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 slide.

Aprendizaje de **Máquina Supervisado**

Sesión 4

Problemas de regresión

Los problemas de regresión son una categoría fundamental en el campo del aprendizaje automático (Machine Learning), donde el objetivo principal es predecir un valor continuo (una cantidad numérica) a partir de una o más variables de entrada (también conocidas como características o predictores).



Problemas de regresión

1 Objetivo

Predecir un valor continuo a partir de variables de entrada.

2 Características

Variable dependiente
continua. Variables independientes
numéricas o categóricas.

3 Aplicaciones

Predicción de precios, consumo de combustible, ventas y comportamientos.

Tipos de Regresión

Regresión Lineal

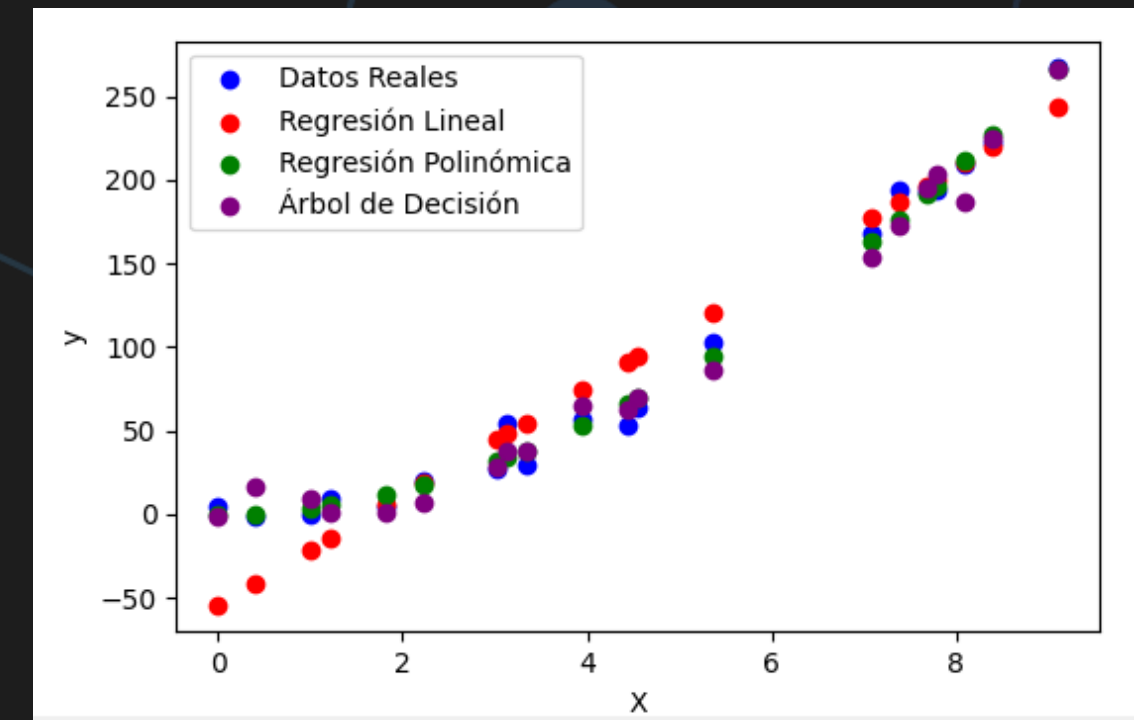
Relación lineal entre variables. Fórmula: $Y = \beta_0 + \beta_1 X + \epsilon$

Simple de entender e implementar. Eficiente computacionalmente.

Regresión No Lineal

Relación curvilínea entre variables. Incluye modelos polinómicos, exponenciales y logarítmicos.

Mayor flexibilidad y precisión. Riesgo de sobreajuste.



Scikit-Learn para Regresión

LinearRegression

Modelo básico para relaciones lineales. Ajusta una línea recta a los datos.

PolynomialFeatures

Transforma características en polinomios para modelar relaciones no lineales.

Ridge y Lasso

Añaden regularización para evitar sobreajuste. Ridge usa penalización L2, Lasso usa L1.

SVR

Support Vector Regression para problemas no lineales con estructuras complejas.

```
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures
from sklearn.pipeline import make_pipeline

# Crear el modelo de regresión lineal
model = LinearRegression()

# Crear el modelo de características polinómicas
poly = PolynomialFeatures(degree=2)

# Crear el modelo de regresión con características polinómicas
model = make_pipeline(poly, model)

# Entrenar el modelo
model.fit(X_train, y_train)

# Predecir
y_pred = model.predict(X_test)
```

Entrenamiento de Algoritmos

Preprocesamiento

Limpieza, normalización y transformación de datos. Codificación de variables categóricas.

División de Datos

Separar en conjuntos de entrenamiento y prueba. Aplicar validación cruzada.

Selección del Modelo

Elegir tipo de regresión según naturaleza del problema.

Entrenamiento

Ajustar parámetros minimizando función de costo.

Validación

Evaluar rendimiento con métricas como MSE y R^2 .



Métricas de Evaluación



Error Cuadrático Medio (MSE)

Promedio de errores al cuadrado entre valores reales y predichos. Menor MSE indica mejor ajuste.



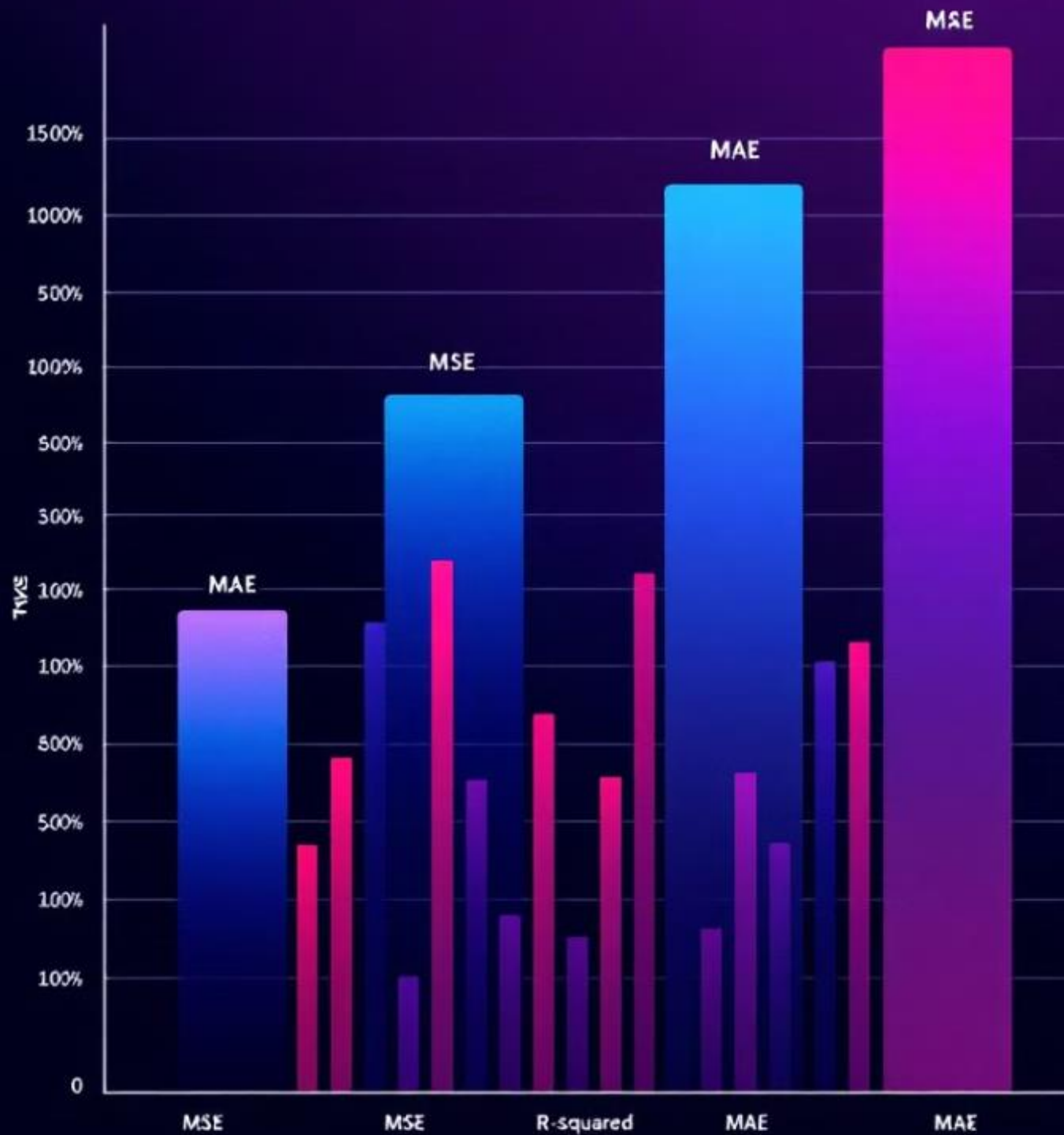
Coeficiente de Determinación (R^2)

Proporción de varianza explicada por el modelo. Valor cercano a 1 indica buen ajuste.



Error Absoluto Medio (MAE)

Promedio de errores absolutos. Menos sensible a valores atípicos que MSE.



Predicciones con el Modelo

1

Entrada de Datos

Nuevos datos con mismo formato que datos de entrenamiento.

2

Uso del Modelo

Aplicar modelo entrenado para calcular predicción.

3

Interpretación

Analizar valor continuo obtenido como estimación.

4

Validación

Verificar calidad de predicción con datos no vistos.

Selección del Modelo óptimo

Analizar Linealidad

Determinar si la relación es lineal o no lineal.

Evaluar Complejidad

Balancear simplicidad vs capacidad predictiva.

Aplicar Regularización

Usar Ridge, Lasso o Elastic Net según necesidad.

Considerar Interpretabilidad

Valorar comprensión del modelo vs precisión.

Medir Rendimiento

Comparar modelos con métricas y validación cruzada.



Actividad Práctica Guiada

MODELOS DE REGRESIÓN CON SCIKIT-LEARN

Objetivo: Comprender y aplicar modelos de regresión para resolver problemas de predicción utilizando Scikit-Learn.

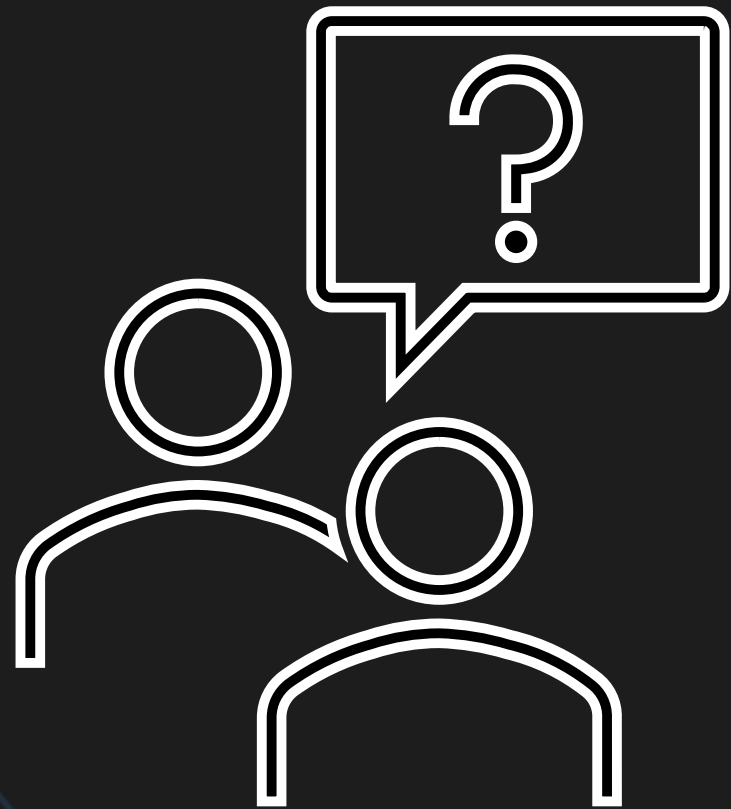
Pasos:

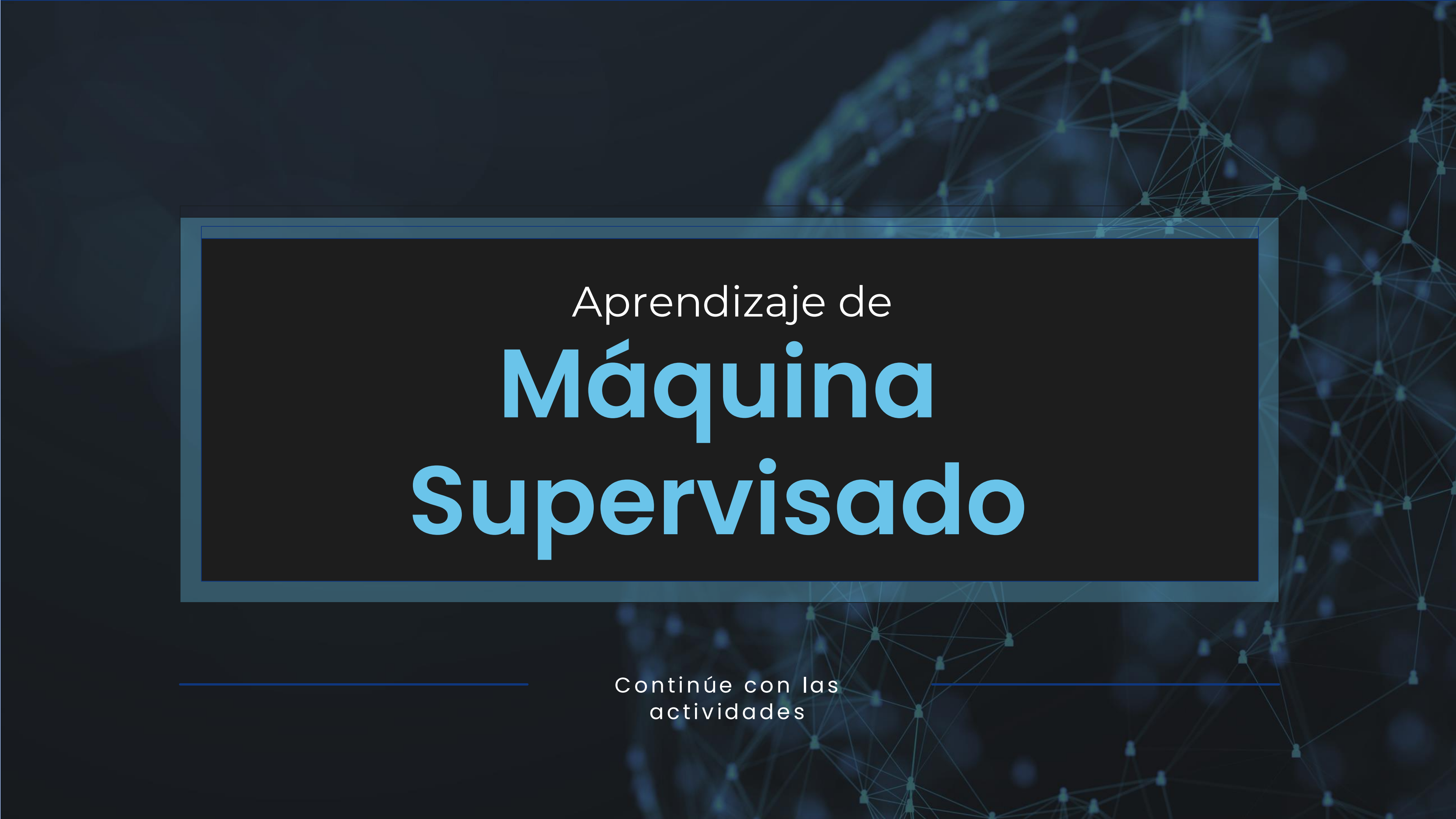
1. Importar librerías: numpy, matplotlib.pyplot, pandas, sklearn.
2. Generar datos sintéticos para mostrar una relación no lineal entre la variable independiente X y la variable dependiente Y.
3. Dividir los datos en conjuntos de entrenamiento y prueba.
4. Aplicar diferentes modelos de regresión.
5. Comparar los modelos y seleccionar el mejor para la ocasión.

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

Preguntas

Sección de preguntas



A background network diagram with blue nodes and connecting lines, creating a mesh-like structure across the entire slide.

Aprendizaje de **Máquina Supervisado**

Continúe con las
actividades
