

# Modelos de regresión lineal con redes neuronales

## Lineal regression models with neural networks

César Menacho Ch.<sup>1</sup>

### Resumen

Los modelos de redes neuronales pueden ser considerados como nuevos paradigmas para el análisis estadístico de regresión lineal. Una de las razones del uso de las redes neuronales es que no necesitan el cumplimiento de supuestos teóricos como en los modelos estadísticos clásicos. El modelo del Perceptron multicapa es equivalente a un modelo de regresión lineal, debido a la similitud de la variable de salida (variable respuesta Y) que se relaciona aplicando la función de activación (función identidad) sobre una combinación lineal de pesos (coeficientes) con las variables de entrada (variables predictoras). El objetivo de la investigación es presentar y comparar una metodología para ajustar los modelos estadísticos de regresión lineal simple, múltiple y multivariado con el modelo de red neuronal Perceptron multicapa. Los resultados muestran que las ecuaciones de predicción estimadas con el modelo neuronal Perceptron multicapa tuvieron un mayor  $R^2$ . En la comparación de la regresión lineal simple, en el 61,2 % de los conjuntos de datos simulados resultaron los modelos de redes neuronales con mayores  $R^2$ .

**Palabras clave:** modelos de regresión lineal; redes neuronales; red neuronal perceptron; red neuronal multicapa.

### Abstract

The neural network models can be considered as new paradigms for statistical analysis of linear regression. One of the reason to use neural networks is that they don't need to comply with theoretical assumptions as in classical statistical models. The multilayer Perceptron model is equivalent to a linear regression model, due to the similarity of the output variable (response variable Y) that is related by applying the activation function (identity function) on a linear combination of weights (coefficients) with input variables (predictors). The objective of the research is to present and compare a methodology to adjust simple linear regression statistical models, multiple and multivariate with the Perceptron Multilayer Neural Network model. The results show that the prediction equations estimated with the Multilayer Perceptron neural model had a higher  $R^2$ . In the comparison of simple linear regression, 61,2 % of the simulated datasets resulted in neural network models with higher  $R^2$ .

**Keywords:** linear regression models; neural networks; perceptron neural network; multilayer neural network.

### 1. Introducción

Recientemente, se han realizado diversos trabajos de investigación, que analizan las relaciones entre los modelos estadísticos clásicos y los modelos de redes neuronales. Los modelos de redes neuronales pueden ser aplicados en el análisis estadístico clásico concernientes a los problemas de predicción (análisis de regresión) y clasificación (análisis de discriminación). Para los problemas de predicción, los modelos de redes neuronales son aplicados en forma similar a una variedad de modelos estadísticos tales como los modelos de regresión lineal, los lineales generalizados, los de regresión no paramétrica y los de análisis discriminante; pero, con la limitación de no poder realizar un análisis explicativo del modelo ajustado.

Entre las propiedades que se destacan en el uso de las redes neuronales con respecto a los modelos estadísticos clásicos, es su aplicación sin la necesidad de considerar

el cumplimiento de supuestos teóricos, requisitos necesarios para el modelamiento estadístico; es así que las redes neuronales en la literatura estadística son definidas como técnicas no paramétricas. Los modelos de redes neuronales son considerados nuevos paradigmas alternativos para el análisis estadístico predictivo; al igual que los modelos estadísticos, estos permiten describir muchos fenómenos del mundo real. Por tal motivo, se hace necesario realizar investigaciones que permitan evaluar y comparar los resultados obtenidos con las redes neuronales y los modelos estadísticos clásicos.

En los diversos estudios comparativos que se han realizado entre los modelos estadísticos y las redes neuronales en problemas de predicción, clasificación y análisis descriptivo, se muestran resultados similares y algunos contrapuestos sobre el rendimiento de ambas técnicas. Así, mientras algunos trabajos no encuentran diferencias en los resultados entre los modelos estadísticos

1. Profesor Principal del Departamento de Estadística Informática. Universidad Nacional Agraria La Molina, Lima, Perú. Email: [cmenacho@lamolina.edu.pe](mailto:cmenacho@lamolina.edu.pe)

y los modelos de redes neuronales (Michie *et al.*, 1994; Ripley, 1993; Thrun *et al.*, 1991), otros resultados tienden a favorecer una ligera superioridad de las redes neuronales sobre las técnicas estadísticas (Garson, 1991; Huang y Lippman, 1987; Navarro, 1998; White, 1994). Pitarque *et al.* (2000) comparan, mediante simulación, las redes neuronales del tipo perceptron multicapa (PM) con modelos estadísticos (regresión múltiple, análisis discriminante y regresión logística) en problemas de predicción y clasificación binaria, considerando diferentes grados de correlaciones entre los predictores (variables de entrada) y entre predictores con la variable criterio (variable de salida). Los resultados muestran que en la tarea de predicción los modelos de redes neuronales y regresión múltiple dan similar resultado. Mientras en los problemas de clasificación, los modelos neuronales dan mejores resultados que el análisis discriminante y regresión logística.

Sarle (1994) considera que los modelos lineales generalizados pueden ser vistos como un perceptron multicapa. En los modelos lineales generalizados, los datos son ajustados por el método de máxima verosimilitud para estimar los parámetros del modelo, mientras que las redes neuronales son usualmente entrenadas por mínimos cuadrados. La función de activación en un perceptron es análoga a la inversa de la función de enlace en un modelo lineal generalizado. Chattersje y Laudato (1995) comparan con tres ejemplos de aplicación la estimación con redes neuronales para funciones sigmoideal, exponencial y logística y los resultados ajustados con los modelos estadísticos de regresión clásicos.

En la presente investigación se expondrán, en primer lugar, los fundamentos para el modelamiento con las redes neuronales usando el perceptron multicapa, que es el modelo de red neuronal de mayor aplicación. Se desarrollará el marco metodológico para ajustar los modelos estadísticos de regresión lineal simple, múltiple y multivariado con redes neuronales. Para comparar y evaluar los modelos de regresión estadísticos con los de redes neuronales, se aplicará la metodología propuesta a conjuntos de datos reales y simulados.

El objetivo de esta investigación es presentar una metodología y su aplicación para ajustar modelos estadísticos de regresión lineal con las redes neuronales. Así como evaluar y comparar ambas técnicas usando diversos conjuntos de datos.

## 2. Materiales y métodos

### Materiales y equipo

Para evaluar y comparar la metodología propuesta sobre el análisis de regresión lineal con redes neuronales, se realizarán cuatro aplicaciones utilizando conjuntos de datos reales y simulados. Para la aplicación de la regresión lineal simple, se usarán los datos sobre generación de electricidad (Y) por molinos de viento y la velocidad del viento (X). Para la regresión lineal múltiple, se tomarán en cuenta los datos sobre calor generado por gramo de cemento (Y) en función de cuatro ingredientes: tricalcio

de aluminato (X1), tricalcio de silicato (X2), tetracalcio de aluminio ferroso (X3) y dicalcio de silicato (X4). Para el caso de la regresión multivariada, se tienen los datos de un estudio sobre una colección de germoplasma de lenteja de la India con 16 accesiones, con la finalidad de analizar la relación entre las características productivas: Y1 (rendimiento por planta kg/ha) y Y2 (altura de la planta cm) y morfológicas: X1 (largo de vainas), X2 (vainas por planta) y X3 (tallos secundarios). Con el propósito de comparar la regresión lineal simple clásica con redes neuronales, se simularán y se ajustarán 500 conjuntos de datos. Para obtener los resultados, se usará el software libre Weka (Universidad Weikato, Nueva Zelanda) desarrollado en Java para técnicas de minería de datos.

### Metodología

La literatura sobre modelos estadísticos y redes neuronales muestran que muchos conceptos teóricos son análogos y la terminología empleada es muy similar, variando solo en los términos o acrónimos utilizados para especificar muchas veces el mismo concepto para ambos métodos (Sarle, 1994; Vicino, 1998).

Masters (1993), en su investigación, sostiene que los supuestos de normalidad, homogeneidad de variancias y aditividad en las variables de entrada son características recomendables para una red neuronal y no son estrictamente necesarias, como sucede en los modelos estadísticos. En la Tabla 1 se presenta la terminología similar entre los modelos estadísticos y redes neuronales.

Las redes neuronales pueden ser consideradas como modelos de regresión lineales o no lineales. La aplicación de los modelos estadísticos clásicos a problemas predictivos, tales como los modelos de regresión lineal simple, regresión lineal múltiple, regresión polinomial, regresión multivariada, regresión no paramétrica,

**Tabla 1.** Terminología similar entre los modelos estadísticos y redes neuronales

Modelos estadísticos	Modelos de redes neuronales
Observación	Patrón
Muestra	Datos de entrenamiento
Muestra de validación	Datos de validación y prueba
Variables explicativas	Variables de entrada
Variable dependiente	Variable de salida
Modelo	Arquitectura
Residual	Error
Error aleatorio	Ruido
Estimación	Entrenamiento o Aprendizaje
Interacción	Conexión funcional
Coeficientes	Pesos de conexión
Intercepto	Peso umbral o sesgo

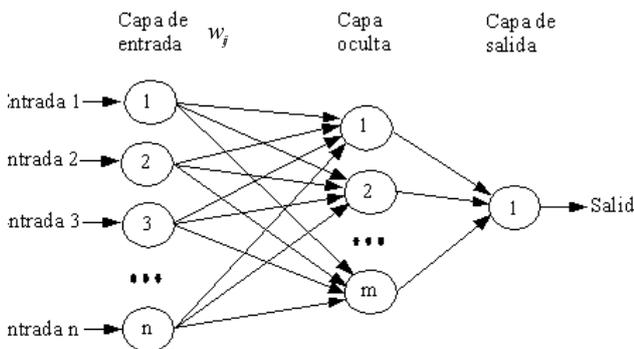
regresión no lineal y, en forma más general, como modelos lineales generalizados, pueden ser analizados en forma similar con el modelo de red neuronal del perceptron.

**Fundamentos de las redes neuronales**

Las redes neuronales artificiales (RNA) son modelos matemáticos y computacionales inspirados en sistemas biológicos, adaptados y simulados en computadoras; han surgido como un intento de desarrollar modelos que emulen las características del cerebro humano. Las RNA están compuestas por un conjunto de elementos simples (neuronas) que se interconectan masivamente en paralelo y con organización jerárquica. Asimismo, las redes neuronales artificiales conforman modelos que permiten describir muchos fenómenos del mundo real y que pueden ser comparados y aplicados con los modelos estadísticos en problemas de predicción, clasificación y agrupamiento.

**Componentes de una red neuronal**

Una RNA está compuesta por un conjunto de unidades de procesamiento simples (neuronas), conectadas entre sí y por donde se envía la información necesaria. El modelo de una RNA se basa en la estructura de un grafo dirigido cuyos nodos representan las neuronales que están interconectadas a través de arcos dirigidos. Cada arco dirigido establece la relación entre nodos  $j \rightarrow i$  y tiene asociado un peso numérico  $w_{ji}$  que determina la fuerza y el signo de la conexión y sirve para propagar la activación de la neurona  $j$  hacia la neurona  $i$ . En la **Fig. 1** se muestra el esquema de una red neuronal con  $n$  neuronas en la capa de entrada,  $m$  neuronas en la capa oculta y una neurona en la capa de salida.



**Figura 1.** Red neuronal

**Neuronas.** Son los elementos o unidades de procesamiento de una RNA. Cada neurona  $i$ -ésima en cada instante tiene un valor numérico o estado de activación. Cada neurona puede tener múltiples entradas y opcionalmente una entrada constante (sesgo o umbral).

**Conexiones entre neuronas.** Las neuronas artificiales están conectadas a través de arcos dirigidos. Cada arco dirigido, tiene asociado un peso numérico  $w_{ij}$  que es el parámetro que refleja la influencia y fuerza de la sinapsis de la conexión.

- Si  $w_{ij}$  es positivo, indica que la interacción (sinapsis) entre la neurona  $i$  y  $j$  y es inactivada; es decir, siempre que la neurona  $i$  esté desactivada, la neurona

recibirá una señal de  $j$  que tenderá a activarla.

- Si  $w_{ij}$  es negativo, la sinapsis es inhibitoria; lo cual indica que si  $i$  está activada, enviará una señal a  $j$  que tenderá a desactivarla.
- Si  $w_{ij}$  es cero, se supone que no hay conexión entre ambas neuronas.

**Capas.** Las neuronas se organizan por niveles o capas con un número determinado de neuronas en cada una de ellas y que describen la forma en que estas se interconectan entre sí. En una RNA, se pueden distinguir tres tipos de capas:

- **Capa de entrada.** Es la capa cuyos nodos (**inputs**) reciben directamente la información proveniente de las fuentes externas de la red (variables de entrada).
- **Capas ocultas.** Son internas a la red y no tienen contacto directo con el entorno exterior. Las neuronas de esta capa pueden estar interconectadas de distintas maneras, lo que determina, junto con su número, las diferentes topologías de las RN.
- **Capa de salida.** Los nodos (**outputs**) de esta capa permiten realizar la transferencia de información de la red hacia el exterior (variables de salida).

**Función de entrada.** Una neurona recibe un conjunto de señales que representa la información del estado de activación de todas las neuronas con las que se encuentra conectadas. Sea  $i$  el valor de entrada de la neurona  $i$  en un instante y la conexión entre la neurona  $x_i$  y  $j$  está ponderada por un peso  $w_{ij}$ . Entonces considerando un efecto aditivo, la entrada neta que recibe una neurona  $j$  está dada por la expresión:

$$Net_j = w_0 + \sum_{i=1}^N w_{ji} x_i$$

**Función de Salida.** Conocida también como función de transferencia o activación y es en parte responsable de la bondad del comportamiento de la red neuronal. La salida de una neurona se denota por  $y$  y se obtiene mediante la aplicación de la función de transferencia. Entre las funciones de activación o transferencia más usadas tenemos:

- Umbral o escalón:  $y_j = \begin{cases} 0 & \text{si } Net_j < 0 \\ 1 & \text{si } Net_j \geq 0 \end{cases}$

- Lineal o identidad:  $y_j = Net_j$

- Logística:  $y_j = \frac{1}{1 + e^{-Net_j}}$

- Tangente hiperbólico:  $y_j = \frac{e^{Net_j} - e^{-Net_j}}{e^{Net_j} + e^{-Net_j}}$

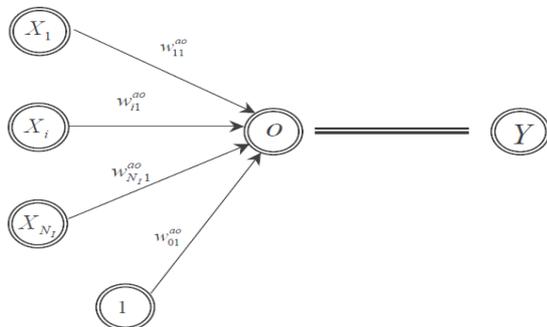
- Gaussiana:  $y_j = e^{-\frac{Net_j^2}{2}}$

Algunas de estas funciones tienen que llevar el conjunto de entrada a un espacio acotado, donde las funciones presentan cualidades que resultan de interés para un estudio determinado. Asimismo, cada nodo puede tener una función de transferencia propia distinta a los demás

nodos; sin embargo, en la mayoría de los casos, los nodos situados en la misma capa presentan todos la misma función de transferencia.

**El modelo red neuronal perceptron**

El perceptron simple consta de una capa de entrada, una capa oculta y una de salida; es un modelo de gran uso y resulta eficiente para modelar muchas situaciones del mundo real. La estructura del perceptron está conformada por una capa de entrada con una neurona, que permite recibir los valores de las variables predictoras; una neurona en la capa oculta, donde se realiza la acumulación de la entrada (entrada neta) y la aplicación de la función de transferencia (generalmente una función umbral) para obtener la salida esperada, y la capa oculta de salida donde se compara la salida esperada con la salida observada; la diferencia sería el cálculo del error o residual (**Fig. 2**).



**Figura 2.** El perceptron simple

La única neurona de salida del perceptron realiza la suma ponderada para obtener la entrada neta y restar un valor de umbral, el resultado se aplica a la función de transferencia tipo escalón. Los valores de los pesos pueden fijarse o hallarse utilizando diferentes algoritmos de entrenamiento de la red.

El aprendizaje del perceptron simple es el proceso de entrenamiento del mismo perceptron. Consiste en calcular la combinación lineal a partir de un conjunto de variables de entrada (patrones de entrenamiento), con un término de sesgo, aplicándole una función de activación, por regla general la función umbral o signo, dando lugar a la salida de la red. Así, los pesos de la red se van ajustando por el método de aprendizaje supervisado por corrección de error, de tal manera que se va comparando la salida esperada con el valor de la variable de salida que se desea obtener, la diferencia es el error o residual. El aprendizaje termina cuando, al considerar cada valor de las variables de entrada, los residuales son cero o muy próximos a cero.

Las ecuaciones del proceso de aprendizaje son:

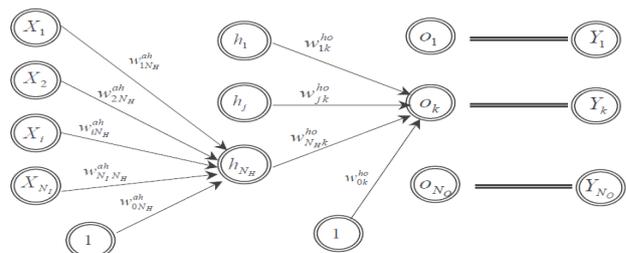
$$q = w_{01}^{a0} + \sum_{i=1}^{N_i} w_{i1}^{a0} * x_i$$

$$O = f(q) \begin{cases} 0 & \text{si } q < 0 \\ 1 & \text{si } q \geq 0 \end{cases}$$

Está constituido por  $N_i$  nodos de entrada y una única neurona de salida, encargada de decidir a cuál de las dos clases posibles pertenece la observación.

El perceptron multicapas (MLP) es una generalización del perceptron simple en que se considera más de una capa oculta. El perceptron simple, con solo dos capas, se limita a resolver problemas con observaciones separadas geoméricamente por hiperplanos. Ante este inconveniente, surgieron dos modelos nuevos, el ADALINE (elemento lineal adaptable) y el MADALINE (elemento lineal adaptable múltiple). La estructura de ADALINE es la del perceptron simple, pero la función que se aplica en el nodo es la identidad; esto permite más flexibilidad en la estructura de la red. Es de tipo feedforward y está compuesta por varias capas ocultas de neuronas entre la capa de entrada y salida, la función de transferencia es continua, no lineal y diferenciable.

El MLP consiste en un conjunto de nodos organizados por capas, de modo que una neurona puede recibir entradas solo de aquellas situadas en la capa inmediatamente inferior. En general, en un MLP cada uno de los nodos calcula una combinación lineal de las entradas que llegan a él, le añade un sesgo y, finalmente, le aplica una función de activación, llamada también de transferencia, que por regla general traslada cualquier entrada real a un rango generalmente acotado, dando lugar así a la salida del nodo, que puede ser una de las entradas de un nuevo nodo. El perceptron simple como multicapas puede tener una o más salidas, cada una de ellas con un conjunto de pesos y un sesgo asociados. Generalmente, se considera una misma función de activación o transferencia para cada nodo de la misma capa, aunque es posible usar diferentes funciones de activación para cada neurona (**Fig. 3**).



**Figura 3.** Red neuronal para una regresión lineal multivariada

Las ecuaciones del proceso de aprendizaje son:

$$h_j = f_h \left( w_{0j}^{ah} + \sum_{i=1}^{N_i} w_{ij}^{ah} * x_i \right) \quad \text{para } j = 1, 2, \dots, N_H$$

$$O_k = f_o \left( w_{jk}^{ho} + \sum_{j=1}^{N_H} w_{jk}^{ho} * h_j \right) \quad \text{para } k = 1, 2, \dots, N_O$$

**Modelos de regresión lineal con redes neuronales**

Una diferencia importante entre los modelos estadísticos y las redes neuronales es el método de estimación de los coeficientes utilizado para minimizar la función de error. Mientras el perceptron normalmente estima los

parámetros (pesos) mediante el criterio de mínimos cuadrados reduciendo la función de error (White, 1989; Cheng y Titterton, 1994; Ripley, 1994), el MLG ajusta el modelo mediante el método de máxima verosimilitud para una variedad de distribuciones de la clase exponencial (Sarle, 1994). Sin embargo, Bishop (1995), entre otros, indica que el criterio de mínimos cuadrados, asumiendo un error con distribución normal, obtiene estimaciones máximo-verosímiles, tal como ocurre en el modelo lineal general. Otra importante diferencia entre RNA y modelos estadísticos consiste en que los parámetros obtenidos por la red neuronal no son susceptibles de una interpretación práctica. No podemos saber inmediatamente cómo los pesos de la red o los valores de activación de las neuronas están relacionados con el conjunto de datos manejados. Así, a diferencia de los modelos estadísticos clásicos, no parece tan evidente conocer en una red el efecto que tiene cada variable explicativa sobre la variable de respuesta. Por tanto, es importante tener en cuenta que las similitudes que se puedan establecer entre RNA y modelos estadísticos siempre harán referencia al aspecto predictivo, pero no al aspecto explicativo. El problema acerca del análisis del efecto de las variables de entrada en una red neuronal constituye una línea de investigación de interés. Bishop (1995) sostiene que las redes neuronales y los modelos estadísticos asumen exactamente los mismos supuestos en cuanto al tipo de distribución; lo que sucede es que en la estadística se estudian las consecuencias del incumplimiento de los supuestos, mientras que los investigadores en redes neuronales simplemente las ignoran.

En el perceptron se usa el concepto de función de error como una medición de la discrepancia entre lo predicho por la red neuronal y lo observado, lo que es equivalente al error residual de una regresión lineal. En el perceptron, lo que se intenta, en general, es minimizar la suma del error cuadrático:

$$E = \sum_{p=1}^P \frac{1}{2} \sum_{k=1}^M (o_{pk} - y_{pk})^2$$

donde P es el número de patrones, M número de neuronas de salida,  $o_{pk}$  es la salida deseada para la neurona de salida k para el patrón p e  $y_{pk}$  es la salida obtenida por la red para la neurona de salida k para el patrón p.

Las redes neuronales pueden ser entendidas como modelos generales de regresión (Haykin, 1999). Se utilizan para predecir valores futuros de una o varias variables objetivo (variables dependientes o respuestas). Los modelos clásicos de regresión lineal y no lineal pueden ser modelados y analizados a través de las redes neuronales. La forma general para representar un modelo de regresión lineal como una red neuronal se basa en considerar n variables independientes  $x = (x_1, x_2, \dots, x_n)$  (variables de entrada) y m variables dependientes  $y = (y_1, y_2, \dots, y_m)$  (variables de salida) pertenecientes a un conjunto de datos D (conjunto de aprendizaje); definido este conjunto de datos de la forma  $(x, y) \in D$ , tal que  $x \in R^n$  e  $y \in R^m$ .

Entonces, la neurona artificial como la unidad básica de una red neuronal, puede ser expresada como una

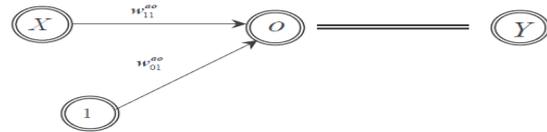


Figura 4. Red neuronal para una regresión lineal simple

función  $f : R^n \rightarrow R^m$ . Esta función se denomina función de salida o transferencia, que transforma la entrada  $x = (x_1, x_2, \dots, x_n)$  obteniendo la salida  $o = (o_1, o_2, \dots, o_n)$  tal que se cumpla que  $f(x) = y, \forall (x, y) \in D$ , siendo el error o residual  $e = (o - y)$ .

Los modelos de regresión lineal simple, múltiple o multivariado pueden ser identificados como un modelo de red neuronal perceptron multicapa. Si el modelo neuronal posee una variable de entrada y una de salida, se tiene una regresión lineal simple. Si hay varias variables de entrada y solo una de salida, el modelo neuronal corresponde al modelo de regresión lineal múltiple. Si hay más de una neurona de salida y de entrada, la red neuronal corresponde a un modelo de regresión lineal multivariado.

**Regresión lineal simple**

Para modelar una regresión lineal simple se usará la estructura correspondiente a un modelo de red neuronal de perceptron multicapa. La salida de la red neuronal será  $o = w_0^{a0} + w_1^{a0} * x$ , que coincide con la predicción proporcionada por una regresión lineal simple. La estructura del perceptron corresponde a una variable de entrada (x) más una constante (I) (sesgo o umbral), sin capa oculta, pero con un nodo que representa la salida observada (o), una variable de salida (y) con función de activación de la identidad.

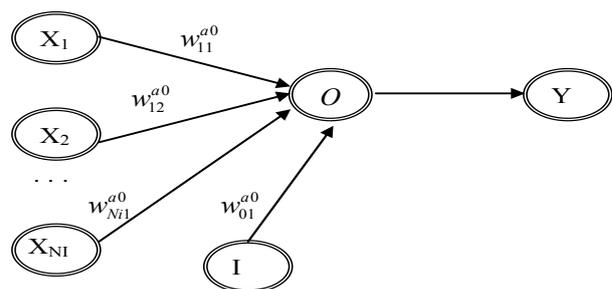


Figura 5. Red neuronal para una regresión lineal múltiple

La ecuación funcional será expresada por:

$$Y = w_{01}^{a0} + w_{11}^{a0} * X + \epsilon$$

Donde Y representa la variable dependiente, X la variable independiente,  $w_{01}^{a0}$  es el sesgo de la capa de salida y  $w_{11}^{a0}$  son los pesos de la entrada i para la salida j. La ecuación de predicción para indicar la salida estimada de la red neuronal será:

$$\hat{y} = w_{01}^{a0} + w_{11}^{a0} * x$$

**Regresión lineal múltiple**

Una regresión lineal múltiple corresponde a un modelo perceptron simple sin capa oculta y función de activación de identidad. En este caso, la red neuronal tendrá tantos nodos de entrada como variables regresoras ( $N_I$ ) y solo un nodo de salida que corresponde a la variable dependiente. El modelo de red neuronal se muestra en el Fig. 5.

La ecuación funcional para representar el modelo estadístico de una regresión lineal múltiple como una red neuronal será la siguiente expresión:

$$Y = w_0 + \sum_{i=1}^{N_I} w_i * X_i + \varepsilon$$

Donde  $w_0$  es el umbral de la neurona de salida  $k$ , que actúa en forma similar a la constante de un modelo de regresión,  $N_I$  es el número de neuronas de entrada,  $w_{ik}$  es el peso entre la neurona de la entrada  $i$  y la neurona de salida  $k$  y  $x_{pi}$  es el valor de la neurona de la entrada  $i$  para el patrón  $k$ . La predicción estimada será:

$$\hat{Y} = w_0^{a0} + \sum_{i=1}^{N_I} w_i^{a0} * x_i$$

**Regresión lineal multivariada**

Para modelar una regresión lineal multivariada donde se tiene más de una variable dependiente, se usará una red neuronal con más de una neurona de entrada y de salida. La ecuación corresponderá a un perceptron multicapa con más de un nodo en la capa oculta y función de activación de la identidad. La salida  $k$ -ésima está definida por:

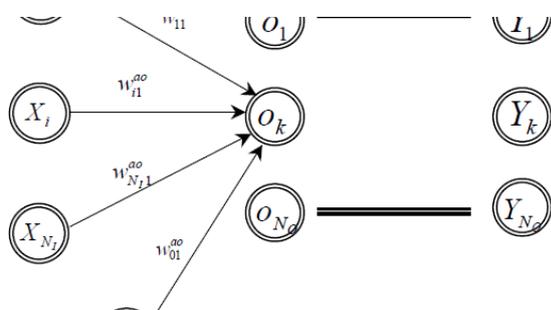
$$O_k = w_{0k}^{a0} + \sum_{i=1}^{N_I} w_{ik}^{a0} * x_i$$

, que corresponde a la red neuronal que tendrá tantos nodos de entrada como variables regresoras ( $N_I$ ), y tantos nodos en la capa de salida como variables dependientes ( $N_0$ ). En la Fig. 6 se muestra la red neuronal para modelar una regresión lineal multivariada. El modelo funcional queda expresado por:

$$Y_k = w_{0k} + \sum_{i=1}^{N_I} w_{ik} * X_i + \varepsilon, \quad \text{para } 1 \leq k \leq N_0$$

Siendo la ecuación estimada de predicción:

$$\hat{Y}_k = w_{0k}^{a0} + \sum_{i=1}^{N_I} w_{ik}^{a0} * X_i + \varepsilon, \quad \text{para } 1 \leq k \leq N_0$$



**Figura 6.** Red neuronal para una regresión lineal multivariada

**Aspectos para el modelamiento con redes neuronales**

- 1) El número de neuronas en la capa de entrada está determinado por el número de variables predictoras. Entonces habrá tantas neuronas de entrada como variables predictoras tenga el modelo de regresión.

- 2) Las variables predictoras categóricas se deben expresar como variables dummy, asociando una neurona a cada variable indicadora; es decir, a las variables predictoras continuas se les asocia una neurona.
- 3) El número de neuronas en la capa de salida está determinada por el número de variables dependientes. Para una regresión lineal simple y múltiple se definirá una sola neurona de salida; para una regresión multivariada el número de neuronas en la capa de salida será igual al número de variables dependientes.
- 4) El número de neuronas en la capa oculta determina la capacidad de aprendizaje de la red, pero se debe tener cuidado con el problema de un ajuste excesivo. Al respecto, los criterios más usados son:  $(Nr. Atributos + Nr. Predictores) / 2$ ,  $Nr. de atributos$ ,  $Nr. de predictores$ .

**3. Resultados y discusión**

Para el análisis de regresión lineal simple con redes neuronales se ajustaron datos reales a la estructura de la red neuronal de un perceptron simple con una neurona de entrada y una neurona de salida que corresponden a la variable independiente y dependiente, respectivamente. Se usó como función de activación la sigmoide en la capa oculta y la identidad en la salida. Las ecuaciones estimadas para el modelo regresión lineal simple (RL) y con la red neuronal (RN), se presentan a continuación:

$$\hat{y}_{RL} = 0,131 + 0,241x$$

$$\hat{y}_{RN} = -1,071 + 3,378 \left( \frac{1}{1 + \exp(-(-1,734 + 0,595x))} \right)$$

En la regresión lineal múltiple se consideró el perceptron multicapa con una neurona en la capa oculta y cuatro nodos de entrada correspondientes a las cuatro variables explicativas y un nodo en la salida para representar a la variable dependiente. Se usó la función de activación sigmoide en la capa oculta y la identidad en la salida. Las ecuaciones estimadas para los modelos de regresión lineal múltiple y de red neuronal son las siguientes:

$$\hat{y}_{RL} = 62,405 + 1,551x_1 + 0,51x_2 + 0,102x_3 - 0,144x_4$$

$$\hat{y}_{RN} = 69,61 + 46,9 \left( \frac{1}{1 + \exp(-(-0,227 + 2,028x_1 + 2,951x_2 + 1,142x_3 + 1,964x_4))} \right)$$

Para la regresión lineal multivariada, se aplicó un perceptron multicapa con cuatro nodos en la capa de entrada (cuatro variables explicativas), dos nodos en la capa oculta y dos nodos en la salida (dos variables dependientes). Se usó como función de activación la sigmoide en la capa oculta y la identidad en la salida. Las ecuaciones estimadas obtenidas para cada variable dependiente en la regresión lineal y red neuronal son:

$$\hat{y}_{1\_RL} = 1,623 - 0,849x_1 + 0,011x_2 + 0,098x_3$$

$$\hat{y}_{2\_RL} = 7,660 + 10,532x_1 + 0,135x_2 + 0,847x_3$$

$$\hat{y}_{1\_RN} = 0,78 + 0,27 \left( \frac{1}{1 + \exp(-(-0,31 - 0,24x_1 - 0,47x_2 - 0,66x_3))} \right) + 1,19 \left( \frac{1}{1 + \exp(-(-1,92 - 1,53x_1 + 0,13x_2 + 0,29x_3))} \right)$$

$$\hat{y}_{2\_RN} = 14,24 - 0,14 \left( \frac{1}{1 + \exp(-(-0,31 - 0,24x_1 - 0,47x_2 - 0,66x_3))} \right) + 14,98 \left( \frac{1}{1 + \exp(-(-1,92 - 1,53x_1 + 0,13x_2 + 0,29x_3))} \right)$$

Los resultados comparativos se muestran en la Tabla 1, se refieren a la suma de cuadrados residual y los coeficientes de determinación obtenidos para la regresión lineal y redes neuronales. Como se observa, para el caso de la regresión lineal simple las redes neuronales obtuvieron un valor de  $R^2$  de 96,8 %, siendo mucho mayor al de la regresión lineal de 87,5 %; para la regresión múltiple ambas técnicas resultaron con un valor similar de  $R^2$ . En la regresión multivariada, para la primera variable dependiente la regresión lineal tuvo un  $R^2$  (54,0 %) mayor y en la segunda variable el mayor  $R^2$  lo obtuvo el modelo red neuronal (87,0 %).

**Tabla 1.** Comparación del análisis de regresión lineal y con redes neuronales

Modelo de regresión		Regresión lineal		Redes neuronales	
		S.C. Residual	$R^2$ %	S.C. Residual	$R^2$ %
Lineal simple		1,279	87,5	0,323	96,8
Lineal Múltiple		47,86	98,2	59,51	97,8
Lineal Multivariado	Y1	1,59	54,0	2,69	22,0
	Y2	63,14	66,5	24,41	87,0

En la Tabla 2 se presenta la distribución de los  $R^2$  de los 500 conjuntos de datos simulados ajustados a una regresión lineal simple y una red neuronal perceptron simple. El 47,2 % del total de los conjuntos simulados tuvieron un  $R^2$  entre 50 % y 75 % para la regresión lineal simple, mientras que 34,4 % de los conjuntos de datos simulados ajustados a una red neuronal correspondieron a este rango.

**Tabla 2.** Distribución de los  $R^2$  de los 500 conjuntos de datos simulados

Rangos de $R^2$	Regresión lineal simple		Red neuronal	
	Número	%	Número	%
] 0 - 25 ]	57	11,4	68	13,6
] 25 - 50 ]	132	26,4	117	23,4
] 50 - 75 ]	236	47,2	172	34,4
] 75 - 100 ]	75	15,0	143	28,6
	500	100,0	500	100,0

En la Tabla 3 se muestran los resultados comparativos de los  $R^2$  para la regresión lineal simple y redes neuronales, resultando que la cantidad de casos con mayores  $R^2$  fue con redes neuronales, mientras que en regresión lineal fue el segundo mayor, en cambio el caso de iguales  $R^2$  fue el más escaso. Se observa que del 61,6 % de los conjuntos simulados resultaron las redes neuronales con mayor  $R^2$ , el 36,6 % fueron mayores los  $R^2$  para el ajuste de una regresión lineal simple y en el 2,2 % ambas técnicas tuvieron similar  $R^2$ .

**Tabla 3.** Comparación de los  $R^2$  para la regresión lineal simple y red neuronal

Resultado comparativo	Número	%
Regresión lineal simple con mayores $R^2$	183	36,6
Iguales $R^2$	11	2,2
Red neuronal con mayores $R^2$	306	61,2
	500	100

#### 4. Conclusiones y recomendaciones

La metodología mostró aspectos que deben ser considerados para ajustar conjuntos de datos a los modelos de regresión lineal simple, múltiple y multivariada usando el modelo de red neuronal perceptron multicapa.

Los resultados comparativos, para el caso de la regresión lineal simple, mostraron valor de  $R^2$  de 96,8 % para las redes neuronales, lo cual es mucho mayor al de la regresión clásica (87,5 %). Para la regresión múltiple, ambas técnicas resultaron con valores similares de  $R^2$  (97,8 % y 98,2 %). En la regresión multivariada, para la primera variable dependiente, los valores de los  $R^2$  para la red neuronal y regresión lineal fueron de 22,0 % y 54,0 % y en la segunda variable dependiente, los valores fueron de 87,0 % y 66,5 %.

Los resultados con los 500 conjuntos de datos simulados ajustados a una regresión lineal simple, mostraron que en el 61,2 % de los conjuntos de datos, el modelo de red neuronal resultó con los mayores  $R^2$ .

El problema acerca del análisis del efecto de las variables de entrada (variables predictoras), conocido como el análisis de sensibilidad, en una red neuronal constituye una línea de investigación que se está desarrollando con gran interés.

Se recomienda realizar estudios comparativos para evaluar la bondad de ajuste de los modelos de redes neuronales y los modelos clásicos de regresión, considerando aspectos sobre los supuestos de los modelos estadísticos (normalidad, multicolinealidad, autocorrelaciones de errores, etc).

#### 5. Literatura citada

- Bishop C.M. 1995.** *Neural networks for pattern recognition*. Oxford: Oxford University Press.
- Bishop, C.M. 1994.** Neural networks and their applications. *Review of Scientific Instruments*, 65(6):1803-1832.
- Chattiesje, S. y Laudato, M. 1995.** *Statistical applications of neural networks*, 1, 1-21. Massachusetts: Northeastern University Boston.
- Cheng, B. y Titterington, D.M. 1994.** Neural networks: a review from a statistical perspective. *Statistical Science*, 9(1): 2-54.
- Garson, G.D. 1991.** A comparison of neural network and expert systems algorithms with common multivariate procedures for analysis of social science data. *Social Science Computer Review*, 9: 399-434.
- Haykin, S. 1999.** *Neuronal networks. A comprehensive foundation*. (2da. ed.). Prentice Hall.
- Huang, W.Y. y Lippmann, R.P. 1987.** Comparisons between neural nets and conventional classifiers. *Proceedings of the IEEE International Conference on Neural Networks*, I: 485-494.
- Masters, T. 1993.** *Practical neuronal networks recipes in C++*. London: Academia Press.
- Michie, D.; Spiegelhalter, D.J. y Taylor, C.C. 1994.** *Machine learning, neural and statistical classification*.

New York: Ellis Horwood.

**Navarro, J.B. 1998.** *Aplicación de redes neuronales artificiales al tratamiento de datos incompletos.* (Tesis doctoral no publicada). Universidad Autónoma de Barcelona.

**Pitarque, A.; Roy, J. y Ruiz, J. 2000.** Las redes neuronales como herramientas estadísticas no paramétricas de clasificación. *Psicológica* 12(2): 459-463.

**Pitarque, A.; Roy, J. y Ruiz, J. 1998.** Redes neuronales vs modelos estadísticos: Simulaciones sobre tareas de predicción y clasificación. *Psicológica* (19): 387-400.

**Ripley, B.D. 1994.** Neural networks and related methods for classification. *Journal of the Royal Statistical Society*, 56(3): 409-456.

**Sarle, Warren S. 1994.** *Neural Networks and Statistical Models.* Proceeding of the 19th annual SAS Users Group International Conference, pp. 1538-50. Cary, NC (SAS Institute).

**Sarle, W.S. 2000.** *How to measure importance of inputs?* Recuperado de <ftp://ftp.sas.com/pub/neural/importance.html>.

**Thrun, S.; Mitchell, T. y Cheng, J. 1991.** The MONK's comparison of learning algorithms: introduction and survey. En S. Thrun, J. Bala, E. Bloedorn e I. Bratko (Eds.), *The MONK's problem: a performance comparison of different learning algorithms*, pp. 1-6. Pittsburg: Carnegie Mellon University.

**Vicino, F. 1998.** Some reflections on artificial neural networks and statistics: two ways of obtaining solutions by working with data. *Substance Use & Misuse*, 3(2): 221-231.

**White, H. 1989.** Neural network learning and statistics. *AI Expert*, Dec. 48-52.

**White, H. 1994.** *Parametric statistical estimation with artificial neural networks: Mathematical perspectives neural networks*, pp. 1-20.