One Class SVM with linear Kernel is not a good idea

Oliver

Mon Nov 9 17:39:36 2015

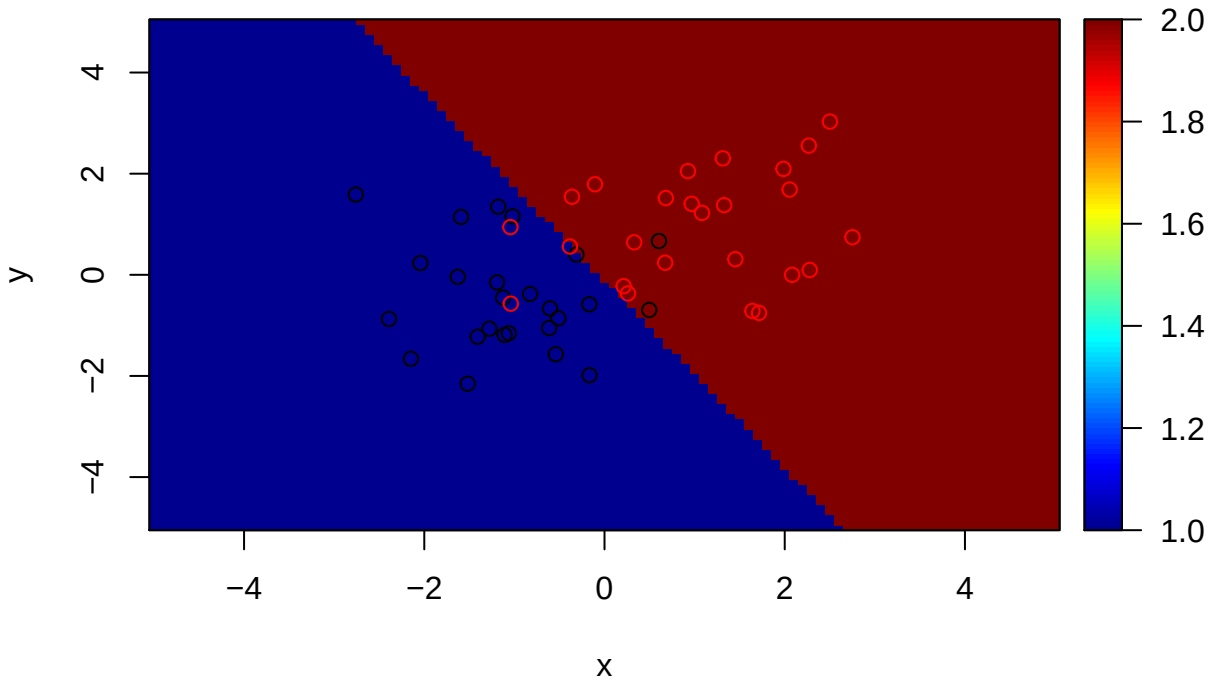
```
library(e1071)

##################
# Regular two class SVM with linear Kernel
n = seq(-5,5,0.1)
y = c(rep(-1,25), rep(1,25))
data = matrix(rnorm(100, y), nrow = 50, ncol = 2)
model <- svm(data, as.factor(y), kernel = 'linear')
print(model)
```

```
##
## Call:
## svm.default(x = data, y = as.factor(y), kernel = "linear")
##
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  linear
##        cost:  1
##       gamma:  0.5
##
## Number of Support Vectors:  18
```

```
preds = outer(n,n, function(x,y) as.numeric(predict(model, matrix(c(x,y), ncol = 2))))
require(plot3D)
```

```
## Loading required package: plot3D
```

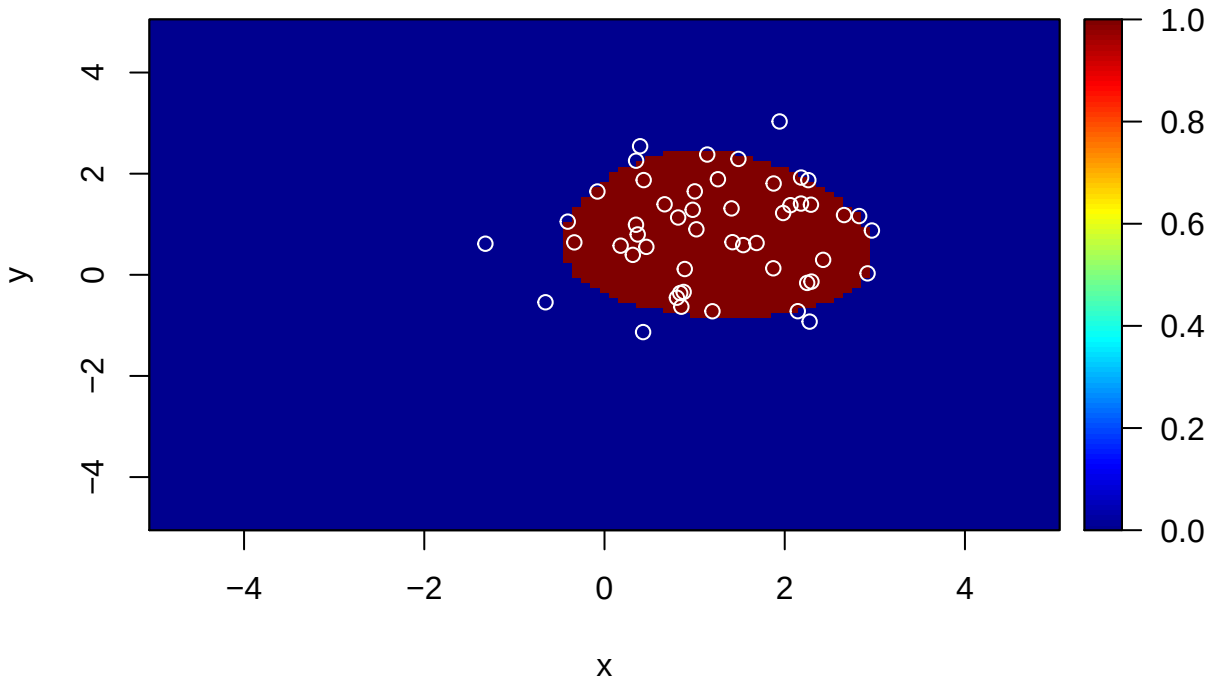
```
image2D(preds, x = n, y = n, main='2 Class Linear Kernel')
points(data, col=as.factor(y))
```




```
###################
# One Class RBF
data = data.frame(x = rnorm(50, 1), y = rnorm(50, 1))
model <- svm(data, type='one-classification', nu=0.2, gamma=0.2)
print(model)
```

```
##
## Call:
## svm.default(x = data, type = "one-classification", gamma = 0.2,
##     nu = 0.2)
##
##
## Parameters:
##    SVM-Type:  one-classification
##  SVM-Kernel:  radial
##       gamma:  0.2
##          nu:  0.2
##
## Number of Support Vectors:  12
```

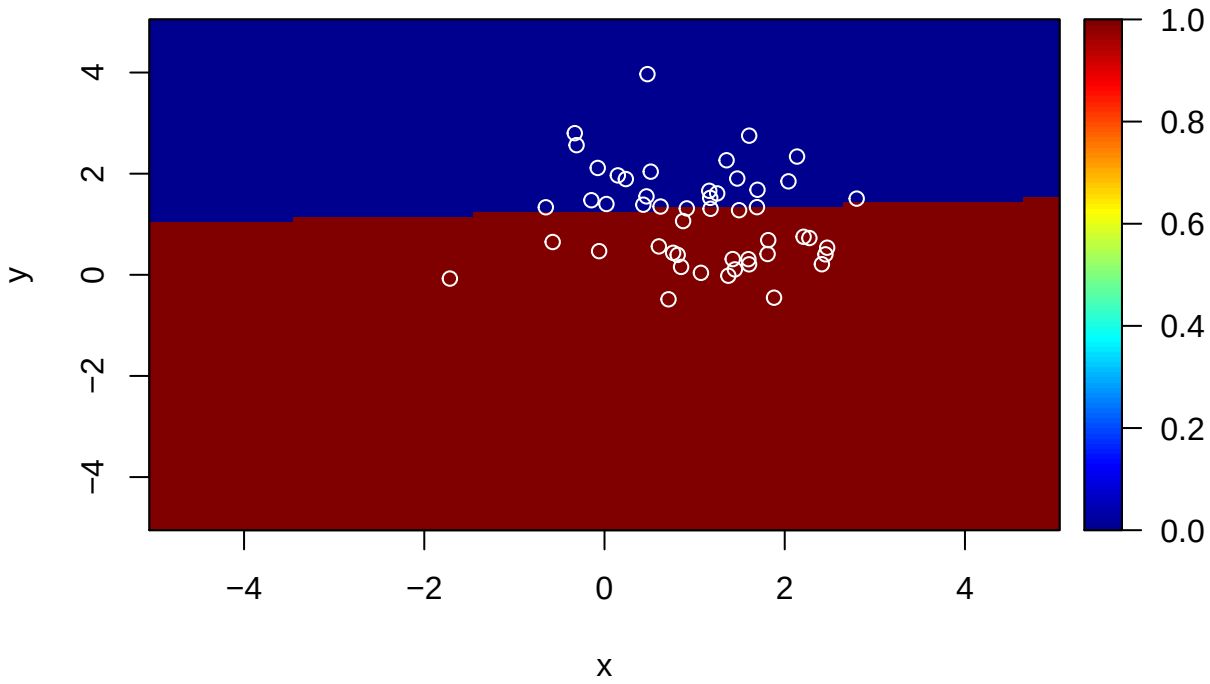
```
preds = outer(n,n, function(x,y) as.numeric(predict(model, data.frame(x = x, y=y))))
image2D(preds, x = n, y = n, rasterImage = FALSE, main='1 Class RBF')
points(data, col='white')
```




```
##################
# One Class Linear Kernel
data = data.frame(x = rnorm(50, 1), y = rnorm(50, 1))
model <- svm(data, type='one-classification', nu=0.2, kernel = 'linear',  main='1 Class RBF')
print(model)
```

```
##
## Call:
## svm.default(x = data, type = "one-classification", kernel = "linear",
##     nu = 0.2, main = "1 Class RBF")
##
##
## Parameters:
##    SVM-Type:  one-classification
##  SVM-Kernel:  linear
##       gamma:  0.5
##          nu:  0.2
##
## Number of Support Vectors:  14
```

```
preds = outer(n,n, function(x,y) as.numeric(predict(model, data.frame(x = x, y=y))))
image2D(preds, x = n, y = n, rasterImage = FALSE, main='1 Class Linear Kernel')
points(data, col='white')
```


1 Class Linear Kernel
y
4
2
0
−2
−4
x
−4
−2
0
2
4
1.0
0.8
0.6
0.4
0.2
0.0