

DATA SCIENCE



Federico Baiocco
baioccofede@gmail.com
3512075440



Clase 22 - Agenda

ROC

¿ Dudas de la clase
pasada ?

¿ Todos pudieron terminar ?

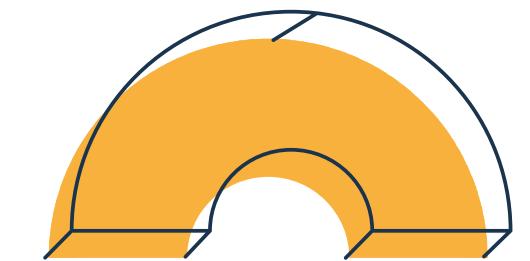




Repaso

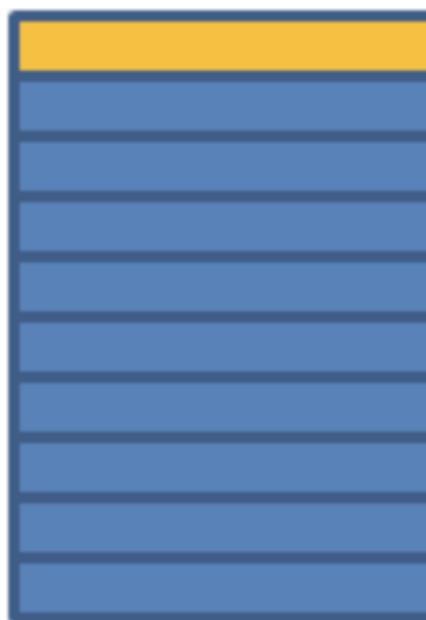
Precision / Recall

Cross validation - K fold

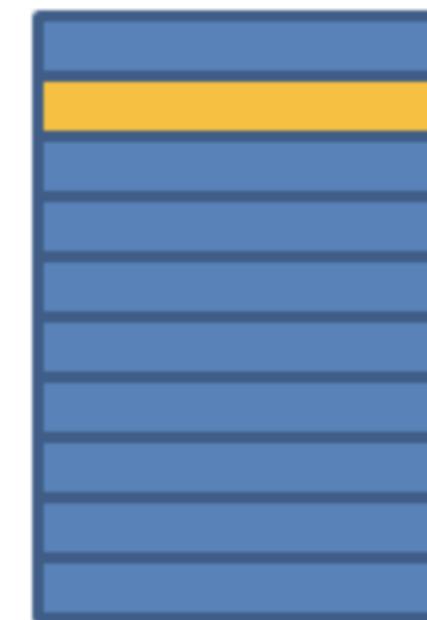


- Validation Set
- Training Set

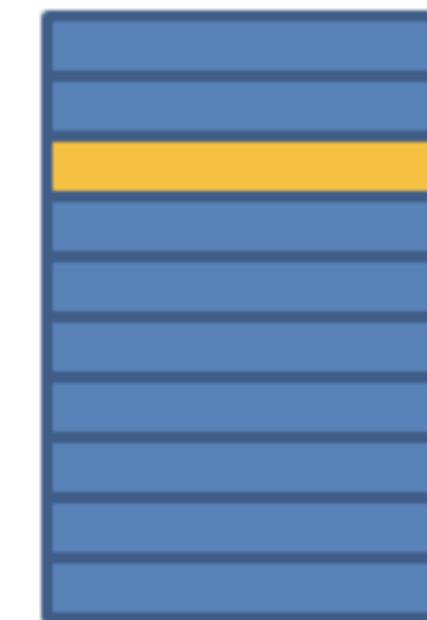
Round 1



Round 2



Round 3

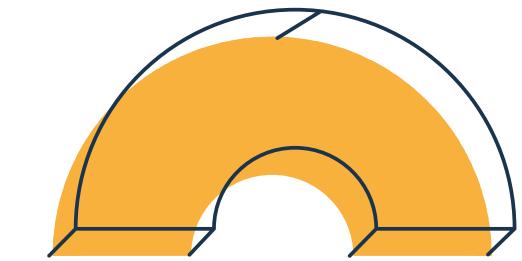


...

Round 10

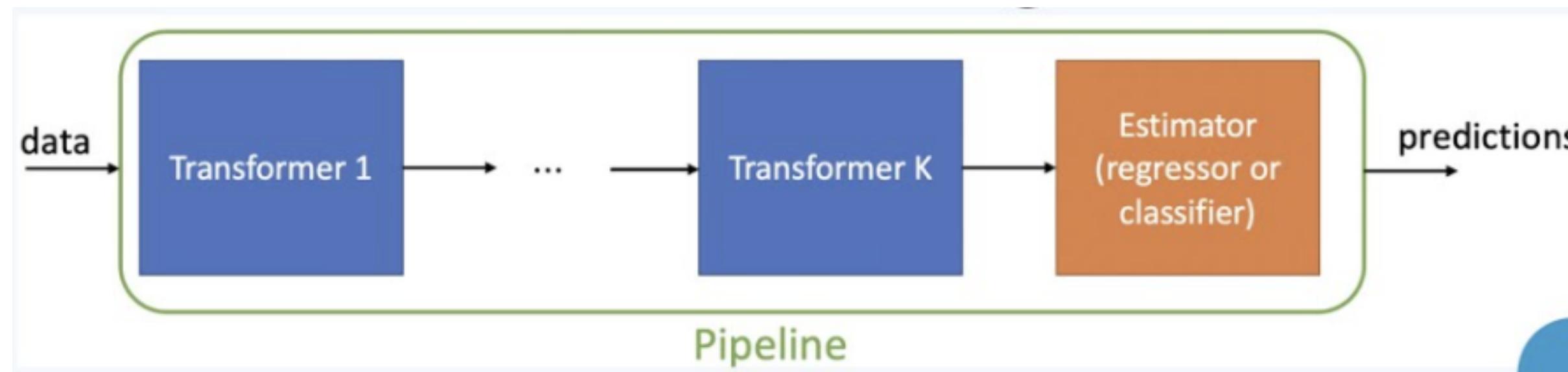


Pipelines



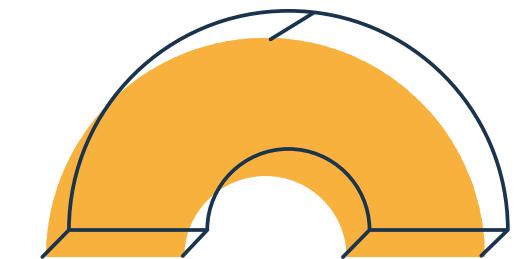
Sklearn nos provee una implementación de "pipeline" que nos permite armar un objeto que se encargue de hacer todo este preprocesamiento y generar las predicciones.

<https://scikit-learn.org/stable/modules/generated/sklearn.pipeline.Pipeline.html>



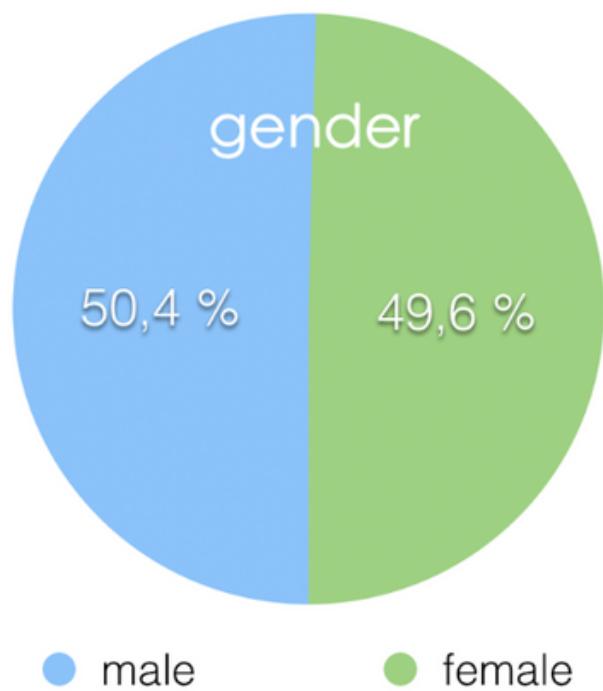
Esto nos va a permitir tener un único objeto al cual hacerle "fit" y "transform" o "predict"

Limitaciones de accuracy score

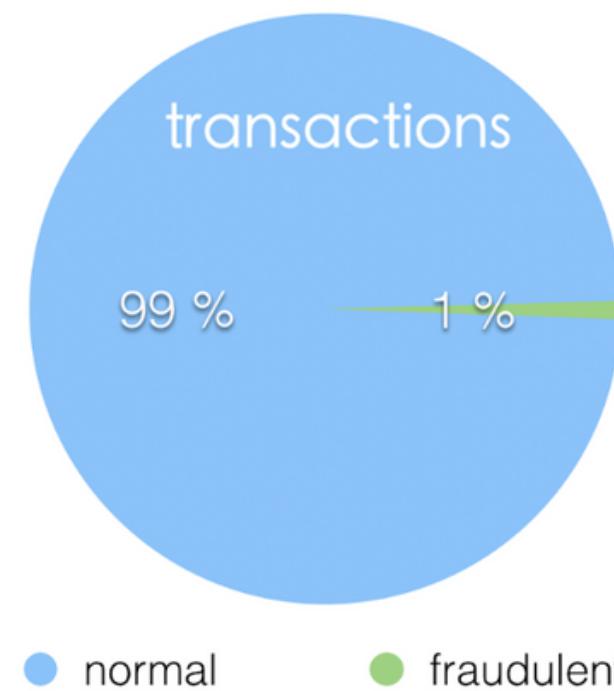


En el caso de la derecha (dataset desbalanceado) si nuestro modelo predice siempre "normal" el accuracy_score es de 0.99, casi perfecto. Sin embargo, el modelo no sirve de nada ya que nunca identifica transacciones fraudulentas.

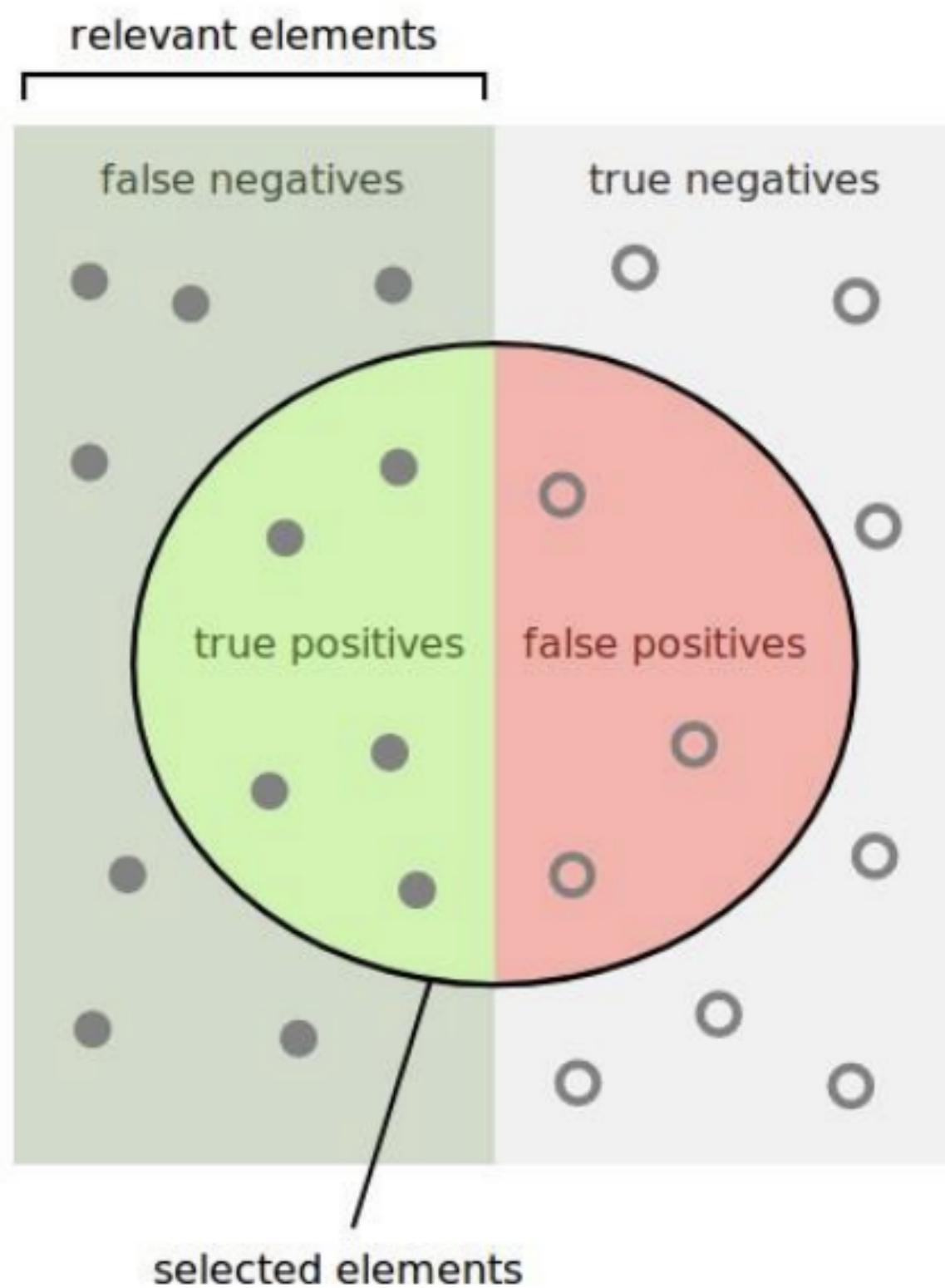
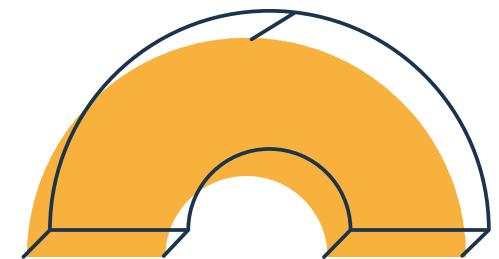
Balanced Dataset



Unbalanced Dataset

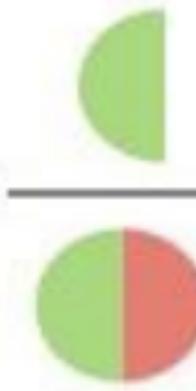


Precision - recall



How many selected items are relevant?

Precision =



How many relevant items are selected?

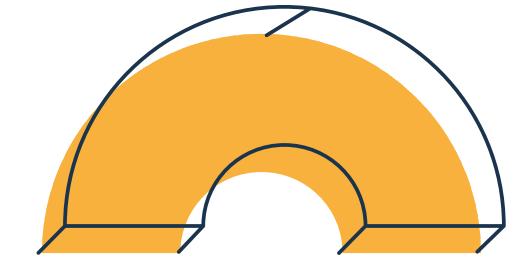
Recall =



$$Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

Precision - recall



Precision: ¿Cuántos de los items que clasificamos como verdaderos, realmente son verdaderos?

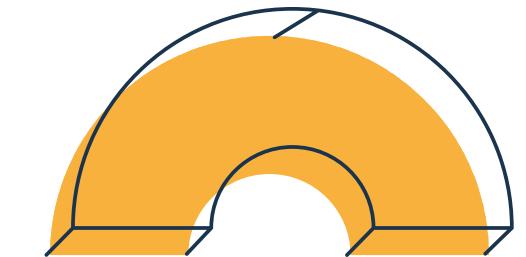
Recall: ¿Cuántos de los items que son verdaderos pudimos identificar?

Problema con estas métricas:

En general no alcanza con medir una sola ya que "compiten" entre sí.

Si decimos que todo es verdadero (todas las personas están embarazadas) nuestro recall va a ser perfecto, pero nuestra precision va a ser muy mala seguramente.

F1 Score



Una métrica que busca un equilibrio entre precision y recall es F1 score:

$$F_1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

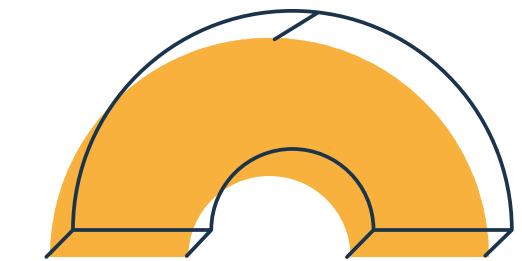


ROC

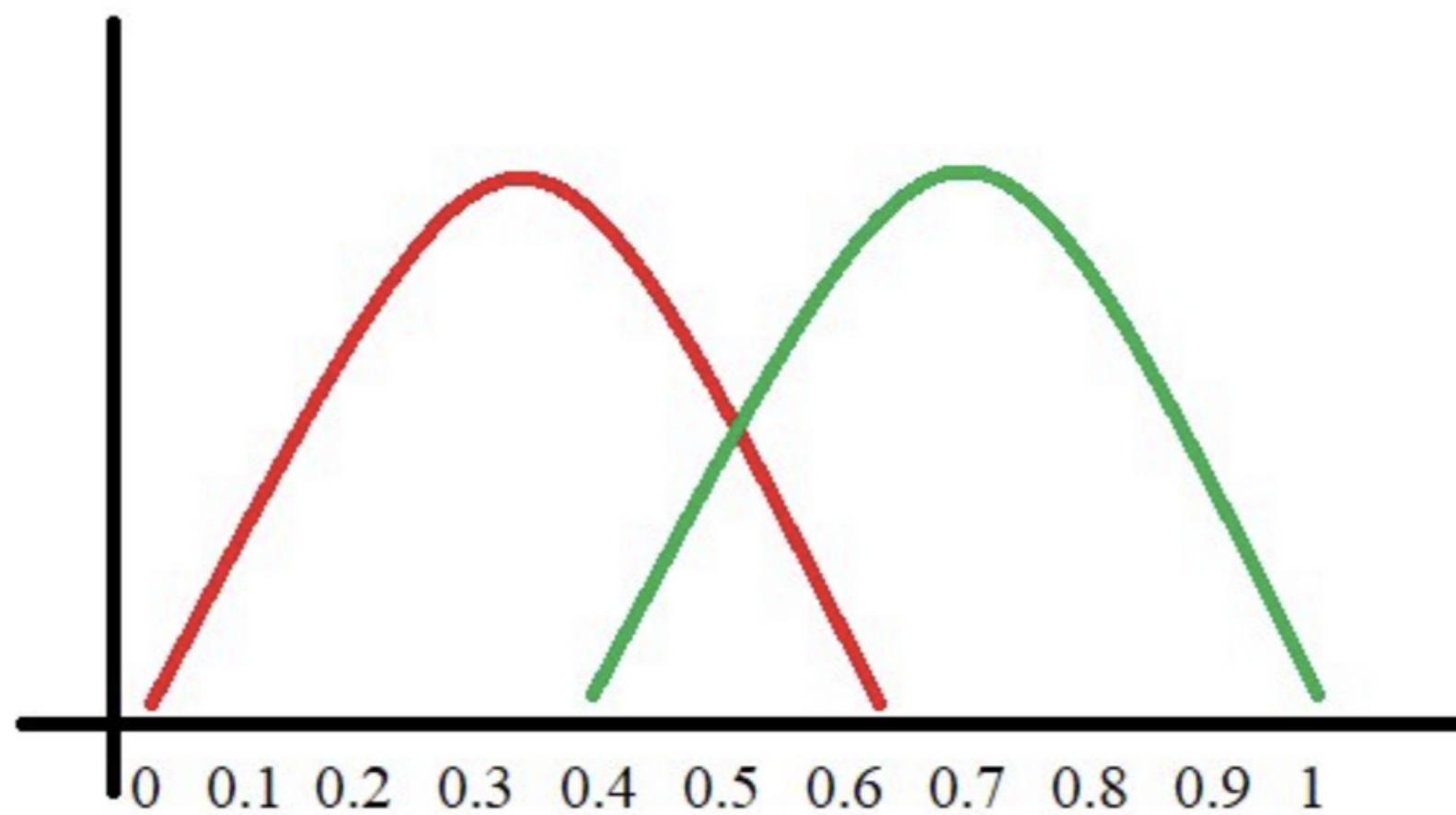
Receiver Operating Characteristic.

Otra métrica para modelos de clasificación que nos dice que tan bien nuestro modelo puede distinguir entre 2 clases.

Umbral

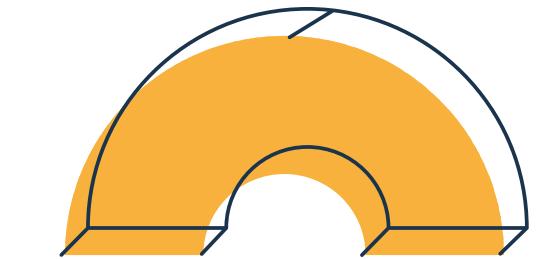


Tenemos un modelo que predice si un paciente está enfermo o no. El modelo, en lugar de retornarnos 1s o 0s como veníamos viendo hasta ahora, nos retorna una probabilidad de que el paciente esté enfermo.

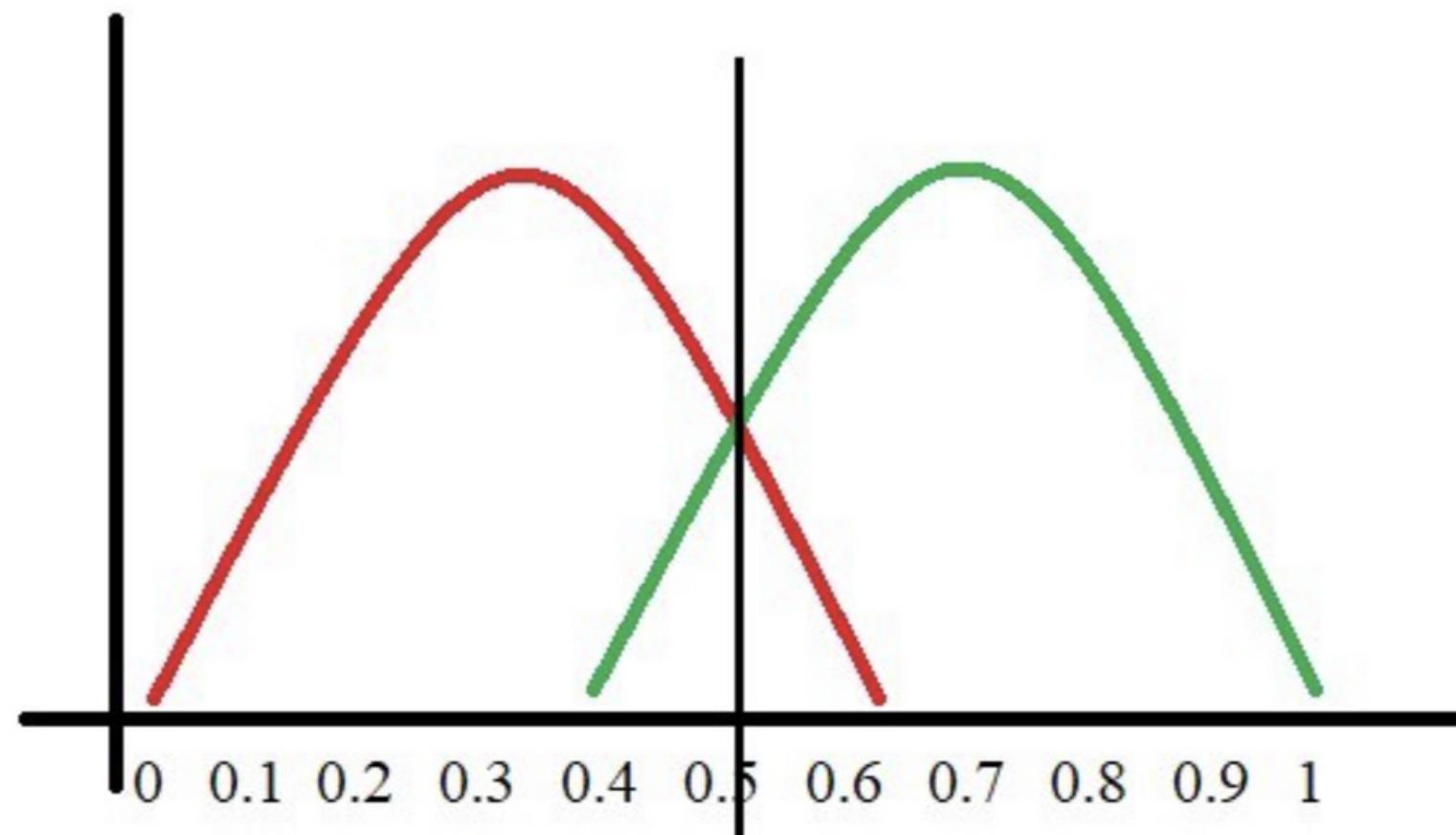


La distribución en rojo, corresponde a pacientes que no tienen la enfermedad. La verde, a pacientes que si la tienen.

Umbral

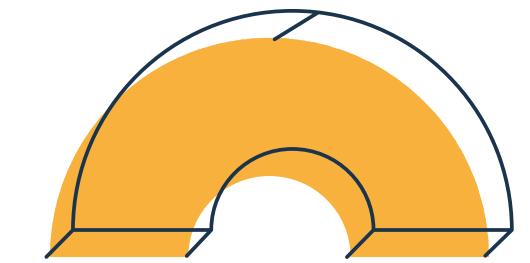


Ahora tenemos que decidir un umbral (threshold) a partir del cual decimos que el paciente está enfermo. Tomemos como ejemplo 0.5:

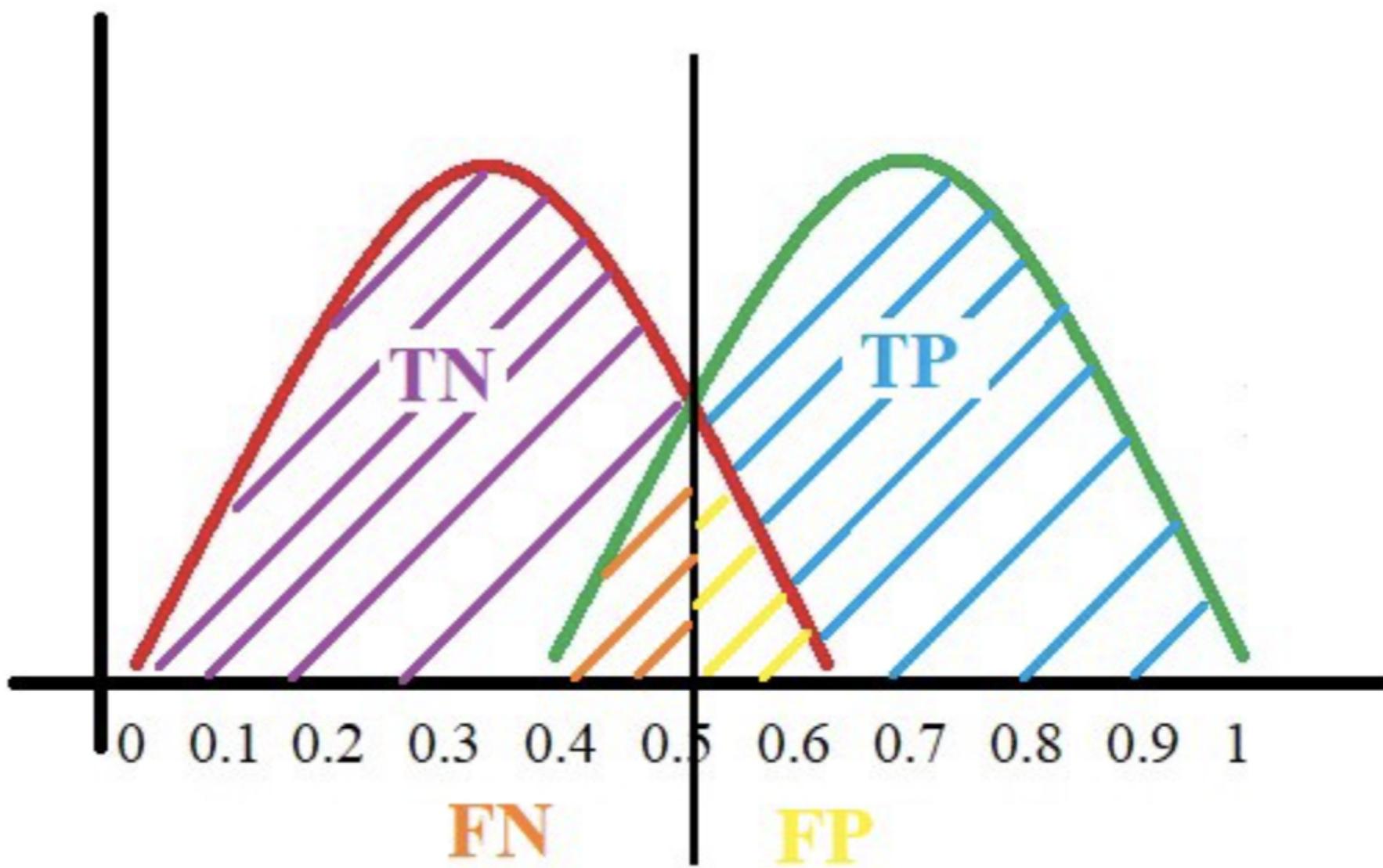


Decimos que cuando el modelo prediga una probabilidad > 0.5 el paciente es considerado enfermo.

Umbral



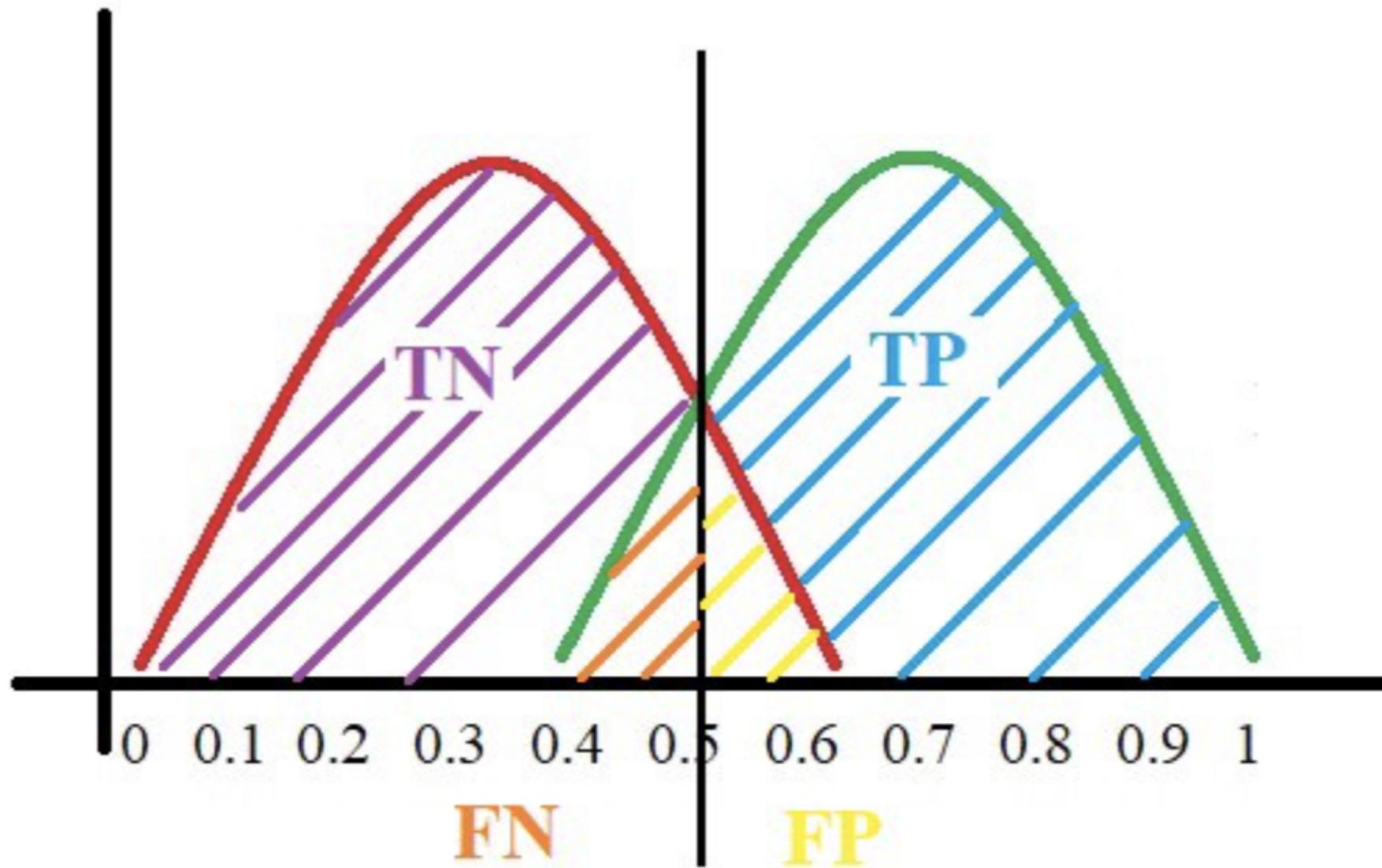
Obtenemos TP, FP, TN, FN:



Umbral



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

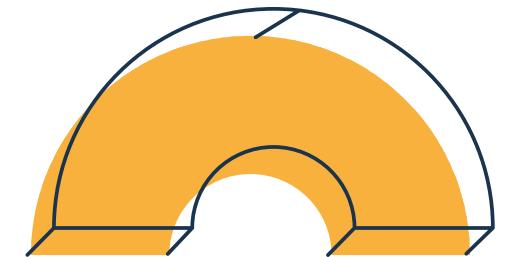


Dependiendo de nuestro problema, podemos definir un threshold más bajo o más alto.

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.

TPR & FPR



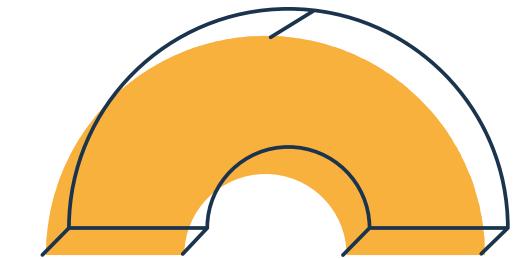
True positive rate: Es lo mismo que la recall!

$$\text{TPR} = \frac{\text{TP}}{\text{P}} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

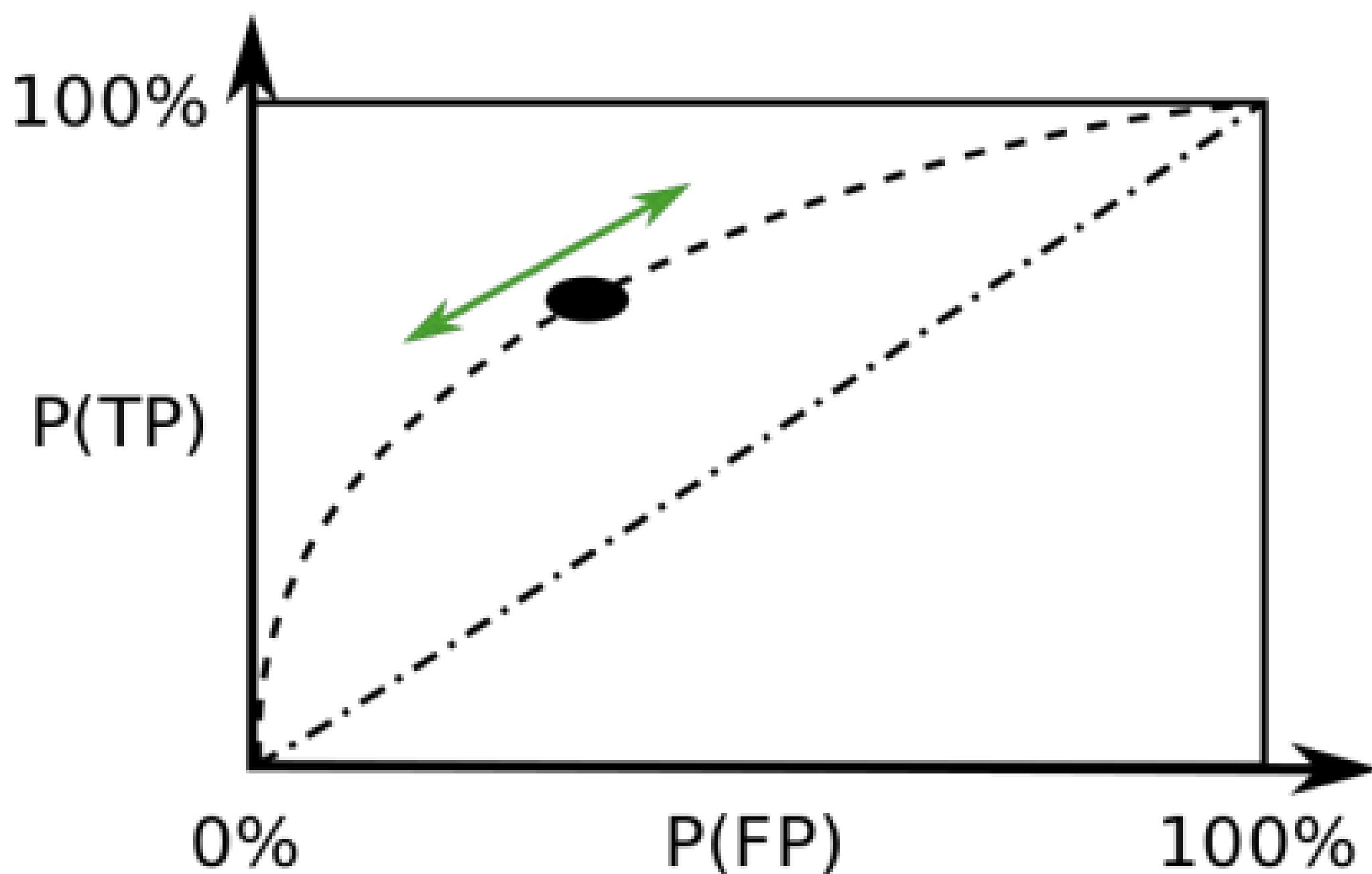
False positive rate: **NO** es lo mismo que precision

$$\text{FPR} = \frac{\text{FP}}{\text{N}} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

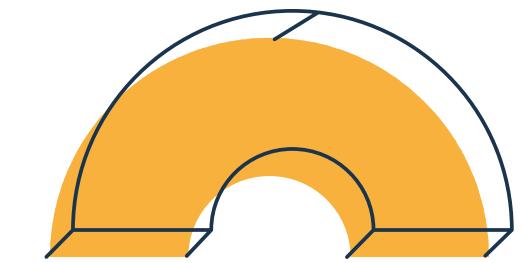
ROC



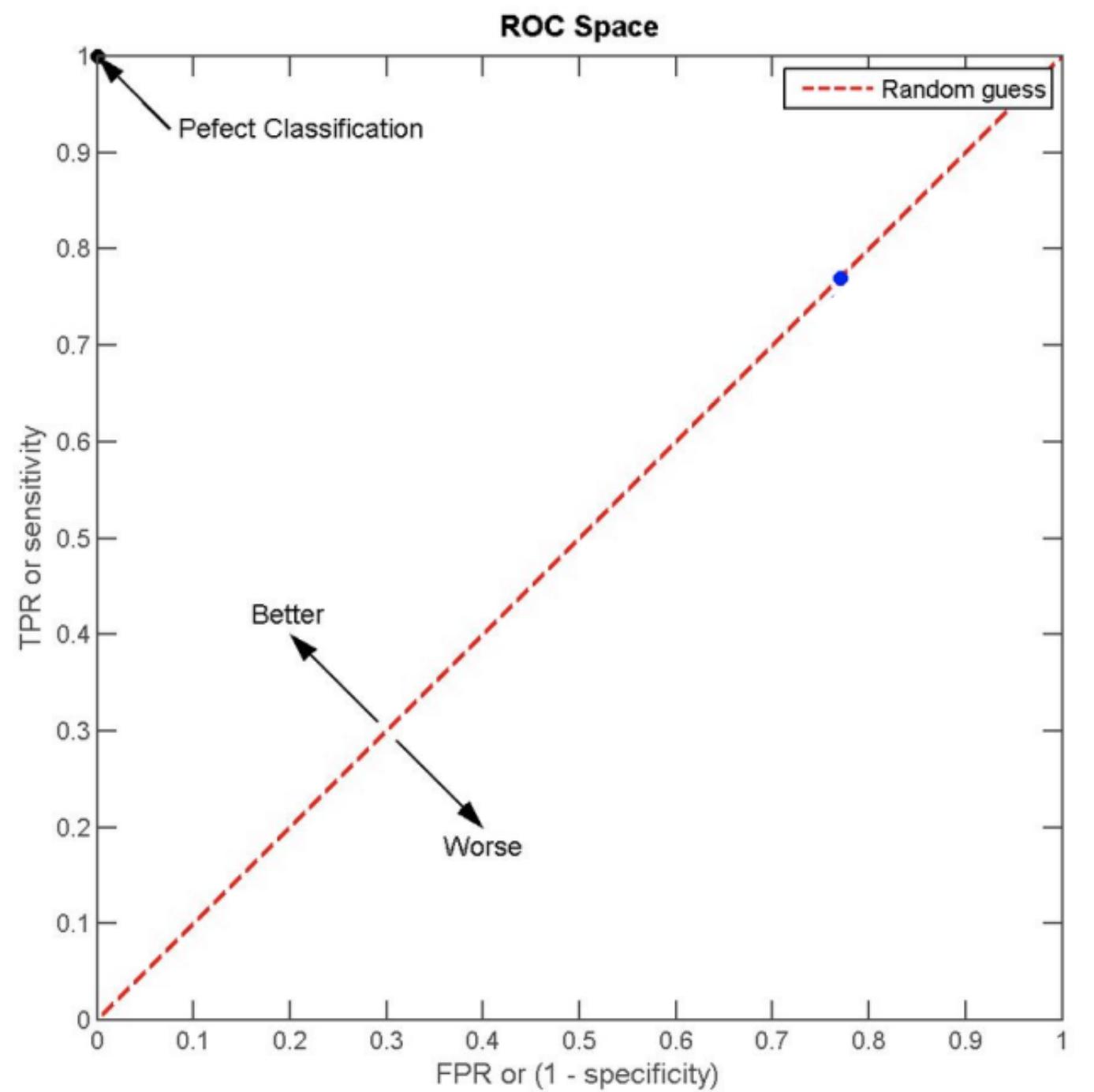
Calculando TPR & FPR para distintos umbrales, podemos armar una curva (ROC)



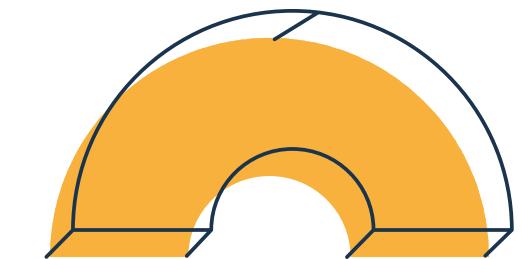
ROC



¿ Cómo interpretamos la calidad ?



AUC ROC

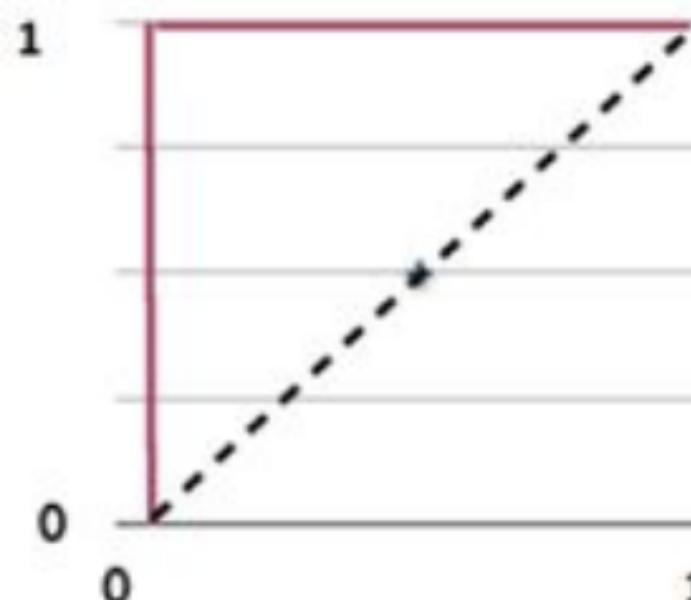


Area bajo la curva ROC.

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

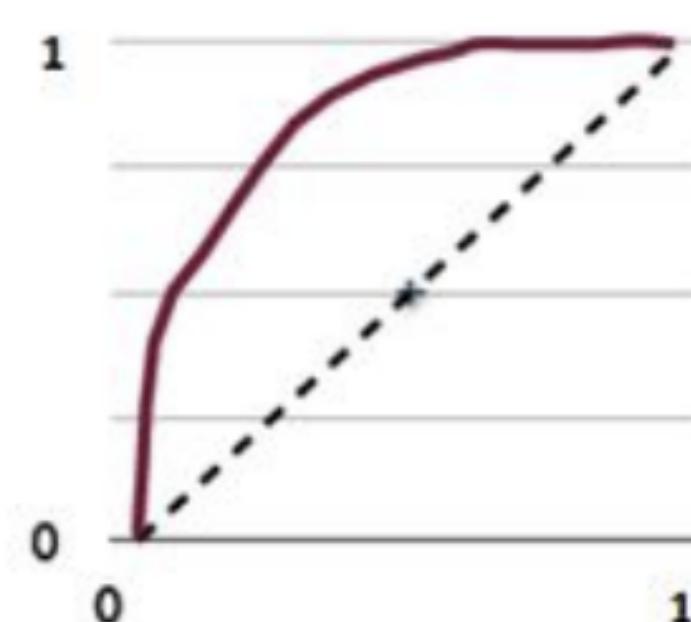
AUC=1

+ valor diagnóstico perfecto



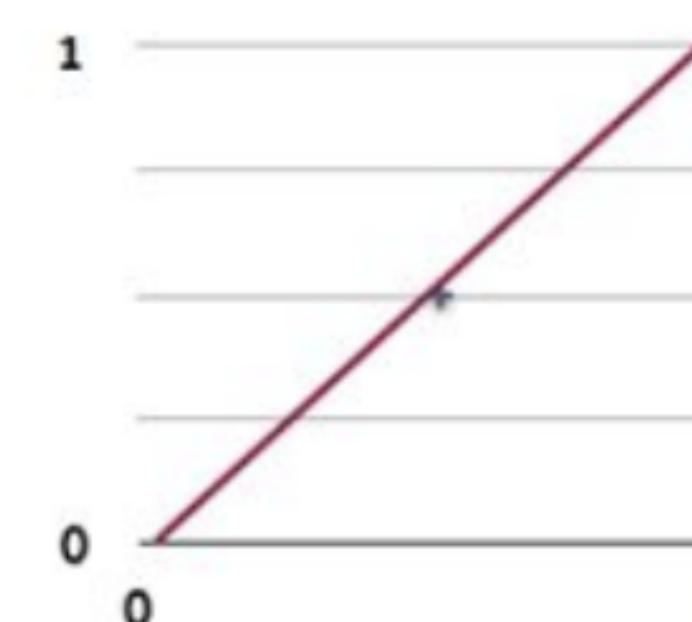
AUC=0,8

+ valor diagnóstico

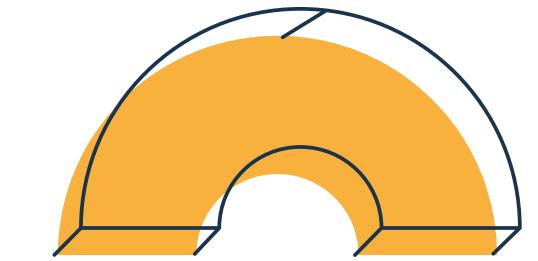
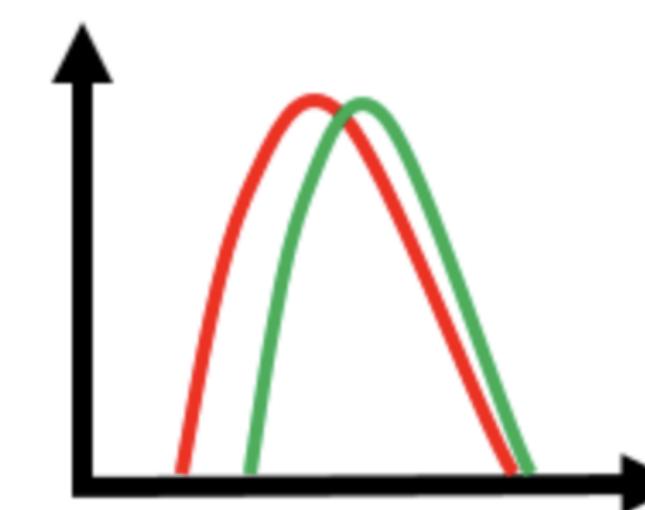
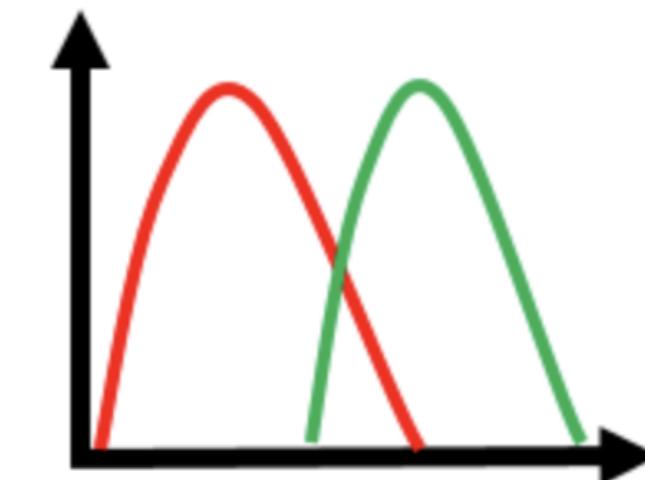
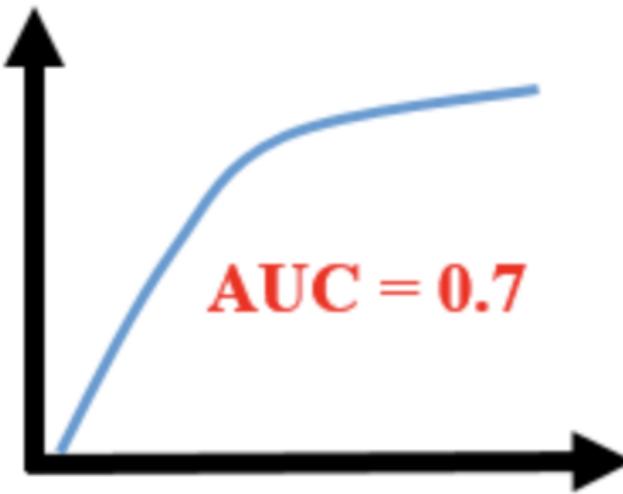
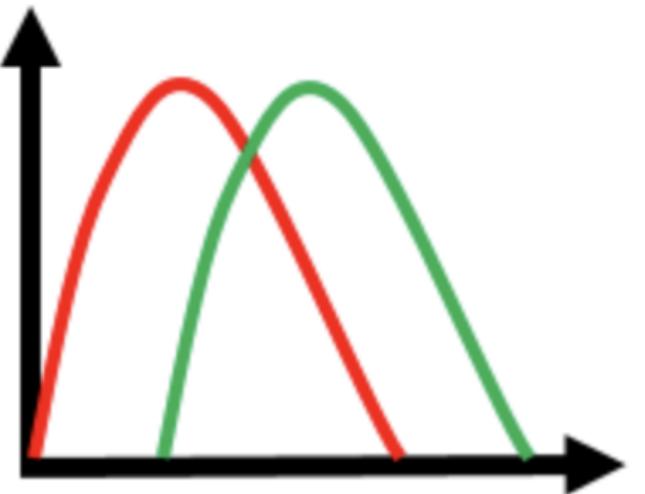
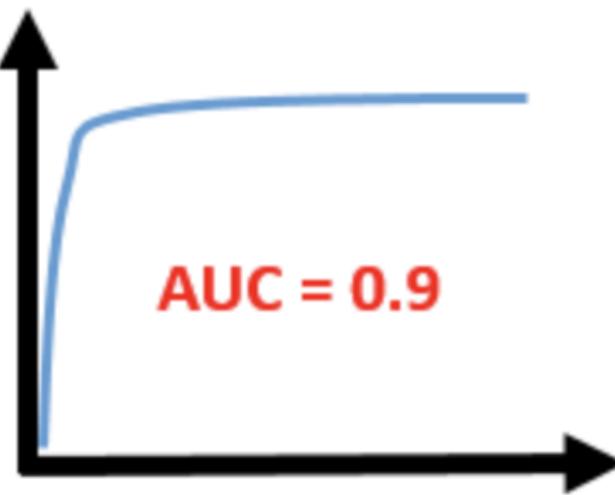
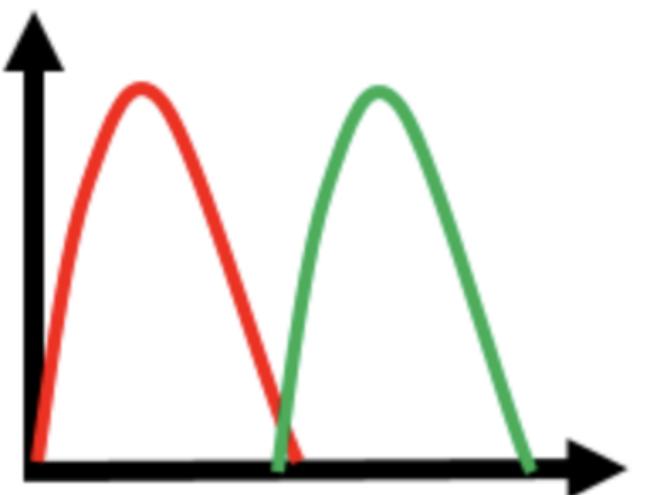


AUC=0,5

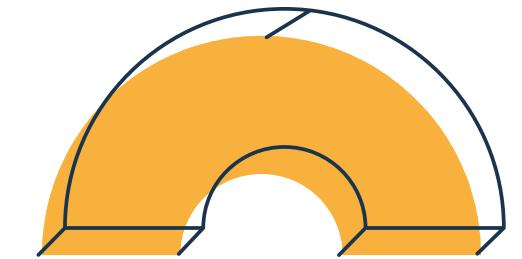
+ sin valor diagnóstico



AUC ROC



AUC ROC



sklearn.metrics.roc_curve

```
sklearn.metrics.roc_curve(y_true, y_score, *, pos_label=None, sample_weight=None, drop_intermediate=True) [source]
```

sklearn.metrics.roc_auc_score

```
sklearn.metrics.roc_auc_score(y_true, y_score, *, average='macro', sample_weight=None, max_fpr=None, multi_class='raise', labels=None) [source]
```

sklearn.metrics.auc

```
sklearn.metrics.auc(x, y) [source]
```