Universidad de Valladolid

ESCUELA DE INGENIERÍAS INDUSTRIALES

Máster en Informática Industrial

MASTER EN INFORMÁTICA INDUSTRIAL

ESCUELA DE INGENIERÍAS INDUSTRIALES

UNIVERSIDAD DE VALLADOLID

TRABAJO FIN DE MÁSTER

FUZZY PCA PARA LA EXTRACCIÓN DE VARIABLES Y LA DETECCIÓN Y DIAGNÓSTICO DE FALLOS

Autor: D. Borja García Hernando
Tutor: D. Gregorio I. Sainz Palmero

Valladolid, Septiembre, 2016

Universidad de Valladolid

ESCUELA DE INGENIERÍAS INDUSTRIALES

Máster en Informática Industrial

MASTER EN INFORMÁTICA INDUSTRIAL

ESCUELA DE INGENIERÍAS INDUSTRIALES

UNIVERSIDAD DE VALLADOLID

TRABAJO FIN DE MÁSTER

FUZZY PCA PARA LA EXTRACCIÓN DE VARIABLES Y LA DETECCIÓN Y DIAGNÓSTICO DE FALLOS

Autor: D. Borja García Hernando
Tutor: D. Gregorio I. Sainz Palmero

Valladolid, Septiembre, 2016

Índice

Introducción

Las técnicas de PCA (Principal Components Analysis) tienen múltiples aplicaciones en distintos campos. En este Trabajo Fin de Máster (TFM) se quiere hacer hincapié en el uso de estas técnicas, en concreto su variante difusa, en los campos de extracción de variables, correspondiente a la primera parte de este trabajo, y a la detección y diagnóstico de fallos, que se podrá encontrar en la segunda parte.

Por último antes de comenzar con el trabajo en sí, cabe recalcar que se pueden encontrar todos los resultados en los anexos correspondientes, pues incluirlos directamente en este trabajo habría perjudicado su legibilidad.

Introducción al problema

El Análisis de Componentes Principales (PCA) es una técnica estadística multivariante que permite proyectar una matriz de datos en un subespacio construido mediante nuevas variables ortogonales y no correlacionadas obtenidas como combinaciones lineales de las variables originales.(MacGregor, J. y Kourti, T., 1995 [16]). Estas técnicas permiten una reducción de variables como se explicará más adelante en la sección 2, así como una compresión de los datos que permiten la obtención de patrones en los mismos. Estas propiedades son las indicadas para la resolución del problema general al que se enfrenta este trabajo.

En el sector industrial se registran multitud de variables a lo largo del proceso de producción, a parte de las que se emplean para el control de calidad de los productos. Las preguntas que plantean el problema principal son las siguientes:

¿Para qué se registran todas esas variables? y *¿Pueden emplearse esas variables para algo productivo?*

Las respuestas a estas preguntas conllevan a dar un uso a los datos de estas variables, generalmente orientado al control de calidad o el control de procesos. Pero el tamaño de los datos registrados suele ser muy grande y dificulta su uso, lo que plantea el siguiente problema:

¿Existe alguna manera de reducir este volumen quedándonos con lo más "importante" de manera eficiente para su empleo?

Una de las posibles respuestas son las técnicas PCA que, como se verá más adelante, permiten mantener una gran parte de la información recogida en unas pocas componentes principales, reduciendo así la dimensionalidad del problema de forma evidente.

Sin embargo, se ha observado que en ciertos procesos (Luukka, P., 2011 [14]), sobre todo en los no lineales, existen otras técnicas PCA más capaces de realizar de forma eficiente su función. Para este problema existe varias soluciones, de las cuales este trabajo se centra en las técnicas Fuzzy PCA o PCA difusas.

Multitud de trabajos (Cundari et al., 2002 [4], Heo et al., 2009 [6], Luukka, P., 2011 [14], Pop, H., 2001 [19], Sârbu, C. y Pop, H., 2005 [22] o Yang, T. y Wang, S., 1999 [28] son algunos ejemplos) han propuesto distintos algoritmos de estas técnicas, aunque en este trabajo se han escogido únicamente 9 para su estudio (sección 2.3). Con estos algoritmos se espera dar solución a este problema general, así como a los problemas más específicos de cada parte que ya se explicarán es sus correspondientes secciones.

Introducción a las Técnicas

Existen multitud de variantes de la técnica más clásica de PCA, cada una con distintas mejoras y consideraciones, por ejemplo, con el fin de reducir el número de falsas alarmas, detectar fallos consecutivos o detectar fallos en estados transitorios (Álvarez, D., 2013 [15]).

A continuación se muestran varias técnicas para dar una idea general de las distintas posibilidades y sus puntos fuertes.

Análisis de Componentes Principales (PCA) desarrollada independientemente por Pearson, K., 1901 [18] y por Hotelling, H., 1930 [7] permite reducir la dimensionalidad de un problema, realizar una extracción de variables, así como la detección y el diagnóstico de fallos mediante el índice de Hotelling o el índice de error de predicción cuadrado y los diagramas de contribución.

PCA Dinámica (Dynamic PCA) desarrollada por Ku et al., 1995 [9] no sólo extrae la relación de las variables en un instante determinado, sino que también tiene en cuenta los instantes previos, con el fin de extraer la dinámica del proceso.

PCA Adaptativo (APCA) desarrollada por Zumoften, D. y Basualdo, M., 2008 [32] actúa de la siguiente manera: Cuando se detecta un fallo, el modelo PCA se actualiza a la nueva situación y los estadísticos vuelven a estar bajo los límites de detección, permitiendo una nueva detección de fallo.

PCA pesado exponencialmente (Exponentially Weighted PCA o EWPCA) desarrollado por Lane et al.,2003 [10] y Tien et al., 2004 [23]considera el momento actual de manera más importante (con mayor peso) que los instantes pasados. Mediante esta técnica se pueden detectar fallos consecutivos y reducir el índice de falsas alarmas. Esto se consigue de la siguiente forma: el modelo PCA se establece de una manera adaptativa, como en la técnica anterior, y pesada de forma exponencial; todo ello de forma *on-line* empleando una ventana movil.

PCA's robustas (Robust PCA's) utilizan estimadores robustos, como el centro de la mediana, en vez de los estimadores clásicos de PCA (media y varianza) para su formulación, consiguiendo de esta forma eliminar gran parte de la sensibilidad a outliers intrínseca a PCA.

PCA's difusas (Fuzzy PCA's) son una alternativa a las PCA's robustas. Estas emplean la *fuzzyficación* de la matriz de datos empleando la lógica difusa con el fin de reducir la influencia de los outliers. Será en este tipo de PCA's donde se centre este trabajo.

PCA's no lineales (Non Linear PCA o NLPCA) son una categoría de técnicas que quieren dar solución a un problema intrínseco en PCA: si la relación de los datos es no lineal, linealizarlos no es la mejor opción, por ello se emplean combinaciones no lineales de las variables para extraer las componentes principales. Kernel PCA es una familia dentro de esta categoría, que, como su propio nombre indica, emplea los métodos kernel para que las operaciones lineales originales de PCA sean realizadas en un espacio kernel Hilbert reproducido con un mapeado no lineal. Schölkpf et al., 1998 [20] y Choi et al., 2005 [3] son algunos autores que han desarrollado sus correspon-

dientes algoritmos dentro de esta familia.

Mientras que para detección de fallos es posible encontrar multitud de técnicas, para el diagnóstico de fallos sólo existen unas pocas (Alcalá, C. F. y Qin, S. J., 2009 [1]), como pueden ser las contribuciones basadas en reconstrucción (Reconstruction-based contribution) desarrollada por Alcalá, C. F. y Qin, S. J., 2009 [1], el cual soluciona el problema de "*smearing*" que está asociado a los diagramas de contribución (Yoon, S. y MacGregor, J., 2001 [29]).

En Liu, J. y Chen, D., 2014 [13] se presenta otro enfoque de la técnica de Alcalá, C. F. y Qin, S. J., 2009 [1], el cual tiene la ventaja frente al anterior de que no necesita de conjuntos de datos con fallos para su ejecución.

Objetivos

Los objetivos principales de este trabajo son:

1. Estudio y análisis de técnicas de análisis de componentes principales, especialmente aquellas basadas en lógica difusa.
2. Análisis y estudio de técnicas de PCA difusas para la selección de variables y/o compresión de la información.
3. Análisis y estudio de técnicas de PCA difusas para su aplicación al problema de detección de fallos.
4. Análisis y estudio de técnicas de PCA difusas para su aplicación al problema de diagnóstico de fallos.

Organización del Trabajo

La organización de este trabajo sigue el siguiente orden:

Después de la introducción que tratará de forma general los problemas a resolver con las técnicas PCA, así como una breve introducción a las técnicas existentes; se expondrán los objetivos de este trabajo.

Seguidamente, en la primera parte del trabajo, se tratarán los temas relacionados con la extracción de variables empleando las técnicas de PCA, en la que se incluye la definición de PCA en su forma más clásica, así como la forma de implementarla y su interpretación geométrica. En esta misma parte se podrá encontrar un estado del arte de las técnicas PCA y una explicación de la lógica difusa (Fuzzy Logic) y como se relaciona con las técnicas PCA con el fin de dibujar un marco teórico para este trabajo. También se exponen los algoritmos a estudio, que serán empleados en la totalidad de este trabajo.

A continuación se presenta la metodología y los criterios de comparación empleados para la parte de extracción de características. Se finaliza la primera parte de este trabajo con los resultados obtenidos y las conclusiones extraídas de los mismos.

En la segunda parte, correspondiente a la detección y diagnóstico de fallos mediante las técnicas Fuzzy PCA, se sigue un patrón similar:

Se aborda primero el marco teórico, seguido de los problemas a enfrentar para las plantas de tratamiento de aguas (EDAR), las redes de distribución de agua y los reactores químicos, para, a continuación, introducir la metodología y los criterios empleados y finalmente los resultados obtenidos y las conclusiones extraídas de los mismos.

En esta segunda parte se reproduce esta organización dos veces, una para la detección de fallos y otra para el diagnóstico de los mismos.

Al final de este trabajo se encuentran las conclusiones generales, que recogen las conclusiones de ambas partes, a modo de síntesis y la bibliografía empleada.

Parte I
Extracción/Construcción de Variables mediante la Técnica de Análisis de Componentes Principales (PCA)

1. Introducción a la Extracción de Variables

Esta primera parte se centra en la extracción de variables mediante las técnicas de PCA. Como ya se dijo anteriormente, la organización es la siguiente:

En primer lugar se define la técnica PCA más clásica y se muestra un pequeño tutorial para su implementación, la cual se complementa mediante una interpretación geométrica de la misma.

Seguidamente se analiza el estado del arte de las técnicas PCA y Fuzzy PCA, lo que pretende proporcionar un marco de las técnicas existentes. Enlazando con el estado del arte correspondiente a Fuzzy PCA a continuación se puede encontrar unas nociones sobre lógica difusa y como se conecta esta con las técnicas PCA para hacerlas más robustas.

Para concluir el marco teórico de esta primera parte se presentan los algoritmos Fuzzy PCA escogidos para su estudio, describiendo en pseudocódigo cada uno de ellos.

En la siguiente sección se presenta una explicación de la extracción de variables, los problemas a resolver correspondientes a esta, la metodología empleada en esta parte, los criterios de análisis que se emplean para comparar los algoritmos escogidos entre sí, así como los datos empleados para realizar la comparación.

Se finaliza esta primera parte con un análisis de los resultados obtenidos de la comparación de los algoritmos escogidos y las conclusiones que se

pueden extraer de los mismos.

2. Análisis de Componentes Principales: Definición y Modelos

El Análisis de Componentes Principales o PCA (Principal Component Analysis) por sus siglas en inglés, es una técnica estadística multivariante que permite proyectar una matriz de datos en un subespacio construido mediante nuevas variables ortogonales y no correlacionadas obtenidas como combinaciones lineales de las variables originales.(MacGregor, J. y Kourti, T., 1995 [16]). Estas nuevas variables, también llamadas componentes principales, son los ejes del subespacio, y se encuentran orientados en las direcciones de máxima varianza, por lo que contienen la mayor parte de información del conjunto de datos original (Pop, H., 2001 [19]).

Este método lleva usándose en el análisis de problemas científicos con grandes cantidades de datos durante las últimas décadas por su capacidad para encontrar interrelaciones en los datos (Venkatasubramanian, V., 2003c [25]).

Las técnicas PCA calculan las componentes principales por medio de combinaciones lineales de las variables originales. Estos componentes principales tienen la particularidad de estar orientadas en las direcciones de máxima varianza, en los que se encuentra la mayor parte de la información de los datos del problema a resolver. Esto permite tratar los datos de la forma más concisa posible de forma objetiva al seleccionar un número de componentes principales menor al total de variables pero sin sacrificar un porcentaje elevado de información.

Desde que esta técnica fuera desarrollada en 1901 por Karl Pearson (LIII. On lines and planes of closest fit to systems of points in space) [18] como un método análogo del teorema de ejes principales, y más tarde desarrollada independientemente en 1930 por Harold Hotelling (British statistics and statisticians today) [7], ha sido muy estudiada y se han desarrollado multitud de variantes. La técnica PCA más básica y simple se realiza de la siguiente forma (Smith, L. I., 2002 [21]):

Algorithm 1 PCA

1: Obtener datos: $X = \{x^1, ..., x^m\}$
2: Normalizar datos: media = 0, desviación = 1
3: Calcular la matriz de covarianza empleando:

$$Cov_{i,j} = \frac{1}{m-1} \sum_{k=1}^{m} (x_i^k - \bar{x}_i) \cdot (x_j^k - \bar{x}_j), i, j = 1, \ldots, n. \quad (1)$$

4: Calcular los vectores propiosw_i y los valores propiosλ_i
5: Seleccionar las componentes principales atendiendo a:

$$\sum_{i=1}^{p} \lambda_i => \text{criterio de corte} \quad (2)$$

6: Obtener los nuevos datos:

$$\text{Datos Finales} = \text{Loadings}^T \times \text{Datos normalizados}^T \quad (3)$$

Se considera necesario aclarar que en el 1° paso X es una matriz de dimensión nxm donde n es el número de observaciones y m es el número de variables, en el cálculo de la matriz de covarianza se debe hacer sobre los datos normalizados (Pop, H., 2001 [19]), los vectores propios indican las direcciones de las posibles componentes principales y los valores propios muestran su significancia, y para terminar las aclaraciones indicar que p es el número de componentes principales que se han de mantener y el criterio de corte atiende al porcentaje de variabilidad que se decide mantener en tanto por uno, por ejemplo 0.85.

Ahora que ya se ha explicado cómo se realiza PCA, se muestra a continuación la interpretación geométrica de las técnicas PCA con datos de dos variables que se ha extraído de "A Tutorial on Principal Components Analysis" de Smith, L. I., 2002 [21]: Figura 1.

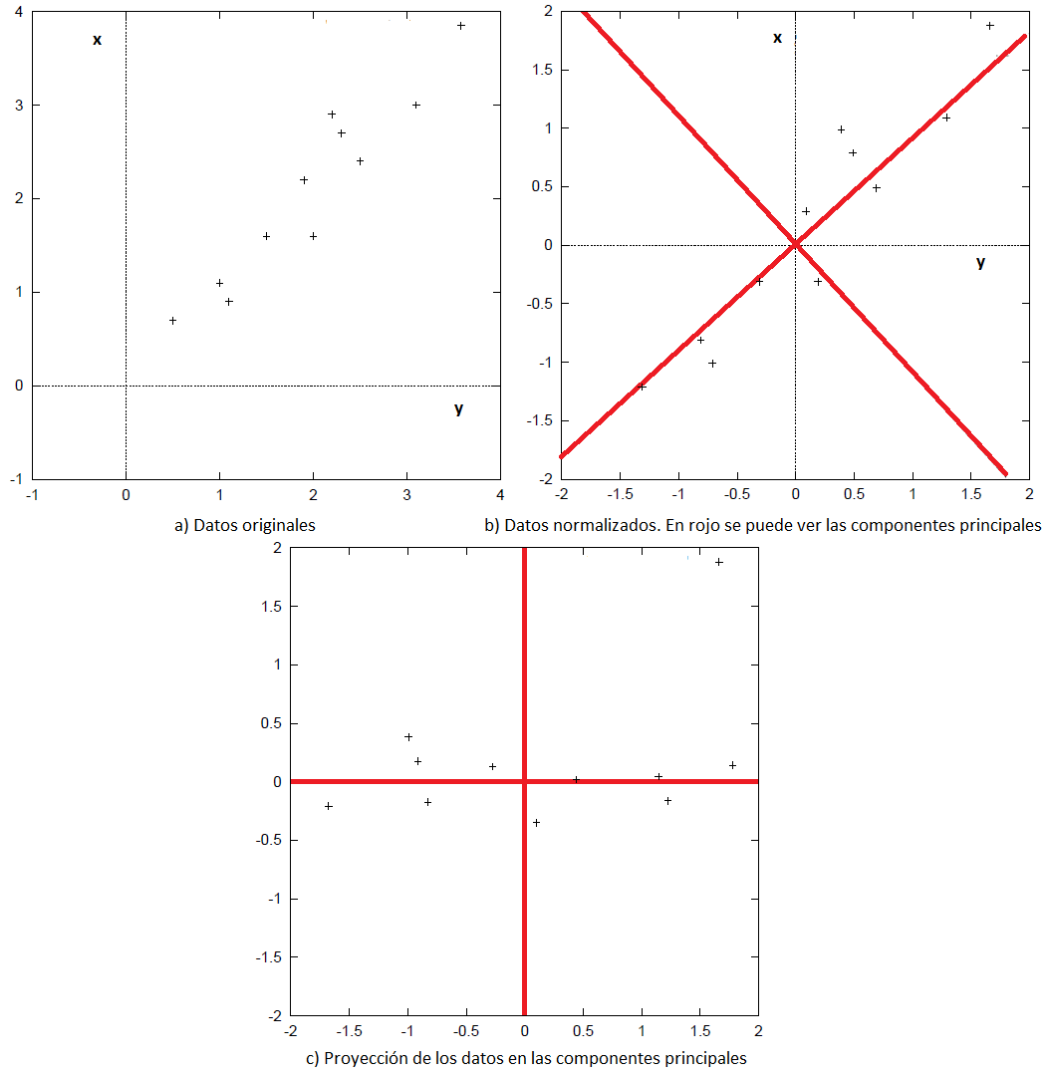

Figura 1: Interpretación geométrica de PCA. Obtenido de Smith, L. I., 2002 [21]

En la figura 1(a) se pueden observar los datos de ejemplo en su estado original. En este punto, no se han normalizado y simplemente se están representando. Si seguimos observando la figura 1(b), se puede ver la normalización de los datos, aunque cabe puntualizar que Smith [21], por ser un ejemplo sencillo, simplemente ha restado la media de cada variable a sus

datos. También en la figura 1(b) se marca en color rojo las componentes principales, que por ser un ejemplo de dos variables, únicamente se pueden obtener una o dos componentes principales, pero en este caso se han mantenido las dos componentes. Por último se obtienen los datos transformados al subespacio reducido formado por las componentes principales (en color rojo), como se puede apreciar en la figura 1(c).

Se puede observar que, en este ejemplo, se han seleccionado dos componentes principales, con lo que no se tiene pérdida de información. Se aprecia la reorientación de los ejes debido a la aplicación de PCA sobre los datos. El eje horizontal corresponde a la primera componente principal, y el vertical a la segunda componente principal. Como cabía esperar, los datos están orientados a lo largo de la primera componente principal puesto que es la dirección de mayor variabilidad.

2.1. Estado del arte del Análisis de Componentes Principales

El Análisis de Componentes Principales es una técnica clásica que ha tenido numerosas aplicaciones a lo largo del tiempo, en especial, en los problemas que requieran el tratamiento de grandes cantidades de datos.

El reconocimiento facial o la compresión de imágenes (según Smith, L. I., 2002 [21]) es uno de los campos de aplicación de PCA's debido a su capacidad para encontrar patrones en los datos y reducir la dimensionalidad de los mismos respectivamente.

El prepocesamiento y reducción de dimensionalidad de datos médicos también puede llevarse a cabo mediante PCA (Luukka, P., 2011 [14]), que unido a un método de clasificación es capaz de diagnosticar, de forma más eficaz que empleando solamente el método clasificador, las posibles enfermedades del paciente.

A la hora de realizar un modelo de un proceso, una opción interesante es utilizar para ello las técnicas PCA, puesto que permiten reducir la complejidad del problema y no necesitan de conocimientos del proceso a priori.

El campo de la detección de fallos y su diagnóstico también es susceptible de realizarse mediante PCA. De hecho, es una de las técnicas más empleadas en este entorno debido a que no es necesario un gran conocimiento del proceso a estudio y los resultados son rápidos y bastante precisos. Así mismo, no se necesita de un conocimiento a priori de los posibles fallos que puedan darse en el proceso. La detección de fallos y el diagnóstico de los mismos se puede aplicar a multitud de procesos distintos, tales como plantas de tratamiento de agua, redes de distribución de agua, reactores químicos, calderas, CSTR (Continuous Stirred Tank Reactor) y un largo etcétera.

Debido a esto, existen multitud de técnicas derivadas de PCA, por lo que aquí no se pretende abarcar todas ellas, aunque a continuación se van a presentar algunas técnicas PCA.

En Tien et al., 2004 [23] se hace un estudio comparativo entre varios métodos de detección y diagnosis usando PCA. Uno de los métodos que presenta este artículo es el método MPCA (Moving PCA) o PCA con ventana deslizante. El enfoque que propone este método es no utilizar la media y la varianza de los datos con los que se hizo el PCA para normalizar o escalar la nueva lectura del proceso, sino utilizar la media y la varianza de las últimas w lecturas recogidas (ventana deslizante), de está forma se busca que el dato se normalice con datos más próximos a su modo de operación y así que no se confunda por un fallo. El algoritmo que hay que seguir para aplicar este método consta de los siguientes pasos:

1. Adquirir información del proceso en condiciones normales. Escalar los datos usando su media y su desviación estándar. Llevar a cabo la descomposición SVD para obtener el modelo PCA del proceso. Calcular los umbrales de las estadísticas T^2_α y Q_α. Todos estos pasos se corresponden al cálculo fuera de línea.

2. Obtener las siguiente lectura de los datos de proceso x. Escalarla con los datos estadísticos de la ventana flotante.

3. Calcular las estadísticas T^2, Q y Z y comprobar si superan el umbral calculado en el paso 1. Si no lo excede la muestra x se considera normal.

4. Si x es normal actualizar la ventana deslizante con la nueva lectura eliminado la más antigua de la ventana. Volver al paso 2.

En Zumoften, D. y Basualdo, M., 2008 [32] se describe un método de detección, aislamiento y estimación de fallos (FDIE). Este método utiliza para la detección de fallos una variante del PCA normal denominada PCA adaptativo (APCA). El fundamento de este método es que cuando se detecta un fallo la media y la varianza con las que se escala o normaliza la siguiente lectura del proceso dejan de ser las del cálculo del PCA fuera de línea; y se pasa a escalar las nuevas lecturas con la media y la varianza de los datos de fallo, es decir, los que superan el umbral. De esta forma se produce un cambio en el espacio de los datos normales de proceso, siendo ahora el espacio normal de proceso el del fallo que se ha producido, se puede decir que el método se adapta al fallo. La ventaja que presenta este método con respecto a los anteriores es que cuando se produzca un nuevo fallo podrá volver a detectarlo, ya que las estadísticas volverán a salirse del umbral

El algoritmo que sigue esté método consta de las siguientes fases:

1. Tomar datos del proceso en funcionamiento normal y crear la matriz X. Normalizar los datos con media 0 y varianza 1 ($\bar{X}$). Calcular el PCA de estos datos. Establecer los límites de las estadísticas T^2_α y Q_α. Establecer el tamaño de la ventana deslizante N_w y la matriz normalizada $\bar{X}_N$, que será una ventana de datos normalizada que servirá para recalcular el PCA y los umbrales si fuese necesario.

2. Obtener la siguiente lectura x y normalizarla $\bar{x}$.

3. Obtener las estadísticas T^2, Q y Z con el modelo PCA actual.

4. Chequear si alguna de las estadísticas supera su umbral, si no lo supera se considera que la medida es normal. Si es normal añadir $\bar{x}$ a la matriz normalizada $\bar{X}_n$. Volver al paso 2.

5. Si la medida no es normal generar una falsa alarma. Si se producen un determinado número de falsas alarmas consecutivas notificar un fallo. Volver al paso 2.

6. Si se ha notificado un fallo, almacenar la medida del proceso en una matriz auxiliar X_{aux} y almacenar medida de fallo durante N_{aux} muestras. Volver al paso 2.

7. Cuando el tamaño de X_{aux} supera N_{aux}, actualizar los valores de la media y la varianza de normalización por la media y la varianza de X_{aux}.

8. Actualizar los límites de las estadísticas T^2_α y Q_α.

9. Actualizar el modelo del PCA con la matriz normalizada $\bar{X}_N$.

10. . Volver al paso 2.

Otro método descrito en Misra et al., 2002 [17] propone utilizar el PCA sobre la descomposición en frecuencias de las señales de las variables medidas, para esta descomposición se utilizan las wavelets. El método es denominado como MSPCA (Multi-scale principal component analysis). En la mayoría de las señales son las componentes de baja frecuencia las que le otorgan a la señal la mayor parte de su información. Mientras que las componentes de alta frecuencia se encargan de incorporar características más particulares. Es por esto por lo que se subdividen las componentes de una señal en dos categorías:

- Aproximaciones (baja frecuencia).
- Detalles (alta frecuencia).

Esta componentes se separan a través de filtros donde S es la señal que se desea analizar, A la salida del filtro pasa baja y D la salida del filtro pasa alta. Los filtros se diseñan de manera que sean complementarios, es decir, la suma de A y D debe de ser S.

Para muchas señales de mayor complejidad, se puede iterar el proceso de filtrado, es decir, aplicar el mismo procedimiento a las señales de salida de la primera etapa, y así sucesivamente hasta el nivel de precisión que se desee.Esto da origen a una descomposición multinivel conocida como ramificación o árbol de descomposición wavelet.

La forma de implementar el método es muy parecida a MPCA, y se puede esquematizar en los siguientes pasos:

1. Fuera de línea hay que recoger datos de las variables medidas la planta en funcionamiento normal, después, hay que realizar su descomposición en frecuencias usando wavelets para cada una de las variables, y juntar las variables por rango de frecuencia en distintas matrices de datos para calcular un PCA para cada una de ellas, después se pueden calcular los umbrales de las estadísticas para cada una de las frecuencias.

2. Para cada uno de las lecturas en línea del proceso x, añadirla a la ventana de datos y calcular la descomposición wavelet en frecuencias de la ventana de datos para obtener nuevas lecturas $D_1, \ldots, D_l$ y A_l.

3. Normalizar la lectura actual y las componentes wavelet $D_1, \ldots, D_l$ y A_l y calcular las estadísticas.

4. Si la estadística de A_l no supera el umbral considerarlo un dato normal y añadirlo a la ventana.

5. Si supera un umbral notificar una falsa alarma. Si ocurre un número determinado de falsas alarmas consecutivas notificar un fallo.

6. Volver al paso 2.

Si al método PCA se le añade información dinámica como se detalla en ku et al., 1995 [9], se obtiene el PCA dinámico. En el PCA dinámico (DPCA) se realiza el cálculo del PCA con las variables medidas en el instante actual y en instantes de tiempo pasado, si se considera sólo el estado anterior se dice que el retardo es uno, y se consideran los dos estados anteriores el retardo es de 2 y así consecutivamente. El modo de implementar este método es igual que el PCA simple salvo que a la hora de construir la matriz X de las muestras tomadas de las variables del proceso hay que tener en cuenta los estados anteriores, es decir, los retardos que se deseen (el operador | se refiere a la concatenación de matrices):

$$X = [X_t | X_{t-1} | \ldots | X_{t-h}] \tag{4}$$

$$\text{donde } X_t = \begin{bmatrix} x_1(t) & x_2(t) & \cdots & x_m(t) \\ x_1(t+1) & x_2(t+1) & \cdots & x_m(t+1) \\ x_1(t+2) & x_2(t+2) & \cdots & x_m(t+2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t+n) & x_2(t+n) & \cdots & x_m(t+n) \end{bmatrix}$$

La ecuación 4 muestra la matriz X con un retardo, si el retardo fuese de dos para cada variable medida tendría que haber tres columnas.

2.2. Lógica difusa (Fuzzy Logic)

Según Zadeh (Fuzzy sets, 1965 [30]):
"*La lógica difusa proporciona un mecanismo de inferencia que permite simular los procedimientos de razonamiento humano en sistemas basados en el conocimiento. La teoría de la lógica difusa proporciona un marco matemático que permite modelar la incertidumbre de los procesos cognitivos humanos de forma que pueda ser tratable por un computador*".

La lógica difusa (Fuzzy logic en inglés) es una lógica multivariada que permite representar matemáticamente la incertidumbre y la vaguedad, proporcionando herramientas formales para su tratamiento.

Según Zadeh (Outline of a new approach to the analysis of complex systems and decision processes, 1973 [31]):

"*Cuando aumenta la complejidad, los enunciados precisos pierden su significado y los enunciados útiles pierden su precisión*".

La lógica difusa (también llamada lógica borrosa) permite resolver problemas de tal forma que, dado un conjunto de variables de entrada (espacio de entrada) se obtenga otro conjunto de variables, en este caso de salida (espacio de salida), atendiendo a criterios de significado (y no de precisión).

Para aplicar la lógica difusa a las técnicas PCA se emplea el *Fuzzy clustering* o agrupamiento difuso (Sârbu, C. y Pop, H., 2005 [22]) que es una herramienta importante en la identificación de la estructura de los datos. En general, un algoritmo de agrupamiento difuso con función objetivo puede formularse de la siguiente forma: Digamos que se tiene $X = \{x^1, ..., x^n\} \subset R^p$ como los datos de entrada organizado en vectores donde n es el número de medidas y p es el número de variables originales, $x_k^j = [x_1^j, x_2^j, ..., x_p^j]^T$ y $L = (L_1, L_2, ..., L^s)$ es una s-tupla de prototipos que cada cual caracteriza uno de los s clusters o grupos componiendo la subestructura de grupos de los datos de entrada; una partición de X en s agrupaciones difusas se consigue al minimizar la función objetivo:

$$J(P, L) = \sum_{i=1}^{s} \sum_{j=1}^{n} (A_i(x^j))^2 d^2(x^j, L^i) \tag{5}$$

Donde $P = \{A_1, ...A_s\}$ es la partición difusa. $A_i(x^j) \in [0, 1]$ representa el grado de pertenencia del punto x^j al grupo A_i. $d(x^j, L^i)$ es la distancia del punto x^j al prototipo del grupo A_i definida por la norma de la distancia euclidea:

$$(x^j, L^i) = \|x^j - L^i\| = [\sum_{k=1}^{p} (x_k^j - L_k^i)^2]^{1/2} \tag{6}$$

El conjunto difuso óptimo se determinará empleando un método iterativo donde J es minimizado respecto a A y a L de forma sucesiva.

Si se emplean clusters lineales mediante el algoritmo fuzzy m-means es posible caracterizar los conjuntos difusos de con prototipos lineales, denotado $L(u, v)$ donde v es el centro de la clase y u, con $\|u\| = 1$, es la dirección principal; se puede llegar a esta generalización difusa de la matriz de covarianza clásica:

$$C_{kl} = \frac{\sum_{j=1}^{n} [A_i(x^j)]^2 (x_{jk} - \bar{x}_k)(x_{jl} - \bar{x}_l)}{\sum_{j=1}^{n} [A_i(x^j)]^2} \tag{7}$$

De esta manera se relacionan la lógica difusa con las técnicas de PCA, pues, empleando esta matriz difusa de covarianza en el cálculo de PCA en vez de la matriz clásica, se consigue una *"fuzzificación"* del algoritmo.

2.3. Algoritmos Fuzzy PCA

Las técnicas PCA también tienen aspectos negativos, y el más importante es la sensibilidad a outliers (valores fuera de rango) y missing values (datos perdidos) (Luukka, P., 2011 [14]). Se han propuesto múltiples soluciones para este problema, pero este trabajo se centra en las modificaciones que emplean lógica difusa (Fuzzy logic).

Se han escogido los siguientes algoritmos por ser relativamente recientes (1999-2011) y por tratarse de algoritmos que introducen componentes difusos o robustos en ellos:

- Robust Fuzzy PCA (RFpca) del artículo "RKF-PCA: Robust kernel fuzzy PCA", Heo et al., 2009 [6]:

Algorithm 2 RFpca

1: Inicializar el vector de pesos $U_0 = [u_1; \ldots ; u_N] = [1, 1, \ldots, 1]$,
el contador $t = 0$
y la matriz de componentes principales W_0 empleando PCA.
2: **while** $\underset{i=1,\ldots,N}{max} |u_i^{t-1} - u_i^t < \varepsilon$ o $t \geq t_{max}|$ **do**
3: $t = t + 1$.
4: Calcular la media (μ_R) y la covarianza (C_R) empleando para ello:
$\mu = \frac{\sum_{i=1}^{N} u_i u_i}{\sum_{i=1}^{N} u_i}$ y
$C_R = \frac{1}{N} \sum_{i=1}^{N} u_i^2 (x_i - \mu_R)(x_i - \mu_R)^T$.
5: Calcular la matriz de componentes principales W utilizando C_R.
6: Calcular un nuevo vector de pesos usando:
$f_1 = exp(-\frac{e(x_i)}{\sigma^2}) = exp(-\frac{\|x_i - WW^T x_i\|^2}{\sigma^2}) = u_i$.
7: **end while**

Los parámetros de este algoritmo son:

- t_{max} (número de iteraciones máximas).
- k (número de componentes principales a utilizar).
- ε (margen de error).
- σ (parámetro libre).

- Fuzzy PCA (fpca) del artículo "Principal component analysis versus fuzzy principal component analysis: A case study: the quality of danube water (1985-1996)", Sârbu, C. y Pop, H., 2005 [22]:

Algorithm 3 fpca

1: Inicializar $\alpha = 0$ y $l = 0$.
2: **while** $\alpha \geq 1$ **do**
3: Minimizar:
$J(A, L, \alpha) = \sum_{j=1}^{n} [A(x^j)]^2 d^2(x^j, L) + \sum_{j=1}^{n} [\bar{A}(x^j)]^2 \frac{\alpha}{1-\alpha}$.
4: Calcular los autovalores.
5: $\alpha = \alpha + paso$.
6: **end while**
7: Normalizar los valores de $X_1, X_2, \ldots, X_p$ para que tengan media 0 y desviación 1.
8: **while** $\underset{i=1,\ldots,N}{max} |A^{l+1} - A^l| < \varepsilon$ **do**
9: Calcular la matriz de covarianza C.
10: Calcular los autovalores y sus correspondientes autovectores.
11: Determinar el nuevo grupo difuso A^{l+1} empleando:
$A(x^j) = \frac{\frac{\alpha}{(1-\alpha)}}{[\frac{\alpha}{(1-\alpha)} + d^2(x^j, L)]}$.
12: $l = l + 1$.
13: **end while**
14: Recalcular la matriz de covarianza C y obtener los autovalores y autovectores.

Los parámetros del algoritmo fpca son:

- $paso$ (paso para iterar α).
- ε (margen de error).
- p (número de componentes principales).

- Fuzzy PCA (fpcapop) del artículo "Principal Components Analysis based on a fuzzy sets approach", Pop, H., 2005 [19].
  Este algoritmo consta de dos algoritmos a su vez:

Algorithm 4 Determine_Fuzzy_Memberships(α)

Inicializar $A(x) = 1$ para todos los x.
while $\underset{i=1,\ldots,N}{max} |A^{l+1} - A^l| < \varepsilon$ **do**
  Determinar $L(u, v)$ con u siendo el mayor de los autovalores obtenidos de la matriz C de covarianza y v determinado por la siguiente ecuación: $v = \frac{\sum_{j=1}^{n} A(x^j)^m x^j}{\sum_{j=1}^{n} A(x^j)^m}$.
  Determinar los nuevos grados de pertenencia $A(x^j)$ empleando: $A(x^j) = \frac{\frac{\alpha}{1-\alpha}}{\frac{\alpha}{1-\alpha} + (d^2(x^j, L))^{\frac{1}{m-1}}}$.
end while

Algorithm 5 Determine_Best_Alpha

Inicializar paso = (valor que proceda) $\alpha_0 = 0$ y $\lambda_0 = 0$.
α = paso.
while $\alpha \geq 1$ **do**
  Llamar a Determine_Fuzzy_Memberships(α).
  Calcular la matriz de covarianza C y los autovalores λ.
  if $\lambda \geq \lambda_0$ **then**
    $\lambda_0 = \lambda$.
    $\alpha_0 = \alpha$.
  end if
  $\alpha = \alpha$+paso.
end while
return α_0

Algorithm 6 fpcapop

1: Llamar a Determine_Best_Alpha().
2: Llamar a Determine_Fuzzy_Memberships(α_0).
3: Calcular la matriz de covarianza C y los autovalores y autovectores.

Los parámetros de este algoritmo son:

- *paso* (paso para la iteración de α).
- m (índice fuzzy).
- ε (margen de error).

- Fuzzy PCA (fuzzypca) del artículo "Robust fuzzy principal component analysis (FPCA). A comparative study concerning interaction of carbon-hydrogen bonds with molybdenum-oxo bonds", Cundari, T. R.; Sârbu, C. y Pop, H., 2005 [4].

Algorithm 7 fuzzypca

1: Inicializar $l = 0$ y $\alpha = 0$.
2: **while** $\alpha \geq 1$ **do**
3: Minimizar:
$J(A, L, \alpha) = \sum_{j=1}^{n} [A(x^j)]^2 d^2(x^j, L) + \sum_{j=1}^{n} [\bar{A}(x^j)]^2 \frac{\alpha}{1-\alpha}$.
4: Calcular la matriz de covarianza C.
5: Calcular los autovalores.
6: $\alpha = \alpha + paso$.
7: **end while**
8: Normalizar los valores de $X_1, X_2, \ldots, X_p$ para que dispongan de media 0 y desviación 1.
9: **while** $\underset{i=1,\ldots,N}{max} |A^{l+1} - A^l| < \varepsilon$ **do**
10: Calcular la matriz de covarianza C.
11: Calcular los autovalores y sus correspondientes autovectores.
12: Determinar el nuevo grupo difuso empleando:
$A(x^j) = \frac{\frac{\alpha}{1-\alpha}}{[\frac{\alpha}{1-\alpha}] + d^2(x^j, L)}$.
13: $l = l + 1$.
14: **end while**
15: Recalcular la matriz de covarianza C y obtener los autovalores y autovectores.

Cuyos parámetros son:

- *paso* (paso para iterar α).
- ε (margen de error).

- Tres algoritmos Fuzzy Robust PCA (FRPCA1, FRPCA2 y FRPCA3) del artículo "Robust algorithms for principal component analysis", Yang, T. y Wang, S., 1999 [28].

Algorithm 8 FRPCA1.

1: Inicializar $t = 1$,
límite de iteración T,
coeficiente de aprendizaje $\alpha_0 \in (0, 1]$,
umbral suave η a un valor positivo pequeño
y de forma aleatoria el peso w.
2: **while** $t < T$ **do**
3: Calcular: $\alpha_t = \alpha_0(1 - \frac{t}{T})$.
4: $i = 1$, $\sigma = 0$.
5: **while** $i < n$ **do**
6: Calcular: $y = w^T x_i$,
$u = yw$ y
$v = w^T u$.
7: Actualizar el peso: $w^{new} = w^{old} + \alpha_T \beta(x_i)[y(x_i - u) + (y - v)x_i]$.
8: Actualizar el contador temporal: $\sigma = \sigma + e_1(x_i)$.
9: $i = i + 1$.
10: **end while**
11: Calcular: $\eta = (\frac{\sigma}{n})$.
12: **end while**

El algoritmo FRPCA2 sólo difiere del algoritmo FRPCA1 en los pasos 7 y 8, los cuales se deben sustituir por los siguientes respectivamente:

7. Actualizar el peso: $w^{new} = w^{old} + \alpha_T \beta(x_i)(x_i y - \frac{w}{w^T w} y^2)$.

8. Actualizar el contador temporal: $\sigma = \sigma + e_2(x_i)$.

Por su parte, los pasos 7 y 8 del algoritmo FRPCA3 son los siguientes:

7. Actualizar el peso: $w^{new} = w^{old} + \alpha_T \beta(x_i)(x_i y - w y^2)$.

8. Actualizar el contador temporal: $\sigma = \sigma + e(x_i)$.

Los parámetros de estos tres algoritmos son:

- T (límite de iteración).
- η (influencia constante de los datos en el cluster de ruido)(umbral).
- α_0 (valor inicial del ratio de aprendizaje).
- m (exponente del peso).

- New nonlinear Fuzzy Robust PCA (nnFRPCA3) del artículo "A new nonlinear fuzzy robust PCA algorithm and similarity classifier in classification of medical data sets", Luukka, P., 2011 [14].

  Este algoritmo es igual a FRPCA1 salvo en los pasos 7 y 8, los cuales son los siguientes:

  7. Calcular $g(y)$,
     $F = \frac{d}{dy}(g(y))$,
     $e_3(x_i) = x_i - w^{old} g(y)$ y
     actualizar el peso: $w^{new} = w^{old} + \alpha_T \beta(x_i)(x_i e_3(x_i)^T w^{old} F + e_3(x_i) g(y))$.

  8. Actualizar el contador temporal: $\sigma = \sigma + e_3(x_i)$.

  Por su parte tiene los mismos parámetros que los algoritmos FRPCA1, 2 y 3.

3. Extracción de variables/características basado en Fuzzy PCA

A la hora de lidiar con datos de gran tamaño se nos presenta una duda importante:

¿Qué variables debemos emplear?

Teniendo en cuenta que se quiere reducir en la medida de lo posible la dimensionalidad del espacio de entrada, lo cual agiliza el proceso de cálculo entero, se nos presentan dos posibles opciones: Seleccionar las variables a emplear o extraer variables nuevas basadas en las originales.

Este trabajo se centra en la extracción de variables, que por medio de PCA se van a crear las nuevas variables basadas en las originales (Componentes Principales) como ya se ha explicado. La extracción de variables es una solución a la pregunta realizada anteriormente, puesto que estas nuevas variables tienen asociada una relevancia, por lo que a la hora de escoger las variables a emplear, se tomarán las más relevantes. Por otra parte, esta técnica tiene un problema asociado, y es que las nuevas variables pierden el significado de las variables originales, por lo que no son fácilmente interpretables.

Llegados a este punto se nos plantea otra duda:

¿Cuantas componentes principales se deben escoger?

Existen varios criterios para resolver este problema, como por ejemplo el *"gap"*, que consiste en representar gráficamente los valores de las componentes principales y encontrar un cambio significativo en sus valores. Entonces se escogerían todas las Componentes Principales que se encontraran a la izquierda de este *"gap"*. Puede presentarse el problema de no encontrarse de forma clara este salto, o que se hayan escogido una cantidad muy elevada de Componentes Principales debido a que el salto se encontraba muy desplazado a la derecha, por lo que no se produciría una reducción considerable de la dimensionalidad.

Otro criterio, que es el que se va a emplear en este trabajo, es el basado en el porcentaje de la variabilidad. En este criterio se decide el porcentaje de variabilidad a emplear, por ejemplo del 60 %, y se escogen de forma ordenada las componentes principales hasta alcanzar esta cota de variabilidad. Se nos presenta de esta forma otro problema: ¿Qué porcentaje de variabilidad debemos escoger?. Este problema se ha solucionado en este trabajo seleccionando un rango de variabilidades como se podrá ver más adelante.

3.1. Metodología Experimental

En esta sección se describe la metodología que se emplea en esta parte del trabajo a la hora de realizar los cálculos necesarios para la obtención de los resultados:

En primer lugar se han escogido los datos que se pueden encontrar en la sección 3.3, los cuales se han sometido a un preprocesamiento, el cual consta de una normalización de los datos en media 0 y desviación 1 para que todas las variables dispongan de la misma relevancia a la hora del cálculo, puesto que si una variable de orden elevado se mantiene constante mientras que otra de orden menor varía en gran medida, no tienen la misma relevancia en los resultados, pero en el cálculo se impondría la variable de mayor orden pese a permanecer constante.

En segundo lugar, se han empleado los algoritmos escogidos para el tratamiento de los datos normalizados de cada conjunto de datos con el fin de obtener de estos los resultados a comparar. Los algoritmos difusos se explican en la sección 2.3. Como algoritmo de control se ha empleado la técnica PCA clásica.

En tercer lugar, se han aplicado los criterios de comparación descritos en la sección 3.2. Una vez aplicados estos criterios se han recogido en tablas donde se muestran los resultados obtenidos disponiendo los resultados de cada criterio para cada algoritmo y conjunto de datos en una misma tabla.

En último lugar se ha realizado una tabla resumen de los resultados, a partir de la cual se extraen las conclusiones de los resultados obtenidos.

3.2. Criterios de análisis

Los criterios de análisis que se van a emplear en esta parte del trabajo son las siguientes:

- Componentes Principales para un Umbral de Variabilidad.
- Robustez: Sensibilidad a Outliers.
- Recursos Computacionales.

A continuación se explican en detalle cada uno de ellos:

3.2.1. Componentes Principales para un Umbral de Variabilidad

Este criterio de comparación se basa en medir la cantidad de componentes principales que necesita el subespacio para explicar un porcentaje de variabilidad para cada algoritmo y cada conjunto de datos.

El Algoritmo que necesite de un número menor de componentes principales para explicar el mismo nivel de variabilidad tiene una mayor capacidad de compresión de los datos.

El número de componentes principales necesario para un cierto nivel de variabilidad se puede calcular de forma sencilla una vez se hayan obtenido los autovalores. Para obtener la proporción de varianza contenida en cada autovalor se tiene que calcular la proporción de cada uno respecto a la suma del total (ecuación 8). Con esto se tendrá la proporción de cada uno de variabilidad de cada componente principal. Empleando a continuación la ecuación 9 se obtiene la proporción acumulativa.

$$\%\lambda_i = \frac{\lambda_i}{\sum_{i=1}^{1} \lambda_i} \tag{8}$$

$$\text{Proporción de varianza acumulativa} = \sum_{i}^{p} \%\lambda_i \tag{9}$$

Para una lectura de la varianza más sencilla se puede expresar en % o en tanto por uno. Este criterio se utiliza en los artículos de Sârbu, C., (2005) [22] y Cundari, R., (2002) [4].

Umbrales de variabilidad	60 %	65 %	70 %	75 %	80 %	85 %	90 %	95 %

Tabla 1: Umbrales de variabilidad.

Los umbrales de variabilidad escogidos se pueden ver en la Tabla 1:

El número de componentes principales para explicar el umbral de variabilidad elegido es aquel en el que su proporción de variabilidad acumulativa iguala al valor del umbral.

3.2.2. Robustez: Sensibilidad a Outliers

Este criterio viene del artículo de Heo, G., (2009) [6] que define el error del autosistema,o lo que es lo mismo dos autovectores correspondientes a la misma componente principal de los datos sin ruido y los datos con ruido.

Heo, G., (2009)[6] define la siguiente fórmula (Ecuaciones 10 - 12) para obtener el error entre dos autovectores:

$$E = \sum_{i=1}^{N_1} \lambda_i^C \theta(w_C^i, w_N^i) \tag{10}$$

Donde N_1 es el número de datos sin ruido, λ_i^C es el autovalor i-ésimo de los datos sin ruido, w_C^i y w_N^i son los autovectores i-ésimos de los datos sin y con ruido respectivamente y $\theta(w_C^i, w_N^i)$ representa el ángulo formado por los dos autovectores que puede calcularse de dos formas diferentes:

$$\cos(\theta(w_C^i, w_N^i)) = w_C^a \cdot w_N^b = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \alpha_i^a \alpha_j^b u_j k(x_i, x_j) \tag{11}$$

$$\theta(w_C^i, w_N^i) = \cos^{-1}(\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \alpha_i^a \alpha_j^b u_j k(x_i, x_j)) \tag{12}$$

Donde N_2 es el número de datos con ruido.

Este error es más pequeño cuanto mejor tolere el ruido el algoritmo.

En el artículo de Heo, G., (2009) [6] se utiliza una proporción del 10 % de outliers, por lo que se han añadido outliers como se muestra en la Tabla 2 cumpliendo con esa proporción.

Data Set	Outliers
Airfoil Self-Noise	167
Breast Tissue	12
Ecoli	37
Energy Efficiency	85
Iris	17
Istambul Stock Exchange	60
Letter Recognition	2222
Blood Transfusion Service Center	83
Wholesale Customers	49

Tabla 2: Número de Outliers introducidos en cada data set.

Los outliers se han creado de forma aleatoria, la semilla tiene el valor del instante de tiempo en que se crean los outliers, con un rango del 95-percentil y un desplazamiento positivo del 99-percentil. En caso de que el 95-percentil sea menor que la unidad, se toma como rango la unidad.

3.2.3. Recursos Computacionales

Este criterio mide la cantidad de recursos que utiliza el ordenador mediante el tiempo de ejecución del algoritmo. A mayor tiempo de ejecución mayor cantidad de recursos consume el algoritmo.

Este criterio se utiliza en el artículo de Heo, G., (2009) [6].

Se ha operado con un ordenador de las siguientes características:

- Modelo: ASPIRE 5755G.
- Procesador: Intel Core i5-2450M 2.5GHz. Con turbo a 3.1GHz.
- Tarjeta Gráfica: NVIDIA GeForce GT 630M con 2GB dedicados.
- Memoria: 8GB DDR3.
- Disco duro: 750GB HDD.

3.3. Datasets empleados

Para la comparación de los distintos algoritmos se han utilizado los siguientes datos, todos obtenidos del UCI Repository of Machine Learning Database [12].

- Airfoil Self-Noise: Variables: 6. Muestras: 1503.
- Breast Tissue: Variables: 9. Muestras: 106.
- Energy Efficiency: Variables: 10. Muestras: 769.
- Ecoli: Variables: 7. Muestras: 336.
- Iris: Variables: 4. Muestras: 150.
- Istambul Stock Exchange: Variables: 9. Muestras: 539.
- Letter Recognition: Variables: 16. Muestras: 20000.
- Blood Transfusion Service Center: Variables: 5. Muestras: 748
- Wholesale Customers: Variables: 8. Muestras: 440.

En las Figuras 2 - 9 se representan los diagramas de caja de los datos arriba mencionados.

Se han tomado estos conjuntos de datos por su diversidad en cuanto a número de variables y observaciones y porque son conjuntos de datos ampliamente empleados para clasificación y regresión.

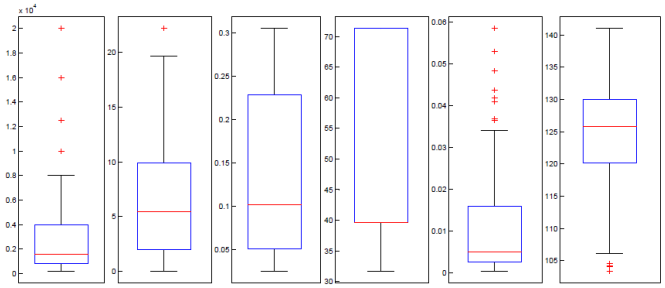

Figura 2: Diagrama de caja de los datos Airfoil Self-Noise.

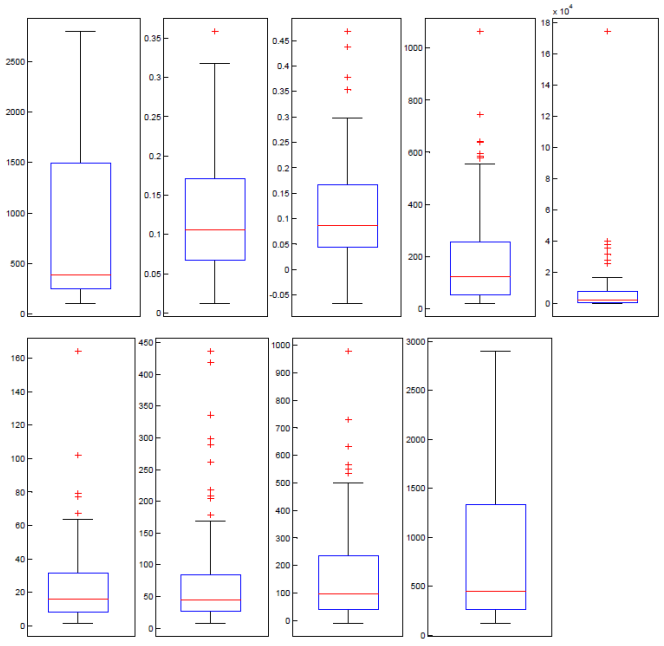

Figura 3: Diagrama de caja de los datos Breast Tissue.

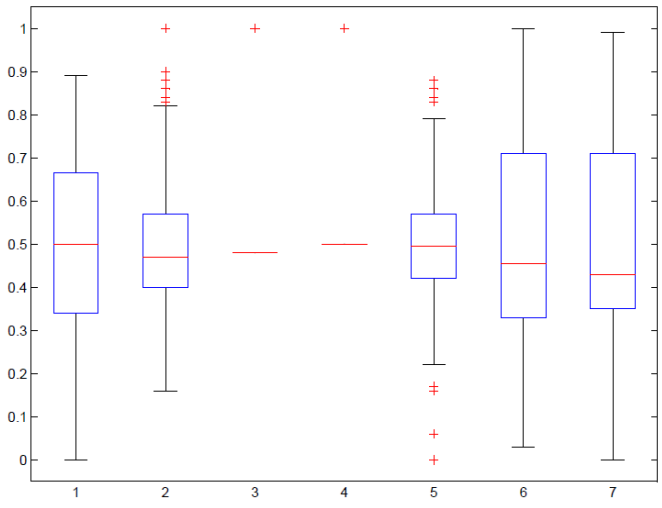

Figura 4: Diagrama de caja de los datos Ecoli.

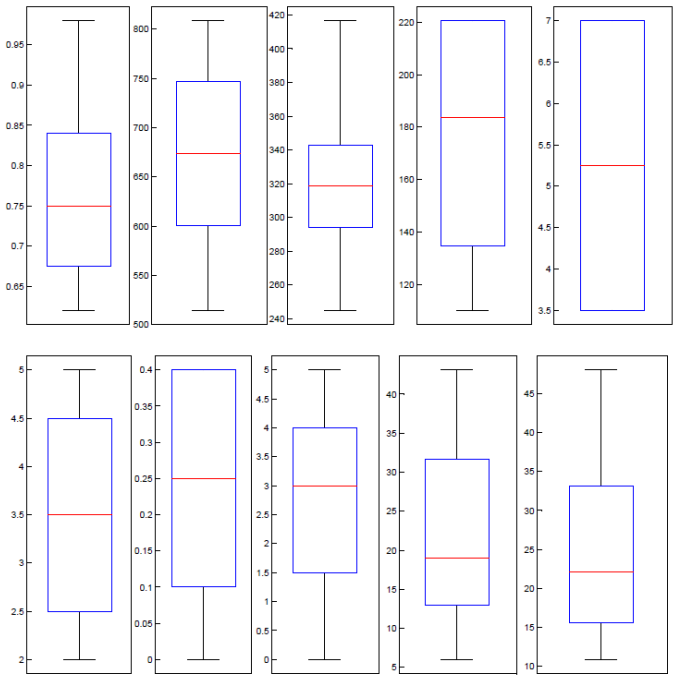

Figura 5: Diagrama de caja de los datos Energy Efficiency.

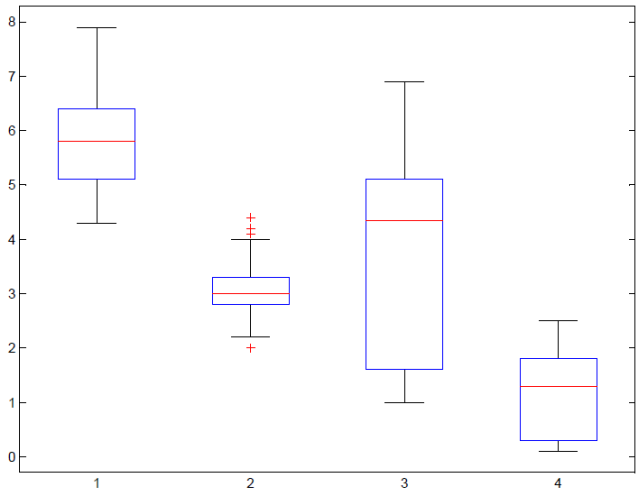

Figura 6: Diagrama de caja de los datos Iris.

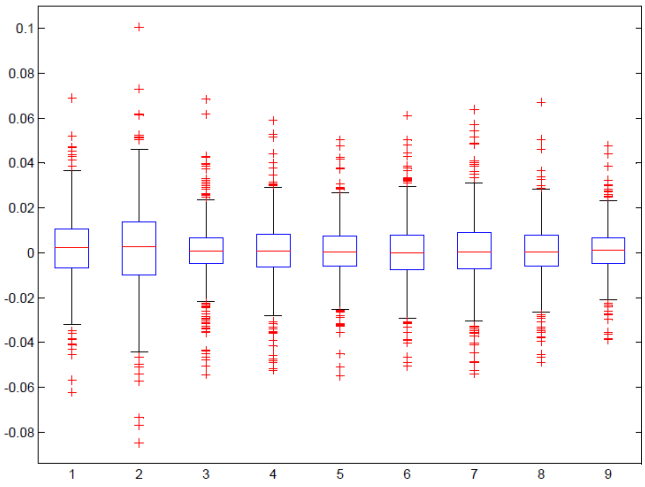

Figura 7: Diagrama de caja de los datos Istambul Stock Exchange.

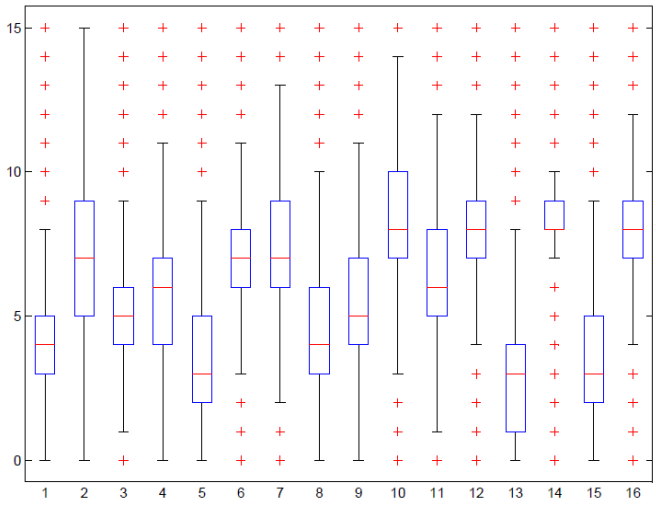

Figura 8: Diagrama de caja de los datos Letter Recognition.

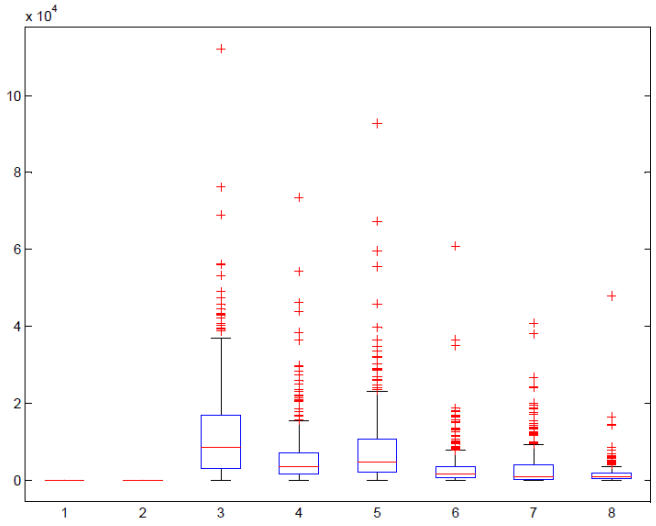

Figura 9: Diagrama de caja de los datos Wholesale Customers.

4. Análisis de los Resultados Obtenidos

Los resultados de la comparación de algoritmos se muestran en las siguientes tablas:

Algoritmos		Cantidad de Varianza						Num Componentes principales								Robustez	R Comp
		1CP	2CP	3CP	4CP	5CP	6CP	60 %	65 %	70 %	75 %	80 %	85 %	90 %	95 %	Error	T Comp
	VP	2.1162	1.4454	1.1133	0.9148	0.2531	0.1572										
PCA	prop	0.3527	0.2409	0.1855	0.1525	0.0422	0.0262	3	3	3	3	4	4	4	5	6.4429	0.8050
	acum	0.3527	0.5936	0.7791	0.9316	0.9738	1.0000										
	VP	2.1148	1.4444	1.1125	0.9142	0.2529	0.1571										
RFpca	prop	0.3527	0.2409	0.1855	0.1525	0.0422	0.0262	3	3	3	3	4	4	4	5	10.6578	0.7420
	acum	0.3527	0.5936	0.7791	0.9316	0.9738	1.0000										
	VP	4.3493	0.3566	0.2925	0.1798	0.0737	0.0210										
fpca	prop	0.8248	0.0676	0.0555	0.0341	0.0140	0.0040	1	1	1	1	1	2	3	4	7.0075	9.4700
	acum	0.8248	0.8925	0.9479	0.9820	0.9960	1.0000										
	VP	3.4271	0.4861	0.4329	0.2665	0.0938	0.0300										
fpcapop	prop	0.7236	0.1026	0.0914	0.0563	0.0198	0.0063	1	1	1	2	2	3	3	4	6.7976	5.1400
	acum	0.7236	0.8262	0.9176	0.9739	0.9937	1.0000										
	VP	4.2274	0.3696	0.3067	0.1882	0.0759	0.0218										
fuzzypca	prop	0.8146	0.0712	0.0591	0.0363	0.0146	0.0042	1	1	1	1	1	2	3	4	6.5916	6.9910
	acum	0.8146	0.8858	0.9449	0.9812	0.9958	1.0000										
	VP	1.9663	1.4063	1.1476	1.0092	0.2466	0.2241										
FRPCA1	prop	0.3277	0.2344	0.1913	0.1682	0.0411	0.0373	3	3	3	3	4	4	4	5	8.8452	211.4340
	acum	0.3277	0.5621	0.7534	0.9216	0.9627	1.0000										
	VP	1.8228	1.4425	1.1368	0.9573	0.3800	0.2606										
FRPCA2	prop	0.3038	0.2404	0.1895	0.1595	0.0633	0.0434	3	3	3	4	4	4	5	5	8.3081	256.0300
	acum	0.3038	0.5442	0.7337	0.8932	0.9566	1.0000										
	VP	1.8228	1.4425	1.1368	0.9573	0.3800	0.2606										
FRPCA3	prop	0.3038	0.2404	0.1895	0.1595	0.0633	0.0434	3	3	3	4	4	4	5	5	6.7579	237.1690
	acum	0.3038	0.5442	0.7337	0.8932	0.9566	1.0000										
	VP	1.7299	1.4333	0.9747	0.7107	0.5906	0.5607										
nnFRPCA3	prop	0.2883	0.2389	0.1625	0.1185	0.0984	0.0934	3	3	4	4	4	5	5	6	10.5879	3.8920
	acum	0.2883	0.5272	0.6897	0.8081	0.9066	1.0000										

Tabla 3: Resultados de la comparación de algoritmos con los datos de Airfoil Self-Noise.

R Comp = Recursos Computacionales y T Comp = tiempo de cómputo.

Algoritmos		Cantidad de Varianza									Num Componentes principales								Robustez	R Comp
		1CP	2CP	3CP	4CP	5CP	6CP	7CP	8CP	9CP	60 %	65 %	70 %	75 %	80 %	85 %	90 %	95 %	Error	T Comp
PCA	VP	5.4631	1.8110	0.7787	0.5059	0.2847	0.0994	0.0531	0.0031	0.0009										
	prop	0.6070	0.2012	0.0865	0.0562	0.0316	0.0110	0.0059	0.0003	0.0001	1	2	2	2	2	3	4	4	3.6262	0.0450
	acum	0.6070	0.8082	0.8948	0.9510	0.9826	0.9937	0.9996	0.9999	1.0000										
RFpca	VP	5.4116	1.7939	0.7714	0.5012	0.2820	0.0984	0.0526	0.0031	0.0009										
	prop	0.6070	0.2012	0.0865	0.0562	0.0316	0.0110	0.0059	0.0003	0.0001	1	2	2	2	2	3	4	4	6.8892	0.0970
	acum	0.6070	0.8082	0.8948	0.9510	0.9826	0.9937	0.9996	0.9999	1.0000										
fpca	VP	5.4105	1.7939	0.7712	0.5011	0.2820	0.0984	0.0526	0.0031	0.0009										
	prop	0.6070	0.2012	0.0865	0.0562	0.0316	0.0110	0.0059	0.0003	0.0001	1	2	2	2	2	3	4	4	6.5304	5.3030
	acum	0.6070	0.8082	0.8948	0.9510	0.9826	0.9937	0.9996	0.9999	1.0000										
fpcapop	VP	5.4116	1.7939	0.7714	0.5012	0.2820	0.0984	0.0526	0.0031	0.0009										
	prop	0.6070	0.2012	0.0865	0.0562	0.0316	0.0110	0.0059	0.0003	0.0001	1	2	2	2	2	3	4	4	18.9817	2.8190
	acum	0.6070	0.8082	0.8948	0.9510	0.9826	0.9937	0.9996	0.9999	1.0000										
fuzzypca	VP	2.0493	0.6202	0.2216	0.0696	0.0282	0.0090	0.0042	0.0002	0.0001										
	prop	0.6825	0.2066	0.0738	0.0232	0.0094	0.0030	0.0014	0.0001	0.0000	1	1	2	2	2	2	3	3	18.8317	2.2360
	acum	0.6825	0.8891	0.9629	0.9861	0.9955	0.9985	0.9999	1.0000	1.0000										
FRPCA1	VP	5.3257	1.7858	0.6054	0.5774	0.4059	0.1889	0.1014	0.0025	0.0012										
	prop	0.5921	0.1986	0.0673	0.0642	0.0451	0.0210	0.0113	0.0003	0.0001	2	2	2	2	3	3	4	5	26.4860	83.4370
	acum	0.5921	0.7907	0.8580	0.9222	0.9673	0.9883	0.9996	0.9999	1.0000										
FRPCA2	VP	5.3340	1.7955	0.6058	0.5739	0.4021	0.1787	0.1005	0.0039	0.0018										
	prop	0.5929	0.1996	0.0673	0.0638	0.0447	0.0199	0.0112	0.0004	0.0002	2	2	2	2	3	3	4	5	5.7582	32.8520
	acum	0.5929	0.7925	0.8598	0.9236	0.9683	0.9882	0.9994	0.9998	1.0000										
FRPCA3	VP	5.3340	1.7955	0.6058	0.5739	0.4021	0.1787	0.1005	0.0052	0.0016										
	prop	0.5928	0.1996	0.0673	0.0638	0.0447	0.0199	0.0112	0.0006	0.0002	2	2	2	2	3	3	4	5	5.5906	23.8540
	acum	0.5928	0.7924	0.8597	0.9235	0.9682	0.9881	0.9992	0.9998	1.0000										
nnFRPCA3	VP	4.8981	1.4641	1.3525	0.5366	0.3785	0.2284	0.1278	0.0107	0.0033										
	prop	0.5442	0.1627	0.1503	0.0596	0.0421	0.0254	0.0142	0.0012	0.0004	2	2	2	3	3	3	4	5	15.4594	2.3400
	acum	0.5442	0.7069	0.8572	0.9168	0.9589	0.9842	0.9984	0.9996	1.0000										

Tabla 4: Resultados de la comparación de algoritmos con los datos de Breast Tissue.

R Comp = Recursos Computacionales y T Comp = tiempo de cómputo.

Algoritmos		Cantidad de Varianza										Num Componentes principales								Robustez	R Comp
		1CP	2CP	3CP	4CP	5CP	6CP	7CP	8CP	9CP	10CP	60 %	65 %	70 %	75 %	80 %	85 %	90 %	95 %	Error	T Comp
PCA	VP	5.2229	1.5334	1.2189	1.0002	0.8048	0.1631	0.0331	0.0194	0.0044	0.0000										
	prop	0.5223	0.1533	0.1219	0.1000	0.0805	0.0163	0.0033	0.0019	0.0004	0.0000	2	2	3	3	4	4	5	5	22.9376	0.0160
	acum	0.5223	0.6756	0.7975	0.8975	0.9780	0.9943	0.9976	0.9996	1.0000	1.0000										
RFpca	VP	5.2161	1.5314	1.2173	0.9989	0.8037	0.1629	0.0330	0.0193	0.0043	0.0000										
	prop	0.5223	0.1533	0.1219	0.1000	0.0805	0.0163	0.0033	0.0019	0.0004	0.0000	2	2	3	3	4	4	5	5	15.3859	0.0620
	acum	0.5223	0.6756	0.7975	0.8975	0.9780	0.9943	0.9976	0.9996	1.0000	1.0000										
fpca	VP	9.0338	0.7488	0.7286	0.6509	0.4934	0.0849	0.0323	0.0166	0.0006	0.0000										
	prop	0.7662	0.0635	0.0618	0.0552	0.0418	0.0072	0.0027	0.0014	0.0000	0.0000	1	1	1	1	2	3	4	5	15.1195	5.5380
	acum	0.7662	0.8297	0.8915	0.9467	0.9886	0.9958	0.9985	1.0000	1.0000	1.0000										
fpcapop	VP	9.0384	0.7491	0.7286	0.6511	0.4934	0.0849	0.0323	0.0166	0.0006	0.0000										
	prop	0.7663	0.0635	0.0618	0.0552	0.0418	0.0072	0.0027	0.0014	0.0000	0.0000	1	1	1	1	2	3	4	5	20.1994	3.8530
	acum	0.7663	0.8298	0.8916	0.9468	0.9886	0.9958	0.9985	1.0000	1.0000	1.0000										
fuzzypca	VP	9.0378	0.7491	0.7287	0.6511	0.4934	0.0849	0.0323	0.0166	0.0006	0.0000										
	prop	0.7663	0.0635	0.0618	0.0552	0.0418	0.0072	0.0027	0.0014	0.0000	0.0000	1	1	1	1	2	3	4	5	20.7595	3.6040
	acum	0.7663	0.8298	0.8916	0.9468	0.9886	0.9958	0.9985	1.0000	1.0000	1.0000										
FRPCA1	VP	5.1481	1.5007	1.2499	1.0668	0.8094	0.1640	0.0343	0.0222	0.0046	0.0000										
	prop	0.5148	0.1501	0.1250	0.1067	0.0809	0.0164	0.0034	0.0022	0.0005	0.0000	2	2	3	3	4	4	5	5	15.5765	200.3650
	acum	0.5148	0.6649	0.7899	0.8966	0.9775	0.9939	0.9973	0.9995	1.0000	1.0000										
FRPCA2	VP	4.8853	1.6021	1.4010	1.0740	0.8061	0.1681	0.0356	0.0226	0.0051	0.0000										
	prop	0.4885	0.1602	0.1401	0.1074	0.0806	0.0168	0.0036	0.0023	0.0005	0.0000	2	3	3	3	4	4	5	5	17.2490	237.4250
	acum	0.4885	0.6487	0.7889	0.8962	0.9769	0.9937	0.9972	0.9995	1.0000	1.0000										
FRPCA3	VP	4.8853	1.6021	1.4010	1.0740	0.8061	0.1681	0.0356	0.0226	0.0051	0.0000										
	prop	0.4885	0.1602	0.1401	0.1074	0.0806	0.0168	0.0036	0.0023	0.0005	0.0000	2	3	3	3	4	4	5	5	14.6768	178.5130
	acum	0.4885	0.6487	0.7889	0.8962	0.9769	0.9937	0.9972	0.9995	1.0000	1.0000										
nnFRPCA3	VP	3.0496	2.1272	1.1763	1.0862	1.0847	0.7836	0.5561	0.0922	0.0441	0.0000										
	prop	0.3050	0.2127	0.1176	0.1086	0.1085	0.0784	0.0556	0.0092	0.0044	0.0000	3	4	4	5	5	5	6	7	13.6307	1.4710
	acum	0.3050	0.5177	0.6353	0.7439	0.8524	0.9308	0.9864	0.9956	1.0000	1.0000										

Tabla 5: Resultados de la comparación de algoritmos con los datos de Energy efficiency.

D.C.P. = Distribución de las componentes principales, Std = desviación estandar, R Comp = Recursos Computacionales y T Comp = tiempo de cómputo.

Algoritmos		Cantidad de Varianza							Num Componentes principales								Robustez	R Comp
		1CP	2CP	3CP	4CP	5CP	6CP	7CP	60 %	65 %	70 %	75 %	80 %	85 %	90 %	95 %	Error	T Comp
PCA	VP	2.2056	1.4612	1.2015	0.8572	0.6694	0.4787	0.1264										
	prop	0.3151	0.2087	0.1716	0.1225	0.0956	0.0684	0.0181	3	3	4	4	4	5	5	6	6.9354	0.0940
	acum	0.3151	0.5238	0.6955	0.8179	0.9136	0.9819	1.0000										
RFpca	VP	2.1991	1.4568	1.1979	0.8546	0.6675	0.4772	0.1260										
	prop	0.3151	0.2087	0.1716	0.1225	0.0956	0.0684	0.0181	3	3	4	4	4	5	5	6	9.6182	0.0310
	acum	0.3151	0.5238	0.6955	0.8179	0.9136	0.9819	1.0000										
fpca	VP	5.8734	0.4176	0.3299	0.2812	0.0286	0.0227	0.0007										
	prop	0.8446	0.0600	0.0474	0.0404	0.0041	0.0033	0.0001	1	1	1	1	1	2	2	3	14.7933	2.5580
	acum	0.8446	0.9046	0.9521	0.9925	0.9966	0.9999	1.0000										
fpcapop	VP	5.4724	0.4397	0.3328	0.2945	0.0298	0.0235	0.0007										
	prop	0.8300	0.0667	0.0505	0.0447	0.0045	0.0036	0.0001	1	1	1	1	1	2	3	4	6.4679	1.5450
	acum	0.8300	0.8967	0.9471	0.9918	0.9963	0.9999	1.0000										
fuzzypca	VP	5.4726	0.4397	0.3328	0.2945	0.0298	0.0235	0.0007										
	prop	0.8300	0.0667	0.0505	0.0447	0.0045	0.0036	0.0001	1	1	1	1	1	2	3	4	14.1787	1.4510
	acum	0.8300	0.8967	0.9471	0.9918	0.9963	0.9999	1.0000										
FRPCA1	VP	2.1404	1.3228	1.0064	1.0025	0.8525	0.5279	0.1477										
	prop	0.3058	0.1890	0.1438	0.1432	0.1218	0.0754	0.0211	3	4	4	4	5	5	5	6	14.4762	49.9670
	acum	0.3058	0.4947	0.6385	0.7817	0.9035	0.9789	1.0000										
FRPCA2	VP	2.0992	1.3216	1.0049	1.0001	0.8368	0.5729	0.1644										
	prop	0.2999	0.1888	0.1436	0.1429	0.1195	0.0818	0.0235	3	4	4	4	5	5	6	6	8.8060	65.2580
	acum	0.2999	0.4887	0.6322	0.7751	0.8947	0.9765	1.0000										
FRPCA3	VP	2.0992	1.3216	1.0049	1.0001	0.8368	0.5729	0.1644										
	prop	0.2999	0.1888	0.1436	0.1429	0.1195	0.0818	0.0235	3	4	4	4	5	5	6	6	7.2824	53.3480
	acum	0.2999	0.4887	0.6322	0.7751	0.8947	0.9765	1.0000										
nnFRPCA3	VP	2.1600	1.2104	1.0549	0.9644	0.9640	0.5134	0.1330										
	prop	0.3086	0.1729	0.1507	0.1378	0.1377	0.0733	0.0190	3	4	4	4	5	5	5	6	11.5077	1.7860
	acum	0.3086	0.4815	0.6322	0.7699	0.9077	0.9810	1.0000										

Tabla 6: Resultados de la comparación de algoritmos con los datos de Ecoli.

D.C.P. = Distribución de las componentes principales, Std = desviación estandar, R Comp = Recursos Computacionales y T Comp = tiempo de cómputo.

Algoritmos		Cantidad de Varianza				Num Componentes principales								Robustez	R Comp
		1CP	2CP	3CP	4CP	60%	65%	70%	75%	80%	85%	90%	95%	Error	T Comp
	VP	2.9108	0.9212	0.1474	0.0206										
PCA	prop	0.7277	0.2303	0.0368	0.0052	1	1	1	2	2	2	2	2	0.8930	0.0000
	acum	0.7277	0.9580	0.9948	1.0000										
	VP	2.8914	0.9151	0.1464	0.0205										
RFpca	prop	0.7277	0.2303	0.0368	0.0052	1	1	1	2	2	2	2	2	2.7724	0.0470
	acum	0.7277	0.9580	0.9948	1.0000										
	VP	4.8612	0.3426	0.1783	0.0265										
fpca	prop	0.8988	0.0633	0.0330	0.0049	1	1	1	1	1	1	2	2	3.3447	1.0610
	acum	0.8988	0.9621	0.9951	1.0000										
	VP	4.8629	0.3426	0.1782	0.0265										
fpcapop	prop	0.8988	0.0633	0.0329	0.0049	1	1	1	1	1	1	2	2	13.5784	0.4680
	acum	0.8988	0.9622	0.9951	1.0000										
	VP	4.8650	0.3427	0.1782	0.0265										
fuzzypca	prop	0.8989	0.0633	0.0329	0.0049	1	1	1	1	1	1	2	2	13.0885	0.5300
	acum	0.8989	0.9622	0.9951	1.0000										
	VP	2.8914	0.9401	0.1474	0.0211										
FRPCA1	prop	0.7229	0.2350	0.0368	0.0053	1	1	1	2	2	2	2	2	8.5402	12.2780
	acum	0.7229	0.9579	0.9947	1.0000										
	VP	2.8836	0.9475	0.1475	0.0214										
FRPCA2	prop	0.7209	0.2369	0.0369	0.0053	1	1	1	2	2	2	2	2	8.9117	16.3800
	acum	0.7209	0.9578	0.9947	1.0000										
	VP	2.8836	0.9475	0.1475	0.0214										
FRPCA3	prop	0.7209	0.2369	0.0369	0.0053	1	1	1	2	2	2	2	2	8.9117	13.1820
	acum	0.7209	0.9578	0.9947	1.0000										
	VP	2.4604	0.9698	0.4845	0.0853										
nnFRPCA3	prop	0.6151	0.2425	0.1211	0.0213	1	2	2	2	2	2	3	3	2.5294	0.3270
	acum	0.6151	0.8576	0.9787	1.0000										

Tabla 7: Resultados de la comparación de algoritmos con los datos de Iris.

D.C.P. = Distribución de las componentes principales, Std = desviación estandar, R Comp = Recursos Computacionales y T Comp = tiempo de cómputo.

Algoritmos		Cantidad de Varianza									Num Componentes principales								Robustez	R Comp
		1CP	2CP	3CP	4CP	5CP	6CP	7CP	8CP	9CP	60 %	65 %	70 %	75 %	80 %	85 %	90 %	95 %	Error	T Comp
PCA	VP	5.7867	1.1354	0.8334	0.5747	0.3248	0.1446	0.1294	0.0425	0.0285										
	prop	0.6430	0.1262	0.0926	0.0639	0.0361	0.0161	0.0144	0.0047	0.0032	1	2	2	2	3	3	4	5	3.1631	0.0620
	acum	0.6430	0.7691	0.8617	0.9256	0.9617	0.9777	0.9921	0.9968	1.0000										
RFpca	VP	5.7759	1.1332	0.8318	0.5736	0.3242	0.1443	0.1292	0.0424	0.0284										
	prop	0.6430	0.1262	0.0926	0.0639	0.0361	0.0161	0.0144	0.0047	0.0032	1	2	2	2	3	3	4	5	4.1958	0.0160
	acum	0.6430	0.7691	0.8617	0.9256	0.9617	0.9777	0.9921	0.9968	1.0000										
fpca	VP	5.7742	1.1328	0.8321	0.5738	0.3238	0.1443	0.1291	0.0424	0.0284										
	prop	0.6429	0.1261	0.0927	0.0639	0.0361	0.0161	0.0144	0.0047	0.0032	1	2	2	2	3	3	4	5	23.6823	7.4100
	acum	0.6429	0.7691	0.8617	0.9256	0.9617	0.9777	0.9921	0.9968	1.0000										
fpcapop	VP	5.7759	1.1332	0.8318	0.5736	0.3242	0.1443	0.1292	0.0424	0.0284										
	prop	0.6430	0.1262	0.0926	0.0639	0.0361	0.0161	0.0144	0.0047	0.0032	1	2	2	2	3	3	4	5	21.5093	3.7750
	acum	0.6430	0.7691	0.8617	0.9256	0.9617	0.9777	0.9921	0.9968	1.0000										
fuzzypca	VP	5.7759	1.1332	0.8318	0.5736	0.3242	0.1443	0.1292	0.0424	0.0284										
	prop	0.6430	0.1262	0.0926	0.0639	0.0361	0.0161	0.0144	0.0047	0.0032	1	2	2	2	3	3	4	5	23.6006	3.5720
	acum	0.6430	0.7691	0.8617	0.9256	0.9617	0.9777	0.9921	0.9968	1.0000										
FRPCA1	VP	5.7683	1.1298	0.8205	0.5909	0.3225	0.1544	0.1400	0.0433	0.0303										
	prop	0.6409	0.1255	0.0912	0.0657	0.0358	0.0172	0.0156	0.0048	0.0034	1	2	2	2	3	3	4	5	22.9777	98.0000
	acum	0.6409	0.7665	0.8576	0.9233	0.9591	0.9763	0.9918	0.9966	1.0000										
FRPCA2	VP	5.7536	1.1288	0.8214	0.5912	0.3238	0.1590	0.1478	0.0441	0.0304										
	prop	0.6393	0.1254	0.0913	0.0657	0.0360	0.0177	0.0164	0.0049	0.0034	1	2	2	2	3	3	4	5	6.2544	123.7700
	acum	0.6393	0.7647	0.8560	0.9217	0.9576	0.9753	0.9917	0.9966	1.0000										
FRPCA3	VP	5.7536	1.1288	0.8214	0.5912	0.3238	0.1590	0.1478	0.0441	0.0304										
	prop	0.6393	0.1254	0.0913	0.0657	0.0360	0.0177	0.0164	0.0049	0.0034	1	2	2	2	3	3	4	5	4.3273	100.7770
	acum	0.6393	0.7647	0.8560	0.9217	0.9576	0.9753	0.9917	0.9966	1.0000										
nnFRPCA3	VP	5.0167	1.5496	0.9580	0.7625	0.3337	0.1469	0.1365	0.0612	0.0350										
	prop	0.5574	0.1722	0.1064	0.0847	0.0371	0.0163	0.0152	0.0068	0.0039	2	2	2	3	3	4	4	5	19.8409	6.4890
	acum	0.5574	0.7296	0.8360	0.9207	0.9578	0.9742	0.9893	0.9961	1.0000										

Tabla 8: Resultados de la comparación de algoritmos con los datos de Istambul Stock Exchange.

D.C.P. = Distribución de las componentes principales, Std = desviación estandar, R Comp = Recursos Computacionales y T Comp = tiempo de cómputo.

Algoritmos		Cantidad de Varianza																Num Componentes principales								Robustez	R Comp
		1CP	2CP	3CP	4CP	5CP	6CP	7CP	8CP	9CP	10CP	11CP	12CP	13CP	14CP	15CP	16CP	60%	65%	70%	75%	80%	85%	90%	95%	Error	T Comp
PCA	VP	4.2954	2.6254	1.7211	1.3691	1.0514	0.9801	0.8893	0.6259	0.5955	0.4919	0.4265	0.2662	0.2539	0.2145	0.1190	0.0749										
	prop	0.2685	0.1641	0.1076	0.0856	0.0657	0.0613	0.0556	0.0391	0.0372	0.0307	0.0267	0.0166	0.0159	0.0134	0.0074	0.0047	4	5	6	6	7	9	10	12	9.3558	0.0310
	acum	0.2685	0.4326	0.5401	0.6257	0.6914	0.7527	0.8082	0.8473	0.8846	0.9153	0.9420	0.9586	0.9745	0.9879	0.9953	1.0000										
RFpca	VP	4.2952	2.6253	1.7210	1.3690	1.0513	0.9800	0.8892	0.6258	0.5954	0.4919	0.4265	0.2662	0.2539	0.2145	0.1190	0.0749										
	prop	0.2685	0.1641	0.1076	0.0856	0.0657	0.0613	0.0556	0.0391	0.0372	0.0307	0.0267	0.0166	0.0159	0.0134	0.0074	0.0047	4	5	6	6	7	9	10	12	27.5158	0.3590
	acum	0.2685	0.4326	0.5401	0.6257	0.6914	0.7527	0.8082	0.8473	0.8846	0.9153	0.9420	0.9586	0.9745	0.9879	0.9953	1.0000										
fpca	VP	1.5935	1.1251	0.8687	0.7342	0.5656	0.5249	0.4721	0.3553	0.2800	0.2644	0.1965	0.1699	0.1521	0.1254	0.0934	0.0560										
	prop	0.2103	0.1485	0.1146	0.0969	0.0746	0.0693	0.0623	0.0469	0.0370	0.0349	0.0259	0.0224	0.0201	0.0166	0.0123	0.0074	5	6	6	7	8	9	11	13	22.8322	326.2280
	acum	0.2103	0.3588	0.4734	0.5703	0.6450	0.7143	0.7766	0.8235	0.8604	0.8953	0.9212	0.9437	0.9637	0.9803	0.9926	1.0000										
fpcapop	VP	4.2952	2.6253	1.7210	1.3690	1.0513	0.9800	0.8892	0.6258	0.5954	0.4919	0.4265	0.2662	0.2539	0.2145	0.1190	0.0749										
	prop	0.2685	0.1641	0.1076	0.0856	0.0657	0.0613	0.0556	0.0391	0.0372	0.0307	0.0267	0.0166	0.0159	0.0134	0.0074	0.0047	4	5	6	6	7	9	10	12	28.0568	115.9860
	acum	0.2685	0.4326	0.5401	0.6257	0.6914	0.7527	0.8082	0.8473	0.8846	0.9153	0.9420	0.9586	0.9745	0.9879	0.9953	1.0000										
fuzzypca	VP	4.2952	2.6253	1.7210	1.3690	1.0513	0.9800	0.8892	0.6258	0.5954	0.4919	0.4265	0.2662	0.2539	0.2145	0.1190	0.0749										
	prop	0.2685	0.1641	0.1076	0.0856	0.0657	0.0613	0.0556	0.0391	0.0372	0.0307	0.0267	0.0166	0.0159	0.0134	0.0074	0.0047	4	5	6	6	7	9	10	12	25.9943	117.2970
	acum	0.2685	0.4326	0.5401	0.6257	0.6914	0.7527	0.8082	0.8473	0.8846	0.9153	0.9420	0.9586	0.9745	0.9879	0.9953	1.0000										
FRPCA1	VP	4.2487	2.5490	1.7333	1.3755	1.0489	0.9218	0.9099	0.6192	0.5936	0.5411	0.4900	0.2681	0.2638	0.2372	0.1208	0.0790										
	prop	0.2655	0.1593	0.1083	0.0860	0.0656	0.0576	0.0569	0.0387	0.0371	0.0338	0.0306	0.0168	0.0165	0.0148	0.0076	0.0049	4	5	6	7	8	9	10	12	25.8198	6427.4140
	acum	0.2655	0.4249	0.5332	0.6192	0.6847	0.7423	0.7992	0.8379	0.8750	0.9088	0.9394	0.9562	0.9727	0.9875	0.9951	1.0000										
FRPCA2	VP	4.2099	2.4977	1.7636	1.3734	1.0791	0.9329	0.9141	0.6125	0.5953	0.5438	0.4941	0.2675	0.2659	0.2482	0.1215	0.0805										
	prop	0.2631	0.1561	0.1102	0.0858	0.0674	0.0583	0.0571	0.0383	0.0372	0.0340	0.0309	0.0167	0.0166	0.0155	0.0076	0.0050	4	5	6	7	8	9	10	12	27.0911	8284.0710
	acum	0.2631	0.4192	0.5294	0.6153	0.6827	0.7410	0.7982	0.8364	0.8736	0.9076	0.9385	0.9552	0.9719	0.9874	0.9950	1.0000										
FRPCA3	VP	4.2099	2.4977	1.7636	1.3734	1.0791	0.9329	0.9141	0.6125	0.5953	0.5438	0.4941	0.2675	0.2659	0.2482	0.1215	0.0805										
	prop	0.2631	0.1561	0.1102	0.0858	0.0674	0.0583	0.0571	0.0383	0.0372	0.0340	0.0309	0.0167	0.0166	0.0155	0.0076	0.0050	4	5	6	7	8	9	10	12	27.1975	6835.9490
	acum	0.2631	0.4192	0.5294	0.6153	0.6827	0.7410	0.7982	0.8364	0.8736	0.9076	0.9385	0.9552	0.9719	0.9874	0.9950	1.0000										
nnFRPCA3	VP	4.2574	2.5263	1.5685	1.4943	1.0783	0.9843	0.9097	0.6070	0.5698	0.5270	0.4923	0.2771	0.2528	0.2519	0.1227	0.0806										
	prop	0.2661	0.1579	0.0980	0.0934	0.0674	0.0615	0.0569	0.0379	0.0356	0.0329	0.0308	0.0173	0.0158	0.0157	0.0077	0.0050	4	5	6	7	7	9	10	12	23.4584	554.2030
	acum	0.2661	0.4240	0.5220	0.6154	0.6828	0.7443	0.8012	0.8391	0.8747	0.9077	0.9384	0.9557	0.9715	0.9873	0.9950	1.0000										

Tabla 9: Resultados de la comparación de algoritmos con los datos de Letter Recognition.

D.C.P. = Distribución de las componentes principales, Std = desviación estandar, R Comp = Recursos Computacionales y T Comp = tiempo de cómputo.

Algoritmos		Cantidad de Varianza					Num Componentes principales								Robustez	R Comp
		1CP	2CP	3CP	4CP	5CP	60 %	65 %	70 %	75 %	80 %	85 %	90 %	95 %	Error	T Comp
	VP	2.5909	1.3389	0.7224	0.3478	0.0000										
PCA	prop	0.5182	0.2678	0.1445	0.0696	0.0000	2	2	2	2	3	3	3	4	2.4856	0.0000
	acum	0.5182	0.7860	0.9304	1.0000	1.0000										
	VP	2.5874	1.3371	0.7214	0.3474	0.0000										
RFpca	prop	0.5182	0.2678	0.1445	0.0696	0.0000	2	2	2	2	3	3	3	4	11.9291	0.0310
	acum	0.5182	0.7860	0.9304	1.0000	1.0000										
	VP	3.2125	0.2908	0.0814	0.0350	0.0000										
fpca	prop	0.8875	0.0803	0.0225	0.0097	0.0000	1	1	1	1	1	1	2	2	4.7869	7.4720
	acum	0.8875	0.9678	0.9903	1.0000	1.0000										
	VP	1.8365	0.3684	0.0595	0.0320	0.0000										
fpcapop	prop	0.7997	0.1604	0.0259	0.0139	0.0000	1	1	1	1	2	2	2	2	2.3083	4.1340
	acum	0.7997	0.9602	0.9861	1.0000	1.0000										
	VP	0.6510	0.5469	0.4423	0.1995	0.0000										
fuzzypca	prop	0.3539	0.2973	0.2404	0.1085	0.0000	2	2	3	3	3	3	4	4	6.7383	3.4950
	acum	0.3539	0.6511	0.8915	1.0000	1.0000										
	VP	2.1813	1.4415	0.9379	0.4393	0.0000										
FRPCA1	prop	0.4363	0.2883	0.1876	0.0879	0.0000	2	2	2	3	3	3	3	4	8.6274	73.6790
	acum	0.4363	0.7246	0.9121	1.0000	1.0000										
	VP	2.0015	1.6398	0.9085	0.4502	0.0000										
FRPCA2	prop	0.4003	0.3280	0.1817	0.0900	0.0000	2	2	2	3	3	3	3	4	6.7424	96.3770
	acum	0.4003	0.7283	0.9100	1.0000	1.0000										
	VP	2.0015	1.6398	0.9085	0.4502	0.0000										
FRPCA3	prop	0.4003	0.3280	0.1817	0.0900	0.0000	2	2	2	3	3	3	3	4	9.2861	78.8730
	acum	0.4003	0.7283	0.9100	1.0000	1.0000										
	VP	2.3758	1.3005	0.8290	0.4947	0.0000										
nnFRPCA3	prop	0.4752	0.2601	0.1658	0.0989	0.0000	2	2	2	3	3	3	3	4	3.7383	0.2970
	acum	0.4752	0.7353	0.9011	1.0000	1.0000										

Tabla 10: Resultados de la comparación de algoritmos con los datos de Blood Transfusion Service Center.

D.C.P. = Distribución de las componentes principales, Std = desviación estandar, R Comp = Recursos Computacionales y T Comp = tiempo de cómputo.

Algoritmos		Cantidad de Varianza								Num Componentes principales								Robustez	R Comp
		1CP	2CP	3CP	4CP	5CP	6CP	7CP	8CP	60 %	65 %	70 %	75 %	80 %	85 %	90 %	95 %	Error	T Comp
PCA	VP	3.1000	1.7900	1.0118	0.7384	0.5566	0.4593	0.2811	0.0628										
	prop	0.3875	0.2237	0.1265	0.0923	0.0696	0.0574	0.0351	0.0078	2	3	3	4	4	5	6	6	3.4798	0.0160
	acum	0.3875	0.6112	0.7377	0.8300	0.8996	0.9570	0.9922	1.0000										
RFpca	VP	3.0930	1.7859	1.0095	0.7367	0.5554	0.4583	0.2805	0.0626										
	prop	0.3875	0.2237	0.1265	0.0923	0.0696	0.0574	0.0351	0.0078	2	3	3	4	4	5	6	6	17.2445	0.0150
	acum	0.3875	0.6112	0.7377	0.8300	0.8996	0.9570	0.9922	1.0000										
fpca	VP	1.4386	0.6773	0.4539	0.2593	0.1875	0.1179	0.0739	0.0318										
	prop	0.4440	0.2090	0.1401	0.0800	0.0579	0.0364	0.0228	0.0098	2	2	3	3	4	4	5	6	2.5109	7.2390
	acum	0.4440	0.6530	0.7931	0.8731	0.9310	0.9674	0.9902	1.0000										
fpcapop	VP	1.3661	0.5841	0.4418	0.2289	0.1758	0.1012	0.0625	0.0264										
	prop	0.4574	0.1956	0.1479	0.0766	0.0588	0.0339	0.0209	0.0088	2	2	3	3	3	4	5	6	7.2095	4.0400
	acum	0.4574	0.6529	0.8009	0.8775	0.9363	0.9702	0.9912	1.0000										
fuzzypca	VP	1.4273	0.3502	0.2501	0.1506	0.1312	0.0736	0.0464	0.0193										
	prop	0.5829	0.1430	0.1022	0.0615	0.0536	0.0301	0.0189	0.0079	2	2	2	3	3	4	5	6	14.6833	4.0870
	acum	0.5829	0.7259	0.8281	0.8895	0.9431	0.9732	0.9921	1.0000										
FRPCA1	VP	2.8901	1.4219	1.0556	0.8285	0.7872	0.6685	0.2829	0.0653										
	prop	0.3613	0.1777	0.1320	0.1036	0.0984	0.0836	0.0354	0.0082	3	3	4	4	5	5	6	6	11.8963	75.0100
	acum	0.3613	0.5390	0.6710	0.7745	0.8729	0.9565	0.9918	1.0000										
FRPCA2	VP	2.8660	1.4160	1.0775	0.8334	0.7861	0.6716	0.2834	0.0660										
	prop	0.3582	0.1770	0.1347	0.1042	0.0983	0.0839	0.0354	0.0083	3	3	4	4	5	5	6	6	14.9610	103.1620
	acum	0.3582	0.5352	0.6699	0.7741	0.8724	0.9563	0.9917	1.0000										
FRPCA3	VP	2.8660	1.4160	1.0775	0.8334	0.7861	0.6716	0.2834	0.0660										
	prop	0.3582	0.1770	0.1347	0.1042	0.0983	0.0839	0.0354	0.0083	3	3	4	4	5	5	6	6	18.8050	80.9180
	acum	0.3582	0.5352	0.6699	0.7741	0.8724	0.9563	0.9917	1.0000										
nnFRPCA3	VP	2.7246	1.4262	1.1470	0.8627	0.7748	0.6980	0.2514	0.1153										
	prop	0.3406	0.1783	0.1434	0.1078	0.0969	0.0873	0.0314	0.0144	3	3	4	4	5	5	6	6	11.4210	4.9080
	acum	0.3406	0.5189	0.6622	0.7701	0.8669	0.9542	0.9856	1.0000										

Tabla 11: Resultados de la comparación de algoritmos con los datos de Wholesale Customers.

D.C.P. = Distribución de las componentes principales, Std = desviación estandar, R Comp = Recursos Computacionales y T Comp = tiempo de cómputo.

5. Conclusiones de la Extracción de Variables

Se presenta a continuación la tabla 12, en la que se puede apreciar las conclusiones extraídas de los resultados de la sección anterior.

Dataset	Componentes Principales	Robustez	Recursos Computacionales
Airfoil Self-Noise	fpca / fuzzypca	PCA	RFpca
Breast Tissue	fuzzypca	PCA	PCA
Ecoli	fpca	fpcapop	RFpca
Energy Efficiency	fpca / fpcapop / fuzzypca	nnFRPCA3	PCA
Iris	fpca / fpcapop / fuzzypca	PCA	PCA
Istambul Stock Exchange	PCA / RFpca / fpca / fpcapop / fuzzypca / FRPCA1 / FRPCA2 / FRPCA3	PCA	RFpca
Letter Recognition	PCA / RFpca / fpcapop / fuzzypca	PCA	PCA
Blood Transfusion Service Center	fpca	fpcapop	PCA
Wholesale Customers	fuzzypca	fpca	RFpca

Tabla 12: Prestaciones de los algoritmos a estudio.

Como se observa en la Tabla 12, el mejor algoritmo en cuanto a reducir el número de componentes principales es *fuzzypca*, mientras que los algoritmos más robustos son *PCA* y *fpcapop*, o lo que es lo mismo, toleran mejor los outliers y el ruido.

En cuanto a recursos computacionales, el algoritmo más rápido es *PCA* o *RFpca* dependiendo del dataset. Esto es así porque el algoritmo PCA está muy depurado en comparación con el resto de algoritmos, por otra parte, los FRPCA1, 2 y 3 trabajan con los valores medios de 10 ejecuciones. Esto explica su tiempo de cómputo elevado.

Entrando ahora a la extracción de variables, se observa que todas las técnicas emplean menos variables que las originales, con lo que se consigue una reducción de la dimensionalidad. Este hecho se aprecia mejor en *fuzzypca*, el algoritmo de Sârbu, C. y Pop, H., 2005 [4], en el cual se emplean menos componentes principales que en el resto.

Se ha apreciado que en cuanto a número de componentes principales el comportamiento de los algoritmos ha sido el esperado, a excepción de los algoritmos *FRPCA1*, *FRPCA2*, *FRPCA3* y *nnFRPCA3*, que han presentado una mayor necesidad de componentes principales que PCA clásica.

En cuanto a la robustez se ha encontrado un comportamiento que no cumple con las expectativas salvo *fpcapop*, el cual tiene un comportamiento similar al de PCA clásica, lo que puede estar indicando una mala implementación de los algoritmos, puesto que se esperaba una mejora significativa de estos algoritmos frente a PCA.

Parte II
Detección y Diagnóstico de Fallos mediante Fuzzy PCA

En esta parte se tratará la detección y diagnóstico mediante los algoritmos Fuzzy PCA analizados en la primera parte.

6. Introducción a la Detección y Diagnóstico de Fallos

En el mundo de la detección y diagnóstico de fallos existen diversas técnicas para lograr tal fin. En Venkatasubramanian et al., (2003a) [26], Venkatasubramanian et al., (2003b) [24] y Venkatasubramanian et al., (2003c) [25] se puede encontrar una revisión bastante completa de los métodos de detección de fallos, así como de diagnóstico de los mismos.

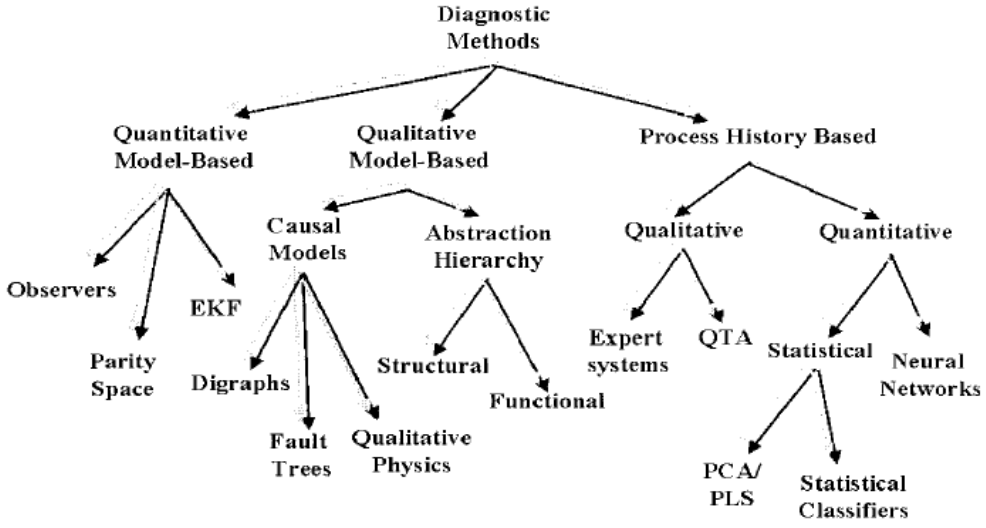

Figura 10: Diagrama de las técnicas de detección y diagnóstico de fallos. (Extraido de Venkatasubramanian et al., (2003a) [26])

En la Figura 10 se puede observar un esquema de los distintas técnicas para la detección de fallos.

Este trabajo se centra en las técnicas Fuzzy PCA, las cuales se engloban dentro de las técnicas PCA. Por ello se puede decir que las técnicas PCA están basadas en los datos históricos del proceso, de forma cuantitativa y estadística en cuanto a detección y diagnóstico de fallos se refiere.

7. Detección de Fallos basada en Fuzzy PCA

Aquí se explicarán el marco teórico y los resultados obtenidos en la detección de fallos.

7.1. Detección de Fallos basada en PCA

Se han seleccionado dos índices para su estudio:

- El estadístico de Hotelling o T^2.
- El error de predicción cuadrado SPE o Q.

Estos índices estadísticos se emplean regularmente en la detección y aislamiento de fallos y son utilizados por Kourti, T. y MacGregor, J., 1996 [8].

A continuación se explican estos índices.

7.1.1. Estadístico de Hotelling T^2

Este índice se basa en la distancia de Mahalanobis y viene definido por la siguiente ecuación:

$$T^2 = z^T P \Lambda_A^{-1} P^T z \tag{13}$$

Donde z es una nueva observación del proceso, P es la matriz de transformación, y sus columnas son conocidas como *loadings* y Λ_A es la matriz diagonal $A \times A$ de los valores propios de mayor valor de la matriz de covarianza.

El proceso se considera bajo control, para un nivel α de significancia dado, si se cumple lo siguiente:

$$T^2 \leq T_\alpha^2 = \frac{(K^2-1)A}{K(K-A)} F_\alpha(A, K-A) \tag{14}$$

Donde $F_\alpha(A, K-A)$ es el valor crítico (percentil $100(1-\alpha)\,\%$) de la distribución F de Fisher-Snedecor con A y $N-A$ grados de libertad, K el número de observaciones y siendo α el nivel de significancia. α toma valores entre 5 y 1 %.

Este índice da una medida de la variabilidad en el proceso capturada por el modelo PCA y se calcula utilizando el subespacio de las componentes seleccionadas, las cuales representan las mayores fuentes de variabilidad.

7.1.2. Error de Predicción Cuadrado Q

Esta estadística se puede calcular mediante la siguiente ecuación:

$$Q = r^T r \tag{15}$$

Con:

$$r = (I - P_{1:A} P^T{}_{1:A}) z \tag{16}$$

Donde r es el vector residual entre la observación y su proyección en el espacio reducido.

De forma análoga al índice de Hotelling, se puede establecer un límite superior de la siguiente manera:

$$Q_\alpha = \theta_1 [\frac{h_0 c_\alpha \sqrt{2\theta_2}}{\theta_1} + 1 + \frac{\theta_2 h_0 (h_0 - 1)}{\theta_2}]^{\frac{1}{h_0}} \tag{17}$$

Con:

$$\theta_i = \sum_{j=a+1}^{m} \lambda_j^i \quad h_0 = 1 - \frac{2\theta_1 \theta_3}{3\theta_2^2} \tag{18}$$

Donde c_α es el percentil normal estándar $100(1-\alpha)$ y λ_j es el valor propio de la matriz de covarianza residual de la PCA $E^T E/(K-1)$.

Por su parte la matriz residual se obtiene de la siguiente forma:

$$E = X - TP_{1:A}^{T} \tag{19}$$

Siendo $T = XP_{1:A}$ la matriz de *scores*, $P_{1:A}$ la matriz de transformación habiendo escogido A componentes principales y X la matriz de datos representando todos los valores de las distintas variables a lo largo del tiempo.

Este índice es una medida de la bondad del ajuste entre la muestra y el modelo PCA y está directamente relacionado con el ruido.

7.2. Detección de Fallos en EDAR, Red de Distribución de Agua y Reactor Químico

Una planta de depuración de aguas residuales es una planta que consta de diferentes elementos que permiten transformar un agua contaminada con los residuos y contaminantes que se vierten en los desagües de una población, en un agua apta para ser vertida en un río, o en cualquier otra masa de agua. Para ello se debe conseguir que la cantidad de contaminantes presentes en ese agua residual sea lo suficientemente baja para que el impacto en la zona de vertido sea lo menor posible.

Las instalaciones de depuración de aguas residuales presentan un grado importante de complejidad, ya que están compuestas de muchos elementos (decantadores, digestores, reactores, etc.) y a su vez, estas partes de las depuradoras funcionan siguiendo una serie de reacciones químicas complejas, sometidas a una gran variedad de factores internos y externos (características del agua residual, diseño de la instalación, temperatura, etc). Debido a esto se hace difícil su manejo, y es muy importante contar con datos fiables y representativos de su estado de funcionamiento.

Esto, a su vez, plantea otros dos problemas: por un lado, como ya se dijo, la cantidad de datos que se pueden extraer de una instalación de este tipo (por ejemplo, oxígeno disuelto, biomasa, alcalinidad, nitritos, amoníaco, etc.) puede ser tan grande que hagan que sea inasequible su estudio en tiempo real, y por otro lado, no suele ser posible acceder instantáneamente a todos los datos de funcionamiento del sistema; es decir, valores como la temperatura o el pH del líquido se pueden medir simplemente introduciendo un sensor en

un punto en el que queramos obtener esos datos, pero no hay sensores que midan, por ejemplo, la cantidad de substrato fácilmente biodegradable (Ss). Para conocer estos datos, habría que tomar una muestra y analizarla en un laboratorio mediante un proceso complejo, lento y caro, que no permite un control en tiempo real de la planta depuradora.

Para ello, se empleará Fuzzy PCA's, las cuales se espera que den solución a la detección temprana de fallos de forma automática, y a la modelización del proceso para posibles simulaciones.

En cuanto a las redes de distribución de agua, estas presentan una gran cantidad de variables debido a la misma red de distribución. Aunque en la realidad no se instalan sensores en cada tramo o nodo de la red, con la herramienta epanet se puede diseñar y simular el comportamiento de una red de distribución de agua de cualquier tamaño, lo que otorga una cantidad de variables directamente proporcional al tamaño de la red. El problema principal que se plantea en estas redes es como tratar la cantidad ingente de datos que puede llegar a aportar la misma.

Por otra parte, si se produce cualquier tipo de fallo en la red, las repercusiones son casi instantaneas para los usuarios, y por ello conviene que se tenga una respuesta lo más temprana posible a estos fallos.

Con la aplicación de Fuzzy PCA's para la detección de fallos se prevee la solución a ambos problemas, puesto que las técnicas de PCA son capaces de tratar con una gran cantidad de datos.

Por último, en los reactores químicos se tiene el mismo problema que en las plantas de tratamiento de aguas residuales: la dificultad del modelado debido a las reacciones químicas. Sin mencionar que en ciertas reacciones químicas peligrosas, un fallo en las concentraciones de los compuestos pueden llegar a ocasionar grandes daños.

En este caso, la aplicación de Fuzzy PCA pretende solventar el proceso de modelado y dar una pronta respuesta en la detección de fallos.

7.3. Metodología Experimental

En esta sección se describe la metodología que se emplea en la parte del trabajo correspondiente a la detección de fallos a la hora de realizar los cálculos necesarios para la obtención de los resultados:

En primer lugar se han escogido los datos que se pueden encontrar en la sección 7.5, los cuales se han sometido a un preprocesamiento, el cual consta de una normalización de los datos en media 0 y desviación 1 para que todas las variables dispongan de la misma relevancia a la hora del cálculo, puesto que si una variable de orden elevado se mantiene constante mientras que otra de orden menor varía en gran medida, no tienen la misma relevancia en los resultados, pero en el cálculo se impondría la variable de mayor orden pese a permanecer constante. Por otra parte se han eliminado todas las variables que permanecían constantes o cerca de serlo atendiendo a la desviación de cada variable ($\sigma < 0{,}001$).

En segundo lugar, se han empleado los algoritmos escogidos para el tratamiento de los datos normalizados de cada conjunto de datos con el fin de obtener de estos los resultados de detección de fallos de cada algoritmo para cada conjunto de datos y porcentaje de variabilidad, el cual se puede encontrar en la sección 7.6, discriminando en conjuntos de datos con y sin outliers. Estos resultados se han recogido en tablas que pueden encontrarse en el anexo I. Los algoritmos difusos, como ya se ha dicho, se explican en la sección 2.3.

En tercer lugar, se han aplicado los criterios de comparación descritos en la sección 7.4. Una vez aplicados estos criterios se han recogido en diagramas de barras, organizados para cada algoritmo y discriminando entre los índices empleados y datos con o sin outliers, que pueden encontrarse en la sección 7.7 correspondiente a las conclusiones de detección de fallos.

En último lugar se ha realizado una tabla resumen de los resultados, a partir de la cual se extraen las conclusiones de los resultados obtenidos.

7.4. Criterios empleados

Los criterios de comparación que se han seleccionado son los siguientes:

- Detección del fallo mediante los índices estadísticos.
- Tiempo de detección del fallo.

A continuación se explican los criterios mencionados.

7.4.1. Detección del fallo mediante los índices estadísticos

Este criterio muestra si el índice es capaz o no de detectar el fallo en cuestión.

Se considera que se ha producido un fallo cuando se cumple la inecuación correspondiente al índice que se esté considerando:

$$T^2 > T^2_\alpha \quad Q > Q_\alpha$$

En los conjuntos de datos que se explicarán más adelante se conoce si existe fallo o no y en qué momento se produce. En este criterio se mostrará la relación de falsos positivos y falsos negativos con sus homólogos de positivos y negativos reales. Todo ello para cada índice.

También se empleará los ratios de detecciones perdidas y falsas alarmas utilizados por Lau et al., (2013) [11], los cuales se explican a continuación.

Positivo	Falso Positivo
Falso Negativo	Negativo

- Positivo: Se considera *Positivo* a toda observación que se sepa que no se encuentra en zona de fallo y se detecta como libre de fallo
- Falso Positivo: Se considera *Falso Positivo* a toda observación que se sepa que se encuentra en zona de fallo y se detecta como libre de fallo.
- Negativo: Se considera *Falso Negativo* a toda observación que se sepa que no se encuentra en zona de fallo y se detecta como fallo.
- Falso Negativo: Se considera *Negativo* a toda observación que se sepa que se encuentra en zona de fallo y se detecta como fallo.

7.4.2. Ratio de detecciones perdidas (MDR)

El ratio de detecciones perdidas o MDR, por sus siglas en inglés, es utilizado por Alcalá, C. F. y Qin, S. J., (2009) [1] y es el porcentaje de observaciones que se sabe que son fallos, pero que se identifican como libres de fallos:

$$MDR = \frac{N^o\text{de Falsos Positivos}}{\text{Falsos Positivos + Negativos}} \times 100\,\% \tag{20}$$

7.4.3. Ratio de falsas alarmas (FAR)

El ratio de falsas alarmas o FAR, por sus siglas en inglés, es utilizado por Alcalá C. F. y Qin S. J., (2009) [1] y se define como el porcentaje de observaciones identificadas como fallos cuando en realidad se encuentran libres de fallos.

$$FAR = \frac{N^o\text{de Falsos Negativos}}{\text{Falsos Negativos + Positivos}} \times 100\,\% \tag{21}$$

7.4.4. Tiempo de detección del fallo

Este criterio muestra el retardo que tienen los índices para registrar el fallo y muestra ese retraso en número de observaciones.

$$t_d = t_i - t_r \tag{22}$$

Donde t_d es el tiempo de detección del fallo, t_i es el instante en el que se detecta el fallo y t_r es el instante en el que realmente ocurre el fallo.

Un tiempo corto permite una reacción más temprana y un posible ahorro en gastos derivados de la calidad.

7.5. Casos de Estudio

Los casos de estudio corresponden a:

- Planta de Tratamiento de Agua.
- Red de Distribución de Agua.
- Planta Química.

A continuación se describen los distintos casos de estudio y los datos que de ellos se extraen.

7.5.1. Planta de Tratamiento de Agua

Este caso de estudio se basa en la planta Benchmark Simulation Model no. 2 (BSM2) [2].

Esta planta se compone de un clarificador primario, un espesador de fangos secundarios, una unidad deshidratante, 5 reactores de fangos activos, un digestor anaeróbico, un clarificador secundario y un tanque de almacenamiento. La Figura 11 muestra un esquema de la planta.

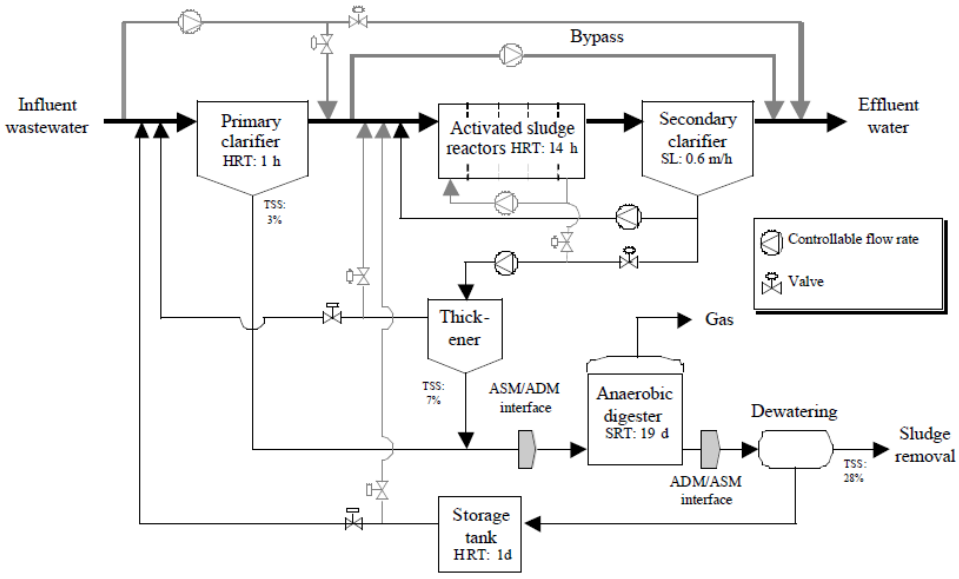

Figura 11: Esquema de la planta BSM2

Se han tomado 5 conjuntos de datos de dos formas distintas, unos mediante sensores que incluyen su propio ruido, y otros de forma directa de las variables del modelo, que ya tiene en cuenta el ruido del ciclo de control.

Las variables que se han tomado, tanto para los datos obtenidos por los sensores como los obtenidos directamente de las variables del modelo se

recogen en la siguiente lista:

- Temperatura en la cuba.
- Oxígeno disuelto en el reactor 1.
- Oxígeno disuelto en el reactor 2.
- Oxígeno disuelto en el reactor 3.
- Oxígeno disuelto en el reactor 4.
- Oxígeno disuelto en el reactor 5.
- Alcalinidad en la cuba.
- Caudal de entrada a la planta.
- Caudal de purga.
- Caudal de recirculación de fangos.
- Demanda química de oxígeno del efluente de salida.
- Demanda química de oxígeno del efluente primario.
- Kg. de demanda química de oxígeno a la entrada del bio-reactor 5.
- Demanda biológica de oxígeno del efluente de salida.
- Demanda biológica de oxígeno del efluente primario.
- Total de sólidos suspendidos en el efluente de salida.
- Total de sólidos suspendidos en el efluente primario.
- Total de sólidos suspendidos en el licor de mezcla.
- Total de sólidos suspendidos en la recirculación de fangos.
- Kg. de sólidos de fangos evacuados al digestor.
- Nitrógeno total a la entrada de la planta.
- Nitrógeno total a la salida de la planta.

La toma de muestras se realiza cada 15 minutos y la simulación dura 609 días, por lo que se tienen 58465 observaciones. Para operar con estos datos de forma más sencilla y para evitar periodos transitorios se ha prescindido de las primeras 10000 observaciones.

Los fallos se han introducido en distintos instantes para los diferentes tipos de fallos. En la Tabla 13 se encuentran los instantes de ocurrencia de los distintos fallos.

Los conjuntos que se han tomado son los siguientes:

- Mediante variables
  - Resultados_OK: Resultados sin fallos.
  - Resultados_caudales_primary: Resultados con fallo en el caudal clarificador primario - digestor.
  - Resultados_kla: Resultados con fallo en el actuador de los reactores.
  - Resultados_O2: Resultados con fallo en el sensor de oxígeno de los reactores.
  - Resultados_Salk_reac1: Resultados con fallo en la alcalinidad.
- Mediante sensores
  - Resultados_sensores_OK: Resultados sin fallos.
  - Resultados_sensores_caudales_primary: Resultados con fallo en el caudal clarificador primario - digestor.
  - Resultados_sensores_kla: Resultados con fallo en el actuador de los reactores.
  - Resultados_sensores_O2: Resultados con fallo en el sensor de oxígeno de los reactores.
  - Resultados_sensores_Salk_reac1: Resultados con fallo en la alcalinidad.
- **Resultados_OK y Resultados_sensores_OK** Estos conjuntos de datos se obtienen de simular de forma normal, es decir, sin ninguna modificación, el modelo BSM2.

En el caso de Resultados_sensores_OK simplemente se han añadido los sensores, los cuales no interfieren con el comportamiento original del modelo.

- **Resultados_caudales_primary y Resultados_sensores_caudales_primary** Estos conjuntos de datos se obtienen al introducir al modelo una desviación en el caudal del flujo secundario del clarificador que va al digestor. El fallo consiste en un salto en el caudal de $-30\,\%$ en el día 200.
- **Resultados_O2 y Resultados_sensores_O2** Estos conjuntos de datos se obtienen al falsear la lectura del sensor de oxígeno de los reactores. El fallo consiste en que en el día 200 se dobla la señal de salida del sensor de oxígeno.
- **Resultados_kla y Resultados_sensores_kla** Estos conjuntos de datos proceden de modificar la señal del actuador de los reactores. El fallo consiste en un salto en la señal dirigida a los actuadores del $-40\,\%$ en el día 300.
- **Resultados_Salk_reac1 y Resultados_sensores_Salk_reac1** Estos conjuntos de datos se obtienen al modificar la alcalinidad simulando un vertido. El fallo consiste en que en el día 350 se aumenta la alcalinidad en un $20\,\%$.

Tipo de fallo	Instante en días	Instante en observaciones
OK	No existe fallo	No existe fallo
caudales_primary	200	19200
kla	300	28800
O2	200	19200
Salk_reac1	350	33600

Tabla 13: Instante de aplicación del fallo para los distintos conjuntos de datos de BSM2. Cada tipo hace referencia al conjunto con y sin sensores.

7.5.2. Red de Distribución de Agua

La red de distribución de agua se han conseguido a partir de un esquema obtenido mediante la herramienta Epanet, que se puede ver en la Figura 12, al que se han ido añadiendo distintos fallos modificando el esquema original.

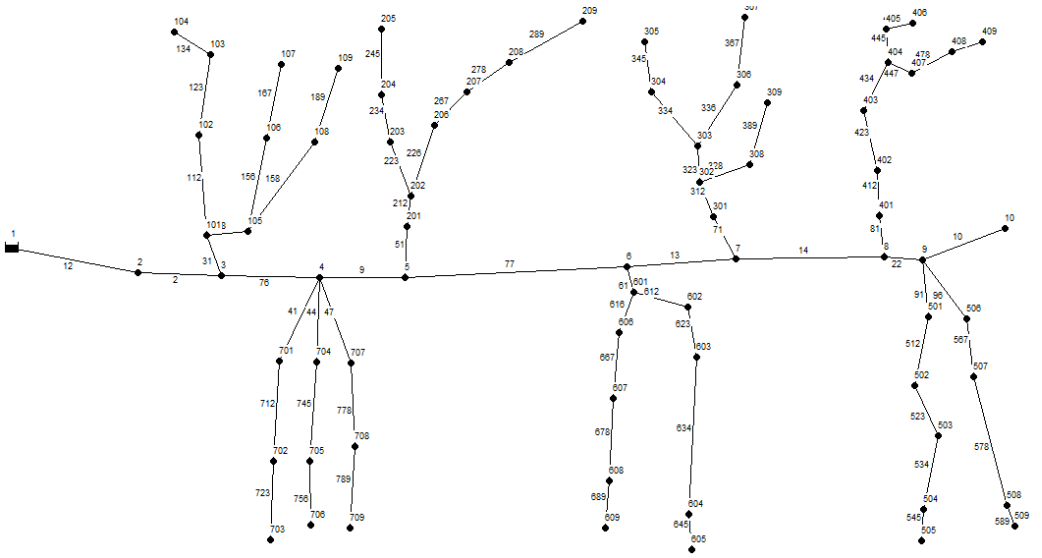

Figura 12: Esquema de la red de distribución de aguas.

Como se puede observar en la Figura 12, la red de distribución de agua consta de un tanque de almacenamiento de agua, el cual se considera infinito puesto que está representando la salida de una planta potabilizadora y varios tramos y nodos, que representan las tuberías de la red y los puntos de consumo respectivamente.

Las variables que conforman los conjuntos de datos son:

- Node head
- Node actual demand
- Node pressure
- Link flow
- Link velocity
- Link unit headloss
- Link friction factor

- Node quality
- Link quality

Para cada nodo se toman las variables “Node” y para cada tramo se toman las variables “Link”.

De esta manera el tamaño de los conjuntos de datos de esta red, tienen unas dimensiones de 201×652.

Los fallos que se han introducido al esquema son los siguientes:

- Fallo de fuga en la red.
- Fallo de contaminante.
- Fallo en la bomba.
- **Fallo de fuga en la red** Este fallo consiste en simular una fuga en el instante 150 abriendo la tubería 1000 que en un primer momento se encuentra cerrada, el esquema se puede apreciar en la Figura 13.

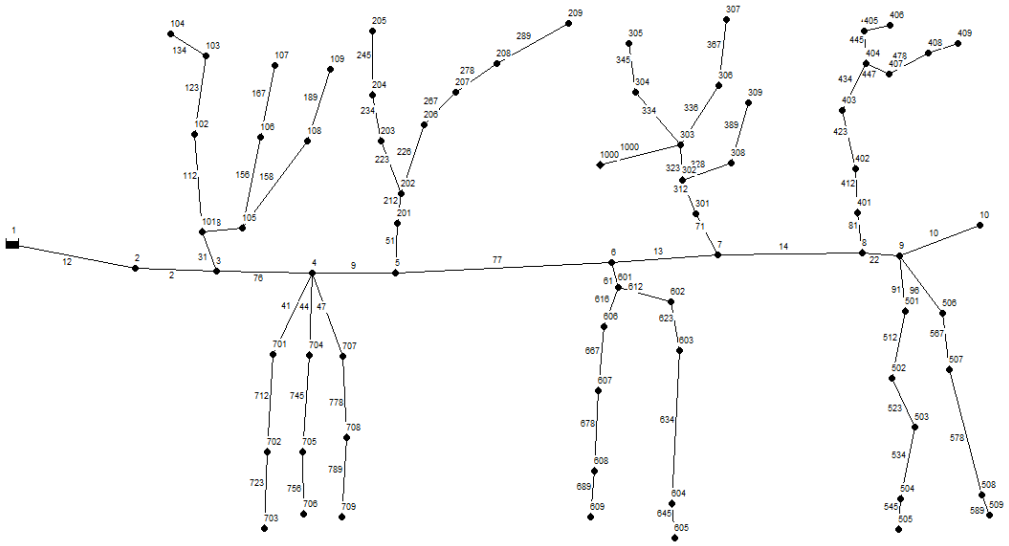

Figura 13: Esquema de la red de distribución de aguas para el fallo de fuga.

- **Fallo de contaminante** Este fallo consiste en simular un vertido contaminante en el instante 150 proveniente del depósito 1000 como se puede observar en la Figura 14.

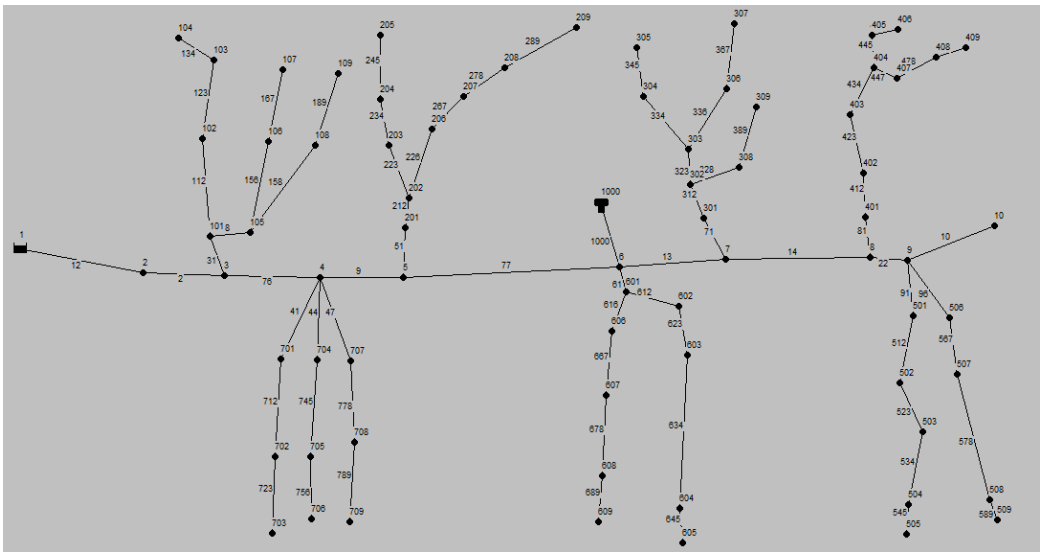

Figura 14: Esquema de la red de distribución de aguas para el fallo de contaminante.

- **Fallo en la bomba** Para este fallo se intercambia el nodo 2 por una bomba, y el fallo se produce al disminuir el rendimiento de la bomba 1 en un 10 % en el instante 150. El esquema se encuentra en la Figura 15.

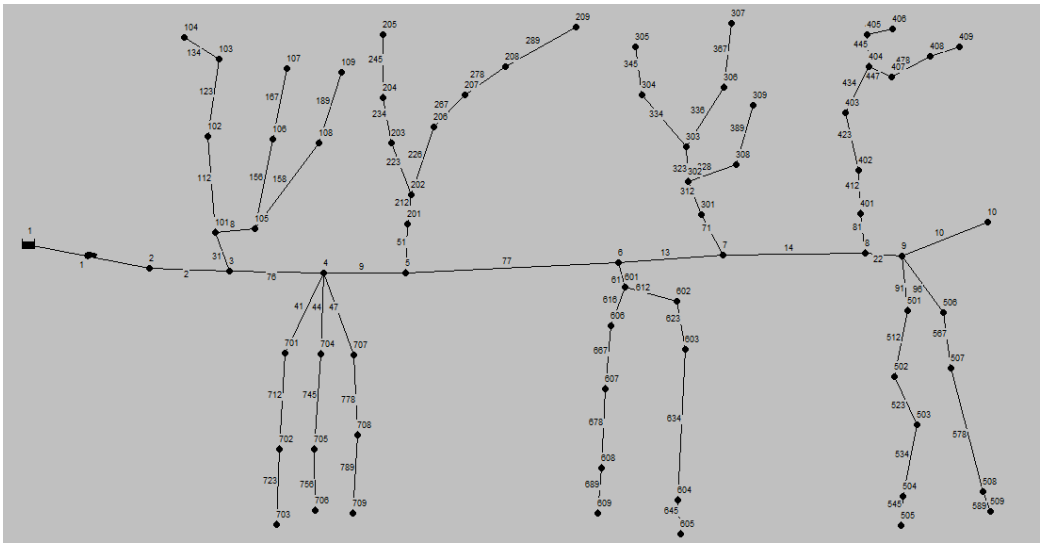

Figura 15: Esquema de la red de distribución de aguas para el fallo en la bomba.

7.5.3. Planta Química

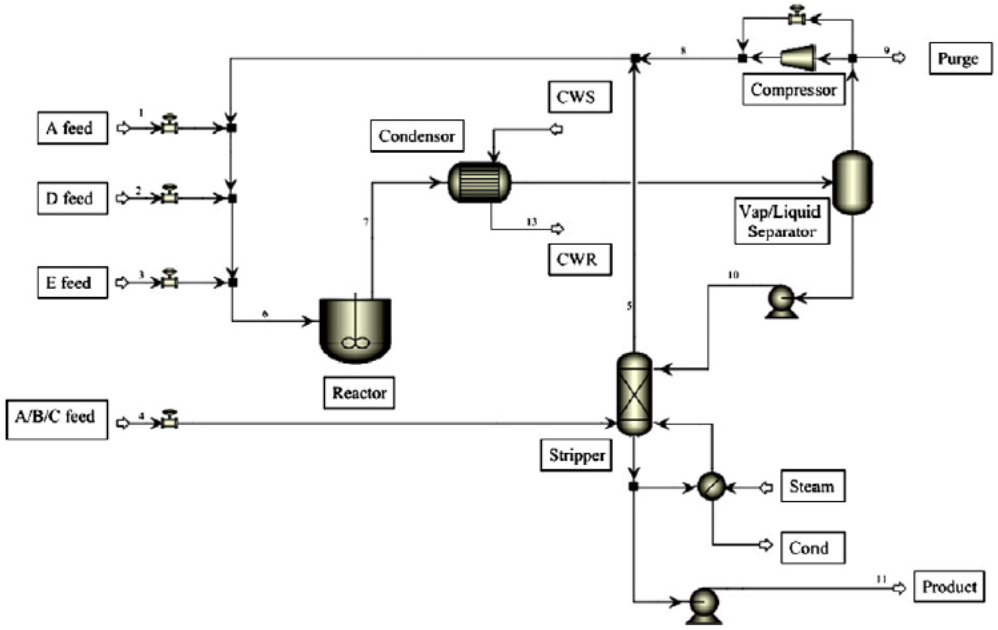

Figura 16: Esquema de la planta Tennessee.

Este caso de estudio se basa en la planta planteada por Downs, J. y Vogel, E., 1993 [5], la cual consiste en 5 unidades principales que son: un reactor, un condensador, un compresor, un separador y un decapante; como se puede ver en la Figura 16.

En esta planta se introducen en el reactor los reactivos A, C, D y E donde los líquidos G y H producen las siguientes reacciones exotérmicas:

$$A(g) + C(g) + D(g) \rightarrow G(l)$$
$$A(g) + C(g) + E(g) \rightarrow H(l)$$
$$A(g) + E(g) \rightarrow F(l)$$
$$3D(g) \rightarrow 2F(g)$$

El proceso tiene un total de 53 variables, de las cuales 22 son medidas continuas del proceso, como las temperaturas, presiones, etc, 19 variables son medidas compuestas y las 12 restantes son variables manipuladas. De las 53 variables una permanece constante, la velocidad del agitador del reactor, por lo que se descarta, con lo que finalmente se tienen 52 variables.

No	Descripción	Tipo
1	Ratio de alimentación de A/C, constante de composición de B.	Salto
2	Composición de B, ratio A/C constante.	Salto
3	Alimentación de D.	Salto
4	Temperatura de la válvula de agua refrigerante del reactor.	Salto
5	Temperatura de la válvula de agua refrigerante del condensador.	Salto
6	Pérdida de alimentación de A.	Salto
7	Pérdida de presión de la cabecera de C - Disponibilidad reducida.	Salto
8	Composición de A, B y C.	Variación aleatoria
9	Temperatura de alimentación de D.	Variación aleatoria
10	Temperatura de alimentación de C.	Variación aleatoria
11	Temperatura de la válvula de agua refrigerante del reactor.	Variación aleatoria
12	Temperatura de la válvula de agua refrigerante del condensador.	Variación aleatoria
13	Cinética de la reacción.	Deriva lenta
14	Válvula de agua refrigerante del reactor.	Bloqueo
15	Válvula de agua refrigerante del condensador.	Bloqueo
16	Desconocido.	-
17	Desconocido.	-
18	Desconocido.	-
19	Desconocido.	-
20	Desconocido.	-
21	Válvula de flujo 4.	Bloqueo

Tabla 14: Tipos de fallos en el proceso Tennessee.

Existen 21 fallos distintos en el proceso Tennessee como se recoge en la Tabla 14. Los conjuntos de datos se dividen en datos de entrenamiento y datos de prueba. Cada conjunto de datos de entrenamiento se ha conseguido mediante una simulación de 25 horas (1500 minutos) de operación con un intervalo de toma de muestras de 3 minutos. Todos los fallos se han introducido al cabo de una hora de simulación, por lo que las primeras 20 muestras de los conjuntos se han descartado puesto que corresponden al régimen normal del proceso. Así pues, el fallo empieza en el instante inicial a efectos prácticos.

Finalmente se tienen 22 conjuntos de datos de entrenamiento, el primero

corresponde al proceso sin fallos y tiene unas dimensiones de 500×52, en cuanto al resto de conjuntos, corresponden a los 21 tipos de fallos y tienen unas dimensiones de 480×52.

En contraposición, los datos de prueba se han obtenido mediante una simulación de 48 horas (960 observaciones) con el fallo introducido transcurridas 8 horas (observación 161). Por ello se tienen 22 conjuntos de datos de prueba, el primero, correspondiente al proceso sin fallos, con unas dimensiones de 960×52 y los 21 restantes, correspondientes a los distintos fallos, con unas dimensiones de 800×52.

Para simplificar, se ha decidido emplear únicamente los datos de test.

7.6. Análisis de los Resultados obtenidos en Detección de Fallos

Antes de obtener los resultados, y para que al normalizar los datos no se formen posibles singularidades, los datos se someten a un preprocesamiento que consiste en prescindir de todas aquellas variables que sean constantes o se encuentren próximas a serlo. Esto se consigue eliminando todas aquellas variables que tengan un valor menor a 0.001 en su desviación estándar. Seguidamente se normalizan los datos para que todas las variables tengan una influencia similar.

Los resultados se han obtenido para distintas tasas de variabilidad explicada por los modelos PCA, esto es:

- 60 % de la variabilidad.
- 65 % de la variabilidad.
- 70 % de la variabilidad.
- 75 % de la variabilidad.
- 80 % de la variabilidad.
- 85 % de la variabilidad.
- 90 % de la variabilidad.

- 95 % de la variabilidad.

De esta forma se puede estudiar un rango en lugar de un caso concreto. Esta decisión se basa en que Liu, J. y Cheng, J., (2014)[13], Wang, D. y Romagnoli, J. A., (2005)[27] y Lau et al., (2013)[11] en concreto, y cada autor en general, utilizan distintas cantidades de variabilidad para sus casos de estudio en sus respectivos artículos; y como no existe un consenso al respecto se ha tomado este rango.

También se quiere señalar que se ha procedido al estudio de los casos sin outliers y los casos con outliers. Los outliers se han creado de forma aleatoria, la semilla tiene el valor del instante de tiempo en que se crean los outliers, con un rango del 95-percentil y un desplazamiento positivo del 99-percentil. En caso de que el 95-percentil sea menor que la unidad, se toma como rango la unidad.

El número de avisos consecutivos que se ha elegido para considerar la existencia de un fallo es la siguiente:

- 40 para los datos de BSM2 y BSM2 con sensores.
- 5 para los datos de Epanet, por ser este número el que mejor comportamiento demuestra entre MDR y FAR.
- 1 para los datos de Tennessee.

Se quiere matizar que para los datos de la red Epanet se empleó un rango $[1, 3, 5, 7, 9]$ para encontrar el consenso entre el mejor comportamiento para MDR y para FAR, en lo que respecta al número de avisos para considerar la ocurrencia de un fallo. El mejor resultado se encontró para 5 avisos consecutivos y por ello se escogió.

El nivel de significancia escogido, tanto para el estadístico de Hotelling(T^2) como para el error de predicción cuadrado(Q), ha sido 95 % lo que conlleva un $\alpha = 0{,}05$.

En caso de encontrarse un 0 en las columnas correspondientes a los tiempos de detección se está indicando que no se ha detectado ningún fallo.

Por cuestiones de espacio, los resultados se encuentran en el Anexo I, pero aquí se muestra un resumen de los resultados, para los cuales se ha empleado la media a lo largo de cada conjunto y para todas las variabilidades.

pca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	2716.8000	27.76	2.11	6.4000	49.71	0.00
65 %	2716.8000	27.76	2.11	6.4000	49.71	0.00
70 %	2027.2000	50.57	1.71	6.8000	43.45	0.00
75 %	2027.2000	50.57	1.71	6.8000	43.45	0.00
80 %	1510.4000	50.10	2.26	6.4000	46.22	0.00
85 %	1505.2000	51.78	2.03	11.2000	33.77	0.00
90 %	1504.6000	51.74	2.43	693.0000	25.48	0.00
95 %	1504.0000	51.63	2.37	9328.4000	25.10	0.00

Tabla 15: Comparación del algoritmo pca con los datos BSM2.

RFpca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	50.2000	24.49	0.00	6.6000	49.66	0.00
65 %	50.2000	24.49	0.00	6.6000	49.66	0.00
70 %	39.2000	24.65	0.00	9.8000	42.07	0.00
75 %	39.2000	24.65	0.00	9.8000	42.07	0.00
80 %	41.2000	23.93	0.00	6.8000	45.66	0.00
85 %	39.8000	24.25	0.00	248.4000	25.70	0.00
90 %	51.8000	23.20	0.00	97.6000	26.76	0.00
95 %	60.0000	22.00	0.00	146.8000	47.62	19.40

Tabla 16: Comparación del algoritmo RFpca con los datos BSM2.

fpca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	2215.8000	34.74	0.89	4564.6000	47.29	0.78
65 %	1816.4000	44.96	0.71	4564.6000	47.35	0.78
70 %	1573.2000	51.05	0.77	5794.8000	25.97	0.48
75 %	1573.2000	51.05	0.77	5794.8000	25.97	0.48
80 %	1510.0000	49.25	2.41	6.4000	47.96	0.00
85 %	517.4000	54.58	6.71	12209.8000	25.11	0.03
90 %	1045.2000	54.42	3.86	232.4000	26.22	0.00
95 %	1513.4000	51.64	3.33	3.8000	33.94	7.31

Tabla 17: Comparación del algoritmo fpca con los datos BSM2.

fpcapop						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	4563.2000	46.00	0.43
65 %	0.0000	0.00	0.00	4563.2000	46.00	0.43
70 %	0.0000	0.00	0.00	4563.2000	46.00	0.43
75 %	1727.2000	47.66	0.55	4568.0000	35.89	0.43
80 %	1510.4000	50.10	2.26	6.4000	46.30	0.00
85 %	517.6000	54.50	7.10	4554.8000	24.99	0.02
90 %	1504.6000	51.74	5.70	693.0000	25.48	0.00
95 %	1504.0000	52.17	2.76	9328.4000	25.10	0.00

Tabla 18: Comparación del algoritmo fpcapop con los datos BSM2.

fuzzypca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	2765.0000	33.13	0.62	4564.0000	46.88	0.78
65 %	1399.0000	40.76	1.07	4564.2000	46.93	0.78
70 %	1573.2000	51.05	0.77	5794.8000	25.97	0.48
75 %	1573.2000	51.05	0.77	5794.8000	25.97	0.48
80 %	1510.4000	50.10	2.26	6.4000	46.22	0.00
85 %	517.6000	54.50	7.10	4554.8000	24.99	0.02
90 %	1504.6000	51.74	3.98	693.0000	25.48	0.00
95 %	1504.0000	51.63	2.37	1672.4000	25.08	0.00

Tabla 19: Comparación del algoritmo fuzzypca con los datos BSM2.

FRPCA1						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	2438.0000	50.06	0.22	4819.2000	27.23	0.38
65 %	2438.0000	50.13	0.22	4819.6000	27.23	0.38
70 %	2034.2000	51.09	1.83	13641.4000	28.46	0.02
75 %	2033.4000	50.82	1.80	13641.4000	26.58	0.03
80 %	2021.0000	51.25	1.60	9507.6000	25.79	0.02
85 %	2022.0000	50.86	1.11	12508.8000	25.20	0.05
90 %	2775.2000	50.91	1.37	10210.0000	25.27	0.02
95 %	2534.2000	51.03	1.40	13630.2000	26.39	0.19

Tabla 20: Comparación del algoritmo FRPCA1 con los datos BSM2.

FRPCA2						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	5095.8000	50.11	0.25	4251.6000	25.45	0.46
65 %	1628.4000	50.89	1.54	16746.4000	25.43	0.07
70 %	1641.2000	50.89	1.47	13535.8000	25.22	0.11
75 %	1500.6000	51.39	2.27	1404.0000	27.51	0.02
80 %	1496.6000	51.61	2.27	2126.4000	25.15	0.02
85 %	1497.0000	51.72	2.59	3238.8000	25.19	0.02
90 %	1035.2000	52.09	3.07	16631.0000	25.12	0.08
95 %	529.2000	52.19	3.13	9234.2000	26.87	0.35

Tabla 21: Comparación del algoritmo FRPCA2 con los datos BSM2.

FRPCA3						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	5095.8000	50.14	0.26	4010.8000	25.71	0.44
65 %	2431.8000	50.16	0.33	4010.6000	25.60	0.42
70 %	2061.4000	50.67	1.54	17955.2000	25.09	0.07
75 %	1497.0000	51.69	2.21	16633.6000	25.02	0.00
80 %	1511.8000	51.58	2.73	1399.2000	30.90	0.00
85 %	1496.0000	51.73	2.45	4039.4000	25.30	0.00
90 %	1496.0000	51.81	2.40	8385.6000	25.22	0.02
95 %	1495.0000	52.08	3.15	13423.4000	40.52	0.21

Tabla 22: Comparación del algoritmo FRPCA3 con los datos BSM2.

nnFRPCA3						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	9273.4000	49.94	0.33	3018.8000	28.34	0.46
65 %	1633.0000	50.98	1.29	12511.4000	25.19	0.03
70 %	1633.0000	50.98	1.15	12511.4000	25.19	0.07
75 %	1610.8000	51.34	1.40	12226.4000	25.19	0.05
80 %	1610.8000	51.37	1.40	12512.4000	25.15	0.03
85 %	1497.2000	51.52	1.81	8088.2000	25.39	0.10
90 %	1495.2000	51.51	1.32	7787.2000	25.47	0.37
95 %	515.8000	52.07	2.64	7779.0000	25.64	0.65

Tabla 23: Comparación del algoritmo nnFRPCA3 con datos BSM2.

pca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	8138.6000	0.71	0.05
65 %	0.0000	0.00	0.00	8138.6000	0.71	0.05
70 %	0.0000	0.00	0.00	8138.6000	0.71	0.05
75 %	0.0000	0.00	0.00	18820.8000	0.46	0.02
80 %	0.0000	0.00	0.00	18820.8000	0.46	0.02
85 %	0.0000	0.00	0.00	5134.2000	1.73	0.03
90 %	0.0000	0.00	0.00	5134.2000	1.73	0.03
95 %	8133.2000	0.43	0.05	7654.8000	0.03	0.00

Tabla 24: Comparación del algoritmo pca con los datos BSM2sen.

RFpca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	8138.6000	0.71	0.05
65 %	0.0000	0.00	0.00	8138.6000	0.71	0.05
70 %	0.0000	0.00	0.00	8138.6000	0.71	0.05
75 %	0.0000	0.00	0.00	18820.8000	0.46	0.02
80 %	0.0000	0.00	0.00	18820.8000	0.46	0.02
85 %	0.0000	0.00	0.00	5545.0000	1.52	0.03
90 %	0.0000	0.00	0.00	5545.0000	1.52	0.03
95 %	0.0000	0.00	0.00	7654.8000	0.03	0.00

Tabla 25: Comparación del algoritmo RFpca con los datos BSM2sen.

fpca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
65 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
70 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
75 %	0.0000	0.00	0.00	7655.4000	0.03	0.00
80 %	0.0000	0.00	0.00	7655.4000	0.03	0.00
85 %	0.0000	0.00	0.00	7655.4000	0.03	0.00
90 %	0.0000	0.00	0.02	5545.4000	0.63	0.05
95 %	5591.2000	2.80	0.10	19372.8000	0.38	0.02

Tabla 26: Comparación del algoritmo fpca con los datos BSM2sen.

fpcapop						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
65 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
70 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
75 %	0.0000	0.00	0.00	7655.4000	0.03	0.00
80 %	0.0000	0.00	0.00	7655.4000	0.03	0.00
85 %	0.0000	0.00	0.00	5134.2000	1.73	0.03
90 %	0.0000	0.00	0.00	5134.2000	1.73	0.03
95 %	8133.2000	0.43	0.05	7654.8000	0.03	0.00

Tabla 27: Comparación del algoritmo fpcapop con los datos BSM2sen.

fuzzypca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
65 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
70 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
75 %	0.0000	0.00	0.00	7655.4000	0.03	0.00
80 %	0.0000	0.00	0.00	7655.4000	0.03	0.00
85 %	0.0000	0.00	0.00	5134.2000	1.73	0.03
90 %	0.0000	0.00	0.00	5134.2000	1.73	0.03
95 %	7094.4000	4.91	0.39	7654.8000	0.03	0.00

Tabla 28: Comparación del algoritmo fuzzypca con los datos BSM2sen.

FRPCA1						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	7655.6000	0.03	0.00
65 %	0.0000	0.00	0.00	21404.0000	0.28	0.02
70 %	0.0000	0.00	0.00	21404.0000	0.28	0.02
75 %	0.0000	0.00	0.00	21404.0000	0.28	0.02
80 %	0.0000	0.00	0.00	21404.0000	0.28	0.02
85 %	0.0000	0.00	0.00	6103.2000	0.50	0.03
90 %	0.0000	0.00	0.00	6103.2000	0.50	0.03
95 %	7614.6000	0.33	0.03	19370.0000	0.33	0.02

Tabla 29: Comparación del algoritmo FRPCA1 con los datos BSM2sen.

FRPCA2						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
65 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
70 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
75 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
80 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
85 %	0.0000	0.00	0.00	6105.6000	0.48	0.03
90 %	0.0000	0.00	0.00	6105.6000	0.48	0.03
95 %	5593.6000	0.60	0.05	21692.0000	0.25	0.02

Tabla 30: Comparación del algoritmo FRPCA2 con los datos BSM2sen.

FRPCA3						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
65 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
70 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
75 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
80 %	0.0000	0.00	0.00	24361.0000	0.23	0.02
85 %	0.0000	0.00	0.00	6105.6000	0.48	0.03
90 %	0.0000	0.00	0.00	6105.6000	0.48	0.03
95 %	5593.6000	0.60	0.05	21692.0000	0.25	0.02

Tabla 31: Comparación del algoritmo FRPCA3 con los datos BSM2sen.

nnFRPCA3						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	8967.8000	0.31	0.00
65 %	0.0000	0.00	0.00	8967.8000	0.31	0.00
70 %	0.0000	0.00	0.00	8967.8000	0.31	0.00
75 %	0.0000	0.00	0.00	19379.4000	0.38	0.02
80 %	0.0000	0.00	0.00	19379.4000	0.38	0.02
85 %	0.0000	0.00	0.00	7615.0000	0.45	0.03
90 %	0.0000	0.00	0.00	7615.0000	0.45	0.03
95 %	7613.6000	0.88	0.08	21692.6000	0.25	0.02

Tabla 32: Comparación del algoritmo nnFRPCA3 con los datos BSM2sen.

pca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	2.6667	25.64	1.12	0.6667	32.05	1.12
65 %	2.6667	25.64	2.24	1.0000	64.10	0.00
70 %	2.6667	25.64	2.24	1.0000	64.10	0.00
75 %	2.3333	28.85	4.47	1.0000	64.10	0.00
80 %	2.3333	28.85	4.47	1.0000	64.10	0.00
85 %	1.0000	28.85	3.36	1.0000	64.10	0.00
90 %	1.0000	28.85	2.24	1.0000	64.10	0.00
95 %	5.0000	32.05	1.12	1.3333	96.15	0.00

Tabla 33: Comparación del algoritmo pca con los datos epanet.

RFpca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	1.0000	64.10	0.00
65 %	0.0000	0.00	0.00	1.0000	64.10	0.00
70 %	0.0000	0.00	0.00	1.0000	64.10	0.00
75 %	0.0000	0.00	0.00	1.0000	64.10	0.00
80 %	0.0000	0.00	0.00	1.0000	64.10	0.00
85 %	2.0000	28.85	0.00	1.0000	64.10	0.00
90 %	0.0000	0.00	0.00	1.0000	64.10	0.00
95 %	1.3333	28.85	0.00	1.3333	96.15	0.00

Tabla 34: Comparación del algoritmo RFpca con los datos epanet.

fpca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	3.0000	25.64	2.24	1.0000	57.69	2.24
65 %	3.0000	25.64	2.24	1.0000	57.69	2.24
70 %	3.0000	25.64	2.24	1.0000	57.69	2.24
75 %	2.3333	28.85	4.47	1.0000	64.10	0.00
80 %	2.3333	28.85	4.47	1.0000	64.10	0.00
85 %	1.0000	28.85	3.36	1.0000	64.10	0.00
90 %	1.0000	28.85	2.24	1.3333	80.13	0.00
95 %	5.3333	64.10	1.12	8.3333	73.72	0.00

Tabla 35: Comparación del algoritmo fpca con los datos epanet.

fpcapop						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	2.6667	25.64	2.24	1.0000	64.10	0.00
65 %	2.6667	25.64	2.24	1.0000	64.10	0.00
70 %	2.6667	25.64	2.24	1.0000	64.10	0.00
75 %	2.3333	28.85	4.47	1.0000	64.10	0.00
80 %	2.3333	28.85	4.47	1.0000	64.10	0.00
85 %	1.0000	28.85	3.36	1.0000	64.10	0.00
90 %	1.0000	28.85	2.24	1.0000	64.10	0.00
95 %	5.0000	32.05	1.12	1.3333	96.15	0.00

Tabla 36: Comparación del algoritmo fpcapop con los datos epanet.

fuzzypca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	2.6667	25.64	2.24	1.0000	64.10	0.00
65 %	2.6667	25.64	2.24	1.0000	64.10	0.00
70 %	2.6667	25.64	2.24	1.0000	64.10	0.00
75 %	2.3333	28.85	4.47	1.0000	64.10	0.00
80 %	2.3333	28.85	4.47	1.0000	64.10	0.00
85 %	1.0000	28.85	3.36	1.0000	64.10	0.00
90 %	1.0000	28.85	2.24	1.0000	64.10	0.00
95 %	5.0000	32.05	1.12	1.3333	96.15	0.00

Tabla 37: Comparación del algoritmo fuzzypca con los datos epanet.

FRPCA1						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	0.0000	0.00	0.00	0.6667	32.05	2.24
65 %	0.0000	0.00	0.00	0.6667	32.05	2.24
70 %	1.3333	28.85	0.00	0.6667	32.05	2.24
75 %	1.3333	28.85	1.12	0.6667	32.05	2.24
80 %	1.0000	28.85	3.36	0.6667	32.05	1.12
85 %	1.0000	28.85	3.36	1.0000	64.10	0.00
90 %	1.0000	28.85	4.47	1.0000	64.10	0.00
95 %	1.0000	44.87	4.47	1.3333	96.15	0.00

Tabla 38: Comparación del algoritmo FRPCA1 con los datos epanet.

FRPCA2						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	2.0000	6.41	2.24	0.6667	32.05	2.24
65 %	2.0000	6.41	2.24	0.6667	32.05	2.24
70 %	1.6667	6.41	2.24	5.0000	38.46	2.24
75 %	1.0000	28.85	2.24	5.0000	38.46	1.12
80 %	1.0000	28.85	3.36	1.0000	64.10	1.12
85 %	1.0000	28.85	4.47	1.0000	64.10	1.12
90 %	1.0000	28.85	4.47	1.3333	96.15	0.00
95 %	0.6667	32.05	3.36	1.3333	96.15	0.00

Tabla 39: Comparación del algoritmo FRPCA2 con los datos epanet.

FRPCA3						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	2.0000	6.41	2.24	0.6667	32.05	2.24
65 %	2.0000	6.41	2.24	0.6667	32.05	2.24
70 %	1.6667	6.41	2.24	5.0000	38.46	2.24
75 %	1.0000	28.85	2.24	5.0000	38.46	1.12
80 %	1.0000	28.85	3.36	1.0000	64.10	1.12
85 %	1.0000	28.85	4.47	1.0000	64.10	1.12
90 %	1.0000	28.85	4.47	1.3333	96.15	0.00
95 %	0.6667	32.05	4.47	1.3333	96.15	0.00

Tabla 40: Comparación del algoritmo FRPCA3 con los datos epanet.

nnFRPCA3						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	10.3333	3.21	0.00	0.6667	32.05	2.24
65 %	1.0000	28.85	1.12	0.6667	32.05	2.24
70 %	0.6667	32.05	1.12	0.6667	32.05	2.24
75 %	0.6667	32.05	4.47	0.6667	32.05	2.24
80 %	0.6667	32.05	4.47	0.6667	32.05	2.24
85 %	0.6667	32.05	4.47	5.0000	35.26	2.24
90 %	0.6667	32.05	4.47	1.0000	64.10	1.12
95 %	0.6667	32.05	4.47	1.3333	96.15	2.24

Tabla 41: Comparación del algoritmo nnFRPCA3 con los datos epanet.

pca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	46.6818	52.21	4.48	15.4091	67.31	5.00
65 %	39.3636	52.58	4.27	10.6818	68.29	4.69
70 %	46.6818	52.75	3.96	9.2727	68.89	5.00
75 %	39.3636	52.98	4.69	8.3182	70.39	5.21
80 %	35.7273	55.22	4.06	6.2273	71.05	4.69
85 %	39.0455	55.28	4.27	5.5909	72.05	5.21
90 %	36.1364	56.77	2.81	8.3182	71.95	5.10
95 %	34.0909	58.84	3.02	10.8636	71.78	5.52

Tabla 42: Comparación del algoritmo pca con los datos Tennessee.

RFpca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	91.0909	38.89	0.00	14.7273	66.71	5.00
65 %	90.5455	30.09	0.00	11.6818	66.76	5.10
70 %	91.0909	38.01	0.00	10.9091	68.15	4.90
75 %	90.5455	37.81	0.00	7.1818	69.14	5.31
80 %	90.7273	38.10	0.00	7.9091	69.87	4.69
85 %	90.8182	37.60	0.00	5.5909	71.11	5.52
90 %	106.9545	43.48	0.00	8.5455	70.66	5.10
95 %	79.2273	44.19	0.00	15.0909	70.28	5.52

Tabla 43: Comparación del algoritmo RFpca con los datos Tennessee.

fpca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	44.5909	52.20	4.48	14.7727	66.28	5.10
65 %	45.7727	52.96	4.27	11.8182	66.91	4.69
70 %	44.5909	54.66	4.79	10.6364	67.34	5.21
75 %	45.7727	55.05	4.48	8.2727	69.14	5.21
80 %	36.0455	56.13	4.06	8.3182	69.35	4.38
85 %	39.6818	55.69	4.58	5.5909	70.89	5.21
90 %	33.2727	57.13	3.33	8.4091	70.40	4.48
95 %	33.8182	59.08	3.44	12.1364	70.20	5.52

Tabla 44: Comparación del algoritmo fpca con los datos Tennessee.

fpcapop						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	49.4545	51.93	4.48	14.6818	66.84	4.69
65 %	47.2273	52.98	4.27	11.0000	67.02	4.69
70 %	49.4545	52.92	4.06	10.9091	68.14	5.00
75 %	47.2273	53.84	4.69	7.1818	69.27	5.21
80 %	39.7727	56.42	4.06	8.1818	69.71	4.69
85 %	38.1818	56.24	4.38	5.5909	71.20	4.69
90 %	34.5455	56.46	2.92	8.5455	70.66	5.10
95 %	38.4091	58.22	3.02	15.0909	70.30	5.52

Tabla 45: Comparación del algoritmo fpcapop con los datos Tennessee.

fuzzypca						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	48.5455	51.93	4.48	14.6818	66.84	5.00
65 %	48.0909	52.92	4.27	11.0000	67.02	4.69
70 %	48.5455	52.92	4.06	10.9091	68.30	5.00
75 %	48.0909	53.82	4.69	7.1818	69.27	5.21
80 %	40.3182	55.30	4.06	7.9091	69.98	4.69
85 %	39.4545	55.60	4.38	5.5909	71.19	5.21
90 %	34.3636	56.50	2.92	8.6818	70.63	5.10
95 %	38.4091	58.22	3.02	15.0909	70.25	5.10

Tabla 46: Comparación del algoritmo fuzzypca con los datos Tennessee.

FRPCA1						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	50.7273	54.14	4.27	18.4545	64.21	5.31
65 %	51.0000	54.23	4.48	13.1364	64.16	5.52
70 %	50.7273	55.96	4.38	14.1364	64.40	4.90
75 %	51.0000	55.70	4.79	11.3182	65.33	4.90
80 %	40.8636	56.46	4.27	10.5909	65.25	4.38
85 %	42.4091	56.71	4.48	10.8182	66.19	5.10
90 %	37.2273	57.00	3.85	12.1818	67.21	4.79
95 %	38.2727	58.78	4.38	17.3182	67.40	6.04

Tabla 47: Comparación del algoritmo FRPCA1 con los datos Tennessee.

FRPCA2						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	50.8182	55.32	5.83	18.0455	63.52	5.00
65 %	52.0455	54.95	5.42	15.7727	63.95	4.48
70 %	50.8182	56.51	5.63	17.5000	63.43	5.00
75 %	52.0455	56.61	5.21	15.9545	64.15	5.21
80 %	48.9545	57.04	4.27	14.5909	64.35	4.69
85 %	57.3636	57.35	5.00	11.0000	65.56	4.90
90 %	39.6364	57.82	4.90	13.7273	66.06	4.90
95 %	40.4545	58.93	4.17	15.4545	66.42	5.31

Tabla 48: Comparación del algoritmo FRPCA2 con los datos Tennessee.

FRPCA3						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	50.8182	55.32	5.83	18.0455	63.52	5.00
65 %	52.0455	54.95	5.42	15.7727	63.95	4.48
70 %	50.8182	56.51	5.63	17.5000	63.43	5.00
75 %	52.0455	56.61	5.21	15.9545	64.15	5.21
80 %	48.9545	57.04	4.27	14.5909	64.35	4.69
85 %	57.3636	57.35	5.00	11.0000	65.56	4.90
90 %	39.6364	57.82	4.90	13.7273	66.06	4.90
95 %	40.4545	58.93	4.17	15.4545	66.42	5.31

Tabla 49: Comparación del algoritmo FRPCA3 con los datos Tennessee.

nnFRPCA3						
	T2			Q		
% Varianza	Tiempo Detección	% Detección	% Falsas alarmas	Tiempo Detección	% Detección	% Falsas alarmas
60 %	42.9091	56.04	5.94	13.2727	64.86	5.10
65 %	42.5455	58.17	5.00	17.7273	64.26	5.00
70 %	42.9091	57.51	5.10	11.4091	64.24	4.48
75 %	42.5455	57.64	4.58	14.0909	64.57	5.00
80 %	39.4091	58.48	4.58	14.9545	64.45	3.75
85 %	43.5909	58.53	5.00	13.0000	64.62	4.27
90 %	38.5909	59.00	3.65	15.0909	65.09	4.48
95 %	34.8636	59.93	3.44	20.6818	66.57	5.00

Tabla 50: Comparación del algoritmo nnFRPCA3 con los datos Tennessee.

7.7. Conclusiones de la Detección de Fallos basado en Fuzzy PCA

Atendiendo a los criterios de comparación expuestos en el apartado correspondiente, se presentan las tablas de comparación, las cuales se pueden encontrar en el anexo I, en las que se muestran las comparaciones de los distintos algoritmos mediante los criterios de comparación seleccionados para cada conjunto de datos empleado.

También se muestran los diagramas de caja en las Figuras 17, 18, 19 y 20 con el comportamiento de los distintos algoritmos respecto a los distintos conjuntos de datos y cantidad de variabilidad escogida.

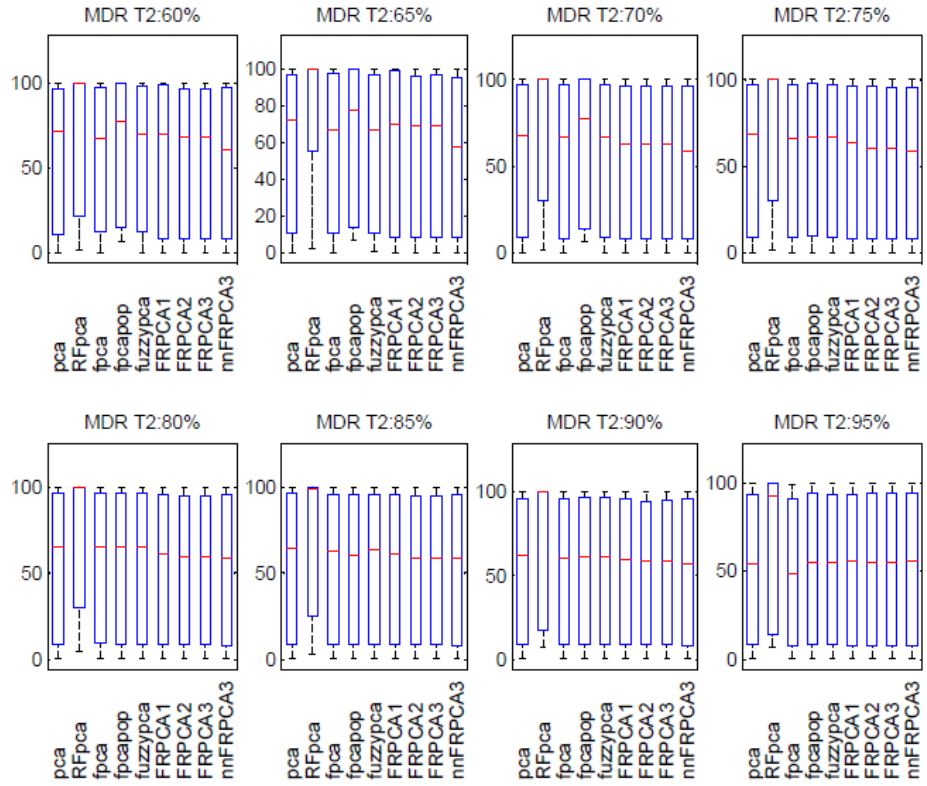

Figura 17: Diagrama de caja del ratio de detecciones perdidas para el índice T2 sin outliers.

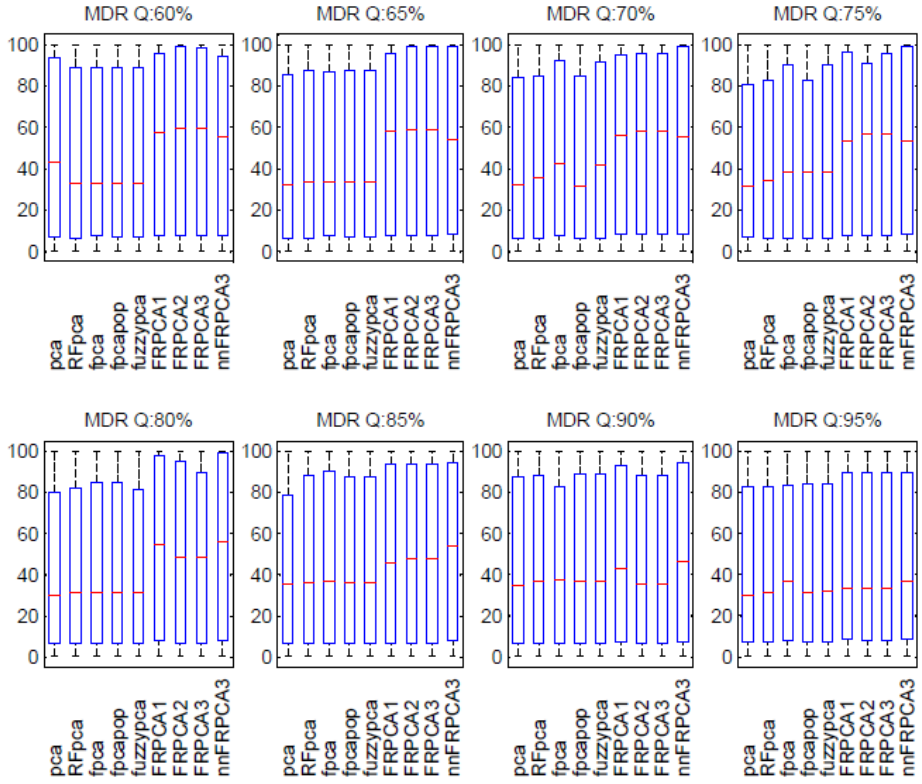

Figura 18: Diagrama de caja del ratio de detecciones perdidas para el índice Q sin outliers.

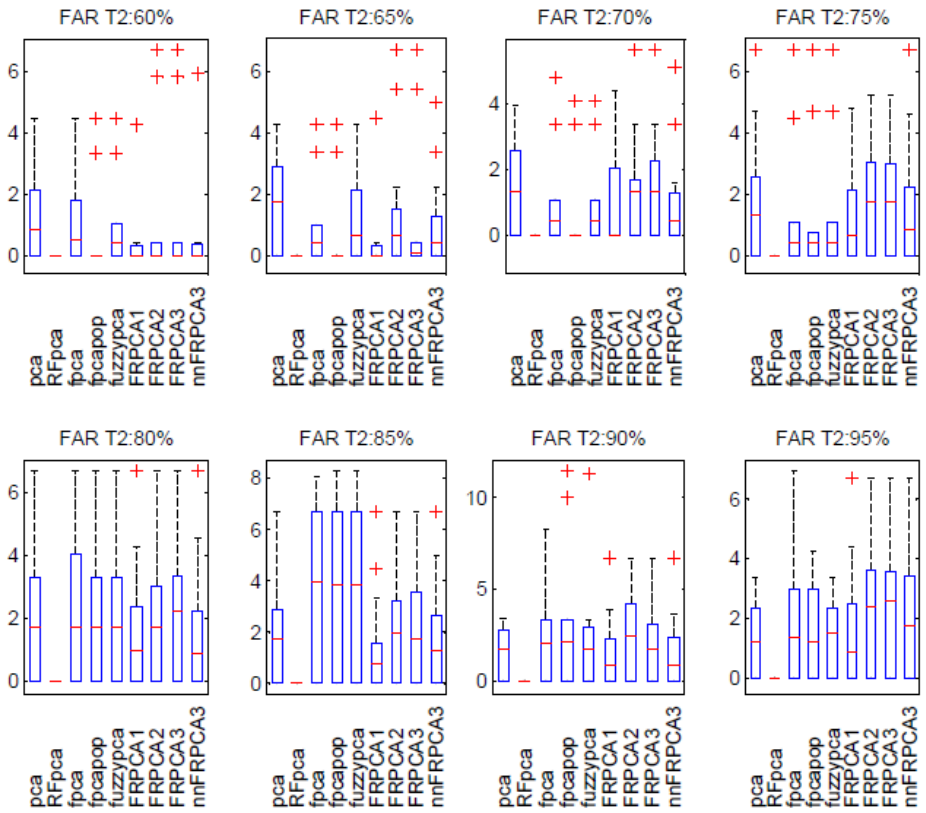

Figura 19: Diagrama de caja del ratio de falsas alarmas para el índice T2 sin outliers.

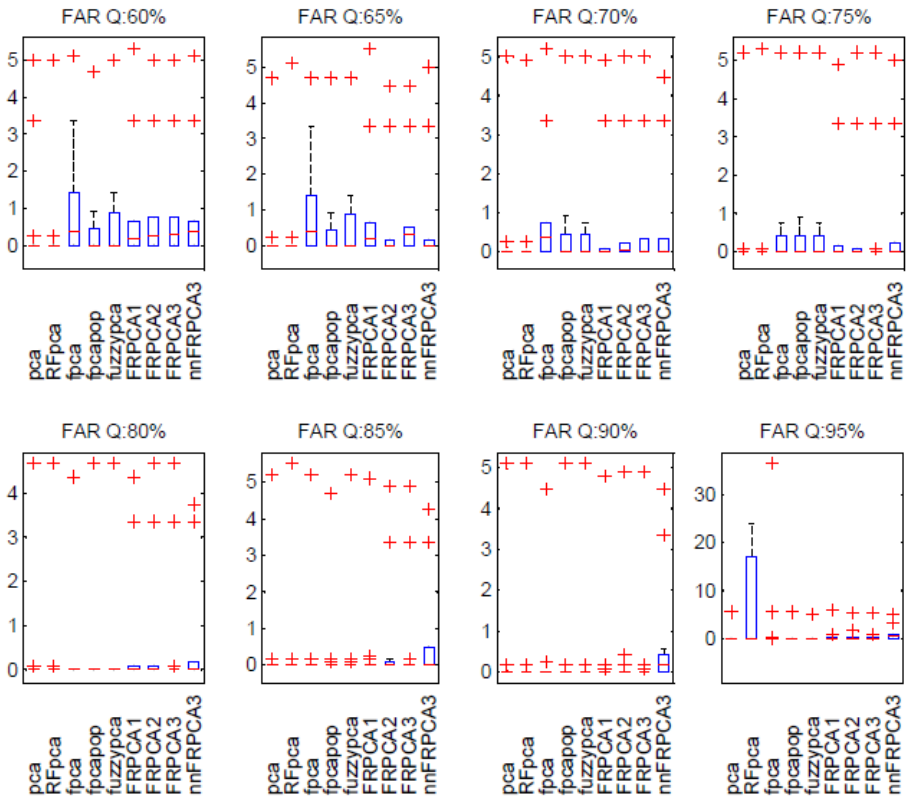

Figura 20: Diagrama de caja del ratio de falsas alarmas para el índice Q sin outliers.

A continuación se muestran los diagramas de caja en las Figuras 21, 22, 23 y 24 con el comportamiento de los distintos algoritmos con datos con outliers.

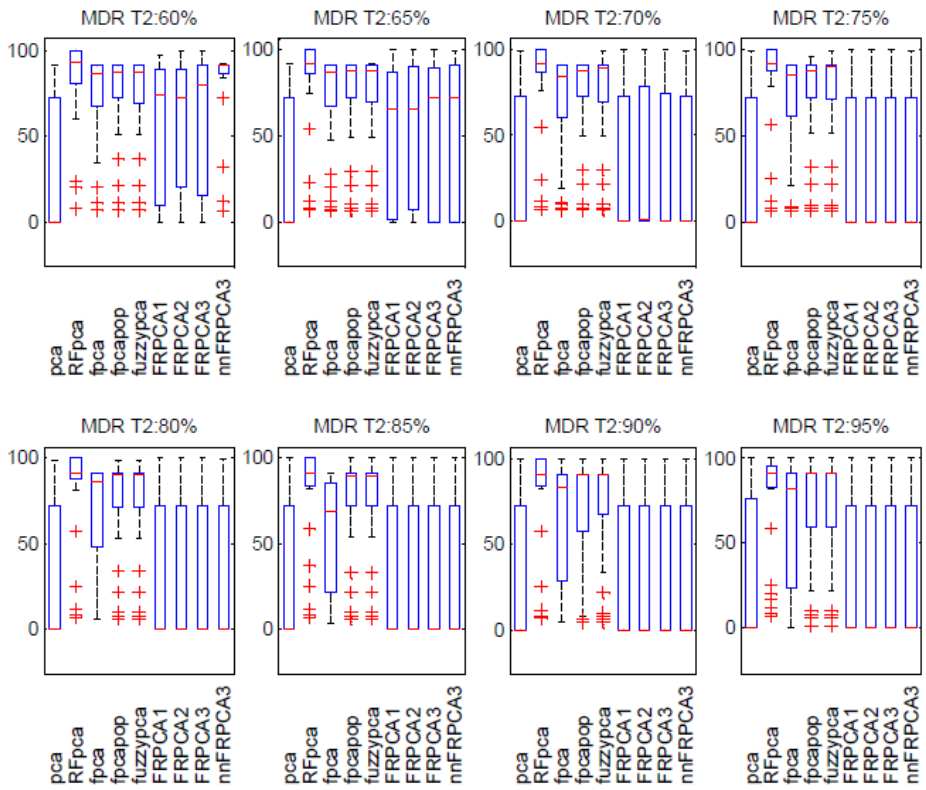

Figura 21: Diagrama de caja del ratio de detecciones perdidas para el índice T2 con outliers.

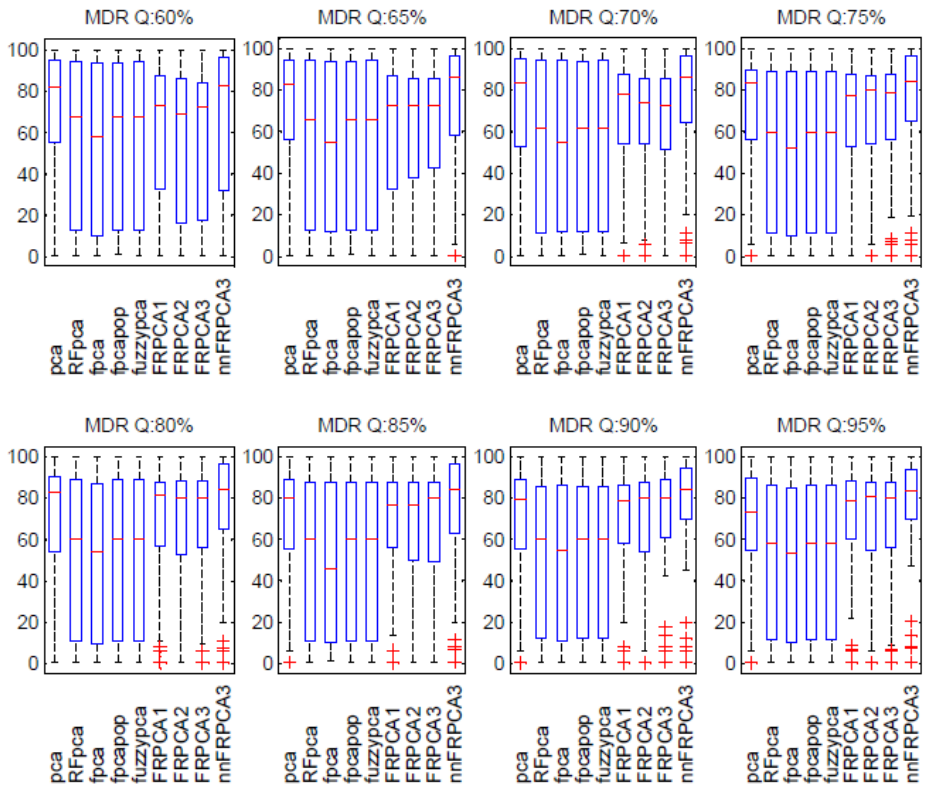

Figura 22: Diagrama de caja del ratio de detecciones perdidas para el índice Q con outliers.

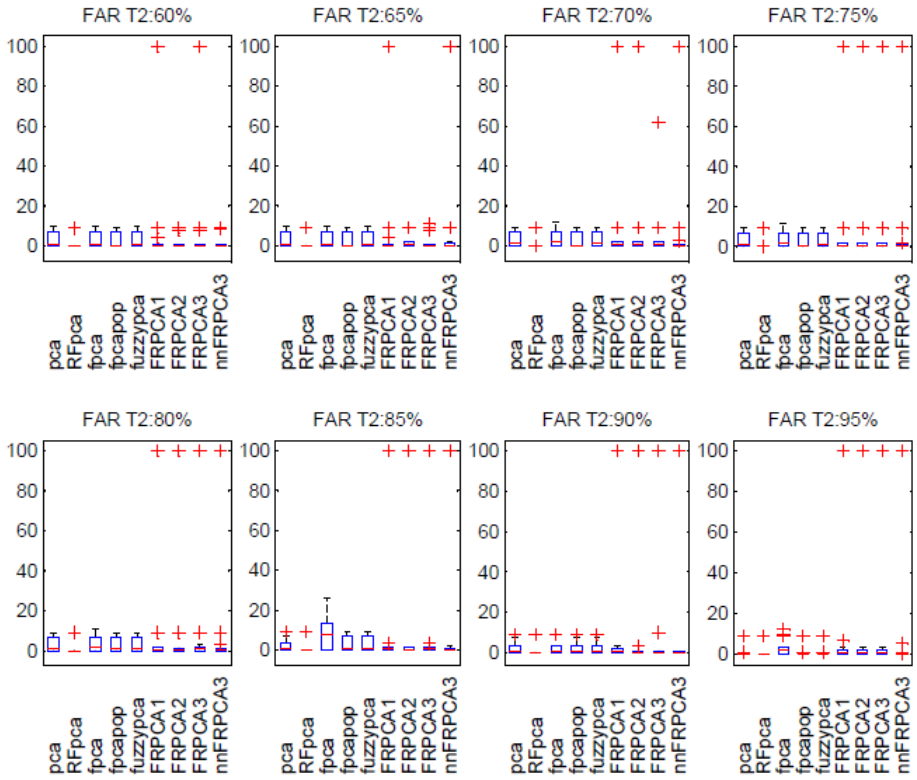

Figura 23: Diagrama de caja del ratio de falsas alarmas para el índice T2 con outliers.

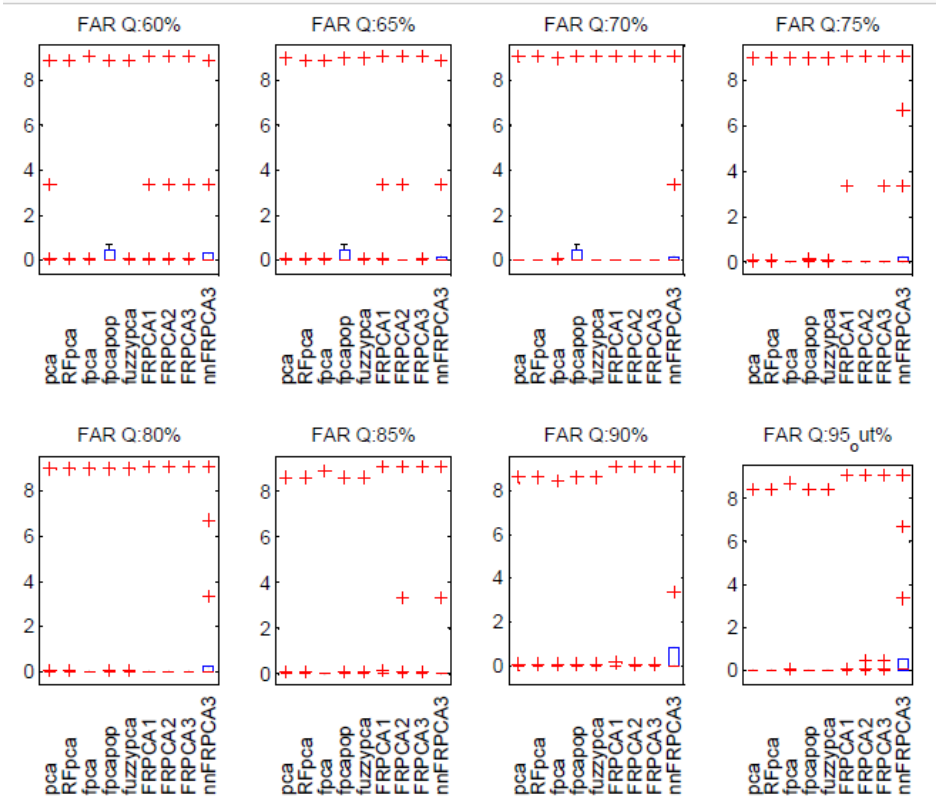

Figura 24: Diagrama de caja del ratio de falsas alarmas para el índice Q con outliers.

A la luz de estos resultados se puede afirmar que no existe un mejor algoritmo para todos los conjuntos de datos en cuanto a detección de fallos se refiere, aunque si que es posible indicar que el algoritmo *RFpca* tiene el peor comportamiento posible al no detectar la mayoría de fallos y detectarlos tarde, si los detecta, independientemente de la cantidad de variabilidad, si son datos con o sin outliers o el índice que se emplee como se puede ver en los diagramas de caja (figuras 17-24).

Por otro lado se aprecia una diferencia general esperable, ya que el índice Q pierde capacidad de detección a medida que aumentamos la variabilidad (figuras 18 y 22) mientras el índice T2 aumenta su capacidad de detección (figuras 17 y 17). Esto se debe a la naturaleza de estos índices, puesto que

T2 opera con el espacio reconstruido, mientras que Q opera con el espacio residual.

Lo ideal en un algoritmo sería que tuviera un FAR bajo y un MDR también bajo, para que no perdiera una cantidad elevada de fallos y no se produjeran falsas alarmas en cantidades excesivas. Bajo estas consideraciones se puede decir que el índice Q tiene un comportamiento considerablemente mejor que el índice T2, al menos para este rango de variabilidad escogido. En cuanto a los algoritmos se puede decir que *pca, fpca, fpcapop y fuzzypca* tienen un mejor comportamiento que los algoritmos RFpca, FRPCA1, FRPCA2, FRPCA3 y nnFRPCA3.

Por otra parte se puede apreciar la influencia de los outliers en los ratios de detecciones perdidas, siendo estas mayores en presencia de outliers, y en los ratios de falsas alarmas, que parecen disminuir en presencia de outliers.

Otra cuestión a considerar es el tiempo que tardan los algoritmos en detectar la ocurrencia del fallo. Para los conjuntos de datos Resultados_OK de la planta BSM2, Resultados_sensores_OK de la planta BSM2 con sensores y Tennessee00 de la planta de Tennessee no se registra ningún fallo puesto que no se produce ninguno. En cuanto al resto de conjuntos como norma general se observa que el índice Q detecta antes conforme disminuye la variabilidad, mientras que el índice T2 tiene una detección más tardía conforme disminuye la variabilidad. Se observa también que generalmente el índice Q tiene una detección más temprana que el índice T2, como se ha podido observar en las tablas 15-50. En cuanto al tiempo de detección de fallos de los distintos algoritmos, se repite el comportamiento del párrafo anterior.

En la siguiente tabla se encuentra un resumen de los resultados, donde:

BSM2 es el conjunto de datos de la planta Benchmark Simulation Model no. 2[2].

BSM2sen es el conjunto de datos BSM2 pero captado a través de sensores.

Epanet es el conjunto de datos correspondiente a la red de distribución de aguas modelada mediante la herramienta de mismo nombre.

Tennessee es el conjunto de datos correspondiente a la planta planteada por Downs y Vogel [5].

T. Det es el tiempo de detección, en esta fila se muestran los algoritmos y su valor de variabilidad en los que el tiempo de detección ha sido el menor.

% Det es el porcentaje de detección, en esta fila se muestran los algoritmos y su valor de variabilidad para los que el porcentaje de detección ha sido máximo.

F. alarm es el porcentaje de falsas alarmas, en esta fila se muestran los algoritmos y su valor de variabilidad para los cuales el porcentaje de falsas alarmas ha sido mínimo.

8. Diagnóstico de Fallos basado en Fuzzy PCA

Aquí se explicarán el marco teórico, la metodología empleada y los resultados obtenidos en el diagnóstico de fallos.

8.1. Diagnóstico de Fallos basado en PCA

Según Alcala y Qin(2009)[1], se detecta un fallo cuando uno o más índices superan su límite de control. Los gráficos de contribución se basan en que las variables con contribuciones mayores a que el índice detecte un fallo suelen ser las causas del mismo. Las gráficas de contribución se construyen determinando la contribución de cada variable al cálculo del índice de detección de fallo. Para poder calcular esas contribuciones, primero se debe poner de manifiesto que las expresiones de los índices atienden a la siguiente forma cuadrática:

$$index(x) = x^T M x = \|x\|_M^2 \tag{23}$$

Donde M viene dada por la Tabla 52 para cada índice.
La ecuación anterior puede expresarse como:

$$index(x) = x^T M x = \|M^{\frac{1}{2}} x\|^2 = \sum_{i=1}^{n} \left(\xi_i^T M^{\frac{1}{2}} x\right)^2 = \sum_{i=1}^{n} c_i^{index} \tag{24}$$

Donde

$$c_i^{index} = \left(\xi_i^T M^{\frac{1}{2}} x\right)^2 \tag{25}$$

es la contribución de la variable x_i al Index(x). Aquí ξ_i es la columna i-ésima de la matriz identidad y la dirección de x_i; por ejemplo, en un sistema de 5 sensores, la dirección del sensor x_3 es:

$$\xi_3 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

8.1.1. Contribución Q

La contribución de variable para el índice Q se obtiene sustituyendo $M = \tilde{C}$

$$c_i^Q = \left(\xi_i^T \tilde{C} x\right)^2 = \tilde{x}_i^2 \tag{26}$$

	T2				Q			
	BSM2	BSM2sen	Epanet	Tennessee	BSM2	BSM2sen	Epanet	Tennessee
T. Det	FRPCA1 95 %	fpca 95 %	pca 90 % mnFRPCA3 90 / 95 % mnFRPCA3 95 %	fpca 90 %	pca 80 % fpca 80 % fpcapop 80 % fuzzypca 80 %	pca 85 / 90 % fpcapop 85 / 90 % fuzzypca 85 / 90 %	todos salvo mnFRPCA3 80 / 85 % FRPCA1 80 %	pca 85 % RFpca 85 % fpca 85 % fpcapop 85 % fuzzypca 85 %
% Det	fpca 85 %	fuzzypca 95 %	pca 95 % fpcapop 95 % fuzzypca 95 %	mnFRPCA3 95 %	fpca 80 %	pca 85 / 90 % fpcapop 85 / 90 % fuzzypca 85 / 90 %	todos menos mnFRPCA3 80 / 85 % FRPCA1 80 %	pca 85 %
F. alarm	FRPCA1 85 %	FRPCA1 95 %	pca 95 % FRPCA2 80 % FRPCA3 80 %	RFpca 80 / 85 / 90 / 95 %	pca 80 / 85 / 90 / 95 % fpca 80 / 90 % fpcapop 80 / 90 / 95 % fuzzypca 80 / 90 / 95 % FRPCA3 80 / 85 %	pca 95 % RFpca 95 % fpca 80 / 85 % fpcapop 80 / 95 % fuzzypca 95 %	pca 80 / 85 / 90 / 95 % RFpca 80 / 85 / 90 / 95 % fpca 90 / 95 % fpcapop 80 / 85 / 90 / 95 % fuzzypca 80 / 85 / 90 / 95 % FRPCA1 95 % FRPCA2 90 / 95 % FRPCA3 90 / 95 %	mnFRPCA3 80 %

Tabla 51: Resumen de resultados.

Index	Q	T^2
M	$\tilde{C}$	D

Tabla 52: Valores de M para la forma general de los índices.

Donde:

$$\tilde{C} = \tilde{P}\tilde{P}^T \tag{27}$$

$\tilde{P}$ es la matriz de *loadings* residuales.

Otra forma de ver esta contribución es como el residuo $r = x(I - PP^T)$ en el/los instante/s de fallo, representando un diagrama de contribución para cada instante de fallo.

8.1.2. Contribución T^2

La contribución de variable para el índice T^2 se obtiene sustituyendo $M = D$

$$c_i^{T^2} = \left(\xi_i^T D^{\frac{1}{2}} x\right)^2 \tag{28}$$

Donde:

$$D = P\Lambda^{-1}P^T \tag{29}$$

.

8.2. Diagnóstico de Fallos en EDAR, Red de Distribución de Agua y Reactor Químico

Las plantas de tratamiento de aguas residuales son complejas, compuestas de varios elementos todos conectados entre sí por medio de tuberias. Esto hace dificil detectar cual es la raiz de un fallo y podría emplearse mucho tiempo en diagnosticarlo.

Al emplear Fuzzy PCA's para el diagnóstico de fallos se espera que se consiga un diagnóstico correcto de forma rápida y automatica.

En cuanto a las redes de distribución de agua, la complejidad de las mismas está ligada a su tamaño. En una ciudad pequeña se podría hablar facilmente de más de 1000 tramos de tuberías con sus correspondientes nodos. Puede pensarse que el tiempo requerido para encontrar el fallo tiende a ser elevado.

Por ello, con la implantación de Fuzzy PCA's se estima que se conseguirá un diagnóstico temprano y correcto desde la detección del fallo.

Por último, en los reactores químicos se tiene que un mal diagnóstico, ya sea este por el tiempo empleado en el, o por ser erroneo puede ocasionar graves daños, tanto personales como materiales.

En este caso, la aplicación de Fuzzy PCA pretende dar respuesta al problema de diagnóstico para evitar los posibles problemas relacionados.

8.3. Metodología Experimental

En esta sección se describe la metodología que se emplea en la parte del trabajo correspondiente a la diagnóstico de fallos a la hora de realizar los cálculos necesarios para la obtención de los resultados:

En primer lugar se han escogido los datos que se pueden encontrar en la sección 7.5, los cuales se han sometido a un preprocesamiento, el cual consta de una normalización de los datos en media 0 y desviación 1 para que todas las variables dispongan de la misma relevancia a la hora del cálculo, puesto que si una variable de orden elevado se mantiene constante mientras que otra de orden menor varía en gran medida, no tienen la misma relevancia en los resultados, pero en el cálculo se impondría la variable de mayor orden pese a permanecer constante. Por otra parte se han eliminado todas las variables que permanecían constantes o cerca de serlo atendiendo a la desviación de cada variable ($\sigma < 0{,}001$).

En segundo lugar, se han empleado los algoritmos escogidos para el tratamiento de los datos normalizados de cada conjunto de datos con el fin de obtener de estos los resultados de diagnóstico de fallos de cada algoritmo para cada conjunto de datos y porcentaje de variabilidad, el cual se puede encontrar en la sección 7.6, discriminando en conjuntos de datos con y sin

outliers. Estos resultados se han recogido en tablas y gráficas que pueden encontrarse en el anexo II. Los algoritmos difusos, como ya se ha dicho, se explican en la sección 2.3.

En tercer lugar, se han aplicado el criterio de comparación descrito en la sección 8.4. Una vez aplicado este criterio se han recogido en cuatro tablas, organizados para cada algoritmo y discriminando entre los índices empleados y datos con o sin outliers, que pueden encontrarse en la sección 8.6 correspondiente a las conclusiones de diagnóstico de fallos. Por otra parte también se ha realizado un estudio de modas de diagnóstico de fallos para cada algoritmo y se han realizado histogramas con las tendencias de diagnóstico para cada algoritmo y conjunto de datos, discriminando entre los índices empleados y datos con y sin outliers. Estos histogramas se pueden encontrar en el anexo II. y el estudio de modas de diagnóstico se puede encontrar en la sección 8.6 correspondiente a las conclusiones de diagnóstico de fallos, organizados de forma análoga a las tablas de la aplicación del criterio comparativo.

8.4. Criterios empleados

Para la sección de diagnóstico de fallos sólo ha sido seleccionado un criterio de comparación:

8.4.1. Ratio de diagnóstico correcto respecto a detección correcta

Este criterio es utilizado por Alcalá y Qin (2009) [1] y se puede expresar como el porcentaje de observaciones que disponen de un diagnóstico correcto respecto a todas las detecciones correctas que ha encontrado el índice empleado.

$$M(\,\%) = \frac{N^o\text{de diagnósticos correctos}}{N^o\text{de detecciones correctas}} \tag{30}$$

Donde M puede ser cualquiera de los índices empleados.

8.5. Análisis de los Resultados obtenidos en Diagnóstico de Fallos

Los resultados obtenidos se basan en los correspondientes a detección de fallo, puesto que no tiene sentido realizar un diagnóstico de fallos si no se ha producido ninguno, por lo que se ha prescindido de los datasets que no presentaban ningún fallo para el cálculo.

Los resultados se han obtenido, análogamente a los de detección, para distintas tasas de variabilidad explicada por los modelos PCA:

- 60 % de la variabilidad.
- 65 % de la variabilidad.
- 70 % de la variabilidad.
- 75 % de la variabilidad.
- 80 % de la variabilidad.
- 85 % de la variabilidad.
- 90 % de la variabilidad.
- 95 % de la variabilidad.

Los resultados obtenidos se pueden encontrar en el anexo II por cuestiones de espacio.

8.6. Conclusiones del Diagnóstico de Fallos basado en Fuzzy PCA

Se han realizado estudios estadísticos de moda sobre los resultados obtenidos para un análisis de los mismos más sencillo.

La moda se ha realizado para los resultados obtenidos por cada algoritmo, a lo largo de las distintas variabilidades estudiadas y para cada conjunto de datos, discriminando datos sin outliers y datos con outliers.

Datos	PCA	RFpca	fpca	fpcapop	fuzzypca	FRPCA1	FRPCA2	FRPCA3	nnFRPCA3
BSMcaupry	6	NaN	18	5	21	22	6	5	12
BSMO2	18	NaN	5	18	5	6	5	5	5
BSMkla	15	NaN	22	18	22	12	5	5	5
BSMSalk	1	1	1	5	1	5	5	5	1
BSMsencaupry	1	NaN	1	1	1	11	11	11	11
BSMsenO2	6	NaN	6	6	6	3	1	1	6
BSMsenkla	15	NaN	6	15	8	13	2	2	6
BSMsenSalk	1	NaN	4	1	1	1	4	4	15
EPAbom	633	NaN	633	633	633	521	NaN	NaN	NaN
EPAfug	NaN	NaN	380	NaN	NaN	NaN	NaN	NaN	NaN
EPAcon	615	623	528	615	615	520	586	586	517
TEN1	10	20	41	37	37	20	20	30	34
TEN2	10	47	10	10	10	10	28	10	10
TEN3	34	NaN	27	34	34	3	41	41	22
TEN4	34	15	34	34	34	34	45	15	34
TEN5	34	22	34	34	34	34	45	15	34
TEN6	28	20	30	39	28	27	4	4	28
TEN7	34	17	39	26	26	14	14	14	14
TEN8	10	23	10	10	10	40	40	10	10
TEN9	20	NaN	3	3	3	23	8	45	1
TEN10	4	20	10	10	10	17	17	17	47
TEN11	24	9	40	39	39	34	34	24	17
TEN12	40	22	40	37	37	10	10	10	36
TEN13	6	40	41	36	26	30	17	17	6
TEN14	6	51	22	36	36	34	6	6	6
TEN15	5	NaN	10	39	12	10	10	10	25
TEN16	29	NaN	1	1	1	38	20	29	38
TEN17	30	21	30	30	30	30	30	10	10
TEN18	39	22	41	41	41	39	41	41	27
TEN19	5	NaN	32	32	32	24	5	5	25
TEN20	45	46	5	3	26	10	5	34	10
TEN21	22	8	22	42	22	39	37	42	4

Tabla 53: Estudio de modas de contribución para el índice de Hotelling con datos sin outliers.

Datos	PCA	RFpca	fpca	fpcapop	fuzzypca	FRPCA1	FRPCA2	FRPCA3	nnFRPCA3
BSMcaupry	4	4	4	22	4	2	22	5	22
BSMO2	5	5	13	18	13	6	18	18	4
BSMkla	NaN	2	5	4	5	20	4	4	20
BSMSalk	4	22	5	6	22	18	5	5	4
BSMsencaupry	1	1	4	5	5	6	6	6	6
BSMsenO2	1	1	6	6	6	6	6	6	1
BSMsenkla	4	4	2	5	5	4	4	4	4
BSMsenSalk	4	4	4	4	4	4	4	4	4
EPAbom	582	582	582	582	582	562	524	524	511
EPAfug	380	380	380	380	380	380	380	380	380
EPAcon	582	582	582	582	582	511	516	564	582
TEN1	29	29	35	29	29	3	3	3	24
TEN2	30	41	41	41	41	30	33	33	41
TEN3	37	36	29	36	36	29	2	2	36
TEN4	35	35	34	35	35	25	28	28	34
TEN5	35	35	35	35	35	25	28	28	29
TEN6	43	32	27	25	25	27	25	25	27
TEN7	27	29	29	29	29	33	33	33	29
TEN8	38	38	38	38	38	24	24	24	24
TEN9	42	23	23	23	23	23	41	41	25
TEN10	42	28	31	28	28	33	33	33	36
TEN11	29	29	29	29	29	24	42	42	21
TEN12	24	40	25	40	40	33	25	25	37
TEN13	8	22	26	27	27	3	3	3	33
TEN14	29	29	29	29	29	29	43	43	35
TEN15	33	33	33	33	33	33	36	36	36
TEN16	43	43	23	43	43	24	27	27	3
TEN17	23	23	23	23	23	23	11	11	26
TEN18	25	25	25	25	25	25	25	25	28
TEN19	23	29	23	23	23	23	27	27	6
TEN20	27	27	27	27	27	27	27	27	27
TEN21	26	26	25	26	26	26	3	3	26

Tabla 54: Estudio de modas de contribución para el índice de Error de predicción cuadrado con datos sin outliers.

Datos	PCA	RFpca	fpca	fpcapop	fuzzypca	FRPCA1	FRPCA2	FRPCA3	nnFRPCA3
BSMcaupry	8	1	21	8	8	8	9	9	15
BSMO2	4	2	1	6	4	6	1	10	6
BSMkla	21	2	1	18	21	18	18	21	4
BSMSalk	18	7	1	7	18	18	1	1	6
BSMsencaupry	6	NaN	4	6	6	6	6	6	6
BSMsenO2	6	6	6	6	6	6	6	6	6
BSMsenkla	2	NaN	2	2	2	1	1	1	1
BSMsenSalk	8	NaN	8	8	8	8	8	8	13
EPAbom	449	NaN	449	449	449	449	451	449	497
EPAfug	449	497	449	449	449	449	449	449	321
EPAcon	450	380	450	450	450	450	497	450	466
TEN1	32	1	1	1	1	36	14	14	37
TEN2	37	10	2	10	10	14	38	14	37
TEN3	49	12	48	10	10	36	14	14	37
TEN4	39	39	39	39	39	36	14	14	37
TEN5	39	39	1	1	1	36	14	14	9
TEN6	42	37	37	37	37	36	14	14	8
TEN7	31	44	46	52	1	36	38	14	12
TEN8	32	1	1	1	1	36	38	14	37
TEN9	12	52	52	44	52	14	14	14	37
TEN10	48	31	32	31	2	36	14	14	19
TEN11	52	32	47	32	39	14	14	14	19
TEN12	12	44	32	32	44	36	14	14	37
TEN13	52	1	32	50	50	36	38	38	37
TEN14	39	52	52	52	52	36	38	14	15
TEN15	49	12	48	39	39	36	14	14	37
TEN16	1	52	52	52	49	36	14	14	37
TEN17	9	52	10	52	52	36	38	14	37
TEN18	12	1	2	1	1	36	38	14	19
TEN19	1	33	14	10	10	36	38	38	19
TEN20	37	37	10	37	37	36	38	14	37
TEN21	49	31	32	31	50	36	38	14	19

Tabla 55: Estudio de modas de contribución para el índice de Hotelling con datos con outliers.

Datos	PCA	RFpca	fpca	fpcapop	fuzzypca	FRPCA1	FRPCA2	FRPCA3	nnFRPCA3
BSMcaupry	4	4	6	1	4	4	4	4	2
BSMO2	5	5	2	2	5	5	13	13	4
BSMkla	10	10	6	4	10	4	4	4	4
BSMSalk	4	1	2	18	4	2	18	18	2
BSMsencaupry	1	1	5	1	1	18	4	4	8
BSMsenO2	6	6	6	6	6	6	6	6	4
BSMsenkla	1	1	4	1	1	10	16	16	9
BSMsenSalk	4	4	10	4	4	1	16	16	9
EPAbom	533	533	533	533	533	107	281	107	NaN
EPAfug	499	499	499	499	499	108	107	106	NaN
EPAcon	106	106	106	106	106	106	107	106	389
TEN1	27	6	28	6	6	36	26	26	1
TEN2	14	32	25	32	32	36	26	26	37
TEN3	32	27	2	27	27	36	26	26	32
TEN4	49	38	27	38	38	36	26	26	37
TEN5	49	38	33	38	38	36	26	26	52
TEN6	14	32	14	32	32	36	26	26	37
TEN7	8	27	28	32	32	36	26	26	37
TEN8	14	14	36	14	14	36	26	26	48
TEN9	32	27	41	27	27	36	26	26	37
TEN10	13	46	46	46	46	36	26	26	37
TEN11	6	5	38	5	5	36	26	26	37
TEN12	38	8	35	8	8	36	26	26	52
TEN13	28	27	27	27	27	36	26	26	37
TEN14	36	32	36	32	32	36	26	26	48
TEN15	26	5	14	5	5	36	26	26	18
TEN16	13	38	23	38	38	36	26	26	48
TEN17	5	51	51	51	52	36	26	26	37
TEN18	47	25	45	25	25	36	26	26	37
TEN19	6	13	3	32	13	36	26	26	13
TEN20	38	10	14	10	10	36	26	26	37
TEN21	14	49	34	49	49	36	26	26	37

Tabla 56: Estudio de modas de contribución para el índice de Error de predicción cuadrado con datos con outliers.

A primera vista puede apreciarse que dependiendo del índice estadístico empleado se consiguen unos resultados u otros. Esto no carece de sentido puesto que, como ya se ha explicado, cada índice emplea un espacio distinto para su cálculo.

Sin embargo no se dispone de todas las variables causantes de fallo. A

continuación se muestra una tabla en la que se pueden ver las variables causantes de fallo disponibles o de las que se tiene indicios de ser las causantes.

Datos	Variables
BSMcaupry	9
BSMO2	5
BSMkla	4 5 6
BSMSalk	2
EPAbom	147 149 220
EPAfug	226 233 174 232
EPAcon	522 523 594
TEN1	45 4
TEN2	45 4
TEN3	42 2
TEN4	51
TEN5	52
TEN6	44 1
TEN7	45 4
TEN8	23 24 25
TEN9	42 2
TEN10	45 4
TEN11	51
TEN12	52
TEN13	6 7 8 9
TEN14	51
TEN15	52
TEN16	-
TEN17	-
TEN18	-
TEN19	-
TEN20	-
TEN21	45 4

Tabla 57: Variables causantes de los fallos para cada conjunto de datos.

Conviene indicar que en esta tabla 57 no se han mostrado las variables causantes de fallo de los conjuntos de datos correspondientes a BSM2 con sensores, puesto que son los mismos que para su análogo sin sensores.

A raíz de estos resultados podemos interpretar como propuestas de fallo para los conjuntos de datos de Tennessee 16, 17, 18, 19 y 20 que las variables responsables son las siguientes:

Datos	Variables
TEN16	43 1
TEN17	30 23
TEN18	25
TEN19	23
TEN20	27

Tabla 58: Propuesta de variables causantes de los fallos para los datos faltantes.

También se han agrupado los resultados de aplicar el criterio de ratio de diagnóstico correcto por algoritmos, y estos divididos a su vez en cuatro tablas correspondientes a los índices de Hotelling y error de predicción cuadrado sin y con outliers como se puede ver en las tablas 59 - 62 donde Alg son los algoritmos y RDC es el ratio de detección correcta.

Alg	pca	RFpca	fpca	fpcapop	fuzzypca	FRPCA1	FRPCA2	FRPCA3	nnFRPCA3
RDC	0.0311	0.0504	0.0355	0.0055	0.0303	0.0052	0.0313	0.0406	0.0303

Tabla 59: Ratio de diagnósticos correctos para el índice de Hotelling con datos sin outliers.

Alg	pca	RFpca	fpca	fpcapop	fuzzypca	FRPCA1	FRPCA2	FRPCA3	nnFRPCA3
RDC	0.0230	0.0138	0.0048	0.0049	0.0049	0.0000	0.0172	0.0089	0.0135

Tabla 60: Ratio de diagnósticos correctos para el índice de error de predicción cuadrado con datos sin outliers.

Alg	pca	RFpca	fpca	fpcapop	fuzzypca	FRPCA1	FRPCA2	FRPCA3	nnFRPCA3
RDC	0.0000	0.0000	0.0085	0.0130	0.0000	0.0087	0.0175	0.0133	0.0085

Tabla 61: Ratio de diagnósticos correctos para el índice de Hotelling con datos con outliers.

Alg	pca	RFpca	fpca	fpcapop	fuzzypca	FRPCA1	FRPCA2	FRPCA3	nnFRPCA3
RDC	0.0508	0.0340	0.0354	0.0287	0.0340	0.0457	0.0246	0.0287	0.0521

Tabla 62: Ratio de diagnósticos correctos para el índice de error de predicción cuadrado con datos con outliers.

Se llega, pues, a las siguientes conclusiones: Atendiendo al criterio empleado en esta parte, se pueden observar valores muy bajos, por lo que conviene destacar que los valores representados se encuentran en tanto por uno. Aún así los ratios de diagnóstico correcto apenas llegan al 5 %; lo cual es un indicador de que estos algoritmos rara vez aciertan en su diagnóstico pese a que, como se ha visto en la sección de detección de fallo, no tengan muchos problemas a la hora de detectar fallos.

Por esta misma razón, se tiene que considerar que las propuestas de variables causantes de fallo aquí expuestas se basan en datos imprecisos o erróneos y, por lo tanto, es muy posible que no sean ni representativos ni acertados.

En las tablas del estudio de modas se ha podido apreciar una disparidad generalizada a la hora de indicar las variables causantes de los fallos. Esto puede ser debido a una mala implementación de los algoritmos.

9. Conclusiones Generales

Llegados a este punto del trabajo, sólo queda recapitular los resultados obtenidos a lo largo del mismo.
Las conclusiones extraídas de la parte de extracción de variables indicaban que el mejor algoritmo en cuanto a reducir el número de componentes principales es *fuzzypca*, ya que, como se vió en las tablas 3-11 y la tabla resumen 12, *fuzzypca* presenta los mejores resultados en cuanto al criterio de "Componentes Principales para un Umbral de Variabilidad", aunque no en todos los conjuntos de datos de pruebas, pero si en la mayoría; mientras que los algoritmos más robustos son *PCA* y *fpcapop* acorde al criterio de "Robustez: Sensibilidad a Outliers", o lo que es lo mismo, toleran mejor los outliers y el ruido. Esta conclusión se puede contrastar con las tablas mencionadas para la conclusión anterior.

Prosiguiendo con la robustez se ha encontrado un comportamiento que no cumple con las expectativas salvo *fpcapop* y en el caso de Energy Efficiency para *nnFRPCA3*, el cual tiene un comportamiento similar al de PCA clásica, lo que puede estar indicando una mala implementación de los algoritmos, puesto que se esperaba una mejora significativa de estos algoritmos frente a PCA.

En cuanto a recursos computacionales, el algoritmo más rápido es *PCA* o *RFpca* dependiendo del dataset. Esto es así porque el algoritmo PCA está muy depurado en comparación con el resto de algoritmos, por otra parte, los FRPCA1, 2 y 3 trabajan con los valores medios de 10 ejecuciones. Esto explica su tiempo de cómputo elevado.

Entrando ahora a la extracción de variables, se observa que todas las técnicas emplean menos variables que las originales, con lo que se consigue una reducción de la dimensionalidad. Este hecho se aprecia mejor en *fuzzypca*, el algoritmo de Sârbu, C. y Pop, H.,2005 [4], en el cual se emplean menos componentes principales que en el resto de forma general.

Se ha apreciado que en cuanto a número de componentes principales el comportamiento de los algoritmos ha sido el esperado, a excepción de los algoritmos *FRPCA1*, *FRPCA2*, *FRPCA3* y *nnFRPCA3*, que han presentado una mayor necesidad de componentes principales que PCA clásica.

En cuanto a los resultados obtenidos en la detección de fallos se puede afirmar que no existe un mejor algoritmo para todos los conjuntos de datos en cuanto a detección de fallos se refiere, aunque si que es posible indicar que el algoritmo *RFpca* tiene el peor comportamiento posible al no detectar la mayoría de fallos y detectarlos tarde, si los detecta, independientemente de la cantidad de variabilidad, si son datos con o sin outliers o el índice que se emplee como se puede ver en los diagramas de caja (figuras 17-24).

Por otro lado se aprecia una diferencia general esperable, ya que el índice Q pierde capacidad de detección a medida que aumentamos la variabilidad (figuras 18 y 22) mientras el índice T2 aumenta su capacidad de detección (figuras 17 y 17). Esto se debe a la naturaleza de estos índices, puesto que T2 opera con el espacio reconstruido, mientras que Q opera con el espacio residual.

Lo ideal en un algoritmo sería que tuviera un ratio de falsas alarmas bajo y un ratio de detecciones perdidas también bajo, para que no perdiera una cantidad elevada de fallos y no se produjeran falsas alarmas. Bajo estas consideraciones se puede decir que el índice Q tiene un comportamiento considerablemente mejor que el índice T2, al menos para este rango de variabilidad escogido. En cuanto a los algoritmos se puede decir que *pca, fpca, fpcapop y fuzzypca* tienen un mejor comportamiento que los algoritmos RFpca, FRPCA1, FRPCA2, FRPCA3 y nnFRPCA3.

Por otra parte se puede apreciar la influencia de los outliers en los ratios de detecciones perdidas, siendo estas mayores en presencia de outliers, y en los ratios de falsas alarmas, que parecen disminuir en presencia de outliers. Esto parece contradecir lo que se esperaba de estos algoritmos, puesto que supuestamente eran capaces de tolerar mejor los valores perdidos y outliers. Pero como ya se ha visto en la extracción de características la robustez de estos algoritmos deja bastante que desear.

Otra cuestión a considerar es el tiempo que tardan los algoritmos en detectar la ocurrencia del fallo. Para los conjuntos de datos Resultados_OK de la planta BSM2, Resultados_sensores_OK de la planta BSM2 con sensores y Tennessee00 de la planta de Tennessee no se registra ningún fallo puesto que no se produce ninguno. En cuanto al resto de conjuntos como norma general se observa que el índice Q detecta antes conforme disminuye la variabilidad,

mientras que el índice T2 tiene una detección más tardía conforme disminuye la variabilidad. Se observa también que generalmente el índice Q tiene una detección más temprana que el índice T2, como se ha podido observar en las tablas 15-50. En cuanto al tiempo de detección de fallos de los distintos algoritmos, se repite el comportamiento del párrafo anterior.
En cuanto al diagnóstico de fallos se pueden observar valores muy bajos en el ratio de diagnósticos correctos puesto que apenas llegan al 5 % como se puede apreciar en las tablas 59-62; lo cual es un indicador de que estos algoritmos rara vez aciertan en su diagnóstico pese a que, como se ha visto en la sección de detección de fallo, no tengan muchos problemas a la hora de detectar fallos.

Por esta misma razón, se tiene que considerar que las propuestas de variables causantes de fallo de la tabla 58 se basan en datos imprecisos o erróneos y, por lo tanto, es muy posible que no sean ni representativos ni acertados.

En las tablas del estudio de modas (53-56) se ha podido apreciar una disparidad generalizada a la hora de indicar las variables causantes de los fallos. Esto puede ser debido a una mala implementación de los algoritmos.

Todo parece indicar que los algoritmos que se han puesto a estudio no cumplen con las expectativas puestas en ellos, lo cual no debería ser así, se sospecha como causa de esto que estos algoritmos hayan sido mal implementados y que ello arrastre un comportamiento tan poco deseado.

Referencias

[1] Carlos F. Alcala and S. Joe Qin. Reconstruction-based contribution for process monitoring. *Automatica*, 45(7):1593 – 1600, 2009.

[2] J. Alex, L. Benedetti, J. Copp, K.V. Gernaey, U. Jeppsson, I. Nopens, M.N. Pons, C. Rosen, J.P. Steyer, and P. Vanrolleghem. *Benchmark Simulation Model no. 2 (BSM2)*, 2008.

[3] Sang Wook Choi, Changkyu Lee, Jong-Min Lee, Jin Hyun Park, and In-Beum Lee. Fault detection and identification of nonlinear processes based on kernel pca. *Chemometrics and intelligent laboratory systems*, 75(1):55–67, 2005.

[4] Thomas R Cundari, Costel Sârbu, and Horia F Pop. Robust fuzzy principal component analysis (fpca). a comparative study concerning interaction of carbon-hydrogen bonds with molybdenum-oxo bonds. *Journal of chemical information and computer sciences*, 42(6):1363–1369, 2002.

[5] J.J. Downs and E.F. Vogel. A plant-wide industrial process control problem. *Computers & Chemical Engineering*, 17(3):245 – 255, 1993. Industrial challenge problems in process control.

[6] Gyeongyong Heo, Paul Gader, and Hichem Frigui. Rkf-pca: Robust kernel fuzzy {PCA}. *Neural Networks*, 22(5 - 6):642 – 650, 2009. Advances in Neural Networks Research: {IJCNN2009} 2009 International Joint Conference on Neural Networks.

[7] Harold Hotelling. British statistics and statisticians today. *Journal of the American Statistical Association*, 25(170):186–190, 1930.

[8] T. Kourti and J.F. MacGregor. Multivariate spc methods for process and product monitoring. *Journal of Quality Technology*, 28(4):409–428, October 1996.

[9] Wenfu Ku, Robert H. Storer, and Christos Georgakis. Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics and intelligent laboratory systems*, 1(30):179–196, 1995.

[10] S Lane, EB Martin, AJ Morris, and P Gower. Application of exponentially weighted principal component analysis for the monitoring of

a polymer film manufacturing process. *Transactions of the Institute of Measurement and Control*, 25(1):17–35, 2003.

[11] C.K. Lau, Kaushik Ghosh, M.A. Hussain, and C.R. Che Hassan. Fault diagnosis of tennessee eastman process with multi-scale {PCA} and {ANFIS}. *Chemometrics and Intelligent Laboratory Systems*, 120:1 – 14, 2013.

[12] M. Lichman. UCI machine learning repository, 2013.

[13] Jialin Liu and Ding-Sou Chen. Fault isolation using modified contribution plots. *Computers & Chemical Engineering*, 61:9 – 19, 2014.

[14] Pasi Luukka. A new nonlinear fuzzy robust pca algorithm and similarity classifier in classification of medical data sets. *International Journal of Fuzzy Systems*, 13(3):153–162, 2011.

[15] Diego García Álvarez. *Monitoring, Fault Detection and Estimation in Processes using Multivariate Statistical Techniques*. PhD thesis, Escuela de Ingenierías Industriales, Universidad de Valladolid, 2013.

[16] J.F. MacGregor and T. Kourti. Statistical process control of multivariate process. *Control Eng. Practice*, 3(3):403–414, 1995.

[17] Manish Misra, H Henry Yue, S Joe Qin, and Cheng Ling. Multivariate process monitoring and fault diagnosis by multi-scale pca. *Computers & Chemical Engineering*, 26(9):1281–1293, 2002.

[18] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[19] H Pop. Principal components analysis based on a fuzzy sets approach. *Mij*, 1(2):1, 2001.

[20] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998.

[21] Lindsay I Smith. A tutorial on principal components analysis. *Cornell University, USA*, 51(52):65, 2002.

[22] C. Sârbu and H.F. Pop. Principal component analysis versus fuzzy principal component analysis: A case study: the quality of danube water (1985-1996). *Talanta*, 65(5):1215 – 1220, 2005. Optical waveguide analysis.

[23] Doan X Tien, K-W Lim, and Liu Jun. Comparative study of pca approaches in process monitoring and fault detection. In *Industrial Electronics Society, 2004. IECON 2004. 30th Annual Conference of IEEE*, volume 3, pages 2594–2599. IEEE, 2004.

[24] Venkat Venkatasubramanian, Raghunathan Rengaswamy, and Surya N Kavuri. A review of process fault detection and diagnosis: Part ii: Qualitative models and search strategies. *Computers & Chemical Engineering*, 27(3):313 – 326, 2003.

[25] Venkat Venkatasubramanian, Raghunathan Rengaswamy, Surya N. Kavuri, and Kewen Yin. A review of process fault detection and diagnosis: Part iii: Process history based methods. *Computers & Chemical Engineering*, 27(3):327 – 346, 2003.

[26] Venkat Venkatasubramanian, Raghunathan Rengaswamy, Kewen Yin, and Surya N. Kavuri. A review of process fault detection and diagnosis: Part i: Quantitative model-based methods. *Computers & Chemical Engineering*, 27(3):293 – 311, 2003.

[27] D. Wang and J.A. Romagnoli. Robust multi-scale principal components analysis with applications to process monitoring. *Journal of Process Control*, 15(8):869 – 882, 2005.

[28] Tai-Ning Yang and Sheng-De Wang. Robust algorithms for principal component analysis. *Pattern Recognition Letters*, 20(9):927–933, 1999.

[29] Seongkyu Yoon and John F MacGregor. Fault diagnosis with multivariate statistical models part i: using steady state fault signatures. *Journal of process control*, 11(4):387–400, 2001.

[30] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[31] Lotfi A Zadeh. Outline of a new approach to the analysis of complex systems and decision processes. *Systems, Man and Cybernetics, IEEE Transactions on*, (1):28–44, 1973.

[32] David Zumoften and Marta Basualdo. From large chemical plant data to fault diagnosis integrated to decentralized fault tolerant control: pulp mill process application. *Industrial & Engineering Chemistry Research*, 4(47):1201 – 1220, 2008.