



RECOMENDACIONES CON MACHINE LEARNING

ESTRUCTURA DE LA CLASE

INTRODUCCIÓN

Presentación del tema de la clase

Introducción al ML

CLASIFICACIÓN

Tipos de clasificación

Aplicaciones

SISTEMAS DE RECOMENDACIÓN

Sistemas basados en contenido

Sistemas colaborativos

Matrices de utilidad

Problema del arranque frío

CLUSTERIZACIÓN

Aprendizaje por clusterización

Vecinos cercanos

ACTIVIDAD PRÁCTICA

ScreenBuddy

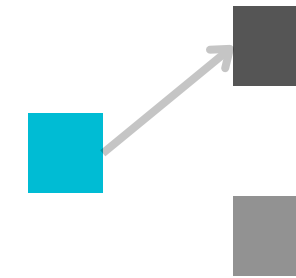
CONCLUSIONES

Recapitulación de los puntos clave de la clase

TIPOS DE CLASIFICACIÓN

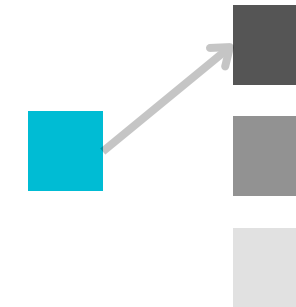
BINARIA

Asignar una instancia a una de las dos clases posibles



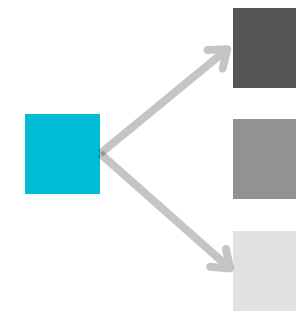
MULTICLASE

Asignar una instancia a una de varias clases posibles



MULTIETIQUETA

Asignar una instancia a una o varias clases posibles.



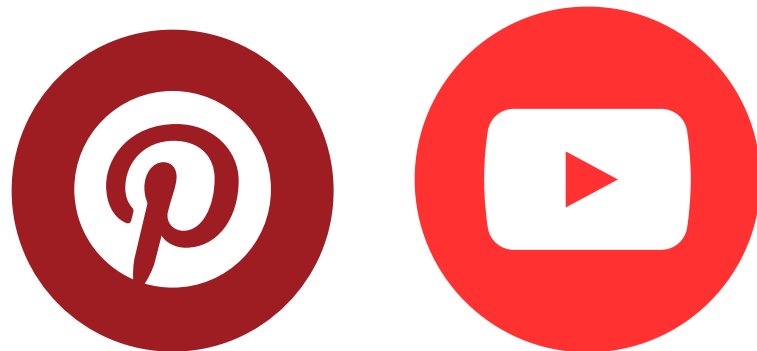
CLASIFICACIÓN Y REGRESIÓN

- **Predecir** el precio de una casa en función de sus características
- **Clasificar** correos electrónicos como spam o no spam
- **Determinar** si una imagen contiene un gato o no
- **Proyectar** el rendimiento de un estudiante en función de sus horas de estudio.
- **Clasificar** opiniones de usuarios en positivas, negativas o neutrales.
- **Predecir** la temperatura máxima diaria en función de datos climáticos.
- **Perfilar** un criminal de acuerdo a sus características.

SISTEMAS DE RECOMENDACIÓN

CONTENIDO

Sugieren elementos similares a los que el usuario ha valorado en el pasado



COLABORATIVO

Sugieren elementos similares a los que otros usuarios con los mismos gustos han valorado en el pasado



MATRICES DE UTILIDAD

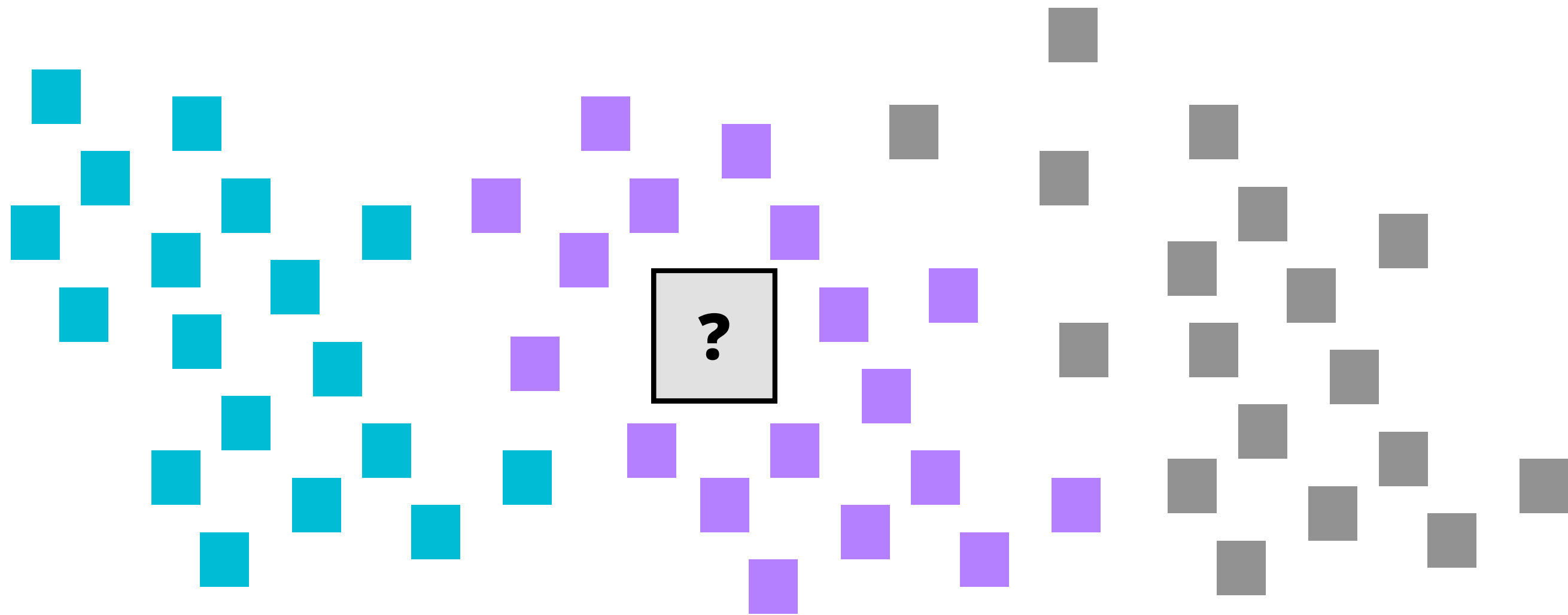
	Post 1	Post 2	Post 3
Usuario 1	Like	Dislike	Like
Usuario 2	Dislike	Dislike	Dislike
Usuario 3	Like	Dislike	Like

En filtrado **colaborativo** se usa la matriz de utilidad (**Usuario - Item**), donde la **interacción** es el valor

PROGRAMA DEL ARRANQUE FRÍO

Un sistema **no puede extraer inferencias** para los usuarios o temas sobre los que aún **no ha reunido suficiente información.**

APRENDIZAJE POR CLUSTERIZACIÓN



VECINOS CERCANOS



```
from sklearn.neighbors import NearestNeighbors

# Entrena el modelo
knn = NearestNeighbors(n_neighbors=numero_vecinos + 1,
algorithm=algoritmo, metric=metrica).fit(X)

# obtiene los vecinos mas cercanos a p
distancias, vecinos = nbrs.kneighbors(p)
```

CONCLUSIONES

CLASIFICACIÓN Y REGRESIÓN

La clasificación permite categorizar las instancias, la regresión permite predecir a partir de una o varias entradas

SISTEMAS DE RECOMENDACIÓN

Recomendaciones por contenido y por filtros colaborativos. Para evitar el arranque en frío usamos sistemas híbridos.

CLUSTERIZACIÓN

Agrupación de elementos por características comunes. Algoritmo de aprendizaje no supervisado.