



Ingeniare. Revista Chilena de Ingeniería

ISSN: 0718-3291

facing@uta.cl

Universidad de Tarapacá

Chile

Velásquez, Juan D.; Olaya, Yris; Franco, Carlos J.  
PREDICCIÓN DE SERIES TEMPORALES USANDO MÁQUINAS DE VECTORES DE SOPORTE  
Ingeniare. Revista Chilena de Ingeniería, vol. 18, núm. 1, 2010, pp. 64-75  
Universidad de Tarapacá  
Arica, Chile

Disponible en: <http://www.redalyc.org/articulo.oa?id=77218811008>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica  
Red de Revistas Científicas de América Latina, el Caribe, España y Portugal  
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

## PREDICCIÓN DE SERIES TEMPORALES USANDO MÁQUINAS DE VECTORES DE SOPORTE

### TIME SERIES PREDICTION USING SUPPORT VECTOR MACHINES

Juan D. Velásquez<sup>1</sup>   Yris Olaya<sup>1</sup>   Carlos J. Franco<sup>1</sup>

Recibido el 12 de agosto de 2008, aceptado el 19 de noviembre de 2009

*Received: August 12, 2008   Accepted: November 19, 2009*

#### RESUMEN

La predicción de series de tiempo es un importante problema de investigación debido a sus implicaciones en ingeniería, economía, finanzas y ciencias sociales. Un importante tópico de esta problemática es el desarrollo de nuevos modelos y su comparación con aproximaciones previas en términos de la precisión del pronóstico. Recientemente, las máquinas de vectores de soporte (SVM) han sido usadas para la predicción de series de tiempo, pero las experiencias reportadas son limitadas y hay algunos problemas relacionados con su especificación. El objetivo de este artículo es proponer una técnica novedosa para estimar algunas constantes en las SVM que usualmente son fijadas en forma empírica por el modelador. La técnica propuesta es usada para estimar varias SVM con el fin de pronosticar cinco series *benchmark*; los resultados obtenidos son comparados con las estadísticas reportadas en otros artículos. La metodología propuesta permite obtener SVM competitivas para las series pronosticadas en comparación con los resultados obtenidos usando otros modelos más tradicionales.

Palabras clave: Series de tiempo, predicción, redes neuronales artificiales, máquinas de vectores de soporte.

#### ABSTRACT

*Time series prediction is an important research problem due to its implications in engineering, economics, finance and social sciences. An important topic about this problematic is the development of new models and its comparison with previous approaches in terms of forecast accuracy. Recently, support vector machines (SVM) have been used for time series prediction, but the reported experiences are limited and there are some problems related to its specification. The aim of this paper is to propose a novel technique for estimating some constants of the SVM usually fixed empirically by the modeler. The proposed technique is used to estimate several SVM with the aim of forecast five benchmark time series; the obtained results are compared with the statistics reported in other papers. The proposed method allow us to obtain competitive SVM for the time series forecasted in comparison with the results obtained using other most traditional models.*

*Keywords: Time series, forecasting, artificial neural networks, support vector machines.*

#### INTRODUCCIÓN

La predicción del comportamiento futuro de diferentes variables físicas, sociales, financieras y económicas ha sido un problema de particular interés para científicos y profesionales de diversas áreas [1]. Se busca encontrar el modelo que permita determinar con la mayor precisión posible los valores futuros de la serie investigada, para uno o más periodos hacia delante. Para abordar este problema se han desarrollado numerosas técnicas que abarcan modelos estadísticos tradicionales tales como los ARIMA y las funciones de transferencia [2], hasta modelos no lineales [3]. Particularmente, los modelos

de redes neuronales artificiales han sido de especial utilidad cuando la serie estudiada presenta características no lineales; en [4] se presenta una revisión general, y aplicaciones específicas en [5-13].

Particularmente, los perceptrones multicapa (MLP, por sus siglas en inglés) han sido ampliamente utilizados para la predicción de series no lineales, debido a que pueden aproximar cualquier función continua definida en un dominio compacto [14-16]. Sin embargo, su especificación se basa fundamentalmente en criterios heurísticos y juicio experto del modelador [17, 18], de tal manera, que dicho proceso se basa en un conjunto de pasos críticos que

<sup>1</sup> Escuela de Ingeniería de Sistemas. Universidad Nacional de Colombia. Carrera 80 N° 65-223. Medellín, Colombia. E-mail: jdvelasq@unal.edu.co; yolayam@unal.edu.co; cjfranco@unal.edu.co

afectan el desempeño final del modelo [19], en términos de su ajuste a los datos históricos y su capacidad de generalización con el fin de realizar pronósticos.

Las máquinas de vectores de soporte (SVM, por sus siglas en inglés) [20, 21] son un tipo de red neuronal que fue originalmente diseñada para la solución de problemas no lineales de clasificación [22, 23], pero que recientemente se ha aplicado a problemas de regresión [21, 24] y predicción de series temporales [25-29]. Ello se debe a su capacidad de generalización [30, 31], la cual es función directa de su estructura y de la metodología utilizada para la estimación de sus parámetros.

Las SVM presentan varias ventajas sobre otras técnicas no lineales (que incluyen los MLP):

- (a) Su especificación se basa en el principio de minimización del riesgo estructural que equivale a minimizar el límite superior del error de generalización del modelo. Así, la estimación de una SVM busca encontrar tanto la estructura óptima del modelo como los valores óptimos de sus parámetros, permitiendo una mayor capacidad de generalización del modelo. La estimación de los parámetros de otros modelos tradicionales, tales como los MLP, se basa en la aplicación del principio de minimización de riesgo empírico que depende fundamentalmente del ajuste a los datos históricos, tomando la estructura del modelo como un parámetro de entrada.
- (b) La estimación de los parámetros de una SVM es equivalente a la solución de un modelo de programación cuadrática con restricciones lineales; ello implica, que la solución óptima es global y única, lo cual representa una clara ventaja sobre modelos como los MLP que se caracterizan por poseer múltiples puntos de mínima local.

El problema de programación cuadrática de cuya solución se obtienen los parámetros de las SVM depende de los datos (la serie de tiempo) y de varias constantes que representan los parámetros de la función de núcleo (o kernel) utilizada, y de los parámetros de la función de riesgo. Su definición detallada se presenta en la próxima sección. Las constantes anteriores son dependientes de cada problema particular, y no existen métodos que permitan su estimación, por lo que sus valores deben ser fijados de manera heurística por el experto; es así como en las referencias analizadas [25-29] no se presentan sugerencias sobre este punto. Este aspecto dificulta enormemente el proceso de desarrollo de un modelo de predicción para una serie de tiempo usando SVM.

Por otra parte, las experiencias reportadas en la literatura sobre la predicción de series de tiempo usando SVM son limitadas, y no es posible realizar comparaciones sistemáticas de las SVM con otros modelos. Un resumen de los estudios seleccionados se presenta en la Tabla 1. La escasa literatura encontrada muestra que el uso de las SVM para la predicción de series de tiempo ha sido poco difundido.

El primer objetivo de este artículo es proponer una metodología para la selección de las constantes que controlan el desempeño de las SVM con el fin de facilitar el desarrollo de modelos de predicción de series temporales.

El segundo objetivo es pronosticar cinco series de tiempo del mundo real que presentan diferentes características estructurales usando la metodología propuesta para comparar los resultados obtenidos con los estadísticos reportados por otros autores que usaron la metodología ARIMA y MLP. Ello permitirá que en el largo plazo se pueda establecer una clara comparación con otras metodologías, ya que las series utilizadas son comúnmente aceptadas para este fin. La decisión de tomar los resultados obtenidos por otros investigadores para los modelos más tradicionales está motivada en el hecho de poder garantizar que se usan estadísticos de error realmente competitivos que han sido estimados por expertos en la metodología; así, se evita una objeción sobre la calidad de la comparación realizada.

El tercer objetivo es contribuir a la difusión del uso de las SVM como una técnica competitiva para la predicción de series de tiempo que puede producir mejores pronósticos que otros métodos más tradicionales.

Este artículo consta de cuatro secciones. En la primera se examina la formulación matemática de las SVM. En la siguiente sección se presenta una metodología para la selección de las constantes que controlan el desempeño de las SVM. En la tercera sección se realiza una comparación del desempeño de las SVM para la predicción de cinco series de tiempo que han sido utilizadas comúnmente como *benchmark* por otros investigadores para evaluar el desempeño de modelos no lineales. En la última sección se concluye.

## MÁQUINAS DE VECTORES DE SOPORTE

En esta sección se presenta una descripción detallada de la arquitectura de las máquinas de vectores de soporte, el proceso de estimación de sus parámetros y las funciones de núcleo comúnmente utilizadas.

Tabla 1. Estudios seleccionados sobre la predicción de series de tiempo usando SVM.

Estudio	Año	Modelos	Serie
Mukherjee, Osuna y Girosi [25]	1997	SVM Polinomios Redes RBF MLP	Mackey-Glass Mapa de Ikeda Lorenz
Muller, Smola, Rätsch, Schölkopf, Kohlmorgen, y Vapnik [26]	1997	SVM RBF	Mackey-Glass Serie D-Santa Fe Competition
Tay y Cao [28]	2001	SVM MLP	5 series de precios en diferencias relativas
Thissen, van Brakel, Weijer, Melssen y Buydens [29]	2003	SVM Red de Elman MLP	Serie sintética generada con un ARIMA Mackey-Glass Diferencias de presión en un filtro
Kim [41]	2003	SVM	Series de precios
Levis and Papageorgiou [39]	2005	SVM	Demanda de consumidores
Pai and Lin [40]	2005	ARIMA SVM	Precios de acciones
Pai y Hong [37]	2005	SVM recurrente ANN Regresión	Demanda de electricidad en Taiwán
Osowski y Garanty [27]	2007	Wavelets + SVM	Polución atmosférica
Fernández [36]	2007	ARIMA UCM SVM Wavelets+SVM	Metales y materiales de la industria manufacturera de Estados Unidos
Chen y Wang [42]	2007	SVM	Demanda turística
Guo, Sun, Li and Wang [38]	2008	Wavelets + SVM	Producción manufacturera

**El modelo**

Sea una serie de tiempo,  $y_t$ , con regresores  $x_t$ , para la cual se poseen  $D$  ejemplos representativos. Una SVM permite aproximar  $y_t$  a través de la función:

$$\hat{y}_t = b + \sum_{d=1}^D w_d \times k(x_t, x_d) \quad (1)$$

Donde  $b$  es una constante y  $w_d$  son los factores de ponderación de la función de núcleo  $k(\cdot, \cdot)$ . Así, una SVM es la combinación lineal del mapeo de  $x_t$  en un espacio de características altamente no lineal definido por los puntos  $x_d$  y la función de transformación no lineal  $k(\cdot, \cdot)$ .

**Estimación**

La estimación de (1) se basa en minimización de la función de riesgo regularizado,  $R(C, \varepsilon)$ , definida como:

$$R(C, \varepsilon) = C \frac{1}{D} \sum_{d=1}^D L_\varepsilon(y_d, \hat{y}_d) + \frac{1}{2} \sum_{d=1}^D w_d^2 \quad (2)$$

Donde el primer sumando mide el error empírico entre el modelo y los datos, mientras que el segundo corresponde a la componente de regularización y depende únicamente de los pesos  $w_d$ . La constante de regularización  $C$  permite variar la importancia de cada una de las componentes; así, valores muy altos de  $C$  enfatizan el ajuste del modelo a los datos, sin que importe que tan grandes deban ser los pesos  $w_d$  para conseguirlo; sin embargo, se sabe que el modelo pierde su capacidad de generalización de los datos a medida que los pesos  $w_d$  aumentan en magnitud, ya que ellos suelen causar una varianza excesiva en el modelo. Cuando  $C$  tiende a cero, la magnitud de la función  $R(C, \varepsilon)$  depende únicamente de  $w_d$ , sin importar el ajuste a los datos, haciendo que los pesos  $w_d$  disminuyan tanto como sea posible.  $L_\varepsilon$  es la función de error  $\varepsilon$ -insensible de Vapnik [30] definida como:

$$L_\varepsilon(y_d, \hat{y}_d) = \begin{cases} |y_d - \hat{y}_d| - \varepsilon & \text{para } |y_d - \hat{y}_d| > \varepsilon \\ 0 & \text{en otro caso} \end{cases} \quad (3)$$

Donde la constante  $\varepsilon$  es la precisión deseada, y representa el radio del tubo dentro del cual el error se considera cero. Este comportamiento es esquematizado en la Figura 1, donde los puntos negros representan los datos disponibles, que están alejados a una distancia mayor de  $\varepsilon$  unidades de la predicción del modelo, y para los cuales el error se considera superior a cero; los puntos blancos representan los datos dentro del túnel de tolerancia del modelo, y para los cuales el error se considera cero. Mientras menor sea el valor dado a  $\varepsilon$ , mayor es la precisión exigida, ya que se tendrá en cuenta una mayor cantidad de puntos que contribuyen al error de ajuste.

La solución de (2) puede ser obtenida mediante la teoría de multiplicadores de Lagrange, y es fácilmente demostrable que el problema original es equivalente a resolver el problema de programación cuadrática [30] [34]:

$$\min_z \frac{1}{2} z' H z + f' z \quad (4)$$

Figura 1. Función de error.  
Fuente: Elaboración propia.

Con restricciones lineales:

$$\sum_{d=1}^D (a_d - a_d^*) = 0 \quad (5)$$

$$0 \leq a_d, a_d^* \leq C \quad (6)$$

Donde  $a_d$  y  $a_d^*$  son los multiplicadores de Lagrange asociados a los datos, los cuales cumplen con las restricciones  $a_d \times a_d^* = 0$ ,  $a_d \geq 0$ , y  $a_d^* \geq 0$ ;  $z = [a_d, a_d^*]'$ ;  $f = [\varepsilon \mathbb{I} y_t, \varepsilon + y_t]$ ; y finalmente,

$$H = \begin{bmatrix} -K & K \\ K & -K \end{bmatrix} \quad (7)$$

En esta última ecuación,  $K$  es una matriz de orden  $D \times D$ , con  $k_{ij} = k(x_i, x_j)$ . Los pesos  $w_d$  se obtienen como:

$$w_d = a_d - a_d^* \quad (8)$$

En la práctica, esto equivale a que  $|w_d| \leq C$ .

La formulación como un problema de programación cuadrática implica: (a) que la solución obtenida es global, ya que la formulación del problema es convexa; (b) que la solución es única, ya que la función de costo es estrictamente convexa, y (c) que muchos de los pesos  $w_d$  serán cero; de tal forma, que solamente una fracción de los datos originales contribuyen en la función de error. Esta conclusión es basada en las condiciones de Karush-Kuhn-Tucker para la programación cuadrática. Los puntos para los cuales sus correspondientes multiplicadores de Lagrange son diferentes de cero (esto es, con  $L_\varepsilon(\cdot, \cdot) > 0$ ) son llamados vectores de soporte, y son los que permiten realizar la aproximación de  $y_t$  en (1). Así, se espera que a medida que aumente el valor de  $\varepsilon$ , disminuya la cantidad de vectores de soporte; pero que también disminuya la capacidad de aproximación del modelo a los datos originales.

### Funciones de núcleo (kernels)

En (1),  $k(x, x_d)$  representa una función de núcleo que permite llevar el punto  $x$  a un espacio altamente dimensional parametrizado por los puntos  $x_d$ . Existen varias funciones que son típicamente utilizadas como núcleos [30-31, 35]:

- Lineal:  $x x_d$
- Polinomial:  $(a_1 x x_d + a_2)^d$ .
- Gaussiana o RBF:  $\exp[-1/a_1^2 \times (x - x_d)^2]$
- Exponencial:  $\exp[-1/a_1^2 \times (x - x_d)]$
- Perceptron multicapa:  $\tanh(a_1 x x_d + a_2)$ .

Las constantes  $a_1$  y  $a_2$  son dependientes del problema y sus valores deben ser ajustados por el modelador.

En muchas de las referencias reportadas en la Tabla 1 se indica que la función gaussiana o RBF presenta un mejor desempeño para la solución de problemas de regresión y de predicción de series temporales. Consecuentemente, en este artículo sólo se considerará el uso de esta función de núcleo, pero la metodología presentada en la siguiente sección es aplicable a cualquiera de las restantes funciones de activación.

### METODOLOGÍA DE SELECCIÓN DE LAS CONSTANTES

Dada una serie temporal, una SVM con función de núcleo RBF (cuya desviación estándar es  $\sigma$ ) y valores para las constantes  $C$  y  $\varepsilon$  es posible determinar los valores óptimos de los parámetros  $b$  y  $w_d$  de la SVM definida en (1), a partir de la solución del problema de programación cuadrática definido por (4), (5) y (6). Tal como ya se

indicó, el modelo definido por las ecuaciones anteriores tiene una solución única y global, pero dependiente de los valores asumidos para  $\sigma$ ,  $C$  y  $\varepsilon$ . La dificultad radica en que estos valores son dependientes del problema particular analizado, y en que no hay métodos heurísticos para la determinación de sus valores, por lo que el experto debe fijar sus valores de forma heurística. En esta sección se propone una metodología para la fijación de sus valores iniciales, y un proceso para su optimización.

### Fijación de los valores iniciales de $\sigma$ , $C$ y $\varepsilon$

Sean los vectores de regresores  $x_i$  y  $x_j$  para la serie de tiempo  $y_t$  obtenidos de la información disponible para la estimación del modelo. En (1),  $k(x_i, x_j)$  se define como la función gaussiana o RBF:

$$k(x_i, x_j) = \exp\left(-\frac{1}{\sigma^2} \|x_i - x_j\|^2\right) \quad (9)$$

Donde  $\|\cdot\|$  representa la distancia entre los vectores  $x_i$  y  $x_j$ . De esta forma (9) puede escribirse en términos de la distancia  $d$  entre los vectores  $x_i$  y  $x_j$ :

$$k(d) = \exp\left(-\frac{1}{\sigma^2} d^2\right) \quad (10)$$

Dicha distancia oscila entre 0 y un valor positivo arbitrario  $d_{max}$  que representa la máxima distancia entre dos vectores de regresores. Bajo esta premisa (10) sólo toma valores en el intervalo  $[0, d_{max}]$ , siendo  $k(0) = 1$  y  $k(d_{max}) \rightarrow 0$ . El problema de la selección del valor  $d$  radica en que para valores muy grandes de  $\sigma$  el valor de  $k()$  en (1) sería prácticamente cero, mientras que para valores muy pequeños,  $k()$  valdría prácticamente la unidad. Ya que  $d_{max}$  puede ser estimado a partir de la información disponible, el valor inicial de  $\sigma$  puede obtenerse como el  $\sigma$  tal que  $k(d_{max}) = a$ ; donde  $a$  es una constante arbitraria cercana a cero, por ejemplo,  $a = 0,05$ . Esto garantiza que la función  $k()$  toma valores en el rango  $[a, 1]$  para la serie de tiempo analizada.

Por otra parte, la constante  $\varepsilon$  toma valores entre 0 y un valor positivo desconocido que representa el error máximo entre el modelo y la serie de datos. Ya que el modelo (1) no ha sido estimado, la serie de errores es desconocida, por lo que no es posible fijar  $\varepsilon$  en un valor inicial adecuado. Un valor inicial para  $\varepsilon$  puede ser obtenido si se supone que  $y_t$  sigue un proceso autorregresivo de orden uno:

$$y_t = \phi \cdot y_{t-1} + e_t \quad (11)$$

Donde  $\phi$  es una constante y  $e_t$  son los residuales. El valor inicial de  $\varepsilon$  puede obtenerse como un porcentaje del valor máximo de  $e_t$  en (11).

En esta investigación no se encontraron mecanismos para obtener un valor inicial para  $C$ , por lo que se supone que su valor inicial es igual a la unidad.

### Optimización de los valores de $\sigma$ , $C$ y $\varepsilon$

Una vez que se han fijado valores para el vector  $(\sigma, C, \varepsilon)$  es posible estimar los valores óptimos de (1) y estimar un estadístico que mida el error de ajuste del modelo a la serie de tiempo. En principio cualquier técnica de optimización podría ser utilizada para estimar los valores óptimos de  $(\sigma, C, \varepsilon)$ , pero la obtención de la solución de problema de programación cuadrática es computacionalmente costosa. En esta investigación se utilizó un proceso de búsqueda discreta en donde se evalúan secuencialmente los puntos en la vecindad de  $(\sigma, C, \varepsilon)$ . Las soluciones vecinas se obtienen como  $(\sigma, C, \varepsilon) + (\pm\Delta\sigma, \pm\Delta C, \pm\Delta\varepsilon)$ ; cuando se detecta una dirección que permite disminuir el error de ajuste, se continúa avanzando en esta dirección hasta encontrar un punto de mínima local; el proceso de optimización termina cuando ningún punto en la vecindad de la solución actual permite disminuir el error de ajuste.

Dependiendo de los valores de  $(\sigma, C, \varepsilon)$ , la SVM puede sobreajustarse a la muestra de entrenamiento, tal como ocurre con otros tipos de redes neuronales. Para evadir este problema, los datos para el entrenamiento del modelo son divididos en un conjunto de ajuste con el que se calculan los valores óptimos de los parámetros de (1) dado  $(\sigma, C, \varepsilon)$ ; y en un conjunto de validación que se usa para determinar si se ha presentado el sobreajuste del modelo. El proceso de optimización de  $(\sigma, C, \varepsilon)$  se detiene cuando:

1.  $(\sigma, C, \varepsilon)$  convergió a un punto de mínima.
2. El error cuadrático medio para la muestra de ajuste (MSE) entre las iteraciones  $k$  y  $k+1$  (esto es, cuando se ha detectado una dirección en que disminuye el error), se reduce muy lentamente. Dicha velocidad, medida como  $(MSE_k - MSE_{k-1})/MSE_k = \varepsilon_1$ , es inferior a un valor  $\varepsilon_1^*$  predefinido por el experto [32]. Y, cuando la proporción entre los errores de ajuste ( $MSE_T$ ) y validación ( $MSE_V$ ) para la iteración actual,  $|MSET - MSEV|/MSET = \varepsilon_2$  es superior a un valor  $\varepsilon_2^*$  predefinido por el experto [32].

### DESEMPEÑO

En esta sección se reportan los resultados obtenidos al pronosticar las series descritas a continuación utilizando SVM obtenidas con la metodología de la sección anterior. Para cada serie considerada, las SVM obtenidas difieren en los rezagos usados para su predicción. Los valores óptimos de  $(\sigma, C, \varepsilon)$  para cada SVM se obtuvieron utilizando el procedimiento descrito en la sección anterior.

- (a) Serie AIRLINE: Corresponde al número de pasajeros transportados mensualmente al exterior por una aerolínea entre ENE1949 y DIC1960; esta serie presenta una tendencia creciente más un patrón cíclico de periodicidad anual. En las experiencias reportadas en la literatura, los primeros 120 datos son utilizados para la estimación del modelo, y los 12 restantes para su validación. La serie modelada corresponde a la serie original transformada usando la función logaritmo natural.
- (b) Serie POLLUTION: Es la cantidad de despachos mensuales de un equipo de polución en miles de francos franceses, entre ENE1986 y OCT1996. La serie modelada corresponde al logaritmo natural de los datos originales. Las primeras 106 observaciones son usadas para la estimación de los modelos, mientras que las 24 restantes son usadas para su validación.
- (c) Serie INTERNET: Es la cantidad de usuarios que acceden a un servidor de internet por minuto durante 100 minutos consecutivos. Los primeros 80 datos son usados para la estimación del modelo, y los 20 restantes para su predicción. Se modela la serie original sin ningún tipo de transformación.
- (d) Serie LYNX: Corresponde al número de lince canadienses atrapados por año en el distrito del río Mckenzie del norte de Canadá entre los años 1821 y 1934. Se modela el logaritmo en base 10 de la serie original. Las primeras 100 observaciones son usadas para el ajuste de los modelos y las 14 restantes para la predicción por fuera de la muestra de calibración.
- (e) Serie PAPER: Corresponde a las ventas mensuales de papel impreso y escrito en miles de francos franceses entre ENE1963 y DIC1972. Las primeras 100 observaciones son usadas para el ajuste de los modelos y las 20 restantes para la predicción por fuera de la muestra de calibración. La serie es modelada sin ninguna transformación.

### Serie AIRLINE

En [8] se comparó la precisión de varios modelos para pronosticar esta serie; en dicha referencia se indica que el mejor modelo ARIMA para representar la serie es de orden  $(0, 1, 1) \times (0, 1, 1)_{12}$ ; sus resultados se presentan en la Tabla 2. Igualmente en [8] se estimaron diferentes MLP que difieren en la cantidad de neuronas de la capa oculta y los rezagos de la serie usados como regresores de la serie; el MLP con menor error de entrenamiento tiene 4 neuronas en la capa oculta y usa los rezagos 1-13; el MLP con menor error de validación tiene 2 neuronas en la capa oculta y usa los rezagos 1 y 12. En la Tabla 2 se reportan los resultados para estos dos modelos.

En esta investigación se estimaron varias SVM con los mismos rezagos usados en [8]. Para cada uno de los SVM se aplicó el proceso de optimización de  $(\sigma, C, \epsilon)$  descrito previamente; en la Tabla 2 se reportan los resultados obtenidos.

Todas las SVM estimadas tienen errores de entrenamiento menores que el obtenido en [8] para el modelo ARIMA, excepto para la SVM-7 con rezagos 1, 2 y 13; todas las SVM estimadas tienen un error de validación inferior al del modelo ARIMA. Para la mejor SVM, presenta una reducción del 82% y del 100% de los errores de entrenamiento y validación en relación a los del modelo ARIMA.

Respecto al MLP con mejor ajuste, la SVM con mejor ajuste presenta una reducción del error de entrenamiento del 23%, pero del 100% respecto al error de validación. Se observa igualmente que el MLP con menor error de validación se encuentra bastante lejano en desempeño respecto a las SVM estimadas.

En la Figura 2 se presenta la predicción un mes adelante para la SVM-3 de la Tabla 2. Los puntos negros representan los datos reales. La línea continua con puntos negros representa la predicción del modelo; las otras líneas representan los intervalos de confianza del 95% para la predicción. La línea punteada vertical del extremo derecho representa la división entre el conjunto de datos de ajuste y el conjunto de datos de predicción.

### Serie POLLUTION

Esta serie fue originalmente utilizada en [33], y se indica que un modelo ARIMA de orden  $(2,1,0)(1,0,0)_{12}$  es adecuado para representar su dinámica. En [32] se reportan los resultados obtenidos al estimar dicho modelo para las primeras 106 observaciones y validar su desempeño para las siguientes 12 y 24 observaciones, respectivamente. De igual manera en [32] se reportan los resultados obtenidos al estimar un MLP que usa como entradas los rezagos 1-12. Los estadísticos para ambos modelos son resumidos en la Tabla 3.

En esta investigación se estimaron SVM para los mismos grupos de rezagos sugeridos en [32]; los resultados presentados en la Tabla 3 muestran que todos los SVM presentan un MSE de entrenamiento igual o inferior al reportado para los modelos MLP y ARIMA, y una MAD siempre inferior. Para la predicción un año adelante, el modelo ARIMA presenta el MSE más bajo, pero este aumenta considerablemente al considerar un horizonte de predicción de 24 meses. Las SVM presentan un desempeño inferior al del modelo ARIMA para la predicción un año

adelante, pero superan claramente a los demás modelos para el horizonte de 2 años. El modelo SVM-1 de la Tabla 3 presenta el mejor desempeño global entre todos los modelos considerados; su predicción un mes adelante es presentada en la Figura 3.

### Serie INTERNET

Esta serie es analizada en [33], y se sugiere que un modelo ARIMA de orden (3,1,0) sería el más adecuado para representar su dinámica. En [32] se reportan los resultados obtenidos al estimar un MLP con rezagos 1, 2, 3 y 4 para dicha serie. Los resultados reportados en [32] son resumidos en la Tabla 4.

En este caso se demuestra que las SVM no siempre pueden ser la mejor alternativa respecto a los modelos ARIMA y los MLP. Para este caso, la SVM presenta un MSE de entrenamiento menor que el modelo ARIMA pero superior al del MLP (Tabla 4); en términos de la MAD, la SVM estimada es inferior en calidad respecto a los dos modelos competidores. Para la predicción por fuera de la muestra de calibración, la SVM presenta un pobre desempeño respecto a los otros modelos considerados; su MSE y su MAD son el 200% y el 154% de los obtenidos para el modelo ARIMA. La predicción para el modelo SVM es presentada en la Figura 4.

El proceso de optimización fue realizado muchas veces, y en ningún caso fue posible obtener estadísticos que fueran inferiores a los reportados para los modelos ARIMA y MLP.

### Serie LYNX

Esta serie es usada en [9] para evaluar el desempeño de un modelo ARIMA, un MLP y un modelo híbrido que combina las dos metodologías anteriores; en [9] se reporta que el MLP tiene una estructura de 7 entradas correspondientes a los primeros 7 rezagos de la serie, 5 neuronas en la capa oculta y una neurona de salida. Igualmente, se señala que el modelo ARIMA tiene el mejor desempeño de los tres modelos considerados.

Ya que en [9] no se reportan los rezagos utilizados para especificar el modelo ARIMA, se consideraron 7 SVM que difieren en los rezagos utilizados por cada modelo. Los estadísticos de ajuste a las muestras de entrenamiento y predicción para las SVM y los modelos reportados en [9] son presentados en la Tabla 5. Para esta serie, el modelo SVM-6, que usa como entradas los rezagos 1 al 3 y 8 al 10, presenta unos estadísticos de ajuste a la muestra de predicción que son inferiores a los reportados para el modelo híbrido en [9]. Esto se presenta en una reducción del 11% y del 22% para el MSE y el MAD, respectivamente. La predicción un mes adelante obtenida con la SVM-6 es presentada en la Figura 5.

### Serie PAPER

En [33] se reporta que el modelo ARIMA con menor MSE para representar la dinámica de la serie es de orden  $(0, 1, 1)(0, 1, 1)_{12}$  usando la totalidad de la información disponible. En [32] este modelo es reestimado usando los primeros 100 datos para el ajuste del modelo y los 20 restantes para la predicción por fuera de la muestra de ajuste. Igualmente en [32] se reportan los resultados obtenidos al estimar un MLP que usa los rezagos 1, 7 y 12 como entradas. Los estadísticos para estos modelos son resumidos en la Tabla 6.

En esta investigación se estimaron 2 SVM que usan los mismos rezagos de los modelos ARIMA (1, 12 y 13) y MLP (1, 7 y 12). Para este caso, los estadísticos de ajuste a la muestra de calibración muestran que las SVM presentan errores (MSE y MAD) más altos que los modelos ARIMA y MLP; esto es, incrementos del 36% y del 14% para el MSE y el MAD respecto a los reportados para el modelo ARIMA. En términos de la predicción por fuera de la muestra de ajuste, los SVM presentan mejores estadísticos que el MLP, pero peores que los del modelo ARIMA. Los SVM presentan un MSE y un MAD que son superiores al menos un 15% y un 11%, respectivamente, en relación a los reportados para el modelo ARIMA. En la Figura 6 se presenta la predicción un mes adelante para la SVM-2.

## CONCLUSIONES

El uso de las SVM para la predicción de series de tiempo se ve dificultado por la necesidad de escoger adecuadamente los valores de las constantes ( $\sigma$ ,  $C$ ,  $\epsilon$ ) que controlan el desempeño del modelo final obtenido. En este artículo se ha propuesto una metodología para fijar el valor inicial de dichas constantes, y una metodología para su optimización con el fin de obtener el mejor SVM posible para pronosticar la serie particular estudiada.

La metodología propuesta fue utilizada para construir SVM para cinco series que han sido comúnmente utilizadas como *benchmark* en la comparación de modelos no lineales de predicción. Igualmente, se compararon los resultados aquí obtenidos con los reportados por varios investigadores que pronosticaron las mismas series utilizando modelos ARIMA y perceptrones multicapa. Contrariamente a lo reportado en otras investigaciones [25-26, 28-29], las SVM sólo fueron superiores en desempeño respecto a los modelos ARIMA y los MLP, en 3 de 5 casos.

Como trabajo futuro debe investigarse el desempeño de SVM con otras funciones de núcleo, así como también otras metodologías para la especificación de las constantes que controlan el desempeño del modelo.



Tabla 2. Sumatoria del error cuadrático (SSE). SVM con kernel gaussiano. Serie AIRLINE.

Modelo	Rezagos	C	$\varepsilon$	$\sigma$	Entrenamiento SSE	Predicción SSE
ARIMA	1, 12, 13				1,08	0,43
MLP con mejor ajuste	1-13				0,26	1,12
MLP con mejor validación	1, 12				2,30	0,34
SVM-1	1-4	268	0,05	1,0	0,88	0,13
SVM-2	1-13	363	0,04	10,5	0,20	0,00
SVM-3	1,12	348	0,02	1,0	0,34	0,01
SVM-4	1,2,12	346	0,02	3,8	0,35	0,03
SVM-5	1,2,12,13	878	0,04	0,6	0,18	0,20
SVM-6	1,12,13	518	0,03	20,0	0,21	0,02
SVM-7	1,2,13	343	0,09	6,7	1,21	0,13
SVM-8	1,4,8,12	562	0,09	3,0	0,36	0,05
SVM-9	1,4,8,12,13	562	0,08	2,3	0,26	0,04
SVM-10	1,4,8,10,12,13	557	0,08	2,4	0,26	0,05

Tabla 3. Error cuadrático medio (MSE) y desviación media absoluta (MAD). SVM con kernel gaussiano. Serie POLLUTION.

Modelo	Rezagos	C	$\varepsilon$	$\sigma$	Entrenamiento MSE (MAD)	Predicción MSE (MAD) 1 año	Predicción MSE (MAD) 2 años
ARIMA	1-3, 12-15				0,052 (0,181)	0,025 (0,135)	0,268 (0,395)
MLP	1-12				0,054 (0,188)	0,193 (0,394)	0,146 (0,334)
SVM-1	1-3, 12-15	220	0,08	32	0,052 (0,176)	0,031 (0,159)	0,042 (0,175)
SVM-2	1-12	1.100	0,10	28	0,049 (0,174)	0,039 (0,159)	0,080 (0,221)

Tabla 4. Error cuadrático medio (MSE) y desviación media absoluta (MAD). SVM con kernel gaussiano. Serie INTERNET.

Modelo	Rezagos	C	$\varepsilon$	$\sigma$	Entrenamiento MSE (MAD)	Predicción MSE (MAD)
ARIMA	1-4				9,76 (2,42)	8,11 (2,23)
MLP	1-4				7,00 (2,10)	9,25 (2,25)
SVM	1-4	5.900	4,7	140	9,30 (2,50)	16,58 (3,44)

Tabla 5. Error cuadrático medio (MSE) y desviación media absoluta (MAD). SVM con kernel gaussiano. Serie LYNX.

Modelo	Rezagos	C	$\varepsilon$	$\sigma$	Entrenamiento MSE (MAD)	Predicción MSE (MAD)
ARIMA	N/D				N/D	0,021 (0,112)
MLP	N/D				N/D	0,021 (0,112)
HÍBRIDO	1-7				N/D	0,017 (0,104)
SVM-1	1-6	261	0,2	3,7	0,036 (0,161)	0,021 (0,111)
SVM-2	1-7	731	0,2	4,8	0,034 (0,160)	0,019 (0,121)
SVM-3	1-8	477	0,2	5,0	0,034 (0,155)	0,020 (0,125)
SVM-4	1-9	444	0,2	2,8	0,026 (0,140)	0,036 (1,163)
SVM-5	1-10	549	0,2	4,4	0,031 (0,157)	0,035 (0,163)
SVM-6	1-3, 8-10	587	0,2	4,0	0,034 (0,152)	0,015 (0,087)
SVM-7	1-4, 8-10	468	0,2	6,9	0,038 (0,160)	0,021 (0,122)

Tabla 6. Error cuadrático medio (MSE) y desviación media absoluta (MAD). SVM con kernel gaussiano. Serie PAPER.

Modelo	Rezagos	C	$\varepsilon$	$\sigma$	Entrenamiento MSE (MAD)	Predicción MSE (MAD)
ARIMA	1, 12, 13				1715,5 (34,04)	4791,7 (54,72)
MLP	1, 7, 12				1951,5 (36,79)	5874,8 (62,62)
SVM-1	1, 12, 13	5.254	80	1.529	2335,8 (38,7)	5572,8 (60,6)
SVM-2	1, 7, 12	5.223	79	1.334	2354,8 (38,1)	5526,4 (61,0)

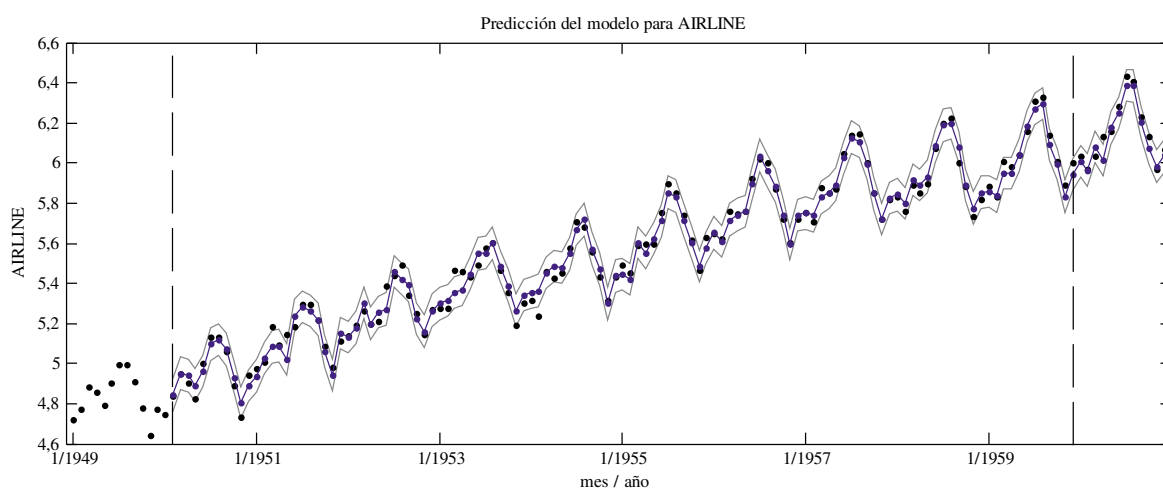


Figura 2. Predicción para la serie AIRLINE usando el modelo SVM-2 de la Tabla 2.

Fuente: Elaboración propia.

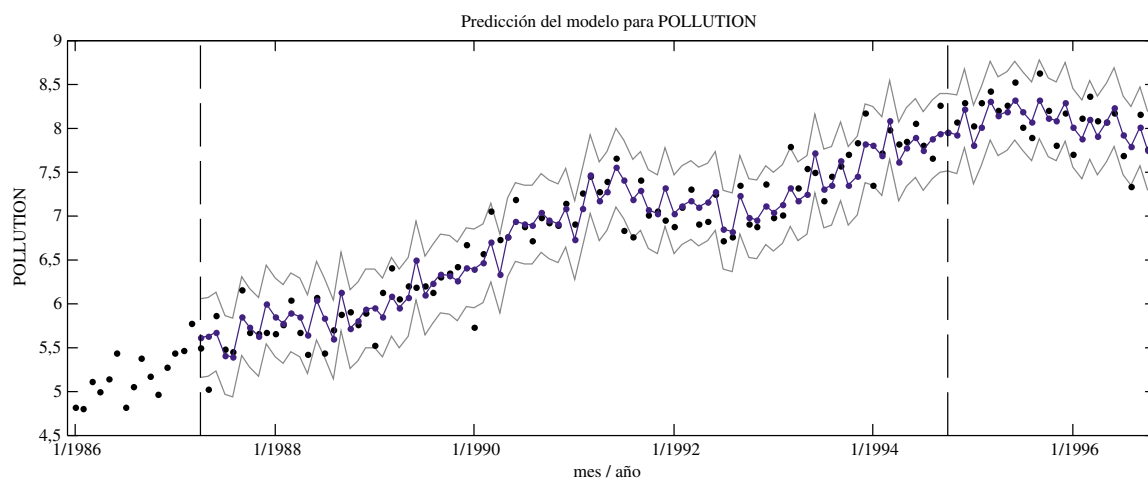


Figura 3. Predicción para la serie POLLUTION usando el modelo SVM-1 de la Tabla 3.

Fuente: Elaboración propia.

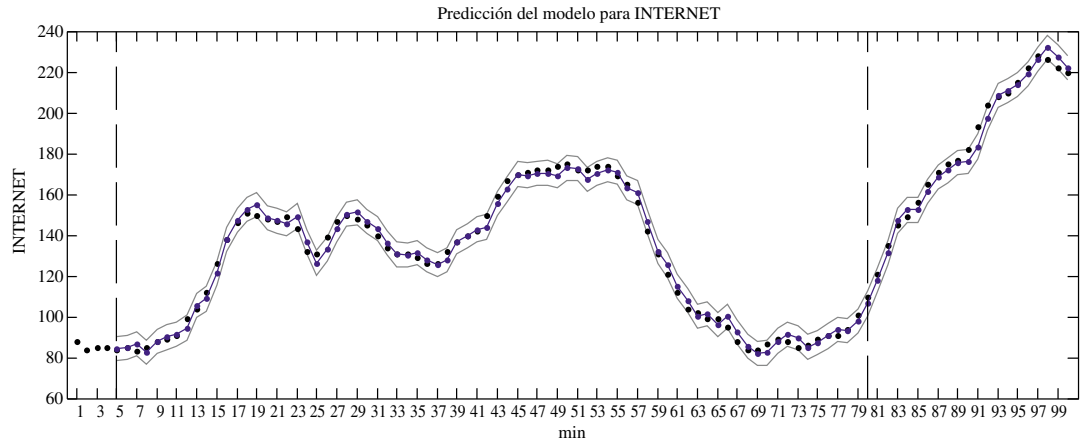


Figura 4. Predicción para la serie INTERNET usando el modelo SVM de la Tabla 4.  
Fuente: Elaboración propia

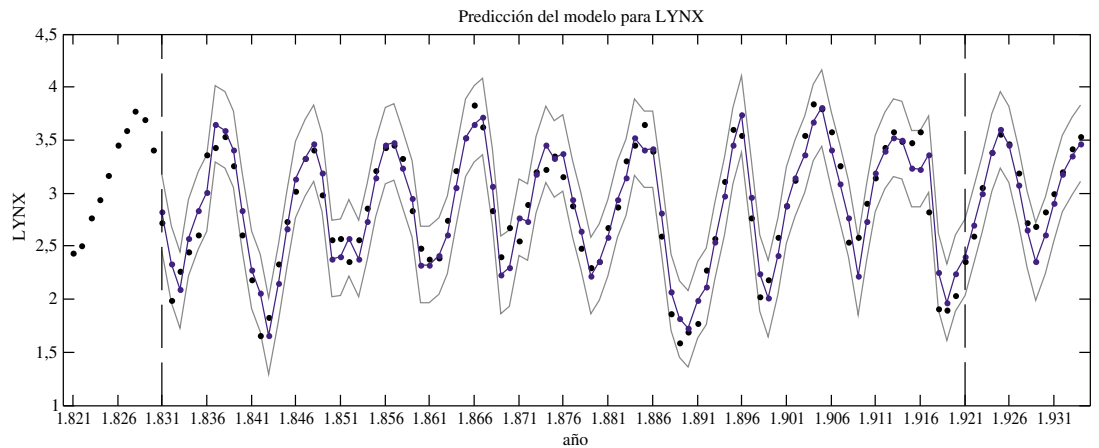


Figura 5. Predicción para la serie LYNX usando el modelo SVM-6 de la Tabla 5.  
Fuente: Elaboración propia.

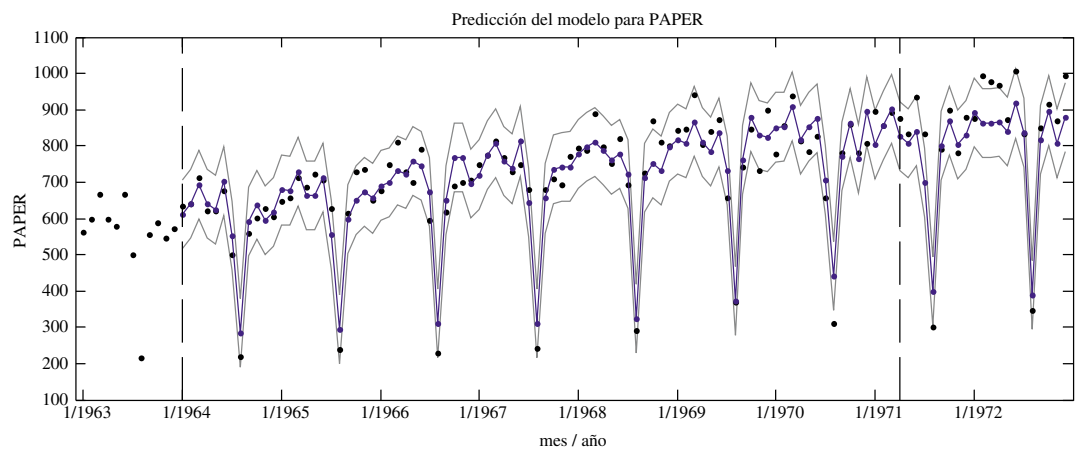


Figura 6. Predicción para la serie PAPER usando el modelo SVM-2 de la Tabla 6.  
Fuente: Elaboración propia.

## REFERENCIAS

- [1] H. Tong. "Non-linear time series a dynamical system approach". Claredon Press Oxford. 1990.
- [2] G.E.P. Box and G.M. Jenkins. "Time Series Analysis: Forecasting and Control". Holden-Day Inc. 1970.
- [3] C. Granger and T. Teräsvirta. "Modeling Nonlinear Economic Relationships". Oxford University Press. 1993.
- [4] G. Zhang, B. Patuwo and M. Hu. "Forecasting with artificial neural networks: the state of the art". International Journal of Forecasting. Vol. 14, pp. 35-62. 1998.
- [5] S. Heravi, D. Osborn and C. Birchenhall. "Linear versus neural network forecasts for european industrial production series". International Journal of Forecasting. Vol. 20, pp. 435-446. 2004.
- [6] N. Swanson and H. White. "Forecasting economic time series using adaptive versus nonadaptive and linear versus non-linear econometric models". International Journal of Forecasting, Vol. 13, pp. 439-461. 1997a.
- [7] N. Swanson and H. White. "A model selection approach to real time macroeconomic forecasting using linear models and artificial neural networks". Review of Economics and Statistics. Vol. 39, pp. 540-550. 1997b.
- [8] J. Faraway and C. Chatfield. "Time series forecasting with neural networks: A comparative study using the airline data". Applied Statistics. Vol. 47 N° 2, pp. 231-250. 1998.
- [9] G. Zhang. "Time Series forecasting using a hybrid ARIMA and neural network model". Neurocomputing. Vol. 50, pp. 159-175. 2003.
- [10] M. Cottrell, B. Girard, Y. Girard, M. Mangeas and C. Muller. "Neural modeling for time series: A statistical stepwise method for weight elimination". IEEE Transactions on Neural Networks. Vol. 6 N° 6, pp. 1355-1364. 1995.
- [11] C. de Groot and D. Wurtz. "Analysis of univariate time series with connectionist nets: A case study of two classical examples". Neurocomputing. Vol. 3, Issue 4, pp. 177-192. 1991.
- [12] G. Darbellay and M. Slama. "Forecasting the short-term demand for electricity. Do neural networks stand a better chance?". International Journal of Forecasting. Vol. 16, Issue 1, pp. 71-83. 2000.
- [13] C. Kuan and T. Liu. "Forecasting exchange rates using feedforward and recurrent neural networks". Journal of Applied Econometrics. Vol. 10, pp. 347-364. 1995.
- [14] K. Hornik, M. Stinchcombe and H. White. "Multilayer feedforward networks are universal approximators". Neural Networks. Vol. 2, pp. 359-366. 1989.
- [15] G. Cybenko. "Approximation by superpositions of a sigmoidal function". Mathematics of Control: Signals and Systems. Vol. 2, pp. 202-314. 1989.
- [16] K. Funahashi. "On the approximate realization of continuous mappings by neural networks". Neural Networks. Vol. 2, pp. 183-192. 1989.
- [17] T. Masters. "Practical Neural Network Recipes in C++". Academic Press, Inc. First edition. 1993.
- [18] T. Masters. "Neural, Novel and Hybrid Algorithms for Time Series Prediction". John Wiley and Sons, Inc. First Edn. 1995.
- [19] I. Kaastra and M. Boyd. "Designing a neural network for forecasting financial and economic series". Neurocomputing. Vol. 10, pp. 215-236. 1996.
- [20] V.N. Vapnik. "The Nature of Statistical Learning Theory". Springer. N.Y. 1995.
- [21] V.N. Vapnik, S.E. Golowich and A.J. Smola. "Support vector method for function approximation, regression estimation, and signal processing". Advances in Neural Information Processing Systems. Vol. 9, pp. 281-287. 1996.
- [22] C.J.C. Burges. "A tutorial on support vector machines for pattern recognition". Data Mining and Knowledge Discovery. Vol. 2 N° 2, pp. 121-167. 1998.
- [23] A.I. Belousov, S.A. Verzakov and J. Von Frese. "A flexible classification approach with optimal generalisation performance: support vector machines". Chemometrics and Intelligent Laboratory Systems. Vol. 64, pp. 15-25. 2002.
- [24] A.J. Smola and B. Schölkopf. "A tutorial on support vector regression". NeuroCOLT Technical Report NC-TR-98-030. Royal Holloway College, University of London. UK. 1998.
- [25] S. Mukherjee, E. Osuna, and F. Girosi. "Nonlinear prediction of chaotic time series using support vector machines". Proceedings of IEEE NNSP'97. Amelia Island, FL. 1997.
- [26] K.R. Müller, A.J. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, and V. Vapnik. "Predicting time series with support vector machines". In: W. Gerstner, A. Germond, M. Hasler and J.-D. Nicoud (Eds.). Proceedings of ICANN '97. Springer LNCS 1327, pp. 999-1004. Berlin. 1997.
- [27] S. Osowski and K. Garanty. "Forecasting of the daily meteorological pollution using wavelets and support vector machine". Engineering Applications

- of Artificial Intelligence. Vol. 20, Issue 6, pp. 745-755. September 2007.
- [28] F.E.H Tay and L. Cao. "Application of support vector machines in financial time series forecasting". Omega, Vol. 29 N° 4, pp. 309-317. 2001.
- [29] U. Thissen, R. Van Brakel, A.P. de Weijer, W.J. Melssen and L.M.C. Buydens. "Using support vector machines for time series prediction". Chemometrics and Intelligent Laboratory Systems. Vol. 69, Issues 1-2, pp. 35-49. 28 November 2003.
- [30] V.N. Vapnik. "Statistical Learning Theory". Wiley. New York, USA. 1998.
- [31] B. Scholkopf and A.J. Smola. "Learning with Kernels". MIT Press. Cambridge, MA. 2002.
- [32] M. Ghiassi, H. Saidane and D.K. Zimbra. "A dynamic artificial neural network model for forecasting time series events". International Journal of Forecasting. Vol. 21, pp. 341-362. 2005.
- [33] S.G. Makridakis, S.C. Wheelwright and R.J. Hyndman. "Forecasting: Methods and applications". John Wiley & Sons. 3rd edition. New York, USA. 1998.
- [34] L. Platt. "Fast training of SVM using sequential optimization". In: B. Scholkopf, B. Burges and A.J. Smola (Eds.), Advances in Kernel Methods-Support Vector Learning. MIT Press, pp. 185-208. Cambridge. 1998.
- [35] S.R. Gunn. "Support vector machines for classification and regression". Technical report, Image speech and intelligent systems research group. University of Southampton. UK. 1997.
- [36] V. Fernández. "Wavelet- and SVM-based forecasts: An analysis of the U.S. metal and materials manufacturing industry". Resources Policy. Vol 32 N° 1-2, pp. 80-89. March-June 2007.
- [37] P.F. Pai and W.C. Hong. "Forecasting regional electric load based on recurrent support vector machines with genetic algorithms". Electric Power Systems Research. Vol. 74 N° 3, pp. 417-425. June 2005.
- [38] X. Guo, L. Sun, G. Li and S. Wang. "A hybrid wavelet analysis and support vector machines in forecasting development of manufacturing". Expert Systems with Applications. Vol. 35 N° 1-2, pp. 415-422. July-August 2008.
- [39] A.A. Levis and L.G. Papageorgiou. "Customer Demand Forecasting via Support Vector Regression Analysis". Chemical Engineering Research and Design. Vol. 83, Issue 8, pp. 1009-1018. August 2005.
- [40] P.F. Pai and C.S. Lin. "A hybrid ARIMA and support vector machines model in stock price forecasting". Omega. Vol. 33, Issue 6, pp. 497-505. December 2005.
- [41] K.J. Kim. "Financial time series forecasting using support vector machines". Neurocomputing. Vol. 55 N° 1-2, pp. 307-319. September 2003.
- [42] K.Y. Chen and C.H. Wang. "Support vector regression with genetic algorithms in forecasting tourism demand". Tourism Management. Vol. 28, Issue 1, pp. 215-226. February 2007.