(Data Mining concepts and Techniques [Han-3ed-2012])

(Chapter 9.3)

Máquina de Soporte Vectorial [9.3]

Data Scientist
Jhoan Esteban Ruiz Borja Msc

En esta sección, estudiamos **Máquinas de Vectores de Soporte (SVM)**, un método para la clasificación de datos lineales y no lineales. En pocas palabras, un SVM es un algoritmo que funciona de la siguiente manera. Utiliza un mapeo no lineal para transformar los datos de entrenamiento originales en una dimensión superior. Dentro de esta nueva dimensión, busca el hiperplano de separación óptima lineal (es decir, un "límite de decisión" que separa las tuplas de una clase de otra). Con un mapeo no lineal apropiado a una dimensión suficientemente alta, los datos de dos clases siempre pueden estar separados por un hiperplano. El SVM encuentra este hiperplano utilizando vectores de soporte (tuplas de entrenamiento "esenciales") y márgenes (definidos por los vectores de soporte). Más adelante profundizaremos en estos nuevos conceptos.

"He oído que los SVM han atraído una gran atención últimamente. ¿Por qué? "El primer documento sobre máquinas de vectores de soporte fue presentado en 1992 por Vladimir Vapnik y sus colegas Bernhard Boser e Isabelle Guyon, aunque la base para SVM ha existido desde la década de 1960 (incluidos los primeros trabajos de Vapnik y Alexei Chervonenkis sobre la teoría del aprendizaje estadístico). Aunque el tiempo de entrenamiento de incluso los SVM más rápidos puede ser extremadamente lento, son altamente precisos, debido a su capacidad para modelar límites de decisión no lineales complejos. Son mucho menos propensos a sobreajuste que otros métodos. Los vectores de soporte encontrados también proporcionan una descripción compacta del modelo aprendido. Los SVM se pueden utilizar para la predicción numérica y la clasificación. Se han aplicado a varias áreas, incluido el reconocimiento de dígitos escritos a mano, el reconocimiento de objetos y la identificación de los hablantes, así como también pruebas de predicción de series de tiempo de referencia.

El caso cuando los datos son linealmente separables [9.3.1]

Para explicar el misterio de las SVM, veamos primero el caso más simple: un problema de dos clases donde las clases son linealmente separables. Permita que el conjunto de datos D se proporcione como $(X_1, y_1), (X_2, y_2), \ldots, (X_{|D|}, y_{|D|})$, donde X_i es el conjunto de tuplas de entrenamiento con etiquetas de clase asociadas, y_i. Cada y_i puede tomar uno de dos valores, $+1$ o -1 (es decir, $y_i \in \{+1, -1\}$), correspondiente a las clases compra la computadora $\boldsymbol{buys\_computer = yes}$ y $\boldsymbol{buys\_computer = no}$, respectivamente. Para ayudar en la visualización, consideremos un ejemplo basado en dos atributos de entrada, A_1 y A_2, como se muestra en la Figura 9.7. Desde el gráfico, vemos que los datos bidimensionales son

linealmente separables (o "lineales", para abreviar), porque se puede dibujar una línea recta para separar todas las tuplas de la clase $+1$ de todas las tuplas de la clase -1.

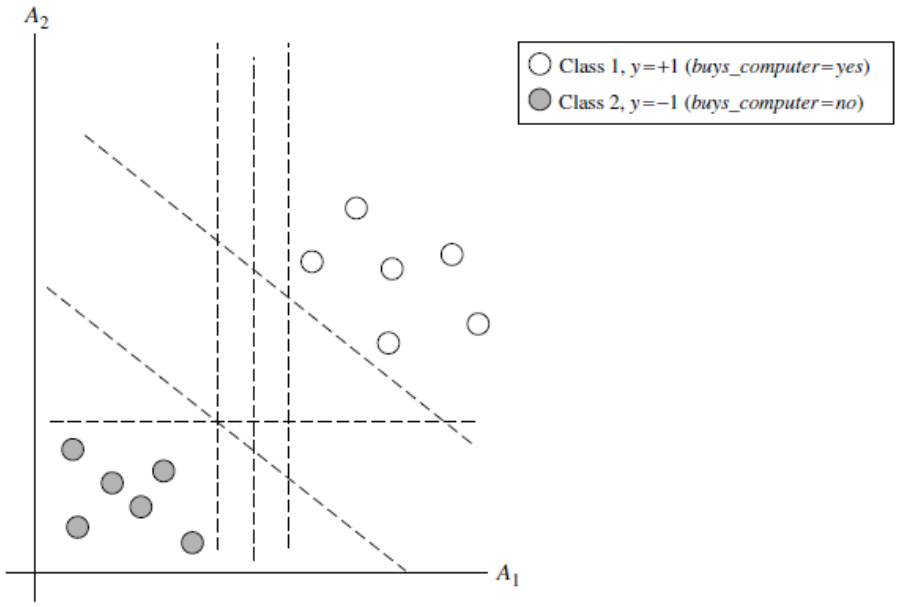

Figure 9.7 The 2-D training data are linearly separable. There are an infinite number of possible separating hyperplanes or "decision boundaries," some of which are shown here as dashed lines. Which one is best?

Hay un número infinito de líneas de separación que se pueden dibujar. Queremos encontrar el "mejor", es decir, uno que (esperamos) tendrá el mínimo error de clasificación en las tuplas no vistos anteriormente. ¿Cómo podemos encontrar esta mejor línea? Tenga en cuenta que si nuestros datos fueran 3-D (es decir, con tres atributos), nos gustaría encontrar el mejor plano de separación. Generalizando a n dimensiones, queremos encontrar el mejor hiperplano. Utilizaremos "hiperplano" para referirnos al límite de decisión que estamos buscando, independientemente del número de atributos de entrada. Entonces, en otras palabras, ¿cómo podemos encontrar el mejor hiperplano?

Una SVM aborda este problema buscando el **hiperplano marginal máximo**. Considere la Figura 9.8, que muestra dos posibles hiperplanos de separación y sus márgenes asociados. Antes de entrar en la definición de los márgenes, echemos un vistazo intuitivo a esta figura. Ambos hiperplanos pueden clasificar correctamente todas las tuplas de datos dadas. Intuitivamente, sin embargo, esperamos que el hiperplano con el margen más grande sea más preciso en la clasificación de las tuplas de datos futuros que el hiperplano con el margen más pequeño. Esta es la razón por la que (durante la fase de aprendizaje o capacitación) SVM busca el hiperplano con el margen más grande, es decir, el hiperplano marginal máximo [máximum marginal hyperplane (MMH)]. El margen asociado proporciona la mayor separación entre clases.

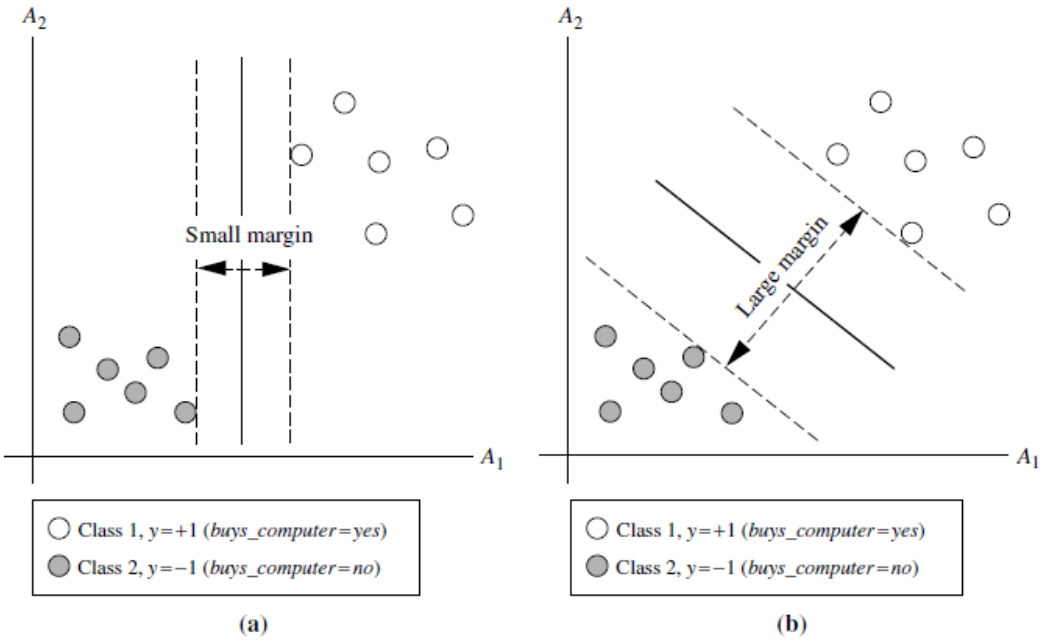

Figure 9.8 Here we see just two possible separating hyperplanes and their associated margins. Which one is better? The one with the larger margin (b) should have greater generalization accuracy.

Al llegar a una definición informal de margen, podemos decir que la distancia más corta desde un hiperplano a un lado de su margen es igual a la distancia más corta desde el hiperplano al otro lado de su margen, donde los "lados" del margen son paralelo al hiperplano. Al tratar con el MMH, esta distancia es, de hecho, la distancia más corta desde el MMH a la tupla de entrenamiento más cercana de cualquier clase.

Un hiperplano de separación se puede escribir como

$$WX + b = 0 \qquad (9.12)$$

donde W es un vector de peso, a saber, $W = \{w_1, w_2, .., w_n\}$; n es el número de atributos; y b es un escalar, a menudo referido como un sesgo. Para ayudar en la visualización, consideremos dos atributos de entrada, A_1 y A_2, como en la figura 9.8 (b). Las tuplas de entrenamiento son 2-D (por ejemplo, $X = (x_1, x_2)$, donde x_1 y x_2 son los valores de los atributos A_1 y A_2, respectivamente, para X. Si pensamos en b como un peso adicional, w_0, podemos reescribir Eq. (9.12) como

$$w_0 + w_1 x_1 + w_2 x_2 = 0 \qquad (9.13)$$

Por lo tanto, cualquier punto que se encuentra por encima del hiperplano de separación satisface

$$w_0 + w_1 x_1 + w_2 x_2 > 0 \qquad (9.14)$$

Del mismo modo, cualquier punto que se encuentra debajo del hiperplano de separación satisface

$$w_0 + w_1 x_1 + w_2 x_2 < 0 \qquad (9.15)$$

Los pesos se pueden ajustar para que los hiperplanos que definen los "lados" del margen se puedan escribir como

$$H_1: \; w_0 + w_1 x_1 + w_2 x_2 \geq 1 \quad para \quad y_i = +1 \qquad (9.16)$$

$$H_2: \; w_0 + w_1 x_1 + w_2 x_2 \leq 1 \quad para \quad y_i = -1 \qquad (9.17)$$

Es decir, cualquier tupla que caiga sobre o sobre H_1 pertenece a la clase $+1$, y cualquier tupla que caiga sobre H_2 o debajo de ella pertenece a la clase -1. Combinando las dos desigualdades de las Ecs. (9.16) y (9.17), obtenemos

$$y_i(w_0 + w_1 x_1 + w_2 x_2) \geq 1, \quad \forall i \qquad (9.18)$$

Cualquier tupla de entrenamiento que caiga en los hiperplanos H_1 o H_2 (es decir, los "lados" que definen el margen) satisface Eq. (9.18) y se llaman vectores de soporte. Es decir, están igualmente cerca del MMH (separador). En la Figura 9.9, los vectores de soporte se muestran rodeados con un borde más grueso. Básicamente, los vectores de soporte son las tuplas más difíciles de clasificar y brindan la mayor información sobre la clasificación.

A partir de esto, podemos obtener una fórmula para el tamaño del margen máximo. La distancia desde el hiperplano de separación a cualquier punto en H_1 es $\frac{1}{\|W\|}$, donde $\|W\|$ es la norma euclidiana de W, es decir, $\sqrt{W * W}$. Por definición, esto es igual a la distancia desde cualquier punto en H_2 hasta el hiperplano de separación . Por lo tanto, el margen máximo es $\frac{2}{\|W\|}$.

Nota: Si $W = \{w_1, w_2, .., w_n\}$, entonces $\sqrt{w_1^2 + w_2^2 + , .., + w_n^2}$

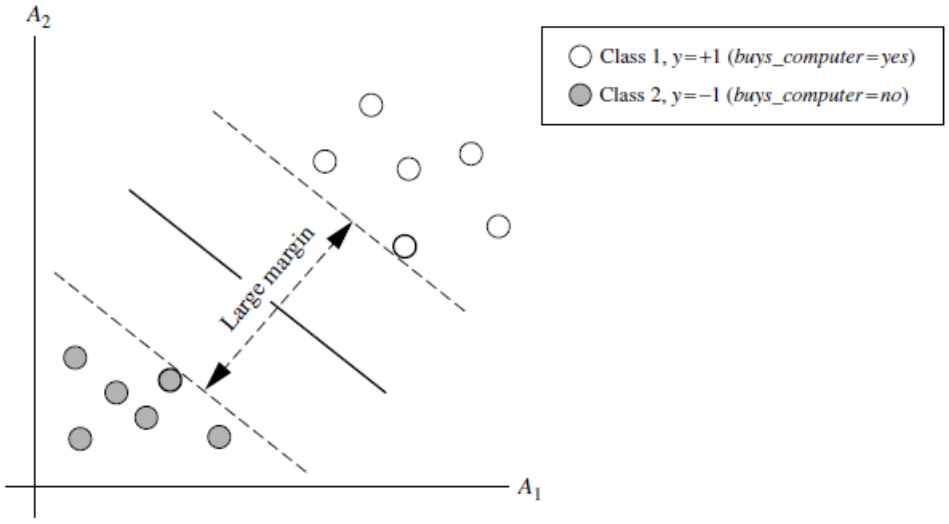

Figure 9.9 Support vectors. The SVM finds the maximum separating hyperplane, that is, the one with maximum distance between the nearest training tuples. The support vectors are shown with a thicker border.

"Entonces, ¿cómo una SVM encuentra el MMH y los vectores de soporte?" Usando algunos "trucos de matemáticas sofisticados", podemos reescribir la ecuación. (9.18) para que se convierta en lo que se conoce como un problema de optimización cuadrático restringido (convexo). Tales trucos de matemática van más allá del alcance de este libro. Los lectores avanzados pueden estar interesados en observar que los trucos implican volver a escribir Eq. (9.18) usando una formulación lagrangiana y luego resolviendo la solución usando las condiciones de Karush-Kuhn-Tucker (KKT). Los detalles se pueden encontrar en las notas bibliográficas al final de este capítulo (Sección 9.10).

Si los datos son pequeños (digamos, menos de 2000 tuplas de entrenamiento), cualquier paquete de software de optimización para resolver problemas cuadráticos convexos restringidos se puede usar para encontrar los vectores de soporte y MMH. Para datos más grandes, se pueden usar algoritmos especiales y más eficientes para el entrenamiento de SVM, cuyos detalles exceden el alcance de este libro. Una vez que hemos encontrado los vectores de soporte y MMH (¡tenga en cuenta que los vectores de soporte definen el MMH!), Tenemos una máquina de vectores de soporte entrenada. El MMH es un límite de clase lineal, por lo que el SVM correspondiente se puede usar para clasificar datos linealmente separables. Nos referimos a una SVM tan entrenada como una SVM lineal.

"Una vez que tengo una máquina de vectores de soporte entrenada, ¿cómo la uso para clasificar las tuplas de prueba (es decir, nuevas)? Según la formulación de Lagrange mencionada anteriormente, el MMH puede reescribirse como el límite de decisión

$$d(X^T) = \sum_{i=1}^{l} y_i \alpha_i X_i X^T + b_0 \qquad (9.19)$$

donde y_i es la etiqueta de clase del vector de soporte X_i; X^T es una tupla de prueba; α_i y b_0 son parámetros numéricos que fueron determinados automáticamente por la optimización o el algoritmo SVM anotado anteriormente; y l es la cantidad de vectores de soporte.

Los lectores interesados pueden notar que α_i son multiplicadores lagrangianos. Para los datos linealmente separables, los vectores de soporte son un subconjunto de las tuplas de entrenamiento reales (aunque habrá un ligero giro con respecto a esto cuando se trata de datos no lineales separables, como veremos a continuación).

Dada una tupla de prueba, X^T, la conectamos a la ecuación. (9.19), y luego verifique para ver el signo del resultado. Esto nos dice en qué lado del hiperplano cae la tupla de prueba. Si el signo es positivo, entonces X^T cae en o encima del MMH, por lo que el SVM predice que X^T pertenece a la clase $+1$ (representando compra de la computadora **buys_computer = yes**, en nuestro caso). Si el signo es negativo, entonces X^T cae sobre o debajo del MMH y la predicción de clase es -1 (lo que representa compra de la computadora **buys_computer = no**).

Observe que la formulación lagrangiana de nuestro problema (ecuación 9.19) contiene un producto escalar entre el vector de soporte X_i y la prueba de tupla X^T. Esto resultará muy útil

para encontrar el MMH y los vectores de soporte para el caso cuando los datos dados sean separables de forma no lineal, como se describe más adelante en la próxima sección.

Antes de pasar al caso no lineal, hay dos cosas más importantes a tener en cuenta. La complejidad del clasificador aprendido se caracteriza por el número de vectores de soporte en lugar de la dimensionalidad de los datos. Por lo tanto, las SVM tienden a ser menos propensas a sobreajuste que algunos otros métodos. Los vectores de soporte son las tuplas de entrenamiento esenciales o críticas: se encuentran más cerca del límite de decisión (MMH). Si se eliminaran todas las otras tuplas de entrenamiento y se repitiera el entrenamiento, se encontraría el mismo hiperplano de separación. Además, la cantidad de vectores de soporte encontrados se puede usar para calcular un límite (superior) en la tasa de error esperada del clasificador SVM, que es independiente de la dimensionalidad de los datos. Una SVM con un pequeño número de vectores de soporte puede tener una buena generalización, incluso cuando la dimensionalidad de los datos es alta.

El caso cuando los datos son linealmente inseparables [9.3.2]

En la Sección 9.3.1 aprendimos acerca de las SVM lineales para clasificar datos linealmente separables, pero ¿y si los datos no son separables linealmente, como en la Figura 9.10? En tales casos, no se puede encontrar una línea recta que separaría las clases. Las SVM lineales que estudiamos no podrían encontrar una solución factible aquí. ¿Ahora qué?

La buena noticia es que el enfoque descrito para SVM lineales puede ampliarse para crear SVM no lineales para la clasificación de datos linealmente inseparables (también llamados datos no lineales separables, o datos no lineales para abreviar). Tales SVM son capaces de encontrar límites de decisión no lineales (es decir, hipersuperficies no lineales) en el espacio de entrada.

"Entonces," puede preguntar, "¿cómo podemos extender el enfoque lineal?" Obtenemos una SVM no lineal ampliando el enfoque para SVM lineales de la siguiente manera. Hay dos pasos principales. En el primer paso, transformamos los datos de entrada originales en un espacio dimensional superior utilizando un mapeo no lineal. Se pueden usar varias asignaciones no lineales comunes en este paso, como describiremos más adelante. Una vez que los datos se han transformado en el nuevo espacio superior, el segundo paso busca un hiperplano de separación lineal en el nuevo espacio. De nuevo terminamos con un problema de optimización cuadrática que puede resolverse usando la formulación lineal de SVM. El hiperplano marginal máximo encontrado en el nuevo espacio corresponde a una hipersuperficie de separación no lineal en el espacio original.

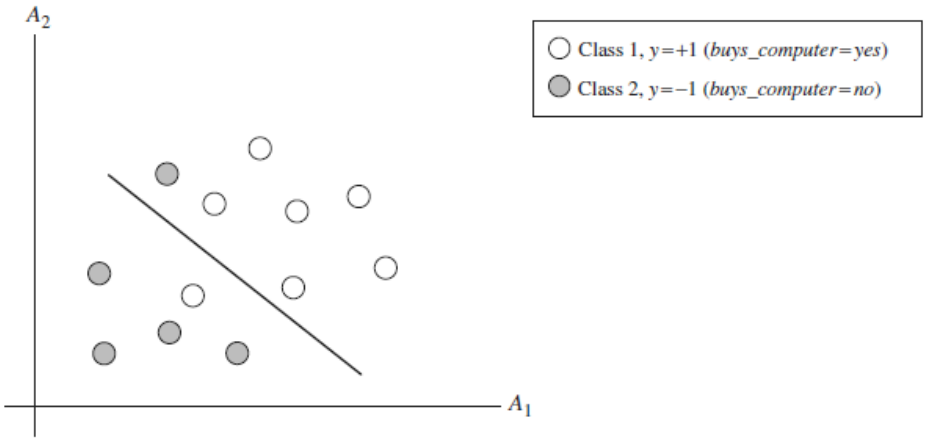

Figure 9.10 A simple 2-D case showing linearly inseparable data. Unlike the linear separable data of Figure 9.7, here it is not possible to draw a straight line to separate the classes. Instead, the decision boundary is nonlinear.

Ejemplo 9.2 Transformación no lineal de datos de entrada originales en un espacio dimensional superior. Considera el siguiente ejemplo. Un vector de entrada 3-D $X = (x_1, x_2, x_3)$ se mapea en un espacio 6-D, Z, usando las asignaciones $\phi_1(X) = x_1$, $\phi_2(X) = x_2$, $\phi_3(X) = x_3$, $\phi_4(X) = (x_1)^2$, $\phi_5(X) = x_1x_2$ y $\phi_6(X) = x_1x_3$. Un hiperplano de decisión en el nuevo espacio es $d(Z) = WZ + b$, donde W y Z son vectores. Esto es lineal. Resolvemos para W y b y luego sustituimos de nuevo para que el hiperplano de decisión lineal en el espacio nuevo (Z) corresponda a un polinomio de segundo orden no lineal en el espacio de entrada 3-D original:

$$d(Z) = w_1x_1 + w_2x_2 + w_3x_3 + w_4(x_1)^2 + w_5x_1x_2 + w_6x_1x_3 + b$$

$$w_1z_1 + w_2z_2 + w_3z_3 + w_4z_4 + w_5z_5 + w_6z_6 + b$$

Pero hay algunos problemas. Primero, ¿cómo elegimos el mapeo no lineal a un espacio dimensional más alto? Segundo, el cálculo involucrado será costoso. Consulte Eq. (9.19) para la clasificación de una tupla de prueba, X^T. Dada la tupla de prueba, tenemos que calcular su producto de puntos con cada uno de los vectores de soporte. En el entrenamiento, tenemos que calcular un producto de punto similar varias veces para encontrar el MMH. Esto es especialmente caro. Por lo tanto, el cálculo del producto de punto requerido es muy pesado y costoso. ¡Necesitamos otro truco!

Afortunadamente, podemos usar otro truco matemático. Ocurre que al resolver el problema de optimización cuadrática de la SVM lineal (es decir, cuando se busca una SVM lineal en el nuevo espacio dimensional superior), las tuplas de entrenamiento aparecen solo en forma de productos de puntos, $\phi(X_i) \cdot \phi(X_j)$, donde $\phi(X)$ es simplemente la función de mapeo no lineal aplicada para transformar las tuplas de entrenamiento. En lugar de calcular el producto escalar en las tuplas de datos transformadas, resulta matemáticamente equivalente aplicar una función de kernel, $K(X_i, X_j)$, a los datos de entrada originales. Es decir,

$$K(X_i, X_j) = \phi(X_i) \cdot \phi(X_j) \qquad (9.20)$$

En otras palabras, en todas partes que $\phi(X_i) \cdot \phi(X_j)$ aparece en el algoritmo de entrenamiento, podemos reemplazarlo por $K(X_i, X_j)$. De esta forma, todos los cálculos se realizan en el espacio de entrada original, que es potencialmente de una menor dimensionalidad. Podemos evitar con seguridad el mapeo, ¡resulta que ni siquiera tenemos que saber qué es el mapeo! Más adelante hablaremos sobre qué tipos de funciones se pueden usar como funciones del kernel para este problema.

Después de aplicar este truco, podemos proceder a buscar un hiperplano de separación máximo. El procedimiento es similar al descrito en la Sección 9.3.1, aunque implica colocar un límite superior especificado por el usuario, C, en los multiplicadores de Lagrange, α_i. Este límite superior se determina mejor experimentalmente.

"¿Cuáles son algunas de las funciones del kernel que podrían usarse?" Se han estudiado las propiedades de los tipos de funciones del kernel que podrían usarse para reemplazar el escenario del producto escalar que se acaba de describir. Tres funciones admisibles del kernel son

Núcleo polinómico de grado h: $K(X_i, X_j) = (X_i \cdot X_j + 1)^h$

Núcleo de la función de base radial de Gauss: $K(X_i, X_j) = e^{-\frac{\|X_i - X_j\|^2}{2\sigma^2}}$

Kernel de Sigmoide: $K(X_i, X_j) = \tanh(kX_i \cdot X_j - \delta)$

Cada uno de estos resultados en un clasificador no lineal diferente en el espacio de entrada (el original). Los aficionados a la red neuronal estarán interesados en observar que los hiperplanos de decisión resultantes encontrados para las SVM no lineales son del mismo tipo que los encontrados por otros clasificadores de redes neuronales bien conocidos. Por ejemplo, una SVM con una función de base radial gaussiana (RBF) da la misma hiperplanea de decisión como un tipo de red neuronal conocida como una red de función de base radial. Una SVM con un núcleo sigmoide es equivalente a una red neuronal simple de dos capas conocida como perceptrón multicapa (sin capas ocultas).

No existen reglas de oro para determinar qué núcleo admisible dará como resultado la SVM más precisa. En la práctica, el núcleo elegido generalmente no hace una gran diferencia en la precisión resultante. El entrenamiento de SVM siempre encuentra una solución global, a diferencia de las redes neuronales, como la retropropagación, donde generalmente existen muchos mínimos locales (Sección 9.2.3).

Hasta ahora, hemos descrito SVM lineales y no lineales para la clasificación binaria (es decir, de dos clases). Los clasificadores SVM se pueden combinar para el caso multiclase. Vea la Sección 9.7.1 para algunas estrategias, como entrenar un clasificador por clase y el uso de códigos de corrección de errores.

Un objetivo de investigación importante con respecto a SVM es mejorar la velocidad en el entrenamiento y la prueba para que las SVM se conviertan en una opción más factible para

conjuntos de datos muy grandes (por ejemplo, millones de vectores de soporte). Otros problemas incluyen determinar el mejor kernel para un conjunto de datos dado y encontrar métodos más eficientes para el caso multiclase.