Package ‘pclogit’
August 10, 2012

Type Package

Title Penalized conditional (unconditional) logistic regression using a network-based penalty for matched (unmatched) case-control data with grouped or graph-constrained variables.

Version 0.1

Date 2012-04-26

Author Hokeun Sun

Maintainer Hokeun Sun <hs2674@columbia.edu>

Depends Matrix

Description An efficient algorithm for fitting the regularization path and providing selection probabilities of each predictor for analysis of high-dimensional matched (unmatched) case-control data. The algorithm uses cyclical coordinate descent in a pathwise fashion.

License GPL-2

R topics documented:

\begin{description}
\item[pclogit] penalized conditional (unconditional) logistic regression for grouped or graph-constrained variables
\end{description}

Description

Fit a regularization path of conditional (unconditional) logistic regression model for a matched (unmatched) case-control response at a grid of values for regularization parameter lambda.

When predictors are correlated within either a group or a network graph, Laplacian matrix can be imposed into the procedure to efficiently select true signals.
pclogit

Usage

pclogit(x,y,strata=NULL,alpha=1.0,nlam=100,lambda=NULL,eps=NULL,
maxit=100000,maxp=p,thre=1e-6,group=NULL,type=c("ring","fcon"),
sgnc=NULL)

Arguments

x
The design matrix \((n \times p)\) without an intercept. pclogit standardizes the data
by default, but the coefficients are always returned on the original scale.

y
The response variable coded as 1 for cases and 0 for the matched controls.

strata
A vector of consecutive integers indicating the stratum of each observation.
Each stratum must have exactly one case and at least one control. If not speci-
fied, pclogit fits an ordinary logistic regression.

alpha
The penalty mixing parameter with \(0 \leq \alpha \leq 1\) and default is 1. See details.

nlam
The number of lambda values and default is 100.

lambda
A user supplied sequence of lambda values. Typically, this is left unspecified,
and the program automatically computes its own lambda sequence based on
nlam and eps.

eps
The smallest value for lambda as a fraction of lambda.max. The value
should be greater than 1E-5. The default is .0001 if \(n > p\) and .05 if \(n \leq p\).

maxit
Maximum number of passes over the data for all lambda values, and default is
10^5.

maxp
Limit the maximum number of variables ever to be nonzero.

thre
Convergence threshold for coordinate descent algorithm. The default value is
1E-6.

group
Either an integer vector of group sizes or a symmetric adjacency matrix. group
describes either grouped or graph structure of predictors x. If no information
between predictors is available, i.e., not specified, the pclogit performs an
elastic-net regularization procedure. See details.

type
A type of grouping network when group is defined as a vector of group sizes.
"ring" and "fcon" represent a ring and fully connected network, respectively.
Default is "ring". See details.

sgnc
Signs of regression coefficients. This can be provided only if group is specified
as either a list of group size or adjacency matrix. The estimated signs of ridge
regression for \(n \leq p\) or ordinary regression for \(n > p\) can be used for an adaptive
network-based regularization procedure. See details.

Details

The penalty function of pclogit is defined as

\[ \alpha \|\beta\|_1 + (1 - \alpha)(\beta^T S^T L S \beta) / 2, \]

where \(S\) is a \(p\) dimensional diagonal matrix with estimated signs of regression coefficients on its di-
goinal entries, and \(L\) is a Laplacian matrix describing a graph structure of covariates. This penalty
is equivalent to the Lasso penalty if alpha=1. When group and sgnc are not defined, \(L\) and \(S\)
in the penalty function are replaced by an identity matrix, respectively. In this case, pclogit performs an elastic-net regularization procedure since the second term of the penalty simply reduces
to the $l_2$ norm of $\beta$.

If group sizes of predictors are listed in `group`, it is assumed that all variables of the same groups are linked with each other like a ring or a fully connected network. In this case, the Laplacian matrix forms a block-wise diagonal matrix. The signs of regression coefficients `sgnc` can provide more accurate estimates in case some variables either in the same group or linked with each other have different signs of their regression coefficients, where the coefficients are not expected to be locally smooth.

**Value**

- `b0`: Intercept sequence of length of `lambda`. This is present only if an ordinary logistic regression is fit, i.e., `strata` was not defined.
- `strata`: The strata of observations if `strata` was defined.
- `beta`: The coefficient matrix with a dimension ($p \times n.lam$), stored in sparse column format ("CsparseMatrix")
- `lambda`: The actual sequence of `lambda` values used
- `df`: The number of nonzero coefficients for each value of `lambda`
- `nobs`: The number of observations, $n$
- `alpha`: The value of `alpha` used
- `iterations`: Total passes over the data summed over all `lambda` values
- `jerr`: The error flag, for warnings and errors (largely for internal debugging)

**Author(s)**

Hokeun Sun <hs2674@columbia.edu>

**References**

H. Sun and S. Wang (2012) Penalized Logistic Regression for High-dimensional DNA Methylation Data with Case-Control Studies, Bioinformatics 28(10), 1368-1375


**Examples**

```r
n<-200
p<-1000
x<-matrix(rnorm(n*p),n,p)

# one-to-one matched set
y<-c(rep(0,n/2),rep(1,n/2))
str<-rep(seq(n/2),2)

# one-to-four matched set
y<-c(rep(0,4*n/5),rep(1,n/5))
str<-c(rep(seq(n/5),rep(4,n/5)),rep(seq(n/5),1))

# a total of 100 groups each of which consists of 5, 10, or 20 members
gr<-c(rep(5,40),rep(10,40),rep(20,20))
```
# an example of adjacency matrix
adjm<-cov(x)
diag(adjm)<-0
adjm[abs(adjm)<=0.3]<-0
adjm[abs(adjm)>0.3]<-1

# an example of signs of coefficients
sg<-sign(rnorm(p))

# Lasso
g1<-pclogit(x,y,stra)

# Elastic-net
g2<-pclogit(x,y,stra,alpha=0.1)

# Ring network of grouped covariates
g3<-pclogit(x,y,stra,alpha=0.1,group=gr)

# Fully connected network of grouped covariates
g4<-pclogit(x,y,stra,alpha=0.1,group=gr,type="fcon")

# Graph-constrained covariates
g5<-pclogit(x,y,stra,alpha=0.1,group=adjm)

# Adaptive graph-constrained covariates
g6<-pclogit(x,y,stra,alpha=0.1,group=adjm,sgnc=sg)

---

**sel.pclogit**

*selection probabilities of regression coefficients*

**Description**

The selection probability of each regression coefficient is computed based on resamplings.

**Usage**

`sel.pclogit(x, y, stra=NULL, ..., psub=0.5, N.lam=5, K=100)`

**Arguments**

- **x**  
The design matrix (n x p) without an intercept. `pclogit` standardizes the data by default, but the coefficients are always returned on the original scale.
- **y**  
The response variable coded as 1 for cases and 0 for the matched controls.
- **stra**  
A vector of consecutive integers indicating the stratum of each observation. Each stratum must have exactly one case and at least one control. If not specified, `sel.pclogit` fits an ordinary logistic regression.
- **...**  
Other arguments that can be passed to `pclogit`.
- **psub**  
The proportion of subsamples used for resamplings, and `psub ∈ [0.5, 1)`. The default is 0.5.
- **N.lam**  
The number of lambda values used for resamplings, and default is 5.
- **K**  
The number of resamplings, and default is 100.
Details

The half of the strata \texttt{stra} are randomly selected without replacement \( K \) times. For each replication, the paired \((x,y)\) in the selected strata are only used for \texttt{pclogit} to find non-zero coefficients along with \texttt{N.lam} \texttt{lambda} values. The selection probability of each coefficient is then computed based on the proportion of non-zeros out of \( K \) replications. In an ordinary logistic model, the half of cases and controls are selected each time.

Value

- \texttt{beta} \quad \text{The selection probability (p x N.lam)}
- \texttt{maxsel} \quad \text{The maximum selection probability of each coefficient are listed in descending order along with the corresponding variable.}
- \texttt{lambda} \quad \text{The actual sequence of lambda values used}
- \texttt{K} \quad \text{The actual number of resamplings used}

Author(s)

Hokeun Sun <hs2674@columbia.edu>

References


Examples

```r
n<-200
p<-1000
x<-matrix(rnorm(n*p),n,p)

# one-to-one matched set
y<-c(rep(0,n/2),rep(1,n/2))
st<-rep(seq(n/2),2)

# one-to-four matched set
y<-c(rep(0,4*n/5),rep(1,n/5))
st<-c(rep(seq(n/5),rep(4,n/5)),rep(seq(n/5),1))

# a total of 100 groups each of which consists of 5, 10, or 20 members
gr<-c(rep(5,40),rep(10,40),rep(20,20))

# an example of adjacency matrix
adjm<-cor(x)
diag(adjm)<0
adjm[abs(adjm)<0.3]<0
adjm[abs(adjm)>0.3]<1

# an example of signs of coefficients
sg<-sign(rnorm(p))

# Lasso
```
g1<-sel.pclogit(x,y,st)

# Elastic-net
g2<-sel.pclogit(x,y,st,alpha=0.1)

# Ring network of grouped covariates
g3<-sel.pclogit(x,y,st,alpha=0.1,group=gr)

# Fully connected network of grouped covariates
g4<-sel.pclogit(x,y,st,alpha=0.1,group=gr,type="fcon")

# Graph-constrained covariates
g5<-sel.pclogit(x,y,st,alpha=0.1,group=adjm)

# Adaptive graph-constrained covariates
g6<-sel.pclogit(x,y,st,alpha=0.1,group=adjm,sgnc=sg)
Index

pclogit, 1
sel.pclogit, 4