Package 'BayesQRCount'

August 5, 2026

Type Package

Title Adaptive Bayesian Quantile Regression for Count Data

Version 0.1.0

Description Implements Bayesian quantile regression for count data using the jittering technique for discrete data smoothing and an asymmetric Laplace distribution likelihood. Supports adaptive variable selection via a random-bridge penalty with a beta prior on the power parameter, as well as fixed-bridge and Lasso penalties. Utilizes Markov chain Monte Carlo with Gibbs sampling and adaptive Metropolis-Hastings algorithms for posterior inference, provides Gelman-Rubin convergence diagnostics, and predicts conditional quantiles for count responses. Methodology and applications are based on the following key references: Luo, Zhou, Hu, and Li (2026, Journal of Mathematics, 2026:1543166, <doi:10.1155/jom/1543166>), Koenker and Bassett (1978, Econometrica, 46, 33-50, <doi:10.2307/1913643>), Machado and Santos Silva (2005, Journal of the American Statistical Association, 100, 1226-1237, <doi:10.1198/016214505000000330>), Yu and Moyeed (2001, Statistics and Probability Letters, 54, 437-447, <doi:10.1016/S0167-7152(01)00124-9>), Polson, Scott, and Windle (2014, Journal of the Royal Statistical Society Series B, 76, 713-733, <doi:10.1111/rssb.12042>), Park and Casella (2008, Journal of the American Statistical Association, 103, 681-686, <doi:10.1198/016214508000000337>), and Roberts and Rosenthal (2009, Journal of Computational and Graphical Statistics, 18, 349-367, <doi:10.1198/jcgs.2009.06134>).

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.0.0)

Imports stats, graphics, grDevices

Suggests testthat (>= 3.0.0)

NeedsCompilation no

Author Shikhar Tyagi [aut, cre] (ORCID:
<https://orcid.org/0000-0003-1606-0844>),
Arvind Pandey [aut],
Bhupendra Singh [aut],
Vrijesh Tripathi [aut]

Maintainer Shikhar Tyagi <shikhar1093tyagi@gmail.com>

Repository CRAN

Date/Publication 2026-08-05 09:30:08 UTC

Contents

`bqr_count` *Bayesian Quantile Regression for Count Data*

Description

Performs Bayesian quantile regression for count data using the jittering-based asymmetric Laplace distribution (ALD) likelihood with random-bridge, fixed-bridge, or Lasso penalty priors. Implements the methodology of Luo, Zhou, Hu, and Li (2026).

Usage

```
bqr_count(
  y,
  X,
  tau = 0.5,
  method = c("random_bridge", "fixed_bridge", "lasso"),
  xi_fixed = 0.5,
  N_iter = 20000,
  burn_in = 10000,
  thin = 10,
```

```
  n_chains = 3,
  lambda_prior = c(a = 1, b = 1),
  xi_prior = c(e = 1, f = 1),
  delta = 0.05,
  sigma0_sq = 0.01,
  seed = NULL
)
```

Arguments

y	Integer vector of non-negative count response data.
X	Numeric matrix of covariates (n x p). An intercept column is added internally; do not include one.
tau	Numeric, quantile level in (0, 1). Default is 0.5.
method	Character string specifying the penalty method: "random_bridge" (default), "fixed_bridge", or "lasso".
xi_fixed	Numeric, power parameter for fixed-bridge penalty (default 0.5). Used only when method = "fixed_bridge". For "lasso", xi is fixed at 1.
N_iter	Integer, total number of MCMC iterations (default 20000).
burn_in	Integer, number of burn-in iterations to discard (default 10000).
thin	Integer, thinning interval (default 10).
n_chains	Integer, number of MCMC chains to run (default 3).
lambda_prior	Numeric vector of length 2, hyperparameters (a, b) for lambda ~ Gamma(a, b). Default is c(a = 1, b = 1).
xi_prior	Numeric vector of length 2, hyperparameters (e, f) for xi ~ Beta(e, f). Default is c(e = 1, f = 1). Used only for random_bridge.
delta	Numeric, probability for fixed (non-adaptive) component in the Metropolis-Hastings mixture proposal for xi. Default is 0.05.
sigma0_sq	Numeric, fixed proposal variance for the non-adaptive M-H component. Default is 0.01.
seed	Integer or NULL, random seed for reproducibility.

Value

An object of class "bqr_count", a list containing:

beta_samples Array of posterior beta samples (n_saved x p x n_chains).

sigma_samples Matrix of posterior sigma samples (n_saved x n_chains).

lambda_samples Matrix of posterior lambda samples (n_saved x n_chains).

xi_samples Matrix of posterior xi samples (n_saved x n_chains).

summary Data frame with posterior mean, median, SD, and 95 percent HPD intervals for all parameters.

coefficients Named numeric vector of posterior mean coefficients.

fitted_quantiles Numeric vector of fitted quantile values.

variable_selection Logical vector indicating selected variables (HPD excludes 0).

tau The quantile level used.

method The penalty method used.

n_chains Number of chains.

y The response data.

X_original The original covariate matrix (without intercept).

X_design The full design matrix (with intercept).

call The matched function call.

Examples

```
set.seed(42)
n <- 100
X <- matrix(rnorm(n * 2), n, 2)
y <- rpois(n, exp(0.5 + 0.3 * X[, 1] - 0.2 * X[, 2]))
fit <- bqr_count(y, X, tau = 0.5, method = "random_bridge",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
print(fit)
```

`coef.bqr_count` *Extract coefficients from bqr_count*

Description

Extract coefficients from bqr_count

Usage

```
## S3 method for class 'bqr_count'
coef(object, ...)
```

Arguments

`object`	A `bqr_count` object.
`...`	Additional arguments.

Value

A numeric vector of coefficients.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
coef(fit)
```

`compare_models` *Compare Multiple Bayesian Quantile Regression Models*

Description

Compares two or more fitted `"bqr_count"` models by RMSE and MAE.

Usage

```
compare_models(...)
```

Arguments

`...`	Two or more objects of class `"bqr_count"`, optionally named.

Value

A data frame with columns: Model, Method, Tau, RMSE, MAE.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit1 <- bqr_count(y, X, tau = 0.5, method = "lasso",
                  N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
fit2 <- bqr_count(y, X, tau = 0.5, method = "random_bridge",
                  N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
compare_models(Lasso = fit1, RandomBridge = fit2)
```

densplot *Posterior Density Plots*

Description

Plots kernel density estimates of posterior distributions for selected parameters.

Usage

```
densplot(object, pars = NULL, ...)
```

Arguments

object	An object of class "bqr_count".
pars	Character vector of parameter names to plot. Default plots all beta coefficients.
...	Additional graphical parameters.

Value

Invisible NULL. Called for side effects (plotting).

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
densplot(fit)
```

fitted.bqr_count *Extract fitted values from bqr_count*

Description

Extract fitted values from bqr_count

Usage

```
## S3 method for class 'bqr_count'
fitted(object, ...)
```

Arguments

object	A bqr_count object.
...	Additional arguments.

Value

A numeric vector of fitted quantiles.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
fitted(fit)
```

---

gelman_rubin *Gelman-Rubin Convergence Diagnostic*

Description

Computes the potential scale reduction factor (R-hat) for all model parameters from a multi-chain Bayesian quantile regression fit.

Usage

```
gelman_rubin(object)
```

Arguments

object	An object of class "bqr_count" with n_chains >= 2.

Value

A data frame with columns Parameter and Rhat.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
```

```
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 2)
gelman_rubin(fit)
```

`plot.bqr_count` *Plot diagnostics for bqr_count*

Description

Plot diagnostics for bqr_count

Usage

```
## S3 method for class 'bqr_count'
plot(x, which = 1, ...)
```

Arguments

`x`	A `bqr_count` object.
`which`	Integer from 1 to 3 specifying the plot type: 1 = density, 2 = trace, 3 = residuals.
`...`	Additional arguments passed to specific plot functions.

Value

Invisible NULL.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
plot(fit, which = 3)
```

plot_effects *Plot coefficient effects across different quantiles*

Description

Plot coefficient effects across different quantiles

Usage

```
plot_effects(fits, var_names = NULL, ...)
```

Arguments

fits	A named list of bqr_count objects fitted at different tau values.
var_names	Optional character vector of variable names to plot.
...	Additional arguments passed to plot.

Value

Invisible NULL.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit1 <- bqr_count(y, X, tau = 0.25, method = "lasso",
                  N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
fit2 <- bqr_count(y, X, tau = 0.5, method = "lasso",
                  N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
fit3 <- bqr_count(y, X, tau = 0.75, method = "lasso",
                  N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
plot_effects(list(Q25 = fit1, Q50 = fit2, Q75 = fit3))
```

predict.bqr_count *Predict Method for Bayesian Quantile Regression Count Models*

Description

Computes predicted conditional quantiles for new or training count data using the fitted model posterior.

Usage

```
## S3 method for class 'bqr_count'
predict(
  object,
  newdata = NULL,
  type = c("quantile", "mean"),
  interval = c("none", "credible"),
  level = 0.95,
  ...
)
```

Arguments

object	An object of class "bqr_count".
newdata	Optional numeric matrix of new covariates (without intercept). If NULL, predictions are made for the training data.
type	Character, type of prediction: "quantile" (default) returns integer count quantile estimates; "mean" returns the posterior predictive mean on the exp scale.
interval	Character, "none" (default) or "credible" for credible intervals.
level	Numeric, credible interval level (default 0.95).
...	Additional arguments (currently unused).

Value

A numeric vector of predicted values (if interval = "none"), or a data frame with columns fit, lower, upper (if interval = "credible").

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
predict(fit)
```

---

print.bqr_count *Print method for bqr_count objects*

Description

Print method for bqr_count objects

Usage

```
## S3 method for class 'bqr_count'
print(x, ...)
```

Arguments

x	A bqr_count object.
...	Additional arguments passed to print.

Value

The original object invisibly.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
print(fit)
```

---

print.summary.bqr_count

Print method for summary.bqr_count

Description

Print method for summary.bqr_count

Usage

```
## S3 method for class 'summary.bqr_count'
print(x, ...)
```

Arguments

x	A summary.bqr_count object.
...	Additional arguments passed to print.

Value

The original object invisibly.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
s <- summary(fit)
print(s)
```

---

residuals.bqr_count *Extract residuals from bqr_count*

Description

Extract residuals from bqr_count

Usage

```
## S3 method for class 'bqr_count'
residuals(object, ...)
```

Arguments

object	A bqr_count object.
...	Additional arguments.

Value

A numeric vector of residuals.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
residuals(fit)
```

selection_metrics *Variable Selection Performance Metrics*

Description

Computes F1-score, precision, recall, MSE, MAE, and average bias given the true regression coefficients. Useful for simulation studies.

Usage

```
selection_metrics(object, true_beta, level = 0.95)
```

Arguments

object	An object of class "bqr_count".
true_beta	Numeric vector of true regression coefficients (excluding intercept). Must match the number of covariates.
level	Numeric, HPD interval level (default 0.95).

Value

A data frame with columns: F1, Precision, Recall, MSE, MAE, Bias.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
selection_metrics(fit, true_beta = c(0.8, 0))
```

sim_count_data *Simulated Count Data for Bayesian Quantile Regression*

Description

A simulated dataset of 200 observations with 5 covariates and a Poisson count response, designed for demonstrating Bayesian quantile regression with variable selection. The true model has three non-zero coefficients (x1, x2, x5) and two zero coefficients (x3, x4).

Usage

```
data(sim_count_data)
```

Format

A data frame with 200 rows and 6 columns:

y Integer, count response variable generated from Poisson distribution.

x1 Numeric, covariate with true coefficient 0.8.

x2 Numeric, covariate with true coefficient -0.5.

x3 Numeric, noise covariate with true coefficient 0.

x4 Numeric, noise covariate with true coefficient 0.

x5 Numeric, covariate with true coefficient 0.3.

Details

The data were generated using: `mu = exp(0.5 + 0.8*x1 - 0.5*x2 + 0*x3 + 0*x4 + 0.3*x5)`, `y ~ Poisson(mu)`, with covariates drawn independently from standard normal distributions. The true intercept is 0.5. The sparsity level is 0.4 (2 out of 5 covariates are zero).

Source

Simulated data for package demonstration.

Examples

```
data(sim_count_data)
head(sim_count_data)
table(sim_count_data$y)
```

---

summary.bqr_count *Summary method for bqr_count objects*

Description

Summary method for bqr_count objects

Usage

```
## S3 method for class 'bqr_count'
summary(object, ...)
```

Arguments

`object`	A `bqr_count` object.
`...`	Additional arguments.

Value

A `summary.bqr_count` object.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
summary(fit)
```

---

traceplot *Trace Plots for MCMC Chains*

Description

Produces trace plots of MCMC samples for selected parameters.

Usage

```
traceplot(object, pars = NULL, ...)
```

Arguments

object	An object of class "bqr_count".
pars	Character vector of parameter names to plot. Default plots all beta coefficients.
...	Additional graphical parameters passed to plot.

Value

Invisible NULL. Called for side effects (plotting).

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
traceplot(fit)
```

`variable_select` *Variable Selection from Bayesian Quantile Regression*

Description

Extracts variable selection results from a fitted model based on highest posterior density (HPD) intervals.

Usage

```
variable_select(object, level = 0.95)
```

Arguments

`object`	An object of class `"bqr_count"`.
`level`	Numeric, HPD interval level for selection (default 0.95).

Value

A list with components:

selected Integer vector of indices of selected covariates (relative to the original X matrix, excluding intercept).

selected_names Character vector of names of selected covariates.

coefficients Named numeric vector of posterior mean coefficients for selected covariates.

n_selected Integer, number of selected covariates.

Examples

```
set.seed(42)
n <- 80
x1 <- rnorm(n)
x2 <- rnorm(n)
X <- cbind(x1, x2)
y <- rpois(n, exp(0.5 + 0.8 * x1))
fit <- bqr_count(y, X, tau = 0.5, method = "lasso",
                 N_iter = 500, burn_in = 200, thin = 2, n_chains = 1)
variable_select(fit)
```

Index