

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

**A study of the general characteristics of
Serial fusion of classifiers**

Antonio Domínguez Navarro

Tutor: Simone Santini

Julio 2015

**A study of the general characteristics of
Serial fusion of classifiers**

AUTOR: Antonio Domínguez Navarro

TUTOR: Simone Santini

**Escuela Politécnica Superior
Universidad Autónoma de Madrid
Julio de 2015**

Resumen

Con el continuo avance tecnológico de las últimas décadas, el volumen y la precisión de los datos a manejar se ha visto incrementado de forma directa, haciéndose necesario a su vez el estudio y el diseño de métodos que permitan analizarlos y clasificarlos de forma automática. Con este cometido se han desarrollado multitud de clasificadores siguiendo, desde modelos matemáticos a modelos bioinspirados, en muchos casos con un desempeño mucho mayor que aceptable en escenarios concretos.

Con el fin de mejorar el rendimiento de estos clasificadores, algunas investigaciones, sobre todo en las últimas dos décadas, se han centrado en el estudio de modelos formados por la unión de varios clasificadores individuales, creando una estructura común que proporciona una solución única al problema, hallada a partir de los resultados obtenidos por los distintos clasificadores que la forman. Las estructuras más comúnmente estudiadas conectan cada uno de los módulos individuales en paralelo entre sí, calculando la salida común una vez que todos los clasificadores individuales han tomado un valor en su salida. Este tipo de combinación de clasificadores se ha probado en diversos estudios como una alternativa con una mejor precisión frente a los clasificadores individuales, siendo además más robustos frente a errores puntuales e intentos de falsificación o suplantación, como pueden darse en sistemas de identificación mediante reconocimiento de patrones biométricos.

Aun así, al combinar varios clasificadores en paralelo se generan algunas desventajas, como la necesidad de desarrollar cada clasificador por separado para cumplir con unas necesidades específicas dentro del conjunto, el aumento de recursos necesarios para la ejecución de cada uno de los módulos que componen la estructura al mismo tiempo, o el aumento del tiempo medio necesario para ofrecer un resultado, limitado en este caso por el clasificador más lento.

Por otro lado, las estructuras de clasificadores en serie o cascada buscan un equilibrio entre el tiempo de respuesta y la precisión, manteniéndose a la vez estables ante suplantaciones y optimizando el uso de recursos en la medida de lo posible. En la última década han proliferado las publicaciones sobre este modelo, investigando la aplicación de varios clasificadores en serie a diversos escenarios, como la identificación mediante patrones biométricos, diagnóstico del nivel de stress en sujetos individuales o reconocimiento de vehículos a motor mediante imágenes, entre otros, ya sea utilizando el resultado de un solo clasificador para decidir en cualquier estado o tomando la fusión de los clasificadores en serie utilizados para calcular la puntuación definitiva.

Palabras clave

Clasificación multi-estado, reconocimiento de patrones, combinación en serie, redes neuronales

Abstract

Along the continuous technological advance of the last decades, the amount of data to be handled has been incremented in a proportional way. Therefore, a new way to analyze and classify huge amounts of data automatically is needed to be studied and designed. Following this goal, a lot of classifiers have been investigated using mathematical models or even bio-inspired models, performing better than expected in specific scenarios.

Aiming to improve those classifiers performance, some researches made in the last twenty years pointed to more complex structures that solve the problem by connecting individual classifiers and fusing each individual score into a global one. The ensembles most commonly studied connect each single classifier in parallel, evaluation system's output once each individual classifier has given its own score. This kind of classifier ensembles has been tested as an alternative to individual classifiers with better precision, more reliable to certain errors and spoofing attacks, most common threat of identification systems based on biometric pattern recognition.

Nevertheless, some problems are shown on combining different classifiers in parallel. Each classifier needs to be designed and trained individually, the amount of resources needed for each module execution at the same time or the increment of the time needed to calculate the final score, limited in this case by the slowest classifier of the ensemble.

On the other hand, serial and cascade ensembles are designed to find a tradeoff between latency and precision, optimize resource usage while being at least as reliable against spoofing attacks as parallel system are. In the last decade, a lot of work related to this model has been published, doing some research on the implementation of serial fusion ensembles in real-life scenarios, like biometric pattern recognition, stress level diagnosis of individual subjects or vehicle recognition based on images. These examples were designed using different approaches, like accepting the score of a single classifier of the ensemble on a certain stage as the final score or performing a fusion of the scores calculated at a certain stage.

Keywords

Multi-stage classification, pattern recognition, serial combination, neural networks

Índice

1. INTRODUCCIÓN.....	1
1.1 MOTIVACIÓN DEL PROYECTO.....	1
1.2 OBJETIVO DEL TRABAJO	2
1.3 ESTRUCTURA DEL DOCUMENTO	2
2. ESTADO DEL ARTE	3
2.1 CLASIFICACIÓN DE DATOS.....	3
2.1.1 Redes Neuronales Artificiales	3
2.2 CONJUNTOS DE CLASIFICADORES	4
2.3 FUSIÓN DE CLASIFICADORES EN SERIE	6
3. DISEÑO	8
3.1 DISEÑO DE LOS CLASIFICADORES INDIVIDUALES.....	8
3.1.1 Capa de entrada.....	8
3.1.2 Capa de salida.....	8
3.1.3 Capas ocultas.....	9
3.1.4 Función de propagación o de red.....	9
3.1.5 Función de activación	9
3.2 ORDEN DE EVALUACIÓN DE LOS CLASIFICADORES.....	10
3.3 DISEÑO DEL CONJUNTO DE CLASIFICADORES EN FUNCIÓN DE LOS DATOS A PROCESAR.....	11
3.3.1 Datos no clasificados en el estado anterior.....	11
3.3.2 Datos incrementales	12
3.3.3 Datos independientes	13
3.4 DISEÑO DEL CONJUNTO DE CLASIFICADORES SEGÚN LA DECISIÓN EN CADA ESTADO.....	15
3.4.1 Decisión simple	15
3.4.2 Decisión múltiple	16
3.4.3 Fusión de las puntuaciones	17
3.5 NÚMERO DE CLASIFICADORES EN LA ESTRUCTURA.....	17
3.5.1 Clasificadores de doble estado.....	17
3.5.2 Clasificadores multi-estado	19
3.6 CONFIGURACIÓN DE LOS UMBRALES DE DECISIÓN DE CADA CLASIFICADOR.....	19
3.6.1 Umbrales en función de las curvas ROC del clasificador	19
3.6.2 Compromiso entre precisión y desempeño.....	20
3.7 FUSIÓN DE LAS PUNTUACIONES DE LOS MARCADORES INDIVIDUALES	21

3.7.1	Funciones lógicas	21
3.7.2	Clasificación de patrones.....	21
3.7.3	Métodos estadísticos.....	22
3.8	ESTRUCTURAS MÁS COMUNES.....	22
3.8.1	Estructura de decisión múltiple.....	22
3.8.2	Estructura en cascada.....	23
4.	DESARROLLO.....	25
4.1	TECNOLOGÍAS UTILIZADAS	25
4.2	ENTORNO DE DESARROLLO	25
4.3	FORMATO DE LAS ESTRUCTURAS DE DATOS AUXILIARES.....	25
5.	CASOS DE PRUEBA	26
5.1	TEST SOBRE CLASIFICADORES LINEALES Y MULTICAPA	26
5.2	CONJUNTO BI-ESTADO DE CLASIFICADOR LINEAL Y MULTICAPA.....	28
5.3	CONJUNTO BI-ESTADO FRENTE A PROBLEMAS REALES.....	30
5.3.1	Predicción de donantes	30
5.4	DETECCIÓN DE FALSAS ALARMAS EN SISTEMAS COMPLEJOS.....	31
5.4.1	Diseño de los clasificadores.....	33
5.4.2	Rendimiento de los clasificadores	34
5.4.3	Estructura final del conjunto	36
5.4.4	Resultados	36
6.	CONCLUSIONES Y TRABAJO FUTURO	37
7.	BIBLIOGRAFÍA	38

Relación de Figuras

- **Figura 1:** modelo de neurona
- **Figura 2:** Esquema de un perceptron de una sola neurona
- **Figura 3:** Clasificador multi-estado con fusión por votación. Hansel/Salamon 1990
- **Figura 4:** Esquema de una red neuronal de una sola neurona con función de activación y función de red
- **Figura 5:** diseño con decisión obligatoria en el último estado
- **Figura 6:** Esquema de entrenamiento con datos incrementales en cada estado
- **Figura 7:** Esquema de clasificación de un conjunto de clasificadores en serie con datos incrementales
- **Figura 8:** Esquema de entrenamiento de un conjunto de clasificadores en serie con datos independientes
- **Figura 9:** Esquema de clasificación de un conjunto de clasificadores en serie con datos independientes
- **Figura 10:** Esquema de un clasificador individual de decisión simple por rechazo. Este clasificador no puede aceptar una muestra, tan solo rechazar o saltar de estado.
- **Figura 11:** Esquema de un clasificador individual de decisión múltiple. Este clasificador puede aceptar una muestra, rechazar o saltar de estado.
- **Figura 12:** figura ilustrativa de la actuación de un clasificador discriminativo (izquierda) y un clasificador por clase o modelo (derecha). Las líneas negras continuas representan las áreas de decisión de cada clase.
- **Figura 13:** figura ilustrativa de la actuación de un clasificador discriminativo seguido de un clasificador por clase sobre datos atípicos y ambiguos.
- **Figura 14:** Curvas ROC en función del umbral de decisión del clasificador. Los puntos S_i^* y S_u^* se proponen como umbrales de decisión para el clasificador. *Pattern Recognition*, G.L.Marcialis, F. Roli, L.Didaci [8]
- **Figura 15:** Sistema de decisión múltiple en todos sus clasificadores. Todos, incluyendo el estado final, pueden rechazar, aceptar o no decidir frente a unos datos de entrada dados
- **Figura 16:** Sistema en cascada con filtrado por rechazo. En esta estructura tan solo el último clasificador puede aceptar los datos, siendo efectivo cuando los datos a clasificar están muy desbalanceado
- **Figura 17:** Representación de ANN creada mediante la librería nnet.
<https://beckmw.wordpress.com>
- **Figura 18:** problema a clasificar, clase 1 representada en azul, clase 0 representada en verde.

- **Figura 19:** Resultado del conjunto de clasificadores lineales. Predicción de clase 1 en azul, predicción de clase 1 cercana al umbral en rojo, predicción de clase 0 en verde, predicción cercana al umbral en amarillo y errores en negro.
- **Figura 20:** Resultado del conjunto de clasificadores. Predicción de clase 1 en azul, predicción de clase 1 cercana al umbral en rojo, predicción de clase 0 en verde, predicción cercana al umbral en amarillo y errores en negro.
- **Figura 21:** Curva ROC del primer clasificador. Puede verse como el rendimiento del mismo es mejor que la función aleatoria representada en gris, aunque con un error importante.
- **Figura 22:** Señal resultante (izquierda) y señal de cada uno de los sensores (derecha) en el paso de un eje sin daños.
- **Figura 23:** Señal resultante (izquierda) y señal de cada uno de los sensores (derecha) en el paso de un eje con un plano fuerte. Puede verse como el sistema registra dos revoluciones de la rueda.
- **Figura 24:** Señal resultante (izquierda) y señal de cada uno de los sensores (derecha) en caso de una mala calibración del sistema. Los sensores quedan fuera de escala desvirtuando la señal.
- **Figura 25:** Curva Roc del primer clasificador, Positivos verdaderos frente a falsos positivos.
- **Figura 26:** Curva Roc del segundo clasificador. Positivos verdaderos frente a falsos positivos.
- **Figura 27:** Curva Roc del tercer clasificador. Positivos verdaderos frente a falsos positivos.

Relación de Tablas

- **Tabla 1:** Funciones sigmoideas, utilizadas muy comúnmente como funciones de activación - 2015 Facultad de Eléctrica, Instituto Superior Politécnico José Antonio Echeverría, Cujae
- **Tabla 2:** características de los clasificadores prueba 1
- **Tabla 3:** parámetros del conjunto de clasificadores prueba 1
- **Tabla 4:** características de los clasificadores prueba 2
- **Tabla 5:** parámetros del conjunto de clasificadores prueba 2
- **Tabla 6:** resultados de la predicción en prueba 2 en cada uno de los estados y predicción final
- **Tabla 7:** características de los clasificadores prueba 3
- **Tabla 8:** parámetros del conjunto de clasificadores prueba 3
- **Tabla 9:** resultados de la predicción en prueba 3 en cada uno de los estados y predicción final
- **Tabla 10:** características de los clasificadores prueba 4
- **Tabla 11:** parámetros del conjunto de clasificadores prueba 4
- **Tabla 12:** Resultados individuales por clasificador de la prueba 4
- **Tabla 13:** Resultados finales de la prueba 4

Glosario

- **Re conocimiento de patrones:** rama del aprendizaje automático que se ocupa de identificar patrones y similitudes en los datos, normalmente mediante procesos de aprendizaje supervisado si existe data sobre la que hacer dicho entrenamiento, o no supervisado si los datos disponibles están sin etiquetar.
- **Aprendizaje supervisado:** se trata de un método de aprendizaje automático en el cual el clasificador conoce a priori el objetivo, adecuando durante cada iteración del entrenamiento su estructura para optimizar el error de clasificación.
- **Aprendizaje no supervisado:** algoritmos que intentan agrupar los datos no etiquetados a priori con el fin de construir dichas etiquetas a partir de las similitudes entre cada vector de datos, intentando que las similitudes entre cada grupo sean escasas.
- **Aprendizaje por refuerzo:** algoritmos de aprendizaje basados en el ensayo y error. En este caso el algoritmo toma la opción que considera idónea entre n opciones disponibles para resolver el problema. Tras intentarlo se introduce un feedback sobre su desempeño que puede variar su comportamiento.
- **Regla de Bayes:** teorema estadístico formulado por Bayes que vincula la probabilidad de que ocurra un hecho A dado un hecho B a la probabilidad de que ocurra B dado A.
- **SVM (Support Vector Machine):** clasificadores basados en algoritmos de aprendizaje supervisado diseñados por Vladimir Vapnik. Tras un proceso de entrenamiento, este modelo crea hiperplanos que separan de manera discriminativa las diferentes clases, pudiendo predecir la clase de nuevas entradas tras su entrenamiento.
- **KNN(K-nearest neighbors):** método de clasificación mediante entrenamiento supervisado que estima la probabilidad de que un elemento x pertenezca a una clase C a partir de la información proporcionada por el conjunto de datos de entrenamiento.
- **ANN (Artificial Neural Networks):** redes neuronales artificiales
- **Regla de Hebb:** modelo desarrollado por Donald Hebb que modela la variación de pesos de las neuronas para el aprendizaje. La abstracción matemática de este modelo supuso un cambio importante en el desarrollo de las redes neuronales artificiales.
- **Su plantacion/Spoofing:** ataque sobre un sistema de identificación en el cual el atacante intenta ser identificado positivamente introduciendo una falsificación en la entrada con el fin de que sea identificada como genuina.
- **Lobos:** son conocidos como lobos aquellos usuarios malintencionados que intentan ingresar en un sistema mediante entradas falsificadas con características muy semejantes a gran parte de las registradas como genuinas en el sistema, o mediante entradas muy semejantes a una muestra genuina en concreto registrada en el sistema.

- **Corderos:** este tipo de usuarios son aquellos que sin ánimo de engañar al sistema, poseen una entrada con unas características muy semejantes a una o más muestras registradas en el sistema, pudiendo identificar su muestra como otra diferente dentro del sistema.
- **Ensemble(of classifiers):** conjunto de clasificadores relacionados entre sí que dada una entrada, proporcionan una única salida a partir de la información ofrecida por cada uno de ellos.
- **SPRT (Sequential probability ratio test):** Test de hipótesis secuencial desarrollado por Abraham Wald, en el cual se acumulan los datos del ratio de verosimilitud logarítmica de la entrada con la hipótesis de la entrada actual y las anteriores. Si esta suma acumulada supera un umbral, se acepta la hipótesis alternativa, y si está por debajo de un segundo umbral, se acepta la hipótesis nula. En caso de no cumplir ninguna de estas opciones, se continúa acumulando información.
- **Curvas ROC: (Receiver Operating Characteristic)** representación gráfica de la sensibilidad de un clasificador binario según varía el umbral de discriminación. Normalmente representa la razón o tasa de verdaderos positivos frente a los falsos positivos del clasificador.

1. Introducción

1.1 MOTIVACIÓN DEL PROYECTO

Tras la publicación del artículo Neural Network Ensembles en 1990[1], la investigación y las publicaciones sobre la agrupación de clasificadores se vieron incrementadas al ver una vía para mejorar el rendimiento obtenido por estos individualmente, así como estudiar las características que modelan el comportamiento, entrenamiento y desempeño de clasificadores complejos resultantes. Este ritmo no ha decaído con el paso de los años, y en muchos casos los estudios y pruebas de estos artículos han dado paso a sistemas que operan en la actualidad en contextos reales, demostrando más aún si cabe las mejoras a las que se apuntaba desde los primeros análisis teóricos.

Aun así, y sobre todo en los primeros años, la mayoría de dichas investigaciones se dirigieron hacia la unión en paralelo de estos sistemas individuales para formar un sistema complejo con una sola decisión final, obteniendo buenos resultados con algunos puntos en contra, como el descenso del rendimiento, la necesidad de más recursos o la necesidad de mayor volumen de datos. Los conjuntos de clasificadores en paralelo han demostrado una buena precisión para múltiples problemas de reconocimiento y clasificación de patrones de gran relevancia, como filtros de spam, OCR o seguridad biométrica, entre otros, superando el rendimiento de los clasificadores individuales en la mayoría de los escenarios, a cambio de un aumento en la dificultad a la hora de construir el clasificador. Esta dificultad reside en su mayor parte en el método de fusión de los distintos resultados de cada módulo individual que compone el sistema, siendo posible usar métodos lógicos, estadísticos, o incluso un clasificador adicional que decida en función de todos los resultados.

Por otra parte, y al inicio de forma poco continuada y sin mucha profundidad, se comenzó a estudiar de forma paralela otro tipo de estructuras construidas mediante el mismo tipo de sistemas individuales, construyendo en este caso un sistema complejo uniéndolas en serie. Teóricamente, el rendimiento y versatilidad de las estructuras en serie eran muy prometedores, pero su estudio y diseño se antoja mucho más complejo: evaluar el orden óptimo en el que sitiar los distintos módulos, definir una condición de salto al siguiente módulo, estrategia de fusión de los resultados de cada módulo, cálculo del rendimiento no trivial, estrategia de entrenamiento de los distintos niveles... Pese a estos inconvenientes, los buenos resultados han incrementado el interés sobre este tipo de ensembles, multiplicándose las publicaciones en la última década y mostrando resultados por encima de los modelos en paralelo en sobre escenarios concretos, aprovechando la complejidad del sistema para adaptarles a cada problema de la mejor manera posible. Bajo este contexto, los conjuntos de clasificadores en serie se presentan como un tema interesante sobre el que indagar, analizando las distintas posibilidades de diseño y comprobando algunos casos concretos.

1.2 OBJETIVO DEL TRABAJO

Tras conocer algunos resultados obtenidos en artículos publicados, así como aplicaciones reales de los conjuntos de clasificadores en serie, resulta de interés conocer algo más sobre esta tecnología, así como comprobar si es aplicable en problemas concretos, reales o lo más cercanos posible a la realidad.

Pese a que el estudio de la combinación de clasificadores en serie ha aumentado, la complejidad de los mismos y la cantidad de variantes a construir hacen que aun quede mucho trabajo por delante. Este trabajo pretende dar una base para el estudio y la comprensión de este tipo de conjuntos, haciendo un repaso sobre los parámetros que regulan su diseño, las opciones de fusión, las distintas variantes existentes y los distintos problemas a los que están enfocadas cada una de ellas.

Por otra parte, en este documento se muestran algunos ejemplos de aplicación que buscan ilustrar algunas de las mejoras que ofrece la combinación serial frente a otras soluciones en problemas concretos el diagnóstico de falsas alarmas o la predicción de ciertos fenómenos.

Por último, se muestran algunas conclusiones, seguidas de distintas opciones de trabajo futuro, como posibles aplicaciones para alguna de las variantes aquí recogidas.

1.3 ESTRUCTURA DEL DOCUMENTO

Tras esta introducción, este documento recopila en el apartado Estado del Arte un resumen de las investigaciones llevadas a cabo sobre los sistemas compuestos por clasificadores en serie, independientemente del tipo de fusión aplicado.

Tras definir con algunas pinceladas el desarrollo actual de la combinación de clasificadores en serie, el siguiente apartado, Tecnologías Utilizadas, recopila tanto el hardware como el software utilizado en el desarrollo de las pruebas descritas en este trabajo.

El apartado de diseño y desarrollo describe de una forma más detallada algunos de los diseños procedentes de estudios, describiendo sus ventajas e inconvenientes en cada caso y realizando un análisis teórico en cada caso.

Tras la teoría, el siguiente apartado pasa al apartado práctico, mostrando el desarrollo del software de prueba creado en cada caso, siguiendo los diseños visto en el apartado anterior.

Las diferentes pruebas a realizar mediante los programas de prueba creados se definen en el apartado siguiente, Casos de Pruebas. Seguidamente a los casos de prueba se mostraran los resultados de las mismas, comprobando si se ajustan a lo esperado.

Por último, se recogen las conclusiones extraídas de este estudio, así como algunas ideas de trabajo futuro sobre la fusión en serie de clasificadores, seguido del listado de recursos bibliográficos que fueron utilizados en el desarrollo de este documento y el trabajo asociado a él.

2. Estado del arte

2.1 CLASIFICACIÓN DE DATOS

Los problemas de clasificación no son una novedad en nuestros días. Hace varias décadas que se buscan nuevos sistemas que clasifiquen grandes cantidades de datos de forma automática, valiéndose para ello de teoremas matemáticos, estudios geométricos e incluso sistemas bio-inspirados. La mayoría de ellos necesitan un periodo de aprendizaje para poder decidir sobre un problema concreto ya sea supervisado, no supervisado, o basado en refuerzo.

Existe una gran variedad de clasificadores, desde los basados en el teorema de Bayes a las SVM, pasando por los clasificadores Parzen, con PCA o las redes neuronales artificiales (ANN), cada uno con sus ventajas y sus inconvenientes según el problema a resolver.

2.1.1 Redes Neuronales Artificiales

En este trabajo utilizaremos mayoritariamente redes neuronales artificiales o ANN como módulos unitarios para construir los conjuntos complejos. Estos clasificadores están basados en las estructuras celulares halladas en diferentes grados de complejidad en el sistema nervioso de ciertos seres vivos; desde el sistema nervioso central más simple de conjunto de los Plelmintos a los Cordados con sus redes neuronales especializadas y una alta cefalización.

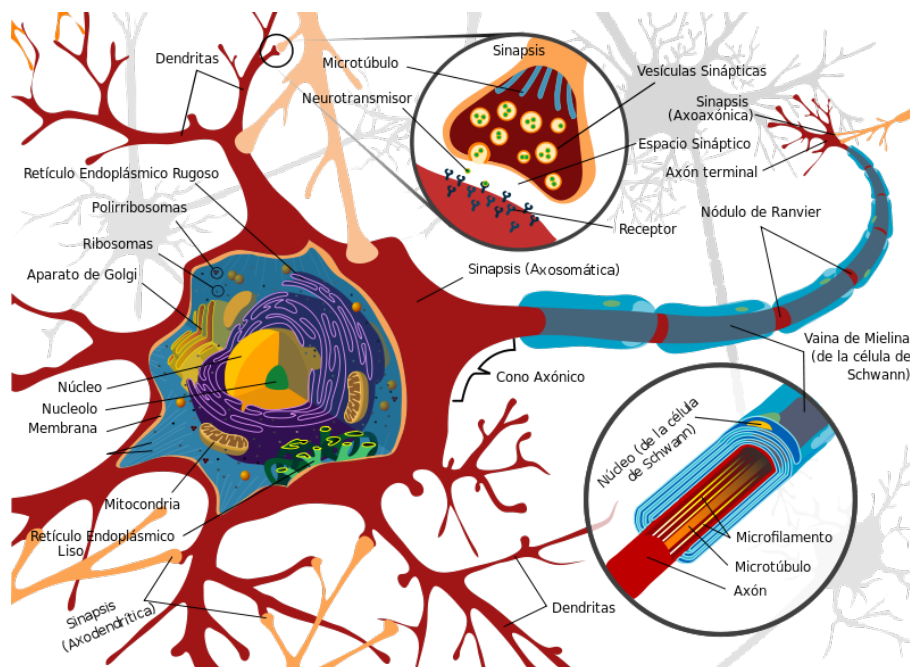


Figura 3: modelo de neurona

Las células que forman estas redes son conocidas como neuronas, capaces de recibir estímulos y propagar impulsos nerviosos a través de sus conexiones con otras neuronas. Las redes neuronales se caracterizan por un alto grado de interconectividad entre sus

neuronas, creando una estructura en paralelo con una alta redundancia. Esto hace que las redes neuronales muestren una importante tolerancia a errores, minimizando el impacto de la pérdida de parte de la red, teniendo además la capacidad de reorganizarse con el fin suplir las conexiones perdidas.

Teniendo en cuenta la estructura real de una red neuronal, en 1943 los neurólogos Warren McCulloch y Walter Pitts[2] desarrollan los primeros modelos de redes neuronales artificiales, buscando imitar el funcionamiento de las redes neuronales biológicas, así como sus principales ventajas. A partir de estas ideas, otros científicos tomaron interés en el estudio de las redes de neuronas artificiales, tal como Hebb, responsable de diversos estudios sobre el aprendizaje y la conexión neuronal; Rosenblatt y su modelo Perceptrón a finales de la década de los 50.

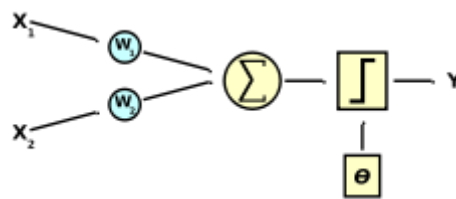


Figura 4: Esquema de un perceptrón de una sola neurona

Pese a que ADALINE, la primera red neuronal artificial en tener una aplicación real, apareció en 1960, las redes neuronales artificiales sufrieron un duro revés a raíz de la publicación del libro Perceptrón de Minsky y Papert[3], donde demostraban las limitaciones de las redes neuronales utilizadas hasta la fecha a la hora de implementar funciones como XOR, frenándose drásticamente su investigación hasta la aparición en la década de los 80 de las redes de Hopfield y del algoritmo de aprendizaje por retropropagación por Rumelhart y McLellan, que aplicado sobre el Perceptrón multicapa solventaría las limitaciones expuestas por Minsky y Papert años antes.

A partir de los 80, las ANN se han aplicado con éxito en la resolución de problemas de lo más variado, mostrando una buena capacidad de generalización, unida a las características ya vistas en su versión biológica: tolerancia a errores, adaptabilidad... Además presentan un muy buen rendimiento a la hora de tomar decisiones a cambio de un proceso de entrenamiento bastante costoso.

2.2 CONJUNTOS DE CLASIFICADORES

Las variantes y mejoras enunciadas en la publicación de Hansen y Salamon[1] en 1990 sobre la unión de clasificadores simples como un solo clasificador complejo resultaron bastante prometedoras. En el artículo mostraban como tres clasificadores que no conseguían clasificar de forma correcta una clase eran capaces de clasificarla de forma ampliamente más precisa tan solo efectuando un proceso de votación entre las decisiones de los 3 clasificadores para fusionarlas en una sola. En su primer planteamiento entrenaron tres clasificadores por separado con el fin de clasificar los elementos de un espacio bidimensional, en una clase los elementos del primer y tercer cuadrante y en otra los pertenecientes al cuarto y segundo. Como puede verse en la imagen, la precisión de los clasificadores por separado era limitada, mientras que, con un sencillo proceso de fusión

de los resultados de los 3 clasificadores mediante votación, la precisión mejora drásticamente hasta clasificar el problema con un error mínimo.

30,000 showings



Figura 3: Clasificador multi-estado con fusión por votación. Hansel/Salamon 1990

Los resultados de Hansen y Salamon abrieron la puerta al uso de clasificadores menos complejos, más baratos y más rápidos. Pese a que individualmente no son capaces de realizar una clasificación precisa, una simple fusión por consenso de varios clasificadores entrenados por separado parecía ofrecer un resultado muy superior. El resultado fue un incremento de la actividad investigadora en los años siguientes sobre los conjuntos de clasificadores en múltiples vertientes, desde modelos genéricos a aplicaciones sobre problemas reales muy concretos, buscando obtener las ventajas esperadas. Un claro ejemplo es el artículo "Design, implementation and análisis of a parallel description classifier" [9], que ilustra además la complejidad añadida a la creación de este tipo de clasificadores al aumentar los parámetros y las variantes por la conexión de los distintos clasificadores que lo componen.

La mayoría de estas investigaciones siguieron la estela ya trazada: un número determinado de clasificadores con un algoritmo o regla de fusión sobre las clasificaciones individuales, formando una estructura en paralelo donde el rendimiento depende directamente del clasificador más lento de la estructura, siempre en caso de contar con los recursos suficientes para que cada clasificador. En este escenario, se debía elegir entre aumentar los recursos requeridos para la clasificación o evaluar cada clasificador cuando existieran recursos suficientes en el sistema, incrementando significativamente el tiempo necesario para la clasificación.

Adicionalmente, uno de los campos que más atención ha mostrado sobre la fusión de múltiples clasificadores es la biométrica, caracterizada por problemas de clasificación con un alto grado de dificultad, en muchos casos basados en la comparación de texturas con un alto coste de clasificación. En estos casos, la fusión de clasificadores en paralelo nos obligaría a desarrollar distintos clasificadores para una misma característica biométrica, o implementar varios clasificadores especializados en diferentes características biométricas, encontrando puntos negativos en ambos casos:

En caso de clasificaciones costosas como el reconocimiento de iris mediante comparación de texturas, crear varios clasificadores para la misma característica biométrica elevará la relación tiempo-recursos necesaria para llevar a cabo la clasificación.

Si se opta por evaluar distintas características del sujeto, el registro en el sistema será más intrusivo, al ser necesario el análisis y almacenado de cada uno de los patrones biométricos del usuario que serán utilizados en la identificación. Más aún, cada vez que el sujeto deba ser identificado, deberá pasar por el mismo trámite, resultando en un proceso muy intrusivo.

Por otra parte, al evaluar estos sistemas ante ataques de suplantación [4] perpetrados por "lobos", o incluso ante sujetos reconocidos como "corderos" [5] por sus similitudes con otros sujetos registrados en el sistema, diferentes estudios muestran la debilidad de los sistemas paralelos en función del proceso de fusión seguido. En caso de votación, un sujeto con ciertas similitudes con uno o más usuarios del sistema en más de uno de sus rasgos puede ser aceptado como otro usuario por el sistema en caso de usar un método por consenso. Por otra parte, un sujeto malicioso podría usar un método de suplantación que le garantizara una puntuación muy alta sobre una de las características y ser aceptado en caso de utilizar el sistema un método de fusión por máximo o con pesos, siendo el clasificador suplantado el de más peso en la decisión.

Pese a que existen ventajas claras, la fusión en paralelo de clasificadores resulta de difícil aplicación en ciertos problemas, en muchos casos por el primer problema que acabamos de ver: aplicaciones con una necesidad de respuesta en tiempo real, o cercana a ella, necesitarían una gran cantidad de recursos para evaluar de forma totalmente paralela cada uno de los clasificadores, yaun así, el mejor rendimiento posible sería el del clasificador más lento más el coste del algoritmo o regla de fusión elegida en su evaluación sobre el vector de resultados. Este hecho ha sido analizado en diversas publicaciones, tanto de forma general ([6][7]), como de forma específica en problemas de clasificación biométrica [8].

2.3 FUSIÓN DE CLASIFICADORES EN SERIE

En torno al año 1994 comenzaron a aparecer publicaciones sobre otra opción a la hora de agrupar clasificadores: estructuras de clasificadores en serie. No es que fueran descubiertos repentinamente, pero los resultados de la organización en paralelo y la complejidad que se les suponía a los clasificadores en serie hicieron que la investigación sobre estos no proliferara de la misma manera.

Estas publicaciones en muchos casos comparaban el rendimiento de clasificadores análogos en paralelo versus serie sobre problemas concretos como el reconocimiento de escritura a mano ([10] [11] [12]), clasificación topológica [13] o verificación de firma [14].

Con el tiempo la investigación se aceleró en distintos campos, con especial atención en la clasificación biométrica, aprovechando características clave de la combinación en serie que se detallan en adelante. Este impulso hizo aparecer algunas nuevas ideas interesantes:

- Clasificación incremental de texturas en problemas tan dispares como la identificación por iris [15], la clasificación de vehículos a motor [16] o simplemente la clasificación de texturas de diferentes materiales [17].
- Sistemas biométricos de múltiples estados donde en caso de puntuación ambigua de un clasificador el sistema requiere la evaluación de una nueva característica biométrica mediante otro clasificador, repitiendo el proceso hasta N clasificadores o hasta obtener un resultado claro. Estos modelos han sido ampliamente analizados por diversos investigadores de la universidad de Cagliari ([8] [18]).
- La forma en la que operan los sistemas de identificación mediante diferentes características biométricas en cascada aventura un mayor índice de acierto en caso de ataques de suplantación. Las investigaciones en esta dirección muestran que, pese a no ser una solución perfecta, el porcentaje de impostores aceptados es menor ([4] [5] [14]).

- En caso de usar diferentes datos para cada clasificador, el registro de los datos puede resultar un problema como en el caso del sistema en paralelo con las características biométricas de un usuario. Por el contrario, la identificación de un usuario válido solo necesitaría 1 o 2 características biométrica para ser aceptado en el caso del sistema en serie, mejorando la usabilidad del sistema.
- Al no usar todos los clasificadores en cada uno de los procesos de clasificación, el uso de recursos se optimizan, sobre todo si el primer clasificador clasifica de forma rápida los casos más claros, dejando el resto de clasificadores para casos especiales.
- En un sistema con n estados, en el estado x con $x < n$, aun sin conocer los resultados de todos n clasificadores, se puede hacer una fusión parcial de los x resultados conocidos para tomar una decisión, ya sea un resultado final o evaluar el estado $x+1$. Este procedimiento es conocido como "serial fusión" o fusión de clasificadores en serie.

De la misma manera, con el aumento de la atención sobre este tema, también aparecieron algunos puntos negativos. Uno de los principales problemas reside en la dependencia de los clasificadores: si dos o más clasificadores son dependientes, el único resultado relevante será el del primero de ellos que sea evaluado, perdiendo los recursos destinados a evaluar los siguientes al no aportar información nueva [19]. Más aun, esto afectaría a la evaluación del sistema en caso de utilizar un tipo de fusión por consenso, afectaría negativamente a cualquier tipo de fusión basada en el teorema de Bayes.

Por otra parte cabe considerar que, al contrario que las agrupaciones de clasificadores en paralelo, la versión en serie toma decisiones parciales, sin tener en cuenta todos los datos disponibles o el patrón al completo.

A mediados de la primera década del presente siglo, las propuestas de aplicación para las agrupaciones de clasificadores en serie aparecían de forma asidua en publicaciones de carácter científico, acumulándose una respetable cantidad de información sobre su rendimiento y sus posibles problemas. Por otra parte algunos grupos de investigadores han intentado modelar los diferentes parámetros de estos conjuntos ([7] [19] [20]) para intentar arrojar algo de luz sobre arduo proceso de diseño que supone optimizar al máximo uno de estos sistemas para un problema en concreto.

Aun así, tanto estas publicaciones como las más actuales coinciden en su mayor parte en dos puntos:

- El diseño de cualquiera de las variantes de los sistemas de clasificación basados en clasificadores simples en serie resulta realmente complejo, siendo necesario en muchos casos aplicar test sucesivos para optimizar los diferentes parámetros que regulan el conjunto y realizar ensayos de prueba y error para dar con el entrenamiento y orden idóneo de cada uno de los clasificadores.
- Existe aún mucho trabajo por realizar sobre el diseño de este tipo de clasificadores y sobre el rendimiento de las muchas variaciones que pueden construirse, variando la manera de tomar la decisión final, los datos que toman cada uno de los clasificadores, la relación entre los distintos clasificadores...

3. Diseño

Las agrupaciones de clasificadores en serie varían drásticamente su comportamiento según una serie de parámetros a definir durante su diseño. Cada clasificador puede ser de diversos tipos como KNN, SVM, etc... En este caso se va a obviar este aspecto, utilizando como clasificadores individuales redes neuronales artificiales. Por otra parte, las variantes de diseño son en su mayoría compatibles entre sí, incrementando las posibilidades de diseño y su versatilidad, pero también la complejidad a la hora de construirlos.

3.1 DISEÑO DE LOS CLASIFICADORES INDIVIDUALES

El diseño de redes artificiales ha sido estudiado a lo largo de varias décadas tal y como se resume previamente en este documento. En este caso no haremos demasiado hincapié en el diseño interno de los clasificadores individuales.

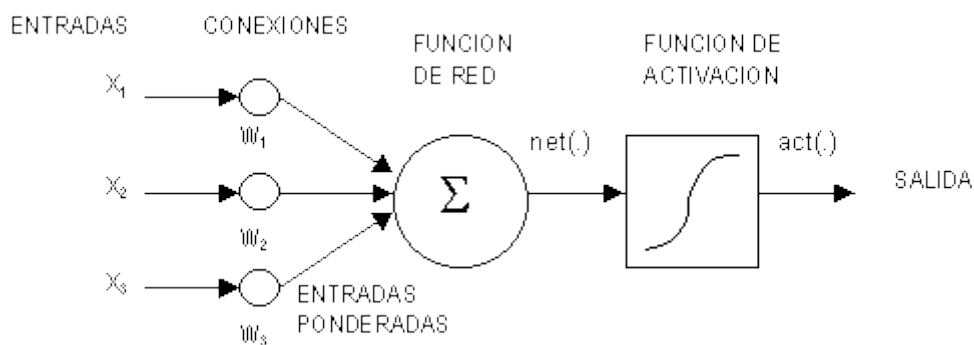


Figura 4: Esquema de una red neuronal de una sola neurona con función de activación y función de red

3.1.1 Capa de entrada

Las neuronas que reciben los datos se sitúan en una de las capas exteriores de la red, llamada capa de entrada. El número de neuronas en esta capa es algo sencillo de averiguar, ya que corresponde al número de parámetros de entrada del set de datos a utilizar. En algunos casos se puede añadir una neurona más, normalmente en estado permanente de excitación, si se quiere añadir un sesgo a la entrada de la red.

3.1.2 Capa de salida

La capa de salida es la otra capa exterior de la red, y al igual que la entrada, el número de neuronas está determinado por los datos a clasificar. En cada caso necesitaremos suficientes neuronas para que el sistema pueda representar cada una de las clases. En caso de tratarse de aprendizaje supervisado, tal y como será en las pruebas realizadas en el presente trabajo, el propio set de entrenamiento muestra las salidas que cada una de las neuronas de la capa de salida debe tomar para cada vector de datos en la entrada.

En muchos casos esta capa tan solo cuenta con una sola neurona, siendo suficiente para representar las clases de aceptación o rechazo de la clasificación de un vector de datos.

3.1.3 Capas ocultas

Las capas internas, entre la entrada y la salida de la red se denominan capas ocultas. Definir el número de capas ocultas de una red neuronal es un problema que ha hecho correr ríos de tinta a lo largo de los años. En caso de que el set de datos sea linealmente separable, la red neuronal no necesita ninguna capa oculta para resolverlo. Por otra parte, tal y como recoge el compendio NN FAQ [21], la precisión que se gana en la gran mayoría de los problemas con una segunda o tercera capa es normalmente insignificante, bastando con una capa oculta para resolverlos. Adicionalmente añadir más de una capa complica drásticamente el entrenamiento y dispara el uso de recursos.

En cuanto al número de neuronas en esta capa, la regla más común es utilizar un número de neuronas comprendido entre el número de neuronas en la capa de entrada y el de la capa de salida. Esto deja bastante margen para la experimentación según el tamaño del vector de entrada, por lo que en muchas ocasiones se opta por usar un número de neuronas igual a la media entre el tamaño de la capa de entrada y la de salida.

3.1.4 Función de propagación o de red

En cada una de las neuronas, una función se encarga de hacer acopio de cada una de las entradas y evaluarlas según los pesos de sus conexiones. La función más común en este caso realiza la suma ponderada de todas las entradas de la neurona y la presenta a la función de activación.

$$net_j(X, W_j) = \sum_{i=1}^n x_i w_{ij}$$

3.1.5 Función de activación

La función de activación define como será el comportamiento de cada neurona frente a los datos de entrada. Existen varios tipos según el resultado que queremos obtener, entre los cuales las más utilizadas son las siguientes:

- **Función de umbral:** implementar esta función significa fijar un límite a partir del cual la neurona se activa y por debajo del cual la neurona permanece inactiva. El resultado es una salida binaria con 1 o 0 como únicos posibles valores.
- **Función sigmoideal:** la función sigmoideal es una función continua que toma valores en el intervalo $[0, 1]$. Esto la hace especialmente útil como función de activación, pudiendo evaluar como de cerca de cada clase se encuentra un vector de datos según el clasificador.
- **Función sigmoideal bipolar:** se trata de una función continua con las mismas características que la anterior, salvo por el hecho de que su imagen se corresponde con el intervalo $[-1, 1]$. Este hecho puede resultar de especial utilidad al diseñar clasificadores que deban decidir entre aceptar, rechazar, o no decidir en caso de no obtener una puntuación suficiente o cercana a 0.

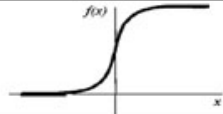

Tabla 1. Características de las funciones sigmoideal			
	Expresión analítica	Rango de las ordenadas	Representación Gráfica
Sigmoideal unipolar o logarítmica	$f(x) = \frac{1}{1+e^{-x}}$	(0, +1)	
Sigmoideal bipolar o hiperbólica	$f(x) = \frac{1-e^{-x}}{1+e^{-x}}$	(-1, +1)	

Tabla 1: Funciones sigmoideales, utilizadas muy comúnmente como funciones de activación

2015 Facultad de Eléctrica, Instituto Superior Politécnico José Antonio Echeverría, Cujae

3.2 ORDEN DE EVALUACIÓN DE LOS CLASIFICADORES

Previamente al análisis del orden de los clasificadores dentro del conjunto deben tenerse en cuenta dos cuestiones importantes:

- Aunque pueda resultar obvio, un clasificador incapaz de realizar clasificación alguna sobre el problema, o con una tasa de error alta, muy probablemente no resulte útil pese a enlazarlo con otros tantos clasificadores en cualquier orden. Normalmente un clasificador así tan solo significa un estado adicional y un desperdicio de los recursos asignados al mismo.
- Como se mencionó durante el análisis del estado del arte, y tal como se demuestra en varias publicaciones anteriores ([7], [19]), por muy bueno que sea el desempeño de un clasificador, si este es dependiente con un clasificador ya evaluado, la información que pueda proporcionar al sistema será irrelevante, añadiendo innecesariamente un estado adicional y retrasando la clasificación de algunas muestras.
- En algunos casos el orden está predefinido. Si se entrena un clasificador con los datos que el clasificador del estado anterior no puede clasificar, alterar su orden es un sinsentido. De igual manera, si se realiza una clasificación en serie aumentando la precisión de los datos en cada clasificador, el orden de los clasificadores es el correcto para lidiar con cada nivel de precisión de forma ordenada.

Tras estas aclaraciones, podemos pasar a analizar cómo afecta el orden de los clasificadores al rendimiento final del conjunto. Teniendo n clasificadores en el conjunto, tendríamos tantas posibilidades como permutaciones de n elementos, $n!$ por lo tanto. Esto hace que probar cada una de las posibilidades resulte una tarea realmente ardua, sobre todo si hay que entrenar cada uno de ellos cada vez como ocurre en algunos casos. Varias publicaciones han dedicado parte de su contenido, si no el artículo al completo, a intentar encontrar un orden concreto que pueda funcionar de forma sencilla, pero finalmente la única vía que parece viable es experimentar con cada uno de los clasificadores para obtener su rendimiento tal y como se recoge en la publicación de Gian Luca Marcialis, Fabio Roli y L. Didaci *Pattern recognition*[8], para después seguir al regla enunciada por los mismos autores en su trabajo *Personal identity verification by serial fusión of fingerprint and facematchers*[22]:

“la mejor combinación en serie posible entre dos clasificadores es siempre es el peor clasificador seguido del mejor clasificador”

siendo mejor el clasificador con mayor precisión clasificando los datos asignados al mismo. Por ejemplo, teniendo un clasificador que evalúa una característica biométrica con un 90% de precisión, y un segundo clasificador entrenado para clasificar una característica biométrica distinta con un 80% de precisión, la mejor combinación entre estos dos clasificadores sería utilizar en primer lugar el clasificador con un 80% seguido del clasificador con un 90%.

Por otra parte, una de las principales razones de adoptar una arquitectura en series es la optimización de recursos frente a su contrapartida en paralelo, siendo razonable pensar en utilizar los clasificadores más rápidos y más sencillos en los primeros estados del conjunto. Con este método, una buena parte de los vectores de datos serán clasificados minimizando el uso de recursos en estos primeros estados, dejando los clasificadores más costosos y complicados para los casos que verdaderamente lo requieran. Aun así, teniendo en cuenta que los clasificadores más rápidos y de menor complejidad suelen tener en contrapartida una precisión menor, podemos decir que la regla anterior confluye con los intentos de optimizar el rendimiento, por lo que la mejor combinación posible es normalmente la más óptima.

3.3 DISEÑO DEL CONJUNTO DE CLASIFICADORES EN FUNCIÓN DE LOS DATOS A PROCESAR

Una de las características que definen a este tipo de conjuntos es la forma en la que se distribuyen los datos a analizar entre los clasificadores. Estos datos pueden pertenecer al mismo set, a un set totalmente distinto, o incluso ser los mismos datos analizados en un estado anterior, haciendo que el comportamiento del ensemble sea totalmente distinto en cada caso.

3.3.1 Datos no clasificados en el estado anterior

En este caso, el primer clasificador de la cadena se entrena con todos los datos disponibles, identificando una o dos clases. En caso de no tomar una decisión clara en función de los umbrales de decisión configurados, este dato se añade al set de entrenamiento del siguiente clasificador, especializándolo en los datos del problema que el primero no es capaz de clasificar correctamente. Este procedimiento busca resolver objetivos más simples para construir a partir de una cantidad finita de pasos el clasificador que resuelva el problema, utilizando para ello clasificadores más rápidos y más sencillos que el clasificador que normalmente resolvería el problema completo en un solo paso.

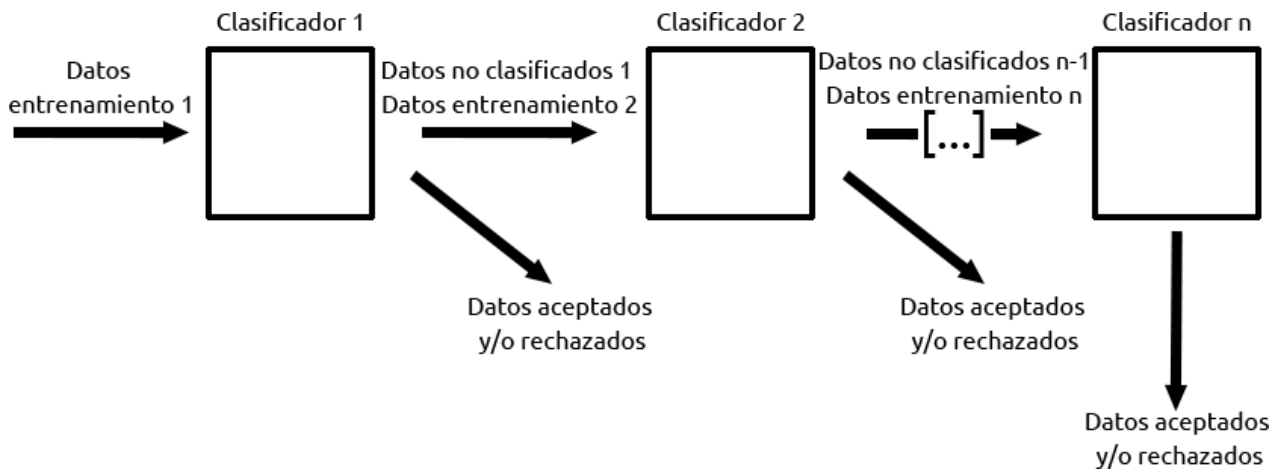


Figura 5: diseño con decisión obligatoria en el último estado

Este tipo de diseño presenta un problema típico: en función de los umbrales de cada clasificador es posible que en un punto del entrenamiento del conjunto, un clasificador no tenga datos suficientes para realizar un entrenamiento válido.

3.3.2 Datos incrementales

Algunos problemas requieren el proceso de vectores de datos muy costosos como los que se basan en análisis y clasificación de texturas, ya sea de un iris, una firma a mano o cualquier tipo de imagen. En estos casos, cuando la base de datos conal que comparar alcanza cierto volumen, comparar cada vector de datos con la base de datos haría recorrer una a una cada una de las texturas, haciendo recorrer la base de datos al completo en el peor de los casos. Al utilizar un solo clasificador, cada uno de los vectores debería estar guardado con la mayor resolución posible, por lo que es obligado llegar a un equilibrio entre precisión y tiempo de respuesta, perdiendo la precisión máxima que se podría obtener.

Utilizando varios clasificadores en serie podemos mejorar el rendimiento al clasificar este tipo de problemas a cambio de aumentar el tamaño de la base de datos con el fin de guardar diferentes versiones de cada muestra, una por cada clasificador a utilizar. Mediante esta técnica podemos utilizar un clasificador mucho más rápido y simple en primer lugar que actúe sobre una versión con menor resolución de la muestra original y de los vectores de la base de datos, haciendo que descarte cualquier vector de datos que sea claramente distinto o incluso lo acepte en primera instancia si las semejanzas son suficientes. En caso de que el clasificador no sea capaz de dar una decisión clara, el mismo par pasaría al siguiente clasificador, más complejo con el fin de recibir en sus entradas una versión más pesada de la misma textura. Este paso se repite hasta que se toma una decisión o se acaban los clasificadores, incrementando en cada paso el volumen de los datos, la complejidad del clasificador y análogamente la precisión obtenida. En este caso, cada clasificador debe entrenarse con diferentes versiones del mismo set de datos, incrementando el coste del proceso de entrenamiento al ser necesaria la creación de tantas versiones de cada vector registrado en la base de datos como clasificadores tiene el conjunto, por lo que podemos decir que este método incrementa notablemente el

volumen de los datos guardados en el sistema y requiere de un entrenamiento bastante más costoso que un solo clasificador, pero a cambio tiene la capacidad de mejorar mucho el rendimiento durante la fase de producción del sistema.

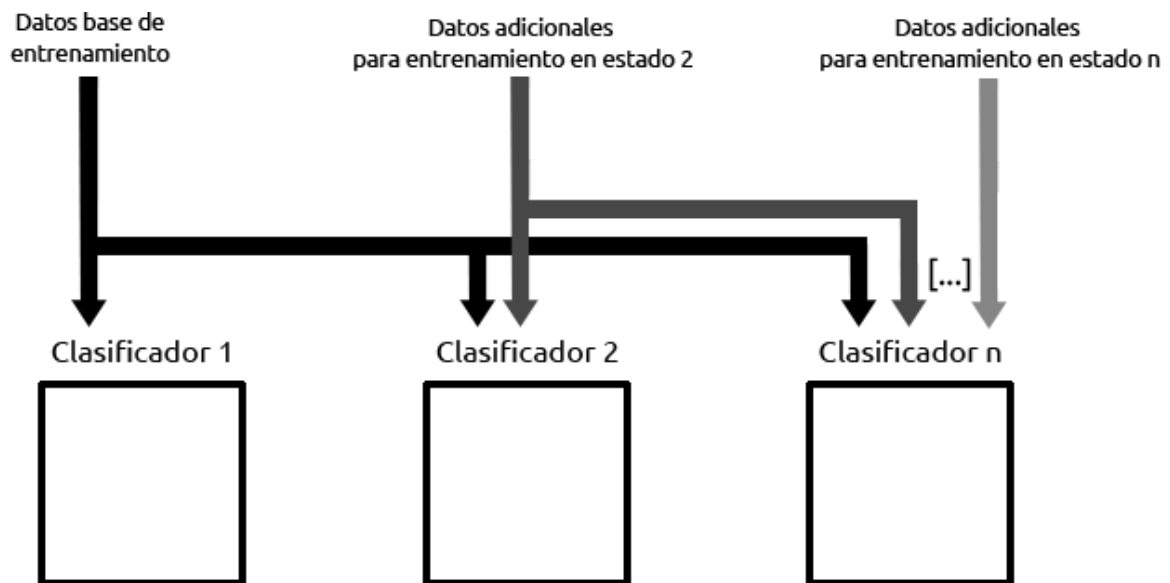


Figura 6: Esquema de entrenamiento con datos incrementales en cada estado

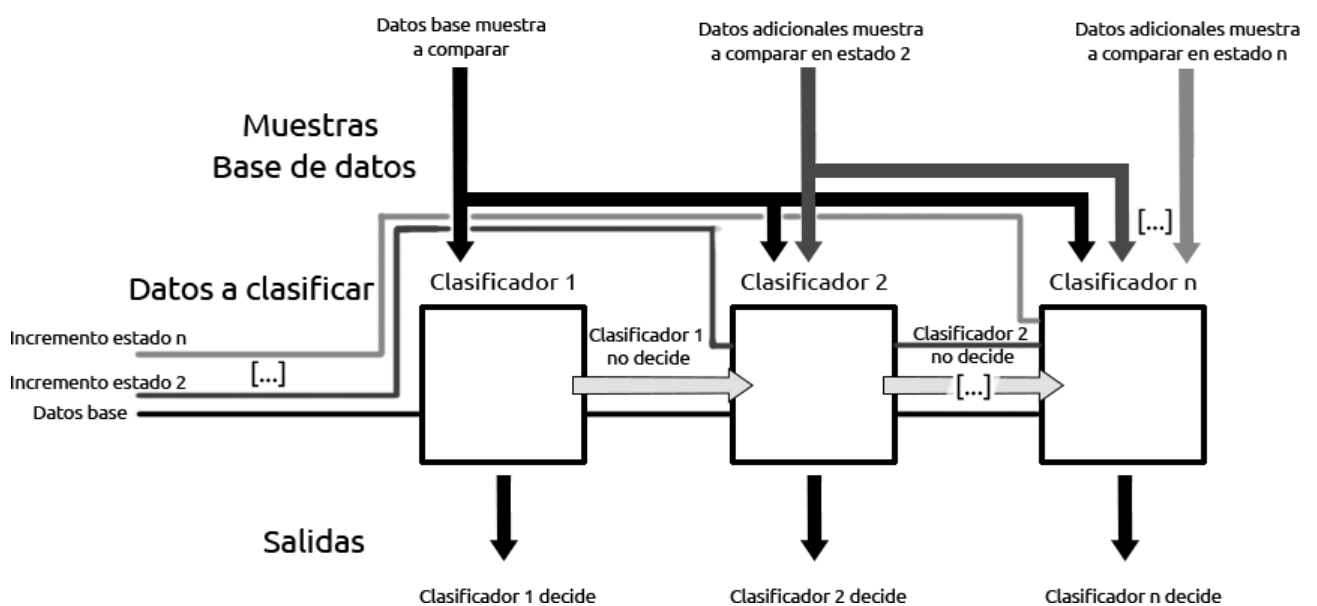


Figura 7: Esquema de clasificación de un conjunto de clasificadores en serie con datos incrementales

3.3.3 Datos independientes

Por último, uno de los métodos más populares en campos como el reconocimiento de patrones biométricos utiliza diferentes sets de datos para cada uno de los clasificadores del conjunto. La principal norma a tener en cuenta en este caso es que cada uno de los sets deben ser independientes posibles entre sí, pero a su vez todas las muestras de un set

deben estar relacionadas con una muestra del resto de sets, como por ejemplo diferentes patrones biométricos presumiblemente independientes entre sí obtenidos de un mismo sujeto, como la textura del iris, la huella dactilar y la forma de su cara. Mediante este método, resolver una entrada típica puede tomar uno o dos estados, mientras que el diseño hace que las entradas ambiguas o valores atípicos requerirán de más estados. En el caso de la biométrica, esto incrementa la seguridad del sistema ante ataques de suplantación, pidiendo cada vez más información al atacante, obligándolo a estar muy preparado para al menos no ser rechazado en cada una de las sucesivas pruebas a las que el sistema pueda someterle.

Siguiendo este método hay que tener en cuenta que cada clasificador puede decidir de forma positiva, de forma negativa o no decidir frente a una entrada dada, a no ser que el diseñador del sistema decida restringir alguna de estas decisiones por alguna razón expresa, como por ejemplo evitar que un clasificador pueda evaluar de forma negativa cualquier muestra para evitar el rechazo de datos válidos por error. En caso de que un clasificador no decida, se pasaría al siguiente. En cuanto al orden de los clasificadores, pueden seguirse las reglas enunciadas en el apartado 3.2 de este mismo documento, situando el clasificador menos preciso y a ser posible el más rápido como primera opción, siguiendo el mismo criterio para los restantes.

Las variantes son numerosas, y de nuevo corresponde al diseñador decidir las líneas de actuación si se llegan a evaluar todos los estados: obligar al último clasificador del conjunto a tomar una decisión o permitir que tras evaluar todos los clasificadores la salida del sistema pueda ser nula.

Una característica interesante de este sistema es su capacidad para operar aun cuando los datos de un estado no existan, pudiendo saltar al siguiente y obviar dicho estado de cara a tomar la decisión final.

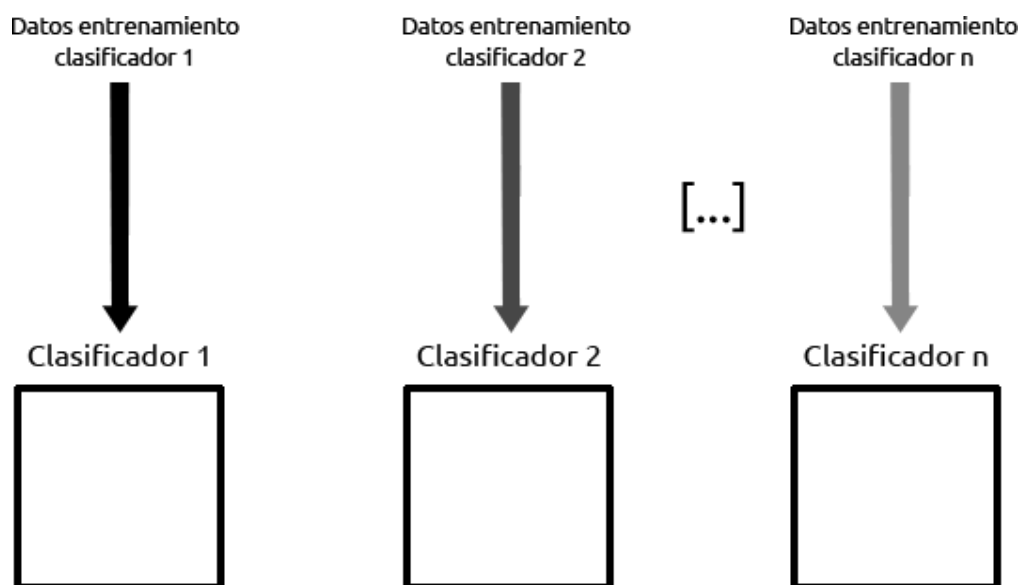


Figura 8: Esquema de entrenamiento de un conjunto de clasificadores en serie con datos independientes

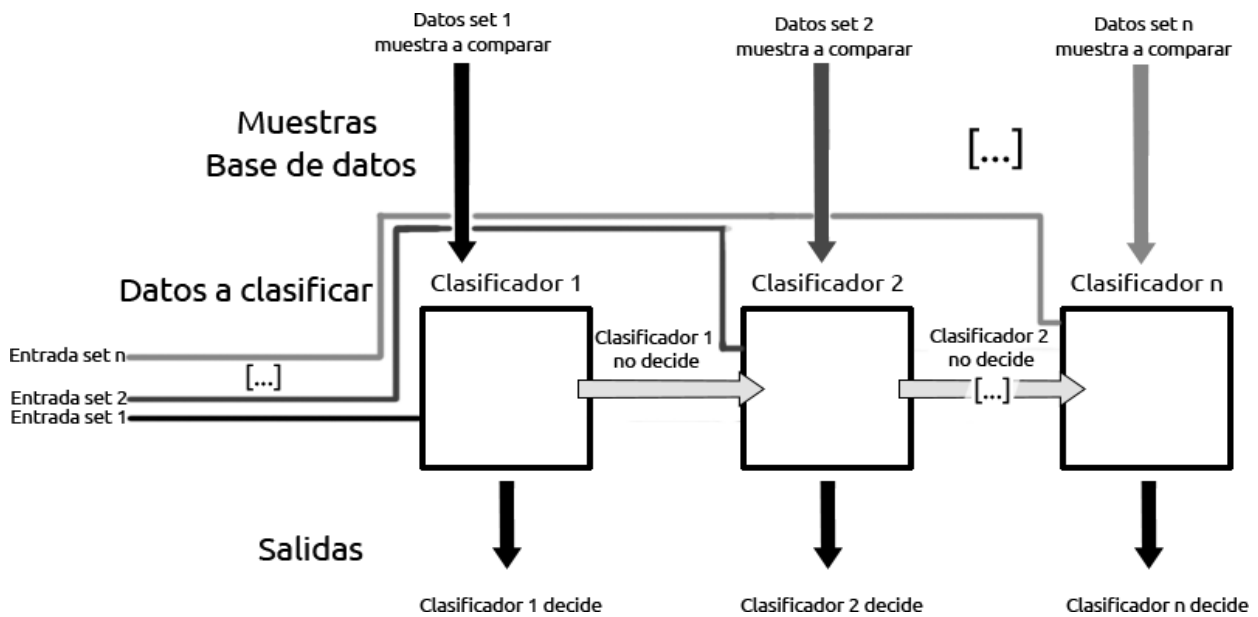


Figura 9: Esquema de clasificación de un conjunto de clasificadores en serie con datos independientes

Como puntos negativos, este método necesita del registro de todos los patrones a utilizar, lo cual es complicado en muchos casos, y en el caso de los patrones biométricos resulta poco agradable e invasivo, además de aumentar el almacenamiento necesario para el funcionamiento del problema. Adicionalmente el proceso de entrenamiento es costoso al ser necesario entrenar todos los clasificadores por separado uno a uno antes de construir el conjunto.

3.4 DISEÑO DEL CONJUNTO DE CLASIFICADORES SEGÚN LA DECISIÓN EN CADA ESTADO

Dentro del conjunto, cada clasificador toma una puntuación para el vector de datos presente en su entrada, pero posteriormente es necesario decidir qué hacer en función de dicha puntuación. Esta decisión puede tomarse de forma distinta en cada uno de los clasificadores que lo componen, elaborándose sistemas de decisión más complejos en función del problema.

Las funciones de activación de las neuronas juegan un papel importante aquí. La decisión simple puede funcionar como una función escalón con un cierto umbral interno, aunque lo normal es que se haga de forma directa con una función sigmoideal para controlar la puntuación de salida y con un umbral dentro del valor $[0, 1]$, normalmente próximo al valor medio del intervalo. Por el contrario, y aunque una función sigmoideal también cumpliría, la decisión múltiple invita a utilizar una función sigmoideal bipolar para tener un rango mayor de puntuación.

3.4.1 Decisión simple

La decisión simple es el tipo más básico, eligiendo entre dos posibles opciones en función del resultado del clasificador. Normalmente un clasificador típico funciona de este mismo modo, evaluando si la puntuación dada por el clasificador sobre la muestra supera un umbral configurado, y por tanto la clasificación es positiva, o por el contrario no alcanza el umbral y la muestra se rechaza. En nuestro caso podemos añadir otro tipo de decisión: pasar al siguiente estado.

De esta manera, una decisión simple en un clasificador serial tiene tres posibles diseños:

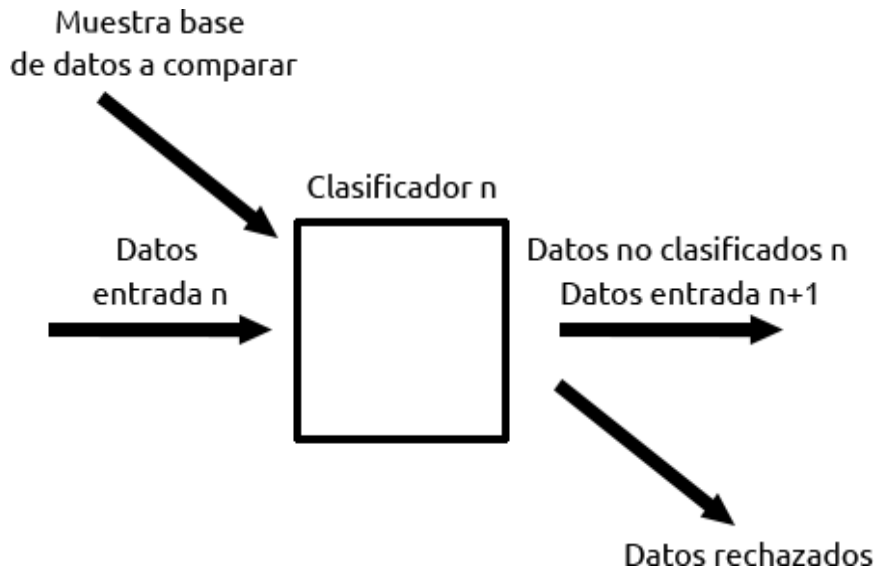


Figura 10: Esquema de un clasificador individual de decisión simple por rechazo. Este clasificador no puede aceptar una muestra, tan solo rechazar o saltar de estado.

- Aceptar o pasar al siguiente estado.
- Rechazar o pasar al siguiente estado.
- Aceptar o rechazar, útil para el último clasificador de la serie.

Este tipo de decisión es muy útil en sistemas de cascada que buscan evaluar de forma rápida las muestras más fácilmente clasificables para especializar el último clasificado en la resolución del problema habiendo filtrado gran parte de los vectores a comparar.

3.4.2 Decisión múltiple

Si añadimos un umbral adicional a la decisión simple obtenemos tres intervalos dentro del posible rango de puntuación, pudiendo elegir entre aceptar, rechazar o no decidir a partir de la puntuación obtenida en estado en el que se encuentre el conjunto, basándose de forma clara en el test SPRT de Wald y Pearson.

Aunque puede utilizarse en conjunto con otros clasificadores con decisión simple, lo más común es encontrar conjuntos de clasificadores en los que todos utilizan este tipo de decisión.

En conjuntos de clasificadores con sets de datos independientes es muy común utilizar este tipo de decisión, ya que al ser datos independientes, el sistema debe ser capaz de rechazar o aceptar los datos de entrada antes de pasar al siguiente estado, por lo que este método resulta idóneo al poder decidir además pasar a evaluar el siguiente set de datos si los resultados con el set evaluado no son concluyentes.

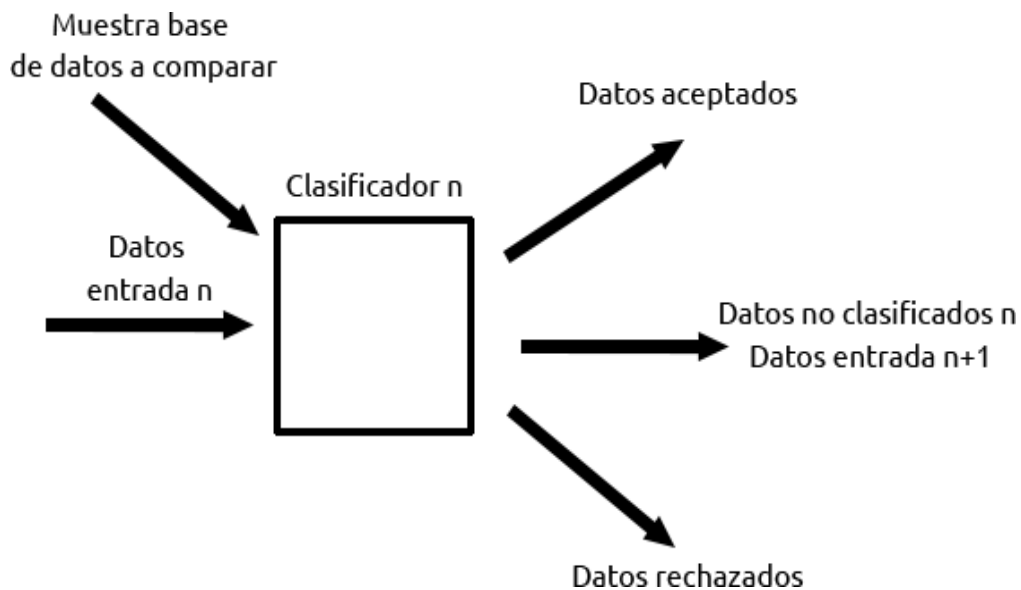


Figura 11: Esquema de un clasificador individual de decisión múltiple. Este clasificador puede aceptar una muestra, rechazar o saltar de estado.

3.4.3 Fusión de las puntuaciones

La fusión de las puntuaciones de los datos, o en este caso también conocida como fusión en serie dada la morfología que adoptan los clasificadores, no es un nuevo método de decisión que sustituya a los anteriores. En realidad la fusión utiliza la decisión simple o múltiple en cada estado a discreción del diseñador, pero en lugar de utilizar la puntuación de un solo clasificador para tomar una decisión, las puntuaciones de todos los clasificadores ya evaluados se combinan formando un vector de resultados. El método de fusión evalúa entonces el vector de resultados emitiendo una decisión, aprovechando de esta forma los resultados obtenidos en los estados anteriores para actuar con la mayor cantidad de información posible.

En adelante se detallan distintos métodos de fusión en función de su tipo y sus propiedades.

3.5 NÚMERO DE CLASIFICADORES EN LA ESTRUCTURA

3.5.1 Clasificadores de doble estado

Los conjuntos de clasificadores de doble estado constan, como bien su nombre indica, de dos clasificadores, formando dos estados en total. En los primeros ensayos sobre esta técnica, normalmente se utilizaban tan solo dos clasificadores para mostrar las propiedades de la combinación en serie de clasificadores. Estos clasificadores evaluaban en su mayoría diferentes sets de datos como patrones biométricos de huellas dactilares y morfología facial [8] o datos dependientes obtenidos con diferentes tecnologías [17], pudiendo añadir más clasificadores para convertirse en un clasificador multi-estado.

Por otro lado, el artículo "Two stage classification system combining Model-based and discriminative approaches" enuncia un tipo de clasificador de doble estado en el que dado el mismo set de datos, el primer clasificador busca simplificar el problema para el segundo, clasificando además aquellos datos sobre los que tenga un nivel de certeza elevado. Este modelo intenta aprovechar las características de distintos clasificadores para lidiar con dos de los problemas más básicos y conocidos de un clasificador: los datos atípicos y los datos ambiguos. Para ello, se aprovecha de las características de los clasificadores discriminativos, los cuales tienen un buen rendimiento a la hora de clasificar valores ambiguos al dividir el espacio en regiones de decisión de forma precisa, separando cada clase de forma clara. El problema en este caso son los valores atípicos, pues al dividir todo el espacio en regiones, estos valores quedan incluidos en una de las clases.

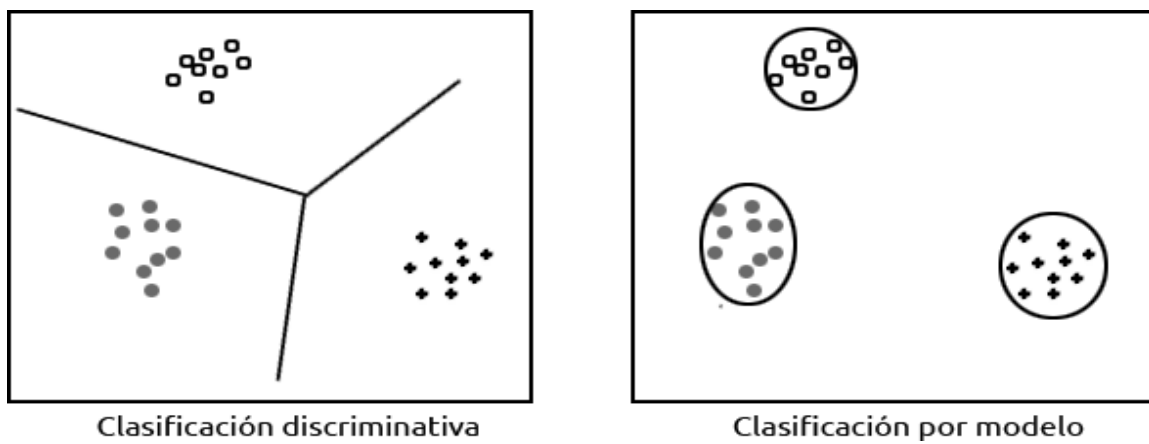


Figura 12: figura ilustrativa de la actuación de un clasificador discriminativo (izquierda) y un clasificador por clase o modelo (derecha). Las líneas negras continuas representan las áreas de decisión de cada clase.

La solución propuesta en este caso es la inclusión de un segundo estado en serie donde evaluar las entradas no clasificadas en el primero, pero en este caso utilizando un clasificador basado en la construcción de modelos estadísticos, los cuales tienen problemas con entradas ambiguas pero clasifican correctamente los valores atípicos. Presumiblemente los valores ambiguos quedarían resueltos en el primer clasificador, simplificando el problema y resolviendo parte del mismo para que el segundo clasificador complete el trabajo, obteniendo una mejora importante en la precisión.

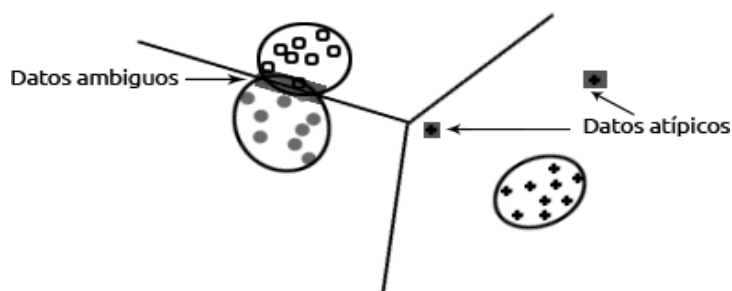


Figura 13: figura ilustrativa de la actuación de un clasificador discriminativo seguido de un clasificador por clase sobre datos atípicos y ambiguos.

3.5.2 Clasificadores multi-estado

En la categoría de conjuntos multi-estado entraría cualquier sistema con una cantidad n de clasificadores.

El número de clasificadores de una configuración en serie de los mismos depende del problema y del tipo de datos a evaluar en la gran mayoría de los casos. En caso de diseñar cada clasificador a partir de sets de datos independientes, el sistema podrá tener al menos tantos clasificadores como sets de datos haya disponibles. Si los clasificadores se entrenarán con los datos no clasificados en el estado anterior, el número de clasificadores queda a discreción del diseñador, aunque tras experimentar en este sentido, a partir del tercer clasificador los datos para entrenar un cuarto pueden ser insuficientes.

3.6 CONFIGURACIÓN DE LOS UMBRALES DE DECISIÓN DE CADA CLASIFICADOR

Uno de los parámetros principales a la hora de regular el comportamiento de cualquier clasificador complejo formado por clasificadores en serie es el umbral de decisión. Este umbral, o umbrales en caso de clasificadores de decisión múltiple, define cuando los datos de entrada no pueden ser clasificados en el estado actual y por tanto, pasarán al siguiente estado para que intente clasificarlos.

En el caso de un clasificador de decisión simple tan solo se implementa un umbral, de forma que en caso de que la puntuación del clasificador sobre los datos de entrada supere el valor de dicho umbral estos quedan clasificados con la clase oportuna, mientras que si la puntuación queda por debajo del clasificador los datos pasan a los siguientes estados hasta que uno de ellos pueda clasificarlo con suficiente confianza.

Los clasificadores de decisión múltiple implementan dos umbrales posibles al necesitar delimitar tres zonas dentro del rango de puntuación del clasificador. Por encima del primer umbral el clasificador aceptaría los datos de entrada como la clase esperada, mientras que por debajo del segundo umbral quedaría la zona en la cual el clasificador podría asegurar que los datos de entrada no pertenecen a dicha clase, quedando descartados. Entre estas dos zonas quedaría un rango de puntuaciones en el cual el clasificador no tiene la confianza necesaria para decidir, pasando en este caso los datos de entrada al siguiente estado.

3.6.1 Umbrales en función de las curvas ROC del clasificador

El cometido principal de un clasificador no es otro que asignar a un vector de datos la clase a la que pertenece sin errores. En la práctica este objetivo es mucho más complicado de cumplir, resultando imposible en muchos casos obtener un 100% de acierto.

Estos errores pueden deberse a datos genuinos rechazados por el clasificador, o por el contrario, datos falsos aceptados por el clasificador como válidos. Respecto a ambos tipos de errores, el clasificador puede definirse en función de la tasa de datos válidos rechazados (FRR) y su tasa de datos falsos aceptada (FAR).

A la hora de añadir un nuevo clasificador a un conjunto de clasificadores es difícil encontrar información sobre la tasa de cada tipo de error, aunque muchos clasificadores comerciales sí que suelen ofrecer las curvas ROC del clasificador. Esta representación muestra las tasas FRR y FAR del clasificador con respecto al valor del umbral de aceptación usado a la hora de clasificar, de tal forma que si el dicho umbral es muy alto, el riesgo de aceptar un dato falso es menor, pero por el contrario se ve aumentado el riesgo de rechazar un valor genuino.

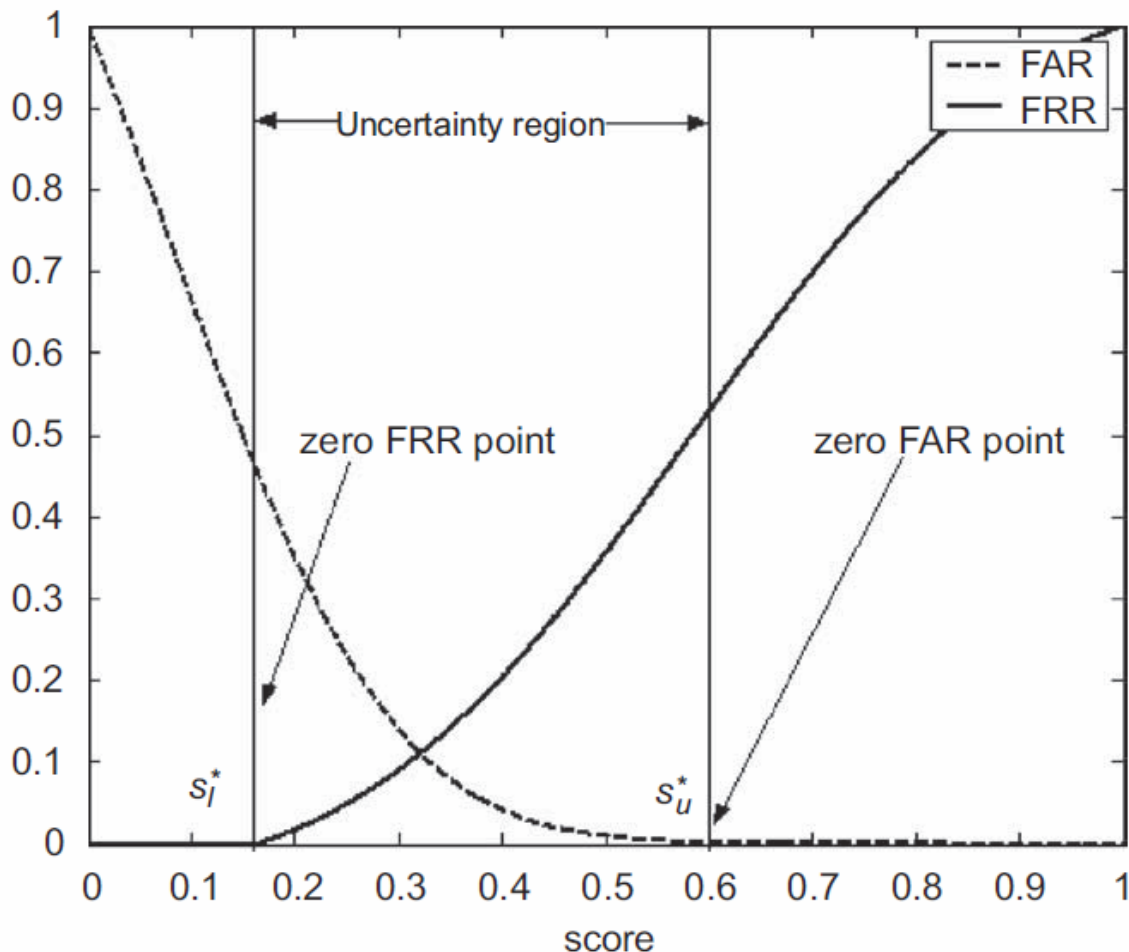


Figura 14: Curvas ROC en función del umbral de decisión del clasificador. Los puntos S_l^* y S_u^* se proponen como umbrales de decisión para el clasificador. Pattern Recognition, G.L.Marcialis, F. Roli, L.Didaci [8]

Estas curvas pueden usarse como referencia a la hora de configurar los umbrales de cada uno de los clasificadores tal y como se apunta en el artículo "Pattern Recognition" [8]. Si queremos evitar los errores, podemos configurar los umbrales de un clasificador de decisión múltiple según los valores donde FRR y FAR se hacen cero en las curvas que definen el comportamiento del clasificador. Si los datos fueran a ser clasificados tan solo mediante este clasificador estos valores en el umbral harían que una parte importante de los datos quedarán sin clasificación, ofreciendo un resultado muy pobre, pero en este caso estos datos serán clasificados por el siguiente clasificador de la cadena.

3.6.2 Compromiso entre precisión y desempeño

Mientras que en la teoría parece factible configurar un clasificador de forma que su tasa de error de cualquier tipo sea 0%, los problemas reales suelen presentar más problemas. Puede darse el caso que el rango de datos que el clasificador puede clasificar sin errores sea muy pequeño o casinulo, haciendo casi irrelevante todo un estado de la cadena. Si un alto porcentaje de los datos que llegan a un clasificador pasan al siguiente al no poder ser clasificados, el rendimiento de todo el conjunto se ve afectado de forma negativa.

Este problema es común en los primeros clasificadores de la cadena si se sigue la norma enunciada anteriormente, la cual insta a implementar el peor clasificador de la cadena en primer lugar. Las tasas de errores de este primer clasificador serán por tanto peores que

las del resto y en consecuencia, los rangos libres de error mostrados en sus curvas ROC serán más reducidos o en algunos casos inexistentes. Adicionalmente las curvas ROC pueden mostrar una precisión reducida y mostrar un valor nulo aunque el clasificador tenga una muy pequeña probabilidad de error en ese rango.

Existen pocas soluciones a este problema más allá del remplazo del clasificador en cuestión, aunque lo normal en este caso sería optar por paliar los efectos ya que más allá de la teoría es realmente difícil encontrar clasificadores ideales con un error nulo en para un rango amplio de valores. Las medidas a tomar para controlar estos efectos se basan en tolerar cierto porcentaje de error a cambio de que el clasificador pueda decidir sobre un rango mayor de los datos de entrada, mejorando el rendimiento en tiempo del conjunto a cambio de una pequeña pérdida de precisión. Por esta razón el análisis de las curvas ROC no es definitivo al definir los umbrales, siendo normalmente necesario un análisis posterior mediante experimentación en torno a los valores teóricos para definir un equilibrio a discreción del diseñador que cumpla con los objetivos previstos. Aun así, mediante el análisis de las curvas ROC puede evitarse un gran volumen de trabajo basado en prueba y error para la configuración del clasificador.

3.7 FUSIÓN DE LAS PUNTUACIONES DE LOS MARCADORES INDIVIDUALES

Una vez que el conjunto de clasificadores hay una última variación posible. La idea más básica del conjunto de clasificadores en serie muestra una salida igual a la salida del último clasificador consultado, pero en cualquier variación de las anteriormente vistas podemos aplicar al final un método de fusión que evalúe la salida de todos los clasificadores consultados, de tal forma que la salida del conjunto de clasificadores sea la de este proceso de fusión en lugar de la salida directa de un clasificador. Hay que tener en cuenta que esta fusión es parcial a diferencia de la fusión en conjuntos de clasificadores en paralelo, puesto que tan solo para un pequeño porcentaje del rango de datos de entrada se llegan a evaluar todos los clasificadores.

Existen diversos métodos de fusión, desde los más triviales a los que son suficientemente complicados como para suponer por sí mismos un nuevo estado en el sistema.

3.7.1 Funciones lógicas

Las funciones lógicas son la forma más básica de fusión. La función OREN este caso sería equivalente a dar la salida del último clasificador, ya que si el sistema ha continuado evaluando los estados sucesivos es debido a que los anteriores no han dado un resultado concluyente. Por otra parte la función AND requiere de una clasificación positiva en más de un estado para dar un resultado concluyente, modificando el comportamiento del sistema y aumentando la precisión. Este caso puede parecer muy parecido al caso con los clasificadores en paralelo, pero como hemos mencionado, la fusión es parcial en todo caso, y no tiene por qué necesitar de la evaluación de todos los estados para dar una solución.

3.7.2 Clasificación de patrones

Un método más costoso es la decisión sobre el vector parcial de puntuaciones mediante un clasificador adicional. Este método puede mejorar notablemente la precisión final del conjunto, pero en contrapartida requiere de un entrenamiento adicional, una mayor cantidad de recursos e incrementa el tiempo necesario para ofrecer un resultado. El

abanico de clasificadores que pueden usarse para este cometido es muy amplio, desde redes neuronales de mayor o menor complejidad a SVM.

3.7.3 Métodos estadísticos

Bajo ciertas condiciones pueden utilizarse reglas estadísticas basadas en la regla de Bayes, obteniendo un incremento de la precisión notable con un coste adicional pequeño durante la clasificación. Este método de fusión solo es de aplicación cuando los datos de entrada en cada uno de los estados son independientes, aunque esto es asumible por ejemplo en sistemas que evalúan diferentes características biométricas en cada clasificador, ya que cada una de estas características se supone independiente. En estos casos se puede aplicar la regla de decisión de Bayes, en la cual la probabilidad de que una entrada pertenezca a una clase se compara con un umbral predefinido para tomar una decisión, tal y como recoge la publicación [23], en la cual además se muestra este método como un tipo de fusión que evita el paso por estados innecesarios, mejorando el rendimiento.

3.8 ESTRUCTURAS MÁS COMUNES

Tal y como hemos visto los sistemas de clasificadores en serie admiten una gran cantidad de variaciones, incluyendo la adopción o no de un método de fusión a las salidas. En los estudios realizados sobre la fusión en serie de clasificadores se repiten ciertas estructuras comunes orientadas a clasificar datos distribuidos de una manera específica, dependiendo por tanto de las características del problema.

3.8.1 Estructura de decisión múltiple

Un conjunto de este tipo está compuesto en su totalidad de clasificadores de decisión múltiple, de forma que el proceso de clasificación actual puede acabar en cada estado aceptando la muestra o rechazándola. Es muy común que esta estructura esté compuesta además por clasificadores entrenados con datos independientes, ya que de esta manera se le da la posibilidad al sistema de decidir en ambos sentidos o no decidir con cada set de datos utilizado.

Esta estructura es muy común en sistemas multi-biométricos de alta seguridad, en el cual se demandan características biométricas adicionales al sujeto a identificar cuando el presente estado no puede decidir con confianza sobre la característica actual. De esta forma, si una característica de un usuario genuino muestra una gran cantidad de ruido, el conjunto de clasificadores puede consultar el siguiente estado, y si por el contrario un usuario malintencionado consigue no ser rechazado en el primer estado, podrá ser rechazado en cualquiera de los estados sucesivos, haciendo el sistema más seguro.

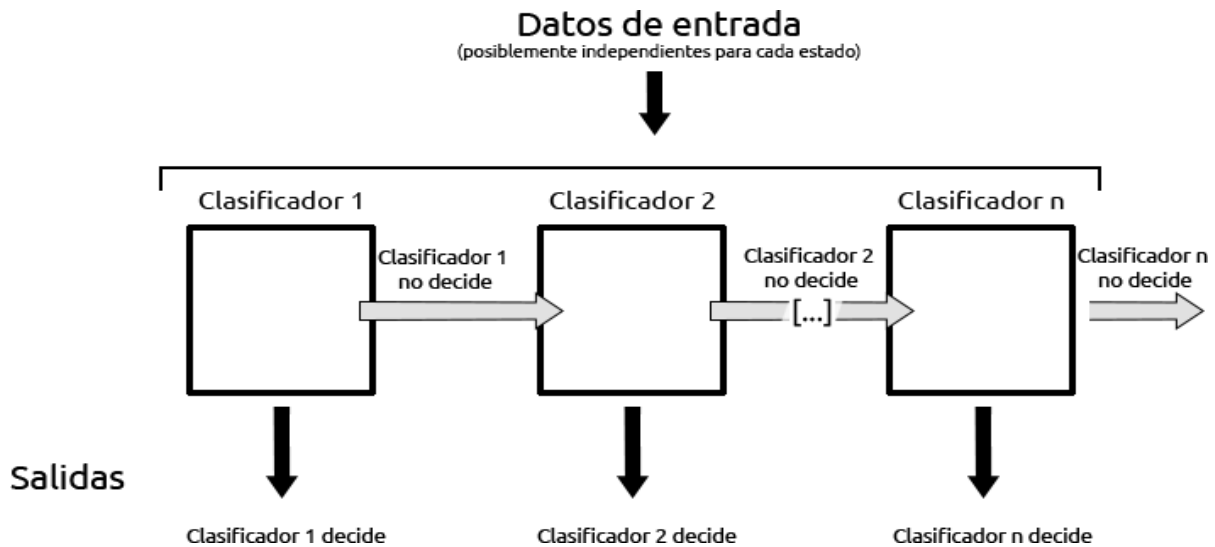


Figura 15: Sistema de decisión múltiple en todos sus clasificadores. Todos, incluyendo el estado final, pueden rechazar, aceptar o no decidir frente a unos datos de entrada dados

Esta estructura de clasificadores en serie además suele implementar algún método de fusión, pudiendo decidir aceptar una muestra de al cual varios clasificadores se hayan quedado muy cerca de la puntuación necesaria para aceptarla o por el contrario rechazar aquellas que muestren puntuaciones muy bajas y cercanas al umbral de rechazo en varios estados.

3.8.2 Estructura en cascada

La estructura en cascada es efectiva al encontrarse un problema en el que los datos se encuentran claramente desbalanceados. Pongamos por ejemplo un sistema en el que tan solo un porcentaje de los datos de entrenamiento deben ser clasificados positivamente, por ejemplo un 10%. Entrenar un solo clasificador con estos datos podría acarrear dificultades y sería necesario un tratamiento previo para intentar paliar el desbalanceo, aumentando el entrenamiento sobre los casos positivos, lo cual puede favorecer el sobre aprendizaje u overfitting.

En este caso, una estructura en cascada de rechazo eliminaría progresivamente la mayoría de los datos de entrada y tan solo una pequeña parte de los datos no válidos y el conjunto completo de los datos llegaría al último clasificador, resultando un problema más sencillo de clasificar y balanceado en condiciones idóneas.

Otro escenario donde la estructura en cascada ofrece un buen rendimiento es la identificación mediante muestras incrementales. Ejemplos de estos problemas son la identificación de la firma autografiada como genuina entre todas las muestras de una base de datos de tamaño considerable, o la identificación de un usuario mediante una textura extraída de su iris comparándola con el resto de usuarios registrados en la base de datos. En estos casos, comparar las muestras de la base de datos con la muestra de la entrada representa un coste elevado, mientras que el sistema en cascada compararía muestras de una resolución menor en el primer estado, incrementando la calidad hasta llegar a la calidad máxima guardada. De este modo el sistema rechazará las muestras menos

semejantes en los primeros estados con un coste menor, dejando los estados más costosos para un conjunto de muestras muy semejantes a la entrada.

Este tipo de sistemas pueden subdividirse a su vez en dos partes: el conjunto de clasificadores que depuran el set de datos y el clasificador final de identificación.

- La primera parte consta normalmente de un número finito de clasificadores de decisión simple con un umbral que permite unas tasas de error muy bajas. Estos clasificadores filtran los datos identificando la misma clase en todos los estados cuando la puntuación es suficientemente clara como para superar el umbral o pasando al siguiente si no se tiene la suficiente confianza para que sea clasificado.
- La segunda parte normalmente cuenta con un solo clasificador y recibe los datos restantes de la clase filtrada en la parte anterior y los datos pertenecientes a la segunda clase. Si este paso está compuesto por un solo clasificador, este suele implementarse mediante un clasificador de decisión simple que identifica ambas clases sin ningún rango que permita al clasificador no decidir sobre una entrada concreta, obligando al sistema completo a etiquetar cada entrada.

Adicionalmente este tipo de conjunto puede implementar una fusión parcial de las puntuaciones de los clasificadores de los clasificadores del primer paso, dedicados al filtrado de datos.

Los datos utilizados suelen ser incrementales, de tal forma que los clasificadores de filtrado actúen de forma progresiva con el fin de decidir sobre la mayor cantidad de datos utilizando los datos más baratos posibles de procesar, aunque en caso de ser independientes la estructura en cascada también sería efectiva.

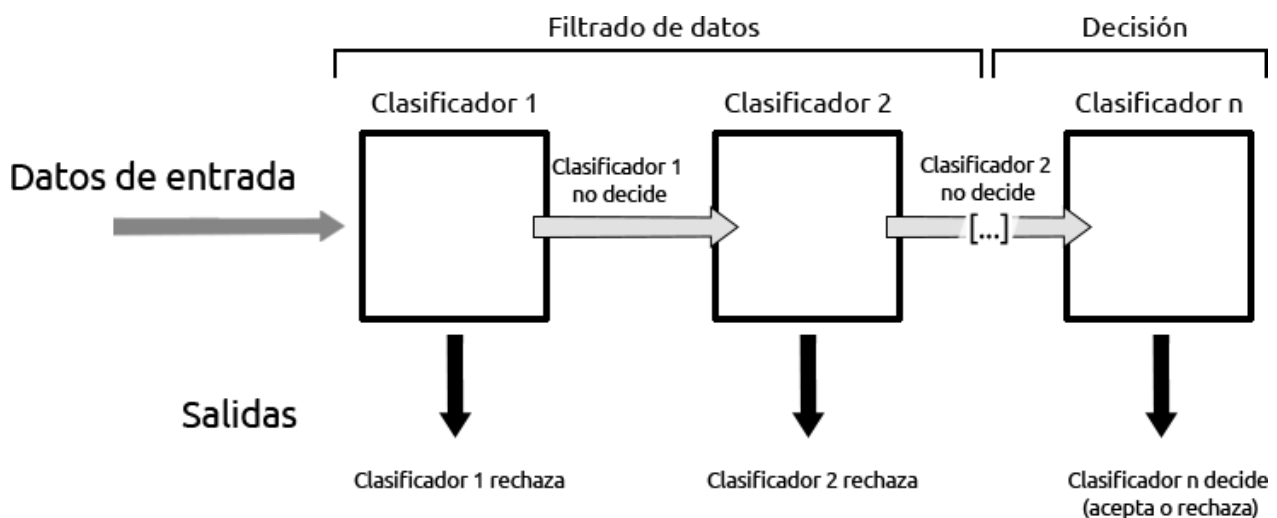


Figura 16: Sistema en cascada con filtrado por rechazo. En esta estructura tan solo el último clasificador puede aceptar los datos, siendo efectivo cuando los datos a clasificar están muy desbalanceados

4. Desarrollo

4.1 TECNOLOGÍAS UTILIZADAS

El lenguaje de programación utilizado para cada uno de los programas de pruebas creados para el presente trabajo es R. R es un lenguaje multiplataforma orientado al cálculo estadístico y a la representación gráfica de resultados disponible de forma gratuita bajo licencia GNU con gran capacidad y facilidad para crear representaciones de gran calidad de forma sencilla y un amplio abanico de técnicas estadísticas implementadas por defecto.

Adicionalmente, existen infinidad de librerías paquetes externos que incrementan las funcionalidades del paquete base de R permitiendo nuevos procesos o nuevas funciones gráficas. Para las pruebas en este proyecto se incluyen las librerías nnet y ROCR. Nnet es una librería que simplifica enormemente el trabajo con clasificadores basados en ANN, implementando todo el proceso desde el entrenamiento hasta la predicción sobre un objeto clasificador creado a partir de un set de datos y unos objetivos.

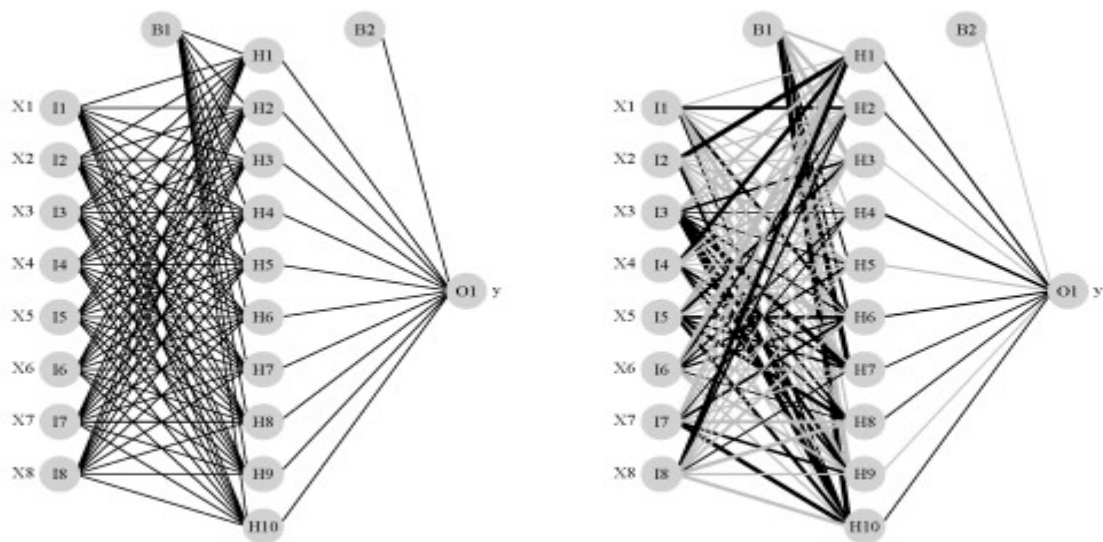


Figura 17: Representación de ANN creada mediante la librería nnet. <https://beckmw.wordpress.com>

Por su parte, ROCR implementa una amplia variedad de funciones para evaluar la precisión de un clasificador basadas en las curvas ROC del mismo.

4.2 ENTORNO DE DESARROLLO

Hay una gran variedad de entornos de desarrollo compatibles con R, pero en este caso el entorno a utilizar es RStudio. Este entorno multiplataforma integra la consola de ejecución, el área de desarrollo y el canvas para las representaciones gráficas en una sola ventana. Adicionalmente muestra el valor y el contenido de cada una de las variables y objetos activos en la sesión en cualquier momento, facilitando la búsqueda de errores y la monitorización del proceso.

4.3 FORMATO DE LAS ESTRUCTURAS DE DATOS AUXILIARES

El desarrollo de las entradas de datos de cada uno de los problemas requiere de un formato establecido que pueda ser interpretado con facilidad dado el volumen que pueden alcanzar en alguno de los problemas.

En este caso se optó por archivos CSV ampliamente utilizados para el almacenamiento de grandes volúmenes de datos previo procesado. Este formato consiste básicamente en archivos de texto plano con variables separadas por un separador definido y con una muestra por línea, exceptuando la primera en la cual se guardan la cabecera con la etiqueta de cada una de las variables.

5. Casos de prueba

5.1 TEST SOBRE CLASIFICADORES LINEALES Y MULTICAPA

La primera prueba realizada sobre un conjunto de clasificadores en serie es una introducción bajo un set de datos experimental muy lejos de un problema real. En este caso se busca ver hasta dónde se puede llegar con los conjuntos de clasificadores y si en caso de que un problema no fuera clasificable por un clasificador individual, podría un conjunto de clasificadores del mismo tipo abordarlo con éxito.

Para la primera parte de esta prueba se implementan 2 clasificadores lineales entrenados mediante la optimización de su error de clasificación. Estos clasificadores se implementan según los siguientes parámetros:

Neuronas entrada	Neuronas salida	Neuronas internas	Umbral rechazo	Umbral aceptación	Cte. aprendizaje	Max iteraciones aprendizaje
2	1	-	< 0.05	>0.95	0.05	5000
2	1	-	-	-	0.05	5000

Tabla 2: características de los clasificadores prueba 1

Mediante estos dos clasificadores lineales se intentará clasificar el problema no lineal representado en la figura siguiente, que en condiciones normales les sería imposible clasificar de forma individual.

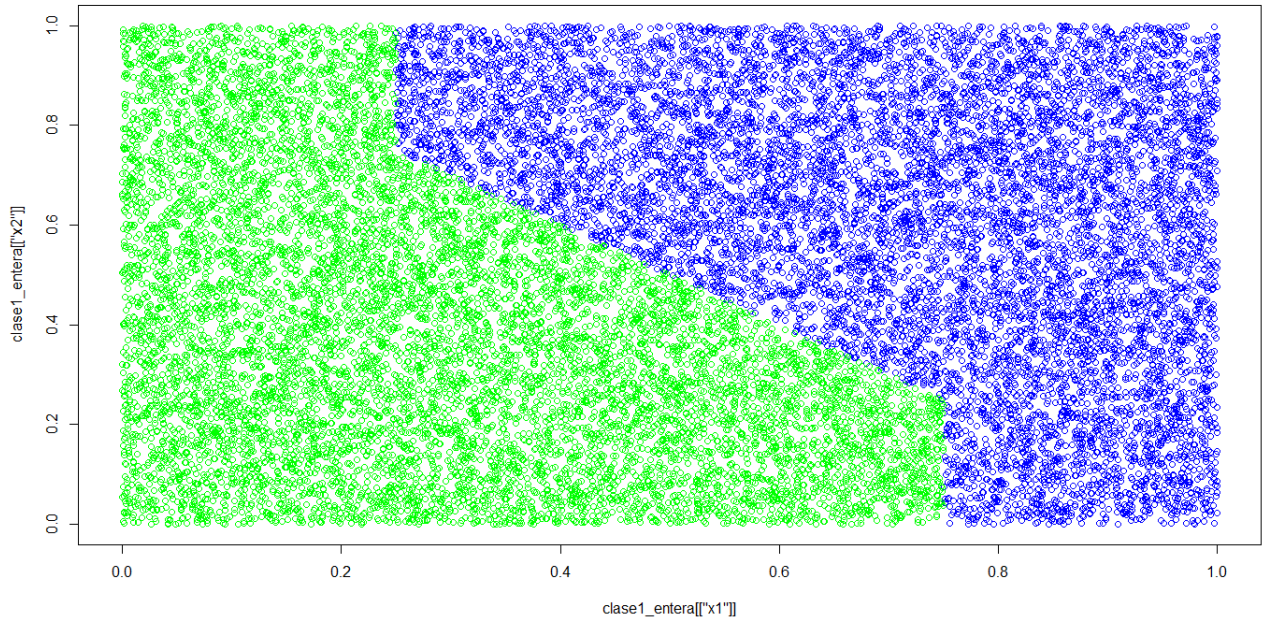


Figura 18: problema a clasificar, clase 1 representada en azul, clase 0 representada en verde.

Clasificador	Tipo de decisión	Activación	Datos	Método de fusión
1º	Múltiple	Sigmoidal	Set completo	Logica: OR
2º	Simple	Sigmoidal	Datos no clasificados por 1º	

Tabla 3: parámetros del conjunto de clasificadores prueba 1

Una vez definido el conjunto entrenamos el sistema con una muestra aleatoria de 5000 puntos entre 1 y 0 etiquetados como se muestra en la figura anterior. Hay que tener en cuenta que el segundo clasificador se entrenará con los datos que no pueda clasificar el primero, por lo que es necesario utilizar un set de datos suficientemente grande para que set de entrenamiento del segundo clasificador contenga suficientes datos.

Al final del proceso de entrenamiento, las predicciones del conjunto sobre una nueva muestra aleatoria de 5000 valores son las siguientes:

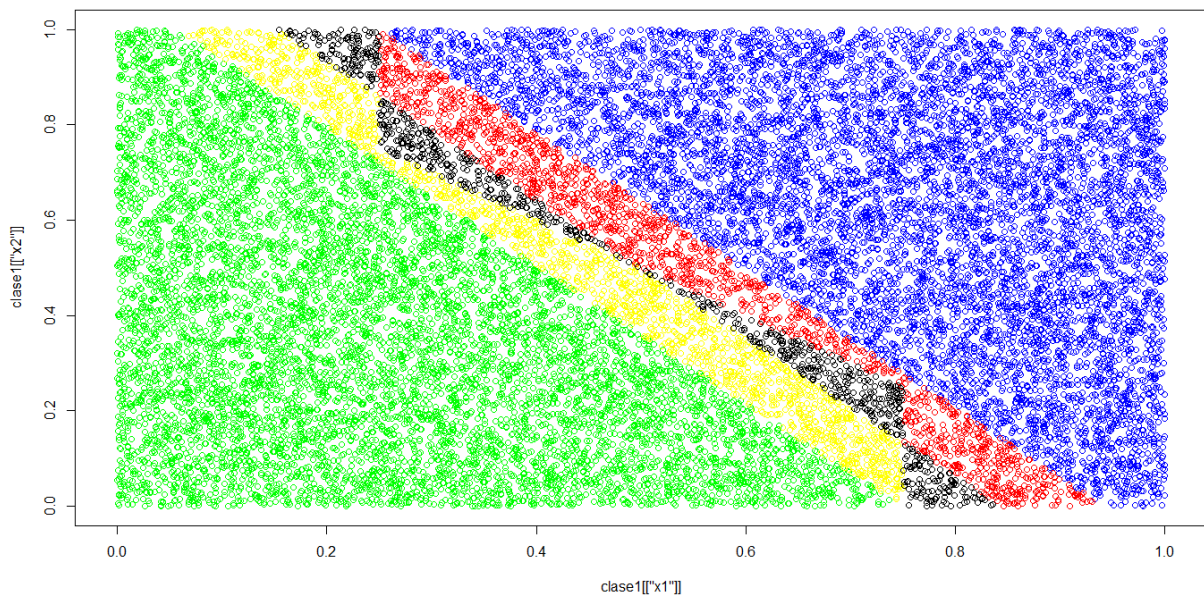


Figura 19: Resultado del conjunto de clasificadores lineales. Predicción de clase 1 en azul, predicción de clase 1 cercana al umbral en rojo, predicción de clase 0 en verde, predicción cercana al umbral en amarillo y errores en negro.

Atendiendo a los resultados, tras el resultado final puede decirse que se ha obtenido la misma solución que se obtendría con un solo clasificador lineal. Esto se debe a que el clasificador 1, pese a clasificar sin errores dentro de los umbrales configurados, no simplifica en ningún caso el problema de cara a la actuación del clasificador 2 al situarse de tal forma que minimiza el error de su clasificación durante el entrenamiento, pero no del conjunto.

La única manera de hacer que esta clasificación fuera posible sería entrenando el sistema de forma que el primer clasificador se situara de cara a reducir el error del segundo mientras anula el suyo mediante el ajuste de sus umbrales, siendo necesario el estudio de un posible nuevo método de entrenamiento.

5.2 CONJUNTO BI-ESTADO DE CLASIFICADOR LINEAL Y MULTICAPA

Utilizando el problema del ejemplo anterior, en este caso cambiaremos el segundo clasificador del conjunto para intentar resolver el problema mediante un clasificador multicapa manteniendo la estructura.

Neuronas entrada	Neuronas salida	Neuronas internas	Umbral rechazo	Umbral aceptación	Cte. aprendizaje	Max iteraciones aprendizaje
2	1	-	< 0.05	>0.95	0.05	5000
2	1	5	-	-	0.001	10000

Tabla 4: características de los clasificadores prueba 2

Clasificador	Tipo de decisión	Activación	Datos	Método de fusión
1º	Múltiple	Sigmoidal	Set completo	Lógica: OR
2º	Simple	Sigmoidal	Datos no clasificados por 1º	

Tabla 5: parámetros del conjunto de clasificadores prueba 2

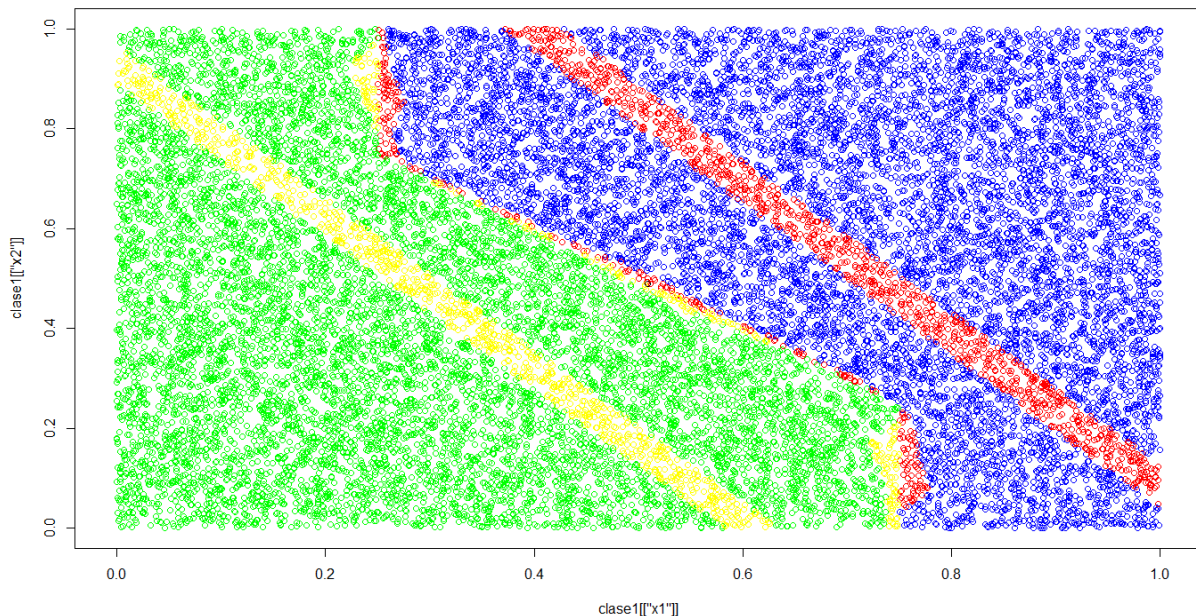


Figura 20: Resultado del conjunto de clasificadores. Predicción de clase 1 en azul, predicción de clase 1 cercana al umbral en rojo, predicción de clase 0 en verde, predicción cercana al umbral en amarillo y errores en negro.

En este caso puede verse claramente los límites de la clasificación llevada a cabo por el primer clasificador, delimitados por valores cercanos a ambos umbrales. Dentro de esos límites puede verse la predicción del segundo clasificador con una tasa de error muy pequeña en las áreas de unión de ambas clases, realizando en conjunto una muy buena clasificación del problema.

Que el sistema sea capaz de clasificar el problema no es relevante en este caso, ya igualmente hubiera sido clasificado utilizando tan solo el clasificador multicapa. Teniendo en cuenta que el clasificador multicapa utilizado no es muy diferente en rendimiento a la hora de predecir, tampoco existe un ahorro de recursos al implementar este tipo de estructura, pero en este caso se puede ver claramente cómo el clasificador lineal hace gran parte del trabajo (59,9%) pese a ser un modelo muy simple, pudiendo determinar los datos que pasa al segundo clasificador de tal manera que el error de clasificación en este paso sea nulo.

Extrapolando esto a clasificadores más complejos, podemos prever la efectividad de este método si conseguimos realizar una parte importante de la clasificación de una forma más barata, dejando los métodos complejos para las muestras que de verdad los requieran.

Clasificador	% Error	% datos clasificados
1°	0	59.95
2°	0.005	40.05
Total	0.005	100%

Tabla 6: resultados de la predicción en prueba 2 en cada uno de los estados y predicción final.

5.3 CONJUNTO BI-ESTADO FRENTE A PROBLEMAS REALES

En este caso realizaremos dos pruebas sobre dos sets de datos diferentes. El primero son los datos de un número finito de donantes de sangre mediante el cual se busca predecir si los donantes volverán a donar en el mes siguiente para predecir las necesidades futuras. El segundo se trata de un set de datos sobre los síntomas de diferentes pacientes registrados con el fin de elaborar un modelo capaz de predecir si el paciente padece de arritmia cardiaca.

5.3.1 Predicción de donantes

El problema que intentamos predecir aquí es de una alta complejidad. Los datos que se aportan son útiles, como la asiduidad con la que acude el donante, la última fecha en la que acudió, el tiempo que lleva como donante asiduo... pero aun así es muy complicado predecir el comportamiento de un donante y de cualquier persona en general.

Neuronas entrada	Neuronas salida	Neuronas internas	Umbral rechazo	Umbral aceptación	Cte. aprendizaje	Max iteraciones aprendizaje
2	1	-	< 0.1	>0.9	0.05	5000
2	1	5	-	-	0.001	10000

Tabla 7: características de los clasificadores prueba 3

Clasificador	Tipo de decisión	Activación	Datos	Método de fusión
1°	Múltiple	Sigmoidal	Set completo	Lógica: OR
2°	Simple	Sigmoidal	Datos no clasificados por 1°	

Tabla 8: parámetros del conjunto de clasificadores prueba 3

Tras evaluar la predicción del conjunto de clasificadores, tal y como cabría esperar después de analizar nuestra primera prueba con los clasificadores lineales, ocurre exactamente lo mismo en este caso. Si un clasificador no puede clasificar un problema, otro clasificador igual tampoco mejorará la clasificación aunque se centre exclusivamente en la parte más complicada del problema. Para mejorar la clasificación necesitaríamos más datos en cada estado o un clasificador diferente que pueda aportar un método más útil para el problema en concreto.

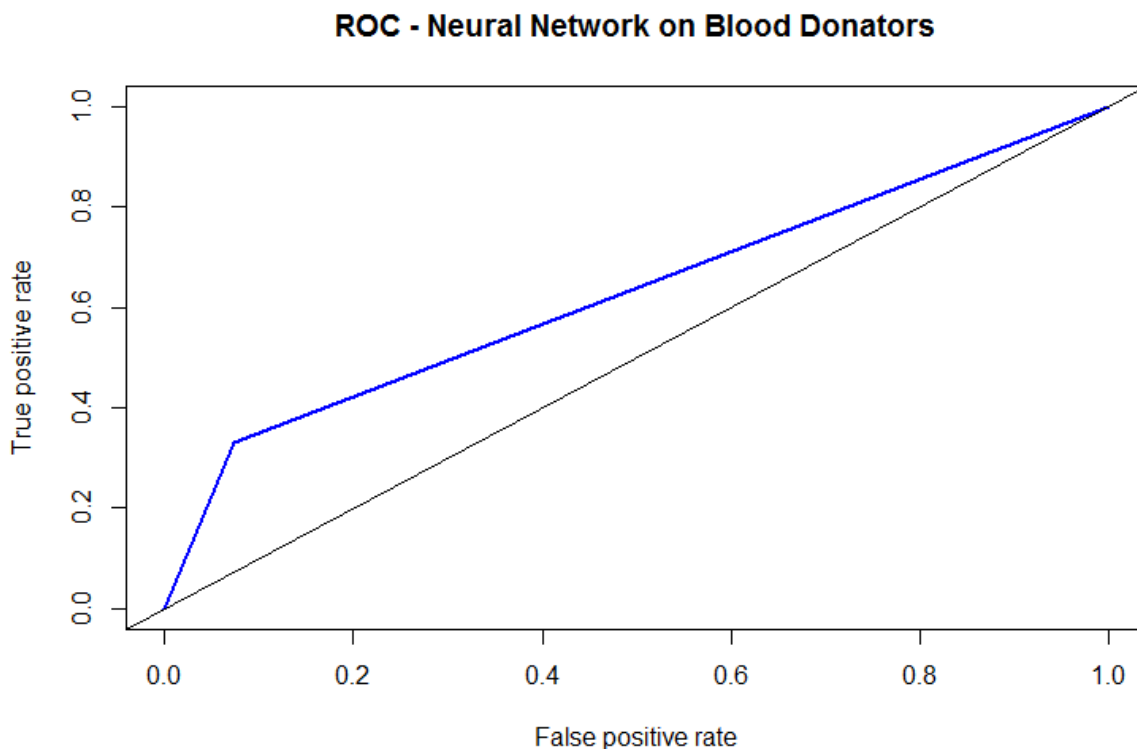


Figura 21: Curva ROC del primer clasificador. Puede verse como el rendimiento del mismo es mejor que la función aleatoria representada en gris, aunque con un error importante.

Clasificador	% Error	% datos clasificados
1°	20.32	50.23%
2°	18.75	49.77%
Total	20,43%	100%

Tabla 9: resultados de la predicción en prueba 3 en cada uno de los estados y predicción final.

5.4 DETECCIÓN DE FALSAS ALARMAS EN SISTEMAS COMPLEJOS

El sector ferroviario cuenta con sistemas de seguridad de una complejidad muy alta que velan por la seguridad de los viajeros, la tripulación, las mercancías y el material rodante. Uno de estos sistemas son los detectores de impacto vertical, capaces de medir la fuerza ejercida por las ruedas de un convoy y decidir si la rueda se encuentra en condiciones de rodar o por el contrario, tiene problemas que puedan provocar vibraciones, golpes en el carril y en casos muy extremos, posibles accidentes.

Como cualquier sistema de alta prioridad, uno de las principales problemas que el sistema debe enfrentar es la gestión de falsas alarmas. Una falsa alarma puede ocasionarse por el fallo de un sensor o una mala calibración del sistema que provoque lecturas de fuerzas desvirtuadas y superiores a la fuerza real. Una falsa alarma puede detener un tren y

provocar graves perjuicios a diferentes actores, como empresas gestoras de ferrocarriles, empresas de transporte de mercancías, o incluso a los propios viajeros que pueden sufrir retrasos durante el trayecto.

Diferenciar estas falsas alarmas es una tarea sencilla para cualquier técnico con cierto conocimiento sobre el sistema, pero resulta inviable revisar las señales de cada eje de cada convoy que atraviesa cada uno de los sistemas instalados alrededor del mundo. Por esta razón es necesario el estudio y detección automatizada falsas alarmas, detectando las máximas posibles para disminuir el impacto causado.

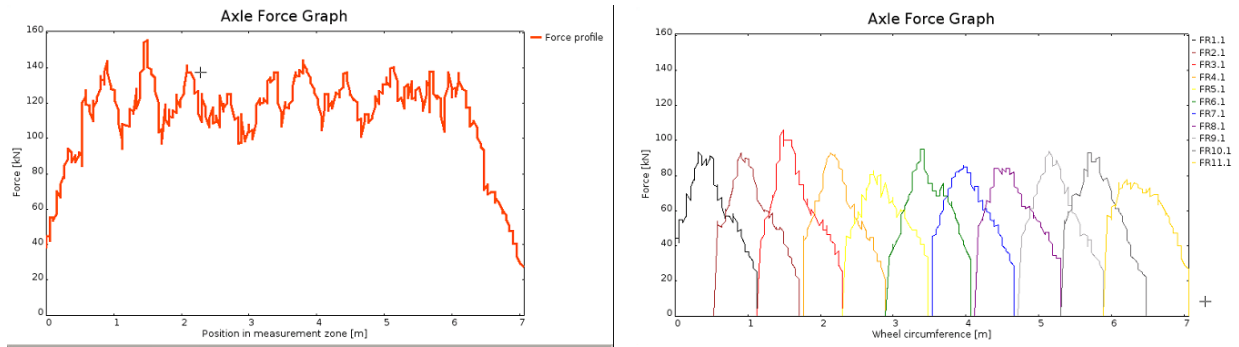


Figura 22: Señal resultante (izquierda) y señal de cada uno de los sensores (derecha) en el paso de un eje sin daños

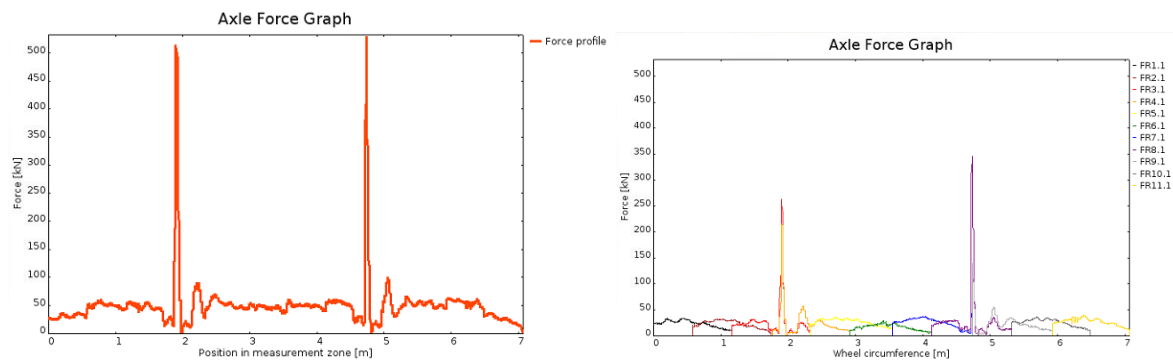


Figura 23: Señal resultante (izquierda) y señal de cada uno de los sensores (derecha) en el paso de un eje con un plano fuerte. Puede verse como el sistema registra dos revoluciones de la rueda.

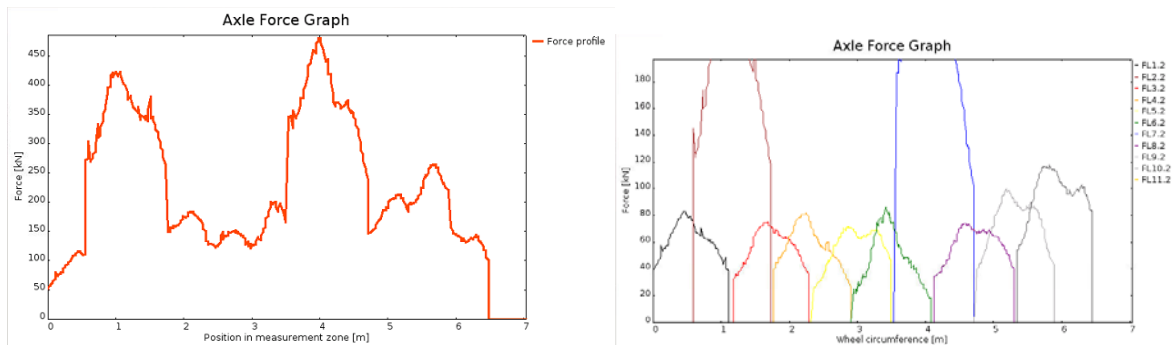


Figura 24: Señal resultante (izquierda) y señal de cada uno de los sensores (derecha) en caso de una mala calibración del sistema. Los sensores quedan fuera de escala desvirtuando la señal

Para detectar este tipo de alarmas se debe considerar en primer lugar el gran desbalanceo de los datos a clasificar. Las alarmas no son comunes, y por tanto una falsa alarma es aún menos común con una proporción aproximada de 1 cada 1000 ejes. Por otra parte los equipos de alta prioridad requieren un tiempo de respuesta corto, por lo que conviene procesar los ejes de la forma más barata posible en cada caso, ya sea posible sin realizar cálculos adicionales sino se requieren.

Con este contexto, la estructura a idónea parece ser en cascada, eliminando en los primeros estados la mayor parte de los datos mediante clasificadores más baratos y dejando los datos más complicados para los últimos estados.

5.4.1 Diseño de los clasificadores

- Clasificador 1: clasificador sencillo basado en datos comunes que el sistema utiliza durante el análisis, como la fuerza media, la fuerza máxima, la velocidad del tren o la desviación típica de los valores de fuerza a lo largo del paso del eje. Este clasificador debería servir para aceptar todos los ejes sin problemas y dejar solo los que sean susceptibles de dar cualquier tipo de alarma para los siguientes estados.
- Clasificador 2: análisis del rango de la fuerza máxima en la señal resultante. Normalmente las alarmas por problemas en las ruedas provocan picos muy bruscos en la señal resultante, mientras que un sensor fuera de escala mantiene valores semejantes en un rango de valores alrededor de su valor máximo. La entrada de este clasificador serán n valores del rango del valor máximo de fuerza medido a lo largo del paso del eje, dependiendo n de la velocidad del tren y estando comprendido entre 5 y 10.
- Clasificador 3: Cuando un sensor queda sobredimensionado, los sensores adyacentes normalmente mantienen valores normales, quedando muy por debajo de este. Sin embargo, cuando hay una alarma real, todos los sensores en el rango del impacto registran el incremento brusco de fuerza. Por ello, este clasificador recibe valores de todos los sensores que actúan en el rango del valor máximo, pudiendo clasificar la alarma en función de la diferencia de fuerza entre ellos. Este análisis resulta costoso para el sistema.

Neuronas entrada	Neuronas salida	Neuronas internas	Umbral rechazo	Umbral aceptación	Cte. aprendizaje	Max iteraciones aprendizaje
4	1	4	-	0.5	0.001	10000
5-10	1	10	-	0.4	0.01	10000
11	1	20	-	0.1	0.001	10000

Tabla 10: características de los clasificadores prueba 4

Clasificador	Tipo de decisión	Activación	Datos	Método de fusión
1º	Simple	Sigmoidal	Independiente	Clasificador (NN multicapa)
2º	Simple	Sigmoidal	Independiente	
3º	Simple	Sigmoidal	Independiente	

Tabla 11: parámetros del conjunto de clasificadores prueba 4

5.4.2 Rendimiento de los clasificadores

Cada uno de los clasificadores será entrenado por un set de entrenamiento al completo pero utilizando para cada uno de ellos datos diferentes, tal y como hemos visto anteriormente. Durante el entrenamiento se puso fuerte empeño en intentar que los datos estuvieran balanceados

Mediante el análisis de sus curvas ROC podemos ver como los dos primeros clasificadores tienen una precisión peor que el último de ellos, por lo que el orden es correcto según la regla vista anteriormente en este documento.

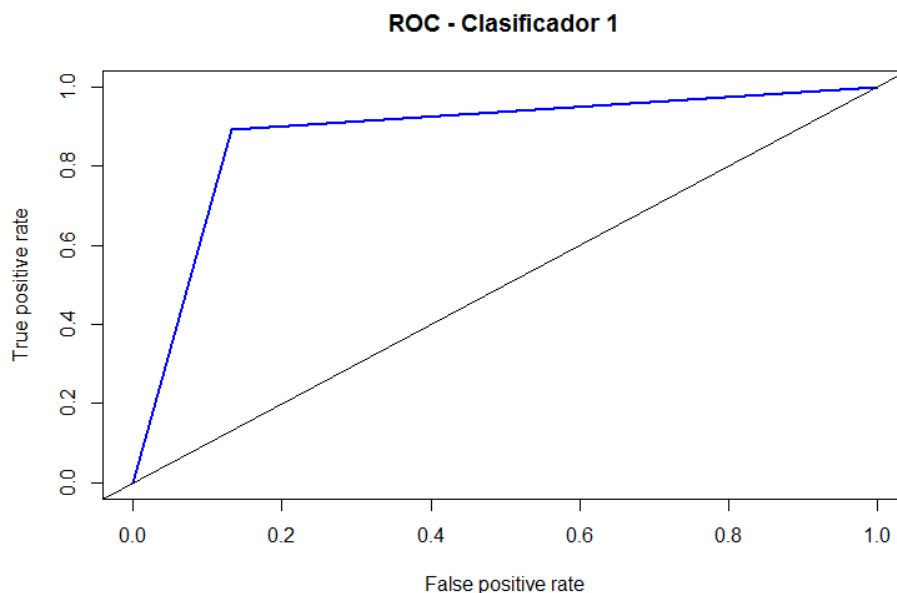


Figura 25: Curva Roc del primer clasificador, Positivos verdaderos frente a falsos positivos.

El primer clasificador muestra una precisión superior a la decisión aleatoria, con un porcentaje aceptablemente bajo de falsos positivos. La situación del umbral de aceptación se situará cerca del punto de corte, pero en este caso habrá que evaluar de forma experimental que umbral ofrece un rendimiento mejor al conjunto al no poder eliminarse los falsos positivos.

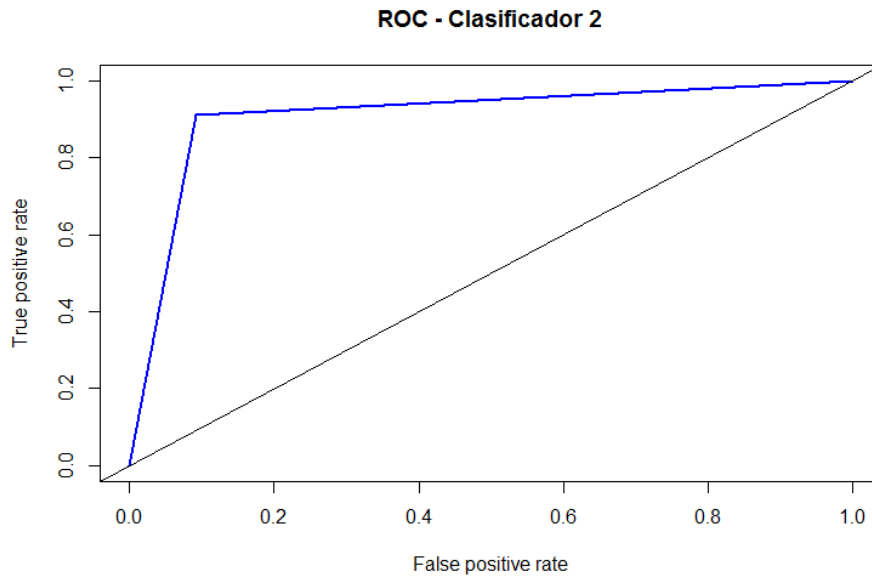


Figura 26: Curva Roc del segundo clasificador. Positivos verdaderos frente a falsos positivos.

El segundo clasificador muestra un rendimiento ligeramente superior al primero según sus curvas ROC, pero de nuevo estamos ante el mismo caso. En ningún caso podemos realizar una clasificación perfecta, luego el umbral debemos elegirlo de forma experimental, aunque sepamos que estará seguramente por encima de 0.2.

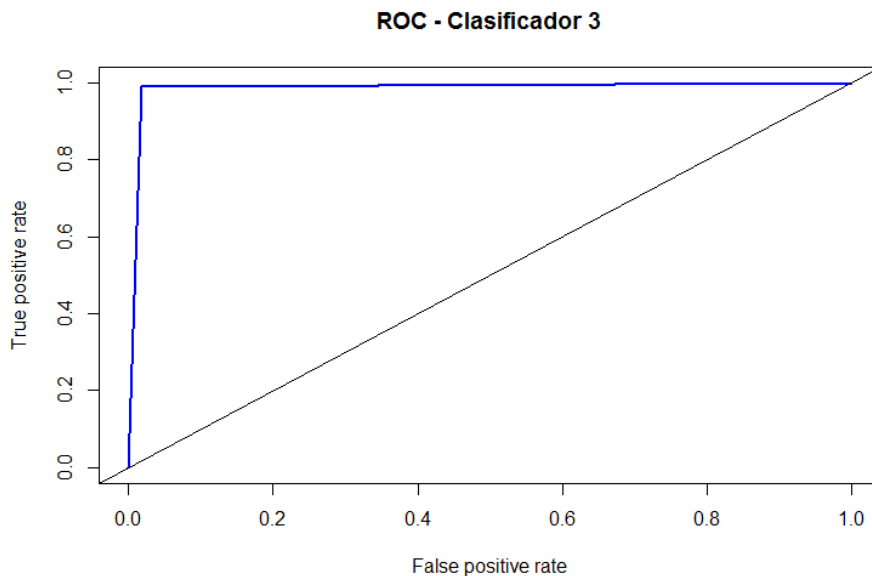


Figura 27: Curva Roc del tercer clasificador. Positivos verdaderos frente a falsos positivos.

Por último el tercer clasificador, y en este caso sí que es capaz de ordenar su parte del problema de forma perfecta o casi perfecta utilizando un umbral muy pequeño. Se utilizará como umbral 0.1 para sentar un punto de partida.

5.4.3 Estructura final del conjunto

El conjunto final sigue, tal y como se decidió desde un primer momento, una estructura de cascada. Esto permite discriminar rápidamente una de las clases, preferentemente la más común y ganar cierto rendimiento como vimos en el apartado de diseño. (Ver Figura 16)

Adicionalmente se añade un proceso de fusión para intentar mejorar la clasificación del sistema, entrenado con las salidas de los clasificadores sobre los distintos sets de entrenamiento individuales. Este método de fusión se aplica tan cuando los 3 clasificadores han actuado, evitando su aplicación en situaciones de información parcial que pueden desvirtuar la clasificación.

5.4.4 Resultados

Tras el entrenamiento los tres clasificadores mostraron una tasa de error similar a lo esperado tras el análisis de las curvas ROC, mejorando la precisión según se avanzan estados en el sistema. Hay que tener en cuenta que estos errores están calculados sobre todo el set de datos, y no sobre el rango que el clasificador evaluará dentro del conjunto, por lo que muestran el caso peor.

Clasificador	% Error	% datos clasificados
1º	11.8%	54.10%
2º	7,8%	36.71%
3º	1.6%	9.19%

Tabla 12: Resultados individuales por clasificador de la prueba 4

Por otra parte, los resultados sobre el set de prueba tampoco mostraron sorpresas, siendo el error semejante al peor clasificador del conjunto. La sorpresa estuvo en la notable mejora de la precisión tras la aplicación de la fusión con las 3 puntuaciones sobre los últimos 3 estados. La precisión del sistema mejoró notablemente, incrementando la precisión sobre todo en las decisiones del segundo estado.

Clasificador	% Error	% datos clasificados
Final	12.6%	100%
Final +Fusión	4.9%	100%

Tabla 13: Resultados finales de la prueba 4

A la vista de estos resultados podemos asegurar varias cuestiones. Con un primer clasificador barato y de precisión reducida, el conjunto de clasificadores lineales clasifica más de la mitad de los datos, liberando a los siguientes estados a un coste muy bajo. Por el contrario, tan solo un 10% de los datos alcanzan el clasificador más costoso, por lo que tan solo un número reducido de casos alcanzan el caso peor en cuanto rendimiento.

Adicionalmente cabe destacar que en esta prueba en concreto, el sistema al que el conjunto de clasificadores le es de aplicación, podría no calcular los datos necesarios para los estados 2 y 3 hasta que no fueran estrictamente necesarios, mejorando el rendimiento del análisis sobre los conjuntos de clasificadores en paralelo, teniendo en cuenta el coste necesario para analizar señales de alta resolución como las que se tratan en este tipo de sensores.

6. Conclusiones y trabajo futuro

Que los conjuntos de clasificadores en serie despertaran repentinamente el interés de la comunidad científica a día de hoy no sorprende a nadie. Muchas publicaciones han demostrado que pueden ser verdaderamente útiles en una gran amplitud de campos, aunque por el momento el más común sigue siendo la biométrica. Una buena razón para su avance sosegado es la complejidad que esconden detrás de las múltiples variantes de diseño y las infinitas combinaciones que forman con el resto de parámetros que regulan los conjuntos, convirtiéndolos en un desafío en cuanto a diseño y a su vez en una herramienta increíblemente versátil, capaz de adecuarse a un conjunto de problemas realmente amplio. Como cualquier otra tecnología también tiene sus limitaciones, algunas de las cuales se han visto en este mismo trabajo, pero como con cualquier otra tecnología, es importante conocerlas para saber cuándo y dónde puede aplicarse de forma óptima.

En estos momentos existen muchos frentes abiertos aun en el diseño de clasificadores multi-estado, y más aún si hablamos de estructuras en serie. Por ahora, y en concreto los sistemas con datos individuales e incrementales han mostrado grandes ventajas en ciertos escenarios, pero queda aún bastante trabajo por hacer sobre el impacto que causa un método de fusión en la salida del conjunto, o cuando y como debe utilizarse y de qué tipo para que sea eficiente. Por otra parte, el entrenamiento de los clasificadores en serie es posible que pueda mejorar el rendimiento de estos conjuntos, mejorando el procedimiento habitual e intentando entrenar cada uno de los clasificadores individuales en función del objetivo común y no de un objetivo individual.

7. Bibliografía

1. L.K. HANSEN P. SALAMON "NEURAL NETWORK ENSEMBLES" 1990
2. W.S. McCULLOCH W.PITTS "THE FIRST MATHEMATICAL MODEL OF A NEURON" 1943
3. M.MINSKY S.PAPERT "PERCEPTRONS: AN INTRODUCTION TO COMPUTATIONAL GEOMETRY" 1969
4. Z. AKHTAR, G. FUMERA, G.L MARCIALIS, F. ROLI "EVALUATION OF SERIAL AND PARALLEL MULTIBIOMETRIC SYSTEMS UNDER SPOOFING ATTACKS"
5. T. MURAKAMI, K. TAKAHASHI K. MATSUURA "TOWARDS OPTIMAL COUNTERMEASURES AGAINST WOLVES AND LAMBS IN BIOMETRICS"
6. TED E. SENATOR DARPA "MULTI-STAGE CLASSIFICATION" 2005
7. K. TRAPEZNIKOV V. SALIGRAMA D. CASTAÑON "MULTI-STAGE CLASSIFIER DESIGN" 2012
8. G.L. MARCIALIS F.ROLI L.DIDACI "PATTERN RECOGNITION"
9. E.R.MELZ R.M. MACGREGOR "DESIGN, IMPLEMENTATION AND ANALYSIS OF A PARALLEL DESCRIPTION CLASSIFIER"
10. K.IANAKIEV V.GOVINDARAJU "ARCHITECTURE FOR CLASSIFIER COMBINATION USING ENTROPY MEASURES"
11. A. UHL P.WILD "PARALLEL VERSUS SERIAL CLASSIFIER COMBINATION FOR MULTIBIOMETRIC HAND-BASED IDENTIFICATION"
12. S. MADHVANATH V. GOVINDARAJU "SERIAL CLASSIFIER COMBINATION FOR HANDWRITTEN WORD RECOGNITION" 1995
13. J. SAN MIGUEL-AYANZ G.S. BIGING "COMPARISON OF SINGLE-STAGE AND MULTI-STAGE CLASSIFICATION APPROEACHES FOR COVER TYPE MAPPING WITH TM AND SPOT DATA 1997"
14. C. SANSONE M.VENTO "SIGNATURE VERIFICATION: INCREASING PERFORMANCE BY A MULTI-STAGE SYSTEM" 2000
15. C. RATHGREB A.UHL P.WILD "INCREMENTAL IRIS RECOGNITION: A SINGLE-ALGORITHM SERIAL FUSION STRATEGY TO OPTIMIZE TIME COMPLEXITY"
16. B.ZHANG "RELIABLE CLASSIFICATION OF VEHICLE TYPES BASED ON CASCADE CLASSIFIER ENSEMBLES"
17. M.J.GANGEH R.P. DUIN HAAR ROMENY M.S.KAMEL "A TWO-STAGE COMBINED CLASSIFIER IN SCALE SPACE TEXTURE CLASSIFICATION"
18. G.L. MARCIALIS F.ROLI P.MASTINU "SERIAL FUSIÓN OF MULTI-MODAL BIOMETRIC SYSTEMS"
19. F.DITTRICH S.PULS H.WOERN "MULTI-STAGE CLASSIFIER DESIGN"
20. D.OPITZ R.MACLIN "POPULAR ENSEMBLE METHODS: AN EMPIRICAL STUDY"
21. W.S.SARLE NN FAQ 1997-2002
22. G.L.MARCIALIS F.ROLI L.DIDACI "PERSONAL IDENTITY VERIFICATION BY SERIAL FUSION OF FINGUERPRINT AND FACE MATCHERS"
23. K.TAKASHI M.MIMURA Y.ISOBE Y.SETO "A SECURE AND USER-FRIENDLY MULTI-MODAL BIOMETRIC SYSTEM"

