

Conceptos generales de Machine Learning

Rafael Zambrano

rafazamb@gmail.com

¿Qué es Machine Learning?

● ¿Qué es un perro?



→ SÍ



→ NO



→ SÍ



→ NO



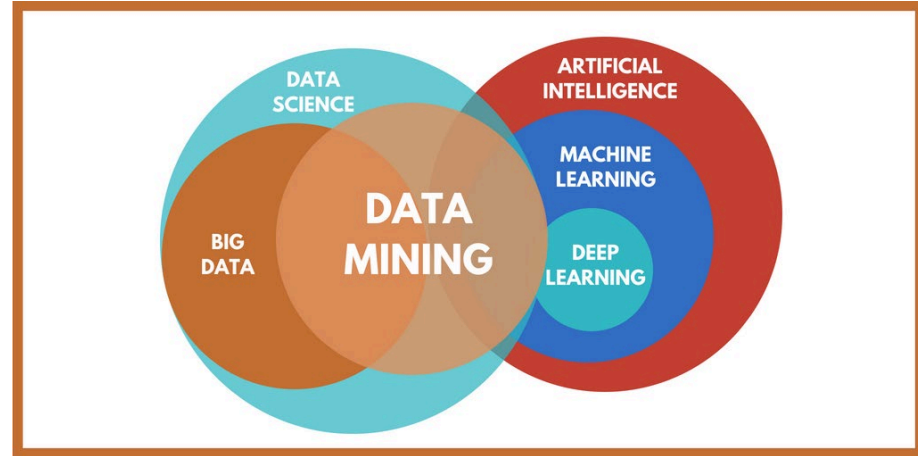
→ SÍ



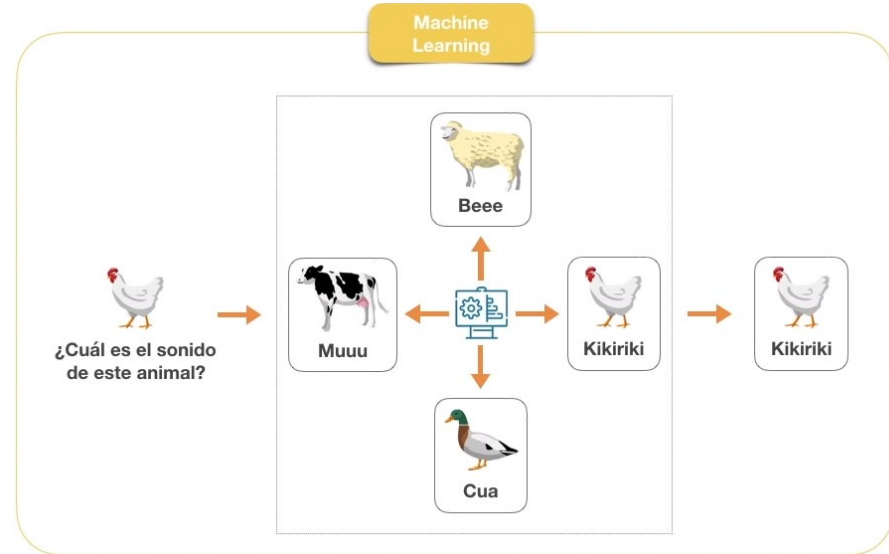
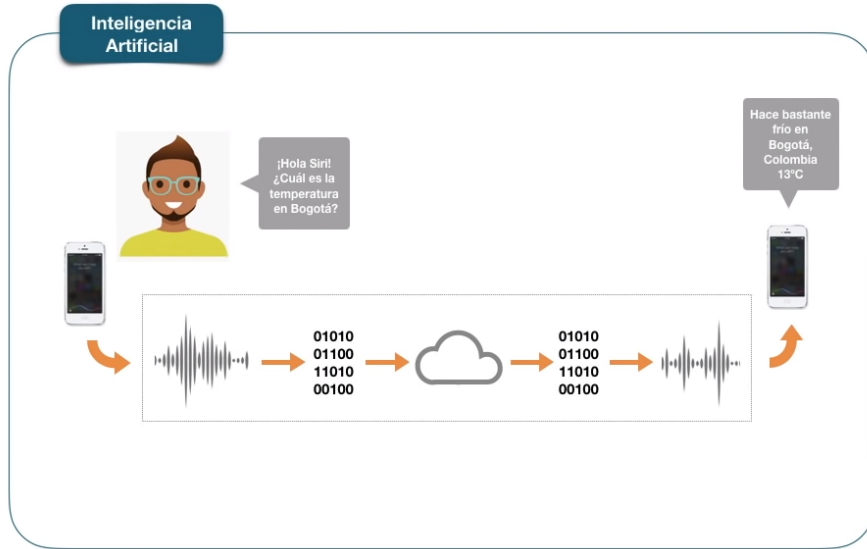
“El ser humano posee mucho más conocimiento del que consigue expresar”

¿Qué es Machine Learning?

- Conjunto de algoritmos capaces de identificar y **aprender** patrones en **datos** para realizar predicciones
- Otros conceptos:
 - **Big Data:** almacenamiento y procesamiento de grandes volúmenes de datos
 - **Minería de datos** (data mining): Extraer valor de los datos
 - **Inteligencia Artificial:** Resolver problemas mediante máquinas, mejorando al ser humano
 - **Deep Learning:** Algoritmos que funcionan de forma similar al cerebro humano a través de redes neuronales

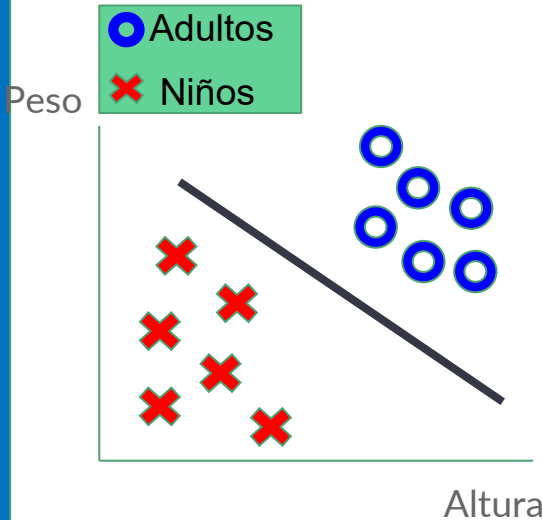


¿Qué es Machine Learning?

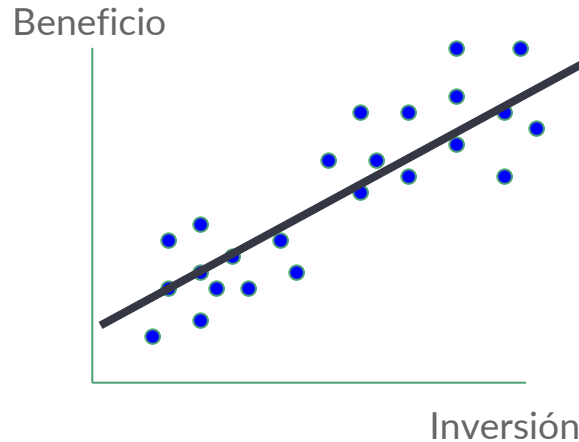


Técnicas de Machine Learning

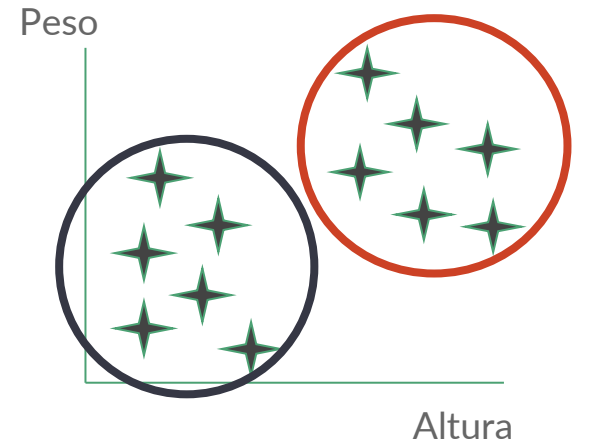
Clasificación



Regresión



Agrupación (*clustering*)

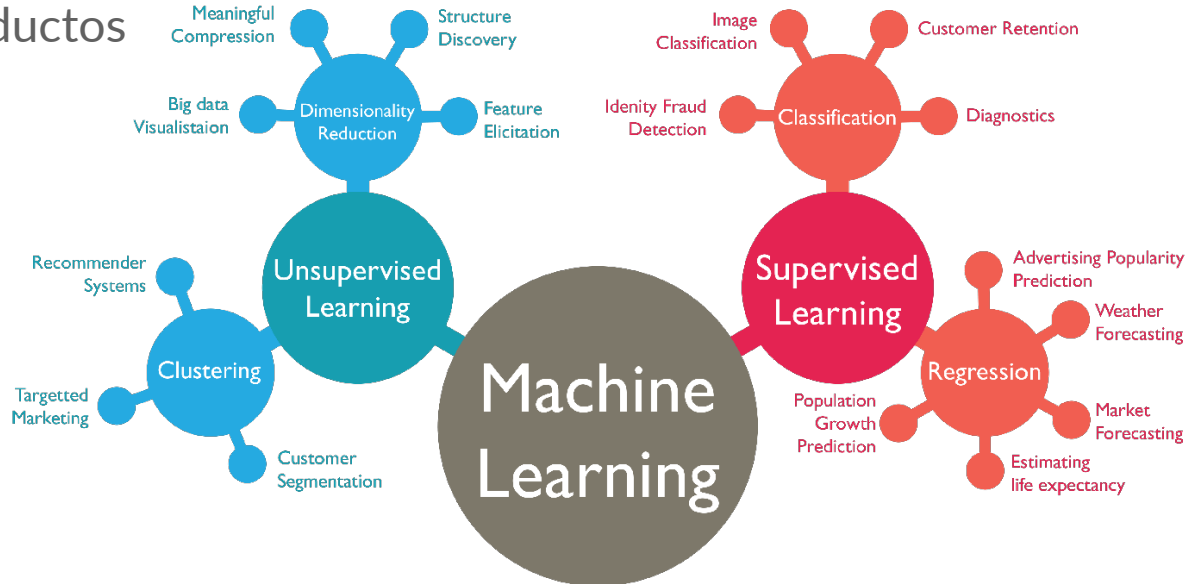


APRENDIZAJE SUPERVISADO

APRENDIZAJE NO SUPERVISADO

Aplicaciones de Machine Learning

- Reconocimiento de imágenes
- Filtros de spam
- Detección de fraude
- Recomendaciones de productos
- Diagnósticos médicos
- Personalización web
- Reconocimiento de voz
- Optimización de precios
- Fugas y cross-selling
- Segmentación de clientes
- Demanda de energía
- Trading
- Etc.



Algoritmos de Machine Learning

- Regresión Lineal
- Regresión Logística
- Árboles de Decisión
- Random Forest
- XGBoost
- Gradient Boosting
- Isolation Forest
- Redes Neuronales
- Support Vector Machines
- Algoritmos de agrupación (k-means)
- Etc.

¿Cuándo usar Machine Learning?

- Cuando un problema está determinado con fórmulas precisas, no es necesario utilizar Machine Learning
- Utilizaremos Machine Learning cuando no se pueden obtener fácilmente las ecuaciones que resuelven nuestro problema

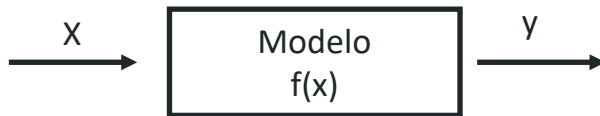


Modelo

Concepto de modelo

Un modelo es un conjunto de fórmulas matemáticas que expresan relaciones y patrones entre variables, utilizados para estudiar comportamientos de sistemas complejos ante situaciones difíciles de observar en la realidad.

Es una manera de traducir a la matemática relaciones entre datos de la vida real.



Variables que me digan la predisposición que tienen los pacientes a contraer cáncer de pulmón

Variables que me indiquen si un cliente es buen o mal pagador para un banco.

Qué producto estaría dispuesto a comprar un usuario X de Facebook, en función de sus gustos y búsquedas.

Con un modelo puedo predecir cierto output, como por ejemplo si el paciente contrae o no cáncer de pulmón, a partir de un conjunto de inputs, que podrían ser edad, fumador, enfermedades previas...

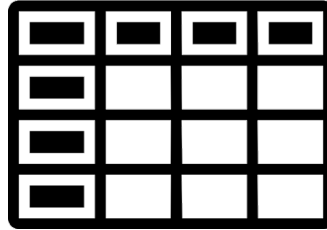
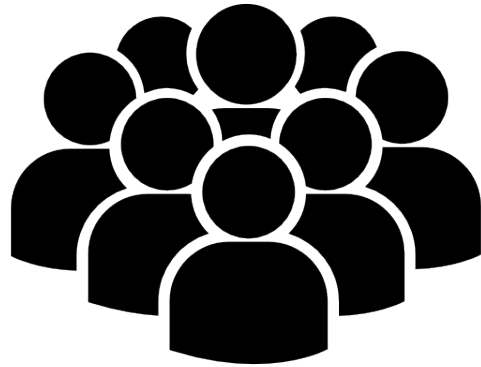
Ejemplo: Predicción de ingresos (Regresión)

- Desde una compañía de seguros, se quiere estimar los ingresos de los clientes. A pesar de ser un dato desconocido 1000 clientes acceden, a través de una encuesta, a proporcionar el dato de sus ingresos
- Se debe contar con datos históricos y extraer todas las variables que puedan tener relación con la variable objetivo (target)

[illegible]

Ejemplo: Predicción de ingresos (Regresión)

- Entrenamiento

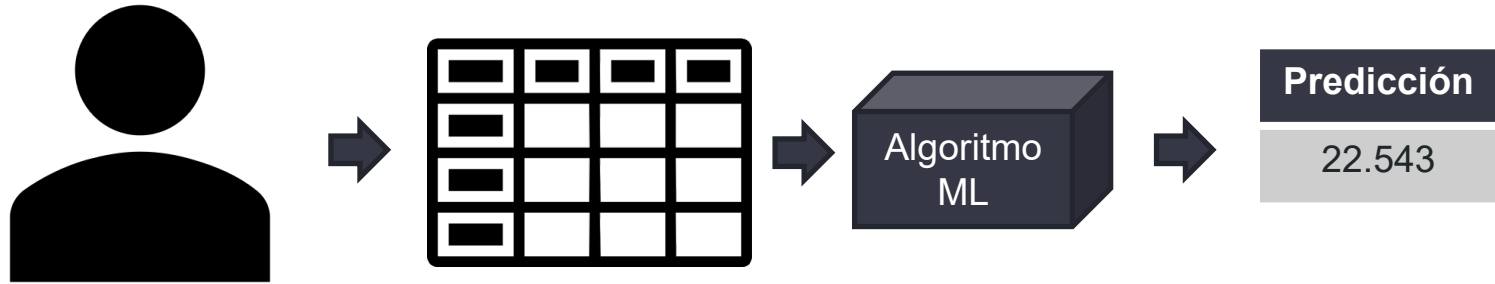


Real	Predicción
32.000	31.918
23.000	28.913
85.000	97.611
...	...



Ejemplo: Predicción de ingresos (Regresión)

- Predicción con nuevos datos



Cómo desarrollar un modelo

1. Entender y definir el problema

- Público objetivo y target
- Datos disponibles
- Tipo de problema
- Aplicación

2. Recopilar datos

- Disponibilidad

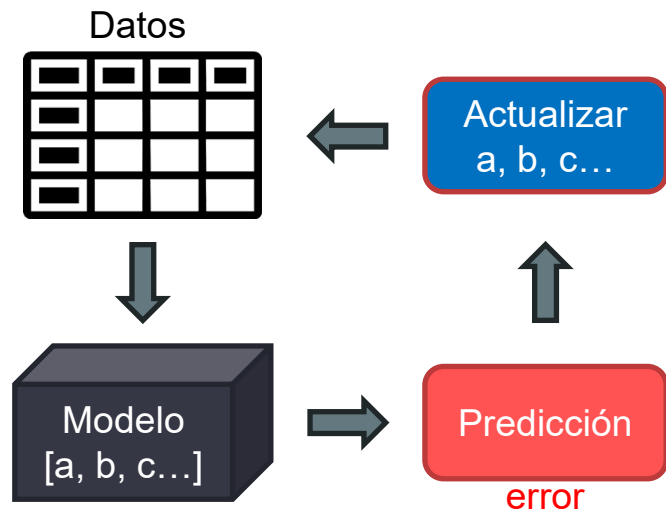
3. Preparar los datos

- Limpieza, valores atípicos, transformaciones...

4. Dividir los datos (entrenamiento, validación y test)

5. Entrenar uno o varios modelos

6. Medir el desempeño

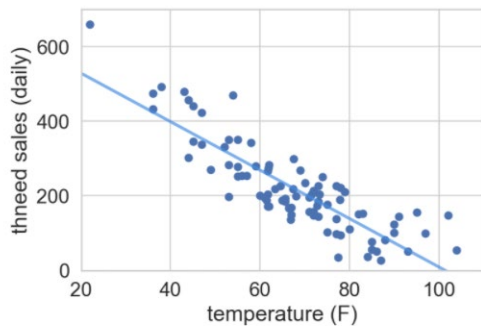


Modelo

¿Cuál es el mejor modelo?

Modelo 1

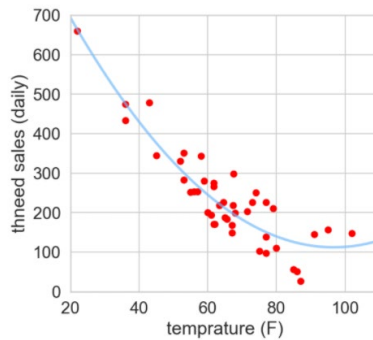
Regresión lineal



Error = 63.0

Modelo 1

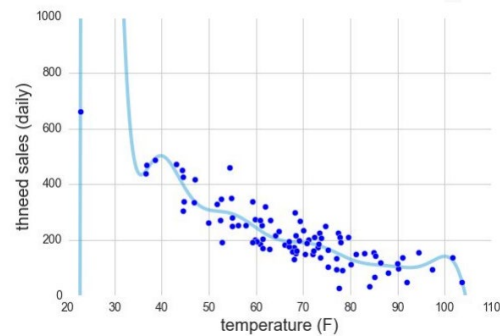
Regresión polinómica grado 2



Error = 51.5

Modelo 1

Regresión polinómica grado > 2



Error = 49.2

¡Bingo! Parece que he encontrado el mejor modelo

¡Ya tenemos el modelo! Vamos a
ponerlo en producción

Objetivo de un modelo

Generalizar ante datos nuevos

Error = 87.9



¿Problema?

*El modelo no **generaliza** bien*

¿Solución?

Dividir en train/test. Guardo unos datos de test, ajenos completamente al entrenamiento. Y pruebo el modelo entrenado, a ver qué tal generaliza con datos nuevos



Entrenamiento, validación y test

- En general, fragmentaremos (aleatoriamente) los datos en tres conjuntos: entrenamiento, validación y test
- El modelo **aprende** con los datos de entrenamiento, **evalúa** con los de validación y **prueba** con los datos de test
- Los datos de entrenamiento deben ser **representativos**
- El conjunto de validación suele incluirse con el de entrenamiento, siendo el propio modelo el que selecciona un conjunto de validación para ajustar los hiperparámetros del modelo
- Los datos de test permiten conocer si el modelo generaliza bien con datos desconocidos para él
- Normalmente, la división es de los datos es del 80% para entrenamiento y validación, y 20% para test

Entrenamiento (60%)

Validación (20%)

Test (20%)

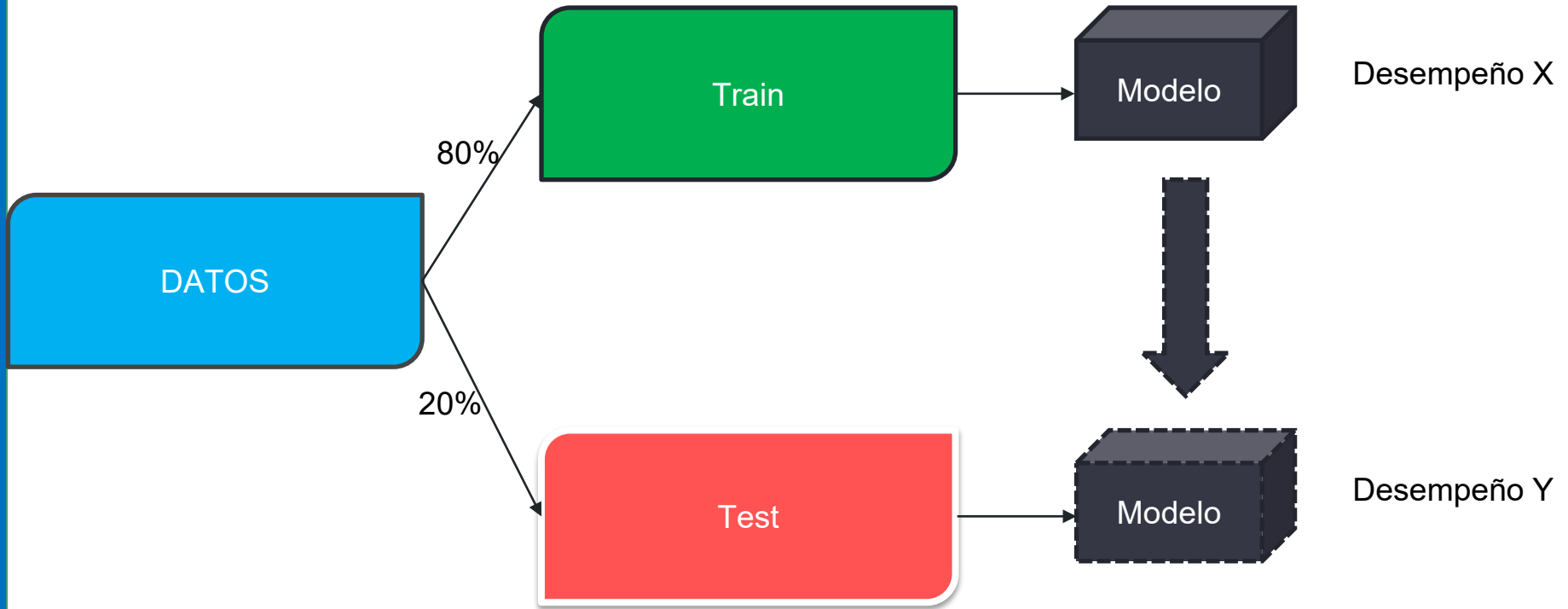
Entrenamiento, validación y test

Entrenamiento y
Validación (80%)

Test (20%)

	FTHG	FTAG	FTR	HTHG	HTAG	HTR	HS	AS	HST	AST	HF	AF	HC	AC	HY	AY	HR	AR	B365H	B365D	B365A	BWH	BWD	BWA	IWH	IWD	IWA	LBH	LBD	LBA	PSH	PSD	PSA	WHH	WHD	WHA	SJH	SJD	SJA
2	0	H		1	0	H	16	15	6	2	13	6	6	5	1	1	0	0	1,73	3,6	4,75	1,72	3,6	4,75	1,7	3,6	4,7	1,66	3,6	5	1,79	3,74	5,12	1,75	3,75	4,5	1,67	3,6	5,5
1	0	H		0	0	D	9	11	1	2	15	23	9	6	3	5	0	0	1,53	4	6	1,57	4	5,5	1,55	3,9	5,6	1,57	3,75	6	1,56	4,33	6,78	1,55	4	6	1,57	4	6
1	2	A		1	1	D	8	13	2	3	10	8	5	5	1	0	0	0	2,5	3,3	2,8	2,6	3,3	2,65	2,4	3,3	2,75	2,6	3,3	2,6	2,69	3,35	2,86	2,5	3,3	2,8	2,63	3,25	2,75
7	0	H		6	0	H	22	4	13	1	15	16	9	3	1	3	0	0	1,08	10	26	1,08	10,5	23	1,1	8	20	1,08	10	23	1,09	14	30	1,08	11	23	1,09	9,5	29
1	2	A		0	2	A	14	13	5	4	15	17	7	6	1	4	0	0	2	3,3	3,75	2	3,3	3,8	2	3,3	3,6	2	3,3	3,5	2,13	3,37	3,95	2,15	3,25	3,5	2	3,4	3,8
2	1	H		1	1	D	20	11	9	4	11	20	5	7	1	2	0	0	1,17	7	17	1,16	7,75	13,5	1,17	6,5	14	1,15	6,5	15	1,16	9	19,3	1,17	7,5	15	1,17	7	17
1	3	A		1	1	D	14	16	5	6	12	13	1	9	4	4	0	0	2,8	3,3	2,5	2,7	3,4	2,5	2,6	3,2	2,6	2,75	3,3	2,5	2,89	3,39	2,64	2,8	3,3	2,5	2,63	3,4	2,63
2	3	A		1	0	H	15	14	2	4	17	14	6	10	2	0	0	0	2,6	3,2	2,75	2,55	3,25	2,75	2,5	3,3	2,65	2,37	3,2	2,87	2,7	3,32	2,87	2,5	3,2	2,9	2,5	3,25	2,88
2	2	D		1	0	H	15	6	10	5	23	12	7	2	4	4	0	0	2,2	3,2	3,4	2,15	3,4	3,3	2,2	3,3	3,1	2,2	3,3	3,25	2,23	3,33	3,69	2,15	3,25	3,5	2,2	3,3	3,4
3	0	H		2	0	H	11	8	5	1	16	11	3	4	2	2	0	0	2,25	3,25	3,25	2,2	3,3	3,25	2,1	3,3	3,3	2,2	3,3	3,3	2,31	3,42	3,39	2,25	3,3	3,2	2,25	3,25	3,3
2	0	H		1	0	H	8	18	2	5	15	20	5	10	2	2	0	0	1,62	3,75	5,5	1,62	3,7	5,75	1,7	3,6	4,7	1,66	3,6	5,5	1,65	4,14	5,82	1,67	3,6	5,5	1,62	4	5,5
2	2	D		1	2	A	14	16	5	7	15	17	2	7	2	1	0	0	1,85	3,5	4,2	1,8	3,4	4,6	1,85	3,45	4	1,85	3,5	4,2	1,91	3,63	4,54	1,83	3,5	4,4	1,83	3,6	4,33
1	1	D		1	0	H	12	11	2	4	9	12	4	2	4	3	0	0	3,2	3,4	2,2	3,4	3,3	2,15	3	3,3	2,25	3,2	3,3	2,25	3,32	3,47	2,27	3,2	3,3	2,25	3,13	3,5	2,25
3	1	H		1	1	D	13	8	3	2	17	12	6	3	3	3	0	0	3	3,25	2,38	3,1	3,3	2,25	2,9	3,3	2,3	3,1	3,3	2,3	3,18	3,43	2,41	3,2	3,3	2,25	3	3,4	2,38
2	1	H		1	1	D	17	6	6	2	11	25	8	3	4	2	0	0	1,85	3,6	4	1,85	3,4	4,33	1,85	3,45	4	1,9	3,4	4	1,88	3,7	4,6	1,85	3,5	4,33	1,83	3,5	4,5
5	0	H		3	0	H	14	7	5	1	11	9	8	3	0	1	0	0	1,29	5,25	10	1,34	4,75	9,25	1,35	4,8	7,6	1,33	5	9	1,32	5,6	11,58	1,33	5	9	1,3	5,25	11
1	2	A		0	0	D	20	13	6	4	9	18	12	8	1	3	0	1	1,8	3,5	4,5	1,72	3,7	4,75	1,7	3,7	4,5	1,7	3,6	5	1,86	3,64	4,78	1,75	3,6	4,75	1,73	3,6	5
0	0	D		0	0	D	8	18	3	3	17	18	5	7	3	5	0	0	3,5	3,3	2,1	3,5	3,25	2,1	2,85	3,3	2,35	3,2	3,25	2,25	3,54	3,45	2,21	3,4	3,3	2,15	3,4	3,3	2,2
0	1	A		0	1	A	9	15	3	11	14	12	2	11	3	2	0	0	13	6	1,22	14	6,25	1,2	10,3	5,5	1,25	12	6	1,22	13,7	6,78	1,25	12	7	1,2	15	6,5	1,2
0	1	A		0	1	A	8	21	3	8	14	10	5	8	4	2	0	0	7,5	5,5	1,33	9,25	5,25	1,3	10	5,2	1,27	9	5,5	1,3	9,11	5,52	1,37	8	5,5	1,33	10	5,5	1,3
2	2	D		2	1	H	9	15	5	9	18	18	4	5	3	5	1	0	2	3,3	3,8	2	3,3	3,8	2,1	3,3	3,3	2,05	3,4	3,5	2,07	3,68	3,79	2,05	3,4	3,6	2,05	3,4	3,8
1	2	A		0	1	A	23	6	8	2	14	17	8	1	4	3	1	0	1,73	3,6	4,75	1,8	3,7	4,2	2	3,3	3,6	1,8	3,5	4,5	1,83	3,74	4,8	1,83	3,75	4	1,83	3,6	4,5
1	1	D		0	0	H	17	3	4	1	9	6	6	5	1	2	0	0	2,2	3,3	3,3	2,2	3,4	2,1	3,3	3,3	2,2	3,3	3,3	2,23	3,47	3,53	2,25	3,2	3,3	2,15	3,3	3,6	
0	3	A		0	2	A	9	9	3	4	15	14	7	8	3	2	0	0	2,5	3,2	2,88	2,5	3,25	2,8	2,5	3,3	2,65	2,45	3,2	2,87	2,65	3,31	2,93	2,62	3,1	2,8	2,6	3,2	2,88
1	0	H		0	0	D	21	16	6	0	19	8	10	5	2	2	0	0	2,1	3,3	3,5	2,1	3,3	3,5	2	3,3	3,6	2,15	3,3	3,4	2,27	3,37	3,53	2,25	3,2	3,3	2,1	3,4	3,6
0	0	D		0	0	D	10	8	3	3	19	14	7	7	2	4	0	0	2,05	3,4	3,6	2	3,5	3,6	1,9	3,45	3,8	2,05	3,4	3,5	2,11	3,58	3,76	2,05	3,4	3,6	2	3,5	3,8
3	1	H		2	0	H	20	12	7	2	15	13	9	6	1	2	0	0	1,17	7	16	1,16	7,75	13,5	1,2	6,5	10,3	1,18	7,5	12	1,18	8,4	17,8	1,17	7,5	15	1,2	7	15
2	2	D		1	1	D	15	15	7	6	17	13	10	1	3	3	0	1	1,62	3,8	5,5	1,57	3,9	5,75	1,65	3,8	4,7	1,66	3,75	5	1,62	4,14	6,27	1,6	4	5,5	1,67	3,75	5,5
1	2	A		0	1	A	10	16	4	9	19	21	5	5	2	4	0	0	2,9	3,3	2,38	2,8	3,25	2,5	2,75	3,3	2,4	2,87	3,4	2,37	2,88	3,51	2,58	2,8	3,3	2,5	2,75	3,4	2,8
2	3	A		2	3	A	16	20	10	9	20	11	5	7	2	3	0	0	5,5	4,33	1,53	6,25	4,4	1,48	5	3,9	1,6	6	4,33	1,5	6,13	4,54	1,57	5,5	4,33	1,55	5,75	4,33	1,57
4	2	H		2	1	H	20	7	8	2	15	17	8	1	2	4	0	0	1,25	6	11	1,26	5,5	11	1,27	5,2	10	1,28	5,5	9,5	1,27	6,32	13,73	1,25	5,5	12	1,25	6,25	12
3	2	H		1	0	H	19	7	10	2	12	17	6	9	2	5	0	0	1,18	7	15	1,18	7	13,5	1,15	7	15	1,16	7,5	13	1,19	8,48	15,71	1,17	7,5	13	1,17	7,5	17

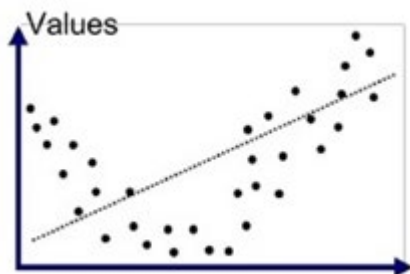
Evaluación de modelos



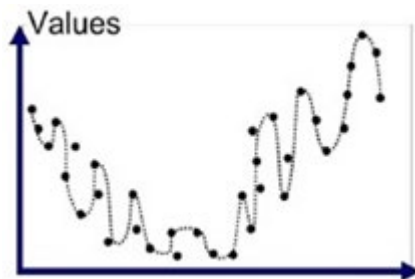
- Si $Y \simeq X \Rightarrow$ Buena generalización del modelo
- Si $Y \ll X \Rightarrow$ Overfitting

Overfitting y Underfitting

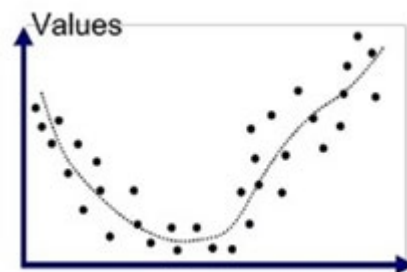
- A la hora de entrenar un modelo, existe un conflicto entre optimización y generalización
- Optimización es el proceso de ajuste de un modelo para conseguir el mejor rendimiento posible de los datos de entrenamiento (proceso de aprendizaje).
- Generalización es cómo de bien se comporta el modelos ante datos no procesados aún. El objetivo es conseguir la mejor capacidad de generalización.
- El modelo comienza a estar sobre-ajustado ó “overfitted” cuando ha aprendido tan bien los datos de entrenamiento, que memoriza pautas que son demasiado específicas de los datos de entrenamiento e irrelevantes para los nuevos datos.



Underfitting

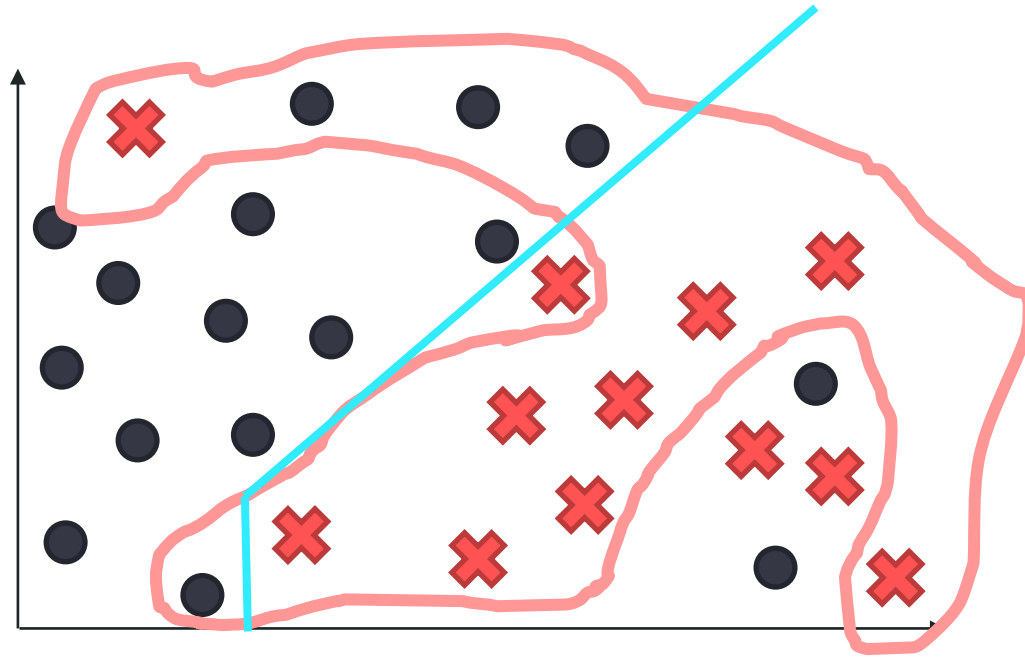


Overfitting



Buen ajuste

Overfitting y Underfitting



Overfitting y Underfitting

Bias

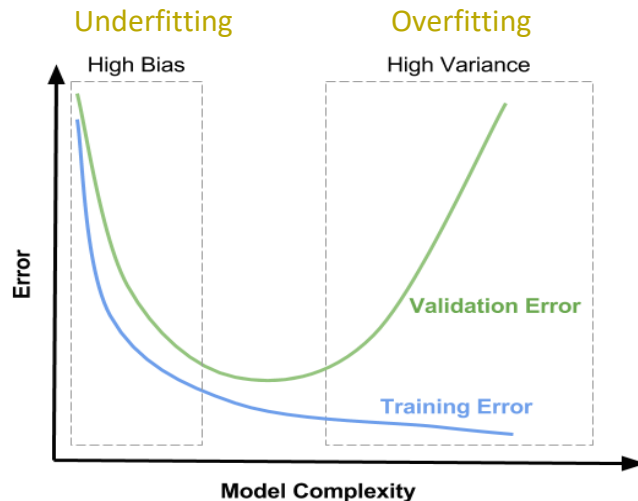
Capacidad de un modelo de ajustarse a los datos de entrenamiento. Si se ajusta de más, tendremos Overfitting, y por tanto un Variance alto.

Variance

Capacidad de un modelo para mantener sus regiones de decisión ante pequeñas variaciones de los datos.

¿Objetivo?

Reducir el bias y el variance al máximo. Con esto conseguimos tener un modelo que se ajuste al patrón de los datos y al mismo tiempo generalizar con datos futuros



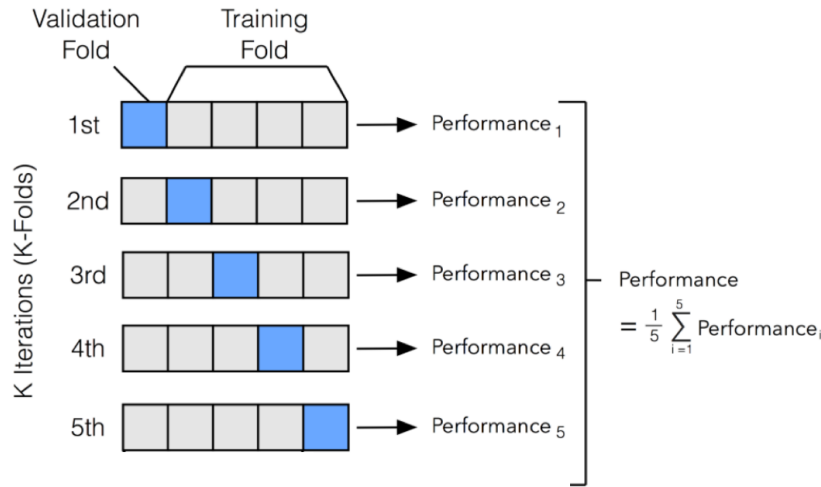
Overfitting y Underfitting

Formas de evitar el “overfitting”:

- **Obtener más datos** es normalmente la mejor solución, un modelo entrenado con más datos generalizará mejor de forma natural.
- La **regularización** es diferente en función del modelo y consiste en simplificarlo para evitar que “memorice” los datos de entrenamiento
- **Validación cruzada** (*Crossvalidation*)

Validación cruzada (crossvalidation)

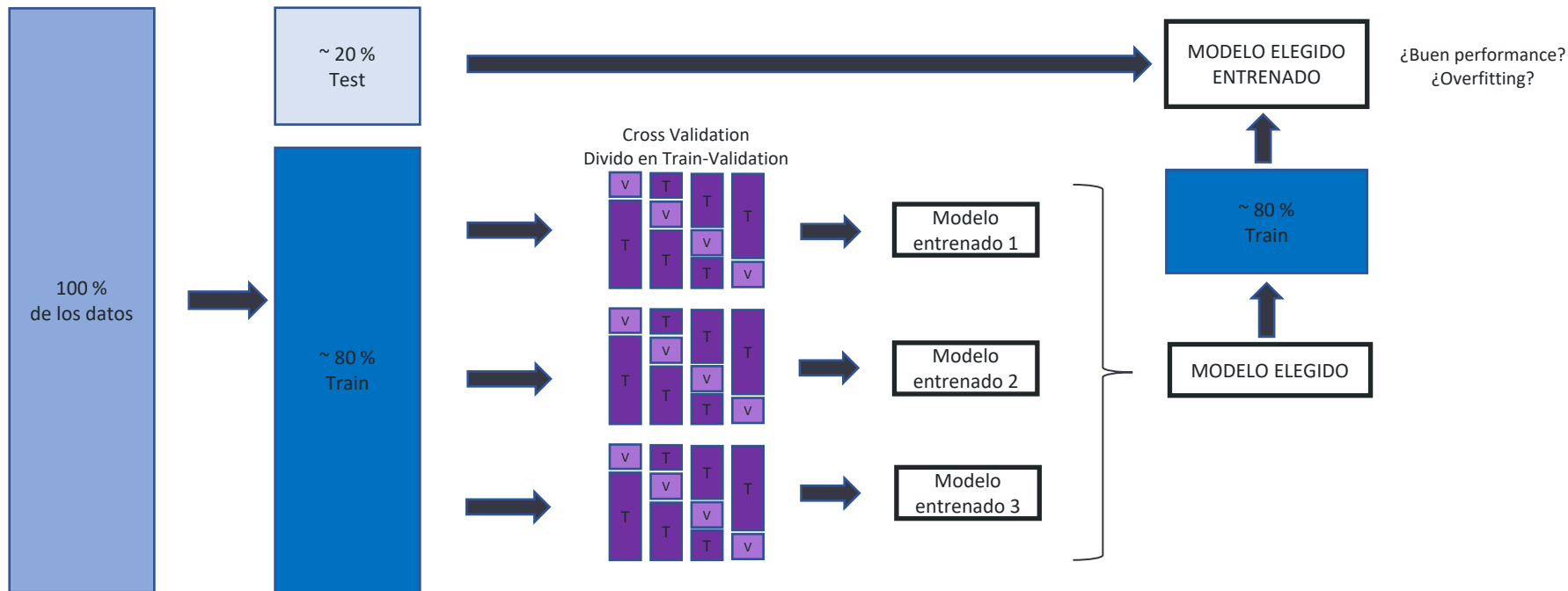
- Divide los datos de entrenamiento en K particiones del mismo tamaño
- Para cada partición p , el modelo entrena con las particiones restantes ($K-1$) y evalúa en la propia partición p
- La evaluación final es la media de las K evaluaciones obtenidas



Entonces con Cross Validation
ya no necesito dividir en train
test..**No exactamente**

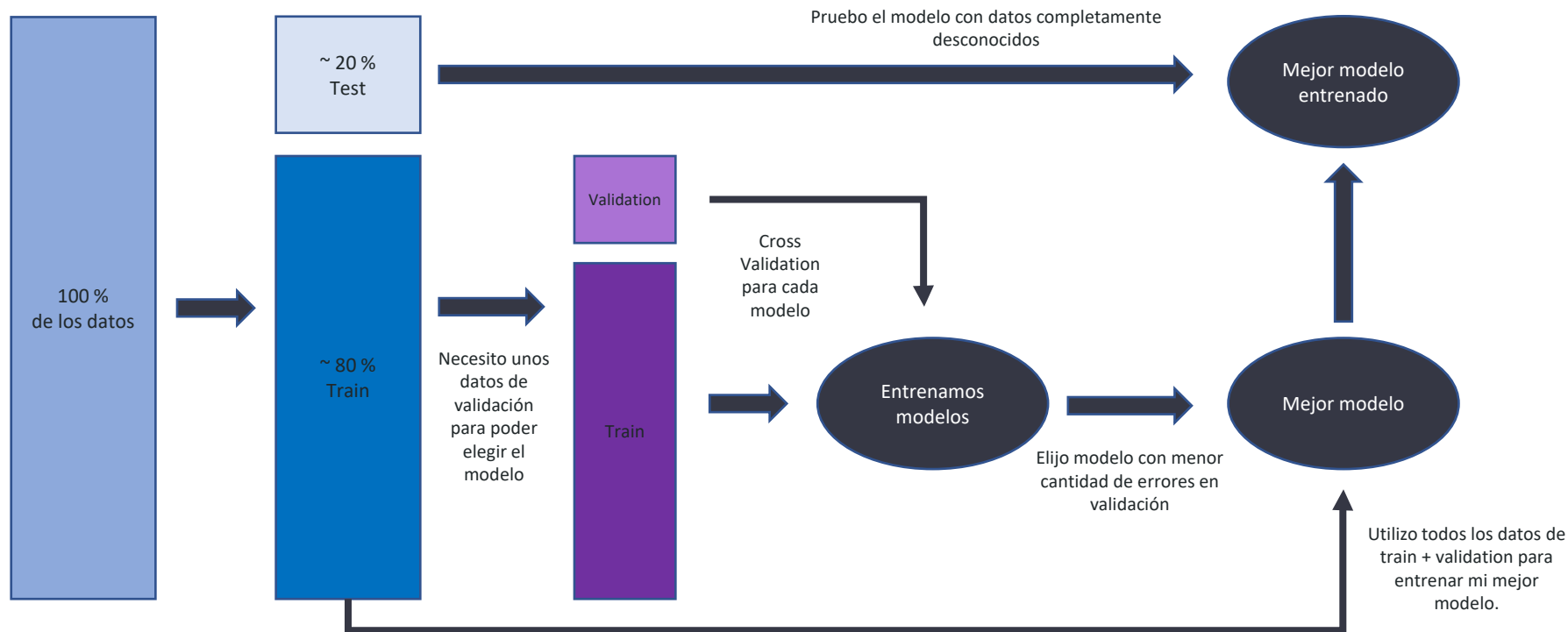
Train – Test - Validation

División del dataset



Train – Test - Validation

División del dataset



Train – Test - Validation

Algunas consideraciones

Cuidado al aplicar modificaciones en los conjuntos de train/test. **No podemos contaminar ambos conjuntos con datos del otro.** El conjunto de test se separa de los datos y no se toca hasta el final. Todas las transformaciones que apliquemos en train, tiene que salir únicamente del conjunto de train.

Ejemplo: si vamos a imputar missings en una columna por su media (imaginemos 31.24 con todos los datos), á la hora de imputarla en el conjunto de train, estaríamos utilizando los datos de test para imputar esa media, cuando en realidad **para los datos de train, los de test son completamente desconocidos.** La media del conjunto de train podría ser de 22.67, muy lejos de la media de train + test.

¿Cómo aplicar las transformaciones/limpieza obtenidas hasta ahora?

Scalers: por ejemplo, si tenemos un StandardScaler en train, usar ese mismo StandardScaler en test.

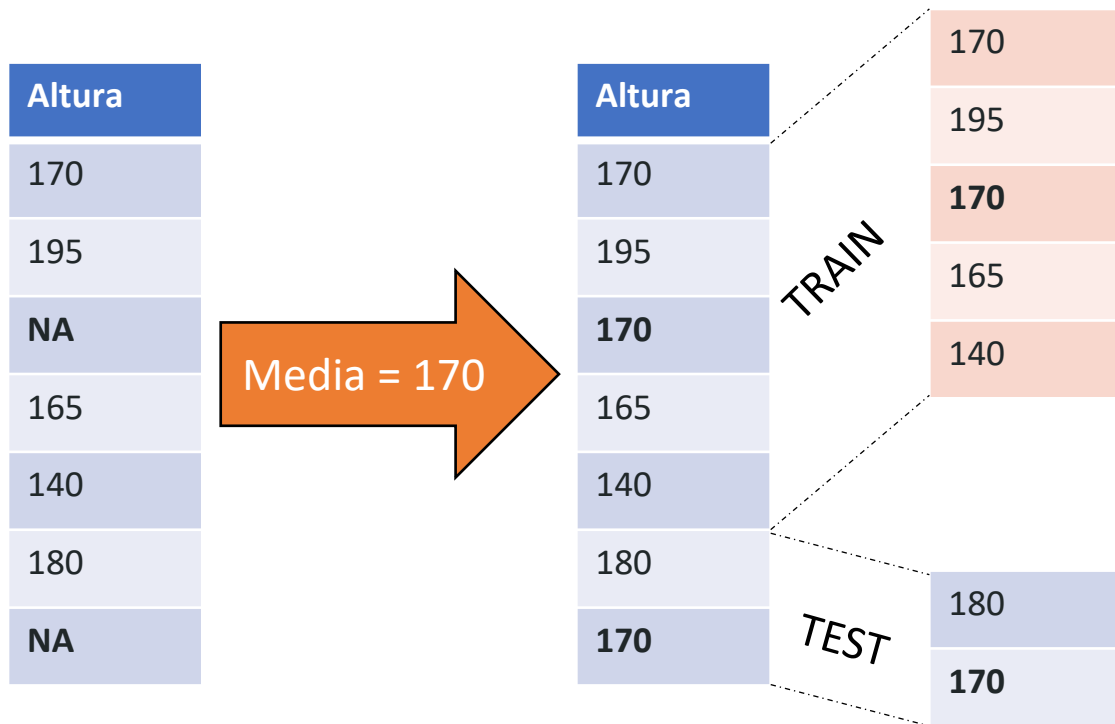
Missings: si aplicábamos la media/mediana/moda en train, aplicar esa misma métrica en test.

Feature reduction: eliminar las features que quitábamos en train (y añadir los dummies vacíos en test)

Feature engineering: mismos cálculos que en train.

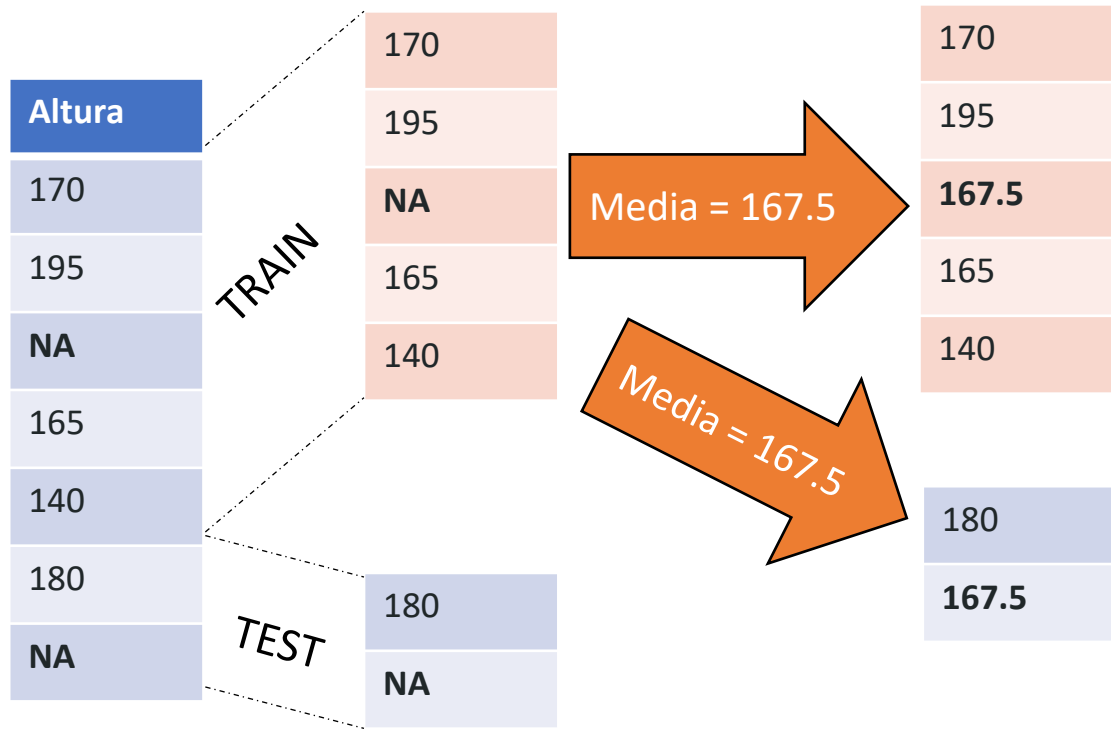
Train – Test - Validation

Algunas consideraciones



Train – Test - Validation

Algunas consideraciones



Resumen

Modelo

A partir de una serie de inputs (features), podré realizar predicciones sobre una variable objetivo (target)

¿Qué necesito?

Datos, y un buen lenguaje de programación con librerías de machine learning (sklearn y keras de python)

¿Cómo construyo un modelo?

Entrenándolo con datos y buscando su error mínimo. Para ello hay que aplicar el método científico.

Divido los datos en train + test. El conjunto de train a su vez lo divido en train + validation que utilizo para escoger el mejor modelo.

Pruebo el mejor modelo en test. Siempre busco que tenga pocos errores y generalice bien.

Objetivo del modelo

Pocos errores y que generalice bien cuando se enfrente a datos nuevos. Tiene que ser capaz de predecir bien ante datos futuros.

Overfitting y Underfitting

Underfitting es que el modelo no encuentra bien las asociaciones entre los datos, y overfitting que se ajusta demasiado al entrenamiento. En ambos casos no va a generalizar bien el modelo. Hay que buscar el equilibrio (bias vs variance)

¡Gracias!

Contacto: Rafael Zambrano

rafazamb@gmail.com