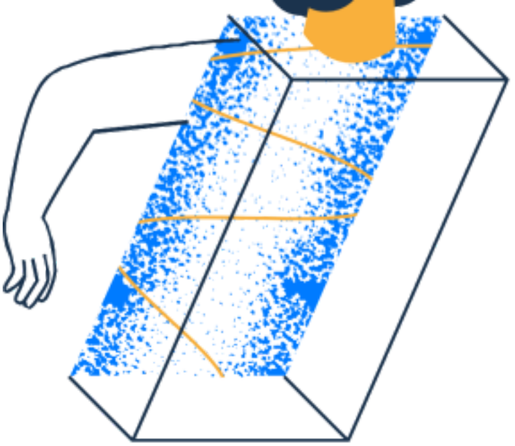
DATA SCIENCE

Martin Jaureguy
martin.jaureguy.95@gmail.com



Clase 23 - Agenda

OPTIMIZACIÓN DE HIPERPARÁMETROS ICARO

2024

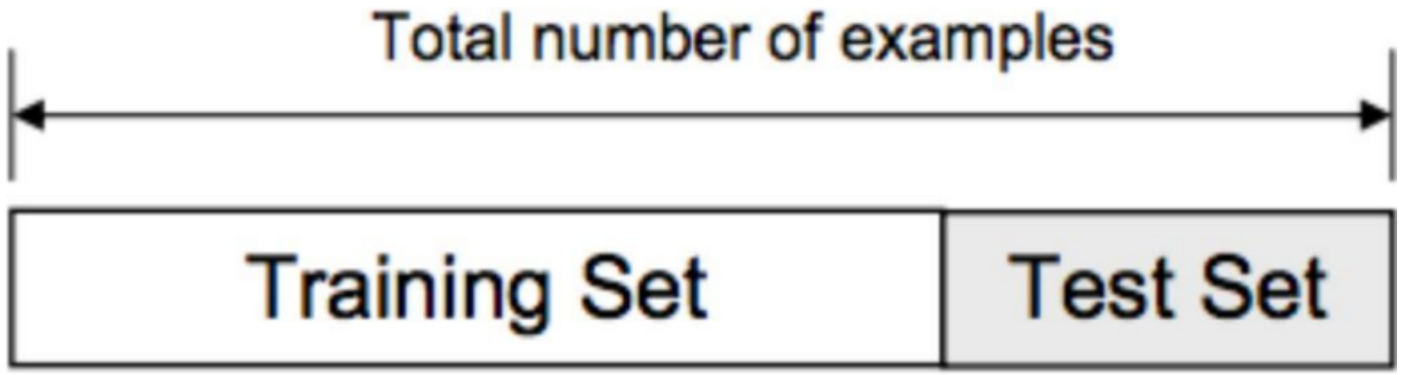
¿ Dudas de la clase pasada ?

¿ Todos pudieron terminar ?

Repaso

Train - test split

¿ Para qué lo hacemos ?

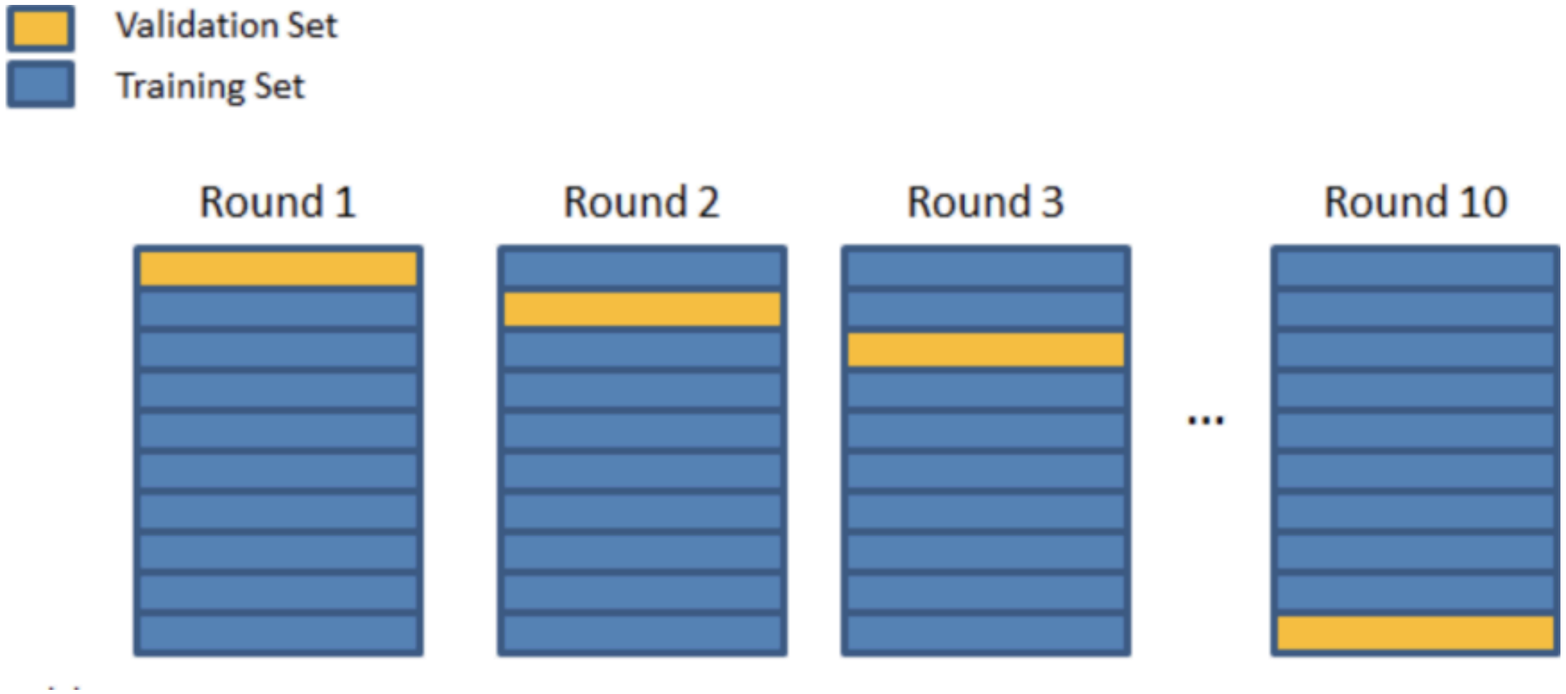


Evaluamos nuestros modelos "simulando" la realidad

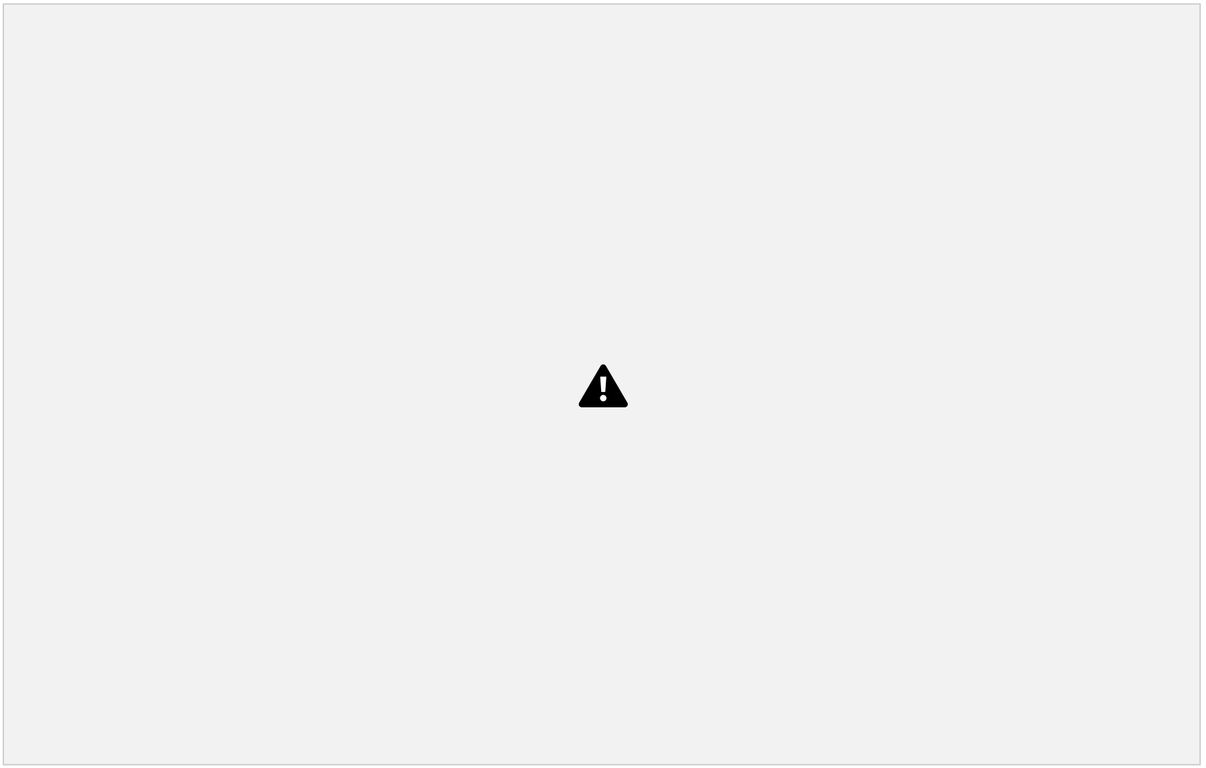
Cross validation - K fold

Este método es conocido

como K-fold cross validation y está implementado en sklearn



Umbral  Obtenemos valores bien clasificados y mal clasificados: TP, FP, TN, FN



Dependiendo de nuestro problema, podemos definir un threshold más bajo o más alto.

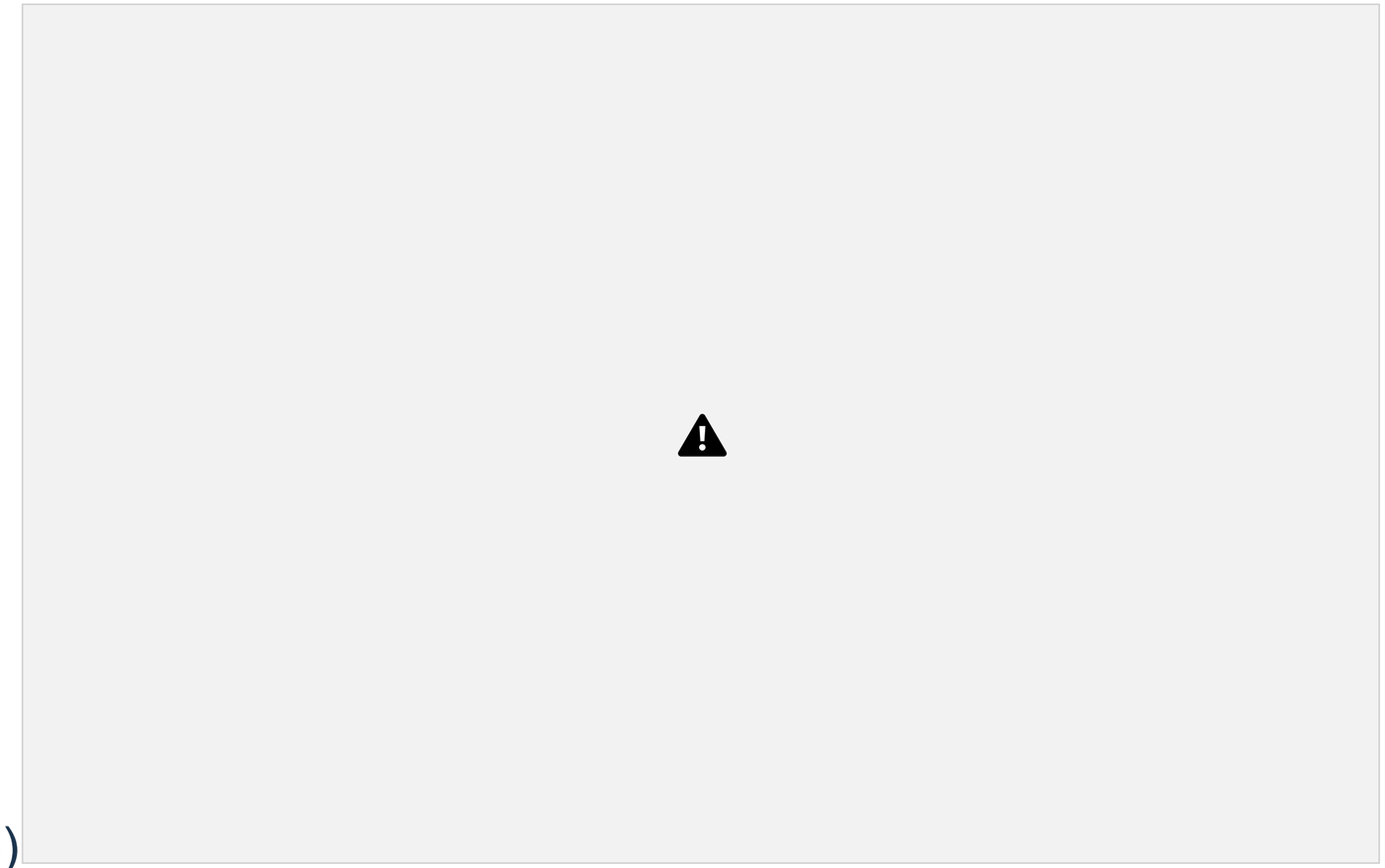
Un threshold más bajo nos permitirá disminuir la cantidad de FN (pero aumentará la cantidad de FP)

Un threshold más alto,nos permitirá disminuir la cantidad de FP aumentando la cantidad de FN.

ROC

Calculando TPR & FPR para distintos umbrales, podemos armar una

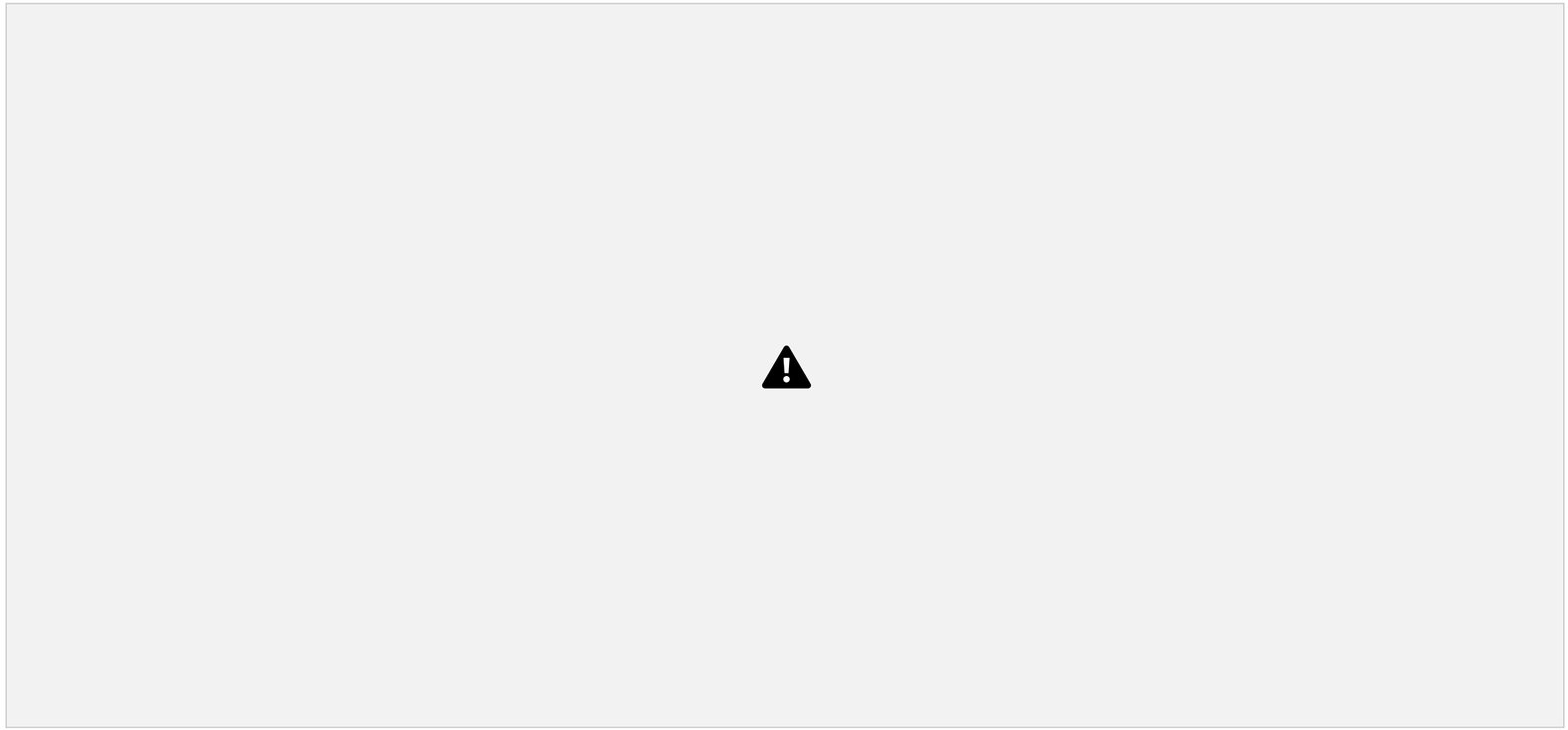
curva (ROC)

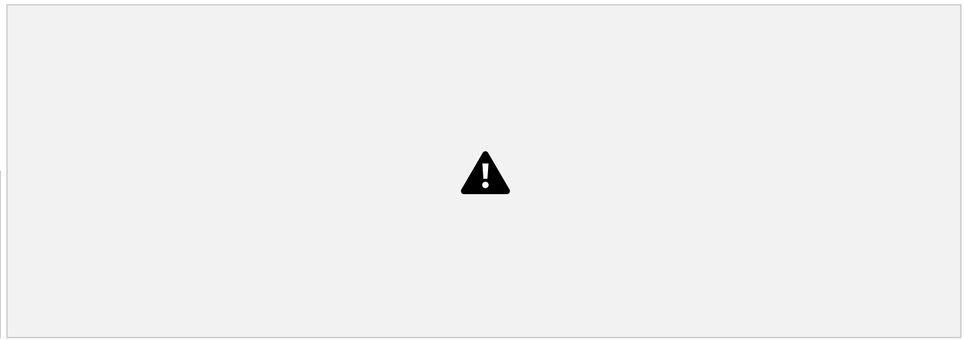


AUC ROC

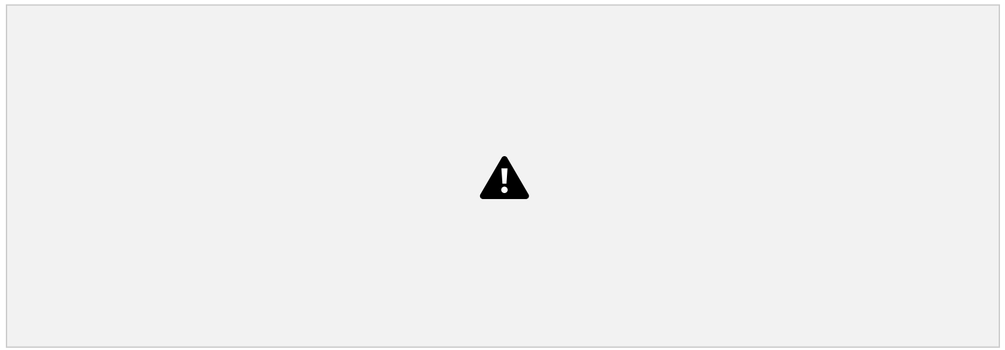
Area bajo la curva ROC.
Para cuantificar que tan bien nuestro modelo separa clases, podemos medir el área bajo la curva

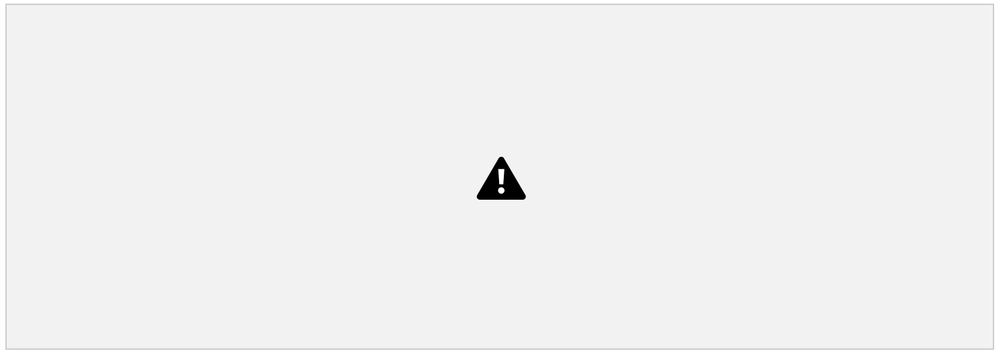
ROC. Mientras más cerca de 1 este, mejor separa las clases nuestro modelo

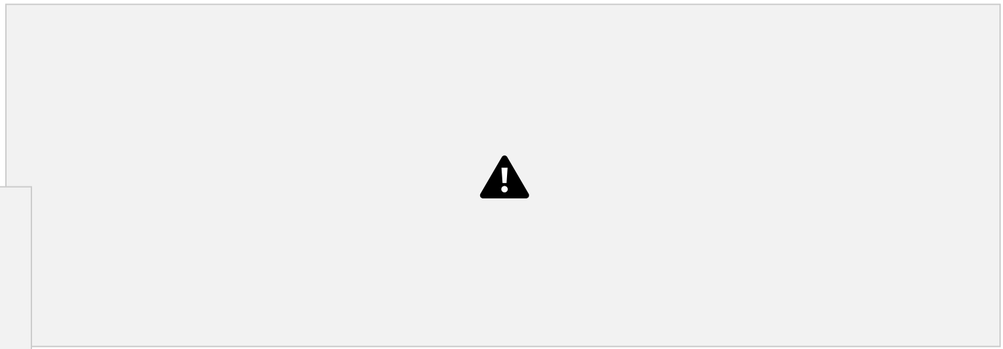


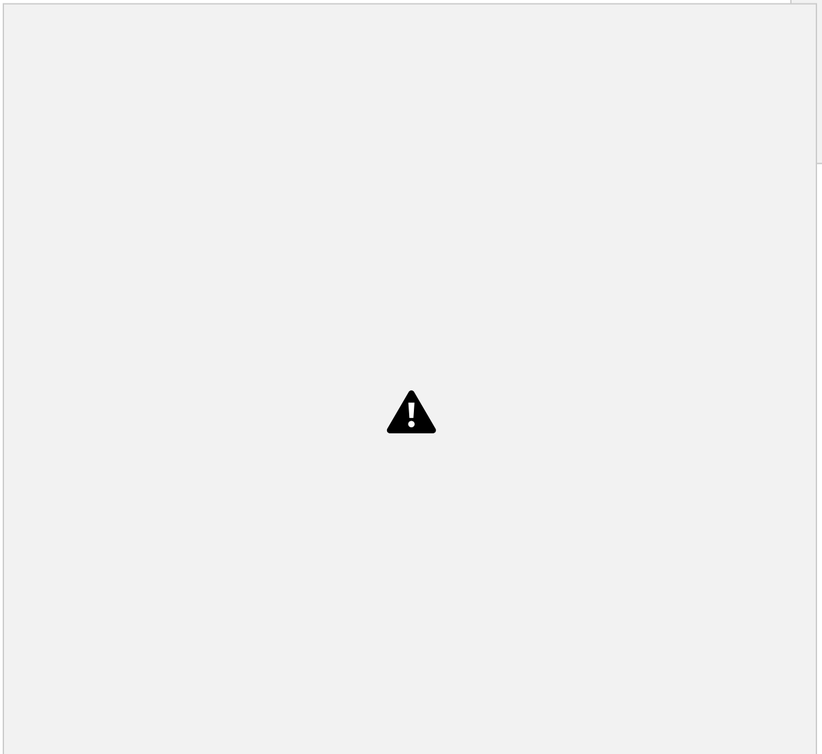


AUC ROC









Optimización de hiperparámetros

Gridsearch

¿ Cuál es la diferencia entre un parámetro / hiperparámetro ?

Selección de hiperparámetros

Los parámetros del modelo se estiman automáticamente a partir de los datos. Es una variable de configuración que es interna al modelo. Ej:los parámetros a y b en una regresión lineal

los hiper parámetros del modelo se establecen manualmente por lo que deberá ser fijado antes del entrenamiento. Es una configuración externa al modelo y cuyo valor no se puede estimar a partir de los datos. Ej: el valor de n_neighbors en knnregressor.

Selección de hiperparámetros

¿ Cómo podemos seleccionar los mejores hiperparámetros para nuestro modelo ?

Primero tenemos que definir **que es "MEJOR"**.

Selección de hiperparámetros

Lo primero que tenemos que definir es la **métrica** que vamos a medir:

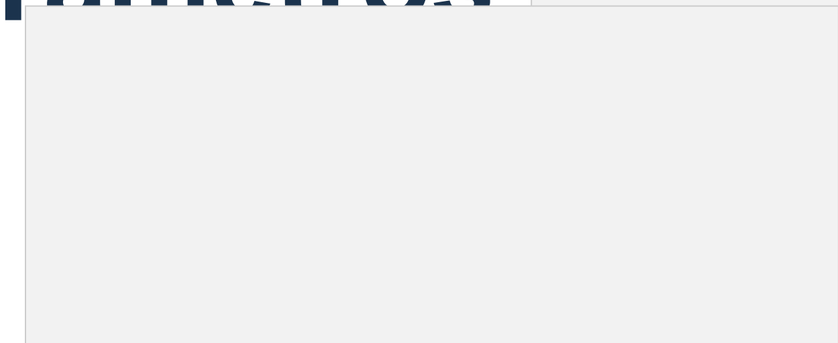
Clasificación:

- Area bajo la curva ROC?
- Accuracy?
- Precision?
- Recall?
- F1Score?

Regresión:

- MAE /MSE ?
- R squared ?

Selección de hiperparámetros

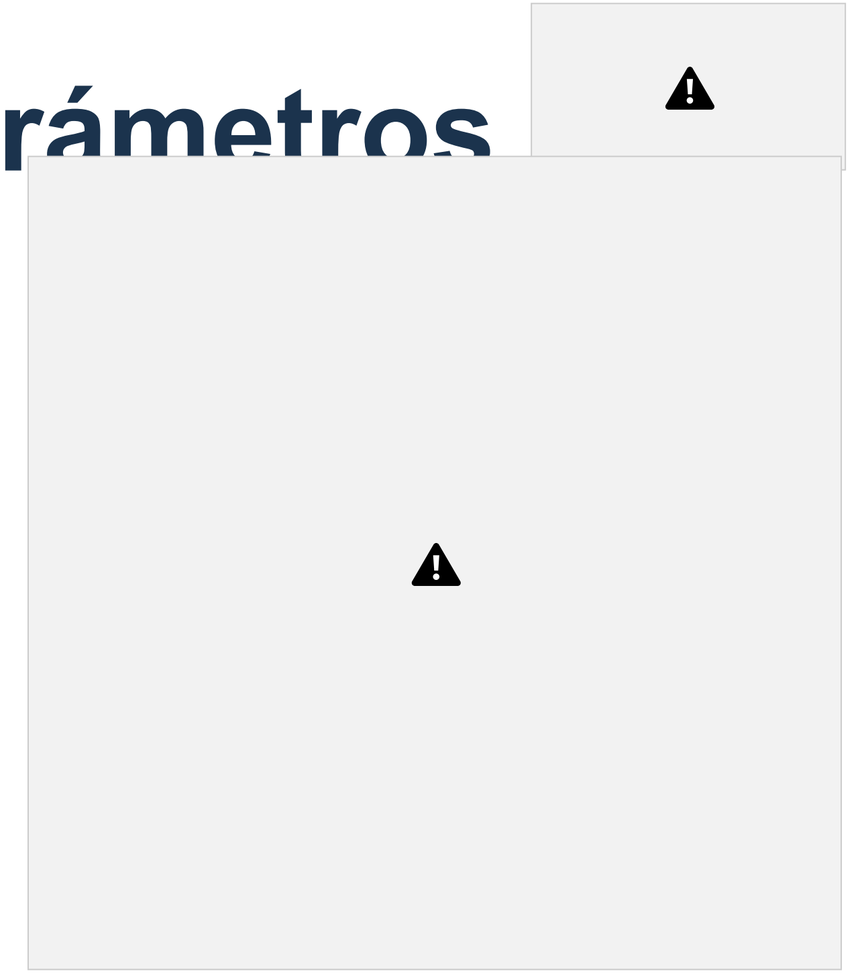


Una vez definida la métrica (por ejemplo AUC

ROC)...

¿ Cómo harían para seleccionar los mejores hiperparámetros ?

Selección de hiperparámetros



¿ Si probamos a mano muchos valores para cada uno de los hiperparámetros ?

Esto en algunos casos puede ser simple (por ejemplo si simplemente estamos ajustando el max_depth de un árbol de decisión)

En casos en los que tenemos muchos

hiperparámetros, un modelo que demora en entrenar, etc, esto se vuelve lento y poco práctico.

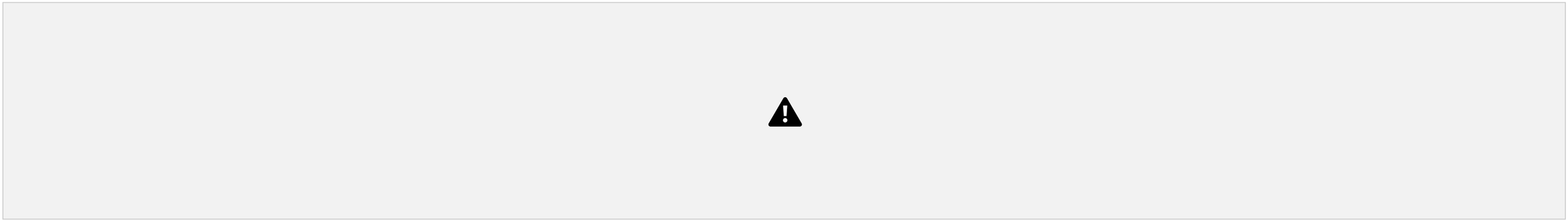
Grid search ⚠

Una técnica que se utiliza para encontrar los mejores hiperparámetros de un modelo es **grid search**

Consiste en una **búsqueda exhaustiva**: prueba automáticamente todas las combinaciones de hiperparámetros que puede y selecciona la que mejor resultados de (de acuerdo a la métrica que elegimos).

Sklearn tiene una implementación de gridsearch: Documentación

Grid search



1. Definimos la métrica que queremos medir
2. Armamos un listado de posibles valores para cada uno de los hiperparámetros -> grilla 3. Se prueban todas las combinaciones de hiperparámetros
4. Se usa la que mejores resultados da

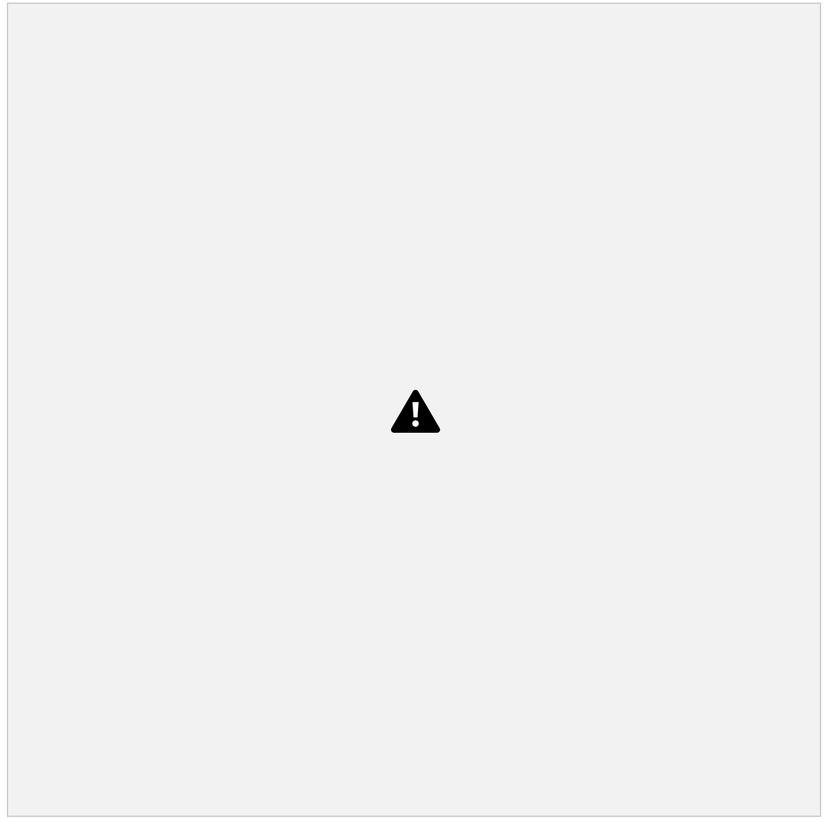
Grid search - Evaluación

Grid search - Evaluación

En el punto 3 decimos que se prueban las combinaciones de hiperparámetros (se evalúa el modelo)...

¿ Cómo lo evaluamos ?

¿ Cómo lo harían ?



¿ Train - Test split ?

¿ Cross validation ?

¿ Cuáles serían las ventajas /desventajas de cada uno ?

Grid search - Evaluación

Al estar entrenando muchos modelos, podría dar la casualidad de que uno de ellos de buenos resultados sobre el conjunto de test por "azar".

Para evitar esto, vimos que podemos utilizar cross validation.

En general, grid search y cross validation se utilizan en conjunto.

¿ Cuál es la principal desventaja de todo esto ?

Grid search

Al entrenar un modelo por cada combinación de hiperparámetros y a su vez evaluar cada uno de estos modelos con cross validation, el proceso se vuelve computacionalmente muy costoso.

A medida que agregamos hiperparámetros, el costo computacional aumenta exponencialmente.

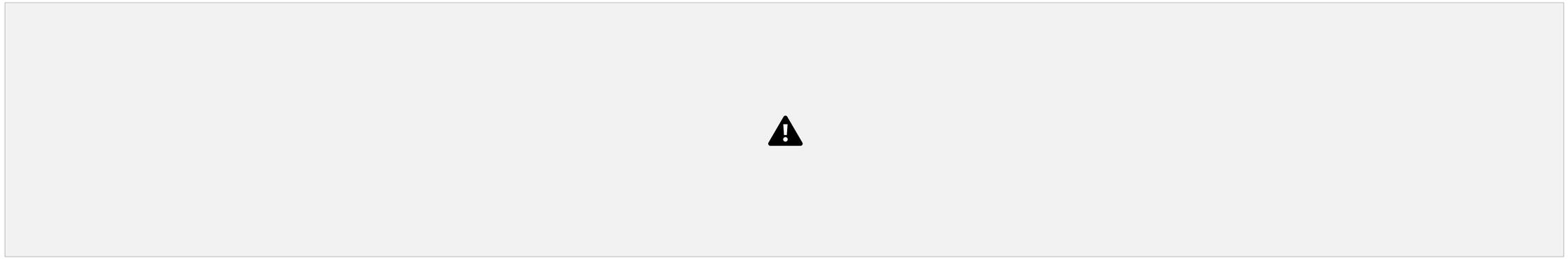
Random search 

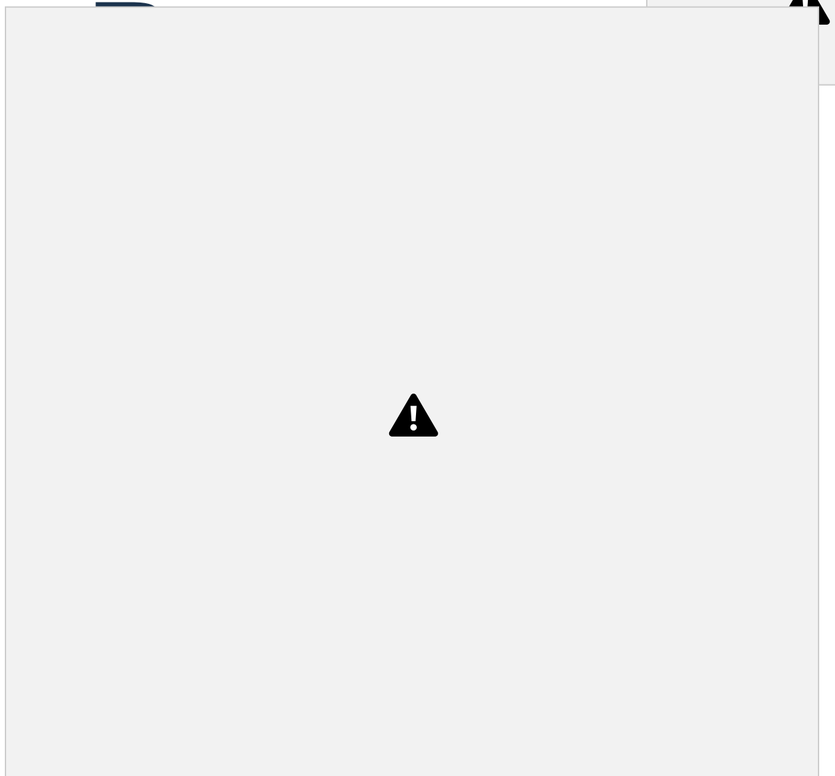
Otro método que se utiliza para buscar los mejores hiperparámetros es random search.

En este caso, en lugar de probar todas las combinaciones de hiperparámetros, se prueban algunas al azar.

Esto suele ser más eficiente y sirve para conseguir buenos resultados.

Documentación sklearn





Entonces, ya sea para Gridsearch o Random search, necesitamos:

- Definir una **métrica**
- Un **modelo** (de regresión o clasificación)
- Un conjunto de **hiperparámetros**

-**Búsqueda** exhaustiva con gridsearch /búsqueda al azar con random search