UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FISICAS Y MATEMATICAS
DEPARTAMENTO DE INGENIERIA INDUSTRIAL

MODELOS DE SELECCIÓN DE ATRIBUTOS PARA SUPPORT VECTOR MACHINES

TESIS POR COMPENDIO DE PUBLICACIONES PARA OPTAR AL GRADO DE DOCTOR EN SISTEMAS DE INGENIERÍA

SEBASTIÁN ALEJANDRO MALDONADO ALARCÓN

SANTIAGO DE CHILE

JUNIO, 2011

UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FISICAS Y MATEMATICAS
DEPARTAMENTO DE INGENIERIA INDUSTRIAL

MODELOS DE SELECCIÓN DE ATRIBUTOS PARA SUPPORT VECTOR MACHINES

TESIS POR COMPENDIO DE PUBLICACIONES PARA OPTAR AL GRADO DE DOCTOR EN SISTEMAS DE INGENIERÍA

SEBASTIÁN ALEJANDRO MALDONADO ALARCÓN

PROFESOR GUIA:
RICHARD WEBER H.

MIEMBROS DE LA COMISIÓN:
EMILIO CARRIZOSA P.
RAÚL GOUET B.
RICARDO MONTOYA M.
ÁLVARO SOTO A.

SANTIAGO DE CHILE

JUNIO, 2011

A mis padres, mis amigos, mi amor

Agradecimientos

Me gustaría expresar mi sincero agradecimiento y afecto al profesor Richard Weber, principal responsable de que este trabajo doctoral haya salido adelante de forma satisfactoria. Durante estos seis años de trabajo conjunto desde mi tesis para el grado de Magíster, he tenido la oportunidad de crecer tanto profesionalmente como a nivel personal, gracias a su larga experiencia y amplio conocimiento. No dudo que la relación profesional y personal que nos une continuará por muchos años. Ha sido un placer y un privilegio trabajar a su lado. También me gustaría agradecer a los miembros de la comisión de tesis y colaboradores externos, en especial a Jayanta Basak, por sus importantes comentarios y cooperación en el desarrollo de las publicaciones.

Asimismo, no puedo dejar pasar la oportunidad de dar las gracias a mis compañeros y amigos, Cristián Bravo y Gastón L'huillier. Ambos han participado de una u otra forma en la realización de los trabajos que conforman esta tesis, es por ello que quisiera reconocer la enorme importancia que ha tenido para mí su colaboración y apoyo.

Además, para que un proyecto académico o profesional tenga éxito no es sólo necesario tener buenos compañeros de trabajo, es también fundamental contar con el apoyo de nuestros seres más queridos. Agradezco el infinito apoyo de mis padres, por su comprensión y consejos. Agradezco de forma especial a mi novia Luz María, por su enorme cariño desinteresado e incondicional.

Finalmente, comentar que la realización de esta tesis doctoral ha sido posible gracias a la concesión al autor de la misma de una beca para estudios doctorales por parte del Gobierno de Chile, a través de la Comisión Nacional de Investigación Científica y Tecnológica (CONICYT). Igualmente agradezco el financiamiento del proyecto FONDECYT 1040926, concedido al Dr. Richard Weber, y el apoyo constante del Instituto Milenio Sistemas de Ingeniería. Agradezco a todos los miembros de este instituto y a las personas que apoyan el Doctorado en Sistemas de Ingeniería, en especial a Julie Lagos y Fernanda Melis por toda la ayuda y apoyo brindado.

Resumen Ejecutivo

Recientemente los datos se han incrementado en todas las áreas del conocimiento, tanto en el número de instancias como en el de atributos. Bases de datos actuales pueden contar con decenas e incluso cientos de miles de variables con un alto grado de información tanto irrelevante como redundante. Esta gran cantidad de datos causa serios problemas a muchos algoritmos de minería de datos en términos de escalabilidad y rendimiento. Dentro de las áreas de investigación en selección de atributos se incluyen el análisis de chips de ADN, procesamiento de documentos provenientes de internet y modelos de administración de riesgo en el sector financiero. El objetivo de esta tarea es triple: mejorar el desempeño predictivo de los modelos, implementar soluciones más rápidas y menos costosas, y proveer de un mejor entendimiento del proceso subyacente que generó los datos.

Dentro de las técnicas de minería de datos, el método llamado *Support Vector Machines* (SVMs) ha ganado popularidad gracias a su capacidad de generalización frente a nuevos objetos y de construir complejas funciones no lineales. Estas características permiten obtener mejores resultados que otros métodos predictivos. Sin embargo, una limitación de este método es que no está diseñado para identificar los atributos importantes para construir la regla discriminante. El presente trabajo tiene como objetivo desarrollar técnicas que permitan incorporar la selección de atributos en la formulación de SVMs no lineal, aportando eficiencia y comprensibilidad al método. Se desarrollaron dos metodologías: un algoritmo *wrapper* (HO-SVM) que utiliza el número de errores en un conjunto de validación como medida para decidir qué atributo eliminar en cada iteración, y un método *embedded* (KP-SVM) que optimiza la forma de un kernel Gaussiano no isotrópico, penalizando la utilización de atributos en la función de clasificación.

Los algoritmos propuestos fueron probados en bases de datos de de diversa dimensionalidad, que van desde decenas a miles de atributos, y en problemas reales de asignación de créditos para entidades financieras nacionales. De los resultados se obtiene que SVMs no lineal con kernel Gaussiano muestra un mejor desempeño que con las funciones de kernel lineal y polinomial. Asimismo, los métodos de selección de atributos propuestos permiten mantener o incluso mejorar el desempeño predictivo de SVMs no lineal, logrando además una reducción significativa en la utilización de atributos. Para las bases de mayor dimensionalidad se reduce de miles a decenas de atributos seleccionados, logrando un desempeño predictivo significativamente mejor que los enfoques alternativos de selección de atributos para SVMs. Se concluye que los enfoques presentados representan la alternativa más efectiva dentro de las estudiadas para resolver el problema de selección de atributos en modelos de aprendizaje computacional. Como trabajo futuro se propone adaptar las metodologías propuestas para problemas con desbalance de clases, donde se requiere una evaluación distinta del desempeño del modelo considerando costos por error de clasificación asimétricos, una problemática común en aplicaciones como detección de fuga y riesgo crediticio.

Índice general

1 Introducción

1.1. Planteamiento

En el escenario actual, las empresas participan en un mercado muy competitivo, donde los clientes se encuentran adecuadamente informados al momento de elegir entre distintas compañías. En mercados donde esto ocurre, la empresa que posea una mayor cantidad de información relevante podrá ejecutar estrategias comerciales efectivas, sobresaliendo del resto de las compañías. Adicionalmente, la información disponible permite tomar diversas decisiones estratégicas, tales como: definir políticas de asignación de créditos en base al comportamiento histórico de clientes, diseño de nuevos productos a partir de preferencias declaradas, definir campañas que eviten que los clientes se fuguen de la empresa, etc.

Si bien obtener información potencialmente útil es cada vez más simple, gracias al importante aumento de la capacidad de almacenaje y la disponibilidad de mejores herramientas para el manejo de datos, el proceso de extracción de información relevante a partir de los datos disponibles sigue siendo complejo y costoso.

Actualmente existen técnicas que permiten analizar patrones de conducta, nichos de mercado, y muchos otros tipos de información no trivial mediante la utilización de sofisticados modelos que combinan métodos estadísticos, aprendizaje de máquinas y optimización. Estas técnicas se engloban bajo el concepto de *minería de datos* (*data mining*) [11]. La investigación en estos modelos ha sido un tema relevante en estas últimas dos décadas, habiéndose han logrado avances significativos en términos de eficiencia y desempeño predictivo [35].

1.1.1. Aprendizaje Supervisado y Proceso KDD

El *aprendizaje automático* es una rama de la inteligencia artificial cuyo objetivo es desarrollar técnicas que permitan a las computadoras *aprender* [26]. De forma más concreta, se trata de crear modelos capaces de generalizar comportamientos a partir de información suministrada en forma de ejemplos. Es, por lo tanto, un proceso de inducción del conocimiento.

Dentro del aprendizaje automático se distinguen dos ramas: el *aprendizaje supervisado*, que permite construir una función a partir de los datos de entrenamiento, los cuales consisten en pares de objetos con los datos de entrada y los resultados deseados. La salida de la función puede ser un valor numérico (como en los problemas de regresión) o una etiqueta de clase (como en los de clasificación). El objetivo del aprendizaje supervisado es crear una función capaz de predecir el valor correspondiente a cualquier objeto de entrada después de haber visto una serie de ejemplos [26]. El *aprendizaje no supervisado*, en cambio, se distingue del aprendizaje supervisado por el hecho de que no hay un conocimiento *a priori*, y su objetivo es describir ciertas características del conjunto de datos de entrada, y entender cómo éstos se encuentran organizados. En este trabajo el aprendizaje será entendido como supervisado, y se verán tanto problemas de clasificación como de regresión.

Para obtener conclusiones válidas y útiles al aplicar minería de datos, es necesario complementar este concepto con una adecuada preparación de los datos previa al proceso de minería y un análisis posterior de los resultados obtenidos. Así, es posible afirmar que la de minería de datos pertenece a un proceso más amplio, reflejado en la Figura 1.1, denominado extracción o descubrimiento de conocimiento en bases de datos (KDD, *Knowledge Discovery in Databases* [10]). El KDD es un campo multidisciplinario, donde las principales áreas contribuyentes son el aprendizaje automático, las bases de datos y la estadística.

El proceso KDD consta de los siguientes pasos [10, 35]:

- **Recopilación y consolidación de datos:** Antes de utilizar los algoritmos de minería de datos, el conjunto de datos objetivo debe construirse. Dado que el objetivo es revelar patrones ocultos presentes en los datos, este conjunto debe ser suficientemente grande para contener estos

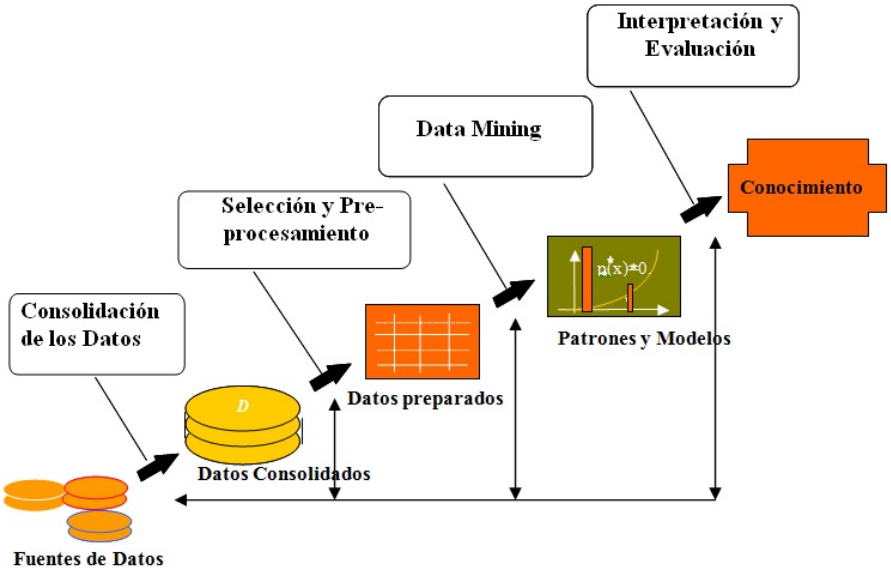

Figura 1.1: Proceso KDD

patrones, pero a la vez suficientemente conciso para ser minado en un tiempo aceptable. Fuentes comunes de datos son bases de datos transaccionales, *data marts* y *data warehouses*.

- **Pre-procesamiento de datos:** La utilidad de la extracción de información de los datos depende en gran medida de la calidad de éstos. El propósito fundamental de esta etapa es el de manipular y transformar los datos en bruto, de manera que la información contenida en el conjunto de datos pueda ser descubierta, o hacer más fácilmente accesible [31]. La lista de tareas que se incluyen en esta fase se puede resumir en tres: limpieza de datos (eliminación de inconsistencias y valores perdidos), transformación (proceso de adecuar los datos al posterior proceso de aprendizaje, con el fin de mejorar su capacidad predictiva) y reducción (eliminación de ejemplos o atributos que no sean relevantes para el problema).

- **Minado de datos:** Esta etapa incluye comúnmente tareas asociadas al aprendizaje supervisado (clasificación y regresión), aprendizaje no supervisado (segmentación) y aprendizaje de reglas de asociación (búsqueda de relaciones entre variables, por ejemplo, hábitos de compra de clientes en supermercados)[11].

- **Evaluación de los resultados:**

Partiendo de la necesidad de evaluar el desempeño de los modelos predictivos, se dispone de diversas estrategias de validación de un sistema de aprendizaje dependiendo de cómo se haga la partición del conjunto. El método de evaluación más básico, la validación simple, utiliza un conjunto de muestras para construir el modelo del clasificador, y otro diferente para estimar el error, con el fin de eliminar el efecto del sobreajuste en los datos (*overfitting*). Entre la variedad de particiones posibles, una de las más frecuentes es tomar aproximadamente dos tercios de las muestras para el proceso de aprendizaje y el tercio restante para comprobar el error del clasificador (método conocido como *holdout* [49]). El hecho de que sólo se utiliza una parte de las muestras disponibles para llevar a cabo el aprendizaje es el inconveniente principal de esta técnica, al considerar que se pierde información útil en el proceso de inducción del clasificador. Esta situación se agrava si el número de muestras para construir el modelo es muy reducido, ya sea por el porcentaje elegido, o por no disponer de más datos.

Otra técnica de evaluación denominada validación cruzada, se plantea para evitar la ocultación de parte de las muestras al algoritmo de inducción y la consiguiente pérdida de información. Con esta técnica el conjunto de datos se particiona en k partes mutuamente exclusivas, conteniendo cada una un número similar de ejemplos, indicándose en muchos casos como validación cruzada con k partes (*k-fold crossvalidation* [49]). En cada evaluación, se deja uno de los subconjuntos para la prueba, y se entrena el sistema con los $k-1$ restantes. Así, la precisión estimada es la media del desempeño de los k subconjuntos de prueba. La ventaja de la validación cruzada (a diferencia del método *holdout*) es que todos los casos son utilizados en el proceso de aprendizaje y en el de prueba, dando lugar a un estimador con sesgo muy pequeño. Un caso particular de este método de evaluación es la validación cruzada dejando una instancia fuera (*leave-one-out crossvalidation*), donde k es igual al número de ejemplos del conjunto de datos.

1.1.2. Support Vector Machines

Dentro de las técnicas de aprendizaje para la minería de datos, el método conocido como *Support Vector Machines* (SVMs) se ha popularizado gracias a su capacidad de generalización ante nuevos objetos y de construir complejas funciones no lineales [47]. Estas ventajas pueden llevar a

una mejor predicción que otros métodos predictivos tanto para las tareas de clasificación como de regresión [47].

En esta sección se describe la derivación matemática de SVMs como técnica de clasificación y su extensión a regresión. Se comenzará con la descripción del enfoque clásico conocido como Minimización del Riesgo Empírico (ERM), para luego presentar el concepto de la Minimización del Riesgo Estructural (SRM), el cual es implementado por SVM.

Minimización del Riesgo Empírico vs. Minimización del Riesgo Estructural

Para el caso de reconocimiento de patrones en clasificación binaria, la tarea de aprender mediante ejemplos puede formularse de la siguiente manera: dado un conjunto de funciones $\{f_\lambda(\mathbf{x}) : \lambda \in \Lambda\}$, $f_\lambda : \Re^N \to \{-1,1\}$, donde Λ es un conjunto de parámetros, y un conjunto de ejemplos $(\mathbf{x}_1,y_1),...,(\mathbf{x}_m,y_m)$, $\mathbf{x}_i \in \Re^N, y_i \in \{-1,1\}$, obtenidos a partir de una distribución desconocida $P(\mathbf{x},y)$, se desea encontrar la función f_{λ^*} que entregue el menor *riesgo esperado*:

$$R(\lambda) = \int |f_\lambda(\mathbf{x}) - y| P(\mathbf{x},y) d\mathbf{x} dy \tag{1.1}$$

Las funciones f_λ se conocen como *hipótesis*, mientras que el conjunto $\{f_\lambda(\mathbf{x}) : \lambda \in \Lambda\}$ se conoce como *espacio de hipótesis* y se denota como H. El conjunto de funciones f_λ pueden ser, por ejemplo, una red *Multi-Layer Perceptron* con un cierto número de unidades ocultas. En este caso, el conjunto Λ representa los pesos de la red neuronal [29]. El riesgo esperado corresponde a medir qué tan buena es una hipótesis para predecir la etiqueta correcta y_i para un ejemplo $\mathbf{x}_i$. Dado que la distribución de probabilidades $P(\mathbf{x},y)$ es desconocida, no es posible calcular (y, por lo tanto, minimizar) el riesgo esperado $R(\lambda)$. Sin embargo, dado que se cuenta con una muestra de $P(\mathbf{x},y)$, es posible calcular una aproximación estocástica de $R(\lambda)$, llamada *riesgo empírico*:

$$R_{emp}(\lambda) = \frac{1}{m} \sum_{i=1}^{m} |f_\lambda(\mathbf{x}_i) - y_i| \tag{1.2}$$

Dado que la ley de los grandes números garantiza que el riesgo empírico converge en probabilidad al riesgo esperado, un enfoque común consiste en minimizar el riesgo empírico en vez del riesgo esperado. La intuición detrás de este enfoque (el *principio de minimización del riesgo empírico* o ERM) es que si R_{emp} converge a R, el mínimo de R_{emp} convergería al mínimo de R. Si esta convergencia no se cumple, el principio de minimización del riesgo empírico no permitiría hacer inferencia alguna basada en el conjunto de datos, y por lo tanto se consideraría no consistente. Vapnik y Chervonenkis [46, 48] muestran que la condición necesaria y suficiente para la consistencia del Principio de Minimización del Riesgo Empírico es que la *dimensión VC h* del espacio de hipótesis sea finita. La dimensión VC es un número natural, posiblemente infinito, el cual representa el mayor número de observaciones que pueden ser separadas de todas las posibles maneras por un conjunto de funciones f_λ. La dimensión VC es una medida de la complejidad del espacio de hipótesis, y es muchas veces proporcional al número de parámetros de la función clasificadora f_λ.

La teoría de convergencia uniforme en probabilidad desarrollada por Vapnik y Chervonenkis también provee cotas para la desviación del riesgo empírico con respecto al riesgo esperado. Una típica cota uniforme de Vapnik y Chervonenkis, la cual se obtiene con probabilidad $1-\eta$, toma la siguiente forma [29, 48]:

$$R(\lambda) \leq R_{emp}(\lambda) + \sqrt{\frac{h(ln\frac{2m}{h}+1) - ln\frac{\eta}{4}}{m}} \qquad \forall \lambda \in \Lambda \tag{1.3}$$

donde h es la dimensión VC de f_λ. A partir de esta cota es claro que tanto el riesgo empírico como el *ratio* entre la dimensión VC y el número de observaciones debe ser pequeño. Dado que el riesgo empírico es usualmente decreciente con respecto a h, se tiene que, para un número fijo de ejemplos, existe un valor óptimo de la dimensión VC. La elección de un valor apropiado de h es crucial para obtener un buen desempeño, especialmente cuando el número de ejemplos es pequeño. Este problema es similar a encontrar el número apropiado de unidades ocultas en una red MLP.

La cota (1.3) sugiere que el Principio de Minimización del Riesgo Empírico puede ser reemplazado por un mejor principio inductivo. Vapnik [46] sugiere la *Minimización del Riesgo Estructural* (SRM) como un intento para resolver el problema de elegir la dimensión VC apropiada. Este principio se basa en que ambos lados de (1.3) deben ser pequeños para obtener un riesgo esperado

bajo. Por lo tanto, tanto la dimensión VC como el riesgo empírico deben ser minimizados a la vez.

Para poder implementar el principio SRM se requiere de una estructura anidada de los espacios de hipótesis $H_1 \subset H_2 \subset ... \subset H_n \subset ...$ con la propiedad que $h(n) \leq h(n+1)$, donde $h(n)$ es la dimensión VC del espacio H_n. Luego, la ecuación (1.3) sugiere que, despreciando los factores logarítmicos, el siguiente problema debe resolverse:

$$\underset{H_n}{\text{Min}} \qquad \left(R_{emp}[\lambda] + \sqrt{\frac{h(n)}{m}} \right) \tag{1.4}$$

La implementación de este principio no es sencilla, pues no es trivial controlar la dimensión VC de una técnica de aprendizaje durante el entrenamiento [29]. El algoritmo SVM consigue este objetivo, minimizando una cota de la dimensión VC y el número de errores de entrenamiento a la vez.

Clasificación Lineal para Problemas Linealmente Separables

Para el caso linealmente separable, SVMs determina el hiperplano óptimo que separa el conjunto de datos. Para este propósito, "linealmente separable" requiere encontrar el par $(\mathbf{w}, b)$ tal que clasifique correctamente los vectores de ejemplos $\mathbf{x}_i$ en dos clases y_i, es decir, para un espacio de hipótesis dado por un conjunto de funciones $f_{\mathbf{w},b} = signo(\mathbf{w}^T \cdot \mathbf{x}_i + b)$ se imponen las siguientes restricciones:

$$\underset{i=1,...,m}{\text{Min}} \qquad |\mathbf{w}^T \cdot \mathbf{x}_i + b| = 1 \tag{1.5}$$

Los hiperplanos que satisfacen (1.5) se conocen como *hiperplanos canónicos* [29]. Si no se imponen restricciones adicionales, la dimensión VC de los hiperplanos canónicos es $n-1$ [47], es decir, el total de parámetros. Para poder aplicar el principio SRM se requiere minimizar tanto el riesgo empírico (errores de clasificación en el conjunto de entrenamiento) como la dimensión VC.

Vapnik [47] demuestra que, asumiendo que los puntos $\mathbf{x}_i$ se encuentran en una esfera n-dimensional, el conjunto $\{f_{\mathbf{w},b} = signo(\mathbf{w}^T \cdot \mathbf{x}_i + b) \mid ||\mathbf{w}|| \leq A\}$ tiene una dimensión VC h que satisface la siguiente cota:

$$h \leq min\{\lceil A^2 \rceil, n\} + 1 \tag{1.6}$$

El acotamiento la norma de $\mathbf{w}$ lleva a la restricción del conjunto de hiperplanos canónicos. La razón geométrica para esto es simple: la distancia desde un punto $\mathbf{x}_i$ al hiperplano asociado al par $(\mathbf{w}, b)$ es:

$$d(\mathbf{x}_i; \mathbf{w}, b) = \frac{|\mathbf{x}_i \cdot \mathbf{w} + b|}{||\mathbf{w}||} \tag{1.7}$$

De acuerdo a la normalización planteada en (1.5) la distancia entre el hiperplano canónico $(\mathbf{w}, b)$ al punto más cercano es simplemente $\frac{1}{||\mathbf{w}||}$. Por lo tanto, si $||\mathbf{w}|| \leq A$ entonces la distancia del hiperplano canónico al punto más cercano debe ser mayor a $\frac{1}{A}$. En este sentido, el objetivo de SVM es encontrar, entre todos los hiperplanos canónicos que clasifican correctamente los datos, aquel con menor norma, o, equivalentemente, con mínimo $||\mathbf{w}||^2$. Es interesante notar que la minimización $||\mathbf{w}||^2$ es equivalente a encontrar el hiperplano separador para el cual la distancia entre dos envolturas convexas (las dos clases del conjunto de datos de entrenamiento, asumiendo que son linealmente separables), medida a lo largo de una línea perpendicular al hiperplano, es maximizada. Esta distancia se conoce como *margen*. Este problema se formula de la siguiente manera:

$$\begin{array}{ll} \text{Min} & \frac{1}{2}\|\mathbf{w}\|^2 \\ \mathbf{w}, b & \end{array} \tag{1.8}$$

sujeto a

$$y_i \cdot (\mathbf{w}^T \cdot \mathbf{x}_i + b) \geq 1 \qquad i = 1, ..., m,$$

A partir de esta formulación se construye el dual mediante la técnica de los multiplicadores de

Lagrange. La formulación dual permitirá construir funciones de clasificación no lineales, lo que usualmente lleva a un mayor poder predictivo. La formulación dual de (1.8) corresponde a:

$$\begin{array}{ll} \text{Max} & \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,s=1}^{m} \alpha_i \alpha_s y_i y_s \mathbf{x}_i \cdot \mathbf{x}_s \\ \alpha & \end{array} \tag{1.9}$$

sujeto a

$$\sum_{i=1}^{m} \alpha_i y_i = 0$$
$$\alpha_i \geq 0 \qquad i = 1, ..., m.$$

donde α_i representan los multiplicadores de Lagrange asociados a las restricciones de (1.8). Los multiplicadores que cumplen con $\alpha_i > 0$ son llamados *Support Vectors*, ya que son los únicos que participan en la construcción del hiperplano de clasificación. Se tiene además que $\mathbf{w}^* = \sum_{i=1}^{m} \alpha_i^* y_i \mathbf{x}_i$ y $b^* = y_i - \mathbf{w}^* \cdot \mathbf{x}_i$ para cada Support Vector $\mathbf{x}_i$. La función de decisión puede escribirse como:

$$f(\mathbf{x}) = signo(\mathbf{w}^* \cdot \mathbf{x} + b^*) = signo(\sum_{i=1}^{m} y_i \alpha_i^* (\mathbf{x} \cdot \mathbf{x}_i) + b^*) \tag{1.10}$$

Clasificación Lineal para Problemas Linealmente no Separables

Ahora se considera el caso en que no existe un hiperplano separador, es decir, no es posible satisfacer todas las restricciones del problema (1.4).

Con el fin de considerar un costo por ejemplo mal clasificado, se introduce un conjunto adicional de variables ξ_i, $i = 1, ..., m$. SVMs resuelve el siguiente problema de optimización:

$$\begin{array}{ll} \text{Min} & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{m} \xi_i \\ \mathbf{w}, b, \xi & \end{array} \tag{1.11}$$

sujeto a

$$y_i \cdot (\mathbf{w}^T \cdot \mathbf{x}_i + b) \geq 1 - \xi_i \qquad i = 1, ..., m,$$
$$\xi_i \geq 0 \qquad i = 1, ..., m.$$

La función de clasificación se mantiene: $f(\mathbf{x}) = signo(\sum_{i=1}^{m} y_i \alpha_i^* (\mathbf{x} \cdot \mathbf{x}_i) + b^*)$, donde $b^* = y_i - \mathbf{w}^* \cdot \mathbf{x}_i$ para cada *Support Vector* $\mathbf{x}_i$ tal que $0 < \alpha_i < C$.

Clasificación no Lineal

Para el caso no lineal, SVMs proyecta el conjunto de datos a un espacio de mayor dimensión $\mathscr{H}$ utilizando una función $\mathbf{x} \rightarrow \phi(\mathbf{x}) \in \mathscr{H}$, donde se construye un hiperplano separador de máximo margen. El siguiente problema de optimización cuadrática debe resolverse:

$$\begin{array}{ll} \text{Min} & \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{m} \xi_i \\ \mathbf{w}, b, \xi & \end{array} \tag{1.12}$$

sujeto a

$$y_i \cdot (\mathbf{w}^T \cdot \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i \qquad i = 1, ..., m,$$
$$\xi_i \geq 0 \qquad i = 1, ..., m.$$

Bajo esta proyección la solución obtenida aplicando SVM toma la siguiente forma:

$$f(\mathbf{x}) = signo(\mathbf{w}^* \cdot \phi(\mathbf{x}) + b^*) = signo(\sum_{i=1}^{m} y_i \alpha_i^* \phi(\mathbf{x}) \cdot \phi(\mathbf{x}_i) + b^*) \tag{1.13}$$

Notar que los únicos valores que deben calcularse son productos escalares de la forma $\phi(\mathbf{x}) \cdot \phi(\mathbf{y})$ [37]. La proyección es realizada por una función de kernel $K(\mathbf{x}, \mathbf{y}) = \phi(\mathbf{x}) \cdot \phi(\mathbf{y})$, que define un producto interno en $\mathscr{H}$. La función de clasificación $f(\mathbf{x})$ dada por SVM corresponde a:

$$f(\mathbf{x}) = signo(\sum_{i=1}^{m} y_i \alpha_i^* K(\mathbf{x}, \mathbf{x}_i) + b^*) \tag{1.14}$$

La formulación dual puede reformularse de la siguiente manera:

$$\underset{\alpha}{\text{Max}} \quad \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,s=1}^{m} \alpha_i \alpha_s y_i y_s K(\mathbf{x}_i, \mathbf{x}_s) \tag{1.15}$$

sujeto a

$$\sum_{i=1}^{m} \alpha_i y_i = 0$$
$$0 \leq \alpha_i \leq C \qquad i = 1, ..., m.$$

Dentro de las distintas funciones de kernel existentes, las funciones polinomiales y la *radial basis function* (RBF) son más frecuentemente utilizadas en diversas aplicaciones [38]:

1. función polinomial: $K(\mathbf{x}_i, \mathbf{x}_s) = (\mathbf{x}_i \cdot \mathbf{x}_s + 1)^d$, donde $d \in \mathbb{N}$ es el grado del polinomio.
2. *Radial basis function* (RBF): $K(\mathbf{x}_i, \mathbf{x}_s) = \exp\left(-\frac{||\mathbf{x}_i - \mathbf{x}_s||^2}{2\rho^2}\right)$, donde $\rho > 0$ es el parámetro que controla el ancho del kernel.

Extensión a Regresión

El principio de minimización del riesgo estructural presentado al principio de esta sección puede ser aplicado a regresión sin mayores adaptaciones. Considerando ahora una variable dependiente continua $y_i \in \Re \ \forall i = 1, ..., m$, Support Vector Regression [8] obtiene la función óptima $f(\mathbf{x})$ que presenta una desviación máxima de ε con respecto a la variable dependiente y_i para todos los ejemplos de entrenamiento $\mathbf{x}_i$, y al mismo tiempo lo más plana posible [40], es decir, los errores se consideran despreciables si son menores que un parámetro ε, mientras que éstos serán penalizados si sobrepasan este umbral. Esto se consigue minimizando la norma euclidiana del vector de coeficientes $\mathbf{w}$ de forma análoga a la formulación (1.11).

$$\underset{\mathbf{w},b,\xi,\xi^*}{\text{Min}} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}(\xi_i + \xi_i^*) \tag{1.16}$$

sujeto a

$$\begin{aligned} y_i - (\mathbf{w}^T \cdot \mathbf{x}_i + b) &\leq \varepsilon + \xi_i \qquad i = 1, ..., m, \\ (\mathbf{w}^T \cdot \mathbf{x}_i + b) - y_i &\leq \varepsilon + \xi_i^* \qquad i = 1, ..., m. \\ \xi_i,\ \xi_i^* &\geq 0 \qquad i = 1, ..., m. \end{aligned}$$

Para obtener una función no lineal, los ejemplos de entrenamiento se proyectan a un espacio de mayor dimensión utilizando funciones de kernel. Esto es posible ya que en la formulación dual de (1.16) las observaciones aparecen sólo en forma de productos punto. La formulación final es la siguiente:

$$\begin{aligned} \underset{\alpha,\alpha^*}{\text{Max}} \quad & -\frac{1}{2}\sum_{i,s=1}^{m}(\alpha_i - \alpha_i^*)(\alpha_s - \alpha_s^*)K(\mathbf{x_i}, \mathbf{x_s}) \\ & -\varepsilon\sum_{i=1}^{m}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{m} y_i(\alpha_i - \alpha_i^*) \end{aligned} \tag{1.17}$$

sujeto a

$$\begin{aligned} \sum_{i=1}^{m}(\alpha_i - \alpha_i^*) &= 0 \\ 0 \leq \alpha_i &\leq C \qquad i = 1, ..., m. \\ 0 \leq \alpha_i^* &\leq C \qquad i = 1, ..., m. \end{aligned}$$

1.1.3. Selección de Atributos para Support Vector Machines

Para la construcción de modelos de clasificación se desea utilizar la menor cantidad de atributos posibles de manera de obtener un resultado considerado aceptable por el investigador. Sin embargo, el problema radica en la elección y el número de atributos a seleccionar, debido a que esta elección

determina la efectividad del modelo de discriminación construido. Este problema se conoce como *selección de atributos* y es combinatorial en el número de atributos originales [2].

Una desventaja del método SVMs es que no está diseñado para identificar los atributos importantes para construir la regla discriminante [21]. La utilización de la norma euclidiana en la formulación primal de SVMs (1.11) para el cálculo del margen en la función objetivo no busca anular componentes del vector w. Por ejemplo, sean los vectores $w_1 = (0,5;0,5;0,5;0,5)$ y $w_2 = (1;0;0;0)$; ambos poseen la misma norma euclidiana ($||w_1||^2 = ||w_2||^2 = 1$), y por ende ambas soluciones tienen el mismo valor en el problema de minimización que formula SVMs. Sin embargo, el primer caso plantea una solución con cuatro atributos, mientras que el segundo caso utiliza sólo un atributo, siendo los tres restantes irrelevantes para la clasificación. Dado que SVMs no distingue entre ambas soluciones, su diseño podría considerarse no adecuado para lograr una clasificación efectiva y a la vez eficaz en identificar los atributos que no contribuyen a ésta.

De acuerdo a Guyon et al. [13], existen tres estrategias principales para la selección de atributos: los métodos de filtro, los métodos *wrapper* o envolventes, y los métodos *embedded* o empotrados. La primera estrategia utiliza propiedades estadísticas para "filtrar" aquellos atributos que resulten poco informativos antes de aplicar el algoritmo de aprendizaje, mirando sólo propiedades intrínsecas de los datos. En muchos casos un puntaje o *score* de relevancia es calculado para cada atributo, eliminando aquellos con bajo puntaje. Esta estrategia es independiente del algoritmo predictivo, lo que implica ventajas y desventajas:

- Son computacionalmente simples y rápidos de ejecutar.
- Son fácilmente escalables a bases de datos de alta dimensionalidad, ya que la selección de atributos sólo necesita ser aplicada una vez, para luego evaluar el desempeño de diferentes métodos de clasificación.
- Estos métodos ignoran las interacciones con el método predictivo, y, por ende, las relaciones entre las distintas variables.

El último punto es particularmente relevante ya que ignorar las interacciones entre las variables puede afectar negativamente el desempeño de clasificación. Atributos presumiblemente re-

dundantes de acuerdo a medidas informativas pero correlacionados entre sí pueden aportar a la clasificación de forma significativa. Los siguientes dos ejemplos ilustran este efecto [13]: La figura 1.2.a muestra la distribución condicional de dos variables con matrices de covarianza idénticas y direcciones principales diagonales. Se observa que una de las variables (arriba, izquierda en la figura 1.2.a) presenta su distribución condicional completamente traslapada con respecto a la variable objetivo (distinción entre barras negras y blancas), mientras la segunda (abajo, derecha) presenta un poder discriminante importante, sin embargo no alcanza una separación perfecta por sí sola. La utilización de ambas variables en conjunto permite lograr una clasificación perfecta en este caso (arriba, derecha y abajo, izquierda), mejorando significativamente el desempeño de clasificación.

El caso más extremo se presenta en el ejemplo ilustrado en la figura 1.2.b: en este caso se tienen ejemplos de dos clases utilizando cuatro distribuciones normales en las coordenadas (0;0), (0;1), (1;0), and (1;1). Las etiquetas para estos cuatro grupos se distribuyen de acuerdo a la función lógica XOR: f(0;0)=1, f(0;1)=-1, f(1;0)=-1; f(1;1)=1. Notar que las proyecciones en los ejes no entregan separación entre clases (diagonales en Fig. 1.2.b), sin embargo, ambas variables en conjunto permiten obtener una clasificación perfecta con algún clasificador no lineal sencillo.

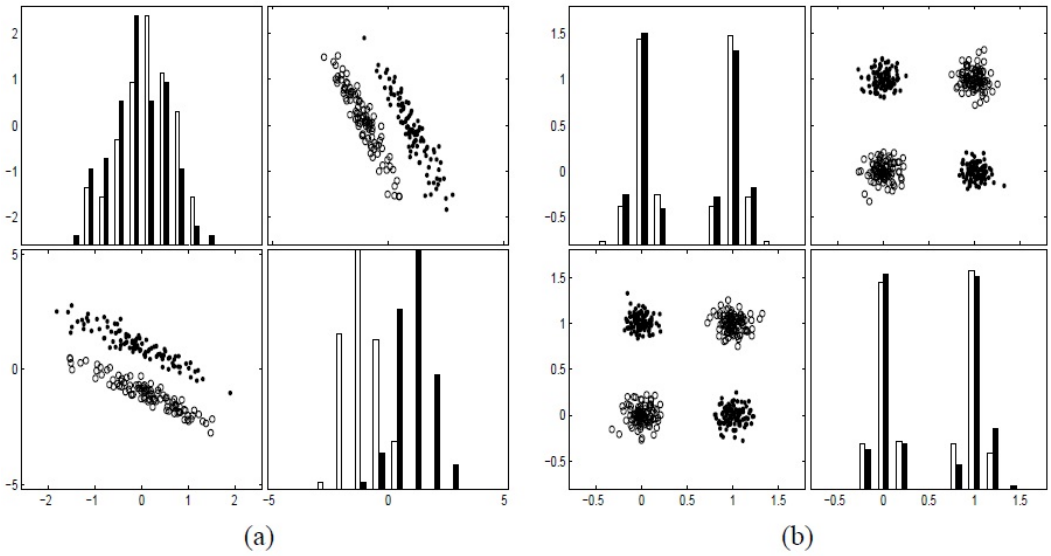

Figura 1.2: Variables irrelevantes por sí solas pero relevantes junto con otras

Una serie de métodos de filtro multivariados han sido introducidos para estudiar la interacción

entre variables. Estas metodologías, sin embargo, suelen ser menos rápidas y escalables que los métodos de filtro univariados [14].

Métodos de filtro univariados utilizados comúnmente son el criterio de Fisher (F), el cual calcula la importancia de cada atributo en forma de score al estimar la correlación de cada atributo con respecto a la variable objetivo en un problema de clasificación binaria. El puntaje $F(j)$ para un atributo particular j viene dado por:

$$F(j) = \left| \frac{\mu_j^+ - \mu_j^-}{(\sigma_j^+)^2 + (\sigma_j^-)^2} \right| \tag{1.18}$$

donde μ_j^+ (μ_j^-) es la media del atributo j para la clase positiva (negativa) y σ_j^+ (σ_j^-) su respectiva desviación estándar. Otras medidas de filtro son el estadístico χ^2, el cual mide la independencia entre la distribución de los ejemplos y clases; y la Ganancia de la Información (*Information Gain*), medida comúnmente utilizada para la construcción de árboles de decisión como método de clasificación, que mide la entropía o "desorden" en el sistema de acuerdo a la Teoría de la Información [45].

Los métodos wrapper o envolventes exploran el conjunto completo de atributos para asignarles un puntaje de acuerdo a su poder predictivo en base a la función de clasificación utilizada, lo cual es computacionalmente demandante, sin embargo, puede traer mejores resultados que la utilización de métodos de filtro. Dado que la búsqueda de subconjuntos de atributos crece de forma exponencial con el número de atributos, heurísticas de búsqueda son utilizadas [13]. Estrategias wrapper frecuentemente utilizadas son la Selección Secuencial hacia Adelante (*Sequential forward selection* o SFS) y la Eliminación Secuencial hacia Atrás (*Sequential backward elimination* o SBE) [18]. Para el primer caso, el modelo parte sin considerar variables, para luego probar cada una de ellas e incluir la más relevante en cada iteración. De la misma manera, SBE parte con todas las variables candidatas a formar parte del modelo, eliminando de forma iterativa aquellas variables irrelevantes para la clasificación.

Una estrategia wrapper para selección de atributos utilizando SVMs que surge de manera natural es considerar los coeficientes w asociados a los atributos como medida para la contribución

de ellos a la clasificación. Una estrategia SBE podría ser aplicada eliminando de forma iterativa los atributos irrelevantes, es decir, aquellos atributos j con un coeficiente w_j asociado cercano a cero en magnitud (considerando datos normalizados), utilizando la formulación primal de SVMs (1.11). La limitación de este método es que la formulación de SVMs no lineal no cuenta con un vector de coeficientes de forma explícita, por lo el método anterior se encuentra limitado a funciones de clasificación lineales. Un popular método wrapper para SVMs basado en la estrategia SBE fue propuesto por Guyon et al.[16] y se conoce como SVM-RFE (SVM- *Recursive Feature Elimination*). El objetivo de este método es encontrar un subconjunto de tamaño r entre las n variables disponibles ($r < n$) que maximice el desempeño de la función de clasificación con SVMs. El atributo que se elimina en cada iteración es aquel cuya extracción minimiza la variación de $W^2(\alpha)$, la cual es una medida de la capacidad predictiva del modelo y es inversamente proporcional al margen:

$$W^2(\alpha) = \sum_{i,s=1}^{m} \alpha_i \alpha_s y_i y_s K(\mathbf{x}_i, \mathbf{x}_s) \tag{1.19}$$

Ventajas de los métodos wrapper incluyen la interacción entre la búsqueda de subconjuntos de atributos y la selección del modelo, y la capacidad de considerar la dependencia entre atributos. Sus principales desventajas son su alto costo computacional y un mayor riesgo de sobre-ajuste del modelo [13]. Dado que los algoritmos de búsqueda wrapper son por lo general de naturaleza *greedy*, existe un riesgo de quedar estancado en un óptimo local y llegar a un subconjunto de atributos insatisfactorio. Para solucionar este problema, una serie de algoritmos de naturaleza aleatoria en la búsqueda han sido creados [14]. Si bien estos algoritmos permiten encontrar un subconjunto más cercano al óptimo, son más costosos aún en términos computacionales.

El tercer y último enfoque de selección de atributos corresponde a las técnicas empotradas o *embedded*. Estos métodos realizan la búsqueda de un subconjunto óptimo de atributos durante la construcción de la función de clasificación. Al igual que los métodos wrapper, estrategias embedded son específicas para un algoritmo de clasificación.

Existen diferentes estrategias para realizar selección de atributos embedded. Por un lado, la selección de atributos puede ser vista como un problema de optimización. Generalmente, la fun-

ción objetivo cumple con dos objetivos: maximización de la bondad de ajuste y minimización del número de atributos [13]. Un método que utiliza esta estrategia fue presentado por Bradley y Mangasarian [3] y minimiza una aproximación de la "norma" cero: $\|\mathbf{w}\|_0 = |\{i : w_i \neq 0\}|$. Esta formulación no puede considerarse una norma ya que la desigualdad triangular no se cumple [3]. La aproximación utilizada por este método, conocido como FSV (*Feature Selection ConcaVe*), es la siguiente:

$$\|\mathbf{w}\|_0 \approx \mathbf{e}^T(\mathbf{e} - \exp(-\beta|\mathbf{w}|) \tag{1.20}$$

donde $\beta \in \Re_+$ es un parámetro de aproximación y $\mathbf{e} = (1,...,1)^T$. El problema se resuelve finalmente con un algoritmo iterativo. Weston et al. [52] demuestra que la minimización de la norma cero para SVM (l_0-SVM) puede aproximarse con una modificación simple del algoritmo *vanilla* SVM:

Algorithm 1.1 Vanilla SVM para selección de atributos

1. Entrenar una SVM lineal de acuerdo a (1.11).
2. Re-escalar las variables multiplicándolas por el valor absoluto de los componentes del vector de pesos $\mathbf{w}$.
3. Iterar los primeros dos pasos hasta convergencia.

Weston argumenta que, en la práctica, esta estrategia permite una mejor generalización que la minimización de la norma cero [52]. La combinación de tres objetivos: bondad de ajuste, un término de regularización (considerando la norma euclidiana o la norma 1 de $\mathbf{w}$) y una penalización a la utilización de atributos ha sido abordado por Perkins et al., donde se propone una estrategia de selección de atributos hacia adelante para optimizar esta formulación. Una limitación de estas estrategias es que se encuentran limitadas a funciones de clasificación lineales [14, 21].

Existen varios enfoques propuestos que utilizan estrategias de selección de atributos que se basan en estrategias de selección hacia adelante o hacia atrás para identificar los atributos relevantes, y de esta manera construir un *ranking*, el cual puede utilizarse a modo de filtro antes de aplicar SVM. Uno de estos métodos es el ya presentado SVM-RFE. Otro método de esta natu-

raleza que permite la utilización de funciones de kernel son los presentados en Rakotomamonjy [32], que utiliza una cota para el error de clasificación *leave-one-out* (LOO) de SVM, el *radius margin bound*[47]:

$$LOO \leq 4R^2||\mathbf{w}||^2 \tag{1.21}$$

donde R denota el radio de la menor esfera inscrita que contiene los datos de entrenamiento. Esta cota también es utilizada en Weston et al. [50] mediante la estrategia conocida como la optimización de factores de escalamiento (*scaling factors*). La selección de atributos con *scaling factors* se realiza mediante el escalamiento de las variables de entrada por un vector $\sigma \in [0,1]^n$. Valores grandes de σ_j indican una mayor relevancia. El problema consiste en encontrar el mejor kernel de la siguiente forma:

$$K_\sigma(\mathbf{x}_i, \mathbf{x}_s) \equiv K(\sigma * \mathbf{x}_i, \sigma * \mathbf{x}_s) \tag{1.22}$$

donde $*$ es el operador para el producto vectorial por componentes. El método presentado por Weston et al. utiliza un algoritmo para actualizar σ mediante el método de descenso del gradiente. Enfoques que utilizan otras cotas para el mismo propósito se presentan en Chapelle et al. [5]. Canu y Grandvalet [4] proponen reducir la utilización de atributos restringiendo los factores de escalamiento en la formulación de SVM mediante un parámetro σ_0 que controla la norma de σ:

$$\begin{array}{cc} \text{Min} & \text{Max} \\ \sigma & \alpha \end{array} \quad \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,s=1}^{m} \alpha_i \alpha_s y_i y_s K_\sigma(\mathbf{x}_i, \mathbf{x}_s) \tag{1.23}$$

sujeto a

$$\sum_{i=1}^{m} \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq C \qquad i = 1, ..., m.$$

$$||\sigma||_p = \sigma_0,$$

con K_σ definido en (1.22). Mientras más cercano a cero sea el parámetro p, más estricta será la selección de atributos, sin embargo, la optimización será también más compleja [14].

A modo general, los métodos *embedded* presentan importantes ventajas como la interacción entre las variables y el modelo de clasificación (en este caso SVMs), la modelación de las dependencias entre variables y ser computacionalmente menos costosos que los métodos *wrapper* [13]. Sin embargo, estos métodos tienden a ser conceptualmente más complejos, y muchas veces las modificaciones impuestas alteran la naturaleza convexa del problema planteado por SVMs, requiriendo algoritmos no lineales que pueden caer en óptimos locales. Adicionalmente, muchos métodos empotrados se encuentran restringidos sólo para SVMs lineal, limitando el potencial que predictivo que otorgan las funciones de Kernel.

Este estudio del estado del arte de la selección de atributos para SVM proporciona una guía general en los diversos aspectos que comprende esta tarea. Además de definir el concepto de selección y de analizar su proceso, se ha clasificado y descrito una gran cantidad de algoritmos existentes. Si bien la investigación en el área de selección de atributos para SVMs tuvo su *peak* para el año 2003, cuyos trabajos se resumen en el libro de Guyon et al. [14], el importante crecimiento de la capacidad de almacenaje, sumado a nuevas aplicaciones de alta dimensionalidad en el mundo de las ciencias de la vida (tales como el estudio del genoma humano) justifican la investigación en esta área. Las últimas publicaciones del estado del arte consideran algoritmos híbridos, que combinan ventajas de distintos modelos de algoritmos (filtros, wrappers, ranking, etc) [42]. Otros trabajos apuntan a abordar el problema de selección de atributos desde el punto de vista de la selección del modelo y no en forma de ranking, independiente de la construcción del modelo predictivo final [15]. El enfoque del presente trabajo es precisamente éste, desarrollar modelos que lleguen a una solución final de clasificación en conjunto con la determinación de los atributos relevantes para el modelo, identificando cuándo la eliminación de atributos comienza a afectar el desempeño de los modelos en el entrenamiento del mismo. Esto trae consigo dos ventajas: primero, es posible establecer un criterio de parada para los métodos, identificando claramente cuando la eliminación de atributos comienza a afectar negativamente el desempeño de los modelos. Segundo, reduce el esfuerzo computacional de construir un modelo final a partir de un ranking de atributos, debiendo posteriormente realizar la selección del modelo mediante algún tipo de evaluación (comúnmente validación cruzada), lo cual es computacionalmente demandante y se corre el riesgo de caer en sobreajuste [15]. Guyon [15]

plantea que la unificación del proceso de selección de atributos y selección del modelo es uno de los tópicos relevantes para la investigación en aprendizaje computacional hoy en día.

1.2. Objetivos

1.2.1. Objetivo General

El presente trabajo de investigación tiene como objetivo principal aportar al desarrollo del aprendizaje computacional mediante el diseño y aplicación de métodos de selección de atributos mediante Support Vector Machines en su versión no lineal. Los métodos diseñados deben ser aplicables a bases de datos de elevada dimensión en el marco del aprendizaje supervisado y cumplir con un desempeño destacable en términos de efectividad y eficiencia.

1.2.2. Objetivos Específicos

Los objetivos específicos son:

1. Desarrollar una guía que involucre todos los elementos relevantes de la selección de atributos en problemas de aprendizaje supervisado, incluyendo un análisis de las metodologías existentes.
2. Mostrar los beneficios de los métodos de selección de atributos propuestos mediante la correcta evaluación y comparación con otros métodos del estado del arte.
3. Estudiar el comportamiento de los métodos propuestos en diferentes bases de datos del mundo real con diversa dimensionalidad.
4. Proponer áreas de trabajo futuro a partir de los métodos desarrollados y potenciales aplicaciones.

1.3. Aportes Originales de la Investigación

Como resultado de la investigación llevada a cabo durante el doctorado se publicaron los siguientes trabajos:

- Maldonado, S., Weber, R. (2009): A wrapper method for feature selection using Support Vector Machines. Information Sciences 179 (13), 2208-2217. Revista ISI, factor de impacto 3.291. Número de citas ISI: seis(actualizado al 14 de Abril de 2011).
- Maldonado, S., Weber, R., Basak, J. (2011): Kernel-Penalized SVM for Feature Selection. Information Sciences 181 (1), 115-128. Revista ISI, factor de impacto 3.291 (actualizado al 14 de Abril de 2011).
- Maldonado, S., Weber, R. (2010): Feature Selection for Support Vector Regression via Kernel Penalization. Proceedings of the 2010 International Joint Conference on Neural Networks, Barcelona, Spain, 1973-1979. ISI Proceedings, factor de impacto 1.83 (actualizado al 14 de Abril de 2011).

Adicionalmente, se presentaron los siguientes trabajos en distintas conferencias y congresos:

- Maldonado, S., Paredes, G.: A Semi-supervised Approach for Reject Inference in Credit Scoring Using SVMs. Industrial Conference on Data Mining (ICDM), Berlín, Alemania, Julio 2010.
- Maldonado, S., Bravo, C., Weber, R.: Practical experiences from credit scoring projects for Chilean financial organizations for micro- entrepreneurs. European OR Conference (EURO), Bonn, Alemania, Julio 2009.
- Maldonado, S.: Feature Selection and Support Vector Machines for Conjoint Analysis. ALIO/ INFORMS International Conference and XV CLAIO (Congreso Latino-Iberoamericano de Investigación Operativa), Buenos Aires, Argentina, Junio 2010.

- Maldonado, S.: Selección de Atributos para Support Vector Machines. Chilean OR Conference (OPTIMA), Termas de Chillán, Chile, Octubre 2009.

1.4. Organización

El contenido de esta Tesis se encuentra dividido en las siguientes partes:

Parte 1: Introducción

Parte 2: A Wrapper Method for Feature Selection using Support Vector Machines [21]

El trabajo presenta un nuevo método *wrapper* de selección de atributos, basado en Support Vector Machines y funciones de Kernel. El enfoque propuesto se basa en una eliminación de atributos secuencial hacia atrás, utilizando el número de errores en un conjunto de validación como medida para decidir el atributo a eliminar en cada iteración. El método propuesto se compara con otras estrategias como métodos de filtro o el algoritmo *wrapper* RFE-SVM para demostrar su efectividad y eficacia. Se destaca como contribución original el planteamiento de una nueva medida para determinar la contribución de un atributo para la clasificación con SVMs y una nuestra estrategia para realizar la selección del modelo y la selección de atributos de forma conjunta, considerando un criterio de parada de forma explícita.

Este trabajo se encuentra disponible en la siguiente dirección web:

http://www.sciencedirect.com/science/article/pii/S0020025509000917

Parte 3: Simultaneous Feature Selection and Classification using Kernel-Penalized Support Vector Machines [23]

Se introduce un método embedded de selección de atributos que simultáneamente selecciona los atributos relevantes durante la construcción de la función de clasificación de SVM. El enfoque presentado, llamado *Kernel-Penalized* SVM (KP-SVM), penaliza la utilización de atributos en la

formulación dual de SVM, optimizando el ancho de una función de kernel RBF no isotrópico, Adicionalmente, KP-SVM emplea un criterio de parada explícito, evitando la eliminación de atributos que puedan afectar negativamente el desempeño de clasificación. Se conducen experimentos sobre cuatro bases de datos utilizadas frecuentemente en la literatura científica, comparando el enfoque propuesto con conocidas técnicas de selección de atributos. KP-SVM presenta un mejor comportamiento empírico en términos de efectividad, mientras utiliza menos atributos que los enfoques alternativos. Además de plantear una estrategia conjunta para la selección del modelo y la selección de atributos, el método plantea como contribución original la penalización de atributos desde el dual, lo cual trae consigo beneficios importantes en términos de reducción de dimensionalidad y desempeño.

Este trabajo se encuentra disponible en la siguiente dirección web:

http://www.sciencedirect.com/science/article/pii/S0020025510004287

Parte 4: Feature Selection for Support Vector Regression via Kernel Penalization [22]

Este trabajo presenta un nuevo enfoque de selección de atributos (KP-SVR), el cual construye una función de regresión no lineal con mínimo error, minimizando de forma simultánea el número de atributos seleccionados, penalizando su utilización en la formulación dual de Support Vector Regression (SVR). El método propuesto optimiza el ancho de un Kernel RBF no isotrópico utilizando un algoritmo iterativo basado en el método del gradiente, eliminando atributos con baja relevancia para el modelo de regresión. El enfoque cuenta además con un criterio explícito de parada, indicando cuando la eliminación de atributos comienza a afectar en forma negativa el desempeño del modelo. Experimentos en dos bases de datos de *benchmark* demuestran que el método desarrollado consigue una mejor *performance* en comparación con estrategias conocidas de selección de atributos, obteniendo menos variables de forma consistente. Esta extensión del trabajo presentado en [23] plantea como contribución original el desarrollo de un modelo embedded de selección de atributos para SVR, tema que prácticamente no se ha tratado en la literatura.

Este trabajo se encuentra disponible en la siguiente dirección web:

http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5596488

Bibliografía

[1] Blazadonakis, M.E., Zervakis, M. (2008), Wrapper filtering criteria via linear neuron and kernel approaches Computers in Biology and Medicine 38(8), 894-912.

[2] Blum, A., P. Langley, P. (1997): Selection of relevant features and examples in machine learning. Artificial Intelligence 97, 245-271.

[3] Bradley, P., Mangasarian, O. (1998): Feature selection vía concave minimization and support vector machines. Machine Learning proceedings of the fifteenth International Conference (ICML'98) 82 -90, San Francisco, California, Morgan Kaufmann.

[4] Canu, S., Grandvalet, Y. (2002): Adaptive scaling for feature selection in SVMs. Advances in Neural Information Processing Systems 15, Cambridge, MA, USA, MIT Press, 553-560.

[5] Chapelle, O., Vapnik, V., Bousquet, O., Mukherjee, S. (2002): Choosing multiple parameters for Support Vector Machines. Machine Learning 46 (1), 131-159.

[6] Coloma, P., Guajardo, J., Miranda, J., Weber, R. (2006): Modelos analíticos para el manejo del riesgo de crédito. Trend Management 8, 44-51.

[7] Cristianini, N., Shawe-Taylor, J. (2000): An Introduction to Support Vector Machines. Cambridge University Press, Cambridge, UK.

[8] Drucker, H., Burges, C., Kaufman, L., Smola, A., Vapnik, V. (1997): Support Vector Regression Machines. Advances in Neural Information Processing Systems 9, NIPS 1996, 155-161, MIT Press.

[9] Famili, A., Shen, W.-M., Weber, R., Simoudis, E. (1997): Data Preprocessing and Intelligent Data Analysis. Intelligent Data Analysis 1, 3-23.

[10] Fayyad, U. (1996): Data mining and knowledge discovery- making sense out of data. IEEE Expert-Intelligent Systems and Their Aplications 11, 20-25.

[11] Fayyad, U., Piatetsky-shapiro, G., Smyth, P. (1996): From Data Mining to Knowledge Discovery in Databases. AI Magazine 17, 37-54.

[12] Guajardo, J., Weber, R. , Miranda, J. (2010): A Model Updating Strategy for Predicting Time Series with Seasonal Patterns. Applied Soft Computing 10, 276-283.

[13] Guyon, I., Elisseeff, A.(2003): An Introduction to Variable and Feature Selection. Journal of Machine Learning research 3, 1157-1182.

[14] Guyon, I., Gunn, S., Nikravesh, M. , Zadeh, L. A. (2006): Feature extraction, foundations and applications. Springer, Berlin.

[15] Guyon, I., Saffari, A., Dror, G., Cawley, G. (2009): Model selection: Beyond the Bayesian frequentist divide. Journal of Machine Learning research 11, 61-87.

[16] Guyon, I., Weston, J., Barnhill, S. ,Vapnik, V. (2002): Gene selection for cancer classification using support vector machines, Machine Learning 46 (1-3), 389-422.

[17] Hettich, S., Bay, S. D.(1999): The UCI KDD Archive http://kdd.ics.uci.edu. Irvine, CA: University of California, Department of Information and Computer Science.

[18] Kittler, J. (1978): Pattern Recognition and Signal Processing, Chapter Feature Set Search Algorithms Sijthoff and Noordhoff, Alphen aan den Rijn, Netherlands, 41-60.

[19] Lal, T.N., Chapelle, O., Weston, J., Elisseeff, A. (2006):Embedded methods. In: I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh (Eds.): Feature Extraction: Foundations and Applications. Studies in Fuzziness and Soft Computing 207, Springer, Berlin Heidelberg, 137-165.

[20] Liu, Y., Zheng, Y. F. (2006). FS-SFS: A novel feature selection method for support vector machines. Pattern Recognition 39, 1333-1345.

[21] Maldonado, S., Weber, R. (2009): A wrapper method for feature selection using Support Vector Machines. Information Sciences 179 (13), 2208-2217.

[22] Maldonado, S., Weber, R. (2010): Feature Selection for Support Vector Regression via Kernel Penalization. Proceedings of the 2010 International Joint Conference on Neural Networks, Barcelona, Spain, 1973-1979.

[23] Maldonado, S., Weber, R., Basak, J. (2011): Kernel-Penalized SVM for Feature Selection. Information Sciences 181 (1), 115-128.

[24] Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M., Euler, T.(2006): YALE: Rapid Prototyping for Complex Data Mining Tasks, in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-06).

[25] Miranda, J., Montoya, R., Weber, R. (2005): Linear Penalization Support Vector Machines for Feature Selection. S.K. Pal et al. (Eds.): PReMI 2005, LNCS 3776, 188-192, Springer-Verlag, Berlin Heidelberg.

[26] Mitchell, T. (1997): Machine Learning, McGraw Hill.

[27] Nemhauser, G., Wolsey, L. (1988): Integer and Combinatorial Optimization. John Wiley and Sons, New York.

[28] Neumann, J., Schnörr, C., Steidl, G. (2005): Combined SVM-Based Feature Selection and Classification. Machine Learning 61 (1-3), 129-150.

[29] Osuna, E., Freund, R., Girosi, F. (1997): Support Vector Machines: Training and Applications. MIT Artificial Intelligence Laboratory, A. I. Memo AIM-1602.

[30] Perkins, S. ,Lacker, K., Theiler, J.(2003): Grafting: Fast incremental feature selection by gradient descent in function space. Journal of Machine Learning research 3, 1333-1356.

[31] Pyle, D. (1999): Data preparation for data mining. Morgan Kaufmann Publishers.

[32] Rakotomamonjy, A. (2003): Variable Selection Using SVM-based Criteria. Journal of Machine Learning research 3, 1357-1370.

[33] Rätsch, G., Onoda, T., and Müller, K-R (2001). Soft margins for AdaBoost. Machine Learning 42(3), 287-320.

[34] Redmond, M. A., Baveja, A. (2002): A Data-Driven Software Tool for Enabling Cooperative Information Sharing Among Police Departments. European Journal of Operational Research 141, 660-678.

[35] Ruiz Sánchez, R. (2006): Heurísticas de selección de atributos para datos de gran dimensionalidad. Tesis Doctoral, Sevilla, Universidad de Sevilla. Mimeografiada.

[36] Reunanen, J., Guyon, I., Elisseeff, A. (2003): Overfitting in Making Comparisons Between Variable Selection Methods. Journal of Machine Learning Research 3, 1371-1382.

[37] Schölkopf, B. and Smola, A. J.(2002). Learning with Kernels. Cambridge, MA, USA: MIT Press.

[38] Shawe-Taylor, J., Cristianini, N. (2004): Kernel Methods for Pattern Analysis. Cambridge University Press, Cambridge.

[39] Shieh, M-D., Yang, C-C. (2008): Multiclass SVM-RFE for product form feature selection. Expert Systems with Applications 35(2), 531-541.

[40] Smola, A. J., Schölkopf, B.(2003): A Tutorial on Support Vector Regression. Statistics and Computing 14(3),199-222.

[41] Suykens, J.A.K., Vandewalle, J. (1999): Least Squares Support Vector Machine Classifiers. Neural Processing Letters 9(3), 293-300.

[42] Uncu, Ö., Türksen, I.B. (2007): A novel feature selection approach: Combining feature wrappers and filters. Information Sciences 177, 449-466.

[43] Unler, A., Murat, A., Chinnam, R.B. (2010): mr^2PSO: A Maximum Relevance Minimum Redundancy Feature Selection Method Based on Swarm Intelligence for Support Vector Machine Classification. Information Sciences, in Press.

[44] Van Gestel, T., Suykens, J.A.K., De Moor B., Vandewalle, J. (2001): Automatic relevance determination for least squares support vector machine classifiers. Proceedings of the European Symposium on Artificial Neural Networks (ESANN 2001), Bruges, Belgium, 13-18.

[45] Van Hulse, J., Khoshgoftaar, T.M., Napolitano, A., Wald, R. (2009): Feature Selection with High-Dimensional Imbalanced Data. Proceedings of the 2009 IEEE International Conference ICDMW '09, 507 - 514.

[46] Vapnik, V. (1982): Estimation of dependences based on empirical data. Springer Verlag, New York.

[47] Vapnik, V. (1998): Statistical Learning Theory. John Wiley and Sons, New York.

[48] Vapnik, V., Chervonenkis, A. (1991): The necessary and sufficient conditions for consistency in the empirical risk minimization method. Pattern Recognition and Image Analysis, 1(3):283-305.

[49] Yang, J., Liu, G. (2002): The evaluation of classification models for credit scoring. Arbeitsbericht Nr. 02-2002 Institut für Wirtschaftsinformatik, Georg-August-Universität Göttingen.

[50] Weston, J., Mukherjee, S., Chapelle, O., Ponntil, M., Poggio, T., Vapnik, V.(2001): Feature selection for SVMs, Advances in Neural Information Processing Systems 13, MIT Press, Cambridge, MA.

[51] Weston, J., Elisseeff, A., Bakir, G., Sinz, F.: The spider. http://www.kyb.tuebingen.mpg.de/bs/people/spider/.

[52] Weston, J., Elisseeff, A., Schölkopf, B., Tipping, M.(2003): The use of zero-norm with linear models and kernel methods. Journal of Machine Learning research 3, 1439-1461.

[53] Zhang, M. L., Pena, J. M., Robles, V.(2009): Feature selection for multilabel naive Bayes classification, Information Sciences 179(19), 3218-3229.