Special Task

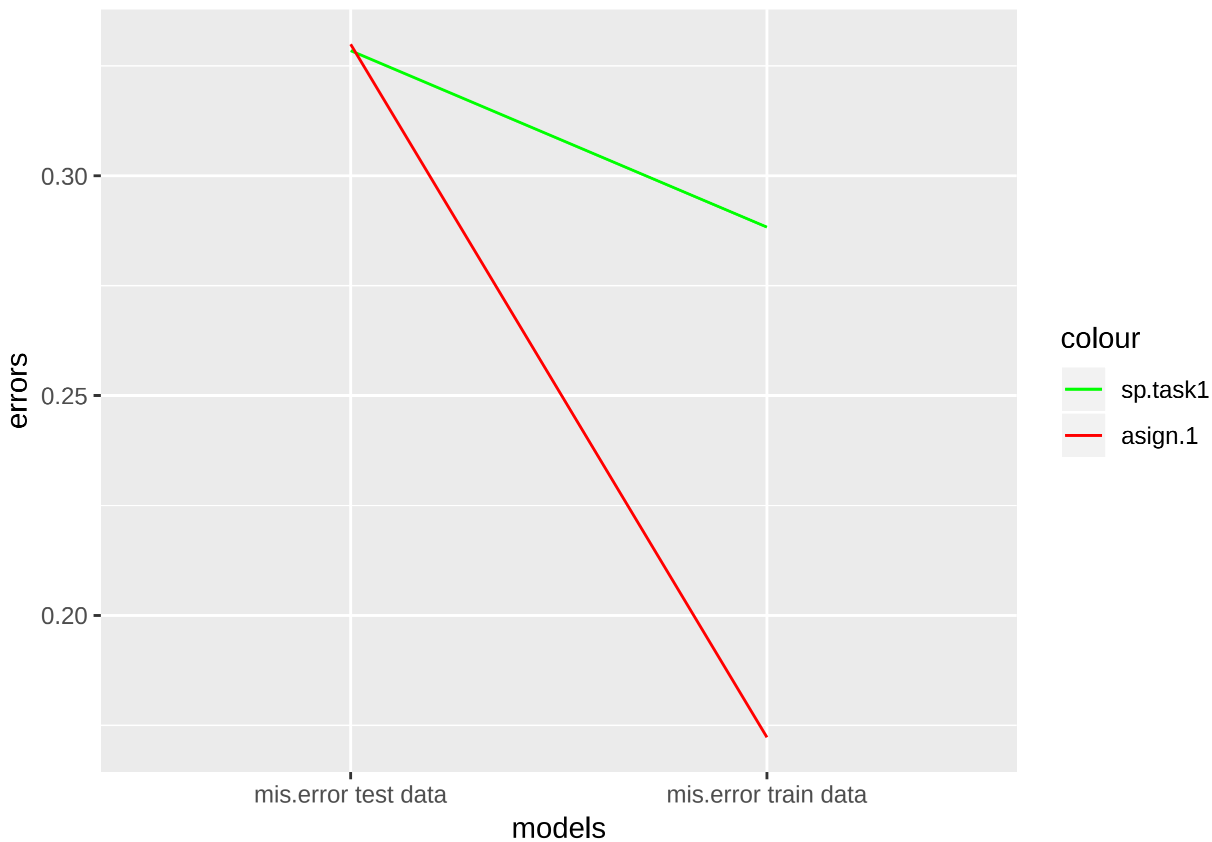
Andreas C Charitos-andch552

25 Nov 2018

Contents

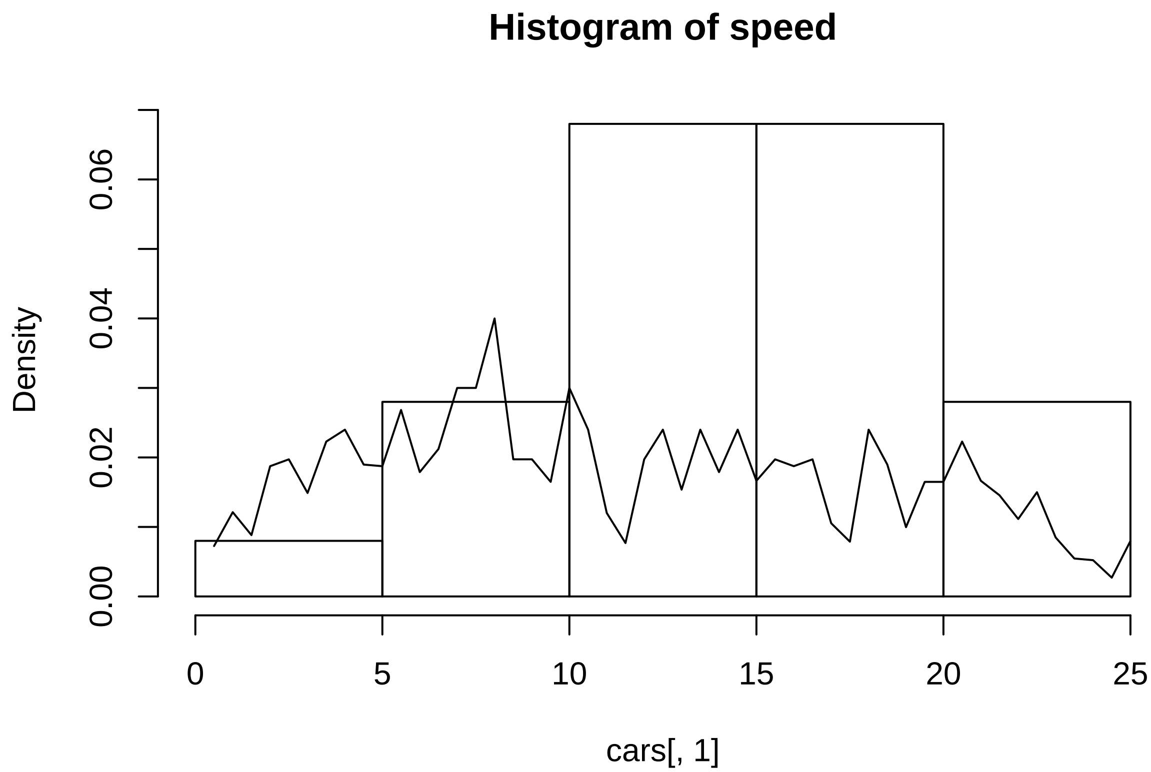
Special Task 1

```
##    sp.task1   asign.1                    M
## 1 0.2883212 0.1722628 mis.error train data
## 2 0.3284672 0.3299270  mis.error test data
```



Comparing the errors for the two methods we can see that the training error from the assignment 1 is smaller compared with the one obtained from implementation of knn algorithm with the same number of neighbors. On the other side the test errors for the 2 methods are very close together.

Special Task 2



Implementing knn kernel density estimation with k=6 on the cars data set and comparing this with the a histogram of the speed column we can see that the density estimation prodused looks very bumpy and doesn't provide a very good approximation on the density of the speed data.

Appendix

```
#cos_distance function ,given 2  matrices X,Y computes
#                         the distance between rows based on cosine similarity

cos_distance<-function(X,Y){
  X<-as.matrix(X)
  Y<-as.matrix(Y)
  X_hat<- t(apply(X,1,function(x){x/sqrt(sum(x*x))})) #sweep(trainS,1,sqrt(rowSums(trainS*trainS)),"/")
  Y_hat<- t(apply(Y,1,function(x){x/sqrt(sum(x*x))}))
  cosine<-X_hat%*%t(Y_hat)
  D<-as.matrix(1-cosine)
  D
}


#knearest function given data,number of neighbors,
#newdata and the true classes of data returns predictions for newdata

knearest<-function(data, k, newdata,cl){
```

```
  cl<-as.data.frame(cl)
  cosdisM<-cos_distance(data,newdata)
  cosdisM<-as.data.frame(cosdisM)
  cosdisM$labels<-cl

  n<-ncol(cosdisM)-1
  bestM<-matrix(0,nrow=k)

  for (i in 1:n){
    ord_cosdisM<-cosdisM[order(cosdisM[,i]),]
    best_ks<-as.vector(ord_cosdisM[2:(k+1),"labels"])
    bestM<-cbind(bestM,best_ks)

  }
  bestM<-as.data.frame(bestM)
  bestM[,1]<-NULL

  #colnames(bestM)<-rownames(newdata)
  #bestM
  #pred_cl<-apply(bestM,2,function(x){names(which(table(x)==max(table(x)))[1])})
  pred_cl<-apply(bestM,2,function(x){ifelse(mean(x)>0.5,1,0)})
  pred_cl

}


spam<-readxl::read_excel("spambase.xlsx")
#spam$Spam<-as.factor(spam$Spam)
#levels(spam$Spam)<-c("not spam","spam")
spam<-as.data.frame(spam)
spam<-spam[sample(nrow(spam)),]
###split data
n=dim(spam)[1]
set.seed(12345)
id=sample(1:n, floor(n*0.5))
train=spam[id,]
test=spam[-id,]

train_X<-train[,!names(train)%in%c("Spam")]
train_labels<-train[,"Spam"]

test_X<-test[,!names(test)%in%c("Spam")]
test_labels<-test[,"Spam"]


#DD1<-cos_distance(train_X,train_X)

mis_error<-function(X,X1){
  n<-length(X)
  return(1-sum(diag(table(X,X1)))/n) #misclassification error function
}

##################################################################
```

```r
K<-knearest(train_X, 30, train_X,train_labels)
mis1<-mis_error(as.vector(K),as.matrix(train_labels))  #0.2627737

K1<-knearest(train_X, 30, test_X,train_labels)
mis2<-mis_error(as.vector(K1),as.matrix(test_labels))  #0.3094891

DF<-data.frame(c(mis1,mis2),c(0.1722628,0.329927))
DF$M<-c("mis.error train data",
                "mis.error test data")
colnames(DF)<-c("sp.task1","asign.1","M")

DF

library(ggplot2)


myplot<- ggplot(data = DF, aes(x =M,y=DF[,1],group=1))+
  geom_line(aes(y=DF[,1],color="asign.1"))+
  geom_line(aes(y=DF[,2],color="sp.task1"))+
  scale_color_manual(labels = c("sp.task1", "asign.1"), values = c("green", "red"))+
  labs(x="models",y="errors")
myplot


#eucl_dist function given a matrix X returns
#                       square matrix of distances between rows.
eucl_dist<-function(X){
  n<-nrow(X)
  dist_mat_e<-matrix(0,nrow = n,ncol=n)
  for (i in 1:nrow(X)){
    for(j in 1:nrow(X)){
      dist_mat_e[i,j]<-sqrt(sum((X[i,]-X[j,])^2))
    }
  }
  dist_mat_e
}

#knn_density function given data,number of neghbors
#                   returns kernel density estimation for each obs

knn_density<-function(data,k){
  distM<-eucl_dist(data)#as.matrix(dist(cars))
  distM<-as.data.frame(distM)
  N<-ncol(distM)
  bestM<-list()
  K<-for (i in 1:N ){
    ord_disM<-distM[order(distM[,i]),]
    best_ks<-ord_disM[(k+1),i]
    bestM[[i]]<-best_ks
    c<-unlist(bestM)
  }
  bestM<-as.data.frame(bestM)
  k/(nrow(data)*c)
```

```
}


dens<-knn_density(cars,6)
hist(cars[,1],freq = F,main = "Histogram of speed")
lines(seq(.5,25,0.5),dens)
```