Package ‘varSelectIP’

February 20, 2015

Type Package
Title Objective Bayes Model Selection
Version 0.2-1
Date 2013-10-28
Author Andrew Womack and Gopal, V. and Novelo, L. L. and Casella, G.
Maintainer Andrew Womack <ajwomack@ufl.edu>
Depends MASS, mvtnorm
Description Objective Bayes Variable Selection in Linear Regression and Probit Models.
License GPL (>= 2)
LazyData yes
NeedsCompilation no
Repository CRAN
Date/Publication 2013-10-31 07:42:57

R topics documented:

  varSelectIP-package .................................................. 1
  varSelectIP ............................................................. 2
  varSelectOBayeslinear ............................................... 4

Index

  varSelectIP-package  Objective Bayes Model Selection

Description

  This package carries out objective Bayes model selection in probit and regression models.

Details

  1
There are now two user functions in this package: varSelectIP() and varSelectOBayeslinear(). The former is the original package functions used for linear and probit models. The latter is a new function that replaces the former for linear regression models as well as provides posterior inference for regression parameters.

Author(s)
Andrew Womack and Gopal, V. and Novelo, L. L. and Casella, G.
Maintainer: Andrew Womack <ajwomack@ufl.edu>

See Also
varSelectIP varSelectOBayeslinear

Description
This function will carry out a low-dimensional stochastic search in order to determine the “best” model, as measured by its posterior probability. The types of model that this function can handle are probit and regression models. For full details on the model set-up and the stochastic search, please refer to the papers listed below.

Usage
varSelectIP(response, covariates.retain = NULL, covariates.test, nsim,
keep, q, a = 0.2, model.type = c("probit", "reg"),
save.every = 50, out.fname = "models.csv")

Arguments
response          The vector of response values. If a probit model, this should be a binary vector with the 0’s coming before the 1’s.
covariates.retain  A matrix or a vector containing the covariates that should always be retained when searching through all possible models.
covariates.test

A matrix or a vector containing all the covariates that should be taken into consideration when searching through all possible models.

nsim

The number of iterations of the stochastic search to run through.

keep

The final number of models to report, along with their Bayes Factors.

q

The maximum number of covariates to be included in each model considered. These covariates will be chosen out of those in covariates.test above.

a

The probability with which the entire set of active coefficients are re-drawn. See page 12 of reference (2) for more details.

model.type

This has to be either "probit" or "reg", specifying the type of model to be fit.

save.every

Specifies how often the models should be written out to a .csv file. This allows a user to monitor progress of models found and to prevent loss of effort in the case of power failure, etc.

out.fname

The name of the .csv file to save models to.

Value

A table containing the top models found is returned. Each row in the table represents a model. A 1 within a row indicates that that covariate, from within the covariates defined by covariates.test, is included in the model. A 0 indicates that that particular covariate is left out of the model. The last column contains the Bayes Factor from comparing the specified model against the intercept-only model.

Author(s)

Gopal, V. and Novelo, L. L. and Casella, G.

Maintainer: Vik Gopal <viknesh.gopal@gmail.com>

References


Examples

n <- 20 # number of observations
p <- 6 # total number of covariates
set.seed(0)

gene_expression <- matrix(runif(n*p)*4,nrow=n,ncol=p)
age <- sample(20:40,n,replace=TRUE)

truth_betavector <- c(-0.1, -.01, 1, -1, rep(0,p+2-4))
design <- cbind(1, age, gene_expression) # sets up the entire design matrix.

# Simulating the z-values and y-values and setting up the data-frame
```
varSelectOBayeslinear <- apply(design, 1, function(xi) {rnorm(n=1, sum(xi * truth_betavector))})
y <- y_tmp[order(y_tmp)]
x <- design[order(y_tmp), -c(1:2)]
n0 <- sum(y<0)
n1 <- n-n0
z <- c(rep(0,n0),rep(1,n1))
mydata <- cbind(z, y, age, x)
colnames(mydata)<-c("z", "y", "age", paste("GE",1:p,sep=""))

# Linear regression function call:
varSelectIP(a=0.2, keep=32, covariates.retain=mydata[,3], model.type="reg",
            q=5, covariates.test=mydata[,4:9], response=mydata[,2], nsim=25)

# NOT RUN
# Probit regression function call:
#varSelectIP(a=0.2, keep=2, covariates.retain=mydata[,3], model.type="probit",
#             q=3, covariates.test=mydata[,4:7], response=mydata[,1], nsim=2)
```

**varSelectOBayeslinear**  
*Objective Bayes Model Selection and inference for Linear Models using mixtures of g-priors*

**Description**

This function carries out a random walk in order to determine the best model, as measured by its posterior probability. The types of models that this function can handle are linear regression models with homoscedastic, independent errors. For full details on the model set-up and the random walk, please refer to the paper listed below.

**Usage**

```
varSelectOBayeslinear(y,X,X0, type = 'IP', prior.par, EFF = TRUE, 
                      model.prior.type = 'Beta-Binomial', model.prior.par = c(1,1),
                      RWFlag = TRUE, N, draws = 10^4, shuf = .2, start, RWpropflag = FALSE,
                      inference = 'both', alpha = 0.05, m.considered = 100)
```

**Arguments**

- `y`  
The vector of response values.
- `X0`  
A matrix containing the covariates that should always be retained when searching through all possible models. These constitute the base model.
- `X`  
A matrix containing all the covariates that should be taken into consideration when searching through all possible models. These constitute the test covariates.
- `type`  
The kind of prior used to compute Bayes' Factors, taking the values 'NP', 'IP', 'ZS', or 'HG' for normal prior, intrinsic prior, Zellner-Siow prior, or hyper g-prior. For the hyper g-prior, the prior on w=1/g is taken to be proportional to w^(a-.5)*(b+w)^(a+1)/2.

---

**Description**

This function carries out a random walk in order to determine the best model, as measured by its posterior probability. The types of models that this function can handle are linear regression models with homoscedastic, independent errors. For full details on the model set-up and the random walk, please refer to the paper listed below.

**Usage**

```
varSelectOBayeslinear(y,X,X0, type = 'IP', prior.par, EFF = TRUE, 
                      model.prior.type = 'Beta-Binomial', model.prior.par = c(1,1),
                      RWFlag = TRUE, N, draws = 10^4, shuf = .2, start, RWpropflag = FALSE,
                      inference = 'both', alpha = 0.05, m.considered = 100)
```

**Arguments**

- `y`  
The vector of response values.
- `X0`  
A matrix containing the covariates that should always be retained when searching through all possible models. These constitute the base model.
- `X`  
A matrix containing all the covariates that should be taken into consideration when searching through all possible models. These constitute the test covariates.
- `type`  
The kind of prior used to compute Bayes' Factors, taking the values 'NP', 'IP', 'ZS', or 'HG' for normal prior, intrinsic prior, Zellner-Siow prior, or hyper g-prior. For the hyper g-prior, the prior on w=1/g is taken to be proportional to w^(a-.5)*(b+w)^(a+1)/2.
prior.par
The parameter of the g-prior, when appropriate. The intrinsic and Zellner-Siow
priors have no parameters. The normal prior has parameter w=1/g, defaulting to
1. The hyper g, has parameter (a,b), defaulting to (2,1).

EFF
A 0-1 variable denoting whether the prior precision for the coefficients of a
model should be scaled by the number of covariates. Defaults to TRUE and a
model with p covariates is scaled by (p+1)/n, if FALSE each model is scaled by
1/n.

model.prior.type
The type of probability distribution for the model space, taking values 'Beta-
Binomial', 'Binomial', or 'Uniform'. The binomial and beta-binomial priors
are formed by putting a Bernoulli(p) prior on inclusion, with the beta-binomial
placing a Beta(a,b) prior on p.

model.prior.par
Parameter for the model space prior. For the binomial case it is 0<p<1, defaulting
to 0.5. For the beta-binomial case it is 0<a,b, defaulting to (1,1).

RWflag
A 0-1 variable denoting whether to do a random walk (TRUE, the default) or
enumerate the models (FALSE).

N.draws
The number of simulations for the random walk, defaults to 10000.

shuf
The percentage of times that the random walk kernel takes an independent draw
from the prior, defaults to 0.2.

start
The starting model for the random walk. Must be either a vector of 0s and 1s
whose length is the number of test covariates, a collapsed string of such 0s and
1s, or a vector of column numbers.

RWpropflag
A 0-1 variable denoting whether to compute posterior probabilities by sample
averages (TRUE) or by renormalization (FALSE, the default).

inference
A variable taking value of 'both' (the default), 'selected', 'averaged', or 'none'
denoting the kind of posterior credible sets to compute for the regression coeffi-
cients. All sets computed are quantile based.

alpha
The level for (1-alpha) credible sets.

m.considered
The number of models to keep in order to for credible sets under model averag-
ing. If 0<m.considered<=1, this denotes keeping the top models that produce at
least m.considered total posterior probability.

Value

gamma
Binary strings of the considered models ordered by their posterior probabilities.

p
The number of covariates (including base covariates) in the considered models.

logPrior
The log of the prior probabilities of the considered models.

R.squared
The coefficients of determination for the considered models.

logB
The log of the Bayes factors of the considered models to the base model.

prob
The posterior probabilities of the considered models.

selected.model
The binary string of the selected model.

selected.model.prob
The posterior probability of the selected model.
The posterior probabilities that each regression coefficient is 0.

A matrix with two columns. The first is the model averaged posterior means of the regression coefficients. The second is the posterior mean conditioned on the selected model.

Quantile based credible sets for the selected model. This is a matrix with rows for the regression coefficients and columns for upper and lower bounds.

Quantile based credible sets under model averaging. This is a list with an element for each regression coefficient. The elements of the list are themselves lists containing the intervals of the credible set and the posterior probabilities of the intervals. The credible sets comprise an open interval and possibly a point mass at 0.

A matrix with two columns. The first column is the residuals from the model averaged mean. The second column is the residuals from the mean conditioned on the selected model.

Andrew Womack
Maintainer: Andrew Womack <ajwomack@ufl.edu>


n <- 20 # number of observations
p <- 6 # total number of covariates
gene_expression <- matrix(runif(n*p)*4,nrow=n,ncol=p)
age <- sample(20:40,n,replace=TRUE)

truth_betavector <- c(-0.1, -.01, 1, -1, rep(0,p+2-4))
design <- cbind(c(1), age, gene_expression) # sets up the entire design matrix.

# Simulating the y-values and setting up the data-frame
y <- apply(design, 1, function(xi){rnorm(n=1, sum(xi * truth_betavector))})
X <- design[, -c(1:2)]
X0<-design[,1:2]

# Linear regression function call:
varSelectOBayeslinear(y,X,X0,RWflag=FALSE,m.con sidered=.99)
Index

*Topic g prior  
  varSelectOBayeslinear, 4

*Topic htest  
  varSelectIP, 2

*Topic linear models  
  varSelectOBayeslinear, 4

*Topic models  
  varSelectIP, 2

*Topic package  
  varSelectIP-package, 1

*Topic regression  
  varSelectIP, 2  
  varSelectOBayeslinear, 4

varSelectIP, 2, 2  
varSelectIP-package, 1  
varSelectOBayeslinear, 2, 4