RotateClassifier.R

Kirill Panin
Fri Jun 16 07:50:31 2017

```
RotateClassifier<-function(data, ntree , control=control)
{
 nname=colnames(data)
 index=which(nname=="Class")
 class=data[,index]
 data=data[,-index]

 nlevel=levels(class)

 dd=dim(data)

 allfit=NULL
 allPCA=NULL
 PCA=matrix(0,dd[2],dd[2])
 for(i in 1:ntree)
 {
  #res=sample(2,dd[2],replace = T,prob = c(0.5,0.5))
  # f1 <- res[sample.ind==1]
  # f2 <- res[sample.ind==2]
  res=sample(dd[2])
  till=ceiling(dd[2]/2)
  f1 <- res[1:till]
  f2 <- res[(till+1):dd[2]]

  sample1=sample(dd[1],replace = TRUE)
  sample2=sample(dd[1],replace = TRUE)
  mPCA1=prcomp(~.,data=data[sample1,f1],center=F,na.action=na.omit)
  m1=dim(mPCA1$rotation)[2]
  PCA[f1,1:m1]=mPCA1$rotation
  mPCA2=prcomp(~.,data=data[sample2,f2],center=F,na.action=na.omit)
  m2=dim(mPCA2$rotation)[2]
  PCA[f2,(m1+1):(m1+m2)]=mPCA2$rotation
  if((m1+m2)<dd[2])
  {
   PCA1=PCA[,1:(m1+m2)]
  }
  else
  {
   PCA1=PCA
  }

  matr=as.matrix(data)
  dataPCA=matr%*%PCA1
```

```
    colnames(dataPCA)=paste("PC",1:(m1+m2),sep="")
    dataPCA=as.data.frame(dataPCA)
    dataPCA$Class=class
    fit <- rpart(Class ~., data = dataPCA, control=control)
    allfit=c(allfit,list(fit))
    allPCA=c(allPCA,list(PCA1))
  }
  return(list(allfit=allfit,PCA=allPCA,nlevel=nlevel))
}

predict.Rotate<-function(fit,data,type)
{
  allfit=fit$allfit
  ntree=length(allfit)
  nlevel=fit$nlevel


  allout=NULL
  matr=as.matrix(data)

  for(i in 1:ntree)
  {
    data=matr%*%fit$PCA[[i]]
    colnames(data)=paste("PC",1:dim(data)[2],sep="")
    data=as.data.frame(data)
    out=predict(allfit[[i]],newdata=data, type = type)
    allout=cbind(allout,out)
  }

  vote=sapply(1:length(out),function(z) tabulate(allout[z,], nbins =
length(nlevel)))
  vote=unlist(vote)
  dim(vote)=c(length(nlevel),length(out))
  rownames(vote)=nlevel

  index.class=apply(vote,2, which.max)
  class.sample=nlevel[index.class]
  return(list(class.sample=class.sample,vote=vote))
}
```