Package ‘glmvsd’

January 7, 2016

Type Package

Title Variable Selection Deviation Measures and Instability Tests for High-Dimensional Generalized Linear Models

Version 1.4

Date 2016-01-06

Author Ying Nan <nanx0006@gmail.com>, Yanjia Yu <yuxxx748@umn.edu>, Yuhong Yang <yyang@stat.umn.edu>, Yi Yang <yi.yang6@mcgill.ca>

Maintainer Yi Yang <yi.yang6@mcgill.ca>

Depends stats, glmnet, ncvreg, MASS, parallel, brglm

Description Variable selection deviation (VSD) measures and instability tests for high-dimensional model selection methods such as LASSO, SCAD and MCP, etc., to decide whether the sparse patterns identified by those methods are reliable.

License GPL-2

URL https://github.com/emeryyi/glmvsd

Date/Publication 2016-01-07 13:55:37

NeedsCompilation no

Repository CRAN

R topics documented:

glmvsd *Variable Selection Deviation (VSD)*

Description

The package calculate the variable selection deviation (VSD) to measure the uncertainty of the selection in terms of inclusion of predictors in the model.

Usage

```
glmvsd(x, y, n_train = ceiling(n/2), no_rep = 100,
                  n_train_bound = n_train - 2, n_bound = n - 2,
                  model_check, psi = 1, family = c("gaussian",
                  "binomial"), method = c("union", "customize"),
                  candidate_models, weight_type = c("BIC", "AIC",
                  "ARM"), prior = TRUE, reduce_bias = FALSE)
```

Arguments

x	Matrix of predictors.
y	Response variable.
n_train	Size of training set when the weight function is ARM or ARM with prior. The default value is n_train=ceiling(n/2).
no_rep	Number of replications when the weight function is ARM and ARM with prior. The default value is no_rep=100.
n_train_bound	When computing the weights using "ARM", the candidate models with the size larger than n_train_bound will be dropped. The default value is n_train-2.
n_bound	When computing the weights using "AIC" or "BIC", the candidate models with the size larger than n_train_bound will be dropped. The default value is n-2.
model_check	The index of the model to be assessed by calculating the VSD measures.
psi	A positive number to control the improvement of the prior weight. The default value is 1.
family	Choose the family for GLM models. So far only gaussian, binomial and tweedie are implemented. The default is gaussian.
method	User chooses one of the union and customize. If method=="union", then the program automatically provides the candidate models as a union of solution paths of Lasso, SCAD, and MCP; If method="customize", the user must provide their own set of candidate models in the input argument candidate_models as a matrix, each row of which is a 0/1 index vector representing whether each variable is included/excluded in the model.
candidate_models	Only available when method="customize". It is a matrix of candidate models, each row of which is a 0/1 index vector representing whether each variable is included/excluded in the model.

`weight_type` Options for computing weights for VSD measure. User chooses one of the `ARM`, `AIC` and `BIC`. The default is `BIC`.

`prior` Whether use prior in the weight function. The default is `TRUE`.

`reduce_bias` If the binomial model is used, occasionally the algorithm might has convergence issue when the problem of so-called complete separation or quasi-complete separation happens. Users can set `reduce_bias=TRUE` to solve the issue. The algorithm will use an adjusted-score approach when ftting the binomial model for computing the weights. This method is developed in Firth, D. (1993). Bias reduction of maximum likelihood estimates. Biometrika 80, 27-38.

Details

See Reference section.

Value

A "glmvsd" object is retured. The components are:

`VSD` Variable selection deviation (VSD) value.

`VSD_minus` The lower VSD value of `model_check`, representing the number of predictors in the model (`model_check`) not quite justified at the present sample size.

`VSD_plus` The upper VSD value of `model_check` model, representing the number of predictors missed by the model (`model_check`).

`weight` The weight for each candidate model.

`DIFF` Counting the variable differences between candidate models and `model_check`.

`candidate_models_cleaned`
Cleaned candidate models: the duplicated candidate models are cleaned; When computing VSD weights using AIC and BIC, the models with more than n-2 variables are removed (n is the number of observaitons); When computing VSD weights using ARM, the models with more than n_train-2 variables are removed (n_train is the number of training observations).

References

Nan, Y. and Yang, Y. (2013), "Variable Selection Diagnostics Measures for High-dimensional Regression," *Journal of Computational and Graphical Statistics*, 23:3, 636-656.
http://dx.doi.org/10.1080/10618600.2013.829780
BugReport: https://github.com/emeryyi/glmvsd

Examples

```
# REGRESSION CASE

# generate simulation data
n <- 50
p <- 8
beta <- c(3,1.5,0,0,2,0,0,0)
```

```
sigma <- matrix(0,p,p)
for(i in 1:p){
   for(j in 1:p) sigma[i,j] <- 0.5^abs(i-j)
}
x <- mvrnorm(n, rep(0,p), sigma)
e <- rnorm(n)
y <- x %*% beta + e

# user provide a model to be checked
model_check <- c(0,1,1,1,0,0,0,1)

# compute VSD for model_check using ARM with prior
v_ARM <- glmvsd(x, y, n_train = ceiling(n/2),
no_rep=50, model_check = model_check, psi=1,
family = ”gaussian”, method = ”union”,
weight_type = ”ARM”, prior = TRUE)

# compute VSD for model_check using AIC
v_AIC <- glmvsd(x, y,
model_check = model_check,
family = ”gaussian”, method = ”union”,
weight_type = ”AIC”, prior = TRUE)

# compute VSD for model_check using BIC
v_BIC <- glmvsd(x, y,
model_check = model_check,
family = ”gaussian”, method = ”union”,
weight_type = ”BIC”, prior = TRUE)

# user supplied candidate models
candidate_models = rbind(c(0,0,0,0,0,0,0,1),
c(0,1,0,0,0,0,0,1), c(0,1,1,1,0,0,0,1),
c(0,1,1,0,0,0,0,1), c(1,1,0,1,1,0,0,0),
c(1,1,0,0,1,0,0,0))

v1_BIC <- glmvsd(x, y,
model_check = model_check, psi=1,
family = ”gaussian”,
method = ”customize”,
candidate_models = candidate_models,
weight_type = ”BIC”, prior = TRUE)

# CLASSIFICATION CASE

# generate simulation data
n = 300
p = 8
b <- c(1,1,1,-3*sqrt(2)/2)
x=matrix(rnorm(n*p, mean=0, sd=1), n, p)
feta=x[, 1:4]%*%b
fprob=exp(feta)/(1+exp(feta))
y=rbinom(n, 1, fprob)
```

```
# user provide a model to be checked
model_check <- c(0,1,1,1,0,0,0,1)

# compute VSD for model_check using BIC with prior
b_BIC <- glmvsd(x, y, n_train = ceiling(n/2),
family = "binomial",
no_rep=50, model_check = model_check, psi=1,
method = "union", weight_type = "BIC",
prior = TRUE)

candidate_models =
rbind(c(0,0,0,0,0,0,0,1),
c(0,1,0,0,0,0,0,1),
c(1,1,1,1,0,0,0,0),
c(0,1,1,0,0,0,0,1),
c(1,1,0,1,1,0,0,0),
c(1,1,0,0,1,0,0,0),
c(0,0,0,0,0,0,0,0),
c(1,1,1,1,1,0,0,0))

# compute VSD for model_check using AIC
# user supplied candidate models
b_AIC <- glmvsd(x, y,
family = "binomial",
model_check = model_check, psi=1,
method = "customize",
candidate_models = candidate_models,
weight_type = "AIC")
```

stability.test *Instability tests*

Description

This function calculate the sequential, parametric bootstrap and perturbation instability measures for linear regression with Lasso, SCAD and MCP penalty.

Usage

```
stability.test(x, y,
method = c("seq", "bs", "perturb"),
penalty = c("LASSO", "SCAD", "MCP"),
nrep = 50, remove = 0.2, tau = 0.5, nfolds = 5,
family=c("gaussian","binomial"))
```

Arguments

x	Matrix of predictors.
y	Response variable.

`method` Type of instability measures. `seq` = sequential instability, `bs` = parametric bootstrap instability, and `perturb` = perturbation instability.

`penalty` Penalty function.

`nrep` Number of repetition for calculating instability, default is 50.

`remove` The portion of observation to be removed when the sequential instability is calculated, default is 0.2.

`tau` The size of perturbation when perturbation instability is calculated. The range of `tau` is (0,1), default is 0.5

`nfolds` number of folds - default is 5.

`family` Choose the family for the instability test. So far only `gaussian`, `binomial` and `tweedie` are implemented. The default is `gaussian`.

Details

See Reference section.

Value

Return the instability index according to the type of instability measures.

References

Nan, Y. and Yang, Y. (2013), "Variable Selection Diagnostics Measures for High-dimensional Regression," *Journal of Computational and Graphical Statistics*, 23:3, 636-656.
http://dx.doi.org/10.1080/10618600.2013.829780
BugReport: https://github.com/emeryyi/glmvsd

Examples

```
# generate simulation data
n <- 50
p <- 8
beta<-c(2.5,1.5,0.5,rep(0,5))
sigma<-matrix(0,p,p)
for(i in 1:p){
   for(j in 1:p) sigma[i,j] <- 0.5^abs(i-j)
}
x <- mvrnorm(n, rep(0,p), sigma)
e <- rnorm(n)
y <- x %*% beta + e

ins_seq <- stability.test(x, y, method = "seq",
penalty = "SCAD", nrep = 20,
remove = 0.1, tau = 0.2, nfolds = 5)
```

Index