

Aproximación evolutiva a la inducción constructiva basada en expresiones algebraicas

Manuel Baena García, Rafael Morales Bueno
y Carlos Cotta Porras

Workshop MOISES
Septiembre 2004

Contenido

- Inducción constructiva
- Algoritmos evolutivos: Programación genética
- Método TDECI
- Resultados

Inducción constructiva

- Generación de nuevos atributos.
- Simplificar el espacio de representaciones mediante la generación de nuevos atributos, eliminando atributos irrelevantes y mediante la abstracción de valores para determinados atributos.

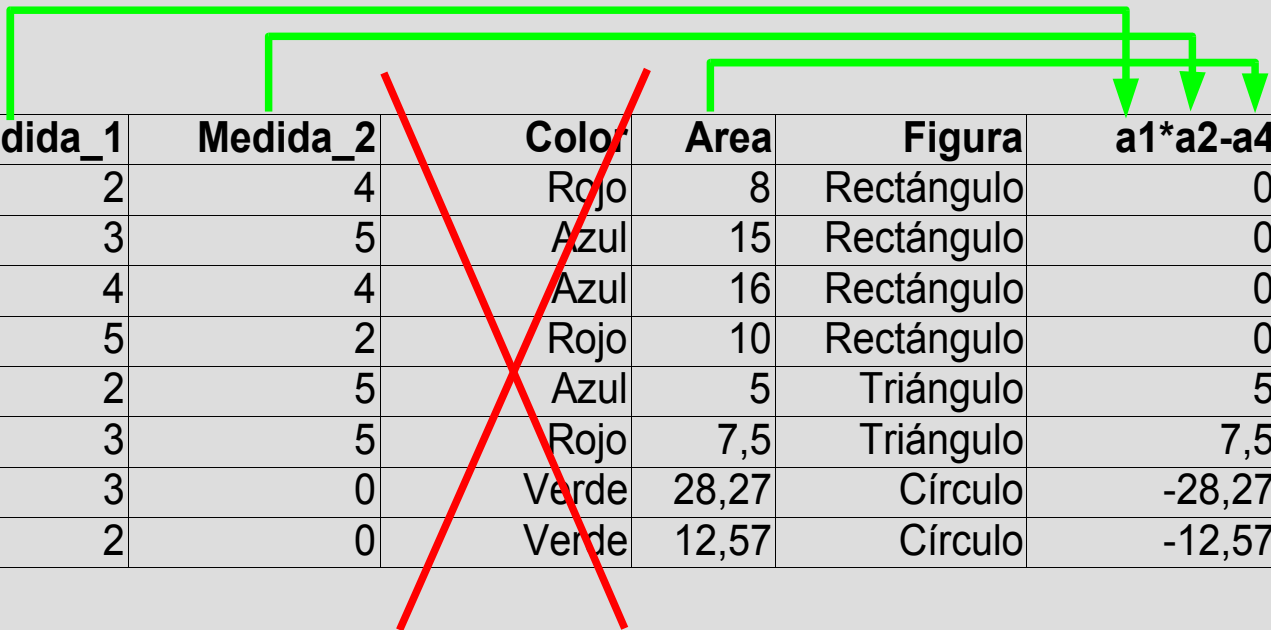
Inducción constructiva

Ejemplo: Identificación de figuras

| Medida_1 | Medida_2 | Color | Area | Figura |
|----------|----------|-------|-------|------------|
| 2 | 4 | Rojo | 8 | Rectángulo |
| 3 | 5 | Azul | 15 | Rectángulo |
| 4 | 4 | Azul | 16 | Rectángulo |
| 5 | 2 | Rojo | 10 | Rectángulo |
| 2 | 4 | Azul | 4 | Triángulo |
| 3 | 5 | Rojo | 7,5 | Triángulo |
| 3 | 0 | Verde | 28,27 | Círculo |
| 2 | 0 | Verde | 12,57 | Círculo |

Inducción constructiva

Ejemplo: Identificación de figuras



| Medida_1 | Medida_2 | Color | Area | Figura | $a1*a2-a4$ |
|----------|----------|-------|-------|------------|------------|
| 2 | 4 | Rojo | 8 | Rectángulo | 0 |
| 3 | 5 | Azul | 15 | Rectángulo | 0 |
| 4 | 4 | Azul | 16 | Rectángulo | 0 |
| 5 | 2 | Rojo | 10 | Rectángulo | 0 |
| 2 | 5 | Azul | 5 | Triángulo | 5 |
| 3 | 5 | Rojo | 7,5 | Triángulo | 7,5 |
| 3 | 0 | Verde | 28,27 | Círculo | -28,27 |
| 2 | 0 | Verde | 12,57 | Círculo | -12,57 |

Inducción Constructiva

- Michalski, R. (1980). Pattern recognition as rule-guide inductive inference. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2, 349-361.
- Zheng, Z. (2000). Constructing X-of-N Attributes for Decision Tree Learning. *Machine Learning*, 40, 35-75.

Algoritmos evolutivos

- Algoritmos heurísticos para optimización, búsqueda y aprendizaje automático

Algoritmo Evolutivo

t:= 0;

inicializar [P(t)];

evaluar [P(t)];

mientras no terminar **hacer**

 P'(t) := variación [P(t)];

 evaluar [P'(t)];

 P(t+1) := seleccionar [P'(t) \cup Q];

 t:=t+1;

fin mientras

Algoritmos evolutivos

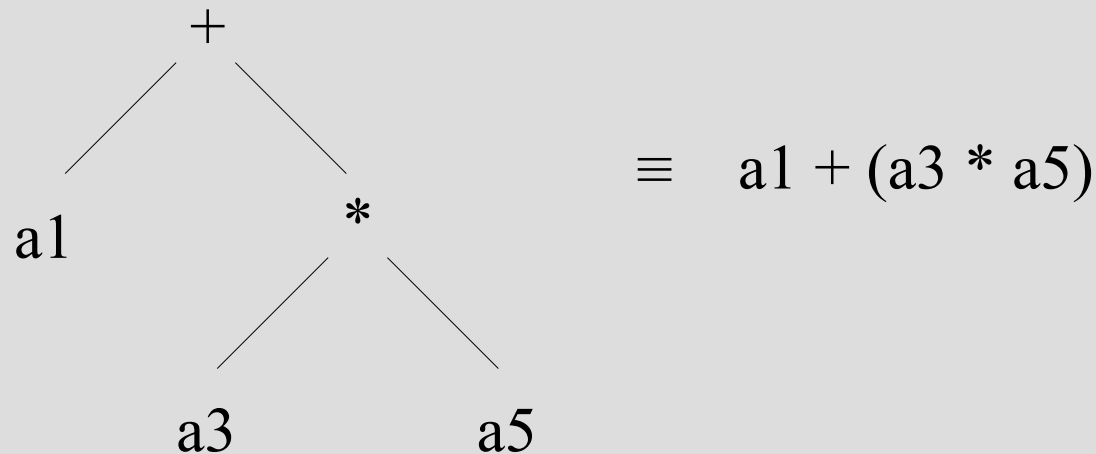
Metodología de Aplicación.

Para usar un AE en general deben definirse los siguientes pasos:

- Definición del genotipo (codificación de las vbles. del problema)
(binario, real, árbol sintáctico, longitud fija/variable, etc.)
- Diseñar la función de adecuación (fitness)
(Introducir conocimiento del problema)
- Determinar el conjunto de operadores y sus parámetros
(tipo de selección, cruce, mutación, reemplazo, etc.)
- Definir el criterio de parada
(máximo número de generaciones, error aceptable, etc.)

Programación Genética

- Algoritmos evolutivos cuyos individuos representan árboles sintácticos.



TDECI (Top Down Evolutionary Constructive Induction)

- Es un algoritmo de aprendizaje basado en árboles de decisión y programación genética.
- Para la generación del árbol se utiliza como base el algoritmo C4.5.
- En cada nodo de decisión se genera un nuevo atributo mediante búsqueda evolutiva que mejore la ganancia de información del atributo elegido por C4.5.

TDECI: Individuos

- Árboles sintácticos formados por:
 - Subconjunto de atributos.
 - Operadores: +, -, *, /, sin, cos, log, exp
 - Generador aleatorio de constantes.

TDECI. Función de Adecuación

```
if (GananciaInformacion(Actual) >= mediaDeGanancia) {  
  if (RatioGanancia(Actual) > RatioGanancia(Mejor)) {  
    fitness = 0.56*(1 - RatioGanancia(Actual)) + 0.1* (numeroAtributos) + 0.34*  
    (tamanoExpresion);  
  } else {  
    fitness = 1.0 + 0.59*(1 - RatioGanancia(Actual)) + 0.01* (numeroAtributos) +  
    0.4*(tamanoExpresion);  
  }  
} else {  
  fitness = 2.0 + (mediaDeGanancia - GananciaInformacion(Actual)) +  
  tamanoExpresion;  
}
```

TDECI: Implementación

- Weka 3: Data Mining Software in Java
<http://www.cs.waikato.ac.nz/~ml/weka/>
- ECJ 11: A Java-based Evolutionary Computation and Genetic Programming Research System
<http://cs.gmu.edu/~eclab/projects/ecj/>

Resultados experimentales

- Base de datos “Vowel” (UCI)
- 528 experiencias de entrenamiento, 462 experiencias de prueba, 10 atributos continuos, 11 clases
- Pasamos de un árbol C4.5 con precisión 39,39% y 63 hojas a un árbol TDECI con precisión 54,98% y 22 hojas

| Method | Train | Test |
|------------------------------------|--------------|--------------|
| Square node network, 88 units | | 54,8 |
| Gaussian node network, 528 units | | 54,6 |
| 1-NN, Euclides, raw | 99,24 | 56,3 |
| Radial Basis Function, 528 units | | 53,5 |
| Gaussian node network, 88 units | | 53,5 |
| FSM Gauss, 10CV na treningowym | 92,6 | 51,94 |
| Square node network, 22 | | 51,1 |
| Multi-layer perceptron, 88 hidden | | 50,6 |
| Modified Kanerva Model, 528 units | | 50 |
| Radial Basis Function, 88 units | | 47,6 |
| Single-layer perceptron, 88 hidden | | 33,3 |
| C4.5 (J48) | | 39,4 |
| TDECI | 99,05 | 54,98 |

Bibliografía

- Goldberg D.E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley.
- Bloedorn, E. and Michalski R.S., Data-Driven Constructive Induction, *IEEE Intelligent Systems, Special issue on Feature Transformation and Subset Selection*, pp. 30-37, March/April, 1998.
- Shafti L., Pérez E. Genetic Approach to Constructive Induction Based on Non-algebraic Feature Representation. M. R. Berthold et al. (Eds.): IDA 2003, LNCS 2810, pp.509 - 520, Springer-Verlag Heidelberg, 2003.