XVII Simposio CEA de Control Inteligente
27-29 de junio de 2022, León

Detección de ataques de inyección de datos falsos en turbinas eólicas mediante sistemas neuro-borrosos

Chicaiza, W. D.[a], Dorado, F.[a], Rodríguez, F.[b], Gómez, J.[a], Escaño, J.M.[a,*]

[a] *Dept. de Ing. de Sistemas y Automatica, Universidad de Sevilla, Camino de los Descubrimientos, s/n, 41092, Sevilla, España.*
[b] *Departamento de Matemáticas Aplicadas II, Universidad de Sevilla, Camino de los Descubrimientos, s/n, 41092, Sevilla, España.*

To cite this article: Chicaiza, W. D., Dorado, F., Rodríguez, F., Gómez, J., Escaño, J.M. 2022. Detección de ataques de inyección de datos falsos en turbinas eólicas mediante sistemas neuro-borrosos. XVII Simposio CEA de Control Inteligente.

Resumen

Uno de los ataques populares en microrredes y sistemas industriales es el conocido como ataque por inyección de datos falsos. Estos constituyen inserciones maliciosas de datos falsos como mediciones de sensores en un sistema ciberfísico, con el fin de llevar al sistema a tomar una acción incorrecta. Los ataques de inyección de datos falsos no atacan los componentes informáticos o de red de los sistemas ciberfísicos, sino la interfaz entre la parte física y la cibernética. En este trabajo se presenta un sistema de detección de datos falsos en la potencia activa y reactiva simultáneamente, provocados por un ciberataque. El sistema se basa en clasificadores neuro-borrosos combinados con proyecciones de datos de entrada sobre espacios reducidos, mediante análisis de componentes principales.

Palabras clave: Ciberataque en la generación eléctrica, Detección de datos falsos, Sistemas Neuro-borroso, Análisis de componentes principales.

False data injection attack detection on wind turbine using neurofuzzy systems

Abstract

One of the popular attacks on microgrids and industrial systems is known as a false data injection attack. These are malicious insertions of fake data such as sensor measurements into a cyber-physical system, in order to lead the system to take an incorrect action. Fake data injection attacks do not attack the computer or network components of cyber-physical systems, but the interface between the physical and the cyber part. This paper presents a system for detecting false data in active and reactive power simultaneously, caused by a cyber-attack. The system is based on neuro-fuzzy classifiers combined with input data projections on reduced spaces, by means of principal component analysis.

Keywords: Cyber-attack in power generation, Fake data detection, Neurofuzzy Systems, Principal Component Analysis.

1. Introducción

Las políticas de transición energética han hecho de las energías renovables, principalmente termosolar, fotovoltaica y eólica, una nueva fuente esencial del mix energético actual, con capacidades de producción en progresivo aumento. (IEA, 2021). Según el informe de la IEA el 12 % de la energía mundial provino de fuentes renovables en el año 2020, representando el 28 % del total de la producción eléctrica. Estas cifras están previsto que se incrementen progresivamente hasta el 26 % y el 60 % respectivamente para el año 2050. Uno de los principales inconvenientes de estas tecnologías, y lo que hace difícil su avance a gran escala es la propia variabilidad de la generación, debiendo adoptarse estrategias de almacenamiento, transporte y distribución inteligente en el seno de smart grids cada vez más avanzadas. Si bien la comunidad científica está trabajando activamente en esta dirección (Bordons et al., 2020) aún quedan pendientes retos que deben ser resueltos. Precisamente los beneficios que aportan esta integración y digitalización en el

*Autor para correspondencia: jescano@us.es

tratamiento de la energía conlleva de forma inherente el riesgo de ciberataques que pueden conducir a la pérdida del control sobre los dispositivos y procesos relacionados con los sistemas eléctricos, causando daño físico e interrupción del servicio. Aunque la protección total de estos ciberataques no es posible, sí que hay que proteger a los sistemas eléctricos de estos. Estas acciones pueden ser realizadas en dos capas distintas del propio sistema. En primer lugar mejorando los protocolos de red y comunicaciones, haciéndolos más seguros e invulnerables. Por otro lado también es posible emplear sitemas inteligentes basados en técnicas de inteligencia artificial capaces de detectar un ataque una vez que haya logrado superar el mencionado primer nivel de protección. Los ataques de inyección de datos falsos (FDI) son inserciones maliciosas de datos falsos como mediciones de sensores en un sistema ciberfísico, con el fin de llevar al sistema a tomar una acción incorrecta. Los ataques de inyección de datos falsos no atacan los componentes informáticos o de red de los sistemas ciberfísicos, sino la interfaz entre la parte física y la cibernética (Wolf and Serpanos, 2020).

Recoge (itUser Tech & Business, 2022) que "la demanda de habilidades de seguridad y experiencia específica en tecnología operativa (OT) ha ido en aumento en las empresas industriales durante los últimos años, debido a la escalada de amenazas y la mayor prevalencia de marcos y regulaciones de seguridad de TI/OT. Sin embargo, la falta de profesionales de seguridad en este ámbito amenaza su ciberprotección. Así lo pone de manifiesto el informe "Kaspersky ICS Security Survey 2022: Las siete claves para mejorar los resultados de la seguridad OT", que muestra que la escasez de equipos de seguridad OT amenaza la protección en el 16 % de las organizaciones industriales europeas".

Comienza a existir en la literatura una gran variedad de métodos propuestos para detectar los ciberataques, de entre los cuales las técnicas de Machine Learning han encontrado gran éxito. (Shahid et al., 2022) Propone una estrategia de detección de ataques de inyección de datos falsos en una red de distribución de corriente alterna empleando técnicas basadas en variables ficticias (dummy values). En (Almasabi et al., 2021) se expone una técnica para detectar ataques de datos falsos en dispositivos de medida de fasores. Los autores de (Qu et al., 2021) sugieren un método de detección basado en genes ciberfísicos, mejorando los problemas de detección en datos complejos con bajo nivel de precisión. Por otro lado, los sistemas de distribución de corriente continua son considerados cada vez más como el próximo salto tecnológico debido a su capacidad de integrar distintos tipos de fuentes de energías renovables y distribuirlos a una gran variedad de cargas y usos. Con esta motivación, (Tan et al., 2022) emplea una aproximación de optimización multiobjetivo para detectar intrusiones en este tipo de redes.

Teniendo en cuenta la mayor penetración e importancia que tienen los sistemas de generación de energía basado en fuentes renovables y la creciente vulnerabilidad de los mismos a los ataques mal intencionados, en este trabajo se propone una estrategia basada en técnicas de redes neuroborrosas capaces de aportar una solución viable e implementable en el mundo industrial a estos ataques. Se realiza el estudio en un aerogenerador empleando medidas de campo obtenidas de una instalación existente. La organización del resto del artículo es la siguente. En la sección 2 se realiza un tratamiento prévio de datos obtenidos de un sistema de supervisión y adquisición de datos (SCADA) y se preparan los conjuntos con fallos inducidos artificialmente. En la misma sección se hace un análisis de correlación de las variables, la organización de los datos y los análisis de las componentes principales de las variables. En la sección 3 se presenta el diseño de los detectores neuro-borrosos propuestos. En la sección 4 se procede a la evaluación de los detectores, terminando con una sección de conclusiones.

2. Preparacion de los datos de operacion

Para el desarrollo del detector neuro-borroso partimos de los datos históricos recogidos por el SCADA de un aerogenerador de 3MW de una instalación onshore con rotor a barlovento de eje horizontal con tres palas como muetra la Figura 1.

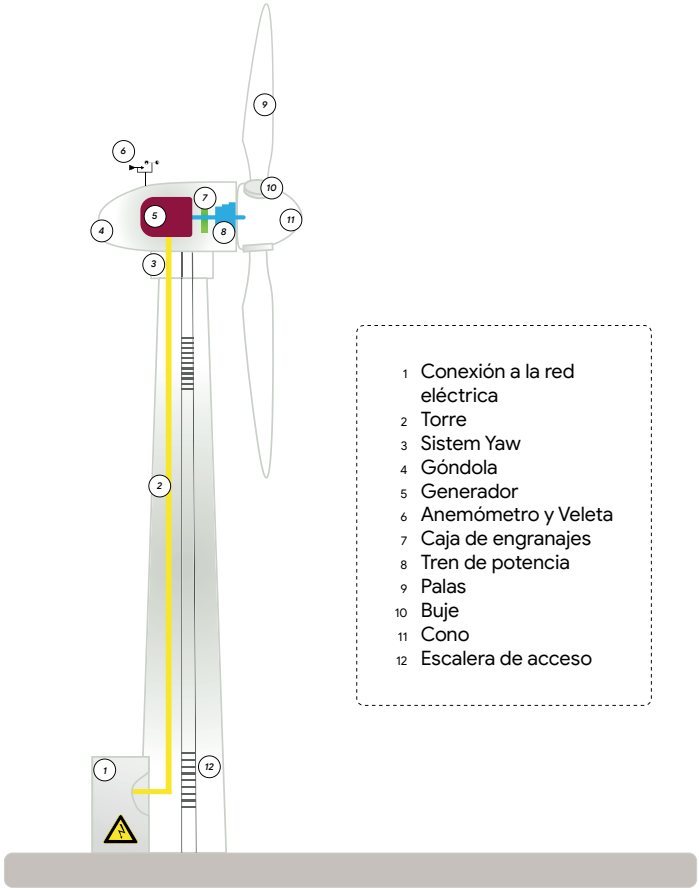


Figura 1: Aerogenerador.

Los datos historicos contienen las medidas realizadas por los sensores de las variables que se muestran en la Tabla 1. Además, el SCADA almacena los datos en una hoja Excel día a día, se hace mensión a que el registro de datos (como es el caso del viento) es de 10 minutos como una práctica estándar de la industria eólica *IEC 61400*.

Tabla 1: Datos historicos de operación del aerogenerador

Descripción	**Unidad**	**Variable**
Fecha de regidto de las variables	-	$Date$
Lectura del contador	-	Mr
Electricidad generada	kWh	Eg
Velocidad del viento	m/s	Ws
Velocidad de giro del rotor	rpm	Rs
Potencia reactiva	kW	Q
Potencia activa	kW	P
Posición de la góndola	°	Np

El sistema SCADA del aerogenerador entrega datos históricos en bruto sin ningún tratamiento, por lo cual es necesario filtrar el ruido de la instrumentación, suprimir valores atípicos y elegir cuidadosamente las variables que afectan al proceso a partir de los datos reales de funcionamiento, en este caso a la potencia activa y reactiva generada por la turbina eólica. El archivo histórico contiene una serie de variables y contiene datos de todo un año de calendario, que se exportó a Matlab como tablas de tiempo con los nombres de cada variable. Inicialmente se realizó un proceso de eliminar datos inconsistentes, datos negativos y una interpolación de datos en las muestras faltantes en cada una de las variables.

Además, la energía que entrega el aerogenerador a la carga conectada en él, en algunos días de varios meses, su potencia activa se ve saturada por una excesiva generación, debido a que no cuenta con una unidad de almacenamiento que pueda capturar este exceso de energía y porque no cuenta con una conexión a la red eléctrica donde se podría verter dicha energía. Por lo cual dichos datos se han eliminado mediante un proceso de agrupación con la finalidad de capturar un comportamiento realista de la potencia y por ende de la energía eléctrica generada como se muestra en la Figura 2.

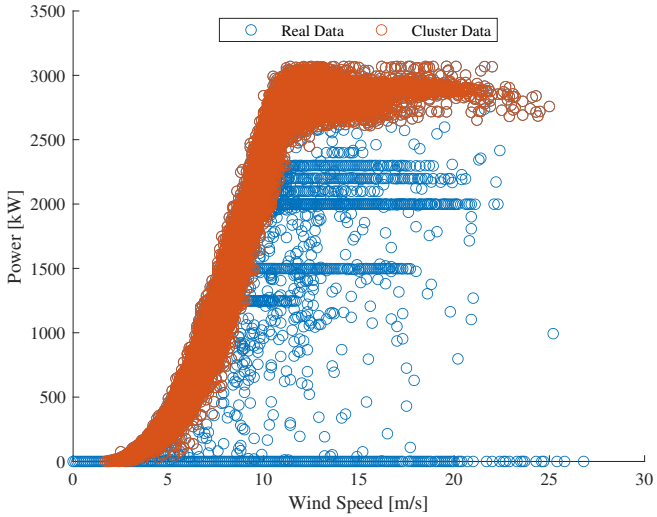


Figura 2: Aerogenerador.

Como resultado se tiene una tabla de tiempo con 16 variables (columnas) que componen un conjunto total de 41426 muestras con un tiempo de muestreo de 10 minutos y se muestra parte de ellos en la Figura 3. Cabe mencionar que se cuenta con los datos mínimos y máximos datos de las variables Ws, Rs, Q y P, que en este trabajo se han omitido. Como la velocidad el viento produce el empuje de las palas, hacen rotar el cono que tiene conexión con el eje a la caja de engranajes, variando así la velocidad de rotación del generador conectado en él, que trasforma las energías de rotación en energía eléctrica. Pero el elemento final que hace de interfaz entre el generador y la carga que ha de consumir dicha energía es un inversor (Alcalá et al., 2011) que adecua la potencia generada a los estándares eléctricos, mediante los sistemas de control potencia montadas sobre DSP, que calcula la estabilidad de la frecuencia de la red , la potencia activa y reactiva de la turbina, es decir asegura que la calidad de la potencia sea la adecuada.

El objetivo de este trabajo es la detección de la inyección de datos falsos desde un enfoque de ciberataque en la tecnología operativa (OT) que conforman la turbina de viento, un ejemplo de ello son los dispositivos del control que conforman el sistema de control, que reciben las mediciones de los sensores-transmisores que aseguran que la estabilidad y calidad de la potencia activa y reactiva. Sin embargo, estas mediciones podrían ser falseadas o modificadas dentro del los sistemas de control y el SCADA lo que provocaría un desequilibro en la red eléctrica o en su caso en la carga conectada en él. Por lo cual este trabajo se enfoca a la detención de datos falsos en la potencia activa y reactiva por ser las variables de mayor interés en los sistemas de control.

Para simular la inyección de datos falsos, partimos de todos los datos de operación y formamos un conjunto de datos con un offset positivo y negativo tanto de la potencia activa y reactiva, el offset añadido se calcula con un umbral del 25 % de la media de la potencia activa y de la misma forma para la potencia reactiva, por lo que el detector neuro-borroso ha de clasificar los datos en tres clases distintas: operación normal, falso positivo y falso negativo en la variables antes mencionadas, como se muestra en la Tabla 2.

Tabla 2: Fallos añadidos en las variables P, Q

Variable	**Unidad**	**inyección de dato falso**
Potencia activa (P)	kW	±272,8
Potencia reactiva (Q)	kW	±28,2

2.1. *Análisis de correlación de las variables*

El análisis de correlación se realiza con el fin de ver el grado relación que existe entre cada variable y elegir las variables que influyen en su variación. Para realizar el análisis de correlación se reorganiza el conjunto de datos inicial, omitiendo las variables máximas y mínimas de la velocidad del viento, velocidad del rotor, potencia activa y reactiva.

Partimos de una matriz de datos $\mathbf{X} \in \Re^{n \times m}$, donde n representa el número total de muestras que son 41426 y m representa el número de variables que en este caso son 7, cada una de las variables de la matriz $\mathbf{X}$ inicialmente se normalizan con media cero y varianza unitaria; cuyo análisis de correlación da como resultado una matriz de coeficientes de correlación $\mathbf{\Gamma} \in \Re^{m \times m}$ como muestra la Figura 4.

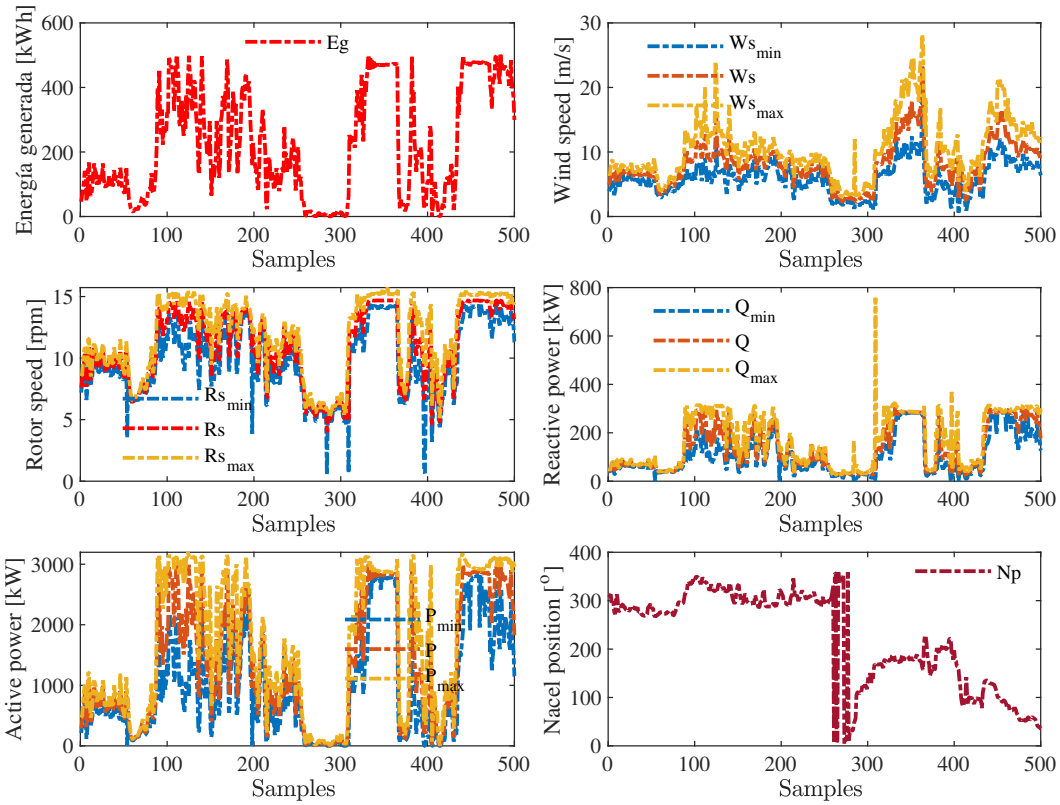


Figura 3: Datos históricos del Aerogenerador.

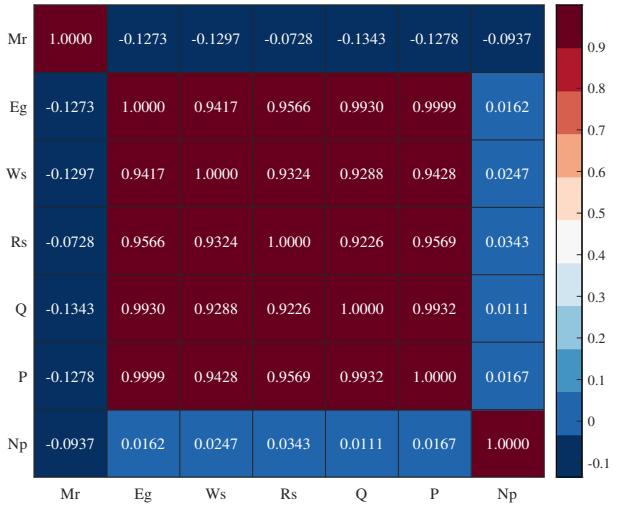


Figura 4: Matriz de coeficientes de correlación de los datos del aerogenerador.

Como se puede observar en la matriz de coeficientes de correlación las variables pueden tomar valores en un rango de $[-1, 1]$, donde -1 representa una correlación negativa, 1 representa una correlación positiva completa y 0 indica que las variables de la matriz $\mathbf{X}$ no están correlacionadas. En este caso se aprecia que las variables lectura del contador y posición de la góndola presentan correlaciones casi nulas en relación con las variables: electricidad generada, velocidad del viento, velocidad el rotor, potencia activa y reactiva, motivo por el cual se excluyen para el diseño del detector neuro-borroso al igual que la electricidad generada ya que está en función de la potencia generada y sería un dato redundante para nuestro fin.

2.2. *Organización de los datos*

Partiendo del análisis de correlación se ha formado los grupos con las variables que afectan a la potencia activa y reactiva: $[Ws,\ Rs]$, como se puede apreciar en la matriz de correlación (ver Figura 4) la velocidad de viento y la velocidad de giro del rotor tienen una alta correlación con la variables en las cuales se detectará la inyección de datos falsos.

Además, se realiza una estructuración previa de los datos antes de diseñar los clasificadores neuro-borros, es así que se forman tres conjuntos: el primer conjunto conformado por datos de operación normal de la potencia activa, mientras que el segundo y tercer conjunto está formado por los umbrales de inyección positiva y negativa (ver tabla 2), resultado de sumar y restar a cada muestra de la potencia activa su respectivo umbral de fallo.

$$\mathbf{D}_P^N = [Ws,\ Rs,\ P]$$
$$\mathbf{D}_P^{Fp} = [Ws,\ Rs,\ P^{Fp}]$$
$$\mathbf{D}_P^{Fn} = [Ws,\ Rs,\ P^{Fn}]$$

Donde, $Nr,\ Fp$ y Fn representan operación normal, faso positivo y falso negativo respectivamente, una vez formado los conjuntos se han de normalizar con media cero y varianza unitaria para su posterior uso. Obtenemos de esta forma diferentes matrices de datos $\mathbf{D}_v^c \in \Re^{n\times k}$, donde c indica la clase y v la variable de interés.

Las matrices de datos se dividen en subconjuntos: entrenamiento, chequeo y validación como muestra la Figura 5.

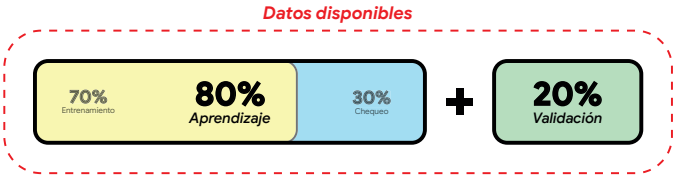


Figura 5: Subconjuntos a utilizar en el aprendizaje y validación de cada ANFIS
.

Tal y como se presenta en (Chicaiza et al., 2021), con los conjuntos de entrenamiento se realiza un análisis de componentes principales (Principal Component Analysis, PCA) para reducir la dimensión del espacio de variables, proyectando los datos originales en las componentes de mayor variabilidad. Adicionalmente se obtiene las componentes principales de todas las clases de cada variable de interés. Dichas proyecciones se emplearán en el aprendizaje y validación de los ANFIS de cada clase que conformaran el clasificador neuro-borroso. El mismo procedimiento se aplicará con los datos de la potencia reactiva para el aprendizaje de su clasificador.

2.3. *Análisis de componentes principales*

Partiendo del conjunto de datos normalizados $\mathbf{D}_v^c \in \Re^{n\times m}$ del cual se obtiene de su matriz de covarianzas $\mathbf{R}_v^c \in \Re^{n\times m}$ (1) con rango r ($r \leq min\{n, m\}$), donde N es el numero total de muestras de la matriz $\mathbf{D}_v^c$:

$$(N-1)\mathbf{R}_v^c = \mathbf{D}_v^{cT}\mathbf{D}_v^c \tag{1}$$

Y conjuntamente con la descomposición de valores singulares de la matriz de datos $\mathbf{D}_v^c$ (2) que esta relacionada con la descomposición propia de la matriz de covarianzas $\mathbf{R}_v^c$ como:

$$\mathbf{D}_v^c = \mathbf{U}_v^c\mathbf{S}_v^c\mathbf{V}_v^{cT} \tag{2}$$

donde, $\mathbf{U}_v^c \in \Re^{n\times r}$, $\mathbf{V}_v^c \in \Re^{m\times r}$ son matrices con columnas ortogonales y $\mathbf{S}_v^c \in \Re^{r\times r}$ es una matriz diagonal cuyos elementos son la raíz cuadrada de los valores propios de $\mathbf{D}_v^c\mathbf{D}_v^{cT}$ en forma descendente. Al reemplazar (2) en (1) se obtiene el PCA(3) de la siguiente manera:

$$(N-1)\mathbf{R}_v^c = \mathbf{V}_v^c\mathbf{\Lambda}_v^c\mathbf{V}_v^{cT} \tag{3}$$

donde, $\mathbf{\Lambda}_v^c \in \Re^{r\times r}$ es una matriz diagonal con los valores propios de $(N-1)\mathbf{R}_v^c$ en forma descendente. Las columnas de $\mathbf{V}_v^c$ son los vectores propios $\mathbf{v_r}$ de los λ_r valores propios y marcan la dirección del nuevo espacio de las componentes principales (CP).

2.4. *Proyección de los datos en las componentes principales*

Tras realizar el algoritmo PCA, la matriz $\mathbf{V}_v^c$ formada por los vectores propios contienen los coeficientes de las componentes principales de cada variable. Los vectores propios se almacenan en una nueva matriz $\mathbf{C}_v^c \in \Re^{r\times r}$ definida como *matriz loading* que se empleara para obtener la proyección de los datos como (4) ;

$$\mathbf{T}_v^c = \mathbf{D}_v^c \times \mathbf{C}_v^c \tag{4}$$

donde, $\mathbf{T}_v^c \in \Re^{n\times m}$ contiene varios vectores columna $\mathbf{t_m}$ resultado de la proyección en su respectiva componente principal $\mathbf{p_m}$ y se define como la *matriz score*, los nuevas variables proyectadas no tienen ninguna correlación entre ellas. Generalmente la matriz de datos se proyecta en la componente que contenga la mayor variabilidad [1].

La calidad de una componente principal (CP) puede ser medida de una manera estándar (Jollife and Cadima, 2016) como la proporción de la varianza total explicada expresada como un porcentaje.

$$\pi_j = \frac{\lambda_j}{\sum_{j=1}^{p}\lambda_j} \times 100\,\% \tag{5}$$

Tras realizar las mediciones de calidad de cada CP se ha determinado que las dos primeras representan un variabilidad del 98,63 %, por lo cual la proyección de cada conjunto total de datos $\mathbf{D}_v^c$ se realiza en las dos primeras componentes de cada uno de ellos.

2.4.1. *Obtención de prototipos proyectados en las CP*

Los dos primeros vectores propios de cada $\mathbf{V}_v^c \rightarrow \{\mathbf{V}_v^{Nr}, \mathbf{V}_v^{Fp}, \mathbf{V}_v^{Fn}\}$ se almacenan en las matrices *load* $\mathbf{C}_v^c \rightarrow \{\mathbf{C}_v^{Nr}, \mathbf{C}_v^{Fp}, \mathbf{C}_v^{Fn}\}$, ya que contienen los coeficientes de cada CP, donde $c \rightarrow \{Nr,\ Fp,\ Fn\}$ representa la clase: operación normal, falso positivo, falso negativo y v representa la variable de estudio. En el caso de el PCA de todas las clases del vector $\mathbf{V}_v^{Tot}$ se toma la primer componente que contiene una variabilidad del 95 %y se almacena en la matriz $\mathbf{C}_v^{Tot}$.

Se determina las proyecciones de cada clase por cada una de las componentes, es decir, se va ha proyectar cada clase $\mathbf{D}_v^c \rightarrow \{\mathbf{D}_v^{Nr}, \mathbf{D}_v^{Fp}, \mathbf{D}_v^{Fn}\}$ en las dos primeras componentes principales de las distintas clases. Las matrices *score* $\mathbf{T}_v^c \rightarrow \{\mathbf{T}_v^{Nr}, \mathbf{T}_v^{Fp}, \mathbf{T}_v^{Fn}\}$ contienen las distintas proyecciones obtenidas de las variables que componen cada clase que llamaremos prototipos, consiguiendo seis nuevas variables por clase en el subespacio de componentes principales, que se empleará posteriormente en el aprendizaje del clasificador.

$$\begin{aligned} {}^{Nr}\mathbf{T}_v^c &= \mathbf{D}_v^{Nr} \times \mathbf{C}_v^c \\ {}^{Fp}\mathbf{T}_v^c &= \mathbf{D}_v^{Fp} \times \mathbf{C}_v^c \\ {}^{Fn}\mathbf{T}_v^c &= \mathbf{D}_v^{Fn} \times \mathbf{C}_v^c \end{aligned} \tag{6}$$

Además, se obtiene la matriz *score* general de cada clase ${}^{g}\mathbf{T}_v^c$, resultado de realizar el producto de cada clase con la matriz $\mathbf{C}_v^{Tot}$.

$${}^{g}\mathbf{T}_v^c = \mathbf{D}_v^c \times \mathbf{C}_v^{Tot} \tag{7}$$

Con las proyecciones obtenidas se conforman los conjuntos de entrenamiento y chequeo para cada ANFIS que conformaran el

[1] El primer componente principal, que es una combinación lineal de las variables originales, define la dirección de la mayor variabilidad en el conjunto de datos, por lo cual tiene la mayor suma de varianza en la matriz $\mathbf{\Lambda}$ (Wold et al., 1987).

clasificador neuro-borroso:

$$\mathbf{Trn}^{Nr} = \left[{}^{Nr}\mathbf{T}_v^{Nr} \;\; {}^{Nr}\mathbf{T}_v^{Fp} \;\; {}^{Nr}\mathbf{T}_v^{Fn} \;\; {}^{g}\mathbf{T}_v^{Nr} \right], \tag{8a}$$

$$\mathbf{Trn}^{Fp} = \left[{}^{Fp}\mathbf{T}_v^{Nr} \;\; {}^{Fp}\mathbf{T}_v^{Fp} \;\; {}^{Fp}\mathbf{T}_v^{Fn} \;\; {}^{g}\mathbf{T}_v^{Fp} \right], \tag{8b}$$

$$\mathbf{Trn}^{Fn} = \left[{}^{Fn}\mathbf{T}_v^{Nr} \;\; {}^{Fn}\mathbf{T}_v^{Fp} \;\; {}^{Fn}\mathbf{T}_v^{Fn} \;\; {}^{g}\mathbf{T}_v^{Fn} \right], \tag{8c}$$

Como se puede apreciar la proyección de cada clase sobre las demás clases y la proyección general de la clase conforman el conjunto de entrenamiento. El procedimiento para obtener los conjuntos de chequeo es el mismo y se utilizan las matrices *load* $\mathbf{C}_v^c$, $\mathbf{C}_v^{Tot}$ que se utilizaron en el conjunto de entrenamiento para el cálculo de las proyecciones.

3. Detector neuro-borroso con proyección de PCA

El detector neuro-borroso desarrollado combina un análisis de componentes principales (PCA) con el fin de proyectarlas en el subespacio de la o las variables que presenten mayor variabilidad con una arquitectura de red ANFIS(sistema de inferencia neuro-borroso adaptativo).

El PCA representa observaciones de un espacio n-dimensional de forma óptima en un espacio de dimensión reducida, además trasforma las variables correladas en variables ortogonales no-correrladas en el subespacio de las variables que presente mayor variabilidad. Mientras que ANFIS combina las ventajas de la lógica borrosa para representar el conocimiento en una forma interpretable basado en reglas con etiquetas lingüísticas del lenguaje humano con las ventajas de aprendizaje de las redes neuronales adaptativas (RNA) para optimizar los parámetros de los antecedentes y consecuentes de las reglas borrosas. Con ambas técnicas se desarrolla el detector neuro-borroso, que se basa en *clasificadores neuro-borroso* para detectar los fallos en la potencia activa y reactiva.

El detector se basa en dos clasificadores neuro-borrosos con proyecciones PCA para detectar la inyección de datos falsos en la potencia activa y reactiva. Como se puede observar en la Figura 6, el clasificador se basa en un sistema de inferencia neuro-fuzzy adaptativo (ANFIS) para obtener un modelo digital capaz de detectar los datos con fallos utilizando como entradas las medidas entregada por los sensores del aerogenerador.

Figura 6: Detector neuro-borroso.

A partir de los valores medidos, el detector indicará los fallos en las variables, cuando los haya, asignando a las variables binarias F_P, F_Q el valor 1 cuando haya un defecto o 0 cuando no lo haya.

3.1. Clasificador neuro-borroso

El clasificador neuro-borroso se conforma de varios sistemas de inferencia borrosa (FIS) obtenidos después de realizar un proceso de aprendizaje. En dicho proceso se emplea una arquitectura de red ANFIS (Jang, 1993) que utiliza los conjuntos de entrenamiento y chequeo de cada clase. Cada ANFIS utiliza como entradas los prototipos de cada clase (proyección de cada clase sobre las demás clases) y el prototipo general de la clase como la salida que debe aprender. Por lo tanto, se utilizan tres ANFIS uno por clase $c \rightarrow \{Nr,\ Fp,\ Fn\}$ que son entrenados para su posterior validación, de tal forma que se obtiene tres FIS, que entregan la estimación del prototipo general ${}^{g}\mathbf{T}_v^c$ correspondientes a cada clase.

El entrenamiento de cada ANFIS se realiza aplicando primero un método de agrupación sustractivo (Chiu, 1994) que estima inicialmente el número de clusters y sus centros para determinar el número de reglas y las funciones de membresía (FM). A continuación, los parámetros de cada capa de ANFIS se actualizan mediante un método de aprendizaje híbrido que combina el descenso de gradiente para optimizar los parámetros antecedentes y los mínimos cuadrados para determinar los parámetros lineales consecuentes en cada época. Los parámetros de la arquitectura ANFIS es la misma en todos los casos debido a la correlación que tienen los datos (similitud entre cada clase, debido a que se añade un offset en positivo y negativo). La Tabla 3 presenta el resumen de cada parámetro ANFIS para cada clase de la potencia activa y reactiva.

Tabla 3: Parámetros ANFIS para cada clase de las variables $P,\ Q$

Description	**ANFIS**		
Tipo FM :	*Gaussiana*		
Método de optimización:	*hibrido*		
Tipo de salida:	*lineal*		
Clase	1	2	3
FIS_P	Nr	Fp	Fn
Número de FM:	2	2	2
Número de reglas:	2	2	2
Rango de influencia:	0,8	0,8	0,8
Epocas:	175	175	175
Clase	1	2	3
FIS_Q	Nr	Fp	Fn
Número de FM:	2	2	2
Número de reglas:	2	2	2
Rango de influencia:	0,8	0,8	0,8
Epocas:	175	175	175

El proceso de entrenamiento de cada ANFIS emplea número de épocas relativamente bajas, mostrando un aprendizaje generalizado de los conjuntos de entrenamiento y chequeo. Esto se refleja en los índices de error que se obtienen tras finalizar el proceso de aprendizaje de cada $\mathbf{FIS}_v^c$ que se muestran en la tabla 4.

El objetivo es detectar la inyección de datos falsos utilizando los sistemas $\mathbf{FIS}_v^c$ que entregan a su salida la estimación de los prototipos generales ${}^{g}\mathbf{T}_v^c$, por lo cual definimos una función de coste que determina que sistema alcanza al prototipo general real ${}^{g}\mathbf{T}_v^{*real}$ de un nuevo dato.

$$J_i = \left\| {}^{g}\mathbf{T}_v^{*real} - {}^{g}\mathbf{T}_v^{c_i} \right\|_2^2 \tag{9}$$

Se emplea entonces, un *algoritmo de búsqueda exhaustiva*

Tabla 4: Índices RMSE obtenido del proceso de aprendizaje de cada ANFIS

RMSE minimo Entrenamiento & Chequeo	FIS obtenidos		
Clase	1	2	3
FIS_P	Nr	Fp	Fn
$RMSE_{Train}$	0.000521981	0.000521977	0.000521981
$RMSE_{Check}$	0.000515779	0.000515776	0.000515780
Clase	1	2	3
FIS_Q	Nr	Fp	Fn
$RMSE_{Train}$	0.000579924	0.000579675	0.000579939
$RMSE_{Check}$	0.000576049	0.000575746	0.000576069

(ESA) que asignará para la clase a la que pertenece el ${}^{g}\mathbf{T}_v^{c,*real}$ del nuevo dato. Este algoritmo evalúa la función de coste con el valor que entrega cada sistema de inferencia borroso $\mathbf{FIS}_v^c$ a su salida, es decir los prototipos generales estimados ${}^{g}\mathbf{T}_v^c$ y determina la clase por comparación directa, eligiendo aquel que minimiza la función coste. El $\mathbf{FIS}_v^c$ que presente el mínimo J_i define la clase a la que pertenece el nuevo dato.

3.2. Estructura del clasificador neuro-borroso

Con cada uno de los sistemas de inferencia borrosos $\mathbf{FIS}_v^c$ obtenidos, que estiman los prototipos generales ${}^{g}\mathbf{T}_v^c$ de cada tipo de fallo inducido de la variable de interes v, se crea la estructura final que tendrá el clasificador neuro-borroso.

Cada $\mathbf{FIS}_v^c$ contiene un conjunto de relas j del tipo TS (Takagi and Sugeno, 1983):

$$\texttt{IF } x_1 \texttt{ is } F_{1j} \texttt{ and } x_2 \texttt{ is } F_{2j} \texttt{ and } x_i \texttt{ is } F_{ij} \,,$$

$$\texttt{THEN} : f_j(x) = g_{0j} + g_{1j}x_1 + \cdots + g_{ij}x_i$$

cada conjunto borroso F_{ij} de cada entrada nítida $x_i : \mathbf{T}_v^c$ contiene varias funciones de pertenencia tipo gaussiana:

$$\mu_{F_{ij}}(x_i) = \frac{1}{1 + \left[\left(\dfrac{x_i - c_{ij}}{a_{ij}}\right)^2\right]^{b_{ij}}} \tag{10}$$

donde, $\{a_{ij}, b_{ij}, c_{ij}\}$ son los *parámetros antecedentes* que la definen y los términos $g_{ij} \in \Re$ de cada función polinómica de primer orden son los *parámetros consecuentes*. La salida final de cada $\mathbf{FIS}_v^c$ es la media ponderada de la salida que entrega cada regla.

La estructura formada del clasificador se muestra en la Figura 7.

$\mathbf{C}_v^c,\ \mathbf{C}_v^{Tot}$ de la potencia activa y reactiva

Las matrices *score* para cada clase $\{\mathbf{C}_P^c,\ \mathbf{C}_Q^c\}$ obtenidas anteriormente se mantienen como parámetros fijos y sirven para definir los nuevos prototipos $\{\mathbf{T}_P^c,\ \mathbf{T}_Q^c\}$ de entrada a cada $\{\mathbf{FIS}_P^c,\ \mathbf{FIS}_Q^c\}$ que estiman el prototipo general $\{{}^{g}\mathbf{T}_P^c,\ {}^{g}\mathbf{T}_Q^c\}$ de cada clase.

Si evaluamos un conjunto de datos de operación normal (no contiene inyección de datos falsos) y se multiplica por la matriz $\{\mathbf{C}_P^{Nr},\ \mathbf{C}_Q^{Nr}\}$ se obtendrá el prototipo proyectado en la clase normal $\{\mathbf{T}_P^{Nr},\ \mathbf{T}_Q^{Nr}\}$, si los mismos datos se multiplican por la matriz $\{\mathbf{C}_P^{Fn},\ \mathbf{C}_Q^{Fn}\}$ obtendremos la proyección en la clase falso positivo $\{\mathbf{T}_P^{Fp},\ \mathbf{T}_Q^{Fp}\}$ y cuando se multipliquen por $\{\mathbf{C}_P^{Fn},\ \mathbf{C}_Q^{Fn}\}$, se proyectan en la clase falso negativo $\{\mathbf{T}_P^{Fn},\ \mathbf{T}_Q^{Fn}\}$. El dato que ingresa pasa por la matriz $\{\mathbf{C}_P^c,\ \mathbf{C}_Q^c\}$ es proyectado en la clase $c \rightarrow \{Nr,\ Fp,\ Fn\}$, obteniendo de esta manera su prototipo correspondiente $\mathbf{T}_P^c$, donde $P,\ Q$ denotan la potencia activa y reactiva. Los prototipos calculados $\{\mathbf{T}_P^c,\ \mathbf{T}_Q^c\}$ forman el vector de entrada con el que se evalúa cada uno de los $\{\mathbf{FIS}_P^c,\ \mathbf{FIS}_Q^c\}$. Finalmente, al pasar los datos por la matriz $\{\mathbf{C}_P^{Tot},\ \mathbf{C}_Q^{Tot}\}$, se obtiene el prototipo general real $\{{}^{g}\mathbf{T}_P^{*real},\ {}^{g}\mathbf{T}_Q^{*real}\}$ con el cual se ha de evaluar la función coste en el **ESA** para determinar la clase a la que pertenece el nuevo dato entrante. **ESA** realiza una comparación directa evaluando la función de coste con el valor estimado de cada $\{\mathbf{FIS}_P^c,\ \mathbf{FIS}_Q^c\}$ y asigna la etiqueta del FIS que minimiza dicha función. Por ejemplo, cuando los datos evaluados contiene inyección de datos negativos, el sistema $\{\mathbf{FIS}_P^{Fn},\ \mathbf{FIS}_Q^{Fn}\}$ minimiza la función de coste y esa clase es la que le asigna.

4. Evaluación del clasificador neuro-borroso

Evaluamos cada $\mathbf{FIS}_P^c$ con datos de cada clase. Si el dato que ingresa no contiene inyección de datos falsos, la salida del $\mathbf{FIS}_P^c$ que más se acerca al prototipo real es la del $\mathbf{FIS}_P^{Nr}$ como se observa en la Figura 8(a). La Figura 8(b) muestra que, si el dato que ingresa tiene inyección de datos falsos positivos , los valores estimados por el $\mathbf{FIS}_P^{Fp}$ seguirán al prototipo general real. Si clasificamos un dato con inyección de datos falsos negativos, la salida del $\mathbf{FIS}_P^{Fn}$ sigue al prototipo general como se puede observar la Figura 8(c).

El algoritmo propuesto para asignar la clase a la que pertenece un nuevo dato evaluado presenta buenos resultados como se puede observar la Figura 4. **ESA** realiza una comparación directa evaluando la función de coste con el valor estimado de cada $\mathbf{FIS}_P^c$ y asigna la etiqueta del FIS que minimiza dicha función. Por ejemplo, cuando los datos evaluados tienen falsos positivos, el $\mathbf{FIS}_P^{Fp}$ minimizará la función de coste, y esa clase se asigna a los datos, como se muestra en la Figura 9(b). Si el dato evaluado son de operación normal, el $\mathbf{FIS}_P^{Nr}$ minimizará la función de coste y se asigna esa clase al dato, ver la Figura 9(a). Finalmente, la Figura 9(c) muestra que si el dato presenta falsos negativos el $\mathbf{FIS}_P^{Fn}$ minimizará la función de coste y el algoritmo asignara esa clase al nuevo dato entrante.

Adicionalmente se ha realizado la matriz de confusión (ver Tabla 5) para evaluar el clasificador propuesto, esta matriz

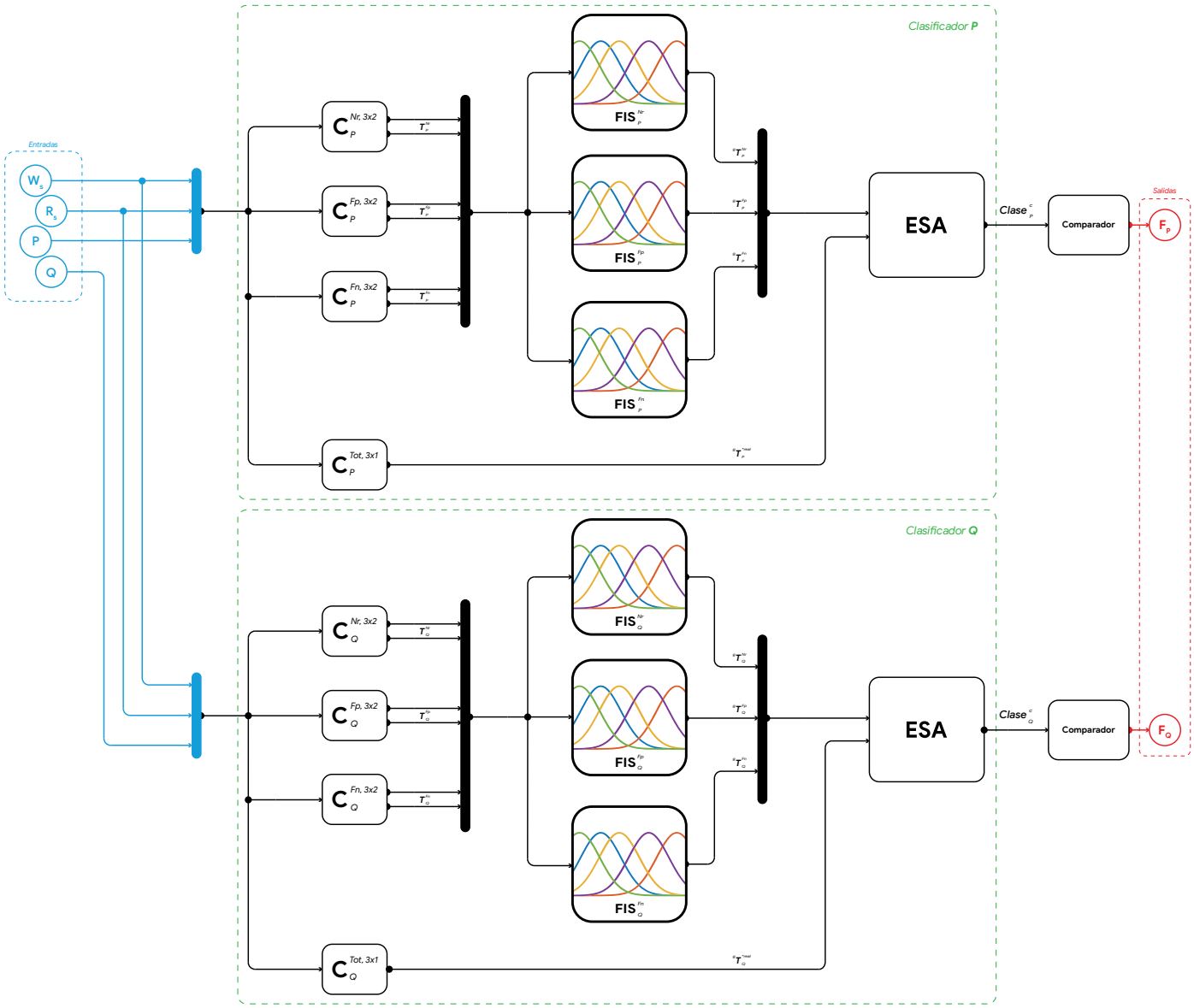


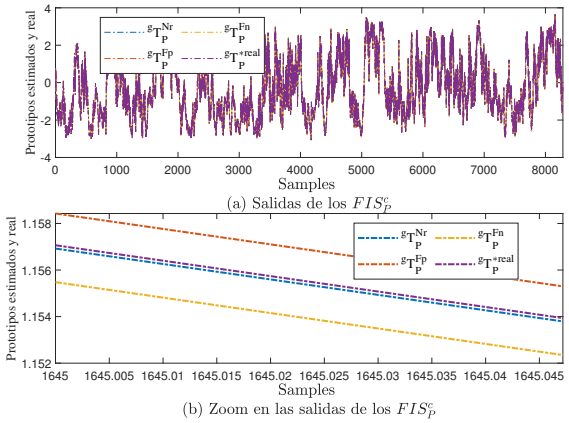
Figura 7: Detector neuro-borroso

muestra la distribución de las clases asignadas a los datos predichos, en comparación con sus clases verdaderas.

Tabla 5: Matriz de confusión **Potencia activa**.

Actual \ Clases estimadas	$\mathbf{F_n}$	Nr	$\mathbf{F_p}$
$\mathbf{F_{neg}}$	8204	81	0
Norm	279	7925	81
$\mathbf{F_{pos}}$	13	266	8006

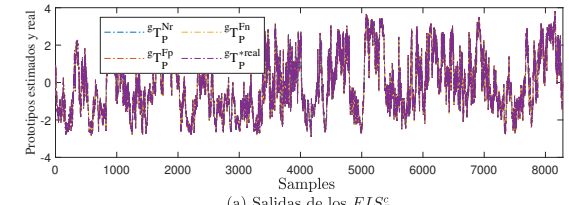
Presicion	99.02 %	95.65 %	96.63 %
Recall	96.56 %	95.81 %	99.00 %
Accuracy	97.10 %		



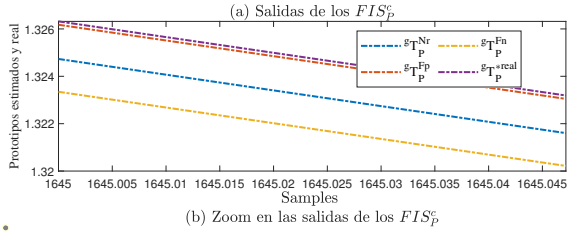
(a) Salidas de los FIS_P^c

(b) Zoom en las salidas de los FIS_P^c

(a) Prototipo real y estimado por cada FIS tras ser evaluado con datos de operación normal



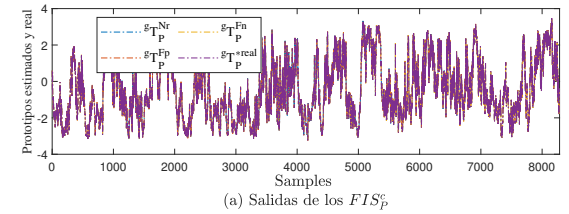
(a) Salidas de los FIS_P^c



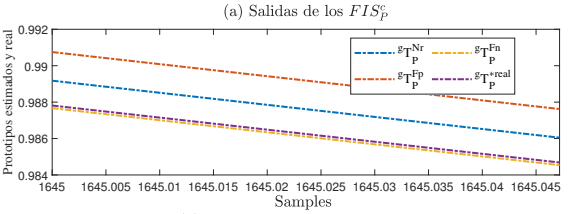
(b) Zoom en las salidas de los FIS_P^c

(b) Prototipo real y estimado por cada FIS tras ser evaluado con datos que contienen falsos positivos

De manera similar ocurrirá con el clasificador que detecta si hay inyección de datos en la potencia reactiva, por ellos los resultados muestran los prototipos generales estimados ${}^g\mathbf{T}_Q^c$ por cada $\mathbf{FIS}_Q^c$ y el prototipo general real ${}^g\mathbf{T}_Q^{*real}$ con datos de operación normal, falsos positivos y falsos negativos en las Figura 4. Mientras que la Figura 4 muestra la clase $c \rightarrow \{Nr,\ Fp,\ Fn\}$ obtenida tras aplicar el algoritmo ESA al dato nuevo evaluado. Además, se puede observar su respectiva matriz de confusión en la Tabla 6.



(a) Salidas de los FIS_P^c



(b) Zoom en las salidas de los FIS_P^c

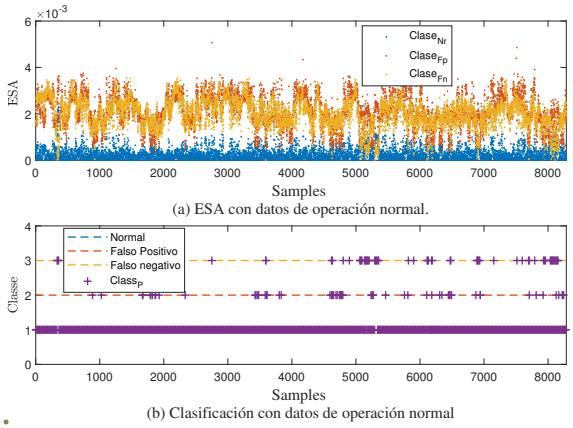
(c) Prototipo real y estimado por cada FIS tras ser evaluado con datos que contienen falsos negativos

Figura 8: Prototipos obtenidos con inyección de datos falsos y operación normal en la potencia activa.

Tabla 6: Matriz de confusión **Potencia reactiva**.

Actual \ Clases estimadas	$\mathbf{F_n}$	Nr	$\mathbf{F_p}$
$\mathbf{F_{neg}}$	7920	345	20
Norm	1174	6779	332
$\mathbf{F_{pos}}$	137	1150	6998

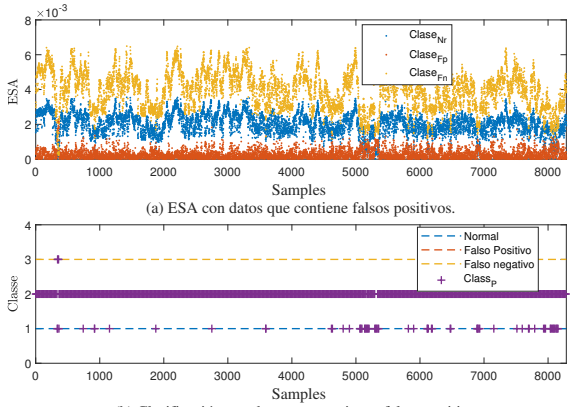
Presicion	95.59 %	81.82 %	84.47 %
Recall	85.80 %	81.93 %	95.21 %
Accuracy	87.29 %		



(a) ESA con datos de operación normal.

(b) Clasificación con datos de operación normal

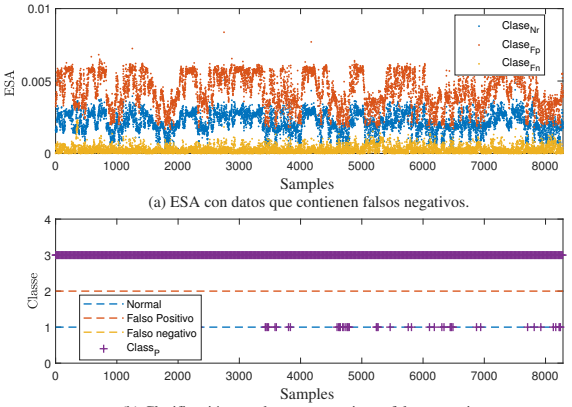
(a) Clase obtenida por comparación directa del ESA con datos de operación normal.



(a) ESA con datos que contiene falsos positivos.

(b) Clasificación con datos que contienen falsos positivos.

(b) Clase obtenida por comparación directa del ESA con datos que contienen falsos positivos.

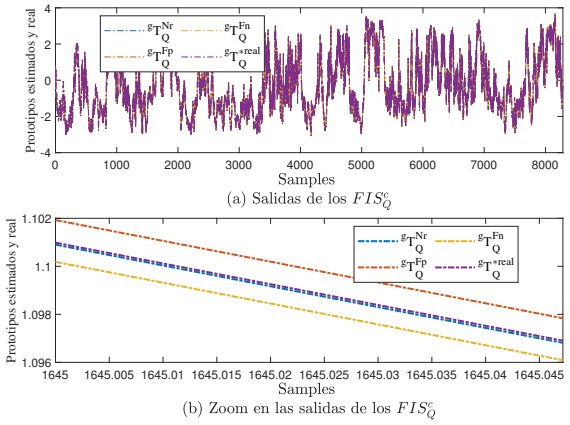


(a) ESA con datos que contienen falsos negativos.

(b) Clasificación con datos que contienen falsos negativos.

(c) Clase obtenida por comparación directa del ESA con datos que contienen falsos negativos.

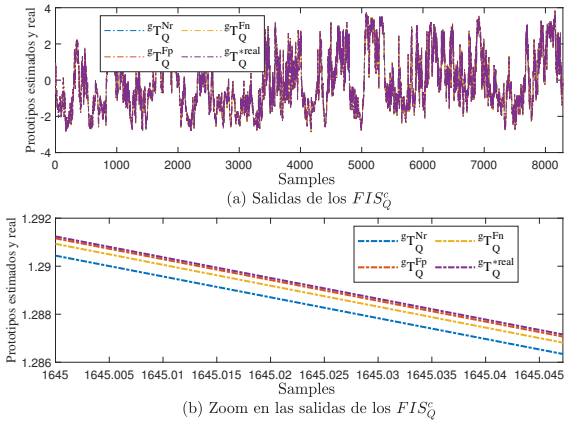
Figura 9: Errores entregados por el ESA tras minimizar la función de coste para la potencia activa.



(a) Salidas de los FIS_Q^c

(b) Zoom en las salidas de los FIS_Q^c

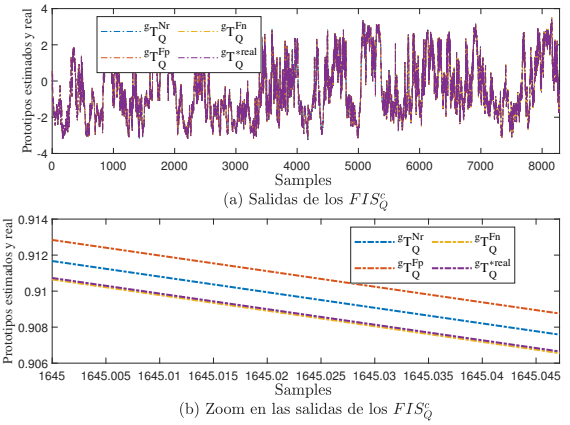
(a) Prototipo real y estimado por cada FIS tras ser evaluado con datos de operación normal



(a) Salidas de los FIS_Q^c

(b) Zoom en las salidas de los FIS_Q^c

(b) Prototipo real y estimado por cada FIS tras ser evaluado con datos que contienen falsos positivos

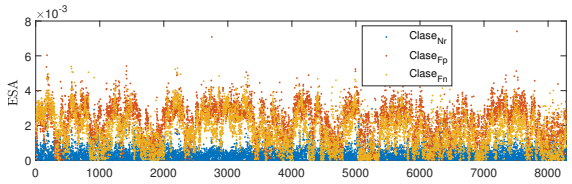


(a) Salidas de los FIS_Q^c

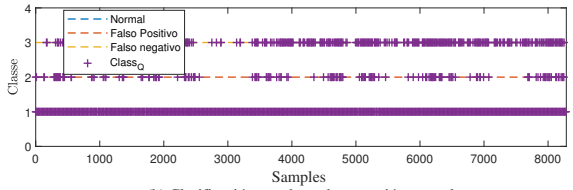
(b) Zoom en las salidas de los FIS_Q^c

(c) Prototipo real y estimado por cada FIS tras ser evaluado con datos que contienen falsos negativos

Figura 10: Prototipos obtenidos con inyección de datos falsos y operación normal en la potencia reactiva.

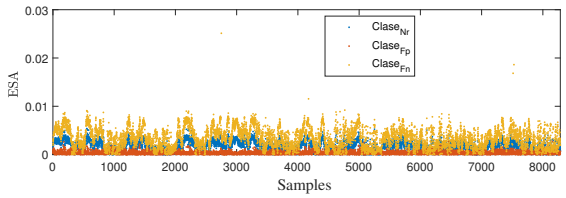


(a) ESA con datos de operación normal.

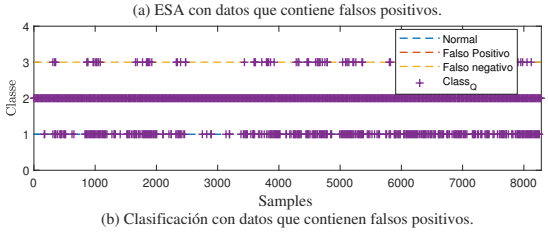


(b) Clasificación con datos de operación normal

(a) Clase obtenida por comparación directa del ESA con datos de operación normal.

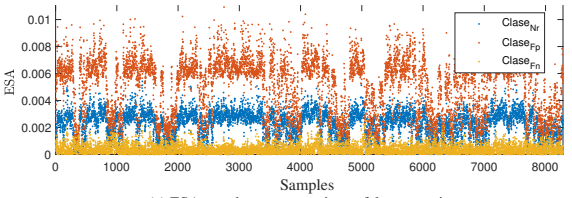


(a) ESA con datos que contiene falsos positivos.

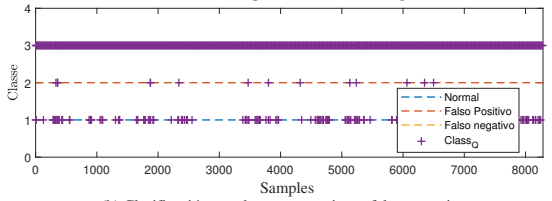


(b) Clasificación con datos que contienen falsos positivos.

(b) Clase obtenida por comparación directa del ESA con datos que contienen falsos positivos.



(a) ESA con datos que contienen falsos negativos.



(b) Clasificación con datos que contienen falsos negativos.

(c) Clase obtenida por comparación directa del ESA con datos que contienen falsos negativos.

Figura 11: Errores entregados por el ESA tras minimizar la función de coste para la potencia reactiva.

5. Conclusiones

Se ha diseñado un detector basado en clasificadores neuro-borrosos que determina la existencia de inyección de datos falsos en la potencia activa y reactiva simultáneamente de un aerogenerador de 3 MW. El clasificador neuro-borroso lo conforma una estructura de diversos sistemas de inferencia borrosos del tipo Takagi-Sugeno, donde los nuevos datos de entrada se proyectan sobre las componentes principales y se utilizan como entradas de cada FIS que estiman el prototipo general que lo define para su posterior clasificación. Se han efectuado pruebas en simulación, cambiando artificialmente los datos de las potencias. Los resultados de la clasificación presenta índices muy adecuados de precisión y sensibilidad.

Agradecimientos

Los autores agradecen a la Comisión Europea la financiación de este trabajo en el marco del proyecto DENiM. Este proyecto ha recibido financiación del programa de investigación e innovación Horizonte 2020 de la Unión Europea en virtud del acuerdo de subvención nº 958339.

The authors thanks to the European Commission for funding this work under project DENiM. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 958339.

Referencias

Alcalá, J., Cárdenas, V., Ramírez-López, A. R., Gudiño-Lau, J., 2011. Study of the bidirectional power flow in back - to - back converters by using linear and nonlinear control strategies. In: 2011 IEEE Energy Conversion Congress and Exposition. pp. 806–813.
DOI: 10.1109/ECCE.2011.6063853

Almasabi, S., Alsuwian, T., Javed, E., Irfan, M., Jalalah, M., Aljafari, B., Harraz, F. A., 2021. A novel technique to detect false data injection attacks on phasor measurement units. Sensors 21 (17).
URL: https://www.mdpi.com/1424-8220/21/17/5791
DOI: 10.3390/s21175791

Bordons, C., Garcia-Torres, F., Ridao, M., 2020. Model Predictive Control of Microgrids. Advances in Industrial Control. Springer Verlag.

Chicaiza, W., Rodríguez, F., Sánchez, A. J., Escaño, J. M., 2021. Deteccion de fallos en datos para la decisión de acción de aprendizaje de gemelos digitales de sistemas de energía. In: XVI Simposio CEA de Control Inteligente: Libro de Actas. Universidad de Las Palmas de Gran Canaria, Las Palmas de Gran Canaria, Spain, pp. 55–60.

Chiu, S. L., may 1994. Fuzzy model identification based on cluster estimation. J. Intell. Fuzzy Syst. 2 (3), 267–278.

IEA, 2021. World Energy Outlook 2021.

itUser Tech & Business, 2022. El 16 % de la industria europea, en riesgo por la escasez de equipos de seguridad ot.

Jang, J.-S., 1993. Anfis: adaptive-network-based fuzzy inference system. IEEE Transactions on Systems, Man, and Cybernetics 23 (3), 665–685.
DOI: 10.1109/21.256541

Jollife, I. T., Cadima, J., 4 2016. Principal component analysis: a review and recent developments. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 374.
DOI: 10.1098/RSTA.2015.0202

Qu, Z., Dong, Y., Qu, N., Li, H., Cui, M., Bo, X., Wu, Y., Mugemanyi, S., 2021. False data injection attack detection in power systems based on cyber-physical attack genes. Frontiers in Energy Research 9.
DOI: 10.3389/fenrg.2021.644489

Shahid, M. A., Ahmad, F., Albogamy, F. R., Hafeez, G., Ullah, Z., 2022. Detection and prevention of false data injection attacks in the measurement infrastructure of smart grids. Sustainability 14 (11).
URL: https://www.mdpi.com/2071-1050/14/11/6407
DOI: 10.3390/su14116407

Takagi, T., Sugeno, M., 1983. Derivation of fuzzy control rules from human operator's control actions. IFAC Proceedings Volumes 16 (13), 55–60, iFAC Symposium on Fuzzy Information, Knowledge Representation and Decision Analysis, Marseille, France, 19-21 July, 1983.
DOI: https://doi.org/10.1016/S1474-6670(17)62005-6

Tan, S., Xie, P., Guerrero, J. M., Vasquez, J. C., 2022. False data injection cyber-attacks detection for multiple dc microgrid clusters. Applied Energy 310, 118425.
DOI: https://doi.org/10.1016/j.apenergy.2021.118425

Wold, S., Esbensen, K., Geladi, P., 1987. Principal component analysis. Chemometrics and Intelligent Laboratory Systems 2 (1), 37–52, proceedings of the Multivariate Statistical Workshop for Geologists and Geochemists.
DOI: https://doi.org/10.1016/0169-7439(87)80084-9

Wolf, M., Serpanos, D., 2020. False Data Injection Attacks. Springer International Publishing, Cham, pp. 73–83.
URL: https://doi.org/10.1007/978-3-030-25808-5-6
DOI: 10.1007/978-3-030-25808-5-6