



## Filtro de Interpolación Basado en una Red Neuronal

### Interpolation Filter Based on a Neural Network

Camila Munar-Torres <sup>1</sup>, Nicolás Segura-Granados  <sup>2</sup>

Para citar este artículo: C. Munar-Torres, N. Segura-Granados, "Interpolation Filter Based on a Neural Network", Revista Vínculos, vol 20, no. 1, p-p 48-62, 2023. <https://doi.org/10.14483/2322939X.17669>

Recibido: 12-12-2021 / Aprobado: 07-02-2023

**Resumen:** El presente trabajo plantea un sistema alternativo de interpolación de varias señales basado en redes neuronales artificiales utilizando MatLab®. Se probaron diversas arquitecturas comenzando con una red neuronal con topología de perceptrón multicapa, con el objetivo de duplicar las muestras de la señal original. El algoritmo de entrenamiento se diseñó a partir de una base de datos construida mediante la adquisición de señales de ECG capturadas en un laboratorio por los autores. Al evaluar el rendimiento de la red en circunstancias más exigentes, como la generación de un mayor número de muestras y respuestas a una mayor variedad de señales, se evidenció una

disminución significativa de su rendimiento, por ello, se reentrenó la misma arquitectura empleando una base de datos pública de mayor tamaño. En búsqueda de alternativas, se desarrolló un modelo fundamentado en la topología de red de base radial, posteriormente, se compararon los resultados obtenidos en ambas topologías, utilizando como señales de entrada un Sin compuesto, SinC, un componente del atractor de Lorenz, un polinomio de alto orden y diferentes señales de ECG. Los resultados obtenidos evidenciaron un mejor desempeño de la topología de red de base radial en comparación con la topología de perceptrón multicapa, sin embargo, ambas

1 Electronic engineerstudent, Universidad Distrital Francisco José de Caldas, Colombia. [cmunart@correo.udistrital.edu.co](mailto:cmunart@correo.udistrital.edu.co).

2 Electronic engineerstudent, Universidad Distrital Francisco José de Caldas, Colombia. [nesegurag@correo.udistrital.edu.co](mailto:nesegurag@correo.udistrital.edu.co).

topologías generaron interpolaciones bastante cercanas a la señal original.

**Palabras clave:** Inteligencia computacional, interpolación, redes neuronales, perceptrón multicapa, redes de base radial, muestreo.

**Abstract:** This paper proposes an alternative system for interpolation of several signals based on artificial neural networks using MatLab®. Different architectures were tested starting with a neural network with multilayer perceptron topology, with the objective of duplicating the original signal samples. The training algorithm was designed from a database constructed by acquiring ECG signals captured in a laboratory by the authors. When evaluating the performance of the network under more demanding circumstances, such as the generation of greater number of samples and responses to a greater variety of signals a significant decrease in its performance was evidenced, therefore, the same architecture was retrained using a large public database. In search of alternatives, a model based on a radial-based network topology was developed, and then results obtained both topologies were compared, using as input signals a composite Sin, SinC, a component of the Lorenz attractor, a high order polynomial and different ECG signals. The results obtained evidenced a better performance of the radial-based network topology, however, both topologies generated interpolations quite close to the original signal.

**Keywords:** Computational intelligence, interpolation, neural network, multilayer perceptron, radial basis neural network, sampling.

## 1. Introducción

La conversión analógica-digital consiste en la transcripción de señales analógicas a digitales con el propósito de facilitar su procesamiento, sin embargo, si no se emplea un número suficiente de niveles de cuantificación en el proceso de digitalización, la relación señal ruido resultante se reducirá con relación a la señal analógica original cuantificada. Este fenómeno es conocido como error de cuantificación [1].

Entorno a este tema, se han desarrollado teorías matemáticas como el Teorema de muestreo de Nyquist-Shannon [2] para garantizar una correcta discretización. Al discretizar una señal se presentan restricciones de hardware con respecto a la frecuencia de muestreo a utilizar, usualmente, al usar componentes económicos que no pueden muestrear a altas tasas, se genera una disminución en el ancho de banda máximo de la señal de entrada dificultando su reconstrucción.

Por ello, se propone un método para realizar dicha reconstrucción de forma digital evitando el uso de hardware muy sofisticados, haciendo uso de las redes

neuronales intentando generar una interpolación de datos de la señal digitalizada, implementando diferentes arquitecturas para identificar las más adecuada para la reconstrucción de señales analógicas digitalizadas.

## 2. Estado del Arte

Para reconstruir la señal a partir de datos de muestreo, generalmente se emplea el método basado en el Teorema de muestreo de Shannon. En este método, el error de reconstrucción aparece cuando la señal no satisface la condición de Nyquist. Este artículo propone un nuevo

método de reconstrucción utilizando un perceptrón lineal y un perceptrón multicapa como filtro FIR. El perceptrón cuyos pesos se obtienen al aprender a adaptar la señal original suprime la diferencia entre la señal reconstruida y la señal original, incluso cuando la condición de Nyquist no se mantiene. Aunque el método propuesto requiere datos de peso, el tamaño total de los datos es mucho más pequeño que el método de muestreo ordinario, ya que el filtro de reconstrucción más adecuado se adapta exclusivamente a los datos de muestreo dados [3]. Por otra parte, en una red neuronal de Newton se declaran varios teoremas que revelan esta red entrenada como el mejor aproximador a cualquier curva o superficie con el menor número de patrones de entrenamiento. El reducir número de muestras de entrenamiento es una ventaja en

este tipo de redes neuronales, ya que reduce sustancialmente el costo computacional y el tiempo de capacitación [4].

Por otro Lado, un nuevo modelo de reconstrucción de señal de audio basado en un algoritmo de estimación no lineal llamado Cartesian Genetic Programming Evolved Artificial Neural Network (CGPANN), permite estimar gráficos no lineales de señales de audio con mayor precisión que métodos convencionales como la interpolación y extrapolación. Este modelo ha sido evaluado en términos de mejora de señal a ruido (SNR) y su capacidad para lidar con disputas de datos. A diferencia de otros algoritmos de reconstrucción convencional, el algoritmo propuesto puede restaurar la señal que está dañada al 50% por ruido.

Un enfoque de vanguardia para la reconstrucción emplea una señal de audio que utiliza el aprendizaje automático. El rendimiento del algoritmo se evalúa midiendo su SNR y la diferencia entre la señal original y la reconstruida en términos de error de porcentaje absoluto medio (MAPE). La mejora de SNR de hasta 20 dB se registra para la estimación de un solo punto con un 25% de muestras faltantes, 19 dB para una estimación multipunto (hasta 5) en la que falta la mitad de los datos y 16 dB para una señal con ruido variable aleatorio [5].

Desde otra perspectiva, un modelo de interpolación de imágenes basado en la red neuronal probabilística (PNN) ajusta automáticamente los parámetros de suavizado para una región de imagen suave

/ borde variado, y tiene en cuenta las características de suavidad (región plana) y nitidez (región borde) en el mismo modelo. Este método utiliza una sola neurona combinada con el entrenamiento de PSO, para la adaptación de nitidez / suavidad. Finalmente se reporta el rendimiento de estos métodos recientemente propuestos en comparación con otras técnicas de interpolación de imágenes [6].

Entre otros avances, un método de interpolación en la red neuronal basado en la descomposición wavelet ha sido utilizado para mejorar la calidad de las imágenes. Mediante la descomposición wavelet, las señales de la imagen se pueden dividir en varias porciones de frecuencias de tiempo. En este trabajo, la señal de descomposición wavelet se usa para entrenar las redes neuronales. Los píxeles de la imagen de baja resolución se utilizan como señal de entrada de la red neuronal para estimar todas las subimágenes wavelet de la imagen de alta resolución correspondiente [7].

### 3. Construcción de la Base de Datos

Para la experimentación se toman en cuenta dos bases de datos construidas independientemente:

**Base de datos A:** Creada mediante 17 señales ECG capturadas experimentalmente por los autores del proyecto, de las cuales una se reserva para validación. Las señales son de dimensiones  $600 * 2$ , Generando una matriz

de entrada  $\mathbf{X}$  de  $6 * n$ , donde  $n$  es el total de datos y  $\mathbf{Y}$  el vector de etiquetas de longitud  $n$ .

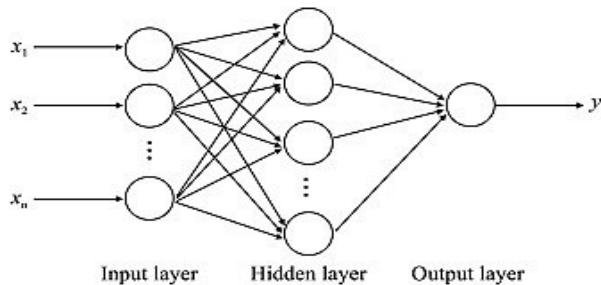
**Base de datos B:** Creada mediante 50 señales ECG extraídas de la base de datos de Physionet. De estas se dejan 40 señales para entrenamiento y 10 para validar, cada una con 10.000 muestras.

Para realizar la interpolación de más muestras, se decide realizar varias reconstrucciones iterativamente, tomando los nuevos puntos generados como entradas para el siguiente ciclo; así, para un vector con  $n$  muestras, con una iteración se genera un nuevo vector con  $2(n - 4) - 1 = 2n - 9$  muestras, y si se ejecuta más veces, con 4 iteraciones se pueden generar hasta 16 muestras entre cada par de puntos, mientras que con 5 iteraciones se generan hasta 32.

### 4. Experimentación con MLP

Para elegir la arquitectura apropiada para este problema, se usó la función de entrenamiento que se encuentra por defecto trainlm en el software MatLab®, y en la capa oculta y la de salida se usó la función de activación tansig. Inicialmente se hizo la experimentación sobre una red neuronal con una capa oculta y la de salida, buscando la interpolación del doble de muestras de la señal original, evaluando los errores de entrenamiento y validación ver Figura 1.

**Figura 1.** Esquema de la topología perceptrón multicapa.



**Fuente:** Implementation of multilayer perceptron (MLP) and radial basis function (RBF) neural networks to predict solution gas-oil ratio of crude oil systems [3].

Con el fin de mejorar la generalización de la red con las señales que no se encuentran en la base de datos, se experimentó con arquitectura de dos capas, utilizando la base de datos A. El entrenamiento se repitió varias veces sobre el mismo número de neuronas hasta conseguir el mínimo error de generalización. Para las pruebas con doble capa oculta, las muestras de la señal utilizadas como entrada a la red neuronal se grafican en color fucsia, en color azul los puntos generados por la red y en rojo la señal original, finalmente la señal obtenida se grafica en negro.

Debido a la alta variabilidad observada en el entrenamiento entre réplica y réplica, se seleccionan como referencia las arquitecturas que mejor se adaptan al entrenamiento, realizando réplicas sucesivas hasta encontrar los mejores resultados como se muestran en la Tabla 1.

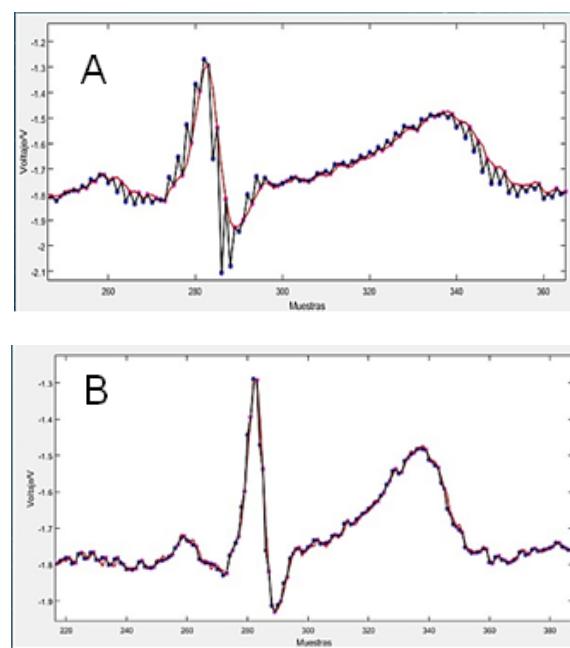
**Tabla 1.** Resultados variando el número de neuronas por capa.

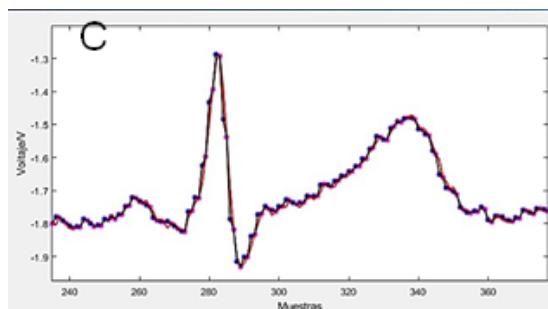
Neuronas capa oculta	Neuronas capa de salida	Error de validación	Error de entrenamiento
45	20	0.0335	0.0015
36	8	0.0450	0.0021
15	8	0.0388	0.005
22	10	0.0157	$7.57 \times 10^{-4}$
8	10	0.0153	$3.9422 \times 10^{-4}$
5	5	0.0155	$3.45 \times 10^{-4}$
4	3	0.0150	$2.98 \times 10^{-4}$
2	3	0.0164	$2.81 \times 10^{-4}$
1	1	0.0150	$2.87 \times 10^{-4}$

**Fuente:** Elaboración propia.

Se adicionaron señales de otros pacientes a los datos de validación, con el fin de evaluar el entrenamiento de la red y las arquitecturas que habían obtenido los mejores resultados, como se ilustra en la Figura 2.

**Figura 2.** A) Señal de validación para la red con 15 neuronas en la capa oculta y 8 en la capa de salida. B) Señal de validación para la red con 5 neuronas en la capa oculta y 5 en la capa de salida. C) Señal de validación para la red con 1 neurona en la capa oculta y 1 en la capa de salida.

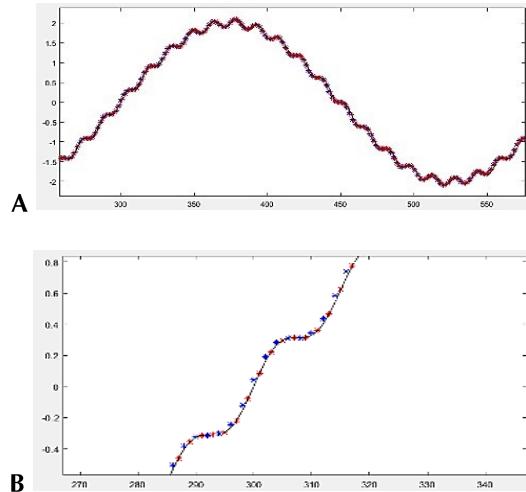




**Fuente:** Elaboración propia.

Como última validación, se experimenta con la red final sobre una señal completamente distinta a las bases de datos, generada en MatLab®, obteniendo excelentes resultados como se muestra en la Figura 3.

**Figura 3.** Interpolación sobre señal generada en MatLab®.



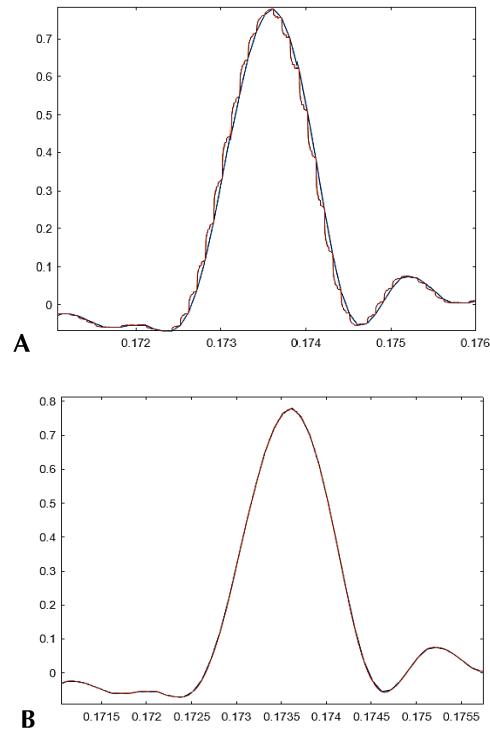
**Fuente:** Elaboración propia.

Durante la experimentación con nuevas señales, se observó un patrón peculiar: al reducir el número de neuronas manteniendo una estructura de dos capas ocultas y una de salida, el error de generalización mejoraba, a tal punto de que con una neurona en cada capa se encontró el menor error. Una vez en

este punto se realizaron pruebas con diferentes redes, una sola capa oculta, tres capas ocultas, e incluso el perceptrón de una neurona; para todos estos casos encontrando que la generalización decayó abruptamente si se tenían menos neuronas y relativamente poco para el caso de tres capas ocultas.

Aplicando el algoritmo de interpolación iterativo sobre la red obtenida en la experimentación anterior con la **base de datos A**, se obtienen los siguientes resultados sobre una señal de validación ECG se representa en traza azul como señal original y en rojo como la interpolación obtenida, ver Figura 4.

**Figura 4.** A) Interpolación con MLP entrenado con base de datos A. B) Interpolación con MLP entrenado con base de datos B.



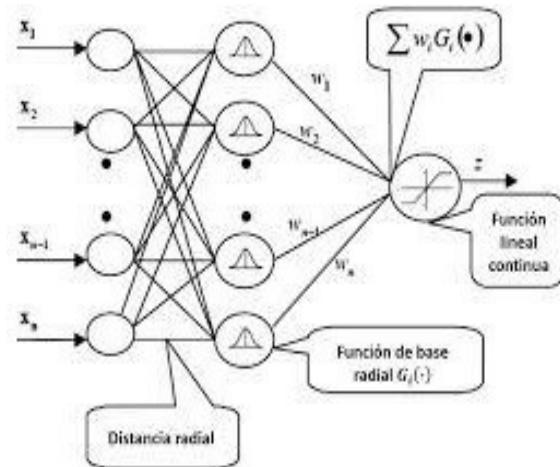
**Fuente:** Elaboración propia.

Como se puede ver en la Figura 4-A, la muestra central entre cada punto tiene un error muy grande, por lo que con cada iteración este error es acumulativo, generando patrones de curvas erróneas entre cada par de puntos de la señal. Para resolver este inconveniente, se decide reentrenar la red con la **base de datos B**. Los resultados del reentrenamiento de la misma arquitectura sobre la misma señal de prueba se muestran en la Figura 4-B, a partir de este momento esta red será conocida como **mejornet1**.

## 5. Experimentación con Redes de Base Radial

Debido a su similitud en aplicaciones con la topología de perceptrón multicapa, se elige la arquitectura red de base radial para resolver de manera alternativa el problema, como nuevo parámetro de evaluación se decide utilizar la función de correlación cruzada disponible en MatLab® eligiendo el valor máximo del arreglo resultante. Se realizó el entrenamiento de estas redes variando el número de neuronas de base radial y el parámetro *spread*, obteniendo las redes mostradas en la Figura 5 obteniendo los resultados mostrados en la tabla 2

**Figura 5.** Esquema de la arquitectura Base Radial.



**Fuente:** "Artificial Neural Model based on radial basis function networks used for prediction of compressive strength of fiber-reinforced concrete mixes" [4].

**Tabla 2.** Redes obtenidas con el entrenamiento en base.

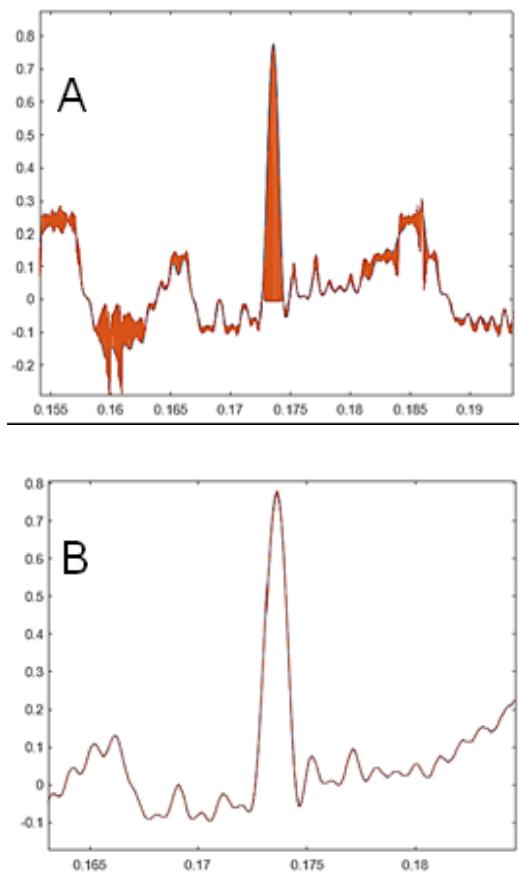
Nombre de la red	Spread	Número de neuronas
r_0135	0.1	35
r_0150	0.1	50
mejornet2	90	9
r_905	90	5
r_509	50	9
r_19	1	9
r_0530	0.5	30
r_1015	10	15
r_3010	30	10
r_04535	0.45	35

**Fuente:** Elaboración propia.

### 5.1 Señal ECG

Se muestran resultados gráficos de algunas de estas redes sobre la misma señal ECG en la Figura 6.

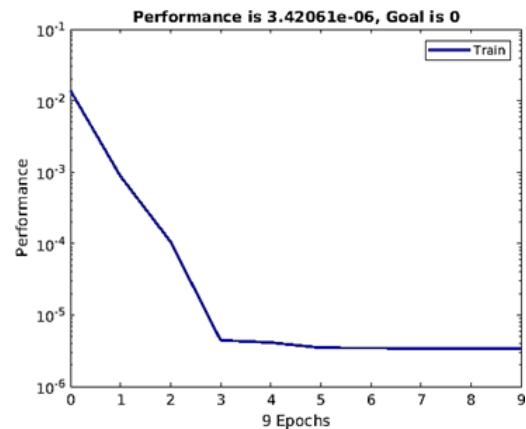
**Figura 6.** A) Señal ECG con la red r\_0150. B) Señal ECG con la red r\_1050.



**Fuente:** Elaboración propia.

Como se puede observar en los resultados gráficos, un mayor spread genera un rendimiento mayor como las mismas neuronas, según el problema, el rendimiento tiende a su valor mínimo más rápido con un spread grande, para luego estabilizarse a partir de cierto número de neuronas, como se muestra en la Figura 7.

**Figura 7.** Rendimiento con spread de 90.



**Fuente:** Elaboración propia.

A simple vista la generalización parece mejorar significativamente con la nueva base de datos, sin embargo, dado que los puntos de señales obtenidos corresponden a señales previamente muestreadas y no a la señal continua original, resulta imposible cuantificar el error; por lo que se opta por cambiar el conjunto de validación a cuatro señales que, a criterio del autor, son desafiantes para el entrenamiento y están definidas a partir de una función continua.

## 5.2 Señal Sinusoidal

Esta señal está definida con  $0 < x < 4\pi$  y se compone de la suma de señales *Sin* a diferente amplitud y frecuencia a modo de ecuación 1.  $y = 2\sin(x) + 0.1\sin(20x)$  (1)

Se usa la señal sinusoidal con las diferentes redes vistas anteriormente. A continuación, se muestran algunos resultados gráficos de las pruebas. Los resultados numéricos se

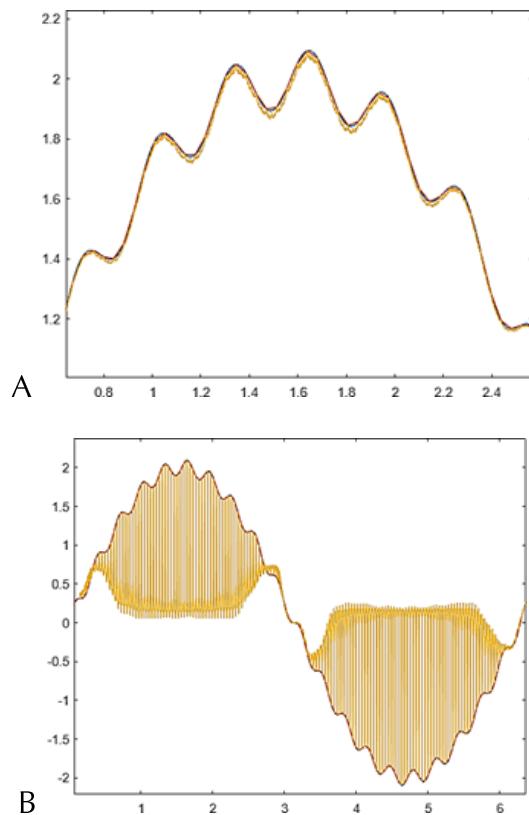
condensan en la Tabla 3. Para esta señal se logró interpolar 16 muestras por cada muestra original, ver Figura 8.

**Tabla 3.** Error y la máxima correlación en la señal seno.

Red neuronal	Error	Máxima correlación
<i>r_0135</i>	1.9262	599.6497
<i>r_19</i>	1.7113	1.3332*10 <sup>3</sup>
<i>mejornet2</i>	8.402*10 <sup>-6</sup>	9.5425*10 <sup>3</sup>
<i>r_905</i>	3.1873*10 <sup>-4</sup>	9.5385*10 <sup>3</sup>
<i>net_vieja</i>	9.3047*10 <sup>-4</sup>	9.5364*10 <sup>3</sup>
<i>mejornet1</i>	0.0038	9.2243*10 <sup>3</sup>

**Fuente:** Elaboración propia.

**Figura 8.** A) Señal sinusoidal con la red *r\_19*. B) Señal sinusoidal con la red *mejornet2*.



**Fuente:** Elaboración propia.

Como prueba adicional, se varió la amplitud de la señal de validación multiplicándola por un factor, con el objetivo de analizar cómo incide la magnitud de la señal de entrada en el error de validación y correlación. Los resultados obtenidos se condensan en la Tabla 4.

**Tabla 4.** Error y la máxima correlación en la señal seno variando su amplitud y con la red *mejornet2*.

Factor de multiplicación	Error	Máxima correlación
<b>10</b>	11.7975	7.5148*10 <sup>5</sup>
<b>1</b>	8.4021*10 <sup>-5</sup>	9.5425*10 <sup>3</sup>
<b>0.1</b>	5.2965*10 <sup>-8</sup>	95.7184
<b>0.01</b>	1.3254*10 <sup>-9</sup>	0.9572
<b>0.001</b>	8.6224*10 <sup>-10</sup>	0.0096

**Fuente:** Elaboración propia.

Como se muestra en la Tabla 4, normalizar previamente la señal de entrada a valores entre 0.5 y -0.5 incrementa el rendimiento del algoritmo de reconstrucción.

Para el caso de la señal sinusoidal en la Tabla 3, el error más bajo se obtuvo con la red *mejornet2*, correspondiente a la red de base radial entrenada en con un spread de 90 y 9 neuronas, así mismo, esta red obtuvo la correlación más alta, lo que significa una relación más estrecha entre la señal interpolada y la original. Por otro lado, la red *r\_0135* presentó el error más alto y la correlación más baja obteniendo los peores resultados debido a que su red era más compleja, contaba con 35 neuronas y un spread muy pequeño de 0.1.

### 5.3 Señal Sinc

Esta señal se define tal que  $-10 < x < 10$ , a modo de ecuación 2, reemplazando la indeterminación en 0 por  $y=1$ .

$$v = \frac{\sin(x)}{x} \quad (2)$$

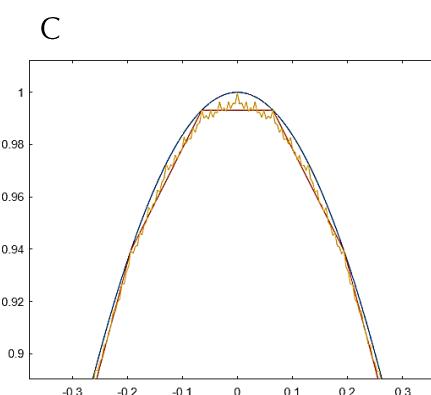
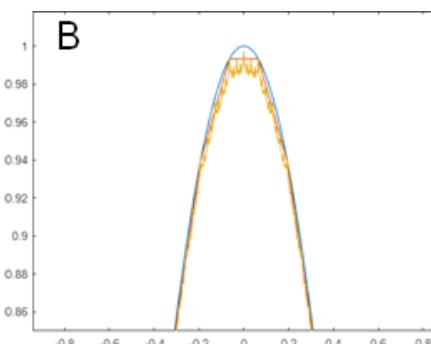
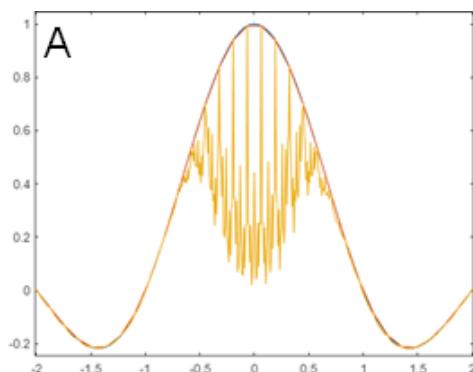
Usando como entrada la señal Sinc, a continuación, se muestran algunos resultados gráficos de las pruebas, en la tabla 5 se condensan los resultados numéricos y en la Figura 9 se logró interpolar 32 muestras por cada muestra original.

**Tabla 5.** Error y la máxima correlación en la señal SinC con diferentes redes.

Red neuronal	Error	Máxima correlación
<b>r_0135</b>	0.0472	19.3673
<b>r_0530</b>	0.0203	114.9768
mejornet2	$9.7330 \times 10^{-7}$	244.4457
r_905	$4.0404 \times 10^{-6}$	243.4674
mejornet1	$3.3726 \times 10^{-6}$	243.6351

**Fuente:** Elaboración propia.

**Figura 9.** A) Señal SinC con la red r\_0530. B) Señal SinC con las red mejornet1. C) Señal SinC con las red mejornet2.



**Fuente:** Elaboración propia.

Al igual que en la señal sinusoidal, se varía la amplitud de las señales para ver su efecto en la interpolación de la red, se pueden ver los resultados en la Tabla 6.

**Tabla 6.** Error y la máxima correlación en la señal SinC variando su amplitud y con la red mejornet2.

Factor de multiplicación	Error	Máxima correlación
10	$5.821 \times 10^{-6}$	934.2063
1	$4.6852 \times 10^{-6}$	936.4074
0.1	$4.794 \times 10^{-6}$	936.5695
0.01	$6.197 \times 10^{-6}$	933.7566
0.001	$5.3182 \times 10^{-6}$	935.6643

**Fuente:** Elaboración propia.

En la tabla 5 se puede evidenciar que el menor error se obtuvo con la red mejornet2, igualmente se obtuvo la correlación más alta, mientras que con la red *r\_0150* se obtuvieron los peores resultados, es una red de 50 neuronas y un *spread* de 0.1.

#### 5.4 Polinomio de Alto Orden

Esta señal se define con  $-1.6 < x < 1.5$ , a modo de ecuación 3, por la suma de un polinomio de grado 15 y una señal sinusoidal con amplitud exponencialmente decreciente.

$$y = x^{15} - x^{13} - x^9 - 10x^5 + 5x^2 + 5e^{-2x} * \sin\left(\frac{x}{0.01}\right) \quad (3)$$

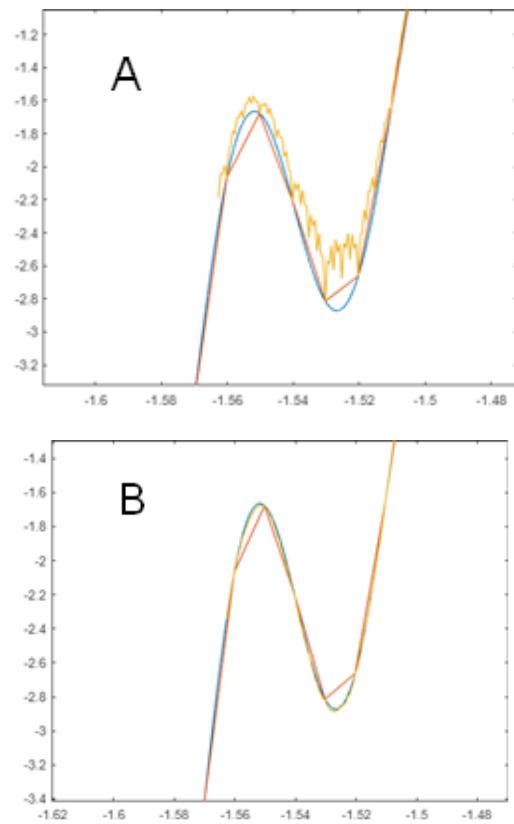
Algunos resultados numéricos de las pruebas se condensan en la Tabla 7, al igual que los resultados gráficos en la Figura 10. logrando interpolar 16 muestras por cada muestra original.

**Tabla 7.** Error y la máxima correlación en el polinomio de alto orden con diferentes redes y un  $\alpha = 0.01$

Red neuronal	Error	Máxima correlación
<i>r_0135</i>	0.1466	58.1016
<i>r_1015</i>	0.0023	702.0201
<i>mejornet2</i>	$2.1343 \times 10^{-5}$	767.6993
<i>r_905</i>	$9.4433 \times 10^{-6}$	771.8631
<i>mejornet1</i>	$7.3327 \times 10^{-4}$	731.0506

**Fuente:** Elaboración propia.

**Figura 10.** A) Señal Polinomio con la red mejornet1.  
B) Señal Polinomio con la red *r\_905*.



**Fuente:** Elaboración propia.

Al igual que en las anteriores señales, se varía la amplitud de las señales para ver su efecto en la interpolación de la red, se pueden apreciar los resultados en la Tabla 8.

**Tabla 8.** Error y la máxima correlación en el polinomio de alto orden variando su amplitud y con la red *r\_905*.

Factor de multiplicación	Error	Máxima correlación
10	$1.2548 \times 10^{-6}$	$1.2437 \times 10^6$
1	0.841	$6.2993 \times 10^4$
0.1	$9.4433 \times 10^{-6}$	771.8631
0.01	$5.1178 \times 10^{-8}$	7.6909
0.001	$5.4087 \times 10^{-9}$	0.0768

**Fuente:** Elaboración propia.

En la Tabla 8, se puede ver que los mejores resultados en cuestión de error y máxima correlación se obtuvieron con la red *r\_905*, una red entrenada en base radial con un spread de 90 y 5 neuronas, red que obtuvo la más alta correlación en contraste con el resultado bastante malo de la red *r\_0135*, la cual tiene un *spread* bastante pequeño comparado con la red con la que se obtuvieron los mejores resultados.

## 5.5 Atractor de Lorenz

Para esta función se usa uno de los componentes tridimensionales (Y) respecto al tiempo del atractor de Lorenz, definido por el conjunto de ecuaciones diferenciales:

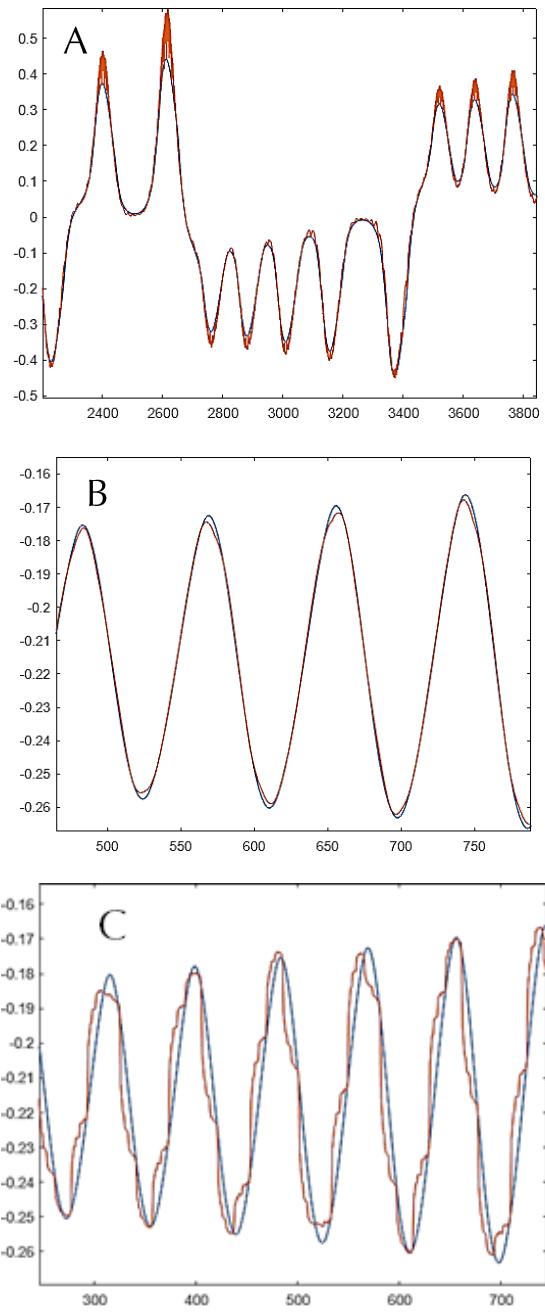
$$\begin{aligned}\dot{x} &= \sigma * (y - x) \\ \dot{y} &= x * (\rho - z) - y \\ \dot{z} &= x * y - \beta * z;\end{aligned}\quad (4)$$

A continuación, se muestran algunos resultados de las pruebas, los resultados numéricos se condensan en la Tabla 9. Para esta señal se logró interpolar 16 muestras por cada muestra original, ver Figura 11.

**Tabla 9.** Error y la máxima correlación en una de las componentes del atractor de Lorenz.

Red neuronal	Error	Máxima correlación
<i>r_19</i>	$6.4749 \times 10^{-4}$	991.4742
<i>r_0150</i>	0.1461	139.8887
<i>r_905</i>	$4.794 \times 10^{-6}$	936.5695
<i>r_509</i>	$4.6852 \times 10^{-6}$	936.4074
<i>net_vieja</i>	$7.3265 \times 10^{-6}$	933.3367
<i>mejornet1</i>	$5.8164 \times 10^{-6}$	934.2139

**Figura 11.** A) Componente del atractor de Lorenz con la red *r\_19*. B) Componente del atractor de Lorenz con la red *r\_905*. C) Componente del atractor de Lorenz con la red *net vieja*.



**Fuente:** Elaboración propia.

**Fuente:** Elaboración propia.

**Tabla 10.** Error y la máxima correlación en la componente del atractor de Lorenz variando su amplitud y con la red *r\_509*.

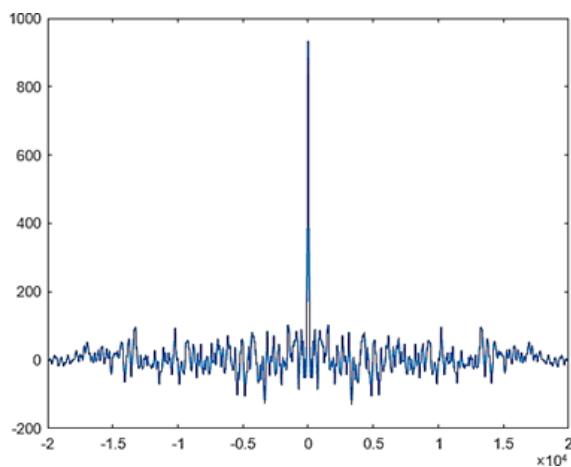
Factor de multiplicación	Error	Máxima correlación
10	0.0061	9.1032*10 <sup>4</sup>
1	4.6852*10 <sup>-6</sup>	936.4074

Factor de multiplicación	Error	Máxima correlación
0.1	5.3875*10 <sup>-8</sup>	9.3628
0.01	7.5314*10 <sup>-9</sup>	0.00932
0.001	7.0161*10 <sup>-9</sup>	8.9435*10 <sup>9</sup>

**Fuente:** Elaboración propia.

Se muestra la forma de la señal de correlación cruzada para el polinomio de alto grado en la Figura 12.

**Figura 12.** Correlación cruzada entre señal original e interpolación con la red *mejornet2*.



**Fuente:** Elaboración propia.

En la Tabla 10 se pueden ver el error y la máxima correlación para cada una de las redes entrenadas, en este caso la red que arrojó el mejor resultado fue la red *r\_509* la

cual fue entrenada en base radial y tiene un spread de 50 con 9 neuronas, mientras que con la red *r\_0150*, se obtuvo el error más alto, y la máxima correlación más baja.

Al observar los resultados de las tablas en las diferentes señales, observamos que la red con mejores resultados es la *red mejornet2*, igualmente, la red *r\_905* también obtiene muy buenos resultados. Mientras que las redes con un spread bajo como lo son las redes *r\_0135* y *r\_0150* arrojan unos resultados bastante malos.

## 6. Conclusiones

Aunque las dos topologías utilizadas en la experimentación obtuvieron un error bastante bajo, las redes neuronales de base radial mostraron un desempeño superior en la interpolación respecto al perceptrón multicapa, superándolo en la mayoría de las pruebas con un margen considerable en el error cuadrático medio. Con el objetivo de determinar si las redes obtenidas pueden implementarse en señales en tiempo real, se hizo necesario evaluar el peso computacional de ambas, sorprendentemente resultaron ser viables para hardware de baja potencia. Sin embargo, en términos de costo computacional en entrenamiento y tiempo de este, se evidenció que la base radial es más costosa, por un margen muy amplio, llegando a necesitar casi 10 veces más recursos que MLP.

Por otro lado, se puede notar que la eficacia de la interpolación es dependiente de la amplitud de la señal de entrada, esto se debe a que dicha señal debe encontrarse en el mismo rango de las señales de entrenamiento. Se concluye que no siempre las redes más complejas o grandes estructuralmente son siempre las mejores para atacar un problema, existen circunstancias en las que una red simple puede generalizar mejor los nuevos datos de entrada, como es este caso. Para futuras aplicaciones los autores sugieren la implementación del sistema con base radial debido a su rendimiento y ligereza, para mejores resultados también se recomienda normalizar la señal de entrada en un rango de valores de -0.5 hasta 0.5 previo al uso de la red.

## Referencias

- [1] R. A. Valdesueiro, "Muestreo digital," p. 12, 2004.
- [2] M. Semeria, "Área: Ingeniería Informática," p. 17, 2020.
- [3] A. Hashemi Fath, F. Madanifar, y M. Abbasi, "Implementation of multilayer perceptron (MLP) and radial basis function (RBF) neural networks to predict solution gas-oil ratio of crude oil systems," *Petroleum*, vol. 6, no. 1, pp. 80–91, mar. 2020. doi: [10.1016/j.petlm.2018.12.002](https://doi.org/10.1016/j.petlm.2018.12.002).
- [4] L. O. González Salcedo, A. P. Guerrero Zúñiga, S. Delvasto Arjona, y A. L. E. Will, "Artificial Neural Model based on radial basis function networks used for prediction of compressive strength of fiber-reinforced concrete mixes," *Cien. Ing. Neogranadina*, vol. 29, no. 2, pp. 37–52, jun. 2019. doi: [10.18359/rcin.3737](https://doi.org/10.18359/rcin.3737).
- [5] A. Sudou, P. Hartono, R. Saegusa, y S. Hashimoto, "Signal reconstruction from sampled data using neural network," en *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing*, Martigny, pp. 707–715, 2002. doi: [10.1109/NNSP.2002.1030082](https://doi.org/10.1109/NNSP.2002.1030082).
- [6] A. Ugena, "The newton neural net: a new approximating network," *Int. J. of Pure and Appl. Math.*, vol. 82, no. 4, Feb. 2013. doi: [10.12732/ijpam.v82i4.13](https://doi.org/10.12732/ijpam.v82i4.13).
- [7] N. M. Khan, "Audio Signal Reconstruction Using Cartesian Genetic Programming Evolved Artificial Neural Network (CGPANN)," p. 6, 2017. <https://doi.org/10.1109/ICMLA.2017.0-100>
- [8] L. H. C. Casallas, E. H. M. Alfonso, y M. L. C. Martínez, "Clasificación de Plasmodium Falciparum por estadio en cultivos sincrónicos de eritrocitos," *Visión electrónica*, vol. 5, no. 1, May 2011. doi: [10.14483/22484728.3519](https://doi.org/10.14483/22484728.3519).
- [9] J. A. P. Plaza, D. R. Zapata, y A. T. Tascón, "Implementación de redes

- neuronales utilizando dispositivos lógicos programables," *Visión electrónica*, vol. 1, no. 1, Jun. 2008. doi: [10.14483/22484728.250](https://doi.org/10.14483/22484728.250).
- [10] O. L. Ramos, D. A. Rojas, y L. A. Góngora, "Reconocimiento de patrones de habla usando MFCC y RNA," *Visión electrónica*, vol. 10, no. 1, Jun. 2016. doi: [10.14483/22484728.11712](https://doi.org/10.14483/22484728.11712).
- [11] E. J. G. Monterroza, "Reconocimiento de primitivas 3D, usando autocorrelación y ANFIS," *Visión electrónica*, vol. 1, no. 1, 2008. doi: [10.14483/22484728.251](https://doi.org/10.14483/22484728.251).
- [12] L. F. P. Martínez, Ó. F. C. Camargo, y J. E. Roa, "Estudio comparativo de técnicas artificiales para la predicción de una serie de tiempo caótica," *Visión electrónica*, vol. 2, no. 2, Dic. 2008. doi: [10.14483/22484728.792](https://doi.org/10.14483/22484728.792).
- [13] A. E. Díaz y L. A. Calderón, "Modelo tridimensional de extremidad inferior basado en imágenes de resonancia magnética," *Visión electrónica*, vol. 3, no. 1, Jun. 2009. doi: [10.14483/22484728.686](https://doi.org/10.14483/22484728.686)