



UNIVERSIDAD
DE PIURA

REPOSITORIO INSTITUCIONAL
PIRHUA

PRONÓSTICO DE DEMANDA DE LLAMADAS EN LOS CALL CENTER, UTILIZANDO REDES NEURONALES ARTIFICIALES

Juan Miguel Jiménez Panta

Piura, noviembre de 2013

Universidad de Piura

Área Departamental de Ingeniería Industrial y de Sistemas

Jiménez, J. (2013). *Pronóstico de demanda de llamadas en los call center, utilizando redes neuronales artificiales*. Tesis de pregrado en Ingeniería Industrial y de Sistemas. Universidad de Piura. Facultad de Ingeniería. Programa Académico de Ingeniería Industrial y de Sistemas. Piura, Perú.



Esta obra está bajo una [licencia Creative Commons Atribución-NoComercial-SinDerivadas 2.5 Perú](#)

Repositorio institucional PIRHUA – Universidad de Piura

UNIVERSIDAD DE PIURA
FACULTAD DE INGENIERÍA



“Pronóstico de demanda de llamadas en los call center, utilizando redes neuronales artificiales”

Juan Miguel Jiménez Panta

Asesor: Dra. Ing. Susana Vegas Chiyón

Piura, Noviembre 2013

Dedicada a mi madre, por haberme brindado todo su apoyo durante mi estancia universitaria; y a mi padre, cuyos trabajos en el área de inteligencia artificial, algoritmos genéticos y redes neuronales artificiales, me inspiraron y motivaron para el desarrollo del siguiente trabajo.

Resumen

En la presente tesis se ha construido una red neuronal artificial utilizada para el pronóstico de demanda de llamadas del Centro de Atención Telefónica (Call Center) de clientes de la empresa “ABC” con el propósito de demostrar que dicha herramienta resuelve este tipo de problemas de manera eficiente, encontrándose resultados altamente satisfactorios.

El trabajo se inicia con una descripción de los modelos tradicionales de pronóstico. Así mismo se da un panorama general de las redes neuronales artificiales destacando las diferentes topologías, los algoritmos de aprendizaje, sus aplicaciones y el por qué son útiles estas herramientas. Luego se define el problema a resolver especificando la serie temporal, se diseñan los modelos para la solución al problema de predicción, utilizando la metodología de descomposición de serie de tiempo, el método de ajuste exponencial de Winter, el método de Box-Jenkins (ARIMA), y la red neuronal en diferentes topologías.

Finalmente se implementa la red neuronal utilizando como interface Microsoft Excel y como algoritmo de aprendizaje, el aplicativo «Solver». Se realizan análisis comparativos de los resultados de errores de estimación para la etapa de entrenamiento y validación de los distintos modelos diseñados como solución al problema.

Índice general

Dedicatoria	
Resumen	
Índice general	
Introducción	1
Descripción general de la empresa	3
Objetivos	5
Capítulo 1: Estudio de modelos de pronóstico	7
1.1. Enfoques para la elaboración de pronósticos	7
1.2. Definición de serie de tiempo	8
1.3. Descomposición de una serie de tiempo	8
1.4. Promedio móviles	11
1.5. Método de Winter	12
1.6. Método de Box-Jenkins (ARIMA)	13
Capítulo 2: Fundamentos de las redes neuronales artificiales	25
2.1. Panorama histórico	25
2.2. La red neuronal biológica y la artificial	26
2.3. Funciones de activación	29
2.4. Características de operación de la RNA	30
2.5. Arquitectura de red	32
2.6. Topologías de las RNA	34
2.7. Metodología para la predicción de una serie de tiempo con RNA	36
2.8. Aplicaciones de las RNA	38
Capítulo 3: Definición del problema y diseño de los modelos de predicción	43
3.1. Definición del problema a pronosticar	43
3.2. Análisis exploratorio inicial de la serie de tiempo	44
3.3. Diseño del pronóstico por el método de Winter	47
3.4. Diseño del pronóstico por descomposición de serie de tiempo	48
3.5. Diseño del pronóstico por el método de Box-Jenkins	49
3.6. Diseño de la red neuronal artificial	54

Capítulo 4: Análisis y discusión de resultados	65
Conclusiones y Recomendaciones	69
Bibliografía	71
Anexos	73

Introducción

Siendo el cálculo de pronósticos un problema que enfrentan las empresas cuando desean tomar decisiones para planificar la producción, la programación del personal, y mantener políticas eficientes en un horizonte de planeamiento definido, es necesario que este cálculo tenga un mínimo porcentaje de error, evaluando e integrando las distintas variables que afectan dicho cálculo.

Para minimizar este error, se utilizan métodos de atenuación en los modelos matemáticos. Estos métodos requieren, en muchos casos, mecanismos de adaptación siguiendo el comportamiento de la demanda, para lo cual es necesaria una herramienta de cómputo que detecte estos cambios y ajuste los coeficientes del modelo de pronóstico.

En la empresa “ABC”, la predicción de demanda de llamadas que ingresan a su central de atención telefónica, es de suma importancia para garantizar el nivel de atención y de servicio¹, lo que se dificulta por el comportamiento errático de la demanda. Dicha predicción se está realizando con una metodología propia que considera un histórico de tres meses para hallarla, y que no está obteniendo los resultados de predicción deseados. Esta situación ha motivado que en el presente trabajo se enfrente al problema, buscando herramientas o modelos que ayuden a la solución de este problema, y a la reducción del error en la predicción de llamadas.

La estadística clásica para poder predecir utiliza, con mucha frecuencia, las metodologías de Box-Jenkins (Box, Jenkins y Reinsel, 1994). Estas metodologías requieren de un conocimiento profundo de los modelos. Los pasos que se realizan dependen del tipo de datos. Estos datos deben ser estacionarios; si no son estacionarios hay que buscar convertirlos a estacionarios. Se analizan los gráficos de correlaciones, se evalúa la adecuación del modelo, se mide el error de predicción, etc. Estos procedimientos pueden resultar complejos, los cuales conducen a intentar dar solución a este problema mediante las nuevas técnicas computacionales que han tomado auge en los últimos años, como lo son las redes neuronales artificiales.

Las redes neuronales artificiales o RNA son técnicas que intentan imitar el pensamiento del ser humano para tomar mejores decisiones. Esta herramienta permite que

¹Nivel de atención: Ratio que mide la cantidad de llamadas aceptadas en el sistema entre el total de llamadas recibidas.

Nivel de Servicio: Ratio que mide la cantidad de llamadas atendidas en menos de 20 segundos entre el total de llamadas aceptadas. (*Indicadores de la empresa "ABC"*)

los coeficientes del modelo se adapten dinámicamente y consigue estimaciones con un error muy pequeño; a tal efecto se construyó una red neuronal artificial para pronosticar la demanda de llamadas del “Call Center” de clientes negocios de la empresa “ABC”.

Descripción general de la empresa

“ABC” es una empresa del sector de las telecomunicaciones que brinda servicios de telefonía fija y móvil, banda ancha y televisión por cable a personas, negocios y empresas de todo el país.

Su actividad se centra fundamentalmente en los servicios de telefonía fija y telefonía móvil, con la banda ancha como herramienta clave de comunicación entre negocios y empresas, quienes utilizan cada vez más las tecnologías de la información como ventaja competitiva para la estrategia del negocios, así como para el soporte en las operaciones del día a día.

Actualmente es uno de los operadores de mayor cobertura, ganando cada vez más clientes a quienes tiene que atender por sus distintos canales de atención. Algunos de estos canales de comunicación son los centros de atención presencial, su página web, y su central de atención telefónica, donde los clientes buscan solucionar sus consultas o cualquier inconveniente que tengan con el servicio.

En el mercado actual existen otras tres empresas que compiten brindando los mismos servicios de telecomunicaciones, convirtiéndose no solo la calidad de los productos ofrecidos en una variable principal de competitividad, sino también la calidad del servicio atención y post-venta que puedan brindarle a sus clientes.

Por ello la empresa “ABC” está considerando dentro de sus operaciones de venta y post-venta indicadores de diagnóstico de la calidad del servicio ofrecido.

El objetivo es garantizar la satisfacción, fidelización y permanencia de sus actuales y nuevos clientes.

Objetivos

Objetivo general

Determinar si es factible utilizar el modelo de redes neuronales artificiales para la predicción de demanda en la central de atención telefónica de la empresa “ABC”

Objetivos específicos

1. Verificar si es factible utilizar el modelo de redes neuronales artificiales para el pronóstico de demanda de llamadas en las centrales de atención telefónica.
2. Determinar qué topología de la red genera un mejor resultado (menos error).
3. Diseñar pronósticos de demanda mediante los modelos de Box-Jenkins, Winter y descomposición de series de tiempo; y comparar sus resultados con los obtenidos con la red neuronal artificial, para determinar qué metodología ofrece el mejor resultado de predicción.
4. Hacer una breve descripción de las observaciones y/o problemas que se encontraron en la elaboración de los diseños de cada modelo.

Capítulo 1

Estudio de modelos de pronósticos

Pronosticar es el arte de especificar información significativa acerca del futuro. Se menciona que los pronósticos jamás son perfectos debido a que, básicamente, se utilizan métodos que generan pronósticos sobre la base de la información previa. Los pronósticos serán menos confiables mientras mayor sea el horizonte que se va a pronosticar (Mendoza, 2011, p. 4)

Estimar el costo de producción de un producto puede llevar a realizar un estudio de regresión lineal simple. Pronosticar la demanda futura de un producto puede llevar a utilizar los métodos de extrapolación y de suavización.

1.1. Enfoques para la elaboración del pronóstico

Hay dos importantes tipos de métodos cuantitativos de predicción: los métodos estadísticos (o de extrapolación) y los métodos determinísticos (o de pronóstico causal), que son utilizados con frecuencia para hacer pronósticos.

Los métodos o técnicas estadísticas se utilizan para pronosticar los valores futuros de series de tiempo a partir de valores anteriores de una serie temporal. Estos métodos suponen que los patrones anteriores y las tendencias en los períodos futuros continuarán; por lo tanto la información anterior relacionada con la serie temporal se utiliza para generar los valores futuros de la serie temporal. Estos métodos no consideran el hecho de que “ocasionó” los datos anteriores; simplemente se asume que las tendencias y los patrones anteriores continuarán en el futuro. En este capítulo se expondrán algunas de las técnicas de pronósticos más utilizados, como los promedios móviles, la atenuación exponencial de Winter, la descomposición de series de tiempo y estimaciones de tendencia y la metodología de Box-Jenkins, siendo esta última uno de los modelos para pronósticos líneas con mayor complejidad en desarrollo del modelo, pero cuya utilización en pronósticos ha sugerido buenos resultados.

Los métodos determinísticos o de pronósticos causales pretenden pronosticar los valores futuros de una variable (llamada variable dependiente) con la ayuda de la información anterior, a fin de estimar la relación entre la variable dependiente y una o más variables independientes. Dentro de este tipo de técnicas se incluyen los modelos de regresión múltiple de series de tiempo.

1.2. Definición de serie de tiempo

Una serie de tiempo es una secuencia cronológica de observaciones de una variable particular (Bowerman y O'Connell, 1993).

Estas observaciones serán denotadas por $\{y(t_1), y(t_2), \dots, y(t_n)\} = \{y(t) : t \in T \subseteq \mathbb{R}\}$ con $x(t_i)$ el valor de la variable y en el instante t_i . Si $T = \mathbb{Z}$ se dice que la serie de tiempo es discreta y si $T = \mathbb{R}$ se dice que la serie de tiempo es continua. Cuando $t_{i+1} - t_i = k$ para todo $i = 1, \dots, n-1$, se dice que la serie es equiespaciada; en caso contrario será no equiespaciada.

Este trabajo está referido a una serie de tiempo discreta, equiespaciada por lo que es posible asumir y sin pérdida de generalidad que: $\{y(t_1), y(t_2), \dots, y(t_n)\} = \{y(1), y(2), \dots, y(n)\}$.

1.3. Descomposición de una serie de tiempo

El primer paso para analizar una serie de tiempo es graficarla, lo que permitirá descubrir patrones históricos, componentes, que puedan ser útiles en la predicción. Para identificar estos patrones es conveniente pensar que una serie de tiempo se compone de varios componentes (Bowerman y O'Connell, 1993):

1. Tendencia $T(t)$: es el componente de largo plazo que representa el crecimiento o disminución en la serie sobre un periodo amplio. También puede definirse como cambios en la media
2. Ciclo $C(t)$: se refiere a movimientos hacia arriba y abajo en forma de onda alrededor de la tendencia. Los patrones cíclicos tienden a repetirse en los datos aproximadamente cada dos, tres o más años.
3. Variación estacional $S(t)$: se refiere al patrón periódico que ocurre y se repite cada determinado tiempo. En el caso de series mensuales, el componente estacional mide la variabilidad de las series cada enero, febrero, etc.
4. Componente irregular $A(t)$: se refiere a la parte de la serie de tiempo que no sigue un patrón regular, ni reconocible. Tal componente representa "lo que queda" en una serie de tiempo después de que la tendencia, el ciclo y la variación estacional han sido explicadas.

Existen tres modelos de series de tiempos, que generalmente se aceptan como buenas aproximaciones a las verdaderas relaciones, entre los componentes de los datos observados. Estos son:

- a) Aditivo: $Y(t) = T(t) + C(t) + S(t) + A(t)$
- b) Multiplicativo: $Y(t) = T(t) \cdot C(t) \cdot S(t) \cdot A(t)$
- c) Mixto: $Y(t) = T(t) \cdot C(t) \cdot S(t) + A(t)$

Donde:

- $Y(t)$ serie observada en instante t
- $T(t)$ componente de tendencia
- $C(t)$ componente cíclico
- $S(t)$ componente estacional
- $A(t)$ componente aleatoria (accidental)

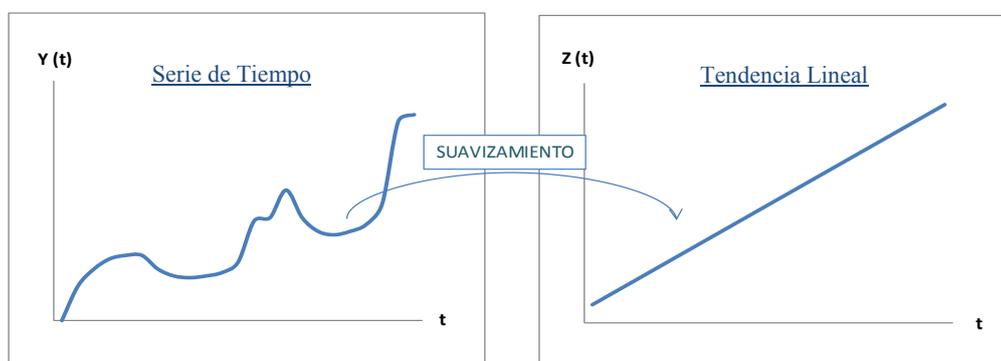
Una suposición usual es que $A(t)$ es un componente aleatorio o ruido blanco con media cero y varianza constante.

Un modelo aditivo (1), es adecuado, por ejemplo, cuando $C(t)$ o $S(t)$ no depende de otras componentes, como $T(t)$. Si por el contrario el componente cíclico y estacional varían con la tendencia, el modelo más adecuado es un modelo multiplicativo (2).

1.3.1. Estimación de la tendencia

Una forma de visualizar la tendencia es mediante suavizamiento de la serie. La idea central es definir, a partir de la serie observada, una nueva serie que suaviza los efectos ajenos a la tendencia (estacionalidad, efectos aleatorios), de manera que podamos determinar la dirección de la tendencia (ver figura).

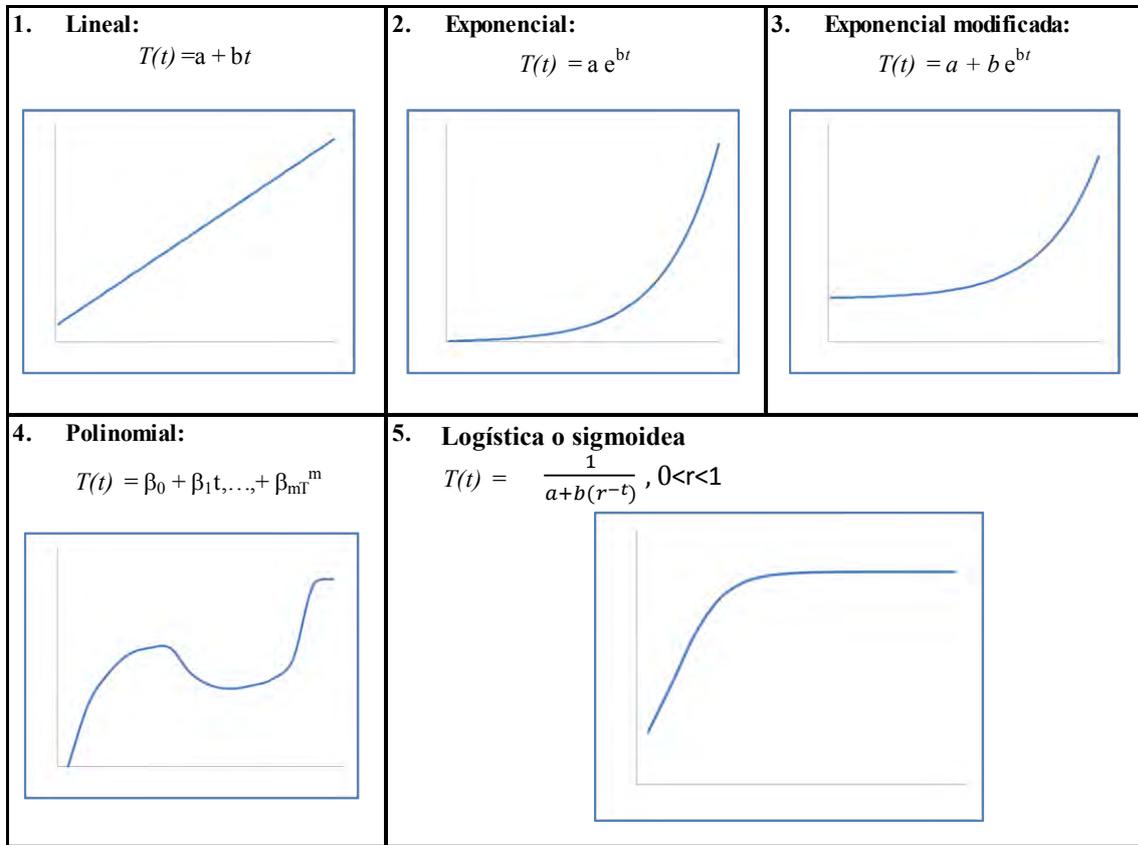
Figura 1
Representación de la tendencia mediante el ajuste lineal de la serie de tiempo



Fuente: (Arellano, 2001, sección Ajuste de una función, 1)

Ajuste de una función:

Figura 2
Formas de curvas suavizadas



Fuente: Arellano, 2001, sección Suavizamiento. Filtros lineales, 1

1.3.2. Estimación de la estacionalidad

La estimación de la estacionalidad no sólo se realiza con el fin de incorporarla al modelo para obtener predicciones, sino también con el fin de eliminarla de la serie para visualizar otras componentes como tendencia y componente irregular que se pueden confundir en las fluctuaciones estacionales.

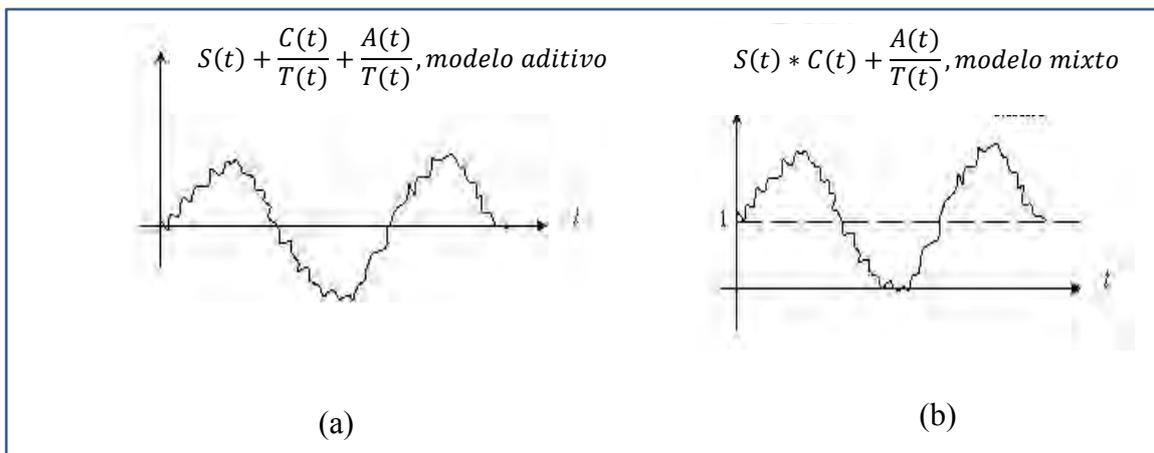
De acuerdo con los modelos clásicos (sección 1.3.1), se asume el siguiente modelo para $T(t)$,

$$a) Y(t) - T(t) = S(t) + C(t) + A(t) \quad , \quad \text{Aditivo}$$

$$b) \frac{Y(t)}{T(t)} = S(t) * C(t) + \frac{A(t)}{T(t)} \quad , \quad \text{Mixto}$$

Una vez removida la tendencia, se obtienen los siguientes gráficos. En la figura 3(a) aparece el modelo aditivo y en la 3(b) el modelo mixto.

Figura 3
Representación de la curvas de estacionalidad



Fuente: Arellano, 2001, sección Estimación de la estacionalidad, 3

Si no hay tendencia, se espera

$$\forall t, T(t) = 0 ; \text{ en modelos aditivos}$$

$$\forall t, T(t) = 1 ; \text{ en modelos mixtos}$$

Como $S(t) = S(t+12) = S(t+24) \dots$ para una serie mensual, entonces basta estimar $S(1), S(2), S(3), \dots, S(12)$. Para una serie trimestral, bastaría conocer: $S(1), S(2), S(3)$ y $S(4)$.

Suponga que se ha estimado la tendencia por alguno de los métodos vistos en la sección previa. Sea $\hat{T}(t)$ la estimación de la tendencia. Entonces:

- Si el modelo es *aditivo*, $Z(t) = Y(t) - \hat{T}(t)$, $t = 1, \dots, n$ representa la serie con los efectos de tendencia removidos.
- Análogamente, si el modelo es *mixto*, $Z(t) = \frac{Y(t)}{\hat{T}(t)}$ representa la serie, una vez removidos los efectos de tendencia.

Estas series generadas a partir de la original por eliminación de la tendencia se denominan «series de residuos» y deberán contener predominantemente fluctuaciones estacionales y/o cíclicas. Para estimar la estacionalidad se requiere haber decidido el modelo a utilizar (mixto, multiplicativo o aditivo), lamentablemente esto no es siempre claro, ya sea porque no contamos con información a priori para suponerlo o porque el gráfico no ha dejado evidencia suficientemente clara como para decidirnos por alguno de ellos. En tal situación se propone calcular ambas series residuales y elegir aquella cuyos valores correspondientes a una estación dada oscilen menos en torno a su promedio.

1.4. Promedios móviles

El método de promedios móviles es una herramienta usualmente utilizada para pronósticos rápidos, baratos, sencillos y de corto plazo, donde se le da un mayor énfasis a

las observaciones más recientes de la serie de tiempo, para el pronóstico del siguiente periodo (Hanke E., Reitsch G., 1996)

El objetivo es eliminar (atenuar) de la serie, las componentes estacionales e irregulares. (Arellano, 2001, sección Promedios móviles, 1)

$$Z_t = \frac{(Y_{t-1} + Y_{t-2} + Y_{t-3} + \dots + Y_{t-n})}{n}$$

donde,

Z_t = valor de pronóstico para el siguiente periodo,

Y_{t-1} = valor real en el periodo (t-1)

n = número de términos en el promedio móvil

Para una serie mensual con estacionalidad anual (s = 12), la serie suavizada se obtiene,

$$Z_k = \frac{(Y_{k-1} + Y_{k-2} + \dots + Y_{k-11} + Y_{k-12})}{12}, \quad 13 \leq k$$

En este caso se asignan ponderaciones iguales a cada observación. Al hacerse disponible, cada nuevo punto de datos se incluye en el promedio y se descarta el más antiguo. La proporción de respuesta a los cambios en el patrón subyacente de los datos depende del número de periodos, n, que se incluyen en el promedio móvil.

1.5. Métodos de atenuación exponencial ajustada a la tendencia y a la variación estacional: modelo de Winter

La atenuación exponencial es un método utilizado para revisar constantemente una estimación a la luz de experiencias más recientes. El método pondera los valores anteriores de una serie, con parámetros de forma decreciente (exponencial) asignándole mayor peso a la información más reciente de la serie.

El modelo de atenuación exponencial lineal y estacional de tres parámetros de Winter, utiliza una constante de atenuación para los valores estimados de la tendencia (enfoque de Brown); una ecuación para atenuar en forma directa la tendencia y la pendiente empleando diferentes constantes de atenuación para cada una de ellas (modelo de Holt); y una ecuación adicional para determinar la estacionalidad. Este método proporciona una forma fácil de explicar la estacionalidad en un modelo, cuando los datos tienen un patrón estacional (Hanke E., Reitsch G., 1996).

Las ecuaciones que emplea el modelo de Winter son:

1. La serie exponencial atenuada:

$$A_t = \alpha \frac{Y_t}{S_{t-L}} + (1 - \alpha)(A_{t-1} + T_{t-1})$$

2. La estimación de la tendencia:

$$T_t = \beta(A_t - A_{t-1}) + (1 - \beta)T_{t-1}$$

3. La estimación de la estacionalidad:

$$S_t = \gamma \frac{Y_t}{A_t} + (1 - \gamma)S_{t-L}$$

4. El pronóstico de p periodos en el futuro:

$$\hat{Y}_{t+p} = (A_t - pT_t)S_{t-L+p}$$

donde,

A_t = nuevo valor atenuado

α = constante de atenuación ($0 < \alpha < 1$)

Y_t = nuevo valor real de la serie en el periodo t

β = constante de atenuación de la estimación de la tendencia ($0 < \beta < 1$)

T_t = estimación de la tendencia

γ = constante de atenuación de la estimación de la estacionalidad ($0 < \gamma < 1$)

S_t = estimación de la estacionalidad

p = periodos a estimar a futuro

L = longitud de la estacionalidad

Y_{t+p} = pronóstico de p periodos a futuro

La atenuación exponencial es una técnica popular para los pronósticos de corto plazo. Sus mejores ventajas son un bajo costo y simplicidad.

1.6. Métodos de Box-Jenkins (ARIMA)

Existen dos metodologías desarrolladas por Box y Jenkins que permiten predecir valores futuros de una serie de tiempo basándose en valores pasados de una sola variable o dos variables entre las que existe una relación causal (Box, Jenkins y Reinsel, 1994).

Para el caso específico de solución al problema expuesto, se expondrá y desarrollará en el método de Box-Jenkins univariable, cuyos resultados se compararán con la solución que brinda las redes neuronales artificiales y los otros métodos clásicos de predicción de series de tiempo.

Modelos univariantes de series temporales:

$$Y_t = f(Y_{t-1}, Y_{t-2}, \dots) + a_t$$

$$Y_t = \hat{Y}_t + a_t$$

donde,

Y_t = Serie observada

\hat{Y}_t = Componente predecible

a_t = Componente Aleatorio (ruido blanco)

Existen dos enfoques básicos para obtener Y_t : postular la forma de \hat{Y}_t , siendo \hat{Y}_t la parte predecible, u obtener a_t y en la serie, siendo a_t la parte no predecible.

Los métodos clásicos buscan \hat{Y}_t y el enfoque Box-Jenkins se centra en a_t

1.6.1. ARIMA: modelo

El proceso Autorregresivo Integrado de Promedio Móvil: ARIMA (de sus siglas en inglés, “Autoregressive Integrated Moving Average”) es denominado también método (univariante) de Box y Jenkins (Box, Jenkins y Reinsel, 1994)

Este enfoque de Box y Jenkins es una de las metodologías de más amplio uso para el análisis de series de tiempo. Es popular debido a su generalidad y cuenta con programas de computación bien documentados. Si bien Box y Jenkins no fueron los creadores ni quienes contribuyeron de manera más importante en el campo de los modelos ARMA (“Autoregressive Moving Average”), sí fueron quienes los popularizaron y los hicieron accesibles (Maddala, 1996).

La metodología de Box y Jenkins requiere que la serie sea estacionaria. Si la serie no es estacionaria, en general se puede convertirá una serie estacionaria mediante el método de diferenciación.

Una serie de tiempo es estacionaria si su media, su varianza y su covarianza (en los distintos rezagos) permanecen contantes sin importar el momento en el cual se midan. Para el tratamiento de la no estacionariedad en la media se propone la diferenciación sucesiva de la serie, aprovechando la propiedad de que gozan una gran parte de los procesos estocásticos, de convertirse en estacionarios al diferenciarlos cierto número de veces. Las primeras diferencias de los valores de la serie de tiempo y_1, y_2, \dots, y_n son:

$$z_t = y_t - y_{t-1} \text{ donde } t = 2, 3, \dots, n$$

Las segundas diferencias son:

$$z_t = (y_t - y_{t-1}) - (y_{t-1} - y_{t-2})$$

La metodología de Box y Jenkins aplica métodos autorregresivos (AR), promedio móviles (MA), autorregresivos y de promedio móvil (ARMA) y, autorregresivo integrado de promedio móvil (ARIMA). A continuación se explicarán brevemente cada uno de estos procesos suponiendo, de manera general, que las series de tiempo son estacionarias. Considérese $Y_t = y_t - \mu$, donde y_t son los valores originales de la serie de tiempo, μ es la media de todos los valores de la series, y Y_t es la desviación del proceso respecto a la media. a_t es una perturbación aleatoria o ruido blanco (con media cero, varianza constante y covarianza cero).

1.6.1.1. El operador de retardo B

Antes de describir los diferentes métodos incluidos en la metodología de Box y Jenkins, se mostrará un resumen de la notación utilizada, el operador de retardo B. (García y Sanchez, nd)

Sobre la notación:

$$Bz_t = z_{t-1}$$

$$B^2z_t = z_{t-2}$$

$$B^pz_t = z_{t-p}$$

$$\nabla z_t = (1 - B)z_t = z_t - z_{t-1}$$

$$(1 - \Phi B)z_t = a_t \Leftrightarrow z_t = (1 + \Phi B + \Phi^2 B^2 + \dots)a_t = a_t + \Phi a_{t-1} + \Phi a_{t-2} + \dots$$

$$(1 - \theta B)a_t = z_t \Leftrightarrow a_t = (1 + \theta B + \theta^2 B^2 + \dots)z_t = z_t + \theta z_{t-1} + \theta z_{t-2} + \dots$$

1.6.1.2. Proceso autorregresivo (AR)

Definición del modelo (Hanke, E. y Reitsch, G., 1996):

$$Y_t = \Phi_1 Y_{t-1} + \Phi_2 Y_{t-2} + \dots + \Phi_p Y_{t-p} + a_t$$

donde p denota el número de términos autorregresivos y $\Phi_1, \Phi_2, \dots, \Phi_p$ es un conjunto finito de pesos o parámetros. Se le denomina proceso autorregresivo de orden p o AR(p). Los términos Φ son coeficientes determinados por regresión lineal. Esos coeficientes son multiplicados por los p valores previos de la serie. Este modelo relaciona el valor pronóstico de Y_t con la suma ponderada de sus valores en periodos pasados, más una perturbación aleatoria en el tiempo t . Equivalentemente, y haciendo uso del operador de retardo B , un proceso autorregresivo puede expresarse como:

$$(1 - \Phi_1 B - \Phi_2 B^2 - \dots - \Phi_p B^p)Y_t = a_t$$

o en forma abreviada

$$\Phi_p(B)Y_t = a_t$$

En la figura 4(a) se muestra la ecuación del modelo AR(1) y en la figura 4(b) el modelo AR(2), y los comportamientos teóricos de los coeficientes de autocorrelación (ACF) y de autocorrelación parcial (PACF), que se explicará más adelante en este capítulo.

Figura 4(a)
 $AR(1): y_t = \Phi_1 y_{t-1} + a_t$

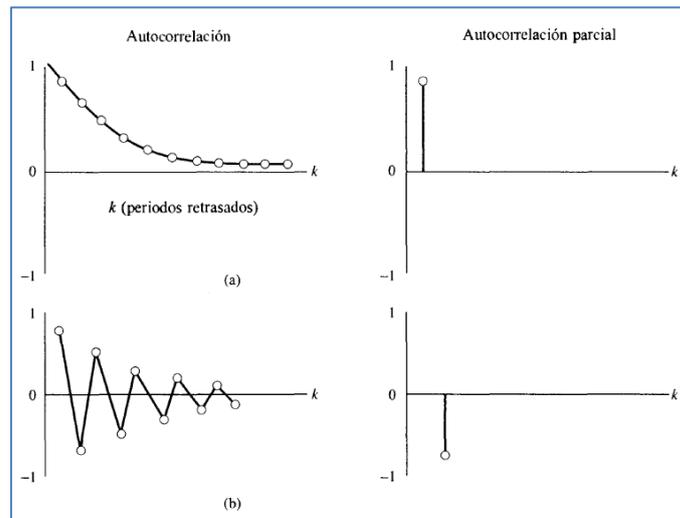
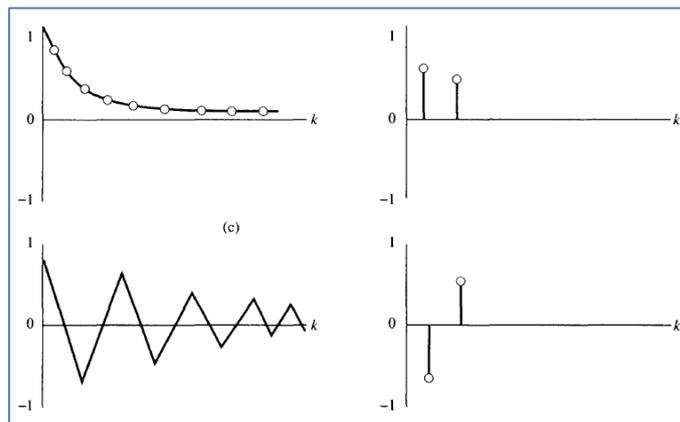


Figura 4(b)
 $AR(2): y_t = \Phi_1 y_{t-1} + \Phi_2 y_{t-2} + a_t$



Fuente: Hanke y Reitsch, 1996, p. 433.

1.6.1.3. Proceso de promedio móvil (MA)

Definición del modelo (Hanke, E. y Reitsch, G., 1996):

$$Y_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q}$$

donde q denota el número de términos de promedio móvil y $\theta_1, \theta_2, \dots, \theta_q$, son el conjunto finito de pesos o parámetros. A este modelo se le denomina proceso de promedio móvil de orden q , o MA(q). Los términos θ son coeficientes determinados mediante métodos iterativos no lineales y son multiplicados por los q errores de predicción previos. En este proceso se relaciona el valor pronóstico a los errores de predicciones previas.

El modelo MA(q) también puede describirse equivalentemente como:

$$Y_t = (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q) a_t$$

O en forma abreviada

$$Y_t = \theta_q(B)a_t$$

En la figura 5(a) se muestra la ecuación del modelo MA(1) y en la figura 5(b) el modelo MA(2), y los comportamientos teóricos de los coeficientes de autocorrelación (ACF) y de autocorrelación parcial (PACF).

Figura 5(a)
MA(1): $Y_t = a_t - \theta_1 a_{t-1}$

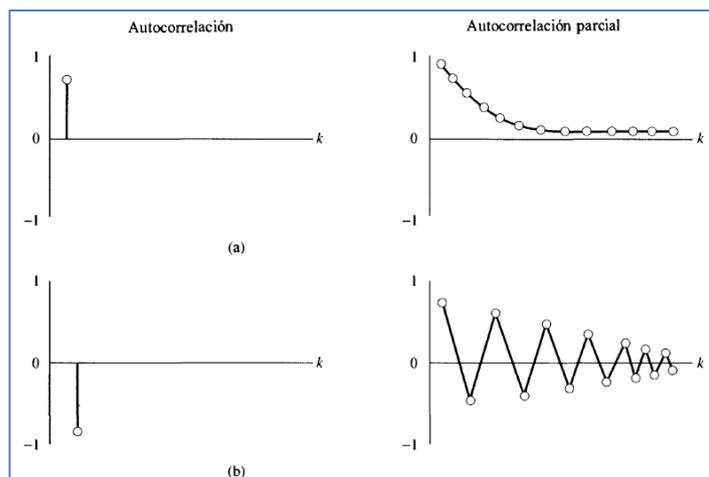
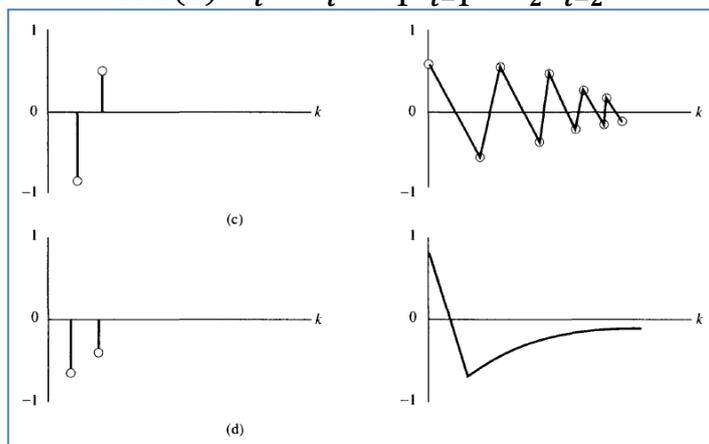


Figura 5(b)
MA(1): $Y_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2}$



Fuente: Hanke y Reitsch, 1996, p. 434.

1.6.1.4. Proceso autorregresivo y promedio móvil (ARMA)

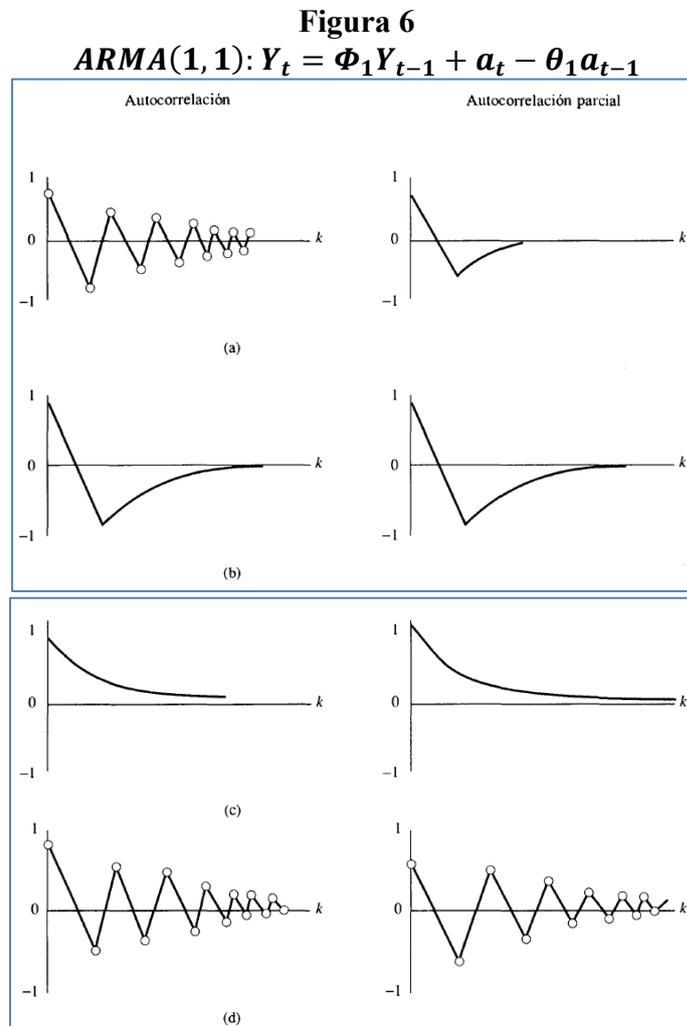
Además de los modelos AR y MA, ambos pueden combinarse en un tercer tipo de modelo denominado ARMA (Hanke, E. y Reitsch, G., 1996).

Definición del modelo:

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \dots - \theta_q a_{t-q}$$

Los modelos ARMA(p,q) utilizan las combinaciones de errores anteriores y valores anteriores y ofrecen un potencial para ajustar modelos que no pudieron ajustarse de forma adecuada mediante los modelos AR y MA por sí solos.

En la Figura 6, se muestra la ecuación del modelo ARMA(1,1) y el comportamiento teóricos de los coeficientes de autocorrelación (ACF) y de autocorrelación parcial (PACF).



Fuente: Hanke y Reitsch, 1996, p. 435.

1.6.1.5. Proceso Autorregresivo Integrado de Promedio Móvil (ARIMA)

Muchos modelos de series de tiempo analizados están basados en el supuesto de que las series de tiempo consideradas son estacionarias (media y varianza constantes); pero se sabe que muchas series de tiempo son no estacionarias, es decir, son integradas.

Si se debe diferenciar una serie de tiempo d veces para hacerla estacionaria y luego aplicar a ésta el modelo ARMA(p,q), se dice que la serie de tiempo original sigue un *proceso autorregresivo integrado de promedio móvil* o *ARIMA(p,d,q)*, donde p denota el número de términos autorregresivos, d el número de veces que la serie debe ser diferenciada para hacerse estacionaria y q el número de términos de promedio móvil. (Gujarati, 1997)

El modelo ARIMA(p,d,q) puede ser escrito como:

$$\Phi_p(B)(1-B)^d Y_t = \theta_q(B) a_t$$

Los procesos ARIMA son suficientes para explicar procesos como tendencia; pero incapaces de representar procesos con estacionalidad y se hace necesaria una generalización de estos para lograr explicar los comportamientos estacionales. Los modelos estacionales consideran los retrasos del proceso y de la perturbación aleatoria periódicamente, es decir, cada s periodos. Por ejemplo, si los datos son mensuales, es lógico considerar el periodo $s=12$. El objeto de estos retardos estacionales (s) es explicar la dependencia que tienen entre sí iguales periodos de años sucesivos, por ejemplo, enero del 2010, con enero del 2011 y enero del 2012 directamente y a través de errores (perturbaciones no explicadas) asociados a estos periodos.

Los modelos estacionales se denotan anteponiéndoles la letra S, y el orden de sus parámetros se escribe con mayúsculas, como sigue: SARMA(P,Q). Los modelos SARMA son análogos al proceso ARMA pero considerando los retardos del ruido blanco y del proceso de s en s .

Sin embargo estos modelos SARMA no son capaces de explicar todos los movimientos estacionales, pues si éstos crecieran de año en año, los SARMA serían incapaces de recoger esta evolución, pues al igual que los ARMA son estacionarios. Esta dificultad se resuelve a través de los modelos autorregresivos de promedio móviles integrados estacionales SARIMA(P,D,Q)

La unión de modelos estacionales con modelos no estacionales conduce a un modelo de gran capacidad de adaptación que puede reflejar la tendencia y la estacionalidad de una serie (enfoque de Box y Jenkins). La combinación de estos modelos se logra a través de la multiplicación de los operadores polinomiales que caracterizan a cada modelo, obteniendo los modelos conocidos ARIMA(p,d,q)xSARIMA(P,D,Q); también denotados como sigue: ARIMA(p,d,q)x(P,D,Q)_s.

El modelo ARIMA(p,d,q)x(P,D,Q)_s puede escribirse como:

$$\Phi_p(B)\Phi_P(B^s)(1-B)^d(1-B^s)^D Y_t = \delta + \theta_q(B)\Theta_Q(B^s)a_t$$

donde:

$$\Phi_p(B) = (1 - \Phi_1 B - \dots - \Phi_p B^p) \quad : \text{operador AR regular de orden "p"}$$

$$\Phi_P(B^s) = (1 - \Phi_1 B^s - \dots - \Phi_P B^{Ps}) \quad : \text{operador AR estacional de orden "P"}$$

$$(1 - B)^d = (\nabla)^d \quad : \text{"d" diferencias regulares}$$

$$(1 - B^s)^D = (\nabla_s)^D \quad : \text{"D" diferencias estacionales}$$

$$\theta_q(B) = (1 - \theta_1 B - \dots - \theta_q B^q) \quad : \text{operador MA regular de orden "q"}$$

$$\Theta_Q(B^s) = (1 - \theta_1 B^s - \dots - \theta_Q B^{Qs}) \quad : \text{operador MA estacional de orden "Q"}$$

$$\delta \quad : \text{es una constante}$$

1.6.1.6. Herramientas para la determinación de los órdenes del modelo

El número de términos en la parte AR y MA del modelo final no son escogidos arbitrariamente. Para esto, Box y Jenkins proporcionan un método estructurado que determina cual modelo ajusta mejor a la serie en cuestión y recomiendan que el modelo se mantenga tan simple como sea posible. En general no hay más de tres términos AR o MA. (Wedding y Cios, 1996).

La herramienta principal para determinar los órdenes del modelo son los correlogramas simple y parcial, que son la representación gráfica de las funciones de autocorrelación simple (FAS o ACF, por sus siglas en inglés “Partial AutoCorrelation Function”) y funciones de autocorrelación parcial (FAP o PACF por sus siglas en inglés, “Partial AutoCorrelation Function”).

Función de autocorrelación simple:

Considere la serie de tiempo Y_1, Y_2, \dots, Y_n . La autocorrelación simple muestral en el retraso k , denotada por r_k es:

$$r_k = \frac{\sum_{t=1}^{n-k} (Y_t \times Y_{t+k})}{\sum_{t=1}^n Y_t^2}$$

Esta cantidad mide la relación lineal entre las observaciones de la serie de tiempo separadas por un retraso de k unidades de tiempo. La autocorrelación r_k toma valores entre -1 y 1.

Para conocer si el valor de r_k es significativo se utiliza el estadístico t_{r_k} :

$$t_{r_k} = \frac{r_k}{s_{r_k}}$$

donde s_{r_k} es el error estándar de r_k , dado por:

$$s_{r_k} = \sqrt{\frac{1 + 2 \sum_{j=1}^{k-1} r_j^2}{n}}$$

La función de autocorrelación simple es el conjunto de autocorrelaciones simples muestrales en los retrasos $k=1,2,\dots$; a la representación gráfica de estas autocorrelaciones se le denomina correlograma simple. (Bowerman y O’Connell, 1993)

Función de autocorrelación parcial

La autocorrelación parcial muestral en el retraso k es:

$$r_{kk} = \begin{cases} r_1 & , \text{ si } k = 1 \\ \frac{r_k - \sum_{j=1}^{k-1} r_{k-1,j} r_{k-j}}{1 - \sum_{j=1}^{k-1} r_{k-1,j} r_j} & , \text{ si } k = 2,3, \dots \end{cases}$$

donde $r_{kj} = r_{k-1,j} - r_{kk} r_{k-1,k-j}$ para $j=1,2,\dots, k-1$

El valor de estas autocorrelaciones puede pensarse intuitivamente como la relación de las observaciones de la serie de tiempo separadas por un retraso de k unidades de tiempo, eliminando el efecto de las observaciones intermedias.

Para conocer si el valor de r_{kk} es significativo se utiliza el estadístico $t_{r_{kk}}$:

$$t_{r_{kk}} = \frac{r_{kk}}{s_{r_{kk}}}$$

donde $s_{r_{kk}}$ es el error estándar de r_{kk} , dado por:

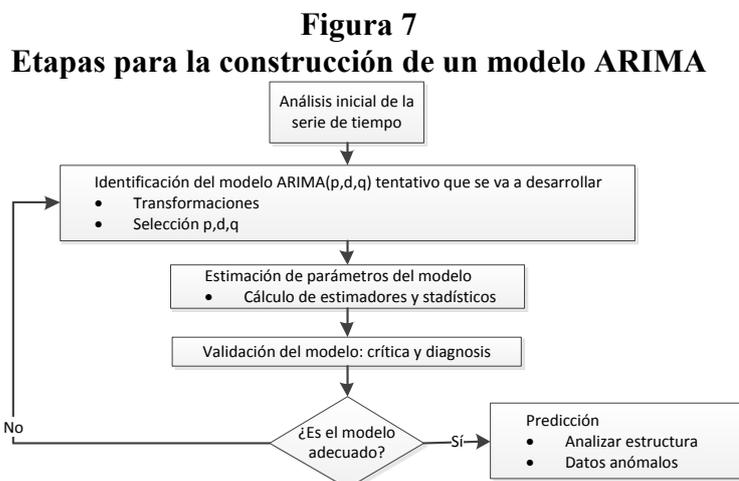
$$s_{r_{kk}} = \frac{1}{\sqrt{n}}$$

La función de autocorrelación parcial es el conjunto de las autocorrelaciones parciales muestrales en los retrasos $k=1,2,\dots,M$. A la representación gráfica de estas autocorrelaciones se le denomina correlograma parcial. (Bowerman y O'Connell, 1993)

1.6.1.7. Etapas del método univariable de Box y Jenkins o metodología ARIMA

Para la construcción de un modelo ARIMA que recoja suficientemente bien las características de la serie, se hace uso de la metodología de Box y Jenkins que puede ser estructurada en cinco etapas.

En la Figura 7 se muestran las 5 etapas para la construcción de un modelo ARIMA(p,d,q)



1. Análisis inicial de la serie:

Consiste en un primer análisis, mediante gráficos y pruebas estadísticas, para obtener estacionariedad (media y varianza constante)

- a) Se grafica la serie a través del tiempo, de manera de observar a priori sus componentes: tendencia, estacionalidad y ciclos. Podría notarse la necesidad de aplicar diferencias, en la parte no estacional o regular, para hacer que la media sea constante. Podría observarse estacionalidad mediante una pauta repetida de acuerdo con el periodo estacional "s", lo que implicaría la necesidad de diferencias en la

parte estacional. Esto se confirmaría en la segunda etapa mediante los correlogramas.

- b) Se realiza un diagrama de caja que permita estudiar el comportamiento de la varianza. Al sospechar que la varianza no es constante se recomienda realizar alguna transformación a la serie (Ej. aplicando logaritmo).

2. Identificación del modelo:

En esta etapa se debe sugerir un conjunto reducido de posibles modelos.

- a) Selección del conjunto de estimación: conjunto de datos que se usará para la estimación y adecuación del modelo; y del conjunto de predicción: conjunto de datos que se guardará para evaluar las predicciones.
- b) Determinación de los correlogramas o funciones de autocorrelación simple y parcial para establecer, conjuntamente con lo observado en la primera etapa, el número de diferencias que se aplicarán y que convertirán el proceso en estacionario. Una serie no estacionaria es estacionaria en la parte regular si en el correlograma simple se observa que los valores decrecen lentamente en los retardos 1,2,3...
- c) Determinación de los órdenes del componente autorregresivo (p) y promedio móvil (q) del modelo ARMA(p,q), haciendo uso de los patrones que se observan en los correlogramas simple y parcial:

Tabla 1: Tipo de modelo ARMA según las características de los correlogramas simple y parcial

Correlograma Simple	Correlograma Parcial	Modelo
Decae lentamente	Se corta después del retardo p	AR(p)
Se corta después del retardo q	Decae lentamente	MA(q)
Decae lentamente	Decae lentamente	ARMA(p,q)

- d) Estudio de la estacionalidad. En caso de presentar estacionalidad con periodo "s", se aplica una diferencia estacional $(1-B)^s$ para convertir la serie en estacionaria. La estacionalidad se manifiesta en el gráfico de la serie (etapa 1) y en el correlograma simple que presentará valores positivos que decrecen lentamente en los retardos s, 2s, 3s, ...
- e) Determinación de los órdenes P y Q del procesos SARMA(P,Q), de la misma manera que en la parte regular, pero, considerando solamente los valores de los correlogramas en los retardos s, 2s, 3s.
- f) Especificación del modelo ARIMA identificado y sugerencias de otros modelos posibles para la verificación en la elección del modelo más adecuado para el pronóstico de la serie de tiempo.

3. Estimación de los parámetros del modelo:

Una vez identificado el modelo $ARIMA(p,d,q) \times (P,D,Q)_s$

$$\Phi_p(B)\phi_p(B^s)(1-B)^d(1-B^s)^D Y_t = \delta + \theta_q(B)\theta_q(B^s)a_t$$

Los valores de los parámetros $\Phi, \phi, \theta, \theta$ y δ se estiman mediante la minimización de la suma de cuadrados de los errores a_t . Generalmente estas estimaciones se hacen con la ayuda de herramientas computacionales como SPSS, Minitab, etc.

4. Validación del modelo:

Inicialmente la adecuación del modelo se logra graficando la serie original y la ajustada por el modelo ARIMA, de manera que se puedan observar sus similitudes y diferencias. Finalmente se considera un modelo ARIMA adecuado para representar el comportamiento de una serie si se cumple lo siguiente:

- a) Los residuos, diferencias entre el valor original de la serie y el valor estimado por el modelo, se aproximan al comportamiento de un ruido blanco (media cero, varianza σ^2 y covarianza 0). Al observar los correlogramas no deberán observarse valores significativamente diferentes de cero, como indicativo de ausencia de correlación serial, así como tampoco patrones (tendencia, ciclos) que indicarían que el modelo no extrajo toda la información posible.
- b) Los parámetros del modelo ARIMA seleccionado son significativamente diferentes de cero.
- c) Los parámetros del modelo están poco relacionados entre sí.
- d) El grado de ajuste es elevado en comparación al de otros modelos alternativos. La bondad de ajuste puede evaluarse con la desviación estándar residual (DER), criterio de información de Akaike (AIC), y con el criterio bayesiano de Schwarz (SBC), entre otros. A continuación se da una breve explicación de cada uno. (Faraway y Chatfield, 1998).

Desviación estándar residual (DER): Su expresión matemática es

$$\hat{\sigma} = \sqrt{\frac{S}{(T-r)}}$$

donde S es la suma cuadrática de los residuos, T es el número de las observaciones efectivas que se usan en el ajuste del modelo (recordar que se pierden observaciones por diferenciación) y r es el número de parámetros estimados en el modelo, incluyendo la contante. La DER es un criterio de selección de modelo. Un valor pequeño indica una mayor adecuación del modelo.

- Criterio de información de Akaike: El estadístico AIC propuesto por Akaike está dado por:

$$AIC = T \ln \frac{S}{T} + 2r$$

Este criterio permite seleccionar un modelo. Se prefiere el modelo que tenga el menor valor de AIC.

- Criterio bayesiano de Schwarz: La expresión matemática de este estadístico es:

$$SBC = T \ln \frac{S}{T} + r \ln T$$

Este criterio es un método para la selección de un modelo. Se prefiere el modelo que tenga el mínimo valor del estadístico. El SBC penaliza los parámetros adicionales más severamente que el AIC, conduciendo a modelos más simples.

5. Predicción:

La predicción se realiza sobre el modelo ARIMA seleccionado. Se predicen “m” periodos correspondientes a los tamaños del conjunto de predicción, con sus intervalos de confianza. Se calculan los errores de predicción. Es importante juzgar la adecuación del modelo en función de qué tan bien se pronostican los datos no empleados para la estimación del modelo. Para evaluar la predicción, se utilizarán dos tipos de mediciones de error.

- Error absoluto medio porcentual:
(MAPE = mean absolute percentage error)

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{Y_{t+1} - \hat{Y}_{t+1}}{Y_{t+1}} \right|$$

donde Y_{t+1} son los valores reales de la serie que pertenecen al conjunto de predicción $Y(t)$ y \hat{Y}_{t+1} son los valores pronosticados por el modelo ARIMA

- Raíz del error cuadrado medio.
(RMSE = root mean squared error)

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_{t+1} - \hat{Y}_{t+1})^2}$$

Otra forma de evaluar la predicción es a través de la correlación entre los valores observados y los pronosticados por el modelo. Valores altos de esta correlación indican una buena adecuación del modelo. Si la serie pareciera cambiar en el tiempo, pudiera ser necesario recalcular los parámetros, o desarrollar un nuevo modelo.

Capítulo 2

Fundamentos de las redes neuronales artificiales

La inteligencia artificial (IA) es un área del conocimiento compuesta por un conjunto de técnicas que se basan en imitar computacionalmente las distintas habilidades relacionadas a la inteligencia del ser humano, como por ejemplo: reconocimiento de patrones, diagnóstico, clasificación, entre otros. Una de estas técnicas imita, específicamente, el comportamiento de las neuronas en el cerebro humano, por lo cual se le ha denominado redes neuronales artificiales (RNA). Es de interés en este trabajo la aplicación de esta metodología para el pronóstico de llamadas en el “Call Center” de la empresa ABC, como solución a la problemática actual de pronóstico.

En este capítulo se introducirán los conceptos básicos de las RNA, su arquitectura, sus diversas topologías, y el modelo perceptron que ha sido utilizado con fines de pronóstico y que se desarrollará como solución al problema expuesto.

2.1 Panorama histórico

Conseguir, diseñar y construir máquinas capaces de realizar procesos con cierta inteligencia ha sido uno de los principales objetivos y preocupaciones de los científicos a lo largo de la historia. Sin embargo, a pesar de disponer de herramientas y de lenguajes de programación diseñados para el desarrollo de máquinas inteligentes, existe un problema de fondo que limita los resultados: estas máquinas se implementan sobre ordenadores basados en la filosofía de Von Neumann¹, y se apoyan en una descripción secuencial del proceso de tratamiento de la información.

Las primeras explicaciones teóricas sobre el cerebro y el pensamiento fueron dadas por algunos filósofos griegos, como Platón y Aristóteles, quienes fueron apoyados después por Descartes y filósofos empiristas.

Alan Turing, en 1936, fue el primero en estudiar el cerebro como una forma de ver el mundo de la computación; pero quienes primero concibieron algunos fundamentos de la computación neuronal fueron Warren McCulloch y Walter Pitts. Después, otras teorías

¹ La filosofía de Von Neumann se refiere a las arquitecturas de computadoras que utilizan el mismo dispositivo de almacenamiento tanto para las instrucciones como para los datos (a diferencia de la arquitectura Harvard)

iniciales fueron expuestas por Donald Hebb; pero solo hasta 1957 Frank Rosenblat comenzó el desarrollo del Perceptron, la red neuronal más antigua.

Más adelante apareció el modelo ADALINE, desarrollado por Bernard Widrow y Marcial Hoff.

Stephen Grossberg realizó Avalancha en 1967. Hasta 1982 el crecimiento se frenó pero surgieron luego investigaciones sobre redes como la de Marvin Minsky y Seymour Papert, después James Anderson desarrolló el asociador lineal, en Japón Kunihiko Fukushima y Teuvo Kohonen que se centraron en redes neuronales para el reconocimiento de patrones; en USA John Hopfield también realizó importantes investigaciones.

Desde 1985 comenzaron a consolidarse los congresos más importantes como “Neuronal Networks for Computing”, la “Neuronal Information Processing Systems”, entre algunas otras.

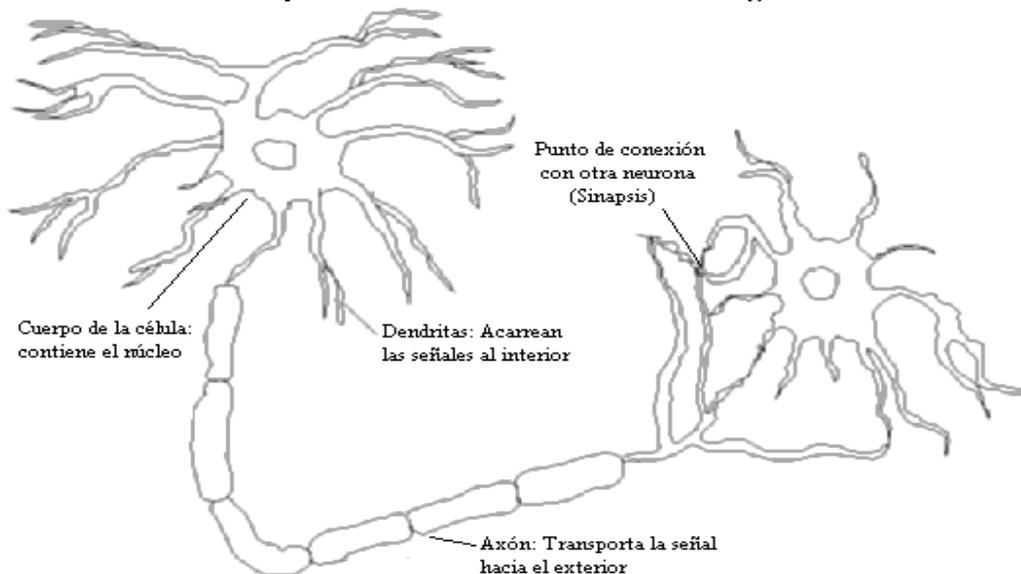
Actualmente, son numerosos los trabajos que se realizan y publican. Revistas como “Neural Networks”, “Transactions on Neural Networks”, entre otras, son las encargadas de la publicación de los últimos avances.

El Departamento de Defensa de los Estados Unidos, la Sociedad Europea de Redes Neuronales, son algunos de los ejemplos del resurgir de la investigación sobre redes neuronales.

2.2 La red neuronal biológica y la artificial

La teoría y modelado de redes neuronales está inspirada en la estructura y funcionamiento de los sistemas nerviosos, donde la neurona es el elemento fundamental.

Figura 8
Esquematización de la neurona biológica



Fuente: Daza, nd

En general, una neurona consta de un cuerpo celular más o menos esférico, de 5 a 10 micras de diámetro, del que salen una rama principal, el axón, y varias ramas más cortas, llamadas dendritas.

Una de las características de las neuronas es su capacidad de comunicarse. En términos generales, las dendritas y el cuerpo celular reciben señales de entrada; el cuerpo celular las combina e integra y emite señales de salida. El axón transmite dichas señales a los terminales axónicos, que distribuyen información o un nuevo conjunto de neuronas. Se calcula que en el cerebro humano existen del orden de 10^{15} conexiones.

Las señales que se utilizan son de dos tipos: eléctrica y química. La señal generada por la neurona y transportada a lo largo del axón es un impulso eléctrico, mientras que la señal que se transmite entre los terminales axónicos de una neurona y las dendritas de la otra es de origen químico.

Para establecer una similitud directa entre la actividad sináptica y la analogía con las redes neuronales artificiales podemos considerar: las señales que llegan a la sinapsis son las entradas a la neurona; éstas son ponderadas (atenuadas o simplificadas) a través de un parámetro, denominado peso asociado a la sinapsis correspondiente. Estas señales de entrada pueden excitar a la neurona (sinapsis con peso positivo) o inhibirla (peso negativo). El efecto es la suma de las entradas ponderadas. Si la suma es igual o mayor que el umbral de la neurona, entonces la neurona se activa (da salida). Esta es una situación de todo o nada; cada neurona se activa o no se activa. La facilidad de transmisión de señales se altera mediante la actividad del sistema nervioso. Las sinapsis son susceptibles a la fatiga, deficiencia de oxígeno y la presencia de anestésicos, entre otros. Esta habilidad de ajustar señales es un mecanismo de aprendizaje.

Las redes neuronales artificiales buscan simular las tres características básicas de una red neuronal biológica (Hilera, 1995):

- a) **Procesamiento paralelo:** capacidad de incorporar y analizar simultáneamente muchos datos
- b) **Memoria distribuida:** cada neurona tiene parte de la información necesaria para obtener una respuesta en la salida
- c) **Adaptabilidad:** habilidad que poseen las neuronas de cambiar sus conexiones, dependiendo de la estimulación y la respuesta deseada

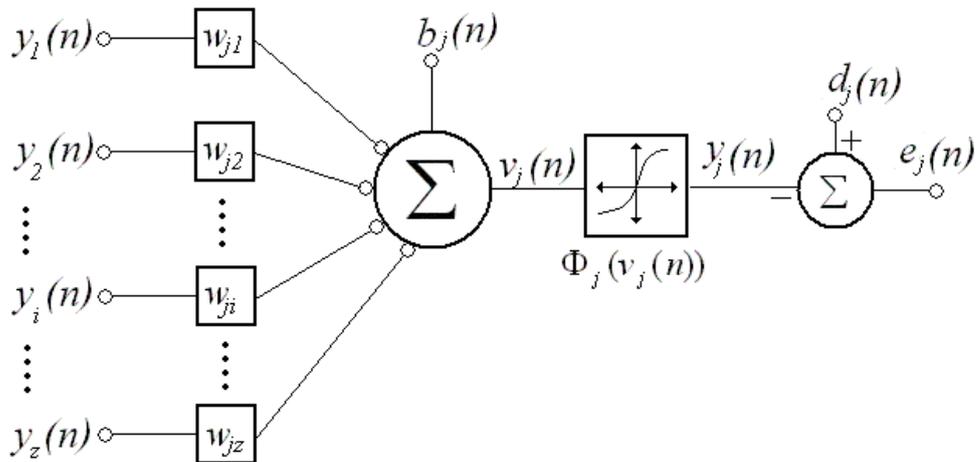
Lo anterior les permite a las redes neuronales artificiales aprender y generalizar a partir de un conjunto de datos de relación matemática desconocida.

Existen muchas definiciones para las RNA. Usando la planteada por Hilera (1995) se establece que una red neuronal artificial es un grafo dirigido que cumple con las siguientes propiedades:

- A cada nodo i se le asocia una variable de estado y_i .
- A cada conexión (i,j) de los nodos i y j se le asocia un peso sináptico w_{ji} donde $w_{ji} \in R$.
- A cada nodo j se le asocia un bias $b_j \in R$.
- Para cada nodo j se define una función de transferencia $\Phi_j(y_i, w_{ji}, b_j)$ que depende de los pesos, conexiones, bias y estados de los nodos i , conectados a él. Esta función proporciona el nuevo estado del nodo.

La figura 9, muestra la estructura de una neurona artificial que pertenece a la capa de salida de una RNA multicapa (Hilera, 1995)

Figura 9
Esquemmatización de una neurona artificial



Donde,

n : denota una iteración cuando se propaga en la red el n-ésimo patrón del conjunto de entrenamiento.

$y_j(n)$: salida de la neurona j

$w_{ji}(n)$: peso de conexión de la entrada i en la neurona j

$b_j(n)$: bias (o sesgo) de la neurona j

$v_j(n)$: valor de activación de la neurona j

Φ_j : función de transferencia de la neurona j

$d_j(n)$: salida esperada de la neurona j al propagar el patrón n

$e_j(n)$: error en la salida de la neurona j al propagar el n-ésimo patrón

En el modelo de una neurona pueden identificarse los siguientes elementos:

- Entradas o nodos de entrada - $y_i(n)$: son escalares que se le proporcionan a la red, de acuerdo al problema en estudio.
- Salidas o nodos de salida - $y_j(n)$: son los valores que arroja la red como resultado del aprendizaje.
- Conjunto de pesos sináptico (pesos) - $w_{ji}(n)$: son valores numéricos que expresa la importancia de la entrada correspondiente.
- Suma de entradas ponderadas: aquí se realiza la combinación lineal o suma de todas las entradas multiplicadas por sus correspondientes pesos.
- Función de activación: es una función, que puede ser lineal o no lineal, que limita el rango de la salida de la neurona. Se explicará en mayor detalle más adelante.

- Sesgo - $b_j(n)$: es un valor formado por una entrada fija e igual a 1, multiplicado por el peso w_{0k} .

Un algoritmo de entrenamiento de una RNA modifica el valor de los pesos de acuerdo con la expresión (1):

$$w_{ji}(n+1) = w_{ji}(n) + \Delta w_{ji}(n) \quad (1)$$

Donde Δw_{ji} es el valor que se obtiene por medio de una regla que define el algoritmo de entrenamiento. Si $E(n)$ es el error de la red cuando se propaga el n -ésimo patrón de entrenamiento se tienen dos posibilidades de medir el desempeño de la red.

$$E(n) = \frac{1}{2N} \sum_z \sum_j (e_j(n))^2 \quad (2)$$

$$E(n) = \frac{1}{2} \sum_j e_j(n)^2 \quad (3)$$

La expresión (2) define el error global en la RNA, sobre todo el conjunto de aprendizaje o sobre los N patrones, mientras que la expresión (3) define el error medio cuadrático (MSE) instantáneo o de un patrón y es una aproximación a la ecuación (2) con la ventaja de requerir un menor esfuerzo computacional; la segunda expresión se utiliza en el presente trabajo.

El algoritmo de propagación hacia atrás (BP) modifica $w_{ji}(n)$ respecto a $E(n)$ usando la información de primer orden (3). Es decir:

$$\Delta w_{ji}(n) = \frac{\partial E(n)}{\partial w_{ji}(n)} \quad (4)$$

Utilizando (3), el algoritmo de propagación hacia atrás establece que (*Haykin (1999)*):

$$\Delta w_{ji}(n) = -\delta_j(n) y_i(n) \quad (5)$$

Siendo $\delta_j(n)$ el gradiente local. Si j es una neurona de la capa de salida

$$\delta_j(n) = e_j(n) \Phi_j'(v_j(n)) \quad (6)$$

Si la neurona j pertenece a una capa oculta

$$\delta_j(n) = \Phi_j'(v_j(n)) \sum_k \delta_k(n) w_{kj}(n) \quad (7)$$

Por lo tanto algoritmo BP modifica iterativamente los pesos con las ecuaciones (5) y (6) hasta que (2) o (3) alcance un valor de tolerancia, lo cual significa que la red ha aprendido el conjunto de entrenamiento.

2.3 Funciones de activación

La selección de la función de activación (FA) depende del criterio del investigador y del problema en estudio. En muchas ocasiones se selecciona por ensayo y error. Existen diversos tipos de FA. Entre los más utilizados están:

Función paso: la salida de este tipo de FA puede ser 0 ó 1, dependiendo si el parámetro de la función es positivo o negativo. Se usa para problemas de clasificación.

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

Función lineal: la entrada de a FA es igual a la salida. Se usa en diversos tipos de redes, con frecuencia, en la capa de salida.

$$f(v) = v$$

Función rampa: su salida está entre -1 y 1.

$$f(v) = \begin{cases} -1 & \text{si } v \leq -1 \\ v & \text{si } -1 < v < 1 \\ 1 & \text{si } v \geq 1 \end{cases}$$

Función logística o sigmoidea: su salida comprende valores entre 0 y 1. Es la FA más usada en redes neuronales y se recomienda para problemas de predicción.

$$f(v) = \frac{1}{1 + e^{-v}}$$

Función tangente hiperbólica: es semejante a la función logística, pero su salida está entre -1 y 1. Se utiliza con frecuencia en redes multicapas.

$$f(v) = \text{TanH}(v)$$

Función gaussiana: Su rango está entre 0 y 1. Se utiliza en redes neuronales de función de base radial, las cuales pueden aplicarse a problemas de predicción.

$$f(v) = e^{-\frac{v^2}{\sigma^2}}$$

2.4 Características de operación de la RNA

La RNA puede tener factores de peso fijos o adaptables: las que tienen pesos adaptables emplean leyes de aprendizaje para ajustar el valor de la fuerza de una interconexión con otras neuronas. Si las neuronas utilizan pesos fijos, entonces su tarea deberá estar previamente definida. Los pesos serán determinados a partir de una descripción completa del problema. Por otra parte, los pesos adaptables son esenciales si no se conoce previamente cuál deberá de ser su valor correcto.

Dos tipos de aprendizaje: existen dos tipos de aprendizaje: supervisado y no supervisado. El primero ocurre cuando se le proporciona a la red tanto la entrada como la salida correcta, y la red ajusta sus pesos tratando de minimizar el error de su salida calculada. Este tipo de entrenamiento se aplica, por ejemplo, en el reconocimiento de patrones. El entrenamiento no supervisado se presenta cuando a la red se le proporcionan únicamente los estímulos, y la red ajusta sus interconexiones basándose únicamente en sus estímulos y la salida de la propia red. Las leyes de aprendizaje determinan cómo la red ajustará sus

pesos utilizando una función de error o algún otro criterio. La ley de aprendizaje adecuada se determina en base a la naturaleza del problema que se intenta resolver.

Dos fases de operación: las RNA adaptables tienen dos fases en su operación:

- a) Entrenamiento de la red. El usuario proporciona a la red un número "adecuado" de estímulos de entrada, y de salida. La red entonces ajusta su pesos de interconexión o sinapsis hasta que la salida de la red está "lo suficientemente cerca" de la salida correcta.
- b) Recuperación de lo aprendido. A la red se le presenta un conjunto de estímulos de entrada y ésta simplemente calcula su salida. Cuando la red emplea entrenamiento no supervisado, algunas veces será necesario que reajuste su sinapsis durante la fase de recuperación.

No son algorítmicas.- La gran diferencia del empleo de las redes neuronales en relación con otras aplicaciones de la computación radica en que no son algorítmicas, es decir, no se programan haciéndoles seguir una secuencia predefinida de instrucciones. Las RNA generan ellas mismas sus propias "reglas", para asociar la respuesta a su entrada; es decir, aprende por ejemplos y de sus propios errores. El conocimiento de una RNA se encuentra en la función de activación utilizada y en los valores de sus pesos.

Asociar y generalizar sin reglas como en el cerebro humano.- Las redes neuronales formadas por los perceptrones se interconectan en forma muy similar a como las neuronas humanas se disponen en la corteza cerebral humana, y lo más importante, son capaces de asociar y generalizar sin reglas. Han sido utilizadas con gran éxito para reconocer retornos de sonar bajo el agua, escritura a mano, voz, topografía de terrenos, controlar brazos de robots, evaluar datos personales, modelar fenómenos cognoscitivos, y, predecir tendencias financieras.

Requieren de algún tipo de patrón.- La clase de problemas que mejor se resuelven con las redes neuronales son los mismos que el ser humano resuelve mejor: asociación, evaluación y reconocimiento de patrones. Las redes neuronales son perfectas para problemas que son muy difíciles de calcular pero que no requieren de respuestas perfectas; sólo respuestas rápidas y buenas. Tal y como acontece con el escenario bursátil en el que se quiere saber ¿compro?, ¿vendo?, ¿mantengo?, o en el reconocimiento cuando se desea saber ¿se parece? ¿es el mismo pero tienen una ligera modificación?

Por otra parte, las redes neuronales son muy malas para cálculos precisos, procesamiento serie, y no son capaces de reconocer nada que no tenga inherentemente algún tipo de patrón. Es por esto que no pueden predecir la lotería, que ya por definición es un proceso al azar.

Existen varias formas de hacer las conexiones en una RNA, así como existen varias formas de conectar neuronas biológicas en el cerebro. Cada tipo sirve para diferentes procesos; el elegir la correcta topología y sus características, es imprescindible para lograr fácilmente la solución del problema (Flores, nd).

2.5 Arquitectura de red

El término arquitectura de red se refiere a la forma o estructura de una RNA. La arquitectura de un modelo de RNA incluye los siguientes componentes: entradas, capas y salidas; así mismo, la interconexión y dirección de la red, que se refiere a la forma en que se interrelacionan las neuronas. El algoritmo de aprendizaje está muy relacionado con la arquitectura de la red.

Componentes:

- **Entradas:** es el canal de alimentación de la red. Se deberán establecer el número de entradas de acuerdo al caso en estudio. Los datos de entradas son numéricos y en muchos casos puede ser conveniente escalarlos y/o pre-procesarlos. El escalamiento se refiere a un cambio de escala, convertirlos a datos entre 0 y 1, entre -1 y 1, o estandarizarlos, de acuerdo al rango de las funciones de activación involucradas. El pre-procesamiento se refiere a la aplicación de algún método estadístico de exploración de datos que permita cualquier transformación que mejora el conjunto original de datos en beneficio de un mejor desempeño de la red.
- **Capas ocultas o intermedias:** se refiere al conjunto de neuronas que se encuentran entre la entrada y salida de la red. Con frecuencia no es suficiente una sola neurona para resolver un problema, sino que se requiere de varias neuronas que operen en paralelo, lo que se denomina capa, e inclusive pudieran ser necesarias varias capas. A la red con una sola capa se le denomina red unicapa, mientras que a la red con dos o más capas se le denomina red multicapa.
El número de capas y de neuronas o nodos que componen cada capa debe especificarse en la arquitectura. El número de capas deberá ser mayor si el problema es no lineal y complejo, pero, en general, un problema podrá representarse bastante bien con una o dos capas. El número de neuronas por capa puede variar entre una capa y otra. Aunque existen algunos criterios para determinar el número de nodos por capa, éstos deberán determinarse por ensayo y error.
 - Un criterio que suele ser muy útil es considerar el promedio entre el número de entradas y salidas como un valor referencial del número de nodos ocultos.
 - *Masters*² propuso la regla de la pirámide geométrica para la creación de la topología. Aquí, la cantidad de neuronas en la capa oculta se calcula como $(n*m)^{1/2}$, donde n es el número de entradas y m el número de salidas que tiene la red.
 - *Ash*³, desarrolló un algoritmo con un criterio dinámico para generar la topología. En su propuesta, un nuevo nodo es generado en una capa oculta cuando el error está por debajo de un valor estimado.
 - *Hirose*⁴, adoptó el método de *Ash* para la creación de un nodo, y lo completó con una técnica para borrar un nodo cuando su valor es cero o muy cercano a cero.

² T. Masters, "Practical Neural Networks recipes in C++". Ed. Academia Press, Inc. 1993. pp. 173-180.

³ T. Ash. "Dynamic node creation in back propagation networks". *Proceedings of Int. Conf. On Neural Networks*. San Diego. 1989. pp. 365-375.

- Yao⁵ y Fiesler⁶ investigaron la aplicación de algoritmos evolutivos para optimizar el número de unidades ocultas y el valor de los pesos en un PMC. En este caso, la programación evolutiva es una técnica estocástica que puede lograr una optimización global.
- **Salidas:** son neuronas o nodos de salida. El número de salidas de la red dependerá del problema en estudio. La salida de la red deberá estar expresada en la misma escala de los datos originales, es decir, si los datos fueron escalados, deberán ser regresados a su escala inicial. Si se realizó un pre-procesamiento deberá hacerse un post-procesamiento.

Interconexión:

Las interconexiones entre capas de las redes pueden clasificarse como:

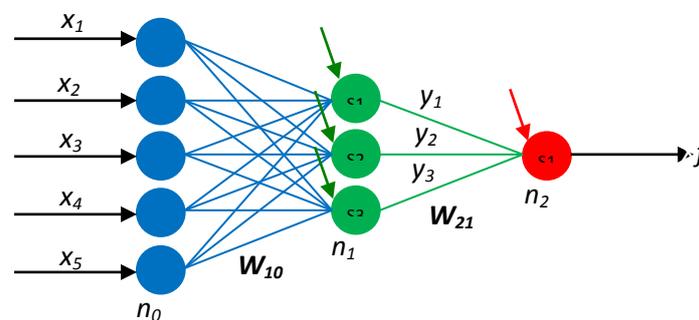
- Totalmente conectadas: la salida de una neurona de la capa i es entrada a todas las neuronas de la capa $i+1$.
- Localmente conectadas: la salida de una neurona de la capa i es entrada a una región de las neuronas de la capa $i+1$

Dirección:

La dirección de la información de las redes pueden clasificarse en:

- Redes de alimentación adelantada: las salidas de las neuronas de una capa sólo se propagan a las neuronas de la capa siguiente. Es decir, la información fluye solamente de la entrada a la salida.
- Redes retroalimentadas: las salidas de las neuronas de una capa pueden ser entradas de las neuronas de las capas anteriores.
- Redes de alimentación lateral: las salidas de las neuronas pueden ser entradas de las neuronas de la misma capa.
- Redes recurrentes: existen lazos cerrados.

Figura 10
Red multicapa, totalmente conectada y de alimentación adelantada - Topología 5: 3: 1



⁴ H. Hirose. "Back-propagation algorithm with varies the number of hidden units". Neural Networks. Vol. 4, 1991. pp. 20-60.

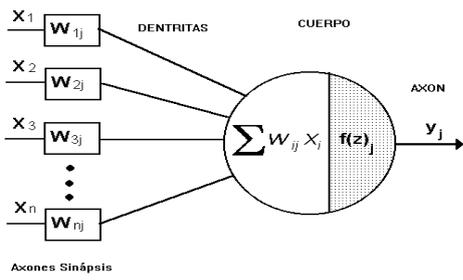
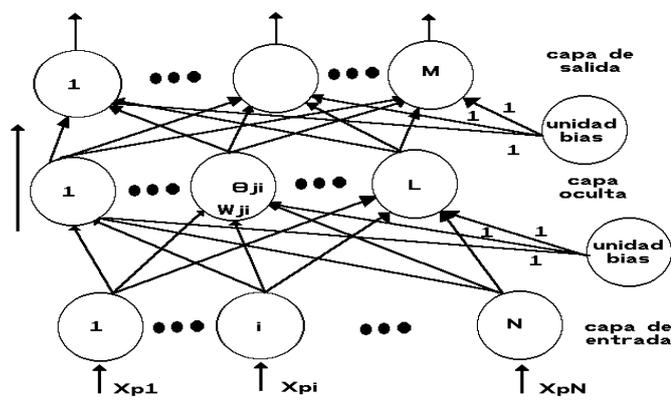
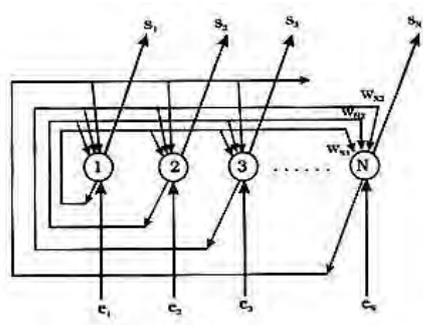
⁵ X. Yao, "Evolving Artificial Neural Networks. School of Computer Science". Proceedings IEEE. Septiembre, 1999.

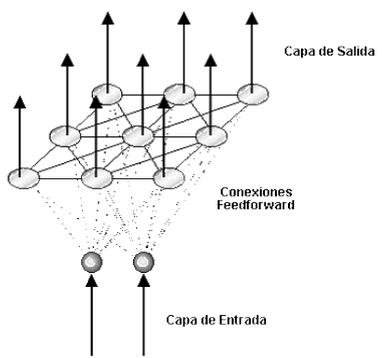
⁶ E. Fiesler. "Comparative Bibliography of Ontogenic Neural Networks". Proceedings of the International Conference on Artificial Neural Networks, ICANN 1994.

2.6 Topologías de las RNA

En la tabla 2 se muestra un resumen de algunas de las topologías de las RNA, en función del número de capas, interconexiones, dirección y función de activación (Wikipedia)

Tabla 2: Resumen de las principales topologías de RNA

Topología	Funcionamiento
<p>Perceptron</p>	<p>Totalmente Conectada / Alimentación Adelantada/Aprendizaje Supervisado El cuerpo de la neurona se representa como un sumador lineal de los estímulos externos z_j, seguida de una función no lineal $y_j = f(z_j)$. La función $f(z_j)$ es llamada la función de activación, y es la función que utiliza la suma de estímulos para determinar la actividad de salida de la neurona.</p>  <p style="text-align: center;"><small>Axones Sinápsis</small></p>
<p>Backpropagation (Red Neuronal de proparacaión hacia atrás)</p>	<p>Totalmente Conectada / Alimentación Adelantada/Aprendizaje Supervisado Red Neuronal formada por la combinación de neuronas perceptron en capas.</p> 
<p>Hopfield</p>	<p>Totalmente Conectada / Retroalimentada y alimentación lateral/ FA: booleana (0 o 1)/Aprendizaje No Supervisado Series de adaptación probabilísticas, recurrentes, funcionalmente entrarían en la categoría de las memorias asociativas, es decir, que aprenden a reconstruir los patrones de entrada que memorizaron durante el entrenamiento.</p> 

Kohonen	<p>Totalmente Conectada / Alimentación Adelantada / FA: Lineales / Aprendizaje No supervisado</p> <p>El objetivo de Kohonen era demostrar que en un estímulo externo (información de entrada) por si solo, suponiendo una estructura propia y una descripción funcional del comportamiento de la red, era suficiente para forzar la formación de mapas (capas bidimensionales). Las unidades de entrada reciben datos continuos normalizados, se normalizan así mismo los pesos de las conexiones con la capa de salida. Tras el aprendizaje de la red, cada patrón de entrada activará una única unidad de salida.</p> 
---------	--

Los criterios de selección de la topología de la RNA están orientados a determinar el número de capas y el número de neuronas en cada capa. Estos criterios son sugeridos luego que el investigador ha decidido qué tipo de RNA va a usar. En este trabajo se utilizará el Perceptrón Multicapa debido a la naturaleza del problema que se va a resolver, como es el pronóstico de una serie de tiempo referida a la demanda de llamadas que recibirá el “Call Center”, y lo que se busca es que la red neuronal encuentre un patrón de comportamiento para luego su posterior predicción aún con patrones no vistos en la etapa de entrenamiento.

Por lo anterior, en este trabajo se planteará un diseño de experimentos para analizar los diferentes comportamientos de la solución utilizando diferentes topologías del Perceptrón Multicapa, mediante una hoja de cálculo de Excel y la herramienta de computación SOLVER.

2.7 Metodología para predicción de un modelo univariable (series de tiempo) con redes neuronales artificiales

A continuación se especifican los pasos que se sigue para construir, entrenar y probar una red neuronal para predecir valores futuros de una serie de tiempo, basada únicamente en sus valores pasados.

1. **Escalamiento de los datos:** transformar los datos a valores comprendidos entre 0 y 1, utilizando la siguiente fórmula, o utilizando alguna otra fórmula para la normalización de los datos.

$$z_t = \frac{y_t - Min}{Max - Min}$$

donde:

y_t : son los valores originales de la serie de tiempo

Min y Max : valor mínimo y máximo de la serie de tiempo.

z_t : serie de tiempo transformada en valores entre 0 y 1

2. **Patrones de entrenamiento y prueba:**

Los valores de la serie de tiempo se dividen en dos conjuntos de datos: patrones de entrenamiento, formado por el 80% de los datos y utilizados en el entrenamiento de la RNA; y patrones de prueba, formado por el 20% restante, utilizado para evaluar la capacidad de generalización o predicción de la red.

3. **Topología de la RNA**

- Dirección de la información: alimentación adelantada.
- Tipo de interconexión: totalmente conectada.
- N° de entradas: p
- N° de capas ocultas: 1
- N° de nodos en la capa oculta: q
- N° de salidas: 1
- Función de activación de los nodos de la capa oculta: tangente hiperbólica.
- Función de activación de la salida: lineal

4. **Determinación de las entradas (p)**

Pueden considerarse varias recomendaciones que ayudarán en la selección de las entradas a la RNA:

- La periodicidad de los datos: como en este trabajo la información de la serie de tiempo es diario, es lógico pensar en considerar 6 retrasos.
- Pruebas de ensayo y error: en algunos casos considerar diferentes retrasos (6, 12, 18, ...) ayudaría a tener mayor precisión para la selección de la mejor topología de red neuronal.

5. **Determinación del número de nodos de la capa oculta(q)**

Una regla ad hoc, descrita anteriormente (*Masters*), que en experimentos previos ha resultado de utilidad, es asumir que el valor inicial del número de nodos de la capa oculta sea igual a la raíz cuadrada del número de entradas(m) y salidas(n), es decir, $\sqrt{m \cdot n}$ (si el valor obtenido es decimal se redondea)

Pueden realizarse pruebas por ensayo y error, agregando más nodos, y comparando los errores de ajuste y predicción (Ej: 3, 4, 5, 6)

6. Algoritmo de entrenamiento: retropropagación

7. Selección de los pesos iniciales:

Escoger los pesos iniciales puede ser crucial y es recomendable probar con diferentes conjuntos de valores iniciales para tratar de obtener buenos resultados. Se selecciona el modelo que obtenga el menor promedio entre la raíz de cuadrados de los errores de ajuste y predicción.

8. Entrenamiento de la RNA seleccionada:

Para entrenar la red es necesario establecer el número máximo de ciclos, el error permitido de convergencia y la tasa de aprendizaje.

Una vez definida la RNA, con su ecuación se generan los valores de la serie de tiempo ajustada o producida por la red, utilizando los patrones de entrenamiento.

9. Predicción:

Usando la ecuación de predicción definida por la RNA se obtiene el valor de predicción $t+1$. Para hacer predicciones más allá del periodo $t+1$, se utiliza ésta como entrada para producir la predicción $t+2$ y así sucesivamente, calculando el error de la predicción.

2.8 Aplicaciones de las RNA

Aplicaciones Generales:

El Perceptrón

El rango de tareas que el Perceptrón puede manejar es mucho mayor que simples decisiones y reconocimiento de patrones. Por ejemplo, se puede entrenar una red para formar el tiempo pasado de los verbos en inglés, leer texto en inglés y manuscrito. El Perceptrón multicapa (MLP) puede ser usado para la predicción de una serie de datos en el tiempo. Este tipo de topología ha tenido éxito en la medición de la demanda de gas y electricidad, además de la predicción de cambios en el valor de los instrumentos financieros, diagnósticos médicos, etc.

NETtalk es un Perceptrón que es capaz de transformar texto en inglés en sonido individual (representaciones fonéticas) y la pronunciación con la utilización de un sintetizador de voz; cuenta con aproximadamente 300 nodos de neuronas (80 de ellos en la capa oculta) y 20,000 conexiones individuales.

Hopfield

Las redes de Hopfield se han aplicado a campos como el reconocimiento de imágenes y optimización de problemas, mostrando gran inmunidad al ruido y robustez. Incluso se han llegado a desarrollar chips específicos para este tipo de redes. El estudio de las representaciones de secuencias temporales es un área de gran interés, con aplicaciones en reconocimiento automático de voces y movimientos.

Hopfield ha mostrado cómo aplicar los mismos principios con funciones de activación continuas como la función sigmoideal, con muy pocas modificaciones.

Aplicaciones médicas

Las redes neuronales proporcionan varias herramientas para ayudar en el diagnóstico de pacientes y la predicción de su evolución. Destacamos a continuación tres tipos de redes que nuestro equipo podría programar.

Diagnóstico

Una red neuronal puede ser entrenada a partir de datos históricos de pacientes y su diagnóstico consolidado. A continuación, los datos de un nuevo paciente alimentan a esta red que proporciona un diagnóstico independiente. Desde el punto de vista de la red, el proceso corresponde a una clasificación.

A diferencia de otros métodos empleados en sistemas expertos, la red se adapta mediante reentrenamiento a cambios de tests, técnicas de diagnosis e incluso evolución de la población. Es una herramienta de ayuda en el proceso de clasificación versátil y dinámica.

Evolución de un paciente

Una red puede ser entrenada con datos que reflejan la evolución de pacientes aquejados de una cierta enfermedad. La red permite realizar una predicción de la

evolución de la misma enfermedad en nuevos pacientes. La red es capaz de adaptarse a cambios en el tratamiento e incluso proporcionar predicciones del resultado de aplicar diferentes tratamientos.

Grupos de riesgo

A menudo se caracterizan ciertos grupos de riesgo (SIDA, cáncer de pecho, etc..) con técnicas estadísticas sencillas. Las redes neuronales permiten realizar clasificaciones mucho más elaboradas (no lineales) con base en ejemplos. La red actúa como una herramienta de "clusterización".

Las redes neuronales proporcionan pues una herramienta cuantitativa sólida para la utilización de bases de datos de pacientes en centros hospitalarios.

Aplicaciones en el campo de las finanzas

Existen una gran cantidad de herramientas para tratar diferentes problemas en el mundo de las finanzas.

Hedging

El problema genérico de valorar opciones ("option pricing") suele utilizar el modelo de "Black&Scholes". La inclusión de volatilidades precisa refinamientos de este modelo. Muchos otros problemas de "risk management" se pueden tratar con técnicas similares.

Predicción de índices bursátiles

El estudio del mercado de valores ha dado lugar a conceptos útiles pero parciales como son las resistencias, soportes, medias móviles, índices de Fibonacci, etc. Una red neuronal puede combinar estos índices en forma no lineal y lograr así una mejor predicción. Una red neuronal puede también ser entrenada con datos históricos de tipo fundamental, técnico y de mercado para realizar una selección de variables representativas y una posterior predicción. La red se basa en datos pasados; pero puede ser reentrenada constantemente para captar nuevos comportamientos del mercado. Constituye una herramienta de predicción dinámica.

Boosters

Existen diferentes técnicas estadísticas para combinar predictores de varios tipos y establecer estrategias de mercado. Las redes neuronales son una de ellas.

Las técnicas estadísticas avanzadas proporcionan herramientas cuantitativas para el tratamiento de grandes bancos de datos y la obtención de predicciones.

Aplicaciones para el sector bancario

La banca viene ya utilizando sistemas expertos combinados con métodos estadísticos para tratar problemas de índole diversa.

Distribución

Una oficina del banco en una zona urbana debe tener en todo momento el mínimo efectivo posible para mantener su funcionamiento normal sin detrimento alguno para el cliente. Se trata pues de un problema de distribución con numerosas restricciones y condicionantes locales; de ahí la necesidad de técnicas estadísticas avanzadas.

Riesgo

Una herramienta para ayudar a un director de zona urbana en el control del riesgo a la hora de conceder un crédito viene proporcionada bien por un sistema experto o por una red neuronal entrenada con casos fallidos.

Promociones (Data Mining)

El nivel de respuesta a una campaña de promoción de un nuevo producto (tarjeta de crédito, crédito personal, etc.) puede optimizarse empleando redes neuronales entrenadas con la respuesta de campañas anteriores.

Aplicaciones para compañías aseguradoras

Las redes neuronales proporcionan varias herramientas para mejorar la comercialización de productos propios de compañías aseguradoras. Se destaca continuación tres tipos de redes que ya se han empleado en este tipo de mercado.

Predicción de bajas

A partir de un histórico (accidentes, edad, nivel socio-económico) sobre anteriores bajas, una red puede ser entrenada para detectar a un mes vista qué clientes tienen una alta probabilidad de dejar la compañía. La red otorga a cada cliente un nivel de *fidelidad*, mejorando la identificación de los clientes más proclives a dejar la entidad en un factor.

Targeted marketing

Una red puede ser entrenada con los resultados de una campaña anterior para mejorar la selección de potenciales clientes en una nueva oferta de producto. A partir de datos históricos, la red ofrece una predicción de *no compra* y, por lo tanto, reduce el conjunto de posibles clientes por contactar. Este método puede llegar a reducir a la mitad el número de clientes contactados, manteniendo en un 90% el nivel de aceptación de una campaña.

Detección de fraude

Una red neuronal puede ser entrenada con datos históricos para detectar fraude no evidente. La red busca correlaciones entre fraudes detectados y datos del cliente, para luego sondear los nuevos casos de fraude.

Aplicaciones para compañías del sector alimentación

Predicción

Una red neuronal puede ser entrenada con datos históricos para hacer predicción de ventas. Es posible, por ejemplo, alimentar a la red con datos de las ventas de tanto un camión como de todo un distribuidor para luego proporcionar una ayuda en la estimación de la predicción de las ventas futuras a corto y largo plazo.

Clasificación

Las redes neuronales pueden aprender criterios de propensión a la compra de un determinado producto. Permiten también crear *cestas de compra*, es decir, predecir qué nuevo producto se debe ofrecer a un determinado cliente dada su respuesta en el pasado.

Capítulo 3

Definición del problema y diseño de los modelos de predicción

En el desarrollo de este capítulo, se expondrá el problema de pronóstico de serie de tiempo, así como el desarrollo de los distintos métodos expuestos en la primera parte del trabajo como herramientas de solución al problema expuesto, por ejemplo: atenuación exponencial de Winter, descomposición clásica de serie de tiempo, metodología de Box-Jenkins (ARIMA), así como también el modelo perceptron multicapa de la red neuronal artificial.

3.1. Definición del problema

ABC es una empresa que ofrece servicios de telecomunicaciones a clientes residenciales, negocios y grandes empresas, a quienes les brinda un número telefónico al que los clientes pueden llamar de forma gratuita por alguna consulta, reclamo o avería que presenten los productos que ABC ofrece.

A inicios del 2011 ABC presentó problemas en sus indicadores de nivel de atención y nivel de servicio en sus centrales de atención telefónica, generando insatisfacción en los clientes a quienes les molestaba que la llamada que realizaban al número de atención al cliente de la empresa, no ingresara o tuviesen que esperar mucho tiempo para ser atendidos (nivel de atención: 88% (meta = 95%), nivel de servicio: 54% (meta = 82%), satisfacción del cliente de 6 (nota máxima = 10 - meta = 9). Esta insatisfacción generó que algunos de sus clientes migraran a una de las empresas de la competencia, buscando no sólo un buen producto (servicio), sino también una mejor atención.

Esta situación preocupó a la gerencia, quienes analizaron la información y determinaron que el principal problema se debía a una mala predicción de la demanda de llamadas, cuyos valores eran mucho menores a las llamadas que realmente recibían, ocasionando un mal dimensionamiento del personal de atención.

Detectado el problema, la gerencia necesitaba un modelo de predicción mejor que el actual, que les ayudara en la predicción de una semana, tiempo suficiente para tomar decisiones a corto plazo y reprogramar a su personal, para la atención de sus clientes.

3.2. Análisis exploratorio inicial de la serie de tiempo

Según lo mencionado en el punto 3.1 del presente capítulo, el problema que se pretende abordar es el diseño de modelos para la demanda de llamadas diarias que recibirá la central de atención telefónica de la empresa “ABC”, con el objetivo de tener el pronóstico diario de una semana de llamadas recibidas. Esto a partir de un histórico de llamadas recibidas diarias (de lunes a sábado) del 5 de enero del 2009 hasta el 26 de febrero del 2011, 672 registros (ver tabla 3). En el anexo C se muestra la base de datos completa.

Tabla 3: Llamadas recibidas diariamente por la central de atención telefónica de la empresa ABC

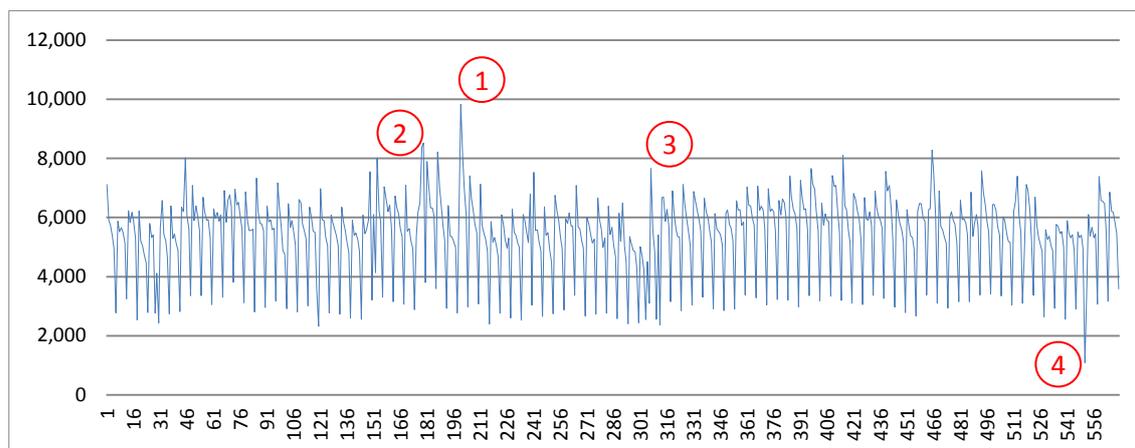
Enero 2009							
SEMANA	L	M	M	J	V	S	D
SEM 1							
SEM 2	7120	5896	5739	5367	4952	2775	0
SEM 3	5870	5519	5661	5495	5155	3254	0
SEM 4	6226	5831	6174	5860	5287	2532	0
SEM 5	6223	5211	5027	4743	4475	2788	
Febrero 2009							
SEM 6	5801	5326	5415	2763	4119	2430	0
SEM 7	5653	6574	5428	5218	4668	2738	0
SEM 8	6387	5289	5444	5094	4906	2822	0
SEM 9	6351	6196	8024	6048	5621	3354	
Marzo 2009							
SEM 10	7083	5894	6395	6067	5566	3353	0
...						

Para efectos del modelamientos, testeo y validación de los modelos, se ha estructurado la información en dos partes:

- Parte 1: del 05/01/2009 al 30/10/2010 (84.8%): Datos iniciales de entrenamiento y de modelación de los métodos predictivos.
- Parte 2: del 01/11/2010 al 26/02/2011 (15.2%): Datos de testeo y validación, para la selección del modelo con menor error obtenido de cada método, y la comparación de los resultados particulares de cada método, respectivamente

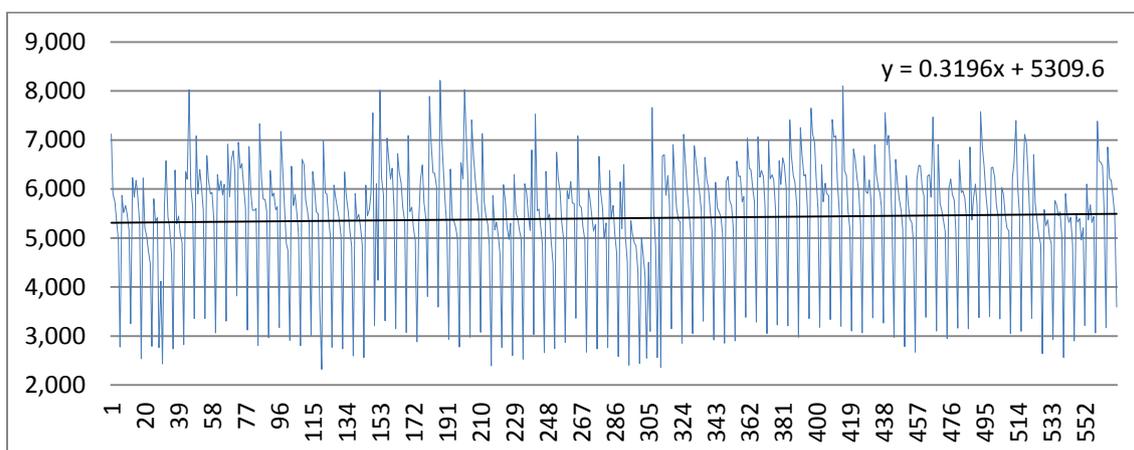
A continuación se grafica la serie de la parte 1 de la serie de tiempo:

Figura 11
Representación gráfica de la parte 1 de la serie de tiempo



En la gráfica se puede apreciar la existencia de datos muy altos y muy bajos (valores atípicos o “outliers”), que podrían afectar las predicciones que se realicen. Se recomienda quitar estos valores atípicos con el objetivo de que éstos no afecten en el modelamiento de las funciones de predicción. Existen distintos métodos de detección y eliminación de “outlier”, pero para este caso en particular se han reemplazado los datos atípicos detectados gráficamente, por el promedio del mismo día de la semana de dos semanas anteriores y dos semanas posteriores. Particularmente se han modificado 6 valores atípicos, (en el anexo D se tiene el cuadro con la base de datos de entrenamiento (parte 1), sin “outliers”).

Figura 12
Representación gráfica de la parte 1 de la serie de tiempo sin “outliers”



Es importante considerar que así como en la parte 1 de los datos de la serie de tiempo se han encontrado valores atípicos, en la parte 2 de los datos se encuentre valores atípicos, los cuales no han sido modificados para poder validar realmente la precisión de los modelos desarrollados.

En la gráfica se puede verificar que existe la forma ascendente de la línea de tendencia, por lo que es claro que la media no es constante, por lo tanto la serie debe diferenciarse. También se observa estacionalidad ($s=6$). La serie no es estacionaria.

Adicionalmente puede obtenerse una idea del comportamiento de la varianza, mediante un diagrama de caja por día de la semana, donde se verifica que existe heterocedasticidad, lo cual puede ser verificado con la prueba de homogeneidad de varianzas de Levene.

Figura 13
Diagrama de caja por día de la semana de la serie de tiempo

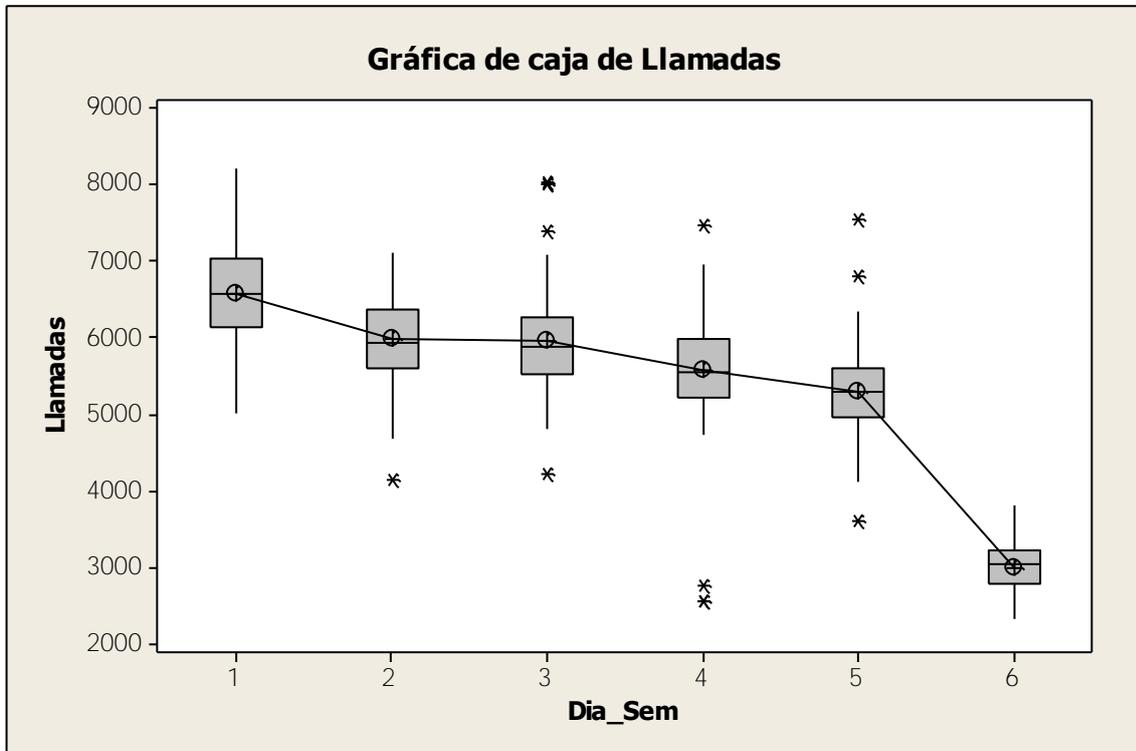


Tabla 4: Resultados de la prueba de homogeneidad de varianza de Levene

Estadístico de Levene	gl1	gl2	Sig.
5,464	5	564	,000

Dada la significación obtenida (0,000) se rechaza la hipótesis nula de homogeneidad de varianzas y se verifica la presencia de heterocedasticidad.

Para la evaluación y comparación de los modelos se evaluarán los pronósticos desarrollados por los distintos modelos, de las llamadas pronosticadas diariamente por los 4 meses restantes (parte 2 de los datos de la serie de tiempo). Estos resultados se realizarán sin variar ningún coeficiente identificado en los modelos, calculando los errores de pronóstico (RSME y MAPE) y el coeficiente de correlación de los pronósticos de cada mes. Para tal efecto, se mostrará al modelo la información del mes actual (M_t), para pronosticar el mes siguiente (\hat{M}_{t+1}). Luego se presentarán los valores reales del mes siguiente (M_{t+1}), para pronosticar el mes próximos \hat{M}_{t+2} y así sucesivamente hasta haber proyectado 4 meses, con la actualización mensual de los datos.

3.3. Diseño del método de atenuación exponencial de Winter

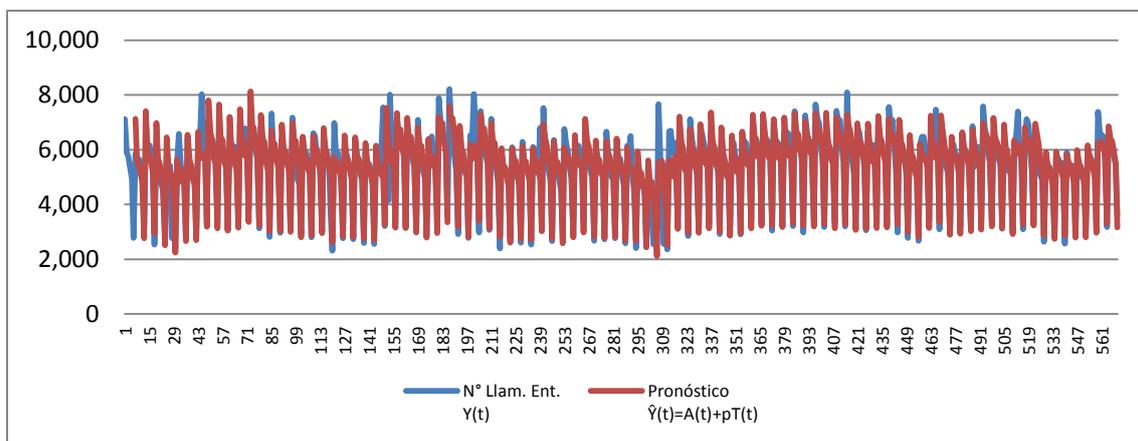
Se ha determinado que la serie de tiempo en estudio tiene un componente tendencial y estación ($s=6$); por tanto es aconsejable utilizar el método de atenuación exponencial de Winter, para la elaboración del pronósticos de los 4 meses siguientes.

Para la elaboración del pronóstico se hace uso de la herramienta Solver, en el cálculo de los parámetros α , β y γ .

Posterior al desarrollo del diseño del método y a la optimización de sus parámetros, se obtienen valores de $\alpha = 0.31819$, $\beta = 0$ y $\gamma = 0.09823$

En la figura 14 puede observarse la curva suavizada sobre los valores reales de la parte 1 de los datos de la serie de tiempo, y en la tabla 5 los valores de los errores de predicción (RMSE y MAPE) y el coeficiente de correlación.

Figura 14
Gráfico de los valores reales de la red y el pronóstico del método de Winter



Adicionalmente se realiza la validación del modelo con la parte 2 de los datos de la serie de tiempo, obteniendo lo siguiente:

Figura 15
Gráfico de los valores reales de la red y el pronóstico del método de Winter de la parte 2 (validación) de los datos.

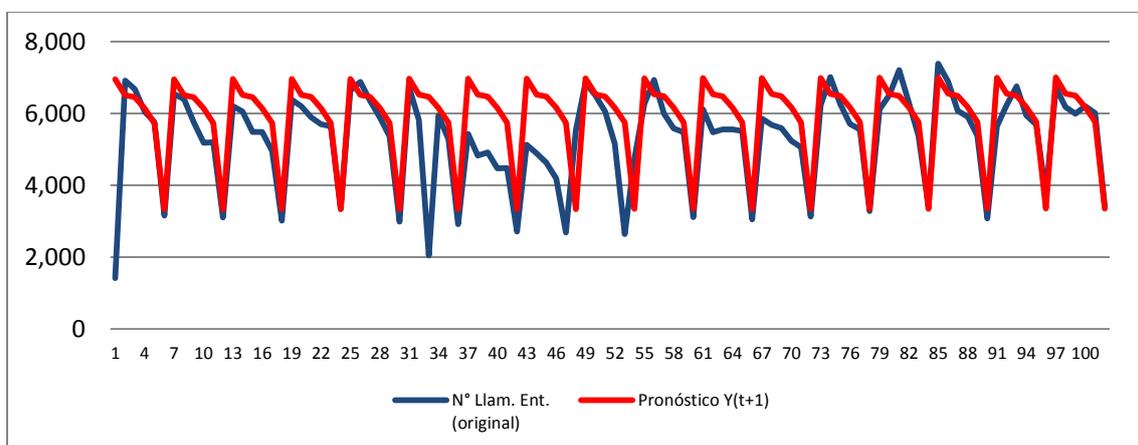


Tabla 5: Análisis de errores de los pronósticos de 4 meses

Indicador	Mes 1	Mes 2	Mes 3	Mes 4	Promedio
RMSE	1,234.12	1,568.89	642.85	452.83	974.67
MAPE	24%	32%	10%	6%	18%
Índice de Correlación	0.6515	0.5038	0.9385	0.9326	0.7566

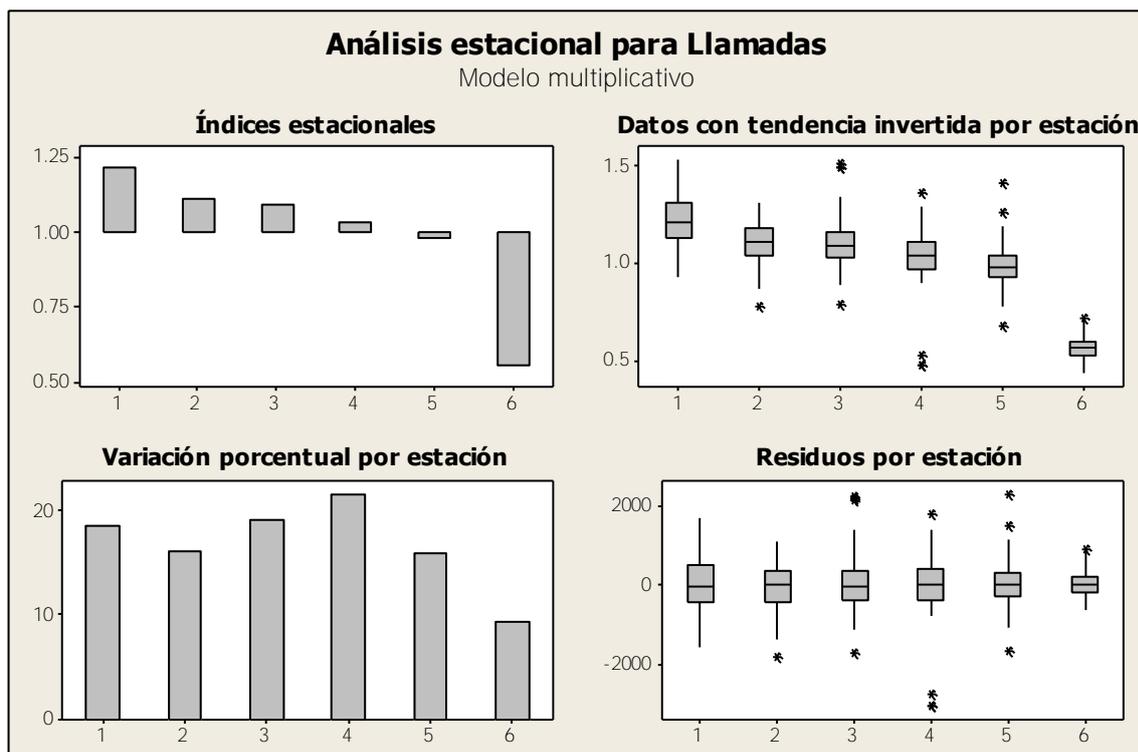
3.4. Diseño de pronóstico por descomposición clásica de serie de tiempo

La descomposición de la serie de tiempo, es una metodología clásica para el tratamiento de este tipo de datos.

Para el caso en estudio se realizará la descomposición de la serie de tiempo utilizando el software Minitab.

En la figura 6 se puede observar el resumen del análisis estacional de la serie. (Ya expuesta de forma preliminar)

Figura 16
Análisis estacional de la serie de tiempo



En el análisis estacional se ha calculado la ecuación de tendencia ajustada, quedando como:

$$Y_t = 5282.1 + 0.417 \times t$$

Y los índices estacionales de cada día de la semana:

Periodo	Índice
1 – Lunes	1.21824
2 – Martes	1.11292
3 – Miércoles	1.09537
4 – Jueves	1.03679
5 – Viernes	0.98300
6 – Sábado	0.55367

Se puede apreciar que al inicio de la semana (lunes) se espera la mayor cantidad de llamadas de la semana, a diferencia de un sábado, donde la cantidad de llamadas esperada es la más baja de la semana.

A continuación se realiza el pronóstico de los 4 meses posteriores quedando como sigue, para la etapa de pronóstico (ver figura 17)

Figura 17
Gráfico de los valores reales de la red y el pronóstico de la parte 2 (validación) de los datos

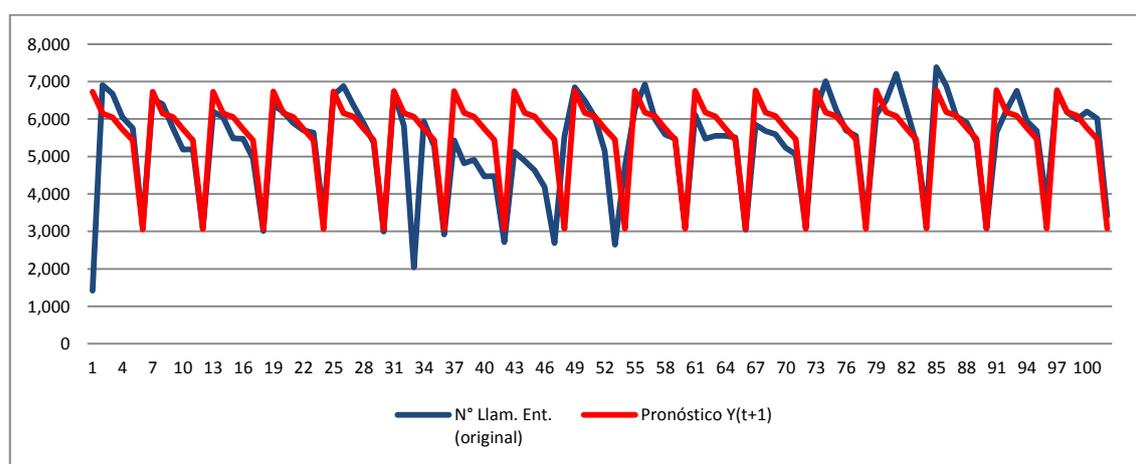


Tabla 7: Análisis de errores de los pronósticos de 4 meses

Indicador	Mes 1	Mes 2	Mes 3	Mes 4	Promedio
RMSE	1,137.99	1,391.75	450.49	497.40	869.41
MAPE	21%	27%	6%	7%	15%
Índice de Correlación	0.6347	0.5106	0.9345	0.9240	0.7510

3.5. Diseño de pronóstico por la metodología de Box-Jenkins

Identificación del modelo

En el análisis inicial de la serie de tiempo se ha identificado que tiene tendencia (1 diferencia regular) y de orden de estacionalidad $s=6$ (1 diferencia estacional), y una varianza que no es contante. Con estos datos se ha identificado que estamos ante un

modelo ARIMA(p,d,q)x(P,D,Q)_s. (No se considerará la constante δ como parámetro de la función).

El modelo ARIMA inicialmente queda de la forma:

$$\text{ARIMA}(p,1,q)\times(P,1,Q)_6$$

$$\Phi_p(B)\phi_P(B^6)(1-B)(1-B^6)Y_t = \theta_q(B)\theta_Q(B^6)a_t$$

A continuación se graficarán los correlogramas simples y parciales del modelo ARIMA(0,1,0)x(0,1,0)₆, para identificar los parámetros autorregresivos y de media móvil.

Figura 18

Correlograma simple con las transformaciones: logaritmo natural y 1^{ra} diferencia en la parte regular y 1^{ra} diferencia en la parte estacional.

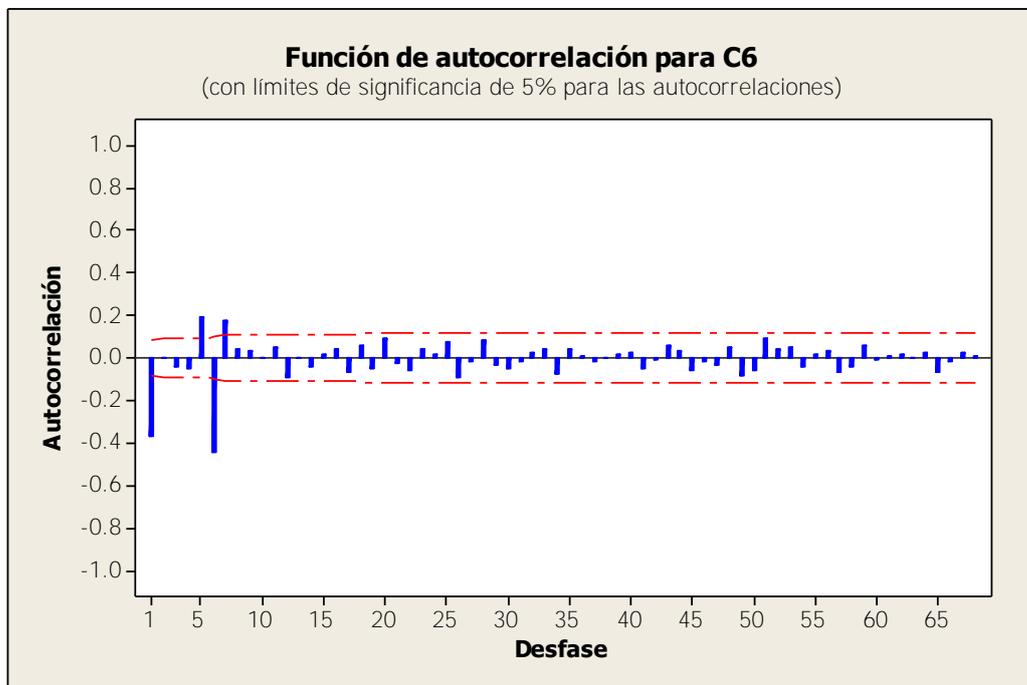
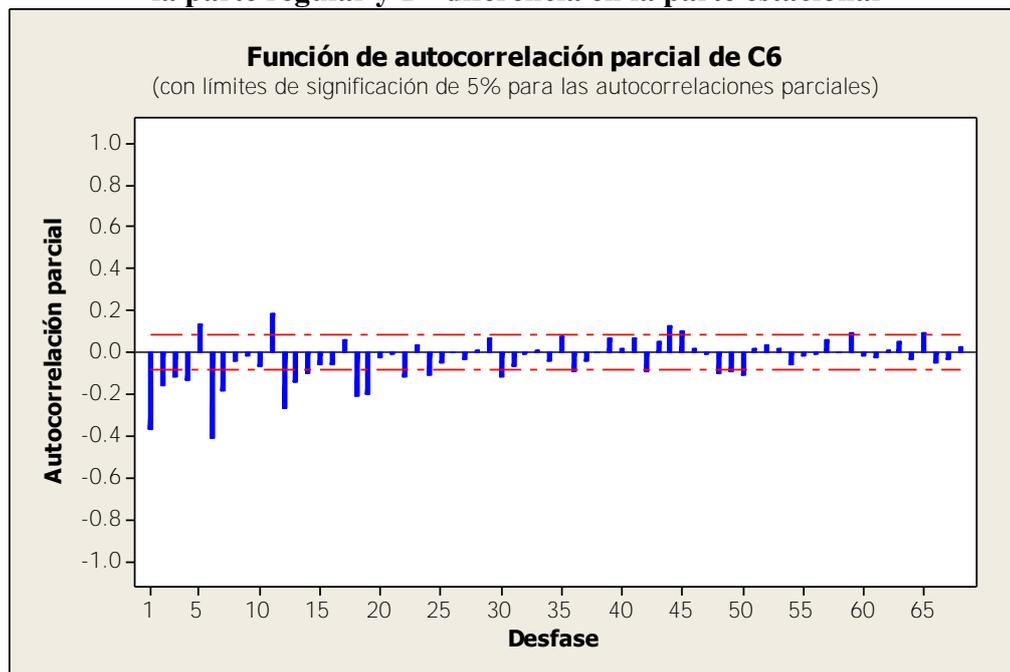


Figura 19
Correlograma parcial con las transformaciones: logaritmo natural y 1^{ra} diferencia en la parte regular y 1^{ra} diferencia en la parte estacional



En la parte regular se observa un patrón MA (Promedio Móvil) de orden 1 (q) y en la parte estacional también un patrón MA (Promedio Móvil) de orden 1(Q)

Se ha identificado el modelo $ARIMA(0,1,1) \times (0,1,1)_6$ inicial. Se deben realizar pruebas con modelos afines.

Estimación de parámetros

Se realizaron numerosas pruebas, siendo los parámetros significativos y obteniéndose resultados satisfactorios con los siguientes modelos:

- A. $ARIMA(0,1,1) \times (0,1,1)_6$
- B. $ARIMA(1,1,0) \times (0,1,1)_6$
- C. $ARIMA(1,1,1) \times (0,1,1)_6$
- D. $ARIMA(2,1,1) \times (0,1,1)_6$**

El modelo de mejor ajuste resultó ser el D, con: $RMSE=490.857$, $MAPE=6.273$ y $R^2=0.855$.

Los parámetros del modelo $ARIMA(2,1,1) \times (0,1,1)_6$ sin constante, para la parte 2 de los datos son:

Parámetro	Estimación	T-Ratio	Approx. Prob.
AR1	0.3743295	7.48938	0.000
AR2	0.1046916	2.21617	0.027
MA1	0.9212069	35.62411	0.000
SMA1	0.9693660	52.59669	0.000

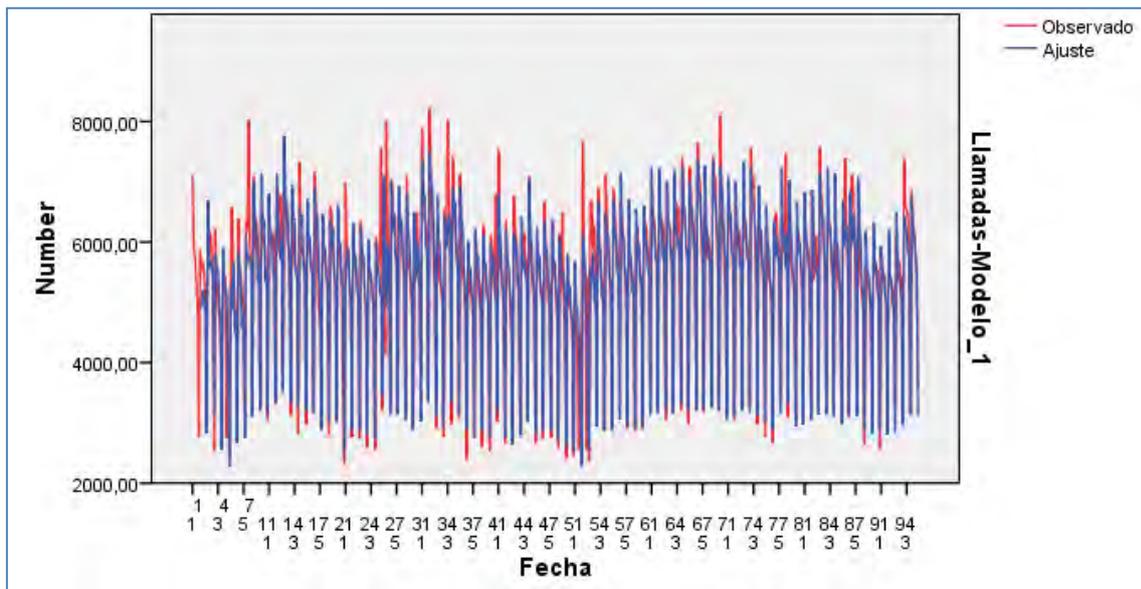
El modelo estimado es:

$$(1 - 0.3743295B - 0.1046916B^2)(1 - B)(1 - B^6)Y_t = (1 - 0.9212069B)(1 - 0.9693660B^6)a_t$$

Adecuación del modelo:

En la siguiente gráfica se representa la serie original y ajustada:

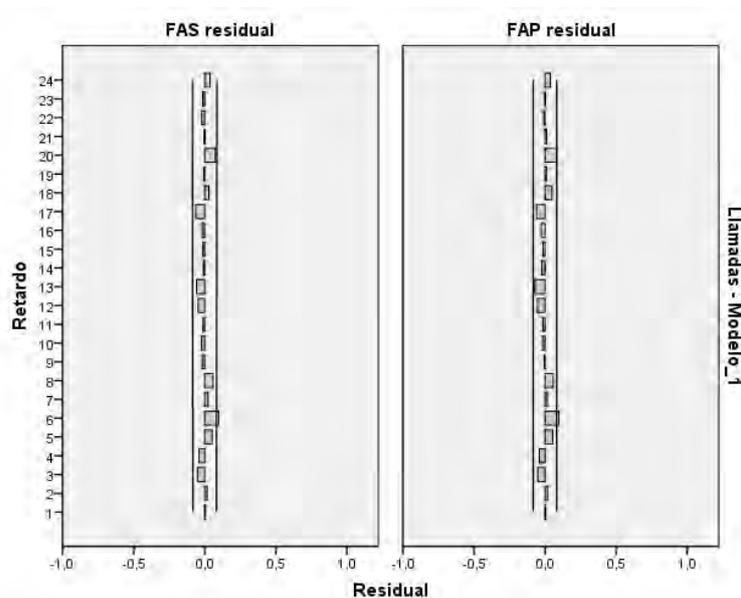
Figura 20
Serie original y ajustada: Parte 1 de los datos



Revisando la gráfica puede decirse que la serie ajustada representa bastante bien a la serie original.

Además puede observarse que en los correlogramas simple y parcial no se observan valores significativos, lo que indica que no queda información sobre la estructura de la serie. (ver figura 21)

Figura 21
Correlogramas simple y parcial de los residuos



Predicción:

Para el desarrollo de las predicciones se hará uso del programa SPSS para el cálculo de los 102 datos restantes (4 meses).

El modelo ARIMA final está representado por la ecuación:

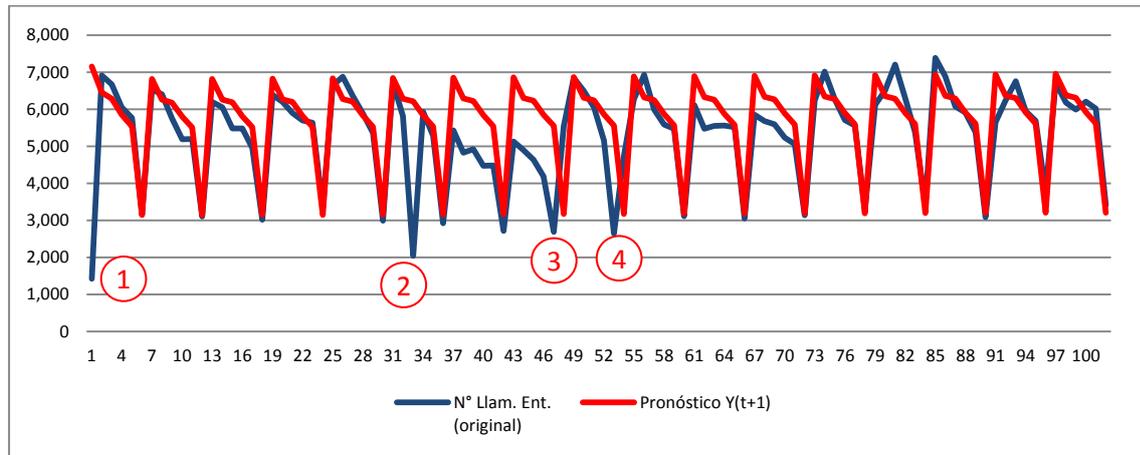
$$(1 - 0.374B - 0.105B^2)(1 - B)(1 - B^6)Y_t = (1 - 0.921B)(1 - 0.969B^6)a_t$$

desarrollando la ecuación,

$$Y_t = 1.374Y_{t-1} - 0.27Y_{t-2} - 0.105Y_{t-3} + Y_{t-6} - 1.374Y_{t-7} + 0.27Y_{t-8} + 0.105Y_{t-9} + a_t - 0.921a_{t-1} - 0.969a_{t-6} - 0.89299a_{t-7}$$

En la figura 22 se representan los valores originales de los 102 datos restantes y las predicciones de los 4 meses siguientes.

Figura 22
Datos originales y proyectados de la serie de tiempo



Analizando la información de los datos reales de la serie de tiempo, se logra identificar que hay ciertos valores con comportamiento irregular, que harán que los errores obtenidos por el método sean mayores. (Así como se vio en el método anterior)

Los errores de pronóstico obtenidos de cada mes se encuentran representados en la tabla 9

Tabla 9: Análisis de errores de los pronósticos de 4 meses

Indicador	Mes 1	Mes 2	Mes 3	Mes 4	Promedio
RMSE	1,221.55	1,449.36	520.62	458.48	912.50
MAPE	22%	29%	7%	6%	0.16
Índice de Correlación	0.6050	0.5085	0.9354	0.9255	0.74

3.6. Diseño de la RNA

Para el desarrollo del diseño y pronóstico de la red neuronal artificial, se utilizará la hoja de cálculo Excel y la herramienta de optimización Solver, para el ajuste de los pesos de las neuronas de la capa de entrada y la capa de salida.

Definición de variables de entrada y salidas:

Las variables de entrada (llamadas recibidas) a la RNA, constituyen los valores de la serie temporal, para lo cual se utilizará solamente la parte 1 de los datos para el aprendizaje o entrenamiento y la Parte 2 (4 meses de pronóstico) para la predicción de la red.

Sin embargo, los datos, tal cual se encuentran, es necesario filtrarlos antes de ser presentados a la red, con el propósito de eliminar ruido y destacar la posible señal; más aún si la función de activación que se va a usar es la función de tangente hiperbólica (Tanh), el recorrido de esta función hace también necesario el filtrado de la información, para no perjudicar la capacidad de aprendizaje de la red, porque los valores muy elevados dificultan el aprendizaje. En consecuencia, en este trabajo, para el filtrado o normalización de la serie se usa la función que sigue:

$$Norm = \frac{y_{t+5} - \frac{\sum_{i=1}^6 y_{t+i}}{6}}{S}, \quad \forall t \geq 0;$$

Donde,

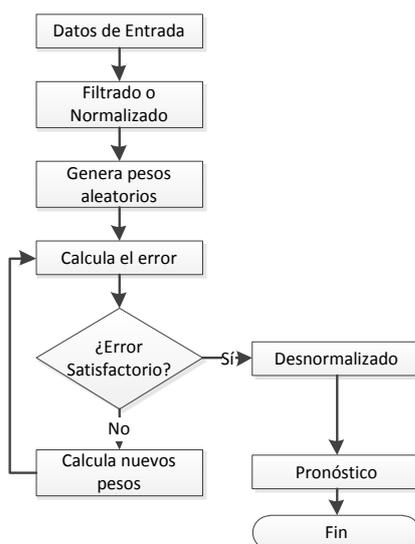
y_t = Cualquier entrada original de la serie temporal.

S = Desviación estándar del conjunto de 5 datos de la serie temporal involucrados en el cálculo

Definidos los datos de entrada normalizados para la red, se debe predecir los datos de salida que constituyen el pronóstico. En nuestro caso los datos, $y_{t-n-1}, \dots, y_{t-2}, y_{t-1}, y_t$, ingresados a la red, deberán predecir el dato y_{t+1} . La cantidad de datos de entrada “n”, a la red es un parámetro por medir mediante el diseño de experimentos, así como también el número de neuronas en la capa oculta. La red que tenga los mejores pronósticos define la topología del Perceptrón Multicapa.

Figura 23

Diagrama de flujo general del proceso de diseño y selección de la topología de la RNA



Selección de la topología de la RNA

Se analizará el comportamiento de las topologías teniendo como factor de variación el número de neuronas en la capa de entrada que se ha definido en base a la estacionalidad $s=6$ y sus múltiplos 12,18 y24; y el número de neuronas en la capa oculta que tiene 4 niveles, lo que hace un total de 16 (4x4), combinaciones o topologías diferentes. De acuerdo a lo recomendado por Masters¹ el número de neuronas en la capa oculta debe ser menor que en la capa de entrada y equivalente a $(n*m)^{1/2}$, donde n es el número de neuronas en la capa de entrada y m el número de salidas que tiene la red, y considerando que en la salida de la red hay una neurona, se ha determinado las siguientes 16 topologías.

¹ T. Masters, “Practical Neural Networks recipes in C++”. Ed. Academia Press, Inc. 1993. pp. 173-180.

Tabla 10: Topologías de RNA para prueba

N° de Topología	N° de neuronas en la capa:		
	Entrada (M)	Ocultas (C)	Salida (N)
1	6	3	1
2	6	4	1
3	6	5	1
4	6	6	1
5	12	3	1
6	12	4	1
7	12	5	1
8	12	6	1
9	18	3	1
10	18	4	1
11	18	5	1
12	18	6	1
13	24	3	1
14	24	4	1
15	24	5	1
16	24	6	1

Definición del mecanismo de aprendizaje.

Durante el proceso de aprendizaje, los pesos de las conexiones de la red sufren modificaciones; por tanto se puede afirmar que este proceso ha terminado (la red ha aprendido) cuando los valores de los pesos permanecen estables ($dw_{ij}/dt=0$).

El mecanismo de aprendizaje de la red neuronal para un perceptrón multicapa tiene varias opciones. En primer lugar se puede utilizar algoritmos genéticos² para minimizar la función del error. También se puede utilizar Simulated Annealing. Así mismo se puede utilizar la función Solver del aplicativo de Microsoft Excel en el aprendizaje de la RNA. Otro algoritmo conocido es *regla delta generalizada* (ver anexo A), que trata de minimizar una delta o diferencia entre el valor observado y el deseado en la salida de la red, basado en que las neuronas de la red tienen una función de activación sigmoideal que es continua y diferenciable.

Sin embargo, debido a que las capas ocultas utilizan una función tangente hiperbólica y la capa de salida una función lineal, es suficiente utilizar para el aprendizaje de la red el aplicativo de Microsoft Excel para este trabajo, tal como se plantea a continuación después del filtrado de los datos comentados en la definición de los datos de entrada.

Definición de parámetros de las RNA:

En esta parte del trabajo se hará referencia al desarrollo de la red número 8, como se puede apreciar en el tabla 10, para efectos demostrativos; pero este mismo procedimiento se debe hacer para las 16 topologías distintas definidas anteriormente.

² M. Domínguez-Dorado, Todo Programación. Nº 12. Págs. 16-20. Editorial Iberprensa (Madrid). DL M-13679-2004. Septiembre, 2005.. Programación de algoritmos genéticos..

Los parámetros por definir son:

- a) Número total de datos de entrada : 672
- b) Número de datos totales después del filtrado : 666
- c) Número de patrones de entrenamiento : 565
- d) Número de patrones de testeo (1 mes) : 101
- e) Número de neuronas en la capa de entrada : 12
- f) Número de neuronas en la capa oculta : 6
- g) Número de neuronas en la capa de salida : 1
- h) Matriz de pesos entre las capas de la red : 6×12 y 1×7
- i) La función tangencial hiperbólica de activación en cada neurona:

$$f_k(\text{net}_{jk}) = \tanh(\text{net}_{jk})$$

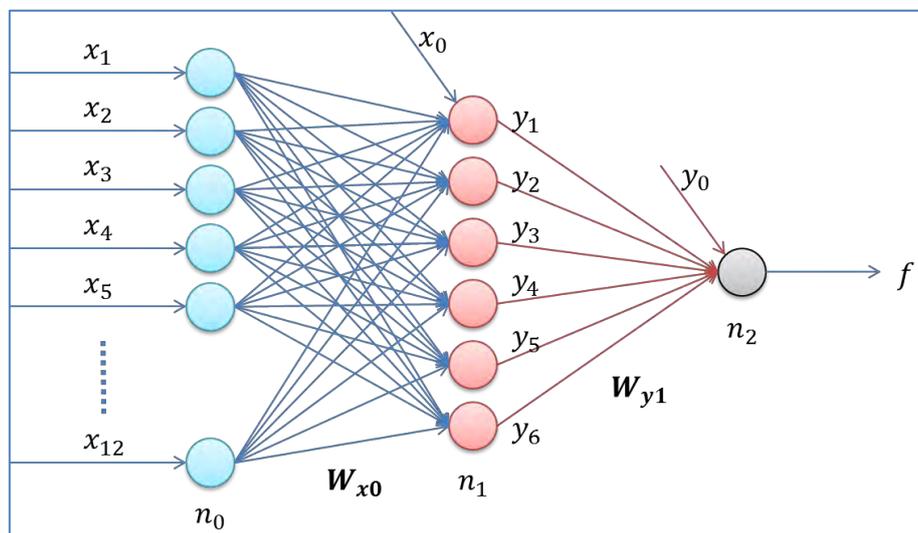
- j) La función lineal en la capa de salida es :

$$f_i(\text{net}_{ji}) = w_{10} \times 0 + y_1 w_{11} + y_2 w_{21} + y_3 w_{31} + y_4 w_{41} + y_5 w_{51} + y_6 w_{61}$$

Red y parámetros

A continuación se presenta la descripción matemática de los parámetros de la red, que se utilizarán en la elaboración de las predicciones en la hoja de cálculo Excel.

Figura 24
Diagrama topológica de la red #8



Donde:

$$W_{x0(6 \times 12)} = \begin{bmatrix} w_{10} & w_{11} & w_{12} & w_{13} & & & w_{1|12} \\ w_{20} & w_{21} & w_{22} & w_{23} & \dots & \dots & w_{2|12} \\ w_{30} & w_{31} & w_{32} & w_{33} & \dots & \dots & w_{3|12} \\ w_{40} & w_{41} & w_{42} & w_{43} & \dots & \dots & w_{4|12} \\ w_{50} & w_{51} & w_{52} & w_{53} & \dots & \dots & w_{5|12} \\ w_{60} & w_{61} & w_{62} & w_{63} & & & w_{6|12} \end{bmatrix}_{n_1 \times (1+n_0)}$$

$$W_{x0_{(6*12)}} = [w_{10} \quad w_{11} \quad w_{12} \quad w_{13} \quad w_{14} \quad w_{15} \quad w_{16}]_{n_2*(1+n_0)}$$

El vector de variables de entrada aumentado es:

$$\bar{x}_o = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ \dots \\ x_{12} \end{bmatrix}_{(n_0+1)*1}$$

La sinapsis de la capa oculta se puede escribir como:

$$S_{n_1*1} = \begin{bmatrix} S_1 \\ S_2 \\ S_3 \\ S_4 \\ S_5 \\ S_6 \end{bmatrix} = W_{x0} * \bar{x}_o$$

Ejemplo:

$$S_1 = w_{10} * 1 + w_{11} * x_1 + w_{12} * x_2 + w_{13} * x_3 + \dots + w_{1|12}x_{12} , \quad S_1 = \sum_{i=0}^{1+n_0} w_{1i} * x_i$$

El vector de variables de entrada aumentado en la capa de salida es:

$$\bar{y} = \begin{bmatrix} 1 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix}_{(n_1+1)*1}$$

La sinapsis de la capa de salida se puede escribir como

$$S_{n_2*1} = [S_1] = W_{y1} * \bar{y}$$

$$S_1 = w_{10} \times 0 + y_1w_{11} + y_2w_{21} + y_3w_{31} + y_4w_{41} + y_5w_{51} + +y_6w_{61}$$

Como la función de activación en la capa de salida es lineal, entonces la salida de la red está dada por:

$$f = w_{10} \times 0 + y_1w_{11} + y_2w_{21} + y_3w_{31} + y_4w_{41} + y_5w_{51} + +y_6w_{61}$$

Implementación de la RNA:

Como ya se ha manifestado, el entrenamiento supervisado está bajo la responsabilidad del aplicativo de Solver de la hoja de cálculo de Microsoft Excel (ver procedimiento de entrenamiento en el anexo B), por lo que para la implementación de la RNA se han considerado 16 hojas de cálculo, una hoja para cada topología, y se utiliza como función objetivo la minimización de la raíz del error cuadrático medio (RMSE, de sus siglas en inglés)². Describiendo la secuencia de operaciones que se indican a continuación, en coherencia con la figura 24, para tal fin se muestra la figura 25 que contiene una porción de la serie temporal y el resultado de la secuencia de operaciones necesarias para resolver mediante la RNA el problema planteado.

En las primeras filas de la figura 25 se han etiquetado las cabeceras que representan el contenido de las operaciones realizadas en su correspondiente columna. Así:

La columna “N° de Llamadas Ent.” de la tabla contiene los valores de la parte 1 de la serie, previo al ajuste de la RNA. La serie se filtró y estandarizó como ya se comentó, obteniéndose la “Serie Filtrada”.

En la parte superior de la hoja de cálculo se tiene otro cuadro con los valores de los pesos de la RNA (ver tabla 11), después del entrenamiento de la RNA.

Tabla 11: Cuadro de los valores de los pesos de la RNA (12-6-1)

N° Neuronas en Capa de Entrada	N° de Neuronas de la Capa Oculta							
	1	2	3	4	5	6		
1	1.923	36.031	17.068	-5.655	10.924	33.099		
2	-0.171	4.587	2.787	-7.031	1.788	-10.638		
3	5.948	15.568	8.267	13.782	-8.314	21.552		
4	26.231	9.272	9.413	9.861	6.975	9.852		
5	15.407	33.030	2.312	8.298	8.611	31.448		
6	30.279	15.379	17.801	-6.785	-1.863	39.412		
7	-13.161	3.158	2.838	0.752	10.875	24.043		
8	11.462	-15.162	-13.415	10.158	12.970	3.564		
9	36.127	-3.825	8.253	17.492	-13.403	21.545		
10	11.289	10.930	27.415	10.853	0.561	-13.247		
11	-10.752	30.411	-0.160	5.934	15.196	-5.900		
12	25.993	27.331	18.220	-10.946	1.190	37.891		
A	B	1	-0.5680	-0.6576	0.7109	0.4183	0.7921	0.8113
N° Neuronas en Capa Salida	N° Neuronas en Capa Oculta							
1	2	3	4	5	6			

En la matriz A se ha definido los pesos de las conexiones de los nodos de la capa de entrada con el primer nodo de la capa oculta. La matriz B contiene los pesos de conexiones entre la capa oculta y la capa de salida.

Para el entrenamiento de la red, se asignó valores aleatorios entre 0 y 10. Obsérvese que los pesos correspondientes a los bías no se incluyen considerándose como ceros.

En la figura 25 pueden observarse los campos “INI1-CO”, “INI2-CO”, hasta “INI6-CO”, donde se ha calculado el input total a las neuronas de la capa oculta (S_{n_1*1}). El input total es la suma ponderada de los valores de la serie.

Posteriormente las salidas de las neuronas “OU1-CO”, “OU2-CO”, hasta “OU6-CO”, se obtienen aplicando la función de activación de tangente hiperbólica a cada valor de los campos “INI1-CO”, “INI2-CO”, hasta “INI6-CO”.

En el campo “Salida de la RNA” de la figura 25, se muestran los resultados de la suma ponderada de los valores de los campos “OU1-CO”, “OU2-CO”, hasta “OU6-CO”, y los pesos de la matriz B, de la tabla 9. Los valores obtenidos en esta operación

corresponderían también a los valores de salida de la RNA, debido a que la función de activación de la neurona de la capa de salida es lineal ($f(v)=v$).

Finalmente en el campo “N° de Llamadas Pronosticadas” se muestran los valores de predicción “des-normalizado”, con los cuales se evaluarán los errores de pronóstico de las distintas redes.

Selección de la RNA con mejores resultados:

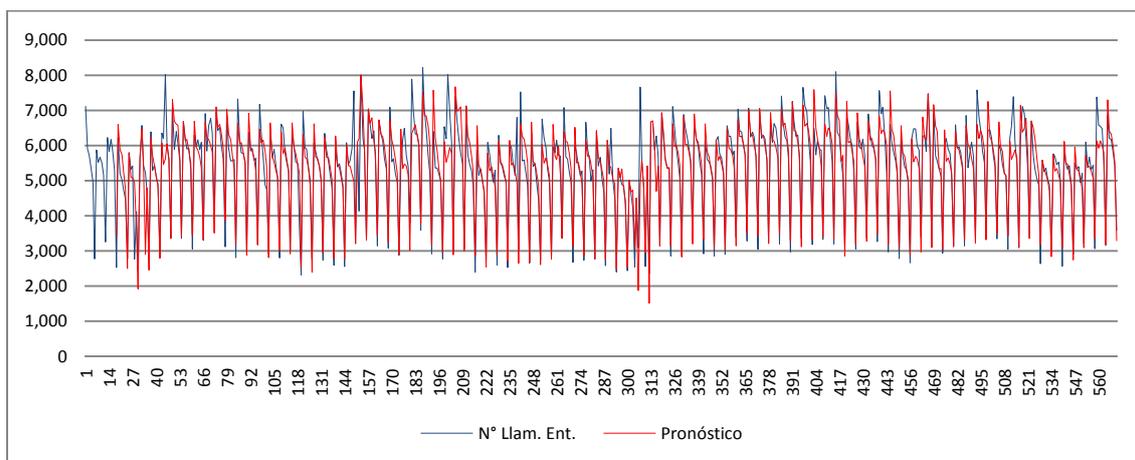
Para la selección de la RNA que ofrece mejores resultados, se ha asignado aleatoriamente 4 juegos de pesos iniciales para la etapa de entrenamiento de cada red. Con el desarrollo de cada red se obtendrán resultados de error de pronósticos (RMSE) distintos.

Tabla 12: Resultados de los valores de RMSE obtenidos en las 4 pruebas de cada topología de RNA.

NE	NO	I	II	III	IV	PROMEDIO
6	3	622.4884	613.8090	642.8797	638.3061	629.3708
6	4	619.3843	617.1598	633.6274	639.4375	627.4023
6	5	661.3389	580.7647	587.3287	619.0156	612.1120
6	6	553.6455	580.2146	580.0114	580.0114	573.4707
12	3	539.0288	618.1248	659.2365	605.7397	605.5325
12	4	632.3474	702.5550	641.6913	626.1236	650.6794
12	5	595.8573	592.1045	621.4697	613.6113	605.7607
12	6	524.9625	537.2294	544.3018	506.5735	528.2668
18	3	607.0468	607.5890	564.5722	608.7694	596.9943
18	4	577.5704	531.3529	555.5711	604.5466	567.2602
18	5	544.8437	521.5116	528.3135	560.3649	538.7584
18	6	592.5786	593.7979	595.2735	580.6663	590.5791
24	3	615.1558	610.7655	588.9915	656.4442	617.8392
24	4	558.4453	585.1066	526.0958	583.5800	563.3069
24	5	602.5519	594.7145	608.6913	632.4796	609.6093
24	6	571.3846	601.6738	641.2776	585.5920	599.9820

En el análisis de varianza del diseño factorial de 4x4, con un coeficiente de variabilidad de 4.32%, no muestran influencia ninguno de los factores en estudio, ni en la interacción en los resultados del RMSE, cuando se analizan todas las topologías de las redes neuronales. Sin embargo, haciendo un análisis más fino, se observa que en comparación de medias de Duncan, la interacción que tiene un mejor comportamiento es E2 – O4, es decir, la red que tiene 12 neuronas en la capa de entrada y 6 neuronas en la capa oculta, siendo estadísticamente mejor a las demás medias.

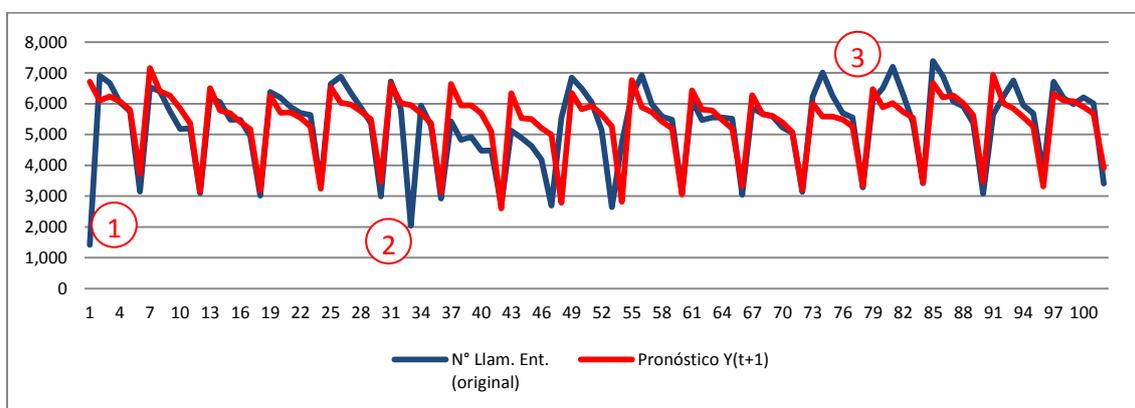
Figura 26
Datos originales y proyectados de la serie de tiempo en la etapa de entrenamiento



Validación del ajuste obtenido

En la figura 27, se representan los valores originales de los 102 datos restantes y las predicciones de los 4 meses siguientes de la red 12-6-1

Figura 27
Datos originales y proyectados de la serie de tiempo en la etapa de validación



Al igual que en las otras metodologías, se logra identificar que hay ciertos valores con comportamiento irregular, que harán que los errores obtenidos por el método sean mayores. Estos valores atípicos tienen que ajustarse cada vez que se realiza la estimación de las llamadas para la semana siguiente.

Puede observarse en la figura 27, que a diferencia de los métodos diseñados anteriormente, el pronóstico de la red intenta seguir las tendencias de corto plazo de las semanas anteriores.

Los errores de pronóstico obtenidos de cada mes se encuentran representados en la tabla 13

Tabla 13: Análisis de errores de los pronósticos de 4 meses

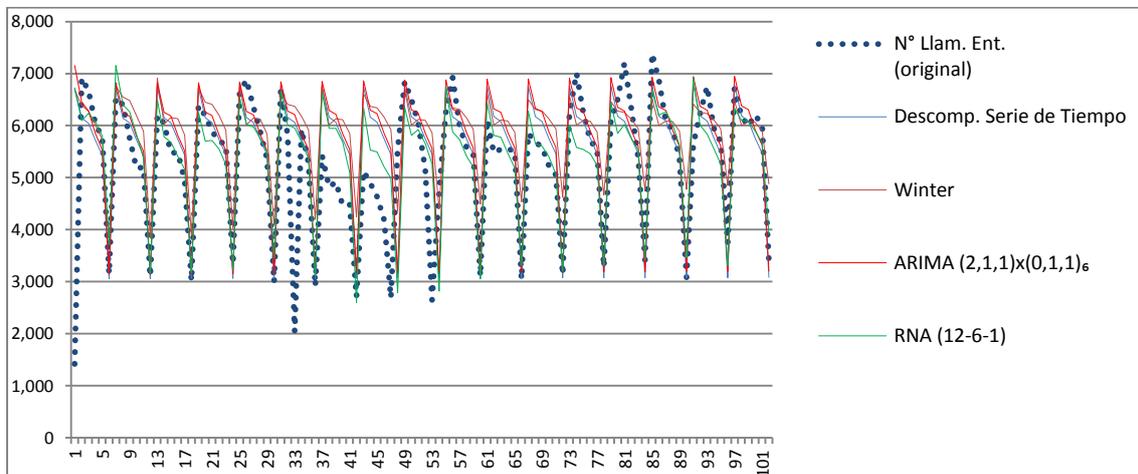
Indicador	Mes 1	Mes 2	Mes 3	Mes 4	Promedio
RMSE	1,136.49	1,295.01	450.19	546.07	856.94
MAPE	21%	25%	5%	8%	15%
Índice de Correlación	0.6189	0.5243	0.9182	0.8941	0.7389

Capítulo 4

Análisis y discusión de resultados

En el capítulo anterior se desarrollaron 4 métodos distintos para la pronóstico de los 4 meses posteriores de la parte 1 de los datos: modelo de descomposición de la serie de tiempo, modelo de ajuste exponencial de Winter, modelo $ARIMA(2,1,1) \times (0,1,1)_6$ y el modelo $RNA(12-6-1)$; consiguiendo para cada uno de ellos, resultados aceptables en los pronósticos.

Figura 28
Comparativo de predicciones de los cuatro modelos desarrollados y los valores reales de la serie de tiempo. - Etapa de validación



En este capítulo se compararán los resultados de los errores de ajuste y pronósticos de los modelos de pronóstico diseñados, así como la selección del modelo que se recomendaría utilizar a la empresa ABC a su problema actual de predicción.

Análisis comparativo en la etapa de entrenamiento (parte 1 de los datos):

En la siguiente tabla se muestra los valores de errores de ajuste de los distintos modelos diseñados:

Tabla 14: Cálculo RMSE, MAPE y de coeficiente de correlación de los modelos de pronóstico.

Método	RMSE	MAPE	Coef. Correlación
Descomposición	574.72	8.17%	0.894
Winter	509.18	8.15%	0.927
ARIMA	490.86	6.27%	0.855
RNA	480.39	6.22%	0.925

Se puede observar que el modelo RNA logra un menor RMSE, alcanzando un valor de 480.39, que es 2% por debajo del Modelo ARIMA que alcanza un valor de RMSE de 490.86, seguido del modelo Winter y el modelo de descomposición. De la misma manera el RNA, alcanza el menor MAPE, comportándose los demás modelos en el mismo orden anterior. Con respecto al coeficiente de correlación, continúa el modelo RNA con un mejor coeficiente, lo que indica que este tipo de modelos se ajustan mejor al comportamiento de las series temporales.

Análisis comparativo en la etapa de validación (parte 2 de los datos):

En la siguiente tabla se muestra los valores de errores de pronósticos de los distintos modelos diseñados:

Por mes:

Tabla 15: Cálculo RMSE, MAPE y de coeficiente de correlación de los modelos de pronóstico en la etapa de validación por mes

Método	RMSE				MAPE				Coef. Correlación			
	M1	M2	M3	M4	M1	M2	M3	M4	M1	M2	M3	M4
Descomposición	1,138	1,392	450	497	21%	27%	6%	7%	0.635	0.511	0.935	0.924
Winter	1,234	1,569	643	453	24%	32%	10%	6%	0.652	0.504	0.938	0.933
ARIMA	1,222	1,449	521	458	22%	29%	7%	6%	0.605	0.508	0.935	0.926
RNA	1,136	1,295	398	553	21%	25%	5%	8%	0.619	0.524	0.934	0.883

En el análisis comparativo de la etapa de validación se verifica que en todos los casos los indicadores se deterioran, debido a la existencia de “outliers” en los datos de validación (principalmente en el mes 1 (M1) y mes 2 (M2)), que de no existir los “outliers” como en los casos de los meses 3 (M3) y 4 (M4), se logra un coeficiente de correlación mayor a 0.92, siendo esto un indicador de la bondad de ajuste de todos los modelos. De igual forma ocurre con los valores de RMSE y MAPE obtenidos en los meses señalados.

Selección del mejor modelo:

La elección del modelo está basada en la facilidad de desarrollo, mayor rapidez y menor error en los promedios de todos los indicadores en estudio. En consecuencia el

modelo de redes neuronales artificiales presenta mejores resultados en la etapa de validación del modelo.

Conclusiones y recomendaciones

Conclusiones

1. Las redes neurales es un método de pronóstico que en este caso en particular resuelve de mejor manera el problema de predicción de las llamadas recibidas en la Central de Atención Telefónica de la empresa “ABC”
2. Dentro de las topologías evaluadas, la topología que mejor responde al pronóstico de la serie temporal, es la que tiene doce neuronas en la capa de entrada, seis neuronas en la capa oculta y una neurona en la capa de salida, con un valor de RMSE de 480.39, un MAPE de 6.28% y un coeficiente de correlación de 0.9285 en la etapa de entrenamiento; y de un RMSE de 856.66, un MAPE de 14.52% y un coeficiente de correlación de 0.7399, en la etapa de validación.
3. El aprendizaje de una RNA tiene la ventaja de que encuentra ella misma sus parámetros y no requiere de estar observando los datos para determinar tendencia, estacionalidad y el error inherente a la aleatoriedad de la información.
4. Una de las características principales observadas en la etapa experimental del diseño de la RNA, es que esta metodología funciona como una caja negra; es decir, la interacción que hace el diseñador con la red se centra en el diseño (inicio) y evaluación (final) de las distintas topologías, sin embargo lo que no se sabemos es cómo la red logra definir las sinapsis (pesos), cómo cada neurona captura y almacena los patrones; lo único que sabemos es que los resultados obtenidos por esta metodología son muy buenos. (Inteligencia artificial)
5. En la comparación y evaluación de los resultados, se determinó que la RNA es el modelo que obtiene mejores resultados de predicción y es la recomendable para su implementación. Es importante considerar que cada vez que el tiempo transcurra y el error de las predicciones siguientes se incrementen, será necesario volver a entrenar a la red para que esta se adapte, con el objetivo de que detecte los nuevos patrones de la serie de tiempo.
6. El modelo de descomposición de series de tiempo en la etapa de entrenamiento no muestra un buen desempeño; esto principalmente porque no detecta en gran medida el componente irregular de la serie; sin embargo, se destaca en la etapa de validación porque proporciona uno de los mejores indicadores de predicción.

7. El método ARIMA es el segundo modelo que presenta mejores resultados tanto en la etapa de entrenamiento, como en la etapa de validación, por lo que sugiere ser también una buena alternativa para la pronóstico de la serie en estudio.
8. Existe una semejanza en la metodología de los modelos ARIMA y de redes neuronales, definida principalmente por la necesidad de diseñar distintos modelos/topologías para evaluar los resultados y seleccionar aquella que es mejor sobre las demás.

Recomendaciones

1. Implementar el modelo de redes neuronales en la empresa “ABC”.
2. Utilizar un software especializado en diseño de redes neuronales que potencien los resultados y faciliten el uso de esta herramienta para el usuario.
3. Ya con un software especializado en redes neuronales, probar otras topologías de red neuronal para la solución del problema de predicción. Por ejemplo: si se quisiera calcular el pronóstico de demanda de llamadas diario por 1 mes, podría diseñarse una red perceptrón multicapa de 90 variables de entrada, 60 neuronas en la capa oculta, y 30 neuronas en la capa de salida, siendo cada valor de salida, un día del mes siguiente pronosticado.
4. Someter a más pruebas y comparar el potencial de solución de los modelos ARIMA y de redes neuronales, así como su facilidad de construcción y diseño.

Bibliografía

Libros:

1. Ash, T. (1989). Dynamic node creation in back propagation networks. Proceedings of Int. Conf. On Neural Networks (pp. 365-375) San Diego.
2. Boweman, B. y O'Connell, R. (1993). Forecasting and time series: an applied approach (3^a ed) California, USA: Duxbury Press.
3. Box, G., Jenkins, G. y Reinsel, G. (1994). Time series analysis, forecasting and control. (3^a ed) New Jersey, USA: Prentice Hall.
4. Chatfield, C. (1978). The analysis of time series: theory and practice Londres: Chapman and Hall.
5. Faraway, J. y Chatfield, C. (1998). Time series forecasting with neural networks: a comparative study using the airline data. Applied Statistic, 47 (2), pp. 231-250.
6. Fiesler, E. (1994). Comparative Bibliography of Ontogenic Neural Networks". Proceedings of the International Conference on Artificial Neural Networks. ICANN
7. Gould, F.J, Eppen G.D, Schmidt C. P. (1992) Investigación de Operaciones en la Ciencia Administrativa México: Hall Hispanoamérica S.A.
8. Gujarati, D. (1997) Econometría (3^a ed) Santa Fé de Bogotá, Colombia: McGraw-Hill.
9. Hanke, E. y Reitsch, G. (1996) Pronósticos en los Negocios (5^a ed) Prentice-Hall.
10. Haykin Symon. (1999) Neural Network. A Comprehensive Foundation (2^a ed): Prentice-Hall.
11. Hilera, J. R., Martínez, V. J. (2000) Redes Neuronales Artificiales. Madrid: Alfaomega.
12. Hilera, J. R. y Martínez V. J. (1995) Redes Neuronales Artificiales. Fundamentos, Modelos y Aplicaciones Madrid: Ra-ma Editorial.

13. Hillier – Lieberman. (1992) Introducción a la Investigación de Operaciones (5ª ed) México: Mc. Graw Hill.
14. Hirose, H. (1991). Back-propagation algorithm with varies the number of hidden units. Neural Networks. Vol. 4 (pp. 20-60)
15. Makridakis Spyros y Wheelwright Steven C. (1992) Manual de Técnicas de pronósticos México: Editorial Limusa S.A.
16. Masters, T. (1993) Practical Neural Networks recipes in C++ (pp. 173-180). Ed. Academia Press, Inc.
17. Maddala, G. (1996) Introducción a la econometría (2ª ed) México: Prentice may Hispanoamericana.
18. Peña, D. (2010) Análisis de Series Temporales :Alianza Editorial.
19. Yao, X. (1999) Evolving Artificial Neural Networks. School of Computer Science: Proceedings IEEE.
20. Wedding II y Cios (1996). Time series forecasting by combining RBF network, certainty factor, and the Box-Jenkins model. Neurocomputing, 10, (pp. 149-168)

Direcciones electrónicas

21. Arellano, M. (2001): Introducción al Análisis Clásico de Series de Tiempo, [en línea] 5campus.com, Estadística <http://www.5campus.com/leccion/seriest>
22. Garcia, C. y Sanchez M. Series temporales univariantes. (nd). Universidad Politécnica de Madrid. http://www.etsii.upm.es/ingor/estadistica/Carol/SeriesUNIV_MIO_MP.pdf
23. Hojas de cálculo para la simulación de redes neuronales artificiales (2002) de <http://www.idescat.cat/sort/questiio/questiio/pdf/26.1.15.Garcia.pdf>
24. Mendoza, L.D. (2011). Uso de la tecnología como herramienta competitiva en la comercialización de la producción artesanal. Universidad estatal de Milagro (p. 4). Ecuador. <http://repositorio.unemi.edu.ec/bitstream/123456789/883/1/Tesis%20Final-%20Lenin%20Mendoza.pdf>
25. Daza, S. P. (nd). Redes neuronales artificiales: fundamentos, modelos y aplicaciones. Universidad militar nueva granada. Colombia. <http://www.monografias.com/trabajos12/redneur/redneur.shtml>
26. Flores, H. (nd). Ventajas de las redes neuronales. <http://hugo-inc.net16.net/RNA/Unidad%201/1.3.html>

Anexos

Anexo A

Algoritmo de retropropagación del error o algoritmo regla delta generalizada.

Según Hilera J. R., Martínez V. J. [5], los pasos y fórmulas a utilizar para aplicar el algoritmo de entrenamiento y cuya demostración se puede observar en el anexo 02.

1. Inicializar los pesos de la red con valores pequeños aleatorios
2. Presentar un patrón de entrada, X_p : $X_{p1}, X_{p2}, \dots, X_{pN}$, y especificar la salida deseada que debe generar la red: d_1, d_2, \dots, d_M (si la red se utiliza como un clasificador, todas las salidas deseadas serán cero, salvo una, que será de la clase a la que pertenece el patrón de entrada).
3. Calcular la salida actual de la red, para ello presentamos las entradas a la red y vamos calculando la salida que presenta cada capa hasta llegar a la capa de salida ésta será la salida de la red y_1, y_2, \dots, y_M . Cómo se indica:
 - a. Se calculan las entradas netas para las neuronas ocultas procedentes de las neuronas de entrada.

Para una neurona “j” oculta:

$$net_{pj}^h = \sum_{i=1}^N w_{ji}^h x_{pi} + \theta_j^h$$

en donde el índice h se refiere a magnitudes de la capa oculta (“hidden”); el subíndice p , al p -ésimo vector de entrenamiento, y j a la j -ésima neurona oculta. El término θ puede ser opcional, pues actúa como una entrada más.

- b. Se calculan las salidas de las neuronas ocultas:

$$y_{pj} = f_j^h(net_{pj}^h)$$

- c. Se realizan los mismos cálculos para obtener las salidas de las neuronas de salida (capa o: output)

$$net_{pk}^o = \sum_{j=1}^L w_{kj}^o y_{pj} + \theta_k^o$$

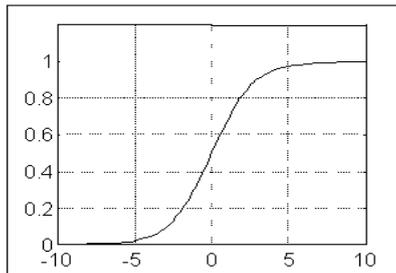
$$y_{pk} = f_k^o(net_{pk}^o)$$

4. Calcular los términos de error para todas las neuronas.

Si la neurona k es una neurona de la capa de salida, el valor de la delta es:

$$\delta_{pk}^o = (d_{pk} - y_{pk}) f_k^{\prime o}(net_{pk}^o)$$

La función f , como se citó anteriormente debe cumplir el requisito de ser derivable, lo que implica la imposibilidad de utilizar una función escalón. En general, disponemos de dos formas de función de salida que nos pueden servir: la función lineal de salida ($f_k(net_{jk}) = net_{jk}$) y la función sigmoideal representada en la Figura y definida por la expresión:



$$f_k(net_{jk}) = \frac{1}{1 + e^{-net_{jk}}}$$

La selección de la función de salida depende de la forma en que se decida representar los datos de salida: si se desea que las neuronas de salida sean binarias, se utiliza la función sigmoideal, puesto que esta función es casi biestable y, además derivable. En otros casos es tan aplicable la función como otra.

Para la función lineal, tenemos: $f_k^{\prime o} = 1$, mientras que la derivada de una función sigmoideal es: $f_k^{\prime o} = f_k^o(1 - f_k^o) = y_{pk}(1 - y_{pk})$, por lo que los términos de error para las neuronas de salida quedan: $\delta_{pk}^o = (d_{pk} - y_{pk})$, para la salida lineal y para la salida sigmoideal el término del error es: $\delta_{pk}^o = (d_{pk} - y_{pk})y_{pk}(1 - y_{pk})$.

Si la neurona j no es de salida, entonces la derivada parcial del error no puede ser evaluada directamente. Por tanto, se obtiene el desarrollo a partir de los valores que son conocidos y otros que pueden ser evaluados.

La expresión obtenida en este caso es: $\delta_{pj}^h = f_j^h (net_{pj}^h) \sum_k \delta_{pk}^o w_{kj}^o$, donde observamos que el error en las capas ocultas depende de todos los términos de error de la capa de salida. De aquí surge el término de propagación hacia atrás. En particular, para la función sigmoideal: $\delta_{pj}^h = x_{pi}(1-x_{pi}) \sum_k \delta_{pk}^o w_{kj}^o$, donde k se refiere a todas las neuronas de la capa superior (siguiente) a la de las neuronas j . Así, el error que se produce en una neurona oculta es proporcional a la suma de los errores conocidos que se producen en las neuronas a las que está conectada la salida de ésta, multiplicando cada uno de ellos por el peso de la conexión. Los umbrales internos de las neuronas se adaptan de forma similar, considerando que están conectados con pesos desde entradas auxiliares de valor constante.

5. Para la actualización de los pesos, utilizamos el algoritmo recursivo, comenzando por las neuronas de salida y trabajando hacia atrás hasta llegar a la capa de entrada, ajustando los pesos de la forma siguiente:

- Para los pesos de las neuronas de la capa de salida:

$$w_{kj}^o(t+1) = w_{kj}^o(t) + \Delta w_{kj}^o(t+1); \quad \Delta w_{kj}^o(t+1) = \alpha \delta_{pk}^o y_{pj}$$

- Para los pesos de las neuronas de la capa oculta:

$$w_{ji}^h(t+1) = w_{ji}^h(t) + \Delta w_{ji}^h(t+1); \quad \Delta w_{ji}^h(t+1) = \alpha \delta_{pj}^h x_{pi}$$

En ambos casos, para acelerar el proceso de aprendizaje, se puede añadir un término momento de valor: $\beta(w_{kj}^o(t) - w_{kj}^o(t-1))$, en el caso de la neurona de salida, y $\beta(w_{ji}^h(t) - w_{ji}^h(t-1))$ cuando se trata de una neurona oculta

6. El proceso se repite hasta que el término de error: $E_p = \frac{1}{2} \sum_{k=1}^M \delta_{pk}^2$, resulta aceptablemente pequeño para cada uno de los patrones aprendidos.

Uno de los problemas que presenta este algoritmo de entrenamiento de redes multicapa es que busca minimizar la función de error, pudiendo caer en un mínimo local o en algún punto estacionario, con lo cual no se llega a encontrar el mínimo global de la función de error. Sin embargo, ha de tenerse en cuenta que no tiene por qué alcanzarse el mínimo global en todas las aplicaciones, sino que puede ser suficiente con un error mínimo preestablecido.

Anexo B

Interfase de Excel y cálculo de los valores óptimos de los pesos de la RNA

En este Anexo se describen los pasos a seguir para el entrenamiento de la RNA utilizando la herramienta SOLVER, de la hoja de cálculo de Microsoft Excel.

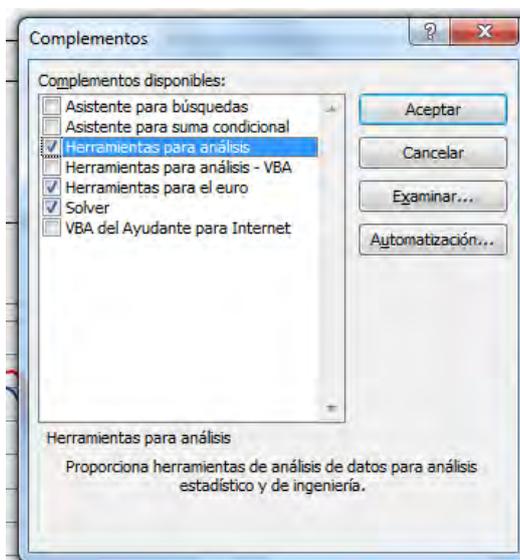
La Función Objetivo a minimizar es una función que calcula la raíz cuadrada de la suma las diferencias obtenidas entre la RNA y los valores deseados elevados al cuadrado (RMSE); esta función depende de los parámetros adaptativos de la RNA conocidos como pesos de las Matrices A y B mostrados en el cuadro 3.8. En el proceso de aprendizaje se lleva a cabo modificando estas celdas de manera que el valor de RMSE sea mínima para el conjunto de patrones de entrenamiento.

Para resolver este problema el aplicativo de Excel que es una Macro conocida como Solver, permite obtener el valor máximo o mínimo de una función, que esté contenida en una celda (RMSE), modificando el contenido o valores de otras celdas (Matriz Ay B)

Para ejecutar «Solver», desde la interface de Microsoft Excel y en la hoja de cálculo, se selecciona en el menú «Datos», dicha opción.



Si no se encuentra en el menú herramientas la opción «Solver», se debe instalar haciendo clic en el ícono «», luego entrar a « Opciones de Excel», «IR» a «Complementos de Excel», y finalmente identificar al función Solver y hacer clic en el botón Aceptar de la ventana emergente para instalar dicho complemento, tal como se ilustra:



Esperar unos segundos hasta que termine de instalarse dicho aplicativo; de manera que en la siguiente vez que se repita el proceso ya se encuentre en las opciones del menú «Datos».

Clickeando sobre «Solver» abrimos el cuadro de diálogo «Parámetros de Solver»



En esta ventana se introduce la celda objetivo (RMSE) y las celdas que se combinan (Matriz A y B), es decir, donde están los parámetros o pesos a ser modificados.

Haciendo clic en «Opciones» del cuadro de diálogo «Parámetros de Solver» se abre el cuadro «Opciones de Solver». En este cuadro podemos modificar las opciones por defecto de los siguientes parámetros:

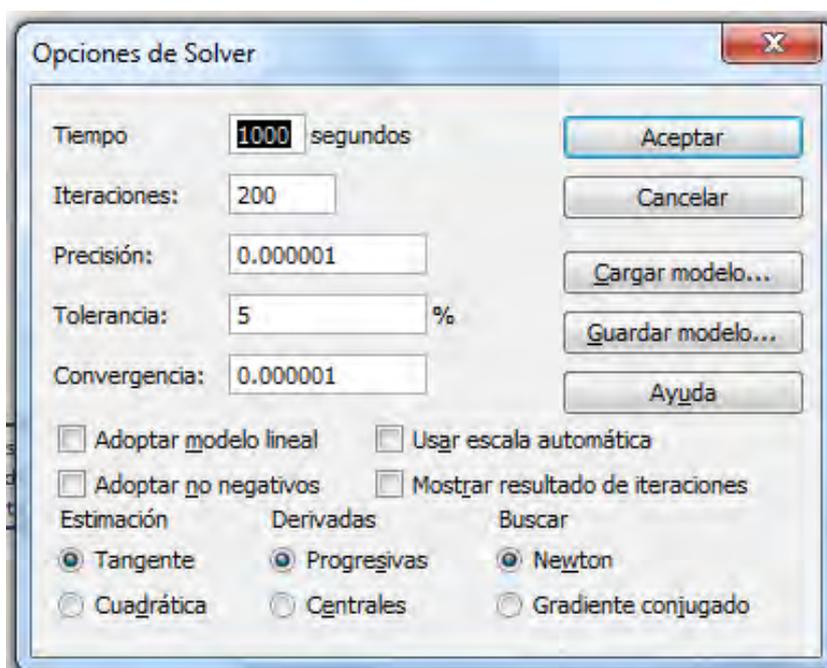
- **Tiempo:** Esta opción establece un límite temporal para encontrar la solución al problema. En este entrenamiento se ha elegido la opción de 1000 segundos.
- **Iteraciones:** Esta opción establece un límite al número de cálculos provisionales. Este valor es el número de ciclos de aprendizaje (épocas) de la red es decir el número de veces que se le presentan los patrones de entrenamiento a la red. Se ha tomado el por defecto 200 épocas.
- **Precisión:** Esta opción controla la precisión de la solución. La precisión debe indicarse como una fracción entre 0 y 1.
- **Tolerancia:** Es el porcentaje máximo de discrepancia entre el valor de la celda objetivo y las restricciones externas si las hubiese. Valores grandes de tolerancia aceleran el proceso de solución. En este trabajo se ha utilizado los valores por defecto 5%.
- **Convergencia:** Si el valor del cambio relativo en la celda objetivo es más pequeño que el valor introducido en esta opción en las cinco últimas iteraciones «Solver» se

detiene. Esta opción solo se aplica a problemas no lineales y debe indicarse mediante una fracción entre 0 y 1.

Además de las opciones anteriores, se ha seleccionado el método Newton para la aproximación de raíces, derivadas progresivas y estimación tangente. La selección de este conjunto de opciones, obedece a que suelen requerir menos tiempo de cálculo. No obstante, en el caso de la selección del método de aproximación de raíces, la elección del método de Newton o del método del gradiente conjugado no es crucial ya que «Solver» es capaz de cambiar automáticamente entre ambos métodos en función de las necesidades de almacenamiento que requiera el problema que se esté resolviendo.

Una vez que se han introducido valores a las opciones anteriores pulsamos en «Aceptar» y volvemos al cuadro «Parámetros de Solver». En este cuadro, pulsamos en «Resolver»

y obtenemos la solución al problema planteado.



Anexo C

Base de datos detallada de las llamadas recibidas por la empresa "ABC"

Llamadas Recibidas por la Central de Atención Telefónica de la Empresa "ABC"													
Semana	L	M	M	J	V	S	Semana	L	M	M	J	V	S
1	7,120	5,896	5,739	5,367	4,952	2,775	41	7,522	5,559	5,587	5,198	4,870	2,659
2	5,870	5,519	5,661	5,495	5,155	3,254	42	6,356	5,404	5,485	4,950	4,478	2,741
3	6,226	5,831	6,174	5,860	5,287	2,532	43	6,749	6,215	5,943	5,324	4,966	2,867
4	6,223	5,211	5,027	4,743	4,475	2,788	44	5,958	5,796	6,152	5,716	5,700	3,365
5	5,801	5,326	5,415	2,763	4,119	2,430	45	7,079	5,671	5,622	5,220	4,962	2,671
6	5,653	6,574	5,428	5,218	4,668	2,738	46	6,000	5,792	5,420	5,138	5,271	2,736
7	6,387	5,289	5,444	5,094	4,906	2,822	47	6,665	5,949	5,603	4,991	5,305	2,766
8	6,351	6,196	8,024	6,048	5,621	3,354	48	6,378	5,416	5,664	5,086	4,686	2,583
9	7,083	5,894	6,395	6,067	5,566	3,353	49	6,145	5,190	6,493	5,037	4,505	2,404
10	6,679	6,188	5,901	5,929	5,345	3,059	50	5,358	5,084	4,895	4,842	4,309	2,438
11	6,295	5,969	6,163	5,877	6,090	3,303	51	5,009	4,674	4,229	2,545	4,500	3,095
12	6,912	5,845	6,581	6,773	6,348	3,816	52	7,661	5,586	4,802	2,560	5,415	2,360
13	6,954	6,426	6,514	6,004	5,619	3,123	53	6,673	6,697	5,869	6,269	5,683	3,148
14	6,864	5,901	5,563	5,562	5,600	2,810	54	6,904	6,142	5,608	5,359	5,335	2,854
15	7,328	6,193	5,794	5,783	5,545	2,963	55	7,112	6,483	5,955	5,504	5,088	3,047
16	6,381	5,849	5,916	5,579	5,637	3,173	56	6,878	6,519	6,236	5,897	5,552	3,304
17	7,170	6,360	5,676	4,883	4,746	2,915	57	6,647	6,181	6,020	5,526	5,169	2,916
18	6,465	5,662	5,891	5,486	5,064	2,803	58	6,134	5,608	5,535	5,418	5,079	2,855
19	6,604	6,497	5,784	5,528	5,286	3,010	59	6,149	6,251	5,765	5,646	5,220	2,901
20	6,353	5,981	5,539	5,493	3,593	2,317	60	6,566	6,252	6,261	5,738	5,847	3,379
21	6,978	5,923	5,895	5,334	5,039	2,772	61	7,033	6,417	6,394	5,999	5,736	3,285
22	6,077	5,832	5,592	5,396	5,039	2,736	62	7,062	6,241	6,378	6,247	5,635	3,046
23	6,346	5,872	5,558	5,179	4,814	2,592	63	6,975	6,206	6,288	6,182	5,542	3,226
24	5,907	5,391	5,479	5,249	4,864	2,559	64	6,573	6,089	6,630	6,495	6,000	3,205
25	6,080	5,451	5,578	5,908	7,551	3,208	65	7,407	6,675	6,266	6,138	5,630	2,978
26	6,103	4,140	8,010	6,227	5,867	3,307	66	7,253	6,693	6,267	6,296	5,579	3,356
27	7,039	6,634	6,202	6,421	5,585	3,148	67	7,651	7,122	6,961	6,223	5,549	3,180
28	6,725	6,345	6,108	5,631	5,334	3,071	68	6,486	5,740	6,117	5,900	5,863	3,333
29	7,090	5,539	5,627	5,213	4,944	2,879	69	7,417	7,055	7,079	6,131	5,529	3,199
30	5,389	6,173	6,487	8,383	8,522	3,803	70	8,096	6,379	6,272	5,556	5,158	3,104
31	7,888	6,973	6,330	6,322	6,049	3,592	71	6,812	6,620	6,215	5,978	5,261	3,059
32	8,213	7,045	6,432	5,700	5,221	2,923	72	6,671	5,956	5,907	6,183	5,706	3,366
33	6,402	5,368	5,351	5,199	4,978	2,777	73	6,900	6,307	6,026	5,921	5,622	3,265
34	6,535	9,824	8,029	6,953	6,047	2,976	74	7,559	6,895	7,086	6,282	5,243	2,971
35	7,409	6,652	6,218	5,770	5,479	3,074	75	6,600	6,088	5,797	5,564	5,187	2,783
36	7,125	5,733	5,466	5,276	4,845	2,394	76	6,266	5,707	5,388	5,315	4,942	2,670
37	5,865	5,158	5,323	5,037	4,681	2,764	77	6,206	6,479	6,472	5,994	5,860	3,381
38	6,087	5,732	5,233	4,968	5,299	2,599	78	6,259	6,292	8,284	7,469	6,022	3,098
39	6,288	5,500	5,409	5,142	4,984	2,529	79	6,903	5,705	5,567	5,340	5,130	2,942
40	6,102	5,909	5,477	5,154	6,796	3,029	80	5,996	6,201	5,895	5,773	5,141	3,158

Llamadas Recibidas por la Central de Atención Telefónica de la Empresa "ABC"						
Semana	L	M	M	J	V	S
81	6,588	5,931	5,962	5,810	5,416	3,148
82	6,853	5,369	5,808	6,098	5,679	3,375
83	7,578	6,815	6,466	5,854	5,509	3,401
84	6,432	6,443	6,234	5,694	5,399	3,348
85	6,024	5,894	5,451	5,206	5,138	3,050
86	6,221	6,537	7,393	6,097	5,494	3,094
87	4,880	7,111	6,918	6,241	5,253	3,365
88	6,698	5,702	5,335	5,062	4,864	2,640
89	5,579	5,254	5,368	5,031	4,878	2,925
90	5,765	5,690	5,454	5,531	4,987	2,563
91	5,900	5,475	5,315	5,423	4,861	2,902
92	5,506	5,325	5,396	4,955	1,094	3,211
93	6,097	5,369	5,672	5,316	5,438	3,062
94	7,384	6,572	6,549	6,469	5,546	3,169
95	6,852	6,208	6,175	5,800	5,451	3,590
96	1,419	6,909	6,672	6,045	5,760	3,151
97	6,535	6,399	5,756	5,183	5,196	3,099
98	6,198	6,048	5,483	5,478	4,950	3,014
99	6,381	6,191	5,892	5,693	5,638	3,338
100	6,634	6,880	6,355	5,880	5,361	2,991
101	6,720	5,818	2,041	5,932	5,279	2,916
102	5,428	4,821	4,916	4,471	4,481	2,709
103	5,124	4,890	4,633	4,184	2,687	5,539
104	6,846	6,498	6,052	5,150	2,643	4,737
105	6,268	6,926	5,987	5,584	5,474	3,108
106	6,113	5,473	5,550	5,554	5,511	3,047
107	5,852	5,678	5,591	5,232	5,053	3,132
108	6,229	7,012	6,256	5,700	5,555	3,278
109	6,116	6,503	7,201	6,303	5,373	3,419
110	7,383	6,883	6,072	5,913	5,379	3,079
111	5,641	6,232	6,752	5,947	5,685	3,738
112	6,716	6,179	5,991	6,203	6,010	3,409

Anexo D

Base de datos detallada de las llamadas recibidas de la empresa "ABC", sin "outliers" del grupo de datos base para el diseño de modelos de pronóstico

Llamadas Recibidas por la Central de Atención Telefónica de la Empresa "ABC", Sin Valores Atípicus													
Semana	L	M	M	J	V	S	Semana	L	M	M	J	V	S
1	7,120	5,896	5,739	5,367	4,952	2,775	41	7,522	5,559	5,587	5,198	4,870	2,659
2	5,870	5,519	5,661	5,495	5,155	3,254	42	6,356	5,404	5,485	4,950	4,478	2,741
3	6,226	5,831	6,174	5,860	5,287	2,532	43	6,749	6,215	5,943	5,324	4,966	2,867
4	6,223	5,211	5,027	4,743	4,475	2,788	44	5,958	5,796	6,152	5,716	5,700	3,365
5	5,801	5,326	5,415	2,763	4,119	2,430	45	7,079	5,671	5,622	5,220	4,962	2,671
6	5,653	6,574	5,428	5,218	4,668	2,738	46	6,000	5,792	5,420	5,138	5,271	2,736
7	6,387	5,289	5,444	5,094	4,906	2,822	47	6,665	5,949	5,603	4,991	5,305	2,766
8	6,351	6,196	8,024	6,048	5,621	3,354	48	6,378	5,416	5,664	5,086	4,686	2,583
9	7,083	5,894	6,395	6,067	5,566	3,353	49	6,145	5,190	6,493	5,037	4,505	2,404
10	6,679	6,188	5,901	5,929	5,345	3,059	50	5,358	5,084	4,895	4,842	4,309	2,438
11	6,295	5,969	6,163	5,877	6,090	3,303	51	5,009	4,674	4,229	2,545	4,500	3,095
12	6,912	5,845	6,581	6,773	6,348	3,816	52	7,661	5,586	4,802	2,560	5,415	2,360
13	6,954	6,426	6,514	6,004	5,619	3,123	53	6,673	6,697	5,869	6,269	5,683	3,148
14	6,864	5,901	5,563	5,562	5,600	2,810	54	6,904	6,142	5,608	5,359	5,335	2,854
15	7,328	6,193	5,794	5,783	5,545	2,963	55	7,112	6,483	5,955	5,504	5,088	3,047
16	6,381	5,849	5,916	5,579	5,637	3,173	56	6,878	6,519	6,236	5,897	5,552	3,304
17	7,170	6,360	5,676	4,883	4,746	2,915	57	6,647	6,181	6,020	5,526	5,169	2,916
18	6,465	5,662	5,891	5,486	5,064	2,803	58	6,134	5,608	5,535	5,418	5,079	2,855
19	6,604	6,497	5,784	5,528	5,286	3,010	59	6,149	6,251	5,765	5,646	5,220	2,901
20	6,353	5,981	5,539	5,493	3,593	2,317	60	6,566	6,252	6,261	5,738	5,847	3,379
21	6,978	5,923	5,895	5,334	5,039	2,772	61	7,033	6,417	6,394	5,999	5,736	3,285
22	6,077	5,832	5,592	5,396	5,039	2,736	62	7,062	6,241	6,378	6,247	5,635	3,046
23	6,346	5,872	5,558	5,179	4,814	2,592	63	6,975	6,206	6,288	6,182	5,542	3,226
24	5,907	5,391	5,479	5,249	4,864	2,559	64	6,573	6,089	6,630	6,495	6,000	3,205
25	6,080	5,451	5,578	5,908	7,551	3,208	65	7,407	6,675	6,266	6,138	5,630	2,978
26	6,103	4,140	8,010	6,227	5,867	3,307	66	7,253	6,693	6,267	6,296	5,579	3,356
27	7,039	6,634	6,202	6,421	5,585	3,148	67	7,651	7,122	6,961	6,223	5,549	3,180
28	6,725	6,345	6,108	5,631	5,334	3,071	68	6,486	5,740	6,117	5,900	5,863	3,333
29	7,090	5,539	5,627	5,213	4,944	2,879	69	7,417	7,055	7,079	6,131	5,529	3,199
30	5,389	6,173	6,487	5,717	5,387	3,803	70	8,096	6,379	6,272	5,556	5,158	3,104
31	7,888	6,973	6,330	6,322	6,049	3,592	71	6,812	6,620	6,215	5,978	5,261	3,059
32	8,213	7,045	6,432	5,700	5,221	2,923	72	6,671	5,956	5,907	6,183	5,706	3,366
33	6,402	5,368	5,351	5,199	4,978	2,777	73	6,900	6,307	6,026	5,921	5,622	3,265
34	6,535	6,200	8,029	6,953	6,047	2,976	74	7,559	6,895	7,086	6,282	5,243	2,971
35	7,409	6,652	6,218	5,770	5,479	3,074	75	6,600	6,088	5,797	5,564	5,187	2,783
36	7,125	5,733	5,466	5,276	4,845	2,394	76	6,266	5,707	5,388	5,315	4,942	2,670
37	5,865	5,158	5,323	5,037	4,681	2,764	77	6,206	6,479	6,472	5,994	5,860	3,381
38	6,087	5,732	5,233	4,968	5,299	2,599	78	6,259	6,292	5,831	7,469	6,022	3,098
39	6,288	5,500	5,409	5,142	4,984	2,529	79	6,903	5,705	5,567	5,340	5,130	2,942
40	6,102	5,909	5,477	5,154	6,796	3,029	80	5,996	6,201	5,895	5,773	5,141	3,158

Llamadas Recibidas por la Central de Atención Telefónica de la Empresa "ABC", Sin Valores Atípicos						
Semana	L	M	M	J	V	S
81	6,588	5,931	5,962	5,810	5,416	3,148
82	6,853	5,369	5,808	6,098	5,679	3,375
83	7,578	6,815	6,466	5,854	5,509	3,401
84	6,432	6,443	6,234	5,694	5,399	3,348
85	6,024	5,894	5,451	5,206	5,138	3,050
86	6,221	6,537	7,393	6,097	5,494	3,094
87	6,131	7,111	6,918	6,241	5,253	3,365
88	6,698	5,702	5,335	5,062	4,864	2,640
89	5,579	5,254	5,368	5,031	4,878	2,925
90	5,765	5,690	5,454	5,531	4,987	2,563
91	5,900	5,475	5,315	5,423	4,861	2,902
92	5,506	5,325	5,396	4,955	5,208	3,211
93	6,097	5,369	5,672	5,316	5,438	3,062
94	7,384	6,572	6,549	6,469	5,546	3,169
95	6,852	6,208	6,175	5,800	5,451	3,590