

Machine Learning

Ing. Remigio Hurtado Ortiz, PhD.

Machine Learning

1. **Software**
2. **Evaluación de Rendimiento**
3. **Métricas clásicas de calidad**

Software

Existen diversos programas que se utilizan para formatear datos, limpiarlos, analizarlos y visualizarlos. **Python** se creó en 1991 como un lenguaje fácil de aprender con muchas bibliotecas utilizadas para manipulación de datos, aprendizaje automático y visualización de datos. Python es un lenguaje flexible que está creciendo y volviéndose cada vez más importante para la ciencia de datos gracias a su flexibilidad y facilidad de aprendizaje.

Jupyter Notebook permite que la instrucción y la programación formen parte del mismo archivo (libreta de anotaciones – cuadernos de Jupyter). Es fácil alterar código en las computadoras portátiles y experimentar cómo diferentes códigos se pueden utilizar para manipular, analizar y visualizar datos.

Bibliotecas:

NumPy: agrega soporte para matrices, tiene muchas funciones matemáticas.

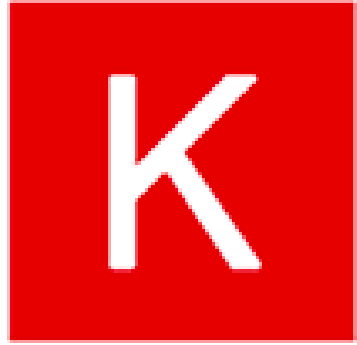
Guía rápida: https://numpy.org/doc/stable/user/absolute_beginners.html

Pandas: agrega soporte para las tablas y las series de tiempo, se utiliza para manipular y limpiar datos, entre otras acciones. Agrega estructuras de datos de alto rendimiento y herramientas para el análisis de grandes conjuntos de datos. Para sistemas o bases de datos distribuidas y Big Data, la opción es **Apache Spark**.

Guía rápida: https://pandas.pydata.org/docs/user_guide/10min.html

Matplotlib: agrega soporte para la visualización de datos. Matplotlib es una biblioteca de trazado de gráficos capaz de crear desde simples diagramas de línea hasta complejos diagramas 3D y de contorno.

Software



Keras

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$



Software

Para Data Science y Machine Learning:

Microsoft Power BI, Orange 3, Pentaho, Tableau, Statgraphics, Minitab, SPSS, S-PLUS, R, MATLAB, GAUSS, OCTAVE, PYTHON, Weka.

Para Machine Learning:

- **Scikit-learn** es una biblioteca popular de **aprendizaje automático**. Esta biblioteca contiene muchas herramientas útiles para el análisis de datos y se basa en NumPy, SciPy y matplotlib.
- **TensorFlow** es una biblioteca de código abierto para aprendizaje automático a través de un rango de tareas, y desarrollado por Google para satisfacer sus necesidades de sistemas capaces de construir y entrenar **redes neuronales** para detectar y descifrar patrones y correlaciones, análogos al aprendizaje y razonamiento usados por los humanos.
- **Keras** es una biblioteca de **redes neuronales** de Código Abierto escrita en Python. Es capaz de ejecutarse sobre TensorFlow, Microsoft Cognitive Toolkit o Theano. Está especialmente diseñada para posibilitar la experimentación con redes de **aprendizaje profundo**.

Software

OpenCV: biblioteca libre de visión artificial originalmente desarrollada por Intel. Ejemplos de aplicación: estimar la velocidad de un vehículo, reconocer imágenes (como logos), etc.



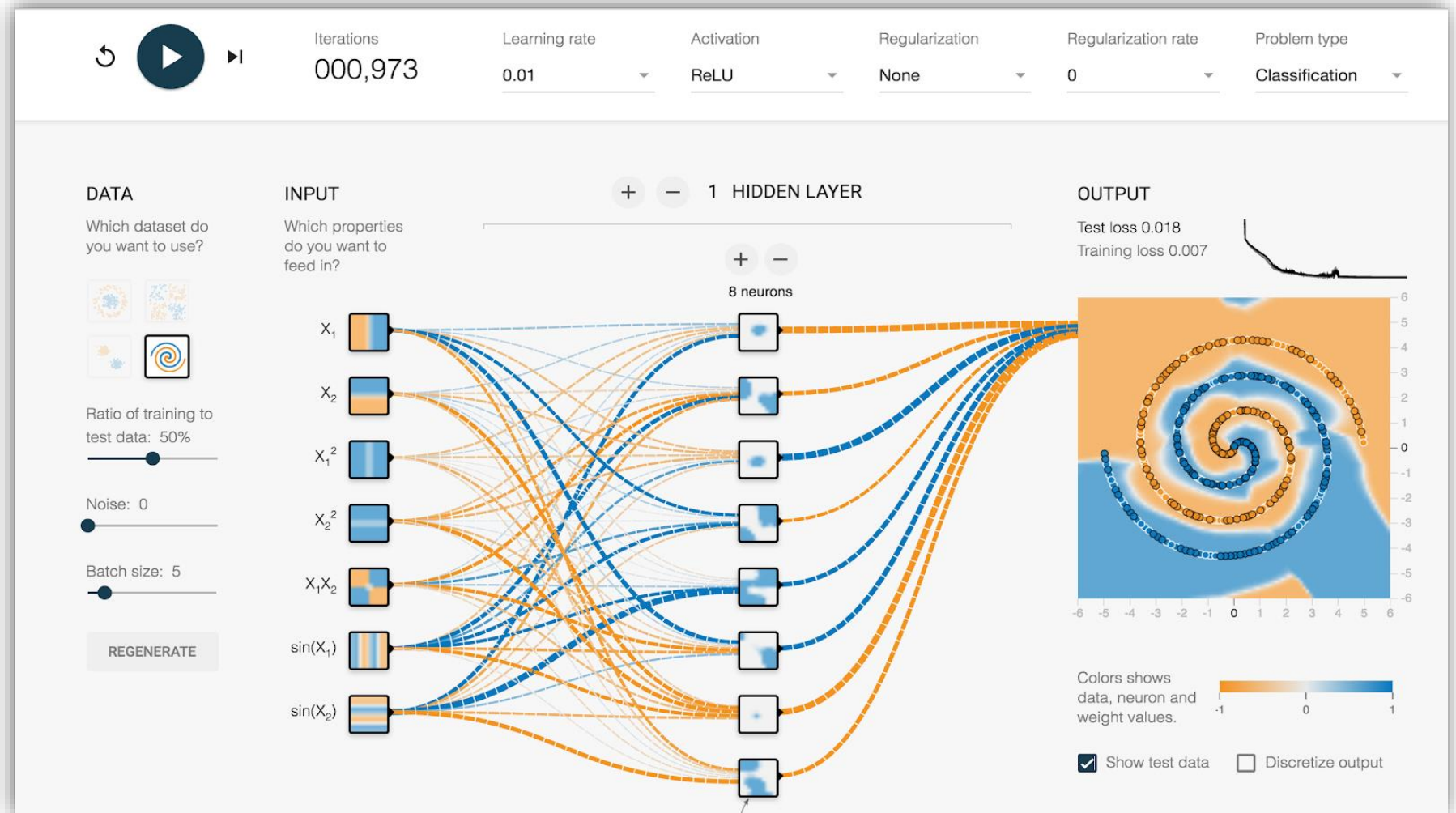
Software



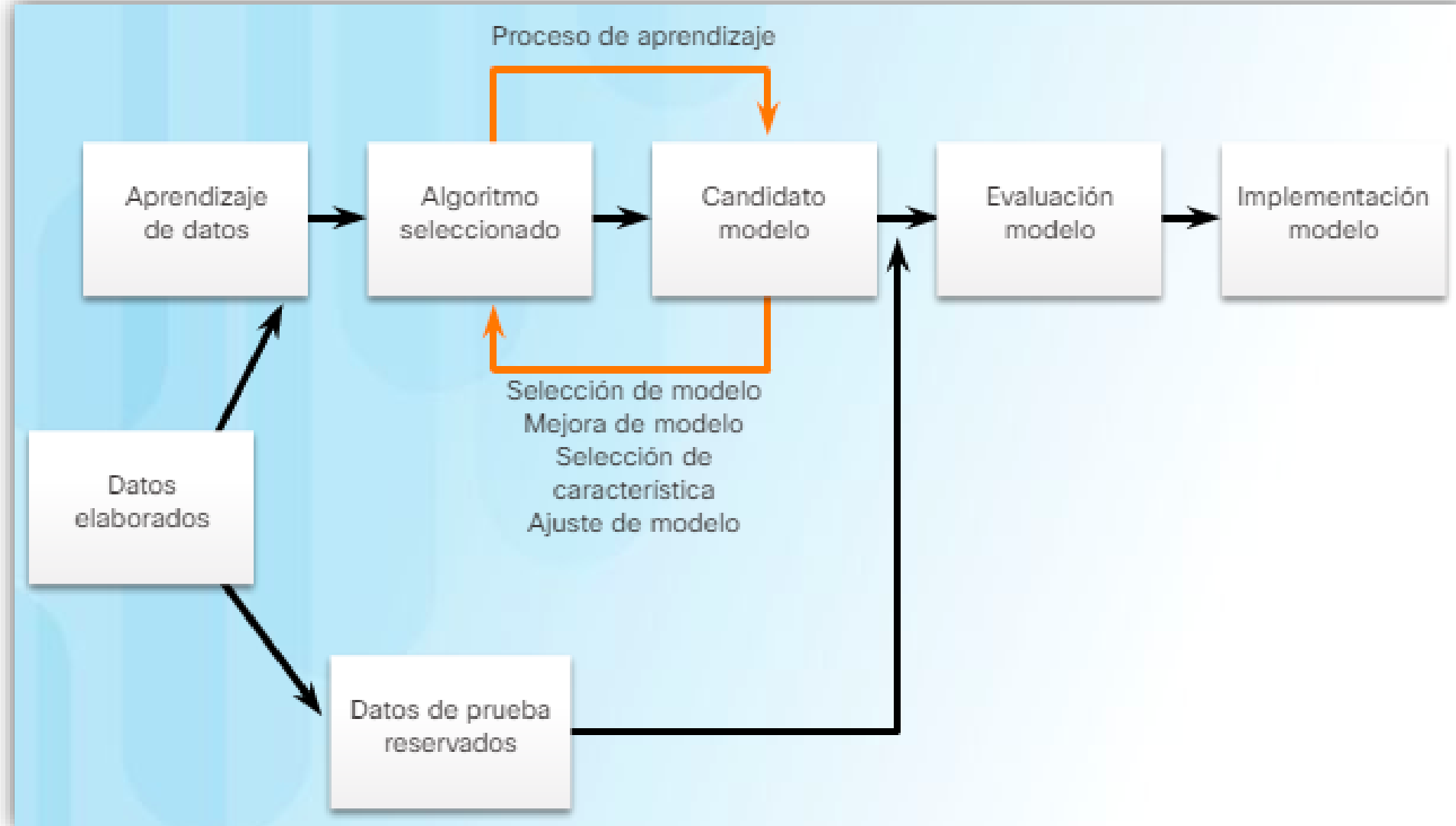
Software

Software para aprendizaje de Redes Neuronales: TensorFlow Playground

TensorFlow Playground es una aplicación **web de visualización interactiva** escrita en JavaScript que permite **simular redes neuronales simples** que se ejecutan en el navegador, y además, permite ver los resultados en tiempo real.



Evaluación de Rendimiento de Algoritmos: Selección y Evaluación de Modelos



Evaluación de Rendimiento de Algoritmos: Diseño de Experimentos

Datasets

Dataset 1
Dataset 2
Dataset 3

Método propuesto

NUESTRO
METODO

Medidas de Calidad

Medidas de Regresión

MAE

MSE

RMSE

Medidas de Clasificación

Precision

Recall

Baselines

METODO 1

METODO 2

METODO 3

METODO 4

METODO 5

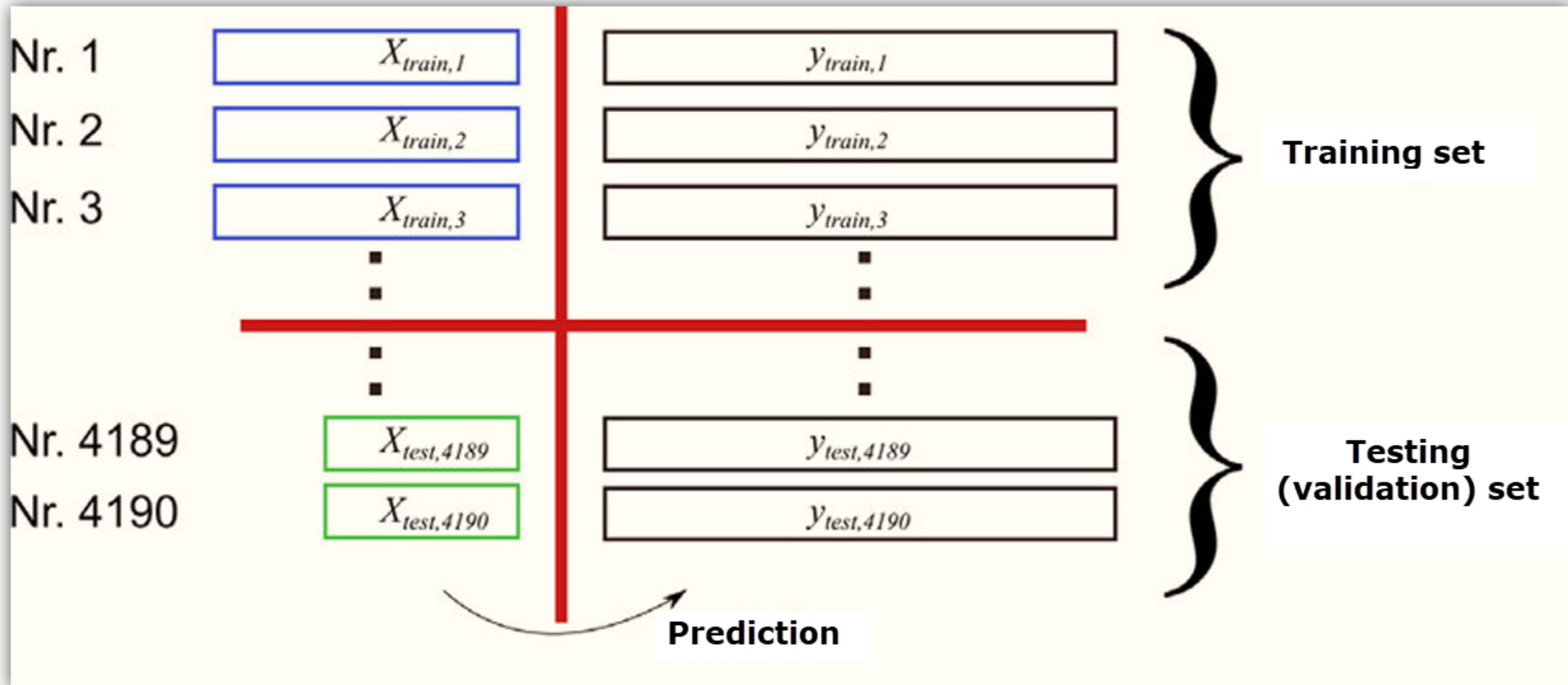
Modelos de Trabajos de Investigación

Modelo 1

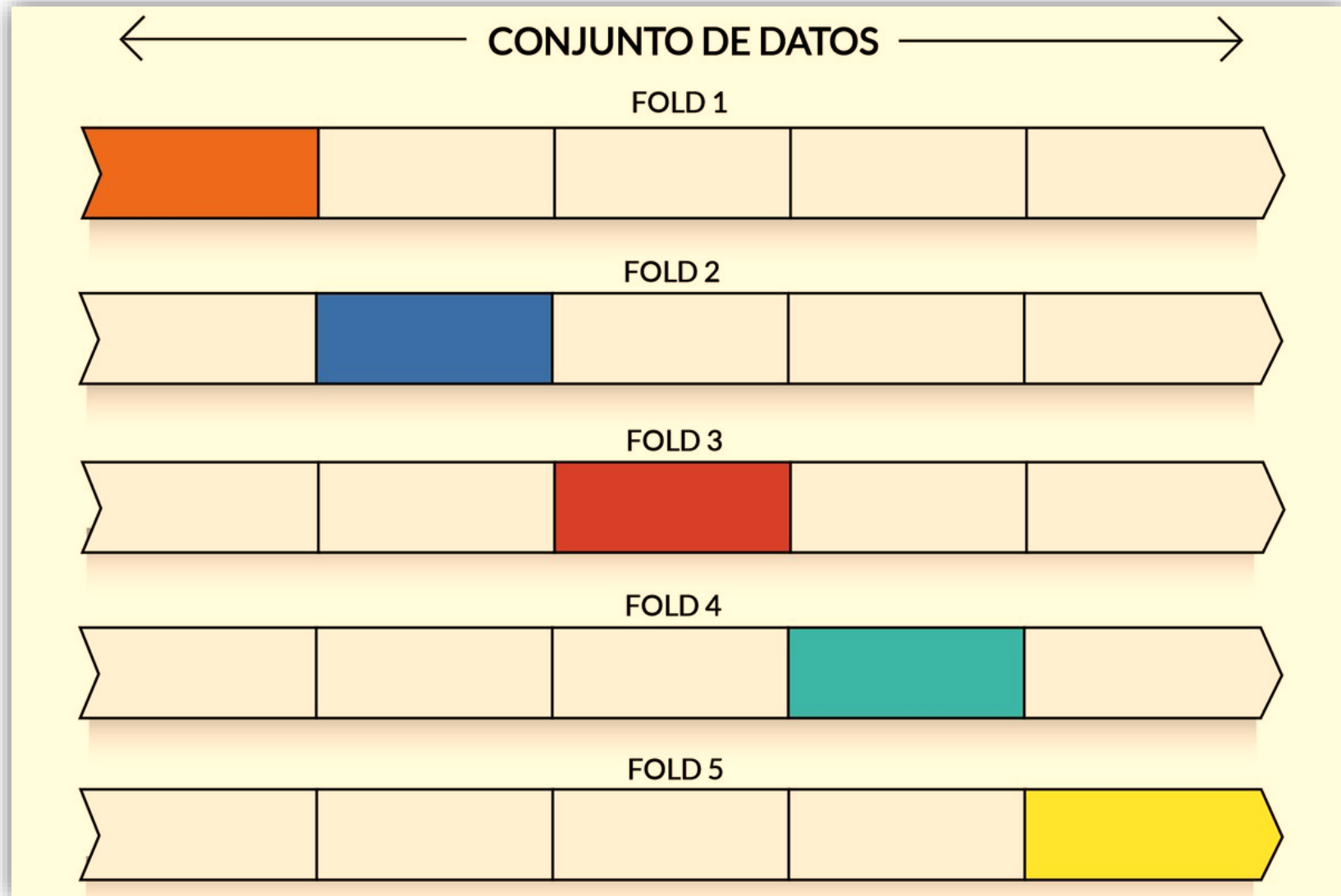
Modelo 2

Modelo 3

Evaluación de Rendimiento de Algoritmos: Validación Cruzada - Split en Train y Test



Evaluación de Rendimiento de Algoritmos: Validación Cruzada (Cross Validation) K Folds/Pliegues (K=5)



Evaluación de Rendimiento de Algoritmos: Validación Cruzada (Cross Validation) K Folds/Pliegues (K=5)



Evaluación de Rendimiento de Algoritmos: Medidas de Calidad

En los **algoritmos de regresión** se utilizan comúnmente tres métricas de evaluación:

1. Mean Absolute Error (MAE):

es la media de los valores absolutos de los errores.

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|$$

2. Mean Squared Error (MSE):

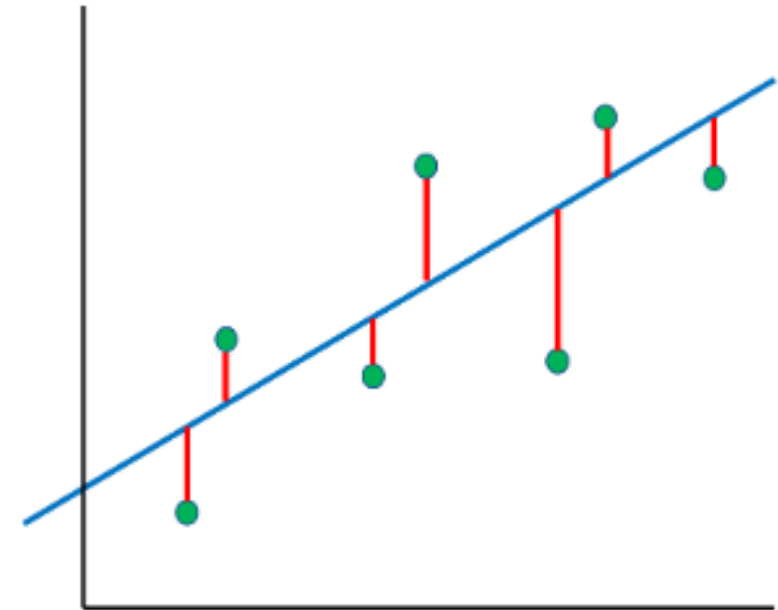
es la media de los errores cuadráticos.

$$MSE = \frac{1}{M} \sum_{i=1}^M (real_i - estimado_i)^2$$

3. Root Mean Squared Error (RMSE):

es la raíz cuadrada de la media de los errores cuadráticos

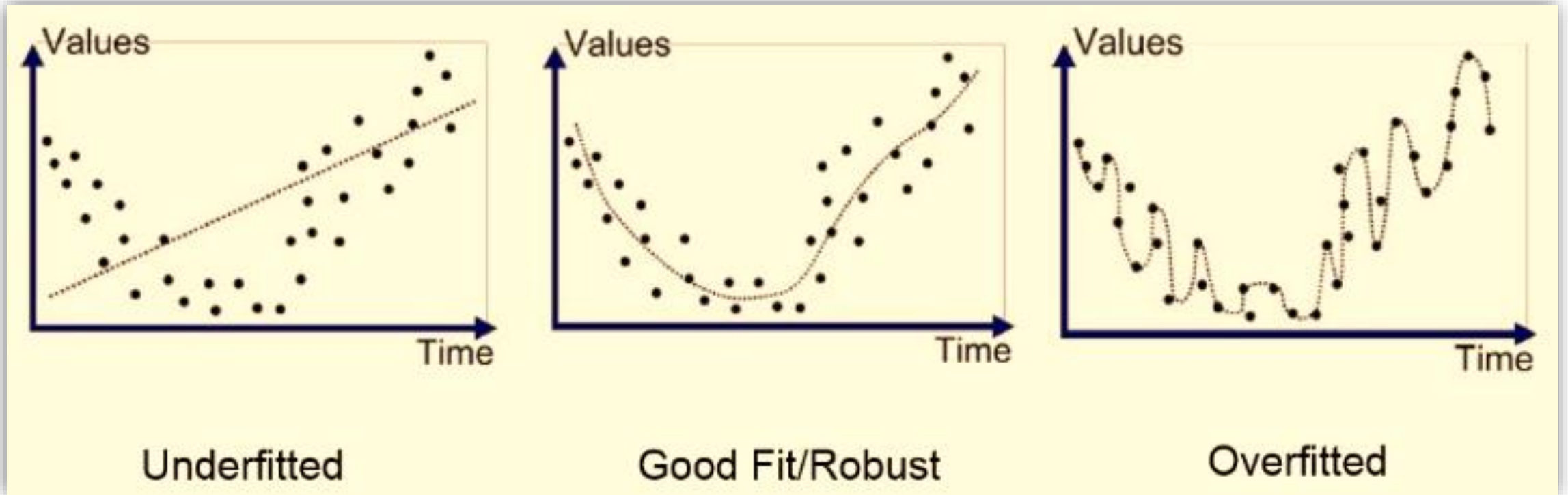
$$RMSE = \sqrt{\frac{1}{M} \sum_{i=1}^M (real_i - estimado_i)^2}$$



Error en Regresión Lineal

Evaluación de Rendimiento de Algoritmos: Medidas de Calidad

Análisis de Resultados



Evaluación de Rendimiento de Algoritmos: Medidas de Calidad

Análisis de Resultados

Dependiendo de la evaluación de los resultados al aplicar un algoritmo, se pueden analizar y obtener varias conclusiones, por ejemplo:

Un bajo accuracy (o alto error) puede indicar que:

- Se **necesita más datos**: solo un año de datos no es mucho, mientras que tener varios años podría ayudar a mejorar la precisión.
- Hay **malas suposiciones**: asumimos que estos datos tienen una relación lineal, pero puede que ese no sea el caso. Visualizar los datos puede ayudar a determinar esto.
- Las **funciones son deficientes**: es posible que las funciones que se utilizaron no hayan tenido una correlación lo suficientemente alta con los valores que se intentaron predecir.

Evaluación de Rendimiento de Algoritmos: Medidas de Calidad

En los **algoritmos de clasificación** se utiliza comúnmente la matriz de confusión:

Error tipo I: Falsos positivos	Error tipo II: Falsos negativos
<p>El resultado es positivo, está Vd. embarazado</p>  A photograph of a male doctor in a white lab coat holding a blue clipboard and pen, looking at an elderly man in a grey shirt who has a surprised expression.	<p>El resultado es negativo, no está Vd. embarazada</p>  A photograph of a female doctor in a white lab coat looking down at a pregnant woman in a white shirt who is also looking down.

Primera palabra:
¿máquina y ser humano están de acuerdo o no?
Falso: no están de acuerdo
Verdadero: están de acuerdo

Evaluación de Rendimiento de Algoritmos: Medidas de Calidad

En los **algoritmos de clasificación** se utiliza comúnmente la matriz de confusión:

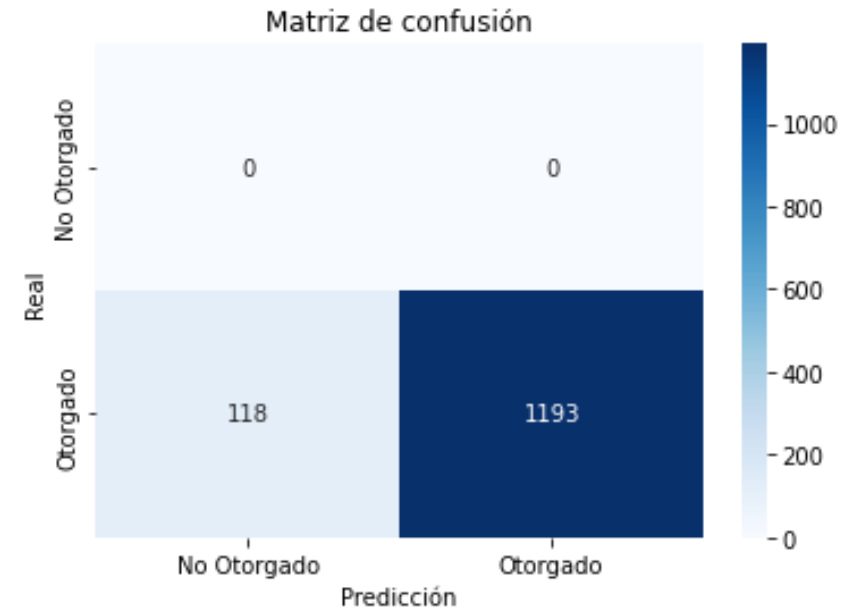
Precision: fracción de predicciones positivas que fueron realmente correctas (es decir, indica si hay errores en las predicciones positivas). Cuando falsos positivos son indeseados o costosos: sistemas de recomendación.

Recall (sensitividad): fracción de casos reales positivos que fueron predichos correctamente. Mide la capacidad del modelo para identificar correctamente todos los casos positivos reales. Cuando falsos negativos son indeseados o costosos: diagnóstico médico. No perder ningún caso de enfermedad.

	no default (0)	default (1)
no default (0)	TN	FP
default (1)	FN	TP

$$P = \frac{TP}{TP + FP}$$

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



Evaluación de Rendimiento de Algoritmos: Medidas de Calidad

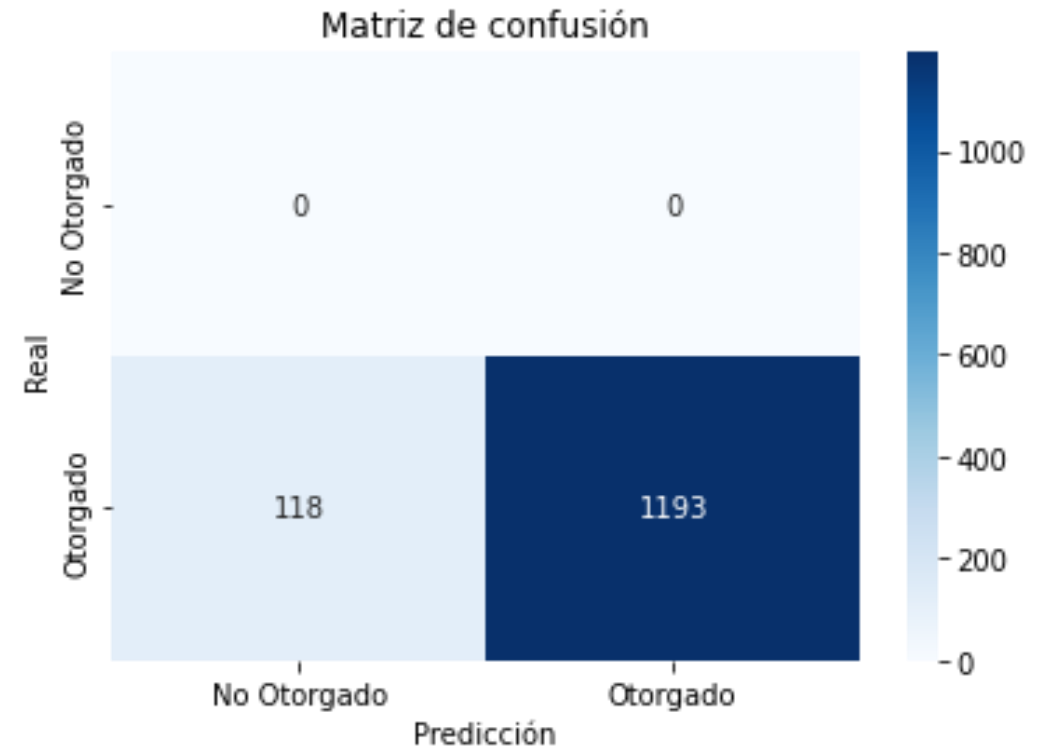
$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$E = \frac{TN}{TN + FP}$$



Evaluación de Rendimiento de Algoritmos: Medidas de Calidad

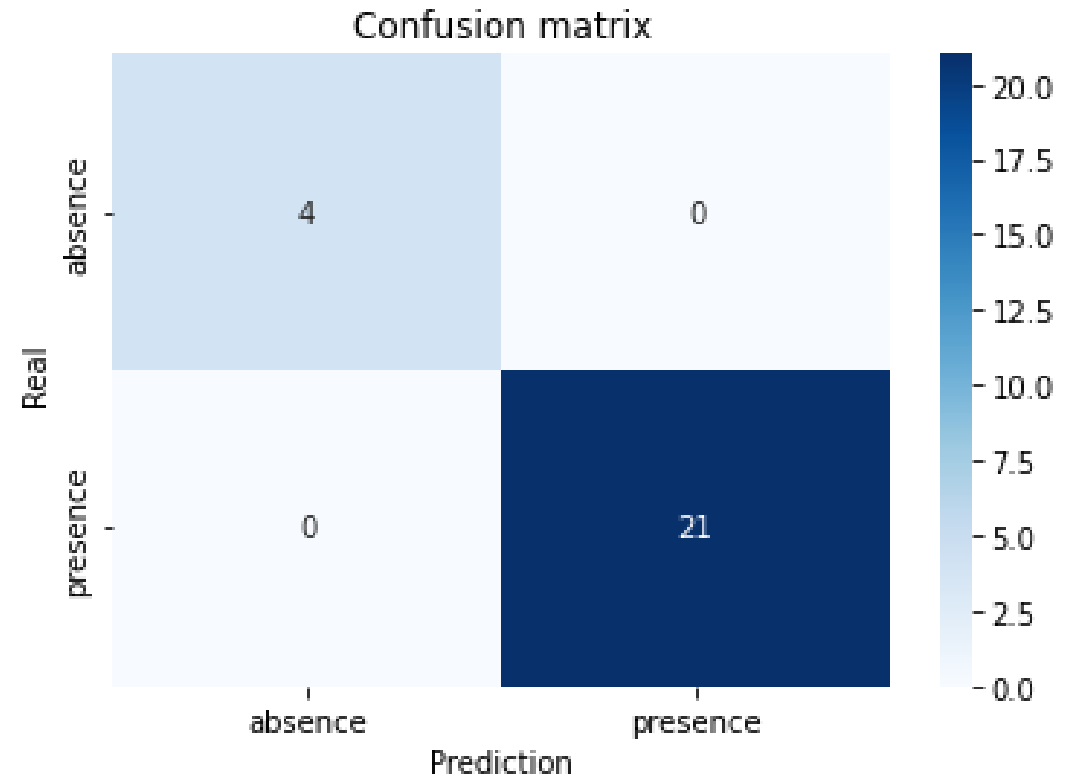
$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

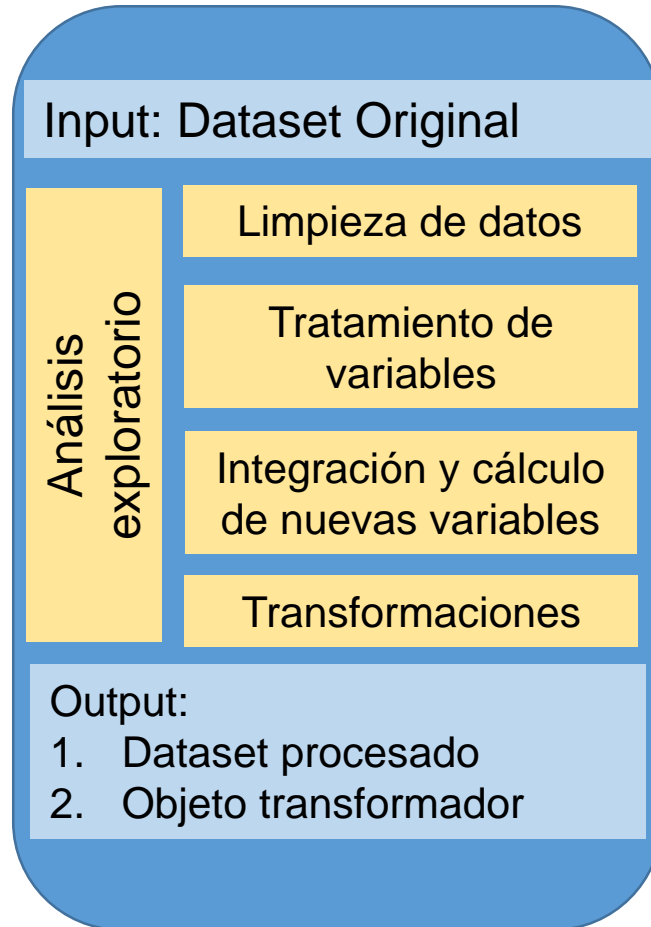
$$E = \frac{TN}{TN + FP}$$



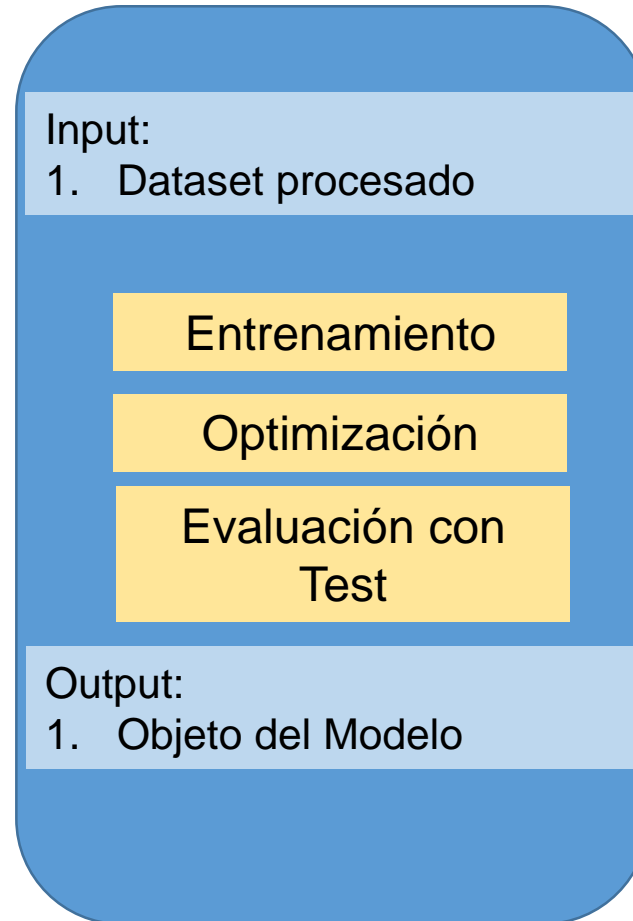
Métrica de Calidad	Definición	Cuándo se Utiliza	Ejemplo
Accuracy	Proporción de predicciones correctas.	Clasificación binaria y multiclase.	Clasificación de correos electrónicos.
Precision	Capacidad de predecir positivos de manera precisa.	Problemas donde los falsos positivos son costosos .	Detección de spam en correos electrónicos. Sistemas de recomendación.
Recall (Sensibilidad)	Capacidad de identificar correctamente todos los positivos reales.	Problemas donde los falsos negativos son costosos .	Detección de enfermedades en pruebas médicas.
F1-Score	Media armónica entre precision y recall.	Problemas donde se busca equilibrar precision y recall.	Clasificación de fraudes en transacciones financieras.
Especificidad (E)	Capacidad de identificar correctamente todos los negativos. Es la fracción de casos reales negativos que fueron predichos correctamente.	Problemas donde los falsos positivos son críticos .	Pruebas de seguridad de aeropuertos.
Curva ROC (Receiver Operating Characteristic)	Gráfico que representa la tasa de verdaderos positivos frente a la tasa de falsos positivos en varios umbrales.	Evaluación de modelos de clasificación binaria.	Evaluación de un modelo de detección de cáncer.

Pipelines

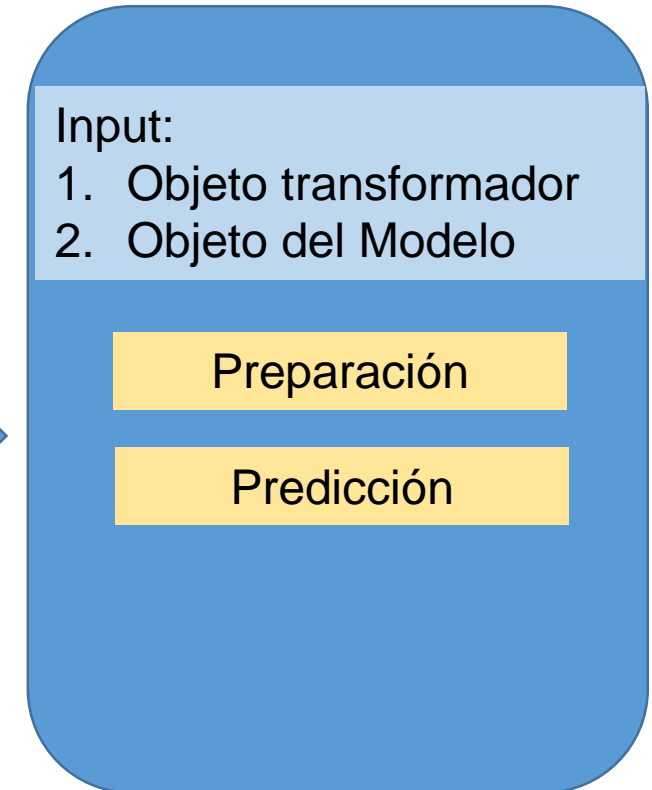
Fase 1: Recolección y Preparación de datos



Fase 2: Modelado (Aprendizaje)



Fase 3: Predicción con Nuevos Samples



Referencias

- Bishop, C. M. (2006). Pattern recognition and machine learning. springer.
- El Naqa, I., & Murphy, M. J. (2015). What is machine learning?. In machine learning in radiation oncology (pp. 3-11). Springer, Cham.
- Duda, R. O., Hart, P. E., & Stork, D. G. (2012). Pattern classification. John Wiley & Sons.
- P. Joshi. (2017). Artificial intelligence with python. Packt Publishing Ltd.
- Brownlee. J. (2019, Agosto 12). A Tour of Machine Learning Algorithms [Machine Learning Mastery]. Recuperado de <https://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/>
- S. Raschka, V. Mirjalili. (2007). Python Machine Learning, Packt Publishing Ltd.
- Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science.
- Müller, A. (2020). More on Pipelines. [Applied Machine Learning with Python]. Recuperado de <https://amueller.github.io/aml/01-ml-workflow/12-pipelines-gridsearch.html>
- Martinez, J. (2020). Precision, Recall, F1, Accuracy en clasificación [IArtificial.net]. Recuperado de <https://www.iartificial.net/precision-recall-f1-accuracy-en-clasificacion/>