

Package ‘R2D2ordinal’

July 21, 2025

Title Implements Pseudo-R2D2 Prior for Ordinal Regression

Version 1.0.1

Description Implements the pseudo-R2D2 prior for ordinal regression from the paper "Pseudo-R2D2 prior for high-dimensional ordinal regression" by Yanchenko (2025) <[doi:10.48550/arXiv.2502.17491](https://doi.org/10.48550/arXiv.2502.17491)>. In particular, it provides code to evaluate the probability distribution function for the cut-points, compute the log-likelihood, calculate the hyper-parameters for the global variance parameter, find the distribution of McFadden's coefficient-of-determination, and fit the model in 'rstan'. Please cite the paper if you use these codes.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.2

Biarch true

Depends R (>= 3.5.0)

Imports extraDistr (>= 1.10.0), GIGrvg (>= 0.8), LaplacesDemon (>= 16.1.6), methods, Rcpp (>= 0.12.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), rstantools (>= 2.4.0)

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), StanHeaders (>= 2.18.0)

SystemRequirements GNU make

Suggests knitr, rmarkdown, ggplot2, dplyr

VignetteBuilder knitr

NeedsCompilation yes

Author Eric Yanchenko [aut, cre]

Maintainer Eric Yanchenko <eyanchenko@aiu.ac.jp>

Repository CRAN

Date/Publication 2025-03-18 11:20:09 UTC

Contents

dcut	2
find_param	3
llike	4
ord_r2d2	4
r2_mc	6
Index	7

dcut	<i>PDF of cut-points</i>
------	--------------------------

Description

This function computes the value of the probability density function for the cut-points. The distribution is induced by a Dirichlet distribution on the prior probabilities of the response.

Usage

```
dcut(tau, W, alpha, log = FALSE)
```

Arguments

tau	cut-points
W	global variance
alpha	concentration parameters for prior probabilities of Y
log	logical; if TRUE, returns log pdf

Value

value of pdf at tau

Examples

```
tau = c(-1,1) # set cut points
W = 1 # set value of global variance
alpha = c(1,1,1) #concentration parameters
dcut(tau, W, alpha, log=FALSE)
```

`find_param`*Find optimal GIG parameters for W prior*

Description

This function finds the optimal GIG parameters for the prior on W which induces a beta prior distribution on McFadden's R^2 .

Usage

```
find_param(  
  a,  
  b,  
  n,  
  K,  
  alpha = rep(1, K),  
  nsims = 1000,  
  nreps = 5,  
  no_cores = 10  
)
```

Arguments

<code>a</code>	hyper-parameter of prior for $R^2 \sim \text{Beta}(a,b)$
<code>b</code>	hyper-parameter of prior for $R^2 \sim \text{Beta}(a,b)$
<code>n</code>	number of observations
<code>K</code>	number of response categories
<code>alpha</code>	prior hyper-parameters for prior Dirichlet distribution on response probabilities
<code>nsims</code>	number of times to simulate data
<code>nreps</code>	number of times to run the algorithm (default = 5)
<code>no_cores</code>	number of cores to parallelize data-generation process

Value

Optimal GIG parameters

Examples

```
a = 1  
b = 5  
n = 100  
K = 3  
find_param(a, b, n, K, no_cores=1)
```

llike	<i>Log-Likelihood for ordinal regression</i>
-------	--

Description

This function evaluates the log-likelihood of the response for a given value of the parameters.

Usage

```
llike(Y, W, tau)
```

Arguments

Y	ordinal response
W	global variance
tau	cut-points

Value

value of log-likelihood at Y, W and tau

Examples

```
set.seed(1234)
K = 3 # number of response categories
Y = sample(1:K, 10, replace=TRUE) # generate responses
W = 1
tau = c(-1, 1) # set parameter values
llike(Y, W, tau)
```

ord_r2d2	<i>Ordinal regression in Stan with R2D2 prior</i>
----------	---

Description

This function carries out a Bayesian ordinal regression model in Stan using the proposed pseudo-R2D2 prior

Usage

```
ord_r2d2(
  x,
  y,
  K,
  a = 1,
  b = 10,
  hyper = NULL,
  alpha = rep(1, K),
  nsims = 1000,
  nreps = 5,
  no_cores = 10,
  progress = FALSE,
  ...
)
```

Arguments

x	covariate matrix
y	response variables
K	number of response categories
a	hyper-parameter of prior for $R2 \sim \text{Beta}(a,b)$
b	hyper-parameter of prior for $R2 \sim \text{Beta}(a,b)$
hyper	hyper-parameters for W prior
alpha	prior hyper-parameters for prior Dirichlet distribution on response probabilities
nsims	number of times to simulate data
nreps	number of times to run the algorithm (default = 5)
no_cores	number of cores to parallelize data-generation process
progress	logical. if TRUE, shows the progress bars from the posterior sampling.
...	optional hyper-parameters for Stan fitting

Value

Stan model fit

Examples

```
# X are covariates, Y are responses, K is number of response categories
# This example will yield low R2 values as the response are independent of the covariates.
set.seed(1234)
n = 100
p = 5
X = matrix(rnorm(n*p), nrow = n, ncol=p)
K = 3
Y = sample(1:K, 100, replace=TRUE)
a = 1
```

```

b = 5
# Pre-computed hyperparameters
fit <- ord_r2d2(X, Y, K, hyper=c(0.002, 0.989, 1.013), no_cores=1)
out <- rstan::extract(fit)
# Plot histogram of posterior W
hist(out$W, xlab="W")

```

r2_mc

Posterior distribution of McFadden's R2

Description

This function finds the posterior distribution of McFadden's R2 given the posterior samples from a Stan model fit

Usage

```
r2_mc(Y, out)
```

Arguments

Y	ordinal response
out	posterior samples from R2D2 model fit in Stan

Value

Posterior samples from McFadden's R2

Examples

```

# Obtain output from ord_r2d2() model fit
set.seed(1234)
# X are covariates, Y are responses, K is number of response categories
# This example will yield low R2 values as the response are independent of the covariates.
n = 100
p = 5
X = matrix(rnorm(n*p), nrow = n, ncol=p)
K = 3
Y = sample(1:K, 100, replace=TRUE)
a = 1
b = 5
# Pre-computed hyperparameters
fit <- ord_r2d2(X, Y, K, hyper=c(0.002, 0.989, 1.013), no_cores=1)
out <- rstan::extract(fit)
# Plot histogram of posterior R2
hist(r2_mc(Y, out), xlab="R2")

```

Index

`dcut`, 2

`find_param`, 3

`llike`, 4

`ord_r2d2`, 4

`r2_mc`, 6