The background of the slide features a complex network diagram with numerous nodes and connecting lines, rendered in a light blue color against a dark blue background. The nodes are small squares, and the lines are thin, creating a web-like structure that fills the entire frame.

Aprendizaje de **Máquina Supervisada**

Sesión 2

Nivel de Ajuste de un Modelo

El nivel de ajuste de un modelo se refiere a su capacidad para representar fielmente los datos observados. Un buen ajuste implica capturar patrones sin caer en el sobreajuste ni en el subajuste. Esto es crucial para lograr predicciones precisas y generalizables.



Tipos de Error de un Modelo

Error de Entrenamiento

Mide la diferencia entre las predicciones del modelo y los valores reales utilizando el conjunto de datos de entrenamiento. Este error es generalmente bajo porque el modelo ha sido ajustado específicamente para este conjunto de datos, pero no garantiza un buen rendimiento en nuevos datos.

Error de Validación

Evalúa el rendimiento del modelo en un conjunto de datos de validación que no se usó durante el entrenamiento. Proporciona una estimación de cómo se comportará el modelo en datos no vistos y ayuda a identificar problemas de sobreajuste.

Error de Prueba

Se mide utilizando un conjunto de datos de prueba independiente, que se mantiene reservado hasta el final del proceso de modelado. Este error proporciona una evaluación final del rendimiento del modelo y asegura que no hubo optimización excesiva.

Sobreajuste, Subajuste y Ajuste Apropiado

Sobreajuste (Overfitting)

Ocurre cuando un modelo se ajusta demasiado bien a los datos de entrenamiento, capturando tanto patrones reales como ruido y anomalías. Esto resulta en un rendimiento pobre en datos nuevos. Un modelo sobreajustado puede tener un error de entrenamiento muy bajo, pero un error de validación o prueba alto.

Subajuste (Underfitting)

Ocurre cuando un modelo es demasiado simple para capturar los patrones subyacentes en los datos. Esto se traduce en un rendimiento pobre tanto en los datos de entrenamiento como en los de validación. Un modelo subajustado tiene tanto un error de entrenamiento alto como un error de validación alto.

Ajuste Apropiado

Se logra cuando el modelo equilibra correctamente la complejidad y la capacidad de generalización. Captura bien los patrones relevantes sin sobreajustarse ni subajustarse, proporcionando un rendimiento sólido tanto en los datos de entrenamiento como en los de validación.

Trade-Off entre Sesgo y Varianza



Equilibrio Crucial

El trade-off entre sesgo y varianza implica encontrar un equilibrio adecuado entre la simplicidad del modelo (bajo sesgo) y su capacidad para generalizar (baja varianza). El objetivo es diseñar un modelo con un buen rendimiento general, minimizando tanto el sesgo como la varianza.



Sesgo (Bias)

Se refiere al error debido a las suposiciones simplificadas que hace el modelo. Un modelo con alto sesgo es demasiado simple y no captura los patrones subyacentes en los datos, resultando en subajuste. Es como usar una línea recta para ajustar datos que claramente forman una curva.



Varianza

Se refiere a la sensibilidad del modelo a las pequeñas fluctuaciones en los datos de entrenamiento. Un modelo con alta varianza se ajusta demasiado a los datos de entrenamiento y tiene un rendimiento pobre en datos nuevos, resultando en sobreajuste.

¿Qué es la Validación Cruzada?

1 Definición

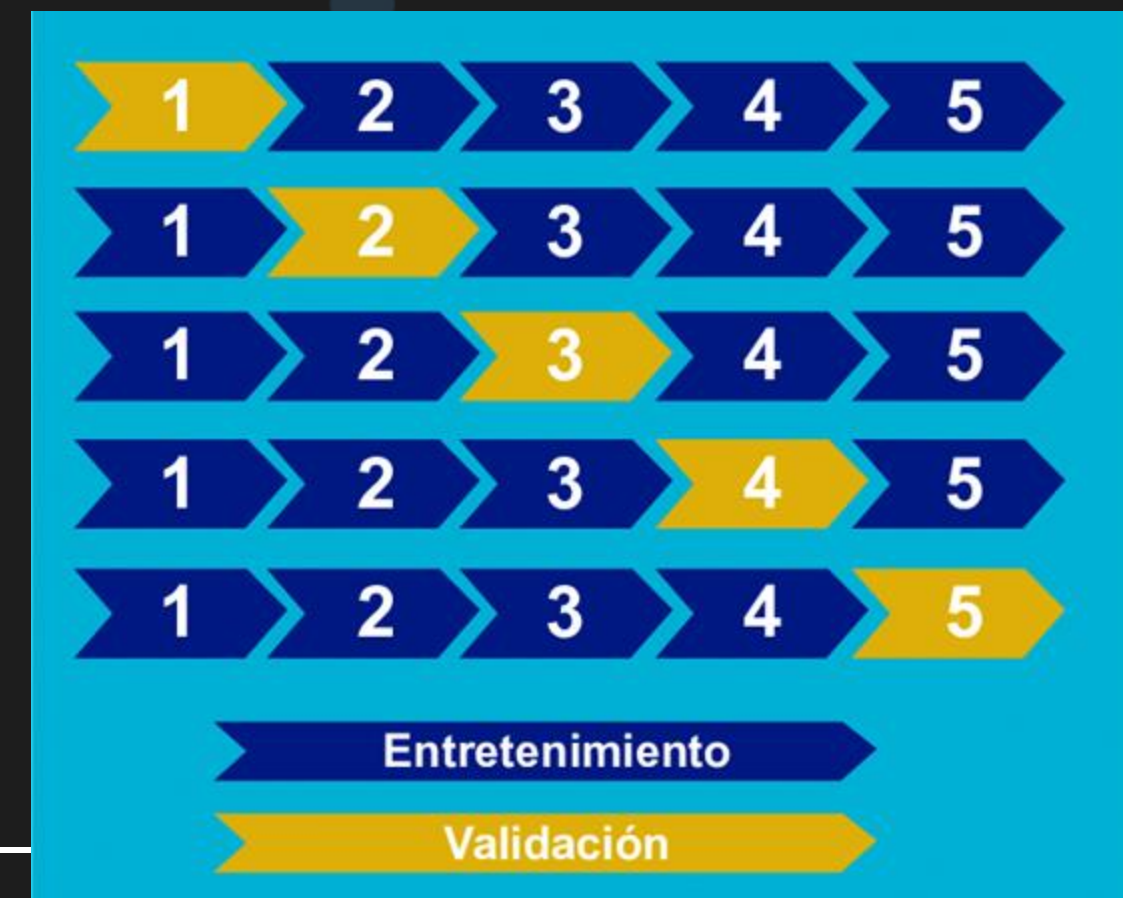
La validación cruzada es una técnica utilizada para evaluar el rendimiento de un modelo de aprendizaje automático. Consiste en dividir los datos disponibles en varios subconjuntos y entrenar el modelo en algunos mientras se valida en otros. Esto se hace de manera iterativa, asegurando que cada subconjunto se use al menos una vez como conjunto de validación.

3 Beneficios

Permite obtener una estimación más robusta y confiable del rendimiento del modelo. Ayuda a identificar si el modelo generaliza bien o se ajusta demasiado a los datos. Maximiza el uso de los datos disponibles, especialmente útil cuando el conjunto de datos es limitado.

2 Objetivo

El propósito principal es medir cómo de bien puede generalizar un modelo a datos no vistos, previniendo el sobreajuste. En lugar de utilizar un único conjunto de entrenamiento y otro de prueba, la validación cruzada repite el proceso de entrenamiento y evaluación varias veces, utilizando diferentes particiones de los datos.



Técnicas de Validación Cruzada

Método de Retención (Hold-Out)

Consiste en dividir el conjunto de datos en dos partes: un conjunto de entrenamiento y un conjunto de prueba. Es rápido y fácil de implementar, pero su estimación del rendimiento puede variar según cómo se dividan los datos. Ideal para conjuntos de datos grandes o cuando se necesita una evaluación rápida.

Random Subsampling

Implica dividir repetidamente los datos en conjuntos de entrenamiento y validación de forma aleatoria y evaluar el modelo en varias iteraciones. Permite una evaluación diversa y reduce el riesgo de divisiones desfavorables, aunque puede ser menos representativa de la distribución completa de los datos.

1

2

3

4

Validación Cruzada k-Fold

El conjunto de datos se divide en k subconjuntos de tamaño aproximadamente igual. El modelo se entrena k veces, utilizando en cada iteración $k-1$ folds para entrenamiento y el fold restante para evaluación. Proporciona un buen equilibrio entre sesgo y varianza en la estimación del rendimiento.

Leave-One-Out (LOO)

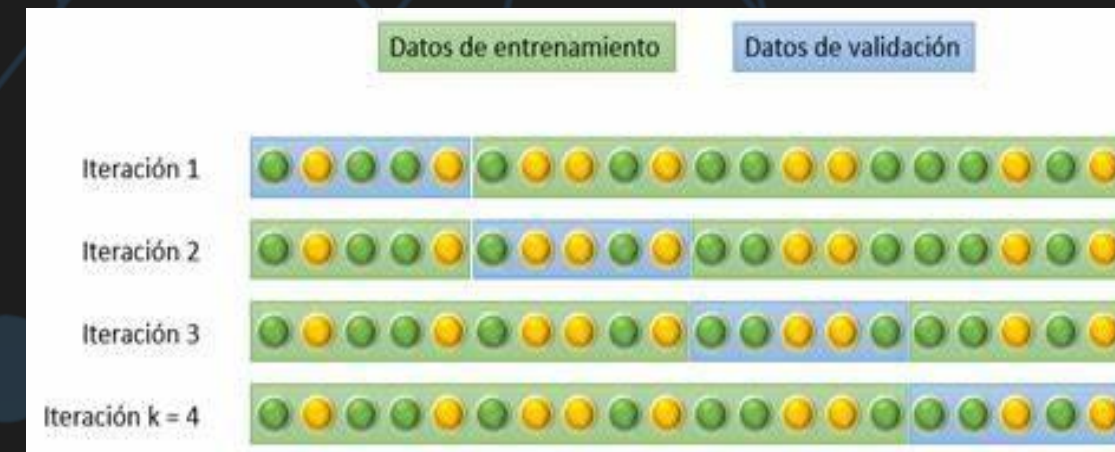
Es un caso especial de k -Fold donde k es igual al número de muestras. En cada iteración, se deja fuera una única muestra para evaluación y se entrena el modelo con el resto de los datos. Útil cuando el conjunto de datos es muy pequeño, maximizando la cantidad de datos para entrenamiento.

Técnicas de Validación Cruzada

Método de Retención (Hold-Out)



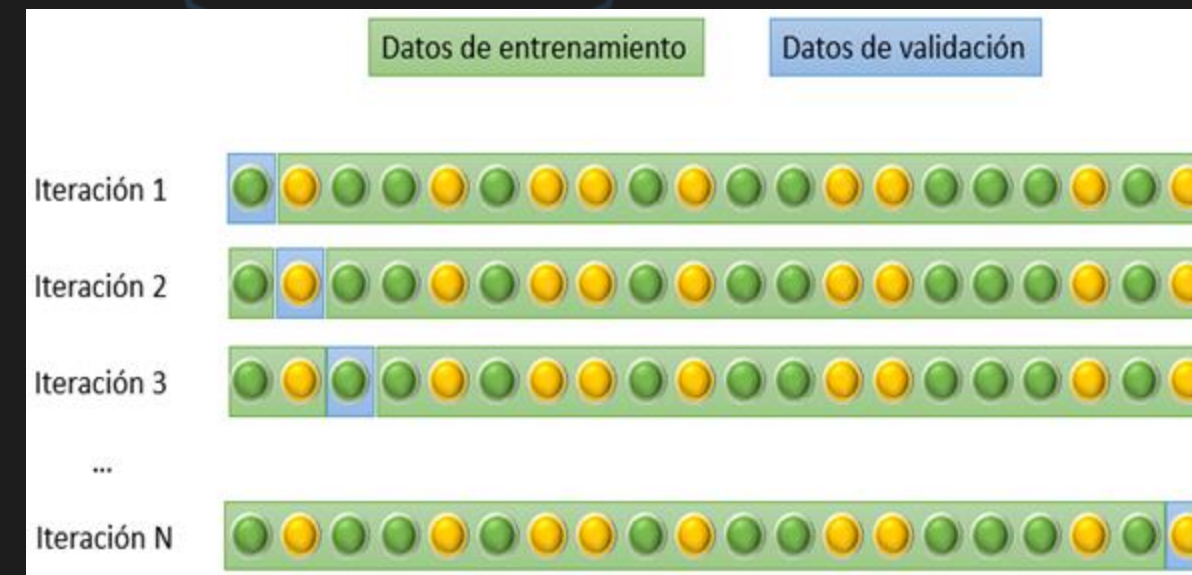
Validación Cruzada k-Fold



Random Subsampling



Leave-One-Out (LOO)



Implementación de Scikit-Learn

Importación de Librerías

Primero, es necesario importar las librerías necesarias, incluyendo Scikit-Learn y otras herramientas como NumPy o pandas para manejar los datos. Esto permite acceder a las funciones y clases específicas para aplicar diversas técnicas de validación cruzada.

Carga y Preparación de Datos

Scikit-Learn incluye conjuntos de datos de ejemplo, como el famoso dataset Iris, que se puede utilizar para demostrar la validación cruzada. Es importante preparar correctamente los datos, separando características y etiquetas.

Selección del Método

Scikit-Learn ofrece clases específicas para cada técnica: KFold para k-Fold, LeaveOneOut para LOOCV y ShuffleSplit para simular Random Subsampling.

Evaluación y Comparación

Utilizando `cross_val_score`, se puede evaluar fácilmente el rendimiento del modelo con diferentes técnicas de validación cruzada y comparar los resultados para seleccionar la más adecuada para el problema específico.

Implementación de Scikit-Learn

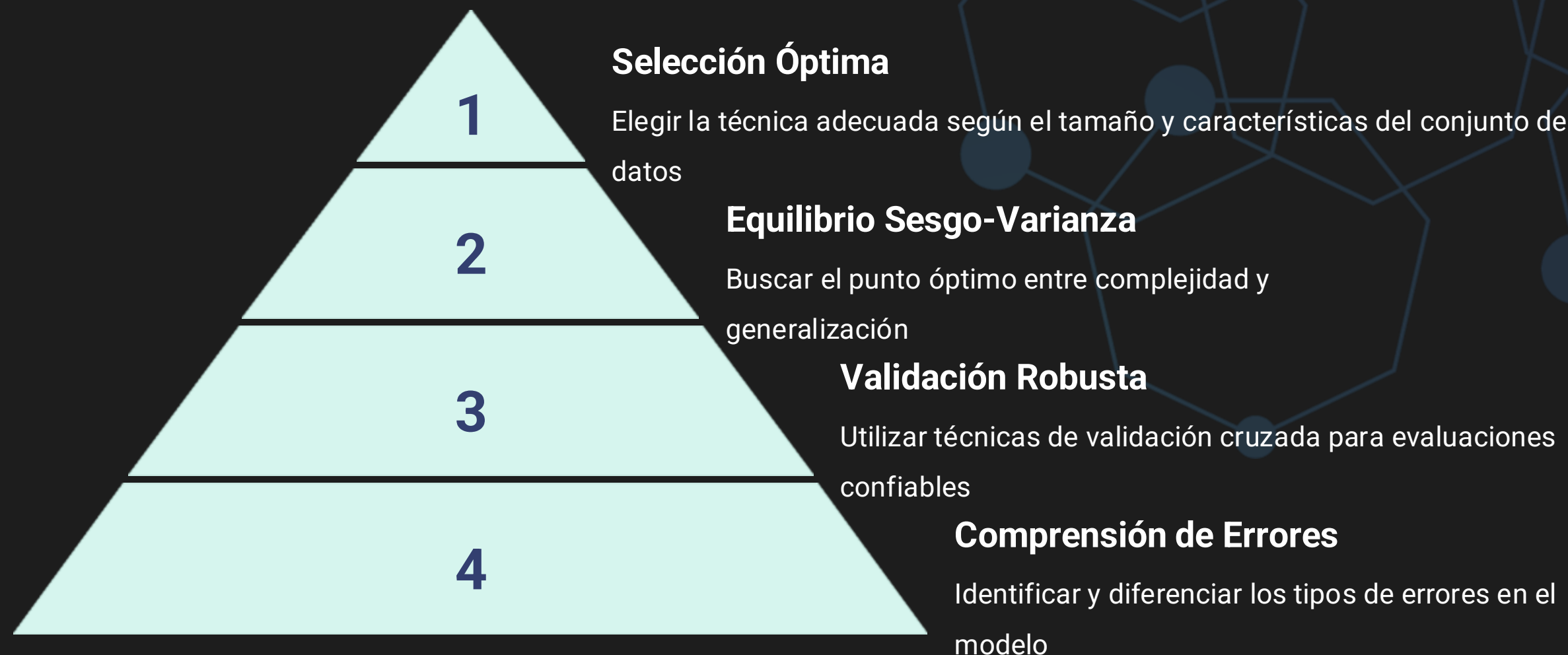
Requisitos:

1. Instala Scikit-Learn si aun no lo has instalado.
2. Importa las librerías.
3. Carga los Datos.
4. Implementa K-Fold Cross-Validation.
5. Implementa Leave-One-Out Cross-Validation.
6. Implementa Random Subsampling.
7. Compara los métodos.



El detalle de la actividad se encuentra en la guía de estudio de la sesión.

Conclusiones y Mejores prácticas

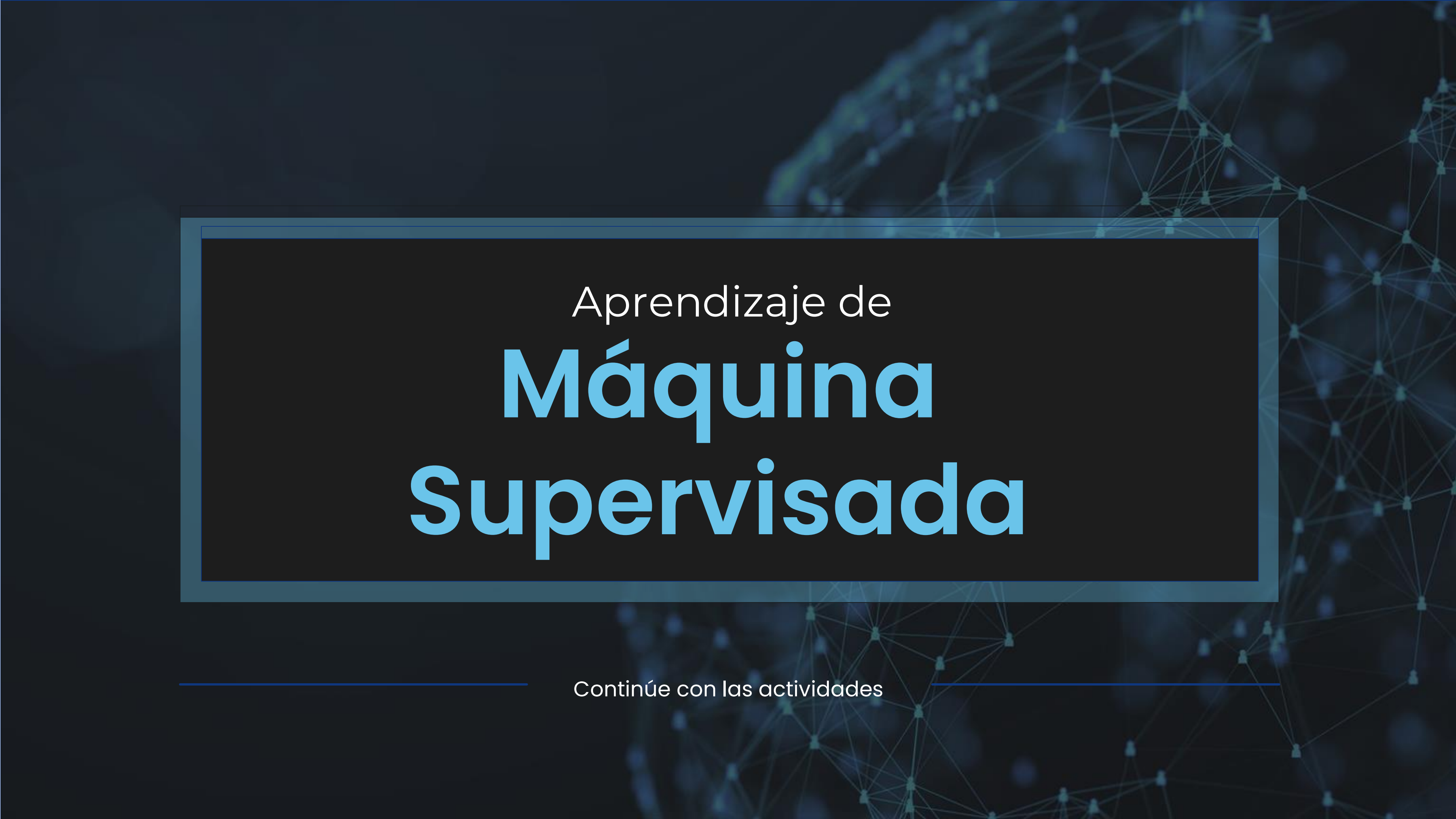


La validación cruzada es una herramienta fundamental para evaluar correctamente el rendimiento de los modelos de aprendizaje automático. Comprender los diferentes tipos de errores y el equilibrio entre sesgo y varianza permite desarrollar modelos que generalicen bien a datos nuevos. La elección de la técnica de validación adecuada depende del contexto específico, y Scikit-Learn facilita enormemente su implementación práctica.

Preguntas

Sección de preguntas



The background of the slide features a dark blue gradient with a complex, glowing network of white and light blue lines and nodes, resembling a neural network or data connectivity map.

Aprendizaje de **Máquina Supervisada**

Continúe con las actividades
