

MÁQUINA DE APRENDIZAJE PARA LA GENERACIÓN DE REGLAS BORROSAS  
(MACHINE LEARNING FOR FUZZY RULE GENERATION)

MARIA M. GARCÍA LORENZO, PEDRO Y. PIÑERO PÉREZ, LETICIA ARCO GARCÍA,

YAILE CABALLERO MOTA, RAFAEL BELLO PÉREZ

*Universidad Central “Marta Abreu” de las Villas. Facultad de Matemática Física y Computación.*

*Dirección: Carretera a Camajuaní Km 71/2, Santa Clara Villa Clara, CUBA Fax: 53-42-281608*

*E-mails: mmgarcia@uclv.edu.cu, ppp@uclv.edu.cu*

**Resumen**— El aprendizaje automatizado es un área de la Ciencia de la Computación donde el fenómeno del aprendizaje es estudiado, y es un campo interdisciplinario con conexiones a la inteligencia artificial, teoría de la información, estadísticas, ciencias cognitivas, pero además, con conexiones a la neurofisiología, psicología y biología. En esta investigación se aplican las técnicas del aprendizaje automatizado en la creación de un generador de reglas borrosas a partir de datos numéricos y simbólicos. Esto se logra con la creación e implementación de los nuevos algoritmos MLRUL y GENRUL5, el desarrollo e implementación de una extensión al algoritmo ID3 clásico y la implementación del algoritmo reportado en la tesis doctoral del Dr. Detlef Nauck [NAU00]. Se implementan además los algoritmos para el discretizado de datos: discretizar por igual frecuencia por intervalo y discretizar por intervalos de igual tamaño. Se desarrollan métodos para obtener funciones de membresía triangulares, campanas Beta y simbólicas a partir de los atributos discretizados y el porcentaje de solapamiento entre las curvas especificado en cada caso.

**Abstract**— Machine Learning is a Computer Science area where the learning process is studied. It is an interdisciplinary field with connections to artificial intelligence, theory information, statistics, cognitive science, but also, with connections to neurophysiology, psychology and biology. The purpose of this paper is to present some machine learning techniques that could effectively be applied for the creation of a fuzzy rule generator starting from numeric and symbolic data. This was achieved through: the implementation of new-created algorithms (GENRUL5 and MLRUL), the implementation of an ID3 extension which we developed and the implementation of an existing algorithm in Dr. Detlef Nauck's thesis. Derived from continuous values we implemented two algorithms for the obtaining of discrete values. One of them with a similar number of frequencies per interval and the other with intervals of equal size. Besides, from discrete attributes and the percentage of overlapping we propose methods to build Beta and triangle membership functions. The future prospect of the generator could be that of serving as a module within a neuro-fuzzy system or a fuzzy system.

**Keywords**— Machine Learning; Máquina de aprendizaje; MLRUL; GENRUL5; RulGeneratorML

## 1- Introducción

Una de las tareas más desafiantes en la Ciencia de la Computación es construir máquinas o programas de computadoras que sean capaces de aprender. Se espera una máquina o programa que aprendan automáticamente con la experiencia acumulada durante su ciclo de vida. Una de las primeras máquinas de aprendizaje que fue construida es el perceptron por Frank Rosenblatt.

El aprendizaje automatizado no sólo se encarga del aprendizaje, sino también de la forma en que se representa el conocimiento obtenido en ese proceso. Una de las formas de representación del conocimiento son las reglas, como consecuencia se desarrollan algoritmos del aprendizaje automatizado que logran a partir de datos obtener reglas.

En este trabajo se desarrolla e implementa el sistema RuleGeneratorML para la generación de reglas borrosas a partir de datos numéricos y simbólicos. Se hace un breve análisis del estado del arte sobre la generación de conocimiento usando el aprendizaje automatizado. Se

implementan métodos para la obtención de funciones de membresía a partir de atributos discretizados. Se desarrolla e implementa una extensión al algoritmo ID3 clásico que permite generar reglas borrosas, se implementa el algoritmo citado en [NAU00] con el mismo propósito y se crean e implementan los algoritmos GENRUL5 y MLRUL.

## 2- El aprendizaje en la generación de reglas

El concepto “aprendizaje” puede ser formulado como un problema de búsqueda completa, en un espacio de hipótesis predefinido; hasta encontrar la hipótesis que más se adapte a los ejemplos de entrenamiento según el conocimiento adquirido a través de experiencias anteriores.

La adquisición de modelos para el aprendizaje incluye aquellos basados en estadísticas, lógica, estructuras neuronales, algoritmos de búsqueda heurística y algoritmos de aprendizaje automatizado.

En este artículo se presenta un sistema que permite al usuario hacer uso de diversos algoritmos que de forma automática identifican patrones en datos

observados con el objetivo de hacer predicciones sobre datos jamás vistos.

Existen diversas formas de representar el conocimiento; entre ellas las reglas constituyen una de las más usadas pues su modo de representación es el del razonamiento lógico de los seres humanos, por ejemplo una regla dura tiene la forma :

*If* está lloviendo y voy a salir *then* llevar capa.

En la bibliografía se reporta la existencia de varios algoritmos de aprendizaje automatizado para la generación de reglas duras, que generalicen determinado conocimiento. [QUI96] [TOM97]

Existe un tipo particular de regla que son las llamadas reglas borrosas o difusas, que utilizan aspectos de la lógica borrosa en la representación del conocimiento.

Una regla borrosa If-then tiene la forma:

If  $x$  es  $A$  then  $y$  es  $B$ ,

donde  $A$  y  $B$  son *valores lingüísticos* definidos por conjuntos borrosos sobre los universos de discurso  $X$  e  $Y$  respectivamente.

En esta regla se describe la relación  $R$  entre las variables lingüísticas  $x$  e  $y$ , que toma valores en el espacio producto  $X \times Y$ .

Los conjuntos borrosos son funciones que toman valores en un dominio determinado y tienen imagen en el intervalo  $[0,1]$ , el valor de la imagen indica el grado de pertenencia de los elementos del dominio al conjunto especificado. Un grado 0 significa que ese valor no está en el conjunto, mientras que un grado 1 indica que el valor es representativo del conjunto. [COX98]

Se les llama reglas borrosas mixtas a aquellas donde las variables lingüísticas que intervienen en sus antecedentes, las forman términos lingüísticos con universo de valores numéricos y simbólicos.

Las reglas borrosas son la base de los sistemas borrosos o difusos para la toma de decisiones y la clasificación.

### 2.1- Generación de reglas duras

Una tendencia actual en el aprendizaje automatizado para la generación de reglas duras es el uso de algoritmos de inducción por medio de particiones recursivas representadas a través de árboles de decisión, tales como ID3 [TOM97] y su extensión C4.5 [QUI96]. En un árbol de decisión cada camino desde la raíz a un nodo hoja constituye una regla.

Como desventaja del ID3 se reporta que sólo trabaja con datos simbólicos, por su parte el C4.5 mejora al ID3 incorporando el tratamiento de datos numéricos. No obstante, ambos algoritmos generan reglas duras.

Estos algoritmos construyen un árbol de decisión top-down, en la expansión de los hijos. La expansión se lleva a cabo por determinado atributo seleccionado previamente mediante el cálculo de la entropía  $E$ , y la ganancia  $G$ .

$$E(S) = - \sum_{i=1}^c P_i \log_2 P_i \quad (1.1)$$

donde,  $P_i$  es la proporción de  $S$  perteneciente a la clase  $i$ .

La ganancia por su parte es una medida de efectividad de un atributo en clasificar el conjunto de entrenamiento se define como:

$$G(S, A) = E(S) - \sum_{v \in \text{val}(A)} \frac{|S_v|}{|S|} E(S_v) \quad (1.2)$$

donde,  $\text{val}(A)$  es el conjunto de valores posibles por el atributo  $A$  y  $S_v$  es el subconjunto de  $S$  para el cual  $A$  tiene el valor  $v$ , es decir,  $S_v = \{s \in S | A(s) = v\}$ .

Otros trabajos [SET00] muestran que las reglas también pueden ser obtenidas por medio del uso de las redes neuronales artificiales. Los algoritmos utilizados con estos fines no son los favoritos para la extracción de reglas, puesto que las redes neuronales se comportan como cajas negras, por tanto carecen de explicatividad.

No obstante, experimentos realizados permiten afirmar que las reglas extraídas desde redes neuronales son comparables con las extraídas desde los árboles de decisión en términos de cantidad de reglas, exactitud de las predicciones y número promedio de condiciones de la regla.

### 2.2- Generación de reglas borrosas

Algunos autores han desarrollado algoritmos que posibilitan la generación de reglas borrosas a partir de conjuntos de casos, incorporando a las reglas las ventajas que brindan el uso de los conjuntos borrosos. En [HON98] se propone un método de aprendizaje para derivar automáticamente reglas borrosas y funciones de membresía. Este algoritmo tiene como dificultades:

- Requiere que los valores del rasgo objetivo sean ordenables. No permite el uso de datos simbólicos, sólo numéricos.
- Trabaja sólo con funciones de membresía triangulares. Además en la búsqueda de funciones de membresía se construyen funciones de membresía iniciales, procedimiento que ante problemas reales acarrea un elevado costo computacional.

En [NAU00] se describen también varios algoritmos para la generación de reglas borrosas. Algoritmos con complejidad  $O(s^2)$  que permiten solamente el tratamiento de datos numéricos y otro algoritmo de complejidad  $O(s^3)$  que posibilita el trabajo con datos numéricos y simbólicos de forma simultánea.

### 3- RuleGeneratorML Sistema para la generación de reglas borrosas.

Como parte de nuestro trabajo se implementó el sistema RuleGeneratorML que se comporta como una caja negra generando reglas borrosas a partir de datos numéricos y simbólicos previamente especificados en un fichero que contiene al conjunto de entrenamiento.

El sistema desarrollado permite la generación de reglas de cuatro formas diferentes como se muestra en la Fig1.

Cada una de las formas de generación de reglas guarda una estrecha relación con algoritmos de aprendizaje automatizado implementados en nuestro sistema.

Se implementó una extensión al ID3 clásico. Se implementó el algoritmo desarrollado por el Dr. Detlef Nauck y reportado en [NAU00]. Se desarrollaron e implementaron los algoritmos GENRUL5 y MLRUL desarrollados por los autores y reportados en [ARC01].

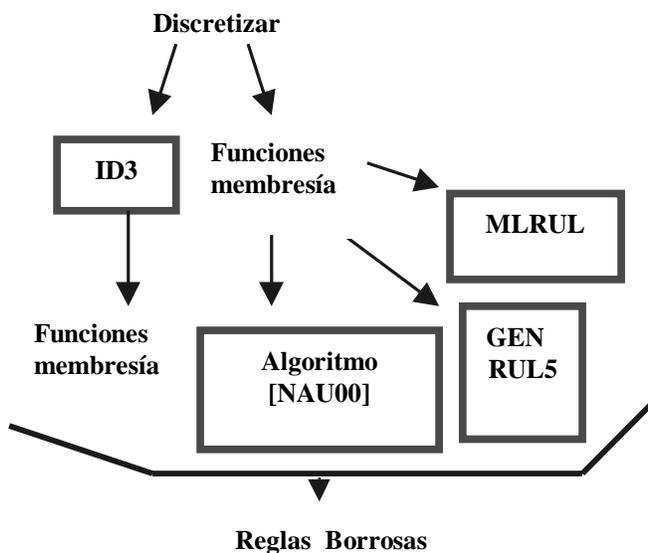


Figura 1: Esquema de los algoritmos implementados en el sistema de generación de reglas borrosas.

#### 3.1- Proceso de discretización de atributos

Cualquiera de las formas para la generación automática de reglas necesita de un proceso de

discretizado de los atributos. El usuario puede escoger durante la corrida del sistema RuleGeneratorML uno de los dos métodos para discretizar atributos que se mencionan a continuación:

- Intervalos de igual tamaño: donde se discretiza teniendo en cuenta la amplitud de las particiones, de forma tal que dicha amplitud sea la misma para todos los intervalos del dominio.
- Igual frecuencia por intervalo: donde el objetivo es lograr que todas las particiones tengan la misma cantidad de elementos o casos.

Se implementaron dos mecanismos para el discretizado de los atributos, aunque el sistema permite la incorporación de otros mecanismos para el discretizado de forma fácil y rápida.

#### 3.2- Funciones de membresía.

Como habíamos explicado las funciones de membresía forman una parte importante de las reglas borrosas. Nuestro sistema posibilita que el usuario especifique el tipo de función de membresía a utilizar por cada atributo entre tres tipos disponibles: funciones de membresía triangulares (Fig.2), funciones de membresía campana Beta (Fig.3), funciones de membresía simbólicas. (Fig.4)

$$\text{Triángulo}(x, a, b, c) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ \frac{c-x}{c-b} & b \leq x \leq c \\ 0 & c \leq x \end{cases}$$

Figura 2: Función de membresía triangular.

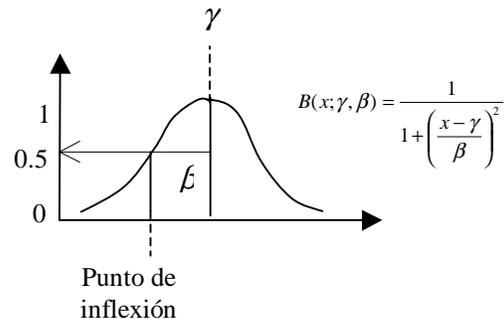


Figura 3 Gráfico de la curva campana Beta.[COX98]

$$\text{Symb}(x) = \begin{cases} P[\text{term1} / \text{Consequent}] & \text{if } x = \text{term1} \\ P[\text{term2} / \text{Consequent}] & \text{if } x = \text{term2} \\ P[\text{term3} / \text{Consequent}] & \text{if } x = \text{term3} \end{cases}$$

Figura 4: Función de membresía simbólica.

Una vez discretizados los atributos, se obtiene por cada uno de ellos una variable lingüística cuyos posibles valores son términos lingüísticos. Se obtendrán tantos términos lingüísticos como intervalos especifique el usuario en el fichero de entrada.

Por cada uno de los términos lingüísticos se genera una función de membresía campana, triangular o simbólica según las características del atributo al que referencia dicho término. Usando Algoritmos Genéticos [PIÑ00] y técnicas del análisis matemático se determinan los parámetros de las funciones de membresía a partir del análisis de los intervalos asociados a cada término lingüístico y teniendo en cuenta el porcentaje de solapamiento entre las funciones de membresía de una misma variable lingüística, especificado previamente.

En el caso de las funciones de membresía simbólicas el valor de membresía de determinado término lingüístico se calcula como la probabilidad condicional de aparición de dicho término dado el consecuente de una regla (Fig 4).

### *3.3- Extensión del algoritmo ID3 para la generación de reglas borrosas.*

Ante la necesidad de obtener reglas borrosas en lugar de reglas duras procedimos a implementar una extensión del ID3, que a diferencia del ID3 clásico genera reglas borrosas.

Como se muestra en la fig 1, la extensión de este algoritmo consiste en procesar las reglas duras obtenidas con la aplicación al conjunto de entrenamiento del ID3 original y utilizando las funciones de membresía calculadas a partir de los datos discretizados convertirlas en reglas borrosas.

La implementación del ID3 extendido tiene una complejidad temporal, en el peor de los casos, de un  $O(s^2)$  donde  $s$  representa el número de casos del conjunto de entrenamiento.

### *3.4- Algoritmos GENRUL5 y MLRUL.*

Analizando las ventajas y deficiencias de los algoritmos consultados en la bibliografía desarrollamos los algoritmos GENRUL5 y MLRUL que permiten la generación de reglas borrosas a partir de datos numéricos y simbólicos. [ARC01]

En el primero de estos algoritmos se recorre el conjunto de entrenamiento generando las reglas necesarias hasta alcanzar un cubrimiento total del conjunto de entrenamiento. Luego se recorren las reglas previamente generadas y se actualiza la

información correspondiente a las funciones de membresía simbólicas a partir de información extraída durante el recorrido del conjunto de entrenamiento. La complejidad computacional temporal en el peor de los casos es un  $O(s^2)$  donde  $s$  es el cardinal del conjunto de entrenamiento. La complejidad espacial es inferior a la de los restantes algoritmos consultados en la bibliografía.

En el segundo algoritmo se crean particiones por ordenamientos recursivos del conjunto de entrenamiento tomando como criterio de comparación, en cada momento, una variable lingüística y sus respectivos términos lingüísticos.

La variable lingüística con mayor relevancia en la clasificación de una partición determinada, se selecciona como criterio de comparación para la creación de subparticiones durante el reordenamiento de los casos. La relevancia se determina por medio de una expresión matemática que propicia la homogeneidad entre casos de una misma clase y la heterogeneidad entre estratos correspondientes a clases diferentes. La complejidad computacional temporal en el peor de los casos es un  $O(s^2)$  donde  $s$  es el cardinal del conjunto de entrenamiento y la complejidad espacial es la menor de todos los algoritmos analizados previamente en este artículo. El algoritmo MLRUL posibilita además la manipulación de rasgos irrelevantes, la ausencia de información y manejo de datos con información ruidosa.

### *3.5- Aspectos generales del uso del sistema.*

El sistema RuleGeneratorML toma de un fichero texto el conjunto de entrenamiento base para la generación de las reglas. En el mismo fichero se especifican además otros aspectos tales como la naturaleza, el porcentaje de solapamiento y la cantidad de intervalos en que se debe discretizar cada atributo; información necesaria en la construcción de las funciones de membresía. (Fig 5)

Las reglas borrosas generadas por el sistema se le presentan al usuario como salida del sistema de dos formas. Una de ellas es un fichero texto (.txt) que posibilita al usuario un fácil entendimiento acerca de las reglas generadas; y la segunda opción es un fichero estructurado (.rul) que posibilita un fácil acoplamiento de RuleGeneratorML con sistemas afines.

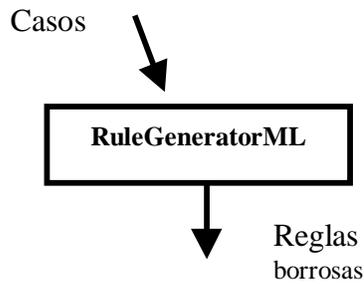


Figura 5 Visión general del sistema como una caja negra.

RuleGeneratorML podría funcionar como un módulo de un sistema neuro-borroso que reajuste los parámetros de las funciones de membresía de las reglas borrosas obtenidas, mejorando la calidad de estas para su uso posterior en un clasificador automático.

#### 4- Conclusiones

Se analizaron diferentes estrategias y algoritmos de aprendizaje automatizado, tales como, los algoritmos ID3 y C4.5, el algoritmo presentado en [HON96], y los presentados en [NAU00]; analizando ventajas y deficiencias de cada uno respecto a la complejidad, tipos de datos que manipulan y reglas que generan.

Se crearon e implementaron los algoritmos GENRUL5 y MLRUL que permiten generar reglas borrosas a partir de datos numéricos y simbólicos. Estos algoritmos permiten el tratamiento de casos con información incompleta.

Se implementó el sistema RuleGeneratorML que genera reglas borrosas por medio del uso de los algoritmos ID3 extendido, el algoritmo reportado en la tesis doctoral de Detlef Nauck y los nuevos algoritmos GENRUL5 y MLRUL.

Se desarrollaron e implementaron métodos que permitieron la construcción de las funciones de membresía a partir de los intervalos discretos y el porcentaje de solapamiento entre funciones de membresía asociadas a una misma variable lingüística. El sistema genera reglas borrosas cuyas funciones de membresía pueden ser simbólicas, triangulares o campanas beta.

El sistema RuleGeneratorML podría funcionar como un módulo de un sistema para la clasificación a partir de las reglas obtenidas, ejemplo de un sistema Fuzzy o de una Red Adaptativa.

#### Referencias Bibliográficas

ARC01 Arco García, Leticia. Mota Caballero Yaile. "Algoritmos para la generación de reglas borrosas".

- Trabajo de Diploma. Universidad Central "Marta Abreu" de las Villas. Cuba 2001
- BER99 Berthold, M. Huber, K. Constructing Fuzzy Graphs from Examples. Int. J. Intelligent Data Analysis. 1999. <http://www.elsevier.com/locate/ida>
- BOY99 Boyen, X. Wehenkel, L. Automatic Induction of Fuzzy Decision Trees and its Application to Power System Security Assessment. Fuzzy Sets and Systems 102. pp 3-19. 1999.
- CHE95 Chen, Joseph E. Otto, Kevin N. Constructing membership functions using interpolation and measurement theory. Fuzzy Sets and systems 73. Pp. 313-327. 1995.
- COX98 Cox Eral, Taber RodMan, O'Hagan Michael, O'Hagen Michael. The Fuzzy Systems Handbook, Second Edition AP Professional, Paperback, 2nd Bk&Cd edition, Published August 1998
- FAB00 Fabri, José Augusto. Vilardi Rissoli, Vador Roberto. Desenvolvimento de um sistema especialista fuzzy aplicado a domínios genéricos do conhecimento. 2000. <http://www.dc.ufscar.br/~fabri>
- FAB99 Fabri, José Augusto. Rissoli, Vador Roberto Vilardi. Camargo, Heloisa de Arruda. Geração Automática de Regras à partir da Arquitetura Neuro-Fuzzy para Classificação de Dados (NEFClass). Anais do 4º Simpósio Brasileiro de Automação Inteligente. USP – São Paulo. 1999.
- GEN97 Genther, Harald. Glesner, Manfred. Advanced data preprocessing using fuzzy clustering techniques. Fuzzy Sets and Systems 85. pp 155 – 164. 1997.
- HON98 Hong, Tzung-Pei. Lee, Chai-Ying. Learning Fuzzy Knowledge from Training Examples. CIKM. pp 161-166. 1998. <http://dblp.uni-trier.de/db/indices/a-tree/h/hong@Tzung=Pei.html>
- KLA95 Klawonn, F. Kruse, R. Derivation of Fuzzy Classification Rules from Multidimensional Data. In: G.E. Lasker, X. Liu (eds.): Advances in Intelligent Data Analysis. The International Institute for Advanced Studies in Systems

- Research and Cybernetics, Windsor, Ontario, pp 90-94. 1995.
- LIU97 Liu, Huan. Setiono, Rudy. Feature Selection via discretization of numeric attributes. IEEE Transactions on Knowledge and Data Engineering, Vol. 9, No. 4. pp 642-645. July/August, 1997. <http://www.comp.nus.edu.sg/~rudys/publications.html>
- NAU00 Nauck, Detlef. Data Analysis with Neuro – Fuzzy Methods. Habilitationsschrift zur Erlangung des akademischen Grades doctor rerum naturalium habitatus (Dr.rer.nat.habil.), Magdeburg, February, 2000. <http://fuzzy.cs.uni-magdeburg.de/papers.html>
- NAU95 Nauck, Detlef. Kruse, Rudolf. NEFCLASS – A Neuro-Fuzzy Approach for the Classification of Data. In Proceedings of the 1995 ACM Symposium on Applied Computing. pp 461-465. Nashville, 1995. <http://fuzzy.cs.uni-magdeburg.de/papers.html>
- PIÑ00 Piñero Pérez, Pedro. "GACom: Biblioteca de componentes para el trabajo con Algoritmos Genéticos". Trabajo de Diploma. Universidad Central "Marta Abreu" de las Villas. Cuba 2000.
- QUI86 Quinlan, J. R. Induction of Decision Trees. Machine Learning. pp 81 - 106. 1986.
- QUI96 Quinlan, J. R. Improved Use of Continuous Attributes in C4.5. Journal of Artificial Intelligence Research 4. pp 77 – 90. 1996.
- SET00 Setiono, Rudy. Kheng Leow, Wee. FERNN: An Algorithm for Fast Extraction of Rules from Neural Networks. Applied Intelligence. Vol. 12, No. ½. pp 15-25. 2000. <http://www.comp.nus.edu.sg/~rudys/publications.html>
- TOM97 Tom, Mitchell. Hill, McGraw. Machine Learning. ISBN: 0-07- 042807-7. 1997. <http://www.cs.cmu.edu/~tom/mlbook.html>
- XIZ98 Xizhao, Wang. Hong, Jiarong. On the handling of fuzziness for continuous valued attributes in decision tree generation. Fuzzy Sets and Systems 99. pp 283-290. 1998.