

Sistema inteligente de ayuda al diagnóstico del Deterioro
Cognitivo Leve basado en computación neuronal: Un
problema de clases no balanceadas

José Manuel Martínez García
jose.martinez104@alu.ulpgc.es

Tutores:

Carmen Paz Suárez Araujo
cpsuarez@dis.ulpgc.es

Antonio Carlos Domínguez Brito
adominguez@iusiani.ulpgc.es

Máster Oficial en Sistemas Inteligentes y Aplicaciones Numéricas en Ingeniería

Instituto Universitario de Sistemas Inteligentes y Aplicaciones Numéricas en
Ingeniería

Universidad de Las Palmas de G.C.
14 de diciembre de 2012

Trabajo de Fin de Máster

Título: Sistema inteligente de ayuda al diagnóstico del Deterioro Cognitivo Leve basado en computación neuronal: Un problema de clases no balanceadas.

Apellidos y nombre del alumno: Martínez García, José Manuel.

Fecha : 14 de diciembre de 2012

Tutora: Suárez Araujo, Carmen Paz

Tutor: Domínguez Brito, Antonio Carlos

Agradecimientos

Antes de comenzar estos agradecimientos me gustaría indicar que no sólo quisiera dar las gracias a todas aquellas personas que me han ayudado y acompañado durante la elaboración de este Trabajo de Fin de Máster, sino también a las que lo hicieron durante toda mi formación académica universitaria, ya que, por motivos de tiempo no pude mencionarlas adecuadamente en mi Proyecto de Fin de Carrera.

En primer lugar quiero dar gracias a mis tutores, porque además de guiarme en la realización de todo el trabajo, me han brindado una formación continua y personalizada. He aprendido tantas cosas de ellos que no sabría sintetizarlas aquí, pero en pocas palabras a D^a Carmen Paz Suárez Araujo quiero agradecerle haber confiado en mí para ingresar en su grupo de investigación COMCIENCIA, y permitirme trabajar como investigador, formándome y aconsejándome en cada momento. Gracias por todas sus recomendaciones, sugerencias y correcciones que tanto me han ayudado a mejorar mis trabajos. Sin duda, hoy siento que dispongo de unas capacidades que dos años atrás no tenía. A D. Antonio Carlos Domínguez Brito quiero agradecerle todo el conocimiento que me aportó durante el desarrollo de una parte de CoolBOT. Hizo que mis capacidades como ingeniero informático mejoraran notablemente gracias a ese proyecto y me sintiera más seguro de mí mismo como profesional de la informática. Es por ello y por muchísimas otras razones más por las que me siento orgulloso de haber tenido la oportunidad de coincidir con estas dos personas para que dirigieran mis trabajos y fueran mis mentores.

Mi siguiente agradecimiento es para mis padres, D. José Manuel Martínez Teja y D^a María Isabel García Delgado. A los dos quiero darles gracias de todo corazón por los valores que me han inculcado, por transmitirme ser perseverante y fiel a unos objetivos y por recibirme cada vez que llegaba a casa después de trabajar con tanta ilusión. Gracias a mi madre por todos los almuerzos que pude disfrutar cada tarde de trabajo, porque me ayudaron a rendir por las tardes “con muy buen sabor de boca”. A mi padre quiero agradecerle su actitud motivadora, sus frases del estilo “si algo no te ha salido hoy, mañana será otro día” y sus noticias matutinas. Si he llegado a dónde he llegado, no me cabe la menor duda que es más mérito de ellos dos que mío.

Una de las personas que merecen un reconocimiento especial es mi amigo Daniel. Muchos almuerzos y muchas tardes hemos vivido juntos trabajando, y él es de esas personas que siempre están dispuestas a ayudar. Cada vez que se me presentó algún problema, ahí estuvo él para escucharlo y tratar de aportarme una solución. Estoy seguro de que si no hubiésemos sido amigos, lo hubiera hecho igualmente, porque esa es la impronta de Daniel.

A mi amigo Abraham le agradezco profundamente todas las asignaturas que compartimos durante la carrera. Ha sido una suerte tenerlo como amigo y como compañero de prácticas, pues es difícil decidir cuál de las dos facetas se le da mejor. A Ibrahim también quiero agradecerle todas las veces que se ha brindado a ayudarme, que ahora mismo no las recuerdo, de tantas que fueron. A Carlos le agradezco todas las prácticas en las que trabajamos juntos, tanto en la carrera como en el máster, y su espíritu trabajador.

Quisiera mencionar a mis compañeros del grupo de investigación COMCIENCIA, por sus valiosas aportaciones. Gracias a Patricio, Pablo, Yeray, y Miguel Ángel. A éste último, le agradezco en especial su buena predisposición a echarme siempre una mano en todo lo que yo necesitara y por transmitirme esa pasión por el trabajo y por lo que uno dedica su tiempo.

Continúo con los agradecimientos hacia mi amiga Soraya, mi compañera de fatigas durante todo el PFC y posterior compañera de trabajo, gracias por cada tarde que trabajamos codo con codo desde nuestra primera aplicación cliente/servidor hasta el último día que nos sentamos juntos en el trabajo. Espero y deseo que algún día volvamos a coincidir en otro trabajo. A mi amiga Adaya quisiera agradecerle su don de la oportunidad para ayudarme a evadirme de vez en cuando de lo que esté haciendo y tomar un café para ponernos al día. Tiene tan desarrollado ese don, que justo ha pasado a visitarme por el laboratorio mientras escribía estas palabras.

No podía faltar en estas letras mi pareja Beatriz, que me ha ayudado durante la elaboración de este trabajo y tantas madrugadas y fines de semana esperó a mi lado a que apagara el ordenador. Sin duda, su compañía y su paciencia me han dado la energía que necesitaba en muchísimos momentos.

Otra persona a la que le doy gracias es a mi hermana Aída y a los dos sobrinos que me ha dado, Néstor y Javier. Sin duda estos últimos han hecho de su tío una persona más feliz.

A pesar de que podría seguir nombrando personas y dar motivos por los que aparecer en esta sección, no quisiera dilatarme más de lo que ya lo he hecho. Los que no hayan aparecido no duden que les aprecio, pero en algún momento tenía que acabar estos agradecimientos. Gracias a Airam, Kilian, María, Pedro, José Carlos, Sara, Andrés, y un largo etcétera.

Índice general

Índice de figuras	3
Indice de tablas	5
1. Introducción	9
1.1. Contextualización del problema	9
1.2. Estado del arte de los sistemas de detección del DCL	10
1.3. Computación neuronal	12
1.3.1. Redes Neuronales Artificiales supervisadas	14
Perceptrón simple	14
Perceptrón multicapa	15
1.3.2. Redes Neuronales Artificiales no supervisadas	16
Mapas de Kohonen	16
Red de Sanger	17
1.3.3. Redes Neuronales Artificiales híbridas	18
Red Counterpropagation	18
2. Método de sobre-muestreo extendido basado en la red neuronal de Sanger (SNEOM)	21
2.1. Distribuciones de datos no balanceadas: Un problema de los sistemas de aprendizaje clásicos	21
2.2. Sistemas auto-organizados	23
2.2.1. Funciones y métodos	25
2.2.2. Métodos proyectivos	25
2.3. SNEOM	26
2.3.1. Motivación	26
2.3.2. Diseño y Desarrollo	27
2.3.3. Aplicación de SNEOM en problemas de clasificación médica no balanceada	32
Conjuntos de datos	34
Paradigmas de clasificación	34
Validación del método	37
2.3.4. SNEOM-GUI: Interfaz Gráfica de Usuario	40
2.3.5. Formatos de ficheros compatibles	42

Índice general

2.3.6. Vistas proporcionadas por la interfaz	45
3. Sistema inteligente de ayuda al diagnóstico del DCL basado en computación neuronal	49
3.1. Análisis de algunos criterios clínicos para la detección del DCL	49
3.2. Cuerpo de datos	51
3.3. Sistema de ayuda a la detección del DCL basado en la red neuronal Counterpropagation	54
3.4. Análisis comparativo de los resultados	55
4. Conclusiones y trabajos futuros	59
Lista de publicaciones	61
Referencias	65

Índice de figuras

1.1.	Representación de una neurona biológica.	12
1.2.	Representación de una neurona abstracta.	13
1.3.	Modelo funcional de una ANN.	13
1.4.	Arquitectura tridimensional de una red de neuronas artificiales.	14
1.5.	Estructura de la red neuronal de Sanger.	17
1.6.	Estructura de la red neuronal híbrida Counter-Propagation.	18
2.1.	Diagrama de flujo del algoritmo de sobre-muestreo SNEOM para una muestra del conjunto de entrada.	28
2.2.	Esquema de las diferentes fases del método SNEOM.	29
2.3.	a) Muestras sintéticas generadas a partir de cada muestra de la clase la clase minoritaria. Las cruces verdes representan las muestras originales, las cruces azules las muestras generadas y las áreas en color rojo las regiones de aceptación para esa clase. b) Conjunto de muestras originales de la clase minoritaria más el conjunto de muestras generadas tras aplicar SNEOM.	31
2.4.	Conjunto de muestras originales más generadas tras aplicar un sobre-muestreo extendido utilizando SNEOM. Lás áreas de color verde corresponden a las regiones de aceptación de la clase mayoritaria y las áreas de color rojo a las de la clase minoritaria.	32
2.5.	a) Variable MEC frente a Barthel del conjunto de datos de entrada. b) Variable MEC frente a Barthel y nivel educativo del conjunto de datos de entrada. c) Variable MEC frente a Barthel del conjunto de datos tras aplicar un sobre-muestreo a la clase minoritaria. d) Variable MEC frente a Barthel y nivel educativo del conjunto de datos tras aplicar un sobre-muestreo a la clase minoritaria.	33
2.6.	Representación de una red Bayesiana.	35
2.7.	Representación de árbol de decisión.	36
2.8.	Representación gráfica de un Perceptrón con una capa oculta.	36
2.9.	Diagrama de los ficheros principales que componen SNEOM-GUI junto a sus métodos más significativos.	42
2.10.	Vista de la configuración paramétrica de la red neuronal de Sanger.	45
2.11.	Vista de la configuración paramétrica del método sobre-muestreo.	46
2.12.	Vista de la ejecución del método SNEOM.	47
2.13.	Vista bidimensional de los datos de entrada.	48
2.14.	Vista tridimensional de los datos de entrada.	48

Índice de figuras

3.1. Vista del componente integrado en EDEVITALZH para visualizar el conjunto de datos de pacientes con DCL.	52
3.2. Diagrama estructural del sistema basado en CPN.	55
3.3. a) Curvas ROC de los clasificadores basados en computación neuronal para discriminar entre pacientes de DCL de pacientes sanos. b) Curva envolvente convexa de los clasificadores basados en computación neuronal para discriminar entre pacientes de DCL de pacientes sanos.	57

Indice de tablas

2.1. Métricas obtenidas para el conjunto de datos de pacientes con DCL.	38
2.2. Métricas obtenidas para el conjunto de datos Breast Cancer Wisconsin.	39
2.3. Métricas obtenidas para el conjunto de datos Vertebral Column.	40
3.1. Métricas de rendimiento obtenidas para CPNs entrenadas con diferentes subconjuntos de variables.	50
3.2. Características demográficas, resultados de los tests neuropsicológicos y nivel educativo de los sujetos.	53
3.3. Parámetros de configuración del sistema basado en CPN.	54
3.4. Parámetros de configuración del sistema basado en MLP.	56
3.5. Resultados de clasificación para diferentes métricas utilizando clasificadores basados en CPN frente a los basados en MLP y las predicciones del facultativo.	56

Resumen y objetivos

La enfermedad de Alzheimer (EA) y otras demencias constituyen uno de los principales retos para la salud pública mundial debido a la relación entre el incremento de la longevidad poblacional y la incidencia de la patología. Los primeros síntomas aparecen varios años después del comienzo de la enfermedad y la progresión del deterioro cognitivo se incrementa con el paso del tiempo. Se hace necesario pues, realizar un diagnóstico de la patología en su estadio más temprano. Este estado, denominado Deterioro Cognitivo Leve (DCL), es muy complejo de diagnosticar, debido a la dificultad para diferenciar sus síntomas sobre todo del envejecimiento normal del cerebro y a la no existencia de una estandarización de los diferentes criterios diagnósticos al uso. Asimismo, esta difícil detección implica desequilibrios en los conjuntos de datos empleados para realizar un sistema de ayuda a la detección, originándose un problema de clasificación no balanceada.

En este trabajo se presenta un método de sobre-muestreo extendido basado en la red neuronal de Sanger, que permita abordar la clasificación de conjuntos de datos donde una de las clases se encuentra representada por un número de muestras significativamente menor que la otra clase. Se realizará un estudio de importancia de criterios diagnósticos del DCL, atendiendo a su relevancia para detectar la enfermedad y se diseñará un sistema de ayuda a la detección de esta enfermedad basado en Redes Neuronales híbridas. Los resultados obtenidos a partir este trabajo resultan de gran valor para la atención primaria, pudiendo ser utilizados también en atención especializada y en cualquier ámbito socio-sanitario.

Los objetivos principales para este Trabajo de Fin de Máster son:

- Desarrollar un método de sobre-muestreo que permita obtener un cuerpo de datos balanceado
- Analizar un conjunto de criterios clínicos practicables en atención primaria para alcanzar una alta fiabilidad en la detección del DCL
- Desarrollar un sistema inteligente basado en redes neuronales para la detección del DCL

Este documento está estructurado como sigue a continuación. En el capítulo 1 se introduce la problemática del Deterioro Cognitivo Leve seguido de un estado de arte de los trabajos desarrollados hasta la fecha para llevar a cabo un diagnóstico de la enfermedad basados en sistemas computacionales. A continuación se expone el enfoque de la computación neuronal, haciendo un barrido por varias arquitecturas neuronales que están presentes en el resto de capítulos. En el

capítulo 2, se presenta un nuevo método de sobre-muestro extendido basado en la red de Sanger (SNEOM). Se comienza el capítulo con la descripción de uno de los problemas estudiados en aprendizaje automático, las distribuciones de datos no balanceadas. Se presentan los sistemas auto-organizados, para descender hasta los métodos proyectivos, enmarcando así las futuras secciones del capítulo, que se centran en los aspectos motivacionales, el diseño y desarrollo de SNEOM, y la validación del mismo mediante su aplicación en problemas de clasificación médica no balanceada. La última sección del capítulo está dedicada a describir la interfaz de usuario desarrollada para aplicar SNEOM de una forma intuitiva y amigable. El capítulo 3 describe el sistema inteligente de ayuda al diagnóstico propuesto, basado en computación neuronal. En primer lugar se realiza un análisis de algunos criterios diagnósticos más relevantes para la detección del DCL seguido del cuerpo de datos utilizado para entrenar y validar dicho sistema y la configuración del mismo. Al término del capítulo se realiza una comparación de los resultados obtenidos. En el capítulo 4 se presentan las conclusiones generadas tras la realización de este trabajo y algunas posibles líneas de trabajo futuras. Finalmente se listan las publicaciones y comunicaciones surgidas fruto de las diferentes partes en las que está estructurado este trabajo, así como las referencias citadas desde el mismo.

Capítulo 1

Introducción

1.1. Contextualización del problema

La Enfermedad de Alzheimer (EA) es un trastorno neurodegenerativo progresivo de incidencia esporádica, en la mayoría de los casos, o familiar, que afecta inicialmente a la memoria episódica, pero que en su evolución implica a todos los sistemas cognitivos llevando al individuo a una situación de incapacidad y dependencia. Desde el punto de vista patológico sus características distintivas son la presencia de las placas seniles, que constituyen acúmulos extracelulares de beta-amiloide y los ovillos neurofibrilares formados por depósito de proteína Tau fosforilada. La EA y otras demencias constituyen uno de los más importantes retos para la salud pública en todo el mundo debido principalmente a la relación entre la longevidad poblacional y el incremento de la incidencia de la patología. Los primeros síntomas aparecen varios años después del comienzo de la enfermedad y la progresión del déficit cognitivo se incrementa con el tiempo. Con respecto a su diagnóstico, los criterios usados con más frecuencia han sido los del DSM-IV [Association, 2000] y los propuestos por el consorcio NINCDS-ADRDA [McKhann et al., 1984]. Ambos se basan en el reconocimiento del síndrome de demencia y en las características clásicas de la demencia tipo Alzheimer. Sin embargo, y a pesar de la amplia difusión de estos criterios clínicos, se ha demostrado que son poco específicos, en torno al 70 %, y que muestran escasa sensibilidad en las fases iniciales de la enfermedad [Kazee et al., 1993] [Varma et al., 1999].

Existe un período transicional entre el envejecimiento normal y el diagnóstico clínicamente probable de la EA en un estadio muy temprano. Este intervalo de tiempo ha sido descrito como Deterioro Cognitivo Leve (DCL). En 1991, Flicker et al. [Flicker et al., 1991] propuso el término DCL como un concepto pragmático emergente de las observaciones de neurólogos y psiquiatras donde personas mayores que presentaban quejas cognitivas mayoritariamente desarrollaron demencia. Seis años más tarde, Petersen et al. [Petersen et al., 1997] propuso un conjunto de criterios diagnósticos para el DCL: quejas de memoria objetivas, habilidad conservada para realizar actividades de la vida diaria, función cognitiva general normal, función de memoria anormal para la edad y ausencia de demencia. A pesar de los criterios propuestos por Petersen hace 15 años, debido a la alta variabilidad relacionada con la caracterización de sujetos con DCL, hoy en día no existe un acuerdo en el campo médico acerca de un único conjunto de criterios para el DCL [Petersen, 2004]. A pesar de esta falta de acuerdo entre diferentes criterios clínicos, varios estudios han demostrado

que el DCL está asociado con un riesgo elevado de desarrollar demencia, normalmente EA, pero este concepto es heterogéneo debido a que muchos factores convergen en una posible progresión a la EA y otras enfermedades [Quintana et al., 2012].

Debido a todo lo anteriormente expuesto, nos encontramos frente a un problema de infra-diagnóstico, llegando a alcanzar en el diagnóstico de la EA y otras demencias hasta un 95 % de casos de demencias leves no detectados [Solomon and Murphy, 2005], repercutiendo en un pronóstico y fiabilidad terapéutica limitadas. Esta precisión diagnóstica es aún menor en entornos de atención primaria o, incluso, en consultas generales de neurología o psiquiatría, fuera del ámbito de las unidades especializadas en patología cognitiva y conductual [Lim et al., 1999]. En una revisión del DCL con recomendaciones para investigaciones futuras, los autores defienden el desarrollo de medidas funcionales y neuropsicológicas apropiadas, métodos fiables para evaluar la progresión e instrumentos orientados a estudios epidemiológicos sensibles a múltiples culturas [Luis et al., 2003]. Análogamente, otros trabajos indican que puede ser necesario una combinación de medidas de funciones cognitivas tales como tests neuropsicológicos, biomarcadores o neuroimágenes puedan ser necesarios para mejorar la precisión diagnóstica [Petersen, 2004] y poder distinguir entre el envejecimiento normal y el DCL, y entre el DCL y la demencia.

Es por todo ello que la búsqueda de nuevos y alternativos métodos de ayuda al diagnóstico del DCL y su uso en todos los ámbitos de la salud, fundamentalmente en atención primaria, representa un reto en el campo. Sus hallazgos resultan cruciales debido a que permitirán mejorar el diagnóstico temprano de la EA y de otras demencias, y diseñar las estrategias terapéuticas necesarias y el tratamiento para estabilizar y retrasar la evolución de la enfermedad.

Considerando las dificultades comentadas para la caracterización y diagnóstico del DCL, el carácter no-lineal y no-estable de este constructo sindrómico y su alto grado de solapamiento entre el envejecimiento normal y el DCL, y entre el DCL y la demencia en su estadio más temprano, en especial la EA, se propone en este trabajo una aproximación computacional neuronal de ayuda al diagnóstico del DCL, y un punto de partida para continuar con el diagnóstico diferencial entre el DCL y la EA o inclusive, abordar la clasificación entre las diferentes demencias y el DCL.

1.2. Estado del arte de los sistemas de detección del DCL

Debido a la dificultad en la detección del DCL utilizando métodos clínicos se hace necesaria la búsqueda de métodos complementarios basados en técnicas computacionales. Concretamente se han desarrollado métodos y sistemas que utilizan esencialmente datos cuantitativos como señales, imágenes, instrumentos de valoración y biomarcadores. Las arquitecturas neuronales más utilizadas en estos desarrollos están dentro del paradigma supervisado, siendo las redes *Backpropagation* una de las arquitecturas más utilizadas. Así también las máquinas de soporte vectorial (SVM) se sitúan como uno de los paradigmas de gran auge en la clasificación de esta patología. Entre otros motivos, esto puede ser debido a su amplia distribución en programas comerciales. Hay que indicar que la inmensa mayoría de los trabajos hasta la fecha versan sobre la detección de la EA, y es por ello que los desarrollos de sistemas computacionales para la detección del DCL reseñados en este estado del arte son bastante recientes. Así, entre los trabajos más destacables podemos citar los siguientes:

Dentro de los trabajos relacionados con el uso de técnicas de imagen podemos encontrar

[Fritzsche et al., 2008], que describe un método computacional para la estimación de cambios atróficos en el cerebro tanto para detectar EA como DCL a partir de imágenes de resonancia magnética (RM) utilizando dos enfoques de clasificación; el discriminante lineal de Fisher (FLD) y redes *backpropagation* (BPN). En [Fritzsche et al., 2008] se llevaron a cabo dos experimentos: el primero de ellos consistió en una clasificación entre sujetos con EA y sujetos control, llegando a clasificar correctamente un 80 % de sujetos con EA y un 85 % de sujetos control, utilizando el clasificador FLD. Para el segundo experimento se realizó una clasificación entre sujetos con DCL y sujetos control, obteniéndose como mejor clasificador un FLD que proporcionó una sensibilidad del 81 % y una especificidad del 80 %. Otros trabajos [Filipovych and Davatzikos, 2011] [Li et al., 2012] [Wee et al., 2012] [Jaramillo et al., 2012] [O'Dwyer et al., 2012] [Cui et al., 2012] también utilizaron imágenes RM como datos de entrada para obtener las características más discriminativas que representaron los patrones del DCL para entrenar SVMs. Otras técnicas de imágenes como la Tomografía Computarizada por Emisión de Fotones (SPECT) y la Tomografía por Emisión de Positrones (PET) han sido utilizadas para la detección del DCL usando técnicas de discriminación lineal (LDA) y SVM [Habert et al., 2011] [Duara et al., 2012].

En base a otra fuente de datos de entrada, como señales de electroencefalogramas (EEG), encontramos en [Buscema et al., 2007] [Rossini et al., 2008] que utilizan la metodología *implicit function as squashing time* (IFAST) basada en redes neuronales artificiales, capaz de comprimir registros electroencefalográficos de ojos cerrados de pacientes en reposo en distribuciones de voltaje instantáneas, para una clasificación entre sujetos con DCL y sanos y entre sujetos con DCL y sujetos con EA.

Otros estudios han combinado varios tipos de datos de entrada, como en [Zhang et al., 2011], donde se propone un método de clasificación multimodal, basado en la combinación de tres biomarcadores: RM, PET y líquido cefalorraquídeo (CSF), para discriminar entre sujetos con EA y sanos, y entre sujetos con DCL y sanos, utilizando SVM. En este estudio, distinguiendo entre sujetos con EA y sanos, utilizando este esquema de combinación de biomarcadores, se obtuvo una precisión de 93 %, mientras que discriminando entre sujetos con DCL y sanos la precisión resultante fue de 76.4 %.

En relación al uso de tests neuropsicológicos, en [Quintana et al., 2012] utilizan como entrada un subconjunto de ítems del test Barcelona abreviado (a-BT) para clasificar pacientes con DCL y EA, comparando el análisis discriminante lineal (LDA) con una red *Backpropagation*, llegando a alcanzar, utilizando esta última, un 98.33 % de precisión clasificando sujetos con DCL y sanos, frente a un 80 % de acierto proporcionado por el clasificador LDA. En [Lv et al., 2010] combinan el test de red de atención (ANT) junto a información demográfica del paciente (edad, género, y nivel educativo) para clasificar utilizando SVM. [García Báez et al., 2008] emplea una colección de cinco test neuropsicológicos para diagnosticar entre cuatro escalas de severidad de deterioro cognitivo, utilizando un ensemble de clasificadores neuronales basados en Counterpropagation, alcanzando unas tasas de sensibilidad y especificidad del 76 y 97 %, respectivamente, para el nivel de severidad del deterioro cognitivo más leve. En [Báez, 2005] se describe una arquitectura neuronal no supervisada, la arquitectura HUMANN (del inglés *Hierarchical Unsupervised Modular Artificial Neural Network*) utilizada en diversos trabajos para asistir, además, el diagnóstico diferencial de demencias empleando test neuropsicológicos [Báez et al., 2009] [Araujo et al., 2010] [García Báez et al., 2011].

1.3. Computación neuronal

Las redes neuronales artificiales tuvieron su origen en la confluencia de tres temas relacionados como, la teoría de la información de Shannon, la identificación de los sistemas que tienden a un fin y sistemas realimentados, y la propia teoría de redes neuronales de McCulloch-Pitts, los cuales confluyeron en el estudio de los procesos volitivos y cognitivos del hombre estableciendo una mecanización de procesos mentales con fines dirigidos a un mayor entendimiento del cerebro y de la resolución de problemas técnicos concretos. Fueron Warren McCulloch y Walter Pitts quienes en 1943 presentaron el primer modelo de neurona artificial, *la neurona formal*, inspirados en la neurona biológica. Ellos así dieron un giro significativo a los enfoques computacionales con su teoría de redes de neuronas formales.

McCulloch era acusado por De Wortis, en la introducción del libro “The Logical Structure of Mind”, de estar siempre intentando escribir una ecuación para el cerebro. Es esta una acusación que define, en pocas palabras, el objetivo último de la computación neuronal. La computación neuronal es distribuida y paralela, adaptativa y autoprogramable, constituyendo la primera alternativa a la programación programada. Su principal estructura de procesamiento de información son las redes neuronales artificiales (RNAs) formadas por múltiples elementos de computación altamente interconectados entre sí, generando arquitecturas modulares tridimensionales, que pueden ser modulares, como sistemas de computación neuronal (figura 1.4). Estos elementos de proceso o neuronas artificiales, tienen su inspiración en la neurona biológica o célula nerviosa. Esta constituye el elemento de construcción básico de las redes neuronales biológicas (RNB). Un cerebro humano contiene alrededor de 10^{11} células electrónicamente activas, denominadas *neuronas*. Éstas existen en una amplia variedad de formas, aunque la mayoría tienen las características comunes indicadas en la figura 1.1.

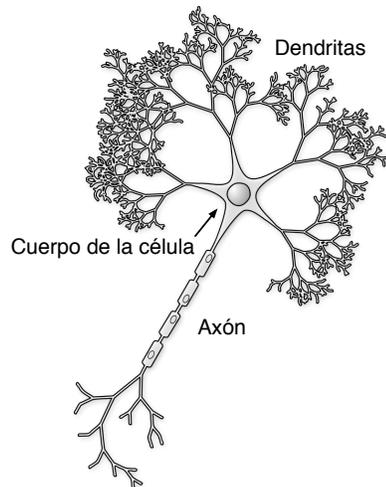


Figura 1.1: Representación de una neurona biológica.

La neurona posee una estructura básica, denominada *cuerpo de la célula* o *soma*. El árbol ramificado de las *dendritas* proporciona un conjunto de entradas a la neurona, mientras que el *axón* actúa como un elemento de salida. Las comunicaciones entre neuronas tienen lugar en las

sinapsis. Cada neurona es una estructura muy compleja que procesa las señales de entrada de muchas formas diferentes. No obstante, las neuronas son relativamente lentas cuando las comparamos con las puertas lógicas electrónicas. Estas pueden lograr tiempos de conmutación del orden de nanosegundos, mientras que las neuronas necesitan varios milisegundos para reaccionar a un estímulo.

Las neuronas artificiales solo recogen algunas de las propiedades estructurales y dinámicas de estas neuronas biológicas. La figura 1.2 muestra la estructura de una neurona abstracta con n entradas. Cada canal de entrada i puede transmitir un valor real x_i . La función primitiva f calculada en el cuerpo de la neurona abstracta puede ser seleccionada arbitrariamente. Normalmente los canales de entrada tienen un peso asociado, que significa que la información de entrada x_i es multiplicada por el peso correspondiente w_i . La información transmitida es integrada en la neurona (normalmente sólo por la adición de diferentes señales) y la función primitiva es entonces evaluada.

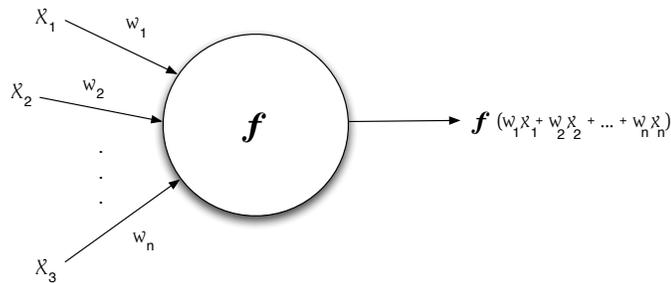


Figura 1.2: Representación de una neurona abstracta.

Si concebimos cada nodo de una RNA como una función primitiva capaz de transformar sus entradas en una salida definida de forma precisa, entonces las RNAs no son más que redes de funciones primitivas. Diferentes modelos de redes neuronales artificiales difieren principalmente en las asunciones acerca de las funciones primitivas usadas, el patrón de interconexión y el tiempo de transmisión de la información.

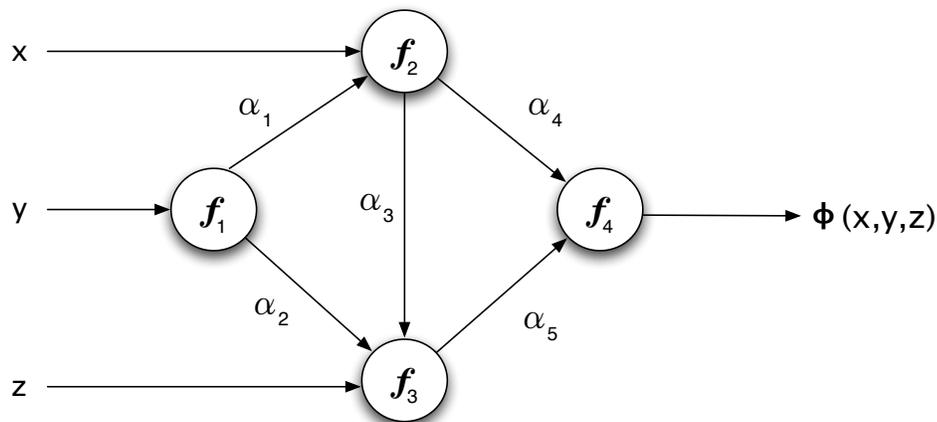


Figura 1.3: Modelo funcional de una ANN.

Las RNAs típicas tienen la estructura del tipo mostrado en la figura 1.3. La red puede ser pensada como una función ϕ que es evaluada en el punto (x,y,z) . Los nodos implementan las funciones primitivas f_1, f_2, f_3, f_4 que son combinadas para producir ϕ . Esta función implementado por una red neuronal será denominada función de la red. Las diferentes selecciones de los pesos $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ produce diferentes funciones de red.

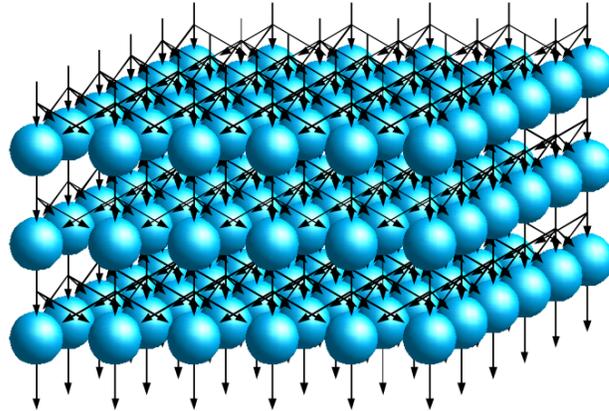


Figura 1.4: Arquitectura tridimensional de una red de neuronas artificiales.

Las RNAs constituyen un paradigma de computación inspirado en las capacidades de procesamiento de la información de los sistemas nerviosos biológicos [Rojas, 1996]. Las RNAs pueden ser definidas como estructuras cognitivas de procesamiento de información basadas en modelos de función del cerebro. Sus reglas surgen como propiedades emergentes y no como estructuras simbólicas. Su propiedad más importante es la capacidad de aprendizaje que poseen, la cual se materializa en la modificación adaptativa de sus pesos sinápticos. Este tipo de mecanismo responde al hecho establecido a nivel biológico, de que el conocimiento reside en las *sinapsis* de la red neuronal biológica. Su comportamiento emerge de cambios estructurales dirigidos por reglas locales de aprendizaje. Alcanzan sus resultados después de procesar el conjunto de entradas que llegan a ellas por sus conexiones aferentes, a través de procesos de auto-organización. Son altamente fiables por el uso de la redundancia, y tolerantes a fallos, tratan cualquier tipo de información: desde ruidosa a fuzzy, y son de un procesado muy rápido debido al paralelismo.

1.3.1. Redes Neuronales Artificiales supervisadas

Las redes supervisadas implementan un tipo de aprendizaje que se realiza mediante un entrenamiento controlado por un agente externo (supervisor) que determina la respuesta que debería generar la red a partir de una entrada determinada.

Perceptrón simple

El modelo del perceptrón [Rosenblatt, 1961] refleja los comienzos del reconocimiento de patrones automático. Consiste en una arquitectura de una sola capa de pesos modificable, con una capa de entrada, que está compuesta de unos sensores de ajuste, y una capa de salida, con respuestas

enviadas por la red. Su neurodinámica está compuesta de una función de red que utiliza la clásica suma de las entradas (ecuación 1.1), y un escalón, bipolar o binario (ecuación 1.2), para la función de activación, con un punto de transición que está determinado por un valor de umbral θ_i almacenado localmente en la neurona.

$$net_i(x) = \sum_j w_{ij}x_j \quad (1.1)$$

$$y_i = \begin{cases} 0 & \text{if } net < \theta_i \\ 1 & \text{if } net \geq \theta_i \end{cases} \quad (1.2)$$

El modelo de aprendizaje que sigue es capaz de adaptar sus pesos y umbrales por medio de un paradigma supervisado utilizando la *regla del perceptrón*, basada en la corrección del error producido en la capa de salida:

$$\Delta w_{ij} = \alpha(x_i - d_i)x_j \quad , \quad (1.3)$$

donde α es el radio de aprendizaje y d_i la salida deseada.

Una de las ventajas de este modelo es que utiliza el *Teorema de Convergencia del Perceptrón* que garantiza la convergencia del aprendizaje en un tiempo finito y que la arquitectura siempre permite representar la solución. Precisamente, es en su capacidad de representación donde reside la gran limitación del modelo. En [Minsky and Papert, 1969] se discute su incapacidad de resolver problemas no separables linealmente, por ejemplo el caso del problema del OR-exclusivo (XOR).

Perceptrón multicapa

Una forma de superar las limitaciones de representación mencionadas en el perceptrón simple es a través de los perceptrones multi-capas (MLP). Un MLP constituye una topología con una o varias capas ocultas, y conexiones alimentadas hacia adelante entre sus capas sucesivas, tanto de una forma parcial o total. Con el objetivo de representar cualquier función *booleana* es necesario que algunas de las neuronas utilicen funciones de activación no lineales (función de umbralizado o función sigmoide, ecuación (1.4), manteniendo el resto de la neurodinámica igual que para el perceptrón simple.

$$f_{act}(net, \theta) = (1 + e^{-net+\theta})^{-1} \quad (1.4)$$

El algoritmo más popular para entrenar un MLP es el *backpropagation* o algoritmo de *retropropagación* [Werbos, 1974]. Está basado en una corrección supervisada del error cuadrático generado en la capa de salida utilizando un método de descenso del gradiente. Este método obliga a la función de activación utilizada ser diferenciable y monótona. Comienza con la capa de salida y ajusta los pesos de las conexiones que la afectan, produciendo una retropropagación de los errores de las capas previas que sucesivamente corrigen los pesos hasta alcanzar la primera capa oculta.

$$\Delta w_{ij} = a\delta_i x_j \quad (1.5)$$

$$\delta_i = \begin{cases} f'_{act}(net_i, \theta_i)(d_i - x_i) & \text{if } i \in Output \\ f'_{act}(net_i, \theta_i) \sum_k \delta_k w_{ki} & \text{otherwise} \end{cases} \quad (1.6)$$

Algunos de los problemas dignos de mención son estudiados en *backpropagation*, desde que el método de descenso del gradiente no asegura alcanzar el error mínimo global, justo al contrario que el perceptrón simple [Minsky and Papert, 1969]. Para solventar este obstáculo, se han propuesto muchas variantes, tales como generalización, velocidad de aprendizaje y tolerancia a fallos.

1.3.2. Redes Neuronales Artificiales no supervisadas

Las redes de aprendizaje no supervisado (auto-supervisado) no requieren influencia externa para ajustar los pesos de las conexiones entre sus neuronas. La red no recibe ninguna información por parte del entorno que le indique si la salida generada en respuesta a una determinada entre es o no correcta; por ello suele decirse que estas redes son capaces de auto organizarse.

Mapas de Kohonen

Los mapas auto-organizados (SOM) [Kohonen, 1989] [Kohonen et al., 2001] describen la idea de que los mapas topográficos, al igual que los que existen en la corteza de los cerebros de los animales altamente desarrollados, extraen las características del espacio de entrada preservando su topología. Por una parte, estos combinan las características de los sistemas competitivos, cuantificando el espacio de entrada en diferentes regiones representadas por un número específico de neuronas de salida. Por otra parte, mantienen una relación de vecindad entre las unidades del espacio de salida, es decir, dos neuronas vecinas representan regiones cercanas en el espacio de entrada. Por tanto, estos métodos generan un mapa discreto, posiblemente con una dimensionalidad inferior a la del espacio de entrada, y preservan la topología de este primero. Entre los métodos más comúnmente utilizados, podemos encontrar las SOMs de Kohonen, aunque existen una gran cantidad de variantes (ART [Grossberg, 1976], Sanger [Sanger, 1989], HUMANN [Báez, 2005], etcétera). Éstas exhiben una topología similar al Aprendizaje Competitivo Simple, pero la capa de salida es organizada en un espacio m -dimensional de acuerdo a la forma que se desee del mapa, y lo más común es una matriz de dos dimensiones.

La neurodinámica que siguen los SOMs de Kohonen acepta las mismas posibilidades que las del Aprendizaje Competitivo Simple. También siguen un paradigma no supervisado y un entrenamiento competitivo. Las principales variaciones están relacionadas con las modificaciones de los pesos sinápticos (ecuación 1.7), que no sólo afectan a la neurona ganadora sino también en un grado inferior al conjunto de neuronas de la vecindad de la ganadora, N , siendo capaz de generar relaciones topológicas. Las relaciones de vecindad entre los nodos vienen dadas normalmente por una rejilla bidimensional de tipo cuadrado o hexagonal, aunque puede ser de cualquier otra forma geométrica, cuyo tamaño decrece durante el período de entrenamiento.

$$\Delta w_{ij} = \begin{cases} \alpha(x_i - w_{ij}) & \text{if } i \in N(\arg \max_k (net_k)) \\ 0 & \text{otherwise} \end{cases} \quad (1.7)$$

Los posibles usos de los mapas SOMs de Kohonen van desde la proyección de datos multivariados hasta la aproximación de densidades en agrupaciones, habiendo sido aplicado en multitud de campos [Kohonen et al., 2001].

Red de Sanger

T. Sanger [Sanger, 1989] propuso el Algoritmo Hebbiano Generalizado (GHA), un método multi-componente igual que la regla de Oja para la primera componente, pero también capaz de extraer el resto de los autovectores de longitud unitaria. El modelo posee m neuronas de salida (y_1, \dots, y_m) y n entradas (x_1, \dots, x_n). La red de Sanger está completamente interconectada con las entradas (figura 1.5); es una red de procesamiento donde la i -ésima unidad tiene asociada la y -ésima salida que es dada por la siguiente expresión:

$$y_i = \sum_{j=1}^n w_{ij} x_j \quad , \quad (1.8)$$

donde n es la dimensionalidad del vector de entrada, x_j , es el valor de la j -ésima entrada y w_{ij} es el peso entre la j -ésima entrada y la i -ésima unidad. La regla de aprendizaje que progresivamente ajusta la matriz de pesos es dada por la siguiente expresión:

$$\Delta w_{ij} = \eta(t) y_i (x_j - \sum_{k=1}^i y_k w_{kj}) \quad (1.9)$$

En la ecuación 1.9 el radio de aprendizaje $\eta(t)$ es linealmente decreciente en función del tiempo t . Esta fórmula hace que los pesos de Sanger converjan hacia las componentes principales del conjunto de datos de entrenamiento.

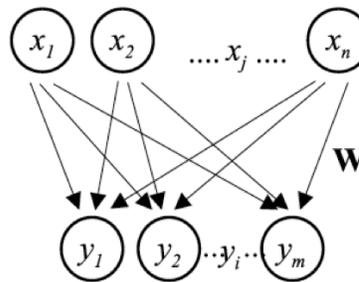


Figura 1.5: Estructura de la red neuronal de Sanger.

El GHA permite la extracción de características de un conjunto de datos por medio de una transformación del espacio de entrada en un nuevo espacio de características con una dimensionalidad inferior (concretamente el espacio de las componentes principales, PCA). Una vez realizada la transformación a este espacio, se obtienen los *scores* (datos de entrada proyectados en el espacio de características) y los pesos o *loads* (coeficientes de correlación entre las cargas y el conjunto de datos de entrada).

Entre las utilidades del GHA se encuentran la de comprimir información sin demasiada pérdida (transformación de Hotelling o Karhunen and Leove [Diamantaras and Kung, 1996]), identificar patrones estadísticos en las distribuciones de datos y visualizar conjuntos de datos de alta dimensionalidad.

1.3.3. Redes Neuronales Artificiales híbridas

Las redes híbridas son redes combinadas que acoplan una capa auto-organizada con otra supervisada.

Red Counterpropagation

La red counterpropagation (CPN) constituye un modelo de red neuronal híbrida propuesto en 1987 por Hech-Nielsen [Hecht-Nielsen, 1987]. En cierta forma, las CPNs son una extensión de la aproximación de Kohonen [Kohonen et al., 2001]. Las redes CPNs son estructuras combinadas que, haciendo un símil con los sistemas naturales, también consiste en una jerarquía de redes, cada una hecha de elementos especializados en diferentes tareas. En general las redes combinadas son más poderosas que las redes monolíticas. La CPN además de poseer una arquitectura modular, es una red híbrida. Las redes híbridas son redes combinadas que combinan una capa auto-organizativa con otra retro-alimentada.

La CPN es una red neuronal modular compuesta de dos capas en cascada con aprendizajes independientes, una capa competitiva o capa de Kohonen y una capa de aprendizaje *outstar* o capa de Grossberg (figura 1.6).

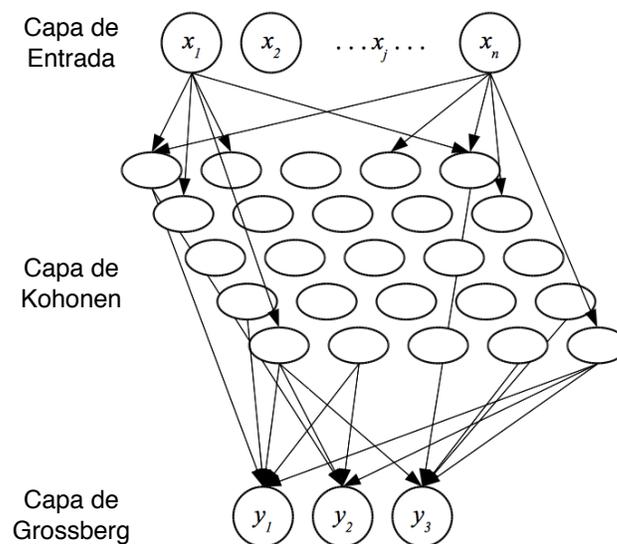


Figura 1.6: Estructura de la red neuronal híbrida Counter-Propagation.

La neurodinámica seguida por la SOM de Kohonen acepta la misma posibilidad que la del Aprendizaje Competitivo Simple. La ecuación 1.10 describe la modificación de los pesos sinápticos que no sólo afecta a la neurona ganadora sino también al conjunto de neuronas en la vecindad ganadora, N , en un menor grado, siendo capaz de generar relaciones topológicas. La relación de vecindad entre los nodos es normalmente dada por una rejilla bidimensional de tipo cuadrado o hexagonal, cuyo tamaño se decrementa durante el periodo de entrenamiento.

$$\Delta w_{ij} = \begin{cases} \alpha(x_i - w_{ij}) & \text{if } i \in N(\arg \max_k (net_k)) \\ 0 & \text{otherwise} \end{cases} \quad (1.10)$$

La segunda fase de la CPN implementa el aprendizaje de la regla delta, recibiendo un único valor de 1 desde la etapa previa. Esta segunda capa (normalmente denominada capa de Grossberg) es equivalente a un aprendizaje *outstar* [Freeman and Skapura, 1993]. El descenso por gradiente sobre esta función de coste proporciona la actualización de pesos necesaria:

$$\Delta z_i = -\frac{dE}{dz_i} = \gamma(f(x) - z_i) \quad , \quad (1.11)$$

donde γ es la constante de aprendizaje. Tras varias iteraciones de aprendizaje supervisado, se espera encontrar una buena aproximación de la función f . El entrenamiento de las capas intermedia y de salida puede ser entre-mezclado o en secuencia.

Según lo anteriormente descrito, el aprendizaje de la red se divide en dos fases diferentes. En primer lugar, la capa de Kohonen es entrenada utilizando de manera estocástica vectores seleccionados del espacio de entrada. Esta capa produce una agrupación (*clustering*) del espacio de entrada que corresponde a un diagrama de Voronoi de n -dimensiones y lo mapea a un espacio de inferior dimensionalidad, generalmente de dos dimensiones. Después de esta fase, cada elemento de esta capa se ha especializado en reaccionar a una cierta región del espacio de entrada. La salida de esta capa puede estar controlada de forma que sólo dispara el elemento con la activación más alta. Una vez encontrada la neurona ganadora (o central) entre el conjunto de neuronas de la capa de Kohonen, se excitará una neurona a un elemento de la capa de Grossberg. Los pesos de ambas capas (capa de Kohonen y capa de Grossberg) son ajustados acorde a los pares de vectores de entrada y objetivo utilizando adecuadamente una tasa de aprendizaje seleccionada y una función de vecindad.

En resumen, una vez entrenada una CPN, los pasos que sigue la red para aproximar un vector de entrada \mathbf{v} , son los siguientes:

1. Se normaliza el vector de entrada:

$$x_i = \frac{v_i}{\sqrt{\sum_n v_i^2}} \quad (1.12)$$

2. Dado que el vector de entrada ya está normalizado, la capa de entrada sólo lo distribuye a las unidades de la capa competitiva

Capítulo 1. Introducción

3. La capa competitiva es del tipo “el ganador se lo lleva todo”. La unidad cuyo vector de peso se asemeje más al vector de entrada será la ganadora, y tendrá un valor de salida igual a 1. Todas las demás unidades tendrán salida nula. La salida de cada unidad se puede calcular de la siguiente manera:

$$\Delta z_i = \begin{cases} 1 & \|neta_i\| > \|neta_j\| \forall j \neq i \\ 0 & otherwise \end{cases} \quad (1.13)$$

4. El ganador único de la capa competitiva excita a un elemento de la *outstar*. Cada unidad de la *outstar* alcanza rápidamente un valor de equilibrio igual al valor del peso que haya en la conexión procedente de la unidad ganadora de la capa competitiva. Si la i -ésima unidad gana en la capa intermedia, entonces la capa de salida produce un vector de salida $\mathbf{y}' = (w_{1i}, w_{2i}, \dots, w_{mi})$, en donde m representa el número de unidades que haya en la capa de salida.

Una de las ventajas más significativas de esta red es su rapidez en tiempo de cómputo. Los entrenamientos suelen ser entre 10 y 100 veces más rápidos que los de las convencionales redes *Backpropagation*, produciendo resultados similares. El incremento de esta velocidad (o reducción de tiempo de cómputo) es atribuido a la simplificación que ocurre en la etapa auto-organizada. La simplificación también permite que la segunda fase emplee un clasificador simple que produce una probada convergencia en problemas de carácter lineal, mejor capacidad de generalización y una reducción de los recursos de cómputo.

Capítulo 2

Método de sobre-muestreo extendido basado en la red neuronal de Sanger (SNEOM)

En este capítulo se presenta un nuevo método de muestreo de datos basado en la red neuronal de Sanger. En primer lugar, se centrará el problema de las clases no balanceadas y se describirán las posibles aproximaciones y técnicas que existen actualmente para abordarlo. A posteriori, se introducen los sistemas auto-organizados, y dentro de ellos los métodos proyectivos, a los cuáles pertenece la red neuronal de Sanger, y por ende el método de muestreo propuesto. Tras estas secciones se dará paso a la presentación del método en sí, finalizando el capítulo con la validación del mismo utilizando diferentes conjuntos de datos biomédicos provenientes de bases de datos propias y repositorios públicos.

2.1. Distribuciones de datos no balanceadas: Un problema de los sistemas de aprendizaje clásicos

El problema de *clases no balanceadas* corresponde a un problema propio de los sistemas de aprendizaje inductivo en los que una clase es representada por un amplio número de muestras mientras que la otra está representada sólo por unas pocas [Japkowicz and Stephen, 2002]. Este problema es de crucial importancia debido a su existencia en múltiples ámbitos como el medioambiental, el financiero, el médico o el comercial y ha demostrado ser un cuello de botella importante en el rendimiento obtenido por los métodos de aprendizaje en general, los cuales asumen una distribución de clases balanceada. Algunas aplicaciones específicas que presentan este problema son la detección de vertidos de petróleo en imágenes radar de satélites, la detección de llamadas de teléfono fraudulentas, la monitorización de fallos en la caja de cambios de helicópteros en vuelo, el filtrado y recuperación de información, el diagnóstico en condiciones médicas extrañas.

Para abordar el problema de *clases no balanceadas* existen dos enfoques:

- **Desde el punto de vista de los datos.** Abordan el problema reestructurando el conjunto

de datos no balanceado y en algunos casos lo modifican. En este último supuesto la modificación se hace añadiendo nuevas muestras al conjunto de elementos de la clase minoritaria C_{MI} , o eliminando muestras de la clase mayoritaria C_{MA} . Se definen pues dos técnicas dentro de esta aproximación:

- **Comité de multi-clasificadores ó *multi-classifier committee*:** En este método se emplean todos los datos contenidos en el conjunto de datos de entrada. En primer lugar, se dividen las muestras de C_{MA} en diferentes subconjuntos y luego se conforman varios conjuntos de entrenamiento con cada subconjunto previo y todas las muestras de C_{MI} . Después de entrenar a diferentes clasificadores o al mismo clasificador con diferentes conjuntos de entrenamiento, se combinan todos los resultados a través de un sistema de votación mayoritario.
 - **Re-muestreo ó *re-sampling*:** Consiste en el añadido o eliminación de muestras del conjunto no balanceado. Dependiendo de la acción a realizar, las técnicas a emplear pueden ser de *over-sampling* o de *under-sampling*, respectivamente. La primera de ellas, la aproximación *over-sampling*, incrementa el número de muestras de C_{MI} para reducir el grado de desequilibrio de la distribución. Un método que implementa esta técnica es SMOTE [Chawla et al., 2002]. SMOTE produce muestras sintéticas de C_{MI} seleccionando algunos vecinos minoritarios más cercanos de una muestra, S , y genera nuevas muestras de C_{MI} a lo largo de la línea entre S y cada vecino minoritario más cercano. Por otra parte, la aproximación *under-sampling* permite reducir el número de muestras de C_{MA} . Existen diferentes métodos basados en esta aproximación. El más simple es el denominado aproximación de sub-muestreo aleatorio ó *random under-sampling approach*, que consiste en seleccionar un subconjunto de C_{MA} aleatoriamente y luego combinarlo con C_{MI} para obtener el conjunto final de entrenamiento. Existen otras aproximaciones más sofisticadas basadas en distancias (*under-sampling approach based on distance*) para seleccionar muestras más representativas. Este método emplea distintos criterios: el más cercano, el más lejano, la media con respecto al más cercano y la media con respecto al más lejano entre C_{MI} y C_{MA} . El principal inconveniente de los métodos basados en distancias es que, para conjuntos de datos amplios, emplean una gran cantidad de tiempo en la selección de las muestras de C_{MA} , por lo que no son eficientes en aplicaciones reales.
- **Desde el punto de vista del algoritmo de aprendizaje.** Con respecto a este enfoque, podemos distinguir dos aproximaciones:
- **Aprendizaje sensible al coste ó *cost-sensitive learning*:** Esta aproximación supone que los costes por errar en la clasificación son conocidos. Un clasificador basado en este paradigma, intenta aprender más características de las muestras de C_{MI} estableciendo un alto coste al fallo en la clasificación de una muestra de esta clase. Sin embargo, a veces son desconocidos los costes por fallo en la clasificación y un clasificador *cost-sensitive* puede resultar en un entrenamiento sobre-especializado. Un método que sigue esta aproximación es Metacost [Domingos, 1999]. En él, se estima la probabilidad de

cada clase por cada muestra y estas son re-etiquetadas de forma óptima con respecto a los costes de clasificar erróneamente. El re-etiquetado de las muestras expande el espacio de decisión ya que crea nuevas muestras a partir de las cuáles el clasificador puede aprender [Domingos, 1999].

- Aprendizaje basado en reconocimiento ó *recognition-based learning*: Esta técnica intenta sobre-especializar al clasificador mediante el aprendizaje de una sola clase (la clase minoritaria, C_{MI}). El aprendizaje de una sola clase es más adecuado que las aproximación de dos clases en datos muy desequilibrados y en espacios de características ruidosos de alta dimensionalidad [Elkan, 2001].

2.2. Sistemas auto-organizados

Los sistemas auto-organizados son aquellos que poseen procesos que permiten aumentar automáticamente su organización interna, sin ser guiados o controlados por una fuente externa [Ross, 1953] [Foerster and Zopf, 1962]. Normalmente, estos sistemas muestran propiedades emergentes, que permiten a partir de reglas simples formar estructuras complejas [Holland, 1998]. El concepto de auto-organización es primordial en la descripción de sistemas biológicos, desde el nivel subcelular al nivel de ecosistema. Por otro lado, desde el punto de vista matemático e informático, ramas tales como la cibernética, los autómatas celulares, los grafos aleatorios, la computación evolutiva y la vida artificial, presentan características de auto-organización [Heylighen, 1992].

En el aprendizaje auto-organizado, también llamado aprendizaje no supervisado, a diferencia del aprendizaje supervisado, no se hace uso de información externa, es decir, no contempla la existencia de una realimentación del entorno que indique cuáles deben ser las salidas a una información de entrada dada, o si las salidas actuales son correctas o no. Esto en la práctica se traduce, en que no hacen uso de un conjunto de patrones proporcionados por un tutor externo, que indica el mapeo de entrada-salida que se desea lograr, por tanto, carece de un conjunto de salidas deseadas. El propósito de este tipo de aprendizaje es descubrir patrones, familiaridades, características significativas, regularidades, correlaciones o categorías dentro de sus datos de entrada y realizar dichos descubrimientos sin un tutor, basándose en la observación y en la redundancia en la información [Haykin,] [J. Hertz, 1991].

Para poder realizar los descubrimientos citados anteriormente, los sistemas auto-organizados hacen uso de reglas de naturaleza local, así por ejemplo, los cambios que sufren los pesos sinápticos de una neurona están influenciados por el efecto de la vecindad inmediata de dicha neurona. Este aspecto no tiene por qué ser una limitación en sí mismo, ya que tal como indicó Alan Mathison Turing [Turing, 1952], el orden global puede surgir de interacciones locales. Incluso muchas interacciones locales, inicialmente aleatorias, entre neuronas vecinas de una red pueden fundirse en estados de orden global que finalmente lleven a una conducta coherente, lo que es en esencia la auto-organización. Esta observación es aplicable tanto al cerebro como a redes neuronales artificiales, de hecho, los esquemas neuronales usados en aprendizajes auto-organizados, tienden a seguir las estructuras neurobiológicas mucho más extensamente que los utilizados en sistemas con aprendizajes supervisados. Un ejemplo de sistema auto-organizado es el juego de la vida propuesto por John Horton Conway [Gardner, 1970]. En éste, los elementos de un autómata celular interac-

Capítulo 2. Método de sobre-muestreo extendido basado en la red neuronal de Sanger (SNEOM)

cionan con sus vecinos adyacentes para definir su estado inmediatamente posterior, que puede ser *sobrevivir, nacer o morir*, produciendo diferentes estructuras a medida que evoluciona el juego que se repiten con cierta frecuencia.

Von der Malsburg propone tres principios básicos que se dan en los sistemas auto-organizados [der Malsburg, 1990]:

1. Las modificaciones en los pesos sinápticos tienden a auto-amplificarse: al revés que en los sistemas de estabilización, los bucles de realimentación entre los pesos sinápticos y los patrones de actividad han de ser positivos para producir la auto-organización. De hecho, este postulado puede verse como una reescritura de la *Ley de aprendizaje de Hebb* básica [Hebb, 1949]. Dicha capacidad de auto-amplificación está limitada debido a la acotación de los propios recursos que la alimentan: la localidad de las señales disponibles y los recursos energéticos.
2. La limitación de los recursos conduce a la competición entre sinapsis y por tanto a la selección de la sinapsis más fuerte a expensas de las restantes: debido a lo limitado de los recursos, la manera de fortalecer una sinapsis debe ser compensada por el decrecimiento o desaparición de otras y mantener de esta forma un equilibrio global gracias a la plasticidad sináptica. En este postulado podemos ver las ideas conducentes a la *ley de aprendizaje competitivo* [der Malsburg, 1973].
3. Las modificaciones en los pesos sinápticos tienden a cooperar: una sola sinapsis por sí misma no puede producir, eficientemente, eventos favorables. Para hacerlo necesita la cooperación entre un grupo de sinapsis coincidentes en una misma neurona que produzca señales coincidentes con suficiente fuerza para activar dicha neurona. De esta forma, la presencia de una sinapsis fuerte puede, en lugar de competir, facilitar el crecimiento de otras.

Los principios anteriores hablan de la red en sí misma, sin embargo el aprendizaje no supervisado únicamente puede llegar a resultados útiles cuando hay redundancia en los datos de entrada, de esta manera, tal como afirma Barlow [Barlow, 1989], la redundancia genera conocimiento. Más formalmente hablando, la información real contenida en el canal de entrada debe de ser menor que la máxima información que puede transportar dicho canal.

Las arquitecturas de los sistemas auto-organizados pueden tener gran variedad de formas, lo más común suele ser una capa de entrada, que distribuye sus datos sobre una capa de salida, que hace las funciones de capa de representación, y donde hay conexiones hacia adelante desde la entrada a la salida e interconexiones laterales entre las neuronas de la capa de salida. Otras veces consisten en redes multicapa con conexiones hacia adelante entre capas consecutivas, donde en cada una de las capas se producen procesos auto-organizativos. También es común, excepto en el caso de mapas de características, que el número de unidades de salida sea mucho menos que las de entrada [Carpenter and Grossberg, 1991].

El hecho de utilizar reglas de aprendizaje local, junto a que no están restringidos a tener que adaptarse a un mapeo determinado, suele producir que los procesos de aprendizaje sean mucho más rápidos y efectivos que en modelos supervisados. Es más, muchas veces se combinan en una misma red varias capas, unas con modelos de aprendizaje no supervisados y otras con supervisados,

generando redes neuronales modulares e híbridas que mejoran la eficacia de otros sistemas más simples [Hrycej, 1992].

Muchos modelos auto-organizados se basan simplemente en ideas intuitivas que en la práctica proporcionan buenos resultados, no obstante, en algunos casos se puede definir formalmente cantidades, tales como cantidad de información, varianza de la salida o funciones de coste, que ha de ser maximizada o minimizada. Estos modelos basados en optimización están más cercanos a aquellos que se utilizan en las ciencias estadísticas, de hecho hay relaciones bastante próximas entre muchos modelos neuronales auto-organizados y otras técnicas de reconocimiento estadístico de patrones [Sarle, 1994].

2.2.1. Funciones y métodos

Entre las funciones que presentan los sistemas no supervisados podemos destacar la visualización, que permite hacer una representación visual de los datos que se manejan. También estos modelos permiten obtener representaciones más eficientes de dichos datos, posiblemente con cierta pérdida de información, pero más aptas para poder almacenarlos, interpretarlos o transmitirlos a través de canales de comunicación. Este aspecto de optimización de la representación también es posible reformarlo para ser utilizado en problemas de optimización combinatoria. Finalmente, una de las funciones más importantes proporcionada por los sistemas auto-organizados, es poder encontrar la forma en que están estructurados los datos, lo que permite realizar y probar diversas hipótesis sobre dichos datos así como generar sistemas de clasificación sobre ellos.

Para poder realizar las funciones anteriores, existen, en mayor o menor medida, diversos métodos dentro del campo del reconocimiento de patrones [Ripley, 1996] [Jain et al., 2000] de los que muchos de ellos pueden ser implementados mediante modelos neuronales. Uno de los más interesantes son los métodos proyectivos, que serán usados en el desarrollo de este trabajo.

2.2.2. Métodos proyectivos

Los métodos proyectivos para la extracción de características generan una transformación del espacio de entrada en un nuevo espacio de características con una dimensionalidad más baja. Habitualmente, dicha transformación se crea en base a combinaciones lineales que tratan de maximizar alguna medida de interés, como el preservar la máxima cantidad de información posible, reduciendo notablemente la dimensionalidad de los datos. Su utilidad se centra en la compresión de datos o codificación óptima y la visualización de datos de elevada dimensionalidad.

Existen modelos de redes neuronales que son capaces de converger a los resultados de métodos clásicos de esta categoría, como el Análisis de las Componentes Principales (PCA, del inglés *Principal Component Analysis*) [Diamantaros and Kung, 1996] y el Análisis de las Componentes Independientes (ICA, del inglés *Independent Component Analysis*) [Oja, 1997]. En concreto, entre las técnicas neuronales cuyos procesos convergen al PCA tenemos la red de Sanger o también llamado Algoritmo Hebbiano Generalizado (GHA, del inglés *Generalized Hebbian Algorithm*), el Extracto de componentes Principales Adaptivo (APEX, del inglés *Adaptive Principal-component Extractor*), el modelo de Foldiak y el modelo de Rubner. También es posible el uso de arquitecturas neuronales autoasociativas provistas de un cuello de botella que codifique las entradas en una

capa oculta, de esta forma las activaciones de las unidades de dicha capa oculta constituirán las características buscadas.

Por otro lado es posible otro tipo de proyecciones denominadas *escalados multidimensionales* basadas en métodos no lineales de extracción de características. Su cometido principal es representar un conjunto de datos multidimensionales dentro de un espacio bi o tri-dimensional, de forma que las distancias entre cualquier par de datos en el espacio inicial sean preservadas con la mayor exactitud posible en el espacio de características. Su utilidad se centra en la visualización de datos multidimensionales, ya que a diferencia de los métodos proyectivos anteriores, no están preparados para reconstruir los datos iniciales a partir de las características obtenidas. El mapeo de Sammon constituye uno de los métodos más utilizados dentro de esta categoría, el cual puede ser implementado neuronalmente [Mao and Jain, 1995].

2.3. SNEOM

2.3.1. Motivación

Los conjuntos de datos no balanceados entrañan un gran problema para los clasificadores debido a la representación desigual de las clases. La mayoría de los algoritmos de aprendizaje no son capaces de tratar con distribuciones no balanceadas. Ellos suelen asumir una distribución balanceada entre las clases, repercutiendo en una significativa reducción en el rendimiento del clasificador. Este inconveniente se encuentra en una multitud de diferentes dominios en los que la colección de muestras positivas (relativas a la clase minoritaria o de interés) es menor que la representación de muestras negativas (clase mayoritaria o popular).

Concretamente, en el campo médico, los conjuntos de datos no balanceados suponen un problema realmente crítico debido a que el coste de clasificar una muestra de la clase minoritaria (p.e. una enfermedad poco frecuente), como perteneciente a la clase mayoritaria (p.e. no padecer esa extraña enfermedad), es especialmente alto. En ese caso, un sujeto que realmente padece la enfermedad podría no estar tratado adecuadamente desde el punto de vista médico.

En algunos escenarios médicos, la existencia de pocas muestras de la clase minoritaria o de interés se debe bien a la complejidad para diagnosticar la enfermedad por los facultativos [Weiss, 2004], o bien por el coste que implica la adquisición de datos necesaria para poder establecer un diagnóstico [Weiss and Tian, 2008]. Además, para alcanzar un buen rendimiento de los clasificadores, esencialmente los neuronales, como es el caso que nos ocupa, es necesario la generación de un conjunto de datos extenso y representativo de las clases que lo componen [Batista et al., 2004].

Es por tanto, necesario disponer de un mecanismo que permita aumentar la representación de las muestras de interés para mejorar la eficacia de los sistemas de aprendizaje utilizados por los sistemas de diagnóstico asistidos por ordenador. De ahí, surge la necesidad de SNEOM (un método de sobre-muestreo basado en la red neuronal de Sanger, en inglés, *Sanger Network based Over-sampling Method*). SNEOM es un método neuronal proyectivo para la generación de datos multivariable. Su principal característica es que el sobre-muestreo se realiza sobre un espacio transformado de los datos de entrada. Con ello puede aplicarse a datos de cualquier dimensionalidad, proporcionando la capacidad de visualización de los mismos.

2.3.2. Diseño y Desarrollo

Cuando un clasificador es entrenado utilizando un conjunto de datos que contiene una representación predominante de una clase con respecto a las demás, o bien, cuando no se dispone de un conjunto de datos lo suficientemente extenso y representativo para generar un conjunto de entrenamiento rico, éste se sobre-especializará en la detección de muestras de la clase mayoritaria, errando aquellas muestras pertenecientes al conjunto minoritario.

En esta sección se describirá SNEOM, un método dirigido a lidiar con problemas de clases no balanceadas desde el punto de vista de los datos, inspirado en las técnicas de sobre-muestreo de datos, también denominadas *over-sampling* en inglés. Este método permite la generación de muestras de las clases minoritarias y mayoritarias de un conjunto de datos no balanceado, a diferencia de los métodos clásicos de *over-sampling* que trabajan únicamente sobre el conjunto de muestras minoritario. De ahí su nombre **extendido**. Esta propuesta está orientada no sólo a aumentar el número de muestras de la clase minoritaria, sino a incrementar la proporción de muestras de ambas clases, con el objetivo de producir un conjunto de datos balanceado más extenso y representativo.

SNEOM es un método neuronal proyectivo para la generación de datos sintéticos multivariantes. La principal característica de este método reside en que el re-muestreo es llevado a cabo sobre el espacio transformado de los datos de entrada. Es por ello que SNEOM puede ser aplicado sobre conjuntos de datos n-dimensionales, proporcionando la visualización de datos de alta dimensionalidad en dos o tres dimensiones. Otra importante característica de este método es la capacidad de manejar conjuntos de datos con ausencia de algunos valores ó *missing*. Esta característica se encuentra incorporada por la arquitectura neuronal que implementa SNEOM, la red de Sanger, específicamente la Red de Sanger Extendida para Tratamiento de Datos Missing [García Báez et al., 2007].

El esquema de SNEOM puede observarse en la figura 2.2, donde se muestran las fases de que consta este método de sobre-muestreo: Proyección, Generación y Reconstrucción-Cuantización.

En la figura 2.1 puede observarse la dinámica del método. En SNEOM se ha utilizado como perturbación un filtro o ruido Gaussiano, ya que nos permite, mediante la parametrización del ancho de la campana Gaussiana, controlar la distancia a la que se generan las muestras sintéticas con respecto a la original. Obtenida la muestra perturbada, se comprueba si está dentro del área de aceptación. En caso de que lo esté, la muestra perturbada será incluida en el conjunto de muestras generadas candidatas. A continuación, se comprueba si se ha alcanzado el factor para esa clase. En caso de que se haya alcanzado dicho valor, el método devolverá el conjunto de muestras candidatas. En caso de que no se hubiese alcanzado el factor para esa clase, se volverá a generar una nueva muestra perturbando la muestra original. Si alguna de las nuevas muestras perturbadas no cae dentro del área de aceptación, se desecharán las muestras contenidas en el conjunto de muestras generadas candidatas y se decrementará el ancho de la campana de Gauss utilizada por el ruido Gaussiano en 0.1 unidades. En caso de que el ancho de la Gaussiana sea igual a 0, el método finalizará sin devolver ninguna muestra generada. En caso de que el ancho de la Gaussiana sea distinto de cero, se volverá al proceso iterativo de perturbar la muestra original utilizando un ancho de la campana Gaussiana inferior, por lo que las sucesivas muestras que se generen estarán más cerca de la muestra original y por tanto, tendrán mayor probabilidad de caer dentro del área de

aceptación.

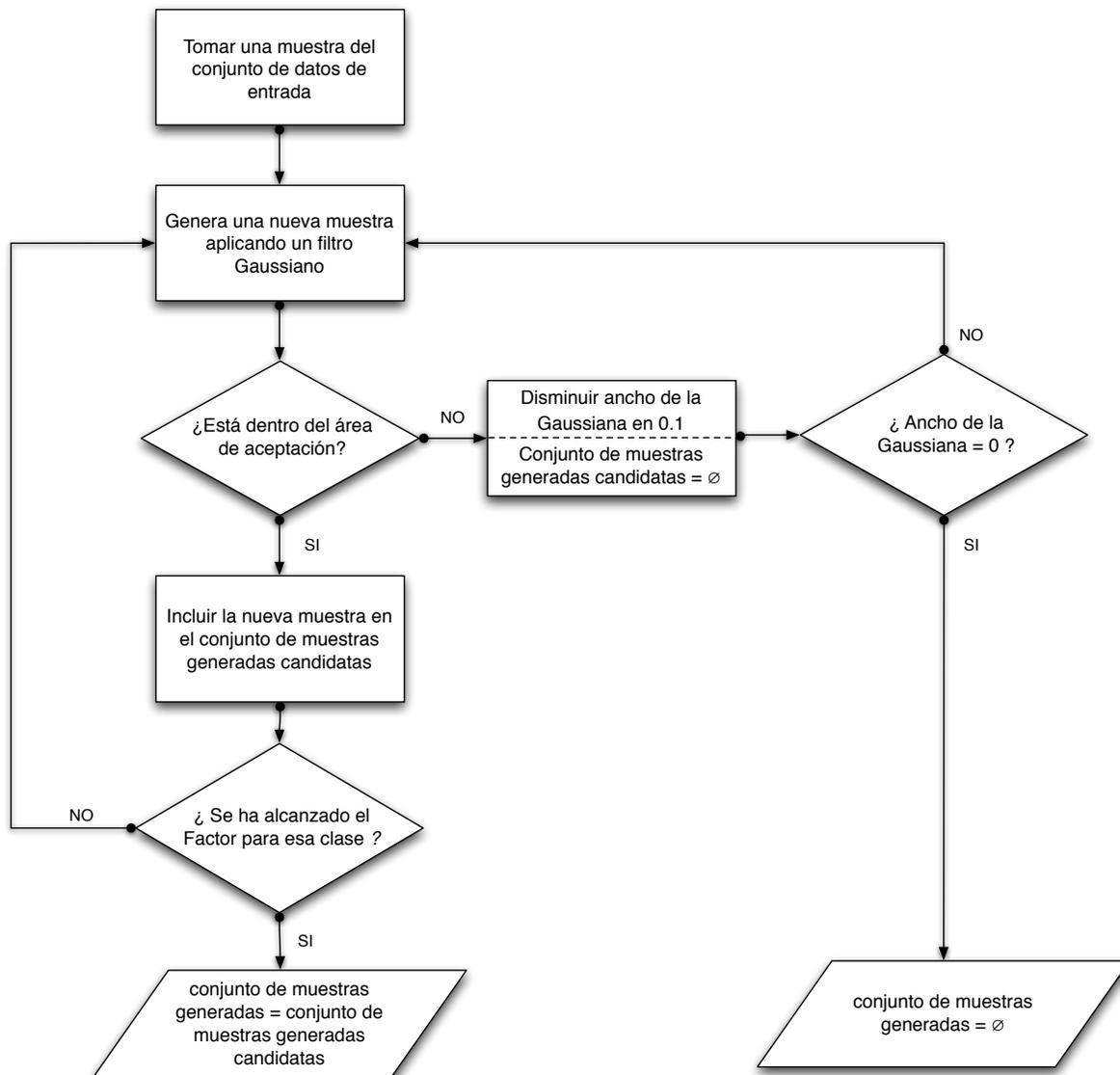


Figura 2.1: Diagrama de flujo del algoritmo de sobre-muestreo SNEOM para una muestra del conjunto de entrada.

El primer paso llevado a cabo por el método es el pre-procesamiento de los datos. Antes de realizar la proyección de los datos de entrada es necesario estandarizarlos. Ésta es una tarea esencial debido a que el método proyectivo utilizado es sensible al escalado relativo de las variables originales. Tras la estandarización, los datos son transformados en una distribución Gaussiana centrada en cero y desviación estándar unitaria.

Después de la fase de pre-procesado, se lleva a cabo un proceso de extracción de características utilizando un método proyectivo neuronal. Se genera una transformación del espacio de entrada en un nuevo espacio de características, concretamente el espacio de las Componentes Principales

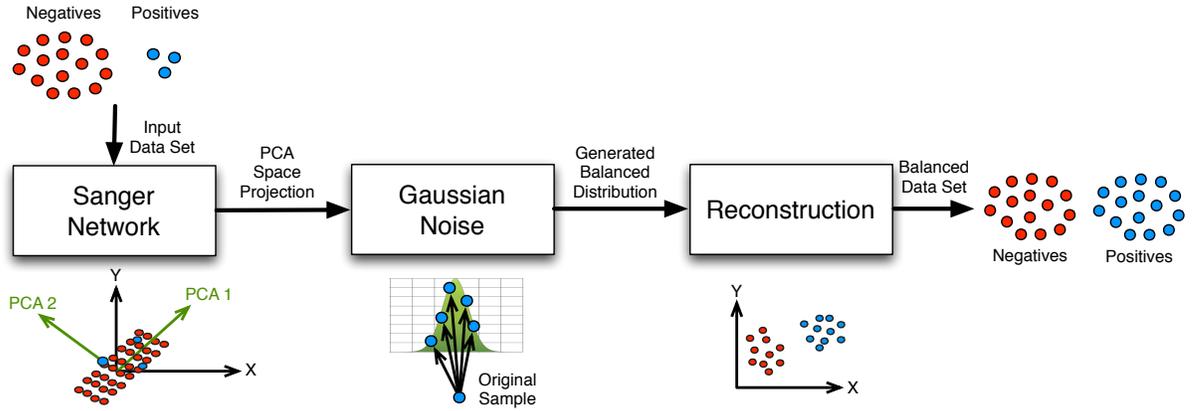


Figura 2.2: Esquema de las diferentes fases del método SNEOM.

(PCA), con una dimensionalidad inferior, preservando la máxima cantidad de información posible. El método propuesto implementa este proceso de transformación utilizando una aproximación neuronal cuyos procesos convergen en PCA, la red de Sanger, también denominada Algoritmo Hebbiano Generalizado adaptada para poder llevar a cabo el procesamiento de datos *missing*. Esta adaptación sigue un esquema similar a [Samad and Harp, 1992], haciendo que estos valores *missing* no contribuyan a la salida ni modifiquen los pesos de la red [García Báez et al., 2007]:

$$y_i = \sum_{j \in P_t} w_{ij} x_j \quad (2.1)$$

Donde P_t es el conjunto de unidades de entrada, j , cuyos valores x_j están disponibles en el instante t . La regla de aprendizaje que progresivamente ajusta esta matriz de pesos viene dada por la siguiente expresión:

$$\Delta w_{ij} = \begin{cases} \eta(t) y_i (x_j - \sum_{k=1}^i y_k w_{kj}) & \text{if } j \in P_t \\ 0 & \text{otherwise} \end{cases} \quad (2.2)$$

En la expresión dada por la ecuación 2.2, el radio de aprendizaje $\eta(t)$ es una función linealmente decreciente en el instante t . Esta fórmula produce que los pesos de la red de Sanger converjan a las componentes principales del conjunto de datos de entrenamiento.

Una vez realizada la transformación al espacio PCA por medio de la red de Sanger, se obtienen los *scores* (datos de entrada proyectados en el espacio de características) y los pesos o *loads* (coeficientes de correlación entre las cargas y el conjunto de datos de entrada).

La cantidad de componentes principales depende del porcentaje de varianza obtenido después de entrenar la red de Sanger. Es importante resaltar que la bondad de la reconstrucción posterior de los datos proyectados al espacio de entrada inicial, está fuertemente relacionada con este parámetro, por lo que es recomendable seleccionar una cantidad de componentes principales suficiente como para proporcionar alrededor del 90 % de la varianza de los datos de entrada.

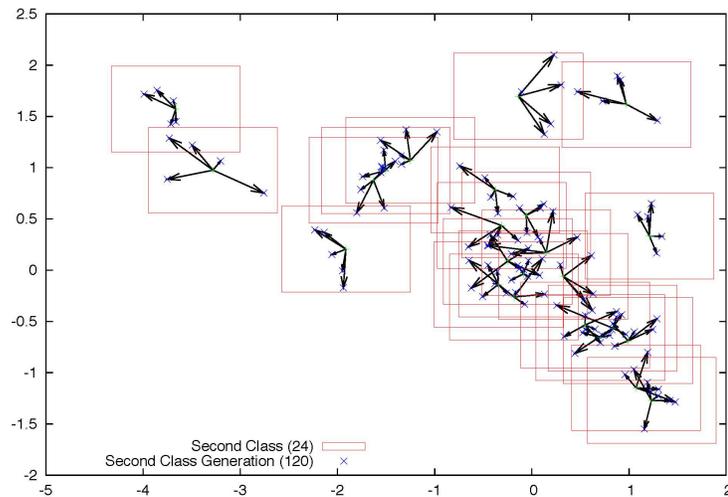
La siguiente fase del método consiste en aplicar el procedimiento de *over-sampling* extendido sobre el espacio de características. En primer lugar, se establecen los factores de las clases minoritaria y mayoritaria a través de dos parámetros introducidos por el usuario, los cuáles indican la cantidad de muestras sintéticas que el algoritmo generará para cada clase. Si el factor de la clase mayoritaria es distinto de cero, la versión del método utilizada es la de *over-sampling* extendido, y se generarán muestras sintéticas tanto para la clase minoritaria como para la clase mayoritaria. Posteriormente, el procedimiento calcula las áreas de aceptación para cada muestra proyectada. Éstas determinan los límites dentro de los cuales pueden generarse nuevas muestras, utilizando las varianzas de la distribución de datos. Las regiones de aceptación son calculadas como las desviaciones estándar para cada variable de cada clase. Una vez calculadas las áreas de aceptación para cada clase, éstas se centran en las coordenadas de cada muestra proyectada. Seguidamente, el método toma iterativamente cada muestra y le añade un ruido Gaussiano, empleando un ancho de campana igual a la unidad, para perturbar la muestra original y genera una nueva muestra cercana 2.3(a). En caso de que la nueva muestra caiga dentro del área de aceptación, una nueva muestra sintética será generada y otras nuevas muestras sintéticas serán generadas hasta que el factor para esa clase sea alcanzado. Si cualquier muestra sintética cae fuera de los límites del área de aceptación, ésta no será aceptada y el ancho de la campana del ruido Gaussiano será decrementado en 0.1 unidades, con el objetivo de generar a continuación nuevas muestras más cercanas a la original y tener así mayor probabilidad de caer dentro de los límites del área de aceptación. La fórmula siguiente describe el ruido Gaussiano utilizado para perturbar cada muestra original:

$$s_{ij} = z_{ij} + \text{sqrt}(\sigma)RNOR_{ij}\rho + \mu \quad (2.3)$$

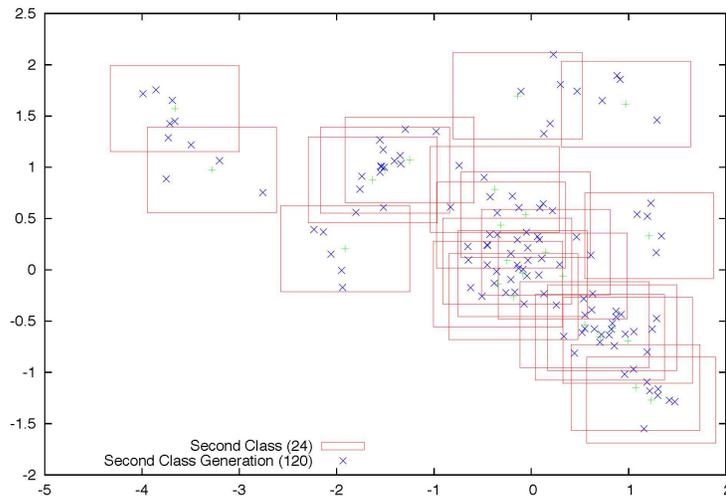
En la ecuación anterior, z_{ij} es la muestra original normalizada, $RNOR_{ij}$ es un valor aleatorio perteneciente a una función de densidad probabilística (PDF) Gaussiana centrada en cero y con una desviación estándar unitaria, obtenida mediante el método de Ziggurat [Marsaglia and Tsang, 2000].

Los datos proyectados en el espacio de las componentes principales preservan las características estadísticas de la distribución para cada clase. Perturbar las muestras en este espacio transformado nos permite modificar unas pocas dimensiones en lugar de todo el conjunto de variables del espacio de entrada. De esta forma, el número de dimensiones perturbadas es reducido y ello facilita también la visualización del proceso completo. En la figura 2.3(b) se puede observar el resultado de perturbar las muestras originales de la clase minoritaria. En la figura 2.4 se muestra el resultado de una aproximación de sobre-muestreo extendido, donde también el factor de la clase mayoritaria es distinto de cero.

Después de la etapa de generación, se encuentra la fase de reconstrucción, que consiste en transformar el espacio de características en el espacio de entrada, nuevamente. El vector original de entrada (x) puede ser reconstruido (Rx) con una mínima pérdida de información desde el vector de m-salidas (y) y los pesos de la red de Sanger (W), lo que es también denominado, el proceso de descompresión. El uso de este método proyectivo como técnica de compresión es conocido también como *Hotelling* o transformada de *Karhunen and Leove* (KL) [Diamantaras and Kung, 1996]. La



(a)



(b)

Figura 2.3: a) Muestras sintéticas generadas a partir de cada muestra de la clase la clase minoritaria. Las cruces verdes representan las muestras originales, las cruces azules las muestras generadas y las áreas en color rojo las regiones de aceptación para esa clase. b) Conjunto de muestras originales de la clase minoritaria más el conjunto de muestras generadas tras aplicar SNEOM.

ecuación que lleva a cabo la reconstrucción o descompresión es la siguiente:

$$Rx_i = \sum_{j=1}^m w_{ij}y_j \quad (2.4)$$

Una vez que los datos han sido proyectados nuevamente en el espacio de entrada, se realiza el proceso inverso a la estandarización además de una cuantización sobre los datos. Este último

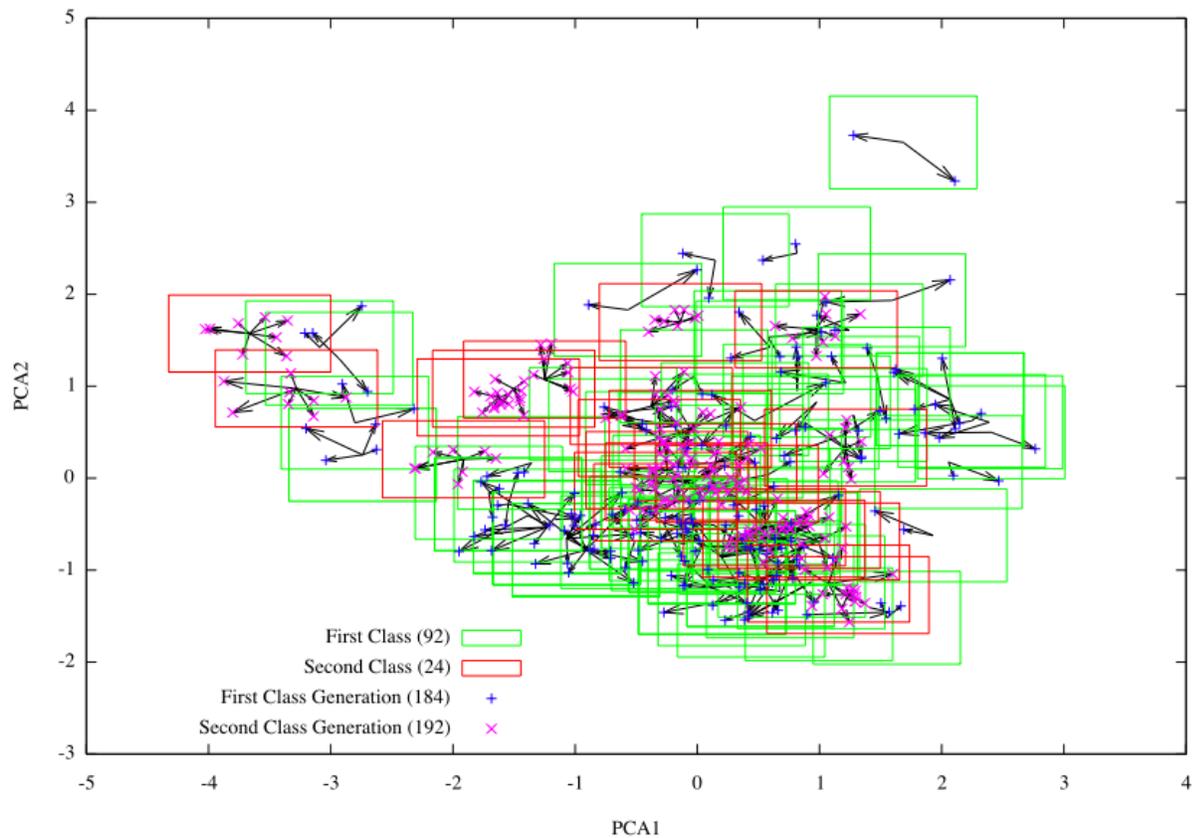


Figura 2.4: Conjunto de muestras originales más generadas tras aplicar un sobre-muestreo extendido utilizando SNEOM. Las áreas de color verde corresponden a las regiones de aceptación de la clase mayoritaria y las áreas de color rojo a las de la clase minoritaria.

paso es crucial para ajustar los datos generados al rango de los datos de entrada. El resultado de aplicar SNEOM puede verse en la figura 2.5. En ésta se visualiza el conjunto de entrada desde 2 y 3 dimensiones (figuras 2.5(a) y 2.5(b)), así como sus homónimas tras haber aplicado un procedimiento de sobre-muestreo sobre la clase minoritaria (figuras 2.5(c) y 2.5(d)).

2.3.3. Aplicación de SNEOM en problemas de clasificación médica no balanceada

Con el objetivo de validar el método SNEOM se han llevado a cabo diferentes experimentos sobre conjuntos de datos no balanceados pertenecientes al ámbito del diagnóstico médico. Se han analizado los resultados de varias clasificaciones bimodales utilizando paradigmas neuronales y no neuronales, concretamente: un clasificador probabilístico Naïve Bayes (NB), un árbol de clasificación C4.5, y un perceptrón multicapa (MLP), embebidos todos ellos en el software para minería de datos *Environment for Knowledge Analysis* (WEKA) [I.H. Witten, 2005]. A continuación se

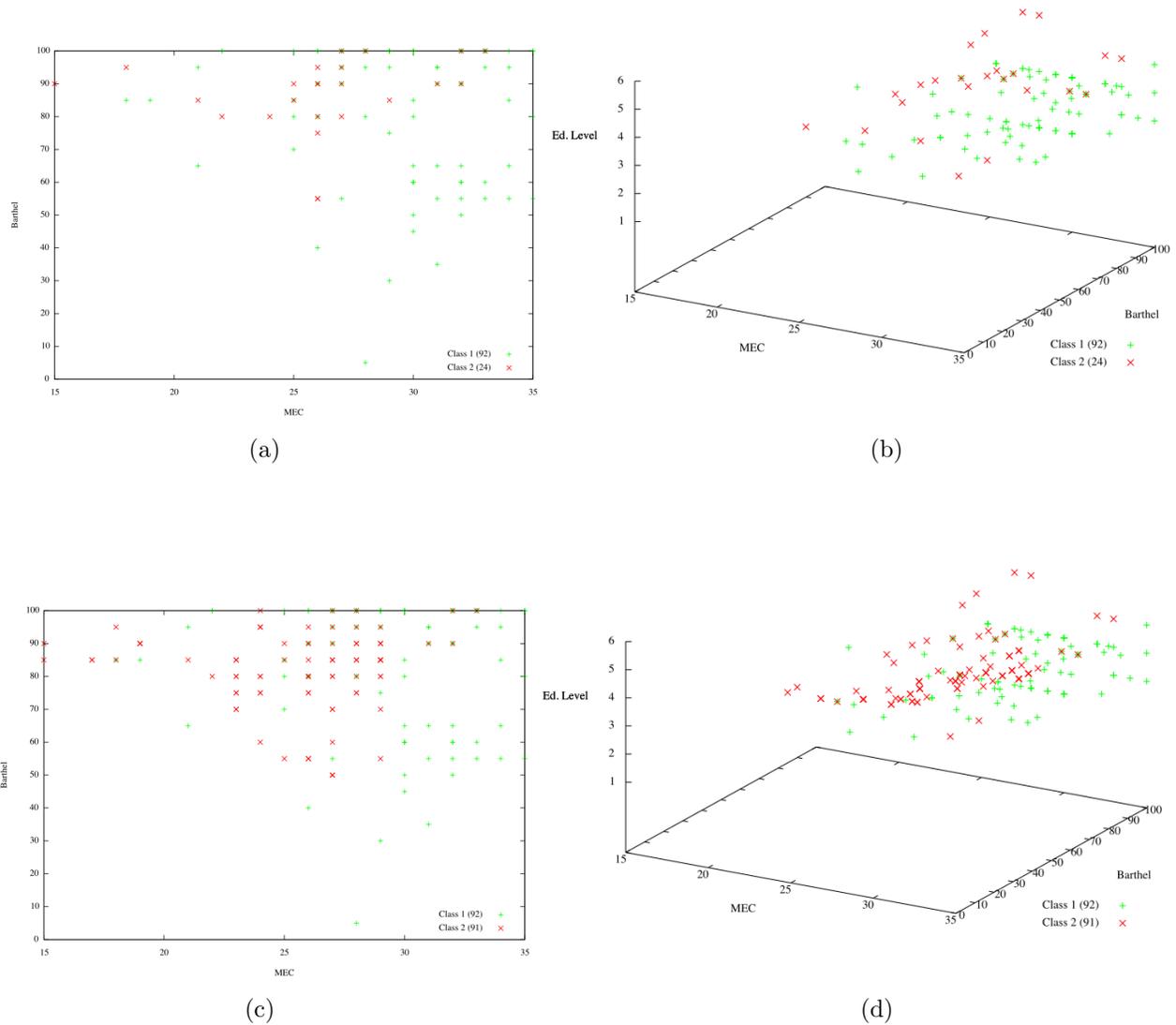


Figura 2.5: a) Variable MEC frente a Barthel del conjunto de datos de entrada. b) Variable MEC frente a Barthel y nivel educativo del conjunto de datos de entrada. c) Variable MEC frente a Barthel del conjunto de datos tras aplicar un sobre-muestreo a la clase minoritaria. d) Variable MEC frente a Barthel y nivel educativo del conjunto de datos tras aplicar un sobre-muestreo a la clase minoritaria.

describirán los conjuntos de datos utilizados, así como los resultados obtenidos como validación del método SNEOM.

Conjuntos de datos

Se han seleccionado tres bases de datos de ámbito médico las cuales se utilizaron para entrenar y testear tres paradigmas de clasificación diferentes. Las clasificaciones estudiadas han sido: discriminar entre sujetos con DCL y sujetos sanos, detección de cáncer de mama, identificando la existencia de tumores malignos, y clasificación de patologías en la columna vertebral, distinguiendo entre exploraciones normales y patológicas. Los dos últimos conjuntos de datos han sido extraídos del repositorio público UCI [Repository, 2007] mientras que el primero ha sido obtenido a partir de consultas reales utilizando el entorno clínico virtual EDEVITALZH [Araujo et al., 2004]. Estos conjuntos de datos difieren en sus tamaños y tasas de desequilibrio, resultando en escenarios variados donde probar SNEOM. Con el objetivo de llevar a cabo un estudio homogéneo del método, con alta fiabilidad en la fase de reconstrucción y fácil visualización, se ha seleccionado un subconjunto de atributos de los conjuntos de datos, tomando como referente la configuración de los datos relacionados con el DCL. Por tanto, se han utilizado los siguientes conjuntos de datos:

1. El conjunto de datos de pacientes con DCL, obtenido desde EDEVITALZH [Araujo et al., 2004]. Este conjunto contiene 116 consultas relativas a un grupo de pacientes de la Unidad de Atención a la Dependencia de Santander (Cantabria, España). Dentro de esta base de datos hay 92 consultas correspondientes a sujetos sanos y las 24 restantes hacen referencia a consultas donde a los pacientes se les ha diagnosticado DCL. Las variables incluidas en esta base de datos son: el test Mini-Examen Cognitivo (MEC), que corresponde a la versión española del test de examen de estado MiniMental (MMSE), el índice de Barthel, que evalúa las capacidades funcionales del individuo, y el nivel educativo. Estas variables se encuentran detalladas en mayor profundidad en la sección 3.2.
2. Conjunto de datos Breast Cancer Wisconsin [Repository, 2007]. Este conjunto de datos está compuesto por 683 muestras, 444 de ellas correspondientes a tumores benignos y las restantes 239 pertenecientes a tumores malignos. Cada instancia está representada por cuatro variables relacionadas con información de citologías de mamas: espesor de la masa, uniformidad en el tamaño de las células, uniformidad en la forma de las células y adhesión marginal.
3. Conjunto de datos Vertebral Column [Repository, 2007]. La cantidad total de muestras contenidas en este conjunto de datos es 310, divididas en 210 muestras representando exploraciones anormales y 100 muestras acordes a exploraciones normales. Cada paciente se encuentra representado en el conjunto de datos por cuatro atributos biomecánicos derivados de la forma y orientación de la pelvis y la espina dorsal: incidencia pélvica, inclinación pélvica, ángulo de lordosis lumbar e inclinación sacra.

Paradigmas de clasificación

Para validar el método SNEOM se han seleccionado tres clasificadores relativos a tres diferentes paradigmas para probar el método en diferentes escenarios de clasificación. Estos clasificadores han

sido lanzados desde WEKA [I.H. Witten, 2005]. A continuación se indica la clase que identifica a cada clasificador dentro de WEKA y se describe brevemente a cada uno de ellos.

Naïve Bayes (`weka.classifiers.bayes.NaiveBayes`) Los clasificadores Bayesianos proporcionan una aproximación simple, con una semántica clara, para representar, utilizar y aprender el conocimiento probabilístico. El método está diseñado para usar en tareas de inducción supervisada, en las cuáles el objetivo es predecir de manera precisa la clase de las instancias de test para las cuales las instancias de entrenamiento han incluido la información de la clase [John and Langley, 1995]. Uno puede ver tal clasificador como una forma especializada de red Bayesiana, denominada *naïve* porque depende de dos suposiciones importantes. En particular, asume que los atributos predictivos son condicionalmente independientes dadas la clase, y presupone ni atributos ocultos ni latentes influyen el proceso de predicción. Por ello, cuando se representa gráficamente, un clasificador Bayesiano naive tiene la forma de la figura 2.6, en la cual los arcos son dirigidos desde el atributo de la clase hacia los atributos observables, predictivos [Buntine, 1994].

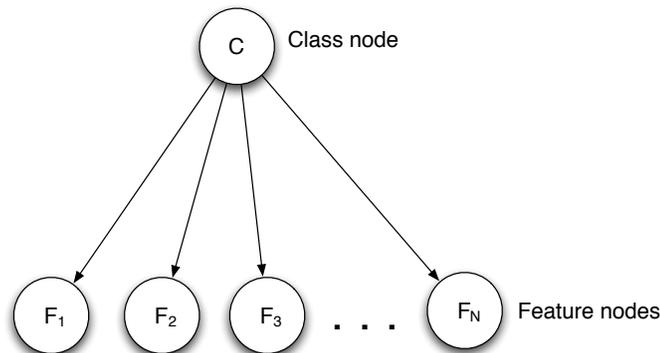


Figura 2.6: Representación de una red Bayesiana.

Árbol de clasificación C4.5 (`weka.classifiers.trees.J48`) Los árboles de decisión comienzan con un conjunto de casos, o ejemplos, y crean una estructura de datos arbórea que puede ser utilizada para clasificar nuevos casos. Cada caso es descrito por un conjunto de atributos (o características) que pueden tener valores numéricos o simbólicos. A cada muestra de entrenamiento se le asocia una etiqueta representando el nombre de la clase. Cada nodo interno de un árbol de decisión contiene una condición, el resultado de la cuál es utilizado para decidir que rama seguir desde ese nodo [Quinlan, 1993]. Por ejemplo, una condición podría ser '*es $x > 4$?*', para un atributo x dado. Si el resultado es verdadero, entonces la muestra procederá hacia la rama de la izquierda, y en el caso contrario, la muestra descenderá por la rama de la derecha. Los nodos hojas contienen etiquetas de clases en lugar de condiciones. En el modo de clasificación, cuando una muestra de test (que no tiene etiqueta) alcanza un nodo hoja, el C4.5 la clasifica utilizando la etiqueta almacenada para ese nodo. El árbol de clasificación C4.5 y su predecesor, el ID3, utilizan fórmulas basadas en la teoría de la información para evaluar la "bondad" de una condición; en particular, ellos eligen las condiciones que extraen la mayor cantidad de información desde un conjunto de casos, dada

la restricción de que sólo un atributo puede ser evaluado [Quinlan, 1993]. En la figura 2.7 puede observarse la representación gráfica de un árbol de decisión.

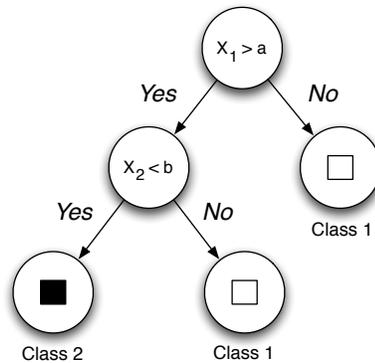


Figura 2.7: Representación de árbol de decisión.

Perceptrón multicapa (weka.classifiers.functions.MultilayerPerceptron) El perceptrón multicapa es una estructura de red neuronal que puede ser utilizado para clasificación y regresión. Un MLP consta de múltiples capas de nodos en un grafo dirigido con cada capa completamente conectada con la siguiente (ver figura 2.8). Excepto para los nodos de la capa de entrada, cada nodo es una neurona (o elemento de procesamiento) con una función de activación no lineal.

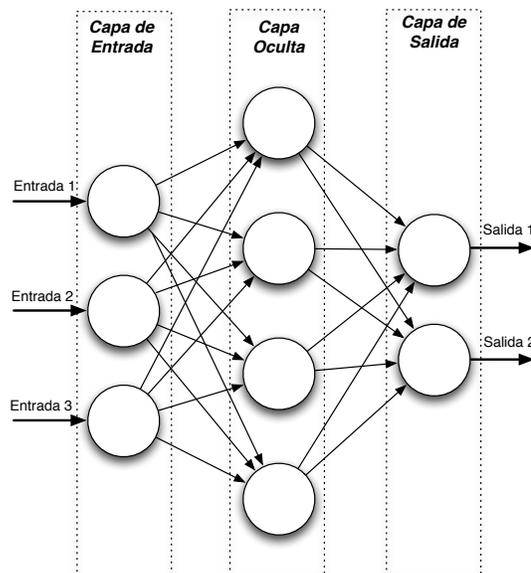


Figura 2.8: Representación gráfica de un Perceptrón con una capa oculta.

El algoritmo más popular para entrenar MLP es el denominado *backpropagation* [Werbos, 1974]. Está basado en una corrección supervisada del error cuadrático generada en la capa de salida utilizando el método del descenso del gradiente. El MLP es una modificación del estándar perceptrón

lineal y puede discriminar entre datos que no son linealmente separables. En la sección 1.3.1 se encuentra explicado este clasificador con mayor nivel de detalle.

Validación del método

Los resultados han sido obtenidos entrenando tres paradigmas de clasificación diferentes con tres distribuciones de datos, un conjunto de datos no balanceado (original), y dos conjuntos de datos balanceados utilizando las aproximaciones de *Over-Sampling* (OS) y de *Over-Sampling* Extendido (EOS) de SNEOM. Para la primera distribución, los clasificadores fueron entrenados utilizando las muestras generadas mediante el método SNEOM. Todos los conjuntos de datos fueron divididos en conjuntos de entrenamiento, test y validación, tomando un 75 % entre los dos primeros conjuntos y el 25 % restante para el conjunto de validación.

Se han tomado en consideración una variedad de medidas que nos han permitido obtener un conjunto de evaluaciones estables y fiables con el objetivo de validar y demostrar la efectividad del método de sobre-muestreo desarrollado como parte de este trabajo, SNEOM. Las métricas utilizadas han sido:

- Precisión: Es la tasa de muestras de validación correctamente clasificadas. La ecuación que la define es la siguiente:

$$Precision = \frac{(TP + TN)}{TP + FP + FN + TN} \quad (2.5)$$

donde TP representa el número de verdaderos positivos, FP denota el número de falsos positivos, y TN y FN indican el número de verdaderos y falsos negativos, respectivamente.

- Sensibilidad: Esta métrica mide la proporción de muestras positivas correctamente identificadas.

$$Sensibilidad = \frac{TP}{TP + FP} \quad (2.6)$$

- Especificidad: Representa la tasa de muestras negativas correctamente clasificadas.

$$Especificidad = \frac{TN}{TN + FN} \quad (2.7)$$

- Área bajo la curva (AUC): El área bajo la curva ROC representa el rendimiento esperado para un clasificador dado en términos de tasa de falsos positivos (FPR) frente a la tasa de verdaderos positivos (TPR).

En las tablas de las métricas resultantes se indica el tipo de dato usado y el clasificador. Así, el conjunto de datos de entrada no balanceado utilizado para el entrenamiento es denotado como *Original*. Los conjuntos de datos compuestos por muestras sintéticas utilizando el método SNEOM están designados por $SNEOM_{OS}$ para la aproximación *Over-Sampling* pura, y $SNEOM_{EOS}$ para

	Precisión	Sensibilidad	Especificidad	AUC
Original+MLP	89.67 %	50.00 %	100 %	0.993
SNEOM _{OS} +MLP	90.00 %	83.33 %	91.30 %	0.964
SNEOM _{EOS} +MLP	82.76 %	100 %	78.26 %	0.986
Original+NB	79.31 %	0 %	100 %	0.500
SNEOM _{OS} +NB	82.76 %	66.67 %	86.96 %	0.808
SNEOM_{EOS}+NB*	93.10 %	100 %	91.30 %	0.928
Original+C4.5	79.31 %	0 %	100 %	0.500
SNEOM _{OS} +C4.5	79.31 %	100 %	73.91 %	0.870
SNEOM _{EOS} +C4.5	82.76 %	100 %	78.26 %	0.906

Tabla 2.1: Métricas obtenidas para el conjunto de datos de pacientes con DCL.

la aproximación *Over-Sampling* extendida. Los mejores resultados están indicados en negrita con un asterisco.

Para el primer conjunto de datos, los resultados obtenidos se muestran en la tabla 2.1. En este caso, entrenando tanto un Naïve Bayes como un árbol de decisión C4.5, utilizando el conjunto de datos no balanceado, todas las muestras fueron predichas como pertenecientes a la clase mayoritaria, por lo que todas las instancias de la clase minoritaria fueron clasificadas incorrectamente (sensibilidad = 0 %). Entrenando un MLP con el mismo conjunto de datos, sólomente la mitad de las muestras de la clase minoritaria fueron aciertos (sensibilidad = 50 %) y todas las muestras del conjunto mayoritario fueron correctamente clasificadas (100 %). Debido a que el conjunto de validación no balanceado está compuesto por aproximadamente cuatro veces más muestras de la clase mayoritaria (23 sujetos sanos) que de la clase minoritaria (6 sujetos con DCL), el valor de AUC es el más alto en la tabla (0.993). Atendiendo a los clasificadores entrenados con el conjunto de datos OS, las sensibilidades fueron significativamente superiores que entrenando a los clasificadores con el conjunto de datos desequilibrado, aunque las especificidades se vieron decrementadas debido a que los clasificadores han realizado una clasificación más justa que en el anterior escenario (en el que tanto las muestras positivas como las negativas fueron identificadas como pertenecientes a la clase mayoritaria). En este caso que nos ocupa, los valores de precisión son siempre mayores que los obtenidos con el conjunto de entrenamiento no balanceado excepto para el clasificador C4.5, en el que coinciden (79.31 %). Cuando se emplea el conjunto de datos EOS, se consigue un 100 % de sensibilidad para todos los clasificadores. Utilizando este conjunto de entrenamiento, las tasas de especificidad fueron menores que las obtenidas con el conjunto de datos no balanceado, debido a la enorme influencia de clasificar todas las muestras del conjunto de validación como muestras de la clase mayoritaria, aunque para casi todos los casos supera a los valores obtenidos con el conjunto de datos OS (excepto para el clasificador MLP). Aunque el valor de AUC para un clasificador MLP entrenado con el conjunto de datos no balanceado es el mayor, este valor no es fiable para indicar una buena tasa de clasificación, debido a que sólo el 50 % de las muestras de la clase minoritaria fueron correctamente clasificadas. La mejor combinación fue la formada por el clasificador NB entrenado con el conjunto de datos EOS, obteniendo las mejores tasas de precisión y sensibilidad

(93.1 % y 100 %, respectivamente) y una alta tasa de especificidad (91.3 %).

En relación al conjunto de datos Breast Cancer Wisconsin, cuyos resultados se muestran en la tabla 2.2, los valores de sensibilidad y AUC resultaron siempre mejor para los conjuntos de datos balanceados OS y EOS, que para el conjunto no balanceado, por lo que, utilizando SNEOM es más probable identificar correctamente las muestras de la clase minoritaria. Adicionalmente, la precisión es mayor para los conjuntos de entrenamiento OS y EOS que para el conjunto *Original* en la mayoría de los casos, excepto para el clasificador C4.5 que es ligeramente menor. Los valores de especificidad se encuentran muy cercanos entre las distribuciones balanceadas y la no balanceada utilizando los clasificadores MLP y NB, obteniéndose el mayor valor (96.64 %) para un clasificador NB entrenado con el conjunto de datos OS. Esta configuración también proporciona el mayor valor de precisión, por lo que es una de las mejores combinaciones junto con la formada por el clasificador MLP entrenado con el conjunto de entrenamiento EOS. Esta última proporciona la mayor tasa de sensibilidad (96.61 %), reconociendo así el mayor número de muestras de la clase minoritaria.

	Precisión	Sensibilidad	Especificidad	AUC
Original+MLP	94.12 %	89.80 %	96.40 %	0.991
SNEOM _{OS} +MLP	95.29 %	94.92 %	95.50 %	0.993
SNEOM_{EOS}+MLP*	95.29 %	96.61 %	94.50 %	0.993
Original+NB	95.29 %	93.22 %	96.40 %	0.992
SNEOM_{OS}+NB*	95.88 %	95.00 %	96.40 %	0.993
SNEOM _{EOS} +NB	95.29 %	95.00 %	95.50 %	0.994
Original+C4.5	94.12 %	93.22 %	94.59 %	928
SNEOM _{OS} +C4.5	93.53 %	96.61 %	91.89 %	0.961
SNEOM _{EOS} +C4.5	93.53 %	95.00 %	92.79 %	0.939

Tabla 2.2: Métricas obtenidas para el conjunto de datos Breast Cancer Wisconsin.

Los resultados para el conjunto de datos Vertebral Column están representados en la tabla 2.3. En ésta se observa que resulta bastante difícil para cualquier clasificador discriminar correctamente entre las clases, como indican los bajos porcentajes obtenidos para todas las métricas. En este escenario, las tasas de precisión son muy similares para todos los casos, excepto para el clasificador C4.5 entrenado con el conjunto Original, que es mayor que el resto de configuraciones debido a que todas las muestras fueron clasificadas como pertenecientes a la clase mayoritaria, y consecuentemente todas las muestras de la clase minoritaria fueron identificadas incorrectamente (sensibilidad = 0 %). Por tanto, para este problema, la precisión no es una medida fiable para comparar diferentes configuraciones. Los valores de sensibilidad obtenidos fueron siempre significativamente superiores para OS y EOS que para el conjunto Original, y los valores de AUC fueron mejores cuando se empleó SNEOM. Las especificidades fueron ligeramente superiores para los conjuntos de entrenamiento no balanceados, debido a que los clasificadores predijeron en la mayoría de los casos que las muestras de validación pertenecían al conjunto mayoritario. Los clasificadores MLP y NB proporcionaron buenos resultados nuevamente. Utilizando un clasificador MLP entrenado con el conjunto OS se obtuvieron las mejores tasas de sensibilidad y de AUC (85 % y 0.743 %, res-

pectivamente), indicando que este clasificador entrenado con el conjunto equilibrado con la técnica de sobre-muestreo pura de SNEOM es capaz de identificar más muestras del conjunto minoritario que el resto de clasificadores. El clasificador NB entrenado con el conjunto EOS consiguió la mejor precisión, siendo el que proporcionó el mejor balance entre la clasificación de muestras mayoritarias y minoritarias (sensibilidad = 72 %, especificidad = 61.54 %).

	Precisión	Sensibilidad	Especificidad	AUC
Original+MLP	64.94 %	40.00 %	67.31 %	0.700
SNEOM_{OS}+MLP	63.64 %	85.00 %	55.77 %	0.743
SNEOM _{EOS} +MLP	63.64 %	72.00 %	59.62 %	0.700
Original+NB	63.64 %	64.00 %	63.46 %	0.708
SNEOM _{OS} +NB	62.34 %	68.00 %	59.62 %	0.718
SNEOM_{EOS}+NB	64.94 %	72.00 %	61.54 %	0.739
Original+C4.5	67.53 %	0 %	100 %	0.500
SNEOM _{OS} +C4.5	63.64 %	72.00 %	59.62 %	0.671
SNEOM _{EOS} +C4.5	62.34 %	85.00 %	53.85 %	0.672

Tabla 2.3: Métricas obtenidas para el conjunto de datos Vertebral Column.

Los resultados obtenidos para estas tres aplicaciones médicas con diferentes niveles de desequilibrio y tamaños han validado el método SNEOM y justificado su bondad para utilizarlo en tareas de clasificación con clases no balanceadas. Además, se ha demostrado la validez de los datos generados, debido a que se ha entrenado a los clasificadores con datos sintéticos y se ha validado con datos originales. En todos los experimentos, los mejores resultados fueron obtenidos tras entrenar a los clasificadores con los conjuntos de datos balanceados utilizando la técnica de *Over-Sampling* puro y la novedosa técnica de *Over-Sampling* extendido propuesta en SNEOM.

2.3.4. SNEOM-GUI: Interfaz Gráfica de Usuario

Con el fin de disponer de una herramienta software amigable mediante la cual poder aplicar el método SNEOM de manera sencilla e intuitiva, se ha desarrollado una aplicación de escritorio, denominada *SNEOM Graphical User Interface* (SNEOM-GUI), que permite sobre-muestrear un conjunto de datos proporcionado por el usuario. Tras configurar rápida y sencillamente los parámetros de la red neuronal de Sanger y del método de sobre-muestreo, la aplicación proporciona al usuario un conjunto de vistas de todo el proceso de generación, desde la visualización de los datos de entrada en 2D y 3D, hasta la reconstrucción final al espacio de entrada de los datos sintéticos generados.

SNEOM-GUI ha sido desarrollada en el lenguaje de programación C de Dennis M. Ritchie [Kernighan and Ritchie, 1991], utilizando la librería *GIMP Tool Kit* (GTK) para el desarrollo de interfaces de usuario de la fundación GNOME [The GTK+ Project, 1998]. Una de las principales ventajas de esta librería de desarrollo es que es multiplataforma, estando disponible para los entornos gráficos GNOME, XFCE, ROX, Windows y Mac OSX, entre otros.

GTK pone a disposición de los desarrolladores un amplio conjunto de *widgets* u objetos para integrar en sus interfaces de usuario, tales como ventanas, botones, menús, etiquetas, deslizadores, pestañas, etc. GTK es software libre y parte del proyecto GNU, permitiendo a los desarrolladores utilizarla sin necesidad de pagar costes de licencia ó derechos de autor.

Para la generación de las gráficas SNEOM-GUI utiliza Gnuplot. Gnuplot es una utilidad para generar gráficas dirigida por comandos disponible para OS/2, Microsoft Windows, Mac OSX, VMS y otras muchas plataformas. El código fuente posee derechos de autor pero puede ser distribuido libremente. Gnuplot fue originalmente creada para permitir a los científicos y estudiantes visualizar funciones matemáticas y datos de forma interactiva, pero ha pasado a soportar muchos usos no interactivos tales como la creación de *scripts* webs. Gnuplot es también utilizada como el núcleo gráfico de aplicaciones de terceros como Octave, la alternativa software libre a Matlab.

SNEOM-GUI proporciona un conjunto de funcionalidades, entre las que podemos destacar:

- **Compatibilidad con múltiples formatos de ficheros.** SNEOM-GUI soporta un amplio número de diversos formatos de ficheros ampliamente utilizados para almacenar patrones y trabajar con software especializado en minería de datos. Entre estos formatos se encuentran las consultas a bases de datos exportadas en texto plano, ficheros CSV, C4.5 y ARFF. En el apartado 2.3.5 se describen en profundidad todos y cada unos de los formatos soportados por SNEOM-GUI.
- **Visualización del procedimiento.** SNEOM-GUI proporciona un conjunto de vistas para visualizar todo el proceso de sub-muestreo: desde la visualización de los datos de entrada en 2D y 3D hasta la reconstrucción de los datos generados, pasando por la proyección en el espacio PCA y la visualización de las muestras generadas en el espacio proyectado dentro de las regiones de aceptación.
- **Gráficas exportables en formato vectorial.** SNEOM-GUI pone a disposición del usuario las gráficas de todo el proceso de sub-muestreo en los formatos de imagen PS, PDF y PNG. En primer lugar, Las imágenes son obtenidas en formato *PostScript* (PS) mediante Gnuplot. A continuación, SNEOM-GUI utiliza las aplicaciones por consola Ghostscript [Ghostscript, 1988] e ImageMagick [ImageMagick, 1999] para convertir las imágenes a PDF y PNG, respectivamente.
- **Diseño modular.** SNEOM-GUI ha sido diseñada con el objetivo de incrementar sus funcionalidades cómodamente, por lo que está estructurada en diversos ficheros que la hacen modular y fácilmente escalable. En la figura 2.9 puede observarse un diagrama de los ficheros fuente principales que componen la aplicación de la interfaz. Esencialmente, los más importantes son los siguientes:
 - *main*: Contiene todas las funciones para dibujar la interfaz gráfica así como todos los manejadores de eventos.
 - *gnuplot-tools*: Contiene todas las funciones de generación de gráficos bidimensionales y tridimensionales, mediante Gnuplot. Estas funciones son altamente parametrizables, para poder generar desde el código C cualquier gráfico que pudiera generarse desde el intérprete de Gnuplot.

- *matrix-tools*: Contiene todas las funciones de tratamiento de matrices. Tanto las funciones para obtener las áreas de aceptación como el filtro Gaussiano están implementadas en este fichero.
- *tools*: Contiene funciones de tratamiento de ficheros así como de manejo de rutas y ristas y otras utilidades para la aplicación.

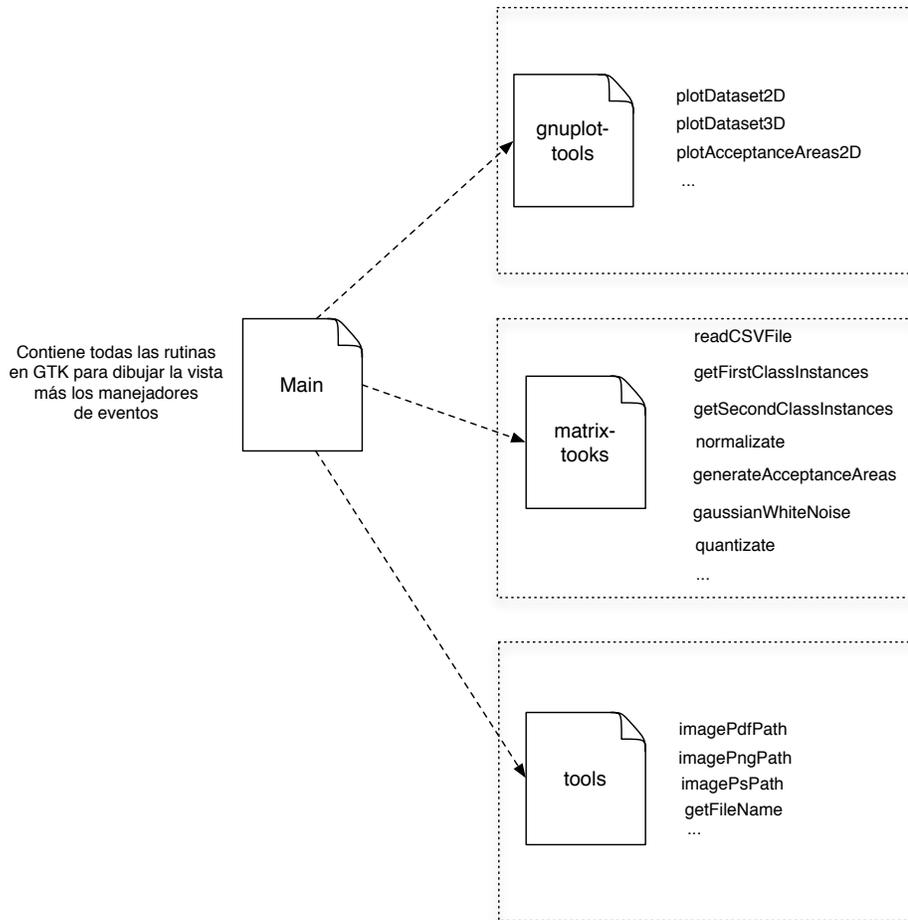


Figura 2.9: Diagrama de los ficheros principales que componen SNEOM-GUI junto a sus métodos más significativos.

2.3.5. Formatos de ficheros compatibles

SNEOM-GUI permite importar datos desde diferentes formatos de ficheros, por lo que resulta bastante simple cargar ficheros de patrones desde repositorios públicos y exportarlos a formatos compatibles con otros programas de minería de datos.

A continuación se describen los formatos de ficheros compatibles con SNEOM-GUI.

Consultas a bases de datos en texto plano SNEOM-GUI permite la lectura de ficheros exportados desde las bases de datos en formato de texto plano, separando cada campo o columna de la base de datos por el símbolo ”,”. A continuación, se puede observar un ejemplo de una de las filas obtenidas tras la consulta a la base de datos del entorno clínico virtual EDEVITALZH [Araujo et al., 2004]:

“21”;“10000505101010151505”;“3”;“DCL”

CSV Los ficheros CSV (del inglés Comma-Separated Values) son un tipo de documento en formato abierto sencillo para representar datos en forma de tabla, en las que las columnas se separan por comas (o punto y coma en donde la coma es el separador decimal: España, Francia, Italia, etc.) y las filas por saltos de línea. Los campos que contengan una coma, un salto de línea o una comilla doble deben ser encerrados entre comillas dobles. Aprovechando el ejemplo anterior, la representación de dicha línea en CSV sería la siguiente:

21,10000505101010151505,3,DCL

C4.5 Para un conjunto de datos denominado “prueba”, existirían tres ficheros: prueba.data, prueba.test y prueba.names (muchos conjuntos de datos no proporcionan los ficheros .test). El fichero .names describe el conjunto de datos, y los ficheros .data y .test contienen muestras que componen el conjunto de datos, que pueden ser utilizadas como conjunto de entrenamiento y de testeo, respectivamente. Los ficheros .name contienen una serie de entradas que describen las clases, atributos y valores del conjunto de datos. Cada entrada es terminada con un punto, que puede ser omitido si se trata del último elemento. La primera entrada es una lista de los nombres de las clases separadas por comas (y terminadas por un punto). Cada línea sucesiva define un atributo, en el orden en el que aparecen en los ficheros .data y .test con el siguiente formato:

nombre-atributo: tipo-atributo .

El nombre-atributo es un identificador seguido de dos puntos, y a continuación el tipo del atributo que debe ser de uno de los listados a continuación:

- continuous: representa un atributo continuo.
- discrete <n>: representa un valor discreto que puede tomar ‘n’ valores.
- <lista de identificadores>: Es un atributo discreto con los valores enumerados. Cada identificador debe ir separado por comas.
- ignore: el atributo no será tomado en consideración.

Capítulo 2. Método de sobre-muestreo extendido basado en la red neuronal de Sanger (SNEOM)

Para representar el ejemplo anterior mediante el formato C4.5, habría que definir al menos dos ficheros:

fichero “.names”:

Sano, DCL

mec: discrete <36>

barthel: discrete

educacion: discrete <7>

fichero “.data”:

21,10000505101010151505,3,DCL

ARFF El formato de ficheros ARFF es propio de WEKA [I.H. Witten, 2005]. Se divide en tres partes: “@relation”, “@attribute” y “@data”:

- @relation <relation-name>: Todo fichero ARFF debe comenzar con esta declaración en su primera línea (no se pueden dejar líneas en blanco al principio). ¡relation-name! será una cadena de caracteres y si contiene espacios habrá que ponerla entre comillas.
- @attribute <attribute-name> <datatype>: En esta sección se incluirá una línea por cada atributo (o columna) que se vaya a introducir en nuestro conjunto de datos, indicando su nombre y el tipo de dato. Con <attribute-name> se indica el nombre del atributo, que debe comenzar por una letra y si contiene espacios tendrá que estar entrecomillado. Con <datatype> se indica el tipo de dato para este atributo (o columna) que puede ser:
 - numeric (numérico)
 - string (texto)
 - date [<date-format>] (fecha). En <date-format> se indicará el formato de la fecha, que será del tipo yyyy-MM-dd'T'HH:mm:ss.
 - <nominal-specification>. Estos son tipos de datos definidos por nosotros mismos y que pueden tomar una serie de valores que indicamos.
- @data: En esta sección incluiremos los datos propiamente dichos. Separaremos cada columna por comas y todas filas deberán tener el mismo número de columnas, número que coincide con el de declaraciones @attribute que añadimos en la sección anterior. Si no se dispone de algún dato, se colocará un signo de interrogación (?) en su lugar. El separador de decimales tiene que ser obligatoriamente el punto y las cadenas de tipo string tienen que estar entre comillas simples.

Volviendo al ejemplo anterior, el fichero ARFF que lo representaría sería el siguiente:

```
@relation PacientesSanosyDCL
```

```
@attribute diagnostico (SANO, DCL)
```

```

@attribute mec integer
@attribute barthel integer
@attribute educacion integer
@data
21,10000505101010151505,3,DCL

```

2.3.6. Vistas proporcionadas por la interfaz

Cuando ejecutamos SNEOM-GUI, la ventana principal de la aplicación es similar a la que se puede observar en la figura 2.10. Inicialmente, la aplicación nos sitúa en la pestaña *Sanger network*, desde la que se pueden configurar los parámetros de la red neuronal de Sanger.

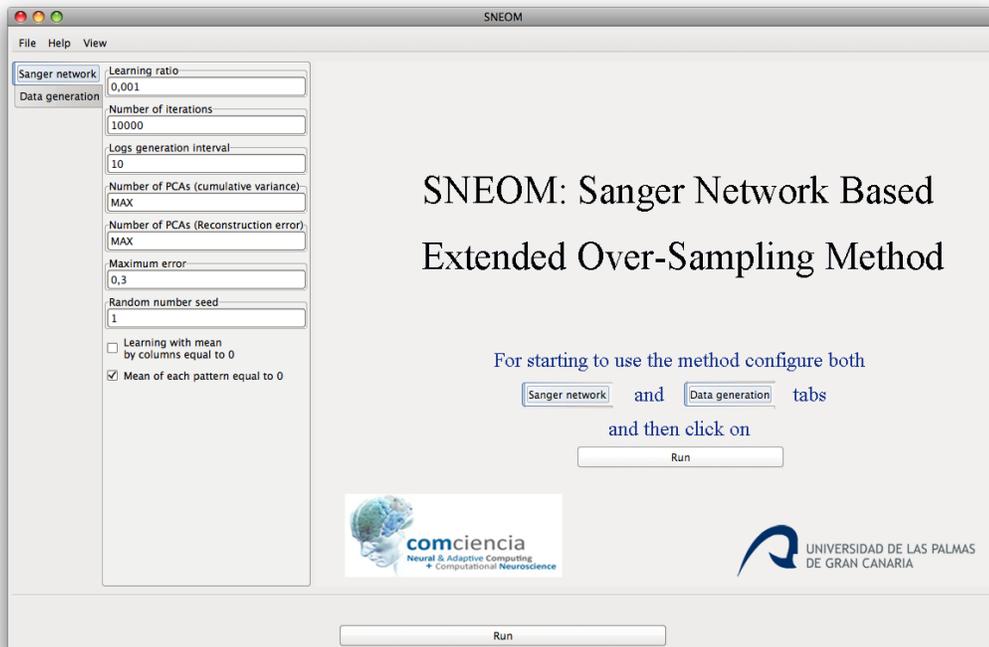


Figura 2.10: Vista de la configuración paramétrica de la red neuronal de Sanger.

En orden a cómo aparecen en pantalla, los parámetros para configurar la red son los siguientes:

- *Learning Ratio*. Ratio inicial de aprendizaje de la red de Sanger.
- *Number of iterations*. Número máximo de iteraciones o ciclos de la red.
- *Logs generation interval*. Intervalo de generación de logs.
- *Number of PCAs (cumulative variance)*. Número de componentes para el cálculo de la varianza acumulada.

- *Number of PCAs (reconstruction error)*. Número de componentes para el cálculo del promedio del error cuadrático (mse) de reconstrucción.
- *Maximum error*. Máximo error admitido.
- *Random number seed*. Valor de la semilla para números aleatorios.
- *Learning with mean by columns equal to 0*. Indica si se quiere aprendizaje con media por columnas a 0.
- *Mean of each pattern equal to 0*. Indica si se quiere la media de cada patrón a 0.

Haciendo click en la pestaña *Data Generation* aparecerá una ventana parecida a la figura 2.11, mediante la cual podremos configurar el método de sobre-muestreo.

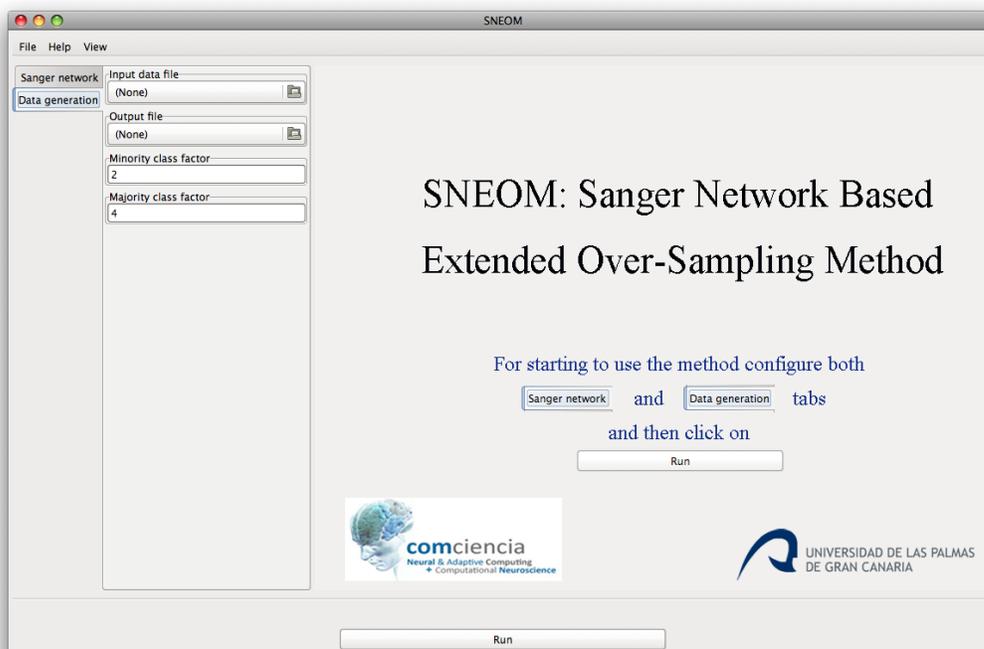


Figura 2.11: Vista de la configuración paramétrica del método sobre-muestreo.

Para configurar el método hay que indicar los parámetros listados a continuación:

- *Input data file*. Fichero de patrones de entrada.
- *Output file*. Fichero de patrones de salida, formado por los patrones de entrada más los generados mediante el método de sobre-muestreo.
- *Minority class factor*. Factor de la clase minoritaria.
- *Majority class factor*. Factor de la clase mayoritaria.

Tras configurar ambas pestañas, SNEOM se encontrará correctamente parametrizado, por lo que sólo es necesario hacer click en el botón *Run*, situado en la parte inferior de la ventana. Una vez se comience a ejecutar el método de sobre-muestreo, aparecerá una pequeña barra de progreso en el centro de la ventana de la aplicación, indicando que el método está ejecutándose y una estimación de cuánto falta para que finalice su ejecución (figura 2.12).

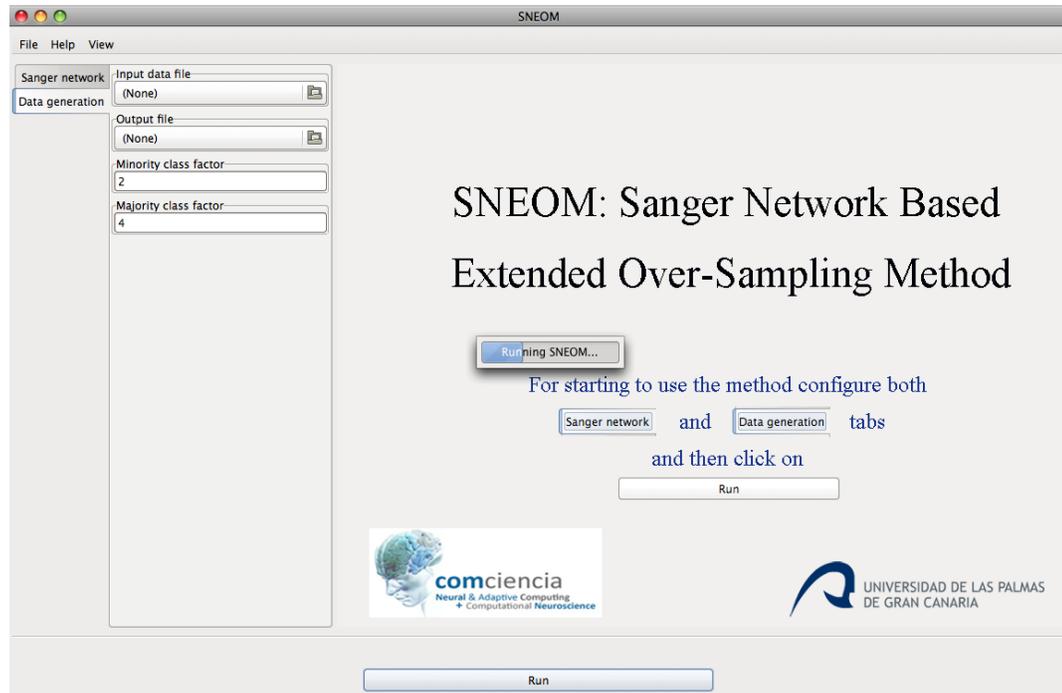


Figura 2.12: Vista de la ejecución del método SNEOM.

Una vez se complete la ejecución del método, se mostrarán diversas pestañas para visualizar todo el proceso del método de sobre-muestreo. En la imagen 2.13 puede observarse la pestaña *Input Data*, en la cual se puede visualizar el conjunto de entrada en dos dimensiones. Mediante el selector que se encuentra en la parte superior derecha de la ventana, puede seleccionarse cualquier combinación de dos variables del conjunto de entrada para ser visualizadas desde esta pestaña.

Análogamente, si se quisiera visualizar el conjunto de entrada en tres dimensiones, se puede hacer seleccionado la opción *3D* del menú combo. De esa forma, el selector que comentábamos anteriormente mostrará tuplas de tres variables, para poder seleccionar cualquier de ellas, representando así tres de las variables del conjunto, como puede apreciarse en la figura 2.14. Desde el resto de pestañas pueden observarse las imágenes relativas a la proyección del conjunto de entrada en el espacio PCA, las áreas de aceptación centradas en cada punto del espacio proyectado, y la reconstrucción desde el espacio proyectado al espacio de entrada, mostradas en la sección 2.3.2.

Capítulo 2. Método de sobre-muestreo extendido basado en la red neuronal de Sanger (SNEOM)

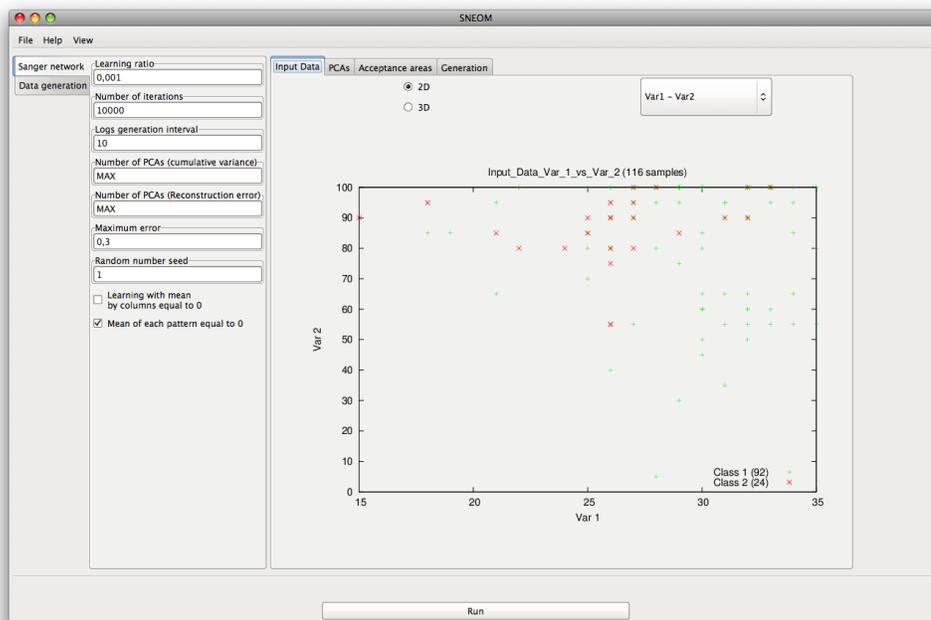


Figura 2.13: Vista bidimensional de los datos de entrada.

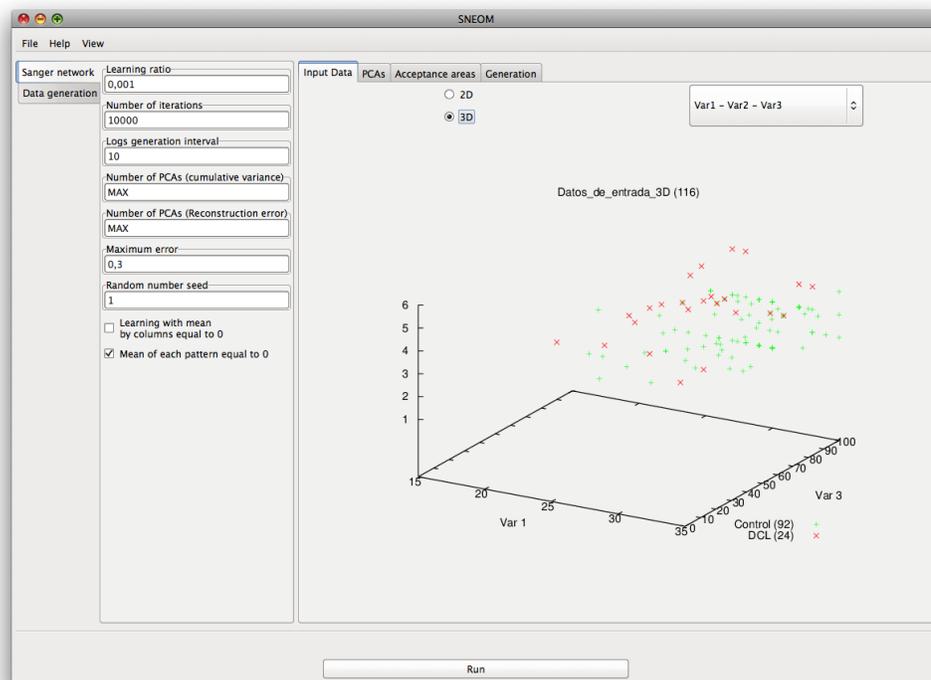


Figura 2.14: Vista tridimensional de los datos de entrada.

Capítulo 3

Sistema inteligente de ayuda al diagnóstico del DCL basado en computación neuronal

3.1. Análisis de algunos criterios clínicos para la detección del DCL

Presentamos un análisis de la influencia de diferentes instrumentos de valoración cognitiva y funcional para la detección del DCL con el objetivo de obtener un conjunto reducido de criterios relevantes para dicho diagnóstico, utilizando para ello el enfoque de la computación neuronal. Emplearemos el mismo tipo de arquitectura neuronal tanto para la selección de características relevantes como para diseñar el sistema de ayuda al diagnóstico del DCL, redes neuronales híbridas, concretamente la red neuronal Counterpropagation (CPN). Trabajaremos con un conjunto de datos extraído de la base de datos de la Iniciativa de Neuroimagen de la Enfermedad de Alzheimer (en inglés, *Alzheimer's Disease Neuroimaging Initiative*, ADNI) [ADNI, 2004]. Para poder hacer uso de los datos obtenidos de ADNI, se han elaborado un conjunto de funciones en Matlab que permitieron leer los ficheros en formato CSV proporcionados por ADNI, almacenarlos en formato matricial, y posteriormente acceder a los mismos como si de una consulta a una base de datos se tratase, seleccionando campos o columnas de interés de diferentes ficheros y combinándolos en uno sólo por medio de los identificadores de cada registro.

Para cada sujeto se seleccionaron cinco características: La puntuación del test MMSE, la Escala de Deterioro Global (GDS), el cuestionario de las actividades funcionales (FAQ), la edad y años de educación de cada paciente. Se tomaron todas las posibles combinaciones entre al menos un instrumento de valoración y el resto de variables, y se construyeron 24 subconjuntos de datos. En total se entrenaron 6624 redes CPN y se seleccionaron las redes que mejores resultados de clasificación proporcionaron para cada subconjunto de datos. Los resultados, mostrados en la tabla 3.1, indicaron que la puntuación del test MMSE en combinación con la del FAQ y la edad del paciente proporcionaron la mejor precisión (94.05 %) y sensibilidad (90 %), que es la capacidad de reconocer satisfactoriamente a un paciente con DCL. Este resultado concuerda con los resultados obtenidos

Capítulo 3. Sistema inteligente de ayuda al diagnóstico del DCL basado en computación neuronal

cuando se analizaron los conjuntos de datos de dos variables, donde la combinación del MMSE y del FAQ consiguieron las mejores tasas de sensibilidad (89.29 % y 73.73 %, respectivamente). Por otra parte, el conjunto de datos compuesto por el valor del test FAQ y los años de educación del paciente, alcanzó una especificidad del 100 %, es decir, permitió clasificar correctamente la totalidad de sujetos sanos. Sin embargo, cuando las variables MMSE, FAQ, edad y años de educación del paciente fueron suministradas a la CPN, las tasas de rendimiento no fueron incrementadas con respecto a los resultados previamente discutidos. En este caso, la precisión obtenida fue de 89.29 %, la tasa de sensibilidad fue 80 % y la especificidad fue igual a 94.44 %.

	Precisión (%)	Sensibilidad (%)	Especificidad (%)
MMSE+FAQ	89.29	73.73	98.15
MMSE+GDS	76.19	63.33	81.48
MMSE+EDUCATION	76.19	70	79.63
MMSE+Age	84.52	73.33	90.74
GDS+FAQ	89.29	86.67	90.74
GDS+EDUCATION	66.67	23.33	90.74
GDS+Age	69.048	60	74.07
FAQ+EDUCATION	89.29	70	100
FAQ+Age	88.10	70	98.15
MMSE+EDUCATION+FAQ	86.91	76.67	92.59
MMSE+EDUCATION+GDS	79.76	70	85.19
MMSE+AGE+FAQ*	94.05	90	96.30
MMSE+AGE+GDS	84.52	66.67	94
MMSE+AGE+EDUCATION	78.571	56.67	90.74
MMSE+GDS+FAQ	88.10	83.33	90.74
GDS+EDUCATION+FAQ	83.33	60	96.30
GDS+AGE+FAQ	89.29	83.33	92.59
GDS+AGE+EDUCATION	66.67	16.67	94.44
MMSE+EDUCATION+FAQ+GDS	89.29	80	94.44
MMSE+AGE+FAQ+GDS	90.48	80	96.30
MMSE+FAQ+AGE+EDUCATION	89.29	80	94.44
MMSE+GDS+AGE+EDUCATION	82.14	60	94.44
GDS+EDUCATION+FAQ+AGE	82.14	60	94.44
MMSE+EDUCATION+FAQ+GDS+Age	86.91	73.33	94.44

Tabla 3.1: Métricas de rendimiento obtenidas para CPNs entrenadas con diferentes subconjuntos de variables.

A la vista de los resultados obtenidos, la combinación de un instrumento de cribado cognitivo como el MMSE con uno funcional como el FAQ pueden proporcionar altas sensibilidades para detectar el DCL. Además, cuando se suministra la edad o el nivel de educación del paciente es

posible detectar fácilmente a los sujetos sanos. Por tanto, el uso combinado de esta información clínica podría permitir llevar a cabo una fiel discriminación entre sujetos sanos y sujetos con DCL. Es por ello, que basándonos en los resultados obtenidos, en las secciones siguientes se utilizarán variables análogas a estas comentadas, que pertenecen a un conjunto de pacientes de los E.E.U.U., pero de aplicación en España, puesto que la base de datos seleccionada corresponde a un conjunto de pacientes diagnosticado por un médico geriatra español, que utiliza instrumentos de valoración estandarizados y ampliamente utilizados en nuestro país.

3.2. Cuerpo de datos

Los datos utilizados para todos los experimentos fueron extraídos del entorno clínico EDEVITALZH [Araujo et al., 2004]. El cuerpo de datos fue conformado con consultas de un grupo de pacientes de la Unidad de Atención a la Dependencia de Santander (Cantabria, España). Para diagnosticar a estos sujetos de forma precisa, el facultativo tomó en consideración para cada uno de ellos los siguientes parámetros clínicos:

- *El proceso de anamnesis.* La anamnesis es la reunión de datos subjetivos, relativos a un paciente, que comprenden antecedentes familiares y personales, signos y síntomas que experimenta en su enfermedad, experiencias y, en particular, recuerdos, que se usan para analizar su situación clínica. Es un historial médico que puede proporcionar información relevante para diagnosticar posibles enfermedades.
- *Las exploraciones física y neurológica.* Corresponden al conjunto de evaluaciones que permiten obtener información acerca del estado de salud de un paciente con los medios de la práctica clínica habitual.
- *Las sintomatologías cognitiva, no cognitiva, funcional y otras.* Son todas aquellas valoraciones que permiten hallar alteraciones en la cognición, en la independencia funcional o encontrar otras posibles sintomatologías que pueda presentar el paciente.
- *Hábitos tóxicos.* Constituyen todas aquellas sustancias dañinas para la salud que el paciente consume regularmente.
- *Enfermedades intercurrentes.* Son posibles enfermedades que se presentan en el curso de otra enfermedad sin que tengan relación directa con la primera.
- *Pruebas complementarias.* Se tratan de pruebas diagnósticas que solicita el médico y que se realiza al paciente tras una anamnesis y diversas exploraciones, para confirmar o descartar un diagnóstico clínico. Algunos ejemplos de pruebas complementarias son los análisis clínicos, pruebas de imagen, electrogramas, etc.
- *Prescripción de fármacos.* Relación de todos los medicamentos que el paciente consume o ha consumido recientemente.

Capítulo 3. Sistema inteligente de ayuda al diagnóstico del DCL basado en computación neuronal

Consultas de pacientes con DCL (27 consultas)									
IdConsulta	IdPaciente	MMSE	Barthel	Exploraciones Neurológicas	Deterioro Cognitivo	Demencia	IdMedico	Nombre	Apellidos
1	1	17	No	NNNBNNNNNNNNNNNNNN	1	0	5	Carmen	Alastuey Gimenez
27	17	16	No	NNNBNNNNNNNNNNNNNN	1	0	5	Carmen	Alastuey Gimenez
68	32	21	No	NNNNNNNNNNNNNNNNNN	1	0	5	Carmen	Alastuey Gimenez
90	39	20	No		1	0	5	Carmen	Alastuey Gimenez
104	45	17	No	NNNNNNNNNNNNNNNNNN	1	0	5	Carmen	Alastuey Gimenez
109	46	21	No	NNNNNNNNNNNNNNNNNN	1	0	5	Carmen	Alastuey Gimenez

Figura 3.1: Vista del componente integrado en EDEVITALZH para visualizar el conjunto de datos de pacientes con DCL.

El cuerpo de datos está formado por 92 consultas relativas a 39 sujetos sanos y 24 sujetos con DCL. En la imagen 3.1 puede observarse una captura del componente integrado en EDEVITALZH para visualizar desde la Web la relación de pacientes con DCL y sanos contenidos en la base de datos de este entorno clínico virtual. Para el desarrollo de este módulo se ha utilizado la librería Flexigrid [Flexigrid, 2010] para JQuery [JQuery, 2009].

Todos los sujetos fueron diagnosticados por un médico geriatra utilizando los criterios de Petersen [Petersen, 2004]. En [Petersen, 2004] Petersen propone un conjunto de criterios para el tipo más extendido de DCL, el DCL amnésico (a-DCL), que incluye: quejas subjetivas de memoria normalmente corroboradas por un informante, déficit de memoria objetivo para la edad, funciones cognitivas preservadas esencialmente, actividades funcionales en gran parte intactas y ausencia de demencia. La estructura del cuerpo de datos incluye: el identificador del paciente, las puntuaciones obtenidas en los dos instrumentos clínicos más utilizados en neurología y geriatría, el Mini-Examen Cognitivo (MEC) y el test funcional de Barthel junto al nivel educativo del paciente. A continuación se describen los test neuropsicológicos utilizados:

- El Mini-Examen Cognitivo de Lobo et al. (MEC) [Lobo et al., 1999] es la versión española del test MMSE, el instrumento estandarizado más extendido, empleado y citado para valorar la función cognitiva. Consiste en un conjunto de cuestiones simples y cortas que permiten la evaluación rápida de diversas áreas cognitivas: orientación, fijación, cálculo y atención, memoria, lenguaje, lectura, escritura, y habilidades visoconstructivas. Existen dos versiones del test MEC, la primera de ellas que puntúa desde 0 hasta 30 puntos, y la segunda y más utilizada en la actualidad, de hasta 35 puntos. Cuanto más alta sea la puntuación obtenida, mejor estado presentan las funciones cognitivas.

- El índice de Barthel [Mahoney and Barthel, 1965] evalúa las funciones puras de las actividades básicas de la vida diaria: continencia, ir al baño, vestirse, asearse, comer, arreglarse y trasladarse. Es considerado un instrumento observacional y una guía objetiva del curso de la enfermedad crónica, como una ayuda para estudiar el proceso de envejecimiento en la rehabilitación. Su rango de puntuación oscila entre 0 (totalmente dependiente) y 100 (independiente).

	Sanos	DCL
Sujetos (s)	39 (68.42 %)	18 (31.58 %)
Consultas (c)	92 (79.31 %)	24 (20.69 %)
<i>Edad (años)</i>		
Media \pm Desv. Est.	80.95 \pm 5.85	80.17 \pm 6.36
Rango	66 - 96	58 - 87
<i>Nivel educativo (s / c)</i>		
Analfabeto	2 / 5	3 / 4
Leer y Escribir	7 / 18	0 / 0
Estudios mínimos	8 / 19	3 / 3
Educación primaria	19 / 41	6 / 10
Educación secundaria	2 / 5	3 / 4
Estudios universitarios	1 / 4	3 / 3
<i>MEC</i>		
Media \pm Desv. Est.	29.50 \pm 3.51	26.04 \pm 4.14
Rango	18 - 35	15 - 33
<i>Barthel</i>		
Media \pm Desv. Est.	84.24 \pm 21.00	86.46 \pm 12.02
Rango	5 - 100	55 - 100

Tabla 3.2: Características demográficas, resultados de los tests neuropsicológicos y nivel educativo de los sujetos.

La tabla 3.2 proporciona información acerca de las características demográficas, los resultados de los tests neuropsicológicos y el nivel educativo de los sujetos que forman parte del estudio. La desproporción entre el número de consultas de cada clase (92 consultas relativas a sujetos sanos frente a 24 consultas de pacientes con DCL), denota que se trata de un problema de clases no balanceadas. Como se comentó previamente en la sección 2.3.1, los conjuntos de datos no balanceados tienen una fuerte influencia sobre los clasificadores, causando una reducción significativa del rendimiento de estos debido al sesgo en la distribución de clases. Esta situación implica un

problema crítico en medicina debido a que los costes de clasificar una muestra minoritaria como perteneciente a la clase mayoritaria son especialmente altos. En ese caso, un sujeto que realmente padece la enfermedad podría no ser tratado de manera adecuada desde el punto de vista médico. Esto supone un alto coste debido a que no se permitiría tratar ese declive cognitivo que en la mayoría de los casos desemboca en la EA o en otra demencia, influyendo de forma clara en la calidad de vida del paciente. Otro inconveniente surgido durante la fase de diseño del sistema artificial inteligente para la ayuda al diagnóstico del DCL desarrollado en este trabajo, es el tamaño del cuerpo de datos, donde el número de muestras es universalmente pequeño (menor que 100) para este tipo de patologías neurológicas. Esta suele ser una situación muy extendida en este ámbito de trabajo [Lonie et al., 2009], a excepción de muy pocos estudios. Estos se encuentran limitados a una fase de entrenamiento del modelo neuronal adecuada así como a considerar el error de generalización de manera precisa. Por esta razón resulta necesario utilizar un método que permita conseguir estos objetivos de manera eficiente. Es por ello, que para enfrentar estos problemas se ha utilizado el método SNEOM, descrito en el capítulo 2.

3.3. Sistema de ayuda a la detección del DCL basado en la red neuronal Counterpropagation

El sistema de ayuda a la detección del DCL se ha diseñado basado en una arquitectura neuronal híbrida, la CPN. Se ha analizado su bondad comparándola con otra arquitectura neuronal supervisada, como es el perceptrón multicapa entrenado con el algoritmo de aprendizaje *Backpropagation*, así como el diagnóstico realizado por un facultativo, proporcionando la CPN propuesta los mejores resultados de detección.

El sistema híbrido basado en una arquitectura CPN ha sido implementado en Matlab utilizando la versión 3.3 de la *Kohonen and CP-ANN toolbox* [Ballabio et al., 2009]. La configuración de la red ha sido la siguiente:

Parámetro	Valor
Dimension del mapa	10
Topología	Cuadrada
Límites del mapa	Normal
Numero de ciclos	100
Inicialización de los pesos	autovalores
Tasa de aprendizaje inicial	1

Tabla 3.3: Parámetros de configuración del sistema basado en CPN.

Con el fin de poder llevar a cabo una detección lo más precisa posible del DCL, a cada clasificador se le suministró dos conjuntos de datos balanceados utilizando SNEOM (ver capítulo 2), uno generado mediante la técnica de sobre-muestreo clásica y el otro mediante la técnica de

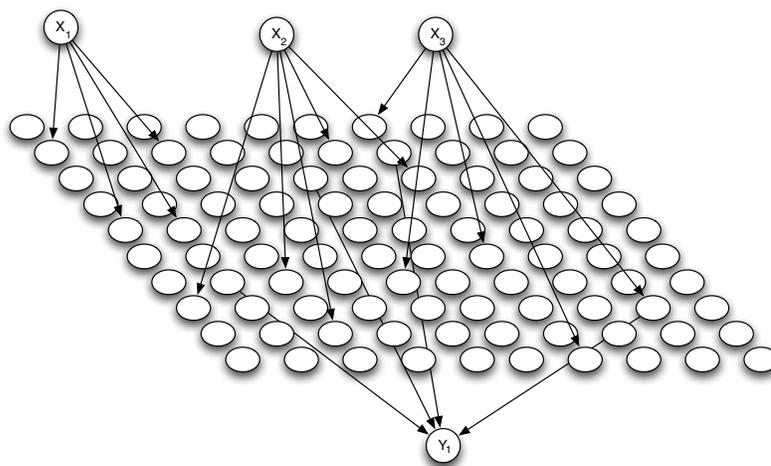


Figura 3.2: Diagrama estructural del sistema basado en CPN.

sobre-muestreo extendido. Esto permitió aliviar el efecto negativo que los conjuntos de entrenamiento no balanceados producen sobre los clasificadores, y comparar así el rendimiento de ambos clasificadores entrenados con dos técnicas diferentes de sobre-muestreo. Para el primer nuevo conjunto de datos (Conjunto OS), la cantidad de muestras minoritarias fue incrementada 4 veces. El segundo conjunto balanceado (Conjunto EOS) fue generado incrementando 2 veces la cantidad de muestras mayoritarias y 8 veces la cantidad de muestras minoritarias. Ambos conjuntos de datos presentaron una representación equilibrada de muestras de cada clase. Los datos fueron divididos en tres conjuntos: conjunto de entrenamiento, conjunto de test (para refinar la configuración del clasificador) y conjunto de validación (para validar la bondad del clasificador entrenado). El reparto de muestras para cada conjunto fue el siguiente:

- Conjunto OS (entrenamiento y test): 92 muestras de pacientes sanos y 93 muestras de pacientes con DCL.
- Conjunto EOS (entrenamiento y test): 184 muestras de pacientes sanos y 192 muestras de pacientes con DCL.
- Conjunto de validación: 51 muestras de pacientes sanos y 31 muestras de pacientes con DCL.

3.4. Análisis comparativo de los resultados

En esta sección se analizará la bondad de los resultados obtenidos y lo adecuado de utilizar el sistema propuesto en este trabajo, comparándolo con un MLP y con el facultativo. El MLP fue entrenado con el algoritmo de aprendizaje *backpropagation* lanzado desde WEKA [Repository, 2007]. La configuración del MLP ha sido la siguiente:

La tabla 3.5 contiene los resultados obtenidos utilizando nuestro sistema basado en CPN y el MLP, empleando tanto los entornos de información OS como EOS, y las predicciones

Parámetro	Valor
Nº Neuronas en la capa oculta	4
Tasa de aprendizaje	0.3
Momentum	0.2
Número de ciclos	500
Numero de ciclos	100
Inicialización de los pesos	aleatoria

Tabla 3.4: Parámetros de configuración del sistema basado en MLP.

del médico geriatra. A cada clasificador y al facultativo se les proporcionó el mismo conjunto de validación. El rendimiento del sistema basado en CPN ha sido analizado empleando diferentes medidas frecuentemente utilizadas en medicina. Esas mismas métricas fueron utilizadas para comparar el rendimiento de la arquitectura CPN con los obtenidos por otros sistemas de predicción. Específicamente, las métricas utilizadas han sido: precisión, sensibilidad y especificidad, que han sido descritas en la sección 2.3.3.

	Precisión (%)	Sensibilidad (%)	Especificidad (%)
CPN _{OS}	87.80	77.42	94.12
CPN _{EOS}	91.46	87.10	94.12
MLP _{OS}	86.59	74.19	94.12
MLP _{EOS}	86.59	87.10	61.97
Facultativo	64.43	00.06	100

Tabla 3.5: Resultados de clasificación para diferentes métricas utilizando clasificadores basados en CPN frente a los basados en MLP y las predicciones del facultativo.

Los clasificadores entrenados con el conjunto de datos EOS proporcionaron mejores resultados que los entrenados utilizando el conjunto de datos OS. Esto recalca la necesidad de tener un conjunto de entrenamiento no sólo representativo para todas las clases implicadas en el proceso de clasificación, sino también amplio, con un enorme número de muestras de cada clase, especialmente para los clasificadores basados en redes neuronales artificiales. Utilizando el conjunto de datos EOS, los clasificadores basados en CPN obtuvieron mejores resultados comparándolos con los basados en MLP. Esto se justifica por el hecho de que la primera etapa de la arquitectura CPN es un proceso no supervisado y este aumento del entorno de información referente al incremento del número de muestras de ambas clases proporciona una capacidad discriminatoria alta, específicamente para la clase de interés, DCL, cuya tasa de acierto está representada por los valores de sensibilidad. Ambos sistemas artificiales proporcionaron resultados satisfactorios, pero el sistema basado en CPN obtuvo mejor rendimiento que el basado en MLP, incluso cuando el primero de ellos fue entrenado con el conjunto de datos OS, indicando que el modelo neuronal híbrido - no supervisado

y supervisado - es más apropiado que una arquitectura retro-alimentada para ser integrado en un sistema de diagnóstico asistido por ordenador, principalmente basado en tests neuropsicológicos practicados al paciente y el nivel educativo del mismo.

En la imagen 3.3 se pueden observar las curvas ROC para los clasificadores CPN y MLP entrenados cada uno de ellos con los conjuntos OS y EOS (figura 3.3(a)) y las curvas envolventes convexas de los mismos (figura 3.3(b)). En ambas gráficas aparece dibujada la curva del clasificador aleatorio, es decir, aquel cuya precisión para detectar ambas clases es de un 50 %.

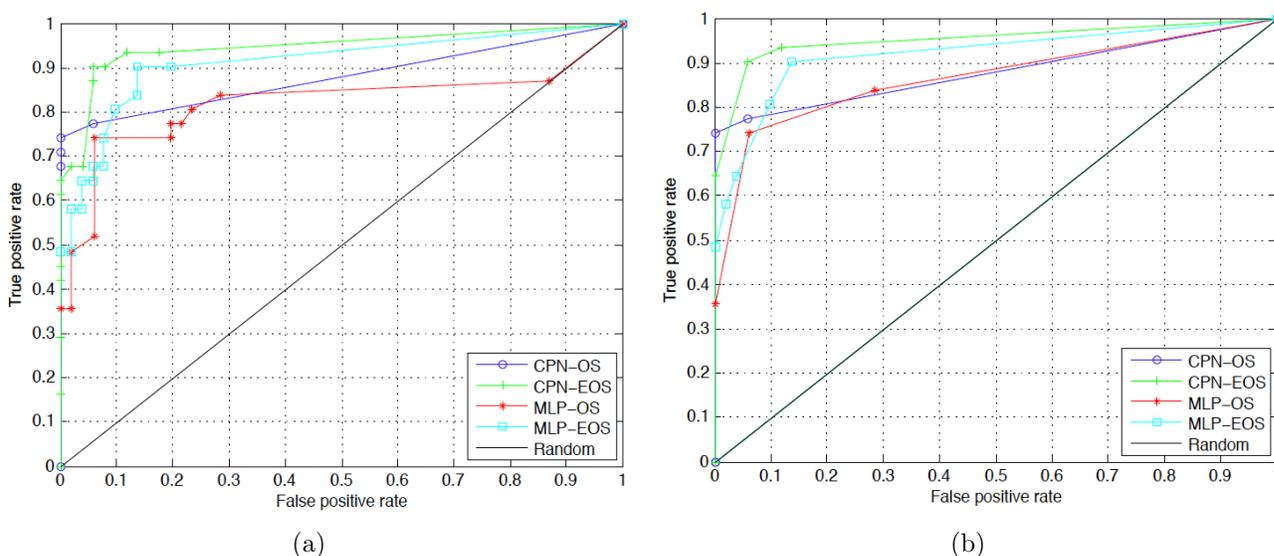


Figura 3.3: a) Curvas ROC de los clasificadores basados en computación neuronal para discriminar entre pacientes de DCL de pacientes sanos. b) Curva envolvente convexa de los clasificadores basados en computación neuronal para discriminar entre pacientes de DCL de pacientes sanos.

Los resultados fueron siempre mejorados por los sistemas basados en computación neuronal comparado con el facultativo. Este último presenta una muy baja sensibilidad (00.06 %) y la mayor especificidad (100 %). Estos resultados, relativos al rendimiento discriminatorio del facultativo, concuerdan con otros estudios previos donde la sensibilidad obtenida fue muy baja utilizando el test MMSE, con algunas pocas excepciones [Callahan et al., 2002], oscilando entre 1 % [Sager et al., 2006] y 49 % [Ravaglia et al., 2005]. De la misma forma, la especificidad clínica, 100 %, se encuentra dentro del rango de la especificidad del MMSE para el DCL, en referencia a la detección de sujetos sanos, cuyas tasas de acierto varían entre 85.5 % y 100 % [Araujo et al., 2010]. A la vista de estos resultados se puede inferir que el facultativo es incapaz de discriminar entre las dos clases utilizando sólo los test MEC y Barthel y el nivel educativo.

El sistema basado en CPN es capaz de diferenciar entre sujetos con DCL y sujetos sanos utilizando sólo la puntuación final de dos test neuropsicológicos y el nivel educativo del paciente, obteniendo una precisión de 91.46 %, con una alta sensibilidad (87.10 %) y especificidad (94.12 %). Además, este sistema es adecuado para utilizar en atención primaria, donde el uso de instrumentos de diagnóstico está limitada y las medidas más útiles son algunos test cognitivos y funcionales,

Capítulo 3. Sistema inteligente de ayuda al diagnóstico del DCL basado en computación neuronal

esencialmente MEC ó MMSE, siendo posible también practicar el test de Barthel. Atendiendo a la complejidad de los datos, con una difícil característica para diagnosticar tal como el analfabetismo, los clasificadores neuronales han realizado adecuadamente la clasificación de los sujetos analfabetos, concretamente los basados en la arquitectura CPN, logrando una alta especificidad, que sugiere que esta red es capaz de diferenciar entre sujetos analfabetos y sujetos con déficit cognitivo.

Capítulo 4

Conclusiones y trabajos futuros

Se puede concluir, de una forma global, que las principales aportaciones de este trabajo de Fin de Máster son los propios objetivos del mismo, los cuales estimamos han sido alcanzados en toda su extensión.

En este trabajo se ha presentado un sistema de diagnóstico inteligente para el DCL basado en la arquitectura neuronal Counterpropagation con un esquema de fusión de datos, en el que se han combinado valoraciones cognitivas y funcionales aplicables en atención primaria y especializada e información demográfica del paciente. Esta propuesta puede ser un paso más allá para desarrollar herramientas computacionales inteligentes apropiadas y fiables para detectar el DCL y para evaluar la progresión de la enfermedad, utilizando esencialmente medidas neuropsicológicas y funcionales, eliminando los sesgos dependientes del experto clínico. Mediante un análisis de características realizado sobre un conjunto de criterios clínicos, hemos mostrado también la conveniencia de utilizar instrumentos de cribado cognitivo tal como los tests MMSE ó MEC en combinación con instrumentos funcionales como el FAQ ó el índice de Barthel, proporcionando altas sensibilidades para detectar correctamente la enfermedad.

Además, el sistema propuesto es capaz de abordar problemas de clasificación no balanceada, muy comunes en medicina, mediante la aplicación de un nuevo método de sobre-muestreo extendido basado en la red de Sanger, SNEOM. Este método es capaz de procesar datos incompletos o *missing* y permite generar muestras tanto para la clase minoritaria como para la mayoritaria. Esto se traduce en que este método de sobre-muestreo no sólo es capaz de balancear cualquier conjunto de datos sea cual sea el nivel de desequilibrio entre ambas clases del conjunto de datos, sino que además es capaz de incrementar conjuntos de datos significativamente pequeños o con pocas muestras.

La Counterpropagation entrenada con el conjunto de datos creado a partir del método de sobre-muestreo extendido proporcionó mejores resultados que otros clasificadores basados en arquitecturas neuronales supervisadas como el MLP. Este sistema ha demostrado, además, un mejor rendimiento que el facultativo, aún tratándose de un médico especialista en geriatría. Por tanto, esta arquitectura neuronal modular e híbrida ha mostrado su capacidad para discriminar entre sujetos con DCL de sujetos sanos con una gran sensibilidad y especificidad, remarcando el hecho de entrenar estas redes con un conjunto de datos balanceado y diverso.

Además, se ha diseñado un sistema inteligente de ayuda a la detección del DCL integrable

Capítulo 4. Conclusiones y trabajos futuros

en EDEVITALZH, que es un entorno clínico virtual, desarrollado en el grupo de investigación *Computación Neuronal y Adaptativa + Neurociencia Computacional (COMCIENCIA)*, centrado en el diagnóstico y pronóstico del déficit cognitivo, aplicable a enfermedades neurodegenerativas tales como la EA y otras demencias. Este entorno propio de tele-medicina, propicia la interactividad *en línea* de los clínicos, permitiendo la comunicación entre diferentes facultativos (inter-consultas). Mediante el sistema inteligente que está siendo implementado e integrado en EDEVITALZH, cualquier facultativo podrá solicitar un diagnóstico al sistema, de forma remota, siendo de gran ayuda como herramienta de detección del DCL en atención primaria.

En resumen, el sistema propuesto basado en Counterpropagation obtuvo unas tasas de acierto elevadas, sugiriendo este esquema computacional como una aproximación adecuada para construir sistemas de ayuda al diagnóstico para el diagnóstico diferencial del DCL y de la EA. El sistema propuesto basado en Counterpropagation supone una contribución significativa en el diagnóstico médico de enfermedades neuro-degenerativas debido a que podría resultar útil para el diseño de nuevas estrategias fármaco-terapéuticas, con el objetivo de retardar la progresión hacia una demencia o Enfermedad de Alzheimer y por ende, mejorar la calidad de vida del paciente. Asimismo, el sistema propuesto permitiría refinar los protocolos de diagnóstico del DCL extendiendo el estudio a otros posibles criterios diagnósticos para esta patología. Como tal, el uso de estos nuevos instrumentos podría aliviar el grado de infra-diagnosis existente actualmente en todas las áreas sanitarias, esencialmente en el ámbito de la atención primaria.

Finalmente se ha demostrado, una vez más, la aproximación de las RNAs, como una innovadora y potente herramienta de modelado para detectar déficit cognitivo y asistir al diagnóstico de enfermedades neuropsicológicas.

Además de las conclusiones previas, es importante resaltar, que este trabajo ha dejado abiertas varias líneas de trabajo, entre las que podemos mencionar:

- *Extender las capacidades del sistema de diagnóstico basado en CPN al diagnóstico diferencial del DCL, EA y sanos:* Tras los buenos resultados proporcionados en este trabajo por la arquitectura CPN como sistema computacional para detectar el DCL, resulta de interés extender sus capacidades al diagnóstico diferencial del DCL, EA y sanos o inclusive, al diagnóstico diferencial del DCL y otras demencias, en general frente a sanos.
- *Diseñar un protocolo clínico basado en unos criterios diagnósticos relevantes y estándares para el DCL que puedan usarse en cualquier ámbito sanitario.*
- *Desarrollo del sistema de diagnóstico como aplicación móvil:* El desarrollo del sistema propuesto basado en CPN podría integrarse como parte de una aplicación móvil que permita a los facultativos de atención primaria y especializada aprovecharse de las ventajas de un entorno clínico virtual portátil con capacidad de asistir el diagnóstico del DCL. Actualmente, este sistema de diagnóstico está siendo integrado en el entorno clínico virtual EDEVITALZH.
- *Extender la capacidad de generación de datos de SNEOM introduciendo distintos tipos de perturbaciones:* Además del ruido Gaussiano utilizado por el método SNEOM para perturbar las muestras originales, podrían implementarse nuevos métodos de filtrado, que puedan ser seleccionados como parámetro en la configuración de la generación de datos de la SNEOM-GUI.

- *Integrar el método SNEOM en WEKA:* En WEKA ya existen algunos filtros supervisados para la generación de muestras sintéticas como es el caso de SMOTE. Sería interesante implementar el algoritmo de SNEOM en este entorno de minería de datos y añadirlo a la batería de métodos de muestreo existentes.

Capítulo 4. Conclusiones y trabajos futuros

Lista de publicaciones y comunicaciones

A continuación se detallan las listas de publicaciones y comunicaciones obtenidas de los desarrollos realizados en este Trabajo de Fin de Máster.

Publicaciones

Martínez-García, J.M., Suárez-Araujo, C.P., García Báez P.: SNEOM: A Sanger network based extended over-sampling method. Application to imbalanced datasets. In: Huang T., Zeng Z., Li C., Leung Ch. (eds.) Neural Information Processing, Part IV. LNCS, vol. 7666, pp. 584-292. Springer, Heidelberg (2012)

Martinez Garcia, J.M.; Garcia Baez, P.; Perez del Pino, M.A.; Fernandez Viadero, C.; Suarez-Araujo, C.P. A Counterpropagation Network based system for screening of Mild Cognitive Impairment, IEEE Xplore, pp.67-72, 20-22 September (2012), doi: 10.1109/SISY.2012.6339488

Comunicaciones

José Manuel Martínez García, Patricio García Báez, Carlos Fernández Viadero, and Carmen Paz Suárez Araujo. Analysis of some clinical criteria for mild cognitive impairment screening using hybrid neural networks. In Fourteenth International Conference On Computer Aided Systems Theory (EUROCAST), Las Palmas de Gran Canaria, Spain, 10-15 February (2013)

J.M. Martinez-Garcia, C.P. Suarez-Araujo, P. Garcia Baez SNEOM: A Sanger network based extended over-sampling method. Application to imbalanced biomedical datasets. In 19th International Conference on Neural Information Processing (ICONIP2012), Doha, Qatar, 12-15 November (2012)

J.M. Martinez-Garcia, C.P. Suarez-Araujo, P. Garcia-Baez, M.A. Perez del Pino and C. Fernandez-Viadero. Artificial neural network system to assist the diagnosis of Mild Cognitive Impairment. Towards a cognitive screening system. In 22nd Alzheimer Europe Conference, Vienna, Austria, 4-6 October (2012)

J.M. Martinez-Garcia, P. Garcia Baez, M.A. Perez del Pino, C. Fernandez Viadero and C.P. Suarez Araujo. A Counterpropagation Network based System for Screening of Mild

Cognitive Impairment. In IEEE 10th Jubilee International Symposium on Intelligent Systems and Informatics (SISY), Subotica, Serbia, 20-22 September (2012)

Referencias

- [ADNI, 2004] ADNI (2004). <http://www.loni.ucla.edu>.
- [Araujo et al., 2010] Araujo, C., Báandez, P., and Viadero, C. (2010). Ganen: A new gating neural ensemble for automatic assessment of the severity level of dementia using neuropsychological tests. In *Broadband and Biomedical Communications (IB2Com), 2010 Fifth International Conference on*, pages 1–6.
- [Araujo et al., 2004] Araujo, C. P. S., Del Pino, M. A. P., Báez, P. G., and López, P. F. (2004). Clinical web environment to assist the diagnosis of alzheimer’s disease and other dementias. In *Proceedings of the 4th WSEAS International Conference on Applied Informatics and Communications, AIC’04*, pages 8:1–8:6, Stevens Point, Wisconsin, USA. World Scientific and Engineering Academy and Society (WSEAS).
- [Association, 2000] Association, A. P. (2000). *Diagnostic and Statistical Manual of Mental Disorders DSM-IV-TR, Fourth Edition*. American Psychiatric Pub.
- [Báez et al., 2009] Báez, P., Pino, M., Viadero, C., and Araujo, C. (2009). Artificial intelligent systems based on supervised humann for differential diagnosis of cognitive impairment: Towards a 4p-hcnds. In Cabestany, J., Sandoval, F., Prieto, A., and Corchado, J., editors, *Bio-Inspired Systems: Computational and Ambient Intelligence*, volume 5517 of *Lecture Notes in Computer Science*, pages 981–988. Springer Berlin Heidelberg.
- [Báez, 2005] Báez, P. G. (2005). *HUMANN: UNA NUEVA RED NEURONAL ARTIFICIAL ADAPTATIVA, NO SUPERVISADA, MODULAR Y JARÁRQUICA. APLICACIONES EN NEUROCIENCIA Y MEDIOAMBIENTE*. PhD thesis, Universidad de Las Palmas de Gran Canaria.
- [Ballabio et al., 2009] Ballabio, D., Consonni, V., and Todeschini, R. (2009). The kohonen and cp-ann toolbox: a collection of matlab modules for self organizing maps and counterpropagation artificial neural networks. *Chemometrics and Intelligent Laboratory Systems*, 98:115–122.
- [Barlow, 1989] Barlow, H. B. (1989). Unsupervised learning. *Neural Computation*, 1:295–311.
- [Batista et al., 2004] Batista, G. E. A. P. A., Prati, R. C., and Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.*, 6(1):20–29.

- [Buntine, 1994] Buntine, W. L. (1994). Operations for learning with graphical models. *Journal of Artificial Intelligence Research*, 2:159–225.
- [Buscema et al., 2007] Buscema, M., Rossini, P., Babiloni, C., and Grossi, E. (2007). The ifast model, a novel parallel nonlinear eeg analysis technique, distinguishes mild cognitive impairment and alzheimer’s disease patients with high degree of accuracy. *Artificial Intelligence in Medicine*, 40(2):127 – 141.
- [Callahan et al., 2002] Callahan, C. M., Unverzagt, F. W., Hui, S. L., Perkins, A. J., and Hendrie, H. C. (2002). Six-item screener to identify cognitive impairment among potential subjects for clinical research. *Medical care*, 40(9):771–781.
- [Carpenter and Grossberg, 1991] Carpenter, G. A. and Grossberg, S. (1991). *Pattern Recognition by Self-Organizing Neural Networks*. MIT Press, Cambridge, MA.
- [Chawla et al., 2002] Chawla, N., Bowyer, K., Hall, L., and Kegelmeyer, W. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.
- [Cui et al., 2012] Cui, Y., Wen, W., Lipnicki, D., Beg, M., Jin, J., Luo, S., Zhu, W., Kochan, N., Reppermund, S., Zhuang, L., Raamana, P., Liu, T., Trollor, J., Wang, L., Brodaty, H., and Sachdev, P. (2012). Automated detection of amnesic mild cognitive impairment in community-dwelling elderly adults: A combined spatial atrophy and white matter alteration approach. *NeuroImage*, 59(2):1209–1217. cited By (since 1996) 0.
- [der Malsburg, 1973] der Malsburg, C. V. (1973). Self-Organization of Orientation Sensitive Cells in the Striate Cortex. *Kybernetik*, 14:85–100.
- [der Malsburg, 1990] der Malsburg, C. V. (1990). Network self-organization. In Zornetzer, S. F., Davis, J., and Lau, C., editors, *An Introduction to Neural and Electronic Networks*, pages 421–432. Academic Press.
- [Diamantaras and Kung, 1996] Diamantaras, K. I. and Kung, S. Y. (1996). *Principal Component Neural Networks, Theory and Applications*.
- [Domingos, 1999] Domingos, P. (1999). Metacost: A general method for making classifiers cost-sensitive. In *In Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining*, pages 155–164. ACM Press.
- [Duara et al., 2012] Duara, R., Loewenstein, D., Shen, Q., Barker, W., Potter, E., Varon, D., Heurlin, K., Vandenberghe, R., and Buckley, C. (2012). Amyloid positron emission tomography with 18f-flutemetamol and structural magnetic resonance imaging in the classification of mild cognitive impairment and alzheimer’s disease. *Alzheimer’s and Dementia*. cited By (since 1996) 0; Article in Press.
- [Elkan, 2001] Elkan, C. (2001). The foundations of cost-sensitive learning. In *IJCAI*, pages 973–978.

- [Filipovych and Davatzikos, 2011] Filipovych, R. and Davatzikos, C. (2011). Semi-supervised pattern classification of medical images: Application to mild cognitive impairment (mci). *NeuroImage*, 55(3):1109–1119. cited By (since 1996) 8.
- [Flexigrid, 2010] Flexigrid (2010). <http://flexigrid.info/>.
- [Flicker et al., 1991] Flicker, C., Ferris, C. H., and Reisberg, B. (1991). Mild cognitive impairment in the elderly: Predictors of dementia. *Neurology*, 41:1006–1009.
- [Foerster and Zopf, 1962] Foerster, H. V. and Zopf, G. (1962). *Principles of Self-Organization*, Pergamon Press, Oxford, UK.
- [Freeman and Skapura, 1993] Freeman, J. A. and Skapura, D. M. (1993). *Redes neuronales. Algoritmos, aplicaciones y técnicas de programación*. Addison-Wesley.
- [Fritzsche et al., 2008] Fritzsche, K. H., von Wangenheim, A., Abdala, D. D., and Meinzer, H.-P. (2008). A computational method for the estimation of atrophic changes in alzheimer’s disease and mild cognitive impairment. *Comp. Med. Imag. and Graph.*, 32(4):294–303.
- [García Báez et al., 2011] García Báez, P., Suárez Araujo, C., and Fernández López, P. (2011). Neural network ensembles with missing data processing and data fusion capacities: Applications in medicine and in the environment. In Cabestany, J., Rojas, I., and Joya, G., editors, *Advances in Computational Intelligence*, volume 6692 of *Lecture Notes in Computer Science*, pages 169–176. Springer Berlin Heidelberg.
- [García Báez et al., 2007] García Báez, P., Suárez Araujo, C., Fernández Viadero, C., and Regidor García, J. (2007). Automatic prognostic determination and evolution of cognitive decline using artificial neural networks. 4881:898–907.
- [García Báez et al., 2008] García Báez, P., Viadero, C., García, J., and Araujo, C. (2008). An ensemble approach for the diagnosis of cognitive decline with missing data. In Corchado, E., Abraham, A., and Pedrycz, W., editors, *Hybrid Artificial Intelligence Systems*, volume 5271 of *Lecture Notes in Computer Science*, pages 353–360. Springer Berlin Heidelberg.
- [Gardner, 1970] Gardner, M. (1970). The fantastic combinations of john conway’s new solitaire game ”life”. *Scientific American*, 223:120–123.
- [Ghostscript, 1988] Ghostscript (1988). <http://www.ghostscript.com/>.
- [Grossberg, 1976] Grossberg, S. (1976). Adaptive pattern classification and universal recoding: I. parallel development and coding of neural feature detectors. *Biological Cybernetics*, 23(3):121–134.
- [Habert et al., 2011] Habert, M.-O., Horn, J.-F., Sarazin, M., Lotterie, J.-A., Puel, M., Onen, F., Zanca, M., Portet, F., Touchon, J., Verny, M., Mahieux, F., Giron, A., Fertil, B., and Dubois, B. (2011). Brain perfusion spect with an automated quantitative tool can identify prodromal alzheimer’s disease among patients with mild cognitive impairment. *Neurobiology of Aging*, 32(1):15 – 23.

- [Haykin,] Haykin, S. *Neural Networks*. Macmillan College Publishing.
- [Hebb, 1949] Hebb, D. O. (1949). *The Organization of Behavior: A Neuropsychological Theory*. Wiley, New York, new edition edition.
- [Hecht-Nielsen, 1987] Hecht-Nielsen, R. (1987). Counterpropagation networks. *Applied Optics*, 26:4979–4984.
- [Heylighen, 1992] Heylighen, F. (1992). Principles of systems and cybernetics: an evolutionary perspective. *Cybernetics and Systems '92*, pages 3–10.
- [Holland, 1998] Holland, J. (1998). *Emergence: From Chaos to Order*, Oxford University Press, Oxford, UK.
- [Hrycej, 1992] Hrycej, T. (1992). *Modular learning in neural networks - a modularized approach to neural network classification*. Sixth-Generation computer technology series. Wiley.
- [I.H. Witten, 2005] I.H. Witten, E. F. (2005). *Data Mining: Practical machine learning tools and techniques, 2nd Edition*. Morgan Kaufmann, San Francisco.
- [ImageMagick, 1999] ImageMagick (1999). <http://www.imagemagick.org>.
- [J. Hertz, 1991] J. Hertz, A. Krogh, R. P. (1991). *Introduction to the Theory of Neural Computation*. Addison-Wesley Publishing, Reading, MA.
- [Jain et al., 2000] Jain, A. K., Duin, R. P. W., and Mao, J. (2000). Statistical pattern recognition: A review. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 22(1):4–37.
- [Japkowicz and Stephen, 2002] Japkowicz, N. and Stephen, S. (2002). The class imbalance problem: A systematic study. *Intell. Data Anal.*, 6(5):429–449.
- [Jaramillo et al., 2012] Jaramillo, D., Rojas, I., Valenzuela, O., García, I., and Prieto, A. (2012). Advanced systems in medical decision-making using intelligent computing. application to magnetic resonance imaging. cited By (since 1996) 0.
- [John and Langley, 1995] John, G. and Langley, P. (1995). Estimating continuous distributions in bayesian classifiers. In *In Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 338–345. Morgan Kaufmann.
- [jQuery, 2009] jQuery (2009). <http://jquery.com/>.
- [Kazee et al., 1993] Kazee, A. M., Eskin, T. A., Lapham, L. W., Gabriel, K. R., and et al (1993). Clinicopathologic correlates in alzheimer disease: Assessment of clinical and pathologic diagnostic criteria. *Alzheimer Disease and Associated Disorders*, 7(3):152–164.
- [Kernighan and Ritchie, 1991] Kernighan, B. W. and Ritchie, D. M. (1991). *El Lenguaje de Programación C*. Pearson Educación.

- [Kohonen, 1989] Kohonen, T. (1989). *Self-Organization and Associative Memory*. Springer Series in Information Sciences, ISBN 3-540-51387-6, Berlin, GE, 3rd edition.
- [Kohonen et al., 2001] Kohonen, T., Schroeder, M. R., and Huang, T. S., editors (2001). *Self-Organizing Maps*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 3rd edition.
- [Li et al., 2012] Li, L., Wang, J., Lozar, C., and Eckert, M. (2012). Automated detection of mild cognitive impairment through mri data analysis. *International Journal on Artificial Intelligence Tools*, 21(5). cited By (since 1996) 0.
- [Lim et al., 1999] Lim, A., Tsuang, D., Kukull, W., Nochlin, D., Leverenz, J., McCormick, W., Bowen, J., Teri, L., Thompson, J., Peskind, E., Raskind, M., and Larson, E. (1999). Clinico-neuropathological correlation of alzheimer’s disease in a community-based case series. *J Am Geriatr Soc*, 47(5):564–9.
- [Lobo et al., 1999] Lobo, A., Saz, P., Marcos, G., D a, J. L., de la C amara, C., and et al. (1999). Revalidaci n y normalizaci n del mini-examen cognoscitivo (primera versi n en castellano del mini-mental status revalidaci n y normalizaci n del mini-examen cognoscitivo (primera versi n en castellano del mini-mental status examination) en la poblaci n general geri trica. *Medicina Cl nica*, 112(20):767–774.
- [Lonie et al., 2009] Lonie, J., Tierney, K. M., and Ebmeier, K. (2009). Screening for mild cognitive impairment: a systematic review. *Int. J. Geriatr. Psychiatry*, 24:902–915.
- [Luis et al., 2003] Luis, C. A., Loewenstein, D. A., Acevedo, A., Barker, W. W., and Duara, R. (2003). Mild cognitive impairment: directions for future research. *Neurology*, 61:438–444.
- [Lv et al., 2010] Lv, S., Wang, X., Cui, Y., Jin, J., Sun, Y., Tang, Y., Bai, Y., Wang, Y., and Zhou, L. (2010). Application of attention network test and demographic information to detect mild cognitive impairment via combining feature selection with support vector machine. *Computer Methods and Programs in Biomedicine*, 97(1):11–18. cited By (since 1996) 2.
- [Mahoney and Barthel, 1965] Mahoney, F. I. and Barthel, D. (1965). Functional evaluation: The barthel index. *Maryland State Medical Journal*, 14:61–65.
- [Mao and Jain, 1995] Mao, J. and Jain, A. K. (1995). Artificial neural networks for feature extraction and multivariate data pro. *IEEE Transactions on Computers*, C-18:401–409.
- [Marsaglia and Tsang, 2000] Marsaglia, G. and Tsang, W. W. (2000). The Ziggurat Method for Generating Random Variables. *Journal of Statistical Software*, 5(8):1–7.
- [McKhann et al., 1984] McKhann, G., Drachman, D., Folstein, M., Katzman, R., Price, D., and Stadlan, E. M. (1984). Clinical diagnosis of alzheimer’s disease: report of the nincls-adrda work group under the auspices of department of health and human services task force on alzheimer’s disease. *Neurology*, 34(7):939–944.

- [Minsky and Papert, 1969] Minsky, M. and Papert, S. (1969). *Perceptrons*. Cambridge, MA: MIT Press.
- [O’Dwyer et al., 2012] O’Dwyer, L., Lamberton, F., Bokde, A., Ewers, M., Faluyi, Y., Tanner, C., Mazoyer, B., O’Neill, D., Bartley, M., Collins, D., Coughlan, T., Prvulovic, D., and Hampel, H. (2012). Using support vector machines with multiple indices of diffusion for automated classification of mild cognitive impairment. *PLoS ONE*, 7(2). cited By (since 1996) 1.
- [Oja, 1997] Oja, E. (1997). The nonlinear pca learning rule in independent component analysis. *Neurocomputing*, 17(1):25 – 45.
- [Petersen, 2004] Petersen, R. C. (2004). Mild cognitive impairment as a diagnostic entity. *J. Intern. Med.*, 256(3):183–194.
- [Petersen et al., 1997] Petersen, R. C., Smith, G. E., Waring, S. C., Ivnik, R. J., Kokmen, E., and Tangelos, E. G. (1997). Aging, memory, and mild cognitive impairment. *International Psychogeriatrics*, 9:65–69.
- [Quinlan, 1993] Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- [Quintana et al., 2012] Quintana, M., Guardia, J., Sanchez-Benavides, G., Aguilar, M., Molinuevo, J. L., Robles, A., Barquero, M. S., Antunez, C., Martinez-Parra, C., Frank-Garcia, A., Fernandez, M., Blesa, R., Pena-Casanova, J., and for the Neuronorma Study Team (2012). Using artificial neural networks in clinical neuropsychology: High performance in mild cognitive impairment and alzheimer’s disease. *Journal of Clinical and Experimental Neuropsychology (Neuropsychology, Developm.*, 34(2):195–208.
- [Ravaglia et al., 2005] Ravaglia, G., Forti, P., Maioli, F., Servadei, L., Martelli, M., Brunetti, N., Bastagli, L., and Mariani, E. (2005). Screening for mild cognitive impairment in elderly ambulatory patients with cognitive complaints. *Aging Clin Exp Res*, 17(5):374–379.
- [Repository, 2007] Repository, U. M. L. (2007). <http://archive.ics.uci.edu/ml>.
- [Ripley, 1996] Ripley, B. D. (1996). *Pattern Recognition and Neural Networks*. Cambridge University Press, Cambridge.
- [Rojas, 1996] Rojas, R. (1996). *Neural Networks*. Springer-Verlag.
- [Rosenblatt, 1961] Rosenblatt, F. (1961). *Principles of neurodynamics*. Spartan Books, Washington, WA.
- [Ross, 1953] Ross, A. (1953). Notices of recent publications. *Brain*, 76(1):146–148.
- [Rossini et al., 2008] Rossini, P. M., Buscema, M., Capriotti, M., Grossi, E., Rodriguez, G., Percio, C. D., and Babiloni, C. (2008). Is it possible to automatically distinguish resting eeg data of normal elderly vs. mild cognitive impairment subjects with high degree of accuracy? *Clinical Neurophysiology*, 119(7):1534 – 1545.

- [Sager et al., 2006] Sager, M. A., Hermann, B. P., La Rue, A., and Woodard, J. L. (2006). Screening for dementia in community-based memory clinics. *WMJ*, 105(7):25–29.
- [Samad and Harp, 1992] Samad, T. and Harp, S. A. (1992). Self-organization with partial data. *Network: Computation in Neural Systems*, 3(2):205–212.
- [Sanger, 1989] Sanger, T. (1989). An optimality principle for unsupervised learning. pages 11–19, San Mateo, CA. Morgan Kaufmann.
- [Sarle, 1994] Sarle, W. S. (1994). Neural networks and statistical models. *Proceedings of the XIX SAS User Group*, pages 1538–1550.
- [Solomon and Murphy, 2005] Solomon, P. R. and Murphy, C. A. (2005). Should we screen for alzheimer’s disease? *Geriatrics*, 60(11):26–31.
- [The GTK+ Project, 1998] The GTK+ Project (1998). <http://www.gtk.org/>.
- [Turing, 1952] Turing, A. M. (1952). The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society*, 237:5–72.
- [Varma et al., 1999] Varma, A. R., Snowden, J. S., Lloyd, J. J., Talbot, P. R., Mann, D. M. A., and Neary, D. (1999). Evaluation of the nincds-adrda criteria in the differentiation of alzheimer’s disease and frontotemporal dementia. *Journal of Neurology, Neurosurgery and Psychiatry*, 66(2):184–188.
- [Wee et al., 2012] Wee, C.-Y., Yap, P.-T., and Shen, D. (2012). Prediction of alzheimer’s disease and mild cognitive impairment using cortical morphological patterns. *Human Brain Mapping*. cited By (since 1996) 0; Article in Press.
- [Weiss, 2004] Weiss, G. M. (2004). Mining with rarity: a unifying framework. *SIGKDD Explor. Newsl.*, 6(1):7–19.
- [Weiss and Tian, 2008] Weiss, G. M. and Tian, Y. (2008). Maximizing classifier utility when there are data acquisition and modeling costs. *Data Min. Knowl. Discov.*, 17(2):253–282.
- [Werbos, 1974] Werbos, P. J. (1974). *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. PhD thesis, Harvard University.
- [Zhang et al., 2011] Zhang, D., Wang, Y., Zhou, L., Yuan, H., and Shen, D. (2011). Multimodal classification of alzheimer’s disease and mild cognitive impairment. *NeuroImage*, 55(3):856–867.