

## midterm.R

محمد حسن شامخی

Wed Jun 03 23:28:34 2015

```
#3
rm(list=ls())
library(ISLR)

attach(Khan)
Data=data.frame(ytrain,xtrain)

#####PCA
library(pls)
pca.fit=pcr(ytrain~.,data=Data,,scale=TRUE ,validation ="CV")
nData=data.frame(y=ytrain,xtrain %%(pca.fit$projection))
خط قبلی داده های آموزش ما را براساس component های جدید ساخته شده توسط PCA می سازد.

name=names(nData)
ntest=xtest %%(pca.fit$projection)
خط قبلی داده های تست ما را براساس component های جدید ساخته شده توسط PCA می سازد.

nData.test=data.frame(y=ytest,ntest)
colnames(nData.test)=name
#####one vs all
y1=ifelse(nData$y==1,1,0);dat1=data.frame(y=y1,nData[, -1])
y1=ifelse(nData$y==2,1,0);dat2=data.frame(y=y1,nData[, -1])
y1=ifelse(nData$y==3,1,0);dat3=data.frame(y=y1,nData[, -1])
y1=ifelse(nData$y==4,1,0);dat4=data.frame(y=y1,nData[, -1])
rm(y1)

حال بر روی داده های با ویژگی های جدید SVM می زنیم.

#####SVM
library(e1071)
t1=proc.time()
svm1=svm(y~.,data=dat1,family=binomial)
svm2=svm(y~.,data=dat2,family=binomial)
svm3=svm(y~.,data=dat3,family=binomial)
svm4=svm(y~.,data=dat4,family=binomial)
svm.pred1=predict(svm1,nData.test,type="response")
svm.pred2=predict(svm2,nData.test,type="response")
svm.pred3=predict(svm3,nData.test,type="response")
svm.pred4=predict(svm4,nData.test,type="response")
proc.time()-t1

##      user      system elapsed
##      0.10       0.00       0.09
```

```
svm.mat=data.frame(svm.pred1,svm.pred2,svm.pred3,svm.pred4)
svm.class=max.col(svm.mat)
e_svm=mean(svm.class!=ytest)
table(svm.class,ytest)
```

```
##          ytest
## svm.class 1 2 3 4
##          1 1 0 0 0
##          2 1 6 2 0
##          3 0 0 4 0
##          4 1 0 0 5
```

```
e_svm
```

```
## [1] 0.2
```

خطای بدست آمده ۲۰ درصد یعنی همانی که اکثر دوستان سر امتحان بدست آوردند شده است.

```
svm.out=data.frame(svm.class,apply(svm.mat, 1,max)/max(apply(svm.mat, 1,max)))
```

حال می خواهیم از RVM استفاده کنیم.

```
#####RVM
```

```
library("kernlab")
```

```
t1=proc.time()
```

```
rvm1=rvm(y~,data=dat1,family=binomial)
```

```
rvm2=rvm(y~,data=dat2,family=binomial)
```

```
rvm3=rvm(y~,data=dat3,family=binomial)
```

```
rvm4=rvm(y~,data=dat4,family=binomial)
```

```
rvm.pred1=predict(rvm1,nData.test,type="response")
```

```
rvm.pred2=predict(rvm2,nData.test,type="response")
```

```
rvm.pred3=predict(rvm3,nData.test,type="response")
```

```
rvm.pred4=predict(rvm4,nData.test,type="response")
```

```
proc.time()-t1
```

```
##      user  system elapsed
```

```
##      0.28    0.00    0.28
```

```
rvm.mat=data.frame(rvm.pred1,rvm.pred2,rvm.pred3,rvm.pred4)
```

```
rvm.class=max.col(rvm.mat)
```

```
e_rvm=mean(rvm.class!=ytest)
```

```
table(rvm.class,ytest)
```

```
##          ytest
## rvm.class 1 2 3 4
##          1 3 0 0 0
##          2 0 5 0 0
##          3 0 0 6 0
##          4 0 1 0 5
```

```
e_rvm
```

```
## [1] 0.05
```

همانطور که مشاهده شد و بنده انتظار داشتم چون داده ها رو ویژگی Sparse بودند روش RVM بخوبی عمل کرد و

فقط یک داده را اشتباه تخمین زد!!!

```
rvm.out=data.frame(rvm.class,apply(rvm.mat, 1,max)/max(apply(rvm.mat, 1,max)))
```

در این مثال ارزش RVM و مدل های بیزین به خوبی مشخص می شود.