac1 <- 2 * pi * Tu * (Hu - Hl)  
  frac2a <- 2 * L * Tu / (log(r / rw) * rw^2 * Kw)  
  frac2b <- Tu / Tl  
  frac2 <- log(r / rw) * (1 + frac2a + frac2b)  
  frac1 / frac2  
}  
  
set.seed(1)  
X_train <- matrix(runif(200 * 8), ncol = 8)  
y_train <- apply(X_train, 1, borehole)  
  
fit <- RQK(X_train, y_train, theta = 0.2, theta_fit = FALSE)  
fit$theta  
fit$q  
head(fit$w)  
  
X_test <- matrix(runif(1000 * 8), ncol = 8)  
pred <- predict(fit, X_test)  
head(data.frame(mean = pred$mean, lower = pred$lower, upper = pred$upper))
```

Index

`initial_lengthscale`, [2](#)

`predict.RQK`, [3](#)

`predict.RQK()`, [6](#)

`RQK`, [5](#)

`RQK()`, [3](#), [4](#)