



***Instituto Politécnico Nacional***

***Escuela Superior de Cómputo***

***Profesor:***

***Alumnos: Hernández Jiménez Erick Yael***

***Patiño Flores Samuel***

***Robert Garayzar Arturo***

***5BV1***

***Practica 4: Métodos de validación***

# Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Desarrollo</b>	<b>2</b>
<b>3. Conclusiones</b>	<b>4</b>
3.1. Hernández Jiménez Erick Yael . . . . .	4
3.2. Patiño Flores Samuel . . . . .	4
3.3. Robert Garayzar Arturo . . . . .	4

# 1. Introducción

La validación en machine learning es un proceso importante para evaluar el rendimiento de los modelos y asegurar que generalicen bien a datos nuevos y no vistos. El objetivo principal de la validación es evitar que el modelo se ajuste demasiado a los datos de entrenamiento, lo que llevaría a un fenómeno conocido como overfitting. Al dividir los datos en conjuntos de entrenamiento y prueba, la validación permite obtener una estimación más realista de cómo se comportará el modelo cuando se aplique a datos del mundo real.

## Métodos de validación

- **Hold-Out (Separación simple):** Este es el método más básico de validación. Consiste en dividir los datos en dos conjuntos:
  - Un conjunto de entrenamiento (generalmente el 70-80 % ) donde el modelo se ajusta.
  - Un conjunto de prueba (el 20-30 % restante) donde se evalúa su rendimiento.
- **K-Fold Cross-Validation:** Este método mejora la estabilidad de la validación. El conjunto de datos se divide en K subconjuntos (o pliegues). El modelo se entrena K veces, usando cada vez un pliegue diferente como conjunto de prueba y los K-1 pliegues restantes como conjunto de entrenamiento. El rendimiento final se obtiene promediando los resultados de las K pruebas.
- **Leave-One-Out Cross-Validation (LOO):** Este es un caso especial de K-Fold donde K es igual al número de observaciones en el conjunto de datos. Esto significa que en cada iteración, un solo dato se utiliza para la prueba y el resto para el entrenamiento. Se repite tantas veces como observaciones haya en los datos, y el rendimiento final es el promedio de todas las evaluaciones.

**¿Por qué es importante la validación?** La validación es esencial para evitar sobreajuste o subajuste del modelo. Un modelo que sobreajusta tiende a tener un rendimiento excelente en el conjunto de entrenamiento, pero falla cuando se enfrenta a datos nuevos. En cambio, un modelo que subajusta no captura bien los patrones subyacentes en los datos y tiene un mal rendimiento tanto en el entrenamiento como en el test. Los métodos de validación ayudan a equilibrar este proceso, brindando una estimación confiable de cómo se comportará el modelo en datos no vistos.

# 2. Desarrollo

Para esta práctica se utilizaron dos conjuntos de datos: el de Iris, que está balanceado, y el de Wine, que presenta un desbalance. Esto se hizo para contrastar el uso de las validaciones de los métodos de validación.

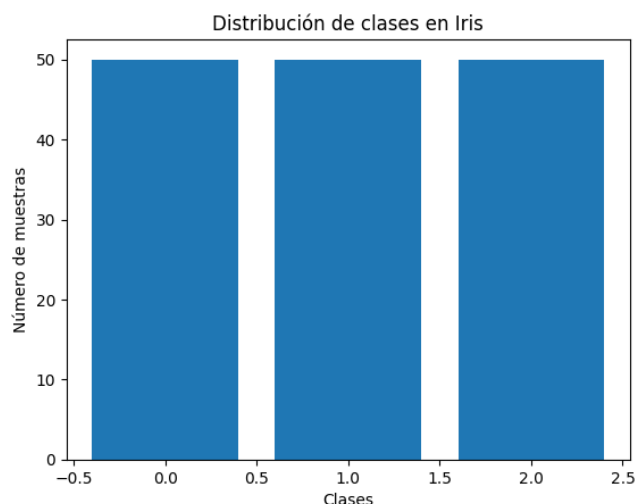


Figura 1: Datos de Iris

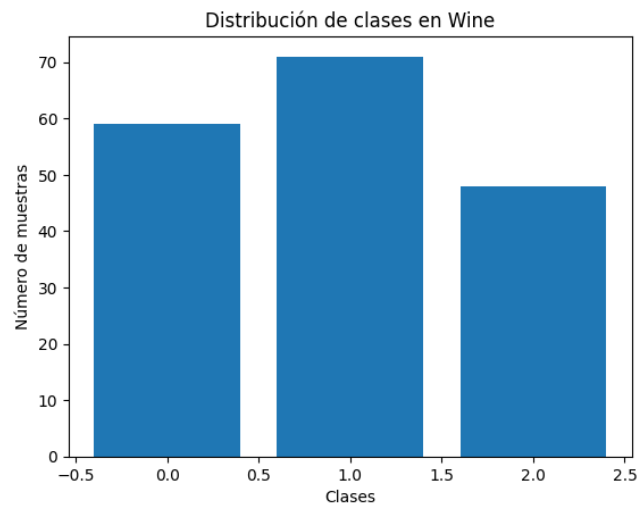


Figura 2: Datos de Wine

Además, se agregó la opción para que el usuario pueda establecer el porcentaje que se utilizará en Hold-Out y el número de pliegues a realizar en K-Fold. Por lo tanto, los resultados mostrados son con un 0.8 para Hold-Out y 10 para K-Fold.

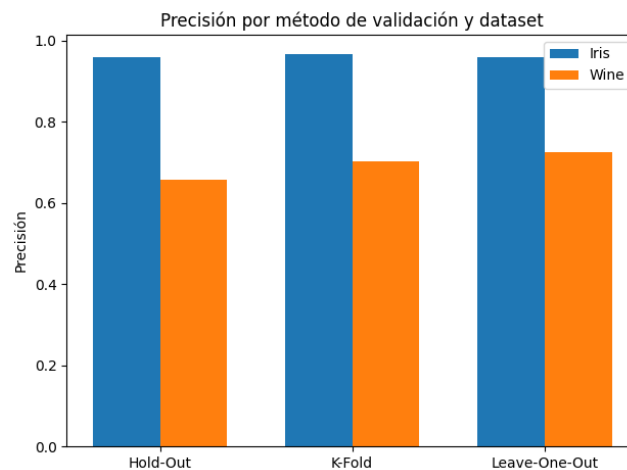


Figura 3: Precisión de las validaciones utilizadas

En la gráfica podemos observar que, **Hold-Out** mostró una alta precisión en el conjunto de datos de Iris, alcanzando un valor cercano a 1.0, lo que indica que el modelo es capaz de generalizar bien en este contexto. Sin embargo, en el conjunto de datos de Wine, la precisión fue notablemente menor, con un valor de aproximadamente 0.6, lo que sugiere que el modelo podría tener dificultades para clasificar correctamente en un conjunto desbalanceado. Para **K-Fold** se observa que la precisión para ambos conjuntos se mantuvo más equilibrada. El conjunto de datos de Iris continuó mostrando un alto rendimiento, mientras que el conjunto de datos de Wine tuvo una precisión intermedia de aproximadamente 0.7, lo que indica que el método es útil para mejorar la estabilidad de las predicciones en conjuntos desbalanceados. Por último, **Leave-One-Out** presentó resultados similares al de K-Fold, destacando nuevamente el rendimiento superior del conjunto de Iris, con una precisión que se mantiene alta, y un rendimiento moderado en Wine. Este resultado refuerza la idea de que la elección del método de validación puede influir significativamente en el desempeño del modelo, especialmente en datasets desbalanceados.

### **3. Conclusiones**

#### **3.1. Hernández Jiménez Erick Yael**

Al implementar los métodos de validación Hold-Out, K-Fold y Leave-One-Out, notamos diferencias significativas en su impacto sobre la precisión de los modelos entrenados. El método Hold-Out, al depender de un único valor de división, es más susceptible a la varianza en función de cómo se dividen los datos, especialmente cuando las clases son de tamaños desiguales. K-Fold, por su parte, ofrece una evaluación más equilibrada, ya que la variabilidad se reduce al promediar los resultados de múltiples divisiones. Finalmente, Leave-One-Out, aunque es el más exhaustivo, resulta computacionalmente costoso lo que lo hace menos práctico, pero obtiene resultados similares a K-Fold.

#### **3.2. Patiño Flores Samuel**

En resumen, las validaciones realizadas muestran que la elección del método de validación es importante para evaluar el rendimiento de los modelos de clasificación, especialmente en conjuntos de datos con características diferentes. La combinación de técnicas de validación, como K-Fold y Hold-Out, permite obtener una visión más clara sobre la capacidad de generalización del modelo, lo que es fundamental para su implementación en problemas del mundo real.

#### **3.3. Robert Garayzar Arturo**

Los tres métodos de validación implementados nos permiten observar cómo se comportan los modelos en diferentes condiciones de entrenamiento y prueba. Si bien Hold-Out es rápido y sencillo de aplicar, su simplicidad puede ser una desventaja en datasets con distribuciones desbalanceadas. Por otro lado, K-Fold resulta una excelente opción por su capacidad de mantener un buen equilibrio entre precisión y tiempo de ejecución. Finalmente, Leave-One-Out, aunque garantiza el máximo aprovechamiento de los datos para el entrenamiento, no ofrece una mejora significativa sobre K-Fold en este experimento, dado su alto costo computacional y poca diferencia en los resultados.