CIENCIA COGNITIVA
Aprendizaje inductivo. Arboles de decisión. ID3

Alvaro Barreiro, Roi Blanco

Dept. Computación
Universidade da Coruña

En esta práctica se trabajará con una variante del algoritmo de aprendizaje de árboles de decisión **ID3** de **Quinlan**, desarrollado por **Andrew Colin**. Esta variante trata con atributos con valores reales y, a partir de un conjunto de ejemplos clasificados, induce árboles binarios donde los nodos suponen el test sobre los atributos:

```
if  (atributo > umbral)  then ... else if ... else ...
```

La implementación consta de tres archivos: id3.c, id3.h, proto.h. Para compilarlo debe introducirse la opción de linkado de la librería matemática.

```
gcc -o id3 id3.c -lm
```

También se proporcionan los archivos `sample-spam.dat` y `sample-spam.tag`. En el primero por cada línea aparecen un ejemplo clasificado como *spam* o *no spam*. El segundo indica que para cada ejemplo los atributos son:

1. Número de caracteres en el mensaje.
2. Presencia (valor 1) o ausencia (valor 0) del string `http` en mayúsculas o minúsculas en el mensaje.
3. Presencia (valor 1) o ausencia (valor 0) del string `click` en mayúsculas o minúsculas en el mensaje.
4. Presencia (valor 1) o ausencia (valor 0) del string `free` een mayúsculas o minúsculas en el mensaje.
5. El número de letras mayúsculas en el mensaje.
6. El número de caracteres dólar ($) en el mensaje.

7. El número de caracteres de exclamación (!) en el mensaje. El último valor en cada línea es el valor de clasificación del ejemplo: 0 *no spam*, 1 *spam*.

En esta práctica se realizarán las siguientes tareas:

1. Aplicar el inductor sobre los ejemplos de muestra:

   ```
   id3 sample-spam
   ```

   y dibujar el árbol inducido.

2. Construir `espera-restaurante.dat` y `espera-restaurante.tag` con los ejemplos del tema de aprendizaje inductivo de árboles de decisión del texto de **Russell y Norvig** vistos en clase. Aplicar el inductor sobre estos ejemplos y dibujar el árbol inducido. Compararlo con el obtenido en el texto y comentar las diferencias.

3. A partir del árbol inducido en [1] construir el clasificador, es decir, un programa que ante la presencia de un nuevo mensaje nos diga si es o no spam. Para ello, es necesario implementar **una** de las siguientes alternativas:

   - Tranformar en código la expresión producida por el inductor. Este clasificador tomará como entrada un archivo de ejemplos con el formato `.dat` pero donde el último valor en cada línea es 2 (con el significado de ejemplo sin clasificar) en lugar 1 (*spam*) o 0 (*no-spam*) y producirá una copia del archivo de ejemplos con todos los ejemplos clasificados (1 ó 0 en el último valor de cada línea) de acuerdo con la salida del inductor.
   - Un clasificador genérico, que interprete la salida del inductor para generar un árbol de decisión de forma dinámica para realizar la clasificación de los ejemplos.
   - Un programa que reconozca cualquier árbol que pueda generar el inductor y que además tome como entrada el archivo con formato `.tag` para leer cualquier tipo expresión inducida.

4. Probar el clasificador con los ejemplos del conjunto de entrenamiento `sample-spam.dat`. El clasificador debe clasificar correctamente todos los ejemplos del conjunto de entrenamiento.

5. Se suministran también los archivos `SpamTestSet` y `NonSpamTestSet` que contienen ejemplos de mensajes spam y no spam, y que usaremos para evaluar el clasificador. Para ello es necesario procesar estos archivos de texto para producir los archivos correspondientes en formato `.dat`. Una vez obtenida la representación de los mensajes con este formato, deben pasarse al clasificador y el software de evaluación debe producir el % de clasificaciones correctas para cada categoría y el % de clasificaciones correctas total.

6. En el conjunto original de entrenamiento cambiar la clasificación de **1, 4 y 10** ejemplos. Para cada caso obtener el árbol inducido, el clasificador y evaluarlo sobre *SpamTestSet* y *NonSpamTestSet*. ¿Cuáles son las conclusiones de este experimento?

Debe construirse una breve memoria con las descripción de los ejercicios, resultados y conclusiones de cada apartado que se entregará en las horas de prácticas los días 29 y 31 de mayo de 2006. No se aceptarán entregas de prácticas fuera de plazo.