Obtención de los dominios de competencia de C4.5 por medio de medidas de separabilidad de clases

Julián Luengo Francisco Herrera
Dept. Ciencias de la Computación e Inteligencia Artificial
CITIC-Universidad de Granada, 18071 Granada
julianlm@decsai.ugr.es herrera@decsai.ugr.es

Resumen

Cuando se trata con problemas usando algoritmos de aprendizaje automático es difícil determinar *a priori* si éstos son adecuados para un problema concreto. En esta contribución se presenta un nuevo enfoque hacia este objetivo, basado en medidas de complejidad de datos para problemas de clasificación, extrayendo los dominios de competencia del algoritmo de aprendizaje.

Empleando medidas de complejidad de datos los dominios de competencia asociados al método de aprendizaje pueden ser extraídos describiendo las regiones en las cuales este algoritmo funciona adecuadamente o no en media. En esta contribución se extraen los dominios de competencia de C4.5 usando una categoría en particular de medidas de complejidad conocidas como medidas de separabilidad de clases. Estos dominios de competencia se definen por medio de reglas que describen tanto el buen como el mal comportamiento de C4.5 sobre un amplio conjunto de bases de datos obtenidas a partir de datos reales. La mayoría de las bases de datos son caracterizadas por estos dominios de competencia, mostrando las buenas capacidades descriptivas de las medidas consideradas.

1. Introducción

El estudio del rendimiento de algoritmos de Machine Learning (ML) no es una tarea reciente. Ha sido tratada tanto empíricamente como teóricamente [24, 16, 22]. Con el aumento de aplicaciones de ML y Minería de Datos y la creciente complejidad de los datos manejados, es útil conocer *a priori* si un algoritmo dado funcionará satisfactoriamente para un problema en concreto. Por tanto, uno de los temas de investigación actuales más importantes es identificar los dominios de competencia de un esquema de clasificación particular. Uno de los mayores obstáculos en este tipo de investigación es la dificultad en caracterizar las diferencias entre varios problemas del mundo real, y relacionar el comportamiento del clasificador con estas diferencias. Con esta información, sería posible por ejemplo estimar para bases de datos reales de gran tamaño si es interesante aplicar el método de aprendizaje (en bioinformática), o elegir los clasificadores relevantes de un ensemble de cara a una base de datos concreta.

La tarea de elegir el mejor algoritmo en función de un problema dado ha sido una cuestión clave en las últimas décadas. El problema del Meta-Aprendizaje (Meta-Learning -MetaL-) formalizó este proceso [10, 18, 6], pero se han encontrado algunos inconvenientes [14]. Una aproximación alternativa al MetaL es el paradigma de Transición de Fase (Phase Transition -PT-) [8]. Ha sido usado para estudiar la escalabilidad y el impacto de la barrera de complejidad en los rendimientos de aprendizaje [9]. Baskiotis y Sebag investigan el uso del paradigma PT para construir mapas de principios de competencia asignados al algoritmo de aprendizaje C4.5 en [3], caracterizando las regiones en las cuales este algoritmo funciona adecuadamente o no en promedio. Aquellos

problemas que son difíciles de caracterizar son descritos también, en la denominada región de transición de fase.

Esta contribución presenta una alternativa a las aproximaciones anteriores, empleando las medidas de complejidad de datos prupuestas por Ho y Basu [13] para analizar *a priori* el rendimiento de los algoritmos de aprendizaje. Estas medidas cuantifican aspectos particulares del problema que se consideran relevantes para la tarea de clasificación [4]. El estudio de las medidas de complejidad y sus aplicaciones se encuentra muy activo actualmente [7, 5, 12, 20, 17].

El objetivo de esta contribución es investigar el uso de las medidas de complejidad de datos para construir los dominios de competencia de algoritmos de aprendizaje de reglas, en particular el algoritmo de aprendizaje C4.5. Tales dominios de competencia tratan de caracterizar los problemas en los cuales C4.5 funcionará adecuadamente o no en promedio.

Para definir los dominios de competencia de C4.5 en esta propuesta emplearemos una categoría particular de medidas de complejidad de datos, formalmente conocidas como "Medidas de Separabilidad de Clases", las cuales han demostrado ser las más informativas para los algoritmos de aprendizaje de reglas [17]. Los dominios de competencia serán construidos a partir de un conjunto extenso de bases de datos, teniendo en cuenta dos conceptos clave:

- La precisión del modelo, considerando la precisión media en entrenamiento y en test, y la diferencia de cada una con el comportamiento global del método.
- La presencia de sobre-aprendizaje, observada en la diferencia entre la precisión en entrenamiento y en test del modelo.

Definiremos los dominios de competencia como reglas. Éstas pueden ser explotadas indicando el o los algoritmos más adecuados para un problema *a priori*. Pueden ser usadas también para determinar los dominios problemáticos de un algoritmo, caracterizando sus limitaciones en los datos.

El resto de la contribución está organizada como sigue. La Sección 2 presenta el enfoque de la complejidad de datos y define los dominios de competencia de un algoritmo de aprendizaje. En la Sección 3 se revisan brevemente los trabajos relacionados con esta contribución. En la Sección 4 se describe el proceso seguido para extraer los dominios de competencia de C4.5. La Sección 5 presenta los intervalos obtenidos y los dominios de competencia extraídos, y proporciona un análisis de los mismos. Finalmente, en la Sección 6 se muestran nuestras conclusiones.

2. Complejidad de datos y dominios de competencia

En esta sección presentamos las medidas de complejidad usadas en esta contribución en la Subsección 2.1, la categoría de las medidas de separabilidad de clases en la Subsección 2.2 y los dominios de competencia así como su definición en la Subsección 2.3.

2.1. Medidas de complejidad de datos

Ho y Basu [13] propusieron y reunieron doce medidas de complejidad de datos, que se encuentran resumidas en el Cuadro 1. Estas medidas de complejidad de datos son una serie de métricas que tratan de capturar diferentes aspectos o fuentes de complejidad que son consideradas complicadas para la tarea de clasificación [4].

Cuadro 1: Medidas de complejidad por categorías

Tipo	Identif.	Nombre
Medidas de solapamiento de atributos	F1	Razón discriminante de Fisher
	F2	Volumen de la región de solapamiento
	F3	Máxima eficiencia individual de los atributos
Medidas de separabilidad de las clases	L1	Mínimización de la suma de error distancia por Programación Lineal
	L2	Error del clasificador lineal por Programación Lineal
	N1	Fracción de puntos en los bordes de las clases
	N2	Media de la distancia de Vecinos Más Cercanos intra/inter-clases
	N3	Error del clasificador 1-NN
Medidas de geometría, topología y densidad	L3	No-linealidad del clasificador lineal por Programación Lineal
	N4	No-linealidad del clasificador 1-NN
	T1	Fracción de puntos con subconjuntos adheridos
	T2	Media de puntos por dimensión

Las medidas de complejidad de datos mencionadas han sido usadas para buscar las zonas de datos de mayor influencia para XCS [7] y la caracterización del rendimiento del método FH-GBML [17]. También han sido empleadas para estudiar el efecto de la complejidad de los datos en el clasificador de vecinos más cercanos [20] o incluso para analizar el comportamiento de la selección de prototipos con algoritmos evolutivos, considerando una medida de complejidad para problemas de clasificación basada en el solapamiento [12]. Una descripción más detallada de las doce medidas se puede encontrar en [4].

2.2. Medidas de separabilidad de las clases

En nuestro trabajo emplearemos una categoría particular de las doce medidas de complejidad, conocidas como "Medidas de Separabilidad de las Clases". Estas medidas proporcionan una caracterización indirecta de la separabilidad de las clases. Asumen que una clase está constituida a partir de una o múltiples variedades que constituyen el soporte de la distribución de probabilidad de la clase considerada. La forma, posición e interconexión de estas variedades indican pistas del grado de separación de dos clases, pero no describen la separabilidad directamente. La definición de las medidas es la que sigue.

L1: *suma del error de la distancia minimizada por programación lineal.* Los clasificadores lineales pueden obtenerse a partir de la formulación para la programación lineal propuesta por Smith [21]. El método minimiza la suma de las distancias de los puntos erróneos al hiperplano separador (substrayendo un margen constante):

$$\begin{aligned} &\text{minimize } \mathbf{a}^\mathbf{t}\mathbf{t} \\ \text{subject to } &\mathbf{Z}^\mathbf{t}\mathbf{w} + \mathbf{t} \geq \mathbf{b} \\ &\mathbf{t} \geq \mathbf{0} \end{aligned}$$

donde $\mathbf{a}$, $\mathbf{b}$ son vectores constantes arbitrarios (ambos elegidos para ser 1), $\mathbf{w}$ es el vector de pesos que debe ser determinado, $\mathbf{t}$ es un vector de error, y $\mathbf{Z}$ es una matriz donde cada columna $\mathbf{z}$ se define sobre un vector de entrada $\mathbf{x}$ (aumentado añadiendo una dimensión con un valor constante de 1) y su clase C (con valor C_1 o C_2) como sigue:

$$\begin{cases} \mathbf{z} = +\mathbf{x} & \text{if } C = C_1, \\ \mathbf{z} = -\mathbf{x} & \text{if } C = C_2. \end{cases}$$

El valor de la función objetivo en esta formulación es empleado como una medida (L1). La medida tiene un valor cero para un problema linealmente separable. Su valor puede verse seriamente afectado por los valores atípicos localizados en el lado erróneo del hiperplano óptimo. La medida se normaliza por el número de puntos en el problema y también por la longitud de la diagonal del la región hiperrectangular que contiene todos los puntos de entrenamiento en el espacio de características.

L2: *error del clasificador lineal definido por Programación Lineal (LP).* Esta medida es el error del clasificador lineal definido para L1, medido con el conjunto de entrenamiento. Con un conjunto de entrenamiento pequeño puede obtenerse una estimación del error verdadero muy sesgada.

N1: *fracción de puntos en la frontera de las clases.* Este método construye un árbol de expansión (sin considerar las clases) sobre la base de datos completa, y se cuenta el número de puntos conectados a una arista que una dos clases diferentes. La fracción de tales puntos sobre todos los puntos contenidos en la base de datos se emplea como una medida. Para dos clases fuertemente solapadas, la mayoría de los puntos se encuentran localizados próximos a la frontera de la clase. Sin embargo, esto puede ser cierto también para un problema linealmente separable que haya sido muestreado de manera dispersa con los márgenes entre las clases más cercanos que las distancias entre puntos de la misma clase.

N2: *media de la distancia de vecinos más cercanos intra/inter-clases.* Para cada instancia de entrada x_p, se calcula la distancia a su vecino más cercano en la clase (intraDist(x_p)) y la distancia al vecino más cercano de cualquier otra clase (interDist(x_p)). Entonces, el resultado es la razón de la suma de las distancias intra-clase y a la suma de las distancias

inter-clases para cada ejemplo de entrada.

$$N2 = \frac{\sum_{i=0}^{m} intraDist(x_i)}{\sum_{i=0}^{m} interDist(x_i)},$$

donde m es el número de ejemplos en cada base de datos. Esta métrica compara la dispersión en la clase con las distancias a los vecinos más cercanos de otras clases. Valores bajos indican que los valores de la misma clase se encuentran dispersos. Es sensible a las clases de los vecinos más cercanos a un punto, y también a la diferencia en magnitud de las distancias entre las clases y las internas a las propias clases.

N3: *error del clasificador 1-NN*. Esta medida es el error de un clasificador de vecinos más cercanos medido sobre el conjunto de entrenamiento. El error es estimado empleando el método *leave-one-out*. La medida denota cómo de cercanos se encuentran los ejemplos de diferentes clases. Valores bajos en esta medida indican que hay grandes espacios entre las fronteras de las clases.

2.3. Dominios de competencia por medio de medidas de complejidad de datos

Bernadó y Ho [7] definieron inicialmente el concepto de dominios de competencia para el clasificador XCS. Sus dominios de competencia indican la "región" del espacio de complejidad de problemas adecuados para las características del método de aprendizaje.

Tal caracterización podría ser útil para concentrar los esfuerzos en las mejoras del algoritmo de aprendizaje en esas áreas difíciles. Para poder establecer dichos límites, emplearon seis de las doce medidas de complejidad de datos. También observaron qué clase de medidas permite discriminar mejor entre los problemas sencillos y complejos para XCS.

En [17] se extiende la noción de dominios de competencia para un Sistema de Clasificación Basado en Reglas Difusas conocido como FH-GBML, empleando las doce medidas de complejidad de datos. Para ello se extraen intervalos específicos de las medidas en las cuales FH-GBML funciona bien o mal en promedio (en vez de relacionar su rendimiento a valores "altos" o "bajos"). Estos intervalos se codifican como reglas que constituyen los dominios de competencia del método de aprendizaje.

Estas reglas pueden usarse para predecir las regiones de comportamiento *a priori* del método de aprendizaje. En este trabajo analizaremos el uso de las medidas de separabilidad de las clases para obtener regiones o intervalos para cada medida, e intentar combinarlas para extraer con precisión los dominios de competencia de C4.5.

3. Trabajos relacionados

En esta sección se revisa brevemente los trabajos relacionados con la estimación *a priori* del rendimiento de un método de aprendizaje. Este problema de estimación fue formalizado como un nuevo problema de aprendizaje en el MetaL [10]. Por tanto el MetaL afronta el problema de la selección y representación de ejemplos de MetaL. Un ejemplo de MetaL implica un par en la mayoría de ocasiones (instancia del problema de ML, algoritmo de ML), etiquetados con el rendimiento del algoritmo en la instancia del problema de ML.

El MetaL presenta dos problemas conocidos:

- Cómo representar un problema de ML ha sido tratado usando diversos descriptores, como el número de ejemplos, número de atributos, porcentaje de valores perdidos, puntos de referencia [18]. La dificultad se debe al hecho de que los descriptores deben tener en cuenta la distribución de los ejemplos, que no es fácil de obtener en la mayoría de los casos.
- La segunda dificultad concierne a la selección de las instancias de problemas de ML. Kalousis [14] indica que la representatividad de los problemas y la perturbación inducen fuertes sesgos en el clasificador de MetaL.

Por otra parte el paradigma PT fue desarrollado inicialmente para comprender mejor los rendimientos de algoritmos de Satisfacción de Restricciones indicando donde se encuentran los problemas verdaderamente difíciles [11]. Por medio de este paradigm se puede observar una superficie de complejidad regular:

la complejidad es despreciable en dos regiones amplias, la región SI y NO, donde la probabilidad de satisfactibilidad es respectivamente cercana a 1 y cercana a 0. Estas regiones están separadas por la denominada transición de fase, donde los problemas más difíciles se concentran en media. Baskiotis y Sebag [3] adaptaron la representación de k-términos DNF de Rückert et al. [19] evaluando el rendimiento de C4.5 respecto al concepto objetivo subyacente.

Las medidas de complejidad de datos han sido usadas para estimar las capacidades de los algoritmos de aprendizaje, gracias a su capacidad de describir los aspectos más difíciles de los datos. Han sido empleadas para mostrar altas correlaciones entre el rendimiento de XCS y regiones de complejidad [7], y recientemente para caracterizar el rendimiento del método FH-GBML [17]. Ambas aproximaciones proporcionan información de los métodos de aprendizaje, que pueden ser usadas para describir los problemas fáciles y difíciles para ambos métodos *a priori*

4. Análisis del problema

En esta sección se presenta el proceso seguido para extraer los dominios de competencia de C4.5. La Subsección 4.1 describe la motivación presente detrás del uso de las medidas de complejidad de datos, y la Subsección 4.2 describe el proceso seguido para extraer dichos dominios.

4.1. Motivación

Determinar cuando un método funcionará bien o mal no es una tarea trivial, considerando la precisión como una medida de rendimiento. Un indicador inicial del rendimiento del método es la precisión en entrenamiento. Sin embargo, no es siempre una medida adecuada. La Figura 1 contienen los resultados de precisión en entrenamiento y test para C4.5 para todas las bases de datos usadas en este trabajo, dibujados en orden ascendente respecto a la precisión en entrenamiento.

Es importante indicar la presencia de sobreaprendizaje. Por tanto, la necesidad de otro

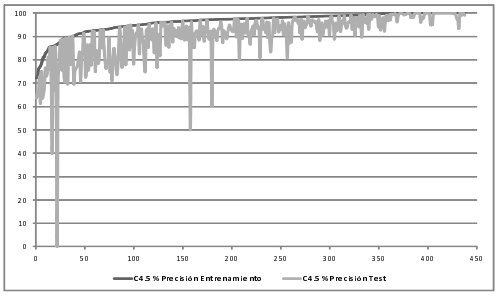

Figura 1: Precisión de C4.5 ordenada por los valores en entrenamiento

tipo de herramientas para caracterizar el comportamiento de los métodos aparece, como se ha discutido en la Sección 3.

En particular, las medidas de complejidad presentadas pueden ser usadas para realizar la caracterización de los métodos de aprendizaje. Una aproximación directa sería analizar la relación entre los valores de las medidas de complejidad para una base de datos, y el rendimiento obtenido por el algoritmo de aprendizaje.

4.2. Construcción de los dominios de competencia

Hemos evaluado C4.5 sobre un conjunto de 438 problemas de clasificación binarios. Hemos restringido nuestra investigación a problemas de dos clases debido a que la mayoría de las medidas de complejidad sólo están bien definidas para este tipo de problemas. Estas bases de datos son generadas a partir de la combinación por parejas de las clases de 21 problemas del repositorio de la Universidad de California, Irvine (UCI) [2]. En particular son *iris, wine, new-thyroid, solar-flare, led7digit, zoo, yeast, tae, balanced, car, contraceptive, ecoli, hayes-roth, shuttle, australian, pima, monks, bupa, glass, haberman* y *vehicle*. Para cada base de datos se calculan los valores de las cinco medidas de separabilidad de clases consideradas. Si una base de datos obtenida mediante este procedimiento resulta ser linealmente separable (la medida de complejidad L1 de [13] indica si un problema es linealmente separable) es

descartado, dado que podría ser tratado con un clasificador lineal sin error alguno.

Para estimar la precisión del método de aprendizaje hemos empleado un esquema de validación 10-*fold cross validation* una vez que las medidas se han calculado sobre la base de datos completa. Tomamos la precisión media sobre las 10 particiones como una medida representativa del rendimiento de C4.5. Hemos considerado los parámetros recomendados para C4.5 empleando la herramienta KEEL[1] [1], que son los que siguen:

- nivel de confianza = 0.25
- número mínimo de items por hoja = 2
- poda del árbol = si

En el Cuadro 2 hemos resumido la precisión media global en entrenamiento y test obtenida por C4.5 para las 438 bases de datos, que serán usadas más adelante como referencia.

Cuadro 2: Precisión media y desviación típica global para C4.5 en entrenamiento y test

	% Global precisión entrenamiento desv. típica global entrenamiento	% Global precisión test desv. típica global test
C4.5	96.29 % 4.44	90.85 % 9.87

Para cada medida de separabilidad de clases, las bases de datos se ordenan por el valor de la medida, permitiendo extraer intervalos de valores de dicha medida. Definimos intervalos de buen y mal comportamiento.

- Entendemos por *buen comportamiento* una precisión en test alta (al menos 80 %) en el intervalo, así como la ausencia de sobre-aprendizaje.
- Por *mal comportamiento* nos referimos a la presencia de sobre-aprendizaje y/o una precisión en test baja en el intervalo.

Estos intervalos pueden ser traducidos a reglas, que emplean los intervalos como antecedentes para definir los dominios de competencia de C4.5.

[1] `http:\\keel.es`

5. Dominios de competencia de C4.5

En esta sección se realiza la extracción y análisis de los dominios de competencia de C4.5 basados en los resultados experimentales usando las medidas de solapamiento entre clases. En la Subsección 5.1 se analizan los intervalos extraídos y sus reglas derivadas. En la Subsección 5.2 se presenta la unión colectiva de las reglas y los dominios de competencia.

5.1. Extracción de las reglas e intervalos

Para cada medida de complejidad considerada (L1, L2, N1, N2 y N3), las bases de datos se ordenan por el valor ascendente de la medida de complejidad correspondiente, y se representan en una figura. En este caso, las bases de datos en el eje X están distribuidas uniformemente, de forma que cada base de datos tenga el mismo espacio en la representación gráfica. Para aquellas medidas donde se pueden encontrar diferentes intervalos *ad-hoc* que presentan el *buen* o *mal comportamiento* de C4.5, usamos una línea vertical para delimitar la región de interés. En el Cuadro 3 indicamos los intervalos obtenidos a partir de las Figuras 2 a 6.

Cuadro 3: Intervalos significativos extraídos

Interval	C4.5 Behavior
$N1 \leq 0{,}167$	*buen comportamiento*
$N2 \leq 0{,}237$	*buen comportamiento*
$L1 \leq 0{,}389$	*buen comportamiento*
$N1 \geq 0.320$	*mal comportamiento*
$N2 \geq 0{,}627$	*mal comportamiento*
$N3 \geq 0{,}164$	*mal comportamiento*
$L2 \geq 0{,}286$	*mal comportamiento*

A partir de estos intervalos *ad-hoc* construimos una serie de reglas que modelan el rendimiento de C4.5. En el Cuadro 4 se muestran las reglas derivadas del Cuadro 3. Dada una base de datos X, se indica el valor de la medida de complejidad CM para X con la notación $CM[X]$. El Cuadro 4 está organizado con las siguientes columnas:

- La primera columna corresponde al identificador de la regla.

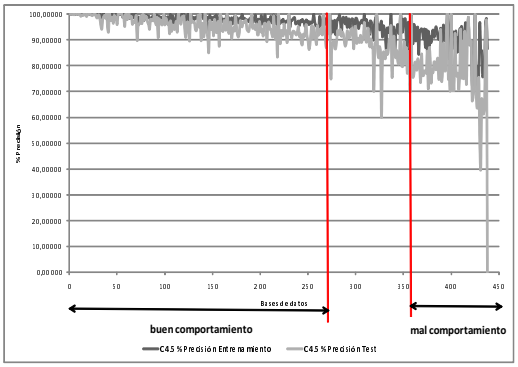

Figura 2: Precisión en entrenamiento/test para C4.5 ordenado por N1

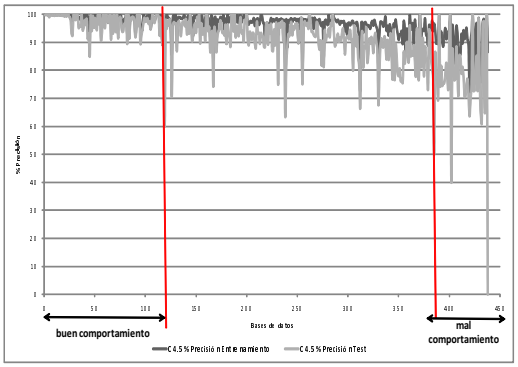

Figura 3: Precisión en entrenamiento/test para C4.5 ordenado por N2

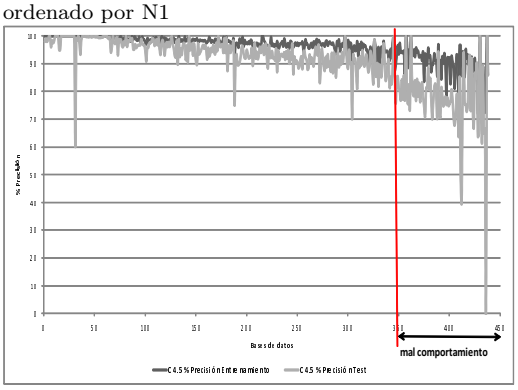

Figura 4: Precisión en entrenamiento/test para C4.5 ordenado por N3

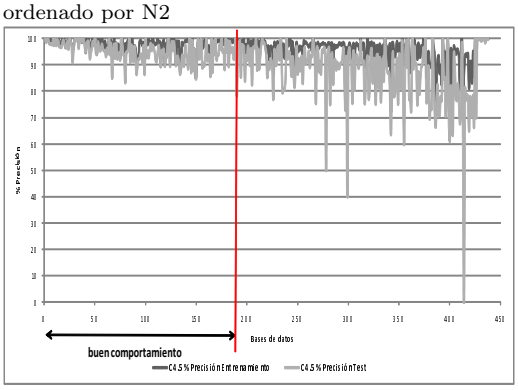

Figura 5: Precisión en entrenamiento/test para C4.5 ordenado por L1

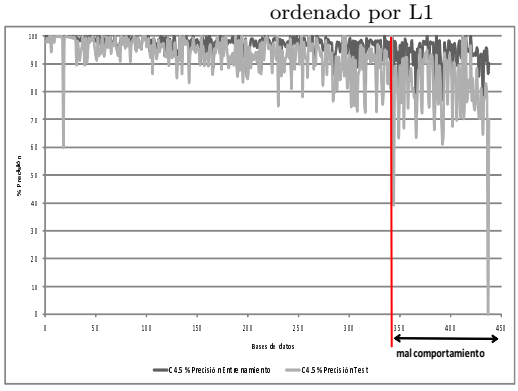

Figura 6: Precisión en entrenamiento/test para C4.5 ordenado por L2

Cuadro 4: Reglas con una medida obtenidas de los intervalos

Id.	Rango	% Soporte	% Entrenamiento	Dif. Entrenamiento	% Test	Dif. Test
R1+	If N1[X] $\leq$ 0.167 then *buen comportamiento*	61.87 %	98.37 %	2.08 %	95.68 %	4.88 %
R2+	If N2[X] $\leq$ 0.237 then *buen comportamiento*	26.94 %	96.63 %	0.34 %	97.30 %	6.45 %
R3+	If L1[X] $\leq$ 0.389 then *buen comportamiento*	43.15 %	97.71 %	1.42 %	95.33 %	4.48 %
R1-	If N1[X] $\geq$ 0.320 then *mal comportamiento*	18.72 %	90.23 %	-6.06 %	78.00 %	-12.85 %
R2-	If N2[X] $\geq$ 0.627 then *mal comportamiento*	12.33 %	90.67 %	-5.62 %	76.55 %	-14.30 %
R3-	If N3[X] $\geq$ 0.164 then *mal comportamiento*	21.00 %	90.47 %	-5.82 %	77.84 %	-13.01 %
R4-	If L2[X] $\geq$ 0.286 then *mal comportamiento*	22.15 %	92.98 %	-3.31 %	80.67 %	-10.18 %

Cuadro 5: Reglas colectivas para C4.5

Id.	Rango	% Soporte	% Entrenamiento	Dif. Entrenamiento	% Test	Dif. Test
URP	If R1+ or R2+ or R3+ then *buen comportamiento*	69.41	97.87	1.58	95.12	4.27
URN	If R1- or R2- or R3- or R4- then *mal comportamiento*	33.79	92.71	-3.58	82.70	-8.15
URP $\wedge$URN	If URP and URN then *buen comportamiento*	8.22	94.70	.1.59	91.60	0.75
URP $\wedge\neg$ URN	If URP and not URN then *buen comportamiento*	61.19	98.29	2.00	95.59	4.74
URN$\wedge\neg$URP	If URN and not URP then *mal comportamiento*	25.57	92.07	-1.22	79.83	-11.02
not characterized	If not (URP or URN) then *mal comportamiento*	5.02	96.01	-0.28	87.91	-2.94

- La columna Rango indica el dominio de la regla.
- La columna Soporte indica el porcentaje de bases de datos cubiertas del total.
- La columna % Entrenamiento indica el porcentaje medio de precisión en entrenamiento de C4.5 en las bases de datos cubiertas por la regla.
- La columna Dif. Entrenamiento muestra la diferencia entre el % Entrenamiento y el porcentaje medio de entrenamiento global de C4.5.
- La columna % Test indica el porcentaje medio de precisión en test de C4.5 en las bases de datos cubiertas por la regla.
- La columna Dif. Test muestra la diferencia entre el % Test y el porcentaje medio de test global de C4.5.

Como se puede observar en el Cuadro 4, las reglas positivas (denotadas con un símbolo "+" en su identificador) siempre muestran una diferencia positiva en precisión con la media global, tanto en entrenamiento como en test. Las reglas negativas (indicadas con símbolo "-" en su identificador) verifican el caso opuesto. El soporte de las reglas muestra que es posible caracterizar un amplio rango de bases de datos con diferencias significativas en precisión.

A partir de este conjunto de reglas se puede afirmar que valores bajos en las medidas N1, N2 y L1 indican un buen comportamiento de C4.5. Por otro lado, un valor alto de N1, N2,

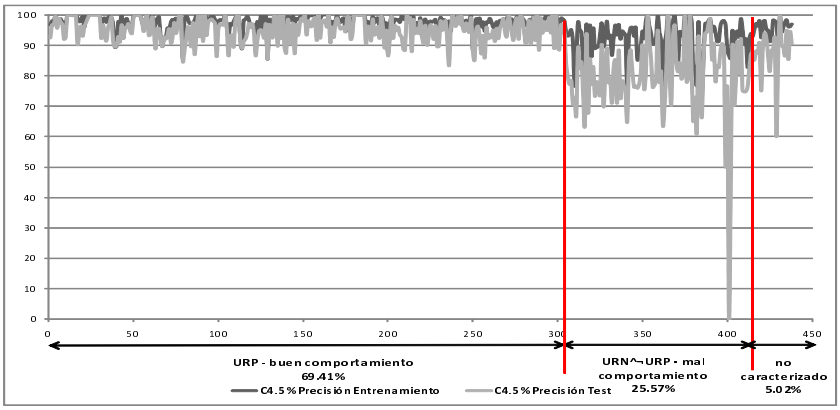

Figura 7: Representación en tres bloques para C4.5: URP, URN∧¬URP y bases de datos no caracterizadas

N3 y L2 se relaciona con el mal comportamiento de C4.5.

5.2. Evaluación conjunta de las reglas simples

El objetivo de esta sección es analizar el efecto de la combinación de las reglas. Hemos considerado la disyunción de todas las reglas positivas para obtener una única regla (Unión de Reglas Positivas -URP-). Esto es, empleamos el operador *or* para combinar las reglas positivas individuales. El mismo procedimiento es realizado con todas las reglas negativas, de manera que se obtiene otra regla (Unión de Reglas Negativas -URN-). Las nuevas reglas disyuntivas se activarán si cualquiera de las reglas que las componen se verifican. Gracias a la unión de las reglas individuales podemos alcanzar una descripción más general y con mayor soporte del comportamiento de C4.5.

Las reglas URP y URN pueden presentar solapamiento en su soporte, y sería deseable una descripción mutuamente exclusiva de las regiones buenas y malas. Para afrontar esta cuestión consideramos la conjunción *and* y la diferencia *and not* entre las reglas URP y URN. La diferencia eliminará aquellas bases de datos para las cuales C4.5 presenta buen o mal comportamiento de la regla URN o URP respectivamente. Por tanto se obtienen tres tipos de intersecciones y una región extra:

- Intersección de la disyunción positiva y la disyunción negativa (URP∧URN).
- Diferencia de la disyunción positiva y la disyunción negativa (URP∧¬URN).
- Diferencia de la disyunción negativa y la disyunción positiva (URN∧¬URP).
- Región *no caracterizada*, en la cual ninguna base de datos está cubierta por ninguna regla.

Todas estas nuevas reglas se encuentran representadas en el Cuadro 5.

Considerando las nuevas reglas disyuntivas y conjuntivas, podemos presentar a URP como una descripción representativa de los dominios de competencia de C4.5 para las buenas bases de datos. De forma complementaria URN∧¬URP puede considerarse como una descripción representativa de los dominios de competencia de C4.5 para las bases de datos malas, siendo ambas descripciones mutuamente exclusivas entre sí. Podemos considerar entonces tres bloques de bases de datos con su respectivo soporte, como se muestra en la Figura 7 (sin ningún orden particular de las bases de datos en cada bloque):

- El primer bloque (a la izquierda) representa las bases de datos cubiertas por la regla URP. Son las bases de datos incluidas en el dominio de competencia de buen comportamiento para C4.5.

- El segundo bloque (en el centro) muestra las bases de datos para la regla URN$\wedge\neg$URP, los cuales son aquellas cubiertas por el dominio de competencia de mal comportamiento de C4.5.
- El tercer y último bloque (a la derecha) contiene las bases de datos no clasificadas por ninguno de los dominios previos.

Aproximadamente el 95 % de las bases de datos analizadas son cubiertas por estas dos reglas. Por tanto es posible definir bien los dominios de competencia de C4.5 empleando las medidas de separabilidad de clases.

6. Conclusiones

En la literatura se han propuesto varios marcos de trabajo para analizar el error de generalización desde un punto de vista teórico, desde el aprendizaje PAC [15] al aprendizaje estadístico y no paramétrico [23].

En esta contribución hemos realizado una aproximación empírica, proponiendo una metodología para extraer los dominios de competencia de un algoritmo de aprendizaje empleando medidas de complejidad de datos. Se ha realizado un estudio sobre un conjunto de bases de datos binarias con el método de aprendizaje C4.5, obteniendo intervalos de las medidas de separabilidad de clases que se traducen a reglas. Estas reglas muestran información acerca de las limitaciones de C4.5, así como la posibilidad de estimar su rendimiento *a priori*.

Como resultado final, hemos obtenido dos reglas que son simples y precisas para describir los dominios de competencia de C4.5, presentando la posibilidad de determinar para que bases de datos C4.5 funcionaría bien o mal de manera previa a su ejecución.

Este trabajo abre nuevas perspectivas que pueden extenderse a otros métodos de aprendizaje, para analizar sus dominios de competencia y desarrollar nuevas medidas que puede dar más información sobre el comportamiento de los clasificadores para el reconocimiento de patrones.

Como trabajo futuro sería necesario considerar el uso de otras medidas de rendimiento del método de aprendizaje. La precisión clásica puede verse afectada por el no balanceo de las clases, aspecto que no ha sido considerado en esta contribución. Es necesario también realizar una validación de los dominios de competencia obtenidos empleando bases de datos no utilizadas en el propio proceso de construcción de los dominios.

Agradecimientos

Este trabajo de investigación ha sido posible gracias a la subvención del proyecto del Ministerio de Ciencia e Innovación TIN2008-06681-C06-01. J. Luengo está subvencionado por una beca FPU del Ministerio de Ciencia e Innovación.

Referencias

[1] J. Alcalá-Fdez, L. Sánchez, S. García, M.J. del Jesus, S. Ventura, J.M. Garrell, J. Otero, C. Romero, J. Bacardit, V.M. Rivas, J.C. Fernández, and F. Herrera. KEEL: A software tool to assess evolutionary algorithms to data mining problems. *Soft Computing*, 13(3):307–318, 2009.

[2] A. Asuncion and D.J. Newman. UCI machine learning repository, 2007.

[3] N. Baskiotis and M. Sebag. C4.5 competence map: a phase transition-inspired approach. In *ICML '04: Proceedings of the twenty-first international conference on Machine learning*, page 8, New York, NY, USA, 2004. ACM.

[4] M. Basu and T. K. Ho. *Data Complexity in Pattern Recognition (Advanced Information and Knowledge Processing)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[5] R. Baumgartner and R. L. Somorjai. Data complexity assessment in undersampled classification of high-dimensional biomedical data. *Pattern Recognition Letters*, 12:1383–1389, 2006.

[6] H. Bensusan and A. Kalousis. Estimating the predictive accuracy of a classifier. In *EMCL '01: Proceedings of the 12th European Conference on Machine Learning*, pages 25–36, London, UK, 2001. Springer-Verlag.

[7] E. Bernadó-Mansilla and T. K. Ho. Domain of competence of XCS classifier system in complexity measurement space. *IEEE Transactions on Evolutionary Computation*, 9(1):82–104, 2005.

[8] D. G. Bobrow, T. Hogg, B. A. Huberman, and C. P. Williams, editors. *Special volume on frontiers in problem solving: phase transitions and complexity*, volume 81. Elsevier Science Publishers Ltd., Essex, UK, 1996.

[9] M. Botta, A. Giordana, L. Saitta, and M. Sebag. Relational learning as search in a critical region. *Journal of Machine Learning Research*, 4:431–463, 2003.

[10] P. Brazdil, C. Giraud-Carrier, C. Soares, and R. Vilalta. *Metalearning: Applications to Data Mining*. Cognitive Technologies. Springer, January 2009.

[11] P. Cheeseman, B. Kanefsky, and W. M. Taylor. Where the really hard problems are. In *IJCAI'91: Proceedings of the 12th international joint conference on Artificial intelligence*, pages 331–337, San Francisco, CA, USA, 1991. Morgan Kaufmann Publishers Inc.

[12] S. García, J. R. Cano, E. Bernadó-Mansilla, and F. Herrera. Diagnose of effective evolutionary prototype selection using an overlapping measure. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(8):2378–2398, 2009.

[13] T. K. Ho and M. Basu. Complexity measures of supervised classification problems. *IEEE Transactions on Pattern Analysins and Machine Intelligence*, 24(3):289–300, 2002.

[14] A. Kalousis. *Algorithm selection via meta-learning*. PhD thesis, Université de Geneve, 2002.

[15] M. J. Kearns and U. V. Vazirani. *An introduction to computational learning theory*. MIT Press, Cambridge, MA, USA, 1994.

[16] T. Lim, W. Loh, and Y. Shih. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. *Machine Learning*, 40(3):203–228, 2000.

[17] J. Luengo and F. Herrera. Domains of competence of fuzzy rule based classification systems with data complexity measures: A case of study using a fuzzy hybrid genetic based machine learning method. *Fuzzy Sets and Systems*, 161(1):3–19, 2010.

[18] B. Pfahringer, H. Bensusan, and C. G. Giraud-Carrier. Meta-learning by landmarking various learning algorithms. In *ICML '00: Proceedings of the Seventeenth International Conference on Machine Learning*, pages 743–750, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[19] U. Rückert, S. Kramer, and L. D. Raedt. Phase transitions and stochastic local search in k-term dnf learning. In *ECML '02: Proceedings of the 13th European Conference on Machine Learning*, pages 405–417, London, UK, 2002. Springer-Verlag.

[20] J. S. Sánchez, R. A. Mollineda, and J. M. Sotoca. An analysis of how training data complexity affects the nearest neighbor classifiers. *Pattern Analysis & Applications*, 10(3):189–201, 2007.

[21] F. W. Smith. Pattern classifier design by linear programming. *IEEE Transactions on Computers*, 17(4):367–372, 1968.

[22] K. Toh. An error-counting network for pattern classification. *Neurocomputing*, 71(7-9):1680–1693, 2008.

[23] V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.

[24] D. H. Wolpert. The lack of a priori distinctions between learning algorithms. *Neural Computation*, 8(7):1341–1390, 1996.