AUTOMATISME I CONTROL ELÈCTRIC

C/ Samontà, 22
Pol. Ind. Fontsanta
08970 Sant Joan Despí
BARCELONA
T. +34 93 477 62 62
acesa@acesa.es
www.acesa.es

Regulación de la temperatura a partir de redes neuronales

Las redes neuronales son una herramienta muy útil para regular la temperatura en diferentes aplicaciones, desde sistemas de climatización hasta procesos industriales. Una red neuronal puede ser entrenada para aprender la relación entre las variables de entrada (como la temperatura actual, la humedad, etc) y la variable de salida (la temperatura deseada). Una vez entrenada, la red puede predecir la temperatura deseada en función de las condiciones actuales y ajustar los controles del sistema para alcanzar esta temperatura.
Las ventajas de las redes neuronales son múltiples ya que son capaces de aprender relaciones no lineales y complejas, lo que las hace adecuadas para sistemas con comportamientos dinámicos y, una vez entrenadas, las redes neuronales pueden realizar predicciones rápidamente, lo que permite un control en tiempo real.
El Departamento Técnico de ACE Automatisme i Contro, Elèctric ha realizado un detallado estudio sobre la Regulación de la temperatura a partir de redes neuronales para la implementación de un control de temperatura con un PLC que se expone a continuación.

Objetivos del estudio

Dado que la regulación de la temperatura en la industria es un elemento de uso muy común pero que nos puede complicar en función de los sistemas, vemos útil la creación de un programa o software que pueda auto aprender.
De este modo, se minimizaría el tiempo para regular los parámetros del PID y los posibles cambios ambientales que afecten al sistema.
El objetivo de este estudio es crear una red neuronal artificial para la implementación de un control de temperatura con un PLC. Esta red estará integrada en un bloque de funciones que sea escalable a otros dispositivos.

Datos

Las pruebas se harán con un simulador que cuenta con una sonda y una resistencia calefactora. De este modo, extraeremos los datos de temperatura de la sonda y actuaremos sobre la resistencia calefactora.

Redes neuronales artificiales

Las redes neuronales consisten en una simulación de las propiedades observadas en los sistemas neuronales biológicos a través de modelos matemáticos recreados mediante mecanismos artificiales.
El funcionamiento de una neurona consiste en multiplicar los valores de entrada (x1, x2, x3...) por sus respectivos pesos (w1, w2, w3...) y sumarlos. Este resultado se multiplica por una función de activación que dependerá del tipo de resultado que necesitamos.

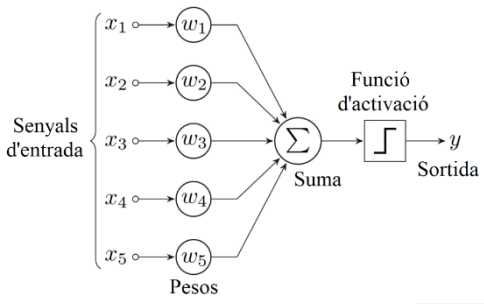


Figura1. Neurona

$$X * W^t = (x_1, x_2, \dots x_n) * \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{pmatrix} = \sum_{i=1} x_i * w_i$$

C/ Samontà, 22
Pol. Ind. Fontsanta
08970 Sant Joan Despí
BARCELONA
T. +34 93 477 62 62
acesa@acesa.es
www.acesa.es

Una red neuronal artificial tiene tres capas en su forma más básica. La información fluye de una capa a otra como lo haría nuestro cerebro. Las diferentes capas son:
Capa de entrada: es el punto de entrada de los datos de nuestro sistema. Habrá tantas como entradas se tenga de información.
Capa oculta: son las capas donde se procesa la información. Como más compleja sea la red, más capas y más neuronas hay a esta capa.
Capa de salida: es la capa donde se obtiene el resultado de salida de todos los cálculos.
Las neuronas se van conectando entre ellas y van formando una red como la de la figura2

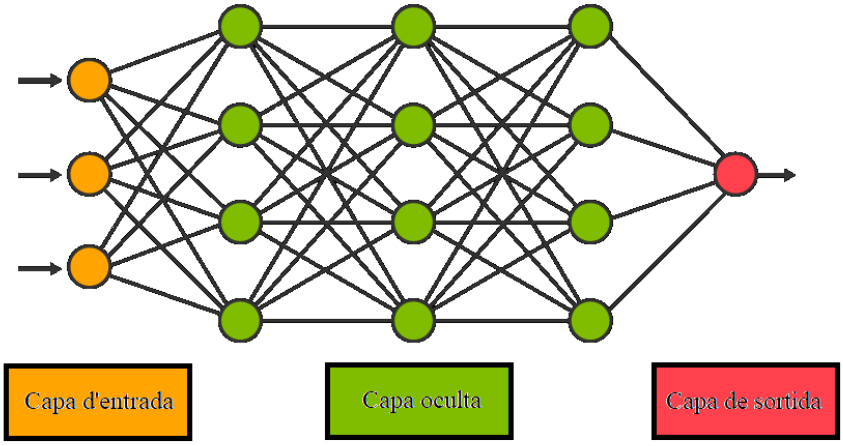


Figura 2. Red neuronal

Modos de aprendizaje del sistema
El desarrollo del aprendizaje automático se centra en tres categorías principales: aprendizaje supervisado, aprendizaje no supervisado y aprendizaje profundo. Para nuestro trabajo nos fijaremos en el primero y el último.
Aprendizaje supervisado
Este proceso de generación de conocimiento se realiza con un grupo de ejemplos o datos etiquetados en los que los resultados son conocidos previamente. Este tipo de modelo aprende de estos resultados e incorpora ajustes en los parámetros interiores para poder adaptarse a datos nuevos que se introduzcan en el sistema.
Mediante este aprendizaje se alimenta un conjunto de resultados que permite realizar predicciones adecuadas del comportamiento de los datos nuevos que todavía no han sido procesados.

Aprendizaje reforzado
Tiene como finalidad construir modelos que aumentan el rendimiento aprendido como base de un resultado o la recompensa que se genera a partir de cada iteración realizada. Este modelo utiliza la recompensa como parámetro de ajuste para acciones futuras, de tal forma que la nueva acción cumpla con la acción correcta. De este modo, por cada proceso que realiza se puede ver si ha mejorado o no en función del proceso anterior y tener en cuenta este dato.

Inicio del proceso
Se propone iniciar con un aprendizaje reforzado. Esto puede ser lo más adecuado puesto que tenemos acceso al error actual que hay entre la temperatura de consigna

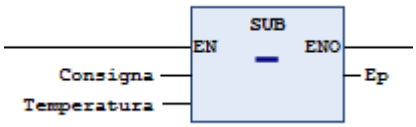
AUTOMATISME I CONTROL ELÈCTRIC

C/ Samontà, 22
Pol. Ind. Fontsanta
08970 Sant Joan Despí
BARCELONA
T. +34 93 477 62 62
acesa@acesa.es
www.acesa.es

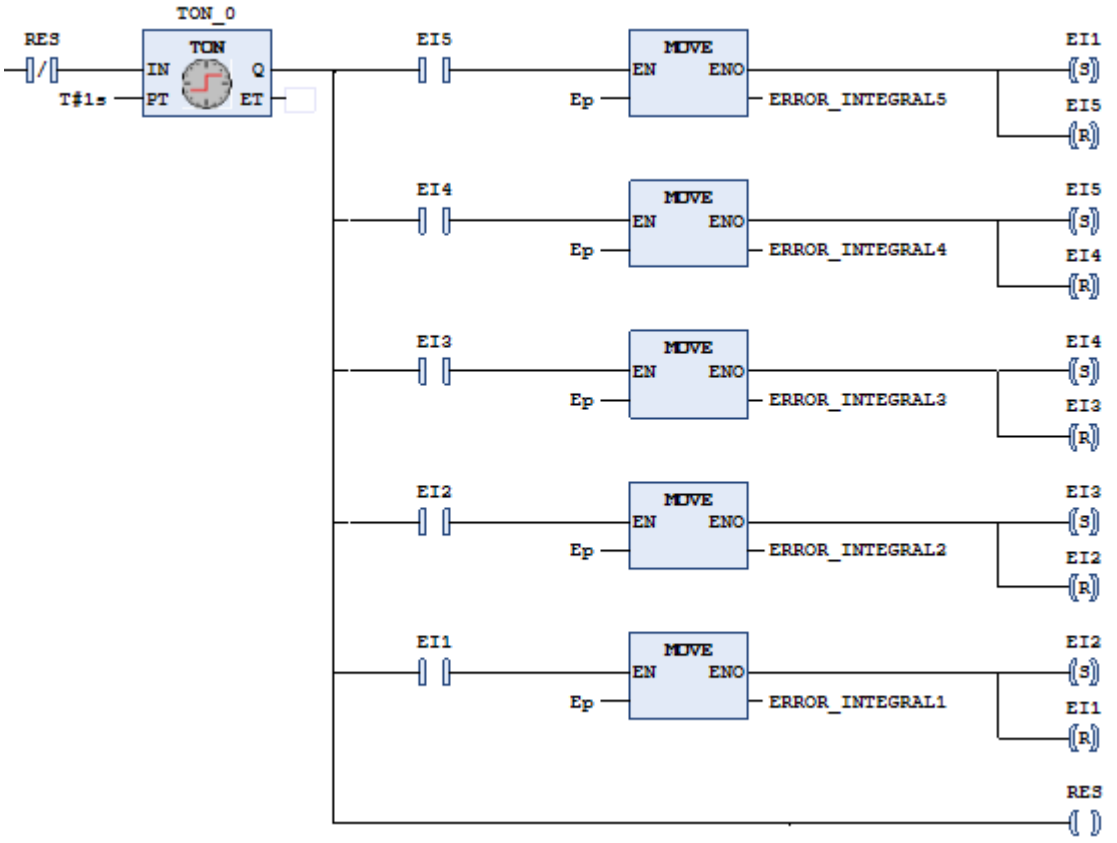
y la temperatura actual. De este modo, se pueden hacer los cálculos del error proporcional, integral y derivativo.

Obtención de los datos
El trabajo empezará con una red neuronal artificial de tres neuronas de entrada (Ep, Ei y Ed), tres neuronas ocultas y una de salida.
Los primeros datos que tendremos del proceso será la temperatura de consigna y la temperatura actual. Trabajando con estos datos obtendremos el error proporcional, el error integral y el error derivativo. Estos tres datos serán las entradas de nuestra red neuronal.

Error proporcional (Ep)
Lo obtendremos de la resta entre los valores de la consigna y la temperatura

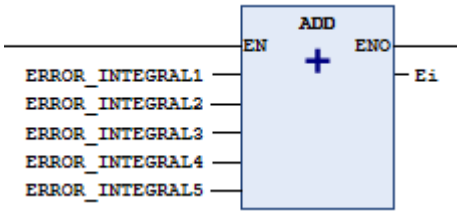


Error integral (Ei)
Para obtener el error integral, sumaremos los valores del error proporcional de los últimos 5 segundos.



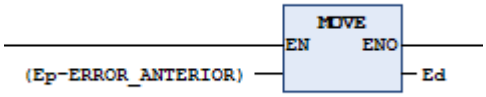
AUTOMATISME I CONTROL ELÈCTRIC

C/ Samontà, 22
Pol. Ind. Fontsanta
08970 Sant Joan Despí
BARCELONA
T. +34 93 477 62 62
acesa@acesa.es
www.acesa.es



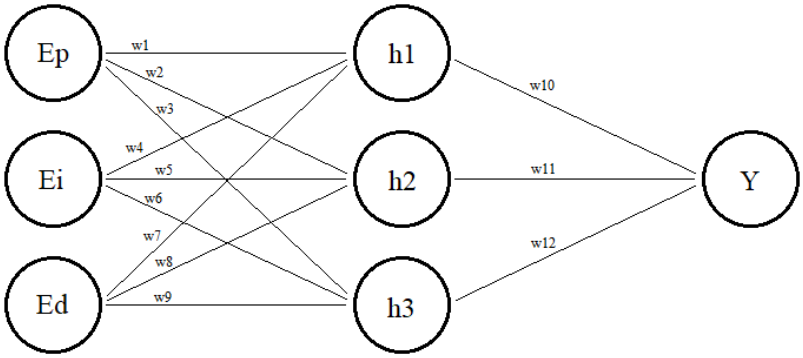
Error derivativo (Ed)

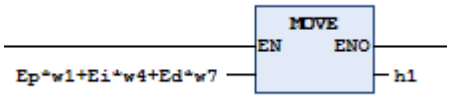
Calcularemos el valor de la pendiente que hay durante un segundo. Para hacerlo restaremos al error proporcional actual, el valor del error proporcional de hace un segundo.

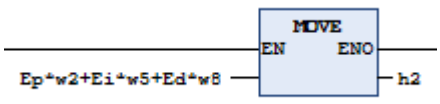


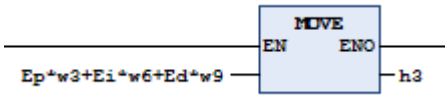
Procesamiento de cálculos neuronales

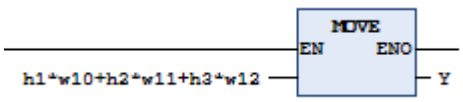
El cálculo de la red neuronal seguirá el procedimiento mostrado a la figura1.











AUTOMATISME I CONTROL ELÈCTRIC

C/ Samontà, 22
Pol. Ind. Fontsanta
08970 Sant Joan Despí
BARCELONA
T. +34 93 477 62 62
acesa@acesa.es
www.acesa.es

Siguiendo con los objetivos anteriores, se quiere crear un programa de autoaprendizaje para hacer un control de temperatura parecida a un PID. De este modo reducir el tiempo para encontrar los parámetros adecuados y hacer que varíen en función de variaciones del entorno que rodea la máquina.
Se buscará crear una red neuronal artificial para la implementación de un control de temperatura con un PLC. Esta red estará integrada a un bloque de funciones que sea escalable a otros dispositivos.
En este estudio seguiremos los siguientes pasos para la implementación de un controlador PID utilizando técnicas de aprendizaje automático

1. **Recopilación de datos:** Primero, será necesario recopilar datos del sistema actual. Esto incluye las entradas en el controlador como el error entre el valor deseado y el valor actual, las salidas del controlador y cualquier otro variable relevante del sistema.
2. **Procesamiento de datos**: A continuación, hay que procesar los datos para adecuarlas para el aprendizaje automático. Esto puede incluir la normalización de los datos, cálculos iniciales, la eliminación del ruido y la identificación de características relevantes.
3. **Entrenamiento del modelo**: A continuación, hay que entrenar un modelo de aprendizaje automático con estos datos. Por eso existen diferentes modelos, como las redes neuronales, los árboles de decisión o los algoritmos de regresión.
4. **Implementación del modelo**: Una vez el modelo esté entrenado y validado, se puede implementar a un PLC para ajustar los parámetros del PID a tiempo real.

1. Recopilación de datos

Este primer ensayo se hará con un kit de temperatura que consta de una resistencia calefactora y una sonda de temperatura PT100. Los datos y los cálculos se obtendrán a partir de un PLC.

Los datos que obtendremos en esta prueba, sueño la temperatura actual (temperatura_act) y la temperatura a la que se quiere llegar (consigna). A partir de estos datos procesaremos los cálculos correspondientes para encontrar los errores proporcional, integral y derivativo para que sean la entrada de nuestra red neuronal.

2. Procesamiento de datos

Se hará los cálculos para poder obtener la información del error proporciona, integral y derivativo del sistema.

Error proporcional

Obtendremos el valor del error proporcional a partir de la diferencia de entre la temperatura a la que queremos llegar y la temperatura actual

$$u(t) = e(t)$$

$$Ep = consigna - temperatura_{act}$$

AUTOMATISME I CONTROL ELÈCTRIC

C/ Samontà, 22
Pol. Ind. Fontsanta
08970 Sant Joan Despí
BARCELONA
T. +34 93 477 62 62
acesa@acesa.es
www.acesa.es

Error integral

El error integral es el sumatorio del error proporcional. Nosotros haremos un sumatorio del valor actual del error proporcional y de los últimos 4. Estos se tomarán en intervalos de 1 segundo

$$u(t) = \int_0^t e(\tau)\, d\tau$$

$$Ei = \sum Ep_{act} + Ep_{act-1} + Ep_{act-2} + Ep_{act-3} + Ep_{act-4}$$

Error derivativo

El error derivativo es la pendiente que genera el error proporcional en función del tiempo. Nosotros calcularemos esta pendiente restando el error proporcional actual, el valor del error proporcional es la media de la diferencia de error de los últimos 5 segundos.

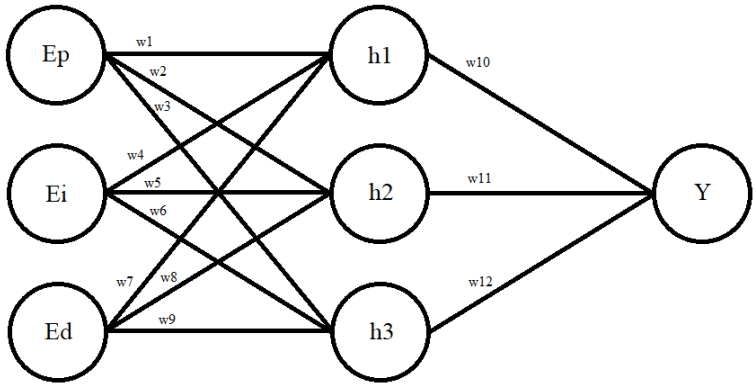
$$u(t) = \frac{de(t)}{dt}$$

$$Ed = \frac{Ep_{act} - Ep_{-4}}{5}$$

Teniendo ya los datos de entrada en la red neuronal, hay que empezar a desarrollar el cálculo de la salida. La siguiente imagen muestra el desarrollo de la red. Ésta solo tendrá una capa oculta y solo 3 neuronas debido a que el cálculo de la red crece de forma exponencial por cada capa oculta y por cada neurona que se añade. Las próximas pruebas se harán con más capas y más neuronas. La salida será el porcentaje de salida del PID. El autoaprendizaje de este estudio será semi-manual. Esto quiere decir que durante la primera regulación le mostraremos al sistema qué es el porcentaje de salida que esperamos en diferentes punto del proceso. De este modo durante este proceso, el sistema irá haciendo los cálculos para hacer el autoaprendizaje de cada momento en función de la entrada del sistema.

AUTOMATISME I CONTROL ELÈCTRIC

C/ Samontà, 22
Pol. Ind. Fontsanta
08970 Sant Joan Despí
BARCELONA
T. +34 93 477 62 62
acesa@acesa.es
www.acesa.es



Para obtener el cálculo de la salida en función de la entrada, primero tenemos que calcular los pasos intermedios h1, h2 y h3. Se calcula de la siguiente manera:

$$h1 = Ep * w1 + Ei * w4 + Ed * w7$$

$$h2 = Ep * w2 + Ei * w5 + Ed * w8$$

$$h3 = Ep * w3 + Ei * w6 + Ed * w9$$

Del mismo modo obtendremos la salida Y.

$$Y = h1 * w10 + h2 * w11 + h3 * w12$$

El valor de las variables w1-12 será definido al inicio del cálculo con un valor aleatorio de entre 0 y 1 para dar un número aleatorio en la salida.

El error de la salida Y (predicción) en función del valor esperado (actual) se calculará con la función del error medio cuadrático. Con la siguiente fórmula:

$$Error = \frac{1}{2}(predicció - actual)^2$$

La retropropagación, nos ayuda a modificar las variables w_x para adecuarlas al error de la salida a partir del descenso del gradiente y hacer así que la red neuronal evolucione y aprenda del error de la salida.
El descenso del gradiente es un algoritmo para encontrar el mínimo en una función y esto nos permite minimizar el error que tenemos en la salida. Para encontrar el mínimo de una función utilizaremos la fórmula:

$$^{*}w_x = w_x - a(\frac{\partial Error}{\partial w_x})$$

Por ejemplo, al actualizar w_12, tomamos la rama w_12 y restamos la derivada parcial del error en función de w_12. Esta estará multiplicar por la variable a (tasa de aprendizaje) Esta variable hará que el proceso de aprendizaje sea más rápido o más lento.

AUTOMATISME I CONTROL ELÈCTRIC

C/ Samontà, 22
Pol. Ind. Fontsanta
08970 Sant Joan Despí
BARCELONA
T. +34 93 477 62 62
acesa@acesa.es
www.acesa.es

$$^{*}w_{12} = w_{12} - a\left(\frac{\partial Error}{\partial w_{12}}\right)$$

La derivación de la función del error se calcula usando la regla de la cadena de este modo:

$$\frac{\partial Error}{\partial w_{12}} = \frac{\partial \frac{1}{2}(predicció - actual)^2}{\partial predicció} * \frac{\partial predicció}{\partial w12}$$

$$\frac{\partial Error}{\partial w_{18}} = (predicció - actual) * \frac{\partial h1 * w10 + h2 * w11 + h3 * w12}{\partial w12}$$

$$\frac{\partial Error}{\partial w_{12}} = (predicció - actual) * h3$$

Del mismo modo encontraremos los valores de w_{11} i w_{10}

$$^{*}w_{12} = w_{12} - a(predicció - actual) * h3$$
$$^{*}w_{11} = w_{11} - a(predicció - actual) * h2$$
$$^{*}w_{10} = w_{10} - a(predicció - actual) * h1$$

Pero, al retroceder para actualizar w_{1-9} la regla de la cadena quedará así:

$$\frac{\partial Error}{\partial w_9} = \frac{\partial \frac{1}{2}(predicció - actual)^2}{\partial predicció} * \frac{\partial predicció}{\partial h3} * \frac{\partial h3}{\partial w_9}$$

$$\frac{\partial Error}{\partial w_9} = (predicció - actual) * w_{12} * Ed$$

$$^{*}w_9 = w_9 - a(predicció - actual) * w_{12} * Ed$$
$$^{*}w_8 = w_8 - a(predicció - actual) * w_{11} * Ed$$
$$^{*}w_7 = w_7 - a(predicció - actual) * w_{10} * Ed$$
$$^{*}w_6 = w_6 - a(predicció - actual) * w_{12} * Ei$$
$$^{*}w_5 = w_5 - a(predicció - actual) * w_{11} * Ei$$
$$^{*}w_4 = w_4 - a(predicció - actual) * w_{10} * Ei$$
$$^{*}w_3 = w_3 - a(predicció - actual) * w_{12} * Ep$$
$$^{*}w_2 = w_2 - a(predicció - actual) * w_{11} * Ep$$
$$^{*}w_1 = w_1 - a(predicció - actual) * w_{10} * Ep$$