//Assignment No 5

> install.packages("e1071")

trying URL 'http://cran.rstudio.com/bin/windows/contrib/3.3/e1071\_1.6-7.zip'

Content type 'application/zip' length 897494 bytes (876 KB)

downloaded 876 KB

package ‘e1071’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Documents and Settings\Administrator\Local Settings\Temp\Rtmpy6qW2l\downloaded\_packages

> library('e1071')

Warning message:

package ‘e1071’ was built under R version 3.3.1

> library(MASS)

> data("cats")

> cats

> testindex <- sample(index, trunc(length(index)/3))

> testset <- cats[testindex,]

> trainset <- cats[-testindex,]

> model <- svm(Sex~., data = trainset)

> prediction <- predict(model, testset[,-1])

> tab <- table(pred = prediction, true = testset[,1])

> tab

true

pred F M

F 12 6

M 2 28

> classAgreement(tab)

$diag

[1] 0.8333333

$kappa

[1] 0.627907

$rand

[1] 0.7163121

$crand

[1] 0.4288318

> svm\_iris<- svm(Species~.,data=iris)

> summary(svm\_iris)

Call:

svm(formula = Species ~ ., data = iris)

Parameters:

SVM-Type: C-classification

SVM-Kernel: radial

cost: 1

gamma: 0.25

Number of Support Vectors: 51

( 8 22 21 )

Number of Classes: 3

Levels:

setosa versicolor virginica

> x <- subset(iris,select=-Species)

>y<-subset(iris, select=Species)

> pred <- predict(svm\_iris,x)

> svm\_tab <- table(pred =pred, true = y[,1])

> svm\_tab

true

pred setosa versicolor virginica

setosa 50 0 0

versicolor 0 48 2

virginica 0 2 48

> print(svm\_iris)

Call:

svm(formula = Species ~ ., data = iris)

Parameters:

SVM-Type: C-classification

SVM-Kernel: radial

cost: 1

gamma: 0.25

Number of Support Vectors: 51

> summary(svm\_iris)

Call:

svm(formula = Species ~ ., data = iris)

Parameters:

SVM-Type: C-classification

SVM-Kernel: radial

cost: 1

gamma: 0.25

Number of Support Vectors: 51

( 8 22 21 )

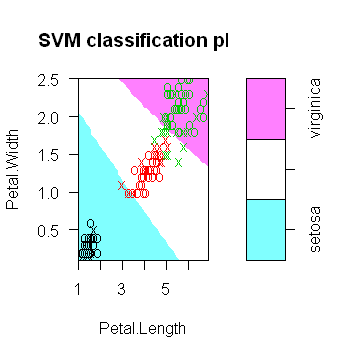
Number of Classes: 3

Levels:

setosa versicolor virginica

>plot(m2, iris, Petal.Width ~ Petal.Length,slice = list(Sepal.Width = 3, Sepal.Length = 4))

|  |  |
| --- | --- |
| slice | a list of named numeric values for the dimensions held constant (only needed if more than two variables are used). Dimensions not specified are fixed at 0. |



Simple SVM example

|  |  |
| --- | --- |
|  | First of all, the plot.svm function assumes that the data varies across two dimensions. The svm function seems to need a data frame as input |

**Factors in R** are stored as a vector of integer values with a corresponding set of character values to use when the **factor** is displayed. The **factor**function is used to create a **factor**. The only required argument to **factor** is a vector of values which will be returned as a vector of **factor**

> day = c(0,1,2,3,4,5,6)

> weather = c(1,0,0,0,0,0,0)

> happy = factor(c(T,F,F,F,F,F,F))

> d = data.frame(day=day, weather=weather, happy=happy)

> model = svm(happy ~ day + weather, data = d)

> plot(model, d)

