Package 'gfmR'

March 14, 2018

Type Package

Title Implements Group Fused Multinomial Regression

Version 1.1-0

Date 2018-03-13

Author Brad Price

Maintainer Brad Price <brad.price@mail.wvu.edu>

Depends R (>= 3.1.0), parallel, nnet, faraway

Suggests knitr, rmarkdown

VignetteBuilder knitr

Description Software to implement methodology to preform automatic response category combinations in multinomial logistic regression. There are functions for both cross validation and AIC for model selection. The method provides regression coefficient estimates that may be useful for better understanding the true probability distribution of multinomial logistic regression when category probabilities are similar. These methods are not recommended for a large number of predictor variables.

License GPL-2

LazyLoad yes

NeedsCompilation yes

URL https://www.r-project.org

Repository CRAN

Date/Publication 2018-03-14 14:41:11 UTC

R topics documented:

GFMR.cv *Tuning parameter selection using validation likelihood for GFMR.*

Description

This routine implements K fold cross validation for group fused multinomial regression.

Usage

```
GFMR.cv(Y,X,lamb,sampID,H,n.cores=1,rho=10^-8,...)
```

Arguments

Y	A matrix of response category counts where the columns represent the categories and rows represent the observations. Currently supported for n=1.
X	A matrix of predictor variables. The columns represent predictors and rows represent observations.
lamb	tuning parameter for fusion penalty
sampID	An identified or the sampleID for the cross validation routine. Should take values 1:k and be user supplied.
H	An indicator matrix representing the edge set of the penalty set. The matrix is square and symmetric with dimension number of response categories, and if two categories are in the penalty set a 1 should be in the row column combination.
n.cores	The number of cores for the mclapply function
rho	Step Size parameter of ADMM
...	Other arguments for Group fused multinomial regression

Details

This routine implements the validation likelihood approach proposed by Price et. al. to obtain the tuning parameter for Group Fused Multinomial Regression which automatically combines response categories in multinomial regression. Show to do well with regard to predicting the true category probabilities.

Value

A list is returned with elements

vl	The validation likelihood for all tuning parameters
vl.sd	The standard deviation of the validation likelihood
lambda	tuning parameter used
vl.mat	Validation likelihood for each

Author(s)

Brad Price, <brad.price@mail.wvu.edu>.

References

Price, B.S, Geyer, C.J. and Rothman, A.J. "Automatic Response Category Combination in Multinomial Logistic Regression." https://arxiv.org/abs/1705.03594.

Examples

```
## Not run: data(nes96)
attach(nes96)
Response=matrix(0,944,7)
for(i in 1:944){
  if(PID[i]=="strRep"){Response[i,1]=1}
  if(PID[i]=="weakRep"){Response[i,2]=1}
  if(PID[i]=="indRep"){Response[i,3]=1}
  if(PID[i]=="indind"){Response[i,4]=1}
  if(PID[i]=="indDem"){Response[i,5]=1}
  if(PID[i]=="weakDem"){Response[i,6]=1}
  if(PID[i]=="strDem"){Response[i,7]=1}
}

Hmat=matrix(1,dim(Response)[2],dim(Response)[2])
diag(Hmat)=0
ModMat<-lm(popul~age,x=TRUE)$x

X=cbind(ModMat[,1],apply(ModMat[,-1],2,scale))

set.seed(1010)
n=dim(Response)[1]
sampID=rep(5,n)
samps=sample(1:n)
mine=floor(n/5)
for(j in 1:4){
  sampID[samps[((j-1)*mine+1):(j*mine)]]=j
}

o1<-GFMR.cv(Response,X,lamb = 2^seq(4.2,4.3,.1),H=Hmat2,sampID = sampID,n.cores =5)

which(o1$vl==max(o1$vl))

## End(Not run)
```

GroupFusedMulti *Group Fused Multinomial Logistic Regression*

Description

This routine fits the group fused multinomial logistic regression model, which uses fusion shrinkage to automatically combine response categories.

Usage

```
GroupFusedMulti(Y,X,lambda,H,tol1=10^-7,tol2=10^-7,TD=2,rho=10^-8,tau1=10^-9,iter=1e3)
```

Arguments

Y	A matrix of response category counts where the columns represent the categories and rows represent the observations. Currently supported for n=1.
X	A matrix of predictor variables. The columns represent predictors and rows represent observations.
lambda	tuning parameter for fusion penalty
H	An indicator matrix representing the edge set of the penalty set. The matrix is square and symmetric with dimension number of response categories, and if two categories are in the penalty set a 1 should be in the row column combination.
tol1	Convergence tolerance for ADMM
tol2	Convergence tolerance of ADMM
TD	Step size Adjustment for iterative step size
rho	ADMM step-size parameter, iterative implementation
tau1	The threshold parameter to 0 for final estimates
iter	Maximum number of iterations of the algorithm

Details

Implements the ADMM algorithm for the group fused multinomial regression estimates proposed by Price et. al.

Value

The function returns a list:

Coeff	Final coefficient estimates for the gfmr model
Approx	Final iterates of beta before threshold
Z	Final iterate of Z in the ADMM algorithm
lambda	Tuning Parameter
Converge	Indicator of algorithm convergence
NGroups	Number of groups in final estimates
Groups	The response category groups in the final estimates

Author(s)

Brad Price, <brad.price@mail.wvu.edu>.

References

Price, B.S, Geyer, C.J. and Rothman, A.J. "Automatic Response Category Combination in Multinomial Logistic Regression." https://arxiv.org/abs/1705.03594.

Examples

```
## Not run: data(nes96)
attach(nes96)
Response=matrix(0,944,7)
for(i in 1:944){
  if(PID[i]=="strRep"){Response[i,1]=1}
  if(PID[i]=="weakRep"){Response[i,2]=1}
  if(PID[i]=="indRep"){Response[i,3]=1}
  if(PID[i]=="indind"){Response[i,4]=1}
  if(PID[i]=="indDem"){Response[i,5]=1}
  if(PID[i]=="weakDem"){Response[i,6]=1}
  if(PID[i]=="strDem"){Response[i,7]=1}
}

Hmat=matrix(1,dim(Response)[2],dim(Response)[2])
diag(Hmat)=0
ModMat<-lm(popul~age,x=TRUE)$x

X=cbind(ModMat[,1],apply(ModMat[,-1],2,scale))
mod<-GroupFusedMulti(Response,X,lambda=2^4.3,H=Hmat2,rho=10^2,iter=50,tol1=10^-4,tol2=10^-4)
predict(mod,X[1,])

## End(Not run)
```

`predict.gfmR` *Predict method for group fused multinomial logistic regression*

Description

This routine fits the group fused multinomial logistic regression model, which uses fusion shrinkage to automatically combine response categories.

Usage

```
## S3 method for class 'gfmR'
predict(object,newdata,type="probs",...)
```

Arguments

`object`	A gfmr object which specifically is the output from the GroupFusedMulti function.
`newdata`	value to be predicted in model matrix form

type	Type of prediction, "probs" returns probabilities, while response returns log of the ratio of probabilities for baseline category.
...	Other arguments

Details

Prediction function for GFMR

Value

A vector or a matrix corresponding to type return.

Author(s)

Brad Price, <brad.price@mail.wvu.edu>.

References

Price, B.S, Geyer, C.J. and Rothman, A.J. "Automatic Response Category Combination in Multinomial Logistic Regression." https://arxiv.org/abs/1705.03594.

Examples

```
## Not run: data(nes96)
attach(nes96)
Response=matrix(0,944,7)
for(i in 1:944){
  if(PID[i]=="strRep"){Response[i,1]=1}
  if(PID[i]=="weakRep"){Response[i,2]=1}
  if(PID[i]=="indRep"){Response[i,3]=1}
  if(PID[i]=="indind"){Response[i,4]=1}
  if(PID[i]=="indDem"){Response[i,5]=1}
  if(PID[i]=="weakDem"){Response[i,6]=1}
  if(PID[i]=="strDem"){Response[i,7]=1}
}

Hmat=matrix(1,dim(Response)[2],dim(Response)[2])
diag(Hmat)=0
ModMat<-lm(popul~age,x=TRUE)$x

X=cbind(ModMat[,1],apply(ModMat[,-1],2,scale))
mod<-GroupFusedMulti(Response,X,lambda=2^4.3,H=Hmat2,rho=10^2,iter=50,tol1=10^-4,tol2=10^-4)
predict(mod,X[1,])

## End(Not run)
```

print.gfmR *print method for group fused multinomial logistic regression*

Description

This routine fits the group fused multinomial logistic regression model, which uses fusion shrinkage to automatically combine response categories.

Usage

```
## S3 method for class 'gfmR'
print(x,...)
```

Arguments

x	A gfmr object which specifically is the output from the GroupFusedMulti function.
...	Other arguments

Details

Prediction function for GFMR

Value

A vector or a matrix corresponding to type return.

Author(s)

Brad Price, <brad.price@mail.wvu.edu>.

References

Price, B.S, Geyer, C.J. and Rothman, A.J. "Automatic Response Category Combination in Multinomial Logistic Regression." https://arxiv.org/abs/1705.03594.

Examples

```
## Not run: data(nes96)
attach(nes96)
Response=matrix(0,944,7)
for(i in 1:944){
  if(PID[i]=="strRep"){Response[i,1]=1}
  if(PID[i]=="weakRep"){Response[i,2]=1}
  if(PID[i]=="indRep"){Response[i,3]=1}
  if(PID[i]=="indind"){Response[i,4]=1}
  if(PID[i]=="indDem"){Response[i,5]=1}
  if(PID[i]=="weakDem"){Response[i,6]=1}
```

```
  if(PID[i]=="strDem"){Response[i,7]=1}
}

Hmat=matrix(1,dim(Response)[2],dim(Response)[2])
diag(Hmat)=0
ModMat<-lm(popul~age,x=TRUE)$x

X=cbind(ModMat[,1],apply(ModMat[,-1],2,scale))
mod<-GroupFusedMulti(Response,X,lambda=2^4.3,H=Hmat2,rho=10^2,iter=50,tol1=10^-4,tol2=10^-4)
mod

## End(Not run)
```

`print.gfmR.cv` *print method for group fused multinomial logistic regression validation likelihood tuning parameter selection*

Description

This routine fits the group fused multinomial logistic regression model, which uses fusion shrinkage to automatically combine response categories. This specifically focuses on tuning parameter selection with validation likelihood.

Usage

```
## S3 method for class 'gfmR.cv'
print(x,...)
```

Arguments

`x`	A gfmr.cv object which specifically is the output from the GroupFusedMulti function.
`...`	Other arguments

Details

print method for gfmR.cv objects.

Value

A readable printout of cross validation

Author(s)

Brad Price, <brad.price@mail.wvu.edu>.

References

Price, B.S, Geyer, C.J. and Rothman, A.J. "Automatic Response Category Combination in Multinomial Logistic Regression." https://arxiv.org/abs/1705.03594.

Examples

```
## Not run: data(nes96)
attach(nes96)
Response=matrix(0,944,7)
for(i in 1:944){
  if(PID[i]=="strRep"){Response[i,1]=1}
  if(PID[i]=="weakRep"){Response[i,2]=1}
  if(PID[i]=="indRep"){Response[i,3]=1}
  if(PID[i]=="indind"){Response[i,4]=1}
  if(PID[i]=="indDem"){Response[i,5]=1}
  if(PID[i]=="weakDem"){Response[i,6]=1}
  if(PID[i]=="strDem"){Response[i,7]=1}
}

Hmat=matrix(1,dim(Response)[2],dim(Response)[2])
diag(Hmat)=0
ModMat<-lm(popul~age,x=TRUE)$x

X=cbind(ModMat[,1],apply(ModMat[,-1],2,scale))

set.seed(1010)
n=dim(Response)[1]
sampID=rep(5,n)
samps=sample(1:n)
mine=floor(n/5)
for(j in 1:4){
  sampID[samps[((j-1)*mine+1):(j*mine)]]=j
}

o1<-GFMR.cv(Response,X,lamb = 2^seq(4.2,4.3,.1),H=Hmat2,sampID = sampID,n.cores =5)
o1

## End(Not run)
```

Index