



# Propuesta de metodología para el diagnóstico de fallos basado en árboles de decisión y lógica difusa.

*Amanda Martí Coll, Adrian Milberberg Rodríguez, David Manresa Gutiérrez, Alberto Sergio Prieto Moreno, Orestes Llanes-Santiago*

## **RESUMEN / ABSTRACT**

La presente investigación tiene como objetivo presentar una propuesta de metodología para el diagnóstico de fallos en la industria basado en árboles de decisión y lógica difusa. Los árboles de decisión tienen entre sus ventajas la sencillez y facilidad de interpretación. Mientras que, la lógica difusa, brinda capacidades para el manejo de la incertidumbre del conocimiento aportado por los expertos del proceso. Esta nueva metodología evita tener que contar con una gran base de datos de entrenamiento durante el proceso de desarrollo del diagnosticador y en su lugar usar una tabla de síntomas de fallos. Además, se hace uso del estándar PMML para transmitir la información del modelo obtenido en la aplicación de configuración a la aplicación que realiza las tareas de monitoreo en línea. La efectividad de la metodología se analiza en un caso real, específicamente en el diagnóstico de fallos de un proceso de tratamiento de agua para inyección de la industria farmacéutica.

Palabras claves: diagnóstico de fallos, árboles de decisión, lógica difusa, estándar PMML, propuesta de metodología

*The objective of this research is to present a methodology proposal for the diagnosis of failures in the industry based on decision trees and fuzzy logic. Decision trees have among their advantages simplicity and ease of interpretation. While fuzzy logic provides capabilities for managing the uncertainty of the knowledge provided by process experts. This new methodology makes the contribution of offering a method to avoid the need to have a large training database during the development process for the diagnosticator, which will imply the use of a fault symptom table. In addition, the PMML standard is used to transmit the information of the model obtained in the configuration application to the application that performs the online monitoring tasks. The effectiveness of the methodology is analyzed in a real case, specifically in the diagnosis of failures of a water treatment process for injection in the pharmaceutical industry.*

**Keywords:** *fault diagnosis, decision trees, fuzzy logic, PMML standard, methodology proposal*

**Methodology proposal for fault diagnosis based on decision trees and fuzzy logic.**

## **1. - INTRODUCCIÓN**

En los sistemas industriales actuales es una necesidad aumentar la productividad, la eficiencia, la calidad de los productos y satisfacer, las cada vez más rigurosas, regulaciones de seguridad y medioambientales. Para contribuir con esos propósitos a los sistemas industriales le son incorporadas capacidades de monitoreo de sus estados que permiten el desarrollo de sistemas de diagnóstico de fallos. Estos sistemas tienen como objetivo detectar, identificar y localizar de manera temprana los fallos que se produzcan para tratar de evitar un comportamiento no adecuado del proceso [1].

A pesar de los grandes avances en la tecnología, el diagnóstico de fallos sigue constituyendo un reto; razón por la cual actualmente sigue siendo un tema ampliamente investigado [2-7]. Las técnicas para el diagnóstico de fallos se pueden dividir en dos grandes grupos: basadas en modelos y basadas en datos [7,8]. No obstante, en algunos trabajos, como por ejemplo [9],

Recibido: 10/06/2022

Aceptado: 15/08/2022

también mencionan el enfoque del análisis de señales como otro grupo. Hay que destacar que los métodos basados en modelos y análisis de señales tienen entre sus desventajas que requieren del conocimiento previo de las respuestas del sistema ante los fallos, así como la obtención de un modelo preciso del proceso. Además, generalmente estos se diseñan para un sistema en específico, lo que dificulta realizar modificaciones para la incorporación de aquellos fallos que posteriormente se deseen diagnosticar. Sin embargo, el acercamiento basado en datos consiste en aprovechar, con el uso de técnicas estadísticas o las de aprendizaje de máquina, los datos históricos adquiridos de los sistemas bajo estudio. Estos algoritmos extraen de las muestras, la información redundante que se encuentran en las variables, lo que les ha permitido convertirse en la alternativa más adecuada para enfrentarse a procesos de alta complejidad [10].

Los métodos basados en datos permiten el desarrollo de tareas como la clasificación y la regresión [11]. Varias técnicas se han desarrollado para hacerle frente a las tareas de clasificación usando el enfoque de minería de datos: redes neuronales, árboles de decisión, máquinas de soporte de vectores, entre otras. El empleo de árboles de decisión suele ser recomendado para cuando se desea lograr un buen entendimiento del proceso de clasificación [12].

Los árboles de decisión empleados en el diagnóstico de fallo no son lo mismo que los árboles de fallo. El análisis de árboles de fallo es una técnica probabilística en la cual se usan puertas lógicas booleanas para estimar el riesgo de un “evento superior” en el sistema basado en la probabilidad de sus “causas raíces” [13]. Algunos trabajos que demuestran la aplicación de esta técnica son [14-17]. En este trabajo nos enfocaremos específicamente en los árboles de decisión y el empleo de la lógica difusa aplicados al diagnóstico de fallos.

El análisis a partir de clasificadores basados en árboles de decisión tiene un amplio uso en el diagnóstico de fallos [18]. Entre las principales ventajas de esta técnica destacan la facilidad para su comprensión; velocidades de aprendizaje y clasificación satisfactorias; robustez ante datos discretos, binarios y continuos y robustez en la interpretación de valores faltantes [19]. En [20] se presenta un resumen de los avances obtenidos relacionados a esta técnica, incluyendo desde los algoritmos de inducción y los criterios de división hasta sus combinaciones con otras técnicas de clasificación.

Aunque los árboles de decisión tienen su origen en el siglo pasado todavía siguen demostrando su aplicabilidad, tal y como se demuestra en [21-25]. Los mismos han demostrado su eficacia en el diagnóstico de fallos de engranajes en sistemas rotatorios a partir del análisis de características de los datos como la desviación estándar, máximo, mínimo y varianza de una señal vibratoria medida con un acelerómetro [26]. Otras investigaciones también han probado la efectividad de la incorporación de la lógica difusa [27-29]. En [8] se añade a una técnica de diagnóstico no destructivo para las aspas de motores en aeronaves. Aquí, el análisis de componentes principales para la reducción de la dimensionalidad de los datos derivaba en pérdida de información. Pero este problema logró solucionarse con la representación difusa de los datos, lo que demostró un aumento en la precisión de la clasificación del sistema de diagnóstico. En [30] demuestran la efectividad de dos mejoras del algoritmo ID3 en la inducción de árboles de decisión para su aplicación en el diagnóstico de fallos de circuitos analógicos. La primera consiste en la introducción de lo que denominan “índice de validez de grupo en la ganancia de información” como criterio de división. La segunda mejora trata de considerar las tasas de fallas en la construcción del árbol, lo que permite tener en cuenta tanto la capacidad de partición (o división) de cada atributo como la prioridad de aislamiento de las fallas con mayores tasas. Para el diagnóstico de fallos de circuitos analógicos, en [11] presentan los denominados árboles de decisión difusos de Fisher de múltiples valores. En este caso, la teoría difusa se aplica durante la etapa de discretización de las variables monitoreadas. Lo anterior permite resolver el problema de no contar con límites definidos para la discretización, así como el de la desviación que pudiese ocurrir en las variables. En [31] se propone un método para el diagnóstico de fallos en termoeléctricas basado en árboles de decisión por ser la que mejores resultados aportó en cuanto a precisión al ser comparada con otras técnicas como red neuronal, algoritmo bayesiano y máquina de soporte de vectores.

Debido a que los árboles de decisión se ubican dentro de los denominados métodos supervisados estos necesitan el empleo de un conjunto de datos de entrenamiento. Investigaciones como la de [32] van dedicadas a cómo mejorar el conjunto de datos de entrenamiento para obtener un árbol más preciso. Por otra parte, trabajos como el de [33], proponen algoritmos para perfeccionar la etapa de discretización difusa. Sin embargo, todo ello implica un requisito indispensable: contar con una base de datos de entrenamiento íntegra, que además del funcionamiento normal, contenga conjuntos de datos representativos de los fallos que se pretenden diagnosticar. Aunque es cierto que el “Internet de las Cosas” (IoT por sus siglas en inglés) ha potenciado la cuarta revolución industrial como una fuente de datos sin precedentes para su posterior análisis [34], no siempre se puede garantizar la confiabilidad de todo ese volumen de datos. Luego, contar con una base de datos de entrenamiento íntegra, que además del funcionamiento normal, contenga conjuntos de datos representativos de los fallos que se pretenden diagnosticar, no constituye un requisito fácil de cumplir. Además, también debe tenerse la clasificación de cada observación en el estado de operación del proceso que le corresponda, pues esta información es la que el algoritmo procesa para llevar a cabo la tarea de inducción del árbol de decisión. Lo anterior constituye una seria desventaja por las dificultades en obtener una base de datos de entrenamiento adecuada. En la literatura científica se presentan algunos métodos no supervisados, como el de [35], que eliminan la etapa de clasificar. Sin embargo, persiste la dificultad de que, si se desea incorporar un nuevo fallo

en el diagnóstico, habrá que esperar a obtener una cantidad suficiente de datos del mismo que permita volver a realizar el procedimiento.

El “Lenguaje de marcado de modelo predictivo” (PMML por sus siglas en inglés) es un lenguaje basado en “Lenguaje de Marcas Extensibles” (XML por sus siglas en inglés) que se ha convertido en el estándar de facto para representar no solo modelos predictivos y descriptivos, sino que también preprocesamiento y post-procesamiento de datos. Además, permite el intercambio de modelos entre diferentes herramientas y ambientes de desarrollo, evadiendo incompatibilidades y problemas de software propietarios. El estándar PMML ha sido utilizado para intercambiar modelos en sistemas ciber-físicos los cuales son una de las tecnologías primarias en la industria 4.0 [36,37].

El objetivo y principal contribución de este trabajo es presentar una metodología que no necesita una base de datos de entrenamiento para la construcción del árbol de decisión difuso, sustituyendo la misma por una tabla de síntomas de fallos construida a partir del criterio de los expertos. Esta tabla conformará el núcleo de conocimiento con el que se llevará a cabo la inducción del árbol de fallos basado en el algoritmo ID3. De esta manera, en caso de que se desee hacer una modificación del diagnosticador, basta con realizar una nueva entrevista a los expertos para que definan las nuevas características del proceso y/o de los fallos que se diagnostican o se quieren empezar a diagnosticar. La otra contribución del trabajo lo constituye el aprovechar también el conocimiento de los expertos para llevar a cabo la etapa de discretización difusa lo que permite contemplar todo el universo de discurso de las variables que los propios expertos determinen como necesarias para el diagnóstico de los fallos. Ha de resaltarse que la incorporación de dicho conocimiento se facilitará gracias al manejo de la incertidumbre con la lógica difusa. En el trabajo se presenta también un esquema general de desarrollo de modelos de diagnóstico que utiliza el lenguaje PMML para el intercambio de modelos de inteligencia artificial. El mismo les da soporte a los árboles de decisión entre dos aplicaciones informáticas con las tareas de configuración y monitoreo en línea de procesos respectivamente.

El resto del trabajo está organizado de la siguiente manera, en la sección 2 se presentan los fundamentos teóricos que soportan la presente investigación. En la sección 3 se presenta y analiza la metodología propuesta para el diseño del sistema de diagnóstico de fallos. La aplicación de la metodología propuesta a una planta de tratamiento de agua de la industria biofarmacéutica y la discusión de los resultados obtenidos se desarrolla en la sección 4. Finalmente se presentan las conclusiones y recomendaciones para trabajos futuros.

## **2.- MATERIALES Y MÉTODOS**

### **2.1.- DISCRETIZACIÓN DIFUSA DE LAS VARIABLES**

La lógica difusa se basa en la descripción del universo de discurso de cada variable a partir de la definición de conjuntos difusos. Estos conjuntos estarán representados por una función de pertenencia  $\mu$  que puede ser de diferentes tipos: triangular, trapezoidal, gaussiana, exponencial, entre otras. Esta selección dependerá de la aplicación en la que se emplee, la percepción del concepto representado y el nivel de detalle requerido. En [14] se ofrecen un resumen con varias definiciones básicas relacionadas a este tema.

Se le denomina discretización de datos al proceso que tiene como objetivo dividir algunas características continuas en un pequeño número de intervalos, que logren mantener una adecuada coherencia de clase [38]. Este tipo de discretización, conocido como discretización clásica, tiene como característica que los diferentes intervalos que se formen serán excluyentes. Sin embargo, este rasgo introduce una dificultad en la extracción del conocimiento si los expertos entrevistados no tienen el mismo criterio en cuanto a los valores que definirán los intervalos e, incluso, un mismo experto pudiera tener dudas al determinar con precisión el valor que separará un intervalo de otro. El proceso de discretización es imprescindible realizarlo cuando se utilizan herramientas para la inducción del árbol como el algoritmo ID3 que sólo trabaja con atributos nominales o discretos.

Para resolver la desventaja mencionada anteriormente en el proceso de discretización se combina este proceso con la lógica difusa con el objetivo de emular mejor el pensamiento y conocimiento de los expertos. Los valores numéricos adquiridos del proceso serán llevados a categorías difusas. De esta forma, los límites entre categorías podrán quedar solapados, como expresión de la incertidumbre que puede manifestarse al definirlos. Además, permitirá que los expertos puedan asignarle a cada valor en el universo de discurso, un valor de pertinencia acorde a su criterio, para cada una de las categorías definidas de las variables bajo análisis.

## 2.2.- TABLA DE SÍNTOMAS DE FALLOS

Una herramienta que permite hacer uso del conocimiento de los expertos para definir la relación entre las distintas variables que caracterizan los fallos del proceso, es la denominada tabla o matriz de síntomas de fallos. Esta se basa en agrupar, en forma de tabla, todos los síntomas que deriven en cada uno de los estados de operación del proceso. Algunas características a destacar son que: la tabla no puede quedar vacía; debe recoger al menos un síntoma por variable para cada fallo que se desee diagnosticar y la unión de todos los síntomas plasma todas las categorías de todas las variables que han sido definidas. De esta manera, toda la información que se necesita para el diagnóstico será tomada del conocimiento de los expertos y no será necesario contar con una base de datos de entrenamiento.

## 2.3.- ÁRBOLES DE DECISIÓN

Un árbol de decisión es un clasificador expresado como una partición recursiva del espacio muestral. El árbol está compuesto por nodos: un nodo raíz, que es al cual no le llega ninguna arista o rama, mientras que al resto de estos les llega una sola rama entrante. Los nodos que tienen varias ramas de salida se llaman nodos internos o de encuesta y los nodos que no tienen ramas de salida se llaman hojas o nodos de decisión [12]. A la unión entre un nodo y el siguiente se le denomina rama y a cada conjunto de nodos y ramas desde el nodo raíz hasta cada una de las hojas se le llama camino. La Fig. 1 presenta un ejemplo de un árbol de decisión para ilustrar su estructura:

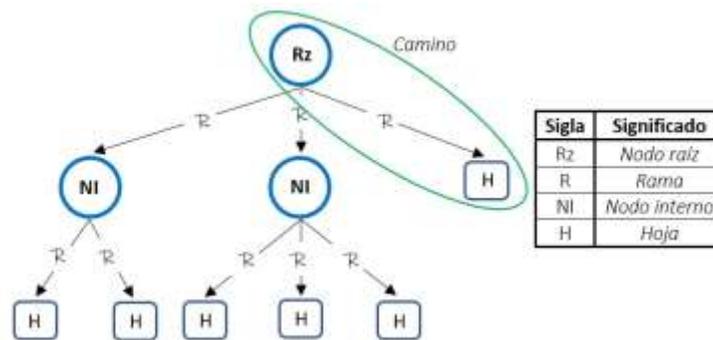


Figura 1  
 Partes de un árbol de decisión.

### 2.3.1.- INDUCCIÓN DEL ÁRBOL DE DECISIÓN

La inducción del árbol de decisión consiste en el desarrollo de un conjunto de reglas de clasificación que permita determinar la clase (estado de operación de proceso) a la que pertenecen una serie de objetos (variables) dados sus atributos (valores sensados de las variables u observaciones) [39].

Existen diferentes algoritmos para llevar a cabo esta tarea, algunos de ellos son ID3, C4.5 y CART [12]. En nuestro caso se seleccionó al ID3 por simplicidad y lo bien que representa la relación real que existe entre las variables y los estados de operación del proceso (clases), lo que facilita la comprensión del árbol de decisión resultante [39]. Este algoritmo usa como criterio de división del conjunto de entrenamiento (tabla de síntomas de fallo) a la ganancia de información y como criterio de parada que todos los síntomas pertenezcan al mismo estado de operación o que la máxima ganancia de información no sea mayor que 0.

El procedimiento de inducción del árbol de decisión con el algoritmo ID3 según [39] es el siguiente: sea  $A = \{A_1, A_2, \dots, A_n\}$  un conjunto de  $n$  variables donde a cada variable  $A_i$  le corresponde un conjunto  $\{v_1, v_2, \dots, v_p\}$  de categorías. Sea  $C = \{C_1, C_2, \dots, C_{r+1}\}$  un conjunto de  $r + 1$  estados de operación de los cuales  $r$  estados serán fallos y el restante es la operación normal del sistema. Cada estado de operación estará caracterizado por uno o varios síntomas y cada síntoma del conjunto de entrenamiento  $S$  estará definido por una combinación única de categorías por cada una de las  $n$  variables del sistema. Sea  $T$  una prueba o evaluación realizada sobre el conjunto de los síntomas  $S$  con posibles salidas  $\{G_1, G_2, \dots, G_n\}$ , una salida por

cada variable del sistema. Si se define  $T$  como el criterio de división del conjunto de entrenamiento, en este caso la ganancia de información, entonces se obtendrá como resultado la variable  $A_i$  que maximiza dicha ganancia de información. Luego,  $A_i$  se tomará como el nodo raíz del árbol actual y el conjunto  $S$  se dividirá en  $p$  subconjuntos, uno por cada categoría asociada a  $A_i$ . Cabe puntualizarse que una vez una variable sea seleccionada como raíz de un árbol, ya no se podrá volver a tener en cuenta en las siguientes pruebas que se realicen. Entonces, si cada uno de los  $p$  subconjuntos de  $S$  puede ser sustituido por un árbol de decisión, se logrará un árbol de decisión para todo  $S$ . O sea, por cada  $S_j$  se generará un sub-árbol de decisión y la unión de todos estos sub-árboles darán lugar al árbol de decisión final.  $T$  se continuará realizando recursivamente hasta que se cumplan cualquiera de las condiciones de parada del algoritmo ID3 mencionadas anteriormente.

El criterio de división a emplear en la inducción del árbol de decisión es el de la ganancia de información [12]:

$$G(A_j, S) = I(S) - E(A_j, S) \quad (1)$$

Donde:  $I(S)$  es la información necesaria para que el árbol pueda clasificar una clase cualquiera en  $S$ :

$$I(S) = \sum_{k \in \text{dom}(E)} -\frac{|C_k|}{|S|} \cdot \log_2 \frac{|C_k|}{|S|} \quad (2)$$

- $|S|$ : Cantidad de síntomas de fallo en  $S$ .
- $|C_k|$ : Cantidad de síntomas de fallo en  $S$  que pertenecen al  $k$ -ésimo estado de operación.

y  $E(A_j, S)$  es la información esperada que se necesita para que  $A_j$  sea la raíz del árbol:

$$E(A_j; S) = \sum_{m=1}^p -\frac{|S_{v_m}|}{|S|} \cdot I(S) \quad (3)$$

- $|S_{v_m}|$ : Cantidad de síntomas de fallo en  $S$  donde  $A_j$  toma el valor  $v_m$ .

Los pasos antes descritos se resumen en Algoritmo 1.

**Algoritmo 1.** Pseudocódigo para la inducción del árbol de decisión propuesta [*NDT\_Induction* ( $S$ )]  
*NDT\_Induction* ( $S$ );

**Input:**  $S$  (Tabla de síntomas de fallos);

**Output:**  $N$  (Nodo del árbol de decisión);

1. **if**  $S \in C_i$  **then**
2.     **return**  $N = C_i$ ;
3. **else**
4.     **for each**  $A_j$  **in**  $S$  **do**
5.          $G(A_j, S) = I(S) - E(A_j, S)$ ;
6.     **end for**
7.      $\text{max}G = \text{MAX}\{G(A_j, S)\}$ ;
8.     **if**  $\text{max}G \leq 0$  **then**
9.          $\text{max}C = \text{MAX}\{|C_k|\}$ ;
10.        **return**  $N = \{C_k / \text{max}C = |C_k|\}$ ;
11.     **else**
12.         $N = \{A_j / \text{max}G = G(A_j, S)\}$ ;
13.        **for each**  $v_p$  **in**  $N$  **do**
14.            **for each**  $s_m$  **in**  $S$  **do**
15.                 $S_v = \{s_m / v_p \in s_m\}$ ;
16.            **end for**
17.        **end for**

```

18.   for each  $S_v$  do
19.       delete  $N$ ;
20.   end for
21.   for each  $S_v$  do
22.       call: DT_Induction ( $S_v$ );
23.   end for
24. end if
25. end if

```

## 2.3.2.- INFERENCIA DEL ÁRBOL DE DECISIÓN

La inferencia en un árbol de decisión clásico define el proceso con el cual, dado una observación de los atributos del sistema, se llega a una clasificación a partir del árbol de decisión ya inducido. Este procedimiento comienza evaluando la variable ubicada en el nodo raíz, la rama que resulte de esta operación dará lugar al nuevo nodo a evaluar y así sucesivamente hasta llegar a un nodo hoja, la clase correspondiente a este último nodo será la clasificación resultante de este procedimiento.

De lo anterior resalta que la forma clásica de inferencia realiza las pruebas a los nodos para un solo camino del árbol; sin embargo, al hacer uso de la lógica difusa, se le deben realizar algunas modificaciones a ese método. De acuerdo al algoritmo planteado en [40] el valor de cada atributo será evaluado en el nodo correspondiente para determinar el valor de pertenencia a cada una de las categorías, de forma tal que todos los caminos del árbol de decisión intervengan en la clasificación; correspondiendo el resultado final a la clase que posea la hoja con mayor valor de pertenencia.

Para todo árbol de decisión inducido se pueden definir:

- $\{F_1, \dots, F_n\}$ : conjunto de los  $n$  caminos.
- $\{L_{p1}, \dots, L_{pm}\}$ : conjunto de los  $m$  niveles del  $p$ -ésimo camino.
- $A_{(ij)(pk)}$ :  $j$ -ésima categoría difusa perteneciente a la  $i$ -ésima variable ubicada en el  $k$ -ésimo nivel del  $p$ -ésimo camino.
- $\mu_{(ij)(pk)}$ : valor de pertenencia correspondiente a la categoría difusa  $A_{(ij)(pk)}$ .
- $\{H_1, \dots, H_n\}$ : conjunto de las  $n$  hojas (habrá tantas hojas como caminos tenga el árbol).
- $\{G_1, \dots, G_z\}$ : conjunto de los  $z$  estados de operación.
- $\{x_1, \dots, x_v\}$ : muestra actual de las variables medidas (observación).

Teniendo en cuenta las definiciones anteriores y siguiendo lo planteado en [39], la inferencia difusa de un árbol de decisión puede llevarse a cabo en cuatro pasos. Primero, se evalúa la muestra correspondiente  $x_i$  de la observación en cada categoría difusa  $A_{ij}$  para obtener el valor de pertenencia  $\mu_{ij}$ . Luego, por cada uno de los  $n$  caminos del árbol, se multiplican los valores de pertenencia y se le asigna el resultado a la hoja  $H_p$  de dicho camino. Posteriormente se suman los valores de las hojas correspondientes al estado de operación  $O_q$ . Por último, se toma como clasificación final al estado de operación que mayor sumatoria obtuvo. Este procedimiento se puede resumir tal y como se muestra en el Algoritmo 2.

**Algoritmo 2.** Pseudocódigo para la inducción del árbol de decisión propuesta [*OE DT\_Inference (Obs)*]

*OE DT\_Inference (Obs)*;

**Input:** Obs: observación;

**Output:** OE: estado de operación resultante;

```

1. for each  $x_i$  do
2.     for each  $A_{ij}$  do
3.          $\mu_{ij} = A_{ij}(x_i)$ ;
4.     end for
5. end for
6. for each  $F_p$  do
7.      $prod_{F_p} = \prod_{k=1}^m \mu_{pk}$ ;
8. end for

```

9. **for each**  $G_q$  **do**
10.  $sum_{G_q} = \sum (prod_{F_p}/H_p \in G_q)$ ;
11. **end for**
12. **return**  $OE = \{G_v/sum_{G_v} = Max \{sum_{G_q}\}$ ;
13. **end if**
14. **end if**

Suponiendo que se tiene un proceso en el que se miden tres variables ( $V1$ ;  $V2$ ;  $V3$ ) y se definen tres estados de operación: el estado normal ( $N$ ) y dos fallos ( $F1$ ;  $F2$ ); al llevar a cabo el método anterior se obtiene lo presentado en la Fig. 2, donde la clasificación final resultó ser el fallo  $F1$  con un 66.25 % de certeza.

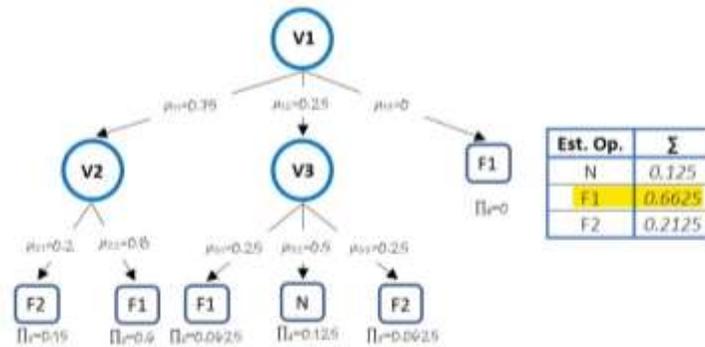


Figura 2

Ejemplo de aplicación del método de inferencia modificado.

Este procedimiento, a diferencia del de inducción del árbol de decisión que solo se aplica una vez se realiza repetidamente cada una determinada cantidad de muestras de las variables medidas.

## 2.4.- ESTRUCTURA DEL ESQUEMA DE MONITOREO

En este trabajo se propone un esquema de desarrollo en el que el árbol de decisión (técnica de monitoreo) es configurado en una aplicación fuera de línea. Luego, la configuración resultante es exportada en un fichero con formato PMML el cual será utilizado por una segunda aplicación. Dicha aplicación (aplicación en línea) es la que realiza el monitoreo en línea del proceso, a partir de la obtención de los valores de las variables directamente del sistema de supervisión, control y adquisición de datos (SCADA por sus siglas en inglés) mediante el protocolo OPC (Open Platform Communication). Una representación del esquema empleado se presenta en la Fig. 3.



Figura 3

Flujo del esquema propuesto.

La información generada por la aplicación que realiza la configuración del proceso de monitoreo consiste en: la identificación de las variables del SCADA que serán utilizadas para el monitoreo y los fallos que se desean diagnosticar, las transformaciones

realizadas a las variables seleccionadas y la definición y diseño de los modelos utilizados para realizar la tarea de diagnóstico. El formato PMML permite darles persistencia a los procedimientos anteriores, por lo que se utilizará como soporte para transmitir la información del modelo configurado a la aplicación que realiza las tareas de monitoreo en línea.

### 3.- METODOLOGÍA PROPUESTA

A continuación, se presentarán la metodología propuesta en este trabajo:

- 1) **Entrevista a los expertos:** Como hemos estado mencionando desde el inicio, una etapa fundamental para el diseño del diagnosticador consiste en la entrevista a los expertos. El objetivo fundamental de este proceso consiste en obtener la base de conocimiento con la cual se construirá el árbol de decisión.
  - a) **Definir los fallos que se desean diagnosticar:** Además de conocer los fallos a diagnosticar es recomendable aprovechar el marco de la reunión para recoger, por cada fallo, las medidas que los expertos consideren que deban ser adoptadas a fin de recuperar el sistema.
  - b) **Definir las variables bajo estudio:** De acuerdo a los fallos anotados en el paso anterior deberá realizarse un profundo análisis que permita determinar todas las variables que, por lo menos a priori, los expertos crean que son imprescindibles para la detección de los primeros. Las variables aquí seleccionadas no se emplearán necesariamente en su totalidad. Lo anterior se debe a que es posible que el algoritmo de inducción sea capaz de construir el árbol prescindiendo de alguna(s) de ellas.
  - c) **Aplicar la discretización difusa al universo de discurso de cada variable:** Cada variable deberá de ser discretizada en categorías difusas. La cantidad de categorías por variable estará en correspondencia con las necesidades que existan para definir los síntomas de los fallos en el próximo paso.
  - d) **Construir la tabla de síntomas de fallos.** Para cada estado de operación del proceso, tanto fallos como el estado normal, se definen todos los síntomas que se consideren necesarios para conocer el estado del sistema. Esta es la fase en la que, gracias a la experiencia de los expertos, podremos prescindir del uso de una base de datos de entrenamiento, evitando con ello todos los inconvenientes que hemos ido mencionando.
- 2) **Configuración del esquema de diagnóstico:** En esta etapa se lleva a cabo la tarea de configuración fuera de línea del algoritmo seleccionado para el diagnóstico. Esta va a permitir realizar ajustes del esquema desarrollado antes de su puesta en funcionamiento en línea.
  - a) **Introducir el conocimiento experto:** El conocimiento extraído de los expertos es incorporado a la aplicación.
  - b) **Realizar la inducción del árbol de decisión:** Se construye al árbol de decisión a partir de la aplicación del algoritmo ID3 a la tabla de síntomas de fallos antes declarada. Esta estructura, gracias a la forma tan simple con la que representa el conocimiento, podrá ser fácilmente interpretada por aquellos trabajadores de la planta que lo necesiten. Eso evitará los tiempos de demora en la toma de decisiones correctivas que a veces implica el no contar con una persona de experiencia justo en el lugar y momento en el que se ha detectado un fallo.
  - c) **Exportar el fichero PMML:** Se exporta el fichero en formato PMML que contiene la configuración resultante para realizar el intercambio de información con la aplicación de monitoreo en línea.
- 3) **Uso en línea del esquema de diagnóstico:**
  - a) **Importar del fichero PMML:** Para usar el modelo seleccionado se carga en la aplicación en línea el fichero en formato PMML con la información del árbol de decisión configurado fuera de línea para su uso en el sistema.
  - b) **Comunicar la aplicación con el sistema SCADA:** Haciendo uso del protocolo OPC, se realiza la conexión de la aplicación con el sistema SCADA. El objetivo de este paso es enlazar las variables que se emplearán en el diagnóstico con sus muestras correspondientes en el sistema de monitoreo.
  - c) **Realizar la inferencia del árbol de decisión:** Finalmente, solo queda hacer uso del árbol de decisión para clasificar el conjunto de muestras que se va obteniendo de las variables del sistema. Para cada uno de esos conjuntos (u observaciones) se tendrá el estado de operación del sistema en ese momento, ya sea estado normal o un determinado fallo.

### 4.- RESULTADOS

La metodología propuesta fue aplicada a un proceso real, específicamente en la planta de tratamiento de agua para inyección de la industria farmacéutica Citostáticos perteneciente a la empresa AICA, ubicada en la provincia La Habana, Cuba.

El proceso consiste en un sistema para guardar y distribuir agua purificada para inyección; garantizando un suministro constante y fiable dentro del sistema de recirculación en sí, así como en todos los puntos de uso. El mismo posee varios modos de operación, pero para la tarea de diagnóstico se estará trabajando específicamente en el modo de producción principal. El sistema cuenta con una bomba de carga de presión que alimenta el circuito de recirculación con el agua del depósito para que

pueda tomarse de los puntos de uso. En la línea de retorno del circuito de recirculación se monitorizan el flujo, la presión, la conductividad, TOC y la temperatura (variables a utilizar en el diagnóstico de fallos). Los datos operativos del proceso se recogen en la Tabla 1.

**Tabla 1**  
**Datos operativos del proceso bajo estudio**

Variables	Unidad de Med.	Valor
<i>Flujo en la línea de retorno</i>	$m^3/s$	4
<i>Presión en la línea de retorno</i>	<i>bar</i>	aprox. 1.4
<i>Temperatura de trabajo</i>	$^{\circ}C$	18
<i>TOC</i>	<i>ppb</i>	< 80
<i>Conductividad a 25°C</i>	$\mu S/cm$	< 1.3

## 4.2.- PRUEBA DE DIAGNÓSTICO REAL

### 1) Entrevista con los expertos.

Según los expertos del sistema los fallos que más suelen afectar el proceso son: Fallo instrumento, Contaminación del lazo, Falta de vapor de entrada y Fallo de la bomba. En la Tabla 2 se recogen algunas de sus características.

**Tabla 2**  
**Características de los fallos a diagnosticar**

Fallos a diagnosticar	Características
<i>Fallo instrumento</i>	Indica que los instrumentos de medición empleados están arrojando valores fuera del rango lógico del canal analógico.
<i>Contaminación del lazo</i>	El agua purificada tiene que cumplir con ciertas normas de calidad, una de ellas es que el valor de TOC no puede sobrepasar cierto límite, pues cuando ello ocurre significa que el agua está contaminada y, por tanto, ya no es adecuada para su uso. Similarmente ocurre con la variable conductividad.
<i>Falta de vapor de entrada</i>	Significa que no está llegando al sistema el vapor industrial que viene de la línea de la caldera y que se requiere en el intercambiador de calor (calentador).
<i>Fallo de la bomba</i>	Ocurre cuando la bomba encargada de mantener la recirculación del agua en el lazo falla y no mantiene el flujo (ni la presión) establecido.

Por otro lado, las variables asociadas a estos fallos, como se mencionó anteriormente, son las que se encuentran en la línea de retorno del circuito de recirculación: flujo, presión, conductividad, TOC y temperatura. La discretización difusa de las mismas en funciones de pertenencia de tipo trapezoidal aparece en la Tabla 3.

**Tabla 3**  
**Discretización difusa de las variables para el diagnóstico**

Variables	Categorías difusas	Parámetros (a; b; c; d)
<i>Flujo en la línea de retorno (m<sup>3</sup>/s)</i>	Bajo	(-∞; -∞; 1; 1.2)
	Normal	(1; 1.2; 4; 5)
	Alto	(4; 5; ∞; ∞)
<i>Presión en la línea de retorno (bar)</i>	Baja	(-∞; -∞; 1; 1.1)
	Normal	(1; 1.1; 2.3; 2.5)
	Alta	(2.3; 2.5; ∞; ∞)
<i>Temperatura de trabajo (°C)</i>	Baja	(-∞; -∞; 10; 15)
	Normal	(10; 15; 20; 25)
	Alta	(20; 25; ∞; ∞)
<i>TOC (ppb)</i>	Normal	(-∞; -∞; 50; 80)
	Alto	(50; 80; ∞; ∞)
<i>Conductividad a 25°C (μS/cm)</i>	Baja	(-∞; -∞; 0.3; 0.4)
	Normal	(0.3; 0.4; 0.97; 1.3)
	Alta	(0.97; 1.3; ∞; ∞)

El último paso de la entrevista fue la conformación de la base de conocimiento (tabla de síntomas de fallos) que se muestra en la Fig. 4.

Estados de Operación	Conductividad	Temperatura	Presión	TOC	Flujo
<b>Normal</b>	Normal	Normal	Normal	Normal	Normal
<b>Falta de vapor entrada</b>	Normal	Baja	Normal	Normal	Normal
<b>Contaminación del lazo</b>	Alta	Normal	Normal	Alto	Normal
	Alta	Normal	Normal	Alto	Bajo
	Alta	Normal	Normal	Normal	Normal
	Normal	Normal	Normal	Alto	Normal
<b>Fallo Bomba</b>	Normal	Normal	Bajo	Normal	Bajo
	Normal	Normal	Bajo	Normal	Normal
	Normal	Normal	Normal	Normal	Bajo
<b>Fallo Instrumento</b>	Baja	Normal	Normal	Normal	Normal
	Normal	Alta	Normal	Normal	Normal
	Normal	Normal	Alto	Normal	Normal
	Normal	Normal	Normal	Normal	Alto

**Figura 4**

**Tabla de síntomas de fallos para del proceso bajo estudio.**

## 2) Inducción del árbol de decisión.

El procedimiento de inducción del árbol de decisión arrojó el resultado mostrado en la Fig. 5, de donde se aprecia que para este proceso todas las variables definidas por los expertos fueron necesarias en el diagnóstico. Resulta interesante hacer un paréntesis en este caso y mencionar que, si bien la tabla de síntomas de fallos constituye una base de reglas con la que se pudiera realizar el diagnóstico directamente, o sea, sin la necesidad de pasar por el trabajo de confeccionar el árbol de fallos;

es fácil darse cuenta de la gran ventaja que constituye la representación gráfica, pues como se observa en la Fig. 5, cualquier trabajador de la planta de tratamiento puede comprender de forma rápida y sencilla los distintos comportamientos de las variables que llevan a los fallos definidos, sin la necesidad de tener contacto con expertos cada vez que se requiera de esta información.

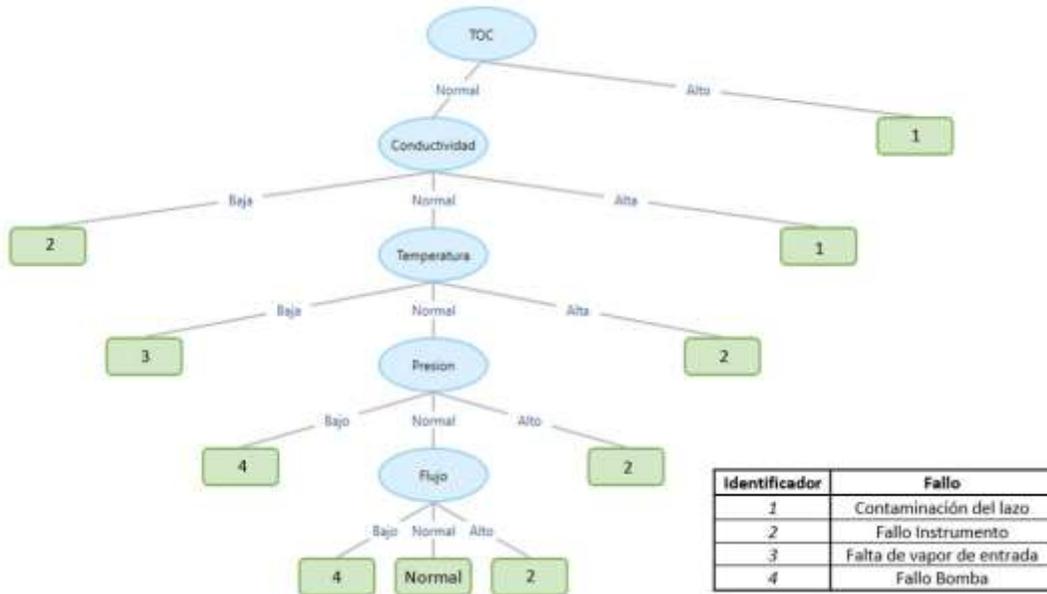


Figura 5

Árbol de decisión del proceso bajo estudio.

### 3) Inferencia del árbol de fallos.

Algunos de los resultados del diagnóstico de los fallos en la planta se observan en las Fig. 6-8, a la izquierda se encuentra una gráfica con una muestra de los datos de las variables que llevaron al fallo diagnosticado y a la derecha una parte de la vista de la aplicación empleada para esta tarea. La Tabla 4 resume una descripción de lo que ocurre en estas figuras.

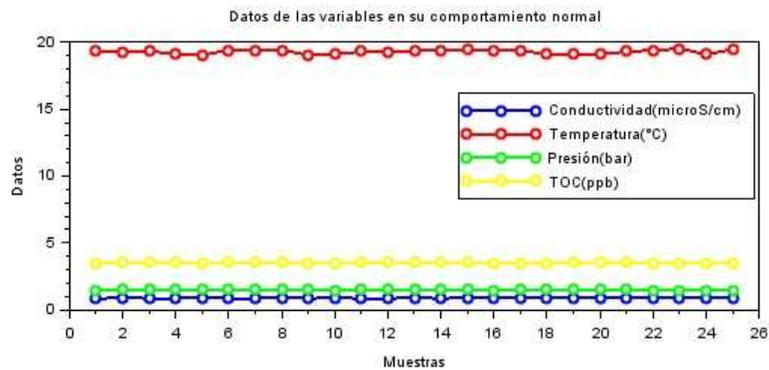


Figura 6

Operación Normal del proceso.

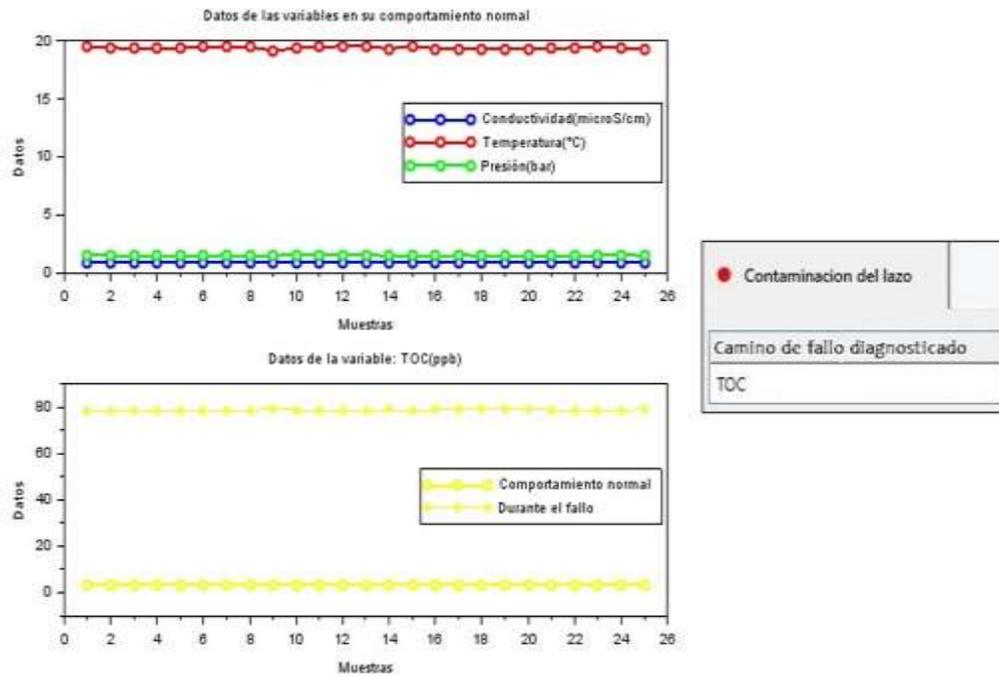


Figura 7  
Contaminación del lazo.

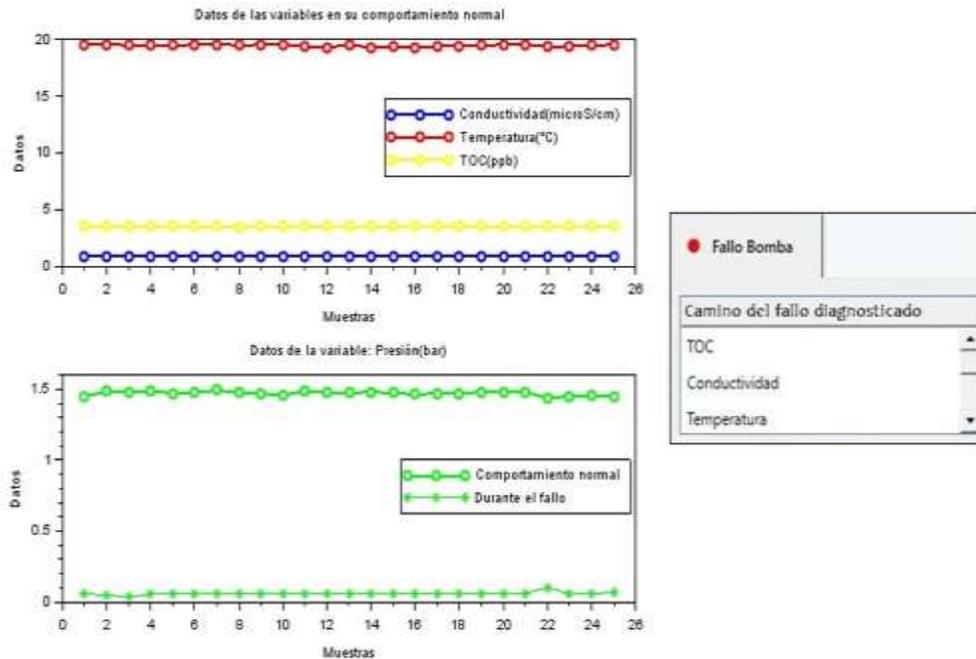


Figura 8  
Fallo de la bomba.

**Tabla 4**  
**Resumen de los aspectos observados en las figuras 6-8.**

Estado de Operación	Notas
<i>Normal</i>	Los datos se encuentran dentro de la categoría Normal para cada una de las variables.
<i>Contaminación del lazo</i>	Los datos del TOC están muy por encima del valor esperado, por lo que corresponden a la categoría Alto. (Ver cuarto síntoma del fallo en la Fig. 4).
<i>Fallo de la bomba</i>	Si bien los datos de TOC, Conductividad y Temperatura entran en la categoría Normal; los de la Presión son menores a 1.1 bar, lo que hace que pertenezcan a la categoría Baja. (Ver segundo síntoma del fallo en la Fig. 4).

## 5.- CONCLUSIONES

Con la presente investigación se obtuvo una nueva metodología para el diagnóstico de fallos basado en árboles de decisión y lógica difusa. Dicha metodología demuestra:

- las facilidades de la lógica difusa aplicada a la etapa de discretización de las variables; por cuanto es capaz de simular mejor el pensamiento de los expertos y, permitir con ello, hacer frente a los problemas relacionados a la incertidumbre que surge cuando se definen rangos de las variables.
- las capacidades que brinda el uso de tablas de síntomas de fallos. Debido a que los expertos cuentan con grandes conocimientos relacionados a la tarea de diagnóstico, pero muchas veces no saben representarlos adecuadamente. Unido al hecho de que da la posibilidad de no necesitar una gran base de datos que sea cien por ciento accesible e íntegra para el entrenamiento del árbol de decisión.
- las potencialidades del estándar PMML. El cual permite el intercambio de modelos de inteligencia artificial que, a su vez, posibilita el desarrollo de un esquema de monitoreo de dos etapas: configuración y monitoreo en línea.
- la sencillez de comprensión de los árboles de decisión para tareas de diagnóstico. En vez de expresar el conocimiento de los expertos mediante una base de reglas, se opta por hacer una representación gráfica de un árbol de decisión, a fin de aprovechar las ventajas de esta técnica en cuanto a la facilidad de comprensión que les ofrece a los trabajadores que tengan que utilizar la herramienta. Además, con el ejemplo de simulación se demostró que a veces esta técnica contribuye a disminuir los costos del diagnóstico, al no incluir aquellas variables que realmente no sean necesarias para la realización de esta tarea.

## AGRADECIMIENTOS

Los autores agradecen a Empresa Laboratorios AICA por permitir el acceso a sus instalaciones para la validación de la propuesta de metodología propuesta en este trabajo; así como a los integrantes de EMSI FARMA S.R.L y del Grupo de Investigación de Diagnóstico de Fallos y Estimación de Parámetros perteneciente a la Universidad Tecnológica de La Habana “José Antonio Echeverría” que con sus ideas contribuyeron al desarrollo de la misma.

## REFERENCIAS

- [1] Abid A., Khan M.T., Iqbal J. A review on fault detection and diagnosis techniques: basics and beyond. *Artificial Intelligence Review*. 2021;54:1-26.
- [2] Zuo L., Zhang L., Zhang Z.-H., Luo X.-L., Liu Y. A spiking neural network-based approach to bearing fault diagnosis. *Journal of Manufacturing Systems*. 2020;61:714-724. doi:10.1016/j.jmsy.2020.07.003.
- [3] Kherif O., Benmahamed Y., Teguar M., Boubakeur A., Ghoneim S.S.M. Accuracy Improvement of Power Transformer Faults Diagnostic Using KNN Classifier With Decision Tree Principle. *IEEE Access*. 2021;9:81693-81701. doi:10.1109/ACCESS.2021.3086135.
- [4] Balaraman S., Madavan R., Vedhanayaki S., Saroja S., Srinivasan M., Stoniert A.A. Fault Diagnosis and Asset Management of Power Transformer Using Adaptive Boost Machine Learning Algorithm. *IOP Conference Series: Materials Science and Engineering*. 2021;1055(1):012133. doi:10.1088/1757-899x/1055/1/012133.
- [5] Furlas G.K., Karras G.C. A Survey on Fault Diagnosis and Fault-Tolerant Control Methods for Unmanned Aerial Vehicles. *Machines*. 2021;9(9):197. doi:10.3390/machines9090197.

- [6] Zhu H., He Z., Wei J., Wang J., Zhou H. Bearing Fault Feature Extraction and Fault Diagnosis Method Based on Feature Fusion. *Sensors*. 2021;21(7):2524. doi:10.3390/s21072524.
- [7] Rodríguez Ramos A., Bernal de Lázaro J.M., Prieto-Moreno A., Silva Neto A.J., Llanes Santiago O. An approach to robust fault diagnosis in mechanical systems using computational intelligence. *Journal of Intelligent Manufacturing*. 2019 30: 1601-1615. doi: 10.1007/s10845-017-1343-1
- [8] Camps Echevarría L., Llanes-Santiago O., Silva Neto A.J. An approach for fault diagnosis based on bio-inspired strategies. *IEEE Congress on Evolutionary Computation*. 2010. pp. 1-7 Barcelona, España. doi:10.1109/CEC.2010.5586357.
- [9] Abid A., Khan M. T., Haq I. U., Anwar S., Iqbal, J. An Improved Negative Selection Algorithm-Based Fault Detection Method. *IETE Journal of Research*. 2020;:1–12. doi:10.1080/03772063.2020.1768158.
- [10] Gonzalez-Jimenez D., del-Olmo J., Poza J., Garramiola F., Madina P. Data-Driven Fault Diagnosis for Electric Drives: A Review. *Sensors* 2021;21:4024. doi:10.3390/s21124024.
- [11] Cui Y., Shi J., Wang Z. Analog circuits fault diagnosis using multi-valued Fisher's fuzzy decision tree (MFFDT). *International Journal of Circuit Theory and Applications*. 2015;44(1):240–260. doi:10.1002/cta.2075
- [12] Rokach L., Maimon O. *Data mining with decision trees: theory and applications*. 2nd ed. Bunke H., Wang P.S.P., editors. World Scientific Publishing Co. Pte. Ltd.; 2014.
- [13] Kerelous W., Mohamed-Salah O. Interpretable logic tree analysis: A data-driven fault tree methodology for causality analysis. *Expert Systems with Applications*. 2019;136:376-391. doi:10.1016/j.eswa.2019.06.042.
- [14] de Gusmão A.P.H., Silva M.M., Poletto T., e Silva L.C., Costa A.P.C.S. Cybersecurity risk analysis model using fault tree analysis and fuzzy decision theory. *International Journal of Information Management*. 2018;43:248–260.
- [15] Cao C., Li M., Li Y., Sun Y. Intelligent fault diagnosis of hot die forging press based on binary decision diagram and fault tree analysis. *Procedia Manufacturing*. 2018;15:459–466.
- [16] Huang W., Liu Y., Zhang Y., Zhang R., Xu M., Dieu G.J.D., et al. Fault tree and fuzzy D-S evidential reasoning combined approach: an application in railway dangerous goods transportation system accident analysis. *Information Sciences*. 2020;520:117–129.
- [17] Kumar M., Kaushik M. System failure probability evaluation using fault tree analysis and expert opinions in intuitionistic fuzzy environment. *Journal of Loss Prevention in the Process Industries*. 2020;67:104236.
- [18] Castellanos M.B., Serpa A.L., Biazussi J.L., Verde W.M., do Socorro Dias Arrifano Sassim N. Fault identification using a chain of decision trees in an electrical submersible pump operating in a liquid-gas flow. *Journal of Petroleum Science and Engineering*. 2020;184:106490.
- [19] Galathiya A.S., Ganatra A.P., Bhensdadia C.K. Improved decision tree induction algorithm with feature selection, cross validation, model complexity and reduced error pruning. *International Journal of Computer Science and Information Technologies*. 2012;3(2):3427–3431.
- [20] Priyanka, Kumar D. Decision tree classifier: a detailed survey. *International Journal of Information and Decision Sciences*. 2020;12(3):246–269.
- [21] Gunasegaran V., Muralidharan V. Fault Diagnosis of Spur Gear System through Decision Tree Algorithm Using Vibration Signal. *Materials Today: Proceedings*. 2020;22:3232–3239. doi:10.1016/j.matpr.2020.03.283.
- [22] Wang X., Gu H., Wang T., Zhang W., Li A., Chu F. Deep convolutional tree-inspired network: a decision-tree-structured neural network for hierarchical fault diagnosis of bearings. *Frontiers of Mechanical Engineering*. 2021;16:814–828. doi.:10.1007/s11465-021-0650-6.
- [23] Balasubramaniam V. Fault Detection and Diagnosis in Air Handling Units with a Novel Integrated Decision Tree Algorithm. *Journal of trends in Computer Science and Smart technology*. 2021;3(1):49-58. doi:10.36548/jtcsst.2021.1.005.
- [24] Huang L., Zeng Q., Zhang R. Fuel Cell Engine Fault Diagnosis Expert System based on Decision Tree. 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference. 2019;:282-286, doi:10.1109/ITNEC.2019.8729556.
- [25] Li J., Jiang S., Li M., Xie J. A Fault Diagnosis Method of Mine Hoist Disc Brake System Based on Machine Learning. *Applied Sciences*. 2020;10(5):1768. doi:10.3390/app10051768.

- [26] Gunasegaran V., Muralidharan V. Fault diagnosis of spur gear system through decision tree algorithm using vibration signal. *Materials Today: Proceedings*. 2020;22(4):3232–3239.
- [27] TayebiHaghighi S., Koo I. Sensor Fault Diagnosis Using an Indirect Fuzzy Feedback Linearization and Decision Trees. 2021 International Conference on Information and Communication Technology Convergence. 2021;:1783-1788. doi: 10.1109/ICTC52510.2021.9620793.
- [28] Krishnakumari A., Elayaperumal A., Saravanan, M., Arvindan C. Fault diagnostics of spur gear using decision tree and fuzzy classifier. *International Journal of Advanced Manufacturing Technology*. 2017;89:3487–3494. doi:10.1007/s00170-016-9307-8.
- [29] Kundur Shantisagar; Jegadeeshwaran R., Sakthivel G., Alamelu Manghai T.M. Vibration Based Tool Inset Health Monitoring Using Decision Tree and Fuzzy Logic. *Structural Durability & Health Monitoring*. 2019;13(3): 303-316. doi:10.32604/sdhm.2019.00355.
- [30] Shi J., He Q., Wang Z. Gmm clustering-based decision trees considering fault rate and cluster validity for analog circuit fault diagnosis. *IEEE Access*. 2019;7:140637–140650.
- [31] Han Y., Zhao D., Hou H. Oil-immersed transformer internal thermoelectric potential fault diagnosis based on decision-tree of KNIME platform. *Procedia Computer Science*. 2016;83:1321–1326.
- [32] Sardari S., Eftekhari M., Afsari F. Hesitant fuzzy decision tree approach for highly imbalanced data classification. *Applied Soft Computing Journal*. 2017;61:727-741.
- [33] Zeinalkhani M., Eftekhari M. Fuzzy partitioning of continuous attributes through discretization methods to construct fuzzy decision tree classifiers. *Information Sciences*. 2014;278: 715-735. doi:10.1016/j.ins.2014.03.087.
- [34] González M., Sepúlveda L.M., Quiza, J., D’Amato J. Mejoramiento del algoritmo ADR en una red de Internet de las Cosas LoRaWAN usando Aprendizaje de Máquinapp. *Revista Ibérica de Sistemas e Tecnologías de Informação*. 2020;39:67-83.
- [35] Barrientos F., Sainz G. Interpretable knowledge extraction from emergency call data based on fuzzy unsupervised decision tree. *Knowledge-Based Systems*. 2012;25:77–87.
- [36] O’donovan P., Gallagher C., Bruton K., O’Sullivan D.T.J.M.L. A fog computing industrial cyber-physical system for embedded low-latency machine learning Industry 4.0 applications. *Manufacturing Letters*. 2018;15:139-42.
- [37] Karer G., Skrjanc I., Music G.J.I.P.V. Fuzzy model xml formulation for production dynamics analysis and control. *IFAC Proceedings Volumes*. 2014;47(3):4352-7.
- [38] Tsai C.F., Chen Y.C. The optimal combination of feature selection and data discretization: An empirical study. *Information Sciences*. 2019;505:282–293.
- [39] Quinlan J.R. Induction of decision trees. *Machine Learning*. 1986;1:81–106.
- [40] Umanol M., Okamoto H., Hatono I., Tamura H., Kawachi F., Umedzu S., et al. Fuzzy decision trees by fuzzy ID3 algorithm and its application to diagnosis systems. In: *Proceedings of 1994 IEEE 3rd International Fuzzy Systems Conference*. 1994;3:2113–2118.

## CONFLICTO DE INTERESES

No existe conflicto de intereses entre los autores, ni con ninguna institución a la que cada uno está afiliado, ni con ninguna otra institución. Las opiniones expresadas aquí son únicamente responsabilidad de los autores y no representan la posición de la Institución o las instituciones a las que están afiliados.

## CONTRIBUCIONES DE LOS AUTORES

**Amanda Martí Coll:** Curación de datos, Análisis formal, Investigación, Metodología, Recursos, Software, Validación - Verificación, Visualización y Redacción- borrador original.

**Adrian Milberberg Rodríguez:** Curación de datos, Investigación, Recursos, Software y Validación -Verificación.

**David Manresa Gutiérrez:** Conceptualización, Análisis formal, Metodología, Administración de proyecto, Software y Supervisión.

**Alberto Sergio Prieto Moreno:** Conceptualización, Adquisición de fondos, Administración de proyecto y Supervisión.

**Orestes Llanes Santiago:** Conceptualización, Adquisición de fondos, Administración de proyecto, Supervisión y Redacción – revisión y edición.

## AUTORES

**Amanda Martí Coll,** Ingeniera en Automática. Universidad Tecnológica de La Habana José Antonio Echeverría (CUJAE), La Habana, Cuba. Email: [abamanda.mc@automatica.cujae.edu.cu](mailto:abamanda.mc@automatica.cujae.edu.cu). No. ORCID: 0000-0002-3046-2447. Sus intereses de investigación se orientan hacia la minería de datos y el control de procesos industriales. Actualmente se enfoca en el estudio de los gemelos digitales aplicados a la industria farmacéutica.

**Adrian Milberberg Rodríguez,** Ingeniero en Automática. Universidad Tecnológica de La Habana José Antonio Echeverría, (CUJAE), La Habana, Cuba. Email: [4Milrod@gmail.com](mailto:4Milrod@gmail.com). No. ORCID 0000-0002-8515-1005. Sus intereses de investigación se orientan hacia la inteligencia artificial, el diagnóstico de fallos en proceso industriales y la implementación de sistemas de monitoreo.

**David Manresa Gutiérrez,** Ingeniero en Automática. CADMART Inc., Florida, Estados Unidos, Email: [davidmanresag@gmail.com](mailto:davidmanresag@gmail.com). No ORCID 0000-0002-8798-2654. Sus intereses de investigación se orientan hacia el diagnóstico de fallos en proceso industriales.

**Alberto Sergio Prieto Moreno,** Ingeniero en Automática. Doctor en Ciencias Técnicas. EMSI FARMA S.R.L, La Habana, Cuba. Email: [albprietom@emsifarma.cu](mailto:albprietom@emsifarma.cu), No. ORCID: 0000-0001-9907-885X. Sus intereses de investigación se orientan hacia informática industrial, análisis de datos, diagnóstico de fallos y automatización de procesos.

**Orestes Llanes-Santiago,** Ingeniero Electricista. Doctor en Ciencias Aplicadas. Profesor e Investigador Titular de la Universidad Tecnológica José Antonio Echeverría (CUJAE). Académico Titular de la Academia de Ciencias de Cuba. Email: [orestes@tesla.cujae.edu.cu](mailto:orestes@tesla.cujae.edu.cu) No. ORCID 0000-0002-6864-9629. Sus principales intereses de investigación están en el monitoreo de condición y diagnóstico de fallos en sistemas industriales, implementación del paradigma Industria 4.0, inteligencia computacional aplicada al control y el control de sistemas no lineales.



Esta revista se publica bajo una [Licencia Creative Commons Atribución-No Comercial-Sin Derivar 4.0 Internacional](https://creativecommons.org/licenses/by-nc-nd/4.0/)