

Jornadas de Automática

Modelos de ensamblaje con redes neuronales para la identificación de sistemas

De La Peña, Ángel^a, Irigoyen, Eloy^{a,*}, Larrea, Mikel^b

^aUPV/EHU, Escuela de Ingeniería de Bilbao, Pl. Ingeniero Torres Quevedo, 1, 48013, Bilbao, España.

^bUPV/EHU, Escuela de Ingeniería de Gipuzkoa, Plaza Europa, 1, 20018, Donostia / San Sebastián, España.

To cite this article: De La Peña, Ángel, Irigoyen, Eloy, Larrea, Mikel. 2025. Ensemble models with neural networks for system identification.. Jornadas de Automática, 46. <https://doi.org/10.17979/ja-cea.2025.46.12205>

Resumen

Este trabajo se centra en la obtención de modelos dinámicos complejos que puedan integrarse en un controlador predictivo basado en modelo (MPC). El objetivo es modelar sistemas no lineales y multivariantes mediante redes neuronales NARX (redes autorregresivas no lineales con entradas exógenas). Para mejorar la precisión y robustez del modelo generado, se propone una estrategia basada en modelos de ensamblaje (*ensemble models*), combinando múltiples redes entrenadas de forma independiente. Se llevan a cabo dos experimentos: el primero, utilizando datos sintéticos de diversa naturaleza, analiza el impacto de la diversidad en el conjunto de modelos; el segundo, empleando datos reales de un reactor químico, evalúa la aplicabilidad del enfoque en entornos reales con múltiples variables. En ambos casos, se demuestra que los métodos de ensamblaje mejoran el rendimiento en comparación con los modelos individuales.

Palabras clave: Identificación de sistemas no lineales, Redes neuronales, Identificación y modelado, Modelos ensamblados, Validación de modelos

Ensemble models with neural networks for system identification

Abstract

This work focuses on the development of complex dynamic models that can be integrated into a model predictive controller (MPC). The aim is to model nonlinear and multivariable systems using NARX neural networks (Nonlinear Autoregressive Networks with Exogenous Inputs). To improve the accuracy and robustness of the model obtained by the network, an ensemble modelling strategy is proposed, combining multiple independently trained networks. Two experiments are conducted: the first uses synthetic data of different natures to analyze the impact of diversity in the ensemble; the second one uses real data from a chemical reactor to assess the applicability of the approach in real-world multivariable environments. In both cases, it is shown that ensemble methods improve performance compared to individual models.

Keywords: Nonlinear system identification, Neural networks, Identification and modelling, Ensemble models, Model validation

1. Introducción

En la actualidad, el control y la automatización cada vez están más presentes en todos los ámbitos. La necesidad de automatizar y desarrollar sistemas inteligentes desempeña un papel fundamental en el progreso tecnológico, especialmente en el control de sistemas cada vez más complejos. Para abordar esta complejidad, es imprescindible desarrollar técnicas de

control cada vez más efectivas, y que sean capaces de trabajar con sistemas dinámicos no lineales con múltiples variables y objetivos Sasidaran and Raja (2022).

En esta tarea destacan los controladores predictivos basados en modelos (MPC) Schwenzer et al. (2021). Los modelos permiten capturar las interacciones entre diversas variables y dinámicas del sistema. Estos controladores son capaces de

*Autor para correspondencia: eloy.irigoyen@ehu.eus
Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)

predecir el comportamiento que va a tener la planta a controlar mediante un modelo interno de la planta. Para esto, es necesario un modelo matemático preciso, pero su obtención no es tarea trivial.

El aprendizaje automático, sustentado en redes neuronales, ofrece una solución prometedora para modelar estos sistemas Ibrahem et al. (2021). Entre las arquitecturas más destacadas para este propósito se encuentran las redes neuronales recurrentes de tipo NARX (redes autorregresivas no lineales con entradas exógenas), que son especialmente eficaces para modelar dinámicas temporales y procesos dependientes del tiempo, por lo que serán las empleadas Cheng and Low (2023).

A pesar de su flexibilidad y capacidad de generalización, las redes neuronales presentan limitaciones inherentes, como la susceptibilidad al sobreajuste o la alta variabilidad en su rendimiento Lee and Kang (2023). Para superar estos desafíos, se han desarrollado enfoques basados en modelos de ensamblaje (*ensemble models*), cuya principal ventaja radica en su capacidad para combinar múltiples modelos, reducir la varianza y mejorar la robustez de las predicciones Barbez et al. (2020).

1.1. Modelos de ensamblaje

La idea de los modelos de ensamblaje tiene sus raíces en 1785, cuando el matemático y revolucionario francés Marquis de Condorcet publicó el Teorema del Jurado. Este teorema demostraba matemáticamente que la probabilidad de que un grupo de personas tome una decisión correcta puede superar el juicio de un solo experto Ganaie et al. (2022). Inspirados en esta idea, los modelos de ensamblaje buscan replicar este principio combinando las fortalezas de múltiples modelos.

La mejora de rendimiento conseguida por los modelos de ensamblaje se basa principalmente en la diversidad de los modelos base. La diversidad se refiere a la diferencia en el comportamiento de los modelos que componen el conjunto. En otras palabras, los modelos individuales no deben cometer los mismos errores, sino que deben ser complementarios entre sí. Si cada modelo base interpreta los datos de una manera distinta, se consigue una mayor generalización del modelo final. Además, mayor diversidad implica menor riesgo de sobreajuste Obregon and Jung (2022).

Existen diversos métodos para conseguir aumentar la diversidad de los modelos base, y estos métodos a su vez se pueden combinar entre sí para favorecer aún más esta diversidad Nugroho et al. (2023). Destacar los siguientes: generar diferentes subconjuntos de datos de entrenamiento para cada modelo base, cambiar la topología de los modelos (número de capas, neuronas, hojas, ramas ...), distintas inicializaciones, emplear distintas técnicas (Redes Neuronales, árboles de decisión, métodos estadísticos...), y por último la división del espacio de características, es decir, entrenar varios modelos utilizando subconjuntos distintos de variables, en lugar de emplear todas las variables disponibles en cada modelo.

Aumentar la diversidad no siempre es una tarea sencilla, ya que, cuanto mayor sea la diversidad del conjunto, mayor será la complejidad del ensamblaje final. Esta dificultad puede provocar que los modelos resultantes no alcancen todo su potencial.

Existen dos tipos de modelos de ensamblaje: homogéneos y heterogéneos Mohammed and Kora (2023). Los primeros

están compuestos únicamente por modelos del mismo tipo, por contra, los modelos heterogéneos están formados por distintos tipos de algoritmos. Para obtener dinámicas complejas se prefieren emplear modelos base homogéneos Roveda et al. (2020); Wu et al. (2019). En general, es más sencillo obtener buenos resultados con un ensamblaje homogéneo que con uno heterogéneo, ya que las predicciones tenderán a ser más similares entre sí, lo que facilita el proceso de ensamblaje. Aunque resulta más difícil lograr diversidad utilizando modelos base homogéneos, esta sigue siendo una opción viable Roveda et al. (2020).

Existen diversos métodos de ensamblaje: *bagging*, *boosting*, *stacking* ... Destacar el *bagging* Nanni et al. (2023), el cual es el método más empleado y sencillo de todos. Su funcionamiento se basa en entrenar varios modelos sobre subconjuntos de datos diferentes, cada modelo se entrena de manera independiente, y sus predicciones se combinan posteriormente, si es un problema de regresión se realizará un promedio. Cuando se trabaja con modelos homogéneos el *bagging* ha demostrado ofrecer buenos resultados Wu et al. (2019).

Entre las variantes más habituales del *bagging* se encuentran el *bagging* simple, que consiste en calcular la media aritmética de las predicciones de los modelos base, y el *Weighted Average Bagging* (WAB), que emplea una media ponderada en la que los pesos asignados a cada modelo dependen de su rendimiento. Este rendimiento suele medirse mediante el error cuadrático medio (MSE).

1.2. Objetivos

Este trabajo tiene como objetivo principal desarrollar una solución eficaz para mejorar el rendimiento de las redes neuronales en la obtención de modelos no lineales multivariantes. Para ello, se propone implementar un enfoque basado en modelos de ensamblaje que aproveche las ventajas de las redes NARX y técnicas de combinación de modelos, maximizando la precisión y la robustez en entornos de alta complejidad. Se usará un ensamblaje homogéneo ya que se quieren modelar dinámicas complejas.

A su vez, al emplear modelos homogéneos se combinarán con *bagging*. Se emplearán las dos técnicas de ensamblaje basadas en *bagging* comentadas: *Bagging* simple y *Weighted Average Bagging* (WAB). La elección de estos métodos se justifica por su sencillez de implementación y su eficacia contrastada en conjuntos homogéneos como las redes NARX Nugroho et al. (2023), ya que permiten reducir la varianza y mejorar la robustez del modelo combinado sin incrementar en exceso la complejidad computacional.

En la primera sección se trabajará con datos generados sintéticamente a partir de una función matemática compleja, poniendo el foco en la diversidad de las características de los datos. Por otro lado, en la segunda prueba se utilizarán datos reales obtenidos de un reactor químico. Esta prueba se centrará en trabajar con múltiples variables. Finalmente, se analizarán los resultados obtenidos y se presentarán las conclusiones derivadas del estudio.

2. Entrenamiento con datos de distinta naturaleza

En esta primera prueba se quieren entrenar múltiples modelos para que aprendan la dinámica no lineal de una función

(Ecuación 1) descrita en Harris (1994). Sin embargo, no se busca que todos los modelos aprendan a partir de los mismos datos de entrada, sino que se pretende entrenarlos con datos de distintas naturalezas. El objetivo final es combinar estos modelos para lograr una predicción más precisa de la dinámica completa del sistema.

$$y(k+1) = \frac{1,5 \cdot y(k) \cdot y(k-1)}{1 + y(k)^2 + y(k-1)^2} + 0,7 \cdot \sin[0,5 \cdot (y(k) + y(k-1))] \cdot \cos[0,5 \cdot (y(k) + y(k-1))] + 1,2 \cdot u(k) \quad (1)$$

2.1. Desarrollo de la prueba

No solo se busca introducir diversidad mediante el uso de distintos conjuntos de entrenamiento, sino también variando la estructura de las redes NARX: el número de neuronas, el número de capas y los retardos tanto en la entrada como en la realimentación. Con este fin, se definieron múltiples configuraciones estructurales, que incluyen una o dos capas ocultas con las siguientes combinaciones de neuronas: [5], [10], [20], [5, 5], [10, 10], [20, 20]; retardos en la realimentación de 1 y 2 pasos; y retardos en la entrada de 0, 1 y 2 pasos. Estas combinaciones dan lugar a un total de 36 configuraciones distintas.

Para alcanzar este objetivo, se siguen los siguientes pasos. En primer lugar, se generan datos de entrada con distintas naturalezas para entrenar diversos modelos base. A continuación, se construye una señal de validación que combina las características de los conjuntos anteriores. Posteriormente, cada modelo individual intenta predecir dicha señal, y a partir de sus predicciones se construye el modelo de ensamblaje. Finalmente, se evalúa la efectividad de este nuevo modelo combinado.

Los datos son generados sintéticamente. Se establecen dos naturalezas diferenciadas: permanente y transitoria (Figura 1). La primera tiende a mantenerse en un estado constante, lo que permite capturar la dinámica del estado estacionario. La segunda, en cambio, presenta cambios más frecuentes, facilitando la representación de las variaciones propias del estado transitorio.

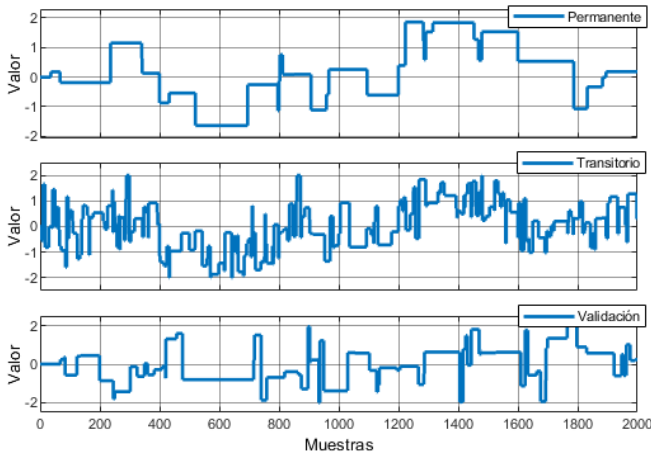


Figura 1: Señales de entrada para el primer experimento

Se entrenarán todas las estructuras para ambas naturalezas. Para cada combinación de arquitectura y tipo de entrada, se entrenarán 50 modelos con diferentes inicializaciones, y se conservará aquel que obtenga el menor error cuadrático medio (MSE). De este modo, se incrementa la exploración del espacio de soluciones, reduciendo la probabilidad de que el entrenamiento quede atrapado en un mínimo local poco representativo.

Una vez finalizado el entrenamiento, se seleccionan los cinco mejores modelos obtenidos para cada una de las dos naturalezas de datos empleadas. En total, se conservan diez modelos, que serán utilizados posteriormente para construir el modelo de ensamblaje.

2.2. Creación del ensamblado y resultados

Tras entrenar los modelos individuales, se procede a evaluarlos utilizando datos más diversos con el fin de verificar su capacidad para modelar con precisión la dinámica completa del sistema. Para ello, se genera aleatoriamente una señal de validación (Figura 1) que incluye tanto transiciones frecuentes, que permiten analizar el comportamiento en estados transitorios, como periodos prolongados constantes, útiles para evaluar el estado permanente. De este modo, se representa una amplia variedad de situaciones en las que el sistema podría encontrarse. Definida esta señal, se comprueba cómo cada modelo individual es capaz de predecirla. Los resultados de estas predicciones se presentan en la Tabla 1.

Tabla 1: Errores MSE de los modelos base al intentar predecir la señal de validación para el ensamblado. En la parte inferior se incluyen también los resultados de los métodos de ensamblaje. Se resalta el mejor modelo base y el mejor método de ensamblaje.

Modelos base	MSE
1º Permanente	0.6634
2º Permanente	0.0742
3º Permanente	0.1542
4º Permanente	0.2442
5º Permanente	0.1456
1º Transitorio	0.3123
2º Transitorio	0.9542
3º Transitorio	0.9599
4º Transitorio	0.2106
5º Transitorio	0.4691
Modelos de ensamblaje	
Bagging simple	0.1152
WAB	0.0290

Se ha realizado el ensamblaje utilizando *bagging* simple y WAB, asignando en este último los pesos en función del MSE. Tras realizar el ensamblaje los resultados se presentan en la Tabla 1. El *bagging* simple no logra superar al mejor modelo base (2º Permanente), aunque sí obtiene un rendimiento superior al del resto de modelos. Mientras tanto, el método WAB consigue mejorar los resultados de todos los modelos base, incluyendo al mejor individual.

Aunque no se ha cuantificado de forma explícita la diversidad entre los modelos base, su impacto positivo puede observarse de forma indirecta en el rendimiento del ensamblado.

El hecho de que el modelo combinado obtenga mejores resultados que los modelos individuales sugiere que la estrategia de introducir diversidad ha sido beneficiosa. En este experimento, dicha diversidad se ha fomentado mediante la variación de las señales de entrada (naturalezas transitoria y permanente) y mediante la utilización de distintas topologías para las redes NARX (número de capas, neuronas y retardos). Es precisamente la diferencia en los modelos lo que permite que se complementen entre sí y se reduzca la varianza del modelo final.

3. Pruebas en una planta real

En este nuevo experimento, se pretende aplicar las técnicas desarrolladas a una planta real, utilizando datos auténticos con el objetivo de comprobar su efectividad en un sistema multivariable.

La capacidad de modelar sistemas reales complejos resulta fundamental en entornos industriales, ya que permite diseñar estrategias de control que garanticen la eficiencia y la seguridad de la planta. En este contexto, los modelos de ensamblaje juegan un papel clave, al mejorar la precisión y la robustez de las predicciones generadas por las redes neuronales, lo cual resulta especialmente relevante ante datos con ruido, variabilidad o un número elevado de características. Estos modelos ofrecen una alternativa eficaz para abordar la complejidad de los procesos multivariables no lineales, como los que se presentan en reactores químicos.

3.1. Reactor químico

En esta segunda fase experimental se va a desarrollar el estudio y análisis sobre un proceso real que tiene lugar en un reactor químico. Para ello se hará uso de la base de datos proporcionada por Fonollosa (2015), donde se especifica las características del proceso. Este conjunto de datos fue obtenido en una plataforma de suministro de gases del laboratorio ChemoSignals, perteneciente al BioCircuits Institute de la Universidad de California en San Diego.

Este conjunto de datos contiene las dinámicas adquiridas de 16 sensores químicos expuestos a mezclas de gases con niveles de concentración variables. En la cámara se generaron dos gases, etileno y monóxido de carbono (CO) en aire, que serán las variables objetivo que la red neuronal deberá predecir.

La base de datos se compone de un experimento en el que se realizaron mediciones de forma continua durante un período aproximado de 12 horas sin interrupciones. Las señales de los 16 sensores se adquirieron de manera continua mientras se variaban los niveles de concentración de diferentes gases. La frecuencia de muestreo fue de 100 Hz, lo que dio lugar a la generación de más de cuatro millones de datos a lo largo del experimento. Dado que cada sensor posee un rango de funcionamiento distinto, se optó por normalizar los datos.

Aprovechando la gran cantidad de datos disponible, se plantea emplear la mayor proporción posible de muestras para el entrenamiento y la validación en la etapa de generación de los modelos base. Por este motivo, se implementará una validación cruzada (*cross-validation*). Este procedimiento consiste en evaluar el rendimiento del modelo dividiendo el conjunto de datos en varias particiones (*folds*) y realizando múltiples

entrenamientos. En cada iteración, se alternan los datos entre los conjuntos de entrenamiento y validación, procurando que todos los datos sean utilizados al menos una vez en la validación.

La realización de una única validación cruzada no siempre permite aprender adecuadamente todos los patrones presentes en los datos, lo que puede dar lugar a resultados inestables y de baja precisión. Sin embargo, al promediar los resultados obtenidos en múltiples ejecuciones, se logran estimaciones más robustas y precisas, reduciendo así la varianza del modelo Barkhordari et al. (2023). Por este motivo, la combinación de la validación cruzada con técnicas de ensamblaje presenta una sinergia muy favorable.

3.2. Desarrollo y resultados

A pesar de tener datos secuenciales, se opta por la validación cruzada estándar, sin alterar el orden de las muestras Wu et al. (2019). Esto es posible porque el sistema responde a cambios aleatorios en las variables de entrada. No se trata de una serie temporal con patrones de estacionalidad o tendencias a largo plazo. En este caso, las redes neuronales utilizadas son capaces de aprender las dinámicas del sistema con solo unas pocas muestras anteriores. En concreto, únicamente necesitan conocer las muestras incluidas en los retardos, que en este trabajo alcanzan un máximo de tres. Por este motivo, al crear las particiones, se reservan las tres primeras muestras para los retardos y el entrenamiento comienza a partir de la cuarta. De este modo, se garantiza la independencia total entre los datos de entrenamiento y los de validación.

Los datos se dividirán en dos conjuntos: el 80 % se utilizará para entrenar y validar los modelos, y el 20 % restante se reservará para realizar el ensamblaje y llevar a cabo la prueba final. El 80 % destinado al entrenamiento se dividirá, a su vez, en 10 particiones que se emplearán en el proceso de validación cruzada. Esta estrategia de particionado se ilustra en la Figura 2.

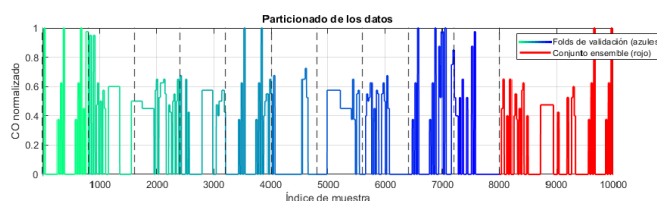


Figura 2: Particionado de los datos

Los modelos se entrenan utilizando nueve de las particiones y se validan con la restante. Este proceso se repite diez veces, de forma que todos los datos participan tanto en el entrenamiento como en la validación, permitiendo aprender todas las dinámicas presentes en el sistema. La validación cruzada se realiza por separado para cada arquitectura. Al finalizar, se seleccionan las cinco mejores redes de cada conjunto de entrenamiento, asegurando así una representación equilibrada de todos los datos. Dado que se utilizan diez particiones, se obtienen un total de 50 modelos que se emplearán para construir el modelo de ensamblaje.

Sin embargo, para poder realizar el ensamblaje, todos los modelos individuales deben predecir sobre los mismos datos,

para poder combinar sus salidas. En este punto, entra en juego el 20 % de datos previamente reservados. Los modelos base intentan predecir estas nuevas muestras, y los resultados obtenidos se muestran en la Tabla 2. Se observa que el rendimiento sobre estos datos reservados es similar al obtenido durante la validación cruzada.

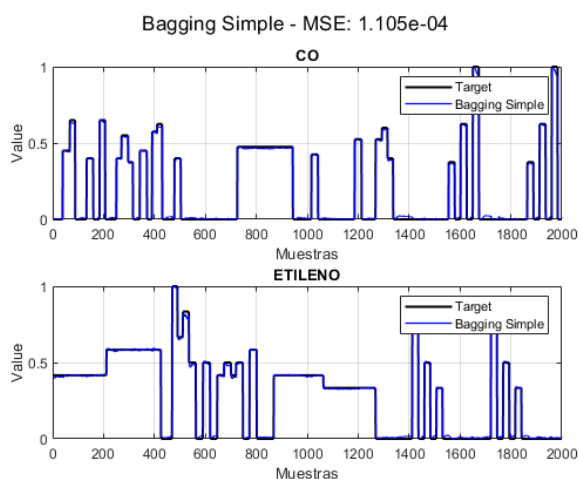


Figura 3: Resultado del ensamblaje empleando *bagging* simple

Finalmente, con estas últimas predicciones se lleva a cabo el ensamblaje, utilizando los mismos métodos empleados anteriormente: *bagging* simple (Figura 3) y WAB (Figura 4). El método WAB requiere una adaptación específica para el caso de múltiples salidas. En este caso, el error se calcula individualmente para cada variable de salida, y los pesos se asignan en función de dichos errores. Esto se debe a que un modelo puede ofrecer un buen rendimiento para una de las variables, pero no haber aprendido adecuadamente la dinámica de la otra. Por este motivo, los pesos se calculan de manera independiente. Los resultados del ensamblaje superan a los obtenidos por los modelos individuales.

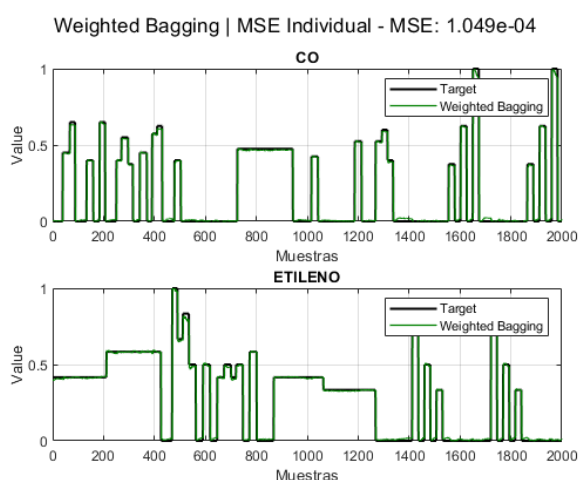


Figura 4: Resultado del ensamblaje empleando WAB

El método de *bagging* simple obtuvo un error cuadrático

medio (MSE) de $1,105 \times 10^{-4}$, mientras que el método WAB alcanzó un valor aún menor de $1,049 \times 10^{-4}$. En comparación, el mejor modelo individual presentó un MSE de $1,113 \times 10^{-4}$. Por tanto, ambos métodos de ensamblaje lograron superar el rendimiento del mejor modelo base.

4. Conclusiones

En este trabajo se han llevado a cabo dos experimentos con el objetivo de evaluar la eficacia de modelos de ensamblaje basados en redes neuronales NARX para la identificación de sistemas dinámicos. El primero se centró en datos sintéticos, permitiendo analizar el impacto de la diversidad estructural y de entrada en el rendimiento de los modelos. El segundo experimento se desarrolló con datos reales procedentes de un sistema multivariable, con el fin de validar la aplicabilidad del enfoque en un entorno complejo y representativo.

En el primer experimento se ha demostrado que el ensamblaje es capaz de mejorar el rendimiento de los modelos individuales. En el segundo experimento también se observó una mejora significativa. En conjunto, ambos resultados ponen de manifiesto que fomentar la diversidad entre los modelos base, tanto mediante la variación en los datos de entrenamiento como en la arquitectura de las redes NARX, permite construir un modelo más robusto y eficaz.

En cuanto a las técnicas de ensamblaje utilizadas, se observa que el método de *bagging* simple mejora el rendimiento promedio, aunque no siempre logra superar al mejor modelo base. En cambio, el método *Weighted Average Bagging* alcanza resultados superiores a los de todos los modelos individuales.

Tras trabajar con datos reales, se puede afirmar que el enfoque propuesto no solo es válido en entornos simulados, sino que también resulta aplicable a condiciones reales de operación, donde los datos no son perfectos y existen ruidos en las mediciones, interferencias externas y otros factores que suponen un reto adicional. Los modelos han demostrado ser capaces de adaptarse a la dinámica de la planta. Además, se ha comprobado que el sistema es eficaz incluso en escenarios con múltiples variables de salida.

Finalmente, al comparar ambas pruebas se puede concluir que el modelo de ensamblaje no puede superar las limitaciones inherentes de los modelos individuales: su rendimiento depende directamente de la calidad de los modelos base que lo componen. Si estos modelos individuales capturan adecuadamente la dinámica del sistema y han sido correctamente entrenados, el ensamblaje tiende a mejorar la precisión general. Sin embargo, si los modelos base son deficientes o están mal ajustados, el rendimiento del conjunto será igualmente limitado. No obstante, en términos generales, el uso de modelos de ensamblaje suele proporcionar una mejora con respecto al rendimiento individual.

Agradecimientos

Este trabajo se ha desarrollado en el marco del proyecto PID2020-120087GB-C22 financiado por el Ministerio de Ciencia e Innovación del Gobierno de España. (AEI / <http://dx.doi.org/10.13039/501100011033>)

Tabla 2: Valores de MSE de los 5 mejores modelos de cada conjunto de validación. Resaltado el mejor modelo base.

	1º modelo	2º modelo	3º modelo	4º modelo	5º modelo
Validación 1	$1,433 \times 10^{-4}$	$1,113 \times 10^{-4}$	$1,560 \times 10^{-4}$	$1,749 \times 10^{-4}$	$1,318 \times 10^{-4}$
Validación 2	$1,224 \times 10^{-4}$	$1,781 \times 10^{-4}$	$2,480 \times 10^{-4}$	$3,159 \times 10^{-4}$	$1,434 \times 10^{-4}$
Validación 3	$2,010 \times 10^{-4}$	$1,248 \times 10^{-4}$	$1,631 \times 10^{-4}$	$1,836 \times 10^{-4}$	$1,868 \times 10^{-4}$
Validación 4	$3,118 \times 10^{-4}$	$1,881 \times 10^{-4}$	$1,506 \times 10^{-4}$	$2,024 \times 10^{-4}$	$1,726 \times 10^{-4}$
Validación 5	$1,552 \times 10^{-4}$	$1,635 \times 10^{-4}$	$1,668 \times 10^{-4}$	$1,958 \times 10^{-4}$	$2,255 \times 10^{-4}$
Validación 6	$1,851 \times 10^{-4}$	$1,539 \times 10^{-4}$	$1,624 \times 10^{-4}$	$2,059 \times 10^{-4}$	$2,273 \times 10^{-4}$
Validación 7	$2,624 \times 10^{-4}$	$2,471 \times 10^{-4}$	$1,603 \times 10^{-4}$	$1,830 \times 10^{-4}$	$1,534 \times 10^{-4}$
Validación 8	$1,316 \times 10^{-4}$	$1,500 \times 10^{-4}$	$1,733 \times 10^{-4}$	$2,185 \times 10^{-4}$	$2,019 \times 10^{-4}$
Validación 9	$1,427 \times 10^{-4}$	$1,389 \times 10^{-4}$	$1,626 \times 10^{-3}$	$2,840 \times 10^{-4}$	$1,913 \times 10^{-4}$
Validación 10	$1,851 \times 10^{-4}$	$1,439 \times 10^{-4}$	$1,795 \times 10^{-4}$	$1,994 \times 10^{-4}$	$1,288 \times 10^{-4}$

Referencias

- Barbez, A., Khomh, F., Guéhéneuc, Y.-G., 3 2020. A machine-learning based ensemble method for anti-patterns detection. *Journal of Systems and Software* 161, 110486.
DOI: 10.1016/j.jss.2019.110486
- Barkhordari, M. S., Armaghani, D. J., Asteris, P. G., 2023. Structural damage identification using ensemble deep convolutional neural network models. *Computer Modeling in Engineering & Sciences* 134, 835–855.
DOI: 10.32604/cmescs.2022.020840
- Cheng, A., Low, Y. M., 10 2023. Improved generalization of narx neural networks for enhanced metamodeling of nonlinear dynamic systems under stochastic excitations. *Mechanical Systems and Signal Processing* 200.
DOI: 10.1016/j.ymssp.2023.110543
- Fonollosa, J., 2015. Gas sensor array under dynamic gas mixtures. *UCI Machine Learning Repository*, DOI: <https://doi.org/10.24432/C5WP4C>.
- Ganaie, M., Hu, M., Malik, A., Tanveer, M., Suganthan, P., 10 2022. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence* 115, 105151.
DOI: 10.1016/j.engappai.2022.105151
- Harris, C. (Ed.), 1994. *Advances in Intelligent Control*. CRC Press.
- Ibrahim, I. M., Akhrif, O., Moustapha, H., Staniszewski, M., 9 2021. Nonlinear generalized predictive controller based on ensemble of narx models for industrial gas turbine engine. *Energy* 230.
DOI: 10.1016/j.energy.2021.120700
- Lee, U., Kang, N., 7 2023. Adaptive neural network ensemble using prediction frequency. *Journal of Computational Design and Engineering* 10, 1547–1560.
DOI: 10.1093/jcde/qwad071
- Mohammed, A., Kora, R., 2 2023. A comprehensive review on ensemble deep learning: Opportunities and challenges. Leído el 08/10/2024.
DOI: 10.1016/j.jksuci.2023.01.014
- Nanni, L., Loreggia, A., Brahnam, S., 10 2023. Comparison of different methods for building ensembles of convolutional neural networks. *Electronics* 12, 4428.
DOI: 10.3390/electronics12214428
- Nugroho, H. A., Astuty, E. Y., Subiantoro, A., Kusumoputro, B., 2023. Ensemble deep learning narx for estimating time series of earthquake occurrence. In: *2023 3rd International Conference on Robotics, Automation and Artificial Intelligence, RAAI 2023*. Institute of Electrical and Electronics Engineers Inc., pp. 301–305.
DOI: 10.1109/RAAI59955.2023.10601278
- Obregon, J., Jung, J. Y., 1 2022. *Explanation of ensemble models*. Elsevier Inc., pp. 51–72.
DOI: 10.1016/B978-0-323-85648-5.00011-6
- Roveda, L., Maskani, J., Franceschi, P., Abdi, A., Braghin, F., Tosatti, L. M., Pedrocchi, N., 11 2020. Model-based reinforcement learning variable impedance control for human-robot collaboration. *Journal of Intelligent and Robotic Systems: Theory and Applications* 100, 417–433.
DOI: 10.1007/s10846-020-01183-3
- Sasidaran, S., Raja, H. V., 2022. Recent trends in model predictive control. *Tech. rep.*
URL: <https://www.researchgate.net/publication/369201653>
- Schwenzer, M., Ay, M., Bergs, T., Abel, D., 11 2021. Review on model predictive control: an engineering perspective.
DOI: 10.1007/s00170-021-07682-3
- Wu, Z., Tran, A., Ren, Y. M., Barnes, C. S., Chen, S., Christofides, P. D., 5 2019. Model predictive control of phthalic anhydride synthesis in a fixed-bed catalytic reactor via machine learning modeling. *Chemical Engineering Research and Design* 145, 173–183.
DOI: 10.1016/j.cherd.2019.02.016