Package ‘interep’

April 20, 2020

Type Package

Title Interaction Analysis of Repeated Measure Data

Version 0.3.1

Author Fei Zhou, Jie Ren, Xiaoxi Li, Cen Wu, Yu Jiang

Maintainer Fei Zhou <feiz@ksu.edu>

Description Extensive penalized variable selection methods have been developed in the past two decades for analyzing high dimensional omics data, such as gene expressions, single nucleotide polymorphisms (SNPs), copy number variations (CNVs) and others. However, lipidomics data have been rarely investigated by using high dimensional variable selection methods. This package incorporates our recently developed penalization procedures to conduct interaction analysis for high dimensional lipidomics data with repeated measurements. The core module of this package is developed in C++. The development of this software package and the associated statistical methods have been partially supported by an Innovative Research Award from Johnson Cancer Research Center, Kansas State University.

Depends R (>= 3.5.0)

License GPL-2

Encoding UTF-8

LazyData true

Imports Rcpp, MASS

LinkingTo Rcpp, RcppArmadillo

URL https://github.com/feizhoustat/interep

BugReports https://github.com/feizhoustat/interep/issues

NeedsCompilation yes

Repository CRAN

Date/Publication 2020-04-20 07:50:11 UTC

RoxygenNote 6.1.1
**R topics documented:**

- `cv.interep` ................................................................. 2
- `dat` ........................................................................... 4
- `dmcp` ........................................................................... 4
- `interep` ........................................................................... 5
- `penalty` .......................................................................... 7
- `reformat` .......................................................................... 7

Index 9

---

`cv.interep`  
\textit{k-folds cross-validation for interep}

**Description**

This function does k-fold cross-validation for interep and returns the optimal value of lambda.

**Usage**

\[
\text{cv.interep}(e, g, y, \beta_0, \lambda_1, \lambda_2, \text{n folds}, \text{corre}, \text{p method}, \text{max its})
\]

**Arguments**

- **e**
  - matrix of environment factors.
- **g**
  - matrix of omics factors. In the case study, the omics measurements are lipidomics data.
- **y**
  - the longitudinal response.
- **beta0**
  - the initial value for the coefficient vector.
- **lambda1**
  - a user-supplied sequence of \( \lambda_1 \) values, which serves as a tuning parameter for individual predictors.
- **lambda2**
  - a user-supplied sequence of \( \lambda_2 \) values, which serves as a tuning parameter for interactions.
- **nfolds**
  - the number of folds for cross-validation.
- **corre**
  - the working correlation structure that is used in the estimation algorithm. interep provides three choices for the working correlation structure: "a" as AR-1", "i" as "independence" and "e" as "exchangeable".
- **p method**
  - the penalization method. "mixed" refers to MCP penalty to individual main effects and group MCP penalty to interactions; "individual" means MCP penalty to all effects.
- **max its**
  - the maximum number of iterations that is used in the estimation algorithm.
Details

When dealing with predictors with both main effects and interactions, this function returns two optimal tuning parameters, $\lambda_1$ and $\lambda_2$; when there are only main effects in the predictors, this function returns $\lambda_1$, which is the optimal tuning parameter for individual predictors containing main effects.

Value

an object of class "cv.interep" is returned, which is a list with components:

- lam1: the optimal $\lambda_1$.
- lam2: the optimal $\lambda_2$.

References


---

### dat

**simulated data for demonstrating the features of interep**

**Description**

Simulated data for demonstrating the features of interep.

**Usage**

```r
data("dat")
```

**Format**

Each data consists of six components: e, z, x, y, coef and index; index shows the location of the true coefficients used to generate y.

**Examples**

```r
data("dat")
```

---

### dmcp

**This function obtains the first derivative function of MCP (Minimax Concave Penalty)**

**Description**

This function obtains the first derivative function of MCP (Minimax Concave Penalty)

**Usage**

```r
dmcp(theta, lambda, gamma = 3)
```
Arguments

theta     a coefficient vector.
lambda    the tuning parameter.
gamma     the regularization parameter in MCP (Minimax Concave Penalty). It balances
           between the unbiasedness and concavity of MCP.

Details

Rigorously speaking, the regularization parameter \( \gamma \) needs to be obtained via a data-driven
approach. Published studies suggest experimenting with a few values, such as 1.8, 3, 4.5, 6, and 10,
then fixing its value. In our numerical study, we have examined this sequence and found that the
results are not sensitive to the choice of value of \( \gamma \), and set the value at 3. In practice, to be prudent,
values other than 3 should also be investigated. Similar discussions can be found in the references
below.

Value

the first derivative of MCP function.

References

Ren, J., Du, Y., Li, S., Ma, S., Jiang, Y. and Wu, C. (2019). Robust network-based regularization and
variable selection for high-dimensional genomic data in cancer prognosis. *Genetic epidemiology*,
43(3), 276-291

for Generalized Linear Models. *R package*, version 0.4.0

between (epi) genetic measurements. *Journal of multivariate analysis*, 168, 119-130

for high dimensional SNP data in the case–control study of Type 2 diabetes. *BMC genetics*, 18(1),
44

Examples

theta=runif(20,-5,5)
lambda=1
dmcp(theta,lambda,gamma=3)
Usage

```
interep(e, g, y, beta0, corre, pmethod, lam1, lam2, maxits)
```

Arguments

e matrix of environment factors.

g matrix of omics factors. In the case study, the omics measurements are lipidomics data.

y the longitudinal response.

beta0 the initial coefficient vector.

corre the working correlation structure that is used in the estimation algorithm. interep provides three choices for the working correlation structure: "a" as AR-1", "i" as "independence" and "e" as "exchangeable".

pmethod the penalization method. "mixed" refers to MCP penalty to individual main effects and group MCP penalty to interactions; "individual" means MCP penalty to all effects.

lam1 the tuning parameter lambda1 for individual predictors.

lam2 the tuning parameter lambda2 for interactions.

maxits the maximum number of iterations that is used in the estimation algorithm. The default value is 30

Value

cocf the coefficient vector.

References


Examples

data("dat")
e=dat$e
g=dat$z
y=dat$y
beta0=dat$coef
index=dat$index
b = interep(e, g, y, beta0, corre="e", pmethod="mixed", lam1=dat$lam1, lam2=dat$lam2, maxits=30)
b[abs(b)<0.05]=0
pos = which(b != 0)
tp = length(intersect(index, pos))
fp = length(pos) - tp
list(tp=tp, fp=fp)
penalty

This function gives the penalty functions

Usage

penalty(x, n, p, q, beta, lam1, pmethod, p1, lam2)

Arguments

x matrix of covariates.
n the sample size.
p the number of predictors.
q the number of environment factors.
beta the coefficient vector.
lam1 the tuning parameter lambda1 for individual penalty.
pmethod the penalization method. "mixed" refers to MCP penalty to individual main effects and group MCP penalty to interactions; "individual" means MCP penalty to all effects.
p1 the number of gene factors.
lam2 the tuning parameter lambda2 for group penalty.

Value

E the penalty function.

reformat

This function changes the format of the longitudinal data from wide format to long format

Description

This function changes the format of the longitudinal data from wide format to long format

Usage

reformat(k, y, x)
Arguments

k  the number of repeated measurement.
y  the longitudinal response.
x  a matrix of predictors, consisting of omics and environment factors, as well as their interactions. In the case study, the omics measurements are lipidomics data.
Index

*Topic datasets
  dat, 4

  cv.interep, 2
  dat, 4
dmcp, 4
  interep, 5
penalty, 7
reformat, 7