



# Machine Learning 101

Métricas



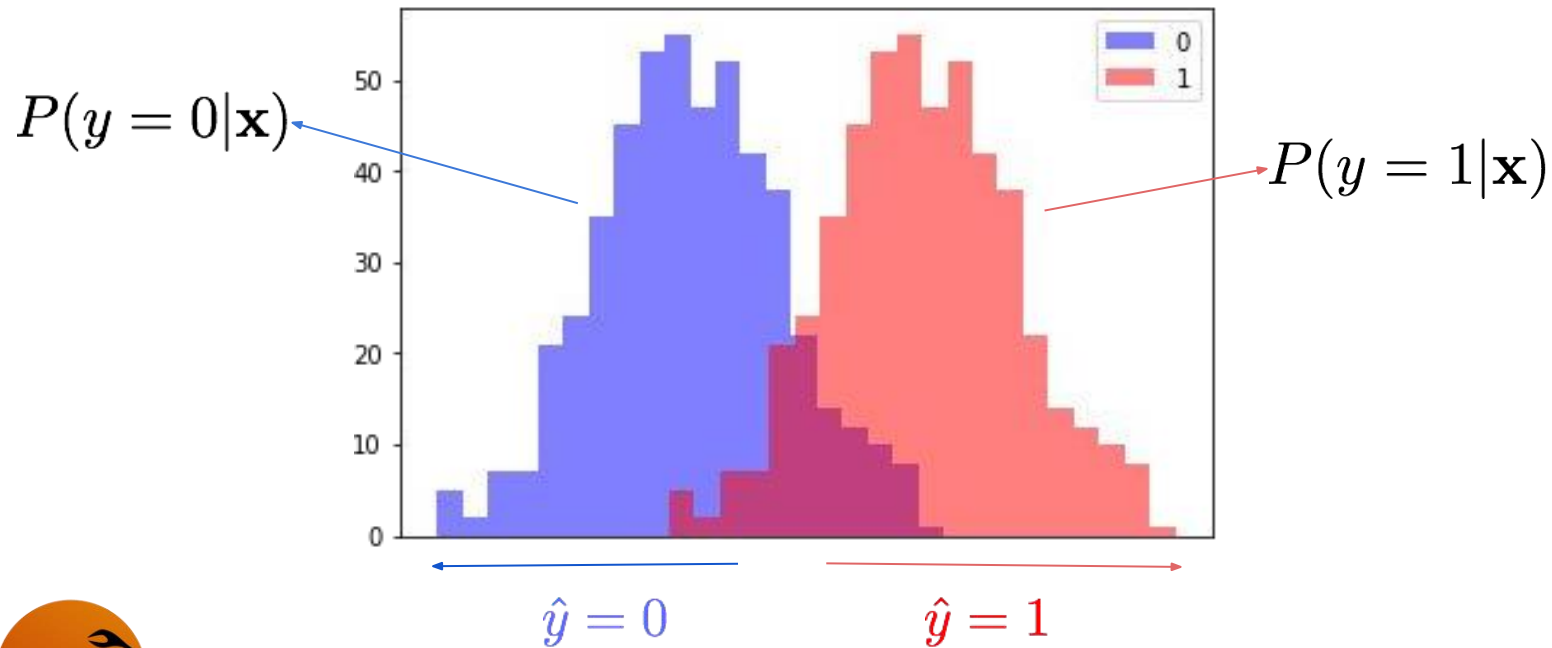
# Índice

1. **Métricas en clasificación**
2. Problemas desbalanceados
3. Métricas en regresión



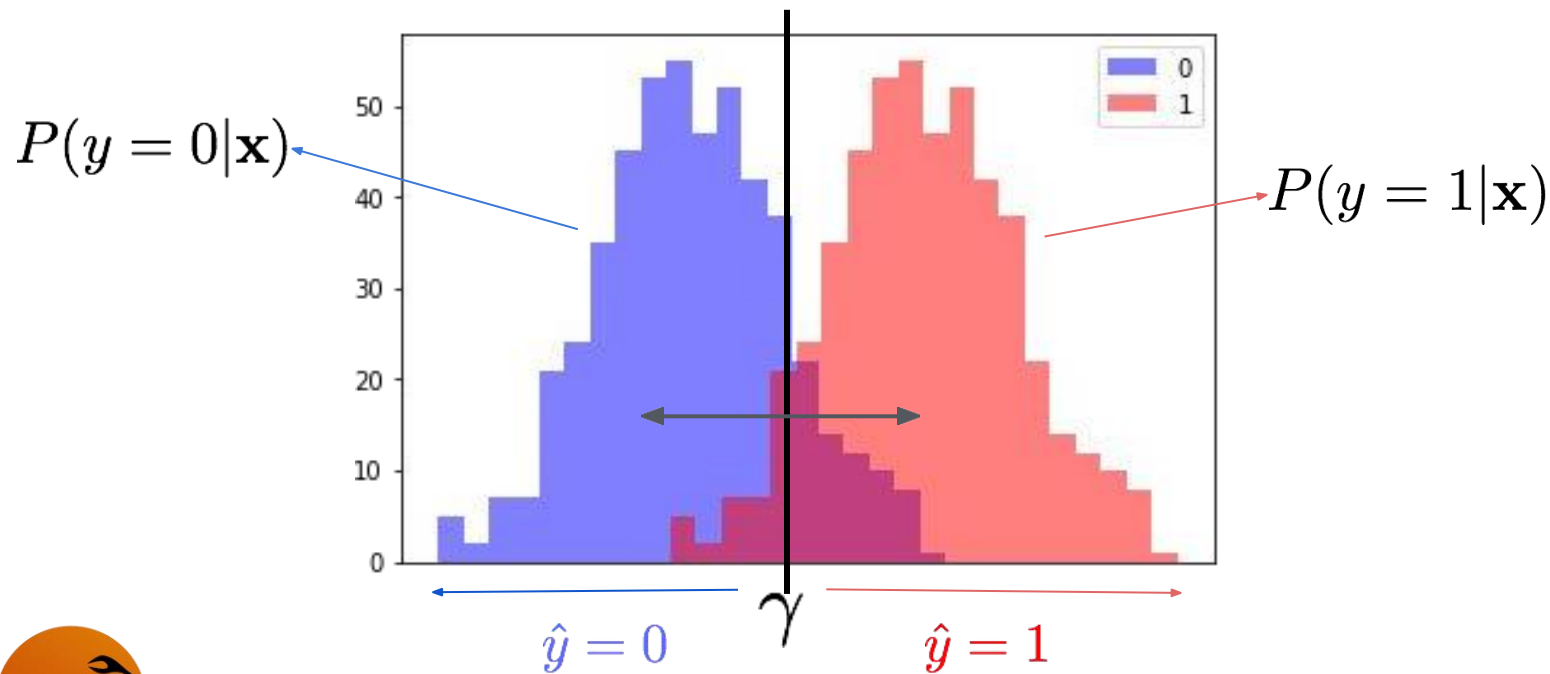
# Teoría de la decisión

- Regresión logística:



# Teoría de la decisión

- Regresión logística:  $P > 0.5 = \gamma$



# ■ Métrica 1: tasa de error

- Contar errores:

True: [1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0]

Pred: [1 0 0 0 0 0 0 1 0 0 1 0 0 0 0 1 1 0 0 1]

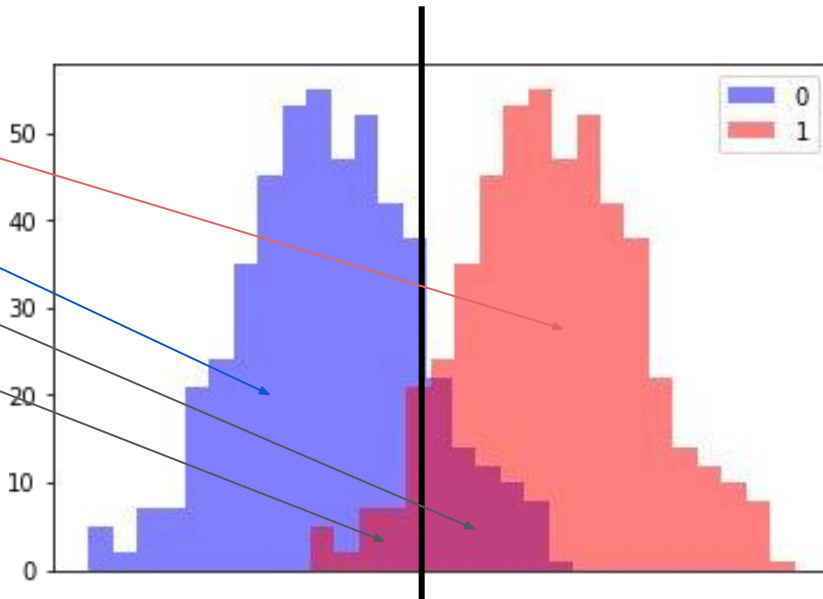
- **Tasa de error (ERR):** # errores / N
- **Tasa de acierto (ACC):** # aciertos / N
  - $ACC = 1 - ERR$

- Da igual el sentido del error



# ■ Otras tasas de interés

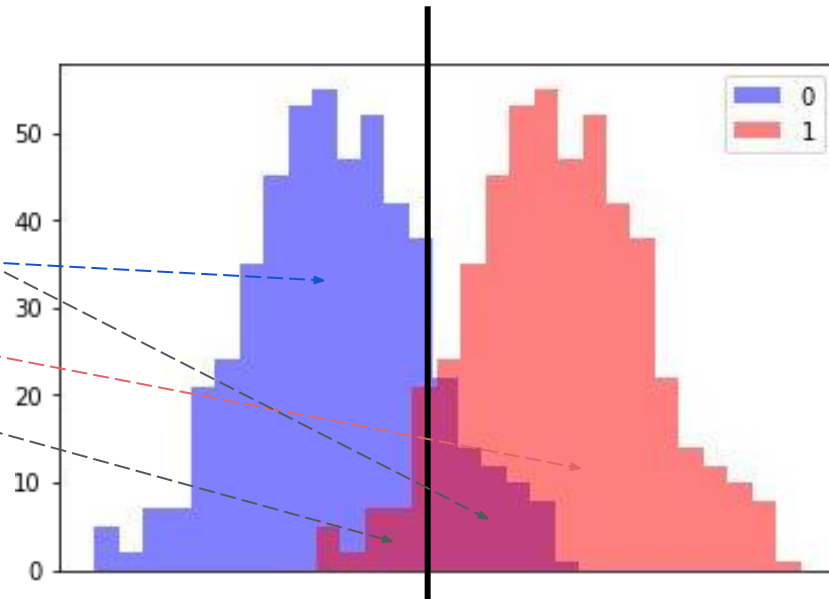
- Si cuento el sentido de los errores, en un problema de clasificación binaria tengo cuatro posibilidades:
  - True Positive (TP)
  - True Negative (TN)
  - False Positive (FP)
  - False Negative (FN)



# Matriz de confusión

- Representamos estas tasas en modo de **matriz de confusión**

		Etiquetas predichas	
		y_pred = 0	y_pred = 1
Etiquetas reales	y_true = 0	<b>TN</b>	<b>FP</b>
	y_true = 1	<b>FN</b>	<b>TP</b>



# Métricas en clasificación

- Sobre la matriz de confusión se definen la siguientes métricas

		Etiquetas predichas	
		y_pred = 0	y_pred = 1
Etiquetas reales	y_true = 0	<b>TN</b>	<b>FP</b>
	y_true = 1	<b>FN</b>	<b>TP</b>

$$\text{ACC} = \frac{TP + TN}{TP + TN + FP + FN}$$

Accuracy

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

Sensitivity

$$\text{PPV} = \text{Precisión} = \frac{TP}{TP + FP}$$

Positive predictive value

$$\text{ESP} = \frac{TN}{TN + FP}$$

Specificity

$$\text{FSC} = \text{F1-score} = \frac{2 \cdot \text{PPV} \cdot \text{SEN}}{\text{PPV} + \text{SEN}}$$

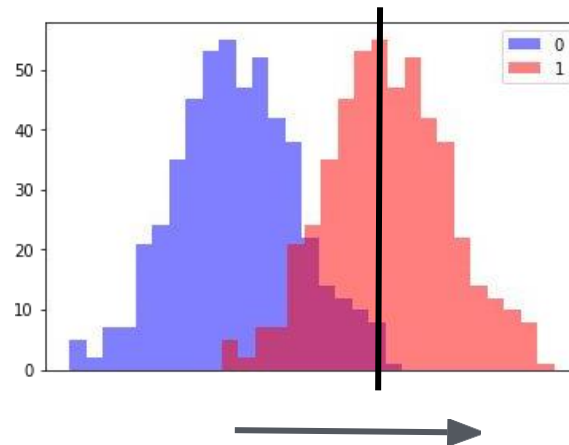




# ■ Compromiso entre métricas (I)

- Hay un compromiso entre las métricas (no se puede tener todo)

		Etiquetas predichas	
		y_pred = 0	y_pred = 1
Etiquetas reales	y_true = 0	<b>TN</b>	<b>FP</b>
	y_true = 1	<b>FN</b>	<b>TP</b>



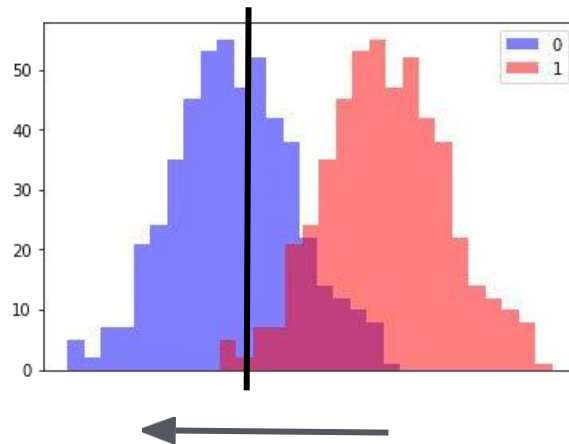
- Si umbral  $\Rightarrow$ , entonces  $TP \downarrow$ ,  $TN \uparrow$ ,  $FP \downarrow$ ,  $FN \uparrow$ 
  - $SEN \downarrow$ ,  $ESP \uparrow$ ,  $PP \uparrow$



# Compromiso entre métricas (II)

- Hay un compromiso entre las métricas (no se puede tener todo)

		Etiquetas predichas	
		y_pred = 0	y_pred = 1
Etiquetas reales	y_true = 0	<b>TN</b>	<b>FP</b>
	y_true = 1	<b>FN</b>	<b>TP</b>

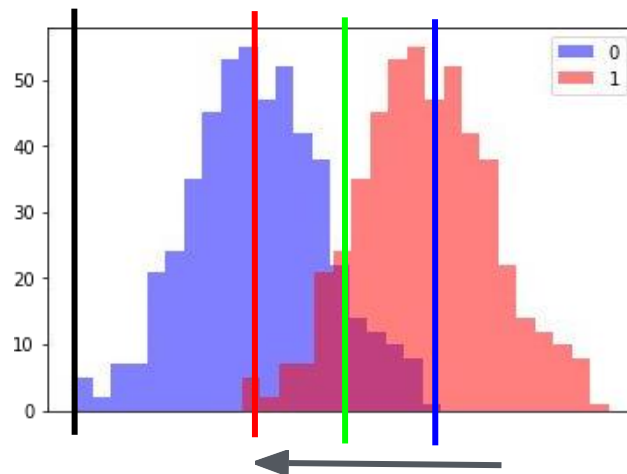
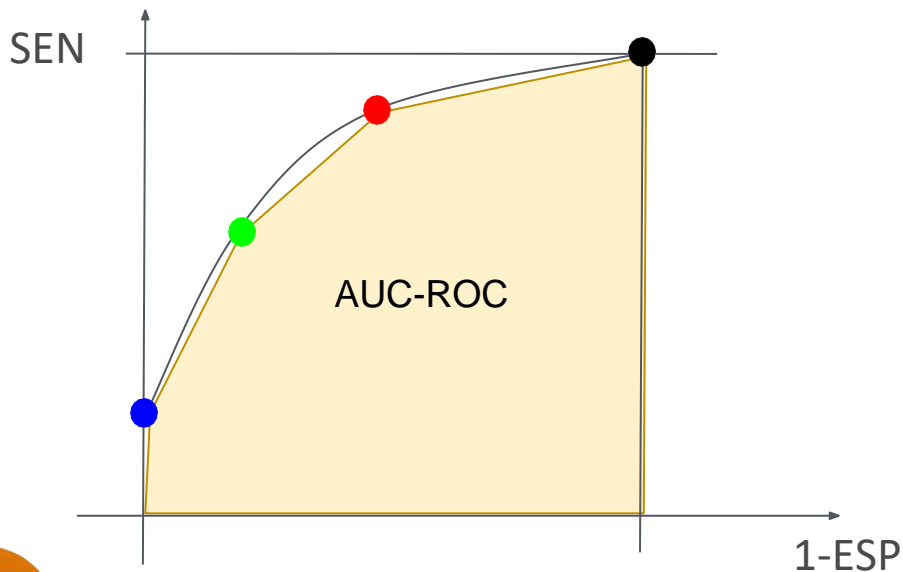


- Si umbral  $\leftarrow$ , entonces  $TP \uparrow$ ,  $TN \downarrow$ ,  $FP \uparrow$ ,  $FN \downarrow$ 
  - $SEN \uparrow$ ,  $ESP \downarrow$ ,  $PP \downarrow$



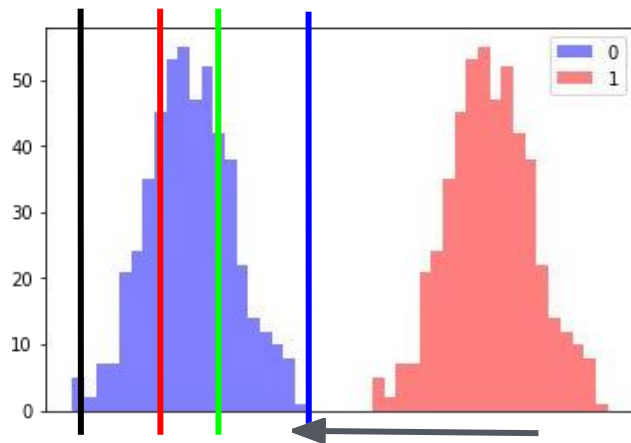
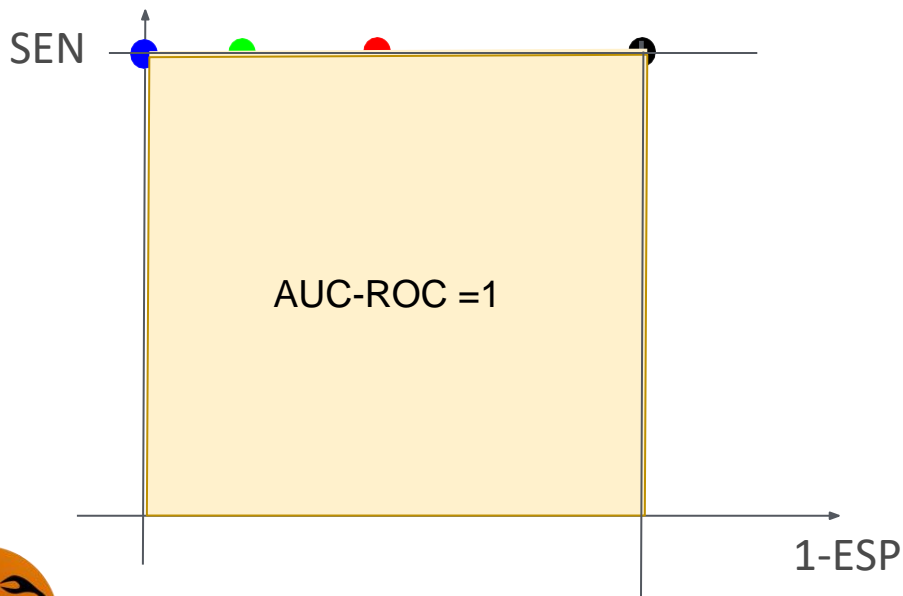
# Curva ROC

- Representa la SEN vs 1-ESP (Tasa de Falsos Positivos) cuando desplazo el umbral



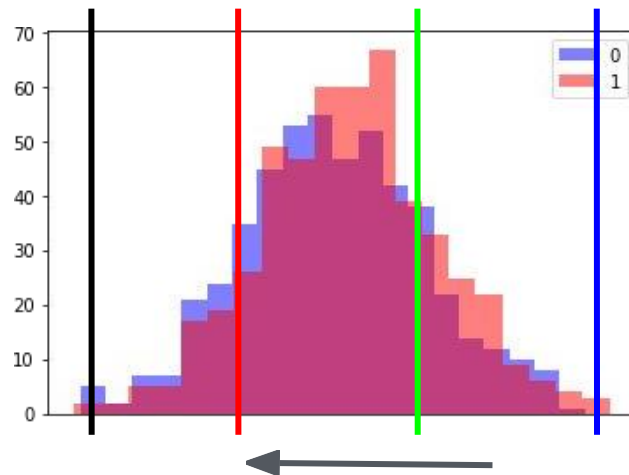
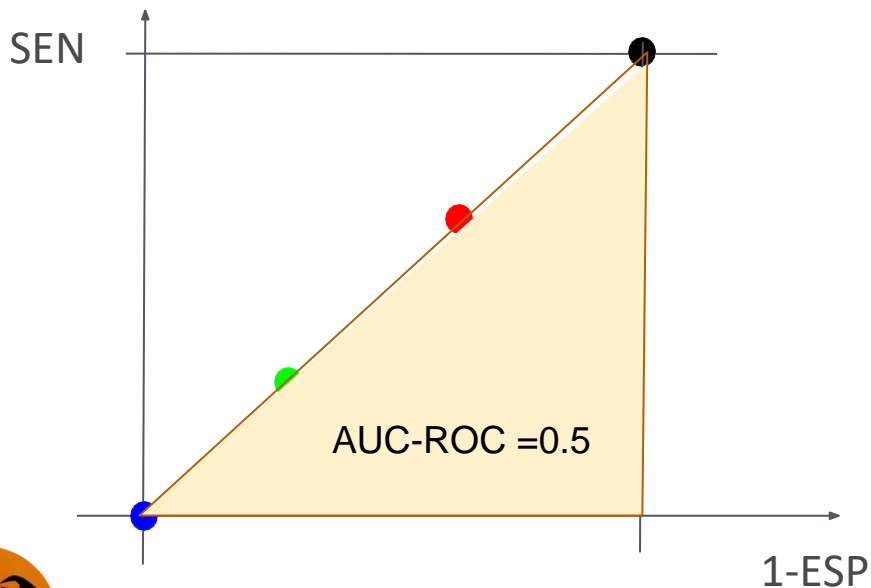
# Curva ROC: situación ideal

- Representa la SEN vs 1-ESP (Tasa de Falsos Positivos) cuando desplazo el umbral



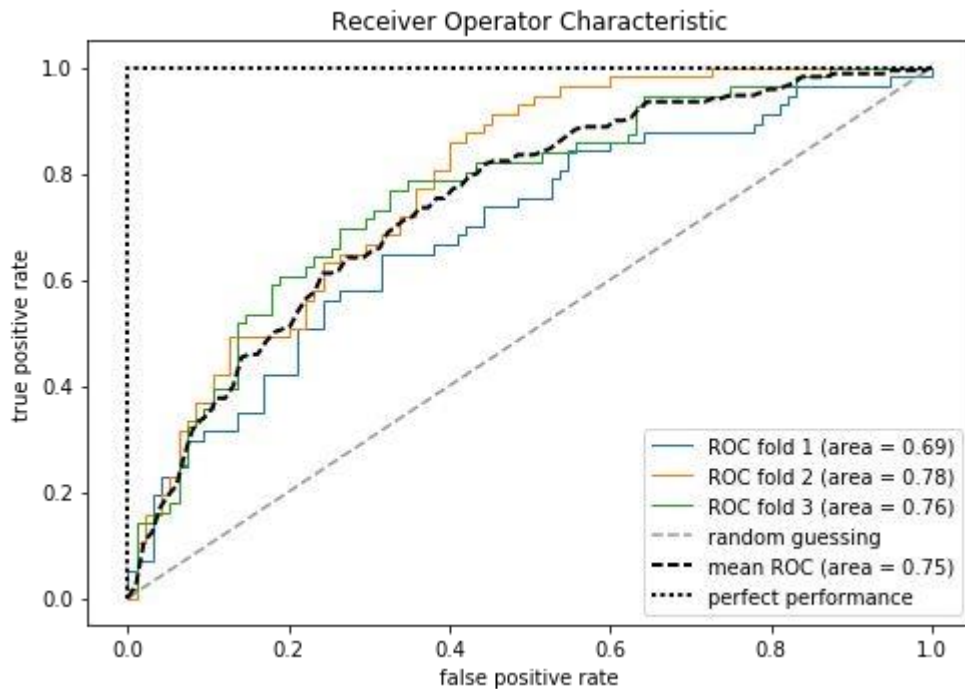
# Curva ROC: peor caso

- Representa la SEN vs 1-ESP (Tasa de Falsos Positivos) cuando desplazo el umbral



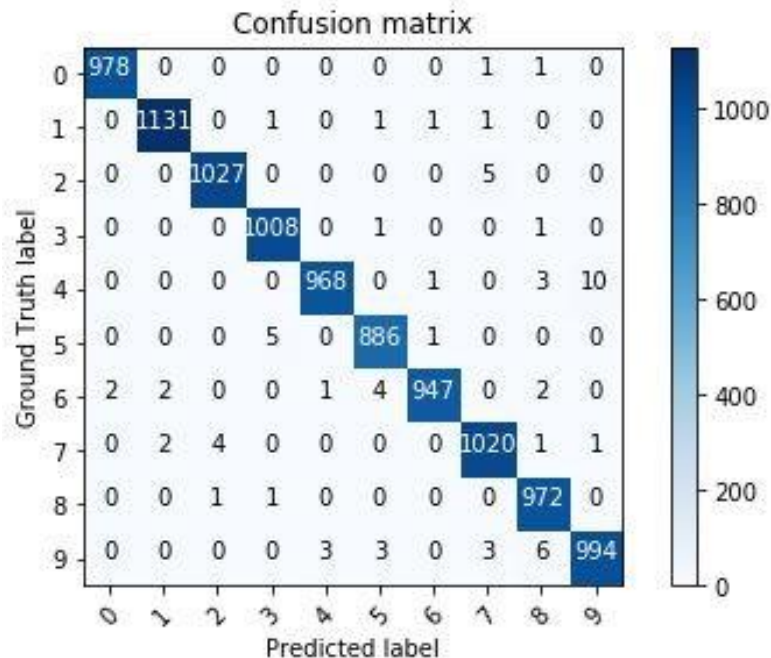
# Curva ROC: utilidad

- Es un método interesante para comparar clasificadores



# ■ Clasificación multiclase

- Podemos calcular la matriz de confusión igualmente
  - Análisis de errores



# ■ Métricas en sklearn

- Podéis consultar la [documentación](#).





# Índice

1. Métricas en clasificación
2. **Problemas desbalanceados**
3. Métricas en regresión



# ■ Problemas desbalanceados

- ¿Qué pasa si la proporción de muestras  $y = \{1/0\}$  es 90/10% y nuestro clasificador tiene una  $ACC = 0.9$ ?
  - Decimos que estamos ante un problema desbalanceado cuando la proporción de una clase es mucho mayor que la proporción de la otra
    - Fraude: 0.1 %
      - Detección de anomalías
    - Fuga: 5-15%
- ¿Cómo entrenamos un clasificador en estas condiciones?
  - La ACC no nos sirve como métrica



# ■ Estrategias

- Utilizar un conjunto de métricas que ponderen la clases
  - FSC
  - $\text{Balanced Error Rate} = 1 - 0.5(\text{SEN} + \text{ESP})$
- Penalizar más los errores en la clase minoritaria: [class weight](#)
- Modificar el conjunto de entrenamiento para balancearlo
  - Sobremuestrear clase minoritaria
  - Crear muestras sintéticas de la clase minoritaria: [SMOTE](#)
  - Bajomuestrear clase mayoritaria



# Índice

1. Métricas en clasificación
2. Problemas desbalanceados
3. **Métricas en regresión**



# Regresión

- Mean Squared Error

$$\text{MSE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} (y_i - \hat{y}_i)^2.$$

- Mean Absolute Value

$$\text{MAE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} |y_i - \hat{y}_i|.$$

- Root Mean Squared Error

$$\text{RMSE}(y, \hat{y}) = \sqrt{\text{MSE}(y, \hat{y})}$$

- $R^2$

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{n_{\text{samples}}-1} (y_i - \hat{y}_i)^2}{\sum_{i=0}^{n_{\text{samples}}-1} (y_i - \bar{y})^2}$$



# ■ Referencias

- Introduction to Statistical Learning.
  - Capítulo 4, Sección 4.4.3
- Hands On Machine Learning.
  - Capítulo 3
- Documentación scikit-learn



Let's code!

