Package 'groupsubsetselection'

April 3, 2017

Version 1.0.3

Date 2017-03-31

Title Group Subset Selection

Author Yi Guo [aut, cre],
Mark Berman [aut]

Maintainer Yi Guo <y.guo@westernsydney.edu.au>

Description Group subset selection for linear regression models is provided in this package. Given response variable, and explanatory variables, which are organised in groups, group subset selection selects a small number of groups to explain response variable linearly using least squares.

License GPL-2

NeedsCompilation yes

Repository CRAN

Date/Publication 2017-04-03 15:31:06 UTC

R topics documented:

`groupsubsetselection` *Group Subset Selection for Linear Regression*

Description

Group subset selection (GSS) carries out exhaustive subset selection in a group fashion, i.e. the variables are grouped and the groups of variables stay or leave the model altogether. It is designed for linear regression model using least squares. It generalises variable subset selection (VSS) , which can be carried out using the R package leaps in two ways. First, it fits groups of variables rather than single variables. Second, it optionally allows the specification of lower bounds on specified coefficients.

Usage

```
groupsubsetselection(y,x,nvarmax,nbest,nb,consind,conslb,ngv=rep(2,30))
```

Arguments

y	dependent variable
x	explanatory variables, data in column in groups
nvarmax	maximum size of combinations of groups to be searched
nbest	number of best combinations of groups to be searched
nb	number of fixed variables
consind	the indicator vector of coefficient constraints
conslb	the lower bounds of the coefficients
ngv	a vector giving the number of variables in each group

Details

GSS carries out exhaustive group subset selection for linear regression. To speed up the computation, QR and local QR updates have been employed. Therefore, it is suitable for middle scale problems, say, select 5 out of 100 groups with several variables in each group under the linear regression model. The speed is affected by the dimensionality of the data, total number of variables, the maximum size of the combinations and how many best combinations are to be found.

There is a coefficient screening embedded in this implementation, i.e. when the regression coefficients of one combination of groups are obtained, they will be compared to the given lower bounds for all variables in each group (in conslb). If any one of the variables in a group is lower than its corresponding lower bound, then this group will be discarded. This is particularly useful for screening some combinations with negative coefficients, namely group nonnegative screening.

There may be nb fixed variables that will be included in the model such as a constant. These nb variables must be located in the first nb columns of x and will be tested with any combinations.

The format of the explanatory variables in x:

x = [f1, f2, ... , fnb, g11, g12, ... , g1n_1, g21, g22, ... , g2n_2, ...]

i.e. fixed variables are in the first nb columns, then group1, group 2, and so on. There should be sum(ngv) non-fixed variables altogether, where ngv = [n_1, n_2, ...]. Therefore x should contain nb+sum(ngv) columns.

The format of the constraint vectors consind and conslb: Both these vectors have length nb+sum(ngv), i.e. the number of columns in x. consind[i] equals 1, if variable i has a lower bound, and equals 0, otherwise. conslb[i] is the lower bound for variable i if consind[i] = 1. It can be any number if consind[i] = 0.

An example:

consind = [0 0 0 0 0 0 1 1 0 1 0 0 1 1 .....],

conslb = [0 1 0 0 0 0 0.1 0.2 0 -0.1 0 0 0.02 -0.05 .....]

Variables 1 to 6, 9, 11 and 12 are unconstrained. Variables 7, 8, 10, 13 and 14 have lower bounds (>=) 0.1, 0.2, -0.1, 0.02 and -0.05 respectively. Note that conslb[2] = 1 is not used because consind[2] = 0. The coefficient of every variable can be constrained separately and the lower bounds can also be specified separately.

Output format

groups = [Best group, 2nd best group, nbest-th best group,

best two groups, 2nd best two groups, ..., nbest-th best two groups,

... ...,

best nvarmax groups, ..., nbest-th best nvarmax groups].

groups is a vector of length sum(1:nvarmax) x nbest. If no combination satisfying the constraints is found, then corresponding elements are 0.

rss: rss is a vector of length nbest x nvarmax giving the residual sum of squares (RSS) corresponding to the best groups of various sizes.

rss = [Best group, 2nd best group, nbest-th best group,

best two groups, 2nd best two groups,

... ...,

best nvarmax groups, ..., nbest-th best nvarmax groups].

vars: vars is a matrix of nbest columns.

vars[, j] = [variables in jth best group,

variables in jth best two groups,

... ...,

variables in jth best nvarmax groups]

The number of rows is (nb * nvarmax + sum(1:nvarmax) * max(ngv)) * nbest. This allows for the largest group (of size max(ngv)) to be chosen. The reason is that it is not possible to determine how many variables are selected at runtime, which is connected to memory allocation. For each of the chosen groups, the variables are given in increasing order, with the nb fixed variables given first. When groups of size less than max(ngv) are chosen, the chosen variables are padded with zeroes at the end.

coef: coef is a matrix of the same dimensions as vars. Each entry in coef is the coefficient corresponding to the chosen variable in vars within its particular group, So

coef[, j] = [coefficients of jth best group,

coefficients of jth best two groups,

... ...,

coefficients of jth best nvarmax groups].

If vars[i, j] = 0, then coef[i, j] = 0.

nvars: nvars is an nvarmax x nbest matrix.

nvars[, j] = [number of variables in jth best group,

number of variables in jth best two groups,

... ...,

number of variables in jth nvarmax groups].

In each entry, the number of variables includes the nb fixed variables.

Value

rss	the RSS for the nbest combinations of up to nvarmax groups
groups	the nbest combinations of up to nvarmax groups
vars	the variables within each combination of groups chosen
coef	the weights of each of the variables chosen
nvars	number of variables for each combination of groups chosen
comptime	the total computation time for this task

Author(s)

Yi Guo <y.guo@westernsydney.edu.au>

References

Yi Guo, Mark Berman and Junbin Gao. Group subset selection for linear regression, Computational Statistics and Data Analysis, Volume 75, Page 39-52, 2014.

Examples

```
# Generate some test data.
D <- 100 # Number of observations
N <- 10 # Total number of variables including fixed variables.
x <- matrix(rnorm(D*N),D) # Explanatory variables

# Form 4 groups with 1, 2, 3 and 1 variables in these groups.
ngv <- c(1,2,3,1)

# Number of fixed variables, 3 altogether.
nb <- N - sum(ngv)

# Generate dependent variable randomly.
# It takes group 2 and 4 plus fixed variables, without any constraints on the coefficients.
# So gss should find the combination of groups 2 and 4 with zero rss.
coef <- 1:6
y <- x[,-c(4,7:9)]%*%coef

# Run it and ask for top 2 combinations of up to 2 groups.
res <- groupsubsetselection(y,x,nvarmax=2,nbest=2,nb,consind=rep(0,N),conslb=rep(0,N),ngv)

# Check groups it selects
res$groups

# [1] 2 4 2 4 1 2

# The first two groups are the 2 best single groups gss found, the 2nd and 4th respectively.
# The following two combinations, i.e. (2, 4) and (2, 3) are the 2 best combinations of two
# groups gss found. groups 2 and 4 is the best combination of two groups

# Check the RSS
res$rss
```

```
# [1] 2326.506 5157.622    0.000 2265.870

# The third entry is the RSS for fitting groups 2 and 4. It is zero apart from numerical error.
# The other fits are much worse.
#
# Now generate another example where the coefficents contains a negative value.

coef <- c(rep(1,3),-1,2,3)
y <- x[,-c(4,7:9)]%*%coef

# Run it and ask for top 2 combinations of up to 2 groups, again without any coefficient
# constraints.
res <- groupsubsetselection(y,x,nvarmax=2,nbest=2,nb,consind=rep(0,N),conslb=rep(0,N),ngv)

# Check groups it selects

res$groups

# [1] 4 2 2 4 3 4

# It still selects groups 2 and 4 because no constraint has been applied.
# Now constrain the model so that only positive coefficients for the non-fixed variables
# are allowed.

res <- groupsubsetselection(y,x,nvarmax=2,nbest=2,nb,consind=c(rep(0,nb),rep(1,N-nb)),
                            conslb=rep(0,N),ngv)

# Note the consind vector has been changed so that the non-fixed variables must have positive
# coefficients (specified by conslb).

# Check groups again

res$groups

# [1] 4 0 1 4 0 0

# Groups 2 and 4 are no longer on the selected list.
# N.B. Because the samples were randomly generated, the groups selected in run time may be
# different from those listed above.
# Zeroes in the list stand for nothing selected for that combination.
```

Index