Imputación de datos censurados mediante redes neuronales: una aplicación a la EPA*

Teresa Villagarcía
Alberto Muñoz
Departamento de Estadística y Econometría
Universidad Carlos III de Madrid

Resumen

En este trabajo se propone el uso de redes neuronales para imputar valores a observaciones censuradas de variables de duración. El procedimiento es no paramétrico y puede utilizar información de un gran número de covariables para predecir la o las variables de interés. Se ha aplicado a datos de edades de jubilación en la encuesta de población activa (EPA) y los resultados han sido semejantes a los obtenidos aplicando los modelos tradicionales que integran datos censurados.

Palabras clave: *censura, imputación, redes neuronales, EPA.*
Clasificación JEL: *B40, J00*

Abstract

In this work we propose the use of neural networks to impute values to right censored data. The proposed method is non parametric, and it is able to cope with a large number of covariates and observations. We have applied the method to retirement ages in the Spanish Population Survey. The results are similar to those obtained using parametric lifetime models that take account of the censoring mechanism.

Key words: *censored data, survival analysis, missing data imputation, neural networks. EPA.*
JEL Classification: *B40, J00*

1. Introducción

Desde hace algunos años, tanto en España como en países de nuestro entorno, se han puesto a disposición de los investigadores los datos individuales de las grandes encuestas públicas. Estos datos han sido de una extraordinaria utilidad y han dado lugar a numerosos trabajos de investigación microeconométrica. Sin embargo, las grandes encuestas económicas como la Encuesta de Población Activa (EPA) o la Encuesta de Presupuestos Familiares (EPF) no están diseñadas para la investigación econométrica y tienen, por tanto, algunas carencias. Así, por ejemplo, la EPA carece de datos sobre ingresos de los encuestados, lo que supone un grave escollo a la hora de utilizarla en el ajuste de modelos de salarios. También, por su carácter de encuesta transversal, plantea determinados problemas en la captación de datos longitudinales como la duración del desempleo o edades de jubilación. En el primer caso, la encuesta recoge la duración del desempleo de los encuestados parados desde que perdieron su empleo hasta la realización de la encuesta. En el segundo, las personas activas de edad avanzada se van a jubilar en el futuro pero no se conoce la

* Los autores agradecen, en la realización de este trabajo, el apoyo financiero de los proyectos DGICYT PB93-0232 y PB94-0374.

edad exacta de jubilación. Este fenómeno en que se conoce que un proceso ha durado más o menos de un determinado valor se denomina censura, y los datos que la presentan son datos censurados. Así, es posible saber que la duración del desempleo de un parado será mayor o igual a la duración que indica en la EPA.

Se han utilizado muchos procedimientos para solventar estos problemas. Un método ampliamente usado es estimar un modelo a partir de fuentes que dispongan del dato e imputar el dato a la fuente que carece de él. Cuando el problema es de datos censurados es necesario utilizar modelos de duración, tanto paramétricos como no paramétricos (Lawless, 1982; Villagarcía, 1995). Estos modelos son complejos y no tienen una interpretación sencilla, por lo que su uso es algo restringido. En este trabajo proponemos la utilización de redes neuronales, concretamente del perceptrón multicapa, para realizar una imputación no paramétrica de variables censuradas que puede utilizarse para resolver los dos problemas que hemos ilustrado.

La estructura del artículo es la siguiente: en la sección 2 se hace una breve introducción a los modelos de redes neuronales que serán utilizados en este trabajo para la imputación de datos censurados. La sección 3 explica cómo se realiza la imputación de datos censurados. La sección 4 presenta los resultados obtenidos para datos de la EPA. Finalmente, la sección 5 resume las conclusiones del trabajo.

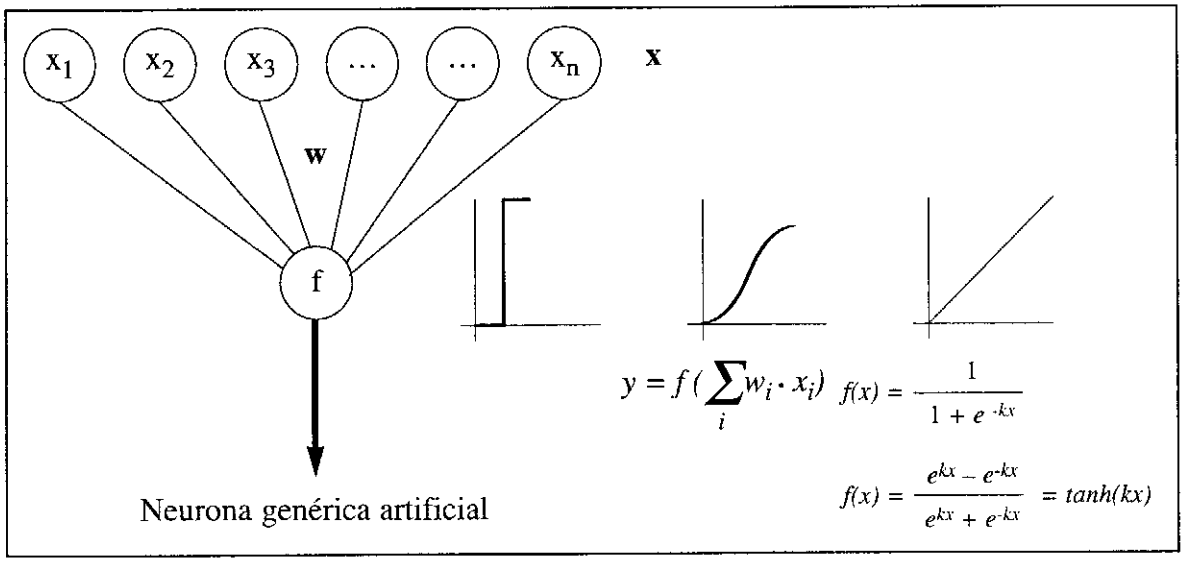
2. Redes neuronales

Una red neuronal es un conjunto de unidades computacionales simples que están altamente interconectadas. Estas unidades se denominan nodos o neuronas, debido a que surgieron como un intento de modelar de modo sencillo las neuronas biológicas.

El modelo de neurona que utilizaremos puede observarse en la Figura 1.

FIGURA 1

MODELO GENERAL DE NEURONA SIMPLE. SE MUESTRAN TRES FUNCIONES DE ACTIVACIÓN POSIBLES



La neurona recibe una suma ponderada de entradas, que a su vez podrán ser las salidas de otras neuronas conectadas con ella. La salida de la neurona es uno (modeliza la emisión de un impulso eléctrico) si la suma ponderada de las entradas es mayor que un determinado valor umbral. Si tal suma es menor, la respuesta es cero. Este modelo puede ser representado matemáticamente por:

$$y = f\left(\sum_{i=1}^{n} w_i x_i - \theta\right) \tag{1}$$

donde y es la salida de la neurona, x_i es la entrada i-ésima a la neurona, w_i es el peso de la conexión existente entre la entrada i-ésima y la neurona. θ es el umbral de respuesta de la neurona, y f es la función de activación escalón, definida como

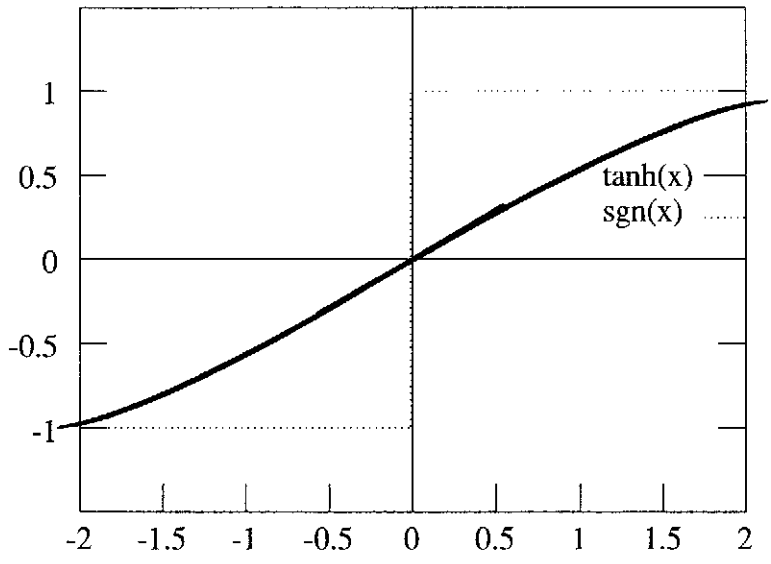
$$f(\mathrm{u}) = \begin{cases} 1, & \text{si } u \geq 0 \\ 0, & \text{en otro caso} \end{cases} \tag{2}$$

Este modelo de neurona es muy simple; sin embargo, la capacidad computacional de una red formada por unidades de este tipo es equivalente a la de un ordenador, en el sentido clásico de von Neumann.

El modelo puede ser extendido cambiando la función de activación de la neurona. La Figura 2 muestra una función escalón (la función signo) y una función no lineal de activación de la neurona.

FIGURA 2

FUNCIÓN SIGNO Y FUNCIÓN TANGENTE HIPERBÓLICA



Observemos que podemos ver las entradas y los pesos como ciertos vectores en un espacio multidimensional. Utilizando la notación $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$ para el vector de entradas, y $w = (w_1, \ldots, w_n) \in \mathbb{R}^n$ para el vector de pesos sinápticos, la salida de la neurona puede escribirse como $f(x \cdot w)$.

El proceso de una neurona simple puede interpretarse geométricamente de la siguiente forma: dado un vector de entrada $\boldsymbol{x}$, en primer se calcula el producto escalar $\boldsymbol{x} \cdot \boldsymbol{w}$ y se le resta el umbral θ. Si el resultado es positivo, la neurona produce una de las dos respuestas posibles; si es negativo, la otra. Por tanto, una neurona con función de activación escalón funciona exactamente como un clasificador lineal. Si utilizamos una función de activación no lineal (por ejemplo, la función logística), puede probarse que la neurona actuará como un clasificador no lineal.

2.1. *Perceptrones multicapa*

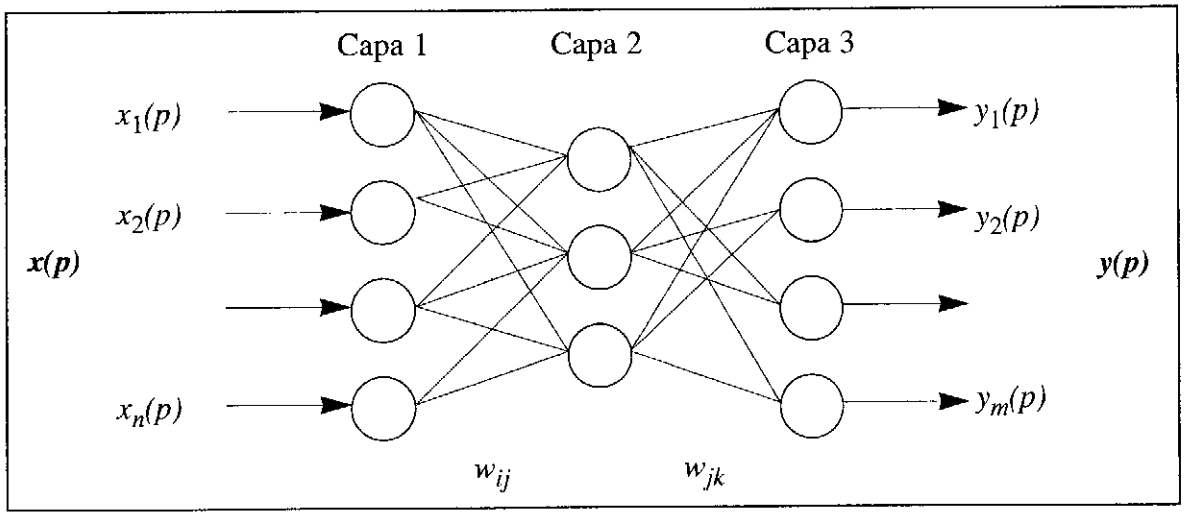
En esta sección vamos a introducir una clase muy importante de redes neuronales, llamadas perceptrones multicapa. Las neuronas en estas redes están organizadas por capas. Siempre hay una capa de entrada, una capa de salida, y una o más capas ocultas.

En favor de la claridad, restringiremos nuestra exposición a redes de tres capas, esto es, redes con una única capa oculta. La extensión a redes con más capas ocultas es inmediata.

El esquema de la red sobre la que vamos a trabajar puede verse en la Figura 3.

FIGURA 3

ARQUITECTURA DE PERCEPTRÓN MULTICAPA CON TRES CAPAS



La capa 1 es la capa de entrada, y está formada por n unidades, una por cada variable de entrada. Su uso es el de almacenar el dato (multivariante) bajo proceso. La capa 2 es la capa oculta, y está compuesta por r unidades, donde, generalmente $r \leq n$. La capa 3 es la capa de salida, y está formada por m unidades, una por cada componente del vector de variables respuesta y. El objetivo de la red es aprender a asociar pares $\{x(p), y(p)\}$, donde $x(p) \in \mathbb{R}^n$, $y(p) \in \mathbb{R}^m$. Esto es, la red debe aprender una función $\mathbb{R}^n \xrightarrow{\Phi} \mathbb{R}^m$, $\Phi(x) = y$, a partir de una muestra $\{x(p), y(p)\}$, $p = 1, \ldots, N$. Este modelo será adecuado para procesar datos multivariantes procedentes de encuestas, donde las $\{x(p)\}$ serán los valores de las variables independientes para la observación p-ésima, y las $\{y(p)\}$ constituirán los valores de las variables respuesta para la misma observación.

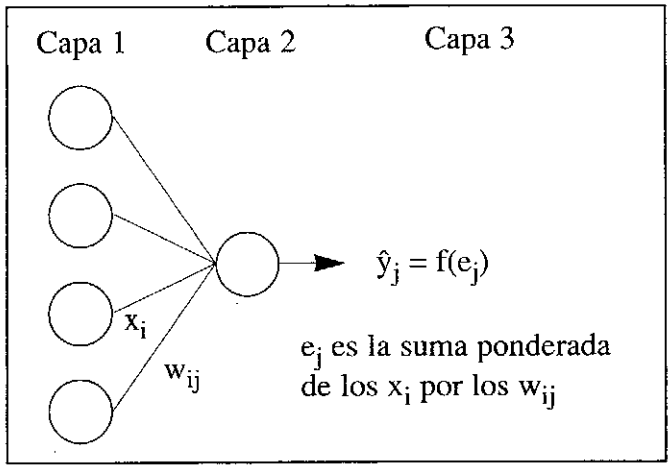
Adoptaremos la siguiente notación:

$y_k(p)$ = componente k-ésima de $y(p) \in \mathbb{R}^m$
= salida objetivo para para el vector de entradas $x(p)$ en el nodo k
$\hat{y}_k(p)$ = salida actual de la red en el nodo k

Cada nodo de la capa oculta de salida tiene la estructura de un perceptrón simple, salvo que la función de activación no es ahora una función escalón, sino que puede ser cualquier función creciente diferenciable. Un ejemplo es la función de activación logística vista anteriormente. El detalle de una de las neuronas de la capa oculta puede verse en la Figura 4.

FIGURA 4

DETALLE DE NEURONA DE LA CAPA OCULTA



La notación es la siguiente:

$$e_j = \sum_{i=1}^{n} w_{ij} x_i \text{: entrada al nodo j de la capa oculta} \quad (3)$$

$$\hat{y}_j = f_j(e_j) \text{: salida del nodo j de la capa oculta} \quad (4)$$

$$f_j = \text{función de activación del nodo j de la capa oculta} \quad (5)$$

donde hemos prescindido del índice p por simplicidad, y $j = 1, \ldots, r$.

Si, por ejemplo, la función de activación es sigmoide:

$$\hat{y}_j = f_j(e_j) = \frac{1}{1 + e^{-\frac{e_j + \theta_j}{\theta_0}}}$$

Al poner el índice j en la función $f(\cdot)$ estamos enfatizando el hecho de que, en general, la función de activación depende del nodo j via el umbral θ_j. Observemos que, en virtud de la ecuación 3, la función de activación de la capa 1 es siempre la identidad: esto es, su misión es simplemente transmitir promediado (por lo pesos w_{ij}) el vector de entradas x_i a la siguiente capa. Para las neuronas de la capa de salida:

$$e_k = \sum_{j=1}^{r} w_{jk}\hat{y}_j\text{: entrada al nodo k de la capa de salida} \quad (6)$$

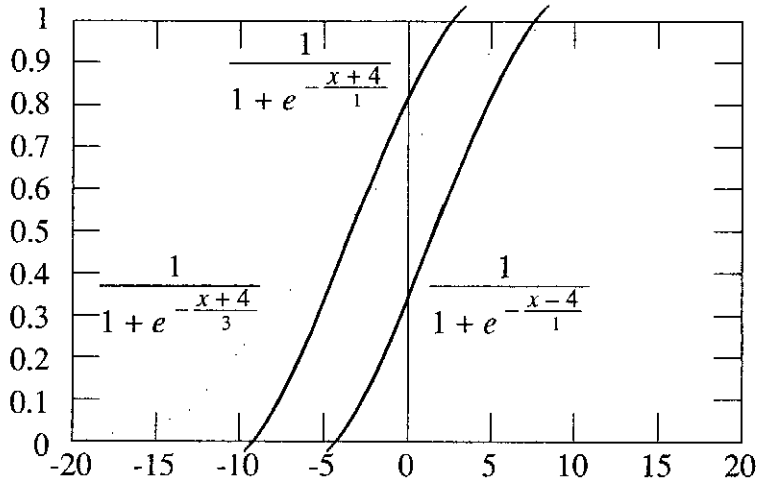
$$\hat{y}_k = f_k(e_k)\text{: salida del nodo k de la capa de salida} \quad (7)$$

$$f_k = \text{función de activación del nodo k de la capa de salida} \quad (8)$$

donde $k = 1, \ldots, m$. Observemos que en la entrada e_k de un nodo de la capa de salida, lo que se pondera no son las variables de entrada x_i; sino las salidas $\hat{y}_j$ de los nodos de la capa anterior (la capa oculta). El efecto de los umbrales es el siguiente: si $\theta_j > 0$, la función de activación se desplaza a la derecha. Así, el umbral θ_j tiene un efecto similar al de la función escalón. Por otra parte, si θ_0 es pequeño, la función de activación se aproxima a la función escalón. Valores grandes de θ_0 hacen que la pendiente sea más suave. La Figura 5 muestra ejemplos de funciones de activación con diversos valores de θ_j y θ_0.

FIGURA 5

EJEMPLOS DE FUNCIONES DE ACTIVACIÓN, MOSTRANDO EL EFECTO DE LOS UMBRALES



La fase de aprendizaje en el perceptrón multicapa se lleva a cabo presentando una observación de entrada $x(p)$, y ajustando los pesos w_{ij} y w_{jk}, y los umbrales θ_j y θ_k de modo que las salidas de la red $\hat{y}_k(p)$ se aproximen lo más posible a las salidas objetivo $y_k(p)$.

Para cada dato p, $p \in \{1, \ldots, N\}$, formado por las variables de entrada $x(p)$ y las variables de salida $y(p)$, se define el error cuadrático por:

$$E(p) = \frac{1}{2}\sum_{k=1}^{m} (y_k(p) - \hat{y}_k(p))^2$$

El error cuadrático medio, para el conjunto de todas las observaciones, se define por:

$$E = \frac{1}{N}\sum_{p=1}^{n} E(p) = \frac{1}{2N}\sum_{p=1}^{N}\sum_{k=1}^{m} (y_k(p) - \hat{y}_k(p))^2 \quad (9)$$

El aprendizaje perfecto se produce cuando $E(p) = 0$ para cada p. El algoritmo de aprendizaje que se utiliza en el modelo básico es la *regla delta generalizada* de Rumelhart,

Hinton y Williams (Rumelhart, *et. al.* 1986). Esta regla busca variar los pesos de modo que se reduzca el error $E(p)$ lo más rápidamente posible. Para ello se presenta una observación cada vez, y se corrigen los pesos para minimizar $E(p)$.

Observemos que la función E depende de los pesos: $E = E(w)$. En efecto, los $\hat{y}_k$ que aparecen en la ecuación 9 pueden ser expresados, en función de los pesos, utilizando las fórmulas 3 a 8, por:

$$\hat{y}_k = f_k(e_k) = f_k(\sum_{j=1}^{r} w_{jk}\hat{y}_j) = f_k(\sum_{j=1}^{r} w_{jk} f_j(ej)) = f_k(\sum_{j=1}^{r} w_{jk} f_j(\sum_{i=1}^{n} w_{ij} x_i)) \qquad (10)$$

Comos los x_i y los y_i son constantes, a efectos de minimizar E, sustituyendo el valor de los $\hat{y}_k$ dado por la ecuación 10 en la ecuación 9, vemos que E es función de los w. Entonces podemos comenzar con unos valores iniciales para los pesos w (al azar, por ejemplo), y hacer cambios en la dirección en la que E decrece más rápidamente. E es una función diferenciable y decrecerá, por tanto, más rápidamente en la dirección opuesta al gradiente, esto es, en la dirección de $\nabla_w E$. Iterando este proceso se generará una sucesión de valores:

$$w_{jk}(t+1) = w_{jk}(t) + \Delta w_{jk}(t)$$

donde

$$\Delta w_{jk}(t) = \eta \frac{\partial E}{\partial w_{jk}}$$

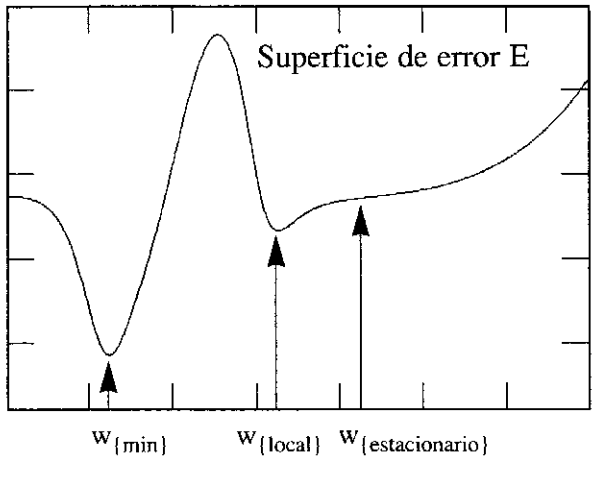
Puede verse que las derivadas parciales del error de un nodo de la capa oculta pueden ser evaluadas en función de la capa de salida. Por ello, empezando en la capa de salida, pueden evaluarse estas derivadas y luego «propagar» el error hacia atrás, hacia la capa de entrada. De ahí que a este algoritmo de aprendizaje se le llame de *retropropagación supervisada*. Los detalles del algoritmo básico pueden consultarse en Rumelhart, *et. al.* (1986).

Resumiento, el algoritmo de adoptación de pesos de la red modifica los pesos w_{ij} buscando el mínimo de una función de error E no definida explícitamente. La situación se ilustra en la Figura 6, donde pueden verse los tres casos posibles en los que el algoritmo de adaptación de pesos convergería.

FIGURA 6

E COMO FUNCIÓN DE LOS PESOS

Problemas en el aprendizaje



Para evitar caer en mínimos locales de la función de error E o salir de ellos, hay diversas modificaciones sobre el algoritmo básico, que serán utilizadas en los experimentos de este trabajo, a fin de obtener un conjunto de pesos w_{ij} óptimo.

3. Imputación de datos mediante redes neuronales

El algoritmo de retropropagación presentado en la sección anterior permite obtener los valores previstos por la red para las variables de interés $y_1, y_2, \ldots, y_m$. La diferencia esencial entre la red y otros métodos paramétricos, es que la red no realiza ninguna hipótesis distribucional sobre los datos. Aprende con el grupo de datos que tiene la o las variables de interés completas y, una vez obtenidos de ese aprendizaje los pesos w_{ji}, al presentarle los datos con variables faltantes o censuradas, la red hace previsiones.

Se puede demostrar que la salida de una red bien entrenada, es decir el vector de variables respuesta, converge a la esperanza de dicho vector condicionada al valor de las variables de entrada. Matemáticamente, llamando $\hat{y}_k(\mathbf{x}, w^*)$ a la salida k-ésima de la red entrenada, donde w^* es el conjunto de los pesos tras el entrenamiento, se verifica:

$$\hat{y}_k(x, w^*) = E(y_k \mid x) = \int y_k f(y_k \mid x) dy_k \quad , \; k = 1, \ldots, m \tag{11}$$

donde $f(y_k \mid x)$ es la función de densidad condicionada de y_k dado x. Concretamente, si tenemos una única variable respuesta, y, el valor que la red predecirá es el valor esperado de y condicionado al conjunto de variables independientes x. Para llegar a esta conclusión, no es necesario conocer la relación funcional entre las x y las y. Los detalles pueden consultarse en Bishop (1995). Es interesante resaltar que este resultado es válido en el caso multivariante, es decir si tenemos que predecir varias variables conjuntamente.

4. Resultados

El procedimiento descrito se ha aplicado a Edades de Jubilación de hombres de edad avanzada. La EPA del segundo semestre del año recoge datos sobre la situación del entrevistado un año antes. Así, es posible obtener una muestra de personas que, estando jubiladas en el momento de la encuesta, eran activas un año antes. Consecuentemente, para este grupo de individuos se conoce la edad exacta de jubilación. Además de este grupo de personas en la encuesta encontramos individuos activos e individuos ya jubilados.

En resumen, con los datos de la EPA es posible encontrar tres grupos de interés para el estudio del tránsito a la jubilación:

- *Activos:* Personas que en el momento de la encuesta están activas. Constituyen un grupo de datos censurados por la derecha pues su edad de jubilación, $T \geq EDAD$.
- *Jubilados de un año:* (JA) Personas que están jubiladas en el momento de la encuesta pero que un año antes estaban activas. Su edad de jubilación, T, se supone igual a su edad.
- *Jubilados:* Personas que llevan más de un año jubiladas. Constituyen un grupo de datos censurados por la izquierda ya que $T < EDAD - 1$.

El ajuste de modelos a este tipo de datos debe hacerse mediante modelos de duración (Lawless, 1982; Villagarcía, 1995). Es importante resaltar que no existen buenas técnicas de estadística descriptiva para este tipo de datos, por lo que el ajuste de modelos es muy laborioso. Conseguir transformar los datos en datos completos es por tanto una buena alternativa.

Para realizar esta labor se han empleado históricamente técnicas paramétricas como el algoritmo EM (Schafer, 1997). Sin embargo, la aplicación del algoritmo exige realizar importantes asunciones distribucionales que, sin técnicas descriptivas aceptables, son muy discutibles.

En este trabajo vamos a utilizar un perceptrón multicapa, para estimar la Edad de Jubilación de los datos incompletos del grupo de trabajadores activos. La técnica es no paramétrica, y por tanto no debemos asumir ningún tipo de modelo a priori.

El grupo de trabajadores jubilados no lo vamos a incluir en el estudio, aunque el procedimiento es análogo, ya que la EPA carece de datos de muchas variables relevantes para este grupo (por ejemplo no se conoce el sector de actividad en que trabajaba, ni si era asalariado o autónomo, etc.).

4.1. *Imputación*

La primera fase del análisis consiste en presentar a la red las observaciones completas, es decir la muestra de jubilados de un año definida al inicio de esta sección. Hemos estudiado una muestra de 1.431 jubilados correspondientes al año 1987, de los que conocemos la edad exacta de jubilación. La variable que queremos predecir es la Edad de Jubilación, Y. Se dispone de 21 variables independientes X_i. Estas variables incluyen Años de estudio, Sector de actividad, Estado civil, Asalariado o no asalariado, etc.

Para entrenar la red, se han utilizado 1.288 datos elegidos al azar. Las 143 observaciones restantes se han utilizado para contrastar las predicciones que proporciona la red. Es importante notar que estas 200 observaciones no se han utilizado para el aprendizaje o ajuste de pesos.

Se ha utilizado una red con 21 unidades de entrada (una para cada variable independiente X_i) 5 unidades ocultas y una unidad de salida, correspondiente a la edad de jubilación, Y. La función de transferencia utilizada en todas las neuronas ha sido la sigmiode.

Con esta muestra la red aprende y ajusta sus pesos. La Tabla 1 presenta los resultados de este proceso de aprendizaje para las edades centrales del proceso de jubilación. Como puede observarse, la red predice correctamente en un número importante de casos. Es importante señalar que las variables X utilizadas como entrada son en general no significativas, por lo que la red está prediciendo los valores utilizando básicamente un subgrupo de variables de entrada.

A continuación se ha presentado a la red entrenada anterior el conjunto de 11.103 trabajadores activos de 1987. Estos son datos censurados de los que se conoce su edad actual pero no su edad de jubilación.

El histograma de la Figura 7 presenta las Edades de estos trabajadores, es decir la edad de jubilación censurada por la derecha. Como puede observarse, la Edad censurada no representa en absoluto el fenómeno que estamos estudiando. De hecho, la jubilación se

observa en este histograma en el número decreciente de activos a medida que la edad crece. A partir de 64/65 años el número de activos decrece de forma clara.

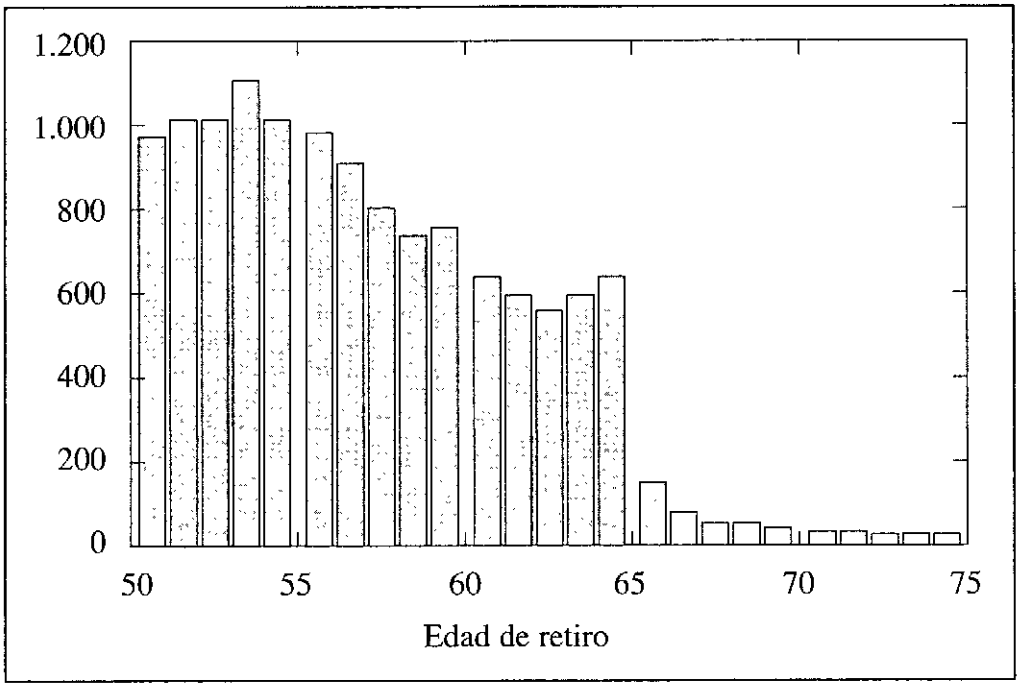
TABLA 1

RESULTADOS DE UNA RED MULTICAPA SOBRE LOS DATOS COMPLETOS DE LA EPA

EDAD	Número de Observaciones	Estimadas con menos de dos años de diferencia	Porcentaje de Aciertos
59	71	32	45
60	159	89	56
61	83	53	64
62	75	61	81
63	74	58	78
64	173	160	92
65	504	460	91
66	72	61	85
67	26	18	69
68	19	10	53
69	12	0	0
70	13	3	23

FIGURA 7

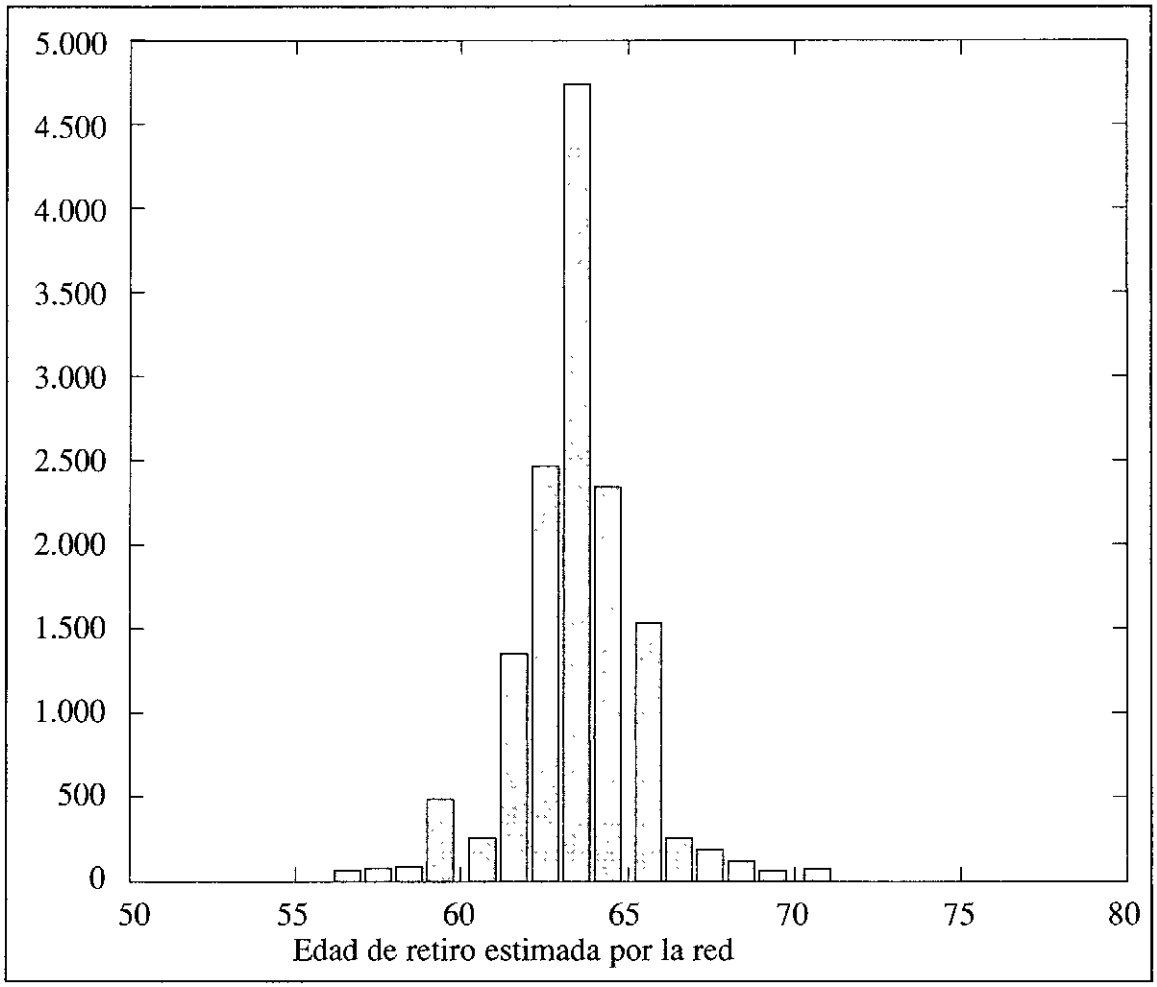
HISTOGRAMA DE EDADES DE RETIRO CENSURADAS



La Figura 8 muestra el histograma de edades de jubilación estimadas por la red. Se ha utilizado la red entrenada con los datos completos cuyas características se han resumido en la Tabla 1. Como puede comprobarse, el histograma obtenido es mucho más fiel al fenómeno que estamos estudiando.

FIGURA 8

HISTOGRAMA DE EDADES DE RETIRO ESTIMADAS



Finalmente hemos realizado un ejercicio para verificar la bondad del análisis. Hemos ajustado un modelo de regresión Normal para los datos sin imputar, es decir considerando observaciones completas y censuradas simultáneamente, y hemos ajustado el mismo modelo para datos completos imputados. El modelo ajustado para los datos censurados tiene la siguiente función de verosimilitud,

$$L = \sum_{i \in C} f(y_i / x\beta) \sum_{i \in A} (1 - F(y_i / x\beta))$$

donde $f(y_i / x\beta)$ representa la función de densidad Gaussiana, $F(y_i / x\beta)$ la función de distribución, C representa el conjunto de observaciones completas y A el conjunto de trabajadores activos, es decir de observaciones censuradas.

En el caso de datos completos el modelo anterior queda reducido a una ecuación de regresión:

$$y_i = x\beta + u_i$$

donde $u_i \sim N(0, \sigma^2 I_N)$. Las previsiones realizadas por ambos modelos han resultado ser muy semejantes. De hecho, la correlación entre ambos vectores de valores previstos ha resultado ser de 0,9627.

Podemos concluir que las previsiones de la red no han sido inferiores a las del modelo paramétrico.

5. Conclusiones

En este trabajo se ha utilizado una red neuronal para prever el valor de una variable censurada. El tratamiento de este tipo de datos mediante modelos de supervivencia es una técnica compleja que requiere una laboriosa preparación de los datos. Un problema adicional es la falta de técnicas descriptivas adecuadas para el análisis preliminar de datos. En determinados casos, como cuando existe censura por la derecha y por la izquierda simultáneamente, las técnicas disponibles no son aplicables.

En este contexto el uso de redes neuronales para predecir los valores de las variables censuradas puede ser muy interesante, pues permitiría utilizar las técnicas estadísticas estándar para tratar datos complejos.

El estudio realizado indica que la red puede ser una eficaz herramienta en el análisis de datos de duración, pues los resultados obtenidos son semejantes a los encontrados en un ajuste paramétrico.

Referencias bibliográficas

[1] BISHOP, C. M. *Neural Networks for Pattern Recognition.* Oxford University Press, New York, 1995.

[2] LAWLESS, J. F. *Statistical Models and Methods for Lifetime Data.* John Wiley & Sons, New York, 1982.

[3] RUMELHART, D. E. and J. L. McCLELLAND (eds.). *Parallel Distributed Processing: Explorations in the Microstructures of Cognition vol. 1.* MIT Press, Cambridge, 1986.

[4] SCHAFER, J. L. *Analysis of Incomplete Multivariate Data.* Chapman & Hall, London, 1997.

[5] VILLAGARCÍA, T. «Análisis econométrico del tránsito a la jubilación para trabajadores de edad avanzada». *Investigaciones Económicas,* XIX(1):65-81, 1995.