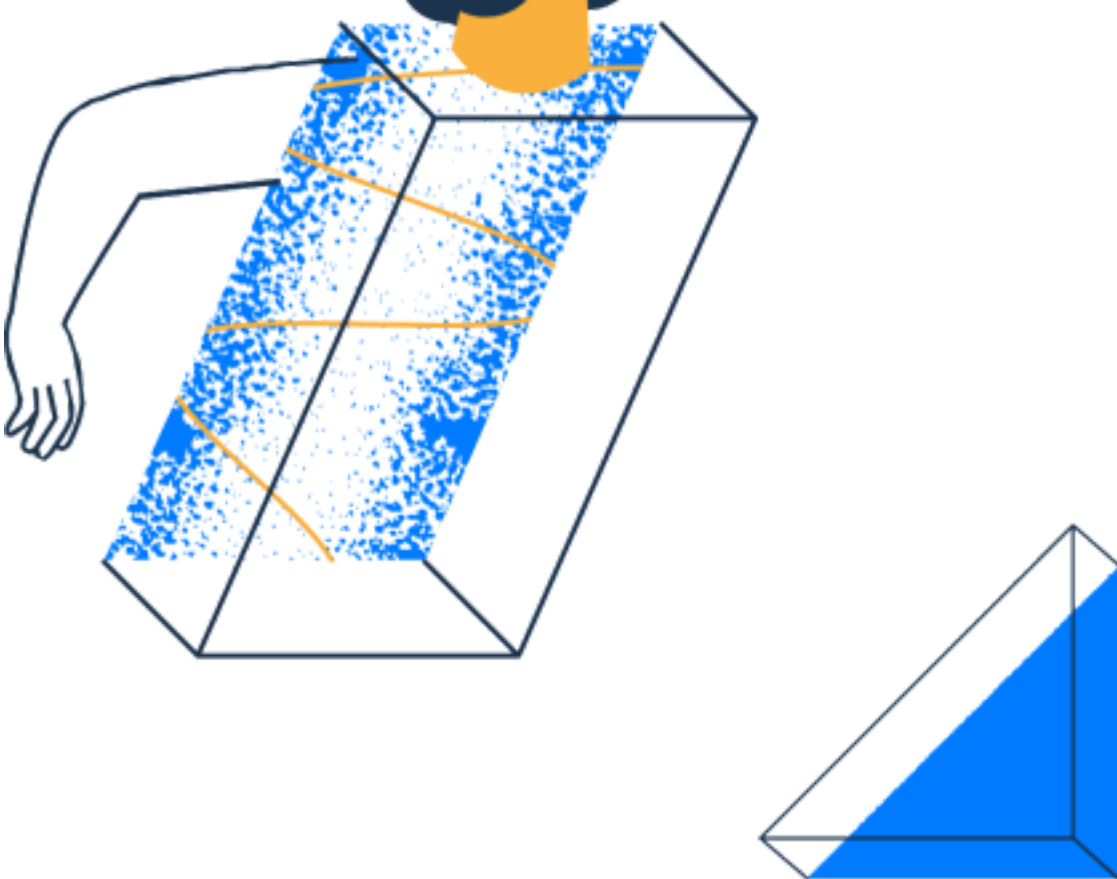


# DATA SCIENCE



Martin Jaureguy  
[martin.jaureguy.95@gmail.com](mailto:martin.jaureguy.95@gmail.com)



# Clase 18 - Agenda







KNN -> K nearest neighbors = K vecinos más



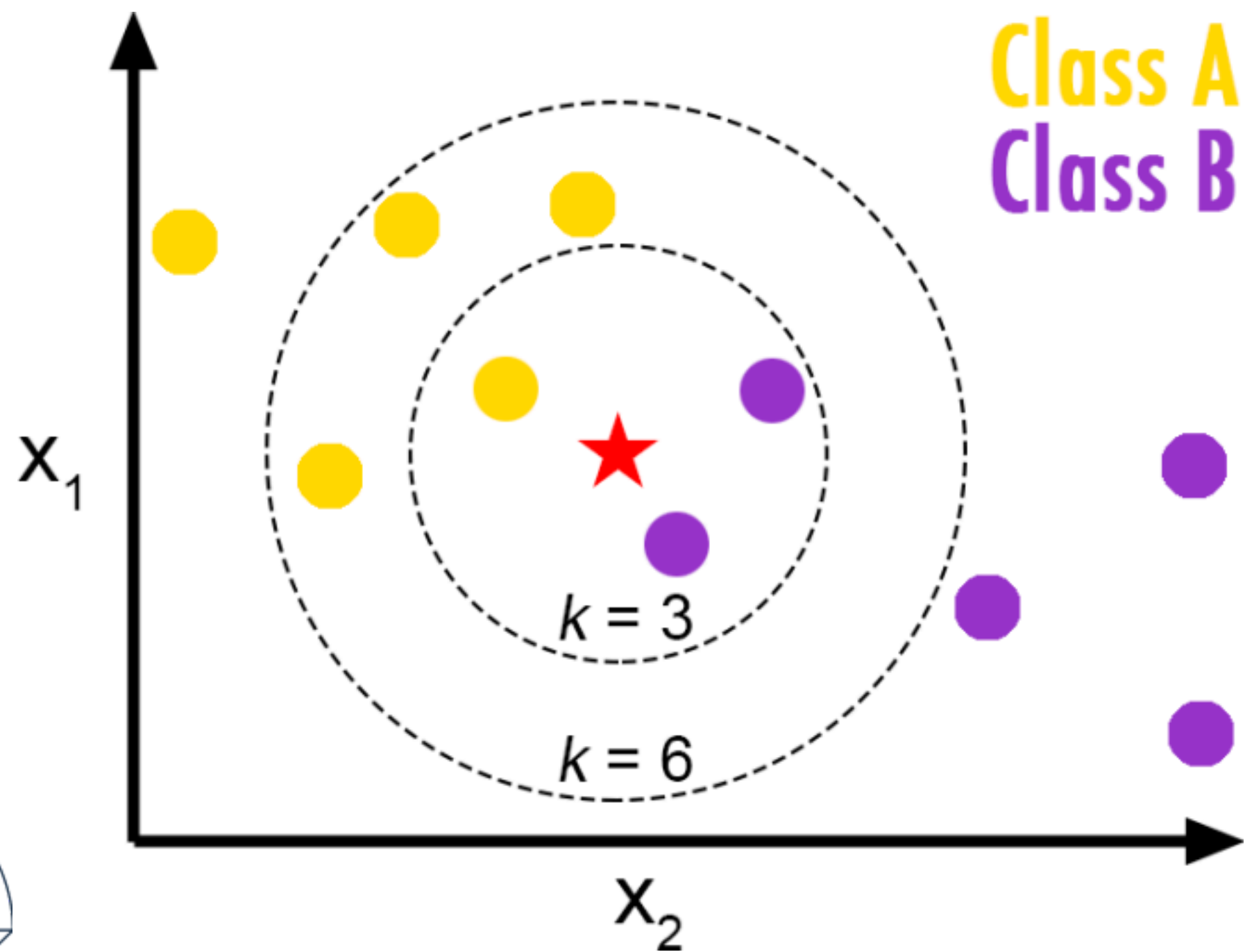
cercanos.

La idea es que dada una nueva instancia la cuál no conocemos a que clase pertenece, la clasifiquemos teniendo en cuenta sus vecinos más cercanos.

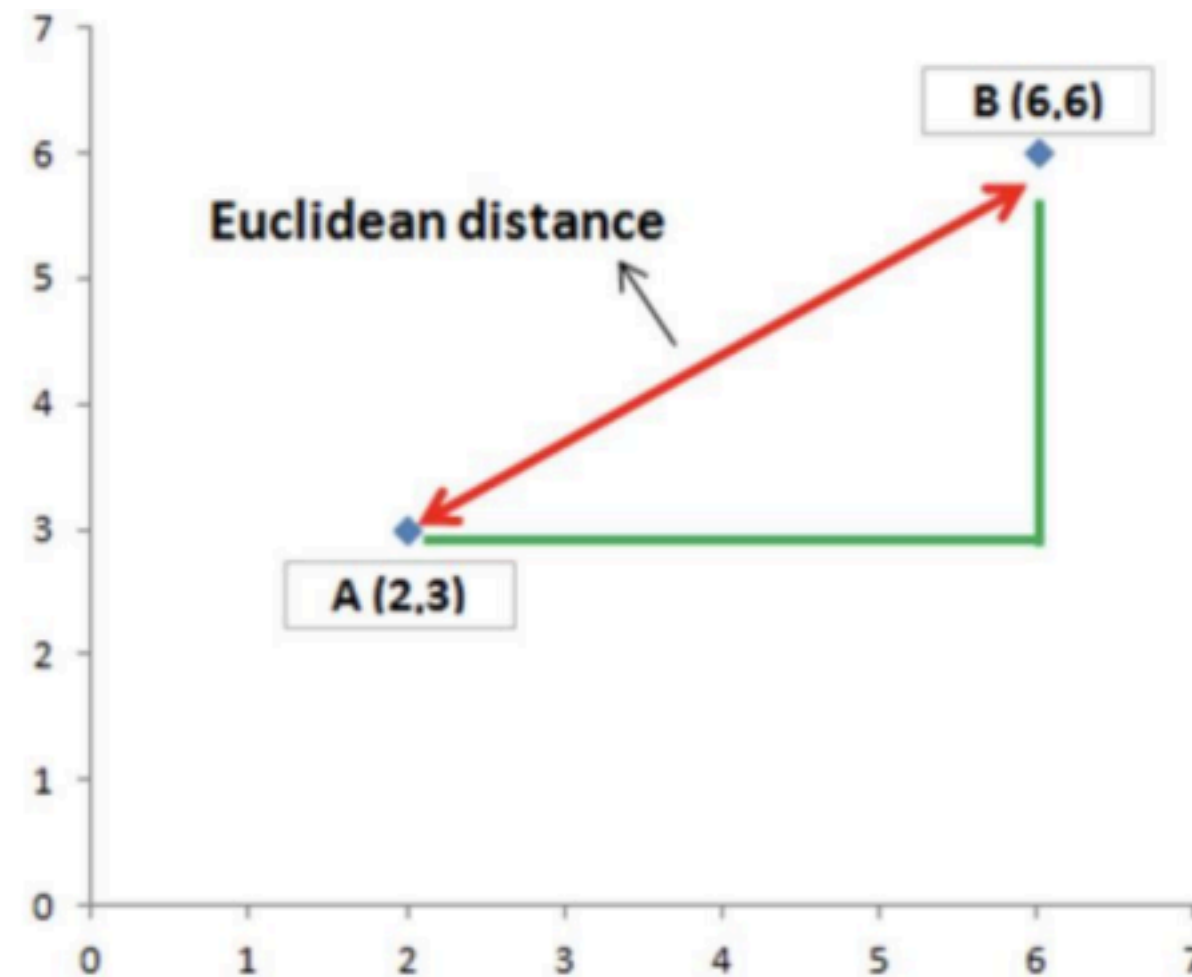
K es el número de vecinos en los que nos vamos a fijar.

Si tomamos  $K=1$  vamos a clasificar simplemente por el vecino más cercano.

**KNN**



# KNN - Distancia



$$\text{Euclidean distance } (a, b) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2}$$

# KNN - Hiperparámetros



¿ Qué son los

hiperparámetros de un modelo ?

¿ Y los parámetros ?

En KNN classifier, 2 de los hiperparámetros más importantes son:





# KNN - El valor de K



No hay una receta para elegir bien el valor de K. Muchas veces se suele hacer a prueba y error.

Si elegimos valores muy chicos de K (por ejemplo 1) corremos peligro de "overfittear" nuestro modelo.

Por el contrario, con valores de K muy grandes, el modelo no va a ser capaz de generar buenas predicciones.



# KNN - Escala de los datos



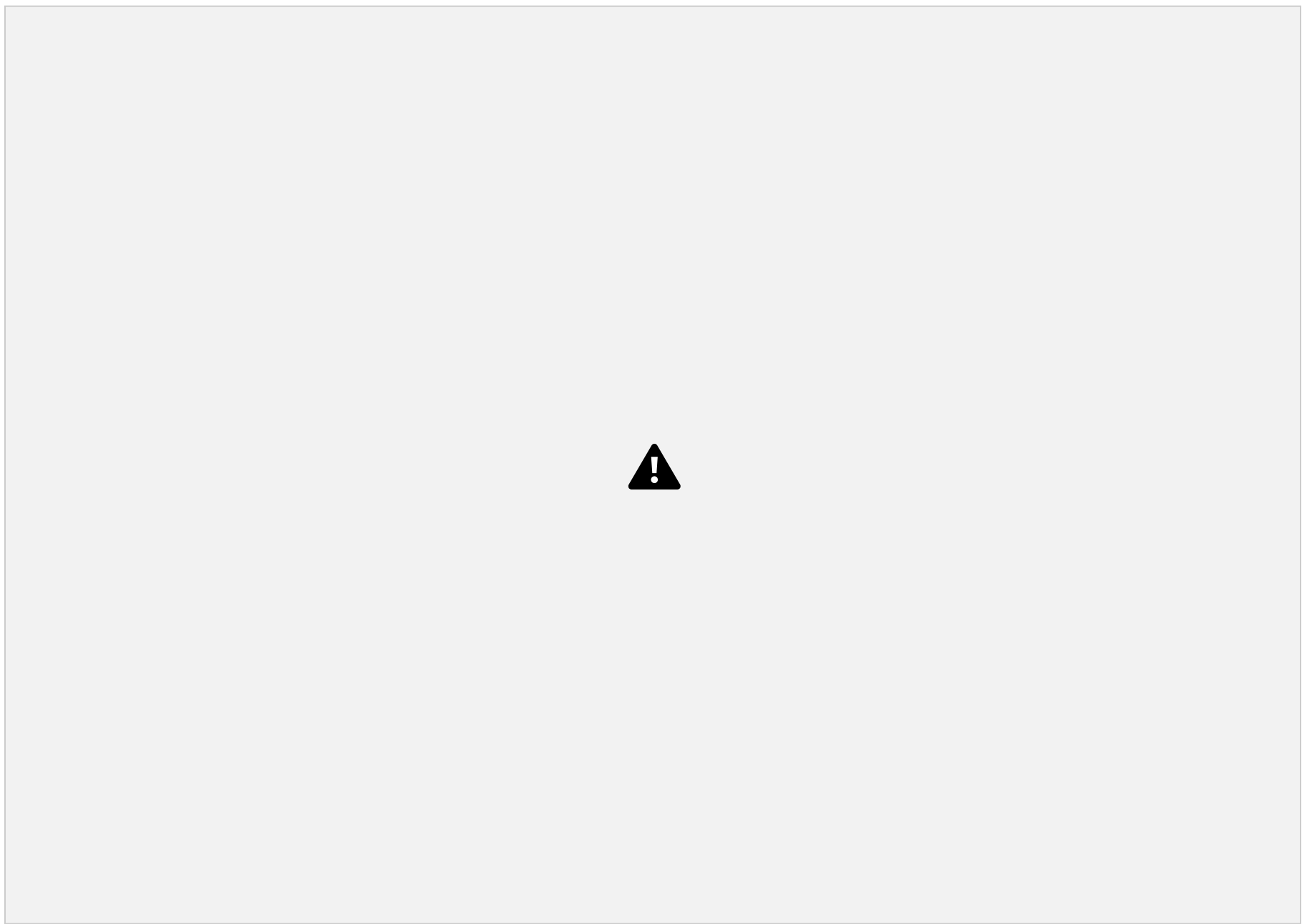
Imaginen que tenemos 2 features de personas: Salario en pesos y edad en años.

Los valores de "Salario en pesos" van a ser números mucho más grandes que los de edad en años.

¿ Ven algún problema con este tipo de features que pueda afectar a el algoritmo de KNN ?

# KNN - Escala de los datos





**KNN - Escala de los datos**



¿ Cómo podemos solucionar esto ?

En KNN (y otros algoritmos que veremos más adelante, como por ejemplo KMeans), es MUY importante que los datos se encuentren en una misma escala.

Esto no quiere decir que no podemos usar las features.

Se soluciona **escalando** las features

# KNN - Escala de los datos



Hay distintas formas de escalar los datos.

Scikitlearn nos provee 2 métodos que son muy utilizados para esto:

<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html>

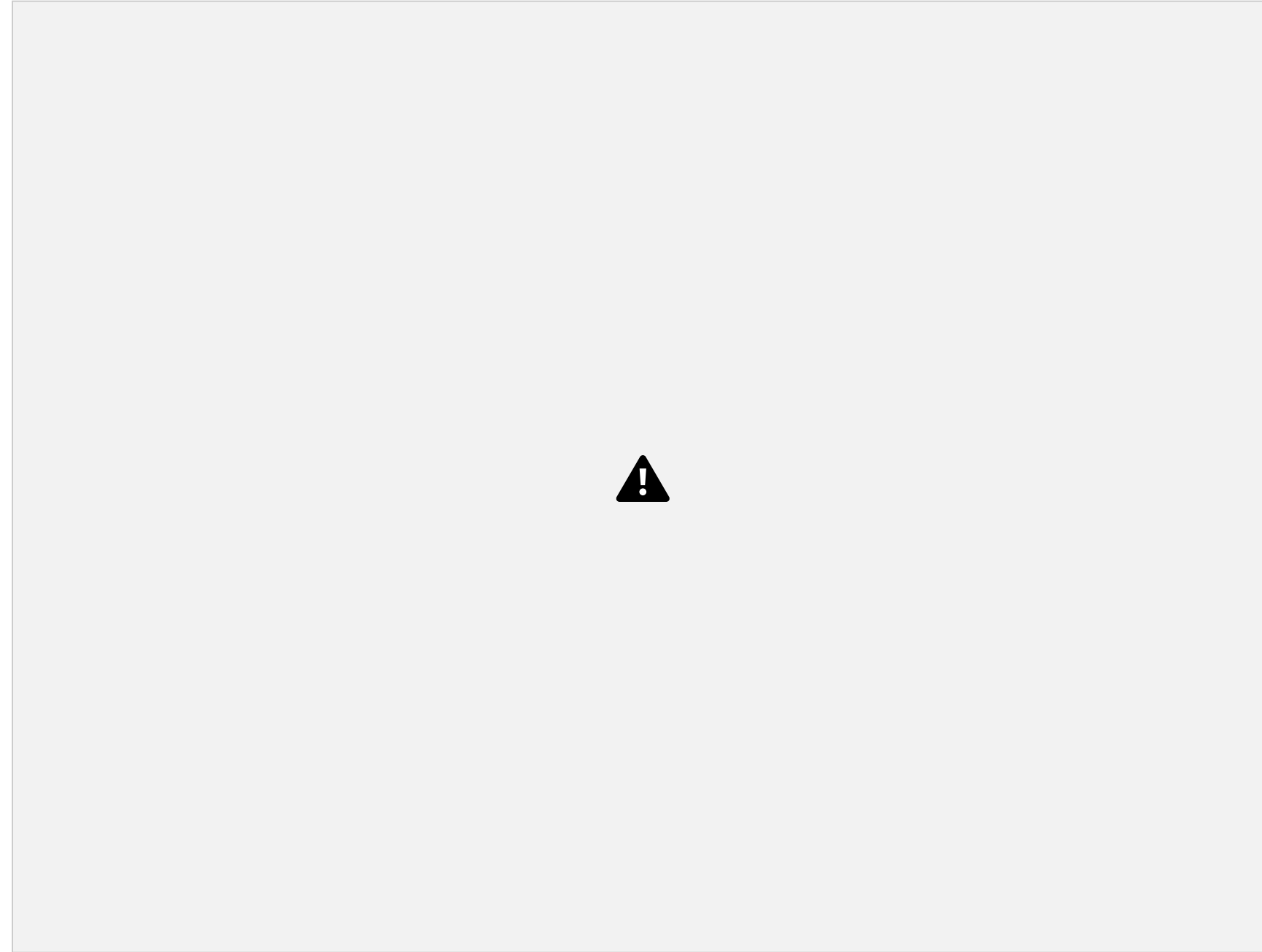


<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>



# KNN - Escala de los datos





# KNN - Pros / Cons



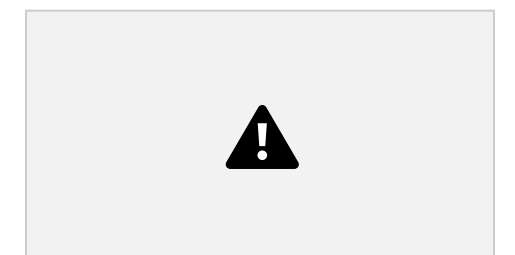
## Ventajas ● fi

- Simple y fácil de interpretar
- Funciona bien en tareas de clasificación, incluso con muchas clases
- Sirve para tareas de clasificación y de regresión
- Entrenamiento rápido

## Desventajas

- Como el modelo necesita tener almacenados todos los puntos, cuando trabajamos con muchos datos esto se vuelve lento y pesado.
- Sensible a outliers, ya que los outliers también "votan"

# KNN - Regresion



Para tareas de regresión, scikit learn también tiene una implementación de KNN.

<https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsRegressor.html>

En este caso, se utiliza la media de los vecinos más cercanos para predecir.