

# **COLEGIO DE POSTGRADUADOS**

---

INSTITUCION DE ESNEÑANZA E INVESTIGACION EN CIENCIAS AGRÍCOLAS

CAMPUS MONTECILLO

POSTGRADO DE SOCIOECONOMÍA, ESTADÍSTICA E INFORMÁTICA

CÓMPUTO APLICADO

## **LAS REDES NEURONALES ARTIFICIALES FEEDFORWARD COMO IDENTIFICADORAS DE ASOCIACIONES ESPURIAS ENTRE CAMINATAS ALEATORIAS**

JUAN PEDRO VÁSQUEZ LÓPEZ

T E S I S

PRESENTADA COMO REQUISITO PARCIAL  
PARA OBTENER EL GRADO DE:

MAESTRO EN CIENCIAS

MONTECILLO, TEXCOCO, EDO. DE MÉXICO

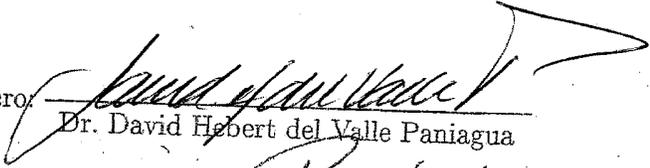
2008

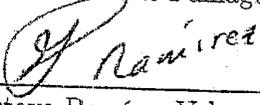
La presente tesis, titulada: Las redes neuronales artificiales feedforward como identificadoras de asociaciones espurias entre caminatas aleatorias, realizada por el alumno: Juan Pedro Vásquez López, bajo la dirección del Consejo Particular indicado, ha sido aprobada por el mismo y aceptada como requisito parcial para obtener el grado de:

MAESTRO EN CIENCIAS  
SOCIOECONOMÍA, ESTADÍSTICA E INFORMÁTICA

CÓMPUTO APLICADO

CONSEJO PARTICULAR

Consejero:   
Dr. David Hebert del Valle Paniagua

Director de tesis:   
Dr. Gustavo Ramirez Valverde

Asesor:   
Dr. Antonio Fernando Martínez Alcántara

Montecillo, Texcoco, México, marzo de 2008.

---

# LAS REDES NEURONALES FEEDFORWARD COMO IDENTIFICADORAS DE ASOCIACIONES ESPURIAS ENTRE CAMINATAS ALEATORIAS

Juan Pedro Vásquez López, M. en C.

Colegio de Postgraduados, 2008

Una regresión espuria es la presencia de una asociación entre variables independientes; y se presenta evidencia de ella cuando no se rechaza la hipótesis  $H_0 : \beta_1 = 0$  al hacer dicha regresión. Desde la década de 1970 se había demostrado que la regresión espuria es un fenómeno frecuente en caminatas aleatorias. Su identificación es de interés principalmente en la Econometría.

En este trabajo se propone el empleo de una red neuronal artificial feedforward multicapa como herramienta de detección de regresión espuria entre caminatas aleatorias, siendo la pendiente cero de la respuesta de la red y la aceptación de la hipótesis sobre la media poblacional  $H_0 : \delta_1 = 0$  en la respuesta de la red como variable dependiente de  $z_t$ , un indicio de la no regresión entre las variables.

Palabras clave: Redes neuronales artificiales, Regresiones espurias, Caminatas aleatorias, R, AMORE.

---

FEEDFORWARD ARTIFICIAL NEURAL NETWORKS AS RANDOM  
WALKS SPURIOUS REGRESSION IDENTIFIERS

Juan Pedro Vásquez López, M. en C.

Colegio de Postgraduados, 2008

A spurious regression is the association between independent variables; its evidence is found through rejecting  $H_0 : \beta_1 = 0$  hypothesis in such regression. Since the 1970 decade it was shown that it is common for random walks. Econometrics has a particular interest in identify them.

This work uses a multilayered feedforward artificial neural network as posible tool for detection of random walks spurious regression, having the zero slope of the network response and its acceptance of the hipotesis of the population mean  $H_0 : \delta_1 = 0$  about the network response as dependent of  $z_t$ , as an indicator of no-regression.

Key words: Artificial neural networks, spurious regression, random walk, R, AMO-RE.

---

## Agradecimientos

---

Al Eterno bendito sea, que siempre me permite decir *gam zu letova* y así declarar que aún las desgracias, a los que conforme a su propósito son llamados, todas las cosas son para bien. A mi esposa y mis padres por su incondicional apoyo.

A los contribuyentes que ha través del CONACyT y el Colegio de Postgraduados han financiado parte de mi formación. Todos los que integran mi consejo particular por su dedicación y paciencia.

---

## Índice general

---

<b>Resumen</b>	<b>III</b>
<b>Abstract</b>	<b>IV</b>
<b>Agradecimientos</b>	<b>v</b>
<b>1. Introducción</b>	<b>IX</b>
<b>2. Objetivos</b>	<b>1</b>
<b>3. Revisión de literatura</b>	<b>2</b>
3.1. La caminata aleatoria: una serie temporal notable . . . . .	2
3.2. Regresión lineal simple en caminatas aleatorias . . . . .	5
3.3. Las redes neuronales y el análisis de regresión . . . . .	7
<b>4. Metodología</b>	<b>10</b>
4.1. Hipótesis . . . . .	10
4.2. Experimentación . . . . .	11
4.3. Datos . . . . .	11
4.4. Predictores . . . . .	12

<b>5. Resultados y discusión</b>	<b>13</b>
5.1. Conjuntos de datos II y III: variables asociadas . . . . .	13
5.2. Variables independientes: conjunto de datos I . . . . .	15
5.3. Caminatas aleatorias . . . . .	16
<b>6. Conclusiones</b>	<b>18</b>
<b>Bibliografía</b>	<b>19</b>
<b>A. Programas en R</b>	<b>21</b>
<b>Índice alfabético</b>	<b>26</b>

---

## Índice de figuras

---

3.1. Caminata aleatoria pura. . . . .	4
3.2. Autocorrelograma de una caminata aleatoria pura. . . . .	6
3.3. Autocorrelograma de ruido blanco. . . . .	6
3.4. Diagrama de la red neuronal artificial multicapa feedforward utilizada en este trabajo. . . . .	7
5.1. Rechazos a las pruebas de hipótesis de los predictores ante los conjuntos de datos II y III. . . . .	14
5.2. Rechazos a las pruebas de hipótesis de los predictores ante el conjunto de datos I. . . . .	15
5.3. Rechazos a las pruebas de hipótesis de los predictores ante el conjunto de datos IV. . . . .	16
5.4. Instancia típica de comportamiento del RLS (línea recta) y de la RNA (triángulos) ante una asociación espuria entre caminatas aleatorias. . . . .	17

# CAPÍTULO 1

---

## Introducción

---

**Espurio, espuria** (Del lat. *spurius*) adj. bastardo, que degenera su origen o naturaleza. || 2. Hijo espurio || 3. Falso, adulterado, que degenera su origen (Real Academia Española, 1992).

La predicción es de suma importancia en muchas de las actividades humanas y principalmente en los procesos de toma de decisiones. Para poder hacer predicciones sobre algún fenómeno, una de las metodologías estadísticas más populares es la *regresión*. Éste es un modelo funcional que investiga la relación que existe entre una variable dependiente (o de respuesta) y varias independientes (o explicativas), con el objetivo de construir una ecuación de predicción del valor medio de la primera respecto de las segundas. Eventualmente, de ser posible, dichas predicciones pueden ocuparse en operaciones de control, relacionadas con el fenómeno de estudio (Bowerman et al., 2005; Gujarati, 2004; Greene, 1999).

Sin embargo, de antaño es sabido que pueden darse casos en los que se aplique un análisis de regresión a un par de variables, las cuales *a priori* se sabe que no están asociadas; y la prueba de hipótesis sobre los parámetros acepten que sí lo están (Yule, 1926).

Mejor conocida como *regresión espuria*, la información aportada por la regresión de este tipo de datos, no es fiable para tareas de descripción y predicción. A

lo largo de la literatura se citan muchos casos, algunos incluso humorísticos como el de la alta relación hallada entre la cantidad de lluvia acumulada y el nivel de precios al consumidor en un cierto periodo en Reino Unido, o la correlación de 0.95 entre la proporción entre los matrimonios efectuados en la Iglesia de Inglaterra y el total de matrimonios, contra la tasa de mortalidad del periodo 1866–1911 (Hendry, 1980).

Noriega reporta que la regresión espuria se da cuando ambas variables tienen una fuerte componente estocástica, como en las series temporales (Noriega y Ventosa-Santualària, 2006). Sin embargo, desde 1974, Granger y Newbold habían mostraron que este fenómeno es frecuente cuando ambas variables son un tipo de serie de tiempo llamado caminata aleatoria (Phillips, 1998).

En otro orden de ideas, recientemente los métodos computacionales de inteligencia artificial han ganado terreno en donde los métodos matemáticos y estadísticos tradicionales hallaban serias dificultades. Un ejemplo de ellos son las *redes neuronales artificiales*. Éstos algoritmos consisten en conjuntos de unidades de cómputo simples interconectados en intrincadas topologías, y su procesamiento masivo en paralelo emula el funcionamiento de una red neuronal biológica: aprendizaje de experiencias pasadas, veloz síntesis de grandes cantidades de información, resistencia a fallos localizados, etc. Tienen la capacidad de desarrollar relaciones funcionales entre datos y proveer una poderosa herramienta de interpolación no-lineal y multidimensional; por lo que se les ha empleado en principalmente en predicción e identificación de patrones. Sus aplicaciones se extienden a diversos campos de la actividad humana: Economía, Electrónica, Física, Medicina, por citar algunos (Haykin, 1994; Orr, 1999).

El presente trabajo se centró en estudiar la respuesta de una red neuronal artificial feedforward que modele regresiones espurias entre caminatas aleatorias; con el fin de proponerla como un medio de detección de éstas.

## CAPÍTULO 2

---

### Objetivos

---

**Objetivo General:** Evaluar la utilización de una red neuronal artificial multicapa feedforward como un método para detectar regresiones espurias entre caminatas aleatorias.

**Objetivo Particular:** Modelación de regresiones espurias entre caminatas aleatorias en redes neuronales artificiales multicapa feedforward.

# CAPÍTULO 3

---

## Revisión de literatura

---

Así entonces, el marco conceptual de este estudio queda delimitado por las caminatas aleatorias, el análisis de regresión lineal simple y su relación con las redes neuronales artificiales.

### 3.1. La caminata aleatoria: una serie temporal no-table

Una *serie temporal*, se define como una secuencia cronológica de observaciones de una variable en particular.

El estudio de éstas resulta importante en las áreas de Econometría y Finanzas, Física, Mercadotecnia, Demografía, Control de procesos y, desde luego, la Estadística y la Minería de Datos, entre muchas otras .

Para motivos de análisis, una serie temporal, se puede describir como una variable  $y_t$  que está dada en términos de factores contemporáneos  $x_t$  (y/o retardados), perturbaciones  $\epsilon_t$  y una o varias instancias de su pasado  $y_{t-1}$  (Ferson et al., 2003):

$$y_t = \beta_1 + \beta_2 x_t + \beta_3 y_{t-1} + \beta_4 y_{t-2} + \dots + \epsilon_t$$

Donde el *término de innovación* o perturbación ( $\epsilon_t$ ), cumple con los siguientes requisitos:

$$\begin{aligned} E(\epsilon_t) &= 0 \\ \text{var}(\epsilon_t) &= \sigma_\epsilon^2 \\ \text{cov}(\epsilon_s, \epsilon_t) &= 0, \forall t \neq s \end{aligned}$$

Por lo que comúnmente se designa a  $\epsilon_t$  como *ruido blanco* (Greene, 1999; Gujarati, 2004).

Chatfield (1996) describe el caso notable de serie temporal, la *caminata aleatoria* pura; misma que se define mediante la expresión :

$$x_t = x_{t-1} + \epsilon_t \tag{3.1}$$

Donde  $\epsilon_t$  es ruido blanco. Éste es el modelo sobre el cual se basa la investigación de nuestro trabajo, a pesar de su simplicidad tiene características interesantes: directamente de la ecuación 3.1 se deduce que el mejor predictor para valores futuros es el valor actual de la caminata, es decir, el siguiente valor de  $x_t$  está determinado por el  $x_{t-1}$  actual.

Cuando el proceso se inicia,  $t = 0$ , la expresiones sucesivas quedan como:

$$\begin{aligned} x_1 &= x_0 + \epsilon_1 \\ x_2 &= x_1 + \epsilon_2 = x_0 + \epsilon_1 + \epsilon_2 \\ &\vdots \\ x_t &= x_0 + \sum_{i=1}^t \epsilon_i \end{aligned} \tag{3.2}$$

De la ecuación 3.2, se puede observar que las caminatas aleatorias tienen *memoria infinita* pues los errores aleatorios persisten: suponiendo que el  $j$ -ésimo error fuese  $\epsilon_j = c$  y el resto  $\epsilon_k = 0$  para toda  $k \neq j$ , todas las  $x_t$  con  $t > j$  serían  $c$  unidades más grandes, por lo que el efecto de este error no se desvanece con el paso del tiempo.

De esa misma ecuación se tiene que los valores esperados quedan de la siguiente forma:

$$E(x_t) = E\left(x_0 + \sum_{i=1}^t \epsilon_i\right) = x_0 + t \cdot 0 = x_0 \quad (3.3)$$

$$\text{var}(x_t) = \text{var}\left(x_0 + \sum_{i=1}^t \epsilon_i\right) = 0 + t\sigma_\epsilon^2 = t\sigma_\epsilon^2 \quad (3.4)$$

Lo que significa que, si bien la media de  $x_t$  permanece constante (ec. 3.3), conforme pasa el tiempo la varianza crece de manera indefinida (3.4). Es decir, que conforme pase el tiempo, la probabilidad de el valor futuro remoto que tome la variable sea similar al valor actual es cada vez menor (Murray, 1994).

Sin embargo, al hacer las primeras diferencias de la ecuación 3.1:

$$\Delta x_t = x_t - x_{t-1} = \epsilon_t \quad (3.5)$$

se forma un proceso puramente aleatorio, de media y varianza constantes (Gujarati, 2004).

Uno de los ejemplos más conocidos de variables que se comporten como caminatas aleatorias, es el precio de las acciones bursátiles en días sucesivos (Chatfield, 1996).

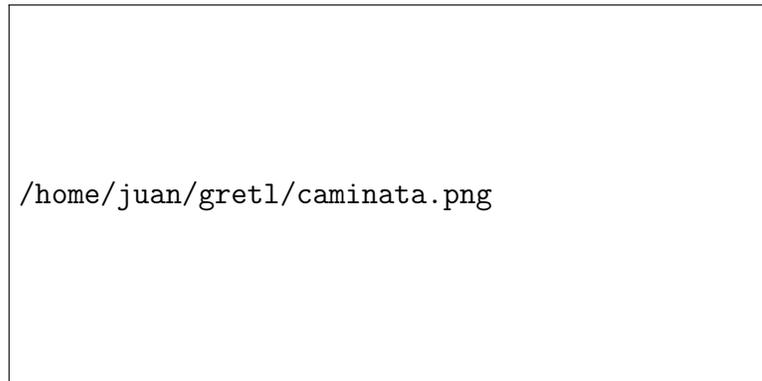


Figura 3.1: Caminata aleatoria pura.

## 3.2. Regresión lineal simple en caminatas aleatorias

Teniendo entonces que la forma funcional del modelo de *regresión lineal simple (RLS)* es:

$$z_t = \beta_0 + \beta_1 x_t + \gamma_t, \quad \gamma_t \sim N(0, \sigma_\gamma^2) \quad (3.6)$$

donde  $z_t$  es la variable dependiente y  $x_t$  es la variable explicativa; mientras que  $\gamma_t$  es el término de error estocástico, que se supone que tiene una distribución normal, media igual a cero y varianza constante (Greene, 1999).

La validez del análisis de regresión requiere el cumplimiento de los supuestos sobre los cuales se fundamenta. Uno de los cuales se puede enunciar exigiendo que  $\forall i \neq j$ , las instancias del término de perturbación de la variable explicativa  $x_t$  cumplen con  $\epsilon_i$  sea estadísticamente independiente de  $\epsilon_j$ . Pero, siendo  $x_t$  una caminata aleatoria, como se comentó anteriormente respecto a la ecuación 3.2, dichas instancias no son independientes, lo que viola el principio (Gujarati, 2004).

Dicho de otra manera, los instancias de la caminata aleatoria presentan *autocorrelación*. Constituyendo una primera causa de regresiones espurias en caminatas aleatorias; si bien en la literatura se tienen procedimientos para comprobar la autocorrelación de variables, como la prueba de Durbin–Watson y técnicas remediales como aplicar ciertas transformaciones a la variable explicativa (Bowerman et al., 2005). Sin embargo, existe una inspección gráfica preliminar que puede orientar si una serie de datos está autocorrelacionada, se trata del *autocorrelograma*; que consiste en una gráfica del *coeficiente de autocorrelación* a través del tiempo (instancias), dicho en términos informales, éste coeficiente expresa qué tanto explica la primera instancia de la variable el resto de las instancias, y está definido como:

$$r_k = \frac{\sum_{t=1}^{n-k} (x_t - \bar{x})(x_{t+k} - \bar{x})}{\sum_{t=1}^n (x_t - \bar{x})^2}$$

Y es de esperarse que ante series autocorrelacionadas dicho coeficiente decrezca menos rápido que en una serie con instancias independientes, como muestran las figuras 3.2 y 3.3.



/home/juan/gretl/correlogramacaminata.png

Figura 3.2: Autocorrelograma de una caminata aleatoria pura.



/home/juan/gretl/correlogramaruidoblanco.png

Figura 3.3: Autocorrelograma de ruido blanco.

Por otra parte, en el modelo de regresión, si ambas caminatas aleatorias presentan una tendencia y aún ambas crecieren o decrecieren aproximadamente a la misma tasa, la RLS hallaría una asociación considerable. De cumplirse esas condiciones, se diría que las variables  $x_t$  y  $z_t$  están *cointegradas*. Y esta sería una segunda

razón de las asociaciones espurias en caminatas aleatorias; por lo que se aplican metodologías de análisis de cointegración, para proponer otros modelos funcionales diferentes al de la ec. 3.6, que relacionen ambas variables (Greene, 1999).

### 3.3. Las redes neuronales y el análisis de regresión

Si bien se han buscado en la inteligencia artificial nuevos enfoques metodológicos para lidiar con los problemas de predicción e identificación de patrones, Warner y Misra muestran que hay bastante similitud en los fundamentos sobre los que operan el análisis de regresión y las *redes neuronales artificiales multicapa feedforward (RNAf)*.

Una red neuronal artificial multicapa se caracteriza por ser un conjunto de neuronas, que reciben información multivariable, la procesan y dan una respuesta; en la arquitectura *feedforward*, el arreglo de neuronas y sus interconexiones hace que la información circule en una sola dirección entrada-salida y nunca pueda pasar más de una vez a través de una neurona antes de generarse la respuesta de salida; tal como se muestra en la figura 3.4.

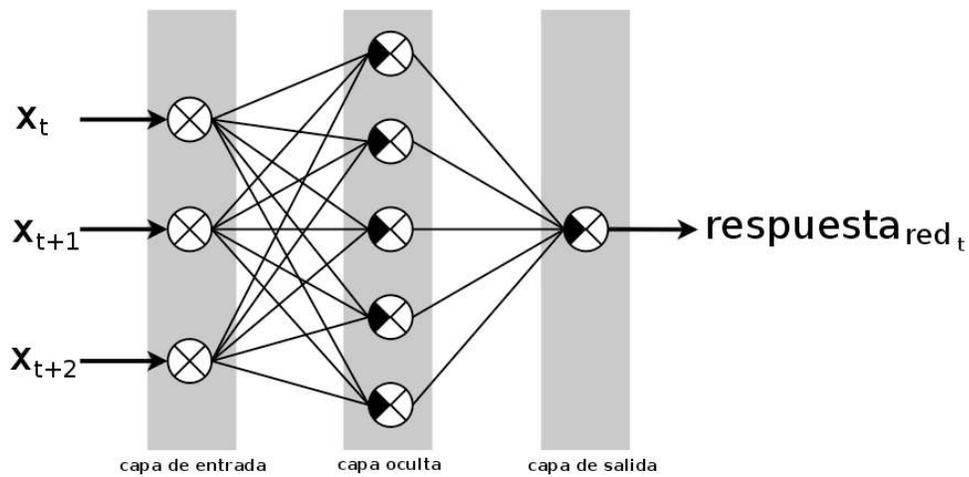


Figura 3.4: Diagrama de la red neuronal artificial multicapa feedforward utilizada en este trabajo.

La capa de entrada consta de neuronas que no modifican los valores que leen y se encargan de pasar la información a la siguiente capa de neuronas. Las capas ocultas o de procesamiento son las neuronas que reciben la información de la capa de entrada, la ponderan mediante *pesos* asignados a cada interconexión de neuronas y mediante una *función de transferencia* discriminan la información procesada que pasará a la siguiente capa. Y las neuronas de la capa de salida sintetizan la información procesada, construyendo la variable o variables con los datos de salida.

Para operar la RNAf mediante el algoritmo de aprendizaje *gradiente descendiente adaptativo* (*ADAPTgd*) se debe tener una fase previa de *entrenamiento de la red*, para la cual se debe disponer de una fracción de los datos de entrada y los correspondientes de datos conocidos de salida para *validación*; y mediante sucesivas corridas, la red es alimentada con dichos datos de entrada y el algoritmo de aprendizaje modifica los pesos de las interconexiones hasta que se llegue al mínimo de variaciones de dichos pesos que es cuando, idealmente, la respuesta de la red es muy aproximada a los datos de validación. Una vez entrenada la red, se le dan el resto de datos de entrada y se obtiene la respuesta de la red propiamente dicha.

En su artículo Warner y Misra (1996) reconocen que los pesos de la RNAf se asemejan a los coeficientes de la regresión y el algoritmo de aprendizaje ADAPTgd está basado en minimizar el error cuadrado medio, al igual que los mínimos cuadrados ordinarios en RLS. Pero la ventaja principal de la RNAf radica en que no exige supuesto alguno sobre la naturaleza de los datos con que se le alimenta, en oposición al RLS que si los tiene; permitiéndole a la red aproximar cualquier función continua sin necesidad de conocer la forma funcional específica –ecuación de un modelo. Sin embargo, esta clara ventaja también representa la dificultad de encontrar una interpretación de la red (Warner y Misra, 1996).

Por otra parte, señalan también que la convergencia de la red depende de las condiciones iniciales que se le den: datos de entrada, valores iniciales de los pesos, función de transferencia elegida, topología y número de neuronas en cada capa, etc. Sin embargo, se dispone de ciertas reglas empíricas para la implementación de una RNAf adecuada para la mayoría de problemas, como por ejemplo: no se requieren más de 10 neuronas en la capa de entrada; es bastante raro necesitar

de más de 3 capas ocultas; cuando la capa oculta tiene aproximadamente el doble de neuronas que la capa de entrada, la respuesta de la red tiene buena aproximación y converge rápidamente; generalmente se utiliza la función de transferencia *sigmoideal tangente hiperbólica* (*tansig*), misma que se define como

$$\textit{tansig}(n) = \frac{2}{(1 + e^{-2n})^{-1}}$$

y mapea todo  $\mathbb{R}$  al intervalo  $[-1, 1]$  (Turban et al., 2005).

# CAPÍTULO 4

---

## Metodología

---

Dadas dos variables estadísticamente independientes  $x_t$  y  $z_t$ , resulta bastante claro cuándo un análisis de RLS presenta una asociación espuria. Sin embargo respecto a las redes neuronales artificiales, no se encuentra en la literatura una condición que de cumplirse, indique si la RNAf ha detectado una regresión espuria.

Como se explicó en la sección 3.3 en la página 8, las RNAf son básicamente *cajas negras* que no exigen requisito alguno sobre los datos de entrada; sólo se toma una fracción de los pares ordenados  $(x_t, z_t)$  para entrenamiento y el resto para validar la capacidad predictiva, se espera que las predicciones sean semejantes a las  $z_t$  correspondientes.

Se propone entonces que la asociación entre  $z_t$  y su predicción sea un indicador de que la RNAf ha detectado una asociación lineal.

### 4.1. Hipótesis

Dadas dos variables estadísticamente independientes  $x_t$  y  $z_t$ , una RNAf que modele la segunda a partir de la primera y considerando un análisis de regresión

lineal simple bajo el modelo:

$$respuesta_{red} = \delta_0 + \delta_1 z_t + \eta_t$$

donde  $respuesta_{red}$  son los valores que aporta la RNaf (previamente entrenada) a los  $x_t$  de entrada, los  $z_t$  correspondientes, son la variable explicativa de dicha respuesta y  $\eta_t$  es ruido blanco. Haciendo la prueba de hipótesis sobre la media poblacional  $H0 : \delta_1 = 0$  contra  $Ha : \delta_1 \neq 0$ . Rechazándose  $H0$  a favor de  $Ha$  con una probabilidad de error tipo I del  $\alpha = 5\%$ , si y sólo si  $pvalue_{red} < \alpha$ .

Se tiene que, si  $H0$  se acepta, entonces la RNaf ha detectado una regresión espuria.

## 4.2. Experimentación

Demostrar teóricamente la validez del criterio de detección para RNaf de asociaciones espurias es una labor complicada. Por lo que se empleó la experimentación *in silico* a través del software estadístico R y su componente para RNaf llamado AMORE (R Development Core Team, 2006; Limas et al., 2007). El código desarrollado es el programa A.2 en la página 23 .

## 4.3. Datos

Se consideraron cuatro conjuntos de datos de entrada. Cada uno de ellos formados por dos variables:  $x_t$  y  $z_t$  que son vectores de 100 componentes  $t = \{1, 2, \dots, 100\}$ , como se especifica en el cuadro 4.1.

Conjunto	$x_t$	$z_t$
I	$x_t \sim N(1, 2)$	$z_t \sim N(0, 1)$
II	$x_t \sim N(0, 10)$	$z_t = 7x + N(0, 145)$
III	$x_t = x_{t-1} + N(0, 1)$	$z_t = 7x + N(25, 30)$
IV	$x_t = x_{t-1} + N(10, 1)$	$z_t = z_{t-1} + N(0, 1)$

Cuadro 4.1: Variables y naturaleza de los datos de entrada empleados en los experimentos.

Respecto a la relación entre las variables, es claro que en los conjuntos II y III  $z_t$  es función de  $x_t$ , es decir, están asociados; mientras que ambas son independientes en los conjuntos I y IV. Pues la intención es ver cómo se comporta la RNAf ante variables estadísticamente dependientes e independientes; ya que la hipótesis de la sección 4.1 implica que la respuesta de la red neuronal ante caminatas aleatorias, será similar a la que presente ante variables independientes.

Para cada conjunto de datos se generaron  $10^3$  instancias a través de simulación. Y para cada instancia se aplicaron los dos predictores RLS y RNAf para modelar  $z_t$  a partir de  $x_t$ .

## 4.4. Predictores

De la RLS con  $x_t$  como la variable explicativa de  $z_t$ , es decir  $z_t \sim x_t$ , se tomó en consideración el  $pvalue_{xz}$  de la prueba de hipótesis  $H0 : \beta_1 = 0$  contra  $Ha : \beta_1 \neq 0$ . Teniéndose que se rechaza  $H0$  a favor de  $Ha$  con una probabilidad de error tipo I del  $\alpha = 5\%$ , si y sólo si  $pvalue_{xz} < \alpha$  (Bowerman et al., 2005).

La RNAf, por su parte, tiene la configuración mostrada en la figura 3.4, es decir: 3 neuronas de entrada, 6 en la única capa oculta y 1 neurona en la capa de salida. Fue entrenada en  $4 \times 10^3$  épocas con la primera mitad de los pares ordenados  $(x_t, z_t)$ , en cada uno de los casos. Cabe aclarar que nuestro objetivo en este trabajo no es la predicción sino la detección, por lo que se consideró emplear el 50% de dichos pares ordenados para la fase de entrenamiento, a diferencia de las implementaciones de RNAf para predicción, que suelen utilizar porcentajes más altos para entrenamiento (Turban et al., 2005; Orr, 1999).

De la segunda mitad se tomaron las  $x_t$  como datos de entrada, almacenando en un vector de 50 componentes llamado  $respuesta_{red}$  los valores de salida de la red.

# CAPÍTULO 5

---

## Resultados y discusión

---

La contabilidad de los rechazos a las hipótesis nulas de las dos pruebas de hipótesis sobre la media poblacional ya mencionadas: una  $z_t \sim x_t$  para la RLS y otra  $respuesta_{red} \sim z_t$  para la RNAf, en los conjuntos de datos del cuadro 4.1 en la página 11, se pueden sintetizar utilizando los histogramas de frecuencias absolutas.

### 5.1. Conjuntos de datos II y III: variables asociadas

Ambos predictores se comportaron, respecto a las pruebas de hipótesis, de manera muy similar ante los datos de los conjuntos II y III, en los cuales  $z_t$  estaba asociada con  $x_t$ . Como se muestra en la figura 5.1.

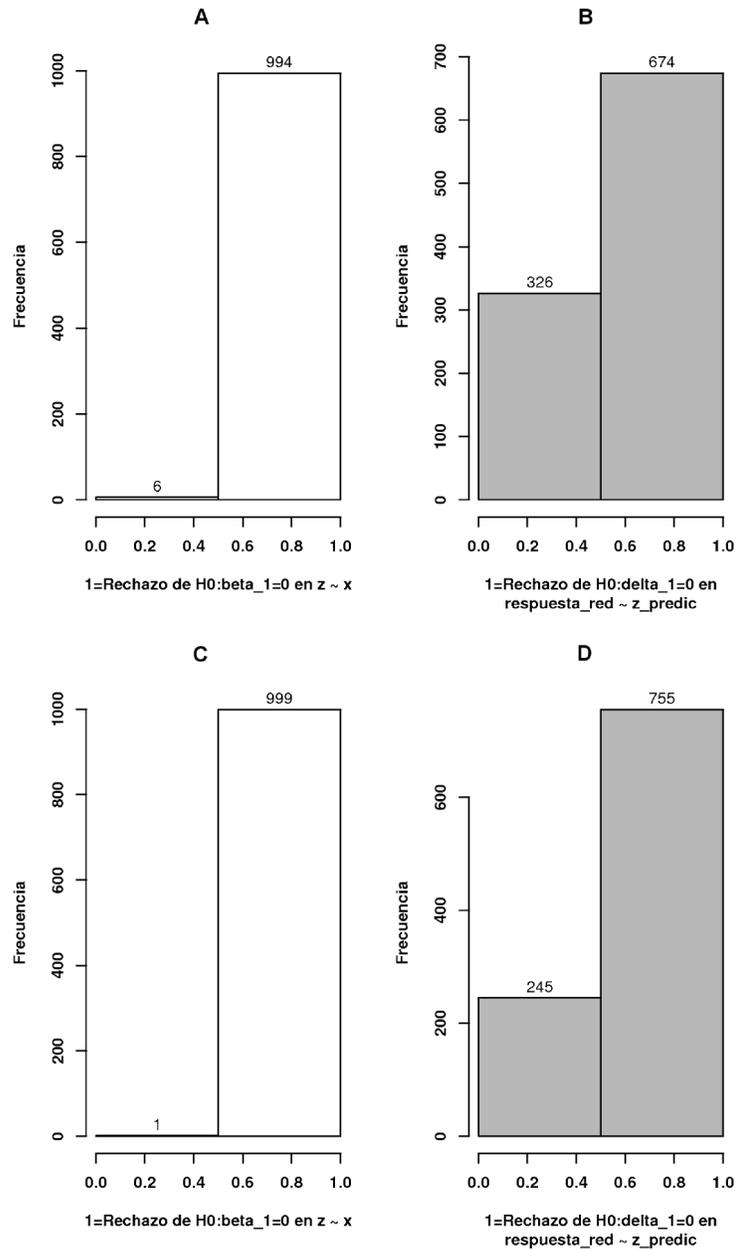


Figura 5.1: Rechazos a las pruebas de hipótesis de los predictores ante los conjuntos de datos II y III.

Donde se aprecia que la RLS (histogramas A y C) rechaza en altos porcentajes la hipótesis de pendiente igual a cero, lo que era de esperarse dada la naturaleza

de los datos. La otra prueba de hipótesis, la hecha sobre la respuesta de la RNAf (histogramas B y D), se comporta de manera similar rechazando en casi el 70 % de los casos la hipótesis de pendiente igual a cero en el modelo de la respuesta de red asociada con la  $z_t$ . Por lo que se puede decir que ambas pruebas evidencian una asociación entre las variables.

## 5.2. Variables independientes: conjunto de datos I

De manera consistente, también los dos predictores, respecto a las pruebas de hipótesis, se comportan de manera muy similar ante el conjunto de datos I, que lo conforman dos variables de distribución normal, es decir, estadísticamente independientes. Los histogramas correspondientes se muestran en la figura 5.2.

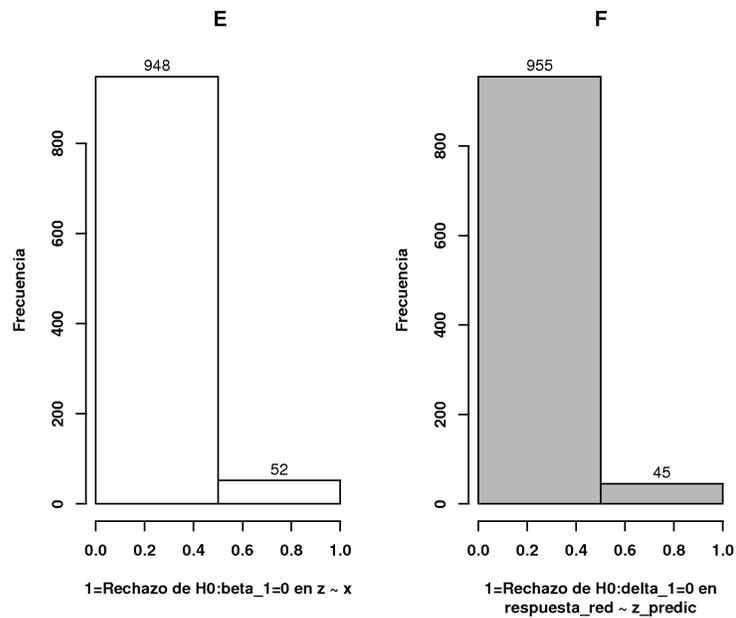


Figura 5.2: Rechazos a las pruebas de hipótesis de los predictores ante el conjunto de datos I.

La prueba de hipótesis correspondiente a la RNAf (histograma F) acepta en más del 95 % de los casos la hipótesis de pendiente cero entre la respuesta de la

red y la variable  $z_t$ , tal como lo hace también la RLS (histograma E) a pesar de darse un pequeño margen de cercano al 5% de regresiones espurias. Indicando que ambos predictores evidencian la ausencia de asociación entre las variables.

### 5.3. Caminatas aleatorias

Pero cuando se trata de caminatas aleatorias independientes, el conjunto de datos IV, los comportamientos en cuanto a las pruebas de hipótesis entre la RLS y la RNAf son bastante dispares, como se muestra en la figura 5.3.

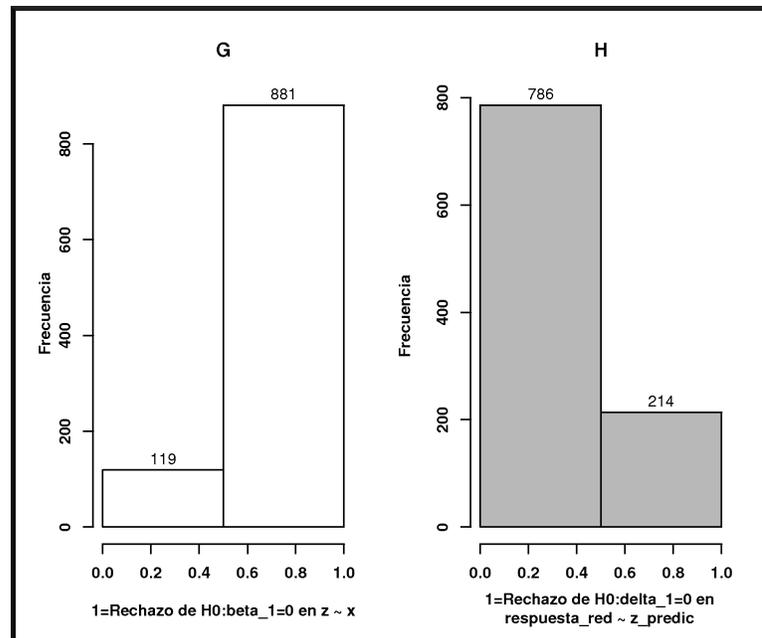


Figura 5.3: Rechazos a las pruebas de hipótesis de los predictores ante el conjunto de datos IV.

Mientras la prueba de hipótesis correspondiente a la RLS (histograma G) presenta un 88% de regresiones espurias, la prueba de hipótesis correspondiente a la respuesta de la RNAf (histograma H) acepta en 79% de los casos la hipótesis de la pendiente cero entre  $respuesta_{red}$  como variable dependiente de  $z_t$ ; un

comportamiento de la red neuronal similar al que presentó ante variables estadísticamente independientes (histograma F). Por lo que una instancia típica de este comportamiento desigual entre los dos predictores ante una regresión espuria se muestra en la figura 5.4.

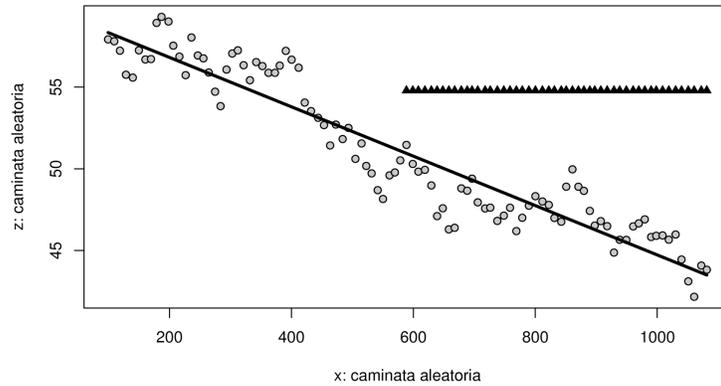


Figura 5.4: Instancia típica de comportamiento del RLS (línea recta) y de la RNA (triángulos) ante una asociación espuria entre caminatas aleatorias.

Indicando que, si bien la RLS se comporta como si las caminatas aleatorias estuviesen asociadas y en la mayoría de los casos presenta una pendiente de regresión distinta de cero, la RNA se comporta como ante variables independientes, teniéndose la pendiente de la respuesta de la red igual a cero respecto de  $z_t$ .

## CAPÍTULO 6

---

### Conclusiones

---

De la información obtenida de los experimentos, es claro que hay evidencia empírica de que las redes neuronales, no tan sólo son una alternativa al análisis de regresión simple en cuanto a predicción se refiere, sino que pueden ser utilizadas para la detección de regresiones espurias. La pendiente cero de la respuesta de la red y la aceptación de la hipótesis sobre la media poblacional  $H_0 : \delta_1 = 0$  en la respuesta de la red como variable dependiente de  $z_t$ , se pueden tomar como un indicio de la no dependencia funcional de las variables  $x$  y  $z$ ; a pesar de que un análisis de regresión simple no rechace dicha hipótesis nula.

---

## Bibliografía

---

- Bowerman, B. L., O’Connell, R. T. y Koehler, A. B. (2005), *Forecasting, time series and regression: an applied approach*, 4ª edn, Thomson Brooks Cole, EU.
- Chatfield, C. (1996), *The analysis of time series: an introduction*, 5ª edn, Chapman & Hall CRC, EU.
- Ferson, W. E., Sarkissian, S. y Simin, T. T. (2003), ‘Spurious regressions in financial economics?’, *The Journal of Finance* **LVIII**(4), 1393–1413.
- Greene, W. (1999), *Análisis econométrico*, 3ª edn, Pearson Educación, Madrid.
- Gujarati, D. (2004), *Econometría*, 4ª edn, McGraw-Hill Interamericana.
- Haykin, S. (1994), *Neural networks. A comprehensive foundation.*, Macmillan College Publishing Company, New York.
- Hendry, D. F. (1980), ‘Econometrics - alchemy or science?’, *Economica, New Series* **47**, 387–406.
- Limas, M. C., Joaquín B. Ordieres Meré, Eliseo P. Vergara González, de Pisón Ascacibar, F. J. M., Espinoza, A. V. P. y Elías, F. A. (2007), *AMORE: A MORE flexible neural network package*. R package version 0.2-11.  
**URL:** <http://wiki.r-project.org/rwiki/doku.php?id=packages:cran:amore>

- Murray, M. P. (1994), ‘A drunk and her dog: an illustration of cointegration and error correction’, *The American Statistician* **48**(1), 37–39.
- Noriega, A. E. y Ventosa-Santualària, D. (2006), ‘Spurious regression and econometric trends’, *Dirección de Investigación Económica, Banco de México*.
- Orr, G. (1999), ‘Cs-449: Neural networks (lecture notes)’. Última visita en abril de 2007.  
**URL:** <http://www.willamette.edu/~gorr/classes/cs449/intro.html>
- Phillips, P. C. B. (1998), ‘New tools for understanding spurious regressions’, *Econometrica* **66**(6), 1299–1325.
- R Development Core Team (2006), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.  
**URL:** <http://www.R-project.org/>
- Real Academia Española (1992), *Diccionario de la lengua española*, 21<sup>a</sup> edn, Espasa-Calpe, Madrid, España.
- Turban, E., Aronson, J. E. y Liang, T.-P. (2005), *Decision support systems and intelligent systems*, 7<sup>a</sup> edn, Pearson Prentice Hall, EU.
- Warner, B. y Misra, M. (1996), ‘Understanding neural networks as statistical tools’, *The American Statistician* **50**(4), 284–293.
- Yule, G. U. (1926), ‘Why do we sometimes get nonsense-correlations between time-series? – a study in sampling and the nature of time-series’, *Journal of the Royal Statistical Society* **89**, 1–63.

# APÉNDICE A

---

## Programas en R

---

Programa A.1: Porcentaje de regresiones espurias entre caminatas aleatorias.

```
1  #MEDIA Y DESVIACIÓN ESTÁNDAR DEL RUIDO NORMAL
2  media = 0
3  desvest = 5
4
5  #NUMERO DE EXPERIMENTOS
6  numexp=1000
7
8  #CONTADOR DE regresiones espurias
9  counter = 0
10
11 #LLAMA A UNA LIBRERÍA ESPECIAL
12 library(gmodels)
13
14 #CICLO DE EXPERIMENTOS
15 for (j in 1:numexp){
16
17     #DEFINE E INICIALIZA LOS VECTORES A CERO
18     #EN TODAS SUS COMPONENTES
19     x <- rep(0, times=100)
20     y <- rep(0, times=100)
```

---

```

21
22      #LA PRIMER COMPONENTE SON ELEMENTOS ALEATORIOS
23      #UNIFORMES ENTRE 0 Y 100
24      x[1] <- runif(1, 0, 100)
25      y[1] <- runif(1, 0, 100)
26
27      #EN CADA SIGUIENTE COMPONENTE SE ESTABLECE EL
28      #TÉRMINO ANTERIOR MÁS UN NÚMERO ALEATORIO NORMAL
29      for (i in 2:100){
30          x[i] <- x[i-1] + rnorm(1, media, desvest)
31          y[i] <- y[i-1] + rnorm(1, media, desvest)
32      }
33
34      #HACE LA REGRESIÓN LINEAL SIMPLE
35      reg <- lm(y~x)
36
37      #Y OBTIENE EL P-VALUE DE LA PRUEBA H0
38      pvalue <- ci(reg)[2,5]
39
40      #SI SE RECHAZA LA HIPÓTESIS ENTONCES TENEMOS
41      #UNA ASOCIACIÓN ESPÚRIA
42      if (pvalue < 0.05){
43          counter = counter + 1
44      }
45
46 }
47 #TABLA QUE PRESENTA LOS RESULTADOS
48 tabla <- matrix(0, nrow=2, ncol=1, dimnames = list(
49 c("Num. Experimentos:", "%de asocs. espurias"), c("RESULTADOS")
50 ))
51 tabla[1,1] <- numexp
52 tabla[2,1] <- counter *100/numexp
53 show(tabla)

```

---

	RESULTADOS
Num. Experimentos:	1000.0
% de asocs. espurias:	77.8

Cuadro A.1: Salida del programa A.1 en la consola R.

---

Programa A.2: Código base para los experimentos con los 4 conjuntos de datos descritos en el capítulo 4 en la página 10.

---

```
1  ##PAQUETES QUE SE NECESITARAN
2  require(graphics) ##HISTOGRAMAS
3  require(AMORE) ##REDES NEURONALES ARTIFICIALES
4
5  ##CONSTANTES
6  num = 1000 ##NUMERO DE INSTANCIAS
7  largo=100 ##LARGO DE X Y Z
8  corte=50 ##CORTE DE PARES (X,Z) PARA ENTRENAMIENTO
9  epocas= 4e3 ##EPOCAS DE ENTRENAMIENTO DE LA RED
10 tasa= 1e-4 ##TASA DE APRENDIZAJE
11
12 ##CONSTRUCCION DE VARIABLES
13 x <- rep(0,times=largo) ##VECTOR DE 100 COMPONENTES
14 z <- rep(0,times=largo) ##VECTOR DE 100 COMPONENTES
15
16 ##VECTOR DE RECHAZOS SOBRE HO:BETA_1=0 EN Z~X
17 rechazo <- rep(9,times=num)
18
19 ##VECTOR DE RECHAZOS SOBRE HO:DELTA_1=0
20 ##EN RESPUESTA_RED Z
21 rechazored <- rep(9,times=num)
22
23 ##IMPLEMENTACION DE LA RED NEURONAL
24 ##TOPOLOGIA,TASA DE APRENDIZAJE, CRIT. DE OPTIMIZAC. ,
25 ##FUNCION DE TRANSFERENCIA Y FUNC. TRANS. SALIDA,
26 ##ALGORITMO DE APRENDIZAJE
27 redneuronal <- newff(n.neurons=c(3,6,1),
28     learning.rate.global=tasa,
29     error.criterium="LMS",
30     hidden.layer="tansig",
31     output.layer="purelin",
32     method="ADAPTgd")
33
34 #####
35 ## CICLO PRINCIPAL DE EXPERIMENTOS ##
36 #####
37 for (i in 1:num){
```

---

```

38
39 # ##      I) X_NORMAL, Z_NORMAL
40 #          x <- rnorm(largo ,1 ,2)
41 #          z <- rnorm(largo ,0 ,1)
42
43 ##        II) X_NORMAL, Z_FUNCLIN
44          x <- rnorm(largo ,0 ,10)
45          z=7*x+rnorm(largo ,0 ,145)
46
47
48 # ##      III) X_CAMALEAT, Z_FUNCLIN
49 #          x[1] <- runif(1,0,5)
50 #          for(t in 2:100){
51 #              x[t] <- x[t-1] + rnorm(1,0,1)}
52 #          z=7*x+rnorm(largo ,25 ,29)
53
54 # ##      IV) X_CAMALEAT, Z_CAMALEAT
55 #          x[1] <- runif(1,0,100)
56 #          z[1] <- runif(1,0,100)
57 #          for(t in 2:100){
58 #              x[t] <- x[t-1] + rnorm(1,10,1)
59 #              z[t] <- z[t-1] + rnorm(1,0,1)
60 #          }
61
62          ##ANALISIS DE REGRESION LINEAL SIMPLE (Z~X)
63          reg <- lm(z~x) #RLS Z~X
64          pvaluexz <- summary(reg)$coefficients[2 ,4]
65
66          ##BANDERAS DE RECHAZO SOBRE HO:BETA_1=0
67          if(pvaluexz <0.05){
68              rechazo[i] <- 1}
69          else{
70              rechazo[i] <- 0}
71
72          ##SECCIONA LOS DATOS PARA ENTRENAMIENTO
73          x_entrena <- rep(0, times=corte)
74          z_entrena <- rep(0, times=corte)
75          for(t in 1:corte){
76              x_entrena[t] <- x[t]
77              z_entrena[t] <- z[t]}

```

---

---

```

78
79     ##SECCIONA LOS DATOS PARA PREDICCION
80     x_predic <- rep(0, times=largo-corte)
81     z_predic <- rep(0, times=largo-corte)
82     for(t in (corte+1):largo){
83         x_predic[t-corte] <- x[t]
84         z_predic[t-corte] <- z[t]}
85
86     ##ENTRENAMIENTO DE LA RNAf
87     ##RED, ENTRADA, OBJETIVO, CRITERIO DE OPTIMIZAC.,
88     ##NO REPORTAR PESOS, EPOCAS DE ENTRENAM.
89     redentrenada <- train(redneuronal,
90         x_entrena,
91         z_entrena,
92         error.criterium="LMS",
93         report=FALSE,
94         show.step=epocas,
95         n.shows=1)
96
97     ##OPERACION DE LA RNAf
98     ##HACE LA PREDICCION RESPUESTA_RED CON X_PREDIC
99     respuesta_red <- sim.MLPnet(redentrenada$net,
100    x_predic)
101
102     ##ANALISIS DE REGRESION RESPUESTA_RED ~ Z
103     regred<-lm(respuesta_red~z_predic)
104     pvaluered <- summary(regred)$coefficients[2,4]
105
106     ##BANDERAS DE RECHAZO SOBRE HO:DELTA_1=0
107     if(pvaluered < 0.05){
108         rechazored[i] <- 1}
109     else{
110         rechazored[i] <- 0}
111
112     ##ANUNCIA CADA 50 ITERACIONES
113     if(i % 50 == 0){
114         show(paste("ITERACION: ", i))}
115 }
116
117 ##EXPORTA GRAFICO A ARCHIVO PNG

```

---

---

```

118 bitmap( file="II.png", type = "pnggray",
119 height = 5, width = 6, res = 300, pointsize=9)
120
121 ##PERMITE DOS GRAFICOS
122 layout(matrix(c(1:2),1,2))
123
124 ##HISTOGRAMA RLS: RECHAZOS HO: BETA_1=0 EN Z^X
125 hist(rechazo, br=2, col="white", labels=TRUE, font.lab=2,
126 font.axis=2, main="C",
127 xlab="1=Rechazo de H0: beta_1=0 en z~x",
128 ylab="Frecuencia")
129
130 ##HISTOGRAMA RNAf: RECHAZOS HO: DELTA_1=0
131 ##EN RESPUESTA RED Z
132 hist(rechazored, br=2, col="darkgray", labels=TRUE,
133 font.lab=2, font.axis=2, main="D",
134 xlab=c("1=Rechazo de H0: delta_1=0 en",
135 "respuesta_red~z_predic"), ylab="Frecuencia")
136
137 ##CIERRA GRAFICOS
138 dev.off()

```

---

---

## Índice alfabético

---

autocorrelación, 5  
autocorrelograma, 5  
cajas negras, 10  
coeficiente de autocorrelación, 5  
cointegración, 6  
entrenamiento de la red, 8  
feedforward, 7  
función de transferencia, 8  
gradiente descendiente adaptativo (ADAPTgd),  
8  
memoria infinita, 3  
pesos, 8  
redes neuronales artificiales multicapa feed-  
forward (RNAf), 7  
regresión lineal simple (RLS), 5  
serie temporal, 2  
sigmoidal tangente hiperbólica (tansig),  
9