IMT2200

INTRODUCCIÓN A CIENCIA DE DATOS

2022-2

CROSS VALIDATION CLASIFICACIÓN

CLASE 21

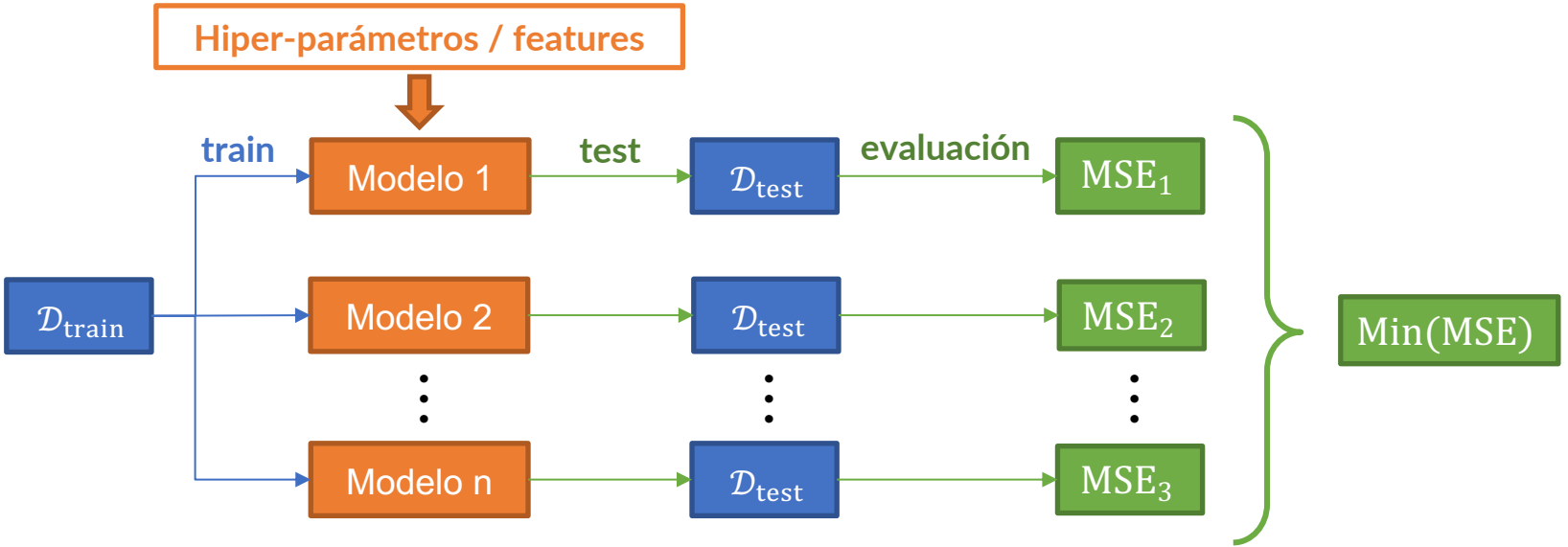
ALGORITMOS DE REGRESIÓN

Modelo de regresión	Función	Función de pérdida	Hiper-parámetro
kNN	No hay	$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$	k
Lineal	$Y = \beta_0 + \beta_1 X$	$\mathcal{L}(\beta) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \frac{1}{n}\sum_{i=1}^{n}\lvert y_i - \beta^{\mathrm{T}} Xi\rvert^2$	
Multilineal	$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_J X_J$		
Polinomial	$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_M X^M$		M
Lasso	$Y = \beta X$	$L_{LASSO}(\beta) = \frac{1}{n}\sum_{i=1}^{n}\lvert y_i - \beta^{\mathrm{T}} Xi\rvert^2 + \alpha\sum_{m=1}^{M}\lvert\beta_m\rvert$	α
Ridge	$Y = \beta X$	$L_{Ridge}(\beta) = \frac{1}{n}\sum_{i=1}^{n}\lvert y_i - \beta^{\mathrm{T}} Xi\rvert 2 + \alpha\sum_{m=1}^{M}\beta_m{}^2$	α

Para un conjunto de datos dado, ¿qué modelo elegimos?

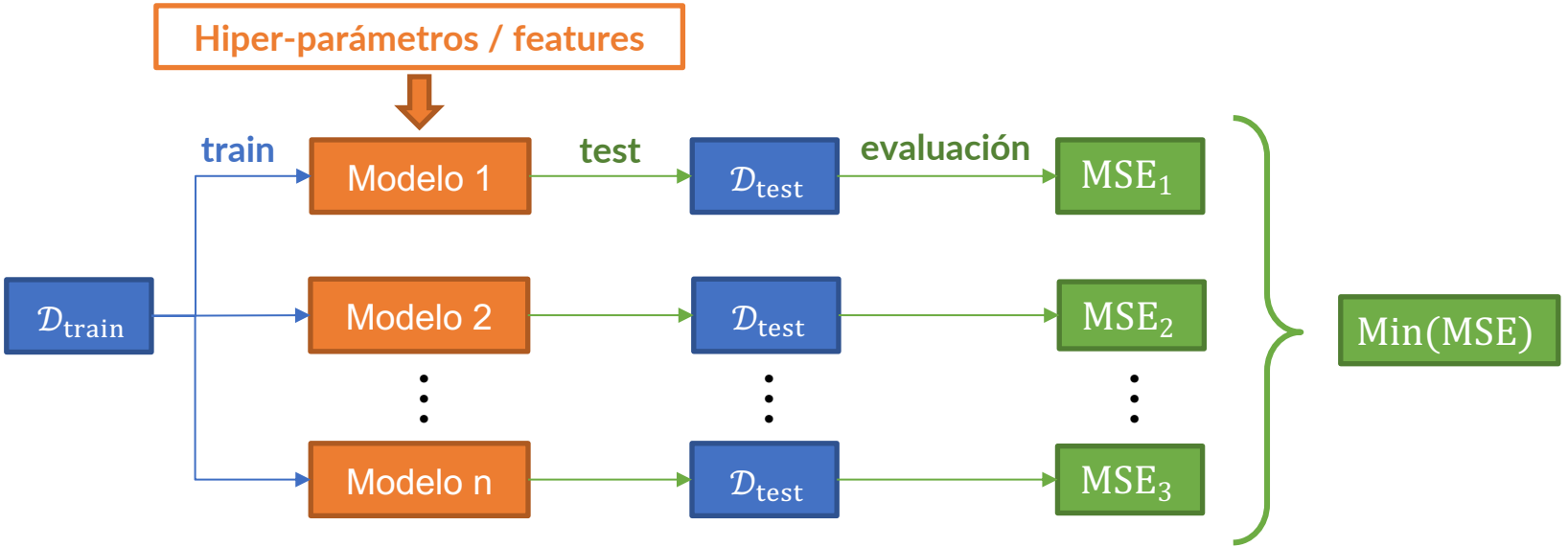
SELECCIÓN DE MODELOS

- Método para determinar la complejidad del modelo a utilizar, y/o el valor óptimo de los hiperparámetros
- El objetivo es **seleccionar el modelo con mejor capacidad de generalización**, es decir, de entregar predicciones correctas para nuevos datos.
- Para ello, se evalúa alguna métrica de rendimiento (MSE, R^2, etc.) de los posibles modelos, y se elige aquél que **optimiza la métrica para los datos de prueba**:



SELECCIÓN DE MODELOS: GRID SEARCH

- Método para determinar la complejidad del modelo a utilizar, y/o el valor óptimo de los hiperparámetros
- El objetivo es **seleccionar el modelo con mejor capacidad de generalización**, es decir, de entregar predicciones correctas para nuevos datos.
- Para ello, se evalúa alguna métrica de rendimiento (MSE, R^2, etc.) de los posibles modelos, y se elige aquél que **optimiza la métrica para los datos de prueba**:

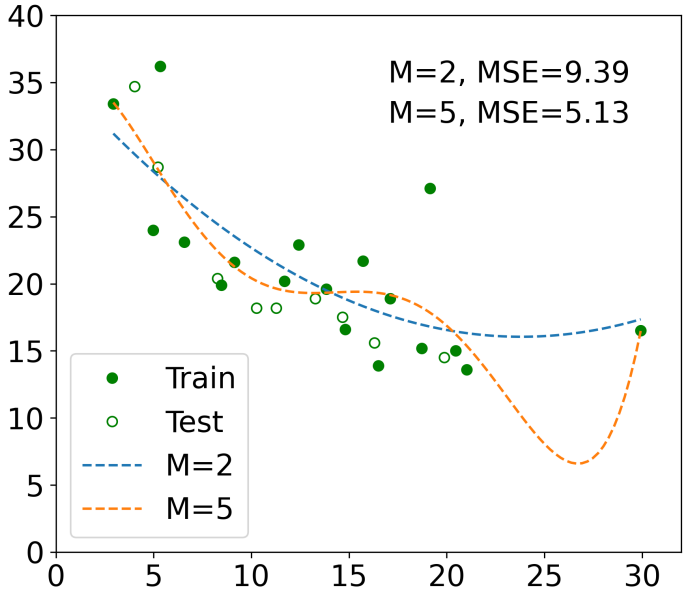


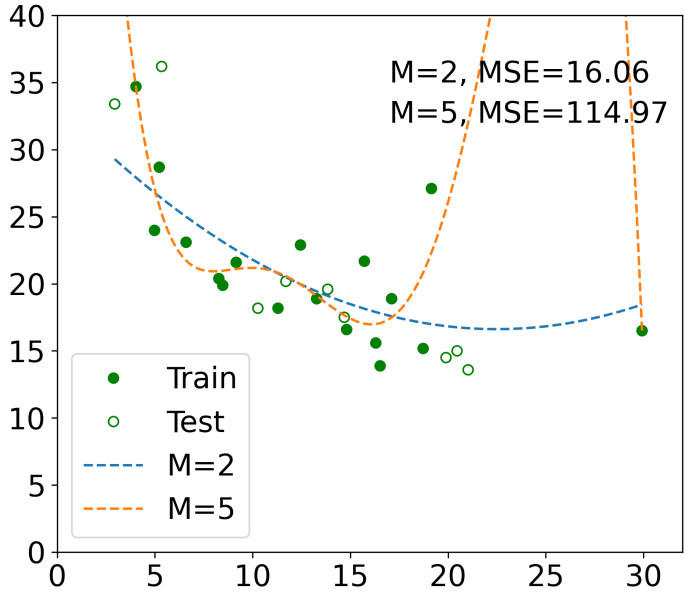
VALIDACIÓN CRUZADA (CROSS VALIDATION, CV)

- Por otra parte, usar una única validación para seleccionar un modelo puede ser problemático, ya que dependiendo de la partición de **datos de entrenamiento / prueba**, podemos llegar a **distintos resultados**.

- Ejemplo:

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_M X^M$$





VALIDACIÓN CRUZADA (CROSS VALIDATION, CV)

Validación cruzada ➔ método estadístico para evaluar la generalización de un modelo de manera más estable que usando un conjunto de datos de entrenamiento/prueba.

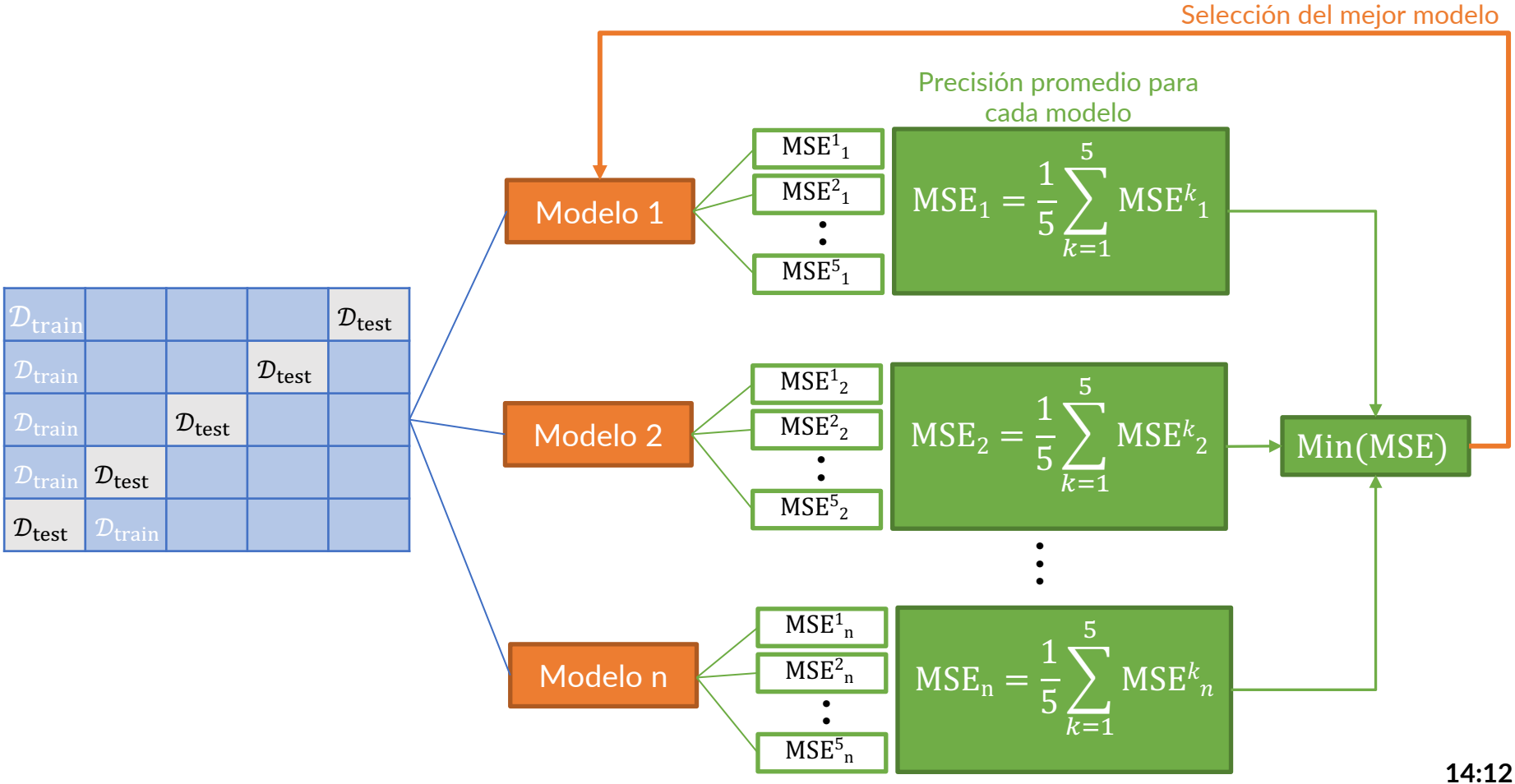
k-fold cross validation:

- Los datos se dividen en k subconjuntos o "folds" (k~5-10).
- Para cada iteración:
  - Se entrena un modelo utilizando un fold como $\mathcal{D}_{\text{train}}$, y los otros como $\mathcal{D}_{\text{test}}$.
  - Se evalúa la precisión de la predicción para $\mathcal{D}_{\text{test}}$.
- Finalmente, se calcula la **precisión promedio** para todas las iteraciones .

$$Error = \frac{1}{5}\sum_{i=1}^{5} Error_i$$

14:12

GRID-SEARCH CROSS-VALIDATION (`GridSearchCV`)



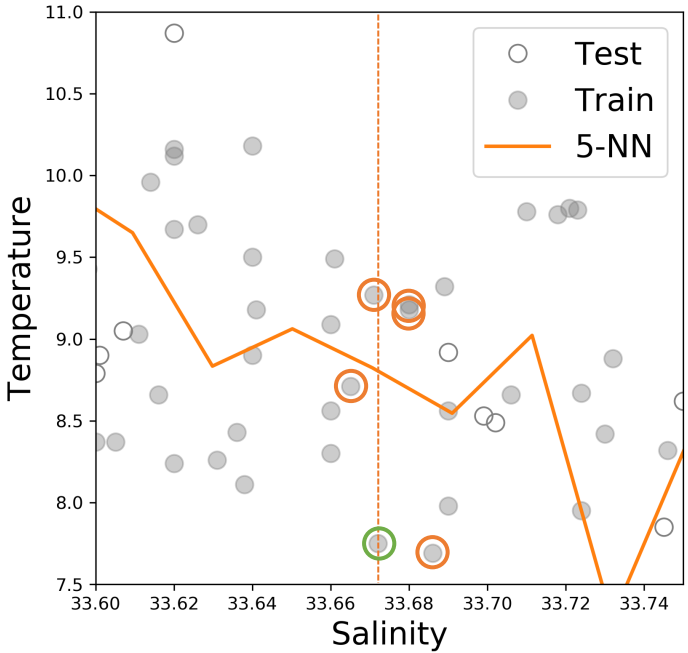
14:12

Aprendizaje Supervisado
Métodos de Clasificación

Clasificación kNN

CLASIFICACIÓN kNN

kNN para regresión ➔ usamos como predictores, las observaciones disponibles (x,y) más similares a la observación (x) que queremos predecir.

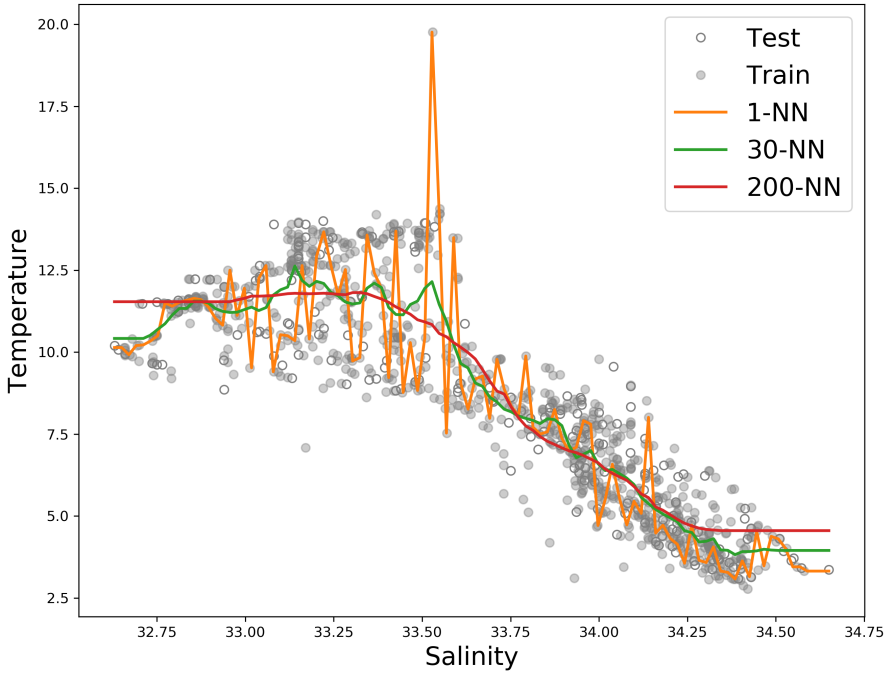


$$\hat{y}_i = \frac{1}{k}\sum_{j=1}^{k} y_{i_j}$$

y_{i_j} son los k vecinos más cercanos a (xi, y_i)

CLASIFICACIÓN kNN

kNN para regresión ➔ usamos como predictores, las observaciones disponibles **(x,y)** más similares a la observación **(x)** que queremos predecir.

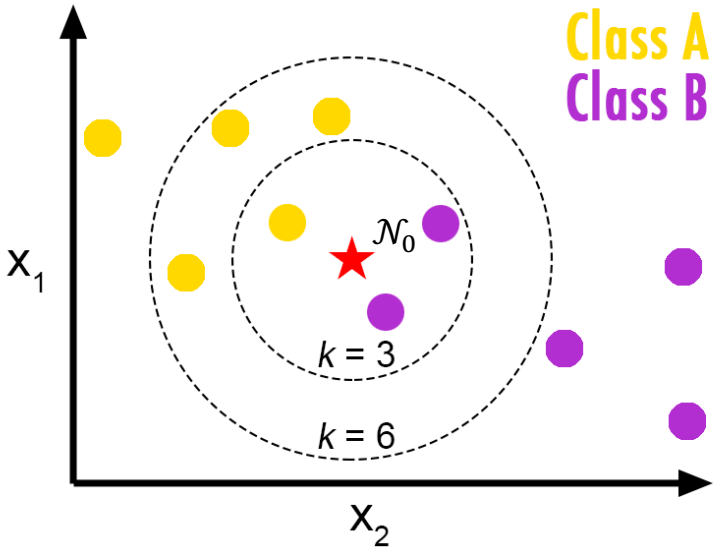


$$\hat{y}_i = \frac{1}{k}\sum_{j=1}^{k} y_{i_j}$$

y_{i_j} son los **k** vecinos más cercanos a (xi, y_i)

CLASIFICACIÓN kNN

kNN para clasificación ➔ clasificamos una observación específica, en base a las categorías de sus vecinos más cercanos.



Para un dato x_0 :

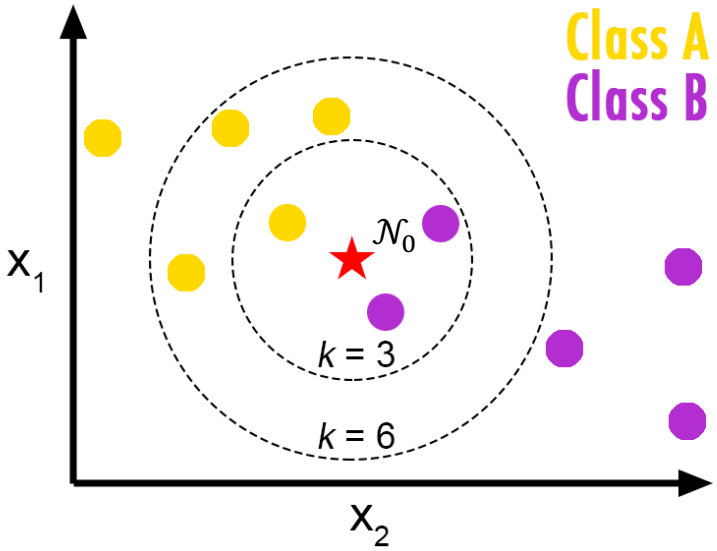
1. Se calcula la distancia a todos los demás puntos x_i:

$$D^2(x_i, x_0) = \sum_{j=1}^{P} (x_{i,j} - x_{0,j})^2$$

2. Se identifican los k puntos del dataset de entrenamiento más cercanos a x_0 ➔ $\mathcal{N}_0$

CLASIFICACIÓN kNN

kNN para clasificación ➔ clasificamos una observación específica, en base a las categorías de sus vecinos más cercanos.



3. Se estima la probabilidad condicional de la clase j, como la fracción de puntos en $\mathcal{N}_0$ cuya respuestas son j

$$P(Y = j|X = x_0) = \frac{1}{k}\sum_{i \in \mathcal{N}_0} I(y_i = j)$$

4. Se aplica la regla de Bayes y se clasifica la observación de prueba x_0 a la clase con la mayor probabilidad estimada.

$k = 3$: para ★

$P(Y = A|X_1, X_2) = \frac{1}{3}, P(Y = B|X_1, X_2) = \frac{2}{3}$ ➔ $Y = B$

CLASIFICACIÓN kNN: NORMALIZACIÓN

Si hay múltiples predictores: se define una medida de distancia multidimensional para identificar las observaciones más similares o "vecinos".

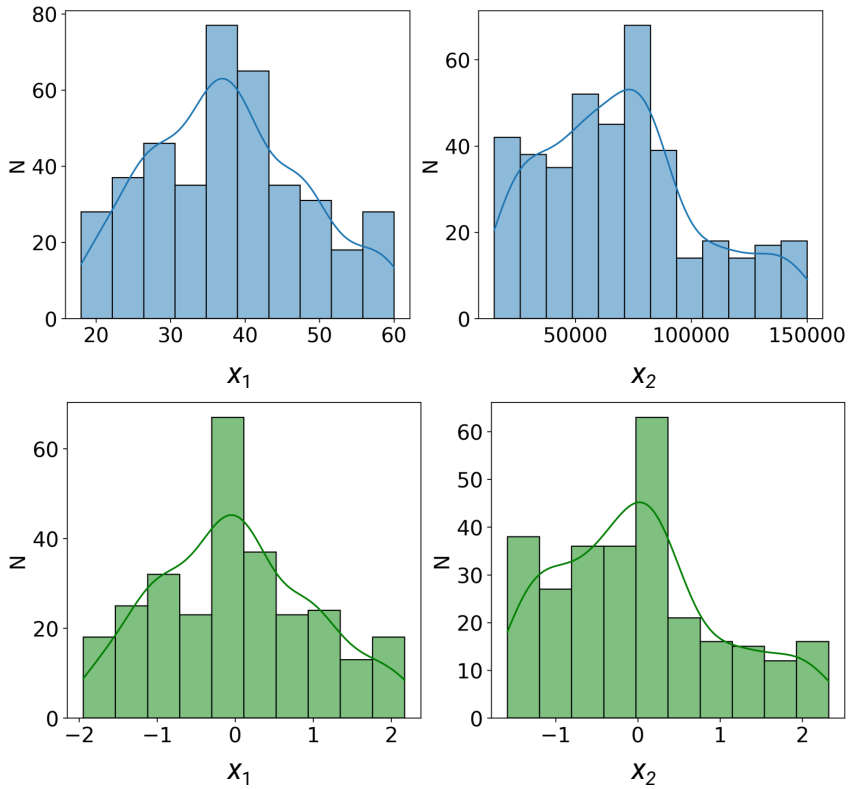
- Distancia euclideana: $D(\boldsymbol{x}_i, \boldsymbol{x}_0) = \sqrt{\sum_{j=1}^{P}(x_{i,j} - x_{0,j})^2}$
- Si los predictores tienen diferentes escalas y variabilidad ➔ se introducen efectos de escala en la medición de distancia.
- Por lo tanto, para $p > 1$, es necesario estandarizar los predictores.
- **Normalización z:** se resta la media, y se divide por la desviación estándar.

$$\Rightarrow \quad x_{scaled} = \frac{x - \mu}{\sigma}$$

CLASIFICACIÓN kNN: NORMALIZACIÓN

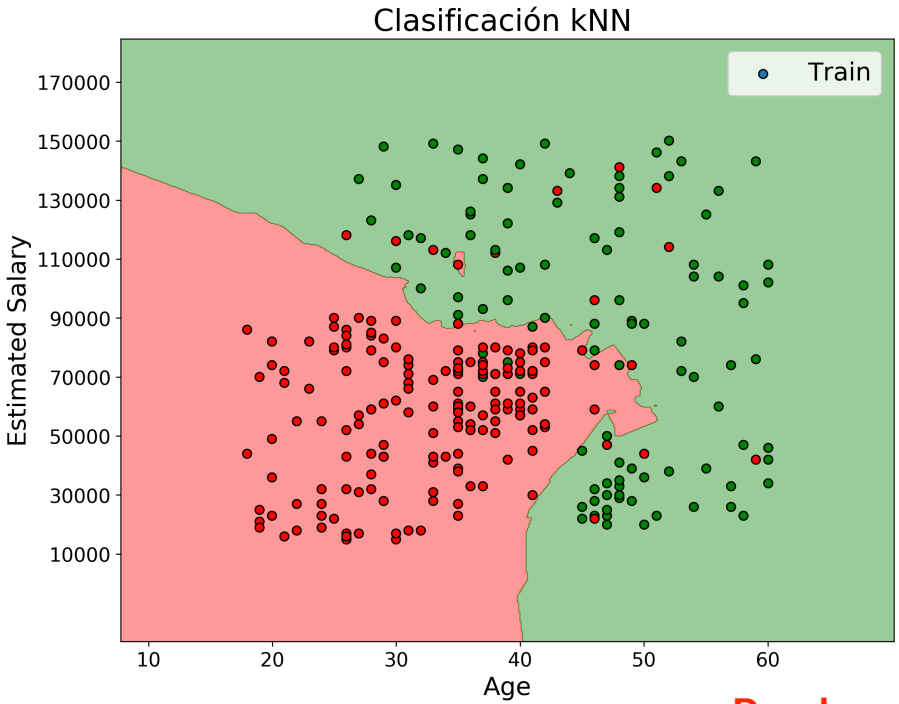
Ejemplo: Predicción de comportamiento de compra de clientes de una RS en base a su edad e ingresos.

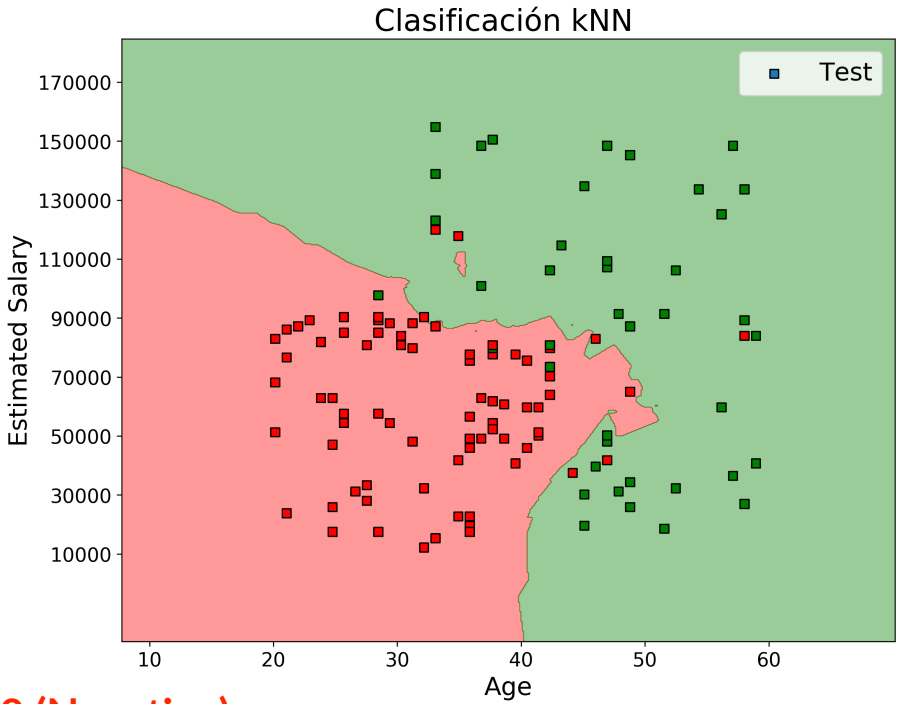
	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0
...	...	...	...	...	...



Datos normalizados

CLASIFICACIÓN kNN: Ejemplo



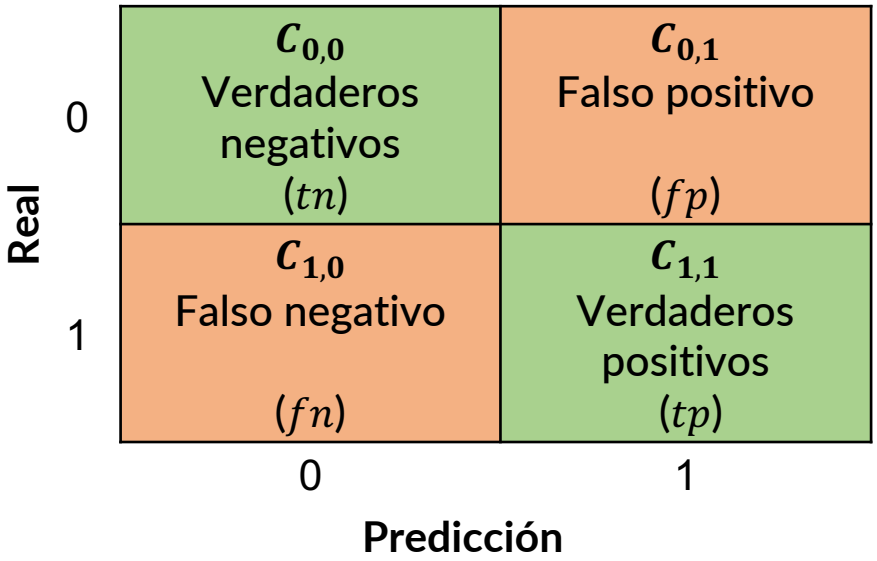


Purchased=0 (Negative)
Purchased=1 (Positive)

CLASIFICACIÓN: EVALUACIÓN DEL MODELO

Matriz de confusión: es usada para evaluar los resultados de la clasificación

- $C_{i,j}$: número de observaciones que se sabe están en el grupo i, y son clasificadas en el grupo j



Accuracy/ Exactitud: fracción de aciertos (en el dataset de prueba)

$$\text{accuracy} = \frac{tp + tn}{tp + fp + tn + fn}$$

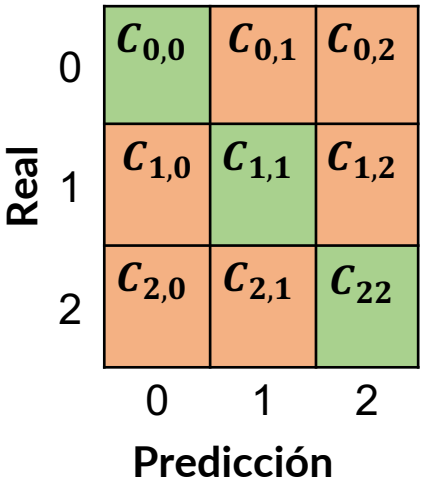
Precision/Precisión: capacidad de no clasificar como "positivo" un negativo

$$\text{precision} = \frac{tp}{tp + fp}$$

CLASIFICACIÓN: EVALUACIÓN DEL MODELO

Matriz de confusión: es usada para evaluar los resultados de la clasificación

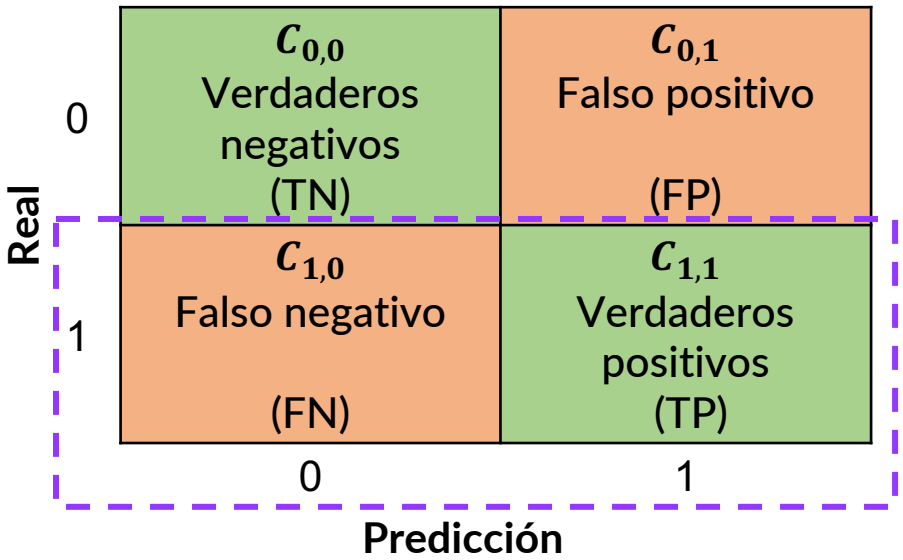
- $C_{i,j}$: número de observaciones que se sabe están en el grupo i, y son clasificadas en el grupo j



CLASIFICACIÓN: EVALUACIÓN DEL MODELO

Matriz de confusión: es usada para evaluar los resultados de la clasificación

- $C_{i,j}$: número de observaciones que se sabe están en el grupo i, y son clasificadas en el grupo j



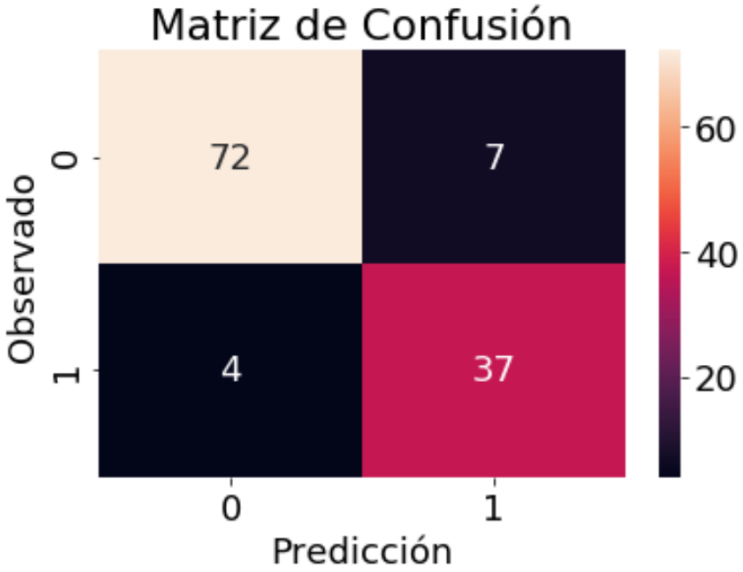
Recall / Sensibilidad: capacidad del clasificador de identificar todos los "positivos"

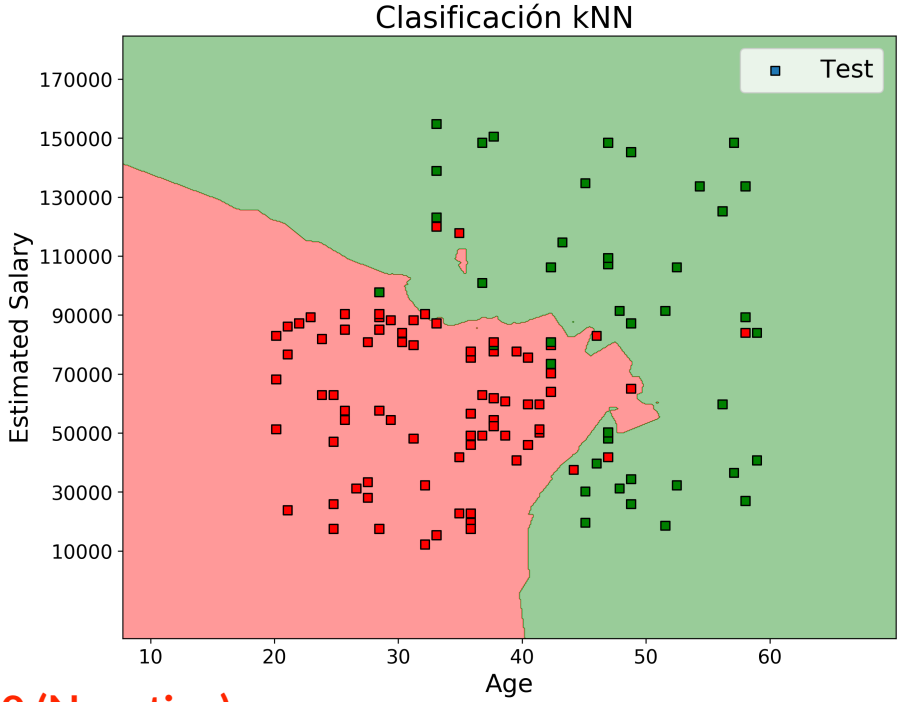
$$\text{recall} = \frac{tp}{tp + fn}$$

F-score: promedio ponderado de precisión y recall

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

CLASIFICACIÓN kNN: Ejemplo





Purchased=0 (Negative)
Purchased=1 (Positive)