

# Memoria de la práctica 2:

## Implementación del Árbol de Decisión (ID3)

**Autor:** Víctor Goicoechea Enrique

Algoritmo ID3


Ramas generadas

TiempoExterior	Condición 1	Condición 2	Condición 3	Jugar
soleado	alta	caluroso	falso	no
soleado	alta	caluroso	verdad	no
soleado	alta	templado	falso	no
soleado	normal	frio	falso	si
soleado	normal	templado	verdad	si
nublado	caluroso	alta	falso	si
nublado	caluroso	normal	falso	si
nublado	frio	normal	verdad	si
nublado	templado	alta	verdad	si
lluvioso	falso	templado	alta	si
lluvioso	falso	templado	normal	si
lluvioso	falso	frio	normal	si
lluvioso	verdad	frio	normal	no
lluvioso	verdad	templado	alta	no

Buscar una rama Salir

Buscar rama

TiempoExterior  Temperatura  Humedad  Viento

 no

Buscar Volver

### 1. Lenguaje utilizado

En la práctica elegí el lenguaje de Java ya que tengo una gran experiencia con este lenguaje y puedo usar el paquete Swing de Java para hacer una interfaz amigable y sencilla de entender para el usuario.

## 2. Procedimiento seguido para su implementación

Lo primero que hice fue leer las especificaciones de la práctica y tomando nota para saber, cuando empezase a programar, que estructuras de datos serían las necesarias para hacer que el programa funcionase correctamente y de la forma más eficiente posible.

Algunas de esas estructuras fueron:

- Clase TDatos, donde almacenare los datos necesarios para que solo con tener acceso a esta clase con los datos rellenos, pueda hacer una búsqueda recursiva, ya que tiene como atributos:
  - TDato padre, por el cual accedemos al padre del TDato hijo donde nos encontremos, sabremos que hemos llegado al final, cuando este sea NULL.
  - String con el nombre del dato, por ejemplo: caluroso, lluvioso, verdad, etc
  - ArrayList<String[]>, donde se guardara la posible matriz restante que se puede generar después de haber calculado el merito y de no haber llegado al final de la rama.
- Clase TResultadosOperaciones, donde se guardará en un String el nombre del dato y se ira haciendo el conteo de cuantas veces aparece el dato, cuantos casos son positivos y negativos, para que a la hora de calcular el merito podamos acceder de una forma más sencilla y ordenada.
- El atributo listaTDatos de tipo ArrayList<TDatos>, lista que almacenará los diferentes TDatos que genere el algoritmo ID3, para poder ser mostrados en una tabla por la interfaz gráfica. Este atributo se ira inicializando en cada llamada al método primeraVuelta(), guardando la información que pudiera contener en un atributo del mismo tipo, pero de forma local, para poder ser tratado por el algoritmo.
- Hay un método llamado primeraVuelta, que será el encargado de gestionar y hacer los cálculos tales como la formula infor, el merito de cada tipo de dato y de guardar todos estos datos en sus correspondientes atributos mencionados anteriormente (TDatos, TResultadosOperaciones, listaTDatos).
- El método darVueltas(), se encargará de hacer llamadas recursivas al método primeraVuelta(), pasandole los datos necesarios para que este puede hacer sus correspondientes cálculos explicados arriba. También detectara cuando no se puedan dar más vueltas y devolverá un false para salir del bucle donde es llamado.
- El método search(), se encargará de hacer la búsqueda de una rama de árbol de decisión, con datos que introducirá el usuario por la interfaz gráfica. Si es exitosa la búsqueda devolverá un String con la respuesta y si no tuvo éxito, devolverá un NULL, siendo tratada más adelante en la capa de presentación.

### 3. Ampliaciones realizadas

Realice las ampliaciones:

A --- Implementar todos los niveles de recursividad.

B --- Comprobar el correcto funcionamiento del algoritmo para los ejemplos de la tabla.

Para más información de cómo como probar la aplicación leer el Manual del Usuario, adjuntado a la entrega.