ACÁMICA

¡Bienvenidos/as a Data Science!

Agenda

¿Cómo anduvieron?

Repaso

Explicación: Curva ROC

Hands-On

Break

Entrega 3 + Dudas comunitarias

Cierre

¿Cómo anduvieron?

Repaso

Aprendizaje supervisado: Clasificación

Modelos

- Árboles de decisión (Hiperparámetros: profundidad, criterio de entrenamiento, etc.)

- KNN (Hiperparámetros: cantidad de vecinos, distancia, etc.)

Métricas de evaluación

- Exactitud
- Precisión/Exhaustividad
- F-Score
- Matriz de Confusión[1]

[1] Bueno, técnicamente no es una métrica

Aprendizaje supervisado: Clasificación

Métricas de evaluación

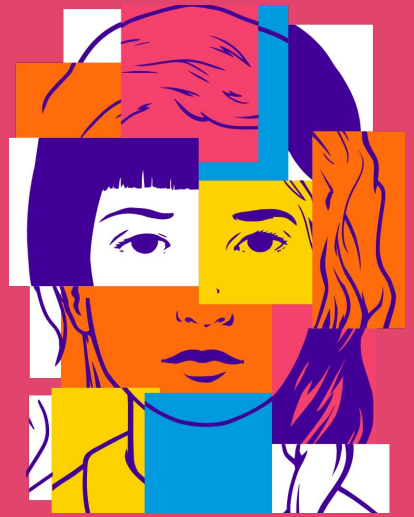
- Exactitud
- Precisión/Exhaustividad
- F-Score
- Matriz de Confusión[1]

También, vimos que:

- Precisión y Exhaustividad compiten entre sí.
- Exactitud puede ser una métrica engañosa en problemas desbalanceados.

Pero todavía hay más

Scores



Clasificación Binaria • Matriz de confusión

Clase Predicha

Clase Verdadera	Clase Positiva	Clase Negativa
Clase Positiva	TP	FN
Clase Negativa	FP	TN

¡Tiene toda la información que necesitamos!

Clasificación Binaria • Matriz de confusión

Clase Predicha

Clase Verdadera	Clase Positiva	Clase Negativa
Clase Positiva	TP	FN
Clase Negativa	FP	TN

¡Tiene casi toda la información que necesitamos!

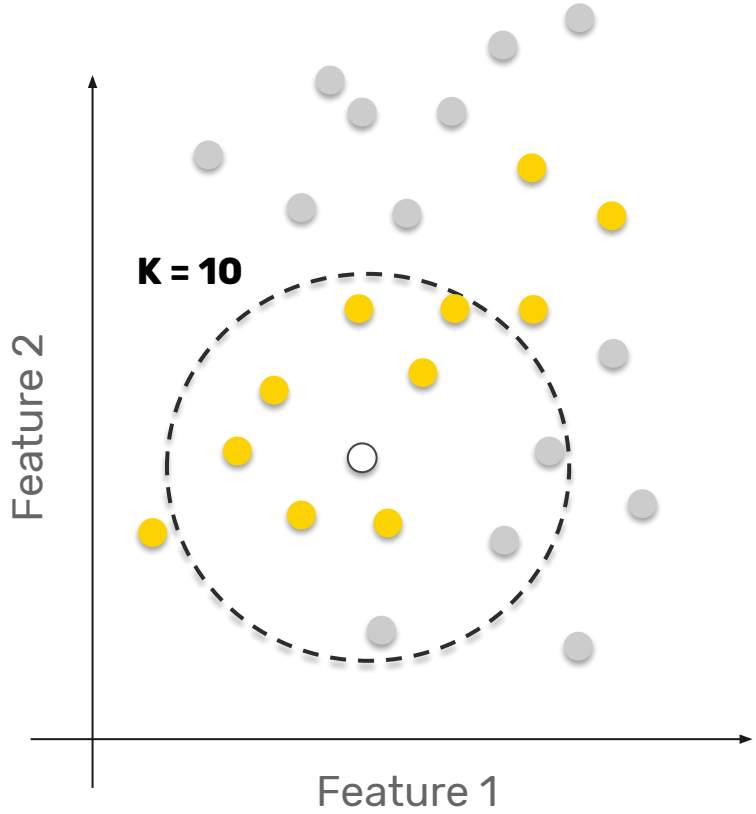
Clasificación Binaria • Score

Pero, ¿cómo llegamos a los aciertos (TPs), Falsos Positivos, etc.?

Por ejemplo, pensemos en un modelo de vecinos más cercanos, con K = 10.

En este caso, de 10 vecinos, 7 son amarillos, por lo que la etiqueta correspondiente sería amarilla.

Lo mismo ocurrirá cuando haya más de 5 vecinos de color amarillo. Si, en cambio, hay menos de 5 vecinos de color amarillo, la etiqueta pasa a ser gris.

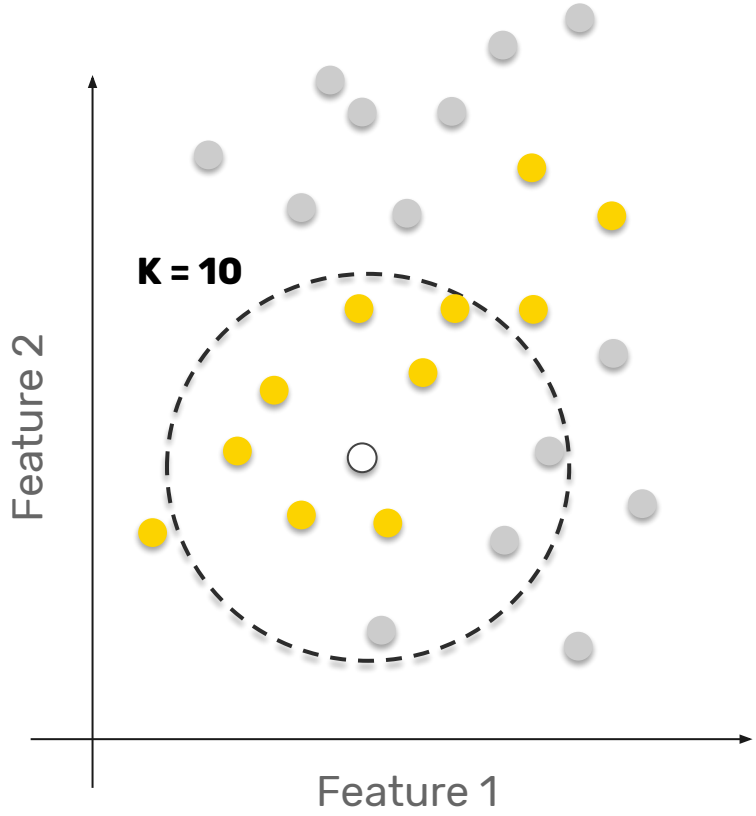


Clasificación Binaria • Score

Entonces, de una instancia que tiene 10 vecinos amarillos, podemos estar más *seguros* que la etiqueta correspondiente es amarrilla que una instancia que solamente tiene 6 vecinos.

Cuando miramos solamente las etiquetas asignadas, esta información la perdemos.

¿Qué podemos hacer?



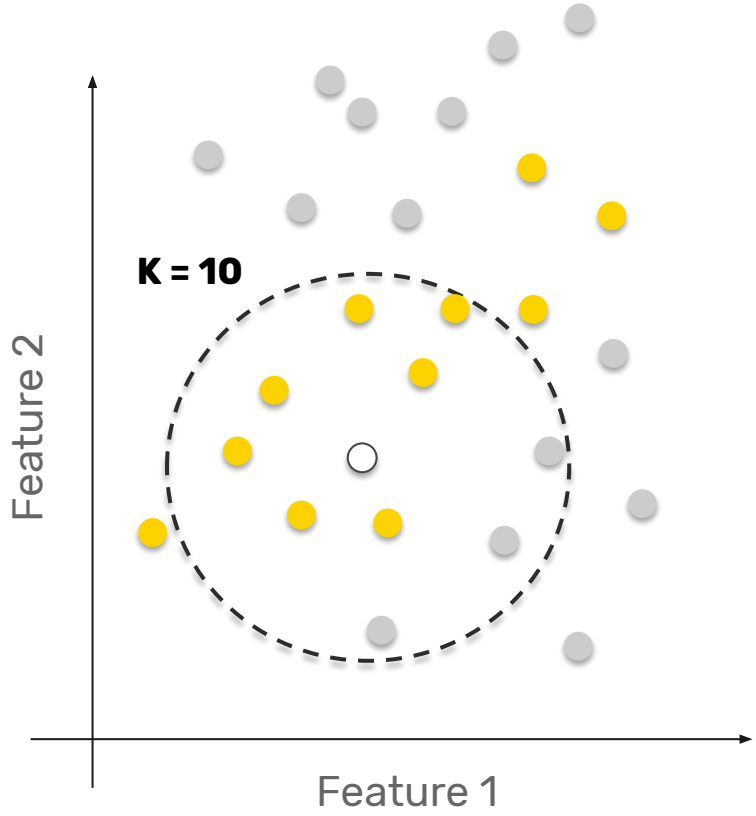
Clasificación Binaria • Score

Entonces, de una instancia que tiene 10 vecinos amarillos, podemos estar más *seguros* que la etiqueta correspondiente es amarrilla que una instancia que solamente tiene 6 vecinos.

Cuando miramos solamente las etiquetas asignadas, esta información la perdemos.

¿Qué podemos hacer?

¡Generar un Score que represente cuán seguro está el modelo de la etiqueta!



Clasificación Binaria • Score

Este razonamiento se puede hacer con (casi) todos los modelos que usemos. En el fondo, lo que un modelo hacer para asignar etiquetas es generar un score y poner el umbral *a la mitad*.

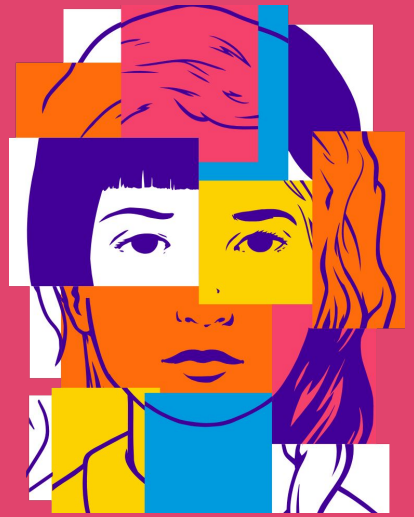
Para pensar: ¿cómo generan los scores los árboles de decisión?

Como siempre, no lo tenemos que programar.
En Scikit-Learn, todos los modelos vienen con un método, predict_proba(X), que calcula los scores.

Importante: si bien a primer orden estos scores pueden ser interpretados como probabilidades, la realidad es que no lo son, porque no están **Calibrados**. Lo que sí podemos usar son los **Rankings** que generan.

¿Podemos usar estos Scores para caracterizar mejor el desempeño de nuestro modelo y, además, tomar mejores decisiones?

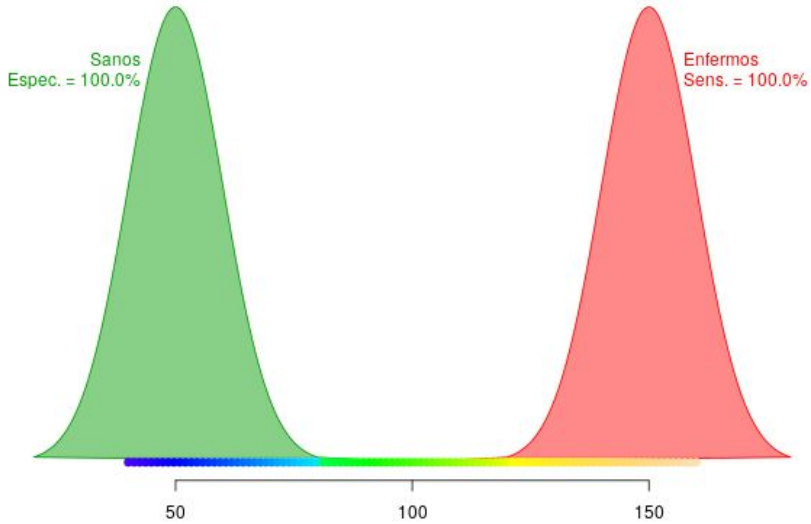
Curva ROC



Umbrales

¿Qué ocurre si usamos un umbral y asignamos etiquetas según el Score sea superiores o inferiores al valor elegido?

Hasta ahora, hacemos esto:



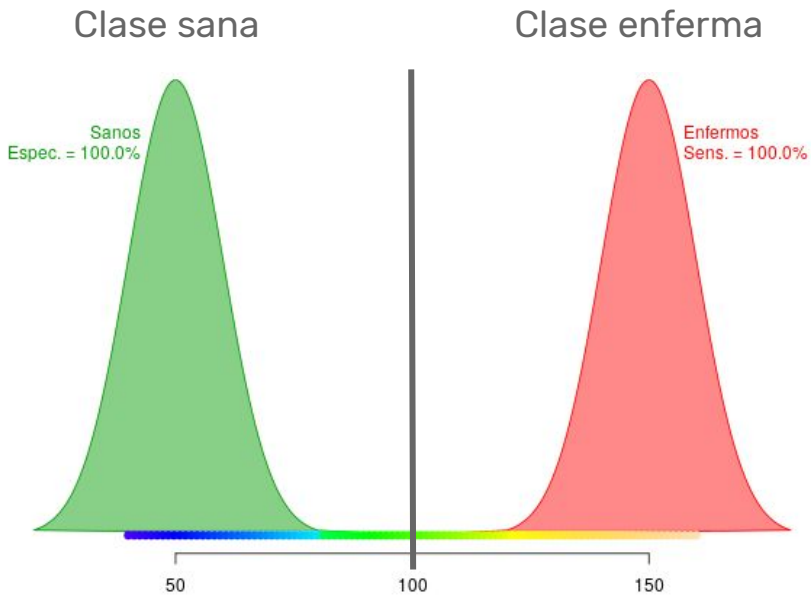
Output (Score) de un modelo de clasificación

Fuente y para jugar un poco: https://www.bioestadistica.uma.es/analisis/roc1/

Umbrales

¿Qué ocurre si usamos un umbral y asignamos etiquetas según el Score sea superiores o inferiores al valor elegido?

Hasta ahora, hacemos esto:



Output (Score) de un modelo de clasificación

Fuente y para jugar un poco: https://www.bioestadistica.uma.es/analisis/roc1/

Umbrales

¿Qué ocurre si usamos un umbral y asignamos etiquetas según el Score sea superiores o inferiores al valor elegido?

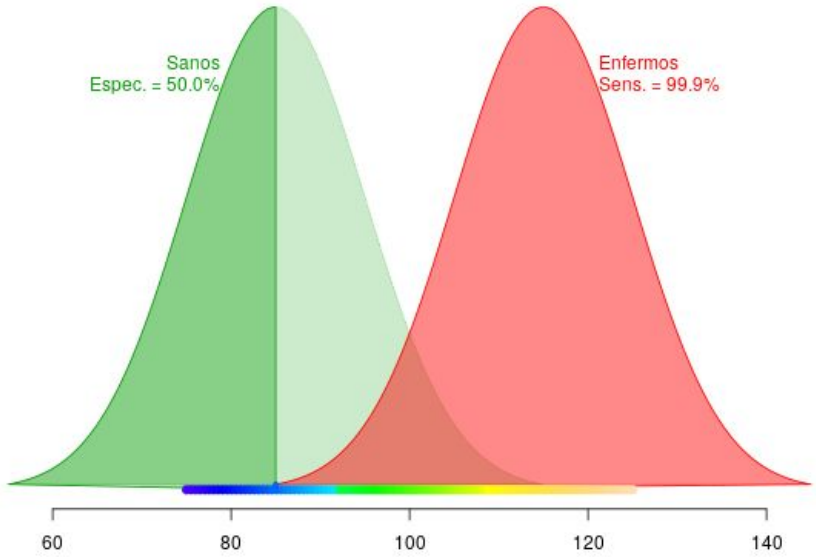
Hasta ahora, hacemos esto:

Output (Score) de un modelo de clasificación

Fuente y para jugar un poco: https://www.bioestadistica.uma.es/analisis/roc1/

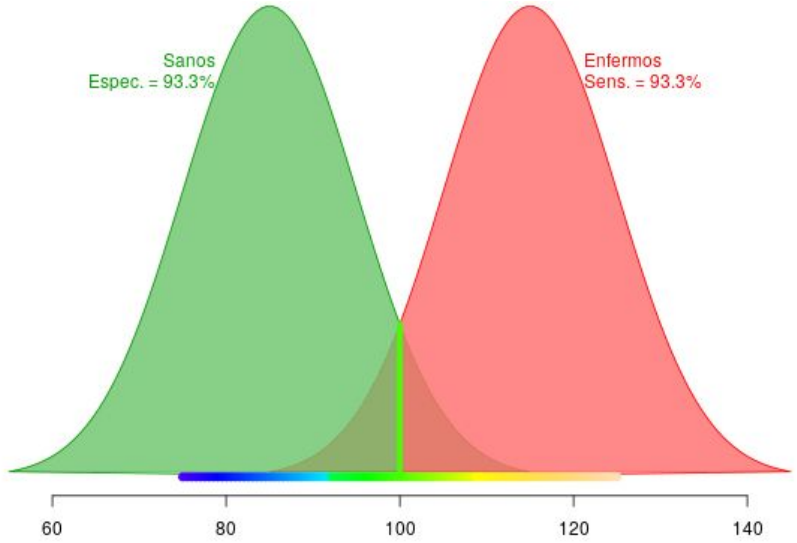
Umbrales

Pero, para nuestro problema,
tal vez sea mejor hacer esto:



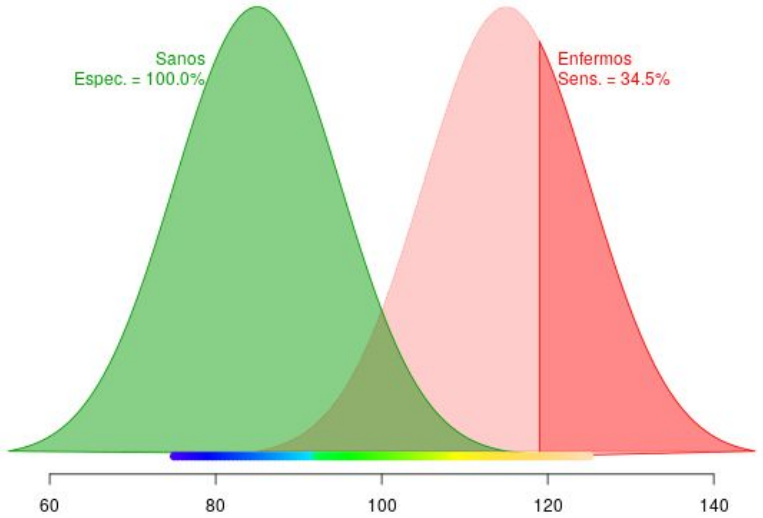
Umbrales

...o esto (mismo caso que antes):

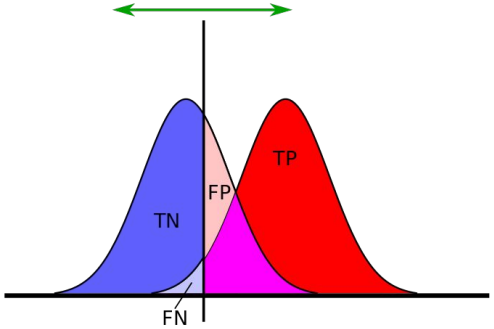


Umbrales

...o esto:



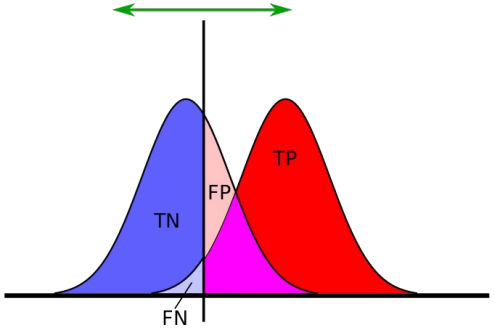
A medida que movemos el umbral, vamos cambiando la cantidad de Aciertos, Falsos positivos, Falsos Negativos y Verdaderos Negativos.



TP	FP
FN	TN

Fuente

A medida que movemos el umbral, vamos cambiando la cantidad de Aciertos, Falsos positivos, Falsos Negativos y Verdaderos Negativos.



TP	FP
FN	TN

Podemos cuantificar con:

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

True Positive Rate → Es la exhaustividad

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

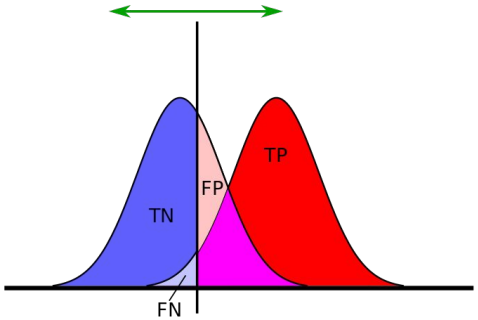
False Positive Rate → NO es la precisión

¡Y hacer una curva de uno contra el otro a medida que variamos el umbral!

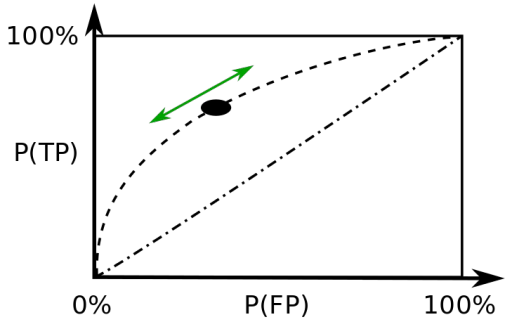
Fuente

Curva ROC

La curva ROC es la representación del TPR vs. el FPR para cada valor de corte.

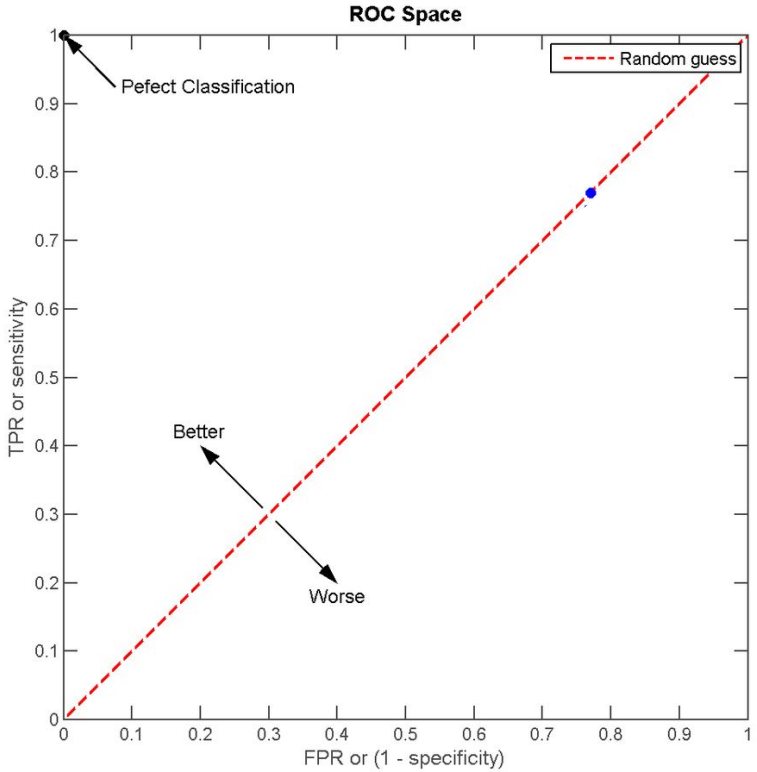


TP	FP
FN	TN



Curva ROC

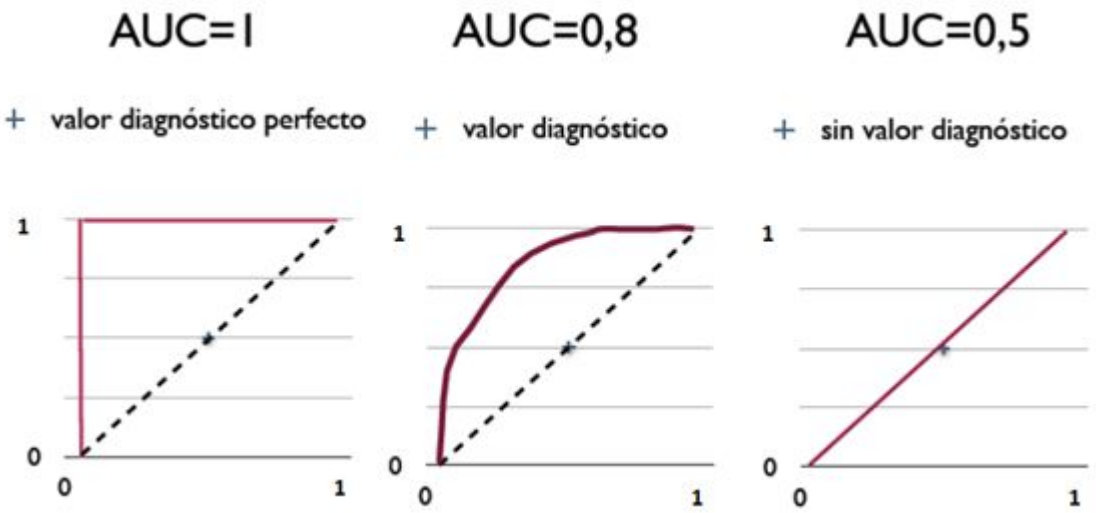
¿Cómo interpretamos la calidad de las curvas ROC?



Fuente

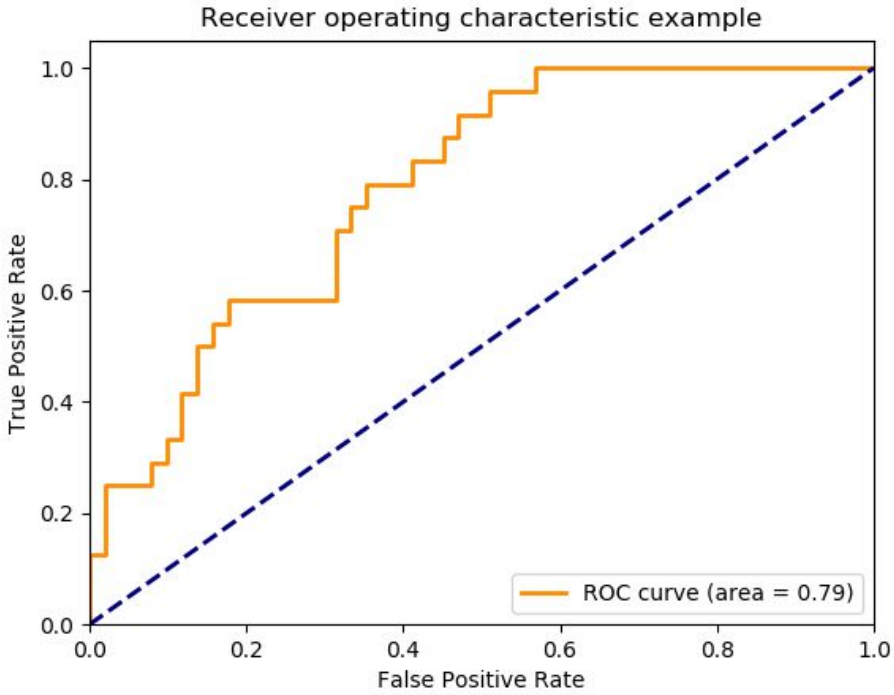
Curva ROC • Área bajo la curva ROC (AUC ROC)

¿Y si queremos cuantificar? La medida de 'cuán buena' es la curva ROC es calculado el área bajo la curva (AUC).
¡No siempre tendremos que graficar la curva!



¿Qué valor de umbral elijo?
Esto va a depender de nuestro problema, si queremos favorecer precisión o exhaustividad.

Pero al área bajo la curva es un indicador de cuán bueno es nuestro modelo independientemente de eso.



Fuente

Conclusiones

- Una **curva ROC** (Receiver Operating Characteristic) es una representación gráfica que ilustra la relación entre TPR y el FPR de un sistema clasificador para diferentes puntos de corte. NO confundir con una curva Precision-Recall.

- Se puede usar **para generar estadísticos** que resumen el rendimiento o la efectividad de un clasificador. El indicador más utilizado en muchos contextos es el área bajo la curva ROC o AUC (AUC- Área Bajo la Curva).

- Dato histórico: fue **desarrollada por ingenieros eléctricos en la II Guerra Mundial**, para medir la eficacia de la detección de objetos enemigos en el campo de batalla mediante señales de radar. Su uso está muy extendido en medicina para validar técnicas diagnósticas.

Curva ROC en Scikit Learn

sklearn.metrics.roc_curve

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

Compute Receiver operating characteristic (ROC)

Note: this implementation is restricted to the binary classification task.

sklearn.metrics.roc_auc_score

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

Compute Area Under the Receiver Operating Characteristic Curve (ROC AUC) from prediction scores.

Note: this implementation is restricted to the binary classification task or multilabel classification task in label indicator format.

sklearn.metrics.auc

`sklearn.metrics.` **`auc`** (*x*, *y*, *reorder='deprecated'*) [source]

Compute Area Under the Curve (AUC) using the trapezoidal rule

This is a general function, given points on a curve. For computing the area under the ROC-curve, see `roc_auc_score`. For an alternative way to summarize a precision-recall curve, see `average_precision_score`.

Hands-on training

Hands-on training

DS_Encuentro_21_DDesb.ipynb

Vamos a agregar a este Notebook la caracterización por curva ROC

¡BREAK!

Ph. Credit: Drew Coffmann

Proyecto 2:
Entrega 3

Proyecto 2: Modelado

Actividad:
Dudas comunitarias

Para la próxima

1. Ver los videos de la plataforma “Ajustes del Modelo”
2. Completar Notebook de hoy y atrasados.
3. Terminar la Entrega 03.

ACÁMICA