Package 'BVSNLP'

October 12, 2022

Type Package

Title Bayesian Variable Selection in High Dimensional Settings using Nonlocal Priors

Version 1.1.9

Author Amir Nikooienejad [aut, cre], Valen E. Johnson [ths]

Maintainer Amir Nikooienejad <amir.nikooienejad@gmail.com>

Description Variable/Feature selection in high or ultra-high dimensional settings has gained a lot of attention recently specially in cancer genomic studies. This package provides a Bayesian approach to tackle this problem, where it exploits mixture of point masses at zero and nonlocal priors to improve the performance of variable selection and coefficient estimation. product moment (pMOM) and product inverse moment (piMOM) nonlocal priors are implemented and can be used for the analyses. This package performs variable selection for binary response and survival time response datasets which are widely used in biostatistic and bioinformatics community. Benefiting from parallel computing ability, it reports necessary outcomes of Bayesian variable selection such as Highest Posterior Probability Model (HPPM), Median Probability Model (MPM) and posterior inclusion probability for each of the covariates in the model. The option to use Bayesian Model Averaging (BMA) is also part of this package that can be exploited for predictive power measurements in real datasets.

License GPL (>= 2)

Encoding UTF-8

LazyData true

Depends R (>= 3.1.0)

Imports Rcpp, doParallel, foreach, parallel

Suggests doMPI

LinkingTo Rcpp, RcppArmadillo, RcppEigen, RcppNumerical

RoxygenNote 7.1.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2020-08-28 17:00:03 UTC

R topics documented:

---

bvs *High dimensional Bayesian variable selection using nonlocal priors*

Description

This function performs Bayesian variable selection for high dimensional design matrix using iMOM prior for non-zero coefficients. It also performs adaptive hyperparameter selection for iMOM prior. Cleaning the data in a preprocessing step and before any data analysis is done by user preference. This function is for binary and survival time response datasets. In the former, MCMC is used to search in the model space while for the latter a stochastic search does that job. This function has the option to do all the mentioned tasks in a parallel fashion, exploiting hundreds of CPUs. It is highly recommended to use a cluster for this purpose. This function also supports fixing covariates in variable selection process, thus making them included in the final selected model with probability 1. Categorical variable are also supported by this function as input covariates to the selection process. They need to be well defined factor variables as part of the input data frame. For the output, this function reports necessary measurements that is common in Bayesian variable selection algorithms. They include Highest Posterior Probability model, median probability model and posterior inclusion probability for each of the covariates in the design matrix.

Usage

```
bvs(
  X,
  resp,
  prep = TRUE,
  logT = FALSE,
  fixed_cols = NULL,
  eff_size = 0.5,
  family = c("logistic", "survival"),
  hselect = TRUE,
  nlptype = "piMOM",
  r = 1,
  tau = 0.25,
  niter = 30,
  mod_prior = c("unif", "beta"),
```

```
  inseed = NULL,
  cplng = FALSE,
  ncpu = 4,
  parallel.MPI = FALSE
)
```

Arguments

X	The n times p input data frame containing the covariates in the design matrix. The columns should represent genes and rows represent the observed samples. The column names are used as gene names so they should not be left as NULL. Moreover, the minimum number of columns allowed is 3. The input data frame can also contain categorical covariates that are appropriately defined as factor variables in R.
resp	For logistic regression models it is the binary response vector which could be either numeric or factor variable in R. For the Cox proportional hazard models this is a two column matrix where the first column contains survival time vector and the second column is the censoring status for each observation.
prep	A boolean variable determining if the preprocessing step should be performed on the design matrix or not. That step contains removing columns that have NA's or all their elements are equal to 0, along with standardizing non-binary columns. This step is recommended and thus the default value is TRUE.
logT	A boolean variable determining if log transform should be done on continuous columns before scaling them in the preprocessing step. Note that those columns should not contain any zeros or negative values.
fixed_cols	A vector of indices showing the columns in the input data frame that are not subject to the the selection procedure. These columns are always in the final selected model. Note that if any of these columns contain NA, they will be removed. Moreover, if a categorical variable with k levels is chosen to be fixed, all k-1 dummy variables associated with it will be selected in the final model.
eff_size	This is the expected effect size in the model for a standardized design matrix, which is basically the coefficient value that is expected to occur the most based on some prior knowledge.
family	Determines the type of data analysis. logistic is for binary outcome data where logistic regression modeling is used, whereas survival is for survival outcome data using Cox proportional hazard model.
hselect	A boolean variable indicating whether the automatic procedure for hyperparameter selection should be run or not. The default value is TRUE. Note that in this setting, r is always chosen to be 1.
nlptype	Determines the type of nonlocal prior that is used in the analyses. It can be "piMOM" for product inverse moment prior, or "pMOM" for product moment prior. The default is set to piMOM prior.
r	The paramter r of the iMOM prior, when no automatic procedure for hyperparameter selection is done. As a result, this is relevant only when hselect = FALSE, otherwise it is ignored.

tau	The paramter tau of the iMOM prior, when no automatic procedure for hyper-parameter selection is done. As a result, this is relevant only when hselect = FALSE, otherwise it is ignored.
niter	Number of iterations. For binary response data, this determines the number of MCMC iterations per CPU. For survival response data this is the number of iterations per temperature schedule in the stochastic search algorithm.
mod_prior	Type of prior used for the model space. unif is for a uniform binomial and beta is for a beta binomial prior. In the former case, both hyper parameters in the beta prior are equal to 1, but in the latter case those two hyper parameters are chosen as explained in the reference papers. The default choice for this variable is the uniform prior.
inseed	The input seed for making the parallel processing reproducible. This parameter is ignored in logistic regression models when cplng = FALSE. The default value is NULL which means that each time the search for model space is started from different starting points. In case it is set to a number, it initializes the RNG for the first task and subsequent tasks to get separate substreams.
cplng	This parameter is only used in logistic regression models, and indicating if coupling algorithm for MCMC output should be performed or not.
ncpu	This is the number of cpus used in parallel processing. For logistic regression models this is the number of parallel coupled chains run at the same time. For survival outcome data this is the number of cpus doing stochastic search at the same time to increase th enumber of visited models.
parallel.MPI	A boolean variable determining if MPI is used for parallel processing or not. Note that in order to use this feature, your system should support MPI and Rmpi and doMPI packages should already be installed. The default is set to FALSE but in case your system have the requirements, it is recommended to set this parameter to TRUE as it is more efficient and results in faster run-time.

Value

It returns a list containing different objects that depend on the family of the model and the coupling flag for logistic regression models. The following describes the objects in the output list based on different combinations of those two input arguments.

1) family = logistic && cplng = FALSE

num_vis_models	Number of unique models visited throughout the search of the model space.
max_prob	Maximum unnormalized probability among all visited models
HPM	The indices of the model with highest posterior probability among all visited models, with respect to the columns in the output des_mat. This is not necessarily the same as the input design matrix due to some changes to categorical variables. The names of the selected columns can be checked using gene_names. The corresponding design matrix is also one of the outputs that can be checked in des_mat. If the output is character[0] it means none of the variables of the design matrix is selected in the HPM and HPM contains only the intercept.
beta_hat	The coefficient vector for the selected model. The first component is always for the intercept.

`MPM` The indices of median probability model. According to the paper Barbieri et. al., this is defined to be the model consisting of those variables whose posterior inclusion probability is at least 1/2. The order of columns is similar to that is explained for `HPM`.

`max_prob_vec` A `1000` by `1` vector of unnormalized probabilities of the first 1000 models with highest posterior probability among all visited models. If the total number of visited models is less than 1000, then the length of this vector would be equal to `num_vis_models` . Note that the intercept is always used in calculating the probabilities in this vector.

`max_models` A list containing models corresponding to `max_prob_vec` vector. Each entry of this list contains the indices of covariates for the model with posterior probability reported in the corresponding entry in `max_prob_vec`. The intercept column is not shown in this list as it is present in all of the models.

`inc_probs` A vector of length `p+1` containing the posterior inclusion probability for each covariate in the design matrix. The order of columns is with respect to processed design matrix, `des_mat`.

`nlptype` The type of nonlocal prior used in the analyses.

`des_mat` The design matrix used in the analysis where fixed columns are moved to the beginning of the matrix and if `prep=TRUE`, the columns containing `NA` are all removed. The reported indices in selected models are all with respect to the columns of this matrix.

`gene_names` Names of the genes extracted from the design matrix.

`r` The hyperparameter for iMOM prior density function, calculated using the proposed algorithm for the given dataset.

`tau` The hyperparameter for iMOM prior density function, calculated using the proposed algorithm for the given dataset.

2) `family = logistic && cplng = TRUE`

`cpl_percent` Shows what percentage of pairs of chains are coupled.

`margin_probs` A `k` by `1` vector of marginal probabilities where element `i` shows the maximum marginal probability of the data under the maximum model for the i^{th} pair of chains. `k` is the number of paired chains which is the same as number of CPUs.

`chains` A `k` by `p` binary matrix, where each row is the model for the i^{th} pair of chains. Note that the index of nonzero elements are not necessarily in the same order as the input design matrix, `X`, depending on existence of fixed columns in selection procedure. As a result, always match the indices to the columns of the design matrix that is reported as an output in `des_mat`.

`nlptype` The type of nonlocal prior used in the analyses.

`cpl_flags` A `k` by `1` binary vector, showing which pairs are coupled, (=`1`) and which are not, (= `0`).

`beta_hat` A `k` by `(p+1)` matrix where each row is the estimated coefficient for each modelin the rows of `Chains` variable.

`uniq_models` A list showing unique models with the indices of the included covariates at each model.

freq	Frequency of each of the unique models. It is used to find the highest frquency model.
probs	Unnormalized probability of each of the unique models.
des_mat	The design matrix used in the analysis where fixed columns are moved to the beginning of the matrix and if prep=TRUE, the columns containing NA are all removed. The reported indices in selected models are all with respect to the columns of this matrix.
gene_names	Names of the genes extracted from the design matrix.
r	The hyperparameter for iMOM prior density function, calculated using the proposed algorithm for the given dataset.
tau	The hyperparameter for iMOM prior density function, calculated using the proposed algorithm for the given dataset.

3) family = survival

num_vis_models	Number of visited models during the whole process.
max_prob	The unnormalized probability of the maximum model among all visited models.
HPM	The indices of the model with highest posterior probability among all visited models, with respect to the columns in des_mat. As a result, always look at the names of the selected columns using gene_names. The corresponding design matrix is one of the outputs that can be checked in des_mat.
MPM	The indices of median probability model. According to the paper Barbieri et. al., this is defined to be the model consisting of those variables whose posterior inclusion probability is at least 1/2. The order of columns is similar to that is explained for HPM.
beta_hat	The coefficient vector for the selected model reported in HPM.
max_prob_vec	A 1000 by 1 vector of unnormalized probabilities of the first 1000 models with highest posterior probability among all visited models. If the total number of visited models is less than 1000, then the length of this vector would be equal to num_vis_models .
max_models	A list containing models corresponding to max_prob_vec vector. Each entry of this list contains the indices of covariates for the model with posterior probability reported in the corresponding entry in max_prob_vec.
inc_probs	A p by 1 vector containing the posterior inclusion probability for each covariate in the design matrix. The order of columns is with respect to processed design matrix, des_mat.
nlptype	The type of nonlocal prior used in the analyses.
des_mat	The design matrix used in the analysis where fixed columns are moved to the beginning of the matrix and if prep=TRUE, the columns containing NA are all removed. The reported indices in selected models are all with respect to the columns of this matrix.
start_models	A k by 3 matrix showing the starting model for each worker CPU. Obviously k is equal to the number of CPUs.
gene_names	Names of the genes extracted from the design matrix.

r	The hyperparameter for iMOM prior density function, calculated using the proposed algorithm for the given dataset.
tau	The hyperparameter for iMOM prior density function, calculated using the proposed algorithm for the given dataset.

Author(s)

Amir Nikooienejad

References

Nikooienejad, A., Wang, W., and Johnson, V. E. (2016). Bayesian variable selection for binary outcomes in high dimensional genomic studies using nonlocal priors. Bioinformatics, 32(9), 1338-1345.

Nikooienejad, A., Wang, W., & Johnson, V. E. (2020). Bayesian variable selection for survival data using inverse moment priors. Annals of Applied Statistics, 14(2), 809-828.

Johnson, V. E. (1998). A coupling-regeneration scheme for diagnosing convergence in Markov chain Monte Carlo algorithms. Journal of the American Statistical Association, 93(441), 238-248.

Shin, M., Bhattacharya, A., and Johnson, V. E. (2017). Scalable Bayesian variable selection using nonlocal prior densities in ultrahigh dimensional settings. Statistica Sinica.

Johnson, V. E., and Rossell, D. (2010). On the use of non-local prior densities in Bayesian hypothesis tests. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 72(2), 143-170.

Barbieri, M. M., and Berger, J. O. (2004). Optimal predictive model selection. The annals of statistics, 32(3), 870-897.

See Also

ModProb, CoefEst

Examples

```
### Simulating Logistic Regression Data
n <- 200
p <- 40
set.seed(123)
Sigma <- diag(p)
full <- matrix(c(rep(0.5, p*p)), ncol=p)
Sigma <- full + 0.5*Sigma
cholS <- chol(Sigma)
Beta <- c(-1.9,1.3,2.2)
X <- matrix(rnorm(n*p), ncol=p)
X <- X%*%cholS
beta <- numeric(p)
beta[c(1:length(Beta))] <- Beta
```

```
XB <- X%*%beta
probs <- as.vector(exp(XB)/(1+exp(XB)))
y <- rbinom(n,1,probs)
colnames(X) <- paste("gene_",c(1:p),sep="")
X <- as.data.frame(X)

### Running 'bvs' function without coupling and with hyperparamter selection
### procedure
bout <- bvs(X, y, family = "logistic", nlptype = "piMOM",
            mod_prior = "beta", niter = 50)

### Highest Posterior Model
bout$HPM

### Estimated Coefficients:
bout$beta_hat

### Number of Visited Models:
bout$num_vis_models
```

`CoefEst` *Coefficient estimation for a specific set of covariates*

Description

This function estimates coefficient vector for a given set of covariates in a logistic regression and Cox proportional hazard models. It uses the inverse moment nonlocal prior (iMOM) for non zero coefficients.

Usage

```
CoefEst(
  X,
  resp,
  mod_cols,
  nlptype = "piMOM",
  tau,
  r,
  family = c("logistic", "survival")
)
```

Arguments

`X`	The design matrix. It is assumed that the preprocessing steps have been done on this matrix. It is recommended that to use the output of `PreProcess` function of the package. Also note that the `X` should NOT have a vector of 1's as the first column. If the coefficients of a selected model by `bvs` is to be estimated, it is highly recommended that the design matrix that is one of the outputs of the `bvs` function and is reported as `des_mat` to be used here.

resp	For logistic regression models, this variable is the binary response vector. For Cox proportional hazard models this is a two column matrix where the first column contains the survival time vector and the second column is the censoring status for each observation.
mod_cols	A vector of column indices of the design matrix, representing the selected model.
nlptype	Determines the type of nonlocal prior that is used in the analyses. It can be "piMOM" for product inverse moment prior, or "pMOM" for product moment prior. The default is set to piMOM prior.
tau	Hyperparameter tau of the iMOM prior.
r	Hyperparameter r of the iMOM prior.
family	Determines the type of data analysis. logistic is for binary outcome and logistic regression model whereas, survival represents survival outcomes and the Cox proportional hazard model.

Value

It returns the vector of coefficients for the given model.

Author(s)

Amir Nikooienejad

References

Nikooienejad, A., Wang, W., and Johnson, V. E. (2016). Bayesian variable selection for binary outcomes in high dimensional genomic studies using non-local priors. Bioinformatics, 32(9), 1338-1345.

Nikooienejad, A., Wang, W., & Johnson, V. E. (2020). Bayesian variable selection for survival data using inverse moment priors. Annals of Applied Statistics, 14(2), 809-828.

See Also

ModProb

Examples

```
### Simulating Survival Data
n <- 400
p <- 1000
lambda <- 0.8
cens_rate <- 0.27
set.seed(123)
Sigma <- diag(p)
full <- matrix(c(rep(0.5, p*p)), ncol=p)
Sigma <- full + 0.5*Sigma
cholS <- chol(Sigma)
Beta <- c(-1.8, 1.2, -1.7, 1.4, -1.4, 1.3)
X = matrix(rnorm(n*p), ncol=p)
```

```
X = X%*%cholS
X <- scale(X)
beta <- numeric(p)
beta[c(1:length(Beta))] <- Beta
XB <- X%*%beta
uvector <- -log(runif(n));
times <- uvector/(lambda*exp(XB))
cens_time <- quantile(times,1-cens_rate)
status <- as.numeric(times < cens_time)
TS <- cbind(times,status)

### Estimating coeffcients of the true model and an arbitrary hyper
### parameter for the iMOM prior density
mod <- c(1:6)
coef <- CoefEst(X, TS, mod, tau = 1.8, r = 2, family = "survival")
coef
```

`cox_bvs` *Non-parallel version of Bayesian variable selector for survival data using nonlocal priors*

Description

This function performs Bayesian variable selection for survival data in a non-parallel fashion. It runs modified S5 algorithm to search the model space but since this is only on one CPU, the number of visited models will not be large and therefore is NOT recommended for high dimensional datasets. This function is called by `bvs` function in a parllel fashion and therefore that function is recommended to be used.

Usage

```
cox_bvs(exmat, cur_cols, nf, tau, r, nlptype, a, b, d, L, J, temps)
```

Arguments

`exmat`	An extended matrix where the first two columns are survival times and status, respectively and the rest is the design matrix which is produced by `PreProcess` function.
`cur_cols`	A vector containing indices of the initial model for variable selector to start the S5 algorithm from. Note that the first `nf` indices are `1` to `nf` where `nf` is the number of fixed covariates that do not enter the selection procedure.
`nf`	The number of fixed covariates that do not enter the selection procedure.
`tau`	The paramter `tau` of the iMOM prior.
`r`	The paramter `r` of the iMOM prior.
`nlptype`	Determines the type of nonlocal prior that is used in the analyses. `0` is for piMOM and `1` is for pMOM.

a	The first parameter in beta distribution used as prior on model size. This parameter is equal to 1 when uinform-binomial prior is used.
b	The second paramter in beta distribution used as prior on model size. This parameter is equal to 1 when uinform-binomial prior is used.
d	This is the number of candidate covariates picked from top variables with highest utility function value and used in S5 algorithm.
L	Number of temperatures in S5 algorithm.
J	Number of iterations at each temperature in S5 algorithm.
temps	Vector of temperatuers used in S5 algorithm.

Value

It returns a list containing following objects:

max_model	A 1 by p binary vector showing the selected model with maximum probability. 1 means a specific variable is selected.
hash_key	A column vector indicating the generated key for each model that is used to track visited models and growing dictionary.
max_prob	The unnormalized probability of the model with highest posterior probability.
all_probs	A vector containing unnormalized probabilities of all visited models.
vis_covs_list	A list containing the covariates in each visited model in the stochastic search process.

Author(s)

Amir Nikooienejad

References

Nikooienejad, A., Wang, W., and Johnson, V. E. (2017). Bayesian Variable Selection in High Dimensional Survival Time Cancer Genomic Datasets using Nonlocal Priors. arXiv preprint, arXiv:1712.02964.

Shin, M., Bhattacharya, A., and Johnson, V. E. (2017). Scalable Bayesian variable selection using nonlocal prior densities in ultrahigh dimensional settings. Statistica Sinica.

Johnson, V. E., and Rossell, D. (2010). On the use of non-local prior densities in Bayesian hypothesis tests. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 72(2), 143-170.

See Also

bvs

Examples

```
### Initializing the parameters
n <- 100
p <- 40
set.seed(123)
Sigma <- diag(p)
full <- matrix(c(rep(0.5, p*p)), ncol=p)
Sigma <- full + 0.5*Sigma
cholS <- chol(Sigma)
Beta <- c(-1.8, 1.2, -1.7, 1.4, -1.4, 1.3)
X = matrix(rnorm(n*p), ncol=p)
X = X%*%cholS
X <- scale(X)
beta <- numeric(p)
beta[c(1:length(Beta))] <- Beta
XB <- X%*%beta
sur_times <- rexp(n,exp(XB))
cens_times <- rexp(n,0.2)
times <- pmin(sur_times,cens_times)
status <- as.numeric(sur_times <= cens_times)
exmat <- cbind(times,status,X)
L <- 10; J <- 10
d <- 2 * ceiling(log(p))
temps <- seq(3, 1, length.out = L)
tau <- 0.5; r <- 1; a <- 6; b <- p-a
nlptype <- 0 ### PiMOM nonlocal prior
cur_cols <- c(1,2,3) ### Starting model for the search algorithm
nf <- 0 ### No fixed columns

### Running the Function
coxout <- cox_bvs(exmat,cur_cols,nf,tau,r,nlptype,a,b,d,L,J,temps)

### The number of visited model for this specific run:
length(coxout$hash_key)


### The selected model:
which(coxout$max_model>0)

### The unnormalized probability of the selected model:
coxout$max_prob
```

`HyperSelect` *Hyperparameter selection for iMOM prior density*

Description

This function finds data specific hyperparameters for inverse moment prior density so that the overlap between the iMOM prior and null MLE density is $1/\sqrt{p}$. In this algorithm `r` is always chosen to be equal to 1 and `tau` is found based on the mentioned overlap.

Usage

```
HyperSelect(
  X,
  resp,
  eff_size = 0.7,
  nlptype = "piMOM",
  iter = 10000,
  mod_prior = c("unif", "beta"),
  family = c("logistic", "survival")
)
```

Arguments

X	The design matrix. NA's should be removed and columns be scaled. It is recommended that the PreProcess function is run first and its output used for this argument. The columns are genes and rows represent the observations. The column names are used as gene names.
resp	For logistic regression models, it is the binary response vector. For Cox proportional hazard model, this is a two column matrix where the first column contains survival time vector and the second column is the censoring status for each observation.
eff_size	This is the expected effect size in the model for a standardized design matrix, which is basically the coefficient value that is expected to occur the most based on some prior knowledge.
nlptype	Determines the type of nonlocal prior that is used in the analyses. It can be "piMOM" for product inverse moment prior, or "pMOM" for product moment prior. The default is set to piMOM prior.
iter	The number of iteration needed to simulate from null model in order to approximate the null MLE density.
mod_prior	Type of prior used for model space. uniform is for uniform binomial and beta is for beta binomial prior. In the former case, both hyper parameters in the beta prior are equal to 1 but in the latter case those two hyper parameters are chosen as explained in the reference papers.
family	Determines the type of data analysis. logistic is for binary outcome data where logistic regression modeling is used whereas survival is for survival outcome data using Cox proportional hazard model.

Value

It returns a list having following object:

tau	The hyperparameter for iMOM prior density function, calculated using the proposed algorithm for the given dataset.

Author(s)

Amir Nikooienejad

References

Nikooienejad, A., Wang, W., and Johnson, V. E. (2016). Bayesian variable selection for binary outcomes in high dimensional genomic studies using nonlocal priors. Bioinformatics, 32(9), 1338-1345.

Johnson, V. E., and Rossell, D. (2010). On the use of nonlocal prior densities in Bayesian hypothesis tests. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 72(2), 143-170.

Examples

```
### Simulating Logistic Regression Data
n <- 20
p <- 100
set.seed(321)
Sigma <- diag(p)
full <- matrix(c(rep(0.5, p * p)), ncol = p)
Sigma <- full + 0.5 * Sigma
cholS <- chol(Sigma)
Beta <- c(1, 1.8, 2.5)
X = matrix(rnorm(n*p, 1, 2), ncol = p)
X = X%*%cholS
beta <- numeric(p)
beta[c(1:length(Beta))] <- Beta
XB <- X%*%beta
probs <- as.vector(exp(XB)/(1+exp(XB)))
y <- rbinom(n,1,probs)
colnames(X) <- paste("gene_",c(1:p),sep="")
Xout <- PreProcess(X)
XX <- Xout$X
hparam <- HyperSelect(XX, y, iter = 1000, mod_prior = "beta",
                      family = "logistic")

hparam$tau
```

`logreg_bvs` *Non-parallel version of Bayesian variable selector for logistic regression data using nonlocal priors*

Description

This function performs Bayesian variable selection for logistic regression data in a non-parallel fashion. It does not contain any pre-processing step or variable initialization. Moreover it does not have the feature to be run in parallel for performing the coupling algorithm. Therefore in general, it is NOT recommended to be used unless the user knows how to initialize all the parameters. However, this function is called by `bvs` function, the recommended way to run Bayesian variable selection for such datasets.

Usage

```
logreg_bvs(
  exmat,
  chain1,
  nf,
  tau,
  r,
  nlptype,
  a,
  b,
  in_cons,
  loopcnt,
  cplng,
  chain2
)
```

Arguments

exmat	An extended matrix where the first column is binary resonse vector and the rest is the design matrix which has its first column all 1 to account for the intercept in the model and is the output of PreProcess code where the fixed columns are moved to the beginning.
chain1	The first chain or initial model where the MCMC algorithm starts from. Note that the first nf+1 elements are 1 where nf is the number of fixed covariates that do not enter the selection procedure and 1 is for the intercept.
nf	The number of fixed covariates that do not enter the selection procedure.
tau	The paramter tau of the iMOM prior.
r	The paramter r of the iMOM prior.
nlptype	Determines the type of nonlocal prior that is used in the analyses. 0 is for piMOM and 1 is for pMOM.
a	The first parameter in beta distribution used as prior on model size. This parameter is equal to 1 when uinform-binomial prior is used.
b	The second paramter in beta distribution used as prior on model size. This parameter is equal to 1 when uinform-binomial prior is used.
in_cons	The average model size. This value under certain conditions and when p is large, is equal to parameter a of the beta-binomial prior.
loopcnt	Number of iterations for MCMC procedure.
cplng	A boolean variable indicating the coupling algorithm to be performed or not.
chain2	Second chain or model for starting the MCMC procedure. This parameter is only used when cplng=TRUE. Thus, it could be simply set to chain1 when it is not used.

Value

It returns a list containing following objects:

max_chain	A 1 by p+1 binary vector showing the selected model with maximum probability. 1 means a specific variable is selected. The first variable is always the intercept.
beta_hat	The coefficient vector for the selected model. The first one is always for the intercept.
max_prop	The unnormalized probability of the model with highest posterior probability.
num_iterations	The number of MCMC iterations that are executed. This is used when cplng=TRUE to check whether the total designated MCMC iterations were used or two chains are coupled sooner than that.
cplng_flag	This is used when cplng=TRUE and indicates whether two chains are coupled or not.
num_vis_models	Number of visited models in search for the highest probability model. This contains redundant models too and is not the number of unique models.
hash_key	This is only used when cplng = FALSE. This is a vector containing real numbers uniquely assigned to each model for distinguishing them.
hash_prob	This is only used when cplng = FALSE. This is a vector of probabilities for each visited model.
vis_covs	This is only used when cplng = FALSE. This is a list where each element contains indices of covariates for each visited model.

Author(s)

Amir Nikooienejad

References

Nikooienejad, A., Wang, W., and Johnson, V. E. (2016). Bayesian variable selection for binary outcomes in high dimensional genomic studies using nonlocal priors. Bioinformatics, 32(9), 1338-1345.

Nikooienejad, A., Wang, W., and Johnson, V. E. (2017). Bayesian Variable Selection in High Dimensional Survival Time Cancer Genomic Datasets using Nonlocal Priors. arXiv preprint, arXiv:1712.02964.

Johnson, V. E., and Rossell, D. (2010). On the use of non-local prior densities in Bayesian hypothesis tests. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 72(2), 143-170.

Johnson, V. E. (1998). A coupling-regeneration scheme for diagnosing convergence in Markov chain Monte Carlo algorithms. Journal of the American Statistical Association, 93(441), 238-248.

See Also

bvs

Examples

```
### Initializing parameters
n <- 200
```

```
p <- 40
set.seed(123)
Sigma <- diag(p)
full <- matrix(c(rep(0.5, p*p)), ncol=p)
Sigma <- full + 0.5*Sigma
cholS <- chol(Sigma)
Beta <- c(-1.7,1.8,2.5)
X <- matrix(rnorm(n*p), ncol=p)
X <- X%*%cholS
colnames(X) <- paste("gene_",c(1:p),sep="")
beta <- numeric(p)
beta[c(1:length(Beta))] <- Beta
XB <- X%*%beta
probs <- as.vector(exp(XB)/(1+exp(XB)))
y <- rbinom(n,1,probs)
exmat <- cbind(y,X)
tau <- 0.5; r <- 1; a <- 3; b <- p-a; in_cons <- a;
loopcnt <- 100; cplng <- FALSE;
initProb <- in_cons/p

### Initializing Chains
schain <- p
while (schain > in_cons || schain == 0) {
chain1 <- rbinom(p, 1, initProb)
 schain <- sum(chain1)
}
chain1 <- as.numeric(c(1, chain1))
chain2 <- chain1
nlptype <- 0 ## PiMOM nonlocal prior
nf <- 0 ### No fixed columns

### Running the function
bvsout <- logreg_bvs(exmat,chain1,nf,tau,r,nlptype,a,b,in_cons,loopcnt,cplng,chain2)

### Number of visited models for this specific run:
bvsout$num_vis_models

### The selected model:
which(bvsout$max_chain > 0)

### Estimated coefficients:
bvsout$beta_hat

### The unnormalized probability of the selected model:
bvsout$max_prob
```

ModProb *Logarithm of unnormalized probability of a given model*

Description

This function calculates the logarithm of unnormalized probability of a given set of covariates for both survival and binary response data. It uses the inverse moment nonlocal prior (iMOM) for non zero coefficients and beta binomial prior for the model space.

Usage

```
ModProb(
  X,
  resp,
  mod_cols,
  nlptype = "piMOM",
  tau,
  r,
  a,
  b,
  family = c("logistic", "survival")
)
```

Arguments

X	The design matrix. It is assumed that the design matrix has standardized columns. It is recommended that to use the output of PreProcess function of the package.
resp	For logistic regression models, this variable is the binary response vector. For Cox proportional hazard models this is a two column matrix where the first column contains the survival time vector and the second column is the censoring status for each observation.
mod_cols	A vector of column indices of the design matrix, representing the model.
nlptype	Determines the type of nonlocal prior that is used in the analyses. It can be "piMOM" for product inverse moment prior, or "pMOM" for product moment prior. The default is set to piMOM prior.
tau	Hyperparameter tau of the iMOM prior.
r	Hyperparameter r of the iMOM prior.
a	First parameter in the beta binomial prior.
b	Second parameter in the beta binomial prior.
family	Determines the type of data analysis. logistic is for binary outcome and logistic regression model whereas, survival represents survival outcomes and the Cox proportional hazard model.

Value

It returns the unnormalized probability for the selected model.

Author(s)

Amir Nikooienejad

References

Nikooienejad, A., Wang, W., and Johnson, V. E. (2016). Bayesian variable selection for binary outcomes in high dimensional genomic studies using nonlocal priors. Bioinformatics, 32(9), 1338-1345.

Nikooienejad, A., Wang, W., & Johnson, V. E. (2020). Bayesian variable selection for survival data using inverse moment priors. Annals of Applied Statistics, 14(2), 809-828.

See Also

CoefEst

Examples

```
### Simulating Logistic Regression Data
n <- 400
p <- 1000
set.seed(123)
Sigma <- diag(p)
full <- matrix(c(rep(0.5, p*p)), ncol=p)
Sigma <- full + 0.5*Sigma
cholS <- chol(Sigma)
Beta <- c(1,1.8,2.5)
X = matrix(rnorm(n*p), ncol=p)
X = X%*%cholS
beta <- numeric(p)
beta[c(1:length(Beta))] <- Beta
XB <- X%*%beta
probs <- as.vector(exp(XB)/(1+exp(XB)))
y <- rbinom(n,1,probs)

### Calling the function for a subset of the true model, with an arbitrary
### parameters for prior densities
mod <- c(1:3)
Mprob <- ModProb(X, y, mod, tau = 0.7, r = 1, a = 7, b = 993,
                 family = "logistic")

Mprob
```

---

predBMA *Predictive accuracy measurement using Bayesian Model Averaging*

Description

This function is used for predictive accuracy measurement of the selected models using Bayesian Model Averaging methodology. The Occam's window with cut out threshold of `thr` is used. That means only models that have posterior probability of at least `thr` * posteior probability of the highest posterior probability model are considered in model averaging. For survival time response datasets, the predictive Area Under Curve (AUC) at each given time point is computed as the output. In this

case, the predictive AUC is obtained using Uno's method for observations in the test set. For binary outcome data, only one AUC is reported which is from the ROC computed on the test set. The training set is used to find the selected model and relevant probabilities.

Usage

```
predBMA(
  bvsobj,
  X,
  resp,
  prep,
  logT,
  nlptype = "piMOM",
  train_idx,
  test_idx,
  thr = 0.05,
  times = NULL,
  family = c("logistic", "survival")
)
```

Arguments

bvsobj	An object that is generated by bvs function. It is the output of the Bayesian variable selection procedure.
X	The same n times p data frame initially used in bvs function. This is the full data frame containing all test and train data.
resp	For logistic regression models, this variable is the binary response vector. For the Cox proportional hazard models this is a two column matrix where the first column contains survival times and the second column is the censoring status for each observation. Note that for survival times, the time section of this variable should be in the same scale and unit (year, days, etc.) as times variable for which the AUC has to be computed.
prep	A boolean variable determining if the preprocessing step has been done on the original data frame in bvs function. This should have the same value as prep variable in the bvs function.
logT	A boolean variable determining if log transform should has been done on continuous columns of the data frame in bvs function. This should have the same value as logT variable in the bvs function.
nlptype	Determines the type of nonlocal prior that is used in the analyses. It can be "piMOM" for product inverse moment prior, or "pMOM" for product moment prior. The default is set to piMOM prior.
train_idx	An integer vector containing the indices of the training set.
test_idx	An integer vector containing the indices of the test set. The set of observations that prediction will be performed on.
thr	The threshold used for Occam's window as explained in the description. The default value for this variable is 0.05.

`times` A vector of times at which predictive AUC is to be computed. This input is only used for prediction in survival data analysis.

`family` Determines the type of data analysis. `logistic` is for binary outcome and logistic regression model whereas, `survival` represents survival outcomes and the Cox proportional hazard model.

Value

The output is different based on the family for the anlysis of data **1)** `family = logistic` The output is a list with the two following objects:

`auc` This is the area under the ROC curve after Bayesian model averaging is used to obtain ROC for the test data.

`roc_curve` This is a two column matrix representing points on the ROC curve and can be used to plot the curve. The first column is FPR and the second column is TPR which represent x-axis and y-axis in the ROC curve, respectively.

2) `family = survival`

`auc` A vector with the same length as `times` variable showing predictive area under the curve at each given time point using Bayesian Model averaging.

Author(s)

Amir Nikooienejad

References

Raftery, A. E., Madigan, D., & Hoeting, J. A. (1997). Bayesian model averaging for linear regression models. Journal of the American Statistical Association, 92(437), 179-191.

Nikooienejad, A., Wang, W., & Johnson, V. E. (2020). Bayesian variable selection for survival data using inverse moment priors. Annals of Applied Statistics, 14(2), 809-828.

Uno, H., Cai, T., Tian, L., & Wei, L. J. (2007). Evaluating prediction rules for t-year survivors with censored regression models. Journal of the American Statistical Association, 102(478), 527-537.

Examples

```
### Simulating Logistic Regression Data
n <- 200
p <- 40
set.seed(123)
Sigma <- diag(p)
full <- matrix(c(rep(0.5, p*p)), ncol=p)
Sigma <- full + 0.5*Sigma
cholS <- chol(Sigma)
Beta <- c(-1.7,1.8,2.5)
X <- matrix(rnorm(n*p), ncol=p)
X <- X%*%cholS
```

```
colnames(X) <- c(paste("gene_",c(1:p),sep=""))
beta <- numeric(p)
beta[c(1:length(Beta))] <- Beta
Xout <- PreProcess(X)
X <- Xout$X
XB <- X%*%beta
probs <- as.vector(exp(XB)/(1+exp(XB)))
y <- rbinom(n,1,probs)
X <- as.data.frame(X)
train_idx <- sample(1:n,0.8*n)
test_idx <- setdiff(1:n,train_idx)
X_train <- X[train_idx,]
y_train <- y[train_idx]
bout <- bvs(X_train, y_train, prep=FALSE, family = "logistic",
           mod_prior = "beta",niter = 50)
BMAout <- predBMA(bout, X, y, prep = FALSE, logT = FALSE,
                  train_idx = train_idx, test_idx = test_idx,
                  family="logistic")
### AUC for the prediction:
BMAout$auc

### Plotting ROC Curve
roc <- BMAout$roc_curve
plot(roc,lwd=2,type='l',col='blue')
```

PreProcess *Preprocessing the design matrix, preparing it for variable selection procedure*

Description

This function preprocesses the design matrix by removing columns that contain NA's or are all zero. It also standardizes non-binary columns to have mean zero and variance one. The user has the choice of log transforming continuous covariates before scaling them.

Usage

```
PreProcess(X, logT = FALSE)
```

Arguments

X	The n times p design matrix. The columns should represent genes and rows represent the observations. The column names are used as gene names so they should not be left as NULL. Note that the input matrix X should NOT contain vector of 1's representing the intercept.
logT	A boolean variable determining if log transform should be done on continuous columns before scaling them. Note that those columns should not contain any zeros or negative values.

Value

It returns a list having the following objects:

X	The filtered design matrix which can be used in variable selection procedure. Binary columns are moved to the end of the design matrix.
gnames	Gene names read from the column names of the filtered design matrix.

Author(s)

Amir Nikooienejad

Examples

```
### Constructing a synthetic design matrix for the purpose of preprocessing
### imposing columns with different scales
n <- 40
p1 <- 50
p2 <- 150
p <- p1 + p2
X1 <- matrix(rnorm(n*p1, 1, 2), ncol = p1)
X2 <- matrix(rnorm(n*p2), ncol = p2)
X <- cbind(X1, X2)

### putting NA elements in the matrix
X[3,85] <- NA
X[25,85] <- NA
X[35,43] <- NA
X[15,128] <- NA
colnames(X) <- paste("gene_",c(1:p),sep="")

### Running the function. Note the intercept column that is added as the
### first column in the "logistic" family
Xout <- PreProcess(X)
dim(Xout$X)[2] == (p + 1) ## 1 is added because intercept column is included
## This is FALSE because of the removal of columns with NA elements
```

Index