Revista Iberoamericana de Automática e Informática Industrial
CEA comité español de automática
RIAI
ISSN: 1697-7912. Vol. 6, Núm. 1, Enero 2009, pp. 94-104
http://www.revista-riai.org

Selección de variables en la predicción de llamadas en un centro de atención telefónica

Manuel R. Arahal * **Manuel Berenguel** ** **Eduardo F. Camacho** *
Fernando Pavón *

* *Dpto. de Ingeniería de Sistemas y Automática, Universidad de Sevilla. Camino Descubrimientos, s/n. 41092. España, (dt: arahal@esi.us.es)*
** *Dpto. de Lenguajes y Computación, Universidad de Almería. La Cañada de San Urbano, s/n. 04120. España.*

Resumen: En este artículo se ilustra la importancia de la selección de variables independientes para modelos neuronales destinados a la predicción de la demanda en un centro de atención telefónica. Los modelos tienen como objetivo ayudar en la planificación semanal del personal del centro, tarea que se realiza con 14 días de antelación.
Los modelos requeridos pueden hacer uso de gran cantidad de variables independientes. Sin embargo, el número de casos que pueden ser usados para obtener los parámetros del modelo es escaso debido a los cambios socio-económicos. Esto plantea la necesidad de seleccionar cuidadosamente las variables independientes y utilizar el menor número posible de ellas, de otro modo la generalización del modelo se degradaría.
Para resolver el problema se utiliza un método mixto que permite trabajar con un alto número de variables candidatas, en una primera fase, y seleccionar más cuidadosamente un número menor de variables en una segunda fase. Los resultados obtenidos por los modelos resultantes de aplicar el método propuesto y sus variantes son analizados utilizando datos reales de un centro de atención telefónica. Los resultados de la comparación muestran que la correcta selección de variables independientes es vital para este tipo de aplicación.

Palabras Clave: Modelos, Predicción, Redes de neuronas artificiales

1. INTRODUCCIÓN

Los centros de atención telefónica (CAT) han experimentado un gran auge en la última década como consecuencia de factores tecnológicos y económicos. Son muchas las empresas que utilizan este tipo de centros para vender sus servicios o para obtener información de sus clientes.

La capacidad de atención de un centro depende fundamentalmente de la cantidad de personas que atienden las llamadas. La calidad del servicio es un aspecto importante que suele medirse considerando la cantidad de llamadas perdidas. Dado que los costes de personal son altos, la calidad del servicio se contrapone a la obtención de beneficios (Pinedo *et al.*, 1999). La planificación del personal (estática y dinámica) es un medio para llegar a un compromiso entre costes y servicio. El método más usado consiste en simular el comportamiento dinámico del CAT mediante la teoría de colas (Koole y Mandelbaum, 2002).

El número de llamadas cada hora constituye la carga del CAT y es la variable más importante para determinar el número de operadores necesarios para proporcionar una determinada calidad de servicio. En la mayoría de los casos la predicción se realiza de forma no automática, por lo que existe una dependencia del concurso de un experto humano. Las técnicas de predicción de secuencias temporales pueden aplicarse en este contexto y cabría esperar una mejora de resultados. Sin embargo existen pocos informes en la literatura acerca de aplicaciones reales de estas técnicas para CAT. Por ejemplo, en (Sze, 1984) se utilizan métodos clásicos simples, en (Andrews y Cunningham, 1995) se proponen modelos ARIMA y en (Jongbloed y Koole, 2001; Antipov y Meade, 2002; Avramidis *et al.*, 2004) métodos estadísticos para modelar la tasa de llegada de llamadas.

En este artículo se propone el uso de modelos realizados por redes de neuronas artificiales. Este tipo de modelos ha sido usado frecuentemente para la predicción de secuencias temporales, pudiendo citarse desde aplicaciones tempranas (Werbos, 1988) hasta libros de texto recientes (Bishop, 2006). El problema que surge al aplicar estos modelos es el de la generalización de las predicciones. Este problema se ve agravado por el hecho de que la secuencia a predecir cambia debido a factores socio-económicos. Por ello el número de observaciones relevantes de la secuencia se restringe al subconjunto formado por las más recientes. Esto quiere decir que se tienen pocos datos para el entrenamiento de la red y por tanto es necesario seleccionar cuidadosamente las variables independientes que constituyen el vector de entrada de la red de neuronas. Este problema es el que motiva el presente trabajo.

En primer lugar se va a plantear de una forma más detallada el problema de predicción en el CAT, indicando los aspectos más relevantes de la secuencia temporal que ha servido para obtener los resultados en este estudio. Posteriormente se mostrará el tipo de modelo utilizado y el método propuesto para la selección de variables independientes. Finalmente, los resultados obtenidos por distintos métodos serán comparados lo cual dará lugar a las conclusiones del trabajo.

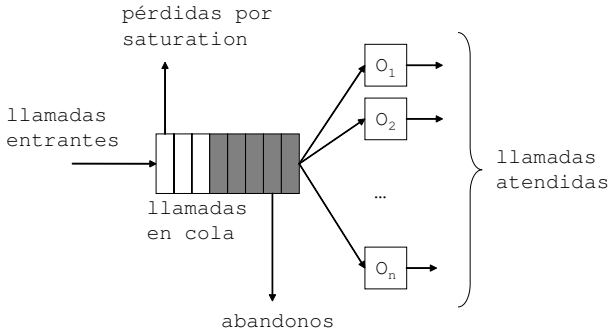

Figura 1. Diagrama simplificado de un CAT mostrando el flujo de llamadas.

2. LA PREDICCIÓN EN EL CAT

La figura 1 muestra el diagrama simplificado de un CAT, indicando mediante flechas el flujo de llamadas. El rectángulo en la parte central simboliza la cola de llamadas, formada por los clientes que esperan a ser antendidos. Los cuadrados simbolizan los operadores que atienden las llamadas, sacándolas de la cola. Las llamadas recibidas en el CAT son transferidas a un operador u otro dependiendo de ciertas variables: idioma, área de donde proviene, tipo de información requerida, etc. Téngase en cuenta que el diagrama mostrado en la figura 1 ha sido simplificado pues las llamadas puede ser transferidas de un operador a otro hasta finalmente ser atendidas. Durante el proceso puede ocurrir que la llamada se pierda debido a:

- 1. Espera excesiva. El cliente cuelga por impaciencia. Este tipo de situaciones afecta negativamente y es lo que se intenta evitar mediante una planificación de operadores adecuada.
- 2. Saturación. Las líneas del CAT no permiten nuevas llamadas. En este caso el problema no es de la planificación de operadores sino de diseño del CAT, por lo que cae fuera del ámbito de este estudio.
- 3. Problemas técnicos. Al igual que en el caso anterior esta situación no depende del número de operadores disponibles y por tanto no se tiene en cuenta.

Es fácil de entender que la calidad del servicio desde el punto de vista del cliente depende de la cualificación profesional del operador que presta el servicio y del número de operadores disponibles. La preparación del operador es un asunto que no concierne para este estudio por lo que se dejará a un lado. Por otra parte, el número de operadores está reñido con el beneficio del CAT pues a mayor número de operadores mayor gasto. La dirección del CAT debe llegar a una solución de compromiso entre tener muchos operarios (con el peligro de que a ciertas horas estén ociosos) o tener demasiado pocos con lo que los tiempos de espera aumentarán y con ello la insatisfacción de los clientes. Queda claro pues que conviene adecuar la oferta de operadores a la demanda y para ello es vital tener una buena previsión del número de llamadas.

El número de llamadas entrantes se denomina carga o volumen horario y se va a denotar como x, por tanto $x(k)$ es el número de llamadas recibidas en la hora k. Esta variable es la más importante para determinar el número de operadores necesarios para proporcionar una determinada calidad de servicio (véase Gans *et al.* (2003) donde se presenta una introducción al problema de la operación de los CAT). La forma más extendida de atacar el problema de predicción consiste en modelar la tasa de llamadas

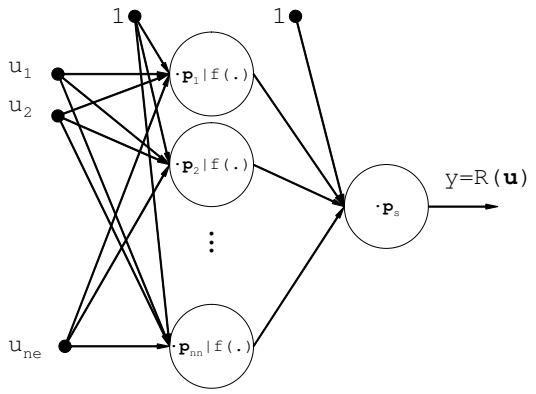

Figura 2. Estructura de la red de neuronas artificiales utilizada.

diaria mediante un proceso estadístico. Esta tasa junto a un modelo de la cola permite simular el comportamiento dinámico del CAT. La simulación permite luego determinar políticas de gestión de operadores.

Las técnicas de predicción de secuencias temporales pueden aplicarse en este contexto y cabría esperar una mejora de resultados. Sin embargo existen pocos informes en la literatura acerca de aplicaciones reales de estas técnicas para CAT. Por ejemplo, en (Sze, 1984) se utilizan métodos clásicos simples, en (Andrews y Cunningham, 1995) se proponen modelos ARIMA en (Shen y Huang, 2008) se combina la descomposición en valores singulares con un modelo ARIMA.

En este trabajo se propone el uso de redes de neuronas y se ataca el problema de determinar la estructura más adecuada. En este contexto resulta que el paso crítico consiste en decidir qué variables se van a usar como parte del vector de entrada de la red de neuronas. Se propone un algoritmo de selección en dos pasos que ha sido especialmente pensado para este tipo de aplicación. A continuación se muestra la forma en que los modelos de redes de neuronas artificiales han sido usados para la predicción del número de llamadas entrantes en el CAT.

2.1 Modelo neuronal

Los modelos usados son realizados por redes de neuronas artificiales con la estructura mostrada en el diagrama de la figura 2. La salida de la red será la predicción de la carga, obtenida como $\hat{x}(t) = \mathrm{R}(\mathbf{u})$, siendo $\mathbf{u}$ el vector de entrada que contiene las variables independientes del modelo y siendo $\mathrm{R}()$ una función continua y derivable realizada por la red de neuronas artificiales. Se utilizan redes estáticas (acíclicas) de una sola capa por lo que la aplicación de $\mathbf{u}$ en $\hat{x}$ es estática. Como es sabido las redes de neuronas artificiales pueden describirse como suma truncada de funciones base (Valverde y Gachet, 2007). La utilización de unas bases u otras carece de importancia siempre y cuando la suma resultante posea suficiente flexibilidad para acomodar las observaciones. En este caso se han utilizado bases del tipo $f(.) = \frac{1}{1+e^{-\cdot}}$ por lo que las redes resultantes son del tipo perceptrón.

El número de capas ocultas de la red se ha limitado a uno pues con eso basta para garantizar la propiedad de aproximación universal. De este modo el único parámetro estructural es el número de nodos en la capa oculta n_n, el cual define la flexibilidad de la red. Los valores de los vectores de pesos $\mathbf{p_i}$ para cada nodo $i \in 1, 2, \cdots, n_n$ se obtendrá mediante el apropiado entrenamiento usando valores pasados conocidos de

la carga. Una vez realizado el entrenamiento la predicción se calculará mediante la expresión $\hat{x}(t) = \mathbf{p_s} \cdot \mathbf{o}(\mathbf{u}(t))$, siendo $\mathbf{u}$, el vector de entrada, $\mathbf{o}$ el vector formado por las salidas de los nodos ocultos y $\mathbf{p_s}$ el vector de pesos del nodo de salida. Por su parte, la salida de cada nodo oculto se calcula mediante:

$$o_i(\mathbf{u}) = f(\mathbf{p_i} \cdot \mathbf{u}) + p_i^0 \quad (1)$$

para $i \in 1, 2, \cdots, n_n$ y siendo $o_0 = 1$ un valor fijo para permitir a la red acomodar valores de continua.

Es vital tener en cuenta que el número de parámetros ajustables de la red es $n_p = (n_u + 1) \cdot n_n + n_n + 1$, siendo $n_u = \dim(\mathbf{u})$ la dimensión del vector de entrada y n_n el número de nodos en la capa oculta. Conviene que n_p sea mucho menor que el número de observaciones disponibles para el entrenamiento de la red $N_E = \mathrm{card}(E)$. Por lo tanto existe un límite al número de parámetros de otro modo se corre el riesgo de perder la capacidad de generalización (Bishop, 2006). Ese límite se ve trasladado al número de nodos de la capa oculta y a la dimensión del vector de entrada.

Sin embargo N_E es posiblemente un número no muy alto debido a que los cambios socio-económicos afectan a la secuencia $\{x(k)\}$, reduciendo el horizonte de observaciones relevantes. En el caso de la aplicación al CAT el conjunto H se ha limitado a un año y medio pues los datos más antiguos resultan obsoletos. De este modo N_E resulta valer $365 \cdot 1{,}5 \cdot 24/2 = 6570$. Este hecho conlleva que sea necesaria una selección del número de variables que conforman el vector de entrada $\mathbf{u}$ y del número de nodos en la capa oculta n_n. Dicha selección constituye el principal objetivo del presente trabajo.

Otra consecuencia importante es que el proceso de construcción de modelos ha de repetirse periódicamente para así actualizar las predicciones a los citados cambios socio-económicos.

2.2 Características de la carga

Se dispone de unos archivos históricos que proporcionan el valor de la carga durante varios años, siendo $k = 1$ la hora de la primera anotación del archivo histórico y $k = N$ la hora de la última o más reciente. Por tanto, la secuencia de datos es $\{x(k)\}$, con $k = 1, \cdots, N$. Ha de tenerse en cuenta sin embargo que, debido a los cambios socio-económicos, solamente los valores más recientes de $\{x(k)\}$ tienen relevancia para el modelado.

Es importante tomar en consideración las características del problema a fin de proporcionar la solución más adecuada. En primer lugar conviene aclarar que el objetivo será predecir el número de llamadas entrantes en el CAT cada hora $x(k)$. Esta predicción ha de hacerse con cierta antelación, de forma que se pueda generar la planificación de personal que más interese teniendo en cuenta el dilema calidad/precio. El avance en la predicción ha de ser de al menos una semana. Esto quiere decir que si el día D en que se calcula la predicción es un lunes entonces ha de generarse la previsión para el siguiente lunes $D{+}7$, para el martes $D + 7$ y así sucesivamente hasta el domingo de la siguiente semana $D + 13$. Para simplificar el problema es mejor considerar simplemente el máximo horizonte. En este caso se ha tomado un horizonte de 14 días. Por tanto el objetivo es producir $\hat{x}(D + 14 \cdot 24 + h)$ para $h = 1, \cdots, 24$.

Conviene representar gráficamente los valores de la carga $x(k)$ para poner de manifiesto sus características, algunas de las cuales servirán de guía durante el proceso de obtención de modelos. En la Fig. 3 se muestra la carga horaria observada en una semana. Puede verse la periodicidad de la carga consistente en ciclos de 24 horas englobados dentro de ciclos de siete días. Esta misma periodicidad se observa en el periodograma de la Fig. 4. Finalmente, la gráfica de la Fig. 5 muestra la carga media calculada de forma separada para cada día de la semana. El perfil horario de lunes, sábados y domingos se distingue claramente del perfil del resto de días.

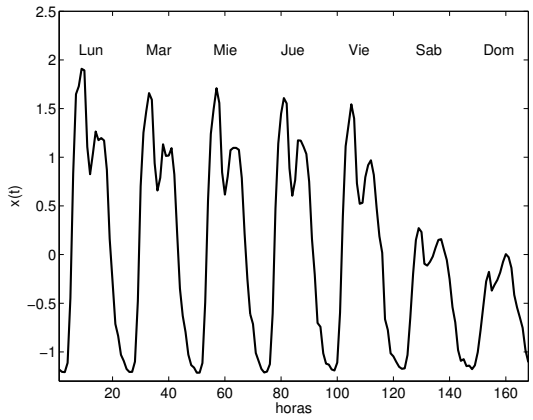

Figura 3. Número de llamadas entrantes por hora en un CAT (normalizado) durante una semana.

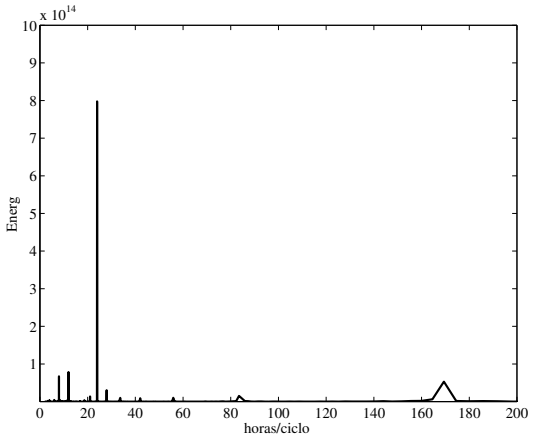

Figura 4. Periodograma de $x(t)$ donde se observan las componentes diaria y semanal de la periodicidad.

La carga de un tipo de día particular, los lunes por ejemplo, tiene un perfil característico, sin embargo conviene observar que ese perfil tiene variaciones de una semana a otra. La media y la desviación típica de la carga dentro de un mismo tipo de día tienen mucha importancia para caracterizar la secuencia de datos. En la parte inferior de la Fig. 5 se muestra con línea continua la media. Por encima y debajo se ha dibujado con línea de puntos la media más y menos la desviación típica respectivamente. Puede verse que la desviación típica es mayor en las horas centrales del día.

Como es habitual el conjunto de datos históricos H ha sido dividido en dos subconjuntos disjuntos de igual número de observaciones, tal que $H = E \bigcup P$ mediante muestreo aleatorio. Una parte forma el conjunto de entrenamiento E que será usado para el entrenamiento de las redes de neuronas y la otra el conjunto de prueba P que será usado para comparar modelos con distintos vectores de entrada.

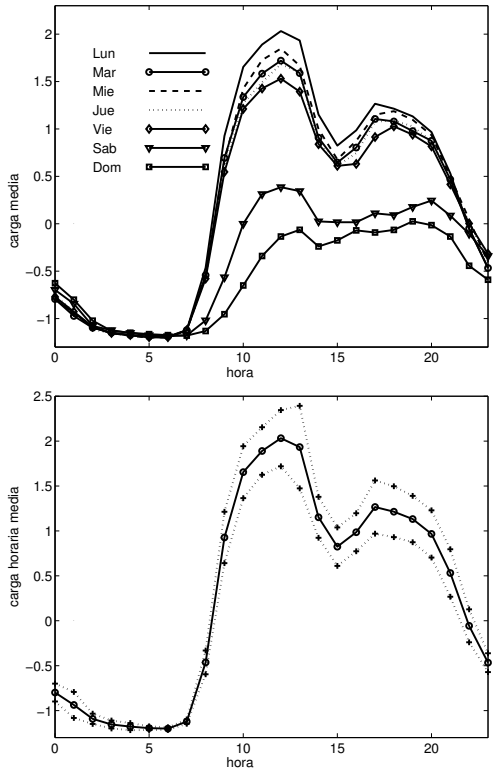

Figura 5. Parte superior: carga horaria media para cada día de la semana. Cada curva se ha obtenido tomando la media de valores (normalizados) de carga horaria usando datos del mismo tipo de día. Parte inferior: La línea continua representa la carga horaria media tomada los lunes. Las curvas superior e inferior corresponden a la media más y menos la desviación típica para cada hora.

3. SELECCIÓN DE VARIABLES

Como es sabido, las redes de neuronas artificiales permiten obtener modelos a la medida de los datos disponibles. Los modelos contienen parámetros que son ajustados para minimizar el error de predicción. Dichos parámetros carecen (en general) de significado en relación a las variables del problema y son meros artefactos necesarios para ajustar la salida proporcionada por la red. Por este motivo se les suele llamar modelos de caja negra Ljung (1987). El ajuste de parámetros (también llamado entrenamiento de la red) se realiza mediante técnicas de optimización, la mayoría de las cuales se basan en el gradiente local de la superficie de error respecto de los parámetros ajustables. Para poder proceder con el entrenamiento es preciso haber definido previamente la estructura neuronal: número de capas, de nodos y funciones base. Son muchas las estructuras que gozan de la propiedad de ser aproximadores universales, por lo que ese problema tiene menos interés. Otro paso previo antes del entrenamiento es decidir las componentes del vector $\mathbf{u}$ o vector de entrada. A este problema se le llama selección de variables y de él se ocupa este trabajo.

La importancia de la selección de variables es clara: por un lado si se usan más variables que las necesarias el modelo resultará demasiado flexible por lo que se ajustará a las particularidades de los datos usados para derivarlo para así disminuir el error en E. Como consecuencia la generalización será pobre. Por otro, si se olvidan variables importantes el modelo no podrá distinguir algunos casos como diferentes y las predicciones serán igualmente inadecuadas. La elección óptima debiera ser la que permita disminuir al máximo el error de gerenalización, o sea el error durante el uso final del modelo. Sin embargo este error no es conocido por lo que a lo sumo puede estimarse usando datos históricos.

En el caso de los modelos neuronales la selección de variables se ve complicada con otros factores como son, la selección del número de nodos de la capa oculta y las características del entrenamiento (algoritmo, número de iteraciones), uso de factores de penalización, etc. El método propuesto tiene en cuenta todos estos factores, pero antes de presentarlo conviene en primer lugar hacer un repaso del estado del arte.

3.1 Estado del arte

La selección de variables para ser usadas como entrada en modelos dinámicos ha sido objeto de estudio desde hace tiempo. Los primeros trabajos aparecieron con la popularización de los modelos de caja negra al estilo Box-Jenkins. Tras los trabajos sobre modelos lineales de Akaike (Akaike, 1974), ha sido la literatura sobre sistemas caóticos (Sauer *et al.*, 1991) la que ha proporcionado métodos que permiten realizar la búsqueda de la mejor combinación de variables para formar el vector de entrada en modelos no lineales. Posteriormente han surgido variaciones en el campo neuronal, de aprendizaje automático y para los sistemas borrosos (Díez *et al.*, 2004). En todos los casos los métodos pueden dividirse en dos grupos: los que necesitan construir el modelo para evaluar la validez del vector de variables de entrada y los que no requieren tal cosa. En el primer grupo se tienen las técnicas basadas en modelos locales (Kuo y Mallick, 1994; Piras y Germond, 1998; Yu *et al.*, 2000), en penalización de complejidad con regularizadores (Akaike, 1974; Moody, 1992; Murata *et al.*, 1994) mediante la longitud descriptora mínima (Rissanen, 1986; Judd y Mees, 1995), riesgo estructural mínimo (Vapnik, 1992) y usando la métrica de los datos (Schuurmans, 1997). Las técnicas más apropiadas para aprendizaje automático se basan en la reutilización de datos (Efron y Tibshirani, 1993), como la validación cruzada de multiplicidad k (Weiss y Kulikowski, 1991; Kohavi, 1995). Como variantes de este tipo de métodos cabe citar aquellos que evitan la exploración de todo el espacio de las combinaciones de las variables candidatas, como la inclusión progresiva o el borrado selectivo (Goutte, 1997; Miller, 1990; LeCun *et al.*, 1990; Reed, 1993; Levin y Leen, 1993).

En el segundo grupo de métodos destacan los que se basan en conceptos de topología para obtener la dimensión donde la dinámica está inmersa, bien para secuencias autónomas (Sauer *et al.*, 1991; Buzug y Pfister, 1992; Kennel *et al.*, 1992) o sistemas con entradas (Rhodes y Morari, 1998; Cao *et al.*, 1998). La estimación de funciones de densidad ha sido usada en (Pi y Peterson, 1994; Poncet y Moschytz, 1996). Otros conceptos como la información mutua (Fraser y Swinney, 1986; Bonnlander, 1996) y el análisis de componentes principales (Back y Cichocki, 1999) también pertenecen a este grupo.

3.2 Método propuesto

El método se basa en (Yuan y Fine, 1998) que propone la aplicación de un filtro para reducir el número de variables candidatas seguido de un procedimiento de búsqueda subóptima. La primera etapa del algoritmo permite clasificar un gran número de variables candidatas con una carga de cálculo moderada.

Las variables mejor clasificadas pasan a la segunda fase en la cual las combinaciones que forman son calificadas de acuerdo a la bondad de los modelos que proporcionan (Kohavi y John, 1997).

En la primera criba del método se usa un índice de diferencia cuadrática (IDC) que proporciona un valor que es menor cuanto mayor es la relación de la variable candidata con la variable dependiente. La selección se realiza creando previamente conjuntos de variables muy correladas entre sí. De cada grupo se seleccionan las que proporcionan un valor más bajo del índice, de este modo se consigue una selección formado por las m mejores candidatas de acuerdo al índice y se eliminan variables muy relacionadas entre sí.

El IDC se calcula como la suma de las diferencias cuadráticas entre valores sucesivos de la variable dependiente $x(t)$ cuando la variable independiente candidata ha sido ordenada. Para aclarar esta idea conviene considerar el conjunto de variables candidatas $\{z_v\}_{v=1}^{v=V}$. Es importante que todas las variables hayan sido escaladas de forma que se elimine la media y la tendencia de primer orden. Para una variable cualquiera del conjunto, como z_v, es posible ordenar las parejas $(z_v(t), x(t))$ de forma que la secuencia resultante de parejas $(z_v(t_i), x(t_i))$ cumpla que $z_v(t_i) \leq z_v(t_{i+1})$ para todo $i = 1, \cdots, N$. Entonces se define el IDC como

$$\mathcal{I}_v = \frac{1}{N-1} \sum_{i=1}^{N-1} \left(x(t_i) - x(t_{i+1})\right)^2 \qquad (2)$$

Las variables con menores valores de IDC son aquéllas que tienen una relación más estrecha con la variable dependiente $x(t)$. Es fácil ver que el IDC es una extensión al caso no lineal de la idea de correlación entre variables. De cada grupo de variables se escogen aquellas con menor valor de $\mathcal{I}$. De este modo se reduce el número de variables candidatas de V a $m < V$.

Posteriormente sería posible evaluar las 2^m combinaciones de variables preseleccionadas. Esta tarea puede conllevar una carga de cálculo prohibitiva pues para cada combinación es preciso evaluar los modelos a que da lugar y ello implica la construcción de cientos de modelos reutilizando los datos como se verá más adelante. Por ello se propone usar un algoritmo subóptimo que emplea un criterio de búsqueda. Este tipo de algoritmos se puede presentar en la versión aditiva (se van añadiendo variables una a una) o substractiva (se eliminan variables una a una). En ambos casos se usa un criterio para juzgar cual variable conviene añadir o eliminar. El criterio suele ser una estimación del error de generalización (Judge *et al.*, 1985).

El problema de esta técnica se presenta en decidir cuándo finalizar el proceso de añadir o retirar variables. Ha de tenerse en cuenta que la estimación del error de generalización no es perfecta y por tanto puede estar sesgada hacia modelos más complejos de lo necesario. Una posibilidad para paliar este problema consiste en introducir una variable candidata falsa creada artificialmente para contener valores aleatorios. Cualquier cambio que produzca unos efectos menores que los que la variable aleatoria produce ha de ser desestimado (Bi *et al.*, 2003).

Finalmente es posible usar métodos de búsqueda aleatorizados como por ejemplo los algoritmos genéticos. De este modo se pueden tener en cuenta muchas combinaciones de variables y explorar de una forma más eficiente el espacio de dimensión 2^m. De este modo pudiera parecer que además se evita el problema de finalizar el proceso de adición o substracción pues tal proceso no está presente. Esto no es así pues el método de búsqueda de soluciones en este caso estará guiado únicamente por el error estimado de generalización, que, como se ha dicho antes, puede conducir a soluciones sesgadas hacia los modelos más complejos.

Tabla 1. Valor del IDC $\mathcal{I}$ de las distintas variables en el ejemplo ilustrativo

Variable	$\mathcal{I}$
z_1	0.0135
z_2	0.0061
z_3	0.0180
z_4	0.0205

En este artículo se proporcionarán resultados obtenidos con las tres técnicas descritas anteriormente, comparando sus resultados. Como medida de la bondad de los modelos se va a usar el error estimado de generalización obtenido mediante validación cruzada de multiplicidad 10 (Weiss y Kulikowski, 1991; Kohavi, 1995).

3.3 Ejemplo de aplicación del IDC

A fin de ilustrar el funcionamiento del proceso de selección mediante el IDC se presenta aquí un problema simulado. En este ejemplo la variable a predecir x ha sido generada mediante la fórmula

$$x(t) = 0{,}1 z_1(t)^2 + 0{,}5 \log z_2(t) + \epsilon(t) \qquad (3)$$

siendo z_1 y z_2 dos variables tomadas de un conjunto de $V = 4$ variables candidatas y siendo ϵ ruido uniformemente distribuido. En la figura 6 se muestra la evolución temporal de las variables consideradas obtenidas por simulación.

Para ilustrar el método conviene dibujar las gráficas de $x(t)$ frente a cada una de las variables z_i con $i \in 1, 2, 3, 4$ como se puede ver en la figura 7. Se verifica que las variables que no influyen en x producen una gráfica informe, mientras que las variables influyentes dejan una marca característica aunque no apreciable por medidas lineales como el coeficiente de correlación. El IDC se basa en esta propiedad, sumando las diferencias cuadráticas que, lógicamente, son menores para las variables que están más relacionadas con x como puede verse en la Tabla 1.

3.4 Variables candidatas

Las variables candidatas se van a agrupar en varios conjuntos de características similares. Previamente conviene tomar en consideración ciertos aspectos de la secuencia que se trata de predecir. A partir del estudio de los datos registrados se ha revelado que:

- La carga en días laborables y en festivos tiene un perfil muy distinto.
- La carga en un día cualquiera se parece más a la carga del día anterior del mismo tipo.

Tomando estos hechos en consideración parece lógico seleccionar las variables de entrada para el modelo de predicción de forma que para predecir un lunes se usen valores de otros lunes. Si esto se lleva a cabo ocurrirá que las variables que forman el

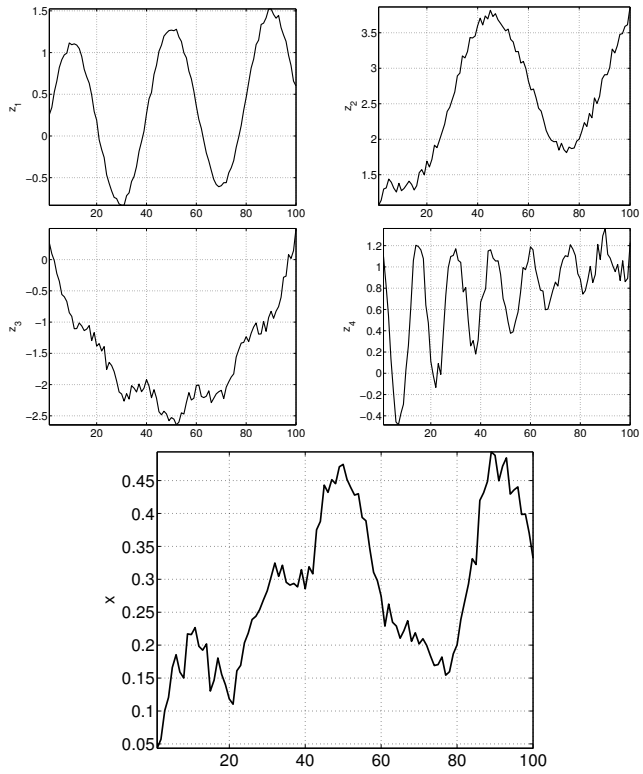

Figura 6. Trayectorias de las variables usadas en el ejemplo ilustrativo del IDC.

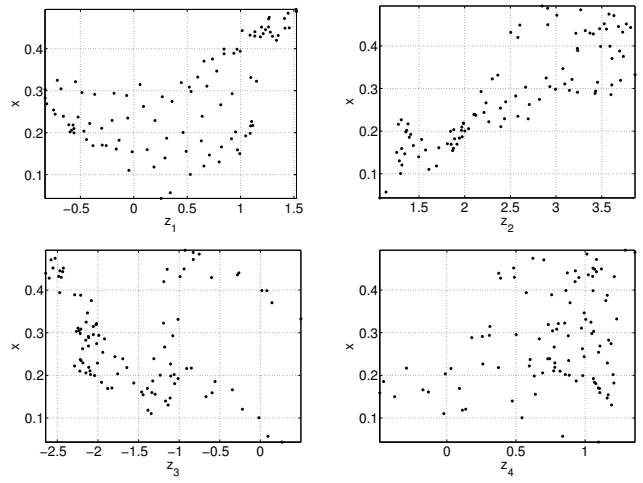

Figura 7. Proyecciones de las variables candidatas sobre x.

vector regresor no tienen igual retraso para cada día. Esto no plantea problemas prácticos pues basta con tomar las variables del regresor del día pasado más cercano del mismo tipo. El esquema se complica un poco por la existencia de festivos en medio de una semana. Conviene observar que:

- Un festivo en medio de la semana tiene una carga similar a la de un sábado, por lo que las variables para el regresor debieran tomarse del sábado pasado más cercano.
- Los datos de un festivo no deben usarse para predecir días no festivos.

Todo esto lleva a que además del adelanto en la predicción d es preciso añadir un término L que no es fijo y que puede describirse como el número mínimo de horas que hay que remontarse en el pasado para encontrar un día del mismo tipo al que que se quiere predecir. Teniendo en cuenta estas consideraciones, los grupos de variables candidatas quedan del modo que se muestra a continuación. En todos los casos se indica mediante t la hora correspondiente a la carga que se desea predecir.

1. **Valores pasados.** Una carga pasada se denota como $x(t - L)$, siendo L el retraso considerado que ha de cumplir $L \geq 24 \cdot 7$.
2. **Errores pasados.** Se calculan a partir de una predicción realizada para un día pasado usando un retraso $L \geq 24 \cdot 7$. De este modo se obtiene $e(t - L) = x(t - L) - \hat{x}(t - L)$.
3. **Valores de días similares.** Es un caso especial del grupo 1 en el que la carga corresponde a la misma hora de un día pasado similar al día de la predicción. A este efecto los días se dividen en tres grupos: domingos, sábados y festivos entre semana y laborables. En este caso el retraso L a aplicar depende del tipo de día. Para el grupo 1 (domingos) se tiene que $L = 7 \cdot 24$, para el grupo 2 (sábados y festivos en medio de la semana) se debe usar el menor valor de L que regresa a otro sábado o festivo en medio de semana, etc.

   Cabe considerar más de un valor pasado correspondiente a día similar, de este modo y en un caso general, para indicar la carga pasada j-ésima se usará la notación $xs_j(t) = x(t - L(t, j))$. Se ha indicado mediante $L(t, j)$ el retraso en horas, que depende del tipo de día y del orden j.
4. **Errores pasados en días similares.** Se definen de forma similar a la carga en días similares pero usando el error de predicción, de este modo se puede escribir que $es_j(t) = x(t - L(t, j)) - \hat{x}(t - L(t, j))$.
5. **Media de cargas pasadas.** Estas variables consideran medias empíricas calculadas usando $2q + 1$ valores pasados centrados en $t - \tau$, de forma que se pueden calcular como $a_{q,\tau}(t) = \sum_{k=-q}^{k=q} x(t - \tau + k)/(2q + 1)$.
6. **Indicadores temporales.** Se trata de variables de evolución cíclica que contienen información acerca del momento de la predicción t, tales como el día de la semana $d(t) \in \{1, \cdots, 7\}$, la hora del día $h(t) \in \{1, \cdots, 24\}$, el grupo al que pertenece el día $w(t) \in \{1, 2, 3\}$. Además se incluyen valores que dependen de forma senoidal de la hora y el día para, de este modo, producir una forma de onda que se ajusta a la periodicidad observada en los datos. Entre otras variables cabe considerar $c_h(t) = \sin(\frac{2\pi}{24}(h(t) - 6))$, $c_d(t) = \sin(\frac{2\pi}{7}d(t))$ y $c_{hd}(t) = c_h(t) \cdot c_d(t)$.
7. **Valores extremos.** Los máximos y mínimos observados recientemente o en días similares también pueden tener importancia para la predicción. Para ello se definen nuevas variables $x^s(t - L) = \text{máx}_k\, x(t - L - k)$ y $x^i(t - L) = \text{mín}_k\, x(t - L - k)$ para $0 \leq k \leq 24$.

Estos tipos de variables están parametrizados en función de uno o más valores como L, j, τ, q, etc. Al proporcionar valores adecuados a dichos parámetros es posible obtener n centenar de variables diferentes. La búsqueda por fuerza bruta de la mejor combinación de variables conlleva el análisis de 2^{100} combinaciones, lo cual implica una carga de cálculo muy elevada. El método de selección permite reducir dicho número como se expone a continuación.

3.5 Implementación del algoritmo

La figura 8 muestra los pasos seguidos en la implementación y pruebas del algoritmo. Las elipses representan conjuntos de

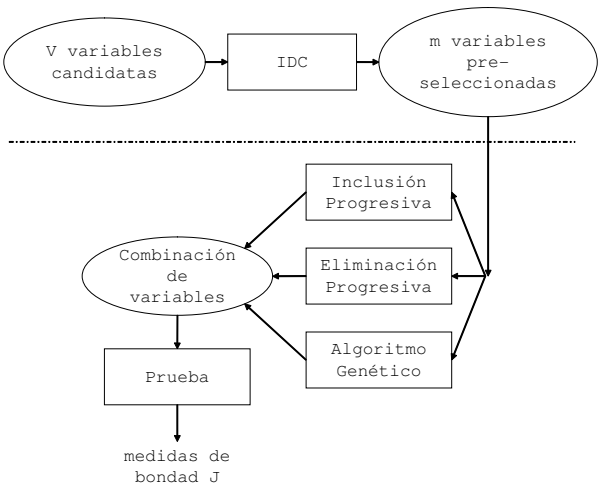

Figura 8. Proceso de selección en dos fases.

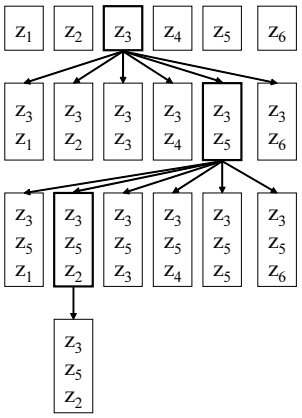

Figura 9. Ejemplo ilustrativo del proceso de inclusión progresiva seleccionando tres variables de un conjunto de seis.

variables, los rectángulos representan procesos. En primer lugar se realiza la pre-selección mediante el IDC como se indica en la parte superior del diagrama. Como resultado se obtienen $m < V$ variables pre-seleccionadas. Estas variables son las únicas consideradas en la segunda parte del método.

La segunda parte se muestra en la zona inferior de la figura 8. Las variables pre-seleccionadas son combinadas de acuerdo con alguno de los tres procedimientos considerados: inclusión progresiva, eliminación progresiva y algoritmos genéticos. En cada caso el resultado es una combinación de variables seleccionadas. Dicha combinación es puesta a prueba usando datos nuevos y como resultado se obtiene un índice de bondad J. La forma de proceder en el método de inclusión progresiva se pone de manifiesto en la figura 9 en un ejemplo en el que se seleccionan tres variables de un conjunto de seis, como puede verse las variables son añadidas una a una y no se explora todo el árbol de posibilidades (Berenguel *et al.*, 1998). La eliminación progresiva seguiría el proceso inverso partiendo de seis variables hasta dejar solamente tres. Finalmente el método de algoritmos genéticos considera cada posible combinación como individuo de una población que evoluciona, seleccionando la combinación más adecuada.

El índice de bondad J se calcula como el error cuadrático medio relativo a la carga media durante los días del mismo tipo que el predicho. Esta carga media $m(t)$ depende del tipo de día $w(t) \in \{1, 2, 3\}$ y se calcula con los datos históricos. De este modo para un conjunto P conteniendo p valores a predecir el índice de bondad se calcula como:

$$J_P = \sqrt{\frac{1}{p}\sum_{t=1}^{p}\left(\frac{100e(t)}{m(t)}\right)^2} \qquad (4)$$

Dada una combinación de variables de entrada C, el valor de J que le corresponde depende de varios factores:

- El conjunto de datos P usados para evaluar el índice.
- El tamaño de la red de neuronas que realiza el modelo.
- Las características del entrenamiento.
- La utilidad de las variables de entrada.

Resulta evidente que interesa potenciar el último factor y disminuir la influencia de los tres primeros. En este caso el conjunto P consiste en datos procedentes del archivo histórico que se han dejado aparte para este fin y que, por tanto, no han sido usados en el entrenamiento. Puesto que los datos son necesariamente escasos debido a la influencia de los cambios socioeconómicos, resulta que P no puede ser tan amplio como uno desee y por tanto hay que recurrir a métodos de reutilización de datos como la validación cruzada de multiplicidad k. En esta aplicación se ha usado un conjunto P que contiene el 50 % de todos los datos históricos y se ha empleado $k = 10$.

El tamaño y el entrenamiento de la red son tenidos en cuenta repitiendo la creación y prueba de modelos con distintas configuraciones y escogiendo los que producen mejor resultado. Téngase en cuenta que este mejor resultado es J_E siendo el conjunto $E = H \setminus P$ la parte de datos históricos H que se utiliza para entrenamiento de redes.

Con todo ello, dada una combinación C, el valor de J_P es una variable aleatoria pues depende de factores aleatorios como el remuestreo usado, los valores iniciales de pesos de las redes, etc. Por ello conviene indicar la media $\hat{\mu}_J$ y la desviación típica $\hat{\sigma}_J$ calculadas de forma empírica.

4. RESULTADOS

En este apartado se muestran los resultados del algoritmo propuesto y sus variantes al aplicarlos al conjunto de datos históricos. De este modo se seleccionarán las combinaciones de variables de entrada que resulten más prometedoras. Posteriormente se compararán los resultados obtenidos por estas combinaciones y otras adicionales usando nuevos datos no vistos hasta ahora.

4.1 Primer paso de selección

El IDC se ha calculado usando los datos en el conjunto $E = H \setminus P$ para cinco de los siete grupos de variables considerados. Las variables de los grupos 2 y 4 no puede ser catalogadas mediante el IDC pues son errores pasados que dependen del modelo particular. En la Tabla 2 se consigna el valor del índice para las mejores variables ordenadas por grupos.

De este modo resultan $m = 19$ variables pre-seleccionadas para tener en cuenta en la fase siguiente. La reducción es importante pues si no se plantean restricciones es posible considerar un número V de variables cercano al centenar.

Tabla 2. Variables independientes con menor valor del IDC $\mathcal{I}$ para cada grupo excepto el 2 y el 4

Grupo	Variable	$\mathcal{I}$
1	$x(t-336)$	0.1626
1	$x(t-672)$	0.1664
1	$x(t-504)$	0.1718
3	$xs_2(t)$	0.0873
3	$xs_1(t)$	0.1132
3	$xs_3(t)$	0.1253
5	$a_{\,2,504}(t)$	0.2316
5	$a_{\,6,504}(t)$	0.5034
5	$a_{\,2,338}(t)$	0.7444
5	$a_{10,504}(t)$	0.9846
5	$a_{10,346}(t)$	1.3778
5	$a_{\,6,342}(t)$	1.9100
6	$d(t)$	0.1373
6	$h(t)$	0.3332
6	$c_h(t)$	0.3742
6	$c_d(t)$	0.1373
6	$c_{hd}(t)$	0.1607
7	$x^s(t-336)$	2.5621
7	$x^i(t-336)$	2.8954

4.2 Segundo paso de selección

Para realizar la comparación entre modelos se va a usar un índice de bondad J_P que se ha definido con anterioridad. Las $m = 19$ variables que aparecen en la Tabla 2 son candidatas para la segunda fase del algoritmo de selección. Es preciso añadir las variables basadas en errores pasados que no pueden ser usadas en la primera fase pues dependen del modelo. A fin de realizar una estructura NARMAX conviene incluir un valor de error pasado por cada variable del grupo 1 (Chen *et al.*, 1990). De este modo se añaden las variables $e(t-336)$, $e(t-672)$, $e(t-504)$, $es_1(t)$, $es_2(t)$ y $es_3(t)$ por lo que se alcanza un total de $m = 25$ variables candidatas.

En la segunda fase del algoritmo se usan los métodos de inclusión progresiva, eliminación progresiva y algoritmos genéticos. Los valores obtenidos para la medida de la bondad de los modelos se muestran en la Tabla 3. Los resultados obtenidos por los distintos métodos son muy similares y no es factible concluir que un método sea superior a los otros. Como conclusión parece que los modelos con vectores de entrada de dimensión 5 a 7 producen los mejores resultados. En otras palabras, el error esperado en la predicción con nuevos datos es menor para modelos con que usan entre 5 y 7 variables independientes. Esta conclusión será puesta a prueba a continuación.

4.3 Prueba con nuevos datos

Es el momento de probar los modelos con las combinaciones de entradas seleccionadas por el algoritmo. La medida de bondad es nuevamente el error cuadrático relativo a la carga media definido en la ecuación (4), pero usando un nuevo conjunto de datos N que es posterior a H y que no ha sido usado con anterioridad.

Los valores consignados en la Tabla 4 corresponden a la bondad medida sobre el conjunto de prueba N. La primera observación que debe hacerse es que los errores son mayores que las estimaciones hechas a partir del conjunto de prueba P. Esta situación es normal y se alivia en cierta medida realizando nuevas selecciones de modelos con periodicidad, por ejemplo cada mes. Se ha optado sin embargo por mostrar los resultados de este modo pues así se pone de manifiesto fenómenos que, en mayor o menor grado ocurrirán siempre.

Tabla 3. Bondad de los modelos para distintos vectores de entrada

Inclusión progresiva

Vector de entrada	$\hat{\mu}_J$	$\hat{\sigma}_J$
xs_1	17.4	2.78
xs_1, es_1	15.7	3.12
xs_1, es_1, xs_2	15.4	3.28
xs_1, es_1, xs_2, xs_3	15.1	3.32
$xs_1, es_1, xs_2, xs_3, a_{\,6,342}$	15.0	3.42
$xs_1, es_1, xs_2, xs_3, a_{\,6,342}, c_{hd}$	14.9	3.46
$xs_1, es_1, xs_2, xs_3, a_{\,6,342}, c_{hd}, es_2$	14.7	3.46

Eliminación progresiva

Vector de entrada	$\hat{\mu}_J$	$\hat{\sigma}_J$
todas las variables	14.2	3.72
$xs_1, xs_2, xs_3, es_1, es_2, a_{\,2,338}, a_{\,6,342}, c_{hd}$	14.6	3.50
$xs_1, xs_2, xs_3, es_1, es_2, a_{\,2,338}, c_{hd}$	14.6	3.48
$xs_1, xs_2, es_1, es_2, a_{\,2,338}, c_{hd}$	14.7	3.47
$xs_1, xs_2, es_1, a_{\,2,338}, c_{hd}$	14.8	3.45
$xs_1, xs_2, es_1, a_{\,2,338}$	15.0	3.36

Algoritmos genéticos

Vector de entrada	$\hat{\mu}_J$	$\hat{\sigma}_J$
$xs_1, xs_2, xs_3, es_1, es_2\, a_{\,2,338}, c_{hd}$	14.6	3.47
$xs_1, xs_2, es_1, es_2, a_{\,2,504}, c_d$	14.4	3.48
$xs_1, xs_2, es_1, es_2, a_{\,2,338}, c_{hd}$	14.7	3.47
$xs_1, xs_2, xs_3, es_1, a_{\,6,342}$	15.0	3.42
$xs_1, xs_2, es_1, a_{\,2,338}$	15.0	3.36

Otra conclusión es que el algoritmo de selección de variables está sesgado hacia modelos de complejidad mayor de lo necesario. Esto se pone de manifiesto por el hecho de que los mejores modelos en la prueba con nuevos datos contienen 4 variables mientras que el algoritmo recomendaba usar entre 5 y 7 variables de entrada. Este resultado tampoco es inesperado y es típico de algoritmos de selección donde se hace un uso repetido de datos. Conviene sin embargo indicar que los resultados obtenidos son los mejores posibles dada la escasez de datos.

Finalmente, la última línea de la Tabla 4 corresponde a un modelo que utiliza la media de las predicciones de los otros modelos de la tabla. Como puede verse este modelo mixto produce resultados bastante buenos. Esta técnica de promediar modelos ha sido propuesta en varias aplicaciones produciendo resultados mejores que los modelos individuales.

Téngase en cuenta que la información de la Tabla 4 no puede usarse para seleccionar modelos pues para construir la tabla se necesitan datos posteriores al instante en el que los modelos se construyen. Sería ingenuo por tanto concluir que el modelo con entradas (xs_1, xs_2, es_1, $a_{\,2,338}$) es el que debe usarse. Lo que interesa son reglas generales para realizar la selección de variables que tengan validez general. De los experimentos mostrados pueden extraerse varias indicaciones que sirven a ese fin.

- El algoritmo de selección frecuentemente proporcionará modelos con más variables de las necesarias.
- El error durante el uso del modelo frecuentemente será mayor que el estimado durante las pruebas.
- El modelo usado para predicción ha de revisarse periódicamente para así tomar en consideración los nuevos datos que se van observando y además desechar los más antiguos.

Tabla 4. Comparación de varios modelos con nuevos datos

Vector de entrada	$\hat{\mu}_J$	$\hat{\sigma}_J$
xs_1, xs_2, es_1, $a_{2,338}$	15.4	3.12
xs_1, es_1, xs_2, xs_3, $a_{6,342}$	15.4	3.13
xs_1, xs_2, es_1, es_2, $a_{2,338}$, c_{hd}	16.1	3.21
xs_1, xs_2, xs_3, es_1, es_2, $a_{2,338}$, c_{hd}	16.7	3.27
-	15.4	3.17

Tabla 5. Comparación de varios tipos de modelos con los nuevos datos

Modelo	Tipo	Nº Variables	$\hat{\mu}_J$	$\hat{\sigma}_J$	valor-p vs. M_1
M_1	RN	4	14.9	2.86	-
M_2	L	6	18.5	3.05	0.0061
M_3	L	14	19.2	3.10	0.0019
M_4	RN	6	17.6	3.15	0.0052
M_5	RN	14	17.9	3.23	0.0049

- En lugar de confiar en un único modelo resulta conveniente tener un conjunto de modelos y promediar sus predicciones.

4.4 Comparación con otros modelos

Las redes de neuronas, los conjuntos borrosos, y en general los aproximadores basados en series truncadas de ciertas bases, poseen la propiedad de ser aproximadores universales. Con esta premisa se justifica el uso de un modelo neuronal. La justificación se basa en que si existe otro modelo mejor éste sería aproximable por una red de neuronas. A pesar de todo ello conviene comparar los resultados obtenidos con los que se consiguen usando otras técnicas, en particular la comparación con métodos más clásicos es obligada.

En la Tabla 5 se indica en cada columna un modelo de los comúnmente propuestos para predicción. La primera fila corresponde a M_1 que una red de neuronas con vector de entrada compuesto por las variables xs_1, es_1, xs_2, xs_3, $a_{6,342}$. Este modelo es uno de los que produce mejores resultados en la fase de selección de variables. Contra este modelo se van a comparar otros más clásicos. Téngase en cuenta que el modelo seleccionado para la comparación no es siquiera el que mejores resultados produce como ya se ha indicado en el punto anterior. En particular, usando un promedio de modelos se obtendría menor error. Se ha preferido mostrar este caso pues constituye un caso típico de los resultados que se pueden obtener con una selección adecuada de variables de entrada.

Siguiendo con la Tabla 5, la segunda fila corresponde a un modelo (M_2) que pertenece a la familia ARIMA y que se ha ajustado para minimizar el error de predicción a dos semanas vista (Ljung, 1987). El modelo usa tres valores autoregresivos y tres términos de media móvil. De forma similar, el modelo M_3 pertenece a la familia ARIMA pero ha sido determinado minimizando la predicción a un paso. El modelo es luego usado de forma iterada para producir la predicción a dos semanas. En este caso se usan 10 valores autoregresivos no consecutivos y cuatro de media móvil. Los retrasos para las variables han sido seleccionadas mediante algoritmos genéticos usando el criterio de Akaike como medida de ajuste.

Los dos últimos modelos de la tabla M_4 y M_5 consisten en redes de neuronas cuyas entradas coinciden con las de M_2 y M_3 respectivamente. El tamaño de las redes ha sido seleccionado usando validación cruzada de multiplicidad 10.

Las columnas 4 y 5 de la Tabla 5 indican la media y la desviación típica empíricas para cada modelo. En la columna 6 se ha indicado el valor p de un contraste de hipótesis. Este valor representa la probabilidad de obtener los valores observados de $\hat{\mu}_J$ siendo verdadera la hipótesis nula. En cada fila se ha tomado como hipótesis nula que el modelo en cuestión es mejor que M_1. Puede verse en la tabla que estas probabilidades son siempre menores al 1 %.

Conviene indicar que los errores de predicción correspondientes a festivos entre semana han sido retirados. De este modo se realiza una comparación más justa con los modelos lineales que disponen de poca capacidad para representar estas excepciones. Los resultados de la Tabla 5 muestran que los modelos neuronales (marcados con RN) producen mejores resultados que los lineales (L). Otra conclusión es que la selección de variables de entrada produce mejores resultados que las otras prácticas usadas frecuentemente. En particular los modelos M_4 y M_5 producen peores resultados que M_1 a pesar de que este último cuenta con menor número de variables independientes. La razón es que M_1 es menos vulnerable al sobreentrenamiento y por ello proporciona una mejor generalización.

5. CONCLUSIONES

El artículo ha mostrado que las redes de neuronas pueden ser usadas para la predicción de la carga horaria en centros de atención telefónica. Los beneficios derivados de una correcta predicción se traducen en mejor servicio a menor coste. Resulta obvio que una pequeño porcentaje de mejora constituye un gran ahorro en una empresa con muchos empleados.

Se ha puesto de manifiesto que el problema de la selección de variables independientes es de vital importancia debido a la escasez de datos. Dicha escasez es inherente en la predicción de variables como la carga de llamadas que dependen estrechamente de cambios socio-económicos. El método de selección usado permite tratar un gran número de variables candidatas en la primera fase, mientras que en la segunda fase emplea un método que clasifica las variables de acuerdo al resultado proporcionado por el modelo. En este contexto se han comparado diversas técnicas de selección de variables observándose que ninguna de ellas es estrictamente mejor que las demás.

Finalmente, la comparación con nuevos datos ha puesto de manifiesto que la selección de variables permite obtener mejores resultados. Se ha revelado la complejidad del problema producida por la necesidad de hallar un equilibrio entre flexibilidad y generalización. Además se han extraido conclusiones que pueden aplicarse con carácter general: contrarrestar la tendencia a la sobreparametrización, la revisión periódica de modelos y el uso de agrupaciones o promedios de modelos.

AGRADECIMIENTOS

Este trabajo ha sido financiado parcialmente por la comisión de la Comunidad Europea a través del proyecto de investigación HYCON FP6-511368.

REFERENCIAS

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Trans. on Automatic Control* **AC-19**(5), 716–723.

Andrews, B. y S.M. Cunningham (1995). L.L. Bean improves call-center forecasting. *Interfaces* **25**, 1–13.

Antipov, A. y N. Meade (2002). Forecasting call frequency at a financial services call centre. *Journal of Operational Research Society* **53**(9), 953–960.

Avramidis, A.N., A. Deslauriers y P. L'Ecuyer (2004). Modeling daily arrivals to a telephone call center. *Management Science* **50**(7), 896–908.

Back, A.D. y A. Cichocki (1999). Input variable selection using independent component analysis y higher order statistics. In: *First International Conference on Independent Component Analysis y Signal Separation*. France. pp. 203–208.

Berenguel, M., M.R. Arahal y E.F. Camacho (1998). Modelling the free response of a solar plant for predictive control. *Control Engineering Practice* **6**, 1257–1266.

Bi, Jinbo, Kristin Bennett, Mark Embrechts, Curt Breneman y Minghu Song (2003). Dimensionality reduction via sparse support vector machines. *Journal of Machine Learning Research* **3**, 1229–1243.

Bishop, C.M. (2006). *Pattern Recognition y Machine Learning*. Springer. New York.

Bonnlander, B. (1996). Nonparametric selection of input variables for connectionist learning.

Buzug, T. y G. Pfister (1992). Optimal delay time y embedding dimension for delay-time coordinates by analysis of the global static y local dynamical behavior of strange attractors. *Phys. Rev. A* **45**, 7073–7084.

Cao, L., A.I. Mees, K. Judd y G. Froyland (1998). Determining the minimum embedding dimensions of input-output time series data. *Internat. J. Bifur. Chaos Appl. Sci. Engrg.* **8**, 1491–1504.

Chen, S., S.A. Billings, C.F.N. Cowan y P.M. Grant (1990). Practical identification of narmax models using radial basis functions. *Int. J. Control* **52**, 1327–1350.

Díez, J. L., J. L. Navarro y A. Sala (2004). Algoritmos de agrupamiento en la identificación de modelos borrosos. *Revista Iberoamericana de Automática e Informática Industrial* **1**(2), 32–41.

Efron, B. y R.J. Tibshirani (1993). *An Introduction to the Bootstrap*. Chapman y Hall. London.

Fraser, A.M. y H.L. Swinney (1986). Independent coordinates for strange attractors from mutual information. *Phys. Rev. A* **33**, 1134–1140.

Gans, N., G. Koole y A. Mandelbaum (2003). Telephone call centers: Tutorial, review y research prospects. *Manufacturing y Service Operations Management* **5**(2), 79–141.

Goutte, C. (1997). Lag space estimation in time series modelling.

Jongbloed, G. y G.M. Koole (2001). Managing uncertainty in call centers using poisson mixtures. *Applied Stochastic Models in Business y Industry* **17**, 307–318.

Judd, K. y A. I. Mees (1995). On selecting models for nonlinear time series. *Physica D* **82**, 426–444.

Judge, G.G., W.E. Griffths, R.C. Hill, H. Lütkepohl y T.C. Lee (1985). *The theory y practice of econometrics*. Wiley. New York.

Kennel, M.B., R. Brown y H.D.I. Abarbanel (1992). Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Phys. Rev. A* **45**, 3403–3411.

Kohavi, R. y G.H. John (1997). Wrappers for feature subset selection. *Artificial Intelligence* **97**(1-2), 273–324.

Kohavi, Ron (1995). A study of cross-validation y bootstrap for accuracy estimation y model selection. In: *International Joint Conference on Artificial Intelligence*. pp. 1137–1145.

Koole, G. y A. Mandelbaum (2002). Queueing models of call centers an introduction. *Annals of Operations Research* **113**, 41–59.

Kuo, L. y B. Mallick (1994). Variable selection for regression models. Technical Report 94-26. Department of Statistics, University of Connecticut, EE.UU.

LeCun, Y., J. Denker, S. Solla, R. E. Howard y L. D. Jackel (1990). Optimal brain damage. In: *Advances in Neural Information Processing Systems II* (D. S. Touretzky, Ed.). Morgan Kauffman. San Mateo, CA. pp. 740–747.

Levin, A.U. y T.K. Leen (1993). Using pca to improve generalization in supervised learning. In: *NATO Workshop on Statistics y Neural Networks*. pp. 740–747.

Ljung, L. (1987). *System Identification – Theory for the user*. Prentice Hall. Englewood Cliffs, NJ.

Miller, A.J. (1990). *Subset Selection in Regression*. Chapman y Hall. London.

Moody, J. (1992). The effective number of parameters: An analysis of generalization y regularization in nonlinear learning systems. In: *Advances in Neural Information Processing Systems* (D. S. Touretzky, Ed.). Vol. 4. Morgan Kaufmann, San Mateo. pp. 29–39.

Murata, N., S. Yoshizawa y S.-I. Amari (1994). Network information criterion. determining the number of hidden units for an artificial neural network model. *IEEE Transactions on Neural Networks* **5**(6), 865–872.

Pi, H. y C. Peterson (1994). Finding the embedding dimension y variable dependences in time series. *Neural Computation* **6**, 509–520.

Pinedo, M., S. Seshadri y J.G. Shanthikumar (1999). Call centers in financial services: strategies, technologies, y operations. In: *Creating Value in Financial Services: Strategies, Operations y Technologies* (E.L.Melnick, P.Ñayyar, M.L. Pinedo y S. Seshadri, Eds.). Chap. 18, pp. 357–388. Kluwer.

Piras, A. y A. Germond (1998). Local linear correlation analysis with the som. *Neurocomputing* **21**(1-3), 79–90.

Poncet, A. y G.S. Moschytz (1996). Selecting inputs y measuring nonlinearity in system identification. In: *Neural Networks for Identification, Control, Robotics, y Signal/Image Processing*. IEEE Computer Society. pp. 2–10.

Reed, R. (1993). Pruning algorithms–a survey. *IEEE Transactions on Neural Networks* **4**, 740–747.

Rhodes, C. y M. Morari (1998). Determining the model order of nonlinear input/output systems. *AIChE Journal* **44**, 151–163.

Rissanen, J. (1986). Stochastic complexity y modeling. *Annals of Statistics* **14**, 1080–1100.

Sauer, T., J. A. Yorke y M. Casdagli (1991). Embedology. *J. Statis. Phys.* **65**, 579–616.

Schuurmans, D. (1997). A new metric-based approach to model selection. In: *Proceedings of the Fourteenth National Conference on Artificial Intelligence*. pp. 552–558.

Shen, H. y J.Z. Huang (2008). Interday forecasting y intraday updating of call center arrivals. *Manufacturing y Service Operations Management* **10**, 391–410.

Sze, D.Y. (1984). A queuing model for telephone operator staffing. *Operations Research* **32**, 229–249.

Valverde, R. y D. Gachet (2007). Identificación de sistemas dinámicos utilizando redes neuronales rbf. *Revista Iberoamericana de Automática e Informática Industrial* **4**(2), 32–42.

Vapnik, V. (1992). Principles of risk minimization for learning theory. In: *Advances in Neural Information Processing Systems* (D. S. Touretzky, Ed.). Vol. 4. Morgan Kaufmann, San

Mateo. pp. 831–839.
Weiss, S.M. y C.A. Kulikowski (1991). *Computer Systems That Learn*. Morgan Kaufmann.
Werbos, Paul J. (1988). Generalization of backpropagation with application to a recurrent gas market model.. *Neural Networks* **1**(4), 339–356.
Yu, D., J.B. Gomm y D. Williams (2000). Neural model input selection for a mimo chemical process. *Engineering Applications of Artificial Intelligence* **13**, 15–23.
Yuan, J.-L. y T.L. Fine (1998). Neural-network design for small training sets of high dimension. *IEEE Transactions on neural networks* **9**, 266–280.