Un Algoritmo Genético para Selección de Características en Sistemas de Clasificación Basados en Reglas Difusas para conjuntos de datos altamente no balanceados

Pedro Villar[1], Alberto Fernández[2], Ana María Sánchez[1], Francisco Herrera[2]

[1]Dpto. de Lenguajes y Sistemas Informáticos,
[2]Dpto. de Ciencias de la Computación e Inteligencia Artificial,
E.T.S. Ing. Informática y de Telecomunicación, Universidad de Granada.
pvillarc@ugr.es,alberto@decsai.ugr.es,
amlopez@ugr.es,herrera@decsai.ugr.es

Resumen. En este trabajo se propone un método para seleccionar las variables más relevantes a la hora de construir un Sistema de Clasificación Basado en Reglas Difusas para problemas con datos no balanceados, una situación que está presente en numerosos problemas reales en los que la distribución de clases no es uniforme. El objetivo de esta propuesta es obtener un sistema de complejidad reducida y con una buena tasa de clasificación utilizando un algoritmo genético para selección de características y un método simple para la generación de las reglas difusas.

Palabras Clave: Sistemas de Clasificación Basados en Reglas Difusas, datos no balanceados, selección de características, Algoritmos Genéticos

1 Introducción

En el entorno de los problemas de clasificación, un conjunto de datos no balanceado es aquel en el que el número de instancias para cada una de las clases difiere mucho entre ellas. Además, la clase menos representada suele ser la que tiene más interés desde el punto de vista del proceso de aprendizaje. Los datos no balanceados aparecen en muchos problemas reales de clasificación, algunos ejemplos son el reconocimiento facial [1], control de riesgos [2] y aplicaciones médicas [3].

En este trabajo nos centraremos en problemas de clasificación binarios no balanceados con un alto porcentaje de no balanceo. Para diseñar el clasificador utilizaremos Sistemas de Clasificación Basados en Reglas Difusas (SCBRDs), que es una herramienta muy utilizada en el campo del aprendizaje automático ya que proporcionan modelos interpretables para el usuario final [4]. Trabajos recientes han demostrado que los SCBRDs tienen un buen comportamiento trabajando con conjuntos de datos no balanceados [5].

Un SCBRD tiene dos componentes principales: el Sistema de Inferencia y la Base de Conocimiento (BC). En un SCBRD lingüístico, la BC está compuesta

de la Base de Reglas (BR), constituida por el conjunto de reglas difusas y la Base de Datos (BD), que contiene las funciones de pertenencia de las particiones difusas asociadas a las variables de entrada. Si no existe información experta sobre el problema a resolver, es necesario utilizar algún proceso de aprendizaje automático de la BC a partir de ejemplos.

Por otro lado, en bastantes problemas de clasificación el elevado número de variables conlleva que la BR tenga un gran número de reglas y, por tanto, no sea demasiado interpretable, o incluso el SCBRD puede presentar un cierto grado de *sobreaprendizaje*. Este problema se puede solucionar reduciendo el número de reglas de la BR o seleccionando las características más relevantes. Los métodos de reducción de reglas tienen problemas cuando la dimensión del problema es muy grande o cuando existe un elevado número de ejemplos. Para estos casos es más aconsejable un proceso de selección de características previo al proceso de aprendizaje del SCBRD o durante dicho proceso [6].

Nuestro principal objetivo es analizar la importancia de la selección de características en problemas de clasificación con datos altamente no balanceados, para lo que desarrollaremos un proceso de aprendizaje evolutivo que nos permita obtener un SCBRD. Dicho proceso emplea un Algoritmo Genético (AG) para seleccionar las variables relevantes y utiliza un método clásico para derivar la BR, el algoritmo propuesto en [7] (método de Chi en adelante). Los resultados obtenidos se compararán con dos métodos sin selección de variables: el mencionado método de Chi además de otro modelo de clasificación no basado en reglas difusas, C4.5 [8], un algoritmo de árboles de decisión que se ha utilizado como referencia en el ámbito de problemas con datos no balanceados [9–11].

Para el estudio experimental hemos seleccionado una amplia colección de conjuntos de datos con alto grado de no balanceo obtenidos del repositorio UCI [12]. Con el propósito de trabajar adecuadamente con datos no balanceados, utilizamos una técnica de preprocesamiento, "Synthetic Minority Over-Sampling TEchnique" (SMOTE) [13], que balancea los ejemplos del conjunto de entrenamiento entre las dos clases. Además, se llevará a cabo un estudio estadístico utilizando tests no paramétricos para comprobar si existen diferencias significativas entre los resultados obtenidos [14–16].

El trabajo está estructurado de la siguiente manera. Primero, en la Sección 2 se realiza una introducción a los problemas que plantean los conjuntos de datos no balanceados, describiendo sus características, cómo se puede trabajar con ellos y las métricas que se emplean en dicho campo. A continuación, en la Sección 3 presentaremos los aspectos fundamentales del método que se propone. La sección 4 muestra el estudio experimental realizado. Finalmente, las conclusiones de este trabajo se exponen en la Sección 5.

2 Conjuntos de datos no balanceados en Clasificación

El aprendizaje a partir de datos no balanceados es un tema importante que ha aparecido recientemente en el ámbito del aprendizaje automático [17]. Su importancia radica en que está presente en bastantes problemas reales de clasificación.

Concretamente, nos referimos a conjuntos de datos no balanceados cuando la distribución de clases no es uniforme, por lo que el número de ejemplos que representa a una de las clases es mucho menor que en las otras clases, además de que la caracterización de esa clase suele tener un mayor interés práctico.

Los algoritmos clásicos para clasificación a partir de ejemplos suelen favorecer a la clase mayoritaria (con un mayor número de ejemplos), ya que las reglas que clasifican correctamente un mayor número de ejemplos son seleccionadas en el proceso de aprendizaje al aumentar la métrica considerada (que suele estar basada en el porcentaje de ejemplos bien clasificados). Por tanto, las instancias de la clase minoritaria son mal clasificadas con una frecuencia mayor que las que pertenecen a la clase mayoritaria [18]. Otra característica importante de este tipo de problemas son los "small disjuncts", que consisten en una concentración de datos de una única clase en un pequeño espacio del problema siendo rodeada por ejemplos de la clase contraria, [19]; este tipo de regiones son difíciles de detectar para la mayoría de algoritmos de aprendizaje. Además, otro de los principales problemas de los conjuntos de datos no balanceados es el solapamiento entre ejemplos de la clase mayoritaria y de la clase minoritaria [20].

Existen diferentes grados de no balanceo entre los datos. En este trabajo utilizamos el "imbalance ratio" (IR) [21] para distinguir las diferentes categorías. Se define como la razón entre el número de ejemplos de la clase mayoritaria y el número de ejemplos de la clase minoritaria. Consideraremos que un conjunto de datos presenta un alto grado de no balanceo cuando su IR es mayor que 9 (menos de un 10% de instancias de la clase minoritaria).

En un trabajo previo sobre este tema [5], se analizó el efecto de utilizar métodos de preprocesamiento de datos antes de aprender el SCBRD, comprobando el buen comportamiento de los métodos de sobremuestreo, especialmente la metodología SMOTE [13], por lo que será empleada para los experimentos desarrollados en esta contribución. Básicamente, SMOTE trata de construir nuevos ejemplos de la clase minoritaria por interpolación entre varios ejemplos de dicha clase que se encuentran próximos. De esta manera, se evita el problema del sobreaprendizaje y las fronteras de decisión para la clase minoritaria se desplazan algo más dentro del espacio de la clase mayoritaria.

La mayoría de propuestas para aprendizaje automático de clasificadores utilizan alguna medida de precisión del modelo como el porcentaje de ejemplos bien clasificados. Sin embargo, ese tipo de medidas pueden llevar a conclusiones erróneas cuando se trabaja con datos no balanceados ya que no tienen en cuenta la proporción de ejemplos de cada clase. Por ese motivo, en este trabajo usaremos la medida denominada "Area Under the Curve" (AUC) [22], que se define como:

$$AUC = \frac{1 + TP_{rate} - FP_{rate}}{2} \tag{1}$$

donde TP_{rate} es el porcentaje de ejemplos de la clase minoritaria bien clasificados y FP_{rate} es el porcentaje de ejemplos de la clase mayoritaria mal clasificados (expresados en tanto por uno).

3 Algoritmo Genético para Selección de Características

En este apartado se propone un método de aprendizaje automático de la BC de un SCBRD mediante un AG generacional binario que permite definir las variables relevantes para el proceso de clasificación (selección de características). El proceso de evaluación de las posibles soluciones incluye un sencillo método de generación de la BR considerando solamente las variables seleccionadas. El algoritmo utilizado es el método de Chi [7]. Las particiones difusas de cada variable tienen tres etiquetas lingüísticas triangulares distribuidas uniformemente a lo largo del dominio de dicha variable. Esta elección se basa en trabajos previos en los que el método de Chi obtuvo un muy buen comportamiento considerando dicha configuración (tres etiquetas en particiones difusas uniformes) en SCBRDs para problemas no balanceados [5, 23].

Notaremos nuestra propuesta como AG-SC (Algoritmo Genético para Selección de Características). El propósito principal de AG-SC es la obtención de SCBRDs con una buena precisión en función de la métrica utilizada y una alta interpretabilidad. Desafortunadamente, es difícil conseguir esos dos objetivos simultáneamente. Normalmente, los SCBRDs con buena precisión tienen un elevado número de variables seleccionadas y bastantes reglas, por lo que resultan poco interpretables, mientras que los sistemas cómodamente interpretables (pocas variables y no muchas reglas) suelen ser poco precisos. Por otro lado, hay que tener en cuenta que, al diseñar la BC, es posible que aparezca un *sobreaprendizaje* del conjunto de datos de entrenamiento que hace que el sistema generalice mal al aplicarlo a nuevos ejemplos. Para intentar evitar esos problemas se penalizarán los SCBRDs con un alto número de variables seleccionadas como se explicará en la sección 3.3. Los siguientes apartados describen los principales componentes de AG-SC.

3.1 Codificación de las soluciones

Para un problema de clasificación de N variables, cada cromosoma está compuesto por un vector binario de longitud N que indica si la variable correspondiente está seleccionada (1) o no (0).

3.2 Población inicial

La población inicial está dividida en seis grupos. El primero de ellos (5% de los individuos) tiene todas las variables seleccionadas. Los cuatro siguientes, cada uno de ellos con el 20% del total de cromosomas, tienen distinto porcentaje (75%, 50%, 25% y 10% respectivamente) de variables seleccionadas de forma aleatoria. En el resto (15% de los individuos), la inicialización es totalmente aleatoria.

3.3 Evaluación de los cromosomas

La evaluación de un cromosoma consta de tres etapas:

1. Generar la BD utilizando la información del cromosoma. Para todas las variables seleccionadas se construye una partición difusa uniforme con 3 etiquetas utilizando funciones de pertenencia triangulares.
2. Generar la BR a partir de la BD mediante el proceso de aprendizaje de reglas difusas considerado (método de Chi)
3. Calcular el valor de la función de evaluación. Tal y como se ha comentado anteriormente, para mejorar la capacidad de generalización del SCBRD resultante y evitar el sobreaprendizaje, penalizaremos ligeramente los SCBRDs con un alto número de variables seleccionadas (VS). Así, una vez calculado el AUC sobre el conjunto de datos de entrenamiento, la función de evaluación que el AG intenta minimizar es:

$$F_C = \omega_1 \cdot (1 - AUC) + \omega_2 \cdot VS$$

siendo VS el número de variables seleccionadas. Para normalizar esos dos valores, se calcula ω_2 a partir del AUC del SCBRD obtenido al ejecutar el método de Chi considerando una BD con todas las variables seleccionadas (AUC_N) y del número de variables(N) de la siguiente manera:

$$\omega_2 = \alpha_{\omega_2} \cdot \frac{AUC_N}{N}$$

por tanto, ω_1 y α_{ω_2} indican, respectivamente, los pesos del error de clasificación y del número de variables seleccionadas en la función de evaluación.

3.4 Operadores genéticos

La selección es por torneo binario, en la que dos cromosomas son seleccionados aleatoriamente de la población y el que tenga mayor fitness se escoge para ser incluido en la siguiente población, después de la aplicación de los operadores genéticos. El mecanismo de recombinación es el operador clásico de cruce en un punto, es decir, se escoge aleatoriamente el punto de cruce p y los dos padres se cruzan sobre la variable p. Respecto a la mutación, se emplea el operador clásico de mutación para codificación binaria ("bit-flip").

4 Estudio experimental

Vamos a analizar el comportamiento de AG-SC en una amplia selección de conjuntos de datos no balanceados con una alta tasa de no balanceo ($IR > 9$). Más especificamente, hemos utilizado 22 conjuntos de datos del repositorio UCI [12] con diferente IR, como se muestra en la Tabla 1, donde se indican el número de ejemplos (#Ej.), número de variables (#Var.), nombre de cada clase (minoritaria y mayoritaria), la distribución de ejemplos en cada clase y la tasa de no balanceo (IR). La tabla se muestra en orden ascendente de IR. Los problemas multiclase se han modificado para obtener problemas no balanceados con sólo dos clases, uniendo una o más clases dentro de la clase mayoritaria o minoritaria (en casi todos los conjuntos de datos se unen en la clase mayoritaria, salvo en *Ecoli013*, que se unen en ambas).

Tabla 1. Descripción resumida de los conjuntos de datos no balanceados

Conj. de datos	#Ej.	#Var.	Clase (min.; may.)	%Clase (min., may.)	IR
Conjuntos de datos altamente no balanceados (IR mayor que 9)					
Yeast2vs4	514	8	(cyt; me2)	(9.92, 90.08)	9.08
Yeast05679vs4	528	8	(me2; mit,me3,exc,vac,erl)	(9.66, 90.34)	9.35
Vowel0	988	13	(hid; remainder)	(9.01, 90.99)	10.10
Glass016vs2	192	9	(ve-win-float-proc; build-win-float-proc, build-win-non_float-proc,headlamps)	(8.89, 91.11)	10.29
Glass2	214	9	(Ve-win-float-proc; remainder)	(8.78, 91.22)	10.39
Ecoli4	336	7	(om; remainder)	(6.74, 93.26)	13.84
Yeast1vs7	459	8	(nuc; vac)	(6.72, 93.28)	13.87
Shuttle0vs4	1829	9	(Rad Flow; Bypass)	(6.72, 93.28)	13.87
Glass4	214	9	(containers; remainder)	(6.07, 93.93)	15.47
Page-blocks13vs2	472	10	(graphic; horiz.line,picture)	(5.93, 94.07)	15.85
Abalone9vs18	731	8	(18; 9)	(5.65, 94.25)	16.68
Glass016vs5	184	9	(tableware; build-win-float-proc, build-win-non_float-proc,headlamps)	(4.89, 95.11)	19.44
Shuttle2vs4	129	9	(Fpv Open; Bypass)	(4.65, 95.35)	20.5
Yeast1458vs7	693	8	(vac; nuc,me2,me3,pox)	(4.33, 95.67)	22.10
Glass5	214	9	(tableware; remainder)	(4.20, 95.80)	22.81
Yeast2vs8	482	8	(pox; cyt)	(4.15, 95.85)	23.10
Yeast4	1484	8	(me2; remainder)	(3.43, 96.57)	28.41
Yeast1289vs7	947	8	(vac; nuc,cyt,pox,erl)	(3.17, 96.83)	30.56
Yeast5	1484	8	(me1; remainder)	(2.96, 97.04)	32.78
Ecoli0137vs26	281	7	(pp,imL; cp,im,imU,imS)	(2.49, 97.51)	39.15
Yeast6	1484	8	(exc; remainder)	(2.49, 97.51)	39.15
Abalone19	4174	8	(19; remainder)	(0.77, 99.23)	128.87

Para reducir los efectos negativos de usar datos altamente no balanceados, emplearemos el método de preprocesamiento SMOTE [13] para todos los experimentos, considerando solamente el vecino más cercano para generar nuevos ejemplos, hasta balancear ambas clases hacia una distribución uniforme (50%).

Con objeto de analizar la selección de variables, compararemos los resultados obtenidos por AG-SC con los obtenidos por el método de Chi sin selección (siempre con tres etiquetas por particición difusa). Además, también se establecerá una comparación con C4.5, un método de referencia en el ámbito de los conjuntos de datos no balanceados [10, 11].

Las características de configuración de los algoritmos de aprendizaje automático de SCBRDs (método de Chi y AG-SC) se presentan a continuación. Esta elección está justificada por resultados previos obtenidos con el método de Chi para clasificación con datos no balanceados [5, 23]: 3 etiquetas difusas, T-norma producto como operador de conjunción, junto con la heurísitca de Factor de certeza penalizado [24] para el peso de las reglas y Método de razonamiento difuso de la regla ganadora.

Para garantizar unos resultados no dependientes de la partición que se realice del conjunto de ejemplos, utilizamos un modelo de validación cruzada en la que se divide cada conjunto de ejemplos en cinco particiones de igual tamaño. Se realizaron entonces cinco experimentos para cada conjunto de datos reservando una de las particiones como conjunto de datos de test y la combinación de las otras cuatro restantes como datos de entrenamiento. Como los AGs son métodos estocásticos, se realizaron tres ejecuciones de AG-SC con diferentes semillas para la secuencia pseudo-aleatoria en cada una de las particiones. De esta manera,

para cada conjunto de datos se consideran los resultados medios de quince ejecuciones de AG-SC (cinco particiones con tres semillas cada una). Por otro lado, el test de rangos de Wilcoxon [25] se usa para una comparación estadística de los resultados obtenidos. Los parámetros específicos del AG del método AG-SC se muestran a continuación, siendo N el número de variables del problema:

- Número de evaluaciones: $500 \cdot N$
- Tamaño de población: 100
- Probabilidad de cruce: 0.6
- Probabilidad de mutación: 0.2
- Pesos para la función de evaluación (Sección 3.3): ω_1: 0.9, α_{ω_2}: 0.1

La Tabla 2 muestra los resultados obtenidos en precisión de los métodos (usando la medida AUC), tanto para AG-SC, como para los algoritmos usados para comparación (Chi y C4.5), siendo AUC_{entr}, el AUC sobre el conjunto de datos de entrenamiento y AUC_{test} el AUC sobre el conjunto de datos de test. La última fila muestra la media del número de reglas (en todas las ejecuciones) de los modelos obtenidos con cada método.

Tabla 2. Resultados detallados para los métodos analizados

Conjunto de datos	Chi		AG-SC		C4.5	
	AUC_{entr}	AUC_{test}	AUC_{entr}	AUC_{test}	AUC_{entr}	AUC_{test}
Yeast2vs4	.8968	.8736	.8717	.8092	.9814	.8588
Yeast05679vs4	.8265	.7917	.8048	.7942	.9526	.7602
Vowel0	.9857	.9839	.9674	.9555	.9967	.9494
Glass016vs2	.6271	.5417	.7156	.5860	.9716	.6062
Glass2	.6654	.5530	.7435	.6104	.9571	.5424
Ecoli4	.9406	.9151	.9440	.9275	.9769	.8310
shuttle0vs4	1.0000	.9912	1.0000	1.0000	.9999	.9997
yeastB1vs7	.8200	.8063	.7756	.7284	.9351	.7003
Glass4	.9527	.8570	.9611	.9267	.9844	.8508
Page-Blocks13vs4	.9368	.9205	.9825	.9831	.9975	.9955
Abalone9-18	.7023	.6470	.7363	.6949	.9531	.6215
Glass016vs5	.9057	.7971	.9226	.8790	.9921	.8129
shuttle2vs4	.9500	.9078	.9949	.9918	.9990	.9917
Yeast1458vs7	.7125	.6465	.6731	.5723	.9158	.5367
Glass5	.9433	.8317	.9406	.8085	.9976	.8829
Yeast2vs8	.7861	.7728	.7978	.7707	.9125	.8066
Yeast4	.8358	.8315	.8374	.8218	.9101	.7004
Yeast1289vs7	.7470	.7712	.7373	.6697	.9465	.6832
Yeast5	.9468	.9358	.9477	.9476	.9777	.9233
Yeast6	.8848	.8809	.8802	.8647	.9242	.8280
Ecoli0137vs26	.9396	.8190	.8910	.8124	.9678	.8136
Abalone19	.7144	.6394	.7230	.6960	.8544	.5202
Media	.8509	.8052	.8567	.8114	.9593	.7825
Media número reglas	68.67		**8.53**		22.45	

Como puede observarse, AG-SC obtiene mejores resultados (en promedio) que el método de Chi, tanto en AUC_{entr} como AUC_{test}, demostrando la influencia significativa de la selección de variables en el comportamiento del clasificador. Además, AG-SC presenta mejores resultados en AUC_{test} que C4.5. Esta situación se representa estadísticamente mediante el test de Wilcoxon en la Tabla 4, donde se puede observar que AG-SC presenta mayor rango. Además, hay que destacar que el método de Chi no mejora al algoritmo C4.5 en este ámbito de conjuntos de datos altamente no balanceados, pero la aplicación de una selección de variables en AG-SC sí obtiene diferencias significativas en la comparación.

Tabla 3. Test de Wilcoxon para comparar AG-SC con el método de Chi y C4.5 de acuerdo a su AUC_{test}. R^+ corresponde a la suma de rangos del primer algoritmo a comparar y R^- a la suma de rangos del segundo

Comparación	R^+	R^-	Hipótesis ($\alpha = 0.05$)	p-valor
AG-SC vs Chi	142.0	111.0	No Rechazada	0.615
Chi vs C4.5	176.0	77.0	No Rechazada	0.108
AG-SC vs C4.5	188.0	65.0	Rechazada para AG-SC	0.046

Tal y como se puede comprobar en la última fila de la Tabla 2, estos buenos resultados se han obtenido con una gran reducción en el número de reglas (respecto al método de Chi sin selección y a C4.5), obteniendo de esta manera modelos de una buena precisión y gran interpretabilidad. Esta situación se ha conseguido mediante una adecuada selección de características.

La Tabla 4 muestra las variables seleccionadas por AG-SC. La primera columna (VS) indica el número medio de variables seleccionadas y el resto de columnas indican, para cada una de las variables del problema, el porcentaje de veces que ha sido seleccionada dentro de las 15 ejecuciones, por lo que un 1.00 indica que la variable siempre ha sido seleccionada mientras que un 0.8 indicaría que ha sido seleccionada en 12 de las 15 ejecuciones.

Tabla 4. Media del número de variables seleccionadas por AG-SC

Conjunto		Variables												
de datos	VS	1	2	3	4	5	6	7	8	9	10	11	12	13
Yeast2vs4	1.27	0.60	0.13	0.53	0.00	0.00	0.00	0.00	0.00	-	-	-	-	-
Yeast05679vs4	1.20	1.00	0.00	0.13	0.00	0.07	0.00	0.00	0.00	-	-	-	-	-
Vowel0	3.40	0.00	0.20	0.20	0.60	1.00	0.40	0.20	0.00	0.00	0.20	0.40	0.00	0.20
Glass016vs2	2.13	0.33	0.20	0.00	0.27	0.73	0.20	0.20	0.00	0.20	-	-	-	-
Glass2	3.00	0.40	0.20	0.00	0.60	0.80	0.00	0.80	0.00	0.20	-	-	-	-
Ecoli4	2.40	0.00	0.40	0.00	0.00	1.00	1.00	0.00	-	-	-	-	-	-
Shuttle0vs4	1.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	-	-	-	-
Yeast1vs7	2.53	0.67	0.07	0.87	0.27	0.00	0.07	0.60	-	-	-	-	-	-
Glass4	3.60	0.20	0.00	0.00	1.00	0.60	0.60	1.00	0.00	0.20	-	-	-	-
Page-Blocks13vs4	2.00	1.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	-	-	-
Abalone9-18	1.40	0.00	0.00	0.00	0.40	0.00	0.00	0.00	1.00	-	-	-	-	-
Glass016vs5	2.07	0.00	0.20	0.87	0.00	0.13	0.07	0.13	0.60	0.07	-	-	-	-
Shuttle2vs4	2.33	0.47	0.00	0.87	0.07	0.00	0.00	0.87	0.00	0.07	-	-	-	-
Yeast1458vs7	2.27	0.60	0.20	0.33	0.27	0.00	0.00	0.07	0.80	-	-	-	-	-
Glass5	3.20	0.20	0.60	0.60	0.00	0.60	0.00	0.00	0.80	0.40	-	-	-	-
Yeast2vs8	2.00	0.60	0.00	0.00	0.20	0.00	1.00	0.00	0.20	-	-	-	-	-
Yeast4	2.20	1.00	0.00	0.20	0.40	0.60	0.00	0.00	0.00	-	-	-	-	-
Yeast1289vs7	2.20	0.87	0.00	0.87	0.07	0.13	0.00	0.13	0.13	-	-	-	-	-
Yeast5	1.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	-	-	-	-	-
Yeast6	1.20	0.60	0.40	0.00	0.00	0.00	0.00	0.00	0.20	-	-	-	-	-
Ecoli0137vs26	1.80	0.40	0.27	0.87	0.00	0.00	0.20	0.07	-	-	-	-	-	-
Abalone19	2.40	0.20	0.20	0.00	0.00	0.80	0.20	0.00	1.00	-	-	-	-	-

Como puede observarse, el número de variables seleccionadas es extraordinariamente bajo. En todos los problemas se reduce el número de variables un mínimo del 60%, y en siete problemas se selecciona menos de dos variables en la media de las 15 ejecuciones. Esta situación ha permitido obtener modelos compactos con pocas variables y pocas reglas, lo que demuestra que la selección de características es una buena opción para obtener modelos precisos y fácilmente interpretables, que era el principal objetivo de este trabajo.

5 Conclusiones

En este trabajo se ha analizado la selección de características en el aprendizaje de SCBRDs para problemas con datos altamente no balanceados. Para la selección de variables se utiliza un AG mientras que se emplea un método eficiente (método de Chi) para la generación de las reglas difusas.

Los resultados del método propuesto (AG-SC) muestran la gran influencia de la selección de variables en el comportamiento de los SCBRDs ya que AG-SC proporciona clasficadores con alta precisión y complejidad muy reducida si es comparada con el método de Chi considerando todas las variables como seleccionadas y con el algoritmo C4.5, usado normalmente en el ámbito de clasificación con datos no balanceados.

Finalmente, nos gustaría destacar dos ventajas de nuestra propuesta:

- Se puede modificar el balance interpretabilidad/precisión del modelo obtenido cambiando los pesos que intervienen en la función de evaluación.
- El AG utilizado se puede utilizar con cualquier método de generación de reglas difusas. En este trabajo se ha utilizado un método sencillo para remarcar la importancia de la selección de características pero podría emplearse otro con mayor precisión.

Agradecimientos

Este trabajo está soportado por el Ministerio de Ciencia y Tecnología en el marco del proyecto TIN2008-06681-C06-01.

Bibliografía

1. Y.H. Liu and Y.T. Chen. Face recognition using total margin-based adaptive fuzzy support vector machines. *IEEE Transactions on Neural Networks*, 18(1):178–192, 2007.
2. Y. M. Huang, C. M. Hung, and H. C. Jiau. Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem. *Nonlinear Analysis: Real World Applications*, 7(4):720–747, 2006.
3. M.A. Mazurowski, P.A. Habas, J.M. Zurada, J.Y. Lo, J.A. Baker, and G.D. Tourassi. Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance. *Neural Networks*, 21(2-3):427–436, 2008.
4. H. Ishibuchi, T. Nakashima, and M. Nii. *Classification and modeling with linguistic information granules: Advanced approaches to linguistic Data Mining*. Springer-Verlag, 2004.
5. A. Fernández, S. García, M.J. del Jesus, and F. Herrera. A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets. *Fuzzy Sets and Systems*, 159(18):2378–2398, 2008.
6. J. Casillas, O. Cordón, M. J. del Jesus, and F. Herrera. Genetic feature selection in a fuzzy rule-based classification system learning process for high dimensional problems. *Information Sciences*, 136(1-4):135–157, 2001.

7. Z. Chi, H. Yan, and T. Pham. *Fuzzy algorithms with applications to image processing and pattern recognition.* World Scientific, 1996.
8. J. R. Quinlan. *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers, San Mateo-California, 1993.
9. G.E.A.P.A. Batista, R.C. Prati and M.C. Monard. A Study of the Behaviour of Several Methods for Balancing Machine Learning Training Data. *SIGKDD Explorations*, 6(1):20–29, 2004.
10. A. Estabrooks, T. Jo, and N. Japkowicz. A multiple resampling method for learning from imbalanced data sets. *Computational Intelligence*, 20(1):18–36, 2004.
11. C.T. Su and Y.H. Hsiao. An evaluation of the robustness of MTS for imbalanced data. *IEEE Transactions on Knowledge Data Engineering*, 19(10):1321–1332, 2007.
12. A. Asuncion and D.J. Newman. UCI machine learning repository, 2007. University of California, Irvine, School of Information and Computer Sciences. `http://www.ics.uci.edu/~mlearn/MLRepository.html`
13. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligent Research*, 16:321–357, 2002.
14. J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.
15. S. García and F. Herrera An Extension on "Statistical Comparisons of Classifiers over Multiple data sets" for all Pairwise Comparisons. *Journal of Machine Learning Research*, 9:2607–2624, 2008.
16. S. García, A. Fernández, J. Luengo and F. Herrera A Study of Statistical Techniques and Performance Measures for Genetics-Based Machine Learning: Accuracy and Interpretability. *Soft Computing*, 13(10): 959–977, 2009.
17. N. V. Chawla, N. Japkowicz, and A. Kolcz. Editorial: special issue on learning from imbalanced data sets. *SIGKDD Explorations*, 6(1):1–6, 2004.
18. G. M. Weiss. Mining with rarity: a unifying framework. *SIGKDD Explorations*, 6(1):7–19, 2004.
19. G.M. Weiss and F. Provost. Learning when training data are costly: The effect of class distribution on tree induction. *Journal of Artificial Intelligence Research*, 19:315–354, 2003.
20. V. García, R.A. Mollineda, and J. S. Sánchez. On the k-NN performance in a challenging scenario of imbalance and overlapping. *Pattern Analysis Applications*, 11(3–4):269–280, 2008.
21. A. Orriols-Puig and E. Bernadó-Mansilla. Evolutionary rule-based systems for imbalanced datasets. *Soft Computing*, 13(3):213–225, 2009.
22. J. Huang and C. X. Ling. Using AUC and accuracy in evaluating learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 17(3):299–310, 2005.
23. A. Fernández, M.J. del Jesus, and F. Herrera. On the influence of an adaptive inference system in fuzzy rule based classification systems for imbalanced data-sets. *Expert Systems With Applications*, 36(6):9805–9812, 2009.
24. H. Ishibuchi and T. Yamamoto. Rule Weight Specification in Fuzzy Rule-Based Classification Systems. *IEEE Transactions on Fuzzy Systems*, 13:428–435, 2005.
25. D. Sheskin. *Handbook of parametric and nonparametric statistical procedures.* Chapman & Hall/CRC, second edition, 2006.