1. Decisiones de Preprocesamiento de Datos

El **preprocesamiento** transforma los datos crudos en un formato adecuado para el análisis, impactando directamente la calidad de los resultados. Las acciones realizadas se justifican a continuación:

a) Limpieza y Manejo de Valores Faltantes

La presencia de valores nulos puede sesgar los modelos y reducir su rendimiento. Para abordarlos, se seleccionaron las siguientes estrategias de imputación:

- **Mediana**: Se utilizó en variables numéricas con distribuciones asimétricas o presencia de *outliers* (valores atípicos), ya que, a diferencia de la media, es una medida de tendencia central robusta que no se ve afectada por valores extremos.
- **Valor Más Frecuente (Moda)**: Ideal para variables categóricas, ya que preserva la distribución original del dato más común, evitando la introducción de ruido o sesgos en la categoría.
- **KNN Imputer**: Este método se aplicó cuando se buscaba una imputación más precisa, aprovechando la similitud entre registros. Al inferir valores faltantes basándose en sus vecinos más cercanos, se captura la estructura subyacente de los datos.

b) Codificación de Variables Categóricas

Los algoritmos de machine learning suelen requerir datos numéricos. La codificación ordinal (Label Encoding) se usó para variables categóricas que tienen una relación jerárquica implícita (ej. "bajo", "medio", "alto"), facilitando su interpretación por modelos como los árboles de decisión. Esta técnica permite asignar un valor numérico a cada categoría, haciendo el dato procesable sin perder su orden inherente.

c) Escalado de Variables

El escalado es fundamental para algoritmos sensibles a la magnitud y variabilidad de las variables, como PCA, SVM, o redes neuronales. Se optó por las siguientes técnicas:

- **StandardScaler**: Normaliza los datos para que tengan una media de 0 y una desviación estándar de 1. Esta técnica es ideal para algoritmos que asumen que los datos siguen una distribución normal o para evitar que una variable con un rango de valores grande domine el modelo.
- **MinMaxScaler**: Escala los datos al rango [0, 1]. Esta opción es útil en casos donde se busca preservar la proporcionalidad entre los valores originales y para modelos que funcionan mejor con entradas acotadas, como las redes neuronales.

2. Elección y Ajuste de Hiperparámetros

La **optimización de hiperparámetros** es un proceso crítico que mejora el rendimiento del modelo y su capacidad de generalización. La estrategia adoptada fue la siguiente:

- **Búsqueda en Cuadrícula (Grid Search)**: Se utilizó para explorar sistemáticamente todas las combinaciones de hiperparámetros definidos en un rango. Esta técnica, combinada con la **validación cruzada**, garantiza la identificación de una configuración óptima que no sufra de sobreajuste. Aunque es computacionalmente costosa, asegura una evaluación exhaustiva de las opciones.
- **Validación Cruzada (Cross-Validation)**: Este método dividió el conjunto de datos en subconjuntos. En cada iteración, se usó un subconjunto para la validación y el resto para el entrenamiento. Esto minimizó el riesgo de sobreajuste, evaluando el desempeño del modelo en datos no vistos durante el entrenamiento y ofreciendo una estimación más fiable de su rendimiento real.

3. Comparación de Métodos y Métricas de Evaluación

Para seleccionar el modelo más adecuado, se evaluaron diversos algoritmos de reducción de dimensionalidad y de aprendizaje supervisado, comparando su rendimiento con métricas apropiadas.

a) Métricas de Evaluación

- **Clasificación**: Se usaron **precisión, recall y F1-score**. La **precisión** mide la exactitud de las predicciones positivas, el **recall** la capacidad del modelo para encontrar todos los casos positivos, y el **F1-score** es la media armónica de ambas, ofreciendo un balance entre precisión y recall.
- **Reducción de Dimensionalidad**: Para evaluar la calidad de los agrupamientos, se emplearon métricas como el **ARI** (Adjusted Rand Index) y el **NMI** (Normalized Mutual Information), que miden la similitud entre los clústeres generados y las etiquetas reales.

b) Análisis Comparativo

- **PCA vs. t-SNE/UMAP**: El **Análisis de Componentes Principales (PCA)** demostró ser rápido y eficaz para capturar la varianza global en datos lineales. Sin embargo, para datos con estructuras no lineales, **t-SNE** y **UMAP** superaron a PCA, revelando agrupamientos que no eran evidentes en la representación lineal. Aunque **t-SNE** es ideal para visualización de agrupamientos locales, no permite la transformación de nuevos datos, a diferencia de **UMAP**, que equilibra la preservación de la estructura global y local, y es más aplicable en un entorno de producción.
- **Modelos Supervisados**: Se compararon **SVM, árboles de decisión, KNN y redes neuronales**. **SVM** y **redes neuronales** mostraron un rendimiento superior en la mayoría de los casos complejos, aunque con un mayor costo computacional. **KNN** y los **árboles de decisión** resultaron más simples e interpretables, adecuados para problemas donde la interpretabilidad es una prioridad.

4. Conclusiones y Recomendaciones

Hallazgos Principales

- La calidad del **pre-procesamiento** (limpieza, imputación y escalado) es la variable más determinante para el éxito de un modelo.
- La **búsqueda sistemática de hiperparámetros** y la **validación cruzada** son cruciales para garantizar que el modelo no solo aprenda de los datos de entrenamiento, sino que pueda generalizar a datos no vistos.
- La elección del método debe basarse en el objetivo del proyecto (visualización vs. predicción) y las características de los datos (lineales vs. no lineales).

Limitaciones y Futuras Mejoras

- La búsqueda en cuadrícula puede ser ineficiente en proyectos a gran escala. Una alternativa es la **búsqueda aleatoria** o la optimización bayesiana para explorar el espacio de hiperparámetros de manera más eficiente.
- Algunos modelos, como las redes neuronales, requieren un alto poder computacional, lo cual puede ser una limitación.
- La calidad de los resultados es sensible a la correcta selección de hiperparámetros y un pre-procesamiento adecuado.

Recomendaciones Finales

1. Priorizar la **exploración y limpieza de datos** antes de cualquier modelamiento.
2. Utilizar técnicas de **reducción de dimensionalidad** para visualizar y entender la estructura de los datos.
3. Emplear **validación cruzada** de manera sistemática para evaluar el desempeño de los modelos.
4. Considerar la **interpretabilidad** y la **aplicabilidad** del modelo en un entorno de producción como criterios clave, no solo su precisión.