

Aprendizaje Profundo: De Perceptrones a Redes Profundas

Introducción

El perceptrón multicapa es una red neuronal artificial (RNA) formada por múltiples capas

Entrenando el Perceptrón

El entrenamiento del perceptrón consiste en alimentarlo con múltiples muestras de entrenamiento y el cálculo de la salida para cada uno de ellos

Redes Neuronales Feedforward para el Aprendizaje Profundo

Una red neuronal es en realidad una composición de perceptrones conectados de diferentes maneras, y que operan con diferentes funciones de activación.

Algoritmo para desarrollar perceptrones

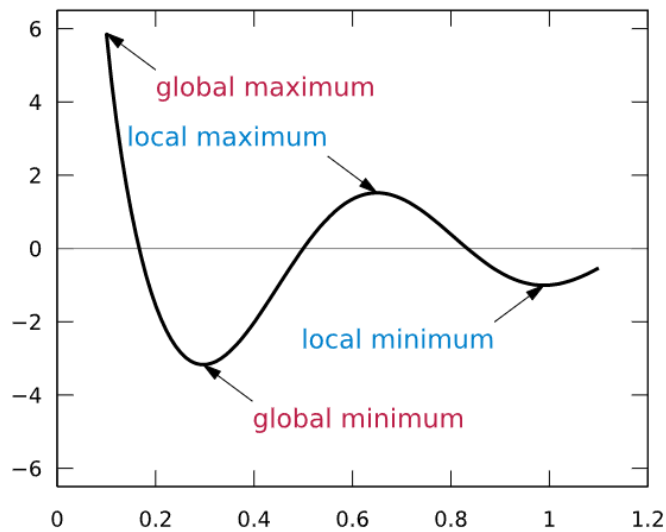
El algoritmo de aprendizaje profundo más común para entrenamiento supervisado de los perceptrones multicapa se conoce como propagación reversa. El procedimiento básico:

Una muestra de entrenamiento se presenta y se propaga hacia adelante a través de la red.

Se calcula el error de salida, por lo general el error cuadrático medio:

$$E = \frac{1}{2}(t - y)^2$$

- Donde t es el valor objetivo e y es la salida real de la red. Otros cálculos de error también son aceptables, pero el MSE es una buena opción.
- Los errores de red se reducen usando un método llamado descenso de gradiente estocástico.



Capa Oculta

La capa oculta es de particular interés. Por el teorema de aproximación universal, una única red de capa oculta con un número finito de neuronas puede ser entrenada para la aproximación de una función arbitraria al azar

- ▶ **La capa oculta es donde la red guarda la representación abstracta interna de los datos de entrenamiento.**
- ▶ La capa oculta es donde la red guarda la representación abstracta interna de los datos de entrenamiento, similar a la forma en que un cerebro humano (analogía muy simplificada) tiene una representación interna del mundo real. En adelante en el tutorial, vamos a ver diferentes formas de jugar un poco con la capa oculta.

Un Ejemplo de Red

Se puede ver una red simple (4-2-3 capa) de feedforward neural que clasifica el conjunto de datos IRIS, implementado en Java aquí, a través del método `testMLPSigmoidBP`. El conjunto de datos contiene tres clases de plantas de iris con características como la longitud sépalo, longitud pétalo, etc. Se proporciona a la red 50 muestras por clase. Las características se sujetan a las unidades de entrada, mientras que cada unidad de salida corresponde a una única clase del conjunto de datos: “1/0/0” indica que la planta es de clase Cetosa, “0/1/0” indica versicolor, y “0/0/1” indica Virginica. El error de clasificación es 2/150 (es decir, no clasifica correctamente 2 muestras de 150).

El Problema con las Grandes Redes

Una red neuronal puede tener más de una capa oculta: en ese caso, las capas superiores están “construyendo” nuevas abstracciones en la parte superior de las capas anteriores. Y como hemos mencionado antes, a menudo se puede aprender mejor en la práctica con las redes más grandes.

Sin embargo, el aumento del número de capas ocultas conduce a dos problemas conocidos:

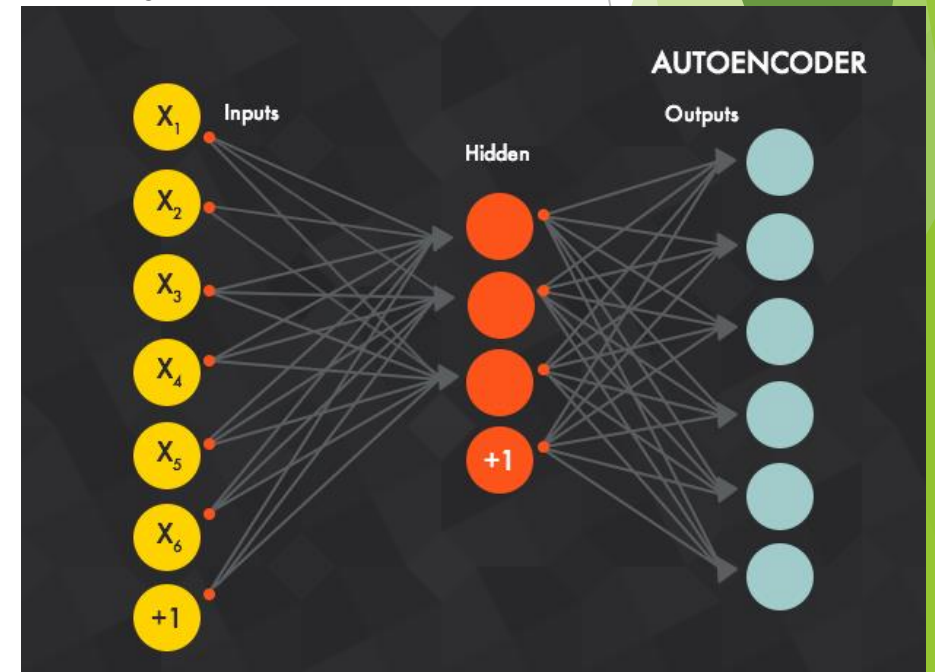
1-Desaparición del Gradiente: a medida que añadimos más y más capas ocultas, la propagación hacia atrás se vuelve cada vez menos útil en la transmisión de información a las capas inferiores. En efecto, como la información se pasa de nuevo, los gradientes comienzan a desaparecer y se hacen más pequeños en relación con los pesos de las redes.

2-Sobreajuste: tal vez éste sea el problema central en el aprendizaje automático. En pocas palabras, el sobreajuste describe el fenómeno de ajuste de los datos de entrenamiento demasiado cerca, tal vez con la hipótesis de que son muy complejos. En tal caso, el alumno termina el montaje de los datos de entrenamiento muy bien pero se desarrollará muy mal en ejemplos reales.

Autoencoder

La mayoría de las clases introductorias de aprendizaje automático tienden a terminar con las redes neuronales feedforward. Pero el espacio de posibles redes es mucho más rico, así que continuemos.

Un autoencoder es típicamente una red neuronal feedforward, que tiene como objetivo aprender una representación (codificación) comprimida y distribuida de un conjunto de datos.



De Vuelta al Aprendizaje de Máquina

En esencia, nuestras dos unidades ocultas han aprendido una representación compacta del conjunto de datos de los síntomas de la gripe. Para ver cómo esto se relaciona con el aprendizaje, volvemos al problema de sobreajuste. Mediante el entrenamiento de nuestra red para aprender una representación compacta de los datos, estamos dando preferencia a una representación más simple en lugar de una hipótesis altamente compleja que sobre-ajusta los datos de entrenamiento.

La Enfermedad de la Gripe

Para demostrar aún más autoencoder, vamos a ver una aplicación más. En este caso, vamos a utilizar un conjunto de datos simple, que consiste en síntomas de la gripe

He aquí cómo el conjunto de datos se desglosa:

Hay seis entidades de entrada binarias.

Los tres primeros son los síntomas de la enfermedad. Por ejemplo, 1 0 0 0 0 0 indica que este paciente tiene temperatura alta, mientras que 0 1 0 0 0 0 indica tos, 1 1 0 0 0 0 indica la tos y alta temperatura, etc.

Las tres últimas características son síntomas “de venta libre”; cuando un paciente tiene uno de estos, es menos probable que él o ella esté enfermo/a. Por ejemplo, 0 0 0 1 0 0 indica que este paciente tiene una vacuna contra la gripe. Es posible tener combinaciones de los dos conjuntos de características: 0 1 0 1 0 0 indica un paciente vacunado con tos, y así sucesivamente.

Vamos a considerar que un paciente está enfermo. Cuando él o ella tiene al menos dos de las tres primeras características, y saludable si él o ella tiene al menos dos de las tres últimas (con desempates que se rompen a favor de los pacientes sanos), por ejemplo:

111000, 101000, 110000, 011000, 011100 = enferma

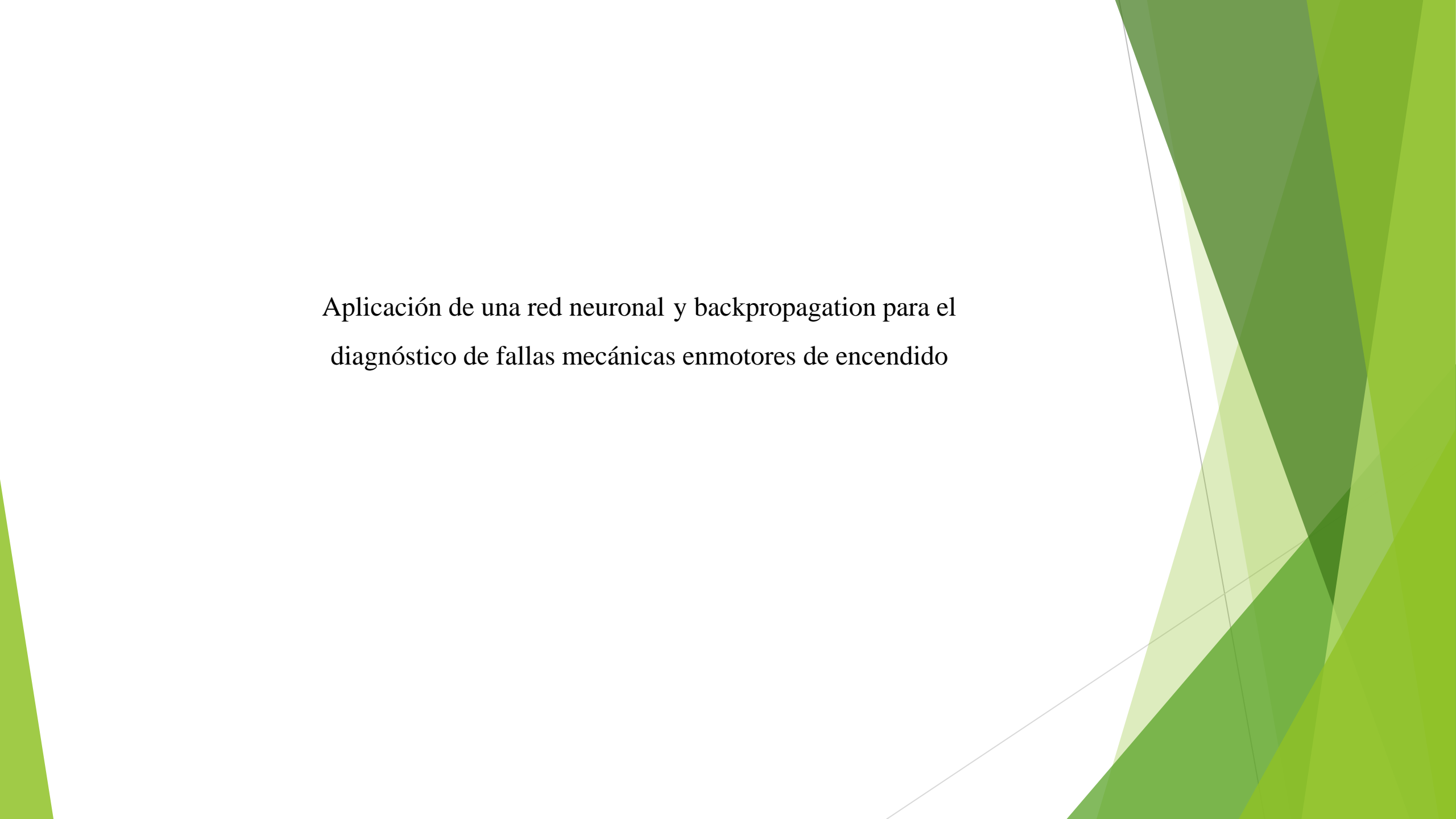
000111, 001110, 000101, 000011, 000110 = sano

Redes Profundas

Hemos demostrado ahora que las capas ocultas de autoencoder y las MBR actúan como detectores de características eficaces; pero es raro que podamos utilizar estas funciones directamente. De hecho, el conjunto de datos anterior, es más una excepción que una regla. En su lugar, tenemos que encontrar la manera de utilizar estas características detectadas, indirectamente.

Conclusión

El objetivo de este paper de aprendizaje profundo de Java era darle una breve introducción al campo de los algoritmos de aprendizaje profundo, comenzando con la unidad más básica de la composición (el perceptrón) y avanzando a través de diversas arquitecturas eficaces y populares, como la de máquina Boltzmann restringida.

The background of the slide features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, ranging from light lime to dark forest green. These shapes are positioned on the right side and bottom of the slide, creating a modern, dynamic look.

Aplicación de una red neuronal y backpropagation para el diagnóstico de fallas mecánicas en motores de encendido

Introducción

- **En la presente investigación se explica la metodología para la creación de un sistema de diagnóstico aplicado a la detección de fallas mecánicas en vehículos con motores a gasolina mediante redes neuronales artificiales, el sistema se basa en el estudio de la fase de admisión del ciclo Otto, el cual es registrado a través de la implementación física de un sensor MAP (*Manifold Absolute Pressure*).**

Red de elementos o unidades de procesamiento simples EP/UP en español o PE/PU en inglés), interconectados entre sí mediante conexiones sinápticas en las que cada conexión tiene un peso (fuerza) que se ajusta a partir de la experiencia (datos).

Métodos y materiales

En la presente sección se desarrollan las temáticas principales que tienen que ver con la selección de unidad experimental e instrumentación mínimamente invasiva, condiciones de toma de muestras, metodología para la adquisición de datos

Selección de la unidad experimental e instrumentación mínimamente invasiva

Se toma en consideración como principal objetivo evitar el desarmado de elementos y piezas del motor para diagnosticar fallas mecánicas, por lo que se determina medir la depresión del motor mediante la instalación de un sensor tipo MAP en una toma de vacío del múltiple de admisión,

Metodología para la obtención de datos

En la Figura 4 se presentan los elementos físicos necesarios para el muestreo de señales. Figura 4

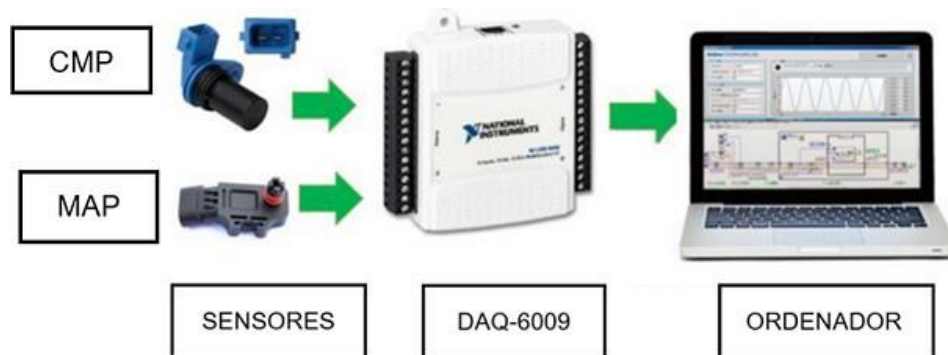


Figura 4. Elementos necesarios para la toma de muestras.

Para obtener los datos de las señales de los sensores MAP y CMP se sigue el flujograma de proceso que se presenta en la Figura 4.

Condiciones de toma de muestras

Se utiliza el *software* Labview 2017 junto con una tarjeta Ni DAQ-6009, para la obtención de muestras. Las muestras se adquieren en ralentí aproximadamente a 850 rpm con un rango de temperatura del MEP de entre 92 °C y 97 °C, carga del motor de 40 % y se aplica el escáner automotor para corroborarlas. Según un estudio preexperimental se determina que la señal del sensor MAP posee picos de mayor frecuencia, de tal manera que se realiza la toma de muestras a una velocidad de 10 KHz en un tiempo de 5 segundos para cada una de las señales, dicha velocidad supera el criterio de Nyquist (1.416 KHz).

Resultados y discusión

A fin de comparar el correcto funcionamiento del sistema de diagnóstico se procede a realizar varias pruebas bajo diversas condiciones de funcionamiento.

En este apartado se presentan dos condiciones de falla en específico: el inyector 1 (200) y el fallo en bobina 2-3 (1100).

RNA. Una vez normalizada la matriz de atributos

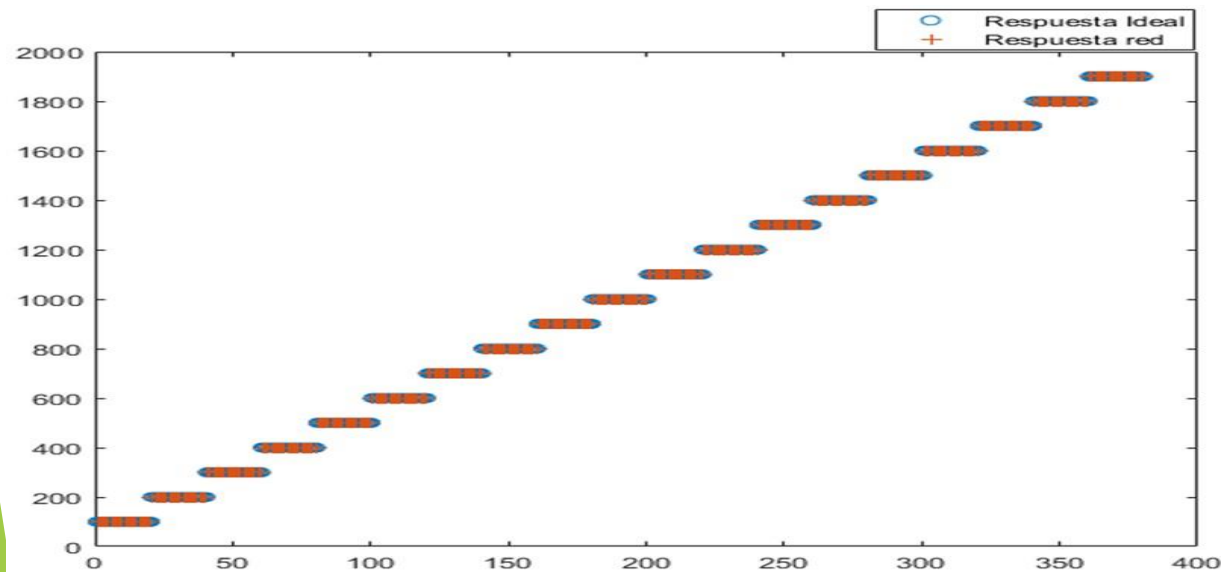


Figura 5. Red neuronal con porcentaje de error de $1.89\text{e}-11$, con función de entrenamiento «trainscg».

se procedió a la creación de la RNA.

del motor ($720^\circ \pm 180^\circ$), con el reglaje de distribución de adelanto a la apertura de admisión (AAA) y retraso de cierre de admisión (RCA), correspondiente a cada uno de los cilindros, del cual se realiza un ventaneo y se obtiene la caracterización de la señal.

Reducción de atributos

Para la selección y reducción del número de atributos se analiza la matriz general a través de 3 métodos estadísticos: ANOVA, matriz de correlación y Random Forest.

Selección de atributos para entrenamiento de la RNA

Para seleccionar los atributos que serán considerados en la entrada de la red neuronal se realizó un análisis de coincidencia en la Tabla 2 en el cual se seleccionaron los atributos que más se repiten entre los resultados de cada método estadístico aplicado. Los atributos que más se repiten se muestran en la Tabla 3, corroborando la efectividad de cada método utilizado.

Conclusiones

El modelo de red neuronal desarrollado posee un error de clasificación de $1,89e-11$ con la función de entrenamiento `trainscg`, lo que permitió la identificación precisa de los diferentes tipos de condiciones mecánicas del MEP, por lo que constituye una alternativa claramente viable para ser integrada en un sistema de diagnóstico, debido a la rapidez computacional que ofrecen las redes neuronales artificiales.