first-classifier.R

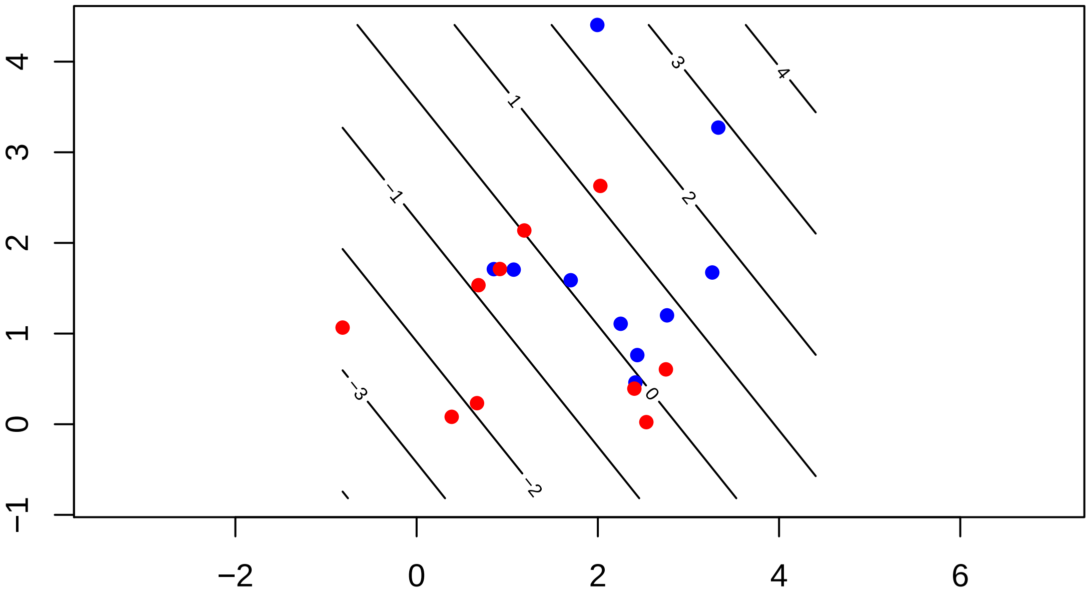
qs26

2019-10-05

```r
# my first kernel classifier
rm(list=ls())
set.seed(0.1)
###############################
## A few kernel functions
###############################
k1 = function(x,y)
  return(sum(x*y))

k2 = function(x,y)
  return(sum(x*y)+1)
k3 = function(x,y)
  return((1+sum(x*y))^2)
d=4
k4 = function(x,y)
  return((1+sum(x*y))^d)
sigma=1
k5 = function(x,y)
  return(exp(-sum((x-y)^2)/(2*sigma^2)))
kappa=1
theta=1
k6 = function(x,y)
  return(tanh(kappa*sum(x*y)+theta))
k = function(x,y)
  return(k1(x,y))
##################################
## generate some data in 2d #####
##################################
n.p=10
n.m=10
n=n.p+n.m
library(mvtnorm)
x.p=rmvnorm(n=n.p,mean=c(2,2),sigma=diag(rep(1,2)))
x.m=rmvnorm(n=n.m,mean=c(1,1),sigma=diag(rep(2,2)))
#x.m=rmvnorm(n=n.m,mean=c(1,1),sigma=diag(rep(10,2)))
y = c(rep(1,n.p),rep(-1,n.m))
x=rbind(x.p,x.m)
#plot(x,col=y+rep(3,n),pch=16)
#points(x=colMeans(x.p)[1],y=colMeans(x.p)[2],col=4,pch=5)
#points(x=colMeans(x.m)[1],y=colMeans(x.m)[2],col=2,pch=5)
###################################
## compute the classifier ########
###################################
k.mm=outer(1:n.m,1:n.m,Vectorize(function(i,j) k(x.m[i,],x.m[j,])))
k.pp=outer(1:n.p,1:n.p,Vectorize(function(i,j) k(x.p[i,],x.p[j,])))
b=(sum(k.mm)/(n.m*n.m)-sum(k.pp)/(n.p*n.p))/2
alpha=c(rep(1/n.p,n.p),rep(-1/n.m,n.m))
```

```
####################################
## evaluate the classifier #######
## over a grid #######
####################################
g.n=50
x.min=min(x)
x.max=max(x)
y.hat=matrix(NA,nrow=g.n,ncol=g.n)
g=seq(from=x.min,to=x.max,length.out=g.n)
for (i in (1:g.n)){
  for (j in (1:g.n)){
    u=c(g[i],g[j])
    k.x=outer(1:n,1,Vectorize(function(i,j) k(x[i,],u)))
    y.hat[i,j]=sum(k.x*alpha)+b
  }
}
contour(x=g,y=g,z=y.hat,asp=1)
points(x.p,col=4,pch=16)
points(x.m,col=2,pch=16)
```


−3
−2
−1
0
1
2
3
4
−2
0
2
4
6
−1
0
1
2
3
4