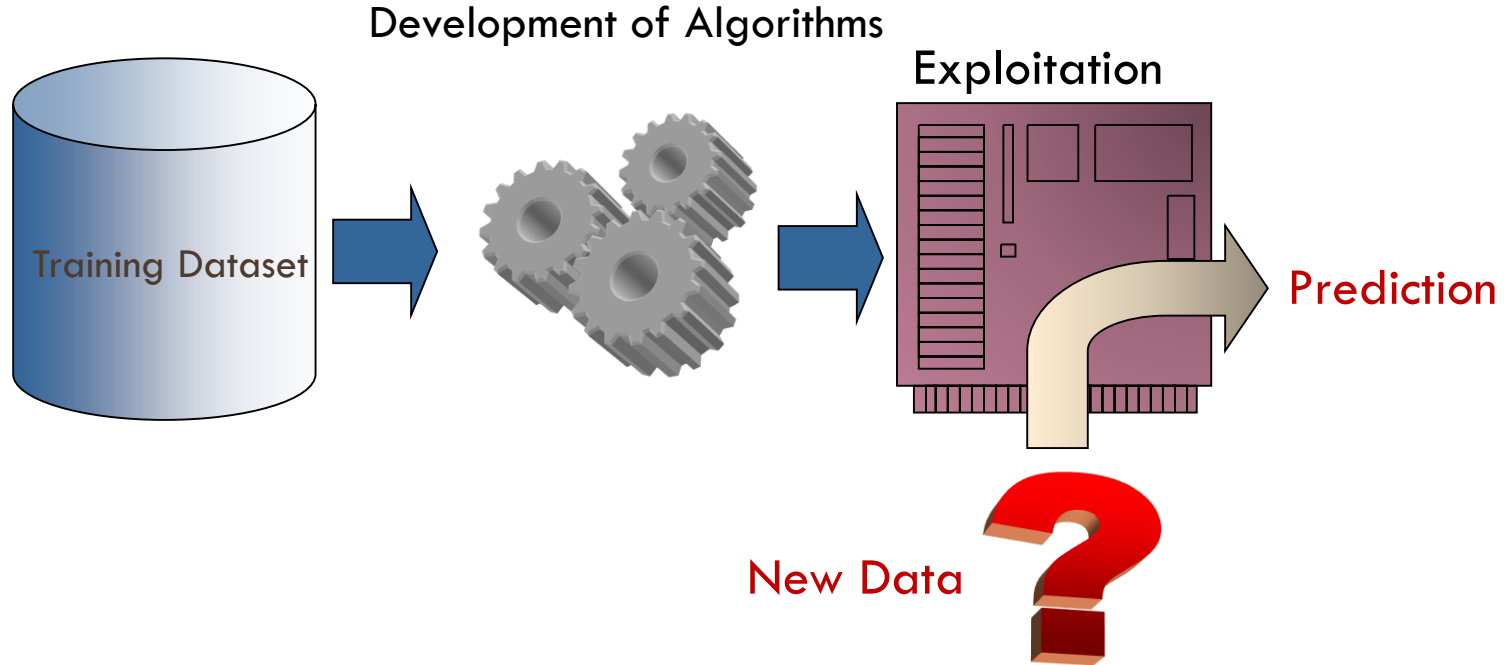


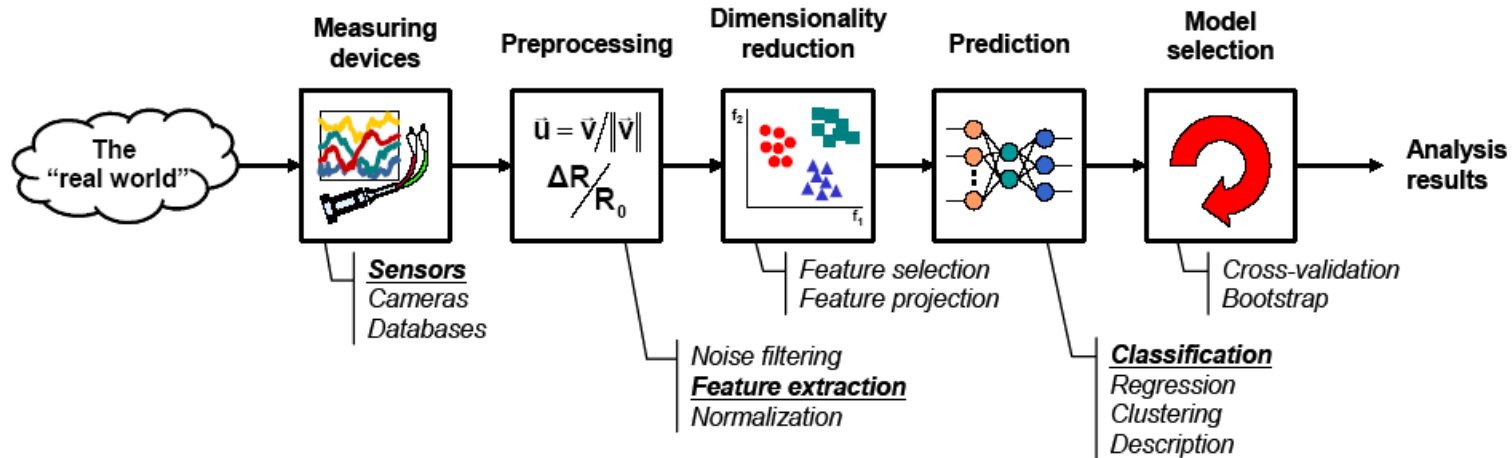
Introducción de Conceptos

Design Cycle

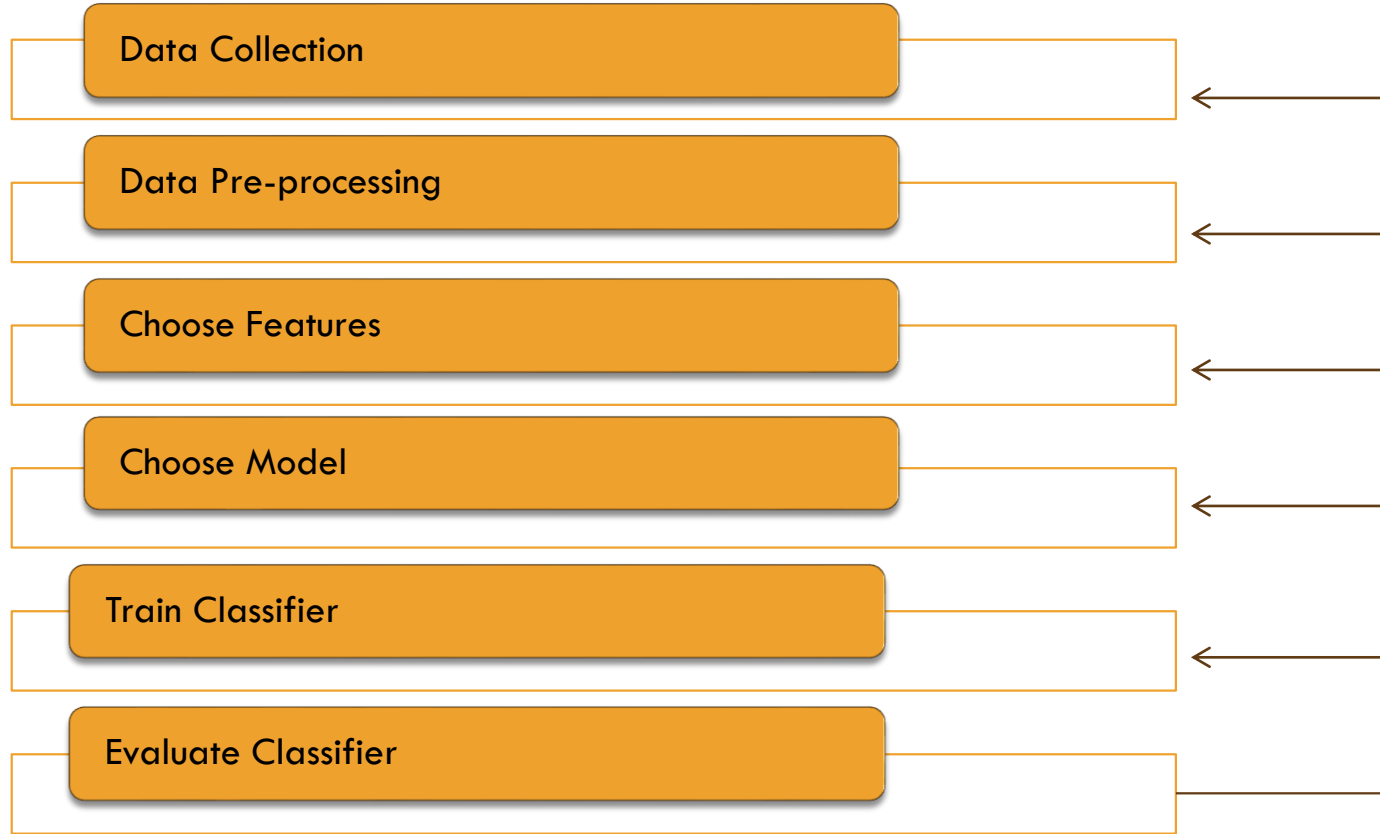


Design Cycle

- A pattern classification system contains:
 - Acquisition Sensor → Raw database
 - Preprocessing Mechanisms
 - Mechanism of dimensionality reduction
 - Learning Algorithms
 - Mechanisms for validation



Design Cycle



Design Cycle

➤ Data Collection

- Probably the most time-consuming part
- How much data?
 - Sufficient large number of instances
 - Representative

➤ Data Preprocessing

- Data Cleaning: Missing Values, Outliers, Signal/Noise Rate
- Data Integration
- Data Transformation
- Data Reduction

➤ Choose Features

- It is critical for success in a Pattern Recognition problem
- It requires a basic understanding of the a priori problem
- Ideally:
 - Simple to extract
 - Invariant to irrelevant transformation
 - Insensitive to noise

Data
Collection

Data Pre-
processing

Choose
Features

Design Cycle

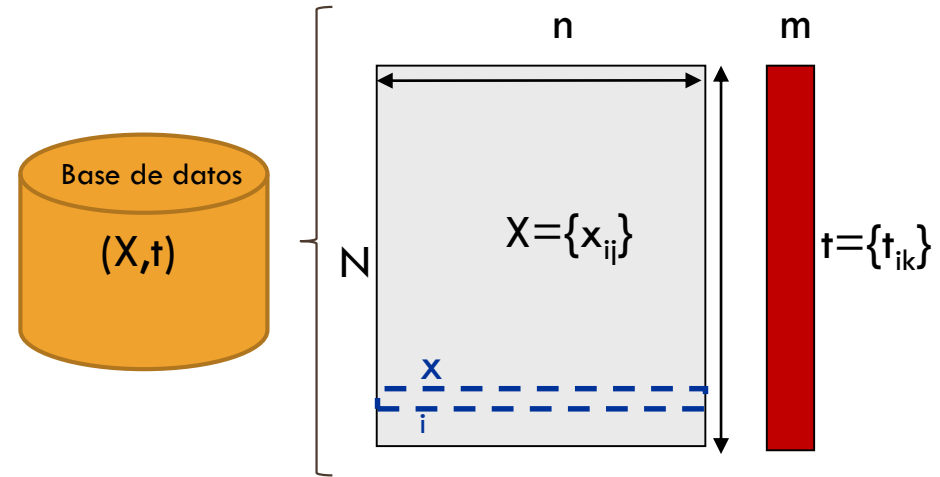
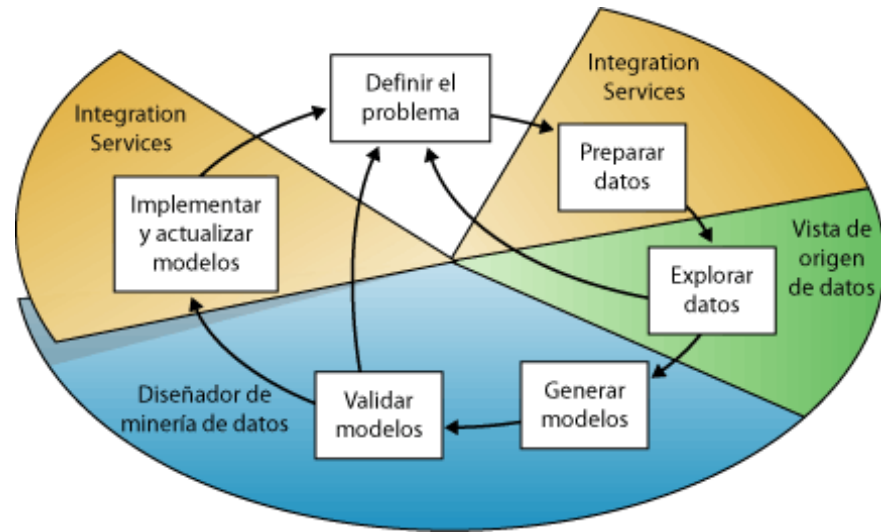
- Choose the model
 - Different types of models
 - Different parameters to play
- Training
 - Use data to obtain a good classifier
 - Identify best model
 - Determine appropriate models
- Evaluation
 - How good is the trained model?
 - Measure error rate → Performance
 - Overfitting versus generalization
 - May suggest switching:
 - From one set of feature to another set or adding new features
 - From one model to another one.

Choose Model

Train
Classifier

Evaluate
Classifier

Design Cycle



Representación de la Base de datos

Base de datos: X

N filas = patrones, casos, ejemplos, instancias, ...

n columns = atributos, variables, covariables, características, ...

Steps to develop a prediction model: training, validation and test sets

- Dividir el conjunto de datos iniciales en subconjuntos de forma aleatoria, manteniendo la representación de partida.

Conjunto de entrenamiento: conjunto de datos con el que

Se construyen distintos modelos para resolver el problema

Se seleccionan uno o varios modelos finales

Steps to develop a prediction model: training, validation and test sets

Conjunto de validación: conjunto de datos con el que

Se validan los modelos
del punto anterior.

Selección de los
parámetros.

Se determina el modelo definitivo según:

- 1.- Modelo con mejor ajuste a los datos
- 2.- Modelo más “cercano” al problema de predicción

Conjunto de prueba/test: conjunto de datos con el que

Se realizan pruebas de funcionamiento del modelo.

Estos datos no han jugado ningún papel en la selección del modelo

Steps to develop a prediction model: training, validation and test sets



Posibles mecanismos de selección de conjuntos →

Validación cruzada (Cross-validation):

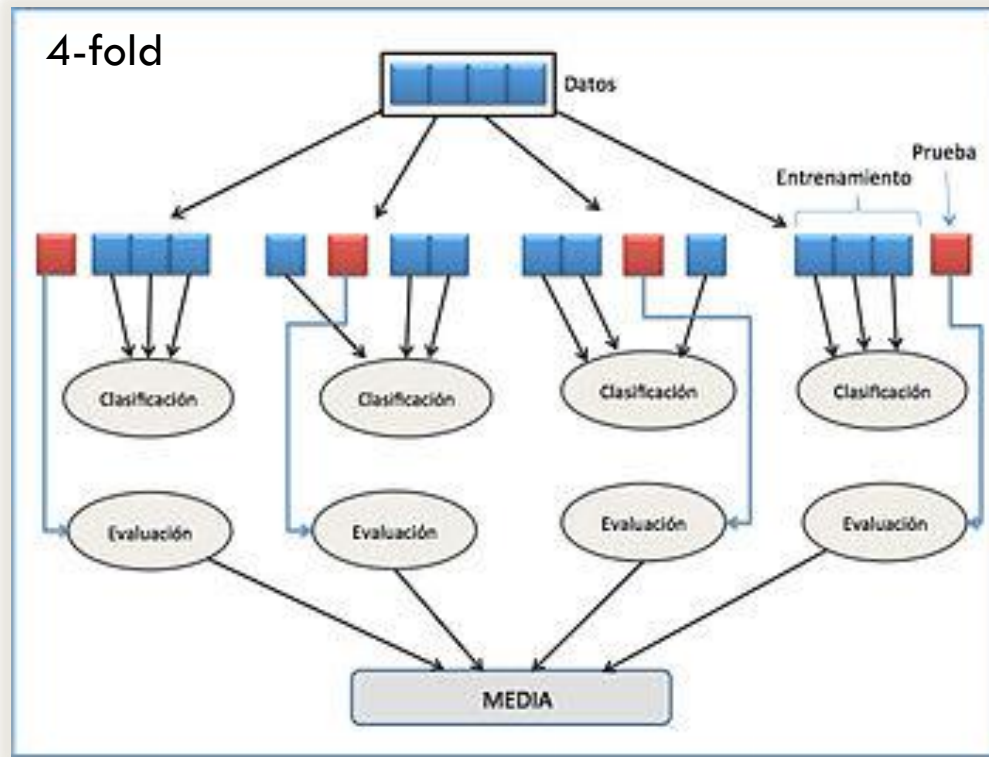
a) Muestra suficiente:

- Con conjunto de validación:
50% Training + 25% Validation + 25% Test
- Sin validación
60% Training + 40% Test

b) Muestras reducidas:

- K-fold (K=10)
- leave-one-out = n-fold
- Bootstrap = muestreo con reemplazamiento o sin reemplazo

Steps to develop a prediction model: training, validation and test sets



Steps to develop a prediction model: training models

Modelos lineales:

- Análisis Discriminantes Lineales / Regresión Lineal

Redes neuronales:

- Perceptrón Multicapa, PCM (“Multilayer Perceptron”, MLP)

Modelos basados en núcleos:

- Máquinas de vectores de soporte (“Support Vector Machine”, SVM)

Modelos probabilísticos:

- Redes Bayesianas

Árboles de decisión:

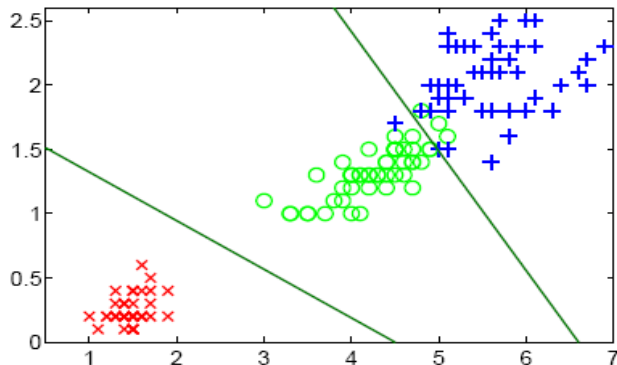
- ID3, C4.5

Métodos de “clustering”:

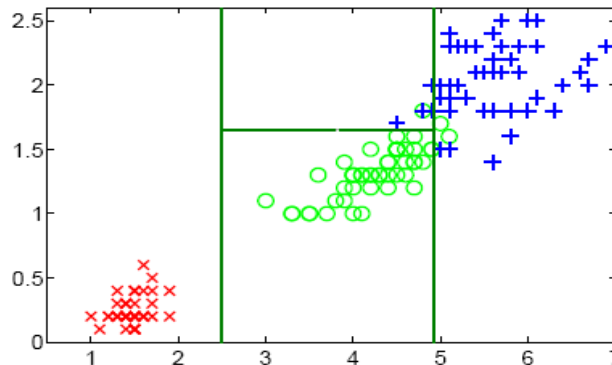
- “K-means”, Métodos Espectrales

Steps to develop a prediction model: select models

Discriminante Lineal



Árboles de decisión

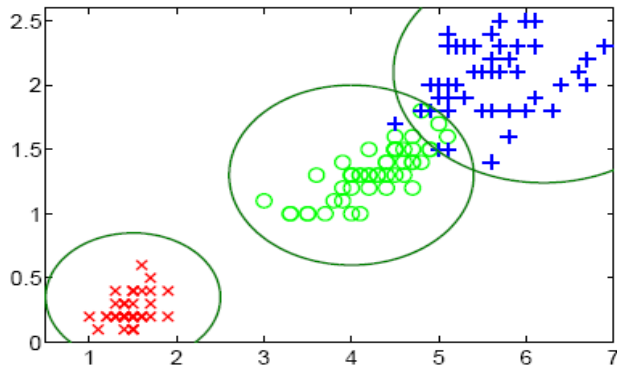


Iris Data Set
(Fisher, 1936)

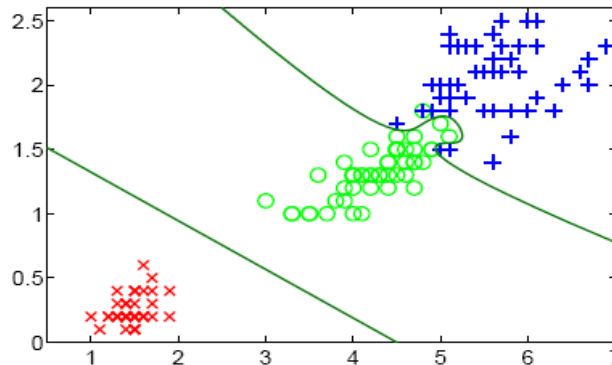
setosa

versicolor

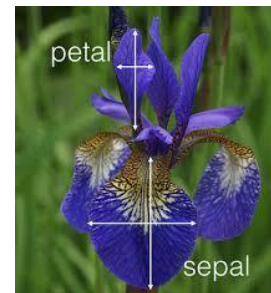
virginica



Mezclas de Gausianas

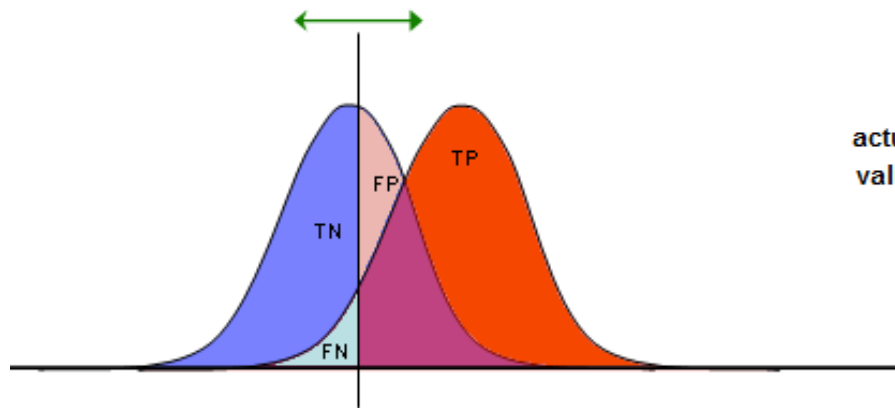


Kernel method (SVM)



Steps to develop a prediction model: select models

➤ Clasification with two classes



Sensitivity=Recall=TPR=Hit Rate

Specificity=Selectivity=1-FNR

		prediction outcome		total
		<i>p</i>	<i>n</i>	
actual value	<i>p'</i>	True Positive	False Negative	<i>P'</i>
	<i>n'</i>	False Positive	True Negative	<i>N'</i>
total		<i>P</i>	<i>N</i>	

$$Sens = \frac{TP}{TP + FN}$$

$$Spec = \frac{TN}{TN + FP}$$

$$acc = \frac{TN + TP}{P + N}$$

$$Precision = \frac{TP}{TP + FP}$$

Steps to develop a prediction model: select models

Ejemplo: (+) Estar enfermo, (-) Estar sano

- La sensibilidad nos indica la capacidad de nuestro estimador para dar como casos positivos:

- los casos realmente enfermos;
- proporción de enfermos correctamente identificados.

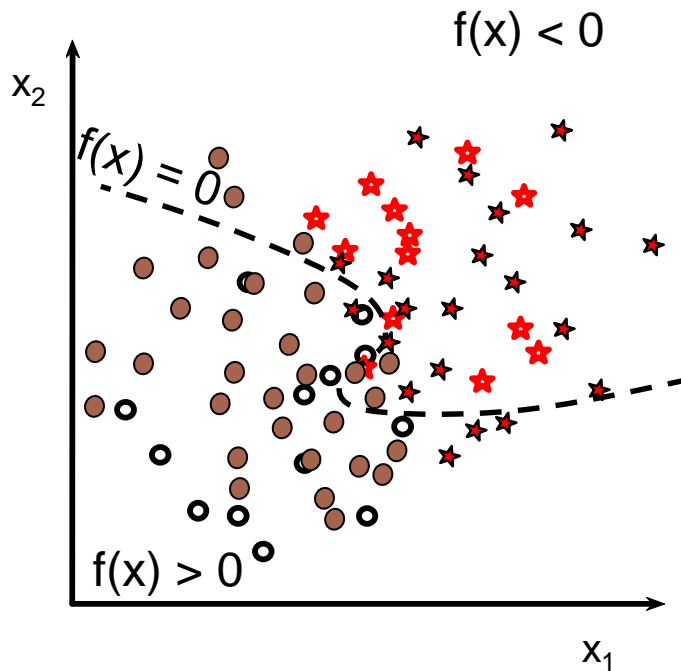
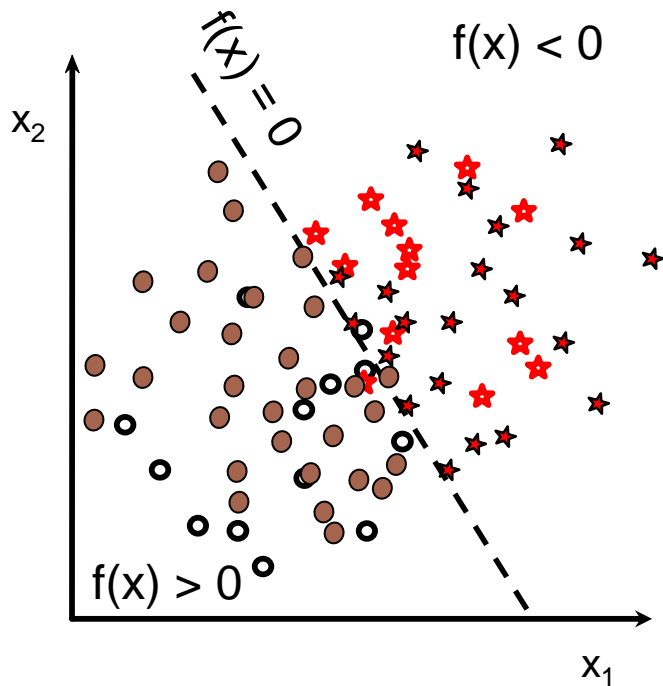
Es decir, la sensibilidad caracteriza la capacidad de la prueba para detectar la enfermedad en sujetos enfermos.

- La especificidad nos indica la capacidad de nuestro estimador para dar como casos negativos los casos realmente sanos:

- proporción de sanos correctamente identificados.

Es decir, la especificidad caracteriza la capacidad de la prueba para detectar la ausencia de la enfermedad en sujetos sanos.

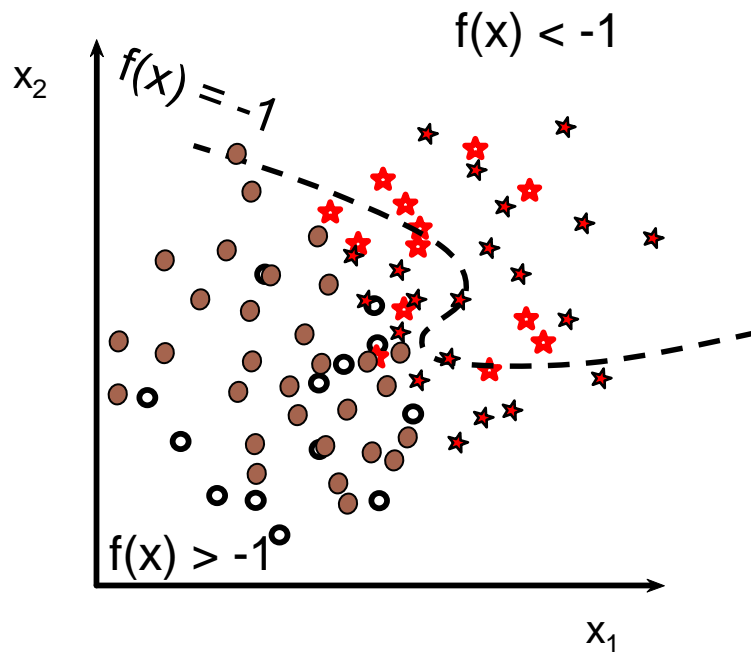
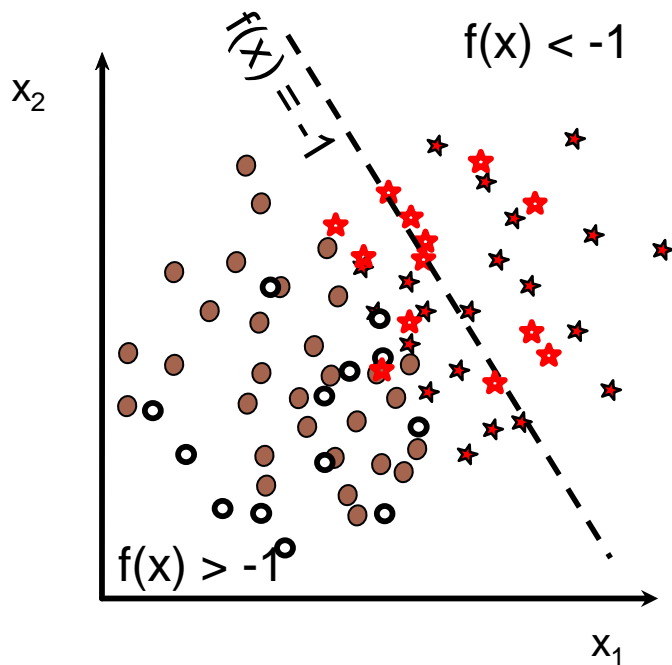
Steps to develop a prediction model: select models



Steps to develop a prediction model: select models

➤ Coste

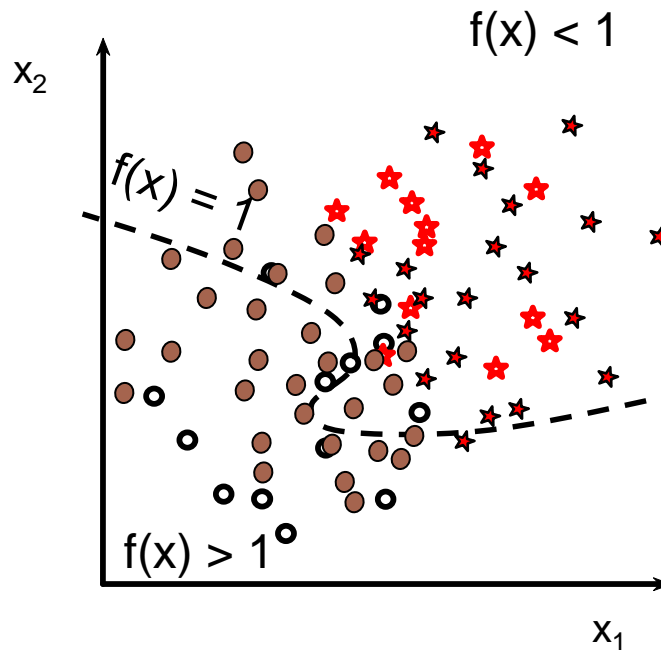
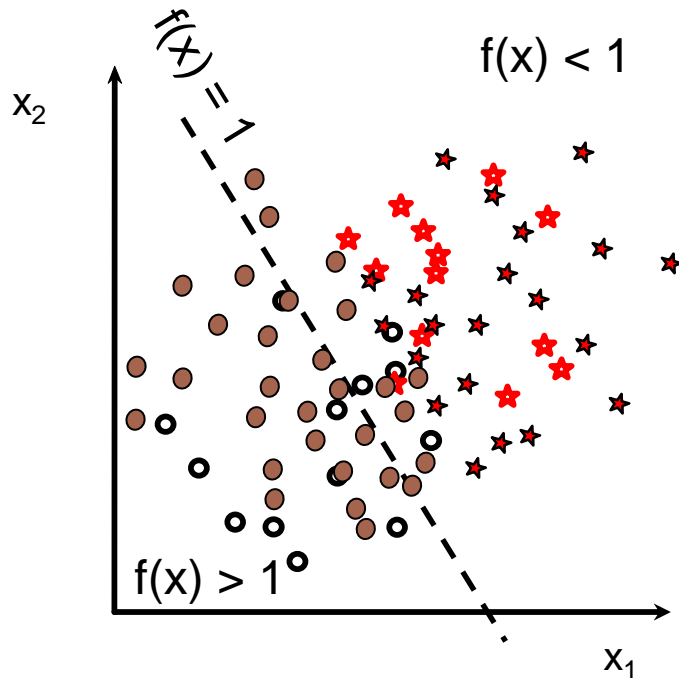
Variación del umbral → Elección de la frontera de decisión



Steps to develop a prediction model: select models

➤ Coste

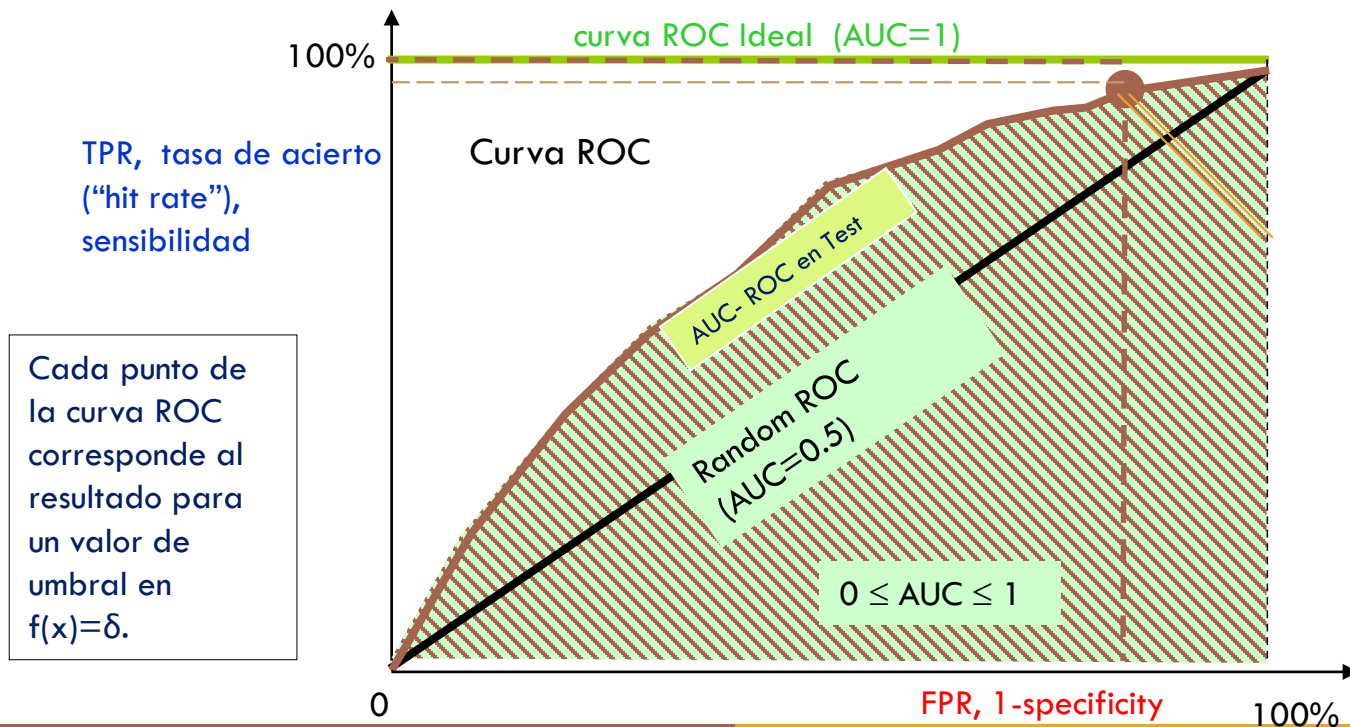
Variación del umbral ➔ Elección de la frontera de decisión



Steps to develop a prediction model: select models

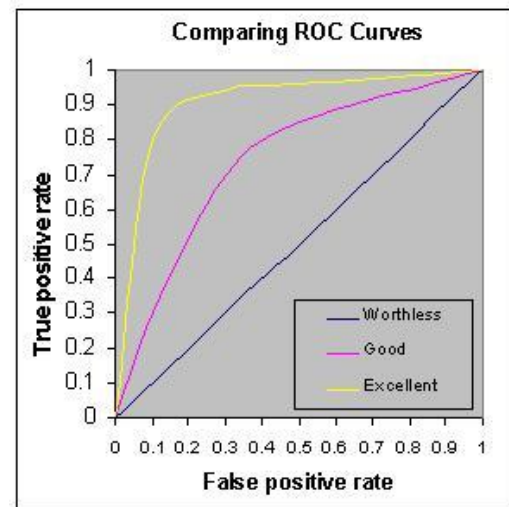
➤ Curva ROC y AUC-ROC

Requisito: un estimador de una variable con un parámetro ajustable



Steps to develop a prediction model: select models

- Puntos de la curva ROC son **distintos niveles de decisión o valores de corte** que permiten una clasificación dicotómica de los valores de la prueba según sean superiores o inferiores al valor elegido.
- La diferencia esencial con el caso más simple es que ahora no tenemos un único par de valores de sensibilidad y especificidad que definan la exactitud de la prueba, sino **un conjunto de pares** correspondientes a cada uno de los distintos niveles de decisión.
- El área bajo la curva (AUC-ROC) es una medida de la calidad del clasificador
 - 0.90-1.00 = excelente
 - 0.80-0.90 = buena
 - 0.70-0.80 = justa
 - 0.60-0.70 = pobre
 - 0.50-0.60 = mala



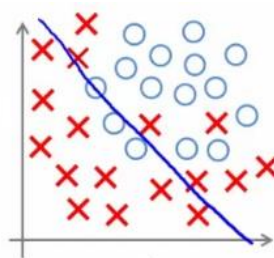
Generalization

➤ Over-fitting

- Typically occurs when the ratio complexity of the model training set size is too high:

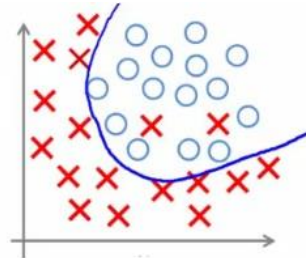
$$\frac{\text{Complexity of the model}}{\text{training set size } (N)}$$

Occam's Razor:
simpler theories are
easier to understand

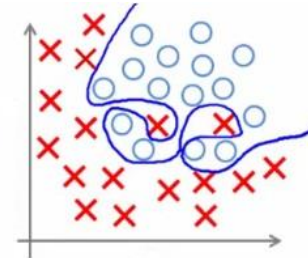


Under-fitting

(too simple to
explain the
variance)



Appropriate-fitting



Over-fitting

(forcefitting -- too
good to be true)

Learning by heart

Generalization

➤ Preventing over-fitting

- Early stopping
- Weight-Decay
 - Adding a penalty to the error function

$$\tilde{E} = E + \lambda \Omega$$

$$\Omega = \frac{1}{2} \sum_i w_i^2$$

- Training with random noise

$$\tilde{x} = x + \epsilon$$

