DEPARTAMENTO DE LENGUAJES Y SISTEMAS INFORMÁTICOS

Técnicas de Submuestreo, Toma de Decisiones y Análisis de Diversidad en Aprendizaje Supervisado con Sistemas Múltiples de Clasificación

TESIS DOCTORAL

PRESENTA
Rosa María Valdovinos Rosas

DIRECTOR
José Salvador Sánchez Garreta

Castelló de la Plana, Junio de 2006

Muchas veces, una oportunidad es lo que necesitamos para demostrar a nosotros mismos y al mundo entero que tenemos la capacidad de alcanzar aquello que sólo en sueños pudimos imaginar… los gigantes se ven grandes porque nosotros estamos de rodillas.

A

Ricardo Barandela Alonso†

Con cariño, gratitud, admiración y respeto a quien es y seguirá siendo un ejemplo a seguir. Esto es parte del fruto de sus enseñanzas.

Los grandes hombres nunca mueren, pero cuando se van dejan un gran vacío.

Agradecimientos

Cualesquiera que hayan sido nuestros logros,
alguien nos ayudó siempre a alcanzarlos.

Althea Gibson

Esta Tesis no es más que el resultado del recorrido de un arduo camino, en él se cruzaron personas que en diferente momento me proporcionaron alicientes para hacerlo menos difícil, a ellas debo agradecer una sonrisa amigable en momentos de tristeza, una palabra de aliento en momentos de crisis, una crítica cuando mis objetivos se desviaban, un hombro para apoyarme en momentos de flaqueza y, un minuto de silencio cuando necesite ser escuchada. A todas ustedes expreso mi gratitud… este logro no es solo mío, es de ustedes también.

Como parte de mi fe, deseo iniciar agradeciendo al creador haberme puesto en el lugar correcto con las personas adecuadas, por haberme dado la sabiduría para tomar la decisión más acertada en cada momento de conflicto y por proporcionarme el coraje y la fortaleza necesaria para enfrentar todas y cada una de las muchas dificultades emocionales, económicas, familiares, laborales y de toda naturaleza por las que en más de una ocasión mis deseos de superación se vieron amenazados.

Mis padres Buenaventura Valdovinos y Valeriana Rosas son seres a los que infinitamente estaré agradecida por haber entendido y apoyado mi decisión, aún cuando esto implicó vivir a un mar de distancia y alejarme en el momento quizá más difícil para nuestra familia. Sus oraciones jamás me han abandonado… espero se sientan orgullosos de mí.

Del mismo modo, agradezco a todos mis hermanos que a su manera cada uno me ha motivado a continuar buscando mis objetivos. Uno de mis mayores agradecimientos para Soco, quien a lo largo de estos años ha seguido paso a paso la metamorfosis de un sueño a una realidad, y que ha vivido como propios los buenos y malos momentos que he pasado… este logro también es tuyo.

A ti, que a distancia has vivido en todas sus facetas este ascenso y que de una u otra forma me has motivado a seguir adelante… no tengo mas que decirte, gracias.

Agradezco también a todas mis amigas mexicanas que de una manera u otra me han demostrado su total apoyo en la distancia: Mónica Prado, Elsa Arzate, Luisa Sánchez, Angélica Álvarez y Citlalli Sánchez. De igual forma, quiero agradecer a los amigos que la vida me ha dado oportunidad de encontrar en el difícil camino del doctorando extranjero y con quienes he compartido tantas y tantas noches de pláticas felices e infelices en las que la distancia con nuestro país nos unió en una bonita hermandad y en las que la convivencia diaria nos permitió conocernos, cuidarnos y apoyarnos en los momentos de flaqueza para seguir adelante con nuestros sueños: Soledad Deferrari, Dioclecio Camelo, Patricio Gonzáles, Vicente García, Roberto Alejo, Luis Puig, Marcos Anicete, Dulce Montoya, Karenina Valdez y Vanesa Díaz. Gracias por brindarme la oportunidad de conocerlos, aún cuando con algunos de ustedes el tiempo se reduce a unos cuantos meses, es grato ver que nuestras diferencias culturales no fueron obstáculo para que entre nosotros nacieran sentimientos tan puros que nos dejan el deseo de demostrar que dos puntos en el espacio se pueden volver a juntar.

De entre este círculo de amigos, deseo resaltar a un gran ser humano: Graciela Bernal. Amiga incondicional, compañera de viajes y dietas, cómplice de mis locuras y hermana de sueños, deseos, luchas, tristezas y alegrías, con esa sobriedad que te caracteriza siempre llegaste a tiempo en los momentos que más te necesité… también para mí este camino fue más fácil de recorrer después de haberte conocido.

En el ámbito profesional, dos personas han sido fundamentales para el inicio, desarrollo y culminación de esta etapa: el Dr. Ricardo Barandela Alonso y el Dr. José Salvador Sánchez Garreta. El origen de este sueño se remonta a mis estudios de Maestría, donde el Dr. Ricardo Barandela transmitió con su ejemplo principios tan valiosos como la tenacidad, la responsabilidad, el respeto, la dignidad y el poder de decisión, sembrando de esta manera la semilla de un sueño acariciado celosamente durante los últimos cinco años. Desafortunadamente, el tiempo no fue suficiente para que su dirección me acompañase hasta el final, con lo que después del dolor de su perdida, la continuidad de mis estudios se vio seriamente amenazada.

Pero, sembrar la semilla es sólo una fase en el proceso general del crecimiento profesional. El vínculo establecido con el Dr. Sánchez ha sido clave para que esa semilla sembrada poco más de tres años atrás lograra germinar en la presente Tesis. Él ha sido el encargado de alimentar, motivar, proporcionar la enseñanza faltante y señalar de forma paciente mis deficiencias, para madurar los conocimientos adquiridos con anterioridad, abriendo la posibilidad de culminar este trabajo de manera exitosa. Infinitamente estaré agradecida con usted que depositó toda su confianza en mí, empeñando su nombre para respaldar un trabajo iniciado sin importar los riesgos que todo esto conllevaba. Espero en un futuro cercano pueda confirmar que sus lecciones de vida son materializadas con actos.

El desarrollo de esta Tesis también contó con el apoyo del Instituto Tecnologico de Toluca y con la motivación de muchos de sus profesores, sus enseñanzas me proporcionaron herramientas útiles para enfrentar este reto, en especial Eduardo Gasca, Erendira Rendón, Norma Calderón y Rafael Cruz. Sus enseñanzas no sólo se dieron en el aula. Gracias por su confianza, su respaldo y su calidad humana.

También es importante reconocer el apoyo prestado por los investigadores que forman el *Grupo de Visión por Ordenador* de esta *Universitat Jaume I*, sin el cual hubiese resultado mucho más difícil llegar a la conclusión de esta Tesis. De la misma manera, deseo agradecer a los doctores del *Departamento de Lenguajes y Sistemas Informáticos* por los conocimientos impartidos en los diferentes cursos de Doctorado, participando de este modo en mi formación como investigadora.

Por último, pero no menos importante deseo agradecer a las diferentes instituciones que económicamente me han apoyado para realizar los estudios de Doctorado: CosNET, DEGEST, Alβan, CICYT y Universitat Jaime I.

Si lo que has conseguido es verdaderamente lo que buscabas, entonces esto es un logro.

Rosa María Valdovinos Rosas
Castellón de la Plana, Junio 2006

Resumen

En la actualidad, los Sistemas Múltiples de Clasificación (SMC) se han consolidado como una fuerte línea de investigación en reconocimiento de patrones. Entre otros aspectos, el éxito de estos sistemas radica en la combinación de varias decisiones individuales para clasificar un mismo patrón de prueba, en lugar de utilizar un clasificador único. Para este fin, es fundamental que el SMC cumpla con dos condiciones básicas: ser diverso y estar formado por elementos suficientemente precisos.

Sin embargo, la responsabilidad de los resultados en la clasificación no es sólo responsabilidad del SMC, la calidad del conjunto de entrenamiento también influye significativamente en su desempeño. Algunos de los principales factores que deterioran la calidad de la clasificación y que están directamente relacionados con el conjunto de datos son: patrones redundantes, atípicos o ruidosos, bases de datos con tamaño excesivo, y desbalance entre las distribuciones de las clases.

En la presente Tesis Doctoral, se desarrollan diversas metodologías orientadas a extender el uso de SMC al ser utilizados como parte de la solución a estos problemas. Para ello, se construyen SMC formados mediante submuestreo de patrones con el mismo grado de representatividad entre clases que el conjunto de datos original, en los cuales la fusión de clasificadores se realiza con dos enfoques: votación por mayoría simple y votación por mayoría ponderada. En este último enfoque, se desarrollan siete diferentes esquemas de ponderación, seis con ponderación dinámica y uno con ponderación estática.

Por otro lado, para disminuir y, en algunos casos, eliminar los aspectos negativos con los que el conjunto de datos cuenta, se utilizan, de forma conjunta, métodos propios de los SMC y diversos algoritmos de preprocesado. Específicamente, para la limpieza del conjunto de datos, se emplea la edición de Wilson, para la reducción del tamaño del conjunto de entrenamiento, se aplica el algoritmo de subconjunto selectivo modificado, y para la generación de patrones sintéticos, se utiliza el algoritmo SMOTE y la eliminación de patrones redundantes.

Para la clasificación de nuevos patrones, se acude a dos reglas de decisión ampliamente estudiadas en reconocimiento de patrones: la regla del vecino más cercano y redes neuronales artificiales. En esta última regla, se implementan dos diferentes estructuras de redes neuronales: perceptrón multicapa y red modular. Por su parte, para la experimentación, se utilizan 17 bases de datos de problemas reales con características muy variadas entre sí, como balance entre clases, cantidad de patrones, dimensionalidad y número de clases, entre otras.

Los resultados experimentales nos permiten evidenciar lo viable y ventajoso que las metodologías propuestas resultan ser como parte de solución a los problemas antes mencionados. Las aportaciones realizadas pueden describirse en función de su impacto de la siguiente manera:

a) Evidenciar como, al utilizar conjuntamente SMC y algoritmos de preprocesado, es posible eliminar (en la mayoría de los casos) los efectos negativos que el desbalance tiene sobre los índices de precisión.
b) Proponer la escalabilidad de algoritmos como alternativa de solución para conjuntos de datos con tamaño excesivo, con una precisión igual o superior a la obtenida cuando se trabaja con el conjunto de datos de forma clásica.
c) Demostrar el alto rendimiento (en términos de precisión) que se tiene cuando se combinan métodos de SMC y la edición de Wilson.
d) Presentar evidencia suficiente que permite demostrar que no siempre los más altos índices de diversidad en las decisiones favorecen los mejores niveles de precisión.
e) Con base al desempeño obtenido con los algoritmos de redes neuronales utilizados, es posible justificar la utilización de la regla 1-NN como algoritmo de clasificación en SMC.
f) Mostrar el buen desempeño que los métodos propuestos de ponderación dinámica tienen sobre la ponderación estática y la votación simple en la fusión de clasificadores.
g) Demostrar la viabilidad de utilizar los métodos de submuestreo selección secuencial, selección aleatoria sin reemplazo y Bagging, para obtener diversidad en un SMC, con resultados satisfactorios.

h) Establecer como número suficiente de miembros en un SMC 5, 7 y 9 clasificadores.

i) Comprobar que la creación de submuestras que mantienen un mismo grado de representatividad entre las clases igual a la del conjunto de datos original, favorece el desempeño de los SMC, en comparación con submuestras formadas sin considerar esta representatividad.

j) Validar la posibilidad de construir redes neuronales con el mínimo de neuronas en la capa oculta sin deteriorar los índices de precisión.

Abstract

In the present, Multiple Classifier Systems (MCS) have become an important research line in Pattern Recognition. Among other aspects, the success of these systems is due to the increase in performance derived from the combination of a set of individual decisions, instead of using a unique classifier. To this end, the MCS has to satisfy two basic conditions: it should be as diverse as possible and it should consist of sufficiently accurate elements.

Nevertheless, the classification results do not depend only on the MCS, but also on the quality of the training set. Some of the main factors that can deteriorate the classification quality and are strongly related to the data sets are: redundant, atypical, and noisy patterns, huge databases, and imbalance in class distributions.

In the present Ph.D. Thesis, we develop several methodologies that are directed to extend the use of the MCS as a part of the solution to those problems. Thus we generate MCS by resampling patterns in such a way that the resulting subsamples keep the a priori class distribution present in the original data set. In this case, the classifier fusion is performed by means of two approaches: simple majority voting and weighted majority voting. In this second approach, seven different weighting schemes are proposed, six of them with dynamic weighting and one with static weighting.

On the other hand, in order to reduce and in some cases, even eliminate the negative aspects of the data set, we combined several MCS methods with preprocessing algorithms. Specifically, for cleaning the data set we employed

Wilson editing, for reducing the training set size we applied the modified selective subset algorithm, and for the generation of synthetic patterns we utilized the SMOTE algorithm and the and removal of redundant patterns.

For the classification of new patterns, we used two decision rules widely studied in pattern recognition: the nearest neighbour rule and the neural networks. With respect to the latter, two distinct types of neural structures were explored: the multilayer perceptron and the modular network. In the experimentation, we employed 17 real-problem databases with very different characteristics such as class imbalance, number of patterns, dimensionality, and number of classes, among others.

The experimental results demonstrated the feasibility and the advantageous of these methodologies to solve the described problems. The main contributions can be described in the following way:

a) It has been proved that the combination of MCS methods whit preprocessing algorithms allows to eliminate (in most cases) the negative effects of class imbalance on the accuracy rates.
b) It has been proposed the scalability of algorithms as an alternative to process huge databases, with accuracy levels equal to or superior than those obtained when the data set is used in a classical way.
c) It has been demonstrated the high performance (in terms of accuracy) given when MCS and Wilson editing are combined.
d) It has been showed that not always the highest diversity in the decisions means the highest accuracy rates.
e) Based on the performance obtained with the neural networks, it can be justified the use of the 1-NN rule as the classifier algorithm within the MCS.
f) It has been proved the good performance of the dynamic weighting methods here proposed when compared to the static weighting and the simple voting for the classifier fusion.
g) It has been demonstrated the feasibility of using the resampling methods based on sequential selection, random selection without replacement and Bagging, to obtain diversity in a MCS with satisfactory results.
h) It has been established that the sufficient number of members in a MCS is 5, 7 and 9 classifiers.
i) It has been showed that the resampling methods based on keeping the a priori class distributions improves the performance when compared to the traditional resampling strategies.
j) It has been proved that it is possible to build neural networks with minimum number of neurons in the hidden layer without degrading the accuracy rates.

ÍNDICE GENERAL DE CONTENIDO

PARTE I
INTRODUCCIÓN Y FUNDAMENTOS TEÓRICOS

3 Clasificación y técnicas de preprocesado

4 Sistemas múltiples de clasificación

PARTE II
METODOLOGÍA Y RESULTADOS

PARTE III
CONCLUSIONES

APÉNDICES

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

TABLA DE NOMENCLATURAS Y SIGLAS

x	Patrón de entrenamiento
y	Patrón de test o de control
n	Número de características o atributos
M	Conjunto de patrones de entrenamiento
m	Número de patrones
c	Número de categorías o clases
d	Métrica de distancia
p	Número de patrones de test o de control
t	Etiqueta de clase
r	Número de regiones en que se divide el conjuntos de datos
PG	Precisión general $\frac{\sum_{e} x}{p}$: e patrones correctamente clasificados
G	Media geométrica $g = \sqrt{a^{+} \cdot a^{-}}$: a^{+} precisión de la clase menos representada, a^{-} precisión de la clase más representada
DE	Desviación estándar $\hat{\sigma} = \sqrt{\frac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x})^2}$
CK	Coeficiente kappa
VK	Varianza kappa
k	Número de vecinos a encontrar por la regla *k*-NN
D	Conjunto de clasificadores en un SMC
H	Número de clasificadores que forman un SMC
sinR1	Selección secuencial
sinRA1	Selección aleatoria sin reemplazo
SMC	Sistemas de Múltiple Clasificación
CE	Conjunto de Entrenamiento
NNR	Nearest Neighbor Rule
RNA	Redes Neuronales Artificiales
PM	Perceptrón Múlticapa
SMOTE	Synthetic Minority Over-sampling Technique
SSM	Subconjunto Selectivo Modificado
CT	Conjunto de Test o prueba

PARTE I

INTRODUCCIÓN Y FUNDAMENTOS TEÓRICOS

Capítulo 1

Introducción

La tarea del reconocimiento de patrones involucra la habilidad de decidir con precisión y eficiencia la asignación de etiquetas a objetos, imágenes, sonidos, pensamientos o ideas. Cada vez más, el uso de estos sistemas ha reemplazado las actividades manuales del hombre, contribuyendo a simplificar muchas de sus tareas diarias.

Las áreas de aplicación del reconocimiento de patrones son muy diversas, pero quizás donde más intensamente ha sido usado es en el área médica, de manera concreta, en el diagnóstico temprano de problemas de salud [Ort., 96]. Como ejemplo de ello, es posible mencionar los sistemas automáticos para detección de isquemia cardiaca [Pap., 01], diagnóstico de afecciones óseas y oculares, detección temprana de cáncer mamario [Woo., 93], análisis de imágenes provenientes de tomografía axial computerizada, estudio y clasificación de cromosomas, reconocimiento de genes en secuencias de ADN, análisis prenatal, diagnósticos de síndromes genéticos, análisis de trazos electrocardiográficos y de encefalogramas [Mic., 00], etc. Sin embargo, también se ha aplicado a otras áreas, como el procesamiento de imágenes satelitales, comunicación acústica con computadoras, reconocimiento de caracteres escritos, reconocimiento de rostros, control de robots que utilizan información visual y vehículos autónomos [Par., 96], lectura de

direcciones postales, sistemas de reconocimiento de voz, desarrollo de diarios y agendas electrónicas, reconocimiento de patrones en procesos sísmicos [Die., 98], entre otras muchas. La asistencia inteligente de estos sistemas durante la toma de decisiones resulta ser de vital importancia, pues de ella depende en gran medida el éxito o fracaso de los procesos.

Debido a la inminente necesidad de contar con sistemas informáticos eficientes, en las últimas cuatro décadas, se ha tenido un fuerte desarrollo en un conjunto de métodos aplicados al logro del proceso de aprendizaje automático eficiente. De estas investigaciones destaca una técnica comúnmente conocida como "ensembles" o Sistemas Múltiples de Clasificación (SMC), que pueden definirse como "el empleo de un grupo de clasificadores, cuyas decisiones individuales son combinadas para clasificar un nuevo patrón" [Die., 97]. De esta forma, $D = \{ D_1,..., D_H\}$ es un grupo de clasificadores. Cada clasificador asigna un patrón de entrada **y** a una de las c clases disponibles. La salida del SMC es un vector H-dimensional que contiene las decisiones de cada uno de los H clasificadores individuales $[D_1(\mathbf{x}),..., D_H(\mathbf{x})]^T$.

Para combinar las decisiones individuales, se distinguen dos estrategias: selección y fusión. En la selección de clasificadores, cada clasificador individual es considerado experto sobre una parte del espacio de representación, por lo que, para asignar la etiqueta de clase que corresponde a **y**, se selecciona un sólo clasificador. Por otro lado, la fusión de clasificadores asume que todos los componentes individuales del SMC son competitivos, en vez de complementarios, es decir, todos son igualmente expertos en todo el espacio de representación, de tal manera que todos deciden qué etiqueta de clase corresponde a **y**.

Por múltiples razones, actualmente estos sistemas se han consolidado como una línea bien establecida de estudio en reconocimiento de patrones. Esto obedece en parte, a varios aspectos en los que los SMC superan al desempeño que hasta ahora se ha observado cuando se utiliza un clasificador único [Kun., 01c]: la decisión combinada toma ventaja sobre la decisión de cada clasificador individual, los errores correlacionados de los componentes individuales pueden ser eliminados cuando se considera el total de las decisiones y, finalmente, el espacio individual de búsqueda puede no contener la función objetivo. Para esto, existen dos condiciones necesarias y fundamentales que un SMC debe satisfacer [Kun., 02a]: que exista diversidad de las decisiones y que el ratio de error de los componentes individuales (principalmente cuando se utiliza el esquema de votación simple) sea superior a 0.5.

Se dice que dos clasificadores son diversos si arrojan decisiones diferentes al clasificar un mismo patrón de entrada. Para que esta diversidad exista, se han propuesto múltiples y muy variados métodos. Algunos incluyen la diversidad entre los algoritmos de clasificación [Bah., 00], [Kol., 91], [Rav., 96], [Ali., 96], y otros la diversidad en el conjunto de entrenamiento (CE) utilizado [Ho., 92], [Bre., 96],

[Kun., 05], [Fre., 96], [Bre., 98], [Par., 96], [Bar., 03a]. Por otro lado, en lo que respecta a la precisión de los elementos que forman el SMC, se ha demostrado que si la proporción del error de los componentes individuales excede el 0.5, la precisión del SMC se ve seriamente afectada [Die., 97], pudiendo en algunos casos tener peores resultados el SMC que cualquiera de los clasificadores que lo forman [Mat., 96].

Otros factores que también influyen en los índices de precisión son algunos problemas presentes en el CE [Bar 01a]: patrones con atributos poco discriminantes, presencia de patrones atípicos o ruidosos, CE resultantes cuyo tamaño requiere de una gran cantidad (gigabytes o terabytes) de memoria RAM para su procesamiento en conjunto, y/o la obtención de CE con desbalance en la representatividad de cada una de sus clases. Estos aspectos aún no han sido tratados de manera conjunta con técnicas de combinación de clasificadores, situación que motiva fuertemente la realización del presente trabajo.

1.1 OBJETIVOS DE LA TESIS

La presente tesis tiene como objetivo principal ampliar el conocimiento en la implementación de SMC, evaluando su desempeño cuando se utiliza la regla del vecino más cercano o algún tipo de redes neuronales artificiales como algoritmo de clasificación. Para ello, se desarrollan e implementan metodologías de trabajo que, mediante la utilización de un SMC, proporcionen alternativas de solución a problemas prácticos de reconocimiento de patrones. En particular, se analiza la conveniencia de esta posibilidad cuando se tratan los siguientes casos:

a) Conjuntos de datos que cuenten con desbalance. Para lograr un adecuado tratamiento de este tipo de datos, se propone trabajar de manera conjunta el SMC con algoritmos de preprocesado (reducción, aumento y limpieza) del CE.
b) Escalabilidad de algoritmos. Para conjuntos de datos que por su gran tamaño no pueden ser cargados totalmente en la memoria del equipo de cómputo, se incluye el estudio y desarrollo de metodologías que faciliten su tratamiento de forma eficiente.
c) Aumento de la eficiencia de la regla del vecino más cercano (1-NN). Para disminuir los efectos negativos que problemas inherentes a la regla 1-NN tiene sobre la clasificación, se implementan nuevos criterios para ponderar las decisiones individuales en el SMC cuando se utiliza el esquema de fusión de clasificadores.
d) Análisis de diversidad. Para esta situación, se acudió a métodos ya existentes que aplicados de manera exhaustiva logran determinar el grado de diversidad de las decisiones individuales en los SMC utilizados.
e) Implementación de redes neuronales artificiales. Como parte de este objetivo, se realiza la implementación y prueba de la capacidad

computacional de dos esquemas de redes neuronales artificiales (RNA): el perceptrón múlticapa con aprendizaje de backpropagation y la red modular con aprendizaje de gradiente estocástico, como algoritmos de clasificación en un SMC.

Mediante experimentos con conjuntos de datos reales, se demuestra que las metodologías propuestas amplían el buen desempeño de los SMC, aumentando además, la eficiencia en la clasificación de nuevos patrones que hasta el momento se ha observado con el uso de un sólo clasificador.

1.2 ESTRUCTURA DEL DOCUMENTO

El presente documento de Tesis Doctoral consta de tres partes principales, cada una de ellas integrada a su vez por un conjunto de capítulos. Además, se incluyen también diversos apéndices con información complementaria.

Parte I. En esta parte, se establecen el objetivo general, los alcances y la motivación de la presente investigación, además de proporcionar los principios básicos de reconocimiento de patrones, métricas de similaridad y criterios de evaluación más utilizados. En uno de los capítulos, también se abordan los dos principales paradigmas estudiados para la clasificación de patrones, haciendo una revisión de los principales factores que deterioran la calidad de clasificación relacionados con el CE y se describen las principales estrategias de preprocesado propuestas en la literatura para resolver algunos de esos problemas. Como capítulo medular, se incluye una revisión bibliográfica de los diferentes métodos que han surgido para integrar un SMC, de las formas de realizar la toma de decisiones y de las diferentes técnicas para medir la diversidad en las decisiones individuales.

Parte II. En este apartado, se detallan las diferentes metodologías de solución propuestas para cada uno de los problemas planteados, las bases de datos utilizadas y la parte experimental que incluye una serie de resultados donde pueden apreciarse las mejoras y el cumplimiento de los objetivos trazados.

Parte III. Conclusiones. En esta parte, se presentan las principales conclusiones de la investigación y se resaltan las aportaciones personales. Con base en la evidencia descrita en la Parte II, se determina el cumplimiento de hipótesis y objetivos; además, se ofrecen directrices para futuras investigaciones.

Apéndices. Resultados a detalle. En el Apéndice I, se incluyen de forma detallada el proceso de aprendizaje backpropagation y el funcionamiento de la red modular. En el Apéndice II, se proporcionan los resultados a detalle al aplicar los diferentes métodos para estimar la diversidad en las decisiones. Por último, en los Apéndices III y IV, presentamos las tablas de los resultados obtenidos en la clasificación de forma detallada. En cada una de las tablas, puede observarse la precisión general y

su respectiva desviación estándar. Todas estas tablas son referencia directa de los resultados incluidos en el grueso del documento y forman parte de los experimentos mostrados en la Parte II.

Capítulo 2

Aspectos básicos del Reconocimiento de Patrones

En 1956, se acuñó formalmente el término de Inteligencia Artificial como una ciencia que pretende desarrollar software y hardware que simulen el comportamiento humano; sin embargo, el estudio de la inteligencia contemplada como el razonamiento humano ha sido abordado por los filósofos desde hace más de 2000 años. A lo largo de este tiempo, muchas ciencias han participado de forma activa en el desarrollo exitoso de herramientas de Inteligencia Artificial. Algunas de ellas son las Matemáticas, que nos proporcionan teorías formales relacionadas con la lógica, la probabilidad, la teoría de decisiones y la computación. Otra es la Psicología, cuyas herramientas nos permiten investigar la mente humana, al mismo tiempo que nos proporciona un lenguaje científico para expresar las teorías que se van obteniendo, en tanto que la Lingüística nos ofrece teorías sobre la estructura y el significado del lenguaje.

Una de las áreas de mayor crecimiento es el reconocimiento de patrones, íntimamente relacionado con el Aprendizaje de Maquinas (en inglés, Machine Learning) o Aprendizaje Automático y podemos definirlo como: la rama de la Inteligencia Artificial que estudia la operación y el diseño de sistemas que permitan extraer similaridades y coincidencias de un conjunto de objetos y ayude a establecer propiedades de o entre conjuntos de datos. También podemos definirlo como el acto de tomar datos sin ningún sentido y clasificarlos de acuerdo a una acción basada en las categorías de un patrón dado, previamente analizado [The., 98]. Para lograr su objetivo, el reconocimiento de patrones involucra una gran variedad de subdisciplinas, como el análisis discriminante, la extracción de carácterısticas, la estimación del error, el análisis de regiones, la inferencia gramatical, etc.

2.1 APLICACIONES DEL RECONOCIMIENTO DE PATRONES

El reconocimiento de patrones cubre una vasta área de procesamiento de información que va (sólo por mencionar algunas de ellas) desde el reconocimiento de voz, de escritura, de señales sísmicas o de radar, hasta el diagnóstico de enfermedades o la detección de fallos en maquinaria y procesos industriales, y tienen muy diversas aplicaciones: agrícolas, industriales, biomédicas, astronómicas o en protección civil. A continuación, se enumeran algunos trabajos que, con la ayuda de técnicas y metodologías de reconocimiento de patrones, resuelven algún problema específico.

a) Reconocimiento de caracteres escritos. Para el ser humano es relativamente sencillo reconocer los patrones de la escritura manuscrita, mientras que esta tarea es sumamente difícil para la computadora. Existen programas que tratan de lograr este objetivo; un caso concreto es el software llamado genéricamente OCR (Optical Character Recognition) que reconoce la escritura manuscrita con estilo cursivo y la trata como si fueran patrones [Dud., 00].
b) Reconocimiento del lenguaje hablado [Rod., 01]. Es una de las tecnologías que ha tenido mayor avance en los últimos años. Sus herramientas tienen la capacidad de reconocer lenguajes de diversas complejidades, ya sea un hablante, con un vocabulario muy limitado, u otro inmerso en vocabularios flexibles compuestos por miles de palabras. En este contexto, se encuentran las aplicaciones que realizan la conversión voz-texto, las cuales permiten al usuario una comunicación con la máquina, una conversión texto-voz simulando el proceso de lectura de un texto almacenado en formato electrónico, o el reconocimiento de locutores con el que se verifica la identidad del hablante de forma automática a partir de la señal de voz; finalmente, la codificación de voz que realiza la búsqueda de

representaciones eficientes en formato digital de la señal de voz para su almacenamiento y/o transmisión.

c) Análisis de imágenes y visión artificial. En este campo, la aplicación del reconocimiento de patrones es sumamente variada, se trata de sistemas que obtienen imágenes de una cámara de vídeo y actúan por sí mismas en tareas como el reconocimiento de entradas y salidas del personal de una empresa, la identificación del sujeto mediante la captura de la imagen de vídeo o la detección de objetos en movimiento. Otras aplicaciones de esta misma naturaleza son la de detección de billetes falsos o de defectos en productos industriales.

d) Biometría. La identificación de individuos a partir de una característica anatómica o un rasgo de su comportamiento es una de las aplicaciones de mayor auge en la actualidad. Características anatómicas como una huella dactilar, el rostro, la voz, la silueta de la mano, patrones de la retina o el iris tienen la cualidad de ser relativamente estables en el tiempo [Hon., 98]. En aplicaciones prácticas, la detección de la huella dactilar es captada por un lector, y puede ser buscada en la base de datos del personal. Otras aplicaciones similares son la lectura de iris por la digitalización de imágenes captadas con rayos infrarrojos. El iris forma un patrón único para cada persona y puede ser usado como método de identificación similar a las huellas digitales.

e) Biología. En esta área, se han realizado amplios estudios para el reconocimiento de patrones biológicos de proteínas y genes, y más recientemente ha servido para estudiar la hipervariabilidad genética de los virus del sida, en los que, con la ayuda de herramientas propias de reconocimiento sintáctico de patrones, se han podido identificar determinadas gramáticas y restricciones sistemáticas que coinciden con segmentos específicos de la conformación genética viral, lo que ha permitido una formalización sintáctica de estos virus [Pat., 89].

2.2 ENFOQUES DE RECONOCIMIENTO DE PATRONES

Existen varios enfoques en el reconocimiento automático de patrones. A continuación, se describen brevemente algunos de lo más utilizados, como son: los que se basan en la teoría de probabilidad y estadística, los que utilizan funciones discriminantes, los que se basan en la neuro-computación y los que utilizan algoritmos de búsqueda o métodos de optimización basados en heurística.

2.2.1 RECONOCIMIENTO ESTADÍSTICO DE PATRONES

Para realizar el proceso de reconocimiento, se tienen dos diferentes enfoques [Fuk., 90]: reconocimiento paramétrico y reconocimiento no paramétrico. El primero de

ellos realiza el reconocimiento de patrones por medio de métodos estadísticos. Estos sistemas utilizan la Teoría de Decisión de Bayes como base para averiguar la probabilidad matemática de cada una de las clases o categorías. Esto quiere decir que de antemano se conocen las probabilidades asociadas a cada una de las clases y, a partir de ellas, se hace la predicción de la etiqueta de clase que corresponda a un caso nuevo.

A diferencia del reconocimiento paramétrico, en el reconocimiento no paramétrico se dispone de un conjunto de patrones representados de forma vectorial, que mediante la utilización de funciones discriminantes se establecen regiones en el universo de estudio donde se encuentran las clases a las que pertenecen los patrones y, para determinar la clase a la que pertenece un patrón nuevo (no conocido), utiliza únicamente la información proporcionada por los patrones de entrenamiento.

2.2.2 RECONOCIMIENTO SINTÁCTICO DE PATRONES

Este tipo de reconocimiento busca las relaciones estructurales que guardan los objetos de estudio, es decir, busca la cantidad de información que un objeto *x* tiene sobre otro objeto *y*, y el metalenguaje con el que éste último puede ser capaz de describirlo. Para ello, hace uso de descriptores sintácticos con la ayuda de la teoría de lenguajes formales [Fu., 82].

Algunas aplicaciones de este tipo de reconocimiento son en biología molecular para el análisis de secuencias de proteínas, de biosecuencias de ADN, para evaluar la eficiencia de alfabetos reducidos de aminoácidos y, en psiquiatría, para proponer nuevas metodologías de investigación tendentes a cuantificar los patrones dinámicos de la relación paciente-terapeuta [Rap., 91], entre otras.

2.2.3 REDES NEURONALES ARTIFICIALES

Este enfoque utiliza aprendizaje supervisado para realizar los entrenamientos de una estructura formada por varios nodos (neuronas) interconectados entre sí mediante pesos, y que son agrupados en diferentes capas (de entrada, oculta y de salida). Esta estructura es "entrenada" con los patrones de cada clase, de tal forma que al finalizar el entrenamiento, la red neuronal tenga la capacidad de etiquetar nuevos patrones de forma eficiente en poco tiempo.

Debido a la capacidad de resolver problemas no lineales, al elevado poder de clasificación, y sobre todo, a la capacidad que tienen para aprender una determinada tarea, se han convertido en una de las herramientas más atractivas para la solución de diversos problemas del reconocimiento de patrones. Sin embargo, pese a las ventajas que ofrecen, cuentan con algunos inconvenientes:

desconocimiento a priori de la estructura de capas y número de nodos necesarios para cada problema, la posibilidad de caer en mínimos locales durante el entrenamiento de la red y, en ocasiones, contar con un aprendizaje excesivamente costoso en cuestiones de tiempo. Debido a la importancia que este enfoque tiene en la investigación, se abordará más ampliamente en apartados posteriores.

2.2.4 RECONOCIMIENTO LÓGICO COMBINATORIO DE PATRONES

Este enfoque se basa en la idea de que la modelación del problema debe ser lo más cercana posible a la realidad del mismo, sin hacer suposiciones que carezcan de fundamento. Uno de sus aspectos esenciales es que las características utilizadas para describir a los objetos de estudio deben ser tratadas cuidadosamente. Para realizar el reconocimiento, se auxilia de formalismos matemáticos (deducción matemática), que le permiten derivar nuevos conocimientos a partir de conocimientos existentes.

2.3 ETAPAS DE UN SISTEMA DE RECONOCIMIENTO DE PATRONES (SRP)

En la actualidad, no se ha establecido de forma concluyente el proceso sobre cómo los seres vivos realizamos el reconocimiento de objetos. No obstante, diversos estudios en el área proponen cuatro etapas básicas que pueden seguirse para reconocer y clasificar patrones [Cor., 01].

El proceso de la Figura 2.1 no debe verse como los pasos a seguir en la construcción de un SRP, sino más bien desde un punto de vista funcional, donde cada uno de los módulos ya está diseñado y funciona adecuadamente. Además, en ocasiones, no todos los SRP requieren cubrir todos estos pasos, y en otros casos, estos pasos no están claramente separados. En general, el sistema inicia con la entrada de un patrón natural y culmina con la salida de la etiqueta de clase que corresponde a un caso nuevo.

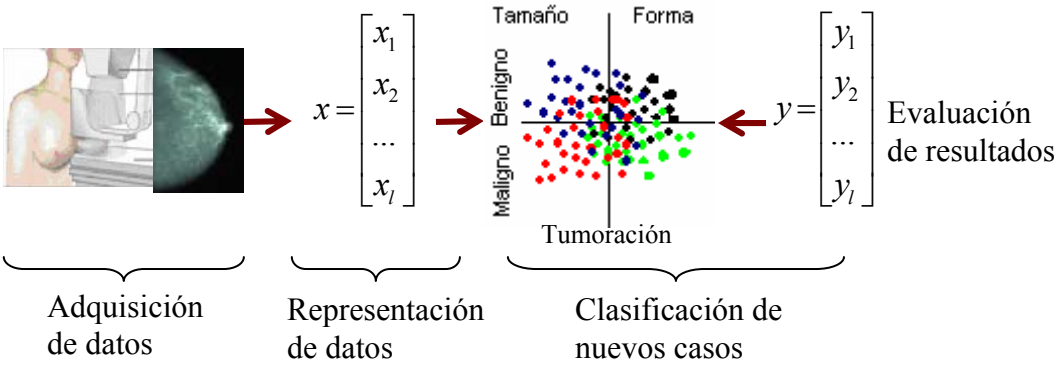


Figura 2.1. Etapas de un SRP

2.3.1 OBTENCIÓN DE LA INFORMACIÓN

La extracción automática de la información es la primera etapa del SRP. Hasta hace no muchos años, la adquisición de datos a procesar se realizaba de forma manual. Por ejemplo, para sistemas de información geográfica, la información era procesada a partir de imágenes a escala obtenidas desde plataformas aéreas o espaciales; durante el proceso de transformación inevitablemente se contaba con pérdida de información, costos elevados de procesamiento y cierto grado de obsolescencia. Actualmente, la tecnología digital nos permite mayores niveles de automatización en la extracción de información, pudiendo obtenerla mediante diferentes dispositivos que varían dependiendo de cada aplicación: cámara de video, micrófono, termómetros, barómetros, ultrasonido, sensores de flexibilidad, sonares, infrarrojos, proximidad, y muchos otros.

2.3.2 REPRESENTACIÓN DE LA INFORMACIÓN

El objetivo perseguido en esta fase evidentemente es representar el conocimiento de forma entendible por la computadora. Esta representación comúnmente se realiza en tres etapas: formación de patrones, selección de características y normalización de variables. En la primera de ellas, se forma estructuras o patrones que contemplan las características obtenidas del mundo real. La segunda etapa consiste en un proceso interpretativo que extrae las características más representativas del objeto. Finalmente, para garantizar la estabilidad en los cálculos, se transforma (normaliza) cada una de las variables del patrón a un rango estándar.

2.3.2.1 Patrones

Un patrón es una unidad de información, integrada de tal forma que capture la esencia descriptiva de un objeto, teniendo como meta principal la representación de cualquier entidad del mundo real a la que se le pueda dar un nombre y sea descriptible. Un patrón puede ser representativo de caracteres escritos, símbolos, dibujos, imágenes, objetos tridimensionales, firmas, huellas dactilares, espectrogramas, etc. Por ejemplo, en una señal sonora, sus características son el conjunto de coeficientes espectrales extraídos de ella (espectrograma) o, en una imagen de una cara humana, sus características están formadas por un conjunto de valores numéricos calculados a partir de la misma.

Toda la información relacionada con las entidades que se desean procesar, se representan mediante la forma de un vector de longitud fija n-dimensional $\mathbf{x} = [x_1, x_2, \ldots, x_n]^T$. Este vector se puede interpretar como un punto en el espacio euclídeo $\mathbf{R}^n$ con n ***características*** o ***atributos*** cuantificables que describen las entidades (estatura, edad, color de piel, color de ojos, medidas, peso, etc.). Estos atributos

pueden ser numéricos (discretos o continuos) y no numéricos (booleanos o k-valentes):

a) De intervalo: en este tipo de datos, todas las variables son cuantitativas, medidas en escala, intervalo o razón. Por ejemplo, un día de verano en el que hay 30° C, es consistente e incrementalmente más caluroso que un día de invierno de 8° C. En este caso, la unidad "grado" forma una referencia consistente entre los dos valores.

b) Frecuencias: en los datos de tipo frecuencia, las variables son categóricas de forma que, por filas, hay objetos o categorías de objetos y, por columnas, las variables con sus diferentes categorías.

c) Datos binarios: en este tipo de datos, las variables analizadas son binarias de forma que 0 indica la ausencia de una característica y 1 su presencia.

d) Numéricos nominales y ordinales**:** los tipos de datos numéricos nominales no conllevan ordenación. Un 3 no es más grande, más intenso, más importante que un 1… tan sólo significa que no es 1. Por su parte, los numéricos ordinales sí llevan implícito un criterio de ordenación, los números forman una progresión de menor a mayor, pero no necesariamente debe haber un paso homogéneo entre ellos. Por ejemplo, si tenemos cinco tipos diferentes de suelos con sus productividades agrícolas asociadas (1 sería la peor y 5 la mejor), un valor de 5 en un suelo no significa necesariamente que sea cinco veces más productivo que otro suelo con un valor de 1.

El conjunto de datos final es descrito mediante una matriz de m patrones × n características (Tabla 2.1), más una columna adicional para la etiqueta de clase, en la que cada fila representa un patrón y cada columna representa el valor de un atributo. Los patrones de entrenamiento considerados son agrupados en c ***categorías*** o ***clases*** denotadas cl_t, donde t = 1, 2,…,c, $\Omega = \{cl_1, cl_2, \ldots, cl_c\}$. De tal forma que los patrones de entrenamiento de una determinada categoría comparten alguna(s) característica(s) que los diferencia de los patrones existentes en otras categorías.

Cabe mencionar que, tanto la selección de las variables a considerar, como los patrones a utilizar y la categorización, son actividades comúnmente realizadas por un experto en el área de estudio.

Tabla 2.1. Categorización de objetos

$x_{1,1}$	$x_{1,2}$	…	$x_{1,n}$	cl_c
$x_{2,1}$	$x_{2,2}$	…	$x_{2,n}$	cl_c
…	…	…	…	…
$x_{m,1}$	$x_{m,2}$	…	$x_{m,n}$	cl_c

2.3.2.2 Selección de características

El problema de la selección de características (o atributos) puede ser planteado del siguiente modo: dado un número de características, ¿cómo se puede seleccionar las más importantes de tal forma que podamos reducir su número y, al mismo tiempo, conservar sus características discriminantes? [Bis., 95]. Por ejemplo, en una imagen, el número de características puede ser tan grande como el número de píxeles (en un patrón de 50 × 50 píxeles, la dimensión del vector de características sería 2500). Debido a esto, surge la necesidad de reducir la dimensionalidad de los patrones, buscando por un lado la optimización del tiempo de cálculo y, por otro lado, evitar la redundancia al tratar con características altamente correlacionadas.

Hay varios tipos de problemas en donde existe un gran número de características y es necesaria la implementación de algoritmos de selección de características. Como ejemplo de ellas, se pueden mencionar las siguientes [Jai., 97]:

a) Aplicaciones donde se fusionan datos provenientes de múltiples sensores.
b) Integración de múltiples modelos, donde se juntan los parámetros de diferentes modelos matemáticos para propósitos de clasificación. Por ejemplo, combinación de características de diferentes modelos de textura en imágenes.
c) Aplicaciones de minería de datos, donde el objetivo es recuperar las relaciones ocultas entre un gran número de características.

Para conjuntos de datos de poco tamaño, aparece el efecto conocido como “la maldición de la dimensionalidad”, en el cual el desempeño del clasificador mejora al agregar nuevas características hasta alcanzar un máximo, para luego decaer. Trunk [Tru., 79] ha demostrado este efecto en un problema simple, en el cual se conocen las funciones de distribución de probabilidad. La probabilidad de error se aproxima sin cambios a cero cuando el número de características aumenta y los valores medios de las distribuciones son conocidos. Por el contrario, la probabilidad de error se aproxima a 0.5 cuando la dimensionalidad crece y los parámetros son estimados a partir de un conjunto de datos de tamaño finito.

El proceso de selección de atributos se enfrenta con una serie de problemas de difícil predicción por los sistemas de aprendizaje [Fri., 00]:

a) Confusión inducida por atributos. En algunas ocasiones, se presenta confusión generada ya sea por uno o varios atributos. Esto sucede cuando dos diferentes objetos tienen el mismo valor en uno o más de los atributos.
b) Datos perdidos. El manejo de datos incompletos puede deberse a pérdida de valores de algún atributo o a la ausencia del mismo. En ambos casos, la incidencia en el resultado dependerá de si el dato incompleto es relevante o

no para el objetivo del sistema de aprendizaje. Por ejemplo, un sistema para aprender a diagnosticar arritmias cardíacas no se verá afectado por la pérdida de datos como el color del pelo del paciente, pero sí por otros como el ritmo cardíaco.

c) Ruido e incertidumbre. El ruido existente en los datos viene determinado tanto por el tipo de valores de los atributos: real (presión arterial), entero (ritmo cardíaco), cadena de caracteres (nombre del paciente) o nominal (tipo de arritmia), como por la exactitud en la medida de dichos valores (especialmente para atributos numéricos). Este problema, por lo general, se debe a la indeterminación existente en muchos aspectos de la realidad y a imprecisiones en el modelado de la misma (no hay que olvidar que en el diseño de una muestra de entrenamiento se modela la realidad y todo modelo no es sino una aproximación más o menos precisa a esa realidad).

Los métodos de selección de características requieren de los siguientes tres elementos:

a) Un criterio de evaluación J para comparar subconjuntos de características que nos indique por qué un subconjunto es mejor que otro.
b) Un procedimiento sistemático de búsqueda de los subconjuntos.
c) Un criterio de detención, típicamente un umbral de significancia o la dimensión del espacio final de características.

El proceso general consiste en la selección de un subconjunto de l características de entre un conjunto original de n características candidatas bajo algún criterio de desempeño. El objetivo principal de esta selección es caracterizar un objeto con medidas o cualidades cuyos "valores" tienden a ser similares entre los objetos de una misma clase y distintos entre objetos de otras clases, es decir, características que tienden a aumentar la distancia entre clases y a disminuir la distancia dentro de las clases.

Además del objetivo que se acaba de exponer, otros propósitos de la selección de características son:

a) Reducir la complejidad del clasificador y su implementación en hardware/software.
b) Reducir el costo de medición al disminuir el número de características.
c) Proporcionar una mejor clasificación debido a efectos por tamaño finito del patrón.
d) Compresión de información. Eliminar datos que no representan información significante o aquellos que estén fuertemente correlacionados, irrelevantes y/o redundantes.

2.3.2.3 Normalización

Una vez que las características han sido seleccionadas, éstas pueden tener un rango diverso de valores. Para la mayoría de las aplicaciones es conveniente transformar los datos de modo que el proceso de cálculo mantenga propiedades de estabilidad. En el caso más sencillo, estas transformaciones son lineales respecto a los datos de entrada, y, a veces, también a los de salida. A este proceso se le llama *normalización* y, básicamente, consiste en la transformación de un conjunto de datos a otro, con media cero y desviación estándar uno. Se pasa de la variable x_n a la variable $\hat{x}_n$ y así transformando cada valor con la siguiente formula:

$$\hat{x}_{i,n} = \frac{x_{in} - \bar{x}_n}{\sigma_n}$$

donde:

$\bar{x}_n = \frac{1}{m}\sum_{i=1}^{m} x_{i,n}$ es la media de cada variable x_n,

$\sigma_n = \sqrt{\frac{1}{m-1}\sum\left(x_{i,n} - \bar{x}_n\right)^2}$ es la desviación estándar, y

m es la talla del conjunto de datos.

2.3.3 CLASIFICACIÓN

La clasificación se puede interpretar como la partición del espacio de características en regiones mutuamente excluyentes, de tal forma que cada región este asociado a una clase t (t = 1,…, *c*) y, dado un patrón particular, decidir a qué clase, de las *c* disponibles, pertenece [Gor., 99]. En otras palabras, un sistema de reconocimiento de patrones se puede considerar como un clasificador que asigna, a un patrón específico, una determinada etiqueta de clase. Este proceso se realiza en dos etapas: aprendizaje y reconocimiento.

2.3.3.1 Aprendizaje

Casi todos los sistemas de reconocimiento de patrones son sistemas complejos, y en ellos, es posible no tener alguna suposición sobre lo que sería el objetivo al diseñar el sistema de clasificación. Debido a ello, cualquier método que incorpora información sobre un patrón o conjunto de entrenamiento en el diseño del clasificador, necesariamente emplea algún tipo de aprendizaje y algún tipo de generalidad, lo que hace necesario utilizar los patrones de entrenamiento para

resolver las incógnitas en los patrones del clasificador. Este aprendizaje se refiere al algoritmo que permitirá reducir la cantidad de errores en la información para el entrenamiento y puede realizarse fundamentalmente con dos vertientes:

a) Aprendizaje Supervisado [The., 98]. Las técnicas de aprendizaje supervisado disponen para su ejecución, de un conjunto de patrones, integrado en lo que se conoce como muestra de entrenamiento o conjunto de datos de entrenamiento (CE). Este conjunto de datos es recolectado por un experto humano en el campo de estudio y agrupa en clases o categorías, de acuerdo a las propiedades que cada uno posee, los casos resueltos previamente. El clasificador es entrenado con este CE y realiza la identificación de la clase correspondiente para nuevos patrones, empleando el conocimiento ya adquirido y tratando de realizar esa identificación con el menor error posible.
b) Aprendizaje no Supervisado [Mic., 00]. Estos algoritmos, también conocidos como auto-asociativos, no requieren un etiquetado previo de cada uno de los patrones de entrada. Mediante el uso de técnicas de agrupamiento (en inglés, clustering), organiza los patrones de acuerdo a rasgos particulares que los identifiquen y diferencien unos de otros, teniendo como resultado final conjuntos que contienen patrones con características lo más parecidas entre ellos y lo más distintas posible con los patrones contenidos en otros grupos.
c) Aprendizaje Semi-supervisado o Parcialmente Supervisado [Man., 05]. Es un enfoque relativamente nuevo, que trata de combinar las bases de las dos vertientes clásicas de aprendizaje. Así, se parte de un conjunto (generalmente) pequeño de patrones de entrenamiento que, progresivamente puede ir ampliándose mediante el uso de un conjunto de nuevos patrones sin etiquetar. Uno de los objetivos de este nuevo modelo de aprendizaje es aumentar el conocimiento durante la fase de clasificación, permitiendo de este modo, simplificar el costoso proceso para obtener patrones etiquetados.

2.3.3.2 Reconocimiento

Cuando el proceso de aprendizaje ha terminado de manera aceptable con todos los patrones considerados para representar el mundo real (en el área de estudio), se puede pasar a una fase de reconocimiento o clasificación, “mostrando” al clasificador (o regla de decisión) los patrones que se desea reconocer.

Los métodos de clasificación basados en distancias son uno de los procedimientos más extendidos. Bajo esta perspectiva, la regla de decisión asume que si el patrón de prueba se localiza dentro de una región cercana al patrón de referencia del CE, el patrón entrante (nuevo) corresponde a la clase o grupo asociado al patrón de referencia. Para este fin, se hace uso de algunas medidas de

proximidad o de distancia entre los objetos que cuantifique el grado de similaridad entre cada par de objetos.

Tanto lo concerniente al aprendizaje como lo referente al reconocimiento se proporcionará más detalladamente en el Capítulo 3 "Clasificación y preprocesado" de este documento.

2.3.3.3 Medidas de proximidad

Las medidas de proximidad miden el grado de semejanza entre dos objetos de forma que, cuanto mayor es su valor, mayor es el grado de similaridad existente entre ellos, y con más probabilidad los métodos de clasificación tenderán a ponerlos en el mismo grupo.

En la literatura, existen multitud de medidas de distancia en función del tipo de variables y datos considerados. Por definición, una **métrica** *d* es una función real de dos puntos en el espacio de representación de los datos, tal que para todos los puntos *x, y* y *z* en el plano de representación general, se satisfacen las siguientes propiedades [Dud., 73].

a) $d(x,y) >= 0$ y $d(x,y) = 0$ si y solo si x = y (positividad)
b) $d(x,y) = d(y,x)$ (simetría)
c) $d(x,y) + d(y,z) >= d(x,z)$ (desigualdad triangular)

A continuación, se mencionan algunas de las métricas más utilizadas [Dud., 73]. En todos los casos, la dimensionalidad de los vectores es *n*.

2.3.3.3.1 Métricas para variables cuantitativas

a) Distancia Euclídea: $d_E(X,Y) = \sqrt{\sum_{j=1}^{n}\left(x_j - y_j\right)^2}$

b) Distancia de Manhattan**:** $d_M(X,Y) = \sum_{j=1}^{n}\left|x_j - y_j\right|$

c) Distancia de Minkowski**:** $d_{Mk}(X,Y) = \sqrt[q]{\sum_{j=1}^{n}\left(x_j - y_j\right)^q}$ con $q>0$.

Estas tres primeras medidas de similaridad son variantes de la distancia de Minkowski con $q = 2$, 1 e ∞, respectivamente. Cuanto mayor es q, más énfasis se le da a las diferencias en cada variable.

Todas estas distancias no son invariantes a cambios de escala, por lo que se aconseja estandarizar los datos si las unidades de medida de las variables son incomparables. Además, no consideran las relaciones existentes entre las variables. Si se quieren tener en cuenta, se aconseja utilizar la distancia de *Mahalanobis,* que se obtiene con la forma cuadrática:

$$d_{Mh} = \sqrt{(\eta - \mu)^T \sum^{-1} (\eta - \mu)}$$

donde:

$\eta \approx \frac{1}{m}\sum_{i=1}^{m} x_i$ Media de los *m* vectores de entrenamiento

$\Sigma \approx \frac{1}{m}\sum_{i=1}^{m} (x_i - \eta)(x_i - \eta)^T$ Matriz de covarianza de los vectores de entrenamiento *x*

$\mu \approx \frac{1}{p}\sum_{i=1}^{p} y_i$ Media de los *p* vectores de prueba

Otras métricas utilizadas son:

d) Métrica del valor absoluto: $d_{abs}(x,y) = |x_1-y_1| + |x_2-y_2|$

e) Métrica del valor máximo: $d_{max}(x,y) = \max\{ |x_1-y_1| , |x_2-y_2|\}$

2.3.3.3.2 Medidas para datos binarios

En este tipo de medidas se construyen, para cada par de objetos r y s, tablas de contingencia de la forma:

Tabla 2.2. Tabla de contingencia para datos binarios

	r = 0	**r = 1**
s = 0	**a**	**b**
s = 1	**c**	**d**

Para utilizar estas tablas, algunas de las medidas de semejanza más recurridas son [Gor., 99]:

a) Coeficiente de Jacard: $d_J = \frac{d}{b+c+d}$

b) Medida de Lance y Williams: $d_{LW} = \frac{b+c}{2d+b+c}$

c) Coeficiente de acuerdo simple: $d_{Ac} = \frac{a+d}{p}$

d) Coeficiente de desacuerdo simple: $d_{Desac} = b+c$

donde:

a = número de variables en las que los objetos r y s toman el valor 0.
b = número de variables en las que el objeto r toman el valor 1 y el objeto s 0.
c = número de variables en las que el objeto r toman el valor 0 y el objeto s 1.
d = número de variables en las que los objetos r y s toman el valor 1.
p = a + b + c + d.

Estas cuatro medidas toman valores entre 0 y 1 y miden el porcentaje de acuerdo con los valores tomados de las *n* variables existentes entre los dos objetos. Difieren en el papel dado a los acuerdos en 0, puesto que el coeficiente de Jacard y la medida de Lance y Williams no los tienen en cuenta. Ello se debe a que, en algunas situaciones, las variables binarias consideradas son asimétricas, en el sentido de que es más informativo el valor 1 que el valor 0. Así, por ejemplo, si el color de los ojos de una persona se codifica como 1 podría significar que los tiene azules y, en caso contrario, se codifica en 0. En este tipo de situaciones, es más conveniente utilizar el coeficiente de Jacard y la medida de Lance y Williams.

2.3.3.3.3 Medidas para datos de tipo mixto

Si en la base de datos existen diferentes tipos de variables (binarias, categóricas, ordinales y/o cuantitativas), no existe una solución universal al problema de cómo combinarlas. Para construir una medida de distancia, se sugieren las siguientes soluciones [And., 73]:

a) Expresar todas las variables en una escala común, habitualmente binaria, transformando el problema en uno de los ya contemplados anteriormente. Esto tiene sus costes, en términos de pérdida de información si se utilizan escalas menos informativas como las nominales u ordinales, o la necesidad

de incorporar información extra si se utilizan escalas más informativas como son las de intervalo o razón.

b) Combinar medidas con pesos de ponderación mediante expresiones de la forma:

$$d(x,y) = \frac{\sum_{i=1}^{n} v_{xyi} \times d_{xyi}}{\sum_{i=1}^{n} v_{xyi}}$$

donde, d_{xyi} es la distancia entre los objetos *x* e *y* en la i-ésima variable, v_{xyi} es 0 ó 1 dependiendo de si la comparación entre *x* e *y* es válida en la i-ésima variable.

c) Realizar el análisis por separado utilizando variables del mismo tipo y utilizar el resto de las variables como instrumentos para interpretar los resultados obtenidos.

2.3.4 MÉTODOS DE EVALUACIÓN

Tanto en la etapa de aprendizaje como en la de reconocimiento, es necesario contar con elementos que permitan evaluar la eficiencia de los procesos. A continuación, se describen una serie de métodos encaminados a la evaluación del clasificador (comúnmente utilizado en la etapa de aprendizaje), y a la evaluación de la clasificación en la etapa de reconocimiento.

2.3.4.1 Evaluación del clasificador

En esta fase, se hace uso de los resultados obtenidos por el clasificador, a través del análisis de la tasa de aciertos o errores, y califica al clasificador para recomendar decisiones y acciones que dependen de un costo o riesgo particular. Para realizar este análisis, se emplean métodos conocidos como **estimadores del error**. Su objetivo es calcular la proporción de patrones clasificados de forma incorrecta por el clasificador. Con la utilización de un conjunto de datos *R* compuesto por *m* patrones distribuidos en *c* clases y, dado un patrón **y** que se presenta a todo patrón de entrenamiento $\mathbf{x}_i$ (i = 1,…, *m*), se obtiene la siguiente función *E* que indica el acierto o error considerando el siguiente criterio, donde $T(x_i)$ y T(y) representan las etiquetas de clase de los patrones x_i e **y**:

$$E(x_i, y) = \begin{cases} 0 \;\; si \;\; T(x_i) = T(y) \\ 1 \;\; si \;\; T(x_i) \neq T(y) \end{cases}$$

La evaluación de *R* puede realizarse con los siguientes métodos [Cor., 01]:

a) Resustitución o Reclasificación (Método R): se construye un conjunto *R* de tamaño *r* utilizando los *m* patrones contenidos en *M*. Se clasifican todos los patrones de *M* utilizando *R* y se observa la cantidad de patrones mal clasificados para posteriormente obtener la estimación dada por:

$$\frac{1}{m}\sum_{x\in M} E(x,y)$$

Los problemas con los que cuenta este método son: un sesgo optimista del error de *R,* debido a que el error es calculado utilizando dos conjuntos de patrones idénticos entre sí, y el requerimiento de una gran cantidad de tiempo y memoria RAM para su ejecución al ser aplicado a grandes conjuntos de datos.

b) Partición mediante un conjunto de prueba (Hola-out) [Koh., 95]: en este método, se divide el conjunto *M* en dos conjuntos disjuntos R_1 (conjunto de aprendizaje) y R_2 (conjunto de prueba) de tal forma que $R_1 \cap R_2 = M$ y $R_1 \cap R_2 = \emptyset$. Para que los patrones que contenga R_2 sean diferentes de los contenidos en R_1, habitualmente se realiza la partición de *M s*eleccionando los patrones de forma aleatoria sin reemplazo, de forma que la unión de los patrones contenidos en los dos grupos R_1 y R_2 sea el total de patrones contenidos en *M*. Para la estimación del error, se clasifican los patrones de R_2 utilizando R_1. La estimación es dada por:

$$\frac{1}{|R_2|}\sum_{x\in R_2} E(x,y)$$

c) Validación cruzada con *V* conjuntos o Rotación (Método π) [Bai., 93]: en este método, se realiza la distribución de los patrones contenidos en *M* de forma aleatoria en *V* conjuntos disjuntos $R_1, R_2, \ldots R_v$ de un tamaño similar.

Para todo R_v, donde $v = 1, 2, \ldots, V$, se construye un conjunto de aprendizaje utilizando $M - \{R_v\}$. La estimación se calcula por:

$$\frac{1}{|R_v|}\sum_{x\in R_v} E(x,y)$$

Para valores grandes de *V*, cada uno de los *V* clasificadores se construye utilizando un conjunto de patrones de tamaño aproximado a $m(1\text{-}1/V)$, casi tan grande como lo es *M.* La estimación final se obtiene:

$$\frac{1}{V}\sum_{v=1}^{V} Estimación\ de(R_v)$$

d) Validación cruzada con $V = m$ (leave–one-out): en este método, a diferencia del anterior, se utiliza un solo patrón para evaluar la calidad de M. Para cada patrón x_i, donde i = 1, 2,..., m, el i-ésimo patrón es descartado y el CE se evalúa utilizando los restantes $M - \{x_i\}$ patrones. Entonces, el patrón descartado se usa para prueba y se estima el error:

$$\frac{1}{m}\sum_{x\in M} E(x, y)$$

Este estimador es el más adecuado cuando se trabaja con conjuntos de datos pequeños. Pero, tiene como gran inconveniente el alto costo computacional que requiere: todos los patrones contenidos en M se usan para obtener la estimación del error, y cada uno de ellos se usa sólo una vez para prueba.

2.3.4.2 Evaluación de la clasificación

Lo más común es utilizar la precisión general para evaluar el desempeño del clasificador. No obstante, este criterio no es adecuado cuando existe un desbalance [Bar., 01a] en el conjunto de datos por la inherente tendencia de clasificar de manera errónea patrones de la clase minoritaria debido a la gran influencia de la clase mayoritaria. Por ejemplo, si consideremos la clasificación de la base de datos Ism (imágenes de mamografía) [Bar., 03b], donde existen dos clases, una de ellas con el 98% de los patrones, una simple estrategia de precisión general proporcionará a la clase mayoritaria un 98% de acierto. No obstante, la naturaleza de esta aplicación requiere de un mayor equilibrio para corregir la detección en la clase minoritaria y para tolerar un pequeño ratio de error en la mayoritaria. En este sentido, a continuación, se proporcionan otros criterios de evaluación, tales como la precisión general y las matrices de confusión.

2.3.4.2.1 Precisión general y media geométrica

Para encontrar la precisión general, se hace uso de la media aritmética de p valores de manera que:

$$\text{P.G.} = \frac{\sum_e x}{p}$$

donde e = patrones correctamente clasificados y, p = total de patrones presentados para su clasificación.

A pesar de que este criterio de evaluación resulta sencillo en su principio e implementación, resulta poco adecuado por considerar de manera conjunta las precisiones individuales de las clases, situación no presente en la media geométrica en la que considera separadamente las precisiones observadas por cada una de las clases, aspecto de especial importancia al trabajar con conjuntos de datos que presentan desbalance:

$$g = \sqrt{a^+ \cdot a^-}$$

donde $a^+ = 1 - errores_{a^+} / Patrones_{a^+}$ (precisión de la clase menos representada), $a^- = 1 - errores_{a^-} / Patrones_{a^-}$ (precisión de la clase más representada).

Una generalización de esta medida es, al ser utilizada para una cantidad finita *q* de valores, la raíz *q*-ésima del producto de todos los valores.

$$\overline{x} = \sqrt[q]{\prod_{i=1}^{q} x_i} = \sqrt[q]{x_1 * x_2, ..., x_q}$$

Por ejemplo la media geométrica de 1, 3 y 9 seria:

$$\sqrt[3]{1 * 3 * 9} = \sqrt[3]{27} = 3$$

La media geométrica sólo es útil si todos los números son positivos. Si uno de ellos es 0, entonces el resultado es 0. Si hay un número negativo (o una cantidad impar de ellos), entonces la media geométrica es, o bien negativa o bien inexistente en los números reales.

2.3.4.2.2 Varianza y desviación estándar

La varianza nos proporciona una forma natural de medir la dispersión de los datos en torno a la media, calculando la media de las diferencias de los valores:

$$s = \sum_{i=1}^{m} (x_i - \overline{x})$$

Sin embargo, como habrá valores que caigan por encima y algunos otros por debajo de la media, el ajuste útil para compensar esta situación consiste en calcular el cuadrado de las diferencias. De tal manera, la varianza de una variable es la media de los cuadrados de las desviaciones de sus *m* valores respecto a su media y se representa por s^2:

$$s^2 = \frac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x})^2$$

Dada la existencia de variables estadísticas y variables aleatorias, donde las primeras consideran una serie de valores concretos, el cálculo de su varianza es conocido como *varianza muestral* y divide el resultado de la suma entre *m*-1. Por otro lado, para el caso de variables aleatorias, se calcula una *varianza estimada*, ya que no se toma en cuenta el conjunto de datos inmensos y, por lo tanto, la media y la varianza son estimadas, no conocidas, de tal forma que la división de la suma se realiza entre el total de elementos analizados.

La desviación estándar o desviación típica es el criterio de evaluación utilizado para analizar la dispersión de los datos y/o resultados obtenidos en los experimentos, mediante la obtención de los cuadrados de las desviaciones de los valores de la variable respecto a su media. Cuanto mayor sea la dispersión mayor es la desviación estándar. Si no hubiera ninguna variación en los datos, es decir, si fueran todos iguales, la desviación estándar sería cero. De igual forma, se puede decir que un valor está muy alejado del centro de los datos si su distancia de él supera dos desviaciones estándar.

$$\hat{\sigma} = \sqrt{\frac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x})^2}$$

Al igual que para la varianza, es necesario distinguir los casos de variables aleatorias y estadísticas. En esta fórmula se expresa la desviación estándar muestral, que es la que se utilizará en el presente trabajo.

2.3.4.2.3 Matriz de confusión

La matriz de confusión o matriz de contingencia es una herramienta utilizada para la presentación y el análisis del resultado de una clasificación debido a su capacidad de plasmar los conflictos entre las clases. Así, no sólo se conoce el porcentaje correcto de clasificación, sino también la fiabilidad para cada una de las clases y las principales confusiones entre ellas.

Se puede considerar a la matriz de confusión como una matriz cuadrada de orden $c \times c$ (Figura 2.2) con varias filas y columnas auxiliares para contabilizar diversos parámetros estadísticos, tales como: totales, la asignación "real" de patrones por clase, la asignación correcta y el porcentaje de bien clasificados.

Esta matriz es construida utilizando las clases del CE; las filas representan los datos de referencia, los valores marginales indican el número de patrones que, perteneciendo a una determinada clase, no fueron incluidos en ella (errores de

omisión) y las celdas no diagonales de las columnas señalan los resultados de clasificar patrones que se incluyeron en una determinada clase cuando realmente pertenecían a otra (errores de comisión). A lo largo de la diagonal principal, se indican los patrones que fueron clasificados de manera correcta [Rus., 99].

La columna "Asignación por clase" muestra el total de patrones que fueron asignados a una determinada clase, mientras que la columna "Asignación correcta" indica el porcentaje de patrones asignados de forma correcta de esa clase. La fila "% Bien clasificados" proporciona el porcentaje de patrones que fueron clasificados de forma correcta en una determinada clase.

Clases	1	2	3	4	5	6	...	*c*	Asignación por clase	Asignación correcta
1	**399**	62	0	0	6	0	...	0	467	0.85
2	0	**58**	113	0	0	0	...	0	174	0.33
3	0	0	**148**	45	0	0	...	0	193	0.77
4	1	0	0	**92**	0	0	...	6	149	0.62
5	2	0	16	0	**17**	0	...	1	36	0.47
6	1	0	0	0	4	**124**	...	0	332	0.37
...	…	…	…	…	…	…	**...**	…	…	…
c	1	0	0	0	0	0	...	**355**	359	0.99
Total	404	120	277	137	27	124	...	362	1733	5.41
% Bien clasificados	0.99	0.48	0.53	0.67	0.63	1.00	...	0.98	1216	0.70

Figura 2.2. Matriz de confusión

A partir de los resultados obtenidos con la matriz de confusión, se pueden obtener varías técnicas estadísticas descriptivas y analíticas. Una de ellas es la precisión general (en inglés, *overall accuracy*), la cual se obtiene dividiendo el total de patrones bien clasificados entre los mal clasificados. Otro cálculo que puede obtenerse es la precisión por clase, la cual se calcula dividiendo el número de patrones bien clasificados de una clase por el total de patrones presentados. Otro estudio más sofisticado que es factible de realizarse con los resultados de la matriz de confusión es el cálculo del coeficiente y la varianza del Kappa [Rus., 99]. La idea fundamental implica el análisis de las diferencias entre los datos de referencia y los datos entrantes determinados por la diagonal principal.

2.3.4.2.4 Coeficiente Kappa

El coeficiente Kappa es otra forma, altamente difundida, de medir la exactitud de la clasificación. El Kappa es un indicador que adquiere valores entre 0 y 1, representando el primero la absoluta falta de concordancia y el segundo, concordancia total. Un valor menor a 0.4 se considera como expresión de concordancia insuficiente entre 0.41-0.60 como moderada y mayor que 0.60 elevada. Es apropiado para resolver objetivos como el siguiente: evaluar la

concordancia de los diagnósticos anatomopatológicos confrontando los producidos por varios patólogos sobre las mismas piezas, como así también, los dados por el mismo patólogo sobre las mismas piezas en diferentes ocasiones (variabilidad ínter e intra-observador). Este método puede medir la exactitud de manera más precisa que la matriz de confusión por calcular, no solamente los valores de sus columnas de los extremos, sino también los contenidos en el interior de la matriz [Gar., 02].

$$\hat{K} = \frac{m\sum_{i=1}^{\gamma} x_{i,i} - \sum_{i=1}^{\gamma} \left(x_{i+} * x_{+i}\right)}{m^2 - \sum_{i=1}^{\gamma} \left(x_{i+} * x_{+i}\right)}$$

donde r es el número de filas, $x_{i,i}$ número de patrones en una fila i y una columna i, x_{i+} y x_{+i} totales marginales de una fila i y una columna i respectivamente, y m número de patrones evaluados.

Cuando el resultado obtenido por el Kappa muestra que la concordancia es pobre, esto nos puede indicar mayor confiabilidad, a diferencia de cuando se obtiene concordancia elevada. Esto se debe a que, si en la muestra estudiada la proporción de una de las alternativas de calificación es mucho más frecuente que la otra (por ejemplo, gran disparidad entre la proporción de individuos con y sin daño) y para ella la concordancia es adecuada, el resultado expresará una elevada concordancia global, pero en realidad, está informando la concordancia para la alternativa de calificación de mayor frecuencia. Por ello, se ha propuesto calcular por separado la concordancia para cada alternativa, permitiendo una visión más equilibrada del comportamiento del método.

Capítulo 3

Clasificación y técnicas de preprocesado

La tarea de clasificar objetos definidos por sus atributos descriptivos es uno de los enfoques más básicos del aprendizaje automático; este proceso es también conocido como *reconocimiento de patrones* [Fri., 00]. Para este fin, el modelo clasificador (o reconocedor) se crea de forma inductiva mediante la generalización de un conjunto de patrones, donde cada patrón no es más que un punto en el espacio de representación de todos los valores posibles que puede tomar. En la Figura 3.1 puede apreciarse gráficamente un caso de tres clases, en el cual los patrones pertenecientes a una misma clase están cercanos en el espacio de representación, mientras que aquellos que pertenecen a clases diferentes están en diferentes regiones de representación.

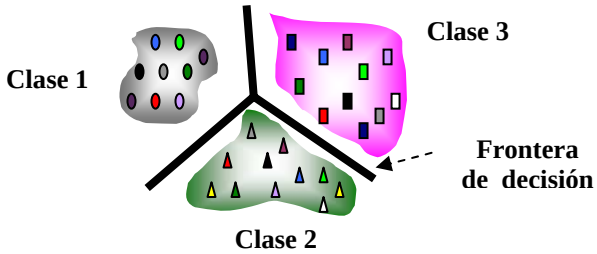


Figura 3.1. Patrones distribuidos en tres clases y su frontera de decisión

El proceso de clasificación tiene dos etapas disjuntas realizadas de forma secuencial:

a) Aprendizaje. En esta etapa, se presentan los patrones de entrenamiento al clasificador con la intención de que, una vez recorrido el total de patrones de entrenamiento, el clasificador esté capacitado para reconocer las clases contenidas en el conjunto de entrenamiento (CE) (ver Figura 3.2).

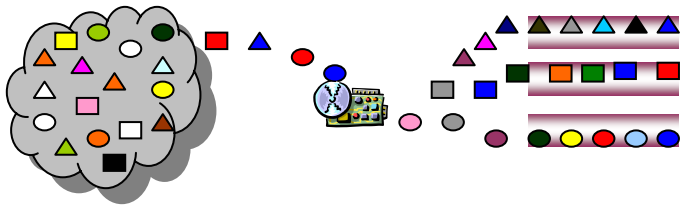


Figura 3.2. Proceso de aprendizaje

b) Producción. Ya sin la supervisión del experto humano y previamente entrenado, el clasificador etiqueta o clasifica a nuevos patrones utilizando para ello únicamente el modelo construido en la etapa de aprendizaje.

Un aspecto importante de mencionar es que, una vez que el clasificador ha aprendido del CE, es incapaz de adaptar su conocimiento ante casos nuevos o errores cometidos durante su producción, siendo esto su principal desventaja.

Como ya se ha comentado en el capítulo anterior, para realizar el aprendizaje del modelo clasificador, se cuenta con dos variantes clásicas: aprendizaje no supervisado y aprendizaje supervisado. La presente investigación se desarrolla sobre el aprendizaje supervisado utilizando métodos no paramétricos, por lo que los métodos incluidos en este grupo serán descritos más ampliamente.

3.1 METODOS DE APRENDIZAJE NO SUPERVISADO

Los métodos no supervisados se suelen usar en el denominado *análisis de datos exploratorio*, es decir, en una fase del análisis de los datos, cuando no se sabe de antemano cuáles son los grupos *naturales* que se forman, ni la relación existente entre ellos, cuando se desea analizar un gran conjunto de datos o, simplemente, cuando existiendo un conocimiento completo de las clases, se desea comprobar la validez del entrenamiento realizado y del conjunto de variables escogido [Zhe., 97]. También se pueden usar como fase inicial de algoritmos de aprendizaje supervisados: un algoritmo como el *k-medias* [Fuk., 90] o el mismo SOM (Self-Organizing Map) [Koh., 88] se pueden usar para inicializar ciertos algoritmos de aprendizaje supervisado, tales como el LVQ (*Learning Vector Quantization)*.

Tradicionalmente, para realizar el agrupamiento de los datos, se cuenta básicamente con dos métodos: agrupamiento jerárquico y agrupamiento por partición. En el algoritmo jerárquico, se van creando clusters pequeños, incluso inicialmente con un solo componente, y se van fusionando hasta obtener clusters de tamaño superior; el resultado final es un árbol de clusters conocido cómo *dendrograma*, que muestra como los cluster se relacionan unos con otros. Este tipo de algoritmos pueden ser, a su vez de dos tipos: aglomerativos y divisivos. El primero caso corresponde a la estructura de algoritmos jerárquicos que acabamos de describir. Por su parte, en los algoritmos divisivos, se parte de todo el conjunto de datos como un único cluster y, en cada paso, se divide uno de los clusters existentes hasta llegar a un resultado final.

Por otro lado, el agrupamiento por partición es aquel que distribuye los objetos del universo de estudio en grupos (cluster), buscando maximizar alguna medida de similitud entre pares de patrones, entre un patrón y un grupo, y finalmente, entre pares de grupos, de forma que los objetos en un mismo grupo sean muy similares entre sí (cohesión interna del grupo) y los de los objetos de cluster diferentes sean distintos (aislamiento externo del grupo).

Generalmente, como medidas de similaridad se emplean métricas de distancia. Algunas de las más habituales son la distancia Euclídea, la distancia Euclídea normalizada, la distancia Euclídea ponderada y la distancia de Mahalanobis [Fis., 91]. Los algoritmos más populares de este grupo son el k-medias y el ISODATA [Car., 90]. Por último, otras técnicas útiles para realizar el agrupamiento son, entre otros, el agrupamiento probabilístico y el agrupamiento basado en la teoría de grafos.

3.2 APRENDIZAJE SUPERVISADO

Dentro del aprendizaje supervisado, pueden distinguirse dos enfoques: aprendizaje paramétrico y aprendizaje no paramétrico. El primero supone un completo conocimiento a priori de la estructura estadística de las clases, por lo que el aprendizaje se reduce a la estimación de los parámetros que determinan las funciones de densidad de probabilidad de las clases, al mismo tiempo que se definen las fronteras de decisión y las distribuciones de probabilidad de clases [Con., 98].

Por otro lado, los clasificadores no paramétricos no cuentan con un determinado modelo probabilístico, bien por desconocimiento o por la imposibilidad de asumir un modelo paramétrico adecuado. Para realizar la clasificación, se han propuesto una gran variedad de funciones discriminantes que dan lugar a diferentes tipos de clasificadores, siendo quizás los mas conocidos, aquellos que utilizan una medida de distancia entre los patrones a clasificar y un conjunto de patrones u objetos de entrenamiento; ejemplo de ellos es la regla de

decisión conocida como la Regla del Vecino más Cercano (NNR, Nearest Neighbor Rule) ó una generalización de esta regla que se conoce como la Regla de los *k* vecinos más cercanos (*k*-NN).

3.3 REGLA DEL VECINO MÁS CERCANO

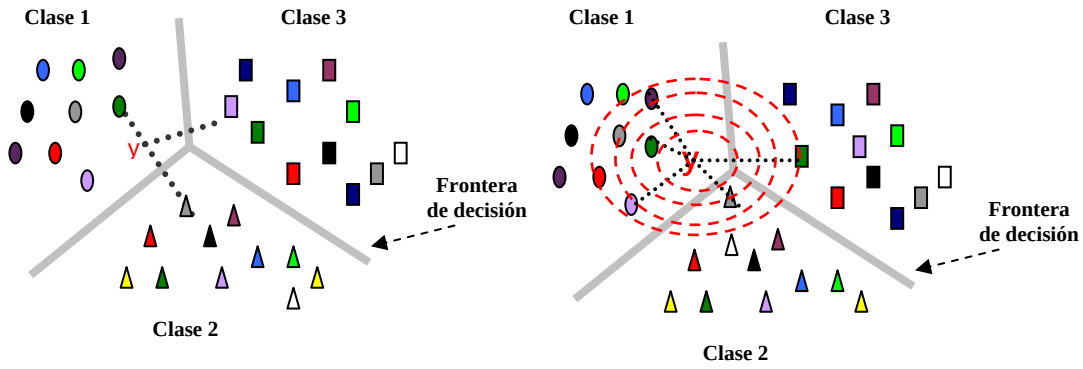
La clasificación mediante este algoritmo, publicado en 1951 por Fix y Hodges [Fix., 51], es una de las primeras investigaciones que proporciona reglas basadas en métodos no-paramétricos para la manipulación de un conjunto de datos. Actualmente, es una de las más utilizadas en reconocimiento de patrones, principalmente por su simplicidad conceptual y por la sencillez de su implementación. La idea básica considera la utilización de un conjunto de patrones de entrenamiento que constituye todo el conocimiento a priori del sistema. Esta regla basa su operación en el supuesto de considerar a los patrones cercanos, como aquellos que tienen la mayor probabilidad de pertenecer a la misma clase. Para ello, cuando se desea clasificar un nuevo caso **y**, se realiza el cálculo de la distancia entre x y los contenidos en el CE, y se asigna la etiqueta correspondiente al patrón que obtuvo la menor distancia (más cercano) a **y**. Esto puede ser formulado de la siguiente manera:

$$d(\mathrm{x_i}, \mathbf{y}) = \min_i(d(\mathrm{x_i}, \mathbf{y}))$$

donde d es cualquier medida de similaridad, **y** es un patrón no conocido por el clasificador, y $\mathrm{x_i}$ es un patrón de entrenamiento (i = 1, 2, 3,...*m* patrones de entrenamiento contenidos en el CE).

La principal desventaja asociada a esta regla respecto a otros clasificadores es el enorme costo computacional que tiene en la etapa de producción. Este coste es lineal al tamaño del CE por lo que, en muchos problemas donde se aplica esta técnica con CE de gran tamaño, resulta ser poco conveniente debido a que el coste inherente al cálculo de las distancias puede ser excesivamente elevado.

Dos de sus modalidades más populares son 1-NN y *k*-NN. En la primera, se busca un sólo vecino, el más cercano al patrón de entrada **y** (ver Figura 3.3a), al cual se asigna la etiqueta de la clase para las que la distancia a uno de sus puntos es mínima. Su mayor utilidad es en casos en los cuales se cuenta con poco solape entre los patrones de las clases en la que se busca un sólo vecino. La segunda es una generalización de la regla 1-NN que realiza la detección de varios vecinos más cercanos a un sólo patrón de entrada y resulta ser más adecuada para aquellos casos que reflejan un alto grado de solape entre clases. En este caso, la estructura del conjunto de patrones viene dada de forma que cada clase c_i contiene al menos *k* patrones ($1 \geq k \geq |c_i|$). Los *k* vecinos más cercanos estarán localizados en un círculo centrado en **y**, y la etiqueta de clase que se asigna al nuevo patrón es la clase predominante entre los *k* vecinos encontrados (ver Figura 3.3b).



(a) Regla 1-NN (b) Regla 5-NN

Figura 3.3. Ejemplo de las reglas 1-NN y *k*-NN. (a) El nuevo patrón se asigna a la clase 1. (b) El nuevo patrón se clasifica como de la clase 1 (3 vecinos)

Para cada asignación de etiqueta de clase, se requiere del cálculo exhaustivo de las distancias existentes entre el patrón a clasificar y el total de los patrones contenidos en el CE hasta encontrar los *k* vecinos.

Existen otras variantes de esta regla: vecinos envolventes (NCN, por sus siglas en ingles Nearest Centroid Neighbor) [San., 98], [San., 02] o el método de vecindad circundante (SN, Surrounding Neighborhood) [Zha., 97], los cuales no se tratarán en esta investigación.

La regla NN es muy utilizada en reconocimiento de patrones y otras muchas disciplinas. Por ejemplo, para realizar consultas a una base de datos. Estas consultas pueden ser de datos concretos, o incluso de datos aproximados (por ejemplo, la búsqueda de los apellidos parecidos a Rojas (Rosas, Rodas, Rolas...)). Esta técnica también puede aplicarse a la detección y corrección de errores tipográficos en los procesadores de texto. El método consiste en la búsqueda en un diccionario de las palabras que más "se parecen" a la que queremos corregir.

3.4 REDES NEURONALES ARTIFICIALES (RNA)

A lo largo de la historia, se han realizado múltiples intentos por imitar el funcionamiento del cerebro. Alan Turing, en 1936, fue el primero en estudiar el cerebro como una forma de ver el mundo de la computación, pero quienes primero concibieron algunos fundamentos de la computación neuronal fueron Warren McCulloch y Walter Pitts. En 1943, McCulloch y Pitts, en su trabajo "A logical calculus of the ideas immanent in nervous activity" [McC., 43], intentaron explicar el funcionamiento del cerebro humano por medio de un modelo de células conectadas entre sí, de tal manera que una célula nerviosa o neurona no es más que

un dispositivo binario con entradas y salidas, desarrollando de esta forma la primer red neuronal conocida. El modelo consistía en la suma de las señales de entrada, multiplicadas por unos valores de pesos escogidos aleatoriamente. La entrada era comparada con un patrón preestablecido para determinar la salida de la red. Si, en la comparación, la suma de las entradas multiplicadas por los pesos era mayor o igual al patrón preestablecido, la salida de la red era 1, en caso contrario, la salida era 0.

Seis años después, el fisiólogo Donald O. Hebb expuso que las redes neuronales podían *aprender* [Heb., 49]. Su propuesta tenia que ver con la conductividad de la sinapsis, es decir, con las conexiones entre neuronas, lugar en donde se memoriza la información. Hebb expuso que "cuando un axón de la célula A excita la célula B y participa en su activación, se produce algún proceso de desarrollo o cambio metabólico en una o en ambas células, de manera que la eficacia de A, como célula excitadora de B, se intensifica". Así mismo, la forma de corrección que emplea esta regla es incrementar la magnitud de los pesos si ambas neuronas están inactivas al mismo tiempo.

Bajo la dirección del científico Burrhus Frederic Skinner, Marvin Minsky diseñó y creó varias máquinas para experimentos de laboratorio. Los conocimientos adquiridos al lado de Skinner le sirvieron como base para crear en 1951 la primera máquina de redes neuronales. Para su construcción, estuvo en coordinación con el electrónico Dean Edmonds. Esta máquina estuvo compuesta básicamente por 300 tubos de vacío y un piloto automático de un bombardero B-24. Llamaron a su creación "Sharc", se trataba de una red de 40 neuronas artificiales que imitaban el cerebro de una rata. Cada neurona hacía el papel de una posición del laberinto y, cuando se activaba, daba a entender que la "rata" sabía en qué punto del laberinto se encontraba. Las neuronas que estaban conectadas alrededor de la activada, hacían la función de alternativas a seguir por el cerebro, la activación de la siguiente neurona, es decir, la elección entre "derecha" o "izquierda" en este caso estaría dada por la fuerza de sus conexiones con la neurona activada.

Para 1957, Frank Rosenblatt inició el desarrollo del Perceptrón, un identificador de patrones ópticos binarios y salida binaria. Las capacidades del Perceptrón se extendieron al desarrollar la *regla de aprendizaje delta*, que permitía emplear señales continuas de entrada y salida [Ros., 59]. Más adelante, en 1959, Bernard Widrow publicó una teoría sobre la adaptación neuronal y dos modelos inspirados en esa teoría: Adaline (Adaptative Linear Neuron) y Madaline (Multiple Adaline) [Wid., 59]. Estos modelos fueron usados en numerosas aplicaciones y permitieron usar, por primera vez, una red neuronal en un problema importante del mundo real: filtros adaptativos para eliminar ecos en las líneas telefónicas.

En 1969, Marvin Minsky y Seymour Papert [Min., 69] realizan la publicación de su libro *Perceptrons: An introduction to Computacional Geometry*. En él, se

realiza una fuerte y detallada crítica del Perceptrón de Rosenblatt. En la publicación se resalta la principal limitación del Perceptrón: la incapacidad para representar la función XOR debido a su naturaleza lineal. Como consecuencia, en la comunidad de investigadores se generaron serias dudas sobre las capacidades de los modelos conexionistas y provocó una caída en el desarrollo de investigaciones.

James Anderson fue uno de los pocos investigadores que continuó con el desarrollo de ingeniería neuronal. En 1977, presentó los estudios realizados con modelos de memorias asociativas, donde destaca el autoasociador lineal conocido como modelo brain-state-in-a-box (BSB) [And., 77].

Teuvo Kohonen continuó el trabajo de Anderson al desarrollar modelos de aprendizaje competitivo basados en el principio de inhibición lateral. Su principal aportación consiste en un procedimiento para conseguir que unidades físicamente adyacentes aprendieran a representar patrones de entrada similares [Koh., 88].

Stephen Grossberg realizó un importante trabajo teórico - matemático tratando de basarse en principios fisiológicos [Gro., 87]. Aportó importantes innovaciones con su modelo ART (Adaptative Resonance Theory) y, junto a Cohen, elaboró un importante teorema sobre la estabilidad de las redes recurrentes en términos de una función de energía.

Gracias a investigaciones como las de Rumelhart y Hopfield, durante los años 80 ocurrió el resurgimiento en el estudio de RNA. En 1982, el físico Jonh Hopfield elaboró un modelo de red basado en unidades de proceso interconectadas, al cual aplicó los principios de estabilidad desarrollados por Grossberg. Como fruto de sus investigaciones escribió dos grandes volúmenes sobre redes neuronales, en los cuales ilustró de manera amplia los mecanismos de almacenamiento y recuperación de la memoria [Hop., 82], lo que revitalizó fuertemente el desarrollo de investigación en RNA, colocándolo para muchos, como padre de la neurocomputación.

Por su parte, en 1986, Rumelhart, McClelland y Hinton crearon el grupo PDP (Parallel Distributed Processing). Como resultado de los trabajos de este grupo, surgieron los manuales: *Parallel Distributed Processing. Explorations in the Microstructure of Cognition* [Rum., 86], y *Explorations in Parallel Distributed Processing. A Handbook of Models. Programs and Exercices* [McC., 88]. En estos manuales, se destacan los capítulos dedicados al algoritmo de retropropagación, que soluciona los problemas planteados por Minsky y Papert, y extienden enormemente el campo de aplicación de los modelos de computación conexionistas, colocándolos de esta forma, en una de las áreas de la inteligencia artificial más ampliamente estudiada en todo el mundo.

3.4.1 EL MODELO BIOLÓGICO

La teoría y modelado de redes neuronales está inspirada en la estructura y funcionamiento del sistema nervioso humano, donde la neurona es el elemento fundamental. Se estima que en cada milímetro del cerebro hay cerca de 50.000 de ellas, conteniendo en total más de cien mil millones de neuronas y sinapsis en el sistema nervioso humano, con conexiones del orden de 10^{15}. La estructura de una neurona se muestra en la Figura 3.4.

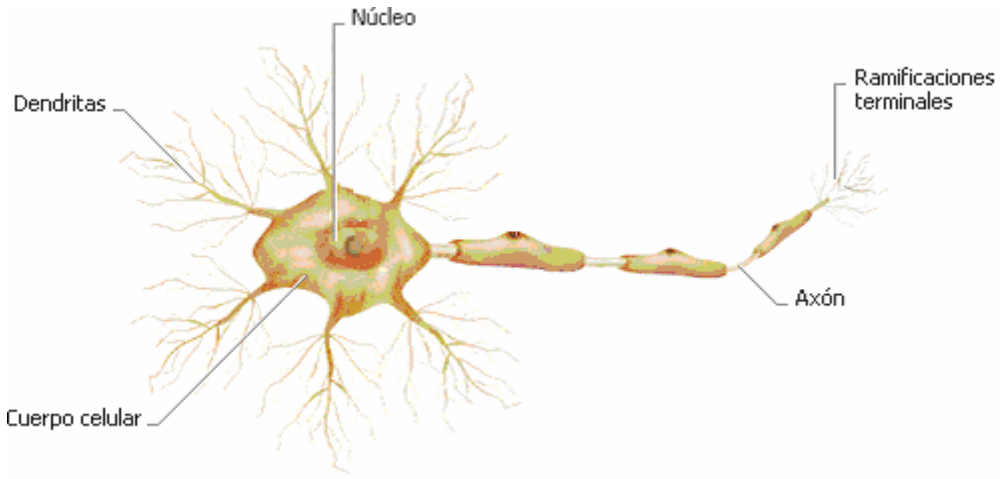


Figura 3.4. La neurona biológica

Una de las principales características de las neuronas es su capacidad de comunicarse. Su tamaño y forma es variable, pero con las mismas subdivisiones que muestra la Figura 3.4 [Hil., 95].

a) El *cuerpo* de la neurona o *soma* contiene el *núcleo*. El cuerpo de una neurona es más o menos esférico, de 5 a 10 micras de diámetro, del que salen una rama principal, el axón y varias ramas más cortas llamadas dendritas. El cuerpo se encarga de todas las actividades metabólicas de la neurona y recibe la información de otras neuronas vecinas a través de las conexiones sinápticas (algunas neuronas se comunican sólo con las cercanas, mientras que otras se conectan con miles), las combina e integra y emite señales de salida.

b) Las *dendritas.* Parten del soma y se encargan de la recepción de señales de otras células a través de conexiones llamadas *sinápticas*. Si pensamos, desde ahora, en términos computacionales podemos decir que las dendritas son las conexiones de entrada de la neurona.

c) El *axón* es la "salida" de la neurona y se utiliza para enviar impulsos o señales a los terminales axónicos de otras neuronas. Cuando el axón está cerca de sus células destino, se divide en muchas ramificaciones que

forman *sinapsis* con el soma o axones de otras células (Figura 3.5). La sinapsis es un proceso químico mediante el cual se realiza la transmisión de una señal de una neurona a otra. Estas señales son de dos tipos: eléctricas y químicas. La señal generada por la neurona y transportada a lo largo del axón es un impulso eléctrico, mientras que la señal que se transmite entre los terminales axónicos de una neurona y las dendritas de la otra es de origen químico. En cualquiera de los casos, el efecto puede ser "inhibidor" o "excitador" de la neurona receptora según el transmisor que las libere. Se estima que cada neurona recibe de 10.000 a 100.000 sinapsis y el axón realiza una cantidad de conexiones similar.

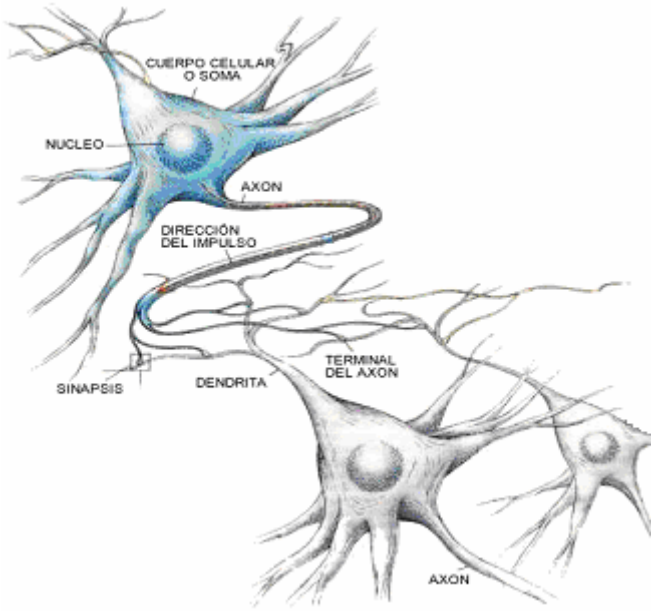


Figura 3.5. Interconexión neuronal

Algunas de las estructuras neuronales son determinadas en el nacimiento, otra parte es desarrollada a través del aprendizaje, proceso en que nuevas conexiones neuronales son realizadas y otras se pierden por completo. Las estructuras neuronales continúan cambiando durante toda la vida, estos cambios consisten en el refuerzo o debilitamiento de las uniones sinápticas.

Este proceso de aprendizaje es uno de los aspectos que se pretende emular con las ampliamente estudiadas redes neuronales artificiales, de las cuales se proporciona una amplia descripción de su implementación en los siguientes apartados.

3.4.2 LA RED NEURONAL ARTIFICIAL

Las redes neuronales, son sistemas de aprendizaje y procesamiento automático que, a través de modelos matemáticos recreados mediante mecanismos artificiales, pretenden imitar a pequeñísima escala la forma de funcionamiento de las neuronas que forman el cerebro humano. Esto hace que ofrezcan numerosas ventajas sobre los sistemas convencionales, entre las cuales podemos destacar:

a) Aprendizaje Adaptativo. Una de las características más atractivas de las redes neuronales es la capacidad de aprender a realizar tareas basadas en un entrenamiento o una experiencia inicial. En el proceso de aprendizaje, los enlaces ponderados de las neuronas se ajustan de manera que se obtengan unos resultados específicos. También existen redes que continúan aprendiendo a lo largo de su vida, después de completado el periodo inicial de entrenamiento.

b) Autoorganización. Las redes neuronales usan su capacidad de aprendizaje adaptativo para organizar la información que reciben durante el aprendizaje y/o la operación. Esta autoorganización permite que las redes neuronales respondan apropiadamente cuando se les presentan datos o situaciones a los que no habían sido expuestas anteriormente.

c) Tolerancia a Fallos. Debido a que las RNA tienen su información distribuida en las conexiones entre neuronas, en caso que se produzca un fallo en algunas de las neuronas, el comportamiento del sistema se ve influenciado, pero no sufre una caída repentina. Esta tolerancia también la encontramos cuado hay problemas como ruido, distorsión de los datos o información incompleta en los datos de entrada (por ejemplo, si la información de entrada es la imagen de un objeto, la respuesta correspondiente no sufre cambios si la imagen cambia un poco su brillo o el objeto cambia ligeramente).

d) Operación en Tiempo Real. Los computadores neuronales pueden ser realizados en paralelo, y se diseñan y fabrican máquinas con hardware especial para obtener esta capacidad.

e) Fácil inserción dentro de la tecnología existente. Debido a que una red puede ser rápidamente entrenada, comprobada, verificada y trasladada a una implementación hardware de bajo costo, es fácil incorporarla en aplicaciones específicas dentro de sistemas existentes (chips, por ejemplo). De esta manera, las redes neuronales se pueden utilizar para mejorar sistemas de forma incremental, y cada paso puede ser evaluado antes de acometer un desarrollo más amplio.

3.4.2.1 Topología de RNA

Existen cuatro aspectos que caracterizan una red neuronal: su topología, el mecanismo de aprendizaje, el tipo de asociación realizado entre la información de entrada y salida, y la forma de representación de esta información.

a) Topología de las RNA. La arquitectura de las redes neuronales consiste en la organización y disposición de las neuronas formando niveles o capas más o menos alejadas de la entrada y salida de la red. Se conoce como capa o nivel a un conjunto de neuronas cuyas entradas provienen de la misma fuente y cuyas salidas se dirigen al mismo destino (ver Figura 3.6).

   En este sentido, los parámetros fundamentales de la red son: el número de capas, el número de neuronas por capa, el grado de conectividad y el tipo de conexiones entre neuronas. Considerando estos parámetros, podemos hacer la siguiente distinción:

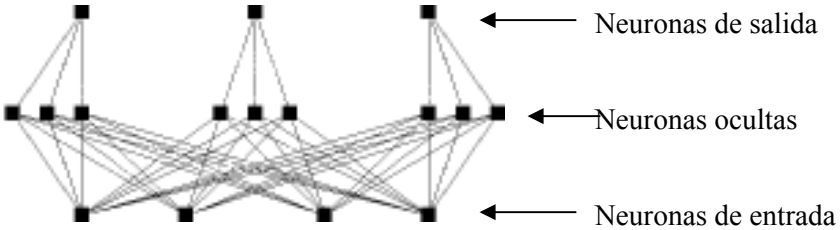


Figura 3.6. Topología de las RNA

Redes Monocapa. Estas redes cuentan con una sola capa de neuronas, que intercambian señales con el exterior estableciendo conexiones laterales, cruzadas o autorrecurrentes (la salida de una neurona se conecta con su propia entrada) y que constituyen a un tiempo la entrada y salida del sistema. Se utilizan en tareas relacionadas con lo que se conoce como autoasociación. Por ejemplo, para generar informaciones de entrada que se presentan a la red incompletas o distorsionadas. Un ejemplo de este tipo de redes es el perceptrón simple.

Redes Multicapa. Son aquellas que disponen de conjuntos de neuronas agrupadas y jerarquizadas en distintos niveles o capas, con al menos una capa de entrada, otra de salida, y, eventualmente una o varias capas intermedias (ocultas). El número de capas se cuenta a menudo a partir del número de capas de pesos (en vez de las capas de neuronas), y el número de capas ocultas está directamente relacionado con las capacidades de la red, aunque se ha demostrado que para la mayoría de problemas bastará con una sola capa oculta [Fun., 1989], [Hor., 89], [Gas., 01]. Hasta estos momentos, no existe evidencia empírica que indique el número óptimo de neuronas de la capa oculta. Algunos autores apuntan a determinar el optimo de neuronas

de esta capa evaluando el rendimiento de diferentes arquitecturas en función de los resultados obtenidos con el grupo de validación [Cot., 95], [Sum., 99], [Sch., 97], o considerando el número de elementos en la capa de entrada [Gas., 01], [Pao., 89], [Sha., 99], y/o número de elementos en la capa de salida [Kan., 97]. Por último, el número de unidades en la capa de salida por lo general está asociado al número de clases en un problema [Gas., 01].

Una forma de distinguir la capa a la que pertenece la neurona consiste en fijarse en el origen de las señales que recibe a la entrada y el destino de la señal de salida. Cuando ninguna salida de las neuronas es entrada de neuronas del mismo nivel o de niveles precedentes, la red se describe como de propagación hacia delante o feedforward, y cuando las salidas pueden ser conectadas como entradas de neuronas de niveles previos o del mismo nivel, incluyendose ellas mismas, la red es de propagación hacia atrás o feedback. Entre las primeras destacan los distintos modelos de Kohonen, el Perceptrón multicapa, las redes Adaline y Madaline, la Memoria Lineal Adaptativa y las Backpropagation. Entre las segundas debemos mencionar el Cognitrón y el Neocognitrón, junto con los modelos de Resonancia y las máquinas multicapa de Boltzman y Cauchy.

Redes modulares *(adaptive mixture of experts)*. Este enfoque, propuesto por Jacobs et al. [Jac., 91], basa su estructura (modular) en la modularidad que tiene el sistema nervioso humano, en el cual cada región cerebral tiene una función especifica, pero a su vez, las regiones se interconectan entre sí. Por ello, se dice que una RNA es modular si la computación realizada por la red puede verse descompuesta en dos o más módulos o subsistemas que trabajan de forma independiente sobre los mismos datos o parte de ellos. Cada uno de estos módulos corresponde a una red neuronal feedforward, y puede ser considerado como neuronas en la red en su conjunto. En su implementación más básica, todos los módulos son de un mismo tipo [Har., 04], [Bau., 04], pero pueden utilizarse esquemas diferentes.

La arquitectura más utilizada es la que cuenta con una capa de módulos de entrada, con un módulo integrador. De tal manera que, cada módulo aporta una solución o parte de ella a un mismo problema, en este caso el módulo integrador determina la solución global a partir de las soluciones individuales de cada red [Jac., 91].

Algunas de las ventajas que estructuras de este tipo tienen sobre modelos múltcapa, son las siguientes:

i. Velocidad de aprendizaje. Si una función compleja se descompone en un conjunto de funciones más simples, una red modular puede implementar dicha descomposición y obtener un aprendizaje más

rápido, en comparación a cuando se utiliza un perceptrón multicapa para aprender la función sin descomponer [Har., 04].

ii. Tratamiento de la información. Las redes modulares son bastante útiles cuando se trabaja con fuentes de información diferentes [Bau., 04], o cuando los datos han sido preprocesados con diferentes técnicas.

iii. Distribución del conocimiento. En una red modular, los módulos de la red tienden a especializarse mediante el aprendizaje de diferentes regiones del espacio de entrada [Har., 04].

b) Mecanismo de Aprendizaje. Una segunda clasificación que se suele hacer es en función del tipo de aprendizaje que requiere la red. Un criterio para diferenciar las reglas de aprendizaje se basa en considerar si la red puede aprender durante su funcionamiento habitual, o si el aprendizaje supone la desconexión de la red.

Otro criterio suele considerar dos tipos de reglas de aprendizaje: las de aprendizaje supervisado y las correspondientes a un aprendizaje no supervisado. Estas reglas dan pie a la siguiente clasificación: redes neuronales con aprendizaje supervisado y redes neuronales con aprendizaje no supervisado. Las primeras requieren de un conjunto de datos de entrada previamente clasificado o cuya respuesta objetivo se conoce. El proceso de aprendizaje se realiza mediante un entrenamiento controlado por un agente externo (supervisor, maestro) que determina la respuesta que debería generar la red a partir de una entrada determinada. El supervisor comprueba la salida de la red y, en el caso de que ésta no coincida con la deseada, se procede a modificar los pesos de las conexiones, con el fin de conseguir que la salida se aproxime a la deseada. Se consideran tres formas de llevar a cabo este tipo de aprendizaje: por corrección, por refuerzo y estocástico. En el aprendizaje por corrección de error, los pesos se ajustan en función de la diferencia entre los valores deseados y los obtenidos en la salida de la red. En el aprendizaje por refuerzo, la función del supervisor se reduce a indicar, mediante una señal de refuerzo, si la salida obtenida en la red se ajusta a la deseada (éxito = +1 o fracaso = -1) y, en función de ello, se ajustan los pesos. Finalmente, en el aprendizaje estocástico, el aprendizaje consiste básicamente en realizar cambios aleatorios en los valores de los pesos de las conexiones de la red y evaluar su efecto a partir del objetivo deseado y de distribuciones de probabilidad. Ejemplos de redes que utilizan aprendizaje supervisado en alguna de sus modalidades son el perceptrón simple, la red Adaline, el perceptrón multicapa y la memoria asociativa bidireccional.

Por otro lado, las redes que utilizan aprendizaje no supervisado no requieren influencia externa para ajustar los pesos de las conexiones entre neuronas. La red no recibe ninguna información por parte del entorno que le

indique si la salida generada es o no correcta, así que existen varias posibilidades en cuanto a la interpretación de la salida de estas redes. En algunos casos, la salida representa el grado de familiaridad o similitud entre la información que se le está presentando en la entrada y las informaciones que se le han mostrado en el pasado. En otros casos, podría realizar una codificación de los datos de entrada, generando a la salida una versión codificada de la entrada, con menos bits, pero manteniendo la información relevante de los datos, o algunas redes con aprendizaje no supervisado que realizan un mapeo de características, obteniéndose en las neuronas de salida una disposición geométrica que representa un mapa topográfico de las características de los datos de entrada, de tal forma que si se presentan a la red informaciones similares, siempre serán afectadas neuronas de salidas próximas entre sí, en la misma zona del mapa.

En general, en este tipo de aprendizaje, se suelen considerar dos tipos: aprendizaje Hebbiano y aprendizaje competitivo o cooperativo. En el aprendizaje Hebbiano, el ajuste de los pesos de las conexiones se realiza de acuerdo con la correlación existente entre neuronas, de tal manera que si las dos neuronas son activas (positivas), se produce un reforzamiento de la conexión. Por el contrario, cuando una es activa y la otra pasiva (negativa), se produce un debilitamiento de la conexión. Por último, en las redes que utilizan un aprendizaje competitivo y cooperativo, las neuronas compiten (y cooperan) unas con otras con el fin de llevar a cabo una tarea dada. Con este tipo de aprendizaje se pretende que, cuando se presente a la red cierta información de entrada, sólo una de las neuronas de salida se active (alcance su valor de respuesta máximo). Ejemplos de redes que utilizan aprendizaje no supervisado son las memorias asociativas, las redes de Hopfield, la máquina de Boltzman y la máquina de Cuchy, las redes de Kohonen y las redes de resonancia adaptativa (ART).

Finalmente, existen redes que utilizan un enfoque mixto de aprendizaje (híbridas), en el que se utiliza una función de mejora para facilitar la convergencia. Un ejemplo de este último tipo son las redes de base radial (Radial Basis Function).

c) Tipo de asociación entre la información de entrada y salida. Las RNA son sistemas que almacenan cierta información aprendida; está información se registra de forma distribuida en los pesos asociados a las conexiones entre neuronas de entrada y salida. Existen dos formas primarias de realizar esa asociación de entrada/salida. Una primera sería la denominada hetero-asociación, que se refiere al caso en el que la red aprende parejas de datos $[(A_1, B_1), (A_2, B_2),\ldots, (A_n, B_n)]$, de tal forma que cuando se presente cierta información de entrada A_i, deberá responder generando la correspondiente salida B_i. La segunda se conoce como auto-asociación, donde la red aprende ciertas informaciones $A_1, A_2 \ldots A_n$, de tal forma que cuando se le presenta

una información de entrada, realizará una auto-correlación, respondiendo con uno de los datos almacenados, el más parecido al de la entrada.

Estos dos mecanismos de asociación dan lugar a dos tipos de redes neuronales: las redes hetero-asociativas y las auto-asociativas. Una red hetero-asociativa podría considerarse aquella que computa cierta función, que en la mayoría de los casos no podrá expresarse analíticamente, entre un conjunto de entradas y un conjunto de salidas, correspondiendo a cada posible entrada una determinada salida. Existen redes hetero-asociativas con conexiones feedforward, feedforward/feedback y redes con conexiones laterales. También existen redes hetero-asociativas multidimensionales y su aprendizaje puede ser supervisado o no supervisado.

Por otra parte, una red auto-asociativa es una red cuya principal misión es reconstruir una determinada información de entrada que se presenta incompleta o distorsionada (le asocia el dato almacenado más parecido). Pueden implementarse con una sola capa, existen conexiones laterales o también auto-recurrentes, y habitualmente son de aprendizaje no supervisado.

d) Representación de la información de entrada y salida. En esta categoría se distinguen tres tipos de redes: analógicas, discretas (generalmente, binarias) e híbridas. Las *redes analógicas* procesan datos de entrada de naturaleza analógica, valores reales continuos, habitualmente acotados y usualmente en el rango [-1,1] o en el [0,1], para dar respuestas también continuas. Las redes analógicas suelen presentar funciones de activación continuas, habitualmente lineales o sigmoides. Entre estas redes neuronales destacan las redes de Backpropagation, la red continua de Hopfield, la de Contrapropagación, la Memoria Lineal Asociativa, la Brain-State-in-Box, y los modelos de Kohonen, (SOM. y LVQ.). Las *redes discretas* (binarias) procesan datos de naturaleza discreta, habitualmente valores lógicos booleanos {0,1}, para acabar emitiendo una respuesta discreta. En este caso, las funciones de activación de las neuronas son de tipo escalón. Entre las redes binarias destacan la máquina de Boltzman, la máquina de Cauchy, la red discreta de Hopfield, el Cognitrón y el Neocognitrón. Finalmente, las *redes híbridas*, procesan entradas analógicas para dar respuestas binarias; entre ellas destacan el Perceptrón, la red Adaline y la Madaline.

3.4.2.2 Funcionamiento

La Figura 3.7 muestra un esquema conceptual de la neurona artificial. Como vemos, la neurona recibe una serie de entradas a través de interconexiones afectadas por un peso *w* y emite una salida. Esta salida viene dada por dos funciones:

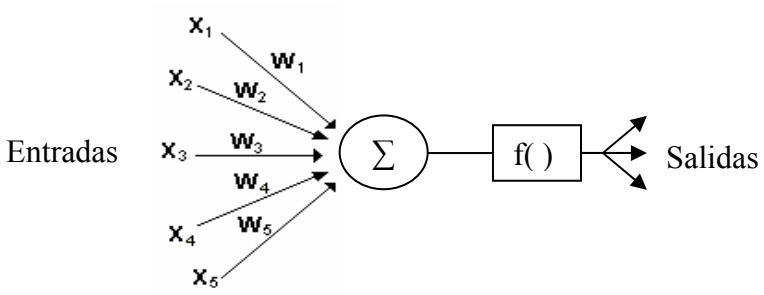


Figura 3.7. Neurona artificial

a) Una función de propagación (también conocida como función de excitación), que por lo general consiste en la sumatoria de cada entrada multiplicada por el peso de su interconexión. Las señales que llegan a la sinapsis son las entradas a la neurona; éstas son ponderadas de acuerdo a la sinapsis correspondiente. Se considera que el efecto de cada señal es aditivo, de tal forma que la entrada neta que recibe una neurona es la suma del producto de cada señal de entrada por el valor de la sinapsis que conecta ambas neuronas, comúnmente conocido como red de propagación.

b) Una función de transferencia para la activación o salida de la neurona. Esta función está asociada con cada neurona, de tal manera que para cada neurona hay una función de salida que transforma el estado actual de activación en una señal de salida. De este modo, las señales de entrada pueden excitar a la neurona (sinapsis con peso positivo) o inhibirla (peso negativo). El efecto es la suma de las entradas ponderadas. Si la suma es igual o mayor que el umbral de la neurona, indica que la relación entre las neuronas es excitadora y, en este caso, la neurona se activa (da salida). Si la suma es menor que el umbral, la sinapsis será inhibidora. En este caso, si la neurona está activada, se desactivará. Finalmente, si la suma es 0, se supone que no hay conexión entre ambas neuronas. Esta es una situación de todo o nada: cada neurona se activa o no se activa. La función de transferencia puede ser *lineal* (Figura 3.8 (a)) o de función escalón (Figura 3.8 (b)) o de función sigmoidal (logística o tangente hiperbólica, Figura 3.8 (c)), entre otras. La variable **f** es la frecuencia de activación o emisión de potenciales y **u** es la intensidad del estímulo.

   Es importante tener en cuenta que para aprovechar la capacidad de las RNA de aprender relaciones complejas o no lineales entre variables, es necesario utilizar funciones no lineales al menos en las neuronas de la capa oculta [Hil., 95]. Por su parte, la elección de la función de activación en las neuronas de la capa de salida depende del tipo de actividad que realicen. Por ejemplo, en tareas de clasificación, las neuronas normalmente toman la función de activación sigmoidal. Así, cuando se presenta un patrón que pertenece a una categoría particular, los valores de salida tienden a dar

como valor 1 para la neurona de salida que representa la categoría de pertenencia del patrón, y 0 ó -1 para las otras neuronas de salida.

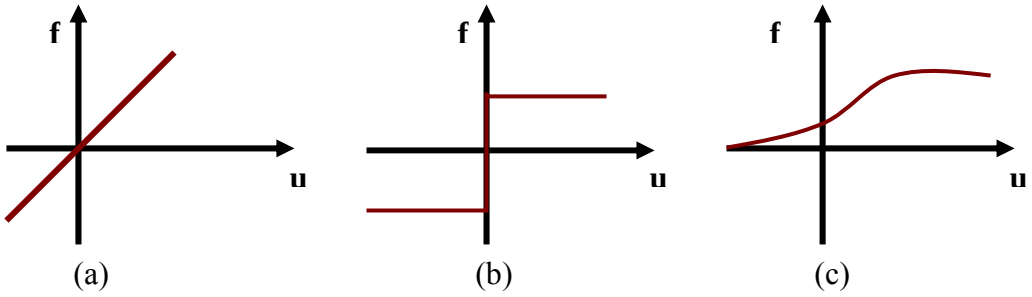


Figura 3.8. Función de transferencia o activación

3.4.2.3 Aprendizaje y entrenamiento

Al igual que en el sistema biológico, el aprendizaje de la RNA puede ser comprendido como la modificación del comportamiento producido por la interacción con el entorno y como resultado de experiencias, conduce al establecimiento de nuevos modelos de respuesta a estímulos externos. Por ejemplo, cuando el sistema humano mediante los ojos capta un objeto A, algunos de los sensores de la visión se activan y envían señales a las neuronas ocultas (aumentan el grado de conexión de ellas). Si el mismo objeto A se presenta una y otra vez, la interconexión de neuronas se refuerza y, por lo tanto, el conocimiento del objeto A. Si tiempo después se presenta nuevamente el objeto A modificado, la unión de las neuronas para el conocimiento de tal objeto es débil. Por tal motivo, las neuronas deben entrenarse para reconocer el objeto A en esta nueva presentación. Luego de algunas sesiones de entrenamiento, el sistema neuronal es capaz de reconocer el objeto A en todas sus formas. Si el objeto A cambia nuevamente, el conocimiento se actualiza.

Algo similar sucede cuando se entrena una RNA. Para que se de el aprendizaje, se parte de un conjunto de datos de entrada suficientemente significativo para conseguir que la red *aprenda* automáticamente las propiedades deseadas. Durante este proceso, los parámetros de la red se adecuan a la resolución del problema, realizando ajustes para las conexiones sinópticas (pesos) existentes entre las neuronas. Los cambios que se producen durante el proceso de aprendizaje se reducen a la destrucción, modificación y creación de conexiones entre las neuronas. La creación de una nueva conexión implica que el peso de la misma pasa a tener un valor distinto de cero, una conexión se destruye cuando su peso pasa a ser cero. Se puede afirmar que el proceso de aprendizaje ha finalizado (la red ha aprendido) cuando los valores de los pesos permanecen estables.

Por otro lado, las modificaciones de los pesos pueden realizarse de dos formas: después de haber presentado todos los patrones de entrenamiento (aprendizaje por lotes o modo *batch*) o actualizar los pesos tras la presentación de cada patrón de entrenamiento (aprendizaje en serie o modo *on line*). En este último modo, es importante observar que la presentación de los patrones sea de forma aleatoria, puesto que si siempre se sigue un mismo orden, el entrenamiento estaría viciado a favor del último patrón del conjunto de entrenamiento, cuya actualización, por ser la última, siempre predominaría sobre las anteriores.

Este entrenamiento, repetido para todos los valores de entrada y salida que se quiera, origina una representación interna del objeto en la red, que considera todas las irregularidades y generalidades del mismo. Por ejemplo, se desea entrenar una red que se va a aplicar al diagnóstico de imágenes médicas. Durante la fase de entrenamiento el sistema recibe imágenes de tejidos que se sabe son cancerígenos y tejidos que se sabe son sanos, así como las respectivas clasificaciones de dichas imágenes. Si el entrenamiento es el adecuado, una vez concluido, el sistema podrá recibir imágenes de tejidos no clasificados y obtener su clasificación *sano*/*no sano* con un buen grado de seguridad. En la Figura 3.9, se presenta el esquema de una neurona artificial durante la etapa de aprendizaje. Las variables de entrada pueden ser desde los puntos individuales de cada imagen hasta un vector de características de las mismas (por ejemplo, procedencia anatómica del tejido de la imagen o la edad del paciente al que se le extrajo la muestra).

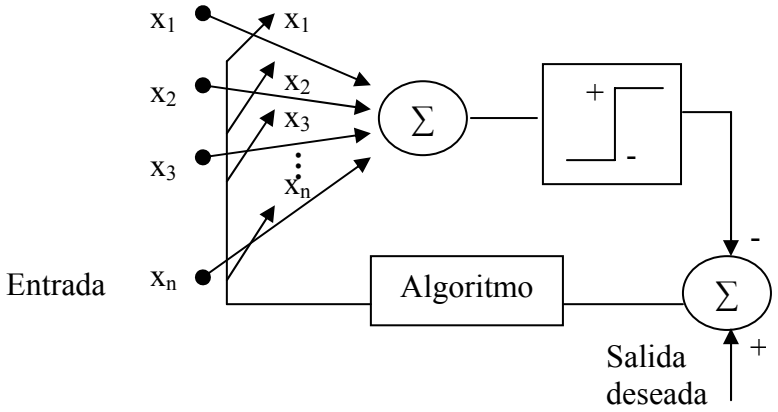


Figura 3.9. Aprendizaje de una neurona artificial

3.4.2.4 Evaluación del rendimiento del modelo

Una vez terminado el proceso de entrenamiento, es necesario evaluar la capacidad de generalización que tiene la red de una forma completamente objetiva a partir de un grupo de datos independiente, el conjunto de test o prueba.

En el proceso de aprendizaje la evaluación consiste en la estimación de una función, normalmente la media cuadrática del error. En tanto que, para evaluar el rendimiento de la red, lo más común es basarse en la frecuencia de clasificaciones correctas e incorrectas. A partir del valor de las frecuencias, se puede construir una tabla de confusión para calcular los diferentes índices de asociación, y el acuerdo entre el criterio y la decisión tomada por la red neuronal. Por último, cuando estamos interesados en discriminar entre dos categorías, se recomienda hacer uso de los índices de sensibilidad, especificidad y eficacia, y del análisis de curvas ROC (*Receiver operating characteristic*) [Ega., 75]

3.5 TÉCNICAS DE PREPROCESADO DEL CONJUNTO DE ENTRENAMIENTO

Pese a que la integración del conjunto de entrenamiento se realiza con el apoyo del experto humano, éste tiene serios problemas que complican y obstaculizan el proceso de clasificación y disminuyen los índices de precisión al clasificar nuevos patrones. El caso ideal para que la clasificación se realice exitosamente sería que los patrones de una misma clase se encuentren lo más cercanos posible entre ellos y lo más alejados a los patrones de otras clases, para que de esta forma se distinga claramente la distribución de las clases en el espacio de representación, favoreciendo los niveles de precisión (ver Figura 3.10). Sin embargo, en problemas reales, los agrupamientos presentes no siempre siguen esta forma ideal.

De entre los principales factores que deterioran la calidad de la clasificación relacionados con el CE, es posible mencionar los siguientes: patrones con atributos poco discriminantes (solapamiento entre clases), presencia de patrones atípicos o ruidosos, CE resultantes cuyo tamaño requiere de una gran cantidad de memoria para su procesamiento en conjunto, que incluso en ocasiones tales requerimientos sobrepasan la capacidad del equipo en uso y, por último, la obtención de CE no balanceados.

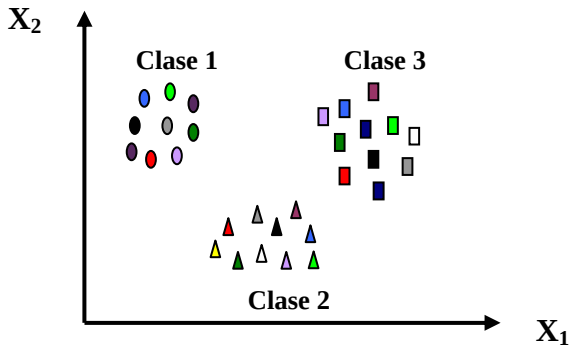


Figura 3.10. Caso ideal de distribución de clases en un espacio bidimensional

3.5.1 PATRONES ATÍPICOS O RUIDOSOS

Un problema derivado, no sólo de una mala construcción del CE, sino de la naturaleza del problema mismo, es la presencia de patrones atípicos o ruidosos poco deseables en el CE. Los patrones atípicos son aquellos que, a pesar de pertenecer a una clase determinada, son significativamente diferentes al resto de los patrones de su misma clase. Por su parte, un patrón ruidoso es aquel que puede confundir al clasificador debido a que guarda cierto parecido con objetos de otras clases. Un objeto atípico puede ser producido por errores (procesamiento, captura, etc.), pero también por nuevos patrones que no pertenezcan a ninguna clase representada en el CE, o simplemente por patrones mal etiquetados.

3.5.2 SOLAPAMIENTO ENTRE CLASES

En la Figura 3.11, se muestra la representación de un conjunto de patrones pertenecientes a tres clases, dos de las cuales se encuentran solapadas. Generalmente, esta situación se presenta cuando algunos de los patrones contenidos en ambas clases comparten información en común en algunos de sus atributos. Debido a esta situación, la discriminación de las clases con estos patrones no resulta fácil de realizar.

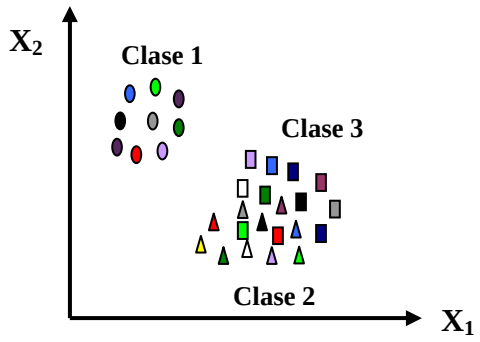


Figura 3.11. Dos clases solapadas en un espacio bidimensional

3.5.3 DESBALANCE ENTRE CLASES

La mayoría de los sistemas que emplean aprendizaje supervisado fueron creados asumiendo que el CE está bien balanceado, es decir, que la representación de patrones por clase es muy pareja. Desgraciadamente, esta suposición frecuentemente no es aplicable en el mundo real, ya que existen dominios en los cuales el CE cuenta con desbalance: una o varias de las clases (minoritarias) está menos representada con respecto al número de patrones pertenecientes a otras clases (mayoritarias). Este problema ocurre en aplicaciones donde el clasificador detecta un evento poco frecuente, pero quizás sumamente importante. Algunos ejemplos pueden encontrarse en la detección de fraude en tarjetas de crédito,

diagnóstico médico, clasificación de textos y procesos de mercadotecnia, entre otros. Para ejemplificar esta situación, consideremos un CE con dos clases (Figura 3.12), la primer clase cuenta con 4 patrones (minoritaria) y la segunda clase con 30 patrones (mayoritaria).

El problema se complica cuando se trata de CE con más de dos clases, pues resulta difícil determinar qué clases serán consideradas como minoritarias y cuáles como mayoritarias (Figura 3.13). Se ha comprobado que esta situación puede deteriorar de forma importante la precisión de la clasificación, en particular con los patrones que pertenecen a la clase menos representada [Bar., 01a].

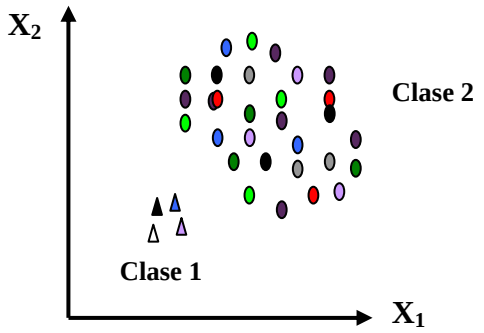


Figura 3.12. Presencia de desbalance en un caso de dos clases

La mayoría de las investigaciones que han desarrollado metodologías para lograr el adecuado tratamiento y la disminución de los efectos nocivos que los CE desbalanceados presentan al momento de la clasificación, se enfocan a corregir el desequilibrio de la cantidad de patrones, con tres vertientes básicamente [Jap., 00]: under-sampling (eliminando patrones) en la clase mayoritaria, over-sampling (replicando patrones) en la clase minoritaria y, por último, internamente predisponer el proceso de discriminación para compensar el desequilibrio del CE.

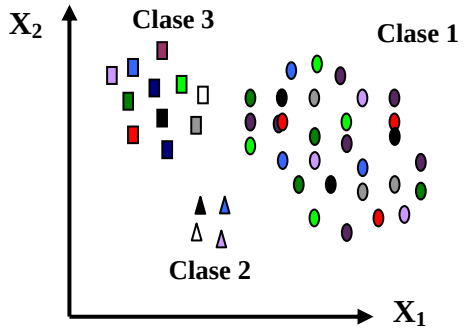


Figura 3.13. Presencia de desbalance en un caso de tres clases. La clase 2 es considerada minoritaria respecto a la clase 1 y 3, mientras que la clase 3 es minoritaria respecto a la clase 1

En el primer enfoque, podemos encontrar que el decremento de la clase mayoritaria se ha realizado de dos formas. La primera de ellas es utilizando el principio de aleatoriedad y la segunda realizando la eliminación de patrones de forma arbitraria, centrándose en aquellos patrones que pudieran proporcionar cierto grado de deterioro en la clasificación. Dentro de este grupo, se pueden mencionar aquellas investigaciones en las que se propone la ejecución de algoritmos de preprocesado a una o varias clases del CE, destinados a reducir el tamaño del conjunto de datos mediante la utilización de una técnica propuesta en 1972 por Wilson, que consiste en la eliminación de aquellos registros que tienen características distintas a las del resto de los patrones de su clase (patrones atípicos) presentes en la frontera de decisión [Wil., 72]. Una variante de esta técnica es la propuesta por Tomek en 1976, en la que propone la múltiple ejecución del algoritmo de Wilson para lograr una reducción más representativa que cuando éste se aplica una sóla vez [Tom., 76]. Otro algoritmo de preprocesado utilizado para estos fines es la integración de un subconjunto representativo del total de los patrones contenidos en el CE [Bar., 05a], con la finalidad de realizar menos cálculos, eliminando aquellos patrones que se consideran inútiles o innecesarios. Finalmente, se ha acudido a algoritmos evolutivos, los cuales engloban una serie de técnicas que están inspiradas en los procesos biológicos de la selección natural. Barandela et al. [Bar., 05b] proponen la reducción de los patrones de la clase mayoritaria mediante principios de algoritmos genéticos, buscando balancear el número de patrones por clase en la solución inicial, tomando para ello todos los patrones de la clase minoritaria y reduciendo el número de patrones de la clase mayoritaria hasta alcanzar la cardinalidad de la clase minoritaria.

Dentro del segundo enfoque, se encuentran las investigaciones que realizan el balance en función del tamaño de la clase mayoritaria, para lo que se incluyen nuevos patrones a la clase menos representada, ya sea mediante la duplicidad de los ya existentes o la creación de nuevos patrones. En este último caso, Chawla et al. proponen el algoritmo SMOTE (Syntetic Minority Over-sampling Technique) en el que muestran que la realización de sobre-entrenamiento de la clase minoritaria, al incluir patrones *sintéticos*, fortalece la región de decisión de la clase minoritaria y obtiene un buen desempeño del clasificador, en este caso, un árbol C4.5, en comparación con situaciones donde se eliminan aleatoriamente patrones de la clase mayoritaria para disminuir su tamaño [Cha., 00]. Para generar cada patrón sintético, considera los *k* vecinos a encontrar y el porcentaje de incremento deseado. Algo similar realiza Domingos [Dom., 99] al expandir el espacio de decisiones con la creación de nuevos patrones, mediante la utilización de *metacostos.*

3.5.4 TAMAÑO EXCESIVO

Nos encontramos en la denominada “era de la información”, donde muchas bases de datos comerciales, transaccionales y científicas crecen a un ritmo extraordinario.

Como ejemplo de ello, podemos considerar el proyecto SKY CAT, en el cual se obtienen imágenes satelitales del espacio exterior, y donde el grueso de información almacenada diariamente se estima que es del orden de un *petabyte* (10^{15} *byte*s) [Fay., 96]. Otros sistemas, como las transacciones realizadas en un supermercado virtual, cabinas de aerolíneas, operaciones de tarjetas de crédito y otras, son susceptibles de generar un gran volumen de datos, que no está lejos de exceder los límites superiores de análisis considerados hasta nuestros días.

Actualmente, se desarrollan proyectos que abordan este problema. Sin embargo, de los avances que se han obtenido, la mayoría están orientados a la obtención de grandes medios de almacenamiento de la información, en tanto que el adecuado análisis de ésta, aún se encuentra en fase de estudio.

Históricamente, el desarrollo de la estadística ha proporcionado métodos para analizar datos y encontrar correlaciones y dependencias entre ellos. Recientemente, han surgido nuevos métodos, principalmente de aprendizaje y representación del conocimiento, desarrollados por la comunidad de inteligencia artificial, estadística y física de dinámicas no lineales [Dec., 95]. Estos métodos complementan a las tradicionales técnicas estadísticas, en el sentido de que son capaces de inducir relaciones cualitativas generales, o *leye*s, previamente desconocidas.

Como parte de estas alternativas de solución, se encuentran los métodos denominadas en la literatura como *escalabilidad de algoritmos* [Bar., 03a], [Val., 02b] y métodos de reducción del tamaño del conjunto de datos [Har., 68], [Wil., 72], [Bar., 01a]. Los primeros de estos métodos incluyen el estudio, desarrollo e implementación de metodologías que permitan procesar eficientemente grandes volúmenes de datos, mediante la combinación de algoritmos ya existentes. Los segundos contemplan la ejecución de algoritmos de preprocesado, destinados a reducir el tamaño del conjunto de datos con la finalidad de disminuir los requerimientos de memoria y los costos computacionales. Esta reducción se realiza de dos formas: eliminando patrones (atípicos) que tienen características distintas a las del resto de su clase, e integrando un subconjunto representativo del CE, donde se eliminan aquellos patrones que se consideran poco "útiles" o redundantes.

3.5.5 LIMPIEZA DEL CONJUNTO DE ENTRENAMIENTO

Como cualquier clasificador no paramétrico, la NNR se ve muy afectada por la presencia de patrones atípicos o ruidosos en la frontera de decisión [San., 03]. Con la finalidad de incrementar la calidad del CE mediante la detección y eliminación de estos patrones, se han propuesto varios algoritmos, siendo uno de los más ampliamente usados la edición de Wilson [Wil., 72].

3.5.5.1 Edición de Wilson

En 1972, surge una de las primeras propuestas que considera la reducción del tamaño del CE original, mediante la eliminación de los patrones mal clasificados al emplear la regla *k*-NN. La mecánica que se propone consiste en el análisis de cada patrón contenido en el CE, se observan las características de sus *k* vecinos y, si la etiqueta de la mayoría de estos vecinos no coincide con la del patrón analizado, se concluye que éste es atípico y, por consiguiente, es marcado para ser eliminado. Una vez marcados los patrones que cumplen la condición, el procedimiento continúa con la eliminación de todos aquellos patrones que hayan sido marcados para ser borrados, dejando una nueva muestra integrada por M – {patrones marcados} (ver Figura 3.14).

Entradas: M = conjunto de *m* patrones, $\{x_i \mid i = 1, 2,...m\}$.
k = Número de vecinos a considerar
Salidas: CEE = conjunto de datos editado.

Método
Para todo i = 1 hasta i = *m* **hacer**
Buscar los k vecinos de x_i en M – $\{x_i\}$
Determinar la clase mayoritaria de los k vecinos
Si clase mayoritaria ≠ etiqueta x_i **entonces**
Marcar x_i como atípico
Fin si
Fin para todo
CEE = M – {los patrones marcados como atípicos}
Fin método

Figura 3.14. Algoritmo de edición de Wilson

Esta técnica es ampliamente recomendada en aquellos casos donde se encuentren clases solapadas, ya que se ha demostrado que la mayor parte de los patrones que son eliminados (atípicos) corresponden a aquellos patrones que se encuentran en la región de solape entre dos o más clases.

3.5.5.2 Edición de Wilson con distancia ponderada

Con la intención de disminuir los efectos negativos al aplicar el algoritmo de Edición de Wilson en un conjunto de datos que presenta desbalance, se propuso la incorporación de una ponderación a la distancia Euclídea, a fin de evitar la eliminación de patrones útiles. La distancia ponderada se define del siguiente modo [Bar., 03b]:

$$d_w(\mathbf{y}, x_0) = (m_i/m)^{1/n}\, d_E(\mathbf{y}, x_0)$$

donde x_0 es un patrón que representa a la clase i, m_i es el número de patrones de la clase i, *m* es el total de patrones de entrenamiento, d_E es la distancia Euclídea, **y** es el patrón de prueba n-dimensional.

Obsérvese que con la utilización de un factor de ponderación mayor para los patrones de la clase mayoritaria y menor para los patrones de la minoritaria, se tiene mayor tendencia a buscar el vecino más cercano de un patrón de prueba entre los patrones de la clase minoritaria.

3.5.6 DISMINUCIÓN DEL TAMAÑO DEL CONJUNTO DE ENTRENAMIENTO

El método común para realizar la búsqueda del vecino más cercano consiste en recorrer el total de los patrones de entrenamiento. Para cada patrón de entrenamiento, se calcula la distancia al patrón de entrada y se guarda aquel que es más cercano hasta el momento.

El coste en número de distancias calculadas es proporcional al tamaño del CE. En muchos problemas en los que se aplica la técnica de búsqueda de el (los) vecino(s) más cercano(s), este método resulta prohibitivo debido a que el coste inherente al cálculo de las distancias puede ser demasiado elevado. Para aminorar este problema, varios investigadores han propuesto la integración de un subconjunto de datos representativo del CE para reducir el número de patrones en ella y, por consiguiente, de las distancias a calcular en la fase de búsqueda.

Entre todos los métodos propuestos existen algunos que necesitan una adecuada representación de los patrones en un espacio vectorial, y dependen directamente del orden en el que se encuentran los patrones en el CE, o requieren de una medida de distancia entre los patrones. Sin embargo, existen otros que son especialmente interesantes porque sólo hacen uso de la medida de distancia, sin importar el orden de los datos; esto hace que sean aplicables a una mayor variedad de problemas prácticos.

3.5.6.1 Regla del vecino más cercano condensada

Este enfoque, propuesto por Hart en 1968 [Har., 68], es uno de los más utilizados. Hart realizó experimentos con datos no reales, de los cuales al finalizar su investigación obtuvo resultados en la reducción de la memoria requerida y obtuvo un incremento en la velocidad de búsquedas al clasificar nuevos patrones. La idea básica de este método consiste en la integración de un subconjunto de patrones a partir de los patrones de entrenamiento, de tal forma que este subconjunto sea consistente con el CE original y, al momento de ser utilizado, tenga un rendimiento

adecuado. Dicho de otra forma, todos los patrones de entrenamiento originales serán clasificados correctamente por el subconjunto consistente (SC) cuando se aplique la regla NN, en el cual para cada patrón contenido en el CE, su vecino más cercano en el SC tiene su misma etiqueta.

La integración de este SC se realiza mediante la eliminación de patrones redundantes o patrones contenidos en el CE que no aportan nada a la capacidad discriminatoria del algoritmo de clasificación (Figura 3.15). Al hacer esta eliminación, se logra disminuir los tiempos de búsqueda y los requerimientos de memoria al momento de aplicar el algoritmo de clasificación por el vecino más cercano.

Entradas: M = CE de *m* patrones, $\{x_i \mid i = 1, 2,..., m\}$.
Salidas: SC = Subconjunto consistente de *p* patrones.

Método
Pasar un patrón contenido en M al SC
$p = 1$; $i = 2$;
Repetir
Para todo j = 1 hasta j = *p* **hacer**
Encontrar vecino más cercano de x_i en SC
Si etiqueta de vecino más cercano ≠ etiqueta x_i **entonces**
$M = M - \{x_i\}$
$SC = SC + \{x_i\}$
$p = p + 1$;
Fin si
$i = i + 1$;
Hasta (M = 0) o (no más reemplazos)
Fin método

Figura 3.15. Regla del vecino más cercano condensada

Hasta nuestros días, esta idea se trata de perfeccionar combatiendo sus principales deficiencias: resultados dependientes del orden en que se analizan los patrones de entrenamiento, no conserva todos los patrones de entrenamiento cercanos a la frontera y mantiene patrones de entrenamiento innecesarios (el subconjunto obtenido no es el mínimo posible).

3.5.6.2 Subconjunto selectivo modificado (SSM)

Este método de reducción realiza la integración de un subconjunto de patrones más pequeño que el tamaño del CE, además de garantizar una mejor aproximación a las fronteras de decisión por mantener los patrones que se encuentran cercanos a las fronteras entre clases.

El proceso general toma en cuenta los *vecinos relacionados* R_i de cada patrón x_i contenido en el CE, donde $\{R_i\}$ es el conjunto de todos los y_i vecinos relacionados al patrón de entrenamiento x_i, de tal forma que y_i es de la misma clase que x_i y es más cercano a x_i que su vecino más cercano en el CE de una clase diferente [Bar., 05a].

A fin de ejemplificar más claramente la metodología, se considera un caso de dos clases, clase 1 y clase 2 (ver Figura 3.16). Para realizar la integración al SSM de varias clases el proceso es similar. Previo a la integración de SSM se consideran los siguientes aspectos:

```
Inicio
    1. Pasar x1 al SSM KN = m1 –1.
    2. Para todo i = 2, ... m1.
          Si d(x1, xi) < di entonces
             Kj = 0; KN = KN – 1
          Sino
             Ki = 1
       Fin para
    3. Para todo i = 2,…, m1.
          IND = 0
          Si KN = 0 entonces
             Terminar
          Sino
             Si Ki = 1 entonces
                   Ki = 0;
                   KN = KN – 1;
                   pasar xi al SSM;
                   IND = 1;
             Fin si
             Para todo j = i + 1,..., m1
                   Si (Kj =1 & d(xi, xj) < dj) entonces
                         Kj = 0;
                         KN = KN – 1;
                         Si IND = 0 entonces
                               Kj = 0;
                               IND = 1;
                               KN = KN – 1;
                               Pasar xi al SSM;
                         Fin si
                   Fin si
             Fin para
          Fin sino
    Fin para
Fin
```

Figura 3.16. Algoritmo del Subconjunto Selectivo Modificado

a) m_1 es el número de patrones de la clase 1.

b) IND es una bandera que impide la duplicidad de patrones en el SSM.

c) Para cada x_i, se obtiene su vecino más cercano de la clase 2 y se almacena su distancia en d_i.

d) Todas las distancias son ordenadas de menor a mayor, $d_1 < d_2 < d_3 \ldots d_m$.

e) KN indica el número de patrones que aún no están representados en el SSM. Cuando KN = 0, el algoritmo termina (todos los patrones ya están representados en el SSM).

Cuando hay dos o más distancias iguales, la decisión sobre qué patrón se incluirá primero en el SSM se realiza en base al patrón que resulte ser vecino relacionado para un número mayor de patrones de entrenamiento y, en caso que esta situación también sea de empate, entonces la selección se realiza de manera aleatoria. Esa condición hace el algoritmo un poco más complejo desde el punto de vista computacional, pero asegura la singularidad de la solución y también tendrá la ventaja de propiciar una mayor reducción del tamaño del SSM que el obtenido con el algoritmo de Hart.

Capítulo 4

Sistemas Múltiples de Clasificación

A partir de 1990, surgió una nueva metodología de clasificación que rompió con el esquema tradicional de clasificar nuevos patrones utilizando un solo clasificador. En la actualidad, los Sistemas Múltiple de Clasificación (SMC) son conocidos con nombres tan variados como ensembles, comité de aprendizaje, mezcla de expertos, etc., y son tan populares que en nuestros días se han convertido en una de las más prometedoras líneas de investigación en reconocimiento de patrones [Die., 97].

Estos sistemas surgen con el principal propósito de aumentar la precisión en la clasificación de patrones que hasta ahora se ha obtenido con la utilización de un algoritmo único. La idea fundamental [Die., 97] considera la utilización de un grupo de clasificadores $D = \{ D_1,..., D_H\}$, donde cada clasificador tendrá como entrada un vector de atributos $\mathbf{y} \in \mathbb{R}^n$, al cual asignan una etiqueta de clase $D_t(\mathbf{y})$, donde $t = 1,...,c$. En la fase final, las decisiones individuales del SMC son combinadas mediante algún esquema de toma de decisiones para determinar la clase definitiva que se asigna al patrón **y**.

Algunos aspectos que estos sistemas pretenden superar y que están presentes al utilizar un clasificador único son [Kun., 01c]: la decisión combinada toma ventaja sobre las decisiones individuales de cada clasificador, los errores correlacionados de los componentes individuales pueden ser eliminados cuando se considera el total de las decisiones, los patrones de entrenamiento pueden no proporcionar información suficiente para seleccionar el mejor clasificador, el algoritmo de aprendizaje puede no ser adaptado para resolver el problema y, finalmente, el espacio individual de búsqueda puede no contener la función objetivo. Esto puede ser analizado más detalladamente desde un punto de vista estadístico, computacional y representacional [Die., 97]:

a) Estadística. Un algoritmo de aprendizaje puede verse como la búsqueda de un espacio H de decisiones, en el que se intenta identificar la mejor decisión en el espacio H-dimensional. El problema estadístico se presenta cuando la cantidad de patrones de entrenamiento es demasiado pequeña, en comparación con el tamaño del espacio de decisiones.

b) Computacional. Muchos algoritmos de aprendizaje trabajan buscando mínimos globales. Por ejemplo, el algoritmo de redes neuronales emplea gradiente por descenso para minimizar el error de una función sobre los patrones de entrenamiento. En casos donde el conjunto de datos es muy grande, todavía puede ser muy difícil para el algoritmo de aprendizaje (desde el punto de vista computacional) encontrar la mejor decisión. Un SMC construido para realizar la búsqueda de diferentes puntos locales de inicio puede proporcionar una mejor aproximación a la decisión correcta aún desconocida, que cualquiera de los clasificadores individuales.

c) Representacional. En la mayoría de las aplicaciones de aprendizaje automático, la decisión correcta puede no ser representada por ninguna de las decisiones individuales. Mediante la suma de las ponderaciones de las decisiones individuales, puede ser posible extender el espacio de funciones representables. Este problema es algo sutil porque, para muchos algoritmos de aprendizaje, el espacio de decisiones es, en principio, el espacio de todos los posibles clasificadores individuales. Si se tiene un CE con un gran número de patrones, estos algoritmos explorarán el espacio de todos los posibles clasificadores individuales. No obstante, con CE finitos, estos algoritmos explorarán sólo un conjunto finito de decisiones, teniendo como condición de parada cuando se encuentre una decisión que ajuste los datos de entrenamiento.

La utilización de un SMC promete reducir (y posiblemente eliminar) la mayor parte de estas limitaciones de los algoritmos de aprendizaje clásicos.

4.1 MÉTODOS PARA LA CONSTRUCCIÓN DE SMC

En la construcción de un SMC es necesario considerar dos aspectos fundamentales: la diversidad en las decisiones individuales y la precisión de los componentes individuales. Los métodos empleados para lograr diversidad pueden ser descritos en cinco grandes grupos [Die., 97]: manipulación de patrones, manipulación de atributos, manipulación de etiquetas, utilización de diferentes algoritmos de clasificación y utilización de la aleatoriedad. En los siguientes apartados, trataremos cada una de estas técnicas.

4.1.1 MANIPULACIÓN DE LOS PATRONES

Mediante estos métodos, se realiza la manipulación de patrones para generar múltiples submuestras o subconjuntos, algunos de ellos afectando al tamaño del CE por disminución en el número de patrones resultantes. Los algoritmos que se engloban dentro de este grupo podemos subdividirlos en dos categorías: los que tienen en cuenta la distribución de los patrones de entrenamiento durante la construcción de las submuestras y los que no la tienen en consideración. Los correspondientes al primer grupo tienen como condición fundamental que las submuestras generadas mantengan un error inferior al 0.5.

4.1.1.1 Algoritmos que no consideran la distribución de los patrones

Uno de los algoritmos más utilizados dentro este grupo es el propuesto por Breiman en 1996, denominado **Bagging** (Bootstrap Aggregating) [Bre., 96]. Con este algoritmo se generan submuestras llamadas *bootstrap* de tamaño *m,* construidas mediante la selección aleatoria con reemplazo de patrones contenidos en el CE original, también de tamaño *m.* La combinación de las decisiones se realiza por votación simple o mayoría. En cada ejecución, el algoritmo de aprendizaje utiliza una submuestra *bootstrap* diferente. Para cada submuestra, cada patrón tiene la probabilidad $1-(1/m)^m$ de ser seleccionado por lo menos una vez de entre las *m* veces que se selecciona un patrón. Para valores grandes de *m*, esto se aproxima a $1- 1/e \approx 63.2\%$, es decir, cada patrón tiene aproximadamente un 63% de probabilidades de aparecer entre los patrones de entrenamiento de la submuestra.

Otro método que no considera distribuciones de probabilidad es el conocido como **Validación cruzada** (en inglés, cross-validated committees) [Par., 96], y consiste en la construcción de submuestras de entrenamiento de menor tamaño mediante la división del CE en subgrupos disjuntos. Cada una de las submuestras se forma con la selección sin reemplazo de patrones contenidos en el CE. El número de submuestras generadas depende directamente del número de clasificadores individuales que se desee construir.

4.1.1.2 Algoritmos que consideran la distribución de los patrones

A diferencia de Bagging que realiza la integración de submuestras de forma independiente, los métodos que utilizan el principio de **Arcing** (Adaptively Resample and Combine) [Bre., 98] construyen submuestras de forma secuencial considerando el error observado en cada una de ellas.

Por su parte, **Boosting** y su variante más usada *AdaBoost* (Adapting Boosting) [Fre., 96], parten de la creación original de una muestra *bootstrap*, en la que a todos los patrones se les asigna inicialmente un mismo peso ($1/m$). Cada vez que se genera un clasificador, se modifican los pesos de los nuevos patrones usados para el siguiente clasificador. La idea es forzar al nuevo clasificador a minimizar el error esperado (Figura 4.1). Para esto, se les asigna más peso a los patrones que fueron mal clasificados, otorgándoles así una mayor probabilidad de ser seleccionados posteriormente con respecto al resto de los patrones.

Entradas: M = CE de *m* patrones etiquetados: $\{x_i \mid i = 1,2,...m\}$.
LEARN = algoritmo de aprendizaje
Inicia
$w_f(i) = 1/m \quad \forall$ i //Inicializa pesos de cada patrón
Para f =0 hasta H //H clasificadores a construir
$p_f(i) = w_f(i)/(\sum_i w_f(i)) \ \forall$ i //Normalización de probabilidades
D_f = LEARN(p_f) //Construcción de D_f considerando p_f
$e_f = \sum_i p_f(i)\ [D_f(x_i) \neq$ etiqueta verdadera $(x_i))]$ //Error del clasificador D_f
Si $e_f > 1/2$ entonces
H = f – 1
fin de algoritmo
Si no
$B_f = e_f/(1-e_f)$
$w_{f+1}(i) = w_f(i)\beta_f^{1-\lfloor D_f(x_i) - etiqueta\,verdadera(x_i)\rfloor}$ $\forall$ i //Nuevos pesos
Fin para
Fin

Figura 4.1. Algoritmo AdaBoost

La decisión final del SMC para un nuevo patrón **y** esta dada por la votación por mayoría ponderada entre los H clasificadores. Esta ponderación es estática, pues el peso se asigna a cada clasificador de acuerdo al desempeño observado durante su integración:

$$D(y) = \arg\max_{t \in c} \sum_{f: D_f(y)=t} \log \frac{1}{\beta_f}$$

Finalmente, Breinman [Bre., 98] propone el algoritmo **Arc-x4** (ver Figura 4.2) como una variante de un algoritmo que realiza *Arcing*. Al igual que AdaBoost,

Arc-x4 contempla la generación secuencial de H submuestras D_1, D_2,…,D_H, pero con dos diferencias importantes: asignación de pesos y esquema de votación. En el primer punto, Arc-x4 realiza el ajuste de los pesos con un esquema mucho más simple que AdaBoost: el peso w_i es proporcional al número de errores que hizo el último clasificador elevado a la cuarta potencia más uno. En lo que respecta al esquema de votación, Arc-x4 combina las decisiones individuales con la votación simple no ponderada.

Entradas: M = Un conjunto de *m* patrones etiquetados: $\{x_i \mid i = 1,2,...m\}$.
LEARN (algoritmo de aprendizaje)
MalClasif = Acumulador de los errores cometidos por cada clasificador
e = Total de errores cometidos por un clasificador
E = Error calculado para el clasificador D_f

Inicia
$w_f(i) = 1/m \quad \forall$ i //Inicializa los pesos para cada patrón
$MalClasif_f(i) = 0 \quad \forall$ i // Inicializa acumulador de mal clasificados
Para f = 1,.., *H*
D_f = LEARN(w_f) // Construcción de D_f considerando w_f
$e_f = \sum_i$ [**1** si $D_f(x_i) \neq$ etiqueta verdadera (x_i), sino **0**] // Determina aciertos
$E_f = \sum_i w_f(i) * e_f(i)$ //Cálculo del error del clasificador D_f
Si $E_f > 0.5$ **entonces**
H = f - 1
terminar // Finaliza la construcción de clasificadores
Sino
$MalClasif_f(i) = MalClasif_{f-1}(i) + e_f(i) \quad \forall$ i //Actualiza mal clasificados
$w_{f+1}(i) = 1 + MalClasif_f(i)^4 \quad \forall$ i // Actualización de pesos
fin Si
$w_{f+1}(i) = w_{f+1}(i) / \sum_i w_{f+1}(i) \quad \forall$ i //Normalización de pesos
fin para
Fin

Figura 4.2. Algoritmo Arc-x4

4.1.2 MANIPULACIÓN DE LOS ATRIBUTOS

A diferencia de los métodos anteriormente mencionados, los métodos contenidos en este grupo manipulan los atributos de los patrones, afectando directamente a la dimensión del CE por reducir la cantidad de atributos manejados en cada una de las submuestras [Kun., 01d].

En su funcionamiento, se realiza la combinación en paralelo de clasificadores utilizando distintos subconjuntos de atributos. La selección de los atributos que se incluirán en cada una de las submuestras puede hacerse de múltiples formas: selección aleatoria con y sin reemplazo, utilización de un algoritmo genético para

determinar los atributos que formarán parte de cada uno de los subconjuntos y adaptación de los algoritmos AdaBoost para estos fines.

En relación a los atributos, la utilización de un sólo clasificador tiene diversas dificultades que pueden ser superadas al utilizar un SMC [Ho., 92]:

a) Medidas de los atributos en escalas diferentes: en los conjuntos de datos se pueden encontrar atributos con valores nominales, ordinales o de escala radial. Estos datos pueden no ser fácilmente normalizados en una sóla escala, lo que dificulta la utilización de una sóla métrica de distancia significativa para todos los datos. Mediante la utilización de un SMC que divide el conjunto de datos en varios subgrupos, las medidas de los atributos con escalas diferentes pueden ser igualadas de forma independiente en sus escalas correspondientes.
b) Diferentes medidas de similaridad: en algunos casos, resulta apropiado utilizar diferentes enfoques de reconocimiento y puede no ser significativa la función de distancia definida en el cálculo de los valores de los atributos para todos los enfoques, ya que cada método puede aportar información no proporcionada por otro método y, de esta forma, tomar decisiones consistentes y más precisas. Al utilizar diferentes medidas de similaridad y distintos procedimientos de clasificación, se pueden definir varios subgrupos de atributos semejantes, de tal forma que la información contenida en cada subgrupo puede ser procesada más eficazmente.
c) Selección dinámica: la competencia de atributos para diferentes patrones de entrada puede ser separada y seleccionada de forma dinámica cuando se dispone de conocimiento sobre las condiciones del patrón de entrada. La selección de clasificación también es posible si una medida de confianza puede ser asociada con las decisiones de la clasificación. La selección puede ser basada en la detección de atributos o en una posible asociación de la ejecución de los clasificadores con distintas características de la misma imagen. Otra alternativa es la construcción de un clasificador por cada una de las clases existentes en el conjunto de datos [Ho., 92].
d) Diferentes enfoques de reconocimiento: para un mismo problema de reconocimiento, pueden tenerse diferentes enfoques en la obtención de la solución, pudiendo no ser muy significativa la función de distancia definida en el cálculo de los valores de los atributos para cada uno de los clasificadores. Cada clasificador puede contribuir con información no proporcionada por ningún otro. Por ejemplo, un clasificador puede reconocer el objeto como un entero, mientas que otro puede reconocer los componentes del objeto y entonces, al conjugar las decisiones individuales, obtener decisiones más consistentes.

Estos métodos son de utilidad cuando se tienen CE formados con muchos atributos, algunos de ellos redundantes, y los patrones de cada grupo describen

distintas áreas (por ejemplo, sonido y visión), o cuando su procesamiento requiere de diferentes tipos de análisis, como la representación de momentos y frecuencias. Generalmente, este método es utilizado con algoritmos genéticos y árboles de decisión.

4.1.3 MANIPULACIÓN DE LAS ETIQUETAS DE CLASE

Estos métodos tienen su mayor utilidad cuando se tienen CE que contienen un gran número de clases. Como ejemplo de ellos, está el método *error-correcting output coding* que consiste en la generación de diferentes clasificadores de forma aleatoria que se entrenan como un problema de dos clases [Die., 95], [Kun., 05]. Para ello, las clases representadas en el CE se re-etiquetan temporalmente como pertenecientes a una de esas dos clases. Al formar los grupos A y B, el CE se descompone en partes iguales (considerando las clases por grupo) y el criterio de selección de las clases que cada grupo contendrá se realiza de forma aleatoria. El proceso se repite tantas veces como clasificadores se desee construir.

El mecanismo de clasificación es el siguiente: disponemos de los dos grupos, cuyas etiquetas para el grupo A es 0 y para el grupo B es 1. Al momento de llegar un nuevo patrón, si el clasificador individual i-ésimo determina que al patrón le corresponde la etiqueta 0, entonces cada una de las clases contenidas en el grupo A recibe un voto, de lo contrario lo reciben las contenidas en el grupo B. Una vez obtenidos los votos de todos los clasificadores, la etiqueta que se le asigna al patrón es la perteneciente a la clase con el mayor número de votos obtenidos.

4.1.4 DIFERENTES CLASIFICADORES CON UN MISMO CONJUNTO DE ENTRENAMIENTO

En este tipo de combinación se tiene, para el entrenamiento, un CE con la que todos y cada uno de los clasificadores realizan su aprendizaje. La naturaleza de los clasificadores deberá ser variada. Por ejemplo, se puede realizar la combinación de las decisiones arrojadas por una red neuronal artificial, un clasificador de regla del vecino más cercano, un árbol de decisión y un algoritmo de regresión lineal. Bahler et al. [Bah., 00] utilizan un conjunto de tres clasificadores heterogéneos: un árbol de decisión, una red bayesiana y una red neuronal backpropagation. Las diferentes opiniones de los clasificadores son combinadas utilizando cinco métodos de fusión, de los cuales algunos realizan la ponderación del clasificador en función de su desempeño. Una última modalidad en este tipo de enfoque es el utilizado por SMC formados por redes neuronales con diferentes arquitecturas o diferentes tipos de redes sobre un mismo conjunto de datos [Gia., 01b].

4.1.5 INYECTANDO ALEATORIEDAD

Estos métodos para generar la combinación de clasificadores se basan en la inyección de aleatoriedad en el algoritmo de aprendizaje. Algunos ejemplos de la utilización de estos métodos son los siguientes:

a) Asignación de los pesos iniciales en la combinación de redes neuronales. Si una red neuronal es utilizada varias veces con un mismo conjunto de datos, pero con diferentes pesos iniciales, los resultados de salida pueden variar de un caso a otro [Kol., 91], [Gia., 01b].

b) Inyección de ruido aleatorio en los valores de las variables. Al utilizar un SMC con la combinación de varias submuestras bootstrap, se realizan mutaciones aleatorias sobre los valores de los atributos de cada patrón de entrenamiento antes de proporcionarlos al algoritmo de aprendizaje [Rav., 96].

c) Selección aleatoria de los patrones y/o atributos que contendrá cada submuestra en un SMC que utilice el submuestreo. Al integrar varias submuestra a partir de una sóla y puesto que la selección de los patrones se realiza de forma aleatoria en todos los casos, difícilmente dos subconjuntos contendrán los mismos patrones [Bar., 03a].

d) Selección aleatoria de la mejor partición variable – valor en combinación de árboles de decisión. Al momento que se desea particionar un nodo padre, se considera un determinado número de opciones equiprobables, de las cuales se selecciona al azar la que se utilizará para generar los siguientes nodos hijos [Ali., 96].

4.2 TOMA DE DECISIONES

En la literatura, se proponen dos estrategias para realizar la combinación de las decisiones individuales de los clasificadores: la *fusión* y la *selección* (también conocida como combinación).

4.2.1 FUSION DE CLASIFICADORES

La fusión de clasificadores asume que todos lo clasificadores son competitivos y complementarios (igualmente *expertos*). Por este motivo, cada uno de ellos emite una decisión respecto a cada patrón de prueba que se presenta. La salida de los clasificadores es un vector H-dimensional que contiene las decisiones de cada uno de los H clasificadores:

$$[D_{i,1}(\mathbf{y}),..., D_{i,H}(\mathbf{y})]^T$$

Este vector puede contener en su interior alguna de las siguientes opciones [Kun., 03]:

a) Un vector c-dimencional $[D_{i,t}(\mathbf{y}),\ldots,D_{i,c}(\mathbf{y})]^T$ con la estimación a posteriori de la distribución de las probabilidades sobre el conjunto de clases $P(p_t|\mathbf{y})$, donde $t = 1,\ldots,c$.

b) Un vector H-dimensional $[D_{i,f}(\mathbf{y}),\ldots,D_{i,H}(\mathbf{y})]^T$ que contiene la etiqueta de clase proporcionada a un determinado patrón de prueba por cada uno de los clasificadores $D_f(\mathbf{y})$, donde $f = 1,\ldots,H$.

c) Decisión correcta o incorrecta. La salida del clasificador $D_f(\mathbf{y})$ es 1 si x_i fue clasificado correctamente por D_f, y 0 en caso contrario. Este tipo de vector es también conocido en la literatura como "*oráculo*" por asumir que se conoce previamente la etiqueta correcta de **y**.

Del conjunto de salidas, se debe tomar una decisión final mediante la aplicación y/o utilización de alguno de los siguientes principios: votación por mayoría simple y votación por mayoría ponderada.

4.2.1.1 Votación por mayoría simple

En este tipo de votación, cada uno de los componentes individuales proporciona un voto con valor de 1. La etiqueta de clase que se asigna a **y** será la que haya obtenido el mayor número de votos [Kun., 02a]. Cuando se trabaja con conjuntos de datos con más de dos clases, usualmente ocurren empates entre algunas clases. Para resolver este problema, se han considerado varios criterios: de entre las clases ganadoras seleccionar de forma aleatoria la ganadora, o con la implementación de un clasificador adicional cuya función es la de inclinar la decisión hacia una determinada clase [Kub., 00]. En caso de que este último clasificador se decidiera por una tercera clase, la posible solución sería determinar la clase ganadora de forma aleatoria o considerando las distancias obtenidas por los clasificadores del empate, es decir, asignar la etiqueta proporcionada por el clasificador que haya mostrado la menor distancia al nuevo patrón.

4.2.1.2 Votación por mayoría ponderada

Este tipo de votación se realiza de la misma forma que la votación simple, con la variante de que cada clasificador cuenta con un peso diferente, por lo general, asignado de acuerdo a la estimación de probabilidad de error [Woo., 97]. La decisión final puede realizarse por mayoría, promedio [Kun., 01c], minoría, mediana [Che., 01], producto de votos, o utilizando algún otro método más complejo: *naive Bayes, Behavior – Knowledge space, fuzzy integral, Dempster –*

Shafer, combinación de *Dempster - Shafer* [Kun., 01a], *decision forest* [Ho., 00], la cuenta de Borda y regresión logística [Ho., 92], entre otras.

4.2.2 SELECCIÓN DE CLASIFICADORES

Cuando se utiliza un SMC, uno de los más populares métodos para realizar la fusión es la votación simple. Sin embargo, cuando el desempeño de los miembros del SMC no es uniforme, la eficiencia de este tipo de votación es afectada negativamente. Hansen y Salomon [Han., 90] demostraron que, si cada uno de los clasificadores es combinado con un error individual inferior al 50%, puede esperarse que la precisión del SMC aumente cuando se agreguen más componentes a la combinación. Sin embargo, esta presunción no siempre es cierta; Matan [Mat., 96] afirma que en algunos casos la votación simple puede tener peor desempeño que cada uno de los miembros individuales del SMC.

En los últimos años, se ha propuesto el desarrollo de SMC basados en el concepto de selección de clasificadores, en el cual para cada patrón de entrada únicamente un clasificador es seleccionado para asignar la etiqueta de clase [Sri., 94]. En la selección de clasificadores, el conjunto de datos es dividido en r regiones de competencia ($r > 1$). Cada región es denotada por $R_1,...R_r$, la cantidad de clasificadores a utilizar no necesariamente es igual al número de regiones y, además, cada clasificador puede ser experto en una o varias regiones del espacio de atributos. Un ejemplo de la partición del conjunto de datos en regiones para un caso de dos clases se muestra en la Figura 4.3 [Kun., 00]. Las dos regiones de clasificación (cuadros y estrellas) son subdivididas utilizando el diagrama de Voronoi[1]. Cuando se presenta un patrón para su clasificación, el clasificador responsable de la región más cercana a dicho patrón es el que toma la decisión final, por lo que para cada patrón de entrada se requiere solamente un clasificador de los que integran el SMC para clasificarlo correctamente.

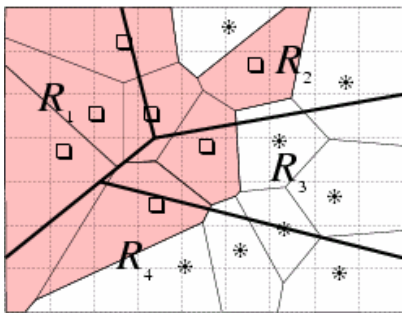


Figura 4.3. Espacio de atributos de un caso de dos clases particionado en cuatro regiones (extraída del artículo original [Kun., 00])

[1] Dado un conjunto de puntos $\{p_1 \ldots p_m\} \in \mathbb{R}^r$, una celda de Voronoi S_i (asociada al punto p_i) se define como: $S_i = \{x \in \mathbb{R}^r : d(x; p_i) < d(x; p_j),\ p = 1...m,\ j \neq i\}$

4.2.2.1 Selección estática

En este tipo de selección, la especificación de las regiones es establecida durante la fase de entrenamiento, previa a la clasificación de patrones. En la fase de operación (clasificación), la región del patrón **y** es primeramente encontrada R_j y, utilizando el clasificador D_j (clasificador responsable de la región j), se asigna la etiqueta que corresponde a **y**.

La asignación *Región – Clasificador* puede hacerse de dos formas:

a) *Primeramente especificar la región y posteriormente asignar un clasificador responsable.* Este método, diseñado por Kuncheva [Kun., 00], [Kun., 02b] y llamado *Clustering and selection*, consta de dos etapas (Figura 4.4): en la *etapa de entrenamiento*, utiliza el algoritmo k-medias para establecer las regiones (grupos) de las cuales se obtienen los centroides que serán utilizados para encontrar el clasificador responsable de clasificar un patrón dado en la etapa de operación o clasificación.

Etapa de entrenamiento

1. Diseñar H clasificadores $D_1,...D_H$, utilizando el conjunto de datos M.
2. Seleccionar el número de regiones *r*.
3. Con el algoritmo k-medias, agrupar M en r grupos, $R_1,...,R_r$. Encontrar el centroide de cada grupo $\mathbf{v_1},...,\mathbf{v_R}$.
4. Para cada grupo R_j, estimar la precisión de clasificación de los H clasificadores, utilizando sólo los patrones de M que están en R_j. Nominar el clasificador con el mayor índice de precisión como $D_i(j)$.
5. Retornar $\mathbf{v_1},...,\mathbf{v_k}$ y $D_{i(1)},...,D_{i(r)}$

Etapa de clasificación

1. Dado un patrón de entrada **y**, encontrar el centroide más cercano de $\mathbf{v_1},...,\mathbf{v_k}$ llamado $\mathbf{v_j}$.
2. Utilizar $D_i(j)$ para etiquetar **y.**

Figura 4.4. Algoritmo de agrupamiento y selección

b) Para cada clasificador, encontrar la región (o grupo de regiones) donde el clasificador tiene mejor desempeño. Hartono et al. [Har., 04] desarrollaron un método aplicado sobre un SMC compuesto por una serie de perceptrones lineales. En su propuesta, la determinación de las regiones se realiza durante el proceso de aprendizaje de la red. Para esto, incorporan a la estructura de la red una neurona supervisora del grado de "confidencia" de las salidas de cada una de las neuronas en la capa de salida. De esta manera, si un miembro del SMC es asignado responsable sólo de un subespacio del

problema general, éste será el miembro del SMC que haya demostrado mayor nivel de confidencia ante un patrón dado. Esquemas similares pueden ser encontrados en los métodos que utilizan RNA con arquitecturas modulares (*mixture system*) [Bau., 04], [Jac., 91], [Har., 04].

4.2.2.2 Selección dinámica

La selección del clasificador que etiquetará a **y** ocurre durante la fase de operación. Esta elección está comúnmente basada en la seguridad de una buena decisión, proporcionándole mayor preferencia al clasificador más seguro. Algunos algoritmos desarrollados para este fin son aquellos que realizan la estimación de precisiones locales. En 1997, Woods et al. [Woo., 97] proponen dos métodos para estimar la precisión local de clasificadores (Figura 4.5). El primero de ellos, llamado *"overall local accuracy"*, es simplemente el porcentaje de patrones contenidos en cada una de las regiones que son correctamente clasificados por cada uno de los H clasificadores. El segundo método, *"local class accuracy"*, considera la clase asignada por un clasificador al patrón de entrada y, entonces, calcula los porcentajes de los patrones de entrenamiento correctamente clasificados hacia la misma clase utilizando los *k* vecinos.

1. Diseñar H clasificadores $D_1,...,D_H$
2. Seleccionar el valor del parámetro *k*.
3. Presentar **y** a cada uno de los clasificadores. Si todos coinciden en la etiqueta de clase, entonces asignar esa etiqueta de clase a **y**.
4. En caso contrario, estimar la precisión local de cada clasificador. Hacer esto considerando la etiqueta de clase proporcionada a **y** por cada clasificador D_f, y encontrar los *k* vecinos más cercanos a **y** para el cual D_f asignó la misma etiqueta. Calcular la proporción de los puntos cuya etiqueta verdadera es **s**. (Es una estimación de la precisión local de D_f con respecto a la clase **s**).
   a. Si hay un único ganador, se asigna esta etiqueta a **y**. En caso contrario verificar si los ganadores empatados dan la misma etiqueta a **y**, de ser así asignar la etiqueta y regresar. Si una clase predomina entre la mayoría de los ganadores, entonces asignar esa etiqueta a **y**.
   b. En caso contrario, si hay una etiqueta de clase entre la mayoría de los clasificadores localmente más competentes, el clasificador con la siguiente competencia más alta asigna la etiqueta de clase. Si todos los clasificadores están empatados y la etiqueta de clase es aún empatada (hay varias etiquetas ganadoras), entonces elegir aleatoriamente una etiqueta de entre las empatadas. Si sólo hay un ganador de la (2°) competencia local y este puede resolver el empate, entonces utilizar la etiqueta ganadora para **y**.

Figura 4.5. Funcionamiento del algoritmo de selección dinámica de clasificadores utilizando precisiones locales

Otros algoritmos basados en este mismo concepto de *precisión local* son el desarrollado por Kuncheva [Kun., 02b], que incluye la selección de forma aleatoria del clasificador ganador cuando se presentan constantes empates, y el propuesto por Giacinto [Gia., 97], que realiza la estimación local utilizando las probabilidades de las clases, a diferencia de Woods que solamente considera la etiqueta de clase asignada al patrón de entrenamiento.

Giacinto [Gia., 99] dispone de H diferentes clasificadores D_f, f = 1,..., H, el conjunto de datos es dividido en *r* regiones R_i, i =1,..., r. Cada región R_i, puede ser subdividida en 2 subregiones $R_i^{l+} = R_i^l \cap R_i^B$ (R_i^B es un calificador Bayesiano) y $R_i^{l-} = R_i^l - R_i^{l+}$, cada clasificador D_f en la región R_i^{l+} corresponde a un óptimo clasificador Bayesiano, mientras que en las regiones R_i^{l-} no se toman decisiones Bayesianas (Figura 4.6).

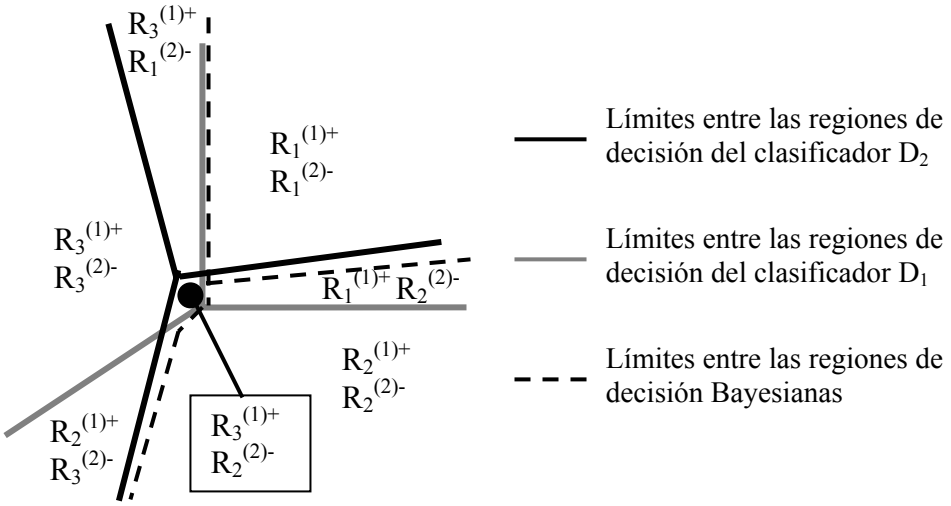


Figura 4.6. División del espacio de representación en regiones con dos clasificadores (extraída del artículo original [Gia., 99])

Para la selección dinámica propone dos métodos partiendo de la estimación de precisiones locales en las que hace uso del algoritmo *k*–NN; *método de selección a priori,* donde la selección se realiza sin conocimiento de la clase asignada por el clasificador D_f al nuevo patrón y, *método de selección a posteriori*, en el que la clase asignada por el clasificador D_f al nuevo patrón es conocida (Figura 4.7).

a) En el algoritmo, el cálculo de $p(correcto_j)$ corresponde a cualquiera de los dos métodos utilizados.

b) En los pasos 1 y 2, se seleccionan k' clasificadores (k'<= H) para desechar los clasificadores que tienen probabilidades menores a 0.5 en la precisión al clasificar el patrón **y**.

Parámetros de entrada: valor del parámetro *k*
Inicio
Para cada patrón de entrada **y hacer**
Si *"selección a posteriori"* **entonces**
$\forall$ D_f (f = 1,...,H): calcular D_j(**y**)
Si todos los clasificadores asignan a **y** la misma etiqueta de clase t ϵ c **entonces**
Clasificar **y**
Devolver t
Sino
Hacer $\forall$ D_f (f = 1,...,H)
Paso 1: calcular $p(correcto_j)$ // Probabilidad de clasificar correctamente a **y**
Paso 2: **si** $p(correcto_j) < 0.5$ **entonces** rechazar D_f
Fin Hacer
Paso 3: identificar el clasificador que maximice $p(correcto_j)$, j = 1,...,*k*'
k' <= H, estableciéndolo como D_s.
Paso 4: $\forall$ D_f f = 1,..., *k*', calcular: $dif_f=[p(correcto_s)-p(correcto_l)]$
Paso 5: **si** $\forall$ f = 1,...,*k*', f $\neq$ s, $dif_f >$ umbral **entonces** seleccionar a D_s
De lo contrario seleccionar el clasificador con $dif_f <$ humbral.
Fin Sino
Fin

Figura 4.7. Selección dinámica propuesta por Giacinto y Roli

c) Las diferencias calculadas en el paso 4 son utilizadas para calcular una clase de *"confidencia"* para la selección. Si se elige la selección *a posteriori*, entonces la diferencia dif_j es calculada sólo para los clasificadores que toman una decisión diferente de la tomada por el clasificador seleccionado D_s. Si todas la diferencias son superiores que un umbral fijo preestablecido (0.1), entonces es razonable designar a D_s como el clasificador más apropiado para clasificar el nuevo patrón.

No es razonable seleccionar el clasificador D_s si otros clasificadores proporcionan resultados similares en el paso 5. Para esta situación, se sugiere realizar la elección del clasificador de forma aleatoria.

4.3 DIVERSIDAD DE LAS DECISIONES

En múltiples investigaciones [Kun., 02c], [Die., 97], [Han., 90], [Gia., 01], [Kun., 01b], [Ban., 03], [Nar., 05], se ha establecido como requisito indispensable y condicional del desempeñó del SMC, el grado de diversidad de los clasificadores. Dos clasificadores son diversos si arrojan decisiones diferentes al clasificar un mismo patrón de entrada. Para ejemplificar esto, consideremos un conjunto de tres clasificadores D_1, D_2 y D_3 y el patrón **y** a clasificar. Si los tres clasificadores en sus decisiones individuales son idénticos, entonces cuando la decisión de D_1 está

equivocada, las decisiones de D_2 y D_3 también serán erróneas. Caso contrario sucede si las decisiones individuales de los clasificadores difieren entre sí, entonces cuando la decisión de D_1 sea errónea, las decisiones individuales de D_2 y D_3 pueden ser correctas; de esta forma, si se toma la decisión final por mayoría, el patrón **y** podrá clasificarse correctamente.

Para obtener esta diversidad se han propuesto varios métodos [Die., 97], siendo más utilizados los que incluyen la manipulación de los patrones de entrenamiento (submuestreo). Estos métodos incluyen Bagging [Bre., 96], Boosting y su variante más usada AdaBoost [Fre., 96], Arcing [Bre., 98] y secuencial [Bar., 03a]. Sin embargo, no es suficiente la integración de las submuestras con los diferentes métodos descritos en la Sección 4.1, es necesario utilizar alguna medida que nos indique cuán diverso es el comportamiento de los componentes individuales, es decir, cuán diversas son las decisiones individuales.

En los últimos años se han desarrollado y adaptado medidas estadísticas útiles que ayudan al análisis de la diversidad, dependiendo de la cantidad de clasificadores evaluados. Estas medidas pueden catalogarse en dos grupos [Kun., 03]: pairwise (Q-estadístico, medida de desacuerdo, medida de doble fallo y coeficiente de correlación) y medidas non-pairwise (varianza de Kohavi-Wolpert, medida de entropía, medida de dificultad, interrater agreement, diversidad generalizada y diversidad de coincidencia de fracaso). Para las medidas pairwise, la medida de diversidad viene dada por el promedio de las medidas sobre todos los pares de clasificadores. La elección de la medida a utilizar va a depender directamente de la cantidad de clasificadores a utilizar. La Tabla 4.1 resume las medidas de diversidad más ampliamente usadas. La columna (↑/↓) indica cuándo la diversidad resulta mayor, si la medida es menor (↓) o mayor (↑).

Tabla 4.1. Medidas de diversidad

Medida	Nomenclatura	↑/↓
Q-estadístico	Q	↓
Coeficiente de correlación	ρ	↓
Medida de desacuerdo	Des	↑
Medida de doble fallo	DF	↓
Varianza de Kohavi-Wolpert	*kw*	↑
Interrater agreement	Ac	↓
Medida de entropía	*Ent*	↑
Medida de dificultad	Dif	↓
Diversidad generalizada	GD	↑
Diversidad de coincidencia de fracaso	CFD	↑

Aún cuando el objetivo primordial de estas medidas es indicar la diversidad existente en un SMC, algunos autores afirman que estas medidas pueden indicarnos también el posible grado de precisión que tendrá el SMC. Por ejemplo, Banfield et al., [Ban., 03] utilizan "Porcentage Correct Diversity Measure (PCDM)", Q-

estadístico y Kappa estadístico para determinar la diversidad en árboles de decisión, y encuentran que la precisión en la clasificación incrementa conforme incrementa el grado de diversidad. Por otro lado, Shipp et al. [Shi., 02] utilizan dos SMC, uno con tres clasificadores lineales y otro con tres clasificadores cuadráticos, y determinan la diversidad utilizando 10 medidas diferentes: Q estadístico, coeficiente de correlación, medida de interrater agreement, diversidad generalizada, varianza de Kohavi-Wolper, diversidad de coincidencia de fracaso, entropía, medida de desacuerdo, medida de doble fallo y medida de dificultad. Experimentalmente concluyeron diciendo "…tenemos muy poca evidencia de cualquier relación existente entre la precisión de los métodos de combinación (mayoría, máximo, mínimo, promedio, producto, naive Bayes, behavior-knowledge space, método Wernecke's y plantillas de decisión) y los valores de las medidas de diversidad…", "Estos resultados son desalentadores porque se supone que las medidas de diversidad indican el desempeño de la combinación de clasificadores, y nosotros encontramos muy poca evidencia de cualquier relación…".

Algunas de las propiedades que deben cumplir las medidas de diversidad son las siguientes [Nar., 05]:

a) Propiedad 1. **La medida de diversidad tiene un valor finito para el SMC.**
b) Propiedad 2. **La medida de diversidad está acotada.** Las medidas de diversidad cuentan con un valor mínimo y un valor máximo para su adecuada interpretación.
c) Propiedad 3. **El resultado de la medida de diversidad puede ser representada en forma vectorial.**
d) Propiedad 4. **La medida de diversidad es simétrica.** Una medida de diversidad puede ser simétrica o no-simétrica con respecto a la correcta o incorrecta clasificación (0 ó 1). El cumplimiento de esta propiedad no se considera ventajoso o perjudicial, simplemente es una característica propia de la medida de diversidad.

Para determinar la diversidad, en el presente trabajo se utilizan las tres primeras medidas presentadas en la Tabla 4.1: Q-estadísitico, coeficiente de correlación y la media de desacuerdo. Para la ejecución de estas medidas, se hace uso del "*oráculo*" [Kun., 03], en el cual las decisiones individuales de los H clasificadores se guardan en una matriz binaria, donde 1 significa una clasificación correcta y 0 la clasificación incorrecta.

Para simplificar la manipulación de las decisiones de los clasificadores individuales, sus decisiones son resumidas en una tabla como la que se muestra en la Tabla 4.2. Esta tabla ejemplifica los resultados de los clasificadores D_i y D_j al clasificar un patrón de prueba **y**. Los valores de cada una de las celdas corresponden a lo siguiente:

a = D_j y D_i clasificaron correctamente a **y**.

b = D_j clasificó erróneamente a **y** y D_i correctamente.

c = D_j clasificó correctamente a **y** y D_i erróneamente.

d = D_j y D_i clasificaron erróneamente a **y**.

Tabla 4.2. Relaciones entre las decisiones individuales de dos clasificadores

	D_j Correcto (1)	D_j Erróneo (0)
D_i Correcto (1)	**a**	**b**
D_i Erróneo (0)	**c**	**d**

4.3.1 Q-ESTADÍSTICO (↓)

Utilizando las relaciones de la Tabla 4.2, el Q-estadístico para dos clasificadores D_i y D_j viene dado por:

$$Q_{i,j} = \frac{ad - bc}{ad + bc}$$

Para cuando se tiene más de dos clasificadores, el Q-estadístico se calcula con la siguiente fórmula [Kun., 03]:

$$Q_{av} = \frac{2}{H(H-1)} \sum_{i=1}^{H-1} \sum_{j=i+1}^{H} Q_{i,j}$$

Para clasificadores independientes (y $m \to \infty$), $Q_{i,j} = 0$. Los valores de Q varían entre -1 y 1 [Kun., 02a], [Kun., 02c].

4.3.2 COEFICIENTE DE CORRELACIÓN (↓)

Para expresar cuantitativamente el grado en que las decisiones individuales de un SMC están relacionadas, es necesario calcular un coeficiente de correlación dado por la siguiente fórmula [Shi., 02], [Kun., 03]:

$$\rho_{i,j} = \frac{ad - bc}{\sqrt{(a+b)(c+d)(a+c)(b+d)}}$$

El coeficiente de correlación cuenta con las siguientes características:

a) El valor del coeficiente de correlación varía entre -1.00 y +1.00. Ambos extremos representan relaciones perfectas entre las decisiones y 0.00 representa la ausencia de asociación.
b) Cuanto más cercano sea a cero el coeficiente de correlación, más débil será la asociación entre los clasificadores.
c) Una relación positiva significa que los clasificadores que obtienen calificaciones altas en una variable tienden a obtener calificaciones altas en la otra. De igual forma, una relación negativa se presenta cuando los clasificadores que obtienen calificación baja en una variable tienden a obtener calificación baja en la otra.

Con la intención de hacer una adecuada interpretación de los índices de correlación, en el presente trabajo se hará uso de la clasificación que aparece en la Figura 4.8:

±0.00 a 0.25 baja o ninguna correlación
±0.26 a 0.50 correlación moderada baja
±0.51 a 0.75 correlación moderada alta
±0.76 a 1.00 alta a perfecta correlación

Figura 4.8. Categorías del coeficiente de correlación

Es importante recalcar que el hecho de encontrar una fuerte asociación entre las decisiones no implica que necesariamente esta correlación sea de carácter causal. Lo único que permite identificar son *co-movimientos* significativos de las decisiones, es decir, el grado de asociación entre el acierto o el error igualmente observado por los clasificadores al clasificar un nuevo patrón.

4.3.3 MEDIDA DE DESACUERDO (↑)

Para encontrar la diversidad en un SMC, esta medida obtiene el porcentaje de las ocasiones en que los clasificadores no coinciden en las decisiones hacia un mismo patrón de entrada (D_i clasifica a **y** de forma correcta y D_j no; celdas a y b de la Tabla 4.2).

La medida viene dada por la siguiente formula [Shi., 02], [Kun., 03]:

$$D_{i,j} = \frac{b+c}{a+b+c+d}$$

Cuanto más grande sea el valor de $D_{i,j}$, mayor será la diversidad.

4.3.4 MEDIDA DE VARIABILIDAD (↑)

Una última forma de determinar la diversidad en las decisiones individuales es mediante la medida de la variabilidad. Esta medida, a diferencia de las medidas anteriores, utiliza una matriz de decisiones que guarda la etiqueta de clase que los clasificadores asignan a cada patrón de prueba.

Para determinar la variabilidad, se hace uso de la siguiente fórmula [Val., 02a]:

$$v = \frac{\sum_{y=1}^{p} a}{p}$$

donde:

$$a = \begin{cases} 0 \quad si \quad D_1(y) = D_2(y) = ,..., \; D_H(y) \\ 1 \quad en \quad caso \quad contrario \end{cases}$$

p = total de patrones evaluados

Esta medida tiene como objetivo determinar el grado de variabilidad en las decisiones individuales, mediante la recopilación de todos aquellos casos donde al menos un clasificador no coincidió con el resto de clasificadores en la etiqueta de clase asignada a **y**. Al igual que la medida de desacuerdo, valores altos indican mayor diversidad.

PARTE II

METODOLOGÍA Y RESULTADOS

Capítulo 5

Estrategias de solución

Para llevar a cabo este trabajo, se realizo una investigación sobre diferentes metodologías encaminadas a extender el uso de SMC para afrontar algunos de los problemas que deterioran la clasificación: manipulación eficiente de muestras de entrenamiento no balanceadas y escalabilidad de algoritmos para poder ejecutarlos incluso cuando los conjuntos de datos tienen un tamaño tan grande que imposibilita su manejo en computadoras estándar. En el primero de estos problemas, se pretende incrementar el porcentaje de clasificación correcta en las clases minoritarias sin afectar significativamente el de las clases mayoritarias (es decir, incrementar los valores de la media geométrica de las precisiones por clase). En el segundo problema, se tratará de encontrar maneras eficientes de aplicar esos algoritmos sin afectar la precisión del método de clasificación. En ambos casos, se trata de aplicaciones novedosas de los SMC.

Las estrategias de solución aquí propuestas se basan en la manipulación de los patrones de un CE (ver Sección 4.1.1), buscando diversidad entre los componentes individuales del SMC y la toma de decisiones individuales en su modalidad de fusión de clasificadores.

5.1 ALGORITMOS DE CLASIFICACIÓN

En el presente trabajo, se utilizaron dos reglas de aprendizaje: la regla 1-NN (Nearest Neighbor) [Das., 91] y redes neuronales de dos tipos múlticapa con aprendizaje backpropagation [Rum., 86] y red con estructura modular.

5.1.1 REGLA 1-NN

La regla 1-NN es un clasificador supervisado no-paramétrico ampliamente conocido que combina una simplicidad conceptual con un ratio de error convenientemente limitado en términos del error óptimo de Bayes [Cov., 67]. En su clásica manifestación, dado un grupo de *m* patrones previamente etiquetados (o conjunto de entrenamiento, CE), el clasificador asigna a algún patrón de prueba la clase indicada por la etiqueta del vecino más cercano en el CE. Por otro lado, debido al bajo coste computacional, la sencillez de su implementación y el tipo de datos utilizado, para el cálculo de la similitud entre patrones se hace uso de la distancia Euclídea.

5.1.2 REDES NEURONALES

Una segunda regla de decisión utilizada para clasificar patrones es la que involucran las redes neuronales artificiales. De manera concreta, se aplican dos redes feed-forward: una con estructura múlticapa y aprendizaje tipo *backpropagation* (retropropagación), y otra con estructura modular y aprendizaje de gradiente estocástico.

Varios aspectos justifican el uso de estas redes en la presente tesis. Como primer punto cabe mencionar que con la arquitectura múlticapa es posible resolver el problema del XOR[1] [Rum., 86]. Además, debido a que las subredes que forman la red múlticapa han sido inicializadas de forma independiente, es posible reconocer adecuadamente los patrones aún cuando alguna de las subredes encuentre ambigüedad o confusión en la clasificación, ya que el resto de las subredes pueden contrarrestar su efecto y favorecer así su clasificación.

Por otro lado, el método conocido como *backpropagation error* (propagación del error hacia atrás) o método de gradiente decreciente, es un método basado en la generalización de la regla delta [Wid., 60] que, a pesar de sus propias limitaciones[2] [Nat., 97], es en la actualidad el método de aprendizaje más ampliamente estudiado en RNA, entre otras razones, debido a la gran capacidad que tiene para organizar la

[1] Función que el Perceptrón unicapa es incapaz de aprender.
[2] Principalmente, coste computacional, ambigüedad en determinar la topología más adecuada a un determinado problema, y el desconocimiento del proceso que ocurre entre las capas ocultas y la de salida durante el entrenamiento de la red.

representación del conocimiento en las capas ocultas y a su elevado poder de generalización.

Algunas consideraciones realizadas al implementar el perceptrón múlticapa son las siguientes:

a) Topología. Para los experimentos aquí realizados, se utilizó el entrenamiento de un perceptrón múlticapa con una capa oculta. Con la intención de disminuir la carga computacional asociada al entrenamiento, el total de unidades en esta capa está relacionado con el número de atributos más la unidad.

b) Pesos de las conexiones. Los pesos de las conexiones son inicializados de forma aleatoria en un rango de valores entre -0.5 y 0.5.

c) Razón de aprendizaje (η) y factor momento (α). Existen múltiples formas para asignar el valor a η y α, sin embargo, basándonos en evidencia empírica que demuestra una adecuada convergencia de la red [Gas., 01], [Pao., 89], [McC., 88], [Dem., 93] al utilizar $\eta \in [0.8, 0.9]$ y $\alpha \in [0.6, 0.7]$, en los experimentos aquí realizados se considera $\eta = 0.9$ y $\alpha = 0.7$.

d) Función de activación de las neuronas ocultas y de salida. Debido a que el algoritmo *backpropagation* exige que la función de activación sea continua y, por tanto, derivable para poder obtener el error o valor delta de las neuronas ocultas y de salida, la función utilizada es la función sigmoidal.

El segundo enfoque de RNA utilizado, es el llamado *mixture of experts* o r*edes modulares* (ver Sección 3.4.2.1). Se trata de métodos ampliamente recomendados para dar solución a problemas de reconocimiento de patrones de forma eficiente, mediante el entrenamiento de expertos locales [Jac., 91].

Existen múltiples variantes en su implementación, las cuales, en su mayoría difieren entre sí en la naturaleza del módulo regulador o integrador. En algunos casos este módulo no es más que una neurona que evalúa el desempeño de cada uno de los módulos expertos [Har., 04], o es una red neuronal entrenada con un conjunto diferente al utilizado para entrenar a los expertos [Bau., 04], [Ste., 91], o el entrenamiento de todos los módulos, incluido el módulo integrador, son entrenados con el completo CE [Jac., 91], [Zam., 99] (Figura 5.1). En el presente estudio, se implementó un SMC, en el cual cada uno de sus elementos corresponde a una red modular (Figura 5.1), en el que todos y cada uno de sus módulos son entrenados con el mismo CE.

Algunas consideraciones realizadas al implementar las redes modulares son las siguientes:

a) Topología. La estructura de cada elemento de la red modular corresponde a un perceptrón lineal, en el cual el número de nodos en la capa de entrada

corresponde al número de atributos del patrón de entrada. Para las redes expertas, el número de neuronas en la capa de salida es igual al número de categorías del problema, en tanto que para la red integradora es igual al número de expertos utilizados.

b) Pesos de las conexiones. Al igual que con el perceptrón múltiplicapa, los pesos de las conexiones son inicializados de forma aleatoria en un rango de valores entre -0.5 y 0.5.

c) Decisión final. La combinación de las decisiones se realiza por mayoría simple.

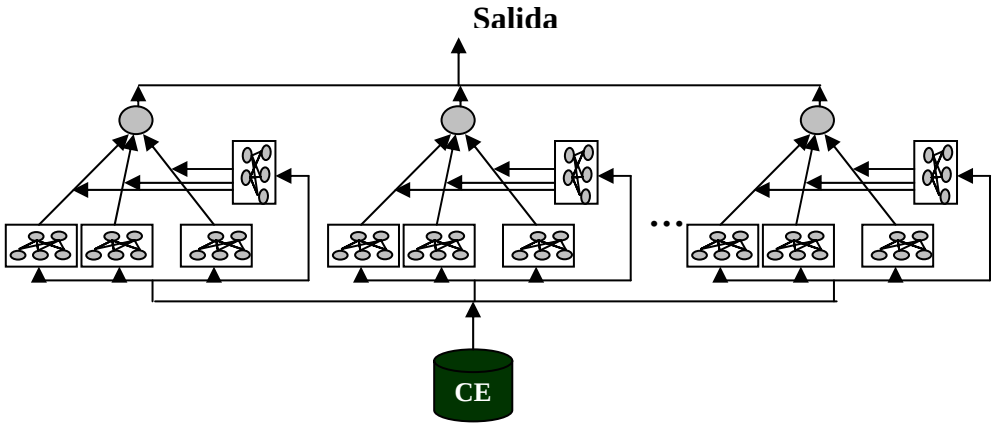


Figura 5.1. SMC con redes modulares

En el Apéndice I, se describen más ampliamente el funcionamiento de estas dos redes.

5.2 CONJUNTOS DE DATOS UTILIZADOS

Con la finalidad de validar las nuevas propuestas presentadas en este trabajo con otras investigaciones relacionadas, los resultados aquí reportados corresponden a experimentos realizados sobre 17 conjuntos de datos que se emplean con gran frecuencia en publicaciones de diversas áreas del reconocimiento de patrones. Algunas de las principales características de estas bases de datos están resumidas en la Tabla 5.1.

Para una mayor descripción de cada uno de los CE, pueden consultarse las siguientes cinco fuentes de donde fueron obtenidas: banco de datos Elena[3], banco de datos UCI[4], Feltwell[5], Ism[5] y de [Bar., 95][6].

[3] http://www.dice.ucl.ac.be/neural-nets/Research/Projects/ELENA/databases/REAL/
[4] Merz, C. J., Murphy, P. M.: UCI Repository of Machina Learning Databases. Univ. Of California, Irving (1988).
[5] http://www.aris.sai.jrc.it/dfc/imágenes.html

Tabla 5.1. Bases de datos utilizadas en la experimentación

	No. clases	No. atributos	Tamaño CE	Tamaño TS
Cayo[4]	11	4	3086	2933
Cáncer[2]	2	9	546	137
Feltwell[3]	5	15	1418	5820
Glass[2]	6	9	174	40
Heart[2]	2	13	216	54
Iris[1]	3	4	120	30
Ism[5]	2	9	10065	1118
Liver[2]	2	6	276	69
Pima[2]	2	8	615	153
Phoneme[1]	2	5	4322	1082
SatImage[1]	6	36	5147	1288
Sonar[2]	2	60	167	41
Vehicle[2]	4	18	678	168
German[2]	2	24	800	200
Waveform[2]	3	21	4000	1000
Segment[2]	7	19	1848	462
Wine[2]	3	13	144	34

A excepción de las bases de datos Ism, Cayo y Feltwell, de cada conjunto de datos se obtuvieron 5 conjuntos mediante validación cruzada (ver Sección 2.3.4.1), en las que se consideró el 80% de los patrones para entrenamiento y el 20% restante para evaluar el sistema (conjunto de test, CT). Del conjunto de datos Ism se extrajeron, mediante validación cruzada, 10 particiones, en las cuales el 90% de los datos originales se utilizan para entrenamiento y el 10% restante para prueba. Del conjunto de datos Cayo se extrajeron, mediante validación cruzada, únicamente tres CE con su respectiva CT. Finalmente, Feltwell constituyó un sólo CE con dos CT obtenidos por validación cruzada.

Resulta difícil estandarizar algún método para tratar de la misma forma todas las bases de datos, pues cada una de ellas cuenta con características muy particulares: tamaño, número de clases, distribución de patrones por clase, dimensionalidad, etc. Por ello, estos conjuntos son separadas en dos grupos que, considerando el número de clases y el balance existente entre ellas son:

a) Bases de datos desbalanceadas de dos clases. Los conjuntos de datos de esta naturaleza son los originales de Ism y Phoneme, así como los conjuntos de

[6] R. Barandela, "Una metodología para el reconocimiento de patrones en la solución de tareas geólogo-geofísicas". Geofísica Internacional, Vol. 34, no.4, Pág. 399 – 405, 1995.
[5] K. Woods, C. Doss, K, Bowyer, J. Solka, C. Priebe & W. Kegelmeyer. *Comparative evaluation of Pattern Recognition techniques for detection of microcalcifications in Mammography.* International Journal of Pattern Recognition and Artificial Intelligence, Vol. 7(6), pp. 1417 – 1436, 1993.

datos Glass, Satimage y Vehicle con algunas adaptaciones para obtener conjuntos de dos clases (dejar los patrones de una clase como *clase minoritaria* y realizar la unión de los patrones del resto de las clases para formar la *clase mayoritaria*). La descripción de estas adaptaciones puede observarse en la Tabla 5.2.

Tabla 5.2. Conversión de bases de datos de varias clases en conjuntos de dos clases

	Clases que forman cada grupo	
	Minoritaria	**Mayoritaria**
Glass	6	1, 2, 3, 4, 5
Vehicle	1	2, 3, 4
SatImage	4	1, 2, 3, 5, 6

b) Bases de datos desbalanceadas de más de dos clases. Estos conjuntos cuentan con patrones distribuidos en varias clases (5 y 11 clases) y, además, no existe balance en cuanto a la cantidad de patrones que la representan (Tabla 5.3).

Tabla 5.3. Bases de datos de más de dos clases con desbalance

	No. Clases	**CE**	**CT**
Cayo	11	3086	2933
Felt	5	1418	5820

c) Bases de datos balanceadas. En este tipo de conjuntos, la cantidad de patrones contenidos en una clase es igual, o con poca diferencia en cuanto a tamaño, con respecto a la cantidad de los patrones contenidos en la otra clase. La cantidad de clases contenidas por muestra varía desde dos hasta 11 clases. En concreto, los conjuntos pertenecientes a este grupo son Cancer, Heart, Liver, Pima, Sonar, Iris, Liver, Pima, Vehicle, German, Segment, Waveform, y Wine.

5.3 SISTEMAS MÚLTIPLES DE CLASIFICACIÓN

En la vida real, muchas bases de datos se caracterizan por presentar desbalance en la distribución de los patrones entre las clases (ver Sección 3.5.3), requiriendo especial interés en la correcta clasificación de los patrones contenidos en la(s) clase(s) minoritaria(s), tal es el caso de las bases de datos de información médica. Por ejemplo, en [Cha., 00] se utiliza una base de datos de mamografías, con 11443 patrones, de los cuales tan solo 260 corresponden a casos de pacientes con micro-calcificaciones en la mama. Al clasificar nuevos casos, la precisión observada es de 97.68% clasificando los nuevos casos como sanos. Esta situación no es nada

deseable, ya que muchos de los casos que presentan micro-calcificaciones no fueron detectados como tal.

Con la finalidad de disminuir el efecto negativo del desbalance, se propone una modificación a la ejecución de los algoritmos existentes para lograr diversidad (selección aleatoria sin reemplazo, Bagging, Boosting, Arc-x4), en el sentido que la selección de patrones se realiza contemplando la distribución existente en cada una de las clases, con dos modalidades: submuestras resultantes con un balance total y submuestras con la misma representatividad por clase que el CE original.

Sin perder de vista los objetivos trazados previamente, el tamaño de las submuestras es un aspecto sumamente cuidado al momento de su integración. La mayoría de los métodos descritos en la Sección 4.1.1 consideran la creación de varias submuestras del mismo tamaño que el CE original. Sin embargo, esta situación es altamente costosa desde el punto de vista computacional, por lo que en los experimentos realizados, el tamaño de cada submuestra es de m/L, donde m es la cantidad de patrones de entrenamiento y L la cantidad de submuestras a formar. Con esta medida, se pretende reducir el posible elevado costo computacional en términos de tiempos de ejecución y requerimientos de memoria [Val., 05b].

En lo que respecta a la toma de decisiones, los experimentos incluyen estrategias de fusión de clasificadores [Bar., 03a], [Val., 05a]. En este sentido, y atendiendo a las principales desventajas detectadas a la votación simple, se implementan algunos métodos de ponderación estática y ponderación dinámica. En los métodos dinámicos, los pesos asignados a los clasificadores individuales pueden cambiar con cada uno de los patrones de prueba evaluados. Caso contrario sucede en la ponderación estática, en la que los pesos son calculados para cada clasificador en la fase de entrenamiento y se mantienen durante la clasificación de los patrones de prueba.

5.4 METODOLOGÍAS PROPUESTAS

Cada uno de los objetivos planteados al inicio de la presente investigación requiere que se establezcan metodologías de trabajo que permitan su cumplimiento. Por ello, a continuación se desglosan las metodologías propuestas para cada uno de los problemas a resolver.

5.4.1 DIVERSIDAD EN UN SMC

Uno de los principios que rigen y favorecen el buen desempeño de los SMC es el grado de diversidad existente entre los componentes que lo forman. Esta diversidad puede ser vista desde dos perspectivas: diversidad en el entrenamiento del sistema y diversidad en las decisiones que proporciona cada uno. Podemos decir que la

diversidad en el entrenamiento nos conduce forzosamente a la diversidad en las decisiones, sin embargo, experimentalmente comprobaremos que esta relación no siempre es verdadera. Para esto, las estrategias de solución propuestas están encaminadas hacia ambos sentidos: generar diversidad en el entrenamiento del SMC y analizar las decisiones proporcionadas por cada uno de sus elementos. Para lograr lo primero, acudimos a la implementación de cuatro métodos para realizar el submuestreo (ver Sección 4.1.1). En tanto que, para analizar la diversidad en las decisiones individuales, se utilizaron métodos ya existentes para este fin (ver Sección 4.3).

5.4.2 ANÁLISIS DE LA EFICIENCIA DE LA REGLA 1-NN CON SMC

La regla de decisión 1-NN es una de las más ampliamente estudiadas en reconocimiento de patrones, sin embargo, su eficiencia ha sido ampliamente cuestionada cuando se aplica sobre conjuntos que presentan problemas como desbalance entre clases, tamaño excesivo, patrones atípicos, ruidosos y/o redundantes. Por tal motivo, en la presente Tesis Doctoral, se propone la implementación de diferentes metodologías que incluyen la utilización de un SMC basado en la regla 1-NN y que, al mismo tiempo, implementa algoritmos de preprocesado (ver secciones 3.5.4 y 3.5.5) con intención de disminuir y, en algunos casos, eliminar los efectos negativos que estos problemas ocasionan. Las estrategias de solución aquí propuestas pueden dividirse de acuerdo al problema objetivo de la siguiente manera:

a) Tratamiento del desbalance en bases de datos de dos clases. Para dar solución a este problema, se proponen tres diferentes estrategias: utilizar un SMC para obtener un balance perfecto en los subconjuntos de datos y, posteriormente, aplicar algoritmos de preprocesado con cuatro diferentes métodos: metodología 1: clasificar directamente con el SMC sin preprocesado, metodología 2 (Figura 5.2): limpieza de submuestras con edición de Wilson, metodología 3 (Figura 5.3): reducción de submuestras con SSM, y metodología 4 (Figura 5.4): reducción de submuestras previamente editadas; sobre-entrenar la clase menos representada y, posteriormente, aplicar algoritmos de preprocesado (con el mismo esquema de las Figuras 5.2, 5.3 y 5.4, con la variante que en lugar de utilizar el SMC, se utilizan los conjuntos balanceados después de sobre-entrenar la clase minoritaria); y favorecer la clase minoritaria mediante la ponderación de la distancia euclídea.

b) Tratamiento del desbalance en bases de datos de más de dos clases. Además del problema del desbalance, en este tipo de conjuntos, es necesario determinar la forma como se tratarán cada una de las clases: mayoritarias, minoritarias o intermedias. Para esto, se propone establecer un umbral que nos permita hacer la discriminación entre clases mayoritarias y clases

minoritarias. Además, se ve la conveniencia de utilizar algoritmos de preprocesado, con la finalidad de elevar la calidad del conjunto de datos.

c) Tratamiento de bases de datos con tamaño excesivo. Uno de los problemas más importantes de la regla 1-NN es el tiempo que requiere para su ejecución en grandes conjuntos. Si a esto le sumamos la dificultad de tratar grandes cantidades de datos de forma conjunta, tenemos que la complejidad de la solución se incrementa considerablemente. La solución aquí propuesta incluye explotar las bondades que los SMC nos ofrecen al permitir la partición del CE en subconjuntos de menor tamaño, y la reducción asociada a los algoritmos de limpieza y reducción (ver Sección 3.5.5 y 3.5.6) (Figuras 5.2, 5.3 y 5.4).

d) Eliminación de la redundancia en la base de datos. Un problema derivado de la utilización de algoritmos que realizan la selección de patrones con reemplazo (ver Sección 4.1.1) es la presencia de patrones redundantes. Estos patrones, quizás no afecten a los índices de precisión, pero sí dificultan el tratamiento del conjunto de datos elevando los tiempos requeridos por la regla 1-NN durante la clasificación. La solución propuesta contempla la eliminación de los patrones repetidos, conservando únicamente un representante de ellos.

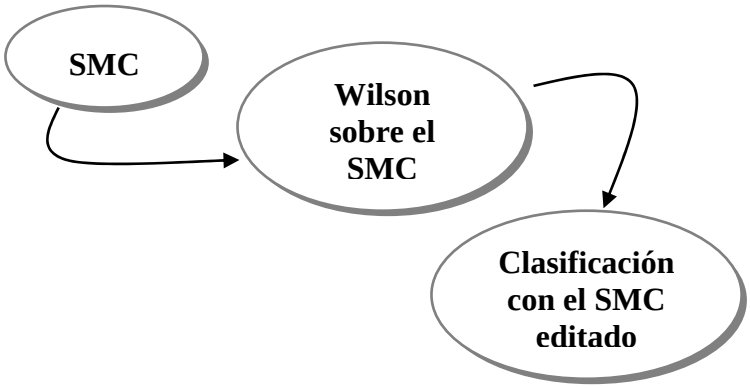


Figura 5.2. Procesos que incluye la metodología 2. Al SMC se le aplica el algoritmo de Edición de Wilson y, posteriormente, se clasifica con la regla 1-NN utilizando el SMC editado. Con esta metodología, se busca eliminar todos aquellos patrones atípicos y/o ruidosos contenidos en el CE y, de este modo, incrementar la precisión en la clasificación

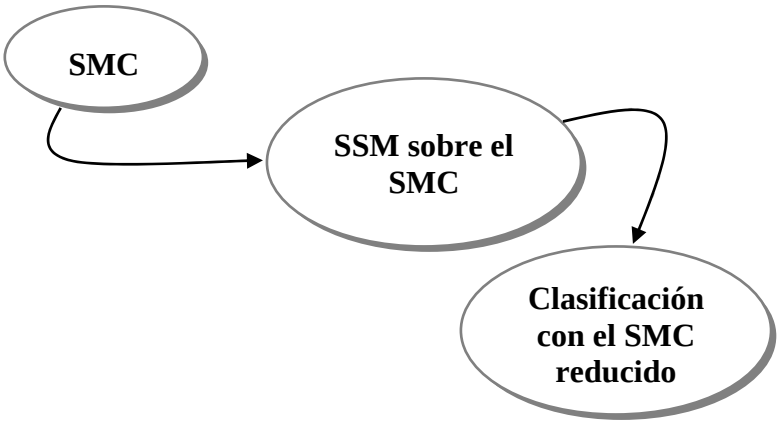


Figura 5.3. Procesos que incluyen la metodología 3. Similar a la metodología 1, aunque en este caso, antes de la fase de clasificación, se ejecuta el algoritmo de reducción SSM

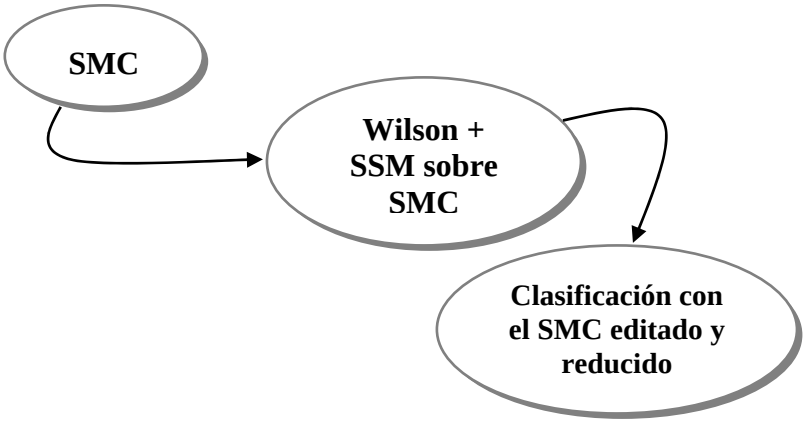


Figura 5.4. Procesos que incluyen la metodología 4. En esta metodología, se ejecutan los dos algoritmos de preprocesado (edición de Wilson y SSM) sobre el SMC antes de iniciar la clasificación de nuevos patrones.

5.4.3 FUSIÓN DE CLASIFICADORES

Como se ha mencionado en temas anteriores, uno de los métodos más utilizado para realizar la combinación de las decisiones individuales es la votación por mayoría simple. Sin embargo, estudios realizados han demostrado la gran debilidad que este método tiene cuando el desempeño de los componentes individuales del SMC no es uniforme [Mat., 96], [Han., 90], [Val., 05a], [Val., 06].

La estrategia de solución propuesta para resolver este problema consiste en la implementación de nuevos métodos para ponderar las decisiones de los clasificadores que forman el SMC. Estos métodos realizan la ponderación de dos diferentes formas: dinámica y estática. Los métodos que ponderan dinámicamente las decisiones de los clasificadores hacen uso de las distancias existentes entre los vecinos más cercanos en los componentes del SMC y el patrón de entrada. Por el contrario, el método que realiza la ponderación estática apoya su funcionamiento en la estimación del error que se obtiene por cada uno de los clasificadores que forman el SMC, el cual será la base para el establecimiento de los pesos.

5.4.4 ANÁLISIS DE LA EFICIENCIA DE SMC FORMADOS POR RNA

Un paradigma que ha sido extensamente estudiado a través de los años, se refiere a las redes neuronales. Diferentes y variados métodos de aprendizaje supervisado han sido propuestos pero, definitivamente el algoritmo *backpropation* se constituye como el más estudiado, evaluado e implementado en diversas áreas del conocimiento. Otros métodos relativamente nuevos, que prometen grandes beneficios a la clasificación de patrones, son los que contempla el principio de modularidad. Algunos de los aspectos que los hace atractivos son entre otros: simplicidad computacional, trabajo independiente, velocidad de procesamiento, y la posibilidad de descomponer problemas complejos en problemas más simples (ver Sección 3.4.2.1).

Como parte final, en la presente Tesis Doctoral, se realiza un amplio estudio del desempeño que un SMC formado por un conjunto de perceptrón múlticapa y otro por redes modulares, tienen cuando se ejecutan en condiciones similares a las contempladas con la regla 1-NN. Se analiza también el desempeño obtenido con cada uno de los SMC formado con estas dos estructuras de red, en términos de precisión y tiempos de ejecución.

Capítulo 6

Diversidad en SMC

Tal como se menciona en [Kun., 01c], es indispensable que un SMC presente un determinado grado de diversidad en los componentes individuales para que, al combinar las decisiones individuales, se disminuya o elimine el efecto negativo de los errores individuales. En la presente investigación, la diversidad es abordada desde dos perspectivas: la diversidad en los componentes individuales (en su construcción) y la diversidad existente en las decisiones individuales.

6.1 CREACIÓN DE SMC DIVERSOS

Existen varios métodos que nos permiten lograr cierto grado de diversidad al construir un SMC (ver sección 4.1): manipulación de patrones, manipulación de atributos, manipulación de las etiquetas de clase, diferentes clasificadores con un mismo CE e inyección de aleatoriedad. Para los experimentos reportados en esta Tesis Doctoral, las submuestras que forman el SMC se integraron con diferentes métodos que realizan el submuestreo o la manipulación de patrones: Bagging, Boosting, Arc-x4, selección aleatoria sin reemplazo y selección secuencial. La

implementación de estos métodos varía de acuerdo a la distribución de los patrones existente en las clases que forman la base de datos. Así, para las base de datos que cuentan con desbalance, la selección de patrones favoreció la creación de submuestras con balance perfecto, en tanto que, para los CE que no cuentan con desbalance muy fuerte, las submuestras resultantes mantienen la misma representatividad por clase que el conjunto original. Estas dos cuestiones serán tratadas más ampliamente en el Capítulo 7.

6.1.1 SUBMUESTREO INDISCRIMINADO Y SUBMUESTREO POR CLASE

La implementación tradicional de los métodos para realizar el submuestreo (seleccionar patrones) contempla la construcción de submuestras del mismo tamaño que el conjunto original [Die., 97]. Esta situación propicia tres importantes cuestiones sobre las submuestras resultantes: distribución de patrones diferente a la existente en la base de datos original (debido a la selección indiscriminada de patrones entre clases), alto coste computacional y gran cantidad de memoria requerida para su tratamiento en conjunto. Al momento de clasificar patrones, estas cuestiones son poco deseadas, ya que merman la efectividad del algoritmo de clasificación.

Para eliminar estos problemas, la mayoría de las submuestras utilizadas en los experimentos de esta tesis consideran el tamaño de cada una de sus clases de acuerdo a su distribución, es decir, m_c/H, donde m_c es la cantidad de patrones de una determinada clase y H es el número de clasificadores que formarán el SMC [Val., 05b]. De este modo, el tamaño final de cada una de las submuestras que forman el SMC es de m/H, donde m es la cantidad de patrones del CE.

Para los experimentos de esta sección, se utilizó un SMC formado por 5, 7 y 9 clasificadores, los cuales utilizaron los métodos de submuestreo Bagging (a), Boosting (b) y Arc-x4 (c). Finalmente, se utilizaron tres esquemas para fusionar las decisiones: votación por mayoría simple, la votación ponderada por promedio de distancias y ponderación con Shepard modificado (ver Secciones 8.1, 8.2.2 y 8.2.8).

Las Tablas 6.1 y 6.2 muestra los mejores resultados obtenidos con submuestras formadas mediante submuestreo clásico y utilizando el submuestreo por clase (SubC). Los números de la segunda fila indican el número de clasificadores que integran el SMC. Para cada caso, hemos incluido el método de submuestreo que proporciona el mejor resultado. Para obtener un mayor detalle de los resultados, pueden consultarse los anexos del Apéndice II.

A partir de estos resultados, podemos ver que para todas las bases de datos, al menos un SMC tiene mayor precisión que cuando se utiliza un clasificador único. Por otro lado, al comparar los tres métodos de submuestreo (Bagging, Boosting y

Arc-x4), en general, Bagging supera el desempeño de los otros dos, independientemente del esquema de votación utilizado. En lo que respecta a los resultados con los dos esquemas de votación, no encontramos diferencias significativas.

Tabla 6.1. Submuestreo con distribución por clase y submuestreo indiscriminado (votación por mayoría simple)

		CE original	5	7	9
Cancer	**Clásico**	95.6	(a) 95.6	(a) 95.8	(c) 95.8
	SubC		(a) 95.3	(a) 96.2	(a) 96.4
Heart	**Clásico**	58.2	(a) 59.3	(a) 60.4	(b) 58.9
	SubC		(a) 64.4	(a) 64.4	(b) 63.0
Liver	**Clásico**	65.2	(b) 63.8	(a) 66.1	(a) 65.8
	SubC		(a) 65.8	(a) 64.1	(b) 65.2
Pima	**Clásico**	65.9	(a) 66.7	(a) 67.5	(a) 66.9
	SubC		(b) 71.1	(a) 71.2	(a) 72.7
Sonar	**Clásico**	82.0	(a) 82.4	(a,b) 81.0	(c) 82.4
	SubC		(a) 73.7	(c) 71.7	(c) 72.2
Glass	**Clásico**	70.0	(a) 70.5	(c) 69.0	(c) 69.5
	SubC		(a) 65.0	(b) 63.0	(b) 61.0
Iris	**Clásico**	96.0	(a) 96.0	(a) 96.0	(a,b,c) 96.0
	SubC		(a) 96.0	(c) 98.0	(a,b) 94.0
Vehicle	**Clásico**	64.2	(b) 64.7	(a) 65.4	(a) 65.0
	SubC		(a) 62.0	(b) 60.7	(b) 62.3
Wine	**Clásico**	72.4	(a) 72.4	(a) 72.9	(b) 71.8
	SubC		(b) 70.0	(b) 73.5	(a) 75.9
German	**Clásico**	65.2	(a) 66.4	(a) 66.0	(a) 66.0
	SubC		(b) 68.8	(c) 69.9	(a) 70.2
Satimage	**Clásico**	83.6	(a) 83.6	(a) 83.6	(a) 83.5
	SubC		(b) 83.0	(b) 82.7	(a) 82.9
Phoneme	**Clásico**	76.1	(a) 76.0	(a) 76.5	(a) 75.9
	SubC		(a) 75.0	(a) 75.0	(a) 75.0
Waveform	**Clásico**	78.0	(a) 78.1	(a) 78.6	(a) 78.1
	SubC		(a) 80.7	(a) 81.9	(a) 83.2
Segment	**Clásico**	94.8	(b) 94.9	(b) 94.6	(c) 94.9
	SubC		(a) 89.8	(b) 87.1	(a) 87.7

Por otro lado, en lo concerniente a la forma como se seleccionan los patrones (en todo el CE o por clase), vemos que con Waveform, German, Pima y Cancer, la precisión obtenida cuando se seleccionan los patrones de forma indiscriminada, siempre es peor que la proporcionada con submuestras que seleccionan los patrones considerando la distribución por clase, independientemente del esquema de fusión utilizado. Estas diferencias son más notorias con problemas que tienen un elevado desbalance entre clases. Por ejemplo, para la base de datos Pima, con un 65% de patrones en la clase mayoritaria, la precisión es de 5-6% mayor cuando se

considera la distribución por clase. En el resto de los casos, existen diferencias poco significativas en los índices de precisión que se obtienen con SMC formados con el submuestreo por clase respecto a cuando se realiza el submuestreo sobre todo el CE. Esta situación igualmente hace recomendable la utilización de la propuesta aquí hecha, al contemplar los elevados requerimientos que el submuestreo clásico tiene asociado y el grado de desbalance en las submuestras resultantes (Apéndice II). En este sentido, la Tabla 6.3 muestra la cantidad de patrones utilizado por cada una de las alternativas.

Tabla 6.2. Submuestreo con distribución por clase y submuestreo indiscriminado (votación ponderada dinámica por promedio)

		CE original	5	7	9
Cancer	**Clásico**	95.6	(b) 95.5	(b) 95.0	(b) 95.0
	SubC		(a) 95.3	(a) 96.2	(a) 96.2
Heart	**Clásico**	58.2	(a) 59.6	(a) 60.0	(b) 59.3
	SubC		(a) 64.8	(a) 66.7	(b) 63.4
Liver	**Clásico**	65.2	(c) 64.5	(a) 65.8	(a) 65.8
	SubC		(a) 64.9	(a) 63.5	(c) 66.1
Pima	**Clásico**	65.9	(a) 66.7	(a) 67.6	(a) 67.1
	SubC		(b) 71.9	(a) 70.9	(a) 72.7
Sonar	**Clásico**	82.0	(b) 82.4	(b) 81.5	(c) 82.9
	SubC		(a) 77.6	(c) 75.6	(c) 74.2
Glass	**Clásico**	70.0	(a) 70.5	(c) 69.5	(c) 70.0
	SubC		(c) 65.0	(c) 64.5	(a) 64.0
Iris	**Clásico**	96.0	(a,c) 96.0	(a,c) 96.0	(a,b,c) 96.0
	SubC		(b) 97.3	(c) 98.0	(a) 95.3
Vehicle	**Clásico**	64.2	(b) 64.6	(a) 65.7	(a) 65.2
	SubC		(b) 63.4	(a) 62.3	(b) 63.9
Wine	**Clásico**	72.4	(a) 73.5	(a) 72.9	(a,b) 71.8
	SubC		(b) 67.7	(a,b) 71.8	(a) 77.7
German	**Clásico**	65.2	(a) 66.3	(a) 65.3	(a) 66.3
	SubC		(b) 68.6	(b) 69.3	(a) 70.8
Satimage	**Clásico**	83.6	(a) 83.7	(a) 83.6	(c) 83.7
	SubC		(b) 83.1	(a,c) 82.6	(b) 83.4
Phoneme	**Clásico**	76.1	(a) 76.0	(a) 76.4	(c) 76.0
	SubC		(a) 75.0	(a) 75.5	(a) 75.2
Waveform	**Clásico**	78.0	(a) 78.1	(a) 78.6	(a) 78.1
	SubC		(a) 80.7	(a) 91.9	(a) 83.2
Segment	**Clásico**	94.8	(b) 94.9	(b) 95.0	(c) 95.0
	SubC		(a) 92.1	(a) 90.4	(a) 90.7

El tamaño de las submuestras resultantes es importante en el sentido que, cuando se utilizan submuestras de poco tamaño, se requieren pocos recursos computacionales (en tiempo y espacio de almacenamiento) para su tratamiento. Es decir, mientras la selección clásica de patrones (indiscriminada) produce nueve

submuestras de tamaño *m*, la selección por clase crea nueve submuestras de tamaño *m*/9, y por lo tanto, el total de patrones en un SMC donde las submuestras se formaron con la selección indiscriminada es H veces el tamaño del CE original.

Tabla 6.3. Tamaños de los conjuntos originales y de las submuestras formadas mediante submuestreo por clase

	CE original	Clasificadores		
		5	7	9
Cancer	546	2730	3822	4914
Heart	216	1080	1512	1944
Liver	276	1380	1932	2484
Pima	615	3075	4305	5535
Sonar	167	835	1169	1503
Glass	174	870	1218	1566
Iris	120	600	840	1080
Vehicle	678	3390	4746	6102
Wine	144	720	1008	1296
German	800	4000	5600	7200
Satimage	5147	25735	36029	46323
Phoneme	4322	21610	30254	38898
Waveform	4000	20000	28000	36000
Segment	1848	9240	12936	16632

En este contexto, en los tamaños de los SMC encontramos diferencias muy significativas. Por ejemplo, para Heart, el CE original consiste de 216 patrones (esos son patrones usados por la regla 1-NN), el SMC compuesto por 9 submuestras y formado por la selección clásica utiliza 1944 patrones (9 veces 216 patrones), en tanto que, el SMC con submuestreo por clase utiliza 216 patrones (la misma cantidad de patrones que el clasificador único). Otro ejemplo más demostrativo es con Waveform, el CE original consta de 4000 patrones, el número total de patrones seleccionados por el método clásico es 36000, y mientras que el utilizado por el método propuesto utiliza 4000.

6.1.2 DETERMINACIÓN DEL NÚMERO DE CLASIFICADORES EN UN SMC

Cuando se desea construir un SMC, es necesario establecer la cantidad de elementos que lo formarán. En este sentido, actualmente no existe una regla que indique la cantidad de clasificadores necesarios para obtener un grado de diversidad suficiente con el que se obtengan resultados óptimos en todos los casos, y menos, que tengan en cuenta los diferentes aspectos que se pueden encontrar en un conjuntos de datos (ver Sección 3.5). Por ello, en esta sección, tratamos de analizar el efecto del número de clasificadores sobre la precisión general.

Para este fin, se utilizaron los catorce conjuntos de datos con los que ya se experimentó en la sección anterior, de los cuales algunos cuentan con dos y más clases, con desbalance, conjuntos con gran cantidad de patrones, etc. La cantidad de patrones que contiene cada una de sus clases se estableció como *m*/H, donde *m* es la cantidad de patrones de entrenamiento y H el número de clasificadores que formarán el SMC. La cantidad de componentes individuales fue variando en número impar con los siguientes valores: 3, 5, 7, 9, 15 y 25. De esta forma, los tamaños de los subconjuntos resultantes están directamente relacionados con el número de clasificadores utilizados, es decir, si se decide integrar 15 clasificadores, cada subconjunto contendrá 1/15 de los patrones del CE original. Sin embargo, esta situación resulta prohibitiva para CE con pocos patrones, pues cabe la posibilidad que en ocasiones, en algunas clases poco representadas, no se pueda seleccionar ni un único patrón. Por tal motivo, se incluyó la condición de que si esta situación se presentara, al menos se eligiese un patrón, garantizando de esta forma que todos y cada uno de los subconjuntos resultantes contengan al menos un patrón por clase.

6.1.2.1 Votación simple: bases de datos de dos clases

Los primeros experimentos de la presente análisis corresponden al método de votación simple en el caso de las bases de datos que constan únicamente de dos clases: Cancer, Heart, Liver, Pima, Sonar, Phoneme y German. Los resultados al clasificar nuevos patrones con estos conjuntos de datos y al variar el tamaño del SMC son mostrados en las gráficas de la Figura 6.1. Estas gráficas constan de dos ejes, uno de ellos para medir la precisión observada con cada una de las técnicas de selección de patrones y la segunda para visualizar el tamaño de cada conjunto de datos utilizado en las diferentes variantes del SMC.

De acuerdo a estos resultados, es posible apreciar un comportamiento poco uniforme en cuanto a técnica de integración de patrones y número de componentes individuales. Sin embargo, es de notarse el buen desempeño que presenta la selección de patrones con Bagging en la mayoría de los casos, principalmente con Cancer, Heart, Pima, Phoneme y German al utilizar de 7 a 25 clasificadores, con poca variación al utilizar 5 clasificadores. Situación opuesta se presenta cuando se utilizan únicamente 3 clasificadores, donde tanto la selección aleatoria sin reemplazo como Bagging proporcionan los peores resultados en la clasificación.

6.1.2.2 Votación simple: bases de datos de más de dos clases

En estos experimentos, se utilizaron los siete conjuntos de datos que cuentan con más de dos clases, con representatividad no uniforme entre los patrones de cada una de ellas. Al igual que para el caso de dos clases, los resultados se muestran en gráficas con dos ejes (Figura 6.2), uno de ellos para la precisión en la clasificación y el segundo para el tamaño de los conjuntos de datos.

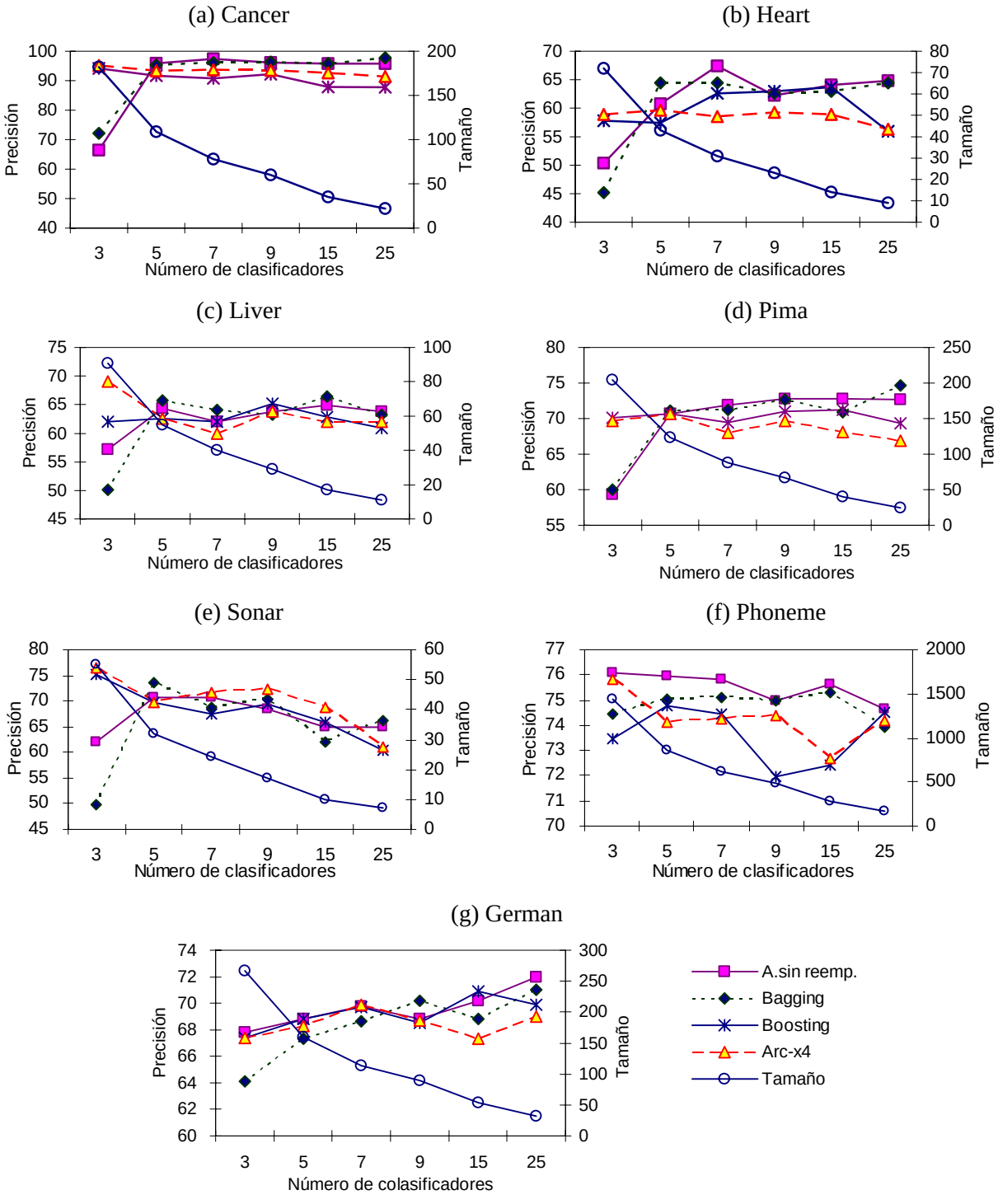


Figura 6.1. Clasificación con votación simple en bases de datos de dos clases

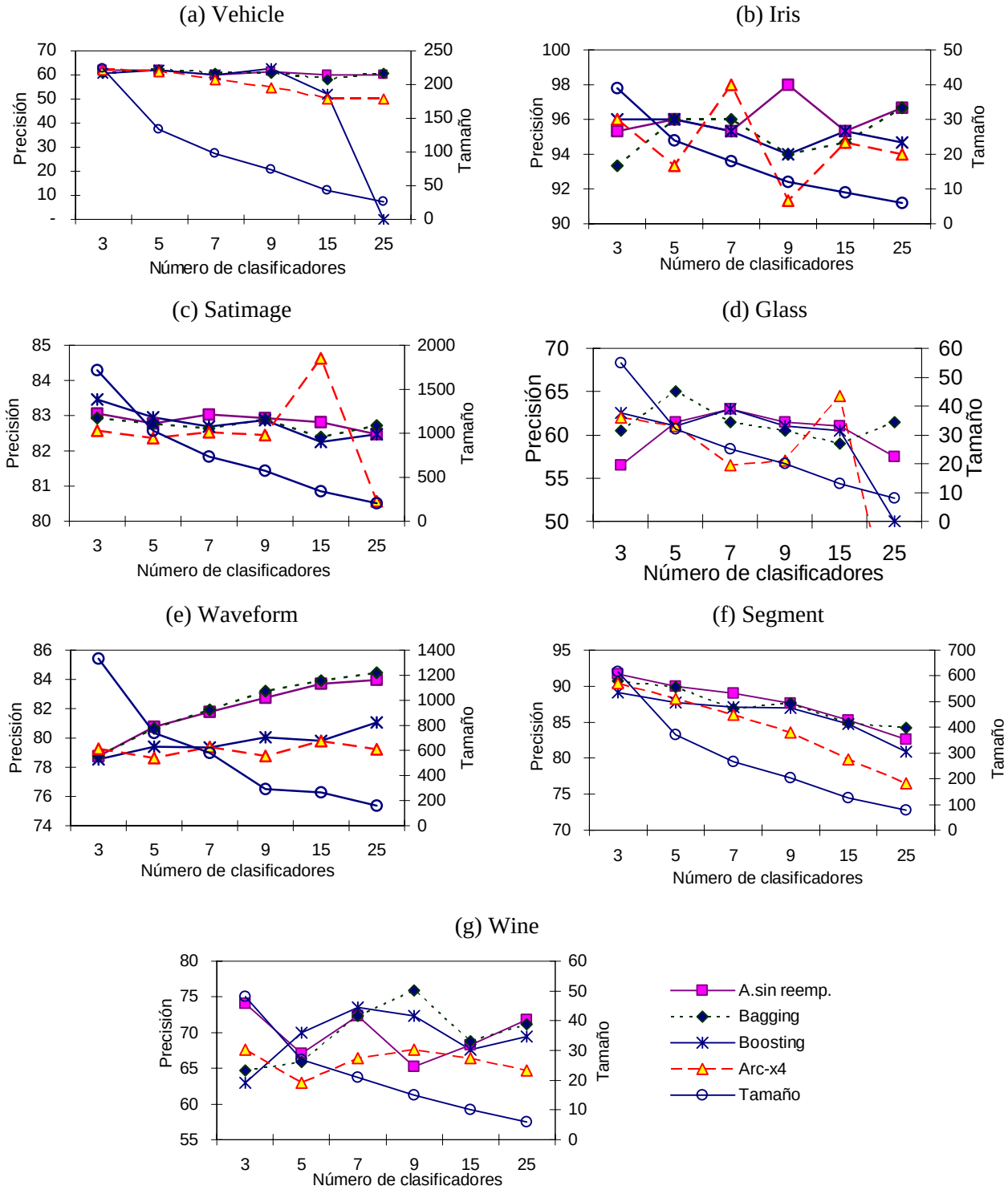


Figura 6.2. Clasificación con votación simple en bases de datos de más de dos clases

A partir de estos resultados, observamos que, para los conjuntos Glass, Vehicle y Satimage, se manifiesta un comportamiento estable y uniforme con todas

las técnicas de selección de patrones, presentando únicamente una ligera variación al utilizar 15 clasificadores, donde la técnica Arc-x4 incrementa considerablemente la precisión en los conjuntos Glass y Satimage. Es importante recalcar que, al utilizar 25 submuestras, la precisión sufre un deterioro significativo con los métodos Arc-x4 (Glass y Satimage) y Boosting (Vehicle y Glass). Por otra parte, este decremento pudiera deberse a la cantidad de clasificadores integrados, ya que estas técnicas integran clasificadores mientras la estimación del error sea menor a 0.5, tal como se muestra en la Tabla 6.6 y 6.7.

En el caso de Wine e Iris, no logra apreciarse con claridad un desempeño adecuado de las diferentes técnicas de integración de datos por cada situación en el de número de elementos individuales del SMC. Las variaciones negativas más importantes se observan con las técnicas Arc-x4 con 5 y 9 clasificadores, en tanto que el mejor resultado se obtuvo con 9 clasificadores utilizando Bagging en la base de datos Wine y con 7 y 9 clasificadores mediante Arc-x4 y aleatoria sin reemplazo, respectivamente, en Iris.

Un comportamiento muy peculiar es el observado con por los conjuntos de datos Waveform y Segment. En el primero de ellos, la precisión va en aumento conforme se incrementa el número de clasificadores (requiere poca cantidad de patrones para un comportamiento eficiente); en el segundo, la precisión va en decremento, lo que nos indica la necesidad de contar con mayor cantidad de patrones, a fin de tener una precisión igual o superior al 90%.

Otro aspecto importante de analizar, se refiere al tamaño de las submuestras utilizadas por cada SMC. Como era de esperar, el tamaño del conjunto de datos disminuye conforme el número de clasificadores aumenta, aspecto de suma importancia al considerar el costo computacional requerido por cada SMC, ya que al evaluar el binomio precisión - costo computacional, resulta muy ventajosa la utilización de un mayor número de clasificadores para fines de clasificación.

A modo de resumen, podemos ver en la Tabla 6.4 el comportamiento general de las técnicas de integración de patrones con cada SMC. El valor de cada celda indica el total de ocasiones en las que la técnica correspondiente obtuvo el mejor resultado respecto a las demás. Los valores entre paréntesis indican la cantidad de veces en que el mejor resultado de una determinada técnica fue igualado por alguna(s) otra(s). Por ejemplo, en el caso del método “Aleatorio sin reemplazo” utilizando 15 clasificadores, de los catorce conjuntos de datos, esta técnica obtuvo en 7 de ellos el mejor resultado, dos de los cuales fueron empatados por los métodos Bagging y Boosting, uno por cada una de ellas. Con esto, es posible ver un índice de precisión más competitivo entre las técnicas que seleccionan patrones de manera aleatoria sin reemplazo y Bagging, dejando por debajo a Arc-x4 y Boosting.

Tabla 6.4. Mejores resultados por técnica de selección de patrones y por número de clasificadores utilizado (votación simple)

	3 Clasif.	5 Clasif.	7 Clasif.	9 Clasif.	15 Clasif.	25 Clasif.
Aleatorio sin reemplazo	4 (1)	5 (1)	6 (1)	5 (1)	5 (2)	5 (2)
Bagging	---	4 (1)	3	5	3 (1)	7 (2)
Boosting	3 (1)	4 (1)	1 (1)	2	1 (1)	---
Arc-x4	5 (2)	---	3	1 (1)	3	---

6.1.2.3 Votación ponderada por promedio: bases de datos de dos clases

En esta sección, se trata de analizar el comportamiento de los SMC en función del número de clasificadores que los componen pero, en este caso, utilizando el método de ponderación dinámica en la toma de decisiones. Estos primeros experimentos se centran en las bases de datos compuestas de dos clases, igual que se ha hecho en la Sección 6.1.2.1.

Al analizar los resultados de la Figura 6.3, vemos que, en comparación con la votación simple, los conjuntos de datos Phoneme, German y Sonar tienen un comportamiento muy similar. Las principales diferencias radican en la estabilidad que el método de selección aleatoria sin reemplazo tiene con respecto al número de clasificadores, y el pobre rendimiento inicial de las técnicas Boosting y aleatorio con reemplazo al utilizar 3 clasificadores.

6.1.2.4 Votación ponderada por promedio: bases de datos de más de dos clases

En los resultados de la Figura 6.4, encontramos unas leves diferencias en cuanto al comportamiento de estas mismas bases de datos al utilizar la votación simple. Las principales mejoras son obtenidas cuando se utilizan 9 clasificadores con las bases de datos Iris (Arc-x4), Satimage (Boosting) y Bagging (Glass), en las cuales se tiene un incremento en los índices de precisión. El resto de las bases de datos mantienen un comportamiento similar al observado con la votación simple. Al igual que con la votación simple, el método de selección aleatoria sin reemplazo mantiene su estabilidad frente a variaciones en el número de clasificadores.

Al igual que en la votación simple, la Tabla 6.5 muestra de manera resumida los mejores resultados obtenidos por cada una de las técnicas de selección de patrones con los diferentes números de clasificadores. Los valores que se encuentran entre paréntesis corresponden a empates en el mejor resultado con otra técnica, es decir, las dos técnicas obtuvieron la mejor precisión sobre un mismo conjunto de datos.

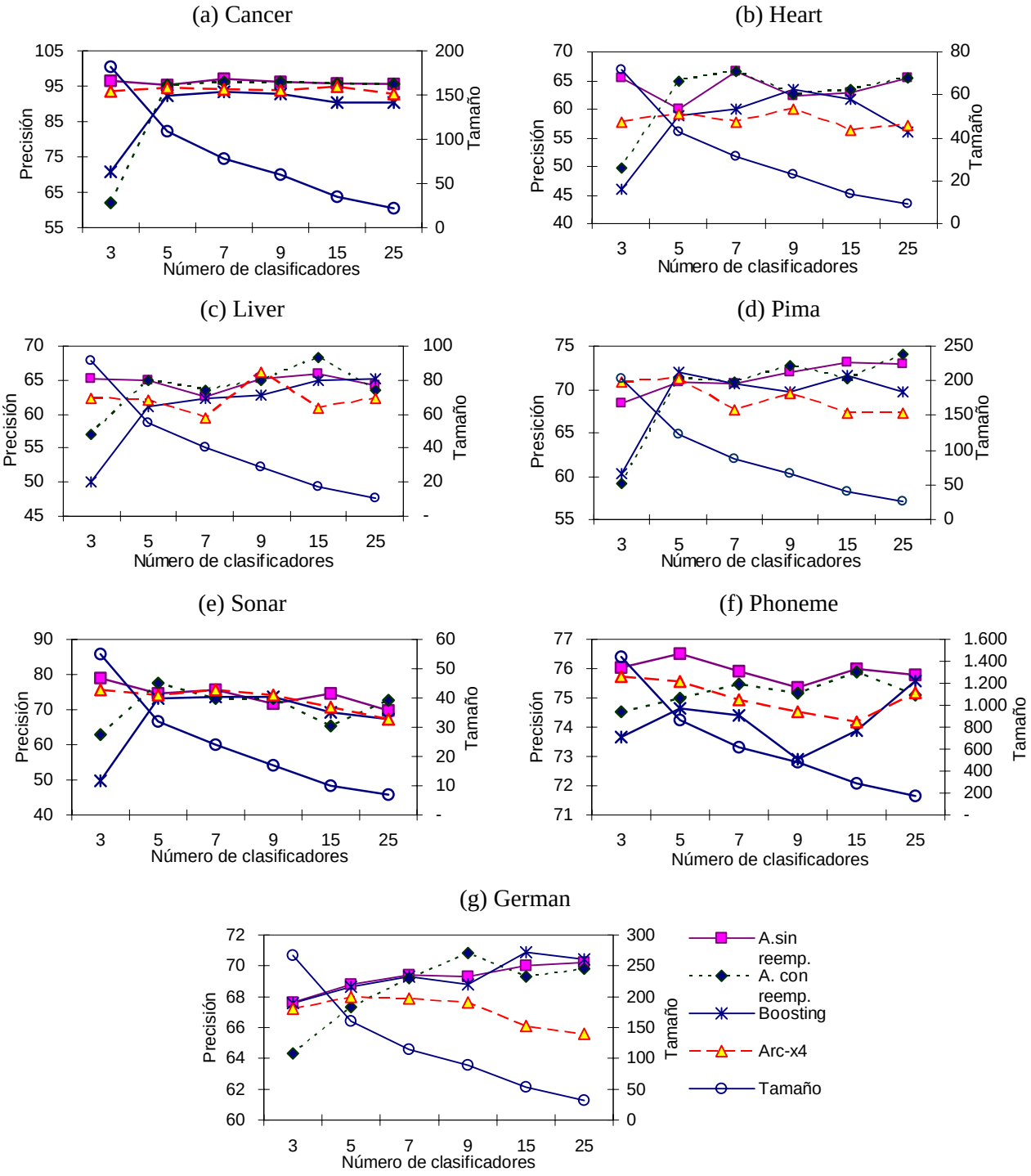


Figura 6.3. Votación ponderada por promedio con bases de datos de dos clases

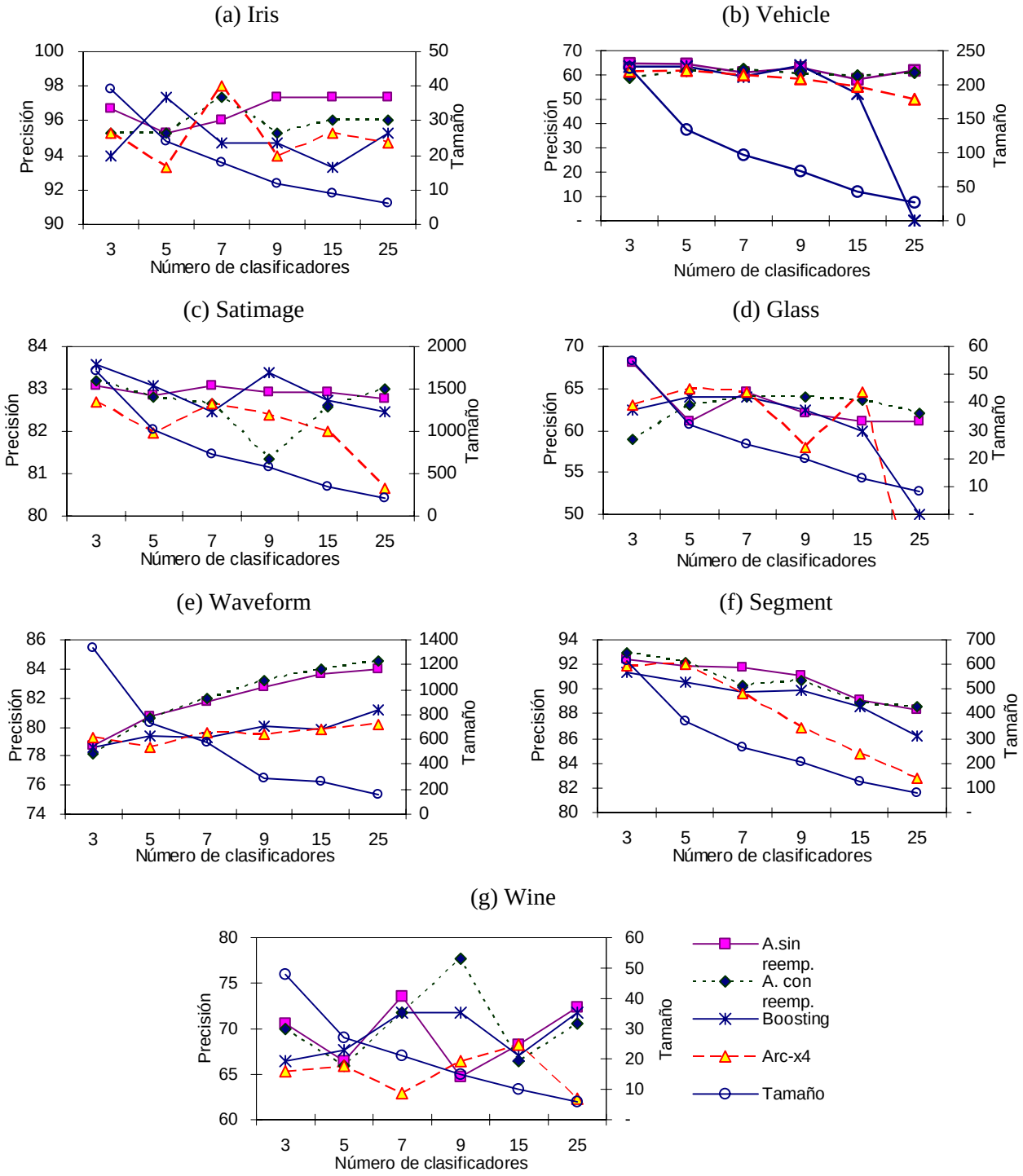


Figura 6.4. Votación ponderada por promedio en bases de datos de más de dos clases

Tabla 6.5. Mejores resultados por técnica de selección de patrones y por número de clasificadores utilizado (votación ponderada por promedio)

	3 Clasif.	5 Clasif.	7 Clasif.	9 Clasif.	15 Clasif.	25 Clasif.
Aleatorio sin reemplazo	10	4 (2)	6 (3)	3 (1)	6 (2)	4 (2)
Bagging	1	3 (2)	4 (1)	5 (1)	5 (1)	7 (2)
Boosting	1	4	---	3	0 (1)	1
Arc-x4	2	1	1 (2)	2	1	---

En esta tabla, se ve más claramente los eficientes resultados de la selección sin reemplazo de patrones al utilizar 3, 5, 7 y 15 clasificadores. Excepcionalmente se ve superada por Bagging cuando se utilizaron 9 y 25 clasificadores.

6.1.2.5 Clasificadores integrados con Arc-x4 y Boosting

Si recordamos, los métodos que utilizan el principio de Arcing cuentan con una condición de parada: producir submuestras mientras el error individual sea inferior a 0.5. De esta forma, pueden existir casos en los que se logre integrar la cantidad de submuestras deseada pero, por el contrario habrá también otros casos (muy frecuentes) en los que la cantidad de submuestras creada sea inferior a la deseada. En las Tablas 6.6 y 6.7, podemos observar la cantidad de clasificadores integrados al utilizar los algoritmos Arc-x4 y Boosting.

Tabla 6.6. Clasificadores integrados mediante Boosting y Arc-x4 (SMC de 3, 5 y 7 clasificadores)

	3 Clasificadores		5 Clasificadores		7 Clasificadores	
	Arc-x4	Boosting	Arc-x4	Boosting	Arc-x4	Boosting
Cancer	3,3,3,3,3	3,3,2,3,3	5,5,4,5,4	5,3,5,5,4	5,4,5,5,4	4,5,4,3,6
Heart	3,3,3,3,3	3,3,3,3,3	4,5,5,5,5	5,5,5,5,5	4,6,7,4,6	7,7,7,7,7
Liver	3,3,3,3,3	3,3,3,3,3	5,5,5,5,5	5,5,5,5,5,	7,7,7,4,7	7,7,7,7,5
Pima	3,3,3,3,3	3,3,3,3,3	4,4,4,5,5	5,5,5,5,5	4,3,4,6,4	7,7,7,7,7
Sonar	3,3,3,3,3	3,3,3,3,3	3,5,5,4,2	5,5,5,5,5	4,7,5,6,5	7,7,7,4,7
Glass	2,3,3,3,3	3,3,3,3,3	2,3,2,2,2	5,5,5,5,3	2,2,3,6,2	5,2,6,7,4
Iris	3,3,3,3,3	3,3,3,3,2	5,4,3,3,5	5,5,2,2,2	5,7,5,4,4	6,3,4,2,2
Vehicle	3,3,3,3,3	3,3,3,3,3	3,3,3,3,3	5,5,5,5,5	3,3,3,3,3	7,6,7,7,7
Wine	3,3,3,3,3	3,3,3,3,3	3,4,2,2,2	5,4,5,5,5	2,3,3,3,4	5,5,7,5,4
German	3,3,3,3,3	3,3,3,3,3	3,4,3,4,4	5,5,5,5,5	4,4,4,3,6	7,7,7,7,7
Satimage	3,3,3,3,3	3,3,3,3,3	4,4,4,4,5	5,5,5,5,5	3,3,4,4,5	6,7,7,7,7
Phoneme	3,3,3,3,3	3,3,3,3,3	5,5,4,5,5	5,5,5,5,5	6,6,6,5,7	7,7,5,7,7
Waveform	3,3,3,3,3	3,3,3,3,3	5,5,5,5,5	5,5,5,5,5	5,6,5,5,5	7,7,7,7,7
Segment	3,3,3,3,3	3,3,3,3,3	5,5,4,5,4	5,5,5,5,5	3,3,5,4,4	6,6,7,6,7

Tabla 6.7. Clasificadores integrados mediante Boosting y Arc-x4 (SMC de 9, 15 y 25 clasificadores)

	9 Clasificadores		15 Clasificadores		25 Clasificadores	
	Arc-x4	Boosting	Arc-x4	Boosting	Arc-x4	Boosting
Cancer	5,5,4,4,5	3,3,4,3,3	4,4,4,3,4	3,4,2,4,4	5,4,5,5,4	2,5,3,4,4
Heart	3,6,2,6,6	7,0,9,4,9	2,2,2,15,6	8,2,2,11,8	2,3,8,6,3	5,3,7,1,2
Liver	9,6,9,8,6	9,5,9,9,9	4,4,12,14,2	12,14,7,7,7	9,8,5,13,1	6,11,9,1,19
Pima	3,3,5,5,4	6,9,9,9,9	5,6,5,4,5	12,10,9,11,7	5,3,7,5,4	4,8,14,8,6
Sonar	2,4,7,7,9	7,9,9,9,9	4,8,2,3,3	12,13,11,8,3	3,6,3,8,8	7,9,8,2,5
Glass	1,1,2,2,2	1,6,8,5,5	1,2,2,3,3	3,7,4,4,2	0,0,1,0,0	0,0,0,0,1
Iris	4,5,5,4,6	3,2,4,5,4	5,4,9,4,5	8,2,2,1,7	5,5,5,4,12	3,2,4,4,4
Vehicle	2,3,2,2,3	6,6,6,7,6	3,2,1,2,0	10,0,1,2,1	1,1,0,0,1	0
Wine	2,3,2,3,4	6,3,4,6,7	1,2,2,2,2	4,7,5,4,5	2,2,2,2,3	8,3,5,1,2
German	4,3,3,3,4	9,7,9,9,9	4,4,3,3,3	9,10,8,8,10	4,4,3,2,4	10,14,5,11,13
Satimage	4,3,3,3,4	7,9,7,9,6	3,3,4,3,4	5,10,5,5,6	3,3,3,3,3	7,7,9,6,8
Phoneme	5,7,7,5,9	8,9,8,9,9	6,7,9,4,9	10,8,8,7,15	7,9,8,4,9	12,12,14,10,13
Waveform	5,5,4,6,4	9,9,9,9,9	7,9,7,5,5	15,15,7,13,15	6,4,5,7,10	15,17,18,17,13
Segment	3,4,3,4,4	7,6,9,7,8	4,3,3,3,5	10,8,7,8,7	3,5,3,4,3	9,5,6,8,5

Como podemos ver, cuanto mayor es la cantidad de clasificadores que se desea integrar, más frecuentes son los casos en los que se integran solamente un clasificador y, en algunos casos, incluso ninguno (Glass y Vehicle con 25 clasificadores). Por otro lado, es interesante ver que, para los CE de mayor tamaño, esta situación no es observada, si bien, se integran menos clasificadores que los deseados pero, en ningún caso, se llega al extremo de haber integrado sólo uno o ningún clasificador. Lo antes mencionado hace pensar que aquellos clasificadores con menos representatividad en sus clases son los que muestran peor desempeño y, por consiguiente, no se supera el umbral establecido por el Arcing.

Es importante mencionar que la precisión en la clasificación al utilizar subconjuntos de datos poco representados (utilizando 25 clasificadores) y en donde el SMC no cuenta con la totalidad de componentes individuales respecto a otros métodos, en un 80% de los casos los índices de precisión quedan por debajo de los resultados proporcionados por otros métodos donde igualmente se tienen subconjuntos con poca representatividad, pero el SMC cuenta con todos sus componentes individuales. En este sentido, es necesario recalcar la poca conveniencia de integrar un SMC con muchos componentes individuales y la utilización de métodos de Arcing cuando se trata de bases de datos con poco tamaño. Esta situación motiva en gran medida, la construcción de submuestras con un tamaño similar al CE original, independientemente de la cantidad de clasificadores utilizados.

6.2 DETERMINACIÓN DE LA DIVERSIDAD EN LAS DECISIONES

Uno de los aspectos fundamentales a considerar cuando se utilizan SMC se refiere a la diversidad en las decisiones individuales. Sin embargo, acerca de está "obligada" diversidad surgen algunos cuestionamientos: ¿qué tan fuerte es la relación existente entre esta diversidad y los índices de precisión general? ¿realmente cuando se tiene baja diversidad el índice de error es elevado? Y, ¿hasta qué grado, el algoritmo de aprendizaje, el método de submuestreo y el método de combinación se relacionan con la diversidad y la precisión general?

Atendiendo a algunos de estos cuestionamientos, se han desarrollado metodologías que demuestran que la diversidad no siempre es un buen indicador del desempeño del SMC. Shipp et. al [Shi., 02] encontraron poca evidencia de "alguna" relación entre diez métodos de combinación y las precisiones de los componentes individuales del SMC. De la misma forma, no encontraron mucha evidencia de la relación existente entre los métodos de combinación y las medidas de diversidad. Kuncheva et al [Kun., 02c], consideraron la posibilidad de una relación entre el algoritmo de clasificación y el grado de diversidad. En su trabajo, implementaron tres variantes de Boosting (agresiva, conservadora e inversa) y asociaron a la condición de parada, los valores de Q con la precisión durante el entrenamiento de un Perceptrón múlticapa y un clasificador cuadrático.

En el presente trabajo, se busca dar respuesta a los cuestionamientos formulados previamente y, en particular, cuando se utiliza la regla del vecino más cercano como algoritmo de clasificación. Para ello, se utilizan cuatro medidas de diversidad: Q-estadístico, coeficiente de correlación, medida de desacuerdo y medida de variabilidad.

Los resultados aquí reportados corresponden a experimentos realizados sobre las 14 bases de datos ya utilizadas en la Sección 6.1. Para formar las submuestras, se realizó mediante la manipulación de los patrones por cada una de las clases. El SMC contiene cinco, siete y nueve clasificadores individuales con cuatro variantes: Bagging, Boosting, Arc-x4 y selección aleatoria sin reemplazo.

Los resultados experimentales corresponden a los promedios de la precisión general con la fusión de clasificadores utilizando la votación simple y cuatro métodos de ponderación dinámica [Val., 05a]: ponderación según Dudani, ponderación por promedio de distancias, ponderación según Shepard y ponderación de Shepard modificada.

A fin de disponer de un análisis mucho más simple, en el presente apartado se muestran resultados concentrados a partir de los anexos del Apéndice II, los cuales pueden consultarse para mayor detalle.

6.2.1 Q - ESTADÍSTICO

Partiendo del supuesto que la mayor diversidad se obtiene con valores pequeños de Q (Sección 4.3.1) y analizando los resultados del Apéndice II, encontramos que, en lo que respecta al grado de diversidad obtenido con los diferentes SMC, éste está directamente relacionado con el número de clasificadores utilizado. Es decir, conforme se aumentan el número de clasificadores utilizado, el valor de Q disminuye, obteniendo de este modo decisiones más diversas con el SMC formado por 9 clasificadores que las obtenidas cuando se utilizan 5 o 7 clasificadores. De acuerdo a esto último y atendiendo a bases teóricas [Ban., 03], [Die., 97], los mejores índices de precisión global deberían ser encontrados con estos SMC. Sin embargo, analizando los resultados de la clasificación, encontramos que esta situación tan sólo se obtiene con la votación según Shepard en las bases de datos Satimage y Glass y, con Shepard modificada en Satimage.

Por otro lado, cuando vemos más ampliamente el comportamiento que el SMC formado por 9 clasificadores tiene con cada uno de los métodos de submuestreo, encontramos que éste obtiene tanto altos como bajos índices de precisión con valores bajos de Q. Para describir un poco más esta cuestión, a continuación se establecen una serie de situaciones mediante las cuales se pueda identificar más claramente la influencia que el grado de diversidad tiene sobre el desempeño de los SMC con las bases de datos aquí utilizadas.

6.2.1.1 Bajo Q y precisión elevada

Este estudio consiste en identificar, por cada uno de los cuatro métodos de selección de patrones (aleatorio sin reemplazo, Bagging, Boosting y Arc-x4), las situaciones donde se presenta un alto grado de diversidad y, al mismo tiempo, se obtiene una precisión elevada al clasificar nuevos casos. La Tabla 6.8 muestra esta situación. Esta tabla, al igual que las demás mostradas en esta sección, es extraída de las tablas de resultados contenidas en el Apéndice II.

En lo referente al método para fusionar las decisiones, tenemos que los resultados donde mayormente encontramos alguna relación entre un bajo valor de Q y los mejores índices de precisión son los obtenidos al ponderar las distancias por promedio de distancias (en 24 de un total de 56 casos), seguida inmediatamente por la ponderación con distancia inversa (22 casos).

En lo que respecta a las bases de datos donde más veces encontramos esta situación, tenemos que Waveform en 13 de 20 casos, Cancer, Wine y Liver en 10 casos (50%) tiene los más altos niveles de precisión al mismo tiempo que también presentan menores valores de Q.

Tabla 6.8. Casos donde los valores **pequeños** de Q y la **mejor** precisión son encontrados, por técnica de integración de submuestras y método de fusión

		Votación ponderada dinámica					
	Votación simple	Por promedio	Distancia inversa	Según Shepard	Shepard modificado	Total	%
Cancer	1	2	3	2	2	**10**	**50**
Heart	0	1	1	1	1	**4**	**20**
Liver	1	3	4	1	1	**10**	**50**
Pima	2	2	2	0	0	**6**	**30**
Sonar	1	1	1	1	2	**6**	**30**
Glass	1	2	2	2	1	**8**	**40**
Iris	2	2	2	2	2	**10**	**50**
Vehicle	1	1	1	1	1	**5**	**25**
Wine	1	3	1	2	3	**10**	**50**
German	2	2	1	1	0	**6**	**30**
Satimage	0	1	1	1	1	**4**	**20**
Phoneme	0	0	0	0	2	**2**	**10**
Waveform	3	4	3	2	1	**13**	**65**
Segment	0	0	0	1	1	**2**	**10**
Total	**15**	**24**	**22**	**17**	**18**	**96**	**34.3**
%	**26.8**	**42.9**	**39.3**	**30.4**	**32.1**	**34.3**	

Finalmente, considerando los cinco métodos de votación, los métodos para seleccionar los patrones donde se observa más influencia de un alto grado de diversidad es al utilizar los métodos de Bagging en 31 casos (44.3), la selección aleatoria sin reemplazo en 22 casos (31.4%) y Boosting en 21 casos (30%).

6.2.1.2 Bajo Q y baja precisión

A diferencia de lo analizado en el punto anterior, ahora se buscan las situaciones en las que al utilizar un SMC con decisiones altamente diversas se obtienen bajos índices de precisión. La Tabla 6.9 muestra un resumen de esta situación.

Considerando cada uno de los cuatro métodos de selección de patrones aplicados sobre las 14 bases de datos y utilizando los cinco métodos de ponderación, tenemos un total de 70 estimaciones (por cada método de selección), de las cuales las técnicas de selección de patrones con las que en mayor porcentaje coincidieron los valores pequeños de Q y la peor precisión son la selección secuencial, Bagging y Arc-x4 con un 35.7% (25 casos cada uno), en tanto que con Boosting en tan sólo 22 casos (31.4%) se presentó esta situación.

En lo que respecta al método de ponderación donde se observa alguna relación entre los valores bajos de Q y la baja precisión es con Shepard modificado en 46 casos (46.4%), en tanto que el método en el que menos relación se observa, es con la ponderación por promedio (15 casos 26.8%).

Tabla 6.9. Casos donde los valores **pequeños** de Q y la **peor** precisión son encontrados, por técnica de integración de submuestras y método de fusión

	Votación simple	Votación ponderada dinámica					
		Por promedio	Distancia inversa	Según Shepard	Shepard modificado	Total	%
Cancer	1	1	1	0	2	**5**	**25**
Heart	1	1	1	0	2	**5**	**25**
Liver	2	1	1	2	2	**8**	**40**
Pima	1	0	1	3	4	**9**	**45**
Sonar	2	3	3	3	2	**13**	**65**
Glass	3	0	1	1	1	**6**	**30**
Iris	1	2	2	1	2	**8**	**40**
Vehicle	2	2	2	2	3	**11**	**55**
Wine	2	1	0	0	1	**4**	**20**
German	1	0	0	0	3	**4**	**20**
Satimage	2	0	2	2	2	**8**	**40**
Phoneme	2	1	1	2	0	**6**	**30**
Waveform	1	1	1	1	2	**6**	**30**
Segment	1	2	1	0	0	**4**	**20**
Total	**22**	**15**	**17**	**17**	**26**	**97**	**34.6**
%	**39.3**	**26.8**	**30.7**	**30.7**	**46.4**	**34.6**	

Por otro lado, las bases de datos donde se obtiene la menor precisión y el valor de Q es mínimo, son Sonar con 13 casos (65%), seguida de Vehicle con 11 casos (55%). En el resto de las bases de datos, esta situación se presenta en menos del 50% de los casos.

6.2.1.3 Alto Q y precisión elevada

Una última situación aquí estudiada corresponde a los casos en los que una baja diversidad (es decir, altos valores de Q) y una precisión elevada son encontradas (Tabla 6.10). Para esto, por cada uno de los cuatro métodos de selección de patrones tenemos un total de 70 estimaciones (es decir, 14 bases de datos por 5 métodos de votación). De estas estimaciones, las técnicas de selección de patrones que en mayor porcentaje siguen esta situación son Bagging y Arc-x4 con un 45% (32 casos), seguida de Boosting con 40% (28 casos) y la selección secuencial con 23 casos (32.9%).

A partir estos resultados, vemos que, en lo que respecta a métodos de ponderación, la votación simple (46.4%) es en la que se encontraron valores elevados de Q y el más alto índice de precisión, seguida inmediatamente por las ponderación con Shepard modificado y según Shepard.

Tabla 6.10. Casos donde los valores **grandes** de Q y la **mejor** precisión son encontrados, por técnica de integración de submuestras y método de fusión

	Votación simple	Votación ponderada dinámica					
		Por promedio	Distancia inversa	Según Shepard	Shepard modificado	Total	%
Cancer	1	0	0	0	1	**1**	**5**
Heart	3	2	2	2	1	**7**	**35**
Liver	3	1	2	3	2	**8**	**40**
Pima	2	2	2	3	2	**9**	**45**
Sonar	3	2	2	2	2	**8**	**40**
Glass	1	1	0	1	1	**3**	**15**
Iris	2	1	1	2	2	**6**	**30**
Vehicle	3	2	3	2	1	**8**	**40**
Wine	1	1	1	1	1	**4**	**20**
German	0	1	1	1	1	**4**	**20**
Satimage	1	1	0	1	3	**5**	**25**
Phoneme	2	3	3	3	2	**11**	**55**
Waveform	0	0	0	0	3	**3**	**15**
Segment	4	4	4	2	2	**12**	**60**
Total	**26**	**21**	**21**	**23**	**24**	**115**	**41.1**
%	**46.4**	**37.5**	**37.5**	**41.1**	**42.9**	**41.1**	

Por último, las bases de datos que, independientemente de la técnica de selección de patrones y del método de ponderación, reflejan alguna relación entre los valores grandes de Q y la mayor precisión son Segment con 12 casos (60%) y Phoneme con 11 casos (55%).

6.2.2 COEFICIENTE DE CORRELACIÓN

El coeficiente de correlación nos permite medir el grado de concordancia en las decisiones de los clasificadores, es decir, el grado de relación que las decisiones individuales tienen entre si. Los valores obtenidos oscilan entre -1 y 1, de manera que los valores más pequeños indican un mayor grado de diversidad. En la Tabla 6.11, se proporcionan de forma concentrada los resultados obtenidos al determinar el coeficiente de correlación.

Utilizando los parámetros de referencia proporcionados en la Sección 4.3.2 y analizando todos y cada uno de los resultados obtenidos con los diferentes métodos de submuestreo, tenemos el siguiente comportamiento del coeficiente de correlación:

a) Sin relación o baja: 47 casos (28.0%).
b) Correlación baja o moderada: 101 casos (68.1%).
c) Correlación alta o moderada: 20 casos (11.9%).

Tabla 6.11. Coeficiente de correlación

	±0.00 a .25	±0.26 a .50	±0.51 a .75	±0.76 a 1.00
Cancer	0	7	5	0
Heart	10	2	0	0
Liver	10	1	1	0
Pima	7	5	0	0
Sonar	11	1	0	0
Glass	1	11	0	0
Iris	0	9	3	0
Vehicle	1	11	0	0
Wine	0	12	0	0
German	7	5	0	0
Satimage	0	1	11	0
Phoneme	0	12	0	0
Waveform	0	12	0	0
Segment	0	12	0	0
Total	**47**	**101**	**20**	**0**
%	**28.0**	**68.1**	**11.9**	**0**

Claramente se puede apreciar la existencia de una correlación moderada baja entre las decisiones individuales, por lo que si analizamos los resultados del coeficiente de correlación con los obtenidos en la variabilidad (Sección 6.2.4), encontramos que en un 68% de los casos se cumple la condición de que valores más bajos de correlación indican mayor grado de diversidad.

En términos de precisión, aquellos casos donde existe una mayor correlación entre los clasificadores, en su mayoría, son también los casos donde se obtienen los valores más bajos de Q, deduciendo con ello un comportamiento similar.

6.2.3 MEDIDA DE DESACUERDO

Esta medida acumula las veces en que las decisiones de los clasificadores no coinciden, es decir, un clasificador D_i clasifica correctamente un patrón x y otro clasificador D_k lo clasifica erróneamente. La Tabla 6.12 muestra los resultados obtenidos al determinar esta medida con SMC formados por cuatro métodos de submuestreo: aleatorio sin reemplazo (A), Bagging (B), Boosting (C) y Arc-x4 (D).

Como puede verse, la menor diversidad en las decisiones ocurre con las bases de datos Cancer e Iris. También se puede apreciar que el grado de desacuerdo se incrementa conforme se utilizan un mayor número de clasificadores, predominando básicamente en 9 clasificadores. Por último, indicar que el desempeño de los clasificadores es similar al obtenido con la medida de variabilidad.

Tabla 6.12. Porcentaje de desacuerdo por técnica de selección de patrones y por número de clasificadores

	# Clasif	Medida de desacuerdo %					# Clasif	Medida de desacuerdo %			
		A	B	C	D			A	B	C	D
Cancer	**5**	0.04	0.00	0.08	0.07	**Vehicle**	**5**	0.31	0.33	0.34	0.36
	7	0.05	0.04	0.08	0.08		**7**	0.34	0.35	0.34	0.37
	9	0.04	0.04	0.09	0.10		**9**	0.34	0.35	0.37	0.39
Heart	**5**	0.36	0.37	0.41	0.45	**Wine**	**5**	0.26	0.29	0.26	0.32
	7	0.40	0.35	0.43	0.46		**7**	0.28	0.26	0.25	0.34
	9	0.41	0.39	0.38	0.46		**9**	0.26	0.28	0.28	0.32
Liver	**5**	0.39	0.38	0.43	0.44	**German**	**5**	0.32	0.32	0.27	0.37
	7	0.41	0.42	0.45	0.50		**7**	0.35	0.35	0.35	0.43
	9	0.42	0.43	0.47	0.48		**9**	0.36	0.36	0.38	0.39
Pima	**5**	0.28	0.32	0.34	0.39	**Satimage**	**5**	0.11	0.12	0.11	0.12
	7	0.13	0.32	0.36	0.41		**7**	0.12	0.12	0.12	0.11
	9	0.33	0.34	0.36	0.42		**9**	0.12	0.12	0.13	0.12
Sonar	**5**	0.39	0.31	0.38	0.42	**Phoneme**	**5**	0.19	0.19	0.23	0.25
	7	0.35	0.38	0.42	0.45		**7**	0.21	0.21	0.25	0.26
	9	0.40	0.39	0.45	0.46		**9**	0.22	0.22	0.26	0.28
Glass	**5**	0.26	0.29	0.26	0.29	**Waveform**	**5**	0.23	0.24	0.25	0.27
	7	0.29	0.31	0.27	0.40		**7**	0.25	0.25	0.27	0.28
	9	0.31	0.30	0.31	0.29		**9**	0.26	0.25	0.28	0.28
Iris	**5**	0.07	0.05	0.07	0.06	**Segment**	**5**	0.11	0.12	0.12	0.13
	7	0.08	0.06	0.05	0.07		**7**	0.13	0.14	0.15	0.15
	9	0.10	0.04	0.05	0.11		**9**	0.16	0.16	0.17	0.18

6.2.4 MEDIDA DE VARIABILIDAD

Los resultados obtenidos al aplicar la medida de diversidad descrita en la Sección 4.3.4 se proporcionan detalladamente en los anexos del Apéndice II. En estos resultados, podemos apreciar que cuantos más clasificadores se integran, el grado de variabilidad es mayor, siendo éste predominante (69.6%) en 9 clasificadores con 39 casos, seguido al utilizar 7 clasificadores (13 casos).

En lo concerniente a la influencia que tiene el grado de variabilidad sobre las técnicas para seleccionar los patrones respecto a la precisión, vemos que tanto la poca como la mucha variabilidad afectan de manera similar a los cuatro métodos, sobresaliendo levemente Bagging al obtener mayor precisión con mayor variabilidad y Boosting para una mayor precisión con menor variabilidad.

Las bases de datos con las que se obtiene una variabilidad inferior al 0.5%, independientemente del número de clasificadores utilizados, son Cancer, Iris, Satimage y Segment. Los resultados mostrados a continuación nos indicarán lo favorable o desfavorable que esta situación resulta ser al clasificar nuevos patrones.

6.2.4.1 Mayor variabilidad y precisión elevada

La Tabla 6.13 muestra las veces donde se encuentra la mayor variabilidad y, al mismo tiempo, los mayores índices de precisión. La tabla incluye para cada base de datos las cuatro técnicas para manipular los patrones y los cinco métodos para ponderar las decisiones individuales.

Tabla 6.13. Casos donde coinciden la **mayor** variabilidad en las decisiones y la **mejor** precisión por técnica de submuestreo y método de fusión

	Votación simple	Votación ponderada dinámica				Total	%
		Por promedio	Distancia inversa	Según Shepard	Shepard modificado		
Cancer	3	3	3	3	2	**14**	**70**
Heart	0	0	0	0	0	**0**	**0**
Liver	2	4	4	1	2	**13**	**65**
Pima	3	3	3	0	0	**9**	**45**
Sonar	0	2	2	1	1	**6**	**30**
Glass	1	2	3	3	2	**11**	**55**
Iris	2	2	2	2	2	**10**	**50**
Vehicle	1	1	1	2	2	**7**	**35**
Wine	1	1	1	0	1	**4**	**20**
German	2	1	1	1	0	**5**	**25**
Satimage	1	1	1	0	1	**4**	**20**
Phoneme	1	0	0	0	1	**2**	**10**
Waveform	4	4	4	3	0	**15**	**75**
Segment	0	0	0	1	1	**2**	**10**
Total	**21**	**24**	**25**	**17**	**15**		
%	**37.5**	**42.9**	**44.6**	**30.4**	**26.8**		

En lo que respecta a la influencia del grado de variabilidad en los índices de precisión, observamos que, en algunas bases de datos, esta influencia es muy poca (Phoneme) o absolutamente nula (Heart) y es, tan sólo en 5 casos (37.7%), donde esta influencia supera o iguala al 50% (Cancer, Liver, Glass, Iris y Waveform). Por otro lado, el método de ponderación que resulta más beneficiado por la alta variabilidad en la decisión final es el que utiliza la distancia inversa.

6.2.4.2 Menor variabilidad y precisión elevada

Contrario a lo revisado en el punto anterior, los resultados de la Tabla 6.14 permiten encontrar situaciones en las que se tiene poca diversidad en los componentes del SMC y, al mismo tiempo, se obtienen un alto rendimiento en la clasificación. Con esto, se pretende verificar si un SMC con poca variabilidad en las decisiones individuales resulta más preciso a causa de la alta coincidencia en las decisiones por parte de los clasificadores.

Tabla 6.14. Casos donde se obtiene **menor** variabilidad en las decisiones y la **mejor** precisión por técnica de submuestreo y método de fusión

		Votación ponderada dinámica					
	Votación simple	Por promedio	Distancia inversa	Según Shepard	Shepard modificado	Total	%
Cancer	2	0	0	0	1	**3**	**15**
Heart	2	2	2	2	3	**11**	**55**
Liver	2	1	2	3	2	**10**	**50**
Pima	1	1	1	4	3	**10**	**50**
Sonar	3	1	1	3	1	**9**	**45**
Glass	1	1	0	1	1	**4**	**20**
Iris	0	0	0	1	1	**2**	**10**
Vehicle	2	1	2	2	1	**8**	**40**
Wine	1	1	1	2	2	**7**	**35**
German	0	1	1	0	0	**2**	**10**
Satimage	2	2	1	2	2	**9**	**45**
Phoneme	2	3	3	3	2	**13**	**65**
Waveform	0	0	0	0	3	**3**	**15**
Segment	4	4	3	2	2	**15**	**75**
Total	**22**	**18**	**17**	**25**	**24**		
%	**39.3**	**32.1**	**30.7**	**44.6**	**42.9**		

Al revisar los métodos de ponderación, podemos ver que los métodos que utilizan la ponderación según Shepard (44.6%) y Shepard modificado (42.9%) son los que más influenciados se ven al obtener los mejores índices de precisión con poca variabilidad en las decisiones individuales, situación que resulta ser contraria a los resultados mostrados en la Tabla 6.13, donde los métodos que tuvieron mayor precisión con mayor diversidad ahora son los que con menor variabilidad tienen menor precisión.

De lo antes mencionado, podemos deducir que el grado variabilidad puede tener efectos favorables y/o desfavorables al utilizar algunos métodos de ponderación, pues en el caso de la votación simple, en la que todos los clasificadores tienen un mismo peso (1), el desempeño del SMC no se ve afectado.

Por otro lado, las bases de datos que más favorecidas se ven al contar con bajos índices de variabilidad son Segment en 15 casos (75%), Phoneme en 13 casos (65%), Heart en 11 casos (55%), y Liver y Pima en 10 casos (50%). Si comparamos estos resultados con los mostrados en la Tabla 6.11, vemos que tanto Segment como Phoneme y Heart son bases de datos que, en muy pocas ocasiones, tuvieron la mejor precisión con el mayor grado de variabilidad, en tanto que Liver tuvo unos de los mejores resultados con la mayor variabilidad y, ahora con menor variabilidad, también tiene un comportamiento similar. Además, si recordamos las bases de datos que, en general, tienen menor variabilidad en sus decisiones son Cancer, Iris, Satimage y Segment, de las cuales sólo ésta última demuestra tener una influencia negativa de la poca variabilidad, lo que nos indica que el grado de

variabilidad tiene poca o nula influencia sobre los resultados en la clasificación, siendo ésta más dependiente del método utilizado para combinar las decisiones individuales de los clasificadores.

6.3 CONCLUSIONES

En el presente capítulo, se abordó la diversidad en un SMC desde dos perspectivas: diversidad en la construcción de las submuestras que forman el SMC y análisis de diversidad en las decisiones individuales.

Para conseguir diversidad en un SMC, un primer estudio consistió en crear submuestras con algoritmos como Bagging, Boosting y Arc-x4. Estos métodos, en su implementación clásica, realizan la selección de patrones de forma indiscriminada entre clases, con igual tamaño que el CE original. Esta situación propicia la obtención de submuestras con características poco deseables: desbalance entre clases, gran tamaño y elevado coste computacional. Como estrategia de solución a estos problemas, se propuso una modificación a la forma en que se realiza la selección de patrones consisten en considerar la distribución existente entre las clases del CE original, es decir, formar submuestras que tengan el mismo grado de representatividad de patrones entre clases como el existente en el CE original. De este modo, la complejidad computacional asociada al procesamiento del SMC disminuye de forma considerable y se elimina la posibilidad de (cuando el CE original no cuenta con desbalance) trabajar con submuestras desbalanceadas, o que el desbalance existente en el CE original no aumente.

Los resultados obtenidos permitieron demostrar que un SMC con submuestras formadas con la selección por clase incrementa notablemente la precisión general del SMC y se requiere menos recursos que cuando se construyen submuestras de la forma clásica. Esto se debe, principalmente, a que el tamaño de una submuestra formada con la selección clásica tiene H veces el tamaño de una submuestras creada con el método modificado (considerando la distribución por clase, con tamaño m/H), manteniendo esta última el mismo costo computacional que cuando se utiliza un único clasificador.

Un segundo estudio contempló la búsqueda del número de componentes que un SMC debe tener para que sea lo suficientemente diverso y tenga un desempeño aceptable. Para esto, se construyeron SMC con diferente cantidad de clasificadores: 3, 5, 7, 9, 15 y 25. Basándonos en los resultados obtenidos en la Sección 6.1.1, las submuestras se formaron considerando la distribución entre clases. Para tomar la decisión final, se utilizaron dos diferentes esquemas de fusión: votación por mayoría simple y votación por mayoría ponderada por promedio de distancias.

Los resultados experimentales demostraron que el mejor rendimiento del SMC se tiene cuando se utilizan 5, 7 ó 9 clasificadores. Por otro lado, en lo que respecta al método de submuestreo, vimos que la selección sin reemplazo y Bagging son los que tienen un comportamiento más uniforme, principalmente cuando se utiliza el esquema por fusión ponderada. Por el contrario, los métodos que incluyen el principio de Arcing (Boosting y Arc-x4) tienen poca factibilidad cuando se forman SMC con grandes cantidades de elementos, particularmente 15 y 25. De la misma manera, esta situación se presentó cuando se trabaja con bases de datos que cuenta con clases poco densas.

Por último, en lo que concierne al análisis de la diversidad en las decisiones, pudo verse que el grado de diversidad no forzosamente está relacionado con elevados índices de precisión, ya que en algunos casos que presentan poca diversidad, la precisión en la clasificación es similar a bases de datos donde los índices de diversidad son superiores a 0.5. Pudo observarse también que el grado de diversidad y de desacuerdo aumentan conforme se incrementa el número de clasificadores que forman el SMC. En lo que respecta al método que más cumple con la condición de que mayor diversidad implica mayor precisión, resultó ser Bagging, en tanto que el método que tiene mayor precisión con menos variabilidad fue Boosting.

Capítulo 7

Análisis de eficiencia del clasificador 1-NN con SMC

Tal como se explicó en el Capítulo 3, existen múltiples factores relacionados con el CE que deterioran el desempeño del clasificador. En el presente capítulo, se desarrollan metodologías con las que se pretende eliminar o disminuir los efectos negativos que algunos de estos factores tienen sobre los índices de precisión. En particular, se tratarán los relacionados con el desbalance, la redundancia y el tamaño de los conjuntos de datos. Para este fin, se hace uso de algoritmos de preprocesado y también de métodos relacionados con los SMC.

7.1 TRATAMIENTO DEL DESBALANCE EN BASES DE DATOS DE DOS CLASES

Se dice que una base de datos cuenta con desbalance cuando alguna(s) de sus clases está claramente menos representada respecto a la cantidad de patrones contenidos en el resto de las clases. Las alternativas aquí propuestas para tratar conjuntos de datos que presentan este problema contemplan tres variantes: búsqueda de un balance perfecto entre clases, aumento del tamaño de la clase minoritaria y beneficio de la clase minoritaria sin alterar la distribución entre clases. La primera de estas propuestas incorpora métodos propios de los SMC y algunos algoritmos de filtrado y reducción del conjunto de datos. La segunda realiza el balance del conjunto de datos con la incorporación de patrones a la clase menos representada. La última hace uso de métodos de ponderación de distancia, tanto al realizar el filtrado del CE como en la clasificación de nuevos casos.

Los resultados presentados en este apartado se realizaron con cuatro bases de datos, todas ellas formadas por dos clases (ver Sección 5.2), y con diferentes grados de desbalance. La cantidad de patrones que representan a cada una de las clases puede verse en la Tabla 7.1.

Tabla 7.1. Bases de datos desbalanceadas de dos clases

	Conjunto de Entrenamiento			Conjunto de Prueba		
	Clase 0	Clase 1	Total	Clase 0	Clase 1	Total
Glass	24	150	174	5	35	40
Phoneme	1268	3054	4322	318	764	1082
SatImage	500	4647	5147	126	1162	1288
Vehicle	170	508	678	42	126	168
Ism	234	9831	10065	26	1092	1118

Cabe también indicar que, cuando se trabaja con bases de datos de este tipo, utilizar la precisión general como criterio de evaluación es poco útil, debido a que esta medida no considera de forma separada las precisiones de cada una de las clases. Por tanto, para evaluar los resultados aquí reportados, se hace uso de la media geométrica $g = (a+ * a-)^{1/2}$, donde a+ es la precisión de la clase minoritaria y a- la correspondiente a la clase mayoritaria.

7.1.1 UTILIZACIÓN DE SMC PARA COMBATIR EL DESBALANCE

Para tratar este tipo de bases de datos, se proponen cuatro metodologías (ver Sección 5.4.2). Estas metodologías incluyen la combinación de un SMC con algoritmos de preprocesado. Los algoritmos de preprocesado seleccionados para este fin son la edición de Wilson y el Subconjunto Selectivo Modificado. Es

importante aclarar que el objetivo de estos algoritmos, en principio, no es lograr una distribución balanceada entre clases (debido a que no tienen control sobre la cantidad de patrones eliminados o seleccionados, e incluso, en ocasiones, pueden degenerar en situaciones poco deseadas, tales como que las clases mayoritarias se conviertan en minoritarias), sino elevar la calidad del conjunto de datos y la eficiencia del clasificador. Para eliminar el desbalance existente en el CE, las submuestras que integran el SMC se formaron considerando el número de patrones de la clase minoritaria, de tal manera que las submuestras resultantes contendrán la misma cantidad de patrones en las dos clases.

El número de submuestras a formar está directamente relacionado con el desbalance existente en el CE, es decir, se generarán tantas submuestras como veces sea menor la clase minoritaria con respecto a la clase mayoritaria. Por ejemplo, el CE de la base de datos Phoneme tiene dos clases, una de ellas con 1268 patrones y la segunda con 3054. Como puede verse, la clase mayoritaria supera 2.41 veces la cantidad de patrones incluidos en la clase minoritaria, lo que nos indica que se deben integrar dos submuestras. Sin embargo, debido a que la posibilidad de empates aumenta cuando se utiliza un número par de clasificadores, se opta por integrar al siguiente número impar y, por tanto, en este caso, el SMC constará de 3 submuestras.

La Tabla 7.2 muestra los mejores resultados obtenidos al emplear cada una de las diferentes metodologías, utilizando como esquema de fusión, la votación simple. Para una consulta más detallada, pueden revisarse los anexos del Apéndice III. En esta misma tabla, las filas tituladas “Técnica” indican la técnica de selección de patrones con la que se obtuvo cada uno de los resultados: selección secuencial sobre la clase mayoritaria (A), selección aleatoria sin reemplazo en la clase mayoritaria (B), Bagging en la clase mayoritaria (C), Bagging en las dos clases por separado (D), y Boosting en la clase mayoritaria (E). Por otro lado, los números que se encuentran bajo el nombre de cada base de datos indican la cantidad de clasificadores individuales utilizados por cada una de ellas. Los valores en la fila DE corresponden a las desviaciones estándar de la media geométrica (g).

A fin de valorar el desempeño del SMC, se incluyen los resultados de la clasificación tradicional (con un sólo clasificador) con los algoritmos de preprocesado. En estos resultados, vemos que la media geométrica obtenida al utilizar alguna de las metodologías propuestas supera, en todos los casos, los mejores resultados obtenidos con un único clasificador.

En lo que respecta a las técnicas de selección de patrones, no se observa un comportamiento uniforme, pues aquella técnica que mejora la precisión para algunas base de datos, en otras no es tan favorable. Pese a ello, las técnicas B (aleatoria sin reemplazo) y C (aleatoria con reemplazo) levemente tienen mejor desempeño. Por último, en lo que a las metodologías se refiere, vemos que la Metodología 2 (SMC+Wilson) resulta ser la mejor alternativa de solución para

tratar este tipo de bases de datos desbalanceadas, aunque las diferencias entre ellas no parecen realmente significativas.

Tabla 7.2. Clasificación con SMC sobre bases de datos desbalanceadas de dos clases

						METODOLOGÍAS			
		Original	Edición	SSM	Ed+SSM	1	2	3	4
Glass 7	Técnica					B	D	C	E
	g	**86.7**	84.6	86.4	84.9	86.6	87.4	87.6	**88.2**
	DE	**12.2**	16.8	11.8	16.1	10.1	9.9	9.0	**8.9**
Vehicle 3	Técnica					C	C	B	E
	g	56.5	47.6	**59.7**	50.1	68.0	**68.4**	66.1	67.3
	DE	4.2	5.2	**1.5**	8.2	3.6	**2.4**	3.8	1.5
Phoneme 3	Técnica					B	B	C	D
	g	73.8	**73.8**	72.2	72.4	74.3	**75.1**	73.0	73.1
	DE	6.0	**5.6**	6.3	6.2	8.2	**8.2**	7.8	6.8
Satimage 11	Técnica					C	B	C	C
	g	**70.9**	66.4	70.1	65.6	**79.5**	78.3	78.8	77.6
	DE	**15.5**	22.9	15.9	24.3	**4.5**	4.7	3.3	4.4

7.1.2 TRATAMIENTO DEL DESBALANCE CON SOBRE-ENTRENAMIENTO

La segunda propuesta para combatir los efectos negativos del desbalance utiliza el método propuesto por Chawla et al [Cha., 00]. Este método, llamado SMOTE (Syntetic Minority Over-sampling Technique), incrementa el tamaño de la clase minoritaria mediante la generación de *patrones sintéticos*. En su implementación, hace uso de la regla *k*-NN y establece el porcentaje de incremento deseado (*inc*). La Figura 7.1 muestra el proceso general que se sigue para incorporar patrones sintéticos a la clase minoritaria mediante este método.

Para los experimentos del presente apartado, el valor de *inc* estará en función del porcentaje de desbalance existente entre las clases [Bar., 05b]. Por ejemplo, en el CE de la base de datos Vehicle, la clase mayoritaria contiene aproximadamente 3 veces el número de patrones de la clase minoritaria, de manera que, el valor de *inc* para este CE sería 300%. Esto se traduce en que, si el parámetro $k = 5$ y el incremento deseado es 300%, entonces, de esos 5 vecinos se eligen aleatoriamente sólo 3, con los que se generará un patrón sintético por cada uno de ellos.

Como es de suponer, al aumentar el tamaño de la clase minoritaria, aumenta también el tamaño de los CE resultantes. Para disminuir el efecto negativo que un CE de gran tamaño tiene sobre la regla 1-NN, se acude a algoritmos de limpieza y reducción, buscando afectar lo menos posible los índices de precisión. Para ello, se

implementan cuatro diferentes metodologías que utilizan algoritmos de preprocesado:

Metodología 1 (M1): CE original → SMOTE → 1-NN
Metodología 2 (M2): CE original → SMOTE → Wilson → 1-NN
Metodología 3 (M3): CE original → SMOTE → SSM → 1-NN
Metodología 4 (M4): CE original → SMOTE → Wilson → SSM → 1-NN

Entradas:
M = Conjunto de *m* patrones de la clase minoritaria:{M_i | i = 1,2,..*m*}.
Inicio
Seleccionar de forma aleatoria un patrón *x* de entrenamiento
Obtener los *k* vecinos de *x* en la misma clase.
Desde s = 0,…, *inc*
v = un *k*-vecino seleccionado aleatoriamente
Para todo i //Patrones de clase minoritaria
Obtener Rn // Número aleatorio entre 0 y 1
// Creación del patrón sintético
$p = (M_{i,j} + Rn) * (v_j - M_{i,j})$ $\forall$ j //Para cada atributo
$M = M \cup p$ //Incluye el patrón sintético
Fin para
Fin desde
Fin de algoritmo

Figura 7.1. Algoritmo SMOTE

Al igual que en los otros experimentos, en la Tabla 7.3, se incluyen los resultados obtenidos con el CE original. Los valores remarcados en negrita corresponden a los mejores resultados obtenidos para cada una de las bases de datos, y los encerrados entre paréntesis corresponden a la desviación estándar.

Tabla 7.3. Precisión al clasificar bases de datos desbalanceadas tratadas con sobre-entrenamiento

	Original	M1	M2	M3	M4
Glass	86.7(12.2)	**88.7(10.9)**	86.4(9.6)	88.2(9.9)	85.9(12.2)
Vehicle	56.5(4.2)	59.7(5.8)	**64.5(5.8)**	57.1(2.7)	62.7(4.5)
Phoneme	73.8(6.0)	73.6(6.3)	**74.9(6.8)**	70.3(9.8)	74.8(7.9)
Satimage	70.9(15.5)	77.1(6.4)	**78.5(4.2)**	74.1(9.1)	76.2(7.2)
Ism	60.2(6.0)	83.3(3.1)	**86.8(3.5)**	80.8(2.8)	86.0(3.4)

En esta tabla, podemos ver que, para Vehicle, Satimage e Ism, la precisión que cualquiera de las cuatro metodologías obtiene es superior a la que se logra con el CE sin balancear. Por otro lado, la metodología con la que se tienen los mejores

resultados en cuatro de los cinco bases de datos es la que utiliza la edición de Wilson después de aplicar la técnica SMOTE (M2).

Como ya se había mencionado, la utilización del SMOTE tiene asociado un aumento en los requerimientos de almacenamiento y de cálculo. Para analizar este aumento, la Tabla 7.4 muestra los tamaños resultantes con cada uno de los CE después de aplicar las diferentes metodologías propuestas.

Tabla 7.4. Tamaño de CE resultantes

	Original	M1	M2	M3	M4
Glass	174	294.0	285.0	27.6	18.6
Vehicle	679	848.2	385.8	321.2	176.2
Phoneme	4322	5590.0	5185.2	1201.2	756.0
Satimage	5147	9147.0	8706.0	1322.8	906.4
Ism	10061	19571.6	17735.3	6932.2	1599.5

De estos tamaños, tenemos que las metodologías 1 y 2 tienen un considerable incremento en su tamaño respecto a los CE originales, siendo en la mayoría de los casos el doble de su tamaño inicial, en tanto que las metodologías 3 y 4, por lo general, disminuyen hasta en un 80%. Es importante notar que los tamaños más pequeños se encuentran con la metodología 4 (combina tanto la limpieza como la reducción) y, de acuerdo a los resultados de clasificación (Tabla 7.3), vemos que (con poca diferencia) es también la metodología que tiene los segundos mejores resultados sobre Vehicle, Phoneme e Ism. Sin embargo, si consideramos el tamaño con el que cuentan los CE utilizados con la metodología 2 y el tamaño con el que cuenta la metodología 4, vemos que hay una diferencia verdaderamente significativa, situación por la que el ahorro en el costo computacional y de almacenamiento justifica ampliamente la recomendación de esta metodología (SMOTE + Wilson + SSM) como alternativa de solución para CE donde se decida incrementar el tamaño de la clase minoritaria para eliminar el desbalance entre clases.

Por otro lado, con la intención de ver con mayor detalle el comportamiento del algoritmo de edición sobre CE que incorporan patrones sintéticos, a continuación se hace un análisis de la naturaleza de los patrones que fueron eliminados en cada uno de los CE. Para ello, la Tabla 7.5 muestra la procedencia de los patrones eliminados al aplicar el algoritmo de limpieza, en el CE original y en el CE con patrones sintéticos incorporados mediante SMOTE. Las primeras dos columnas contienen los patrones que fueron eliminados al aplicar el algoritmo de edición sobre las dos clases de los CE originales, las siguientes dos columnas (3 y 4) contienen los patrones que al realizar la edición son eliminados tanto de la CE original como en el CE con patrones sintéticos. Las columnas tituladas “Adicionales” (5 y 6) nos indican la cantidad de patrones que adicionalmente a los contemplados en las columnas 3 y 4 fueron eliminados en el CE balanceado y que, existiendo en el CE original, no se eliminaron. Finalmente, se proporcionan los

patrones que se eliminaron en el CE original y que por el contrario, no se eliminaron en el CE con patrones sintéticos.

Tabla 7.5. Promedio del número de patrones eliminados con la edición de Wilson

	CE ORIGINAL + WILSON		ORIGINAL + SMOTE + WILSON					
			IGUAL ELIMINADOS		ADICIONALES		SI(Original) No(SMOTE)	
	Clase 1 (1)	Clase 2 (2)	Clase 1 (3)	Clase 2 (4)	Clase 1 (5)	Clase 2 (6)	Clase 1 (7)	Clase 2 (8)
Glass	5.6	2.8	0.8	2.8	3.0	2.4	5.2	0
Vehicle	105.6	64.0	33.2	64.0	9.6	70.4	72.4	0
Phoneme	256.8	171.8	98.4	171.8	56.0	78.6	158.4	0
Satimage	125.2	127.0	0	127.0	3.8	310.2	125.2	0
Ism	146.1	28.7	25.8	28.7	1009.3	778.4	120.3	0

De estos datos, obtenemos que, en los CE que incorporan patrones sintéticos, se detectan más patrones atípicos de la clase mayoritaria que los encontrados en el CE original. Por otro lado, existen muchos patrones atípicos de la clase minoritaria que no se logran detectar en el CE original y que, por el contrario, sí son detectados en el CE con patrones sintéticos. Esto último nos podría indicar que muchos de los patrones sintéticos incorporados durante el proceso de balance fueron formados a partir de algunos patrones atípicos ya existentes.

Para ver más claramente el comportamiento que la edición de Wilson tiene sobre los CE que incorporan patrones sintéticos, la Figura 7.2 ilustra los porcentajes de patrones eliminados de cada una de las clases. Es importante aclarar que los datos de la Tabla 7.5 incluyen solamente aquellos patrones que, existiendo en el CE original, fueron (o no) detectados en el CE balanceado. Los patrones sintéticos no son considerados en ningún momento, en tanto que los resultados mostrados en la Figura 7.2 incluyen todos los patrones atípicos del CE balanceado con patrones sintéticos.

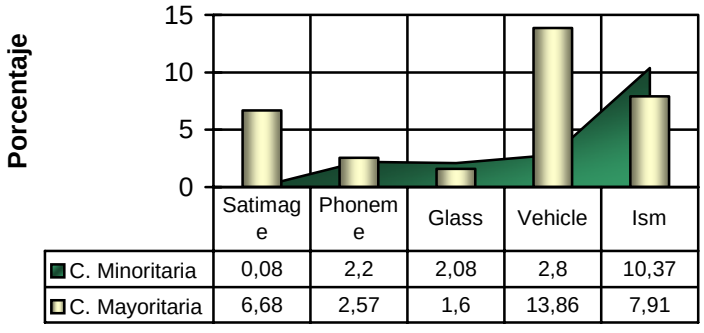


Figura 7.2. Porcentaje de patrones eliminados con la edición de Wilson

Como puede verse, en Satimage y Vehicle, se eliminan más patrones de la clase mayoritaria que de la clase minoritaria. Un caso particular sucede con Ism, donde la eliminación de patrones es mayor en la clase minoritaria que en la mayoritaria. Tal como se vio en la Tabla 7.5, desde su origen, el CE cuenta con una gran cantidad de patrones atípicos, los cuales pudieron participar en la generación de los patrones sintéticos, elevando con esto la obtención de un CE con mayor cantidad de patrones sintéticos atípicos.

7.1.3 FAVORECIMIENTO DE LA CLASE MINORITARIA

En esta última propuesta, se trata de balancear las bases de datos favoreciendo a la clase minoritaria de dos diferentes formas: mediante la ejecución de los algoritmos de limpieza únicamente en la clase mayoritaria y por medio de la implementación de una función de distancia ponderada:

a) Preprocesado en clase mayoritaria. Para estos experimentos, los algoritmos de preprocesado son aplicados únicamente sobre la clase mayoritaria, mientras que la clase minoritaria queda con los pocos patrones que originalmente ya tenia. Para esto, se distinguen dos métodos: el primero de ellos inicia con la edición de la clase mayoritaria y, posteriormente, se realiza la clasificación. El segundo ejecuta primeramente el algoritmo de limpieza en la clase mayoritaria y, posteriormente, balancea con la incorporación de patrones sintéticos mediante el algoritmo SMOTE.

b) Ponderación de distancia. Este método utiliza la distancia Euclídea ponderada, tanto en la edición de Wilson como en la clasificación de nuevos casos. El objetivo de esta ponderación es compensar el desbalance del conjunto de datos sin alterar la distribución real de clases. La ponderación se asigna a las respectivas clases y no a los patrones contenidos en cada una de ellas, de tal forma que el peso asignado a la clase mayoritaria sea mayor que el asignado a la minoritaria. Para la ponderación, se utiliza la siguiente formula [Bar., 03b]:

$$d_W(X,Y) = \left(\frac{m_i}{m}\right)^{1/n} d_E(X,Y)$$

donde x es un patrón de entrenamiento perteneciente a la clase *i*, y es un patrón a clasificar, m_i es el número de patrones que contiene la clase *i, m* es el tamaño total del CE, n es la de dimensionalidad del espacio, y d_E es la distancia Euclídea.

En la Tabla 7.6, se plasman los resultados correspondientes a la media geométrica en la clasificación cuando se utilizan estas propuestas con sus dos

variantes. Los valores resaltados en negrita indican la mejor precisión obtenida en cada una de las bases de datos.

Tabla 7.6. Clasificación de bases de datos desbalanceadas donde se favorece a la clase minoritaria

	CE original		PREPROCESADO EN CLASE MAYORITARIA				
			Clasificación con distancia Euclídea		Clasificación con distancia ponderada		Clasificación con distancia Euclídea
	Clasificación Euclídea	Clasificación ponderada	Wilson c/Euclídea	Wilson c/ponderada	Wilson c/Euclídea	Wilson c/ponderada	Wilson +SMOTE
Glass	86.7	**88.2**	86.2	87.9	87.9	86.2	**88.2**
Vehicle	55.8	59.6	64.0	**67.2**	65.8	65.6	66.4
Phoneme	73.8	**76.0**	74.9	75.3	75.7	75.0	74.2
Satimage	70.9	75.9	73.0	77.8	76.2	74.5	**77.9**
ISM	60.2	75.8	63.9	**78.7**	76.2	70.0	54.2

En esta tabla, es posible apreciar la influencia que la distancia ponderada tiene sobre los índices de precisión. Al utilizar los CE sin preprocesado, en todos los casos, la clasificación con distancia ponderada supera a la clasificación con la distancia Euclídea simple.

Al aplicar el preprocesado sólo sobre la clase mayoritaria, se distinguen los siguientes puntos:

a) Cuando tanto la edición como la clasificación se realizan con distancia Euclídea.

b) Cuando la edición se realiza con distancia Euclídea y la clasificación con distancia ponderada.

c) Cuando la edición se realiza con distancia ponderada y la clasificación con distancia Euclídea.

d) Cuando tanto la edición como la clasificación se realizan con distancia ponderada.

e) Cuando se clasifica con distancia Euclídea después de balancear el CE.

Al revisar los resultados de los casos donde se utiliza la edición con distancia Euclídea, vemos que al realizar la clasificación con distancia ponderada se mejora claramente el comportamiento del clasificador. Por el contrario, cuando se realiza la edición con distancia ponderada, los mejores resultados se obtienen al clasificar con la distancia Euclídea. Por otro lado, cuando se busca el balance del CE posterior a su edición, vemos que las mejoras son prácticamente nulas, siendo poco significativas en la base de datos Vehicle.

A modo de conclusión, tenemos que, en los resultados de todas las combinaciones existentes, no se observa un comportamiento uniforme, ya que los mejores resultados se encuentran divididos entre la clasificación con distancia ponderada del CE sin preprocesar (Glass y Phoneme), la clasificación con distancia Euclídea al realizar la edición con distancia ponderada (Vehicle e Ism), y cuando se incorporan patrones sintéticos después de editar el CE (Glass y Satimage). Sin embargo, cuando comparamos las mejoras que se obtienen con esta última propuesta (Wilson+SMOTE) y el tamaño del CE con el que se trabaja, vemos que la cantidad de patrones con los que cuentan estos CE es casi el doble respecto a cualquiera de las demás propuestas utilizadas (ver Tabla 7.7), lo que hace poco recomendable su uso.

Tabla 7.7. Cantidad de patrones resultantes en los CE

	Original	Wilson con distancia ponderada	Wilson con distancia Euclídea	Wilson +SMOTE
Glass	174.0	168.6	171.2	291.2
Vehicle	678.0	512.0	584.8	784.2
Phoneme	4322.0	3997.8	4150.2	5418.2
Satimage	5147.00	4820.6	4971.6	9020.0
Ism	10062.0	9818.5	10032.9	19604.5

7.2 TRATAMIENTO DEL DESBALANCE EN BASES DE DATOS DE MÁS DE DOS CLASES

El tratamiento de conjuntos de datos que presentan desbalance y cuentan con más de dos clases no es una tarea fácil. Para su adecuado tratamiento, es necesario tener en cuenta los siguientes problemas: existe más de una clase minoritaria y más de una que es mayoritaria, y estas mayoritarias a su vez son minoritarias respecto a otras clases. Como ejemplo de esto, en el gráfico de la Figura 7.3, podemos ver la distribución de patrones entre clases de la base de datos Feltwell y, en el gráfico de la Figura 7.4, la distribución en la base de datos Cayo.

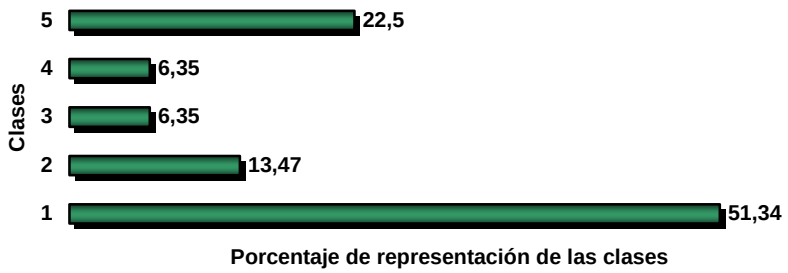


Figura 7.3. Distribución de los patrones en la base de datos Fetlwell

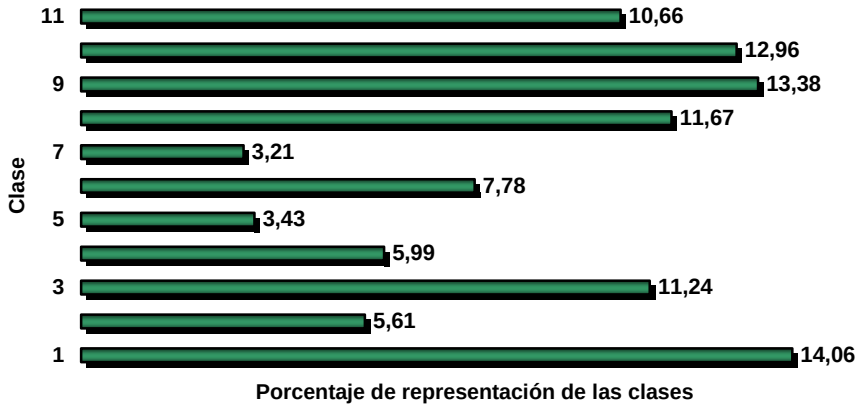


Figura 7.4. Distribución de los patrones en la base de datos Cayo

Al integrar las submuestras, aquellas clases que contienen un número de patrones por encima del umbral *u* se consideraron clases mayoritarias, en caso contrario clases minoritarias. Sólo las clases consideradas mayoritarias fueron ajustadas al tamaño establecido por el umbral, en tanto que las minoritarias conservaron su tamaño original.

La Tabla 7.8 muestra los resultados de la votación simple, obtenidos con la selección de patrones de forma aleatoria sin reemplazo y Bagging, utilizando las dos diferentes formas de calcular el umbral. Como se mencionó en la Sección 5.2, con Feltwell se utilizan dos conjuntos de prueba o test. Como punto de referencia, también se incluyen los resultados obtenidos con la utilización de un clasificador único.

Tabla 7.8. Clasificación utilizando bases de datos de más de dos clases

		CE original	Max-Min		Promedio	
			Bagging	SinReemplazo	Bagging	SinReemplazo
Cayo	PG	76.0	76.3	76.3	79.0	79.1
	CK	0.729	0.733	0.734	0.764	0.765
	VK	0.000077	0.000076	0.000076	0.000070	0.000069
Feltwell Conjunto De test 1	PG	72.8	73.9	73.4	73.9	73.5
	CK	0.638	0.653	0.646	0.654	0.648
	VK	0.000051	0.000050	0.000051	0.000049	0.000050
Feltwell Conjunto De test 2	PG	87.7	90.1	90.1	90.2	89.8
	CK	0.813	0.847	0.846	84.830	0.841
	VK	0.000074	0.000070	0.000071	0.000069	0.000072

Debido a que estas bases de datos cuentan con una variación importante en la representatividad de los patrones en cada una de las clases, se utilizaron como medidas de evaluación la precisión general (PG), el coeficiente Kappa (CK) y la varianza del coeficiente Kappa (VC).

En estos resultados, vemos que la técnica que considera el valor promedio para el establecimiento del umbral es la que obtiene mayores índices de precisión, principalmente con el método de selección de patrones Bagging. También podemos notar que con ambos esquemas se logra superar la precisión obtenida mediante la clasificación tradicional.

7.3 ESCALABILIDAD DE ALGORITMOS CON SMC

El tratamiento de CE con tamaño excesivo es un problema que ha sido abordado de varias formas (ver Sección 3.5.4), pero ninguna de ellas considera la utilización de SMC. La propuesta aquí hecha contempla la simulación de situaciones donde es imposible tratar el CE en forma conjunta. Para esto, se establecen cuatro metodologías (ver Sección 5.4.2) que utilizan un SMC de manera conjunta con algoritmos de limpieza y de reducción, situación no estudiada anteriormente. Además de los métodos de Bagging, Boosting y selección aleatoria sin reemplazo (sinRA1), para estos experimentos, se utiliza también la selección secuencial de patrones (sinR1). Este tipo de submuestreo puede ejemplificarse de la siguiente forma: consideremos un CE con dos clases, y se desea construir un SMC con tres clasificadores. Los patrones de cada una de las clases se dividen en 3 partes que son seleccionados de forma secuencial, es decir, en la primer submuestra se colocan el primer 33% de los patrones de cada una de las clases, en la segunda submuestra se coloca el siguiente 33% y, en la ultima, el resto de los patrones (Figura 7.5).

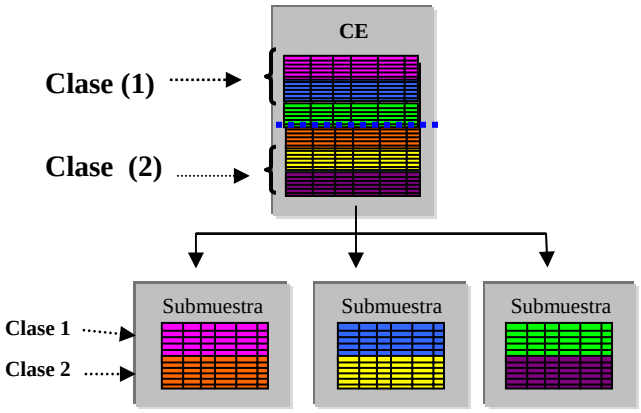


Figura 7.5. Selección de patrones secuencial en una base de datos de dos clases

En la Tabla 7.9, se muestran, de manera resumida, los resultados obtenidos al realizar la escalabilidad de algoritmos con esta propuesta. Para más detalles, puede consultarse las tablas de resultados del Apéndice III. Estos resultados corresponden a la precisión general (PG) cuando se emplea la votación simple en un SMC que consta de tres elementos. En esta tabla, también se incluyen los resultados que se obtienen cuando se utiliza un clasificador único. La fila de “Técnica” indica la técnica de selección de patrones utilizada para integrar las submuestras que obtuvo

el mejor resultado: selección secuencial (A), selección aleatoria sin reemplazo (B), Bagging (C), y Boosting (D). La fila "DE" indica las desviaciones estándar sobre la precisión general (PE). Finalmente, los valores resaltados en negrita corresponden al mejor resultado obtenido sobre cada una de las bases de datos utilizadas.

Tabla 7.9. Clasificación con la escalabilidad de algoritmos

		CE ORIGINALES				VOTACIÓN SIMPLE			
		Original	Wilson	SSM	Wilson +SSM	1	2	3	4
Cancer	**Técnica**						A		
	PG	95.6	96.3	94.7	**96.8**	**96.9**	96.8	96.5	95.3
	DE	2.5	2.3	2.0	**1.9**	**2.0**	2.3	1.4	4.2
Heart	**Técnica**						A		
	PG	58.6	**64.4**	58.5	63.3	65.2	**67.4**	61.5	67.4
	DE	6.2	**1.6**	7.2	2.4	4.2	**3.8**	4.4	5.0
Liver	**Técnica**						A		
	PG	65.2	**69.3**	60.6	67.0	63.8	**69.9**	59.1	66.4
	DE	4.8	**7.0**	6.3	4.2	3.7	**3.9**	4.7	3.0
Pima	**Técnica**					D	A	D	A
	PG	65.9	**72.0**	63.0	70.9	70.1	**71.5**	69.0	71.2
	DE	5.2	**2.9**	4.9	2.3	6.2	**3.2**	6.3	3.6
Sonar	**Técnica**						A		
	PG	**82.0**	75.6	78.5	71.7	**79.0**	66.8	76.6	66.3
	DE	**9.4**	14.9	6.3	14.7	**7.0**	10.7	11.9	10.3
Glass	**Técnica**					A	D	A	A
	PG	**70.0**	65.6	67.0	64.5	68.0	61.0	65.0	61.0
	DE	**5.3**	6.7	4.1	6.9	8.6	7.0	8.5	9.1
Iris	**Técnica**					A	B	A	B
	PG	96.0	**96.7**	95.3	95.3	96.0	**97.3**	95.9	**97.3**
	DE	1.5	**2.4**	5.1	3.0	1.5	**1.5**	2.6	**2.8**
Vehicle	**Técnica**					A	D	A	C
	PG	**64.2**	61.3	62.9	60.9	**64.5**	57.9	63.1	57.8
	DE	**1.8**	2.8	1.7	2.8	**2.8**	1.8	2.8	3.1
Wine	**Técnica**					B	A	B	A
	PG	**72.4**	71.2	70.6	70.0	**74.1**	71.8	67.7	72.4
	DE	**3.4**	9.2	6.2	8.4	**8.7**	5.3	8.6	5.3

A partir de estos resultados, podemos ver mejoras significativas al utilizar un SMC para realizar la escalabilidad, principalmente con las bases de datos Heart, Pima y Sonar. En lo concerniente al método de preprocesado, vemos que tanto los resultados con el clasificador único como los obtenidos con el SMC (en los dos esquemas de fusión), en general, mejoran con la Metodología 1 (SMC sin preprocesado) en el caso de Cancer, Sonar, Vehicle y Wine, y con la Metodología 2 (utiliza edición de Wilson) en Heart, Liver, Pima e Iris.

Al analizar las mejoras que en términos de precisión se tienen con los métodos de preprocesado, ya sea aplicado sobre el SMC o con un clasificador único, encontramos que el SMC se ve altamente favorecido al clasificar Heart, Iris, Cancer, Pima y Wine con las metodologías 1 y 2 pues, en la mayoría de los caos, los resultados obtenidos sobre estos CE superan a los obtenidos con el clasificador único. Por otro lado, la clasificación con un solo clasificador tiene su mejor desempeño en Liver, Sonar, Glass y Vehicle, principalmente con la Metodología 4.

En lo que respecta a la técnica de selección de patrones, el método con el que se obtiene mayor precisión es la selección secuencial sin reemplazo (A). Por otro lado, como ya se mencionó, uno de los problemas que más dificultan el tratamiento de grandes bases de datos es precisamente el tamaño con el que cuentan. Por tal motivo, además de los índices de precisión, se incluye también un análisis de los tamaños con que cuentan cada una de las submuestras con las que se entrena el SMC.

La Tabla 7.10 muestra los tamaños que, en promedio, cada una de las submuestras obtuvo posterior a su tratamiento con las técnicas de preprocesado. La primer línea contiene los tamaños promedios de las submuestras utilizadas en la escalabilidad de algoritmos, mientras que la segunda línea proporciona los tamaños del CE original que también fue procesada con los algoritmos de limpieza y reducción de tamaño. No se incluye la Metodología 1 debido a que los tamaños originales y los obtenidos con esta metodología son los mismos.

En estos resultados, vemos que tanto el SMC como el clasificador único tienen unos índices de reducción muy similares. Como era de esperar, la combinación del SMC con los algoritmos de edición y SSM (Metodología 4) cuenta con el menor tamaño, sin sufrir una considerable degradación en la clasificación.

La menor reducción del tamaño se obtiene al combinar el algoritmo de limpieza y el SMC (Metodología 2). Sin embargo, debemos tener en mente que la finalidad de este algoritmo es elevar la calidad del CE al eliminar los patrones atípicos, reduciendo muy poco el tamaño del CE. En este sentido, y en base a los resultados de la Tabla 7.9, vemos que los mejores resultados se obtienen con las metodologías que incluyen su utilización.

Por último, las técnicas de selección de patrones con las que más se reduce su tamaño son, para las Metodologías 2 y 4, la selección aleatoria sin reemplazo (B) y Bagging, en tanto que para la metodología 3, la mayor reducción se obtiene con Boosting. Es importante notar que, tanto el método de Bagging como el de Boosting favorecen la creación de subconjuntos con un alto contenido de patrones redundantes [Die., 97], situación por la que tal vez, al utilizar la Metodología 4, se reduce tan significativamente el tamaño del conjunto de datos. De ser así, podemos entonces decir que muchos de los patrones con los que cuentan estos subconjuntos

son repetidos, aunque no necesariamente atípicos, y que un gran número de ellos son patrones que se encuentran alejados de las fronteras de decisión. Para evaluar esta situación, en el siguiente apartado se incluye un análisis de la cantidad de patrones redundantes y su influencia sobre el grado de precisión.

Tabla 7.10. Tamaño de las submuestras en la escalabilidad de algoritmos

	CE	METODOLOGÍA 2				METODOLOGÍA 3				METODOLOGÍA 4			
	Original	A	B	C	D	A	B	C	D	A	B	C	D
Cancer													
SMC		528	337	347	519	56	438	427	60	20	47	44	34
Original	546	**529**				**56**				**29**			
Heart													
SMC	216	129	112	106	179	121	152	152	113	36	35	26	38
Original		**138**				**122**				**36**			
Liver													
SMC	276	163	153	48	175	166	172	194	137	58	46	48	51
Original		**175**				**164**				**59**			
Pima													
SMC	615	423	259	361	451	311	403	382	256	92	85	85	87
Original		**428**				**303**				**77**			
Sonar													
SMC		117	90	90	117	76	108	120	72	35	26	28	33
Original	167	**133**				**70**				**38**			
Glass													
SMC		101	93	103	106	114	117	101	94	52	49	47	49
Original	174	**119**				**99**				**43**			
Iris													
SMC		113	113	114	115	32	28	28	24	20	22	20	19
Original	120	**116**				**21**				**11**			
Vehicle													
SMC		384	384	414	407	458	462	391	382	143	139	148	140
Original	678	**235**				**419**				**183**			
Wine													
SMC		95	90.4	96	104	65	66	60	93	15	15	17	16
Original	144	**104**				**60**				**16**			

7.4 INFLUENCIA DE LA REDUNDANCIA EN LA CLASIFICACIÓN

Algunos estudios realizados sobre algoritmos que seleccionan patrones con reemplazo establecen que, en promedio, el 63% de los patrones resultantes en la nueva submuestra son repetidos [Die., 97]. Se considera también que esta redundancia puede contribuir a un deterioro de la clasificación. Para determinar tal situación, en este apartado, se presentan resultados experimentales, que incluyen la implementación de un algoritmo capaz de detectar y eliminar patrones que se encuentran repetidos en cada una de las clases, dejando un solo representante de

ellos. Para esto, se incluyen los resultados en la clasificación antes y después de haber eliminado los patrones repetidos, así como también los resultados obtenidos con la clasificación tradicional.

Los resultados de la Tabla 7.11 corresponden a experimentos realizados con un SMC que consta de tres elementos. Los métodos de submuestreo con los que se formó el SMC tres: Arc-x4, Boosting y Bagging. La ejecución de cada uno de estos métodos se realizó de forma separada sobre cada una de las clases del CE. Las bases de datos utilizadas son las mismas que se emplearon en los experimentos de la Sección 7.1, excepto Ism. El esquema de fusión utilizado es la votación por mayoría simple. Finalmente, los resultados plasmados en la Tabla 7.11, corresponden a la media geométrica (g) y su correspondiente desviación estándar (entre paréntesis).

Tabla 7.11. Influencia de patrones redundantes en la clasificación

	Original	CON REDUNDANCIA			SIN REDUNDANCIA		
		Arc-x4	Boosting	Bagging	Arc-x4	Boosting	Bagging
Glass	86.7(12.2)	88.2(10.8)	84.1(11.5)	84.9(10.6)	83.9(15.8)	83.6(15.0)	83.1(15.2)
Vehicle	56.5(4.2)	67.0(3.3)	65.9(2.44)	66.0(3.9)	63.1(7.3)	62.7(4.89)	61.5(6.0)
Phoneme	73.8(6.0)	74.2(8.2)	74.3(8.1)	74.3(7.7)	73.0 (6.8)	73.8(5.5)	73.8(7.2)
Satimage	70.9(15.5)	76.3(7.4)	75.3(5.0)	76.3(3.6)	74.0(7.4)	73.3(6.7)	73.2(10.7)

Analizando estos resultados, vemos que el grado de redundancia existente en las clases tiene muy poca influencia en la precisión, debido a que la precisión obtenida con el SMC libre de redundancia siempre es menos preciso que el SMC entrenado con la redundancia. Esto, entre otras causas, puede estar dado por la compensación que se tiene al fusionar las decisiones individuales. Por otro lado, al comparar el SMC sin redundancia con la clasificación tradicional, vemos que, a excepción de Glass, el SMC sin redundancia se mantiene ligeramente por encima y, en el peor de los casos, igual que el clasificador único.

Por último, el porcentaje estimado de redundancia cuando se utilizan algoritmos que contemplan la selección con reemplazo es un aspecto que se analiza en las Tablas 7.12, 7.13 y 7.14. Los resultados mostrados en estas tablas son la cantidad y porcentaje de patrones redundantes encontrados (y eliminados) en cada una de las clases de las bases de datos con las que se ha experimentado.

Los resultados de estas tablas nos indican que las submuestras que más patrones repetidos tienen son las formadas con Boosting y Bagging, predominando en el caso de Phoneme. Asimismo, la base de datos donde se encuentran menos patrones repetidos corresponde a Vehicle. Es interesante notar que, en general, la clase que más patrones repetidos tiene es la clase minoritaria, clase en donde el algoritmo de selección no tiene influencia pues, como se mencionó en la Sección 7.1, los patrones de la clase minoritaria (o clase 1) se mantienen intactos,

ejecutando únicamente los métodos de selección de patrones sobre la clase mayoritaria. Esto nos indica que más del 60% de los patrones contenidos en esta clase son repetidos desde su inicio. Con esto, surgen varias interrogantes: si la mayor cantidad de patrones de la clase minoritaria son redundantes, ¿qué sucedería si estos fuesen eliminados desde antes de integrar el SMC?; de entrada, el tamaño de las submuestras resultantes se reduciría en un 60%, pero ¿qué sucedería con los índices de precisión?, ¿se mantendrían o disminuirían? Estas interrogantes se establecen para ser tratadas en trabajos futuros.

Tabla 7.12. Redundancia por clase con el método Arc-x4

	Arc-x4					
	Clase1	Clase2	Total	%Clase1	%Clase2	%Global
Glass	14.5	4.1	18.6	60.4	16.9	38.7
Vehcle	107.7	62.7	170.4	8.5	5.4	7.0
Satimage	265.9	74.9	340.8	53.2	15.0	33.5
Phoneme	808.6	479.5	1,288.1	63.8	37.8	50.8

Tabla 7.13. Redundancia por clase con el método Boosting

	Boostig					
	Clase1	Clase2	Total	%Clase1	%Clase2	%Global
Glass	16.6	7.5	24.1	69.2	31.0	44.1
Vehicle	115.0	67.6	182.6	9.1	5.3	7.2
Satimage	324.3	147.1	471.4	64.9	29.4	47.1
Phoneme	866.9	665.9	1,532.7	68.4	52.5	60.4

Tabla 7.14. Redundancia por clase con el método Bagging

	Bagging					
	Clase1	Clase2	Total	%Clase1	%Clase2	%Global
Glass	14.8	3.8	18.6	61.7	15.7	38.7
Vehicle	108.4	50.5	158.9	8.6	4.0	6.3
Satimage	313.6	50.5	364.1	62.7	10.1	36.4
Phoneme	811.1	433.1	1,244.2	64.0	34.2	49.1

7.5 CONCLUSIONES

En el presente capítulo, se desarrollaron metodologías que adaptan ciertas técnicas de SMC, de limpieza, de reducción y de aumento del tamaño del CE, para tratar problemas relacionados con el desbalance, el tamaño excesivo y la redundancia existente en el conjunto de datos.

Para tratar el problema del desbalance, se utilizaron bases de datos de dos tipos: de dos clases y de más de dos clases. Las propuestas de solución para bases de datos de dos clases contemplan el empleo de un SMC, la incorporación de

patrones sintéticos y la utilización de distancias ponderadas para editar el conjunto de datos y/o al ejecutar la regla 1-NN. Además de estos métodos, se hace uso de los algoritmos de preprocesado: edición de Wilson y el algoritmo de reducción SSM. En los resultados obtenidos con estas propuestas, vemos que el método más adecuado para el tratamiento de bases de datos desbalanceadas está directamente relacionado con el grado de desbalance existente entre las clases. De tal manera que, para CE que tienen desbalance poco significativo, la mejor opción es utilizar un SMC construido con selección aleatoria sin reemplazo de patrones. En tanto que para CE que tienen un desbalance muy elevado (como el caso de Ism), el sobre-entrenamiento resulta ser la mejor opción. Pudo observarse también que, cuando estas propuestas van acompañadas del algoritmo de edición de Wilson, su desempeño mejora significativamente.

En lo que respecta a la utilización de la distancia ponderada, para la edición de Wilson o para la regla 1-NN, vemos que, si bien no muestran un mejor desempeño que las otras dos propuestas, no es despreciable del todo, pues con algunos CE presentan un comportamiento bastante favorable, sobre todo cuando se utiliza la edición de Wilson con distancia ponderada, al mismo tiempo que se clasifica con la distancia Euclídea. El tamaño de los conjuntos de datos fue un aspecto sumamente estudiado. En este sentido, encontramos que, en definitiva, tanto el SMC como la utilización de distancias ponderadas son los que menos recursos computacionales requieren para su ejecución. Por el contrario, la propuesta que maneja más grandes conjuntos de datos es la que usa el SMOTE. Sin embargo, aún cuando estos tamaños se acercan al doble de los patrones contenidos en la base de datos original, vemos que, al utilizar los algoritmos de SSM y Wilson, este problema es parcialmente resuelto, manteniendo al mismo tiempo, buenos índices de precisión.

Por otro lado, como pudo verse, el tratamiento de bases de datos desbalanceadas de más de dos clases, requiere además la determinación de clases minoritarias y clases mayoritarias. Para esto, se propuso establecer un umbral que determina el tamaño máximo que una clase *debe* tener para ser considerada como minoritaria. En los experimentos realizados, pudimos ver que la propuesta aquí formulada para calcular el umbral, considerando el promedio de patrones por clase, supera en su totalidad a la propuesta realizada por García [Gar., 02].

Para tratar CE con tamaño excesivo, se propusieron cuatro metodologías que incluyen la utilización de un SMC. Los resultados experimentales obtenidos con estas metodologías nos permitieron demostrar el buen rendimiento de estos sistemas al ser aplicados a tal situación, manteniendo y, en la mayoría de los casos, superando los resultados que se obtienen cuando los CE se tratan de forma conjunta. De una manera más concreta, la metodología que contempla la utilización del SMC + Wilson y que el SMC es integrado con la selección secuencial de patrones resulta ser la mejor alternativa de solución.

El último aspecto abordado en este capítulo se centró en el desarrollo e implementación de un algoritmo que nos permitiera determinar el grado de influencia de la redundancia sobre los índices de precisión. El algoritmo fue ejecutado sobre los SMC formados con métodos que incluyen la selección de patrones con reemplazo: Bagging, Boosting y Arc-x4. Una vez analizados los resultados, fue posible darnos cuenta que, si bien existe una importante cantidad de patrones redundantes en las clases del CE, estos no son perjudiciales. Sobre este punto queda todavía mucho por estudiar y, en este sentido, algunas de las líneas abiertas para futuras investigaciones se refieren a un análisis más profundo y exhaustivo de la naturaleza de los patrones redundantes y de su localización en el espacio de representación.

Capítulo 8

Fusión de clasificadores

Cuando se trabaja con SMC, uno de los métodos más populares para combinar las decisiones individuales es la votación por mayoría. Sin embargo, cuando el desempeño de los miembros del SMC no es uniforme, la eficiencia de este tipo de votación puede verse seriamente afectada. Por todo ello, en el presente capítulo se proponen nuevos métodos para combinar las decisiones individuales. Estas propuestas incluyen diversas técnicas para ponderar las decisiones de los miembros de un SMC. En particular, se estudian dos modalidades para asignar los pesos: de forma dinámica y de forma estática. Para la primera de estas modalidades, se acude a métodos de ponderación de distancias ya existentes, que, en este trabajo, se adecuan para su funcionamiento con un SMC. Por su parte, la segunda modalidad hace uso de un método de estimación de la probabilidad del error.

8.1 VOTACIÓN POR MAYORÍA

Cuando se opta por utilizar la votación por mayoría, la decisión final se obtiene de acuerdo al total de los votos otorgados por los clasificadores a cada una de las clases, asignándole al patrón de entrada, la etiqueta de aquella clase que haya

obtenido el mayor número de votos. Para ello, se distinguen dos diferentes formas en que el clasificador proporciona este voto: voto ponderado y voto no ponderado.

Cuando se utiliza un voto no ponderado o simple, se dice que el valor de cada uno de los votos es igual a 1, de tal forma que, al fusionar las decisiones de todos los clasificadores, se obtiene un valor entero. Por el contrario, cuando se utiliza un voto ponderado, cada uno de los votos tiene un valor comprendido entre 0 y 1, de tal forma que cuanto más cercano sea este valor a 0, menor influencia tendrá al fusionar las decisiones y, por el contrario, cuanto más cercano sea a 1, mayor será su influencia sobre la decisión final.

8.2 VOTACIÓN POR MAYORÍA PONDERADA

Debido a que la precisión individual de cada clasificador no siempre es la misma, resulta necesario implementar métodos que otorguen mayor peso a aquel clasificador que tenga mejor desempeño y/o más información con respecto a un patrón de prueba. De esta forma, la opinión de un clasificador con mayor efectividad tendrá también mayor influencia sobre la decisión final.

En el presente estudio, se realizó la implementación de métodos de ponderación de las decisiones individuales, con dos ideas generales: ponderación dinámica y ponderación estática de los clasificadores. En la primera, se hace uso de las distancias obtenidas hacia un patrón de prueba por cada uno de los clasificadores y, mediante la utilización de fórmulas especificas, se asigna mayor peso a aquel clasificador donde se encuentre el vecino con la menor distancia al patrón de prueba. Por su parte, la ponderación estática utiliza, para la asignación de pesos, el método de estimación del error *leave-one-out* (ver Sección 2.3.4.1), con el que se evalúa el desempeño que tiene cada uno de los clasificadores. La asignación de los pesos se realiza con el siguiente criterio: a menor error, mayor peso y, por el contrario, a mayor error menor peso.

8.2.1 PONDERACIÓN DINÁMICA SEGÚN DUDANI

En su trabajo [Dud., 76], Dudani propuso la ponderación de la regla *k*-NN en función de las distancias. La diferencia de esta propuesta con la regla *k*-NN usual estriba en que, a cada uno de los *k* patrones que resultaron ser los vecinos más cercanos, se le asocia un factor de ponderación o peso. Un patrón desconocido se asigna entonces a aquella clase en que la suma de los pesos de sus representantes (entre los *k* vecinos más cercanos) alcance el valor máximo. Al aplicar este mismo criterio para realizar la ponderación dinámica de los componentes individuales en un SMC, el valor de *k* es sustituido por el número de clasificadores que constituyan el SMC. La forma de asignar los pesos tiene el siguiente procedimiento:

1.- Sea d_j (j = 1,..., H) la distancia de **y** a su vecino más cercano j-ésimo en cada clasificador individual (H representa el número de clasificadores utilizados).

2.- Ordenar las distancias de manera no decreciente: $d_1 \leq d_2 \leq,\ldots, \leq d_H$.

3.- El peso para el clasificador D_j es calculado mediante:

$$w(D_j) = \begin{cases} \dfrac{d_H - d_j}{d_H - d_1} & si\ \ d_H \neq d_1 \\ 1\ \ en\ \ caso\ \ contrario \end{cases}$$

8.2.2 PONDERACIÓN DINÁMICA POR PROMEDIO DE DISTANCIAS

Este método, para establecer el peso del clasificador, también utiliza las distancias obtenidas por cada componente del SMC hacia un determinado patrón **y**, pero, en este caso, para asignar los pesos, se realizan los siguientes pasos:

1.- Sea d_j (j = 1,..., H) la distancia de **y** a su vecino más cercano j-ésimo en cada clasificador individual.

2.- La ponderación está dada por: $w(D_j) = \sum_{i=1}^{H} \dfrac{d_i}{d_j}$

8.2.3 PONDERACIÓN DINÁMICA SEGÚN EL ORDEN

Este tipo de ponderación toma parte del principio de ponderación para los *k* vecinos que Dudani incluye en su publicación [Dud., 76]. Para ponderar los clasificadores, también requiere una ordenación de las distancias. El peso w_j asignado al clasificador D_j toma valores enteros desde *k* hasta 1 y está directamente relacionado con la posición en que se encuentre la distancia proporcionada por cada uno de los *k* vecinos. La ponderación considera la siguiente fórmula:

$$w(D_j) = k - j + 1$$

donde *k* es el patrón más alejado a **y** entre los k vecinos y *j* es el vecino más cercano a **y** encontrado en el clasificador D_j.

Para implementar este método con el SMC, se realizan las siguientes adecuaciones: *k* corresponde al número de componentes individuales (es decir, H) y *j* es la posición en la que se encuentra cada clasificador después de haber realizado la ordenación de las distancias.

8.2.4 PONDERACIÓN ESTÁTICA CON MÉTODO L (LEAVE-ONE-OUT)

Este método de ponderación es estático y se realiza en la etapa de entrenamiento del SMC. Para realizar la ponderación, utiliza un método de estimación del error que hace uso de la siguiente función *E* indicadora de acierto y error.

$$E(y, x_i) = \begin{cases} 0 & \text{si } t(y) = t(x_i) \\ 1 & \text{en caso contrario} \end{cases}$$

donde t es la etiqueta de clase, x_i es un patrón de entrenamiento (i = 1,...,m), **y** es el patrón evaluado.

Para realizar la ponderación se realiza el siguiente procedimiento:

1. Para todo j (j = 1,..., H), determinar la estimación del error según [Bai., 93]:

$$e_j = \frac{1}{m} \sum_{x \in M} E(y, x)$$

2. Obtener $e_{total} = \sum_{j=1}^{H} e_j$

3. Asignar peso para h_j mediante: $w(D_j) = 1 - \frac{e_j / m}{e_{total}}$

8.2.5 CLASIFICACIÓN CON PONDERACIÓN ESTÁTICA vs DINÁMICA

La Tabla 8.1 muestra los resultados correspondientes a los promedios de la precisión general, obtenidos en la fusión con seis conjuntos de datos, utilizando los métodos de votación introducidos en las secciones anteriores, sobre un SMC formados por tres clasificadores. Además, para cada uno de los cinco métodos de votación, se ha experimentado con cuatro técnicas de submuestreo: selección secuencial, selección aleatoria sin reemplazo, Bagging y Boosting. Cabe indicar que los valores en negrita dentro de esta tabla hacen referencia a los índices más elevados de precisión con alguno de los cuatro métodos de submuestreo utilizados.

Como punto de referencia, en la Tabla 8.1, se proporcionan también los resultados de la clasificación sobre el conjunto de datos original (mediante la clasificación tradicional con un único clasificador). A partir de estos resultados, vemos que, para todos los conjuntos de datos, existe al menos un método de fusión que supera (en términos de precisión) la clasificación obtenida al emplear un

clasificador único. Algo similar sucede cuando comparamos los resultados obtenidos con la votación simple y la votación ponderada, donde puede apreciarse que los mejores resultados se obtienen siempre con alguno de los cuatro métodos de votación ponderada. De forma específica, la ponderación según Dudani tiene mejor desempeño que cualquier otro método en el caso de la base de datos Liver. Análogamente, la ponderación dinámica según el orden es mejor en los casos de Cancer y Glass, mientras que la ponderación por promedio tiene los mayores índices de precisión en Heart, Pima y Vehicle.

Tabla 8.1. Resultados experimentales de la fusión de clasificadores mediante votación ponderada

	Cancer	Heart	Liver	Pima	Glass	Vehicle
Clasificación original	95,6	58,2	65,2	65,9	70,0	64,2
Votación simple						
Sel. secuencial	**96,9**	**65,2**	**63,8**	68,9	**68,0**	**64,5**
Sel. A. sin remplazo	66,4	50,4	57,1	59,4	56,5	62,1
Bagging	72,1	45,2	50,1	60,0	60,5	60,6
Boosting	94,2	57,8	62,0	**70,1**	62,5	60,4
Ponderación según Dudani						
Sel. secuencial	95,6	58,2	***65,5***	***68,4***	***70,0***	***64,2***
Sel. A. sin remplazo	68,5	53,0	56,2	59,1	67,0	61,0
Bagging	74,2	47,4	52,2	60,3	65,0	60,9
Boosting	***95,9***	***58,5***	60,9	67,6	66,5	64,2
Ponderación según el orden						
Sel. secuencial	95,9	***61,1***	***62,6***	68,2	***71,0***	***64,5***
Sel. A. sin remplazo	65,8	54,1	53,0	62,1	62,0	62,3
Bagging	72,4	47,8	49,3	60,9	61,5	60,8
Boosting	***99,3***	57,4	59,4	***70,1***	66,0	62,8
Con promedio de distancias						
Sel. secuencial	***96,5***	***65,6***	***65,2***	68,4	***68,0***	***64,7***
Sel. A. sin remplazo	62,0	49,6	57,1	59,1	59,0	59,0
Bagging	70,8	45,9	50,1	60,3	62,5	63,4
Boosting	93,6	57,8	62,3	***70,9***	63,0	61,5
Ponderación con método L						
Sel. secuencial	***96,9***	***65,2***	***63,8***	68,9	***68,5***	***63,7***
Sel. A. sin remplazo	66,4	50,4	57,1	59,4	56,0	62,9
Bagging	72,1	45,2	50,1	60,0	60,5	59,8
Boosting	94,2	59,6	62,0	***70,1***	63,0	61,0

Por otro lado, en lo que respecta a la comparación entre la votación ponderada dinámica y la votación ponderada estática, vemos que, en el 100% de los casos, los mejores resultados se obtienen con algún método de ponderación dinámica. Finalmente, en lo que se refiere al método de submuestreo, encontramos que, en general, la selección secuencial (en 22 de los 30 casos) es la que más favorece la precisión, seguida de Boosting con los mejores resultados en tan solo 3 casos. Por último, la selección aleatoria sin reemplazo y Bagging, para estos conjuntos de datos, no proporcionan mejora alguna.

8.2.6 PONDERACIÓN DINÁMICA CON DISTANCIA INVERSA

Esta es una ponderación también propuesta por Dudani [Zav., 97]. A diferencia de los métodos anteriores, este tipo de ponderación no requiere de una ordenación de las distancias. Para este tipo de ponderación, se incluye una condición par evitar la división entre 0, la fórmula utilizada es la siguiente:

$$w(D_j) = \frac{1}{d_j} \qquad si\ d_j \neq 0$$

8.2.7 PONDERACIÓN DINÁMICA SEGÚN SHEPARD

Este tipo de ponderación se basa en una función propuesta por Shepard [She., 87] que establece la existencia de una ley universal generalizada que relaciona las distancias entre un par de elementos en un espacio psicológico y la probabilidad de que estos elementos se confundan uno con otro. Para determinar la probabilidad de la confusión, propone una función exponencial negativa entre el par de elementos, en tanto que la distancia la define como la distancia más corta que transforma la representación de los dos elementos de interés en otra, es decir, la distancia logarítmica de la información.

La ley se considera universal debido a que minimiza cada distancia calculada. En la aplicación de la distancia, los parámetros $\boldsymbol{\alpha} = \boldsymbol{\beta} = 1$ son *constantes* y no varían durante el proceso de clasificación [Zav., 97].

$$w(D_j) = e^{-ad_j^{\beta}}$$

8.2.8 PONDERACIÓN DINÁMICA DE SHEPARD MODIFICADA

Este método considera la combinación de los métodos descritos en las Secciones 8.2.3 y 8.2.7. La fórmula utilizada es la propuesta por Shepartd, con la variante que el valor del parámetro **α** no es *constante*, sino que varía con cada patrón de prueba que se analiza. Para asignar el valor de **α**, se realiza el mismo mecanismo utilizado en el método de la Sección 8.2.3, en el cual hay una etapa previa a la asignación de pesos, en la que las distancias obtenidas por cada componente individual del SMC son ordenadas de forma decreciente. De este modo, el valor de **α** será mayor para aquel componente que proporcione la mayor distancia a un patrón dado y, por consiguiente, el peso asignado será menor.

1.- Sea d_j(j = 1,..., H) la distancia de **y** a su vecino más cercano j-ésimo en cada clasificador individual.

2.- Ordenar las distancias de manera decreciente: $d_H \geq,\ldots, \geq d_2 \geq d_1$.

3.- $\boldsymbol{\alpha} = k - j + 1$ (k es el patrón más alejado a **y** entre los k vecinos)

4.- El peso para el clasificador D_j es calculado mediante: $w(D_j) = e^{-ad_j^{\beta}}$

8.2.9 CLASIFICACIÓN CON PONDERACIÓN DINÁMICA

A diferencia de los experimentos de la Sección 8.2.5, en este apartado, se incluye el análisis de cuatro métodos de ponderación dinámica sobre un SMC formado por 5, 7 y 9 clasificadores, utilizando para ello 14 bases de datos.

Debido a la gran cantidad de resultados obtenidos con cada uno de estos métodos, en la Tabla 8.2, sólo se han incluido los mejores resultados observados en la clasificación. La tabla cuenta con cinco diferentes métodos de fusión de las decisiones, cuatro que utilizan la ponderación dinámica (por promedio de distancias, según Shepard, Shepard modificada y ponderación con distancia inversa) y la votación simple. Los números que aparecen en la segunda fila de la tabla indican la cantidad de clasificadores utilizados. Los valores resaltados en negrita corresponden a los mejores resultados obtenidos por cada uno de los métodos de ponderación y la(s) letra(s) mayúscula(s) que aparece(n) sobre los valores de precisión indica(n) el método de selección de patrones con el que se obtuvo dicho resultado: (A) selección aleatoria sin reemplazo, (B) Bagging, (C) Boosting, y (D) Arc-x4 (no se incluye la selección secuencial, por ser prohibitiva para algunos CE utilizados, ver Sección 5.3). Para una consulta a detalle, puede acudirse a los anexos del Apéndice IV.

Primeramente, de manera similar a lo mostrado en las Tablas 8.2 y 8.3, al comparar la precisión obtenida en la clasificación con el SMC respecto a la que se tiene cuando se usa un clasificador único, vemos que a excepción de Segment, para todos las restantes bases de datos, con algún método de fusión, ya sea ponderado o simple, se logra tener mejor desempeño.

En lo que respecta al número de clasificadores utilizados para tener los mejores resultados, en general, se obtienen al utilizar 5 y 9 clasificadores. De forma más detallada, vemos que los mejores resultados para la votación simple y la votación ponderada por promedio de distancias se obtuvieron al considerar nueve clasificadores en seis de los conjuntos empleados (la votación simple con Pima, Iris, Vehicle, Wine, German y Waveform, y la votación por promedio en Liver, Pima, Wine, German, Satimage y Waveform). Para la ponderación según Shepard, los mejores resultados se obtienen al utilizar siete clasificadores en ocho casos (Cancer, Heart, Glass, Iris, Wine, German, Satimage y Phoneme). Para la ponderación que utiliza la modificación a la distancia de Shepard, vemos que los mejores resultados se obtienen al utilizar cinco clasificadores en siete bases de datos (Liver, Pima, Iris, Vehicle, German, Phone y Waveform). Finalmente, cuando se utiliza la distancia inversa para ponderar las decisiones, tenemos que los mejores resultados se obtienen cuando se utilizan siete y nueve clasificadores, cada

uno de ellos en cinco diferentes problemas (con siete clasificadores en Cancer, Heart, Sonar, Iris, Satimage, y con nueve clasificadores en Liver, Glass, Wine, German, Waveform).

Tabla 8.2. Precisión con ponderación dinámica por promedio y según Shepard

	CE original	VOTACIÓN SIMPLE			POR PROMEDIO			SEGÚN SHEPARD		
		5	7	9	5	7	9	5	7	9
Cancer	95.6	A 95.9	**A** **97.4**	B 96.4	A, B 95.3	**A** **97.1**	A, B 96.2	A 95.8	**A** **96.7**	B 96.4
Heart	58.2	A 60.7	**A** **67.4**	C 63.0	B 64.8	**A, B** **66.7**	B 63.4	B 61.1	**B** **62.6**	C 61.1
Liver	65.2	**B** **65.8**	B 64.1	C 65.2	A, B 64.9	B 63.5	**D** **66.1**	B 65.5	D 59.7	**A** **65.8**
Pima	65.9	D 70.7	A 71.9	**A** **72.8**	C 71.9	B 70.9	**B** **72.7**	**B** **68.4**	A 67.3	A 66.9
Sonar	82.0	**B** **73.7**	D 71.7	D 72.2	**B** **77.6**	A, D 75.6	D 74.2	**B** **75.6**	D 74.2	B 73.7
Glass	70.0	**B** **65.0**	C 63.0	A 61.5	**D** **65.0**	A, D 64.5	B 64.0	D 65.0	**D** **65.5**	B, C 63.5
Iris	96.0	B,A,C 96.0	**D** **98.0**	**A** **98.0**	C 97.3	**D** **98.0**	A 97.3	**B** **97.3**	**D** **97.3**	**A** **97.3**
Vehicle	64.2	A 62.2	B 60.7	**C** **62.3**	**A** **64.5**	A 61.0	C 63.9	A 65.1	A 63.4	**C** **65.6**
Wine	72.4	C 70.0	C 73.5	**B** **75.9**	C 67.7	A 73.5	**B** **77.7**	A, B 70.0	**B** **74.1**	B 73.5
German	65.2	C 68.8	D 69.9	**B** **70.2**	A 68.8	A 69.4	**B** **70.8**	B 67.4	**C** **68.5**	B 68.1
Satimage	83.6	C 83.0	**A** **83.0**	A 82.9	C 83.1	A 83.1	**C** **83.4**	A 83.2	**B** **83.2**	B 83.0
Phoneme	76.1	**A** **75.9**	A 75.8	A 75.0	**A** **76.5**	A 75.9	A 75.4	**A** **76.0**	**A** **76.0**	A 75.0
Waveform	78.0	A 80.8	B 81.9	**B** **83.2**	A 80.8	B 81.9	**B** **83.2**	A 80.5	B 81.5	**B** **83.1**
Segment	94.8	**A** **89.9**	A 89.0	B 87.7	**B** **92.1**	A 91.8	A 91.0	C 93.0	A, B 92.9	**A** **93.5**

Cuando analizamos el método de submuestreo con el que se obtienen los mejores resultados, vemos que, en general, la selección aleatoria sin reemplazo ofrece el mejor resultado en ocho bases de datos, seguida de Bagging en cinco casos y, finalmente, Arc-x4 en cuatro y Boosting en tres.

Por otro lado, respecto a los métodos con los que se integraron los subconjuntos de datos y que tienen el mejor desempeño, vemos que, en la votación simple, predomina la selección aleatoria sin reemplazo (en siete de los casos). En la votación por promedio de distancias, en la de Shepard y en la de Shepard modificada, los mejores resultados corresponden a Bagging. Por último, en la

ponderación con distancia inversa, los mejores resultados se obtienen con los métodos aleatorio sin reemplazo y Bagging, en cinco casos cada uno.

Tabla 8.3. Precisión con ponderación dinámica con Shepard modificado y con distancia inversa

	CE original	SHEPARD MODIFICADA			CON DISTANCIA INVERSA		
		5	7	9	5	7	9
Cancer	95.6	A 96.1	A 95.8	**B 96.4**	A 95.3	**A 97.1**	A, B 96.2
Heart	58.2	B 61.9	**A 62.6**	C 60.7	B 64.8	**A, B 66.7**	C 63.4
Liver	65.2	**B 65.8**	A 63.8	A 64.4	A 64.6	B 62.9	**D 64.9**
Pima	65.9	**B 68.2**	A 67.7	A 66.8	**C 71.9**	B 70.9	C 69.7
Sonar	82.0	A 82.0	**A 82.9**	B 77.1	A 74.6	**B, D 75.6**	C, D 73.7
Glass	70.0	C 69.0	**B 70.5**	B 68.0	C 83.1	B, D 82.6	**C 83.4**
Iris	96.0	**C 96.7**	**D 96.7**	**A 96.7**	C 97.3	**D 98.0**	A 97.3
Vehicle	64.2	**A 65.2**	A, B 63.4	A 63.2	**A 64.5**	B 62.3	C 63.9
Wine	72.4	B, C 70.0	**B 74.1**	**B 74.1**	A 66.5	A 73.5	**B 77.7**
German	65.2	**B 66.9**	A 67.2	C 66.1	A 68.8	A 69.4	**B 70.8**
Satimage	83.6	A 83.1	**B 83.2**	A 82.7	C 63.5	**D 65.5**	B 64.0
Phoneme	76.1	**A 76.6**	A 75.5	A 75.7	**A 76.5**	A 75.9	A 75.4
Waveform	78.0	**B 78.2**	B 78.2	B 78.1	A 80.8	B 81.9	**A 83.5**
Segment	94.8	C 93.0	A, C 92.9	**A 93.3**	**B 91.5**	A 91.0	A 90.2

Finalmente, si buscamos el mejor resultado por cada uno de las bases de datos sin considerar la cantidad de clasificadores utilizados, tenemos que con la votación simple, la ponderada por promedio y la que utiliza la distancia inversa proporcionan los mejores resultados en cinco diferentes casos, en tanto que las ponderaciones que utilizan Shepard o Shepard modificado tan sólo en dos casos (cada una). Es importante aclarar que, en el caso de Iris, con más de un método de ponderación se obtiene el mejor resultado.

8.3 CONCLUSIONES

En el presente capítulo, se compararon los resultados obtenidos al realizar la fusión de clasificadores con seis métodos de ponderación dinámica y uno de ponderación estática, sobre un SMC formado por 3, 5, 7 y 9 clasificadores. Con la intención de validar las posibles mejoras, en el análisis se incluyeron también los resultados obtenidos con el clasificador único y con la votación simple.

La experimentación parte con pruebas sobre seis conjuntos de datos y cuatro métodos de ponderación: tres métodos de ponderación dinámica y uno de ponderación estática (Tabla 8.1). Estos primeros resultados se realizaron con un SMC formado por tres clasificadores. Para la integración del SMC, se utilizaron cuatro métodos de submuestreo, de los cuales, contundentemente, la selección secuencial es el método de selección de patrones ideal. Los resultados de la Tabla 8.1 nos permitieron ver el pobre desempeño (en términos de precisión) y el alto costo computacional que la votación con ponderación estática tiene, frente a los métodos de votación con ponderación dinámica y la votación simple, motivo por el cual no resulta recomendable su uso. Entre los métodos de ponderación dinámica, no se logra observar un comportamiento uniforme, situación que dificulta la recomendación de alguno de ellos en particular. Por tal motivo, se implementaron nuevos métodos de ponderación para que, al ser aplicados sobre SMC con un mayor número de clasificadores y diferentes métodos de submuestreo, nos permitan encontrar y determinar el método más adecuado para ponderar las decisiones individuales de un SMC.

En la Sección 8.2.9, se proporcionaron los resultados de tres nuevos métodos de ponderación de las decisiones individuales. Para los experimentos, se utilizaron 14 conjuntos de datos y SMC formados por 5, 7 y 9 clasificadores. Estos resultados nos permiten ver que los métodos más recomendables para fusionar las decisiones son la votación simple, la votación ponderada por promedio y la votación ponderada con distancia inversa. Por el contrario, los métodos más adecuados para construir el SMC son, principalmente, la selección aleatoria sin reemplazo y, como alternativas, Bagging y Arc-x4. Esto último resulta ser muy conveniente, ya que estos tres métodos son los que tienen asociado un menor coste computacional, debido a la simplicidad de sus procesos.

Capítulo 9

Sistemas de Múltiples Redes

En el presente capítulo, se proporcionan los resultados experimentales obtenidos al entrenar SMC con dos esquemas de redes: múltiples capa y modular. Los experimentos se realizaron sobre 13 conjuntos de datos, los cuales tienen varias diferencias entre si: cantidad de patrones, número de clases y número de atributos. Las submuestras utilizadas para entrenar el SMC se consiguieron mediante los métodos de selección aleatoria sin reemplazo (A), Bagging (B), Boosting (C) y Arc-x4 (D) (ver Sección 4.1.1). Resultados previos (ver Sección 6.1.2), nos permitieron determinar que 5, 7 y 9 es un número suficiente de clasificadores con los que debe contar un SMC a fin de tener un rendimiento adecuado, situación por la que los SMC aquí utilizados constan de 7 y 9 clasificadores. En todos los casos, el esquema de fusión es la votación por mayoría simple.

9.1 PERCEPTRÓN MÚLTICAPA BACKPROPAGATION

El Perceptrón Múlticapa utilizado consta de 3 capas de neuronas: 1 de entrada, 1 oculta y 1 de salida. Cada una de las capas cuenta con la siguiente cantidad de neuronas:

a) Capa de entrada: número de atributos del patrón.

b) Capa oculta. Se utilizaron dos diferentes estructuras de PM: PMa y PMb. En el primero de ellos, la cantidad de neuronas en la capa oculta es igual al número de atributos más 1 y en el segundo es igual al número de capas ocultas más 1.

c) Capa de salida: número de clases en el problema.

Los pesos iniciales se establecieron de forma aleatoria en el rango [-0.5 y 0.5]. Los valores constantes para la razón de aprendizaje (η) y el momento (α) son 0.9 y 0.7, respectivamente. Para todos los conjuntos de datos, el proceso de aprendizaje se realizó en 5000 iteraciones. Debido a que, en algunas ocasiones, puede alcanzarse el número de iteraciones sin decrecer el error por patrón o el error global, durante el entrenamiento, se evaluó el error observado, estableciendo un máximo error permitido por patrón en 0.01 y máximo error global en 0.001. Finalmente, la función de activación utilizada es la sigmoidal.

En la Tabla 9.1, se proporcionan los resultados de precisión general (PG) y desviación estándar (DE) al utilizar la red PMa. A fin de tener una mejor validación, en la tabla, se incluyen también los resultados correspondientes a la clasificación con sólo un clasificador.

Los resultados obtenidos con el segundo SMC formado con PMb, se muestran en la Tabla 9.2. La única diferencia existente entre los PM utilizados en estos experimentos y los utilizados en los resultados de la Tabla 9.1 se refiere a la cantidad de neuronas que uno y otro tienen en la capa oculta. Esto es, en el SMC formado por PMb, el número de neuronas en la capa oculta es igual al número de capas ocultas más la unidad.

Primeramente, en los resultados obtenidos con los tres esquemas de SMC, encontramos mejoras poco significativas, respecto a cuando se utiliza una sola red, siendo las más importantes en Heart, German, Iris y Waveform con la estructura PMa y, Sonar, German, Phoneme y Segment con la estructura PMb.

Por otro lado, haciendo un análisis comparativo entre el desempeño de los dos SMC (formados con PMa y PMb), tenemos que, en general, el SMC formado por PMb proporciona mejores resultados que el PMa. Más concretamente, en el caso de las bases de datos Liver, Pima, Satimage, Wine y Heart, se obtienen las mejores diferencias en cuanto al rendimiento de ambas configuraciones. Es

importante notar la mejora tan importante que tiene sobre Satimage, con la que mantiene un a precisión superior, hasta en un 40% (con Boosting, 7 clasificadores), sobre la obtenida con PMa. Por otro lado, en ambos esquemas, los métodos de submuestreo que más favorecen su desempeño son la selección aleatoria sin reemplazo (A) y Bagging (B). Por último, la utilización de 9 clasificadores resulta ser la cantidad más conveniente de componentes individuales.

Tabla 9.1. Clasificación con Perceptrón Múlticapa (neuronas en capa oculta = n+1)

		CE original	7 Clasificadores				9 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	PG	94.6	96.5	96.5	92.2	94.6	96.3	95.6	93.0	94.6
	DE	2.0	1.4	1.8	3.0	2.7	2.5	3.4	2.8	1.9
Heart	PG	69.6	80.7	80.7	77.8	75.6	83.3	82.2	82.4	79.6
	DE	2.5	5.2	2.8	3.7	3.3	2.3	5.0	2.4	4.3
Liver	PG	63.2	60.9	65.8	61.7	60.9	62.3	63.8	58.8	60.3
	DE	5.3	7.8	4.7	5.9	4.1	3.1	3.6	7.7	8.4
Pima	PG	71.1	74.1	73.6	72.6	69.0	72.2	72.2	70.1	70.1
	DE	5.9	2.1	2.1	3.5	3.5	3.1	3.5	2.0	2.2
Sonar	PG	74.2	72.7	72.7	74.6	74.6	73.2	74.2	73.2	75.1
	DE	8.2	7.4	5.8	3.7	9.9	6.0	9.9	4.6	9.2
Iris	PG	94.7	96.7	96.0	96.7	95.3	95.3	97.3	90.7	94.7
	DE	4.5	-	2.8	2.4	1.8	3.8	1.5	6.4	3.0
Vehicle	PG	57.7	55.7	46.0	50.6	42.0	44.1	47.3	43.6	40.1
	DE	11.0	5.3	6.7	8.5	6.5	10.3	3.3	6.0	12.6
Wine	PG	94.7	94.1	95.3	92.9	90.6	94.1	95.9	90.6	90.0
	DE	3.8	3.6	2.6	4.5	2.5	4.7	4.9	8.2	6.4
German	PG	65.5	72.2	73.3	74.1	71.3	74.6	71.9	72.8	68.8
	DE	4.4	2.0	1.7	2.4	0.9	2.2	3.1	2.8	1.5
Satimage	PG	70.4	43.1	62.5	32.4	41.2	42.9	44.1	34.6	41.5
	DE	7.8	3.9	22.4	18.2	7.3	16.0	27.4	14.4	21.1
Phoneme	PG	68.7	71.0	71.2	68.4	65.8	71.7	72.6	66.7	67.1
	DE	6.2	6.6	3.1	4.4	10.4	6.7	5.7	11.9	11.1
Waveform	PG	80.7	82.4	82.9	82.0	80.7	83.6	83.6	82.9	79.5
	DE	2.1	2.7	1.0	1.6	2.5	1.7	1.0	1.6	1.8
Segment	PG	94.1	94.5	93.0	93.3	92.5	93.8	93.5	93.0	91.5
	DE	1.6	1.8	1.6	2.0	2.3	1.8	1.5	1.7	1.2

El tiempo requerido para el entrenamiento de cada uno de estos sistemas constituye un factor sumamente importante, ya que el hecho de utilizar n+1 neuronas, en el caso de PMa, o únicamente dos neuronas, en la configuración PMb, supone una diferencia muy significativa en cuanto al coste computacional, resultando altamente ventajosa la opción de la red PMb. Otro aspecto determinante a la hora de elegir un esquema u otro es el índice de precisión obtenido por cada uno de ellos, en lo que claramente se demuestra también la superioridad de la configuración PMb.

Tabla 9.2. Clasificación con Perceptrón Múltiplicapa (neuronas en capa oculta = capas ocultas + 1)

		CE original	7 Clasificadores				9 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	PG	97.1	96.2	96.8	91.8	94.4	96.6	96.6	93.7	93.3
	DE	2.3	1.2	1.7	1.7	2.6	2.1	1.7	3.5	3.3
Heart	PG	81.9	81.5	81.1	80.0	77.8	83.0	81.1	82.9	81.5
	DE	3.3	3.5	1.6	4.0	3.5	2.0	2.8	2.8	5.2
Liver	PG	65.2	60.9	64.1	64.9	62.3	62.6	63.5	62.6	62.9
	DE	7.4	3.7	10.9	2.8	4.2	5.3	5.1	4.0	2.4
Pima	PG	75.2	75.4	75.8	75.3	72.0	72.7	73.1	72.3	71.9
	DE	5.2	3.0	3.1	2.7	2.6	1.7	1.7	2.3	0.9
Sonar	PG	69.3	72.7	74.6	78.5	75.1	75.6	73.7	71.2	69.8
	DE	8.0	8.7	6.6	8.2	9.2	6.9	9.0	6.1	4.8
Iris	PG	94.7	96.0	97.3	90.7	94.7	96.0	97.3	90.7	94.7
	DE	4.5	3.7	2.8	6.4	3.0	3.7	2.8	6.4	3.0
Vehicle	PG	46.7	45.4	51.6	44.6	38.9	51.3	43.0	37.2	37.3
	DE	10.6	5.6	1.7	9.9	10.7	10.6	4.8	7.2	12.0
Wine	PG	92.9	95.9	96.5	93.5	88.2	95.3	95.9	90.6	90.0
	DE	3.4	2.6	2.5	6.4	10.2	4.0	4.5	9.8	6.1
German	PG	73.3	73.8	74.5	73.5	73.7	74.1	74.2	72.7	70.7
	DE	2.9	2.4	1.7	3.2	4.4	3.0	2.1	2.9	2.3
Satimage	PG	70.4	72.9	72.9	72.7	66.2	70.5	71.9	74.1	70.8
	DE	7.8	9.0	6.3	8.1	11.7	9.5	8.0	6.5	9.4
Phoneme	PG	69.7	70.1	69.9	70.1	68.5	70.9	70.9	69.7	70.0
	DE	2.7	2.2	1.5	2.7	2.6	2.5	3.8	2.0	0.9
Waveform	PG	82.0	84.0	83.9	81.3	81.7	83.6	84.1	81.8	81.6
	DE	2.1	1.0	1.0	1.7	3.0	0.8	1.4	0.8	2.5
Segment	PG	73.3	75.9	80.2	79.0	71.8	80.1	81.4	78.8	68.4
	DE	5.1	4.0	2.5	4.4	8.3	2.2	1.6	3.2	6.4

9.2 RED MODULAR

El segundo esquema utilizado en este capítulo es la implementación de SMC formado por redes con estructura modular. Cada componente individual del SMC corresponde a una red modular, todas ellas con la misma estructura (ver Sección 5.1.2). Cada red modular está formada por 5 expertos y una red integradora. Por último, el esquema de fusión utilizado es, al igual que en los experimentos con el perceptrón múltiplicapa, la votación por mayoría simple.

La Tabla 9.3 muestra los resultados obtenidos en la clasificación con este SMC. Las letras mayúsculas de la segunda fila corresponden a los métodos de submuestreo utilizados: aleatorio sin reemplazo (A), Bagging (B), Boosting (C) y Arc-x4 (D).

Tabla 9.3. Clasificación con SMC de redes modulares

		CE original	7 Clasificadores				9 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	PG	88.4	88.4	87.9	82.7	84.9	87.1	86.5	84.5	83.3
	DE	4.6	3.1	3.0	3.7	7.1	4.7	4.2	4.4	3.5
Heart	PG	73.7	81.5	81.5	78.5	79.3	78.9	80.4	83.3	81.9
	DE	8.6	5.4	4.5	1.7	2.4	4.7	7.4	5.5	4.8
Liver	PG	63.5	54.8	62.9	61.2	60.6	62.0	67.0	62.3	59.4
	DE	5.4	8.1	6.9	7.2	3.6	4.9	3.8	4.4	7.9
Pima	PG	66.5	68.0	67.6	66.7	65.8	66.1	67.8	67.2	62.8
	DE	1.6	1.8	3.2	1.4	1.4	2.3	2.4	3.1	7.1
Sonar	PG	65.9	73.7	67.8	66.8	70.7	77.1	70.7	68.3	66.8
	DE	6.2	3.2	4.7	5.1	10.2	12.2	7.1	6.0	3.3
Iris	PG	80.7	78.0	82.0	75.3	80.7	78.0	80.0	75.3	78.0
	DE	11.4	6.9	8.0	11.2	9.3	7.7	6.7	7.7	9.3
Vehicle	PG	36.4	47.1	42.8	45.4	39.6	42.2	43.5	41.7	39.9
	DE	7.1	3.7	10.9	6.2	8.1	4.0	3.8	9.3	6.3
Wine	PG	97.7	93.5	95.9	94.7	92.9	93.5	96.5	90.6	88.8
	DE	2.5	2.5	1.6	7.0	3.4	3.8	3.8	5.7	4.8
German	PG	61.8	73.7	72.4	70.5	71.6	73.2	72.7	67.4	63.3
	DE	18.0	1.3	4.5	9.0	5.1	1.9	4.1	9.1	8.7
Satimage	PG	34.9	38.4	49.5	41.7	34.9	58.9	48.9	45.7	48.4
	DE	14.2	11.3	9.8	18.8	14.5	5.3	7.5	3.7	6.7
Phoneme	PG	67.9	67.2	68.9	68.2	68.2	67.7	68.1	68.0	68.4
	DE	4.5	5.5	3.5	3.7	3.9	4.2	4.2	3.7	3.4
Waveform	PG	77.2	81.6	82.0	79.1	79.8	79.2	80.2	79.8	78.1
	DE	2.7	1.7	2.6	4.1	2.8	3.8	3.3	2.2	4.1
Segment	PG	78.2	75.0	74.9	77.7	77.6	76.9	74.5	75.9	75.0
	DE	5.6	2.2	2.2	4.2	5.1	2.4	1.8	1.9	1.4

En estos resultados, vemos que, el mejor desempeño del SMC se encuentra sobre los conjuntos Heart, Sonar, Vehicle, German, Satimage, Phoneme y Waveform, con incrementos entre el 10 (Sonar, Vehicle y German) y 24% (Satimage) sobre los índices de precisión obtenidos con la utilización de una sola red.

También encontramos que, análogamente a lo ocurrido con los SMC que utilizan PM, los mejores resultados se obtienen con los SMC formados con la selección sin reemplazo (A) y con Bagging (B). Por otro lado, contrario a los resultados de las Tablas 9.1 y 9.2, al utilizar este tipo de red (modular), los resultados son mejores cuando se utilizan 7 clasificadores, situación muy ventajosa, ya que el costo computacional asociado a 7 clasificadores es inferior al necesario cuando se utilizan 9 clasificadores (Tabla 9.4).

9.3 TIEMPOS DE PROCESAMIENTO

Una de las principales limitaciones de las RNA es el elevado costo computacional requerido durante el proceso de aprendizaje. Esto hace que el binomio tiempo-precisión sea una situación necesaria e indispensable de analizar antes de decidir qué modelo utilizar. En este sentido, la Tabla 9.4 muestra los minutos consumidos por cada uno de los SMC (formados por PMa, PMb y modular (Mod.)), durante los procesos de entrenamiento y clasificación.

Tabla 9.4. Tiempo de entrenamiento y clasificación

		CE orig.	7 Clasificadores				9 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PMa**	4,5	0.4	0.3	0.3	0.3	0.4	0.3	0.1	0.2
	PMb	5,2	1.1	0.4	1.4	1.5	0.3	0.4	0.3	0.6
	Mod.	11,3	9.4	9.2	6.3	6.7	9.5	9.2	2.6	5.0
Heart	**PMa**	2,6	0.2	0.2	0.2	0.1	0.2	0.2	0.1	0.1
	PMb	2,5	0.4	0.4	0.5	0.3	0.2	0.2	0.2	0.1
	Mod.	5,5	4.3	4.1	4.2	2.5	3.7	2.5	1.7	1.0
Liver	**PMa**	4,8	2.5	1.9	2.7	2.1	1.9	1.7	1.6	1.4
	PMb	1,6	1.9	1.8	1.9	1.8	1.7	1.6	1.5	1.5
	Mod.	3,3	1.5	1.3	2.0	1.4	1.7	1.7	1.5	1.5
Pima	**PMa**	8,5	9.5	9.6	8.9	4.8	5.7	6.8	5.2	3.0
	PMb	3,6	4.0	4.0	4.4	2.7	4.1	3.9	3.9	2.0
	Mod.	9,8	9.8	9.9	10.4	6.4	9.6	9.8	9.5	4.7
Sonar	**PMa**	1,2	1.7	0.6	0.5	0.4	0.5	0.5	0.4	0.3
	PMb	3,1	0.3	0.4	0.4	0.3	0.2	0.2	0.2	0.2
	Mod.	9,1	0.3	0.4	0.3	0.3	0.3	0.3	0.3	0.2
Iris	**PMa**	1,0	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
	PMb	0,4	0.1	0.1	0.0	0.0	0.1	0.1	0.04	0.03
	Mod.	2,3	2.3	2.2	1.1	0.2	1.8	1.8	0.5	1.1
Vehicle	**PMa**	60,5	37.9	33.7	36.1	18.5	36.3	32.4	23.2	12.0
	PMb	10,7	7.9	7.4	7.2	3.9	7.4	7.0	5.3	2.37
	Mod.	0,3	0.3	0.3	0.3	0.2	0.3	0.3	0.3	0.1
Wine	**PMa**	0,3	0.2	0.2	0.1	0.1	0.1	0.2	0.1	0.1
	PMb	0,1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.03
	Mod.	0,1	0.2	0.3	0.2	0.1	0.2	0.2	0.2	0.1
German	**PMa**	104,4	4.7	3.2	4.2	2.8	4.0	3.1	3.2	1.4
	PMb	13,3	10.0	8.9	9.3	6.0	9.2	8.3	8.7	3.9
	Mod.	31,4	21.6	22.0	22.3	13.7	25.2	22.9	21.0	8.5
Satimage	**PMa**	134,4	934.2	933.6	910.4	556.7	913.9	829.4	783.0	398.5
	PMb	134,5	90.0	82.6	87.6	52.7	88.3	82.4	76.4	38.6
	Mod.	593,2	498.3	434.8	420.2	238.2	420.6	409.0	349.9	169.0
Phoneme	**PMa**	60,5	39.2	42.4	39.2	37.4	39.4	41.9	40.1	31.4
	PMb	21,3	23.9	22.7	23.7	21.6	22.6	23.7	22.8	17.8
	Mod.	66,8	61.6	55.6	55.6	51.7	57.4	59.0	56.1	43.3
Waveform	**PMa**	437,9	246.6	197.7	217.7	176.5	167.3	131.2	172.8	82.5
	PMb	65,1	45.3	40.9	44.2	34.0	44.8	41.1	44.6	25.0
	Mod.	174,5	131.5	124.7	131.5	100.1	133.0	127.2	133.2	74.3
	PMa	210,9	95.8	78.6	76.5	53.3	81.8	63.1	60.3	32.3
Segment	**PMb**	35,0	28.2	27.3	22.8	14.4	24.4	23.0	20.4	10.8
	Mod.	1,4	2.0	2.1	1.8	1.2	2.1	2.1	1.8	0.6

Todos los experimentos se realizaron bajo las mismas condiciones, sobre una computadora portátil, con procesador Centrino a 1.30 GHz y 512 MB de capacidad en memoria RAM.

En los resultados de la Tabla 9.4, claramente podemos notar las grandes diferencias (en cuanto a tiempos de procesamiento), entre los tres esquemas de SMC y la utilización de un sólo clasificador. En la mayoría de los casos, estas diferencias son sumamente importantes, siendo en ocasiones diez veces mayor (ó más) el tiempo requerido cuando se utiliza una sola red. Por ejemplo, la base de datos Iris para su procesamiento con una red PMa requiere de un minuto, en tanto que al utilizar un SMC son suficientes diez segundos para realizar el entrenamiento y aprendizaje, incluso, con mejores índices de precisión.

En lo que respecta a los tiempos requeridos por cada uno de los sistemas, es posible notar que la utilización del SMC con PMb, en la mayoría de los casos, consume menos tiempo en el entrenamiento y la clasificación, inmediatamente seguido por el SMC formado por PMa. Además, si recordamos los resultados de clasificación obtenidos por las tres opciones, se puede concluir que, en general, la red con configuración PMb nos proporciona los mejores resultados en cuanto al balance entre tiempos y precisión.

9.4 SMC CON RNA Y SMC CON LA REGLA NN

Revisando los resultados obtenidos en las Tablas 9.1, 9.2 y 9.3, y comparándolos con los SMC que utilizan la regla NN (ver anexo 15), podemos hacer las siguientes observaciones:

a) SMC con NN vs. SMC con BPPa. Se observa un desempeño muy pobre del SMC con la regla NN pues, en general, sólo sobre las bases de datos Vehicle, Satimage y Phoneme logra superar los resultados obtenidos con el SMC que utiliza la red PMa. Por el contrario, este último mantiene un desempeño superior en Heart, Sonar, Wine, German, Waveform y Segment, respecto al obtenido con SMC formado por clasificadores NN.

b) SMC con NN vs. SMC con BPPb. A diferencia del inciso anterior, en estos resultados hay un comportamiento muy similar entre los dos SMC. De estos dos esquemas, el que utiliza la regla NN mejora en Vehicle, Satimage, Phoneme y Segment, en tanto que el segundo mejora en Heart, Sonar, Wine, German y Waveform. En el resto de las bases de datos, los dos esquemas mantienen índices de precisión muy similares. Uno mejora con algún método de submuestreo, mientras que otro mejora en otros métodos.

c) SMC con NN vs. SMC con redes modulares. En estos resultados, vemos que, en la mayoría de las bases de datos utilizadas, la regla NN tiene

mayores índices de precisión que los correspondientes al SMC formado por redes modulares. Estas mejores las encontramos en Cancer, Pima, Iris, Vehicle, Satimage, Phone, Liver, Waveform y Segment.

Aún cuando existen algunos casos en los que el SMC formado por clasificadores NN tiene menor desempeño que cualquiera de los SMC que utilizan algún método de RNA, no es un esquema despreciable, pues los costos computacionales asociados a la regla NN para los conjuntos de datos utilizados es inferior a los tiempos requeridos por cualquiera de los esquemas que utilizan redes.

9.5 CONCLUSIONES

En este capítulo, se han presentado los resultados sobre diversos experimentos realizados con SMC formados por redes de dos tipos: con estructura modular y con estructura múlticapa. En la topología de las redes múlticapa, se utilizaron dos criterios para determinar el número de neuronas en la capa oculta (siempre una única capa oculta): número de atributos más 1 y número de capas más 1. Por otra parte, el número de expertos en cada red modular, se estableció en 5 de manera constante.

De los dos diferentes esquemas que utilizan PM, encontramos el mejor desempeño con el SMC que utiliza la configuración PMb. Del mismo modo, este comportamiento se repite al compararlo con los resultados obtenidos con el SMC formado por redes modulares. Todo esto, junto con el hecho de tener unos menores requerimientos computacionales tanto en la etapa de aprendizaje como en la de clasificación, hacen que la red PMb se convierta en la mejor opción para grandes bases de datos.

También fue posible realizar un estudio comparativo entre SMC formados por RNA y SMC formados por clasificadores NN. En cuanto a tiempos de ejecución y coste computacional, vimos que la utilización de clasificadores NN resulta muy ventajosa, ya que tanto el coste computacional como el tiempo requerido durante el proceso de clasificación son significativamente inferiores al de cualquiera de los tres modelos de redes neuronales.

Con base a la evidencia de los resultados aquí mostrados y de los Anexos 14 y 15, es posible establecer como alternativas de solución la utilización de SMC con clasificadores NN y SMC formados con perceptrón múlticapa (con neuronas ocultas = número de capas ocultas +1).

Los experimentos de este capítulo deben entenderse como el principio de un estudio más amplio que debería llevarse a cabo en investigaciones futuras, dado que los resultados aquí obtenidos parecen ser bastante prometedores. Inicialmente,

para la estructura modular, se fijó el número de expertos en 5, mientras que para las redes múlticapa, la razón de aprendizaje se estableció en 0.9 y el momento en 0.7. Actualmente, estos parámetros, en la mayoría de las ocasiones, son adquiridos por prueba y error, por lo que hasta nuestros días no se cuenta con parámetros estandarizados para todos los casos. Por tanto, en trabajos futuros, se realizará la variación en algunos parámetros de entrenamiento y en la misma topología de las redes utilizadas para analizar su influencia sobre los resultados de clasificación.

PARTE III

CONCLUSIONES

Capítulo 10

Conclusiones finales y líneas abiertas

En la presente Tesis Doctoral, se desarrollaron metodologías que permiten abordar eficientemente algunos de los problemas que actualmente se presentan en la práctica del reconocimiento de patrones. Para este fin, las principales herramientas utilizadas son los SMC, los algoritmos de preprocesado y los algoritmos de clasificación RNA y la regla 1-NN.

Intrínsecamente, todas estas herramientas cuentan con problemas que pueden deteriorar los índices de precisión. Por un lado, los SMC implican la ejecución en paralelo de varios clasificadores pero, para que exista un adecuado desempeño, es necesario que estos clasificadores cumplan con dos condiciones básicas: ser diversos y ser lo "suficientemente" precisos. Por otro lado, los algoritmos de preprocesado, en especifico, la edición de Wilson y el algoritmo SSM, en su ejecución, no se tiene un control sobre la cantidad de patrones a eliminar o a seleccionar, situación por la cual, en muchas ocasiones, los conjunto de entrenamiento resultantes (editados o reducidos) pueden tener clases poco

representadas y, en casos extremos, algunas de estas clases pueden llegar, incluso, a desaparecer del conjunto.

Los métodos basados en RNA son ampliamente reconocidos por la eficacia en el proceso de clasificación; sin embargo, el coste computacional asociado al proceso de entrenamiento es sumamente elevado y poco se sabe sobre su funcionamiento exacto. Esta situación los hace prohibitivos al querer utilizarlos sobre conjuntos de datos que cuentan con gran tamaño y/o dimensionalidad elevada. Por último, la regla 1-NN es una de las reglas de decisión más ampliamente estudiadas y usadas en multitud de aplicaciones; sin embargo, el coste computacional requerido (en tiempo y espacio de memoria) la hace poco práctica para ser aplicada a problemas de la vida real con grandes volúmenes de datos.

Las metodologías propuestas a lo largo de esta tesis van encaminadas a utilizar SMC, aprovechando los aspectos positivos que cada una de estas herramientas puede tener y, mediante la utilización combinada de dos o más de ellas, disminuir y, en algunos casos, eliminar sus limitaciones, elevando de este modo los índices de precisión que hasta el momento se han observado cuando se utiliza un clasificador único.

10.1 PROBLEMAS RELACIONADOS CON LOS SMC

Como se mencionó en la Sección 4.1, es fundamental que un SMC sea suficientemente diverso para que proporcione un rendimiento adecuado. La experimentación realizada abordó la diversidad desde dos perspectivas: diversidad en la construcción de las submuestras que forman el SMC y diversidad en las decisiones individuales. Para la primera cuestión, se implementaron cinco métodos de submuestreo: Bagging, Boosting, Arc-x4, selección aleatoria sin reemplazo y selección secuencial.

En su implementación clásica, estos métodos seleccionan los patrones de forma indiscriminada entre clases y producen submuestras con tamaño similar al del CE original. Esta situación propicia situaciones poco deseables: desbalance entre clases, aumento del tamaño e incremento en los requerimientos computacionales para su posterior procesamiento. En este sentido, se propuso una modificación sobre la forma de implementar los métodos: en la selección de patrones, considerar la distribución existente entre las clases del CE original. Experimentalmente, se demostró que seleccionar patrones mediante esta variante, nos permite resolver la mayoría de los problemas que pueden producirse cuando se utilizan en su forma clásica, manteniendo y, en muchos casos, mejorando los índices de precisión.

Ante la interrogante de cuál es el número adecuado de clasificadores que se debe utilizar en la solución de un problema, no existe una respuesta que generalice

todos los casos posibles. Dependiendo del conjunto de datos, de la cantidad de patrones y del número de clases que integren el problema, será la decisión a tomar. Para esto, se realizó un estudio exhaustivo en el cual se construyeron SMC formados por 3, 5, 7, 9, 15 y 25 clasificadores, siguiendo el criterio de mantener la misma proporción de representatividad entre las clases. Con los resultados obtenidos, pudo observarse que aquellos conjuntos de datos con poca representatividad de patrones son los menos beneficiados al utilizar un número de clasificadores elevado, pero tampoco es recomendable la utilización de un número excesivamente pequeño de clasificadores, pues el costo computacional es más elevado y el índice de precisión poco significativo. Sin embargo, en general, construir SMC con 5, 7 ó 9 clasificadores es suficiente para mantener (en su mayoría) los índices de precisión por encima del uso de un solo clasificador.

En lo referente al método de submuestreo que, independientemente de la cantidad de clasificadores utilizado, mantiene un desempeño aceptable, las técnicas de selección de patrones sin reemplazo y Bagging fueron las que mejores resultados mostraron en la mayoría de los casos. También pudo verse la poca (y, en algunos casos, nula) utilidad que los métodos Boosting y Arc-x4 tienen cuando el número de clasificadores está por encima de 15.

Finalmente, se ha establecido que los índices de diversidad en las decisiones individuales están fuertemente relacionados con los índices de precisión, de tal manera que un elevado grado de diversidad, teóricamente, nos conduce a mayores índices de precisión. Para estudiar esta cuestión en los SMC, acudimos a medidas ya existentes en la literatura: Q-estadístico, coeficiente de correlación, medida de desacuerdo y una nueva medida de variabilidad como aportación personal.

El estudio realizado nos permitió identificar que los índices de diversidad se incrementan de forma paralela al aumento en el número de clasificadores. Por otro lado, los resultados experimentales nos permitieron demostrar que la condición “mayor diversidad implica mayor precisión” no siempre se cumple. Al contrario, encontramos que los mejores resultados no siempre se corresponden con las situaciones en las que se identifica un mayor grado de diversidad, y viceversa, que un bajo grado de diversidad no implica menor precisión, ya que en ambos casos se obtienen mejores resultados.

10.2 AUMENTO DE LA EFICIENCIA DE LA REGLA 1-NN

Como se ha visto, muchos de los problemas que deterioran y obstaculizan el proceso de clasificación con la regla 1-NN están asociados con el CE utilizado. De forma especifica, estos problemas se refieren al desbalance entre clases, al gran tamaño del CE, y a la presencia de patrones atípicos, redundantes y/o ruidosos. Para ello, se propusieron diversas metodologías que involucran métodos ya existentes de preprocesado, encaminadas ahora a dar solución a cada uno de estos

problemas mediante el uso combinado de SMC y, de este modo, aumentar la eficiencia del clasificador.

En términos de clasificación, cuando se trata de combatir los efectos negativos del desbalance, los criterios empleados en esta investigación fueron: mejorar la precisión general y aumentar el coeficiente kappa en el tratamiento de bases de datos de más de dos clases, y mejorar la media geométrica en el caso de bases de datos de dos clases. Por otro lado, cuando se trata de escalabilidad de algoritmos, el criterio consistió en lograr la escalabilidad con un nivel de precisión similar al obtenido si el CE se hubiera trabajado de forma tradicional, es decir, si no fuera de un tamaño tan inmenso y, por lo tanto, pudiera tratarse completamente en la memoria del equipo.

En el Capítulo 7, se demostró experimentalmente que las metodologías propuestas para combatir los problemas de desbalance han manifestado un mayor grado de precisión al momento de realizar la clasificación de nuevos patrones, con respecto a las técnicas que utilizan un solo clasificador, o aquellas que utilizan de forma aislada el algoritmo de clasificación y los algoritmos de limpieza y reducción aplicados sobre el CE original. También se logró comprobar que, mediante la utilización de un SMC, es posible realizar la escalabilidad de algoritmos cuando se trabaja con grandes bases de datos. El diseño del SMC permitió establecer las condiciones necesarias para lograr una adecuada ejecución de algoritmos de preprocesado que enriquecen el comportamiento de la regla 1-NN durante la etapa de clasificación.

Del conjunto de técnicas propuestas para realizar el balance y que utilizan un SMC, para el caso de base de datos desbalanceadas de dos clases, es más recomendable la técnica de selección aleatoria de patrones sin reemplazo pues, en la mayoría de los casos, resultan submuestras de las que menos patrones son eliminados con el algoritmo de limpieza, debido tal vez a la forma en que se realizó la selección de patrones (aleatoria sin reemplazo), controlando de esta manera la presencia de patrones redundantes. Este último aspecto no es considerado como móvil del comportamiento reflejado, pues la técnica Boosting es la que cuenta con la menor cantidad de patrones atípicos detectados y, sin embargo, no proporciona la mejor precisión. En este sentido, al analizar la verdadera influencia que la redundancia implícita de los métodos de selección con reemplazo tienen sobre el desempeño del SMC, se pudo observar que el grado de redundancia existente en las clases tiene muy poca influencia en la precisión, entre otras cosas, pudiendo estar dado por la compensación que se tiene al fusionar las decisiones individuales.

Por otro lado, cuando se sobre-entrena la clase minoritaria mediante la incorporación de patrones sintéticos (algoritmo SMOTE), se observó que los resultados con estos métodos tienen un índice de precisión superior al obtenido con el CE original, utilizando para ello menos del doble de los requerimientos de memoria empleados para el tratamiento del CE original. De entre todas las

metodologías propuestas con este enfoque, mostró con mejor desempeño la llamada Metodología 2, que incluye la incorporación de patrones sintéticos y, posteriormente, la ejecución de los algoritmos de preprocesado.

Un caso particular de estudio corresponde a la base de datos de mamografía (Ism). Ésta presenta un desbalance impresionante que dificulta considerablemente la identificación correcta de la clase minoritaria. El comportamiento observado en términos de precisión es realmente favorable cuando se utiliza la Metodología 2 (incluye la incorporación de patrones sintéticos y, posteriormente, realiza la edición de Wilson) respecto al obtenido cuando se utiliza la edición de Wilson en el CE original. También pudo verse que, al ejecutar el algoritmo de limpieza, son muy pocos los patrones sintéticos eliminados y, por el contrario, sí es considerable la cantidad de patrones de la clase mayoritaria que se eliminan y no fueron detectados al momento de limpiar el CE original, aspecto que favorece significativamente los resultados mostrados anteriormente.

Considerando estos resultados, una última propuesta de solución consistió en la implementación de métodos que favorecen a la clase minoritaria y que utilizan la ponderación de distancias al momento de realizar la clasificación y/o la edición de Wilson. A partir de estos resultados, pudimos ver que la clasificación con distancia ponderada resulta ser una adecuada alternativa de solución para CE sin preprocesado y para cuando se utiliza la edición con distancia Euclídea. Por otra parte, cuando se realiza la edición con distancia ponderada, la mejor opción es clasificar con la distancia Euclídea.

Concretamente, para el tratamiento de CE que no tienen un desbalance muy severo, se propone la implementación de la edición y la clasificación con distancia ponderada, así como la utilización de un SMC que incluya la edición de Wilson (Metodología 2), con selección de patrones de forma aleatoria sin reemplazo en clases mayoritarias. Esta propuesta permite un balance beneficioso entre las clases presentes en el CE, sin tener que recurrir a despreciar información potencialmente útil (como seria la eliminación de patrones de la clase mayoritaria), ni a la duplicación de patrones de la clase minoritaria, que encarece el coste computacional sin aportar nueva información útil. Por otro lado, cuando el desbalance es muy grande, como el caso de Ism, lo más recomendable es realizar el sobre-entrenamiento de la clase menos representada. En esta situación, limpiar y reducir el tamaño del conjunto de datos, después de incorporar los patrones sintéticos, siempre resulta favorable, ya que disminuye los requerimientos computacionales de la regla NN e incrementa su efectividad.

Se trabajó también con conjuntos de más de dos clases que presentan desbalance. Adicional al desbalance existente en estos casos, se distingue un problema no trivial: determinar qué clases deben ser consideradas como minoritarias y cuáles clases deben tratarse como mayoritarias. Para esto, se propuso establecer un umbral que hiciese división, considerando el promedio de patrones

que *debería* tener cada clase. Se estudió la eficiencia de la propuesta, en comparación con la realizada por otros trabajos [Gar., 02]. Los resultados experimentales demostraron, de forma contundente la superioridad (en términos de precisión) de la propuesta efectuada en esta Tesis Doctoral.

Otro punto importante del presente trabajo es la investigación realizada sobre la utilidad de un SMC cuando se manipulan CE cuyo tamaño no permite emplear algoritmos de limpieza de los datos. Para realizar la escalabilidad de estos algoritmos de forma eficiente, se han explorado diversas técnicas para la construcción de un SMC. Los resultados experimentales que se han reportado en este trabajo muestran que el empleo de un SMC permite la escalabilidad de estos algoritmos, con un comportamiento similar, tanto en precisión de clasificación como en el tamaño de las submuestras resultantes, al que se hubiera obtenido trabajando en la forma tradicional.

10.3 FUSIÓN DE CLASIFICADORES

La fusión de clasificadores es, en la actualidad, el método de combinación más ampliamente usado. Sin embargo, este método tiene serios problemas cuando los componentes que forman el SMC no son igualmente precisos [Mat., 96]. Por tal motivo, se implementaron nuevas formas de fusionar las decisiones individuales en un SMC. En concreto, ponderando dinámica y estáticamente las decisiones individuales. Para los métodos de ponderación estática, se asignaron los pesos a los clasificadores de acuerdo a la estimación del error realizada por el método leaving-one-out. Por su parte, para los métodos de ponderación dinámica, se utilizó la menor distancia proporcionada por el vecino más próximo encontrado entre los clasificadores que componen el SMC.

En el Capítulo 6, se describen los métodos propuestos: seis de ponderación dinámica y uno de ponderación estática de clasificadores. Los diferentes experimentos realizados nos permitieron demostrar la superioridad (en términos de precisión) de los métodos de ponderación dinámica sobre la utilización de un clasificador único, la votación ponderada estática y la votación simple. Esta situación sobresale más cuando se utilizan SMC con 5 y 9 clasificadores.

Por otro lado, de los métodos de ponderación dinámica, los que tienen un desempeño más uniforme son la votación por promedio de distancias y la ponderación con distancia inversa. Por último, los métodos más adecuados para construir el SMC son principalmente la selección secuencial (para aquellos conjuntos de datos que lo permiten), la selección aleatoria sin reemplazo y, como opciones alternativas, Bagging y Arc-x4. Esto último resulta ser muy conveniente si consideramos que tanto los métodos de ponderación como los de selección de patrones requieren procesos sumamente simples para su ejecución y representan un bajo coste computacional cuando se trabaja con la regla NN.

10.4 SMC CON REDES

La experimentación realizada bajo este esquema no es más que el principio de una investigación más amplia que debería llevarse a cabo como líneas de trabajo futuras. En esta fase inicial, se implementaron SMC formados por dos tipos de redes: múlticapa y modular.

Para el SMC que utiliza la combinación de varios perceptrones, la topología constó de tres capas: una de entrada, una oculta y una de salida. Basándonos en propuestas previas, se establecieron como parámetros fijos de entrenamiento, la razón de aprendizaje en 0.9 y el momento en 0.7. Así mismo, se utilizaron en la capa de entrada tantas neuronas como atributos tiene el patrón de entrada, y en la capa de salida, tantas neuronas como clases se dispongan en el CE. Por último, para determinar la cantidad de neuronas en la capa oculta, se consideraron dos criterios: la cantidad de atributos en el patrón más 1, y la cantidad de capas ocultas más 1. Por otro lado, en lo que respecta a la estructura de la red modular, se estableció la utilización de cinco redes feedforward expertas y una integradora.

Los resultados experimentales nos permitieron validar el desempeño de estos tres modelos, así como también el tiempo requerido para su entrenamiento y la clasificación de nuevos casos. En los resultados, encontramos que la utilización de un esquema simple para la capa oculta ofrece los mejores índices de precisión, con un costo asociado inferior al requerido por los otros esquemas, y tan solo un poco superior al necesario cuando se utiliza una sola red.

Un último análisis consistió en el estudio del desempeño que estos modelos y el utilizado con la regla 1-NN tienen al momento de clasificar nuevos patrones. En este punto, pudimos constatar experimentalmente que, pese a las múltiples desventajas que suelen asociarse a la regla 1-NN, los índices de precisión obtenidos sobre trece conjuntos de datos, en promedio, con siete de ellos, se mantuvo por encima de cualquier esquema que utiliza la combinación de redes. Entre estos casos están las bases de datos Vehicle, Phoneme y Satimage, conjuntos que cuentan en su interior con un elevado grado de desbalance, solape entre clases y gran tamaño, situaciones que dificultan su tratamiento adecuadamente y que, sin embargo, logran superarse con el SMC que utiliza clasificadores 1-NN.

10.5 PRINCIPALES APORTACIONES

En resumen, las principales aportaciones que pueden distinguirse en este trabajo son de tres tipos: las que están relacionadas directamente con las bases de datos, las relacionadas con el SMC y las relacionadas con el algoritmo de clasificación.

En lo relativo a las aportaciones relacionadas con la base de datos, debemos destacar el adecuado tratamiento de bases de datos desbalanceadas de dos clases,

mediante la obtención de un balance perfecto utilizando métodos propios de los SMC, la adaptación de algoritmos de preprocesado para su ejecución en forma conjunta con el algoritmo SMOTE, la implementación de una nueva forma de tratar las bases de datos desbalanceadas de más de dos clases, la implementación de un algoritmo que permite hacer una limpieza de la base de datos mediante la eliminación de patrones redundantes, y la realización exitosa de la escalabilidad de algoritmos con la ayuda de métodos de SMC y algoritmos de preprocesado.

En segundo lugar, en cuanto a las aportaciones relacionadas con el SMC, podemos mencionar, primeramente, las que se refieren a la fusión de clasificadores, que incluye la implementación de seis nuevos métodos de ponderación dinámica y uno para ponderación estática de las decisiones individuales de un SMC. Por otra parte, las conclusiones provenientes del análisis de diversidad, nos permitieron demostrar que la hipótesis “mayor diversidad implica mayor precisión” no es 100% verdadera, ni se cumple en la totalidad de los casos. Por último, demostramos que la implementación de los métodos de submuestreo tiene igual o mejor desempeño cuando se tiene en mente la representatividad entre clases que al aplicarlos en su configuración clásica.

Finalmente, en lo relacionado al algoritmo de clasificación, para los experimentos, utilizamos la regla 1-NN y tres modelos de RNA. Los resultados obtenidos nos permiten demostrar la viabilidad de utilizar en un SMC la regla 1-NN y la estructura múlticapa con poca cantidad de neuronas ocultas, como alternativas de solución a problemas prácticos de reconocimiento de patrones.

10.6 LÍNEAS ABIERTAS

El área de SMC es muy amplia y, en consecuencia, todavía existen espacios poco explorados. En esta Tesis Doctoral, tratamos de cubrir algunos de los aspectos más importantes a considerar cuando se trabaja con SMC: diversidad en las decisiones, evaluación de los clasificadores individuales, fusión de las decisiones y mejoras sobre los CE utilizados.

Al alcanzar este punto en la investigación desarrollada, el cual no debe considerarse como un punto final, hemos podido constatar la necesidad de profundizar los conocimientos sobre SMC en otros muchos temas que no han tenido cabida en esta Tesis, como la Selección de Clasificadores. En esta línea, se buscan métodos que ayuden a determinar el conocimiento verdadero de cada uno de los clasificadores sobre el espacio de representación, para así poder establecer responsables o expertos para cada caso que se estudie. Dentro de este apartado, una de las posibilidades más interesantes puede consistir en la utilización de medidas de complejidad de los datos para establecer la conveniencia de aplicar un determinado clasificador sobre cada región del espacio de representación. En cierta manera, se trataría de determinar el mejor clasificador (experto) para cada región

en función de las características (de complejidad) de los datos (separabilidad, densidad, solapamiento, etc.)

También se reportaron en está Tesis los primeros resultados de SMC basados en redes. El elevado costo computacional requerido por estos modelos nos permite ver la poca conveniencia de su utilidad como componentes del SMC. Sin embargo, nos sugiere la posibilidad de su uso para establecer redes expertas en un SMC, en particular cuando se utilizan redes modulares. En este sentido, los trabajos futuros en esta línea estarán dirigidos hacia la utilización de redes modulares como alternativa para la selección de clasificadores. Así mismo, otra alternativa de solución para establecer las regiones en la selección de clasificadores y para lograr diversidad en los componentes individuales, consistirá en la utilización de principios de algoritmos genéticos.

Otra línea abierta de investigación, esta relacionada con la incorporación de herramientas de aprendizaje no supervisado como alternativas para lograr diversidad en los componentes individuales del SMC. Finalmente, al ver los resultados reportados en el Capítulo 7, es posible pensar en ampliar la utilización del algoritmo de edición de Wilson a los experimentos realizados en el resto del documento.

10.7 PUBLICACIONES RESULTANTES

De las diferentes aportaciones y resultados obtenidos a partir de la investigación realizada en la presente Tesis Doctoral, se han originado diversas publicaciones, tanto en forma de artículos en revista internacionales como de comunicaciones en congresos. A continuación enumeramos las publicaciones generadas hasta el momento de la edición de esta Tesis. Sin embargo, es previsible que todavía puedan producirse otras que recojan partes aún no publicadas de esta investigación.

10.7.1 PUBLICACIONES EN REVISTAS INTERNACIONALES

a) R. Barandela, J. S. Sánchez, R. M. Valdovinos: “New applications of ensembles of classifiers”, *Pattern Analysis and Applications* 6, 2003, pp. 245 – 256.

b) R. M. Valdovinos, J. S. Sánchez, “On the relation between diversity and accuracy in ensembles of nearest neighbour classifiers”, *Pattern Recognition Letters*, 2006 (enviado, en revisión).

c) R. M. Valdovinos, J. S. Sánchez, “Ensembles of multilayer perceptron and modular neural networks for fast and accurate learning”, *Neurocomputing*, 2006 (enviado, en revisión).

10.7.2 PUBLICACIONES EN CONGRESOS INTERNACIONALES Y OTROS

a) R. M. Valdovinos, R. Barandela: "Sistema de múltiples clasificadores, una alternativa para la escalabilidad de algoritmos", in *Proceedings of 9th Internacional Conference of Research on Computer Science* (*CIICC02)*, J. Ortiz, J. Pérez (Eds), Puebla (México), ISBN 970-18-8526-0, 2002.

b) R. Barandela, R. M. Valdovinos, J. S. Sánchez, F. J. Ferri: "The imbalanced training sample problem: Under or over sampling?", in *Structural, Syntactic, and Statistical Pattern Recognition*, *Lecture Notes in Computer Science*, Vol. 3138, A. Fred, T. Caelli, R.P.W. Duin, A. Campilho and D. de Ridder (Eds.), Springer-Verlag, ISBN 3-540-22570-6, pp. 806-814, 2004.

c) R. M. Valdovinos, J. S. Sánchez, R. Barandela: "Dynamic and static weighting in classifier fusion", in *Pattern Recognition and Image Analysis, Lecture Notes in Computer Science*, Vol. 3523, J.S. Marques, N. Pérez de la Blanca and P. Pina (Eds.), Springer-Verlag, ISBN 3-540-26154-0, pp. 59-66, 2005.

d) R. M. Valdovinos, J. S. Sánchez: "Class-dependant resampling for medical applications", in *Proceedings of 4th International Conference on Machine Learning and Applications, ICMLA'05*, ISBN 0-7695-2495-8, pp. 351-356, Los Angeles (EE.UU.), Diciembre 2005.

e) R. M. Valdovinos, J. S. Sánchez: "Combining multiple classifiers with dynamic weighted voting", in *Proceedings of International Workshop on Intelligent Computing in Pattern Analysis/Synthesis*, Xi'An (China), 2006 (aceptado, pendiente de publicación).

f) R. M. Valdovinos, J. S. Sánchez "Comparison of dynamic and static weighting functions for classifier fusion", in *Pattern Recognition: Progress, Directions and Applications*, F. Pla, P. Radeva, J. Vitrià (eds.), Publicaciones de la Universitat Autónoma de Barcelona, ISBN 84-933652-6-2, pp. 352-361, 2006.

g) R. M. Valdovinos, J. S. Sánchez "Performance comparison of neural network ensembles and nearest neighbor classifier ensembles", in *Proceedings of 11th Iberoamerican Congress on Pattern Recognition*, Cancun (México), 2006 (enviado, en revisión).

APÉNDICE I

REDES NEURONALES ARTIFICIALES

Anexo 1

Modelos de RNA utilizados

En este anexo, se describen el algoritmo de backpropagation (propagación del error hacia atrás) utilizado en el entrenamiento del perceptrón multicapa, y el algoritmo de aprendizaje estocástico que emplea la red modular.

A1.1 RED NEURONAL BACKPROPAGATION

En este tipo de arquitectura, todas las conexiones de la red son de tipo adaptativo, las conexiones entre neuronas son siempre hacia delante, es decir, las conexiones van desde las neuronas de una determinada capa hacia las neuronas de la siguiente capa; no hay conexiones laterales, ni conexiones hacia atrás, es decir, conexiones que van desde una capa hacia la capa anterior. Por tanto, la información siempre se transmite desde la capa de entrada hacia la capa de salida.

La Figura A.1 muestra gráficamente el modelo correspondiente a la red múltiplicapa utilizada en los experimentos.

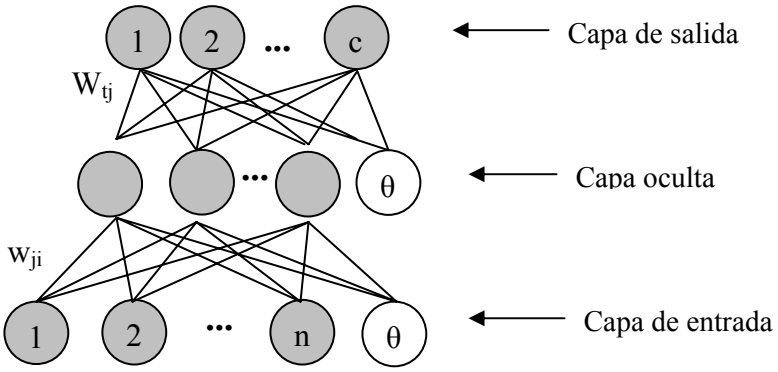


Figura A.1. Perceptrón multicapa

Como puede verse, la red implementada consta de una sola capa oculta con (n + 2) neuronas, la capa de entrada consta de (n + 1) neurona y la capa de salida está formada por c neuronas. Para los experimentos reportados en esta Tesis Doctoral, n denota la cantidad de atributos de los patrones y c representa el número de clases. θ se considera como una neurona más que tiene la función de umbral, con un peso asociado igual a 1. Finalmente, w_{ii} corresponde al peso de conexión entre la neurona de entrada i y la neurona oculta j, y w_{tj} es el peso de conexión entre la neurona oculta j y la neurona de salida c.

A1.1.1 PROPAGACIÓN HACIA ADELANTE

Cuando se presenta un patrón x de n características como entrada a la red, éste se transmite a través de los pesos w_{ji} desde la capa de entrada hacia la capa oculta. Las neuronas de la capa intermedia transforman las señales recibidas mediante la aplicación de una función de activación, proporcionando de este modo, un valor de salida o "net". La entrada total o neta que recibe una neurona oculta j, net_j, es:

$$net_j = \sum_{i=1}^{n} w_{ji} * x_i + \theta_i$$

donde x_1, x_2,…, x_n son las señales de entrada; w_1, w_2,…, w_n son los pesos sinápticos de la neurona j.

Este se transmite a través de los pesos w_{tj} hacia la capa de salida, donde aplicando la misma operación que en el caso anterior, las neuronas de esta última capa proporcionan la salida de la red. El valor de salida de la neurona oculta j, Z_j, se obtiene aplicando una función de activación f(.) sobre su entrada neta:

$$Z_j = f(net_j)$$

donde *f(net)* es la función sigmoidal: $f(net_j) = \dfrac{1}{1 + \exp\left[-(net + \theta)\right]}$

De igual forma, la entrada neta que recibe una neurona de salida t, net_t, esta dada por:

$$net_t = \sum_{j=1}^{c} w_{tj} * Z_j + \theta_j$$

Por último, el valor de salida de la neurona de salida t, Z_t, es:

$$Z_t = f(net_t)$$

A1.1.2 PROPAGACIÓN HACIA ATRÁS

La arquitectura backpropagation es de tipo feed-forward. Esta característica implica que, a partir de la recepción de datos en la primera capa, se van procesando las salidas de las neuronas de las capas consecutivas, hasta llegar a la última capa. Como objetivo principal, el algoritmo busca minimizar el error cuadrático medio (ECM), el cual cuantifica la diferencia entre la salida obtenida por la red y la salida deseada ante la presentación de un conjunto de patrones de entrenamiento. Esta minimización, se lleva a cabo mediante un procedimiento de gradiente descendente

donde se requiere el cálculo del gradiente del ECM respecto a los pesos de los enlaces. La función de error a minimizar para cada patrón x está dada por:

$$E_x = \frac{1}{2}\sum_{t=1}^{c}(s_t - Z_t)^2$$

donde z_i es la salida deseada para la neurona de salida t ante la presentación del patrón x, y está determinada por 1 si x Є c, ó en caso contrario 0. A partir de esta expresión, se puede obtener una medida general del error mediante:

$$E_{prom} = \frac{1}{m}\sum_{x=1}^{m} E_x$$

Para realizar los ajustes en los pesos, el algoritmo *backpropagation* utiliza el gradiente descendente [Rum., 86]. El gradiente toma la dirección que determina el incremento más rápido en el error, mientras que la dirección opuesta (negativa) determina el decremento más rápido en el error. Teniendo en cuenta que E_x es función de todos los pesos de la red, el gradiente de E_x es un vector igual a la derivada parcial de E_x respecto a cada uno de los pesos. Por tanto, el error puede reducirse ajustando cada peso en la dirección:

$$\Delta w_{ji} = -\eta \frac{\partial E_x}{\partial w_{ji}}$$

donde η es la tasa de aprendizaje. El valor de η tiene un papel crucial en el proceso de entrenamiento de la red, ya que controla el tamaño del cambio de los pesos en cada iteración. De este modo, el ajuste de los pesos está dado por:

$$\Delta w(I+1) = \eta * \delta * Z$$

donde *I* indica la iteración, *Z* la salida de una neurona y δ es el gradiente local, definido por:

$$\delta = \begin{cases} (s_t - Z_t) f'(net_t) & \text{para las neuronas de la capa de salida} \\ f'(net_j) Z_i \sum_t \delta_t w_{t,j} & \text{para las neuronas de la capa oculta se aplica la regla delta.} \end{cases}$$

Se puede observar que el error o valor delta asociado a una neurona de la capa oculta j está determinado por la suma de los errores que se cometen en las t neuronas de salida que reciben como entrada la salida de esa neurona oculta j. De ahí que el algoritmo también se denomine propagación del error hacia atrás.

Con el fin de acelerar el proceso de convergencia de los pesos, Rumelhart et al [Rum., 86] sugirieron incluir, en la función para incrementar los pesos, un factor momento α, el cual tiene en cuenta la dirección del incremento tomada en la iteración anterior. Este factor permite filtrar las oscilaciones en la superficie del error provocadas por la razón de aprendizaje y acelera la convergencia de los pesos. Así, el ajuste de los pesos está dado por:

$$\Delta w(I+1) = \eta * \delta * Z + \alpha * \Delta w(I)$$

A1.2 RED NEURONAL MODULAR

Este tipo de arquitectura utiliza la estructura feedforward, con la propagación de la información hacia delante, es decir, desde las neuronas de la capa de entrada hacia las neuronas de la capa de salida.

Las redes neuronales que utilizan esta estructura, cuentan con una serie de módulos "expertos" y un módulo integrador. Durante el proceso de aprendizaje, cada patrón de entrada es conectado a todos los módulos. Posteriormente, las salidas de todos los módulos expertos son conectadas al módulo integrador (Figura A.2).

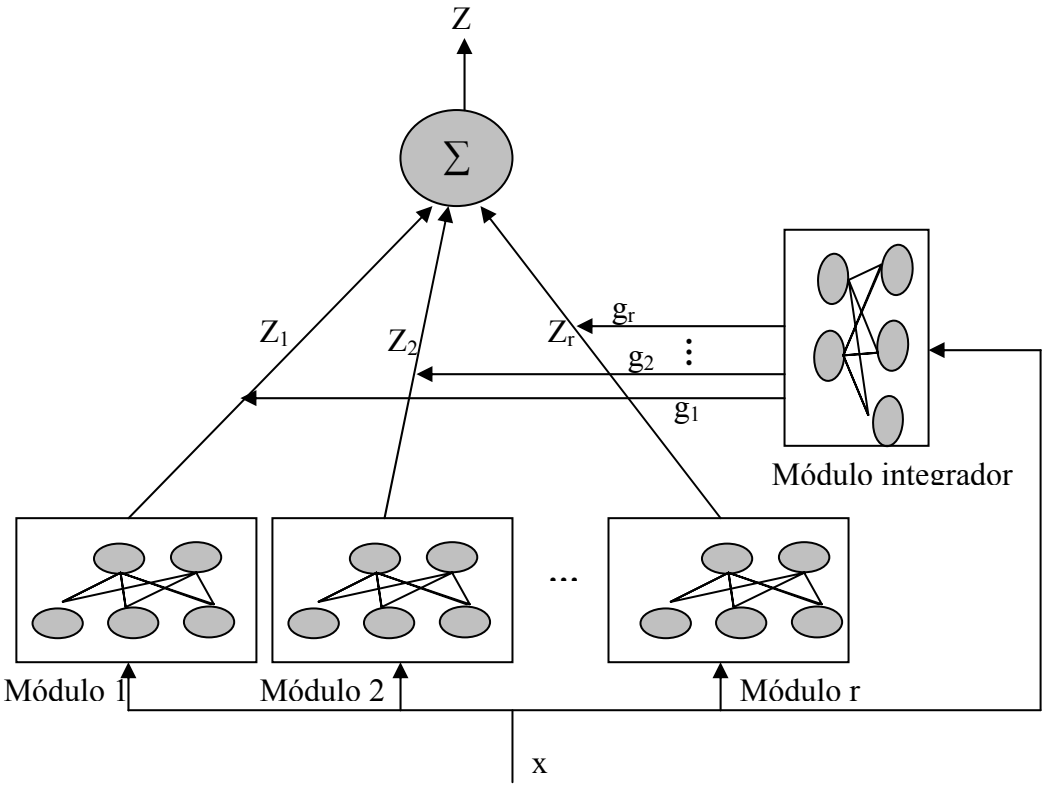


Figura A.2. Red Neuronal Modular

De acuerdo al esquema de la red modular presentado en la Figura A.2, se tienen r módulos con un perceptrón lineal cada uno, y una red integradora (también un perceptrón lineal) llamada "gating network". Para su creación, las especificaciones necesarias son las siguientes:

a) Número de entradas por módulo = n, número de variables del patrón
b) Número de módulos de entrada = r, tantos como se desee.
c) Número de salidas para los módulos = c, número de clases existentes en el problema.
d) Número de salidas para la red integradora = r, tantos módulos haya.

En el proceso de aprendizaje, hace uso del gradiente estocástico y tiene como función objetivo:

$$-\ln\left(\sum_{i=1}^{r} g_i * \exp\left(-\frac{1}{2}\|s - Z_i\|^2\right)\right)$$

donde s es la salida deseada para la entrada x, y Z_i es el vector de salidas de la i-esima red experta.

Ante un patrón x n-dimensional como entrada, el proceso general de aprendizaje de la red modular considera los siguientes pasos:

1. Inicialización aleatoria de los pesos sinápticos a las diferentes redes (expertas y de integración), con valores pequeños uniformemente distribuidos. En adelante, consideraremos w_{ji} como los pesos de la red experta, y w_{ti} como los pesos de la red integradora.
2. El patrón x es presentado a todas y cada una de las redes (expertas e integradora), de tal forma que la salida de cada la red experta está dada por:

$$Z_i^m = x * w_{ji}^m$$

donde x es el vector de entrada, y el superíndice *m* es indicativo de "módulo".
De la misma manera, la salida de la red integradora se obtiene mediante:

$$g_i = \frac{\exp(u_i)}{\sum_{j=1}^{r} \exp(u_j)}$$

donde $u_i = x * w_{ti}$

3. Para realizar el ajuste de pesos, se toman en cuenta dos criterios: el ajuste de los pesos de las redes expertas y el ajuste de los pesos en la red integradora.

    a. Para modificar los pesos de las diferentes redes expertas, se utiliza la siguiente función.

$$w_{ji}^{m}(I+1) = w_{i}^{m}(I) + \eta * h_i(s - Z_i^{m})x$$

    b. Por otra parte, la modificación de los pesos de la red integradora se realiza mediante:

$$w_{ti}(I+1) = w_{ti}(I) + \eta(h_i(I) - g_i(I))x$$

    donde

$$h_i = \frac{g * \exp\left(-\frac{1}{2}\left\|s - Z_i^{m}\right\|^2\right)}{\sum_{j=1}^{r} g * \exp\left(-\frac{1}{2}\left\|s - Z_i^{m}\right\|^2\right)}$$

4. Finalmente, la red integradora decide cómo serán combinadas las salidas de los módulos para formar la salida final de la red modular, mediante:

$$Z = \sum_{i=1}^{r} g_i * Z_i$$

APÉNDICE II

DIVERSIDAD

Anexo 2

Clasificación con SMC de 5 clasificadores formados con submuestreo clásico

		CE Original	V. simple			V. por promedio			V. Shepard modificado		
			A	B	C	A	B	C	A	B	C
Cancer	**PG**	95,6	95,5	95,6	95,6	94,2	95,5	94,2	95,8	95,8	95,8
	DE	2,5	2,8	2,5	2,3	3,5	2,4	3,1	2,2	2,5	2,6
Heart	**PG**	58,2	59,3	56,7	57,0	59,6	56,7	57,8	58,2	58,5	58,2
	DE	6,2	6,7	6,6	7,7	6,1	6,4	7,2	6,2	6,9	6,2
Liver	**PG**	65,2	62,6	63,8	63,2	62,6	64,4	64,5	65,5	65,2	65,5
	DE	4,8	6,6	5,3	5,2	6,4	4,8	5,4	5,0	4,9	5,4
Pima	**PG**	65,9	66,7	65,5	65,4	66,7	65,5	65,4	65,8	65,6	65,8
	DE	5,2	4,5	4,8	5,8	4,9	4,9	5,8	4,9	5,1	5,2
Sonar	**PG**	82,0	80,0	82,4	81,5	80,5	82,4	82,0	82,0	83,9	82,0
	DE	9,4	8,7	5,6	8,9	8,1	5,6	8,2	9,4	9,2	9,4
Glass	**PG**	70,0	70,5	68,5	68,0	70,5	68,5	69,0	70,1	69,0	70,0
	DE	5,3	5.0	4,9	4,1	5,4	4,9	5,2	5,1	5,2	5,0
Iris	**PG**	96,0	96,0	94,0	96,0	96,0	94,7	96,0	96,0	94,7	96,0
	DE	1,5	1,5	4,4	1,5	1,5	4,5	1,5	1,5	4,5	1,5
Vehicle	**PG**	64,2	63,4	64,4	63,9	63,8	64,6	63,5	64,1	64,1	64,4
	DE	1,8	2,5	2,1	1,3	2,5	2,1	1,0	1,7	2,0	1,7
Wine	**PG**	72,4	72,4	72,4	71,8	73,5	72,9	71,8	71,8	72,4	72,4
	DE	3,4	2,6	3,4	5,3	2,9	3,2	5,3	3,4	3,4	3,4
German	**PG**	65,2	66,4	64,6	65,3	66,3	64,7	65,2	64,8	64,6	65,0
	DE	2,6	4,9	2,8	2,0	5,0	2,8	2,0	2,6	2,8	2,5
Satimage	**PG**	83,6	83,6	83,1	83,6	83,7	83,1	83,6	83,6	83,5	83,6
	DE	11,6	12,1	11,4	11,8	12,1	11,4	11,8	11,7	11,6	11,6
Phoneme	**PG**	76,1	76,0	75,8	75,8	76,0	75,9	75,9	75,8	6,0	76,0
	DE	8,4	8,9	7,8	8,5	8,8	7,9	8,5	8,3	8,0	8,2
Waveform	**PG**	78,0	78,1	77,0	77,9	78,1	77,0	77,9	78,0	7,7	78,0
	DE	2,9	3,1	2,5	3,0	3,1	2,5	3,0	2,9	2,8	2,8
Segment	**PG**	94,8	94,1	94,9	94,8	94,4	94,9	94,5	94,7	4,9	94,8
	DE	1,4	1,2	0,9	1,3	1,3	0,9	1,6	1,6	1,3	1,4

A = Bagging
B = Boosting
C = Arc-x4

Anexo 3

Clasificación con SMC de 7 clasificadores formados con submuestreo clásico

		CE Original	V. simple			V. por promedio			V. Shepard modificado		
			A	B	C	A	B	C	A	B	C
Cancer	**PG**	95,6	95,8	95,0	95,6	94,5	95,0	94,3	95,6	95,6	95,6
	DE	2,5	2,6	3,0	2,5	3,5	3,0	3,4	2,5	2,7	2,5
Heart	**PG**	58,2	60,4	55,2	56,3	60,0	55,2	57,8	58,2	58,2	58,2
	DE	6,2	5,3	7,3	7,4	5,5	7,3	7,2	6,2	6,2	6,2
Liver	**PG**	65,2	66,1	60,6	61,7	65,8	61,5	62,6	65,2	64,4	64,6
	DE	4,8	6,3	3,9	5,9	5,9	4,2	6,6	4,8	3,3	3,8
Pima	**PG**	65,9	67,5	65,2	66,0	67,6	65,2	66,5	65,9	65,9	65,9
	DE	5,2	5,8	6,1	5,3	5,7	6,6	5,5	5,2	5,2	5,2
Sonar	**PG**	82,0	81,0	81,0	80,5	81,0	81,5	81,0	82,0	81,5	82,0
	DE	9,4	8,3	7,6	10,5	8,3	7,6	9,7	9,4	8,9	9,4
Glass	**PG**	70,0	68,5	67,5	69,0	68,5	68,5	69,5	70,0	70,0	70,5
	DE	5,3	5,5	4,0	4,9	5,5	6,0	5,4	5,3	5,3	5,4
Iris	**PG**	96,0	96,0	94,7	94,7	96,0	95,3	96,0	96,0	96,0	96,0
	DE	1,5	1,5	3,0	3,0	1,5	3,0	1,5	1,5	1,5	1,5
Vehicle	**PG**	64,2	65,4	63,9	64,2	65,7	64,4	64,1	64,2	64,4	64,2
	DE	1,8	2,1	2,2	1,5	2,3	2,4	1,6	1,8	1,5	1,8
Wine	**PG**	72,4	72,9	71,8	72,4	72,9	71,2	72,4	72,4	72,4	72,4
	DE	3,4	4,8	3,0	3,4	3,2	4,8	3,4	3,4	3,4	3,4
German	**PG**	65,2	66,0	64,9	64,6	65,3	64,8	64,5	65,4	65,0	65,0
	DE	2,6	3,3	3,0	2,1	3,9	3,3	2,2	2,6	2,7	2,7
Satimage	**PG**	83,6	83,6	83,0	83,5	83,6	83,0	83,5	83,6	83,5	83,6
	DE	11,6	12,0	11,0	11,6	12,0	11,0	11,6	11,6	11,6	11,6
Phoneme	**PG**	76,1	76,5	75,3	76,1	76,4	75,5	75,9	76,0	75,6	76,0
	DE	8,4	8,1	7,7	7,7	8,3	8,1	7,9	8,1	8,2	8,0
Waveform	**PG**	78,0	78,6	76,5	78,5	78,6	76,5	78,4	77,9	7,7	78,0
	DE	2,9	3,0	2,7	2,3	3,0	2,8	2,3	2,9	2,9	2,9
Segment	**PG**	94,8	94,3	94,6	94,4	94,9	95,0	94,6	94,8	4,9	94,8
	DE	1,4	1,5	1,3	1,5	1,6	1,1	1,4	1,4	1,4	1,4

A = Bagging
B = Boosting
C = Arc-x4

Anexo 4

Clasificación con SMC con 9 clasificadores formados con submuestreo clásico

		CE Original	V. simple			V. por promedio			V. Shepard modificado		
			A	B	C	A	B	C	A	B	C
Cancer	**PG**	95,6	95,5	94,9	95,8	94,3	95,0	94,5	95,8	95,8	95,6
	DE	2,5	2,7	2,1	2,5	3,2	2,2	3,5	2,6	2,6	2,5
Heart	**PG**	58,2	58,9	58,9	56,7	58,2	59,3	57,8	58,2	57,8	58,2
	DE	6,2	7,5	5,8	5,7	7,4	4,5	6,2	6,2	6,3	6,2
Liver	**PG**	65,2	65,8	62,0	62,3	65,8	62,3	61,5	64,9	64,9	64,4
	DE	4,8	4,7	5,2	7,0	4,9	4,7	5,6	4,7	5,0	4,5
Pima	**PG**	65,9	66,9	64,1	64,2	67,1	64,7	64,3	65,9	65,9	65,9
	DE	5,2	4,6	4,7	4,9	4,8	5,2	5,3	5,2	5,2	5,4
Sonar	**PG**	82,0	81,0	80,5	82,4	81,0	81,0	82,9	82,0	81,5	82,0
	DE	9,4	9,8	7,1	6,5	9,8	6,3	6,9	9,4	9,2	9,4
Glass	**PG**	70,0	69,0	68,5	69,5	69,5	67,5	70,0	70,0	69,5	70,0
	DE	5,3	4,5	7,2	5,7	4,8	5,9	5,3	5,3	5,7	5,3
Iris	**PG**	96,0	96,0	96,0	96,0	96,0	96,0	96,0	96,0	96,0	96,0
	DE	1,5	1,5	1,5	1,5	1,5	1,5	1,5	1,5	1,5	1,5
Vehicle	**PG**	64,2	65,0	63,5	63,8	65,2	63,5	63,8	64,2	64,4	64,2
	DE	1,8	2,3	1,2	1,9	2,0	1,6	2,3	1,8	1,9	1,8
Wine	**PG**	72,4	70,6	71,8	71,2	71,8	71,8	71,2	72,4	72,4	72,4
	DE	3,4	3,6	4,5	5,7	4,5	4,5	5,7	3,4	3,4	3,4
German	**PG**	65,2	66,0	63,9	64,5	66,3	63,8	64,8	65,4	65,4	65,2
	DE	2,6	2,1	2,7	2,5	2,4	2,6	2,6	2,7	2,7	2,3
Satimage	**PG**	83,6	83,5	82,9	83,4	83,4	83,0	83,5	83,6	83,6	83,6
	DE	11,6	12,2	10,7	11,5	12,2	10,7	11,5	11,6	11,7	11,6
Phoneme	**PG**	76,1	75,9	75,5	76,1	76,0	75,5	76,0	76,0	75,9	76,0
	DE	8,4	8,4	7,4	8,1	8,4	7,6	8,3	8,0	8,1	8,1
Waveform	**PG**	78,0	78,1	76,1	77,8	78,1	76,1	77,9	78,0	77,9	77,9
	DE	2,9	2,8	2,4	3,1	2,8	2,4	3,1	2,9	2,8	2,8
Segment	**PG**	94,8	94,2	94,6	94,9	94,7	94,6	95,0	94,8	94,8	94,8
	DE	1,4	1,1	1,0	1,1	1,2	1,2	1,2	1,4	1,3	1,4

A= Bagging
B = Boosting
C = Arc-x4

Anexo 5

Distribución de patrones por clase: submuestras de dos clases

			5 Clasificadores			
	CE Original		Arc-x4		Boosting	
	Clase 1	Clase 2	Clase 1	Clase 2	Clase 1	Clase 2
Cancer	355	191	346,5	199,9	266,0	280,4
Heart	120	96	112,8	103,3	112,9	103,8
Liver	116	160	120,3	155,7	122,0	154,0
Pima	400	215	364,8	250,2	352,4	262,6
Sonar	89	78	82,8	84,2	81,2	85,8
German	560	240	497,8	302,2	474,4	325,6
Phoneme	1.268	3.054	1130,6	2905,5	1825,4	2496,6

	7 Clasificadores				9 Clasificadores			
	Arc-x4		Boosting		Arc-x4		Boosting	
	Clase 1	Clase 2	Clase 1	Clase 2	Clase 1	Clase 2	Clase 1	Clase 2
Cancer	340,5	205,9	222,4	324,0	333,3	213,1	212,1	334,3
Heart	112,5	103,5	110,8	105,2	110,8	105,2	110,3	105,7
Liver	124,7	151,3	127,1	148,9	124,8	151,2	127,6	148,4
Pima	357,6	257,4	355,4	259,6	354,0	261,0	346,6	268,4
Sonar	79,1	87,9	77,9	89,1	76,7	90,4	72,4	94,6
German	480,9	319,1	462,9	336,5	469,9	330,1	464,7	335,1
Phoneme	1509,9	2812,1	1904,4	2381,6	1611,7	2710,3	1992,5	2329,5

Anexo 6

Distribución de patrones por clase: submuestras de tres clases

	CE Original		
	Clase1	Clase2	Clase3
Iris	40	40	40
Wine	48	57	39
Waveform	1326	1317	1357

	5 Clasificadores					
	Arcx4			Boosting		
	Clase1	Clase2	Clase3	Clase1	Clase2	Clase3
Iris	38,4	39,3	42,3	16,5	43,1	60,4
Wine	40,5	61,4	42,1	34,9	57,4	51,8
Waveform	1371,2	1301,1	1326,9	1393,3	1286,1	1299,8

	7 Clasificadores					
	Arcx4			Boosting		
	Clase1	Clase2	Clase3	Clase1	Clase2	Clase3
Iris	36,2	40,5	38,7	14,9	46,1	59,1
Wine	37,6	58,9	47,5	32,8	52,9	59,0
Waveform	1401,8	1289,9	1307,6	1412,5	1292,3	1294,4

	9 Clasificadores					
	Arcx4			Boosting		
	Clase1	Clase2	Clase3	Clase1	Clase2	Clase3
Iris	31,9	43,1	45,0	15,0	49,7	55,4
Wine	34,3	59,8	49,9	27,9	58,6	57,5
Waveform	1412,9	1274,6	1291,6	1422,9	1298,9	1277,3

Anexo 7

Distribución de patrones por clase: submuestras de cuatro, seis y siete clases (SMC con 5 y 7 clasificadores)

	Original						
	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7
Vehic	175	170	174	159	---	---	---
Glass	56	61	14	11	8	24	---
Satimage	1226	562	1.086	501	566	1.206	---
Segment	264	264	264	264	264	264	264

5 Clasificadores

	Arcx4						
	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7
Vehic	142,8	202,1	208,5	124,8	---	---	---
Glass	53,2	61,4	19,0	9,5	9,6	21,4	---
Satimage	1117,7	522,4	1095,0	614,0	553,4	1245,3	---
Segment	253,4	242,9	280,0	283,7	289,4	247,8	250,9

	Boosting						
	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7
Vehic	123,6	223,5	216,7	94,5	---	---	---
Glass	51,3	60,4	23,8	12,7	8,1	17,7	---
Satimage	238,8	346,1	1199,9	1018,2	620,8	1296,1	---
Segment	238,8	102,0	451,8	324,9	494,4	119,0	117,2

7 Clasificadores

	Arcx4						
	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7
Vehic	135,5	217,3	212,1	113,3	---	---	---
Glass	50,7	184,1	23,6	10,9	8,8	18,2	---
Satimage	1016,2	499,7	1094,4	710,5	572,7	1254,6	---
Segment	247,9	230,4	288,6	292,4	316,6	229,9	242,2

	Boosting						
	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7
Vehic	116,8	232,8	222,7	105,9	---	---	---
Glass	53,0	60,1	24,1	10,8	9,2	16,9	---
Satimage	596,5	308,1	1195,4	1072,5	616,3	1359,6	---
Segment	276,5	78,1	498,3	295,2	490,6	111,3	97,7

Anexo 7

Distribución de patrones por clase: submuestras de cuatro, seis y siete clases (SMC con 9 clasificadores)

	Arcx4						
	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7
Vehic	127,4	225,0	220,2	105,6	---	---	---
Glass	51,7	59,8	26,4	10,4	8,8	16,8	---
Satimage	964,5	458,7	1102,5	772,1	583,3	1266,8	---
Segment	235,8	222,4	310,4	307,5	324,6	223,2	224,1

	Boosting						
	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7
Vehic	116,1	237,4	225,2	99,4	---	---	---
Glass	53,4	60,3	24,6	9,6	9,3	16,8	---
Satimage	531,4	292,8	1237,9	1160,8	615,8	1309,2	---
Segment	245,8	61,5	483,1	365,1	500,9	102,6	88,9

Anexo 8

Análisis de diversidad (variabilidad y Q-estadístico)

	#	Variabilidad				Q estadístico			
	Clasif	SinRA1	Bagging	Boosting	Arcx4	SinRA1	Bagging	Boosting	Arcx4
Cancer	5	0,074	0,091	0,157	0,138	0,966	0,957	0,745	0,787
	7	0,116	0,106	0,161	0,303	0,896	0,915	0,816	0,840
	9	0,100	0,285	0,142	0,182	0,922	0,919	0,918	0,851
Heart	5	0,719	0,744	0,815	0,833	0,487	0,412	0,315	0,235
	7	0,904	0,782	0,907	0,896	0,301	0,494	0,226	0,134
	9	0,922	0,870	0,815	0,800	0,265	0,322	0,392	0,132
Liver	5	0,780	0,736	0,829	0,867	0,365	0,387	0,229	0,177
	7	0,887	0,893	0,930	0,951	0,345	0,271	0,157	0,019
	9	0,945	0,974	0,959	0,974	0,167	0,160	0,116	0,037
Pima	5	0,582	0,618	0,667	0,724	0,642	0,549	0,484	0,316
	7	0,728	0,723	0,817	0,714	0,532	0,519	0,384	0,297
	9	0,800	0,808	0,844	0,714	0,451	0,408	0,394	0,221
Sonar	5	0,746	0,634	0,776	0,649	0,329	0,452	0,245	0,246
	7	0,751	0,839	0,849	0,863	0,371	0,312	0,184	0,098
	9	0,907	0,893	0,985	0,813	0,321	0,321	0,103	0,039
Glass	5	0,635	0,650	0,630	0,420	0,711	0,707	0,681	0,691
	7	0,800	0,775	0,645	0,605	0,609	0,627	0,665	0,372
	9	0,785	0,830	0,731	0,408	0,570	0,554	0,678	0,749
Iris	5	0,133	0,120	0,087	0,300	0,530	0,569	0,484	0,438
	7	0,187	0,220	0,073	0,147	0,300	0,297	0,762	0,757
	9	0,280	0,213	0,107	0,207	0,099	0,349	0,239	0,470
Vehicle	5	0,691	0,719	0,743	0,653	0,625	0,587	0,554	0,515
	7	0,800	0,813	0,792	0,657	0,578	0,534	0,547	0,463
	9	0,836	0,855	0,831	0,582	0,476	0,469	0,477	0,407
Wine	5	0,553	0,588	0,571	0,412	0,681	0,645	0,721	0,406
	7	0,606	0,606	0,547	0,535	0,644	0,680	0,688	0,450
	9	0,624	0,653	0,547	0,477	0,573	0,549	0,641	0,531
German	5	0,661	0,650	0,692	0,611	0,556	0,561	0,496	0,419
	7	0,800	0,645	0,800	0,750	0,490	0,498	0,438	0,223
	9	0,882	0,857	0,889	0,624	0,397	0,405	0,382	0,336
Satimage	5	0,253	0,264	0,263	0,428	0,914	0,905	0,909	0,910
	7	0,304	0,308	0,304	0,230	0,899	0,717	0,898	0,907
	9	0,340	0,350	0,339	0,234	0,851	0,860	0,888	0,890
Phoneme	5	0,391	0,389	0,473	0,473	0,829	0,821	0,723	0,679
	7	0,500	0,480	0,567	0,544	0,784	0,796	0,512	0,577
	9	0,548	0,546	0,612	0,591	0,684	0,703	0,667	0,606
Waveform	5	0,471	0,485	0,502	0,533	0,676	0,657	0,644	0,594
	7	0,558	0,562	0,603	0,571	0,640	0,632	0,331	0,398
	9	0,623	0,618	0,669	0,549	0,490	0,499	0,567	0,535
Segment	5	0,240	0,261	0,266	0,260	0,844	0,837	0,849	0,820
	7	0,319	0,339	0,370	0,284	0,817	0,806	0,685	0,726
	9	0,370	0,400	0,430	0,320	0,780	0,780	0,750	0,730

Anexo 8

Análisis de diversidad (correlación y desacuerdo)

	#	Coeficiente de correlación				Medida de desacuerdo %			
	Clasif	SinRA1	Bagging	Boosting	Arcx4	SinRA1	Bagging	Boosting	Arcx4
Cancer	5	0,596	0,560	0,447	0,442	0,04	0,00	0,08	0,07
	7	0,473	0,564	0,464	0,436	0,05	0,04	0,08	0,08
	9	0,553	0,515	0,483	0,343	0,04	0,04	0,09	0,10
Heart	5	0,267	0,232	0,177	0,099	0,36	0,37	0,41	0,45
	7	0,181	0,257	0,123	0,076	0,40	0,35	0,43	0,46
	9	0,161	0,203	0,221	0,072	0,41	0,39	0,38	0,46
Liver	5	0,334	0,215	0,121	0,093	0,39	0,38	0,43	0,44
	7	0,160	0,136	0,089	0,022	0,41	0,42	0,45	0,50
	9	0,127	0,119	0,550	0,015	0,42	0,43	0,47	0,48
Pima	5	0,357	0,288	0,253	0,159	0,28	0,32	0,34	0,39
	7	0,275	0,274	0,216	0,143	0,13	0,32	0,36	0,41
	9	0,258	0,231	0,215	0,102	0,33	0,34	0,36	0,42
Sonar	5	0,138	0,277	0,165	0,121	0,39	0,31	0,38	0,42
	7	0,212	0,156	0,094	0,050	0,35	0,38	0,42	0,45
	9	0,156	0,174	0,073	0,003	0,40	0,39	0,45	0,46
Glass	5	0,461	0,402	0,434	0,396	0,26	0,29	0,26	0,29
	7	0,402	0,366	0,434	0,219	0,29	0,31	0,27	0,40
	9	0,383	0,341	0,404	0,433	0,31	0,30	0,31	0,29
Iris	5	0,439	0,482	0,544	0,500	0,07	0,05	0,07	0,06
	7	0,353	0,304	0,631	0,458	0,08	0,06	0,05	0,07
	9	0,334	0,363	0,505	0,404	0,10	0,04	0,05	0,11
Vehicle	5	0,354	0,315	0,303	0,269	0,31	0,33	0,34	0,36
	7	0,302	0,296	0,313	0,257	0,34	0,35	0,34	0,37
	9	0,304	0,300	0,264	0,217	0,34	0,35	0,37	0,39
Wine	5	0,439	0,385	0,462	0,292	0,26	0,29	0,26	0,32
	7	0,365	0,419	0,401	0,283	0,28	0,26	0,25	0,34
	9	0,430	0,360	0,355	0,303	0,26	0,28	0,28	0,32
German	5	0,303	0,300	0,268	0,213	0,32	0,32	0,27	0,37
	7	0,264	0,366	0,251	0,097	0,35	0,35	0,35	0,43
	9	0,234	0,226	0,194	0,169	0,36	0,36	0,38	0,39
Satimage	5	0,560	0,542	0,554	0,552	0,11	0,12	0,11	0,12
	7	0,533	0,528	0,537	0,550	0,12	0,12	0,12	0,11
	9	0,524	0,526	0,518	0,506	0,12	0,12	0,13	0,12
Phoneme	5	0,500	0,491	0,404	0,375	0,19	0,19	0,23	0,25
	7	0,445	0,464	0,375	0,350	0,21	0,21	0,25	0,26
	9	0,439	0,432	0,364	0,317	0,22	0,22	0,26	0,28
Waveform	5	0,333	0,320	0,329	0,300	0,23	0,24	0,25	0,27
	7	0,311	0,301	0,296	0,265	0,25	0,25	0,27	0,28
	9	0,290	0,285	0,284	0,258	0,26	0,25	0,28	0,28
Segment	5	0,400	0,398	0,428	0,376	0,11	0,12	0,12	0,13
	7	0,383	0,386	0,380	0,379	0,13	0,14	0,15	0,15
	9	0,370	0,380	0,360	0,340	0,16	0,17	0,18	0,18

APÉNDICE III

ANÁLISIS DE EFICIENCIA DEL CLASIFICADOR 1-NN CON SMC

Anexo 9

Utilización de SMC para combatir el desbalance (Metodologías 1 y 2)

		VOTACIÓN SIMPLE									
		Metodología 1					Metodología 2				
		A	B	C	D	E	A	B	C	D	E
Glass 7	g	84,4	86,6	86,1	84,9	84,1	84,4	84,4	84,4	87,4	83,6
	PG	91,5	92,0	91,0	92,5	91,0	91,5	91,5	91,5	93,5	90,0
	DE(g)	11,2	10,1	11,8	10,6	11,5	11,2	11,2	12,6	9,9	9,9
	DE(PG)	4,2	3,7	6,3	2,5	4,9	4,2	4,2	6,8	2,9	3,1
Vehicle 3	g	62,9	68,4	68,0	66,0	65,9	65,1	67,0	68,4	66,1	67,8
	PG	56,9	69,3	67,7	70,5	67,7	59,5	68,2	68,6	69,1	67,0
	DE(g)	2,7	3,3	3,6	3,9	2,4	1,7	5,1	2,4	4,8	1,3
	DE(PG)	3,4	2,1	2,7	3,6	2,6	1,1	4,0	3,1	2,5	2,3
Phoneme 3	g	71,6	74,3	74,0	74,3	74,3	77,4	75,1	73,7	74,9	73,9
	PG	67,4	73,4	72,1	73,8	72,7	74,6	73,9	71,8	74,1	71,1
	DE(g)	10,0	8,0	8,0	7,7	8,1	5,4	8,2	7,7	7,4	10,0
	DE(PG)	20,0	10,7	11,2	10,4	11,0	5,9	10,7	10,7	10,1	12,9
Satimage 11	g	75,0	79,4	79,5	76,7	75,3	74,8	78,2	78,0	77,6	75,0
	PG	61,6	80,4	79,7	81,3	77,7	61,6	80,5	80,1	81,3	76,4
	DE(g)	15,1	4,0	3,9	4,5	5,0	14,7	4,7	4,4	5,3	6,3
	DE(PG)	22,6	16,7	17,4	15,5	18,7	22,4	17,3	17,5	16,3	19,4

A = Selección secuencial (mayoritaria)
B = Selección aleatoria sin reemplazo (mayoritaria)
C = Bagging (mayoritaria)
D = Bagging (en dos clases)
E = Boosting (mayoritaria)

Nota: el número debajo el nombre de la base de datos corresponde al número de clasificadores que forman el SMC

Anexo 9

Utilización de SMC para combatir el desbalance (metodologías 3 y 4)

		VOTACIÓN SIMPLE									
		Metodología 3					Metodología 4				
		A	B	C	D	E	A	B	C	D	E
Glass 7	g	86,8	85,3	87,6	87,4	85,5	84,7	85,8	83,6	87,4	88,2
	PG	89,0	89,5	90,5	93,5	90,0	88,5	90,5	90,0	93,5	91,5
	DE(g)	8,6	9,2	9,0	9,8	10,6	9,0	9,3	11,9	9,8	8,9
	DE(PG)	5,8	4,8	4,8	1,4	5,9	6,0	4,8	5,6	1,4	5,2
Vehicle 3	g	61,1	66,2	65,9	64,7	63,6	64,2	65,9	66,3	64,8	67,3
	PG	55,7	67,1	67,0	69,6	64,3	58,6	66,4	67,0	67,9	65,8
	DE(g)	3,0	3,8	1,3	6,1	2,6	2,3	4,2	1,0	4,1	1,5
	DE(PG)	3,2	3,1	2,1	3,3	1,8	2,7	4,4	1,0	2,7	1,6
Phoneme 3	g	70,0	72,1	73,0	71,5	71,4	70,3	72,7	72,8	73,1	71,6
	PG	65,6	71,8	71,6	72,4	70,4	65,9	72,1	71,3	72,9	69,3
	DE(g)	8,8	8,0	7,8	8,0	7,1	9,2	7,4	7,8	6,8	9,9
	DE(PG)	96,0	9,7	10,1	9,0	10,0	10,2	9,5	9,7	9,1	12,7
Satimage 11	g	76,6	78,3	78,8	76,2	75,5	75,6	77,1	77,6	76,8	74,7
	PG	64,8	81,9	81,4	83,0	78,7	63,9	82,6	81,9	82,8	77,5
	DE(g)	12,7	4,3	3,3	6,9	5,7	11,6	5,0	4,4	6,4	6,8
	DE(PG)	20,3	14,4	14,6	12,9	16,6	19,1	14,8	15,6	14,4	18,7

A = Selección secuencial (mayoritaria)
B = Selección aleatoria sin reemplazo (mayoritaria)
C = Bagging (mayoritaria)
D = Bagging (en dos clases)
E = Boosting (mayoritaria)

Nota: el número debajo el nombre de la base de datos corresponde al número de clasificadores que forman el SMC

Anexo 10

Tamaño de submuestras balanceadas con SMC

	Metodología 1					Metodología 2				
	A	B	C	D	E	A	B	C	D	E
Glass										
SMC			336,0			298,4	289,6	293,6	307,0	260,0
Original			**174,0**					**165,6**		
Vehicle										
SMC			1020,0			777,0	700,6	706,6	763,6	490,2
Original			**679,0**					**508,6**		
Phoneme										
SMC			7608,0			6916,4	6635,8	6705,2	6811,0	6981,8
Original			**4322,0**					**3893,4**		
Satimage										
SMC			11000,0			10339,6	9840,0	9877,4	9965,2	5442,2
Original			**5147,0**					**4894,8**		

	Metodología 3					Metodología 4				
	A	B	C	D	E	A	B	C	D	E
Glass										
SMC	60,8	72,80	70,20	58,00	31,40	55,6	66,0	63,4	55,0	27,0
Original			**21,2**					**6,0**		
Vehicle										
SMC	427,4	521,6	492,2	390,0	357,0	283,4	36,2	354,0	320,8	154,4
Original			**294,2**					**86,8**		
Phoneme										
SMC	1724,8	2191,2	2021,0	1695,2	1421,4	1557,4	1982,4	1849,8	1564,0	795,6
Original			**1046,8**					**524,8**		
Satimage										
SMC	2020,8	3010,4	2913,4	2430,4	1089,2	1720,0	2599,6	2545,2	2230,8	785,9
Original			**805,6**					**442,4**		

A = Selección secuencial (mayoritaria)
B = Selección aleatoria sin reemplazo (mayoritaria)
C = Bagging (mayoritaria)
D = Bagging (en dos clases)
E = Boosting (mayoritaria)

Anexo 11

Tratamiento de bases de datos desbalanceadas de más de dos clases (umbral = promedio de patrones por clase)

		CE originales			
		Original	Wilson	SSM	Wilson+ SSM
Cayo 3	PG	75,95	77,70	67,00	78,15
	CK	0,7290	0,7485	0,7290	0,7537
	VK	0,000077	0,000073	0,000077	0,000072
Feltwell Test 1 3	PG	72,84	72,97	70,09	71,01
	CK	0,6375	0,6387	0,5991	0,6103
	VK	0,000051	0,000052	0,000054	0,000054
Fetwell Test 2 3	PG	87,71	87,27	87,55	90,59
	CK	0,8126	0,7561	0,8042	0,9059
	VK	0,000074	0,000015	0,000090	0,000008

		SMC							
		Bagging				Sel. Aleatoria sin reemplazo			
		Original	Wilson	SSM	Wilson+ SSM	Original	Wilson	SSM	Wilson+ SSM
Cayo 3	PG	79,03	80,98	78,62	80,46	79,10	80,98	77,80	79,41
	CK	0,7638	0,7854	0,7592	0,7797	0,7645	0,7854	0,7504	0,7681
	VK	0,000070	0,000065	0,000071	0,000066	0,000069	0,000065	0,000072	0,000068
Feltwell Test 1 3	PG	73,92	73,73	69,38	74,79	73,51	73,61	72,13	73,95
	CK	0,6544	0,6530	0,5906	0,6667	0,6482	0,6503	0,6303	0,6550
	VK	0,000049	0,000050	0,000054	0,000048	0,000050	0,000050	0,000051	0,000049
Feltwell Test 2 3	PG	90,20	88,93	87,87	92,52	89,77	89,35	91,29	91,89
	CK	84,8300	0,8300	0,8105	0,8863	0,8414	0,8353	0,8664	0,8761
	VK	0,000069	0,000077	0,000087	0,000053	0,000072	0,000075	0,000062	0,000057

Nota: el número que aparece bajo del nombre corresponde al número de clasificadores integrados.

Anexo 12

Tamaño de submuestras obtenidas con bases de datos debalanceadas de más de dos clases

	Inicial		Wilson		SSM		Wilson + SSM	
	sinRA1	Bagging	sinRA1	Bagging	sinRA1	Bagging	sinRA1	Bagging
Cayo								
SMC	7467,0		7014,3	7024,3	1637,4	1623,0	893,3	885,7
Original	**3086,0**		**2858,0**		**746,7**		**379,3**	
Feltwell								
SMC	2817,0		2795,0	2788,0	342,0	324,0	312,0	273,0
Original	**1418,0**		**1408,0**		**173,0**		**139,0**	

Anexo 13

Escalabilidad de algoritmos con SMC (Metodologías 1 y 2)

		VOTACIÓN SIMPLE							
		Metodología 1				Metodología 2			
		A	B	C	D	A	B	C	D
Glass	**PG**	68,0	56,5	50,5	62,5	59,5	59,0	58,5	61,0
	DE	8,6	10,7	7,4	7,7	11,2	10,8	10,8	7,0
Iris	**PG**	96,0	95,3	93,3	96,0	95,3	97,3	95,3	95,3
	DE	1,5	1,8	3,3	2,8	3,0	1,5	3,8	3,8
Vehicle	**PG**	64,5	62,1	60,6	60,4	56,3	57,6	57,0	57,9
	DE	2,8	3,4	2,5	1,6	4,0	3,9	3,8	1,8
Wine	**PG**	68,2	74,1	64,7	62,9	71,8	71,8	67,7	69,4
	DE	5,7	8,7	7,8	2,6	5,3	6,4	2,9	8,2
Cancer	**PG**	96,9	66,4	72,1	94,8	96,8	68,6	75,9	92,7
	DE	2,0	6,0	12,6	2,3	2,3	9,0	15,5	2,3
Heart	**PG**	65,2	50,4	45,2	57,6	67,4	49,6	47,0	64,1
	DE	4,2	5,3	3,8	3,0	3,8	11,5	7,2	5,0
Liver	**PG**	63,8	57,1	50,1	62,0	69,9	62,3	51,0	64,6
	DE	3,7	3,9	3,3	5,5	3,9	2,5	6,7	6,9
Pima	**PG**	68,9	59,4	60,0	70,1	71,5	62,6	62,0	70,3
	DE	3,3	1,9	3,6	6,2	3,2	2,2	1,7	3,1
Sonar	**PG**	79,0	62,0	49,8	75,1	66,8	61,0	52,2	63,9
	DE	7,0	8,7	5,1	9,4	10,7	9,9	9,2	8,3

A = Selección secuencial
B = Selección aleatoria sin reemplazo
C = Bagging
E = Boosting

Anexo 13

Escalabilidad de algoritmos con SMC (Metodologías 3 y 4)

		VOTACIÓN SIMPLE							
		Metodología 1				Metodología 2			
		A	B	C	D	A	B	C	D
Glass	PG	65,0	55,5	57,5	61,0	61,0	58,0	57,0	61,0
	DE	8,5	10,8	5,9	9,1	9,1	2,7	11,4	6,3
Iris	PG	95,9	95,9	92,9	93,5	94,7	9,3	92,7	96,0
	DE	2,6	2,6	3,4	4,4	3,0	2,8	4,4	4,4
Vehicle	PG	63,1	62,6	59,1	59,6	55,9	57,6	57,8	57,0
	DE	2,8	3,7	3,5	1,3	5,0	4,0	3,1	1,9
Wine	PG	66,5	67,7	63,5	61,8	72,4	71,2	67,1	69,4
	DE	4,5	8,6	9,9	2,9	5,3	6,7	2,5	8,2
Cancer	PG	96,5	67,6	72,9	93,7	95,3	74,2	78,4	93,1
	DE	1,4	5,6	12,9	2,9	4,2	11,2	15,2	3,8
Heart	PG	61,5	51,5	46,7	56,7	67,4	52,2	47,8	64,1
	DE	4,4	4,6	1,6	5,3	5,0	9,6	6,6	3,1
Liver	PG	59,1	53,6	51,9	60,3	66,4	62,0	53,0	65,2
	DE	4,7	4,4	5,9	3,2	3,0	5,3	6,8	3,7
Pima	PG	65,1	57,8	57,9	69,0	71,2	61,7	62,0	70,3
	DE	3,8	2,9	4,0	6,3	3,6	1,9	2,3	2,0
Sonar	PG	76,6	58,5	52,7	73,2	66,3	59,5	48,8	62,4
	DE	11,9	11,4	2,2	8,3	10,3	7,8	6,7	5,6

A = Selección secuencial
B = Selección aleatoria sin reemplazo
C = Bagging
E = Boosting

Anexo 14

Tamaños de submuestras utilizadas en la escalabilidad de algoritmos (Metodología 2 y 3)

	CE	VOTACIÓN SIMPLE							
		Metodología 2				Metodología 3			
	Original	A	B	C	D	A	B	C	D
Cancer									
SMC		527,6	337,2	346,6	519,2	56,2	438,2	427,2	59,6
Original	546,0	**528,8**				**55,6**			
Heart									
SMC	216,0	129,0	112,2	106,0	178,8	120,6	152,4	151,8	112,8
Original		**138,0**				**122,2**			
Liver									
SMC	276,0	163,0	152,6	48,4	174,8	166,0	171,6	193,8	137,2
Original		**175,0**				**164,4**			
Pima									
SMC	615,0	423,3	259,0	361,2	450,8	310,8	403,4	382,2	256,2
Original		**427,8**				**303,0**			
Sonar									
SMC		117,2	90,2	89,8	117,0	75,6	108,0	119,6	72,0
Original	167,0	**133,4**				**70,0**			
Glass									
SMC		101,0	92,6	102,6	105,6	114,0	116,8	101,4	94,0
Original	174,0	**119,4**				**99,4**			
Iris									
SMC		112,8	113,2	113,8	114,6	32,0	27,5	28,0	23,6
Original	120,0	**115,6**				**20,6**			
Vehicle									
SMC		383,6	383,6	413,6	407,4	458,2	461,8	391,4	382,0
Original	678,0	**234,6**				**419,0**			
Wine									
SMC		95,4	90,4	96,2	104	65,0	66,4	59,6	93,4
Original	144,0	**103,8**				**60,0**			

A = Selección secuencial
B = Selección aleatoria sin reemplazo
C = Bagging
E = Boosting

Anexo 14

Tamaños de submuestras utilizadas en la escalabilidad de algoritmos (Metodología 4)

	CE	Metodología 4			
	Original	A	B	C	D
Cancer					
SMC		20,0	47,2	43,8	34,0
Original	546,0		**28,8**		
Heart					
SMC	216,0	36,2	35,2	25,6	38,4
Original			**36,4**		
Liver					
SMC	276,0	57,8	46,0	48,0	51,2
Original			**58,8**		
Pima					
SMC	615,0	92,2	84,8	85,2	87,4
Original			**77,2**		
Sonar					
SMC		34,6	26,4	28,0	33,2
Original	167,0		**38,4**		
Glass					
SMC		52,4	48,6	47,2	49,2
Original	174,0		**43,2**		
Iris					
SMC		19,8	22,2	19,8	18,8
Original	120,0		**11,4**		
Vehicle					
SMC		143,0	139,2	148,2	139,6
Original	678,0		**182,8**		
Wine					
SMC		15,4	14,6	16,6	15,8
Original	144,0		**15,8**		

A = Selección secuencial
B = Selección aleatoria sin reemplazo
C = Bagging
E = Boosting

APÉNDICE IV

FUSIÓN DE CLASIFICADORES

Anexo 15

Votación simple (SMC de 3 y 5 clasificadores)

		CE original	3 Clasificadores				5 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	66.4	72.1	94.2	95.2	95.9	95.3	91.7	93.4
	DE	2.5	6.0	12.6	1.0	2.8	3.0	3.2	3.6	4.2
Heart	**PG**	58.2	50.4	45.2	57.8	58.9	60.7	64.4	57.4	59.6
	DE	6.2	5.3	3.8	3.0	10.1	3.8	6.6	6.8	6.2
Liver	**PG**	65.2	57.1	50.1	62.0	69.0	64.4	65.8	62.6	62.6
	DE	4.8	3.9	3.3	5.5	5.0	5.1	6.6	4.5	3.1
Pima	**PG**	65.9	59.4	60.0	70.1	69.6	70.5	70.7	71.1	70.7
	DE	5.2	1.9	3.6	6.2	3.2	3.4	3.4	2.3	3.3
Sonar	**PG**	82.0	62.0	49.8	75.1	76.6	70.7	73.7	69.8	69.8
	DE	9.4	8.7	5.1	9.4	12.5	8.6	11.0	10.0	9.4
Glass	**PG**	70.0	56.5	60.5	62.5	62.0	61.5	65.0	61.0	61.0
	DE	5.3	10.7	7.4	7.7	9.1	4.9	5.9	5.2	5.5
Iris	**PG**	96.0	95.3	93.3	96.0	96.0	96.0	96.0	96.0	93.3
	DE	1.5	1.8	3.3	2.8	3.7	2.8	1.5	3.7	4.1
Vehicle	**PG**	64.2	62.1	60.6	60.4	62.1	62.2	62.0	61.9	61.0
	DE	1.8	3.4	2.5	1.6	2.6	3.6	3.0	4.4	2.7
Wine	**PG**	72.4	74.1	64.7	62.9	67.7	67.1	65.9	70.0	62.9
	DE	3.4	6.7	7.8	2.6	7.5	5.7	11.7	4.4	10.3
German	**PG**	65.2	67.8	64.1	67.4	67.4	68.8	67.3	68.8	68.3
	DE	2.6	2.7	1.6	2.1	2.0	2.9	3.8	4.1	2.3
Satimage	**PG**	83.6	83.1	82.9	83.5	82.6	82.8	82.8	83.0	82.4
	DE	11.6	12.2	13.5	12.6	12.7	12.1	12.4	12.7	12.7
Phoneme	**PG**	76.1	76.1	74.4	73.5	75.8	75.9	75.0	74.8	74.1
	DE	8.4	8.5	8.1	10.5	9.4	9.8	8.5	10.9	12.0
Waveform	**PG**	78.0	78.7	78.7	78.5	79.3	80.8	80.7	79.4	78.6
	DE	2.9	2.2	1.5	1.9	2.3	2.4	1.3	2.1	1.7
Segment	**PG**	94.8	91.7	90.7	89.1	90.4	89.9	89.8	87.8	88.2
	DE	1.4	2.6	1.1	1.3	2.0	1.8	2.0	1.3	2.2

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 15

Votación simple (SMC de 7 y 9 clasificadores)

		CE original	7 Clasificadores				9 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	97.4	96.2	90.8	93.7	96.2	96.4	92.3	93.6
	DE	2.5	1.4	2.4	4.8	3.9	2.9	2.8	1.7	3.8
Heart	**PG**	58.2	67.4	64.4	62.6	58.5	62.2	62.6	63.0	59.3
	DE	6.2	5.8	4.8	9.2	5.7	2.1	5.0	5.5	3.9
Liver	**PG**	65.2	62.0	64.1	62.0	60.0	63.8	63.2	65.2	63.8
	DE	4.8	3.9	6.7	6.4	8.8	7.2	5.2	4.7	6.9
Pima	**PG**	65.9	71.9	71.2	69.4	68.0	72.8	72.7	71.0	69.7
	DE	5.2	2.2	4.8	3.2	2.9	5.0	1.2	2.6	2.9
Sonar	**PG**	82.0	70.7	68.8	67.3	71.7	68.3	70.2	69.3	72.2
	DE	9.4	12.2	10.3	9.9	10.1	12.6	8.3	8.4	9.5
Glass	**PG**	70.0	63.0	61.5	63.0	56.5	61.5	60.5	61.0	57.0
	DE	5.3	10.8	8.0	9.6	5.8	7.2	8.2	6.8	9.9
Iris	**PG**	96.0	95.3	96.0	95.3	98.0	98.0	94.0	94.0	91.3
	DE	1.5	1.83	3.7	1.8	3.0	1.8	4.4	1.5	5.1
Vehicle	**PG**	64.2	60.1	60.7	60.3	58.3	61.4	60.6	62.3	54.8
	DE	1.8	1.8	2.8	4.9	3.8	1.9	2.3	4.7	4.3
Wine	**PG**	72.4	72.4	72.4	73.5	66.5	65.3	75.9	72.4	67.7
	DE	3.4	4.5	8.7	8.8	4.9	4.8	9.8	2.6	7.5
German	**PG**	65.2	69.8	68.6	69.7	69.9	68.8	70.2	68.5	68.7
	DE	2.6	0.9	2.1	3.2	1.9	3.4	3.0	2.1	2.5
Satimage	**PG**	83.6	83.0	82.6	82.7	82.5	82.9	82.9	82.9	82.5
	DE	11.6	12.1	14.1	14.8	13.5	14.5	14.3	14.9	14.3
Phoneme	**PG**	76.1	75.8	75.1	74.5	74.3	75.0	75.0	71.9	74.4
	DE	8.4	9.1	9.2	10.8	10.6	10.0	9.4	13.7	11.1
Waveform	**PG**	78.0	81.8	81.9	79.3	79.4	82.7	83.2	80.0	78.8
	DE	2.9	1.8	1.1	2.0	1.7	1.8	1.4	1.9	2.3
Segment	**PG**	94.8	89.0	86.9	87.1	86.0	87.6	87.7	87.0	83.6
	DE	1.4	2.8	2.3	2.8	2.8	2.6	2.1	2.9	1.6

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 15

Votación simple (SMC de 15 y 25 clasificadores)

		CE original	15 Clasificadores				25 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	95.8	95.8	87.9	92.7	95.8	95.8	87.7	91.4
	DE	2.5	4.0	3.1	5.8	4.5	3.3	3.6	4.8	6.0
Heart	**PG**	58.2	64.1	63.0	63.7	58.9	64.8	64.4	55.9	56.3
	DE	6.2	3.4	5.9	3.4	6.1	3.5	3.3	7.2	8.1
Liver	**PG**	65.2	64.9	66.4	62.9	62.0	63.8	63.2	60.9	62.0
	DE	4.8	5.2	5.9	3.0	7.4	4.2	3.3	6.4	5.9
Pima	**PG**	65.9	72.8	70.9	71.2	68.1	72.7	74.6	69.3	66.9
	DE	5.2	4.8	4.9	3.8	3.3	2.0	2.2	1.2	3.1
Sonar	**PG**	82.0	64.9	62.0	65.9	68.8	64.9	66.3	60.5	61.0
	DE	9.4	7.4	2.2	12.1	8.0	5.1	6.8	8.2	3.9
Glass	**PG**	70.0	61.0	59.0	60.5	64.5	57.5	61.5	50.0	37.5
	DE	5.3	9.6	6.8	12.3	1.1	7.3	10.1	---	---
Iris	**PG**	96.0	95.3	94.7	95.3	94.7	96.7	96.7	94.7	94.0
	DE	1.5	1.8	3.8	5.1	3.8	3.3	4.1	3.8	2.8
Vehicle	**PG**	64.2	59.8	58.2	52.0	50.0	59.8	60.4	---	50.1
	DE	1.8	7.2	3.6	2.7	23.0	3.3	5.2	---	4.9
Wine	**PG**	72.4	68.2	68.8	67.7	66.5	71.8	71.2	69.4	64.7
	DE	3.4	1.3	9.7	3.6	9.9	11.1	10.7	6.1	10.4
German	**PG**	65.2	70.2	68.8	70.9	67.3	72.0	71.0	69.9	69.0
	DE	2.6	2.7	2.8	1.6	2.7	2.4	3.1	1.7	2.1
Satimage	**PG**	83.6	82.8	82.4	82.3	84.6	82.5	82.7	82.5	80.6
	DE	11.6	14.4	14.6	15.3	7.1	14.8	14.6	14.4	14.4
Phoneme	**PG**	76.1	75.6	75.3	72.4	72.7	74.6	73.9	74.5	74.2
	DE	8.4	8.5	9.4	12.0	7.8	10.5	11.4	9.2	10.4
Waveform	**PG**	78.0	83.7	83.9	79.8	79.8	84.0	84.5	81.1	79.2
	DE	2.9	1.6	1.4	0.9	1.9	1.2	1.5	2.2	3.0
Segment	**PG**	94.8	85.3	84.8	84.7	79.8	82.6	84.2	80.9	76.6
	DE	1.4	2.4	2.1	2.5	1.8	3.3	3.7	3.1	3.9

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 16

Votación ponderada dinámica por promedio de distancias (SMC de 3 y 5 clasificadores)

		CE original	3 Clasificadores				5 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	96.5	62.0	70.8	93.6	95.3	95.3	92.3	94.6
	DE	2.5	2.3	6.4	12.0	0.3	2.9	3.2	2.5	3.7
Heart	**PG**	58.2	65.6	49.6	45.9	57.8	60.0	64.8	58.9	59.3
	DE	6.2	3.6	4.6	4.0	2.4	5.0	5.9	6.5	3.7
Liver	**PG**	65.2	65.2	57.1	50.1	62.3	64.9	64.9	61.2	62.0
	DE	4.8	4.7	3.9	3.3	6.2	4.5	7.4	5.2	3.1
Pima	**PG**	65.9	68.4	59.1	60.3	70.9	70.9	71.2	71.9	71.2
	DE	5.2	3.4	2.6	3.4	6.6	3.5	2.9	2.9	3.2
Sonar	**PG**	82.0	79.0	62.9	49.8	75.6	74.6	77.6	73.2	74.2
	DE	9.4	7.0	10.1	5.1	9.0	6.6	7.8	11.3	6.6
Glass	**PG**	70.0	68.0	59.0	62.5	63.0	61.0	63.0	64.0	65.0
	DE	5.3	8.7	9.6	7.7	8.7	8.9	2.1	10.1	8.5
Iris	**PG**	96.0	96.7	95.3	94.0	95.3	95.3	95.3	97.3	93.3
	DE	1.5	2.4	1.8	3.7	3.1	1.8	1.8	2.8	4.1
Vehicle	**PG**	64.2	64.7	59.0	63.4	61.5	64.5	61.7	63.4	61.9
	DE	1.8	3.5	2.2	3.6	1.4	2.5	3.1	3.8	3.5
Wine	**PG**	72.4	70.6	70.0	66.5	65.3	66.5	65.9	67.7	65.9
	DE	3.4	5.5	7.0	4.9	4.4	6.1	11.5	5.9	10.7
German	**PG**	65.2	67.6	64.3	67.6	67.2	68.8	67.3	68.6	68.0
	DE	2.6	2.9	1.5	2.5	1.9	2.6	3.7	3.4	2.4
Satimage	**PG**	83.6	83.1	83.2	83.6	82.7	82.9	82.8	83.1	82.0
	DE	11.6	12.3	13.6	12.7	13.6	13.1	12.5	12.5	12.3
Phoneme	**PG**	76.1	76.0	74.5	73.7	75.7	76.5	75.0	74.6	75.6
	DE	8.4	8.6	8.2	10.2	9.4	9.5	8.6	10.8	8.8
Waveform	**PG**	78.0	78.8	78.2	78.6	79.2	80.8	80.7	79.4	78.6
	DE	2.9	2.2	2.5	1.9	2.3	2.4	1.3	2.1	1.7
Segment	**PG**	94.8	92.5	93.0	91.3	91.9	91.9	92.1	90.6	92.0
	DE	1.4	2.0	1.1	1.5	2.2	1.8	1.3	1.6	1.6

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 16

Votación ponderada dinámica por promedio de distancias (SMC de 7 y 9 clasificadores)

		CE original	7 Clasificadores				9 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	97.1	96.2	93.4	94.2	96.2	96.2	92.9	93.9
	DE	2.5	1.6	2.4	4.1	2.9	2.9	2.7	1.4	4.0
Heart	**PG**	58.2	66.7	66.7	60.0	57.8	62.2	62.6	63.4	60.0
	DE	6.2	4.7	5.6	6.2	4.4	4.8	5.1	3.5	5.9
Liver	**PG**	65.2	62.6	63.5	62.3	59.4	65.2	64.9	62.9	66.1
	DE	4.8	4.0	4.0	4.9	7.7	7.7	7.2	5.1	4.8
Pima	**PG**	65.9	70.6	70.9	70.6	67.7	72.0	72.7	69.7	69.5
	DE	5.2	2.3	4.3	4.1	4.4	4.5	1.7	3.2	3.1
Sonar	**PG**	82.0	75.6	73.2	73.7	75.6	71.7	73.2	73.7	74.2
	DE	9.4	14.2	10.6	6.8	10.2	10.3	11.3	8.3	7.2
Glass	**PG**	70.0	64.5	64.0	64.0	64.5	62.0	64.0	62.5	58.0
	DE	5.3	8.9	10.8	11.5	10.4	7.8	10.6	7.3	9.3
Iris	**PG**	96.0	96.0	97.3	94.7	98.0	97.3	95.3	94.7	94.0
	DE	1.5	1.5	1.5	1.8	1.8	2.8	3.8	1.8	4.4
Vehicle	**PG**	64.2	61.0	62.3	59.4	60.1	63.1	60.7	63.9	58.4
	DE	1.8	2.3	2.5	4.7	4.2	3.0	2.4	3.9	2.7
Wine	**PG**	72.4	73.5	71.8	71.8	62.9	64.7	77.7	71.8	66.5
	DE	3.4	7.8	9.9	8.7	7.1	4.7	10.1	5.3	9.0
German	**PG**	65.2	69.4	69.2	69.3	67.9	69.3	70.8	68.8	67.6
	DE	2.6	1.6	1.8	3.2	1.3	3.3	3.1	2.2	2.8
Satimage	**PG**	83.6	83.1	82.6	82.5	82.6	82.9	81.4	83.4	82.4
	DE	11.6	14.1	14.3	14.7	13.6	14.6	17.6	14.0	13.7
Phoneme	**PG**	76.1	75.9	75.5	74.4	74.9	75.4	75.2	72.9	74.5
	DE	8.4	9.0	9.3	10.4	10.2	9.6	9.2	11.8	11.4
Waveform	**PG**	78.0	81.8	81.9	79.3	79.6	82.7	83.2	80.0	79.5
	DE	2.9	1.9	1.1	1.9	1.4	1.7	1.4	1.9	1.9
Segment	**PG**	94.8	91.8	90.4	89.8	89.6	91.0	90.7	89.9	86.9
	DE	1.4	2.6	1.7	2.1	2.9	2.0	1.8	2.2	1.9

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 16

Votación ponderada dinámica por promedio de distancias (SMC de 15 y 25 clasificadores)

		CE original	15 Clasificadores				25 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	95.8	95.8	90.4	94.9	95.6	95.6	90.4	92.9
	DE	2.5	4.0	3.4	3.6	3.6	3.3	3.9	3.9	4.1
Heart	**PG**	58.2	63.0	63.3	61.9	56.3	65.6	65.6	56.0	57.0
	DE	6.2	5.6	4.8	11.5	8.0	4.3	4.5	4.9	7.3
Liver	**PG**	65.2	65.8	68.4	64.9	60.9	64.4	63.5	65.2	62.3
	DE	4.8	5.0	6.4	3.5	8.7	6.5	5.0	4.6	5.2
Pima	**PG**	65.9	73.1	71.2	71.6	67.2	72.9	74.0	69.7	67.2
	DE	5.2	4.2	5.1	4.4	5.3	3.9	3.5	2.6	4.0
Sonar	**PG**	82.0	74.6	65.4	69.3	70.7	69.8	72.7	67.3	67.3
	DE	9.4	6.6	3.2	11.5	5.7	8.0	3.2	7.0	2.8
Glass	**PG**	70.0	61.0	63.5	60.0	64.5	61.0	62.0	50.0	37.5
	DE	5.3	8.6	9.5	14.1	3.3	7.2	7.6	---	---
Iris	**PG**	96.0	97.3	96.0	93.3	95.3	97.3	96.0	95.3	94.7
	DE	1.5	1.5	3.7	6.2	3.0	2.8	3.7	3.8	3.0
Vehicle	**PG**	64.2	58.2	59.7	52.3	55.4	61.9	61.0	---	50.1
	DE	1.8	4.1	4.1	2.8	8.4	3.6	5.9	---	4.9
Wine	**PG**	72.4	68.2	66.5	67.1	68.2	72.4	70.6	71.8	62.4
	DE	3.4	8.7	11.1	5.3	7.3	6.8	10.4	6.1	8.4
German	**PG**	65.2	70.0	69.3	70.9	66.1	70.2	69.8	70.4	65.6
	DE	2.6	3.2	2.7	2.6	2.0	1.2	3.5	2.2	1.2
Satimage	**PG**	83.6	82.9	82.6	82.7	82.0	82.8	83.0	82.5	80.7
	DE	11.6	14.2	14.7	14.3	11.8	14.7	14.5	14.7	14.2
Phoneme	**PG**	76.1	76.0	75.9	73.9	74.2	75.8	75.1	75.6	75.2
	DE	8.4	9.1	9.2	9.5	9.5	9.5	10.6	8.8	7.7
Waveform	**PG**	78.0	83.7	83.9	79.8	79.8	84.0	84.5	81.2	80.1
	DE	2.9	1.6	1.4	0.9	1.9	1.3	1.6	2.2	1.7
Segment	**PG**	94.8	89.2	88.9	88.6	84.7	88.4	88.6	86.3	82.7
	DE	1.4	2.1	1.6	1.8	1.2	3.4	2.9	3.2	3.1

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 17

Votación ponderada dinámica con distancia inversa (SMC de 5 y 7 clasificadores)

		CE original	5 Clasificadores				7 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	95.3	95.3	92.1	94.6	97.1	96.2	93.3	94.2
	DE	2.5	2.9	3.2	2.6	3.7	1.6	2.4	3.9	2.9
Heart	**PG**	58.2	60.0	64.8	58.5	58.9	66.7	66.7	60.0	57.8
	DE	6.2	5.0	5.9	7.2	3.0	4.7	5.6	6.2	4.4
Liver	**PG**	65.2	64.6	64.4	60.6	61.2	61.7	62.9	62.0	58.8
	DE	4.8	4.7	7.0	5.2	3.0	3.8	3.9	4.4	7.3
Pima	**PG**	65.9	70.9	71.2	71.9	71.2	70.6	70.9	70.6	67.7
	DE	5.2	3.5	2.9	2.9	3.2	2.3	4.3	4.1	4.4
Sonar	**PG**	82.0	74.6	77.6	73.2	74.2	75.6	73.2	73.7	75.6
	DE	9.4	6.6	7.8	11.3	6.6	14.2	10.6	6.8	10.2
Glass	**PG**	70.0	82.9	82.8	83.1	82.0	78.0	82.6	82.5	82.6
	DE	5.3	13.1	12.5	12.5	13.3	14.2	14.3	14.7	13.6
Iris	**PG**	96.0	95.3	95.3	97.3	93.3	96.0	97.3	94.7	98.0
	DE	1.5	1.8	1.8	2.8	4.1	1.5	1.5	1.8	1.8
Vehicle	**PG**	64.2	64.5	63.4	62.9	61.9	61.0	62.3	59.4	60.1
	DE	1.8	2.5	2.5	3.4	3.5	2.3	2.5	4.7	4.2
Wine	**PG**	72.4	66.5	65.9	65.9	65.9	73.5	71.8	71.8	62.9
	DE	3.4	6.1	11.5	4.9	10.7	7.8	9.9	8.7	7.1
German	**PG**	65.2	68.8	67.3	68.6	68.0	69.4	69.2	69.3	67.9
	DE	2.6	2.6	3.7	3.4	2.4	1.6	1.8	3.2	1.3
Satimage	**PG**	83.6	61.0	62.5	63.5	65.0	64.0	64.0	64.0	65.5
	DE	11.6	8.9	1.8	9.6	8.5	9.6	10.8	11.5	11.2
Phoneme	**PG**	76.1	76.5	75.0	74.6	75.6	75.9	75.5	74.4	74.9
	DE	8.4	9.5	8.6	10.8	8.8	9.0	9.3	10.4	10.2
Waveform	**PG**	78.0	80.8	80.7	79.4	78.6	81.8	81.9	79.3	79.6
	DE	2.9	2.4	1.3	2.1	1.7	1.9	1.1	1.9	1.4
Segment	**PG**	94.8	91.0	91.5	89.8	91.2	91.0	89.3	89.0	88.7
	DE	1.4	2.0	1.1	1.6	1.4	2.5	1.9	2.3	2.8

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 17

Votación ponderada dinámica con distancia inversa (SMC de 9 clasificadores)

		CE original	9 Clasificadores			
			A	B	C	D
Cancer	**PG**	95.6	96.2	96.2	92.9	93.9
	DE	2.5	2.9	2.7	1.4	4.0
Heart	**PG**	58.2	62.2	62.6	63.4	60.0
	DE	6.2	4.8	5.1	3.5	5.9
Liver	**PG**	65.2	64.6	64.4	62.6	64.9
	DE	4.8	7.9	6.7	5.0	5.0
Pima	**PG**	65.9	72.0	72.7	69.7	69.5
	DE	5.2	4.5	1.7	3.2	3.1
Sonar	**PG**	82.0	71.7	73.2	73.7	73.7
	DE	9.4	10.3	11.3	8.3	7.2
Glass	**PG**	70.0	82.9	83.0	83.4	82.4
	DE	5.3	14.6	14.3	14.0	13.7
Iris	**PG**	96.0	97.3	95.3	94.7	94.0
	DE	1.5	2.8	3.8	1.8	4.4
Vehicle	**PG**	64.2	63.1	60.7	63.9	58.4
	DE	1.8	3.0	2.4	3.9	2.7
Wine	**PG**	72.4	64.7	77.7	71.2	70.0
	DE	3.4	4.7	10.1	5.7	9.6
German	**PG**	65.2	69.3	70.8	68.8	67.6
	DE	2.6	3.3	3.1	2.2	2.8
Satimage	**PG**	83.6	61.5	64.0	62.5	58.0
	DE	11.6	8.0	10.6	7.3	9.3
Phoneme	**PG**	76.1	75.4	75.2	72.9	74.5
	DE	8.4	9.6	9.2	11.8	11.4
Waveform	**PG**	78.0	83.5	83.2	80.0	79.5
	DE	2.9	0.8	1.4	1.9	1.9
Segment	**PG**	94.8	90.2	89.5	88.6	85.9
	DE	1.4	1.9	1.8	2.1	1.6

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 18

Votación ponderada dinámica según Shepard (a = β = 1, SMC de 5 y 7 clasificadores)

		CE original	5 Clasificadores				7 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	95.8	95.5	94.0	94.9	96.7	96.1	94.6	94.3
	DE	2.5	3.3	2.7	3.0	3.5	1.5	2.5	2.8	3.4
Heart	**PG**	58.2	54.8	61.1	55.9	56.7	61.5	62.6	57.8	53.7
	DE	6.2	6.1	3.9	5.9	4.8	3.0	5.3	5.8	3.5
Liver	**PG**	65.2	62.3	65.5	59.7	64.1	63.2	58.8	59.7	55.9
	DE	4.8	4.2	4.2	3.1	6.5	6.7	5.1	4.4	4.7
Pima	**PG**	65.9	67.5	68.4	67.6	65.5	67.3	66.4	64.7	66.5
	DE	5.2	4.8	4.6	3.8	4.0	1.2	4.0	5.3	2.9
Sonar	**PG**	82.0	74.2	75.6	71.2	74.6	73.2	71.2	72.2	74.2
	DE	9.4	7.2	10.1	11.9	7.0	12.6	10.0	6.6	11.8
Glass	**PG**	70.0	60.5	63.5	64.0	65.0	64.5	63.5	65.0	65.5
	DE	5.3	8.6	3.8	8.4	8.5	9.4	10.1	10.8	8.7
Iris	**PG**	96.0	96.0	96.0	97.3	93.3	95.3	96.0	95.3	97.3
	DE	1.5	2.8	1.5	2.8	4.1	1.8	3.7	3.0	2.8
Vehicle	**PG**	64.2	65.1	63.2	63.1	61.5	63.4	62.8	62.2	61.5
	DE	1.8	3.3	2.7	2.7	2.1	2.5	1.9	2.1	4.3
Wine	**PG**	72.4	67.1	70.0	70.0	67.1	72.4	74.1	70.0	61.2
	DE	3.4	2.5	9.6	3.8	9.6	4.9	8.7	7.0	6.7
German	**PG**	65.2	67.1	67.4	66.2	66.3	67.8	66.6	68.5	67.0
	DE	2.6	2.3	2.6	3.8	2.5	2.1	4.3	3.5	1.7
Satimage	**PG**	83.6	83.2	83.1	83.0	82.6	83.2	83.2	82.8	82.8
	DE	11.6	12.0	11.8	11.6	12.0	11.9	12.6	12.4	12.6
Phoneme	**PG**	76.1	76.0	75.1	74.8	75.6	76.0	75.2	74.7	74.8
	DE	8.4	9.7	8.5	10.9	8.9	8.9	9.3	10.8	10.2
Waveform	**PG**	78.0	80.5	80.3	79.0	78.3	81.0	81.5	79.0	79.2
	DE	2.9	2.7	1.2	2.5	1.8	1.6	1.4	2.8	2.0
Segment	**PG**	94.8	92.6	92.4	93.0	92.8	92.9	92.9	91.9	90.2
	DE	1.4	1.1	1.6	1.6	1.8	2.2	1.1	2.2	3.1

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 18

Votación ponderada dinámica según Shepard (a = β = 1, SMC de 9 clasificadores)

		CE original	9 Clasificadores			
			A	B	C	D
Cancer	**PG**	95.6	95.3	96.4	93.0	94.5
	DE	2.5	3.0	2.6	1.3	2.5
Heart	**PG**	58.2	58.2	59.6	61.1	58.5
	DE	6.2	3.4	10.2	4.0	8.1
Liver	**PG**	65.2	65.8	62.6	57.4	63.8
	DE	4.8	4.1	5.6	8.6	3.4
Pima	**PG**	65.9	66.9	66.1	65.0	65.6
	DE	5.2	5.6	4.7	4.7	5.5
Sonar	**PG**	82.0	72.2	73.7	73.2	72.7
	DE	9.4	10.1	11.4	8.3	7.9
Glass	**PG**	70.0	62.5	63.5	63.5	58.0
	DE	5.3	7.3	10.4	8.2	9.3
Iris	**PG**	96.0	97.3	94.7	94.0	92.7
	DE	1.5	2.8	3.8	1.5	5.5
Vehicle	**PG**	64.2	63.7	61.5	65.6	59.2
	DE	1.8	2.2	2.9	1.5	3.4
Wine	**PG**	72.4	69.4	73.5	71.8	65.3
	DE	3.4	4.5	10.8	4.9	10.7
German	**PG**	65.2	67.3	67.1	68.1	65.7
	DE	2.6	2.1	2.7	2.8	1.3
Satimage	**PG**	83.6	82.9	83.0	82.9	82.0
	DE	11.6	13.0	12.9	12.4	12.2
Phoneme	**PG**	76.1	75.0	74.8	72.9	74.4
	DE	8.4	9.9	9.4	12.2	11.0
Waveform	**PG**	78.0	82.1	83.1	79.5	79.3
	DE	2.9	1.9	1.7	2.3	1.7
Segment	**PG**	94.8	93.5	92.8	91.7	88.8
	DE	1.4	1.6	1.7	2.1	1.7

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 19

Votación ponderada dinámica Shepard modificado (a =según el orden y β = 1, SMC de 3 y 5 clasificadores)

		CE original	3 Clasificadores				5 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	67.7	74.5	94.9	94.6	96.1	95.9	95.3	95.3
	DE	2.5	8.9	9.8	1.2	2.3	2.3	2.2	2.9	3.3
Heart	**PG**	58.2	52.6	46.7	58.5	54.4	54.4	61.9	56.7	56.7
	DE	6.2	6.2	5.0	1.0	7.4	5.0	5.9	5.9	4.7
Liver	**PG**	65.2	55.9	51.9	60.9	60.3	60.3	65.8	58.3	62.0
	DE	4.8	4.1	4.5	4.2	2.8	6.2	3.8	3.3	16.8
Pima	**PG**	65.9	57.7	59.2	67.6	66.1	66.9	68.2	66.9	65.1
	DE	5.2	2.2	3.7	6.3	3.7	5.1	4.9	3.2	3.3
Sonar	**PG**	82.0	63.9	53.7	77.1	79.0	82.0	81.0	80.5	73.7
	DE	9.4	13.1	10.2	6.6	5.9	12.5	9.0	6.9	8.3
Glass	**PG**	70.0	64.5	64.5	67.5	69.0	66.5	66.0	69.0	66.0
	DE	5.3	6.5	11.0	7.1	7.6	8.6	6.0	9.8	8.4
Iris	**PG**	96.0	95.3	94.0	94.0	96.0	96.0	96.0	96.7	93.3
	DE	1.5	8.4	2.8	3.7	1.5	1.5	1.5	2.4	4.1
Vehicle	**PG**	64.2	61.0	60.9	64.2	65.1	65.2	63.2	63.1	61.1
	DE	1.8	3.7	2.6	1.3	2.3	2.6	2.4	2.1	2.2
Wine	**PG**	72.4	70.0	70.0	69.4	67.7	68.2	70.0	70.0	67.1
	DE	3.4	6.4	9.2	4.9	6.6	2.5	9.6	3.8	9.6
German	**PG**	65.2	66.4	64.1	64.4	62.9	65.0	66.9	65.5	64.7
	DE	2.6	3.2	2.0	3.2	3.3	2.8	1.8	3.3	2.1
Satimage	**PG**	83.6	83.4	83.0	83.4	82.5	83.1	83.0	82.8	82.9
	DE	11.6	11.5	12.3	12.1	12.0	12.0	11.7	121.4	11.7
Phoneme	**PG**	76.1	75.9	74.8	74.0	76.2	76.6	74.9	73.7	75.5
	DE	8.4	8.6	8.5	10.0	9.1	8.5	8.5	10.9	8.9
Waveform	**PG**	78.0	77.6	78.1	77.6	76.7	78.0	78.2	76.9	77.0
	DE	2.9	2.5	2.4	2.9	1.7	2.2	2.0	2.5	2.5
Segment	**PG**	94.8	93.2	93.7	93.2	93.1	92.5	92.1	93.0	92.8
	DE	1.4	1.6	1.5	0.8	2.2	1.0	1.5	1.2	1.9

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 19

Votación ponderada dinámica Shepard modificado (a =según el orden y β = 1, SMC de 7 y 9 clasificadores)

		CE original	7 Clasificadores				9 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	95.8	95.3	95.3	94.9	95.8	96.4	93.4	94.5
	DE	2.5	1.6	1.9	1.7	3.0	1.7	2.3	1.8	2.0
Heart	**PG**	58.2	62.6	61.5	55.9	54.1	58.9	59.6	60.7	58.9
	DE	6.2	5.3	4.8	5.8	4.0	3.8	9.3	3.2	7.9
Liver	**PG**	65.2	63.8	57.4	60.0	56.5	64.4	61.5	56.8	62.0
	DE	4.8	6.8	4.3	3.3	4.8	4.2	5.0	5.6	4.3
Pima	**PG**	65.9	67.7	66.0	64.7	66.8	66.8	66.0	65.8	65.1
	DE	5.2	1.0	2.8	4.6	3.0	5.4	4.8	5.0	5.4
Sonar	**PG**	82.0	82.9	82.0	80.0	75.1	75.1	77.1	75.1	75.1
	DE	9.4	10.5	11.8	7.6	5.6	9.4	16.0	10.7	6.1
Glass	**PG**	70.0	69.0	70.5	68.0	68.5	66.0	68.0	63.0	58.0
	DE	5.3	4.2	5.4	10.5	4.9	3.8	3.3	5.7	9.3
Iris	**PG**	96.0	96.0	96.0	94.0	96.7	96.7	96.0	94.7	94.7
	DE	1.5	1.5	1.5	4.4	2.4	2.4	1.5	3.0	3.0
Vehicle	**PG**	64.2	63.4	63.4	62.8	61.4	63.2	61.1	65.7	59.2
	DE	1.8	2.0	2.0	1.9	4.3	2.5	2.8	0.8	3.4
Wine	**PG**	72.4	71.2	74.1	70.6	61.8	68.2	74.1	71.8	65.3
	DE	3.4	4.8	8.7	6.6	7.8	4.4	10.9	4.9	10.7
German	**PG**	65.2	67.2	65.4	66.5	64.2	64.3	64.4	66.1	65.8
	DE	2.6	2.5	4.1	3.8	3.0	1.9	2.7	1.7	1.1
Satimage	**PG**	83.6	82.9	83.2	82.2	82.8	82.7	82.5	82.5	81.9
	DE	11.6	11.6	12.4	12.2	12.4	12.3	12.4	12.2	12.0
Phoneme	**PG**	76.1	75.5	75.1	74.0	74.6	75.7	75.5	73.2	73.9
	DE	8.4	8.9	9.9	10.3	10.3	10.0	8.7	10.9	11.2
Waveform	**PG**	78.0	77.9	78.2	77.0	76.3	77.7	78.1	76.3	76.2
	DE	2.9	1.8	2.2	2.5	3.3	2.5	2.1	3.1	1.6
Segment	**PG**	94.8	92.9	92.9	92.1	90.2	93.3	92.9	91.8	88.7
	DE	1.4	2.0	1.1	2.4	3.1	1.3	1.7	2.0	1.8

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 19

Votación ponderada dinámica Shepard modificado (a =según el orden y β = 1, SMC de 15 y 25 clasificadores)

		CE original	15 Clasificadores				25 Clasificadores			
			A	B	C	D	A	B	C	D
Cancer	**PG**	95.6	95.5	96.1	91.2	93.9	96.1	95.5	91.0	93.3
	DE	2.5	3.2	2.7	3.4	2.9	2.2	2.4	1.9	2.4
Heart	**PG**	58.2	58.2	58.5	54.4	54.1	57.0	60.0	54.4	57.0
	DE	6.2	5.7	8.6	7.2	7.6	6.9	3.8	8.3	8.8
Liver	**PG**	65.2	64.4	59.4	63.2	56.2	63.5	62.6	57.1	61.7
	DE	4.8	5.0	6.2	2.6	7.0	4.7	4.7	4.4	6.5
Pima	**PG**	65.9	66.9	65.5	63.9	63.3	65.2	67.1	65.1	62.6
	DE	5.2	5.0	4.4	6.7	1.8	8.6	5.6	2.6	5.1
Sonar	**PG**	82.0	82.9	80.0	74.6	69.3	78.5	81.5	74.6	68.3
	DE	9.4	8.1	7.4	8.2	7.2	9.4	10.4	9.9	5.2
Glass	**PG**	70.0	66.0	63.5	60.0	64.0	66.5	64.5	50.0	37.5
	DE	5.3	6.0	10.1	9.8	3.8	8.0	4.8	---	---
Iris	**PG**	96.0	96.0	95.3	93.3	96.0	96.0	95.3	94.0	92.7
	DE	1.5	1.5	3.0	6.2	2.8	1.5	1.8	4.4	4.4
Vehicle	**PG**	64.2	61.9	61.7	54.7	56.1	64.4	65.7	---	50.1
	DE	1.8	2.0	2.0	2.9	8.5	3.5	9.0	---	4.9
Wine	**PG**	72.4	73.5	71.2	65.9	68.2	71.8	71.8	70.0	62.4
	DE	3.4	4.7	8.4	6.1	7.3	3.4	8.2	5.7	8.4
German	**PG**	65.2	65.7	64.2	63.7	63.2	62.8	65.5	64.4	61.7
	DE	2.6	2.5	3.4	2.8	2.2	4.7	3.2	1.4	3.7
Satimage	**PG**	83.6	83.1	83.3	81.8	81.8	82.5	83.2	81.9	80.5
	DE	11.6	11.0	12.7	12.1	11.1	12.4	11.6	11.9	12.8
Phoneme	**PG**	76.1	75.2	75.5	73.1	73.8	75.8	76.1	73.8	73.1
	DE	8.4	9.0	9.3	10.6	10.3	9.6	9.5	9.5	7.9
Waveform	**PG**	78.0	77.8	77.9	76.6	75.6	77.6	78.4	76.0	75.0
	DE	2.9	2.9	2.4	1.4	1.7	3.0	1.9	2.1	2.4
Segment	**PG**	94.8	93.2	93.2	90.4	85.8	93.6	93.6	87.0	83.5
	DE	1.4	1.2	1.3	1.7	1.3	1.4	2.0	3.5	3.1

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 20

Votación ponderada dinámica según el orden (SMC de 3 clasificadores)

		CE original	3 Clasificadores			
			A	B	C	D
Cancer	**PG**	95.6	95.9	65.8	72.4	99.3
	DE	2.5	2.5	4.8	13.1	0.9
Heart	**PG**	58.2	61.1	54.1	47.8	57.4
	DE	6.2	3.7	4.8	5.6	3.5
Liver	**PG**	65.2	62.6	53.0	49.3	59.4
	DE	4.8	4.0	0.8	7.0	3.7
Pima	**PG**	65.9	68.2	62.1	60.9	70.1
	DE	5.2	3.9	1.7	1.9	6.1
Sonar	**PG**	82.0	78.1	62.4	53.7	73.7
	DE	9.4	6.5	12.9	4.6	10.6
Glass	**PG**	70.0	71.0	62.0	61.5	66.0
	DE	5.3	4.9	10.4	6.3	7.4
Iris	**PG**	96.0	96.7	95.3	94.7	94.7
	DE	1.5	2.4	1.8	3.0	3.0
Vehicle	**PG**	64.2	64.5	62.3	60.8	62.8
	DE	1.8	1.2	3.8	2.2	0.8
Wine	**PG**	72.4	70.0	68.8	64.7	66.5
	DE	3.4	7.0	7.7	2.1	1.6

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 21

Votación ponderada dinámica según Dudani (SMC de 3 clasificadores)

		CE original	3 Clasificadores			
			A	B	C	D
Cancer	**PG**	95.6	95.6	68.5	74.2	94.9
	DE	2.5	2.5	6.9	9.1	1.6
Heart	**PG**	58.2	58.2	53.0	47.4	58.5
	DE	6.2	6.2	2.1	5.0	1.0
Liver	**PG**	65.2	65.5	56.2	52.2	60.9
	DE	4.8	4.4	2.4	4.7	4.2
Pima	**PG**	65.9	68.4	59.1	60.3	67.6
	DE	5.2	3.4	2.6	3.4	6.3
Sonar	**PG**	82.0	82.0	65.4	55.1	77.1
	DE	9.4	9.4	16.5	5.3	6.6
Glass	**PG**	70.0	70.0	67.0	65.0	66.5
	DE	5.3	5.3	4.8	9.0	5.8
Iris	**PG**	96.0	96.7	96.0	94.7	94.0
	DE	1.5	2.4	1.5	3.0	3.7
Vehicle	**PG**	64.2	64.2	61.0	60.9	64.2
	DE	1.8	1.8	3.7	2.6	1.3
Wine	**PG**	72.4	72.4	70.0	65.3	69.4
	DE	3.4	3.4	6.4	3.8	4.9

A = Selección aleatoria sin reemplazo
B = Bagging
C = Boosting
D = Arc-x4

Anexo 22

Votación ponderada estática con método leave-one-out (SMC de 3 clasificadores)

		CE original	3 Clasificadores			
			A	B	C	D
Cancer	**PG**	95.6	96.9	66.4	72.1	94.2
	DE	2.5	2.0	6.0	12.6	1.0
Heart	**PG**	58.2	65.2	50.4	45.2	59.6
	DE	6.2	4.2	5.3	3.8	4.2
Liver	**PG**	65.2	63.8	57.1	50.1	62.0
	DE	4.8	3.7	3.9	3.3	5.5
Pima	**PG**	65.9	68.9	59.4	60.0	70.1
	DE	5.2	3.3	1.9	3.6	6.2
Sonar	**PG**	82.0	79.0	62.0	49.8	75.1
	DE	9.4	7.0	8.7	5.1	9.4
Glass	**PG**	70.0	68.5	56.0	60.5	63.0
	DE	5.3	7.3	10.3	7.4	10.4
Iris	**PG**	96.0	96.0	95.3	93.3	96.0
	DE	1.5	1.5	1.8	3.3	2.8
Vehicle	**PG**	64.2	63.7	62.9	59.8	61.0
	DE	1.8	3.4	3.0	1.8	2.6
Wine	**PG**	72.4	68.2	70.6	64.7	62.9
	DE	3.4	5.7	7.5	7.8	4.5

A = Selección aleatoria sin reemplazo (mayoritaria)
B = Bagging
C = Boosting
D = Arc-x4

Anexo 23

Patrones por clase

	CE original	3 Clasif.	5 Clasif.
Cancer	355, 191	119, 63	71, 38
Heart	120, 96	40, 32	24, 19
Liver	116, 160	38, 53	23, 32
Pima	400, 215	133, 71	80, 43
Sonar	89, 78	29, 26	17, 15
Glass	56, 61, 14, 11, 8, 24	18, 20, 4, 3, 2, 8	11, 12, 2, 2, 1, 4
Iris	40, 40, 40	13, 13, 13	8, 8, 8
Vehicle	175, 170, 174, 159	58, 56, 58, 53	35, 34, 34, 31
Wine	48, 57, 39	16, 19, 13	9, 11, 7
German	560, 240	187, 80	112, 48
Satimage	1226, 562, 1086, 501, 566, 1206	409, 187, 362, 167, 189, 402	245, 112, 217, 100, 113, 241
Phoneme	1268, 3054	423, 1018	254, 611
Waveform	1326, 1317, 1357	442, 439, 452	205, 263, 271
Segment	264 c/u	88 c/u	53 c/u

	7 Clasif.	9 Clasif.	15 Clasif.	25 Clasif.
Cancer	51, 27	39, 21	23, 12	14, 8
Heart	17, 14	13, 10	8, 6	5, 4
Liver	17, 23	12, 17	7, 10	5, 6
Pima	57, 31	44, 23	26, 14	16, 9
Sonar	13, 11	9, 8	5, 5	4, 3
Glass	8, 9, 2, 2, 1, 3	6, 7, 2, 1, 1, 3	4, 4, 1, 1, 1, 2	2, 2, 1, 1, 1, 1
Iris	6, 6, 6	4, 4, 4	3, 3, 3	2, 2, 2
Vehicle	25, 24, 25, 23	19, 18, 19, 17	11, 11, 11, 10	7, 7, 7, 6
Wine	7, 8, 6	5, 6, 4	3, 4, 3	2, 2, 2
German	80, 34	62, 27	37, 16	22, 10
Satimage	175, 80, 155, 72, 81, 172	136, 63, 121, 56, 63, 134	82, 37, 72, 33, 38, 80	49, 22, 43, 20, 23, 48
Phoneme	181, 436	141, 339	85, 204	51, 122
Waveform	198, 188, 194	147, 146, 151	88, 88, 90	53, 53, 54
Segment	38 c/u	29 c/u	18 c/u	11 c/u

Nota: la cantidad de patrones por cada una de las clases esta separada por coma. Por ejemplo, la base de datos Cancer original tiene dos clases, la primera de ellas consta de 355 patrones y la segunda de 191.

GLOSARIO DE TÉRMINOS

Glosario de términos

Algoritmo de aprendizaje. Mecanismo definido de reglas para la solución de un problema de aprendizaje.

Algoritmo de edición. Método utilizado para “limpiar” el conjunto de entrenamiento. En su funcionamiento, estos algoritmos eliminan los patrones atípicos que afectan los índices de precisión obtenido por el clasificador.

Algoritmo de reducción. Algoritmos mediante los cuales se busca obtener un subconjunto representativo en el que se descartan únicamente aquellos patrones que no afectan (o afecten poco) los resultados de clasificación.

Aprendizaje. Proceso de adaptación a un entorno mediante el cual se proporciona a la regla de decisión elementos (conocimiento) para realizar un adecuado reconocimiento de patrones.

Aprendizaje estocástico. Este tipo de aprendizaje consiste básicamente en realizar cambios aleatorios en los valores de los pesos del sistema y evaluar su efecto a partir del objetivo deseado y de distribuciones de probabilidad. De tal manera que, posterior a cada ajuste de pesos se evalúa el desempeño del sistema, si aumenta se establece el cambio, de lo contrario, se acude a distribuciones de probabilidad preestablecidas y en función de éstas, se acepta o se rechazan los cambios.

Aprendizaje no supervisado. Estos métodos, no requieren de influencia externa para aprender. Es decir, no reciben ninguna información por parte del entorno

(como el conjunto de categorías en las que se dividen los casos conocidos) que le indique al sistema si la salida generada en respuesta a una determinada entrada es o no es correcta. Dentro de este grupo destacan los algoritmos de clustering.

Aprendizaje parcialmente supervisado. Estos algoritmos, en su etapa inicial disponen de un grupo de patrones etiquetados y un segundo grupo de patrones sin etiquetar. De tal forma que, para su tratamiento, requieren combinar tanto métodos de aprendizaje supervisado, como métodos de aprendizaje no supervisado.

Aprendizaje por corrección de error. Proceso iterativo en el cual, se presentan al sistema de clasificación un conjunto de pares de datos que representan la entrada y la salida deseada para tal entrada. Durante el aprendizaje, se realizan ajustes en función del error observado en iteraciones previas.

Aprendizaje por refuerzo. A diferencia del aprendizaje por corrección de error, este tipo de aprendizaje no dispone de la salida deseada para cada entrada, únicamente se conoce el comportamiento general que el sistema debería tener ante diferentes entradas. Durante el entrenamiento se dispone de una *señal de refuerzo* que mide el funcionamiento del sistema e indica la acción a seguir: refuerzo en caso de acierto o disminución en caso de error.

Aprendizaje supervisado. Estos un algoritmos son capaces de discriminar datos entre un conjunto de posibles categorías (datos previamente etiquetados). De tal forma que al entrena y probar el clasificador con un número suficiente de casos conocidos, se convierte en un generalizador y es capaz de clasificar correctamente datos provenientes de casos desconocidos. En este grupo se encuentran los algoritmos de aprendizaje por corrección de error, por refuerzo y estocástico.

Arcing. (Adaptively Resample and Combine) Métodos de submuestreo que construyen subconjuntos de forma secuencial teniendo en consideración el error observado por cada una de ellas. Algunos métodos que utilizan este principio son Boosting y sus variantes, y Arc-x4, entre otros.

Atributo o característica. Se denomina de esta forma a las propiedades descriptivas de un patrón. Estas pueden ser numéricas (discretos o continuos) o no numéricas, y representan cada una de las características del patrón, por ejemplo estatura, edad, color de piel, color de ojos, medidas, etc.

Clase o categoría. Grupos de patrones que guardan un alto grado de similitud entre si y un alto grado de disimilitud entre los patrones de otros grupos. Generalmente las clases representan la (s) solución (es) en que se divide el problema.

Clasificación. Proceso que consiste en proporcionar nuevos patrones al sistema de reconocimiento, independientes a los utilizados en el aprendizaje, para que éste los etiquete utilizando el conjunto de clases o grupos de patrones disponibles.

Conjunto de entrenamiento. Conjunto de objetos previamente etiquetados y separados en clases utilizados durante el proceso de aprendizaje supervisado.

Conjunto de test. Se llama de esta forma al conjunto de patrones utilizados para evaluar la eficiencia del clasificador posterior a la etapa de aprendizaje supervisado.

Clustering. Métodos también conocidos como clasificación no supervisada de patrones. El sistema automáticamente extrae las características diferenciadoras entre clases y establece las fronteras entre grupos de puntos.

Desbalance. Término utilizado para definir un problema presente en el conjunto de datos, en el cual, una o varias de las clases (llamada minoritaria) está menos representada con respecto al número de patrones pertenecientes a otras clases (llamadas mayoritarias). Generalmente, en estas bases de datos la clase minoritaria es la de mayor interés.

Desviación estándar. Técnica estadística que permite analizar la dispersión existente en los datos, con objeto de tener una visión más acorde a la realidad al momento de describirlos e interpretarlos para la toma de decisiones.

Dimensionalidad. Longitud del vector que alberga un patrón. Esta longitud esta dada por la cantidad de características utilizadas para describir dicho patrón.

Escalabilidad de algoritmos. Se denomina de esta forma a las técnicas y métodos desarrollados para hacer posible el procesamiento eficiente de grandes volúmenes de datos.

Espacio de representación. Espacio de dimensionalidad determinada por el número de atributos o características consideradas para describir un patrón, en el cual pueden representarse todos y cada uno de los patrones del conjunto de datos.

Estimación de error. Métodos utilizados para calcular la proporción de patrones clasificados de forma incorrecta por un clasificador.

Inteligencia artificial. Bajo este concepto se agrupan todas las tecnologías que estudian y desarrollan sistemas que simulan procesos de la inteligencia humana. Estos procesos incluyen el aprendizaje (la adquisición de información y reglas para usarla), el razonamiento (uso de las reglas para llegar a conclusiones definitivas o aproximadas) y autocorrección.

Media geométrica. Medida de tendencia central, ampliamente utilizada para análisis de datos en busca de incrementos o decrementos en una serie de resultados.

Métrica. Regla definida para calcular la distancia entre dos puntos en el espacio.

Neurona artificial. Unidad de procesamiento de información fundamental para la operación de una RNA.

Normalización. Proceso de estandarización, mediante el cual los valores de los atributos de un patrón son ajustados dentro de un mismo rango de valores.

Patrón atípico. Se llama de esta forma a aquellos patrones que generalmente se encuentran ubicados cerca de las fronteras de decisión y que tienen diferencias significativas con el resto de patrones de su misma clase debido a su etiquetado incorrecto al momento de formar el conjunto de entrenamiento.

Patrón de entrenamiento. Representación de objetos de la vida real, comúnmente realizado mediante un vector con dimensión limitada a la cantidad de atributos que describen al objeto.

Patrón ruidoso. Es aquel patrón que puede confundir al clasificador debido a que guarda cierto parecido con objetos de otras clases.

Peso sináptico. En RNA se llama peso sináptico al valor ponderado de la sinapsis existente entre dos neuronas artificiales.

Ponderación. Proceso mediante el cual se establece el grado de influencia que un elemento tendrá en la solución de un problema.

Potencial de activación: Valor de entrada aplicado a una neurona artificial, dado por la suma de las señales de entrada que parten de las neuronas conectadas a ella (salidas de éstas), ponderada por las sinapsis de la neurona.

Preprocesado. Tratamiento realizado al conjunto de entrenamiento, previo a la etapa de clasificación. Mediante este tratamiento se busca mejorar la calidad del conjunto de datos, o eliminar algún problema relacionado con los datos. Entre los algoritmos más utilizados para este fin se encuentran, los que realizan limpieza, aumento o decremento de tamaño de los conjuntos de entrenamiento.

Reconocimiento de patrones. Rama de la inteligencia artificial que estudia la operación y el diseño de sistemas que permitan extraer similaridades y coincidencias de un conjunto de objetos y ayude a establecer propiedades de o entre los datos.

Red neuronal artificial (RNA). Son programas de Inteligencia Artificial capaz de simular algunas de las funciones de aprendizaje del ser humano. Sin reglas convencionales, una red neuronal obtiene experiencia analizando automática y sistemáticamente una cantidad de datos, para determinar reglas de comportamiento. En base a estas reglas, puede realizar predicciones sobre nuevos casos.

Redundancia. Situación existente en algunos conjuntos de entrenamiento, en los cuales un mismo patrón aparece varias ocasiones en una o varias de las clases existentes.

Solapamiento. Problema del conjunto de datos en el cual las fronteras de dos o más clases están intersectadas.

Submuestreo. (En esta Tesis) Métodos utilizados para lograr diversidad en un sistema múltiple, el cual consiste en la extracción de subconjuntos de datos, a partir de una base de datos.

Teorema de Bayes. En la teoría de la probabilidad, el teorema de Bayes, es el resultado que da la distribución de probabilidad condicional de una variable aleatoria A dada B en términos de la distribución de probabilidad condicional de la variable B dada A y la distribución de probabilidad marginal de sólo A.

Referencias Bibliográficas

[And., 73] M. R. Anderberg: *Cluster analysis for applications*. Academic Press, New York, 1973.

[And., 77] J. A. Anderson, J. W. Silverstein, S. A. Ritz, R. S. Jomnes: "Distinctive features, categorical perception and probability learning: some applications of a neural model". *Psychological Review*, (84), 1977, pp. 413-451.

[Ali., 96] K. M. Ali, M. J. Pazzani: "Error reduction through learning multiple descriptions", *Machine Learning* 24 (3), 1996, pp. 173 – 202.

[Bai., 93] T. L. Bailey, C. Elkan: "Estimating the accuracy of learned concepts", in *Proceedings of the International Joint Conference on Artificial Intelligence*, (IJCAI'93), Chambery (France), 1993. pp. 895–900.

[Ban., 03] B. E. Banfield, L. O. Hall, K. W. Bowyer, W. P. Kegelmeyer: "A new ensemble diversity measure applied to thinning ensembles", in *Proceedings of* 4th *International Workshop on Multiple Classifier Systems (MCS 2003), Lecture Notes in Computer Science*, 2003, pp 306 – 316.

[Bah., 00] D. Bahler, L. Navarro: “Methods for combining heterogeneous sets of classifiers”, in *Proceedings of 17th National Conference on Artificial Intelligence (AAAI-2000), Workshop on new research problems for Machine Learning*, 2000.

[Bar., 95] R. Barandela: “Una metodología para el reconocimiento de patrones en la solución de tareas geólogo-geofísicas”, *Geofísica Internacional* 34 (4), 1995. pp. 399 – 405.

[Bar., 01a] R. Barandela, J. S. Sánchez, V. García, E. Rangel: “Fusion of techniques for handling the imbalanced training sample problem”, in *Proc. of 6th Simposium Iberoamericano de Reconocimiento de Patrones*, Brasil, 2001.

[Bar., 03a] R. Barandela, J. S. Sánchez, R. M. Valdovinos: “New applications of ensembles of classifiers”, *Pattern Analysis and Applications* 6, 2003, pp. 245 – 256.

[Bar., 03b] R. Barandela, J. S. Sánchez, V. García, F. J. Ferri: “Learning from imbalanced sets through resampling and weighting”, in *Proceedings of 1st Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2003), Lecture Notes in Computer Science,* Puerto de Andratx, Mallorca, (Spain), 2003, pp. 80 -88

[Bar., 04] R. Barandela, R. M. Valdovinos, J. S. Sánchez, F. J. Ferri: “The imbalanced training sample problem: Ander or over sampling?”, in *Proceedings of 5th. International Workshop on Statistical Pattern Recognition,* Lisbon (Portugal), 2004, pp. 806 – 814.

[Bar., 05a] R. Barandela, F. J. Ferri, J. S. Sánchez: “Decision boundary preserving prototype selection for nearest neighbor classification”, *International Journal of Pattern Recognition and Artificial Intelligence* 19 (6), 2005, pp. 787-806.

[Bar., 05b] R. Barandela, J. K. Hernández, J. S. Sánchez, F. J. Ferri: “Imbalanced training set reduction and feature selection through genetic optimization”, in *Artificial Intelligence Research and Developments, Frontiers in Artificial Intelligence and Applications* 131, 2005, pp. 215-222.

[Bau., 04] C. Bauckhage, C. Thurau: “Towards a Fair’n Square Aimbot Using Mixture of Experts to Learn Context Aware Weapon Handling”, in *Proccedings of (GAME-ON’04)*, Ghent, Belgium, 2004, pp. 20 – 24.

[Bis., 95] C. M. Bishop: *Neuronal Networks for Pattern Recognition*, Clarendon Press, Oxford, 1995.

[Bre., 96] L. Breiman: "Bagging predictors", *Machine Learning* 26 (2), 1996, pp. 123 - 140.

[Bre., 98] L. Breiman: "Arcing classifiers", *The Annals of Statistics* 26 (3), 1998, pp. 801 – 849.

[Car., 90] C. S. Carman, M. B. Merickel : "Supervising ISODATA with an information theoretic stopping rule", *Pattern Recognition* 23 (1-2), 1990, pp. 185 – 197.

[Cha., 00] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer: "SMOTE: synthetic minority over-sampling technique", *Journal of Artificial Intelligence Research* 16, 2000, pp 321 - 357.

[Che., 01] D. Chen, X. Cheng: "An asymptotic analysis of some expert fusion methods", *Patter Recognition Letters* 22, 2001, pp. 901 – 904.

[Con., 98] W. J. Conover, *Practical Nonparametric Statistics*. 3rd ed. New York: John Wiley and Sons, 1998.

[Cor., 01] F. J. Cortijo: "Introducción al reconocimiento de formas", http://www-etsi2.ugr.es/depar/ccia/rf/www/tema1_00-01_www/tema1_00-01_www.html, Octubre 2001.

[Cot., 95] M. Cottrell, B. Girard, Y. Girard, M. Mangeas, C. Muller: "Neural Modelling for Time Series: A Statistical Stepwise method for Wight Elimination", *Neural Networks*, 6 (6), pp. 1355 – 1364, 1995.

[Cov., 67] T. M. Cover, P. E. Hart: "Nearest neighbour pattern classification", *IEEE Trans. on Information Theory* 13, 1967, pp. 21-27.

[Das., 91] B. V. Dasaraty: *Nearest Neighbor (NN) Norms: NN Pattern classification techniques*, Ed. IEEE Computer Society press, Los Alamitos, CA, 1991.

[Dec., 95] K. M. Decker, S. Focardi: "Technology overview: a report on data mining", in *CSCS TR-95-02 Technical Report*, Swiss Scientific Computing Center, mayo 1995, pp. 1-29.

[Dem., 93] H. Demuth, M. Beale: "Neural Networks Toolbox for Use with MATLAB: User's guide", Natick, MA: *The Math Works*, Inc, 1993.

[Die., 95] T. G. Dietterich, G. Bakiri: "Solving multiclass learning problems via error-correcting output codes", *Journal of Artificial Intelligence Research* 2, 1995, pp. 263-286.

[Die., 97] G. T. Dietterich: “Machine learning research: four current directions”, *AI Magazine* 18 (4), 1997, pp. 97 – 136.

[Die., 98] W. R. Dietrich, J. Hornegger: *Applied Patter Recognition* 2a Edición, Editorial Vieweg, 1998, pp. 5 – 10.

[Dom., 99] P. Domingos: “Metacost: a general method for making classifiers cost – sensitive”, in *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1999, pp. 155 – 154.

[Dud., 76] S. A. Dudani: “The distance weighted k-nearest neighbor rule”, *IEEE Trans. on Systems, Man and Cybernetics* 6, 1976, pp. 325-327.

[Dud., 00] R. O. Duda: *Pattern Classification*, 2nd Edition, Wiley Interscience, 2000.

[Dud., 73] R. O. Duda, P. E. Hart: *Pattern classification and scene analysis*, Jhon Wiley and Sons, 1973.

[Ega., 75] J. P. Egan: *Signal Detection Theory and ROC Analysis*, Academic Press, New York, 1975.

[Fay., 96] U. M. Fayyad: “Data mining and knowledge discovery: making sense out of data”, *IEEE Expert* 11 (5), 1996, pp. 20 - 25.

[Fis., 91] D. H. Fisher, M. J. Pazzani, P. Langley: *Concept formation: Knowledge and experience in unsupervised learning*. Morgan Kaufmann, San Mateo, California, 1991.

[Fix., 51] E. Fix, J. Hodges: “Discriminatory analysis, nonparametric discrimination: consistency properties”, Technical Report 4, Project No. 21-49-004, USAF School of Aviation Medicine, Randolph field, Texas, 1951.

[Fre., 96] Y. Freund, R.E. Schapire: “Experiments with a new boosting algorithm”, in *Proceedings of the 13th International Conference on Machine Learning*, Morgan Kaufmann, 1996, pp. 148 -156.

[Fri., 00] M. Friedman, A. Kendel: “Introduction to pattern recognition, statistical, structural, neural and fuzzy logic approaches”, series in *Machine Perception Artificial Intelligence* 32, Ed. World Scientific, 2000, pp. 1 - 6.

[Fu., 82] K. S. Fu: *Sintactic Pattern Recognition and Application*, New Jersey: Prentice-Hall, Inc., 1982.

[Fuk., 90] K. Fukunaga: *Introduction to Statistical Pattern Recognition*, San Diego, CA. Eds. Academic Press, 1990.

[Fun., 89] K. Funahashi: "On the Aproximate Realization of Continuous Mapping by Neural Networks", *Neural Networks* 2, 1989, pp. 183 – 192.

[Gas., 01] E. Gasca, L. Mora, R. Barandela: "Feature Selection for Multilayer Perceptron", in *Proceedings of the 6th Simposio IberoAmericano de Reconhecimento de Padröes (SIARP'01)*, 2001.

[Gar., 02] V. García: "Adaptación de técnicas de preprocesamiento para tratar el problema de las muestras de entrenamiento desbalanceadas". Tesis de Maestría en Ciencias, en Ciencias Computacionales, Instituto Tecnológico de Toluca, Metepec Estado de México, México. Diciembre, 2002.

[Gia., 97] G. Giacinto, F. Roli: "Adaptive selection of image classifiers", in *Proceedings of 9th International Conference on Image Analysis and Processing (ICIAP '97)*, Florence (Italy) 1997, pp. 7 - 19.

[Gia., 99] G. Giacinto, F. Roli: "Methods for dynamic classifier selection", in *Proceedings of 10th International Conference on Image Analysis and Processing (ICIAP '99)*, Venice (Italy), 1999, pp. 659-664.

[Gia., 01] G. Giacinto, F. Roli, G. Vernazza: "Methods for designing multiple classifiers systems", in *Proceedings of International Workshop on Multiple Classifier Systems (MCS'22001), Lecture Notes in Computer Science,* Cambridge (UK), 2001, pp. 78 – 87.

[Gia., 01b] G. Giacinto, F. Roli: "Design of effective neural network ensembles for image classification purposes". *Image Vision Comput,* 19 (9-10), 2001, pp. 699-707.

[Gor., 99] A. D. Gordon: *Classification*, 2nd Edition, Chapman and Hall,1999.

[Gro., 87] S. Grossberg: "Competitive learning: from interactive activation to adaptative resonance", *Cognitive Science*, (11), 1987, pp. 23 - 63.

[Han., 90] L. K. Hansen, P. Salomon: "Neural network ensembles", *IEEE Trans. on Pattern Analysis and Machine Intelligence* 12, 1990, pp. 993-1001.

[Har., 68] P. E. Hart: "The condensed nearest neighbor rule", *IEEE Trans. on Information Theory* 6 (4), 1968, pp. 515 – 516.

[Har., 04] P. Hartono, S. Hashimoto: "Ensemble of Linear Perceptrons with Confidence Level Output", in *Proceedings of the 4th International*

Conference on Hybrid Intelligent Systems (HIS'04), 2004, pp. 186 – 191.

[Heb., 49] D.O. Hebb: *Organization of behavior*, New York: Science Editions, 1949.

[Hil., 95] J. R. Hilera, V. J. Martínez: *Redes Neuronales Artificiales, fundamentos modelos y aplicaciones*, Madrid: RA-MA, 1995.

[Hon., 98] L. Hong, A. Jain: "Integrating faces and fingerprints for personal identification", *IEEE Trans. on Pattern Analysis and Machine Intelligence* 20 (12), 1998, pp. 1295-1307.

[Ho., 92] T. K. Ho, J. J. Hull, S. N. Srihari: "Combination of decisions by multiple classifiers", *Structured Document Image Analysis*, 1992, pp. 188 – 202.

[Ho., 00] T. K. Ho: "Complexity of Classification Problems and Comparative Advantages of Combined Classifiers", in *Proceedings of the 1st International Workshop on Multiple Classifier Systems*, New Jersey (USA), 2000, pp. 97 - 106.

[Hop., 82] J. L. Hoplfield: "Neural networks and Physical Systems with Emergent Collective Computational Abilities", in *Proceedings of the National Academy of Science USA*, (79), 1982, pp. 2554-2558.

[Hor., 89] K. Hornik, M. Stinchcombe, H. White: "Multilayer Feedforward Networks are Universal Approximators", *Neural Networks* 2, 1989, pp. 359 – 366.

[Jac., 91] R. Jacobs, M. Jordan, G. Hinton: "Adaptive Mixture of Local Experts", *Neural Computation*, 3 (1), 1991, pp. 79 – 87.

[Jai., 97] A. Jain, D. Zongker: "Feature slection: evaluation, application and sample performance", *IEEE Trans. on Pattern Analysis and Machine Intelligence* 19 (2), 1997, pp. 153-158.

[Jap., 00] N. Japkowicz, T. Eavis: "A recognition-based alternative to discrimination-base multilayer perceptrons", in *Proceedings of the AAAI-2000 Workshop on Learning from Imbalanced Data sets*, 2000.

[Kan., 97] I. Kanellopoulos, G. Wilkinson: "Strategies and Best Practice for Neural Networks Image Classification", *International Journal of Remote Sensing* 18 (4), pp. 711 – 725, 1997.

[Koh., 88] T. Kohonen: "Self-organized formation of topologically correct feature maps", *Biological Cybernetics*, (43), pp. 59 – 69, 1982.

Reprinting in: J. Anderson, E. Rosenfeld (Eds.), *Neurocomputing: Foundations of Research*, Cambridge, Massachusetts: M.I.T. Press, 1988.

[Koh., 95] R. Kohavi: "A study of cross-validation and Bootstrap for overall accuracy estimation and model selection", *Appears in the International Joint Conference on Artificial Intelligence* (IJCAI), 1995.

[Kol., 91] J. F. Kolen, J. B. Pollack: "Back propagation is sensitive to initial conditions", in *Advances in Neural Information Processing Systems*, San Francisco, CA., 1991, pp. 860 - 867.

[Kub., 00] M. Kubat, M. Cooperson Jr.: "Voting Nearest-Neighbor Subclassifiers", in *Proceedings of the 17th International Conference on Machine Learning*, Morgan Kaufmann, Stanford, CA (2000), pp. 503-510.

[Kun., 00] L. I. Kuncheva: "Clustering-and-selection model for classifier combination", in *Proceedings of the 4th International Conference on Knowledge-Based Intelligent Engineering Systems. Allied technologies (KES'200)*, Brighton (UK), 2000.

[Kun., 01a] L. I. Kuncheva, J. C. Bezdek, R. P. W. Duin: "Decision templates multiple classifier fusion: an experimental comparison", *Pattern Recognition* 34 (2), 2001, pp. 299 - 314.

[Kun., 01b] L. I. Kuncheva, C. J. Whitaker: "Ten measures of diversity in classifier ensembles: limits for two classifiers", in *IEEE Workshop on Intelligent Sensor Processing*, Birmingham, 2001.

[Kun., 01c] L. I. Kuncheva: "Using measures of similarity and inclusion of multiple classifier fusion by decision templates", *Fuzzy Sets and Systems* 122 (3), 2001, pp. 401 - 407.

[Kun., 01d] L. I. Kuncheva, C. J. Whitaker: "Feature subsets for classifer combination: an enumerative experiment", in *Proceedings of International Workshop on Multiple Classifier Systems (MCS'22001), Lecture Notes in Computer Science*, Cambridge, 2001, pp. 228 -237.

[Kun., 02a] L. I. Kuncheva, Roumen K. Kountchev: "Generating classifier outputs of fixed accuracy and diversity", *Pattern Recognition letters* 23, 2002, pp. 593 – 600.

[Kun., 02b] L. I. Kuncheva: "Switching between selection and fusion in combining classifeiers: an experiment", *IEEE Trans. on Systems Man and Cybernetics, Part B-Cybernetics* 32 (2), 2002, pp 1467 – 156.

[Kun., 02c] L. I. Kuncheva, C. J. Whitaker: “Using diversity with tree variants of Boosting: aggressive, conservative and inverse”, in *Proceedings MCS 2002*, Cagliari (Italy), 2002, pp. 81 – 90.

[Kun., 03] L. I. Kuncheva, C. J. Whitaker: “Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy”, *Machine Learning* 51 (2), 2003, pp 181 – 207.

[Kun., 05] L. I. Kuncheva: “Using diversity measures for generating error-correcting output codes in classifier ensemble”, *Pattern Recognition Letters* 26, 2005, pp. 83 – 90.

[Man., 05] P. Mantero, G. Moser, S. B. Serpico: Partially supervised classification of remote sensing images through SVM-based probability density estimation, *IEEE Trans. on Geoscience and Remote Sensing* 43, 2005, pp. 559-570.

[Mat., 96] O. Matan: “On voting ensembles of classifiers”, in *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI-96), Workshop on Integrating Multiple Learned Models*, 1996, pp. 84 – 88.

[McC., 43] W. S. McCulloch, W. Pitts: “A logical calculus of the ideas immanent in nervous activity”, *Bulletin of Mathematical Biophysics* (5), 1943, pp. 115 – 133.

[McC., 88] J. L. McClelland, D. E. Rumelhart : *Explorations in Parallel Distributed Processing. A Handbook of Models. Programs and Exercises*. Cambridge, MA: MIT Press, 1988.

[Mer., 98] C. J. Merz, P. M. Murphy: *UCI Repository of Machine Learning Databases*, Dept. of Information and Computer Science, University of California, Irvine (CA), 1988. http://www.ics.uci.edu/~mlearn

[Mic., 00] E. Micheli – Tzanakou: *Supervised and Unsupervised Pattern Recognition*, Editor Industrial Electronics series, 2000.

[Min., 69] M. L. Minsky, S. Papert: *Perceptrons: An introduction to Computacional Geometry*. Cambridge, MA: MIT Press, 1969.

[Nar., 05] A. Narasimhamurty: “Evaluation of diversity measures for binary classifiers ensembles”, in *Proceedings of 6th International Workshop on Multiple Classifier Systems (MCS2005)*, Monterey, California, 2005, pp. 13-15.

[Nat., 97] R. Nath, B. Rajagopalan, R. Ryker: "Determining the Saliency of Input Variables in Neural Networks", in *Neural Network Classifiers, Journal of Computers Ops Res.* 24 (8), pp. 767 – 773, 1997.

[Ort., 96] M. R. Ortiz, J. F. Martínez, J Ruiz: "A new approach to differential diagnosis of diseases", *Int J Biomed Comput*. 40 (3), 1996, pp 179-85.

[Pao., 89] Y. H. Pao: *Adaptive Pattern Recognition and Neural Networks*, MA: Addison- Wesley, 1989.

[Pap., 01] S. Papadimitriou, S. Mavroudi, L. Vladut: "Ischemia detection whit a self-organizing map supplemented by supervised learning", *IEEE Trans. on neural networks* 12 (3), May 2001.

[Par., 96] B. Parmanto, P. W. Munro, H. R. Doyle: "Improving committee diagnosis with resampling techniques", in *Advances in Neural Information Processing Systems*, Cambridge (UK), 1996, pp. 882—888.

[Pat., 89] H. H. Pattee: "Simulations realizations and theories of life", in C. Langton (Ed.), *Artificial Life Sta. Fe Inst. Sci. Complexity* 6, Addison-Wesley Pub., 1989, pp. 63-78.

[Rap., 91] P. E. Rapp, M. A. Jiménez-Montaño, R. J. Langs, L. Thomson, A. I. Mees: "Toward a quantitative characterization of patient-therapist communication", *Mathematical Biosciences* 105, 1991, pp 207-227.

[Rav., 96] Y. Raviv, N. Intrator: "Bootstrapping with noise: an effective regularization technique", *Connection Science*, 8, 1996, pp. 356 - 372.

[Rod., 01] M. A. Rodríguez, I. Cortázar, D. Tapias, J. Relaño: "Estado del arte en tecnologías de voz", in *Comunicaciones de telefónica Investigación y Desarrollo* 20, Madrid, 2001.

http://www.tid.es/presencia/publicaciones/comsid/esp/20/8XX.PDF

[Ros., 59] R. Rosemblatt: *Principles of neurodynamics*. New York: Spartan Books, 1959.

[Rum., 86] D. E. Rumelhart, J. L. McClelland, P. R. Group: *Parallel Distributed Processing, Explorations in the Microstructure of Cognition.* Cambridge, MA: MIT Press, 1986.

[Rus., 99] G. C. Russell, G. Kass: *Assessing the Overall Accuracy of Remotely Sensed Data: Principles and Practices*, New York, Washington, D. C., Ed. Lewis Publishers, 1999, pp. 43 – 64.

[San., 98] J. S. Sánchez, F. Pla, F. J. Ferri. “Improving the k-NCN classification rule through heuristic modifications”, Pattern Recognition Letters 19 (13), 1998, pp. 1165-1170.

[San., 02] J. S. Sánchez, R. Barandela, F. J. Ferri: “On filtering the training prototypes for nearest neighbour classification”, in *Topics in Artificial Intelligence*, 2002, pp. 239 – 248.

[San., 03] J. S. Sánchez, R. Barandela, A. J. Márquez, R. Alejo, J. Badenas: “Analysis of new techniques to obtain quality training sets”, *Pattern Recognition Letters* 24 (7), 2003, pp. 1015 – 1022.

[Sch., 97] C. Schittenkopf, G. Deco, W. Brauer: “Two Strategies to Avoid Overfitting in Feedforward Networks”, *Neural Networks* 10 (3), pp. 505 – 516, 1997.

[Sha., 99] J. Shah, C. Poon: “Linear Independence of Internal Representations in Multilayer Perceptrons”, *IEEE Trans. on Neural Networks* 10 (1), pp. 10 – 18, 1999.

[She., 87] R. N. Shepard: “Towards a universal law of generalization for psychological science”, *Science* 237, 1987, pp. 1317 – 1323.

[Shi., 02] C. A. Shipp, L. I. Kuncheva: “Relationships between combination methods and measures of diversity in combining classifier”, *Information Fusion* 3 (2), 2002, pp 135 – 148.

[Sri., 94] S. N. Srihari, .K. Ho, J. J. Casco: “Decision combination in multiple classifiers systems”, *IEEE Trans. on Pattern Analysis and Machine Intelligence* 16 (1), 1994, pp. 66 - 75.

[Ste., 91] J. Steven, E. Geoffrey, E. Hilton: “Evaluation of Adaptive Mixtures of Competing Experts”, *Advances in Neural Information Processing Systems*, 3. R. P. Lippman, J. E. Melody, D. S. Touretzky (Eds.), Morgan Kaufman: San Mateo Ca. 1991.

[Sum., 99] J. Sum, C. Leung, G. Young, W. Kan: “On the Kalman Filtering Method in Neural Network Training and Pruning”, *IEEE Trans. on Neural Networks* 10 (1), pp. 161 – 166, 1999.

[The., 98] S. Theodoridis, K. Koutroumbas: *Pattern Recognition*, Academic Press, 1998.

[Tru., 79] G. V. Trunk: “A problem of dimensionality: a simple example”, *IEEE Trans. on Pattern Analysis and Machine Intelligence* 1, 1979, pp. 306-307.

[Tom., 76] I. Tomek: "Two modifications of CNN", *IEEE Trans. on Systems. Man and Cybernetics* 6, 1976, pp. 769 – 722.

[Val., 02a] R. M. Valdovinos: "Nuevas aplicaciones de los SMC en el aprendizaje supervisado". Tesis de Maestría en Ciencias, en Ciencias Computacionales, Instituto Tecnológico de Toluca, Metepec, Estado de México (México), Octubre 2002.

[Val., 02b] R. M. Valdovinos, R. Barandela: "Sistema de múltiples clasificadores, una alternativa para la escalabilidad de algoritmos", in *Proceedings of 9th Internacional Conference of Research on Computer Science* (*CIICC02)*, Puebla (México), 2002.

[Val., 05a] R. M. Valdovinos, J. S. Sánchez, R. Barandela: "Dynamic and static weighting in classifier fusion", in *Proceedings of 2nd Iberian Conference of Pattern Recognition and Image Analysis*, Lisbon (Portugal), 2005, pp. 59-66.

[Val., 05b] R. M. Valdovinos, J. S. Sánchez: "Class-dependant resampling for medical applications", in *4th International Conference on Machine Learning and Applications*, Los Angeles (USA), Diciembre 2005.

[Val., 06] R. M. Valdovinos, J. S. Sánchez: "Combining Multiple Classifiers with Dynamic Weighted Voting", in *18th International Conference on Pattern Recognition*, Hong Kong, 2006 (enviado, en revisión).

[Wid., 59] B. Widrow: "Adaptative sampled-data systems, a statistical theory of adaptation". *1959 IRE WESCON Convention Record*, part 4. New York: Institute of Radio Engineers, 1959.

[Wid., 60] B. Widrow, M. Hoff: "Adaptive Switching Circuits", *IREWESCON Convention Record*, Part 4, pp. 96 – 104, 1960. Reprinting in *Neurocomputing*, ed. J. Anderson, E. Rosenfeld, pp. 126 – 134, MIT Press, 1988.

[Wil., 72] D. Wilson: "Asymptotic properties of nearest neighbor rules using edited data sets", *IEEE Trans. on Systems, Man and Cybernetics* 2, 1972, pp. 408 – 421.

[Woo., 93] K. Woods, C. Doss, K, Bowyer, J. Solka, C. Priebe and W. Kegelmeyer: "Comparative evaluation of pattern recognition techniques for detection of microcalcifications in mammography", *International Journal of Pattern Recognition and Artificial Intelligence* 7 (6), 1993, pp. 1417 – 1436.

[Woo., 97] K. Woods, W. P. Kegelmeyer Jr., K. Bowyer: “Combination of multiple classifiers using local accuracy estimates”, *IEEE Trans. on Pattern Analysis an a Machine Intelligence* 19 (4), 1997.

[Zam., 99] R. Zaman, D. Wunsch II: “TD Methods Applied to Mixture of Experts for Learning 9x9 Go Evaluation Function”, in *Proceedings of the IEEE/INNS International Joint Conference on Neural Networks (IJCNN-99)*, 1999.

[Zav., 97] J. Zavrel: “An empirical re-examination of weighted voting for k-NN”, in *Proceedings of the 7th Belgian-Dutch Conference on Machine Learning (BENELEARN’97)*, Tilburg, 1897.

[Zha., 97] J. Zhang, Y. Yim, J. Yang: “Intelligent selection of instances for prediction functions in lazy learning algorithms”, *Artificial Intelligence Review* 11, 1997, pp. 175-191.

[Zhe., 97] H. Zhezue: “A fast clustering algoritm to cluster very large categorical data sets in data mining”. In *Workshop on Research on issues on data mining an knowledge discovery*, Tucson, Arizona, 1997.