Aplicación del Modelo de Conjuntos Aproximados de Precisión Variable para Estimar la Probabilidad que tiene cada Elemento de ser Excepcional

Alberto Fernández Oliva[1], Miguel Abreu Ortega[1], Carlos Alberto Iglesias Alvarez[1], Armando Rodríguez Fonte[1], Covadonga Fernández Baizán[2] y Francisco Maciá Pérez[3]

[1] Departamento de Ciencia de la Computación, Facultad de Matemática y Computación, Universidad de la Habana, Cuba
afdez@matcom.uh.cu, miguel87@lab.matcom.uh.cu, calberto@lab.matcom.uh.cu, mandy271088@yahoo.es

[2] Departamento de Lenguajes y Sistemas Informáticos e Ingeniería del Software, Facultad de Informática de la Universidad Politécnica de Madrid (U.P.M), España
cfbaizan@fi.upm.es

[3] Departamento de Tecnología Informática y Computación, Universidad de Alicante, España
pmacia@dtic.ua.es

Resumen. En un proceso de Data Mining, la detección de outliers intenta aprovechar la elevada marginalidad de estos objetos para detectarlos midiendo su grado de desviación respecto a los patrones de comportamiento representativos y desentrañar así conocimiento relevante. Si bien la aplicación de la Teoría de Conjuntos Aproximados (Rough Sets-RS) al campo de los procesos de búsqueda de información en grandes volúmenes de datos (KDD) viene realizándose desde su formulación en la década de los 80, en los últimos años se ha comenzado a considerar la detección de outliers como un proceso de KDD con interés en sí mismo. La combinación de ambos enfoques, Rough Sets como fundamento para la caracterización y detección de outliers, es un punto de vista absolutamente nuevo, con un gran potencial de interés teórico y aplicabilidad práctica. En el presente artículo se presenta un marco teórico basado en el Modelo de Conjuntos Aproximados de Precisión Variable que permite establecer una aproximación estocástica a la solución del problema de determinar si un elemento dado es outlier dentro de un determinado universo de datos.

1 Introducción

La detección de *casos excepcionales* (*outliers detection*) es un campo de creciente relevancia dentro del más general de *Data Mining*. Dentro del mismo, estos objetos, pueden reportar hallazgos de conocimiento de suma importancia en una variada gama

de aplicaciones: detección de fraudes, detección de accesos ilegales a redes corporativas, detección de errores en datos de entrada, etc.

El modelo básico de conjuntos aproximados *(Rough Sets Basic Model —RSBM)* propuesto por el profesor Z. Pawlak [1] en 1982 es un modelo con una base matemática simple y sólida: la teoría de relaciones de equivalencia que permite describir particiones constituidas por clases de objetos indiscernibles. La idea del *Rough Set* consiste en aproximar un conjunto a partir de un par de conjuntos, llamados aproximación inferior y aproximación superior del mismo. En general su enfoque se basa en la habilidad para clasificar datos que han sido obtenidos por diversas vías. En los últimos años este modelo se ha aplicado exitosamente en diversos contextos [2], [3], [4], por lo que el estudio del mismo ha acaparado la atención de los académicos e investigadores a nivel internacional, especialmente, en la solución de problemas donde se desean establecer relaciones entre los datos.

En [5] se propone un método de detección de *outliers* que constituye la primera aplicación de *RSBM* a dicho problema, pero una implementación computacional del mismo deviene en un problema intratable por ser de orden exponencial. En [6] se presenta una extensión del marco teórico de la propuesta anterior a partir del cual se implementa un algoritmo de detección de *outliers* basado en *RSBM —Algoritmo RSBM—* con orden de complejidad temporal y espacial no exponencial. No obstante, este resultado hereda el carácter determinista de dicho modelo en lo que respecta a la clasificación.

Una generalización de *RSBM* es la propuesta por W.Ziarko [7], el modelo de conjuntos aproximados de precisión variable (*Variable Precision Rough Sets Model VPRSM*). Este modelo subsana el carácter determinista de *RSBM* a partir de una nueva concepción de la inclusión de conjuntos, la *inclusión de conjuntos mayoritaria* [5], [6], donde se permite manejar unos umbrales definidos por el usuario. En [10] se presenta un algoritmo *—Algoritmo VPRSM—* detección no determinista de *outliers* basado en *VPRSM* computacionalmente viable.

Los algoritmos *RSBM* y *VPRSM* dan solución al siguiente problema:
«A partir de un *umbral de excepcionalidad establecido* (μ) y un determinado *error de clasificación permitido* (β), extraer un conjunto de *outliers* a partir de un *universo* de datos dado.»

En este artículo, basándonos en los resultados antes mencionados se propone un nuevo enfoque del problema de la detección de *outliers*, que resuelve las limitaciones de los anteriores. Por lo tanto, el nuevo enfoque estará en correspondencia con el siguiente objetivo:
«Establecer un método, computacionalmente viable, que proporcione la probabilidad que tiene cada elemento de un *universo de datos* dado de ser *excepcional*, sin necesidad de haber establecido las condiciones previas —referidas a la determinación de los umbrales que intervienen en el análisis— en función de un contexto específico de aplicación.»

Dicho objetivo se establece sobre la base de la siguiente hipótesis:
«Es posible desarrollar una nueva teoría basada en la extensión de los conceptos básicos y las herramientas formales que nos proporciona la *Teoría de Conjuntos Aproximados* [1], [11] y el *Modelo de Conjuntos Aproximados de Precisión Variable* (*VPRSM*) [7], aplicados al problema de la detección de *outliers*, que permita obtener, de forma no supervisada, para cada elemento de un *universo de datos*, la región de

valores de los umbrales en la cual dicho elemento es *outlier*. A partir de dicho resultado, es posible determinar la probabilidad de cada elemento del *universo* de ser *outlier* con relación al mismo.»

Para desarrollar el método expresado en el objetivo planteado se propone una ampliación del marco teórico desarrollado en [6], [10], a partir de los elementos conceptuales de la teoría de *RS* así como de *VPRSM*, junto a la propuesta teórica de [5]. Todos ellos permiten demostrar, formalmente, los elementos teóricos propuestos en la nueva concepción del método y sirven de marco de referencia para el diseño e implementación de un algoritmo, computacionalmente viable, con el que se valida la hipótesis de partida.

En función de lo expuesto, el resto del artículo se estructura de la siguiente forma: en el apartado 2 se propone un marco teórico junto con un algoritmo y su implementación computacional, para determinar la región de excepcionalidad para cada elemento del *universo*; en el apartado 3 se proponen nuevos elementos teóricos y se aplican técnicas estadísticas que permiten determinar la probabilidad de cada elemento del *universo* de ser *outlier* dentro del mismo; en el apartado cuatro se presentan algunas pruebas que permitieron validar los resultados alcanzados mientras que en el apartado 5 se presentan las conclusiones de la investigación y se exponen algunas *líneas abiertas* y trabajos futuros que permiten dar continuidad a la investigación.

2 Determinación para cada elemento del universo de su región de excepcionalidad. Algoritmo BM

El método propuesto se fundamenta en determinar la región de excepcionalidad para cada elemento del universo en función del valor de los umbrales que intervienen en el análisis y estimar su probabilidad de ser outlier en dicho universo.

Formalmente, el problema a resolver en esta fase puede enunciarse de la siguiente forma:

$\forall x: x \in U$, determinar R. Donde R es la región que establece el conjunto de valores de los umbrales β y μ para los cuales x es outlier en U.

- Para solucionar el mismo, se amplía el marco teórico existente, con nuevas definiciones y proposiciones, lo cual permite establecer un algoritmo, computacionalmente eficiente, que resuelve el problema **y** al cual hemos denominado algoritmo BM (*Beta-Miu*), atendiendo al nombre por el que se conocen los umbrales que intervienen en el análisis: *grado de desclasificación-*β y *umbral de excepcionalidad -*μ.

2.1 Marco Teórico – Algoritmo BM

- El análisis se divide en dos partes:

1. Determinación de la *región de excepcionalidad* con respecto al umbral β.
2. Determinación de la *región de excepcionalidad* con respecto al umbral μ.

— Finalmente, se integran estas dos soluciones particulares para determinar *la región de excepcionalidad* (β, μ) para cada elemento del ***universo*.**

2.2 Región de Excepcionalidad con respecto a β

En este apartado se determina la *región de excepcionalidad* con respecto al conjunto de valores de β (referido a un *β-error admisible en la clasificación*). Para ello se resuelven tres subproblemas particulares:

- **Subproblema No. 1:** Determinar el rango de valores de β para los cuales $B_i \subseteq B_j$, $i \neq j$, $1 \leq i, j \leq m$

De acuerdo con el marco teórico expuesto en [6] se sabe que si ninguna *frontera interna* B_i es subconjunto de otra *frontera interna* B_j, entonces todos los elementos de B_j son candidatos a ser *outliers* en el *conjunto de datos*, o *universo*, U. En función de esto el problema a resolver se replantea de la siguiente forma:

Determinar el conjunto de valores de β para los cuales una *frontera interna* B_i, $i \neq j$, es subconjunto de la *frontera interna* B_j, o sea, $B_i \subseteq B_j$. Una vez calculados éstos, $\forall i \neq j$, $1 \leq i \leq m$, entonces el complemento de la unión de todos los intervalos de valores de β calculados, será el conjunto de valores, con respecto a dicho parámetro, para los cuales todos los elementos de B_j son candidatos a ser *outliers*.

- **Subproblema No. 2:** Determinar el rango de valores de β para los cuales una frontera interna dada es nula.

Igualmente, en el marco teórico en el cual se basa el método de detección se supone que las *fronteras internas* tenidas en cuenta en el análisis no son nulas. Atendiendo a esto se determinan los valores de β para los cuales esto se cumple, partiendo de una *frontera interna* cualquiera B_i y posteriormente este resultado se generaliza para cualquier otra *frontera interna* mediante un análisis similar.

- **Subproblema No. 3:** Determinar el conjunto de valores de β para los cuales $B_i = B_j$, $i \neq j$, $1 \leq i, j \leq m$.

En el marco teórico en el cual se basa el método de detección tampoco se contempla la existencia de dos *fronteras internas* iguales, por tanto, es necesario determinar el conjunto de valores de β para los cuales esto ocurre.

En esta ocasión, el problema consiste en determinar el conjunto de valores de β para los cuales $B_i = B_j$, $i \neq j$, $1 \leq i, j \leq m$ y esto se deduce fácilmente a través de la siguiente secuencia de equivalencias:

$$B_i = B_j \Leftrightarrow B_i \subseteq B_j \wedge B_j \subseteq B_i \Leftrightarrow \beta \in I_{ij} \wedge \beta \in I_{ji} \Leftrightarrow \beta \in I_{ij} \cap I_{ji}$$

A partir de lo cual podemos resumir que el conjunto de valores de β para los cuales $B_i = B_j$, $i \neq j$, $1 \leq i, j \leq m$, es el siguiente:

$$EQ_{ij} = \{\beta : \beta \in I_{ij} \cap I_{ji}\}$$

donde, I_{ij}: conjunto de valores de β para los que $B_i \subseteq B_j$. $i \neq j$, $1 \leq i, j \leq m$

Una vez concluido el análisis de los tres subproblemas planteados se puede establecer un criterio general en relación a cuándo una *frontera interna* es subconjunto de otra. Estableciendo la siguiente secuencia de conjuntos podemos llegar a conclusiones:

- A: Conjunto de valores de β para los cuales existe una fron*tera interna* no vacía que es *subconjunto propio* de la *frontera interna j*.

$(I_{1j} - EQ_{1j} - N_1) \cup (I_{2j} - EQ_{2j} - N_2) \cup ... \cup (I_{mj} - EQ_{mj} - N_m) = A$,
donde, N_i: conjunto de valores de β para los cuales $B_i = \phi$, $1 \leq i \leq m$

- A^c: Conjunto de valores de β para los cuales ninguna *frontera interna* no vacía es *subconjunto propio* de la *frontera interna j*.
- $S_j = A^c - N_j$

S_j**:** Conjunto de valores de β para los cuales ninguna *frontera interna* no vacía es *subconjunto propio* de la *frontera interna j* excluidos los valores para los cuales dicha *frontera* es vacía.

Sabiendo que, para que todos los elementos de B_j sean *outliers* debe cumplirse que no exista otra *frontera interna* que sea subconjunto de ella, a partir de los resultados anteriores se puede plantear que esto sucede sólo cuando $\beta \in S_j$.

S_j representa el intervalo de valores de β para el cual, un elemento x del *universo*, $x \in B_j$, pertenece a algún *conjunto excepcional no redundante*, por lo cual x es un posible *outlier*.

— A continuación se hace un análisis similar para determinar el conjunto de valores del *umbral de excepcionalidad* μ para el cual cada elemento del *universo* podrá ser considerado un *outlier***.**

2.3 Región de Excepcionalidad con respecto a μ

— Nuestro problema ahora es el siguiente:

Dado un elemento $a \in U$, determinar el rango de valores del umbral μ, para los cuales el *grado de excepcionalidad* de a es mayor que μ.

Los elementos teóricos que resultan necesarios para garantizar la solución de este problema se plantean siguiendo la secuencia lógica que se expresa a continuación:

- Definir el conjunto de valores de β para los cuales $\forall a: a \in U$ pertenece a la *frontera interna* B_i, $1 \leq i \leq m$.
- Establecer una nueva definición de *grado de excepcionalidad* $\forall a: a \in U$, bajo una nueva interpretación en la cual intervienen los valores de β
- Determinar $\forall a \in U$ el rango de valores de μ *para los* cuales $GrExcep(a,\beta) \geq \mu$ para un valor β dado.

Siguiendo la secuencia expresada, se define, primeramente, el conjunto de valores de β para los cuales $a \in U$ pertenece a la *frontera interna* B_i, $1 \leq i \leq m$

Definición 1: Sea U un *universo* de datos dado y sea X el conjunto de valores de U que cumplen un *concepto*. $\forall a \in U$, $1 \leq i \leq m$, W es una *clase de equivalencia* de la partición inducida por la *relación de equivalencia* r_i en U, tal que, $a \in W$. El conjunto de valores de β para los cuales a pertenece a la frontera interna B_i se define como:

$$M_i(a) = \begin{cases} \{\beta : \beta < c(W, X) < 1 - \beta\} & si \quad a \in X \\ \phi & si \quad a \notin X \end{cases}$$

Según lo establecido por $M_i(a)$, las restricciones que deben cumplir los valores del parámetro β para garantizar que a pertenezca a la *frontera interna* B_i, son las siguientes: $\beta < c(W, X) < 1-\beta \Rightarrow [\beta < 1-c(W, X)] \wedge [\beta < c(W, X)]$.

En función de esto, a partir de $M_i(a)$ se puede establecer el siguiente intervalo de valores de β dentro del cual se garantiza $a \in B_i$: $\forall \beta$: $\beta \in [0, min(-c(W, X), 1-c(W, X))$.

Este resultado garantiza un criterio necesario para afirmar que un elemento $a \in U$ puede ser candidato a *outlier*. En este caso, se trata de su pertenencia a alguna *frontera interna.* Teniendo en cuenta el mismo, se establece a continuación una nueva definición de *grado de excepcionalidad* de un elemento $a \in U$, bajo una nueva interpretación: dependencia de la misma con relación a los valores de β.

Como paso previo se hace necesario dar una nueva definición y establecer una proposición a partir de la misma.

Definición 2: $\forall a \in U$, $1 \leqslant i \leqslant m$

$$\lambda_i(a) = \begin{cases} Sup(M_i(a) & si \quad M_i(a) \neq \phi \\ 0 & otro \quad caso \end{cases}$$

$Sup(M_i(a))$: menor valor de β que es mayor que todos los valores del intervalo $M_i(a)$. Para todo $\beta < \lambda_i(a)$ el elemento a pertenece a la *frontera interna* B_i. Por tanto:

Proposición 3: $\forall a \in U$, $1 \leqslant i \neq j \leqslant m$, si $\lambda_i(a) \leqslant \lambda_j(a) \Rightarrow \forall \beta$: $\beta < \lambda_i(a)$, $a \in B_i \wedge a \in B_j$

A partir del análisis hecho, se puede obtener para cada elemento $a \in U$ una ordenación particular de los supremos $\lambda_i(a)$, $1 \leqslant i \leqslant n$ asociados a cada una de las *fronteras internas* B_i, $Z_i(a)$. Sea $Z_1(a),\ldots, Z_m(a)$, tal que $\lambda_{Z1(a)}(a) \leqslant \cdots \leqslant \lambda_{Zm(a)}(a)$ una permutación de índices que ordena a los $\lambda_i(a)$.

Definición 4: Sean $a \in U$, $\beta \in [0; 0,5)$ y sea m la cantidad de *fronteras internas* tenidas en cuenta en el análisis. Se define el *Total* de *fronteras internas* a las que pertenece el elemento a para el valor β dado, de la siguiente forma:

$$Total(a, \beta) = \begin{cases} m & si \quad \beta < \lambda_{Z1(a)}(a) \\ 0 & si \quad \beta \geq \lambda_{Zm(a)}(a) \\ m - máx_k(\beta \geq \lambda_{Zk(a)}(a)) & en \quad otro\_caso \end{cases}$$

En función de las **definiciones 2 *y* 4** así como de la **proposición 3** se redefine el concepto de *grado de excepcionalidad* de un elemento $a \in U$ en función de los valores de β.

Definición 5: Sean $a \in U$, un valor $\beta \in [0; 0,5)$ y sea m la cantidad de *fronteras internas* tenidas en cuenta en el análisis. Se define el *grado de excepcionalidad* del elemento a para el valor β dado, de la siguiente forma:

$GrExcep(a, \beta) = Total(a, \beta)/m$

Esta definición no entra en contradicción con la propuesta en [6]. A partir de ella, $\forall a \in U$ se puede obtener el *grado de excepcionalidad* de dicho elemento para cualquier valor de β y por consiguiente, los valores de μ para los cuales $GrExcep(a,\beta) \geq \mu$.

2.4 Integrando resultados

El marco teórico alcanzado permite establecer el siguiente procedimiento (método) general para determinar los valores de β y μ para los cuales el elemento $a \in U$ es *outlier* en U:

1. Determinar $M_i(a)$: Valores de β para los cuales el elemento $a \in B_i$.
2. Determinar S_i: Valores de β para los cuales no existe una *frontera interna* que sea subconjunto de la *frontera interna* B_i.
3. Determinar $D_i(a) = M_i(a) \cap S_i$: Valores de β para los cuales el elemento a pertenece a B_i y no existe una *frontera interna* que sea subconjunto de la *frontera interna* B_i.

Para los valores de $\beta \in D_i(a)$, el elemento a pertenece a algún *conjunto excepcional no redundante* y es el único representante de la *frontera interna* B_i en dicho conjunto, o sea, para los valores de β en $D_i(a)$, $a \in Ei$.

4. $\forall \beta_o, \mu_o$: $\beta_o \in \bigcup_{k=1}^{m} D_k(a) \wedge \mu_o \leq GrExcep(a, \beta_o)$, entonces: *a es un outlier en U*

Un $\beta_o \in \bigcup_{k=1}^{m} D_k(a)$ representa un valor para el cual el elemento a pertenece a alguna *frontera interna* de la cual ninguna otra *frontera interna* es subconjunto y, en tal caso, se restringe μ_o a ser menor o igual que $GrExcep(a, \beta_o)$.

La **Fig. 1** ilustra la región de valores β -μ en la cual un elemento a cualquiera del *universo* es *outlier* en U. En este caso se supuso que:

intervalo (1) $\cup$ *intervalo (2)* = $\bigcup_{k=1}^{m} D_k(a)$

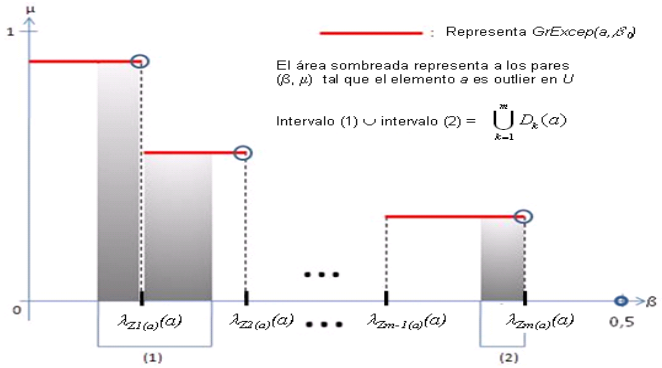

Fig. 1. Región de valores β-μ para los cuales un elemento a cualquiera del *universo* es *outlier* en U

– La viabilidad computacional del procedimiento expuesto se valida a partir de un algoritmo que determina, de forma no supervisada, la región de valores de los umbrales β-μ en la cual cada elemento del *universo* es outlier. A continuación se dan detalles sobre la implementación computacional de dicho algoritmo.

2.5 Implementación computacional

En la concepción general del algoritmo se destacan dos tareas fundamentales:

- **Tarea No. 1:** Cálculo de las dependencias entre las *fronteras internas* (relación de inclusión entre ellas).
- **Tarea No. 2:** Búsqueda de la región β-μ para la cual cada $x \in U$ es *outlier.*

Las estructuras de datos que intervienen en el algoritmo, desglosadas por tareas, son las siguientes:

- **Tarea No. 1**

case1[i][ec]: Dada la *clase de equivalencia ec* inducida por la *relación de equivalencia i,* la estructura almacena la solución para uno de los casos que interviene en el cálculo de I_{ij}.

case2[i][j][ec]: Dada la *clase de equivalencia ec* inducida por la *relación de equivalencia i* y ante el problema de determinar si *ec* es subconjunto de la *frontera interna* asociada a la *relación de equivalencia j*, la estructura almacena la solución para otro caso que interviene en el cálculo de I_{ij}.

Subset[i][j]: Dadas las *fronteras internas i* e *j,* la estructura almacena los valores de β para los cuales la *i* es subconjunto de la *j,* $1 \leq i, j \leq m, i \neq j$.

Equal[i][j]: Dadas las *fronteras internas i* e *j,* la estructura almacena los valores de β para los cuales la *frontera interna i* es igual a la *frontera interna j,* $1 \leq i, j \leq m, i \neq j$.

Null[i]: Dada la *frontera interna i*, la estructura almacena los valores de β para los cuales la *frontera interna i* es nula, donde: $1 \leq i \leq m$.

S[i]: Dada la *frontera interna i*, la estructura almacena los valores de β que conforman el conjunto S_i, donde: $1 \leq i \leq m$.

- **Tarea No. 2**

Lambda[e][i]: Dado $e \in U$ y la *frontera interna i*, la estructura almacena el valor λ_i asociado al elemento *e*, donde: $1 \leq i \leq m$.

GrExcep[e]: Dado $e \in U$, la estructura almacena la función $F(\beta)=GrExcep(e, \beta)$. Nótese que $F(\beta)$ es una proyección de la función $GrExcep(e, \beta)$ sobre el eje β.

Pertain[e][i]: Dados $e \in U$ y la *frontera interna i*, la estructura almacena los valores que conforman el conjunto $M_i(e)$, o sea, los valores de β para los cuales el elemento *e* pertenece a la *frontera interna i,* donde: $1 \leq i \leq m$.

***Outlier*[e]:** Dado $e \in U$, la estructura almacena los valores de β para los cuales el elemento *e* pertenece a algún *conjunto excepcional no redundante:* $D_i(e)$.

Result[e]: Dado $e \in U$, almacena la región *beta-miu* tal que, *e* es un *outlier* en *U*.

Por cada una de estas estructuras de datos se implementó un método que calcula los valores correspondientes a cada una de ellas. La complejidad espacial de cada estructura, afecta la complejidad temporal del método que la conforma.

La **Tabla 1** refleja la complejidad espacial de las estructuras de datos. De igual forma, muestra la complejidad temporal de los métodos que calculan los valores de las mismas y la complejidad temporal general del algoritmo *BM*, para el caso peor.

2.6 Conclusiones

El aspecto más original del algoritmo *BM* es que permite, de forma no supervisada; establecer las región de valores de los umbrales (parámetros β y μ) bajo la cuales cada elemento del universo será considerado *outlier*. No obstante, la complejidad temporal y espacial del algoritmo, es de un orden mayor que la de los algoritmos *RSBM* y *VPRSM*. Esto se debe a que el resultado que ofrece el algoritmo *BM* es más general.

Al ejecutar una vez el algoritmo *BM* para un univ*erso* de datos dado; se pueden obtener las salidas particulares de los algoritmos previos para cualquier valor de (β, μ). Al determinar para cada elemento del *universo* la región total de valores de dichos umbrales bajo los cuales tal elemento es *outlier*, se garantiza que posteriormente se pueda hacer un recorrido por todo el *universo* buscando si pares particulares de valores de los umbrales (β, μ) pertenecen a la *región de excepcionalidad* de cualquier elemento del mismo. Por tanto, el uso de este algoritmo es recomendable cuando se necesite obtener un resultado acerca de la condición de *outlier* de los elementos del *universo* para un conjunto de valores de los umbrales.

Tabla 13. Complejidad espacial de las estructuras de datos y complejidad temporal de los métodos que calculan sus valores.

Estructura de datos	**Complejidad espacial (*caso peor*)**	**Complejidad temporal del método que calcula el valor de la estructura de datos (*caso peor*)**
Case1[i][ec]	$O(n \times m)$	$O(n \times m \times c)$
Case2[i][j][ec]	$O(n \times m^2)$	$O(n \times m^2 \times c)$
Subset[i][j]	$O(n \times m^2)$	$O(n \times m \times log(n))$
Equal[i][j]	$O(n \times m^2)$	$O(n \times m^2)$
Null[i]	$O(m)$	$O(n \times m)$
S[i]	$O(n \times m^2)$	$O(n \times m^2 \times log(m))$
Lambda[e][i]	$O(m^2)$	$O(n \times m \times c)$
GrExcep[e]	$O(n \times m)$	$O(n \times m \times log(m))$
Pertain[e][i]	$O(n \times m)$	$O(n \times m)$
***Outlier*[e]**	$O(n^2 \times m^2)$	$O(n^2 \times m^2 \times log(m))$
Result[e]	$O(n^2 \times m^2)$	$O(n^2 \times m^2)$
Complejidad Espacial TOTAL del Algoritmo	*Complejidad espacial total del algoritmo:* $O(n^2 \times m^2)$	*Complejidad temporal total del algoritmo:* $O(n^2 \times m^2 \times log(m))$

El resultado que se obtiene tras la ejecución del algoritmo *BM* contiene cualquier resultado particular que pudiese obtenerse a partir de la ejecución de los algoritmos

anteriores. Esto constituye la principal ventaja de este algoritmo frente a la desventaja que supone el aumento en la *complejidad temporal* y *espacial* del mismo.

A pesar del orden de *complejidad temporal* señalado para el *caso peor*; el algoritmo puede llegar a alcanzar un orden de *complejidad temporal* similar al de los anteriores (*RSBM* y *VPRSM*), casi lineal para el *caso mejor* ($\Omega(n \times m^2 \times c)$).

Las región obtenida durante permite establecer una aproximación estocástica a la solución del problema de determinar si un elemento dado es *outlier* dentro de un determinado *universo* de datos (establecer un criterio probabilístico sobre dicha condición). En función de esto, se diseña un algoritmo que proporciona la probabilidad que tiene cada elemento del *universo* de ser *excepcional.*

3 Estimación de la probabilidad de cada elemento de ser *outlier*

3.1 Marco Teórico. Algoritmo BM/Probabilístico

En el diseño del algoritmo, se utilizan algunos conceptos sencillos de Teoría de Probabilidades, cuyas definiciones pueden encontrarse en cualquier texto básico de esa disciplina. [12] constituye una referencia significativa a tales efectos.

Como ya sabemos, tras la primera fase del proceso se obtiene como resultado, para cada $x \in U$, la región de valores β y μ, en la cual dicho elemento es un *outlier*. Denotemos por **¡Error! Marcador no definido.¡Error! Marcador no definido.¡Error! Marcador no definido.¡Error! Marcador no definido.¡Error! Marcador no definido.¡Error! Marcador no definido.¡Error! Marcador no definido.** R_x a la región encontrada para un elemento dado, $x \in U$.

Considerando β y μ dos *variables aleatorias*, denotemos por $\varphi(\beta, \mu)$ a la *función de densidad de probabilidad* del *vector aleatorio* (β, μ). Entonces, la *función de distribución* de (β, μ) quedaría:

$$P(\beta \leq i, \mu \leq j) = \int_{-\infty}^{i} \int_{-\infty}^{j} \varphi(\beta, \mu) d\beta d\mu \qquad (1)$$

Luego, la probabilidad que estamos interesados en calcular P_x, o sea la probabilidad de que $x \in U$ sea *outlier* conociendo R_x, puede calcularse a partir de (1) de la siguiente forma:

$$P_x = P((\beta, \mu) \in R_x) = \int_{R_x} \varphi(\beta, \mu) d\beta d\mu \qquad (2)$$

teniendo en cuenta que x es *outlier* para los valores de β y μ que pertenecen a R_x.
como β y μ son dos *variables aleatorias* independientes**,** entonces:

$\varphi(\beta, \mu) = f(\beta) g(\mu)$

donde, $f(\beta)$ y $g(\mu)$ son las *funciones de densidad de probabilidad* de β y μ respectivamente.

Por tanto:

$$P_x = \int_{R_x} f(\beta) g(\mu) d\beta d\mu \qquad (3)$$

Para calcular P_x sólo tenemos que sustituir en (3) las *funciones de densidad de probabilidad* de los parámetros β y μ y luego calcular la integral resultante.

En la práctica, lo más común es que no se cuente con ninguna información con respecto a la *distribución* de los parámetros β y μ, por ello asumiremos que ambos *distribuyen uniformemente*. Si en algún contexto esta *distribución* es diferente a la supuesta; sólo es necesario modificar el cálculo de P_x con las nuevas *funciones* utilizando algún *método numérico* para el cálculo de la integral, si fuera necesario. A partir de esta suposición, la integral resultante es muy fácil de calcular.

La *función de densidad de probabilidad* para una variable *t* que *distribuye uniformemente* en el intervalo [a, b], es la siguiente:

$$f(t) = \begin{cases} \dfrac{1}{b-a} & si\ a < t < b \\ 0 & en\ otro\ caso \end{cases} \tag{4}$$

Como, $0 \leqslant \beta < 0{,}5$ y $0 \leqslant \mu \leqslant 1$, entonces bajo *hipótesis de uniformidad* para el valor de estos umbrales, su función de *densidad de probabilidad* sería

$$f(\beta) = \frac{1}{0{,}5 - 0} = 2 \tag{5}$$

$$g(\mu) = \frac{1}{1 - 0} = 1 \tag{6}$$

Sustituyendo estos valores en (3), tenemos:

$$P_x = \int_{R_x} 2 * 1 d\beta d\mu$$

$$P_x = 2 \int_{R_x} d\beta d\mu \tag{7}$$

y como $\int_{R_x} d\beta d\mu$ es el área de la región R_x, entonces:

$$P_x = 2 * Area(R_x) \tag{8}$$

este resultado puede interpretarse como:

$$P_x = \frac{Area(R_x)}{0{,}5} \tag{9}$$

que no es más que el cociente entre el área de la **región favorable** (región de valores (β, μ) para los cuales *x* es *outlier*) y el **área total** (rectángulo que define el dominio de los valores (β, μ) en el plano).

3.2 Implementación computacional. Algoritmo BM/Probabilístico

Los siguientes elementos constituyen las entradas del mismo: El *universo U* (*conjunto de datos*). El *concepto C*. $\Re = \{r_1, r_2, ..., r_m\}$: *relaciones de equivalencia.*

La salida del algoritmo es una estimación de la probabilidad para cada elemento de *U* en cuanto a su condición de *outlier* en dicho *universo*.

Una vez que *BM/*Probabilístico recibe como entrada la región calculada por el algoritmo *BM*, sólo agrega a su funcionalidad un método principal donde se hace el cálculo de la probabilidad en la forma expresada en (9). Una descripción en *pseudo*-código de dicho método, se ilustra a continuación:

```
(1) result = Result[e] // Salida del algoritmo BM
(2) for e in U
(3)         r = ptos (β,μ) de la región Result[e] tal que β<0,5
(4)         p[e] = (Área de result) / 0,5
```

Algoritmo 1. Método principal del algoritmo *BM*/Probabilístico

La *complejidad temporal* del algoritmo *BM/P* se ve afectada por la *complejidad temporal* del proceso de determinación de la región de excepcionalidad, pues, la salida que se obtiene tras la ejecución del algoritmo *BM* es la entrada del algoritmo *BM*/Probabilístico. Por tanto, el análisis se estructura de la siguiente forma:

Coste de **Determinar la región de excepcionalidad**:

Complejidad temporal del algoritmo *BM:* $O(n^2 \times m^2 \times log(m))$ (1)

Coste de **Determinar la probabilidad:**

(cardinalidad del *conjunto de datos*) × (cota para la cantidad de rectángulos que pueden pertenecer a una región *beta-miu*) =

$= (n) \times (n \times m^2) = O(n^2 \times m^2)$ (2)

Complejidad temporal **TOTAL** del algoritmo *BM*/Probabilístico para el *caso peor*:

O(máx (Coste de (1)*,* Coste de (2)*)) =* $O(n^2 \times m^2 \times log(m))$

4 Validación de los resultados

Las pruebas de validación de los resultados tuvieron como objetivos principales comparar los tiempos de ejecución entre los algoritmos *VPRSM* y *BM/Probabilístico* y evaluar la calidad en la detección del algoritmo *BM/Probabilístico.* Para ello se utilizaron conjuntos de datos aleatorios generados automáticamente y conjuntos de datos del mundo real.

4.1 Prueba No. 1

Objetivo de la prueba: Validar *tiempo de ejecución* del algoritmo *BM/Probabilístico* — comparando el mismo con el algoritmo *VPRSM* [10] — al trabajar con *conjuntos de datos* de *gran tamaño* y *alta dimensionalidad.* **Conjunto de datos utilizado:**

Conjunto de datos sintético (conjunto de datos aleatorio generado automáticamente a partir del uso de técnicas estadísticas).**Tipo:** *multivariado.* **Tipo/atributos**: *categóricos* y *continuos.* **No. Filas:** 500.000. **No. columnas:** 100. **Dispositivo de cálculo utilizado:** Procesador: *Intel(R) Core(TM)2 Quad CPU Q6600 @ 2.40Ghz 2.39Ghz* Memoria: *3.25GB.* Sistema Operativo: *Windows 7 Ultimate*

4.1.1 Descripción de la prueba

La **Fig. 2** muestra los resultados obtenidos —en cuanto a *tiempo de ejecución*— tanto por el algoritmo *BM*/Probabilístico como por el algoritmo *VPRSM*.

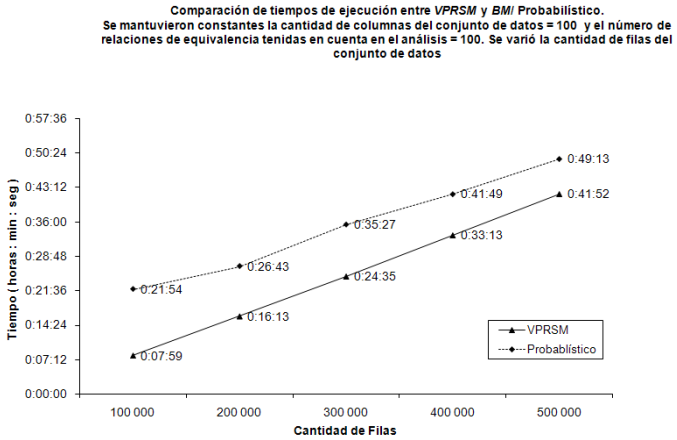

Fig. 2. Comparación de tiempos de ejecución entre los algoritmos *VPRSM* y *BM/Probabilístico*

4.1.2 Interpretación de los resultados

Las curvas muestran que ambos algoritmos se comportan de manera similar —en cuanto a *tiempo de ejecución*— y que son computacionalmente eficientes al ejecutarse sobre un *conjunto de datos* de *gran tamaño* y *alta dimensionalidad*. Se aprecia la linealidad en los *tiempos de ejecución* alcanzados.

El resultado alcanzado pone de manifiesto que, a pesar de que el *orden de complejidad temporal* del algoritmo es *cuadrático,* para el caso peor, este puede llegar a alcanzar un orden de *complejidad temporal casi lineal* al ejecutarse sobre *conjuntos de datos* con las características antes mencionadas.

4.2 Prueba No. 2

Objetivo de la prueba: Validar *calidad de la detección*. **Conjunto de datos utilizado:** *Arrhythmia Data Set (datos de pacientes con problemas cardiovasculares).* **Fuente:** *UCI Machine Learning Data Repository* [13]. **Tipo:** *multivariado*. **Tipo/ atributos:** *reales, enteros* y *categóricos.* **No. filas:** 452. **No. columnas:** 279 **Dispositivo de cálculo utilizado:** INTEL(R) Core(TM) 2 Duo, CPU T5450 @ 1.66 Ghz (2 CPUs), 2046 MB de RAM. **Plataforma:** Windows Vista.

Descripción de la Prueba 2:

concepto: *personas* con *peso* ◘40 *kg* (*weigth* ◘ 40) *–personas de bajo peso–*

Relaciones de equivalencia:

- ***relación de equivalencia-1:*** Se estableció a partir del atributo *heart rate*: cantidad promedio de *latidos por minuto* del corazón de las personas. La relación de equivalencia particiona al *conjunto de datos* en dos clases de equivalencia: [44, 61] y [62, 163]
- ***relación de equivalencia-2:*** Se estableció a partir del atributo *number of intrinsic deflections:* cantidad de *desvíos arteriales* de cada persona. La relación de equivalencia particiona al *conjunto de datos* en dos clases de equivalencia: [0, 59] y [60, 100]
- ***relación de equivalencia-3:*** Se estableció a partir del atributo *height: altura* de una persona expresada en centímetros. La relación de equivalencia particiona al *conjunto de datos* en dos clases de equivalencia: [60, 175] y [176, 190]

Se introdujeron intencionalmente en el *conjunto de datos*, 12 *outliers* con valores contradictorios para *personas de bajo peso*.

Los valores *normales* de los atributos tenidos en consideración en las *relaciones de equivalencia* para las *personas de bajo peso*, son los siguientes: *heart rate*: > 65, *intrinsic deflections:* < 50, *height : < 170 cm.*

La **Tabla 2** refleja los *outliers* introducidos. Los valores en **negrita** e *itálica* representan los valores contradictorios.

En la prueba, los valores de μ tenidos en consideración fueron los siguientes: 0,2; 0,4; 0,6; 0,8 y 1. Para cada uno, se varió β a partir de la siguiente secuencia de valores: 0; 0,1; 0,2 y 0,3. Los valores 0,4 y 0,5 no se mencionan porque a partir de β=0,3 la cantidad de *outliers* detectados se mantuvo en 0.

La **Fig. 3** muestra los resultados alcanzados en esta ocasión. Se consideraron diferentes valores de *k*, se analizaron los *k* elementos del *conjunto de datos* que el algoritmo detectó con mayor probabilidad de ser *outliers* y se determinó, de ellos, cuántos pertenecían al conjunto de *outliers* que fueron introducidos en el *conjunto de datos*. Valores de *k* tenidos en consideración: 5, 10, 15 y 20.

La **Tabla 3** muestra el valor de la probabilidad determinado por el algoritmo para los *outliers* que fueron introducidos en el *conjunto de datos*.

Tabla 2. *Outliers* introducidos. Prueba 2: *Arrhythmia DS*

Id	***weight (kg)***	***heart rate***	***number of intrinsic deflections***	***height (cm)***
1	15	***60***	17	***180***
2	31	93	***68***	***178***
3	39	***50***	***82***	130
4	10	***53***	16	***188***
5	19	***45***	***90***	***190***
6	20	***48***	***86***	***183***
7	25	***50***	***71***	***180***
8	29	***55***	***75***	***179***
9	33	90	***60***	***176***
10	40	***61***	20	***186***
11	26	***50***	***99***	***180***
12	38	92	***100***	***178***

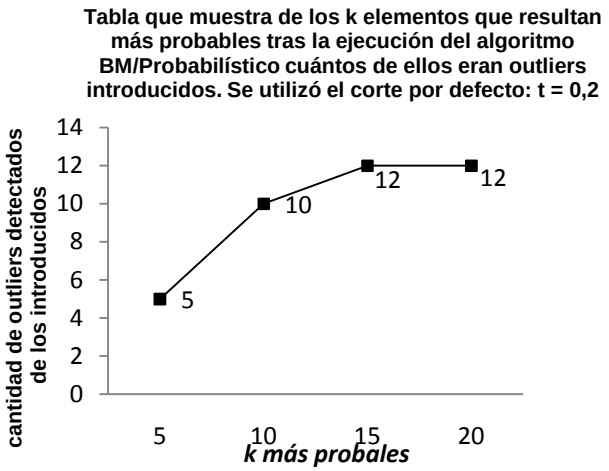

Fig. 3. Resultados de la Prueba 2.1 realizada al algoritmo *BM*/Probabilístico

4.2.1 Interpretación de los resultados

Según la cantidad de elementos *más probables* (*k*) tenidos en consideración en cada caso pudo observarse que cuando *k=5,* los 5 elementos con mayor probabilidad de ser *outliers,* resultaron ser los 5 elementos más contradictorios introducidos en el *conjunto de datos*.; cuando *k=10*, los 10 con mayor probabilidad eran *outliers* introducidos en el *conjunto de datos* y cuando *k=15 y k= 20,* ya aparecían entre los *k más probables* los 12 *outliers* que fueron introducidos intencionalmente.

Tabla 3. Probabilidad para los *outliers* introducidos. Prueba 2.1: *Arrhythmia DS*

Id	*weight (Kg)*	*Heart rate*	*number of intrinsic deflections*	*height (cm)*	*probabilidad de ser outlier*
1	15	***60***	17	***180***	0.61884
2	31	93	***68***	***178***	0.7557252
3	39	***50***	***82***	130	0.6151009
4	10	***53***	16	***188***	0.61884
5	19	***45***	***90***	***190***	**0.8779342**
6	20	***48***	***86***	***183***	**0.8779342**
7	25	***50***	***71***	***180***	**0.8779342**
8	29	***55***	***75***	***179***	**0.8779342**
9	33	90	***60***	***176***	0.7557252
10	40	***61***	20	***186***	0.61884
11	26	***50***	***99***	***180***	**0.8779342**
12	38	92	***100***	***178***	0.7557252

5 Conclusiones

Si bien existen varias aplicaciones de *VPRSM* a la solución de diversos problemas [14], [15], [16], [17] y, en especial, a problemas estadísticos [18], el resultado que se presenta en este trabajo representa una aplicación novedosa de dicho modelo al problema de la detección de *outliers* rompiendo con el esquema tradicional seguido por la mayoría de los métodos de detección existentes. A partir del establecimiento de ciertas condiciones iniciales —*concepto y relaciones de equivalencia*—, proporciona, de forma no supervisada, resultados generales con respectos a todos los elementos del *conjunto de datos* (*universo*).

En especial, el algoritmo *BM/Probabilístico* proporciona la probabilidad de cada elemento del *universo* de ser *outlier* en dicho *universo* sin la necesidad de haber establecido —excepto las señaladas en el párrafo anterior— las condiciones previas para ello en función del contexto de aplicación. Este hecho da trascendencia y originalidad a este resultado pues a partir de él, se allana el camino para el análisis y la solución de otros problemas particulares. Permite tener una visión general sobre los datos y así poner a prueba su representatividad.

Los algoritmos presentados permitieron validar la viabilidad computacional de los métodos propuestos. Constituyen además, soluciones computacionales eficientes —en cuanto a complejidad temporal y espacial— para la solución de los problemas para los cuales fueron concebidos. Esto es una ventaja que cualquier analista/ingeniero de datos valora considerablemente.

El método propuesto resuelve, además, otras limitaciones de varios métodos de detección: pueden ser aplicados a conjuntos de datos donde exista mezcla de tipos de atributos (continuos y discretos); para su aplicación no se requiere conocimiento a priori sobre la *distribución* de los datos; dentro del ámbito de aplicación de los mismos, el tamaño y la dimensionalidad del *conjunto de datos* no es una limitación para su correcto funcionamiento; ninguno requiere para su aplicación el establecimiento de criterios de distancia o de densidad con relación a los datos del conjunto; el método permiten obtener de forma no supervisada —en lo que respecta al establecimiento del valor de los umbrales que intervienen en el análisis— resultados generales para cada elemento del *universo*. Sin embargo, el establecimiento de *umbrales de excepcionalidad* por parte del usuario son requisitos indispensables para garantizar el funcionamiento correcto de varios métodos de detección de *outliers*.

Los resultados expuestos en el presente trabajo no son más que el comienzo de una investigación más profunda en el contexto del problema general de la detección de outliers basada en el modelo de RS. Por tanto, se pueden identificar varios problemas que aún no han sido solucionados y que pueden constituir objetivos inmediatos para dar continuidad a la investigación. En tal sentido se han identificado los siguientes:

- Con el objetivo de mejorar aún más el tiempo de ejecución de los algoritmos se podría crear un mecanismo distribuido de ejecución para aprovechar el poder de cómputo de varias máquinas en un dominio. En la versión actual de los algoritmos el usuario tiene que ejecutar los mismos en una sola PC.

- En la versión actual del algoritmo *BM*/Probabilístico el dominio del umbral β es [0; 0,5]. Sin embargo, el establecimiento de una nueva cota superior podría permitir ganar en precisión en el cálculo de la probabilidad, especialmente, para el caso de los elementos *muy contradictorios* para pocos valores de β. En tal sentido se propone hacer una modificación al algoritmo *BM/probabilístico* a partir de la cual se determine de forma automática cuál es el valor más adecuado para dicha cota.

Referencias

1. Pawlak, Z., (1982) Rough Sets. International Journal of Computer and Information Sciencies, 11(5):341-356
2. Sang Wook, H., & Jae-Yearn, K. (2007). Rough Set-based Decision Tree using the Core Attributes Concept. ICICIC'07, The Second International Conference on Innovative Computing, Information and Control. Kumamoto, Japan: IEEE Computer Society.
3. Ching-Hsue, C., You-Shyang, C., & Jr-Shian, C. (2007). Classifying Initial Returns of Electronic Firm's IPOs Using Entropy Based Rough Sets in Taiwan Trading Systems. ICICIC'07, The Second International Conference on Innovative Computing, Information and Control. Kumamoto, Japan: IEEE Computer Society.
4. Hirokane, M., Konishi, H., Miyamoto, A., Nishimura, F. (2007). Extraction of minimal decision algorithm using rough sets and genetic algorithm. Wiley InterSciences, Systems and Computers in Japan , Volume 38, Issue 4, Pages: 39-51.
5. Jiang, F., Sui, Y., & Cao, C., (2005). Outlier detection using rough sets theory. Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC 2005). Springer.
6. Fernández, A., Abreu, M., Fernández, M.C., Maciá, F. (2009). Algoritmo para la detección de casos excepcionales basado en la Teoría de Conjuntos Aproximados. Jornadas para el desarrollo de grandes aplicaciones de red (JDARE'09). Alicante, España. ISSN: 1889-7819, pp. 109-130.
7. Ziarko, W., (1993). Variable Precision Rough Set Model. s.l. : Journal of Computer and System Sciences, 46(1):39-59.
8. Ziarko, W., (1994). Rough Sets, Fuzzy Sets and Knowledge Discovery, Springer Verlag, 326-334.
9. Ziarko, W., (2001). Probabilistic Decision tables in the Variable Precision Rough Set Model. Computer Science Department, University of Regina, Regina, Saskatchewan, S4S 0A2, Canada.
10. Fernández, A., Abreu, M., Fernández, M.C., Maciá, F. (2009). Método de detección no determinista de outliers basado en el modelo de conjuntos aproximados de precisión variable. Jornadas para el desarrollo de grandes aplicaciones de red (JDARE'09). Alicante, España.ISSN: 1889-7819, pp. 131-148.
11. Pawlak, Z., (1991). Rough Sets: Theoretical Aspects of Reasoning About Data. s.l.: Spinger.
12. Fristedt, B. E., Gray, L. F. (1996). A Modern Approach to Probability Theory, Birkhäuser, 1996, Chapter 2.
13. UCI Machine Learning Repository. http://cml.ics.uci.edu. Última consulta: 30/05/09
14. Gong, ZT., Sun, BZ., Shao, YB., Chen, DG., He, Q. (2004) Variable precision rough set model based on general relations. Proceedings of the Third Conference on Machine Learning and Cybernetics, Shanghai.

15. Beynon, M. J., Driffield, N. (2005). An illustration of variable precision rough sets model: an analysis of the findings of the UK Monopolies and Mergers Commission. Computers and Operations Research Volume 32, Issue 7, Pages: 1739-1759.
16. Su, ChT., Hsu, JH. (2006). Precision parameter in the variable precision rough sets model: an application. Volume 34, Issue 2, Pages: 149-157.
17. Maheswari, V.U., Siromoney, A. Mehata, K.M. (2001). The Variable Precision Rough Set Model for Web Usage Mining. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, Volume 2198/2001. Web Intelligence Research and Development, pages: 520-524
18. Ziarko, W., (1999). Decision making with probabilistic decision tables. Proceedingas of the 7th International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing (RSFDGrC' 99). Yamaguchi, Japan, Lectures Notes in AI 1711, Springer Verlag, pp. 463-471.