

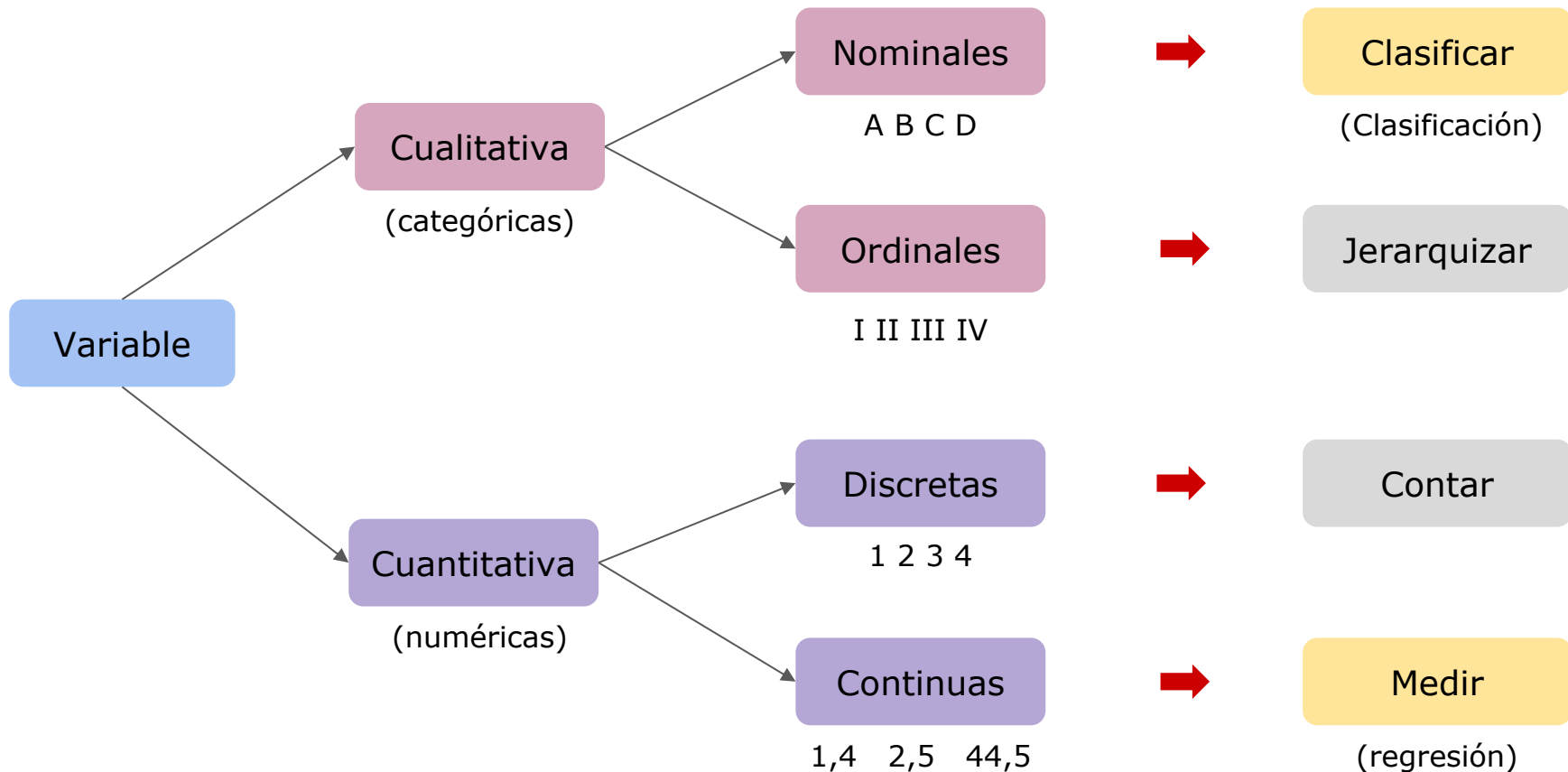


Ingeniería de datos

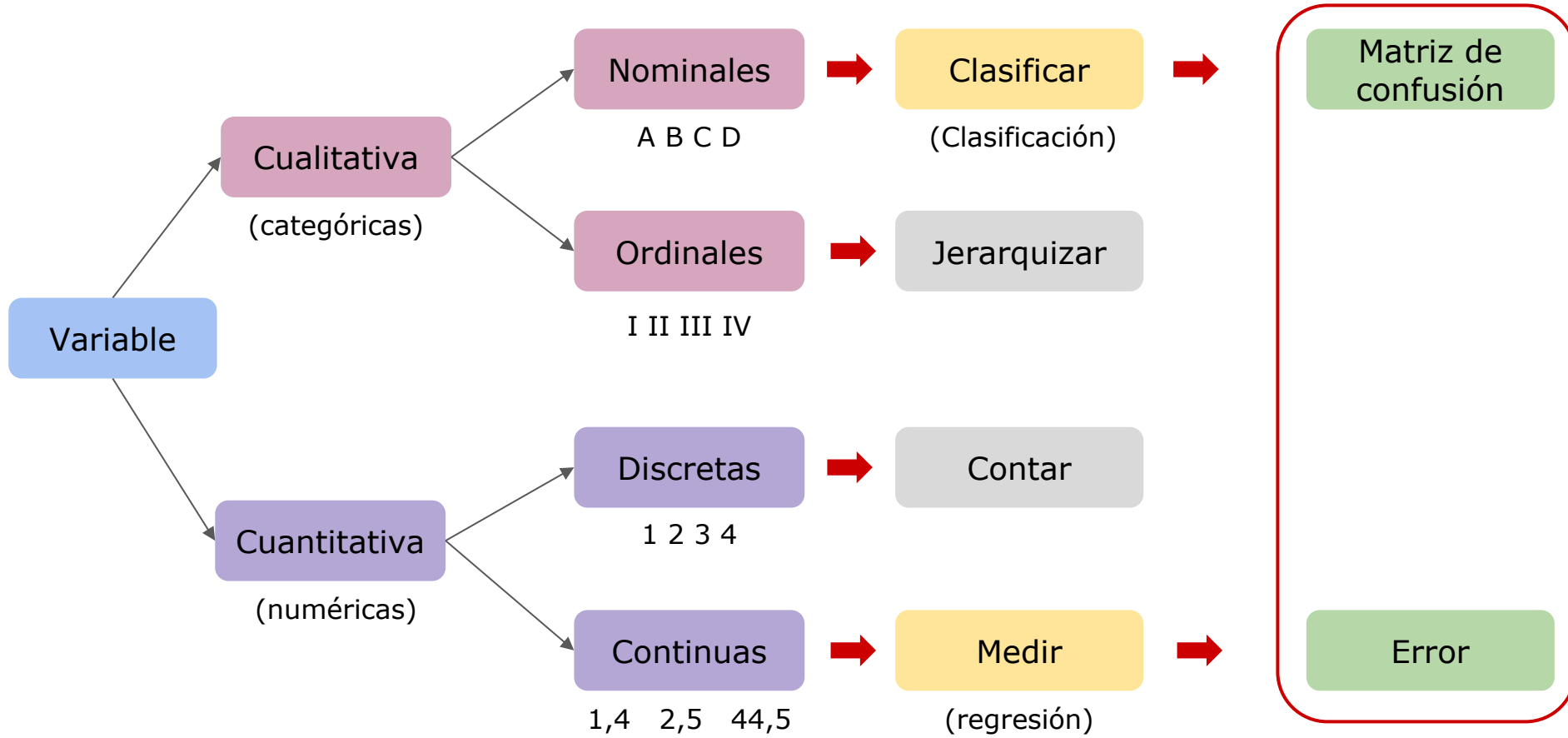
Pablo Valenzuela

pablo.valenzuela@ufrontera.cl

Tipos de datos

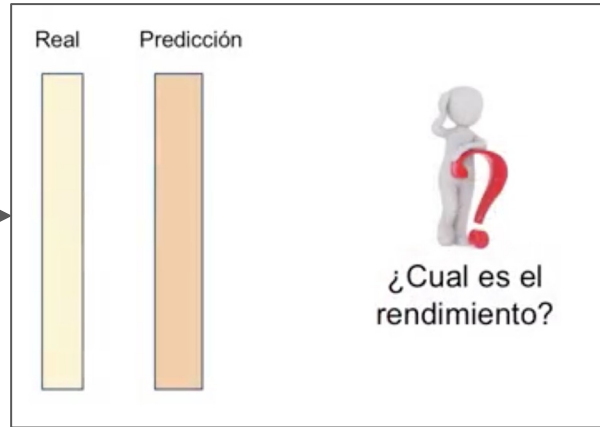


Métricas



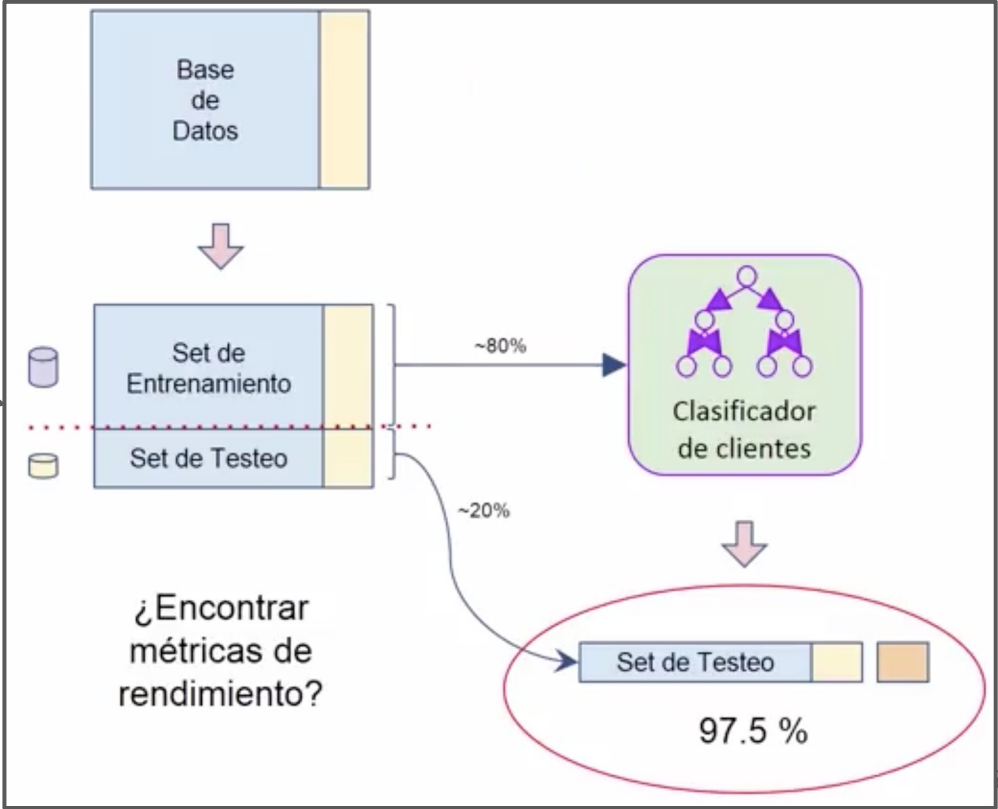
Métricas

Rendimiento



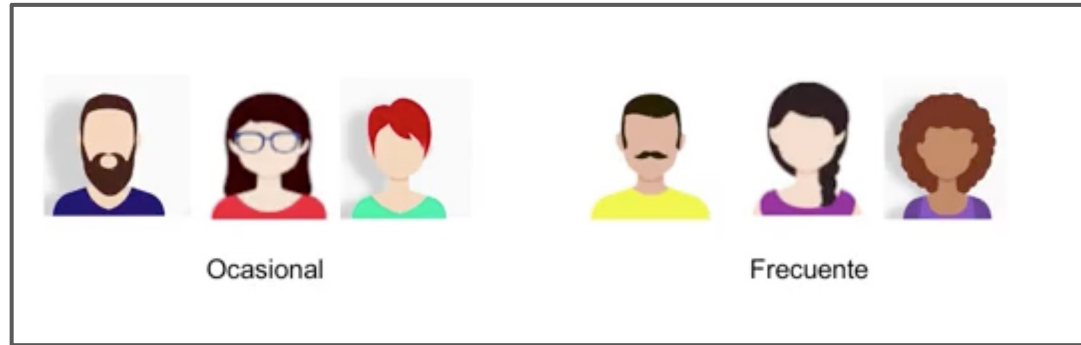
Métricas

Rendimiento



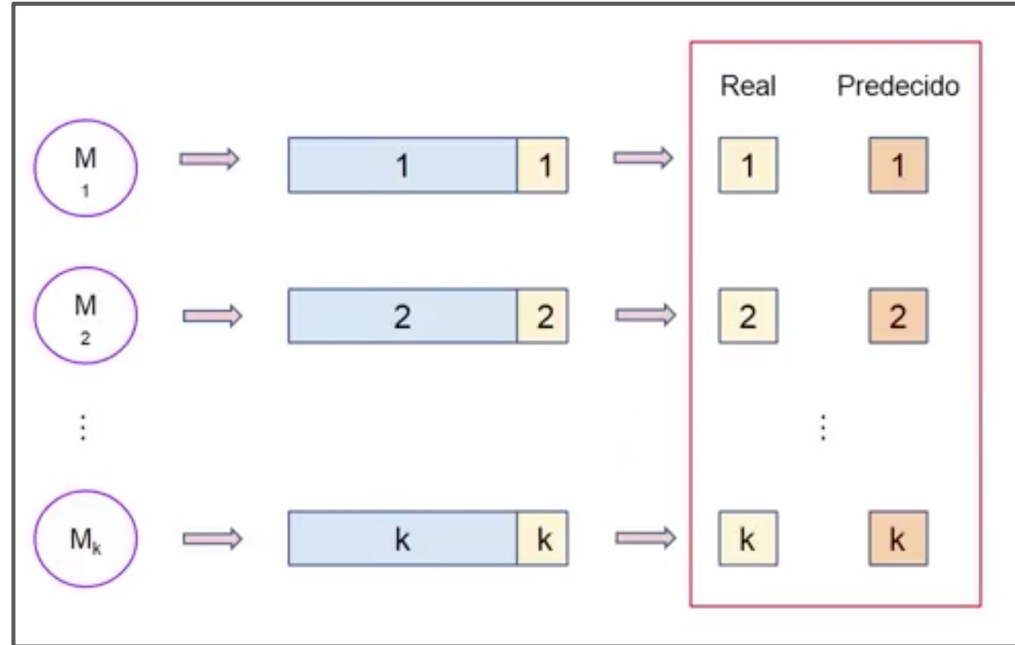
Métricas

Supongamos que tenemos una base de datos con dos tipos de clientes













Métricas

Supongamos además que
ajustamos un modelo para
predecir su tipo



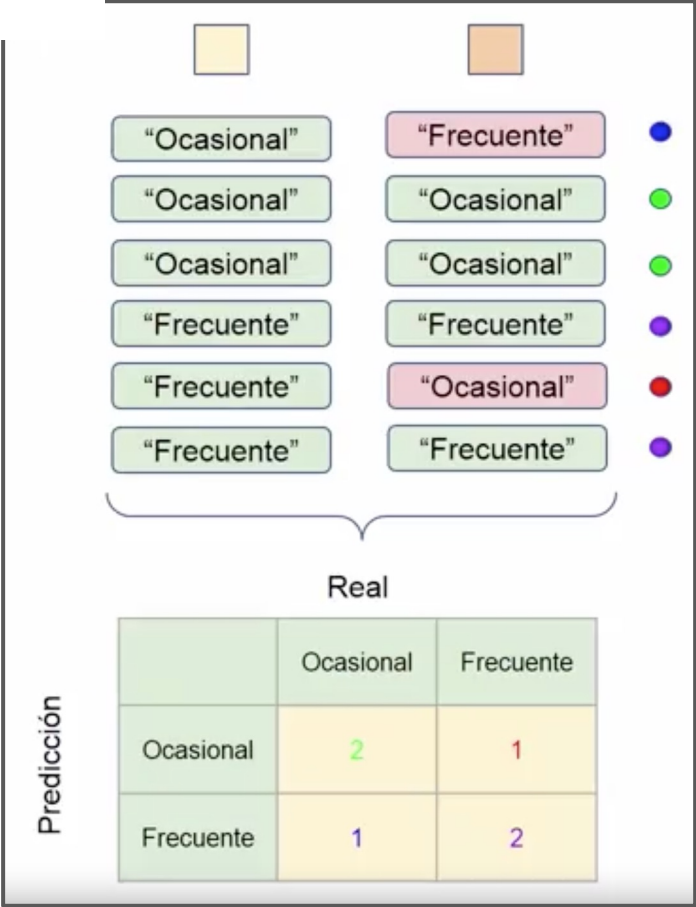
Métricas

- Supongamos que se obtienen las predicciones de clase para cada uno de los clientes
 - Verde las correctas
 - Rojo las incorrectas

			
			
	Ocasional	Frecuente	✗
	Ocasional	Ocasional	✓
	Ocasional	Ocasional	✓
	Frecuente	Frecuente	✓
	Frecuente	Ocasional	✗
	Frecuente	Frecuente	✓

Métricas

Los resultados numéricos se ingresan en la matriz de confusión



Métricas

Una matriz de confusión es una tabla utilizada para describir el rendimiento de un modelo de clasificación en un conjunto de datos de prueba para los cuales se conocen los valores reales

Orden Alfabético			
		Predicted: NO	Predicted: YES
Actual:	NO	50	10
	YES	5	100

¿Qué podemos aprender de esta matriz?

Métricas

¿Qué podemos obtener de esta matriz?

- Hay dos posibles clases: **"YES"** y **"NO"** (Usualmente se denominan como: **caso +** y **caso -**).
- El clasificador hizo un **total de 165 predicciones** (por ejemplo, 165 pacientes estaban siendo evaluados para detectar la presencia de una enfermedad).
- De esos 165 casos, el **clasificador predijo "YES" 110 veces y "NO" 55 veces**.
- En realidad, **105 pacientes de la muestra tienen la enfermedad y 60 pacientes no la tienen**.

n=165	Predicted: NO	Predicted: YES
	Actual: NO	Actual: YES
	50	10
	5	100

Métricas

- **True positives (TP):** casos en los que **se predijo que sí** (tienen la enfermedad), **y sí la tienen**.
- **True negatives (TN):** casos en que **predijimos que no**, y ellos **no tienen la enfermedad**.
- **False positives (FP):** casos en que **predijimos que sí**, pero en realidad **no tienen la enfermedad**. (También conocido como "**Error de tipo I**".)
- **False negatives (FN):** casos en que **predijimos que no**, pero en realidad **sí tienen la enfermedad**. (También conocido como "**Error de tipo II**").

Casos que buscamos maximizar

n=165	Predicted: NO	Predicted: YES
Actual: NO	50 TN	10
Actual: YES	5 FN	100

Todos los valores son valores que indican cantidad. No %, No rangos, etc.

Métricas

Lista de **indicadores** que a menudo se calculan a partir de una matriz de confusión para un clasificador binario:

Accuracy (Exactitud): ¿con qué frecuencia es correcto el clasificador?	$(TP+TN)/total = (100+50)/165$ $= 0.91 = \mathbf{91\%}$
Misclassification Rate (Clasificación errónea): en general, ¿con qué frecuencia la clasificación está mal?	$(FP+FN)/total = (10+5)/165 = 0.09 = \mathbf{9\%}$ Equivalente a 1 menos Accuracy También conocido como "tasa de error"
True Positive Rate (Tasa + verdadera): Cuando en realidad es YES, ¿con qué frecuencia predice YES?	$TP/actual\ YES = 100/105 = 0.95 = \mathbf{95\%}$ También conocido como "Sensitivity" o "Recall"
False Positive Rate (Tasa + falsa): cuando en realidad es NO, ¿con qué frecuencia predice YES?	$FP/actual\ NO = 10/60 = 0.17 = \mathbf{17\%}$
True Negative Rate (Tasa - verdadera): cuando en realidad es NO, ¿con qué frecuencia predice NO?	$TN/actual\ NO = 50/60 = 0.83 = \mathbf{83\%}$ Equivalente a 1 menos False Positive Rate También conocido como "Specificity"

Métricas

Lista de **indicadores** que a menudo se calculan a partir de una matriz de confusión para un clasificador binario:

Precision (Precisión): cuando predice YES, ¿con qué frecuencia es correcto?	TP/predicted YES = $100/110 = 0.91 = 91\%$
Prevalence (Prevalencia): ¿Con qué frecuencia se produce realmente la condición YES en nuestra muestra?	real YES / total = $105/165 = 0.64 = 64\%$

Métricas

¿Cómo elegir la métrica de evaluación correcta para un problema determinado?

Depende de los objetivos de cada proyecto.

- **Ejemplo 1:**

- Filtro Spam: Optimizar para Precisión, porque FN (spam en el inbox), “es más aceptable que” FP (correos no spam, son capturados por el filtro spam).

- **Ejemplo 2:**

- Detector de transacciones fraudulentas (clase positiva es “fraude”): Optimizar para Recall (sensibilidad), porque FP (una transacción normal que es clasificada como fraudulenta) “es más aceptable que” FN (transacciones fraudulentas no detectadas)

- **En general lo que se hace es:**

- Evaluar qué error minimizar
- Elegir la métrica de acuerdo a la evaluación previa

Métricas

¿Por qué la precisión es a menudo una métrica engañosa?

Ejemplo 1: ¿es útil proponer este rango en un problema de clasificación binaria?

- 1.0: Predicción perfecta
- 0.9: Predicción excelente
- 0.8: Predicción buena
- 0.7: Predicción mediocre
- 0.6: Predicción pobre
- 0.5: Predicción aleatoria
- <0.5 : Algo está mal!

Rangos son inútiles!!

Métricas

¿Por qué la precisión es a menudo una métrica engañosa?

Ejemplo 2: Suponga una compañía de tarjetas de crédito que necesita un detector de transacciones fraudulentas.

- Si presenta un modelo que tiene un 99% en accuracy es insuficiente
- Es posible obtener un 99% en accuracy sin esfuerzo al, por ejemplo, predecir que no existe fraude
 - Esto se conoce como “class imbalance”: Hay que preguntar qué considera la compañía como suficiente

Métricas

¿Por qué la precisión es a menudo una métrica engañosa?

Ejemplo 3: Suponga una compañía de inversiones

- La compañía necesita un modelo de predicción de acciones que permita saber si estas subirán su valor el día siguiente
- Si se presenta un modelo con un 60% en precisión, será millonario en una semana
- Por lo tanto:

El valor de precisión depende del problema

Referencias

Referencias

1. Géron, A. (2017). Hands-on machine learning with Scikit-Learn and TensorFlow: concepts, tools, and techniques to build intelligent systems. " O'Reilly Media, Inc.".
2. Formulate Your Problem as an ML Problem. (2020). Retrieved 2020, from <https://developers.google.com/machine-learning/problem-framing/formulate>