

Sistema de reconocimiento inteligente de señales mioeléctricas del movimiento de mano humana

Vicario Vázquez, S. A.¹, Oubram, O.², Bassam, A.³, Velázquez Aguilar, J. G.²,
Ordóñez López, E.²

Fecha de recepción: 23 de junio de 2017– Fecha de aprobación: 14 de julio de 2017

RESUMEN

En este trabajo se presenta el diseño de un algoritmo de reconocimiento de señales mioeléctricas por medio de una arquitectura de redes neuronales artificiales, para la fabricación de un prototipo de prótesis de una mano humana. En la primera etapa del proyecto se diseñó el sensor mioeléctrico que ayudo a la captura de las señales que corresponden al movimiento de cada dedo de una mano humana. Se generó una base de datos con las señales mioeléctricas capturadas, la cual sirvió para el entrenamiento de la red neuronal artificial (RNA), obteniendo los pesos y bias de dicha red. El rendimiento de la arquitectura fue evaluado con criterios estadísticos para la validación de la RNA, comparando datos simulados con datos experimentales. Encontramos que la mejor arquitectura en este trabajo cuenta con 7 neuronas en la capa oculta, una en la capa de salida y con un 96% de coeficiente de correlación, dicha arquitectura es la #7 de la tabla 1 la cual contiene un reporte de rendimiento del algoritmo de aprendizaje de las diferentes arquitecturas propuestas.

Palabras clave: señales mioeléctricas, reconocimiento inteligente, prótesis, redes neuronales artificiales.

Intelligent recognition system of myoelectric signals of human hand movement

ABSTRACT

In this paper presents the recognition of myoelectric signal's algorithm design thru artificial neural network architecture, for the manufacture of a prototype of human hand prosthesis. At the beginning of the project the myoelectric sensor was designed to help capture the signals that correspond to the movement of each finger of a human hand. A database was generated with captured myoelectric signals, which was used for the training of artificial neural (RNA), obtaining the weights and bias. The performance of the architecture was evaluated with statistical criteria for the validation of RNA, comparing between simulated data and experimental data. We found, that the best architecture in this project has 7 neurons in the hidden layer, one in the output layer and 96% correlation coefficient, this architecture is # 7 in Table 1 which contains a performance report learning algorithm of the different architectures proposed.

Keywords: Myoelectric signals, intelligent recognition, prosthesis, neural network.

¹ Estudiante de licenciatura en Ingeniería, Ingeniería Eléctrica. Facultad de Ciencias Químicas e Ingeniería-UAEM.

² Departamento de Ingeniería eléctrica y electrónica. Facultad de Ciencias Químicas e Ingeniería-UAEM.

Autor de correspondencia: oubram@uaem.mx

³ Departamento de Energías Renovables. Facultad de Ingeniería-UADY.

Nota: Este artículo de investigación es parte de Ingeniería–Revista Académica de la Facultad de Ingeniería, Universidad Autónoma de Yucatán, Vol. 21, No. 2, 2017, ISSN: 2448-8364.

INTRODUCCIÓN

Una prótesis es un artefacto desarrollado con el fin de mejorar la calidad de vida de las personas que por algún accidente perdieron alguna extremidad de su cuerpo. Por medio del avance de la tecnología en las últimas décadas las prótesis cada vez son más avanzadas.

El cuerpo humano es capaz de generar señales eléctricas en sus músculos, a estas señales se les conoce como señales bioeléctricas. Las señales bioeléctricas se dividen en diferentes tipos dependiendo de la procedencia de dicha señal. Las señales mioeléctricas son las generadas por la contracción de algún músculo de cualquier extremidad como lo son los brazos y las piernas, pueden medirse con un equipo adecuado y así utilizar la información que estas nos proporcionan en el diseño de prótesis.

El trabajo de (P. Parimal *et al.*, 1998), realiza un sistema micro controlado, basado en el microcontrolador 68HC11, El sistema amplifica las señales mioeléctricas, las filtra, las digitaliza y el algoritmo de control decide el movimiento de un robot de dos grados de libertad. Otro trabajo similar (Alonso. R *et al.*, 2001), en el cual, a diferencia del anterior, se acciona una prótesis mioeléctrica. En sistemas orientados a la asistencia en ambientes de trabajo industrial, se encuentra el trabajo de (N. López *et al.*, 2006). En el presente trabajo se diseñó un sensor mioeléctrico, el cual es el encargado de capturar, amplificar y filtrar las señales generadas por la contracción de los músculos responsables del movimiento de cada dedo. Este tipo de sensores está caracterizado por contar con las siguientes etapas:

- Pre amplificación.
- Filtro pasa altas.
- Filtro pasa baja.
- Amplificación y acoplamiento de señal.
- Selección de cables para EMG.

La electromiografía (EMG) es adquirida del miembro seleccionado por medio de electrodos conectados a un sensor mioeléctrico cuya salida va a una tarjeta de adquisición de datos. Con el uso del software Matlab se realizan diferentes procesos a la señal adquirida, con los resultados obtenidos se generó una base de datos, la cual posteriormente es utilizada para el entrenamiento de la RNA.

Las RNA se definen como un sistema de mapeo no lineales. Constan de un número de procesadores simples ligados por conexiones con pesos. Las unidades de procesamiento se denominan neuronas. Cada unidad recibe entradas de otros nodos y generan una salida simple escalar que depende de la información local disponible, y guardada internamente o que llega a través de las conexiones con pesos.

Las neuronas artificiales simples fueron introducidas por McCulloch y Pitts en 1943. Una red neuronal artificial se caracteriza por los siguientes elementos:

1. un conjunto de unidades de procesamiento o neuronas.
2. Un estado de activación para cada unidad, equivalente a la salida de la unidad.
3. conexiones entre las unidades, generalmente definidas por un peso que determina una señal de entrada a la unidad.
4. Una regla de propagación, que determina la entrada efectiva de una unidad a partir de las entradas externas.
5. Una función de activación que actualiza el nuevo nivel de activación basándose en la entrada efectiva y la activación anterior.
6. Una entrada externa que corresponde a

- un término determinado como bias para unidad.
7. Un método para reunir la información, correspondiente de la regla de aprendizaje.
 8. Un ambiente en el que el sistema va a operar, con señal de entrada e incluso e incluso señales de error.

En muchas redes las unidades de proceso tienen respuesta de la forma:

$$y = f\left(\sum_k \omega_k x_k\right) \quad (1)$$

Dónde:

x_k = Señales de salida de otros nodos o entradas externas.

ω_k = Pesos de las ligas de conexión.

$f(*)$ =Función no lineal simple.

Un modelo abstracto y simple de una neurona artificial (Figura 1) compuesto por un conjunto de entradas $X = (x_1, x_2, x_3, \dots, x_i)$ la cual simula la función de las dendritas; un vector de pesos $W = (w_1, w_2, w_3, \dots, w_i)$ el cual emula la función de una sinapsis; un umbral de acción o también llamado bia θ ; una función de activación el cual se considerado el equivalente al soma y una salida (I. Pedro & G. Inés, 2004).

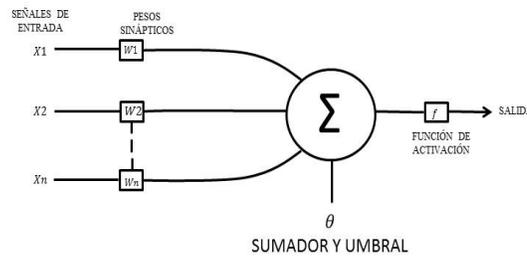


Figura 1. Elementos naturales de una neurona.

La descripción de los componentes que conforma una neurona artificial son:

- Vector de entrada: este vector $X = (x_1, x_2, x_3, \dots, x_i)$ contiene la información que la neurona va a procesar; donde “i” es el número de entradas a la neurona.
- Vector de pesos sinápticos: Cada uno de los elementos del vector de entradas $X = (x_1, x_2, x_3, \dots, x_i)$ es multiplicado por un valor ajustable denominado peso sináptico y se representa por el vector $W = (w_1, w_2, w_3, \dots, w_i)$. Este vector es ajustado durante el entrenamiento, con la finalidad de minimizar el error de la salida de la neurona con respecto a la salida esperada.
- Umbral o bia: Se denota por el símbolo θ

y es considerado como un peso adicional que recibe una entrada con un valor igual a la unidad.

- Función de activación o de transferencia: la regla que establece el efecto de la entrada total u_t en la activación de la unida se denomina función de activación F_k (C. Pedro, 2010).

Algunas de las funciones más utilizadas son la lineal, sigmoidea y tangente hiperbólica (V. V. A. Sergio, 2017).

Las ecuaciones son las que se en listan respectivamente:

$$f(x) = x \quad (2)$$

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

$$f(x) = \frac{2}{1 + e^{-2x}} - 1 \quad (4)$$

La RNA utilizada en este caso se conforma por un vector de 8 entradas, 7 neuronas en la capa oculta, una neurona en la capa de salida

$$OUT = \sum_{j=1}^7 \left[WO_{(n,j)} \left(\frac{2}{1 + e^{-2(\sum_{i=1}^8 (WI_{(j,i)} * In_{(i)}) + B2)}} - 1 \right) \right] + B1 \quad (5)$$

DONDE:

- $WI_{(j,i)}$ = matriz de pesos sinápticos de las entradas
- $WO_{(n,j)}$ = matriz de pesos sinápticos de la capa oculta
- $In_{(i)}$ = datos de entrada.
- $B1$ = bias de la capa oculta.
- $B2$ = bia de la capa de salida.

Partiendo de una serie de pesos sinápticos aleatorios, el proceso de aprendizaje busca un conjunto de pesos que permitan a la red desarrollar una determinada tarea. La mayoría de los métodos de entrenamiento utilizados en las redes neuronales con conexión hacia delante consisten en proponer una función de error que mida el rendimiento actual de la red en función de los pesos sinápticos. El objetivo del método de entrenamiento es encontrar el conjunto de pesos sinápticos que minimizan (o maximizan) la función. Existen dos tipos de aprendizaje, si la red aprende durante su funcionamiento habitual o si se requiere la desconexión de la red hasta que el proceso termine redes on-line y off-line, respectivamente (I. Pedro & G. Inés, 2004).

y una salida. Teniendo en cuenta la ecuación 4 para la capa oculta y la ecuación 2 para la capa de salida, su salida está dada por:

Algunos de los algoritmos de aprendizaje se enlistan a continuación:

- Algoritmos de gradiente conjugado.
- Gradiente conjugado escalado (trainsecg).
- Algoritmo BFGS (trainbfg).
- Levenberg-Marquardt (trainlm).

Posterior al entrenamiento se sometieron a diferentes procesos estadísticos con la finalidad de comparar el desempeño de las diferentes arquitecturas RNA obtenidas.

A continuación se describen algunos de las ecuaciones estadísticas utilizadas para comparar grupos de datos generados con redes neuronales artificiales. La salida de la RNA son OUT_{SIM} los datos simulados, los experimentales son nombrados con OUT_{EXP} y n representa el número de valores de las muestras (V. V. A. Sergio, 2017).

Coefficiente de determinación mide el grado de dependencia entre variables, tomando el valor 0 en caso de correlación nula o el valor 1 en caso de correlación total. Equivale al cuadrado del coeficiente de correlación.

$$R^2 = 1 - \frac{\sum_{i=1}^n (OUT_{exp(i)} - OUT_{sim(i)})^2}{\sum_{i=1}^n (OUT_{exp(i)} - OUT_{exp})^2} \quad (6)$$

El RMSE (Root Mean Square Error, por sus siglas en inglés) es una medida del

grado de dispersión de los datos con respecto al valor promedio. Es conocida

como la desviación estándar para una distribución de probabilidad discreta:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (OUT_{sim(i)} - OUT_{exp(i)})^2}{n}} \quad (7)$$

MPE (Mean Percentage Error, por sus siglas en inglés) es la media del error porcentual. Es una métrica simple, que sirve para ver si el error del pronóstico

tiene un sesgo (*bias*) positivo o negativo. También se dice que el pronóstico está subestimado o sobrestimado.

$$MPE = \frac{\sum_{i=1}^n \left(\frac{OUT_{exp(i)} - OUT_{sim(i)}}{OUT_{exp(i)}} \right)}{n} \times 100(\%) \quad (8)$$

El objetivo principal de este trabajo consiste en usar RNAs para hacer reconocimiento de señales mioeléctricas. En los siguientes apartados se describirán la metodología, que consiste en una breve explicación del sistema experimental, la división de la base de datos y el método para ajustar la arquitectura de RNA. Finalmente, se presentan los resultados obtenidos y la validación del modelo matemático de la red así como un prototipo de prótesis de una mano humana (A Bassam *et al.*, 2014).

METODOLOGÍA

Sistema experimental y formación de base de datos la metodología de desarrollo de este proyecto de investigación inicia con el diseño de un sensor mioeléctrico para la adquisición de las señales. Posteriormente se realizó una base de datos la cual fue formada a partir de señales mioeléctricas generadas por la contracción de los músculos responsables de los movimientos de cada dedo de una mano humana.

Sensor Mioeléctrico

Dado que los parámetros de las señales mioeléctricas a capturar son:

- La frecuencia de la señal mioeléctrica se encuentra 50 y 200 Hz.
- Estado de Reposo: es el estado donde las células excitables mantienen un potencial $-90\text{mV} < V < -50\text{mV}$.
- Estado Activo: en dicho estado las células excitables tienen un potencial eléctrico entre los $-55\text{mV} < V < 30\text{mV}$.

Se decidió diseñar un sensor mioeléctrico con las siguientes etapas:

- Pre amplificación:
- Filtro pasa altas.
- Filtro pasa baja.
- Amplificación y acoplamiento de señal.
- Selección de cables para EMG.

Las etapas del sensor mioeléctrico utilizadas así como los circuitos que se emplearon, se muestran en la figura 3.

Una vez que se obtuvo el sensor mioeléctrico, se procedió a realizar la EMG, es decir, a realizar capturas de señales de los músculos responsables del movimiento de los dedos de la mano. La EMG consta en colocar electrodos superficiales sobre la extremidad

requerida como se muestra en la figura 4, dichos sensores están conectados por cables a la entrada del sensor mioeléctrico, la salida del sensor mioeléctrico se conecta a la placa de desarrollo Arduino Mega 2560 la cual envía los datos a una PC. Con ayuda de la herramienta Matlab las señales obtenidas fueron graficadas y guardadas, para generar una base de datos necesaria para el entrenamiento de la red neuronal. En este trabajo los movimientos sobre los cuales se trabajó fueron la contracción de cada uno de los dedos de la mano y el conjunto de ellos al mismo tiempo, es decir, el movimiento de abrir y cerrar la mano completa, las señales se muestran en las siguientes figuras 5a-5g.

Una vez capturadas las señales y generando una base de datos, posteriormente se procedió a realizar una depuración de la misma, esta fue realizada sometiendo a la base de datos a un proceso estadístico el cual nos arrojó las variables que fueron consideradas para el vector de entrada de la RNA, dichas variables son: varianza, desviación estándar, valor medio cuadrático, valor RMS, coeficiente de simetría y curtosis. Para el entrenamiento de la red neuronal se utilizaron las variables derivadas de la depuración de la base de datos y la paquetería de Matlab, la cual nos ofrece una serie de funciones predefinidas las cuales facilitan el proceso de entrenamiento (Demuth *et al.*, 2007).

Diseño y Entrenamiento de Red Neuronal Artificial



Figura 2. Diagrama de estas del proyecto.

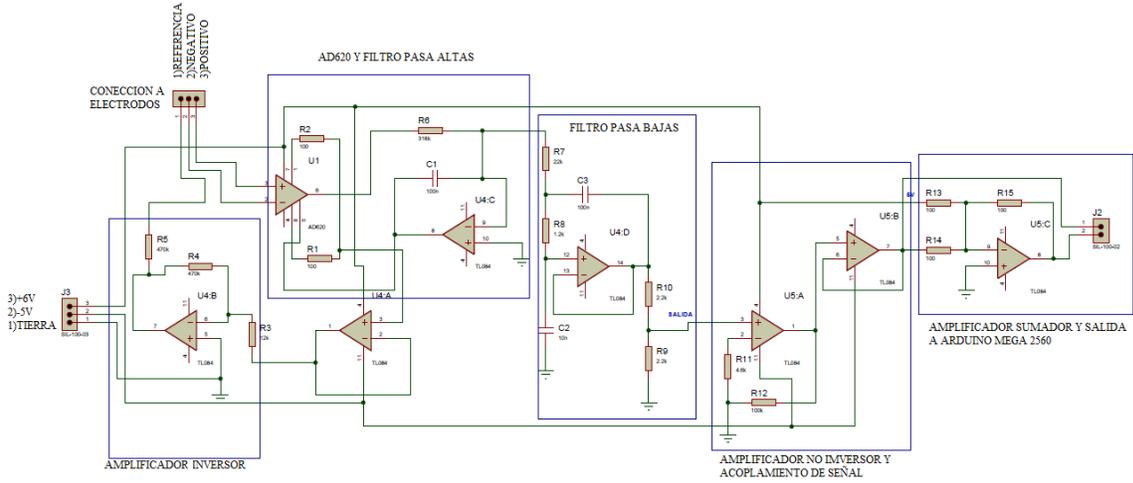


Figura 3. Sensor de señales mioeléctricas con todas sus etapas.



Figura 4. Colocación de los electrodos superficiales para realización de EMG.

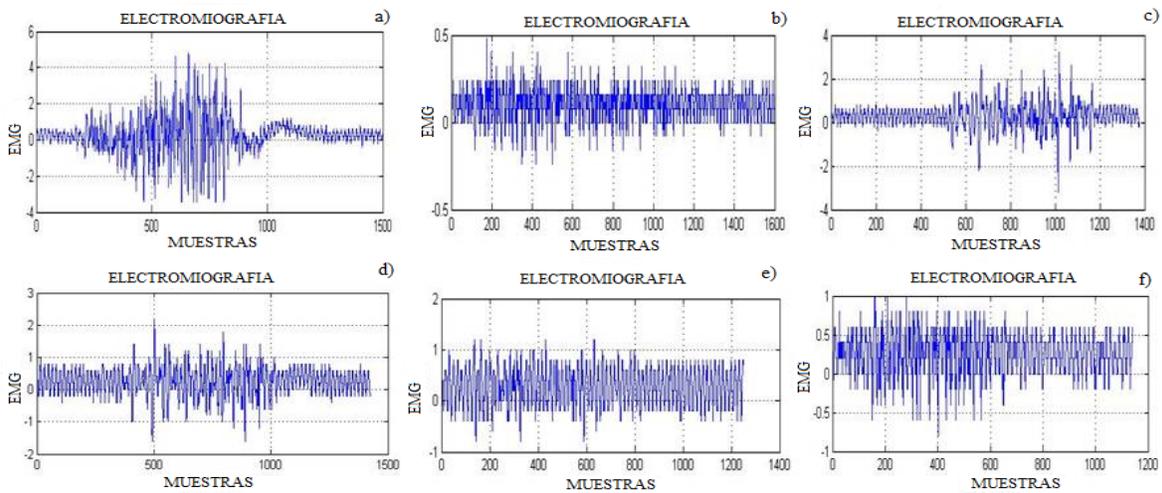


Figura 5

- a) Señal mioeléctrica producida por el puño completo, adquirida por arduino y graficada en Matlab
- b) Señal mioeléctrica producida por el dedo índice, adquirida por arduino y graficada en Matlab

- c) Señal mioeléctrica producida por el dedo medio, adquirida por arduino y graficada en Matlab
- d) Señal mioeléctrica producida por el dedo anular, adquirida por arduino y graficada en Matlab
- e) Señal mioeléctrica producida por el dedo menique, adquirida por arduino y graficada en Matlab
- f) Señal mioeléctrica producida por el dedo pulgar adquirida por arduino y graficada en Matlab.

Se realizaron diferentes pruebas dividiendo la base de datos en 3 en diferentes porcentajes, es decir, 60% para entrenamiento, 20% de validación y 20% para test. Previo al entrenamiento se consideró definir diferentes criterios, tales como número de neuronas de la capa oculta, el factor de aprendizaje entre otros.

El algoritmo utilizado para el entrenamiento de las pruebas fue el Levenberg-Marquardt, ya que en estudios comparativos (V. V. A. Sergio, 2017) ha demostrado un mejor desempeño con respecto a los algoritmos de backpropagation, gradiente descendente, entre otros.

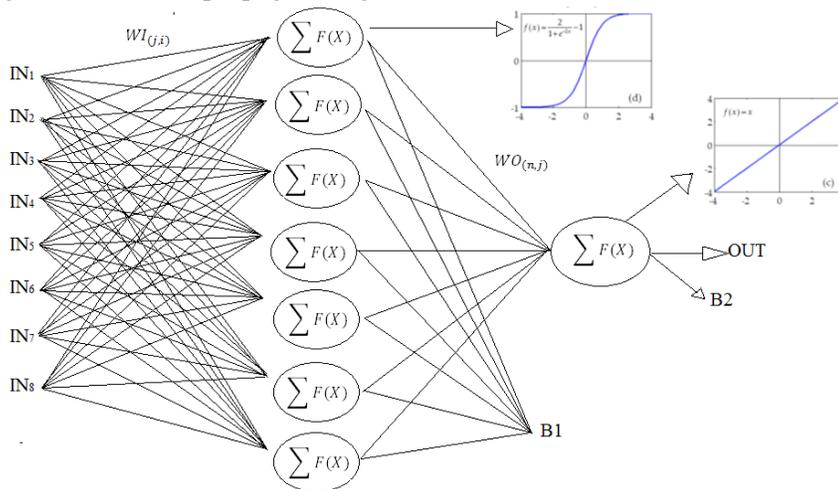


Figura 6. Modelo grafico de red neuronal.

Se propuso un algoritmo el cual ayudo a calcular los siguientes parámetros:

- Numero de neuronas en la capa oculta.
- Factor de aprendizaje.
- Coeficiente de correlación lineal.
- El RMSE.
- Los pesos sinápticos y bias.
- Almacenamiento de resultados de cada entrenamiento

La red neuronal artificial la cual cuenta con las siguientes características (figura 6):

- Tipo de red: perceptron.
- 8 entradas, 7 neuronas en la capa oculta, una neurona en la capa de

salida y una salida.

- Función hiperbólica en la capa oculta de la RNA.
- Función lineal en la capa de salida de la RNA.
- Un algoritmo de optimización Levenberg-Marquardt.

En la Tabla 1 se presenta un extracto de un reporte generado a partir del entrenamiento. Este reporte se obtuvo utilizando la base datos de las señales mioeléctricas y con las 8 variables derivadas de la depuración de la base de datos con la ayuda del método estadístico.

Tabla 1. Extracto de reporte generado por el algoritmo de aprendizaje.

Number of Architecture	Architecture ANN	Numbers of Neurons	Epoch	Root Means Square Error (RMSE)	Means Percentage Error (MPE)	R ²	Best Linear Equation
1	8--1--1	1	1000	0.554209372	17.36174307	0.9004	0.81T+0.71
2	8--2--1	2	1000	0.285012061	12.64856695	0.9502	0.88T+0.43
3	8--3--1	3	1000	0.283046879	13.77088346	0.9504	0.89T+0.04
4	8--4--1	4	1000	0.27890845	11.34057548	0.9516	0.89T+0.35
5	8--5--1	5	1000	0.213592679	11.66547187	0.9628	0.92T+0.3
6	8--6--1	6	1000	0.230296651	10.53668406	0.9599	0.94T+0.2
7	8--7--1	7	1000	0.21687368	10.19372467	0.9622	0.93T+0.27
8	8--8--1	8	1000	0.264669891	12.85161597	0.9539	0.9T+0.33
9	8--9--1	9	1000	0.275548412	11.76859909	0.9517	0.92T+0.28
10	8--10--1	10	1000	0.267190521	11.66423641	0.9535	0.94T+0.22

RESULTADOS

Los resultados obtenidos con esta arquitectura fueron satisfactorios obteniendo los siguientes parámetros:

- RMSE (): 0.2168 (Root Means Square Error).
- MPE (): 10.194%. (Percentage Error)
- Alineación: 0.93T+0.27.
- Pesos sinápticos (WI y WO).
- vías (B1 y B2)

En este ejemplo la arquitectura No. 7 exhibe la mejor correlación (0.9622). Los casos se calcularon los parámetros: la R (test), RMSE

(global y test) y la CR de las variables. El RMSE global, incluye el 100% de los datos de entrenamiento y el de test solo el 20%.

En la figura 7 se muestra la correlación lineal entre la base de datos medida y la estimada de la RNA obtenida en la prueba 7, en las tablas 2 y 3 se muestran los valores obtenidos de pesos sinápticos de la capa de entrada (WI), pesos sinápticos de la capa oculta (WO), bias de la capa de entrada (B1) y bia de la capa de salida (B2) respectivamente.

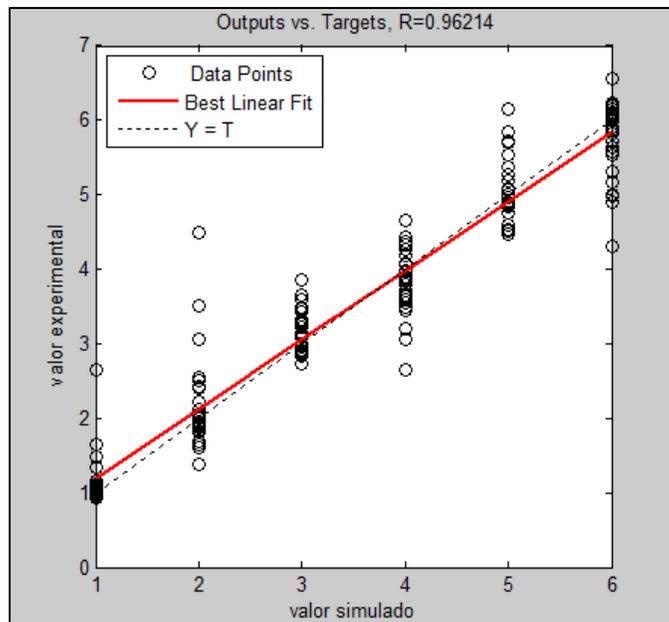


Figura 7. Gráfica de correlación lineal entre los resultados experimentales y los simulado.

Finalmente presentamos la ecuación propuesta obtenida a partir de la arquitectura #7, dada por:

$$OUT = \sum_{j=1}^7 \left[WO_{(n,j)} \left(\frac{2}{1 + e^{-2(\sum_{i=1}^8 (WI_{(j,i)} * In_{(i)}) + B2)}} - 1 \right) \right] + B1$$

DONDE:

- $WI_{(j,i)}$ = matriz de pesos sinápticos de las entradas
- $WO_{(n,j)}$ = matriz de pesos sinápticos de la capa oculta
- $In_{(i)}$ = datos de entrad.
- $B1$ = bias de la capa oculta.
- $B2$ = bia de la capa de salida.

La aplicación final de los procesos anteriores, se decidió utilizar una placa de desarrollo ARDUINO MEGA 2560, con la cual se pudo realizar una aplicación demostrativa en la cual se ingresó la ecuación propuesta de la RNA de manera simplificada. Posterior a

realizar el proceso de ingreso de los datos estos son introducidos a la ecuación de la RNA obteniendo un valor de salida el cual accionara el movimiento de un uno de los dedos del prototipo de la prótesis.

Tabla 2. Pesos sinápticos y vías de la capa oculta y de la capa de salida.

Pesos sinápticos y bias de la capa oculta y capa de salida							
Vector WO	-1.9619	1.0649	1.2106	1.004	1.8337	-1.2558	0.9283
Vector B1	6.8032	7.7847	-3.5825	3.3002	5.0344	10.8505	2.6925
Vector B2	1.4902						

Tabla 3. Pesos sinápticos de la arquitectura # 7, de las 8 entradas para las 7 neuronas de la capa oculta.

Matriz de pesos sinápticos WI							
0.8475	2.2854	2.1476	0.1976	-0.5232	-3.9432	-2.5756	-3.3085
3.7104	-9.9741	-10.029	-5.5328	-3.5506	-5.6621	-7.5008	-6.102
14.8244	13.2162	-0.6775	4.2638	-3.1937	-3.8813	-6.7063	0.6947
-7.9776	-15.0781	3.212	6.7119	1.209	1.3704	9.7667	1.7998
1.5851	0.6765	-0.6552	2.0935	-1.1748	1.1137	-1.5177	-0.7433
-3.9121	-41.0433	-11.7778	8.4399	-1.998	10.4343	-22.3041	0.3034
6.7575	3.4153	1.9836	-19.1374	1.4781	-6.1953	8.4232	-3.6328

Con la finalidad de validar el buen funcionamiento del entrenamiento y el

resultado de la red neuronal, se realizó un programa en arduino, en el cual se ingresa

una parte de la base de datos, los pesos sinápticos (WI y WO) y las bias (B1 y B2).

El programa que se desarrolló consiste en tres pasos los cuales se explicaran de manera detallada (figura 8):

- Paso uno: el programa genera un número aleatorio, dicho número representara una fila de la base de datos la cual es una matriz de 8x24 (columnas por filas), los datos que se encuentran en dicha fila se ingresaran a la función resultante del entrenamiento de la red neuronal.
- Paso dos: los datos ingresados a la función de la red neuronal dan como resultado una salida predicha de acuerdo a la fila que fue

seleccionada cada resultado obtenido generara una acción.

- Paso tres: cada acción está en función de la salida calculada, cabe mencionar que dichas salidas cuentan con un porcentaje de error de 10.94% (MPE). el orden de las acciones es el siguiente.
 - Out =1: se accionan todos los servos, imitando como se cierra el puño completo (figura 9a).
 - Out =2: se acciona el servo uno imitando la contracción del dedo índice (figura 9b).

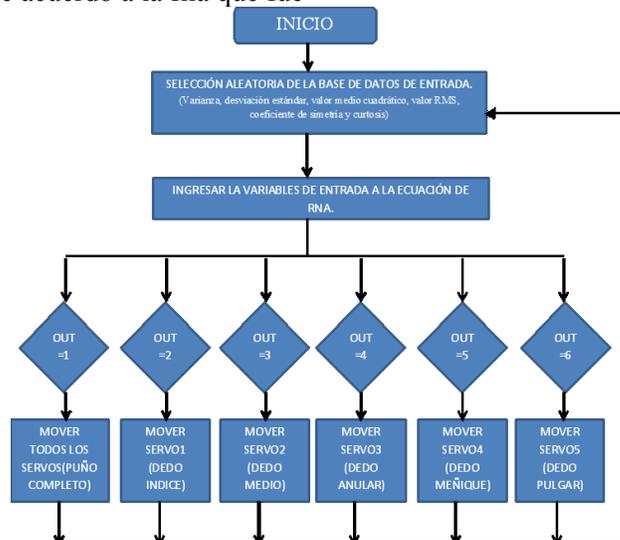


Figura 8. Diagrama de funcionamiento de aplicación de RNA en placa de desarrollo ARDUINO MEGA2560.

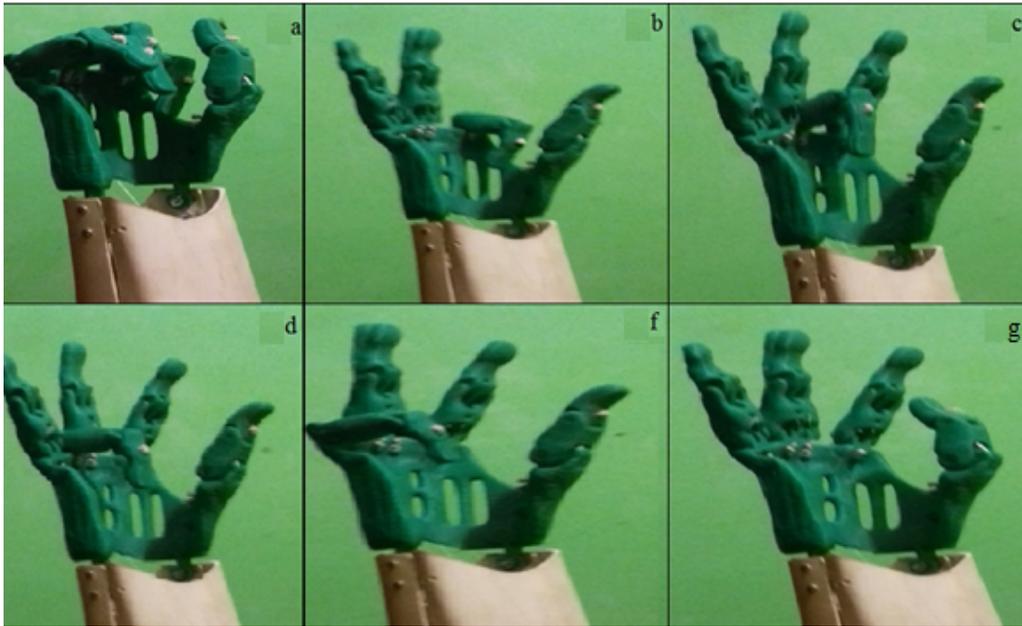


Figura 9

- a). Out = 1 puño completo cerrado
- b) Out = 2 dedo índice contraído
- c) Out = 3 dedo medio contraído
- d) Out = 4 dedo anular contraído
- e) Out = 5 dedo meñique contraído
- f) Out = 5 dedo meñique contraído
- g) Out = 6 dedo pulgar contraído.

CONCLUSIONES

A través del diseño de un sensor mioeléctrico fue posible obtener señales mioeléctricas, de la contracción de los músculos responsables de los movimientos de cada dedo de una mano humana así como desarrollar una manera de selección y caracterización de señales mioeléctricas.

- Out =3: se acciona el servo dos imitando la contracción del medio (figura 9c).
- Out =4: se acciona el servo tres imitando la contracción del dedo anular (figura 9c).
- Out =5: se acciona el servo cuatro imitando la contracción del dedo

meñique (figura 9d).

- Out =6: se acciona el servo cinco imitando la contracción del dedo pulgar (figura 9f).

Se desarrolló e implementó una arquitectura de RNA (8-7-1) multicapas

Se desarrolló e implementó una arquitectura de RNA (8-7-1) multicapas Utilizando redes neuronales podemos obtener una selección de señales or pequeño, con esto ofrecemos un buen funcionamiento del prototipo de la prótesis. Se obtuvo una ecuación derivada del entrenamiento de la RNA, dicha ecuación fue validada y probada obteniendo resultados satisfactorios.

REFERENCIAS

A Bassam, RA Conde-Gutiérrez, J Castillo, G Laredo, JA Hernández, 2014, Direct neural network modeling for separation of linear and branched paraffins by adsorption process for gasoline octane number improvement, "Fuel". Vol. 124, 158-167.

A. Bassam, E. Santoyo, J. Andaverde, J. A. Hernández, O. M. Espinoza-Ojeda, 2010, Estimation of static formation temperatures in geothermal wells by using an artificial neural network approach, "Computers & Geosciences", vol. 32, 1191-9.

AlibakhshKasaeian, Mohammad Sameti, 2014, Mini Review on the Application of Neural Networks in Solar Energy Conversion, "Journal of Middle East Applied Science and Technology (JMEAST)", ISSN (Online): 2305-0225.

Alonso, R. Hornero, P. Espino, R. de la Rosa, L. Liptak, (2001), Entrenador Mioeléctrico de Prótesis para Amputados de Brazo y Mano. Universidad de Valladolid.

C. P. Pedro, 2010, inteligencia artificial con aplicaciones en ingeniería, México D.F., "Alfaomega".

H. Demuth, M. Beale & M. Hagan, 2007, Neural Network Toolbox 5: User's Guide, "The Math Works Inc".

N. López, F. di Sciasco, E. Orosco, N. Ledesma, A. Echenique, & M. Valentinuzzi, (2006), Control Mioeléctrico de un brazo robótico, IV Congreso Iberoamericano sobre Tecnologías de Apoyo a la Discapacidad. "Vittoria-Espiritu Santo-Brasil": ISBN: 8496023451.

P. Parimal, S. Allen, & G. Rapach, (1988). Microcontroller based prosthetic device using myoelectric signal. Engineering in Medicine and Biology Society, "Proceedings of the Annual International Conference of the IEEE", 1427 – 1428.

V. I. Pedro & G.L. Inés, 2004, Redes de Neuronas Artificiales, Un enfoque Práctico, Madrid España, "Pearson Education".

V. V. A. Sergio, 2017, Adquisición, Procesamiento e Interpretación de Señales Mioeléctricas para el Diseño y Fabricación Robótica, tesis de licenciatura, Universidad Autónoma del Estado de Morelos, Cuernavaca.

Este documento debe citarse como: Vicario Vazquez, S. A., Oubram, O., Bassam, A., Velazquez Aguilar, J. G., Ordoñez Lopez, E. (2017). **Sistema de reconocimiento inteligente de señales mioeléctricas del movimiento de mano humana**. Ingeniería, Revista Académica de la FI-UADY, 21-2, pp. 41-53, ISSN 2448-8364.