Resolviendo un gap: Ingeniería inversa de la polimerización del Butil acrilato mediante machine learning

Desarrollado por Sebastian Ortube, con el aporte de ideas de Ignacio Ponzoni y Mónica Diaz.

INDICE

Resumen

- Se revisó Fiosina Et al., 2023 [1] con propósitos de agrandar las actuales bases de datos en la linea de investigacion, en particular se está buscando incorporar información de las curvas de distribución de pesos moleculares (MMD) para entrenar modelos de predicción.
- No resultó útil la incorporación, pero se detectó que los datos podrían ser usados para documentar un progreso utilizando un enfoque diferente de aprendizaje automático.
- Los métodos multiobjetivo utilizados para la ingenieria inversa acarrean problemáticas debido a como fueron utilizados los datos en la metodología usada en Fiosina Et al.,2023.
- Se plantea una transformación de datos y entrenamientos de nuevos modelos optimizados.

Objetivo

1. Desarrollar y evaluar un modelo de ingeniería inversa para predicción de la receta de polimerización mediante machine learning (ML), en específico se buscan predecir cuatro parámetros:

1. **cBA_0:** concentración inicial de monómero butil acrilato en (mol/L)
2. **cAIBN_0:** concentración inicial del iniciador azobisisobutironitrilo (mol/L)
3. **temp:** temperatura (°C)
4. **time:** tiempo de reacción (segundos)

2. Encontrar un mayor rendimiento en la técnica de predicción al incorporar nuevas características al sistema, viéndose representado en el valor final del coeficiente de determinación.
3. Comparar el rendimiento del modelo sin agregar los datos adicionales, obteniendo un valor de referencia para la construcción del modelo. No se espera obtener mismo rendimiento que Fiosina.

Escenario inicial

Fiosina, realizo la predicción de ingenieria inversa (PIV) de los parámetros de polimerización utilizando como datos de entrada a un modelo de Random forest (RF) entrenado con un vector de 100 valores, el cual representa la distribución de pesos moleculares (MMD).

Sin embargo, las curvas provienen de muestras simuladas mediante algoritmos basados en trayectorias estocásticas, como el Monte Carlo cinético (KMC). Al simular las muestras, se obtiene una mayor cantidad de datos además de las curvas MMD, los cuales podrían ser utilizados como información relevante para el modelo.

Incluyen datos propios del resultado de polimerización; estos son la *concentración del butil acrilato en el tiempo (**cBA_time**),* el peso molecular *numérico* y en *peso* **(Mn y Mw,**

respectivamente). Para saber más de cómo se obtuvieron los valores y como se relacionan, se aconseja leer (Fiosina et. al., 2023)

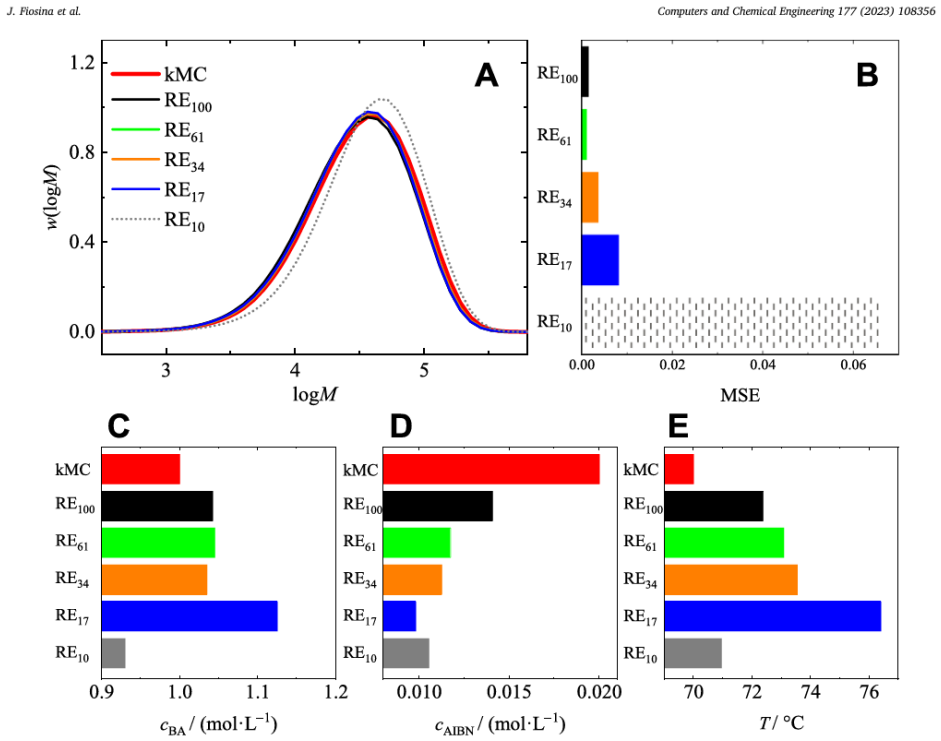

Figura 1: Predicción de la mezcla inicial de reacción y la temperatura de polimerización para obtener una MMD predefinida (en rojo): ejemplo de resultados para el experimento con T = 70 °C, cBA_0 = 1 mol·L^{-1}, cAIBN_0 = 20 mmol·L^{-1}. Tamaño del conjunto de entrenamiento: 432, utilizando conjuntos de datos completos y reducidos. Predicción de la MMD (A), recetas y temperatura predichas mediante bosque aleatorio (C, D, E) y errores cuadráticos medios (B).

Pasos de Fiosina para un ejemplo en concreto para comprender mejor:

1. Entrenó un modelo de ML con varias MMD simuladas y le dio por objetivo la receta de polimerización (**cBA_0; cAIBN_0**; **temp; time**)
2. Simuló una muestra con KMC, utilizando una receta de polimerización especifica y obtuvo una curva MMD del resultado de la polimerización (linea roja) con la respectiva receta inicial (barras rojas del grafico C, D, E).
3. Utilizó la resultante curva MMD como entrada de un modelo de ML, el cual predice la receta de polimerización (**Cba; Caibn; Temp**) que va a darme por resultado esa MMD a los 3600 segundos. En las imágenes C, D y E se muestran los resultados (receta) de la ingenieria inversa para diferentes tamaños de dataset.
4. Con los nuevos parámetros predichos se simularon curvas MMD usando KMC. La falta de similitud entre las nuevas curvas y la curva original (linea roja) tiene concordancia con la métrica de MSE obtenida en el error del modelo de random forest (figura 1,B)

En el modelo general se obtiene un R^2=0.68 (rendimiento pobre) y se estima que puede deberse al hecho de que el tiempo de reacción simulado se corta a los 3600 segundos y se puede esperar que las conversiones para las diferentes recetas sean diferentes (diferencias en Cba_remanente). En otras palabras, al solo utilizar valores de la curva MMD y no utilizar los datos de la concentración del reactivo, se pierde información rica

para el modelo. Se espera que, al incorporarlo como nueva variable de entrenamiento, el modelo pueda aprender a utilizar el dato para generar nuevos patrones y aumente el rendimiento general.

Metodología

Transformación de datos

Se plantea utilizar el modelo base planteado en la literatura donde únicamente se utilizaron las distribuciones de masas molares (MMD) como variables de entrada, pero en esta oportunidad se busca mejorar la PIV incorporando al conjunto de características las variables: concentración remanente de monómero, peso molecular promedio numérico y peso molecular promedio en peso.

a) **cBA_time:** concentración remanente del monómero (al tiempo dado).

b) **Mn:** peso molecular promedio numérico (al tiempo dado).

c) **Mw:** peso molecular promedio en peso (al tiempo dado).

d) **MMDi:** distribución de masas molares (vector de 100 dimensiones (i=1...100))

Todos los datos provienen del resultado de las simulaciones corridas del KMC, solo se plantea la utilización de una base de datos compuesta, que enriquezca el proceso PIV en un entorno de interpolación, mejorando el rendimiento global al utilizar todos los datos simultáneamente [2]

Interpretando el dominio del modelo

Entendiendo el origen de las muestras y manteniendo el foco en el posterior entrenamiento de un modelo de aprendizaje, es vital comprender la limitación en la fiabilidad del modelo. Para que un modelo de resultados confiables es ideal que las predicciones las realice interpolando datos, caso contrario, se está trabajando por fuera del "dominio de validez" y extrapola las predicciones obteniendo poca o nula confiabilidad [3].

Figura 2: Explicación gráfica, en la izquierda se encuentra un proceso de interpolación y en la derecha se aprecia un proceso de extrapolación, donde la cobertura de la predicción sale por fuera de espacio abarcado por los datos de entrenamiento (cuadrados azules). Fuente: [4]

Comprender ambas regiones es de vital importancia al hacer uso de un modelo de ML, más cuando estamos siendo usuarios y no sabemos si la arquitectura de la red fue diseñada para explorar eficazmente zonas por fuera del dominio de entrenamiento. En

este caso, al ser una arquitectura simple, se mantiene dentro del especto de interpolación. El aprendizaje que se logró con un rango de temperatura desde los 60 a 80 °C, nunca podrá ser aplicado fuera de estos límites, mismo sucede con las otras características y respectiva receta a predecir. La palabra adecuada seria que no generaliza, por ejemplo, no podría ser usado para otro tipo de monómero o iniciador, como tambien alguna mezcla no contemplada por los datos.

Por otro lado, si se buscan proceso de polimerización extensos, el modelo quedaría limitado por un sesgo temporal implícito en las simulaciones al ser cortadas a los 3600 segundos, esto acarrea la imposibilidad de aprender la cinética en etapas tardías con efectos de alta conversión.

En un principio pensé que esta limitación temporal afectaría en las predicciones para altas conversiones, por ejemplo, para conversiones mayores al 90%, pero en realidad sería una estimación por no conocer en profundidad los procesos de polimerización ideales como KMC.
En nuestro conjunto de datos se tienen 86 muestras con conversión exactas al 90%, que requirieron de un tiempo de reacción menor o igual a 3600 segundos. Cada muestra corresponde a diferentes recetas de polimerización, claro está que, si alcanzo esa reacción a valores debajo de los 3600 segundos, a los 3600 segundos se encontraría una conversión todavia mayor, asi que el filtrado de los datos fue bastante metodológico para no encontrar muestras con similitud en los datos. Como se dijo, se encontraron 86 muestras diferentes en un total de 432 simulaciones de KMC, representando un 19,907% de las muestras totales.

Asegurados de que los datos no son provenientes de una misma corrida de simulación (misma muestra), hay que analizar que no existe un sesgo en las condiciones paramétricas del proceso (por ej. sólo a 80 °C y alta cAIBN) y, con este último filtro se terminaría de comprobar que no se extrajo un grupo muy pequeño y sesgado de todas las recetas posibles.

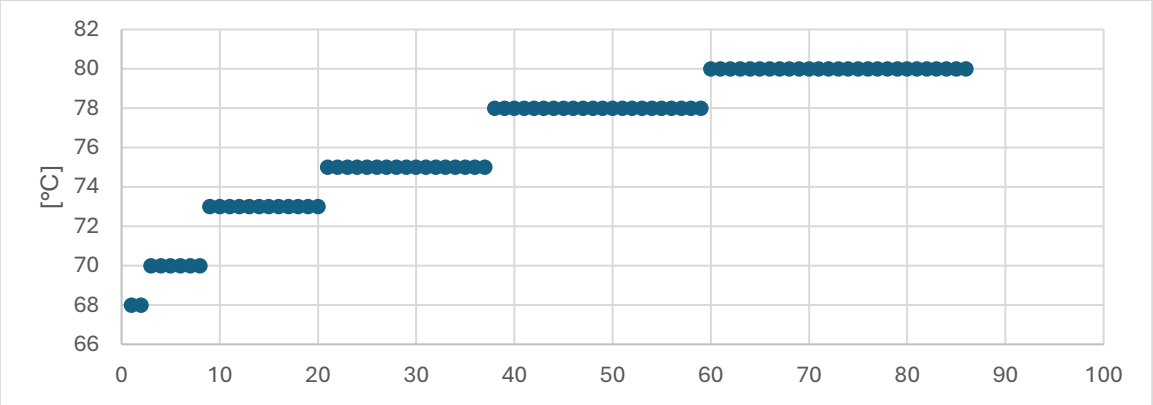

Figura 3: Rango de temperaturas cubierto para conversión mayor o igual a 90%.

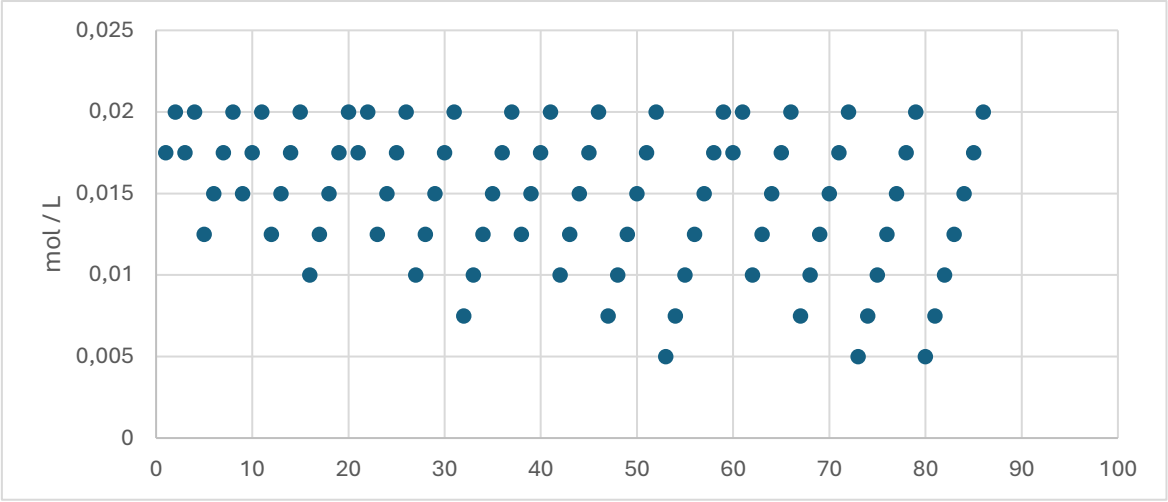

Figura 4: Rango de iniciador cubierto para conversión mayor o igual a 90%.

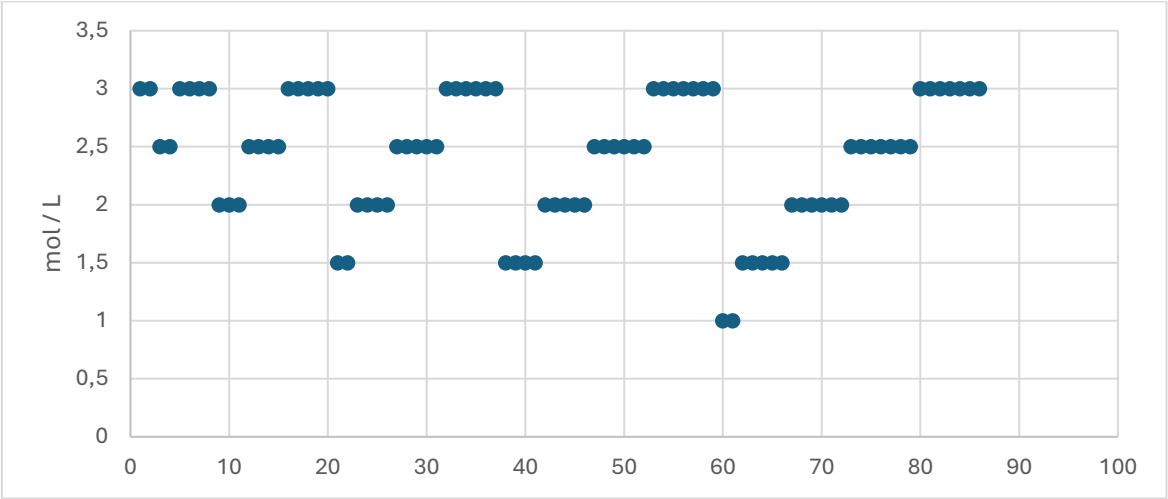

Figura 5: Rango de reactivo cubierto para conversión mayor o igual a 90%.

Se puede concluir que existe una uniformidad en el uso de condiciones iniciales del proceso de polimerización, comprendiendo que el modelo de ML va a poder predecir correctamente grados de conversiones altos y teniendo en consideración varias combinaciones como resultado. Resumiendo, el modelo sí puede generalizar razonablemente a altas conversiones, porque "vio" caminos diversos para llegar a los resultados, por ende, no hay sesgo fuerte. Se concluye que el unico limitante será el tiempo y claramente el rango de características como se explicó previamente.

¿Es viable replicar el intento de Fiosina para comparar resultados?

Hay muchos factores que desconocemos, como la simple implementación del codigo, además para diferentes capacidades de cómputo, las métricas y los resultados encontrados para el modelo pueden llegar diferir, por ende, la réplica exacta de los resultados es complicada, por no decir imposible [5]

Dada la situacion, no se pretende llegar al R^2=0.68. Se propone la implementación de un algoritmo optimizado en el que solo se tenga en cuenta las variables de las curvas MMD para la predicción de la receta y el posterior uso de las nuevas características.
Este nuevo enfoque permite analizar y ver si verdaderamente implica una mejora en la

predicción de ingenieria inversa (PIV) utilizando el mismo modelo de aprendizaje que en la literatura original (Random Forest), independizándome de las características operativas de cómputo y del algoritmo implementado.

Técnica de aprendizaje utilizada

Las primeras implementaciones que se realizaron fueron con metodologías que soportan nativamente múltiples salidas. Los métodos tradicionales de regresión múltiple (o que no tienen un soporte nativo) predicen cada variable de salida de manera independiente. Las primeras aproximaciones es conveniente modelarlas de forma correlacionada en un único vector de salida mediante soportes nativos. [6]

1. **Random forest:** Se aplicó random forest con la biblioteca de sklearn, la cual proporciona una salida en la cual los valores de la regresión estan relacionados en un único vector de salida. [7]
2. **Red neuronal:** Mediante keras y tensorflow. En muchos casos de ingenieria inversa utilizar este tipo de enfoques es prácticamente imposible, ya que estaríamos tratando de predecir variables de salida a partir de un unico dato de entrada [8, pero estamos en un proceso de ingenieria inversa que es prácticamente similar a un problema de predicción “normal” debido a la situación de multiobjetivo.

Una consideración técnica ante estos problemas de regresión de varias características y objetivos con diferentes rangos en los valores numéricos es la importancia del escalado independientemente al mismo rango, evitando sesgos y predominancia en los resultados del entrenamiento. [9]

Resultados Random Forest (RF)

Usando la base de datos con solamente las MMD

	MAE relativo	R^2
cBA_0	5.39%	0.9445
cAIBN_0	24.77%	0.1851
temp	16.10%	0.5685
time	14.39%	0.5999

Tabla 1: Métricas por variable RF_MMD

MSE Global: 113545.5647

R^2 Global: 0.5745

Usando la base de datos compuesta

	MAE relativo	R^2
cBA_0	2.25 %	0.9869
cAIBN_0	23.32%	0.2386
temp	7.52%	0.8997
time	8.71%	0.8562

Tabla 2: Métricas por variable RF_compuesta

MSE Global: 40822.8319

R^2 Global: 0.7454

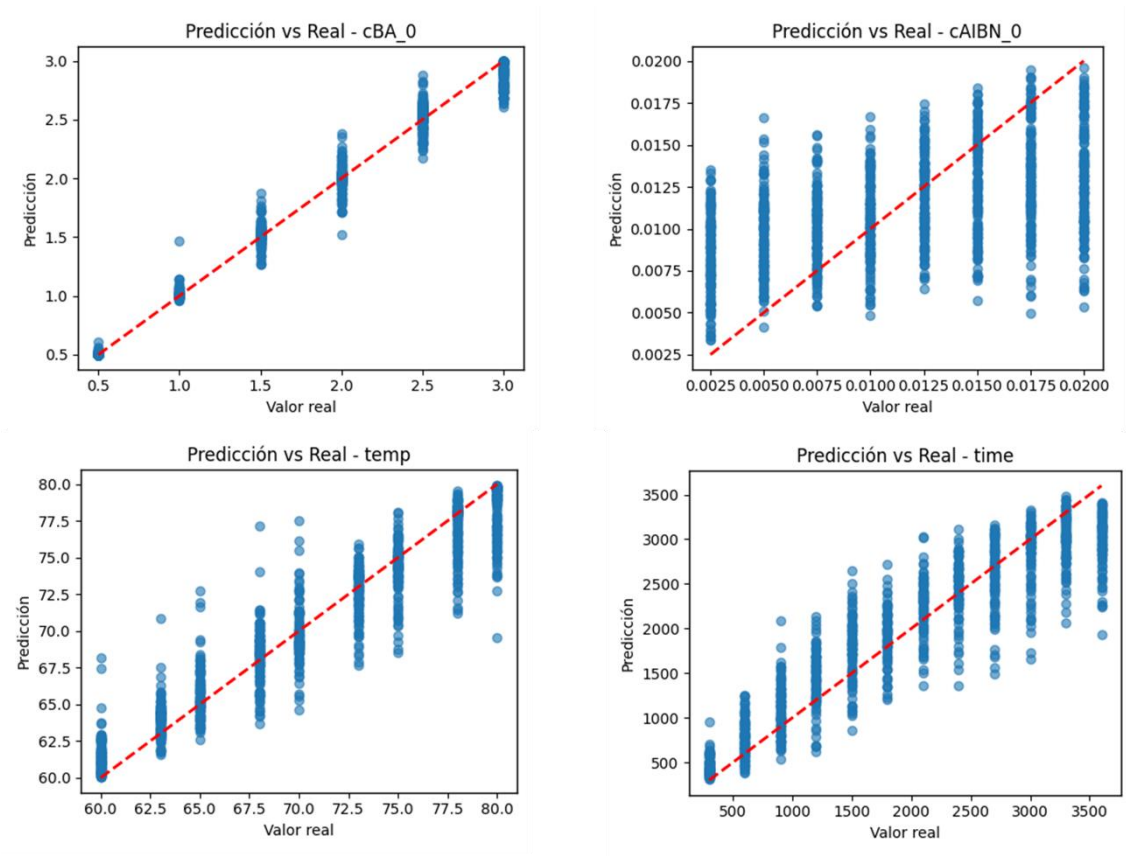

Figura 6: Valor de las predicciones usando RF contra valores reales usando la nueva base de datos. La linea punteada roja representa los valores reales y los puntos azules las predicciones.

Resultados Red Neuronal (RN)

Usando la base de datos compuesta

	MAE relativo	R^2
cBA_0	1.77%	0.9940
cAIBN_0	15.22%	0.5927
temp	3.93%	0.9754
time	6.24%	0.9193

Tabla 3: Métricas por variable RN_compuesta

MSE Global: 22916.5960

R^2 Global: 0.8703

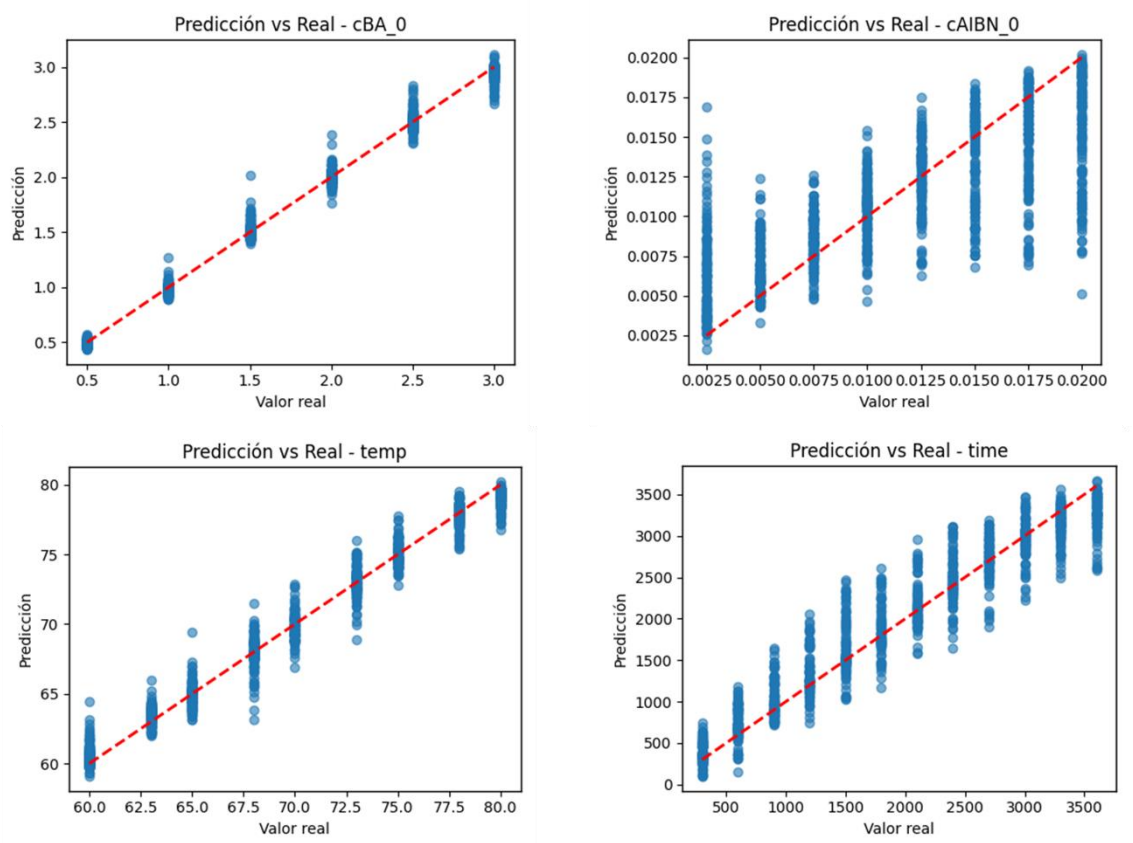

Figura 7: Valor de las predicciones usando RN contra valores reales usando la nueva base de datos. La linea punteada roja representa los valores reales y los puntos azules las predicciones.

Análisis de resultados

Para cBA_0, el MAE es bajo relativo al rango [0.5–3.0 mol/L], indicando buena precisión para ambos modelos.

Para cAIBN_0, el MAE relativo al rango [0.0025–0.02 mol/L] indica menor precisión y, se puede ver representado en el valor de determinación asociado a la variable, en el cual el modelo no pudo aprender efectivamente la relacion de las variables con el objetivo. Notar que se puede apreciar una leve mejora al utilizar una red neuronal.

La temperatura quedará a interpretación del usuario, dónde tendrá que ser consciente del margen que pueda abarcar la aplicación del modelo a un proceso de polimerización, donde las temperaturas juegan un rol crucial... sin embargo, los valores que se consiguieron son aceptables tanto para el MAE relativo y el R^2 en ambos modelos.

El tiempo es un parámetro dificil de analizar dada la particularidad del origen de datos donde los tiempos de reacción fueron frenados a los 3600 segundos. Se obtuvo un MAE alto en RF y RN, de 313 segundos (aproximadamente 5 minutos) y 224 (3 minutos con 44 segundos), respectivamente, pero es consecuencia de la magnitud de los valores, teniendo una mejor interpretación en los MAE relativos. Podría no afectar significativamente a la obtención de las propiedades deseadas, pero queda pendiente en análisis en profundidad.

Gráficamente estas conclusiones se pueden ver en la figura 6 y 7, donde al existir una gran densidad de puntos sobre la linea punteada roja, se puede interpretar como buena precisión. Por ejemplo, el mejor caso para ambos modelos, analítica y gráficamente, es la concentración inicial del reactivo (cBA_0).

Acerca de las métricas globales, el coeficiente de determinación termina siendo superior que el modelo base, en el cual solamente eran usadas las MMD como dato de entrada al RF. Otra vez, se encuentra mejor rendimiento global en la RN.

Por último, la incorporación de variables adicionales mejora significativamente la predicción global del modelo de ingeniería inversa, especialmente para parámetros críticos como cBA_0, temperatura y tiempos, cumpliendo con los objetivos propuestos.

Conclusiones

La inclusión de características en los modelos como los valores de cBA_time resulta coherente en un contexto informático e ideal. Fuera de esta situación, en condiciones experimentales reales, particularmente en laboratorios que operan con reactores químicos, es prácticamente imposible obtener mediciones exactas de compuestos remanentes o conversiones en tiempo real, debido a las limitaciones operativas para detener el proceso. Siempre y cuando estemos hablando del origen/diseño de un material desde cero.

En cambio, este tipo de datos podría ser útil en modelos de ingeniería inversa orientados al análisis de materiales ya sintetizados, aunque en el ámbito industrial este tipo de procedimientos podría entrar en conflicto con regulaciones sobre propiedad intelectual.

Finalmente, en el plano informático, la incorporación de una mayor cantidad de datos sigue siendo conveniente para mejorar el rendimiento predictivo del modelo.

Archivos complementarios

En el siguiente repositorio van a poder encontrar las bases de datos utilizadas en formato ".xlsx" y el codigo implementado para los modelos que fueron utilizados.
https://github.com/sepro22/Ingenier-a-inversa-sobre-la-polimerizacion-de-butil-acrilato

Referencias

1. **Polymer Reaction Engineering meets Explainable Machine Learning**
   J. Fiosina, P. Sievers, M. Drache, S. Beuermann, *ChemRxiv* (2023)
   doi: 10.26434/chemrxiv-2023-5vd8h
2. Kabir, H., et al., Neural network inverse modeling and applications to microwave filter design.IEEE Transactions on Microwave Theory and Techniques, 2008. 56(4): p. 867-879.]
3. Brooks D.G., Carroll S.S., Verdini W.A.Characterizing the domain of a regression modelThe American Statistician, 0003-1305, 42 (3) (1988), pp. 187-190, 10.2307/2684998
4. Lee, Junhyeong & Park, Donggeun & Lee, Hugon & Park, Kundo & Ryu, Seunghwa. (2023). Machine learning-based inverse design methods considering data characteristics and design space size in materials design and manufacturing: a review. 10.31224/2845.
5. Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., & Meger, D. (2018). Deep Reinforcement Learning That Matters. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1). https://doi.org/10.1609/aaai.v32i1.11694
6. https://scikit-learn.org/stable/modules/multiclass.html
7. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html#sklearn.ensemble.RandomForestRegressor
8. L Lee, JW., Park, W.B., Do Lee, B. *et al.* **Dirty engineering data-driven inverse prediction machine learning model.** *Sci Rep* **10**, 20443 (2020). https://doi.org/10.1038/s41598-020-77575-0
9. https://www.tensorflow.org/tutorials/keras/regression?hl=es-419