Package ‘BayesESS’

November 25, 2019

Type Package

Title Determining Effective Sample Size

Version 0.1.19

Author Jaejoon Song, Satoshi Morita, J. Jack Lee

Maintainer Jaejoon Song <jaejoonsong@gmail.com>

Description Determines effective sample size of a parametric prior distribution
in Bayesian models. For a web-
based Shiny application related to this package, see <https://implement.shinyapps.io/bayesess/>.

Depends MCMCpack, stats, LaplacesDemon

Imports Rcpp, dfcrm, MatrixModels, MASS

LinkingTo Rcpp, RcppArmadillo, RcppEigen

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 5.0.1

Suggests knitr, rmarkdown

NeedsCompilation yes

Repository CRAN

Date/Publication 2019-11-25 19:50:06 UTC

R topics documented:

ess *Determines effective sample size of a parametric prior distribution*

Description

Determines effective sample size of a parametric prior distribution in Bayesian conjugate models (beta-binomial, gamma-exponential, gamma-Poisson, dirichlet-multinomial, normal-normal, inverse-chi-squared-normal, inverse-gamma-normal), Bayesian linear and logistic regression models, Bayesian continual reassessment method (CRM), and Bayesian time to event model.

Usage

```
ess(model,label,prior,m,nsim,ncov,svec1,svec2,
    PI,betaSD,target,
    obswin,rate,accrual,
    shapeParam,scaleParam,
    fast=TRUE)
```

Arguments

model	Model specifications. Options are: 'betaBin' for beta-binomial model; 'gammaEx' for gamma-exponential model; 'gammaPois' for gamma-Poisson model; 'dirMult' for dirichlet-multinomial model; 'normNorm' for normal-normal model; 'invChisqNorm' for scaled-inverse-chi-squared-normal model; or 'invGammaNorm' for inverse-gamma-normal models. For non-conjugate models, options are: 'linreg' for linear regression model; and 'logistic' for logistic regression models. In addition, the continual reassessment method (CRM) can be specified as: model = 'crm', or as 'tite.crm' for the time-to-event CRM (TITE CRM). Finally, time to event models can be specified as model = 'surv'
label	Optional labeling for hyperparameters. Please see examples for sample usage.
prior	Prior distribution specification specified as a list. Options are 'beta', 'gamma', 'dirichlet', 'norm' (for normal prior), 'scaled-inverse-chisquared', and 'inverse-gamma'.
m	A positive integer specified as an maximum value in which ESS is searched.
nsim	Number of simulations for numerical approximation (specified only for model = 'linreg' or model = 'logistic').
ncov	(Required for linear or logistic regression model) Number of covariates
svec1	(Required for linear or logistic regression model) Specification of first subvector for calculating ESS
svec2	(Required for linear or logistic regression model) Specification of second subvector for calculating ESS
PI	(Required for CRM) A vector of the true toxicity probabilites associated with the doses.
betaSD	(Required for CRM) Standard deviation of the normal prior of the model parameter.

`target`	(Required for CRM) The target dose limiting toxicity (DLT) rate.
`obswin`	(Further required for TITE (time-to-event) CRM) The observation window with respect to which the maximum tolerated dose (MTD) is defined. Default is obswin=30.
`rate`	(Further required for TITE (time-to-event) CRM) Patient arrival rate: Expected number of arrivals per observation window. Example: obswin=6 and rate=3 means expecting 3 patients arrive in 6 time units. Default is rate=2.
`accrual`	(Further required for TITE (time-to-event) CRM) Patient accrual scheme. Default is accrual="poisson". Alternatively use accrual="fixed" whereby inter-patient arrival is fixed.
`shapeParam`	(Required for time to event model) Shape parameter of the inverse gamma prior
`scaleParam`	(Required for time to event model) Scale parameter of the inverse gamma prior
`fast`	Accelerate ESS computation for linear or logistic regression models with C++ code? Default is fast=TRUE.

Value

`ESS`	Returns ESS
`ESSsubvec1`	(For linear or logistic regression model) ESS for the first sub-vector
`ESSsubvec2`	(For linear or logistic regression model) ESS for the second sub-vector

Author(s)

Jaejoon Song <jjsong2@mdanderson.org>, Satoshi Morita <smorita@kuhp.kyoto-u.ac.jp>, J. Jack Lee <jjlee@mdanderson.org>

References

Morita, S., Thall, P. F., and Muller, P. (2008). Determining the effective sample size of a parametric prior. Biometrics, 64, 595-602.

Morita, S., Thall, P. F., and Muller, P. (2010). Evaluating the impact of prior assumptions in Bayesian biostatistics. Stat Biosci, 2, 1-17.

O'Quigley J., Pepe M., Fisher, L. (1990).Continual reassessment method: A practical design for phase I clinical trials in cancer. Biometrics, 46, 33-48.

Thall, P. F., Wooten, L. H., Tannir, N. M. (2005). Monitoring event times in early phase clinical trials: some practical issues. Clinical Trials, 2, 467-478.

See Also

https://biostatistics.mdanderson.org/SoftwareDownload/

Examples

```
library(BayesESS)
# Calculating ESS for a beta-binomial model with
# beta(1,2) prior
ess(model='betaBin',prior=c('beta',1,2))

# Calculating ESS for a gamma-exponential model with
# gamma(2,4) prior
ess(model='gammaEx',prior=c('gamma',2,4))

# Calculating ESS for a gamma-Poisson model with
# gamma(2,4) prior
ess(model='gammaPois',prior=c('gamma',2,4))

# Calculating ESS for a dirichlet-multinomial model with
# dirichlet(10,15,20) prior
ess(model='dirMult',prior=c('dirichlet',10,15,20))

# Calculating ESS for a scaled-inverse-chi-squared-normal model
# when mean is known and variance is unknown
# with scaled-inverse-chi-squared(nu_0=10,sigma^2_0=1) prior for variance
# ESS for such model can be found analytically
ess(model='invChisqNorm',prior=c(10,1))

# Calculating ESS for a normal-normal model
# when mean is unknown and variance is known
# with normal(mu_0=10,sigma^2=1,n_0=30) prior (i.e. known variance is sigma^2=1)
# ESS for such model can be found analytically (ESS = n_0)
ess(model='normNorm',prior=c('norm',10,1,30))

# Calculating ESS for a scaled-inverse-chi-squared-normal model
# when mean and variance are both unknown
# with scaled-inverse-chi-squared(nu_0=10,sigma^2_0=1) prior for variance
# and normal(mu_0=1,sigma^2/phi=sigma^2/30) prior for mean
# Smaller nsim = 1000 is specified for illustration purposes
# The user can use nsim = 10,000 to carry out the most accurate ESS computations.
## Not run:
ess(model='invChisqNorm',prior=c(10,1,1,30),m=20,nsim=1000)

## End(Not run)

# Calculating ESS for a inverse-gamma-normal model
# when mean is known and variance is unknown
# with inverse-gamma(alpha=5,beta=5) prior
# Note: the inverse-gamma(nu_0/2,nu_0*sigma^2_0/2) prior is
# equivalent to scaled-inverse-chi-squared(nu_0,sigma^2_0)
# ESS for such model can be found analytically
ess(model='invGammaNorm',prior=c(5,5))

# Calculating ESS for a inverse-gamma-normal model
# when mean and variance are both unknown
# with inverse-gamma(alpha=5,beta=5) prior
```

```
# and normal(mu_0=1,sigma^2/phi=sigma^2/30) prior for mean
# Note: the inverse-gamma(nu_0/2,nu_0*sigma^2_0/2) prior is
# equivalent to scaled-inverse-chi-squared(nu_0,sigma^2_0)
# Smaller nsim = 1000 is specified for illustration purposes
# The user can use nsim = 10,000 to carry out the most accurate ESS computations.
## Not run:
ess(model='invGammaNorm',prior=c(5,5,1,30),m=20,nsim=1000)

## End(Not run)

# Calculating ESS for a linear regression model with
# three covariates, with priors specified as
# beta0 ~ N(0,1); beta1 ~ N(0,.1); beta2 ~ N(0,.2); beta3 ~ N(0,.3); tau ~ Gamma(1,1);
# Smaller nsim = 50 is specified for illustration purposes
# The user can use nsim = 10,000 to carry out the most accurate ESS computations.
# The value of nsim as low as 1,000 may be used to reduce runtime.
## Not run:
ess(model='linreg',label=c('beta0','beta1','beta2','beta3','tau'),
prior=list(c('norm',0,1),c('norm',0,.1),c('norm',0,.2),c('norm',0,.3),c('gamma',1,1)),
ncov=3,m=50,nsim=50,svec1=c(0,1,1,1,0),svec2=c(0,0,0,0,1))

## End(Not run)

# Calculating ESS for a linear regression model with
# two covariates, with priors specified as
# beta0 ~ N(0,1); beta1 ~ N(0,.1); beta2 ~ N(0,.2); tau ~ Gamma(1,1);
# Smaller nsim = 50 is specified for illustration purposes
# The user can use nsim = 10,000 to carry out the most accurate ESS computations.
# The value of nsim as low as 1,000 may be used to reduce runtime.
## Not run:
ess(model='linreg',label=c('beta0','beta1','beta2','tau'),
prior=list(c('norm',0,1),c('norm',0,.1),c('norm',0,.2),c('gamma',1,1)),
ncov=2,m=50,nsim=50,svec1=c(0,1,1,0),svec2=c(0,0,0,1))

## End(Not run)

# Calculating ESS for a logistic regression model with
# three covariates, with priors specified as
# beta0 ~ N(0,1); beta1 ~ N(0,1); beta2 ~ N(0,1); beta3 ~ N(0,1)
# Smaller nsim = 50 is specified for illustration purposes
# The user can use nsim = 10,000 to carry out the most accurate ESS computations.
# The value of nsim as low as 1,000 may be used to reduce runtime.
## Not run:
ess(model='logistic',label=c('beta0','beta1','beta2','beta3'),
prior=list(c('norm',0,1),c('norm',0,1),c('norm',0,1),c('norm',0,1)),
ncov=3,m=50,nsim=50,svec1=c(1,0,0,0),svec2=c(0,1,1,1))

## End(Not run)

# Calculating ESS for a continual reassessment method (CRM)
# with true toxicity probabilites PI=c(.02,.06,.10,.18,.30)
# prior is specified as N(0,2.5) with target DLT = 0.2
# Smaller nsim = 50 is specified for illustration purposes
```

```
# The user can use nsim = 10,000 to carry out the most accurate ESS computations.
# The value of nsim as low as 1,000 may be used to reduce runtime.
## Not run:
ess(model='crm',prior=c(.02,.06,.10,.18,.30),
       m=7,nsim=50,
       PI=c(.02,.06,.10,.18,.30),
       betaSD=sqrt(2.5),target=0.2)

## End(Not run)

# Calculating ESS for a TITE CRM
# with true toxicity probabilites PI=c(.02,.06,.10,.18,.30)
# prior is specified as N(0,1.5) with target DLT = 0.2
# Smaller nsim = 50 is specified for illustration purposes
# The user can use nsim = 10,000 to carry out the most accurate ESS computations.
# The value of nsim as low as 1,000 may be used to reduce runtime.
## Not run:
ess(model='tite.crm',prior=c(.02,.06,.10,.18,.30),
       m=7,nsim=50,
       PI=c(.02,.06,.10,.18,.30),
       betaSD=sqrt(1.5),target=0.2,obswin=30,rate=2,
       accrual="poisson")

## End(Not run)

# Calculating ESS for a time to event model
# prior is specified as inverse-gamma(5.348,30.161)
# Smaller nsim = 50 is specified for illustration purposes
# The user can use nsim = 10,000 to carry out the most accurate ESS computations.
# The value of nsim as low as 1,000 may be used to reduce runtime.
## Not run:
ess(model='surv',shapeParam=5.348,scaleParam=30.161,m=7,nsim=50)

## End(Not run)
```

Index