SURF

Lihui Liu

July 20, 2020

```
SURF(Xo,y,X=NULL,fold=10,Alpha=1,prop=0.1,weights=FALSE,B=1000,C=200,
   ncores=1,display.progress=TRUE,family=stats::binomial(link="logit"),pval=0.05)
```

Performs variable selection based on subsampling, ranking forward selection. Xo is the matrix of predictor variables. y is the response variable. X is a matrix of additional predictors which should be scaled to have sum 1 prior to analysis. fold is the number of folds for cross-validation. Alpha is the parameter for the elastic net method used in the subsampling procedure: the default value of 1 corresponds to LASSO. prop is the proportion of variables to remove in the each subsample. weights indicates whether observations should be weighted by class size. When the class sizes are unbalanced, weighting observations can improve results. B is the number of subsamples to use for ranking the variables. C is the number of permutations to use for estimating the critical value of the null distribution. If the doParallel package is installed, the function can be run in parallel by setting ncores to the number of threads to use. If the default value of 1 is used, or if the doParallel package is not installed, the function does not run in parallel. display.progress indicates whether the function should display messages indicating its progress. family is a family variable for the glm fitting. Note that the `glmnet` package does not currently permit the use of non-standard link functions, so will always use the default link function. However, the glm fitting will use the specified link. The default is binomial with logistic regression, because this is a common use case. pval is the p-value for inclusion of a variable in the model. Under the null case, the number of false positives will be geometrically distributed with this as probability of success, so if this parameter is set to p, the expected number of false positives should be $\frac{p}{1-p}$.

Example:

```
> library(SuRF)
> ################################################
> #Example 1: continuous case (simulated data)
> ################################################
> #simulate a multivariate matrix with 1000 columns as predictors;
> #The response y is simulated based on a model of X1,X20 and X40 only
>
> set.seed(1234)
```

```
> p=10
> n=20
> corr=0.3
> # Using library(MASS) we can simulate as
> #Covmatrix <- outer(1:p, 1:p, function(x,y){corr^abs(x-y)})
> ##Xmat <- mvrnorm(n, rep(0,p), Covmatrix)
>
>
> #Using base R, we simulate
> Zmat <- rnorm(n*p)
> dim(Zmat)<-c(n,p)
> W<-matrix(0,p,p)
> for(i in seq_len(p)){
+     for(j in seq_len(i)){
+         W[i,j]<-corr^(i-j)
+     }
+ }
> W[,-1]<-W[,-1]*sqrt(1-corr^2)
> ###Now W%*%t(W)=Covmatrix
>
> Xmat <- Zmat%*%W
> truep=c(1,4,10)
> beta=1
> noise=rnorm(n)
> yc=beta*apply(Xmat[,truep],1,sum)+noise
> #All variables in 'Xmat' are not count variables and are not to be scaled;
> #these variables should be passed into Xo, not X (X is set to NULL in this case);
> #when there are both count variables and other types of variables, they can be passed into
> #Alpha=1 represent lasso method in 'glmnet';
> #prop=0.1 indicates 10% of samples are left in each subsample (e.g.,you keep 90% of sample
> #the cross validation size is 5 when ranking the variables
> #B=1000 represents the size of subsampling for the ranking step
> #C=50 represents the size of the permutation size for selecting the new variable;
> #specify family = stats::gaussian(link = "identity") for a continuous outcome
> #Use family = stats::binomial(link="logit") for a binary outcome instead
> #specify the alpha level for the permutation test  pval = 0.05
>
> mod=SURF(X=NULL,Xo=Xmat,y=yc,fold=5,Alpha=1,prop=0.1,weights=FALSE,B=100,C=50,ncores = 1,c
> #Set B=1000 for more thorough analysis
>
> #selected variables
> mod$selmod$vslist

[1] "X10" "X1"  "X4"  "X8"
```

Example:

```
> library(SuRF)
> ########################################################
> # Example 2 Binary outcome (Iris data in R)
> ########################################################
> data(iris)
> data=iris[iris$Species=="versicolor"|iris$Species=="setosa",]
> N=dim(data)[1]
> data$Species=as.character(data$Species)
> y=ifelse(data$Species=="setosa",0,1)
> ind=sample(1:N,floor((2/3)*N))
> Xtr=data[ind,1:4]
> ytr=y[ind]
> Xte=data[-ind,1:4]
> yte=y[-ind]
> mod=SURF(X=NULL,Xo=Xtr,y=ytr,fold=5,Alpha=1,prop=0.1,weights=FALSE,B=400,C=50,ncores = 1,c
> #selected variables
> selvar=mod$selmod$vslist
> selvar

[1] "Petal.Length"

> #prepare training data and test data with selected variables
> dat.tr=data.frame(as.matrix(Xtr[,selvar]),y=ytr)
> colnames(dat.tr)[1:length(selvar)]=selvar
> dat.te=data.frame(as.matrix(Xte[,selvar]),y=yte)
> colnames(dat.te)[1:length(selvar)]=selvar
> #fit the model with only selected variables from SuRF
> fitmod=glm(ytr~.,data=dat.tr,family=binomial(link = "logit"))
> #predict the new test samples
> pred=predict.glm(fitmod,newdata=dat.te,type="response")
> ypred=pred>0.5
> #show the classification matrix
> tab=table(yte,ypred)
> tab

   ypred
yte FALSE TRUE
  0    13    0
  1     0   21

> #mislcassification error rate
> MCER=1-sum(diag(tab))/(sum(tab))
> MCER

[1] 0

>
```