



# UNIVERSIDAD DE GUADALAJARA

CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E INGENIERÍA  
DEPARTAMENTO DE CIENCIAS COMPUTACIONALES

## Seminario de Solución de Problemas de Sistemas Basados en Conocimiento

### Tarea No. 2

Nombre: Hurtado González Edgar Arturo  
Código: 212597894

### Métricas y Técnicas de validación para un modelo de IA.

La Inteligencia Artificial (IA), es una parte integral de muchas industrias y va creciendo exponencialmente el uso de la misma en una mayor cantidad de ramas. Los modelos de IA son diseñados para aprender de los Datos para poder hacer predicciones y dar decisiones que se puedan tomar, todo basándose en los datos proporcionados. Conforme va avanzando es importante garantizar que los modelos sean precisos y confiables, por lo que será necesario una evaluación del modelo. Esto implica evaluar su rendimiento utilizando varias métricas, validando resultados y asegurando la calidad del modelo.

Seleccionar Métricas Adecuadas para un modelo IA es uno de los aspectos más importantes de la evaluación. Estas se utilizan para medir el rendimiento de un modelo IA y ayuda a determinar si esta cumpliendo con los objetivos previstos. Según el problema específico que se aborde y el tipo de modelo que se use, existen métricas que se pueden usar para evaluar los modelos IA. Algunas, por ejemplo, incluyen exactitud, precisión, recuperación, etc. Cada una de estas métricas proporciona una perspectiva diferente sobre el rendimiento del modelo, y es importante elegir la métrica correcta para la tarea en cuestión.

Al utilizar diferentes métricas para la evaluación del rendimiento, deberíamos estar en posición de mejorar el poder de predicción general de nuestro modelo antes de que lo pongamos en marcha para la producción sobre datos no vistos antes.

Si no se realiza una evaluación adecuada del modelo aprendizaje automático utilizando diferentes métricas, y se usa sólo la precisión, puede darse un problema cuando el modelo respectivo se despliega sobre datos no vistos y puede dar lugar a malas predicciones. Esto sucede porque, en casos como éste, nuestros modelos no aprenden, sino que memorizan; por lo tanto, no pueden generalizar bien sobre datos no vistos.

El Objetivo de las Métricas es estimar la precisión de la generalización de un modelo sobre los datos futuros (no vistos/fuera de muestra), haciendo de estas un componente integral de cualquier proyecto de ciencia de los datos.

Algunas de las métricas de validación más comunes incluyen:

## Precisión (Accuracy):

La precisión (Accuracy) es una métrica fundamental para evaluar el rendimiento de un modelo de clasificación en inteligencia artificial y aprendizaje automático. Se calcula como la proporción de predicciones correctas realizadas por el modelo sobre el total de predicciones hechas.

La fórmula para calcular la precisión es:  $Precisión = \frac{\text{Número de predicciones correctas}}{\text{Total de Predicciones}}$

En un problema de clasificación binaria, donde se tiene una clase positiva y una clase negativa, la precisión se calcula de la siguiente manera:

- Verdaderos positivos (TP): Casos en los que el modelo predice correctamente la clase positiva.
- Verdaderos negativos (TN): Casos en los que el modelo predice correctamente la clase negativa.
- Falsos positivos (FP): Casos en los que el modelo predice incorrectamente la clase positiva.
- Falsos negativos (FN): Casos en los que el modelo predice incorrectamente la clase negativa.

La fórmula de precisión se expresa como:  $Precisión = \frac{TP}{TP + FP}$

La precisión es una métrica esencial, pero puede no ser suficiente en casos donde las clases están desbalanceadas en el conjunto de datos. Por ejemplo, si una clase es mucho más prevalente que la otra, un modelo puede tener una alta precisión al predecir la clase dominante, pero podría estar fallando en la predicción de la clase minoritaria. En tales casos, es crucial considerar otras métricas como la exhaustividad (recall), el F1-score o la curva ROC junto con el área bajo la curva (AUC) para obtener una imagen más completa del rendimiento del modelo.

La precisión es una métrica fácil de interpretar, pero debe utilizarse junto con otras métricas dependiendo de la naturaleza del problema y el equilibrio de las clases para tener una evaluación más completa del rendimiento del modelo de clasificación.

## Exhaustividad (Recall)

La exhaustividad se refiere a la proporción de casos positivos (verdaderos positivos) que el modelo identifica correctamente sobre todos los casos que son realmente positivos. Se calcula con la fórmula:

$$Exhaustividad = \frac{TP}{TP + FN}$$

Donde:

- TP (Verdaderos Positivos): Los casos en los que el modelo predice correctamente la clase positiva.
- FN (Falsos Negativos): Los casos en los que el modelo predice incorrectamente la clase negativa cuando en realidad es positiva.

La exhaustividad, también llamada sensibilidad o recall, mide la capacidad del modelo para identificar correctamente todos los casos positivos. Un valor alto de exhaustividad indica que el modelo puede capturar la mayoría de los casos positivos, minimizando los falsos negativos.

Estas dos métricas a menudo están en tensión: aumentar la precisión puede reducir la exhaustividad y viceversa. Es importante considerar ambas métricas en conjunto para evaluar el rendimiento del modelo, especialmente cuando las clases no están equilibradas. Por ejemplo, en un problema médico donde identificar correctamente todos los casos positivos es crucial, se daría más importancia a la exhaustividad, aunque podría haber más falsos positivos (menor precisión). Por otro lado, en problemas donde la precisión es fundamental, como la detección de spam, se priorizaría la precisión, aunque pueda haber más falsos negativos (menor exhaustividad).

## F1-Score

El F1-Score es una métrica que combina la precisión y la exhaustividad en un solo número. Se utiliza comúnmente en problemas de clasificación, especialmente cuando hay un desequilibrio entre las clases o cuando ambas métricas (precisión y exhaustividad) son igualmente importantes. El F1-Score se calcula como la media armónica entre la precisión y la exhaustividad y se expresa de la siguiente manera:

$$F1 - Score = 2 * \frac{Precisión * Exhaustividad}{Precisión + Exhaustividad}$$

El F1-Score alcanza su mejor valor en 1 (mejor rendimiento) y su peor valor en 0. Es útil cuando se desea encontrar un equilibrio entre la precisión y la exhaustividad, ya que castiga de manera equitativa los errores de clasificación de falsos positivos y falsos negativos.

El F1-Score es especialmente útil cuando las clases están desbalanceadas, es decir, cuando una clase es mucho más frecuente que la otra. En estas situaciones, la precisión por sí sola puede ser engañosa, ya que un modelo podría predecir siempre la clase mayoritaria y aun así obtener una alta precisión. El F1-Score tiene en cuenta tanto los falsos positivos como los falsos negativos, proporcionando una medida más equilibrada del rendimiento del modelo.

Es una métrica valiosa cuando se busca un equilibrio entre la precisión y la exhaustividad, especialmente en problemas de clasificación con clases desbalanceadas, ya que proporciona una visión más completa del rendimiento del modelo que tener solo en cuenta la precisión o la exhaustividad por separado.

## Matriz de Confusión

La matriz de confusión es una herramienta que permite visualizar el rendimiento de un modelo de clasificación al mostrar el conteo de las predicciones en función de las clases reales.

En una matriz de confusión típica, las filas representan las clases reales y las columnas representan las clases predichas por el modelo. Para un problema de clasificación binaria (dos clases), la matriz de confusión tiene la siguiente estructura:

Verdaderos positivos	Falsos Positivos
Falsos Negativos	Verdaderos Negativos

Dentro de la matriz de confusión:

- Verdaderos Positivos (TP): Representan los casos en los que el modelo predice correctamente la clase positiva.
- Falsos Positivos (FP): Representan los casos en los que el modelo predice incorrectamente la clase positiva cuando en realidad es negativa.
- Falsos Negativos (FN): Representan los casos en los que el modelo predice incorrectamente la clase negativa cuando en realidad es positiva.
- Verdaderos Negativos (TN): Representan los casos en los que el modelo predice correctamente la clase negativa.

La matriz de confusión permite evaluar el rendimiento del modelo al proporcionar información detallada sobre cómo el modelo clasifica las muestras en cada clase. Además de calcular métricas individuales como la precisión, la exhaustividad y el F1-Score, la matriz de confusión es útil para tener una comprensión más completa del comportamiento del modelo en diferentes situaciones.

Visualmente, la matriz de confusión puede ser una tabla numérica, pero también se puede representar gráficamente para una mejor comprensión y análisis del rendimiento del modelo de clasificación. Es una herramienta fundamental para evaluar y ajustar modelos, especialmente en problemas de clasificación con múltiples clases o cuando se necesita comprender en detalle el comportamiento del modelo en diferentes situaciones de predicción.

## Curva ROC (Receiver Operating Characteristic) y AUC (Area Under the Curve)

La Curva ROC (Receiver Operating Characteristic) es una métrica utilizada para evaluar y visualizar el rendimiento de un modelo de clasificación binaria a varios umbrales de discriminación.

La Curva ROC representa la relación entre la tasa de verdaderos positivos (exhaustividad/sensibilidad) y la tasa de falsos positivos a medida que se varía el umbral de clasificación. Aquí están los conceptos clave asociados con la Curva ROC:

- **Tasa de Verdaderos Positivos (TPR) / Exhaustividad / Sensibilidad:** Es la proporción de positivos reales que se identifican correctamente como tales por el modelo. Se calcula como  $\frac{TP}{TP+FN}$ , donde TP es el número de verdaderos positivos y FN es el número de falsos negativos.
- **Tasa de Falsos Positivos (FPR):** Es la proporción de negativos reales que se identifican incorrectamente como positivos. Se calcula como  $\frac{FP}{FP+TN}$ , donde FP es el número de falsos positivos y TN es el número de verdaderos negativos.

La Curva ROC representa gráficamente la relación entre la TPR y la FPR a medida que se varía el umbral de clasificación. En el gráfico, el eje x muestra el FPR (1 - especificidad) y el eje y muestra el TPR (sensibilidad). Cada punto en la curva ROC representa un umbral de clasificación diferente.

Un modelo perfecto tendría una curva ROC que sube rápidamente hacia el extremo superior izquierdo del gráfico, lo que indica una alta TPR y un bajo FPR. El área bajo la Curva ROC (AUC-ROC) mide la integral de la curva ROC y proporciona una métrica resumida del rendimiento del modelo:

- AUC-ROC de 1 indica un modelo perfecto.
- AUC-ROC de 0.5 indica un modelo que clasifica aleatoriamente.

Una AUC-ROC mayor a 0.5 indica que el modelo es mejor que una predicción aleatoria. Cuanto mayor sea el AUC-ROC (más cerca de 1), mejor es el rendimiento del modelo para clasificar las clases.

La Curva ROC y el AUC-ROC son útiles porque proporcionan una forma de evaluar el rendimiento del modelo independientemente del umbral de clasificación y permiten comparar diferentes modelos entre sí. Es una herramienta valiosa para evaluar el equilibrio entre la sensibilidad y la especificidad del modelo en la toma de decisiones de clasificación binaria.

## Validación Cruzada (Cross-Validation)

La validación cruzada (cross-validation) es una técnica utilizada en el aprendizaje automático para evaluar el rendimiento de un modelo y, al mismo tiempo, maximizar el uso de los datos disponibles en el entrenamiento y prueba del modelo.

El objetivo principal de la validación cruzada es estimar cómo se comportará un modelo en un conjunto de datos independiente y no visto, al mismo tiempo que se maximiza el uso de los datos para el entrenamiento y la evaluación. Esto es especialmente útil cuando se tienen conjuntos de datos limitados.

El proceso de validación cruzada implica dividir el conjunto de datos en varios subconjuntos llamados "folds" (pliegues). Por lo general, se utiliza la validación cruzada k-fold, donde el conjunto de datos se divide en k pliegues. El proceso sigue estos pasos:

1. Se divide el conjunto de datos en k pliegues (subconjuntos) de tamaño aproximadamente igual.
2. Se entrena el modelo k veces, utilizando k-1 pliegues como datos de entrenamiento y uno de los pliegues como datos de prueba.
3. Se evalúa el rendimiento del modelo en cada iteración utilizando el pliegue de prueba y se registra una métrica de rendimiento.
4. Se calcula la métrica de rendimiento final promediando los resultados de las k iteraciones.

La validación cruzada k-fold ayuda a mitigar la variabilidad en la estimación del rendimiento del modelo, ya que utiliza múltiples particiones de datos para entrenar y evaluar el modelo. Además, utiliza todos los datos disponibles tanto para entrenamiento como para prueba, lo que puede proporcionar una estimación más precisa del rendimiento del modelo en datos no vistos.

Una variante común de la validación cruzada es la validación cruzada estratificada, que preserva la proporción de clases en cada pliegue para problemas de clasificación, asegurando que cada pliegue sea representativo de la distribución de clases del conjunto de datos original.

La validación cruzada es una técnica poderosa para evaluar modelos de aprendizaje automático y estimar su rendimiento de manera más robusta, especialmente cuando se dispone de un conjunto de datos limitado y se desea maximizar su utilización para entrenar y evaluar modelos.

## Validación Holdout

La Validación Holdout es una técnica fundamental utilizada en el aprendizaje automático para evaluar el rendimiento de un modelo. A diferencia de la validación cruzada, la Validación Holdout divide el conjunto de datos en dos conjuntos distintos: el conjunto de entrenamiento y el conjunto de prueba.

El proceso de Validación Holdout generalmente sigue estos pasos:

1. **División de datos:** Se divide el conjunto de datos en dos partes: un conjunto de entrenamiento y un conjunto de prueba. Por lo general, se asigna una gran proporción de los datos al conjunto de entrenamiento (por ejemplo, 70-80% del total) y el resto al conjunto de prueba.
2. **Entrenamiento del modelo:** Se utiliza el conjunto de entrenamiento para ajustar (entrenar) el modelo de aprendizaje automático.
3. **Evaluación del modelo:** Se evalúa el rendimiento del modelo utilizando el conjunto de prueba, que contiene datos que el modelo no ha visto durante el entrenamiento. Se utilizan métricas como la precisión, la exhaustividad, el F1-Score, la matriz de confusión, entre otras, para evaluar cómo el modelo generaliza a datos nuevos y no vistos.
4. **Ajuste del modelo (opcional):** En algunos casos, se puede ajustar el modelo utilizando el conjunto de entrenamiento y prueba para mejorar su rendimiento, aunque es importante tener cuidado con el sobreajuste al hacer esto.

La Validación Holdout es una técnica simple y rápida que proporciona una evaluación inicial del rendimiento del modelo. Sin embargo, la principal limitación de la Validación Holdout es que la evaluación del modelo puede depender en gran medida de cómo se haya realizado la partición de los datos, especialmente si el conjunto de datos es pequeño o si hay desequilibrios significativos entre las clases.

En situaciones donde se dispone de una cantidad limitada de datos, la Validación Holdout puede ser utilizada en conjunto con otras técnicas como la validación cruzada para obtener una evaluación más robusta del rendimiento del modelo.

## Validación Bootstrap

La Validación Bootstrap es una técnica de remuestreo que se utiliza para estimar la precisión de un modelo y evaluar su rendimiento.

En la Validación Bootstrap, se generan múltiples conjuntos de datos de entrenamiento y prueba mediante el muestreo con reemplazo del conjunto de datos original. Este proceso se realiza mediante los siguientes pasos:

1. **Remuestreo con reemplazo:** Se toman muestras aleatorias del conjunto de datos original con reemplazo, lo que significa que un mismo ejemplo puede aparecer múltiples veces en un conjunto de datos bootstrap y que algunos ejemplos pueden no aparecer en un conjunto de datos bootstrap en absoluto.
2. **Creación de conjuntos de entrenamiento y prueba:** Se utiliza cada conjunto de datos bootstrap generado para formar un conjunto de entrenamiento y un conjunto de prueba. El conjunto de entrenamiento se utiliza para entrenar el modelo y el conjunto de prueba se utiliza para evaluar su rendimiento.
3. **Evaluación del modelo:** Se entrena y evalúa el modelo con cada conjunto de entrenamiento y prueba bootstrap, registrando las métricas de rendimiento (por ejemplo, precisión, exhaustividad, F1-Score, etc.) en cada iteración.
4. **Promedio de resultados:** Una vez completadas las iteraciones (que pueden ser muchas), se promedian los resultados de las métricas obtenidas en cada iteración para obtener una estimación más robusta del rendimiento del modelo.

La Validación Bootstrap es útil cuando se dispone de un conjunto de datos limitado y se desea utilizar al máximo la información disponible. Al generar múltiples conjuntos de entrenamiento y prueba a partir de remuestreos con reemplazo, la validación bootstrap proporciona una estimación más estable y robusta del rendimiento del modelo, especialmente en conjuntos de datos pequeños donde la Validación Holdout o la validación cruzada pueden no ser suficientemente representativas.

Sin embargo, la Validación Bootstrap puede ser computacionalmente costosa si se utilizan muchas iteraciones o si se aplica a conjuntos de datos muy grandes. Además, al incluir muestras duplicadas en los conjuntos de datos bootstrap, se pueden presentar problemas de sobreajuste si no se manejan adecuadamente.

## Particionar la Información para El Modelo

La partición de datos es crucial para entrenar y evaluar modelos de inteligencia artificial. La elección de la técnica depende del tamaño y la naturaleza de tus datos, así como de la naturaleza del problema que estás abordando. Es importante seleccionar la técnica que mejor se adapte a tus necesidades y que proporcione una evaluación robusta del rendimiento del modelo. Aquí se presentan algunas técnicas comunes de partición de datos:

### 1. Conjunto de Entrenamiento y Conjunto de Prueba (Train-Test Split):

- **División simple:** Los datos se dividen en dos conjuntos: entrenamiento y prueba. Por ejemplo, se puede asignar el 70-80% de los datos para entrenamiento y el 20-30% restante para pruebas.

El conjunto de entrenamiento y el conjunto de prueba son dos componentes esenciales en el desarrollo y evaluación de modelos de aprendizaje automático:

- **Conjunto de Entrenamiento (Train Set):** Es la porción de datos utilizada para entrenar el modelo. Aquí es donde el algoritmo de aprendizaje automático aprende patrones, relaciones y reglas a partir de los datos proporcionados. Cuanto más variado y representativo sea el conjunto de entrenamiento, mejor será la capacidad del modelo para generalizar y hacer predicciones precisas en datos nuevos.
- **Conjunto de Prueba (Test Set):** Es una porción separada de los datos que el modelo nunca ha visto durante el entrenamiento. Se utiliza para evaluar qué tan bien generaliza el modelo a datos desconocidos. El rendimiento del modelo se mide utilizando este conjunto de prueba para verificar su capacidad para hacer predicciones precisas en nuevos datos.

El objetivo clave de dividir los datos en conjuntos de entrenamiento y prueba es asegurar que el modelo no solo memorice los datos de entrenamiento (lo que se conoce como sobreajuste o overfitting), sino que también pueda generalizar bien a datos nuevos y no vistos.

Por lo general, se asigna una proporción de los datos para el conjunto de entrenamiento y otra para el conjunto de prueba, por ejemplo, un 70-30 o 80-20 (70% para entrenamiento, 30% para prueba) o incluso se puede usar validación cruzada para hacer múltiples divisiones y evaluaciones.

Es esencial mantener estos conjuntos separados y no utilizar datos de prueba durante el entrenamiento para obtener una evaluación precisa del rendimiento del modelo en situaciones del mundo real.

## 2. Validación Cruzada (Cross-Validation):

La partición de la información en validación cruzada implica separar el conjunto de datos en múltiples partes. Las dos estrategias principales son:

- **Hold-Out Method (Método de retención):** Consiste en dividir los datos en dos conjuntos, uno para entrenamiento y otro para prueba. Por lo general, se reserva una parte importante de los datos para el entrenamiento (por ejemplo, 70-80%) y el restante para la evaluación del modelo. Sin embargo, esta estrategia puede ser sensible a cómo se realiza la división inicial y puede no ser representativa si la división no es aleatoria.
- **K-Fold Cross-Validation (Validación Cruzada con K subconjuntos):** Divide los datos en K subconjuntos (folds) aproximadamente del mismo tamaño. El modelo se entrena K veces, cada vez utilizando un subconjunto diferente como conjunto de prueba y los restantes (K-1) como conjunto de entrenamiento. Esto asegura que cada punto de datos se use tanto para entrenamiento como para prueba, reduciendo el sesgo potencial de una única división de datos.

La idea central de estas técnicas es utilizar de manera óptima la información disponible, asegurando que el modelo se entrene en datos diversos y se evalúe en conjuntos de prueba independientes, lo que proporciona una estimación más realista de su rendimiento en datos no vistos.

## 3. Leave-One-Out (LOO):

La técnica Leave-One-Out (LOO) es una forma específica de validación cruzada donde se crea una partición de datos de manera que se use un solo punto de datos como conjunto de prueba en cada iteración. El proceso de Leave-One-Out Cross-Validation (LOOCV) se lleva a cabo de la siguiente manera:

1. **Iteración individual:** En cada iteración, se elige un solo punto de datos como conjunto de prueba y el resto de los datos se utilizan como conjunto de entrenamiento.
2. **Entrenamiento del modelo:** El modelo se entrena con el conjunto de entrenamiento formado por todos los datos excepto el punto seleccionado.
3. **Evaluación del modelo:** Se evalúa el rendimiento del modelo utilizando el punto de datos excluido como conjunto de prueba. Se calcula una métrica de rendimiento (por ejemplo, precisión, error, etc.).
4. **Repetición para todos los puntos:** Este proceso se repite para cada punto de datos en el conjunto, de modo que cada punto se utiliza como conjunto de prueba exactamente una vez.
5. **Promedio de resultados:** Finalmente, se promedian los resultados de todas las iteraciones para obtener una medida general del rendimiento del modelo.

LOOCV tiene la ventaja de maximizar el uso de los datos disponibles para entrenamiento y prueba. Sin embargo, puede ser costoso computacionalmente, especialmente con conjuntos de datos grandes, ya que implica entrenar y evaluar el modelo tantas veces como puntos de datos haya en el conjunto.

Esta técnica puede ser especialmente útil cuando se tienen conjuntos de datos pequeños y se desea obtener una estimación más precisa del rendimiento del modelo, ya que cada punto de datos se utiliza para evaluar el modelo en un escenario similar al de "nuevos datos".

## Conclusión:

Las métricas de evaluación de modelos, como la precisión, la exhaustividad (recall) y el F1-Score, ofrecen distintas perspectivas sobre el rendimiento de un modelo de clasificación. La precisión mide la proporción de predicciones correctas sobre el total de predicciones, mientras que la exhaustividad se enfoca en la capacidad del modelo para identificar correctamente todos los casos positivos. El F1-Score, al combinar precisión y exhaustividad en una única métrica, resulta útil para encontrar un equilibrio entre ambas métricas, siendo especialmente valioso en escenarios con clases desbalanceadas.

La matriz de confusión, por su parte, proporciona un panorama detallado del rendimiento del modelo, mostrando los verdaderos positivos, falsos positivos, verdaderos negativos y falsos negativos. Esta herramienta es fundamental para entender cómo el modelo clasifica las muestras en cada clase y para calcular métricas como la precisión y la exhaustividad.

Además, la Curva ROC y el AUC-ROC ofrecen una visión gráfica del rendimiento de un modelo de clasificación binaria a varios umbrales de discriminación. La Curva ROC muestra la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos, mientras que el AUC-ROC representa la integral de esta curva, proporcionando una medida resumida del rendimiento del modelo.

En cuanto a las técnicas de validación, la validación cruzada, ya sea k-fold o estratificada, y la Validación Holdout son herramientas cruciales para evaluar modelos de aprendizaje automático. La validación cruzada permite maximizar el uso de los datos al entrenar y evaluar el modelo en múltiples divisiones del conjunto de datos, mientras que la Validación Holdout divide los datos en conjuntos de entrenamiento y prueba, ofreciendo una evaluación inicial del rendimiento del modelo.

La Validación Bootstrap, por otro lado, utiliza remuestreos con reemplazo para generar múltiples conjuntos de entrenamiento y prueba, proporcionando una estimación más robusta del rendimiento del modelo, especialmente en conjuntos de datos limitados.

En conjunto, estas métricas y técnicas brindan una visión completa del rendimiento de un modelo de clasificación, permitiendo evaluar su precisión, exhaustividad, equilibrio entre ambas métricas y capacidad para generalizar a datos no vistos, todo mientras se maximiza la utilización de los datos disponibles para el entrenamiento y la evaluación del modelo.

Por otro lado, La partición de datos es esencial para entrenar y evaluar modelos de inteligencia artificial de manera efectiva. Las dos técnicas comunes son el Train-Test Split y la Validación Cruzada, que incluye métodos como Hold-Out y K-Fold Cross-Validation.

El Train-Test Split divide los datos en conjuntos de entrenamiento y prueba, permitiendo al modelo aprender patrones y luego evaluar su rendimiento en datos no vistos. Esta técnica es rápida pero puede generar estimaciones sesgadas debido a una única división de datos.

La Validación Cruzada, especialmente el método K-Fold, aborda la limitación de la tendencia al dividir los datos en múltiples subconjuntos para entrenamiento y prueba, mejorando la robustez de la evaluación del modelo. Sin embargo, es más costoso computacionalmente.

El Leave-One-Out Cross-Validation (LOOCV) maximiza la información al usar cada punto de datos como conjunto de prueba una vez, siendo útil para conjuntos de datos pequeños. Sin embargo, su alto costo computacional puede ser una limitación en conjuntos de datos grandes.

Elegir la técnica de partición adecuada depende del tamaño del conjunto de datos, la necesidad de estimaciones precisas y la capacidad computacional disponible. La selección de la técnica correcta es crucial para entrenar modelos precisos y generalizables en datos no vistos.