Implementación de algoritmos de entrenamiento de modelos probabilísticos para predecir el rendimiento de celdas solares de perovskita con datos incompletos

Daniel Andrés Cerro Ramos y Yan Carlos Velasquez Meneses

Trabajo de Grado para Optar al título de Ingenieros Electrónicos

Director

Franklin Alexander Sepúlveda Sepúlveda

Doctor en Ingeniería Electrónica

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Ingeniería Electrónica

Bucaramanga

2024

Dedicatoria

Este trabajo representa el esfuerzo y la dedicación de los años recorridos en la universidad. Cada paso, cada desafío y cada logro me ha llevado a este momento, en el que puedo mirar hacia atrás con orgullo y satisfacción. Primero, quiero agradecer a Dios, quien me ha guiado y fortalecido en cada paso de este camino. A mi madre, Mónica, por su amor incondicional, su apoyo constante y por ser mi ejemplo de perseverancia. A mis hermanas, quienes siempre han estado a mi lado brindándome su cariño y motivación. A mis amigos, quienes han compartido conmigo momentos de alegría y aprendizaje, y a la selección de atletismo de la UIS, por enseñarme el valor del esfuerzo y la disciplina. Con gratitud y emoción, me siento orgulloso de estar más cerca de alcanzar mi meta de convertirme en ingeniero electrónico.

Daniel Andrés Cerro Ramos

Este proyecto ha sido un gran reto superado que representa la sabiduría y conocimientos adquiridos durante mi trayectoria académica. Lo dedico a mi madre, su compañía y palabras de apoyo fueron fuente de fortaleza para superar los desafíos presentados. Lo dedico a mis hermanos, a mi abuela, igualmente a aquellos amigos que hicieron de esta etapa muy especial, que me brindaron su sabiduría, acompañamiento y apoyo incondicional.

Yan Carlos Velasquez Meneses

Agradecimientos

Queremos agradecer a Dios, quien nos ha dado la sabiduría, entendimiento y fortalezas en esta etapa académica, y en nuestra vida en general. Agradecemos profundamente a nuestro director de proyecto, Franklin Sepúlveda, por brindarnos esta oportunidad que ha sido muy enriquecedora. Agradecemos al grupo de investigación Control, Electrónica, Modelado y Simulación (CEMOS) por recibirnos y darnos el privilegio de ser parte de su visión en la investigación de este proyecto. Igualmente, agradecemos a MINCIENCIAS-Colombia por el apoyo financiero en el marco del contrato 2022 0724 convocatoria 890, y a ICETEX-Colombia por la administración de los recursos.

Daniel Cerro y Yan Velasquez

Tabla de Contenido

Pág.

Introducción **11**

1. Objetivos **13**

1.1 Objetivo general 13

1.2 Objetivos específicos 13

2. Conceptos previos **14**

2.1 Modelos de imputación 14

2.1.1 MICE 14

2.1.2 GAIN 15

2.2 Modelos de mezcla Gaussiana 15

2.2.1 GMM 15

2.2.2 MGMM 16

2.3 Regresión mediante Modelos de Mezcla Gaussiana 16

2.3.1 Fundamentos Teóricos 16

2.4 Tipos de datos faltantes 19

3. Desarrollo de la solución **20**

3.1 Selección de parámetros 22

3.1.1 Preprocesamiento de datos 24

3.2 Implementación de modelos 24

3.2.1 Imputación con MICE 25

3.2.2 Imputación con GAIN 26

3.2.3 Modelo MGMM 28

3.2.3.1 Algoritmo ECM y Actualización de Parámetros. 30

3.2.4 Modelo GMM con GAIN y MICE 32

3.2.5 Predicción mediante Modelos de Mezcla Gaussiana 33

3.3 Métricas de evaluación 35

3.4 Análisis de resultados 37

3.4.1 Comparaciones GAIN y MICE 37

3.4.2 Desempeño de modelos de mezclas gaussianas GMM+MICE, GMM+GAIN y MGMM 39

4. Conclusiones 44

5. Recomendaciones 46

Referencias Bibliográficas 48

Apéndices 52

Lista de Figuras

Pág.

Figura 1 Esquema metodológico 21

Figura 2 Principales pasos utilizados en MICE 25

Figura 3 Arquitectura GAIN 27

Figura 4 Entrenamiento de modelos 29

Figura 5 Predicción de parámetros de rendimiento 34

Figura 6 Rendimiento de RMSE: (a) Base de datos 1 N= 5613 (b) Base de datos 2 N = 1070 38

Figura 7 Rendimiento de sMAPE: (a) Base de datos 1 N= 5613 (b) Base de datos 2 N = 1070 38

Figura 8 RMSE promedio K-Fold de JV_PCE: 40

Figura 9 MAPE promedio K-Fold de JV_PCE: 41

Lista de Tablas

Pág.

Tabla 1 Rendimiento de RMSE y sMAPE para la base de datos 1 37

Tabla 2 Rendimiento de RMSE y sMAPE para la base de datos 2 38

Tabla 3 Promedio de rendimientos de los k-Folds - RMSE y MAPE para el JV_PCE 40

Tabla 4 Promedio de rendimientos del k-Folds 2 - RMSE y MAPE para el JV_PCE 42

Lista de Apéndices

Apéndice A. Repositorio Github AEPRCES-231 52

Apéndice B. Tabla de desviaciones estándar del JV_PCE de cada k-fold 52

Apéndice C. Promedio de rendimientos de los k-Folds - RMSE y MAPE para el JV_Voc, JV_Jsc y JV_FF 52

Resumen

Título: Implementación de algoritmos de entrenamiento de modelos probabilísticos para predecir el rendimiento de celdas solares de perovskita con datos incompletos. *

Autor: Daniel Andrés Cerro Ramos, Yan Carlos Velasquez Meneses. **

Palabras Clave: Modelos probabilísticos, modelos de imputación de datos incompletos, celdas solares de perovskita, predicción, aprendizaje automático.

Descripción: La calidad de los datos utilizados para implementar modelos de aprendizaje automático es crucial para su funcionamiento óptimo. Sin embargo, en numerosos estudios de investigación, la recopilación de datos de múltiples fuentes con diferentes estructuras resulta en datos con valores faltantes o erróneos. Este problema está presente y afecta la base de datos existente para el diseño de celdas de perovskita, dificultando la aplicación de métodos de aprendizaje automático. En este proyecto, se abordó este problema en los datos faltantes de una base de datos de celdas solares de perovskita mediante la implementación de Modelos de Mezcla Gaussiana Conscientes de Faltantes (MGMM). Estos métodos permiten entrenar modelos probabilísticos directamente sobre datos incompletos. Para evaluar el desempeño de los MGMM, se compararon sus resultados con aquellos obtenidos al utilizar datos imputados mediante los algoritmos imputaciones múltiples mediante ecuaciones encadenadas (MICE) y El algoritmo de Redes de imputación generativas adversarias (GAIN). La evaluación del rendimiento de los algoritmos de imputación demostró un mejor desempeño del modelo MICE y se evidenció que el número de muestras y parámetros puede influir en el rendimiento de los algoritmos. Posteriormente, se utilizaron los datos imputados por cada modelo para evaluar las predicciones de los modelos de mezcla gaussiana (GMM) con las métricas error cuadrático medio (RMSE) y error porcentual medio absoluto (MAPE). Los resultados mostraron que el modelo GMM con datos imputados por MICE tuvo mejores predicciones en comparación con GAIN. La implementación del modelo MGMM con la base de datos de menor tamaño demostró que la distribución de los datos seleccionados impacta en el rendimiento del modelo, obteniendo resultados inferiores a medida que aumenta el porcentaje de valores faltantes. Este estudio demuestra que datos de celdas solares de perovskita pueden ser tratados eficazmente con modelos de imputación como MICE y GAIN, considerando la naturaleza y características de los datos.

---

* Trabajo de Grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: Franklin Alexander Sepulveda Sepulveda. PhD.

Abstract

Title: Implementation of probabilistic model training algorithms for predicting Perovskite solar cell performance with incomplete data. *

Author(s): Daniel Andrés Cerro Ramos, Yan Carlos Velasquez Meneses. **

Key Words: Probabilistic models, incomplete data imputation models, Perovskite solar cells, prediction, machine learning.

Description: The quality of the data used to implement machine learning models is crucial for their optimal performance. However, in numerous research studies, collecting data from multiple sources with different structures results in data with missing or erroneous values. This problem is present and affects the existing database for perovskite cell design, making it difficult to apply machine learning methods. In this project, this problem was addressed in the missing data of a perovskite solar cell database by implementing Gaussian Mixture Models Aware of Missing Values (MGMM). These methods allow training probabilistic models directly on incomplete data. To evaluate the performance of the MGMMs, their results were compared with those obtained when using data imputed using the Multiple Imputations via Chained Equations (MICE) and the Generative Adversarial Imputation Networks (GAIN) algorithms. The evaluation of the performance of the imputation algorithms showed a better performance of the MICE model and it was evidenced that the number of samples and parameters can influence the performance of the algorithms. Subsequently, the data imputed by each model were used to evaluate the predictions of the Gaussian mixture models (GMM) with the metrics root mean square error (RMSE) and mean absolute percentage error (MAPE). The results showed that the GMM model with MICE imputed data had better predictions compared to GAIN. Implementation of the MGMM model with the smaller database size showed that the distribution of the selected data impacts the model performance, obtaining lower results as the percentage of missing values increases. This study demonstrates that perovskite solar cell data can be effectively treated with imputation models such as MICE and GAIN, considering the nature and characteristics of the data.

---

* Bachelor Thesis

**Faculty of Physical and Mechanical Engineering, School of Electrical, Electronic and Telecommunications Engineering. Director Franklin Alexander Sepulveda Sepulveda. PhD.

Introducción

Las celdas solares de perovskita se están popularizando notablemente en el campo de la conversión de energía solar. Con una eficiencia de conversión de energía sin precedentes, estas celdas solares han desencadenado una verdadera revolución en la búsqueda de un rendimiento fotovoltaico superior. Su alta eficiencia, combinadas con el bajo costo de materiales y procesos, son los principales puntos a favor de estas celdas en comparación con el silicio comercial y otras celdas solares orgánicas e inorgánicas.

En el ámbito de la investigación se suele obtener información combinando conjuntos de datos múltiples y heterogéneos. Esto implica la utilización de más de un conjunto de datos proveniente de diversas fuentes o con diferentes estructuras, formatos y tipos de información. Sin embargo, esta combinación de diferentes fuentes a menudo resulta en conjuntos de datos incompletos o con valores faltantes, debido a un formato incoherente, un almacenamiento no documentado o una difusión inadecuada. Recientemente, se han reportado métodos de aprendizaje automático aplicados al diseño de celdas solares de perovskita. Aunque se dispone de un conjunto de datos de más de 40.000 registros con más de 100 parámetros (Jacobsson et al., 2021) , de celdas de perovskita, solamente 15 de ellos corresponden a observaciones completas. Por lo general, cuando se encuentran conjuntos de datos incompletos, se tiende a descartarlos, lo que resulta en una pérdida importante de información y menos datos para el entrenamiento de modelos de aprendizaje automático, ya que muchos de los métodos de aprendizaje automático no pueden manejar los valores faltantes (Hasan et al., 2021a).

Una forma de resolver este problema es mediante técnicas de imputación de datos, las cuales desempeñan un papel relevante en diversas áreas, incluida la ciencia de materiales. Sin embargo, el uso de datos imputados ya de por sí agrega el sesgo propio del mismo estimador de esos datos faltantes. En este proyecto se implementó un algoritmo de entrenamiento de modelos probabilísticos (GMM por sus siglas en inglés) adecuado para datos incompletos (Mccaw et al., 2022). Además, se evaluó el rendimiento de este tipo de métodos en el contexto de datos de síntesis de celdas solares de perovskita.

Los resultados de este proyecto permitirían comprender que el enfoque de diseño de celdas solares de perovskita resulta más eficiente y preciso, con el cual se obtendrían beneficios significativos en términos de una fabricación más eficiente y económica de estas celdas solares. Además, este enfoque también puede tener un impacto positivo en otros campos de la industria donde existe una falta de datos para implementar métodos de aprendizaje automático. También, se espera que este trabajo de investigación contribuya a estimar la posible mejora en el desempeño de los métodos de entrenamiento en datos incompletos en comparación con algunas técnicas de imputación de datos.

1. Objetivos

1.1 Objetivo general

Implementar un algoritmo de entrenamiento de modelos probabilísticos mediante el criterio de máxima verosimilitud a partir de datos incompletos que permita disminuir el error de predicción de desempeño de celdas solares de perovskita con respecto a métodos típicos de imputación de datos.

1.2 Objetivos específicos

- Implementar un algoritmo de entrenamiento de modelos de mezclas gaussianas mediante el algoritmo de maximización de la esperanza condicional adecuado para la problemática de datos incompletos.
- Comparar el desempeño del algoritmo mediante criterios de error de predicción estimado, respecto a métodos de imputación de datos faltantes reportados en el estado del arte (MICE, GAIN).

2. Conceptos previos

2.1 Modelos de imputación

La necesidad de contar con bases de datos completas y fiables es crucial para la implementación efectiva de algoritmos de aprendizaje automático. Para abordar este desafío, se han desarrollado diversos modelos y técnicas capaces de manejar conjuntos de datos incompletos, con el objetivo de imputar los datos faltantes.

Existen diferentes modelos diseñados para la imputación de datos faltantes. En los últimos años, se han estudiado extensamente diversas metodologías en este ámbito (Hasan et al., 2021b). Dos de los modelos más relevantes por los resultados obtenidos en sus imputaciones son:

2.1.1 MICE

Las imputaciones múltiples mediante ecuaciones encadenadas (MICE, por sus siglas en inglés) representan un algoritmo avanzado que emplea modelos de regresión lineal (Jafrasteh et al., 2023), y técnicas de imputación univariante, como: regresión lineal bayesiana, imputación de medias incondicionales, coincidencia predictiva de medias (Buuren & Groothuis-Oudshoorn, 2011), para imputar datos faltantes de manera efectiva. En este enfoque, las variables con valores faltantes se tratan como la variable dependiente dentro de cada modelo. Para imputar los valores faltantes en variables continuas, se ajusta un modelo de regresión lineal a los datos observados, prediciendo la media condicional para cada valor faltante y luego imputando de manera aleatoria un valor tomado de una distribución normal centrada en dicha media condicional (Shah et al., 2014).

2.1.2 GAIN

El algoritmo de Redes de imputación generativas adversarias (GAIN, Generative Adversarial Imputation Nets), es una técnica de imputación de datos faltantes basada en redes neuronales generativas adversarias (GAN, por sus siglas en inglés)(Dong et al., 2021), estos son modelos generativos que consisten en dos redes neuronales: un generador y un discriminador. En el contexto de las Redes de Imputación Generativas Adversarias (GAIN), el generador tiene como objetivo imputar con precisión los datos faltantes, mientras que el discriminador busca distinguir entre los datos observados y los imputados. El discriminador se entrena para minimizar la pérdida de clasificación, optimizando su capacidad para identificar correctamente los datos originales, mientras que el generador se entrena para maximizar la tasa de clasificación errónea del discriminador, mejorando continuamente la calidad de las imputaciones (Yoon et al.,)

2.2 Modelos de mezcla Gaussiana

2.2.1 GMM

El modelo de mezcla gaussiana (GMM) es un modelo probabilístico paramétrico que postula que los datos se originan a partir de una combinación de un número finito de distribuciones gaussianas. Estas distribuciones definen el modelo mediante una suma ponderada de componentes gaussianos, cada uno caracterizado por tres parámetros esenciales: media, covarianza y peso (Michieletto et al., 2021). Estos parámetros se estiman a partir de los datos de entrenamiento utilizando el algoritmo de maximización de expectativas (EM), un método estadístico iterativo que determina los parámetros de máxima verosimilitud del modelo probabilístico cuando las ecuaciones directas no son resolubles.

2.2.2 MGMM

A partir del modelo GMM se ha ajustado en un paquete o biblioteca del lenguaje de programación R, para presentar modelos de mezcla gaussiana conscientes de faltantes, (MGMM por sus siglas en inglés). A diferencia de las implementaciones tradicionales de GMM que manejan datos faltantes, el modelo MGMM no impone restricciones en la estructura de la matriz de covarianza (Mccaw et al., 2022). El modelo de mezcla gaussiana modificado (MGMM) utiliza un algoritmo de maximización condicional de expectativas (ECM) (Meng & Rubin, 1993), el cual mejora la eficiencia de la estimación al descomponer la maximización directa de la función objetivo del EM en una serie de maximizaciones condicionales más sencillas, cada una de las cuales puede resolverse de forma explícita.

2.3 Regresión mediante Modelos de Mezcla Gaussiana

La regresión mediante Modelos de Mezcla Gaussiana (GMM) es una técnica que emplea mezclas de distribuciones gaussianas para modelar relaciones no lineales y multivariadas entre variables predictoras y una variable objetivo. A diferencia de los métodos de regresión lineal tradicionales, los GMM ofrecen mayor flexibilidad al capturar estructuras de datos más complejas (Sepúlveda et al., 2013).

2.3.1 Fundamentos Teóricos

Para entender cómo funciona la regresión mediante Modelos de Mezclas Gaussiana (Regresión GMM), debemos analizar primero los parámetros de una GMM, ya que estos describen la distribución de cada componente, los parámetros son los siguientes.

1. **Proporciones** ($\boldsymbol{\pi_j}$): La fracción de datos correspondiente a cada componente gaussiano j de la mezcla.

2. **Medias** $(\boldsymbol{\mu_j})$: Cada componente tiene una media para X y Y. La media de salida $\boldsymbol{\mu_j^Y}$ y la media de entrada $\boldsymbol{\mu_j^X}$ son claves para las predicciones.
3. **Matrices de Covarianza** $(\boldsymbol{C_j})$: Describe la dispersión y correlación entre las variables dentro de cada componente, incluyendo las covarianzas cruzadas C_j^{XY} y C_j^{YX} que relacionan las entradas con las salidas.

Estos parámetros permiten la predicción condicional de las salidas $\boldsymbol{Y}$ dado $\boldsymbol{X = x}$.

El objetivo es predecir Y, dado $X = x$, utilizando el valor esperado condicional $E[Y|X = x]$:

$$\boldsymbol{\hat{y} = m(x) = E[Y|X = x] = \int y \cdot f_Y(y|X = x)\, dy}$$

La densidad conjunta de las variables X y $\boldsymbol{Y}$, $\boldsymbol{Z = [Y, X]}$, se modela como una mezcla de $\boldsymbol{K}$ distribuciones gaussianas:

$$\boldsymbol{f_Z(Z) = \sum_{j=1}^{K} \pi_j \cdot \mathcal{N}(Z; \mu_j, C_j)}$$

Donde $\boldsymbol{\pi_j}$ son las proporciones de cada componente gaussiano, $N(Z; \mu_j, C_j)$ es la distribución gaussiana multivariada con media $\boldsymbol{\mu_j}$ y matriz de covarianza Cj.

La matriz de covarianza $\boldsymbol{C_j}$ se descompone en:

$$C_j = \begin{bmatrix} C_j^{YY} & C_j^{YX} \\ C_j^{XY} & C_j^{XX} \end{bmatrix}$$

Donde:

- C_j^{YY} es la varianza de las variables de salida Y

- C_j^{XX} es la matriz de covarianza de las variables de entrada X
- C_j^{XY} y C_j^{YX} representan las covarianzas cruzadas entre X y Y.

Cálculo de la Predicción

La predicción condicional $E[Y|X = x]$: para el componente j se calcula como:

$$\boldsymbol{m_j(x) = \mu_j^Y + C_j^{YX} \cdot \left(C_j^{XX}\right)^{-1} \cdot \left(x - \mu_j^X\right)}$$

Esta fórmula ajusta la predicción de $\boldsymbol{Y}$ para cada componente basado en la media y covarianzas del componente.

Responsabilidades (Pesos): Los pesos $\beta_j(x)$, que reflejan la probabilidad de que el dato x pertenezca al componente j, se calculan como:

$$\boldsymbol{\beta_j(x) = \frac{\pi_j \cdot \mathcal{N}\left(x; \mu_j^X, C_j^{XX}\right)}{\sum_{k=1}^{K} \pi_j \cdot \mathcal{N}\left(x; \mu_j^X, C_j^{XX}\right)}}$$

Estos pesos determinan la contribución de cada componente en la predicción final.

La predicción de las salidas Y, dado $X = x$, es una combinación ponderada de las predicciones de cada componente:

$$\boldsymbol{\hat{y} = \sum_{j=1}^{K} \beta_j(x) \cdot m_j(x)}$$

Esta combinación toma en cuenta tanto las predicciones individuales de cada componente como sus respectivas responsabilidades, proporcionando una estimación robusta y precisa de las salidas.

2.4 Tipos de datos faltantes

En las aplicaciones de recopilación de datos o en bases de datos, pueden surgir diversos tipos de valores faltantes. La literatura identifica tres mecanismos principales de estos valores ausentes:

Los datos faltantes completamente al azar (MCAR) ocurren cuando la ausencia de información no tiene ninguna relación con los datos mismos, ya sean los que tenemos o los que no. Es como si los datos faltantes fueran elegidos al azar, sin seguir ningún patrón.

Los datos faltantes al azar (MAR) suceden cuando la información que falta en una variable está relacionada con los valores que sí tenemos de otras variables. Es decir, la probabilidad de que un dato esté ausente depende de otros datos que conocemos.

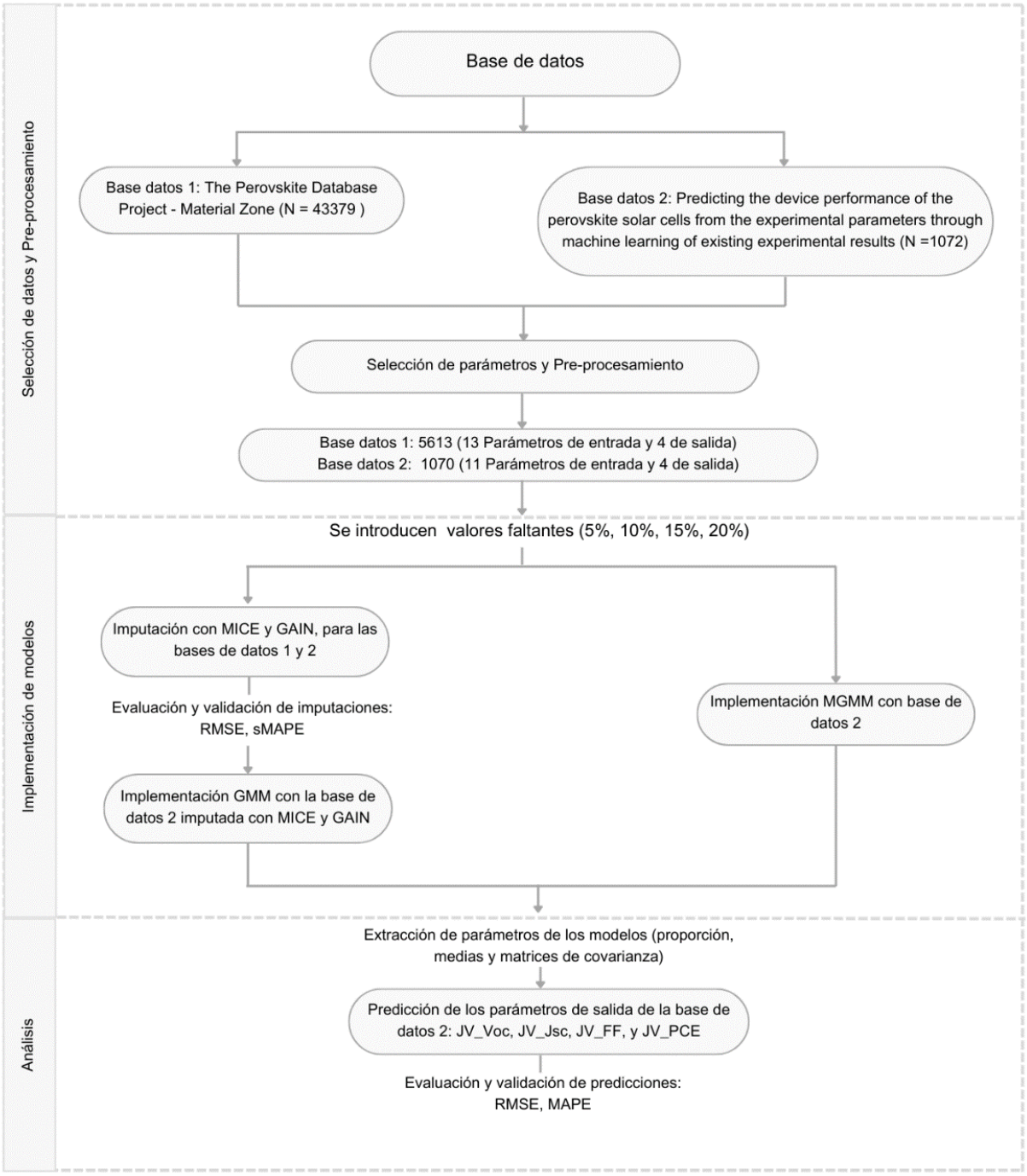
Los datos faltantes no al azar (MNAR) ocurren cuando la información que falta en una variable está directamente relacionada con el valor que debería tener esa misma variable. Es decir, la probabilidad de que un dato esté ausente depende del valor que intentamos medir (Jafrasteh et al., 2023). Un ejemplo cotidiano: Si las personas con ingresos muy altos tienen más probabilidades de no declarar su sueldo, estamos ante un caso de datos faltantes no al azar (MNAR). La falta de información sobre el ingreso está relacionada con el valor del ingreso en sí.

3. Desarrollo de la solución

La metodología propuesta se organiza en tres etapas principales: **Selección de parámetros**, que incluye la extracción de datos relevantes de la base recopilada sobre la fabricación de celdas solares de perovskita, con el fin de obtener un conjunto de datos con un mayor número de registros completos, y la identificación de las variables significativas del conjunto de datos; **Implementación de modelos**, aplicando el modelo MGMM a los datos seleccionados, que contienen valores faltantes, para realizar predicciones de valores de eficiencias de las celdas de perovskita basadas en los predictores identificados, junto con la implementación de los modelos de imputación MICE y GAIN, y el modelo GMM con los datos imputados; y **Análisis**, utilizando métricas de evaluación para comparar los resultados. Esta metodología se representa gráficamente mostrado en la figura 1.

Figura 1

Esquema metodológico



3.1 Selección de parámetros

A partir de la base de datos recopilada por Material Zone del "*Proyecto de base de datos sobre la perovskita*" proyecto de perovskita que llamaremos Base datos 1 donde ponen a disposición datos significativos, que son recopilados de artículos sobre celdas solares de perovskita (Jacobsson et al., 2021) y junto con los datos proporcionados por los autores del artículo "*Predicting the device performance of the perovskite solar cells from the experimental parameters through machine learning of existing experimental results*"(Lu et al., 2022) que llamaremos Base datos 2, donde predicen mediante aprendizaje de máquina la eficiencia de conversión de potencia (PCE por sus siglas en ingles) de las celdas solares de perovskita. Estas bases de datos incluyen parámetros del proceso de fabricación de las celdas y variables asociadas al PCE. Los parámetros fueron seleccionados por la información relevante que ofrecen sobre el rendimiento de las celdas solares de perovskita (Vélez et al., 2024). En el caso de la Base datos 1 se incluyeron las siguientes 15 parámetros, donde 11 son de entrada:

MA_ratio : Proporción de metilamonio.

Cs_ratio : Proporción de cesio.

FA_ratio: Proporción del Catión Formamidinio

Pb_ratio: Proporción de Plomo.

I_ratio: Proporción de iodo

Br_ratio: Proporción de bromo

DMF: Dimetilformamida

DMSO: Dimetilsulfóxido

th_solvent: Solvente

Cell_area_measured: Área de la celda medida

DMSO_DMF_ratio: Relación entre DMSO y DMF en la preparación

y 4 son de salida:

Voc: Voltaje de circuito abierto.

Jsc: Densidad de corriente de cortocircuito.

FF: Factor de llenado.

PCE: Eficiencia de conversión de potencia

Para la Base datos 2, se seleccionaron 15 parámetros, 11 corresponde a entradas:

ETL_EM: Energía de la capa de transporte de electrones.

Cell_A: Área de la celda solar.

AntiSolvent_Xl: Longitud de cadena del anti-disolvente.

AntiSolvent_bp: Punto de ebullición del anti-disolvente.

Annealing_t: Tiempo de recocido.

AntiSolvent_v: Volumen del anti-disolvente.

Cs_ratio, FA_ratio, MA_ratio, Br_ratio, DMSO_DMF, y 4 a variables de salida: Voc, Jsc, FF, y PCE.

Las variables restantes no fueron consideradas debido a la alta cantidad de datos faltantes y al tipo de dato, ya que solo se trabajó con datos numéricos completos. Esto permitió evaluar los modelos MICE y GAIN en escenarios con diferentes porcentajes de datos faltantes, integrando diferentes porcentaje de datos faltantes a las bases de datos completas definidas anteriormente.

3.1.1 Preprocesamiento de datos

Dado la naturaleza, distribución y método de recolección de los datos en las bases de datos, se llevó a cabo un preprocesamiento para identificar y eliminar datos anómalos, es decir, aquellos con valores incongruentes dentro de cada conjunto de datos. Este proceso se realizó mediante un código desarrollado en lenguaje de programación R (véase Anexo 1), que evalúa las características de cada variable y detecta valores que se desvían significativamente de los valores típicos o esperados. Después del filtrado, e identificar valores atipicos, es decir aquellos que parecen no pertencer a la misma distribución del resto de datos, fueron seleccionados un total de datos con sus respectivos parámetros de 5613 para la base de datos 1 y 1070 para la Base datos 2 (se eliminaron dos observaciones que tenian datos faltantes). Finalmente, se introdujeron de forma aleatoria valores faltantes en ambos conjuntos de datos para crear conjuntos de datos faltantes, utilizando porcentajes del 5%, 10%, 15%, 20%, y 25%, mediante un código diseñado en R, con un patrón de datos faltantes completamente al azar, cumpliendo con la distribución de datos MCAR.

3.2 Implementación de modelos

A partir de la revisión del estado del arte sobre algoritmos de imputación de datos, se seleccionaron los algoritmos de acuerdo con los siguientes criterios: disponibilidad de código fuente, licencias que permitieran su uso con fines de investigación, y con base en las investigaciones sobre técnicas de imputación de valores faltantes presentadas en el artículo "*Missing value imputation affects the performance of machine learning: A review and analysis of the literature (2010–2021)"* (Hasan et al., 2021a). Se eligieron los modelos de imputación MICE

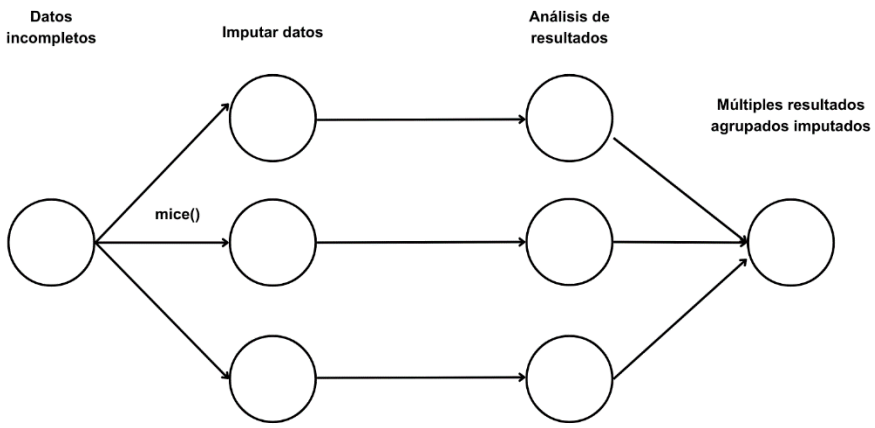
y GAIN, que han demostrado ser de los más robustos y eficientes para la imputación de datos. Para tratar conjuntos de datos con valores faltantes y su posterior aplicación en modelos de aprendizaje automático, se seleccionó el modelo MGMM. Para comparar su rendimiento, se implementó el modelo GMM con los datos imputados por MICE y GAIN.

3.2.1 Imputación con MICE

Se implementó la librería *mice versión 3.16* en el entorno RStudio para llevar a cabo la imputación múltiple de los datos(Buuren & Groothuis-Oudshoorn, 2011) . MICE opera mediante la construcción de modelos de regresión para cada variable con valores faltantes, utilizando las demás variables observadas como predictores. A partir de una imputación inicial, se itera un proceso de actualización de los modelos y generación de nuevas imputaciones hasta alcanzar la convergencia. Este enfoque produce múltiples conjuntos de datos completos, cada uno con diferentes valores imputados, lo que permite una estimación más robusta de la incertidumbre asociada a los valores faltantes (Hegde et al., 2019). En la figura 2 se ilustra los principales pasos de MICE.

Figura 2

Principales pasos utilizados en MICE



Los parámetros usados para la MICE fueron: Método de imputación: Correspondencia predictiva de medias (pmm por sus siglas en inglés). Se tomó este método a causa de usar solo valores numéricos, imputaciones: 15, iteraciones: 130.

3.2.2 Imputación con GAIN

El modelo GAIN fue implementado en Python siguiendo la arquitectura del código fuente original. A diferencia de otros métodos que se enfocan en imputar el valor más probable o la media, GAIN modela la distribución condicional completa de los datos observados, lo que permite generar múltiples imputaciones y capturar la incertidumbre asociada a los valores imputados.

GAIN consiste en dos redes neuronales: *Generator (G),* esta red toma como entrada los datos con valores faltantes y produce una versión imputada de los datos y *Discriminator (D),* esta red evalúa la calidad de la imputación generada por G y trata de distinguir entre los datos reales y los datos imputados. A continuación, se resume la arquitectura de GAIN:

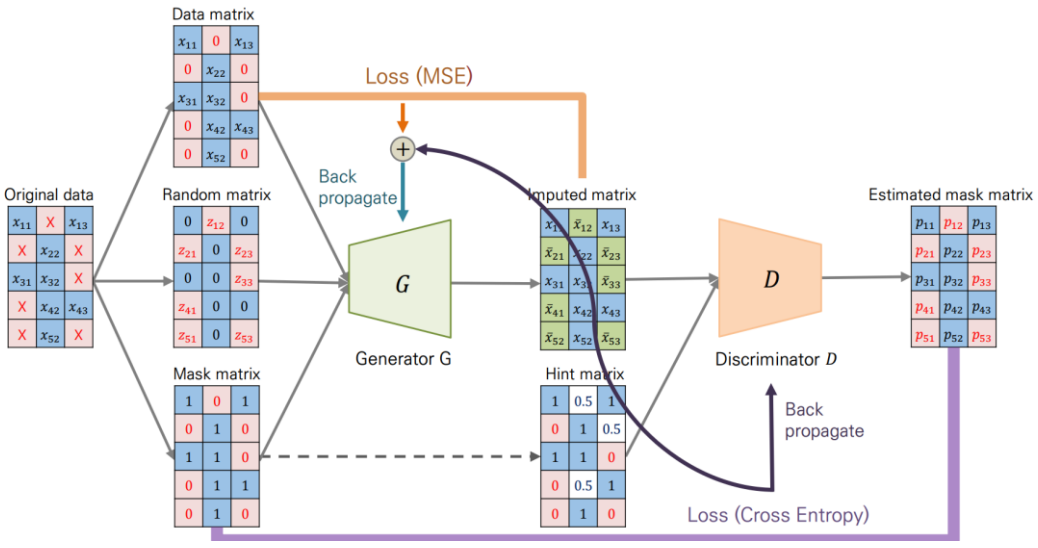
Generator (G): Toma como entrada los datos con valores faltantes (X) y la máscara de valores faltantes (M). Concatena X y M. Utiliza varias capas de perceptrón multicapa (MLP) con activaciones ReLU y Batch Normalization Produce una versión imputada de los datos (G_sample).

Discriminator (D): Toma como entrada los datos imputados (Hat_X) y un vector de pistas (H). Concatena Hat_X y H. Utiliza varias capas de perceptrón multicapa (MLP) con activaciones ReLU y Batch Normalization. Produce una probabilidad de que los datos sean reales (D_prob).

El entrenamiento de GAIN se realiza mediante un proceso de competencia entre G y D, donde G trata de generar imputaciones creíbles y D trata de distinguir entre los datos reales y los datos imputados. Esto se logra mediante la minimización de la función de pérdida de GAN, que combina la pérdida de log-verosimilitud de D con la pérdida de MSE (Media de Errores Cuadrados) entre los datos reales y los datos imputados (Yoon et al.,). La arquitectura de GAIN se ilustra en la figura 3.

Figura 3

Arquitectura GAIN



Se aplico GAIN a las bases de datos 1 y 2 con los distintos porcentajes de datos faltantes mencionados anteriormente. Para ello, después de realizar pruebas con distintos valores en los parámetros, se configuraron algunos de los parámetros del algoritmo según las recomendaciones

del autor principal (Yoon et al.,) como los siguientes: batch_size: 128, hint_rate: 0.9 alpha: 10, y el número de iteraciones se modificó en 20000.

3.2.3 Modelo MGMM

El MGMM se ajustó utilizando la función FitGMM (Mccaw et al., 2022), que forma parte de un paquete en R. Para garantizar la coherencia con investigaciones anteriores y aprovechar la calidad de los datos, el modelo se entrenó utilizando la Base de Datos 2, la cual había sido empleada por sus autores originales para realizar predicciones. Estos datos fueron normalizados antes de los entrenamientos. Además, se utilizó validación cruzada k-fold con k=5, aplicando porcentajes crecientes de datos faltantes (5%, 10%, 15%, y 20%) en el entrenamiento.

El modelo MGMM trabaja con las siguientes entradas $\boldsymbol{X}$, salidas $\boldsymbol{Y}$, y los parámetros adicionales que deben ser inicializados:

1. **Entradas** X: ETL_EM, Cell_A, Cs_ratio, FA_ratio, MA_ratio, Br_ratio, DMSO_DMF, AntiSolvent_Xl, AntiSolvent_bp, AntiSolvent_v y Annealing_t

   Salidas Y: 4 variables objetivo: JV_Voc, JV_Jsc, JV_FF, y JV_PCE

   Además de las entradas y salidas, es necesario inicializar los parámetros del modelo:

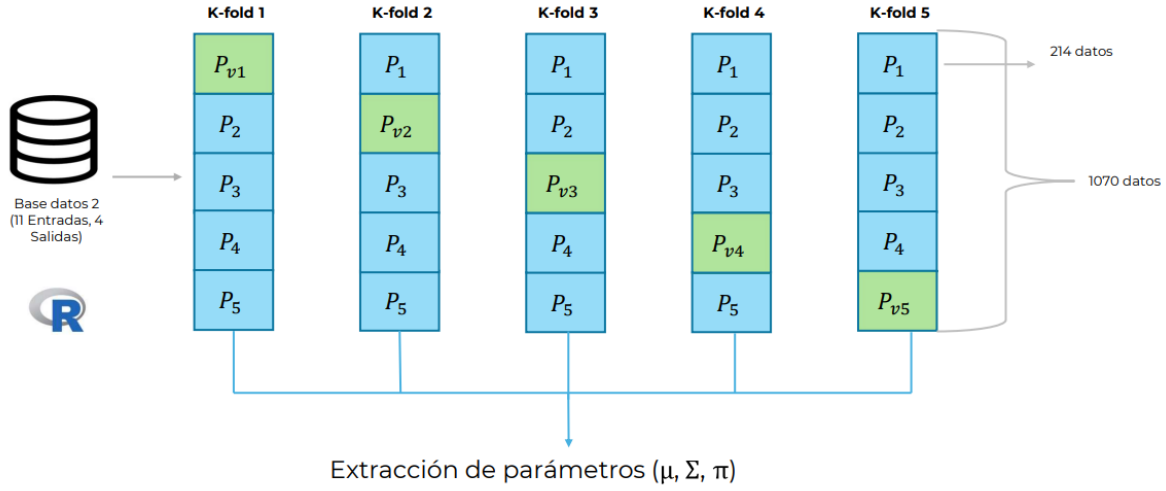
- **Proporciones ($\boldsymbol{\pi_j}$)**: Se inicializan en $\frac{\mathbf{1}}{\boldsymbol{K}}$, donde $\boldsymbol{K = 3}$. Por lo tanto, cada componente gaussiano comienza con una proporción inicial de $\frac{1}{3}$. Mediante un análisis exhaustivo de los resultados obtenidos con distintos valores de k, se pudo evidenciar que el valor seleccionado (k= 3) ofrecía el mejor ajuste a los datos.

- **Medias (μ_j)**: Las medias se inicializan como las medias de cada variable, tanto de las entradas como de las salidas.
- **Matrices de Covarianza (C_j)**: Inicialmente se configuran como matrices diagonales, lo que implica que se asume independencia entre las variables al comienzo del entrenamiento.

En la validación cruzada los 1070 datos se dividieron en 5 conjuntos. En cada fold, se entrena el modelo en 4 subconjuntos de los datos y se prueba en el subconjunto restante. Este proceso se repitió con datasets que tenían los porcentajes de faltantes mencionados anteriormente. Esto permitió evaluar cómo el modelo maneja diferentes niveles de datos incompletos en la validación cruzada, en la figura 4 se tiene un esquema de su implementación.

Figura 4

Entrenamiento de modelos



La función FitGMM en R es responsable de ajustar el modelo MGMM en presencia de datos faltantes, utilizando el algoritmo ECM (Expectación-Condicional-Maximización), una variante del algoritmo EM. La ventaja del ECM es que permite entrenar el modelo directamente con datos faltantes sin necesidad de imputar valores antes del entrenamiento.

El ajuste del modelo se realiza en dos fases principales:

E-step (Expectación): Calcula las probabilidades de pertenencia de cada dato a los diferentes componentes gaussianos.

M-step (Maximización): Actualiza los parámetros del modelo, incluyendo las proporciones, medias y matrices de covarianza, basándose en las responsabilidades calculadas en el E-step.

3.2.3.1 Algoritmo ECM y Actualización de Parámetros. El algoritmo ECM ajusta los parámetros del modelo en cada iteración, comenzando con estimaciones iniciales de las proporciones, medias y matrices de covarianza. A continuación, se detallan los pasos y las ecuaciones que describen cómo se actualizan estos parámetros en cada iteración:

Actualización de las Proporciones ($\boldsymbol{\pi_j}$):

Las proporciones de los componentes se actualizan basándose en las responsabilidades (probabilidades de pertenencia de cada dato a los componentes gaussianos). La nueva proporción para el componente j se calcula como:

$$\boldsymbol{\pi_j^{(t+1)} = \frac{1}{N} \sum_{i=1}^{N} \beta_j^{(i)}}$$

Donde $\boldsymbol{\beta}_{\boldsymbol{j}}^{(\boldsymbol{i})}$ es la responsabilidad del componente j para el dato i, t es la iteración actual, y N es el número total de datos. Esto garantiza que la fracción de datos que pertenece a cada componente gaussiano se ajuste durante cada iteración.

Actualización de las Medias ($\boldsymbol{\mu_j}$):

Las medias de cada componente j se ajustan ponderando cada dato en función de su pertenencia al componente. La nueva media se calcula como:

$$\boldsymbol{\mu}_{\boldsymbol{j}}^{(\boldsymbol{t+1})} = \frac{\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\beta}_{\boldsymbol{j}}^{(\boldsymbol{i})} \boldsymbol{x_i}}{\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\beta}_{\boldsymbol{j}}^{(\boldsymbol{i})}}$$

Aquí x_i, es el valor del dato i, y $\beta_j^{(i)}$ representa la responsabilidad del componente j. Este cálculo asegura que las medias de los componentes se actualicen para reflejar mejor los datos observados en cada iteración.

Actualización de las Matrices de Covarianza ($\boldsymbol{C_j}$)

Las matrices de covarianza se ajustan considerando la dispersión de los datos en torno a las medias actualizadas. La nueva matriz de covarianza para el componente j es:

$$\boldsymbol{C}_{\boldsymbol{j}}^{(\boldsymbol{t+1})} = \frac{\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\beta}_{\boldsymbol{j}}^{(\boldsymbol{i})} \left(\boldsymbol{x_i} - \boldsymbol{\mu}_{\boldsymbol{j}}^{(\boldsymbol{t+1})}\right)\left(\boldsymbol{x_i} - \boldsymbol{\mu}_{\boldsymbol{j}}^{(\boldsymbol{t+1})}\right)^{\boldsymbol{T}}}{\sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \boldsymbol{\beta}_{\boldsymbol{j}}^{(\boldsymbol{i})}}$$

Una vez que el modelo fue entrenado, se extrajeron los siguientes parámetros (Proporciones, Medias, Matrices de Covarianza) del modelo para cada fold y cada porcentaje de datos faltantes. Con los valores extraídos de los modelos entrenados en cada fold, es posible realizar predicciones sobre nuevos datos.

3.2.4 Modelo GMM con GAIN y MICE

Después de imputar los valores faltantes en los datos con MICE y GAIN, se usaron para la implementación del modelo de Mezcla Gaussiana (GMM). El entrenamiento del modelo GMM se realizó utilizando la Base de Datos 2 y se aplicó validación cruzada k-fold con k=5.

El modelo GMM se ajustó sobre los datos ya imputados, con los siguientes parámetros de la base de datos 2:

1. Entradas (X): 11 parámetros predictoras.
2. Salidas (Y): 4 variables objetivo.

El GMM utiliza la función FitGMM de un paquete en R. En este caso, como los datos ya han sido imputados y no hay valores faltantes, la inicialización de los parámetros se realiza utilizando el algoritmo de k-means, que divide los datos en $K = 3$ componentes gaussianos. Las proporciones, medias, y matrices de covarianza se ajustan iterativamente mediante el algoritmo EM (Expectación-Maximización).

Al igual que en el modelo MGMM, se utilizó validación cruzada k-fold con k=5, donde los 1070 datos se dividen en 5 subconjuntos. Cada fold se entrena con 4 subconjuntos y se valida en el subconjunto restante.

La función FitGMM en R permite ajustar el modelo GMM sobre los datos ya imputados por MICE o GAIN. Dado que los datos ya no contienen valores faltantes, FitGMM realiza la inicialización de los parámetros utilizando el algoritmo de k-means. A partir de esta inicialización, el ajuste se realiza mediante el algoritmo EM (Expectación-Maximización), sin necesidad de realizar imputación adicional.

El modelo GMM se entrenó con los datos imputados con MICE y GAIN para los diferentes porcentajes de datos faltantes. Este proceso se repitió para cada fold del k-fold, evaluando el rendimiento del modelo en cada conjunto de datos imputados. Después de entrenar el modelo GMM con los datos imputados, se extrajeron los parámetros, Proporciones (π_j), Medias (μ_j) y Matrices de Covarianza (C_j).

3.2.5 Predicción mediante Modelos de Mezcla Gaussiana

La regresión mediante Modelos de Mezcla Gaussiana (GMM Regression) es un enfoque probabilístico para realizar predicciones basadas en la relación entre las variables predictoras X y las variables objetivo Y. Este modelo asume que la relación entre X y Y puede modelarse como una combinación de varias distribuciones gaussianas. Al modelar la distribución conjunta de las variables, este enfoque proporciona predicciones que consideran la incertidumbre en los datos y permiten capturar relaciones no lineales. A continuación, se describen las entradas X del modelo: ETL_EM, Cell_A, Cs_ratio, FA_ratio, MA_ratio, Br_ratio, DMSO_DMF, AntiSolvent_X, AntiSolvent_bp, AntiSolvent_v, Annealing_t.

Las salidas Y del modelo corresponden a las siguientes variables de rendimiento de las celdas solares:

Eficiencia de conversión de potencia **PCE** [%]

Voltaje de circuito abierto **Voc** [mV]

Densidad de corriente de cortocircuito **Jsc** [mA/cm^2]

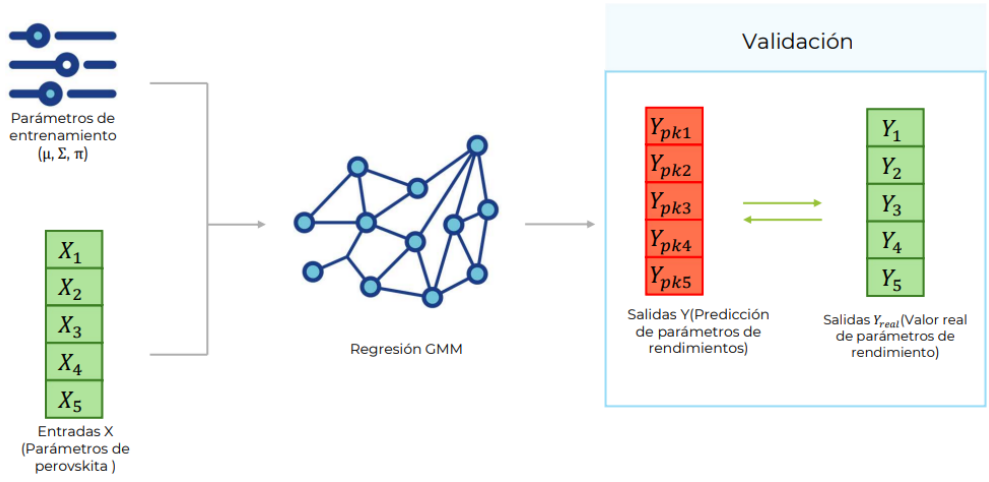
Factor de llenado **FF**

El número de componentes en la mezcla gaussiana en este modelo es $\boldsymbol{k = 3}$, lo que significa que la relación entre las variables de entrada y salida se modela como una combinación de tres distribuciones gaussianas.

El modelo ajusta los datos extrayendo los tres parámetros clave de las distribuciones gaussianas: las proporciones, las medias y las matrices de covarianza. La predicción final se hace combinando las predicciones de cada componente gaussiano, ponderadas por su responsabilidad (qué tan probable es que cada componente sea relevante para el dato de entrada). En términos sencillos, el modelo identifica patrones entre las entradas y las salidas para hacer una predicción precisa basada en la probabilidad de cada componente. En la figura 5 se tiene una ilustración de la implementación de la predicción de parámetros.

Figura 5

Predicción de parámetros de rendimiento



Para realizar las predicciones se implementó en el entorno RStudio mediante el script gmmpost.R (véase Anexo 1), que utiliza un modelo de mezcla gaussiana para realizar predicciones. El script carga las entradas X, ajusta el modelo GMM, y luego usa los parámetros de

las mezclas gaussianas para predecir las salidas Y. Además, el script maneja la predicción combinada, ponderando los resultados de cada componente gaussiano según su relevancia para los datos de entrada. Esta combinación toma en cuenta tanto las predicciones individuales de cada componente como sus respectivas responsabilidades, proporcionando una estimación robusta y precisa de las predicciones de las variables de rendimiento de las celdas solares, como JV_Voc, JV_Jsc, JV_FF, y JV_PCE, en función de las entradas relacionadas con la composición y fabricación de las celdas solares.

3.3 Métricas de evaluación

Para evaluar la efectividad de los modelos de imputación GAIN y MICE, y los modelos de mezclas gaussianas GMM con MICE, GMM con GAIN y MGMM, se emplearon tres métricas de rendimiento: el error cuadrático medio (RMSE), el porcentaje absoluto medio simétrico (sMAPE), y el error porcentual medio absoluto (MAPE), definidas en las ecuaciones 1, 2, y 3 respectivamente. Estas métricas permiten una valoración precisa de la calidad de las imputaciones y predicciones generadas respectivamente.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{1}$$

$$sMAPE = \frac{100}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{\frac{|y_i| + |\hat{y}_i|}{2}} \tag{2}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_{i-\hat{y}_i}}{y_i}\right| \times 100 \quad (3)$$

En las ecuaciones del RMSE (1), sMAPE (2) y MAPE (3), *N* representa el tamaño de los datos, y_i denota el valor real *i*, y $\hat{y}_i$ el valor imputado o predicho según corresponda.

La elección de estas métricas se basó en su relevancia para evaluar la precisión de predicciones e imputaciones, considerando que sMAPE es útil ante valores cercanos a cero y la presencia de valores atípicos. Además, se valoró su facilidad de interpretación en diversos contextos, incluyendo modelos de aprendizaje automático.

El RMSE mide la magnitud del error de las imputaciones, calculando la diferencia promedio entre los valores observados y los imputados, penalizando más los errores grandes. Un valor bajo de RMSE indica que el modelo ha logrado imputar los datos de manera precisa (Kambezidis, 2012). En la evaluación de modelos, el MAPE es una métrica de desempeño ampliamente utilizada. Al calcular el porcentaje promedio de error entre los valores predichos y los reales, el MAPE permite comparar de manera efectiva la precisión de diferentes modelos, siendo valores más bajos indicativos de un mejor ajuste (Gao et al., 2020). El RMSE y MAPE fueron usadas para evaluar las predicciones de los modelos de mezclas gaussianas mencionados anteriormente. El sMAPE, por su parte, ofrece una medida porcentual de la precisión de las imputaciones, permitiendo una comparación equitativa entre modelos, independientemente de la escala de los datos. Al ser simétrico, penaliza por igual sobreestimaciones y subestimaciones.

Además, está acotado entre 0% (imputación perfecta) y 100% (imputación completamente errónea) (Chicco et al., 2021).

Este enfoque cuantitativo y la combinación de sMAPE, MAPE y RMSE captura la magnitud real de las desviaciones entre los datos imputado, predichos y los datos reales, lo que resulta esencial para una comprensión más profunda de la precisión de las predicciones e imputaciones.

3.4 Análisis de resultados

3.4.1 Comparaciones GAIN y MICE

Se evaluó el desempeño de los modelos GAIN y MICE en las bases de datos 1 y 2, calculando las métricas RMSE y sMAPE para diferentes porcentajes de datos faltantes (5%, 10, 15, 20, 25%). Los resultados obtenidos se presentan en las tablas 1 y 2, respectivamente

Tabla 1

RMSE y sMAPE para la base de datos 1

Base de datos 1				
	RMSE		**sMAPE (%)**	
Faltantes (%)	**GAIN**	**MICE**	**GAIN**	**MICE**
5	0.1045	0.0577	14.1707	4.0201
10	0.1331	0.0734	15.9416	4.9033
15	0.1447	0.0865	18.4759	6.0768
20	0.1631	0.0965	22.9489	6.9821
25	0.1872	0.1079	33.0159	8.0650

Tabla 2

RMSE y sMAPE para la base de datos 2

Base de datos 2				
	RMSE		sMAPE (%)	
Faltantes (%)	GAIN	MICE	GAIN	MICE
5	0.2357	0.0742	28.7789	1.9225
10	0.2111	0.2023	25.6457	16.1386
15	0.2218	0.2348	29.7341	20.9630
20	0.2629	0.2303	35.9420	21.1965
25	0.2597	0.2251	33.3094	20.4393

Para analizar el desempeño de los modelos de imputación ante distintos niveles de datos faltantes y tamaños de muestra (5613 y 1070 observaciones para las bases de datos 1 y 2, respectivamente), se presentan los resultados de RMSE en la Figura 6 y de sMAPE en la Figura 7.

Figura 6

Rendimiento de RMSE: (a) Base de datos 1 N= 5613 (b) Base de datos 2 N = 1070

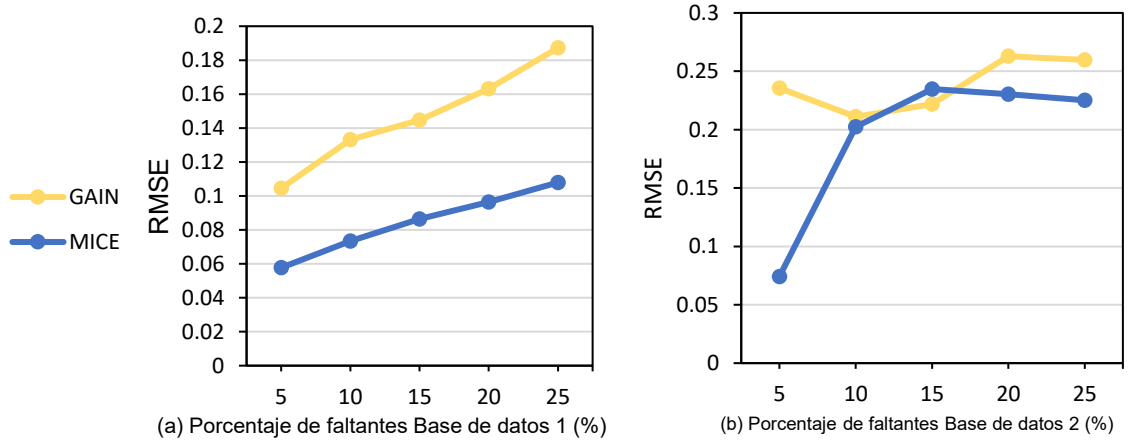
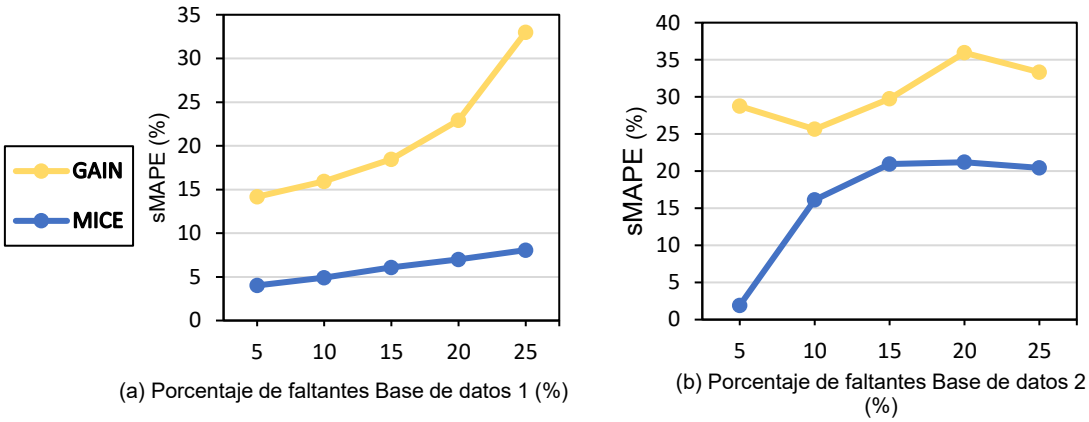


Figura 7

Rendimiento de sMAPE: (a) Base de datos 1 N= 5613 (b) Base de datos 2 N = 1070



Los resultados presentados en las Figuras 4 y 5 revelan que, aunque el RMSE y el sMAPE aumentan con la tasa de datos faltantes para ambos modelos, MICE supera a GAIN en ambas configuraciones, independientemente del tamaño de la muestra. Sin embargo, GAIN demostró ser computacionalmente más eficiente. Las Figuras 4 y 5 evidencian una mejora en los valores de RMSE y sMAPE para ambos modelos a medida que aumenta el tamaño de las muestras, a pesar de las diferencias mínimas en el número de parámetros.

3.4.2 Desempeño de modelos de mezclas gaussianas GMM+MICE, GMM+GAIN y MGMM

Se analizó el rendimiento de los modelos MGMM, GMM+GAIN y GMM+MICE sobre las bases de datos 2, utilizando las métricas RMSE y MAPE para distintos niveles de datos faltantes (5%, 10%, 15%, 20%, 25%). A continuación, se presentan los promedios de los K-Folds de las métricas usadas en las predicciones de JV_PCE, esto se muestra en la siguiente tabla.

Tabla 3

Promedio de rendimientos de los k-Folds - RMSE y MAPE para el JV_PCE

Porcentaje faltantes %	MGMM		GMM+GAIN		GMM+MICE	
	RMSE(%)	MAPE (%)	RMSE(%)	MAPE (%)	RMSE(%)	MAPE (%)
0	1.8965	9.1984	1.9999	9.0013	1.9119	9.0106
5	2.1617	9.9367	2.0962	9.6901	2.0124	9.3984
10	2.2365	10.5892	2.1674	10.0066	2.0155	9.4206
15	2.3978	10.7352	2.1465	10.0069	2.0648	9.6082
20	2.3913	11.0453	2.2736	10.6349	2.2531	10.4392

Las desviaciones estándar de los k-folds se presentarán junto con los resultados de los otros tres parámetros (JV_Voc, JV_Jsc, y JV_FF) en los apéndices 2 y 3.

Para ver el comportamiento de los modelos de mezclas gaussianas frente a diferentes porcentajes de datos faltantes en la base de datos 2, se muestran los resultados de RMSE en la Figura 8 y los de MAPE en la Figura 9.

Figura 8

RMSE promedio K-Fold de JV_PCE:

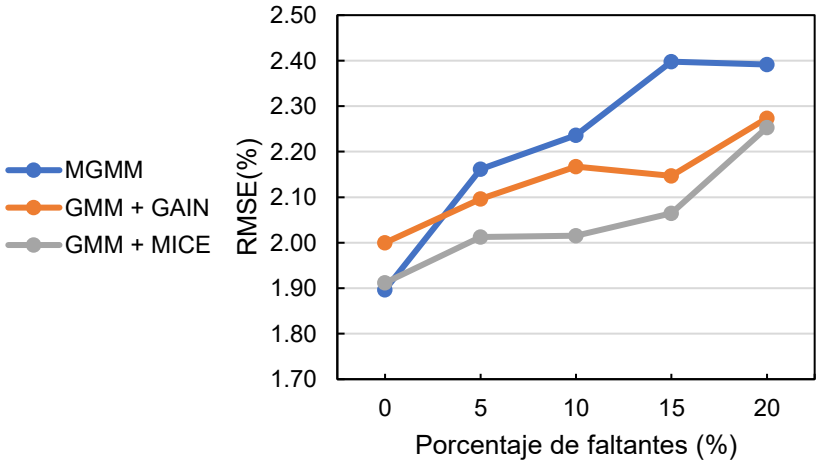
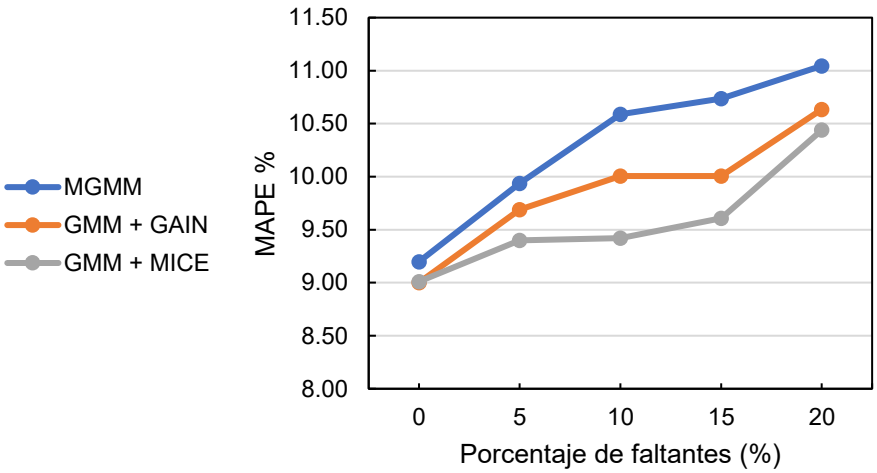


Figura 9

MAPE promedio K-Fold de JV_PCE:



El análisis conjunto de las métricas MAPE y RMSE revela un comportamiento consistente entre ambos indicadores de error. En ambos casos, el modelo GMM+MICE demuestra ser el más preciso, manteniendo los valores más bajos de MAPE y RMSE a medida que aumenta el porcentaje de datos faltantes. Esto sugiere que, aunque computacionalmente más demandante, el GMM+MICE es la mejor opción cuando se prioriza la exactitud de las predicciones, especialmente en situaciones con mayor cantidad de datos faltantes.

El modelo GMM+GAIN, si bien no alcanza la misma precisión que GMM+MICE, ofrece una ventaja en términos de eficiencia computacional, haciendo de él una alternativa viable cuando se busca un equilibrio entre rendimiento y consumo de recursos.

En cuanto al modelo MGMM, tanto en MAPE como en RMSE, su desempeño se deteriora considerablemente conforme aumenta el porcentaje de datos faltantes, lo que sugiere una pérdida de precisión y, en algunos casos, problemas de convergencia. Esta tendencia indica que el MGMM

es menos robusto frente a escenarios con altos niveles de datos faltantes, en comparación con los otros dos modelos.

Se evidencio que en el k-fold número 2 se lograron mejores métricas de RMSE y MAPE en comparación de los otros k-fold, tal y como se muestra en la tabla 4.

Tabla 4

Promedio de rendimientos del k-Folds 2 - RMSE y MAPE para el JV_PCE

JV_PCE Promedio de kfold #2						
Porcentaje de faltantes (%)	**MGMM**		**GMM + GAIN**		**GMM + MICE**	
	RMSE(%)	MAPE (%)	RMSE(%)	MAPE (%)	RMSE(%)	MAPE (%)
0	1.6028	7.0135	1.6151	7.1711	1.6228	7.2515
5	1.7674	7.0877	1.6382	7.2574	1.6500	7.5922
10	2.0127	7.1962	1.9156	7.9885	1.6753	7.6699
15	2.1168	7.2484	2.0562	9.2689	1.7947	8.1003
20	2.1358	8.2463	2.0846	9.3655	1.8145	8.1304

Al comparar los resultados del PCE del mejor k-fold con los obtenidos en la predicción realizada por los autores de la base de datos 2, quienes reportaron un **RMSE de 1.28%** para este parámetro de rendimiento, teniendo en cuenta que estos valores de RMSE representan la diferencia o cantidad de error entre los valores predichos por los modelos y los valores reales. Observamos que, para todos los porcentajes de datos faltantes, nuestro RMSE es inferior en los diferentes. No obstante, es importante señalar que nuestros modelos incluyen valores faltantes e imputados, y además no se utilizaron todos los parámetros de entrada de la base de datos 2. Esta combinación

de factores contribuye a que nuestros resultados sean inferiores a los reportados en el estado del arte.

4. Conclusiones

Los resultados del modelo MGMM fueron inferiores, especialmente al aumentar los porcentajes de datos faltantes. Esto se debe a que los datos que fueron seleccionados de celdas de perovskita no siguen una distribución normal, lo cual afecta negativamente la estabilidad y la convergencia del modelo MGMM. Con un alto porcentaje de valores faltantes, MGMM pierde precisión y no logra converger de manera efectiva, lo que lo hace inadecuado para este tipo de datos. La imputación, en cambio, ha permitido a los modelos GMM con GAIN y MICE superar estas limitaciones y mejorar el rendimiento de las predicciones.

El modelo GMM+MICE demostró ser el más efectivo para manejar datos faltantes en el contexto de las celdas solares de perovskita. Los resultados indican que la imputación de datos utilizando MICE no solo mejoró la precisión en las predicciones, sino que también permitió que el modelo GMM mantuviera su estabilidad incluso con altos porcentajes de valores faltantes. Esto sugiere que, para este tipo de datos, la imputación es una estrategia altamente conveniente, ya que los modelos GMM con imputación, como GMM+MICE y GMM+GAIN, superaron significativamente al modelo MGMM en todos los escenarios evaluados.

De acuerdo con los resultados obtenidos, el modelo MICE mostró un mejor rendimiento en la imputación de valores faltantes. No obstante, ambos modelos de imputación demostraron ser efectivos al trabajar con los parámetros distintivos de fabricación de celdas solares de perovskita, lo que sugiere que estos tipos de datos pueden ser adecuadamente tratados en presencia de datos incompletos, una situación común en diversos contextos de investigación. Es fundamental, sin embargo, un preprocesamiento adecuado, como la detección de valores atípicos. Dado que las

implementaciones de estos modelos en este tipo de datos son limitadas, es necesario continuar con investigaciones futuras para optimizar el manejo de bases de datos relacionadas con perovskitas.

La implementación exitosa de modelos de imputación en datos de celdas solares de perovskita requiere no solo un sólido conocimiento técnico, sino también una cuidadosa selección de variables. En este estudio, se seleccionó un número reducido de parámetros de celdas solares de perovskita en comparación con el total de variables disponibles en la base de datos original, lo que limitó la capacidad de los modelos GAIN y MICE para capturar las complejas relaciones entre las variables, afectando así el rendimiento de la imputación. Estos resultados subrayan la importancia de una evaluación exhaustiva de la calidad y la naturaleza de los datos faltantes antes de aplicar cualquier técnica de imputación.

5. Recomendaciones

La base de datos utilizada en este estudio, aunque en constante mejora, presenta las limitaciones comunes asociadas a la recopilación de datos a partir de múltiples fuentes. Se recomienda implementar técnicas robustas de detección y eliminación de datos atípicos para garantizar la calidad de los datos, especialmente en bases de datos construidas sin criterios de integración claros.

Para obtener una comprensión más profunda del comportamiento de los modelos de aprendizaje automático en el contexto de las celdas solares de perovskita, se sugiere incluir un mayor número de parámetros de fabricación en futuras investigaciones. Esta variabilidad en los datos permitirá evaluar la robustez y generalización de los modelos.

Dado que los modelos GMM asumen que los datos provienen de una combinación de distribuciones gaussianas, es recomendable aplicar técnicas de transformación a la base de datos para acercar la forma de los datos a este supuesto. Métodos como la transformación Box-Cox o la transformación logarítmica pueden ayudar a que las variables sigan una distribución más cercana a la normalidad. Esto podría mejorar la capacidad de los modelos GMM para capturar la estructura subyacente de los datos y generar predicciones más precisas, sobre todo en contextos con datos faltantes.

Desarrollar técnicas de selección de variables para mejorar la eficiencia del modelo: En futuros proyectos, sería recomendable implementar métodos de selección de variables que identifiquen cuáles parámetros son más relevantes para las predicciones de eficiencia en celdas solares de perovskita. Herramientas como la reducción de dimensionalidad mediante técnicas de

PCA (Análisis de Componentes Principales) o algoritmos de selección de características podrían ayudar a simplificar el modelo sin sacrificar precisión. Esto optimizaría el rendimiento computacional, reduciría el ruido en los datos y haría los modelos GMM más eficientes y manejables para futuras investigaciones.

Optimización de imputación para mejorar la estabilidad del modelo: Para futuros proyectos, se recomienda investigar y desarrollar estrategias de imputación que optimicen tanto la precisión como la estabilidad del modelo en presencia de valores faltantes. Si bien MICE ha mostrado ser efectivo, explorar enfoques más avanzados como redes neuronales profundas (ej., GAIN) o técnicas bayesianas podría mejorar aún más el rendimiento, especialmente en casos donde los datos faltantes superan el 20% del total de los datos. Estas técnicas podrían ofrecer una mayor flexibilidad y estabilidad frente a la complejidad de los datos de perovskita, manteniendo la convergencia del modelo incluso con mayores niveles de incertidumbre.

Referencias Bibliográficas

Buuren, S. v., & Groothuis-Oudshoorn, K. (2011). mice: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software, 45*, 1–67. 10.18637/jss.v045.i03

Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science, 7*, e623. 10.7717/peerj-cs.623

Dong, W., Fong, D. Y. T., Yoon, J., Wan, E. Y. F., Bedford, L. E., Tang, E. H. M., & Lam, C. L. K. (2021). Generative adversarial networks for imputing missing data for big data clinical research. *BMC Medical Research Methodology, 21*(1), 78. 10.1186/s12874-021-01272-3

Alexander Sepúlveda. *Regression by using Probability Density Functions estimated with Gaussian mixture models.* Github. https://github.com/alexander-sepulveda/GMM-Regression/tree/main

Gao, Y., Fang, Y., Dong, H., & Kong, Y. (2020). A Multifactorial Framework for Short-Term Load Forecasting System as Well as the Jinan's Case Study. *IEEE Access, 8*, 203086–203096. 10.1109/ACCESS.2020.3036675

Hasan, M. K., Alam, M. A., Roy, S., Dutta, A., Jawad, M. T., & Das, S. (2021b). Missing value imputation affects the performance of machine learning: A review and analysis of the literature (2010–2021). *Informatics in Medicine Unlocked, 27*10.1016/j.imu.2021.100799

Hegde, H., Shimpi, N., Panny, A., Glurich, I., Christie, P., & Acharya, A. (2019). MICE vs PPCA: Missing data imputation in healthcare. *Informatics in Medicine Unlocked, 17*, 100275. 10.1016/j.imu.2019.100275

Jacobsson, T. J., Hultqvist, A., García-Fernández, A., Anand, A., Al-Ashouri, A., Hagfeldt, A., Crovetto, A., Abate, A., Ricciardulli, A. G., Vijayan, A., Kulkarni, A., Anderson, A. Y., Darwich, B. P., Yang, B., Coles, B. L., Perini, C. A. R., Rehermann, C., Ramirez, D., Fairen-Jimenez, D., . . . Unger, E. (2021). An open-access database and analysis tool for perovskite solar cells based on the FAIR data principles. *Nature Energy, 7*(1)10.1038/s41560-021-00941-3

Jafrasteh, B., Hernández-Lobato, D., Lubián-López, S. P., & Benavente-Fernández, I. (2023). Gaussian processes for missing value imputation. *Knowledge-Based Systems, 273*, 110603. 10.1016/j.knosys.2023.110603

Kambezidis, H. D. (2012). 3.02 - The Solar Resource. In A. Sayigh (Ed.), *Comprehensive Renewable Energy* (pp. 27–84). Elsevier. 10.1016/B978-0-08-087872-0.00302-4

Lu, Y., Wei, D., Liu, W., Meng, J., Huo, X., Zhang, Y., Liang, Z., Qiao, B., Zhao, S., Song, D., & Xu, Z. (2022). Predicting the device performance of the perovskite solar cells from the experimental parameters through machine learning of existing experimental results. *Journal of Energy Chemistry, 77*, 200. 10.1016/j.jechem.2022.10.024

Mccaw, Z. R., Aschard, H., & Julienne, H. (2022). Fitting Gaussian mixture models on incomplete data. *BMC Bioinformatics, 23*(1)10.1186/s12859-022-04740-9

Meng, X., Rubin, D. B. (1993). Maximum Likelihood Estimation via the ECM Algorithm: A General Framework. *Biometrika, 80*(2), 267–278. 10.2307/2337198

Michieletto, S., Stival, F., & Pagello, E. (2021). Chapter 3 - A probabilistic approach to reconfigurable interactive manufacturing and coil winding for Industry 4.0. In M. Ram (Ed.), *Advances in Mathematics for Industry 4.0* (pp. 61–93). Academic Press. 10.1016/B978-0-12-818906-1.00003-6

Sepúlveda, A., Guido, R. C., & Castellanos-Dominguez, G. (2013). Estimation of relevant time–frequency features using Kendall coefficient for articulator position inference. *Speech Communication, 55*(1), 99–110. 10.1016/j.specom.2012.06.005

Shah, A. D., Bartlett, J. W., Carpenter, J., Nicholas, O., & Hemingway, H. (2014). Comparison of Random Forest and Parametric Imputation Models for Imputing Missing Data Using MICE: A CALIBER Study. *American Journal of Epidemiology, 179*(6), 764–774. 10.1093/aje/kwt312

Vélez, J., Botero L., M. A., & Sepulveda, A. (2024). Measurement of information content of Perovskite solar cell's synthesis descriptors related to performance parameters. *Emergent Materials,* 10.1007/s42247-024-00667-4

Yoon, J., Jordon, J., & Van Der Schaar, M.*GAIN: Missing Data Imputation using Generative Adversarial Nets.* ().

Apéndices

Apéndice A. Repositorio Github AEPRCES-231

En el siguiente enlace puede consultar los códigos usados para el preprocesamiento de las bases de datos, los modelos de mezclas gaussianas en R y los resultados de las métricas para cada parámetro de RMSE y sMAPE, para las imputaciones de GAIN y MICE en las dos bases de datos.

https://github.com/Yan-Velasquez/AEPRCES-231.git

Apéndice A. Tabla de desviaciones estándar del JV_PCE de cada k-fold

Porcentaje faltantes %	**MGMM**		**GMM+GAIN**		**GMM+MICE**	
σ	**RMSE**(%)	**MAPE (%)**	**RMSE**(%)	**MAPE (%)**	**RMSE**(%)	**MAPE (%)**
0	0.1915	2.6338	0.3259	1.8393	0.2228	1.9104
5	0.2479	1.9499	0.3355	2.0713	0.3074	1.8946
10	0.1746	1.9348	0.3017	2.0284	0.2732	1.7377
15	0.3070	2.1328	0.2352	1.6941	0.2331	1.6467
20	0.3667	3.9631	0.3231	1.7826	0.5100	2.6300

Apéndice B. Promedio de rendimientos de los k-Folds - RMSE y MAPE para el JV_Voc, JV_Jsc y JV_FF

- MGMM

Porcentaje faltantes %	JV_Voc		JV_Jsc		JV_FF	
	RMSE (mV)	MAPE (%)	RMSE (mA/cm^2)	MAPE (%)	RMSE	MAPE (%)
0	0.0546	3.5583	1.4364	5.1205	0.0495	5.4842
5	0.0576	3.4429	1.4380	5.5994	0.0488	5.3691
10	0.0492	3.4986	1.4778	4.9754	0.0501	5.6140
15	0.0502	3.6485	1.5236	5.0947	0.0525	5.7117
20	0.1530	4.0684	1.6570	5.4008	0.0523	5.7247

- GMM+GAIN

Porcentaje faltantes %	JV_Voc		JV_Jsc		JV_FF	
	RMSE (mV)	MAPE (%)	RMSE (mA/cm^2)	MAPE (%)	RMSE	MAPE (%)
0	0.0513	3.1698	1.3605	4.3205	0.0497	5.0870
5	0.0504	3.3681	1.3548	4.7106	0.0481	5.1619
10	0.0467	3.3925	1.3419	4.8464	0.0507	5.2013
15	0.0466	3.4317	1.3367	4.6556	0.0470	5.1092
20	0.0479	3.5240	1.4256	5.0651	0.0524	5.2867

- GMM+MICE

Porcentaje faltantes %	JV_Voc		JV_Jsc		JV_FF	
	RMSE (mV)	MAPE (%)	RMSE (mA/cm^2)	MAPE (%)	RMSE	MAPE (%)
0	0.0463	3.1314	1.2420	4.2514	0.0474	5.0573
5	0.0485	3.2933	1.2907	4.4013	0.0466	5.0841
10	0.0459	3.3548	1.2712	4.3996	0.0471	5.0733
15	0.0473	3.4220	1.2216	4.1459	0.0476	5.1283
20	0.0490	3.6942	1.3670	4.5419	0.0503	5.2462