



# **INTELIGENCIA ARTIFICIAL Y MACHINE LEARNING**

# **ESTRUCTURA DE LA CLASE**

## **INTRODUCCIÓN**

**Presentación del tema de la clase**

**Introducción a la IA y ML**

## **METODOLOGÍA**

**Pasos a seguir**

**Tipos de entrenamiento**

## **MANIPULACIÓN DE DATOS**

**Lectura de archivos con pandas**

**Análisis de datos con pandas**

**Conjuntos de datos de entrenamiento y pruebas**

## **MODELOS**

**Selección de modelos**

**Regresión logística y perceptrón**

**Evaluación de modelos: Exactitud**

## **PARÁMETROS**

**Significado y uso**

## **ACTIVIDAD PRÁCTICA**

**ChomAI**

## **CONCLUSIONES**

**Resumen de los puntos clave de la clase**



INTRODUCCIÓN A LA IA Y ML

LA IA ES SOLO UN  
NOMBRE FANCY PARA LLAMAR AL AJUSTE DE  
CURVAS



# M E T O D O L O G Í A D E I A

## **PROBLEMA**

**Dar contexto del problema que se quiere resolver para entender qué se quiere lograr**

## **SOLUCIÓN**

**Descripción a la solución deseada para resolver el problema**

## **OBJETIVOS**

**Planteamiento de los objetivos para lograr obtener la solución propuesta**

## **RECOPILACIÓN DE DATOS**

**Búsqueda de los datos del problema**

# M E T O D O L O G Í A D E I A

## **ANÁLISIS Y PREPROCESAMIENTO DE LOS DATOS**

Entender cómo usar esos datos y manipularlos para su uso

## **IMPLEMENTACIÓN Y ENTRENAMIENTO DE MODELOS**

Seleccionar un modelo que resuelva el problema

## **ANÁLISIS DE RESULTADOS**

Evaluar el modelo y extraer conclusiones de los resultados

## **DESPLIEGUE DEL MODELO**

Llevar el modelo a un ambiente de producción

# LECTURA DE ARCHIVO CON PANDAS



```
# Leer archivo csv con Pandas
```

```
import pandas as pd
```

```
df = pd.read_csv(archivo_csv, encoding='utf-8')
```

# SERIES Y DATAFRAMES

## DATAFRAME

Estructura de datos en forma de tabla (bidimensional)

Col	Col2	Col3
a	1	4
b	2	5
c	3	6

## SERIES

Estructura de datos en forma de lista (unidimensional)

- a
- b
- c
- d
- e

# FILAS Y COLUMNAS DE DATAFRAMES



```
# Acceder columna y fila  
fila = df.loc[indice_fila]  
columna = df[nombre_columna]
```



# ANÁLISIS DE DATOS CON PANDAS



*# Imprime las primeras 5 líneas del dataset*

```
print(df.head())
```

*# Obtiene estadísticas relevantes para el dataset*

```
print(df.describe())
```

*# Obtine información del dataset como tipos de datos o si hay valores nulos*

```
print(df.info())
```

# ENTRADAS Y SALIDAS

## FEATURES

**Son los datos que entran al modelo y que usa para generar una respuesta**

## TARGETS

**Es la respuesta que se espera que genere el modelo ante un conjunto de features**

**Cuando el aprendizaje es supervisado, a un modelo se le entrena dándole las features y targets para que aprenda a generalizarlas.**

# SEPARACIÓN DE DATOS

**ENTRENAMIENTO**

**VALIDACIÓN**

**PRUEBAS**

# SEPARACIÓN DE DATOS

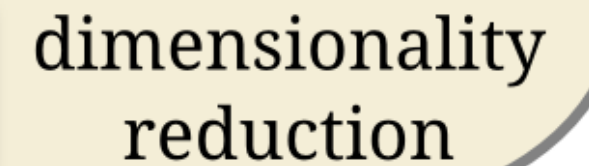
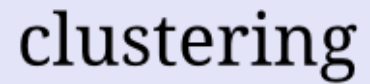
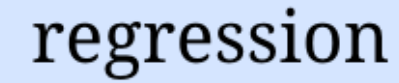


*# Separa los datos*

```
from sklearn.model_selection import train_test_split
```

```
X_train, X_test, y_train, y_test = train_test_split(features,  
targets, test_size=0.2, random_state=42)
```

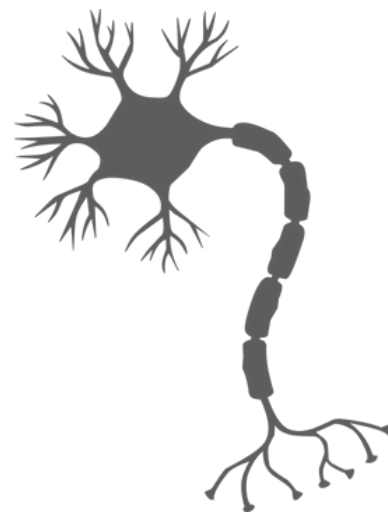
# classification



# PERCEPTRÓN Y REGRESIÓN LOGÍSTICA

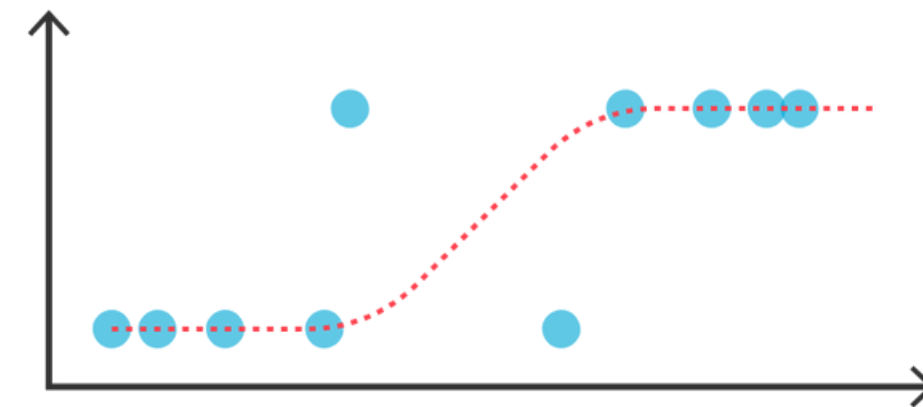
## PERCEPTRON

---



## REGRESIÓN LOGÍSTICA

---



**A efectos prácticos, no existe diferencia entre un perceptrón único con función de activación sigmoide y una regresión logística**

# CÁLCULO DE EXACTITUD

La exactitud es la proporción de instancias evaluadas correctamente



```
# Calcula la exactitud  
from sklearn.metrics import accuracy_score  
  
accuracy = accuracy_score(y_test, y_pred)
```

## PARÁMETROS

$$25X_1 + \frac{3}{4}X_2 + 3.14159265X_3 + 16 = 0$$



# CONCLUSIONES

## IA Y ML

Creación de modelos que aprendan a generalizar a partir de unas entradas dadas

## MANIPULACIÓN DE DATOS

Leemos archivos con pandas que genera dataframes y series.  
Analizamos esos datos y los separamos en conjuntos.

## REGRESIÓN LOGÍSTICA

Uno de los modelos más sencillos y fundamentales del ML y que se relaciona con el DeepLearning a través del perceptrón