



# Tema I: KDD - Selección de Variables

## **I.1 Proceso KDD**

## **I.2 Selección de Variables**

I.2.1 Evaluación filter

I.2.2 Evaluación wrapper

## **I.3 Métodos de búsqueda**

I.3.1 Búsqueda secuencial

I.3.2 Búsqueda estocástica

I.3.3 Búsqueda híbrida

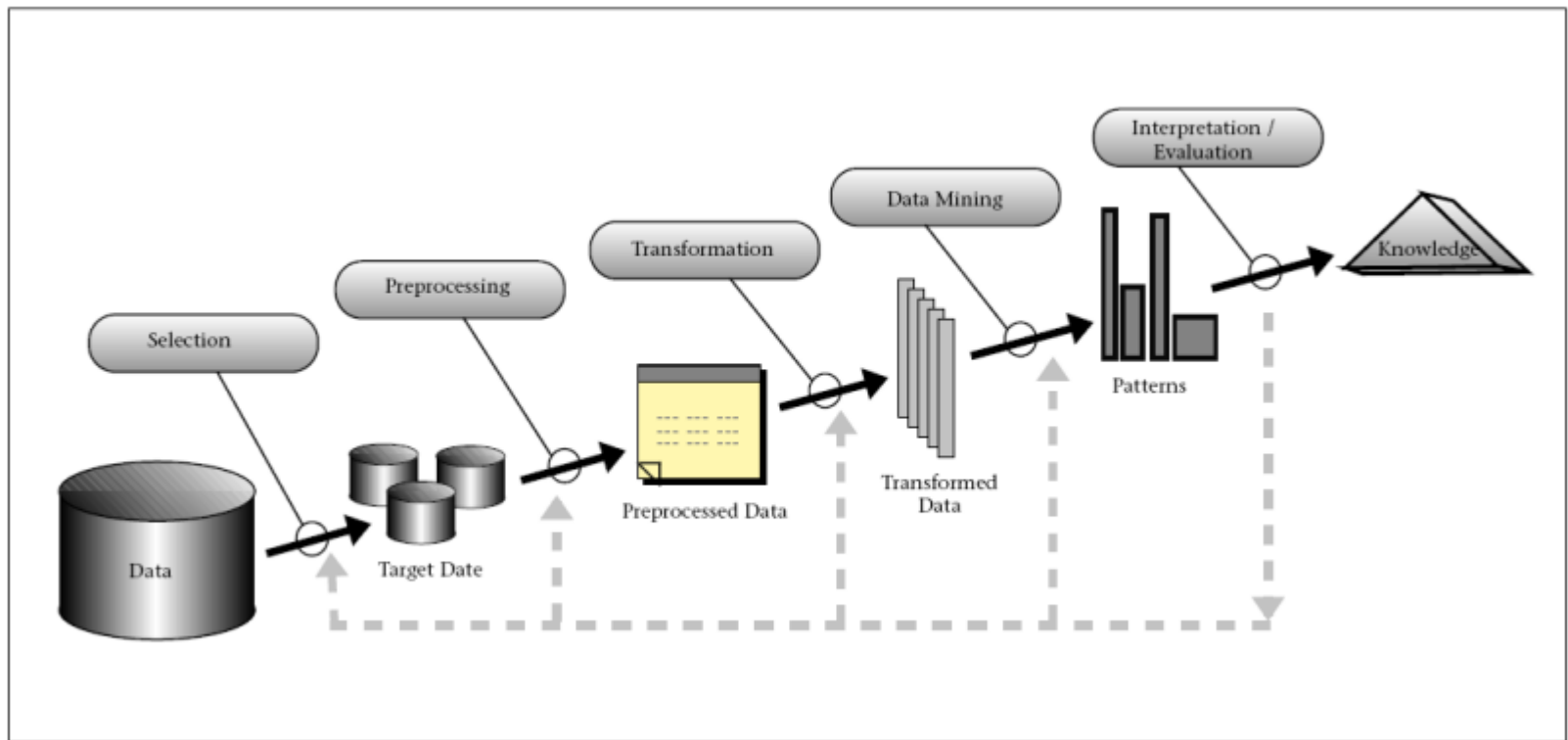
## **I.4 Herramienta Weka**

Máster Universitario en Ingeniería Informática

Pablo.Bermejo@uclm.es

# I.I Proceso KDD

- Proceso KDD: Knowledge Discovery in Databases → Descubrimiento del Conocimiento



*Fayyad 1996. American Association for Artificial Intelligence*

# I.I Proceso KDD

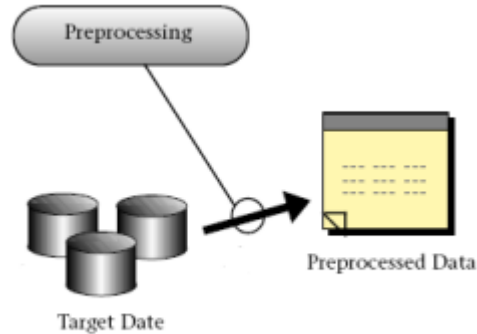


- De entre todos los datos (crudos o con formato) disponibles, debemos decidir:

- Qué buscamos
- Qué queremos estudiar
- Qué datos son irrelevantes

- Esto es a priori, más adelante puede descubrirse información que no esperábamos: registros innecesarios, variables redundantes, patrones inesperados...

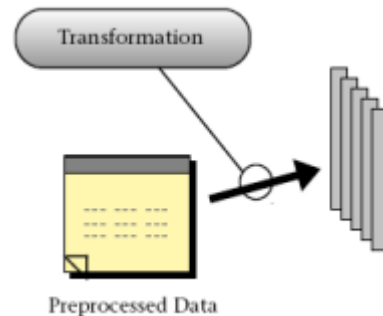
# I.I Proceso KDD



Instance ID	$X_1$	$X_2$	...	$X_n$	$C$
1	$x_{11}$	$x_{12}$	...	$x_{1n}$	$c_1$
2	$x_{21}$	$x_{22}$	...	$x_{2n}$	$c_2$
3	$x_{31}$	$x_{32}$	...	$x_{3n}$	$c_3$
4	$x_{41}$	$x_{42}$	...	$x_{4n}$	$c_4$
...	...	...	...	...	...
N	$x_{N1}$	$x_{N2}$	...	$x_{Nn}$	$c_N$

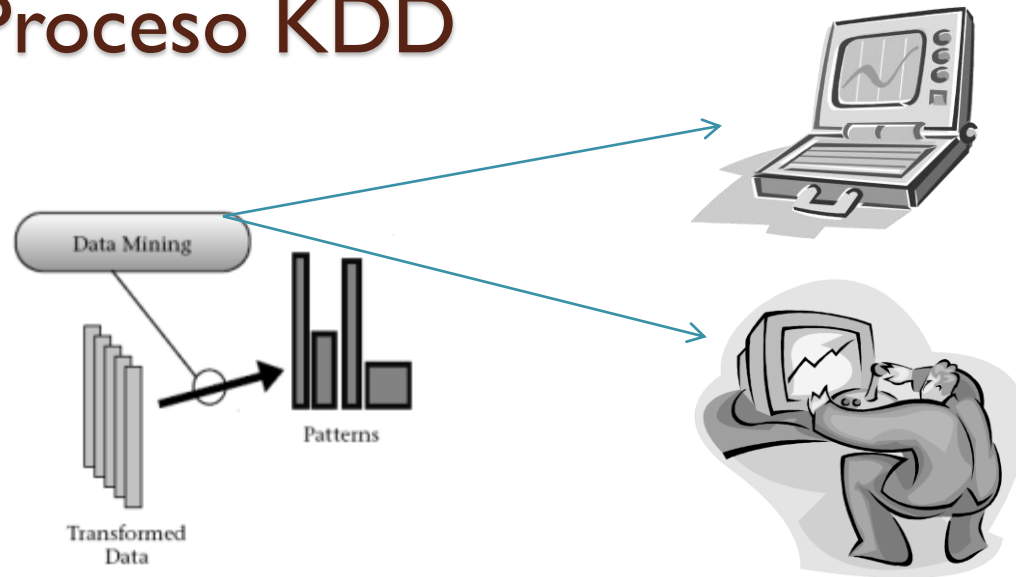
- Una vez sabemos con qué datos queremos trabajar:
  - Elegir herramienta de trabajo (Weka, Matlab, Excel...)
  - Dar el formato correspondiente a los registros
  - **Selección de variables**
  - Selección de instancias
  - Limpiar: outliers, valores perdidos...

# I.I Proceso KDD



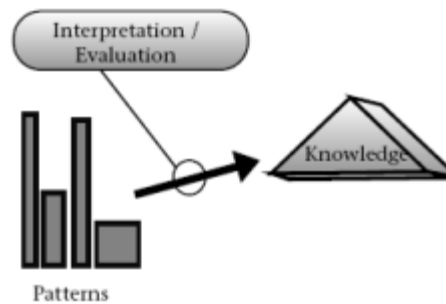
- Los valores originales de las variables pueden transformarse a un formato que mejore la eficacia de los algoritmos de aprendizaje y descubrimiento de patrones:
  - Agregación o construcción de variables
  - Discretización
  - Cambio de representación:  $tf$ ,  $tf*idf$

# I.I Proceso KDD



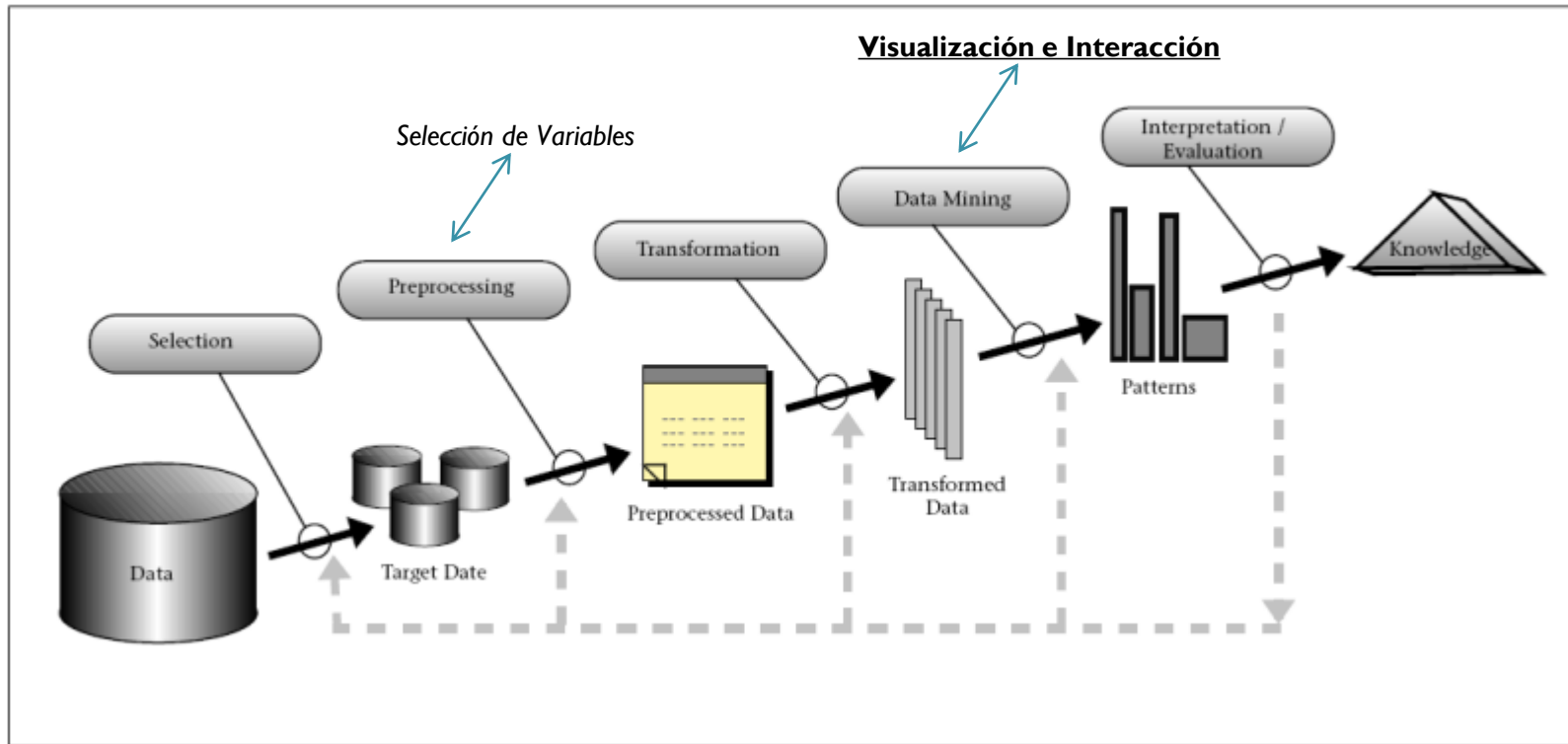
- Algoritmos **automáticos** de aprendizaje/predictivos para encontrar patrones: Clasificación, Regresión, Reglas, Clusters.
- El **humano** también puede encontrar patrones utilizando *técnicas de visualización e interacción* adecuadas:
  - Proyección geométrica
  - Iconográficas
  - Jerárquicas
  - Basadas en píxeles

# I.I Proceso KDD



- ¿Qué indican los patrones encontrados?
- ¿Qué tasa de ciertos aportan estos patrones en nuevos registros?
- ¿Cuánto tiempo se tarda en encontrar los patrones?
- ¿Es información útil?
- Para mejorar los resultados: volver a alguna o todas de las fases anteriores.

# I.I Proceso KDD



- Normalmente los registros se describen con varios atributos o variables.
- **La dimensionalidad afecta directamente a las técnicas de Visualización** → Selección de Variables es muy importante



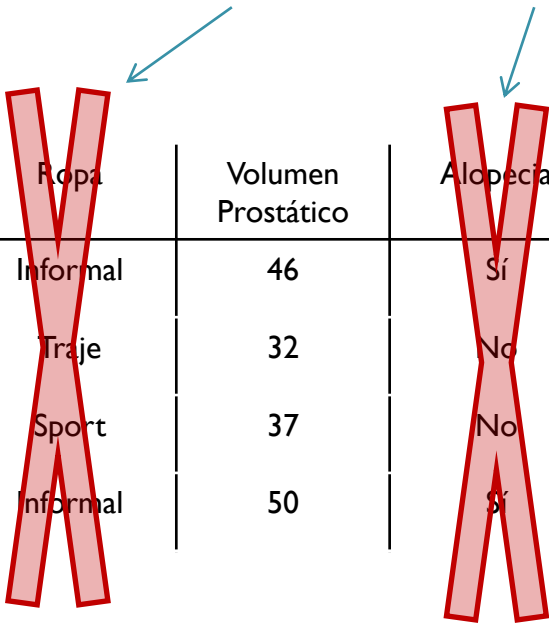
## I.2 Selección de Variables

- Proceso generalmente conocido como FSS (Feature Subset Selection)
- **FSS** es el proceso de identificar las variables relevantes a un algoritmo de minería de datos.
- Cuando la selección se realiza atendiendo a lo bien que los algoritmos de minería de datos predicen una variable en concreto (variable clase) hablamos de **FSS supervisada**.

Fuma	Edad	Ropa	Volumen Prostático	Alopecia	<b>Cáncer Próstata</b>
Sí	75	Informal	46	Sí	<b>Sí</b>
No	56	Traje	32	No	<b>No</b>
Sí	60	Sport	37	No	<b>No</b>
No	80	Informal	50	Sí	<b>Sí</b>

## I.2 Selección de Variables

- FSS elimina las variables más irrelevantes o redundantes.



Fuma	Edad	Ropa	Volumen Prostático	Alopecia	Cáncer Próstata
Sí	75	Informal	46	Sí	Sí
No	56	Traje	32	No	No
Sí	60	Sport	37	No	No
No	80	Informal	50	Sí	Sí

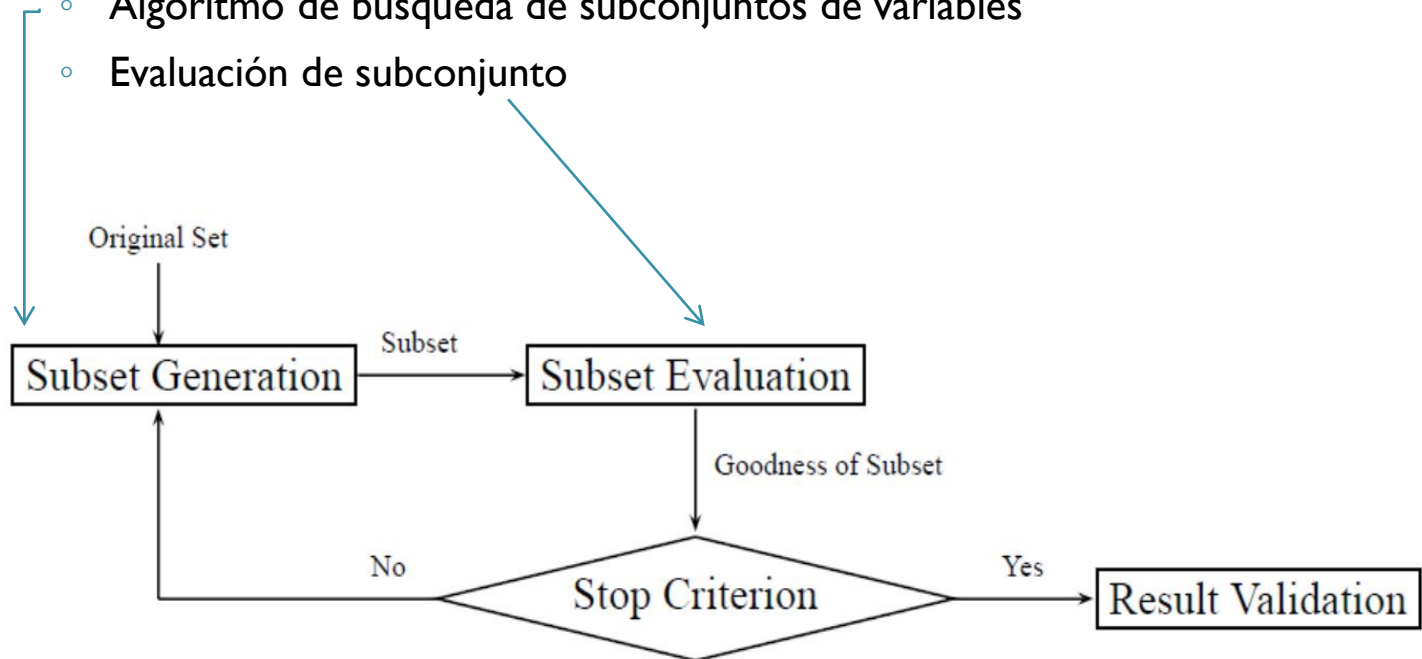
- Este caso es muy claro y puede hacerlo un humano, pero por lo general las Independencias no son tan obvias y es preferible utilizar **algoritmos de FSS automáticos**

## I.2 Selección de Variables

- Al realizar FSS sobre nuestra base de datos esperamos aprovecharnos de las siguientes **ventajas**:
  - Eliminar variables que afectan el poder predictivo de otras (*maldición de la dimensionalidad*); por ejemplo:
    - los estadísticos aprendidos se afectan por las probabilidades conjuntas de una variable importante con otra no importante
    - Reglas predictivas con condiciones sin sentido
  - Mayor facilidad de generalización
  - Los algoritmos aplicados en la fase de Minería de Datos son más rápidos:
    - Construir Clasificador
    - Proyección geométrica a 2D desde 5D es más rápido que desde 8D
  - El modelo aprendido o visualizado se más sencillo de interpretar
  - Reduce la base de datos sin alterar la representación

## I.2 Selección de Variables

- FSS consta de 2 fases interrelacionadas:
  - Algoritmo de búsqueda de subconjuntos de variables
  - Evaluación de subconjunto



## I.2 Selección de Variables

- **Evaluación de subconjunto** de variables: cálculo de una métrica que representa la bondad de ese subconjunto:
  - Métodos Filter
  - Métodos Wrapper

Los **métodos filter** utilizan propiedades intrínsecas de los datos:

- Estadísticas: tests, probabilidades conjuntas,...
- Basadas en Teoría de la Información: entropía, ganancia de información, ...
- Son rápidas
- La métrica calculada es independiente de cualquier modelo de clasificación o regresión a utilizar posteriormente.
- Hay estudios que relacionan ciertas métricas con el tipo de bondad a medir del clasificador (Information Gain mejora la Precisión)

## I.2.1 Selección de Variables – evaluación filter

- Evaluación **univariada**: 1 variable respecto a la variable clase
- Evaluación **multivariada**: 1 nueva variable junto a las seleccionadas hasta ahora, respecto a la clase.
- Una variable relevante de forma univariada puede hacer perder la bondad de un subconjunto relevante de forma multivariada
- Métricas Filter Univariadas:

$$Entropy(X) = H(X) = - \sum_{i=1}^n p(x_i) \log_2 p(x_i)$$

**Info Gain (IG)**: mide el cambio de entropía de la variable clase C cuando se conoce el valor de la variable X:

$$IG(C, X) = H(C) - H(C|X)$$

## I.2.1 Selección de Variables – evaluación filter

**Info Gain Ratio(IGR):** penaliza el valor IG cuando la variable X tiene muchos estados (posibles valores)

$$IGR(C, X) = \frac{IG(C, X)}{H(X)}$$

**Symmetrical Uncertainty (SU):** IG normalizada en el intervalo [0,1]

$$SU(C, X) = 2 \times \frac{IG(C, X)}{H(C) + H(X)}$$

**Chi-Cuadrado ( $\chi^2$ ):** test estadístico, cuya hipótesis de partida es que X y C son independientes.

## I.2.1 Selección de Variables – evaluación filter

- Métricas Filter Univaridas Condicionales, impracticable, pero existen aproximaciones:

Conditional IG-Battiti:

$$IG(X_i, C|S') = IG(X_i, C) - \beta \sum_{s \in S'} IG(X_i, s)$$

Conditional IG-Peng:

$$IG(X_i, C|S') = IG(X_i, C) - \frac{1}{|S'|} \sum_{s \in S'} IG(X_i, s)$$



## I.2.1 Selección de Variables – evaluación filter

- Métricas Filter Multivariadas:

**Correlation-based Feature Selection (CFS):** evalúa SU entre pares de variables  $X_i$  y cada  $X_i$  respecto a la clase.

$$CFS(X_1, \dots, X_n) = \frac{\sum_{i=1}^n SU(C, X_i)}{\sqrt{\sum_{i=1}^n \sum_{j=1}^n SU(X_i, X_j)}}$$

## I.2.2 Selección de Variables – evaluación wrapper

Los **métodos de evaluación wrapper** ejecutan, para cada subconjunto candidato de variables, la fase de minería de datos seleccionada para construir un modelo (clasificador, regresión, reglas) predictivo el cual se aplicará a nuevos datos.

- Más lentos que los métodos filter
- Suelen obtener mejores subconjuntos

Los aciertos y fallos del modelo creado se plasman en la *Matriz de Contingencia o Confusión*:

		Predicted			
		a	b	c	d
Real	a	6	2	0	1
	b	0	7	2	1
	c	1	0	3	0
	d	2	0	0	5

TP

TN

FP

FN

## I.2.2 Selección de Variables – evaluación wrapper

- Métricas Wrapper:

**Accuracy:** es la tasa de aciertos al predecir el valor de la variable clase C

$$Accuracy = \frac{\sum_c |C| TP(c)}{\sum_c |C| TP(c) + FP(c)}$$

**Precisión para el valor c de la variable C:** probabilidad de clasificar correctamente los registros cuya variable clase tiene el valor c.

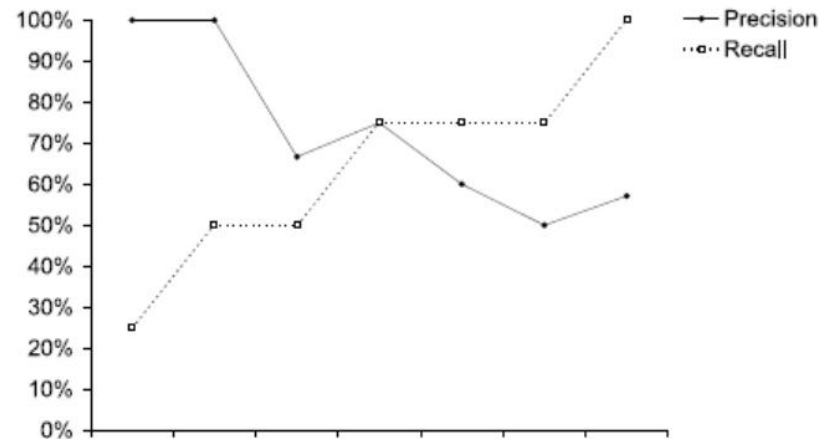
$$Precision(c) = \frac{TP(c)}{TP(c) + FP(c)}$$

**Recall para el valor c de la variable C:** probabilidad de clasificar todos los registros con C=c como c, sin importar los fallos en los demás registros.

$$Recall(c) = \frac{TP(c)}{TP(c) + FN(c)}$$

## I.2.2 Selección de Variables – evaluación wrapper

- Breakeven point:



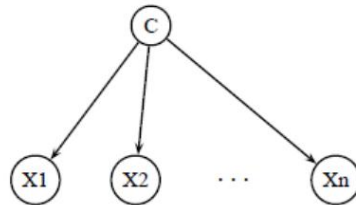
**F<sub>β</sub>-measure:** media entre Precisión y Recall. Si  $\beta=1$ , media armónica.

$$F_{\beta} - measure(c) = (1 + \beta^2) \times \frac{Precision(c) \times Recall(c)}{\beta^2 * Precision(c) * Recall(c)}$$

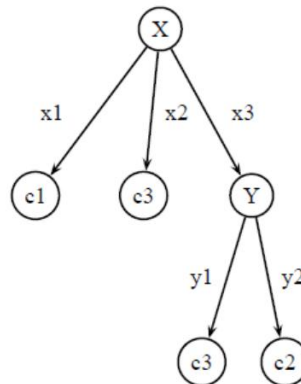
## I.2.2 Selección de Variables – evaluación wrapper

- Existe una gran variedad de modelos predictivos a construir.

**Clasificadores Bayesianos:** probabilidad conjunta de las variables predictivas, dado  $c$ .  
Caso más simple, Naive Bayes.

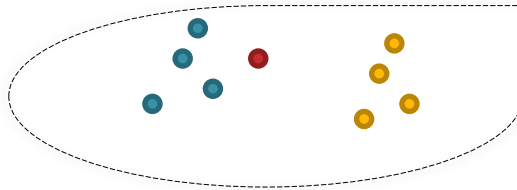


**Árboles de clasificación:** una variable predictiva por nivel del árbol; los nodos hoja son el valor predicho para la variable clase.



## I.2.2 Selección de Variables – evaluación wrapper

**Vecinos cercanos:** los registros se proyectan en el espacio multidimensional y se calcula la distancia de un nuevo registro al resto.



**Regresión lineal:** expresión matemática que captura una relación lineal entre la variable clase y las variables predictivas

$$c = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k + \varepsilon$$

## I.3 Métodos de Búsqueda

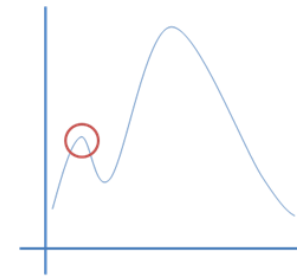
### Búsqueda Completa o Exhaustiva

Se evalúan todas las posibles combinaciones de subconjuntos de variables. Si nuestra base de datos tiene  $n$  variables, el número de evaluaciones es  $2^n$ . Para bases de datos medianas o grandes, esto es intratable.

### Búsqueda Secuencial o Determinista

Siempre devuelve la misma solución. Construye un subconjunto a partir de la evaluación del anterior, hasta cumplir un criterio de parada:

- Número máximo de evaluaciones
- Tiempo
- Bondad de la solución
- Puede atascarse en un máximo local

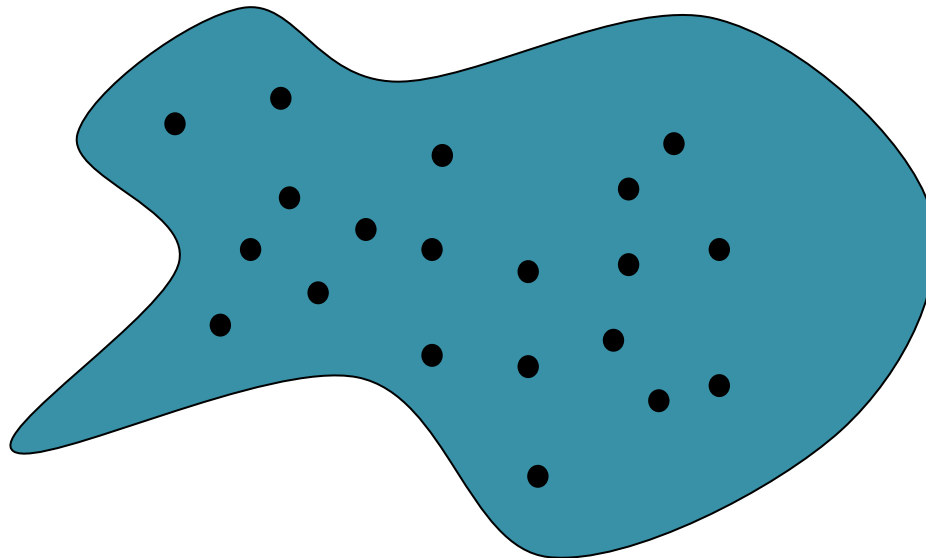


## I.3 Métodos de Búsqueda

### Búsqueda Estocástica o Aleatorizada

Normalmente no devuelve la misma solución en distintas ejecuciones.

- Pueden abordar la búsqueda de forma secuencial o global.
- Más posibilidades de encontrar el máximo global
- Para la creación de nuevas soluciones, además de las últimas evaluaciones se utilizan métodos heurísticos.





## 1.3.1 Búsqueda Secuencial

- Algoritmos de Búsqueda Secuencial

### Fast Correlation-Based Filter (FCBF):

- 1) Crea un ranking filter por SU y selecciona las variables según un umbral fijado
- 2) Entre las seleccionadas, elimina las redundantes según otra métrica filter.

### Sequential Forward Selection (SFS):

- 1) Comienza con el conjunto vacío
- 2) Va añadiendo la variable que mejor evaluación devuelve, hasta que ninguna nueva variable aumenta el mejor valor encontrado.

### Sequential Backward Selection (SBS):

- 1) Comienza con todas las variables
- 2) Va eliminando la variable que mejor evaluación devuelve, hasta que ninguna eliminación aumenta el mejor valor encontrado.

## I.3.1 Búsqueda Secuencial

### Hill-Climbing (HC):

- 1) Comienza desde el conjunto vacío u otro dado
- 2) Va añadiendo o eliminando la variable que mejor evaluación devuelve, hasta que ningún paso aumenta el mejor valor encontrado.

### Mutual Information-based Feature Selection (MIFS):

- 1) Es un SFS que utiliza como evaluación filter la *IG Conditional-Battiti*
- 2) Para al seleccionar un número  $k$  de variables preestablecido

## I.3.2 Búsqueda Estocástica

- Algoritmos de Búsqueda Global Estocástica

### Algoritmos Genéticos (GAs):

- 1) Se crean  $n$  soluciones de forma aleatoria y se evalúan
- 2) Basándose en la evaluación de una generación, se eligen soluciones para realizar:
  - Cruces
  - Mutaciones
  - Torneos
- 3) Criterio de parada: número de generaciones o convergencia de las evaluaciones

## I.3.2 Búsqueda Estocástica

### Algoritmos de Estimación de Distribuciones (EDAs):

- 1) Se crean  $n$  soluciones de forma aleatoria y se evalúan
- 2) Basándose en la evaluación de una generación, se eligen soluciones para aprender la distribución de cada variable.
- 3) Se muestrea una nueva generación.
- 4) Criterio de parada: número de generaciones o convergencia de las evaluaciones

## I.3.2 Búsqueda Estocástica

### Optimización de Colonias de Hormigas (ACO):

- 1) Se crean  $n$  soluciones de forma aleatoria y se evalúan
- 2) Los caminos que construyen los mejores subconjuntos tienen mayor nivel de feromona
- 3) Las soluciones se van construyendo siguiendo en la mayoría de los casos los niveles de feromonas
- 4) Criterio de parada: número de iteraciones o convergencia de las evaluaciones

*Ejemplo de Visualización de un algoritmo de búsqueda de caminos basado en ACOs*

## I.3.3 Búsqueda Híbrida

- Nuevos **métodos de búsqueda híbrida** han aparecido recientemente para aprovechar las ventajas de las evaluaciones filter y wrapper, y librarse de los inconvenientes de ambos.
- Basados en ranking: métrica filter para crear ranking; búsqueda wrapper sobre un subconjunto

### Algoritmos de Búsqueda Secuencial Híbrida

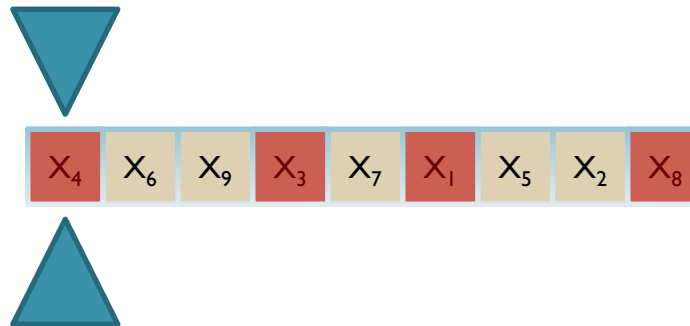
#### Linear Forward Selection (LFS):

- 1) Ranking de variables por SU
- 2) Búsqueda SFS con evaluación wrapper

## I.3.3 Búsqueda Híbrida

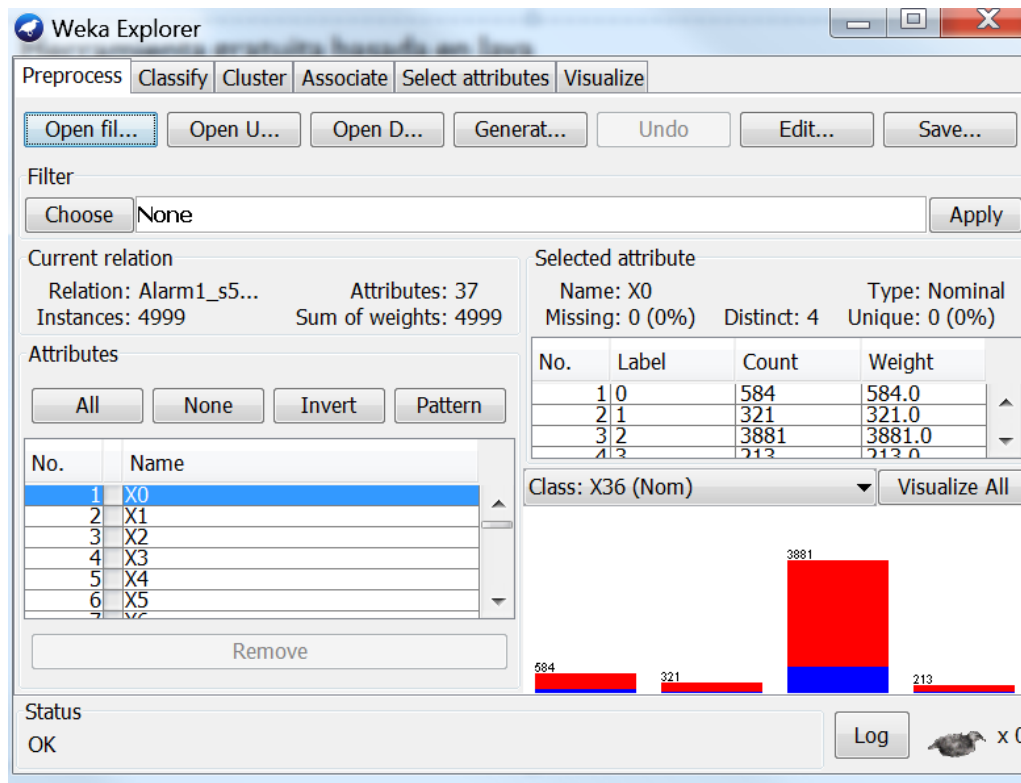
Best Incremental Ranked Subset for Feature Selection (BIRS):

- 1) Ranking de variables por SU
- 2) Búsqueda incremental añadiendo por orden cuando mejora la evaluación wrapper



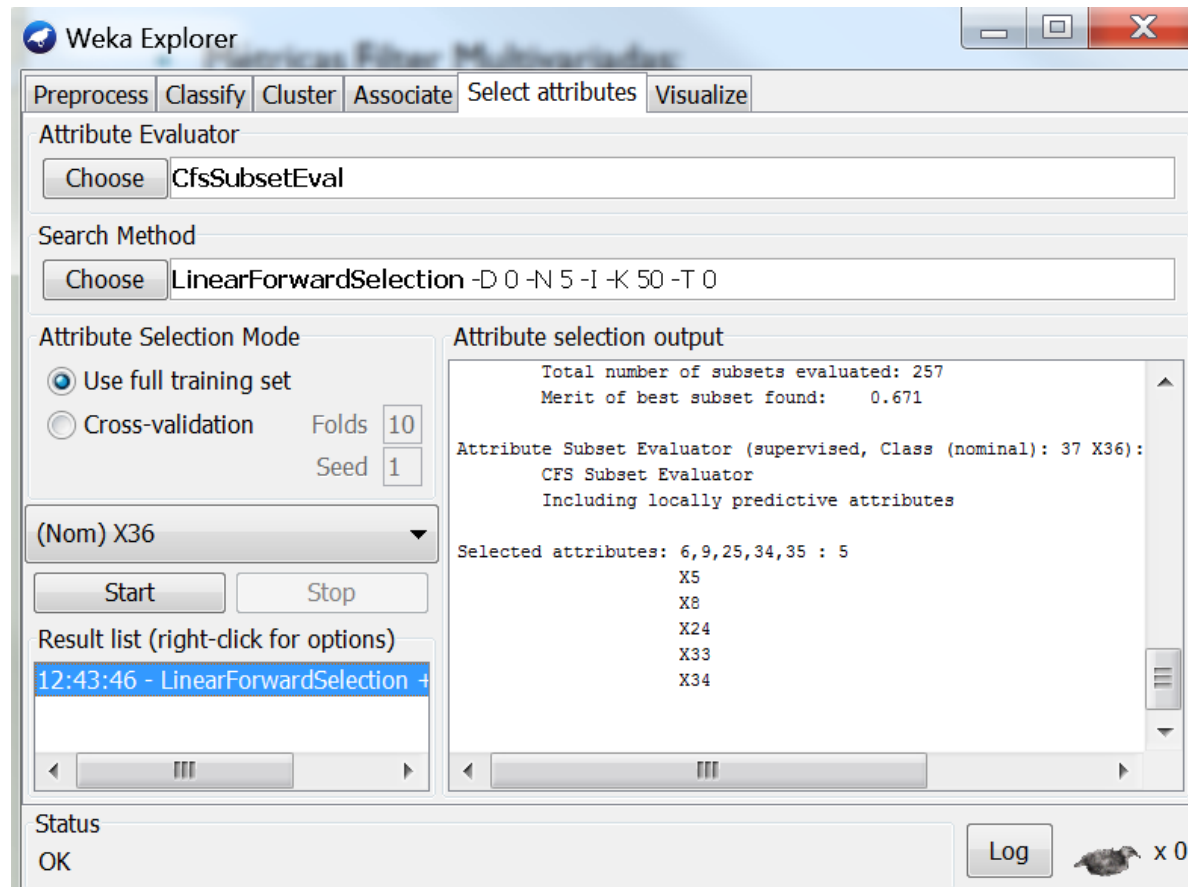
# 1.4 Herramienta Weka

- Herramienta gratuita basada en Java
- Permite realizar casi todas las fases del proceso KDD

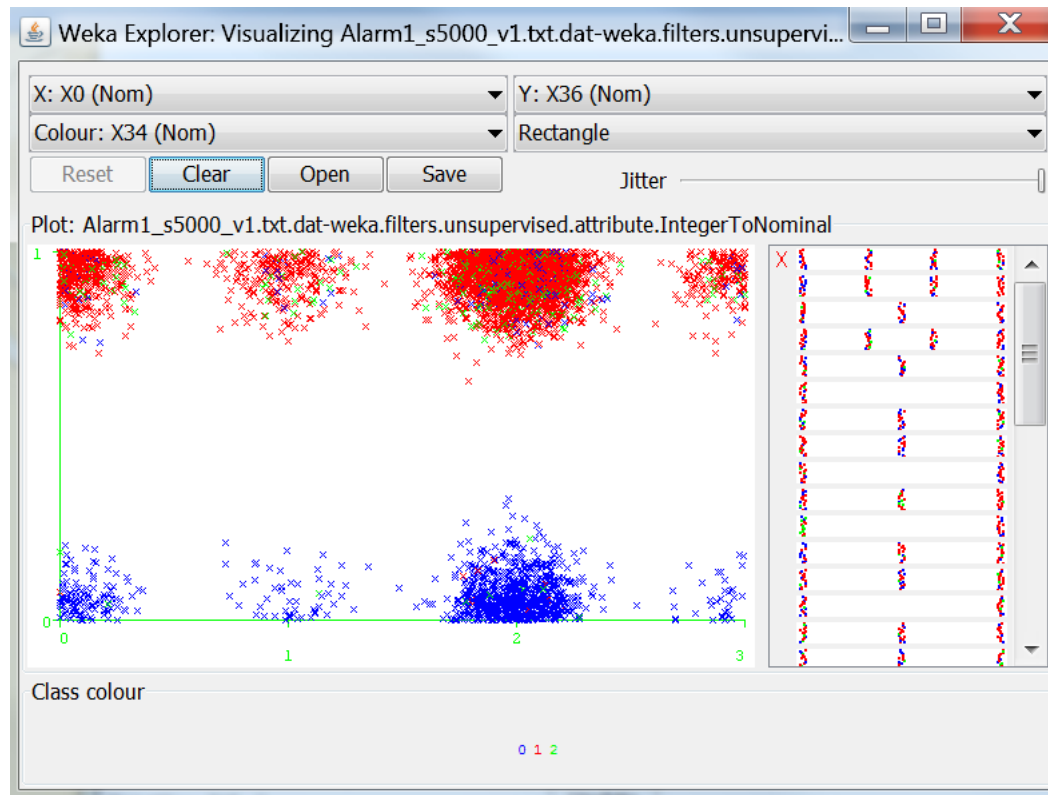




## I.4 Herramienta Weka

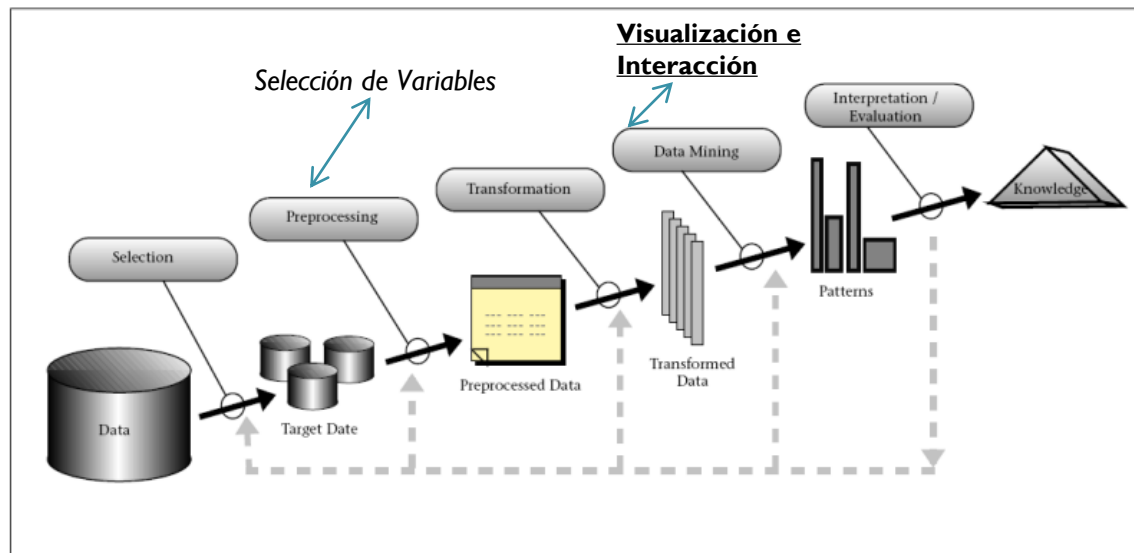


## I.4 Herramienta Weka



# Conclusiones

- Las técnicas de visualización a mostrar en el curso intentan presentar en 2 dimensiones los registros de  $n > 3$  dimensiones.
- Cuanto más dimensiones, peor funciona la visualización y más difícil es la interpretación e interacción
- La solución es aplicar la técnica de Selección de Variables previamente a la visualización.



# ¿Te ha quedado claro?

- Si tienes una base de datos de alta dimensionalidad (1000 variables), ¿qué tipo de evaluación usarías en el algoritmo de búsqueda
  - si tu máquina tiene pocos recursos?
  - si no te importa el coste, sino que quieres los mejores resultados posibles?
- ¿Qué ventajas tiene un proceso de FSS
  - al crear un modelo predictivo?
  - al visualizar la base de datos?