

# Package ‘sdPrior’

October 14, 2022

**Title** Scale-Dependent Hyperpriors in Structured Additive  
Distributional Regression

**Version** 1.0-0

**Maintainer** Nadja Klein <nadja.klein@hu-berlin.de>

**Description** Utility functions for scale-dependent and alternative hyperpriors. The distribution parameters may capture location, scale, shape, etc. and every parameter may depend on complex additive terms (fixed, random, smooth, spatial, etc.) similar to a generalized additive model. Hyperpriors for all effects can be elicited within the package. Including complex tensor product interaction terms and variable selection priors. The basic model is explained in Klein and Kneib (2016) <[doi:10.1214/15-BA983](https://doi.org/10.1214/15-BA983)>.

**Depends** R (>= 3.1.0)

**Imports** splines, GB2, MASS, stats, pscl, mvtnorm, mgcv, graphics,  
doParallel, parallel

**LazyData** true

**License** GPL-2

**NeedsCompilation** no

**RoxygenNote** 6.0.1

**Author** Nadja Klein [aut, cre]

**Repository** CRAN

**Date/Publication** 2018-10-06 21:00:03 UTC

## R topics documented:

dapprox_unif . . . . .	2
DesignM . . . . .	3
get_theta . . . . .	4
get_theta_aunif . . . . .	5
get_theta_ga . . . . .	6
get_theta_gbp . . . . .	8
get_theta_ig . . . . .	9
get_theta_linear . . . . .	11
hyperpar . . . . .	12

hyperparlin . . . . .	13
hyperpar_mod . . . . .	15
mdbeta . . . . .	16
mdf_aunif . . . . .	17
mdf_ga . . . . .	18
mdf_gbp . . . . .	19
mdf_ig . . . . .	20
mdf_sd . . . . .	21
papprox_unif . . . . .	22
rapprox_unif . . . . .	23
zambia_graph . . . . .	24
zambia_height92 . . . . .	24

<b>Index</b>	<b>26</b>
--------------	-----------

---

dapprox_unif	<i>Compute Density Function of Approximated (Differentiably) Uniform Distribution.</i>
--------------	--

---

## Description

Compute Density Function of Approximated (Differentiably) Uniform Distribution.

## Usage

```
dapprox_unif(x, scale, tildec = 13.86294)
```

## Arguments

x	denotes the argument of the density function.
scale	the scale parameter originally defining the upper bound of the uniform distribution.
tildec	denotes the ratio between scale parameter $\theta$ and $s$ . The latter is responsible for the closeness of the approximation to the uniform distribution. See also below for further details and the default value.

## Details

The density of the uniform distribution for  $\tau$  is approximated by

$$p(\tau) = (1/(1 + \exp(\tau\tilde{c}/\theta - \tilde{c}))/(\theta(1 + \log(1 + \exp(-\tilde{c}))))$$

. This results in

$$p(\tau^2) = 0.5 * (\tau^2)^{-1/2} (1/(1 + \exp((\tau^2)^{1/2}\tilde{c}/\theta - \tilde{c}))/(\theta(1 + \log(1 + \exp(-\tilde{c}))))$$

for  $\tau^2$ .  $\tilde{c}$  is chosen such that  $P(\tau \leq \theta) \geq 0.95$ .

**Value**

the density.

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

**See Also**

[rapprox\\_unif](#), [papprox\\_unif](#)

---

DesignM

*Computing Designmatrix for Splines*

---

**Description**

This function computes the design matrix for Bayesian P-splines as it would be done in BayesX. The implementation currently works properly for default values (knots=20, degree=3).

**Usage**

```
DesignM(x, degree = 3, m = 20, min_x = min(x), max_x = max(x))
```

**Arguments**

x	the covariate vector.
degree	of the B-splines, default is 3.
m	number of knots, default is 20.
min_x	the left interval boundary, default is min(x).
max_x	the right interval boundary, default is max(x).

**Value**

a list with design matrix at distinct covariates, design matrix at all observations, index of sorted observations, the difference matrix, precision matrix and the knots used.

**Author(s)**

Nadja Klein

## References

Stefan Lang and Andy Brezger (2004). Bayesian P-Splines. *Journal of Computational and Graphical Statistics*, **13**, 183–212.

Belitz, C., Brezger, A., Klein, N., Kneib, T., Lang, S., Umlauf, N. (2015): BayesX - Software for Bayesian inference in structured additive regression models. Version 3.0.1. Available from <http://www.bayesx.org>.

---

get\_theta

*Find Scale Parameter for (Scale Dependent) Hyperprior*

---

## Description

This function implements a optimisation routine that computes the scale parameter  $\theta$  of the scale dependent hyperprior for a given design matrix and prior precision matrix such that approximately  $P(|f(x_k)| \leq c, k = 1, \dots, p) \geq 1 - \alpha$

## Usage

```
get_theta(alpha = 0.01, method = "integrate", Z, c = 3,
          eps = .Machine$double.eps, Kinv)
```

## Arguments

alpha	denotes the $1-\alpha$ level.
method	either integrate or trapezoid with integrate as default. trapezoid is a self-implemented version of the trapezoid rule.
Z	the design matrix.
c	denotes the expected range of the function.
eps	denotes the error tolerance of the result, default is <code>.Machine\$double.eps</code> .
Kinv	the generalised inverse of K.

## Value

an object of class `list` with values from `uniroot`.

## Author(s)

Nadja Klein

## References

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

**Examples**

```
## Not run:

set.seed(91179)
library(BayesX)
library(MASS)
# prior precision matrix to zambia data set
K <- read.gra(system.file("examples/zambia.gra", package="sdPrior"))
# generalised inverse of K
Kinv <- ginv(K)

# read data
dat <- read.table(system.file("examples/zambia_height92.raw", package="sdPrior"), header = TRUE)

# design matrix for spatial component
Z <- t(sapply(dat$district, FUN=function(x){1*(x==rownames(K))}))

# get scale parameter
theta <- get_theta(alpha = 0.01, method = "integrate", Z = Z,
                  c = 3, eps = .Machine$double.eps, Kinv = Kinv)$root

## End(Not run)
```

---

get_theta_aunif	<i>Find Scale Parameter for Hyperprior for Variances Where the Standard Deviations have an Approximated (Differentiably) Uniform Distribution.</i>
-----------------	--

---

**Description**

This function implements a optimisation routine that computes the scale parameter  $\theta$  of the prior  $\tau^2$  (corresponding to a differentiably approximated version of the uniform prior for  $\tau$ ) for a given design matrix and prior precision matrix such that approximately  $P(|f(x_k)| \leq c, k = 1, \dots, p) \geq 1 - \alpha$

**Usage**

```
get_theta_aunif(alpha = 0.01, method = "integrate", Z, c = 3,
               eps = .Machine$double.eps, Kinv)
```

**Arguments**

alpha	denotes the $1-\alpha$ level.
method	with integrate as default. Currently no further method implemented.
Z	the design matrix.
c	denotes the expected range of the function.
eps	denotes the error tolerance of the result, default is <code>.Machine\$double.eps</code> .
Kinv	the generalised inverse of K.

**Value**

an object of class `list` with values from `uniroot`.

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

Andrew Gelman (2006). Prior Distributions for Variance Parameters in Hierarchical Models. *Bayesian Analysis*, **1**(3), 515–533.

**Examples**

```
set.seed(123)
library(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=20 inner knots
# yielding dim(K)=m+1-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K
Kinv <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B,nrow=1)
theta <- get_theta_aunif(alpha = 0.01, method = "integrate", Z = Z,
                        c = 3, eps = .Machine$double.eps, Kinv = Kinv)$root
```

---

get\_theta\_ga

*Find Scale Parameter for Gamma (Half-Normal) Hyperprior*

---

**Description**

This function implements a optimisation routine that computes the scale parameter  $\theta$  of the gamma prior for  $\tau^2$  (corresponding to a half-normal prior for  $\tau$ ) for a given design matrix and prior precision matrix such that approximately  $P(|f(x_k)| \leq c, k = 1, \dots, p) \geq 1 - \alpha$

**Usage**

```
get_theta_ga(alpha = 0.01, method = "integrate", Z, c = 3,
             eps = .Machine$double.eps, Kinv)
```

**Arguments**

alpha	denotes the $1-\alpha$ level.
method	with integrate as default. Currently no further method implemented.
Z	the design matrix.
c	denotes the expected range of the function.
eps	denotes the error tolerance of the result, default is <code>.Machine\$double.eps</code> .
Kinv	the generalised inverse of K.

**Value**

an object of class `list` with values from `uniroot`.

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

Andrew Gelman (2006). Prior Distributions for Variance Parameters in Hierarchical Models. *Bayesian Analysis*, **1**(3), 515–533.

**Examples**

```
set.seed(123)
require(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=20 inner knots
# yielding dim(K)=m+1-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K
Kinv <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B,nrow=1)
theta <- get_theta_ga(alpha = 0.01, method = "integrate", Z = Z,
                     c = 3, eps = .Machine$double.eps, Kinv = Kinv)$root
```

## Not run:

```
set.seed(91179)
library(BayesX)
library(MASS)
# prior precision matrix to zambia data set
K <- read.gra(system.file("examples/zambia.gra", package="sdPrior"))
# generalised inverse of K
Kinv <- ginv(K)
```

```

# read data
dat <- read.table(system.file("examples/zambia_height92.raw", package="sdPrior"), header = TRUE)

# design matrix for spatial component
Z <- t(sapply(dat$district, FUN=function(x){1*(x==rownames(K))}))

# get scale parameter
theta <- get_theta_ga(alpha = 0.01, method = "integrate", Z = Z,
                     c = 3, eps = .Machine$double.eps, Kinv = Kinv)$root

## End(Not run)

```

---

get_theta_gbp	<i>Find Scale Parameter for Generalised Beta Prime (Half-Cauchy) Hyperprior</i>
---------------	---

---

## Description

This function implements a optimisation routine that computes the scale parameter  $\theta$  of the gamma prior for  $\tau^2$  (corresponding to a half cauchy for  $\tau$ ) for a given design matrix and prior precision matrix such that approximately  $P(|f(x_k)| \leq c, k = 1, \dots, p) \geq 1 - \alpha$

## Usage

```
get_theta_gbp(alpha = 0.01, method = "integrate", Z, c = 3,
              eps = .Machine$double.eps, Kinv)
```

## Arguments

alpha	denotes the $1-\alpha$ level.
method	with integrate as default. Currently no further method implemented.
Z	the design matrix.
c	denotes the expected range of the function.
eps	denotes the error tolerance of the result, default is <code>.Machine\$double.eps</code> .
Kinv	the generalised inverse of K.

## Value

an object of class `list` with values from `uniroot`.

## Author(s)

Nadja Klein



## References

- Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.
- Andrew Gelman (2006). Prior Distributions for Variance Parameters in Hierarchical Models. *Bayesian Analysis*, **1**(3), 515–533.

## Examples

```

set.seed(123)
require(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=20 inner knots
# yielding dim(K)=m+1-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K
Kinvs <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B, nrow=1)
theta <- get_theta_gbp(alpha = 0.01, method = "integrate", Z = Z,
                      c = 3, eps = .Machine$double.eps, Kinvs = Kinvs)$root

## Not run:

set.seed(91179)
library(BayesX)
library(MASS)
# prior precision matrix to zambia data set
K <- read.gra(system.file("examples/zambia.gra", package="sdPrior"))
# generalised inverse of K
Kinvs <- ginv(K)

# read data
dat <- read.table(system.file("examples/zambia_height92.raw", package="sdPrior"), header = TRUE)

# design matrix for spatial component
Z <- t(sapply(dat$district, FUN=function(x){1*(x==rownames(K))}))

# get scale parameter
theta <- get_theta_gbp(alpha = 0.01, method = "integrate", Z = Z,
                      c = 3, eps = .Machine$double.eps, Kinvs = Kinvs)$root

## End(Not run)

```

**Description**

This function implements a optimisation routine that computes the scale parameter  $b$  of the inverse gamma prior for  $\tau^2$  when  $a = b = \epsilon$  with  $\epsilon$  small for a given design matrix and prior precision matrix such that approximately  $P(|f(x_k)| \leq c, k = 1, \dots, p) \geq 1 - \alpha$  When  $a$  unequal to the shape parameter  $a$  has to be specified.

**Usage**

```
get_theta_ig(alpha = 0.01, method = "integrate", Z, c = 3,
  eps = .Machine$double.eps, Kinv, equals = FALSE, a = 1,
  type = "marginalt")
```

**Arguments**

alpha	denotes the $1-\alpha$ level.
method	with integrate as default. Currently no further method implemented.
Z	the design matrix.
c	denotes the expected range of the function.
eps	denotes the error tolerance of the result, default is <code>.Machine\$double.eps</code> .
Kinv	the generalised inverse of $K$ .
equals	saying whether $a=b$ . The default is FALSE due to the fact that $a$ is a shape parameter.
a	is the shape parameter of the inverse gamma distribution, default is 1.
type	is either numerical integration ( <code>integrate</code> ) of to obtain the marginal distribution of $z'_p \beta$ or the theoretical marginal t-distribution ( <code>marginalt</code> ). <code>marginalt</code> is the default value.

**Details**

Currently, the implementation only works properly for the cases  $a$  unequal  $b$ .

**Value**

an object of class `list` with values from `uniroot`.

**Author(s)**

Nadja Klein

**References**

- Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.
- Stefan Lang and Andreas Brezger (2004). Bayesian P-Splines. *Journal of Computational and Graphical Statistics*, **13**, 183-212.

**Examples**

```

set.seed(123)
library(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=20 inner knots
# yielding dim(K)=m+1-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K
Kinv <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B,nrow=1)
theta <- get_theta_ig(alpha = 0.01, method = "integrate", Z = Z,
                    c = 3, eps = .Machine$double.eps, Kinv = Kinv,
                    equals = FALSE, a = 1, type="marginalt")$root

```

---

get_theta_linear	<i>Find Scale Parameter for Inverse Gamma Hyperprior of Linear Effects with Spike and Slab Prior</i>
------------------	--

---

**Description**

This function implements a optimisation routine that computes the scale parameter  $v_2$  and selection parameter  $r$  of the inverse gamma prior  $IG(v_1, v_2)$  for  $\tau^2$  when  $\tau^2 \sim N(0, r(\delta)\tau^2)$  and given shape parameter such that approximately  $P(\beta \leq c_2 | spike) \geq 1 - \alpha_2$  and  $P(\beta \geq c_1 | slab) \geq 1 - \alpha_1$ .  $\alpha_1$  and  $\alpha_2$  should not be smaller than 0.1 due to numerical sensitivity and possible instability. Better change  $c_1, c_2$ .

**Usage**

```

get_theta_linear(alpha1 = 0.1, alpha2 = 0.1, c1 = 0.1, c2 = 0.1,
                eps = .Machine$double.eps, v1 = 5)

```

**Arguments**

alpha1	denotes the $1-\alpha_1$ level for $v_2$ .
alpha2	denotes the $1-\alpha_2$ level for $r$ .
c1	denotes the expected range of the linear effect in the slab part.
c2	denotes the expected range of the linear effect in the spike part.
eps	denotes the error tolerance of the result, default is <code>.Machine\$double.eps</code> .
v1	is the shape parameter of the inverse gamma distribution, default is 5.

**Value**

an object of class `list` with values from `uniroot`.

**Warning**

$\alpha_1$  and  $\alpha_2$  should not be smaller than 0.1 due to numerical sensitivity and possible instability. Better change  $c_1, c_2$ .

**Author(s)**

Nadja Klein

**References**

Nadja Klein, Thomas Kneib, Stefan Lang and Helga Wagner (2016). Automatic Effect Selection in Distributional Regression via Spike and Slab Priors. *Working Paper*.

**Examples**

```
set.seed(123)
result <- get_theta_linear()
r <- result$r
v2 <- result$v2

get_theta_linear(alpha1=0.1,alpha2=0.1,c1=0.5,c2=0.1,v1=5)
```

---

hyperpar

*Find Scale Parameters for Inverse Gamma Hyperprior of Nonlinear Effects with Spike and Slab Prior (Simulation-based)*

---

**Description**

This function implements a optimisation routine that computes the scale parameter  $b$  and selection parameter  $r$ . . Here, we assume an inverse gamma prior  $IG(a,b)$  for  $\psi^2$  and  $\tau^2 \sim N(0, r(\delta)\psi^2)$  and given shape parameter  $a$ , such that approximately  $P(f(x) \leq c|spike, \forall x \in D) \geq 1 - \alpha_1$  and  $P(\exists x \in D.s.t.f(x) \geq c|slab) \geq 1 - \alpha_2$ .

**Usage**

```
hyperpar(Z, Kinv, a = 5, c = 0.1, alpha1 = 0.1, alpha2 = 0.1,
  R = 10000, myseed = 123)
```

**Arguments**

Z	the row of the design matrix (or the complete matrix of several observations) evaluated at.
Kinv	the generalised inverse of $K$ .
a	is the shape parameter of the inverse gamma distribution, default is 5.
c	denotes the expected range of eqnf .
alpha1	denotes the $1-\alpha_1$ level for $b$ .

alpha2 denotes the  $1-\alpha_2$  level for  $r$ .  
 R denotes the number of replicates drawn during simulation.  
 myseed denotes the required seed for the simulation based method.

**Value**

an object of class `list` with root values  $r$ ,  $b$  from `uniroot`.

**Author(s)**

Nadja Klein

**References**

Nadja Klein, Thomas Kneib, Stefan Lang and Helga Wagner (2016). Spike and Slab Priors for Effect Selection in Distributional Regression. *Working Paper*.

**Examples**

```
set.seed(123)
library(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=22 inner knots
# yielding dim(K)=m+1-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K (same as if we used mixed model representation!)
Kinv <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B,nrow=1)
fgrid <- seq(-3,3,length=1000)
mdf <- hyperpar(Z,Kinv,a=5,c=0.1,alpha1=0.05,alpha2=0.05,R=10000,myseed=123)
```

---

hyperparlin

*Find Scale Parameter for Inverse Gamma Hyperprior of Linear Effects  
with Spike and Slab Prior*

---

**Description**

This function implements a optimisation routine that computes the scale parameter  $b$  and selection parameter  $r$ . Here, we assume an inverse gamma prior  $IG(a,b)$  for  $\tau^2$  and  $\beta|\delta, \tau^2 \sim N(0, r(\delta)\tau^2)$ . For given shape paramter  $a$  the user gets  $b$ ,  $r$  such that approximately  $P(\beta \leq c_2|spike) \geq 1 - \alpha_2$  and  $P(\beta \geq c_1|slab) \geq 1 - \alpha_1$  hold.

Note that if you observe numerical instabilities try not to specify  $\alpha_1$  and  $\alpha_2$  smaller than 0.1.

**Usage**

```
hyperparlin(alpha1 = 0.1, alpha2 = 0.1, c1 = 0.1, c2 = 0.1,  
  eps = .Machine$double.eps, a = 5)
```

**Arguments**

alpha1	denotes the $1-\alpha_1$ level for $b$ .
alpha2	denotes the $1-\alpha_2$ level for $r$ .
c1	denotes the expected range of the linear effect in the slab part.
c2	denotes the expected range of the linear effect in the spike part.
eps	denotes the error tolerance of the result, default is <code>.Machine\$double.eps</code> .
a	is the shape parameter of the inverse gamma distribution, default is 5.

**Value**

an object of class `list` with root values  $r, b$  from `uniroot`.

**Warning**

$\alpha_1$  and  $\alpha_2$  should not be smaller than 0.1 due to numerical sensitivity and possible instability. Better change  $c_1, c_2$ .

**Author(s)**

Nadja Klein

**References**

Nadja Klein, Thomas Kneib, Stefan Lang and Helga Wagner (2016). Automatic Effect Selection in Distributional Regression via Spike and Slab Priors. *Working Paper*.

**Examples**

```
set.seed(123)  
result <- hyperparlin()  
r <- result$r  
b <- result$b  
  
hyperparlin(alpha1=0.1,alpha2=0.1,c1=0.5,c2=0.1,a=5)
```

hyperpar\_mod

*Find Scale Parameter for modular regression***Description**

Find Scale Parameter for modular regression

**Usage**

```
hyperpar_mod(Z, K1, K2, A, c = 0.1, alpha = 0.1, omegaseq, omegaprob,
  R = 10000, myseed = 123, thetaseq = NULL, type = "IG",
  lowrank = FALSE, k = 5, mc = FALSE, ncores = 1, truncate = 1)
```

**Arguments**

Z	rows from the tensor product design matrix
K1	precision matrix1
K2	precision matrix2
A	constraint matrix
c	threshold from eq. (8) in Klein & Kneib (2016)
alpha	probability parameter from eq. (8) in Klein & Kneib (2016)
omegaseq	sequence of weights for the anisotropy
omegaprob	prior probabilities for the weights
R	number of simulations
myseed	seed in case of simulation. default is 123.
thetaseq	possible sequence of thetas. default is NULL.
type	type of hyperprior for tau/tau <sup>2</sup> ; options: IG => IG(1,theta) for tau <sup>2</sup> , SD => WE(0.5,theta) for tau <sup>2</sup> , HN => HN(0,theta) for tau, U => U(0,theta) for tau, HC => HC(0,theta) for tau
lowrank	default is FALSE. If TRUE a low rank approximation is used for Z with k columns.
k	only used if lowrank=TRUE. specifies target rank of low rank approximation. Default is 5.
mc	default is FALSE. only works im thetaseq is supplied. can parallel across thetaseq.
ncores	default is 1. number of cores is mc=TRUE
truncate	default is 1. If < 1 the lowrank approximation is based on on cumsum(values)/sum(values).

**Value**

the optimal value for theta

**Author(s)**

Nadja Klein

**References**

Kneib, T., Klein, N., Lang, S. and Umlauf, N. (2017) Modular Regression - A Lego System for Building Structured Additive Distributional Regression Models with Tensor Product Interactions *Working Paper*.

---

 mdbeta

---

*Marginal Density of  $\beta$* 


---

**Description**

This function computes the marginal density of  $\beta$  and for  $\beta$  on an equidistant grid specified by the user. Currently only implemented for  $\dim(\beta) = 1, 2$ .

**Usage**

```

mdbeta(D = 1, rangebeta, ngridbeta, a = 5, b = 25, r = 0.00025,
       a0 = 0.5, b0 = 0.5, plot = FALSE, log = FALSE)

```

**Arguments**

D	dimension of $\beta$ .
rangebeta	a vector containing the start and ending point of $\beta$ to be computed for.
ngridbeta	the number of grid values.
a	shape parameter of inverse gamma prior of $\psi^2$ .
b	scale parameter of inverse gamma prior of $\psi^2$ .
r	the scaling parameter $r(\delta = 1)$ in the variance $r(\delta)\psi^2$ of prior of $\tau^2$ .
a0	shape parameter of beta prior of $\omega$ .
b0	scale parameter of beta prior of $\omega$ .
plot	logical value (default is FALSE). If TRUE, a plot is also returned as the function <code>pl()</code> .
log	logical value (default is FALSE). If TRUE, $\log(p(\beta))$ is also returned in <code>logval</code> . as well as, if necessary, a plot function <code>logpl()</code> .

**Value**

the marginal density, the sequence of  $\beta$  and depending on specified plot, log arguments also the log-density and plot functions.

**Author(s)**

Nadja Klein



## References

Nadja Klein, Thomas Kneib, Stefan Lang and Helga Wagner (2016). Spike and Slab Priors for Effect Selection in Distributional Regression. *Working Paper*.

## Examples

```
set.seed(123)
#1-dimensional example
D = 1
ngridbeta = 1000
rangebeta = c(0.000001,1)
a0 = b0 = 0.5
a = 5
b = 50
r = 0.005
mdf <- mdbeta(D=1,rangebeta,ngridbeta,a=a,b=b,r=r,a0=a0,b0=b0)

#2-dimensional example
D = 2
ngridbeta = 100
rangebeta = c(0.000001,8)
a0 = b0 = 0.5
a = 5
b = 50
r = 0.005
mdf <- mdbeta(D=2,rangebeta,ngridbeta,a=a,b=b,r=r,a0=a0,b0=b0,plot=TRUE,log=TRUE)
mdf$logpl()
```

---

mdf\_aunif

---

*Marginal Density for Given Scale Parameter and Approximated Uniform Prior for  $\tau$* 


---

## Description

This function computes the marginal density of  $z_p^t \beta$  for approximated uniform hyperprior for  $\tau$

## Usage

```
mdf_aunif(f, theta, Z, Kinv)
```

## Arguments

f	point the marginal density to be evaluated at.
theta	denotes the scale parameter of the approximated uniform hyperprior for $\tau$ .
Z	the row of the design matrix evaluated.
Kinv	the generalised inverse of K.

**Value**

the marginal density evaluated at point  $x$ .

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

**Examples**

```
set.seed(123)
library(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=20 inner knots
# yielding dim(K)=m+1-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K
Kinv <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B,nrow=1)
fgrid <- seq(-3,3,length=1000)
mdf <- mdf_aunif(fgrid,theta=0.0028,Z=Z,Kinv=Kinv)
```

---

mdf\_ga

*Marginal Density for Given Scale Parameter and Half-Normal Prior for  $\tau$*

---

**Description**

This function computes the marginal density of  $z_p' \beta$  for gamma priors for  $\tau^2$  (referring to a half-normal prior for  $\tau$ ).

**Usage**

```
mdf_ga(f, theta, Z, Kinv)
```

**Arguments**

**f** point the marginal density to be evaluated at.  
**theta** denotes the scale parameter of the gamma hyperprior for  $\tau^2$  (half-normal for  $\tau$ ).  
**Z** the row of the design matrix evaluated.  
**Kinv** the generalised inverse of **K**.

**Value**

the marginal density evaluated at point x.

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

**Examples**

```
set.seed(123)
library(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=20 inner knots
# yielding dim(K)=m+l-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K
Kinv <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B,nrow=1)
fgrid <- seq(-3,3,length=1000)
mdf <- mdf_ga(fgrid,theta=0.0028,Z=Z,Kinv=Kinv)
```

---

mdf\_gbp

*Marginal Density for Given Scale Parameter and Half-Cauchy Prior for  $\tau$*

---

**Description**

This function computes the marginal density of  $z_p' \beta$  for generalised beta prior hyperprior for  $\tau^2$  (half-Chauchy for  $\tau$ )

**Usage**

```
mdf_gbp(f, theta, Z, Kinv)
```

**Arguments**

f	point the marginal density to be evaluated at.
theta	denotes the scale parameter of the generalised beta prior hyperprior for $\tau^2$ (half-Chauchy for $\tau$ ).
Z	the row of the design matrix evaluated.
Kinv	the generalised inverse of K.

**Value**

the marginal density evaluated at point  $x$ .

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

**Examples**

```
set.seed(123)
library(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=20 inner knots
# yielding dim(K)=m+1-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K
Kinv <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B,nrow=1)
fgrid <- seq(-3,3,length=1000)
mdf <- mdf_gbp(fgrid,theta=0.0028,Z=Z,Kinv=Kinv)
```

---

mdf\_ig

*Marginal Density for Given Scale Parameter and Inverse Gamma  
Prior for  $\tau^2$*

---

**Description**

This function computes the marginal density of  $z_p' \beta$  for inverse gamma hyperpriors with shape parameter  $a=1$ .

**Usage**

```
mdf_ig(f, theta, Z, Kinv)
```

**Arguments**

**f** point the marginal density to be evaluated at.  
**theta** denotes the scale parameter of the inverse gamma hyperprior.  
**Z** the row of the design matrix evaluated.  
**Kinv** the generalised inverse of **K**.

**Value**

the marginal density evaluated at point x.

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

**Examples**

```
set.seed(123)
library(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=20 inner knots
# yielding dim(K)=m+1-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K
Kinv <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B, nrow=1)
fgrid <- seq(-3,3,length=1000)
mdf <- mdf_ig(fgrid, theta=0.0028, Z=Z, Kinv=Kinv)
```

---

mdf\_sd

*Marginal Density for Given Scale Parameter and Scale-Dependent Prior for  $\tau^2$*

---

**Description**

This function computes the marginal density of  $z_p' \beta$  for scale-dependent priors for  $\tau^2$

**Usage**

```
mdf_sd(f, theta, Z, Kinv)
```

**Arguments**

f                    point the marginal density to be evaluated at.  
theta                denotes the scale parameter of the scale-dependent hyperprior for  $\tau^2$ .  
Z                    the row of the design matrix evaluated.  
Kinv                the generalised inverse of K.

**Value**

the marginal density evaluated at point  $x$ .

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

**Examples**

```
set.seed(123)
library(MASS)
# prior precision matrix (second order differences)
# of a spline of degree l=3 and with m=20 inner knots
# yielding dim(K)=m+l-1=22
K <- t(diff(diag(22), differences=2))%*%diff(diag(22), differences=2)
# generalised inverse of K
Kinv <- ginv(K)
# covariate x
x <- runif(1)
Z <- matrix(DesignM(x)$Z_B,nrow=1)
fgrid <- seq(-3,3,length=1000)
mdf <- mdf_sd(fgrid,theta=0.0028,Z=Z,Kinv=Kinv)
```

---

papprox\_unif

*Compute Cumulative Distribution Function of Approximated (Differentiably) Uniform Distribution.*

---

**Description**

Compute Cumulative Distribution Function of Approximated (Differentiably) Uniform Distribution.

**Usage**

```
papprox_unif(x, scale, tildec = 13.86294)
```

**Arguments**

<code>x</code>	denotes the argument of cumulative distribution function
<code>scale</code>	the scale parameter originally defining the upper bound of the uniform distribution.
<code>tildec</code>	denotes the ratio between scale parameter $\theta$ and $s$ . The latter is responsible for the closeness of the approximation to the uniform distribution. See also below for further details and the default value.

**Details**

The cumulative distribution function of `dapprox_unif` is given by

$$(1/(\log(1 + \exp(-\tilde{c})) + \tilde{c})) * (\tilde{c} * (\tau^2)^{1/2}/\theta - \log(\exp((\tau^2)^{1/2}) * \tilde{c}/\theta) + \exp(\tilde{c})))$$

$\tilde{c}$  is chosen such that  $P(\tau^2 \leq \theta) \geq 0.95$ .

**Value**

the cumulative distribution function.

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

**See Also**

[rapprox\\_unif](#), [dapprox\\_unif](#)

---

rapprox_unif	<i>Draw Random Numbers from Approximated (Differentiably) Uniform Distribution.</i>
--------------	---

---

**Description**

Draw Random Numbers from Approximated (Differentiably) Uniform Distribution.

**Usage**

```
rapprox_unif(n = 100, scale, tildec = 13.86294, seed = 123)
```

**Arguments**

n	number of draws.
scale	the scale parameter originally defining the upper bound of the uniform distribution.
tildec	denotes the ratio between scale parameter $\theta$ and $s$ . The latter is responsible for the closeness of the approximation to the uniform distribution. See also below for further details and the default value.
seed	denotes the seed

**Details**

The method is based on the inversion method and the quantile function is computed numerically using `uniroot`.

**Value**

n draws with density `papprox_unif`.

**Author(s)**

Nadja Klein

**References**

Nadja Klein and Thomas Kneib (2015). Scale-Dependent Priors for Variance Parameters in Structured Additive Distributional Regression. *Working Paper*.

**See Also**

`rapprox_unif`, `papprox_unif`

---

zambia\_graph

*Prior precision matrix for spatial variable in Zambia data set*

---

**Description**

This is a 57x57 matrix containing row- and columnwise the regions of Zambia, and the entries define the neighbourhoodstructure. The corresponding map `sambia.bnd` can be downloaded from [http://www.stat.uni-muenchen.de/~kneib/regressionsbuch/daten\\_e.html](http://www.stat.uni-muenchen.de/~kneib/regressionsbuch/daten_e.html). from the `bnd` file the prior precision matrix is obtained by `library(BayesX) map <- read.bnd("zambia.bnd") K <- bnd2gra(map)`

---

zambia\_height92

*Malnutrition in Zambia*

---

**Description**

The primary goal of a statistical analysis is to determine the effect of certain socioeconomic variables of the child, the mother, and the household on the child's nutritional condition

- `zscore` child's Z-score
- `c_breastf` duration of breastfeeding in months
- `c_age` child's age in months
- `m_agebirth` mother's age at birth in years



- m\_height mother's height in centimeter
- m\_bmi mother's body mass index
- m\_education mother's level of education
- m\_work mother's work status
- region region of residence in Zambia
- district district of residence in Zambia

**Format**

A data frame with 4421 rows and 21 variables

**Source**

[http://www.stat.uni-muenchen.de/~kneib/regressionsbuch/daten\\_e.html](http://www.stat.uni-muenchen.de/~kneib/regressionsbuch/daten_e.html)

# Index

dapprox\_unif, [2](#), [23](#)  
DesignM, [3](#)

get\_theta, [4](#)  
get\_theta\_aunif, [5](#)  
get\_theta\_ga, [6](#)  
get\_theta\_gbp, [8](#)  
get\_theta\_ig, [9](#)  
get\_theta\_linear, [11](#)

hyperpar, [12](#)  
hyperpar\_mod, [15](#)  
hyperparlin, [13](#)

mdbeta, [16](#)  
mdf\_aunif, [17](#)  
mdf\_ga, [18](#)  
mdf\_gbp, [19](#)  
mdf\_ig, [20](#)  
mdf\_sd, [21](#)

papprox\_unif, [3](#), [22](#), [24](#)

rapprox\_unif, [3](#), [23](#), [23](#), [24](#)

uniroot, [4](#), [6–8](#), [10](#), [11](#), [13](#), [14](#), [24](#)

zambia\_graph, [24](#)  
zambia\_height92, [24](#)