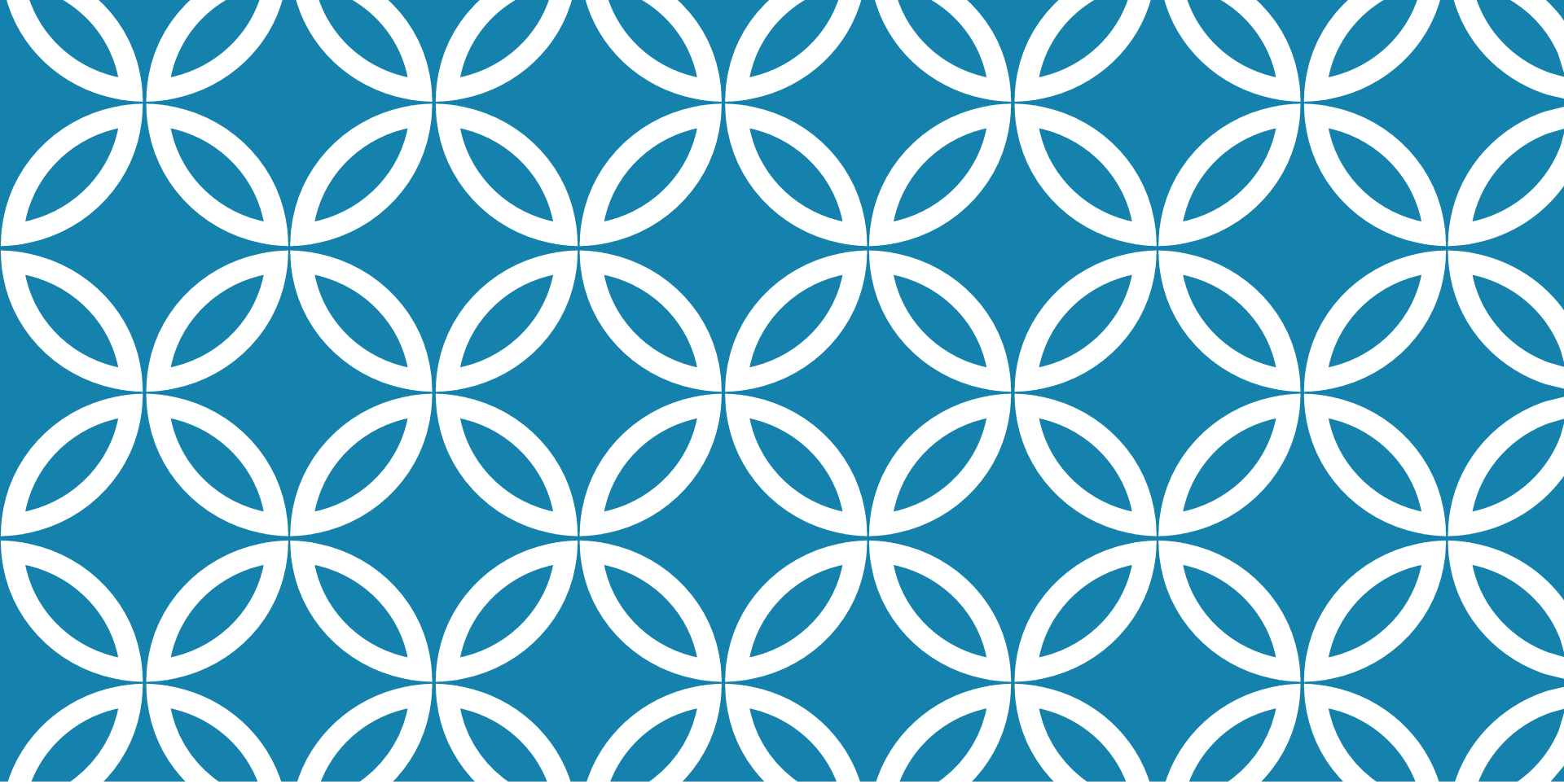


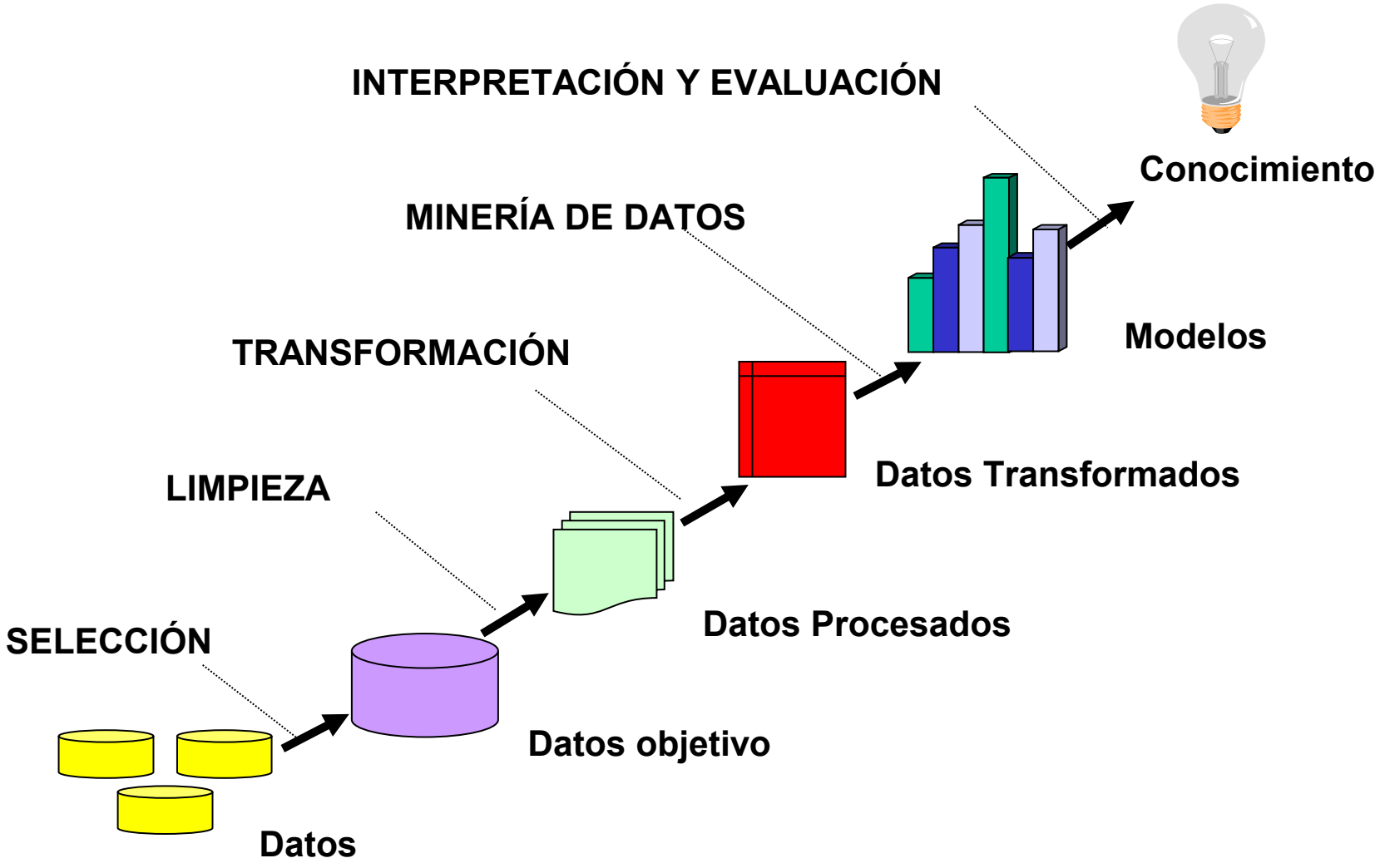
APRENDIZAJE AUTOMÁTICO

José Manuel Molina López
Grupo de Inteligencia Artificial Aplicada

METODOLOGÍA

1. Formular el problema
2. Determinar la representación (atributos y clases)
   - directamente
   - hablando con expertos
   - a partir de otras técnicas (filtros)
3. Identificar y recolectar datos de entrenamiento (bases de datos, ficheros, …)
4. Preparar datos para análisis
5. Selección de modelo, construcción y entrenamiento
6. Evaluar lo aprendido
   - validación cruzada, expertos
7. Integrar la base de conocimiento a la espera de nuevos datos tras acciones

EL PROCESO DE KDD (KNOWLEDGE DISCOVERY IN DATABASES)



ELEMENTOS BÁSICOS DE ENTRADA

Concepto: qué se quiere aprender (estructura inteligible y útil para cada tipo de problema). Salida: descripción del concepto

- Clasificación
- Predicción/Estimación
- Asociación
- Agrupamiento

Atributo: qué características (variables) se van a utilizar para describir el concepto

- Ej.: salario, crédito solicitado, categoría a la que pertenece, ...
- Tipos: continuos, nominales/categóricos

Clase: diferentes valores (etiquetas) del concepto aprendido

- Ej.:sí, no, necesita-aval, etc.

Instancia o ejemplo: cada muestra a partir de la cual se extrae el concepto

TABLA DE DATOS

Entrada: Instancias (o ejemplos):

SALARIO	CLIENTE	EDAD	…	HIJOS	CRÉDITO
Poco	Sí	Joven	…	Uno	NO
Mucho	No	Adulto	…	Dos	SI
Mucho	No	Adulto	…	Dos	SI
Medio	Sí	Mayor	…	Tres	NO
⋮	⋮	⋮	⋱	⋮	⋮

Salida: Árboles, tablas de decisión, reglas, clusters, modelos regresión, etc.

PREPARACIÓN DE DATOS

La preparación de datos puede suponer 60-90% del tiempo

- objetivo: única tabla de datos (instancias, atributos)
- ensamblar, integrar formatos, agregar, …

Filas: Agregación: selección de dato unitario

- asociar datos, calcular resúmenes, …

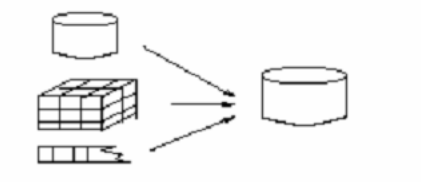
Columnas: Selección de atributos

- eliminar campos redundantes o inapropiados (ID)
- crear atributos de interés de campos textuales
  - fecha/hora->edad, estación, vacaciones, mañana/tarde/noche,
  - dirección/código postal-> lugar geográfico, área, ciudad
- transformar atributos

PREPARACIÓN DE DATOS

Integración

- Múltiples fuentes (bases de datos, ficheros, …)



Limpieza

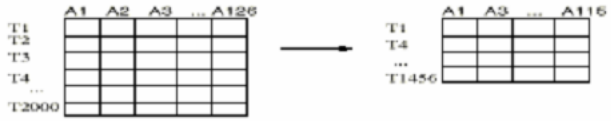
- Valores faltantes, outliers, inconsistencias

Transformación

- Normalizar, proyectar, discretizar

Reducción

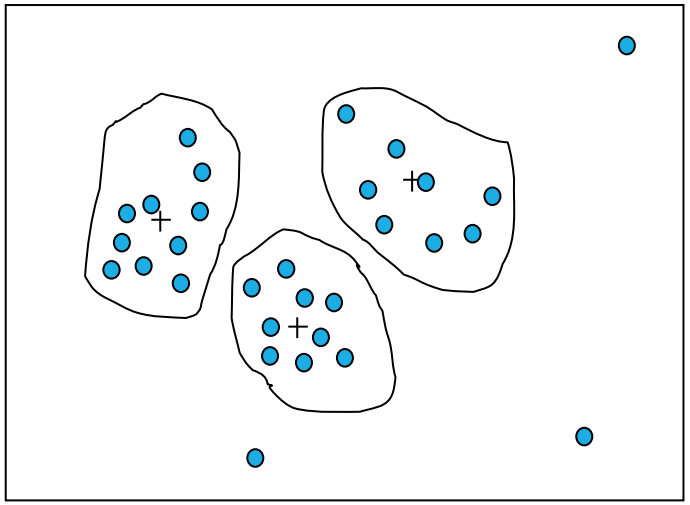
- Representación reducida



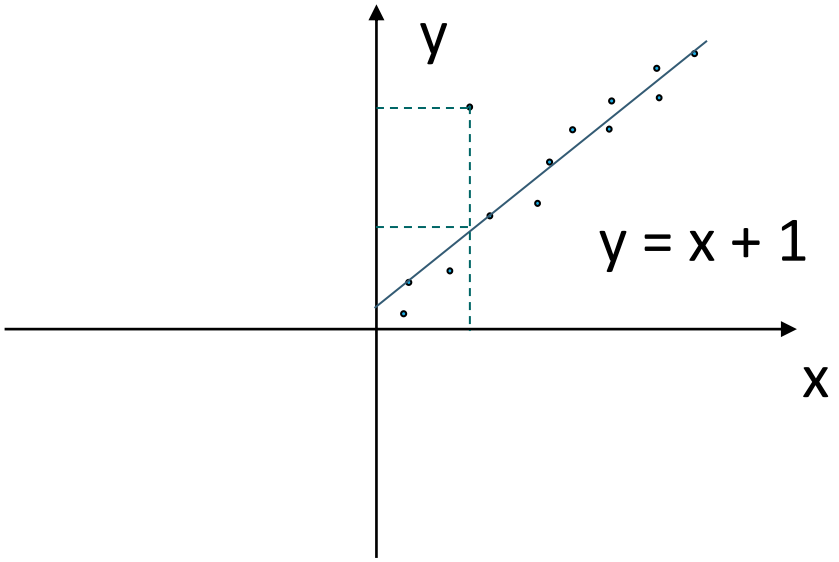
LIMPIEZA: RUIDO EN LOS DATOS

- Outliers : detección y eliminación (o tratamiento individual)

Analisis de clusters



Regresión



LIMPIEZA: DATOS INCOMPLETOS, REDUNDANTES

Datos Redundantes: detectar relaciones causales o funcionales en datos

- Frecuente al integrar múltiples bases de datos
  - El mismo atributo con diferente nombre
  - Relaciones directas: atributo "calculado"
- Se detecta con análisis de correlación/contingencia

Datos incompletos: faltan atributos en algunos ejemplos

- Relleno manual: tedioso o no posible
- Ignorar ejemplo: cuando son pocos casos
- Nuevo valor "desconocido"
- Valor medio, valor más probable según resto, …

SELECCIÓN DE DATOS DE ENTRADA

Selección de datos

- Aleatoriamente: conjuntos grandes. Verificación
- Aquellos que se parecen más entre sí
- Aquellos que se diferencian mas entre sí
- Los datos que están en las fronteras entre las clases
- Los datos que tienen mayores errores de clasificación se tratan (proporcionalmente) más veces
  - Boosting
- Incremental: incorporar sucesivamente datos de un conjunto reserva

Pre-procesamiento

- Reducción del ruido (filtrado de datos)
- Selección de atributos
- Tratamiento de los valores desconocidos, discretización de valores numéricos

FILTRADO DE ATRIBUTOS

Los errores en los datos son muy comunes y pueden degradar fuertemente el análisis

Se pueden aplicar técnicas que permitan identificar potenciales problemas, evitando o agilizando la supervisión manual.

Mejora de árboles de decisión

- El ruido en los atributos debe incorporarse también en el entrenamiento para aprender a combatirlo
- Descartar los ejemplos mal clasificados (y re-entrenar) frecuentemente reduce la complejidad de la estructura, con diferencias no significativas de prestaciones
  - Equivale a un proceso de poda global

Regresión robusta

- Eliminar ejemplos separados más de 3σ
- Estimadores de mínimo error absoluto o de mínima mediana de error cuadrático

BÚSQUEDA DE ATRIBUTOS

Espacio de búsqueda: subconjuntos posibles de los atributos

- Con F atributos hay 2^F grupos posibles

Una exploración exhaustiva no es factible con atributos numerosos (>30)

Se puede comenzar por

- conjunto de atributos de entrada completo (backward elimination)
- conjunto vacío de atributos (forward selection)

Se puede realizar búsqueda

- en escalada (greedy): mueve 1 atributo cada vez. encuentra óptimo local
- mejor-primero: mantiene todas las ramas y puede hacer backtracking. Es exhaustivo si no se para

La evaluación de cada nodo (subconjunto de atributos) se realiza llamando al algoritmo seleccionado (wrapper) o independientemente

SELECCIÓN DE ATRIBUTOS

Algunos atributos pueden ser **redundantes** (como "salario" y "categoría laboral")

- Hacen más lento el proceso de aprendizaje (Ej: C4.5 $O(m*n^2)$ SVM $O(m*n)$)
- Pueden confundir a algunos clasificadores (como el Naive Bayes)

Otros son **irrelevantes** (como el DNI para predecir si una persona va a devolver un crédito)

- En algunos estudios, un solo atributo irrelevante (aleatorio) perjudica un 5% o 10% al clasificador (C4.5 en este caso)

Todos los esquemas se degradan por incorporar atributos irrelevantes

- máximo efecto en IBL, mínimo bayesianos
- los árboles se degradan en los niveles más bajos (menos datos)

SELECCIÓN DE ATRIBUTOS

Maldición de la dimensionalidad:

- El número de datos necesarios puede crecer exponencialmente con el número de dimensiones
- El exceso de atributos puede llevar a sobreaprendizaje, pues incrementa la complejidad del modelo en relación al número de datos disponibles

En ocasiones es útil tener el conocimiento de qué atributos son relevantes para una tarea

- Cuantos menos atributos, mas fácil de interpretar es el modelo
- Algunos algoritmos (como C4.5 árboles de decisión) son capaces de descartar atributos. Pero también existen algoritmos de selección de atributos que se pueden usar para preprocesar los datos

SELECCIÓN DE ATRIBUTOS

Hay métodos automáticos que permiten seleccionar los atributos basándose en los 3 objetivos mencionados:

- mejorar las prestaciones
- aumentar la velocidad de ejecución
- aumentar la legibilidad de la representación

Dos aspectos a considerar

- Tipo y algoritmo de búsqueda
- Evaluación de conjuntos de atributos: independientes o dependientes del esquema (“wrapper”)

TIPOS DE SELECCIÓN

La selección de atributos se puede clasificar en 3 grupos:

1. Métodos filtro: un método estadístico determina si un atributo es relevante o no respecto a la variable dependiente: Ej. Chi-cuadrado, coef. de correlación

2. Métodos Wrapper: la selección de atributos es un problema de búsqueda combinatoria, Ej. Algoritmo de eliminación recursiva de atributos

3. Métodos Embedded: utilizan métodos de aprendizaje a la vez que se construye el modelo, Ej. LASSO, Elastic Net, Ridge Regression

PROYECCIÓN DE ATRIBUTOS

- Transformaciones simples que pueden aportar una mejora significativas en prestaciones
- Ejemplos de transformaciones (no se asegura la mejora sistemática):
  - Diferencias de atributos
  - Cociente de atributos
  - Concatenación de valores de atributos nominales
  - Pertenencia a cluster
  - Adición de ruido
  - Eliminar datos aleatoriamente o selectivamente

DIFERENCIAS EN LOS ENFOQUES

Aunque ambos persiguen el mismo objetivo, no es lo mismo seleccionar atributos que reducir dimensionalidad

Reducir atributos o seleccionarlos implica escoger aquellos más representativos/relevantes

Reducir la dimensionalidad crea nuevos atributos a partir de la combinación de otros que posteriormente serán eliminados

TRANSFORMACIÓN DE DATOS

Agregación: resumen, cubos de datos

Normalización: re-escalar las variables (para distancias)

- normalización min-max

$$v' = \frac{v - min_A}{max_A - min_A}$$

- normalización estadística (tipificar)

$$v' = \frac{v - media_A}{stand\_dev_A}$$

Selección/transformación de atributos

- Discretizar
- Quitar atributos redundantes
- Proyectar a espacios reducidos: PCA

TRANSFORMACIÓN: DISCRETIZACIÓN

Discretización

- Reducir número de valores, o poner intervalos a variables continuas. Ej.: edad->(joven, adulto, mayor)
- Reduce el tamaño de datos y mejora precisión

- Misma amplitud:
  - Cajas: W = (Max-min)/N.
  - El más directo. Problemas de escala y con outliers
- Misma frecuencia:
  - Cada caja el mismo numero de muestras
- Métodos supervisados: análogo al filtro wrapper

PROYECCIÓN DE ATRIBUTOS

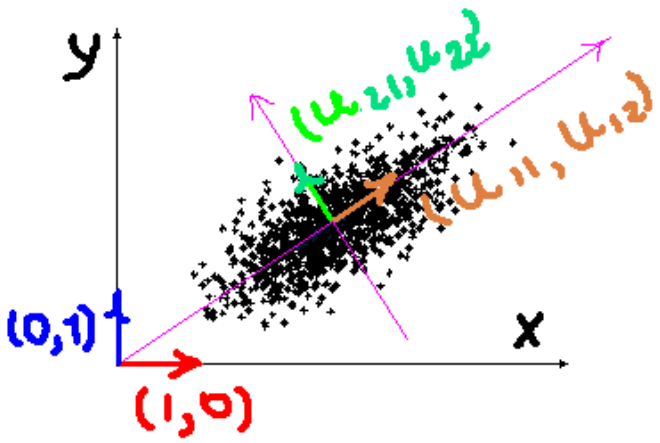
PCA: Principal Component Analysis

- Método no supervisado para identificar las direcciones principales del conjunto de datos
- Rotación de los datos sobre el sistema de coordenadas (reducido) dado por estas direcciones
- PCA es un método de reducción de dimensiones
- Algoritmo:
  1. Encontrar la dirección (eje) de máxima varianza
  2. Encontrar la dirección de máxima varianza perpendicular a la anterior y repetir
- Implementación: encontrar los autovectores de la matriz de covarianza de los datos
  - Los autovectores (ordenados por autovalores) son las direcciones
  - Detalles bien conocidos como método probabilístico

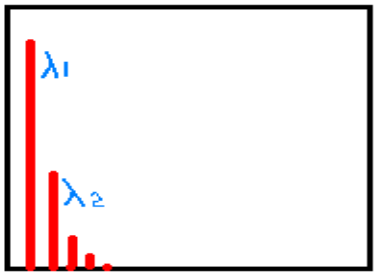
TRANSFORMACIÓN: PROYECCIÓN PCA

Dados vectores *k*-dimensionales, buscar vectores ortogonales de dimensión *c* <= *k*

Aplicable a datos numéricos de muchas dimensiones



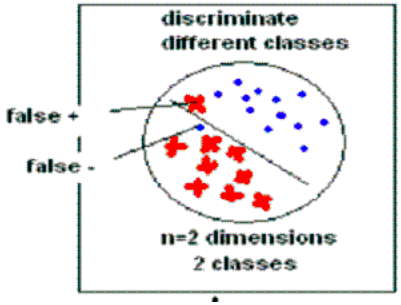
$$C = \frac{1}{n} X^t X = U \Lambda U^t = \begin{pmatrix} u_{11} & u_{21} \\ u_{12} & u_{22} \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{pmatrix}$$



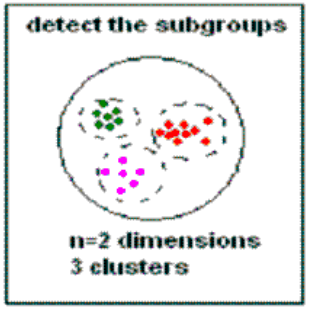
TÉCNICAS MD-AA

Técnica

- Supervisada
  - Clasificación
    - Tabla de Decisión
    - Árboles de Decisión
    - Reglas
    - Bayesiana
    - Basado en Ejemplares
    - Redes de Neuronas
  - Predicción
    - Regresión
    - Árboles de Predicción
    - Estimador de Núcleos



- No Supervisada
  - Agrupamiento
    - Numérico K-MEDIAS
    - Conceptual
    - Probabilístico
  - Asociación
    - A Priori



TIPOS DE TÉCNICAS

Paramétricas, no paramétricas

Grado de supervisión

- Supervisadas, no supervisadas, por refuerzo

Tipo de información resultante

- Simbólica, subsimbólica/numérica, mixta

Número de técnicas empleadas

- Sencillos, meta-clasificadores

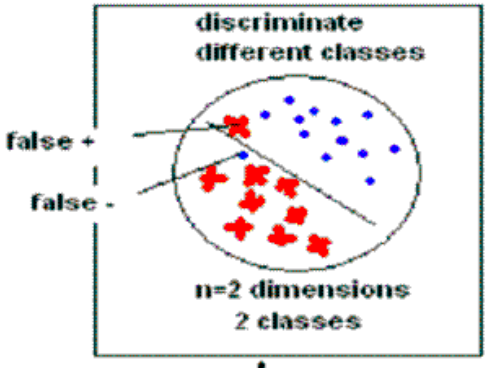
Tipo de clases

- Discretas, continuas, desconocidas

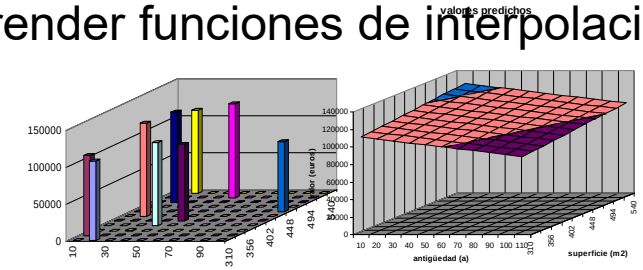
MINERÍA DE DATOS: TÉCNICAS SUPERVISADAS

Técnica

- Supervisado
  - Clasificación
    Separar instancias de cada categoría (aprender fronteras de clases)



  - Predicción
    Predecir valores numéricos (aprender funciones de interpolación)



- No Supervisado

LA CLASIFICACIÓN

La clasificación es el proceso de dividir un conjunto de datos en **grupos mutuamente excluyentes,** de tal forma que cada miembro de un grupo esté lo mas cerca posible de otros y grupos diferentes estén lo más lejos posible de otros, donde la distancia se mide con respecto a las variables especificadas, que se quieren predecir.

PRECISIÓN

La precisión de un subconjunto S de atributos para todos los ejemplos de entrenamientos se calculará

- para el caso de que la clase sea simbólica

$$precisión(S) = \frac{ejemplos\ bien\ clasificados}{ejemplos\ totales}$$

- en el caso de que la clase sea numérica:

$$precisión(S) = -RMSE = -\sqrt{\frac{\sum_{i \in I} (y_i - \hat{y}_i)^2}{n}}$$

RMSE es la raíz cuadrada del error cuadrático medio [root mean squared error], n es el número de ejemplos totales, y_i el valor de la clase para el ejemplo i y $\hat{y}_i$ el valor predicho por el modelo para el ejemplo i.

TABLAS DE DECISIÓN

La tabla de decisión constituye la forma más simple y rudimentaria de representar la salida de un algoritmo de aprendizaje, que es justamente representarlo como la entrada.

Estos algoritmos consisten en seleccionar subconjuntos de atributos y calcular su precisión [accuracy] para predecir o clasificar los ejemplos. Una vez seleccionado el mejor de los subconjuntos, la tabla de decisión estará formada por los atributos seleccionados (más la clase), en la que se insertarán todos los ejemplos de entrenamiento únicamente con el subconjunto de atributos elegido. Si hay dos ejemplos con exactamente los mismos pares atributo-valor para todos los atributos del subconjunto, la clase que se elija será la media de los ejemplos (en el caso de una clase numérica) o la que mayor probabilidad de aparición tenga (en el caso de una clase simbólica).

TABLAS DE DECISION. EJEMPLO

Ejemplo Nº	*Sexo*	*Tipo*	*Fijo*
1	Hombre	Asociado	No
2	Mujer	Catedrático	Si
3	Hombre	Titular	Si
4	Mujer	Asociado	No
5	Hombre	Catedrático	Si
6	Mujer	Asociado	No
7	Hombre	Ayudante	No
8	Mujer	Titular	Si
9	Hombre	Asociado	No
10	Mujer	Ayudante	No
11	Hombre	Asociado	No

TABLAS DE DECISION. EJEMPLO

Si se toma como primer subconjunto el formado por el atributo sexo, y se eliminan las repeticiones:

Ejemplo Nº	*Sexo*	*Fijo*
1	Hombre	No
2	Mujer	Si
3	Hombre	Si
4	Mujer	No

La probabilidad de clasificar bien es del 50%.

TABLAS DE DECISION. EJEMPLO

Si se elimina el atributo Sexo:

Ejemplo Nº	Tipo	Fijo
1	Asociado	No
2	Catedrático	Si
3	Titular	Si
7	Ayudante	No

precisión de aciertos del 100%: ésta tabla es la que se debe tomar como tabla de decisión.

El resultado es lógico ya que el atributo sexo es irrelevante a la hora de determinar si el contrato es o no fijo.

ÁRBOLES DE DECISIÓN

Un árbol de decision puede interpretarse como una serie de reglas compactadas para su representación en forma de árbol.

Cada eje está etiquetado con un par atributo-valor y las hojas con una clase, de forma que la trayectoria que determinan desde la raíz los pares de un ejemplo de entrenamiento alcanzan una hoja etiquetada con la clase del ejemplo.

La clasificación de un ejemplo nuevo del que se desconoce su clase se hace con la misma técnica, solamente que en ese caso al atributo clase, cuyo valor se desconoce, se le asigna de acuerdo con la etiqueta de la hoja a la que se accede con ese ejemplo.

ÁRBOLES DE DECISIÓN. PROBLEMAS APROPIADOS

Que la representación de los ejemplos sea mediante vectores de pares atributo-valor, especialmente cuando los valores son disjuntos y en un número pequeño.

Que el atributo que hace el papel de la clase sea de tipo discreto y con un número pequeño de valores.

Que las descripciones del concepto adquirido deban ser expresadas en forma normal disyuntiva.

Que posiblemente existan errores de clasificación en el conjunto de ejemplos de entrenamiento, así como valores desconocidos en algunos de los atributos en algunos ejemplos

EL SISTEMA ID3

Seleccionar un atributo como raíz del árbol y crear una rama con cada uno de los posibles valores de dicho atributo.

- Se selecciona el atributo que mejor separe (ordene) los ejemplos de acuerdo a las clases. Para ello se emplea la entropía.

TEORÍA DE LA INFORMACIÓN (SHANNON)

Dado un conjunto de eventos A={A1, A2,..., An}, con probabilidades {p1, p2,..., pn}

Información en el conocimiento de un suceso A_i (bits)

$$I(A_i) = log_2\left(\frac{1}{p_i}\right) = -log_2(p_i)$$

Información media de A (bits)

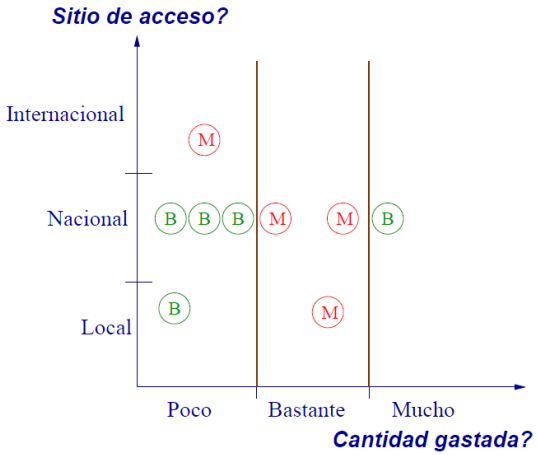
$$I(A) = \sum_{i=1}^{n} p_i I(A_i) = -\sum_{i=1}^{n} p_i log_2(p_i)$$

Ganancia de Información $G(A_i) = I - I(A_i)$

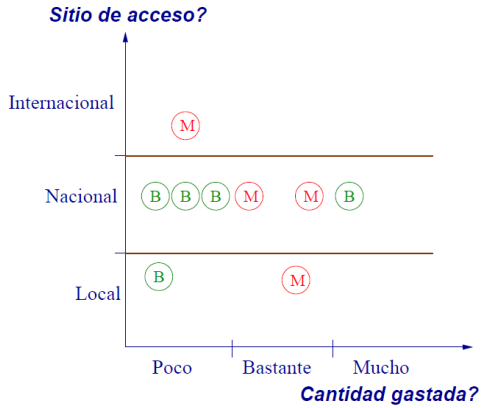
EL SISTEMA ID3. EJEMPLO

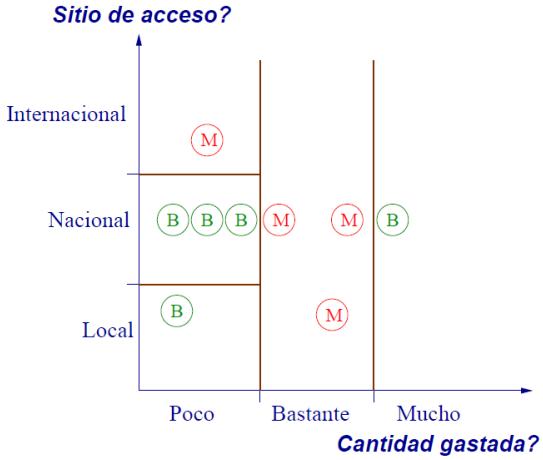
Ejemplo	Sitio de acceso A_1	1ª cantidad gastada A_2	Vivienda (zona) A_3	Última compra A_4	Clase
1	1	0	2	Libro	Bueno
2	1	0	1	Disco	Malo
3	1	2	0	Libro	Bueno
4	0	2	1	Libro	Bueno
5	1	1	1	Libro	Malo
6	2	2	1	Libro	Malo

A



B





Ejemplo	Sitio de acceso A_1	1ª cantidad gastada A_2	Vivienda (zona) A_3	Última compra A_4	Clase
1	1	0	2	Libro	Bueno
2	1	0	1	Disco	Malo
3	1	2	0	Libro	Bueno
4	0	2	1	Libro	Bueno
5	1	1	1	Libro	Malo
6	2	2	1	Libro	Malo

$$I(A_1) = \sum_{j=1}^{nv(A_1)} \frac{n_{ij}}{n} I_{ij} = \sum_{j=1}^{3} \frac{n_{ij}}{6} I_{ij} =$$
$$\frac{n_{10}}{6} I_{10} + \frac{n_{11}}{6} I_{11} + \frac{n_{12}}{6} I_{12} = \frac{1}{6} I_{10} + \frac{4}{6} I_{11} + \frac{1}{6} I_{12}$$
$$I_{10} = -\sum_{k=1}^{2} \frac{n_{10k}}{n_{10}} \log_2 \frac{n_{10k}}{n_{10}} = -\frac{1}{1} \log_2 \frac{1}{1} - \frac{0}{1} \log_2 \frac{0}{1} = 0$$
$$I_{11} = -\sum_{k=1}^{2} \frac{n_{11k}}{n_{11}} \log_2 \frac{n_{11k}}{n_{11}} = -\frac{2}{4} \log_2 \frac{2}{4} - \frac{2}{4} \log_2 \frac{2}{4} = 1$$
$$I_{12} = -\sum_{k=1}^{2} \frac{n_{12k}}{n_{12}} \log_2 \frac{n_{12k}}{n_{12}} = -\frac{0}{1} \log_2 \frac{0}{1} - \frac{1}{1} \log_2 \frac{1}{1} = 0$$

$$I(A_1) = \frac{1}{6} I_{10} + \frac{4}{6} I_{11} + \frac{1}{6} I_{12} = \frac{1}{6} 0 + \frac{4}{6} 1 + \frac{1}{6} 0 = 0,66$$

$$I(A_2) = \frac{2}{6} I_{20} + \frac{1}{6} I_{21} + \frac{3}{6} I_{22} = \frac{2}{6} 1 + \frac{1}{6} 0 + \frac{3}{6} (-\frac{2}{3} \log_2 \frac{2}{3} - \frac{1}{3} \log_2 \frac{1}{3}) = 0,79$$

$$I(A_3) = \frac{1}{6} I_{30} + \frac{4}{6} I_{31} + \frac{1}{6} I_{32} = \frac{1}{6} 0 + \frac{4}{6} (-\frac{1}{4} \log_2 \frac{1}{4} - \frac{3}{4} \log_2 \frac{3}{4}) + \frac{1}{6} 0 = 0,54$$

$$I(A_4) = \frac{1}{6} I_{4Disco} + \frac{5}{6} I_{4Libro} = \frac{1}{6} 0 + \frac{5}{6} (-\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5}) = 0,81$$

EL SISTEMA ID3. EJEMPLO

Ejemplo	Sitio de acceso A_1	1ª cantidad gastada A_2	Última compra A_4	Clase
2	1	0	Disco	Malo
4	0	2	Libro	Bueno
5	1	1	Libro	Malo
6	2	2	Libro	Malo

$$I(A_1) = \frac{1}{4} I_{10} + \frac{2}{4} I_{11} + \frac{1}{4} I_{12} = \frac{1}{4} 0 + \frac{2}{4} 0 + \frac{1}{4} 0 = 0$$

$$I(A_2) = \frac{1}{4} I_{20} + \frac{1}{4} I_{21} + \frac{2}{4} I_{22} = \frac{1}{4} 0 + \frac{1}{4} 0 + \frac{2}{4} 1 = 0,5$$

$$I(A_4) = \frac{1}{4} I_{4Disco} + \frac{3}{4} I_{4Libro} = \frac{1}{4} 0 + \frac{3}{4} (-\frac{1}{3} \log_2 \frac{1}{3} - \frac{2}{3} \log_2 \frac{2}{3}) = 0,23$$

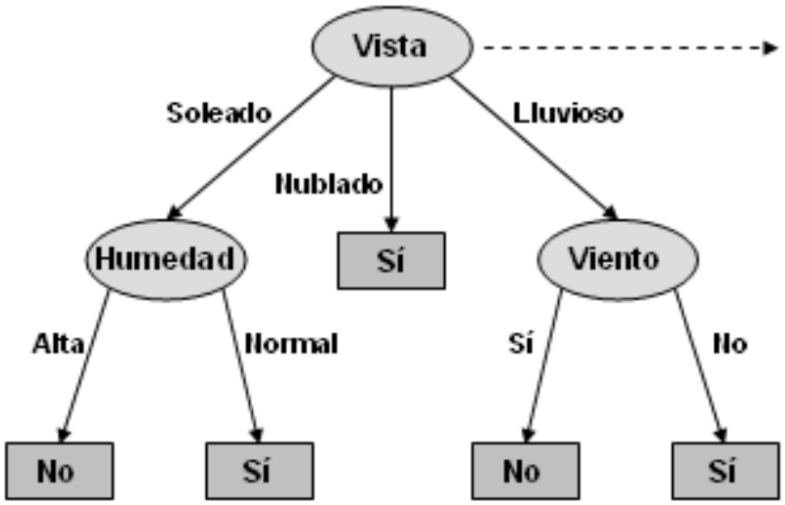
EL SISTEMA ID3. EJEMPLO

PSEUDOCÓDIGO DEL ALGORITMO ID3

1. Seleccionar el atributo A_i que maximice la ganancia $G(A_i)$.
2. Crear un nodo para ese atributo con tantos sucesores como valores tenga.
3. Introducir los ejemplos en los sucesores según el valor que tenga el atributo A_i.
4. Por cada sucesor:
   a. Si sólo hay ejemplos de una clase, C_k, entonces etiquetarlo con C_k.
   b. Si no, llamar a ID3 con una tabla formada por los ejemplos de ese nodo, eliminando la columna del atributo A_i.

EJEMPLO DE USO DEL ALGORITMO ID3

Ejemplo	Vista	Temperatura	Humedad	Viento	Jugar
1	Soleado	Alta (85)	Alta (85)	No	No
2	Soleado	Alta (80)	Alta (90)	Sí	No
3	Nublado	Alta (83)	Alta (86)	No	Sí
4	Lluvioso	Media (70)	Alta (96)	No	Sí
5	Lluvioso	Baja (68)	Normal (80)	No	Sí
6	Lluvioso	Baja (65)	Normal (70)	Sí	No
7	Nublado	Baja (64)	Normal (65)	Sí	Sí
8	Soleado	Media (72)	Alta (95)	No	No
9	Soleado	Baja (69)	Normal (70)	No	Sí
10	Lluvioso	Media (75)	Normal (80)	No	Sí
11	Soleado	Media (75)	Normal (70)	Sí	Sí
12	Nublado	Media (72)	Alta (90)	Sí	Sí
13	Nublado	Alta (81)	Normal (75)	No	Sí
14	Lluvioso	Media (71)	Alta (91)	Sí	No



$$I = -\sum_{c=1}^{nc} \frac{n_c}{n} \log_2\left(\frac{n_c}{n}\right) = -\frac{5}{14}\log_2\left(\frac{5}{14}\right) - \frac{9}{14}\log_2\left(\frac{9}{14}\right) = 0.9403$$

$$G(vista) = I - I(vista) = 0.2468$$

$$I(vista) = \sum_{j=1}^{nv(vista)} \frac{n_{vista=j}}{n} I_{vista=j} = \frac{5}{14} I_{soleado} + \frac{4}{14} I_{nublado} + \frac{5}{14} I_{lluvioso} = 0.6935$$

$$I_{vista=soleado} = -\frac{3}{5}\log_2\left(\frac{3}{5}\right) - \frac{2}{5}\log_2\left(\frac{2}{5}\right) = 0.9709$$

$$I_{vista=nublado} = -\sum_{k=1}^{nc} \frac{n_{vista=nublado,\ clase=k}}{n_{vista=nublado}} \log_2\left(\frac{n_{vista=nublado,\ clase=k}}{n_{vista=nublado}}\right) = 0$$

$$I_{vista=lluvioso} = -\sum_{k=1}^{nc} \frac{n_{vista=lluvioso,\ clase=k}}{n_{vista=lluvioso}} \log_2\left(\frac{n_{vista=lluvioso,\ clase=k}}{n_{vista=lluvioso}}\right) = 0.9709$$

$$G(temperatura) = I - I(temperatura) = 0.0292$$

$$G(humedad) = I - I(humedad) = 0.1519$$

$$G(viento) = I - I(viento) = 0.0481$$

REGLAS DE CLASIFICACIÓN

La inducción de reglas se puede lograr:

Generando un árbol de decisión y extrayendo de él las reglas

Mediante una estrategia de covering, consistente en tener en cuenta cada vez una clase y buscar las reglas necesarias para cubrir [cover] todos los ejemplos de esa clase; cuando se obtiene una regla se eliminan todos los ejemplos que cubre y se continúa buscando más reglas hasta que no haya más ejemplos de la clase.

ALGORITMO 1R

Este algoritmo genera un árbol de decisión de un nivel expresado mediante reglas:

```
1R (ejemplos) {

    Para cada atributo (A)

        Para cada valor del atributo (Ai)
            Contar el número de apariciones de cada
clase con Ai
            Obtener la clase más frecuente (Cj)
            Crear una regla del tipo Ai -> Cj
        Calcular el error de las reglas del atributo A
    Escoger las reglas con menor error

}
```

atributo	*reglas*	*errores*	*error total*
vista	Soleado → no Nublado → si Lluvioso → si	2/5 0/4 2/5	4/14
temperatura	Alta → no Media → si Baja → si	2/4 2/6 1/4	5/14
humedad	Alta → no Normal → si	3/7 1/7	4/14
viento	Falso → si Cierto → no	2/8 3/6	5/14

EVALUACIÓN DE UN SISTEMA DE APRENDIZAJE

Medida de la calidad de un esquema de análisis de datos: tasa de error de clasificación, desviación de predicción,…

Evaluación principalmente de métodos predictivos (clasificación/predicción)

Evaluación de la capacidad para **generalizar**

- La evaluación sobre las instancias de aprendizaje siempre es optimista e irreal: evaluación sobre instancias independientes

PROBLEMAS EN EVALUACIÓN

- Fiabilidad estadística de las prestaciones obtenidas ($\rightarrow$ tests de significatividad)
- Medidas de calidad:
  - Número de aciertos (clasificaciones correctas)
  - Precisión de estimadores de probabilidad
  - Error en predicción numérica
- Costes que implican los diferentes tipos de error
  - Relevante en la mayoría de aplicaciones

DILEMA DE LA EVALUACIÓN

El clasificador puede hacerse con el conjunto completo de datos de entrenamiento, pero no la evaluación

A mayor conjunto de entrenamiento mejor clasificador

A mayor conjunto de test, más precisa la estimación del error

Holdout: division del conjunto de datos en entrenamiento y test

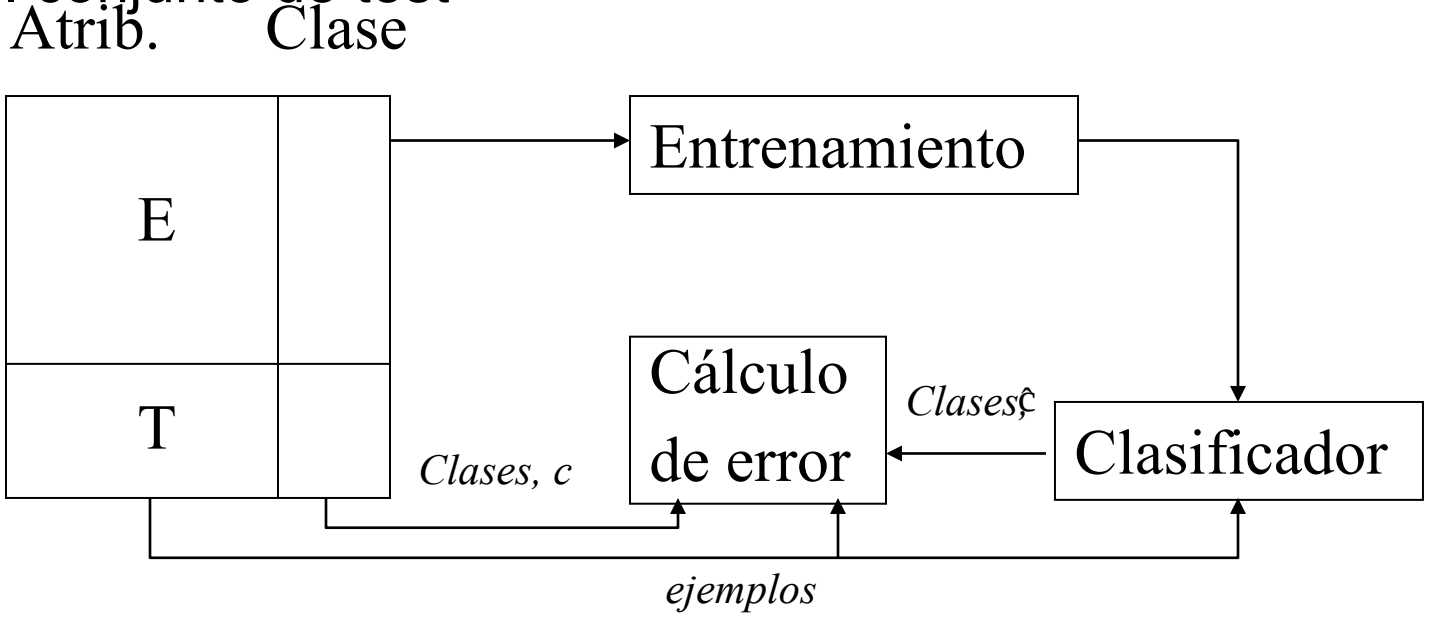
- Dilema: idealmente ambos deberían ser grandes!!

EVALUACIÓN DEL APRENDIZAJE

el conjunto de ejemplos se divide en dos partes: entrenamiento (E) y test (T)

se aplica la técnica (p.e. Naïve Bayes) al conjunto de entrenamiento, generando un clasificador

se estima el error (o tasa de aciertos) que el clasificador comete en el conjunto de test



ESTIMACIÓN CON CONJUNTO INDEPENDIENTE (HOLDOUT)

- Con un solo conjunto de datos
- Este método reserva un conjunto independiente para test y usa el resto para entrenamiento
  - Típico: un tercio test, dos tercios entrenamiento
- Problema: muestras no representativas
  - Ejemplo: alguna clase falta en el conjunto de test
- Solución: estratificación
  - Asegura que cada clase se representa con las mismas proporciones en ambos conjuntos

VALIDACIÓN CRUZADA

Problema de la evaluación: sesgo de los conjuntos E y T seleccionados

Solución: validación cruzada k -veces (k-fold cross validation)

Se divide el conjunto de ejemplos en k partes iguales, E_i

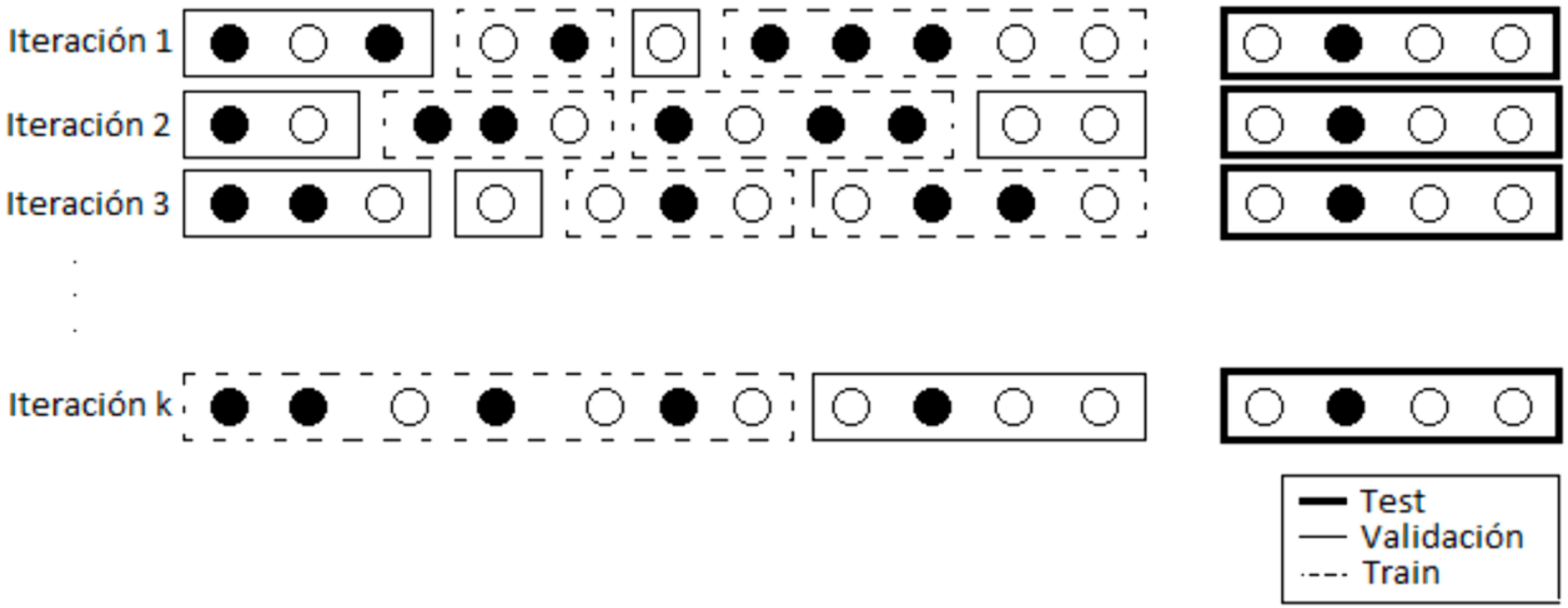
Se realiza lo siguiente k veces:

- se entrena con E - E_i (i=1.. k)
- se calcula el error con el E_i , e_i (o el éxito, f_i)

se estima la tasa de error/éxito haciendo la media de los errores

Extremo: leave-one-out

VALIDACIÓN CRUZADA: K-FOLDS



LEAVE-ONE-OUT CROSS-VALIDATION

- Leave-one-out: caso particular de of *k*-fold cross-validation:
  - Número de carpetas (folds) es el número de ejemplos de entrenamiento
  - Implica construer el clasificador *n* veces
- Máximo aprovechamiento de datos
- No hay aleatoriedad en muestreo
  - Método más costoso
  - No permite estratificación (un solo ejemplo en cada evaluación)
  - Sujeto a discusión CV vs LOO