VNiVERSiDAD
D SALAMANCA

Sistema genérico de razonamiento basado en casos (CBR) multi-clase como soporte al diagnóstico médico mediante técnicas de reconocimiento de patrones

Xiomara Patricia Blanco Valencia

Universidad de Salamanca
Facultad de ciencias
Departamento de informática y automática
Salamanca, España

2017

Sistema genérico de razonamiento basado en casos (CBR) multi-clase como soporte al diagnóstico médico mediante técnicas de reconocimiento de patrones

Xiomara Patricia Blanco Valencia

Tesis presentada como requisito para el grado de:
Doctor en informática y automática

Directores:
Ph.D. Juan Manuel Corchado
Ph.D. Diego Hernán Peluffo Ordóñez
Ph.D. Juan Francisco De Paz Santana

Tema de investigación:
Sistema de razonamiento basado en casos y máquinas de aprendizaje
Grupo de investigación:
BISITE

Universidad de Salamanca
Facultad de ciencias
Departamento de informática y automática
Salamanca, España

2017

Dr. Juan Manuel Corchado, Departamento de Informática y Automática de la Universidad de Salamanca.
Dr. Diego Hernán Peluffo Ordóñez, Facultad de Ingeniería en Ciencias Aplicadas de la Universidad Técnica del Norte, Ecuador.
Dr. Juan Francisco De Paz Santana, Departamento de Informática y Automática de la Universidad de Salamanca.

HACEN CONSTAR:

Que el trabajo de tesis titulado ”Sistema genérico de razonamiento basado en casos (CBR) multi-clase como soporte al diagnóstico médico mediante técnicas de reconocimiento de patrones”ha sido realizado y escrito bajo su supervisión por D.ª Xiomara Patricia Blanco Valencia, como requisito parcial para optar al titulo de doctor.

Y para que así conste a todos los efectos oportunos.

En Salamanca, a 28 de junio de 2017

Dr. Juan Francisco De Paz Santana
Dpto. Informática y Automática
Universidad de Salamanca

Dr. Juan Manuel Corchado
Dpto. Informática y Automática
Universidad de Salamanca

Dr. Diego Hernán Peluffo Ordóñez
Facultad de Ingeniería en Ciencias Aplicadas
Universidad Técnica del Norte, Ecuador

A mi esposo y a mi hija

Agradecimientos

Primero, agradezco a Dios por concederme todos los elementos necesarios para culminar mi trabajo de investigación.

Quisiera dedicar este trabajo a mi esposo, Raúl Abel Bravo Rabassa por su apoyo y comprensión, por creer en mí y junto a mi hija ser mi mayor fuente de inspiración.

A mis padres y hermanos por su confianza, y porque desde la distancia siempre estuvieron presentes con sus palabras de apoyo y expresiones de cariño, que alimentaron mi corazón permitiéndome seguir adelante.

Doy mis más sinceros agradecimientos a mi asesor y amigo, el profesor Diego Peluffo (compañero de lucha) por su compañía, por despertar mi interés en temas como el aprendizaje de máquina y su invaluable apoyo, de quien aprendí bastante en términos de mi formación académica y personal.

En particular, me gustaría dar las gracias al profesor Juan Francisco De Paz Santana por toda su asesoría y apoyo en el desarrollo del presente trabajo.

Agradezco también a mi asesor, el profesor Juan Manuel Corchado por brindarme los medios necesarios para facilitar el desarrollo de la presente investigación.

Me gustaría felicitar y agradecer a Andrés Castro, Mabel Ortega y Diana Viveros por el trabajo realizado, por su incansable entusiasmo, sus aportes contribuyeron a orientar y fortalecer la investigación.

Desde lo personal a Juan Carlos Alvarado Pérez por su amistad y apoyo. Sin su colaboración esto no habría sido posible.

A mis amigos, Ana Eddy Monsalve Torra, Karla Díaz-Granados, Samuel Robayo y Lola Bautista, su compañía desde la distancia, me dio la fuerza necesaria para seguir adelante en momentos difíciles durante el desarrollo del presente trabajo.

A mis suegros María del Carmen y Raúl por su paciencia y buena disposición.

Expresarle mi gratitud a los amigos de Sancti-Spíritus por sus oraciones, por estar en los buenos y malos momentos que viví durante este proceso. Y a todas las personas que, de alguna manera, contribuyeron al desarrollo de este trabajo, gracias.

Resumen

El aprendizaje a partir de la experiencia es un proceso que se da de forma natural en los seres humanos, y el conocimiento generado con dicho proceso se convierte en la base para establecer soluciones a problemas cotidianos. En el campo de la inteligencia artificial, específicamente en el área del aprendizaje de máquina, pretendiendo emular esta habilidad del ser humano, ha surgido la metodología denominada razonamiento basado en casos (CBR). El núcleo de un sistema de CBR es el caso, que denota usualmente una situación problema o experiencia previa, la cual ha sido capturada y aprendida, y puede ser reutilizada para resolver problemas futuros.

El ciclo de vida de un sistema basado en CBR consiste en cuatro etapas principales: Recuperación, donde se identifica el problema y se encuentran casos pasados similares al nuevo caso; adaptación, donde se sugiere una solución a partir de los casos recuperados; revisión, en la cual se evalúa la solución propuesta; y, finalmente, aprendizaje, donde se actualiza el sistema para aprender de la experiencia. Los sistemas de CBR han demostrado su alta aplicabilidad en el campo de la salud, específicamente en diagnóstico médico de forma que los síntomas representan el problema (nuevo caso) y, por tanto, la solución obtenida será el diagnóstico recomendado.

En el estado del arte de CBR aplicado a diagnóstico médico, se encuentran algunos estudios que principalmente se enfocan en mejoras de la etapa de recuperación. No obstante, aún existen problemas abiertos relacionados con la representación de los casos y la solución de problemas multiclase. En efecto, si la representación de los casos no es adecuada, los resultados de la recuperación no serán óptimos. Además, la mayoría de los sistemas de CBR han sido diseñados para resolver problemas biclase, limitando entonces la etapa de adaptación automática a dos únicas posibles soluciones (típicamente, normal o patológico), con lo cual dichos sistemas pierden la capacidad de categorizar el estado de una patología o de identificar diagnósticos diferenciales.

En este trabajo de tesis, se presenta una propuesta de sistema genérico de CBR para la identificación de múltiples casos diagnósticos usando etapas de recuperación y adaptación mejoradas. Para este propósito, se plantea SAM (Sistema de Adaptación Mejorada) que consiste en un sistema que utiliza dos clasificadores en cascada que mejora el desempeño de la clasificación de los pacientes enfermos. Dicha propuesta surge como resultado de un estudio comparativo de técnicas de representación de datos para obtener el vector de casos y de diferentes clasificadores multiclase en la etapa de adaptación. Además, como aporte significativo

de este trabajo, se desarrolla una interfaz que comunica al especialista las probabilidades de pertenencia del nuevo caso a cada uno de los posibles diagnósticos. Experimentalmente, se comprueba que SAM, usando dos clasificadores en cascada basados en *K*-NN y con una apropiada selección de características en el pre-proceso, genera resultados satisfactorios en términos de medidas de clasificación mientras provee al especialista de forma inteligible los resultados de la recuperación de casos.

Palabras clave

Clasificación en casacada, clasificación multiclase, diagnóstico médico, estimación de probabilidades, razonamiento basado en casos.

Abstract

Learning from experience is a process that occurs naturally in humans, and the knowledge generated by this process becomes the basis for solutions to everyday problems. In the field of artificial intelligence, specifically in the area of machine learning, aimed at emulating such ability, the methodology called case-based reasoning (CBR) has arisen. The core of a CBR system is the case, usually denoting a previous problem or experience, which has been captured and learned, and can be then reused to solve future problems.

The life cycle of a CBR-based system consists of four main stages: Recovery, wherein the problem is identified and past cases similar to the new case are found; Adaptation, wherein a solution is suggested from the recovered cases; Revision, in which the proposed solution is evaluated; And finally learning, wherein the system is updated to learn from experience. The CBR systems have demonstrated their high applicability in the field of health, specifically in medical diagnosis so that the symptoms represent the problem (new case) and, therefore, the solution obtained is to be the recommended diagnosis.

In the state of the art of CBR applied to medical diagnosis, there have been developed some studies mainly focusing on improvements of the recovery stage. Nonetheless, there are still some open issues related to case representation and multiclass problem solving. In fact, if the representation of the cases is not adequate, the results of the recovery stage are not expected to be optimal. In addition, most CBR systems have been designed to solve biclass problems, thereby limiting the automatic adaptation stage to two possible solutions (typically, normal or pathological). Then such systems are not able to categorize the condition of a pathology nor to identify differential diagnoses.
In this thesis, a proposal of a generic CBR system for the identification of multiple diagnostic cases using improved recovery and adaptation stages is presented. For this purpose, SAM (Improved Adaptation System) is proposed, which consists of a system that uses two cascade classifiers that improves the classification performance of ill patients. This proposal arises as a result of a comparative study of data representation techniques to obtain the case vector and different multiclass classifiers for the adaptation stage. In addition, as a significant contribution of this work, an interface is developed that communicates to the specialist the belonging probabilities of the new case to each of the possible diagnoses. Experimentally, it is verified that SAM -using two classifiers in cascade based on K-NN along with an appropriate selection of characteristics in the pre-process- generates satisfactory results in terms

of classification measures while providing the specialist with intelligible results of the case recovery.

Keywords

Case-based reasoning, estimation of probabilities, classification in cascade, medical diagnosis, multiclass classification.

Índice general

Índice de figuras

Índice de cuadros

List of Algorithms

Notación

Variables y funciones

Notación	Denominación	Descripción
$\boldsymbol{x}_i$	Vector de atributos o características representando el i-ésimo caso	$\boldsymbol{x}_i \in \mathbb{R}^D$
$\boldsymbol{X}$	Matriz de datos representando el conjunto de casos	$\boldsymbol{X} \in \mathbb{R}^{N \times D},\ \boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)^\top$
C	Número total de clases o categorías	$C \in \mathbb{N}$
$\hat{y}_i$	Etiquetas o asignación de clase del i-ésimo caso en problemas biclase	$y_i \in \{-1, 1\}$
$\hat{\boldsymbol{y}}$	Vector de etiquetas o asignación de clases en problemas biclase	$\hat{\boldsymbol{y}} \in \mathbb{R}^N$
y_i	Etiquetas o asignación de clase del i-ésimo caso en problemas multiclase	$y_i \in \{1, \ldots, C\}$
$\boldsymbol{y}$	Vector de etiquetas o asignación de clases en problemas multiclase	$\boldsymbol{y} \in \mathbb{R}^N$
$\widetilde{\boldsymbol{x}}$	Nuevo caso o caso problema	$\widetilde{\boldsymbol{x}} \in \mathbb{R}^D$
$\widetilde{y}$	Clase o etiqueta asignada al nuevo caso o caso problema	$\widetilde{y} \in \mathbb{R}^N$
$\boldsymbol{A}$	Matriz o conjunto de datos inicial	$\boldsymbol{A} \in \mathbb{R}^{N \times P},\ \boldsymbol{A} = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_N)^\top$

Acrónimos

Término	Descripción
CBR	Case-based reasoning (Razonamiento basado en casos)
ML	Machine Learning (Aprendizaje de máquina)
CFS	Correlation Feature Selection (Selección de características por correlación)
SAM	Sistema de Adaptación Mejorada

Parte I.

PRELIMINARES

1. INTRODUCCIÓN

El aprendizaje a partir de la experiencia es un proceso que se da de forma natural en los seres humanos, y el conocimiento generado con dicho proceso se convierte en la base para establecer soluciones a problemas cotidianos [1], [2], [3]. Pretendiendo emular esta habilidad del ser humano, surge el razonamiento basado en casos (CBR por su nombre en inglés - *Case Based Reasoning*). CBR es una metodología utilizada para el razonamiento en ordenadores que intenta imitar el comportamiento de un ser humano experto en la toma de decisiones con respecto alguna temática y aprender de la experiencia de casos pasados [4]. Técnicamente, el CBR es una metodología de desarrollo de sistemas inteligentes que actualmente, ha demostrado ser apropiada para aplicar estrategias de analogía en dominios poco estructurados y en aquellos donde la adquisición de conocimiento es difícil [1]. Por lo tanto, la elección de esta metodología es ideal para el desarrollo de sistemas de apoyo diagnóstico, particularmente, en dominios de alta complejidad conceptual como los servicios médicos [5] [6]. Hoy en día la tendencia en los sistemas de apoyo a las decisiones médicas es integrar a los sistemas de información hospitalaria (HIS) existentes con diferentes metodologías y tecnologías, capaces de cooperar de una manera transparente en relación con el usuario, y promover el intercambio de información entre las personas involucradas en el manejo de la enfermedad. Los proveedores de salud (y en ocasiones los pacientes) puede confiar en un servicio (en lugar de en una herramienta aislada), capaces de ayudarles y que les proporcione la información pertinente en el momento adecuado, en la forma correcta. De esta forma sólo queda la responsabilidad de una interpretación contextual y la evaluación de la información en sí misma [6] [7]. En general, en la práctica clínica, se puede comenzar con algunas experiencias iniciales (casos resueltos) y, subsecuentemente, utilizar estas experiencias para resolver un nuevo problema, esto conlleva algún tipo de ajuste en las soluciones y al enriquecimiento del conjunto de experiencias. El CBR es un proceso de razonamiento, que se ha aplicado en diferentes casos médicos [4] [8]. En [1] se menciona el hecho de que los seres humanos usamos casos pasados como modelos para aprender a resolver problemas, particularmente en un aprendizaje temprano. Otros resultados [1], indican que los expertos quienes conocen mucho sobre un tema en particular, pueden recordar hechos en su dominio de experiencia con mayor facilidad que los no expertos, ya que tienen pocos hechos para recordar. El núcleo de un CBR es el caso, que denota usualmente una situación problema; unas experiencia previas, la cual ha sido capturada y aprendida y puede ser reutilizada para resolver problemas futuros. En [3] han descrito un ciclo de vida del CBR que consiste en cuatro etapas principales: re-

cuperación, identificar el problema actual y encontrar casos pasados similares al nuevo caso; reutilización, usar los casos recuperados y sugerir una solución; revisión, evaluar la solución propuesta y retención, actualizar el sistema para aprender de la experiencia. En [7] sugieren trabajar en reducción de dimensiones junto con CBR, con el fin de mejorar los sistemas de datos cada vez más grandes, complejos e inciertos de los entornos clínicos. Por otro lado, la mayoría de los CBR construidos da como resultado un dato binario (normal o patológico), son pocos los casos que se encuentran con resultados multiclase. A menudo, con el aumento en el número de clases aumenta la complejidad y el coste computacional. Además, las dificultades en la clasificación pueden estar presentes solo para algunas clases [9] [10]. La presente investigación, propone trabajar en algunas etapas del CBR con el fin de realizar un proceso aprendizaje de casos complejos y multiclase. Para ello, se realiza la integración de tres áreas: Representación de datos, clasificadores multiclase y razonamiento basado en casos. Bajo el supuesto de que los datos son complejos (alta dimensión y estructura compleja), se propone utilizar en la etapa de pre-proceso técnicas adecuadas de representación de datos, es decir, de selección de variables o reducción de dimensión. Asimismo, en las etapas de recuperación y reutilización o adaptación, incorporar clasificadores multiclase y de acuerdo con la naturaleza de los clasificadores, obtener la probabilidad o valor de pertenencia del nuevo caso a cada una de las clases. Proporcinando una respuesta que ayuda al personal médico, que se enfrenta ante un nuevo caso, a tomar una mejor decisión.

1.1. Hipótesis del trabajo

La representación de los casos mediante selección de variables y/o reducción de dimensiones, así como la recuperación eficiente a partir de la integración del CBR convencional y clasificadores multiclase permitirá obtener un sistema robusto y eficiente para enfrentar los desafíos que las ciencias de la salud ofrecen a la comunidad científica. Específicamente, el sistema propuesto permitirá al usuario obtener diagnósticos de múltiples clases y más cercanos a la realidad de acuerdo con el análisis del histórico de pacientes.

1.2. Objetivos

1.2.1. Objetivo general

Diseñar un sistema genérico de razonamiento basado en casos para asistencia diagnóstica computarizada que permita representar e identificar múltiples casos diagnósticos.

1.2.2. Objetivos específicos

- Realizar un estudio comparativo de métodos de selección de variables y reducción de dimensión, considerando criterios de caracterización y separabilidad de clases, con el fin de realizar una representación adecuada de la base de casos.
- Proponer una estrategia de recuperación y adaptación multiclase basada en técnicas supervisadas de reconocimiento de patrones capaz de identificar casos de clases poco frecuentes y casos nuevos.
- Implementar un sistema completo de razonamiento basado en casos que integre etapas adecuadas de pre-procesado, recuperación y adaptación de casos de múltiples clases con el fin de recomendar eficientemente posibles diagnósticos a los especialistas.

1.3. Contribuciones de esta tesis

A continuación se menciona las contribuciones que el desarrollo de esta tesis doctoral podría aportar a las áreas de sistema basado en casos y máquinas de aprendizaje:

- Extensión de la metodología de razonamiento basado en casos para facilitar la inclusión de clasificadores multiclase en el proceso de recuperación y adaptación.
- Desarrollo de un esquema de representación de casos para problemas de diagnóstico médico usando combinación adecuada de métodos de selección de variables y/o reducción de dimensión que facilite la separabilidad entre las clases.
- Estimación de probabilidades o valores de pertenencia de cada nuevo caso con respecto a las clases aprendidas y pre-establecidas.
- Diseño de un sistema de CBR versátil y multiclase.
- Optimización de rutinas de programación para realizar de forma eficiente y eficaz la recuperación de casos.

1.4. Organización del documento

El manuscrito esta divido en 5 grandes partes, I Preliminares, II Contexto y estado del arte de los CBR y marco teórico de las máquinas de aprendizaje, luego viene la propuesta III en métodos, los comentarios finales en la sección IV , y V los Apéndices.
Dentro de cada una de estas partes se trabajan uno o dos capítulos de los 9 capítulos principales, los cuales se organizan de la siguiente manera:

- Capítulo 2, en este capítulo se presenta el estado del arte de los temas principales que se desarrollan en la presente investigación:

  CBR: Se presentan los resultados de una investigación realizada sobre el estado del arte de los sistemas basados en CBR aplicados al sector salud; tocando temas como el ciclo de vida de un sistema CBR, sistemas de CBR aplicados al sector salud en los años 2007 al 2016, ventajas y desventajas de la metodología.

  Aprendizaje de máquina: Se pueden observar ejemplos de diferentes técnicas de predicción, agrupamiento y clasificación aplicadas en sistemas CBR.

- Pre-proceso de datos capítulo 3, este capítulo esta dedicado a tratar temas que facilitan la clasificación, tales como, selección de atributos y reducción de dimensiones.

- Capítulo 4, marco teórico de las técnicas utilizadas en las diferentes pruebas del presente trabajo.

- Capítulo 5, considerado el capítulo principal de la presente investigación y en el cual se describe la propuesta de sistema de CBR aplicado a diagnóstico médico. Se muestra en detalle cada una de las técnicas utilizadas en cada fase del sistema.

- Los experimentos son descritos en el capítulo 6.

- Capítulo 7, una vez descritos los experimentos, se muestran y se discuten los resultados de cada uno de ellos.

- Finalmente el capítulo 8, donde el lector puede ver las conclusiones alcanzadas con el desarrollo del presente trabajo y los posibles trabajos futuros.

2. CONTEXTO Y ESTADO DEL ARTE

El objetivo principal de este trabajo es diseñar un sistema de razonamiento basado en casos para la asistencia diagnóstica. Para poder llevar a cabo este objetivo, es necesario realizar un estudio de los trabajos desarrollados en esta área en los últimos años. Este capítulo se centra en describir los hallazgos encontrados en la literatura sobre los sistemas de CBR aplicados a la medicina.

2.1. Razonamiento basado en casos

Al finalizar la sección anterior se mostró la organización del presente documento. En esta sección se desarrolla el segundo capítulo del documento, contexto y estado del arte de los sistemas de CBR y el marco teórico del aprendizaje de máquina.

El origen del CBR se puede remontar a la Universidad de Yale y el trabajo de Schank y Abelson en 1977 [11]. El CBR se inspira en la razón humana, es decir, resolver un problema mediante la aplicación de experiencias anteriores adaptadas a una situación actual.
El núcleo de un CBR es el caso, usualmente denota una situación problema; una experiencia previa, la cual ha sido capturada y aprendida, y puede ser reutilizada para resolver problemas futuros; se habla de caso pasado, caso previo, caso almacenado, o caso retenido. Un caso (una experiencia episódica) contiene normalmente un problema, una solución, y su resultado.
Dentro de las ventajas que se encuentran al utilizar el CBR están: La reducción de la tarea de adquisición de conocimiento, evitar errores del pasado, proveer flexibilidad en el modelado del conocimiento, realizar predicciones del éxito probable de una solución, aprender a través del tiempo, razonar en un dominio con poco conocimiento y con datos y conceptos imprecisos o incompletos, evitar repetir todos los pasos necesarios para llegar a una solución, proveer de significado a una explicación y reflejar el razonamiento humano [12].

2.1.1. Ciclo de vida de un sistema CBR

El ciclo de vida para resolver un problema utilizando el CBR, consiste principalmente en llevar a cabo las 4 fases descritas por Aamodt y Plaza en [13] [4]: Recuperar, reutilizar, revisar y retener. Lo primero que debe hacerse en un sistema basado en CBR es identificar la situación problema actual; luego encontrar un caso pasado similar al nuevo caso, usar el caso y sugerir una solución; evaluar la solución propuesta, y actualizar el sistema para aprender de la experiencia, como se muestra en la Figura 2.1

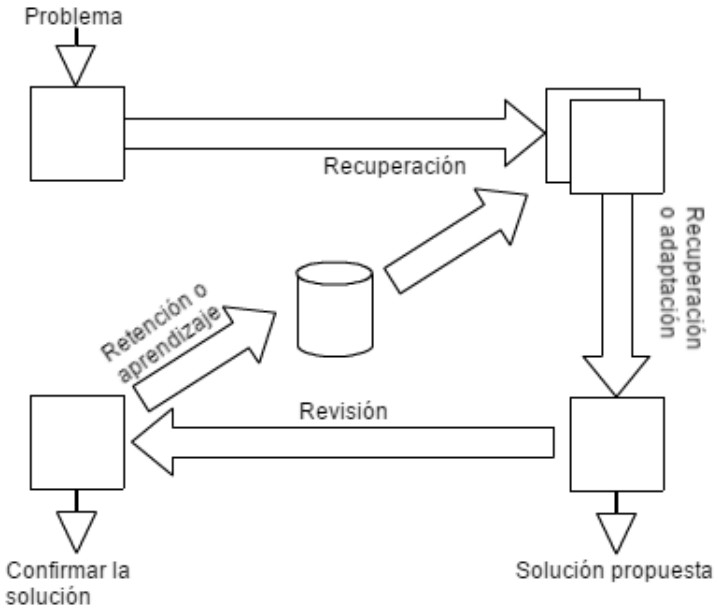


Figura 2.1.: Diagrama de las fases del CBR

En los últimos años se han utilizado técnicas de soft computing tales como: Lógica difusa, algoritmos genéticos y redes neuronales, con el fin de superar problemas existentes en los sistemas basados en CBR y los resultados han sido sorprendentes, los porcentajes de acierto son cada vez mayores.

Recuperación del caso más similar o casos:

En esta fase se pueden utilizar medidas de distancia euclídea (la técnica más usada) o se puede recurrir a técnicas más complejas como la aplicación de lógica difusa. Siendo esta última utilizada para sistemas más robustos, de forma que las características numéricas pueden ser convertidas a términos difusos y simplificar la comparación, los conjuntos difusos hacen más fácil la transferencia de conocimiento, adicionalmente permiten la utilización de modificadores para aumentar la flexibilidad en el caso recuperado [12].

Los algoritmos genéticos, inspirados en los principios biológicos de la selección natural y la genética; también han sido utilizados para resolver problemas de búsqueda y optimización [14, 15].
El uso de pesos locales y globales sobre los atributos de los casos permiten mejorar la eficiencia en la búsqueda de casos. Estos pesos indican la importancia de las características dentro de un caso con respecto a las características de la solución. La información sobre estos pesos puede mejorar el diseño de métodos de recuperación, y la exactitud de los sistemas CBR [12]

Reutilización o adaptación de la información:

Varias aplicaciones del mundo real, requieren una fase de adaptación después de la etapa de recuperación. El objetivo es identificar las diferencias entre el caso actual y el recuperado y analizar que puede ser transferido al nuevo caso.
Cuando se habla de problema resuelto, tiene un sentido muy amplio; el problema resuelto no significa necesariamente una solución concreta, por ejemplo, la justificación o la crítica a una solución propuesta por el usuario, la interpretación a una situación problema o la generación de un conjunto de posibles soluciones o poder generar expectativas en los datos observables, son también la solución a un problema.

En aplicaciones prácticas, el paso de la adaptación puede ser omitido y generalmente la solución de los casos recuperados es transferida directamente al nuevo caso como solución; alternativamente, la adaptación se deja al usuario final [16] [17].
Wike y Bergmann [18] presentaron tres tipos de adaptación:

- Adaptación transformacional: La solución del caso recuperado se transforma en una nueva solución para el nuevo problema. Este tipo de adaptación se realiza modificando, agregando o eliminando algunos elementos del resultado del caso o casos recuperados.

- Adaptación generativa: Implica la inserción de un solucionador de problemas en el sistema. Se requiere de un conocimiento previo que ayude a construir una solución desde cero.

- Adaptación composicional: Es la combinación de adaptación transformacional y generativa.

En [17] expone las metodologías que se han utilizado en la fase de adaptación, pueden dividirse en cuatro grupos:

- Adaptación basada en reglas: Existen reglas de adaptación que especifican como debe modificarse el valor de una característica bajo una situación determinada o cómo debe insertarse o eliminarse ciertas características en la representación de casos para adaptar el resultado al nuevo problema.

- Adaptación basada en casos: Se integran los expertos en el proceso de adaptación, como una forma natural de adquirir conocimiento. Las soluciones adaptadas por los expertos se añaden a la base de casos y el sistema genera nuevas reglas de adaptación.

- Recuperación guiada por adaptación: Se desarrolla para sistemas donde la influencia de cada atributo en la solución no es lineal en todo el rango de valores para ese atributo. La adaptación se realiza utilizando conocimientos contenidos en la propia base de casos. La fase de recuperación busca un caso similar al nuevo caso y los casos adecuados para la adaptación, y todos se usan para dar una respuesta adaptada.

- Adaptación basada en el aprendizaje automático: Se utilizan herramientas de aprendizaje de máquina para predecir los valores de variables numéricas basadas en los casos recuperados. La ventaja de utilizar esta metodología es que no depende del experto para generar reglas de adaptación.

Mientras que los dos primeros son enfoques intensivos en conocimiento, la recuperación orientada a la adaptación y la adaptación basada en el aprendizaje de máquina (ML por su nombre en inglés - *Machine Learning*) pueden clasificarse como *knowledge-light*.

Revisión de la solución obtenida:

Revisar si la solución propuesta es exitosa puede tomar muchas formas, dependiendo del dominio.
De acuerdo con [19], la revisión de casos comprende básicamente dos fases:

- Evaluar la solución: Decidir si la solución dada es la correcta al problema planteado. Esta fase normalmente se realiza por medio de algún método externo al sistema basado en CBR, generalmente la realiza un experto humano. En determinados casos la revisión puede se automática puesto que al aplicar la solución se conocen los resultados y por tanto se puede determinar el resultado final.

- Reparar los fallos: Si la solución no es aceptable, el caso debe ser modificado de nuevo [1]. La revisión de la solución de un caso generado por el proceso de adaptación es necesaria cuando la solución resulta incorrecta. Esto brinda la oportunidad de aprender del fracaso [16].

En esta fase, la lógica difusa también puede ser muy útil, logrando que el caso sea más flexible [6].

Retención o aprendizaje de la experiencia:

El nuevo caso se almacena para ser usado y resolver futuros problemas. Si la solución es correcta, se incorpora dentro de la base de casos como una solución exitosa, es necesario tener una política de actualización de la memoria de casos para evitar que crezca de manera indefinida. El sistema de CBR debe decidir si una nueva solución exitosa es lo suficientemente diferente de las ya existentes para garantizar el almacenamiento. Si se garantiza el almacenamiento, el sistema debe decidir cómo será indexado el nuevo caso, cuál será su nivel de abstracción, y donde será almacenado dentro de la organización de la base de casos [1]. En esta fase se pueden utilizar redes neuronales artificiales (ANNs por su nombre en inglés - *Artificial Neural Networks*), debido a que los dominios suelen ser complejos y la clasificación de casos en cada nivel es normalmente no lineal y por lo tanto cada clasificación puede requerir una red multicapa. Además trabajan muy bien con datos incompletos e imprecisos. Casos redundantes pueden ser eliminados utilizando reglas difusas [6].

2.1.2. Sistemas basados en CBR aplicados al sector salud

Una exploración temprana del CBR en el ámbito médico se llevó a cabo por Koton [20] y Bareiss [21] a finales de 1980. El dominio de las ciencias de la salud ofrece a la comunidad científica desafíos para ser llevados a cabo, ofreciendo una variedad de tareas complejas, que son difíciles de resolver con otros métodos y enfoques. El CBR es una metodología adecuada para explorar en un contexto médico, donde los síntomas representan el problema, y el diagnóstico o el tratamiento representan la solución. El CBR es un proceso de razonamiento aplicado en diferentes áreas y parece llamar cada vez más la atención [4], [8], [1]. Anderson ha demostrado como la gente usa casos pasados como modelos para aprender a resolver problemas, particularmente en un aprendizaje temprano. Otros resultados como el de Kolodner [1], indican que los expertos quienes conocen mucho sobre un tema en particular, pueden recordar hechos en el área que dominan con mayor facilidad que los no expertos, los no expertos tienen pocos hechos para recordar [1].
El aprendizaje a través de la experiencia ocurre naturalmente en los humanos y se usa ese conocimiento para resolver problemas diariamente. Hoy en día, se tiende a integrar los sistemas

de apoyo a la toma de decisiones dentro de los sistemas de información de los hospitales, con el fin de mejorar los servicios de salud; por lo tanto, la interpretación contextual y la evaluación de la información, es de interés en los últimos años [22]. Existen muchos sistemas de CBR enfocados al sector salud [23–26] y varios artículos estudian su evolución a través de los últimos años. Un análisis de sistemas de CBR publicados entre 1999 y 2003 fue dirigido por Nilsson y Sollenborn en [27], concluye que los sistemas híbridos entre CBR y otras técnicas de Inteligencia Artificial (IA) están aumentando, debido a que el dominio de aplicación es complejo y no puede ser manejado solo con la metodología de CBR. Otro estudio encontró un uso relativamente alto para este tipo de sistemas en el sector salud, pero concluye que todavía existe mucho trabajo por realizar. Estudios más recientes del CBR aplicados al sector salud [11], [28], [29] [30], concluyen que existen oportunidades para aplicar esta metodología, por ejemplo, Bichindaritz [19], habla del CBR como una metodología apropiada para el cuidado de ancianos y apoyo a discapacitados. Montani [7] enfatiza en tres áreas de mejora en la construcción de estos sistemas:

- Reducir el espacio de búsqueda en la fase de recuperación de casos, usando el conocimiento contextual almacenado, de esta manera el proceso de recuperación es más significativo.

- Mantener valido el conocimiento, e incorporar conocimiento en los casos para ayudar a revisar las soluciones propuestas.

- Trabajar en métodos de adaptación que tenga en cuenta restricciones locales, utilizando la información de contexto.

Otro estudio publicó las tendencias y desarrollos actuales de los sistemas basados en CBR, analizó artículos publicados entre 2004 y 2008, concluyendo que el CBR ha sido aplicado en diferentes escenarios médicos como apoyo al diagnóstico, la clasificación y el tratamiento [30], [20]. En años recientes el CBR está usando técnicas basadas en probabilidad e informática estadística, abriendo oportunidades prometedoras para mejorar los sistemas de datos cada vez más grandes y complejos, además de tener datos inciertos [7]. Otra conclusión del estudio es que la adaptación automática es una debilidad, especialmente en el campo médico [30].

Una revisión de literatura de sistemas basados en CBR aplicados al sector salud entre los años 2007 y 2016 realizada por la autora del presente trabajo en conjunto con un grupo de colaboradores se resume en la Tabla A.1 y en las figuras que se observan a continuación.

Se puede observar en la Figura 2.2 que la mayoría de los sistemas de CBR se han desarrollado para aplicaciones diagnósticas y un porcentaje muy pequeño se ha dedicado al tratamiento.

Tabla 2.1.: Regiones y dominio de aplicación.

Dominio	País
Diagnóstico	Taiwan, Uruguay-España, Libia, China, Singapur-Francia Suiza-Rumania, España, Corea del Sur, Botsuana-Nigeria, Alemania, Estados Unidos,Vienna-Austria, Portugal, Dubai, Arabia Saudita y Egipto
Tratamiento	Reino Unido, China, Suecia, Francia y Grecia

En la Figura 2.3 se aprecia una gran contribución del continente asiático. 15 de las 33 revisiones se realizaron en China y otros países de Asia, lo cual corresponden a un 45.45 %, seguido por Europa con un 39.39 % y África y América Latina con un 3.03 %.

El algoritmo de vecinos cercanos utilizando distancia euclidiana (ED) es el más utilizado en la etapa de recuperación. En el año 2009 el uso de técnicas combinadas incluye redes neuronales (NN), lógica difusa (FL), algoritmos genéticos, y máquinas de soporte vectorial (SM), entre otros. Esta combinación de técnicas fue ampliamente usada en los años 2011 y 2012, podemos ver como en los años 2013 al 2015 se retorna al uso de vecinos cercanos como técnica de recuperación de casos y se utilizan las ontologás para representación del conocimiento. En el año 2016 se encuentran trabajos que fusionan la etapa de recuperación y adaptación utilizando redes neuronales y bayes para tratamiento médico, como se puede observar en la Figura 2.4

La Figura 2.5 muestra la inexistencia del mecanismo de adaptación en la mayoría de los trabajos. Se utilizan algoritmos basados en reglas, adaptación manual, reglas fuzzy y algoritmos genéticos. En los últimos dos años las investigaciones han buscado construir mecanismos de adaptación que se fusionan con la fase de recuperación o se aplica la técnica de K-vecinos más cercanos para recuperar casos pasados y posteriormente se utiliza un algoritmo de adaptación.

Se puede observar en la Tabla A.1 que la mayoría de los trabajos no reporta la etapa de revisión. Esta etapa es manual, es decir, el experto es quien revisa la solución y da una

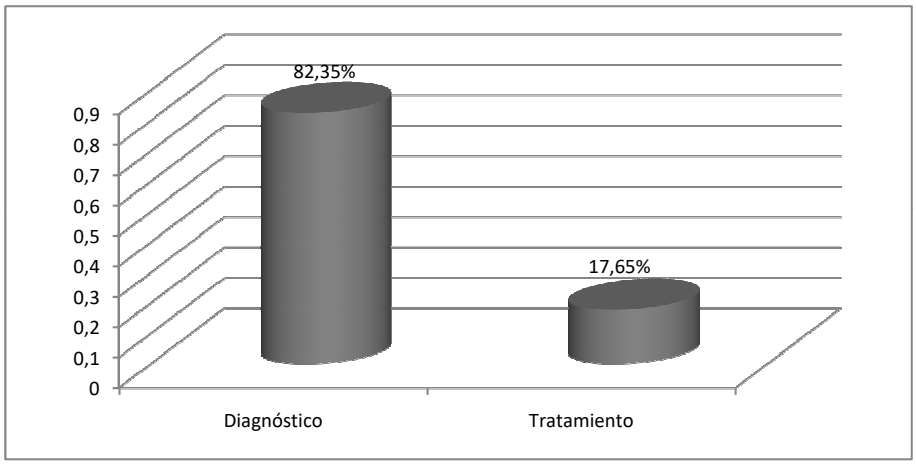


Figura 2.2.: Diagrama de frecuencias por dominio de aplicación.

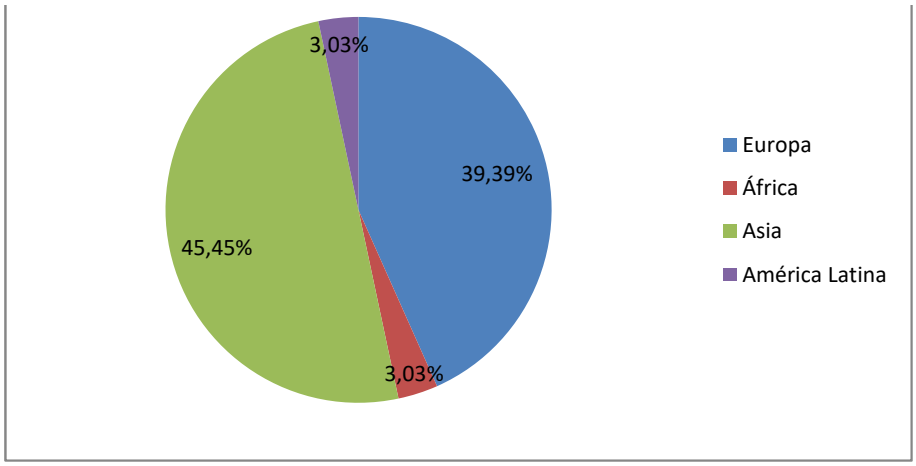


Figura 2.3.: Regiones y porcentaje de trabajos de investigación de sistemas de CBR aplicados al sector salud

respuesta positiva o negativa con el fin de que el sistema aprenda. Pocos trabajos realizan una revisión automática o semiautomática.

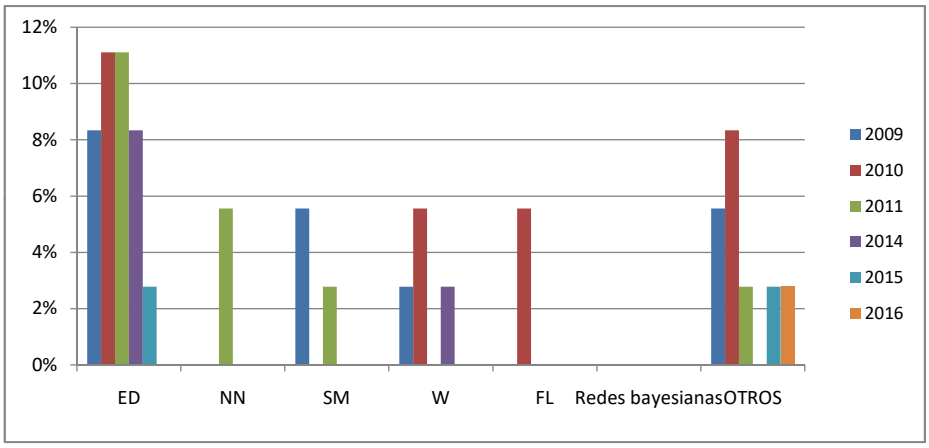


Figura 2.4.: Técnicas utilizadas en años recientes para la etapa de recuperación.

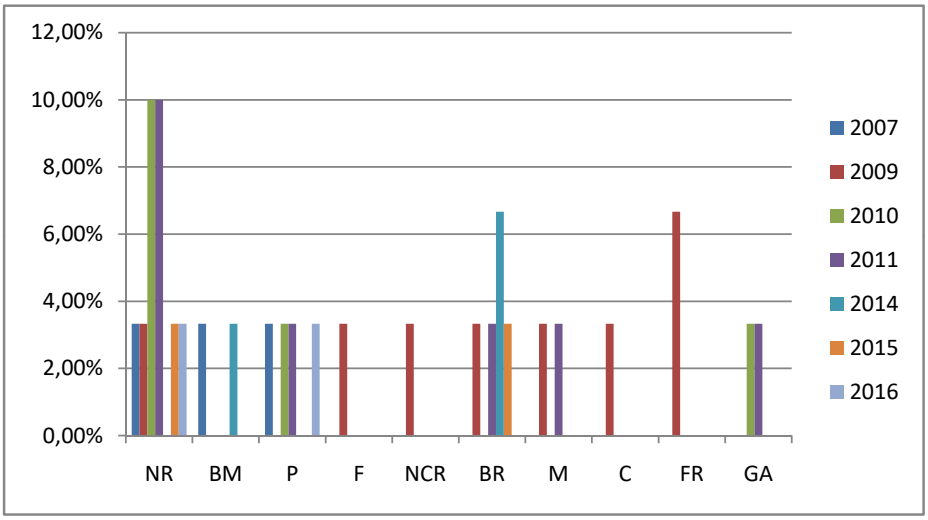


Figura 2.5.: Técnicas utilizadas en años recientes para el mecanismo de adaptación. No reporta(NR), mejor coincidencia (BM), probabilidad (P), marcos (F), reglas neuronales (NCR), basado en reglas (BR), manual (M), cluster (C), reglas fuzzy (FR) y algoritmos genéticos (GA).

Tabla 2.2.: Resultado de validación, Sensibilidad (Se) y Especificidad (Sp)

Título	**Num. Casos**	**Técnica de evaluación**	*Acc*	*Se*	*Sp*	**ROC**
Case-based reasoning support for liver disease diagnosis	166	10-fold cross-validation	93 95	91 98	96 94	93 96
Discovering interobserver variability in the cytodiagnosis of breast cancer using decision trees and Bayesian networks	692	10 fold cross - validation	93.49 (±0,93) 95.37 (±0,79) 94.07 (±0,89) 94.79 (±0,84) 94.65 (±0,85) 94.79 (±0,84)	89 (84-–93) 92 (88-–95) 87 (83-–92) 90 (86-–94) 89 (85-–93) 89 (85-–93)	96 (94-–98) 97 (96-–99) 98 (96-–99) 97 (96-–99) 97 (96-–99) 98 (96—99)	93.33 94.77 97.95 98.05 98.18 98.29
A reuse-based CBR system evaluation in critical medical scenarios	89	leave-one-out	73 71 75 71 68 76	85 79 79 61 89 83	58 59 70 87 37 66	
Influenza Forecast: Case-Based Reasoning or Statistics?	3569	statistical frequency	95	45 69 80 80	95 96.7 97.2 96.1	
Case-Based Reasoning Computer Algorithm that Uses Mammographic Findings for Breast Biopsy Decisions	500	statistical frequency		100 98	25 41	

Tabla 2.2 –

Título	**N Casos**	**Técnica de evaluación**	**Precisión %**	**Sn**	**Sp**	**ROC**
Automated assessment of myocardial SPECT perfusion scintigraphy: A comparison of different approaches of case-based reasoning	168	ROC	A 70 75 77 B 75 77 74	A 74 85 80 B 77 81 72	A 61 51 69 B 72 67 79	B 79 80 80
An intelligent decision support algorithm for diagnosis of colorectal cancer through serum tumor markers	578	Serial test	73.87	73.87	67.74	
A hybrid intelligent system for medical data classification		9:1 (10-fold cross-validation)	95.26 95.71 98.84	95.46 95.8 98.84	95.46 90.02 98.84	
Prediction of human breast and colon cancers from imbalanced data using nearest neighbor and support vector machines		10-fold cross validation	96.71 95.18 96.50	97.57 93.04 100	95.84 97.32 93.01	

Tabla 2.2 –

Título	**N Casos**	**Técnica de evaluación**	**Precisión %**	**Sn**	**Sp**	**ROC**
DERMA: A Melanoma Diagnosis Platform Based on Collaborative Multilabel Analog Reasoning		leave one out	92 96 95 98 100	86 93 91 94 100	89 97 96 99 100	

La sensibilidad y especificidad es reportada tan solo en 10 de los 33 artículos revisados. Los dos valores son muy buenos, superiores a 80 % y con un área bajo la curva mayor a 0.80.

2.2. Aprendizaje de máquina (ML)

A continuación se presenta una revisión bibliográfica de algunos trabajos relevantes de sistemas que aplican la metodología de CBR en el sector salud, considerando dos enfoques: supervisado y no supervisado. Además se dedica un apartado a la aplicacíón de clasificadores multiclase.

2.2.1. Clasificación supervisada

La clasificación supervisada en el CBR se utiliza principalmente como método de comparación, se compara la salida del sistema basado en CBR con la salida de una red neuronal y en algunos casos se propone un sistema híbrido entre la metodología de CBR y una técnica de clasificación supervisada, logrando con esto muy buenos resultados.
El estudio realizado en [31] para ayudar en el diagnóstico de enfermedad hepática, compara el resultado de clasificar utilizando cinco técnicas diferentes: Una ANN de retropropagación, árboles de regresión tipo CART (Classification And Regression Tree), regresión lineal, CBR con 10 vecinos cercanos recuperados utilizando distancia euclídea y un modelo CBR-ANN. El sistema híbrido CBR-ANN compara la respuesta que da el sistema de CBR con la salida de la red neuronal, si las respuestas son iguales, el resultado esta dado por la solución propuesta por sistema basado en el CBR, de lo contrario, cuenta con un sistema de reglas que toma la decisión dependiendo de la cantidad de casos sanos o enfermos recuperados.
De los cinco sistemas comparados, la mejor respuesta se obtiene con el sistema híbrido CBR-ANN.
En el trabajo [32], se investiga el diagnóstico de la meningitis aguda bacteriana. Se comparan los resultados de utilizar un clasificador bayesiano con un sistema de CBR que recupera casos similares con distancia euclídea. En la etapa de recuperación del sistema de CBR se evalúa la similitud entre el caso problema y un caso de la base de casos, si sobrepasa el umbral establecido, se reutiliza la solución sin cambios, si no es así, existen casos de adaptación previamente construidos y basados en la diferencia de los síntomas entre dos casos. El sistema basado en CBR obtiene la mejor respuesta, comparado con el clasificador bayesiano.
Al igual que en el trabajo anteriormente mencionado, el estudio [33] utiliza en paralelo un sistema de CBR y un clasificador bayesiano. Este sistema ayuda en la prescripción de medicamentos, utilizando la metodología de CBR para recuperar los casos más parecidos a un caso problema y en paralelo realiza una clasificación aplicando razonamiento bayesiano con los patrones de prescripción de experiencias previas. Se compara la respuesta de los dos sistemas a través de reglas IF-THEN, si los medicamentos son iguales en las dos respuestas,

esa es la solución; en caso contrario, pasa a otro sistema de reglas para calcular los medicamentos y las dosis correspondientes.
En [34] se propone un sistema genérico para diagnóstico médico y combina las redes neuronales de retropropagación, un sistema basado en reglas y la metodología de CBR para construir un híbrido que posteriormente se compara con la respuesta de una ANN. Se obtienen los mejores resultados con el sistema híbrido. El trabajo presenta una técnica novedosa a través de la cual utiliza las redes neuronales para generar un sistema de reglas. Se prueba con una base de casos para el diagnóstico de hepatitis, con 6 entradas se generan sesenta y cuatro reglas. Una vez generadas las reglas el sistema esta preparado para la entrada de un nuevo caso, se recuperan los vecinos más cercanos aplicando distancia euclídea y posteriormente se aplica el sistema de reglas generado por la red neuronal, y de esta forma brindar una respuesta.
El estudio realizado en [35] emplea en una primera fase las redes neuronales de retropropagación para clasificar al paciente como enfermo o sano. Si esta enfermo, utiliza la metodología de CBR para recuperar los casos más cercanos, posteriormente le asigna una probabilidad de pertenencia a una clase que calcula con los casos recuperados y la función de similitud de vecinos cercanos.
En [36], presenta un sistema basado en CBR que calcula la fracción de casos malignos de todos los casos recuperados para mostrar este dato al especialista. Se compara el resultado entregado por el sistema de CBR con el resultado dado por un clasificador basado en ANN, mostrando que la metodología de CBR obtiene los mejores resultados.
Para el diagnóstico de diabetes en [37] utilizan un clasificador basado en *K*-NN, el algoritmo de probabilidades bayesianas, C4.5 y lógica difusa para la selección de características. Posteriormente se realiza la representación del caso a través de la ponderación de características y ontologías. Se trasladan todos los datos a una base de datos relacional, se lleva a cabo un pre-proceso, se codifica y se fuzzifica. Para recuperar los casos combina el razonamiento de ontologías con una similitud difusa.
Petrovic en [17] recupera los casos utilizando distancia euclídea. En la fase de adaptación utiliza una ANN que tiene como entradas la diferencia de atributos entre dos casos y como salida tres clases 0, 1 y -1, que corresponden a los valores a tener en cuenta para adaptar la solución. Si es cero la solución no cambia, si es 1 se aumenta el número de haces de luz en el tratamiento de radioterapia y si es -1 se resta el número de haces. Con el clasificador de Naive Bayes se calcula la pertenencia a cada una de las clases.

2.2.2. Clasificación no supervisada

La técnica más utilizada en los sistemas basados en CBR es la de clúster para el pre-proceso de los datos, disminuyendo el esfuerzo en la fase de recuperación. También se utiliza la clasificación no supervisada para generar reglas de asociación y emplearlas en la fase de

adaptación.
En [38] se propone un sistema híbrido entre CBR y árboles de decisión difusos para aplicarse en diagnóstico médico. Se realizan pruebas con bases de datos de cáncer de mama y enfermedad del hígado. En una primera fase utilizan un análisis de regresión para ponderar los pesos de cada característica, este enfoque transforma la matriz de similitud en una matriz de equivalencia, con el fin de agrupar los casos equivalentes entre sí y clusterizar. Una vez se tienen los clúster, se aplica un árbol de decisión difuso a cada clúster para construir un sistema de reglas para la toma de decisiones.
El proceso de preparación de los datos lleva mucho tiempo en los sistemas de análisis de datos, por ejemplo, en [39] con el fin de agilizar el análisis, la base de casos se guarda en una base de datos relacional SQL, para su análisis posterior. Se entrenan un clasificador no supervisado *K* means con el fin de generar grupos y poder clasificar el caso problema. En la etapa de recuperación se traen los casos más cercanos de cada grupo y se muestra la información al usuario.

2.2.3. Clasificación supervisada y no supervisada

Se encuentran trabajos que utilizan clasificación supervisada y no supervisada en diferentes etapas del proceso del CBR, a continuación se mencionan algunos de ellos:
El trabajo realizado en [40], es un sistema de CBR construido para ayudar en el diagnóstico de leucemia a través del análisis de microarray. Clusteriza utilizando dendrogramas basados en el método de grupo de pares no ponderado con media aritmética UPGMA (Unweighted Pair Group Method with Arithmetic Mean) y realiza la extracción de conocimiento utilizando el algoritmo de CART para poder crear reglas que permitan dar una explicación al especialista de la decisión tomada.
En el estudio [41], utilizan la técnica de distancia de aprendizaje (DML), los casos parecidos están muy unidos y los casos positivos se distancian bastante de los negativos. Trabajan con técnicas adaptadas a multi-etiqueta y la trasladan al mundo del CBR. Existen dos subsistemas de CBR que por votación deciden cuál es la respuesta, con el fin de simular varios perfiles médicos, como se haría en la vida real. En la etapa de recuperación aplican distancia euclídea, luego calculan las probabilidades de ocurrencia, teniendo en cuenta los casos positivos y negativos recuperados.
Los autores del trabajo [42], crearon una base de conocimiento basada en ontologías del dominio ontocológico. La representación del caso se realiza por el componente de taxonomía de un modelo de ontología médica. Utilizan el razonamiento taxonómico del modelado de la ontología y la similitud semántica en la etapa de recuperación. La base de conocimiento esta dividida en clúster para hacer más rápida la búsqueda. La adaptación se realiza a través de un sistema de reglas, si no es posible, la realiza el experto manualmente.

2.2.4. Clasificación multiclase

Como se ha podido leer en parráfos anteriores, en los últimos años se han utilizado técnicas de aprendizaje de máquina para clasificar los casos y poder dar una respuesta más acertada. Sin embargo, en varios de estos trabajos la clasificación es biclase, se clasifica el caso como enfermo o sano [35, 43].
De los 33 casos revisados tan solo 4 soportan una clasificación multiclase. En [32], el sistema calcula las probabilidades acumuladas de las diferentes enfermedades, aplicando repetidamente el teorema de bayes sobre los síntomas y los signos. En [17], se entrena una red neuronal con la diferencia de las distancias entre atributos de dos casos y la salida es 0, 1 o -1 dando lugar a un tratamiento diferente. El problema con los sistemas multiclase desarrollados hasta el momento, se encuentra en que el entrenamiento es tedioso y consume muchos recursos computacionales, que se ven reflejados en el tiempo de respuesta. Además, una vez el sistema de CBR esta preparado para recibir nuevos casos, difícilmente se vuelve a entrenar, perdiendo con esto capacidad de aprendizaje.

2.2.5. Conclusiones

- Se demuestra una vez más que la metodología de CBR se sigue aplicando en muchas situaciones médicas para diversas tareas tales como diagnóstico y tratamiento.

- La investigación sobre CBR aplicada al sector salud está creciendo cada vez más. Es interesante ver como en la mayoría de los trabajos se expresa el interés de que sus prototipos sean utilizados por la comunidad científica, pero no se encuentran disponibles ni como software libre, ni como productos comerciales. En el año 2011, en [13] se llega a la misma conclusión.

- El uso de sistemas híbridos aplicando técnicas de aprendizaje de máquina, lógica borrosa y técnicas estadísticas permiten gestionar fácilmente las complejidades subyacentes a los datos médicos y obtener así un mejor desempeño.

- Basado en la evidencia observacional, la mayoría de los sistemas de CBR tienen bien definida la fase de recuperación de casos, pero existe una deficiencia en el mecanismo de adaptación como se puede observar en la Figura 2.5. Muchos de los estudios no hacen referencia a dicha fase y otros informan que se realiza manualmente.

- A través de los años se observa un comportamiento interesante: Artículos publicados entre 2009 y 2012 muestran una optimización en la fase de recuperación mediante la combinación de diferentes técnicas de IA con el fin de evitar la fase de adaptación. Los trabajos publicados entre el 2013 y el 2014 retoman el algoritmo de K-vecinos más cercanos para aplicar en la etapa de recuperación y se centran los esfuerzos en

optimizar la etapa de adaptación y dar así mejores respuestas. En los últimos años la tendencia es a fusionar la etapa de recuperación con la de adaptación.

- La mayoría de los sistemas basados en CBR que se revisaron, dan como respuesta al usuario los *K* casos recuperados y dejan en manos del especialista el análisis.
- Algunos sistemas han utilizado clasificadores biclase, donde los casos son etiquetados como enfermos o sanos, existen pocos trabajos que soporten la clasificación multiclase.

En el presente capítulo se trabaja el estado del arte de los sistemas de CBR aplicados al sector salud, se pueden ver los fundamentos teóricos y los algoritmos aplicados en cada una de las fases del sistema de CBR. Se presentaron gráficos y tablas que resumen los 33 estudios revisados, y por último se ilustra acerca de algunos trabajos que incorporan en la metodología de CBR algoritmos de aprendizaje de máquina. En el siguiente capítulo se muestran los fundamentos teóricos de la reducción de dimensiones y la selección de atributos.

Parte II.

MARCO TEÓRICO

3. PRE-PROCESO

En el capítulo anterior se revisa la literatura del CBR aplicado al sector salud desde el año 2007 hasta el año 2016. Se pudo concluir que se requiere realizar más investigación en esta área, principalmente, en sistemas CBR capaces de identificar múltiples diagnósticos. Por tanto, surge la necesidad de estudiar alternativas existentes dentro del campo del ML que realicen clasificación multiclase, para obtener un mejor resultado en la aplicación de éstos clasificadores es necesario realizar una etapa de pre-proceso.
Generalmente, los datos reales pueden ser ruidosos, inconsistentes o incompletos; y por tanto el subsecuente proceso de extracción de patrones puede ser erróneo o poco útil. Con el preprocesamiento se busca que los datos que van a ser utilizados en tareas de análisis o descubrimiento de conocimiento conserven su coherencia. Con frecuencia, el preprocesamiento de los datos tiene un impacto significativo en el desempeño general de los algoritmos de aprendizaje, aplicar algunas técnicas de preprocesamiento permite que sean más eficientes.
Las técnicas que comúnmente se utilizan son: Limpieza, reducción, integración, selección y transformación de datos.

3.1. Notación

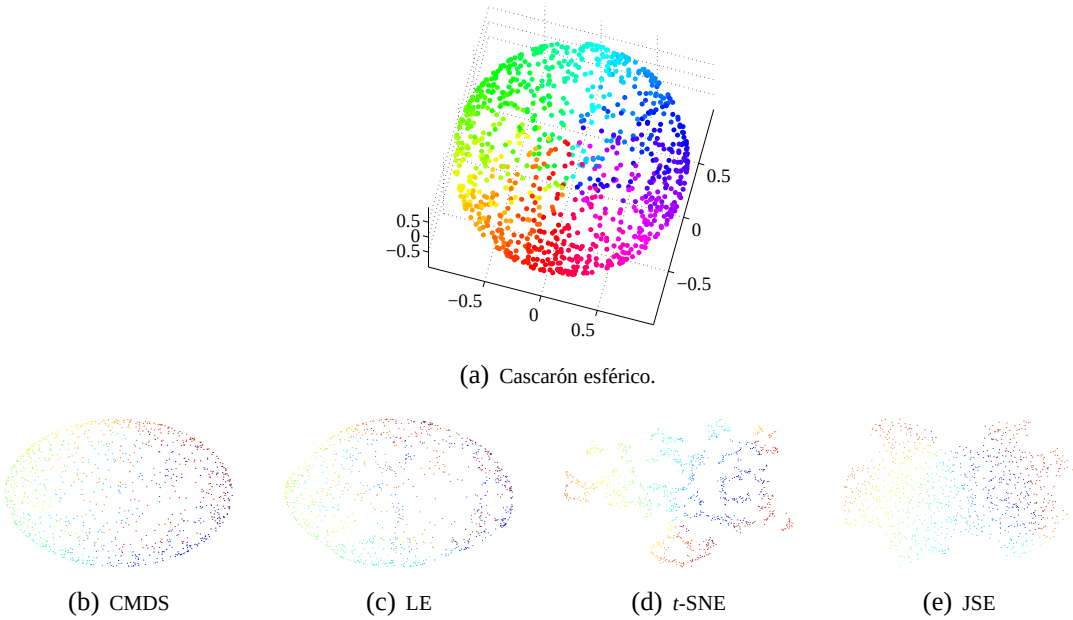
Se tiene una matriz de datos $\boldsymbol{A} \in \mathbb{R}^{N \times P}$ y un vector de etiquetas $\boldsymbol{y} \in \mathbb{R}^{N}$, donde P es el número de características o atributos de la base de datos original de diagnóstico médico. Después de aplicar un método de reducción o selección de atributos se obtiene una matriz $\boldsymbol{X} \in \mathbb{R}^{N \times D}$, con D características o atributos, menor que en la base de datos original, es decir, $D < P$.

3.2. Reducción de dimensiones

Una forma natural de representar los casos en un espacio de representación menor, que a su vez es una forma de visualizarlos de manera inteligible para el usuario, es a través de un diagrama de dispersión de 2 ó 3 dimensiones. Esto supone que el conjunto inicial de datos debe representarse en un espacio de dimensión menor que la original y se denomina reducción de dimensión (RD) y es una etapa importante dentro de los sistemas de reconocimiento de patrones y visualización de datos, puesto a que está orientada a representar los datos en una dimensión menor en donde el desempeño tanto perceptual (por parte del humano) como el costo computacional mejoren sustancialmente y a su vez debe representar la naturaleza intrínseca de las variables de los casos [44] para obtener una visualización realística y más inteligible para el usuario [45].
Entre los métodos clásicos de RD, se encuentra el análisis de componentes principales (PCA por su nombre en inglés - *principal component analysis*) y (CMDS por su nombre en inglés - *classical multidimensional scaling*), los cuales se basan en criterios de conservación de la varianza y la distancia, respectivamente [46]. Recientemente, los métodos de RD se enfocan en criterios orientados a la preservación de la topología de los datos. Normalmente, dicha topología se representa mediante una matriz de similitud o afinidad que representa el grado de relación o conexión entre los puntos coordenados (coordenadas cartesianas que representan los datos). Desde un punto de vista de teoría de grafos, los datos pueden representarse a través de un grafo ponderado (grafo con un valor de peso por cada adyacencia o arista) y no dirigido, en el cual los nodos representan los puntos coordenados, y la matriz de similitud o afinidad contiene los pesos de cada arista. Los métodos pioneros en incluir similitudes son (LE por su nombre en inglés - *Laplacian eigenmaps*) [47] y (LLE por su nombre en inglés - *locally linear embedding*) [48], los cuales son de tipo espectral, es decir que usan la información de los valores vectores y vectores propios. Por otra parte, dado que la matriz de similitud normalizada puede interpretarse como distribuciones de probabilidad, han surgido otros enfoques basados en divergencias, tales como (SNE - *stochastic neighbour embedding*) [49], y sus variantes y mejoras, *t*-SNE que usa una distribución *t*-Student y JSE que usa la divergencia de Jensen-Shanon [50, 51]. En la Figura 3.1 se muestra los espacios de baja dimensión resultantes de aplicar algunos métodos de RD sobre un conjunto de datos artificiales que representa un cascarón esférico. Este conjunto de datos es simple y la tarea de reducción consiste, de algún modo, en *desdoblar* la esfera, es decir, generar una representación plana de la esfera conservando la relación entre puntos vecinos.

3.3. Selección de atributos

Las técnicas de minería de datos que extraen modelos a partir de ejemplos tienden a obtener modelos complejos conforme crece el volumen de datos del conjunto sobre el cual se aplican.



(a) Cascarón esférico.

(b) CMDS (c) LE (d) *t*-SNE (e) JSE

Figura 3.1.: Representaciones en baja dimensión 2D de un cascarón esférico 3D usando diferentes técnicas de RD.

El elevado tamaño de los conjuntos de datos provoca inconvenientes adicionales tales como: Aumento en el tiempo de respuesta de los modelos, aumento en la sensibilidad al ruido y la posibilidad de sobreajuste de sobre el conjunto de entrenamiento. Al extraerse modelos de gran tamaño, la solución obtenida es poco comprensible para la mente humana. Se hace por tanto necesario un preprocesamiento previo que disminuya el tamaño del conjunto almacenado [52].

Debido al efecto negativo de atributos irrelevantes en la mayoría de esquemas de aprendizaje automático, es común que se lleve a cabo un proceso de selección de atributos previo al aprendizaje. La selección de características, consiste en escoger las muestras más representativas de un conjunto determinado. Disminuyendo el conjunto inicial de datos, se consigue reducir tanto la complejidad en tiempo de cálculo, como los recursos de almacenamiento. La eliminación de instancias no produce una degradación de los resultados, debido a que se eliminan ejemplos repetidos o ruidosos evitando así el sobre aprendizaje.

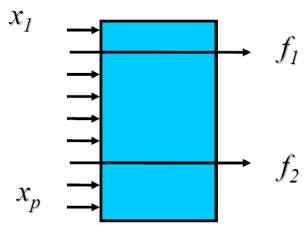


Figura 3.2.: Selección de características. Se parte de un conjunto de atributos y con el proceso de selección se obtienen f_D atributos.

Variables ordinales: Corresponde a variables cuyos valores no son números, pero se pueden ordenar. Los atributos que se registran pueden mantener entre si una relación de jerarquía, pero estas relaciones no permiten más que una cantidad determinada de análisis, como por ejemplo, el año de escolaridad, la pertenencia a un grupo socioeconómico o un grupo etario. Cuando se elaboran instrumentos que contemplen rotulaciones de atributos o estimaciones cualitativas, se habla de variables que poseen jerarquía.

Variables cardinales: Son aquellas variables en donde su valor tiene pleno significado numérico, es decir, que no sólo presentan las propiedades ordinales de los números, sino también las cardinales. Dichas variables se dividen en:

- Continuas: Variables que pueden tomar cualquier valor dentro de un intervalo (edad, salarios, estatura, producción anual, entre otras).
- Discretas: Aquellas que toman solo algunos valores dentro de un intervalo (hijos por familia, número de huelgas anuales, producción mensual de automóviles, entre otras).

3.3.1. Selección de características basada en filtros de correlación y búsqueda en profundidad

(CFS por su nombre en inglés - *Correlation Feature Selection*), es un algoritmo que elabora una jerarquización de subconjuntos de atributos de acuerdo a su correlación basada en una función de evaluación heurística. Dicha función de evaluación se basa en el cálculo de la correlación estadística, buscando atributos que están muy poco correlacionados entre si, pero tienen una buena correlación con la clase. Las características irrelevantes por tanto son ignoradas, debido a que se mantendrá una muy baja o nula correlación con la clase. La información redundante será penalizada, debido a que el atributo redundante tendrá una alta correlación con una o varias de las características restantes. La inclusión de una característica depende de si ésta es capaz de explicar la clase en fragmentos del espacio de instancias

que no han sido ya explicadas por otros atributos. La función de evaluación que se utiliza es la siguiente:

$$M_{S_D} = \frac{D\overline{r_{cf}}}{\sqrt{D + D(D-1)\overline{r_{ff}}}}, \tag{3.1}$$

donde M_{S_D} es el mérito heurístico del subconjunto S con D características, $\overline{ref}$ es el valor promedio de todas las correlaciones entre clase-característica y $\overline{rff}$ es el valor promedio de todas las correlaciones característica-característica del subconjunto S. Entonces, CFS se define como:

$$CFS = \max_{S_D}\left[\frac{r_{cf_1} + r_{cf_2} + \cdots + r_{cf_D}}{\sqrt{D + 2(r_{f_1f_2} + \cdots + r_{f_if_j} + \cdots + r_{f_Df_1})}}\right], \tag{3.2}$$

donde r_{cf_l} y $r_{f_if_j}$ son correlaciones $...l\ j$. El método CFS asume que las características son condicionalmente independientes dada la clase, esto puede ser una simplificación aceptable en algunos casos, pero si existe una fuerte interacción entre distintos atributos, entonces CFS no puede garantizar que los atributos seleccionados sean relevantes [53]. En este trabajo se utiliza como método de búsqueda el algoritmo de *BestFirst*.

BestFirst

La búsqueda BestFirst es un algoritmo de búsqueda en profundidad, aplicando *backtraking* o vuelta atrás hasta un límite de retrocesos. Básicamente la búsqueda se desarrolla usando un árbol y consiste en ir eliminando atributos hasta llegar a un cierto número de atributos (pre-determinados por el usuario). El subconjunto de atributos es evaluado mediante una métrica monotónica y el valor obtenido es guardado como una cota. A continuación, se procede a quitar otros atributos del conjunto original, siguiendo un esquema ordenado de eliminación de atributos (esquema de enumeración); cada subconjunto obtenido es evaluado. Si algún subconjunto obtiene un valor igual o peor que la cota, se detiene la exploración de esa rama (es decir, se realiza una poda), debido a que continuar la exploración es inútil, conduciendo a una mejor solución. Por otro lado, si todos los subconjuntos evaluados resultan mejor que la cota, se actualiza la cota con el nuevo valor, y se repite el procedimiento hasta que no existan más ramas a explorar. Este procedimiento ahorra tiempo de procesamiento y garantiza que la solución sea la óptima [53].

3.4. Conclusiones

Reducir la dimensionalidad del conjunto de datos cuenta con ventajas como las siguientes:

- Se reduce el tiempo de ejecución del proceso de clasificación.
- Se necesita menos espacio físico para el almacenamiento de los datos.
- Mejora el rendimiento de los modelos de aprendizaje.
- Los datos se visualizan mejor a través de los diagrama de dispersión de 2 ó 3 dimensiones.

El preproceso es una etapa muy importante en el análisis de datos y quizá la tarea más compleja de realizar, en la propuesta metodológica que se desarrolla en el siguiente capítulo, se aplican los conceptos de selección de atributos vistos en el presente capítulo.

4. CLASIFICACIÓN MULTICLASE

Una vez vistas algunas técnicas de limpieza, reducción, integración, selección y transformación de datos, en el capítulo anterior; se presenta en este capítulo los fundamentos teóricos de clasificadores multiclase representativos del ML que pueden ser acoplados a extensiones de sistemas de CBR. En general, todos los clasificadores se entrenan con un conjunto $\boldsymbol{X} \in \mathbb{R}^{N \times D}$ de N vectores formados por D atributos (casos), tal que $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)^\top$, donde $\boldsymbol{x}_i \in \mathbb{R}^D$ es el i-ésimo vector de atributos (caso).

4.1. Notación

Todos los métodos estudiados en este capítulo clasifican la matriz de casos $\boldsymbol{X} \in \mathbb{R}^{N \times D}$ dividendo las muestras en subconjuntos de entrenamiento y validación. La clasificación se realiza en un conjunto de C clases.

4.2. Máquinas de vectores de soporte

En lo últimos años, los clasificadores basados en una máquina de vectores de soporte (SVM por su nombre en inglés - *Support Vector Machine*) han demostrado su capacidad en aplicaciones como clasificación y reconocimento de patrones en general. Este clasificador mapea los puntos de entrada (vectores de atributos) a un espacio de características de una dimensión mayor, y, subsecuentemente, se calcula un hiperplano que separa dichos puntos maximizando el margen entre las clases [54].

4.3. Clasificador biclase basado en SVM

Para un grupo de entrenamiento de tamaño N compuesto de pares atributo-etiqueta $(\boldsymbol{x}_i, y_i)_{1 \leq i \leq N}$, siendo $\boldsymbol{x}_i \in \mathbb{R}^D$ y $y_i \in \{-1, 1\}$, se desea obtener una ecuación para un hiperplano que divida

dicho grupo de entrenamiento, de manera que aquellos puntos que tengan la misma asignación de clases se encuentren geométricamente ubicados en el mismo bisector que forma el hiperplano. En términos matemáticos, dicha ecuación está dada por:

$$f(\boldsymbol{x}_i) = \boldsymbol{w}^\top \boldsymbol{x}_i + b, \tag{4.1}$$

donde $\boldsymbol{w} \in \mathbb{R}^D$ es un vector normal al hiperplano y b representa un término de sesgo, los cuales definen la dirección y traslación del hiperplano, respectivamente [55]. Con lo anterior, la condición de asignación de clase para el i-ésimo punto puede escribirse como:

$$y_i f(\boldsymbol{x}_i) = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1, \tag{4.2}$$

de tal manera que el punto más cercano al hiperplano tenga como distancia $1/\|\boldsymbol{w}\|$.

Si existe un hiperplano que satisfaga la expresión (4.2), se dice que los datos son linealmente separables, como se puede observar en la Figura 4.7.

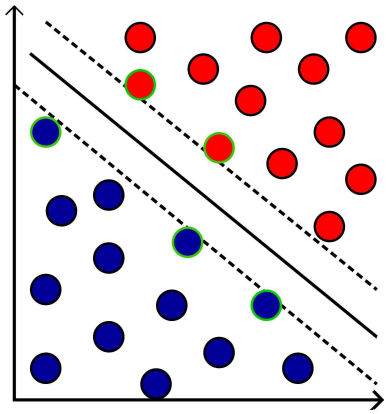


Figura 4.1.: Hiperplano de separación en un espacio bidimensional de un conjunto de ejemplos separables en dos clases.

Así, entre todos los posibles hiperplanos, aquel cuya distancia al punto más cercano es máxima se denomina el "óptimo hiperplano de separación" (OSH). Mientras la distancia al hiperplano óptimo sea $1/\|\boldsymbol{w}\|$, encontrar el OSH equivale a resolver el siguiente problema de optimización:

$$\begin{aligned} &\min_{\boldsymbol{w},b} \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{w} = \frac{1}{2}\|\boldsymbol{w}\|^2 \\ &\text{Sujeto a} \quad y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1, \forall i, \end{aligned} \tag{4.3}$$

donde $\|\cdot\|$ denota la norma Euclidiana.

La cantidad $2/\|\boldsymbol{w}\|$ se denomina "margen" y el hiperplano que maximiza dicho margen, OSH.

El margen puede ser visto como un indicativo de la dificultad del problema de separación de las clases, de forma que cuánto más pequeño sea el margen, más difícil es el problema; o de otro modo, se espera una mejor capacidad de generalización si el margen es más grande, como se observa en la Figura 4.8.

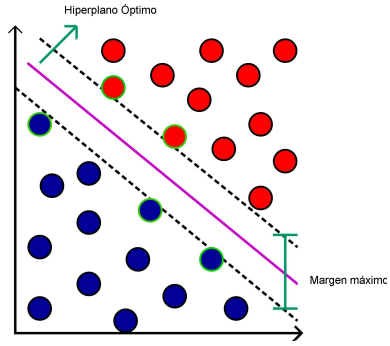


Figura 4.2.: Hiperplano de separación óptimo y su margen asociado.

Dada la forma cuadrática de $\boldsymbol{w}^\top \boldsymbol{w}$, una solución del problema de minización (4.3) es posible utilizando multiplicadores de Lagrange [54]. Considerando $\boldsymbol{\alpha} = \{\alpha_i, \ldots, \alpha_N\}$ como los N multiplicadores de Lagrange no negativos asociados con la condición dada en (4.2), la función de Lagrange de (4.3) [55] es:

$$\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\alpha}) = (1/2)\boldsymbol{w}^\top \boldsymbol{w} - \sum_{1=1}^{N} \alpha_i [\boldsymbol{y}_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) - 1]. \tag{4.4}$$

Encontrar el óptimo de (4.3) es equivalente a maximizar $\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\alpha})$ con respecto a los multiplicadores de Lagrange $\boldsymbol{\alpha}_1 \geq 0$. Una forma práctica de obtener el máximo es aplicando las condiciones de Karush-Kuhn-Tucker (KKT) [54]:

$$\frac{\partial \mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\alpha})}{b} = \sum_{1=1}^{N} \boldsymbol{y}_i \boldsymbol{\alpha}_i = 0, \tag{4.5}$$

$$\frac{\partial \mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\alpha})}{\boldsymbol{w}} = \boldsymbol{w} - \sum_{1=1}^{N} \boldsymbol{\alpha}_i \boldsymbol{y}_i \boldsymbol{x}_i = 0.$$

Substituyendo (4.5) en (4.4), se obtiene un nuevo problema de optmización:

$$\text{máx} \sum_{i}^{N} -\frac{1}{2} \sum_{i,l}^{N} \boldsymbol{\alpha}_i \boldsymbol{\alpha}_j \boldsymbol{y}_i \boldsymbol{y}_j \boldsymbol{x}_i' \boldsymbol{x}_j \tag{4.6}$$

$$\text{Sujeto a} \quad \sum_{i=1}^{N} \boldsymbol{y}_i \boldsymbol{\alpha}_i = 0 y \boldsymbol{\alpha}_i \geq 0, \forall i.$$

Dada su forma polinómica, el problema (4.6) puede resolverse a través de métodos de programación cuadrática estándar [54]. Una vez el vector $\boldsymbol{\alpha}^0 = \{\boldsymbol{\alpha}_i^0, \ldots, \boldsymbol{\alpha}_N^0\}$ solución de 4.6 ha sido encontrado, a partir de 4.5, el OSH $(\boldsymbol{w}, b)$ tiene la siguiente forma:

$$\boldsymbol{w}_0 = \sum_{i=0}^{N} \alpha_i^0 \boldsymbol{y}_i \boldsymbol{x}_i, \tag{4.7}$$

mientras b_0 puede ser obtenido a partir de las condiciones de KKT:

$$\alpha_i^0 [\boldsymbol{y}_i (\boldsymbol{w}^\top \boldsymbol{x}_i + b) - 1] = 0. \tag{4.8}$$

De la ecuación 4.8, se observa que los puntos donde $\boldsymbol{\alpha}_i^0 > 0$, satisfacen la desigualdad en 4.2. Geométricamente, significa que aquellos puntos son los más cercanos al OSH. Estos puntos juegan un papel importante debido a que son los únicos valores necesarios en la expresión para el OSH (ver ecuación 4.7) y son llamados "vectores de soporte" (SV), debido a que dan "soporte" a la expansión de $\boldsymbol{w}_0$.

Dado un vector de soporte $\boldsymbol{x}_i$, el parámetro b puede ser obtenido de las condiciones KKT como:

$$b_0 = \boldsymbol{y}_i - \boldsymbol{w_0}^\top \boldsymbol{x}_i.$$

El problema de clasificar un nuevo punto x, es resuelto examinando el signo de $\boldsymbol{w}_0^\top \boldsymbol{x} + b_0$, es decir, considerando la expansión 4.7 de $\boldsymbol{w}_0$, la clase asignada a un nuevo punto (nuevo caso) $\widetilde{\boldsymbol{x}}$, está dada por:

$$\widetilde{y} = f(\widetilde{\boldsymbol{x}}) = \text{sign}\left(\sum_{i=1}^{N} \boldsymbol{\alpha}_i^0 \boldsymbol{y}_i \widetilde{\boldsymbol{x}}^\top \widetilde{\boldsymbol{x}} + b \right).$$

El término $\widetilde{\boldsymbol{x}}^\top \widetilde{\boldsymbol{x}}$ representa un elemento de la matriz de covarianza o varianza de $\widetilde{\boldsymbol{x}}$ desde una perspectiva estadística, denota la proyección de un vector sobre otro desde una perspectiva geométrica, y corresponde a un producto interior (también denominado producto interno, producto escalar en el espacio Euclídeo) desde una perspectiva de análsis funcional. Dicho producto interior puede generalizarse con el concepto de función kernel como se explica en la siguiente sección.

4.4. Funciones kernel

En general, el término *kernel* se utiliza para definir una función que establece la similitud entre los elementos de entrada dados. Los Kernels permiten mapear desde un espacio de entrada de dimensión d ($\boldsymbol{X}$) a un espacio de mayor dimensión d_h ($\boldsymbol{\Phi}$), donde $d_h >> d$. En términos de clasificación, la ventaja de mapear el espacio de datos original en uno superior reside en el hecho de que el último espacio puede proporcionar mayor separabilidad de las clases. Una explicación gráfica de este mapeo se aprecia en la Figura 4.3. Además, debe tenerse en cuenta que el mapeo se realiza antes de cualquier proceso de clasificación.

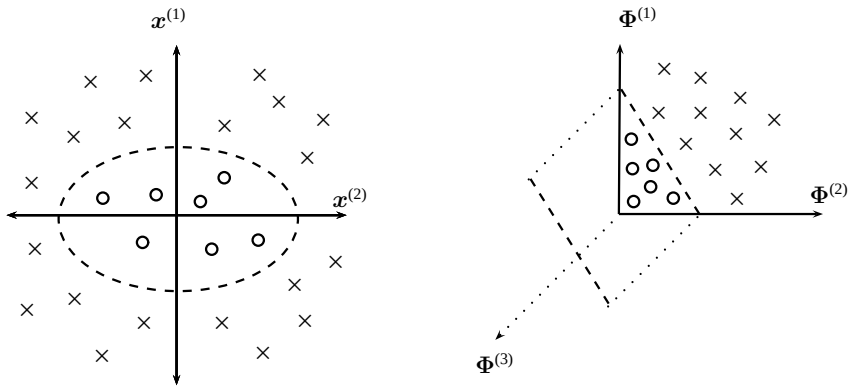


Figura 4.3.: Espacio de características en alta dimensión.

El objetivo de esta sección es describir algunos aspectos básicos y fundamentos con respecto a los kernels, en particular, con fines de clasificación.
La sección 4.4.1 presenta en términos generales la definición de kernel y algunas propiedades y conceptos relacionados. Finalmente, en la sección 4.4.2, se describen las funciones kernel más comunes.

4.4.1. Funciones kernel

En términos de la teoría del aprendizaje humano [56], uno de los problemas fundamentales es la discriminación entre elementos u objetos. Por ejemplo, en un conjunto de objetos formados por dos clases diferentes. En este caso, la tarea de clasificación consiste en determinar a qué clase pertenece un nuevo objeto, desconocido en principio. Esto se hace generalmente teniendo en cuenta las propiedades del objeto, así como similitudes y diferencias con respecto a las dos clases previamente conocidas. De acuerdo con lo anterior, y con respecto a la teoría de funciones kernel, es necesario crear o elegir una medida de similitud o afinidad para comparar los objetos (representados en vectores de atributos).
Las funciones kernel consideradas en este trabajo son definidos positivos. En términos matemáticos, una función kernel es de la forma:

$$\begin{aligned}\mathcal{K}(\cdot,\cdot) :&\mathbb{K}^d \times \mathbb{K}^d \longrightarrow \mathbb{K}\\ &\boldsymbol{x}_i, \boldsymbol{x}_j \quad \longmapsto \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j),\end{aligned} \tag{4.9}$$

donde $\mathbb{K} = \mathbb{C}$ ó $\mathbb{R}$. Nótese que se ha asumido que los objetos $\boldsymbol{x}_i$ son reales y de dimensión d. Por tanto,si se tiene un total de N objetos, puede escribirse una matriz $\boldsymbol{K}$ de dimensión $N \times N$ cuya entrada ij está dada por $k_{ij} = \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Dicha matriz se denomina matriz de Gram o matriz kernel. Consecuentemente, la matriz kernel debe ser una matriz definida positiva que satisface:

$$\sum_{i=1}^{N}\sum_{j=1}^{N} c_i \bar{c}_j k_{ij} \geq 0, \tag{4.10}$$

para todo $c_i \in \mathbb{C}$, donde $\bar{c}_i$ denota el complejo conjugado de c_i. De forma similar, una maitrz $\boldsymbol{K}$ real simétrica y de dimensión $N \times N$ también se denomina positiva definida si cumple la condición (4.10) para todo $c_i \in \mathbb{R}$ [57]. Adicionalmente, una matriz es simétrica positiva definida si y sólo si todos sus valore propios son positivos.

Para los siguientes desarrollos matemáticos, se considera la siguiente notación: La función que mapea desde un espacio de dimensión d a un espacio de dimensión d_h es de la forma $\phi(\cdot)$, tal que:

$$\begin{aligned}\phi(\cdot) :&\mathbb{R}^d \longrightarrow \mathbb{R}^{d_h}\\ &\boldsymbol{x}_i \longmapsto \phi(\boldsymbol{x}_i).\end{aligned} \tag{4.11}$$

A partir de la función ϕ puede obtenerse la matriz $\boldsymbol{\Phi} = [\phi(\boldsymbol{x}_1)^\top, \ldots, \phi(\boldsymbol{x}_N)^\top]^\top$, $\boldsymbol{\Phi} \in \mathbb{R}^{N \times d_h}$, la cual corrresponde a una representación de alta dimensión del espacio de entrada $\boldsymbol{X} \in \mathbb{R}^{N \times d}$ (Ver Figura 4.4).

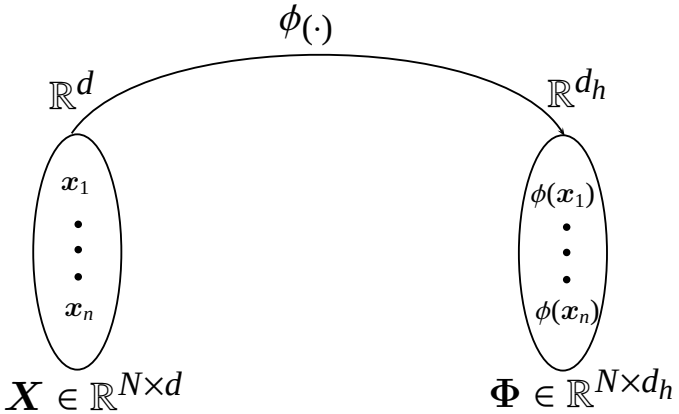


Figura 4.4.: Mapeo de alta dimensión del espacio de entrada $\boldsymbol{X}$ usando la función ϕ.

Una propiedad interesante y muy útil es la denominada *kernel trick* que se deriva de los criterios de Mercer [58]. Esta propiedad gana importancia en la teoría de funciones kernel, ya que permite reemplazar una función kernel positiva definida con otra función kernel que es finita y aproximadamente positiva definida. Por ejemplo, dado un determinado algoritmo formulado en términos de una función kernel positiva definida $\mathcal{K}$, uno puede construir un algoritmo alternativo reemplazando este por otra función kernel positiva definida $\tilde{\mathcal{K}}$ [57], de tal manera que $\boldsymbol{\Phi}\boldsymbol{\Phi}^{\boldsymbol{T}} = \boldsymbol{\Omega}$. Entonces, en este caso, la función kernel $\boldsymbol{\Phi}\boldsymbol{\Phi}^{\boldsymbol{T}}$ ha sido estimada como $\boldsymbol{K}$. El hecho de usar $\boldsymbol{K}$ como una estimación alternativa de $\boldsymbol{\Phi}\boldsymbol{\Phi}^{\boldsymbol{T}}$ es denominada kernel trick.

4.4.2. Tipos de funciones kernel

Las funciones kernel de base radial (*Radial basis function* - RBF) son aquellas que pueden escribirse en términos de una medida de similitud o disimilitud, así:

$$\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = f(\mathrm{d}(\boldsymbol{x}_i, \boldsymbol{x}_j)), \tag{4.12}$$

donde $\mathrm{d}(\cdot,\cdot)$ es una medida en el dominio de $\boldsymbol{X}$, es decir en el espacio $\mathbb{R}^d$, así:

$$\begin{aligned} \mathrm{d}(\cdot,\cdot) :& \mathbb{R}^d \times \mathbb{R}^d \longrightarrow \mathbb{R}^+ \\ & \boldsymbol{x}_i, \boldsymbol{x}_j \longmapsto \mathrm{d}(\boldsymbol{x}_i, \boldsymbol{x}_j), \end{aligned} \tag{4.13}$$

y f es una función definida en $\mathbb{R}^+$. Típicamente, dichas medidas se basan en el producto interior: $\mathrm{d}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\| = \sqrt{\left\langle \boldsymbol{x}_i - \boldsymbol{x}_j, \boldsymbol{x}_i - \boldsymbol{x}_j \right\rangle}$.

En la Tabla 4.1 se describe algunas funciones kernels recomendadas por la literatura.

Denominación	Definición	Dominio
Lineal	$\left\langle \boldsymbol{x}_i, \boldsymbol{x}_j \right\rangle$	$\mathbb{R}^d$
Polinomial	$\left\langle \boldsymbol{x}_i, \boldsymbol{x}_j \right\rangle^d$	$\mathbb{R}^d$
Racional cuadrático	$1 - \dfrac{\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2}{\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2 + \sigma}, \sigma \in \mathbb{R}^+$	$\mathbb{R}^d$
Exponencial	$\exp\left(-\dfrac{\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|}{2\sigma^2}\right), \sigma \in \mathbb{R}^+$	$\mathbb{R}^d$
Gaussiano	$\exp\left(-\dfrac{\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|^2}{2\sigma^2}\right), \sigma \in \mathbb{R}^+$	$\mathbb{R}^d$

Tabla 4.1.: Ejemplos de funciones kernel.

En la forma convencional del clasificador biclase basado en SVM, se usa una función kernel lineal, como se explica en la sección 4.3. No obstante, dada su capacidad de hacer un análisis local de la similitud, la función kernel Gaussiana es una de las más utilizadas [59].

4.5. Extensión multiclase

Cuando se trata de problemas que involucran más de dos clases, es necesario una extensión multiclase de la formulación descrita en la sección anterior. Se encuentran varios tipos de aproximaciones para lograrlo, uno contra uno, un grafo acíclico dirigido y uno contra todos (OaA por su nombre en inglés *one against all*), que en general alcanzan un desempeño similar [60, 61]. En particular, dentro de un marco basado en OaA, se introduce una formulación SMV multiclase, que consiste en construir un número de modelos SVM, uno por clase. Aplicando C veces el enfoque biclase descrito en la sección 4.3, se logra un enfoque de varias clases. En general, en el caso de utilizar enfoques SVM, la clase ℓ se compara con las restantes de tal manera que se asocia con una etiqueta positiva, mientras que las otras con una etiqueta negativa [60]; de manera que se forma un vector de etiquetado binario por cada clase individual.

En el caso multiclase cada pareja ordenada $\{\boldsymbol{x}_i, y_i\}$ para la i-ésima muestra puede corresponder con un vector de etiquetas donde $y_i \in 1, 2, \ldots, C$ donde C es el número total de clases.

En esta sección el objetivo es hacer frente a las diversas clases ℓ, tal que $\ell \in \{1, \ldots, c\}$, donde C es la cantidad de clases consideradas.

Concretamente, el vector que referencia las etiquetas $\bar{\boldsymbol{y}}^{(\ell)}$ asociadas a la clase ℓ asumiendo que: $\bar{y}_i^{(\ell)} = 1$ si $\boldsymbol{x}_i$ pertenece a la clase ℓ, de lo contrario 0. En este sentido, el enfoque propuesto en (4.3) puede generalizarse como:

$$\begin{aligned} &\min_{\boldsymbol{w}^{(\ell)}, b^{(\ell)}} \frac{1}{2}\boldsymbol{w}^{(\ell)\top}\boldsymbol{w}^{(\ell)} = \frac{1}{2}\|\boldsymbol{w}^{(\ell)}\|^2 \\ &\text{Sujeto a } \ \bar{y}_i(\boldsymbol{w}^{(\ell)\top}\boldsymbol{x}_i + b^{(\ell)}) \geq 1, i \in \{1, \ldots, N\}, \ell \in \{1, \ldots, C\}, \end{aligned} \tag{4.14}$$

de esta manera se construyen C vectores $\boldsymbol{w}^{(\ell)}$ (es decir, C modelos de SVM biclase) donde:

$$\begin{aligned} &\boldsymbol{w}^{(1)\top}\boldsymbol{x}_i + b^{(1)}, \\ &\quad\vdots \\ &\boldsymbol{w}^{(C)\top}\boldsymbol{x}_i + b^{(C)}, \end{aligned}$$

desde la extensión multiclase también proporciona una etiqueta para el caso nuevo $\widetilde{\boldsymbol{x}}$ denotada con $\widetilde{y} \in \{1, \ldots, C\}$ está dada por:

$$\widetilde{y} = \arg \max_{\ell \in \{1,\ldots,c\}} \boldsymbol{w}^{(\ell)\top}\widetilde{\boldsymbol{x}} + b^{(\ell)}. \tag{4.15}$$

4.6. Clasificador basado en densidades usando el método de Parzen

Para explicar la necesidad de estimar probabilidades a posteriori usando el método de Parzen en el procesos de clasificación, se explica continuación el concepto de clasificador de máxima esperanza dentro de un modelo iterativo genérico.

4.6.1. Modelo iterativo genérico

Una forma generalizada e iterativa de realizar clasificación basada en densidades (*Density-based classification* - DBC) puede obtenerse estudiando la proporción o grado de pertenencia de un elemento a una clase y la influencia de cada elemento en el cálculo de la actualización de la iteración siguiente. El grado de pertenencia de un elemento a una clase es determinado por una función de membresía que se denota con $m(\ell|\boldsymbol{x}_i)$: grado de pertenencia de $\boldsymbol{x}_i$ a la clase ℓ. El grado de membresía es un valor no negativo y la pertenencia absoluta es 1, por tanto la función m debe satisfacer

$$m(\ell|\boldsymbol{x}_i) \geq 0 \quad \text{y} \quad \sum_{\ell=1}^{C} m(\ell|\boldsymbol{x}_i) = 1.$$

El grado de influencia o peso de cada punto $w(\boldsymbol{x}_i)$ en el cálculo de las actualizaciones, es un factor de ponderación de los datos $\boldsymbol{x}_i$. Ambas funciones, m y w, están directamente relacionadas con la naturaleza de la función objetivo.

La actualización de las clases para la iteración r se puede escribir como:

$$\ell^{(r)} = \frac{\sum_{i=1}^{N} m(\ell^{(r-1)}|\boldsymbol{x}_i) w(\boldsymbol{x}_i) \boldsymbol{x}_i}{\sum_{i=1}^{N} m(\ell^{(r-1)}/\boldsymbol{x}_i) w(\boldsymbol{x}_i)}, \quad \ell \in \{1, \ldots, C\} \tag{4.16}$$

La anterior ecuación es análoga a la expresión usada para el cálculo de un centro de masa, que comúnmente se emplea en geometría: $\boldsymbol{q} = \sum_i g(\boldsymbol{r}_i)\boldsymbol{r}_i / \sum_i g(\boldsymbol{r}_i)$, donde $\boldsymbol{r}_i$ es el vector de posición del i-ésimo elemento y $g(\cdot)$ es la función de densidad de masa.

Dado que las funciones de membresía y peso se pueden ajustar a cualquier función objetivo (conservando las restricciones discutidas anteriormente) y que la actualización de las clases se hace de forma iterativa, se puede decir que este método es un modelo iterativo genérico de clasificación. El modelo de entrenamiento se explica en el Algoritmo 1.

A continuación, se explica como este enfoque genérico se aplica en el clasificado de máxima esperanza Gaussiana.

Algorithm 1 Modelo iterativo genérico para actualización de clases

1. Inicialización: $(\ell)^{(0)}$, fijar el máximo número de iteraciones N_{iter}, inicializar el contador: $r = 1$
2. Calcular las funciones de membresía $m(\ell^{(r-1)}|\boldsymbol{x}_i)$ y peso $w(\boldsymbol{x}_i)$ para cada punto
3. Actualizar las clases: $\ell^{(r)} = \dfrac{\sum_{i=1}^{N} m(\ell^{(r-1)}|\boldsymbol{x}_i)w(\boldsymbol{x}_i)\boldsymbol{x}_i}{\sum_{i=1}^{N} m(\ell^{(r-1)}|\boldsymbol{x}_i)w(\boldsymbol{x}_i)}$ $\ell = 1, \ldots, C$
4. $r \leftarrow r + 1$ y repetir los pasos 2 y 3 hasta que el algoritmo converja o hasta que $r = N_{iter}$
5. Asignar el modelo a la clase $\widetilde{y}$ con $\arg\max_m m(\ell|\widetilde{\boldsymbol{x}})$

4.6.2. Clasificador de máxima esperanza Gaussiana

El método de clasificación basado en la máxima esperanza gaussiana (*GEMC*), hace parte de los métodos de(*DBC*) y tiene como función objetivo la combinación lineal de distribuciones gaussianas centradas en los valores medios de cada clase, así:

$$f_{GEMC}(\boldsymbol{X}, \ell) = -\sum_{i=1}^{N} \log\left(\sum_{\ell=1}^{C} p(\boldsymbol{x}_i|\ell)p(\ell)\right) \tag{4.17}$$

donde $p(\boldsymbol{x}_i/\ell)$ es la probabilidad de $\boldsymbol{x}_i$ dado que es generado por una distribución gaussiana centrada en $\boldsymbol{\mu}_\ell$ (la media de la clase ℓ) y $p(\ell)$ es la probabilidad a priori de la clase ℓ. Se emplea la función logaritmo por facilidad matemática (hace el crecimiento de la función más lento) y el signo menos con el fin de que la tarea sea minimizar la función objetivo.

Las funciones correspondientes a la membresía y el peso de cada elemento son, respectivamente,

$$m_{GEMC}(\boldsymbol{q}_j/\boldsymbol{x}_i) = \frac{p(\boldsymbol{x}_i/\boldsymbol{q}_j)p(\boldsymbol{q}_j)}{p(\boldsymbol{x}_i)}, \tag{4.18}$$

y

$$w_{GEMC}(\boldsymbol{x}_i) = 1. \tag{4.19}$$

Para este clasificador, la función de membresía es un valor de probabilidad, por tanto la regla de Bayes puede emplearse para el cálculo de su valor, considerando $p(\boldsymbol{x}_i)$ como la evidencia:

$$p(\boldsymbol{x}_i) = \sum_{\ell=1}^{C} p(\boldsymbol{x}_i|\ell)p(\ell).$$

El factor $p(\boldsymbol{x}_i|\ell)$ puede obtenerse fácilmente con:

$$p(\boldsymbol{x}_i|\ell) = f(\boldsymbol{x}_i, \boldsymbol{\mu}_\ell, \boldsymbol{\Sigma}_\ell) = \frac{1}{\det(\boldsymbol{\Sigma}_\ell)^{\frac{1}{2}}}(2\pi)^{-d/2} e^{-\frac{1}{2}(\boldsymbol{x}_i - \boldsymbol{\mu}_\ell)\boldsymbol{\Sigma}_\ell^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}_\ell)^\top}, \tag{4.20}$$

donde $\boldsymbol{\mu}_\ell$ es la media de la clase ℓ, d es la dimensión, $\boldsymbol{\Sigma}_\ell$ representa la matriz de covarianza y det(·) denota el determinante de su matriz argumento.

De acuerdo con la regla de Bayes, la matriz $\boldsymbol{\Sigma}_\ell$ puede ser única ($\boldsymbol{\Sigma}_\ell = \boldsymbol{\Sigma} = \text{cov}(\boldsymbol{X})$) o puede calcularse por cada clase. La segunda opción es recomendable porque considera la varianza de cada uno de las clases, además, la matriz $\boldsymbol{\Sigma}_\ell$ podría calcularse y actualizarse por cada iteración, aunque esto podría aumentar el coste computacional.

4.6.3. Estimación de probabilidad a posteriori usando el método de Parzen (PC)

Una variante del método *GEMC* puede obtenerse calculando la probabilidad a posteriori $p(\boldsymbol{x}_i|\ell)$ de forma no paramétrica, empleando el método de Parzen, que consiste en la superposición de distribuciones gaussianas de un tamaño fijo h centradas en cada $\boldsymbol{x}_i$ [62]. A este método se le denomina DBC no paramétrico o PC (Clasificador de Parzen). El valor óptimo de h se puede obtener con validación cruzada.

Matemáticamente, la distribución de probabilidad empleando el método de Parzen es

$$p(\boldsymbol{x}) = \frac{1}{nh} \sum_{i=1}^{n} \mathcal{K}\left(\frac{\boldsymbol{x} - \boldsymbol{x}_i}{h}\right) \tag{4.21}$$

donde $\mathcal{K}$ es un kernel gaussiano definido como:

$$\mathcal{K}(\boldsymbol{z}) = \frac{1}{(2\pi)^{-d/2}} \exp^{-\frac{1}{2}\boldsymbol{z}\boldsymbol{z}^\top} \tag{4.22}$$

Se opta en utilizar las funciones de núcleo gaussiano por dos razones. En primer lugar, la función gaussiana es suave y, por tanto, la función de densidad estimada también varía suavemente. En segundo lugar, si asumimos una forma especial de la familia gaussiana en la cual la función es radialmente simétrica, la función puede ser completamente especificada por un parámetro de varianza solamente. En la Figura 4.9 se aprecia una explicación gráfica de la estimación de la función de probabilidad no paramétrica de Parzen.

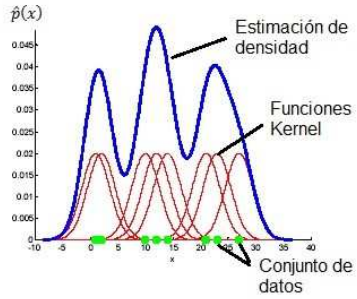


Figura 4.5.: La estimación ventana Parzen puede ser considerada como una suma de gaussianas centradas en los puntos. La función de Kernel determina la forma de las gaussianas. El parámetro *h*, también llamado el parámetro de suavizado o ancho de banda, determina su tamaño.

4.7. K-vecinos más cercanos

El método de los K-vecinos más cercanos o K-NN es un método supervisado, cuyo argumento principal es la distancia entre instancias. El método básicamente consiste en comparar la nueva instancia a clasificar con los datos k más cercanos conocidos, y dependiendo del parecido entre los atributos el nuevo caso se ubicará en la clase que más se acerque al valor de sus propios atributos.

4.7.1. Métodos basados en vecindad

Los métodos basados en vecindad son fundamentalmente dependientes de la distancia y en consecuencia poseen características propias de ésta como la cercanía, la lejanía y la magnitud de longitud, entre otras. Los métodos basados en vecindad, además de servir para tareas de clasificación, también se usan para agrupación de datos. Existen dos grupos de métodos de vecindad, según la forma en que se realiza el aprendizaje. El grupo de los métodos retardados y los no retardados. En los métodos retardados como K-NN, cada vez que se va a clasificar un dato, en la fase de entrenamiento, se elabora un modelo específico para cada nuevo dato, y una vez que éste se clasifica sirve como un nuevo caso de entrenamiento para clasificar una nueva instancia. En los métodos no retardados se generaliza un solo modelo (también a partir de casos conocidos) para todos los nuevos datos que se desean clasificar, y éstos únicamente son tomados en cuenta como datos de entrenamiento cuando se vuelve a construir un nuevo modelo general [63].
Las métricas, alternativas, usadas para medir la distancia son:

- Distancia de Manhattan.
- Distancia de Chebychev.

- Distancia del coseno.
- Distancia de Mahalanobis.
- Distancia usando la función delta.
- Distancia entre dos conjuntos.

4.7.2. Distancia Euclidiana

Se trata de una función no negativa usada en diversos contextos para calcular la distancia entre dos puntos, primero en el plano y luego en el espacio. También sirve para definir la distancia entre dos puntos en otros tipos de espacios de tres o más dimensiones. Y para hallar la longitud de un segmento definido por dos puntos de una recta, del plano o de espacios de mayor dimensión [63]. La distancia euclidiana entre dos puntos se define en la ecuación 4.23, donde $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^d$:

$$\mathrm{d}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \sqrt{\sum_{j=1}^{d} (x_{1j} - x_{2j})^2}. \tag{4.23}$$

El efecto de la vecindad determinada por distancias, se observa gráficamente en la Figura 4.6.

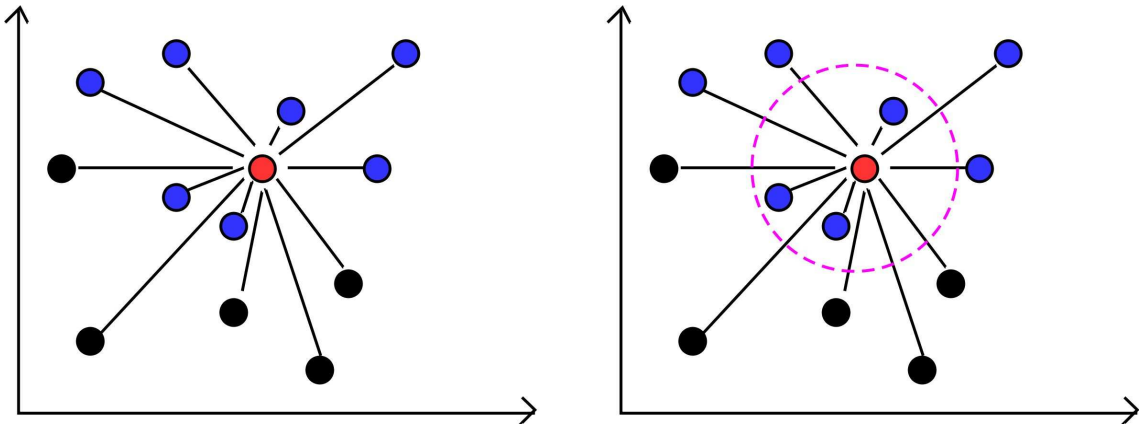


Figura 4.6.: Método de los K-vecinos más cercanos. Se calcula los vecinos cercanos de la muestra por medio de la distancia euclidiana. El punto rojo representa la nueva muestra y los conjuntos de cada clase están representados por los puntos azules y negros. La circunferencia punteada encierra a los casos similares recuperados para la nueva muestra.

4.8. Redes Neuronales artificiales

La red neuronal artificial es un procesador paralelo distribuido constituido por unidades simples de procesamiento que tienen una disposición natural para almacenamiento de conocimiento experimental. Imitan al cerebro en dos aspectos:

- El conocimiento es adquirido por la red desde su ambiente a través de un proceso de aprendizaje.
- La fuerza de conexión entre las neuronas (conocido como los pesos sinápticos) son usados para almacenar el conocimiento adquirido [64].

Las redes neuronales artificiales (ANNs por su nombre en inglés - *Artificial neural networks*) se caracterizan principalmente por:

- Tener una inclinación natural a adquirir el conocimiento a través de la experiencia, el cual es almacenado, al igual que en el cerebro.
- Poseen un alto nivel de tolerancia a fallos, es decir, pueden sufrir un daño considerable y continuar teniendo un buen comportamiento, al igual como ocurre en los sistemas biológicos.
- Tener un comportamiento altamente no lineal, lo que les permite procesar información procedente de otros fenómenos no-lineales [65].

Las neuronas biológicas interactúan dinámicamente entre ellas y cambian sus relaciones en el tiempo. Las interacciones son bastante complejas y dependen de la estructura de las neuronas. En una estructura simple, se puede observar la interconexión tipo *"feedforward"*. La información de una célula se pasa a otra y puede ser más fuerte que otras conexiones [64]. El modelo de una red neuronal se indica en la Figura 4.7. Sus variables de entrada, parámetros y variables de salida son:

- Cualquier vector de entradas $\boldsymbol{x}$: $\boldsymbol{x} = (x_1, \ldots x_d)^\top$.
- Vector de pesos: $\boldsymbol{w} = (w_1, \ldots, w_d)^\top$.
- Un factor de desplazamiento (bias) b.
- Una función de activación $f(\overline{x})$.
- Salida: y.

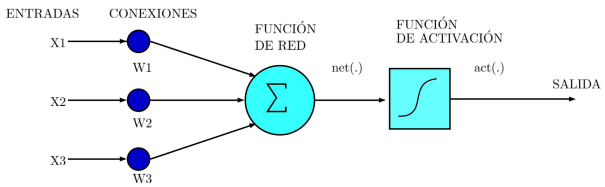


Figura 4.7.: Modelo de una red neuronal. Esta red neuronal de una sola capa cuenta con 3 entradas, una función de red y una función de activación.

- Una variable temporal $\overline{x}$:

$$\overline{x} = \sum_{j=1}^{d} w_j x_j + b = \boldsymbol{x}^\top \boldsymbol{w} + b. \tag{4.24}$$

- La función de activación determina la señal de salida

$$y = f(\overline{x}) = f\left(\sum_{i=1}^{n} w_i x_i + b\right) = f\left(\boldsymbol{x}^\top \boldsymbol{w} + b\right) \tag{4.25}$$

- La salida y generalmente se normaliza en un rango $y \in [0, 1]$ o $\in [-1, 1]$. Las funciones de activación pueden ser, la función de umbral, función sigmoidal, función tangente hiperbólica, lineal entre otras. La función de activación utilizada en los experimentos del presente trabajo es la sigmoidal.

4.8.1. Función sigmoidal

Es una función matemática que aparece en diversos modelos de crecimiento de poblaciones, propagación de enfermedades epidémicas y difusión en redes sociales. Dicha función constituye un refinamiento del modelo exponencial para el crecimiento de una magnitud. Modela la función sigmoidea de crecimiento de un conjunto P.
El estudio inicial de crecimiento es aproximadamente exponencial; al cabo de un tiempo, aparece la competencia entre algunos miembros de P por algún recurso crítico y la tasa de crecimiento disminuye; finalmente, en la madurez, el crecimiento se detiene como se puede observar en la Figura 4.8
La función logística simple se define mediante la expresión matemática:

$$f(\overline{x}) = \frac{1}{1 + exp(-\overline{x})}. \tag{4.26}$$

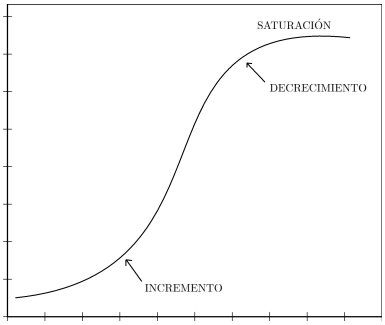


Figura 4.8.: Función sigmoidal. La curva varía en el tiempo indicando un instante en el que se presenta un crecimiento, seguido de un leve decrecimiento y finalmente unasaturación.

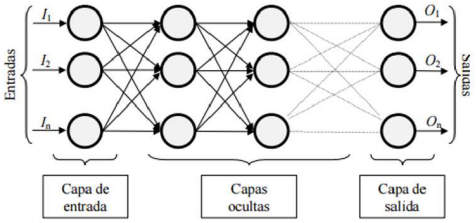


Figura 4.9.: Red neuronal. Los datos ingresan por medio de la “capa de entrada”, pasan a través de la “capa oculta” y salen por la “capa de salida”

Las diferentes clases de ANNs se distinguen entre sí por los siguientes elementos:

- Las neuronas o nodos constituyen elementos básicos de procesamiento.
- La arquitectura de la red esta descrita por las conexiones ponderadas entre los nodos.
- El algoritmo de entrenamiento, usado para encontrar los parámetros de la red.

4.8.2. Redes Neuronales hacia adelante (Feedforward)

Este tipo de redes recibe un vector de entrada equivalente en magnitud al número de neuronas de la primera capa. La red neuronal procesa el vector de entrada en paralelo elemento por elemento. La información, modificada por los factores multiplicativos de los pesos en cada neurona, y transmitida hacia adelante por la red, pasa por las capas ocultas (si las hay) y finalmente es procesada por la capa de salida.

4.8.3. Redes de propagación hacia atrás (Backpropagation)

El nombre de backpropagation resulta de la forma en que el error es propagado hacia atrás a través de la red neuronal, en otras palabras, el error se propaga hacia atrás desde la capa de

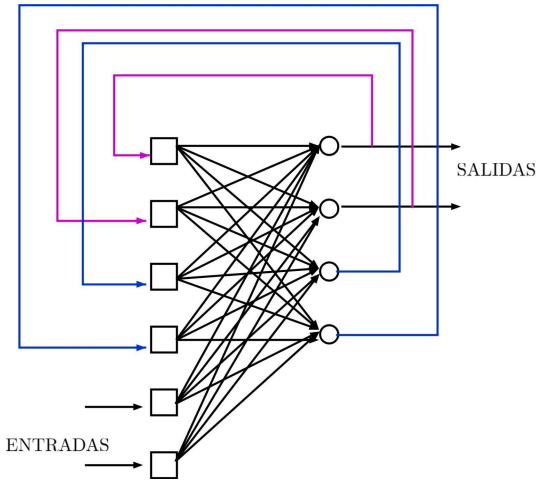


Figura 4.10.: Red neuronal de propagación hacia atrás (backpropagation)

salida. Esto permite que los pesos sobre las conexiones de las neuronas ubicadas en las capas ocultas cambien durante el entrenamiento. El cambio de los pesos en las conexiones de las neuronas además de influir sobre la entrada global, influye en la activación y por consiguiente en la salida de una neurona. Por lo tanto, es de gran utilidad considerar las variaciones de la función activación al modificarse el valor de los pesos. Esto se llama sensibilidad de la función activación, de acuerdo al cambio en los pesos [66].

4.9. Conclusiones

A continuación se listan algunas de las ventajas y las desventajas de los diferentes clasificadores.

- El clasificador basado en SVM trabaja muy bien con conjuntos de entrenamiento pequeños. Es muy útil evaluar este clasificador, debido a que en diagnóstico médico pueden existir bases de casos de enfermedades raras o poco comunes que no cuenten con datos suficientes para poder entrenarse otro clasificador.

- Debido a la robustez de los clasificadores basados en SVM, pueden presentar un elevado coste computacional.

- El clasificador PC obtiene muy buenos resultados si se cuenta con muestras suficientes, esto es una desventaja al trabajar con un conjunto de datos pequeño.

- *K*NN es un algoritmo muy simple e intuitivo y es utilizado en el CBR clásico, como se puede ver en el capítulo 2.

- El clasificador basado en ANNs trabaja muy bien con modelos no lineales y es ampliamente utilizado en modelos predictivos para resultados dicotómicos en medicina, pero con una gran desventaja, su naturaleza de caja negra.

Parte III.

MÉTODOS

5. METODOLOGÍA PROPUESTA: SAM

Como se menciona en el estado del arte de los sistemas basados en CBR en medicina, la mayoría de los sistemas basados en CBR aplicados al diagnóstico médico, se limitan a presentar los casos más cercanos al nuevo caso que se quiere diagnosticar, sin aportar mayor información que pueda ayudar en la toma de decisiones. Por esta razón, se presenta en este capítulo una propuesta para mejorar la etapa de adaptación del CBR, denominada SAM (Sistema de Adaptación Mejorada).
SAM utiliza dos clasificadores *K*-NN en cascada, y una serie de algoritmos de cálculo de probabilidades para favorecer la clasificación de los pacientes enfermos y ofrecer al experto las probabilidades de pertenencia del nuevo caso a los posibles diagnósticos, proporcionando así una capacidad autoadaptativa al sistema. Todo esto enmarcado en dos grandes procesos que se describen en las secciones 5.1.1 y 5.1.2.

5.1. Descripción de la propuesta SAM

SAM es una propuesta de sistema de diagnóstico médico basado en la metodología de CBR. En una primera etapa propone una clasificación bi-clase entre enfermos y sanos, para después pasar a una segunda fase donde se calculan las probabilidades de pertenencia a los diferentes diagnósticos. En la Figura 5.1 se puede ver el diagrama de flujo que resume la propuesta.
En el primer bloque se describe gráficamente el pre-proceso, entra una base de datos a la cual se le aplican algoritmos de reducción de dimensión, convirtiéndose en la base de casos que posteriormente se utiliza para el entrenamiento de los clasificadores y para el proceso CBR.
En el bloque CBR, la entrada de un nuevo caso problema, la base de casos y los clasificadores entrenados son los elementos principales, y con los cuales se llevan a cabo las fases de recuperación y adaptación, revisión y aprendizaje.

Además en el gráfico se puede observar el diagrama de flujo del algoritmo de recuperación-adaptación y el de aprendizaje.

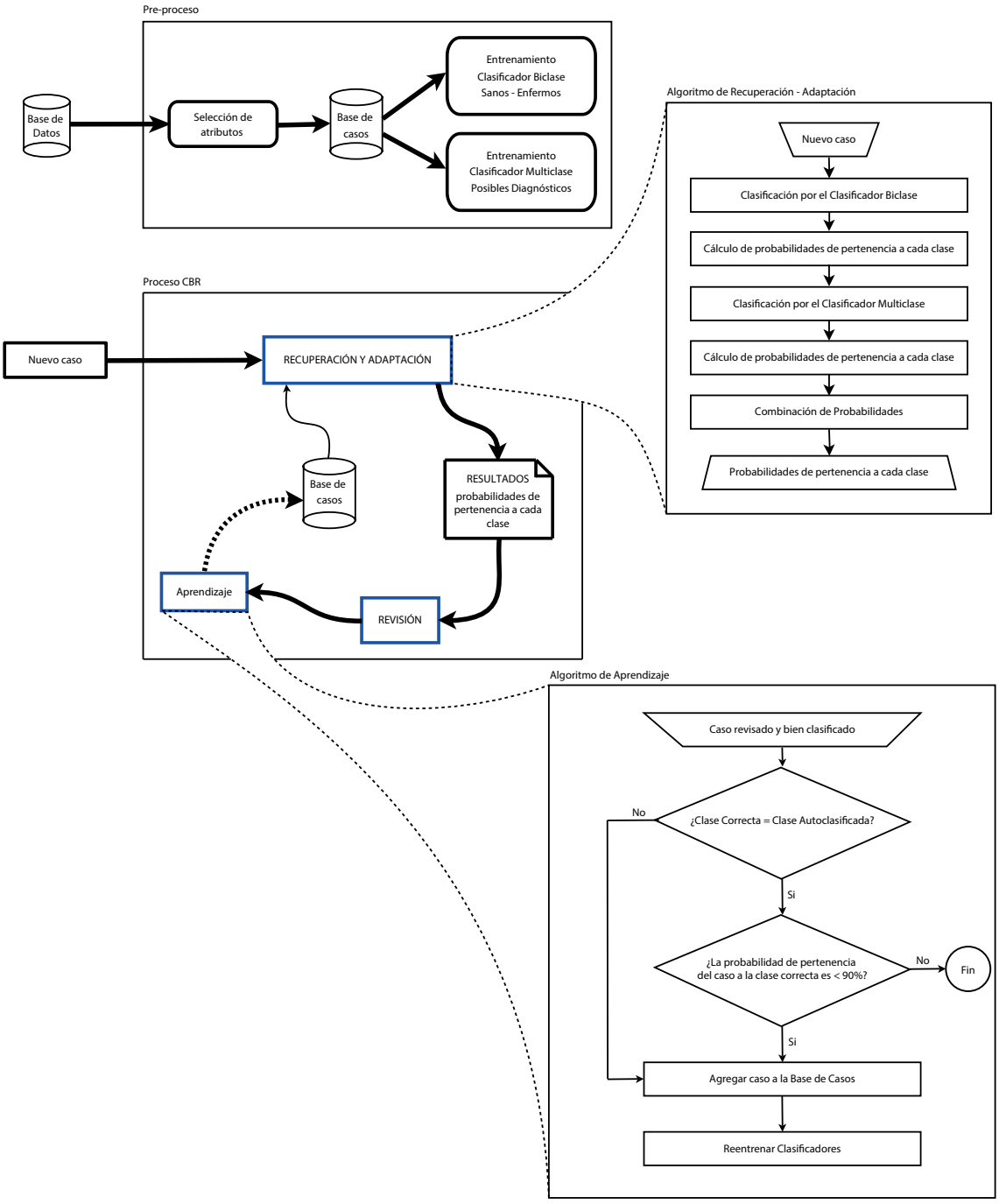


Figura 5.1.: Estructura del sistema SAM

5.1.1. Pre-proceso

El pre-proceso se lleva a cabo en tres etapas como se describe en las siguientes sub-secciones: Selección de atributos, entrenamiento del clasificador biclase, y entrenamiento del clasificador multiclase.

Selección de atributos

Se aplica el método de selección de características descrito en la sección 3.3.1 sobre la matriz inicial de datos $\boldsymbol{A} \in \mathbb{R}^{N \times P}$ y un vector de etiquetas $\boldsymbol{y} \in \mathbb{R}^{N}$, con el fin de obtener una matriz reducida de representación $\boldsymbol{X} \in \mathbb{R}^{N \times D}$. Donde N es el número de registros o casos, P y D representan el número de atributos o características, tal que $D < P$. Por tanto, $\boldsymbol{X}$ corresponde a la representación final de la base de casos (formada por N casos $\boldsymbol{x}_i \in \mathbb{R}^{D}$, $i \in \{1, \ldots, N\}$).

Entrenamiento del clasificador biclase

En esta etapa, se divide el vector de etiquetas $\boldsymbol{y}$ en dos clases: sanos y enfermos. Los pacientes sanos, generalmente pertenecen a la primera clase etiquetada como 1 y las clases restantes se condensan en una sola etiqueta denominada enfermos o -1, creando de esta forma un vector $\hat{\boldsymbol{y}} \in \mathbb{R}^{N}$ de etiquetas o asignación de clases biclase.
Se entrena entonces un clasificador biclase basado en el algoritmo de K-NN, utilizando la matriz $\boldsymbol{X}$ y el vector de etiquetas $\hat{\boldsymbol{y}}$.

Entrenamiento del clasificador multiclase

Para esta etapa se eliminan de la matriz $\boldsymbol{X}$ los casos etiquetados como sanos, y sus correspondientes etiquetas del vector $\boldsymbol{y}$, obteniendo una matriz $\tilde{\boldsymbol{X}} \in \mathbb{R}^{M \times D}$ ($M < N$) y sus correspondientes etiquetas o asignaciones de clase $\tilde{\boldsymbol{y}} \in \mathbb{R}^{M}$.
Se entrena un clasificador multiclase basado en el algoritmo de K-NN, utilizando la matriz $\tilde{\boldsymbol{X}}$ y el vector de etiquetas $\tilde{\boldsymbol{y}}$.
Finalmente, tras la etapa de pre-proceso, se cuenta con dos clasificadores entrenados listos para utilizar en las siguientes fases del sistema.

5.1.2. CBR

Recuperación y adaptación

Un nuevo caso o caso problema $\widetilde{\boldsymbol{x}}$ es evaluado por el clasificador biclase entrenado en la etapa de pre-proceso, luego se calculan las probabilidades de pertenencia a la clase sano (1) y a la clase enfermo (-1). El cálculo de las probabilidades se realiza mediante el algoritmo descrito en 3.
Seguidamente, el nuevo caso $\widetilde{\boldsymbol{x}}$ es evaluado por el clasificador multiclase entrenado en la

Algorithm 2 Algoritmo de clasificación

Require: Nuevo caso $\widetilde{\boldsymbol{x}}$, $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)^\top$

1: Aplicar clasificador K-NN biclase a $\widetilde{\boldsymbol{x}}$, obteniendo como resultado una matriz de distancias $\boldsymbol{D}^k_{\widetilde{\boldsymbol{x}}}$
2: Estimar $p(1, -1)$ aplicando el Algoritmo 3
3: Aplicar clasificador K-NN multiclase a $\widetilde{\boldsymbol{x}}$, obteniendo como resultado una matriz de distancias $\boldsymbol{D}^k_{\widetilde{\boldsymbol{x}}}$
4: Estimar $p(2, \ldots, C)$ aplicando el algoritmo 3
5: Hallar $p(2, \ldots, C| - 1)$
6: **return** $p(1), p(2, \ldots, C| - 1)$

etapa anterior y se calculan las probabilidades de pertenencia del nuevo caso a cada uno de los posibles diagnósticos. El cálculo de éstas probabilidades se realiza aplicando el algoritmo 3. El resultado de las probabilidades de pertenencia a los posibles diagnósticos es multiplicado por la probabilidad de encontrarse enfermo que se obtuvo del clasificador biclase. El algoritmo 2 resume los pasos de la etapa de recuperación y adaptación.

Cálculo de probabilidades: Para calcular las probabilidades de pertenencia a cada una de las clases, utilizando los elementos de salida de los clasificadores se tiene en cuenta la propuesta desarrollada por Duin y Tax en [67] y se explica en el algoritmo 3.

Revisión

La revisión se realiza manualmente. El especialista es quien analiza las probabilidades calculadas, asumiendo que la clase con mayor probabilidad corresponde a la clasificación final del sistema, e indica si dicha clasificación es correcta o no. En caso de ser incorrecta, asigna la etiqueta correcta según su experiencia.

Aprendizaje

En esta sección se presenta el algoritmo 4, propuesto para la etapa de aprendizaje. Las entradas de este algoritmo son: El nuevo caso, la etiqueta asignada por el especialista y las probabilidades halladas en la etapa de adaptación. Si la etiqueta asignada es igual a la clase con la máxima probabilidad y dicha probabilidad supera el umbral T, el sistema no realiza ninguna acción, debido a que es capaz de clasificar correctamente el nuevo caso. De lo contrario, el caso entra en la base de casos y se reentrenan los clasificadores biclase y multiclase. Si la etiqueta asignada es diferente a la clase con la máxima probabilidad, el sistema incluirá el caso con la etiqueta asignada a la base de casos y reentrena los clasificadores, con el fin de aprender del error.

Algorithm 3 Algoritmo para hallar probabilidades de pertenencia a cada clase, con el clasificador K-NN

Require: Matriz de distancias $\boldsymbol{D}$, K

if K=1 **then**

$$p(\ell|\widetilde{\boldsymbol{x}}) = \left[1 - \frac{(C-1)d_\ell}{\sum_{l=1}^{C} d_l} \right], \tag{5.1}$$

donde C es el número de clases, ℓ denota la clase, d_ℓ distancia de $\widetilde{\boldsymbol{x}}$ a la ℓ-ésima clase y d_l las distancias a los objetos de otras clases. [67]

return Datos normalizados, para garantizar que la suma de las probabilidades posteriores sobre todas las clases es 1 para cada objeto que se clasifica.

else

$$p(\ell|\widetilde{\boldsymbol{x}}) = \frac{n_\ell + 1}{K + C}, \tag{5.2}$$

donde n_ℓ es el número de casos recuperados etiquetados con la clase ℓ, C es el número total de clases y K número total de casos recuperados. [67]

return $p(\ell|\widetilde{\boldsymbol{x}})$

end if

Algorithm 4 Algoritmo llevado a cabo por el sistema para aprender

Require: Nuevo caso $\widetilde{\boldsymbol{x}}$, $\widetilde{y}$ y $p(\ell|\widetilde{\boldsymbol{x}})$

if $\widetilde{y}$= arg máx$_p$ $p(\ell|\widetilde{\boldsymbol{x}})$ **then**

if arg máx$_p$ $p(\ell|\widetilde{\boldsymbol{x}}) \geq T$ **then**

El sistema no realiza ninguna acción.

else

El caso entra en la base de casos

Se reentrenan los dos clasificadores

end if

else

El caso $\widetilde{\boldsymbol{x}}$ y $\widetilde{y}$ entran en la base de casos

Se reentrenan los clasificadores

end if

5.2. Interfaz Gráfica

En esta sección se muestra la interfaz gráfica diseñada, en donde el usuario puede trabajar de forma interactiva y establecer un contacto fácil e intuitivo con el sistema SAM. Esta interfaz está desarrollada en MATLAB. Se puede ver una imagen en la Figura 5.2.

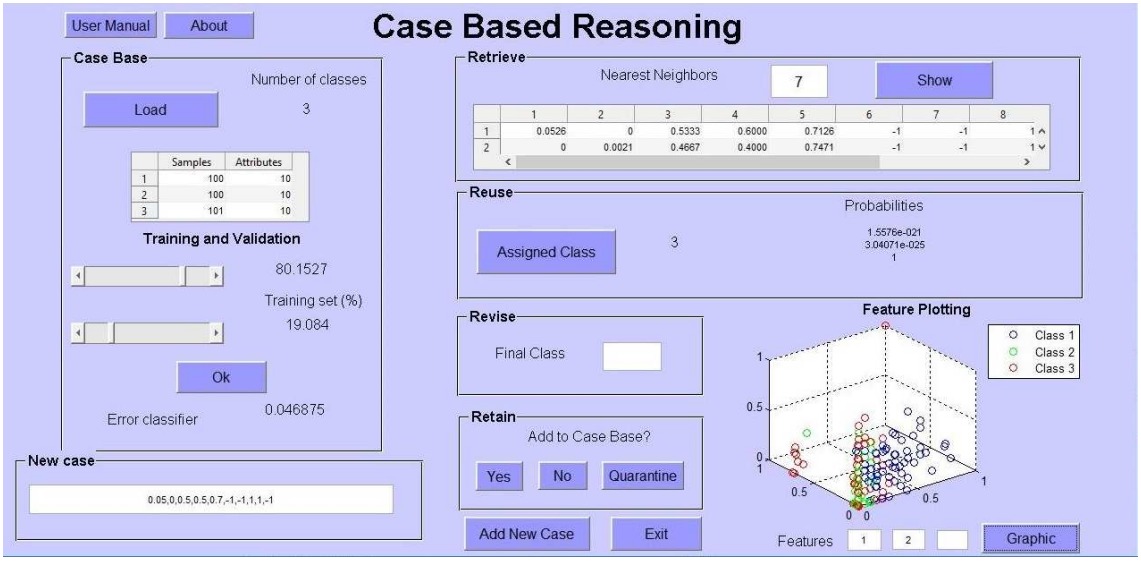


Figura 5.2.: Interfaz SAM

La primera interacción con la interfaz permite:

- Seleccionar la base de datos que desea usar.
- Establecer el número de datos con el que desea entrenar y probar los clasificadores
- Agregar un nuevo caso para evaluar con el CBR.

Una vez se tenga la base de datos, se muestra lo siguiente:

- El número de clases que posee la base de datos seleccionada.
- Número de atributos de las muestras.
- El error del clasificador.
- Es posible recuperar los k casos similares al nuevo problema.
- Estimación de probabilidades de pertenencia del nuevo caso a cada clase.

Cuando se entrenan los clasificadores, el sistema permite:

- Visualizar la clase asignada por el clasificador y las probabilidades de pertenencia a cada clase.
- Revisar el resultado y si es necesario asignar la clase correspondiente para el nuevo caso.
- Decidir si el caso que se está analizando debe añadirse a la base de casos.
- Graficar un conjunto de 1,2 o 3 características del conjunto de datos.

5.3. Conclusiones

- En la revisión de tema que se realiza en el capítulo 2 de esta tesis, se puede ver que los sistemas basados en CBR que utilizan técnicas del ML, ofrecen perspectivas poderosas para el desarrollo de sistemas robustos que apoyan las decisiones médicas. La propuesta descrita en el presente capítulo, fusiona la metodología de CBR y los clasificadores supervisados, con el fin de diseñar un sistema que puede aplicarse a cualquier base de datos de diagnóstico médico y que tiene la capacidad de trabajar con múltiples etiquetas.
- Uno de los descubrimientos más significativos que surgen de este estudio es que al combinar dos clasificadores supervisados se pueden obtener mejores resultados como se demuestra en el capítulo de experimentos. SAM es un sistema que combina dos clasificadores *K*-NN con algoritmos de cálculo de probabilidades que proporciona una mejor respuesta al usuario.

6. DESCRIPCIÓN DE LOS EXPERIMENTOS

La propuesta descrita anteriormente surge como resultado de un estudio comparativo de técnicas de representación de datos y de diferentes clasificadores multiclase en la etapa de recuperación y adaptación, como se puede observar en el desarrollo del presente capítulo. Las pruebas crecen en complejidad, permitiendo cumplir con los objetivos planteados al inicio de la presente investigación y demostrando finalmente que el sistema SAM es apropiado para trabajar con bases de datos de diagnóstico médico. Los experimentos más relevantes se describen detalladamente en las siguientes secciones.

6.1. Bases de datos

En este trabajo se consideran cuatro bases de datos con múltiples diagnósticos médicos de dominio público.

6.1.1. Base de datos de Cardiotocografía

La cardiotocografía, es un método de evaluación fetal que registra simultáneamente la frecuencia cardíaca fetal, los movimientos fetales y las contracciones uterinas, permitiendo al obstetra o matrona valorar la respuesta del bebé a las contracciones durante el trabajo de parto, y hasta el nacimiento [68].
El conjunto de datos registra las mediciones de la frecuencia cardíaca fetal (FHR por su nombre en inglés - *Fetal heart rate*) y las características de la contracción uterina (UC - *Uterine contraction*) en los cardiotocogramas clasificados por obstetras expertos .

- Fuente: [69]. [1]

[1] http://archive.ics.uci.edu/ml/datasets/Cardiotocography

- Número de instancias *N*: 2126.

  1655 de la clase 1, normal.

  295 de la clase 2, sospechoso.

  176 de la clase 3, patológico.

- Número de atributos *P*: 33

La información detallada de cada uno de los atributos se encuentra en el apéndice C

6.1.2. Base de datos de Hipotiroidismo

El hipotiroidismo, enfermedad que se caracteriza por la disminución de la actividad funcional de la glándula tiroides; provoca disminución del metabolismo basal, cansancio, sensibilidad al frío y en la mujer, alteraciones menstruales.
El conjunto de datos condensa la información de 10 bases de datos del Instituto Garavan.

- Fuente: [69]. [2]

- Número de instancias *N*: 3772.

  3481 de la clase 1, negativo.

  95 de la clase 2, hipotiroidismo primario.

  194 de la clase 3, hipotiroidismo compensado.

  2 de la clase 4, hipotiroidismo secundario.

- Número de atributos *P*: 29

La información detallada de cada uno de los atributos se encuentra en el apéndice C.2

6.1.3. Base de datos de Cleveland

La base de datos contiene variables que pueden inferir la presencia de enfermedad cardíaca en el paciente. Los posibles diagnósticos se representan con un entero valorado entre 0 (sin presencia) y 4 (diferentes enfermedades cardíacas). Los experimentos que se encuentran en la literatura con esta base de datos se han concentrado simplemente en intentar distinguir la presencia (valores 1,2,3,4) o ausencia (valor 0) de enfermedad cardíaca. La procedencia de éstos datos es de V.A. Medical Center, Long Beach and Cleveland Clinic Foundation: Robert Detrano, M.D., Ph.D. Esta base de datos contiene 76 atributos, pero la mayoría de los experimentos publicados utilizan los 14 atributos más relevantes.

[2] https://archive.ics.uci.edu/ml/datasets/Thyroid+Disease

- Fuente: [69]. [3]
- Número de instancias *N*: 303.

  164 de la clase 1, ausencia de enfermedad cardíaca.

  55 de la clase 2, estrechamiento medio de uno de los vasos mayores o importantes.

  36 de la clase 3, estrechamiento moderado de uno de los vasos mayores o importantes.

  35 de la clase 4, estrechamiento severo de uno de los vasos mayores o importantes.

  13 de la clase 5, estrechamiento muy severo de uno de los vasos mayores o importantes.
- Número de atributos *P*: 76

Información de los 14 atributos

1. #3 edad
2. #4 sexo
3. #9 cp
4. #10 trestbps
5. #12 chol
6. #16 fbs
7. #19 restecg
8. #32 thalach
9. #38 exang
10. #40 oldpeak
11. #41 slope
12. #44 ca
13. #51 thal
14. #58 num — Clases

La información detallada de cada uno de los atributos se encuentra en el apéndice C.3

[3] https://archive.ics.uci.edu/ml/datasets/Heart+Disease

6.1.4. Base de datos enfermedades cardíacas otros hospitales

Base de datos que contiene la información de enfermedades cardíacas de diferentes hospitales:

- Hungarian Institute of Cardiology. Budapest: Andras Janosi, M.D.
- University Hospital, Zurich, Switzerland: William Steinbrunn, M.D.
- University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D.

Contiene la misma estructura de los atributos de la base de datos del hospital de Cleveland. Para los experimentos del presente trabajo se denomina Cleveland ampliado.

- Fuente: [69]. [4]
- Número de instancias N: 830.

  371 de la clase 1, ausencia de enfermedad cardíaca.

  180 de la clase 2, estrechamiento medio de uno de los vasos mayores o importantes.

  123 de la clase 3, estrechamiento moderado de uno de los vasos mayores o importantes.

  116 de la clase 4, estrechamiento severo de uno de los vasos mayores o importantes.

  40 de la clase 5, estrechamiento muy severo de uno de los vasos mayores o importantes.
- Número de atributos P: 76

La información detallada de cada uno de los atributos se encuentra en el apéndice C.3

[4] https://archive.ics.uci.edu/ml/datasets/Heart+Disease

6.2. Medidas de desempeño

En esta sección se describen las medidas supervisadas empleadas para evaluar el desempeño de los experimentos propuestos, basados en técnicas de aprendizaje de máquina.
Para esta sección se considerará un conjunto de N datos, cada uno de dimensión D (con d características) y organizados en una matriz $\boldsymbol{X} \in \mathbb{R}^{N \times D}$.

6.2.1. Técnicas de validación

La finalidad de cualquier técnica de aprendizaje supervisado es generalizar a partir de un conjunto finito de datos, para los cuales se conoce el valor de salida o la clase a la cual pertenece. Sin importar la aplicación que se esté desarrollando o sobre la cual se esté estudiando, obtener un conjunto de datos que presente todos los posibles casos de salida es prácticamente imposible e inviable, lo más usual es contar con un conjunto de datos con un número de muestras representativas, en nuestro caso N, etiquetadas y a partir del cual se debe entrenar el sistema con la finalidad de generalizar, es decir, que tenga un buen desempeño de clasificación de datos nuevos y desconocidos.
La obtención de datos nuevos, dependiendo del problema de estudio, puede ser lento o complejo, en el campo de aprendizaje es muy común extraer un subconjunto a partir de los datos que se tienen a la mano y usarlos como datos nuevos o desconocidos.
La Figura 6.1 muestra un primer enfoque orientado a la validación de resultados, en el cual un conjunto de datos se divide en dos subconjuntos: el primero, llamado entrenamiento, almacena un porcentaje $p\,\%$ de los datos junto con sus etiquetas y el segundo, llamado prueba, almacena el $(100 - p)\,\%$ de los datos, además de las etiquetas respectivas $\boldsymbol{y}$, con estas se debe tener un cuidado especial como se explicará a continuación.

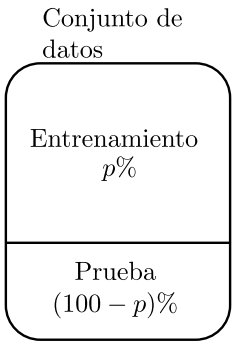


Figura 6.1.: División de un conjunto de datos en entrenamiento y prueba

Con el conjunto entrenamiento y sus etiquetas se estiman los parámetros necesarios para la técnica de clasificación a usar, es decir, se entrenará un sistema de aprendizaje supervisado. El conjunto de prueba se usará sin etiquetas, con el fin de simular datos nuevos, desconocidos

y lo más importante, que no han sido usados para la fase de entrenamiento. Estos datos sin etiquetas se entregarán al clasificador previamente entrenado con el conjunto entrenamiento, lo cual dará como respuesta un conjunto de etiquetas estimadas $\hat{\boldsymbol{y}}$.

6.2.2. Medidas derivadas de la matriz de confusión

Teniendo un vector de etiquetas estimadas por un clasificador y un vector de etiquetas reales o ground truth previamente conocidas, se puede realizar una comparación entre ellas buscando evaluar el desempeño del clasificador. Entre las medidas que existen, las más comunes son las derivadas a partir de la matriz de confusión. Esta matriz contiene la relación entre dos vectores de etiquetas biclase, de la siguiente manera:
Sean + y − las dos etiquetas de la clase, se tiene entonces cuatro casos:

- True Positive (T_P) - Verdaderos Positivos, es el número de datos pertenecientes a la clase positiva + y clasificados correctamente como de la clase positiva.
- True negative (T_N) - Verdaderos Negativos, es el número de datos pertenecientes a la clase negativa − y clasificados correctamente como de la clase negativa.
- False positive (F_P) - Falsos Positivos, es el número de datos pertenecientes a la clase negativa, pero erróneamente clasificados como de la clase positiva.
- False negative (F_N) - Falsos Negativos, es el número de datos pertenecientes a la clase positiva, pero erróneamente clasificados como de la clase negativa.

Lo cual se puede apreciar en la siguiente matriz de confusión:

		Estimadas	
		+	-
Reales	+	TP	FN
	-	FP	TN

Para el caso de clasificación multiclase las medidas serían de la siguiente forma:

- True Positive (T_P) - Verdaderos Positivos, es el número de datos pertenecientes a la clase de interés y clasificados correctamente.
- True negative (T_N) - Verdaderos Negativos, es la suma de los datos no pertenecientes a la clase de interés y clasificados correctamente como no pertenecientes a la clase de interés.
- False positive (F_P) - Falsos Positivos, es la suma de los datos no pertenecientes a la clase de interés, pero erróneamente clasificados como la clase de interés.

– False negative (F_N) - Falsos Negativos, es el número de datos pertenecientes a la clase de interés, pero erróneamente clasificados como no pertenecientes.

Lo cual se puede apreciar en la siguiente matriz de confusión:

		Clasificación			
		Clase 1	Clase 2	Clase 3	Clase 4
Clase Real	Clase 1	Tp_1	E_{12}	E_{13}	E_{14}
	Clase 2	E_{21}	Tp_2	E_{23}	E_{24}
	Clase 3	E_{31}	E_{32}	Tp_3	E_{34}
	Clase 4	E_{41}	E_{42}	E_{43}	Tp_4

A partir de los valores calculados en la matriz, la sensibilidad (Se), especificidad (Sp) y exactitud (Acc) se estiman como:

$$Se = \frac{T_P}{T_P + F_N}, \tag{6.1}$$

$$Sp = \frac{T_N}{T_N + F_P}, \tag{6.2}$$

$$Acc = \frac{T_N + T_P}{T_N + F_P + T_P + F_N}. \tag{6.3}$$

La sensibilidad y especificidad cuantifican la proporción de elementos de la clase positiva o clase de interés (+) y negativa o no pertenecientes a la clase de interés (−) que son clasificados correctamente, respectivamente. La exactitud entrega una relación de los datos correctamente clasificados con respecto al número total de datos del conjunto de prueba [70].

6.2.3. Validación cruzada

Al dividir el conjunto de datos en entrenamiento y prueba, donde p normalmente se usa como 60, 70 u 80, puede ocurrir que se llegué a un resultado que depende de la división del conjunto de datos, la cual se basa en la aleatoriedad, es decir, se puede llegar a un resultado sobresaliente o deficiente debido al azar. Con el fin de evitar este inconveniente, surgen algunas estrategias, entre ellas la validación cruzada, una forma común de evaluar el desempeño de un clasificador o validar sus parámetros. Esta técnica divide el conjunto de datos en k-folds o k subconjuntos. Se realizan k iteraciones, en donde cada k-ésima iteración se usa como conjunto de prueba y los demás $(k-1)$ subconjuntos son utilizados como conjunto de entrenamiento. Finalmente se tendrán k vectores de etiquetas estimados (uno por cada iteración usada como prueba), lo cual permitirá obtener k matrices de confusión con los valores de desempeño basados en ellas, permitiendo tener un valor más cercano de cómo se desempeñará el clasificador al recibir nuevos datos.

En la Figura 6.2 se puede apreciar un ejemplo, donde el conjunto de datos es dividido en 5 subconjuntos. Y por ejemplo, en la tercera iteración, el Fold-3, será usado como prueba, mientras los Folds 1,2,4 y 5 serán usados como conjunto de entrenamiento. Así se obtendrá una matriz de características de la cual se podrá estimar la sensibilidad, especificidad y exactitud. Al hacer todas las iteraciones se obtendrán cinco valores de sensibilidad, cinco valores de especificidad y cinco de exactitud, lo que permitirá computar un valor medio y una medida de dispersión [70].

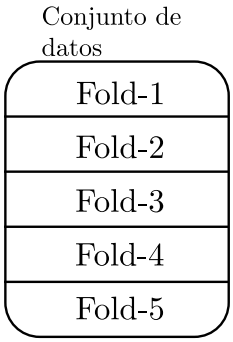
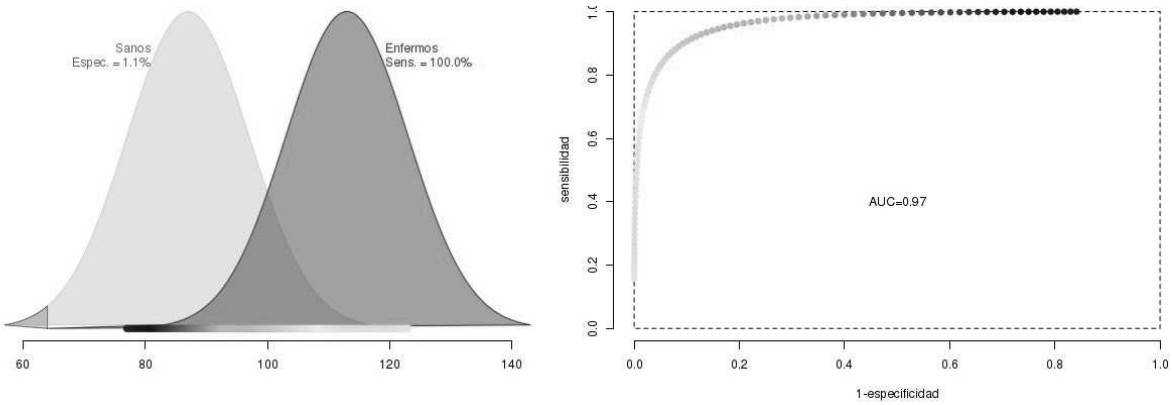


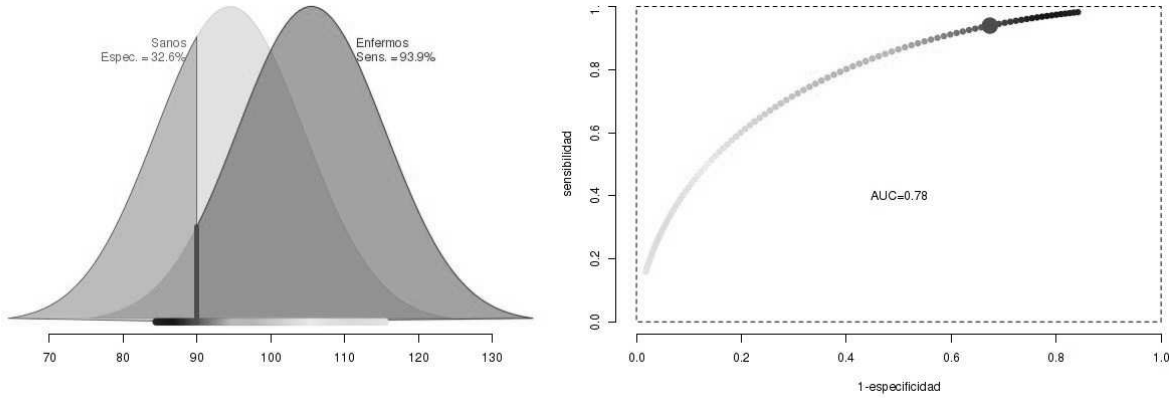
Figura 6.2.: Validación cruzada para 5 subconjuntos

6.2.4. Curvas ROC (Receiver-Operating Characteristic)

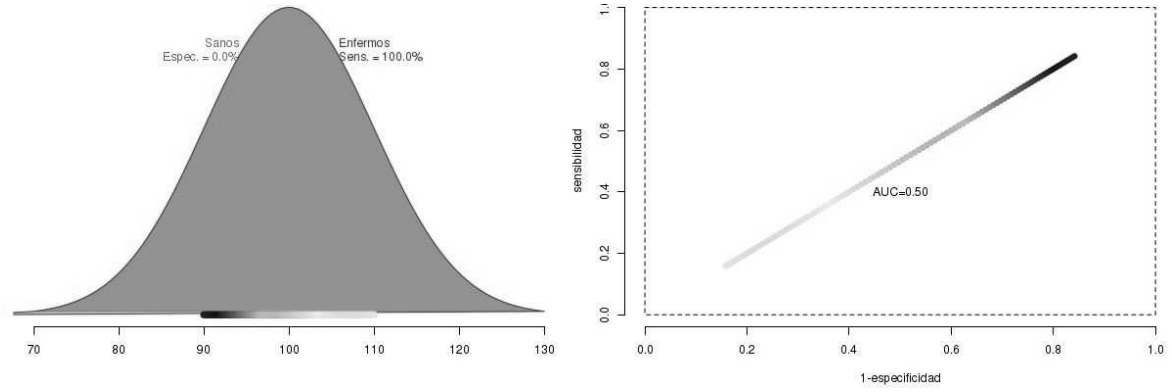
La característica operativa del receptor ROC es una técnica de visualización y selección de clasificadores basado en su desempeño. Presenta la sensibilidad de una prueba diagnóstica que produce resultados continuos en función de los falsos positivos (complementario de la especificidad), para distintos puntos de corte. Otra interpretación de ésta curva es la representación de la razón o ratio de verdaderos positivos (T_P) frente a la razón o ratio de falsos positivos (F_P) [71]



(a) Separación entre sanos y enfermos alta



(b) Separación entre sanos y enfermos intermedia



(c) Separación entre sanos y enfermos baja

Figura 6.3.: Gráficos de separación entre los grupos de enfermos y sanos versus Área bajo la Curva (AUC)

La separación entre los grupos de enfermos y sanos representa la capacidad de clasificar sanos como sanos y enfermos como enfermos. En la Figura 6.3 se puede apreciar como diferentes separaciones entre grupos de datos generan diferentes curvas ROC. La mejor curva esta dada por grupos con una separación alta. Uno de los parámetros que se utiliza para evaluar el desempeño del clasificador es el área bajo la curva (AUC), que se puede interpretar como la probabilidad que se tiene de clasificar correctamente.
En [72], se definen los valores entre los cuales un AUC se considera un buen o un mal test de validación:

- Entre 0.5-0.6 es un mal test
- Entre 0.6-0.75 es regular
- Entre 0.75-0.9 es un buen test
- Entre 0.9-0.97 es muy bueno
- entre 0.97-1 es excelente

6.3. Experimentos realizados

La preparación de los datos para los experimentos que se llevan a cabo en este trabajo, genera un conjunto de datos más pequeño que el original, lo cual mejora el porcentaje de clasificación.

6.3.1. Pre-proceso

Se realiza un estudio comparativo de diferentes métodos de selección de atributos sobre las bases de datos utilizadas en el marco de esta investigación. Se calcula el error de clasificación supervisada antes y después de la selección con diferentes técnicas, obteniendo como resultado, que el mejor algoritmo de selección de atributos es el CFS junto con el método de búsqueda best-first. La descripción detallada del experimento y los resultados se pueden observar en el apéndice B
Después de la aplicación del algoritmo de selección de variables o atributos sobre cada una de las bases de datos, se alcanza una reducción de dimensión en cada una de ellas, de tal forma que el número de atributos P pasa a ser D, donde $D < P$:

- En la base de datos de cardiotocografía, la dimensión D es 10 y los atributos seleccionados son: AC, FM, UC, DP, ASTV, D, E, LD, FS, CLASS.
- En las bases de datos de arritmias la dimensión D pasa a tener un valor de 7 atributos, los cuales son: Cp, Thalach, Exang, Oldpeak, Slope, Ca, Thal
- En la base de datos de hipotiroidismo la dimensión D es 5 y los atributos más relevantes son: Enfermo, bocio, TSH, medida T3, TT4

Una vez se seleccionan las variables más importantes para la clasificación, ya se cuenta con la estructura necesaria para trabajar con cada una de las bases de casos. Los experimentos se describen en el presente capítulo.

6.3.2. Experimento: Validación de clasificadores con reducción de dimensión

Pre-proceso

Reducción de dimensión: Con el objetivo de mejorar la inspección visual y el desempeño de clasificación, se aplican dos técnicas de reducción de dimensiones: LE - *Laplacian Eigenmaps* y t-SNE -*t-distributed Stochastic Neighbor Embedding*.
Utilizando LE y t-SNE, la base de casos de cardiotocografía se reduce a espacios de 2, 3, 5 y 8 dimensiones. De la misma forma, las bases de casos de Cleveland se reducen a espacios

de 2, 3 y 5 dimensiones. Se comparan los resultados aplicando la reducción de dimensiones contra el conjunto de datos $\boldsymbol{X}$ completo.

Entrenamiento de clasificadores: Se consideran algunos clasificadores multiclase representativos debido a sus características:

- K-NN (K-Nearest Neighbor o K-vecinos más cercanos), basado en distancias geométricas.
- Redes neuronales artificiales (ANN), basado en búsqueda heurística.
- Máquinas de vectores de soporte (SVM), basado en un modelo de distancias a un hiperplano o un conjunto de hiperplanos.
- Parzen (PC), basado en densidades.

Se entrena cada uno de los clasificadores con las bases de casos de cardiotocografía y con la base de casos de arritmias de Cleveland.
Se describe a continuación el marco experimental:

- **K-NN:** Esta técnica de clasificación basada en muestras, necesita el valor del número de vecinos (K), para el presente experimento este parámetro se optimiza mediante de la estrategia *leave-one-out*.
- **ANN:** Esta técnica de clasificación heurística requiere un número de unidades o neuronas por capa oculta. Para este experimento, se utiliza una única capa oculta y el número de neuronas se calcula a partir de los datos, como la mitad del número de muestras dividido por el número de dimensiones más el número de clases.
- **SVM:** Este método de clasificación aprovecha el truco kernel para calcular el hiperplano no lineal más discriminante entre clases. Por lo tanto, su desempeño depende en gran parte de la selección y ajuste del tipo de kernel. Para este experimento, se selecciona el kernel Gaussiano dada su habilidad de generalización y su parámetro de ancho de banda, se sintoniza mediante la regla de Silverman [73].
- **PC:** Este método de clasificación basado en probabilidades requiere un parámetro de suavizado para el cálculo de la distribución Gaussiana, dicho parámetro se optimiza durante el entrenamiento.

6.3.3. Experimento: Metodología de CBR aplicando clasificadores supervisados con balanceo de datos

El primer experimento compara el error de clasificación dado por 4 clasificadores aplicados a dos bases de casos, y contrasta dichos resultados con los obtenidos al aplicar técnicas

de reducción de dimensiones. Teniendo este experimento como base y con el objetivo de evaluar el desempeño de los clasificadores dentro de la metodología de CBR, se construye un sistema que une la etapa de recuperación y adaptación utilizando las ventajas que ofrecen los clasificadores.

Pre-proceso

Se realiza un balanceo de datos con la técnica SMOTE *Syntetic Minority Over-sampling Technique*, algoritmo de sobremuestreo que genera instancias "sintéticas" o artificiales. Se equilibran las muestras de datos utilizando la regla del vecino más cercano [74]. Se puede apreciar en el algoritmo 5, la aplicación de la técnica.
En la Figura 6.4 se puede observar un diagrama resumen del algoritmo SMOTE.

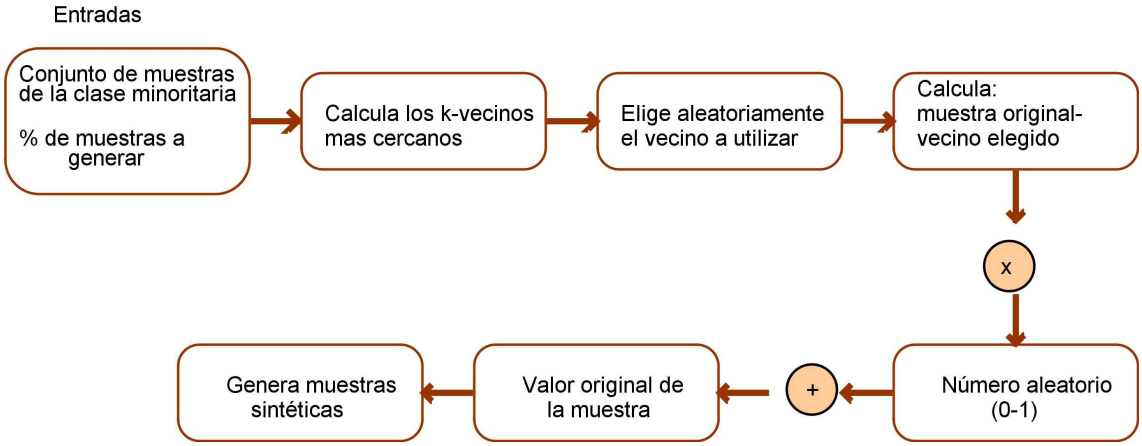


Figura 6.4.: Diagrama de bloques del algoritmo SMOTE para generar muestras sintéticas de la clase minoritaria

Entrenamiento de clasificadores: La configuración de los clasificadores se realiza de la misma forma que en el experimento anterior 6.3.2. El 80 % de los datos se utiliza para el entrenamiento y el 20 % para realizar las pruebas y hallar el error de clasificación.

Recuperación y adaptación

Se fusionan las etapas de recuperación y adaptación, esto es posible gracias a que al aplicar la clasificación multiclase y dar el resultado en términos de probabilidades de pertenencia del nuevo caso a cada una de las clases; no es necesario recuperar los vecinos más cercanos y se cuenta además con una solución ya adaptada. Cada uno de los casos de prueba, es clasificado por cada uno de los clasificadores *K*-NN, ANN, SVM y PC, obteniendo como resultado la asignación de la etiqueta de clasificación por parte del clasificador. Además se

Algorithm 5 SMOTE (*T*, *N*, *K*)

Require: Número de muestras de la clase minoritaria *T*; Cantidad de SMOTE *N* %; Número de vecinos más cercanos *K*.

Ensure: (N / 100)*T Muestras sintéticas de la clase minoritaria

if N ¡100 **then**
  Aleatorizar las muestras de la clase minoritaria *T*
  (*N* / 100)**T*
  N = 100
else
  N = (int)(*N* / 100)(* La cantidad de SMOTE por defecto son múltiplos enteros de 100 *)
  K = Número de vecinos cercanos
  numattrs = Número de atributos
  Sample [ ] [ ]: Arreglo para muestras minoritarias originales
  newindex: Guarda un recuento del número de muestras sintéticas generadas, inicializado en 0
  Synthetic [ ] [ ]: Arreglo para las muestras sintéticas (* Calcula *K* vecinos más cercanos para cada muestra de la clase minoritaria solamente *)
end if
for *l*=1 to *T* **do**
  Calcula los *K* vecinos más cercanos para cada muestra de la clase minoritaria solamente
  Populate(*N*, *l*, nnarray)
end for
Populate(*N*, *l*, nnarray) (* Función para generar las muestras sintéticas *)
while *N* ≠ 0 **do**
  Escoge un número aleatorio entre 1 y *K*, llamado nn. Este paso escoge uno de los *K* vecinos más cercanos de *l*
  for *attr*=1 to *numattrs* **do**
    Calcular: *dif = Sample[nnarray[nn]][attr] – Sample[l][attr]*
    Calcular: *gap = número aleatorio entre 0 y 1*
    *Synthetic[newindex][attr] = Sample[l][attr] + gap*dif*
  end for
  newindex++
  N = N-1
end while
return (* Final de Populate *)

calcula la probabilidad de pertenencia a cada clase usando el método de Parzen, detallado en la sección 4.6.

Aprendizaje

Una de las principales características de un sistema basado en CBR es poder recordar los nuevos casos y su solución; para ello es fundamental poder almacenar estos casos en la base de casos, e ir aumentando y enriqueciendo gracias a las soluciones de los nuevos problemas. La forma de estructurar la base de casos y las políticas de aprendizaje del sistema, facilitarán el buen funcionamiento; debido a esto, el primer problema que debe tratar un sistema de aprendizaje es decidir de qué casos se aprende. La eficiencia de un sistema de CBR se puede degradar cuando el número de casos crece excesivamente y por lo tanto, se debe evitar incluir casos que no aporten información nueva al sistema. El rango de posibilidades va desde los sistemas que, de forma autónoma deciden qué casos deben incluir hasta los que delegan esta posibilidad en el mismo usuario [75].
Para el presente experimento la decisión sobre si el nuevo caso debe añadirse o no, está relacionada con la probabilidad de pertenencia a cada clase; si dicha probabilidad es muy alta para una clase determinada podría interpretarse como que el sistema de CBR posee suficiente información para decidir a que categoría pertenece el nuevo caso, por lo tanto no sería necesario incluirlo. El sistema aprende de aquellos casos que se clasifican correctamente y su probabilidad de pertenencia a la clase esta entre 0.6-0.9, por tanto no cuenta con la información suficiente para dar una mayor probabilidad. Además aprende de aquellos casos que se clasifican mal; para este experimento los datos de prueba cuentan con la clase real a la que pertenecen, permitiendo simular la acción del experto de clasificarlo y alimentar la base de casos.

6.3.4. Experimento: Aplicando AdaBoost y Random forest

Con el objetivo de observar el rendimiento al aplicar varios clasificadores en serie, el presente experimento utiliza los algoritmos de clasificación de Random forest y AdaBoost sobre las bases de casos de cardiotocografía, Cleveland e hipotiroidismo.

Pre-proceso

No se realiza ninguna acción sobre los casos, solamente se lleva a cabo el algoritmo de selección de variables descrito en la sección 6.3.1.
Entrenamiento de clasificadores: Para llevar a cabo el proceso de entrenamiento, se utiliza el 80 % de los casos y el restante 20 % se reserva para realizar las pruebas.
Para cada uno de los algoritmos es necesario configurar las variables de entrada que se definen de la siguiente forma, para Random forest:

- El número de árboles de decisión a utilizar es de 100.
- Tamaño de los subconjuntos de características que se configuran por nodo, para el presente experimento el valor es de una característica por nodo.

Y para AdaBoost:

- El clasificador a utilizar, en este caso se realizan pruebas con SVM, ANN, PC y *K*-NN.
- Regla de combinación; votación por pesos (wvote), regla de votación (vote), regla de promedio (mean), regla del producto (prod), regla del máximo (max), regla del mínimo (min) y regla de la mediana (median). En el experimento se comparan los resultados obtenidos por todas y cada una de estas reglas de combinación.
- El número de clasificadores que conforman el algoritmo, el valor configurado es de 100.

6.3.5. Experimento: Metodología de CBR aplicando clasificadores supervisados

En este experimento se retoma el experimento 6.3.3, pero ahora el objetivo es validar el desempeño al utilizar clasificadores multiclase en las fases de recuperación y adaptación de un sistema basado en CBR, sin balanceo de datos.

Pre-proceso

Entrenamiento de clasificadores: La parametrización de cada uno de los clasificadores se realiza de la misma forma que esta descrita en 6.3.2. Para el presente experimento se utilizan los clasificadores basados en ANN, PC y *K*-NN. Se elimina el clasificador basado en SVM debido al alto coste computacional que se ve traducido en mayor tiempo de entrenamiento, 3 veces más que los demás clasificadores. Adicionalmente si se analizan los resultados del experimento 7.2, el clasificador basado en SVM tiene el error más alto para la base de datos de Cleveland, con un valor de cero en la medida de sensibilidad, en 3 de las 5 clases existentes.

En el experimento se trabaja con tres grupos diferentes de entrenamiento y de pruebas:

- 30 % de entrenamiento - 70 % de prueba: al que se denomina en varias partes del documento como 30-70.
- 50 % de entrenamiento - 50 % de prueba: se denomina 50-50.
- 70 % de entrenamiento - 30 % de prueba: se encuentra en varias partes del documento como 70-30.

Recuperación y adaptación

Al igual que en el experimento 6.3.3, se fusionan las etapas de recuperación y adaptación, calculando las probabilidades de pertenencia del nuevo caso a cada clase; pero en esta ocasión las probabilidades se calculan aplicando la propuesta desarrollada por Duin en [67].

En el algoritmo 6, se pueden observar los pasos que se siguen desde la aplicación del clasificador al nuevo caso hasta la obtención de las probabilidades de pertenencia a cada clase. La salida resultante de evaluar el nuevo caso con el clasificador, es una matriz de densidades o de distancias. Dicha matriz se utiliza en el cálculo de probabilidades; si el clasificador es K-NN se hace uso del algoritmo 3, descrito en la sección 5.1.2, si es el clasificador basado en ANN o PC, se normalizan los datos de la matriz de salida y se utilizan como probabilidades, de esta forma se garantiza que la suma sea 1 [67].

Algorithm 6 Algoritmo de recuperación y adaptación

Require: Nuevo caso $\widetilde{\boldsymbol{x}}$, $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N)^\top$

Aplicar clasificador K-NN, ANN y PC a $\widetilde{\boldsymbol{x}}$, obteniendo como resultado una matriz de distancias o de densidades $\boldsymbol{D}_{\widetilde{\boldsymbol{x}}}$

Estimar $p(1, \ldots, C)$

if clasificador=K-NN **then**

Utilizar algoritmo 3

return $p(1, \ldots, C|\widetilde{\boldsymbol{x}})$

else

Normalizar los datos de la matriz de salida

return $p(1, \ldots, C|\widetilde{\boldsymbol{x}})$

end if

Revisión y aprendizaje

Se simula la etapa de revisión con los casos de prueba. Evaluando uno a uno los casos dentro del sistema basado en CBR, se compara la respuesta del sistema contra la etiqueta real que se tiene de cada caso. Si la clasificación es incorrecta; el caso pasa a la fase de aprendizaje, donde se incluye el caso y su correspondiente etiqueta en la base de casos con el fin de aprender de la experiencia. Si la clasificación es correcta, y el porcentaje de probabilidad es menor de 0.9; el caso también pasa a la fase de aprendizaje. El algoritmo 7 detalla los pasos que se siguieron en las fases de revisión y aprendizaje.

Algorithm 7 Algoritmo llevado a cabo en el experimento para revisión y aprendizaje

Require: Nuevo caso $\widetilde{x}$, $\widetilde{y}$ y $p(\ell|\widetilde{x})$
  if $\widetilde{y}$= arg máx$_p$ $p(\ell|\widetilde{x})$ **then**

    if arg máx$_P$ $P(\ell|\widetilde{x}) \geq 0{,}6$ *and* $\leq 0{,}9$ **then**
      El caso entra en la base de casos
      Se reentrena el clasificador
    else if arg máx$_P$ $P(\ell|\widetilde{x}) \geq 0{,}9$ **then**
      El sistema no realiza ninguna acción, debido a que es capaz de clasificar perfectamente el caso
    else
      El caso entra a una base de datos de cuarentena
    end if
  else
    El caso $\widetilde{x}$ y $\widetilde{y}$ entran en la base de casos
    Se reentrena el clasificador
  end if

6.3.6. Experimento: Metodología de CBR aplicando clasificadores supervisados en cascada

Pre-proceso

En el anterior experimento se compara la aplicación de 3 clasificadores en la etapa de recuperación y adaptación de un sistema basado en CBR; el sistema aprende de los casos donde la probabilidad de pertenencia esta entre 0.6 y 0.9 y de los casos clasificados erróneamente. El presente experimento tiene como base el experimento anterior 6.3.5, pero en esta ocasión se valida el comportamiento obtenido al aplicar clasificadores en cascada en la etapa de recuperación y adaptación; además el sistema aprende de todos aquellos casos que están bien clasificados y con probabilidad de pertenencia inferior a 0.9.

Entrenamiento de clasificadores:
Para el presente experimento se utilizan las 4 bases de casos; Cleveland, Cleveland ampliado, cardiotocografía e hipotiroidismo. Para cada una de ellas se llevan a cabo las siguientes tareas:

- La parametrización de cada uno de los clasificadores se realiza de la misma forma que esta descrita en 6.3.2, utilizando el 70 % de los casos para el entrenamiento y el 30 % para las pruebas.
- Se dividen los casos entre sanos (1) y enfermos (-1), los casos enfermos son todos

aquellos casos etiquetados con clases diferentes a 1. Con la matriz $\boldsymbol{X}$ y el vector de etiquetas biclase $\hat{\boldsymbol{y}}$, se entrenan tres clasificadores biclase: Uno basado en ANN, otro basado en PC y un tercero basado en *K*-NN.

- De la base de casos $\boldsymbol{X}$ se eliminan todos aquellos casos etiquetados como sanos, creándose una nueva base de casos $\tilde{\boldsymbol{X}} \in \mathbb{R}^{M \times D}$ y un nuevo vector de etiquetas $\tilde{\boldsymbol{y}} \in \mathbb{R}^{M}$, elementos con los cuales se entrenan tres clasificadores multiclase, que al igual que en el paso anterior están basados en ANN, PC y *K*-NN.

Recuperación y adaptación

Siguiendo con el modelo de los experimentos 6.3.3 y 6.3.5 se fusiona la etapa de recuperación con la de adaptación, el calculo de probabilidades se lleva a cabo como lo indica Duin en [67]; pero al tener dos clasificadores actuando en el sistema, se hace necesario hallar la probabilidad conjunta, la cual explicaremos más adelante.
Se realizan pruebas con los tres clasificadores biclase en cascada con los tres clasificadores multiclase de la siguiente forma:

- Clasificador 1.ANN como clasificador biclase y clasificadores multiclase: 1.1.ANN, 1.2.PC y 1.3.*K*-NN.
- Clasificador 2.PC como clasificador biclase y clasificadores multiclase: 2.1.ANN, 2.2.PC y 2.3.*K*-NN.
- Clasificador 3.*K*-NN como clasificador biclase y clasificadores multiclase: 3.1.ANN, 3.2.PC y 3.3.*K*-NN.

A cada uno de los casos de prueba se les aplica el algoritmo 8 de recuperación y adaptación:

- La base de casos de arritmias-Cleveland cuenta con 88 casos de prueba.
- La base de casos de Cleveland ampliado cuenta con 247 casos de prueba.
- La base de casos de cardiotocografía cuenta con 638 casos de prueba.
- La base de casos de hipotiroidismo cuenta con 1012 casos de prueba.

En el algoritmo 8, se puede observar como el nuevo caso $\widetilde{\boldsymbol{x}}$ es evaluado por cada uno de los clasificadores biclase, en el primer *FOR*, dando como resultado la probabilidad de pertenencia a la clase enfermo o a la clase sano. Después, el caso $\widetilde{\boldsymbol{x}}$ entra a un segundo *FOR* donde es evaluado por los clasificadores multiclase, y se obtiene como resultado la probabilidad de pertenencia a cada uno de los posibles diagnósticos. Para hallar la probabilidad conjunta

Algorithm 8 Algoritmo de recuperación y adaptación

Require: Nuevo caso $\widetilde{\boldsymbol{x}}$, $\boldsymbol{X}$, $\hat{\boldsymbol{y}}$, $\tilde{\boldsymbol{X}}$, $\tilde{\boldsymbol{y}}$
  for $j = 1$ hasta 3 **do**
    Utilizando $\widetilde{\boldsymbol{x}}$, $\boldsymbol{X}$, $\hat{\boldsymbol{y}}$ aplicar clasificador j biclase. Se obtiene como salida una matriz $\boldsymbol{D}_{\widetilde{\boldsymbol{x}}}$
    Estimar $p(1, -1)$
    if clasificador=K-NN **then**
      Utilizar algoritmo 3
      return $p(1, -1)$
    else
      Normalizar los datos de la matriz de salida
      return $p(1, -1)$
    end if
    for $c = 1$ hasta 3 **do**
      Utilizando $\widetilde{\boldsymbol{x}}$, $\tilde{\boldsymbol{X}}$, $\tilde{\boldsymbol{y}}$ aplicar clasificador c multiclase. Se obtiene como salida una matriz $\tilde{\boldsymbol{D}}_{\widetilde{\boldsymbol{x}}}$
      Estimar $p(2, \ldots, C)$
      if clasificador=K-NN **then**
        Utilizar algoritmo 3
        return $p(2, \ldots, C)$
      else
        Normalizar los datos de la matriz de salida
        return $p(2, \ldots, C)$
      end if
    end for
    Hallar $p(2, \ldots, C| - 1)$
  end for
  return $p(1), p(2, \ldots, C| - 1)$

$p(2,\ldots,C|-1)$, se multiplica la probabilidad de estar enfermo $p(-1)$ por las probabilidades que resultan de la evaluación del segundo clasificador $p(2,\ldots,C)$, debido a que al ser sucesos dependientes, se hace uso de la fórmula:

$$p(A \cap B) = p(A)p(B/A). \tag{6.4}$$

Los resultados se guardan en una matriz de probabilidades p, para posteriormente realizar las pruebas de desempeño.

Revisión y aprendizaje

En esta etapa se siguen los mismos pasos realizados en el experimento anterior sub-sección 6.3.5; excepto por el uso de dos clasificadores en lugar de uno. Adicionalmente el caso entra en la base de casos y los clasificadores son reentrenados, si se cumple la condición de estar bien clasificado el caso y que la probabilidad de pertenencia sea inferior a 0.9.

Algorithm 9 Algoritmo llevado a cabo en el experimento para revisión y aprendizaje

Require: Nuevo caso $\widetilde{x}$, $\widetilde{y}$ y $p(\ell|\widetilde{x})$
  if $\widetilde{y}$= arg máx$_p$ $p(\ell|\widetilde{x})$ **then**
    if arg máx$_P$ $P(\ell|\widetilde{x}) \leq 0{,}9$ **then**
      El caso $\widetilde{x}$ y $\widetilde{y}$ entran en la base de casos
      Se reentrenan los clasificadores
    else
      El sistema no realiza ninguna acción, debido a que es capaz de clasificar perfectamente el caso
    end if
  else
    El caso $\widetilde{x}$ y $\widetilde{y}$ entran en la base de casos
    Se reentrenan los clasificadores
  end if

6.3.7. Conclusiones

En el presente capítulo se describen detalladamente cada uno de los experimentos que se llevaron a cabo para el desarrollo de la investigación.

- Las bases de datos seleccionadas, al tener diferente número de instancias, un amplio número de atributos y pertenecer a diferentes diagnósticos médicos, permiten evaluar si un sistema basado en CBR para diagnóstico médico es posible de generalizar.

- Una extensión de la matriz de confusión para problemas biclase, admite la evaluación de cada una de las clases en problemas multiclase y de esta forma hacer uso de las medidas de sensibilidad, especificidad y curvas ROC; las cuales son medidas de desempeño representativas en diganóstico médico.

- Los experimentos que se plantean en el presente capítulo, tienen la finalidad de aportar al cumplimiento de los objetivos de esta investigaicón. En el primer experimento se realiza un estudio comparativo del comportamiento de 4 clasificadores aplicados a las bases de datos seleccionadas y adicionalmente se comparan algoritmos de reducción de dimensiones, que permiten visualizar los datos en 2D y 3D, lo cual facilita el análisis de la separabilidad de las clases en cada uno de los conjuntos de datos; aportando de esta forma al cumplimiento del primero de los objetivos específicos propuestos.

- En el segundo y cuarto experimento, se diseña, se desarrolla y se valida un sistema basado en CBR que fusionan las etapas de recuperación y adaptación, aplicando diferentes algoritmos de clasificación. En el experimento 6.3.3, se realiza balanceo de datos, mientras que en el experimento 6.3.5 se suprime este paso. Con estos dos experimentos se aporta al cumplimiento del segundo de los objetivos específicos propuestos.

En el siguiente capítulo, se pueden apreciar los resultados obtenidos de los experimentos propuestos en el presente capítulo. El análisis de dichos resultados permitirá la implementación de un sistema completo basado en CBR que reúna las mejores características y que integre etapas adecuadas de pre-proceso, recuperación y adaptación de casos de múltiples clases, para dar una respuesta eficiente del posible diagnóstico al especialista. Cumpliendo de esta forma con el tercer objetivo específico y en consecuencia con el objetivo general de la propuesta de investigación.

Parte IV.

COMENTARIOS FINALES

7. RESULTADOS Y DISCUSIÓN

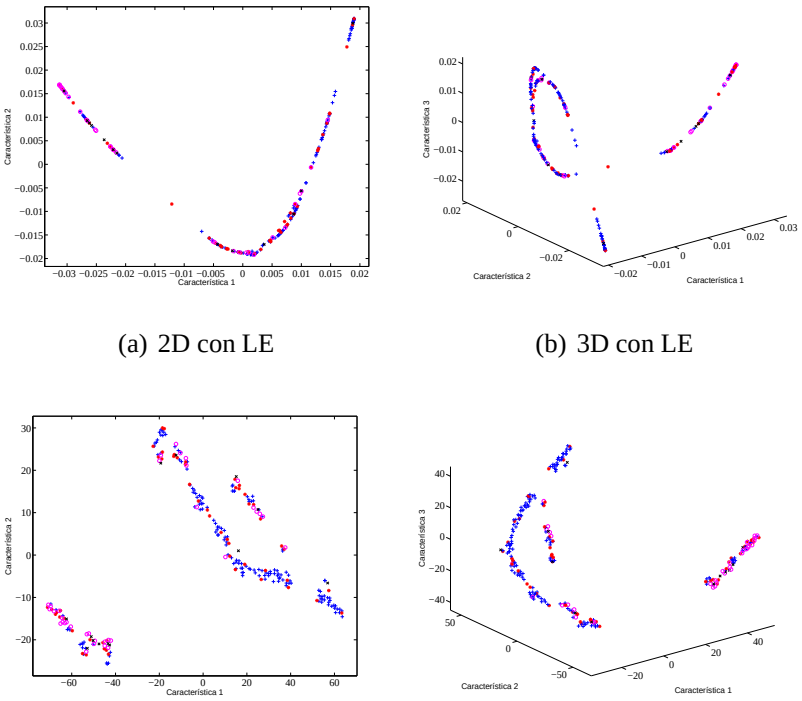
Los resultados obtenidos de los experimentos descritos en el capítulo anterior, se muestran a través de tablas y figuras en el presente capítulo. Al finalizar cada uno de los experimentos, se pueden ver conclusiones preliminares, con el fin de facilitar el seguimiento paso a paso.

7.1. Resultados: Validación de clasificadores

Para cada una de las técnicas de clasificación, se utiliza un procedimiento de validación cruzada con 20 subconjuntos o folds, con el propósito de evitar el sesgo dado por el azar. Como medida de desempeño se usa el error medio de clasificación estándar.
Los errores de clasificación obtenidos con los diferentes clasificadores seleccionados, unido a cada uno de los valores de reducción de dimensiones establecidos, y para cada una de las bases de casos; se muestran en la Tabla 7.1. Además se pueden ver en la Figura 7.1, los gráficos de dispersión de los datos en 2D y 3D para la base de datos de Cleveland y la de cardiotocografía. Los boxplot o diagramas de caja resultado de los errores de clasificación, se muestran en la Figura 7.4. El cálculo del valor medio y la desviación estándar se obtienen mediante la validación cruzada.

DB	Téc. Red	dim	*K*-NN	ANN	SVM	PC
Cleveland	t-SNE	2	0.381 ± 0.08	0.389 ± 0.067	0.389 ± 0.013	0.393 ± 0.093
		3	0.382 ± 0.06	0.367 ± 0.09	0.389 ± 0.013	0.393 ± 0.069
		5	0.397 ± 0.07	0.362 ± 0.089	0.389 ± 0.028	0.4 ± 0.087
		7	0.397 ± 0.07	0.347 ± 0.062	0.401 ± 0.029	0.393 ± 0.069
	LE	2	0.408 ± 0.069	0.393 ± 0.077	0.389 ± 0.013	0.393 ± 0.041
		3	0.397 ± 0.066	0.397 ± 0.075	0.389 ± 0.013	0.374 ± 0.047
		5	0.389 ± 0.067	0.404 ± 0.085	0.412 ± 0.036	0.389 ± 0.067
		7	0.389 ± 0.065	0.382 ± 0.065	0.397 ± 0.07	0.404 ± 0.064
Cardio tocografía	t-SNE	2	0.037 ± 0.015	0.084 ± 0.038	0.071 ± 0.017	0.077 ± 0.017
		3	0.036 ± 0.016	0.073 ± 0.02	0.054 ± 0.019	0.076 ± 0.018
		5	0.032 ± 0.017	0.088 ± 0.017	0.039 ± 0.019	0.075 ± 0.016
		8	0.035 ± 0.016	0.079 ± 0.016	0.033 ± 0.017	0.075 ± 0.019
		10	0.031 ± 0.017	0.082 ± 0.036	0.028 ± 0.016	0.076 ± 0.019
	LE	2	0.045 ± 0.014	0.078 ± 0.016	0.086 ± 0.017	0.102 ± 0.023
		3	0.054 ± 0.018	0.072 ± 0.015	0.061 ± 0.016	0.09 ± 0.02
		5	0.042 ± 0.014	0.075 ± 0.031	0.048 ± 0.014	0.09 ± 0.016
		8	0.041 ± 0.015	0.067 ± 0.019	0.038 ± 0.015	0.065 ± 0.016
		10	0.039 ± 0.015	0.06 ± 0.017	0.038 ± 0.013	0.063 ± 0.016

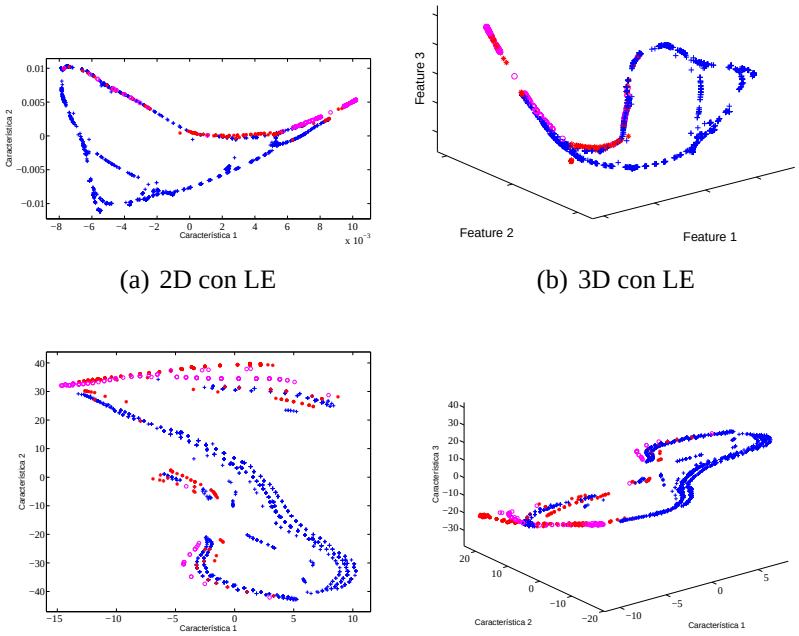
Tabla 7.1.: Desempeño de clasificación de la validación cruzada con 20 folds para las bases de casos y técnicas de dimensión consideradas



(a) 2D con LE (b) 3D con LE

(c) 2D con t-SNE (d) 3D con t-SNE

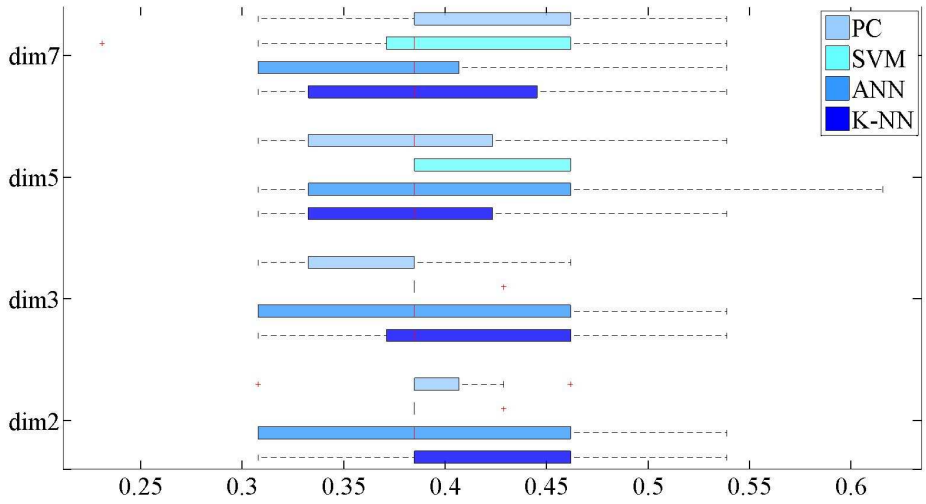
Figura 7.1.: Gráficos de dispersión en bajas dimensiones para la base de casos de Cleveland



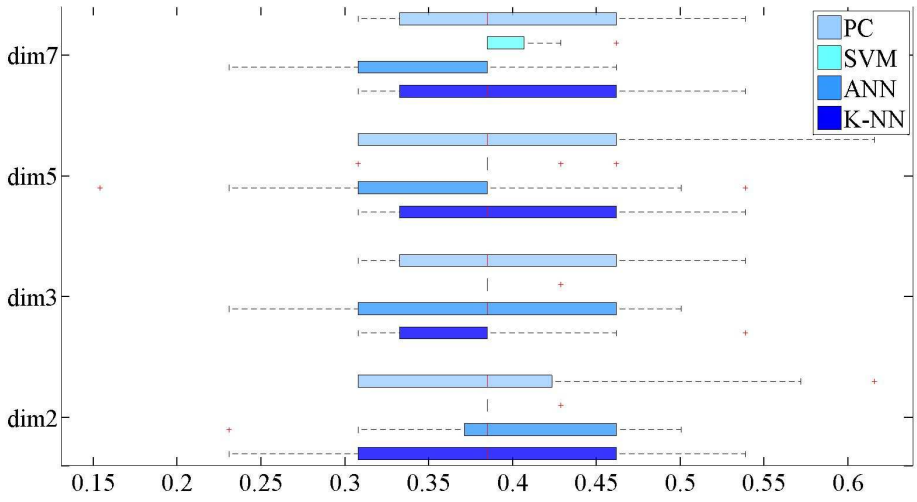
(a) 2D con LE (b) 3D con LE

(c) 2D con t-SNE (d) 3D con t-SNE

Figura 7.2.: Gráficos de dispersión en bajas dimensiones para la base de casos de cardiotocografía

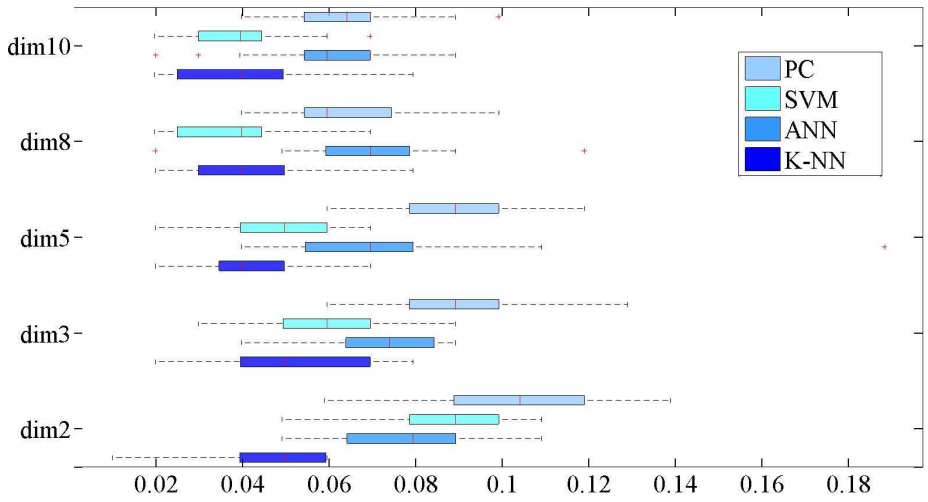


(a) Cleveland DB LE

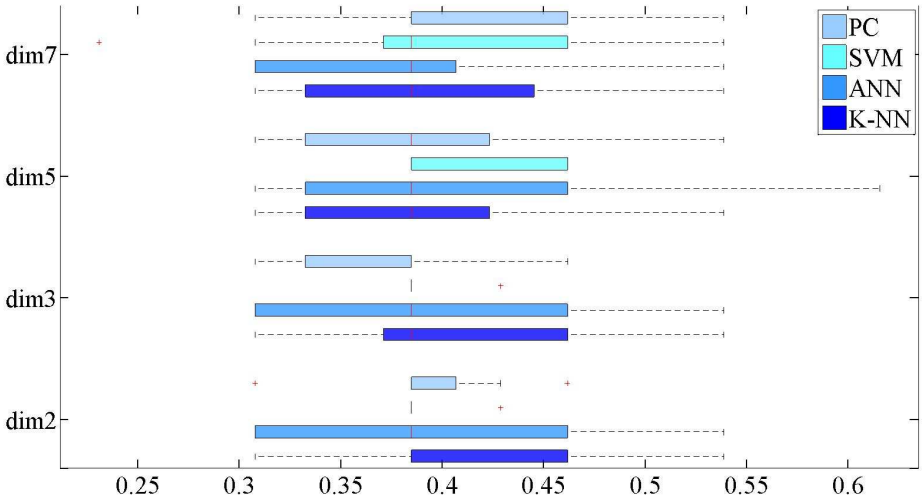


(b) Cleveland DB t-SNE

Figura 7.3.: Boxplot de los errores de clasificación para las técnicas de clasificación consideradas para la base de casos de Cleveland (Validación cruzada 20 folds)



(a) Cardiotocograms DB LE



(b) Cardiotocograms DB t-SNE

Figura 7.4.: Boxplot de los errores de clasificación para las técnicas de clasificación consideradas y las bases de casos de cardiotocografía (Validación cruzada 20 folds)

Como conclusiones preliminares se puede decir que:

- Cleveland representa un conjunto de datos complejo, el desempeño es bajo para todos los clasificadores. El mejor valor del error de clasificación es de: 0.347 ± 0.062, y se obtiene con el clasificador basado en ANN y con todos los atributos de la base de casos.

- No se hallaron diferencias en la clasificación, al reducir las dimensiones a 2,3 y 5. Sin embargo, existe una ganancia en el análisis visual de los datos como se puede apreciar en la Figura 7.1. Particularmente en 2 dimensiones (Figuras 7.1(a) y 7.1(c)) y 3 dimensiones (Figuras 7.1(b) y 7.1(d)).

- Los datos de la base de casos de Cleveland están altamente traslapados, en consecuencia los resultados obtenidos no son buenos. El resultado de la exactitud del clasificador SVM es de 0.603 ± 0.07, el cual no está lejos del resultado obtenido en [76], donde la exactitud de clasificación con 7 atributos es de 0.70.

- En el conjunto de datos de cardiotocografía, la separabilidad de las clases es evidente al verla en bajas dimensions, i.e. 2D y 3D, como se muestra de la Figura 7.2(a) a la Figura 7.2(d), lo que conlleva a resultados sobresalientes como se puede ver en la Tabla 7.1. Sin embargo, la reducción de dimensión a 2,3,5 y 8 no mejora sustancialmente el desempeño de clasificación, aunque sí mejora la visualización de los datos. Se puede apreciar que el mejor resultado se logró usando el clasificador SVM, con una exactitud de 0.972 ± 0.016, mejorando los resultados alcanzados en [77], en el cual obtienen un desempeño promedio de 0.9328.

- Al realizar una evaluación de estabilidad, se puede apreciar en la Figura 7.4, por los anchos de los boxplots de errores, que los clasificadores SVM y *K*-NN alcanzan los mejores resultados para los conjuntos de datos considerados.

7.2. Resultados: Metodología CBR aplicando clasificadores supervisados con balanceo de datos

Al igual que en el experimento anterior, se construyen tablas con los errores de clasificación de los diferentes clasificadores aplicados a las bases de casos de Cleveland y cardiotocografía. Las tablas 7.2 y 7.3 proporcionan dicha información, para los conjuntos de datos establecidos.

Prueba	SVM	ANN	PC	*K*NN
1	0.62	0.41	0.28	0.26
2	0.62	0.56	0.26	0.24
3	0.61	0.42	0.32	0.33
4	0.63	0.45	0.32	0.30
5	0.62	0.52	0.25	0.24
6	0.61	0.54	0.26	0.26
Promedio	0.61	0.48	0.28	0.27

Tabla 7.2.: Errores de los clasificadores con el conjunto de datos de Cleveland

Prueba	SVM	ANN	PC	*K*NN
1	0.016	0.016	0.016	0
2	0.016	0.033	0.033	0.016
3	0.016	0.050	0.033	0.066
4	0.050	0.033	0.033	0.050
5	0.050	0.033	0.050	0.033
6	0	0	0	0
Promedio	0.024	0.027	0.027	0.027

Tabla 7.3.: Errores de los clasificadores con el conjunto de datos de cardiotocografía

Validación del CBR

Para la base de casos de Cleveland, se toman 10 muestras de cada una de las 5 clases, para un total de 50. Se calculan los valores de sensibilidad y especificidad del sistema basado en CBR. Los resultados de estas medidas con cada uno de los clasificadores se pueden observar en las Tablas 7.4, 7.5, 7.6 y 7.7.

Medida	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Promedio
Se	1	0	1	0	0	0.4
Sp	0.98	1	0.28	1	1	0.85

Tabla 7.4.: Se y Sp, base de casos Cleveland con el clasificador SVM

Medida	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Promedio
Se	0.3	0.4	0.7	1	0.3	0.54
Sp	1	0.98	0.55	0.98	0.93	0.89

Tabla 7.5.: Se y Sp base de casos Cleveland con el clasificador ANN

Medida	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Promedio
Se	0.9	1	1	1	0.7	0.92
Sp	1	0.93	0.98	1	1	0.98

Tabla 7.6.: Se y Sp base de casos Cleveland con el clasificador PC

Para la base de casos cardiotocografía se utilizan 15 muestras de cada clase, para un total de 45. Los resultados de la Se y Sp para cada uno de los clasificadores se indican en las tablas 7.8, 7.9, 7.10 y 7.11.

Medida	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Promedio
Se	1	0.9	1	1	1	0.98
Sp	1	1	1	1	0.98	1

Tabla 7.7.: Se y Sp base de casos Cleveland con el clasificador K-NN

Medida	Clase 1	Clase 2	Clase 3	Promedio
Se	1	1	1	1
Sp	1	1	1	1

Tabla 7.8.: Se y Sp base de casos cardiotocografía con el clasificador SVM

Medida	Clase 1	Clase 2	Clase 3	Promedio
Se	1	0.93	1	0.98
Sp	0.97	1	1	0.99

Tabla 7.9.: Se y Sp base de casos cardiotocografía con el clasificador ANN

Medida	Clase 1	Clase 2	Clase 3	Promedio
Se	1	1	1	1
Sp	1	1	1	1

Tabla 7.10.: Se y Sp base de casos cardiotocografía con el clasificador PC

Medida	Clase 1	Clase 2	Clase 3	Promedio
Se	1	1	1	1
Sp	1	1	1	1

Tabla 7.11.: Se y Sp base de casos cardiotocografía con el clasificador K-NN

Como conclusiones preliminares se puede decir que:

- Los resultados de los errores de clasificación con el conjunto de datos de Cleveland, son negativos para todos los clasificadores. Con los clasificadores PC y K-NN se identifican los mejores resultados, siendo 0.24 el error más bajo.
- El clasificador basado en SVM da el error más alto para la base de casos de Cleveland y la sensibilidad es de cero en 3 de las 5 clases existentes.
- Con el conjunto de datos de Cardiotocografía se observan errores de clasificación pequeños con todos los clasificadores.
- En las tablas 7.4, 7.5, 7.6 y 7.7; donde se muestran los resultados de Se y Sp del sistema basado en CBR, utilizando los datos de prueba de la base de casos de Cleveland con 50 casos, es muy interesante observar que el mejor clasificador es K-NN con valores muy cercanos a 1.
- En las tablas 7.8, 7.9, 7.10 y 7.11; donde se pueden observar los resultados de Se y Sp del sistema basado en CBR, validado con 45 muestras de la base de casos de cardiotocografía, la mayoría de los clasificadores obtienen valores cercanos a 1 en las dos medidas.
- Se puede afirmar para este experimento que la tendencia general, es que el sistema se comporta mejor con un clasificador basado en K-NN. La anterior afirmación la podemos comprobar observando las tablas 7.7, 7.11.

7.3. Resultados: Aplicando AdaBoost y Random forest

Para cada una de las técnicas de clasificación, se utiliza un procedimiento de validación cruzada con 20 subconjuntos o folds, con el propósito de evitar el sesgo dado por el azar. Como medida de desempeño se usa el error medio de clasificación estándar. En las tablas

7.12, 7.13 y 7.14; se comparan los resultados del error y desviación estándar σ obtenidos al aplicar el algoritmo de Random forest sobre cada una de las bases de casos.

Tabla 7.12.: Medidas de desempeño Random forest sobre la base de casos de Cleveland

Error	σ
0.78	0.07

Tabla 7.13.: Medidas de desempeño Random forest sobre la base de casos de cardiotocografía

Error	σ
0.40	0.09

Tabla 7.14.: Medidas de desempeño Random forest sobre la base de casos de hipotiroidismo

Error	σ
0.31	0.15

Las tablas 7.15, 7.16 y 7.17 proporcionan una visión general del error promedio que se obtiene al aplicar el algoritmo de AdaBoost con los clasificadores SVM, ANN, PC y *K*-NN, sobre las diferentes bases de casos.

Tabla 7.15.: Error del algoritmo AdaBoost sobre la base de casos de Cleveland

Clasificador	Regla de combinación	Error	σ
SVM	Mean	0.78	0.05
	Vote	0.80	0.00
	Min	0.79	0.05
	Max	0.72	0.06
	Prod	0.71	0.08
	Wvote	0.79	0.04
	Median	0.76	0.06
ANN	Mean	0.68	0.08
	Vote	0.71	0.10
	Min	0.76	0.07
	Max	0.69	0.80
	Prod	0.75	0.09
	Wvote	Na	Na
	Median	0.70	0.07
PC	Mean	0.68	0.09
	Vote	0.68	0.09
	Min	0.80	0.07
	Max	0.68	0.07
	Prod	0.74	0.07
	Wvote	0.73	0.08
	Median	0.74	0.10
*K*NN	Mean	0.68	0.08
	Vote	0.71	0.07
	Min	0.72	0.08
	Max	0.68	0.09
	Prod	0.68	0.08
	Wvote	0.70	0.09
	Median	0.69	0.06

Tabla 7.16.: Error del algoritmo AdaBoost sobre la base de casos de cardiotocografía

Clasificador	Regla de combinación	Error	σ
SVM	Mean	0.36	0.13
	Vote	0.08	0.05
	Min	0.56	0.12
	Max	0.39	0.17
	Prod	0.48	0.13
	Wvote	0.38	0.11
	Median	0.24	0.06
ANN	Mean	0.06	0.04
	Vote	0.08	0.05
	Min	0.39	0.09
	Max	0.19	0.13
	Prod	0.37	0.14
	Wvote	Na	Na
	Median	0.10	0.05
PC	Mean	0.04	0.03
	Vote	0.23	0.11
	Min	0.58	0.10
	Max	0.04	0.02
	Prod	0.50	0.08
	Wvote	0.12	0.09
	Median	0.17	0.85
*K*NN	Mean	0.44	0.23
	Vote	0.42	0.29
	Min	0.44	0.22
	Max	0.34	0.16
	Prod	0.47	0.21
	Wvote	0.56	0.29
	Median	0.42	0.27

Tabla 7.17.: Error del algoritmo AdaBoost sobre la base de casos de hipotiroidismo

Clasificador	Regla de combinación	Error	σ
SVM	Mean	0.55	0.06
	Vote	0.54	0.07
	Min	0.60	0.07
	Max	0.53	0.08
	Prod	0.53	0.08
	Wvote	0.57	0.08
	Median	0.53	0.07
ANN	Mean	0.19	0.08
	Vote	0.19	0.09
	Min	0.48	0.15
	Max	0.30	0.03
	Prod	0.28	0.11
	Wvote	Na	Na
	Median	0.19	0.10
PC	Mean	0.18	0.04
	Vote	0.31	0.08
	Min	0.60	0.06
	Max	0.18	0.04
	Prod	0.54	0.10
	Wvote	0.56	0.15
	Median	0.34	0.07
*K*NN	Mean	0.31	0.21
	Vote	0.31	0.20
	Min	0.39	0.14
	Max	0.38	0.13
	Prod	0.31	0.22
	Wvote	0.32	0.22
	Median	0.33	0.21

Se puede concluir que:

- Si se compara el error obtenido al aplicar el algoritmo de Random forest sobre las diferentes bases de casos contra el algoritmo de AdaBoost, es mucho más alto el error en el primero.
- Los valores de error más bajos se obtienen aplicando las reglas de combinación promedio (mean) y votación (vote).
- Si se observa la tabla de error del algoritmo AdaBoost aplicado sobre la base de casos de cardiotocografía, el mejor resultado, es decir, el menor error esta dado por los clasificadores basados en ANN y PC.
- El resultado que emerge de aplicar AdaBoost sobre la base de casos de Cleveland es poco alentador. Los errores son muy altos, siendo el error más bajo registrado de 0.68.
- Basados en los datos de error al aplicar AdaBoost sobre la base de casos de hipotiroidismo, se observa que los mejores resultados se obtienen con los clasificadores basados en ANN y PC, al igual que ocurre con cardiotocografía.

7.4. Resultados: Metodología de CBR aplicando clasificadores supervisados

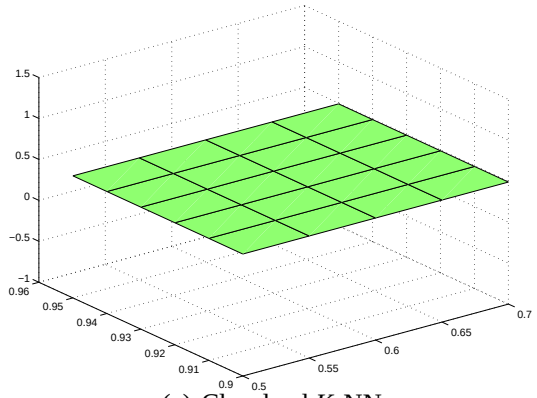
Límites de clase con mayor probabilidad para reentrenar el clasificador:
En el algoritmo de aprendizaje 7, diseñado para el presente experimento; el sistema aprende del nuevo caso, si y solo si, los límites de probabilidad de pertenencia de un caso correctamente clasificado, están entre 0.6 y 0.9. Dicho valor se elige a partir de los resultados que se muestran en la Figura 7.5, donde se puede ver que el límite inferior varía desde 0.5 hasta 0.7, y el límite superior va desde 0.9 hasta 0.96; el gráfico resultante es la evaluación de la exactitud límite a límite del sistema de CBR utilizando un clasificador *K*-NN sobre las bases de casos de Cleveland, cardiotocografía e hipotiroidismo.

Sensibilidad, especificidad y exactitud:
En las tablas 7.20, 7.19 y 7.18, se muestran los resultados de sensibilidad, especificidad y exactitud; de las pruebas realizadas con 3 clasificadores, las 4 bases de casos, y diferentes porcentajes de casos utilizados para el entrenamiento.

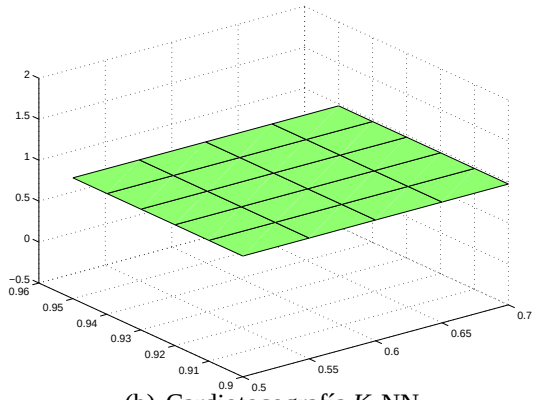
Base de casos	Clase	ANN			PC			*K*-NN		
		Acc	*Se*	*Sp*	*Acc*	*Se*	*Sp*	*Acc*	*Se*	*Sp*
Cleveland	$k=1$	0.752	0.884	0.596	0.592	0.955	0.160	0.699	0.902	0.457
	$k=2$	0.777	0.216	0.899	0.801	0.000	0.976	0.752	0.054	0.905
	$k=3$	0.830	0.208	0.912	0.883	0.000	1.000	0.854	0.250	0.934
	$k=4$	0.859	0.292	0.934	0.845	0.167	0.934	0.859	0.250	0.940
	$k=5$	0.937	0.000	0.980	0.956	0.000	1.000	0.951	0.000	0.995
Cardiotocografía	$k=1$	0.978	0.987	0.948	0.976	0.998	0.897	0.974	0.993	0.909
	$k=2$	0.978	0.917	0.988	0.974	0.845	0.995	0.974	0.859	0.992
	$k=3$	0.999	0.992	1.000	0.993	0.935	0.999	0.995	0.959	0.999
Cleveland ampliado	$k=1$	0.734	0.815	0.670	0.583	0.950	0.287	0.595	0.938	0.318
	$k=2$	0.703	0.246	0.830	0.740	0.254	0.874	0.734	0.270	0.863
	$k=3$	0.784	0.163	0.893	0.841	0.023	0.984	0.848	0.012	0.994
	$k=4$	0.795	0.185	0.894	0.864	0.049	0.996	0.843	0.049	0.972
	$k=5$	0.934	0.179	0.973	0.952	0.000	1.000	0.952	0.000	1.000
Hipotiroidismo	$k=1$	0.938	0.986	0.419	0.940	0.998	0.308	0.966	0.982	0.788
	$k=2$	0.939	0.127	0.987	0.946	0.075	0.998	0.964	0.694	0.980
	$k=3$	0.984	0.750	0.991	0.990	0.719	0.997	0.991	0.781	0.997

Tabla 7.18.: *Se*, *Sp* y *Acc* con 30 % de los casos para entrenamiento y el 70 % para pruebas

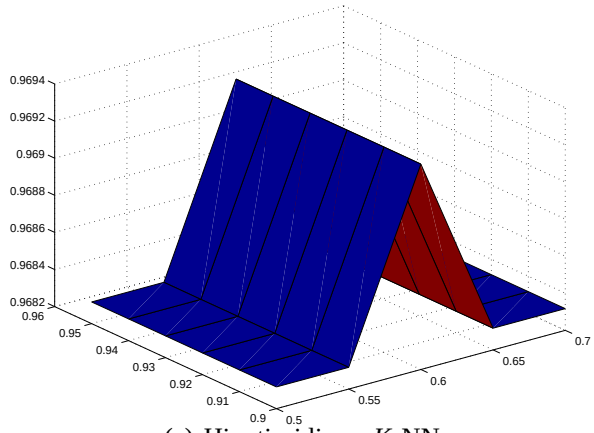
Curvas ROC:
A partir de los datos de sensibilidad y especificidad, se crean las curvas ROC por clase, para cada una de las bases de casos. Adicionalmente se halla el valor del AUC, como se puede observar en las Figuras 7.6, 7.7, 7.8, 7.9.



(a) Cleveland *K*-NN



(b) Cardiotocografía *K*-NN



(c) Hipotiroidismo *K*-NN

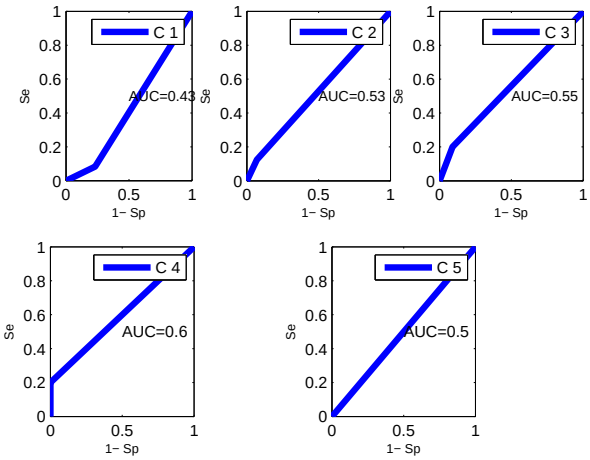
Figura 7.5.: Exactitud evaluada con diferentes límites de probabilidad

Base de casos	Clase	ANN			PC			*K*-NN		
		Acc	*Se*	*Sp*	*Acc*	*Se*	*Sp*	*Acc*	*Se*	*Sp*
Cleveland	$k = 1$	0.782	0.925	0.612	0.605	1.000	0.134	0.721	0.938	0.463
	$k = 2$	0.769	0.148	0.908	0.816	0.000	1.000	0.776	0.074	0.933
	$k = 3$	0.878	0.235	0.962	0.884	0.000	1.000	0.850	0.353	0.915
	$k = 4$	0.830	0.235	0.908	0.864	0.176	0.954	0.850	0.118	0.946
	$k = 5$	0.939	0.333	0.965	0.959	0.000	1.000	0.959	0.000	1.000
Cardiotocografía	$k = 1$	0.975	0.982	0.949	0.976	0.998	0.898	0.973	0.990	0.911
	$k = 2$	0.975	0.918	0.984	0.977	0.857	0.997	0.972	0.850	0.991
	$k = 3$	1.000	1.000	1.000	0.994	0.943	0.999	0.995	0.977	0.997
Cleveland ampliado	$k = 1$	0.734	0.859	0.633	0.577	0.946	0.279	0.580	0.930	0.297
	$k = 2$	0.696	0.244	0.821	0.720	0.244	0.852	0.737	0.233	0.877
	$k = 3$	0.836	0.164	0.952	0.853	0.000	1.000	0.843	0.016	0.986
	$k = 4$	0.790	0.207	0.885	0.855	0.017	0.992	0.855	0.103	0.978
	$k = 5$	0.925	0.000	0.972	0.952	0.000	1.000	0.952	0.000	1.000
Hipotiroidismo	$k = 1$	0.938	0.989	0.387	0.941	0.995	0.352	0.964	0.982	0.768
	$k = 2$	0.942	0.094	0.994	0.944	0.083	0.996	0.961	0.656	0.980
	$k = 3$	0.988	0.848	0.991	0.994	0.870	0.998	0.994	0.848	0.998

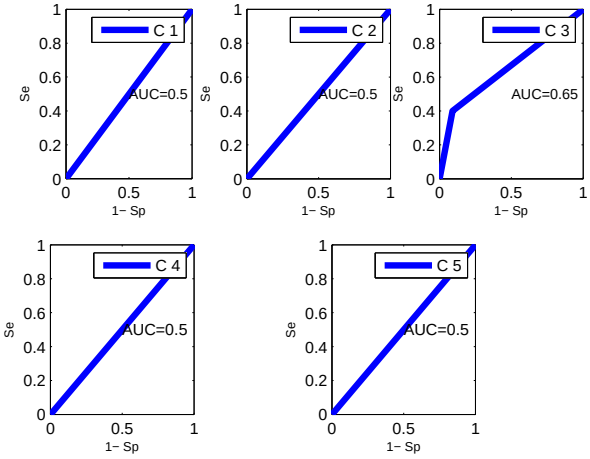
Tabla 7.19.: *Se*, *Sp* y *Acc* con 50 % de los casos para entrenamiento y el 50 % para pruebas

Base de casos	Clase	ANN			PC			*K*-NN		
		Acc	*Se*	*Sp*	*Acc*	*Se*	*Sp*	*Acc*	*Se*	*Sp*
Cleveland	$k = 1$	0.793	0.896	0.667	0.609	0.938	0.205	0.713	0.917	0.462
	$k = 2$	0.736	0.188	0.859	0.816	0.000	1.000	0.759	0.062	0.915
	$k = 3$	0.874	0.300	0.948	0.885	0.000	1.000	0.897	0.300	0.974
	$k = 4$	0.851	0.300	0.922	0.805	0.200	0.883	0.793	0.100	0.883
	$k = 5$	0.943	0.000	0.976	0.966	0.000	1.000	0.966	0.000	1.000
Cardiotocografía	$k = 1$	0.983	0.988	0.964	0.984	0.998	0.936	0.980	0.992	0.936
	$k = 2$	0.983	0.943	0.989	0.986	0.920	0.996	0.975	0.898	0.987
	$k = 3$	1.000	1.000	1.000	0.995	0.942	1.000	0.995	0.942	1.000
Cleveland ampliado	$k = 1$	0.721	0.811	0.647	0.595	0.955	0.301	0.591	0.919	0.324
	$k = 2$	0.721	0.407	0.808	0.757	0.352	0.870	0.761	0.370	0.870
	$k = 3$	0.773	0.083	0.891	0.854	0.000	1.000	0.850	0.000	0.995
	$k = 4$	0.822	0.147	0.930	0.858	0.000	0.995	0.850	0.059	0.977
	$k = 5$	0.943	0.083	0.987	0.947	0.000	0.996	0.951	0.000	1.000
Hipotiroidismo	$k = 1$	0.945	0.991	0.429	0.945	0.999	0.345	0.971	0.991	0.750
	$k = 2$	0.944	0.105	0.994	0.948	0.088	0.999	0.971	0.632	0.992
	$k = 3$	0.983	0.778	0.989	0.995	0.852	0.999	0.994	0.889	0.997

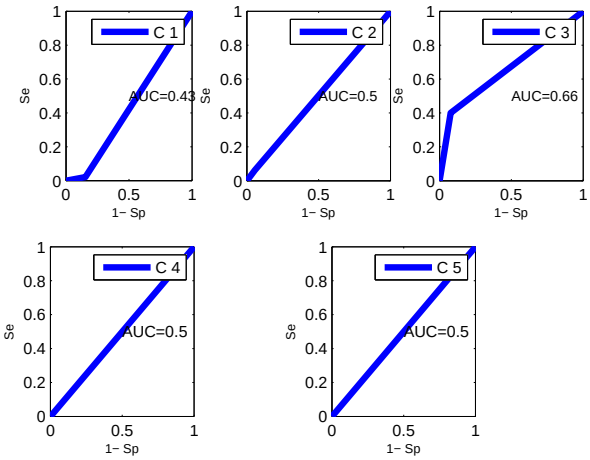
Tabla 7.20.: *Se*, *Sp* y *Acc* con 70 % de los casos para entrenamiento y el 30 % para pruebas



(a) ANN

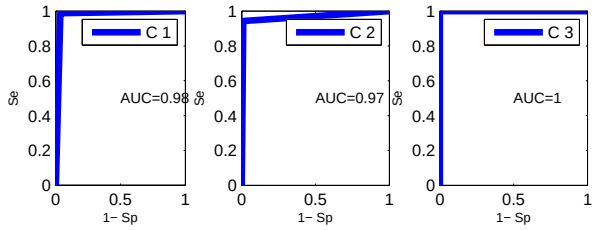


(b) PC

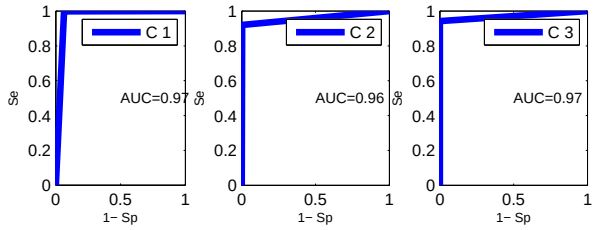


(c) *K*NN

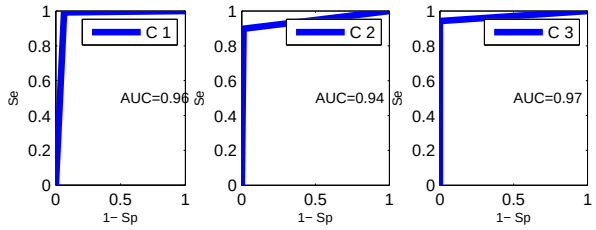
Figura 7.6.: Curvas ROC para la base de casos de Cleveland, aplicando los diferentes clasificadores en la etapa de adaptación



(a) ANN

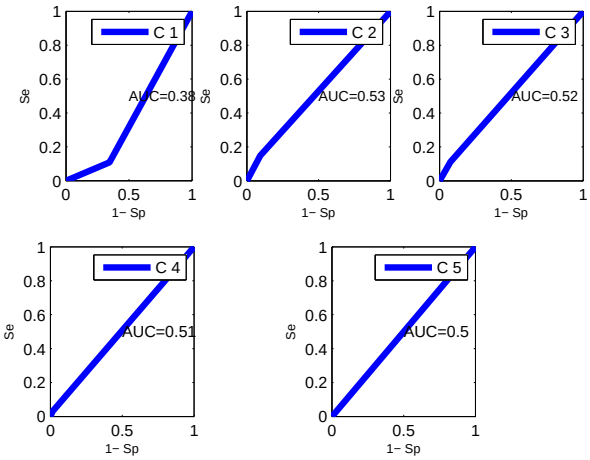


(b) PC

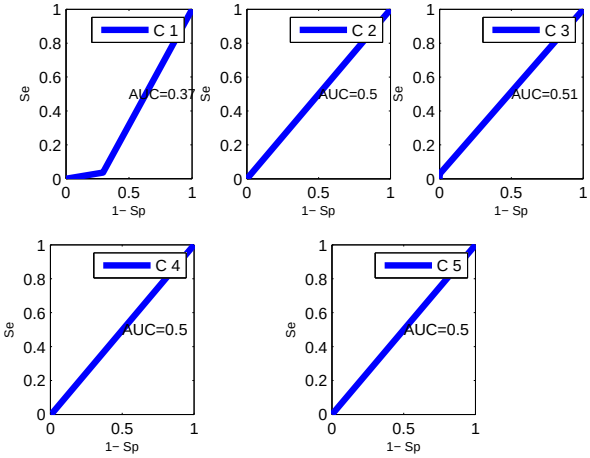


(c) *K*NN

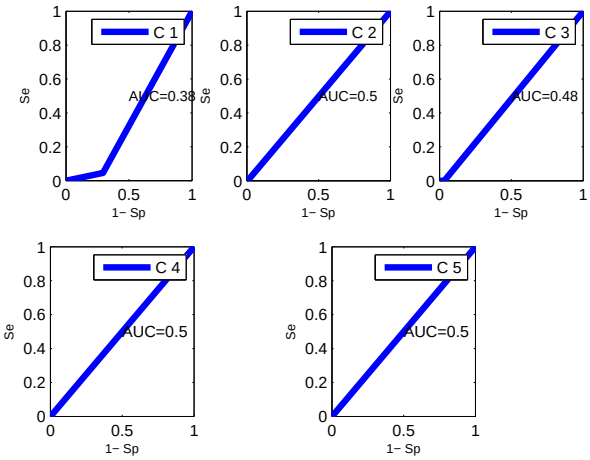
Figura 7.7.: Curvas ROC para la base de casos de cardiotocografía, aplicando los diferentes clasificadores en la etapa de adaptación



(a) ANN

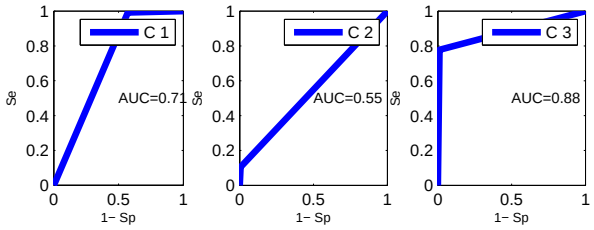


(b) PC

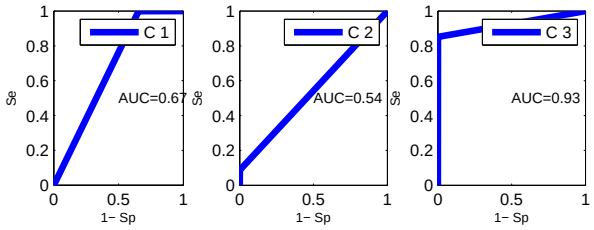


(c) *K*NN

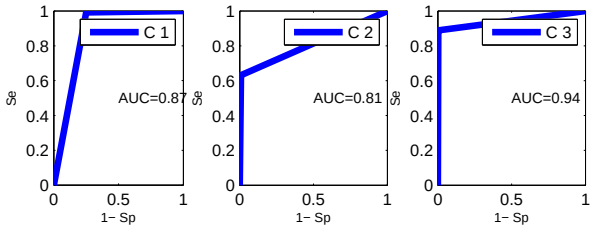
Figura 7.8.: Curvas ROC para la base de casos de Cleveland ampliado, aplicando los diferentes clasificadores en la etapa de adaptación



(a) ANN



(b) PC



(c) *K*NN

Figura 7.9.: Curvas ROC para la base de casos de hipotiroidismo, aplicando los diferentes clasificadores en la etapa de adaptación

Lo interesante de estos datos es que:

- Se ve un incremento en el valor de la exactitud, cuando aumenta el número de casos de entrenamiento; es decir, el mejor valor para la exactitud se puede apreciar en la Tabla 7.20, cuando se entrena con el 70 % de los casos.
- Con el clasificador basado en ANN, se obtienen los mejores resultados de Se en la mayoría de las pruebas realizadas.
- En la mayoría de las pruebas, los mejores resultados para la Sp, se pueden observar con el clasificador basado en PC.
- La exactitud es mucho más alta en los sistemas basados en CBR con el clasificador K-NN.
- Los valores del AUC en las base de casos de Cleveland y Cleveland ampliado, son muy bajos; solamente en la clase 3 de la base de datos de Cleveland, aplicando los clasificadores basados en PC y K-NN se logran valores por encima del 0.6, como se puede ver en las Figuras 7.6(b) y 7.6(c).
- El AUC obtenido para la base de casos de cardiotocografía, tiene un excelente valor en todas las pruebas realizadas, para todas las clases y todos los clasificadores, esto se puede observar en la Figura 7.7.
- Los mejores valores del AUC, para las clases de la base de casos de hipotiroidismo, se pueden observar con la aplicación del clasificador K-NN. Dichos valores están entre un buen test (0.8) y un muy buen test (0.94). Como se aprecia en la Figura 7.9(c)

7.5. Resultados: Metodología CBR aplicando clasificadores supervisados en cascada

Sensibilidad, especificidad y exactitud:
En las Tablas 7.21, 7.22 y 7.23 se pueden apreciar los resultados de sensibilidad, especificidad y exactitud de las pruebas realizadas con los clasificadores ANN, PC y *K*-NN como clasificadores biclase y en cada una de las Tablas se pueden observar los valores por clasificador multiclase, por base de casos y por clase.

Base de casos	Clase	ANN			PC			*K*-NN		
		Acc	*Se*	*Sp*	*Acc*	*Se*	*Sp*	*Acc*	*Se*	*Sp*
Cleveland	$k = 1$	0.782	0.938	0.590	0.701	0.958	0.385	0.839	0.958	0.692
	$k = 2$	0.793	0.375	0.887	0.805	0.188	0.944	0.747	0.313	0.845
	$k = 3$	0.885	0.000	1.000	0.885	0.000	1.000	0.885	0.000	1.000
	$k = 4$	0.862	0.500	0.909	0.839	0.300	0.909	0.851	0.500	0.896
	$k = 5$	0.966	0.000	1.000	0.966	0.000	1.000	0.966	0.000	1.000
Cardiotocografía	$k = 1$	0.981	0.986	0.964	0.981	0.996	0.929	0.986	0.996	0.950
	$k = 2$	0.981	0.943	0.987	0.983	0.898	0.996	0.987	0.932	0.996
	$k = 3$	1.000	1.000	1.000	0.998	0.981	1.000	0.998	0.981	1.000
Hipotiroidismo	$k = 1$	0.950	0.989	0.524	0.949	0.992	0.464	0.974	0.988	0.821
	$k = 2$	0.949	0.333	0.985	0.947	0.246	0.988	0.968	0.737	0.982
	$k = 3$	0.992	0.741	0.999	0.992	0.741	0.999	0.992	0.741	0.999
Cleveland ampliado	$k = 1$	0.717	0.892	0.574	0.664	0.847	0.515	0.794	0.901	0.706
	$k = 2$	0.632	0.463	0.679	0.632	0.426	0.689	0.700	0.778	0.679
	$k = 3$	0.854	0.028	0.995	0.846	0.028	0.986	0.842	0.000	0.986
	$k = 4$	0.866	0.029	1.000	0.862	0.000	1.000	0.862	0.000	1.000
	$k = 5$	0.951	0.000	1.000	0.951	0.000	1.000	0.951	0.000	1.000

Tabla 7.21.: *Se*, *Sp* y *Acc* clasificador multiclase: ANN

Curvas ROC:
A partir de los datos de *Se* y *Sp*, se crean las curvas ROC con su correspondiente AUC; estas figuras se generan por clase, para cada una de las bases de casos, y por combinación de clasificadores en cascada; dando como resultado 144 gráficos que se pueden apreciar en el apéndice D.

Base de casos	Clase	ANN			PC			*K*-NN		
		Acc	*Se*	*Sp*	*Acc*	*Se*	*Sp*	*Acc*	*Se*	*Sp*
Cleveland	$k = 1$	0.747	0.938	0.513	0.747	0.979	0.462	0.851	0.979	0.692
	$k = 2$	0.690	0.125	0.817	0.759	0.188	0.887	0.747	0.438	0.817
	$k = 3$	0.885	0.000	1.000	0.885	0.000	1.000	0.885	0.000	1.000
	$k = 4$	0.839	0.200	0.922	0.862	0.300	0.935	0.839	0.200	0.922
	$k = 5$	0.966	0.000	1.000	0.966	0.000	1.000	0.966	0.000	1.000
Cardiotocografía	$k = 1$	0.981	0.988	0.957	0.978	0.996	0.914	0.991	1.000	0.957
	$k = 2$	0.981	0.932	0.989	0.978	0.875	0.995	0.989	0.932	0.998
	$k = 3$	0.997	0.981	0.998	0.997	0.962	1.000	0.998	0.981	1.000
Hipotiroidismo	$k = 1$	0.942	0.991	0.393	0.948	0.997	0.405	0.984	0.994	0.881
	$k = 2$	0.942	0.175	0.987	0.946	0.193	0.991	0.976	0.842	0.984
	$k = 3$	0.990	0.667	0.999	0.990	0.630	1.000	0.990	0.630	1.000
Cleveland ampliado	$k = 1$	0.741	0.892	0.618	0.676	0.937	0.463	0.769	0.910	0.654
	$k = 2$	0.595	0.463	0.632	0.692	0.426	0.767	0.672	0.648	0.679
	$k = 3$	0.854	0.000	1.000	0.854	0.028	0.995	0.854	0.028	0.995
	$k = 4$	0.862	0.000	1.000	0.862	0.000	1.000	0.862	0.000	1.000
	$k = 5$	0.951	0.000	1.000	0.951	0.000	1.000	0.951	0.000	1.000

Tabla 7.22.: *Se*, *Sp* y *Acc* clasificador multiclase: PC

Base de casos	Clase	ANN			PC			*K*-NN		
		Acc	*Se*	*Sp*	*Acc*	*Se*	*Sp*	*Acc*	*Se*	*Sp*
Cleveland	$k = 1$	0.747	0.875	0.590	0.713	0.938	0.436	0.805	0.917	0.667
	$k = 2$	0.724	0.250	0.831	0.724	0.063	0.873	0.759	0.375	0.845
	$k = 3$	0.851	0.400	0.909	0.862	0.200	0.948	0.862	0.300	0.935
	$k = 4$	0.885	0.100	0.987	0.862	0.100	0.961	0.851	0.100	0.948
	$k = 5$	0.966	0.000	1.000	0.966	0.000	1.000	0.966	0.000	1.000
Cardiotocografía	$k = 1$	0.981	0.992	0.943	0.984	0.998	0.936	0.995	0.998	0.986
	$k = 2$	0.981	0.909	0.993	0.984	0.898	0.998	0.995	0.977	0.998
	$k = 3$	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
Hipotiroidismo	$k = 1$	0.934	0.983	0.393	0.946	0.997	0.381	0.979	0.992	0.833
	$k = 2$	0.943	0.175	0.988	0.947	0.105	0.997	0.977	0.737	0.992
	$k = 3$	0.987	0.778	0.993	0.993	0.852	0.997	0.994	0.889	0.997
Cleveland ampliado	$k = 1$	0.737	0.883	0.618	0.632	0.901	0.412	0.781	0.910	0.676
	$k = 2$	0.700	0.481	0.762	0.757	0.370	0.865	0.745	0.630	0.777
	$k = 3$	0.818	0.194	0.924	0.838	0.222	0.943	0.846	0.306	0.938
	$k = 4$	0.854	0.000	0.991	0.858	0.000	0.995	0.858	0.000	0.995
	$k = 5$	0.951	0.000	1.000	0.951	0.000	1.000	0.951	0.000	1.000

Tabla 7.23.: *Se*, *Sp* y *Acc* clasificador multiclase: *K*-NN

Algunos datos interesantes que se pueden observar son:

- Los mejores resultados de sensibilidad, especificidad y exactitud con la base de casos de Cleveland, se obtienen combinando dos clasificadores basados en *K*NN, como se puede observar en la Figura D.3(c).
- Con la base de casos de cardiotocografía se obtienen muy buenos resultados con todas las combinaciones, sin embargo, los valores cercanos a 1 se dan con las combinaciones del clasificador basado en *K*-NN como clasificador multiclase y como clasificador biclase cualquiera de los tres clasificadores, sea *K*-NN o PC o ANN; como se puede apreciar en la Figura D.6.
- Para la base de casos de hipotiroidismo los resultados mejoran considerablemente combinando *K*-NN como primer clasificador biclase y cualquier clasificador multiclase, como se aprecia en las Figuras D.10(c), D.11(c) y D.12(c).
- El AUC para la base de casos de Cleveland combinando dos clasificadores basados en *K*-NN, es muy buena para la clase 1 (0.9-0.97) y regular para las clases 2,3 y 4 (0.6-0.75).
- Combinando dos clasificadores basados en *K*-NN y aplicándolos sobre la base de casos de cardiotocografía, los valores del AUC son excelentes (0.97-1) para todas las clases.
- Si se observan los valores que se obtienen del AUC para la base de casos de Cleveland ampliado aplicando dos clasificadores basados en *K*-NN, se aprecia que mejoran con respecto a las demás combinaciones de clasificadores. Aunque los valores siguen siendo bajos, buen test para la clase 1, test regular para las clases 2 y 3, y un mal test para las clases 4 y 5.
- Para la base de casos de hipotiroidismo, clasificando con dos clasificadores basados en *K*-NN, el AUC tiene valores que se pueden clasificar entre bueno (0.75-0.9) y muy bueno (0.9-0.97).

7.6. Discusión de resultados

En el primer experimento se aprecia el comportamiento de los clasificadores multiclase sobre las bases de datos elegidas; donde se concluye que los clasificadores basados en SVM y *K*-NN son los que presentan mejor estabilidad. Sin embargo, a partir del experimento 7.4, se suprime el clasificador basado en SVM, debido al alto coste computacional que se presenta al querer evaluar uno a uno los casos de prueba de todas las bases de casos, con el sistema de CBR y este clasificador en la etapa de adaptación; impidiendo obtener resultados con dicho clasificador. La causa es el tiempo que conlleva el reentrenamiento del clasificador basado en SVM.
En el segundo experimento 7.2, los resultados para la Se y la Sp son excelentes cuando en la fase de adaptación del sistema de CBR el clasificador utilizado es el basado en *K*-NN; sin embargo, el balanceo de datos cuenta con una serie de desventajas:

- Impone costos de clasificación errónea no uniforme, es decir, si se altera la distribución de clases del conjunto de entrenamiento de manera que la proporción de ejemplos positivos a negativos cambia de 1:1 a 2:1, entonces se asigna una proporción de coste de clasificación errónea de 2:1. Esto fue formalmente descrito por Elkan en [78].

- Con el sobre muestreo es muy común que se genere una regla de clasificación para cubrir un solo ejemplo [79].

- Otra desventaja es que aumenta el número de casos y de esta forma el tiempo de aprendizaje [79]. Si se considera lo planteado por Montani en el trabajo [7], donde se concluye que es necesario reducir el espacio de búsqueda en la fase de recuperación de casos de los sistemas basados en CBR; el balanceo de datos aumentaría dicho espacio.

Se puede apreciar que al segmentar la clasificación, dividiendo los casos entre enfermos y sanos, mejoran los resultados de las medidas de desempeño; en comparación con la aplicación de clasificadores sobre las bases de casos sin segmentar, esto se debe a que existe una primera clasificación para determinar si el caso pertenece a la clase enfermo o a la clase sano, para luego pasar a un segundo clasificador especializado solamente en casos enfermos. Por otra parte, el clasificador basado en *K*-NN, es el que finalmente triunfa sobre los demás, es menos sensible a las clases minoritarias y a la baja separación existente entre las clases en bases de casos como la de arritmias de Cleveland; porque al no tener un modelo predefinido y recuperar casos similares utilizando distancia Euclídea, tiene mayor probabilidad de clasificar correctamente dichas las clases.
Bajo la premisa de que el clasificador basado en SVM trabaja muy bien con conjuntos de entrenamiento pequeños, enfermedades poco comunes en el campo del diagnóstico médico; da una idea intuitiva de que la aplicación de un clasificador de esta naturaleza en las etapas de recuperación y adaptación de un sistema de CBR, puede mejorar los resultados. En el primer

experimento 7.1, se evalúan los clasificadores con dos de las bases de datos seleccionadas y sus resultados indican que la premisa puede ser cierta, debido a que junto con el clasificador basado en *K*-NN son los más estables, además de ser el clasificador que presenta los errores más bajos. Posteriormente se aplican los clasificadores en la etapa de recuperación y adaptación de un sistema de CBR 7.2, obteniendo con clasificador basado en SVM niveles de sensibilidad y especificidad muy bajos para la base de casos de Cleveland, contrario a lo que ocurre al aplicar el clasificador basado en *K*-NN, donde los resultados son excelentes para las dos bases de casos. Y finalmente, en el experimento 7.4 se elimina el clasificador basado en SVM de las pruebas debido al alto coste computacional. Como conclusión, se puede decir que el clasificador basado en SVM al tener un alto coste computacional limita su utilización en sistemas basados en CBR donde el aprendizaje es continuo y es necesario reentrenar el clasificador cada vez que el sistema comete un error.

Cuando se analiza el clasificador basado en ventanas de Parzen, se encuentra que es un clasificador muy robusto, pero necesita muestras suficientes para obtener buenos resultados de clasificación. Al aplicar el clasificador basado en Parzen en las etapas de recuperación y adaptación del sistema basado en CBR, se obtienen los mejores valores de especificidad en los experimentos 7.2 y 7.4. Estos resultados son debido a que siempre existen más casos negativos o no pertenecientes a la clase de interés y se puede concluir que no es un clasificador apto para ser utilizado en sistemas basados en CBR que tengan pocos casos de alguna clase en particular.

El clasificador basado en ANN, ampliamente utilizado en medicina para resultados dicotómicos, presenta buenos resultados de clasificación en los experimentos 7.1 y 7.3, pero cuando se aplica en las etapas de recuperación y adaptación del sistema basado en CBR los resultados no son buenos. Se puede concluir que no es un buen algoritmo para combinar con un sistema basado en CBR.

8. CONCLUSIONES Y TRABAJO FUTURO

Este capítulo presenta las conclusiones obtenidas durante el desarrollo del presente trabajo y se proponen posibles líneas de trabajo futuro. Además se resaltan expresamente los aportes realizados con este trabajo de investigación.

8.1. Conclusiones

Considerando lo expuesto en este estudio, puede concluirse lo siguiente:

- La representación de casos es un aspecto de alto interés en el área del CBR. En este trabajo se comprueba que la selección de características utilizando CFS junto con el método de selección best first es una buena alternativa, mejorando no solo el coste computacional al reducir la dimensión de características, sino también mejorando el proceso de clasificación.

- Tradicionalmente los sistemas de CBR para diagnóstico médico han sido diseñados para la representación de dos estados: Normal o patológico. En el presente trabajo se explora la extensión de sistemas de CBR a través del uso de clasificadores multiclase en las etapas de recuperación y adaptación. Particularmente el clasificador basado en K-NN, dadas sus características geométricas y específicamente aplicando este clasificador en cascada, primero a un conjunto de casos etiquetados como enfermos o sanos y luego sobre un conjunto de casos etiquetados con los diagnósticos diferenciales, demuestra ser un clasificador muy potente para ser aplicado en sistemas basados en CBR de diagnóstico médico.

- La estrategia de recuperación-adaptación propuesta en esta tesis comprueba ser una alternativa adecuada para presentar información al especialista. En este sentido las probabilidades estimadas a partir de distancias o de la exploración de las vecindades geométricas de los puntos (representación de los casos), permite hacer una recuperación-adaptación automática, y presentar la información de una forma inteligible para el especialista. En general se comprueba que los resultados son admisibles, no obstante

cuando hay datos ruidosos, clases minoritarias o clases traslapadas, los resultados se ven afectados.

- A partir de la evidencia experimental y el marco teórico de la metodología desarrollada, se verifica que un sistema adecuado y completo de CBR incluyendo una recuperación-adaptación mejorada (SAM) puede lograrse mediante etapas de: pre-proceso, filtrar datos; recuperación, utilizando clasificadores en cascada basados en *K*-NN; adaptación, calculando probabilidades de pertenencia a cada una de las clases. Se puede resaltar que es robusto para la clasificación de clases minoritarias y para bases de casos donde las clases están altamente traslapadas. De esta forma se soporta la hipótesis planteada al inicio de la presente investigación: *"La representación de los casos mediante selección de variables y/o reducción de dimensiones, así como la recuperación eficiente a partir de la integración del CBR convencional y clasificadores multiclase permitirán obtener un sistema robusto y eficiente para enfrentar los desafíos que las ciencias de la salud ofrecen a la comunidad científica. Especialmente, el sistema propuesto permitirá al usuario obtener diagnósticos de múltiples clases y más cercanos a la realidad de acuerdo con el análisis del histórico de pacientes"*. En el presente trabajo se desarrolla un sistema basado en CBR capaz de estimar probabilidades de pertenencia en problemas multiclase, a partir de bases de casos de diagnóstico médico. SAM se convierte en un punto de partida válido para la creación de sistemas basados en CBR aplicados al diagnóstico de patologías, donde es común trabajar con varias etiquetas o clases.

- Entre las aplicaciones que puede llegar a tener el presente trabajo, vale la pena destacar que puede utilizarse como modelo de referencia en la enseñanza y entrenamiento de especialistas médicos, enriqueciendo el proceso de aprendizaje.

- En la actualidad, la mayoría de los hospitales o centros especializados en atención de pacientes, guardan todos los datos de forma digital; pero todavía existen muchas barreras para poder acceder a dicha información y poder realizar estudios como este. La mayor dificultad encontrada para desarrollar la presente investigación, fue la de encontrar bases de datos de diagnóstico médico; los directivos de las instituciones no comparten la información por miedo a vulnerar la privacidad del paciente. Pero deberían tener presente que los investigadores no necesitan las identidades de los pacientes, en cambio el beneficio que podrían obtener a través de estas investigaciones es incalculable.

8.2. Trabajo futuro

En la realización del presente trabajo se presentan algunos inconvenientes tales como:

- Se utilizan bases de datos que ya han sido ampliamente utilizadas en el área de ML. Explorar diferentes bases de datos de diagnóstico médico que no hayan sido trabajadas con ML es un punto abierto.
- Otro punto que requiere un trabajo más profundo es el referente a trabajar con clases donde existen pocas muestras. Por ello se propone una extensión que considere en la etapa de pre-proceso la implementación de un algoritmo de balanceo de datos, que no sugiera un problema adicional debido al coste computacional que puede tenerse al crecer la base de casos.
- Queda abierta la posibilidad de desarrollar un sistema de CBR basado en la metodología propuesta en el presente trabajo, sobre plataformas de software libre.

Parte V.

APÉNDICES

A. Tabla resumen revisión de tema CBR aplicados al sector salud

Tabla A.1.: Muestra las técnicas utilizadas en el CBR para cada fase: Recuperación, adaptación, revisión y aprendizaje y el porcentaje de éxito del sistema.

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[31]	BPN BPN-Euclidian distance				ten-fold cross-validation	93 95
[80]	Nearest neighbourhood (*K*-NN) Dempster–Shafer	Annealing weight			leave one out strategy (LOOCV)	81.94 84.72
[32]	Bayes probability. *K*-NN	Rule based expert systems		Repair case	Bernoulli distribution	90 93 97
[33]	Euclidean distance	Bayesian probability of drugs	Manual	stored case automatically	Usability test	58 86
[22]	Modified distance Similarity matrix. *K*-NN – Fuzzy Similarity	Top most retrieved case. Average retrieved cases. Weighted average.		Manual	Goodness-of-fit (R2) Absolute mean difference	R2 64 46 86 Abs mean 57 71 21

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[14]	EWCBR. Cbr equal weighted UWCBR Unequal wighted CBR NWCBR nonlinear wighted ANN C5.0 SOFTWARE CART				3-fold cross validation	77.7 91.9 92.8
[81]	Decision Tree (DT) Logistic regression (LR) ANN Adaptative neuro-fuzzy inference system (ANFIS)				Area under the curve	
[82]	Fuzzy Mathematics and Euclidian-Lagrangian Distance	Adaptation rules base	Manual	Manual	Statistical frecuency	99.7
[83]	*K*-NN	Manual				
[84]	Knowledge guided sematic indexing	Frames processing and analysis			local grading	80

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[40]	RMA Robust multiarray average fuzzy discretization and pattern selection	Hierarchical cluster called dendogram, classification tree	Manual	Manual		
[15]	Typycal CBR *K*-NN (TYCBR) Feature selection using GA (FSCBR) Feature Weighting using GA (FWCBR) Instance selection using GA (ISCBR) FISCBR Feature and Instance Selection use GA Feature and Instance Selection (FWISCBR) Global optimization of feature weights (GOCBR)				Fold cross 5	95.58 96.46 97.35 97.35 97.35 98.23 99.12

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[34]	CBR *K*-NN	NeuroSolutions 5.0 (Multilayer Perceptron Network) Rule base subsystem Inference Engine			Cross validation	
[28]	Cluster analysis for segmentation	Fuzzy rule base (type I) Fuzzy rule base (type II)			Frecuency	73.68 83.16
[85]	ID3 J48 NB Naive Bayes Tree Augmented Naive Bayes (TAN) BN Hill Climber: Build Bayesian network. BN (repeat Hill-Climber)				holdout	93.49 95,37 94.07 94.79 94.65 94.79

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[27]	J48 RIDOR *K*-NN Recovering three closest cases	NA Adaptation mutually exclusive. The first list Optimistic adaptation rules Pessimistic adaptation rules Degree of dependency			Leave-one-out	73 71 75 75 71 68 76
[38]	SVM *K*-NN Naïve Bayes FDT. Fuzzy Decision Tree CBFDT	A stepwise regression (SRA) fuzzy rules				liver dis. 77.6 73.7 70.2 68.3 90.4 Breast 98.1 96.9 91.4 90.2 98.9

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[35]	BPN 39-9-1 learning rate 0,1 momentum 0,7 *K*-NN - Analytic hierarchy process (AHP). 10 most similar cases.				5-fold cross-validation	98.04 94.57
[86]	*K*-NN with temporal abstraction 1 season in case base. 2 Season in case base. 3 Season in case base. 4 Season in case base.					95
[36]	*K*-NN with benign biopsies avoided					

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[87]	Global comparison of polar map data (GLOB). Case retrieval based on a territorial comparison of polar map data (TER) Case retrieval based on a comparison of a given case with eight sub-libraries classified according to the involvement of the three major coronary vessels using a group similarity measure (GROUP)	Best Match(BM) Probability adaptation				BM 70 75 77 B 75 77 74

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[88]	*K*-NN	The adapted solution Q is obteined by adding the classified solutions of the retrieved cases, multiplied by the inverse of its global distance measurement				Valve model 85 Valve model 77
[89]	*K*-NN	Rule-based reasoning (RBR)				85
[43]	*K*-NN	Probability calculation and GA				73.87
[90]	*K*-NN SVM CBR approach based on artificial immune system algorithm (AISCBR)				5-fold-cross-validation	Breast-cancer 96.64 98 Echocar 98.67

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[91]	Breast Cancer Wisconsin data set Fuzzy min-max(FMM) FMM-CART(Classification and regression tree) FMM-CART-RF (Random Forest)				10-fold cross-validation	95.26 95.71 98.84
[92]	Fuzzy rule base Numeric data: machine learning, pattern recognition. Symbolic data: Logic based methods, decision trees, fuzzy DTs, Bayesian Nets Numeric & symbolic: Neuro-fuzzy systems, decision trees cognition, association rules					

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[93]	Heart STATLOG C5.0 CART CSBR Retrieves neighbors based on the probabilistic distribution	GA			10-fold cross validation	
[94]	Use eXiTCDSS					83.23
[95]	Mega - trend diffusion and suport vector machine (MTD-SVM) cancer/non-cancer breast/non-breast cancer colon/non-colon cancer				10-fold cross validation	96.71 95.18 96.50

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[41]	*K*-NN	multilabel dermoscopy Multilabel confocal Multilabel collaborative Multilabel collaborati-ve+rules Multilabel collaborati-ve+rules +DML			Leave one out	92 96 95 98 100
[96]	*K*-NN	Adaptation and reaso-ning rules			5-fold va-lidation	Thyroid 93.49 99.53 Mammo 78.62 99.33

Tabla A.1 –

Referencia	Recuperación	Adaptación	Revisión	Aprendizaje	Análisis	Acc
[97]	*K*-NN flexible auto-set tolerance	Modify the retrieved cases or just add a new case to the proposed manually				
[98]	*K*-NN				statistical fre-quency	

B. Estudio Comparativo De Métodos De Selección Y Clasificación Supervisados.

B.0.1. Materiales y métodos

Tomando como referencia las bases de datos de cardiotocografía y la de arritmias de Cleveland, utilizando el software de minería de datos Weka y un ordenador con un procesador Intel Core i5 de 2.3 GHz, con memoria RAM de 8GB; se realizan diferentes pruebas para encontrar la mejor combinación de un método de selección de atributos junto con un algoritmo de clasificación supervisado.
Los métodos de selección que se utilizan son el Best first y el Ranker. Best first, es un algoritmo heurístico que hace una búsqueda del nodo más prometedor formando una especie de árbol, los nodos (atributos) que tienen mayor influencia en la representación de la determinación de las clases se mostraran al final del algoritmo, viene acompañado de un atributo evaluador, en este caso CFS, el cual evalúa un subconjunto de atributos considerando la habilidad predictiva individual de cada variable, así como el grado de redundancia entre ellas. Y el algoritmo Ranker devuelve una lista ordenada de los atributos según su calidad, el atributo evaluador utilizado es Gain Attribute que evalúa los atributos midiendo la ganancia de información de cada uno con respecto a la clase [99].
Los métodos de clasificación seleccionados para realizar las pruebas, se determinaron por ser muy utilizados en los procesos de ML, estos son:

- Naive Bayes
- Multilayer Perceptrón
- Vecinos cercanos *K*-NN
- AdaBoost
- Iterative Classifier Optimizer
- Random Forest

B.0.2. Marco experimental

Para la evaluación de los métodos de selección de características se emplean dos conjuntos de datos. El primer conjunto de datos contiene 3 clases, 2126 observaciones y 32 características, que corresponden al registro de lecturas de frecuencia cardíaca fetal, los movimientos fetales y contracciones uterinas llamadas registros de cardiotocografía. El segundo conjunto contiene 4 clases, 297 observaciones y 13 características.
Con la base de datos de cardiotocografía, se realiza la primera prueba tomando los 32 atributos, sin utilizar ningún método de selección y aplicando los 6 métodos de clasificación. Los resultados de esta prueba sirven como referencia para verificar que tan efectivos son los emparejamientos cuando se utilicen los diferentes métodos de selección. Se aplica CFS junto con el método de selección Best first, luego se aplican los algoritmos de clasificación, los cuales se evalúan teniendo en cuenta si la clasificación es correcta o no, el tiempo de ejecución, el error absoluto relativo (RAE por su nombre en inglés - *Relative Absolute Error*) y el error absoluto medio (MAE por su nombre en inglés - *Mean Absolute Error*).

B.0.3. Resultados y discusión

Los resultados de aplicar el proceso experimental anterior en la base de datos de cardiotocografía se pueden observar en las tablas B.1, B.2 y B.3:

Método	Bien (%)	Mal (%)	Tiempo	RAE	MAE
1. Naive Bayes	86.92	13.07	0.09	35.70	0.08
2. Multilayer Perceptrón	99.01	0.98	11.44	3.25	0.01
3. *K*-NN	99.01	0.98	0.01	4.06	0.0078
4. AdaBoost	87.11	12.88	0.24	56.04	0.13
5. Iterative Classifier Optimizer	98.77	1.22	2.51	8.10	0.019
6. Random Forest	98.91	1.08	0.75	7.76	0.019

Tabla B.1.: Métodos de clasificación vs Ningún método de selección para la base de datos de Cardiotocografía.

Al comparar los resultados de la tabla B.1 con la tabla B.2, se puede observar como disminuyen los tiempos, debido a la menor cantidad de atributos que pasan de 32 a solo 7 con el método ”Best first”.

Método	Bien (%)	Mal (%)	Tiempo	RAE	MAE
1. Naive Bayes	92.85	7.14	0	29.20	0.07
2. Multilayer Perceptrón	98.58	1.41	1.82	6.27	0.015
3. *K*-NN	98.58	1.41	0	6.03	0.014
4. AdaBoost	87.11	12.88	0.03	56.04	0.13
5. Iterative Classifier Optimizer	98.49	1.50	0.61	10.22	0.025
6. Random Forest	98.77	1.22	0.23	6.56	0.016

Tabla B.2.: Métodos de clasificación vs método de selección Best first para la base de datos de Cardiotocografía.

Si se comparan los resultados del seleccionador Ranker, Tabla B.3 con los resultados de la Tabla B.1, se observa una disminución de tiempo de ejecución. Si se comparan los resultados anteriores con la tabla B.2, los tiempos aplicando Ranker son 0.05s en promedio más altos que los resultantes al utilizar el algoritmo Best first.
Se puede apreciar que los clasificadores que tienen un mejor desempeño son el basado en *K*-NN y el basado en multilayer perceptrón, utilizando como seleccionador Best first, con un porcentaje de aciertos del 0.99. Con respecto al error absoluto el clasificador con el mayor y menor valor en las tres tablas es el de AdaBoost y *K*-NN respectivamente.
Teniendo en cuenta el análisis descrito anteriormente se puede decir que el método que presenta los mejores valores de clasificación, tiempo y error es el método de selección Best first. Y el clasificador basado en *K*-NN es el que mejores resultados presenta.
Un proceso similar se realiza con la base de datos de Cleveland, y se obtienen los siguientes resultados:
Se puede apreciar en las tablas B.5 y B.6, menor tiempo de proceso, comparado con los resultados observados en la tabla B.4. El método de selección Best first es el que ofrece mejores resultados, unido a los clasificadores 3, 5 y 6. Y con el método de selección Ranker se puede apreciar que al igual que con el seleccionador Best first los mejores resultados están dados por los clasificadores 3, 5 y 6.
AdaBoost es el clasificador con el error absoluto medio más alto en todas las pruebas. Otro aspecto interesante a tener en cuenta es el número de datos correctamente clasificados, el que mejor resultado presenta es la combinación del seleccionador Best first junto con el clasificador iterative classifier optimizer con un 67 % de datos correctamente clasificados.

Método	Bien (%)	Mal (%)	Tiempo	RAE	MAE
1. Naive Bayes	91.58	8.41	0	25.05	0.061
2. Multilayer Perceptrón	98.25	1.74	1.98	7.36	0.018
3. *K*-NN	97.9	2.06	0	5.92	0.014
4. AdaBoost	87.11	12.88	0.03	56.04	0.137
5. Iterative Classifier Optimizer	98.21	1.78	0.69	11.63	0.028
6. Random Forest	98.68	1.31	0.27	7.77	0.019

Tabla B.3.: Métodos de clasificación vs método de selección Ranker para la base de datos de Cardiotocografía.

Método	Bien (%)	Mal (%)	Tiempo	RAE	MAE
1. NaiveBayes	55.55	44.44	0	70.97	0.18
2. Multilayer Perceptrón	54.88	45.11	0.59	70.88	0.18
3. *K*-NN	55.21	44.78	0	75.33	0.19
4. AdaBoost	51.51	48.48	0	113.57	0.29
5. Iterative Classifier Optimizer	60.26	39.73	0.23	74.73	0.19
6. Random Forest	57.57	42.42	0.09	77.92	0.20

Tabla B.4.: Métodos de clasificación vs Ningún método de selección para la base de datos de Cleveland.

Método	Bien (%)	Mal (%)	Tiempo	RAE	MAE
1. Naive Bayes	57.91	42.08	0.01	70.34	0.18
2. Multilayer Perceptrón	51.85	48.14	0.36	74.83	0.19
3. *K*-NN	58.58	41.41	0	72.50	0.18
4. AdaBoost	53.87	46.12	0	120.84	0.31
5. Iterative Classifier Optimizer	67.003	33.00	0	70.21	0.18
6. Random Forest	54.56	45.43	0.36	78.38	0.22

Tabla B.5.: Métodos de clasificación vs método de selección Best first para la base de datos de Cleveland.

Método	Bien (%)	Mal (%)	Tiempo	RAE	MAE
1. Naive Bayes	57.91	42.08	0.01	70.34	0.18
2. Multilayer Perceptrón	51.85	48.14	0.54	74.83	0.19
3. *K*-NN	58.58	41.41	0	72.50	0.18
4. AdaBoost	51.51	48.48	0.02	113.57	0.29
5. Iterative Classifier Optimizer	58.92	41.07	0.53	75.55	0.19
6. Random Forest	58.92	41.07	0.09	74.99	0.19

Tabla B.6.: Métodos de clasificación vs método de selección Ranker para la base de datos de Cleveland.

B.0.4. Conclusiones

- En el diseño de sistemas de clasificación y reconocimiento de patrones, los métodos de selección presentan diversos factores a tener en cuenta tales como: Costo computacional, conocimiento del conjunto de datos y objetivo de la clasificación. En este trabajo se presentan los resultados de la combinación de diferentes seleccionadores y clasificadores, con esto se resalta la importancia de aplicar algoritmos de reducción de dimensiones con el fin de mejorar los resultados en procesos de aprendizaje de máquina.
- El proceso de selección, es necesario en este tipo de bases de datos. Al determinar las características más importantes de un conjunto de datos, hace que la base de datos sea más pequeña y por ende más rápida de procesar al aplicar un clasificador.

C. Atributos de las bases de datos

C.1. Información de los atributos de la base de datos de Cardiotocografía

1. LB - Línea de base de la FHR (latidos por minuto)
2. AC - Número de aceleraciones por segundo
3. FM - Número de movimientos fetales por segundo
4. UC - Número de contracciones uterinas por segundo
5. DL - Número de deceleraciones de luz por segundo
6. DS - Número de deceleraciones severas por segundo
7. DP - Número de deceleraciones prolongadas por segundo
8. ASTV - Porcentaje de tiempo con variabilidad anormal a corto plazo
9. MSTV - Valor medio de la variabilidad a corto plazo
10. ALTV - Porcentaje de tiempo con variabilidad anormal a largo plazo
11. MLTV - Valor medio de la variabilidad a largo plazo
12. Width - Anchura del histograma de FHR
13. Min - Mínimo del histograma de FHR
14. Max - Máximo del histograma de FHR
15. Nmax - Número de picos del histograma
16. Nzeros - Número de ceros del histograma
17. Mode - Moda del histograma
18. Mean - Media del histograma

19. Median - Mediana del histograma
20. Variance - Varianza del histograma
21. Tendency - Tendencia del histograma. 1 = izquierda asimétrica; 0 = simétrico; 1 = antisimétrico a la derecha
22. A - Dormido calmado
23. B - Fase REM
24. C - Calmado en estado de vigilancia
25. D - Activo
26. SH - Cambio de patrón (A or Susp with shifts)
27. AD - Situación de estrés
28. DE - Estimulación vagal
29. LD - Gran patrón desacelerativo
30. FS - Patrón sinusoidal plano
31. SUSP - Patrón sospechoso
32. CLASS - Código patrón de clase FHR (1 a 10)
33. NSP - Código de clase del estado fetal (N = normal, S = sospechoso, P = patológico)

C.2. Información de los atributos de la base de datos de hipotiroidismo

1. Edad: continua
2. Sexo: M, F
3. Tiroxina: falso (f), verdadero (t)
4. Consulta sobre tiroxina: f, t
5. Sobre la medicación antitiroidea: f, t
6. Enfermo: f, t

7. Embarazada: f, t
8. Cirugía de tiroides: f, t
9. I131 tratamiento: f, t
10. Hipotiroidismo?: f, t
11. Hipertiroidismo?: f, t
12. Litio: f, t
13. Bocio: f, t
14. Tumor: f, t
15. Hipopituario: f, t
16. Psych: f, t
17. ¿Conoce el valor de la hormona estimulante del tiroides (TSH)?: f, t
18. Valor de la TSH: continua
19. ¿Tiene la medida de la T3?: f, t
20. T3: continua
21. ¿Tiene la medida de la TT4?: f, t
22. TT4: continua
23. ¿Tiene la medida de la T4U?: f, t
24. T4U: continua
25. ¿Tiene la medida de la FTI?: f, t
26. FTI: continua
27. ¿Tiene la medida de la globulina fijadora de la tiroxina (TBG)?: f, t
28. TBG: continua
29. Hipotiroidismo, hipotiroidismo primario, hipotiroidismo compensado, Hipotiroidismo secundario, negativo — Clases

C.3. Información de los atributos de la base de datos de enfermedades cardíacas

1. id: Número de identificación del paciente
2. ccf: Número de seguro social (se reemplazó por un valor ficticio de 0)
3. age: Edad en años
4. sex: Sexo (1 = varón, 0 = mujer)
5. painloc: Localización del dolor torácico (1 = subesternal, 0 = de lo contrario)
6. painexer: Dolor por el ejercicio (1 = provocado por el esfuerzo, 0 = en caso contrario)
7. relrest: Reposo (1 = aliviado después del reposo, 0 = en caso contrario)
8. pncaden: suma de 5, 6 y 7
9. cp: Tipo dolor de pecho
   Valor 1: Angina típica
   Valor 2: Angina atípica
   Valor 3: Dolor no anginal
   Valor 4: Asintomático
10. trestbps: Presión arterial en reposo (en mm Hg al ingreso al hospital)
11. htn
12. col: Colesterol sérico en mg / dl
13. smoke: 1 = sí; 0 = no (es o no es fumador)
14. cigs: Cigarrillos por día
15. years: Número de años como fumador
16. fbs: ¿Azúcar en sangre en ayunas¿120 mg / dl? 1 = verdadero, 0 = falso
17. dm: 1 = Historia de diabetes, 0 = ninguna historia
18. famhist: ¿Antecedentes familiares de enfermedad coronaria?1 = sí; 0 = no)

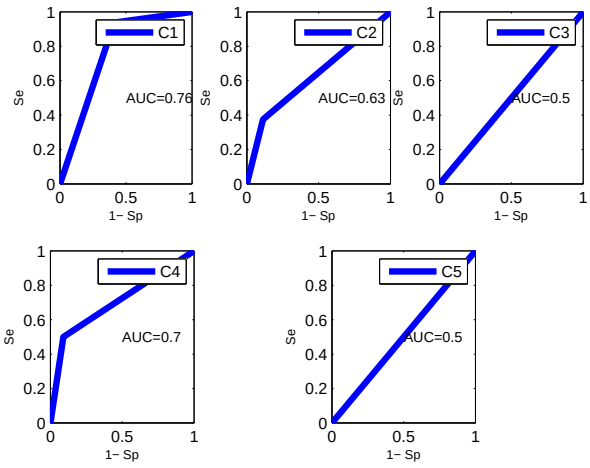
19. restecg: Resultados electrocardiográficos en reposo
    Valor 0: Normal
    Valor 1: Con anomalía de onda ST-T (inversión de onda T y / o elevación o depresión de ST¿0.05 mV)
    Valor 2: Muestra hipertrofia ventricular izquierda probable o definitiva según el criterio de Estes
20. ekgmo: Mes de lectura del ECG
21. ekgday: Día de lectura de ECG
22. ekgyr: Año de lectura del ECG
23. dig: Digitales utilizadas durante el ECG. 1 = si; 0 = no
24. prop: Bloqueador beta utilizado durante el ECG. 1 = si; 0 = no
25. nitr: Nitratos utilizados durante el ECG: 1 = si; 0 = no
26. pro: Bloqueador de canales de calcio utilizado durante el ECG. 1 = si; 0 = no
27. diuretic: Diurético utilizado durante el ejercicio ECG. 1 = si; 0 = no
28. proto: Protocolo
    - 1 = Bruce
    - 2 = Kottus
    - 3 = McHenry
    - 4 = Fast Balke
    - 5 = Balke
    - 6 = Noughton
    - 7 = bike 150 kpa min/min
    - 8 = bike 125 kpa min/min
    - 9 = bike 100 kpa min/min
    - 10 = bike 75 kpa min/min
    - 11 = bike 50 kpa min/min
    - 12 = arm ergometer
29. thaldur: Duración de la prueba de esfuerzo en minutos
30. thaltime: Tiempo en que se notó la depresión de la medida ST

31. met: Logrado
32. thalach: Ritmo cardíaco máximo alcanzado
33. thalrest: Frecuencia cardíaca en reposo
34. tpeakbps: Presión arterial máxima (primera de 2 partes)
35. tpeakbpd: Pico de presión arterial durante el ejercicio (segundo de 2 partes)
36. dummy
37. trestbpd: Presión arterial en reposo
38. exang: Angina inducida por el ejercicio (1 = si; 0 = no)
39. xhypo: 1 = si; 0 = no)
40. oldpeak = Depresión ST inducida por el ejercicio relativo al descanso
41. slope: La pendiente del segmento ST durante el máximo ejercicio
    Valor 1: upsloping
    Valor 2: flat
    Valor 3: downsloping
42. rldv5: Altura en reposo
43. rldv5e: Altura en el ejercicio máximo
44. ca: Número de vasos principales (0-3) coloreados por fluoroscopia
45. restckm: Irrelevante
46. exerckm: Irrelevante
47. restef: Fracción de eyección
48. restwm: Anormalidad de movimiento
    - 0 = none
    - 1 = mild or moderate
    - 2 = moderate or severe
    - 3 = akinesis or dyskmem
49. exeref: Fracción de eyección
50. exercem: Movimiento en la pared

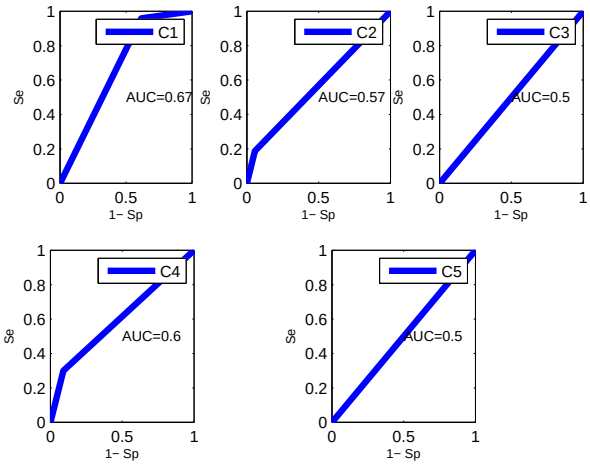
51. thal: 3 = normal; 6 = fixed defect; 7 = reversable defect
52. thalsev: No utilizado
53. thalpul: No utilizado
54. earlobe: No utilizado
55. cmo: Mes de cateterismo cardíaco
56. cday: Día de cateterismo cardíaco
57. cyr: Año de cateterismo cardíaco
58. num: Diagnóstico de enfermedad cardíaca (estado de la enfermedad angiográfica)
    Valor 0: Estrechamiento de diámetro menor que 50 %
    Valor 1: Estrechamiento de diámetro mayor que 50 %
    (los atributos 59 a 68 son vasos mayores)
59. lmt
60. ladprox
61. laddist
62. diag
63. cxmain
64. ramus
65. om1
66. om2
67. rcaprox
68. rcadist
69. lvx1: No utilizado
70. lvx2: No utilizado
71. lvx3: No utilizado
72. lvx4: No utilizado
73. lvf: No utilizado

74. cathef: No utilizado
75. junk: No utilizado
76. name: Apellido del paciente (se reemplazó con la cadena ficticia ”nombre”)

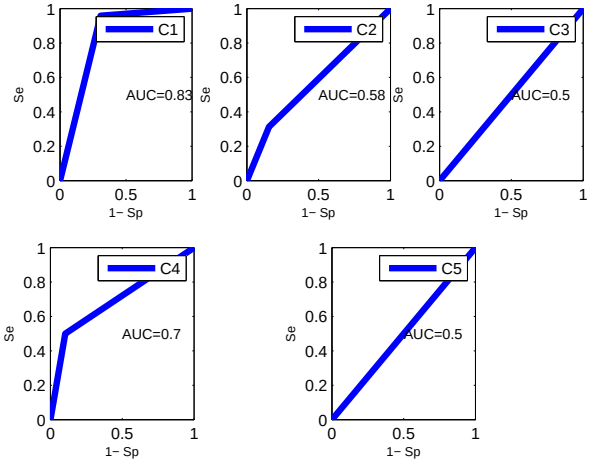
D. Curvas ROC, aplicación de clasificadores en cascada



(a) ANN

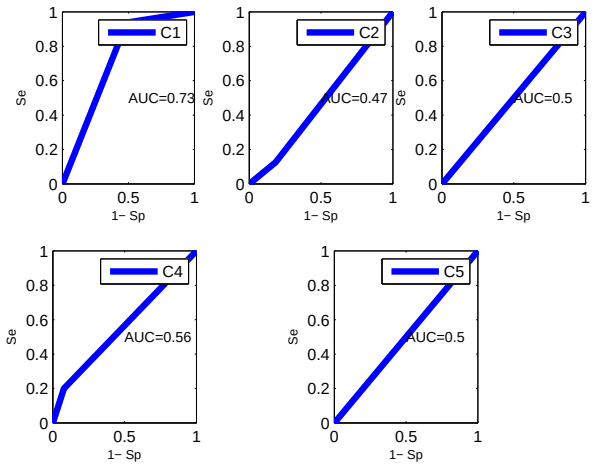


(b) PC

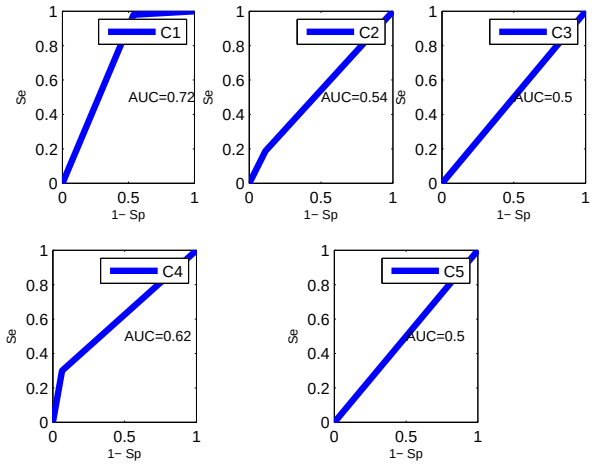


(c) *K*-NN

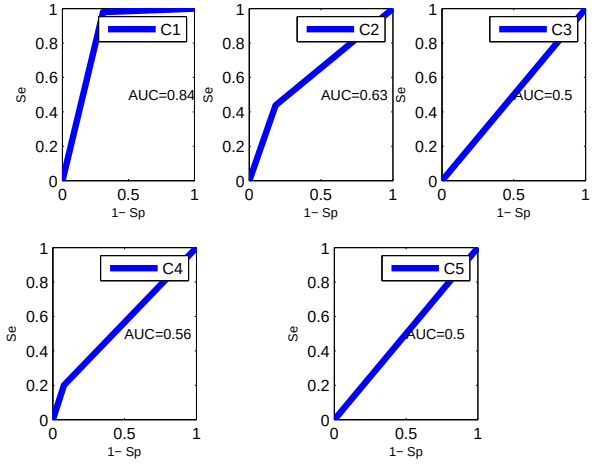
Figura D.1.: Curvas ROC para la base de datos de Cleveland, aplicando diferentes clasificadores biclase en cascada con ANN como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

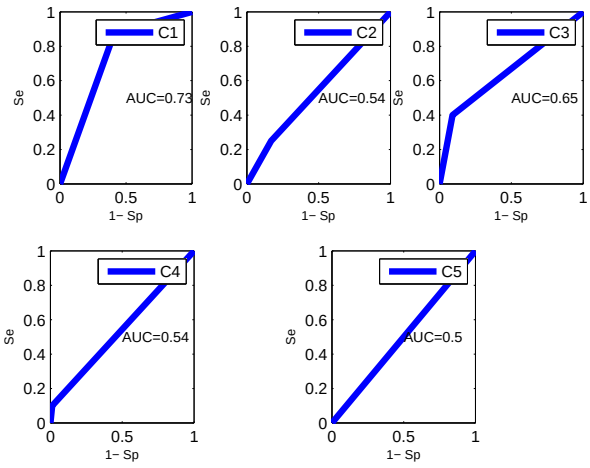


(b) PC

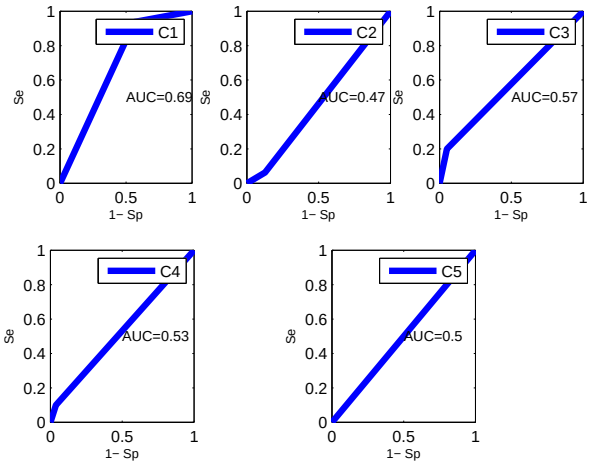


(c) *K*-NN

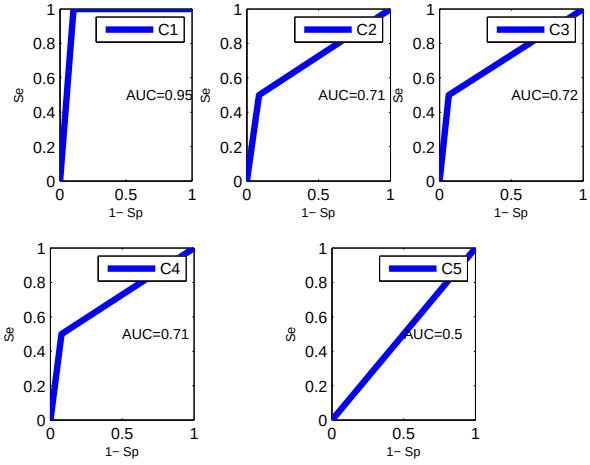
Figura D.2.: Curvas ROC para la base de datos de Cleveland, aplicando diferentes clasificadores biclase en cascada con PC como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

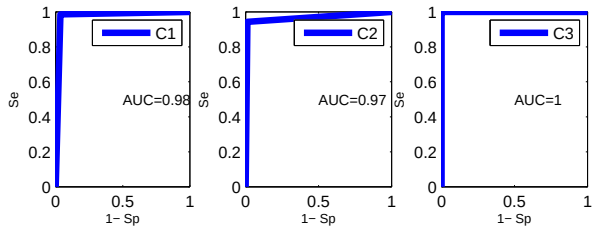


(b) PC

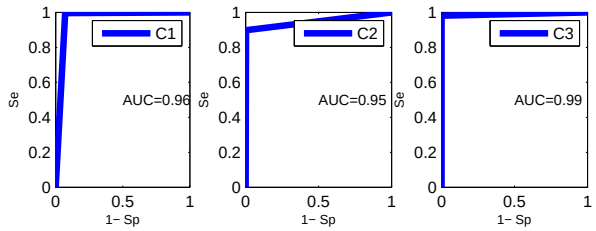


(c) *K*-NN

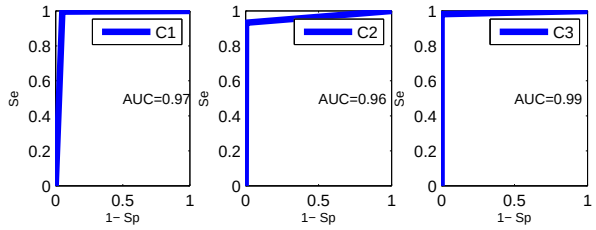
Figura D.3.: Curvas ROC para la base de datos de Cleveland, aplicando diferentes clasificadores biclase en cascada con *K*-NN como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

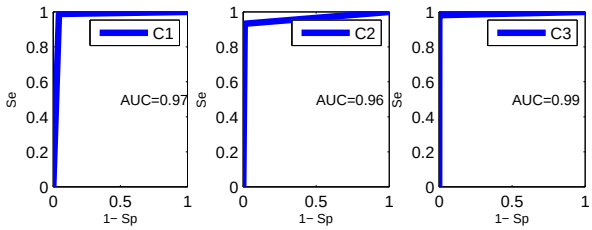


(b) PC

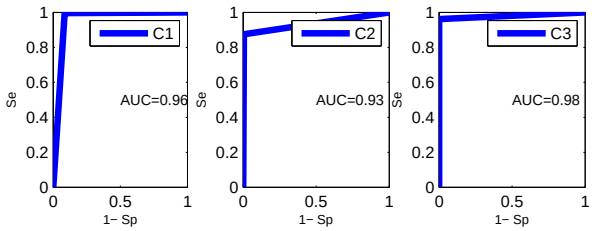


(c) *K*-NN

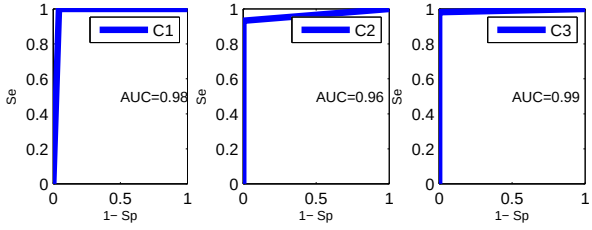
Figura D.4.: Curvas ROC para la base de datos de cardiotocografía, aplicando diferentes clasificadores biclase en cascada con ANN como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

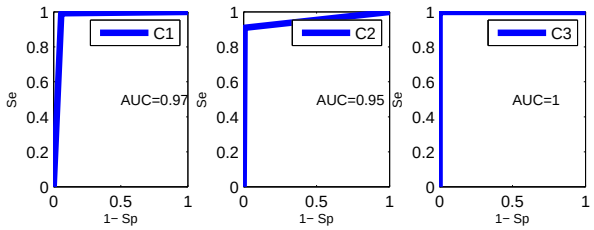


(b) PC

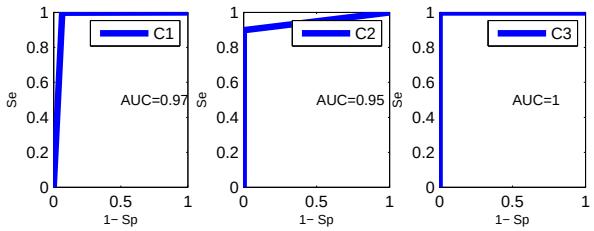


(c) *K*-NN

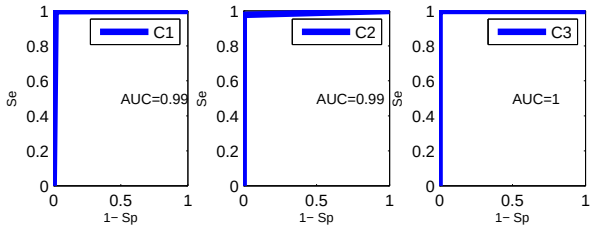
Figura D.5.: Curvas ROC para la base de datos de cardiotocografía, aplicando diferentes clasificadores biclase en cascada con PC como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

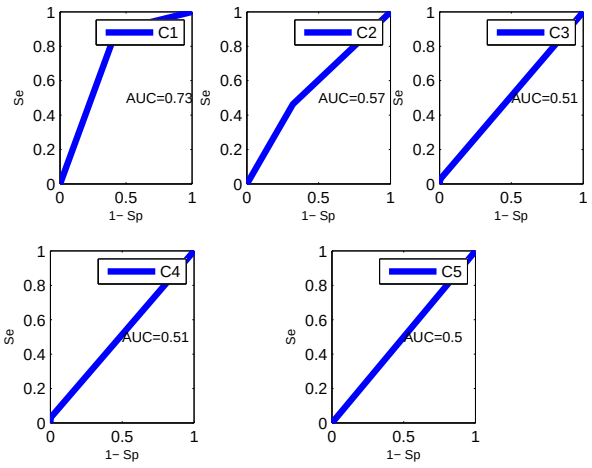


(b) PC

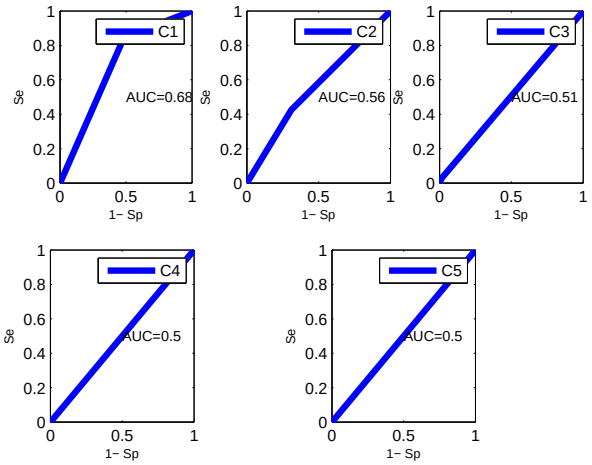


(c) *K*-NN

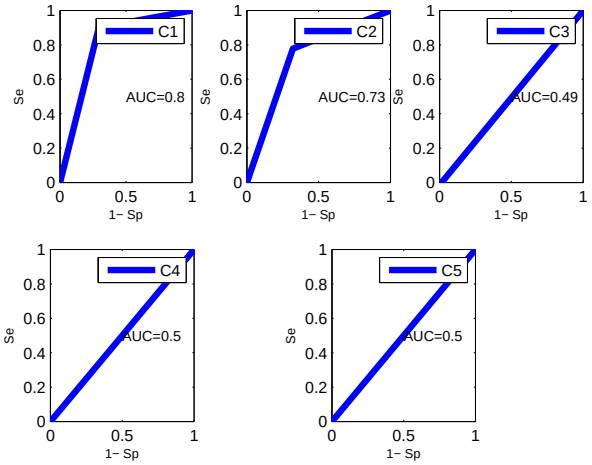
Figura D.6.: Curvas ROC para la base de datos de cardiotocografía, aplicando diferentes clasificadores biclase en cascada con *K*-NN como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

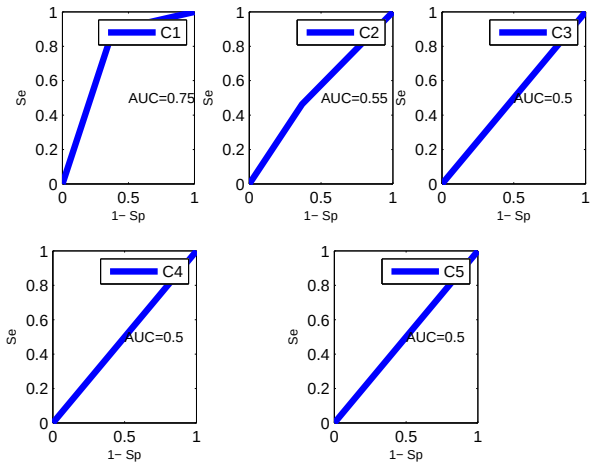


(b) PC

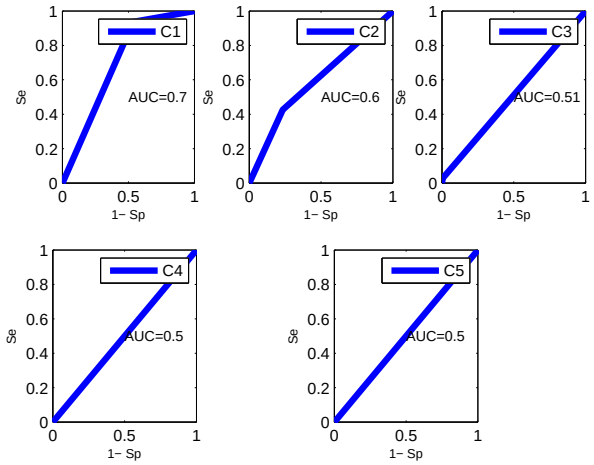


(c) *K*-NN

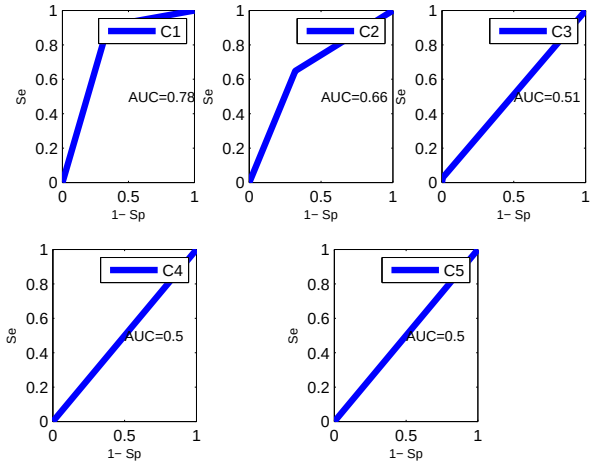
Figura D.7.: Curvas ROC para la base de datos de Cleveland ampliado, aplicando diferentes clasificadores biclase en cascada con ANN como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

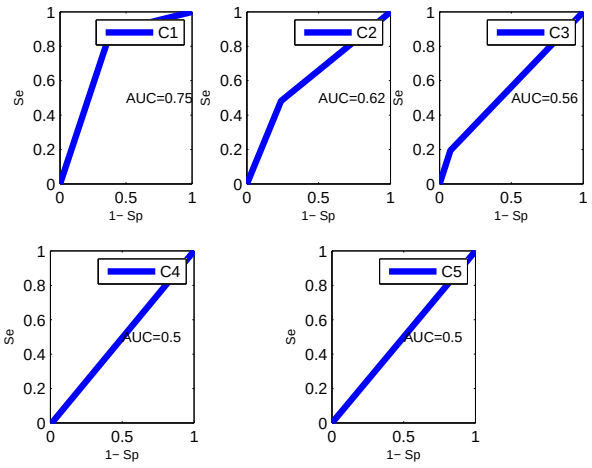


(b) PC

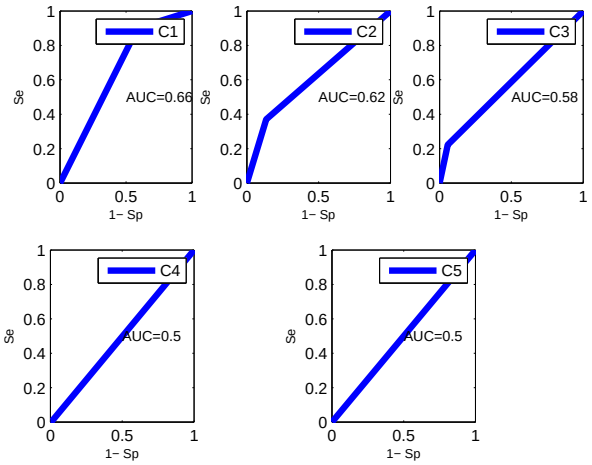


(c) *K*-NN

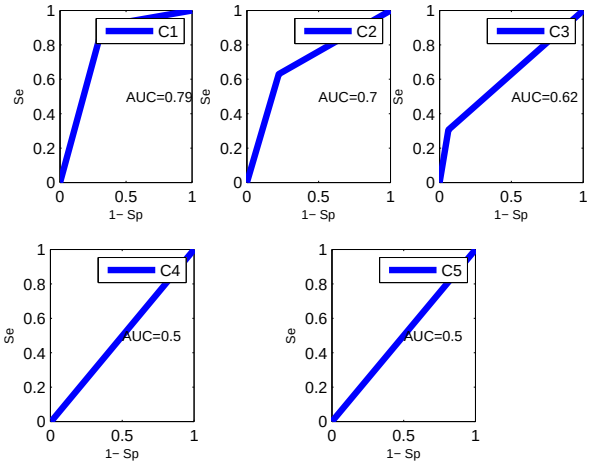
Figura D.8.: Curvas ROC para la base de datos de Cleveland ampliado, aplicando diferentes clasificadores biclase en cascada con PC como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

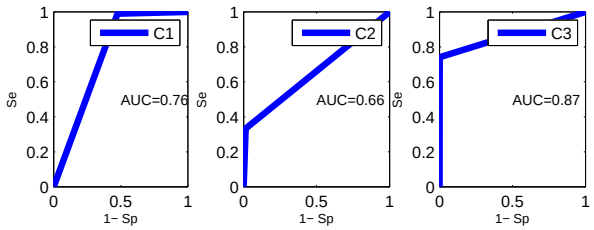


(b) PC

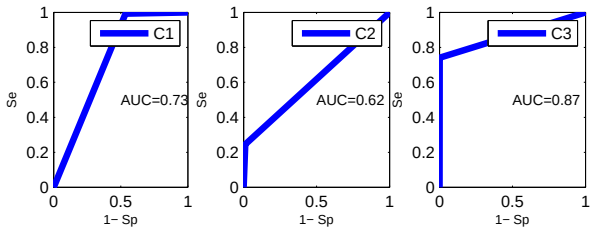


(c) *K*-NN

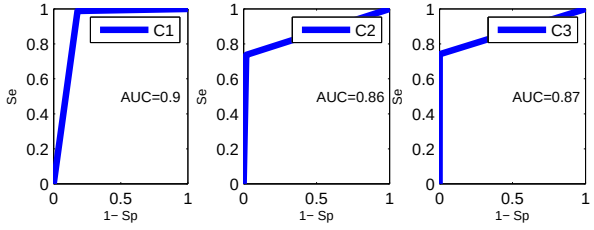
Figura D.9.: Curvas ROC para la base de datos de Cleveland ampliado, aplicando diferentes clasificadores biclase en cascada con *K*-NN como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

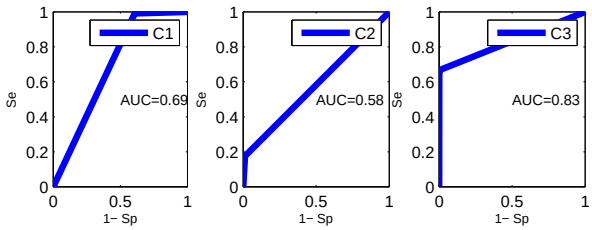


(b) PC

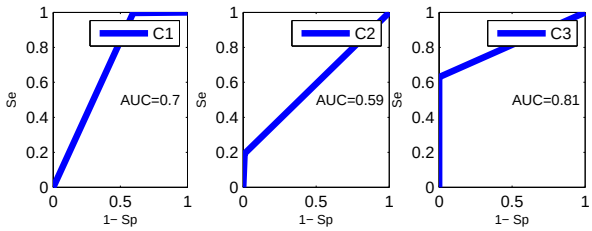


(c) *K*-NN

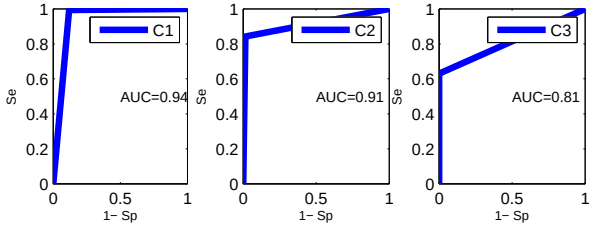
Figura D.10.: Curvas ROC para la base de datos de hipotiroidismo, aplicando diferentes clasificadores biclase en cascada con ANN como segundo clasificador multiclase en la etapa de adaptación



(a) ANN

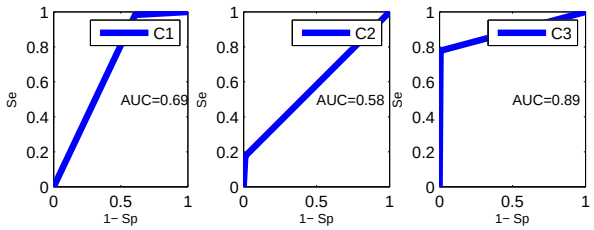


(b) PC

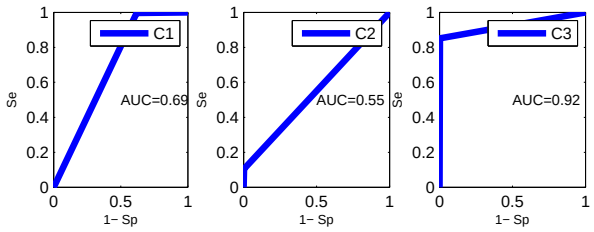


(c) *K*-NN

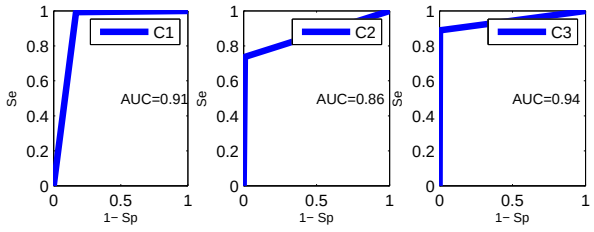
Figura D.11.: Curvas ROC para la base de datos de hipotiroidismo, aplicando diferentes clasificadores biclase en cascada con PC como segundo clasificador multiclase en la etapa de adaptación



(a) ANN



(b) PC



(c) *K*-NN

Figura D.12.: Curvas ROC para la base de datos de hipotiroidismo, aplicando diferentes clasificadores biclase en cascada con *K*-NN como segundo clasificador multiclase en la etapa de adaptación

Bibliografía

[1] Janet L. Kolodner. Maintaining organization in a dynamic long-term memory*. *Cognitive Science*, 7(4):243–280, 1983.

[2] Francisco Azuaje. Witten ih, frank e: Data mining: Practical machine learning tools and techniques 2nd edition. *BioMedical Engineering OnLine*, 5(1):51, 2006.

[3] Mark A. Hall. Correlation-based feature selection for machine learning. Technical report, 1999.

[4] E. Plaza A.Aamodt. Case-based reasoning: Foundational issues, methodological variations, and system approaches, 1994.

[5] Otto Kühn and Andreas Abecker. Corporate memories for knowledge management in industrial practice: Prospects and challenges. *j-jucs*, 3(8):929–954, August 1997.

[6] Simon C.K. Shiu and Sankar K. Pal. Case-based reasoning: Concepts, features and soft computing. *Applied Intelligence*, 21(3):233–238, 2004.

[7] Stefania Montani. How to use contextual knowledge in medical case-based reasoning systems: A survey on very recent trends. *Artif. Intell. Med.*, 51(2):125–131, February 2011.

[8] John Anderson. *The Architecture of cognition*. Lawrence Erlbaum Associates, 1983.

[9] Bartosz Krawczyk, MichałWoźniak, and Francisco Herrera. On the usefulness of one-class classifier ensembles for decomposition of multi-class problems. *Pattern Recogn.*, 48(12):3969–3982, December 2015.

[10] Seokho Kang, Sungzoon Cho, and Pilsung Kang. Multi-class classification via heterogeneous ensemble of one-class classifiers. *Engineering Applications of Artificial Intelligence*, 43:35 – 43, aug 2015.

[11] Roger C Schank and Robert P Abelson. *Scripts, plans, goals and understanding: An inquiry into human knowledge structures*. Oxford, England, 1977.

[12] Simon C.K. Shiu and Sankar K. Pal. Case-based reasoning: Concepts, features and soft computing. *Applied Intelligence*, 21(3):233–238, 2004.

[13] S. Begum, M. U. Ahmed, P. Funk, N. Xiong, and M. Folke. Case-based reasoning systems in the health sciences: A survey of recent trends and developments. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(4):421–434, July 2011.

[14] Kuang-Hung Hsu, Chaochang Chiu, Nan-Hsing Chiu, Po-Chi Lee, Wen-Ko Chiu, Thu-Hua Liu, and Chorng-Jer Hwang. A case-based classifier for hypertension detection. *Knowledge-Based Systems*, 24(1):33 – 39, 2011.

[15] Hyunchul Ahn and Kyoung-jae Kim. Global optimization of case-based reasoning for breast cytology diagnosis. *Expert Syst. Appl.*, 36(1):724–734, January 2009.

[16] Essam Amin M. Lotfy Abdrabou and Abdel-Badeeh M. Salem. Case-based reasoning tools from shells to object-oriented frameworks. In *Proceedings of the 12th WSEAS International Conference on Computers*, ICCOMP'08, pages 781–786, Stevens Point, Wisconsin, USA, 2008. World Scientific and Engineering Academy and Society (WSEAS).

[17] Sanja Petrovic, Gulmira Khussainova, and Rupa Jagannathan. Knowledge-light adaptation approaches in case-based reasoning for radiotherapy treatment planning. *Artif Intell Med*, 68(C):17–28, March 2016.

[18] Wolfgang Wilke and Ralph Bergmann. *Techniques and knowledge used for adaptation during case-based problem solving*, pages 497–506. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[19] Isabelle Bichindaritz and Cindy Marling. Case-based reasoning in the health sciences: What's next? *Artif. Intell. Med.*, 36(2):127–135, February 2006.

[20] Phyllis Koton. Using experiences for in learning and problem solving. *Eng. Comut. Sci.mMIT/LCS/TR-441*, 1989.

[21] Ray Bareiss. *Exemplar Based Knowledge Acquisition: A Unified Approach to Concept Representati on, Classification, and Learning*. Academic Press Professional, Inc., San Diego, CA, USA, 1989.

[22] Mobyen Uddin Ahmed, Shahina Begum, Peter Funk, Ning Xiong, and Bo von Scheele. A multi-module case-based biofeedback system for stress treatment. *Artificial Intelligence in Medicine*, 51(2):107 – 115, 2011. Advances in Case-Based Reasoning in the Health Sciences.

[23] Juan F. De Paz, Javier Bajo, Vivian F. López, and Juan M. Corchado. Biomedic organizations: An intelligent dynamic architecture for kdd. *Information Sciences*, 224:49 – 61, 2013.

[24] Juan M. Corchado, Juan F. De Paz, Sara Rodríguez, and Javier Bajo. Model of experts for decision support in the diagnosis of leukemia patients. *Artificial Intelligence in Medicine*, 46(3):179 – 200, 2009.

[25] Juan F. De Paz, Sara Rodríguez, Javier Bajo, and Juan M. Corchado. Case-based reasoning as a decision support system for cancer diagnosis: A case study. *Int. J. Hybrid Intell. Syst.*, 6(2):97–110, April 2009.

[26] Juan F. De Paz, Javier Bajo, Vicente Vera, and Juan M. Corchado. Microcbr: A case-based reasoning architecture for the classification of microarray data. *Applied Soft Computing*, 11(8):4496 – 4507, 2011.

[27] J.M. Juárez, M. Campos, A. Gomariz, J. Palma, and R. Marin. A reuse-based cbr system evaluation in critical medical scenarios. In *Tools with Artificial Intelligence, 2009. ICTAI '09. 21st International Conference on*, pages 261–268, Nov 2009.

[28] M. H. Fazel Zarandi, M. Zarinbal, and M. Izadi. Systematic image processing for diagnosing brain tumors: A type-ii fuzzy expert system approach. *Applied Soft Computing*, 11(1):285–294, January 2011.

[29] Daqing Chen and Phillip Burrell. Case-based reasoning system and artificial neural networks: A review. *Neural Computing & Applications*, 10(3):264–276, 2001.

[30] Barbara A. Kitchenham. Systematic review in software engineering: Where we are and where we should be going. In *Proceedings of the 2Nd International Workshop on Evidential Assessment of Software Technologies*, EAST '12, pages 1–2, New York, NY, USA, 2012. ACM.

[31] Chun-Ling Chuang. Case-based reasoning support for liver disease diagnosis. *Artif. Intell. Med.*, 53(1):15–23, September 2011.

[32] Ernesto Ocampo, Mariana Maceiras, Silvia Herrera, Cecilia Maurente, Daniel Rodríguez, and Miguel A. Sicilia. Comparing bayesian inference and case-based reasoning as support techniques in the diagnosis of acute bacterial meningitis. *Expert Systems with Applications*, 38(8):10343 – 10354, 2011.

[33] S.L. Ting, S.K. Kwok, Albert H.C. Tsang, and W.B. Lee. A hybrid knowledge-based approach to supporting the medical prescription for general practitioners: Real case in a hong kong medical center. *Knowledge-Based Systems*, 24(3):444 – 456, 2011.

[34] O. U. Obot and Faith-Michael E. Uzoka. A framework for application of neuro-case-rule base hybridization in medical diagnosis. *Appl. Soft Comput.*, 9(1):245–253, January 2009.

[35] Rong-Ho Lin and Chun-Ling Chuang. A hybrid diagnosis model for determining the types of the liver disease. *Computers in Biology and Medicine*, 40(7):665 – 670, 2010.

[36] Carey E. Floyd, Joseph Y. Lo, and Georgia D. Tourassi. Case-based reasoning computer algorithm that uses mammographic findings for breast biopsy decisions. *American Journal of Roentgenology*, 2000.

[37] Shaker El-Sappagh, Mohammed Elmogy, and A.M. Riad. A fuzzy-ontology-oriented case-based reasoning framework for semantic diabetes diagnosis. *Artificial Intelligence in Medicine*, 65(3):179 – 208, 2015.

[38] Chin-Yuan Fan, Pei-Chann Chang, Jyun-Jie Lin, and J.C. Hsieh. A hybrid model combining case-based reasoning and fuzzy decision tree for medical data classification. *Applied Soft Computing*, 11(1):632 – 644, 2011.

[39] João Vilhena, Henrique Vicente, M. Rosário Martins, José M. Grañeda, Filomena Caldeira, Rodrigo Gusmão, João Neves, and José Neves. A case-based reasoning view of thrombophilia risk. *Journal of Biomedical Informatics*, 62:265 – 275, 2016.

[40] Sara Rodríguez, Juan F. De Paz, Javier Bajo, and Juan M. Corchado. Applying cbr systems to micro array data classification. *Advances in Soft Computing*, 49:102–111, 2009.

[41] Ruben Nicolas, Albert Fornells, Elisabet Golobardes, Guiomar Corral, Susana Puig, and Josep Malvehy. Derma: A melanoma diagnosis platform based on collaborative multilabel analog reasoning. *The Scientific World Journal*, 2014, 2014.

[42] Alsane Sene, Bernard Kamsu-Foguem, and Pierre Rumeau. Telemedicine framework using case-based reasoning with evidences. *Computer Methods and Programs in Biomedicine*, vol. 121(n?1):pp. 21–35, August 2015. Thanks to Elsevier editor. The definitive version is available at : http://www.journals.elsevier.com/computer-methods-and-programs-in-biomedicine/.

[43] Jinghua Shi, Qiang Su, Chenpeng Zhang, Gang Huang, and Yan Zhu. An intelligent decision support algorithm for diagnosis of colorectal cancer through serum tumor markers. *Computer Methods and Programs in Biomedicine*, 100(2):97 – 107, 2010.

[44] Enrico Bertini and Denis Lalanne. Surveying the complementary role of automatic data analysis and visualization in knowledge discovery. *Proceedings of the ACM SIGKDD Workshop on Visual Analytics and Knowledge Discovery Integrating Automated Analysis with Interactive Exploration - KDD '09*, pages 12–20, 2009.

[45] Diego H Peluffo-Ordónez, John A Lee, and Michel Verleysen. Short review of dimensionality reduction methods based on stochastic neighbour embedding. In *Advances in Self-Organizing Maps and Learning Vector Quantization*, pages 65–74. Springer, 2014.

[46] Ingwer Borg. *Modern multidimensional scaling: Theory and applications*. Springer, 2005.

[47] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

[48] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[49] Geoffrey E Hinton and Sam T Roweis. Stochastic neighbor embedding. In *Advances in neural information processing systems*, pages 833–840, 2002.

[50] John A Lee, Emilie Renard, Guillaume Bernard, Pierre Dupont, and Michel Verleysen. Type 1 and 2 mixtures of kullback-leibler divergences as cost functions in dimensionality reduction based on similarity preservation. *Neurocomputing*, 2013.

[51] John A Lee, Diego H Peluffo-Ordóñez, and Michel Verleysen. Multi-scale similarities in stochastic neighbour embedding: Reducing dimensionality while preserving both local and global structure. *Neurocomputing*, 2015.

[52] J. Cano F. Herrera. Técnicas de reducción de datos en kdd. el uso de algoritmos evolutivos para la selección de instancias. *Actas del I Seminario Sobre Sistemas Inteligentes*, 2006.

[53] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The WEKA data mining software: An update. *SIGKDD Explorations*, 11(1):10–18, 2009.

[54] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998.

[55] Ricardo Henao and Jorge Eduardo Hurtado Gómez. Selección de hiperparámetros en máquinas de soporte vectorial. *Universidad Nacional de Colombia*, 2004.

[56] J.E. Ormrod and K.M. Davis. *Human learning*. Merrill, 2004.

[57] B. Schölkopf and A. J. Smola. *Learning with Kernels*. 2002.

[58] Hetal Bhavsar and Amit Ganatra. Radial basis polynomial kernel (rbpk): A generalized kernel for support vector machine. *International Journal of Computer Science and Information Security*, 14(4):296, 2016.

[59] Liming Shen, Huiling Chen, Zhe Yu, Wenchang Kang, Bingyu Zhang, Huaizhong Li, Bo Yang, and Dayou Liu. Evolving support vector machines using fruit fly optimization for medical data classification. *Knowledge-Based Systems*, 96:61 – 75, 2016.

[60] Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425, 2002.

[61] Yi Liu and Yuan F Zheng. One-against-all multi-class svm classification using reliability measures. In *IEEE International Joint Conference on Neural Networks*, volume 2, pages 849–854. IEEE, 2005.

[62] Robert Breunig. Nonparametric density estimation for stratified samples. *The Australian National University working papers in economics and econometrics*, (459), 2005. Center for Economic Policy Research, Economics Program, Research School of Social Sciences, The Australian National University, Canberra ACT 0200, AUSTRALIA.

[63] R. O. Camachoo J. E. Rodríguez, E. A. Blanco. Clasificación de datos usando el método k-nn. *Revista Vínculos*, 2013.

[64] S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR Upper Saddle River, NJ, USA, 2004.

[65] C. Saavedra F. Izaurieta. Redes nerunales artificiales. departamento de física, universidad de concepción chile. 2000.

[66] D. J. Matich. Redes neuronales: Conceptos básicos y aplicaciones. *Cátedra de informática aplicada a la ingeniería de procesos - orientación I*, 2001.

[67] Robert P. W. Duin and David M. J. Tax. *Classifier conditional posterior probabilities*, pages 611–619. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[68] National Guideline Clearinghouse (NGC). Intrapartum care: care of healthy women and their babies during childbirth. 2014.

[69] M. Lichman. UCI machine learning repository, 2013.

[70] Edward R. Dougherty. Performance of error estimators for classification. *Current Bioinformatics*, 5(1):53–67, 2010.

[71] Jerome Friedman Trevor Hastie, Robert Tibshirani. *The Elements of Statistical Learning*. Springer-Verlag New York, 2009.

[72] P.A. Flach. The geometry of roc space: understanding machine learning metrics through roc isometrics. In *Proc. 20th International Conference on Machine Learning (ICML'03)*, pages 194–201. AAAI Press, January 2003.

[73] Simon J. Sheather. Density estimation. *Statist. Sci.*, 19(4):588–597, 11 2004.

[74] J Moreno, D Rodríguez, MA Sicilia, JC Riquelme, and R Ruiz. Smote-i: mejora del algoritmo smote para balanceo de clases minoritarias. *Actas de los Talleres de las Jornadas de Ingeniería del Software y Bases de Datos*, 3(1), 2009.

[75] Manuel Miguel Giménez Arjona, Ivan López Arévalo, and Aida Valls Mateu. Estudio para la implementación de un sistema de razonamiento basado en casos.

[76] Sumit Bhatia, Praveen Prakash, and G. N. Pillai. Svm based decision support system for heart disease classification with integer-coded genetic algorithm to select critical features, 2008.

[77] Sundar.c, M.chitradevi, and G. Geetharamani. Article: Classification of cardiotocogram data using neural network based machine learning technique. *International Journal of Computer Applications*, 47(14):19–25, June 2012. Full text available.

[78] Charles Elkan. The foundations of cost-sensitive learning. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'01, pages 973–978, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[79] Gary Weiss, Kate McCarthy, and Bibi Zabar. Cost-sensitive learning vs. sampling: Which is best for handling unbalanced classes with unequal error costs? In Robert Stahlbock, Sven F. Crone, and Stefan Lessmann, editors, *DMIN*, pages 35–41. CSREA Press, 2007.

[80] Sanja Petrovic, Nishikant Mishra, and Santhanam Sundar. A novel case based reasoning approach to radiotherapy planning. *Expert Systems with Applications*, 38(9):10759 – 10769, 2011.

[81] Mei-Ling Huang, Yung-Hsiang Hung, Wen-Ming Lee, R. K. Li, and Tzu-Hao Wang. Usage of case-based reasoning, neural network and adaptive neuro-fuzzy inference system classification techniques in breast cancer dataset classification diagnosis. *Journal of Medical Systems*, 36(2):407–414, 2012.

[82] Dong-Xiao Gu, Chang-Yong Liang, Xing-Guo Li, Shan-Lin Yang, and Pei Zhang. Intelligent technique for knowledge reuse of dental medical records based on case-based reasoning. *J. Med. Syst.*, 34(2):213–222, April 2010.

[83] Cindy Marling, Jay Shubrook, and Frank Schwartz. Toward case-based reasoning for diabetes management: A preliminary clinical study and decision support system prototype. *Computational Intelligence*, 25(3):165–179, 2009.

[84] Adina Eunice Tutac, Daniel Racoceanu, Wee-Kheng Leow, Henning Müller, Thomas Choudary Putti, and V-I. Cretu. Toward translational incremental similarity-based reasoning in breast cancer grading. In *Proc. SPIE Medical Imaging: Computer Aided Diagnosis*, volume 7260, page 1–12, 02/2009 2009.

[85] Nicandro Cruz-Ramírez, Héctor-Gabriel Acosta-Mesa, Humberto Carrillo-Calvet, and Rocío-Erandi Barrientos-Martínez. Discovering interobserver variability in the cytodiagnosis of breast cancer using decision trees and bayesian networks. *Applied Soft Computing*, 9(4):1331–1342, September 2009.

[86] Tina Waligora Rainer Schmidt. *Influenza Forecast: Case-Based Reasoning or Statistics?* Springer Berlin Heidelberg, Berlin, Germany, 2007.

[87] Aliasghar Khorsand, Senta Graf, Heinz Sochor, Ernst Schuster, and Gerold Porenta. Automated assessment of myocardial {SPECT} perfusion scintigraphy: A comparison of different approaches of case-based reasoning. *Artificial Intelligence in Medicine*, 40(2):103 – 113, 2007.

[88] Andres El-Fakdi, Francisco Gamero, Joaquim Meléndez, Vincent Auffret, and Pascal Haigron. exitcdss: A framework for a workflow-based cbr for interventional clinical decision support systems and its application to tavi. *Expert Systems with Applications*, 41(2):284 – 294, 2014.

[89] K. Ashwin Kumar, Yashwardhan Singh, and Sudip Sanyal. Hybrid approach using case-based reasoning and rule-based reasoning for domain independent clinical decision support in {ICU}. *Expert Systems with Applications*, 36(1):65 – 71, 2009.

[90] Shih-Wei Lin and Shih-Chieh Chen. Parameter tuning, feature selection and weight assignment of features for case-based reasoning by artificial immune system. *Applied Soft Computing*, 11(8):5042 – 5052, 2011.

[91] Manjeevan Seera and Chee Peng Lim. A hybrid intelligent system for medical data classification. *Expert Syst. Appl.*, 41(5):2239–2249, April 2014.

[92] Elpiniki I. Papageorgiou. A new methodology for decisions in medical informatics using fuzzy cognitive maps based on fuzzy rule-extraction techniques. *Applied Soft Computing*, 11(1):500 – 513, 2011.

[93] Yoon-Joo Park, Se-Hak Chun, and Byung-Chun Kim. Cost-sensitive case-based reasoning using a genetic algorithm: Application to medical diagnosis. *Artificial Intelligence in Medicine*, 51(2):133 – 145, 2011. Advances in Case-Based Reasoning in the Health Sciences.

[94] Pablo Gay, Beatriz López, Albert Plà, Jordi Saperas, and Carles Pous. Enabling the use of hereditary information from pedigree tools in medical knowledge-based systems. *J. Biomed. Inform.*, 46(4):710 – 720, 2013.

[95] Abdul Majid, Safdar Ali, Mubashar Iqbal, and Nabeela Kausar. Prediction of human breast and colon cancers from imbalanced data using nearest neighbor and support vector machines. *Computer Methods and Programs in Biomedicine*, 113(3):792 – 808, 2014.

[96] Dina A. Sharaf-El-Deen, Ibrahim F. Moawad, and M. E. Khalifa. A new hybrid case-based reasoning approach for medical diagnosis systems. *Journal of Medical Systems*, 38(2), 2014.

[97] Subhagata Chattopadhyay, Suvendu Banerjee, Fethi A. Rabhi, and U. Rajendra Acharya. A case-based reasoning system for complex medical diagnosis. *Expert Systems*, 30(1):12–20, 2013.

[98] Rached Omer Agwil and Divya Prakash Shrivastava. Integrated thallassaemia decision support system. *W. Trans. on Comp.*, 9(8):857–867, August 2010.

[99] Roxana Martín Ramos, Rosa María Ramos Palmero, Ricardo Grau Ávalos, and María Matilde García Lorenzo. Aplicación de métodos de selección de atributos para determinar factores relevantes en la evaluación nutricional de los niños. *Gaceta Médica Espirituana*, 9(1):7, 2012.