Práctica 3 – SVM

Fynn Meffert, José Antonio Chaires Monroy, Luis David López Magaña

Aprendizaje Computacional. Universidad de Málaga.

1. Abstract

En nuestro proyecto, nos sumergimos en las complejidades de las Máquinas de Soporte Vectorial (SVM) utilizando el versátil lenguaje de programación R y el paquete 'kernlab'. Nuestro enfoque se centró en comprender de manera exhaustiva la funcionalidad de las SVM mediante la exploración de diversos conjuntos de datos pequeños. Utilizando la función ksvm, navegamos a través de componentes esenciales de las SVM, como la identificación de vectores de soporte, la extracción de valores de kernel y la determinación del ancho de la frontera de decisión.

Nuestra investigación se extendió para revelar elementos críticos como los vectores de peso y el vector independiente B, componentes fundamentales en la interpretación del modelo SVM. La culminación de nuestros esfuerzos llevó a la derivación de la ecuación del hiperplano, delineando los planos de soporte positivo y negativo. Este viaje analítico nos dotó de la capacidad para clasificar puntos de manera efectiva, una aplicación fundamental de las SVM.

Para demostrar la aplicación práctica de nuestro enfoque, aplicamos estas metodologías al renombrado conjunto de datos IRIS. Al entrenar una máquina de soporte vectorial en el conjunto de datos, desarrollamos con éxito un modelo robusto capaz de clasificar flores según sus atributos distintivos. Este proyecto no solo iluminó el funcionamiento interno de las SVM, sino que también demostró su eficacia en tareas de clasificación del mundo real.

2. Tarea a)

2.1. Script R Usado para las Computaciones

```r
# Importamos las Librerias necesarias
library (kernlab)
library(e1071)
source("lib.R")

# Creamos la funcion que dice a que clase pertenece cada punto


## APARTADO A ##################################

# Creamos el conjunto de datos
dataA <- data.frame(
  x1 = c(0, 4),
  x2 = c(0, 4),
  y = c(1, -1)
)
# Indicamos que la columna y es la importante
dataA$y <- as.factor(dataA$y)

# Creamos el SVM con los datos del A con un kernel lineal
svmA <- svm(y~., dataA , kernel="linear")

#Vectores de soporte
vsA <- dataA[svmA$index,1:2]

# Calculamos los valores del kernel
x1=c(0,0)
x2=c(4,4)
KAA=t (x1) %*% x1
KAB=t (x1) %*% x2
KBB=t (x2) %*% x2

# Vector de pesos normal al hiperplano (W)
# Hacemos el CrosProduct entre los vectores soporte y el coe. de Lagrange
wA <- crossprod(as.matrix(vsA), svmA$coefs)

# Calcular ancho del canal
widthA = 2/(sqrt(sum((wA)^2)))

# Calcular vector B
bA <- -svmA$rho

# Calcular la ecuacion del hiperplano y de los planos de soporte positivo
# y negativo
paste(c("[",wA,"]' * x + [",bA,"] = 0"), collapse=" ")
paste(c("[",wA,"]' * x + [",bA,"] = 1"), collapse=" ")
paste(c("[",wA,"]' * x + [",bA,"] = -1"), collapse=" ")
```

```
# Determinamos a la clase que pertenece cada uno
print_clasificacion(c(5, 6),wA, bA)
print_clasificacion(c(1, -4),wA, bA)
```

2.2. Los Vectores Soporte

$$\overrightarrow{v_1} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \qquad \overrightarrow{v_2} = \begin{pmatrix} 4 \\ 4 \end{pmatrix}$$

2.3. Los Valores del Kernel

$$K_{AA} = 0, \qquad K_{AB} = K_{BA} = 0, \qquad K_{BB} = 32,$$

2.4. El Ancho del Canal

$$ancho \approx 0{,}7071$$

2.5. El Vector $\overrightarrow{w}$

$$\overrightarrow{w} = \begin{pmatrix} -2 \\ -2 \end{pmatrix}$$

2.6. El Vector $\overrightarrow{b}$

$$\overrightarrow{b} = 0$$

2.7. La Ecuación del Hiperplano y de los Planos de Soporte Positivo y Negativo

$$\begin{pmatrix} -2 \\ -2 \end{pmatrix} * \overrightarrow{x} = 0, \qquad \begin{pmatrix} -2 \\ -2 \end{pmatrix} * \overrightarrow{x} = 1, \qquad \begin{pmatrix} -2 \\ -2 \end{pmatrix} * \overrightarrow{x} = -1$$

2.8. Determinación de la Clase

$$clase\left(\begin{pmatrix} 5 \\ 6 \end{pmatrix}\right) = -1, \qquad clase\left(\begin{pmatrix} 1 \\ -4 \end{pmatrix}\right) = 1$$

3. Tarea b)

3.1. Script R Usado para las Computaciones

```r
library(kernlab)
library(e1071)
source("lib.R")

# Define data
data <- data.frame(
  x1 = c(2, 0, 1),
  x2 = c(0, 0, 1),
  y = c(1, -1, -1)
)

# 1. Determine support vectors
data$y <- as.factor(data$y)
svmB <- svm(y~., data, kernel="linear")
supportVectors <- data[svmB$index,1:2]
supportVectors
plot(supportVectors)

# 2. Kernel values

# Exclude last column, since it is only labels
kernel_matrix <- get_kernel_matrix(data[,-3], dot_product_kernel)
kernel_matrix

# 3. Width of street
wB <- crossprod(as.matrix(supportVectors), svmB$coefs)
widthB = 2/(sqrt(sum((wB)^2)))
widthB
# 4. Vector of weights, normal to hyperplane (W)
wB
# 5. Vector B
bB <- svmB$rho
bB
# 6. Hyperplane equation, negative and possitive support planes
paste(c("[",wB,"]' * x + [",bB,"] = 0"), collapse=" ")
paste(c("[",wB,"]' * x + [",bB,"] = 1"), collapse=" ")
paste(c("[",wB,"]' * x + [",bB,"] = -1"), collapse=" ")

# 7. Classification
print_clasificacion(c(5,6), wB, bB)
print_clasificacion(c(1,-4), wB, bB)
```

3.2. Los Vectores Soporte

$$\overrightarrow{v_1} = \begin{pmatrix} 2 \\ 0 \end{pmatrix}, \qquad \overrightarrow{v_2} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \qquad \overrightarrow{v_3} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

3.3. Los Valores del Kernel

$$K_{AA} = 4, \qquad K_{AB} = K_{BA} = 0, \qquad K_{BB} = 0, \quad K_{BC} = K_{CB} = 0, \quad K_{CC} = 2$$

3.4. El Ancho del Canal

$$ancho \approx 1{,}898$$

3.5. El Vector $\overrightarrow{w}$

$$\overrightarrow{w} = \begin{pmatrix} 0{,}9995 \\ -0{,}3335 \end{pmatrix}$$

3.6. El Vector $\overrightarrow{b}$

$$\overrightarrow{b} = 0.\overline{3}$$

3.7. La Ecuación del Hiperplano y de los Planos de Soporte Positivo y Negativo

$$\begin{pmatrix} 0{,}9995 \\ -0{,}3335 \end{pmatrix} * \overrightarrow{x} + 0.\overline{3} = 0, \qquad \begin{pmatrix} 0{,}9995 \\ -0{,}3335 \end{pmatrix} * \overrightarrow{x} + 0.\overline{3} = -1$$

$$\begin{pmatrix} 0{,}9995 \\ -0{,}3335 \end{pmatrix} * \overrightarrow{x} + 0.\overline{3} = -1$$

3.8. Determinación de la Clase

$$clase\left(\begin{pmatrix} 5 \\ 6 \end{pmatrix}\right) = 1, \qquad clase\left(\begin{pmatrix} 1 \\ -4 \end{pmatrix}\right) = 1$$

4. Tarea c)

4.1. Script R Usado para las Computaciones

```r
library(kernlab)

source("lib.R")

# Define data
data <- data.frame(
  x1 = c(2, 2, -2, -2, 1, 1, -1, -1),
  x2 = c(2, -2, -2, 2, 1, -1, -1, 1),
  y = c(1, 1, 1, 1, -1, -1, -1, -1)
)


svm <- ksvm(y~., data, type="C-svc",C = 100, kernel="rbfdot")

# 1. Determine support vectors
supportVectors <- data[svm@SVindex, -3]
supportVectors

# 2. Kernel values

# Exclude last column, since it is only labels
kernel_matrix <- get_kernel_matrix(data[,-3], dot_product_
    kernel)
kernel_matrix

# 3. Width of street
w <- colSums(coef(svm)[[1]] * data[SVindex(svm),])
b <- svm@b

widthB = 2/(sqrt(sum((w)^2)))
widthB

# 4. Vector of weights, normal to hyperplane (W)
W <- w

# 5. Vector B
b

# Plot
plot(x2 ~ x1, data = data, col = ifelse(y == -1, "red", "blue")
    , pch = 19, main = "SVM Decision Boundary", xlab = "x1",
    ylab = "x2")
cat (-w[1]/w[2],"*x+",-b/w[2],"=1")
cat (-w[1]/w[2],"*x+",-b/w[2],"=-1")
cat (-w[1]/w[2],"*x+",-b/w[2],"=0")

abline(b/w[2],-w[1]/w[2])
abline((b+1)/w[2],-w[1]/w[2],lty=2)
abline((b-1)/w[2],-w[1]/w[2],lty=2)

```

```r
# 6. Hyperplane equation, negative and possitive support planes
paste(c("[",w,"]' * x + [",b,"] = 0"), collapse=" ")
paste(c("[",w,"]' * x + [",b,"] = 1"), collapse=" ")
paste(c("[",w,"]' * x + [",b,"] = -1"), collapse=" ")

test.data <- data.frame(
  x1=c(0, 4),
  x2=c(0,4)
)

predict(svm, test.data, type="response")
```

4.2. Los Vectores Soporte

$$\overrightarrow{v_1} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \qquad \overrightarrow{v_2} = \begin{pmatrix} 2 \\ -2 \end{pmatrix}, \qquad \overrightarrow{v_3} = \begin{pmatrix} -2 \\ -2 \end{pmatrix}, \qquad \overrightarrow{v_4} = \begin{pmatrix} -2 \\ 2 \end{pmatrix},$$

$$\overrightarrow{v_5} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad \overrightarrow{v_6} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \qquad \overrightarrow{v_7} = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \qquad \overrightarrow{v_8} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

4.3. Los Valores del Kernel

$$\begin{pmatrix} 8 & 0 & -8 & 0 & 4 & 0 & -4 & 0 \\ 0 & 8 & 0 & -8 & 0 & 4 & 0 & -4 \\ -8 & 0 & 8 & 0 & -4 & 0 & 4 & 0 \\ 0 & -8 & 0 & 8 & 0 & -4 & 0 & 4 \\ 4 & 0 & -4 & 0 & 2 & 0 & -2 & 0 \\ 0 & 4 & 0 & -4 & 0 & 2 & 0 & -2 \\ -4 & 0 & 4 & 0 & -2 & 0 & 2 & 0 \\ 0 & -4 & 0 & 4 & 0 & -2 & 0 & 2 \end{pmatrix}$$

4.4. El Ancho del Canal

$$ancho \approx 0{,}08184$$

4.5. El Vector $\overrightarrow{w}$

$$\overrightarrow{w} = \begin{pmatrix} 6{,}411 * 10^{-5} \\ -1{,}515 * 10^{-3} \\ 1{,}197 \end{pmatrix}$$

4.6. El Vector $\overrightarrow{b}$

$$\overrightarrow{b} = -2{,}060$$

4.7. La Ecuación del Hiperplano y de los Planos de Soporte Positivo y Negativo

$$\begin{pmatrix} 6{,}411 * 10^{-5} \\ -1{,}515 * 10^{-3} \\ 1{,}197 \end{pmatrix} * \overrightarrow{x} + -2{,}060 = 0, \qquad \begin{pmatrix} 6{,}411 * 10^{-5} \\ -1{,}515 * 10^{-3} \\ 1{,}197 \end{pmatrix} * \overrightarrow{x} + -2{,}060 = -1,$$

$$\begin{pmatrix} 6{,}411 * 10^{-5} \\ -1{,}515 * 10^{-3} \\ 1{,}197 \end{pmatrix} * \overrightarrow{x} + -2{,}060 = -1$$

4.8. Determinación de la Clase

$$clase\left(\begin{pmatrix}0\\0\end{pmatrix}\right) = -1, \qquad clase\left(\begin{pmatrix}4\\4\end{pmatrix}\right) = 1$$

5. Tarea d)

5.1. Script R Usado para las Computaciones

```r
library(kernlab)

source("lib.R")

transform <- function(x1, x2) {
  if (sqrt(x1^2 + x2^2) > 2) {
    transformed.x1 <- 4 - x2 + abs(x1 - x2)
    transformed.x2 <- 4 - x1 + abs(x1 - x2)
    return(c(transformed.x1, transformed.x2))
  }
  return(c(x1, x2))
}

apply_transform <- function(row) {
  transformed <- transform(row$x1, row$x2)
  row$x1 <- transformed[1]
  row$x2 <- transformed[2]
  return(row)
}

# Define data
data <- data.frame(
  x1 = c(2, 2, -2, -2, 2, 2, -2, -2, 1, 1, -1, -1),
  x2 = c(2, -2, -2, 2, 2, -2, -2, 2, 1, -1, -1, 1),
  y = c(1, 1, 1, 1, 1, 1, 1, 1, -1, -1, -1, -1)
)

# Apply the transformation to each row using by
transformed_data <- by(data, INDICES = seq_len(nrow(data)), FUN
    = apply_transform)

# Combine the result back into a data frame
data <- do.call(rbind, transformed_data)

svm <- ksvm(y~., data, type="C-svc",C = 100, kernel="vanilladot
   ", scaled=c())

# 1. Determine support vectors
supportVectors <- data[svm@SVindex, -3]
supportVectors

# 2. Kernel values

# Exclude last column, since it is only labels
kernel_matrix <- get_kernel_matrix(data[,-3], dot_product_
   kernel)
kernel_matrix

# 3. Width of street
w <- colSums(coef(svm)[[1]] * data[SVindex(svm),])
```

```r
# (Removes the 'y' column from w vector)
w <- w[-3]
b <- svm@b

widthB = 2/(sqrt(sum((w)^2)))
widthB

# 4. Vector of weights, normal to hyperplane (W)
w

# 5. Vector B
b

# Plot
plot(x2 ~ x1, data = data, col = ifelse(y == -1, "red", "blue")
    , pch = 19, main = "SVM Decision Boundary", xlab = "x1",
    ylab = "x2")
cat (-w[1]/w[2],"*x+",-b/w[2],"=1")
cat (-w[1]/w[2],"*x+",-b/w[2],"=-1")
cat (-w[1]/w[2],"*x+",-b/w[2],"=0")

abline(b/w[2],-w[1]/w[2])
abline((b+1)/w[2],-w[1]/w[2],lty=2)
abline((b-1)/w[2],-w[1]/w[2],lty=2)

# 6. Hyperplane equation, negative and possitive support planes
paste(c("[",w,"]' * x + [",b,"] = 0"), collapse=" ")
paste(c("[",w,"]' * x + [",b,"] = 1"), collapse=" ")
paste(c("[",w,"]' * x + [",b,"] = -1"), collapse=" ")

# 7. Point classification
print_clasificacion(c(8,8), w, b)
print_clasificacion(c(-2,-2), w, b)
```

5.2. Los Vectores Soporte

$$\overrightarrow{v_1} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \qquad \overrightarrow{v_2} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

5.3. Los Valores del Kernel

$$\begin{pmatrix}
8 & 32 & 24 & 32 & 8 & 32 & 24 & 32 & 4 & 0 & -4 & 0 \\
32 & 136 & 96 & 120 & 32 & 136 & 96 & 120 & 16 & 4 & -16 & -4 \\
24 & 96 & 72 & 96 & 24 & 96 & 72 & 96 & 12 & 0 & -12 & 0 \\
32 & 120 & 96 & 136 & 32 & 120 & 96 & 136 & 16 & -4 & -16 & 4 \\
8 & 32 & 24 & 32 & 8 & 32 & 24 & 32 & 4 & 0 & -4 & 0 \\
32 & 136 & 96 & 120 & 32 & 136 & 96 & 120 & 16 & 4 & -16 & -4 \\
24 & 96 & 72 & 96 & 24 & 96 & 72 & 96 & 12 & 0 & -12 & 0 \\
32 & 120 & 96 & 136 & 32 & 120 & 96 & 136 & 16 & -4 & -16 & 4 \\
4 & 16 & 12 & 16 & 4 & 16 & 12 & 16 & 2 & 0 & -2 & 0 \\
0 & 4 & 0 & -4 & 0 & 4 & 0 & -4 & 0 & 2 & 0 & -2 \\
-4 & -16 & -12 & -16 & -4 & -16 & -12 & -16 & -2 & 0 & 2 & 0 \\
0 & -4 & 0 & 4 & 0 & -4 & 0 & 4 & 0 & -2 & 0 & 2
\end{pmatrix}$$

5.4. El Ancho del Canal

$$ancho \approx 1{,}414$$

5.5. El Vector $\overrightarrow{w}$

$$\overrightarrow{w} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

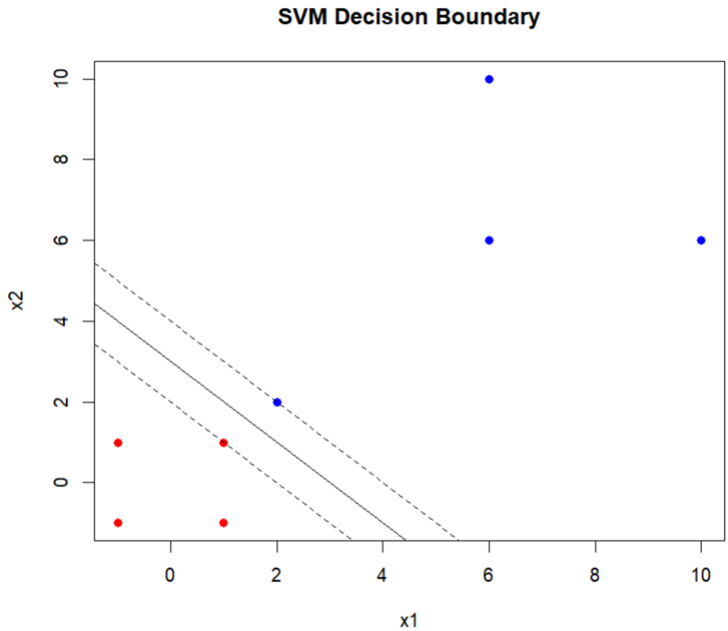
5.6. El Vector $\overrightarrow{b}$

$$\overrightarrow{b} = 3$$

5.7. La Ecuación del Hiperplano y de los Planos de Soporte Positivo y Negativo

$$\begin{pmatrix} 1 \\ 1 \end{pmatrix} * \overrightarrow{x} + 3 = 0, \qquad \begin{pmatrix} 1 \\ 1 \end{pmatrix} * \overrightarrow{x} + 3 = 1, \quad \begin{pmatrix} 1 \\ 1 \end{pmatrix} * \overrightarrow{x} + 3 = -1$$

5.8. Plot



6. Tarea e)

6.1. Script R Usado para las Computaciones

```r
library(kernlab)
source("lib.R")

transform <- function(x1, x2) {
  if (sqrt(x1^2 + x2^2) > 2) {
    transformed.x1 <- 4 - x2 + abs(x1 - x2)
    transformed.x2 <- 4 - x1 + abs(x1 - x2)
    return(c(transformed.x1, transformed.x2))
  }
  return(c(x1, x2))
}

apply_transform <- function(row) {
  transformed <- transform(row$x1, row$x2)
  row$x1 <- transformed[1]
  row$x2 <- transformed[2]
  return(row)
}

# Define data
data <- data.frame(
  x1 = c(3, 3, 6, 6, 1, 0, 0, -1),
  x2 = c(1, -1, 1, -1, 0, 1, -1, 0),
  y = c(1, 1, 1, 1, -1, -1 ,-1 ,-1)
)

# Apply the transformation to each row using by
transformed_data <- by(data, INDICES = seq_len(nrow(data)), FUN = apply_transform)

# Combine the result back into a data frame
data <- do.call(rbind, transformed_data)

svm <- ksvm(y~., data, type="C-svc",C = 100, kernel="vanilladot", scaled=c())

# 1. Determine support vectors
supportVectors <- data[svm@SVindex, -3]
supportVectors

# 2. Kernel values

# Exclude last column, since it is only labels
kernel_matrix <- get_kernel_matrix(data[,-3], dot_product_kernel)
kernel_matrix

# 3. Width of street
w <- colSums(coef(svm)[[1]] * data[SVindex(svm),])
w <- w[-3]
```

```
b <- svm@b

widthB = 2/(sqrt(sum((w)^2)))
widthB

# 4. Vector of weights, normal to hyperplane (W)
w

# 5. Vector B
b

# Plot
plot(x2 ~ x1, data = data, col = ifelse(y == -1, "red", "blue")
    , pch = 19, main = "SVM Decision Boundary", xlab = "x1",
    ylab = "x2")
cat (-w[1]/w[2],"*x+",-b/w[2],"=1")
cat (-w[1]/w[2],"*x+",-b/w[2],"=-1")
cat (-w[1]/w[2],"*x+",-b/w[2],"=0")

abline(b/w[2],-w[1]/w[2])
abline((b+1)/w[2],-w[1]/w[2],lty=2)
abline((b-1)/w[2],-w[1]/w[2],lty=2)

# 6. Hyperplane equation, negative and possitive support planes
paste(c("[",w,"]' * x + [",b,"] = 0"), collapse=" ")
paste(c("[",w,"]' * x + [",b,"] = 1"), collapse=" ")
paste(c("[",w,"]' * x + [",b,"] = -1"), collapse=" ")

# 7. Classify
print_clasificacion(c(4, 5), w, b)
```

6.2. Los Vectores Soporte

$$\overrightarrow{v_1} = \begin{pmatrix} 5 \\ 3 \end{pmatrix}, \qquad \overrightarrow{v_2} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

6.3. Los Valores del Kernel

$$\begin{pmatrix} 34 & 60 & 49 & 75 & 5 & 3 & -3 & -5 \\ 60 & 106 & 87 & 133 & 9 & 5 & -5 & -9 \\ 49 & 87 & 73 & 111 & 8 & 3 & -3 & -8 \\ 75 & 133 & 111 & 169 & 12 & 5 & -5 & -12 \\ 5 & 9 & 8 & 12 & 1 & 0 & 0 & -1 \\ 3 & 5 & 3 & 5 & 0 & 1 & -1 & 0 \\ -3 & -5 & -3 & -5 & 0 & -1 & 1 & 0 \\ -5 & -9 & -8 & -12 & -1 & 0 & 0 & 1 \end{pmatrix}$$

6.4. El Ancho del Canal

$$ancho = 5$$

6.5. El Vector $\overrightarrow{w}$

$$\overrightarrow{w} = \begin{pmatrix} 0{,}32 \\ 0{,}24 \end{pmatrix}$$

6.6. El Vector $\overrightarrow{b}$

$$\overrightarrow{b} = 1{,}32$$

6.7. La Ecuación del Hiperplano y de los Planos de Soporte Positivo y Negativo

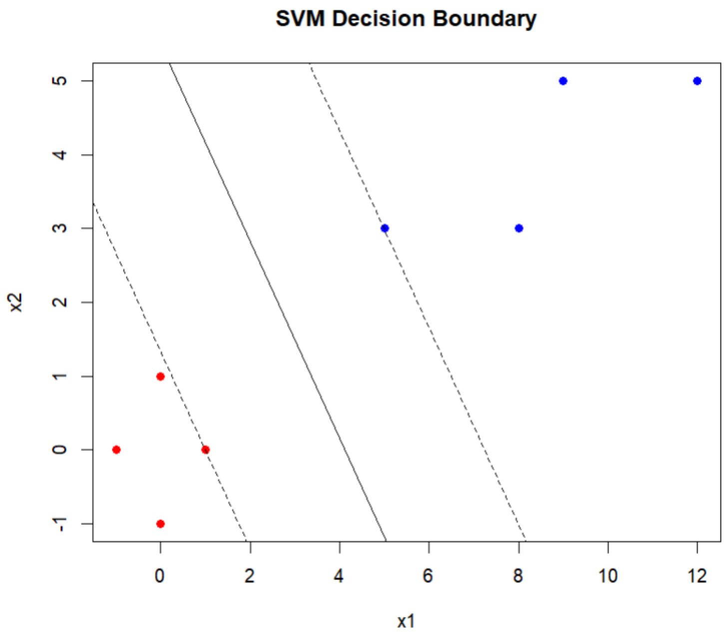
$$\begin{pmatrix} 0{,}32 \\ 0{,}24 \end{pmatrix} * \overrightarrow{x} + 1{,}32 = 0, \qquad \begin{pmatrix} 0{,}32 \\ 0{,}24 \end{pmatrix} * \overrightarrow{x} + 1{,}32 = 1,$$

$$\begin{pmatrix} 0{,}32 \\ 0{,}24 \end{pmatrix} * \overrightarrow{x} + 1{,}32 = -1$$

6.8. Determinación de la Clase

$$clase\left(\begin{pmatrix}4\\5\end{pmatrix}\right) = 1$$

6.9. Plot



7. Tarea f)

7.1. Script R Usado para las Computaciones

```r
library(kernlab)

source("lib.R")

# Define data
data <- iris
svm <- ksvm(Species~., data, type="C-svc",C = 100, kernel="vanilladot")

# 1. Determine support vectors
supportVectors <- data[svm@SVindex, -3]
supportVectors

# 2. Kernel values

# Exclude last column, since it is only labels
kernel_matrix <- get_kernel_matrix(data[,-3], dot_product_kernel)
kernel_matrix

# 3. Width of street
w <- colSums(coef(svm)[[1]] * data[SVindex(svm),])
w <- w[-5]
b <- svm@b

widthB = 2/(sqrt(sum((w)^2)))
widthB

# 4. Vector of weights, normal to hyperplane (W)
w

# 5. Vector B
b

# 6. Hyperplane equation, negative and possitive support planes
paste(c("[",w,"]' * x + [",b,"] = 0"), collapse=" ")
paste(c("[",w,"]' * x + [",b,"] = 1"), collapse=" ")
paste(c("[",w,"]' * x + [",b,"] = -1"), collapse=" ")
```

7.2. Los Vectores Soporte

Sepal.Length	Sepal.Width	Petal.Width	Species
5.1	3.3	0.5	setosa
4.5	2.3	0.3	setosa
4.9	2.4	1.0	versicolor
5.9	3.2	1.8	versicolor
6.3	2.5	1.5	versicolor
6.7	3.0	1.7	versicolor
6.0	2.7	1.6	versicolor
5.1	2.5	1.1	versicolor
4.9	2.5	1.7	virginica
6.0	2.2	1.5	virginica
6.2	2.8	1.8	virginica
6.1	3.0	1.8	virginica
7.2	3.0	1.6	virginica
6.3	2.8	1.5	virginica
6.0	3.0	1.8	virginica

7.3. Los Valores del Kernel

```
      [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11] [,12] [,13] [,14] [,15] [,16] [,17] [,18] [,19]
[1,] 39.30 36.53 36.21 35.35 39.14 42.27 36.42 38.44 33.63 36.86 41.53 37.42 36.00 33.45 44.62 45.55 42.27 39.32 43.43
[2,] 36.53 34.05 33.67 32.88 36.34 39.24 33.80 35.74 31.30 34.33 38.60 34.76 33.54 31.09 41.46 42.21 39.24 36.55 40.39
[3,] 36.21 33.67 33.37 32.58 36.06 38.94 33.56 35.42 31.00 33.97 38.26 34.48 33.18 30.83 41.10 41.95 38.94 36.23 40.01
[4,] 35.35 32.88 32.58 31.81 35.20 38.01 32.76 34.58 30.27 33.17 37.35 33.66 32.40 30.10 40.12 40.94 38.01 35.37 39.06
[5,] 39.14 36.34 36.06 35.20 39.00 42.12 36.30 38.28 33.48 36.68 41.36 37.28 35.82 33.32 44.44 45.42 42.12 39.16 43.24
[6,] 42.27 39.24 38.94 38.01 42.12 45.53 39.22 41.34 36.15 39.59 44.67 40.26 38.66 35.96 48.00 49.10 45.53 42.31 46.72
     [,20] [,21] [,22] [,23] [,24] [,25] [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37] [,38]
[1,] 40.37 40.48 40.04 37.10 38.66 37.42 37.04 38.48 39.81 39.46 36.21 36.37 40.52 41.89 43.79 36.88 37.74 41.34 38.61
[2,] 37.45 37.70 37.17 34.38 35.99 34.76 34.54 35.78 37.02 36.72 33.67 33.86 37.74 38.80 40.59 34.35 35.14 38.49 35.83
[3,] 37.19 37.30 36.89 34.18 35.63 34.48 34.14 35.46 36.68 36.36 33.37 33.52 37.34 38.58 40.33 33.99 34.78 38.09 35.57
[4,] 36.30 36.42 36.01 33.36 34.79 33.66 33.34 34.62 35.81 35.50 32.58 32.73 36.46 37.65 39.36 33.19 33.96 37.19 34.72
[5,] 40.24 40.28 39.90 37.00 38.48 37.28 36.84 38.32 39.64 39.28 36.06 36.20 40.32 41.78 43.66 36.70 37.56 41.14 38.48
[6,] 43.48 43.50 43.13 39.96 41.61 40.26 39.78 41.42 42.81 42.42 38.94 39.09 43.58 45.11 47.16 39.63 40.56 44.43 41.54
     [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49] [,50] [,51] [,52] [,53] [,54] [,55] [,56] [,57]
[1,] 33.98 38.95 38.81 32.06 34.68 38.87 40.39 36.04 40.35 35.70 41.02 38.09 49.18 46.14 48.34 38.36 45.25 41.13 46.00
[2,] 31.60 36.23 36.06 30.01 32.20 36.12 37.47 33.58 37.43 33.18 38.11 35.44 46.18 43.26 45.41 36.11 42.55 38.59 43.09
[3,] 31.32 35.89 35.76 29.57 31.96 35.82 37.21 33.22 37.17 32.90 37.79 35.10 45.42 42.62 44.65 35.47 41.81 38.01 42.49
[4,] 30.58 35.04 34.91 28.89 31.20 34.97 36.32 32.44 36.28 32.12 36.89 34.27 44.40 41.66 43.65 34.69 40.88 37.16 41.53
[5,] 33.84 38.78 38.66 31.84 34.56 38.72 40.26 35.86 40.22 35.56 40.86 37.92 48.80 45.82 47.96 38.04 44.88 40.84 45.70
[6,] 36.54 41.88 41.77 34.39 37.32 41.89 43.52 38.74 43.44 38.40 44.13 40.95 52.84 49.64 51.95 41.19 48.62 44.22 49.53
     [,58] [,59] [,60] [,61] [,62] [,63] [,64] [,65] [,66] [,67] [,68] [,69] [,70] [,71] [,72] [,73] [,74] [,75] [,76]
[1,] 35.59 46.07 38.25  34.7 42.89 40.50 43.54 40.97 47.30 41.36 41.23 41.62 39.53 43.65 43.17 43.18 43.15 45.05 46.44
[2,] 33.41 43.30 35.86  32.7 40.21 38.20 40.87 38.40 44.41 38.74 38.72 39.28 37.16 40.87 40.55 40.67 40.53 42.32 43.62
[3,] 32.91 42.56 35.36  32.1 39.63 37.44 40.23 37.86 43.69 38.22 38.10 38.48 36.54 40.33 39.89 39.91 39.87 41.62 42.90
[4,] 32.18 41.61 34.57  31.4 38.74 36.62 39.33 37.01 42.71 37.36 37.25 37.64 35.73 39.42 39.00 39.03 38.98 40.69 41.94
[5,] 35.34 45.70 38.00  34.4 42.60 40.12 43.22 40.70 46.94 41.10 40.92 41.22 39.22 43.38 42.84 42.80 42.82 44.70 46.08
[6,] 38.22 49.47 41.17  37.2 46.16 43.38 46.81 44.07 50.83 44.54 44.25 44.66 42.43 47.06 46.38 46.37 46.34 48.39 49.90
     [,77] [,78] [,79] [,80] [,81] [,82] [,83] [,84] [,85] [,86] [,87] [,88] [,89] [,90] [,91] [,92] [,93] [,94] [,95]
[1,] 46.76 47.01 43.05 40.37 38.67 38.65 41.27 42.37 40.34 44.82 47.32 42.44 41.32 39.06 39.39 43.89 40.92 35.75 40.27
[2,] 44.00 44.17 40.40 37.93 36.37 36.35 38.76 39.82 37.76 41.92 44.43 40.03 38.70 36.71 36.99 41.17 38.46 33.60 37.80
[3,] 43.20 43.43 39.78 37.31 35.75 35.73 38.14 39.16 37.28 41.40 43.71 39.23 38.18 36.11 36.41 40.55 37.82 33.06 37.22
[4,] 42.24 42.46 38.89 36.48 34.96 34.94 37.29 38.29 36.44 40.46 42.73 38.37 37.32 35.31 35.60 39.64 36.98 32.33 36.39
[5,] 46.36 46.64 42.74 40.06 38.36 38.34 40.96 42.04 40.10 44.56 46.96 42.04 41.06 38.76 39.10 43.58 40.60 35.48 39.98
[6,] 50.20 50.56 46.31 43.32 41.50 41.46 44.33 45.57 43.46 48.30 50.87 45.51 44.46 41.97 42.32 47.20 43.94 38.37 43.29
     [,96] [,97] [,98] [,99] [,100] [,101] [,102] [,103] [,104] [,105] [,106] [,107] [,108] [,109] [,110] [,111] [,112]
[1,] 41.81 41.48 44.03 36.98  41.13  47.18  42.41  50.13  45.64  47.09  52.68  37.08  50.74  46.28  52.82  47.75  45.47
[2,] 39.17 38.89 41.34 34.71  38.59  44.27  39.90  47.21  42.93  44.29  49.66  34.85  47.83  43.69  49.58  44.85  42.84
[3,] 38.63 38.33 40.68 34.19  38.01  43.67  39.28  46.39  42.25  43.59  48.74  34.37  46.95  42.85  48.86  44.19  42.10
[4,] 37.76 37.47 39.77 33.43  37.16  42.71  38.43  45.38  41.33  42.64  47.68  33.63  45.93  41.93  47.78  43.22  41.19
[5,] 41.54 41.20 43.70 36.72  40.84  46.88  42.10  49.72  45.30  46.74  52.22  36.84  50.30  45.86  52.46  47.42  45.10
[6,] 44.96 44.61 47.31 39.73  44.22  50.89  45.61  53.88  49.05  50.68  56.58  39.89  54.45  49.65  56.92  51.38  48.85
     [,113] [,114] [,115] [,116] [,117] [,118] [,119] [,120] [,121] [,122] [,123] [,124] [,125] [,126] [,127] [,128]
[1,]  48.60  41.22  42.86  47.30  47.01  56.01  51.83  41.60  49.85  41.76  52.47  44.94  49.14  51.28  44.78  44.97
[2,]  45.74  38.83  40.30  44.42  44.21  52.57  48.99  39.30  46.87  39.24  49.53  42.33  46.15  48.24  42.14  42.25
[3,]  44.98  38.19  39.70  43.78  43.51  51.79  47.97  38.54  46.13  38.68  48.55  41.61  45.47  47.44  41.46  41.63
[4,]  44.00  37.37  38.84  42.82  42.56  50.64  46.94  37.72  45.12  37.84  47.50  40.71  44.47  46.40  40.56  40.72
[5,]  48.22  40.90  42.56  46.98  46.66  55.62  51.32  41.22  49.48  41.48  51.98  44.58  48.80  50.88  44.44  44.66
[6,]  52.26  44.33  46.20  50.96  50.52  60.28  55.64  44.58  53.66  44.96  56.30  48.27  52.89  55.08  48.12  48.36
     [,129] [,130] [,131] [,132] [,133] [,134] [,135] [,136] [,137] [,138] [,139] [,140] [,141] [,142] [,143] [,144]
[1,]  45.86  50.54  50.92  56.99  45.88  45.23  43.49  53.23  47.51  46.85  44.46  49.46  48.50  49.50  42.41  49.34
[2,]  43.18  47.60  48.04  53.51  43.20  42.57  40.97  50.19  44.55  44.02  41.76  46.53  45.61  46.57  39.90  46.38
[3,]  42.46  46.76  47.12  52.69  42.48  41.87  40.27  49.25  43.97  43.36  41.16  45.77  44.89  45.81  39.28  45.66
[4,]  41.54  45.74  46.10  51.52  41.56  40.96  39.40  48.18  43.00  42.41  40.26  44.77  43.91  44.81  38.43  44.66
[5,]  45.50  50.12  50.46  56.58  45.52  44.88  43.14  52.76  47.22  46.52  44.16  49.08  48.14  49.12  42.10  48.98
[6,]  49.32  54.22  54.64  61.28  49.36  48.54  46.64  57.20  51.24  50.37  47.82  53.19  52.23  53.27  45.61  53.12
     [,145] [,146] [,147] [,148] [,149] [,150]
[1,]  49.22  48.13  44.26  47.05  46.98  43.95
[2,]  46.23  45.29  41.75  44.25  44.04  41.27
[3,]  45.55  44.55  40.99  43.55  43.48  40.69
[4,]  44.55  43.58  40.11  42.60  42.52  39.80
[5,]  48.88  47.76  43.88  46.70  46.70  43.66
[6,]  53.05  51.80  47.53  50.60  50.66  47.28
[ reached getOption("max.print") -- omitted 144 rows ]
```

7.4. El Ancho del Canal

$$ancho \approx 0{,}3397$$

7.5. El Vector $\overrightarrow{w}$

$$\overrightarrow{w} = \begin{pmatrix} -5{,}469 \\ 0{,}2363 \\ 1{,}962 \\ 0{,}9175 \end{pmatrix}$$

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
-5.4691320	0.2362545	1.9623159	0.9175388

7.6. El Vector $\overrightarrow{b}$

$$\overrightarrow{b} = \begin{pmatrix} -1{,}474 & -0{,}2958 & 10{,}58 \end{pmatrix}$$

7.7. La Ecuación del Hiperplano y de los Planos de Soporte Positivo y Negativo

$$\begin{pmatrix} -5{,}469 \\ 0{,}2363 \\ 1{,}962 \\ 0{,}9175 \end{pmatrix} * \overrightarrow{x} + \begin{pmatrix} -1{,}474 & -0{,}2958 & 10{,}58 \end{pmatrix} = 0,$$

$$\begin{pmatrix} -5{,}469 \\ 0{,}2363 \\ 1{,}962 \\ 0{,}9175 \end{pmatrix} * \overrightarrow{x} + \begin{pmatrix} -1{,}474 & -0{,}2958 & 10{,}58 \end{pmatrix} = 1$$

$$\begin{pmatrix} -5{,}469 \\ 0{,}2363 \\ 1{,}962 \\ 0{,}9175 \end{pmatrix} * \overrightarrow{x} + \begin{pmatrix} -1{,}474 & -0{,}2958 & 10{,}58 \end{pmatrix} = -1$$