

6. Machine Learning

Instructor: Darian Harrison Ragle
IMJU León
<https://campusimju.com/>
León, Gto. Mx.



Introducción

Que es la Inteligencia Artificial


Utilización de herramientas análisis numérico y abstracto para solucionar algún problema de forma mecánica o matemática.

Aplicaciones

Principales aplicaciones


- Procesamiento de Lenguaje Natural
- Visión computarizada
- Motores de recomendación
- Detección de anomalías
- Modelos generativos

Procesamiento de lenguaje natural




My experience so far has been fantastic!

POSITIVE



The product is ok I guess

NEUTRAL



Your support team is useless

NEGATIVE

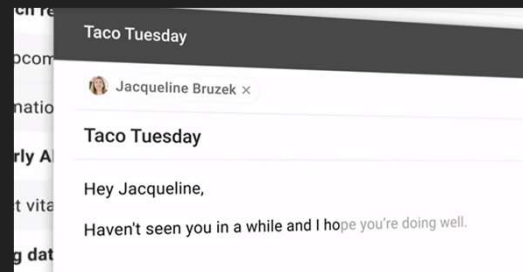


If you have any questions about my rate or **if you** find it necessary to increase or decrease **the scope** for this ~~project~~ **project's scope**, please let me know.

Rephrase

Dismiss

! Spam	22
Trash	
Categories	
👤 Social	4,500
📄 Updates	8,147
💬 Forums	75
📌 Promotions	10,482



hola mundo

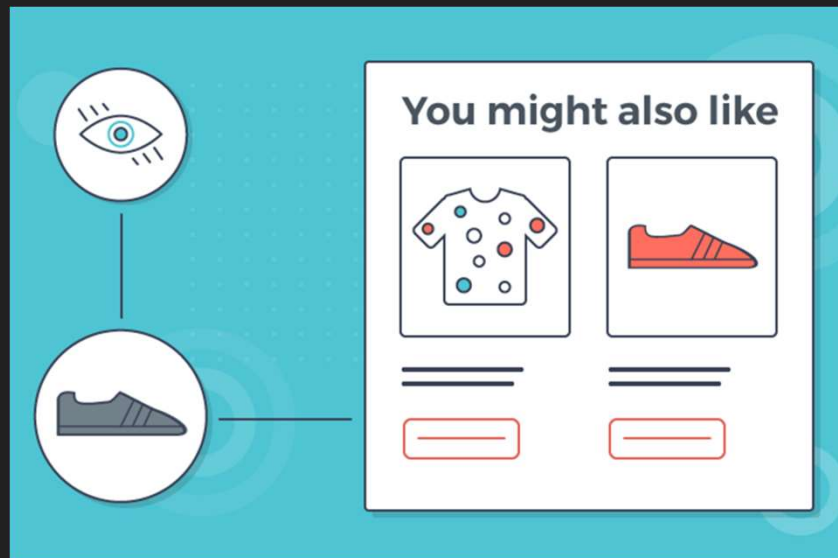
🔊

こんにちは世界

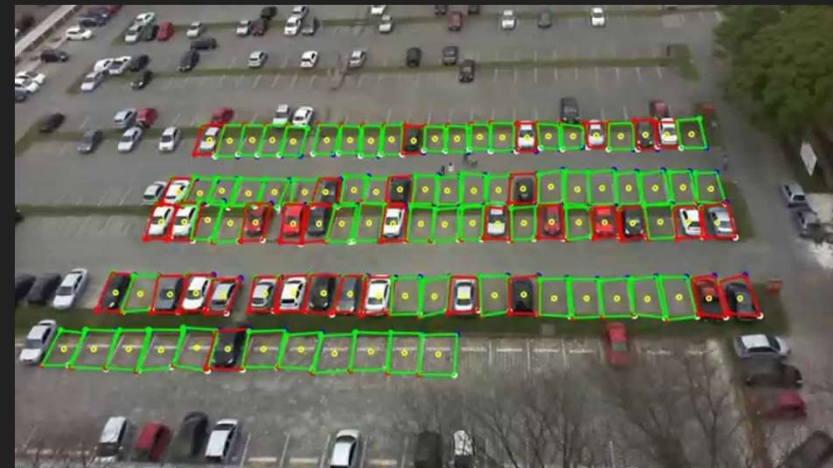
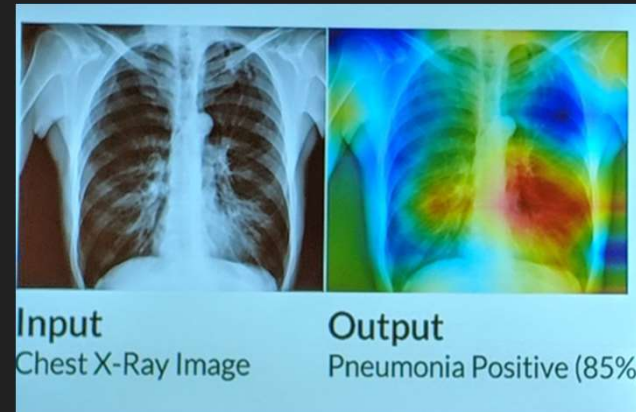
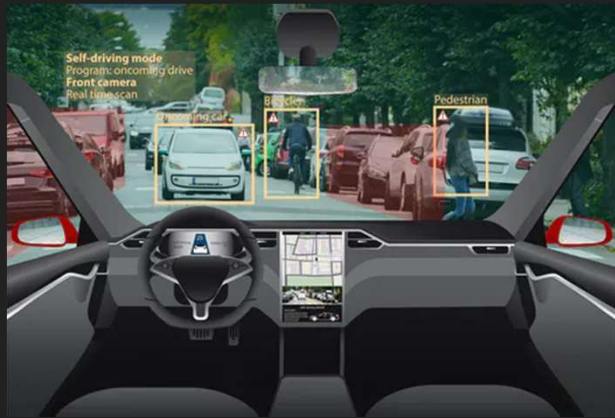
Kon'nichiwa sekai

📄 🔊

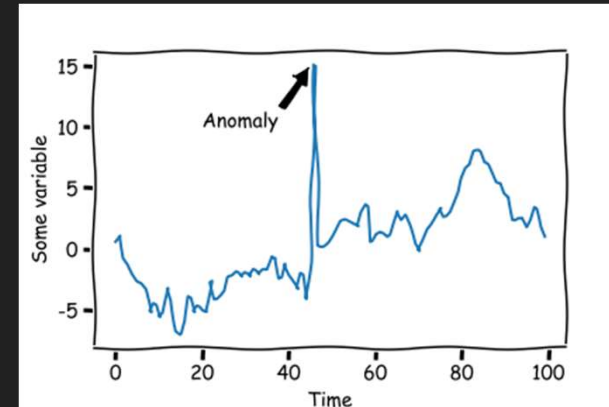
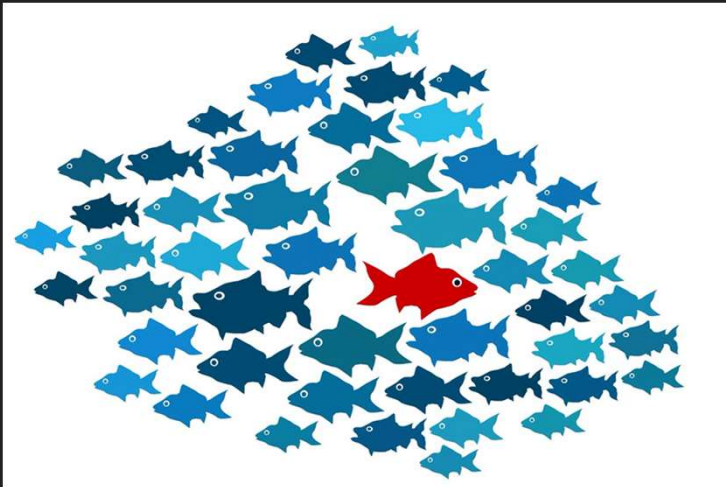
Motores de recomendación



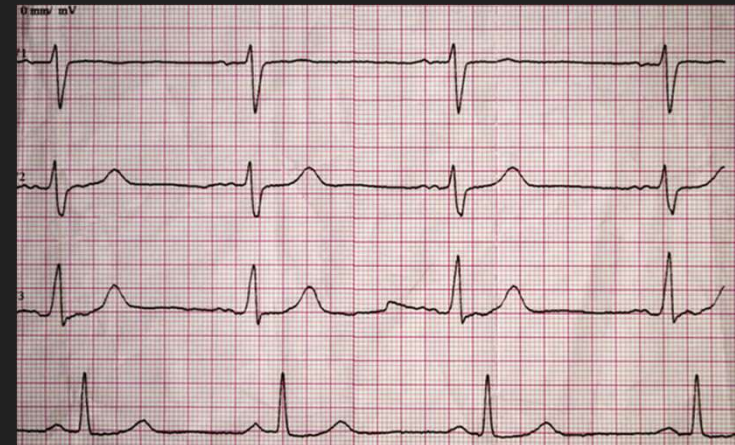
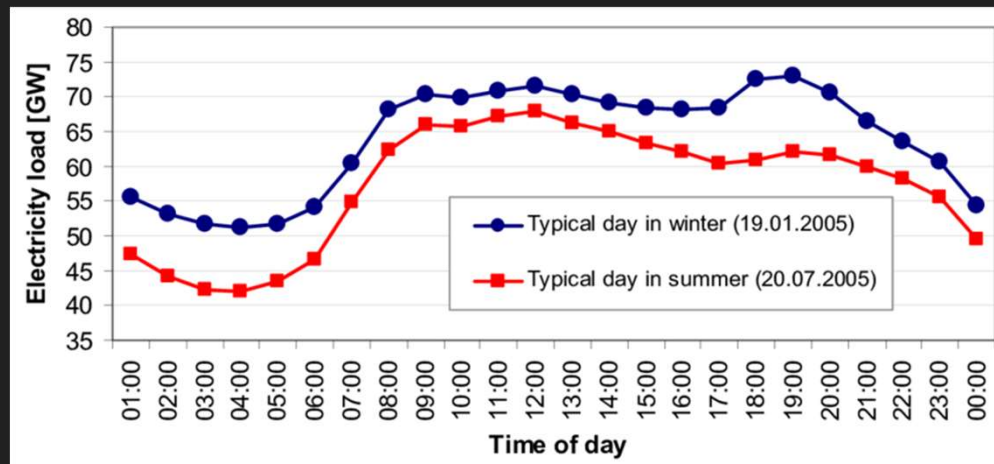
Visión computarizada



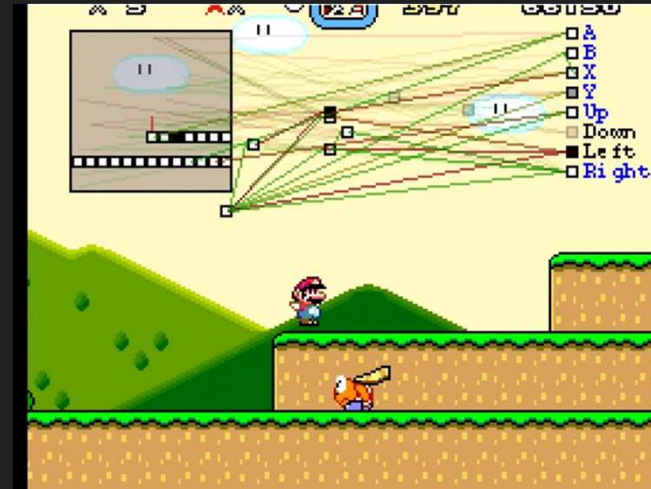
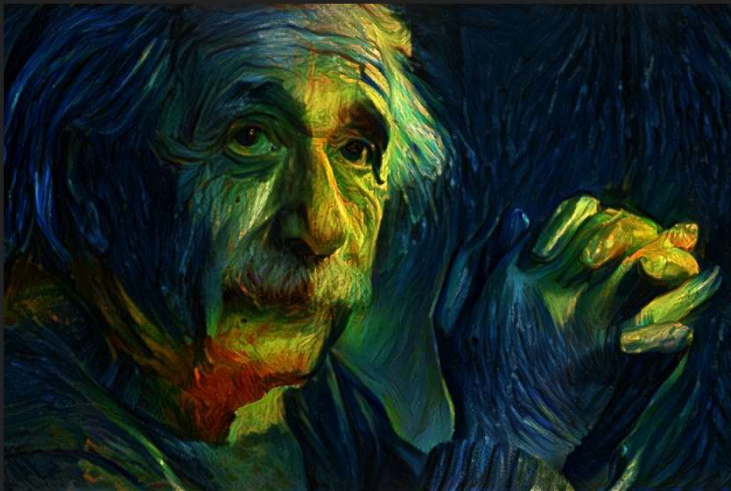
Detección de Anomalías



Series de tiempo



Modelos generativos



Hello, my name is Darian Mache, I am 14 years old and live in Millwood, Arkansas.

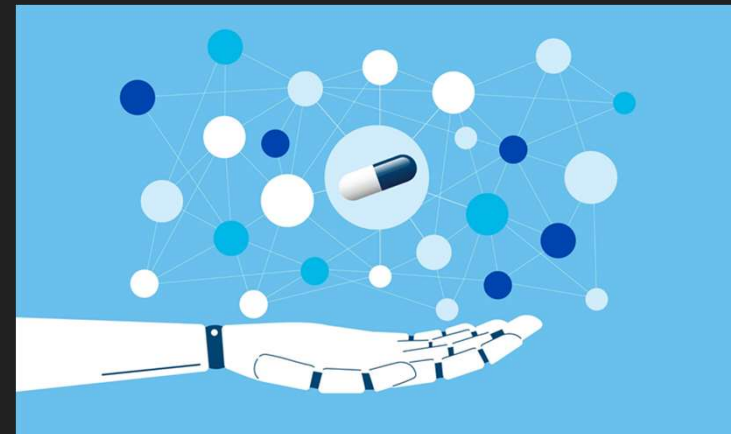
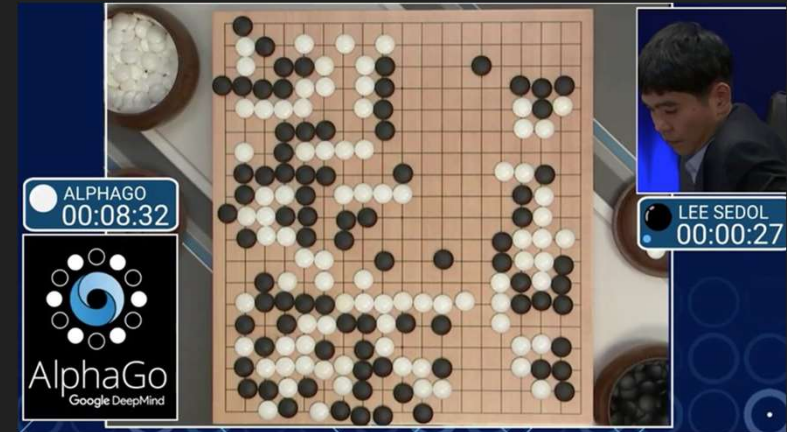
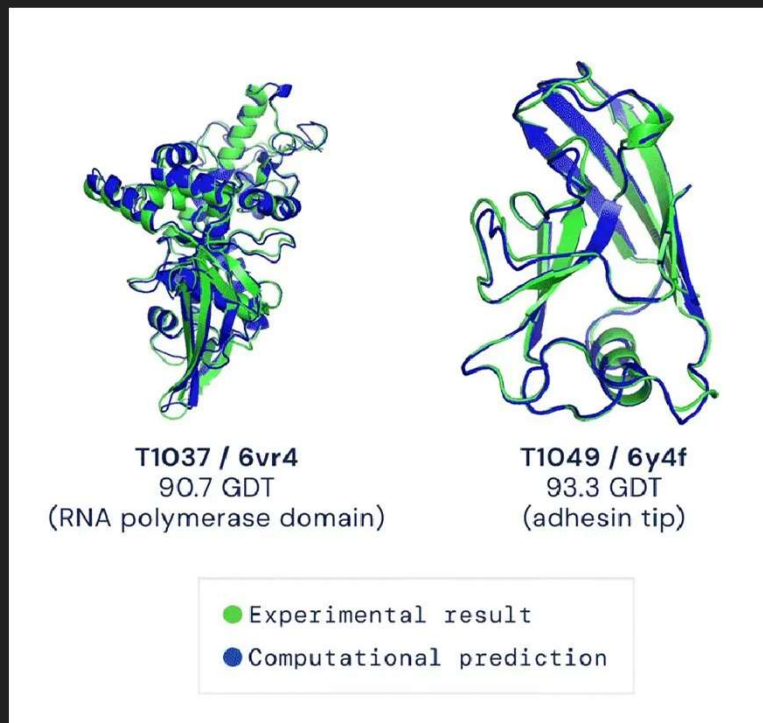
I am just a kid who likes to do things on my own terms.

I play basketball, football, run track, and cheerlead.

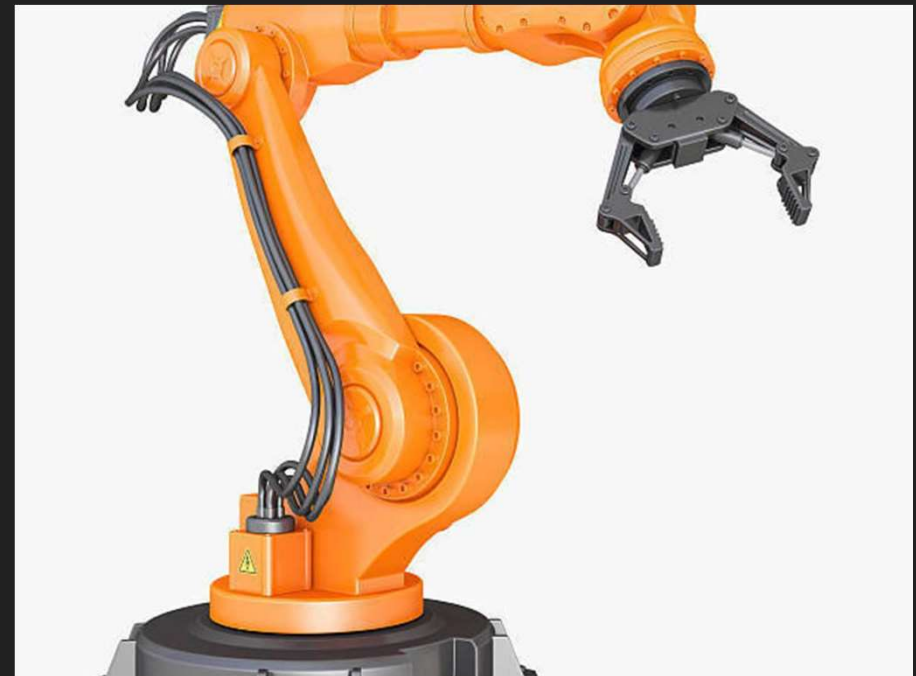
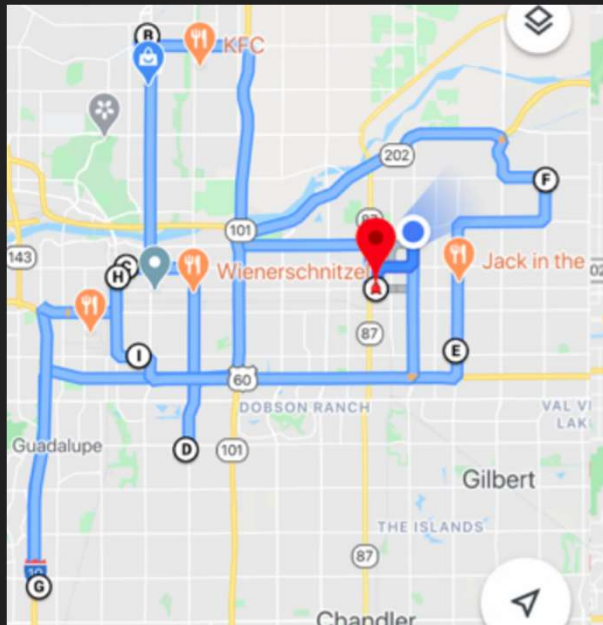
I have no girlfriend, and no

Generate Text

Modelos Generativos 2



Modelos Generativos 3



Teoría

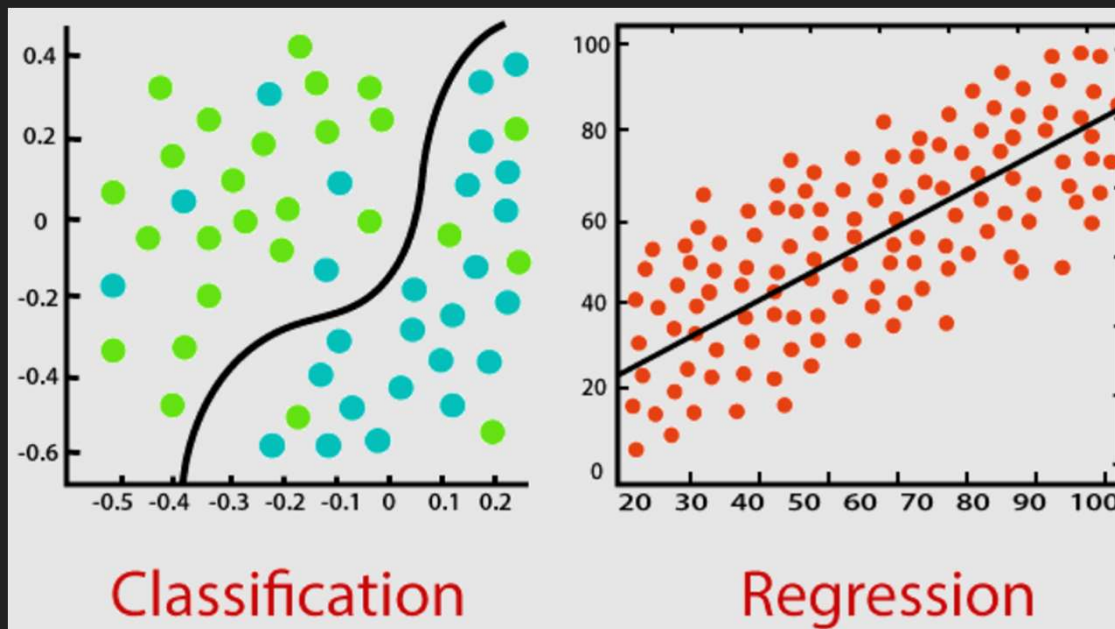
Tipos de Aprendizaje Automático

1. Supervisado/Predicción/Clasificación
2. No-Supervisado/Agrupamiento/Detección de comunidades
3. Detección de anomalías
4. Aprendizaje reforzado (en etapa de adopción)
5. Inferencia Causal (en etapa de exploración)

Extensiones

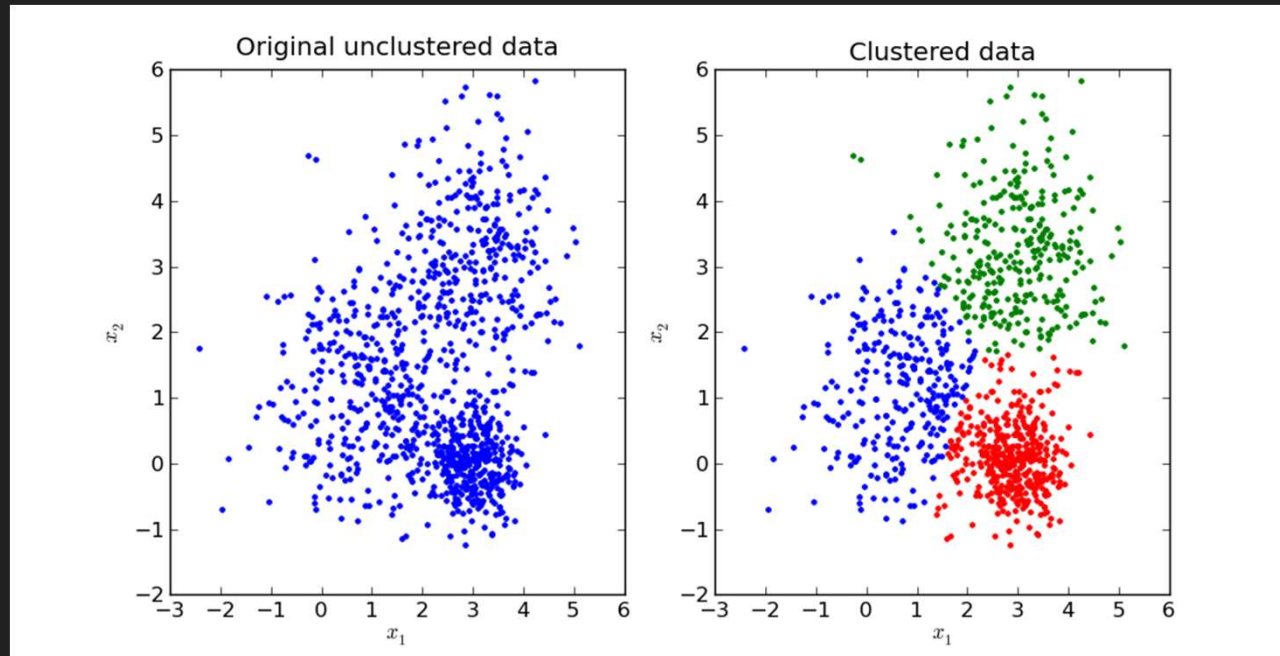
- Generativos/Augmentativos/
- Reducción Dimensional / Importancia de características
- Semi-supervisado/

1. Supervisado/Predicción/Clasificación

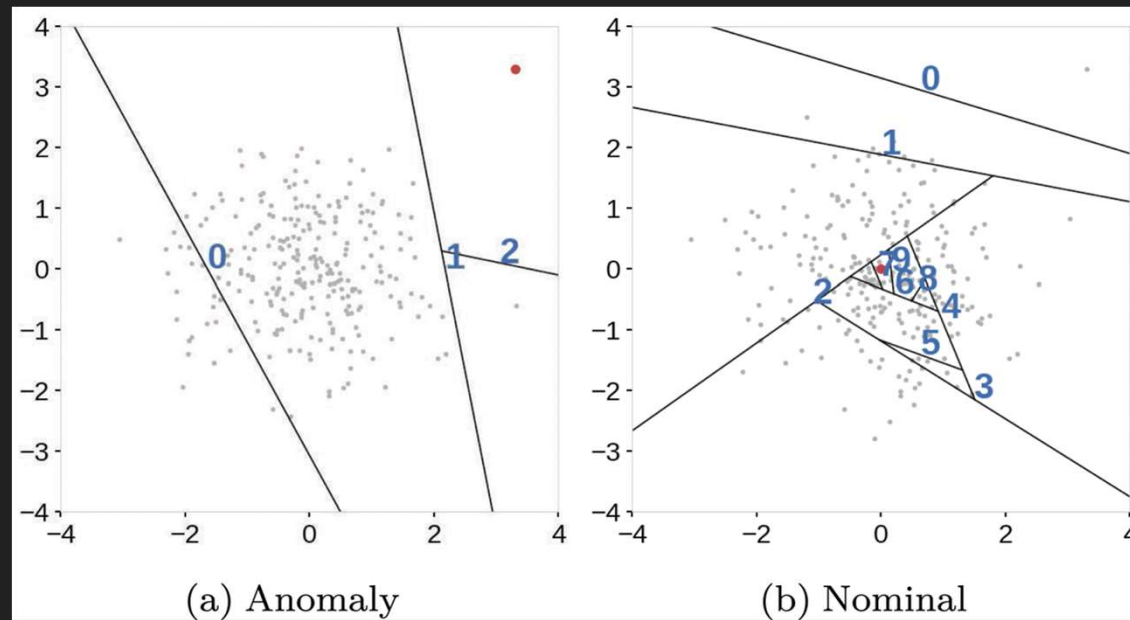


Day	Outlook	Humidity	Wind	Play
D1	Sunny	High	Weak	No
D2	Sunny	High	Strong	No
D3	Overcast	High	Weak	Yes
D4	Rain	High	Weak	Yes
D5	Rain	Normal	Weak	Yes
D6	Rain	Normal	Strong	No

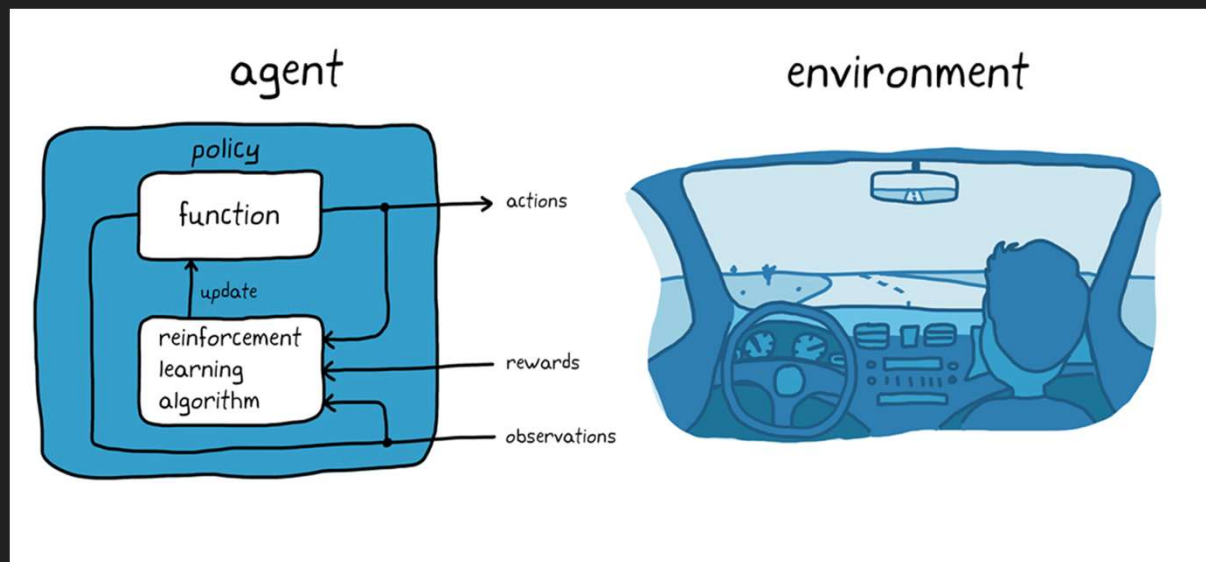
2. No-Supervisado/Agrupamiento/Comunidades



3. Detección de anomalías



4. Aprendizaje Reforzado



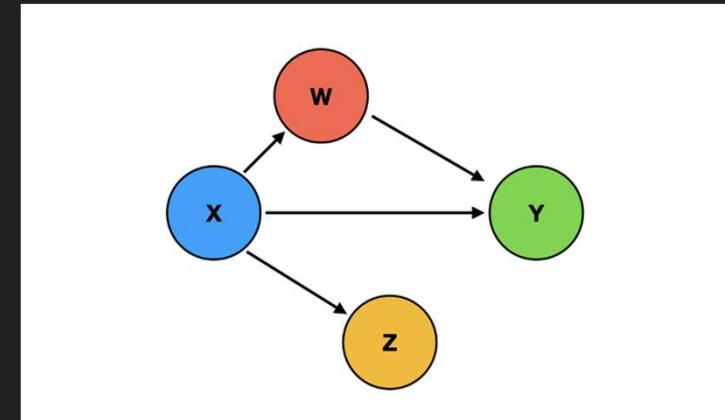
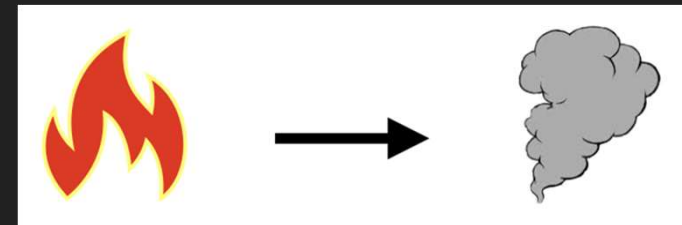
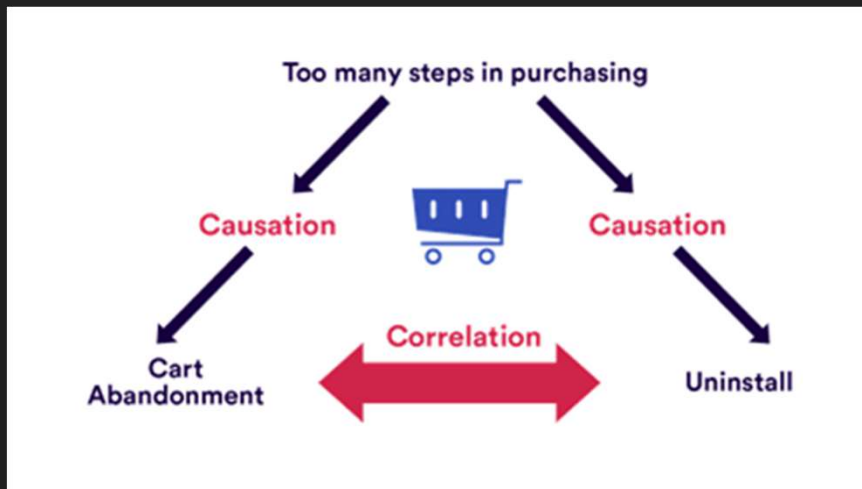
4. Aprendizaje Reforzado (primeras iteraciones)



4. Aprendizaje Reforzado (... algunas iteraciones después)



5. Inferencia Causal



Procedimiento Común

7 pasos

Un procedimiento genérico para resolver distintos tipos de “Machine Learning” :

1. **Determinar Problema:**
 - a. Qué problema queremos solucionar
 - b. Qué queremos aprender de los datos, es decir, nuestro modelo aprendido servirá para:
 - i. predicción, agrupar datos similares, generar contenido, detectar anomalías, detectar parámetros que aportan mayor importancia a un resultado, etc.
2. **Obtener datos (Dataset)**
 - a. csv, grafos, json, base de datos, texto, imagenes, video, audio, etc.
3. **Representar datos (Preprocesamiento)**
 - a. Representar los datos de forma numérica para poder realizar cálculos matemáticos con ellos.
4. **Dividir datos (Data Split)**
 - a. Utilizar el ~80% de datos para entrenamiento, ~10% testeo, y ~10% para validación
5. **Entrenamiento (Aprendizaje)**
 - a. Determinar Algoritmo(s) y proceso(s) de aprendizaje
 - b. Entrenar/Aprender parámetros que maximizan/optimizan el valor del problema determinado en el paso 1
6. **Almacenar Modelo Aprendido**
 - a. Almacenar los parámetros
7. **Utilizar Modelo Aprendido**
 - a. ejemplo: predicción, recomendacion, etc.

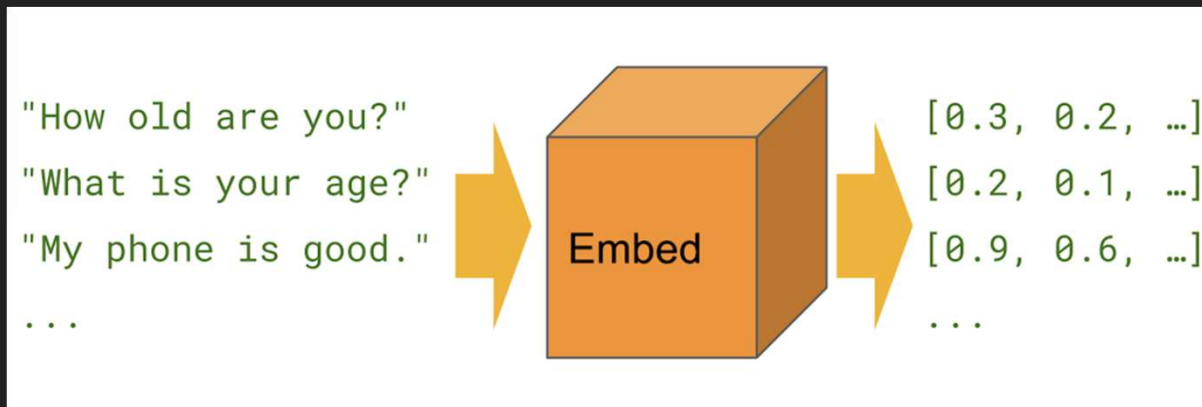
1. Determinar Problema



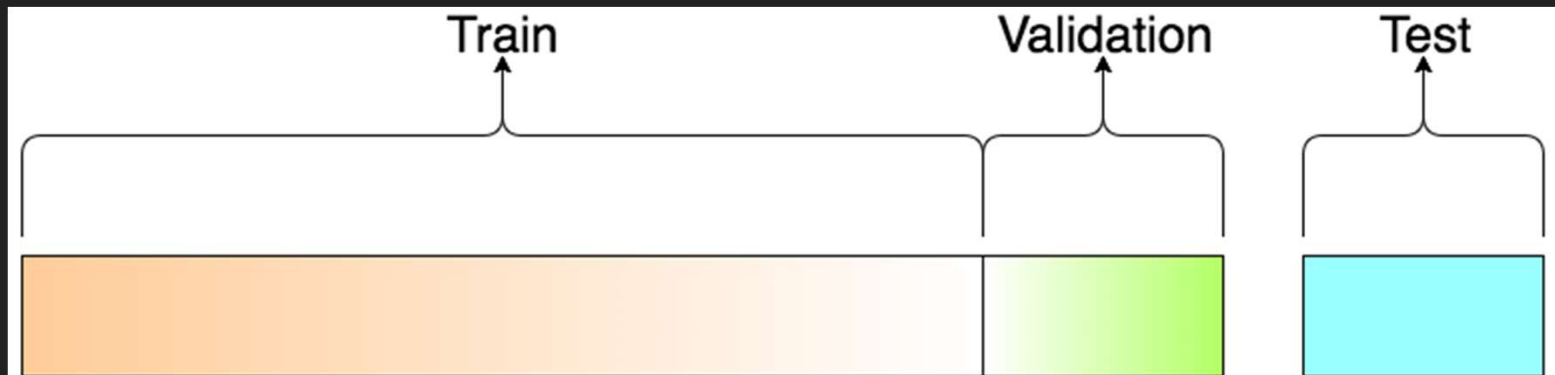
2. Obtener Datos



3. Representar Datos

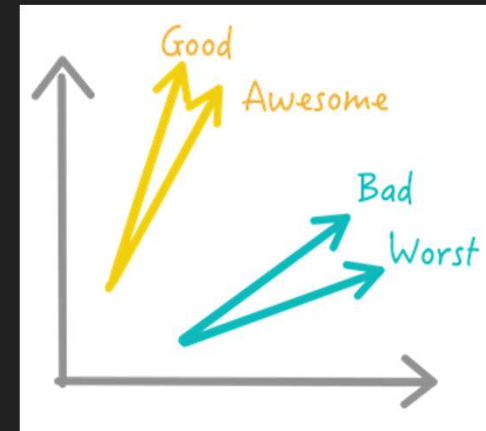
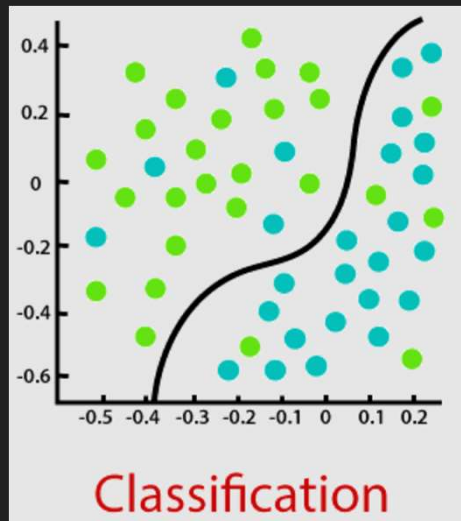


4. Separar Datos

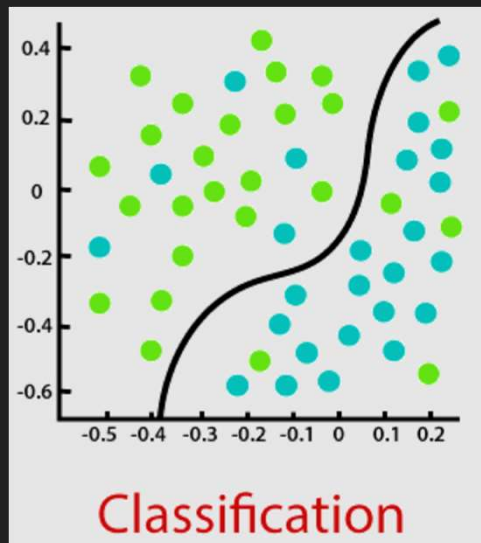


5. Entrenamiento

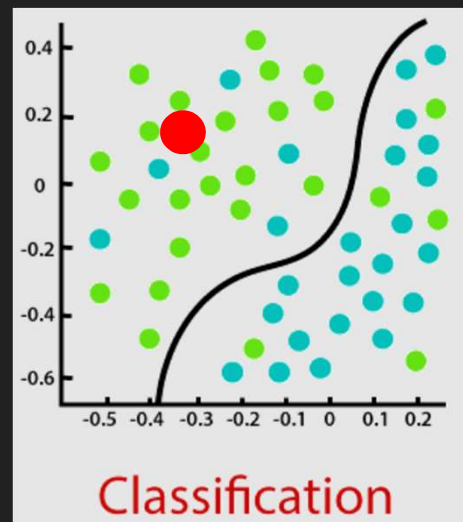
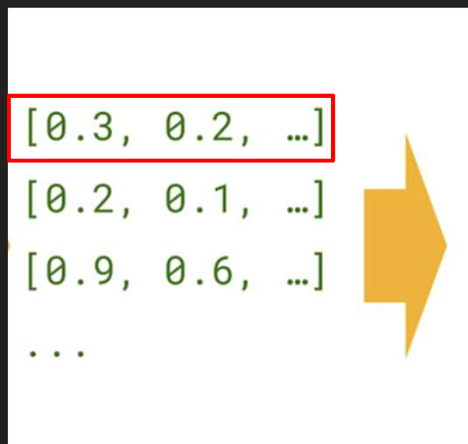
[0.3, 0.2, ...]
[0.2, 0.1, ...]
[0.9, 0.6, ...]
...



6. Almacenar Modelo



7. Predicción



Ejemplo:

Clasificación y Agrupamiento de texto

Aplicación de Procesamiento de Lenguaje Natural

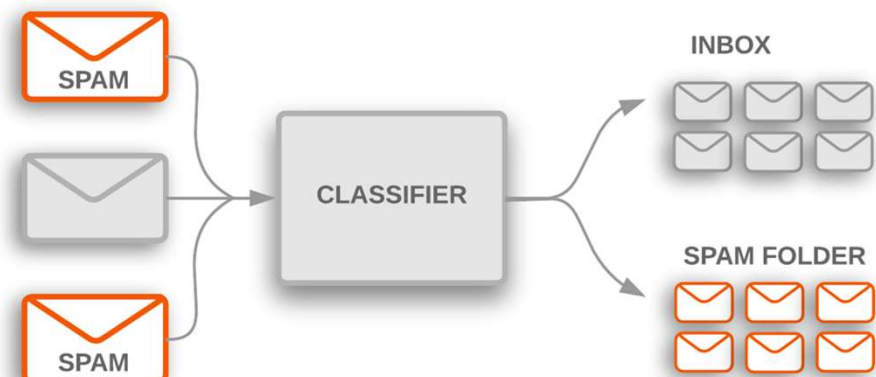
Objetivo:

1. Clasificar correos a distintas bandejas: Deseado, No-deseado, Promociones, Sociales, etc.
2. Encontrar algunas categorías comunes de correos y artículos, para saber como mejor clasificarlos correos/artículos
3. Recomendar contenido de artículos/videos etc..

Parte 1: Clasificación de artículos/emails

Problema: queremos automatizar la clasificación de correos en los siguientes grupos: Deseado, No-Deseado, Promociones, Sociales, Actualizaciones.

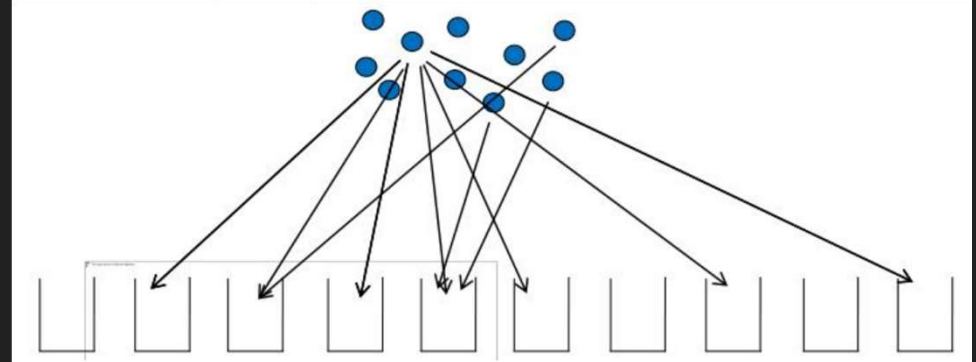
1. Obtenemos un dataset que contenga texto y una asociación a su categoría: deseado, no-deseado, promociones, sociales, etc..
2. Representamos texto en formato de vector
3. Dividimos datos
4. Utilizamos un algoritmo de entrenamiento
5. Almacenar modelo
6. Explotar modelo:
 - a. Cuando llegue un correo nuevo a la bandeja de entrada, pasar el contenido de texto por nuestro modelo de predicción para determinar la categoría correo.



Parte 2: Determinar Categorías de artículos/emails

Supongamos que tenemos el contenido de texto de muchos correos, pero no sabemos cuáles son algunas categorías comunes en las cuales podríamos agruparlos. Queremos encontrar las 5 categorías principales para clasificar el correo

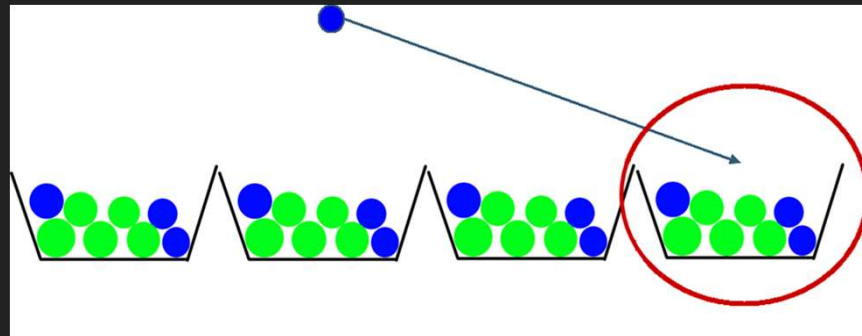
1. Obtener un dataset de texto de correos que no tengan una categoría asignada
2. Abstraer los datos
3. Dividir Datos
4. Entrenar un Modelo no supervisado para “Clustering” o detección de comunidades.
5. Explotar modelo:
 - a. Detectar las los 5 agrupamientos de datos más comunes.



Parte 3: Recomendador de Contenido Básico (artículos)

Otra forma de explotar modelo generado en la **Parte 1**: no-supervisado/agrupamiento/detección de comunidades)

1. Utilizar los agrupamientos de datos para recomendar contenido
 - a. Si a una persona le gusta (consume mayor contenido de) “ciencia y cultura”
 - b. Podemos recomendar otros artículos o contenido que radique en la misma comunidad/cubeta de datos.



Práctica

Procesamiento de Lenguaje Natural

Nota: La práctica la haremos juntos durante la última parte de la sesión.

I. Prerrequisitos

- A. Python 3x instalado
- B. Descargar proyecto: <https://github.com/DarianHarrison/fundamentos-computacion-programacion/tree/main/6.machine-learning>

II. Seguir instrucciones de la práctica

- A. Instalar dependencias
- B. Ir ejecutando el código por bloques para analizar cómo se van procesando los datos
 - 1. Importar dependencias
 - 2. Cargar datos csv al programa
 - 3. Analizar datos
 - 4. Preprocesar Datos
 - 5. Dividir Datos
 - 6. Entrenar
 - 7. Generar una predicción con el modelo entrenado