hw4

Ying Xiang

4/4/2021

#Problem 1 # load datasets

```r
data3=read.table("train_3.txt",sep=",")
data3$Y=(rep(-1,length(data3[,1])))
data8=read.table("train_8.txt",sep=",")
data8$Y=(rep(1,length(data8[,1])))
data3_8=rbind(data3,data8)
w=rep(1/length(data3_8[,1]),length(data3_8[,1]))
```

#Implement the AdaBoost algorithm in R.

```r
adaBoost <- function(X,y,B) {
  n <- dim(X)[1]
  w <- rep(1/n,times=n)
  alpha <- rep(0,times=B)
  allPars <- rep(list(list()),B)

  for(b in 1:B) {
    allPars[[b]] <- train(X,w,y)
    missClass <- (y != classify(X,allPars[[b]]))
    e <- (w %*% missClass/sum(w))[1]
    alpha[b] <- log((1-e)/e)
    w <- w*exp(alpha[b]*missClass)
  }
  return(list(allPars=allPars , alpha=alpha))
}

agg_class <- function(X,alpha ,allPars) {
  n <- dim(X)[1]
  B <- length(alpha)
  Labels <- matrix(0,nrow=n,ncol=B)
  for(b in 1:B) {
    Labels[,b] <- classify(X,allPars[[b]])
  }
  Labels <- Labels %*% alpha
  c_hat <- sign(Labels)
  return(c_hat)
}

train <- function(X,w,y) {
  n <- dim(X)[1]
  p <- dim(X)[2]
  mode <- rep(0,times=p)
```

```r
  theta <- rep(0,times=p)
  loss <- rep(0,times=p)
  for(j in 1:p) {
    indx <- order(X[,j])
    x_j <- X[indx,j]
    w_cum <- cumsum(w[indx] * y[indx])
    m <- max(abs(w_cum), na.rm=TRUE)
    maxIndx <- min(which(abs(w_cum)==m))
    mode[j] <- (w_cum[maxIndx] < 0)*2 - 1
    theta[j] <- x_j[maxIndx]
    c <- ((x_j > theta[j])*2 - 1)*mode[j]
    loss[j] <- w %*% (c != y)
  }
  m <- min(loss)
  j_star <- min(which(loss==m))
  pars <- list(j=j_star, theta=theta[j_star], mode=mode[j_star])
  return(pars)
}

classify <- function(X,pars) {
  label <- (2*(X[,pars$j] > pars$theta) - 1)*pars$mode
  return(label)
}
```

#Run the algorithm on the USPS data and evaluate results using cross validation.

```r
B_max <- 50
nCV <- 5
X <- data3_8[,1:256]
y <- data3_8[,257]
n <- dim(X)[1]
testErrorRate <- matrix(0,nrow=B_max,ncol=nCV)
trainErrorRate <- matrix(0,nrow=B_max,ncol=nCV)
p <- sample.int(n)
for(B in 1:B_max){

  for(i in 1:nCV){
  testIndx=p[((i-1)*240+1):(i*240)]
  trainIndx=p[-(((i-1)*240+1):(i*240))]

  ada <- adaBoost(X[trainIndx ,], y[trainIndx], B_max)
  allPars <- ada$allPars
  alpha <- ada$alpha

  c_hat_test <- agg_class(X[testIndx,],alpha[1:B],allPars[1:B])
  testErrorRate[B,i] <- mean(y[testIndx] != c_hat_test)
  c_hat_train <- agg_class(X[trainIndx ,], alpha[1:B], allPars[1:B])
  trainErrorRate[B,i] <- mean(y[trainIndx] != c_hat_train)
  }
}
```

#Plot the training error and the test error as a function of b.

```r
testER=rowSums(testErrorRate)
trainER=rowSums(trainErrorRate)
```

```
plot(1:B_max,testER,main="error rate",type="l",col="blue")
lines(1:B_max,trainER,type="l",col="red")
legend("topright",c("train error","test error"),fill=c("red","blue"))
```

error rate