



**Universidad de Valladolid**



**ESCUELA DE INGENIERÍAS  
INDUSTRIALES**

**UNIVERSIDAD DE VALLADOLID**

**ESCUELA DE INGENIERIAS INDUSTRIALES**

**Grado en Ingeniería en Tecnologías Industriales**

**DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA  
PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA  
Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL**

**Autor:**

**Sobrino Sesmero, Sara**

**Tutor:**

**De La Fuente Aparicio, María Jesús**

Departamento de Ingeniería de Sistemas y  
Automática

Valladolid, Julio 2023.



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA  
INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE  
INTELIGENCIA COMPUTACIONAL

Universidad de Valladolid



ESCUELA DE INGENIERÍAS  
INDUSTRIALES



## ÍNDICE DE CONTENIDOS:

|   |           |
|---|-----------|
| <b>RESUMEN:</b> .....   | <b>5</b>  |
| <b>ABSTRACT:</b> .....  | <b>6</b>  |
| <b>CAPITULO I: INTRODUCCIÓN</b> .....   | <b>7</b>  |
| 1.1. <b>INTRODUCCIÓN:</b> .....   | <b>9</b>  |
| 1.2. <b>OBJETIVOS:</b> .....  | <b>10</b> |
| 1.3. <b>ORGANIZACIÓN DE LA MEMORIA:</b> .....                                     | <b>11</b> |
| <b>CAPITULO II: ESTUDIO TEORICO</b> .....   | <b>13</b> |
| 2.1. <b>CONTROL DE CALIDAD:</b> .....   | <b>15</b> |
| 2.2. <b>CONTROL ESTADÍSTICO DE PROCESOS:</b> .....                                | <b>16</b> |
| 2.2.1. <b>VARIABILIDAD EN LOS PROCESOS:</b> .....                                 | <b>17</b> |
| 2.2.2. <b>GRÁFICOS DE CONTROL:</b> .....  | <b>20</b> |
| 2.3. <b>ANÁLISIS DE COMPONENTES PRINCIPALES (PCA):</b> .....                      | <b>22</b> |
| 2.3.1. <b>ESTADÍSTICOS EMPLEADOS EN LA DETECCIÓN DE FALLOS:</b> .....             | <b>25</b> |
| 2.4. <b>MÉTODO FDA (ANÁLISIS DISCRIMINANTE DE FISHER):</b> .....                  | <b>27</b> |
| 2.5. <b>MÉTODO t-SNE:</b> .....   | <b>28</b> |
| 2.6. <b>TÉCNICAS DE APRENDIZAJE AUTOMÁTICO:</b> .....                             | <b>30</b> |
| 2.6.1. <b>REDES NEURONALES ARTIFICIALES:</b> .....                                | <b>30</b> |
| 2.6.2. <b>ÁRBOLES DE DECISIÓN (RANDOM FOREST):</b> .....                          | <b>32</b> |
| 2.6.3. <b>MATRIZ DE CONFUSIÓN:</b> .....  | <b>35</b> |
| 2.7. <b>METODO VISUAL DE DIAGNOSIS DE FALLOS: FDA-t-SNE-NN:</b> .....             | <b>36</b> |
| <b>CAPITULO III: PLANTA TENNESSEE-EASTMAN</b> .....                               | <b>39</b> |
| 3.1. <b>DESCRIPCIÓN DE LA PLANTA:</b> .....                                       | <b>41</b> |
| 3.2. <b>DATOS DE LA PLANTA:</b> .....   | <b>43</b> |
| <b>CAPITULO IV: APLICACIÓN DESARROLLADA</b> .....                                 | <b>47</b> |
| 4.1. <b>INTRODUCCIÓN A LA APLICACIÓN DESARROLLADA:</b> .....                      | <b>49</b> |
| 4.2. <b>DETECCIÓN DE FALLOS APLICANDO ANÁLISIS DE COMPONENTES PRINCIPALES:</b> .. | <b>49</b> |
| 4.3. <b>DIAGNÓSTICO DE FALLOS MEDIANTE TÉCNICAS DE APRENDIZAJE AUTOMÁTICO:</b> 57 |           |
| 4.3.1. <b>FDA – TSNE – RED NEURONAL:</b> .....                                    | <b>57</b> |
| 4.3.2. <b>BOSQUE ALEATORIO (RANDOM FOREST):</b> .....                             | <b>71</b> |
| 4.4. <b>COMPARACIÓN DE RESULTADOS:</b> .....                                      | <b>80</b> |
| <b>CAPITULO V: CONCLUSIONES Y TRABAJO FUTURO</b> .....                            | <b>83</b> |
| 5.1. <b>CONCLUSIONES:</b> .....   | <b>85</b> |
| 5.2. <b>TRABAJO FUTURO:</b> .....   | <b>86</b> |
| <b>BIBLIOGRAFÍA:</b> .....  | <b>87</b> |



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS INDUSTRIALES

Universidad de Valladolid



## RESUMEN:

El control de calidad tiene una gran importancia dentro de la industria hoy en día, y la llegada de las nuevas tecnologías hacen que se puedan recoger un gran número de datos de los procesos industriales, que pueden, posteriormente, ser analizados y tratados.

En este trabajo se tratará de mejorar la calidad de un proceso industrial mediante la aplicación de diferentes técnicas que nos permiten detectar anomalías que ocurren en planta y hacer un posterior diagnóstico de las mismas. Una de estas técnicas usadas será el Análisis de Componentes Principales (PCA), que permite mediante la reducción de la dimensionalidad del proceso, detectar los fallos que ocurren en una planta industrial. Una vez detectado el fallo, para intentar diagnosticar de que fallo se trata, se estudiará un método que combina técnicas como el Análisis discriminante de Fisher (FDA), la incrustación de vecinos estocásticos distribuidos (t-SNE) y redes neuronales (ANN), así como un el método denominado '*Random Forest*' (bosque aleatorio), realizándose una comparación entre ambos para tratar de encontrar la mejor técnica para el diagnóstico de fallos de la planta.

Los datos con los que se trabaja son extraídos de la planta química Tennessee Eastman, contando con datos de la planta en condiciones normales de funcionamiento y cuando ocurren anomalías.

**Palabras Clave:** Planta Tennessee Eastman, Análisis de componentes principales (PCA), Análisis discriminante de Fisher (FDA), incrustación de vecinos estocásticos distribuidos (t-SNE), redes neuronales, bosque aleatorio (RF).



## **ABSTRACT:**

Quality control has a great importance in the industry today, and the advent of new technologies means that a large number of industrial process data can be collected, which can then be analysed and processed.

In this work we will try to improve the quality of an industrial process by applying different techniques that allow us to detect anomalies that occur in the plant and make a subsequent diagnosis of them. One of these techniques used will be the Main Component Analysis (PCA), which allows by reducing the dimensionality of the process, detect the failures that occur in an industrial plant. Once the failure has been detected, a method that combines techniques such as Fisher's Discriminant Analysis (FDA), Distributed Stochastic Neighbor Embedding (t-SNE) and Neural Networks (ANN) will be studied to try to diagnose the failure involved, as well as a method called 'Random Forest' (random forest), a comparison between both to try to find the best technique for diagnosing plant failures.

The data is extracted from the Tennessee Eastman chemical plant, with data from the plant under normal operating conditions and when anomalies occur.



# *CAPITULO I: INTRODUCCIÓN*



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS INDUSTRIALES

Universidad de Valladolid





## 1.1. INTRODUCCIÓN:

Mantener la calidad del producto o servicio final de un proceso productivo y garantizar que cumple con las especificaciones del cliente, es un aspecto fundamental en la industria de hoy en día. Es por ello, que se da tanta importancia al control de calidad, con el que se pretende hacer un seguimiento de la producción y evitar así que se produzcan fallos que traen consigo notables consecuencias, no solo en el proceso, sino también en lo que respecta a salud humana, medioambiental o pérdidas económicas.

Para un buen control de calidad es esencial implementar sistemas que permitan una rápida detección de cualquier fallo/anomalía que se pueda dar a lo largo del proceso, y determinar las causas y consecuencias de estos, para así poder actuar y devolver el proceso a su estado de normalidad.

Nos encontramos ante una época donde el avance de la tecnología está marcando un antes y un después en base a la automatización y aparición de fábricas inteligentes. Nos encontramos ante la “Cuarta revolución Industrial”, o también denominada, Industria 4.0, que incorporando nuevas tecnologías en las instalaciones de producción y en las operaciones, tal como el Internet de las cosas (IoT), análisis y computación en la nube, Inteligencia Artificial o ‘machine learning’, conduce a un mayor mantenimiento predictivo, optimización automática en la mejora del proceso y, sobre todo, a un nivel de capacidad para alcanzar con las necesidades del cliente que antes era inimaginable.

Esta industria integra sensores que trabajan junto con la maquinaria interconectados entre sí, produciendo una cantidad importante de datos, “Big Data” que, permite a los fabricantes investigar tendencias históricas, identificar patrones y mejorar su toma de decisiones.

Tanto la calidad como la seguridad de la producción son dos objetivos esenciales en la industria moderna, lo que demanda la implantación de métodos de supervisión de la planta, y por tanto de métodos de detección y diagnóstico de fallos. Se pueden distinguir tres métodos: los métodos basados en conocimiento, los basados en modelos o métodos analíticos, y los métodos basados en datos.

La cantidad de datos de la que se hablaba anteriormente, es difícil de tratar e interpretar, por lo que, en este trabajo nos centraremos en la detección y diagnóstico de fallos a partir de métodos basados en datos.



El control estadístico de los procesos (*Statistical Process Control*, SPC) es una de las técnicas más utilizadas para la detección de fallos, que utiliza estadísticas y gráficos multivariados. Entre ellas se encuentra el *Análisis de Componentes Principales* (PCA), el cual reduce la dimensionalidad, se encarga de capturar la variabilidad de los datos y realiza una extracción de características. Para realizar la detección de fallos con este método se emplean dos estadísticos, uno para monitorizar los primeros “a” componentes principales (el estadístico de Hotelling’s  $T^2$ ) y otro para los residuos (el estadístico Q).

Otras de las técnicas que se usan es el *Análisis Discriminante de Fisher* (FDA), que permite también reducir la dimensionalidad, obteniendo un número de variables menor que son en las que realmente se encuentra la mayor parte de la información, o también, la incrustación de vecinos estocásticos distribuidos (t-SNE), técnica no lineal basada también en esa reducción de dimensiones. Además, se usan técnicas basadas en el aprendizaje automático como las Redes Neuronales, que consisten en un conjunto de nodos interconectados que transmiten señales, y aprenden características de la planta a través de los datos. Entre ellas cabe destacar las Redes de retropropagación (*Back-Prop*, BP), usadas en este proyecto. Otra técnica de aprendizaje automático muy utilizada en la literatura científica para problemas de clasificación, son los árboles de decisión y en concreto los bosques aleatorios, ‘random forest’ que será otra de las técnicas utilizadas en este trabajo para la clasificación/diagnóstico de fallos, considerando cada fallo que puede producirse en la planta como una clase.

## 1.2. OBJETIVOS:

Obtener una visualización del estado operativo de un proceso industrial es clave para identificar y diagnosticar fallos de manera rápida y eficaz, y por tanto para asegurar el control de calidad del mismo. Luego, en este trabajo se pretende desarrollar un método de detección y diagnóstico de fallos que detecte y diagnostique cualquier anomalía o funcionamiento defectuoso en nuestra planta.

Para cumplir con este objetivo, se realizará mediante la técnica de Análisis de componentes principales (PCA) una detección de anomalías. Posteriormente se procederá a la clasificación de estos mediante este método estudiado, que combinará técnicas como es el Análisis discriminante de Fisher (FDA), el implante de Vecinos Estocásticos Dispersos (t-SNE) y redes neuronales artificiales (NN), dado que t-SNE carece de una matriz de transformación durante la reducción de dimensionalidad.



Después, se hará una comparación entre este método FDA-tSNE-NN con otro método de aprendizaje automático, denominado ‘*Random Forest*’ (bosque aleatorio), determinando cual de estos es más efectivo a la hora de hacer la diagnosis/clasificación de fallos de una planta industrial.

Los datos con los que se trabajarán provienen de la planta Tennessee Eastman, una planta química usada como benchmark en la literatura científica, y en la cual se realizan pruebas de control y de monitorización del proceso.

### 1.3. ORGANIZACIÓN DE LA MEMORIA:

Con fin de mantener una estructura ordenada dentro del trabajo, este ha sido dividido en seis capítulos diferentes:

Un primer capítulo donde se hará una pequeña introducción sobre el tema que se trata, destacando así las técnicas de detección y clasificación de fallos. Y donde también se escribe sobre los objetivos del trabajo y organización de la memoria.

En un segundo capítulo se tratará de manera teórica el control de calidad y las diferentes técnicas usadas para la detección y diagnóstico de fallos, así como han sido implementadas estas técnicas para el análisis de los datos.

En un tercero, se habla sobre la planta industrial de donde se han obtenido los datos, la planta Tennessee Eastman. Explicando su proceso industrial, las variables de la planta y los fallos que se van a producir e intentar detectar.

En el cuarto se desarrollará la aplicación práctica del proyecto. Se explicará de manera detallada los pasos que se han ido realizando para llegar a los resultados de la detección de fallos y su posterior diagnóstico, exponiendo en el último apartado del capítulo una comparación de los resultados obtenidos.

En el quinto, se presentarán las conclusiones a las que se ha llegado después de los resultados obtenidos y que fueron analizados en el anterior capítulo, y también plantean algunas ideas para realizar en un trabajo futuro.

Por último, en un sexto capítulo se documentarán todas las referencias que han sido consultadas en la realización de este trabajo.



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS INDUSTRIALES

Universidad de Valladolid



# CAPITULO II: ESTUDIO TEORICO



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA  
INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE  
INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS  
INDUSTRIALES

Universidad de Valladolid



## 2.1. CONTROL DE CALIDAD:

El control de calidad es un conjunto de tareas que tratan de verificar que un producto sea seguro, útil y que cumpla con todas sus funciones de manera adecuada, además de garantizar que todas las actividades y acciones emprendidas para el desarrollo del mismo, satisfagan las necesidades del cliente y los estándares de la empresa, ya sea materiales, costos, o tiempos de producción, entre otros.

Esto lo define como una herramienta capaz de unificar el proceso productivo y garantizar los estándares y requisitos en cada etapa de elaboración e incluso distribución de los productos.

Por eso es indispensable la realización de un control estadístico que establezca variables y controle cada etapa del proceso productivo, genere frecuencias, mediciones, intervalos, rangos, probabilidades, correlaciones, atributos y demás datos de control que permitan la producción de un producto sin errores. Para ello, el control de calidad se debe aplicar a todos los procesos de la empresa [1].

Cuando la calidad de un producto se ve alterada, es decir, hay algo que provoca que este no cumpla con los estándares establecidos, es necesario encontrar el origen del problema. Ya sea si estos han sido causados por variaciones en la materia prima, fallas en la maquinaria, errores humanos, el uso incorrecto de métodos o procedimientos de producción, o bien, inspecciones fallidas.

El concepto de calidad siempre ha ido ligado a la esencia del ser humano. Desde los orígenes, el hombre comprende que hacer las cosas de manera correcta y lo mejor posible le da la ventaja competitiva sobre el resto de individuos y sobre el entorno con el que se relaciona. Hoy en día, los cambios en el panorama empresarial mundial hacen que la calidad deje de ser una novedad como se percibía en los años anteriores, para convertirse en algo esencial que debe de abordar cualquier empresa que pretenda mantenerse y perdurar en el tiempo. La llegada de la cuarta Revolución Industrial, o también llamada Industria 4.0, ha llevado el panorama empresarial a la digitalización, potenciando el uso de máquinas y fábricas inteligentes. Pero esto introduce una complicación, y es que ahora estas fábricas generan un gran número de datos cada día, haciendo un uso inadecuado de la mayoría de estos. Para abordar este problema, la transformación digital está cambiando a métodos combinados de recopilación en línea y fuera de línea que permiten a las organizaciones tomar medidas basadas en información en tiempo real [2].

Gracias a estos avances en los sistemas de control y a la conectividad entre las máquinas es posible llevar a cabo una monitorización del proceso de manera más



efectiva. La monitorización del proceso es una de las etapas presentes en el control de calidad, en la cual se adquiere información del proceso, se detectan y se realiza un posterior diagnóstico de los fallos (FDD), y, por último, se actúa sobre el proceso para devolverlo a su estado habitual.

Una de las técnicas más habitual utilizada en la detección y diagnóstico de los fallos basado en datos recopilados de las plantas es el Control Estadístico de Procesos (SPC), aunque el avance del Machine Learning y la inteligencia artificial está haciendo que cada vez sea más habitual ver estas últimas técnicas en plantas industriales modernas, visualizando un aumento de las capacidades en la monitorización de los procesos.

## 2.2. CONTROL ESTADÍSTICO DE PROCESOS:

La variabilidad en los procesos industriales es uno de los motivos de la ausencia de calidad que puede sufrir una entidad productora. Si no se lleva a cabo una monitorización adecuada, se elevará la posibilidad de que ocurra algún fallo que afecte directamente a la producción y también a la empresa. Las técnicas que definen el Control Estadístico de Procesos serán de gran utilidad a la hora de detectar y terminar con este tipo de desviaciones.

El Control Estadístico de Procesos (SPC) tiene como objetivo hacer predecible un proceso en el tiempo. Las herramientas usadas para este fin son las gráficas de control que permiten distinguir causas especiales de las causas comunes de variación. Luego de identificarlas con el gráfico, el paso siguiente es eliminar las causas especiales, ya que son ajenas al desenvolvimiento natural del proceso con lo que se logra el estado de Proceso Bajo Control Estadístico; es decir, un proceso predecible y afectado exclusivamente por causas comunes de variación [3].

El SPC se puede aplicar a cualquier proceso productivo donde el producto tenga unas especificaciones medibles. Al implementarlo se toman muestras aleatorias del proceso productivo con las que posteriormente se realizarán los gráficos de control, es decir, permite garantizar estabilidad y el cumplimiento de las especificaciones requeridas por el cliente sin tener que inspeccionar el 100% de la producción.

El Control Estadístico de procesos, además de usarse para alertar de las fallas que hacen un producto o servicio defectuosos, se suele utilizar también para informar a la dirección sobre cambios introducidos en los procesos que hayan afectado de manera positiva en el resultado de la producción.



### 2.2.1. VARIABILIDAD EN LOS PROCESOS:

Una de las dificultades más frecuentes en la mejora de la calidad es la presencia continua de variaciones en los procesos que afectan directamente en los productos o servicios ofrecidos. Del mismo modo que dos personas nunca son exactamente iguales, dos partes manufacturadas o dos servicios entregados nunca serán exactamente iguales. Siempre habrá una variación en el tamaño, peso, ajuste o tiempo de entrega de un material, una pieza, un ensamble o un servicio. La diferencia puede ser muy pequeña o difícil de medir, pero siempre existirá. Esto es lo que recibe el nombre de Variabilidad, [4].

Una de las herramientas utilizadas para visualizar de manera gráfica las causas que explican la variabilidad en un proceso es el Diagrama de Ishikawa o Diagrama de Causa Efecto, también conocido como Diagrama de Espina de Pescado o Diagrama de las 6M. Denominación que refleja los 6 posibles factores que serán analizados de manera individual para identificar cual de ellos causa el efecto general (Figura 2.1).

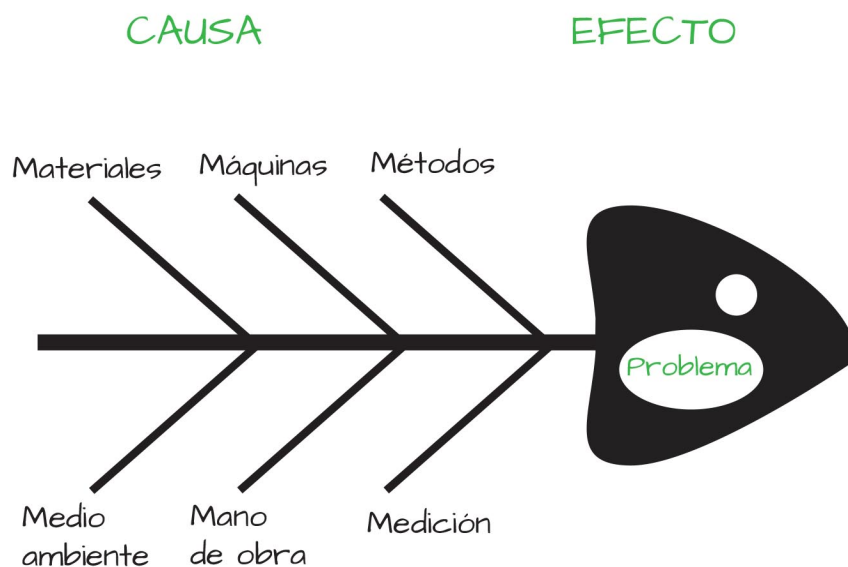


Figura 2.1. Diagrama de las 6M [5].

A continuación, veremos cómo se clasifica esta variabilidad:

- **Variabilidad en la calidad por causas comunes**  
Las causas comunes de variación están relacionadas directamente con el proceso, afectando a todo lo relacionado con él. Este tipo de causas son



predecibles, ya que son originadas por la variación normal que puede existir entre el personal, herramientas, materia prima, medio ambiente, instrumentos de medición o por los métodos que se llevan a cabo.

Cuando toda la variación en un sistema se debe a este tipo de causas se dice que el sistema se encuentra dentro de control estadístico. Para disminuir la variación que generan se requiere el cambio de algunas de las características inherentes del proceso. Este tipo de variabilidad, por lo regular, proporciona oportunidades de mejora en mediano y largo plazo.

- **Variabilidad en la calidad por causas especiales**

Las causas especiales de variación son sucesos fuera de lo habitual en las condiciones en las que se trabaja, no tienen relación con el diseño. Ocurren en circunstancias puntuales no afectando a todo el proceso. Se puede decir que son impredecibles, ya que no se sabe cuándo puede presentarse otro fallo de este tipo y por ello también impredecible el efecto que tendrá.

Los fallos generados serán detectables y con ello, deberán de ser eliminados devolviendo el proceso a su estado de funcionamiento normal.

En la Figura 2.2, en la parte izquierda podemos observar cómo las causas comunes afectan al proceso. Con motivo de que el proceso permanece sin cambios, el promedio, la varianza y la forma se puede controlar y predecir a través del tiempo. En la parte derecha de la imagen se observa el proceso afectado por causas especiales, afectando al promedio, la varianza y la forma de la distribución normal. Por lo que en dichas causas el comportamiento es impredecible a través del tiempo.

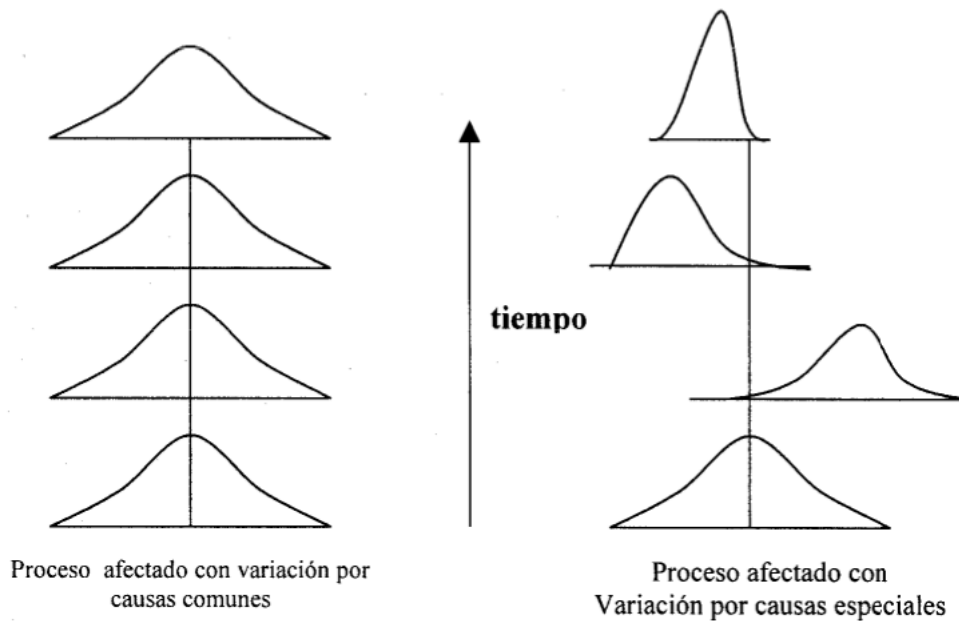


Figura 2.2. Variabilidad por causas comunes y especiales [6].

Una vez que los procesos están afectados únicamente por las causas predecibles, pueden ser controlados a través del Control Estadístico de Procesos, beneficiando así su monitorización y comprobar que se mantiene en control. En todo proceso es necesario medir la variación para poder estar informado de cómo se está produciendo, y mostrar a los clientes que se está cumpliendo con las especificaciones requeridas [6].

Para ello, se definen unos límites superior e inferior dentro de los cuales se cumplen las especificaciones. En la Figura 2.3 se muestran 3 tipos de salida del proceso. En la Figura 2.3 (a) podemos ver un proceso fuera de control por causas de variación especiales. En la Figura 2.3 (b) vemos un proceso bajo control, el cual está afectado únicamente por causas naturales de variación pero que se encuentra fuera de los límites que se han especificado, por lo que la calidad de producción no será lo suficientemente adecuada. En la Figura 2.3 (c) vemos un proceso afectado solamente por causas de variación naturales y que se encuentra dentro de los límites de especificación, luego se dice que será capaz de producir con la calidad adecuada.

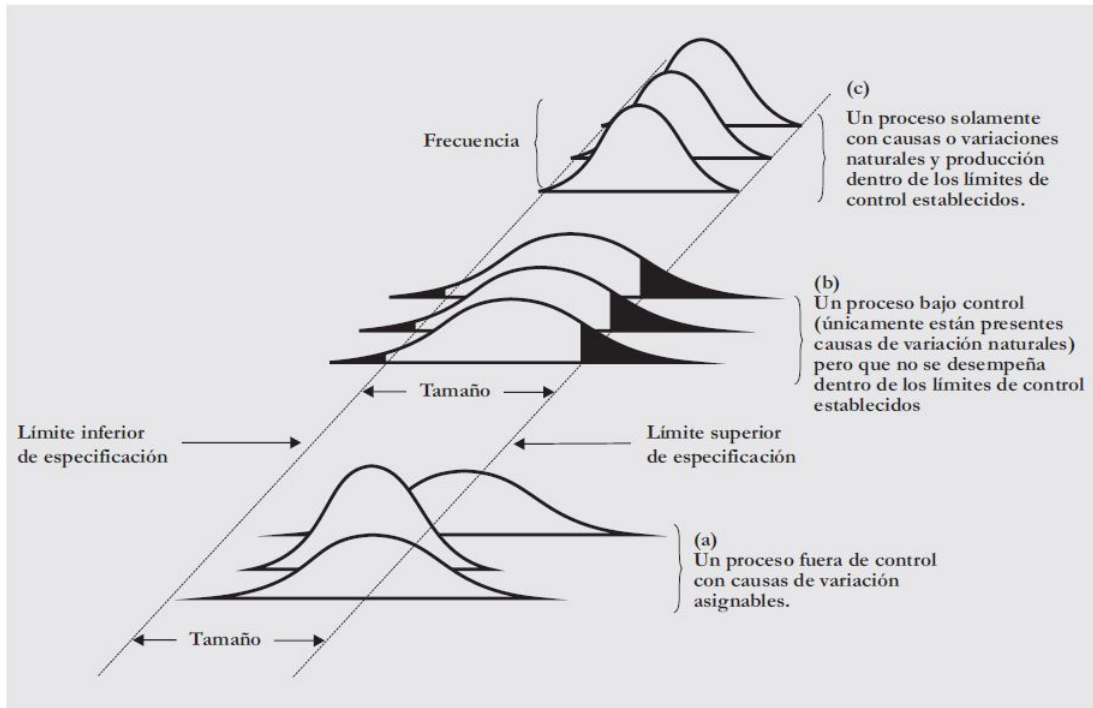


Figura 2.3. Control del proceso: tres tipos de salida del proceso. (a) Proceso fuera de control con causas especiales de variación. (b) Proceso bajo control con causas comunes de variación, pero fuera de los límites de especificación. (c) Proceso bajo control con causas comunes de variación y dentro de los límites establecidos [7]

## 2.2.2. GRÁFICOS DE CONTROL:

Las gráficas de control se usan para determinar estadísticamente si las variaciones presentes en un proceso productivo son de carácter normal o hay anomalías. Para ello se mide y se traza la gráfica de una característica de calidad tomada de la planta mediante un proceso de muestreo (Figura 2.4). El eje vertical muestra la magnitud, y el eje horizontal simboliza el tiempo o el número de la muestra.

La gráfica consta de La Línea Central (LC), que por lo general representa el valor medio de la característica bajo estudio y objetivo que se desea alcanzar, y las Líneas Superior (UCL, del Inglés, Upper Control Limit) e Inferior de Control (LCL, del Inglés, Lower Control Limit), límites de control que, si una muestra se encuentra entre ellos, indicará que el proceso se encuentra ante causas comunes de variación, en cambio, si hay puntos aislados que caen fuera de los límites serán debidos a causas especiales de variación impredecibles, que serán sobre los que se debe poner atención y eliminar sus causas para poder llevar de nuevo al sistema a su comportamiento normal.

En caso de que sean varios los puntos que caen fuera de los límites, puede significar que el proceso es inestable y que está fuera de control. En este caso es posible que se necesiten herramientas más avanzadas para poder analizarlo con más detalle.

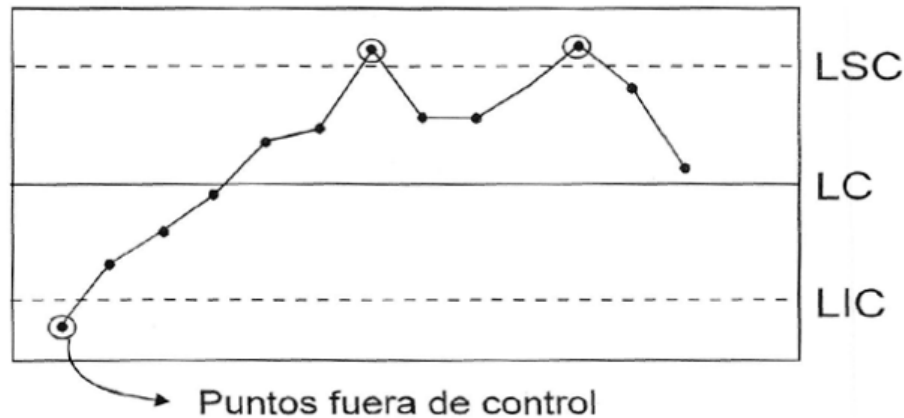


Figura 2.4. Gráfico de Control. Se muestra la Línea Central (LC) y los Límites tanto Superior (LCS) como Inferior (LCI) [8].

En este tipo de gráficos se suelen tomar como límites:

$$\begin{aligned}LCL &= LC - 3 \cdot \sigma \\LC &= \mu \\UCL &= LC + 3 \cdot \sigma\end{aligned}$$

El procedimiento que se suele llevar a cabo para vigilar el proceso mediante estos gráficos es el siguiente:

1. Se toma una muestra aleatoria del proceso y se mide la característica a analizar, calculándose una medida variable.
2. Si la estadística sobresale de los límites de control, busca una causa asignable.
3. Se elimina la causa si ésta perjudica la calidad del proceso, o bien se incorpora si con ella se viese mejorada.
4. Repetir reiteradamente el procedimiento.

Este tipo de gráficos recibe el nombre de Gráficos de control de Shewhart, pero, por otra parte, también podemos encontrar Gráficos de control por suma acumulativa (Cumulative Sum, CUSUM), los cuales son una buena alternativa para medir de una manera más exacta pequeñas variaciones en los datos, de modo



que tienen una mayor precisión a través del tiempo que los gráficos tipo Shewhart. O también, encontramos los Gráficos de control EWMA (promedio móvil ponderado en forma exponencial), ideales para graficar mediciones de manera individual. Se asemeja bastante al anterior mencionado con la diferencia de que la gráfica es un promedio móvil ponderado de la muestra y lo compara con su media ideal para así determinar su estabilidad. Que los límites estén también ponderados, hacen que la precisión de las mediciones sea aún mayor.

### 2.3. ANÁLISIS DE COMPONENTES PRINCIPALES (PCA):

Cuando se recoge información de una muestra de datos, lo más habitual es tomar el mayor número de variables posible. No obstante, si el número de variables es demasiado grande, la dificultad de visualizar las relaciones que tienen entre ellas aumenta. Otro inconveniente que tiene es la fuerte correlación que se puede dar entre ellas, es decir, el hecho de que tomemos muchas variables hace que sea normal que muchas de ellas estén relacionadas o que midan lo mismo bajo distintos puntos de vista.

Es por ello que se hace necesario reducir el número de variables. Cabe destacar que el concepto de más información está relacionado con el de mayor variabilidad o varianza, es decir, se considera que existe más información cuanto mayor sea la variabilidad de los datos.

Una técnica que se desarrolló para ello es el Análisis de Componentes Principales (PCA). Trata de estudiar las relaciones que se presentan en  $p$  variables correlacionadas transformando el grupo inicial de variables en otro nuevo de variables desconectadas entre sí llamado componentes principales.

Estas nuevas variables son combinaciones lineales de las de partida y se irán formando según el orden de importancia en cuanto a la información que recogen de la muestra. De modo que se buscan  $m < p$  variables que sean combinaciones lineales de las  $p$  iniciales y que no tendrán correlación entre ellas, recogiendo aun así la mayor parte de la información de los datos.

Una forma evidente de entender el proceso de PCA es interpretar las componentes principales desde el punto de vista geométrico (Figura 2.5). Suponiendo un conjunto de observaciones para las que se dispone de dos variables ( $X_1, X_2$ ), el vector que define la primera componente principal ( $Z_1$ ) sigue la dirección en la que las observaciones varían más (línea roja). La proyección de cada observación sobre esa dirección equivale al valor de la primera componente para dicha observación [9].

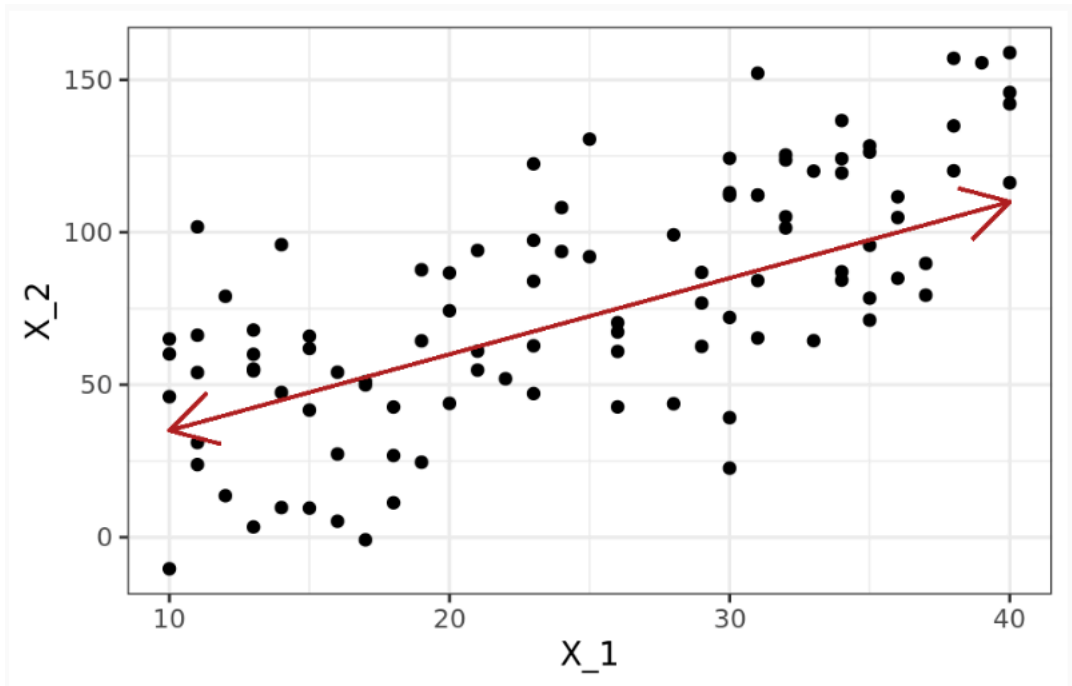


Figura 2.5. Interpretación geométrica PCA. Dirección de la primera componente principal

La segunda componente ( $Z_2$ ) (Figura 2.6) sigue la segunda dirección en la que los datos muestran mayor varianza y que no está correlacionada con la primera componente (línea verde) [9].

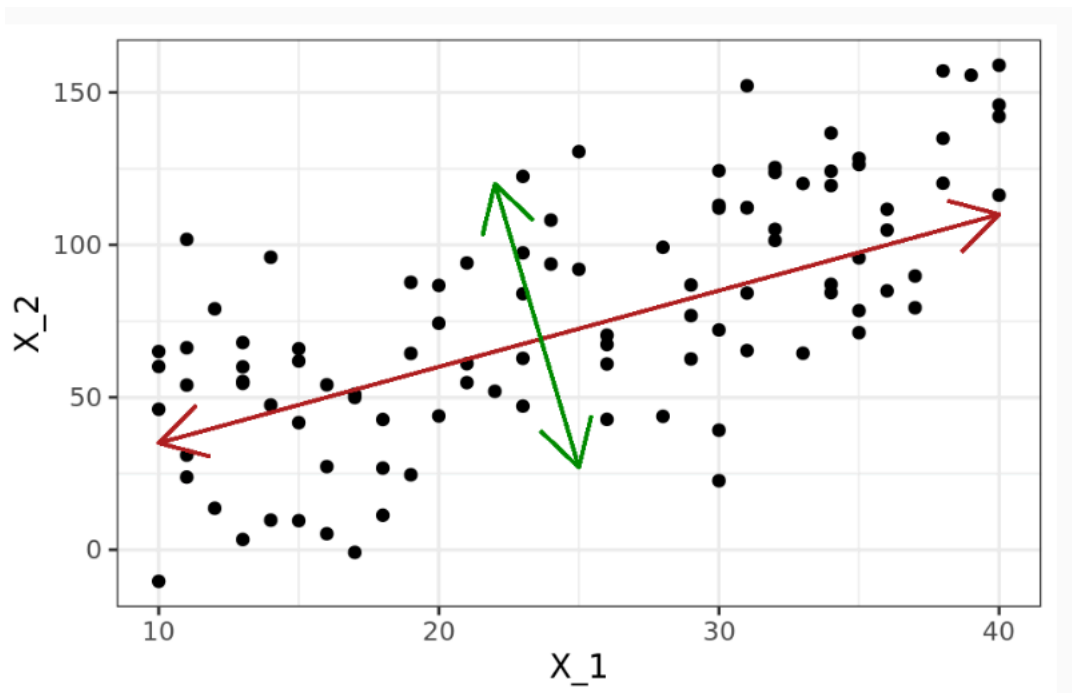


Figura 2.6. Interpretación geométrica PCA. Ejes de un elipsoide

Se describe ahora la forma de calcular los componentes principales. Dado un conjunto de datos  $X$  con  $n$  observaciones y  $m$  variables, lo primero de todo será hacer un pre-tratamiento de los datos, eliminando las variables inadecuadas que no aportan información real del proceso, como pueden ser los que tienen errores de medida muy grandes. Además, debemos de realizar una estandarización de los datos a media cero y varianza uno, de manera que tengan una magnitud comparable entre ellas, evitando así el problema de que unas variables tengan más influencia en el proceso de monitorización que otras.

Terminado el pre-tratamiento se pasa a aplicar el método a una matriz de datos  $X$  cuyas dimensiones son  $n \times m$ .

$$X = \begin{pmatrix} X_{11} & \cdots & X_{1m} \\ \vdots & \ddots & \vdots \\ X_{n1} & \cdots & X_{nm} \end{pmatrix} \quad (1)$$

A partir de esta, se calcula su matriz de correlación, con el objetivo de obtener los vectores propios ortogonales ordenados según la variabilidad que aporten al sistema.

$$R = \frac{1}{n-1} \cdot X^T \cdot X \quad (2)$$

Posteriormente, se realiza la descomposición de  $R$  en valores singulares, de manera que se obtienen los valores propios reales, almacenados en la matriz  $\Lambda$ . Y además los vectores propios, llamados también vectores de carga, y almacenados en la matriz  $V$ :

$$R = V \cdot \Lambda \cdot V^T \quad (3)$$

$V$  se trata de una matriz ortogonal de dimensión  $m \times m$  cuyas columnas corresponden a los vectores propios de  $X$  (o vectores de carga), y que en PCA normalmente reciben el nombre de 'scores'. La matriz  $\Lambda$  de dimensiones  $m \times m$  es una matriz que contiene los valores propios reales no negativos en orden decreciente de  $R$  de la manera  $\lambda_1 \geq \lambda_2 \geq \lambda_m \geq 0$  y donde cada uno de estos valores propios es igual al cuadrado del valor propio de la matriz  $X$ :  $\lambda_i = \sigma_i^2$ .

Con el objetivo de reducir la dimensionalidad del sistema manteniendo la máxima información posible de los datos, se escogen ' $a$ ' vectores propios, correspondientes a los ' $a$ ' valores propios más grandes. Nos quedaremos con los más grandes hasta que la varianza acumulada sea mayor o igual que un porcentaje cuyo valor será a elegir, y el cual recibirá un valor de 90% en la realización de este trabajo.



$$\frac{\lambda_1}{\text{traza}(\Lambda)} \cdot 100 + \frac{\lambda_2}{\text{traza}(\Lambda)} \cdot 100 + \frac{\lambda_3}{\text{traza}(\Lambda)} \cdot 100 + \dots \geq 90 \quad (4)$$

Los vectores propios escogidos serán almacenados en una matriz  $P$  de dimensiones  $m \times a$ , que junto a la matriz de datos  $X$  se logrará proyectar los datos iniciales a un nuevo espacio reducido. Obtenemos así la matriz de componentes principales  $T$ , de dimensión  $n \times a$ .

$$T = X \cdot P \quad (5)$$

Si se define cada columna  $i$ -ésima de  $T$  como  $t_i$  obtenemos las siguientes características [10]:

- $\text{var}(t_1) \geq \text{var}(t_2) \dots \geq \text{var}(t_a)$  es decir, la varianza se encuentra ordenada de mayor a menor.
- Se encuentra centrado en la media:  $\mu(t_i) = 0 \forall i$
- Presenta descomposición ortogonal:  $t_i^T \cdot t_j = 0; \forall i \neq j$
- No existe otra expansión ortogonal que contenga  $a$  componentes y tenga más información de la variación de los datos analizados.

Se puede volver al espacio original sin reducir, a partir de las matrices  $T$  y  $P$  obtenidas, obteniendo la matriz original  $X$  recalculada:

$$\hat{X} = T \cdot P^T \quad (6)$$

Esta última nos permitirá obtener la matriz de residuos  $E$  de la siguiente forma:

$$E = X - \hat{X} \quad (7)$$

De esta forma, podemos entender que los datos de partida son una composición de una parte más significativa y por otra parte que actúa como ruido o residuo.

$$X = \hat{X} + E \quad (8)$$

### 2.3.1. ESTADÍSTICOS EMPLEADOS EN LA DETECCIÓN DE FALLOS:

Para llevar a cabo el control del estado del proceso a partir del análisis de componentes principales, se hace uso de dos estadísticos que permitirán crear gráficos de control con los que la inspección será realizada de manera multivariable, es decir teniendo en cuenta todas las variables, pero mostrando toda

la información en dos gráficas y que además posibilitan observar el umbral del estado normal, todo ello en la misma gráfica.

Estos estadísticos son:

- Estadístico de Hotellings ( $T^2$ ):  
Se puede definir este estadístico de la siguiente forma:

$$T^2 = x^T P \Lambda_a^{-1} P^T x \quad (9)$$

Donde  $x \in \mathcal{R}^{m \times 1}$  es cada una de las filas que forman la matriz  $X$ , y  $\Lambda_a$  está formado por las  $a$  primeras filas y columnas de  $\Lambda$ , siendo  $a$  el número de componentes principales.

Para saber si se ha producido un fallo, se tendrá que definir un umbral. Si es superado querrá decir que ha ocurrido el fallo, suponiendo entonces que el proceso está fuera de control estadístico, aunque también puede ocurrir que se trate de una falsa alarma. El umbral se calculará de la siguiente forma:

$$T_a^2 = \frac{(n^2-1) \cdot a}{n \cdot (n-a)} F_\alpha(a, n-a) \quad (10)$$

Donde  $n$  es el número de observaciones, y  $F_\alpha(a, n-a)$  es el valor crítico de la distribución  $F$  (distribución de Fisher-Snedecor) con  $n$  y  $n-a$  grados de libertad y  $\alpha$  el nivel de significancia, que especificará el grado de compromiso de falsas alarmas [11].

El estadístico  $T^2$  está basado en los  $a$  primeros componentes principales, de tal forma que los fallos detectados sólo se ofrecerán si la variación de las variables latentes es mayor que la que se da por causas comunes. Es por ello que se hace uso de otro tipo de estadístico, denominado  $Q$  o SPE, que detectará eventos debidos a las variables restantes, es decir, a las que hemos considerado anteriormente como ruido.

-Estadístico  $Q$  o SPE:

El estadístico se define de la siguiente forma:

$$Q = r^T r \quad (11)$$

Siendo  $r$  el vector de residuos:

$$r = (I - PP^T)x \quad (12)$$

Al igual que con el estadístico anterior, se necesita un umbral que nos indique cuándo se produce un fallo. Este umbral será calculado fuera de línea de la siguiente forma:

$$Q_{\alpha} = \theta_1 \left[ \frac{h_0 c_{\alpha} \sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_0 (h_0 - 1)}{\theta_1^2} \right]^{\frac{1}{h_0}} \quad (13)$$

Siendo  $c_{\alpha}$  el valor de la distribución de probabilidad normal con media 0 y desviación estándar 1. Y  $\alpha$  el nivel de significancia al igual que en el caso anterior.

Con el método PCA descrito en esta sección se pueden detectar fallos o anomalías en la planta, pero no se puede saber la causa que los produjo, es decir, diagnosticar el fallo, para poder hacer este paso, diagnosticar, se van a usar diferentes técnicas de clasificación, donde cada posible fallo se considerará como una clase. Ahora se explicarán diferentes métodos usados para clasificar/diagnosticar los fallos que ocurren en la planta, una vez detectados por el PCA.

## 2.4. MÉTODO FDA (ANÁLISIS DISCRIMINANTE DE FISHER):

El Análisis Discriminante Lineal (ADL) es una metodología usada en estadística que permite reconocer patrones basándose en el aprendizaje automático, de tal forma que trata de encontrar relaciones lineales entre características que califiquen o separen dos o más clases de objetos o eventos.

Este Análisis Discriminante Lineal es una generalización del Análisis Discriminante de Fisher (FDA). Una aproximación en la que se reduce un espacio de  $p$  dimensiones a otro de menos dimensiones formado por combinaciones lineales de las variables que mayor influyen en la separación de las clases. Seguidamente de obtener dichas combinaciones se realizará la clasificación en este subespacio, siendo idóneo aquel que maximiza la distancia entre grupos desde el punto de vista de la varianza.

El procedimiento llevado a cabo en este método es el siguiente:

Partiendo de la base que la matriz  $X$  de dimensiones  $n \times m$  contiene  $c$  clases, de tal forma que  $X_i$  es un subconjunto de  $X$  que contiene  $n_i$  muestras y que pertenece a la clase  $i$ , el primer paso será calcular los vectores medios de las diferentes clases y de la matriz  $X$  original.

$$\bar{x}_i = \frac{1}{n_i} \sum_{x \in X_i} x, i = 1, 2, \dots, c \quad (14)$$

$$\bar{x} = \frac{1}{n} \sum_{x \in X} x \quad (15)$$

A continuación, se calcularán la matriz de dispersión de la clase:

$$S_w = \sum_{i=1}^c S_i \quad (16)$$

Donde:

$$S_i = \sum_{x \in D_i} (x - \bar{x}_i)(x - \bar{x}_i)^T \quad (17)$$

Y la matriz de dispersión entre clases:

$$S_b = \sum_{i=1}^c n_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T \quad (18)$$

Conseguimos así una matriz cuadrada  $S_w^{-1}S_b$ , de la que haremos una descomposición de valores propios para así obtener los vectores propios mediante la siguiente ecuación:

$$S_b w = \lambda S_w w \quad (19)$$

Siendo  $\lambda$  el valor propio. Obtenemos la matriz de vectores propios en  $w$  correspondientes a los  $c-1$  autovalores superiores.

## 2.5. MÉTODO t-SNE:

Otro de los métodos que se han utilizado en la realización de este trabajo es la incrustación de vecinos estocásticos distribuidos en  $t$ , o también denominado t-SNE (t-Distributed Stochastic Neighbor Embedding). Se trata de un algoritmo no directo diseñado para la disminución de la dimensionalidad y la visualización de un conjunto de datos de alta dimensión.

Este método se trata de una mejora de su predecesor, técnica denominada SNE (Stochastic Neighbor Embedding), que se encarga de representar objetos de gran dimensión en el plano, manteniendo la relación entre los próximos. Se comienza construyendo una distribución de probabilidad sobre parejas de datos en el espacio original, de tal manera que cuando la pareja de datos comparados es semejante reciben alta probabilidad de ser escogidos, mientras que si son muy diferentes recibirán una baja probabilidad de selección. En otras palabras, la similitud entre un punto  $x_j$  y el punto  $x_i$  es la probabilidad condicional de  $x_i$  eligiera a  $x_j$  como su vecino. En segundo lugar, los puntos del espacio de alta dimensión se llevarán a

otro espacio de baja dimensionalidad, definiendo una distribución de probabilidad semejante en virtud del espacio reducido, y tratará de minimizar la disparidad entre las probabilidades condicionadas. Es decir, SNE intenta reproducir la distribución que existía en el espacio original de alta dimensión en el espacio de menor dimensión.

Como se ha mencionado anteriormente, el método t-SNE perfecciona esta técnica consiguiendo mejores representaciones en la baja dimensión con tan solo mínimos cambios en su algoritmo de funcionamiento.

En primer lugar, al igual que en SNE, el primer paso del algoritmo es transformar las distancias euclidianas entre pares de datos en probabilidades condicionales manteniendo la similitud de los próximos. En lugar de ver la similitud de dos puntos  $x_i$  y  $x_j$  mediante la distancia euclidiana, se mostrará mediante la probabilidad condicional  $p_{ij}$  de que  $x_j$  fuese seleccionado como vecino si los puntos perteneciesen a una distribución de densidad gaussiana centrada en  $x_i$ . La probabilidad  $p_{ij}$  será alta para puntos cercanos, mientras que será baja para puntos alejados. Esta probabilidad  $p_{ij}$  mencionada se expresa de la siguiente forma:

$$p_{ij} = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / 2\sigma_i^2)} \quad (20)$$

Donde  $k$  es el número efectivo de vecinos.

El siguiente paso es obtener otros puntos  $y_i, y_j$  equivalentes a  $x_i, x_j$  en una dimensión menor, y cuya distancia se define como la probabilidad condicional  $q_{ij}$ :

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq i} (1 + \|y_k - y_l\|^2)^{-1}} \quad (21)$$

Lo ideal sería que  $p_{ij} \approx q_{ij}$  para que el espacio de menor dimensión fuera igual que el inicial y así poder trabajar con él de la misma forma que trabajaríamos en el espacio de partida, pero siempre hay una pequeña diferencia entre ambas probabilidades. Para rebajar esta diferencia lo máximo posible se usará la divergencia de Kullback-Leibler (KL) la cual hace las veces de función de costo:

$$C = KL(P|Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \quad (22)$$

Que con su ecuación de descenso de gradiente se minimizarán esas distancias consiguiendo que  $p_{ij}$  y  $q_{ij}$  sean similares.

$$\frac{dC}{dy_i} = 4 \sum_{j \neq i} (p_{ij} - q_{ij}) q_{ij} (y_i - y_j) (1 + \|y_i - y_j\|^2)^{-1} \quad (23)$$

## 2.6. TÉCNICAS DE APRENDIZAJE AUTOMÁTICO:

Una vez que el fallo ha sido detectado, se procederá a hacer un estudio del mismo para hacer una valoración de cuál es su causa y origen. Existen una serie de técnicas que constan de capacidad para adaptarse y tomar decisiones en base a lo que aprenden viendo datos. Estas técnicas reciben el nombre de Machine Learning o aprendizaje automático, y nos permiten realizar una clasificación de las anomalías en diferentes tipos de clases mediante la identificación de patrones.

Los algoritmos de Machine Learning emplean métodos computacionales para “aprender” información de los datos tras la búsqueda de patrones naturales en ellos, y mejorar en la toma de decisiones, predicción de comportamientos o detección de similitudes o anomalías. Existen dos tipos fundamentales de aprendizaje automático, el supervisado, y el no supervisado. El aprendizaje supervisado se basa en aprender a partir datos de entrada de los cuales también conocemos su respuesta (salida) y entrenar un modelo para generar predicciones para la respuesta de nuevos datos. Dentro de este, hay técnicas que permiten clasificar a los conjuntos de datos asignándolos en categorías, y técnicas de regresión, que predicen respuestas numéricas. El aprendizaje no supervisado trata de encontrar patrones ocultos en los datos, es decir, se utiliza para conjuntos de datos de los cuales conocemos las entradas, pero las respuestas no son conocidas.

Por otro lado, un tipo especializado de Machine Learning es el Deep Learning, o aprendizaje profundo, que se centra en el aprendizaje por capas, es decir, procesan la información recibida en varias etapas sucesivas permitiendo ser capaz de tomar decisiones complejas con gran precisión.

### 2.6.1. REDES NEURONALES ARTIFICIALES:

La Red Neuronal Artificial (ANN por sus siglas en inglés, Artificial Neural Network) es un modelo computacional inspirado en la estructura y funcionamiento del cerebro humano [12]. El objetivo de estas redes es crear un modelo capaz de aprender a partir de la experiencia, al igual que lo hacen nuestras neuronas, es decir, aprender a partir de unos datos iniciales. Estas redes consisten en capas de nodos que, contienen una capa de entrada, una o varias capas ocultas y una capa de salida. De tal forma que cada nodo, o cada unidad de neurona artificial (Figura 2.7), se conecta a otro y tiene un peso y un umbral asignado.

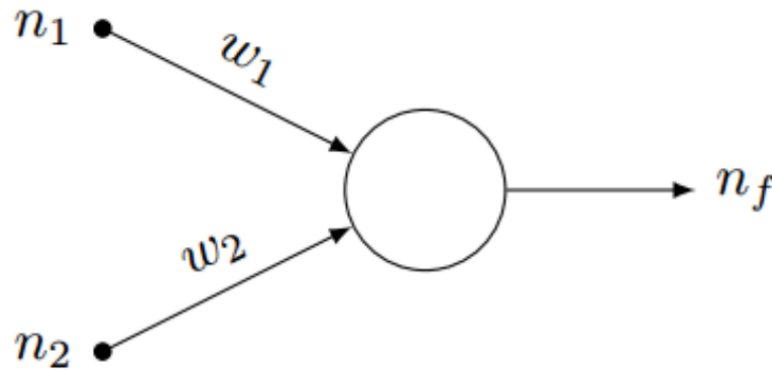


Figura 2.7: Esquema de neurona artificial [12]

Como vemos en la anterior figura, cada unidad neuronal artificial va a estar formada por:

- Unos valores de entrada ( $n_i$ ), que contienen la información exterior que entra a la red, o aquella que procede de otras neuronas. Estos valores se van a ir transformando a través de un conjunto de operaciones matemáticas.
- Unos pesos de las entradas ( $w_i$ ), que son los valores que se multiplican con los valores de entrada, y como se ha mencionado anteriormente, modifican los valores determinando la importancia de la información recibida por cada conexión. Se trata de un valor que poco a poco, a medida que la red neuronal es entrenada, son optimizados para dar los mejores resultados posibles.
- Un bias ( $b$ ) asociado a cada una, que representa cómo de alejado está el resultado real del deseado. Un bias bajo significará que el resultado se asemeja al valor deseado.
- Un conjunto de funciones, cuya misión es controlar el aprendizaje de la neurona, o generalmente de la red neuronal. Estas funciones relacionan las entradas ponderadas con sus respectivos pesos para obtener el valor de salida de la neurona. Específicamente, lo que hace es sumar los valores ponderados que le llegan, y el valor resultante, junto con el bias, se pasa por una función de activación que determina si la neurona está activa o no, es decir, si produce un valor de salida o no. Hay muchas funciones de activación, pero las más usadas son la sigmoide y la ReLu.

- Una salida donde se entrega al exterior, o a la siguiente neurona, el resultado obtenido por dicha neurona.

$$y = F(b + \sum_i w_i \cdot n_i) \quad (24)$$

Como se ha mencionado anteriormente, una red neuronal artificial es simplemente una combinación de muchas unidades neuronales formando distintas capas (Figura 2.8). La información va fluyendo a través de la red hasta llegar a la salida final, donde se ha de comparar con la información real. Esto se consigue con la función de pérdida o de coste, que nos indicará lo mucho o poco que ha aprendido. De esta forma, los pesos se optimizarán para minimizar esta función permitiendo que la próxima vez los resultados sean mucho mejores.

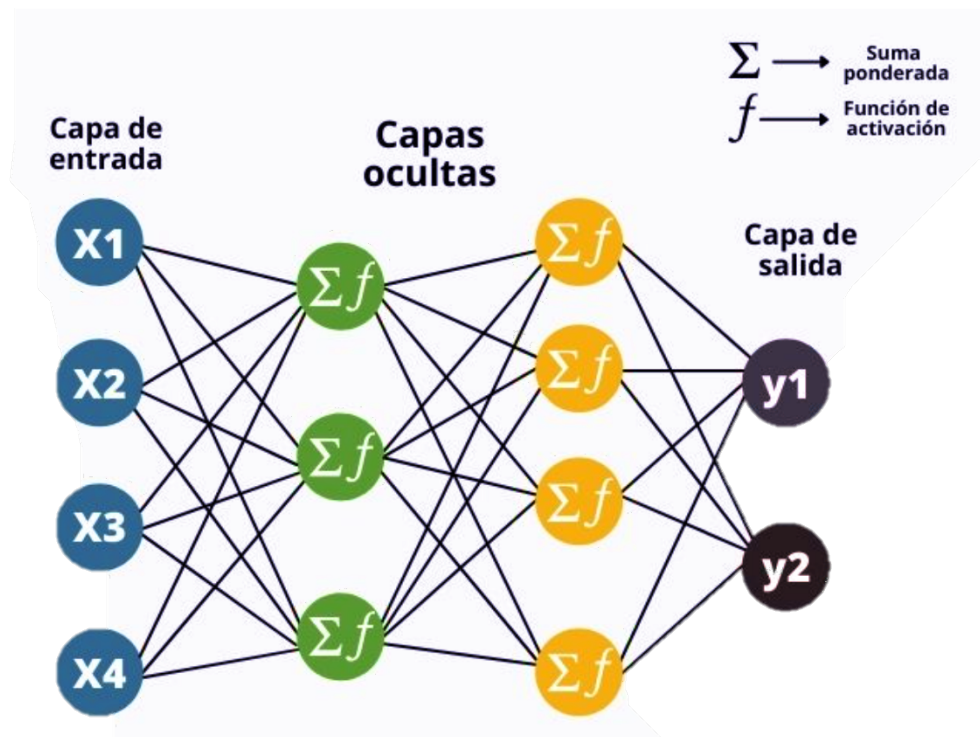


Figura 2.8: Red neuronal artificial [13].

## 2.6.2. ÁRBOLES DE DECISIÓN (RANDOM FOREST):

Random forest (o bosque aleatorio de decisión en español) es una técnica de Machine Learning muy popular en el manejo de datos, ya que son muchas las ventajas que presenta en comparación con otros algoritmos de datos. Es una técnica fácil de interpretar, estable, se puede utilizar en tareas de regresión o de



clasificación y que por lo general se obtienen buenos resultados, es decir, cubre gran parte de los problemas que se intenta solventar con el Machine Learning.

Como bien su nombre indica, en random forest primero encontramos la palabra “forest” (bosque, en español), entendiendo que este algoritmo se basa en árboles que se suelen llamar árboles de decisión. Se trata de una técnica de aprendizaje automático supervisado cuya ventaja principal es que obtiene un mejor rendimiento de generalización para un rendimiento durante entrenamiento similar. Esta mejora en la generalización la consigue compensando los errores de las predicciones de los distintos árboles de decisión [14].

Un árbol de decisión (Figura 2.9) consta de tres componentes: nodos de decisión, nodos de hoja y nodo raíz. El algoritmo de árbol divide un conjunto de datos de entrenamiento en ramas, que a su vez se pueden dividir en otras ramas hasta llegar a un nodo de hoja en el cual, no hay más divisiones. Los nodos representarán características o atributos, cada rama una decisión o regla, y cada hoja un resultado.

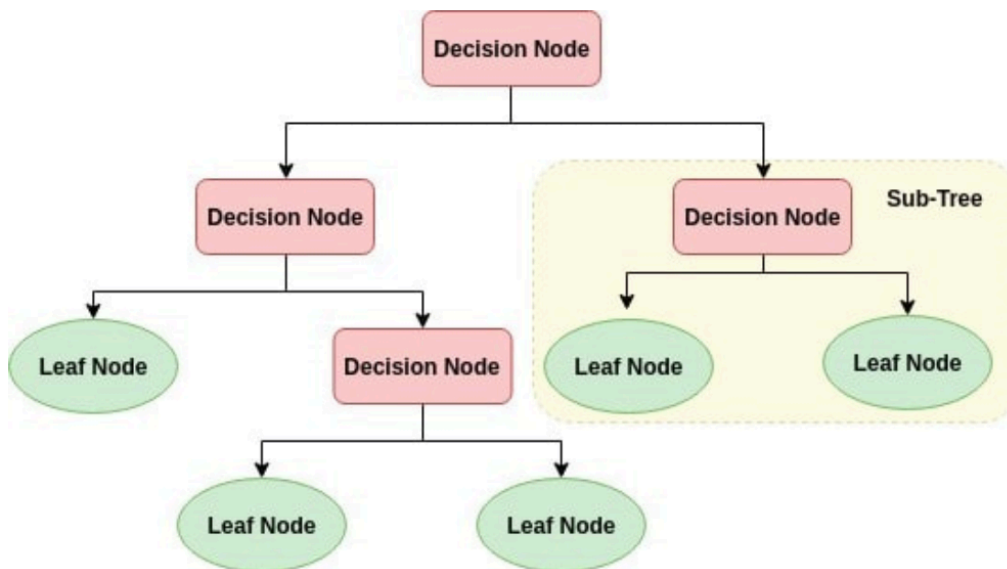


Figura 2.9. Árbol de decisión [15]

El entrenamiento de estos árboles se hace generando repetitivamente particiones sobre el espacio de características, obteniendo agrupaciones de datos cada vez más uniformes. Para clasificar un nuevo dato, o bien realizar la regresión, simplemente se verifican las condiciones para ese dato en cada nodo del árbol hasta que sea asignado a una hoja en particular.



La desventaja de los árboles de decisión es el *overfitting*, es decir, funcionan muy bien durante su entrenamiento, pero no tanto a la hora de hacer las predicciones (cuando se introducen datos nuevos). Es por ello que se hace uso de los Bosques Aleatorios (Random Forests).

Los bosques aleatorios entrenan varios árboles, cada uno con una parte distinta del conjunto de datos original y seleccionados aleatoriamente. Una vez el bosque aleatorio está entrenado, se puede hacer la predicción para un dato nuevo, para ello se deben agregar los resultados de cada árbol que lo forma. Es decir, si lo que se lleva a cabo es una clasificación se asignará al dato la categoría elegida por la mayoría de los árboles, mientras que si se lleva a cabo una regresión se le asignará al dato el promedio de las predicciones por cada árbol.

Gracias a esta aleatoriedad cada árbol será entrenado con un conjunto de datos diferente, por tanto, si este conjunto tuviera ruido afectará probablemente sólo a una parte de los árboles. Además, con la agregación los resultados para la generación de la predicción, los árboles que no trabajan de manera adecuada tendrán un menor impacto en el resultado final.

Llegados a este punto, es importante mencionar los dos algoritmos esenciales en el funcionamiento de los Bosques aleatorios: el *Bootstrapping* durante el entrenamiento, y la agregación de los resultados para obtener la predicción. La combinación de ambos es conocida como *bagging*.

*Bootstrapping* [16] es una técnica de remuestreo estadístico que supone la extracción de muestras aleatorias de un conjunto de datos con reemplazo. Esto permite generar nuevas muestras de un conjunto de datos sin tener que buscar nuevos datos de entrenamiento. La idea es muestrear repetitivamente los datos con el reemplazo del conjunto de entrenamiento original para producir múltiples conjuntos de entrenamiento separados. Con ello, se reduce la varianza de las predicciones, mejorando así en buena parte el rendimiento predictivo.

Gracias a esto, se consigue que cada árbol del que se compone el bosque aleatorio tenga una escasa correlación con los demás que lo forman. Los errores que se pueden dar en cada árbol son individuales, de tal forma que no afectarán a los otros del bosque. Además, cuenta con la ventaja del buen manejo de grandes números de variables y datos gracias a la aleatoriedad y el poco efecto del ruido en la predicción. Es por ello, que este método se posiciona como uno de los más eficientes en la clasificación de datos. [17]



### 2.6.3. MATRIZ DE CONFUSIÓN:

Con lo visto hasta ahora podemos corroborar que existen diferentes modelos de aprendizaje automático supervisado, surgiendo a todo esto la cuestión de poder verificar que los resultados obtenidos se asemejan a los deseados, comprobando así la calidad del modelo, y también poder determinar cuál de todos estos métodos es el mejor. Si el número de variables de salida fuera bajo, se podría hacer la comprobación fácilmente mirando directamente los resultados, sin embargo, estamos tratando con números altos de variables de salida, lo que conlleva a buscar herramientas que permitan visualizar estos resultados de manera más sencilla.

Una de estas herramientas es la matriz de confusión, también conocida como matriz de error. Se trata de una tabla que compara los valores reales de los datos de prueba con los valores que se han predicho por el modelo usado, mostrando qué cantidad de datos han sido clasificados correctamente y qué cantidad de datos han sido clasificados incorrectamente.

En la Figura 2.10 se puede observar un ejemplo de este tipo de gráficas, donde las filas corresponden a los valores predichos y las columnas corresponden a los valores reales o deseados. Las celdas diagonales corresponden a las observaciones que se han clasificado correctamente, siendo entonces las de fuera las que se han clasificado de manera errónea. En cada celda se observa tanto el número de observaciones como porcentaje del número total.

En el extremo de la derecha de la matriz a base de columna se muestran los porcentajes de todas las predicciones del algoritmo que pertenecen a cada clase, mostrándose los porcentajes correctos e incorrectos. Recibe el nombre de precisión el valor predictivo positivo, y tasa de descubrimiento falso el porcentaje incorrecto. La fila inferior de la matriz representa la misma métrica anterior, pero con los valores de salida deseados. Recibiendo el nombre de tasa de verdadero positivo las clasificadas correctamente y tasa de falso negativo las incorrectas. La celda situada en la parte inferior derecha de la celda muestra la precisión general. Este último valor es el más importante de la matriz, ya que será el que nos dará una mayor visión en la toma de decisiones.

| Output Class \ Target Class | 0           | 1           | 2           | 3           | 4           | 5            | 6           | 7           | 8           | 9           | Accuracy | Loss |
|-----------------------------|-------------|-------------|-------------|-------------|-------------|--------------|-------------|-------------|-------------|-------------|----------|------|
| 0                           | 480<br>9.6% | 0<br>0.0%   | 0<br>0.0%   | 0<br>0.0%   | 0<br>0.0%   | 0<br>0.0%    | 2<br>0.0%   | 0<br>0.0%   | 1<br>0.0%   | 0<br>0.0%   | 99.4%    | 0.6% |
| 1                           | 0<br>0.0%   | 470<br>9.4% | 2<br>0.0%   | 0<br>0.0%   | 4<br>0.1%   | 0<br>0.0%    | 6<br>0.1%   | 1<br>0.0%   | 0<br>0.0%   | 11<br>0.2%  | 95.1%    | 4.9% |
| 2                           | 8<br>0.2%   | 18<br>0.4%  | 482<br>9.6% | 5<br>0.1%   | 0<br>0.0%   | 0<br>0.0%    | 3<br>0.1%   | 5<br>0.1%   | 7<br>0.1%   | 1<br>0.0%   | 91.1%    | 8.9% |
| 3                           | 0<br>0.0%   | 1<br>0.0%   | 1<br>0.0%   | 470<br>9.4% | 0<br>0.0%   | 0<br>0.0%    | 1<br>0.0%   | 0<br>0.0%   | 7<br>0.1%   | 4<br>0.1%   | 97.1%    | 2.9% |
| 4                           | 0<br>0.0%   | 1<br>0.0%   | 0<br>0.0%   | 0<br>0.0%   | 492<br>9.8% | 0<br>0.0%    | 1<br>0.0%   | 0<br>0.0%   | 0<br>0.0%   | 2<br>0.0%   | 99.2%    | 0.8% |
| 5                           | 1<br>0.0%   | 1<br>0.0%   | 0<br>0.0%   | 17<br>0.3%  | 0<br>0.0%   | 498<br>10.0% | 9<br>0.2%   | 1<br>0.0%   | 5<br>0.1%   | 1<br>0.0%   | 93.4%    | 6.6% |
| 6                           | 3<br>0.1%   | 3<br>0.1%   | 4<br>0.1%   | 0<br>0.0%   | 0<br>0.0%   | 1<br>0.0%    | 463<br>9.3% | 0<br>0.0%   | 3<br>0.1%   | 1<br>0.0%   | 96.9%    | 3.1% |
| 7                           | 0<br>0.0%   | 6<br>0.1%   | 1<br>0.0%   | 0<br>0.0%   | 0<br>0.0%   | 0<br>0.0%    | 1<br>0.0%   | 490<br>9.8% | 0<br>0.0%   | 1<br>0.0%   | 98.2%    | 1.8% |
| 8                           | 0<br>0.0%   | 0<br>0.0%   | 2<br>0.0%   | 4<br>0.1%   | 2<br>0.0%   | 1<br>0.0%    | 2<br>0.0%   | 0<br>0.0%   | 473<br>9.5% | 2<br>0.0%   | 97.3%    | 2.7% |
| 9                           | 8<br>0.2%   | 0<br>0.0%   | 8<br>0.2%   | 4<br>0.1%   | 2<br>0.0%   | 0<br>0.0%    | 12<br>0.2%  | 3<br>0.1%   | 4<br>0.1%   | 477<br>9.5% | 92.1%    | 7.9% |
| Overall                     | 96.0%       | 94.0%       | 96.4%       | 94.0%       | 98.4%       | 99.6%        | 92.6%       | 98.0%       | 94.6%       | 95.4%       | 95.9%    | 4.1% |

Figura 2.10. Ejemplo de matriz de confusión con un 95.9% de predicciones correctas

## 2.7. METODO VISUAL DE DIAGNOSIS DE FALLOS: FDA-t-SNE-NN:

También se estudiará un modelo de diagnóstico de fallos que combina alguna de las técnicas vistas hasta ahora, como es el Análisis Discriminante de Fisher (FDA), la incrustación de vecinos estocásticos distribuidos en t (t-SNE), y redes neuronales artificiales [18].

En primer lugar, se realiza un modelado fuera de línea, donde se usa la técnica FDA para extraer información del conjunto de datos de entrada. Teniendo en cuenta que la entrada es el conjunto de datos de entrenamiento  $X$ , se obtiene un nuevo vector de características  $Z$ .

$$X = [X_0, X_1, \dots, X_{c-1}] \quad (25)$$

$$Z = W_{FDA}^T X = [Z_0, Z_1, \dots, Z_{c-1}] \quad (26)$$

Se aplica la técnica t-SNE, con la que se consigue obtener los puntos que contienen la información extraída de los datos de entrada,  $Y$ , y que serán los puntos para realizar la representación gráfica que permite clasificar la anomalía de la planta.



$$Y = [Y_0, Y_1, \dots, Y_{c-1}] \quad (27)$$

Para obtener la relación que tiene el conjunto de entrada X, con el conjunto de mapeo Y, se lleva a cabo el entrenamiento de una red neuronal BP que interioriza esa relación.

Posteriormente, se lleva a cabo el monitoreo en línea, donde una muestra de datos de la planta nueva, es llevada a la red neural BP que se ha entrenado anteriormente, obteniendo así, los nuevos puntos de mapeo para la nueva muestra correspondiente. Estos nuevos puntos de mapeo representan el estado de funcionamiento en tiempo real de la planta de estudio.



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS INDUSTRIALES

Universidad de Valladolid



# CAPITULO III: PLANTA TENNESSEE-EASTMAN



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS INDUSTRIALES

Universidad de Valladolid





### 3.1. DESCRIPCIÓN DE LA PLANTA:

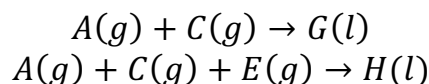
Los datos que se han utilizado para la realización de este trabajo son extraídos de la planta Tennessee Eastmann (TEP). Planta cuyo objetivo es servir como banco de pruebas para científicos o ingenieros dedicados al control estadístico de procesos.

En 1993, la empresa estadounidense Eastman Chemical Company y la universidad de Tennessee cooperaron para sacar adelante un proyecto basado en la creación de una planta que sirviese como modelo experimental a partir de una planta real de la propia compañía química. El modelo de planta, que fue publicado por Downs y Vogel, quedó conocido como planta Tennessee Eastman (TEP). El propósito de estos fue plantear un problema que sirviera como referencia para desarrollar y evaluar diferentes tecnologías de control de procesos, destinado principalmente, para estudios relacionados con el control total de la planta y control multivariable. La planta se caracteriza por su alta no linealidad y complejidad, y por las numerosas variables distintas de las que está formada, lo que hace que tenga la capacidad de verse en una multitud de escenarios distintos, permitiendo estudiar y analizar distintos métodos de detección y diagnóstico de fallos [10].

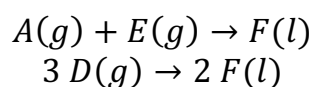
En la Figura 3.1, podemos observar un diagrama simplificado de la planta TE. El proceso consiste en cinco operaciones unitarias: una de reacción que se lleva a cabo en un reactor, una de condensación, una de separación de líquido-vapor, una de compresión de reciclo y una operación de desorción, que se lleva a cabo en una columna stripper. Esta última operación también es conocida como stripping.

El reactor es un tanque de agitado continuo (CSTR) de dos fases, en el cual se generan los productos G y H y un subproducto F a partir de cuatro reactivos A, C, D y E y un inerte B. Las reacciones que se llevan a cabo en el reactor para obtener los productos mencionados son las siguientes:

- Reacciones de formación de productos:



- Reacciones de formación del subproducto F:



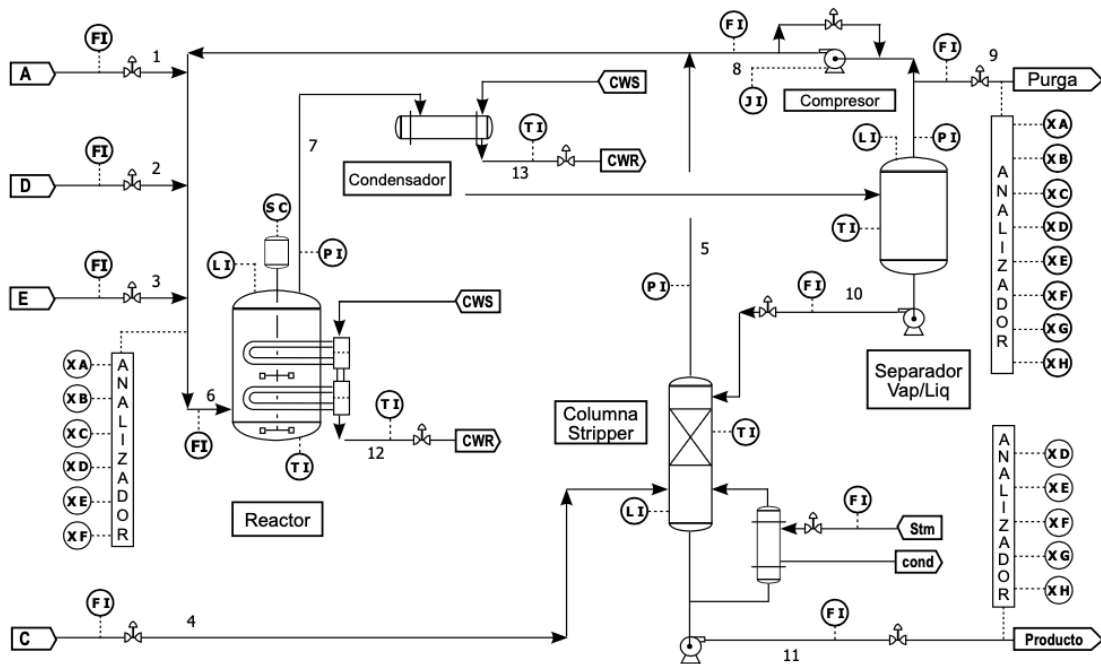


Figura 3.1: Diagrama de proceso de la planta TE [19].

Todas las reacciones se caracterizan por ser reversibles, exotérmicas, por tener una relación lineal entre concentración y tiempo, y por su velocidad dependiente de la temperatura. La reacción por la que obtenemos el producto G tiene una energía de activación mayor, lo que quiere decir que tiene una mayor sensibilidad a la temperatura.

El proceso además consta de 12 válvulas para manipulación y 41 mediciones para realizar el seguimiento y control, de las cuales 22 son continuas y las otras 19 restantes discretas.

El proceso se inicia con la incorporación de los reactivos gaseosos al reactor, donde van a reaccionar hasta obtener los productos de fase líquida. Para aumentar la velocidad de reacción de esta fase gaseosa, se va a añadir un catalizador no volátil disuelto en fase líquida, y que en ningún momento abandonará el reactor. Además, mediante un sistema de refrigeración interno se eliminará el calor que se genera en la reacción.

Del reactor salen los productos en forma de vapor junto con los reactivos que no han reaccionado, haciéndose pasar por un condensador para enfriar la fase gas y transformarla en líquida, hasta llegar a un separador de líquido y vapor. Los componentes no condensados se reciclan de nuevo para aumentar así la eficacia del proceso. Para ello serán introducidos en un compresor centrífugo alimentando



así de nuevo al reactor. Los componentes condensados serán dirigidos a una columna de extracción de productos, que junto con la alimentación del reactivo C, se realizará una operación de desorción eliminando así los reactivos sobrantes.

Finalmente, los productos G y H saldrán de la base de la columna y se separan en una sección de refinación de aguas abajo que no está incluida en el problema al no tener importancia en el estudio. El inerte y el subproducto F serán purgados del sistema principalmente como vapor al hacerse pasar por el separador líquido-vapor.

### 3.2. DATOS DE LA PLANTA:

A continuación, en la Tablas 3.1, 3.2 y 3.3 se muestran los datos correspondientes a la planta Tennessee Eastmann [19]. Teniendo en cuenta el número de válvulas de los que dispone la planta, el modelo cuenta con 12 variables manipuladas (XMV), representadas en la Tabla 3.1, y 41 variables medidas (XMEAS). De estas últimas, las 22 primeras variables se miden de manera continua, y se representan en la Tabla 3.2, mientras que las 19 restantes son variables discretas que proceden de los distintos analizadores que realizan mediciones periódicamente, y que serán representadas en la Tabla 3.3. También se dispone de datos correspondientes a las distintas anomalías que pueden suceder, 21 datos que son presentados en la Tabla 3.4.

| Variable                                    | Número de la variable | Unidades                       |
|---|-----------------------|--------------------------------|
| Flujo de la variable D                      | XMV(1)                | Kg h <sup>-1</sup>             |
| Flujo de alimentación E                     | XMV(2)                | Kg h <sup>-1</sup>             |
| Flujo de alimentación A                     | XMV(3)                | Kscm h                         |
| Flujo de alimentación A y C                 | XMV(4)                | Kscm h                         |
| Válvula de recirculación del compresor      | XMV(5)                | %                              |
| Válvula de purga                            | XMV(6)                | %                              |
| Flujo de líquido del separador LV           | XMV(7)                | m <sup>3</sup> h <sup>-1</sup> |
| Flujo de líquido de la columna de stripping | XMV(8)                | m <sup>3</sup> h <sup>-1</sup> |



# DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



Universidad de Valladolid

ESCUELA DE INGENIERÍAS INDUSTRIALES

|   |         |                                |
|---|---------|--------------------------------|
| Válvula de vapor de la columna de stripping | XMV(9)  | %                              |
| Flujo de agua de refrigeración del reactor  | XMV(10) | m <sup>3</sup> h <sup>-1</sup> |
| Flujo de agua en el condensador             | XMV(11) | m <sup>3</sup> h <sup>-1</sup> |
| Velocidad del agitador del reactor          | XMV(12) | rpm                            |

Tabla 3.1: Variables manipuladas del proceso

| Variables                             | Número de la variable | Unidades                       |
|---------------------------------------|-----------------------|--------------------------------|
| Flujo de alimentación A               | XMEAS(1)              | kscmh                          |
| Flujo de alimentación D               | XMEAS(2)              | Kscmh                          |
| Flujo de alimentación E               | XMEAS(3)              | Kscmh                          |
| Flujo de alimentación A y C           | XMEAS(4)              | Kscmh                          |
| Flujo de recirculación                | XMEAS(5)              | Kscmh                          |
| Flujo alimentación reactor            | XMEAS(6)              | Kscmh                          |
| Presión del reactor                   | XMEAS(7)              | KPa                            |
| Nivel del reactor                     | XMEAS(8)              | %                              |
| Temperatura del reactor               | XMEAS(9)              | °C                             |
| Flujo de purga                        | XMEAS(10)             | Kscmh                          |
| Temperatura del separador de producto | XMEAS(11)             | °C                             |
| Nivel del separador de producto       | XMEAS(12)             | %                              |
| Presión del separador de producto     | XMEAS(13)             | KPa                            |
| Corriente del separador de producto   | XMEAS(14)             | m <sup>3</sup> h <sup>-1</sup> |
| Nivel del stripper                    | XMEAS(15)             | %                              |
| Presión del stripper                  | XMEAS(16)             | KPa                            |
| Corriente del stripper                | XMEAS(17)             | m <sup>3</sup> h <sup>-1</sup> |
| Temperatura del stripper              | XMEAS(18)             | °C                             |
| Flujo de vapor del stripper           | XMEAS(19)             | Kg h <sup>-1</sup>             |
| Potencia del compresor                | XMEAS(20)             | KW                             |



|  |           |    |
|--|-----------|----|
| Temperatura de la salida del agua de refrigeración del reactor   | XMEAS(21) | °C |
| Temperatura de la salida del agua de refrigeración del separador | XMEAS(22) | °C |

Tabla 3.2: Variables medidas del proceso

| Muestreo de la alimentación del reactor |                       |          |
|---|-----------------------|----------|
| Variable                                | Número de la variable | Unidades |
| A                                       | XMEAS(23)             | %mol     |
| B                                       | XMEAS(24)             | %mol     |
| C                                       | XMEAS(25)             | %mol     |
| D                                       | XMEAS(26)             | %mol     |
| E                                       | XMEAS(27)             | %mol     |
| F                                       | XMEAS(28)             | %mol     |
| Muestreo del gas de purga               |                       |          |
| Variable                                | Número de la variable | Unidades |
| A                                       | XMEAS(29)             | %mol     |
| B                                       | XMEAS(30)             | %mol     |
| C                                       | XMEAS(31)             | %mol     |
| D                                       | XMEAS(32)             | %mol     |
| E                                       | XMEAS(33)             | %mol     |
| F                                       | XMEAS(34)             | %mol     |
| G                                       | XMEAS(35)             | %mol     |
| H                                       | XMEAS(36)             | %mol     |
| Muestreo del producto final             |                       |          |
| Variable                                | Número de la variable | Unidades |
| D                                       | XMEAS(37)             | %mol     |
| E                                       | XMEAS(38)             | %mol     |
| F                                       | XMEAS(39)             | %mol     |
| G                                       | XMEAS(40)             | %mol     |
| H                                       | XMEAS(41)             | %mol     |
| F                                       | XMEAS(42)             | %mol     |

Tabla 3.3: Variables medidas muestreadas



# DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



Universidad de Valladolid

ESCUELA DE INGENIERÍAS INDUSTRIALES

| Variable  | Número de fallo | Tipo                                 |
|---|-----------------|--------------------------------------|
| Relación de flujo de alimentaciones A/C, composición de B constante | IDV(1)          | Salto                                |
| Composición de B, con relación A/C constante                        | IDV(2)          | Salto                                |
| Temperatura de la alimentación de D                                 | IDV(3)          | Salto                                |
| Temperatura de entrada del agua de refrigeración del reactor        | IDV(4)          | Salto                                |
| Temperatura de entrada del agua de condensación                     | IDV(5)          | Salto                                |
| Pérdida en la alimentación de A                                     | IDV(6)          | Salto                                |
| Pérdida de presión en la alimentación de C                          | IDV(7)          | Salto                                |
| Composición de A, B, C  | IDV(8)          | Variación aleatoria                  |
| Temperatura de la alimentación de D                                 | IDV(9)          | Variación aleatoria                  |
| Temperatura de la alimentación de C                                 | IDV(10)         | Variación aleatoria                  |
| Temperatura de entrada del agua de refrigeración del reactor        | IDV(11)         | Variación aleatoria                  |
| Temperatura de entrada del agua de condensación                     | IDV(12)         | Variación aleatoria                  |
| Cinética de reacción  | IDV(13)         | Variación lenta                      |
| Apertura de la válvula del agua de refrigeración del reactor        | IDV(14)         | Sin variación, constante en un valor |
| Apertura de la válvula del agua de condensación                     | IDV(15)         | Sin variación, constante en un valor |
| Desconocido   | IDV(16)         | Desconocido                          |
| Desconocido   | IDV(17)         | Desconocido                          |
| Desconocido   | IDV(18)         | Desconocido                          |
| Desconocido   | IDV(19)         | Desconocido                          |
| Desconocido   | IDV(20)         | Desconocido                          |
| Desconocido   | IDV(21)         | Desconocido                          |

Tabla 3.4: Fallos posibles del proceso

Para la realización de este trabajo se han utilizado diferentes conjuntos de datos. El primero corresponde a los datos de simulación de la planta frente a un comportamiento normal de la misma, es decir, sin fallo alguno. Y los demás corresponden a las diferentes anomalías que pueden darse en la planta, resultando un total de 21 conjuntos de datos diferentes. Cabe señalar que el comportamiento de los fallos 3,9 y 15 se asemeja bastante al comportamiento en condiciones normales de funcionamiento, haciendo que sean fallos difíciles de detectar.



# CAPITULO IV: APLICACIÓN DESARROLLADA



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS INDUSTRIALES

Universidad de Valladolid





#### 4.1. INTRODUCCIÓN A LA APLICACIÓN DESARROLLADA:

Para el desarrollo de este proyecto, como ya se ha mencionado con anterioridad, se van a utilizar los datos de simulación del proceso Tennessee Eastman, y a aplicar las técnicas explicadas en el Capítulo II. Se comenzará haciendo una detección de fallos y seguidamente se realizará un análisis y clasificación de los mismos.

Se van a utilizar dos conjuntos de datos, cada uno de ellos con 22 simulaciones diferentes. El primer conjunto cuenta con una primera simulación que corresponde al comportamiento normal de la planta, y que tendrá 500 observaciones de cada una de las 52 variables, y 21 simulaciones restantes correspondientes a cada una de las 21 anomalías diferentes que pueden existir, con 479 observaciones de las variables. El segundo conjunto de datos que se va a usar cuenta, al igual que el conjunto anterior, con 22 simulaciones, esta vez cada una con 960 observaciones de las 52 variables. La gran diferencia con el primer bloque, a parte del número de observaciones, es que las primeras 160 observaciones siempre serán de comportamiento normal, es decir, del comportamiento de la planta antes de que ocurra el fallo.

Tendremos en cuenta también que los fallos 3,9 y 15 son anomalías difíciles de detectar y clasificar por su gran similitud con el comportamiento normal de la planta, luego no se hará uso de los mismos en la realización de los experimentos.

Se trabajará con el lenguaje de programación Python, realizando en primer lugar un Análisis de los Componentes Principales para la posterior detección de los fallos, y más adelante, se pasará a evaluar distintas técnicas de clasificación basadas en Machine Learning, para el diagnóstico de fallos.

#### 4.2. DETECCIÓN DE FALLOS APLICANDO ANÁLISIS DE COMPONENTES PRINCIPALES:

El primer paso para la detección de fallos aplicando PCA será analizar la planta fuera de línea, es decir, con los datos relativos al comportamiento normal de la planta. Se hará una reducción de la dimensionalidad y se calcularán los estadísticos  $T^2$  y  $Q$  con sus respectivos umbrales para la posterior detección de fallos.

A continuación, se analizará la planta en línea con los datos de fallo obteniendo los dos estadísticos mencionados y detectando así las correspondientes anomalías.



#### 4.2.1. ANÁLISIS FUERA DE LÍNEA:

El procedimiento se empieza cargando los datos correspondientes al comportamiento normal y almacenándolos en una matriz  $X$  de dimensión  $960 \times 52$ . Esto quiere decir que partimos de un espacio muestral de datos de  $\mathcal{R}^{960 \times 52}$  que pasaremos a reducir a un espacio de  $\mathcal{R}^{960 \times a}$ , siendo  $a$  el número de componentes principales que obtendremos más adelante. La matriz será normalizada a media cero y varianza uno, ya que se pretende que todos los datos tengan la misma importancia. Se calculará la matriz de correlación  $R$  de la matriz  $X$  normalizada (Ecuación 2) y se realizará su descomposición en valores singulares, obteniendo la matriz  $\Lambda$  que almacena los valores propios en su diagonal en orden decreciente, y la matriz  $V$  que almacena los vectores propios correspondientes a cada uno de los anteriores (Ecuación 3). A partir de esta matriz  $V$  y haciendo un test de porcentaje de varianza, podremos obtener el número de componentes principales con los que nos quedaremos. Trabajaremos con una variabilidad máxima del 90%, de manera que, aplicando lo relativo a la ecuación 4, llegamos a un resultado de 31 componentes principales con una varianza de 89,46%.

Una vez hemos definido el valor de  $a$ , podemos definir la matriz de carga  $P \in \mathcal{R}^{52 \times 31}$  formada por los 31 vectores de carga correspondientes a los valores singulares más grandes. Gracias a esta matriz podremos proyectar los datos de la matriz  $X$  normalizada a un espacio reducido, que conoceremos como  $T$  (Ecuación 5) de dimensión  $\mathcal{R}^{960 \times 31}$ .

Tras la obtención del espacio reducido, podremos recalculer el espacio de partida a partir de la matriz de carga  $P$  y el espacio  $T$  (Ecuación 6). Esto nos servirá para obtener la matriz de residuos  $E$ , ya que se trata de la diferencia entre el espacio inicial original y el espacio inicial recalculado (Ecuación 7).

Llegados a este punto, se procederá a calcular los estadísticos  $T^2$  y  $Q$  que nos permiten monitorizar el proceso. En primer lugar, se calculará el estadístico  $T^2$  que a partir de cada vector de observación de dimensión  $1 \times 52$ , la ecuación 9 y teniendo en cuenta que contamos con 960 observaciones, se llega a una matriz de dimensiones  $1 \times 960$ , es decir, un valor estadístico  $T^2$  para cada una de las observaciones. Para el cálculo del umbral será necesario elegir un nivel de significancia  $\alpha$ , elegido en este caso un valor de 0,01. Mediante la ecuación 10 se procede a calcular el umbral estadístico de  $T^2$ , que nos permitirá más adelante detectar las anomalías para las distintas simulaciones con los datos de fallo. El umbral que se ha obtenido en este caso toma un valor de 50.64.

Si representamos cada uno de los valores del estadístico  $T^2$  calculado y el umbral correspondiente (Figura 4.1), podemos observar que no se supera ese umbral

calculado salvo en alguna ocasión puntual que consideraremos despreciable. Luego se puede intuir que lo realizado es correcto.

Pasamos ahora a calcular el estadístico Q, que como se observa en la ecuación 11 será necesario definir un vector de residuos  $r$  obtenido a partir de la ecuación 12, que al igual que en caso anterior, corresponde a un vector por cada una de las observaciones. Será necesario también el cálculo de su umbral (Ecuación 13), que tomará en este caso un valor de 11.18.

Al igual que con el estadístico  $T^2$ , podremos representar sus valores y comprobar que el funcionamiento del proceso es correcto (Figura 4.2).

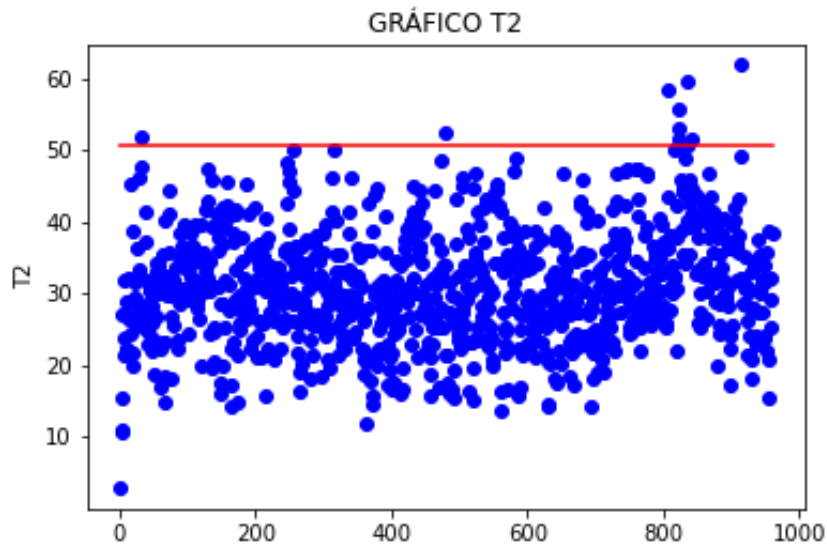


Figura 4.1: Estadístico  $T^2$  para los datos de funcionamiento normal de la planta

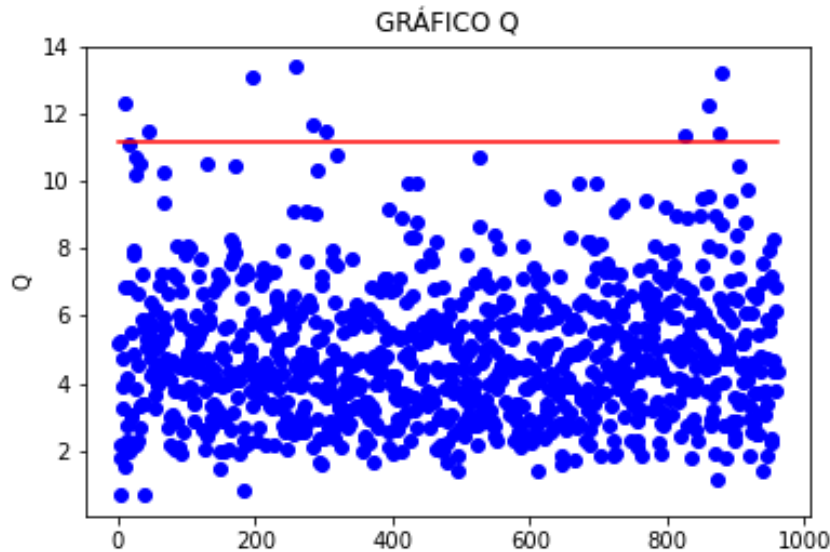


Figura 4.2: Estadístico Q para los datos de funcionamiento normal de la planta

#### 4.2.2. ANÁLISIS EN LÍNEA. DETECCIÓN DE ANOMALÍAS:

Una vez obtenido el modelo PCA y los umbrales de los estadísticos del funcionamiento normal de la planta, se puede pasar a la parte de detección de fallos. Para ello se usarán las 21 simulaciones que contienen los datos relativos a cada tipo de fallo que puede ocurrir, constanding cada una de 960 observaciones de las 52 variables. Al igual que en procedimiento anterior, la matriz X de datos con dimensión 960x52 será normalizada con la media y varianza de la matriz de datos original de comportamiento normal del apartado anterior.

Una vez la matriz con los datos está normalizada, se pasa a calcular sus estadísticos  $T^2$  y Q, y a compararlos con los umbrales propios del comportamiento normal  $T^2_a$  y  $Q_a$ . Para obtenerlos se procede a aplicar las mismas ecuaciones que en su cálculo fuera de línea (ecuación 9 y ecuación 11) con la matriz de fallo normalizada correspondiente, obteniendo los estadísticos correspondientes para las distintas observaciones.

Finalmente, se representarán los resultados obtenidos y se procederá a analizarlos. Puede ocurrir que el estadístico supere el umbral de manera puntual sin tener relación con fallo alguno, es decir, que ocurra una falsa alarma. Para evitar esto, vamos a considerar que el fallo es detectado cuando ocurran al menos seis alarmas seguidas. Además, se van a calcular una serie de parámetros que nos facilitarán el análisis de la detección. Estos parámetros serán: el tiempo de detección, que nos indica el número de observación en la que el fallo es detectado, es decir, cuando

se detecten 6 alarmas seguidas, el porcentaje de alarmas detectadas, que indica el porcentaje de datos que superan el umbral estadístico una vez ha ocurrido el fallo, y el porcentaje de falsas alarmas, que indica el porcentaje de datos que supera el umbral antes de que se produzca el fallo.

A continuación, se mostrarán los gráficos de las simulaciones para fallo 1, 21 y 9, con los parámetros anteriores mencionados. En la Figura 4.3 están representados los gráficos estadísticos  $T^2$  y  $Q$  para el fallo 1 “Relación de flujo de alimentaciones A/C, composición de B constante”, observando que a partir de un punto los datos superan el umbral claramente, pudiendo comprobar la existencia de ese fallo.

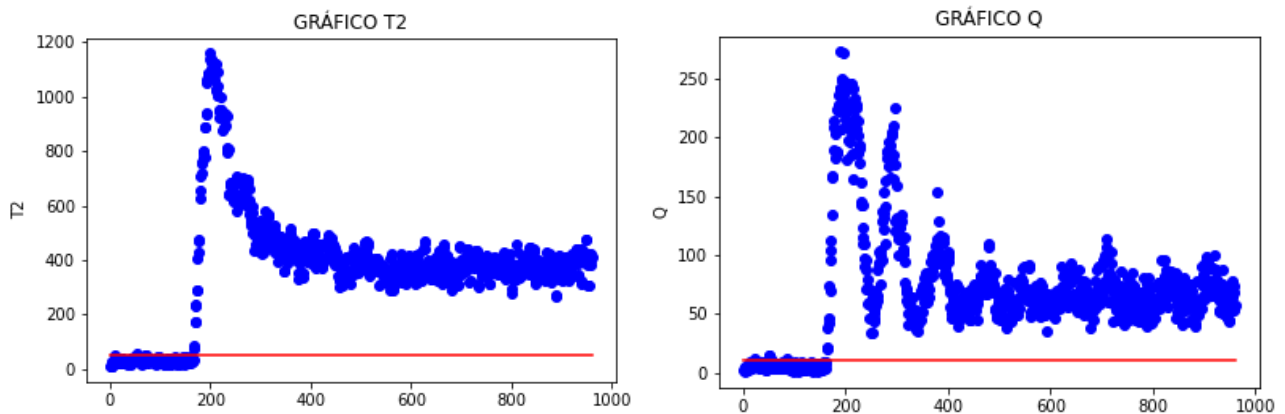


Figura 4.3: Gráficos estadísticos  $T^2$  y  $Q$  para el fallo 1

Tanto para el estadístico  $T^2$  como para el  $Q$ , los fallos son detectados cerca de la observación 160, observación a partir de la cual ocurre el fallo, esto quiere decir que el método detecta el fallo producido rápidamente. Esto podemos verlo tanto en los gráficos como en la tabla siguiente (Tabla 4.1), donde también se indica el índice de alarmas detectadas, que es considerablemente alto, lo que quiere decir que han sido detectadas correctamente, y el índice de falsas alarmas, de bajo valor en este caso, luego son minoría las alarmas que puntualmente han podido sobrepasar el umbral correspondiente.

|       | TIEMPO DE DETECCIÓN | ALARMAS DETECTADAS | ÍNDICE DE ALARMAS DETECTADAS(%) | FALSAS ALARMAS | ÍNDICE DE FALSAS ALARMAS(%) |
|-------|---------------------|--------------------|---------------------------------|----------------|-----------------------------|
| $T^2$ | 167                 | 794                | 99.25                           | 2              | 1.25                        |
| $Q$   | 163                 | 798                | 99.75                           | 6              | 3.75                        |

Tabla 4.1: Parámetros calculados para fallo 1

Igualmente, en la Figura 4.4 se representan los gráficos correspondientes al fallo 21 de origen desconocido, donde podemos comprobar que al igual que en caso anterior los estadísticos permiten detectar el fallo correctamente.

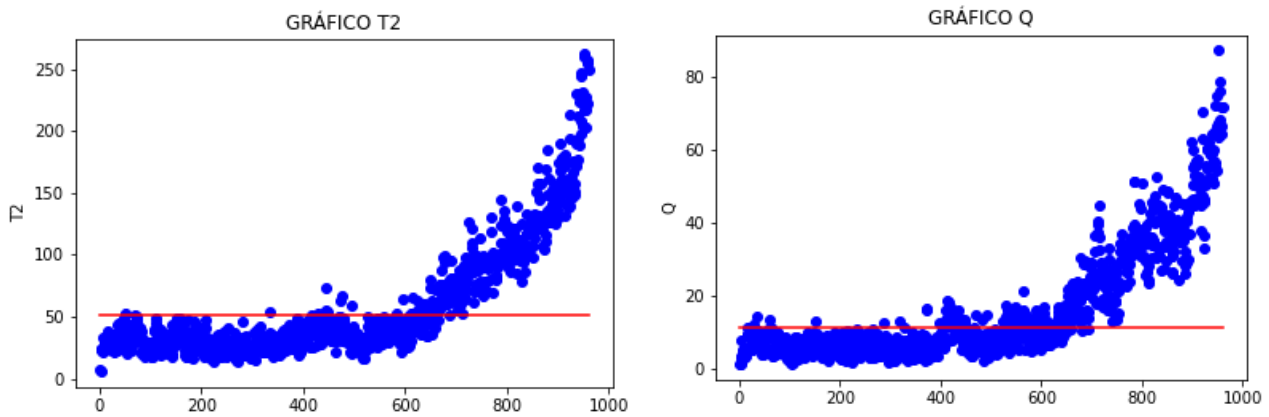


Figura 4.4: Gráficos estadísticos  $T^2$  y  $Q$  para el fallo 21

Esta vez podemos observar que el fallo no es detectado hasta la observación 331 en el caso del estadístico  $T^2$ , y la observación 401 en el caso del estadístico  $Q$ . Luego este fallo tarda más en ser detectado, como se muestra en la tabla 4.2.

|       | TIEMPO DE DETECCIÓN | ALARMAS DETECTADAS | ÍNDICE DE ALARMAS DETECTADAS(%) | FALSAS ALARMAS | ÍNDICE DE FALSAS ALARMAS(%) |
|-------|---------------------|--------------------|---------------------------------|----------------|-----------------------------|
| $T^2$ | 665                 | 331                | 41.375                          | 2              | 1.25                        |
| $Q$   | 410                 | 401                | 50.125                          | 6              | 3.75                        |

Tabla 4.2: Parámetros calculados para fallo 21

Por último, en la Figura 4.5 se representan los gráficos relativos al fallo 9 “Temperatura de la alimentación de D”. Este fallo, como ya hemos mencionado en anteriores ocasiones, es difícil de detectar, los gráficos son muy similares a los gráficos de comportamiento normal que se mostraban en las figuras 4.1 y 4.2.

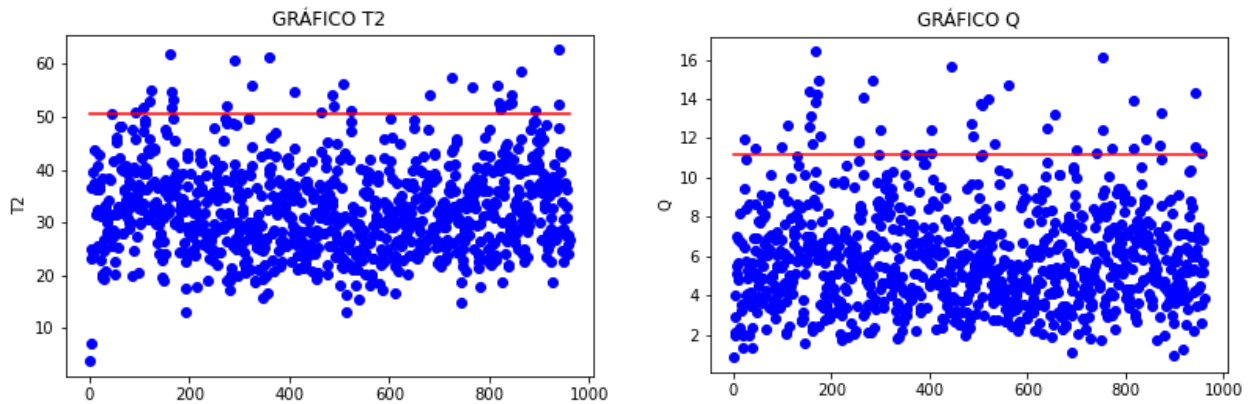


Figura 4.5: Gráficos estadísticos  $T^2$  y  $Q$  para el fallo 9

Este fallo no ha sido detectado por el método, lo que quiere decir que no se han conseguido encontrar al menos 6 alarmas seguidas por encima del umbral (Tabla 4.3).

|       | DETECTADO | TIEMPO DE DETECCIÓN | ALARMAS DETECTADAS | ÍNDICE DE ALARMAS DETECTADAS(%) | FALSAS ALARMAS | ÍNDICE DE FALSAS ALARMAS(%) |
|-------|-----------|---------------------|--------------------|---------------------------------|----------------|-----------------------------|
| $T^2$ | <b>NO</b> | -                   | <b>27</b>          | <b>3.375</b>                    | <b>5</b>       | <b>3.125</b>                |
| $Q$   | <b>NO</b> | -                   | <b>37</b>          | <b>4.625</b>                    | <b>7</b>       | <b>4.375</b>                |

Tabla 4.3: Parámetros calculados para fallo 9

A continuación, se muestra una tabla resumen con todos los resultados que se han obtenido analizando todas las simulaciones del proceso. La Tabla 4.4 muestra los resultados del estadístico  $T^2$  y la Tabla 4.5 los resultados del estadístico  $Q$ .

| FALLO    | DETECTADO | TIEMPO DE DETECCIÓN | ALARMAS DETECTADAS | ÍNDICE DE ALARMAS DETECTADAS(%) | FALSAS ALARMAS | ÍNDICE DE FALSAS ALARMAS(%) |
|----------|-----------|---------------------|--------------------|---------------------------------|----------------|-----------------------------|
| <b>1</b> | SÍ        | 167                 | 794                | 99.25                           | 2              | 1.25                        |
| <b>2</b> | SÍ        | 172                 | 789                | 98.625                          | 3              | 1.875                       |
| <b>3</b> | NO        | -                   | 15                 | 1.875                           | 2              | 1.25                        |
| <b>4</b> | SÍ        | 272                 | 335                | 41.875                          | 1              | 625                         |
| <b>5</b> | SÍ        | 171                 | 206                | 25.75                           | 1              | 625                         |



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



Universidad de Valladolid

ESCUELA DE INGENIERÍAS INDUSTRIALES

|    |    |     |     |              |   |       |
|----|----|-----|-----|--------------|---|-------|
| 6  | SÍ | 170 | 791 | 98.875       | 1 | 625   |
| 7  | SÍ | 161 | 800 | 100          | 0 | 0     |
| 8  | SÍ | 183 | 778 | 97.25        | 0 | 0     |
| 9  | NO | -   | 27  | 3.375        | 5 | 3.125 |
| 10 | SÍ | 264 | 246 | 30.75        | 3 | 1.875 |
| 11 | SÍ | 211 | 410 | 51.249999999 | 0 | 0     |
| 12 | SÍ | 167 | 790 | 98.75        | 3 | 1.875 |
| 13 | SÍ | 206 | 757 | 94.625       | 3 | 1.875 |
| 14 | SÍ | 161 | 796 | 99.5         | 1 | 625   |
| 15 | NO | -   | 20  | 2.5          | 2 | 1.25  |
| 16 | SÍ | 471 | 121 | 15.125       | 6 | 3.75  |
| 17 | SÍ | 189 | 629 | 78.625       | 1 | 625   |
| 18 | SÍ | 253 | 715 | 89.375       | 2 | 1.25  |
| 19 | NO | -   | 98  | 12.25        | 1 | 625   |
| 20 | SÍ | 247 | 250 | 31.25        | 4 | 2.5   |
| 21 | SÍ | 665 | 331 | 41.375       | 2 | 1.25  |

Tabla 4.4: Parámetros con el estadístico  $T^2$  calculados para todos los fallos

| FALLO | DETECTADO | TIEMPO DE DETECCIÓN | ALARMAS DETECTADAS | ÍNDICE DE ALARMAS DETECTADAS(%) | FALSAS ALARMAS | ÍNDICE DE FALSAS ALARMAS(%) |
|-------|-----------|---------------------|--------------------|---------------------------------|----------------|-----------------------------|
| 1     | SÍ        | 163                 | 798                | 99.75                           | 6              | 3.75                        |
| 2     | SÍ        | 175                 | 789                | 98.625                          | 7              | 4.375                       |
| 3     | NO        | -                   | 60                 | 7.5                             | 4              | 2.5                         |
| 4     | SÍ        | 161                 | 800                | 100                             | 4              | 2.5                         |
| 5     | SÍ        | 161                 | 271                | 33.875                          | 4              | 2.5                         |
| 6     | SÍ        | 161                 | 800                | 100                             | 4              | 2.5                         |
| 7     | SÍ        | 161                 | 800                | 100                             | 4              | 2.5                         |
| 8     | SÍ        | 178                 | 775                | 96.875                          | 7              | 4.375                       |
| 9     | NO        | -                   | 37                 | 4.625                           | 7              | 4.375                       |
| 10    | SÍ        | 208                 | 368                | 46                              | 0              | 0                           |
| 11    | SÍ        | 167                 | 555                | 69.375                          | 7              | 4.375                       |
| 12    | SÍ        | 183                 | 761                | 95.125                          | 7              | 4.375                       |



|    |    |     |     |        |   |       |
|----|----|-----|-----|--------|---|-------|
| 13 | SÍ | 201 | 761 | 95.125 | 2 | 1.25  |
| 14 | SÍ | 162 | 799 | 99.875 | 7 | 4.375 |
| 15 | NO | -   | 49  | 6.125  | 5 | 3.125 |
| 16 | SÍ | 281 | 348 | 43.5   | 5 | 3.125 |
| 17 | SÍ | 182 | 765 | 95.625 | 5 | 3.125 |
| 18 | SÍ | 244 | 721 | 90.125 | 7 | 4.375 |
| 19 | NO | -   | 175 | 21.875 | 7 | 4.375 |
| 20 | SÍ | 245 | 447 | 55.875 | 6 | 3.75  |
| 21 | SÍ | 410 | 401 | 50.125 | 6 | 3.75  |

Tabla 4.5: Parámetros con el estadístico Q calculados para todos los fallos

### 4.3. DIAGNÓSTICO DE FALLOS MEDIANTE TÉCNICAS DE APRENDIZAJE AUTOMÁTICO:

Tras la detección del fallo que se produce en la planta, llega el paso de saber de dónde procede, es decir, la anomalía que ha sido detectada tras aplicar el método de PCA se debe analizar y clasificar dentro de una de las clases de fallo que pueden ocurrir en la planta.

Para la clasificación de los fallos se hará uso de técnicas Machine Learning, planteándose diferentes procedimientos. Primero se va a desarrollar un método de clasificación compuesto por el análisis discriminante de Fisher (FDA), la incrustación de vecinos estocásticos distribuidos en t (t-SNE) y una red neuronal de retropropagación. Y, por otro lado, se usará la técnica de árboles de decisión, Random Forest (RF), para después analizar y comparar los resultados obtenidos.

#### 4.3.1. FDA – TSNE – RED NEURONAL:

Para poder explicar claramente paso a paso el desarrollo de este modelo se va a realizar la clasificación correspondiente para los fallos 4 y 7, posteriormente, se pasará a ir aumentando el número de fallos para ver si son clasificados de manera correcta y analizar así la validez de este método hasta que, como ya veremos más adelante, deje de funcionar correctamente. Es decir, probaremos el método, en primer lugar, con los fallos 4 y 7, con los fallos 1, 4 y 7, con los fallos 1,2,4 y 7, y por último, con los fallos 1, 2, 4, 7 y 12, donde se ve que el modelo propuesto no funciona bien con ese número de fallos. Ante este suceso se va a desarrollar una alternativa con la que se va a intentar corregir este problema.

Este modelo empieza aplicando el discriminante de Fischer (FDA), técnica con la que se trata de reducir la dimensionalidad inicial con la que se trabaja. El conjunto de datos que va a ser usado para el desarrollo de este modelo es aquel en el que la simulación referida al funcionamiento normal del proceso cuenta con 500 observaciones y 52 variables, y las 21 simulaciones restantes cuentan con 479 observaciones de 52 variables. Teniendo en cuenta esto, el primer paso será obtener la matriz de datos  $X$ , que contiene los datos de la simulación de la planta trabajando en condiciones normales, la simulación cuando ocurre el fallo 1, y la simulación cuando ocurre el fallo 4. Hay que tener en cuenta que los datos de comportamiento normal cuentan con 21 observaciones más, luego se procede a eliminar las 21 observaciones primeras. Obtenemos así la matriz  $X$  de dimensiones  $((n + 1) \cdot 479) \times 52$ , siendo  $n$  el número de fallos a clasificar, luego en este caso  $X \in \mathfrak{R}^{1916 \times 52}$ . Una vez tenemos la matriz de partida, se calculará su vector medio (Ecuación 15), y también, el vector medio y la matriz de dispersión de cada una de las clases que la componen (Ecuación 14 y Ecuación 17). Gracias a estos valores, se pueden obtener la matriz de dispersión de la clase,  $S_w$  (Ecuación 16) y la matriz de dispersión entre clases,  $S_b$  (Ecuación 18), con las que después podemos obtener una descomposición de valores propios aplicando la ecuación 19. Una vez realizada la descomposición, podemos ver la contribución que tiene cada variable sobre las clases de fallo. Esta contribución la obtenemos en la matriz  $W_{vwFDA}$ , que contiene los vectores propios correspondientes a los  $n-1$  valores propios mayores, en este caso, a los 2 primeros valores propios.

Gracias a la matriz de contribuciones de cada variable  $W_{vwFDA}$  podremos obtener un nuevo conjunto de muestras de condiciones normales y para cada fallo de una menor dimensión sin perder la información que tiene mayor influencia sobre los resultados. Obtenemos así mediante la siguiente ecuación, las nuevas muestras de datos de dimensión  $479 \times 2$ , con las que será más fácil trabajar y analizar los resultados.

$$Z_i = W_{vwFDA}^T \cdot X_i^T$$

En la siguiente gráfica (Figura 4.6), podemos ver una representación de estas nuevas muestras, donde los puntos de color azul corresponden a los datos de condiciones normales,  $Z_0$ , los puntos de color amarillo corresponden a los datos de tipo de fallo 4,  $Z_4$ , y los puntos de color rosa corresponden a los datos de tipo de fallo 7,  $Z_7$ . Podemos observar así, que cada conjunto de datos está separado, permitiendo distinguirlos perfectamente de una manera visual.

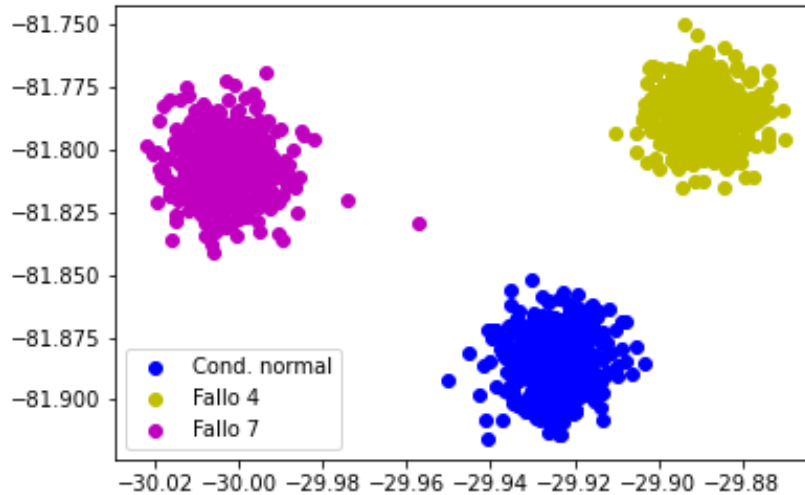


Figura 4.6: Representación de las nuevas muestras que tienen en cuenta la contribución de cada variable ( $Z_0$ ,  $Z_4$  y  $Z_7$ )

Una vez tenemos este conjunto, el siguiente paso será aplicar la técnica t-SNE. Este es un paso que se realiza de manera sencilla gracias a la librería “*sklearn.manifold*” que incorpora Python, que permite realizar esa reducción de dimensión tan solo con aplicar la función “*TSNE(perplexity=250).fit\_transform*”. El parámetro “*perplexity*” nos permitirá obtener una representación de los datos de manera más espaciada a medida que aumentamos su valor, haciendo aún más sencilla la visualización de los datos, en este caso un valor de 250 se considera apropiado. Con esta función podríamos elegir el número de variables a las que reducir el espacio de dimensiones, pero en este caso se ha reducido a la dimensión por defecto, es decir, se ha reducido a dos dimensiones. Este método t-SNE se aplicará a una matriz que contiene los conjuntos de datos de condiciones normales y de cada tipo de fallo anteriormente calculados, es decir, se aplica a una matriz  $Z \in \mathbb{R}^{1437 \times n}$ , de tal forma que se obtiene la matriz de dimensión reducida  $Y \in \mathbb{R}^{1427 \times 2}$ . Que estará formada por los datos referentes a comportamiento normal y a cada tipo de fallo, luego se procede a dividirla en las matrices individuales correspondientes  $Y_0$ ,  $Y_4$  y  $Y_7 \in \mathbb{R}^{479 \times 2}$ .

En la Figura 4.7 podemos ver la representación de estas matrices reducidas junto con el centroide que representa el valor medio de los datos que componen cada una de ellas, y que será clave para más adelante comparar y realizar la clasificación de fallo. Al igual que en la figura 4.6, los puntos de color azul corresponden a la matriz referida a comportamiento normal, los de color amarillo a la referida a fallo 4, y los de color rosa a la referida a fallo 7.

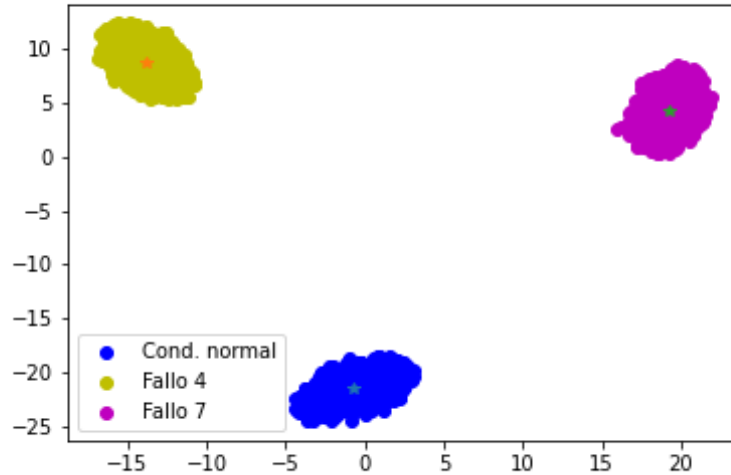


Figura 4.7: Representación de las matrices reducidas  $Y_0$ ,  $Y_4$  y  $Y_7$

El siguiente paso, será llevar a cabo una red neuronal de retropropagación (BP). Se comienza haciendo un entrenamiento de dicha red, en la que se ha de definir la entrada y la salida a red. Estas entradas deberán estar normalizadas, estando sus valores entre cero y uno. La entrada a la red será en este caso la matriz  $X$  original normalizada de comportamiento normal y cada tipo de fallo concatenados. La salida de la red será la matriz  $Y$ , que contiene al igual que en la entrada, las matrices  $Y$  de comportamiento normal y cada tipo de fallo obtenidas anteriormente mediante el método t-SNE, normalizada. El objetivo de esta red es obtener una matriz de predicción que en función de los datos de entrada nos de los valores obtenido por el método t-SNE en los cuales los fallos están perfectamente separados. La podemos ver representada en la figura 4.8.

Si comparamos la Figura 4.7, donde está representada la salida que estamos buscando, con la Figura 4.8, donde se representa la matriz de salida que hemos obtenido en el entrenamiento de la red, vemos que ambos resultados son muy parecidos, luego podemos intuir que este entrenamiento de la red se ha realizado de manera correcta y podremos pasar a probarla con los datos de planta de la simulación que se desea analizar.

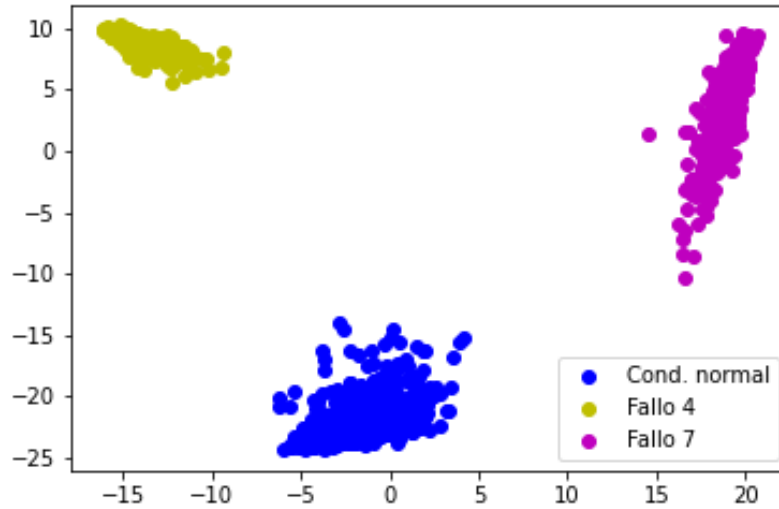


Figura 4.8: Representación de los resultados predichos por la red ( $Y_0_{predicha}$ ,  $Y_4_{predicha}$  y  $Y_7_{predicha}$ )

#### 4.3.1.1. CLASIFICACIÓN DE ANOMALÍAS:

Una vez la red ha sido entrenada, se procede a hacer la clasificación del fallo. Se comenzará normalizando una matriz que contenga los datos de la simulación que, como se ha mencionado anteriormente, se quiere analizar, y la matriz  $Y$  que se predijo en el entrenamiento de la red. Esta normalización se debe de hacer con los valores máximos y mínimos de las matrices con las que se entrenó la red. Una vez normalizadas, la primera se impone como entrada a la red, y la segunda como salida deseada, de tal manera que, si los datos de entrada son referidos a funcionamiento normal de planta, la salida de la red dará una matriz, que denominaremos como  $Y_{clase}$ , de valores que correspondan con comportamiento normal, si son de fallo 4, una matriz de valores que correspondan con fallo 4, y si son de fallo 7, una matriz de valores que correspondan con fallo 7. Representando estos resultados se puede ver la clasificación correctamente (Figura 4.9), donde todos los puntos de  $Y_{clase}$  representados en negro se van hacia la clase correspondiente, azul o comportamiento normal en la figura de la izquierda y al rojo o comportamiento de fallo 4 en la figura de la derecha.

Otra forma de ver que la clasificación se ha realizado correctamente, y que nos permite ver la cantidad exacta de observaciones que se han clasificado correctamente en un tipo de clase es mediante el cálculo de distancias mínimas, es decir, calcular la distancia de cada valor de la matriz predicha  $Y_{clase}$  a cada uno de los centroides de las  $Y$  calculadas, y ver a qué clase pertenece el dato concreto que

es el correspondiente a la distancia mínima entre ese dato y el centroide de cada clase, y esa distancia mínima indicará la clase donde se ha clasificado el punto. A continuación, se mostrará una tabla (Tabla 4.6) con los resultados obtenidos para cada tipo de comportamiento de planta.

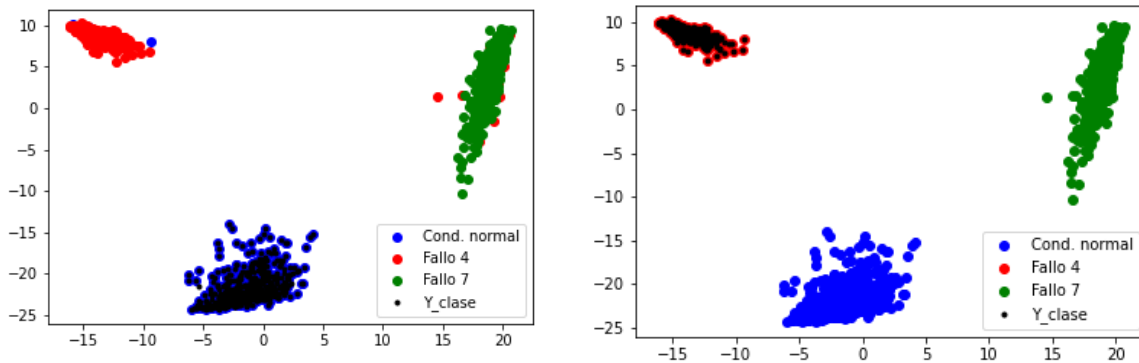


Figura 4.9: Clasificación de simulación de planta con comportamiento normal (Izquierda) y con fallo tipo 4 (Derecha)

|                              | <b>Comportamiento normal</b> | <b>Fallo 4</b> | <b>Fallo 7</b> | <b>Porcentaje de acierto</b> |
|------------------------------|------------------------------|----------------|----------------|------------------------------|
| <b>Comportamiento normal</b> | 479                          | 0              | 0              | 100                          |
| <b>Fallo 4</b>               | 0                            | 479            | 0              | 100                          |
| <b>Fallo 7</b>               | 0                            | 0              | 479            | 100                          |

Tabla 4.6: Clasificación de cada observación a un tipo de clase (3 posibilidades) con datos de entrenamiento

Para evaluar la eficacia de clasificación del método, se va a probar con datos de simulación distintos a los de entrenamiento, es decir, con el conjunto de datos de testeo, que incluye 160 observaciones de comportamiento normal de funcionamiento y 800 observaciones de cuando ocurre el tipo de anomalía correspondiente en cada simulación.

|                       | Comportamiento normal | Fallo 4 | Fallo 7 | Porcentaje de acierto |
|-----------------------|-----------------------|---------|---------|-----------------------|
| Comportamiento normal | 960                   | 0       | 0       | 100                   |
| Fallo 4               | 160                   | 800     | 0       | 100                   |
| Fallo 7               | 161                   | 0       | 799     | 99.8958               |

Tabla 4.7: Clasificación de cada observación a un tipo de clase (3 posibilidades) con datos de testeo

Esta vez, vemos que la clasificación obtenida no es totalmente perfecta para la simulación en el que aparece una anomalía de clase 7 (Tabla 4.7), aún así, es solamente un dato el que difiere, siendo un porcentaje bastante alto y próximo a la clasificación perfecta.

A continuación, se evaluará el método diseñado añadiendo un tipo de fallo más a clasificar, es decir, evaluamos si el modelo funciona correctamente al trabajar con un número mayor de clases a distinguir (4 posibilidades). En concreto, ha sido añadido el fallo 1.

En la Figura 4.10 se puede observar los resultados de la clasificación de manera visual, trabajando con la simulación en la que ocurre un fallo de tipo 1 en la figura de la izquierda, y trabajando con la simulación en la que ocurre un fallo de tipo 4 en la figura de la derecha. Y en la Tabla 4.8 figuran los resultados exactos obtenidos en la evaluación de la clasificación con este número de fallos para las distintas simulaciones.

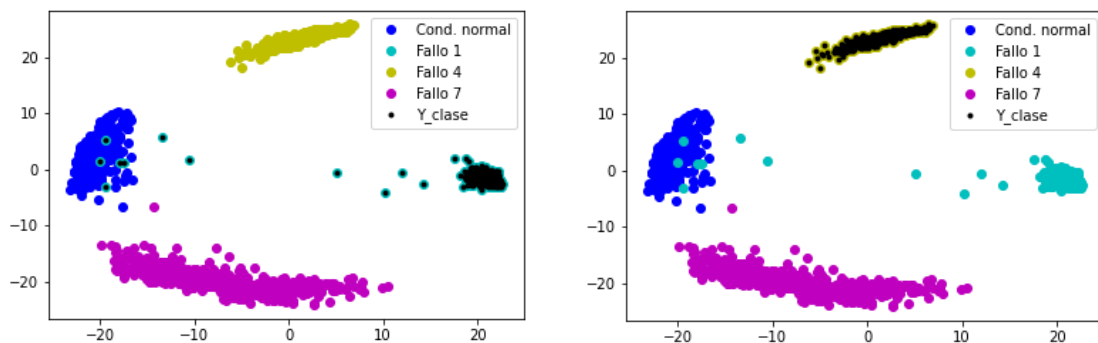


Figura 4.10: Clasificación de simulación de planta con fallo 1 (Izquierda) y con fallo tipo 4 (Derecha)

|                              | <b>Comportamiento normal</b> | <b>Fallo 1</b> | <b>Fallo 4</b> | <b>Fallo 7</b> | <b>Porcentaje de acierto</b> |
|------------------------------|------------------------------|----------------|----------------|----------------|------------------------------|
| <b>Comportamiento normal</b> | 479                          | 0              | 0              | 0              | 100                          |
| <b>Fallo 1</b>               | 7                            | 472            | 0              | 0              | 98.5386                      |
| <b>Fallo 4</b>               | 0                            | 0              | 479            | 0              | 100                          |
| <b>Fallo 7</b>               | 1                            | 0              | 0              | 478            | 99.7912                      |

Tabla 4.8: Clasificación de cada observación a un tipo de clase (4 posibilidades) con datos de entrenamiento

Vemos que esta vez la clasificación cuando ocurre el fallo 1 y para la simulación cuando ocurre el fallo 7 ya no tienen un porcentaje de acierto del 100%. Esto es porque hay algunos datos que el método los detecta como otro tipo de fallo, pero aun así vemos que este número es bastante pequeño en comparación con los que clasifica de manera correcta, es decir, se obtiene un porcentaje de acierto muy próximo al 100%, luego podemos decir que el funcionamiento del método con este número de posibilidades de clases a distinguir es correcto.

Con el conjunto de simulaciones con datos de testeo se obtienen los siguientes resultados (Tabla 4.9). Comprobamos que la clasificación del método sigue siendo bastante buena, tan sólo han sido unos pocos datos los que se han clasificado de manera errónea.

|                              | <b>Comportamiento normal</b> | <b>Fallo 1</b> | <b>Fallo 4</b> | <b>Fallo 7</b> | <b>Porcentaje de acierto</b> |
|------------------------------|------------------------------|----------------|----------------|----------------|------------------------------|
| <b>Comportamiento normal</b> | 960                          | 0              | 0              | 0              | 100                          |
| <b>Fallo 1</b>               | 168                          | 792            | 0              | 0              | 99.1666                      |
| <b>Fallo 4</b>               | 160                          | 0              | 800            | 0              | 100                          |
| <b>Fallo 7</b>               | 163                          | 0              | 0              | 797            | 99.6875                      |

Tabla 4.9: Clasificación de cada observación a un tipo de clase (4 posibilidades) con datos de testeo

Probaremos ahora con un tipo de fallo más, es decir, con 5 posibilidades de clasificación. Añadimos el tipo de fallo 2. En la Figura 4.11 se muestra la clasificación de manera visual, aunque esta vez no se ve de forma tan clara como las anteriores veces ya que la separación entre los datos no es tan grande.



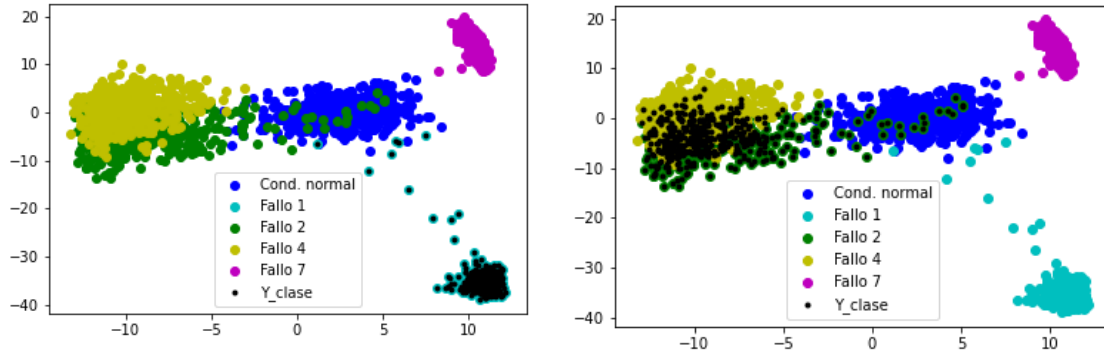


Figura 4.11: Clasificación de simulación de planta con fallo 1 (Izquierda) y con fallo tipo 2 (Derecha)

En la Tabla 4.10 se ve el porcentaje de acierto de la clasificación, que ahora de manera clara ha bajado, especialmente para los fallos 2 y 4. Observamos por tanto que, a medida que aumentamos el número de clases a clasificar el porcentaje de acierto disminuye. En la prueba del método con las simulaciones de testeo vemos que el porcentaje de acierto también baja, sobretodo en los fallos anteriormente mencionados que se confunden entre ellos (Tabla 4.11).

|                       | Comportamiento normal | Fallo 1 | Fallo 2 | Fallo 4 | Fallo 7 | Porcentaje de acierto |
|-----------------------|-----------------------|---------|---------|---------|---------|-----------------------|
| Comportamiento normal | 470                   | 0       | 8       | 1       | 0       | 98.1210               |
| Fallo 1               | 7                     | 472     | 0       | 0       | 0       | 98.5386               |
| Fallo 2               | 26                    | 0       | 341     | 112     | 0       | 71.1899               |
| Fallo 4               | 0                     | 0       | 107     | 372     | 0       | 77.6617               |
| Fallo 7               | 0                     | 0       | 0       | 0       | 479     | 100                   |

Tabla 4.10: Clasificación de cada observación a un tipo de clase (5 posibilidades) con datos de entrenamiento

Se va a probar a añadir un tipo de fallo más, comprobando el mal funcionamiento del método ante 6 posibilidades de clasificación. Añadimos así, también la posibilidad de clasificación ante fallo 12. En la Tabla 4.12, se puede observar la mala clasificación. A parte de que nos aparecen datos de hasta 3 tipos diferentes de comportamiento, la cantidad de datos de cada tipo es demasiado grande y similar entre ellas, bajando el porcentaje de acierto hasta menos del 50%. Podemos confirmar entonces el mal funcionamiento del método ante esta cantidad de clases de datos a clasificar. Lo mismo se puede confirmar viendo la Figura 4.12, resultado

del método cuando hay 6 clases distintas. Donde se ve que tres de las clases (comportamiento normal, fallo 12 y fallo 4) aparecen prácticamente solapadas, por lo que es muy difícil diferenciarlos.

|                       | Comportamiento normal | Fallo 1 | Fallo 2 | Fallo 4 | Fallo 7 | Porcentaje de acierto |
|-----------------------|-----------------------|---------|---------|---------|---------|-----------------------|
| Comportamiento normal | 946                   | 0       | 13      | 1       | 0       | 98.5416               |
| Fallo 1               | 163                   | 791     | 6       | 0       | 0       | 99.0625               |
| Fallo 2               | 180                   | 0       | 637     | 143     | 0       | 83.0208               |
| Fallo 4               | 161                   | 0       | 176     | 623     | 0       | 81.5625               |
| Fallo 7               | 165                   | 0       | 0       | 0       | 795     | 99.4791               |

Tabla 4.11: Clasificación de cada observación a un tipo de clase (5 posibilidades) con datos de testeo

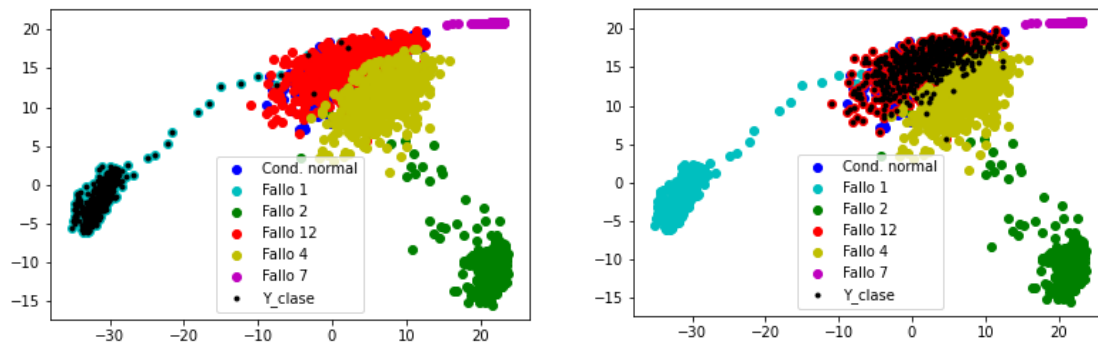


Figura 4.12: Clasificación de simulación de planta con fallo 1 (Izquierda) y con fallo tipo 12 (Derecha)

|                       | Comportamiento normal | Fallo 1 | Fallo 2 | Fallo 12 | Fallo 4 | Fallo 7 | Porcentaje de acierto |
|-----------------------|-----------------------|---------|---------|----------|---------|---------|-----------------------|
| Comportamiento normal | 230                   | 0       | 0       | 202      | 47      | 0       | 48,016                |
| Fallo 1               | 0                     | 469     | 0       | 10       | 0       | 0       | 97,912                |
| Fallo 2               | 13                    | 0       | 450     | 0        | 16      | 0       | 93,945                |
| Fallo 12              | 181                   | 0       | 0       | 240      | 58      | 0       | 50,104                |
| Fallo 4               | 35                    | 0       | 0       | 0        | 444     | 0       | 92,69                 |
| Fallo 7               | 0                     | 0       | 0       | 0        | 0       | 479     | 100                   |

Tabla 4.12: Clasificación de cada observación a un tipo de clase (6 posibilidades) con datos de entrenamiento

Probando, al igual que hemos hecho en los casos anteriores, con los datos de testeo observamos que los porcentajes de acierto disminuyen aún más, bajando incluso hasta un 30% aproximadamente para fallo 7 (Tabla 4.13).

|                              | <b>Comportamiento normal</b> | <b>Fallo 1</b> | <b>Fallo 2</b> | <b>Fallo 12</b> | <b>Fallo 4</b> | <b>Fallo 7</b> | <b>Porcentaje de acierto</b> |
|------------------------------|------------------------------|----------------|----------------|-----------------|----------------|----------------|------------------------------|
| <b>Comportamiento normal</b> | 401                          | 0              | 0              | 469             | 90             | 0              | 41.7708                      |
| <b>Fallo 1</b>               | 76                           | 789            | 0              | 78              | 17             | 0              | 90.1041                      |
| <b>Fallo 2</b>               | 83                           | 0              | 769            | 82              | 26             | 0              | 88.75                        |
| <b>Fallo 12</b>              | 338                          | 0              | 0              | 495             | 127            | 0              | 68.2291                      |
| <b>Fallo 4</b>               | 115                          | 0              | 0              | 83              | 722            | 0              | 87.1875                      |
| <b>Fallo 7</b>               | 230                          | 0              | 0              | 126             | 475            | 129            | 30.1041                      |

Tabla 4.13: Clasificación de cada observación a un tipo de clase (6 posibilidades) con datos de testeo

Ante este suceso, se va a proponer una alternativa del método para intentar mejorarlo.

#### 4.3.1.2. MODELO ALTERNATIVO:

Se pretende que cuando la clasificación en clases sea de cinco o más tipos, sea usado este modelo en lugar del anterior propuesto, ya que se intenta corregir el error de funcionamiento que hemos comprobado que ocurre con el original ante este número de clases.

Este modelo alternativo no deja de ser una ligera modificación del modelo estudiado anteriormente. El procedimiento es el mismo que el anterior, con la única variante de que en lugar de disponer de una sola red neuronal con la que se trabaja con una matriz de entrada que contiene concatenados todos los  $X_i$ , una salida que contiene al igual, todos los  $Y_i$  concatenados, siendo la predicha de la misma forma, se va a trabajar esta vez con una red neuronal para cada clase individual, es decir, una red para clasificar el comportamiento normal, otra red para el fallo 1, otra red para fallo 2, y así para cada clase que participa.

Luego las técnicas usadas y el procedimiento que se lleva a cabo antes de llegar a usar redes neuronales es el mismo que el anterior, una vez tenemos calculadas las matrices reducidas  $X_0$ ,  $X_1$ ,  $X_2$ ,  $X_4$  y  $X_7$ , se procede a normalizar cada una de ellas entre valores de cero a uno, normalizando también las salidas deseadas  $Y_0$ ,  $Y_1$ ,

Y2, Y4 y Y7. Una vez normalizadas procedemos a entrenar cada red, obteniendo una Y predicha para cada una de ellas. En la Figura 4.13 podemos ver representadas estas matrices predichas, que al igual que en el caso anterior, nos servirán de referencia para ver a cuál de estas se aproxima la obtenida con los datos de simulación tipo.

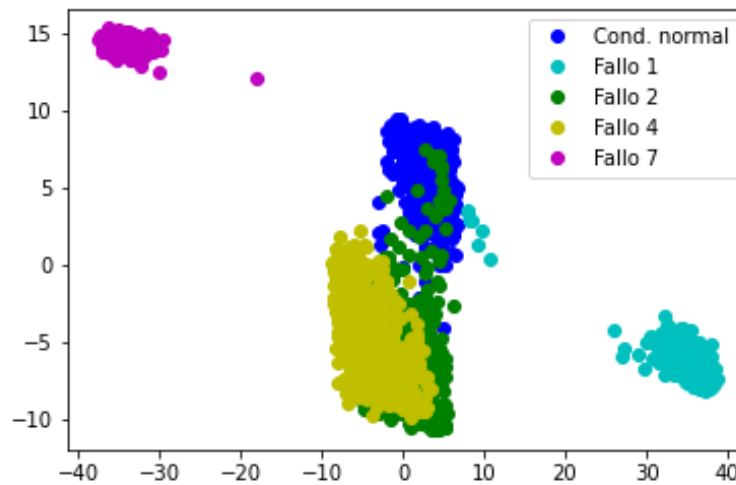


Figura 4.13: Representación de los resultados predichos por la red ( $Y_0$ \_predicha,  $Y_1$ \_predicha,  $Y_2$ \_predicha,  $Y_4$ \_predicha y  $Y_7$ \_predicha)

Una vez tenemos estas salidas predichas de referencia y las redes están entrenadas, pasaremos a realizar la clasificación del tipo de fallo.

Al igual que en modelo anterior, se cargarán los datos de la simulación de planta que queremos clasificar, pero esta vez se normalizará con cada mínimo y máximo de cada clase de fallo con las que hemos normalizado los ficheros de datos en el entrenamiento. Una vez normalizados los valores de entrada y salida, procedemos a calcular las Y predichas por clase. Al tener esta vez una por cada tipo de comportamiento que se puede dar en la planta, se ha de obtener la matriz  $Y_{clase}$ , matriz que contendrá los datos predichos referidos a cada clase donde la distancia de esta predicha y la predicha durante el entrenamiento es mínima.

Tanto en la representación gráfica de los resultados (Figura 4.14), como en los resultados analíticos (Tabla 4.14), podemos observar que el modelo alternativo, con el que se intentaba corregir el mal funcionamiento del método anterior, no da buenos resultados, ya que los porcentajes de acierto son bastante bajos, vemos que llegan a bajar hasta menos de un 30% con el fallo 2, que claramente se confunde

con el fallo 4 como se ve en la Figura 4.14. Además, estos porcentajes son aún menores que los obtenidos con el método anterior (Tabla 4.10).

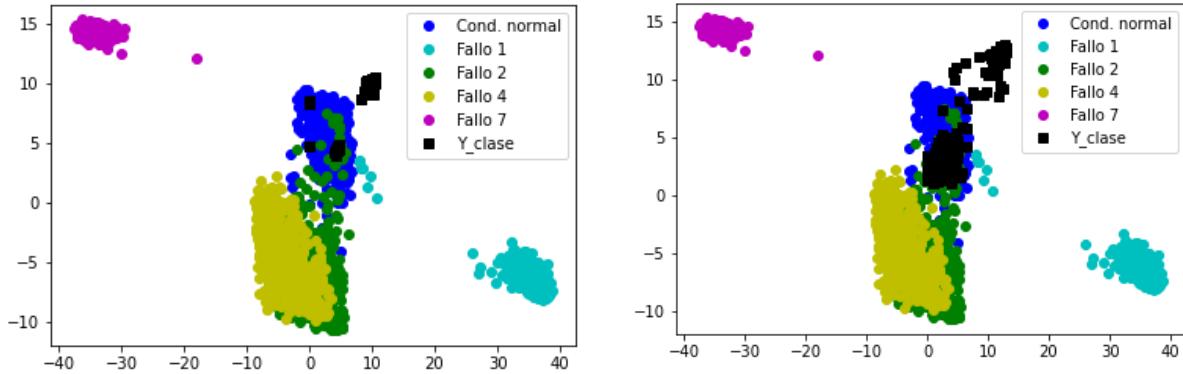


Figura 4.14: Clasificación de simulación de planta con fallo 1 (Izquierda) y con fallo tipo 2 (Derecha)

|                       | Comportamiento normal | Fallo 1 | Fallo 2 | Fallo 4 | Fallo 7 | Porcentaje de acierto |
|-----------------------|-----------------------|---------|---------|---------|---------|-----------------------|
| Comportamiento normal | 415                   | 0       | 0       | 1       | 63      | 86.6388               |
| Fallo 1               | 2                     | 464     | 0       | 0       | 13      | 96.8684               |
| Fallo 2               | 19                    | 0       | 140     | 7       | 313     | 29.2275               |
| Fallo 4               | 1                     | 0       | 121     | 232     | 125     | 48.4342               |
| Fallo 7               | 1                     | 0       | 0       | 0       | 478     | 99.7912               |

Tabla 4.14: Clasificación de cada observación a un tipo de clase (5 posibilidades) con datos de entrenamiento

|                       | Comportamiento normal | Fallo 1 | Fallo 2 | Fallo 4 | Fallo 7 | Porcentaje de acierto |
|-----------------------|-----------------------|---------|---------|---------|---------|-----------------------|
| Comportamiento normal | 805                   | 0       | 0       | 2       | 153     | 83.8542               |
| Fallo 1               | 141                   | 781     | 0       | 1       | 37      | 96.0416               |
| Fallo 2               | 146                   | 0       | 266     | 6       | 542     | 42.9166               |
| Fallo 4               | 131                   | 0       | 222     | 347     | 260     | 49.7916               |
| Fallo 7               | 132                   | 0       | 0       | 0       | 828     | 97.0833               |

Tabla 4.15: Clasificación de cada observación a un tipo de clase (5 posibilidades) con datos de testeo

Probando con los datos de test observamos que los porcentajes de acierto siguen siendo considerablemente bajos (Tabla 4.15), y, menores que los obtenidos anteriormente con el método sin modificar (Tabla 4.11).

Se va a probar también con 6 posibilidades de clasificación, es decir, con comportamiento normal, fallo 1, fallo 2, fallo 4, fallo 7 y fallo 12, como se ha hecho anteriormente con el otro método, comparando los resultados y afirmando que este modelo alternativo propuesto nos genera peores resultados.

Comparando los resultados obtenidos por este método (Tabla 4.16), con los obtenidos anteriormente (Tabla 4.12), podemos observar que los resultados obtenidos con la modificación del modelo generan mejores resultados como es con el fallo 12, pero peores para el resto de comportamientos.

|                       | Comportamiento normal | Fallo 1 | Fallo 2 | Fallo 12 | Fallo 4 | Fallo 7 | Porcentaje de acierto |
|-----------------------|-----------------------|---------|---------|----------|---------|---------|-----------------------|
| Comportamiento normal | 122                   | 0       | 0       | 212      | 143     | 2       | 25.4697               |
| Fallo 1               | 2                     | 469     | 0       | 8        | 0       | 2       | 97.9123               |
| Fallo 2               | 3                     | 0       | 447     | 5        | 24      | 0       | 93.3194               |
| Fallo 12              | 113                   | 0       | 0       | 303      | 63      | 0       | 63.2567               |
| Fallo 4               | 47                    | 0       | 0       | 84       | 348     | 0       | 72.6513               |
| Fallo 7               | 0                     | 0       | 0       | 0        | 2       | 477     | 99.5824               |

Tabla 4.16: Clasificación de cada observación a un tipo de clase (6 posibilidades) con datos de entrenamiento

Probando con los datos de testeo vemos que ocurre lo mismo (Tabla 4.17), el método alternativo propuesto no serviría para mejorar los resultados de clasificación ante este número de comportamientos de planta a clasificar.

|                       | Comportamiento normal | Fallo 1 | Fallo 2 | Fallo 12 | Fallo 4 | Fallo 7 | Porcentaje de acierto |
|-----------------------|-----------------------|---------|---------|----------|---------|---------|-----------------------|
| Comportamiento normal | 227                   | 0       | 0       | 212      | 143     | 2       | 23.6458               |
| Fallo 1               | 48                    | 786     | 0       | 105      | 21      | 0       | 86.875                |
| Fallo 2               | 50                    | 0       | 775     | 110      | 45      | 0       | 85.9375               |
| Fallo 12              | 244                   | 0       | 2       | 578      | 136     | 0       | 76.875                |
| Fallo 4               | 120                   | 0       | 0       | 251      | 589     | 0       | 73.8542               |
| Fallo 7               | 41                    | 0       | 0       | 98       | 25      | 496     | 55.9375               |

Tabla 4.17: Clasificación de cada observación a un tipo de clase (6 posibilidades) con datos de testeo



#### 4.3.2. BOSQUE ALEATORIO (RANDOM FOREST):

Para concluir la experimentación, se va a realizar una clasificación mediante árboles de decisión. Como ya se ha visto en el Capítulo II, no sólo se va a utilizar un solo árbol de decisión, sino que se van a entrenar varios árboles, por ello lo denominamos bosque aleatorio o “random forest” (RF). El número de árboles que se entrenan en este caso va a ser buscado por el propio programa, siendo el parámetro con el que se obtienen mejores resultados.

El bosque aleatorio debe de ser creado y entrenado, para que posteriormente, realice el análisis y clasificación de manera correcta. Para ello se han usado los dos conjuntos de datos distintos de los que hablamos al principio del capítulo. Para realizar el entrenamiento se va a usar el conjunto de datos que contiene menos observaciones, eliminando como en el modelo propuesto anterior las 21 primeras observaciones de la simulación de comportamiento normal, es decir, dispondremos de simulaciones con 479 observaciones de 52 variables. En cambio, a la hora de hacer la evaluación del modelo, se usará el conjunto de datos de fallo diferente, cuyas primeras 160 observaciones siempre son de comportamiento normal, y las restantes de cada tipo de fallo.

En este apartado se van a realizar dos evaluaciones para ver el rendimiento del modelo. Primero se hará la evaluación con cinco fallos, y después se hará la evaluación con todos los tipos de fallo. Las clases a evaluar en el primer caso son concretamente el comportamiento normal de planta, y cuando ocurre un fallo de tipo 1, tipo 2, tipo 4 y tipo 7.

El procedimiento comienza cargando los datos de entrada, la cual será una matriz que contiene concatenados uno detrás de otro todos los datos referidos a cada clase de fallo. Es decir, se crea una matriz de entrada  $X$  que contiene 2395 observaciones de 52 variables, siendo las 479 observaciones primeras de comportamiento normal, las siguientes 479 de fallo 1 y así seguidamente.

Se debe definir también la salida deseada. Lo que se pretende con el método es que al analizar los datos de entrada que hemos fijado, nos los clasifique en las 5 clases de funcionamiento de planta de manera correcta, es decir, tendremos que crear un vector de tamaño  $(5 \times 479)$  en el que los primeros 479 valores estén asociados a comportamiento normal, los 479 siguientes a fallo 1, los siguientes a fallo 2, los siguientes a fallo 4 y los últimos siguientes a fallo 7. Se asociará el valor 0 a comportamiento normal, el valor 1 a fallo 1, el valor 2 a fallo 2, el valor 4 a fallo 4, y el valor 7 a fallo 7.

Una vez definidas la entrada y la salida deseada se crea el bosque. Como se ha mencionado anteriormente, el propio Python se encargará de buscar el número de árboles con el que el rendimiento del modelo es óptimo. En este caso, el número de árboles entrenados son 204. La salida creada por el bosque será una matriz de predicción que, en este caso, coincide al cien por cien con los deseados, comprobando la eficacia del método. Esto puede verse de manera clara en una matriz de confusión (Figura 4.15).

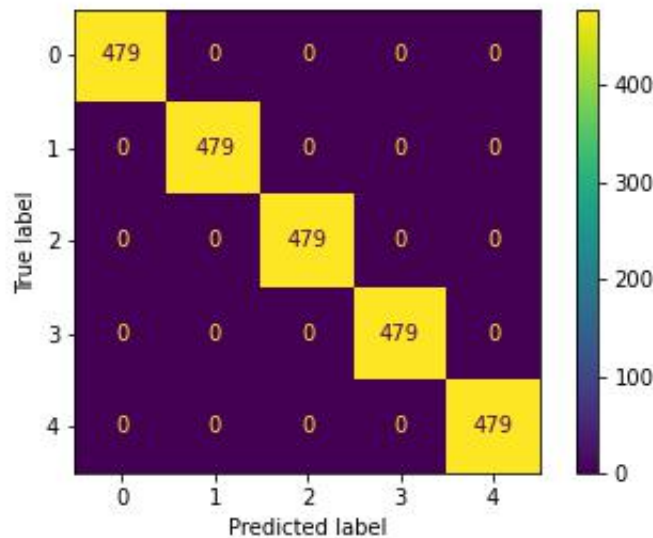


Figura 4.15: Matriz de resultados con datos de entrenamiento.

#### 4.3.2.1. CLASIFICACIÓN DE ANOMALÍAS:

A continuación, se evaluará la capacidad de clasificación del método probando con los datos de testeo. Esta vez la matriz de entrada corresponderá a una sola de las simulaciones que se desea evaluar, obteniendo, tras hacer la clasificación mediante el árbol anteriormente entrenado, la predicción correspondiente. Se probará con cada una de las simulaciones de las distintas clases a evaluar.

En la siguiente figura (Figura 4.16) se muestran los resultados obtenidos de clasificación para la simulación cuando ocurre el fallo 1 y para aquella en la que ocurre el fallo 7. Lo perfecto es que 160 se hubieran clasificado como comportamiento normal y el resto como el tipo de fallo correspondiente, como ocurre con la simulación de fallo 7 (figura de la derecha), sin embargo, podemos ver que para la simulación de fallo 1 (figura izquierda) algunos datos de fallo se han clasificado como comportamiento normal. Aún así, estas desviaciones son muy pequeñas, luego la clasificación es bastante buena.



En la siguiente tabla (Tabla 4.18) se muestran estos resultados analíticamente, mostrando el número exacto de aciertos, así como el porcentaje de clasificación, incluyendo también los resultados obtenidos con el resto de fallos. Finalmente, el porcentaje total de acierto de este método con 5 fallos es del 98.05%, precisión de la predicción del modelo, cuyo valor podemos obtener directamente mediante Python.

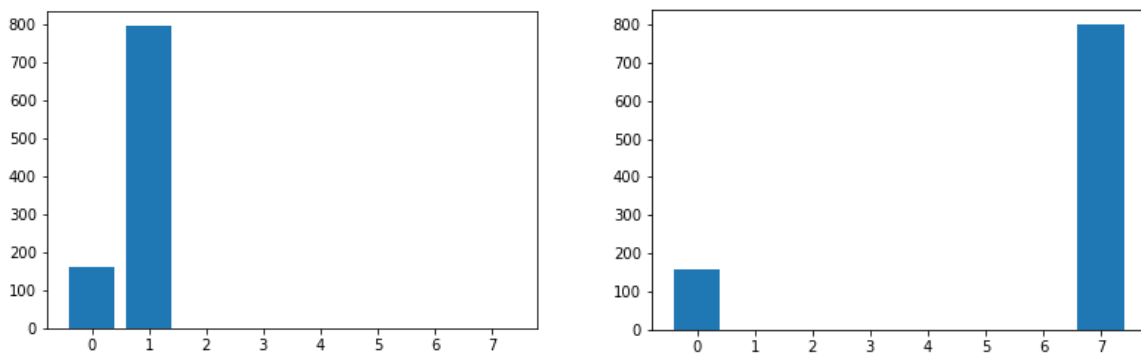


Figura 4.16: Histograma de clasificación de RF para simulación de fallo 1 (Izquierda) y para simulación de fallo 7 (Derecha)

|                      | Condiciones normales | Fallo 1 | Fallo 2 | Fallo 4 | Fallo 7 | Porcentaje de acierto |
|----------------------|----------------------|---------|---------|---------|---------|-----------------------|
| Condiciones normales | 884                  | 0       | 65      | 0       | 11      | 92.083                |
| Fallo 1              | 162                  | 789     | 0       | 0       | 0       | 99.7916               |
| Fallo 2              | 175                  | 0       | 785     | 0       | 0       | 98.4375               |
| Fallo 4              | 160                  | 0       | 0       | 800     | 0       | 100                   |
| Fallo 7              | 160                  | 0       | 0       | 0       | 800     | 100                   |

Tabla 4.18: Clasificación de cada simulación a un tipo de clase (5 posibilidades) con datos de testeo

Vamos a probar también la capacidad de clasificación del bosque aleatorio ante 6 posibilidades de clases. Añadimos, al igual que hicimos en el apartado anterior, la simulación de planta ante fallo 12.

Se va a llevar a cabo el mismo procedimiento. En primer lugar, se va a realizar el entrenamiento del bosque, que, teniendo en cuenta la entrada y la salida deseada correspondiente, obtenemos una predicción perfecta con un total de 163 árboles.

En la figura 4.17 se representa la matriz de confusión donde podemos ver esta clasificación.

Si probamos con los datos de testeo observamos que los porcentajes de acierto en la clasificación siguen siendo considerablemente altos (Tabla 4.19). El porcentaje más bajo que se obtiene es para la simulación de comportamiento normal, cuyo porcentaje de acierto es del 83.75%, podemos ver la clasificación representada en la figura 4.18 (Izquierda). En la figura 4.18 (Derecha) se representa la clasificación obtenida con la simulación de fallo 4, cuyo porcentaje de acierto ha sido del 100%. El porcentaje total de acierto de este método con 6 posibilidades es del 96%, un porcentaje menor que con 5 fallos, luego podemos deducir que al aumentar en número de fallos la eficacia del método disminuye. Pero mucho más alta si la comparamos con el método anterior.

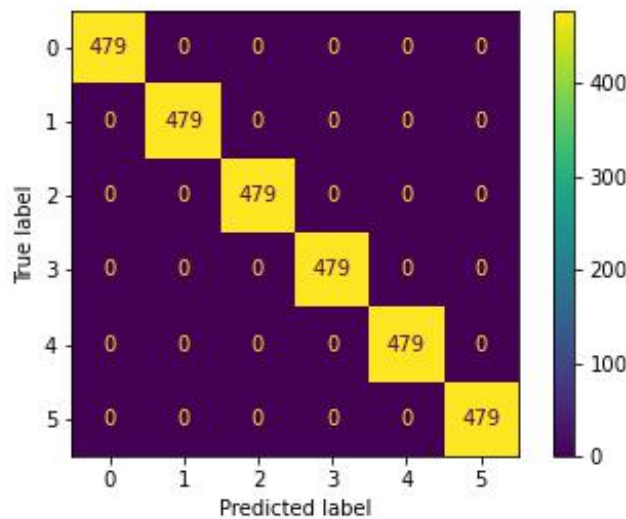


Figura 4.17: Matriz de resultados con datos de entrenamiento.

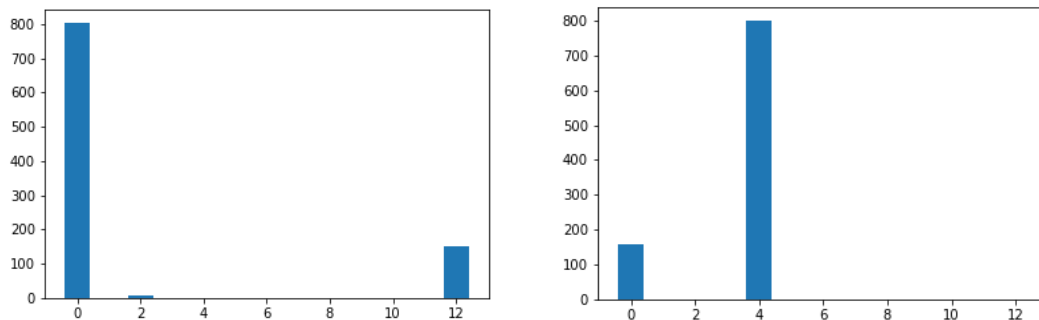


Figura 4.18: Histograma de clasificación de RF para simulación de comportamiento normal (Izquierda) y para simulación de fallo 4 (Derecha)



|                      | Condiciones normales | Fallo 1 | Fallo 2 | Fallo 4 | Fallo 7 | Fallo 12 | Porcentaje de acierto |
|----------------------|----------------------|---------|---------|---------|---------|----------|-----------------------|
| Condiciones normales | 804                  | 0       | 6       | 0       | 0       | 150      | 83.75                 |
| Fallo 1              | 162                  | 794     | 0       | 0       | 0       | 4        | 99.375                |
| Fallo 2              | 177                  | 0       | 783     | 0       | 0       | 0        | 98.2291               |
| Fallo 4              | 160                  | 0       | 0       | 800     | 0       | 0        | 100                   |
| Fallo 7              | 154                  | 0       | 1       | 0       | 797     | 8        | 99.0625               |
| Fallo 12             | 158                  | 16      | 1       | 0       | 4       | 781      | 97.8125               |

Tabla 4.19: Clasificación de cada simulación a un tipo de clase (6 posibilidades) con datos de testeo

Se va a probar ahora la capacidad de clasificación con todos los fallos que pueden darse en planta, a excepción de los tres fallos que, como se dijo anteriormente, son difíciles de clasificar (fallo 3, fallo 9 y fallo 15).

El procedimiento a seguir es igual que los anteriores. Teniendo en cuenta la entrada, y la salida deseada correspondiente, se realiza el entrenamiento del bosque, obteniendo con un número de 462 árboles una predicción con los datos de entrenamiento perfecta. En la figura 4.19 podemos ver esta clasificación.

Posteriormente, probando con los datos de testeo se puede comprobar que al aumentar el número de clases a clasificar la eficacia del método va a disminuir. En la siguiente figura (Figura 4.20) vemos la clasificación para la simulación de fallo 2 (figura Izquierda), que es en la que se ha obtenido mejor porcentaje de aciertos, con un 97.39%. Aunque también se producen casos en los que la desviación es mayor, como vemos en la figura de la derecha, donde se presenta el resultado de clasificación para la simulación de fallo 13. Observamos que se clasifican bastantes datos como fallos distintos a los que son, obteniendo un porcentaje de aciertos del 40.31%.

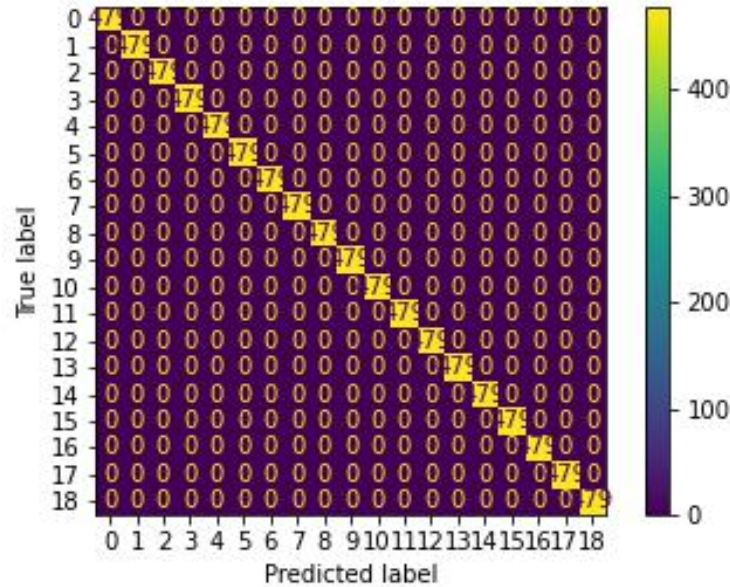


Figura 4.19: Matriz de resultados con datos de entrenamiento

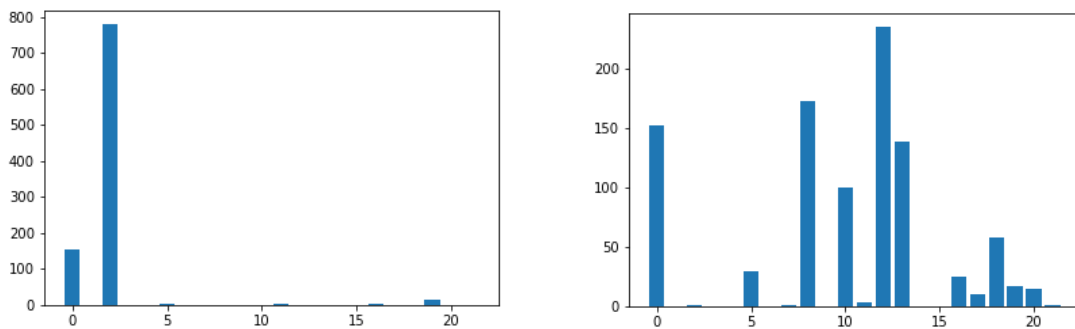


Figura 4.20: Histograma de clasificación de RF para simulación de fallo 2 (Izquierda) y para simulación de fallo 13 (Derecha) con datos de testeo

En la siguiente tabla (Tabla 4.20) se presentan los porcentajes de aciertos obtenidos para cada una de las simulaciones de cada clase. Si consideramos todos los fallos al mismo tiempo el porcentaje total de acierto es de un 72.89%, como se puede ver en la matriz de confusión de la Figura 4.21.

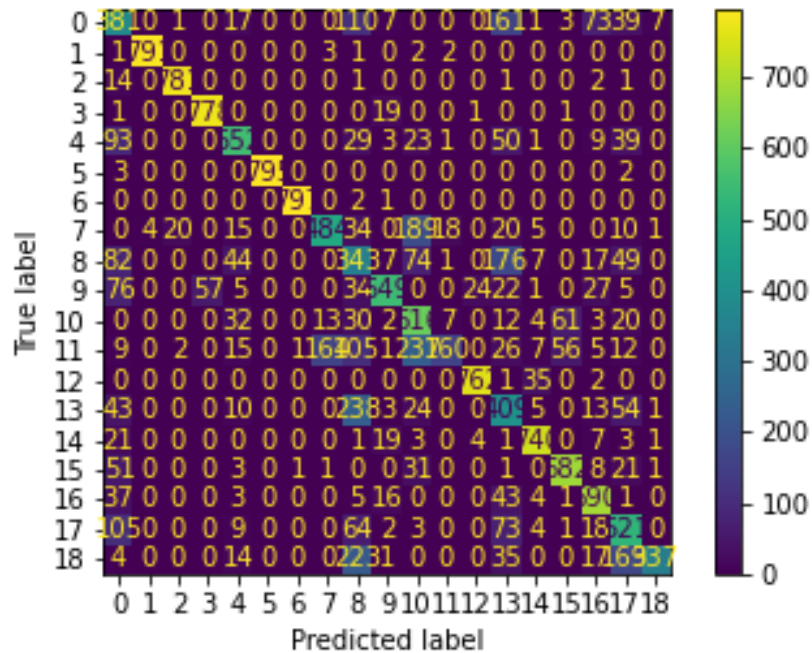


Figura 4.21: Matriz de resultados con datos de entrenamiento

|                              | Porcentaje de acierto |
|------------------------------|-----------------------|
| <b>Comportamiento normal</b> | 53.5416               |
| <b>Fallo 1</b>               | 97.1875               |
| <b>Fallo 2</b>               | 97.3958               |
| <b>Fallo 4</b>               | 92.7083               |
| <b>Fallo 5</b>               | 68.8542               |
| <b>Fallo 6</b>               | 96.6666               |
| <b>Fallo 7</b>               | 93.9583               |
| <b>Fallo 8</b>               | 60.1042               |
| <b>Fallo 10</b>              | 50.625                |
| <b>Fallo 11</b>              | 72.5                  |
| <b>Fallo 12</b>              | 75                    |
| <b>Fallo 13</b>              | 40.3125               |
| <b>Fallo 14</b>              | 89.7917               |
| <b>Fallo 16</b>              | 54.4792               |
| <b>Fallo 17</b>              | 86.5625               |
| <b>Fallo 18</b>              | 85.8333               |
| <b>Fallo 19</b>              | 91.0416               |
| <b>Fallo 20</b>              | 64.0625               |
| <b>Fallo 21</b>              | 44.4791               |

Tabla 4.20: Clasificación de cada simulación a un tipo de clase (19 posibilidades)

Por último, se va a probar a considerar solamente las simulaciones donde ocurre anomalía, es decir, no vamos a considerar los datos de comportamiento normal, ya que se puede usar PCA para detectar fallos, y una vez detectado el fallo, se puede usar este método de Random forest sólo para diagnosticar/clasificar el fallo que ha ocurrido en la planta. Se entrena el bosque, cuyos mejores resultados se obtienen con 487 árboles, dando lugar a una clasificación de los datos de entrenamiento perfecta (Figura 4.22).

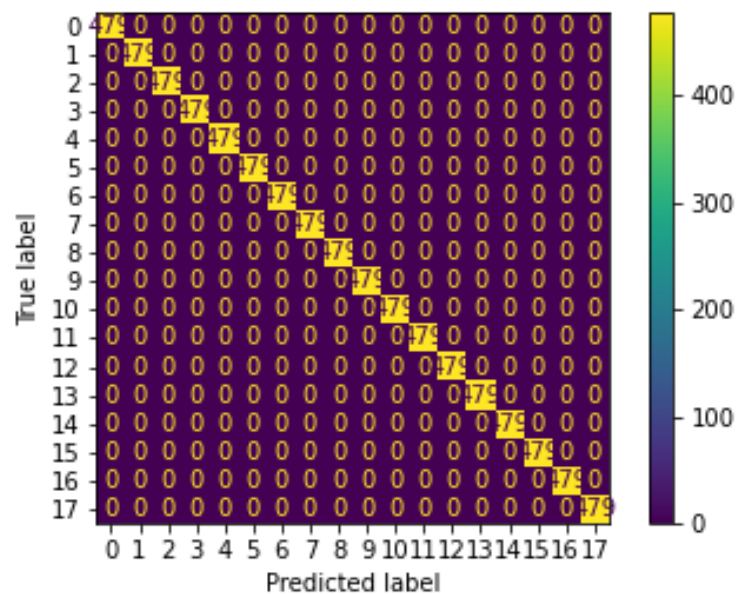


Figura 4.22: Matriz de resultados con datos de entrenamiento

Realizando la clasificación con los datos de testeo vemos que la clasificación obtenida es mejor que en el anterior caso para la mayoría de casos (Tabla 4.21). Se obtiene un porcentaje total de acierto del 75%, lo que corrobora que, si no tenemos en cuenta los datos de comportamiento normal de planta, los datos se clasifican mejor.

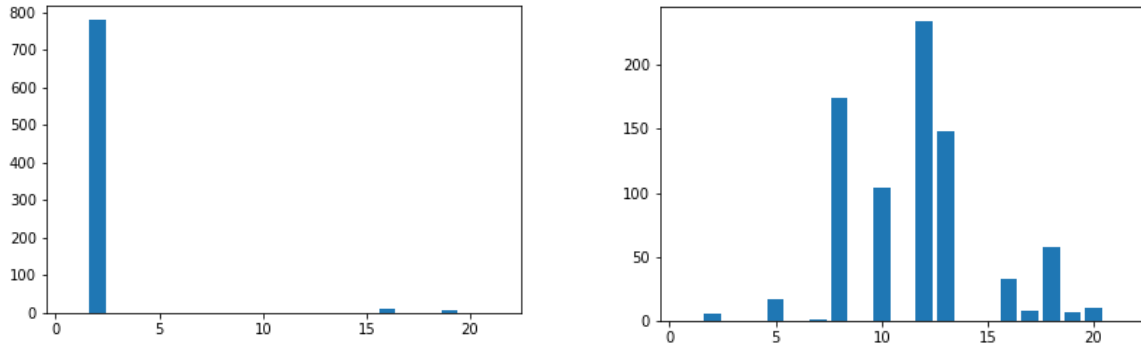


Figura 4.23: Histograma de clasificación de RF para simulación de fallo 2 (Izquierda) y para simulación de fallo 13 (Derecha)

|                 | Porcentaje de acierto |
|-----------------|-----------------------|
| <b>Fallo 1</b>  | 98.75                 |
| <b>Fallo 2</b>  | 97.625                |
| <b>Fallo 4</b>  | 95.875                |
| <b>Fallo 5</b>  | 74                    |
| <b>Fallo 6</b>  | 99.75                 |
| <b>Fallo 7</b>  | 99.75                 |
| <b>Fallo 8</b>  | 61.875                |
| <b>Fallo 10</b> | 43.625                |
| <b>Fallo 11</b> | 70                    |
| <b>Fallo 12</b> | 76.125                |
| <b>Fallo 13</b> | 29.25                 |
| <b>Fallo 14</b> | 94.625                |
| <b>Fallo 16</b> | 56.25                 |
| <b>Fallo 17</b> | 92.5                  |
| <b>Fallo 18</b> | 84.75                 |
| <b>Fallo 19</b> | 88                    |
| <b>Fallo 20</b> | 65.5                  |
| <b>Fallo 21</b> | 39.375                |

Tabla 4.21: Clasificación de cada simulación a un tipo de clase (18 posibilidades) con los datos de testeo

#### 4.4. COMPARACIÓN DE RESULTADOS:

En este último apartado del capítulo, se contrastarán los resultados obtenidos por ambos métodos que se han analizado, para así, averiguar cual de ellos es el más efectivo a la hora de diagnosticar los fallos de la planta. En la tabla 4.22 encontramos los resultados obtenidos con el modelo de FDA-tSNE-NN para cada una de las pruebas realizadas, y en la tabla 4.23 los obtenidos con el modelo de árboles de decisión.

|  | Datos de entrenamiento | Datos de testeo |
|--|------------------------|-----------------|
| <b>3 posibilidades (fallos 0, 4 y 7)</b>       | 100                    | 99.965          |
| <b>4 posibilidades (fallos 0, 1, 4 y 7)</b>    | 99.58                  | 99.713          |
| <b>5 posibilidades (fallos 0, 1, 2, 4 y 7)</b> | 89.102                 | 92.3333         |
| <b>6 posibilidades (fallos 0,1,2,4,7 y 12)</b> | 80.44                  | 67.69           |

Tabla 4.22: Comparación de resultados para la clasificación con FDA-tSNE\_NN

|  | Datos de entrenamiento | Datos de testeo |
|--|------------------------|-----------------|
| <b>5 posibilidades (fallos 0, 1, 2, 4 y 7)</b>                   | 100                    | 98.05           |
| <b>6 posibilidades (fallos 0, 1, 2, 4, 7, 12)</b>                | 100                    | 96              |
| <b>19 posibilidades (todos los fallos menos el 3, 9 y 15)</b>    | 100                    | 72.89           |
| <b>18 posibilidades (todos los fallos menos el 0, 3, 9 y 15)</b> | 100                    | 75              |

Tabla 4.23: Comparación de resultados para la clasificación con bosque aleatorio

Como se puede observar, a medida que se aumenta el número de fallos a clasificar el porcentaje de clasificación disminuye con ambos métodos.

Cuando el número de fallos a clasificar es pequeño, con 3 o 4 posibilidades de fallo, cualquiera de los dos métodos valdría para realizar la clasificación. Es a partir de ese número de fallos posibles a clasificar cuando el modelo de FDA-





# DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS  
INDUSTRIALES

Universidad de Valladolid

tSNE-NN no funcionaría correctamente, bajando su porcentaje de clasificación hasta un 67.69% cuando se hace una clasificación con 6 posibilidades. Sin embargo, con el método de bosque aleatorio el porcentaje de clasificación ante 6 posibilidades de fallo sigue siendo muy alto, 96%. Más aún, con este último método se ha probado a realizar la clasificación cuando se ponen datos de todo tipo de los fallos detectables, obteniendo aún así buenos resultados. Esto lo proclama como el mejor método, entre ambos, para el diagnóstico del comportamiento de planta.



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS INDUSTRIALES

Universidad de Valladolid



# CAPITULO V: CONCLUSIONES Y TRABAJO FUTURO



DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS INDUSTRIALES

Universidad de Valladolid



## 5.1. CONCLUSIONES:

En este trabajo se comenzaba hablando sobre la importancia que tiene hoy en día el control de calidad y su monitorización, requerida para trabajar con la gran cantidad de datos de la que se suele disponer. Es por ello que, en él, se ha hecho uso de diferentes técnicas de control estadístico de procesos y de aprendizaje automático con las que se ha experimentado tratando los datos procedentes de la planta química Tennessee Eastman.

En primer lugar, se ha desarrollado una aplicación basada en PCA, con la que se consigue detectar con éxito las anomalías que pueden darse en la planta. Para ello se ha hecho un estudio fuera de línea con los datos de funcionamiento normal, obteniendo los umbrales estadísticos correspondientes que nos permiten, al hacer el análisis posterior con los datos de fallo, ver cuándo ocurre o no el fallo.

Posteriormente, se han desarrollado diferentes modelos basados en el aprendizaje automático para diagnosticar estos fallos detectados. Se comienza con una aplicación que combina las técnicas FDA, t-SNE y redes neuronales, para la cual, comprobamos su correcto funcionamiento cuando las posibilidades de fallo no supera un valor de cuatro, y, por tanto, su mala clasificación si añadimos más fallos. A todo esto, se desarrolla un modelo alternativo, el cual, es una simple modificación de este. En el primero se entrenaba una sola red que tenía en cuenta el conjunto total de datos de las simulaciones, en cambio, este modelo alternativo entrena una red por cada fallo que puede darse en la planta. No obstante, los resultados obtenidos con este modelo alternativo no son mejores.

Finalmente, se crean bosques aleatorios, con los que se comprueba la gran eficacia de este método ya sea trabajando pocas posibilidades de clasificación de fallo, o con todas las 19 máximas posibles que pueden darse en la planta. Además, puede verse que los porcentajes de clasificación son mejores que con las otras aplicaciones desarrolladas.

Se comprueba también que, a medida que el número de posibilidades de fallo a clasificar aumenta, los resultados de clasificación obtenidos por cada uno de los métodos disminuyen.

Como conclusión final podemos decir, que el método PCA nos ha permitido detectar todos los fallos, excepto los fallos 3,9 y el 15 cuyo comportamiento es similar a los datos de comportamiento normal, aunque un conjunto de fallos se detectan pronto y con una tasa alta de aciertos, como pueden ser los fallos 1, 7, 12



o 14 y otros en cambio, aunque se detectan se tarda más tiempo en detectarles y su tasa de acierto es menor, como los fallos 16 o 21.

Para el diagnóstico de fallos podemos concluir que el mejor método es el de Random Forest o bosque aleatorio, aunque cuando se pretenden clasificar 18 fallos distintos, el rendimiento baja del 80% pero aún así se puede considerar satisfactorio.

## 5.2. TRABAJO FUTURO:

Respecto a lo que se podría hacer en un futuro para mejorar los resultados obtenidos en este trabajo, sería mirar más técnicas de control estadístico de procesos para mejorar la detección de fallos obtenida con PCA, como puede ser el Análisis de variables canónicas (CVA), o el Análisis de componentes independientes (ACI).

También se podría trabajar con técnicas de aprendizaje automático para detectar fallos, no solo para clasificarlos, como técnicas de aprendizaje profundo autoencoders (AE) o autoencoders variacionales (VAE).

Por último, en este trabajo se ha visto que las técnicas de aprendizaje automático que han sido usadas en la diagnosis/clasificación de fallos, nos dan unos resultados que podrían ser mejorados. Luego, se puede trabajar también con otras técnicas de aprendizaje automático como máquinas de vectores de soporte (SVM), o redes convolucionales (CNN), para intentar mejorarlo.



## BIBLIOGRAFÍA:

[1] J. Rodríguez, (2021), “Qué es el control de calidad y para qué sirve en una empresa”, HubSpot <https://blog.hubspot.es/sales/control-de-calidad> [Último acceso: 2 mayo 2023]

[2] P. Murphy, (2019), “The Manufacturing Side of Digital Transformation: Smart Factories”, IBM <https://www.ibm.com/blogs/think/2019/03/the-manufacturing-side-of-digital-transformation-smart-factories/> [Último acceso: 2 mayo 2023]

[3] R. Carro Paz y D. González Gómez, “Control Estadístico de Procesos”, Universidad Nacional de Mar del Plata <https://www.studocu.com/es-mx/document/fundacion-universitaria-iberoamericana-mexico/planificacion-y-gestion-de-proyectos/12-control-estadistico/10777921> [Último acceso: 6 mayo 2023]

[4] Software SuperCEP, La variabilidad y la calidad ¿cómo se relacionan? <https://www.softwaresupercep.com.mx/2012/05/24/la-variabilidad/> [Último acceso: 11 mayo 2023]

[5] Design Thinking España, Diagrama Ishikawa <https://designthinkingespaña.com/diagrama-ishikawa-2> [Último acceso: 11 mayo 2023]

[6] FJ. Soto González, (2001), “Relación entre estudios de capacidad y metodología seis sigma”, Universidad de Sonora

[7] La librería del conocimiento, Control Estadístico de los Procesos (SPC) <https://lalibreriadelconocimiento.blogspot.com/2016/10/control-estadistico-de-los-procesos.html> [Último acceso: 11 mayo 2023]

[8] M.A. Vargas Guevara, “Gráficas de control para variables. Capítulo 2” [https://controlestadisticocarloscastillo.weebly.com/uploads/3/9/2/0/3/9203091/capitulo\\_2.pdf](https://controlestadisticocarloscastillo.weebly.com/uploads/3/9/2/0/3/9203091/capitulo_2.pdf) [Último acceso: 14 mayo 2023]

[9] J. Amat Rodrigo, (2017), “Análisis de Componentes Principales (Principal Component Analysis, PCA) y t-SNE” [https://cienciadedatos.net/documentos/35\\_principal\\_component\\_analysis](https://cienciadedatos.net/documentos/35_principal_component_analysis) [Último acceso: 14 mayo 2023]



- [10] I. Pérez Franco, (2020), “Control de calidad de un proceso mediante la detección y diagnóstico de anomalías usando técnicas de control estadístico de procesos. Trabajo Fin de Grado en Ingeniería en Tecnologías Industriales”, EII, Universidad de Valladolid
- [11] M.J. Fuente, D. García-Alvarez , (2011), “*Estudio comparativo de técnicas de detección de fallos basadas en el Análisis de Componentes Principales (PCA)*” . Revista Iberoamericana de Automática e Informática industrial, Vol.8, pp:182–195.  
<https://www.elsevier.es/index.php?p=revista&pRevista=pdf-simple&pii=S1697791211000070&r=331> [Último acceso: 16 mayo 2023]
- [12] Gamco, Red Neuronal Artificial-ANN <https://gamco.es/glosario/red-neuronal-artificial-ann>[Último acceso: 23 mayo 2023]
- [13] R. Cañadas, (2021), “Redes neuronales artificiales”, Amdatum <https://abdatum.com/tecnologia/redes-neuronales-artificiales>[Último acceso: 10 junio 2023]
- [14] J. Martínez Heras, (2020), “Random Forest (Bosque Aleatorio): combinando árboles), IArtificial <https://www.iartificial.net/random-forest-bosque-aleatorio/>[Último acceso: 10 junio 2023]
- [15] Sitiobigdata, Árbol de decisión en Machine Learning <https://sitiobigdata.com/2019/12/14/arbore-de-decision-en-machine-learning-parte-1/>[Último acceso: 10 junio 2023]
- [16] Quantstart, Bootstrap Aggregation, Random Forests and Boosted Trees <https://www.quantstart.com/articles/bootstrap-aggregation-random-forests-and-boosted-trees/>[Último acceso: 10 junio 2023]
- [17] A. Pequeño Alonso, (2020), “Mejor del control de calidad de un proceso mediante técnicas de aprendizaje automático. Trabajo Fin de Grado en Ingeniería Química”, EII, Universidad de Valladolid
- [18] W. Lu y X. Yan, “Variable-weighted FDA combined with t-SNE and multiple extreme learning machines for visual industrial process monitoring”, ISA Transactions, vol.122, pp.163-171, 2022





DIAGNÓSTICO DE FALLOS/ANOMALÍAS EN UNA PLANTA INDUSTRIAL MEDIANTE EL MÉTODO FDA Y TÉCNICAS DE INTELIGENCIA COMPUTACIONAL



ESCUELA DE INGENIERÍAS INDUSTRIALES

Universidad de Valladolid

- [19] J. J. Downs y E.F. Vogel, "A plant-wide industrial process control problem.", Computer Chemical Engineering, vol.17, no3, pp. 245-255,1993
- [20] L. A. Alvarez Toro y O. Sotomayor, "Integración Optimización- Control predictivo y aplicación a la planta Tennessee Eastman", Rev. Per. Quím. ing. Quím. vol. 11 n.º 1, 2008.