Efecto de casos anómalos en Máquinas de vectores de soporte

Por
Alba Milena Restrepo Henao
Tesis sometida en cumplimiento parcial de los requerimientos para el grado de
MAESTRO EN CIENCIAS
en
MATEMATICAS (ESTADISTICA)

UNIVERSIDAD DE PUERTO RICO
RECINTO UNIVERSITARIO DE MAYAGUEZ
Marzo, 2009

Aprobada por:

_____	_____
Edgar Acuña Fernández, Ph.D. Presidente, Comité Graduado	Fecha
_____	_____
Edgardo Lorenzo, Ph.D. Miembro, Comité Graduado	Fecha
_____	_____
Robert W. Smith, Ph.D. Miembro, Comité Graduado	Fecha
_____	_____
Rosario A. Ortiz Rodriguez, Ph.D Representante de Estudios Graduados	Fecha
_____	_____
Julio C. Quintana Díaz, Ph.D. Director del Departamento	Fecha

Abstract of thesis presented to the Graduate School
of the University of Puerto Rico in Partial Fulfillment of the
Requirements for the Degree of Master in Mathematics (Statistics)

Effects of outliers in Support Vector Machines

By
Alba.M Restrepo Henao

March 2009

Chair: Edgar Acuña, Ph.D.
Major Department: Mathematical Sciences

Support Vector Machines (SVM) is a new technique of classification that has received much attention in recent years. In many applications, the SVM has shown better performance than machine learning methods, and it has been introduced as a powerful tool for solving classification problems. The SVM was originally developed for binary classification, but was later generalized to problems with various classes using different approaches. Currently, the SVM can be applied to an extensive list of scientific and real life problems. This thesis describes the SVM method for the separable case, the non-separable case, and for multiple classes. Several outlier detection methods are discussed including the support vector description method as well as the use of SVM for one class classification to detect outliers. Experiments with the SVM classifier were made and it is empirically shown that after the elimination of the detected outliers the misclassification error rate of the SVM classifier is improved. All experiments were carried out on 5 data sets available at the Machine Learning Database Repository of the University California, Irvine.

Resumen de Disertación presentado a Oficina de Estudios Graduados
de la Universidad de Puerto Rico como requisito parcial de los
Requerimientos para el grado de Maestría en Ciencias

Efectos de casos anómalos en Maquinas de vectores de soporte

Por
Alba. M Restrepo Henao

Marzo 2009

Consejero: Edgar Acuña, Ph.D.
Departamento: Departamento de Ciencias Matemáticas

Las Máquinas de Soporte Vectorial (SVM), son una nueva técnica de clasificación que ha recibido mucha atención en años recientes. En muchas aplicaciones, el SVM ha mostrado tener mejor desempeño que las máquinas de aprendizaje y ha sido introducido como una herramienta poderosa para resolver problemas de clasificación. Originalmente, el SVM fue desarrollado para clasificación binaria, pero fue posteriormente más tarde generalizado para varias clases usando diferentes variantes. Actualmente el SVM se puede aplicar a una lista extensa de problemas científicos y de la vida real. La presente tesis describe el método SVM, tanto para el caso separable, como el no separable, y para múltiple clases. Varios métodos para detectar "outliers" son discutidos incluyendo el método de descripción por vectores de soporte así como el clasificador SVM con una sola clase para detectar "outliers". Se realizaron experimentos con el clasificador SVM con una sola clase, y se realizan experimentos con el SVM en clasificación y se prueba empíricamente que después de la eliminación de los "outliers" encontrados, se logra una mejora en la tasa de error de mala clasificación del clasificador SVM. Todos los experimentos se realizaron en 5 conjuntos de datos disponible en el repositorio de bases de datos Máquinas de aprendizaje de la Universidad de California en Irvine.

A mi amado niño Brandon Stiven y mis sobrinos Juan Pablo y Sara, quien viene en camino. Por ser mi inspiración y mi alegría. Su dulzura y ternura fueron mi guía, y sus sonrisas mi apoyo.

AGRADECIMIENTOS

A mi consejero Dr. Edgar Acuña, presidente de mi comité, por sus consejos y su apoyo incondicional durante este trabajo de tesis.

A mi comité graduado, Dr. Edgardo Lorenzo, Dr. Robert Smith, por toda la ayuda que me brindaron para terminar mi tesis.

Al Departamento de Ciencias Matemáticas por darme la oportunidad de realizar mí grado en esta institución.

A los miembros de CASTLE por compartir conmigo esta etapa.

A mis padres por darme la vida, sus valiosos consejos y su educación, por siempre estar ahí, a pesar de la distancia. Por ser esas personas en las siempre que se puede confiar. Gracias por ser mi guía desde mi principio hasta lo que soy hoy día.

A mis hermanas por ser mis confidentes y darme ánimo cada día.

A la Oficina de Investigación Naval (ONR) de los Estados Unidos por el apoyo económico recibido a través del Grant N00014-06-1098.

.

.

Tabla de Contenido

Lista de Tablas

Lista de Figuras

Lista de Abreviaturas

KKT Karush-Kuhn-Tucker.

SMO Algoritmo de Optimización Mínima Secuencial.

SV Vectores de soporte

SVDD Descripción de datos usando vectores de soporte.

SVM Maquinas de vectores de soporte

CAPITULO I

INTRODUCCIÓN

Las máquinas de vectores de soporte (SVM por sus siglas en ingles, "Support Vector Machine"), fueron desarrolladas por Vapnik, primero para el problema de clasificación y extendidas posteriormente a regresión por el mismo autor en los laboratorios de AT&T [VGS97]. SVM ha ganado gran popularidad como herramienta para la identificación de sistemas no lineales, debido principalmente a que está basado en el principio de minimización del riesgo estructural (SRM por sus siglas en inglés, "Structural Risk Minimization"). Este principio de minimización es un proceso de inferencia desarrollado por Vapnik sobre la teoría de aprendizaje estadístico. El principio de SRM proporciona un balance entre la complejidad del espacio de hipótesis (la dimensión VC) y la calidad del ajuste de un conjunto de entrenamiento (error empírico). La dimensión VC, llamada así por Vapnik-Chervonenkis, es una medida de la capacidad de un algoritmo para llevar a cabo clasificación estadística. Está definida como la cardinalidad del mayor conjunto de puntos que el algoritmo puede separar. El concepto fue originalmente definido por Vapnik y A. Chervonenkis [VC71].

Algunas de las razones por las que este método ha tenido éxito es que no padece de mínimos locales y el modelo solo depende de las instancias con más información, llamados vectores de soporte (SV por sus siglas en ingles, "Support Vectors").

Uno de los hechos más atrayentes que tiene el SVM es la habilidad de condensar la información de la muestra de entrenamiento y la “sencilla generalización” para usar familias de superficies de decisión no lineales.

La idea principal del SVM es la siguiente: Dada una muestra de entrenamiento S, la cual contiene instancias provenientes de dos clases, un SVM separa las clases a través de un hiperplano determinado por ciertos puntos de S, denominados vectores de soporte. Un conjunto

linealmente separable, es decir un conjunto donde se puede hallar un hiperplano que separe totalmente las clases. En este caso, el hiperplano hallado maximiza el margen, es decir, el doble de la distancia mínima a cualquiera de las dos clases. Por lo tanto, todos los vectores de soporte se encuentran a la mínima distancia del hiperplano, esto se denomina como margen de soporte.

Si el conjunto S no es linealmente separable, o no se conoce si es o no linealmente separable, el problema de encontrar el hiperplano de separación óptimo se hace más complejo. En la mayoría de los casos la separación lineal, es una hipótesis difícil de tener en la práctica. Afortunadamente, la teoría puede ser extendida a superficies de separación no lineales por medio de una función de R^n a Z, la cual transforma los puntos originales a puntos de características en un espacio de Hilbert Z de dimensión mayor a través de funciones kernels [CS04]. Esto es, si $\mathbf{x} \in R^n$ es un punto original, entonces $\varphi(\mathbf{x})$ es su correspondiente punto de característica, donde φ es una función en R^n. La función kernel realiza una transformación no lineal del espacio original a un espacio de mayor dimensión donde el problema es más probable que sea linealmente separable. El SVM usualmente usa los siguientes kernels: lineal, polinomial, radial y sigmoidal.

Entre las variantes del SVM tenemos el C-clasificación, nu–clasificación, one–clasificación, e–regresión, y nu–regresión.

1.1 Motivación del uso de SVM

El SVM usualmente evita la sobreestimación y tiene mejor capacidad de generalización, por lo cual algunas veces se desempeña mejor que otros clasificadores. El entrenamiento es relativamente fácil, además no tiene óptimo local. El algoritmo escala relativamente bien para datos en espacios dimensionales altos. El SVM transforma los datos a un espacio de dimensión mayor donde es posible encontrar una hipersuperficie de separación lineal.
Una de las debilidades del SVM es que necesita una "buena" función kernel y parámetros de inicialización. Se necesitan metodologías eficientes para encontrar dichos parámetros.

1.2 Objetivos

El objetivo principal de esta tesis es explicar el efecto que tiene la presencia de casos anómalos en la muestra de entrenamiento en el rendimiento del clasificador SVM. En particular, se aplicará el mismo SVM para detectar casos anómalos en la muestra de entrenamiento.

Objetivos específicos:

- Describir el SVM para el caso de dos clases, y para el caso de varias clases. Así como sus aplicaciones a conjuntos de datos usados para clasificación supervisada.
- Comparar el rendimiento de clasificadores SVM usando diferentes kernels, utilizando para ello librerías de R. La comparación es hecha en cinco conjuntos de datos disponibles en el repositorio la Universidad de California en Irvine (UCI). La estimación de la tasa de error de mala clasificación es hecha usando validación cruzada.
- Detectar los casos anómalos en la muestra de entrenamiento, usando el clasificador SVM para una sola clase.
- Comparar la detección de casos anómalos usando SVM con otros métodos de detección de casos anómalos.

1.3 Organización de la tesis

La presente tesis está estructurada en seis capítulos. En el segundo capítulo se introducen conceptos básicos del SVM tales como la dimensión VC y la definición de kernels y sus propiedades. En el tercer capítulo se describe el método SVM, tanto para el caso separable, como para el no separable, así como para dos clases y para varias clases. También se describe técnicas para solucionar el problema de optimización del problema cuadrático del SVM. En el cuarto capítulo se describen tres métodos para detectar datos anómalos, y se realizan los experimentos en los siguientes conjuntos de datos Iris, Bupa, Diabetes, Vehículo e Ionosfera. Además, se introduce el método de descripción de datos usando vectores de soporte (SVDD) para detectar "outliers" y su relación con el clasificador SVM para una sola clase. Además, se

describe un método que permite mejorar el rendimiento del clasificador SVM aplicando una penalidad a los “outliers”.

En el quinto capítulo, primero se realizan experimentos con el SVM para clasificación y luego se eliminan los datos anómalos hallados por lo métodos del tercer capítulo. Finalmente, se aplica de nuevo el clasificador SVM para ver si los datos anómalos afectan o no el rendimiento del clasificador. También se detectan los casos anómalos usando SVM para clasificación de una clase, luego se eliminan estos datos anómalos hallados y se aplica de nuevo el clasificador SVM para observar el rendimiento.

En el sexto y último capítulo, se presentan las conclusiones de los resultados obtenidos y recomendaciones para trabajos futuros.

CAPITULO II

CONCEPTOS BASICOS DE SVM

2.1 La dimensión VC

Supongamos que tenemos un conjunto de datos de *m* observaciones, obtenidos de una distribución conjunta P(x,y)

$$(\mathbf{x}_1, y_1),...,(\mathbf{x}_m, y_m), x_i \in R^n, y_i \in \{-1,1\}.$$

y que tenemos un conjunto de funciones de decisión

$$\{f_\lambda : \lambda \in \Lambda\}$$

Donde Λ es un conjunto índice.

$$f_\lambda : R^n \to \{-1,1\}$$

Para una función dada f_λ, el riesgo esperado al predecir el valor de y para un **x** dado usando f_λ, denotado por $R(\lambda)$, es el posible error promedio cometido por f_λ. Esto es,

$$R(\lambda) = \int \frac{1}{2} |f_\lambda(\mathbf{x}) - y| dP(\mathbf{x}, y)$$

$R(\lambda)$ es una medida de cuán bien una función de decisión f_λ predice el valor ***y*** para una entrada x, y P(**x**,y) es la función de probabilidad conjunta de (**x**,y).

Como la función de distribución conjunta P es desconocida, entonces el riesgo debe ser inducido en base al conjunto de datos tomados. Luego, el riesgo actual es aproximado por el llamado riesgo empírico (o error de entrenamiento) $R_{emp}(\lambda)$ y está definida por

$$R_{emp}(\lambda) = \frac{1}{m} \sum_{i=1}^{m} |f_\lambda(\mathbf{x_i}) - y_i|$$

Donde *m* es el número de muestras.

La dimensión VC (dimensión de Vapnik- Chervonenkis) de un conjunto de funciones es el número máximo, h, de vectores $Z_1,...,Z_h$ que pueden ser separados en dos clases en todas las 2^h formas posibles, mediante el uso de funciones aplicadas sobre el conjunto. En otras palabras, la dimensión VC es el número máximo de vectores que pueden ser separados por el conjunto de funciones. En la Figura 2.1, cada línea discontinua separa tres vectores en dos clases. Para tres vectores, el máximo posible de formas de separarlos en dos clases por una función que es una línea recta es de 8. Luego, la dimensión VC-de un conjunto de líneas rectas es 3.

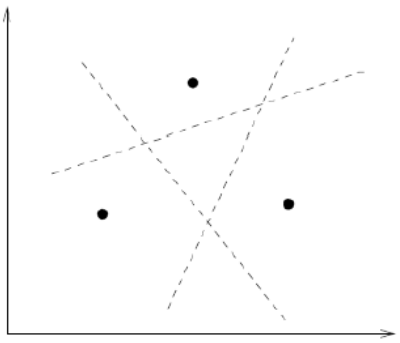


Figura 2. 1 La dimensión VC de una función lineal {f} en un plano es 3, porque ellas pueden separar 3 vectores.

Por otro lado, la Figura 2.2 muestra que ninguna línea recta puede separar cuatro vectores en dos clases. La dimensión VC del conjunto de funciones lineales en el espacio n-dimensional es n+1 [Va95].

La habilidad de un conjunto de funciones (un espacio de hipótesis) para dividir un conjunto de instancias, está estrechamente relacionado con la maximización del margen del espacio de hipótesis. La dimensión VC de un espacio de hipótesis es una medida de la complejidad o expresividad del espacio de hipótesis. Cuanto mayor es la dimensión VC, más grande es la capacidad de una máquina de aprendizaje para aprender sin ningún error de clasificación.

En los años setenta, Vapnik y Chervonenkis [Va95] dieron una cota superior del riesgo esperado, R, con una probabilidad de $1-\psi$, dada por:

$$R(\lambda) \le R_{emp}(\lambda) + \sqrt{\frac{h(\ln\frac{2m}{h}+1)-\ln\frac{\psi}{4}}{m}} \qquad \forall\lambda \in \Lambda \qquad [2.1]$$

Donde h es la dimensión VC de f_λ y *m* es el número de muestras. El segundo término del lado derecho de la ecuación [2.1] se llama la confianza VC.

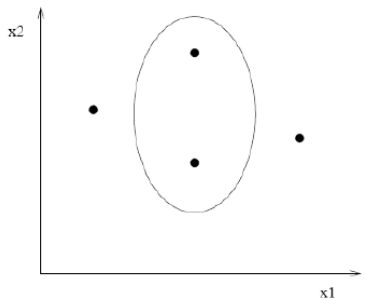


Figura 2. 2 Separando 4 vectores. Ninguna línea recta puede separar 4 vectores. En todas las 2^4 posibles maneras

Para obtener una R pequeña, se requiere un pequeño R_{emp} y una pequeña proporción de *h/l* al mismo tiempo. R_{emp} depende del f_λ y decrece con el incremento de *h*. Se tiene que elegir una h óptima, especialmente cuando el número de instancias *m*, es pequeño, para poder obtener un buen rendimiento. Para esto se usa el principio de minimización del riesgo estructural.

2.2 Minimización del riesgo estructural

Sea $S = \{f_\lambda : \lambda \in \Lambda\}$, un conjunto de funciones y $S_k = \{f_\lambda : \lambda \in \Lambda_k\}$, donde Λ_k un subconjunto de Λ. Definimos una estructura de subconjuntos anidados

$$s_1 \subset s_2 \subset \ldots \subset s_n \subset \ldots \qquad [2.2]$$

de tal forma que la correspondiente dimensión VC de cada subconjunto S_n satisface la condición:

$h_1 \leq h_2 \leq \ldots \leq h_n \leq \ldots$

Cada elección de una estructura S_i produce un algoritmo de aprendizaje. Para un conjunto dado de *m* muestras $z_1,\ldots,z_m$, el principio de mínimo riesgo estructural (SRM, por sus siglas en

inglés). elige la función $f_{\lambda_m^n}$ del conjunto $\{f_\lambda, \lambda \in \Lambda\}$, para el cual el segundo término de la ecuación [2.1] es mínimo.

Sin embargo, implementar SRM puede ser difícil debido a que la dimensión VC de S_n podría ser difícil de calcular. Inclusive, si podemos calcular h_n de S_n, la búsqueda de

$\min\left(R_{emp}(\lambda) + \sqrt{\frac{h_n}{l}} \right)$ entre las h_n es difícil.

Las máquinas de soporte vectorial, SVM, son capaces de alcanzar el objetivo de minimizar la cota superior de $R(\lambda)$ minimizando una cota en la dimensión VC, h, y $R_{emp}(\lambda)$ al mismo tiempo, durante el entrenamiento. La estructura en la que se basa el SVM es un conjunto de hiperplanos de separación.

Sea X un espacio de instancias n-dimensionales. Se tiene un conjunto de vectores $\{x_1,...,x_m\}$ donde $x_i \in X$. Cada hipersuperficie $\mathbf{w} \bullet \mathbf{x} + b = 0$ corresponderá a un par canónico (único par) $(\mathbf{w}, b)$ si ponemos una restricción que

$$\min || \mathbf{w} \bullet \mathbf{x_i} + b || = 1 \qquad i = 1,...,m \qquad [2.3]$$

Esto significa que el punto más cercano al hiperplano tiene una distancia de $1/\|\mathbf{w}\|$. La dimensión VC del hiperplano canónico es n+1. Para que la minimización de riesgo estructural sea aplicable, se tiene que construir conjuntos de hiperplanos de diversas dimensiones VC a fin de que podamos minimizar la dimensión VC y el riesgo empírico simultáneamente. Esto se logra añadiendo una restricción a **w** de la siguiente forma.

Sea R el radio de la bola más pequeña $B_{x_1,...,x_l}$ que contiene a $\{x_1,...,x_l\}$.

También definimos una función de decisión $f_{\mathbf{w},b}$ tal que:

$$f_{\mathbf{w},b} : B_{x_i,\ldots x_l} \rightarrow \{\pm 1\}$$

$$f_{\mathbf{w},b}(\mathbf{x}) = \text{sgn}((\mathbf{w} \bullet \mathbf{x}) + b)$$

La posibilidad de introducir una estructura en el conjunto de hiperplanos está basada en el resultado de Vapnik [Va95] que el conjunto de hiperplanos canónicos:

$$\{f_{\mathbf{w},b} = \text{sgn}((\mathbf{w} \bullet \mathbf{x}) - b) \quad \|\mathbf{w}\| \leq A\} \qquad [2.4]$$

Tiene una dimensión VC, h, la cual satisface la siguiente cota,

$$h \leq \min\{\lfloor R^2 A^2 \rfloor, n\} + 1 \qquad [2.5]$$

Agregando la restricción $\|\mathbf{w}\| \leq A$, se puede obtener una dimensión VC que es mucho más pequeña que n (n es la dimensión de X) y así nos permite trabajar en un espacio de dimensión más alta.

Geométricamente hablando, la distancia de un punto **x** al hiperplano definida por $(\mathbf{w}, b)$ es

$$d(\mathbf{x} : \mathbf{w}, b) = \frac{|\mathbf{w} \bullet \mathbf{x} + b|}{\|w\|}$$

La ecuación 2.3 nos dice que la distancia más cercana entre el hiperplano $(\mathbf{w}, b)$ y los puntos del conjunto es $1/\|\mathbf{w}\|$.

Como $\|\mathbf{w}\| \leq A$, la distancia más cercana entre el hiperplano y los puntos tiene que ser mayor a 1/A (ver ecuación 2.4). El conjunto de hiperplanos restringidos tendrá una distancia de al menos 1/A del conjunto de puntos. Esto es equivalente a poner una esfera de radio 1/A alrededor de cada punto y considerar solo aquellas hiperplanos que no interceptan ninguna de las esferas.

Si el conjunto de entrenamiento es linealmente separable, entonces el SVM el cual está basado en la minimización de riesgo estructural, tratará de encontrar entre las hiperplanos canónicos a aquel con la norma mínima ($\|\mathbf{w}\|^2$) debido a que una pequeña norma nos da una pequeña dimensión VC, h, (ver ecuación 2.5).

2.3 Kernels y sus propiedades

Como la separación lineal es una situación difícil de tener en la práctica, la teoría puede ser extendida a superficies de separación no lineales por medio de una función ϕ de los puntos originales a puntos de características. Para lograr esto hay que cambiar la representación de los datos. Es decir,

$$\mathbf{x} = (x_1, \ldots x_n) \rightarrow \phi(\mathbf{x}) = (\phi_1(\mathbf{x}), \ldots \phi_m(\mathbf{x})) \qquad [2.6]$$

Esto es equivalente a transformar el espacio de entrada X en un nuevo espacio $F = \{\phi(\mathbf{x}) / \mathbf{x} \in X\}$

Ejemplo:

Consideremos la siguiente función objetivo $f(m_1, m_2, r) = \frac{c m_1 m_2}{r^2}$. Esta es la segunda ley gravitacional de Newton entre dos cuerpos con masa m_1, m_2, y separación r. Claramente, notamos que éste no es un modelo lineal, pero haciendo un cambio de coordenadas podemos convertir esta función a un modelo lineal

$$(m_1, m_2, r) \rightarrow (x, y, z) = (\ln m_1, \ln m_2, \ln r)$$

$$g(x, y, z) = \ln f(m_1, m_2, r) = \ln(c) + \ln m_1 + \ln m_2 - 2 \ln r$$
$$= k + x + y - 2z$$

donde este último es claramente un modelo lineal.

Definición de Kernel

Una función k es llamada un kernel si, está definida por

$$k(\mathbf{x}, \mathbf{z}) = \langle \phi(\mathbf{x}), \phi(\mathbf{z}) \rangle$$

para todo x,z en X, donde < > representa producto interno y ϕ está definida en [2.6]

2.3.1 Ejemplos de kernel

1. Kernel identidad

$k(\mathbf{x},\mathbf{z}) = (\mathbf{x} \bullet \mathbf{z})$

2. Dada una matriz **A**, entonces lo siguiente define un kernel

$\mathbf{k(x,z)} = \mathbf{Ax} \bullet \mathbf{Az} = \mathbf{(Ax)'Az} = \mathbf{x}' A' A\mathbf{x} = \mathbf{x}' B\mathbf{x}$

Donde B es una matriz cuadrada semi-definida positiva y simétrica.

3. Kernel Cuadrático

$$\langle \mathbf{x}, \mathbf{z} \rangle^2 = (\sum_i x_i z_i)^2 = (\sum_i x_i z_i)(\sum_j x_j z_j)$$

$$= (\sum_i \sum_j x_j x_i z_i z_j) = \sum_{(i,j)=(1,1)}^{(m,m)} (x_i z_i)(x_j z_j)$$

Por lo tanto,

$$\phi(\mathbf{x}) = (x_i x_j)_{(i,j)=(1,1)}^{(m,m)}$$

Es decir, producto interno con elementos de la forma $(x_i x_j)$

En este caso las características son monomios de grado 2, notar que cuando $i \neq j$ la característica $(x_i x_j)$ ocurre dos veces.

Consideremos el siguiente caso particular, donde x y z son de dimensión dos:

$x = (x_1, x_2)$

$z = (z_1, z_2)$

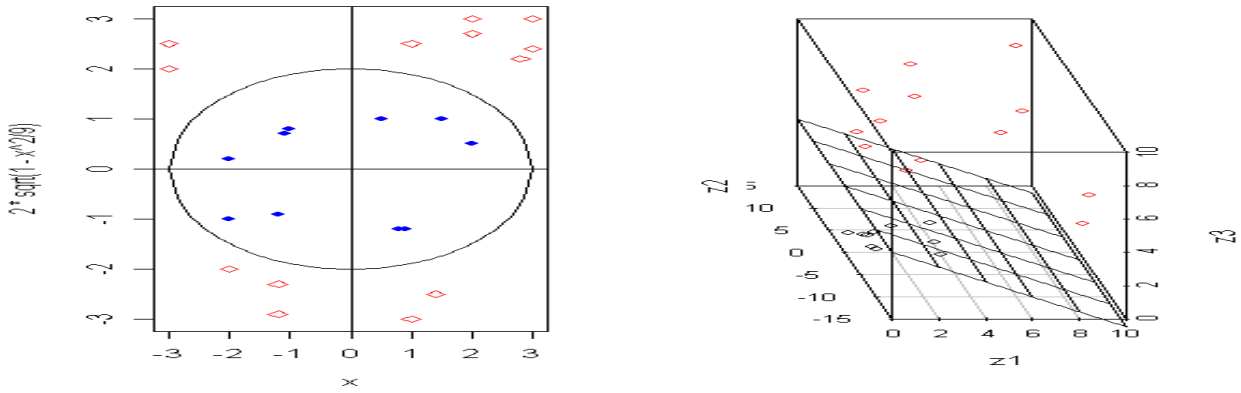
$\langle x, z \rangle^2 = (x_1 z_1 + x_2 z_2)^2 = {x_1}^2 {z_1}^2 + {x_2}^2 {z_2}^2 + 2x_1 z_1 x_2 z_2$

$= \langle ({x_1}^2, {x_2}^2, \sqrt{2} x_1 x_2), ({z_1}^2, {z_2}^2, \sqrt{2} z_1 z_2) \rangle$

$= \langle \phi(x), \phi(z) \rangle$

En la Figura 2.3 se puede observar el gráfico de la siguiente función:

$$\Phi : R^2 \rightarrow R^3$$
$$(x_1, x_2) \rightarrow (z_1, z_2, z_3) := (x_1^2, \sqrt{2}x_1x_2, x_2^2)$$



(a) **(b)**

Figura 2. 3 (a) Datos no linealmente separables, (b) Los datos son separables después de aplicar la función Φ.

4. Kernel Gaussiano o radial con anchura sigma

$$k(\mathbf{x},\mathbf{y}) = \exp(-\|\mathbf{x}-\mathbf{y}\|^2 / s^2)$$

5. Kernel Polinómico de grado c

$$k(\mathbf{x},\mathbf{y}) = (\mathbf{x}^t\mathbf{y} + b)^c$$

6. Kernel Sigmoidal con parámetros a y b

$$k(\mathbf{x},\mathbf{y}) = \tanh(a\mathbf{x}^t\mathbf{y} + b) \text{ donde } \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

2.3.2 Caracterización de los kernels

Consideremos un espacio de entrada $X = (\mathbf{x}_1, ...\mathbf{x}_n)$ y supongamos que $k(\mathbf{x},\mathbf{z})$ es una función simétrica de X. Consideremos la siguiente matriz $K = (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^n$ y que existe una matriz ortogonal V tal que $K = V\Lambda V'$ donde Λ es una matriz diagonal que contiene los valores propios λ_t de K, y las columnas de V son los correspondientes vectores propios $v_t = (v_{ti})_{i=1}^n$.

Asumamos que todos los valores propios son no negativos y consideremos

$$\phi : \mathbf{x}_i \to (\sqrt{\lambda_t} v_{ti})_{t=1}^{n} \in R^n \quad i = 1,...,n$$

Luego, $\left\langle \phi(\mathbf{x}_i) \bullet \phi(\mathbf{x}_j) \right\rangle = \sum_{t=1}^{n} \lambda_t v_{ti} v_{tj} = (V \Lambda V')_{ij} = K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$

De donde tenemos que $k(\mathbf{x},\mathbf{z})$ es una función kernel correspondiente a la función de característica ϕ.

Ahora supongamos que los valores propios λ_s son negativos con vector propio v_s y consideremos el vector columna *z* definido por

$z = \sum_{i=1}^{n} v_{si} \phi(x_i) = \sqrt{\Lambda} V' v_s$ en el espacio de características. Luego,

$$\|\mathbf{z}\|^2 = \mathbf{z} \bullet \mathbf{z} = v_s' V \sqrt{\Lambda} \sqrt{\Lambda} V' v_s = v_s' V \Lambda V' v_s$$
$$= v_s' K v_s < 0$$

Lo cual es una contradicción pues $\|\mathbf{z}\|^2$ siempre es positivo, por lo tanto, tenemos que la condición de que los valores propios son no negativos es necesaria. Luego, se obtiene la siguiente proposición.

Proposición 2.1

Sea X espacio de entrada finito con k(***x,z***) una función simétrica de ***x***, entonces k(***x,z***) es un kernel si y solo si la matriz $K = (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^{n}$ es semi-definida positiva (tiene valores propios no negativos

Proposición 2.2

Sean K_1 y K_2 kernels sobre $X \times X$, $\mathbf{x} \in R^n, a \in R^n$ y positivo, f una función de valor real sobre X, $\phi : X \to R^m$, K_3 un kernel sobre $R^m x R^m$ y **B** una matriz simétrica semi-definida positiva. Entonces los siguientes son también kernels:

1. $K(\mathbf{x},\mathbf{z}) = K_1(\mathbf{x},\mathbf{z}) + K_2(\mathbf{x},\mathbf{z})$
2. $K(\mathbf{x},\mathbf{z}) = aK_1(\mathbf{x},\mathbf{z})$
3. $K(\mathbf{x},\mathbf{z}) = K_1(\mathbf{x},\mathbf{z})K_2(\mathbf{x},\mathbf{z})$
4. $K(\mathbf{x},\mathbf{z}) = f(\mathbf{x})f(\mathbf{z})$
5. $K(\mathbf{x},\mathbf{z}) = K_3(\phi(\mathbf{x}),\phi(\mathbf{z}))$
6. $\mathrm{K}(\mathbf{x},\mathbf{z}) = \mathbf{x}' B\mathbf{z}$

Demostración: Fijemos un conjunto de puntos finitos $\{x_1 \ldots x_m\}$ y sea $\mathbf{K_1}$ y $\mathbf{K_2}$ las matrices restringidas a estos puntos. Consideremos un vector $\alpha \in R^m$.

1. $\alpha'(\mathbf{k_1} + \mathbf{k_2})\alpha = \alpha'\mathbf{k_1}\alpha + \alpha'\mathbf{k_2}\alpha \geq 0$

Luego $\mathbf{k_1} + \mathbf{k_2}$ es semi-definida positiva por lo tanto $K_1 + K_2$ es un kernel.

2. $\alpha' a\mathbf{k_1}\alpha = a\alpha'\mathbf{k_1}\alpha \geq 0$ por lo tanto tenemos aK_1 es un kernel

3. Definamos $\mathbf{K} = \mathbf{K_1} \otimes \mathbf{K_2}$ el producto tensorial de dos matrices semi-definidas positivas es también semi-definida positiva. Los valores propios del producto tensorial son el producto de todos los pares de productos de los valores propios de las dos componentes. Es decir,

$$(\mathbf{K_1} \otimes \mathbf{K_2})(\mathbf{x} \otimes \mathbf{z}) = K_1\mathbf{x} \otimes K_2\mathbf{z} = \boldsymbol{\lambda}\mathbf{x} \otimes \boldsymbol{\mu}\mathbf{z} = \boldsymbol{\lambda}\boldsymbol{\mu}(\mathbf{x} \otimes \mathbf{z})$$

La matriz correspondiente a K_1K_2 es conocida como el producto Schur H de $\mathbf{K_1}$ y $\mathbf{K_2}$, donde

H es una sub matriz de **K,** donde para algún $\alpha \in R^l$ existe un correspondiente $\alpha_1 \in R^{l^2}$ tal que $\boldsymbol{\alpha}'\mathbf{H}\boldsymbol{\alpha} = \boldsymbol{\alpha}_1\mathbf{K}\boldsymbol{\alpha}_1 \geq \mathbf{0}$

Donde **H** es semi-definida positiva por lo tanto se concluye que K_1K_2 es kernel.

4. $$\alpha' K\alpha = \sum_i \sum_j \alpha_i \alpha_j K(x_i, x_j) = \sum_i \sum_j \alpha_i \alpha_j f(x_i) f(x_j)$$
$$= \sum_i \alpha_i f(x_i) \sum_j \alpha_j f(x_j) = (\sum_i \alpha_i f(x_i))^2 > 0$$

Así tenemos que $f(x)f(z)$ es kernel.

5. Claramente, K_3 es un kernel ya que la matriz obtenida por restringir K_3 a los puntos $\varphi(x_1),...\varphi(x_l)$ es semi-definida positiva como se requiere.

6. Consideremos $\mathbf{B} = \mathbf{V}'\boldsymbol{\Lambda}\mathbf{V}$ donde **V** es matriz ortogonal, y Λ matriz diagonal que contiene los valores propios no negativos. $\sqrt{\Lambda}$ matriz diagonal con la raíz cuadrada de los vectores propios.

Sea $\mathbf{A} = \sqrt{\Lambda}\mathbf{V}$

$$K(\mathbf{x},\mathbf{z}) = \mathbf{x}'\mathbf{B}\mathbf{z} = \mathbf{x}'\mathbf{V}'\boldsymbol{\Lambda}\mathbf{V}\mathbf{z} = \mathbf{x}'V'\sqrt{\Lambda}\sqrt{\Lambda}\mathbf{z} = \mathbf{x}'\mathbf{A}'\mathbf{A}\mathbf{z} = \langle \mathbf{A}\mathbf{x}, \mathbf{A}\mathbf{z} \rangle$$

Luego se tiene que $x'\mathbf{B}z$ es kernel.

Corolario 2.1

Si p(x) es un polinomio con coeficientes positivos las siguientes funciones también son kernels:

1. $K(\mathbf{x},\mathbf{z}) = p(K_1(\mathbf{x},\mathbf{z}))$

2. $K(\mathbf{x},\mathbf{z}) = \exp(K_1(\mathbf{x},\mathbf{z}))$

3. $K(\mathbf{x},\mathbf{z}) = \exp(-\|x-z\|^2/\sigma^2)$

Demostración:

1. El resultado es inmediato usando las partes 1, 2, 3 y 4 de la proposición anterior.

2. Tenemos que:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots$$

Así tendríamos $e^{k_1(x,z)} = 1 + k_1(x,z) + \frac{k_1(x,z)^2}{2!} + \frac{k_1(x,z)^3}{3!} + \ldots$

Luego por las partes 1,2 ,3 y 4 de la proposición tenemos que $e^{k_1(x,z)}$ es kernel.

3. $$\exp(-\|x-z\|^2/\sigma^2) = \exp(-\|x\|^2/\sigma^2 - \|z\|^2/\sigma^2 - 2\langle x,z\rangle/\sigma^2)$$
$$= \exp(-\|x\|^2/\sigma^2)\exp(\|z\|^2/\sigma^2)\exp(-2\langle x,z\rangle/\sigma^2)$$

Por la parte 4 de la proposición y parte 2 del corolario concluimos que $\exp(-\|x-z\|^2/\sigma^2)$ es kernel.

CAPITULO III

SVM PARA CLASIFICACIÓN

SVM juega un papel importante en la minería de datos, con una gran cantidad de aplicaciones en el mundo real. Entre estas aplicaciones se incluyen categorización de texto, reconocimiento de caracteres editados, clasificación de imágenes, detección de pozos petrolíferos, detección de armamentos nucleares, predicciones de energía eléctrica, clasificación de tipos de proteína basada en DNA, aplicaciones de crédito entre otras. En el presente capítulo, describiremos primero, la aplicación de SVM para problemas de dos clases, tanto en el caso que haya separabilidad como cuando no hay separabilidad. En este capítulo también se discutirán los kernels y sus propiedades. También, se presentarán los métodos más usados para aplicar el SVM en el caso de tener varias clases, así como los métodos más usados para resolver el problema de programación cuadrática asociada al SVM.

3.1 SVM para dos clases linealmente separables

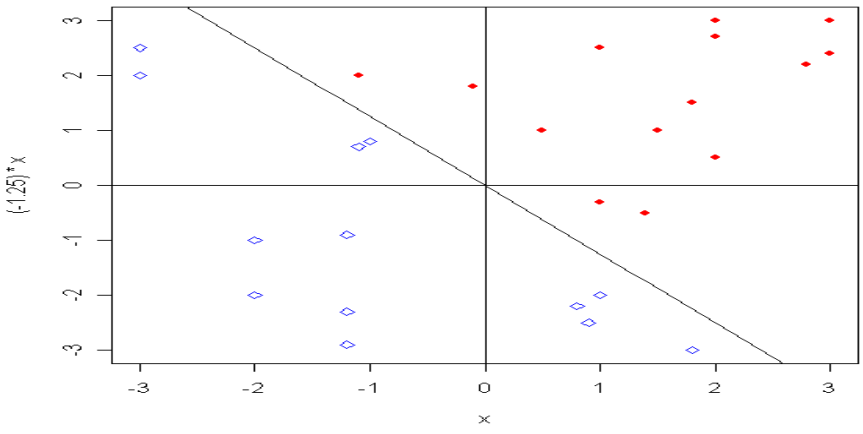


Figura 3. 1 Ejemplo de dos clases linealmente separables por un hiperplano.

La clasificación binaria con clases, digamos 1 y -1 frecuentemente es ejecutada usando funciones f de valor real tal que $f : R^n \to R$

$(x_1.....x_n) \to \{1,-1\}$

Donde f asigna el objeto $\mathbf{x} = (x_1,...,x_n)$ a la clase positiva si $f(\mathbf{x}) \geq 0$ en otro caso a la clase negativa.

Consideremos el caso donde $f(\mathbf{x}) = \langle \mathbf{w} \bullet \mathbf{x} \rangle + b = \sum_{i=1}^{n} w_i x_i + b$ es una función lineal.

La regla de decisión está dada por:

$$\operatorname{sgn} f(\mathbf{x}) = \begin{cases} 1 & \text{si } f(\mathbf{x}) > 0 \\ 0 & \text{si } f(\mathbf{x}) = 0 \\ -1 & \text{si } f(\mathbf{x}) < 0 \end{cases}$$

pero se usará la convención de sgn(0) =1

3.1.1 Interpretación geométrica del SVM para el caso separable

Un hiperplano en un espacio unidimensional, es un punto que divide una línea en dos segmentos. En un espacio bidimensional, un hiperplano es una recta que divide el plano en dos partes. En un espacio tridimensional, un hiperplano es un plano usual que divide el espacio en dos partes. Este concepto también puede ser aplicado a espacios de cuatro o más dimensiones donde estos objetos divisores se llaman simplemente hiperplanos.

Un hiperplano se puede definir en términos del producto escalar $\langle \mathbf{w} \bullet \mathbf{x} \rangle + b = 0$

El vector ***w*** es ortogonal al hiperplano, y $\frac{|b|}{\|\mathbf{w}\|}$ es la distancia del origen al hiperplano.

Lo que se desea es separar las dos clases mediante un hiperplano como se muestra en la Figura 3.2:

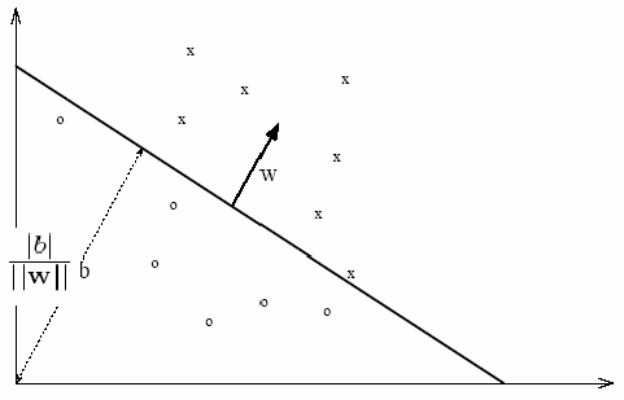


Figura 3. 2 Un hiperplano y su distancia al origen.

Existe un único hiperplano óptimo que es el que proporciona un mayor margen de separación entre las clases.

El objetivo es establecer un hiperplano que divida el conjunto de datos dejando todos los puntos de la misma clase a un lado, al mismo tiempo que maximiza la distancia mínima entre cualquier observación de las dos clases al hiperplano.

Definición 3.1: Conjunto linealmente separable

El conjunto S con *m* instancias es **linealmente separable** con margen 1 si existen $\mathbf{w} \in R^n$ y $b \in R$ tales que:

$$\{\mathbf{w} \bullet \mathbf{x}_i + b \geq 1, \text{ si } y_i = 1 \quad \text{y} \quad \mathbf{w} \bullet \mathbf{x}_i + b \leq -1, \text{ si } y_i = -1\} \text{ Para todo } \mathbf{x}_i \in S \quad i = 1,...,m$$

En notación compacta, se tiene: $y_i(\mathbf{w} \bullet \mathbf{x}_i + b) \geq 1$

Además, la distancia del punto $\mathbf{x}_i$ al hiperplano de separación está dada por: $d_i = \dfrac{|\mathbf{w} \bullet \mathbf{x}_i + b|}{\|\mathbf{w}\|}$

Por lo tanto $y_i d_i \geq \frac{1}{\|\mathbf{w}\|}$

Así, $\frac{1}{\|\mathbf{w}\|}$ es una cota inferior de $y_i \frac{|\mathbf{w} \bullet \mathbf{x}_i + b|}{\|\mathbf{w}\|}$.

Definición 3.2: Hiperplano de separación óptima

Dado un conjunto linealmente separable el **hiperplano de separación óptima**, es el hiperplano de separación que maximiza la distancia al punto más cercano de cada clase, la cual está dada por $\frac{1}{\|\mathbf{w}\|}$.

3.1.2 El SVM formulado como un problema de optimización

Lo que se quiere maximizar es $\frac{2}{\|\mathbf{w}\|}$ esto es máximo cuando $\|\mathbf{w}\|$ es mínimo y por monotocidad esto es equivalente a minimizar $\frac{\|\mathbf{w}\|^2}{2}$

Por consiguiente, el problema es:

Minimizar $\frac{1}{2}\mathbf{w}' \bullet \mathbf{w}$

sujeto a: $y_i(\mathbf{w} \bullet \mathbf{x}_i + b) \geq 1 \quad \text{i} = 1,...,\text{m}$

La solución de este problema de optimización convexa se obtiene usando multiplicadores de Lagrange.

En general, una condición necesaria para que x_0 sea solución del problema:

$\min f(x)$

sujeto a $g_i(x) \leq 0$

Se obtiene resolviendo

$$\frac{\partial}{\partial x}(f(x)+\sum \alpha_i g_i(x))_{x=x_0}=0$$

Donde, los α_i son llamados multiplicadores de Lagrange, y los g_i son las restricciones del problema.

En nuestro caso tenemos que:

Minimizar $\frac{1}{2}\|\mathbf{w}\|^2$ [3.1]

sujeto a: $1-y_i(\mathbf{w}\bullet\mathbf{x}_i+b)\le 0$ [3.2]

Sea $l=\frac{1}{2}\mathbf{w}^{\mathbf{t}}\mathbf{w}+\sum_{i=1}^{m}\alpha_i(1-y_i(\mathbf{w}\bullet\mathbf{x}_i+b))$ [3.3]

Derivando la función Lagrangiana [3.3] con respecto a ***w*** e igualando a cero se obtiene:

$$\mathbf{w}+\sum_{i=1}^{m}\alpha_i(-y_i)\mathbf{x}_i=0$$

Luego, $\mathbf{w}=\sum_{i=1}^{m}\alpha_i\mathbf{x}_i y_i$ [3.4]

Derivando la función Lagrangiana [3.3] con respecto a *b*, e igualando a cero se tiene que

$$\sum_{i=1}^{m}\alpha_i y_i=0 \quad [3.5]$$

Sustituyendo las ecuaciones [3.4] y [3.5] en la función Lagrangiana, el problema es equivalente a:

Maximizar

$$l(\alpha)=\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i\bullet\mathbf{x}_j)+\sum_{i=1}^{m}\alpha_i-\sum_{i=1}^{m}\alpha_i y_i(\mathbf{w}\bullet\mathbf{x}_i)+b\sum_{i=1}^{m}\alpha_i y_i$$

$$l(\alpha)=\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i\bullet\mathbf{x}_j)+\sum_{i=1}^{m}\alpha_i-\sum_{i=1}^{m}\sum_{i=1}^{m}\alpha_j\alpha_i y_j y_i(\mathbf{x}_i\bullet\mathbf{x}_j)$$

$$l(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j (\mathbf{x}_i \bullet \mathbf{x}_j)$$

Maximizar $\quad l(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \mathbf{w}' \mathbf{w}$ [3.6]

Sujeto a las restricciones

$$\begin{cases} \alpha_i \geq 0, \quad i = 1, \text{K}\ , m \\ \sum_{i=1}^{m} \alpha_i y_i = 0 \end{cases}$$ [3.7]

La primera de las restricciones de [3.7] es debido a la sexta condición del teorema de Kuhn-Tucker. Los vectores $\mathbf{x_i}$ cuyos coeficientes α_i son positivos en la ecuación $\mathbf{w} = \sum_{i=1}^{m} \alpha_i \mathbf{x}_i y_i$ son llamados vectores de soporte (SV) del hiperplano que separa a la clase C_1 y a la clase C_2, de donde se deriva el nombre de Maquinas de Soporte Vectorial para el clasificador.

Para hallar b, sabemos que:

$(\mathbf{w} \bullet \mathbf{x}_i + b) \geq 1$ para todo i, tal que $y_i = 1$. Luego,

$$\begin{aligned} &\mathbf{w} \bullet \mathbf{x}_i \geq 1 - b \\ &\min_{y_i = 1} (\mathbf{w} \bullet \mathbf{x}_i) \geq 1 - b \\ &b \geq 1 - \min_{y_i = 1} (\mathbf{w} \bullet \mathbf{x}_i) \end{aligned}$$ [3.8]

También tenemos que

$(\mathbf{w} \bullet \mathbf{x}_i + b) \leq -1$ para todo i, tal que $y_i = -1$, por lo tanto

$$
\begin{aligned}
1+b &\le -\mathbf{w}\bullet\mathbf{x}_i \\
1+b &\le \min_{y_i=-1}(-\mathbf{w}\bullet\mathbf{x}_i) \\
1+b &\le -\max_{y_i=-1}(\mathbf{w}\bullet\mathbf{x}_i) \\
-b &\ge 1+\max_{y_i=-1}(\mathbf{w}\bullet\mathbf{x}_i)
\end{aligned}
\qquad [3.9]
$$

Restando las últimas desigualdades obtenidas en [3.8] y [3.9] se tiene que

$$
\begin{aligned}
2b &\ge -\min_{y_i=1}(\mathbf{w}\bullet\mathbf{x}_i) - \max_{y_i=-1}(\mathbf{w}\bullet\mathbf{x}_i) \\
b &\ge -\frac{\min_{y_i=1}(\mathbf{w}\bullet\mathbf{x}_i) + \max_{y_i=-1}(\mathbf{w}\bullet\mathbf{x}_i)}{2}
\end{aligned}
\qquad [3.10]
$$

Observemos que el problema de optimización [3.6] sujeto a las restricciones [3.7], cumple con las condiciones del teorema de Kuhn-Tucker (ver apéndice A.1), donde la condición complementaria Karush-Kuhn-Tucker provee información acerca de la estructura de la solución. Por la condición 5 del teorema KKT, la solución óptima satisface:

$$
\alpha_i[y_i(\langle \mathbf{w}\bullet\mathbf{x}_i\rangle + b) - 1] = 0 \quad \text{para } i = 1,...,m \qquad [3.11]
$$

Por lo tanto, el hiperplano óptimo se puede expresar de la siguiente forma:

$$
f(\mathbf{x},\alpha,b) = \sum_{i=1}^{m} y_i\alpha_i\langle \mathbf{x}_i \bullet \mathbf{x}\rangle + b = \sum_{i\in SV} y_i\alpha_i\langle \mathbf{x}_i \bullet \mathbf{x}\rangle + b
$$

Gráficamente, los vectores de soporte son los puntos que tocan el límite del margen, ver Figura 3.3.

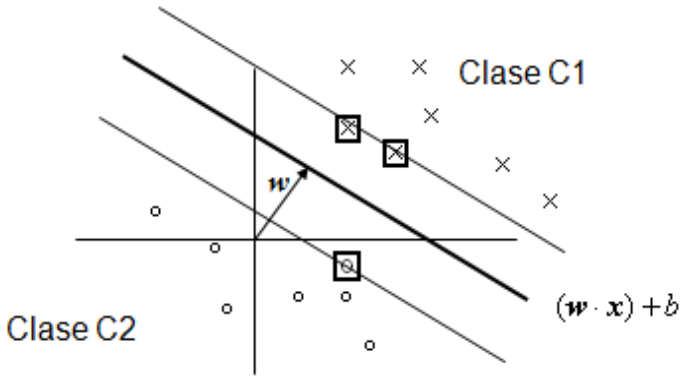


Figura 3. 3 Vectores de soporte para un problema de clasificación de 2 clases.

Otra importante consecuencia de la condición complementaria Karush-Kuhn-Tucker es la siguiente:

De la ecuación [3.11] se tiene que $y_i\langle \mathbf{w} \bullet \mathbf{x}_i \rangle + y_i b = 1$. Además, sabemos que $\mathbf{w} = \sum_j y_j \alpha_j \mathbf{x}_j$

por lo tanto $\langle \mathbf{w} \bullet \mathbf{x}_i \rangle = \sum_j y_j \alpha_j \langle \mathbf{x}_j \bullet \mathbf{x}_i \rangle$

De lo anterior tenemos que,

$$y_i \left\{ \left[\sum_j y_j \alpha_j \langle \mathbf{x}_j \bullet \mathbf{x}_i \rangle \right] + b \right\} = 1 \qquad [3.12]$$

Es decir, $y_i f(\mathbf{x}_i, \alpha, b) = 1$ para todo $i \in SV$

Además, de esto tenemos que

$$\langle \mathbf{w} \bullet \mathbf{w} \rangle = \sum_{i,j=1}^{m} y_i y_j \alpha_i \alpha_j \langle \mathbf{x}_i \bullet \mathbf{x}_j \rangle$$

$= \sum_{j \in SV} \alpha_j y_j \sum_{i \in SV} \alpha_i y_i \langle \mathbf{x}_i \bullet \mathbf{x}_j \rangle$ Usando la anterior afirmación obtenemos que

$$= \sum_{j \in SV} \alpha_j (1 - y_j b)$$
$$= \sum_{j \in SV} \alpha_j - \sum_{j \in SV} \alpha_j y_j b$$
$$= \sum_{j \in SV} \alpha_j - b \sum_{j \in SV} \alpha_j y_j$$
$$= \sum_{j \in SV} \alpha_j$$

De esto obtenemos la siguiente proposición:

Proposición 3.1: Consideremos una muestra de entrenamiento linealmente separable $S = ((\mathbf{x}_1, \mathbf{y}_1),...,(\mathbf{x}_m, \mathbf{y}_m))$ y supongamos que α^* y b^* soluciona el problema de optimización [3.6], con las restricciones [3.7]. Entonces, el vector $\mathbf{w} = \sum_{i=1}^{m} y_i \alpha^*_i \mathbf{x_i}$ define el hiperplano con máximo margen, el cual está dado por:

$$\gamma = 1/\|\mathbf{w}\|_2 = (\sum_{i \in SV} \alpha^*_i)^{-1/2}$$

3.2 SVM para clases no linealmente separables

Para tratar con datos que no son linealmente separables, ver Figura 3.4, se puede generalizar el análisis previo introduciendo algunas variables no-negativas ξ_i, para conjuntos no linealmente separable se cumple:

$$y_i(\mathbf{w} \bullet \mathbf{x}_i + b) \geq 1 - \xi_i \quad i = 1,...m$$

Los puntos x_i para los cuales $\xi_i \neq 0$, no satisfacen la definición de linealmente separable. Entonces, el término $\sum_{i=1}^{m} \xi_i$ puede ser tomado como un tipo de error en la clasificación. Por lo tanto, el problema del hiperplano óptimo es redefinido como la solución al problema

$$\min\left\{\frac{1}{2}\mathbf{w}\bullet\mathbf{w}+C\sum_{i=1}^{m}\xi_i\right\} \qquad [3.13]$$

$$y_i(\mathbf{w}\bullet\mathbf{x}_i+b)\geq 1-\xi_i \qquad i=1,...,m \qquad [3.14]$$

$$\xi_i\geq 0 \qquad i=1,...,m \qquad [3.15]$$

Donde *C* es una constante, que puede ser definido como un parámetro de regularización. Este es el único parámetro que necesita ser ajustado en la formulación del SVM. El ajuste de éste parámetro puede hacer un balance entre la maximización del margen y el error de la clasificación.

Este nuevo problema de optimización puede ser resuelto construyendo el Lagrangiano y transformándolo en su forma dual. Por el apéndice [A.1] tenemos que el Lagrangiano está dado por:

$$L(\mathbf{w},b,\xi,\boldsymbol{\alpha},\mathbf{r})=\frac{1}{2}\langle\mathbf{w}\bullet\mathbf{w}\rangle+C\sum_{i=1}^{m}\xi_i-\sum_{i=1}^{m}\alpha_i[y_i(\langle\mathbf{x}_i\bullet\mathbf{w}\rangle+b)-1+\xi_i]-\sum_{i=1}^{m}r_i\xi_i$$

[3.16]

donde $\xi_i\geq 0$ y $r_i\geq 0$

Derivando con respecto a $\mathbf{w},\xi$ y b e igualando a cero tenemos

$$\frac{\partial}{\partial\mathbf{w}}L=\mathbf{w}-\sum_{i=1}^{m}\alpha_i y_i x_i=0 \qquad [3.17]$$

$$\frac{\partial}{\partial\xi_i}L=C-\alpha_i-r_i=0 \qquad [3.18]$$

$$\frac{\partial}{\partial b}L=\sum_{i=1}^{m}\alpha_i y_i=0 \qquad [3.19]$$

Reescribiendo [3.16] tenemos

$$L(\mathbf{w},b,\xi,\boldsymbol{\alpha},\mathbf{r}) = \frac{1}{2}\langle \mathbf{w} \bullet \mathbf{w} \rangle + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\alpha_i y_i(\langle \mathbf{x}_i \bullet \mathbf{w} \rangle) - b\sum_{i=1}^{m}\alpha_i y_i + \sum_{i=1}^{m}\alpha_i - \sum_{i=1}^{m}\alpha_i\xi_i - \sum_{i=1}^{m} r_i\xi_i$$

$$L(\mathbf{w},b,\xi,\boldsymbol{\alpha},\mathbf{r}) = \frac{1}{2}\langle \mathbf{w} \bullet \mathbf{w} \rangle - \sum_{i=1}^{m}\alpha_i y_i(\langle \mathbf{x}_i \bullet \mathbf{w} \rangle) - b\sum_{i=1}^{m}\alpha_i y_i + \sum_{i=1}^{m}\alpha_i + \sum_{i=1}^{m}\xi_i(C - \alpha_i - r_i)$$

[3.20]

Sustituyendo [3.17], [3.18] y [3.19] en [3.20]

$$L(\boldsymbol{\alpha}) = \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j \langle x_i \bullet x_j \rangle - \sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j \langle x_i \bullet x_j \rangle + \sum_{i=1}^{m}\alpha_i$$

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j \langle x_i \bullet x_j \rangle$$

La restricción $C - \alpha_i - r_i = 0$, junto con la relación $r_i \geq 0$, obliga a que $\alpha_i \leq C$, mientras que, $\xi_i \neq 0$, solo si $r_i = 0$ y por lo tanto $\alpha_i = C$.

Por lo tanto el problema dual es:

Maximizar $L(\boldsymbol{\alpha}) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j y_i y_j \langle x_i \bullet x_j \rangle$ [3.21]

sujeto a: $0 \leq \alpha_i \leq C \qquad \forall$i [3.22]

$$\sum_{i=1}^{m}\alpha_i y_i = 0 \qquad [3.23]$$

El teorema de KKT [A.2] juega un papel importante en la teoría de SVM. De acuerdo a este teorema la solución del problema dual anterior satisface:

$$\begin{aligned} &\alpha_i[y_i(\langle \mathbf{x}_i \bullet \mathbf{w} \rangle + b) - 1 + \xi_i] = 0 \qquad \forall\text{i} \\ &\xi_i r_i = \xi_i(\alpha_i - C) = 0 \end{aligned} \qquad [3.24]$$

De donde las variables ξ_i son diferentes de cero solo cuando $\alpha_i = C$

El punto x_i correspondiente a $\alpha_i > 0$ es llamado vector de soporte.

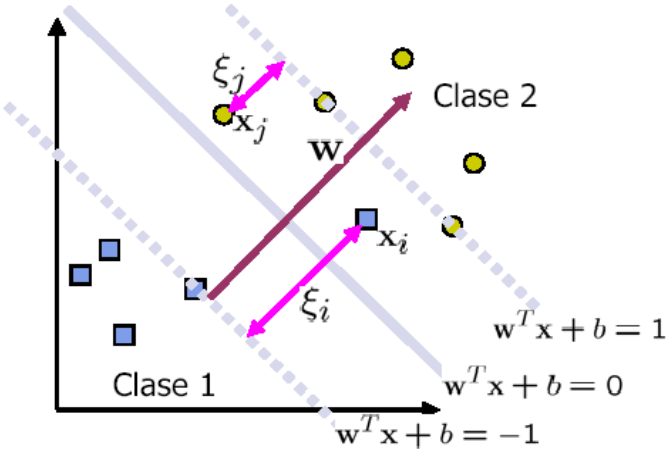


Figura 3. 4 El parámetro del error ξ_i cuando las clases no son linealmente separables. Tomada de [Bet05].

3.3 SVM no lineal

El proceso de clasificación se hará considerando el siguiente problema de optimización:

$$\text{Minimizar} \frac{\| w \|^2}{2} \qquad [3.25]$$

$$\text{sujeto a:} \quad y_i(\mathbf{w} \bullet \phi(\mathbf{x}_i) + b) \geq 1 \qquad [3.26]$$

Siguiendo los mismos procedimientos de las dos secciones anteriores se llega al siguiente problema dual:

$$\text{Maximizar } L(\boldsymbol{\alpha}) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \left\langle \phi(x_i) \bullet \phi(x_j) \right\rangle \qquad [3.27]$$

Los parámetros **w** y b pueden ser derivados usando las siguientes ecuaciones:

$$\mathbf{w} = \sum_{i=1}^{m} \alpha_i y_i \phi(\mathbf{x}_i)$$

$$\alpha_i \{ y_i \left\{ \left[\sum_j y_j \alpha_j \left\langle \phi(\mathbf{x}_j) \bullet \phi(\mathbf{x}_i) \right\rangle \right] + b \right\} - 1 \} = 0$$

Luego, la regla de decisión para clasificar casos de prueba se puede evaluar usando el producto interno entre los casos de prueba y los casos de entrenamiento.

$$f(\mathbf{x}) = (\sum_{j=1}^{m} \alpha_j y_j \left\langle \phi(\mathbf{x}_j) \bullet \phi(\mathbf{x}) \right\rangle + b)$$

Este producto interno da paso a la importante definición de kernel, que se trató en el capítulo anterior. La función de decisión se convierte en:

$$f(\mathbf{x}) = (\sum_{j=1}^{m} \alpha_j y_j K(\mathbf{x_i}, \mathbf{x_j}) + b)$$

3.4 Generalización del clasificador SVM para varias clases

La forma estándar del SVM usualmente es usado para dos clases, sin embargo hay situaciones en las cuales es necesario extender esta técnica a varias clases. En esta sección, vamos a describir algunos métodos de cómo el SVM estándar puede ser extendido de un problema de clasificación binaria a un problema de clasificación con k-clases.

3.4.1 Método uno contra uno

Este método fue introducido por [KPD90], y usado por primera vez en SVM por [Fri96, Kre99]. Este método construye k(k-1)/2 clasificadores, donde cada clasificador es entrenado en datos que contienen dos clases diferentes de las k existentes.

Para entrenar el clasificador con datos de la i-ésima y j-ésima clase resolvemos el siguiente problema de clasificación binaria:

$$\min_{\mathbf{w}^{ij}, b^{ij}, \xi^{ij}} \frac{1}{2} (\mathbf{w}^{ij})' \mathbf{w}^{ij} + C (\sum_t (\xi^{ij})_t)$$

$$(\mathbf{w}^{ij})'(\mathbf{x}_t) + b^{ij} \geq 1 - \xi_t^{ij} \quad \text{si} \quad \mathbf{x}_t \quad esta \quad en \quad la \quad i-esima \ clase$$

sujeto a: $(\mathbf{w}^{ij})'(\mathbf{x}_t) + b^{ij} \leq -1 + \xi_t^{ij}$ si $\mathbf{x}_t$ *esta en la j − esima clase*

$$\xi_t^{ij} \geq 0$$

Para clasificar una muestra de prueba ***x*** se usa el método de votación. La muestra x es asignado a la clase con mayor número de votos. En el caso de que exista el mismo número de votos se asigna a una clase con índice menor. Este método es usado por la función **svm** en el paquete de **R** [DLMW06].

3.4.2 Método uno versus todos

Este método [BCDDGJLMSV94] construye k clasificadores, donde k es el número de clases. Para entrenar el clasificador i-ésimo se usa todas las muestras de entrenamiento de la clase i-ésima con etiquetas positivas y todas las otras muestras con etiquetas negativas

Dado m muestras de entrenamiento $(x_1, y_1),...,(x_m, y_m)$, donde $x_i \in R^n, i = 1,...,m \quad y_i \in \{1,...k\}$ es la clase de $\boldsymbol{x}_i$

Para entrenar el i-ésimo clasificador SVM se resuelve el siguiente problema

$$\min_{\mathbf{w}^i, b^i, \xi^i} \frac{1}{2}(\mathbf{w}^i)'\mathbf{w}^i + C\left(\sum_t (\xi^i)_t\right)$$

$$(\mathbf{w}^i)'(\mathbf{x}_t) + b^i \geq 1 - \xi_t^i \quad \text{si} \quad y_j = \text{i}$$

sujeto a: $(\mathbf{w}^i)'(\mathbf{x}_t) + b^i \leq -1 + \xi_t^i$ si $y_j \neq \text{i}$

$$\xi_t^i \geq 0$$

Después de resolver el problema anterior se tiene k funciones de decisión:

$$(\mathbf{w}^1)'(\mathbf{x}) + b^1$$

M

M

$$(\mathbf{w}^k)'(\mathbf{x}) + b^k$$

La observación ***x*** es asignada a la clase que tenga el valor de la función de decisión mayor es decir:

$$clase \ \text{de} \ \mathbf{x} = \arg\max_{i=1,\ldots,k}((\mathbf{w}^i)'(x) + b^i)$$

3.4.3 Método basado en gráficos acíclicos dirigidos (DAGSVM)

Es método fue propuesto por [PCS00]. La fase de entrenamiento es similar al de uno contra uno, en el cual se encuentran k(k-1)/2 clasificadores SVM binarios. En la fase de prueba este método tiene k(k-1)/2 nodos internos y k hojas. Cada nodo es un clasificador SVM binario de la i-ésima y j-ésima clase. Una muestra de prueba ***x*** es evaluada con el nodo de decisión que corresponde al primer y último elemento de una lista de clases. Si el nodo prefiere una de las dos clases la otra clase es eliminada de la lista y el algoritmo DAG, luego el algoritmo DAG prueba el primero y el último elemento de la nueva lista. Este algoritmo termina cuando solo hay una clase restante en la lista. Así para un problema de k clases, se evaluarán k-1 nodos de decisión para poder clasificar la observación ***x.***

La Figura 3.5 muestra el algoritmo DAG con 4 clases.

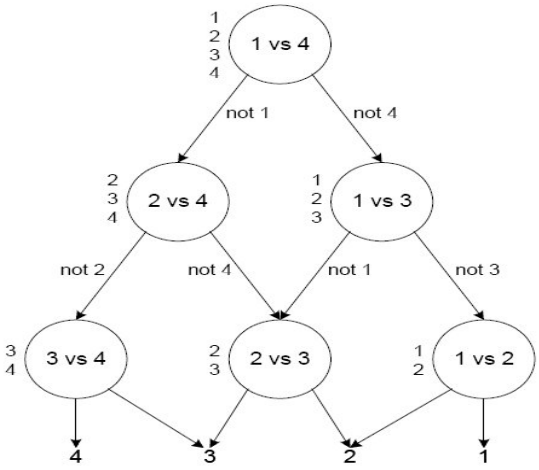


Figura 3.5 Gráfica que muestra cómo trabaja el algoritmo DAG para encontrar la mejor entre 4 clases. Tomada de [PCS00].

3.5 Técnicas para solucionar el problema de optimización cuadrática del SVM

3.5.1 Método "Chunking"

Cuando se tiene un conjunto de datos de tamaño grande, el problema de optimización cuadrática QP no puede ser resuelto fácilmente a través de técnicas estándar QP. La forma cuadrática en el problema QP del SVM implica almacenar una matriz de dimensión igual al número de muestras de entrenamiento. Esta matriz no puede ser almacenada en un computador con menos de un gigabyte si se tiene más de 20,000 muestras de entrenamiento.

Vapnik [CS04] describe un método para resolver el problema QP del SVM, que es conocido como fragmentación ("chunking"). El método de Fragmentación se basa en el hecho de que la eliminación de las muestras con multiplicadores de Lagrange, $\alpha_i = 0$, no cambia la solución. Por lo tanto, este método divide el problema QP en una serie de pequeños sub-problemas QP, cuyo objetivo es identificar las muestras de entrenamiento con α_i diferente de cero. Cada sub-problema QP actualiza el subconjunto de los α_i que están asociados con el sub-problema, dejando el resto de los α_i sin cambios. El sub-problema QP consiste de todos los α_i diferentes de cero del sub-problema anterior junto con M muestras que no cumplen con las condiciones KKT, para algún M. En el último paso queda determinado todo el conjunto de los α_i que no son ceros, por lo tanto, el último paso resuelve todo el problema QP.

La fragmentación reduce la dimensión de la matriz de la muestra de entrenamiento aproximadamente al número de α_i distintos de cero.

3.5.2 El método de descomposición de Osuna

Dado que el número de variables en el problema QP del SVM puede llegar a ser tan grande como igual al número de puntos de datos, entonces el problema de optimización QP se puede volver pesado computacionalmente, porque la forma cuadrática es completamente densa y los requisitos

de la memoria crecen con el cuadrado del número de puntos de datos. Osuna y otros [OFG97] proponen un algoritmo de descomposición que garantiza la optimalidad global, y puede ser utilizado para entrenar la SVM para conjuntos de datos muy grandes (por ejemplo 50,000 puntos de datos). La idea principal detrás de la descomposición es la solución iterativa de subproblemas y la evaluación de las condiciones de optimalidad que se utilizan tanto para generar valores iterativos mejorados, y también establecer los criterios para detener el algoritmo.
A diferencia del método "Chunking", Osuna sugiere mantener una matriz de tamaño constante para cada sub-problema QP, lo cual implica añadir y eliminar el mismo número de muestras en cada paso.

3.5.3 Algoritmo Optimización Mínima Secuencial (SMO)

La Optimización Secuencial Mínima (SMO) [PCS98] es un algoritmo simple que rápidamente puede resolver el problema cuadrático (QP) del SVM sin ningún tipo de almacenamiento extra para matrices y sin usar optimización numérica QP en todos los pasos. SMO descompone el problema QP en sub-problemas QP. A diferencia de otros métodos, SMO elige resolver el problema más pequeño posible de optimización en cada paso. Para el problema estándar QP del SVM, el problema más pequeño posible de optimización implica dos multiplicadores de Lagrange, debido a que éstos deben obedecer a una igualdad de restricción lineal. En cada paso, SMO elige dos multiplicadores de Lagrange para optimizar conjuntamente, luego encuentra los valores óptimos para estos multiplicadores, y se actualiza la SVM para reflejar los nuevos valores óptimos considerados. La ventaja del SMO radica en el hecho de que la solución de dos multiplicadores de Lagrange se puede hacer analíticamente. Por lo tanto, la optimización numérica QP se evita por completo. A pesar de que más sub-problemas de optimización se resuelven en el algoritmo, cada sub-problema es tan rápido que todo el problema QP se resuelva rápidamente. Debido a que no se utilizan algoritmos matriciales en el SMO, éste es menos susceptible a problemas de precisión numérica.

3.5.4 El algoritmo SMO modificado de Kerthi

Kerthi y otros [KSBM01] mejoraron la eficiencia del SMO. En particular, ellos señalan una importante fuente de ineficiencia causada por la forma en que el SMO mantiene y actualiza un único valor umbral. Usando criterios relacionados con las condiciones KKT para el dual, ellos sugieren el uso de dos parámetros umbrales y proponen dos versiones modificadas del SMO que son más eficientes que el original SMO. Comparaciones experimentales fueron llevadas a cabo en varios conjuntos de datos y se concluyó que los algoritmos modificados se desempeñaron más rápido que el original SMO en la mayoría de las situaciones.

CAPITULO IV

DETECCIÓN DE CASOS ANOMALOS

Los datos anómalos ("outliers") de la muestra de entrenamiento, son importantes detectarlos y posiblemente eliminarlos ya que pueden influir de manera significativa en el rendimiento de un clasificador. La presencia de "outliers" es un indicativo de que algunas observaciones o grupo de observaciones presentan un comportamiento que es muy diferente de lo normal.

La detección de "outliers" es un paso importante en la minería de datos, específicamente en pre procesamiento de datos. Los "outliers" tienen muchas aplicaciones, como por ejemplo en la detección de fraudes, intrusos cibernéticos y limpieza de datos. Aunque en muchas ocasiones se remueve los "outliers" antes de construir el modelo de aprendizaje, éstos pueden contener información útil. La eliminación de los "outliers" se hace para mejorar el modelo de aprendizaje que se intenta construir. Una práctica usual en minería de datos es ranquear las observaciones usando una medida del grado de anomalía de las mismas, en vez de clasificar las observaciones como "outlier" o no "outlier".

A continuación describiremos 3 métodos para detectar "outliers" multivariados.

4.1 Detección de "outliers" multivariados

Sea D un conjunto de datos con p variables y n observaciones. En el caso de clasificación supervisada cada observación pertenece a una clase. Usualmente, la matriz de datos contiene como última columna las clases a las cuales pertenecen las observaciones. En lo que sigue discutiremos tres métodos de detectar "outliers" en clasificación supervisada. Aquellas observaciones que sean anómalas en cada clase serán identificadas.

4.1.1 Detección de "outliers" basado en métodos estadísticos

Sea $\mathbf{x}$ una observación en el conjunto de datos y $\bar{\mathbf{x}}$ el centroide del conjunto de datos, que es un vector p dimensional cuyas componentes son la media de cada una de las variables. Además, sea

X la matriz original de los datos con las columnas centradas por sus medias. Entonces, sea $S = (\frac{1}{n-1})\mathbf{X'X}$ la matriz de covarianza de las p variables. Luego, la observación **x** es un outlier si

$$D^2(\mathbf{x}, \bar{\mathbf{x}}) = (\mathbf{x} - \bar{\mathbf{x}})'\mathbf{S}^{-1}(\mathbf{x} - \bar{\mathbf{x}}) > k$$

D^2 es llamada la distancia cuadrada de Mahalanobis de **x** al centroide del conjunto de datos. En particular si se asume que el conjunto de datos tiene una distribución normal multivariada, entonces cuando el número de observaciones es grande la distribución de la distancia de Mahalanobis se comporta como una Chi-cuadrado con p grados de libertad. Luego el punto de corte está dado por $k = \chi^2_{(p,1-\alpha)}$. Donde, χ^2 es una distribución Chi-cuadrado con p grados de libertad y con nivel de significancia α, usualmente se toma $\alpha = 0.05$. [RL87].

Los conjuntos de datos con "outliers" están sujetos a los efectos de "masking" y "swamping".

a) **Efecto de "masking":** Este nos dice que un "outlier" enmascara a un segundo outlier que está cerca. Es decir, este segundo es considerado un "outlier" por sí mismo, pero no cuando va acompañado del primer "outlier". Después de eliminar un "outlier", entonces la otra observación puede resultar como un "outlier".
b) **Efecto "swamping":** Se dice que un "outlier" sumerge a otro, si el último puede considerarse "outlier" solo bajo la presencia del primero. Es decir, después de eliminar el primer "outlier", el otro puede resultar ser una observación que no es anómala.

Estos dos efectos pueden resolverse usando estimadores robustos del centroide y de la matriz de covarianza, los cuales son menos afectados por los "outliers".

Rousseeuw [Rou85] introdujo dos estimadores robustos del centroide y la covarianza; el estimador de determinante de covarianza mínima (MCD: "Minimum covariance determinant") y el estimador de elipsoide de volumen mínimo (MVE: "mínimum volumen ellipsoid").

a) **Estimador de elipsoide de volumen mínimo *(MVE)***: Este estimador es la media y la covarianza de una sub-muestra de tamaño h ($h \leq n$) que minimiza el volumen de la matriz de covarianza asociada a esta sub-muestra. Es decir,

$$\mathbf{MVE} = (\overline{x_J}^*, S^*_J)$$

Donde $J = \left\{ \text{conjunto de h instancias} : vol(S_J^*) \le vol(S_K^*) \text{ para todo K tal que } \#(\text{K}) = \text{h} \right\}$ y h puede ser considerado como el número mínimo de observaciones que pueden ser "outliers", y es usualmente igual a $\left[\frac{n+p+1}{2} \right]$, donde [.] es la función parte entera, n es el número de observaciones y p el número de variables. El volumen de elipsoide es calculado por:

$vol(S_K) = \left\{ |S_K| mediana_{i=1,\ldots,h} d_i^{\,2} \right\}^{\frac{1}{2}}$, d_i representa la distancia de Mahalanobis

b) **Estimador de determinante de covarianza mínimo (MCD):** Este estimador es la media y la covarianza de una sub-muestra de tamaño h (h < n) que minimiza el determinante de la matriz de covarianza asociada a esta sub-muestra. Es decir,

$$\mathbf{MCD} = (\overline{x_J}^*, S^*_J)$$

Donde $J = \left\{ \text{conjunto de h instancias} : \left|S_J^*\right| \le \left|S_K^*\right| \text{ para todo K tal que } \#\text{K} = \text{h} \right\}$

h usualmente se toma por $\left[\frac{n+p+1}{2} \right]$.

Reemplazando en la distancia de Mahalanobis, los estimadores clásicos del centroide y la covarianza por el estimador MVE o el estimador MCD, pueden identificarse "outliers" para valores grandes de la distancia de Mahalanobis. En los experimentos de esta tesis se ha usado la librería **dpred** de **R,** la cual contiene la función **robout,** que permite hallar los "outliers".

4.1.2 Detección de "outliers" basado en distancia

Dada una medida de distancia en un espacio de características, tenemos dos definiciones de "outliers" basado en distancia.

1. Una observación **x** en un conjunto de datos D es un outlier con parámetros p y λ si al menos una fracción p de objetos están a una distancia mayor que λ de **x.** [KN97] [KN98] [KN00]
2. Dado los enteros k y n (k < n) los "outliers" son las n observaciones con las mayores distancias a sus k-esimos vecinos más cercano. [RRS00].

Bay y Schwabacher [BS03] proponen un algoritmo que recopila las definiciones 1 y 2. La idea principal de este algoritmo es que para cada observación en el conjunto D, uno va haciendo un seguimiento de los k vecinos más cercanos que se hayan encontrado. Cuando una observación con k vecinos mas cercanos alcanza un puntaje que es menor que un punto de corte entonces esta observación es removida de la lista de los posibles candidatos de "outliers", porque esta observación no puede ser un outlier. El algoritmo está descrito en [BS03].
Para los experimentos se utilizó la librería **dpred** en **R** que contiene la función **baysout,** con la cual se determinó los "outliers" basados en distancia.

4.1.3 Detección de "outliers" basado en densidad local

Este tipo de "outliers" fue introducido por Breunig y otros [BKNS00]. Para cada observación se calcula un factor de "outlier" local (LOF), basado en el grado de concentración de sus vecinos, el cual dará un indicativo de cuán fuerte se encuentra una observación de ser un "outlier". Se requieren varias definiciones para formalizar el algoritmo para encontrar "outliers" basados en densidad los cuales se mencionan a continuación

Definición 4.1: k-distancia de una observación x

Para cualquier entero positivo k, la k distancia de una observación ***x***, denotada por k-distancia(x), es definida como la distancia d(x,y) entre la observación x y la $y \in$ D tal que:

i) para al menos k observaciones $y' \in$ D-{x}, d(x,y') $\leq$ d(x,y)
ii) para a lo más k-1 observaciones $y' \in$ D-{x}, d(x,y') < d(x,y)

Así mismo, la vecindad de x basada en la k - distancia está dada por:

$$N_{K-dis\tan cia(\mathbf{x})} = \{y \in D - \{x\} \text{ tal que d(x , y)} \leq \text{k - distancia (x)}\} \quad [4.1]$$

La distancia de alcanzabilidad de x con respecto a y se define por

$$reach - dist_k(x, y) = \max\{k - dis\tan cia(y), d(x, y))\} \quad [4.2]$$

El algoritmo para detectar "outliers" basado en densidad local requiere solo un parámetro, *MinPts,* el cual es el número de vecinos más cercanos usados en definir la vecindad local de una observación.

Definición 4. 2: densidad de alcanzabilidad local de una observación x

Dada una observación x del conjunto de datos D la densidad de alcanzabilidad local está definida por:

$$lrd_{MinPts}(x) = \left\{ \frac{\sum_{y \in N_{Minpts}(x)} reach - dist_{MinPts}(x, y)}{|N_{MinPts}(x)|} \right\}^{-1} \quad [4.3]$$

Definición 4.3: Factor de "outlier" local (LOF) de una instancia x

El LOF mide el grado el cual una observación x puede considerarse outlier y está definido por:

$$LOF_{MinPts}(x) = \left\{ \frac{\sum_{y \in N_{MinPts}(x)} \frac{lrd_{MinPts}(y)}{lrd_{MinPts}(x)}}{|N_{MinPts}(x)|} \right\}$$

En los experimentos se usó la librería **dpred** en **R** que contiene la función **maxlof** con la cual hallaremos los "outliers" basados en densidad local.

4.2 Detección de "outliers" usando clasificador SVM con una sola clase

Para describir el dominio del conjunto de datos, se puede considerar que todos los elementos del mismo quedan encerrados en una hiperesfera con volumen mínimo. Minimizando el volumen del espacio de características capturado, se espera minimizar la posibilidad de aceptar objetos "outliers" dentro de la hiperesfera. La detección de "outliers" se puede considerar como clasificar una observación que cae dentro de la hipersesfera. Este método también llamado "*Support vector data description*" (SVDD), fue introducido por Tax [TD01]. Supongamos que tenemos un conjunto de entrenamiento N, cuyos objetos son x_i, $i = 1,...,m$ y la hiperesfera es descrita por un centro **a** y radio R. Una representación gráfica se muestra en la Figura 4.2

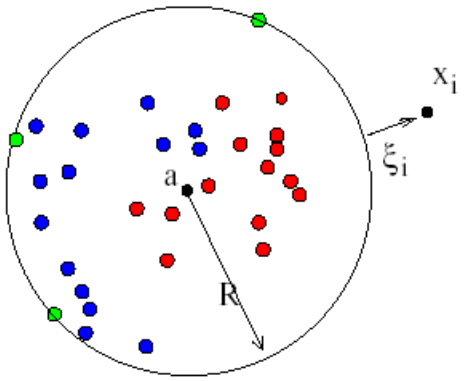


Figura 4. 1 La gráfica muestra una esfera con algunas muestras de entrenamiento. Un objeto es rechazado por la descripción. Figura tomada de [TD01].

Para permitir la posibilidad de "outliers" en el conjunto de entrenamiento, la distancia de $\boldsymbol{x_i}$ al centro **a** no debería ser estrictamente más pequeño que R^2. De esto ocurrir entonces se aplicará una penalidad a esta distancia. Luego, se introduce la variable de holgura ξ_i, que mide la distancia a la frontera, si el i-ésimo objeto está fuera de la descripción. Un parámetro extra C es introducido para crear un balance entre el volumen de la hiperesfera y el número de outliers que se detectan.

Para minimizar la función de error L que envuelve el radio de la hiperesfera y la distancia de la frontera a los objetos "outliers", se restringe la solución, usando el requerimiento que todos los datos están dentro de la hiperesfera:

El problema se formula de la siguiente forma:

Minimizar $L(R,a,\xi) = R^2 + C\sum_i \xi_i$ [4.15]

Sujeto a $\|x_i - a\|^2 \le R^2 + \xi_i$, $\xi_i \ge 0$ [4.16]

Las restricciones de la última ecuación pueden ser incorporadas en el error aplicando multiplicadores de Lagrange [A3].

$$L(R,a,\alpha,\xi) = R^2 + C\sum_i \xi_i - \sum_i \alpha_i \{R^2 + \xi_i - (x_i^2 - 2a \cdot x_i + a^2)\} - \sum_i \gamma_i \xi_i \quad [4.17]$$

Con multiplicadores de Lagrange $\alpha_i \ge 0$ y $\gamma_i \ge 0$. Esta función tiene que ser minimizada con respecto a *R, a, y* ξ_i y maximizar con respecto a α_i y γ_i. Calculando las derivadas parciales de L con respecto a *R* y *a* ξ_i igualando a cero tenemos.

$$\frac{\partial}{\partial R} L = 2R - \sum_i 2R\alpha_i = 0 \rightarrow \sum_i \alpha_i = 1 \quad [4.18]$$

$$\frac{\partial}{\partial a} L = \sum_i 2\alpha_i x_i - 2a = 0 \rightarrow \sum_i \alpha_i x_i = a \quad [4.19]$$

$$\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \gamma_i = 0, \quad \forall \text{i} \quad [4.20]$$

De la última ecuación $\alpha_i = C - \gamma_i$ $\forall$i y debido a $\alpha_i \ge 0$ y $\gamma_i \ge 0$, entonces los multiplicadores de Lagrange γ_i son ceros para cualquier muestra de entrenamiento que es erróneamente clasificada. Sustituyendo estos valores en el lagrangiano [4.17] tenemos:

$$L(R,a,\alpha,\xi) = R^2 + C\sum_i \xi_i - \sum_i \alpha_i R^2 - \sum_i \alpha_i \xi_i + \sum_i \alpha_i x_i^2 - \sum_i \alpha_i (2a \cdot x_i) + \sum_i \alpha_i a^2 - \sum_i \gamma_i \xi_i$$

$$L(R,a,\alpha,\xi) = R^2 + C\sum_i \xi_i - \sum_i \alpha_i R^2 + \sum_i \alpha_i x_i^2 - \sum_i \alpha_i (2a \cdot x_i) + a^2 \sum_i \alpha_i - \sum_i \xi_i (\gamma_i + \alpha_i)$$

Sustituyendo [4.18], [4.19] y [4.20] en la anterior se tiene:

$$L(R,a,\alpha,\xi) = R^2 + C\sum_i \xi_i - R^2 + \sum_i \alpha_i x_i^2 - 2\sum_{i,j} \alpha_i \alpha_j (x_i \cdot x_j) + \sum_{i,j} \alpha_i \alpha_j (x_i \cdot x_j) - C\sum_i \xi_i$$

$$L(R,a,\alpha,\xi) = \sum_i \alpha_i x_i^2 - \sum_{i,j} \alpha_i \alpha_j (x_i \cdot x_j)$$

Por lo tanto el nuevo problema es el siguiente:

Maximizar $L = \sum_{I} \alpha_i (x_i \cdot x_i) - \sum_{i,j} \alpha_i \alpha_j (x_i \cdot x_j)$ [4.21]

Sujeto a: $0 \le \alpha_i \le C$ [4.22]

$$\sum_i \alpha_i = 1$$

Este nos da un problema de optimización cuadrática. El error *L* tiene que ser maximizado con respecto a α_i. En la práctica una fracción grande de α_i tiende a cero y para una pequeña fracción $\alpha_i > 0$ y los correspondientes objetos son llamados vectores de soporte. Estos objetos aparecen en la frontera. El centro de la hiperesfera depende solamente de algunos vectores soporte. Los objetos con $\alpha_i = 0$ pueden omitirse de la descripción del conjunto de datos.

Un objeto **z** es aceptado por la descripción cuando

$$\|z - a\|^2 = (z \cdot z) - 2\sum_i \alpha_i (z \cdot x_i) + \sum_{i,j} \alpha_i \alpha_j (x_i x_j) \le R^2 \quad [4.23]$$

El radio R puede determinarse calculando la distancia del vector soporte x_i en la frontera al centro ***a***.

El modelo de la hiperesfera no siempre se puede cumplir. Usando el método de Vapnik, se puede reemplazar los productos internos por la función kernel K, el cual da más flexibilidad al método cuando se reemplaza el producto interno por el kernel gaussiano.

$$(\mathbf{x} \bullet \mathbf{y}) \rightarrow K(\mathbf{x},\mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2 / s^2)$$

Donde *s* representa la desviación estándar, se obtiene:

$$L = \sum_i \alpha_i \exp(-\|x_i - x_i\|^2 / s^2) - \sum_{i,j} \alpha_i \alpha_j \exp(-\|x_i - x_j\|^2 / s^2)$$

$$L = \sum_i \alpha_i \exp(0) - \sum_{i=j} \alpha_i \alpha_i \exp(0) - \sum_{i \neq j} \alpha_i \alpha_j \exp(-\|x_i - x_j\|^2 / s^2)$$

$$L = 1 - \sum_{i=j} \alpha_i{}^2 - \sum_{i \neq j} \alpha_i \alpha_j \exp(-\|x_i - x_j\|^2 / s^2)$$

Luego [4.21] cambia a:

$$L = 1 - \sum_i \alpha_i{}^2 - \sum_{i \neq j} \alpha_i \alpha_j K(x_i, x_j)$$

[4.24]

La maximización de la ecuación [4.24] da como solución los multiplicadores de Lagrange $\boldsymbol{\alpha}$, los cuales son usados en el cómputo del centro **a**. Para que un nuevo objeto **z** caiga dentro de la hiperesfera se tiene que cumplir de [4.23] la siguiente desigualdad:

$$-\sum_i \alpha_i K(\mathbf{z}, \mathbf{x_i}) \leq \frac{1}{2}(R^2 - 1 - \sum_{i,j} \alpha_i \alpha_j K(x_i, x_j))$$

Se define un nuevo parámetro ν:

$$\nu = \frac{1}{mC}$$

Donde ν ε (0,1]. Para ν = 0 (C = ∞) las muestras no son permitidas fuera de la descripción (no hay "outliers"). Cuando ν = 1/4 un cuarto de la muestra puede estar fuera de la descripción (1/4 de los datos serían "outliers").

A continuación vamos a describir un algoritmo [SPSSW01] que retorna una función f, que tome la función +1 en una región pequeña capturando la mayor parte de los puntos del conjunto de datos y, -1 en otro caso, la estrategia consiste en aplicar el conjunto de datos en un espacio de característica correspondiente al kernel y separarlos del origen con margen máximo. Para un nuevo punto **x,** el valor de f(**x**) es determinado evaluando a qué lado del hiperplano **x** es asignado en el espacio de característica. Para separar los datos del origen se resuelve el siguiente problema cuadrático.

$$\min_{w,\xi,\rho} \frac{1}{2}\|w\|^2 + \frac{1}{\nu m}\sum_{i=1}^{m}\xi_i - \rho \qquad [4.25]$$
$$\text{sujeto a} \quad (w \bullet \phi(x_i)) \geq \rho - \xi_i, \quad \xi_i \geq 0$$

Donde *m* es el tamaño de la muestra de entrenamiento, ρ es un parámetro de traslación, ϕ es una función que va de la muestra de entrenamiento a un espacio de producto interno, el cual puede ser hallado evaluando algún kernel.

En la siguiente proposición se probará que este parámetro es una cota superior de la fracción de objetos fuera de la descripción. Es decir, la fracción de "outliers".

Proposición 4.1 [SPSSW01]

Asumiendo que la solución de las ecuación [4.25] satisface $\rho \neq 0$, se cumplen los siguientes enunciados:

i) **ν** es una cota superior en la fracción de "outliers", esto es los puntos de entrenamiento fuera de la región estimada.

ii) **ν** es la cota inferior en la fracción de los SVs.

Demostración: Cuando **ν** es pequeño ρ es grande. Cuando cambiamos ρ el término $\sum_i \xi_i$ en [4.25] cambiará proporcionalmente al número de puntos que tengan ξ_i no cero ("outliers"), más aún, cuando cambiamos ρ en la dirección positiva el número de puntos que van a tener un ρ no cero, aquellos que están en el hiperplano (SVs), aumentará. En el óptimo de la ecuación [4.25] se cumple las partes *i*) y *ii*).

Se puede mostrar, que asintóticamente, ν es igual a la fracción de los SVs y la fracción de los "outliers", con probabilidad 1 [SPSSW01].

4.3 Mejorando el rendimiento del SVM penalizando los "outliers".

Zhan y Shen [ZS05] proponen incrementar la eficiencia del clasificador SVM usando una función no-lineal para penalizar la presencia de "outliers". El método que ellos proponen evita que la separación de hipersuperficies esté localmente deformada por causa de "outliers" en el conjunto de entrenamiento.

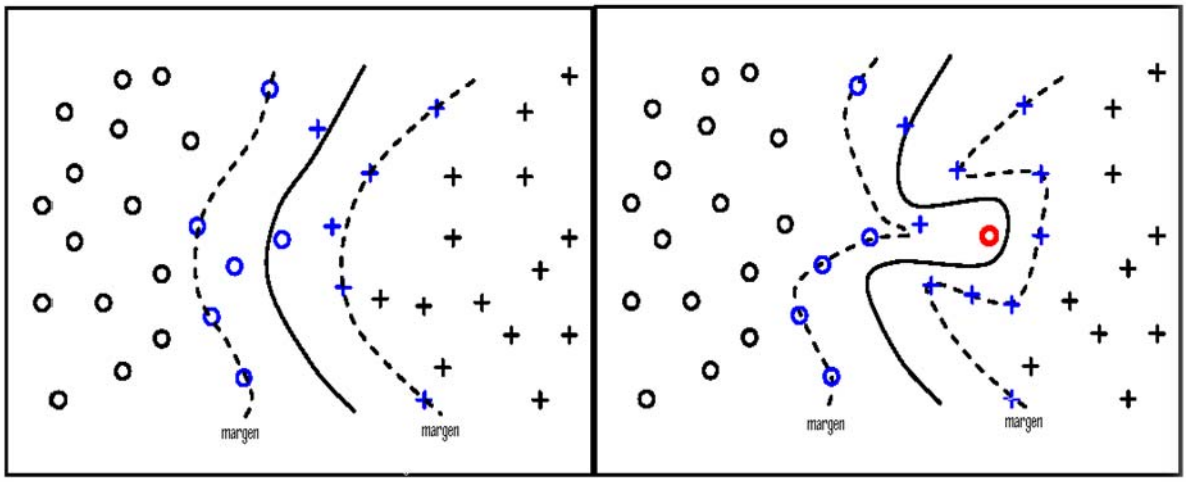
4.2.1 Descripción del problema

Los vectores de soporte pueden ser categorizados en dos tipos. El primer tipo de vectores de soporte, son aquellas instancias que se localizan en el margen de separación de la hipersuperficie. El segundo tipo de vectores de soporte son aquellas instancias que se localizan dentro de los márgenes. Para el SVM el segundo tipo de vectores de soporte son considerados como muestras mal clasificadas.

El SVM usualmente tiene un gran número de vectores de soporte, cuando las distribuciones de las muestras de entrenamiento de dos clases se superponen unas con otra. Algunos vectores de soporte pueden ser redundantes para parametrizar la hipersuperfice de separación. Por lo general, el SVM genera una hipersuperficie de separación deformada la cual es difícil ser parametrizada por un número pequeño de vectores de soporte. La única forma de decrecer este número de vectores es simplificando la hipersuperfice. La hipersuperficie deformada de SVM es crítica para la separación de clases no linealmente separables que están superpuestas en espacio de característica original.

En la Figura 4.1 (a), la hipersuperficie de separación del SVM tiene 12 vectores de soporte (cruces y círculos azules). La distribución de la muestra de entrenamiento en la Figura 4.2 (b) es casi la misma que el de la Figura 4.2(a), excepto una muestra adicional (círculo rojo). Sin embargo, la hipersuperfice de separación en la Figura 4.2 (b) es más deformada para satisfacer esta muestra adicional, y la muestra de entrenamiento para el SVM tiene 16 vectores de soporte. Esta muestra adicional que se encuentra en una región aislada puede ser un "outlier" producido por ruido o error. En consecuencia, la hipersuperficie deformada usada para satisfacer esta

muestra adicional decrecerá la habilidad de generalización del SVM e incrementará el costo computacional del SVM.



(a) **(b)**

Figura 4. 2 a) Ejemplo de hipersuperficie sin outlier y b) hipersuperficie con outlier. Tomada de [ZS05].

4.2.2 Reformulación de la función objetivo en SVM

Dado una muestra de entrenamiento tenemos $\{(\mathbf{x}_i, y_i) / \mathbf{x}_i \in R^n, y_i \in \{-1,1\}, i = 1,..., m\}$. El entrenamiento de SVM puede ser formulado como la solución de un problema de optimización cuadrática:

$$\min_{\mathbf{w},b,\xi_i} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\xi_i$$
sujetoa :
$$y_i(\mathbf{w} \bullet \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i \qquad [4.4]$$
$$\xi_i \geq 0$$

En esta función objetivo el segundo término es un término de penalidad, que consiste de un número de variables de holgura no-negativa ξ_i las cuales son usadas para construir un hiperplano con un margen suave. La sumatoria lineal de todas las variables de holgura ξ_i está

restringida por el segundo término en la función objetivo para evitar la solución trivial de que todas las ξ_i tomen valores grandes.

Zhan y Shen [ZS05] proponen una solución para que los "outliers" no degraden la habilidad de generalización del SVM. Ellos introducen un término de penalidad no lineal en la función objetivo del SVM. Esto hace que los "outliers" no tengan un impacto sobre la función objetivo. La función objetivo del SVM es reformulada como sigue:

$$\min_{\mathbf{w},b,\xi_i} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m} erf(\xi_i;\sigma)$$
sujeto a :
$$y_i(\mathbf{w}\bullet\phi(\mathbf{x}_i)+b) \geq 1-\xi_i \qquad [4.5]$$
$$\xi_i \geq 0$$

Donde $erf(\xi_i;\sigma)$ es una función de error no-lineal definida por:

$$erf(\xi_i;\sigma) = \frac{2}{\sqrt{\pi}\sigma}\int_0^{\xi_i} e^{-z^2/\sigma} dz \qquad [4.6]$$

para penalizar los "outliers" adaptativamente. La función de error *erf* suprimirá las variables de holgura cuando ellas son grandes, de esta forma la función objetivo no será dominada por las variables de holgura grandes, y así la hipersuperficie de separación ya no será distorsionada por un "outlier" extremo.

Ahora usando la teoría lagrangiana [A.1] podemos resolver el problema cuadrático [4.5], para esto introducimos los multiplicadores de Lagrange α_i y r_i

$$L(\mathbf{w},b,\xi_i,\alpha,r) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m} erf(\xi_i) - \sum_{i=1}^{m}\left[\alpha_i(y_i(\mathbf{w}\bullet\phi(\mathbf{x}_i)+b)-(1-\xi_i)) + r_i\xi_i\right] \qquad [4.7]$$

$\xi_i \geq 0$ y $r_i \geq 0$

Usando la condición de Kuhn-Tucker [A.2] podemos expresar la ecuación anterior como un problema dual. Diferenciando la ecuación [4.7] con respecto a $\mathbf{w}$, b y ξ_i e igualando derivadas a cero tenemos que:

$$\frac{\partial}{\partial \mathbf{w}} L = \mathbf{w} - \sum_{i=1}^{m} \alpha_i y_i \phi(x_i) = 0 \qquad [4.8]$$

$$\frac{\partial}{\partial \xi_i} L = Cerf'(\xi_i) - \alpha_i - r_i = 0 \qquad [4.9]$$

$$\frac{\partial}{\partial b} L = \sum_{i=1}^{m} \alpha_i y_i = 0 \qquad [4.10]$$

Reescribiendo [4.7] tenemos

$$\begin{aligned} L(\mathbf{w}, b, \xi, \boldsymbol{\alpha}, \mathbf{r}) &= \frac{1}{2}\langle \mathbf{w} \bullet \mathbf{w} \rangle + C\sum_{i=1}^{m} erf(\xi_i) - \sum_{i=1}^{m} \alpha_i y_i (\langle \phi(\mathbf{x}_i) \bullet \mathbf{w} \rangle) - b\sum_{i=1}^{m} \alpha_i y_i + \sum_{i=1}^{m} \alpha_i - \sum_{i=1}^{m} \alpha_i \xi_i - \sum_{i=1}^{m} r_i \xi_i \\ L(\mathbf{w}, b, \xi, \boldsymbol{\alpha}, \mathbf{r}) &= \frac{1}{2}\langle \mathbf{w} \bullet \mathbf{w} \rangle - \sum_{i=1}^{m} \alpha_i y_i (\langle \phi(\mathbf{x}_i) \bullet \mathbf{w} \rangle) - b\sum_{i=1}^{m} \alpha_i y_i + \sum_{i=1}^{m} \alpha_i + C\sum_{i=1}^{m} erf(\xi_i) - \sum_{i=1}^{m} \xi_i (\alpha_i + r_i) \end{aligned} \qquad [4.11]$$

Sustituyendo [4.8], [4.9] y [4.10] en [4.11] se obtienen:

$$\begin{aligned} L(\mathbf{w}, b, \xi, \boldsymbol{\alpha}, \mathbf{r}) &= \frac{1}{2}\sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \langle \phi(x_i) \bullet \phi(x_j) \rangle - \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \langle \phi(x_i) \bullet \phi(x_j) \rangle + \sum_{i=1}^{m} \alpha_i + C\sum_{i=1}^{m} erf'(\xi_i) - \\ &\quad \sum_{i=1}^{m} C\xi_i erf'(\xi_i) \\ L(\mathbf{w}, b, \xi, \boldsymbol{\alpha}, \mathbf{r}) &= \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \langle \phi(x_i) \bullet \phi(x_j) \rangle - C\sum_{i=1}^{m} \xi_i erf'(\xi_i) + erf(\xi_i) \end{aligned}$$

Por lo tanto el problema dual es:

Maximizar

$$L(\mathbf{w}, b, \xi, \boldsymbol{\alpha}, \mathbf{r}) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \langle \phi(x_i) \bullet \phi(x_j) \rangle - C\sum_{i=1}^{m} [\xi_i erf'(\xi_i, \sigma) + erf(\xi_i, \delta)] \qquad [4.12]$$

Donde $erf'(\xi_i)$ es la derivada de la función de error [4.6]

Sujeto a $0 \le \alpha_i \le Cerf'(\xi_{i,}\sigma)$ $\forall$i [4.13]

$$\xi_i \geq 0 \qquad [4.14]$$

La ecuación anterior no está restringida por una constante global C, porque $C \cdot erf'(\xi_i;\sigma)$ es adaptativa a cada muestra de acuerdo a su correspondiente ξ_i. Como $erf'(\xi_i;\sigma)$ es actualmente una función gaussiana el multiplicador de Lagrange α_i de la muestra de entrenamiento con variables de holgura grandes ξ_i está restringida por una cota superior pequeña. En consecuencia, esta muestra tiene poca contribución en la construcción de la hipersuperficie de separación.

Puesto que la función objetivo tiene una forma similar a la forma estándar del SVM se puede diseñar un sistema iterativo para entrenar el SVM reformulado. Los α_i serán optimizados utilizando un método de entrenamiento similar al estándar SVM excepto usando la restricción adaptativa $0 \leq \alpha_i \leq C \cdot erf'(\xi_i;\sigma)$, donde ξ_i es calculado como sigue:

$$\xi_i = \sum_{i=1}^{m} \alpha_i y_i \mathbf{K}(\mathbf{x}_i, \mathbf{x}) + b - 1$$

Si el parámetro σ tiene un valor grande, la función de error se comporta en forma similar como un término de holgura lineal usado en el SVM estándar. Si el parámetro σ se va disminuyendo cada vez más entonces la función de error empieza a suprimir las variables de holgura grandes y de esta manera se genera la hipersuperficie de separación óptima para el SVM reformulado.

CAPITULO V

RESULTADOS EXPERIMENTALES

5.1 Conjuntos de datos usados

Los experimentos se realizaron en los siguientes conjuntos de datos: Iris, Bupa, Diabetes, Vehículo e Ionosfera. Estos conjuntos de datos están disponibles en el repositorio de base de datos de aprendizaje automático en la Universidad de California en Irvine [BM98]. Un resumen de las características de los conjuntos de datos aparece en la Tabla [5.1].

Conjunto de Datos	**Número de instancias**	**Número de atributos**	**Número de clases**
Iris	150	4	3
Ionosfera	351	32	2
Bupa	345	6	2
Diabetes	768	8	2
Vehículo	846	18	4

Tabla 5. 1 Descripción de los conjunto de datos.

5.2 Estimaciones de la tasa de error de clasificación usando el SVM

Aplicando la función **svm** de la librería **e1071** de **R** para clasificación con SVM, a los conjuntos de datos de la tabla 5.1, se obtienen los estimados por validación cruzada del error de mala clasificación. Estos estimados aparecen en la siguiente tabla:

Conjuntos de datos	Error %			SV		
	Lineal	Radial	Polinomial	Lineal	Radial	Polinomial
Iris	3.333	4.000	8.666	29	51	54
Bupa	30.725	29.855	40.869	258	269	284
Diabetes	23.698	23.568	25.651	401	435	425
Vehículo	20.213	23.286	25.886	437	590	649
Ionosfera	17.664	5.983	26.781	102	121	182

Tabla 5. 2 El error de clasificación y número de vectores de soporte usando los kernels lineal, radial y polinomial

Se puede notar que los kernels lineal y radial tienen similar rendimiento, el cual es mucho mejor que el del kernel polinomial. Salvo en Ionosfera, donde el kernel radial supera ampliamente a los otros dos kernels. Por otro lado, el número de vectores de soporte requeridos por el kernel lineal es menor que el de los otros dos kernels.

5.3. Detección de casos anómalos y su efecto en el clasificador SVM

Los "outliers" que aparecen listado mas adelante fueron encontrados usando tres método al mismo tiempo: detección de "outliers" basado en técnicas estadísticas (**robout**), detección de "outliers" basado en distancia (**baysout**) y detección de "outliers" basados en densidad local (**maxlof**). Las funciones que detectan estos "outliers" se encuentran en la librería **dprep** de **R** [AR06].

Iris

Outliers en la clase 1: (9)

16, 15, 34, 42, 44, 24, 23, 19,45

Outliers en la clase 2: (9)

71, 63, 58, 61, 94, 99, 69, 84, 88

Outliers en la clase 3: (7)

107, 119, 132, 118, 120, 123, 110

Bupa

Outliers en la clase 1: (21)

190, 317, 316, 182, 205, 335, 345, 343, 189, 312, 344, 175, 168, 183, 25, 172, 311, 167, 326, 148, 261

Outliers en la clase 2: (20)

85, 36, 134, 233, 331, 300, 179, 323, 342, 111, 115, 77, 186, 252, 294, 139, 307, 224, 286, 157

Diabetes

Outliers en la clase 1: (38)

229, 372, 454, 488, 59, 623, 50, 61, 82, 427, 495, 523, 76, 183, 343, 538, 460, 295, 685, 457, 124,337,476, 496, 490, 264,149, 675, 248, 704, 8, 287, 154, 487, 646, 259, 520, 261

Outliers en la clase 2: (26)

707, 580, 126, 10, 350, 503, 371, 194, 358, 14, 5, 46, 446, 662, 333, 485, 436, 79, 585, 410, 9, 656, 754,187, 716, 255.

Vehículo

Outliers en clase 1 (19):

5, 101, 128, 545, 816, 688, 836, 734, 86, 382, 322, 532, 397, 6, 55, 275, 156, 684, 687

Outliers en clase 2 (18):

613, 114, 412, 182, 797, 74, 90, 729, 580, 592, 464, 516, 161, 503, 350, 517, 16, 211

Outliers en clase *3* (27):

124, 615, 12, 420, 423, 232, 290, 368, 566, 663, 184, 250, 835, 352, 689, 379, 139, 27, 261, 395, 637,600, 311, 662, 777, 440, 643

Outliers en clase 4 (9):

389, 38, 136, 707, 292, 524, 392, 273, 353

Ionosfera

Outliers en la clase 1: (4)

27, 143, 275, 290

Outliers en la clase 2: (8)

242, 261,170, 237,19, 209, 3, 58

Con la finalidad de determinar si los "outliers" han sido detectados en forma eficiente, se extraen cinco muestras aleatorias del conjunto original de datos. Estas muestras son de un tamaño igual al conjunto original menos el número de "outliers" que encontramos con los métodos anteriores. También se asegura que en la muestra no salgan los "outliers" detectados anteriormente. La siguiente tabla reporta el promedio de las tasa de errores de clasificación obtenidos con las 5 muestras.

Conjuntos de datos	**Error %**			**SV**		
	Lineal	Radial	Polinomial	Lineal	Radial	Polinomial
Iris	3.680	4.648	8.720	27	46	48
Bupa	32.237	30.461	41.118	224	245	253
Diabetes	23.125	24.024	26.619	369	402	390
Vehículo	19.845	23.570	27.115	403	548	598
Ionosfera	16.873	5.664	26.372	101	119	178

Tabla 5. 3 El error de clasificación y número de vectores de soporte usando los kernels lineal, radial y polinomial, después de sacar una muestra aleatoria.

Al igual que en la Tabla 5.2, se puede notar que los kernels lineal y radial tienen similar rendimiento, el cual es mucho mejor que el del kernel polinomial. En Ionosfera, el kernel radial supera ampliamente a los otros dos kernels. El número de vectores de soporte requeridos por el kernel lineal es menor que el de los otros dos kernels.

Para evaluar el efecto de los "outliers" en el rendimiento del clasificador, se eliminan los "outliers" en cada conjunto de datos y aplicamos de nuevo el clasificador SVM. Las tasa estimadas de los errores de clasificación se muestran en tabla 5.4.

Comparando las Tablas 5.3 y 5.4 se observa que:

1) Para Iris, cuando se eliminan los "outliers", el error de clasificación es menor en un 1.28% y se usan 3 vectores de soporte menos que cuando se elimina una muestra aleatoria considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 2.258% y los vectores de soporte son 2 menos. Para el kernel polinomial, el error es menor en un 1.82% y los vectores de soporte son 4 menos.

Conjuntos de datos	Error %			SV		
	Lineal	Radial	Polinomial	Lineal	Radial	Polinomial
Iris	2.400	2.400	6.900	24	44	44
Bupa	26.316	24.671	34.539	197	223	231
Diabetes	21.591	22.869	24.574	338	382	357
Vehículo	17.852	20.957	21.086	353	505	535
Ionosfera	15.929	5.605	25.663	101	112	171

Tabla 5. 4 El error de clasificación y número de vectores de soporte usando los kernels lineal, radial y polinomial después de eliminar los datos anómalos.

2) Para Bupa, cuando se eliminan los "outliers", el error de clasificación es menor en un 5.921% y se usan 27 vectores de soporte menos que cuando se elimina una muestra aleatoria considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 5.789% y los vectores de soporte son 22 menos. Para el kernel polinomial, el error es menor en un 6.579% y los vectores de soporte son 22 menos.

3) Para Diabetes, cuando se eliminan los "outliers", el error de clasificación es menor en un 1.534% y se usan 31 vectores de soporte menos que cuando se elimina una muestra aleatoria considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 1.155% y los vectores de soporte son 20 menos. Para el kernel polinomial, el error es menor en un 2.045% y los vectores de soporte son 33 menos.

4) Para Vehículo, cuando se eliminan los "outliers", el error de clasificación es menor en un 1.992% y se usan 50 vectores de soporte menos que cuando se elimina una muestra aleatoria considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 2.613% y los vectores de soporte son 43 menos. Para el kernel polinomial, el error es menor en un 6.028% y los vectores de soporte son 63 menos.

5) Para Ionosfera, cuando se eliminan los "outliers", el error de clasificación es menor en un 0.944% y se usan el mismo número de vectores de soporte que cuando se elimina una muestra

aleatoria considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 0.059% y los vectores de soporte son 7 menos. Para el kernel polinomial, el error es menor en un 0.708% y los vectores de soporte son 7 menos.

Se nota que en general, la tasa de error de clasificación eliminando "outliers" es menor que la tasa de error eliminando una muestra aleatoria del mismo tamaño que la cantidad de "outliers" detectados. Eso significa, que la detección de "outliers" ha funcionado.

5.4 Detección de casos anómalos usando SVM para clasificación con una sola clase.

Para detectar "outliers" usando SVM para clasificación con una sola clase usaremos el kernel radial. En los experimentos se usó la función **svm** de la librería **e1071** de R. Esta función tiene dos parámteros ν y s definidos en la Sección 4.3. Se ha cambiando los valores de estos parámetros de tal manera que haya la mayor coincidencia posible entre los "outliers" determinados por el SVM y los detectados por los métodos de la sección 5.3. Usualmente, se tomó $\nu = 0.1$ ó $\nu = 0.2$. Esto es al menos un 20% de los datos fueron considerados como "outliers". El listado de "outliers" para cada conjunto aparece a continuación

Iris

Clase 1	ν=0.2(10) s: 0.25	13 14 19 24 25 26 37 42 44 45
Clase 2	ν =0.1(10) s: 0.25	61 65 68 69 71 74 78 84 85 99
Clase 3	ν =0.2(9) s: 0.25	101 107 110 115 118 119 120 135 136

Tabla 5. 5 Outliers detectados en Iris usando el SVM.

Bupa

Clase1	ν = .02(26) s: 0.1666667	20 22 25 93 106 109 147 172 175 183 189 190 195 205 211 212 244 261 312 316 317 326 329 335 344 345
Clase 2	ν = 0.2(39) s: 0.1666667	36 38 41 53 63 69 77 85 111 115 122 123 134 136 154 156 157 179 185 186 187 188 193 218 220 224 233 237 252 278 286 300 304 323 330 331 337 339 342

Tabla 5. 6 Outliers detectados en Bupa usando el SVM.

Ionosfera

Clase 1	ν =0.1 (22) s: 0.01	14 26 27 115 143 164 167 172 176 198 200 220 231 257 263 270 272 275 290 294 307 313
Clase 2	ν =0.2(21) s: 0.03125	3 33 34 73 85 108 124 152 154 170 174 195 201 209 211 242 255 261 262 323 332

Tabla 5. 7 Outliers detectados en ionosphera usando el SVM.

Diabetes

Clase 1	ν =0.1(50) s: 0.003	8 13 50 58 59 61 76 82 87 107 146 154 183 212 213 229 248 259 287 336 337 343 363 372 427 431 454 460 487 488 490 495 519 523 538 550 559 590 602 622 623 646 659 673 674 685 698 745 764
Clase 2	ν =0.1(27) s: 0.01	5 9 10 14 46 79 126 160 178 194 270 304 320 350 358 371 410 446 503 580 585 605 620 692 707 754 760

Tabla 5. 8 Outliers detectados en Diabetes usando el SVM.

Vehículo

Clase1	ν =0.1(21) s: 0.001	5 6 24 86 101 179 247 280 296 322 397 418 495 532 565 684 688 734 791 831 836
Clase 2	ν: 0.1(20) s: 0.009	72 74 114 143 161 203 252 350 412 448 487 516 517 587 590 613 616 676 828 841
Clase 3	ν =0.2(43) s: 0.01	12 27 45 51 91 118 121 124 141 167 184 197 232 244 250 261 279 284 290 343 347 352 379 420 423 506 507 518 537 560 571 606 615 621 624 637 689 722 762 798 810 835 844
Clase 4	ν =0.1(19) s: 0.01	9 36 38 42 136 147 220 231 245 273 292 389 524 586 656 661 707 752 760

Tabla 5. 9 Outliers detectados en Vehículo usando el SVM.

Comparando las tablas 5.5 a 5.9 con los “outliers” encontrados en la Sección 5.3 obtenemos los siguientes resultados.

Iris

Clase 1

SVM una clase 13, 14, 19, 24, 25, 26, 37, 42, 44, 45

Anteriores 16, 15, 34, 42, 44, 24, 23, 19, 45

Coincidencias (SVM una clase/anteriores = 5/9) 55.5%

Clase2

SVM una clase 61, 65, 68, 69, 71, 74, 78, 84, 85, 99

Anteriores 71, 63, 58, 61, 94, 99, 69, 84, 88

Coincidencias (SVM una clase/anteriores = 5/9) 55.5%

Clase 3

SVM una clase 101, 107, 110, 115, 118, 119, 120, 135, 136

Anteriores 107, 119, 132, 118, 120, 123, 110

Coincidencias (SVM una clase/anteriores = 5/7) 71.4%

Bupa

Clase 1

SVM una clase 20, 22, 25, 93, 106, 109, 147, 172, 175, 183, 189, 190, 195, 205, 211, 212, 244, 261, 312, 316, 317, 326, 329, 335, 344, 345

Anteriores 190, 317, 316, 182, 205, 335, 345, 343, 189, 312, 344, 175, 168, 183, 25, 172, 311, 167, 326, 148, 261

Coincidencias (SVM una clase/anteriores = 14/21) 66.6%

Clase 2

SVM una clase 36, 38, 41, 53, 63, 69, 77, 85, 111, 115, 122, 123, 134, 136, 154, 156, 157, 179, 185, 186, 187, 188, 193, 218, 220, 224, 233, 237, 252, 278, 286, 300, 304, 323, 330, 331, 337, 339, 342

Anteriores 85, 36, 134, 233, 331, 300, 179, 323, 342, 111, 115, 77, 186, 252, 294, 139, 307, 224, 286, 157

Coincidencias (SVM una clase/anteriores = 17/20) 85%

Ionosfera

Clase 1

SVM una clase 14, 26, 27, 115, 143, 164, 167, 172, 176, 198, 200, 220, 231, 257, 263, 270, 272, 275, 290, 294, 307, 313,

Anteriores 27, 143, 275, 290

Coincidencias (SVM una clase/anteriores = 4/4) 100%

Clase 2

SVM una clase 3, 33, 34, 73, 85, 108, 124, 152, 154, 170, 174, 195, 201, 209, 211, 242, 255, 261, 262, 323, 332

Anteriores 3, 19, 58, 170, 209, 242, 261, 237

Coincidencias (SVM una clase/anteriores = 5/8) 62.5%

Diabetes

Clase 1

SVM una clase 8, 13, 50, 58, 59, 61, 76, 82, 87, 107, 146, 154, 183, 212, 213, 229, 248, 259, 287, 336, 337, 343, 363, 372, 427, 431, 454, 460, 487, 488, 490, 495, 519, 523, 538, 550, 559, 590, 602, 622, 623, 646, 659, 673, 674, 685, 698, 745, 764

Anterior 229, 372, 454, 488, 59, 623, 50, 61, 82, 427, 495, 523, 76, 183, 343, 538, 460, 295, 685, 457, 124, 337, 476, 496, 490, 264, 149, 675, 248, 704, 8, 287, 154, 487, 646, 259, 520, 261

Coincidencias (SVM una clase/anteriores = 28/38) 73.7%

Clase 2

SVM una clase 5, 9, 10, 14, 46, 79, 126, 160, 178, 194, 270, 304, 320, 350, 358, 371, 410, 446, 503, 580, 585, 605, 620, 692, 707, 754, 760

Anteriores 707, 580, 126, 10, 350, 503, 371, 194, 358, 14, 5, 46, 446, 662, 333, 485, 436, 79, 585, 410, 9, 656, 754, 187, 716, 255

Coincidencias (SVM una clase/anteriores =18 /25) 72%

Vehículo

Clase 1

SVM una clase 5, 6, 24, 86, 101, 179, 247, 280, 296, 322, 397, 418, 495, 532, 565, 684, 688, 734, 791, 831, 836

Anteriores 5, 101, 128, 545, 816, 688, 836, 734, 86, 382, 322, 532, 397, 6, 55, 275, 156, 684, 687

Coincidencias (SVM una clase/anteriores = 11/19) 57.9%

Clase 2

SVM una clase 72, 74, 114, 143, 161, 203, 252, 350, 412, 448, 487, 516, 517, 587, 590, 613, 616, 676, 828, 841

Anteriores 613, 114, 412, 182, 797, 74, 90, 729, 580, 592, 464, 516, 161, 503, 350, 517, 16, 211

Coincidencias (SVM una clase/anteriores = 8/17) 47%

Clase 3

SVM una clase 12, 27, 45, 51, 91, 118, 121, 124, 141, 167, 184, 197, 232, 244, 250, 261, 279, 284, 290, 343, 347, 352, 379, 420, 423, 506, 507, 518, 537, 560, 571, 606, 615, 621, 624, 637, 689, 722, 762, 798, 810, 835, 844

Anteriores 124, 615, 12, 420, 423, 232, 290, 368, 566, 663, 184, 250, 835, 352, 689, 379, 139, 27, 261, 395, 637, 600, 311, 662, 777, 440, 643

Coincidencias (SVM una clase/anteriores = 16/27) 59.2%

Clase 4

SVM una clase 9, 36, 38, 42, 136, 147, 220, 231, 245, 273, 292, 389, 524, 586, 656, 661, 707, 752, 760

Anteriores 389, 38, 136, 707, 292, 524, 392, 273, 353

Coincidencias (SVM una clase/anteriores = 7/9) 77.7%

La siguiente tabla resume los resultados anteriores

Datos	Numero de "outliers" con métodos clásicos	Numero de "outliers" con SVM	Coincidencias %
Iris	25	29	51.72
Bupa	41	65	75.60
Ionosfera	12	43	75.00
Diabetes	64	77	71.87
Vehiculo	73	103	57.53

Tabla 5. 10 porcentaje de coincidencias de los "outliers" con los métodos SVM una clase y los métodos de la sección 4.1.

Notar que en todos los conjuntos el SVM de una clase detecta mucho más "outliers" que los métodos usuales descritos en la sección 4.1.

5.5 Efecto en el clasificador SVM después de eliminar los casos anómalos hallados usando SVM para clasificación con una sola clase.

Con la finalidad de determinar si los "outliers" han sido detectados en forma eficiente con el clasificador SVM de una clase, se extraen cinco muestras aleatorias del conjunto original de datos. Estas muestras son de un tamaño igual al conjunto original menos el número de "outliers" que encontramos con los métodos anteriores. También se asegura que en la muestra no salgan los "outliers" detectados anteriormente. La siguiente tabla reporta el promedio de las tasa de errores de clasificación usando el SVM usual obtenidos con las 5 muestras.

Conjuntos de datos	Error %			SV		
	Lineal	Radial	Polinomial	Lineal	Radial	Polinomial
Iris	4.297	6.281	10.413	27	47	46
Bupa	32.071	32.5	40.286	210	231	229
Diabetes	23.642	23.931	26.301	367	393	387
Vehículo	20.484	23.984	27.537	396	532	576
Ionosfera	16.039	5.454	28.248	89	114	161

Tabla 5. 11 El error de clasificación y número de vectores de soporte usando los kernels lineal, radial y polinomial, después de sacar una muestra aleatoria.

Para evaluar el efecto de los "outliers" en el rendimiento del clasificador, se eliminan los "outliers" en cada conjunto de datos y aplicamos de nuevo el clasificador SVM. Las tasas estimadas de los errores de clasificación se muestran en tabla 5.12.

Conjuntos de datos	Error %			SV		
	Lineal	Radial	Polinomial	Lineal	Radial	Polinomial
Iris	1.652	0.826	4.958	22	42	45
Bupa	27.500	27.142	37.500	189	209	219
Diabetes	21.242	22.109	23.699	328	380	348
Vehículo	18.438	21.668	23.418	355	495	528
Ionosfera	14.935	4.220	21.103	88	100	153

Tabla 5. 12 El error de clasificación y numero de vectores de soporte usando los kernels lineal, radial y polinomial después de eliminar los datos anómalos hallados usando SVM.

Comparando las Tablas 5.11 y 5.12 se observa que:

1) Para Iris, cuando se eliminan los "outliers", el error de clasificación es menor en un 2.645% y se usan 5 vectores de soporte menos que cuando se elimina una muestra aleatoria considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 5.455% y los vectores de soporte son 5 menos. Para el kernel polinomial, el error es menor en un 5.455% y los vectores de soporte son 1 menos.

2) Para Bupa, cuando se eliminan los "outliers", el error de clasificación es menor en un 4.571% y se usan 21 vectores de soporte menos que cuando se elimina una muestra aleatoria

considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 5.358% y los vectores de soporte son 22 menos. Para el kernel polinomial, el error es menor en un 2.785% y los vectores de soporte son 10 menos.

3) Para Diabetes, cuando se eliminan los "outliers", el error de clasificación es menor en un 2.399% y se usan 39 vectores de soporte menos que cuando se elimina una muestra aleatoria considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 1.821% y los vectores de soporte son 13 menos. Para el kernel polinomial, el error es menor en un 2.601% y los vectores de soporte son 39 menos.

4) Para Vehículo, cuando se eliminan los "outliers", el error de clasificación es menor en un 2.046% y se usan 41 vectores de soporte menos que cuando se elimina una muestra aleatoria considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 2.316% y los vectores de soporte son 37 menos. Para el kernel polinomial, el error es menor en un 4.119% y los vectores de soporte son 48 menos.

5) Para Ionosfera, cuando se eliminan los "outliers", el error de clasificación es menor en un 1.104% y se usa un vector de soporte menos que cuando se elimina una muestra aleatoria considerando un kernel lineal. Mientras que si se usa el kernel radial, la tasa de error es menor en un 1.234% y los vectores de soporte son 14 menos. Para el kernel polinomial, el error es menor en un 7.144% y los vectores de soporte son 8 menos.

De lo anterior, se puede ver que la tasa de error de clasificación eliminando "outliers" es menor que la tasa de error eliminando una muestra aleatoria del mismo tamaño que la cantidad de "outliers" detectados. Esto significa, que la detección de "outliers" ha funcionado.

Comparando las Tablas 5.4 y 5.12 se observa que cuando se detecta los "outliers" usando el SVM de un clase en lugar de los métodos usuales:
1) Para Iris, el error de clasificación disminuye en un 0.748% y los vectores de soporte disminuyen en 2 para el kernel lineal, un 1.574% y los vectores de soporte disminuyen en 2 con el kernel radial y un 1.942% y los vectores de soporte aumentan en 1 con el kernel polinomial.

2) Para Bupa, el error de clasificación aumenta en un 1.184% y los vectores de soporte disminuye en 8 para el kernel lineal, un 2.470% y los vectores de soporte disminuye en 14 con el kernel radial y un 2.960 % y los vectores de soporte disminuye en 12 con el kernel polinomial.

3) Para Diabetes, el error de clasificación disminuye en un 0.349% y los vectores de soporte disminuyen en 10 para el kernel lineal un 0.76% y los vectores de soporte disminuyen en 2 con el kernel radial y un 0.874 % y los vectores de soporte disminuyen en 9 con el kernel polinomial.

4) Para Vehículo, el error de clasificación aumenta en un 0.585% y los vectores de soporte aumentan en 2 para el kernel lineal, un 0.710% y los vectores de soporte disminuyen en 10 con el kernel radial y un 2.331 % y los vectores de soporte disminuyen en 7 con el kernel polinomial.

5) Para Ionosfera, el error de clasificación disminuye en un 0.994% y los vectores de soporte disminuyen en 3 para el kernel lineal, un 1.385% y los vectores de soporte disminuyen en 2 con el kernel radial y un 4.560 % y los vectores de soporte disminuyen en 18 con el kernel polinomial.

CAPITULO VI

CONCLUSIONES Y TRABAJOS FUTUROS

El estudio experimental llevado a cabo en esta tesis nos conduce a las siguientes conclusiones:

- En los conjuntos considerados en esta tesis el clasificador SVM con los kernel lineal y radial tienen mejor rendimiento, que el SVM con kernel polinomial.

- Se aplicó el clasificador SVM al conjunto de datos después de remover los "outliers" detectados con una combinación de métodos estadísticos, métodos de distancia y métodos de densidad local y se encontró que tanto la tasa de error de clasificación como el número de vectores de soporte disminuyeron.

- Por otro lado se aplicó una variante del SVM con kernel radial, para clasificación con una clase, con la finalidad de detectar casos anómalos en la muestra de entrenamiento. Luego, se aplicó el clasificador SVM al conjunto de datos después de remover los "outliers" hallados usando esta metodología y se encontró que tanto la tasa de error de clasificación como el número de vectores de soporte disminuyeron.

- La tasa de error de clasificación obtenida después de remover los "outliers" detectados con ambas metodologías, es bastante similar. A pesar de que el porcentaje de coincidencias entre los "outliers" detectados por ambas metodologías es cerca del 70%, solamente. Sin embargo, el número de "outliers" detectados con el SVM es mucho mayor que con los métodos usuales.

Como trabajos futuros podemos mencionar los siguientes:

- Investigar cómo detectar casos anómalos usando otras variantes del SVM.
- Hacer un estudio más amplio para tratar de caracterizar de alguna manera los conjuntos de datos donde los “outliers” son detectados más eficientemente usando SVM.
- Estudiar maneras de penalizar a los “outliers” para que no afecten el rendimiento del clasificador SVM.

Bibliografía

[AR06] E. Acuña y C. Rodríguez. Dprep: Data Preprocessing and Visualization Functions for Classification. URL: http://cran.r-project.org/web/packages/dprep/index.html 2006

[BS03] S.D., Bay, y M. Schwabacher. Mining distance-based outliers in near linear time with randomization and a simple pruning rule. Proceedings from the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2003.

[Bet05] G. Betancourt. Las Máquinas De Soporte Vectorial (SVMs), Scientia et Technica Año XI, No 27, Abril 2005.

[BM98] C.L. Blake y C.J. Mertz, UCI Repository of machine learning databases [http://www.ics.uci.edu/mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science, 1998.

[BCDDGJLMSV94]L. Bottou, C. Cortes, J. Denker, H. Drucker, L. Jackel, Y. Lecum, U. Muller, E. Sanckinger, P. Simard, y V. Vapnik. Comparison of classifier methods; a case study in handwriting digit recognition. In International Conference on Pattern Recognition, pages 77-87.1994.

[BKNS00] M., Breuning, H., Kriegel, R.T, Ng, y J. Sander. LOF: Identifying density-based local outliers. In Proceedings of the ACM SIGMOD International Conference on Management of Data. 2000.

[CS04] N. Cristianini y J. Shawe-Taylor, .An Introduction to Support Vector Machines and Other kernel based learning methods. Cambridge University Press, 2004.

[DLMW06] E. Dimitriadou, K. Hornik, F. Leischz, D. Meyer y A. Weingessel. Misc Functions of the Department of Statistics (e1071). URL: http://cran.r-project.org/web/packages/e1071/index.html. 2006

[Fri96] J. Friedman. Another approach to polychotomous classification. Technical Report, Department of Statistics, Stanford University, 1996.

[KSBM01] S.S Keerthi, S.K. Shevade, C. Bhattacharyya y K.R.K. Murthy. improvements to platt's SMO algorithm for SVM classifier design. *Neural Computation*, 13:637–649, 2001.

[KPD90] S. Knerr, L. Personnaz, y G. Dreyfus. Single-layer learning revisited: A Stepwise Procedure for Building and Training a Neural Network. In J.Folgelmal, ed Neurocomputing: Algorithms, Architectures and Applications.Springer-Verlag,1990.

[KN97] E. Knorr, y R. Ng. A unified approach for mining outliers. Proc. KDD: 219–222. 1997.

[KN98] E. Knorr, y R. Ng. Algorithms for mining distance-based outliers in large datasets. Proc. 24th Int. Conf. Very Large Data Bases, VLDB, 392–403, 24–27. 1998.

[KN00] E. Knorr., R. Ng, y V. Tucakov. Distance-based outliers: Algorithms and applications. VLDB Journal: Very Large Data Bases, 8(3–4):237–253. 2000.

[Kre99] U. Krebel. Pairwise Classification and Support Vector Machines. In B. Scholkopf, C.J.C.Burgues, and A.J.Smola, editors, Advances in kernel methods- support vector learning, pages 225-268, Cambridge,MA, MIT Press. 1999.

[OFG97] E. Osuna, R Freund y F. Girosi. An Improved training algorithm for support vector machines. Proc of IEEE NNSP'97,Amelia island,FL, 24-26 sep 1997.

[P98] J.C. Platt. Sequencial minimal optimization: A fast algoritmo for training support vector machines. Microsoft research . Technical Report MSR-TR-98-14,1998.

[PCS00] J.C. Platt, N. Cristianini, y J. Shawe-Taylor. Large margin DAGs for multiclass classification. In Advances in neural Information processing systems, volume 12, pages 547-553.MIT Press, 2000.

[RRS00] S. Ramaswamy, R., Rastogi, y K. Shim, Efficient algorithms for mining outliers from large datasets. In Proceedings of the ACM SIGMOD International Conference on Management of Data. 2000.

[Rou85] P. Rousseeuw. Multivariate estimation with high breakdown point. Mathematical statistics and applications. 1985.

[RL87] P. Rousseeuw, y A. Leroy. *Robust Regression and Outlier Detection*. John Wiley, New York. 1987.

[SPSSW01] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, R. C. Williamson. Estimating the Support of a High-Dimensional Distribution. Neural Computation 13, 1443–1471 c°2001 Massachusetts Institute of Technology. 2001

[TD01] D.M.J. Tax , R.P.W. Duin. Outliers and data descriptions, Pattern Recognition Group, Delft University of Technology, Lorentzweg 1, 2628 CJ Delft, The Netherlands.2001

[Va95] V. Vapnik. The Nature of Statistical Learning Theory. Springer-Verlag, New York, 1995.

[VC71] V. Vapnik y A. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. Theory of Probability and its Applications, 16(2):264-280, 1971.

[VGS97] V. Vapnik, S. Golowich, y A. Smola. Support vector method for function approximation.regression, estimation, and signal processing, In M. Mozer, M. Jordan, and T. Petsche, editors, Advances in Neural Information Processing Systems 9, pg. 281-287,Cambridge, MA, 1997.

[Web02] A. Webb. Statistical Pattern Recognition, John Wiley & Sons Ltd, 2002.

[ZS05] Y. Zhan y D. Shen. Increasing efficiency of SVM by adaptively penalizing outliers. EMMCVPR pag. 539-551. 2005.

Apéndice

[A.1] Lagrangiano cuando se tienen dos restricciones

Dado un problema de optimización en su forma estándar

$$\begin{aligned} \min \quad & f_o(x) \\ \text{sujeto a} \quad & f_i(x) \le 0, \quad i \in \{1,...,m\} \\ & h_i(x) = 0, \quad i \in \{1,...,p\} \end{aligned}$$

La función lagrangiana se define por:

$$L(x,\lambda,\nu) = f_o(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x)$$

[A.2] Teorema de Karush-Kuhn-Tucker (KKT)

Modelo de optimización convexa

CP: minimizar *f(x)*

Sujeto a:

$$\begin{aligned} & g_1(x) \le 0 \\ & \quad \text{M} \\ & g_m(x) \le 0 \\ & Ax = b \\ & x \in R^n \end{aligned}$$

CP es llamado problema de optimización convexa si $f(x), g_1(x),...,g_m(x)$ son funciones convexas.

Teorema Karush-Kuhn-Tucker (KKT)

Supóngase que f(x), g_i(x) con j = 1,..., m, son funciones convexas. Entonces, $\bar{x}$ resuelve CP, si y solo si existen m números $\mu_1,...,\mu_n$ que satisfagan todas las condiciones necesarias siguientes:

$$\left.\begin{array}{ll} 1) & \dfrac{\partial l}{\partial x_i} = \dfrac{\partial f(\bar{x})}{\partial x_i} + \sum_{j=1}^{m} \mu_j \dfrac{\partial g_j(\bar{x})}{\partial x_i} \geq 0 \\ 2) & \bar{x}_i \dfrac{\partial l}{\partial x_i} = \bar{x}_i \left(\dfrac{\partial f(\bar{x})}{\partial x_i} + \sum_{j=1}^{m} \mu_j \dfrac{\partial g_j(\bar{x})}{\partial x_i} \right) = 0 \\ 3) & \bar{x}_i \geq 0 \end{array}\right\} \forall i = 1,..., n$$

$$\left.\begin{array}{ll} 4) & \dfrac{\partial l}{\partial u_j} = g_i(\bar{x}) - b_j \leq 0 \\ 5) & u_j \dfrac{\partial l}{\partial u_j} = u_j (g_i(\bar{x}) - b_j) = 0 \\ 6) & u_j \geq 0 \end{array}\right\} \forall i = 1,..., m$$

Donde la quinta relación es conocida como la condición complementaria Karush-Kuhn-Tucker

Ejemplo: Considere el siguiente problema:

$$\begin{array}{ll} \min f(x_1, x_2) = -\ln(x_1 + 1) - x_2 & \\ sa & 2x_1 + x_2 \leq 3 \\ & x_1 x_2 \geq 0 \end{array}$$

Solución: El langragiano del problema queda como:

$$L = -\ln(x_1 + 1) - x_2 + \mu_1(2x_1 + x_2 - 3)$$

Resolviendo las condiciones de KKT es posible obtener una solución óptima:

1a. $\dfrac{-1}{x_1+1}+2\mu_1 \geq 0$

1*b*. $-1+\mu_1 \geq 0$

2a. $x_1\left(\dfrac{-1}{x_1+1}+2\mu_1\right)=0$

2*b*. $x_1(-1+\mu_1)=0$

3. $2x_1+x_2 \leq 3$

4. $\mu_1(2x_1+x_2-3)=0$

5. $x_1 \geq 0, x_2 \geq 0$

6. $\mu_1 \geq 0$

De donde tenemos:

1. $\mu_1 \geq 1$ de la condición 1b.

2. $x_1 \geq 1$ de la condición 5.

3. Por lo tanto $\dfrac{-1}{x_1+1}+2\mu_1 > 0$

4. Por lo tanto $x_1 = 0$ de la condición 2a.

5. $\mu_1 \neq 0$ implica que $2x_1+x_2-3=0$ de la condición 4.

6. Los pasos 3 y 4 implican $x_2 = 3$

7. $x_2 \neq 0$ implica que $\mu_1 = 1$ de la condición 2b.

8. Los valores $x_1 = 0, x_2 = 3$ y $\mu_1 = 1$ no violan ninguna condición.

9. Por lo tanto $\bar{x} = (0,3)$ es solución óptima para este problema.

[A.3] La condición de Mercer

Un función kernel K puede ser expresado como

$K(u,v) = \Phi(u)\Phi(v)$

Si y solo si, para cualquier función *g(x)* tal que

$\int g(x)^2 dx$ es finita, entonces

$\int K(x,y)g(x)g(y)dxdy \geq 0$