

# Package ‘InvStablePrior’

August 22, 2023

**Type** Package

**Title** Inverse Stable Prior for Widely-Used Exponential Models

**Version** 0.1.1

**Description** Contains functions that allow Bayesian inference on a parameter of some widely-used exponential models. The functions can generate independent samples from the closed-form posterior distribution using the inverse stable prior. Inverse stable is a non-conjugate prior for a parameter of an exponential subclass of discrete and continuous data distributions (e.g. Poisson, exponential, inverse gamma, double exponential (Laplace), half-normal/half-Gaussian, etc.). The prior class provides flexibility in capturing a wide array of prior beliefs (right-skewed and left-skewed) as modulated by a parameter that is bounded in  $(0,1)$ . The generated samples can be used to simulate the prior and posterior predictive distributions. More details can be found in Cahoy and Sedransk (2019) <[doi:10.1007/s42519-018-0027-2](https://doi.org/10.1007/s42519-018-0027-2)>. The package can also be used as a teaching demo for introductory Bayesian courses.

**License** GPL ( $\geq 3$ )

**Encoding** UTF-8

**Imports** stats, fdrtool, nimble

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Dexter Cahoy [aut, cre],  
Joseph Sedransk [aut]

**Maintainer** Dexter Cahoy <dexter.cahoy@gmail.com>

**Repository** CRAN

**Date/Publication** 2023-08-21 22:10:02 UTC

## R topics documented:

InvStablePrior . . . . .	2
isdexp . . . . .	2
isexp . . . . .	4
ishalfn . . . . .	6
isinvgam . . . . .	8
ispoi . . . . .	9

**Index****12**


---

InvStablePrior	<i>InvStablePrior Package</i>
----------------	-------------------------------

---

**Description**

Contains random number generation, plotting, and estimation algorithms for performing Bayesian inference on a parameter of some widely-used exponential data models. The package contains algorithms for the inverse stable-Poisson, inverse stable-exponential, inverse stable-double exponential, inverse stable-inverse gamma, and inverse stable-half-normal models.

**Details**

## References:

Cahoy and Sedransk (2019)<doi:10.1007/s42519-018-0027-2>

Meerschaert and Straka (2013)<doi:10.1051/mmnp/20138201>

Mainardi, Mura, and Pagnini (2010) <doi:10.1155/2010/104505>

**Author(s)**

Dexter Cahoy <cahoyd@uhd.edu>

---

isdexp	<i>Bayesian inference for the true rate of double exponential (Laplace) distribution.</i>
--------	-------------------------------------------------------------------------------------------

---

**Description**

Generates random numbers from the prior and posterior distributions of the inverse stable-double exponential model. The random variates can be used to simulate prior and posterior predictive distributions as well.

**Usage**

```
isdexp(x, B, alp, rho)
```

**Arguments**

x	vector of double exponential (Laplace) data.
B	test size for the adaptive rejection sampling algorithm.
alp	value between 0 and 1 that controls the shape of the inverse stable prior.
rho	positive value that scales the mean of the inverse stable prior.

**Value**

list consisting of the vectors of random numbers from the prior and posterior distributions, the accepted sample size, and the acceptance probability of the adaptive rejection sampling procedure (Algorithm 2 of the first reference below).

**References**

Cahoy and Sedransk (2019). *Inverse stable prior for exponential models*. Journal of Statistical Theory and Practice, 13, Article 29. <doi:10.1007/s42519-018-0027-2>

Meerschaert and Straka (2013). *Inverse stable subordinators*. Math. Model. Nat. Phenom., 8(2), 1-16. <doi:10.1051/mmnp/20138201>

Mainardi, Mura, and Pagnini (2010). *The M-Wright Function in Time-Fractional Diffusion Processes: A Tutorial Survey*. Int. J. Differ. Equ., Volume 2010. <doi:10.1155/2010/104505>

**Examples**

```
alp=0.95
require(nimble)
dat=rdexp(30, location = 0, rate = 2)
rho=1/sd(dat)

#b=n
#a=sum(abs(dat) )
#rho=optimize(function(r){exp(-b)*(b/a)^b - (r^b)*exp(-a*r)}, c(0,20), tol=10^(-50) )$min

out= isdexp(dat, B=1000000, alp , rho)
#prior samples
thetprior=unlist(out[2])
summary(thetprior)

#posterior samples
thet=unlist(out[1])

#95% Credible intervals
quantile (thet, c(0.025,0.975) )
summary(thet)

#The accepted sample size:
unlist(out[3])

#The acceptance probability:
unlist(out[4])

#Plotting with normalization to have a maximum of 1
#for comparing prior and posterior
out2=density(thet)
ymaxpost=max(out2$y)
out3=density(thetprior)
ymaxprior=max(out3$y)
```

```

plot(out2$x,out2$y/ymaxpost, xlim=c(0,5), col="blue", type="l",
     xlab="theta", ylab="density", lwd=2, frame.plot=FALSE)
lines(out3$x,out3$y/ymaxprior,lwd=2,col="red")
#points(dat,rep(0,length(dat)), pch='*')

#Generating 1000 random numbers from the Inverse Stable (alpha=0.4,rho=5) prior
U1 = runif(1000)
U2 = runif(1000)
alp=0.4
rho=5
stab = ( ( sin(alp*pi*U1)*sin((1-alp)*pi*U1)^(1/alp-1) )
        / ( ( sin(pi*U1)^(1/alp) )*abs(log(U2))^(1/alp-1)) )
#Inverse stable random numbers are below:
#rho*stab^(-alp)

```

---

isexp

*Bayesian inference for the true rate of exponential distribution.*


---

## Description

Generates random numbers from the prior and posterior distributions of the inverse stable-exponential model. The random variates can be used to simulate prior and posterior predictive distributions as well.

## Usage

```
isexp(x, B, alp, rho)
```

## Arguments

x	vector of exponential data.
B	test size for the adaptive rejection sampling algorithm.
alp	value between 0 and 1 that controls the shape of the inverse stable prior.
rho	positive value that scales the mean of the inverse stable prior.

## Value

list consisting of the vectors of random numbers from the prior and posterior distributions, the accepted sample size, and the acceptance probability of the adaptive rejection sampling procedure (Algorithm 2 of the first reference below).

## References

- Cahoy and Sedransk (2019). *Inverse stable prior for exponential models*. Journal of Statistical Theory and Practice, 13, Article 29. <doi:10.1007/s42519-018-0027-2>
- Meerschaert and Straka (2013). *Inverse stable subordinators*. Math. Model. Nat. Phenom., 8(2), 1-16. <doi:10.1051/mmnp/20138201>
- Mainardi, Mura, and Pagnini (2010). *The M-Wright Function in Time-Fractional Diffusion Processes: A Tutorial Survey*. Int. J. Differ. Equ., Volume 2010. <doi:10.1155/2010/104505>

## Examples

```

alp=0.5
dat=rexp(10,rate=0.5)
rho=1/mean(dat)
#rho=1/mean(dat) + 3*sd(dat)
#rho=1/mean(dat) - 3*sd(dat)

#b=length(dat)
#a=sum(dat)
#rho=optimize(function(r){exp(-b)*(b/a)^b - (r^b)*exp(-a*r)}, c(0,20), tol=10^(-50) )$min

out= isexp(dat, B=1000000, alp , rho)
#prior samples
thetprior=unlist(out[2])
summary(thetprior)

#posterior samples
thet=unlist(out[1])

#95% Credible intervals
quantile (thet, c(0.025,0.975) )
summary(thet)

#The accepted sample size:
unlist(out[3])

#The acceptance probability:
unlist(out[4])

#Plotting with normalization to have a maximum of 1
#for comparing prior and posterior
out2=density(thet)
ymaxpost=max(out2$y)
out3=density(thetprior)
ymaxprior=max(out3$y)
plot(out2$x,out2$y/ymaxpost, xlim=c(0,1), col="blue", type="l",
      xlab="theta", ylab="density",lwd=2, frame.plot=FALSE)
lines(out3$x,out3$y/ymaxprior,lwd=2, col="red")
#points(dat,rep(0,length(dat)), pch='*')

```

```
#Generating 1000 random numbers from the Inverse Stable (alpha=0.4,rho=5) prior
U1 = runif(1000)
U2 = runif(1000)
alp=0.4
rho=5
stab = ( ( sin(alp*pi*U1)*sin((1-alp)*pi*U1)^(1/alp-1) )
/ ( ( sin(pi*U1)^(1/alp) )*abs(log(U2))^(1/alp-1)) )
#Inverse stable random numbers are below:
#rho*stab^(-alp)
```

---

ishalfn	<i>Bayesian inference for the true inverse mean/rate of half-normal/half-Gaussian distribution.</i>
---------	-----------------------------------------------------------------------------------------------------

---

### Description

Generates random numbers from the prior and posterior distributions of the inverse stable-half-normal model. The random variates can be used to simulate prior and posterior predictive distributions as well.

### Usage

```
ishalfn(x, B, alp, rho)
```

### Arguments

x	vector of half-normal/half-Gaussian data.
B	test size for the adaptive rejection sampling algorithm.
alp	value between 0 and 1 that controls the shape of the inverse stable prior.
rho	positive value that scales the mean of the inverse stable prior.

### Value

list consisting of the vectors of random numbers from the prior and posterior distributions, the accepted sample size, and the acceptance probability of the adaptive rejection sampling procedure (Algorithm 2 of the first reference below).

### References

- Cahoy and Sedransk (2019). *Inverse stable prior for exponential models*. Journal of Statistical Theory and Practice, 13, Article 29. <doi:10.1007/s42519-018-0027-2>
- Meerschaert and Straka (2013). *Inverse stable subordinators*. Math. Model. Nat. Phenom., 8(2), 1-16. <doi:10.1051/mmnp/20138201>
- Mainardi, Mura, and Pagnini (2010). *The M-Wright Function in Time-Fractional Diffusion Processes: A Tutorial Survey*. Int. J. Differ. Equ., Volume 2010. <doi:10.1155/2010/104505>

**Examples**

```

alp=0.95
require(fdrtool)
dat=rhalfnorm(100, theta=sqrt(pi/2) )
rho=1/mean(dat)

#b=length(dat)/2
#a=sum(dat^2)/pi
#rho=optimize(function(r){exp(-b)*(b/a)^b - (r^b)*exp(-a*r)}, c(0,20), tol=10^(-50) )$min

out= ishalfn(dat, B=1000000, alp , rho)
#prior samples
thetprior=unlist(out[2])
summary(thetprior)

#posterior samples
thet=unlist(out[1])

#95% Credible intervals
quantile (thet, c(0.025,0.975) )
summary(thet)

#The accepted sample size:
unlist(out[3])

#The acceptance probability:
unlist(out[4])

#Plotting with normalization to have a maximum of 1
#for comparing prior and posterior
out2=density(thet)
ymaxpost=max(out2$y)
out3=density(thetprior)
ymaxprior=max(out3$y)
plot(out2$x,out2$y/ymaxpost, xlim=c(0,2), col="blue", type="l",
      xlab="theta", ylab="density", lwd=2, frame.plot=FALSE)
lines(out3$x,out3$y/ymaxprior, lwd=2,col="red")
#points(dat,rep(0,length(dat)), pch='*')

#Generating 1000 random numbers from the Inverse Stable (alpha=0.4,rho=5) prior
U1 = runif(1000)
U2 = runif(1000)
alp=0.4
rho=5
stab = ( ( sin(alp*pi*U1)*sin((1-alp)*pi*U1)^(1/alp-1) )
/ ( ( sin(pi*U1)^(1/alp) )*abs(log(U2))^(1/alp-1)) )
#Inverse stable random numbers are below:
#rho*stab^(-alp)

```

---

 isinvgam

*Bayesian inference for the true scale of inverse gamma distribution.*


---

### Description

Generates random numbers from the prior and posterior distributions of the inverse stable-inverse gamma model. The random variates can be used to simulate prior and posterior predictive distributions as well.

### Usage

```
isinvgam(x, B, alp, rho, sh)
```

### Arguments

x	vector of data from inverse gamma population.
B	test size for the adaptive rejection sampling algorithm.
alp	value between 0 and 1 that controls the shape of the inverse stable prior.
rho	positive value that scales the mean of the inverse stable prior.
sh	a required known shape parameter value for the inverse gamma distribution.

### Value

list consisting of the vectors of random numbers from the prior and posterior distributions, the accepted sample size, and the acceptance probability of the adaptive rejection sampling procedure (Algorithm 2 of the first reference below).

### References

Cahoy and Sedransk (2019). *Inverse stable prior for exponential models*. Journal of Statistical Theory and Practice, 13, Article 29. <doi:10.1007/s42519-018-0027-2>

Meerschaert and Straka (2013). *Inverse stable subordinators*. Math. Model. Nat. Phenom., 8(2), 1-16. <doi:10.1051/mmnp/20138201>

Mainardi, Mura, and Pagnini (2010). *The M-Wright Function in Time-Fractional Diffusion Processes: A Tutorial Survey*. Int. J. Differ. Equ., Volume 2010. <doi:10.1155/2010/104505>

### Examples

```
alp=0.95
require(nimble)
sh=2.1 # a>2 so variance exists, known
dat=rinvgamma(50, shape=sh, scale = 4)
rho= (sh-1)*mean(dat)
```



```

#b=n
#a=sum(1/dat )
#rho=optimize(function(r){exp(-b)*(b/a)^b - (r^b)*exp(-a*r)}, c(0,20), tol=10^(-50) )$min

out= isinvgam(dat, B=1000000, alp , rho,sh)
#prior samples
thetprior=unlist(out[2])
summary(thetprior)

#posterior samples
thet=unlist(out[1])
summary(thet)

#95% Credible intervals
quantile (thet, c(0.025,0.975) )
summary(thet)

#The accepted sample size:
unlist(out[3])

#The acceptance probability:
unlist(out[4])

#Plotting with normalization to have a maximum of 1
#for comparing prior and posterior
out2=density(thet)
ymaxpost=max(out2$y)
out3=density(thetprior)
ymaxprior=max(out3$y)
plot(out2$x,out2$y/ymaxpost, xlim=c(0,5), col="blue", type="l",
     xlab="theta", ylab="density",lwd=2, frame.plot=FALSE)
lines(out3$x,out3$y/ymaxprior,lwd=2,col="red")
#points(dat,rep(0,length(dat)), pch='*')

#Generating 1000 random numbers from the Inverse Stable (alpha=0.4,rho=5) prior
U1 = runif(1000)
U2 = runif(1000)
alp=0.4
rho=5
stab = ( ( sin(alp*pi*U1)*sin((1-alp)*pi*U1)^(1/alp-1) )
        / ( ( sin(pi*U1)^(1/alp) )*abs(log(U2))^(1/alp-1)) )
#Inverse stable random numbers are below:
#rho*stab^(-alp)

```

**Description**

Generates random numbers from the prior and posterior distributions of the inverse stable-Poisson model. The random variates can be used to simulate prior and posterior predictive distributions as well.

**Usage**

```
ispoi(x, B, alp, rho)
```

**Arguments**

x	vector of Poisson count data.
B	test size for the adaptive rejection sampling algorithm.
alp	value between 0 and 1 that controls the shape of the inverse stable prior.
rho	positive value that scales the mean of the inverse stable prior.

**Value**

list consisting of the vectors of random numbers from the prior and posterior distributions, the accepted sample size, and the acceptance probability of the adaptive rejection sampling procedure (Algorithm 2 of the first reference below).

**References**

Cahoy and Sedransk (2019). *Inverse stable prior for exponential models*. Journal of Statistical Theory and Practice, 13, Article 29. <doi:10.1007/s42519-018-0027-2>

Meerschaert and Straka (2013). *Inverse stable subordinators*. Math. Model. Nat. Phenom., 8(2), 1-16. <doi:10.1051/mmnp/20138201>

Mainardi, Mura, and Pagnini (2010). *The M-Wright Function in Time-Fractional Diffusion Processes: A Tutorial Survey*. Int. J. Differ. Equ., Volume 2010. <doi:10.1155/2010/104505>

**Examples**

```
alp=0.9
dat=rpois(50,lambda=10)
rho=mean(dat)
#rho=mean(dat) + 3*sd(dat)
#rho=mean(dat) - 3*sd(dat)

#a=length(dat)
#b=sum(dat)
#rho=optimize(function(r){exp(-b)*(b/a)^b - (r^b)*exp(-a*r)}, c(0,20), tol=10^(-50) )$min

out= ispoi(dat, B=1000000, alp , rho)
#prior samples
thetprior=unlist(out[2])
summary(thetprior)
```

```
#posterior samples
thet=unlist(out[1])

#95% Credible intervals
quantile (thet, c(0.025,0.975) )
summary(thet)

#The accepted sample size:
unlist(out[3])

#The acceptance probability:
unlist(out[4])

#Plotting with normalization to have a maximum of 1
#for comparing prior and posterior
out2=density(thet)
ymaxpost=max(out2$y)
out3=density(thetprior)
ymaxprior=max(out3$y)
plot(out2$x,out2$y/ymaxpost, xlim=c(0,15), col="blue", type="l",
      xlab="theta", ylab="density", lwd=2, frame.plot=FALSE)
lines(out3$x,out3$y/ymaxprior,lwd=2,col="red")
#points(dat,rep(0,length(dat)), pch='*')

#Generating 1000 random numbers from the Inverse Stable (alpha=0.4,rho=5) prior
U1 = runif(1000)
U2 = runif(1000)
alp=0.4
rho=5
stab = ( ( sin(alp*pi*U1)*sin((1-alp)*pi*U1)^(1/alp-1) )
/ ( ( sin(pi*U1)^(1/alp) )*abs(log(U2))^(1/alp-1)) )
#Inverse stable random numbers are below:
#rho*stab^(-alp)
```

# Index

InvStablePrior, [2](#)  
InvStablePrior-package  
    ([InvStablePrior](#)), [2](#)  
isdexp, [2](#)  
isexp, [4](#)  
ishalfn, [6](#)  
isinvgam, [8](#)  
ispoi, [9](#)