Universidad Autónoma de Madrid

TESIS DOCTORAL

Radiografía del Análisis Discriminante no Lineal

AUTORA

Ana María González Marcos

DIRECTOR DE TESIS
José Ramón Dorronsoro Ibero

PROGRAMA DE DOCTORADO
Ingeniería Informática
Departamento de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid

Enero 2004

Memoria presentada para optar al título de Doctor en Ingeniería Informática

Dedicada a

- mis padres: Agustín y Adela
- mis abuelos (para mí, mis segundos padres): Angel y Ascensión
- mis hermanas: María Jesús y Susana.

Agradecimientos

Aunque es lo primero que aparece en una tesis, es en realidad, o al menos en mi caso, de lo último que escribes. Antes no puedes hacerlo porque no tienes todos los datos. La pregunta es: ¿A quién o quienes tengo que agradecer?. Más bien, yo diría: ¿Quienes han estado ahí durante todo este tiempo que han hecho que nunca les olvide?, pues ahí va la respuesta, no hay más demora.

Los más importantes, como no a mis padres. Sin ellos habría sido imposible, cuantas veces escucharon: "esto no tiene sentido y lo dejo". Pero al final, todo se arregla y la cosa va para adelante, tomando forma poco a poco. Alguien me dijo: "La tesis es como el punto, cada día haces un poco y al final te encuentras con que has terminado el jersey" y qué razón tenía.

Los siguientes van por orden cronológico a lo largo de mi trayectoria por estos confines. Comienzo por aquel *Despacho 15*, sin omitir a nadie. En él se presentaban todas las etapas de un doctorado, desde los que preparaban la tesina, éste era el caso de María, hasta los que estaban acabando y pensando en irse a un postdoc, como Eduardo. Incluso estaban presentes aquellos que buscaban un hueco en el mundo exterior más que en la propia Universidad, como ocurría con Mariano. Ahí quizás era dónde realmente sentías que tú formabas parte de esa cadena que era el doctorado. Sin duda fue una de las mejores etapas, pero todo se acaba en esta vida.

Cuando se terminó el *Despacho 15*, comenzó la peor etapa, un gran socavón del que hubo que remontar. Exceptuando Alejandro y Paco, no hay mucho más de donde tirar.

Actualmente, no puedo hablar de un *Despacho 15* pero sí de los K*–vecinos*, donde K es un número no demasiado grande. Estos K*–vecinos* han ido evolucionando de un modo semejante a la formación de un cluster según el método K-means, con pequeños pasitos se concluye en algo sólido. Gracias a todos y creo que no necesito dar nombres: "A buen entendedor con pocas palabras basta".

Mención especial me suponen los que se solapan en alguna de las tres etapas y ellos son: Paco, Alejandro y Eduardo.

Por último, me queda la parte más importante, el trabajo de verdad: José (alias Dorron). No sé muy bien como explicarlo, pero quién de los dos, me refiero a ti o a mí, ha sido más cabezota. Yo no hubiera dado un duro (ya a estas alturas serían 5 céntimos de Euro, como mínimo) porque esto fuera a salir adelante. Pero al final, aquí está. Gracias por tu tesón, José.

Índice general

Índice de figuras

Índice de cuadros

Planificación

0.1. Radiografía del Análisis Discriminante no Lineal

Comenzaremos por explicar el título de la tesis: *"Radiografía del Análisis Discriminante no Lineal"*. Es posible que a muchos les extrañe este título para un trabajo desarrollado dentro del programa de doctorado de ingeniería informática; más bien podría parecer un título para un doctorado en medicina, pero en ingeniería informática... ¿qué sentido tiene?, la respuesta es bien sencilla, qué mejor que un título humilde, sin palabras pretenciosas donde cada vez que lo lea me venga a la memoria el contenido de la tesis que aquí presento. Para mí este título lo cumple por las razones que expondré a continuación.

¿Por qué incorpora la palabra *radiografía*? La radiografía no es más que una película sensible a los fotones de los rayos X. Para hacer una radiografía con fines médicos se sitúa al paciente entre la fuente emisora de rayos X y la película con el objetivo de impresionar ésta y posteriormente revelarla en un laboratorio fotográfico. Cuando observamos una radiografía de cualquier zona del cuerpo, lo que vemos en primera instancia es nuestro esqueleto que son las estructuras densas, aquellas que bloquean la mayoría de los fotones y aparecen de color blanco al revelar la película; en segundo lugar podemos ver sombras grises procedentes de los músculos, la grasa y los líquidos, todos ellos dejan pasar parcialmente los fotones e impresionan también parcialmente la película; por último las estructuras que contienen aire se verán negras, ya que dejan pasar a la mayoría de los fotones, impresionando así casi completamente la película. Si entre la placa o película y la fuente de rayos X hubiera algún metal, como éste bloquearía casi todos los fotones de su sección, en la película revelada aparecería con un blanco brillante.

Así pues, la radiografía de un paciente nos muestra el interior del paciente a distintos grados de apreciación; aquí está el símil con esta tesis: vamos a intentar obtener la mejor percepción posible del funcionamiento de la red discriminante no lineal ADnL, evaluándolo a lo largo de esta memoria, de tal forma que al final de la misma podamos decir que hemos conseguido fotografiar el interior de ADnL.

0.2. Organización de la memoria

Los inicios de ADnL parten del trabajo desarrollado con anterioridad por uno de los grupos de investigación del Instituto de Ingeniería del Conocimiento (IIC). Carlos Santa Cruz y José Dorronsoro en [Santa Cruz y Dorronsoro, 1998] presentan el armazón del primer prototipo de ADnL. En esta primera publicación sobre ADnL se presentaba la arquitectura básica de la red y se desarrollaba una primera versión del algoritmo de aprendizaje con un tratamiento matemático excesivamente complejo; posteriormente y a lo largo de diversas publicaciones este tratamiento ha sido revisado varias veces hasta llegar al actual, el que se presenta en esta memoria, cuya característica principal es ser lo más compacto posible, buscando a su vez que sea comprensible y fácil de seguir. En algunos capítulos, por exigencia del guión, se han realizado desarrollados matemáticos casi hasta el último detalle, perdiéndose en estos casos la compactación, pero sin embargo creemos que la ganancia ha sido positiva.

La red ADnL es una red sencilla con la funcionalidad de un discriminante de clases. ADnL es similar a un PMC con la diferencia de que a la salida de la última capa oculta se aplica un discriminante lineal de Fisher, lo que obliga a que la capa de salida tenga $(C-1)$ unidades, donde C es el número de clases existente en la muestra. En esta introducción de la red ADnL, vemos que está compuesta por dos módulos, muy dispares entre sí, acoplados para formar la red ADnL definitiva. Los capítulos 1 y 2 se centran en el estudio de estos dos componentes por separado: el análisis discriminante lineal de Fisher y el perceptron multicapa, respectivamente y en el capítulo siguiente a estos, capítulo 3, es donde se efectúa el engranaje de los constituyentes y se detalla qué es ADnL y cuáles son sus peculiaridades.

En un segundo plano, una vez definida nuestra red ADnL, buscamos métodos para acelerar su convergencia. Nos basamos en métodos de segundo orden por ser éstos más efectivos y en concreto prestamos especial atención al método del gradiente natural desarrollado por Amari [Amari, 1998], donde la búsqueda de los parámetros libres de la red no se realiza dentro del espacio euclídeo, que es el espacio de los métodos convencionales de minimización de funciones, sino que pasamos al espacio de Riemann, donde se tiene en cuenta la curvatura del espacio de parámetros. La medida de la curvatura se realiza a través del conocimiento de la matriz de información de Fisher y por tanto es preciso definir cuál es la matriz de información para la red ADnL. Todo esto se explica al detalle en el capítulo 4.

En última instancia, en el capítulo 5 investigamos la forma de obtener una red ADnL óptima; con ello queremos decir que buscamos una red mínima en el sentido del número de parámetros libres, pero con la capacidad de poder generalizar para conjuntos de muestras no vistos con anterioridad. Dentro de este marco, hemos adaptado diversas técnicas, desde las más sencillas como las implementadas por los métodos OBD [Le Cun et al., 1990] y OBS [Hassibi y Stork, 1993] para el PMC a técnicas estadísticas más complejas en las que los pesos de la red son tratados distintamente en función de que sean pesos de capas ocultas anteriores a la última

oculta o bien que pertenezcan a ésta última capa oculta, la correspondiente al discriminante lineal de Fisher. De este modo, en capas anteriores a la última capa oculta se trabaja con desarrollos estadísticos, donde la matriz de información de Fisher junto con el test de Wald nos indica la relevancia de los pesos en las unidades pertenecientes a dichas capas. En cuanto a las unidades de la última capa oculta, es posible aplicar procesamiento de atributos de entrada para un discriminante de Fisher. Con ayuda del test de Wilks generalizado para C clases es posible estimar cuál es el número de unidades que se necesitan para la última capa oculta.

Por último, en los dos capítulos siguientes se exponen los resultados experimentales que avalan la veracidad de todos los desarrollos teóricos expuestos con anterioridad.

0.3. Publicaciones

A parte de los resultados experimentales expuestos en esta memoria, existe una serie de publicaciones de carácter tanto nacional como internacional, que de alguna forma muestran cuál ha sido el desarrollo cronológico de la tesis y que garantizan resultados anteriores a la escritura de esta memoria. A continuación enumeramos las publicaciones que han surgido durante el proceso de elaboración de esta tesis:

1. Dorronsoro J.R., González A.M., Serrano E., "Linear Unit Relevance in Multiclass NLDA Networks", Proceedings of the 7th International Work-Conference on Artificial an Natural Neural Networks IWANN 2003, LNCS 2686 , Springer, pp 174-181.

2. Dorronsoro J.R., González A.M., "Natural Gradient and Multiclass NLDA Networks", Proceedings of the International Conference on Artificial Neural Networks, ICANN 2002, LNCS 2415, Springer, pp 673-678.

3. Dorronsoro J.R., González A.M., Santa Cruz C., "Extreme Sample Classification and Credit Card Fraud Detection", E-Commerce and Intelligent Methods, Studies in Fuzziness and SoftComputing 105, Springer–Verlag 2002, pp 136–155.

4. Dorronsoro J.R., González A.M., Santa Cruz C., "Natural Gradient Learning in NLDA Networks", Proceedings of the 6th International Work-Conference on Artificial an Natural Neural Networks IWANN 2001, LNCS 2084 , Springer–Velag, pp 427-434.

5. Dorronsoro J.R., González A.M., Santa Cruz C., "Architecture Selection in NLDA Networks", Proceedings of the International Conference on Artificial Neural Networks, ICANN 2001, LNCS 2130, Springer–Verlag, pp 27-32.

6. Dorronsoro J.R., González A. M., Santa Cruz C., " Weight Saliency in NLDA Networks", Proceedings of Learning 2000, Madrid.

7. Dorronsoro J.R., González A.M., Santa Cruz C., "Multilayer Perceptrons, Non–Linear Discriminant Analysis and Extreme Sample Classification", Recent Advances in Pattern Recognition 3, Transworld Research 2002, pp 401-421.

Si bien, éstas son las publicaciones ya realizadas, pero pensando en un futuro muy inmediato creemos que tenemos material más que suficiente como para publicar al menos un par de artículos relacionados con esta memoria.

Parte I

Planteamiento Teórico

Capítulo 1

Análisis Discriminante Lineal

1.1. Introducción

Dentro del marco de *reconocimiento de patrones*, existen dos problemas que están íntimamente ligados: la reducción de la dimensionalidad del conjunto muestral y la extracción de las características de los patrones.

Es bien sabido que al aumentar la dimensionalidad del espacio de características de una muestra, el cálculo computacional requerido para resolver el problema que llevemos entre manos, se incrementa considerablemente, pudiendo llegar a ser completamente impracticable.

Otro factor a añadir es que a medida que aumenta la dimensión del espacio muestral, el número de patrones que se requieren para tener una buena representación de la muestra aumenta de forma exponencial con la dimensión. Con frecuencia, ocurre que el número de patrones o medidas que están disponibles es muy limitado siendo en la mayoría de las veces muy inferior al número deseado.

Así pues, varias técnicas han sido desarrolladas con el propósito de reducir la dimensionalidad de las características de la muestra y a su vez incentivar la esperanza de disminuir la complejidad del problema inicial. Esta primera fase es lo que se conoce como *preprocesamiento* de los datos muestrales.

Uno de los métodos más conocidos para reducir la dimensionalidad del espacio de características, es el discriminante lineal de Fisher [Fisher, 1936] donde lo que se busca es alcanzar una reducción lineal óptima de la dimensionalidad. En sí mismo, no se le puede considerar como un discriminante como tal, pero puede ser aplicado de tal forma que el resultado final tenga el efecto de un discriminante.

La forma más simple de reducir la dimensionalidad de una muestra es proyectar los patrones de dimensión d en un nuevo subespacio lineal de dimensión $\tilde{d}$ que pasa por el origen del espacio $\Re^d$; lógicamente se cumple que $\tilde{d}$ es menor que la dimensión original d.

Aquí cabría imaginar que al pasar de un espacio de dimensión d a uno de dimensión $\tilde{d}$ menor, se podría crear tal confusión que los patrones que inicialmente estuvieran bien separados en el espacio original pasarían a ser una mezcla homogénea en el nuevo espacio, perdiéndose gran parte de información que contiene la muestra original.

El enfoque en cualquier estudio de extracción de características reside en minimizar la pérdida de separabilidad, aún sabiendo que siempre que se disminuye el número de características del espacio muestral original habrá pérdida de información. Expresándolo en forma matemática, se tiene que la probabilidad de error medio en el espacio original, $\Re^d$, nunca es superior a dicha probabilidad en el nuevo espacio reducido, $\Re^d$

$$PEM_d \leq PEM_{\tilde{d}}.$$

Revisaremos brevemente qué se entiende por probabilidad de error medio dentro del contexto de separación de clases; remitimos al libro [Duda et al., 2001] para una revisión más exhaustiva. Partimos de una muestra compuesta por patrones pertenecientes a múltiples clases; cada clase la denotaremos por Ω_c $(c = 1, \cdots, C)$, donde C indica el número de clases distintas en la muestra; las probabilidades *a priori* de cada clase las denotaremos como $\pi_c \equiv P(\Omega_c)$ y sus probabilidades condicionales por $P(\mathbf{X}|\Omega_c)$. La probabilidad total será la suma de las probabilidades conjuntas $P(\mathbf{X}, \Omega_c)$

$$P(\mathbf{X}) = \sum_{c=1}^{C} P(\mathbf{X}, \Omega_c) = \sum_{c=1}^{C} \pi_c P(\mathbf{X}|\Omega_c).$$

La probabilidad *a posteriori* de que un patrón pertenezca a la clases Ω_k vendrá dada por

$$P(\Omega_k|\mathbf{X}) = \frac{\pi_k P(\mathbf{X}|\Omega_k)}{\sum_{c=1}^{C} \pi_c P(\mathbf{X}|\Omega_c)} = \frac{\pi_k P(\mathbf{X}|\Omega_k)}{P(\mathbf{X})}.$$

Todo clasificador está representado por una regla de decisión $\delta(\mathbf{X})$ que asigna a cada patrón la clase a la que pertenece. Si definimos por E_c el conjunto de patrones que el clasificador ha asignado como pertenecientes a la clase Ω_c, $\{\mathbf{X} : \delta(\mathbf{X}) = \Omega_c\}$, la probabilidad de que el clasificador se equivoque en su asignación es lo que denominamos probabilidad de error medio y viene dada por la expresión

$$PEM(\delta) = \sum_{c=1}^{C} \sum_{\substack{k=1 \\ k \neq c}}^{C} \int_{E_k} P(\Omega_c|\mathbf{X})\, d\mathbf{X}.$$

Como iremos viendo a lo largo de este capítulo, lo interesante del método de Fisher radica en que mediante una rotación del subespacio lineal de proyección en torno al origen es posible encontrar una orientación óptima, de tal forma que disminuyendo la complejidad del problema se minimice la pérdida de separabilidad que contiene el conjunto inicial de patrones.

1.2. Análisis Discriminante Lineal en dos clases

Para explicar cómo funciona el discriminante lineal de Fisher, vamos a introducirlo inicialmente del mismo modo que Fisher [Fisher, 1936] lo hizo. Para ello, vamos a plantear la resolución de un problema de clasificación de patrones para dos clases ($C = 2$).

En este caso, el subespacio óptimo será una línea recta que pasa por el origen y que al proyectar el conjunto de patrones de dimensión d en dicha línea, las proyecciones de las dos clases están lo más separadas posible, o lo que es lo mismo, fijándose en las proyecciones de los patrones se puede distinguir con confianza qué patrón pertenece a qué clase. Es decir, vamos a enfocar el estudio desde el punto de vista de un discriminante. Si lo viéramos como reducción de la dimensionalidad, estaríamos pasando de un espacio d–dimensional a un espacio unidimensional ($(C-1)$ dimensional).

Comenzaremos denotando por $\mathcal{H}$ un conjunto con patrones de dimensión d que pertenecen a dos clases, lo que implica que el conjunto $\mathcal{H}$ está formado por dos subconjuntos $\mathcal{H} = \{\mathcal{H}_1, \mathcal{H}_2\}$, siendo $\mathcal{H}_1$ y $\mathcal{H}_2$ los subconjuntos de los patrones de la clase C_1 y C_2, respectivamente. La proyección de un patrón $\mathbf{x}$ en una línea recta cuya dirección es $\mathbf{w}$ puede ser expresada matricialmente de la forma

$$y = \mathbf{w}^T\mathbf{x}. \tag{1.1}$$

En el nuevo espacio unidimensional, se define un conjunto $\mathcal{L}$ con las proyecciones de los patrones pertenecientes al conjunto $\mathcal{H}$ e igualmente existen dos subconjuntos $\mathcal{L}_1$ y $\mathcal{L}_2$, ($\mathcal{L} = \{\mathcal{L}_1, \mathcal{L}_2\}$) formados por las proyecciones de sus antecesores $\mathcal{H}_1$ y $\mathcal{H}_2$.

Ahora, el problema es elegir la dirección de $\mathbf{w}$ tal que para un patrón $\mathbf{x}_i$ procedente bien del subconjunto $\mathcal{H}_1$ o del $\mathcal{H}_2$, su proyección a lo largo de la línea caiga exactamente en el grupo que le pertenece, $\mathcal{L}_1$ o $\mathcal{L}_2$. Idealmente, si $y_i \in \mathcal{L}_1$, entonces $\mathbf{x}_i \in \mathcal{H}_1$ y si por el contrario $y_i \in \mathcal{L}_2$, entonces $\mathbf{x}_i \in \mathcal{H}_2$.

Lo más natural para identificar las dos clases existentes en el conjunto $\mathcal{H}$ es que las correspondientes proyecciones $\mathcal{L}_1$ y $\mathcal{L}_2$ estén suficientemente separadas o lo que lo mismo, se busca que los subconjuntos $\mathcal{L}_1$ y $\mathcal{L}_2$ queden lo más disjuntos posible.

Una medida de la separación de los dos subconjuntos $\mathcal{L}_1$ y $\mathcal{L}_2$ es la diferencia de sus medias. Para cada clase C_k, su media proyectada en el subconjunto $\mathcal{L}_k$ ($k = 1, 2$) es un escalar dado por

$$\tilde{m}_k = E[y|y \in C_k] = E[\mathbf{w}^T\mathbf{x}|\mathbf{x} \in \mathcal{H}_k] = \mathbf{w}^T\mathbf{m}_k$$

donde $\mathbf{m}_k$ es la media de los patrones de la muestra que pertenecen a la clase C_k, o lo que es lo mismo pertenecen al subconjunto $\mathcal{H}_k$

$$\mathbf{m}_k = E[\mathbf{x}|\mathbf{x} \in \mathcal{H}_k].$$

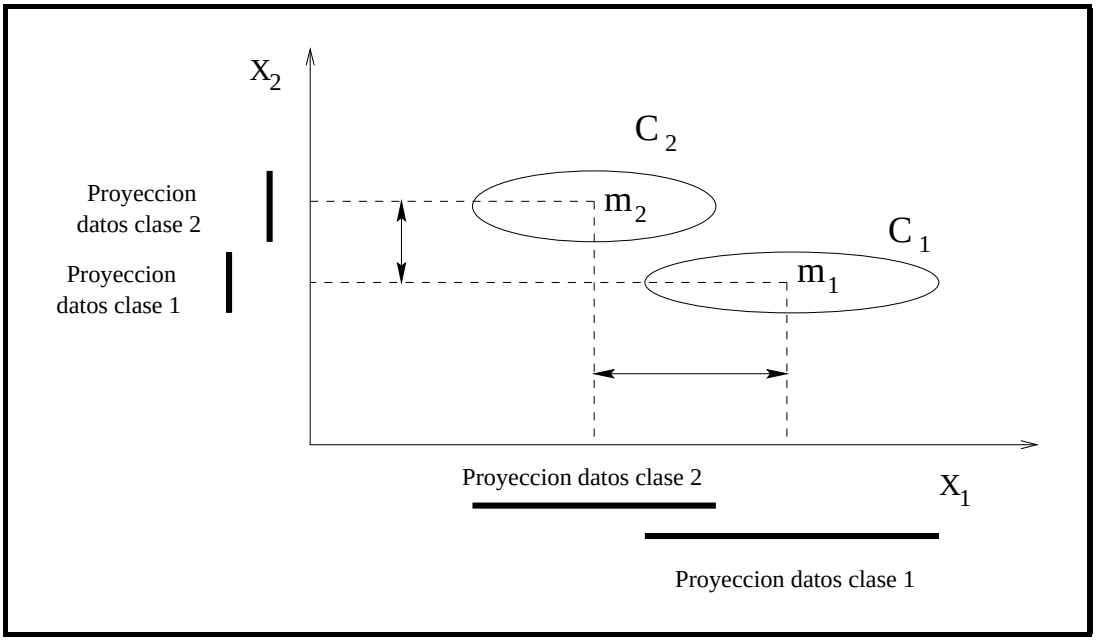


Figura 1.1: Importancia de la matriz de covarianza intra–clases en la separación de los datos proyectados

Con la notación $E[\,\cdot\,]$ indicamos el valor esperado y $E[\,\cdot\,|\,\cdot\,]$ la esperanza condicional. Con el objetivo de seguir fijando notaciones, cuando nos refiramos al espacio de la proyección añadiremos una tilde ($\tilde{\cdot}$) al argumento que estemos usando. A modo de ejemplo, $\tilde{m}$ indica la media total en el conjunto de las proyecciones $\mathcal{L}$ y $\mathbf{m}$ la media total en el conjunto de patrones $\mathcal{H}$.

La diferencia de las medias proyectadas en función de las medias de la muestra viene dada por la expresión

$$\tilde{m}_1 - \tilde{m}_2 = \mathbf{w}^T(\mathbf{m}_1 - \mathbf{m}_2),$$

donde puede verse que esta diferencia se puede hacer todo lo grande que se quiera con tan sólo escalar el vector $\mathbf{w}$. Por ello, considerando sólo la diferencia de las medias proyectadas no es posible obtener un buen clasificador.

En la figura (1.1) se muestran dos clases que están bien separadas en el espacio original 2D. Al proyectar las medias en los dos ejes principales, se aprecia que la distancia de las medias proyectadas en el eje x_1 es mayor que en el eje x_2; sin embargo, la separación de los datos proyectados en el eje x_2 es mucho mejor que en el eje x_1, donde es fácil ver que existe solapamiento. Luego, parece evidente que algún otro factor adicional interviene en la búsqueda del eje óptimo de proyección.

Observando la figura (1.1), se podría pensar que la forma de distribuirse los datos de la muestra tiene mucho que decir. Parece justificable que para obtener un buen clasificador, deberíamos añadir a la separación de las medias proyectadas alguna medida de la dispersión de los datos proyectados (subconjuntos $\mathcal{L}_1$

y $\mathcal{L}_2$). Precisamente, son las matrices de covarianza las que miden cómo están distribuidos los datos muestrales.

La dispersión en la proyección de los patrones de la clase C_k viene dada por la expresión

$$\tilde{s}_k^2 = E\left[(y - \tilde{m}_k)^2 \,|y \in C_k\right]. \tag{1.2}$$

La cantidad $\pi_1\tilde{s}_1^2 + \pi_2\tilde{s}_2^2$ se denomina *dispersión total intra–clases* y es una medida de lo agrupadas que están las clases. π_1 y π_2 indican la probabilidad a priori de pertenecer bien a la clase C_1 o bien a la clase C_2.

Las dos condiciones que debe cumplir un buen clasificador son que las medias de las proyecciones, $\tilde{m}_i$ estén bien separadas y a la vez los subconjuntos $\mathcal{L}_k$ estén lo menos entremezclados posible. Entonces, para obtener una buena separación de los patrones proyectados a lo largo de la dirección de $\mathbf{w}$, deberíamos fijarnos en que la distancia entre las medias proyectadas sea grande relativa a la medida de la dispersión intra–clases en la proyección.

Fisher sugirió como función criterio

$$J(\mathbf{w}) = \frac{(\tilde{m}_1 - \tilde{m}_2)^2}{\pi_1\tilde{s}_1^2 + \pi_2\tilde{s}_2^2}, \tag{1.3}$$

que cumple las condiciones anteriores: J aumenta si la separación de las medias en la proyección aumenta o bien si la dispersión intra–clases en la proyección disminuye. Con lo cual, maximizar el criterio J supondría obtener el mejor vector $\mathbf{w}$ que define la línea de proyección óptima.

Para determinar la dirección del vector $\mathbf{w}$ óptimo es necesario reescribir J en función explícita de $\mathbf{w}$. El vector $\mathbf{w}$ influye tanto en el numerador como en el denominador de la ecuación (1.3), ya que tanto $\tilde{m}_k$ como $\tilde{s}_k$ pueden ser expresadas en función del discriminante $y = \mathbf{w}^T\mathbf{x}$. Si definimos en la muestra original las matrices de dispersión de las clases $\mathbf{S}_i$ e intra–clases $\mathbf{S}_W$ como

$$\begin{aligned} \mathbf{S}_k &= E\left[(\mathbf{x} - \mathbf{m}_k)(\mathbf{x} - \mathbf{m}_k)^T \,|\mathbf{x} \in \mathcal{H}_k\right], \\ \mathbf{S}_W &= \pi_1\mathbf{S}_1 + \pi_2\mathbf{S}_2, \end{aligned}$$

escribiendo $\tilde{s}_k^2$ (1.2) en función de $\mathbf{w}$ se tiene

$$\begin{aligned} \tilde{s}_k^2 &= E\left[\left(\mathbf{w}^T\mathbf{x} - \mathbf{w}^T\mathbf{m}_k\right)^2 |\mathbf{x} \in \mathcal{H}_k\right] \\ &= E\left[\mathbf{w}^T(\mathbf{x} - \mathbf{m}_k)(\mathbf{x} - \mathbf{m}_k)^T\mathbf{w} |\mathbf{x} \in \mathcal{H}_k\right] = \mathbf{w}^T\mathbf{S}_k\mathbf{w}. \end{aligned}$$

La suma de las dispersiones de las dos clases en función de $\mathbf{w}$ quedaría de la forma

$$\pi_1\tilde{s}_1^2 + \pi_2\tilde{s}_2^2 = \pi_1\mathbf{w}^T\mathbf{S}_1\mathbf{w} + \pi_2\mathbf{w}^T\mathbf{S}_2\mathbf{w} = \mathbf{w}^T(\pi_1\mathbf{S}_1 + \pi_2\mathbf{S}_2)\mathbf{w} = \mathbf{w}^T\mathbf{S}_W\mathbf{w}.$$

Del mismo modo, la separación de las medias proyectadas en función de $\mathbf{w}$ viene dada por

$$(\tilde{m}_1 - \tilde{m}_2)^2 = \left(\mathbf{w}^T\mathbf{m}_1 - \mathbf{w}^T\mathbf{m}_2\right)^2 = \mathbf{w}^T(\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T\mathbf{w} = \mathbf{w}^T\mathbf{S}_B\mathbf{w},$$

donde $\mathbf{S}_B = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T$ es una matriz equivalente a la *matriz de dispersión inter–clases* en el espacio original. En posteriores secciones veremos que este valor de $\mathbf{S}_B$ difiere de la matriz real de *dispersión inter–clases* en una constante, pero debido a que no afecta al desarrollo que nos ocupa actualmente, vamos a obviarlo.

Tanto la *matriz de dispersión intra–clases*, $\mathbf{S}_W$, como la equivalente *matriz de dispersión inter–clases*, $\mathbf{S}_B$, son matrices simétricas y semidefinidas positivas. Además, $\mathbf{S}_W$ es en general una matriz no singular; sin embargo, no ocurre lo mismo con $\mathbf{S}_B$: como $\mathbf{S}_B$ procede de un producto externo de un vector consigo mismo, su rango es por tanto la unidad.

La función de criterio J definida en (1.3) expresada en términos de $\mathbf{S}_B$ y $\mathbf{S}_W$, viene dada por

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}}, \tag{1.4}$$

donde los subconjuntos $\mathcal{H}_1$ y $\mathcal{H}_2$ son los que determinan $\mathbf{S}_B$ y $\mathbf{S}_W$, o lo que es lo mismo será el espacio original el que determine las dos matrices de dispersión.

El vector $\mathbf{w}$ que maximiza la ecuación (1.4) se obtiene cuando $\partial J / \partial \mathbf{w} = 0$. Así pues, la derivada de (1.4) respecto $\mathbf{w}$ viene dada por

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{w}} &= \frac{2}{\left(\mathbf{w}^T \mathbf{S}_W \mathbf{w}\right)^2} \left[\mathbf{S}_B \mathbf{w} \left(\mathbf{w}^T \mathbf{S}_W \mathbf{w}\right) - \mathbf{S}_W \mathbf{w} \left(\mathbf{w}^T \mathbf{S}_B \mathbf{w}\right)\right] \\ &= \frac{2}{\left(\mathbf{w}^T \mathbf{S}_W \mathbf{w}\right)} \left(\mathbf{S}_B \mathbf{w} - J\, \mathbf{S}_W \mathbf{w}\right), \end{aligned}$$

y al tomar $\partial J / \partial \mathbf{w} = 0$, se obtiene

$$\mathbf{S}_B \mathbf{w} - J\, \mathbf{S}_W \mathbf{w} = 0,$$

o lo que es lo mismo

$$\mathbf{S}_B \mathbf{w} = J\, \mathbf{S}_W \mathbf{w}.$$

Si $\mathbf{S}_W^{-1}$ existe, multiplicando a la izquierda en los dos términos de la igualdad por $\mathbf{S}_W^{-1}$ se obtiene la expresión

$$\mathbf{S}_W^{-1} \mathbf{S}_B \mathbf{w} = J\, \mathbf{w}.$$

En definitiva, se trata de un *problema generalizado de autovalores*, donde el escalar J de valor máximo corresponde al mayor autovalor de $\mathbf{S}_W^{-1} \mathbf{S}_B$.

Como para definir la línea de proyección nos interesa la dirección de $\mathbf{w}$ y no así su magnitud, entonces es posible tomar cualquier vector reescalado $\mathbf{w}$ que mantenga dicha dirección. Lo más habitual es tomar el vector $\mathbf{w}$ normalizado, es decir $\|\mathbf{w}\| = 1$.

Una solución para $\mathbf{w}$ podríamos encontrarla resolviendo $\left(\mathbf{S}_W^{-1} \mathbf{S}_B\right)$ para los vectores canónicos $\mathbf{e}$. Otra solución alternativa y más sencilla que la anterior,

está basada en que $\mathbf{S}_B\mathbf{w}$ tiene la misma dirección que $\mathbf{m}_1 - \mathbf{m}_2$, pues

$$\mathbf{S}_B\mathbf{w} = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T\mathbf{w} = (\mathbf{m}_1 - \mathbf{m}_2)\kappa$$

donde κ es un escalar. Entonces el vector $\mathbf{w}$ que define la dirección óptima de la línea de proyección viene dado por

$$\mathbf{w} = \mathbf{S}_W^{-1}(\mathbf{m_1} - \mathbf{m_2}). \tag{1.5}$$

Como hemos indicado anteriormente, nos interesa la dirección de $\mathbf{w}$; entonces el valor de κ es indiferente y puede tomar cualquier valor distinto de cero.

Como conclusión, hemos encontrado un vector $\mathbf{w}$ que maximiza el criterio de Fisher (1.4). Si proyectamos los datos muestrales según el discriminante expresado por la ecuación (1.1), el problema de clasificación para dos clases en un espacio inicial d–dimensional se reduce a un problema más sencillo, en el que se busca separar las dos clases explorando en un nuevo conjunto de datos unidimensional, que sin duda alguna es mucho más manejable.

1.2.1. Complejidad en el Análisis de Fisher para dos clases

Hasta este momento, se ha presentado el desarrollo teórico del discriminante de Fisher. Pero, cuando pasamos a la práctica, lo que en realidad se tiene es una muestra definida por un número finito N de patrones. Es decir, tenemos un conjunto $\mathcal{H}$ con N patrones de dimensión d divididos en dos clases, con N_1 patrones para la primera clase C_1 y N_2 para la segunda clase C_2, donde $N_1 + N_2 = N$

$$\mathcal{H} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\} = \{\mathcal{H}_1, \mathcal{H}_2\},$$

$\mathcal{H}_1$ y $\mathcal{H}_2$ son los subconjuntos de los patrones de la clase C_1 y C_2, respectivamente.

Luego, tenemos que definir las matrices $\hat{\mathbf{S}}_B$ y $\hat{\mathbf{S}}_W$, procedentes de la estimación de la muestra. Las expresiones matemáticas para $\hat{\mathbf{S}}_B$ y $\hat{\mathbf{S}}_W$ en el caso de dos clases son:

$$\begin{aligned}
\hat{\mathbf{S}}_W &= \frac{1}{N}\sum_{k=1}^{2} N_k\,\hat{\mathbf{S}}_k, \\
\hat{\mathbf{S}}_B &= (\hat{\mathbf{m}}_\mathbf{1} - \hat{\mathbf{m}}_\mathbf{2})(\hat{\mathbf{m}}_\mathbf{1} - \hat{\mathbf{m}}_\mathbf{2})^T
\end{aligned}$$

donde $\hat{\mathbf{m}}_k$ y $\hat{\mathbf{S}}_k$ $(k = 1, 2)$ son respectivamente la estimación de la media y covarianza de los patrones de la clase k, esto es

$$\begin{aligned}
\hat{\mathbf{m}}_k &= \frac{1}{N_k}\sum_{i=1}^{N_k} \mathbf{x}_i, \\
\hat{\mathbf{S}}_k &= \frac{1}{N_k}\sum_{i=1}^{N_k} (\mathbf{x}_i - \hat{\mathbf{m}}_k)(\mathbf{x}_i - \hat{\mathbf{m}}_k)^T.
\end{aligned}$$

El coste computacional que requiere encontrar el $\hat{\mathbf{w}}$ óptimo para el discriminante lineal de Fisher (1.5) está dominado por el cálculo de $\hat{\mathbf{S}}_W$ y su inversa. Calcular la matriz de dispersión supone un coste de $\mathcal{O}(d^2 N)$, recordando que N es el número total de patrones y d la dimensión del espacio original, y el cálculo de la inversa de un matriz tiene un coste de $\mathcal{O}(d^3)$ [Press et al., 1992]. Como en general, $N \gg d$, será el cálculo de la matriz de dispersión el que domina en el coste computacional del análisis discriminante de Fisher; en definitiva su coste será $\mathcal{O}(d^2 N)$.

1.3. Análisis Discriminante Lineal Múltiple

La generalización del discriminante lineal de Fisher a más de dos clases fue realizada por Rao [Rao, 1948], aproximadamente una década después de la primera publicación sobre el tema.

Al plantear la generalización a cualquier número de clases, hay que tener presente que en cualquier caso, la dimensionalidad del espacio muestral d nunca debe ser menor que el número de clases C ($d \geq C$). En el discriminante lineal múltiple habrá que tener en cuenta $(C-1)$ funciones discriminantes similares a las deducidas para dos clases (expresión 1.1). Por tanto, la proyección se realizará en un espacio $(C-1)$–dimensional.

La matriz de dispersión intra–clases para C clases viene dada por

$$\begin{aligned} \mathbf{S}_W &= \sum_{k=1}^{C} \pi_k \mathbf{S}_k, \\ \mathbf{S}_k &= E[(\mathbf{x}-\mathbf{m}_k)(\mathbf{x}-\mathbf{m}_k)^T | \mathbf{x} \in H_k]. \end{aligned} \tag{1.6}$$

El valor π_k será por tanto, la probabilidad a priori de la clase k y como tal probabilidad, debe cumplir la siguiente relación $\sum_{k=1}^{C} \pi_k = 1$.

Para obtener la expresión de la matriz $\mathbf{S}_B$, primero nos fijaremos en la matriz de covarianza de toda la muestra

$$\mathbf{S}_T = E\left[(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T\right]$$

donde $\mathbf{m}$ es la media de la muestra original

$$\mathbf{m} = E[\mathbf{x}] = \sum_{k=1}^{C} \pi_k \mathbf{m}_k. \tag{1.7}$$

La matriz de covarianza total puede descomponerse en la suma de dos matrices: la matriz de *covarianza intra–clases* y la matriz de *covarianza inter–clases*, es decir

$\mathbf{S}_T = \mathbf{S}_W + \mathbf{S}_B$. La demostración es sencilla:

$$\begin{aligned}
\mathbf{S}_T &= E\left[(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T\right] = \sum_{k=1}^{C} \pi_k E_k\left[(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^T\right] \\
&= \sum_{k=1}^{C} \pi_k E_k\left[(\mathbf{x}-\mathbf{m}_k+\mathbf{m}_k-\mathbf{m})(\mathbf{x}-\mathbf{m}_k+\mathbf{m}_k-\mathbf{m})^T\right] \\
&= \sum_{k=1}^{C} \pi_k E_k\left[(\mathbf{x}-\mathbf{m}_k)(\mathbf{x}-\mathbf{m}_k)^T\right] + \sum_{k=1}^{C} \pi_k E_k\left[(\mathbf{m}_k-\mathbf{m})(\mathbf{m}_k-\mathbf{m})^T\right] \\
&+ 2\sum_{k=1}^{C} \pi_k E_k\left[(\mathbf{m}_k-\mathbf{m})(\mathbf{x}-\mathbf{m}_k)^T\right]. \qquad (1.8)
\end{aligned}$$

El primer término de los tres finales de la expresión de $\mathbf{S}_T$ en (1.8) puede reducirse a $\sum_{k=1}^{C} \pi_k E_k\left[(\mathbf{x}-\mathbf{m}_k)(\mathbf{x}-\mathbf{m}_k)^T\right] = \sum_{k=1}^{C} \pi_k \mathbf{S}_k$; se trata pues, de la *matriz de dispersión intra–clases* $\mathbf{S}_W$, expresión (1.6). Por otro lado, sabiendo que el valor esperado de una constante es el valor que toma dicha constante, el segundo término de la expresión (1.8) se simplifica a $\sum_{k=1}^{C} \pi_k E_k\left[(\mathbf{m}_k-\mathbf{m})(\mathbf{m}_k-\mathbf{m})^T\right] = \sum_{k=1}^{C} \pi_k (\mathbf{m}_k-\mathbf{m})(\mathbf{m}_k-\mathbf{m})^T$, donde tanto los distintos $\mathbf{m}_k$ como $\mathbf{m}$ son constantes para los conjuntos $\mathcal{H}_k$ $(k = 1, \cdots, C)$ y $\mathcal{H}$, respectivamente. Este término es en realidad, la *matriz de dispersión inter–clases* generalizada. Se trata de una medida de lo separadas que están las medias de las distintas clases $\mathbf{m}_k$ respecto a la media global de la muestra $\mathbf{m}$

$$\mathbf{S}_B = \sum_{k=1}^{C} \pi_k (\mathbf{m}_k-\mathbf{m})(\mathbf{m}_k-\mathbf{m})^T. \qquad (1.9)$$

Por último, demostraremos que el tercer sumando de la expresión (1.8) es idéntico a cero, $\sum_{k=1}^{C} \pi_k E_k[(\mathbf{m}_k-\mathbf{m})(\mathbf{x}-\mathbf{m}_k)^T] = 0$. Para la clase genérica k, se cumple que $E_k[(\mathbf{m}_k-\mathbf{m})(\mathbf{x}-\mathbf{m}_k)^T] = (\mathbf{m}_k-\mathbf{m})\, E_k[(\mathbf{x}-\mathbf{m}_k)^T]$, y dado que $E_k[x] = \mathbf{m}_k$, entonces $E_k[(\mathbf{x}-\mathbf{m}_k)^T] = 0$.

Con todo esto, hemos demostrado que la dispersión de la muestra puede expresarse en función de dos dispersiones, la intra–clases, $\mathbf{S}_W$ y la inter–clases, $\mathbf{S}_B$, tal y como vimos en la sección 1.2 para dos clases. A continuación, veremos la relación de la expresión genérica de $\mathbf{S}_B$ (1.9) con la expresión dada en la sección 1.2 para dos clases, recordando que $\mathbf{S}_B = (\mathbf{m}_1-\mathbf{m}_2)(\mathbf{m}_1-\mathbf{m}_2)^T$. La figura (1.2) muestra de una forma esquemática la equivalencia entre las dos expresiones.

Para dos clases, la media global, $\mathbf{m}$, estará situada a lo largo de la línea recta que une los puntos correspondientes a las medias de las dos clases $\mathbf{m}_1$ y $\mathbf{m}_2$

$$\mathbf{m} = \pi_1 \mathbf{m}_1 + \pi_2 \mathbf{m}_2. \qquad (1.10)$$

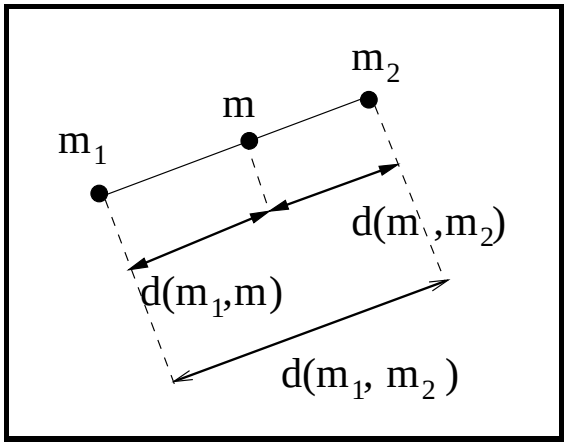


Figura 1.2: Esquema de equivalencias de $\mathbf{S}_B$ para dos clases

Si la probabilidad a priori π_1 es mayor que π_2, la media global estará más próxima a $\mathbf{m}_1$, en el caso contrario estará más cerca de $\mathbf{m}_2$; si $\pi_1 = \pi_2 = 0{,}5$, la media $\mathbf{m}$ estará justo en el punto medio entre $\mathbf{m}_1$ y $\mathbf{m}_2$. En cualquier caso, se cumple que la distancia de $\mathbf{m}_1$ a $\mathbf{m}_2$ será la suma de la distancia de $\mathbf{m}_1$ a $\mathbf{m}$ más la distancia de $\mathbf{m}$ a $\mathbf{m}_2$, o lo que es equivalente

$$(\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T = (\mathbf{m}_1 - \mathbf{m})(\mathbf{m}_1 - \mathbf{m})^T + (\mathbf{m}_2 - \mathbf{m})(\mathbf{m}_2 - \mathbf{m})^T,$$

donde el término de la derecha de la igualdad difiere de la expresión (1.9) en los factores multiplicativos π_1 y π_2; mientras que el término de la izquierda es el valor de $\mathbf{S}_B$ dado en la sección 1.2. Desarrollando la expresión (1.9) para dos clases se obtiene que

$$\mathbf{S}_B = \pi_1 (\mathbf{m}_1 - \mathbf{m})(\mathbf{m}_1 - \mathbf{m})^T + \pi_2 (\mathbf{m}_2 - \mathbf{m})(\mathbf{m}_2 - \mathbf{m})^T.$$

Usando la expresión (1.10), podemos expresar $(\mathbf{m}_1 - \mathbf{m})$ y $(\mathbf{m}_2 - \mathbf{m})$ en función de las medias parciales $\mathbf{m}_1$ y $\mathbf{m}_2$

$$\begin{aligned} (\mathbf{m}_1 - \mathbf{m}) &= \pi_2 (\mathbf{m}_1 - \mathbf{m_2}) \\ (\mathbf{m}_2 - \mathbf{m}) &= \pi_1 (\mathbf{m}_2 - \mathbf{m_1}), \end{aligned}$$

que sustituidas en $\mathbf{S}_B$ se obtiene

$$\begin{aligned} \mathbf{S}_B &= \pi_1\pi_2^2 (\mathbf{m}_1 - \mathbf{m_2})(\mathbf{m}_1 - \mathbf{m_2})^T + \pi_2\pi_1^2 (\mathbf{m}_1 - \mathbf{m_2})(\mathbf{m}_1 - \mathbf{m}_2)^T \\ &= \pi_1(\pi_1 + \pi_2)\pi_2 \ (\mathbf{m}_1 - \mathbf{m_2})(\mathbf{m}_1 - \mathbf{m}_2)^T \\ &= \pi_1\pi_2 \ (\mathbf{m}_1 - \mathbf{m_2})(\mathbf{m}_1 - \mathbf{m}_2)^T. \end{aligned}$$

Se puede apreciar que es el producto de las probabilidades *a priori* $\pi_1\pi_2$ lo que difiere del valor definido anteriormente en la sección 1.2 para la matriz $\mathbf{S}_B$. Sin embargo, este factor no influye en el cálculo del vector de proyección y por este

motivo no se tuvo en cuenta en dicha sección, aliviando así la preocupación por factores insignificantes.

La proyección del espacio d–dimensional al espacio $(C-1)$–dimensional es llevada a cabo por $(C-1)$ funciones discriminantes de la forma

$$y_j = \mathbf{w}_j^T \mathbf{x} \qquad\qquad j = 1, ..., C-1$$

que en modo matricial se simplificaría a $\mathbf{y} = \mathbf{W}^T\mathbf{x}$, donde $\mathbf{W}$ es una matriz de dimensión $d \times (C-1)$. Teniendo esto en cuenta, las matrices de dispersión en el espacio $(C-1)$–dimensional correspondientes a las proyecciones quedarían de la forma

$$\begin{aligned}
\tilde{\mathbf{S}}_B &= \sum_{k=1}^{C} \pi_k \left(\tilde{\mathbf{m}}_k - \tilde{\mathbf{m}}\right)\left(\tilde{\mathbf{m}}_k - \tilde{\mathbf{m}}\right)^T \\
\tilde{\mathbf{S}}_W &= \sum_{k=1}^{C} \pi_k \tilde{\mathbf{S}}_k \\
\tilde{\mathbf{S}}_k &= E[(\mathbf{y} - \tilde{\mathbf{m}}_k)\left(\mathbf{y} - \tilde{\mathbf{m}}_k\right)^T | \mathbf{y} \in C_k]
\end{aligned}$$

donde

$$\tilde{\mathbf{m}}_k = E[\mathbf{y}|\mathbf{y} \in C_k] \qquad \text{y} \qquad \tilde{\mathbf{m}} = E[\mathbf{y}] = \sum_{k=1}^{C} \pi_k \tilde{\mathbf{m}}_k$$

La relación entre estas matrices de covarianza $(C-1)$–dimensionales y las matrices de covarianza del espacio muestral viene dada por

$$\tilde{\mathbf{S}}_W = \mathbf{W}^T \mathbf{S}_W \mathbf{W}, \qquad\qquad \tilde{\mathbf{S}}_B = \mathbf{W}^T \mathbf{S}_B \mathbf{W}.$$

De nuevo, buscamos un escalar que aumente su valor cuando se den las condiciones siguientes: la dispersión inter–clases en el espacio de la proyección sea grande y la dispersión intra–clases en el mismo espacio sea pequeña. Es decir, buscamos una matriz de transformación $\mathbf{W}$ que en algún sentido maximice alguna medida de la dispersión inter–clases, $\tilde{\mathbf{S}}_B$ y a la vez minimice alguna medida de la dispersión intra–clases, $\tilde{\mathbf{S}}_W$.

Hay muchos criterios que cumplen las condiciones impuestas [Fukunaga, 1990]. De forma genérica, la expresión del criterio J podría venir dada por ecuaciones del tipo

$$J(\mathbf{W}) = \phi\left(\tilde{\mathbf{S}}_1\right) / \varphi\left(\tilde{\mathbf{S}}_2\right) \qquad \text{o} \qquad J(\mathbf{W}) = \psi\left(\tilde{\mathbf{S}}_2^{-1}\tilde{\mathbf{S}}_1\right) \tag{1.11}$$

donde $\tilde{\mathbf{S}}_1$, $\tilde{\mathbf{S}}_2$ se refiere a las matrices de covarianza $\tilde{\mathbf{S}}_B$, $\tilde{\mathbf{S}}_W$ o $\tilde{\mathbf{S}}_T$ en el espacio de la proyección. Son muchas las posibles combinaciones para $\{\tilde{\mathbf{S}}_1, \tilde{\mathbf{S}}_2\}$; dos de las más comunes son $\{\tilde{\mathbf{S}}_B, \tilde{\mathbf{S}}_W\}$ y $\{\tilde{\mathbf{S}}_B, \tilde{\mathbf{S}}_T\}$. En realidad, estas dos combinaciones son equivalentes puesto que $\tilde{\mathbf{S}}_T = \tilde{\mathbf{S}}_B + \tilde{\mathbf{S}}_W$. La triada ϕ, φ y ψ se refiere a funciones

u operadores aplicables a matrices; normalmente ϕ y φ son la misma función u operador.

El criterio más popular en el contexto de un discriminante lineal de Fisher es

$$J(\mathbf{W}) = \frac{\left|\tilde{\mathbf{S}}_B\right|}{\left|\tilde{\mathbf{S}}_W\right|} = \frac{\left|\mathbf{W}^T\mathbf{S}_B\mathbf{W}\right|}{\left|\mathbf{W}^T\mathbf{S}_W\mathbf{W}\right|} \tag{1.12}$$

donde $|\cdot|$ indica el determinante de la matriz. El determinante de las matrices de dispersión es una buena medida, dado que es el producto de los autovalores de la matriz de dispersión, o lo que es lo mismo, desde el punto de vista geométrico, es el producto de la longitud de los ejes principales del hiper–elipsoide definido por la matriz de dispersión y cuyas direcciones vienen representadas por los autovectores de dicha matriz. En definitiva, se mide de alguna forma el volumen de la dispersión de los datos en las direcciones de los correspondientes autovectores.

Así pues, encontrar una matriz rectangular $\mathbf{W}$ que maximice el criterio (1.12) se reduce a encontrar los autovectores de mayor autovalor de la expresión

$$\mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{w}_j = \lambda_j\mathbf{w}_j \tag{1.13}$$

la demostración de que el $\mathbf{W}$ óptimo se obtiene a partir de la expresión (1.13) se detalla en el apéndice A, donde además se muestran otros posibles criterios de Fisher, basados en el operador traza de matrices.

De la expresión (1.9) se observa que $\mathbf{S}_B$ está compuesta por la suma de C matrices, cada una de las cuales es un producto externo de dos vectores idénticos y por tanto, individualmente son matrices de rango 1. Fijándonos en la expresión (1.7), se puede apreciar que sólo $(C-1)$ de estas matrices son independientes; entonces el rango de $\mathbf{S}_B$ es a lo sumo $(C-1)$. Como conclusión, el número de autovalores distintos de cero es como máximo $(C-1)$. Esto explica por qué la proyección pasa de un espacio d-dimensional a otro $(C-1)$–dimensional, cumpliéndose siempre que $d \geq (C-1)$. Los vectores $\mathbf{w}_j$ buscados corresponden a los autovectores de autovalor distinto de cero.

1.3.1. Complejidad en el Análisis de Fisher Múltiple

En la sección anterior hemos realizado el estudio teórico del discriminante de Fisher para cualquier número de clases, pero al introducirlo a ejemplos prácticos es necesario definir las matrices de dispersión procedentes de la estimación de una muestra con N patrones y C clases, donde $N = \sum_{k=1}^{C} N_k$ y cada N_k es el número de patrones que pertenecen a la clase C_k. El conjunto muestral vendrá definido por

$$\mathcal{H} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N\} = \{\mathcal{H}_1, \cdots, \mathcal{H}_C\}.$$

Las expresiones para las nuevas matrices $\hat{\mathbf{S}}_B$ y $\hat{\mathbf{S}}_W$ son

$$\begin{aligned}\hat{\mathbf{S}}_W &= \frac{1}{N}\sum_{k=1}^{C} N_k\,\hat{\mathbf{S}}_k,\\ \hat{\mathbf{S}}_B &= \frac{1}{N}\sum_{k=1}^{C} N_k\,(\hat{\mathbf{m}}_k-\hat{\mathbf{m}})\,(\hat{\mathbf{m}}_k-\hat{\mathbf{m}})^T,\end{aligned}$$

donde $\hat{\mathbf{m}}_k$ y $\hat{\mathbf{S}}_k$ $(k=1,\cdots,C)$ son respectivamente la estimación de la media y covarianza de los patrones de la clase C_k y $\hat{\mathbf{m}}$ representa la estimación de la media total de la muestra, esto es

$$\begin{aligned}\hat{\mathbf{m}} &= \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i,\\ \hat{\mathbf{m}}_k &= \frac{1}{N_k}\sum_{i=1}^{N_k}\mathbf{x}_i \qquad \text{donde} \quad \mathbf{x}_i\in\mathcal{H}_k,\\ \hat{\mathbf{S}}_k &= \frac{1}{N_k}\sum_{i=1}^{N_k}(\mathbf{x}_i-\hat{\mathbf{m}}_k)\,(\mathbf{x}_i-\hat{\mathbf{m}}_k)^T.\end{aligned}$$

Al igual que ocurría en la sección 1.2.1, el coste computacional que requiere encontrar los $\hat{\mathbf{w}}_j$ óptimos para el discriminante lineal de Fisher (1.13), está dominado por el cálculo de $\hat{\mathbf{S}}_W$ y $\hat{\mathbf{S}}_B$. Como ya vimos en (1.2.1), calcular las matrices de dispersión $\hat{\mathbf{S}}_W$ y $\hat{\mathbf{S}}_B$ supone un coste de $\mathcal{O}(d^2\,N)$ para cada una de ellas, recordando que N es el número total de patrones y d la dimensión del espacio original. El coste del cálculo de la inversa de $\hat{\mathbf{S}}_W$ para la expresión (1.13) es $\mathcal{O}(d^3)$. Hallar el producto $\hat{\mathbf{S}}_W^{-1}\hat{\mathbf{S}}_B$ también tiene un coste de $\mathcal{O}(d^3)$. Hallar los autovalores y autovectores de la matriz $\hat{\mathbf{S}}_W^{-1}\hat{\mathbf{S}}_B$ una vez más tiene un coste de $\mathcal{O}(d^3)$ [Press et al., 1992]. Como en general, $N\gg d$, será el cálculo de las matrices de dispersión el que domine en el coste computacional del análisis discriminante de Fisher, en definitiva su coste será $\mathcal{O}(d^2\,N)$, idéntico al visto en la sección 1.2.1 para el análisis de Fisher para dos clases.

1.4. El Análisis de Fisher como Herramienta de Clasificación

1.4.1. Construcción de Clasificadores

En sentido estricto, el método de Fisher es un procedimiento de reducción de dimensionalidad y de extracción de características, más que uno de clasificación.

Sin embargo, es relativamente sencillo construir a partir del mismo diversas reglas de clasificación.

Para problemas de clasificación con C clases, es posible definir C funciones discriminantes $g_1(\mathbf{x}), \cdots, g_C(\mathbf{x})$ de tal forma que un patrón $\mathbf{x}$ es asignado a la clase C_k si se cumple

$$g_k < g_j \qquad \forall\ j \neq k$$

Obtener funciones discriminantes aplicando el método de Fisher es muy simple, basta con tomar alguna medida de la distancia de la proyección del patrón de características $\mathbf{x}$ a cada una de las medias de las clases proyectadas $\tilde{\mathbf{m}}_j$ ($j = 1, \cdots, C$). Una posible función discriminante sería

$$g_j(\mathbf{x}) = \| \mathbf{W}^T \mathbf{x} - \mathbf{W}^T \mathbf{m}_j \| = \| \mathbf{y} - \tilde{\mathbf{m}}_j \|$$

Por tanto, al patrón $\mathbf{x}$ será asignado a la clase en la que el procesamiento de la función discriminante emita un valor menor; es decir, el patrón $\mathbf{x}$ será asignado a la clase k si cumple

$$\left.\begin{array}{ccc} \| \mathbf{W}^T \mathbf{x} - \mathbf{W}^T \mathbf{m}_k \| & < & \| \mathbf{W}^T \mathbf{x} - \mathbf{W}^T \mathbf{m}_j \| \\ & \text{o} & \\ \| \mathbf{y} - \tilde{\mathbf{m}}_k \| & < & \| \mathbf{y} - \tilde{\mathbf{m}}_j \| \end{array}\right\} \quad \forall\ j \neq k \qquad (1.14)$$

Como ya hemos indicado en la sección 1.2 para el caso de dos clases y por extensión lo aplicamos para C clases, la solución del conjunto de vectores $\mathbf{W}$ óptimo no es única, permite translaciones y reescalado de los ejes, pero todas estas transformaciones lineales que van de un espacio $(C-1)$–dimensional a otro igualmente $(C-1)$–dimensional, dejan invariante la dirección del subespacio lineal de proyección, con lo que el clasificador (1.14) permanece sin cambios.

1.4.2. Clasificación en un Problema Modelo

Como problema modelo para estudiar la efectividad del análisis discriminante de Fisher vamos a considerar el problema de clasificación usado por el propio Fisher. Se trata de poder clasificar tres especies distintas de flores pertenecientes a la familia de los iris: Setosa, Versicolor y Virgínica; para ello contamos con una muestra que contiene las tres especies y las variables o características de cada medida de laboratorio son la longitud y anchura de los pétalos y sépalos, lo que indica que la dimensión del espacio original es cuatro. El discriminante de Fisher reduce el espacio de características, $\Re^4$, a un nuevo espacio bidimensional (número de especies menos uno) correspondiente a las proyecciones. En total, la muestra original contiene 50 medidas completas de cada especie de iris.

La figura (1.3) representa la proyección de las características de los iris en el subespacio bidimensional formado por los vectores procedentes de resolver el análisis discriminante de Fisher (1.13), $\mathbf{w}_1 = (-0{,}086, -0{,}455, 0{,}459, 0{,}759)$ y

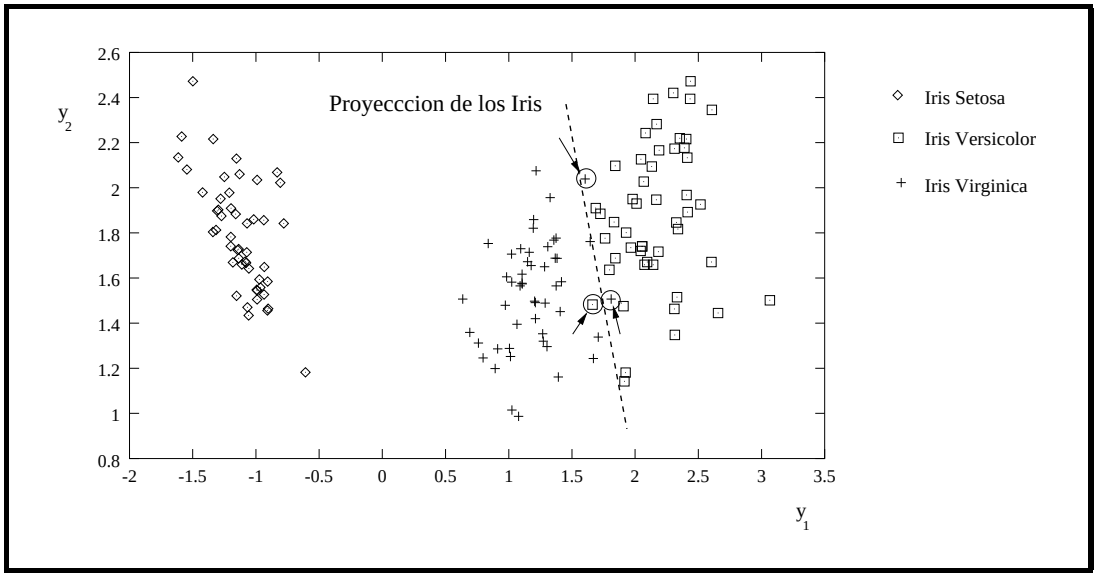


Figura 1.3: Proyección de los datos Iris

$\mathbf{w}_2 = (-0{,}043, 0{,}624, -0{,}219, 0{,}749)$. De todos los posibles $\mathbf{w}$ que optimizan (1.13), hemos tomado los que están normalizados, es decir, $\| \mathbf{w}_1 \| = \| \mathbf{w}_2 \| = 1$.

En el nuevo subespacio, los datos proyectados son clasificados utilizando un discriminante como el expresado en (1.14). La clasificación que se obtiene con los 150 datos originales viene dada en la siguiente matriz de clasificación

IRIS	*Setosa*	*Versicolor*	*Virgínica*
Setosa	50	0	0
Versicolor	0	48	2
Virgínica	0	1	49

El error cometido en clasificación es de un 2 %. En la figura (1.3) se señalan los datos mal clasificados por el discriminante de Fisher. Como se puede apreciar, los datos mal clasificados corresponden a puntos situados en los límites fronterizos de separación impuestos por el discriminante (1.14) entre las dos clases involucradas: iris Versicolor e iris Virgínica. Así mismo, en la figura (1.3) puede apreciarse que los iris Setosa quedan perfectamente clasificados.

En este ejemplo, hemos visto que el análisis discriminante lineal de Fisher permite obtener resultados buenos, pudiéndose separar en el subespacio de proyección las tres clases involucradas. Pero no siempre ocurre lo mismo: cuando a priori el problema no es linealmente separable no es posible obtener con el discriminante de Fisher una buena separación de las clases.

1.4.3. Dificultades con el Discriminante de Fisher

El principal inconveniente con el análisis discriminante de Fisher es que no es un buen clasificador para problemas en los que la frontera de separación entre clases en la proyección es no lineal o bien cuando no existe una frontera como tal, pues es tan amplio el solapamiento entre clases que es imposible definir la frontera de separación. Este último caso es el que vamos a presentar a continuación.

Para demostrarlo, tomaremos un clásico en problemas de clasificación. De la base de datos de UCI [Blake y Merz, 1998] hemos seleccionado el conjunto muestral correspondiente al estudio de la diabetes de los indios Pima. Se trata de un conjunto con 768 entradas correspondientes a 500 entradas etiquetadas como indios Pima *sin diabetes* y 268 *con diabetes*. Cada entrada está compuesta por 8 medidas médicas correspondientes a variables numéricas que pueden revelar la existencia o no de la enfermedad.

La matriz clasificación para este conjunto utilizando como discriminante el método de Fisher es la siguiente

INDIOS PIMA	*Sin Diabetes*	*Con diabetes*
Sin Diabetes	396	104
Con Diabetes	74	194

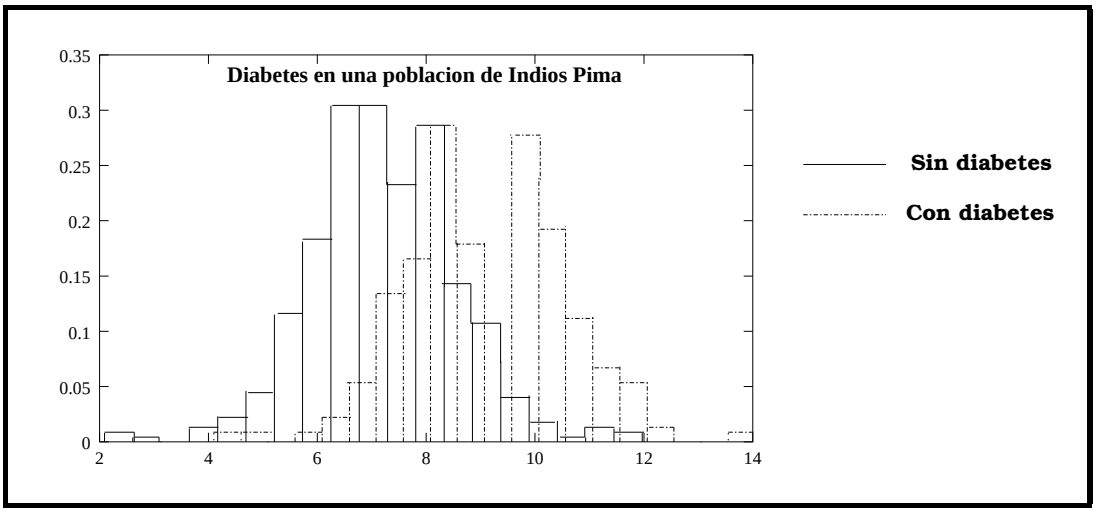


Figura 1.4: Solapamiento de los datos proyectados. Conjunto Indios Pima

La matriz de clasificación nos muestra que el error que comete el discriminante de Fisher es alto, exactamente el 23,18 %. Este error tan alto es debido a que en la recta de proyección solapan muchos de los puntos proyectados. En el histograma de la figura (1.4) se representa la población proyectada de las dos clases, y es fácil observar que existe una zona de solapamiento muy grande, lo que impide que el discriminante de Fisher sea un método eficiente.

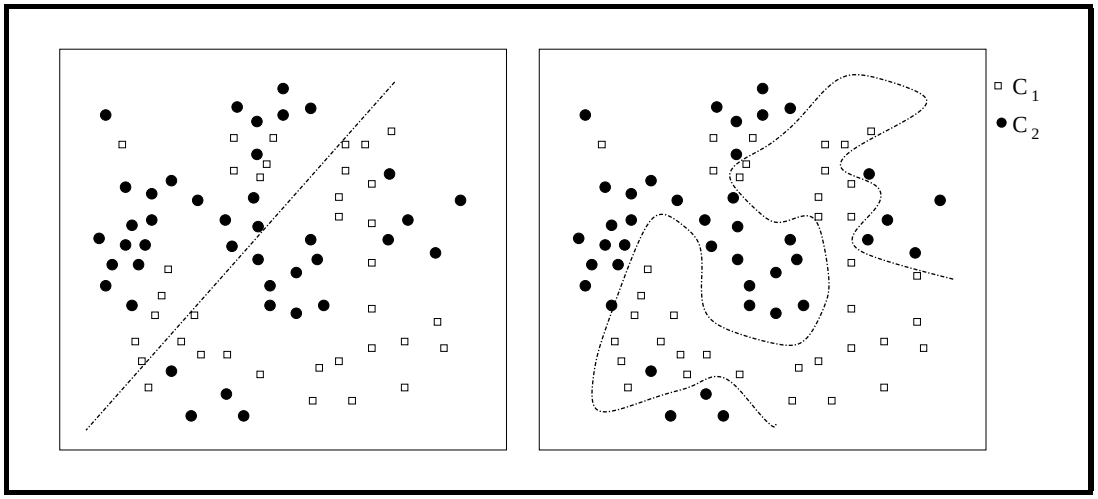


Figura 1.5: Flexibilidad en la definición de frontera al introducir no linealidad

1.5. Conclusiones

Como hemos visto a lo largo del capítulo, el análisis discriminante de Fisher es un método de reducción de dimensionalidad que puede ser aplicado como discriminante de clases. Para ello, debe ser posible proyectar las características del espacio muestral a un subespacio de dimensión $\tilde{d}$, siendo $\tilde{d}$ menor que la dimensión del espacio muestral d. Posteriormente, se definen fronteras entre las distintas clases, formadas por subespacios de dimensión $(\tilde{d}-1)$, lo que indica que las fronteras son lineales. Al no permitir fronteras con curvatura distinta de cero, sólo es posible separar problemas cuyas proyecciones sean linealmente separables.

En la figura (1.5) se ha simulado un conjunto de datos bidimensionales pertenecientes a dos clases. En la representación de la izquierda ilustra la separación de las dos clases utilizando para ello un método lineal y como se puede observar es imposible separar eficientemente los patrones pertenecientes a cada clase; por contra en la imagen de la derecha se define una frontera de separación no lineal y como se puede apreciar es sustancialmente más efectiva en su cometido: separar las dos clases.

A la vista de la ilustración de la figura (1.5) podríamos afirmar que análisis discriminantes no lineales parecen ser los métodos óptimos a la hora de buscar un buena separación de las clases, puesto que permiten una flexibilidad en la definición de la frontera de separación que no es posible con métodos lineales. En posteriores capítulos tendremos ocasión de ver alguno de los posibles métodos no lineales utilizados como discriminantes.

No obstante, no debemos desechar los métodos lineales, es más en el trabajo que nos ocupa, el análisis discriminante lineal de Fisher será relevante dentro del método de aprendizaje de la red multicapa no lineal desarrollada en el capítulo (3).

Luego, estudiar en detalle el análisis discriminante lineal de Fisher en este capítulo se ha realizado en base a dos objetivos. Primero, definir qué es un discriminante y familiarizarse con la notación a utilizar durante el resto del trabajo; para ello lo mejor es comenzar por un discriminante sencillo lineal. Segundo, asentar los conocimientos necesarios para seguir el desarrollo del capítulo (3).

Capítulo 2

Perceptrones Multicapa (PMC)

2.1. Introducción

En el capítulo anterior se demostró cómo los discriminantes lineales sólo son efectivos cuando es posible realizar una separación lineal de las clases. Como la mayoría de los problemas reales no son separables linealmente, la necesidad de buscar nuevos métodos de discriminación no lineales se hizo imperante.

Si fuera posible encontrar una función no lineal φ como función discriminante, ocurriría que las regiones de decisión estarían definidas de una forma más precisa, es decir, dejarían de estar limitadas por hiperplanos como ocurre con los discriminantes lineales, evitándose así la rigidez de las fronteras de separación entre las distintas clases (ver figura (1.5)).

El objetivo de este capítulo es presentar al lector una solución tanto para el problema de clasificación como para el de extracción de características, cuando éstos no pueden ser tratados linealmente. Se mostrará el más común de los discriminante no lineales, el Perceptron Multicapa (PMC), evaluándose en cualquier caso, por qué este perceptron es válido como discriminante, como clasificador e incluso la más sencilla de sus utilidades, como función de transformación en problemas de modelización.

Al final del capítulo se presentará la conexión que existe entre un PMC y los discriminantes de Fisher introducidos en el capítulo anterior.

2.2. Introducción a los PMCs

Métodos basados en redes neuronales artificiales han sido propuestos para introducir la no linealidad en problemas de clasificación de patrones. Los métodos utilizados se han basado tanto en tratamientos supervisados como en no supervisados [Mao y Jain, 1995], siendo quizás, el Perceptron MultiCapa (PMC) la he-

rramienta básica más utilizada hasta el momento.

Son muchas las modificaciones que se han hecho desde la aparición del primer perceptron [Rosenblatt, 1962], pero la base fundamental sigue siendo, aún hoy en día, la misma.

Lo asombroso de los PMCs es que no hay nada de mágico en su concepto. Tan sólo se trata de discriminantes lineales que trabajan en un espacio en el que las características originales han sido transformadas de forma no lineal. El gran poder de los PMCs proviene de la sencillez de sus algoritmos, consolidándose por la consiguiente facilidad de implementación. A lo que hay que añadir que la forma intrínseca de la no linealidad del problema es aprendida a partir de un conjunto de datos de entrenamiento, lo cual no es excesivamente costoso.

La figura (2.2) representa un esquema sencillo de un PMC con sólo tres capas; además se muestra la nomenclatura que se usará en la definición de la red. Un PMC es una red multicapa con al menos tres capas distribuidas de la siguiente forma:

- Una *capa de entrada*, que se podría considerar como la recepción de los atributos o características que son introducidos en la red.
- Una o varias *capas ocultas*, donde se lleva a cabo el procesamiento de la red. Se denominan capas ocultas porque lo que ocurre en las activaciones de sus unidades no puede ser observado directamente desde los entornos externos, es decir, la entrada y salida de la red. Desde este punto de vista, el interior del PMC podríamos considerarlo como una caja negra. El número de capas ocultas lo indicaremos como H.
- Una *capa de salida*, donde se alcanza el estado final de la red y se obtienen los resultados de aplicar el procesamiento matemático anterior a los atributos de entrada.

Dentro de cada capa existen una o varias unidades, también denominadas *neuronas* por analogía con el comportamiento de las neuronas biológicas, ver esquema de la figura (2.1). La imagen superior de la figura (2.1) ilustra la morfología de una neurona biológica y cómo fluye la información desde las dendritas a las ramificaciones terminales del axón. Las dendritas reciben información procedente de las terminales axónicas de otras neuronas; a estas conexiones se les denomina *sinapsis*. En el cuerpo celular o *soma* es donde se procesa toda la información que recibe la neurona procedente de las múltiples sinapsis efectuadas con neuronas próximas a ella. Finalmente a través del axón se propaga la nueva información procesada a otras neuronas próximas, produciéndose nuevas sinapsis axón–dendrita. En la imagen inferior se representa el símil adoptado por la neurona artificial. El modelo presentado es un modelo muy sencillo y primitivo; hoy en día se sabe que son muchos los mecanismos que intervienen en el proceso de transmisión de los impulsos nerviosos [Stevens, 1987], [Kandel et al., 2000].

Las distintas capas de un PMC están interconectadas por pesos adaptativos; en las figuras (2.1 b) y (2.2), esto viene representado por uniones entre unidades de distintas capas. Siguiendo con la analogía con neurobiología, estas conexiones son llamadas *sinapsis* y a sus valores *pesos sinápticos*.

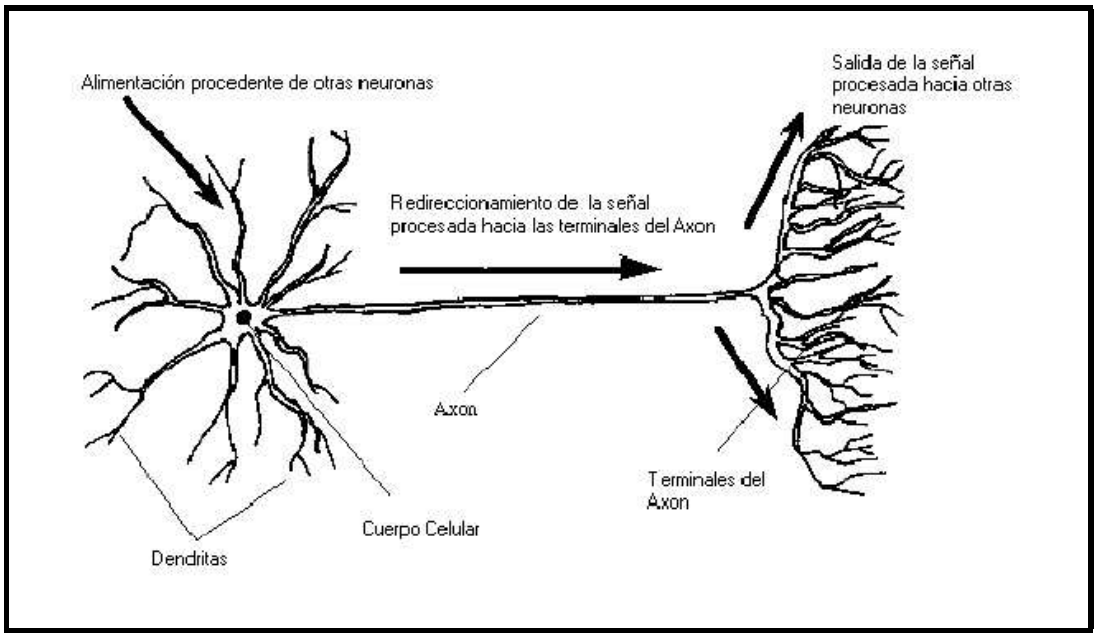
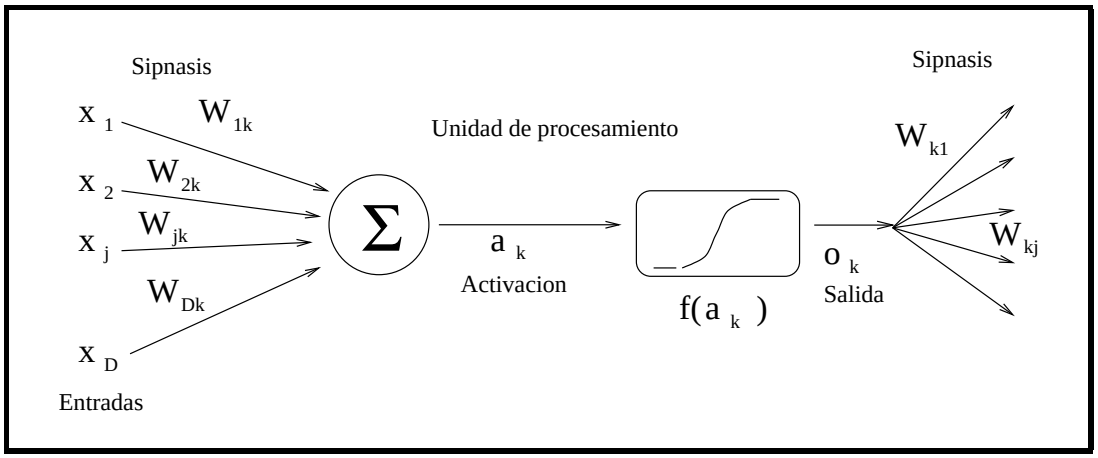


Figura 2.1: Morfología y similitud entre dos neuronas: a) biológica y b) artificial.



En cada capa, exceptuando la capa de salida, existe una neurona o unidad especial cuyo valor siempre es la unidad (1) y cuyo peso sináptico se denomina *sesgo*. Al introducir esta unidad se permite la posibilidad de desplazar el origen del espacio de pesos respecto del origen de coordenadas. En la figura (2.3) se puede apreciar el efecto que produce introducir el peso correspondiente al sesgo. En dicha figura se ha definido un conjunto de patrones pertenecientes a dos clases

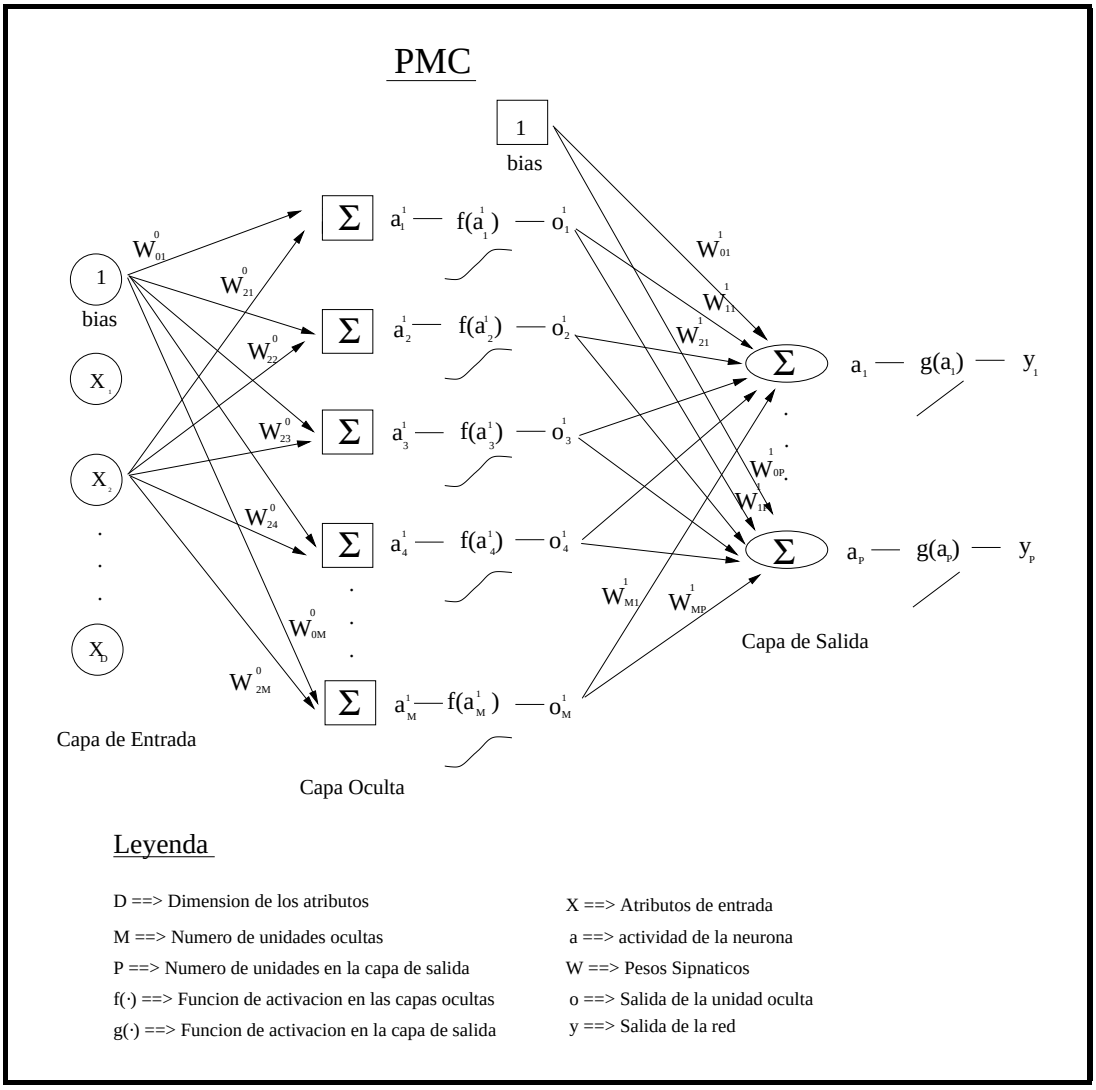


Figura 2.2: Esquema de la arquitectura de un PMC con una única capa oculta.

y se busca un separador sencillo lineal. En el caso de la izquierda no se tiene en cuenta la posibilidad de introducir un sesgo, con lo cual el separador está forzado a pasar por el origen de coordenadas y en el caso de la derecha se libera al separador de esta restricción. Como se puede apreciar, en este segundo caso la mejoría en el cometido de separar las dos clases es notable.

Partiendo de la capa de entrada, la información se va transmitiendo de capa en capa hacia la capa de salida. Las redes que se comportan así se dicen que son *redes de flujo directo*. Dentro de cada capa oculta, cada unidad procesa la información que le llega procedente de unidades de la capa anterior y propaga dicho procesamiento a las unidades de la capa siguiente, hasta llegar a la capa de salida, donde se obtiene el resultado de la red.

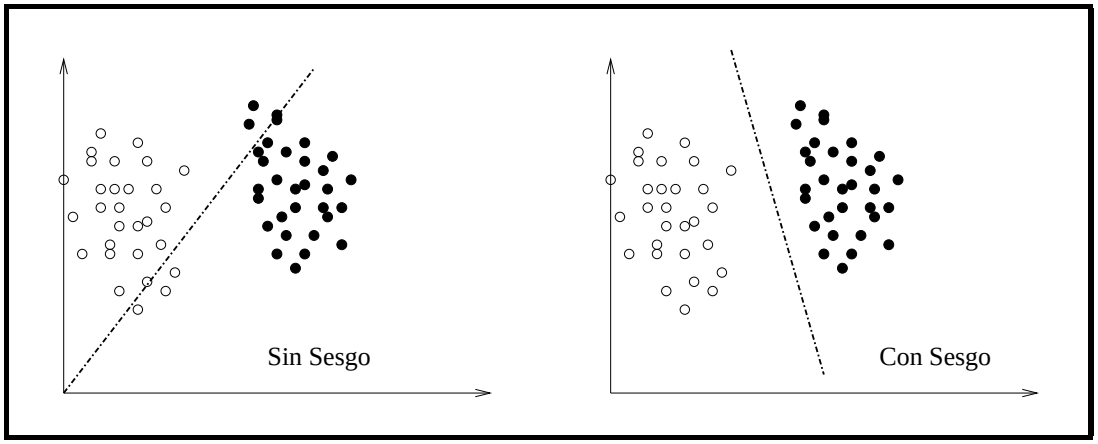


Figura 2.3: Efecto de considerar el sesgo.

Cada unidad oculta computa la suma ponderada de las entradas que le llegan; a este escalar se le denomina *activación de la neurona*. La notación matemática de la actividad de una unidad genérica k perteneciente a la capa oculta h viene dada por

$$a_k^h = \sum_{j=1}^{n^{h-1}} w_{jk}^{h-1} o_j^{h-1} + w_{0k}^{h-1} = \sum_{j=0}^{n^{h-1}} w_{jk}^{h-1} o_j^{h-1} = (\mathbf{w}_k^{h-1})^T \mathbf{o}^{h-1}. \qquad (2.1)$$

Antes de continuar explicando qué es cada término, haremos un inciso para precisar que significan los subíndices y superíndices de la expresión anterior. Los subíndices indican unidades dentro de las capas y los superíndices indican capas. Los valores que pueden tomar los superíndices van de 0 a H ($h = 0, \cdots, H$), siendo H el número de capas ocultas. Si el valor que toma el superíndice es el 0, indica que está involucrada la capa de entrada; en el caso de que tomara el valor H, la capa involucrada sería la última capa oculta. Cuando los subíndices están formados por un par, por ejemplo jk en w_{jk}, quiere indicar conexiones entre una unidad emisora y otra receptora pertenecientes a capas contiguas. Los subíndices jk siempre indican conexiones en el sentido directo ($\rightarrow$), es decir, el que va de la capa de entrada hacia la capa de salida.

Continuando, w_{jk}^{h-1} es el peso de la conexión entre la unidad j de la capa emisora $h-1$ a la unidad k de la capa receptora h. El término genérico n^h indica el número de unidades de la capa h, luego el subíndice correspondiente a las unidades de esta capa toma los valores de $j = 0, \cdots, n^h$, donde $j = 0$ corresponde al término del sesgo. o_j^{h-1} representa la salida de la unidad j en la capa emisora, ($j = 0, \cdots, n^{h-1}$). Si la capa emisora fuera la de entrada, o_j^0 sería reemplazado por la característica j del patrón original, $o_j^0 \equiv x_j$.

En cualquier caso, el valor de o_0^h (subíndice $j = 0$) será siempre la unidad (1), y corresponde al término del sesgo. Para una mayor comprensión de los índices,

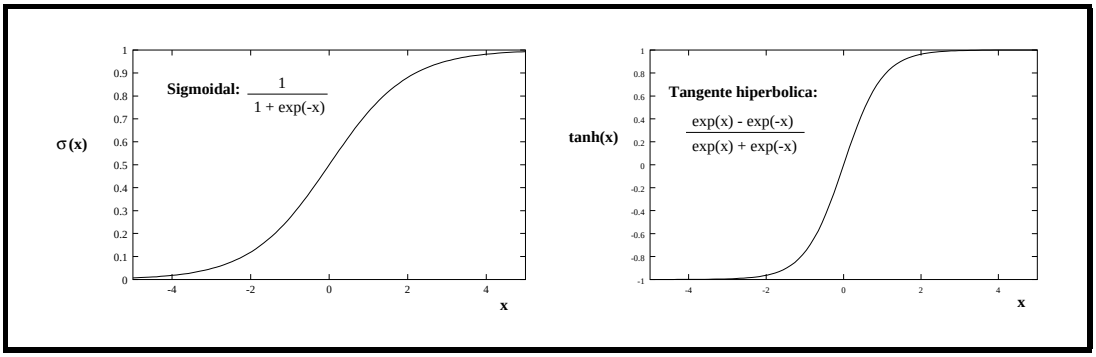


Figura 2.4: Funciones de activación sigmoidal y tangente hiperbólica

remitimos a la figura (2.2).

Cuando la unidad oculta ha computado su activación, emite una salida que es el resultado de aplicar una función no lineal a su activación, $o_k^h = f(a_k^h)$. A esta función no lineal se le denomina *función de activación*. Las funciones de activación más utilizadas en las unidades ocultas son la función sigmoidal o bien la tangente hiperbólica, también denominada sigmoidal bipolar. Estas dos funciones están representadas en la figura (2.4), donde la ilustración de la izquierda es la función sigmoidal expresada por

$$\sigma(x) = \frac{1}{1 + \exp(-x)}$$

y la imagen de la derecha es la función tangente hiperbólica

$$\tanh(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)}.$$

El dominio de las dos funciones es $(-\infty, +\infty)$ y ambas funciones son diferenciables en todo su dominio. La principal diferencia entre ambas funciones radica en el recorrido: (0,1) para la función sigmoidal y (-1,1) para la tangente hiperbólica.

Una característica muy útil en ambas funciones es que sus derivadas pueden ser expresadas de forma sencilla a partir de los valores de la propia función. A modo de ejemplo, la derivada de la función sigmoidal es tan simple como

$$\sigma'(x) = \sigma(x)(1 - \sigma(x)).$$

Las funciones sigmoidal y tanh están relacionadas, podríamos expresar una en función de otra, por ejemplo $\tanh(x/2) = 2\sigma(x) - 1$, siendo $\sigma(x)$ la función sigmoidal. Con lo cual, un PMC entrenado con la función tanh es esencialmente equivalente a un PMC entrenado con la función sigmoidal; la diferencia estará en los valores que tomen los pesos de ambas redes. En general, parece que la función tanh converge más rápidamente que la sigmoidal. Esto es debido a su simetría en el rango de definición.

Por último, las unidades de la capa de salida hallan su valor calculando igualmente su activación

$$a_k = \sum_{j=1}^{n^H} w_{jk}^H o_j^H + w_{0k}^H = \sum_{j=0}^{n^H} w_{jk}^H o_j^H = (\mathbf{w}_k^H)^T \mathbf{o}^H, \tag{2.2}$$

y según la aplicación a la que esté destinada la red, para obtener la salida y_k, a la activación a_k se le puede aplicar una función no lineal, $y_k = g(a_k)$. En muchos casos, es la propia activación la salida de la red, $y_k = a_k$. Es decir, la función de transferencia que se aplica es lineal $g(a) = a$.

Por tratarse de la capa de salida y dado que es única, hemos optado por no incluir superíndices en la representación de sus activaciones a_k. El subíndice k ($k = 1, \cdots, n^S$) representa a las unidades de la capa de salida; n^S indica el número de unidades en dicha capa.

2.2.1. Aproximación universal en PMCs

Kolmogorov [Kolmogorov, 1957] demostró que cualquier función continua de varias variables, $\Psi(\mathbf{x})$, definida en un dominio cerrado y acotado puede ser representada por la superposición de un número relativamente pequeño de funciones de una sola variable. Basándose en el teorema de Kolmogorov, posteriormente se demostró que cualquier función puede ser representada exactamente por un PMC de tres capas, referencias en [Kürková, 1991], [Kürková, 1992] y [Girosi y Poggio, 1989].

Con el teorema de Kolmogorov se garantiza la existencia de una red ideal, pero en sí el teorema no es nada práctico debido a que la topología de la red que demuestra dicho teorema presenta grandes inconvenientes a la hora de implementarla en una aplicación real. Esta red requiere un número fijo de unidades ocultas que depende directamente de la dimensión de los patrones de entrada. Estamos pues, ante una utopía: si se quiere modelizar una función $\Psi(\mathbf{x})$, fijar a priori el número de parámetros libres restringe la flexibilidad en la búsqueda de la nueva función de modelización.

Por otro lado, en cuanto a las funciones de activación, existen dos puntos muy frágiles: las funciones de activación son dependientes de la función a modelizar $\Psi(\mathbf{x})$ y hoy en día no hay técnicas desarrolladas que permitan saber cuál es la forma funcional de dichas funciones de activación. El otro punto débil es que están permitidas funciones de activación abruptas; es más si se obligara a que las funciones fueran suaves, el teorema podría malograrse. El inconveniente de trabajar con funciones de activación abruptas es que la red se vuelve muy sensible a las variables de entrada, perdiéndose la tan deseada *generalización*.

En la práctica, en redes neuronales se realiza el proceso contrario al presentado en la teoría de aproximación universal de PMCs (referencias en [Scarselly y Tsoi, 1998], [Kürková, 1991], [Kürková, 1992] y

[Girosi y Poggio, 1989]). Primero, se fija para todas las unidades ocultas la expresión de las funciones de activación independientemente de la función a modelizar y a lo largo del entrenamiento de la red se ajustan el número de unidades ocultas, el valor de los pesos y según la red los parámetros libres de la propia función de activación. Las redes que permiten modificar parámetros en la función de activación se dice que usan métodos semiparamétricos. Normalmente, los parámetros de la función de activación son ajustados mediante métodos heurísticos.

Luego, hemos visto que una red multicapa puede modelizar cualquier función, pero decidir cuál es la arquitectura óptima de la red es un problema con el que hay que enfrentarse en cada aplicación y en ningún caso se asegura poder llegar a obtener la arquitectura óptima con tan sólo construir un PMC. Con frecuencia, son varias las pruebas que se realizan con distintas e incluso con la misma arquitectura antes de confirmar una arquitectura como satisfactoria.

2.3. Entrenamiento de un PMC

El aprendizaje en un PMC está basado en la definición de una función de error apropiada, que será minimizada respecto a los pesos de la red. El algoritmo desarrollado para evaluar el gradiente de la función de error en un PMC se denomina *retropropagación*; su nombre se debe a que se trata de una propagación del error hacia atrás a lo largo de la red. La técnica de *retropropagación* fue popularizada por Rumelhart, Hinton y Williams [Rumelhart et al., 1986].

Los algoritmos de aprendizaje de una red involucran un proceso iterativo. Cada iteración en la jerga de redes neuronales se denomina *época*. A su vez, cada época puede ser dividida en dos etapas. En la primera etapa, la derivada de la función de error respecto a los pesos es evaluada. De aquí parte la importancia del método de retropropagación; computacionalmente es un método muy eficiente. Las derivadas son halladas por propagación del error hacia atrás. Es importante indicar que como derivadas podríamos incluir tanto las matrices correspondientes al Jacobiano como posibles alternativas a éste, todo ello dependiendo del método de descenso elegido en la segunda etapa; en el capítulo 4 se refleja para casos concretos a qué tipo de matrices alternativas nos referimos.

En la segunda etapa, las derivadas obtenidas anteriormente son utilizadas para modificar los pesos de la época anterior, de modo que son estas pequeñas modificaciones locales, realizadas a lo largo de varias épocas, las que globalmente van a minimizar la función de error.

Minimizar numéricamente una función continua, multiparamétrica y diferenciable ha sido extensamente estudiado [Press et al., 1992] y las distintas aproximaciones encontradas pueden ser aplicadas en el campo de redes neuronales. Luego, métodos tales como descenso por gradiente, Quasi–Newton, gradiente conjugado, Levenberg–Marquardt, ... son los más utilizados en la segunda etapa del

entrenamiento.

Originalmente, la función de error del PMC es el error cuadrático medio y el método de minimización utilizado es el descenso por gradiente. Pero hoy en día, los PMCs son implementados con diversas funciones de error y distintos métodos de minimización.

2.3.1. Evaluación de la derivada de la función de error

El objetivo de esta sección es desarrollar el procedimiento de retropropagación con el propósito de evaluar la derivada de la función de error respecto de los pesos de la red. Consideramos una función de error genérica $\mathcal{E}$ diferenciable y dependiente de la salida de la red $\mathbf{y}$, $\mathcal{E} = \mathcal{E}(y_1, \cdots, y_{n^s})$. La dependencia de la función de error $\mathcal{E}$ con el peso w_{jk} se realiza a través de la activación de la unidad k de la capa receptora, pudiendo ser ésta bien una capa oculta (2.1) o bien la capa de salida (2.2); entonces $\partial\mathcal{E}/\partial w_{jk}^h$ se obtiene aplicando la regla de la cadena en derivadas parciales

$$\frac{\partial \mathcal{E}}{\partial w_{jk}^h} = \frac{\partial \mathcal{E}}{\partial a_k^{h+1}} \frac{\partial a_k^{h+1}}{\partial w_{jk}^h}. \tag{2.3}$$

Comenzaremos primero por ver que ocurre si w_{jk}^h es un peso correspondiente a la conexión entre la última capa oculta y la de salida, w_{jk}^H. Dado que $\mathcal{E} = \mathcal{E}(\mathbf{y})$, entonces $\partial\mathcal{E}/\partial a_k$ también puede descomponerse en

$$\frac{\partial \mathcal{E}}{\partial a_k} = \frac{\partial \mathcal{E}}{\partial y_k} \frac{\partial y_k}{\partial a_k},$$

y como el valor de la unidad k en la capa de salida es $y_k = g(a_k)$, entonces $\partial y_k/\partial a_k = g'(a_k)$. Luego, la expresión de $\partial\mathcal{E}/\partial a_k$ queda representada por términos todos ellos conocidos en la capa de salida

$$\frac{\partial \mathcal{E}}{\partial a_k} = g'(a_k) \frac{\partial \mathcal{E}}{\partial y_k}. \tag{2.4}$$

Partiendo de la expresión de la activación de una unidad perteneciente a la capa de salida (ecuación (2.2)) el segundo término de la expresión (2.3) viene dado por la salida en la última capa oculta de la unidad emisora j

$$\frac{\partial a_k}{\partial w_{jk}^H} = o_j^H.$$

La expresión final para (2.3) en función de términos conocidos, cuando el peso involucrado corresponde a una conexión entre la última capa oculta y la capa de salida, nos queda de la forma

$$\frac{\partial \mathcal{E}}{\partial w_{jk}^H} = o_j^H g'(a_k) \frac{\partial \mathcal{E}}{\partial y_k}. \tag{2.5}$$

A continuación, veremos como podemos obtener expresiones similares a (2.5) cuando los pesos involucrados pertenecen a conexiones cuyas unidades receptoras son capas ocultas. Se trata pues de hallar el valor de $\partial\mathcal{E}/\partial w_{jk}^h$, ecuación (2.3), donde h puede tomar los valores de 0 al número de unidades ocultas menos uno $(h = 0, \cdots, H-1)$.

Comenzaremos por evaluar $\partial\mathcal{E}/\partial {a_k}^{h+1}$, donde a_k^{h+1} es la activación de la unidad k de la capa receptora $(h+1)$, $(h = 0, \cdots, H-1)$. Aplicando una vez más la regla de la cadena

$$\frac{\partial\mathcal{E}}{\partial a_k^{h+1}} = \sum_l \frac{\partial\mathcal{E}}{\partial a_l^{h+2}} \frac{\partial a_l^{h+2}}{\partial a_k^{h+1}}, \tag{2.6}$$

donde el sumatorio se efectúa sobre todas las unidades l con las que la unidad emisora k está conectada.

Como la expresión de la actividad en la unidad receptora l puede expresarse por $a_l^{h+2} = \sum_k w_{kl}^{h+1} f(a_k^{h+1})$, su derivada respecto de la actividad de una unidad emisora será

$$\frac{\partial a_l^{h+2}}{\partial a_k^{h+1}} = w_{kl}^{h+1} f'(a_k^{h+1})$$

y la expresión (2.6) en función de términos conocidos de capas por encima a la actual emisora, vendrá dada por

$$\frac{\partial\mathcal{E}}{\partial a_k^{h+1}} = f'(a_k^{h+1}) \sum_l w_{kl}^{h+1} \frac{\partial\mathcal{E}}{\partial a_l^{h+2}}.$$

Por sencillez, vamos a denominar $\partial\mathcal{E}/\partial a_l^{h+2}$ como Υ_l, de este modo la expresión anterior queda de la forma

$$\Upsilon_k^{h+1} = f'(a_k^{h+1}) \sum_l w_{kl}^{h+1}\ \Upsilon_l^{h+2}. \tag{2.7}$$

La expresión (2.7) indica que el valor Υ para una determinada unidad oculta se obtiene por retropropagación de los Υ de las capas superiores, partiendo inicialmente por los correspondientes a la capa de salida (ecuación (2.4)) y posteriormente iterando recursivamente según la ecuación (2.7) hasta la capa oculta en consideración.

El segundo término de la expresión (2.3), es idéntico al que vimos anteriormente para el caso de las conexiones entre la última capa oculta y la capa de salida, es decir, el valor de $\partial a_k^{h+1}/\partial w_{jk}^h$ será la salida de la unidad emisora

$$\frac{\partial a_k^{h+1}}{\partial w_{jk}^h} = o_j^h.$$

La expresión final para (2.3) vendría dada por

$$\frac{\partial\mathcal{E}}{\partial w_{jk}^h} = \Upsilon_k^{h+1}\ o_j^h = o_j^h\ f'(a_k^{h+1})\ \sum_l w_{kl}^{h+2} \Upsilon_l^{h+2}, \qquad \text{donde} \quad \Upsilon_l^{h+2} = \partial\mathcal{E}/\partial a_l^{h+2}. \tag{2.8}$$

Resumiendo, el procedimiento de retropropagación para evaluar la derivada del error total respecto del peso w_{kj}^h seguirá los siguientes pasos:

1. Usando la metodología explicada en la sección 2.2 de este capítulo, se obtiene a partir del vector de características $\mathbf{x}$ la salida de la red $\mathbf{y}$.

2. Se evalúan los $\Upsilon_k^h = \partial\mathcal{E}/\partial a_k^h$ para todas las unidades de la capa de salida; para ello se utiliza la fórmula (2.4).

3. Se retropropagan los Υ usando la ecuación recursiva (2.7) con el objetivo de calcular todos los Υ_k^h de las unidades ocultas de la red.

4. Por último, se calcula la derivada de la función de error respecto de los pesos de la red, aplicando para ello las ecuaciones (2.5) o (2.8) en función de que la conexión involucre como capa receptora a la capa de salida o bien a una capa oculta.

2.3.2. Actualización de los pesos en un PMC

Una vez que ya sabemos calcular las derivadas de la función de error respecto a los pesos de la red con el algoritmo de retropropagación, nos queda ver cómo éstas son utilizadas en la actualización de los pesos. Dependiendo del tipo entrenamiento de la red, la actualización de los pesos variará. En esta sección veremos cuáles son los métodos de actualización de pesos más usuales.

Los dos protocolos base en el entrenamiento de redes son: *batch* y *on–line*. Por tanto, los entrenamientos se diferenciarán entre aquellos que usan métodos que funcionan sólo con procesos *batch*, o bien sólo con procesos *on–line* y los que pueden trabajar con los dos tipos con tan sólo efectuar una pequeña adaptación o modificación.

En el entrenamiento en *batch*, todos los patrones del conjunto de entrenamiento son presentados a la red antes de que tenga lugar la actualización de los pesos. Precisamente, ésta es la definición de una época; con el entrenamiento en *batch* se repetirá el proceso de actualización de pesos época a época hasta llegar a la convergencia.

En entrenamientos tipo *on–line* el patrón es presentado una y sólo una vez, y la actualización de los pesos es lógicamente patrón a patrón. Con este método se tiene la ventaja de que no es necesario almacenar los patrones en memoria y será especialmente útil cuando son muchos los patrones a disposición o bien en aplicaciones de control donde se necesita seguir en todo momento la evolución del sistema, y no se conocen entradas futuras.

Existe un caso particular del entrenamiento *on–line* que se le denomina entrenamiento estocástico, en el cual los patrones son elegidos aleatoriamente del

conjunto de entrenamiento y los pesos de la red son actualizados en cada presentación del patrón. El orden a seguir en los patrones de entrada es indistinto, puede ser secuencial o aleatorio e incluso en este último no es necesario que todos los patrones de entrenamiento sean mostrados con la misma probabilidad, lo que permite que alguno de los patrones del conjunto de entrenamiento sean visto más veces que otros. La diferencia conceptual entre el entrenamiento *on–line* puro y el estocástico es que los patrones en este último pueden ser vistos por la red más de una vez. Por otro lado, con el entrenamiento *on–line* no se tiene un conjunto de entrenamiento como tal, sino que es una secuencia de datos que le van llegando a la red, pero sin tener por ello la visión de un conjunto muestral predeterminado.

En virtud de las aproximaciones *on–line* y estocásticas podríamos señalar que al permitir modificar los valores de los pesos patrón a patrón, el error disminuye localmente con la presentación del patrón, pero globalmente respecto al conjunto de entrenamiento, se puede llegar a producir un aumento en la función de error total. Estas pequeñas fluctuaciones en la búsqueda de los nuevos pesos podrían permitir escapar de mínimos locales en la función de error total. No ocurre lo mismo con métodos *batch* que admiten técnicas de segundo orden, tales como quasi–Newton o gradiente conjugado en los que se garantiza que la función error total nunca aumenta. Como conclusión, si estos algoritmos de minimización alcanzan en su camino un mínimo local quedarían atrapados en él de por vida.

Si además, como ocurre con el entrenamiento estocástico, se introduce la posibilidad de aleatoriedad en la presentación de los patrones, se consigue que el entrenamiento de la red sea no determinista. No ocurre lo mismo en procesos *batch*, pues éstos son deterministas; será la elección de los pesos iniciales la que determine a qué mínimo converge el algoritmo.

Descenso por Gradiente

Uno de los algoritmos más sencillos utilizados en el entrenamiento de redes es el denominado *descenso por gradiente*. Es un algoritmo que está adaptado tanto para procesos en *batch* como procesos *on–line* y por extensión estocásticos.

Su sencillez en la implementación es lo que le ha permitido ser ampliamente utilizado. Pero tiene una gran desventaja, se trata de un algoritmo de convergencia lenta, por lo que en muchos de los casos procesados con entrenamiento en modo *batch* es sustituido por otros algoritmos de convergencia más rápida, tales como el método quasi–Newton, gradiente conjugado, Levenberg–Marquardt, $\cdots$.

Sabiendo que la función criterio a minimizar $\mathcal{E}$ es una función diferenciable respecto a los pesos $\mathbf{W}$ y asumiendo la versión *batch*, el procedimiento a seguir con el método de descenso por gradiente será el siguiente:

1. Se inicializan los pesos de la red, $\mathbf{W}^0$, normalmente de forma aleatoria.
2. Posteriormente cada vector de pesos es actualizado mediante un pequeño

desplazamiento en el espacio de pesos y en la dirección en la que $\mathcal{E}$ decrece globalmente más rápidamente, es decir, en la dirección de $-\nabla_{\mathbf{W}}\mathcal{E}$. Iterando el proceso, se genera una secuencia de vectores $\mathbf{W}^t$, cuyas componentes son calculadas como

$$w_{kj}^{t+1} = w_{kj}^t - \eta \frac{\partial \mathcal{E}}{\partial w_{kj}}\bigg|_{\mathbf{W}^t}$$

donde η es el parámetro denominado constante de aprendizaje y en su forma más sencilla se trata de una constante de magnitud pequeña y siempre positiva ($0 < \eta < 1$).

Observese que en la fórmula anterior el superíndice del peso genérico w_{kj} se refiere a la evolución temporal y no como en los casos anteriores a la capa de procedencia. Hemos eliminado el indicador de la capa porque consideramos que no es transcendental para la discusión que nos ocupa en este momento y se simplifica la nomenclatura asociada.

Bajo determinadas condiciones, la secuencia de pesos converge al punto en el que $\mathcal{E}$ es mínimo.

La elección de η es claramente crítica: si η es muy pequeña la convergencia a los pesos óptimos $\mathbf{W}^*$ es muy lenta, pues en cada paso el avance hacia el mínimo es ínfimo. Si por el contrario, η fuera muy grande, se podrían producir oscilaciones divergentes que impedirían alcanzar el mínimo. Otra posibilidad es elegir η de forma adaptativa, esto es, que comience con valores relativamente altos, de tal forma que al principio los pasos son más amplios (sería movernos en la estructura grosera de la superficie de error) y a medida que se va avanzando hacia el mínimo el paso se hace más pequeño, afinando en la zona del mínimo. Una elección común de η adaptativa es $\eta_t \propto 1/t$, aunque esta elección puede conducirnos a una muy lenta convergencia si el número de iteraciones requeridas para alcanzar el mínimo fuera muy alto. Variaciones del tipo: $\eta_{t+1} = (1 - \eta_t)\ \eta_t$ permite disminuciones progresivas de η_t durante el entrenamiento, afinando la evolución hacia el mínimo y con la ventaja de no ralentizar el proceso.

Hay que hacer hincapié que con entrenamientos *on–line* no existe el concepto de época como el de presentar a la red el conjunto de entrenamiento al completo; aquí el aprendizaje se rige por el número de patrones presentados a la red lo que también se denomina como número de épocas. Si consideramos la versión *on–line*, en donde el aprendizaje se hace patrón a patrón, la actualización del conjunto de pesos $\mathbf{W}^t$ a $\mathbf{W}^{t+1}$, es decir completar una época, implica evaluar el gradiente por cada uno de los patrones de entrada

$$w_{kj}^{t+1} = w_{kj}^t - \eta \frac{\partial \mathcal{E}\left(\mathbf{X}^{t+1}, \mathbf{W}\right)}{\partial w_{kj}}\bigg|_{\mathbf{W}^t}$$

Un gran inconveniente en el descenso por gradiente procede de aquellos casos en los que la curvatura de la superficie de $\mathcal{E}$ varia significativamente con la di-

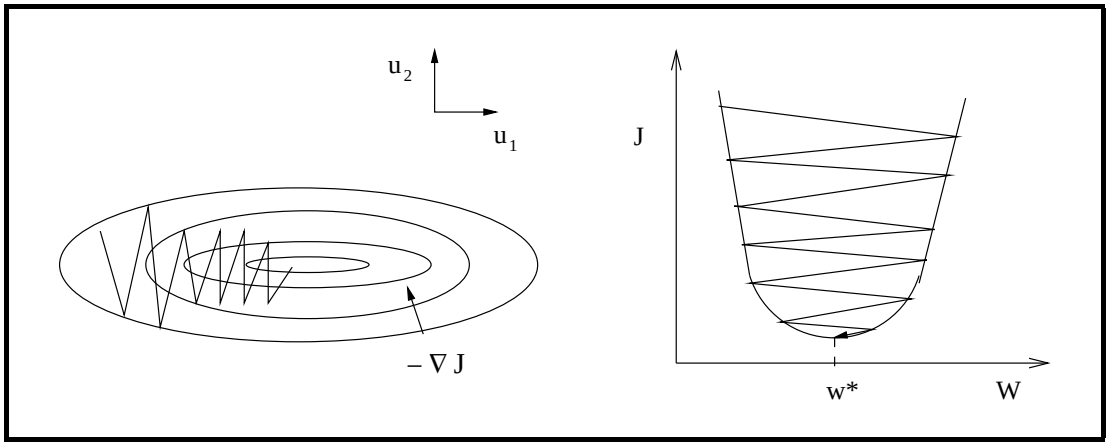


Figura 2.5: Evolución del descenso por gradiente cuando la función a minimizar tiene diferente curvatura a lo largo de las distintas direcciones

rección. En estos casos, en la mayoría de los puntos de la superficie su gradiente local no apunta en la dirección del mínimo, lo que provoca que el recorrido hacia el mínimo se realice en forma zig–zag, con un avance muy poco efectivo, necesitándose muchos pequeños pasos antes de converger; esto hace que el método sea bastante ineficiente.

La figura (2.5) muestra precisamente ese efecto. En la parte de la izquierda se representan distintas superficies de nivel de $\mathcal{E}$ en dos dimensiones, como se puede ver la forma de $\mathcal{E}$ es un valle profundo y en la derecha se representa un corte seccional del valle. Los vectores u_1 y u_2 representan los autovectores de la matriz del hessiano. En el esquema, se aprecia que el gradiente local $-\nabla\mathcal{E}$ de la mayoría de los puntos en el espacio de pesos no apunta hacia el mínimo de $\mathcal{E}$, lo que provoca que sean muchos los pasos que realiza el descenso por gradiente oscilando a lo largo del valle y dificultando así el progreso directo hacia el mínimo.

Métodos en los que se tiene en cuenta la información de la curvatura de la superficie o lo que es lo mismo, métodos que introducen el hessiano de la función $\mathcal{E}$ en su procesamiento, son más eficientes en cuanto al tiempo de convergencia. Alguno de estos métodos de segundo orden serán presentados en capítulos posteriores como mejoras en el entrenamiento de la red que allí se presentará.

2.3.3. Final del entrenamiento

Hasta ahora, hemos visto como se entrena la red con distintos protocolos de entrenamiento, pero no hemos dicho nada de cuándo se para el proceso iterativo, es decir, cuándo se prevee que la red está bien entrenada. Algunos de los posibles criterios de parada a utilizar son:

1. Parar después un número fijo de iteraciones. El problema con este criterio

es que es difícil saber por adelantado cuántas iteraciones son las que se necesitan, aunque una idea aproximada se puede obtener a partir de pruebas preliminares.

2. Parar cuando una cantidad determinada de UCP (Unidad Central de Procesos) ha sido consumida. De nuevo, es difícil saber cuál es la cantidad necesaria y la estimación requiere como en el caso anterior realizar pruebas preliminares con distintas arquitecturas.

3. Parar cuando la función de error es menor que una cierta cantidad. El problema radica en que el valor especificado puede que nunca sea alcanzado, luego restricciones en el tiempo de UCP deben ser añadidas.

4. Parar cuando el cambio en la función de error es menor que un valor especificado. El inconveniente es que puede conducirnos a una terminación prematura en el caso de que la función de error decrezca muy lentamente, cosa que ocurre en zonas donde la curvatura de la función, medida por el hessiano, está muy próxima a cero. Sería el equivalente a lo que denomina un *plato* en el transcurso fluvial de un río, es decir aquella zona en la que el cauce del río es muy ancho y el agua discurre con tal tranquilidad que da la sensación de estar estancada, pero no es cierto en absoluto, existe un avance hacia la desembocadura que sería el simil con el mínimo de la función.

5. Parar el entrenamiento cuando el error medido en un conjunto de patrones de validación comienza a aumentar. Cuando pensamos en un problema de clasificación, un buen criterio de parada sería el aumento del porcentaje de patrones mal clasificados.

6. Basado en el anterior punto, parar cuando la suma del error de entrenamiento y el de validación comienza a aumentar. Esta medida se denomina validación cruzada y es quizás la mejor de las alternativas propuestas.

2.4. PMCs como clasificadores

Los métodos probablemente más difundidos hoy en día para la construcción de clasificadores se basan en traspasar el problema de clasificación a un problema de minimización de una cierta función criterio, generalmente una suma de errores cuadráticos. Precisamente éste es el enfoque de los PMCs, con lo cual es fácil deducir que una de las principales aplicaciones de los PMCs será la construcción de clasificadores no lineales.

Existen dos formas distintas de enfocar la resolución del problema de clasificación utilizando PMCs. La más simple es identificar al PMC con una función discriminante no lineal, de tal modo que cuando un patrón nuevo es presentado al PMC entrenado, la salida de la red asignaría directamente a qué clase pertenece

dicho patrón de entrada. La segunda aproximación, utiliza el PMC para modelar la *probabilidad a posteriori* de cada clase $p(\Omega_c|\mathbf{X})$, donde Ω_c indica la clase c y $\mathbf{X}$ el vector de características de entrada a la red. La decisión final para clasificar el patrón $\mathbf{X}$ vendrá dada por la regla óptima de Bayes [Duda et al., 2001]: el patrón $\mathbf{X}$ es asignado a la clase Ω_c si cumple que

$$p(\Omega_c|\mathbf{X}) > p(\Omega_l|\mathbf{X}) \qquad \forall \quad l \neq c. \tag{2.9}$$

Cuando un PMC se entrena con el objetivo de clasificar un conjunto de patrones, por cada patrón original $\mathbf{X}_i$, se necesita añadir una etiqueta $\mathbf{t}_i$ que obedece a algún tipo de codificación para distinguir las distintas clases. Entonces, cada patrón vendrá identificado por el par $(\mathbf{X}_i,\mathbf{t}_i)$. Quizás, para un conjunto de patrones distribuidos en C clases, la codificación más usada es aquella en la que cada clase es identificada por un vector canónico C–dimensional, de tal modo que la clase Ω_c vendrá representada por el vector $\mathbf{e}^{\Omega_c} = (0, \cdots, 1, \cdots, 0)^T$, donde el único "1" que aparece en el vector corresponde a la posición c-ésima.

Para la construcción del clasificador se parte entonces de una cierta transformación $\mathbf{Y} = \mathcal{F}(\mathbf{X}, \mathbf{W})$, dependiente de un cierto vector de parámetros $\mathbf{W}$ que representa el conjunto de pesos de la red. $\mathcal{F}(\mathbf{X}, \mathbf{W})$ es la función de transferencia definida por la propia arquitectura del PMC y sus correspondientes pesos; esta función de transferencia transforma el vector D–dimensional $\mathbf{X}$ en otro C–dimensional $\mathbf{Y} = (y_1, \cdots, y_C)^T = (F_1(\mathbf{X}, \mathbf{W}), \cdots, F_C(\mathbf{X}, \mathbf{W}))^T$, en donde la capa de salida de la red tiene una unidad por cada una de las clases.

Richard y Lippmann [Richard y Lippmann, 1991] demostraron que tomando como etiquetas los vectores canónicos, $\mathbf{t} \equiv \mathbf{e}^{\Omega(\mathbf{X})}$ y considerando como función de error el error cuadrático $\mathcal{E}(\mathbf{W}) = E\left[\| \mathcal{F}(\mathbf{X}, \mathbf{W}) - \mathbf{e}^{\Omega(\mathbf{X})} \|^2\right]$, la salida del PMC como clasificador realiza una estimación de la probabilidad *a posteriori* de Bayes, $p(\Omega_c|\mathbf{X})$. Con $E[\cdot]$ queremos indicar el valor esperado y con el superíndice $\Omega(\mathbf{X})$ representamos la clase a la que pertenece el patrón $\mathbf{X}$.

Luego, el vector óptimo $\mathbf{W}^*$ se determina minimizando la función de error

$$\begin{aligned}
\mathcal{E}(\mathbf{W}) &= E\left[\sum_{c=1}^{C}\left(F_c(\mathbf{X}, \mathbf{W}) - e_c^{\Omega(\mathbf{X})}\right)^2\right] \\
&= \int \sum_{c=1}^{C}\left(F_c(\mathbf{X}, \mathbf{W}) - e_c^{\Omega(\mathbf{X})}\right)^2 p(\mathbf{X})\, d\mathbf{X} \\
&= \int \sum_{m=1}^{C}\left\{\sum_{c=1}^{C}\left(F_c(\mathbf{X}, \mathbf{W}) - e_c^{\Omega(\mathbf{X})}\right)^2\right\} p(\mathbf{X}, \Omega_m)\, d\mathbf{X}, \qquad (2.10)
\end{aligned}$$

donde $p(\mathbf{X}, \Omega_m)$ es la densidad de probabilidad conjunta y puede expresarse como

$p(\mathbf{X}, \Omega_m) = p(\Omega_m|\mathbf{X})\ p(\mathbf{X})$, que sustituyéndola en la expresión (2.10) se obtiene

$$\begin{aligned}\mathcal{E}(\mathbf{W}) &= \int \left\{ \sum_{m=1}^{C} \sum_{c=1}^{C} \left(F_c(\mathbf{X}, \mathbf{W}) - e_c^{\Omega(\mathbf{X})}\right)^2 p(\Omega_m|\mathbf{X}) \right\} p(\mathbf{X})\ d\mathbf{X} \\ &= E\left[\sum_{m=1}^{C} \sum_{c=1}^{C} \left(F_c(\mathbf{X}, \mathbf{W}) - e_c^{\Omega(\mathbf{X})}\right)^2 p(\Omega_m|\mathbf{X}) \right]. \end{aligned} \tag{2.11}$$

Expandiendo el cuadrado de la expresión (2.11) se obtiene

$$\mathcal{E}(\mathbf{W}) = E\left[\sum_{m=1}^{C} \sum_{c=1}^{C} \left\{ F_c(\mathbf{X}, \mathbf{W})^2 - 2\ F_c(\mathbf{X}, \mathbf{W})\ e_c^{\Omega(\mathbf{X})} + (e_c^{\Omega(\mathbf{X})})^2 \right\}\ p(\Omega_m|\mathbf{X}) \right]. \tag{2.12}$$

Dado que $\sum_{m=1}^{C} p(\Omega_m|\mathbf{X}) = 1$, la expresión (2.12) puede ser redistribuida como

$$\begin{aligned}\mathcal{E}(\mathbf{W}) &= E\left[\sum_{c=1}^{C} \left\{ F_c(\mathbf{X}, \mathbf{W})^2 - 2\ F_c(\mathbf{X}, \mathbf{W}) \sum_{m=1}^{C} e_c^{\Omega(\mathbf{X})} p(\Omega_m|\mathbf{X}) \right.\right. \\ &+ \left.\left. \sum_{m=1}^{C} (e_c^{\Omega(\mathbf{X})})^2 p(\Omega_m|\mathbf{X}) \right\} \right]. \end{aligned} \tag{2.13}$$

El término $\sum_{m=1}^{C} e_c^{\Omega(\mathbf{X})} p(\Omega_m|\mathbf{X})$ es la esperanza condicional de $e_c^{\Omega(\mathbf{X})}$ o lo que es lo mismo $E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right]$ y dado que $\mathbf{e}^{\Omega(\mathbf{X})}$ es un vector C–dimensional donde todos los elementos son cero excepto la posición c–ésima, entonces se cumple que $E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right] = \sum_{m=1}^{C} e_c^{\Omega(\mathbf{X})} p(\Omega_m|\mathbf{X}) = p(\Omega_c|\mathbf{X})$, o lo que es lo mismo $E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right]$ representa la probabilidad *a posteriori* de que un patrón $\mathbf{X}$ pertenezca a la clase Ω_c.

Del mismo modo, el término $\sum_{m=1}^{C} (e_c^{\Omega(\mathbf{X})})^2 p(\Omega_m|\mathbf{X})$, es la esperanza condicional de $(e_c^{\Omega(\mathbf{X})})^2$. Puesto que el escalar $(e_c^{\Omega(\mathbf{X})})^2$ toma los valores 1 ó 0 en función de que el patrón pertenezca o no a la clase Ω_c, entonces se cumple que $E\left[(e_c^{\Omega(\mathbf{X})})^2|\mathbf{X}\right] = E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right] = p(\Omega_c|\mathbf{X})$.

La función de error (2.13) expresada en función de las esperanzas condicionales de $e_c^{\Omega(\mathbf{X})}$ y $(e_c^{\Omega(\mathbf{X})})^2$ queda de la forma

$$\mathcal{E}(\mathbf{W}) = E\left[\sum_{c=1}^{C} \left\{ F_c(\mathbf{X}, \mathbf{W})^2 - 2\ F_c(\mathbf{X}, \mathbf{W}) E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right] + E\left[(e_c^{\Omega(\mathbf{X})})^2|\mathbf{X}\right] \right\} \right]. \tag{2.14}$$

Sumando y restando el término $\sum_{c=1}^{C} \left(E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right]\right)^2$ a la expresión (2.14) se

puede expresar el error $\mathcal{E}$ de una forma más intuitiva

$$\begin{aligned}
\mathcal{E}(\mathbf{W}) &= E\left[\sum_{c=1}^{C}\left\{F_c(\mathbf{X},\mathbf{W})^2 - 2\,F_c(\mathbf{X},\mathbf{W})E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right] + \left(E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right]\right)^2\right\}\right] \\
&+ E\left[\sum_{c=1}^{C}\left\{E\left[(e_c^{\Omega(\mathbf{X})})^2|\mathbf{X}\right] - \left(E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right]\right)^2\right\}\right] \\
&= E\left[\sum_{c=1}^{C}\left(F_c(\mathbf{X},\mathbf{W}) - E\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right]\right)^2\right] + E\left[\sum_{c=1}^{C} var\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right]\right] \\
&= E\left[\sum_{c=1}^{C}\left(F_c(\mathbf{X},\mathbf{W}) - p(\Omega_c|\mathbf{X})\right)^2\right] + E\left[\sum_{c=1}^{C} var\left[e_c^{\Omega(\mathbf{X})}|\mathbf{X}\right]\right], \qquad (2.15)
\end{aligned}$$

donde $var[\,\cdot\,]$ indica la varianza. En el segundo término de (2.15) se ha hecho uso de la igualdad $var\,[A] = E\,[A^2] - (E\,[A])^2$.

El segundo término de la expresión (2.15) es independiente de la salida de la red; entonces minimizar $\mathcal{E}$ es equivalente a minimizar sólo el primer término de la expresión (2.15), es decir, basta con minimizar la función $E\left[\sum_{c=1}^{C}\left(F_c(\mathbf{X},\mathbf{W}) - p(\Omega_c|\mathbf{X})\right)^2\right]$ para hallar el valor óptimo $\mathbf{W}^*$.

En definitiva, si utilizamos la codificación de las etiquetas de los patrones de entrenamiento correspondiente a los vectores canónicos $\mathbf{e}^{\Omega(\mathbf{X})}$, considerando la capa de salida del PMC con tantas unidades como clases $\mathbf{Y} = (y_1, \cdots, y_C)^T = (F_1(\mathbf{X},\mathbf{W}), \cdots, F_C(\mathbf{X},\mathbf{W}))^T$ y tomando como función de error a minimizar el error cuadrático medio entre la salida de la red y la etiqueta de identificación del patrón $\mathbf{e}^{\Omega(\mathbf{X})}$, cada una de las transformaciones $F_c(\mathbf{X},\mathbf{W}^*)$ proporciona la distancia cuadrática mínima a la probabilidad *a posteriori* $p(\Omega_c|\mathbf{X})$. Dado que el clasificador óptimo de Bayes se define según la expresión (2.9), esto es, según la regla de decisión

$$\delta_B(\mathbf{X}) = argmax_c\{p(\Omega_c|\mathbf{X})\}, \qquad (2.16)$$

entonces como $F_c(\mathbf{X},\mathbf{W}^*) \approx p(\Omega_c|\mathbf{X})$, parece evidente que un procedimiento natural de clasificación en un PMC como el estudiado en esta sección corresponderá a asignarle al patrón $\mathbf{X}$ la clase correspondiente a la componente máxima de $\mathcal{F}(\mathbf{X},\mathbf{W}^*) = (F_1(\mathbf{X},\mathbf{W}^*), \cdots, F_C(\mathbf{X},\mathbf{W}^*))$; es decir la regla de decisión es tan sencilla como

$$\delta(\mathbf{X}) = argmax_c\{F_c(\mathbf{X},\mathbf{W}^*)\}. \qquad (2.17)$$

2.5. Análisis Discriminante de Fisher y PMCs

En esta sección discutiremos la relación que existe entre discriminantes lineales del tipo de los estudiados en el capítulo 1 y los perceptrones multicapa no lineales cuando ambos métodos son utilizados en aplicaciones de clasificación de patrones.

Webb y Lowe en [Webb y Lowe, 1990] [Webb y Lowe, 1991] demostraron que existe una conexión directa entre el discriminante lineal de Fisher y un PMC no lineal. Contemporáneamente, [Gallinari et al., 1991] demuestran que para un PMC lineal que minimiza el error cuadrático medio entre la salida de la red y las etiquetas de los patrones de entrada, se cumple que el conjunto de los pesos de la capa interna que actua como cuello de botella (la capa con menor número de unidades) realiza un análisis discriminante de los datos; esto es se produce una proyección de los datos al subespacio elegido durante el entrenamiento del PMC, de tal forma que en el nuevo espacio se percibe un tendencia al reagrupamiento por clases. Los pesos de esta capa interna $\mathbf{W}_h$ maximizan el criterio de Fisher en la proyección

$$J(\mathbf{W}_h^T) = \frac{\left|\tilde{\mathbf{S}}_B^H\right|}{\left|\tilde{\mathbf{S}}_T^H\right|},$$

donde $\tilde{\mathbf{S}}_B^H$ y $\tilde{\mathbf{S}}_T^H$ son respectivamente las convencionales matrices de covarianza inter–clases y total definidas a la salida de la última capa oculta del PMC lineal.

Sin embargo, Webb y Lowe llegan a la conclusión que en un PMC no lineal utilizado como clasificador, y cumpliéndose las tres restricciones siguientes:

1) considerar sólo conexiones lineales en la capa de salida,

2) el número de unidades en la capa de salida coincide con las clases existentes en la muestra y

3) las etiquetas de los patrones de entrada son los vectores canónicos $\mathbf{e}^{\Omega(\mathbf{x})}$,

se tiene entonces, que los pesos óptimos del PMC son aquellos que maximizan la cantidad

$$J(\mathbf{W}) = Tr\left(\mathcal{S}_B^H(\mathbf{S}_T^H)^{-1}\right),$$

donde $Tr(\cdot)$ indica la traza de matrices y $\mathcal{S}_B^H$ es una variante de la matriz de covarianza inter–clases. El criterio a maximizar J es un criterio que cumple las condiciones impuestas por el discriminante lineal de Fisher para obtener un buen clasificador; remitimos al capítulo 1, sección 1.2.

Para verificar el argumento de Webb y Lowe vamos a construir un clasificador partiendo del entrenamiento de un PMC no lineal. Para ello, se necesitará un conjunto de entrenamiento formado por N patrones etiquetados con una etiqueta $\mathbf{t}_i$ $(i = 1, \cdots, N)$ que permite distinguir a cuál de las C clases pertenece el patrón i–ésimo; la dimensión del vector $\mathbf{t}_i$ será el número de clases C.

Además, la función de error a minimizar vendrá dada por el error cuadrático

medio

$$\begin{aligned}\mathcal{E}(\mathbf{W}) &= \frac{1}{2N}\sum_{i=1}^{N} \parallel \mathcal{F}(\mathbf{X}_i, \mathbf{W}) - \mathbf{t}_i \parallel^2 \\ &= \frac{1}{2N}\sum_{i=1}^{N}\sum_{k=1}^{C} \left(F_k(\mathbf{X}_i, \mathbf{W}) - t_{i,k}\right)^2 .\end{aligned}$$

Partiendo de los conocimientos de la sección 2.3, las salidas del perceptrón con conexiones lineales en la capa de salida pueden expresarse como

$$y_k = \sum_{j=0}^{n^H} w_{jk}^H o_j^H(\mathbf{X}, \tilde{\mathbf{W}}),$$

donde el trio n^H, o_j^H y w_{jk}^H son, respectivamente, el número de unidades en la última capa oculta, la salida de esas unidades y los pesos de conexión con la capa de salida. El vector $\tilde{\mathbf{W}}$ representa al resto de los pesos de la red, es decir todos los pesos, incluyendo los sesgos, que no sean los que conectan directamente con la capa de salida. La función de error a minimizar será

$$\mathcal{E}(\mathbf{W}) = \frac{1}{2N}\sum_{i=1}^{N}\sum_{k=1}^{C}\left(\sum_{j=0}^{n^H} w_{jk}^H o_j^H(\mathbf{X}_i, \tilde{\mathbf{W}}) - t_{i,k}\right)^2 \tag{2.18}$$

A continuación, buscaremos minimizar la función de error respecto de los pesos w_{jk}^H, manteniendo los pesos $\tilde{\mathbf{W}}$ fijos. Para ello, comenzaremos por calcular cuál es el valor del sesgo w_{0k}^H, que se obtendrá despejando w_{0k}^H de la siguiente igualdad

$$\frac{\partial \mathcal{E}}{\partial w_{0k}^H} = \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{j=1}^{n^H} w_{jk}^H o_j^H(\mathbf{X}_i, \tilde{\mathbf{W}}) + w_{0k}^H - t_{i,k}\right) = 0,$$

el resultado vendrá dado por

$$w_{0k}^H = \bar{t}_k - \sum_{j=1}^{n^H} w_{jk}^H \bar{o}_j^H \tag{2.19}$$

donde las cantidades promediadas $\bar{t}_k$ y $\bar{o}_j^H$ son respectivamente

$$\bar{t}_k = \frac{1}{N}\sum_{i=1}^{N} t_{i,k} \qquad \text{y} \qquad \bar{o}_j^H = \frac{1}{N}\sum_{i=1}^{N} o_j^H(\mathbf{X}_i, \tilde{\mathbf{W}}).$$

La expresión (2.19) nos indica que el papel de los sesgos es compensar la diferencia que existe entre el valor promedio de las etiquetas y la suma ponderada

de los promedios de las salidas de las unidades ocultas. Si sustituimos la expresión (2.19) en la función de error cuadrático medio (2.18), se obtiene

$$\mathcal{E}(\mathbf{W}) = \frac{1}{2N}\sum_{i=1}^{N}\sum_{k=1}^{C}\left(\sum_{j=1}^{n^H} w_{jk}^H \tilde{o}_j^H(\mathbf{X}_i, \tilde{\mathbf{W}}) - \tilde{t}_{i,k}\right)^2 \tag{2.20}$$

donde

$$\tilde{t}_{i,k} = t_{i,k} - \bar{t}_k \qquad \text{y} \qquad \tilde{o}_j^H(\mathbf{X}_i, \tilde{\mathbf{W}}) = o_j^H(\mathbf{X}_i, \tilde{\mathbf{W}}) - \bar{o}_j^H;$$

es importante fijarse que el subíndice j corresponde a las unidades ocultas excluyendo el término del sesgo.

Ahora, ya estamos en condiciones de minimizar la función de error (2.20) respecto de los pesos w_{jk}^H, obteniéndose

$$\frac{\partial \mathcal{E}}{\partial w_{jk}^H} = \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{l=1}^{n^H} w_{lk}^H \tilde{o}_l^H(\mathbf{X}_i, \tilde{\mathbf{W}}) - \tilde{t}_{i,k}\right)\tilde{o}_j^H(\mathbf{X}_i, \tilde{\mathbf{W}}) = 0,$$

que en notación matricial nos queda

$$\mathbf{O}^T\mathbf{O}\mathbf{W} - \mathbf{O}^T\mathbf{T} = 0, \tag{2.21}$$

donde los elementos de las respectivas matrices son $(\mathbf{W})_{jk} = w_{jk}^H$, $(\mathbf{T})_{ik} = \tilde{t}_{i,k}$ y $(\mathbf{O})_{ij} = \tilde{o}_j^H(\mathbf{X}_i, \tilde{\mathbf{W}})$, siendo los rangos de definición $(i = 1, \cdots, N)$, $(j = 1, \cdots, n^H)$ y $(k = 1, \cdots, C)$.

Despejando la matriz de pesos $\mathbf{W}$ de la expresión (2.21) resulta que

$$\mathbf{W} = \mathbf{O}^\dagger\mathbf{T} \tag{2.22}$$

donde $\mathbf{O}^\dagger$ es la pseudo–inversa de Moore–Penrose de la matriz $\mathbf{O}$, definida como $\mathbf{O}^\dagger = (\mathbf{O}^T\mathbf{O})^{-1}\mathbf{O}^T$. En cualquier caso, estamos asumiendo que la matriz $(\mathbf{O}^T\mathbf{O})$ es no singular. Una de las propiedades de las matrices pseudo–inversas es que se cumple $\mathbf{O}^\dagger\mathbf{O} = \mathbf{I}$, siendo $\mathbf{I}$ la matriz identidad; sin embargo $\mathbf{O}\mathbf{O}^\dagger \neq \mathbf{I}$.

Igualmente, la expresión (2.20) se puede transformar en notación matricial de la siguiente forma

$$\mathcal{E}(\mathbf{W}) = \frac{1}{2N} Tr\left((\mathbf{O}\mathbf{W} - \mathbf{T})(\mathbf{O}\mathbf{W} - \mathbf{T})^T\right)$$

sustituyendo en esta expresión el valor de $\mathbf{W}$ por el que se obtuvo en (2.22) se llega a la siguiente expresión

$$\mathcal{E}(\mathbf{W}) = \frac{1}{2N} Tr\left((\mathbf{O}\mathbf{O}^\dagger\mathbf{T} - \mathbf{T})(\mathbf{O}\mathbf{O}^\dagger\mathbf{T} - \mathbf{T})^T\right) \tag{2.23}$$

Aplicando las siguientes propiedades del operador matricial traza

$$\begin{aligned} Tr(\mathbf{A}_1\mathbf{A}_2\cdots\mathbf{A}_{p-1}\mathbf{A}_p) &= Tr(\mathbf{A}_p\mathbf{A}_1\cdots\mathbf{A}_{p-2}\mathbf{A}_{p-1}), \\ Tr(\mathbf{A}_1+\mathbf{A}_2+\cdots+\mathbf{A}_p) &= Tr(\mathbf{A}_1)+Tr(\mathbf{A}_2)+\cdots+Tr(\mathbf{A}_p) \\ Tr(\mathbf{A}) &= Tr(\mathbf{A}^T) \end{aligned}$$

y teniendo en cuenta la definición y propiedades de la pseudo–inversa de Moore–Penrose

$$\begin{aligned} \mathbf{O}^\dagger &= (\mathbf{O}^T\mathbf{O})^{-1}\mathbf{O}^T \\ \mathbf{O}^\dagger\mathbf{O} &= \mathbf{I} \\ (\mathbf{O}\mathbf{O}^\dagger)^T &= \mathbf{O}\mathbf{O}^\dagger \end{aligned}$$

la expansión de la expresión (2.23) se desarrollaría de la siguiente forma

$$\begin{aligned} \mathcal{E}(\mathbf{W}) &= \frac{1}{2N}\ Tr\left(\mathbf{O}\mathbf{O}^\dagger\mathbf{T}\mathbf{T}^T\mathbf{O}\mathbf{O}^\dagger - \mathbf{O}\mathbf{O}^\dagger\mathbf{T}\mathbf{T}^T - \mathbf{T}\mathbf{T}^T\mathbf{O}\mathbf{O}^\dagger + \mathbf{T}\mathbf{T}^T\right) \\ &= \frac{1}{2N}\ \left\{Tr\left(\mathbf{T}\mathbf{T}^T\right) - Tr\left(\mathbf{T}\mathbf{T}^T\mathbf{O}(\mathbf{O}^T\mathbf{O})^{-1}\mathbf{O}^T\right)\right\} \\ &= \frac{1}{2N}\ \left\{Tr\left(\mathbf{T}\mathbf{T}^T\right) - Tr\left(\mathbf{O}^T\mathbf{T}\mathbf{T}^T\mathbf{O}(\mathbf{O}^T\mathbf{O})^{-1}\right)\right\} \end{aligned}$$

La matriz $\mathbf{O}^T\mathbf{O}$ en realidad representa la matriz de covarianza en la salida de la última capa oculta con respecto a los patrones de entrada, es decir

$$\mathbf{S}_T = \frac{1}{N}\mathbf{O}^T\mathbf{O} = \frac{1}{N}\sum_{i=1}^{N}(o^H(\mathbf{X}_i,\tilde{\mathbf{W}}) - \bar{o}^H)(o^H(\mathbf{X}_i,\tilde{\mathbf{W}}) - \bar{o}^H)^T$$

De igual modo, $\mathbf{O}^T\mathbf{T}\mathbf{T}^T\mathbf{O}$ viene a representar una variación de la matriz de dispersión *inter*–clases a la salida de la última capa oculta que, en general, no coincide con la matriz de dispersión $\mathbf{S}_B$. A la nueva matriz la denominaremos por

$$\mathcal{S}_B = \mathbf{O}^T\mathbf{T}\mathbf{T}^T\mathbf{O}.$$

Con todo esto, la expresión final de la función de error cuadrático en notación matricial, cuando estamos considerando un perceptrón en el que las conexiones finales son lineales, viene dada por

$$\mathcal{E}(\mathbf{W}) = \frac{1}{2}\ \left\{Tr\left(\frac{\mathbf{T}^T\mathbf{T}}{N}\right) - Tr\left(\mathcal{S}_B\mathbf{S}_T^{-1}\right)\right\} \tag{2.24}$$

El primer término de (2.24) depende exclusivamente de las etiquetas de los patrones de entrenamiento, $(\mathbf{T})_{ik} = \tilde{t}_{i,k}$, luego este término no interviene en la búsqueda los vectores $\mathbf{W}$ que minimizan la función de error cuadrático medio. Por tanto, minimizar el error cuadrático medio respecto de los pesos $\mathbf{W}^H$ equivaldría a maximizar la función discriminante

$$J = Tr\left(\mathcal{S}_B\ \mathbf{S}_T^{-1}\right),$$

que como se puede observar tiene un gran parecido con uno de los posibles criterios a tomar dentro del análisis discriminante de Fisher ($J = Tr\left(\mathbf{S}_B\ \mathbf{S}_T^{-1}\right)$), tal y como se propuso en el capítulo 1 sección 1.3. En el caso de que la matriz $\mathbf{S}_T$ no fuera invertible, la sustituiríamos por su pseudo–inversa $\mathbf{S}_T^{\dagger}$; luego la expresión más general para el criterio a maximizar sería

$$J = Tr\left(\mathcal{S}_B\ \mathbf{S}_T^{\dagger}\right). \tag{2.25}$$

Vamos a ver ahora, la importancia de elegir unas buenas etiquetas $\mathbf{t}_i$, pues dicha elección modificará la definición de la matriz $\mathcal{S}_B$.

Eligiendo las etiquetas $\mathbf{t}_i$ como los vectores canónicos $\mathbf{e}^{\Omega(\mathbf{x}_i)}$, tal y como lo hizo Gallinari [Gallinari et al., 1991], la definición de $\mathcal{S}_B$ viene dada por

$$\mathcal{S}_B = \sum_{k=1}^{C} N_k^2 (\bar{\mathbf{o}}^{(k,H)} - \bar{\mathbf{o}}^{H})(\bar{\mathbf{o}}^{(k,H)} - \bar{\mathbf{o}}^{H})^T,$$

donde N_k es el número de patrones de la clase Ω_k y $\bar{\mathbf{o}}^{(k,H)}$ representa el vector de la media de la salida de la última capa oculta con el conjunto de patrones que pertenecen a la clase Ω_k.

Salvo el factor $(1/N)$, $\mathcal{S}_B$ difiere de la matriz $\mathbf{S}_B$ convencional en que el factor N_k^2 es reemplazado por N_k, lo que provoca que las clases con mayor número de patrones sean las que pesen más, sesgando el discriminante a su favor. Esto puede ser un gran inconveniente, especialmente cuando las probabilidades a priori del conjunto de entrenamiento y el de test son sustancialmente distintas.

Con el fin de mantener la igualdad de oportunidades entre las clases, Webb y Lowe [Webb y Lowe, 1990] [Webb y Lowe, 1991] propusieron como vectores alternativos a los canónicos $\tilde{\mathbf{e}}^{\Omega(\mathbf{X})} = (0, \cdots, 1/\sqrt{N_c}, \cdots, 0)$ si $\mathbf{X}$ pertenece a la clase Ω_c. En este caso la definición de la matriz $\mathcal{S}_B$ viene dada por

$$\mathcal{S}_B = \sum_{k=1}^{C} N_k (\bar{\mathbf{o}}^{(k,H)} - \bar{\mathbf{o}}^{H})(\bar{\mathbf{o}}^{(k,H)} - \bar{\mathbf{o}}^{H})^T,$$

que salvo el factor $(1/N)$ coincide con la matriz $\mathbf{S}_B$ convencional. Luego, el uso de estas etiquetas alternativas evita que el discriminante esté sesgado hacia las clases con mayor número de patrones en el conjunto de entrenamiento.

Como conclusión del resultado obtenido, debemos destacar que eligiendo los pesos de la salida de la capa oculta como aquellos que minimizan el error cuadrático medio estamos forzando a que el conjunto de pesos de las capa anteriores $\tilde{\mathbf{W}}$, sean elegidos de tal forma que la transformación que va de la capa de entrada a la última capa oculta maximice el criterio discriminante de Fisher (2.25), cuando éste se efectúa sobre la salida de la última capa oculta.

Sin embargo, en ocasiones esta última codificación de las clases puede resultar incorrecta; obsérvese que siguiendo los pasos para la deducción de la expresión (2.15), se demostró que cada transformación $F_c(\mathbf{X}, \mathbf{W})$ se aproxima a su

correspondiente $E\left[\tilde{e}_c^{\Omega(\mathbf{X})}|\mathbf{X}\right]$, cuyo valor en este caso no coincide exactamente con la probabilidad *a posteriori*, $p(\Omega_c|\mathbf{X})$:

$$F_c(\mathbf{X}, \mathbf{W}) \approx E\left[\tilde{e}_c^{\Omega(\mathbf{X})}|\mathbf{X}\right] = \sum_{m=1}^{C} \tilde{e}_c^{\Omega(\mathbf{X})}\ p(\Omega_m|\mathbf{X}) = \frac{1}{\sqrt{N_c}}\ p(\Omega_c|\mathbf{X}).$$

Si consideramos la regla de decisión del PMC para clasificar patrones, expresión (2.17), que la recordaremos aquí

$$\delta(\mathbf{X}) = argmax_c\{F_c(\mathbf{X}, \mathbf{W}^*)\},$$

nos encontramos con la sorpresa de que este clasificador no aproxima la regla de decisión óptima de Bayes (2.16). Es más, se puede ver que puede ser muy susceptible a pequeñas variaciones en N_c. Dado que N_c/N es una estimación de la probabilidad $P(\Omega_c)$ de pertenencia *a priori* a la clase Ω_c, está claro que si dicha clase es reducida, variaciones aleatorias de N_c en las muestras a clasificar pueden dar lugar a estimaciones erróneas en la clasificación.

Con lo visto en esta sección queremos dejar constancia de la importancia de seleccionar correctamente el tipo de etiquetas para los patrones de entrenamiento. Así mismo, se ha visto que con determinadas etiquetas existe una relación entre los discriminantes de Fisher y los PMCs; será dicha relación la que aprovecharemos en el próximo capítulo para introducir la red que vamos a proponer como una mejora a los PMCs en determinadas condiciones extremas.

Capítulo 3

Análisis Discriminante No Lineal

3.1. Introducción

Los problemas de clasificación, como ya hemos visto en el capítulo anterior, son probablemente una de las áreas de mayor aplicación de los Perceptrones Multicapa (PMCs), con resultados excelentes en muchos casos. Entre las razones para ello están su versatilidad, su relativa facilidad de uso, su aplicabilidad en un rango amplio de problemas y su disponibilidad a través de múltiples implementaciones. Sin embargo, existen también puntos débiles en su utilización; ya indicamos en el capítulo 2, sección 2.4, la dificultad de traducir sus resultados en reglas de decisión. Igualmente, señalamos la dificultad de construir un PMC como clasificador cuando el tamaño de las clases muestrales implicadas es muy dispar y por añadidura cuando además existe un alto grado de solapamiento.

En este escenario, en cierto modo confuso, es en donde la efectividad de los PMCs tradicionales puede ser menor de lo que cabría esperar. Como solución alternativa a los PMCs proponemos la red que iremos viendo a lo largo de este capítulo; dicha red la denominaremos análisis discriminante no lineal (ADnL). Su estructura es muy similar a la de un PMC, pero con ciertas modificaciones en su configuración, lo que promueve que el nuevo prototipo de red presente características robustas frente a problemas de clasificación en los que existe una desigualdad numérica evidente de patrones en las distintas clases.

El capítulo queda estructurado de la siguiente forma: en primer lugar se presenta la estructura de la nueva red, comparándola con sus precesoras y se indica de dónde parte la idea de dicha red. A continuación, se introduce lo que será la base del aprendizaje de la red, que se irá desarrollando al detalle en la sección siguiente para varios criterios seleccionados y por último se discute cuál es su complejidad.

3.2. Análisis Discriminante no Lineal (ADnL)

La red que vamos a presentar en esta sección es en líneas generales un algoritmo no lineal supervisado de extracción de características; de ahí parte su acrónimo: Análisis Discriminante no Lineal(ADnL).

La red ADnL ha sido construida combinando los conocimientos previos del PMC y del discriminante lineal de Fisher; ambos métodos han sido presentados en detalle en los capítulos anteriores. La combinación de los dos métodos de extracción de características se realiza en base a la búsqueda de soluciones a los problemas que cada método tiene por separado y sin embargo, al fusionarlos se obtienen resultados que superan las expectativas individuales de cada método aislado.

La gran limitación del discriminante de Fisher se debe a que es sólo efectivo cuando se manipulan problemas con resolución lineal, bien como discriminante lineal o bien como herramienta para la extracción de características. Por el contrario, con el PMC se obtienen buenos resultados en problemas de resolución no lineal; pero como ya hemos indicado también se puede inhibir su potencia en casos como los señalados de desigualdad en el tamaño de las clases en problemas de discriminación, donde como ya hemos visto en la sección 2.5, la elección correcta de la etiqueta representativa de las clases es un factor a considerar.

Otra fuente de dificultades en la construcción de PMCs radica en que patrones con similares características pueden pertenecer a clases distintas, lo que se traduce en un valor alto en la medida de la probabilidad de error de Bayes. Con estas características, es difícil construir discriminantes que partiendo de los atributos originales de los patrones puedan resolver eficientemente el problema. Una posible solución y quizás la más obvia sería manipular las características de los patrones dentro de un preprocesamiento con el fin de intentar disminuir la zona de solapamiento entre clases en el nuevo espacio de estados.

La combinación de las dos obstáculos anteriores, clases muy desiguales y mezcla de patrones en problemas reales, en particular la detección de fraude en operaciones realizadas con tarjetas de crédito, es lo que llevo al grupo de investigación del Instituto de Ingeniería del Conocimiento (IIC) [Dorronsoro et al., 1997], [Santa Cruz y Dorronsoro, 1998] a plantear una nueva red en la que por un lado, no existiera la dependencia de la elección correcta de una etiqueta muestral; por otro lado la existencia de clases no equilibradas dejara de ser un problema de difícil superación y por último la red fuera capaz de discriminar, de una forma viable, la zona conflictiva de frontera entre clases. A modo de ejemplo y siguiendo con el problema de detección de fraude, localizar con un margen de error pequeño la sutil disimilitud que puede existir entre operaciones legales e ilegales.

La arquitectura de la red ADnL es similar a la de un PMC. Se trata de una red multicapa con al menos tres capas distribuidas de la misma forma que un PMC. La figura (3.1) representa un esquema sencillo de una red ADnL con una única

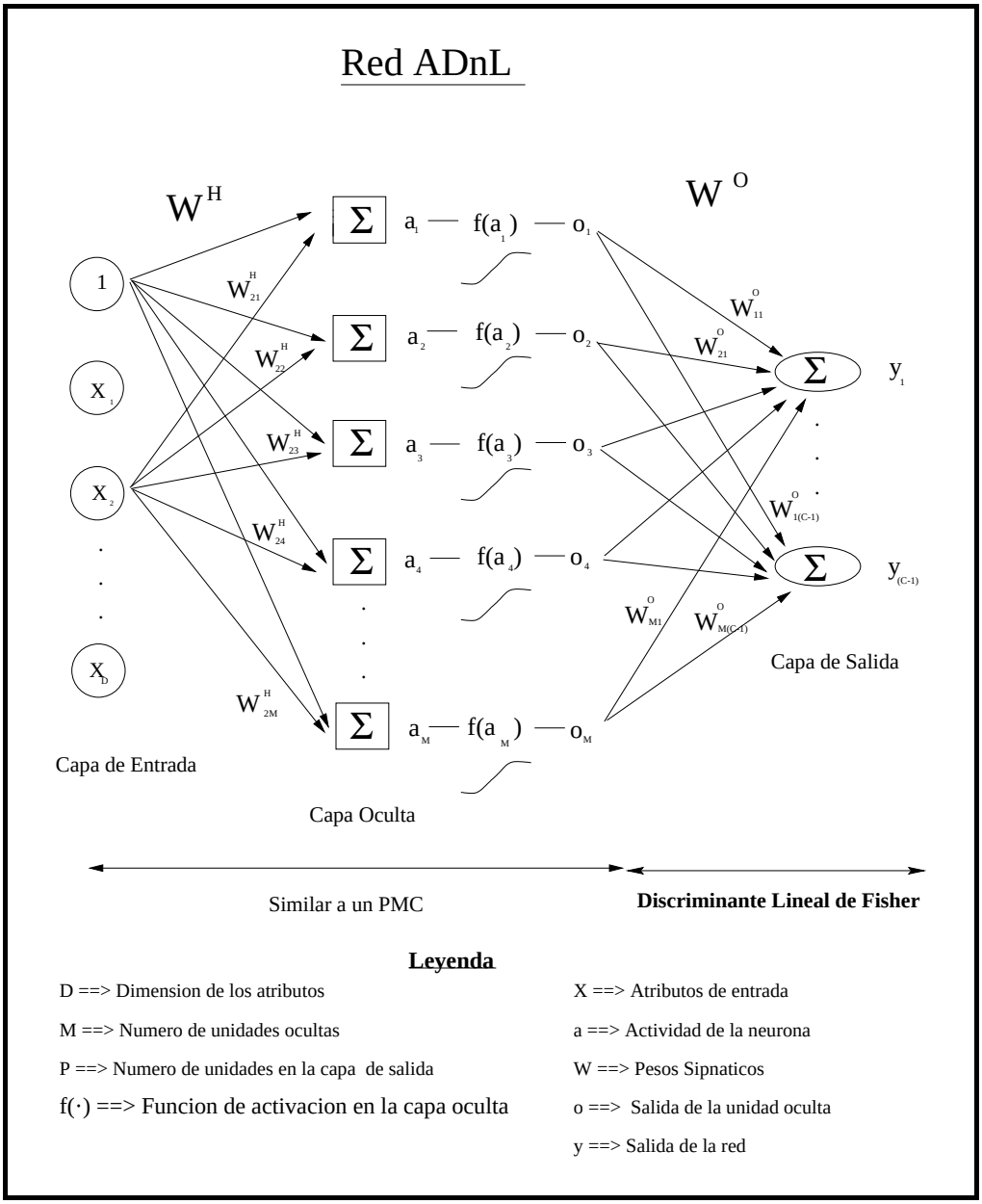


Figura 3.1: Esquema de la arquitectura de una red ADnL con una única capa oculta.

capa oculta; además se muestra la nomenclatura que se usará en la definición de la red.

Los pesos o conexiones de la red se van propagando de la capa de entrada a la capa de salida; se trata pues, de una red de flujo directo. Dependiendo de las dos capas que estén involucradas en una conexión, la propagación obedece a dos tipos distintos: aquellas conexiones que involucren como capa receptora una capa oculta son de tipo sigmoidal y en las conexiones cuya capa receptora sea la de salida, la conexión es lineal. Por tanto, en función del tipo de conexión, los pesos de la red se dividen en dos grupos $\mathbf{W} = (\boldsymbol{\mathcal{W}}, \mathbf{W}^O)$. Al primer grupo pertenecen los pesos con conexiones sigmoidales y los identificaremos por $\boldsymbol{\mathcal{W}} = (\mathbf{W}^1, \cdots, \mathbf{W}^H)$, donde H es el número de capas ocultas y al segundo grupo pertenecen los pesos asociados directamente con la capa de salida, $\mathbf{W}^O$.

Hasta este momento, la similitud con el PMC es total; la diferencia radica en que la función criterio utilizada en el entrenamiento de la red es distinta. En la red ADnL, tomamos como función criterio cualquiera de los criterios válidos para el análisis del discriminante lineal de Fisher. Aquí haremos hincapié en que estamos hablando de minimizar una función criterio tal y como se realiza en un PMC, mientras que en el capítulo 1 buscábamos maximizar un criterio de Fisher. Luego los criterios de Fisher expuestos en el capítulo 1 deben ser reconvertidos para utilizarlos en procesos de minimización con el fin de obtener los vectores $\mathbf{W}^*$ óptimos. La reconversión del criterio estudiado en el capítulo 1 es trivial:

$$J(\mathbf{W}) = \frac{\left|\tilde{\mathbf{S}}_W\right|}{\left|\tilde{\mathbf{S}}_B\right|} = \frac{\left|\mathbf{W}^T\mathbf{S}_W\mathbf{W}\right|}{\left|\mathbf{W}^T\mathbf{S}_B\mathbf{W}\right|}.$$

El entrenamiento de la red ADnL es iterativo, tal y como ocurre con el entrenamiento de un PMC; sin embargo otra disimilitud con el PMC es que en cada época se vislumbran dos pasos sustancialmente diferenciados. En primer lugar, los pesos $\mathbf{W}^O$ son hallados resolviendo el discriminante lineal de Fisher para la función criterio elegida, donde las entradas al discriminante corresponden a las salidas de cada patrón en la última capa oculta de la red ADnL, con lo cual estos pesos están predeterminados por lo que ocurra anteriormente en la red. Posteriormente, los pesos $\boldsymbol{\mathcal{W}}$ son hallados por retropropagación del criterio como función error de la red, buscando alcanzar aquellos pesos que minimizan dicho criterio.

Cuando la red ADnL es utilizada como clasificador, se hace palpable la existencia de dos claras restricciones en su topología. Al aplicar un discriminante lineal de Fisher a la salida de la última capa oculta, el número de unidades de la capa de salida está restringido a $(C-1)$, siendo C el número de clases existentes en la muestra. De igual modo, el número de unidades de la última capa oculta deberá ser igual o superior que el valor de $(C-1)$. Si el número de unidades de la última capa oculta fuera menor de $(C-1)$, el problema sería irresoluble: no es posible expandir a un espacio mayor que el de las entradas del discriminante. Luego, la topología de una red ADnL estará definida por una capa de entrada con

$(D+1)$ unidades: los D atributos del patrón de entrada más la unidad correspondiente al sesgo cuyo valor es siempre "1" y diversas capas ocultas, cada una de ellas con un número arbitrario de unidades, donde se incluye en cada capa la unidad correspondiente al sesgo. La última capa oculta será la excepción; como acabamos de ver el número de unidades de esta capa depende del número de clases, no pudiendo ser menor de $(C-1)$ y además no tiene unidad del sesgo; por último, en la capa de salida que también tiene una dependencia con el número de clases, tal y como hemos indicado, el número de unidades será estrictamente $(C-1)$.

Otra similitud con los PMCs es que la red ADnL utiliza un método de *aprendizaje supervisado* al igual que ocurre con el PMC. La diferencia entre ambos radica en que el PMC necesita que cada patrón esté identificado con la etiqueta de la clase a la que pertenece, de tal forma que podríamos evocar el símil entre la forma de aprender de un PMC y el modo de aprender de un alumno en el que su maestro está indicándole en cada paso si el ejercicio lo va realizando bien o está cometiendo errores. Mientras que ADnL precisa conocer exclusivamente cómo están distribuidos los patrones por clases,con el único propósito de determinar las correspondientes matrices de covarianza para la función criterio y, por tanto, no requiere definir vectores objetivo.

La idea de minimizar un análisis discriminante lineal como función criterio está relacionada con el desarrollo de Webb y Lowe en [Webb y Lowe, 1990] y [Webb y Lowe, 1991]. Recordaremos que ellos demostraron que existía una conexión directa entre el discriminante lineal de Fisher y un PMC. Probaron que para un PMC utilizado como clasificador, en el que las etiquetas elegidas para identificar las clases fueran bien los vectores canónicos $\mathbf{e}^{\Omega(\mathbf{X})}$ o bien una pequeña transformación de ellos, a los que llamamos vectores pseudo-canónicos $\tilde{\mathbf{e}}^{\Omega(\mathbf{X})}$ y restringiendo las conexiones con la capa de salida a conexiones lineales; entonces los pesos óptimos del PMC maximizarían la cantidad $J = Tr\left\{\mathcal{S}_B\ \mathbf{S}_T^{\dagger}\right\}$, donde $\mathcal{S}_B$ es una variante de la matriz de covarianza inter–clases. La definición exacta de esta matriz va a depender de la etiqueta elegida [Webb y Lowe, 1991], en concreto si las etiquetas de los patrones son los vectores canónicos y para muestras equidistribuidas $\mathcal{S}_B$ y $\mathbf{S}_B$ son proporcionales, esto es $\mathcal{S}_B = \pi \mathbf{S}_B$, donde π es la probabilidad a priori idéntica para todas las clases $\pi_1 = \cdots = \pi_C = \pi$, pues

$$\mathcal{S}_B = \sum_{c=1}^{C} \pi_c^2 (\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}})(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}})^T, \tag{3.1}$$

$$\mathbf{S}_B = \sum_{c=1}^{C} \pi_c (\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}})(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}})^T. \tag{3.2}$$

$\mathbf{S}_T$ es la convencional matriz de covarianza total; con el símbolo † queremos indicar la pseudo-inversa de Moore-Penrose. Las dos matrices $\mathcal{S}_B$ y $\mathbf{S}_T$ son medidas en la salida de la última capa oculta del PMC; para más detalles, remitimos al capítulo 2, sección 2.5 donde se explica con profundidad.

Al hilo de lo anterior, parece casi evidente que la red ejemplar para resolver problemas de clasificación sería una red similar al PMC, con conexión lineal en la capa de salida y que busque de un modo directo maximizar (o minimizar, dependiendo del criterio elegido) un buen criterio de Fisher; entonces, parece justificable la red ADnL, puesto que cumple todas las premisas anteriores. La interpretación sencilla de la red ADnL es que se trata de la combinación del análisis discriminante lineal de Fisher con una transformación previa no lineal de los vectores de características de los datos a clasificar.

A simple vista, podríamos pensar que un PMC y una red ADnL son construcciones similares; sin embargo, cuando se quiere adaptar un PMC para que actúe de la misma forma que el discriminante no lineal ADnL, se tiene que cumplir los siguientes requisitos:

1. **Arquitectura:** Para que un PMC actúe como discriminante es necesario que la capa de salida sea del mismo número de unidades como componentes tenga el vector objetivo. Si los vectores objetivo son los vectores canónicos, entonces el número de unidades en la capa de salida coincide con el número de clases existente en el conjunto muestral. Por otro lado, la red ADnL está sujeta a la ligadura intrínseca del número de unidades en la capa de salida a $(C-1)$.

   Con todo esto, desembocamos en que las dos últimas capas del PMC adaptado son lineales con $(C-1)$ y C unidades (en el caso de que los vectores objetivo sean los vectores canónicos). En modo esquemático, las unidades de las tres últimas capa de un PMC son $\cdots \otimes H \otimes C-1 \otimes C$, donde H representa el número de unidades de la última capa no lineal y se cumple que $H \geq C-1$; la fracción equivalente de una red ADnL contempla una capa lineal menos $\cdots \otimes H \otimes C-1$.

2. **Clasificador:** A la hora de clasificar un patrón, las dos redes deben usar el mismo clasificador. La convergencia para las dos es clasificar por las distancias a las medias de las clases proyectadas. Luego, si ADnL usa como características de la clasificación las correspondientes a la salida de ADnL, en total $(C-1)$ características, entonces con el PMC tenemos que tomar el mismo número de características y éstas se corresponden con la salida de la última capa oculta que tiene exactamente $(C-1)$ unidades.

   Nótese que ésta no es la forma de utilizar un PMC como clasificador. Lo habitual es clasificar el patrón por la proximidad de la salida del PMC a los vectores canónicos que son los identificadores de clase.

3. **Muestras equidistribuidas:** Cuando el conjunto de entrenamiento no está equidistribuido, jamás un PMC será similar a una red ADnL. En el caso de muestras con probabilidades a priori dispares, se tiene que los pesos óptimos del PMC que maximizan $J = Tr\left\{\mathcal{S}_B\ \mathbf{S}_T^{\dagger}\right\}$ no se corresponden con

ninguno de los criterios de ADnL; ello se debe a que la matriz $\mathcal{S}_B$ deja de ser proporcional a la matriz $\mathbf{S}_B$, ecuaciones (3.1) y (3.2).

En resumen, sólo cuando se cumplen los requisitos anteriores, un PMC y una red ADnL se puede decir que tienen comportamientos similares; en cualquier otro caso no hay equivalencia entre PMCs y redes ADnL.

3.3. Entrenamiento de una red ADnL

A diferencia de un PMC donde el entrenamiento de la red puede realizarse tanto en modo *batch* como en modo *on–line* (capítulo 2, sección 2.3.2), en la red ADnL sólo consideraremos entrenamiento en modo *batch*. Ello se debe a que la función criterio a minimizar requiere estadísticas de segundo orden; en el próximo punto veremos cuales son los criterios que hemos estudiado, pero adelantamos que todos ellos se basan en matrices de covarianza que, en definitiva, son modos de segundo orden.

Así, en cada época el conjunto completo de patrones de entrenamiento es presentado a la red antes de actualizarse los pesos. La actualización de los pesos en una época se realiza en dos etapas, encargándose cada etapa de uno de los dos tipos de pesos $\mathbf{W} = (\boldsymbol{\mathcal{W}}, \mathbf{W}^O)$, donde $\boldsymbol{\mathcal{W}} = (\mathbf{W}^1, \cdots, \mathbf{W}^H)$ son los pesos que involucran conexiones con unidades receptoras pertenecientes a capas ocultas (H indica el número de capas ocultas) y $\mathbf{W}^O$ los pesos de las conexiones con la capa de salida. Actualizar los pesos en dos tiempos se debe a que la función criterio global depende por separado de cada conjunto de pesos $\boldsymbol{\mathcal{W}}$ y $\mathbf{W}^O$.

Al ser el entrenamiento de la red ADnL un proceso iterativo, los nuevos pesos $\mathbf{W}_{t+1} = (\boldsymbol{\mathcal{W}}_{t+1}, \mathbf{W}^O_{t+1})$ se hallan a partir de los pesos computados en la época anterior $\mathbf{W}_t = (\boldsymbol{\mathcal{W}}_t, \mathbf{W}^O_t)$. Los pesos iniciales $\boldsymbol{\mathcal{W}}_0$ se obtienen a partir de números aleatorios generados según una distribución de probabilidad seleccionada. Los pasos a seguir en la actualización de pesos son los siguientes:

- En primer lugar, manteniendo fijos los pesos de las capas ocultas en la última época $\boldsymbol{\mathcal{W}}_t$, se calculan las correspondientes salidas de la última capa oculta. Los pesos asociados $\mathbf{W}^O_{t+1}$ se hallan resolviendo un análisis discriminante lineal múltiple efectuado sobre las salidas anteriormente halladas. La función criterio elegida para el discriminante cumple que $J^O(\mathbf{W}^O_{t+1}) = J(\boldsymbol{\mathcal{W}}_t, \mathbf{W}^O_{t+1})$, lo que muestra que será el conjunto de entrenamiento y el conjunto de pesos $\boldsymbol{\mathcal{W}}$ los que determinen los pesos de la capa de salida $\mathbf{W}^O$. Dado que las proyecciones de la última capa oculta sobre el subespacio definido por los pesos $\mathbf{W}^O_{t+1}$ son invariantes respecto a dilataciones del vector $\mathbf{W}^O_{t+1}$ entonces, por sencillez, los pesos finales son normalizados, cumpliéndose que $\| \mathbf{W}^O_{t+1} \| = 1$.

- La segunda etapa comienza tras obtener los pesos $\mathbf{W}^O_{t+1}$. Los nuevos pesos $\boldsymbol{\mathcal{W}}_{t+1}$ se resuelven minimizando numéricamente en modo *batch* iterativo la

función criterio $\mathcal{J}(\boldsymbol{\mathcal{W}}_{t+1}) = J(\boldsymbol{\mathcal{W}}_{t+1}, \mathbf{W}^O_{t+1})$, donde ahora los pesos $\mathbf{W}^O_{t+1}$ se consideran fijos. Son varios los métodos numéricos que se pueden elegir para hallar los nuevos pesos óptimos $\boldsymbol{\mathcal{W}}^*{}_{t+1}$; podríamos citar los métodos: descenso por gradiente, Quasi–Newton, gradiente conjugado y un sinfín más. La base de todos los métodos de minimización reside en el cálculo del gradiente de $\mathcal{J}$ respecto a los pesos $\boldsymbol{\mathcal{W}}$ y en el caso de considerar métodos de segundo orden sería necesario añadir el cálculo del correspondiente hessiano. En el próximo capítulo, se revisarán estos métodos de minimización.

3.4. Gradiente de J en ADnL

Ha llegado el momento de decidir cuál va a ser la función criterio a minimizar dentro de la red ADnL. A lo largo del estudio del criterio de Fisher se han barajado varias posibilidades como funciones criterio, pero todas ellas convergen en las dos estructuras siguientes

$$J(\mathbf{W}) = \phi(\mathbf{S}_2^{-1}\mathbf{S}_1) \qquad \text{ó} \qquad J(\mathbf{W}) = \phi(\mathbf{S}_1)/\phi(\mathbf{S}_2)$$

Las matrices $\mathbf{S}_1$ y $\mathbf{S}_2$ corresponden, normalmente, a las matrices de dispersión $\mathbf{S}_T$, $\mathbf{S}_B$ o $\mathbf{S}_W$ definidas por

$$\begin{aligned}
\mathbf{S}_T &= E[(\mathbf{Y} - \overline{\mathbf{Y}})(\mathbf{Y} - \overline{\mathbf{Y}})^T] \\
\mathbf{S}_B &= \sum_{c=1}^{C} \pi_c (\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}})(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}})^T \\
\mathbf{S}_W &= \sum_{c=1}^{C} \pi_c \mathbf{S}^{\Omega_c} \\
\mathbf{S}^{\Omega_c} &= E[(\mathbf{Y} - \overline{\mathbf{Y}}_c)(\mathbf{Y} - \overline{\mathbf{Y}}_c)^T | \mathbf{Y} \in \Omega_c]
\end{aligned}$$

donde $\overline{\mathbf{Y}} = E[\mathbf{Y}]$ y $\overline{\mathbf{Y}}_c = E[\mathbf{Y}|\Omega_c]$ representan respectivamente, la media total y la media condicional a la clase Ω_c en la salida de la red, con lo cual la dimensión de las tres matrices de dispersión $\mathbf{S}_T$, $\mathbf{S}_W$ y $\mathbf{S}_B$ es $(C-1)\times(C-1)$; por π_c se entiende la probabilidad a priori de la clase Ω_c. Además, ϕ es un operador de matrices que transforma una matriz en un escalar; dentro de los más utilizados se encuentran la traza y el determinante de matrices cuadradas. Ambos son operadores sencillos especialmente la traza; sin embargo aunque algo más costoso computacionalmente, el determinante de una matriz de dispersión nos da una medida de la dispersión de la muestra bastante más fiable que la que obtendríamos con el operador traza.

Posibles funciones criterio a utilizar en la red ADnL son:

$$\begin{aligned} J_1(\mathbf{W}) &= \frac{|\mathbf{S}_T|}{|\mathbf{S}_B|}\,, \\ J_2(\mathbf{W}) &= \frac{Tr(\tilde{\mathbf{\Lambda}}\,\mathbf{S}_T)}{Tr(\mathbf{S}_B)}\,, \\ J_3(\mathbf{W}) &= \frac{1}{Tr(\mathbf{S}_T^{-1}\mathbf{S}_B)}\,. \end{aligned}$$

En el apéndice A se explica como al maximizar el criterio $J = Tr(\mathbf{S}_B)/Tr(\mathbf{S}_W)$ para obtener los vectores que definen el subespacio de proyección en el discriminante de Fisher, se llega a un sistema degenerado que interesa evitar si se quiere obtener una solución completa. Para obtener tal solución, es preciso realizar una pequeña transformación del criterio. Con el criterio J_2 ocurre exactamente lo mismo: si minimizamos el criterio simplificado $J = Tr(\mathbf{S}_T)/Tr(\mathbf{S}_B)$ para obtener el conjunto de vectores $\mathbf{W}$ óptimos se llega a un sistema degenerado, luego tendremos que realizar la misma transformación que en el apéndice A para obtener el criterio J_2 anterior. Donde la matriz $\tilde{\mathbf{\Lambda}}$ es una matriz diagonal $\tilde{\mathbf{\Lambda}} = \mathbf{\Lambda}/\lambda_1$, siendo $\mathbf{\Lambda}$ la matriz real de autovalores de $\mathbf{S}_T^{-1}\mathbf{S}_B$ y λ_1 es el mayor de los autovalores $(\lambda_1 > \lambda_2 > \cdots > \lambda_{C-1})$; de este modo el elemento $\tilde{\lambda}_1$ de la matriz $\tilde{\mathbf{\Lambda}}$ es la unidad y el resto de los elementos de $\tilde{\mathbf{\Lambda}}$ son positivos y menores de la unidad.

El último criterio, J_3, se puede considerar como una transformación lineal del criterio simplificado $J = Tr(\mathbf{S}_T)/Tr(\mathbf{S}_B)$ intentando superar la carencia de este último. La transformación a J_3 se lleva a cabo en los siguientes pasos: en primer lugar es necesario transformar la matriz $\mathbf{S}_T$ en la matriz identidad $\mathbf{I}$, objetivo que puede ser alcanzado fácilmente con la transformación $\mathbf{U}^T\mathbf{S}_T\mathbf{U}$, de tal forma que $\mathbf{U} = \mathbf{S}_T^{-1/2}$. A continuación, aplicamos esta transformación lineal a los dos términos de J_2 obteniéndose finalmente el criterio J_3

$$\frac{Tr(\mathbf{U}^T\mathbf{S}_T\mathbf{U})}{Tr(\mathbf{U}^T\mathbf{S}_B\mathbf{U})} = \frac{1}{Tr(\mathbf{U}\mathbf{U}^T\mathbf{S}_B)} = \frac{1}{Tr(\mathbf{S}_T^{-1}\mathbf{S}_B)} = J_3.$$

Aunque los tres criterios parecen esencialmente distintos, la convergencia teórica al mínimo conduce al mismo conjunto de vectores $\mathbf{W}$ óptimos. Sin embargo, cuando buscamos numéricamente el vector óptimo, es el primer criterio, J_1, el que acumula mayor información intrínseca, lo que permite evolucionar hacia resultados más fiables. Con el criterio J_2, sí se tiene en cuenta la variabilidad existente intra–clases pero, al considerar el operador traza de matrices, se ignora el efecto de la separación real de las clases debido a la elusión de la correlación entre los distintos atributos de los patrones, lo que desemboca en una menor representación del sistema en el valor del escalar J_2.

Seguidamente, iremos viendo cómo se obtienen los gradientes de los criterios anteriores dentro de la red ADnL.

3.4.1. Función criterio J_1: Razón de determinantes

En esta sección, hallaremos el gradiente del criterio $J_1(\mathbf{W}) = |\mathbf{S}_T|/|\mathbf{S}_B|$. Comenzaremos nuestro propósito calculando $\partial J_1/\partial W_{kl}$ para un peso genérico W_{kl}:

$$\begin{aligned}\frac{\partial J_1}{\partial W_{kl}} &= \frac{1}{|\mathbf{S}_B|^2}\left(\frac{\partial|\mathbf{S}_T|}{\partial W_{kl}}|\mathbf{S}_B| - |\mathbf{S}_T|\frac{\partial|\mathbf{S}_B|}{\partial W_{kl}}\right)\\ &= \frac{1}{|\mathbf{S}_B|}\left(\frac{\partial|\mathbf{S}_T|}{\partial W_{kl}} - J_1\frac{\partial|\mathbf{S}_B|}{\partial W_{kl}}\right). \end{aligned} \tag{3.3}$$

En el apéndice B se demuestra que si $\mathbf{S}(\mathbf{W})$ es una matriz simétrica, la derivada de su determinante respecto del peso W_{kl} viene dado por

$$\frac{\partial|\mathbf{S}(\mathbf{W})|}{\partial W_{kl}} = |\mathbf{S}(\mathbf{W})|\ Tr\left(\mathbf{S}(\mathbf{W})^{-1}\frac{\partial\mathbf{S}(\mathbf{W})}{\partial W_{kl}}\right). \tag{3.4}$$

Dado que las matrices $\mathbf{S}_T$ y $\mathbf{S}_B$ son matrices simétricas, es posible sustituir la expresión (3.4) en (3.3) para las dos matrices $\mathbf{S}_T$ y $\mathbf{S}_B$, obteniéndose con ello

$$\begin{aligned}\frac{\partial J_1}{\partial W_{kl}} &= \frac{1}{|\mathbf{S}_B|}\left\{|\mathbf{S}_T|\ Tr\left(\mathbf{S}_T^{-1}\frac{\partial\mathbf{S}_T}{\partial W_{kl}}\right) - J_1\ |\mathbf{S}_B|\ Tr\left(\mathbf{S}_B^{-1}\frac{\partial\mathbf{S}_B}{\partial W_{kl}}\right)\right\}\\ &= J_1\ Tr\left(\mathbf{S}_T^{-1}\frac{\partial\mathbf{S}_T}{\partial W_{kl}} - \mathbf{S}_B^{-1}\frac{\partial\mathbf{S}_B}{\partial W_{kl}}\right). \end{aligned} \tag{3.5}$$

A partir de las expresiones de $\mathbf{S}_T$ y $\mathbf{S}_B$ se obtiene que sus derivadas son

$$\begin{aligned}\frac{\partial\mathbf{S}_T}{\partial W_{kl}} &= E\left[\left(\frac{\partial\mathbf{Y}}{\partial W_{kl}} - \frac{\partial\overline{\mathbf{Y}}}{\partial W_{kl}}\right)\left(\mathbf{Y}-\overline{\mathbf{Y}}\right)^T + \left(\mathbf{Y}-\overline{\mathbf{Y}}\right)\left(\frac{\partial\mathbf{Y}}{\partial W_{kl}} - \frac{\partial\overline{\mathbf{Y}}}{\partial W_{kl}}\right)^T\right]\\ &= E\left[\frac{\partial\mathbf{Y}}{\partial W_{kl}}\left(\mathbf{Y}-\overline{\mathbf{Y}}\right)^T + \left(\mathbf{Y}-\overline{\mathbf{Y}}\right)\left(\frac{\partial\mathbf{Y}}{\partial W_{kl}}\right)^T\right] = E\left[\mathbf{D}_{kl} + \mathbf{D}_{kl}^T\right] \end{aligned} \tag{3.6}$$

$$\begin{aligned}\frac{\partial\mathbf{S}_B}{\partial W_{kl}} &= \sum_{c=1}^{C}\pi_c\left(\left(\frac{\partial\overline{\mathbf{Y}}_c}{\partial W_{kl}} - \frac{\partial\overline{\mathbf{Y}}}{\partial W_{kl}}\right)\left(\overline{\mathbf{Y}}_c-\overline{\mathbf{Y}}\right)^T + \left(\overline{\mathbf{Y}}_c-\overline{\mathbf{Y}}\right)\left(\frac{\partial\overline{\mathbf{Y}}_c}{\partial W_{kl}} - \frac{\partial\overline{\mathbf{Y}}}{\partial W_{kl}}\right)^T\right)\\ &= \sum_{c=1}^{C}\pi_c\left(\frac{\partial\overline{\mathbf{Y}}_c}{\partial W_{kl}}\left(\overline{\mathbf{Y}}_c-\overline{\mathbf{Y}}\right)^T + \left(\overline{\mathbf{Y}}_c-\overline{\mathbf{Y}}\right)\left(\frac{\partial\overline{\mathbf{Y}}_c}{\partial W_{kl}}\right)^T\right)\\ &= \sum_{c=1}^{C}\pi_c\left(\mathbf{D}_{kl}^{\Omega_c} + (\mathbf{D}_{kl}^{\Omega_c})^T\right), \end{aligned} \tag{3.7}$$

donde para simplificar las expresiones finales de (3.6) y (3.7) hemos definido las matrices $\mathbf{D}_{kl}$ y $\mathbf{D}_{kl}^{\Omega_c}$ de la siguiente forma

$$\begin{aligned}\mathbf{D}_{kl} &= \frac{\partial\mathbf{Y}}{\partial W_{kl}}\left(\mathbf{Y}-\overline{\mathbf{Y}}\right)^T,\\ \mathbf{D}_{kl}^{\Omega_c} &= \frac{\partial\overline{\mathbf{Y}}_c}{\partial W_{kl}}\left(\overline{\mathbf{Y}}_c-\overline{\mathbf{Y}}\right)^T. \end{aligned}$$

Sustituyendo las expresiones (3.6) y (3.7) en la ecuación (3.5), se obtiene la siguiente expresión para el gradiente de J_1

$$\frac{\partial J_1}{\partial W_{kl}} = J_1 \ Tr\left(\mathbf{S}_T^{-1} E\left[\mathbf{D}_{kl} + \mathbf{D}_{kl}^T\right] - \mathbf{S}_B^{-1} \sum_{c=1}^{C} \pi_c \left(\mathbf{D}_{kl}^{\Omega_c} + (\mathbf{D}_{kl}^{\Omega_c})^T\right)\right). \qquad (3.8)$$

De la expresión anterior se sigue que el cálculo de las derivadas $\partial \mathbf{Y}/\partial W_{kl}$, $\partial \overline{\mathbf{Y}}/\partial W_{kl}$ y $\partial \overline{\mathbf{Y}}_c/\partial W_{kl}$ será el siguiente paso necesario para deducir la expresión final del gradiente de J_1. Luego, será indispensable calcular cada una de las derivadas $\partial y_p/\partial W_{kl}$, $\partial \overline{y}_p/\partial W_{kl}$ y $\partial \overline{y}_p^{\Omega_c}/\partial W_{kl}$, donde $p = (1, \cdots, C-1)$ y $\Omega_c = (\Omega_1, \cdots, \Omega_C)$.

En primer lugar, afrontaremos el cálculo de las derivadas $\partial y_p/\partial W_{kl}$; para ello, recordaremos las expresiones de la activación y salida de una unidad oculta en una red ADnL, $a_l = \sum_k W_{kl}^h o_k$ y $o_l = f(a_l)$, donde f es la función de activación. Aplicando la regla de la cadena en derivación, $\partial y_p/\partial W_{kl}$ puede descomponerse en

$$\frac{\partial y_p}{\partial W_{kl}} = \frac{\partial y_p}{\partial o_l}\frac{\partial o_l}{\partial a_l}\frac{\partial a_l}{\partial W_{kl}} = \frac{\partial y_p}{\partial o_l} f'(a_l) o_k.$$

Si la unidad l pertenece a la última capa oculta, estamos en el caso más sencillo de derivar: $\partial y_p/\partial o_l = W_{lp}^O$, puesto que $y_p = \sum_{q=1}^{n^H} W_{qp}^O o_q$. Pero, si la unidad l pertenece a cualquier otra capa oculta, para calcular la derivada $\partial y_p/\partial o_l$ será necesario aplicar, una vez más, la regla de la cadena involucrando capas superiores a la capa de la unidad l :

$$\frac{\partial y_p}{\partial o_l} = \sum_m \frac{\partial y_p}{\partial o_m}\frac{\partial o_m}{\partial a_m}\frac{\partial a_m}{\partial o_l} = \sum_m \frac{\partial y_p}{\partial o_m} f'(a_m) W_{lm},$$

donde W_{lm} es el peso que conecta la unidad l con la unidad m de la capa siguiente. Como puede apreciarse estamos frente a un método de retropropagación, ya que $\partial y_p/\partial o_l$ en una capa dada puede calcularse con las componentes $\partial y_p/\partial o_m$ de las capas superiores ($m = l+1, \cdots, H$).

Dado que $\overline{y}_p = E[y_p]$, su derivada será

$$\frac{\partial \overline{y}_p}{\partial W_{kl}} = E\left[\frac{\partial y_p}{\partial W_{kl}}\right] = E\left[\frac{\partial y_p}{\partial o_l} f'(a_l) o_k\right],$$

y lo mismo ocurre con la esperanza condicional, $\overline{y}_p^{\Omega_c} = E\left[y_p | \mathbf{Y} \in \Omega_c\right]$, de donde se deduce que su derivada será

$$\frac{\partial \overline{y}_p^{\Omega_c}}{\partial W_{kl}} = E\left[\frac{\partial y_p}{\partial W_{kl}} | \mathbf{Y} \in \Omega_c\right] = E\left[\frac{\partial y_p}{\partial o_l} f'(a_l) o_k | \mathbf{Y} \in \Omega_c\right].$$

Con todo esto, ya estamos en condiciones de hallar el gradiente de J_1 expresado según la ecuación (3.8).

3.4.2. Función criterio J_2: Razón de trazas

La expresión del gradiente del criterio J_2 para un peso genérico W_{kl} viene dada por

$$\begin{aligned}\frac{\partial J_2}{\partial W_{kl}} &= \frac{1}{Tr(\mathbf{S}_B)}\left\{\frac{\partial Tr(\tilde{\mathbf{\Lambda}}\mathbf{S}_T)}{\partial W_{kl}} - J_2\,\frac{\partial Tr(\mathbf{S}_B)}{\partial W_{kl}}\right\} \\ &= \frac{1}{Tr(\mathbf{S}_B)}\left\{Tr\left(\tilde{\mathbf{\Lambda}}\frac{\partial \mathbf{S}_T}{\partial W_{kl}}\right) - J_2\,Tr\left(\frac{\partial \mathbf{S}_B}{\partial W_{kl}}\right)\right\} \qquad (3.9) \\ &= \frac{1}{Tr(\mathbf{S}_B)}\left\{Tr\left(\tilde{\mathbf{\Lambda}}\;E\left[\mathbf{D}_{kl}+\mathbf{D}_{kl}^T\right]\right) - J_2\,Tr\left(\sum_{c=1}^{C}\pi_c\left(\mathbf{D}_{kl}^{\Omega_c}+(\mathbf{D}_{kl}^{\Omega_c})^T\right)\right)\right\} \qquad (3.10)\end{aligned}$$

Dado que $\tilde{\mathbf{\Lambda}}$ es una matriz diagonal, entonces sólo intervendrán los elementos de la diagonal de la matriz $\partial \mathbf{S}_T/\partial W_{kl}$ y de modo similar nos ocurre con la derivada $\partial \mathbf{S}_B/\partial W_{kl}$, pues al tomar la traza sólo intervienen los términos de la diagonal. Desarrollando los términos diagonales de las dos derivadas, la expresión anterior nos queda de la siguiente forma

$$\begin{aligned}\frac{\partial J_2}{\partial W_{kl}} &= \frac{2}{Tr(\mathbf{S}_B)}\left\{\frac{1}{N}\sum_{p=1}^{C-1}\tilde{\lambda}_p\sum_{i=1}^{N}\frac{\partial y_{ip}}{\partial W_{kl}}\left(y_{ip}-\overline{y}_p\right)\right. \\ &- \left. J_2\sum_{p=1}^{C-1}\sum_{c=1}^{C}\pi_c\,\frac{\partial \overline{y}_p^{\Omega_c}}{\partial W_{kl}}\,\left(\overline{y}_p^{\Omega_c}-\overline{y}_p\right)\right\}, \qquad (3.11)\end{aligned}$$

donde N es el número de patrones de entrenamiento y C el número de clases en la muestra. El resto del cálculo del criterio J_2 es seguir los pasos desarrollados en la sección 3.4.1 para el cálculo de las derivadas $\partial y_{ip}/\partial W_{kl}$ y $\partial \overline{y}_p^{\Omega_c}/\partial W_{kl}$, para los valores de los índices $i=(1,\cdots,N)$, $p=(1,\cdots,C-1)$ y $c=(1,\cdots,C)$. De este modo, la expresión final para el gradiente de J_2 es la siguiente

$$\begin{aligned}\frac{\partial J_2}{\partial W_{kl}} &= \frac{2}{Tr(\mathbf{S}_B)}\left\{\sum_{p=1}^{C-1}\tilde{\lambda}_p\left(E\left[y_p\frac{\partial y_p}{\partial o_l}f'(a_l)o_k\right] - \overline{y}_p E\left[\frac{\partial y_p}{\partial o_l}f'(a_l)o_k\right]\right)\right. \\ &- \left. J_2\sum_{p=1}^{C-1}\left(\left(\sum_{c=1}^{C}\pi_c\overline{y}_p^{\Omega_c}E_c\left[\frac{\partial y_p}{\partial o_l}f'(a_l)o_k\right]\right) - \overline{y}_p E\left[\frac{\partial y_p}{\partial o_l}f'(a_l)o_k\right]\right)\right\},\end{aligned}$$

que aglutinando en nuevas variable, la expresión anterior nos queda de la siguiente forma

$$\frac{\partial J_2}{\partial W_{kl}} = \frac{2}{Tr(\mathbf{S}_B)}\sum_{p=1}^{C-1}\left(\tilde{\lambda}_p E\left[A_{kl;p}\right] - \tilde{\Lambda}_p\,E\left[B_{kl;p}\right] - J_2\sum_{c=1}^{C}\pi_c\,\overline{y}_p^{\Omega_c}\,E_c\left[B_{kl;p}\right]\right), \qquad (3.12)$$

donde los valores de las nuevas variables $\tilde{\Lambda}_p$, $A_{kl;p}$ y $B_{kl;p}$ son los siguientes

$$\begin{aligned} \tilde{\Lambda}_p &= \overline{y}_p \, (\tilde{\lambda}_p - J_2) \\ A_{kl;p} &= y_p \, \frac{\partial y_p}{\partial o_l} \, f'(a_l) \, o_k \\ B_{kl;p} &= \frac{\partial y_p}{\partial o_l} \, f'(a_l) \, o_k. \end{aligned}$$

En todos los casos, $E[\,\cdot\,]$ es el operador esperanza y $E_c[\,\cdot\,]$ la esperanza condicionada a la clase Ω_c.

El criterio J_2 es el que permite obtener la fórmula más cerrada del gradiente. En el gradiente del criterio J_1 intervienen operadores determinantes que impiden simplificar la fórmula de la misma forma que con J_2, donde el operador de matrices involucrado en el gradiente es la traza, escalar mucho más sencillo de calcular que el determinante.

3.4.3. Función criterio J_3

Si consideramos como función criterio $J_3 = 1/Tr(\mathbf{S}_T^{-1}\mathbf{S}_B)$, el gradiente de J_3 puede hallarse de forma similar a los dos anteriores

$$\frac{\partial J_3}{\partial W_{kl}} = -\ {J_3}^2 \ \frac{\partial Tr(\mathbf{S}_T^{-1}\mathbf{S}_B)}{\partial W_{kl}}.$$

Nos centraremos en el cálculo de $\partial Tr(\mathbf{S}_T^{-1}\mathbf{S}_B)/\partial W_{kl} = Tr(\partial(\mathbf{S}_T^{-1}\mathbf{S}_B)/\partial W_{kl})$, donde se aprecia la necesidad de calcular la derivada de una matriz inversa; pero esto no es un desafio demasiado grande puesto que la derivada de la inversa de una matriz simétrica viene dada por

$$\frac{\partial \mathbf{S}(\mathbf{W})^{-1}}{\partial W_{kl}} = -\mathbf{S}(\mathbf{W})^{-1} \frac{\partial \mathbf{S}(\mathbf{W})}{\partial W_{kl}} \mathbf{S}(\mathbf{W})^{-1}. \tag{3.13}$$

Con todo ello, el gradiente de J_3 viene dado por

$$\frac{\partial J_3}{\partial W_{kl}} = {J_3}^2 \ Tr\left(\mathbf{S}_T^{-1} \frac{\partial \mathbf{S}_T}{\partial W_{kl}} \mathbf{S}_T^{-1} \mathbf{S}_B \ - \ \mathbf{S}_T^{-1} \frac{\partial \mathbf{S}_B}{\partial W_{kl}} \right). \tag{3.14}$$

Los pasos a seguir a partir de este momento son completamente similares a los realizados para el gradiente de J_1, es decir, sustituir las expresiones de $\partial \mathbf{S}_T/\partial W_{kl}$ y $\partial \mathbf{S}_B/\partial W_{kl}$ en (3.14) y a continuación calcular las distintas derivadas $\partial \mathbf{Y}/\partial W_{kl}$, $\partial \overline{\mathbf{Y}}/\partial W_{kl}$ y $\partial \overline{\mathbf{Y}}_c/\partial W_{kl}$, tal y como se expuso en la sección 3.4.1.

De los tres criterios que se han propuesto en esta sección, sin duda el menos complicado es el criterio J_2, pues con él no es necesario ni el cálculo de las inversas de las matrices $\mathbf{S}_T$ y $\mathbf{S}_B$, ni el cálculo de las matrices totales de las derivadas de éstas, así como tampoco es necesario realizar los correspondientes productos de

matrices. Los otros dos criterios J_1 y J_3 son similares en complejidad, aunque algo más complejo es el criterio J_1, ya que necesita la inversa de la matriz $\mathbf{S}_B$, que no es necesaria en el criterio J_3 y por otro lado, calcular el valor de J_1 que conlleva una pequeña sobrecarga al tener que calcular los determinantes de $\mathbf{S}_T$ y $\mathbf{S}_B$. En la próxima sección veremos la complejidad intrínseca del cálculo de los gradientes de los tres criterios revisados en esta sección.

3.5. Complejidad Computacional en la red ADnL

En esta sección vamos a estudiar la complejidad de la red ADnL para una época. Por supuesto, la sobrecarga total dependerá del número de épocas que necesita la red antes de llegar a la convergencia.

Para facilitar el cálculo de la complejidad de la red ADnL, segmentaremos cada época en tres fragmentos. Partiendo del conjunto de pesos de la época anterior $\mathbf{W} = (\boldsymbol{\mathcal{W}}, \mathbf{W}^O)$ o bien si se trata de la primera época, de los pesos iniciales $\mathbf{W}_0$, revisaremos en primer lugar el coste que conlleva calcular las salidas de la última capa oculta a partir de los pesos $\boldsymbol{\mathcal{W}}$ de la época anterior. A continuación analizaremos el coste del segundo fragmento, es decir, desde la salida de la última capa oculta hasta la salida final de la red, lo que implica realizar un análisis discriminante de Fisher con la salida de la última capa oculta obteniéndose de este modo los pesos $\mathbf{W}^O$. Por último calcularemos el coste de renovar los pesos $\boldsymbol{\mathcal{W}}$ para la siguiente época, lo que conlleva calcular el gradiente del criterio elegido J respecto de los pesos de las capas ocultas. El coste computacional total por época será esencialmente la suma de los tres fragmentos anteriores.

Si el número de capas ocultas en la red es H y en cada capa oculta el número de unidades, sin contar con el sesgo, es $n_h (h = 0, \cdots, H)$, donde n_0 es la dimensión D de los atributos de los patrones que entran en la red, el cálculo de todas las activaciones $a = \mathbf{W}^T \mathbf{o}$ correspondientes a cada unidad a lo largo de las capas ocultas supone un coste por cada patrón de entrenamiento de

$$\mathcal{O}\left(\sum_{h=0}^{H-1}(n_h+1)n_{h+1}\right).$$

Ahora, a cada activación hay que aplicarle la correspondiente función de activación para calcular la salida de la unidad oculta, $o = f(a)$. Suponiendo que el coste de la función de activación es una constante $Q = \mathcal{O}(1)$, entonces para cada patrón el coste de calcular todas las funciones de activación viene dado por

$$\mathcal{O}\left(\sum_{h=1}^{H} n_h\, Q\right) = \mathcal{O}\left(\sum_{h=1}^{H} n_h\right).$$

Luego, el coste hasta la salida de la última capa oculta es la suma de las dos contribuciones anteriores. Si consideramos todos los patrones de entrenamiento el coste es el siguiente

$$\mathcal{O}\left(N\,(\sum_{h=0}^{H-1} n_{h+1}(n_h+Q'))\right),$$

donde $Q' = Q+1 = \mathcal{O}(1)$. El coste de hallar la salida de la red en una etapa es por tanto $\mathcal{O}\left(N\ \sum_{h=0}^{H-1} n_{h+1}n_h\right)$.

Comenzaremos ahora con el cálculo del segundo tercio, donde está involucrado el coste del análisis de Fisher, que tal y como ya hemos visto en la sección 1.3.1. Este puede ser desdoblado en dos: por un lado, el cálculo de las matrices de dispersión en la última capa oculta, cuyo coste es $\mathcal{O}(N\,n_H^2)$, donde N es el número de patrones de entrenamiento y n_H es el número de unidades en la última capa oculta y por otro lado, el cálculo de los autovalores con un coste $\mathcal{O}((C-1)\,n_H^3)$, donde C es el número de clases en el conjunto de entrenamiento. Como normalmente $N >> n_H\,(C-1)$, será la matriz de dispersión la que domine el coste de este análisis y el coste computacional final hasta la salida de la red será la suma de las dos partes:

$$\mathcal{O}\left(N\,(n_H^2 + \sum_{h=0}^{H-1} n_{h+1}n_h)\right). \tag{3.15}$$

Por último, debemos calcular la complejidad en la actualización de los pesos $\boldsymbol{\mathcal{W}}$. Como en dicha actualización interviene el gradiente del criterio elegido, iremos desglosando el cálculo computacional en función de los criterios vistos en la sección anterior.

Coste Computacional del gradiente de J_1 respecto de los pesos no lineales

Comenzaremos recordando la derivada del criterio $J_1(\mathbf{W}) = |\mathbf{S}_T|/|\mathbf{S}_B|$ para un peso genérico W_{kl} perteneciente al conjunto de pesos $\boldsymbol{\mathcal{W}}$, ecuación (3.8)

$$\begin{aligned}
\frac{\partial J_1}{\partial W_{kl}} &= \frac{1}{|\mathbf{S}_B|}\left(\frac{\partial|\mathbf{S}_T|}{\partial W_{kl}} - J_1\frac{\partial|\mathbf{S}_B|}{\partial W_{kl}}\right)\\
&= J_1\ Tr\left(\mathbf{S}_T^{-1}E\left[\mathbf{D}_{kl}+\mathbf{D}_{kl}^T\right] - \mathbf{S}_B^{-1}\sum_{c=1}^{C}\pi_c\left(\mathbf{D}_{kl}^{\Omega_c}+(\mathbf{D}_{kl}^{\Omega_c})^T\right)\right),\\
\mathbf{D}_{kl} &= \frac{\partial\mathbf{Y}}{\partial W_{kl}}\left(\mathbf{Y}-\overline{\mathbf{Y}}\right)^T,\\
\mathbf{D}_{kl}^{\Omega_c} &= \frac{\partial\overline{\mathbf{Y}}_c}{\partial W_{kl}}\left(\overline{\mathbf{Y}}_c-\overline{\mathbf{Y}}\right)^T.
\end{aligned}$$

Observando el gradiente de J_1, vemos la necesidad de calcular las inversas y

determinantes de las matrices $\mathbf{S}_T$ y $\mathbf{S}_B$ cuyo coste es $\mathcal{O}\left((C-1)^3\right) \approx \mathcal{O}\left(C^3\right)$, donde $(C-1)$ es la dimensión del espacio de proyección del análisis discriminante de Fisher y C es el número de clases en la muestra. Sin embargo, ésta no es la parte costosa del gradiente, lo realmente costoso es el cálculo de la matriz esperanza $E\left[\mathbf{D}_{kl} + \mathbf{D}_{kl}^T\right]$ y la matriz $\sum_{c=1}^{C} \pi_c \left(\mathbf{D}_{kl}^{\Omega_c} + (\mathbf{D}_{kl}^{\Omega_c})^T\right)$, aunque ésta última se puede ir realizando a medida que se hace la anterior, con lo cual nos centraremos en el desarrollo del coste de $E\left[\mathbf{D}_{kl} + \mathbf{D}_{kl}^T\right]$ que sin duda alguna es lo que va a determinar el coste total del gradiente.

Calcular las matrices $(\mathbf{Y} - \overline{\mathbf{Y}})^T$ e $(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}})^T$ de las expresiones de $\mathbf{D}_{kl}$ y $\mathbf{D}_{kl}^{\Omega_c}$ respectivamente, no incrementa el coste del gradiente pues es una labor realizada con anterioridad y su coste es el mismo que el de obtener la salida de la red (ver expresión (3.15)), pero no interviene en el cómputo de la complejidad del cálculo del gradiente.

La parte más tediosa pertenece al cálculo de las derivadas $\partial\mathbf{Y}/\partial W_{kl}$ e $\partial\overline{\mathbf{Y}}_c/\partial W_{kl}$ para cada patrón de entrada. En la sección 3.4.1 vimos como se calcula cada una de las derivadas $\partial y_p/\partial W_{kl}$ $(p = 1, \cdots, C-1)$, cuyas expresiones son

$$\frac{\partial y_p}{\partial W_{kl}} = \frac{\partial y_p}{\partial o_l} f'(a_l) o_k, \tag{3.16}$$

donde la derivada $\partial y_p/\partial o_l$ se halla a partir de resultados conocidos de capas superiores a la capa receptora L

$$\frac{\partial y_p}{\partial o_l} = \sum_m \frac{\partial y_p}{\partial o_m} f'(a_m) W_{lm}.$$

Cuando el peso W_{kl} pertenece a una conexión con la última capa oculta, esta derivada es inmediata $\partial y_p/\partial o_l = W_{lp}^O$. Si seguimos con la expresión (3.16), el coste de calcular la derivada de la función de activación respecto a la activación $f'(a)$ es constante $\mathcal{O}(1)$. Manteniendo que el peso W_{kl} pertenece a la conexión con la última capa oculta, el coste de hallar $\partial y_p/\partial W_{kl} = W_{lp}^O f'(a_l) o_k$ será igualmente $\mathcal{O}(1)$. Luego, el cálculo del vector completo $\partial\mathbf{Y}/\partial W_{kl}$ será de $\mathcal{O}(C-1)$.

El siguiente paso es añadir el coste del producto $\mathbf{D}_{kl} = \partial\mathbf{Y}/\partial W_{kl}\ (\mathbf{Y} - \overline{\mathbf{Y}})^T$, el producto de las dos matrices es de orden $\mathcal{O}\left((C-1)^2\right)$ y el coste final del cálculo de $\mathbf{D}_{kl}$ es la suma del coste de calcular la derivada más realizar el producto de matrices, esto es: $\mathcal{O}(C-1) + \mathcal{O}\left((C-1)^2\right) = \mathcal{O}\left(C_{-1}(1+C_{-1})\right)$; para simplificar, a partir de ahora vamos a identificar $C_{-1} = C-1$.

Como se necesita hallar la esperanza de $\mathbf{D}_{kl}$, entonces el coste se ve incrementado por el número de patrones N, de la forma $\mathcal{O}\left(N\, C_{-1}(1+C_{-1})\right)$. Todo este cálculo es para un peso que conecta con la última capa oculta; como el número de pesos en estas condiciones es $n_H\ (n_{H-1}+1)$, donde n_H es el número de unidades en la última capa oculta y n_{H-1} el número de unidades, sin considerar el sesgo, de la capa anterior a la última oculta, de este modo el cálculo del gradiente de J_1 para los pesos que van de la última capa oculta a la de salida tiene un coste de $\mathcal{O}\left(\, n_H\ (n_{H-1}+1)\ N\, C_{-1}\,(1+C_{-1})\right)$.

Para el resto de los pesos W_{kl} no lineales, que son los que proceden de la capa h y se dirigen a la capa $(h+1)$, donde $(h = 0, \cdots, H-2)$, la derivada $\partial y_p / \partial W_{kl}$ se halla a partir de términos $\partial y_p / \partial o_l$ ya calculados por el método de retropropagación y guardados en memoria. Entonces es posible generalizar que el coste de las derivadas $\partial y_p / \partial W_{kl}$ es $\mathcal{O}(n_{h+2})$, donde n_{h+2} son las unidades, sin considerar el sesgo, de la capa siguiente a la capa receptora; de aquí se obtiene que el cálculo de $\partial \mathbf{Y} / \partial W_{kl}$ tendrá un coste de $\mathcal{O}(C_{-1}\, n_{h+2})$ y el cálculo de la matriz $\mathbf{D}_{kl}$ correspondiente al producto de las matrices $\partial \mathbf{Y} / \partial W_{kl}$ e $(\mathbf{Y} - \overline{\mathbf{Y}})^T$, será $\mathcal{O}\left(C_{-1}\, n_{h+2} + C_{-1}^2\right) = \mathcal{O}(C_{-1}(n_{h+2} + C_{-1}))$. Al incorporar los N patrones para calcular la esperanza de $\mathbf{D}_{kl}$, el coste anterior se incrementa por N, esto es $\mathcal{O}(N\, C_{-1}(n_{h+2} + C_{-1}))$.

Englobando todos los pesos no lineales de la red, el coste del cálculo del gradiente será el siguiente:

$$\mathcal{O}\left\{ N\, C_{-1} \left[(1 + C_{-1})\; n_H\; (n_{H-1} + 1) \;+\; \sum_{h=0}^{h=H-2} (n_{h+2} + C_{-1})\; (n_h + 1)\; n_{h+1} \right] \right\},$$

que podemos distribuirlo del siguiente modo

$$\mathcal{O}\left\{ N\, C_{-1} \left[\sum_{h=0}^{h=H-1} (1 + C_{-1})\; (n_h + 1)\; n_{h+1} \;+\; \sum_{h=0}^{h=H-2} (n_{h+2} - 1)\; n_{h+1}\; (n_h + 1) \right] \right\}$$

donde la cantidad

$$\mathcal{W} = \sum_{h=0}^{H-1} (n_h + 1)\; n_{h+1}$$

es el número de pesos no lineales en la red; así sustituyendo $\mathcal{W}$ en la expresión anterior, el coste final en el cálculo del gradiente de J_1 es:

$$\mathcal{O}\left\{ N\, C_{-1} \left(C_{-1} \mathcal{W} \;+\; \sum_{h=0}^{h=H-2} (n_{h+2} - 1)\; n_{h+1}\; (n_h + 1) \right) \right\} \approx$$
$$\mathcal{O}\left\{ N\, C^2\, \mathcal{W} \;+\; N\, C \sum_{h=0}^{h=H-2} n_{h+2}\;\; n_{h+1}\;\; n_h \right\}.$$

En general, como el número de unidades en las capas ocultas es mayor que el de la salida, es previsible que sea el segundo término de la ecuación anterior el que domine, es decir

$$C\, \mathcal{W} \leq \sum_{h=0}^{h=H-2} n_{h+2}\;\; n_{h+1}\;\; n_h,$$

No es así para el caso de una sola capa oculta, donde el término $C\, \mathcal{W} \approx C\, D\, K$ es el dominante, K se corresponde con el número de unidades en la capa oculta y D con el número de atributos. En el resto de los casos, es presumible que domine el

segundo término, acentuándose más este dominio cuanto más aumenta el número de capas ocultas. Luego, podemos considerar que la carga dominante en el cálculo del gradiente de J_1 es:

$$\left.\begin{array}{ll} \mathcal{O}\left(N\, C^2\, D\, K\right) & \text{para una sóla capa oculta} \\ \mathcal{O}\left(N\, C \sum_{h=0}^{h=H-2} n_{h+2}\;\; n_{h+1}\;\; n_h\right) & \text{para dos o más capas ocultas.} \end{array}\right\} \quad (3.17)$$

Coste Computacional del gradiente de J_2 respecto de los pesos no lineales

De nuevo, introducimos el gradiente del criterio $J_2 = Tr(\tilde{\mathbf{\Lambda}}\, \mathbf{S}_T)/Tr(\mathbf{S}_B)$ desarrollado en la sección anterior, cuya expresión final (3.12) es

$$\begin{aligned} \frac{\partial J_2}{\partial W_{kl}} &= \frac{2}{Tr(\mathbf{S}_B)} \sum_{p=1}^{C-1} \left(\tilde{\lambda}_p E\left[A_{kl;p}\right] - \tilde{\Lambda}_p\, E\left[B_{kl;p}\right] - J_2 \sum_{c=1}^{C} \pi_c\, \overline{y}_p^{\Omega_c}\, E_c\left[B_{kl;p}\right] \right) \\ \tilde{\Lambda}_p &= \overline{y}_p\, (\tilde{\lambda}_p - J_2) \\ A_{kl;p} &= y_p\, \frac{\partial y_p}{\partial o_l}\, f'(a_l)\, o_k \\ B_{kl;p} &= \frac{\partial y_p}{\partial o_l}\, f'(a_l)\, o_k. \end{aligned}$$

Hallar cada una de los valores de $A_{kl;p}$ o $B_{kl;p}$ es de orden $\mathcal{O}(1)$ cuando el peso W_{kl} pertenece a la conexión con la última capa oculta o bien $\mathcal{O}(n_{h+2})$ cuando se refiere al resto de los pesos no lineales (se deduce del mismo modo que lo hemos hecho con el criterio J_1). Al ser necesario hallar la esperanza de estos valores el coste por cada componente de la red es de $\mathcal{O}(N)$ o $\mathcal{O}(N\, n_{h+2})$ en función de que el peso esté asociado con la última capa oculta o bien no lo esté. Considerando cada una de las salidas, nos ponemos en un coste $\mathcal{O}(N\,(C-1))$ o bien $\mathcal{O}(N\, n_{h+2}\,(C-1))$ por cada peso W_{kl}.

Añadiendo todos los pesos no lineales de la red nos queda que el cálculo del gradiente de J_2 es del orden

$$\begin{aligned} &\mathcal{O}\left\{ N\,(C-1) \left((n_{H-1}+1)\, n_H \;+\; \sum_{h=0}^{h=H-2} n_{h+2}\;\; n_{h+1}\;\; (n_h+1) \right) \right\} \;= \\ &\mathcal{O}\left\{ N\,(C-1) \left(\sum_{h=0}^{h=H-1} n_{h+1}\,(n_h+1) \;+\; \sum_{h=0}^{h=H-2} (n_{h+2}-1)\;\; n_{h+1}\;\; (n_h+1) \right) \right\} \;= \end{aligned}$$

$$\mathcal{O}\left\{N\,(C-1)\left(\mathcal{W}+\sum_{h=0}^{h=H-2}(n_{h+2}-1)\;\;n_{h+1}\;\;(n_h+1)\right)\right\}\;\approx$$
$$\mathcal{O}\left\{N\,C\,\left(\mathcal{W}+\sum_{h=0}^{h=H-2}n_{h+2}\;\;n_{h+1}\;\;n_h\right)\right\}$$

En este caso, es sin duda el segundo término el que determina el coste computacional del gradiente de J_2 con el conjunto de pesos no lineales

$$\mathcal{O}\left\{N\,C\,\left(\sum_{h=0}^{h=H-2}n_{h+2}\;\;n_{h+1}\;\;n_h\right)\right\}.$$

Para una red de una sola capa oculta el coste del cálculo del gradiente J_2 será $\mathcal{O}\,(N\,C\,D\,K)$. En resumen, el coste del cálculo del gradiente J_2 es:

$$\left.\begin{array}{ll}\mathcal{O}\,(N\;C\;D\;K) & \text{para una sóla capa oculta}\\ \mathcal{O}\left(N\;C\sum_{h=0}^{h=H-2}n_{h+2}\;\;n_{h+1}\;\;n_h\right) & \text{para dos o más capas ocultas.}\end{array}\right\}$$

El coste del cálculo del gradiente de J_2 es menor que el de J_1 en un factor C cuando el número de capas ocultas es la unidad; si el número de capas ocultas es mayor o igual a dos, el coste del cálculo del gradiente J_2 es exactamente igual que el de J_1, expresión (3.17). Sin embargo debemos tener en cuenta que en el criterio J_2 éste es el coste total, mientras que en el criterio J_1 es el coste dominante porque intervienen más factores como los correspondientes al cálculo de la inversa de las matrices. Luego siempre será más rápido el cálculo del gradiente J_2 que el del gradiente J_1.

Coste Computacional del gradiente de J_3 respecto de los pesos no lineales

La derivada del criterio $J_3 = 1/Tr(\mathbf{S}_T^{-1}\mathbf{S}_B)$ según la expresión (3.14) es

$$\frac{\partial J_3}{\partial W_{kl}} = {J_3}^2\;\;Tr\left(\mathbf{S}_T^{-1}E\left[\mathbf{D}_{kl}+(\mathbf{D}_{kl})^T\right]\mathbf{S}_T^{-1}\mathbf{S}_B\;-\;\mathbf{S}_T^{-1}\sum_c\pi_c\left(\mathbf{D}_{kl}^{\Omega_c}+(\mathbf{D}_{kl}^{\Omega_c})^T\right)\right).$$

Una vez más, los términos dominantes en el coste del cálculo computacional son $E\left[\mathbf{D}_{kl}+(\mathbf{D}_{kl})^T\right]$ y el cálculo simultáneo $\sum_c\pi_c\left(\mathbf{D}_{kl}^{\Omega_c}+(\mathbf{D}_{kl}^{\Omega_c})^T\right)$. Como ya hemos visto éste es precisamente el coste del cálculo del gradiente para el criterio J_1, luego el coste del cálculo del criterio J_3 es similar al del criterio J_1, tal y como habíamos mencionado anteriormente.

Como conclusión final a la carga computacional, tras los estudios anteriores se deduce que la red ADnL usando como función criterio J_2 es la más eficiente desde el punto de vista de menor coste computacional.

Capítulo 4

Aceleración de la Convergencia

4.1. Introducción

El entrenamiento de una red neuronal normalmente se formula como un problema de minimización. El objetivo es buscar el conjunto de parámetros óptimos $\mathbf{W}^*$ que minimizan una función de error $e(\mathbf{W})$ dependiente de los parámetros $\mathbf{W}$ involucrados en la definición de la arquitectura de la red.

En la sección 2.3.2 indicamos que en el aprendizaje de una red, métodos en los cuales la curvatura de la función de error se tenía en cuenta aceleraban el proceso de aprendizaje. Eso es lo que precisamente vamos a tratar en este capítulo, buscaremos cómo disminuir el tiempo de convergencia durante el entrenamiento.

Hasta el momento, los algoritmos de aprendizaje que hemos presentado en el capítulo anterior se basan en que el espacio de parámetros es considerado como un espacio euclídeo. Es decir, se trata de una estructura plana en donde entre cada par de puntos hay una única recta que es la curva más corta entre los dos puntos. De este modo lo más fácil e intuitivo es utilizar el gradiente $\nabla e(\mathbf{W})$ para conseguir los pesos óptimos de la red.

Pero por lo general, el espacio de parámetros es un espacio altamente dimensionado, cuya estructura es bastante compleja y considerarla como euclídea sería una simplificación considerable. El espacio de parámetros estará bien definido por la geometría de Riemann, que es una generalización de la geometría euclídea. Ya no hablamos exclusivamente de espacios planos, estamos ante espacios curvados; podríamos imaginar pues, que todos los parámetros están situados en la superficie de un cilindro o bien en la superficie de una esfera o bien en otras infinitas posibles superficies en las que la curva más corta entre dos puntos cualesquiera situados en la superficie ya no es una línea recta como ocurre en un plano.

En el espacio de Riemann, el gradiente ordinario de la función de error ya no representa la dirección óptima de descenso; sin embargo esta dirección estará bien representada por el que denominaremos *gradiente natural*.

A lo largo de este capítulo, veremos distintos métodos de aprendizaje en los que se tiene en cuenta la curvatura del espacio. Comenzaremos revisando el más estudiado, el *gradiente conjugado* y paulatinamente iremos introduciendo métodos mejores hasta llegar al más novedoso el *gradiente natural* y es en el que nos centraremos por su versatilidad y lo bien que se adapta a nuestro modelo.

Al incorporar el gradiente natural en el entrenamiento de una red se aprecia, en relación con otros métodos, una mejora en la convergencia; evitándose las zonas pseudo–estacionarias en las que los parámetros se quedan atrapados durante el proceso de entrenamiento. Incluso, si llegara a caer en una de estas poco fructíferas zonas, el gradiente natural disminuye considerablemente el número de épocas necesarias para liberarse del estancamiento, en [Dorronsoro et al., 2001b] se muestra este evento para el caso de discernimiento entre dos clases y en [Dorronsoro y González, 2002] para C clases.

4.2. Métodos de convergencia de segundo orden

En esta sección vamos a revisar varios métodos de segundo orden aplicables al algoritmo de minimización durante el aprendizaje de la red. Comenzaremos por el más simple, el método de Newton, y una simplificación de éste hecha por Gauss. Después analizaremos dos métodos: gradiente conjugado y Quasi–Newton. Ambos métodos son muy eficaces y no hay mucha diferencia en cuanto a rendimiento entre ellos; por tradición quizás es más usado el gradiente conjugado, pero hoy en día no hay motivo alguno para desechar el método Quasi–Newton. A continuación veremos el método de Levenberg–Marquardt que es sólo valido para funciones de error del tipo suma de errores cuadráticos; si se cumple dicha condición el método funciona muy bien.

Todos los métodos de segundo orden que hemos indicado hasta el momento requieren que el entrenamiento de la red sea efectuado en modo *batch*. Esta exigencia es debida a que todos los métodos necesitan bien la matriz del hessiano exacta o bien una aproximación a ésta y para calcular dicha matriz se requiere el conjunto de entrenamiento completo.

Por último veremos la definición de gradiente natural que es el método más sofisticado de todos los que se van a exponer en esta sección. Este es el único de los métodos que se verán en esta sección que puede aplicarse para entrenamientos *on–line* cuando el prototipo de la red lo permite.

4.2.1. Método de Newton

El más simple de los métodos de segundo orden es el método de Newton. Por su sencillez y como base para el desarrollo de los próximos métodos de segundo orden lo incluiremos en este apartado.

Los cambios que se producen en la función criterio debido a la modificación de los pesos durante el entrenamiento pueden ser descritos según el desarrollo en serie de Taylor

$$\begin{aligned}\Delta J(\mathbf{W}) &= J(\mathbf{W}+\Delta\mathbf{W}) - J(\mathbf{W}) \\ &\approx (\nabla J(\mathbf{W}))^T \Delta\mathbf{W} + \frac{1}{2}\Delta\mathbf{W}^T\ \mathbf{H}\ \Delta\mathbf{W},\end{aligned}$$

donde $\mathbf{H}$ es la matriz del Hessiano, cuyos elementos son las derivadas segundas de la función de error respecto a los pesos de la red, esto es

$$H_{ij} = \frac{\partial^2 J(\mathbf{W})}{\partial w_i\ \partial w_j}$$

Si diferenciamos la ecuación anterior respecto a $\Delta\mathbf{W}$ para buscar el mínimo de $\Delta J(\mathbf{W})$ se tiene que

$$\nabla J(\mathbf{W}) + \mathbf{H}\ \Delta\mathbf{W} = \mathbf{0}.$$

Luego, el cambio óptimo en los pesos puede ser expresado como

$$\Delta\mathbf{W} = -\mathbf{H}^{-1}\nabla J(\mathbf{W}).$$

El descenso por gradiente simple, tal y como vimos en la sección 2.3.2 usaba la dirección del vector gradiente local $-\nabla J(\mathbf{W})$ para buscar el mínimo, con el inconveniente de que en general el vector gradiente no apunta hacia el mínimo de la función de error; sin embargo si tomamos la dirección de descenso del vector de Newton dada por $-\mathbf{H}^{-1}\nabla J(\mathbf{W})$ y evaluando para un vector $\mathbf{W}$ dentro de una superficie de error cuadrática, se tiene que el vector de Newton siempre apunta directamente al mínimo de la función de error.

Finalmente, las iteraciones del algoritmo de descenso por gradiente utilizando el método de Newton nos quedan de la forma

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \Delta\mathbf{W} = \mathbf{W}_t - \rho_t\ \mathbf{H}_t^{-1}\ \nabla J(\mathbf{W}_t),$$

donde ρ_t dentro del contexto de aprendizaje en redes neuronales se denomina constante o razón de aprendizaje y se trata de un parámetro que puede ser fijo o variar con el tiempo.

4.2.2. Aproximación de Gauss y Método de Gauss–Newton

El método de Newton junto con la aproximación al hessiano que vamos a ver a continuación es lo que se conoce como método de Gauss–Newton.

Si el criterio a minimizar puede expresarse como la suma sobre todo el conjunto de patrones de errores cuadráticos

$$J = \frac{1}{2}\sum_{n=1}^{N}(\mathcal{E}_n)^2 = \frac{1}{2}\parallel\mathcal{E}\parallel^2, \tag{4.1}$$

entonces los elementos de la matriz del hessiano de la función de error suma de cuadrados (4.1) vienen dados por

$$H_{jk} = \frac{\partial^2 J}{\partial w_j \, \partial w_k} = \sum_{n=1}^{N} \left\{ \frac{\partial \mathcal{E}_n}{\partial w_j} \frac{\partial \mathcal{E}_n}{\partial w_k} + \mathcal{E}_n \frac{\partial^2 \mathcal{E}_n}{\partial w_j \, \partial w_k} \right\}.$$

El término de las derivadas segundas puede despreciarse, pues normalmente es pequeño frente al término de las primeras derivadas. En efecto, si tenemos en cuenta que el factor que precede a las segundas derivadas es el error cometido para cada patrón, en una red adecuada estos errores serán valores aleatorios con signo y en general no deberían estar correlacionados. De modo que al sumar sobre todos los patrones el término de la segunda derivada tenderá a cancelarse. Por otro lado, incluir este término podría desestabilizar el aprendizaje de la red si el modelo no se ajusta correctamente o bien si existen patrones excéntricos imposibles de compensar con otros de signo opuesto. A la simplificación de despreciar el término de segundas derivadas se la conoce como aproximación de Gauss.

Si consideramos como función criterio la media de (4.1), la aproximación de Gauss puede expresarse como

$$\mathbf{H}_{Gauss} = E\left[\nabla \mathbf{J} \; (\nabla \mathbf{J})^T\right], \tag{4.2}$$

lo que simplifica mucho el algoritmo de minimización, pero aún así el método de Gauss–Newton tiene varias desventajas:

1. Elevado coste computacional en la evaluación de la aproximación del hessiano en una red no lineal. Se requieren para ello $\mathcal{O}\left(N\mathcal{W}^2\right)$ operaciones, donde N es el número de patrones y $\mathcal{W}$ es el número de pesos en la red.

2. La necesidad de invertir la matriz del hessiano, lo que incrementa en cada iteración un coste de $\mathcal{O}\left(\mathcal{W}^3\right)$. Por lo general, el número de patrones de entrenamiento N es mucho mayor que el número de parámetros de la red $\mathcal{W}$ con lo que el coste de la inversión de la matriz no es tan importante para el cómputo final, esto es $\mathcal{O}\left(N\mathcal{W}^2\right) >> \mathcal{O}\left(\mathcal{W}^3\right)$.

3. En un mínimo global la matriz del hessiano es definida siempre positiva; luego es posible admitir que cerca del mínimo el hessiano es una matriz simétrica definida positiva; pero si se diera el caso de tener una matriz que no es positiva, nos indica que existen direcciones de curvatura negativa tales como máximos o puntos de silla y en estos casos no es posible garantizar la reducción del error en cada iteración. Además hay que tener en cuenta que el método de Gauss–Newton utiliza una aproximación del hessiano, que a lo sumo da una matriz semidefinida positiva.

4. El método de Gauss–Newton sólo es aplicable en funciones de error cuadrático.

4.2.3. Método del Gradiente Conjugado

Un método rápido de aprendizaje es el descenso por gradiente conjugado. En este método se buscan sucesivas líneas de descenso basándose en direcciones conjugadas, o dicho de otro modo direcciones perpendiculares, dentro del espacio de parámetros.

Partiendo de una dirección de descenso inicial nos movemos en esta dirección hasta que se alcanza el mínimo local. En este punto se halla la segunda dirección que tiene la particularidad de ser conjugada respecto a la primera, lo que permite que el descenso a lo largo de esta nueva dirección no deteriora la contribución efectuada por el paso anterior. Estos pasos están ilustrados en la figura (4.2).

Si tomamos $\delta \mathbf{W}_t$ como la dirección de la línea de descenso en el paso t, la siguiente dirección $\delta \mathbf{W}_{t+1}$, tiene la propiedad de no estropear la minimización realizada a lo largo de la dirección anterior, donde se tiene que

$$(\nabla J(\mathbf{W}_{t+1}))^T \ \delta \mathbf{W}_t = 0. \tag{4.3}$$

La siguiente dirección $\delta \mathbf{W}_{t+1}$ se elige de forma que la componente pararela del cambio en el gradiente con la nueva dirección $\nabla J(\mathbf{W}_{t+1} + \lambda \ \delta \mathbf{W}_{t+1})$ respecto a la dirección anterior $\delta \mathbf{W}_t$ permanece inalterable o, dicho de otro modo, el cambio en el gradiente debe ser perpendicular a $\delta \mathbf{W}_t$:

$$(\nabla J(\mathbf{W}_{t+1} + \lambda \ \delta \mathbf{W}_{t+1}))^T \ \delta \mathbf{W}_t = 0.$$

Expandiendo la ecuación anterior a primer orden respecto de λ y dado que según la ecuación (4.3) el término de orden cero es nulo, se obtiene pues que

$$\delta \mathbf{W}_{t+1}^T \ \mathbf{H} \ \delta \mathbf{W}_t = 0, \tag{4.4}$$

donde $\mathbf{H}$ es la matriz del hessiano evaluada en el punto $\mathbf{W}_{t+1}$. Pares de direcciones de descenso que obedecen la ecuación anterior se dice que son *vectores conjugados*. Si el hessiano es proporcional a la matriz identidad, entonces tales direcciones son ortogonales en el espacio de parámetros. En cualquier caso, si el hessiano es una matriz definida positiva, los dos vectores serán linealmente independientes.

La dirección de descenso en una nueva iteración viene dada por la dirección del gradiente en el punto actual más una componente procedente de la anterior dirección de descenso

$$\delta \mathbf{W}_{t+1} = -\nabla J(\mathbf{W}_{t+1}) + \beta_t \ \delta \mathbf{W}_t. \tag{4.5}$$

El vector $\mathbf{W}_{t+1}$ de la expresión (4.5) se obtiene por minimización en línea a lo largo de la dirección de $\delta \mathbf{W}_t$, esto es

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \lambda_t \ \ \delta \mathbf{W}_t$$

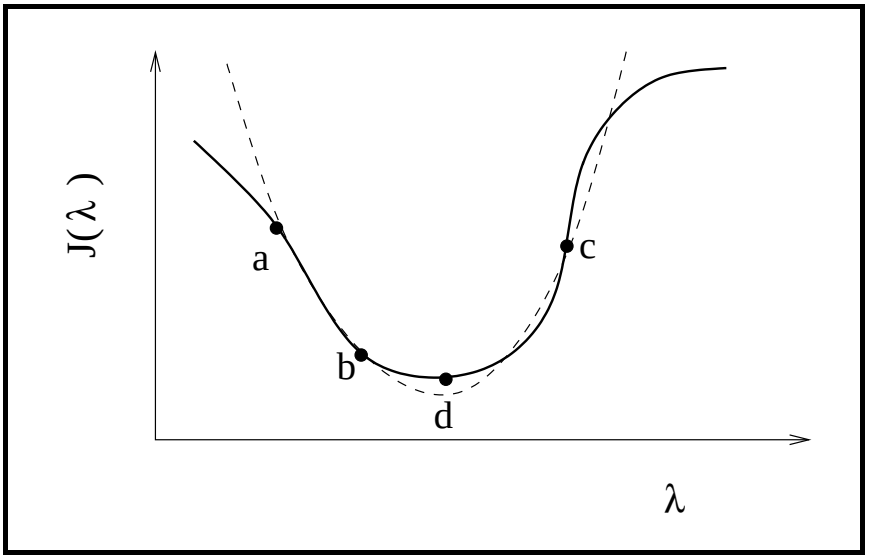


Figura 4.1: Minimización en línea con interpolación parabólica.

donde el parámetro λ_t debe cumplir

$$\frac{\partial}{\partial \lambda_t} J(\mathbf{W}_t + \lambda_t \ \ \delta \mathbf{W}_t) = 0.$$

Encontrar $\mathbf{W}_{t+1}$ desemboca en un problema de minimización unidimensional donde el parámetro a buscar es el factor λ. El proceso de minimización en línea se realiza en dos etapas. La primera es acotar el mínimo en la dirección $\delta \mathbf{W}_t$ con tres puntos $a < b < c$, de modo que se cumpla las condiciones $J(a) > J(b)$ y $J(c) > J(b)$, tal como se indica en la figura (4.1). La función de error J como una función de la distancia λ a lo largo de la dirección $\delta \mathbf{W}_t$ está representada en la figura (4.1) como una línea continua. Como $J(\lambda)$ es una función continua, la elección de los tres puntos anteriores nos asegura que en el intervalo (a, c) existe un valor mínimo.

La segunda etapa es encontrar el mínimo en sí. Dado que la función de error es continua y suave es posible realizar esta búsqueda por interpolación parabólica; es decir se intenta ajustar un polinomio de orden dos, evaluado en los tres puntos anteriores, a la función de error $J(\lambda)$ y se toma el mínimo de la parábola como la aproximación al mínimo de $J(\lambda)$. La figura (4.1) también representa la interpolación parabólica, el nuevo punto está representado por d y corresponde a la posición del mínimo de la parábola que pasa por los puntos a, b y c, representada por la línea punteada. El proceso se repite ajustando otra parábola que pasa por el nuevo punto y por los dos puntos de menor error de la etapa anterior. Este proceso iterativo se repite hasta la convergencia al mínimo, obteniéndose el factor λ buscado.

El coeficiente β_t de la expresión (4.5) se obtiene imponiendo la condición de conjugación, expresión (4.4), que asegura que la dirección de descenso en la itera-

ción $t+1$ no estropea los cambios efectuados por la anterior iteración t y por inducción por todas las anteriores a ésta, cumpliéndose sólo para funciones cuadráticas. Existen tres posibilidades de elegir el coeficiente β_t; todas ellas son equivalentes si la función de error es cuadrática:

1. La primera fórmula es la expresión de Hestenes–Stiefel

$$\beta_t = \frac{(\nabla J(\mathbf{W}_{t+1}))^T \ (\nabla J(\mathbf{W}_{t+1}) - \nabla J(\mathbf{W}_t))}{(\delta \mathbf{W}_t)^T \ (\nabla J(\mathbf{W}_{t+1}) - \nabla J(\mathbf{W}_t))}$$

2. La siguiente es la fórmula de Fletcher–Reeves

$$\beta_t = \frac{(\nabla J(\mathbf{W}_{t+1}))^T \ \nabla J(\mathbf{W}_{t+1})}{(\nabla J(\mathbf{W}_t))^T \ \nabla J(\mathbf{W}_t)}$$

3. Por último la fórmula de Polak–Ribiere

$$\beta_t = \frac{(\nabla J(\mathbf{W}_{t+1}))^T \ (\nabla J(\mathbf{W}_{t+1}) - \nabla J(\mathbf{W}_t))}{(\nabla J(\mathbf{W}_t))^T \ \nabla J(\mathbf{W}_t)}$$

Las tres expresiones anteriores son equivalentes cuando las funciones de error a minimizar son cuadráticas, es decir, su expresión algebraica es de la forma $f(\mathbf{x}) = a + \mathbf{p}^T\mathbf{x} + \mathbf{x}^T\mathbf{q}\mathbf{x}$, donde $\mathbf{p}$ y $\mathbf{q}$ son un vector y una matriz cuadrada, respectivamente. En este caso se garantiza que el gradiente conjugado converge al mínimo $\mathbf{W}^*$ en tantas iteraciones como el número de pesos $\mathcal{W}$. No vamos a demostrarlo, pero la equivalencia entre las tres expresiones se debe a las siguientes propiedades:

I. El gradiente en el paso t es ortogonal a todas las direcciones de descenso anteriores

$$(\delta \mathbf{W}_k)^T \ \nabla J(\mathbf{W}_t) = 0 \qquad \forall \quad k < t \leq \mathcal{W}.$$

II. El gradiente en el paso t es ortogonal a todos los gradientes anteriores

$$(\nabla J(\mathbf{W}_k))^T \nabla J(\mathbf{W}_t) = 0 \qquad \forall \quad k < t \leq \mathcal{W}.$$

Si las funciones a minimizar no son algebraicamente cuadráticas, los valores de las tres expresiones de β_t difieren entre sí; de las tres expresiones la más robusta es la última. La explicación recae en la situación en la que el algoritmo está realizando pequeños progresos y los sucesivos gradientes son similares; con la fórmula de Polak–Ribiere se obtiene el valor de β_t más pequeño y la búsqueda de la siguiente dirección de descenso según la ecuación (4.5) tiene la tendencia de reiniciar el proceso con la dirección negativa del gradiente, lo que equivale a reiniciar el gradiente conjugado.

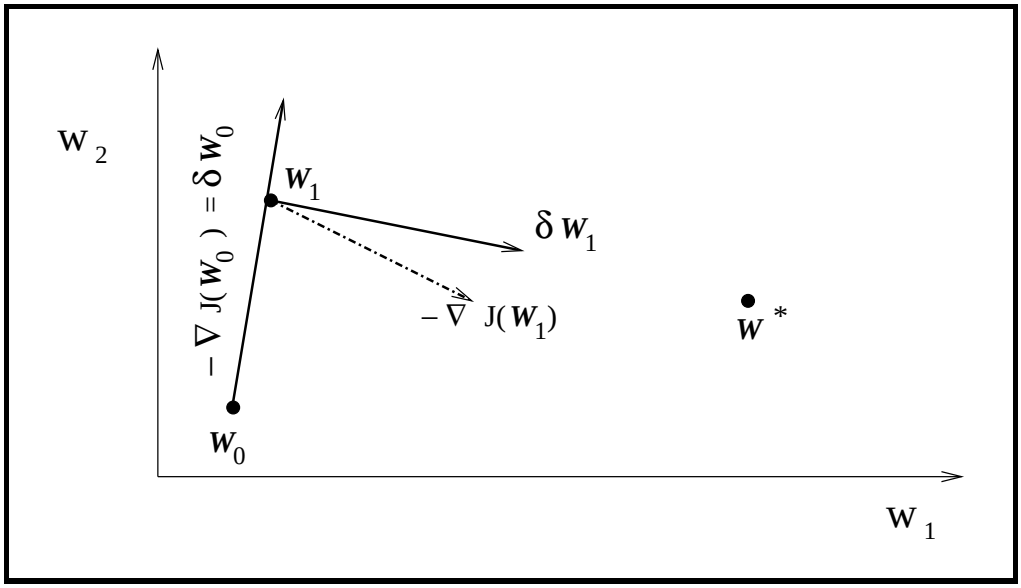


Figura 4.2: Primera iteración en el gradiente conjugado.

En la figura (4.2) se representa esquemáticamente en dos dimensiones la primera iteración del algoritmo de descenso por gradiente, aunque bien puede extrapolarse a cualquier otra iteración. Se ha tomado un peso inicial $\mathbf{W}_0$ y como primera dirección de descenso se toma el gradiente negativo para este peso $-\nabla J(\mathbf{W}_0)$. Sobre esta dirección se busca el mínimo con el método de minimización en línea y se alcanza en el punto $\mathbf{W}_1$. Ahora hay que buscar la nueva dirección de descenso según la fórmula (4.5); en la figura se ilustra esta nueva dirección como $\delta\mathbf{W}_1$ que es distinta de su gradiente negativo $-\nabla J(\mathbf{W}_1)$ por el factor de corrección $\beta_0\ \delta\mathbf{W}_0$. El siguiente paso sería buscar sobre la nueva dirección $\delta\mathbf{W}_1$ el mínimo, obteniéndose el punto $\mathbf{W}_2$ y se continuaría con el proceso hasta alcanzar el mínimo global $\mathbf{W}^*$. Sabiendo que si la función de error fuera cuadrática se llegaría al mínimo en tantos pasos como el número de dimensiones del espacio de parámetros, en esta ilustración con tan sólo dos pasos llegaríamos al mínimo $\mathbf{W}^*$.

El gradiente conjugado está desarrollado para minimizar funciones cuadráticas, siendo su efectividad en este caso absoluta. Sin embargo para funciones no cuadráticas, se cumple que en la vecindad de un punto el comportamiento de la función será aproximadamente cuadrático, con lo cual cabría esperar que repetidas iteraciones del método condujeran con eficiencia a la convergencia al mínimo de la función.

4.2.4. Método Quasi–Newton

Ya hemos discutido que una aplicación directa del método de Newton

$$\mathbf{W}^* = \mathbf{W} - \mathbf{H}^{-1}\ \nabla J(\mathbf{W}) \tag{4.6}$$

puede ser computacionalmente costosa, ya que requiere por iteración $\mathcal{O}(N\mathcal{W}^2)$ operaciones para calcular la matriz del hessiano y $\mathcal{O}(\mathcal{W}^3)$ para hallar su inversa; recordamos que N es el número de patrones y $\mathcal{W}$ es el número de pesos en la red.

El método Quasi–Newton o también conocido por de *métrica variable* es una aproximación al método de Newton. Se basa en la metodología de Newton, pero en vez de hallar el hessiano y su inversa realiza una aproximación a la inversa en un número finito de pasos; de ahí procede el término *Quasi*.

Al igual que con el gradiente conjugado, éste método encuentra el mínimo de una función cuadrática en un máximo de $\mathcal{W}$ pasos con un coste computacional total de $\mathcal{O}(N\mathcal{W}^2)$.

La idea básica del método Quasi–Newton es construir iterativamente una buena aproximación a la inversa de la matriz del hessiano usando información del gradiente de la función de error, sin necesidad de llegar a segundo orden en la derivación. Para cualquier tipo de función y considerando aproximación cuadrática cerca del mínimo se calcula una secuencia de matrices $\mathbf{G}_t$ con la propiedad de que

$$\lim_{t\to\infty} \mathbf{G}_t = \mathbf{H}^{-1}.$$

En la realidad el número de iteraciones es finito y depende de la precisión de la máquina. En concreto, para un problema cuadrático $\mathcal{W}$–dimensional, la secuencia de matrices $\mathbf{G}_t$ converge exactamente a la inversa del hessiano en un máximo de $\mathcal{W}$ pasos, alcanzándose el mínimo exacto de la función de error.

La principal desventaja que tiene el método Quasi–Newton frente al gradiente conjugado reside en la necesidad de almacenar la matriz $\mathbf{G}$ de dimensión $\mathcal{W}\times\mathcal{W}$ frente a la dirección de descenso $\delta\mathbf{W}$ de dimensión $\mathcal{W}$; pero hoy en día con el gran capacidad de memoria que tienen las máquinas convencionales y para redes moderadas ésto es un problema insignificante.

En general, lejos del mínimo no tenemos la garantía de que el hessiano sea una matriz definida positiva; esto es lo que hace ineficiente al método de Newton, que puede derivar a la situación en la que la función de error incremente en cada iteración. El método Quasi–Newton comienza con una matriz $\mathbf{G}_0$ simétrica y definida positiva que normalmente es la matriz identidad, lo que equivale a realizar el primer paso en la dirección impuesta por el gradiente negativo, y a partir de ahí se van construyendo las sucesivas matrices aproximadas $\mathbf{G}_t$, de modo que siempre se mantienen simétricas y definidas positivas. Con esto, lejos del mínimo se garantiza que siempre nos movemos en la dirección de descenso y cerca del mínimo al ser la aproximación $\mathbf{G}$ semejante al hessiano real estamos en las condiciones de eficacia en la convergencia cuadrática del método de Newton, expresión (4.6).

Partiendo de la fórmula (4.6), vemos que los pesos en el paso $t+1$ y t están relacionados con sus correspondientes gradientes por la expresión

$$\mathbf{W}_{t+1} - \mathbf{W}_t = -\mathbf{H}^{-1}(\nabla J(\mathbf{W}_{t+1}) - \nabla J(\mathbf{W}_t)),$$

que es conocida como la condición Quasi–Newton. La aproximación de la inversa del hessiano $\mathbf{G}$ debe satisfacer también dicha condición.

La actualización de la matriz aproximada $\mathbf{G}$ puede realizarse por dos procedimientos distintos: *Davidson–Fletcher–Powell* (DFP) y *Broyden–Fletcher–Goldfard–Shanno* (BFGS). Ambos procedimientos difieren tan sólo en detalles de errores de redondeo, tolerancia a la convergencia, ...; sin embargo, está reconocido que BFGS es superior a DFP.

La fórmula adaptativa de DFP es la siguiente

$$\begin{aligned}\mathbf{G}_{t+1} &= \mathbf{G}_t + \frac{(\mathbf{W}_{t+1}-\mathbf{W}_t)\,(\mathbf{W}_{t+1}-\mathbf{W}_t)^T}{(\mathbf{W}_{t+1}-\mathbf{W}_t)^T\,(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))} \\ &- \frac{\mathbf{G}_t\,(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))\;\;(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))^T\,\mathbf{G}_t}{(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))^T\;\;\mathbf{G}_t\;\;(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))}.\end{aligned}$$

La fórmula de BFGS es la anterior añadiendo el siguiente término

$$\cdots + \left[(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))^T\;\;\mathbf{G}_t\;\;(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))\right]\;\;\mathbf{U}\,\mathbf{U}^T$$

donde el vector $\mathbf{U}$ está definido por

$$\begin{aligned}\mathbf{U} &= \frac{(\mathbf{W}_{t+1}-\mathbf{W}_t)}{(\mathbf{W}_{t+1}-\mathbf{W}_t)^T\;\;(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))} \\ &- \frac{\mathbf{G}_t\;\;(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))}{(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))^T\;\;\mathbf{G}_t\;\;(\nabla J(\mathbf{W}_{t+1})-\nabla J(\mathbf{W}_t))}.\end{aligned}$$

No vamos a entrar en detalles de cómo se derivan las expresiones de actualización de la matriz $\mathbf{G}$. Para una consulta exhaustiva remitimos al libro clásico de métodos de optimización [Polak, 1971].

En cada paso del método Quasi–Newton, la dirección dada por $-\mathbf{G}\nabla J$ garantiza ser una dirección de descenso, puesto que la matriz $\mathbf{G}$ es definida positiva. Sin embargo, un paso completo del método de Newton, expresión (4.6), puede hacer que se salga del rango de validez de la aproximación cuadrática. La solución a este problema es buscar en la dirección de descenso, dada por $-\mathbf{G}\nabla J$, el mínimo de la función de error, del mismo modo que se hizo con el gradiente conjugado. Así de esta forma el vector de pesos se actualiza como

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \alpha_t \mathbf{G}_t \nabla J(\mathbf{W}_t)$$

donde α_t es un factor dependiente del método de minimización en la dirección de descenso. Una ventaja significativa del método Quasi–Newton frente al gradiente conjugado es que en el primero las líneas de descenso no tienen que ser encontradas con una gran precisión, ya que no es ése el factor crítico del algoritmo; mientras que en el gradiente conjugado tienen que conocerse exactamente, pues de ello depende el sistema de vectores conjugados, expresión (4.4). En cualquier caso, hemos asumido que la minimización a lo largo de las direcciones de descenso $-\mathbf{G}\nabla J$ se realiza con total eficacia.

4.2.5. Método de Levenberg–Marquardt

Los algoritmos de optimización anteriormente expuestos son métodos diseñados para ser utilizados en un amplio rango de funciones de error; el algoritmo de Levenberg–Marquardt está diseñado específicamente para minimizar una suma de errores cuadráticos

$$E = \frac{1}{2}\sum_{n=1}^{N}(\mathcal{E}_n)^2 = \frac{1}{2} \parallel \mathcal{E} \parallel^2 \tag{4.7}$$

donde $\mathcal{E}_n$ es el error para el patrón n–ésimo y $\mathcal{E}$ es el vector con todos los elementos $\mathcal{E}_n$ y por tanto será de dimensión $N \times 1$. Aquí haremos hincapié en que estamos delante de la función de error de un PMC, luego el método Levenberg–Marquardt estará especialmente indicado para entrenar PMCs.

Si durante el entrenamiento de la red, estamos en el estado gobernado por los pesos $\mathbf{W}_t$ y pasamos al gobernado por los pesos $\mathbf{W}_{t+1}$, manteniendo la diferencia entre los dos estados pequeña, entonces es posible expandir el vector de error $\mathcal{E}$ según el desarrollo en serie de Taylor de primer orden, de tal modo que

$$\mathcal{E}(\mathbf{W}_{t+1}) = \mathcal{E}(\mathbf{W}_t) + \nabla\mathcal{E}(\mathbf{W}_t) \ \ (\mathbf{W}_{t+1} - \mathbf{W}_t).$$

La función de error (4.7) para el estado de $\mathbf{W}_{t+1}$ considerando la anterior aproximación puede reescribirse como

$$E = \frac{1}{2} \parallel \mathcal{E}(\mathbf{W}_t) + \nabla\mathcal{E}(\mathbf{W}_t) \ \ (\mathbf{W}_{t+1} - \mathbf{W}_t) \parallel^2 . \tag{4.8}$$

Si minimizamos el error E respecto a los nuevos pesos $\mathbf{W}_{t+1}$ se obtiene que

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \left[(\nabla\mathcal{E}(\mathbf{W}_t))^T \ \nabla\mathcal{E}(\mathbf{W}_t) \ \right]^{-1} \ \ (\nabla\mathcal{E}(\mathbf{W}_t))^T \ \ \mathcal{E}(\mathbf{W}_t).$$

Si nos fijamos en la la expresión anterior, se aprecia que implícitamente está incorporada la aproximación de Gauss (4.2) para el hessiano; de este modo la actualización de pesos se puede expresar como

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \mathbf{H}_{Gauss}^{-1} \ (\nabla\mathcal{E}(\mathbf{W}_t))^T \ \ \mathcal{E}(\mathbf{W}_t). \tag{4.9}$$

En principio, la fórmula de actualización de pesos (4.9) puede ser perfectamente válida en la búsqueda del mínimo de la función y en consecuencia encontrar los pesos óptimos $\mathbf{W}^*$. Pero existe un problema: si $\mathbf{H}_{Gauss}$ pasa a ser una matriz singular o quasi–singular, entonces el tamaño del paso se desborda y la aproximación lineal de la función de error (4.8), en la que está basado todo el desarrollo anterior, deja de cumplirse.

El algoritmo de Levenberg–Marquardt resuelve este problema buscando el mínimo de la función de error mientras que al mismo tiempo se intenta mantener el tamaño del paso pequeño, con el fin de asegurarse una aproximación lineal (4.8) válida. Para ello, define una función de error modificada

$$\tilde{E} = \frac{1}{2} \parallel \mathcal{E}(\mathbf{W}_t) + \nabla\mathcal{E}(\mathbf{W}_t) \ \ (\mathbf{W}_{t+1} - \mathbf{W}_t) \parallel^2 + \lambda \parallel \mathbf{W}_{t+1} - \mathbf{W}_t \parallel^2, \tag{4.10}$$

donde el parámetro λ gobierna el tamaño del paso.

Si minimizamos la función de error modificada (4.10) respecto del estado $\mathbf{W}_{t+1}$ del mismo modo que en (4.8), se obtiene que

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \left[(\nabla\mathcal{E}(\mathbf{W}_t))^T \ \nabla\mathcal{E}(\mathbf{W}_t) \ + \ \lambda \ \mathbf{I} \ \right]^{-1} \ \ (\nabla\mathcal{E}(\mathbf{W}_t))^T \ \ \mathcal{E}(\mathbf{W}_t), \quad (4.11)$$

que utilizando la versión aproximada del hessiano nos queda de la forma

$$\mathbf{W}_{t+1} = \mathbf{W}_t - (\mathbf{H}_{Gauss} + \lambda\mathbf{I})^{-1} \ (\nabla\mathcal{E}(\mathbf{W}_t))^T \ \ \mathcal{E}(\mathbf{W}_t),$$

donde $\mathbf{I}$ es la matriz identidad.

Se puede apreciar que para valores muy pequeños de λ recuperamos la fórmula de Gauss–Newton, lo que indica que nos encontramos cerca del mínimo y el progreso del algoritmo es muy rápido; mientras que para valores de λ grandes estamos frente a un descenso por gradiente estándar, en el que el tamaño del paso está determinado por λ^{-1}, con lo cual, si el valor de λ es grande se genera un pequeño paso en la dirección negativa del gradiente, lo que implica que aunque lentamente (el valor $\| \ \mathbf{W}_{t+1} - \mathbf{W}_t \ \|^2$ tenderá a hacerse pequeño) nos acercamos hacia el mínimo de la función de error.

Los valores de λ son adaptativos y se van modificando en el transcurso de la búsqueda del mínimo. Es habitual comenzar por un valor de λ modesto como $\lambda_0 =$ 0,001 y en cada paso controlar su variación con el cambio en el error E. Si el error decrece después de una iteración según (4.11), entonces se toma el nuevo vector de pesos, λ se disminuye en un factor de 10 y se realiza una nueva iteración. Si por el contrario, el error aumenta tras la actualización de pesos en (4.11), se desecha este nuevo peso, se mantiene el antiguo y se repite el proceso con una λ incrementada en un factor de 10. El conjunto del proceso se repite hasta llegar a alguna condición de parada; podrían servir las que ya hemos visto en la sección 2.3.3. Nunca se debe parar después de un paso en el que el error E ha aumentado, pues ello implica que los valores de λ no han sido ajustado óptimamente y se debe continuar con el proceso de minimización.

4.2.6. Método del Gradiente Natural

Sea $\mathcal{S} = \{\mathbf{w} \in \Re^d\}$ el espacio de parámetros en el cual está definida la función $L(\mathbf{w})$. Cuando $\mathcal{S}$ es un espacio euclídeo con un sistema de coordenadas ortonormal definido por $\mathbf{w}$, el cuadrado de la distancia de un pequeño vector diferencial $d\mathbf{w}$ que conecta $\mathbf{w}$ y $\mathbf{w} + d\mathbf{w}$ viene dada por

$$\| \ d\mathbf{w} \ \|^2 = \sum_{i=1}^{d} (dw_i)^2,$$

donde dw_i son las componentes de $d\mathbf{w}$. Sin embargo, cuando el espacio no es euclídeo, esto es, cuando $\mathcal{S}$ es un espacio curvado genéricamente denominado

como espacio de Riemann, se tiene que las coordenadas del sistema ya no son ortonormales y la longitud de $d\mathbf{w}$ viene dada por la ecuación

$$\| d\mathbf{w} \|^2 = \sum_{i,j} g_{ij}(\mathbf{w})\, dw_i\, dw_j. \tag{4.12}$$

donde g_{ij} son los elementos de la matriz cuadrada $\mathbf{G}$, conocida como el tensor métrico de Riemann y describe la curvatura local del espacio de parámetros en el punto definido por $\mathbf{w}$. La matriz $\mathbf{G}$ depende de $\mathbf{w}$ y es una matriz simétrica y definida positiva. En notación vectorial, la ecuación (4.12) se reduce a

$$\| d\mathbf{w} \|^2 = (d\mathbf{w})^T\ \mathbf{G}\ d\mathbf{w}.$$

Si $\mathcal{S}$ es el espacio euclídeo, la matriz $\mathbf{G}$ se restringe a la matriz identidad $\mathbf{I}$ y vectorialmente $\| d\mathbf{w} \|^2 = d\mathbf{w}^T d\mathbf{w}$. En el espacio de Riemann la distancia entre dos puntos $\mathbf{x}$ e $\mathbf{y}$ está definida por

$$< \mathbf{x}, \mathbf{y} >_{\mathrm{G}} = \mathbf{y}^T \mathbf{G} \mathbf{x}.$$

En nuestro problema de intentar aprender los parámetros óptimos de la red $\mathbf{W}^*$, moviéndonos en la dirección de descenso de la función de error $e(\mathbf{W})$ definida en el espacio de parámetros, nos preguntamos: si tuviéramos en cuenta la curvatura del espacio de parámetros de la red, ¿sería más eficaz el avance hacia el mínimo de la función de error?. Vamos a ver que la respuesta es afirmativa y ahí es donde entra el gradiente natural definido en el espacio de Riemann.

Teorema 4.1 *La dirección de descenso máximo de la función $L(\mathbf{w})$ en un espacio de Riemann cuyo tensor métrico es $\mathbf{G}$ viene dada por*

$$-\tilde{\nabla} L(\mathbf{w}) = -\mathbf{G}^{-1}(\mathbf{w})\ \nabla L(\mathbf{w}),$$

donde $\mathbf{G}^{-1}$ es la inversa del tensor $\mathbf{G}$ y $\nabla L(\mathbf{w})$ es el gradiente de $L(\mathbf{w})$,

$$\nabla L(\mathbf{w}) = \left(\frac{\partial L(\mathbf{w})}{\partial w_1} \quad \cdots \quad \frac{\partial L(\mathbf{w})}{\partial w_d} \right)^T.$$

Luego, el *gradiente natural* de L en el espacio de Riemann viene dado por

$$\tilde{\nabla} L(\mathbf{w}) = \mathbf{G}^{-1}(\mathbf{w}) \nabla L(\mathbf{w}),$$

que en el espacio euclídeo es precisamente el gradiente ordinario $\tilde{\nabla} L(\mathbf{w}) = \nabla L(\mathbf{w})$. El algoritmo de descenso por gradiente natural estará definido por

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t \tilde{\nabla} L(\mathbf{w}_t), \tag{4.13}$$

donde η_t es la razón de aprendizaje que determina el tamaño del paso, pudiéndose ser ésta adaptativa (η_t) o bien una constante (η).

El cambio respecto al gradiente normal es obvio, tan sólo hemos sustituido $\nabla L(\mathbf{w})$ por la nueva definición de gradiente natural $\tilde{\nabla} L(\mathbf{w})$. Aunque aparentemente parecen métodos similares veremos en las próximas secciones que el gradiente natural mejora el rendimiento en el entrenamiento de una red y en particular lo veremos con la red ADnL.

4.3. Gradiente Natural en PMCs

En la sección anterior hemos visto qué se entiende por gradiente natural, pero no hemos dicho nada sobre cómo se define el tensor $\mathbf{G}$. A lo largo de esta sección, propondremos posibles matrices $\mathbf{G}$ dentro del contexto del entrenamiento de una red neuronal de retropropagación. Enfocaremos la búsqueda de la matriz $\mathbf{G}$ desde dos puntos de vista: el más formal procedente de estudios de Amari [Amari, 1990], donde la matriz $\mathbf{G}$ es la matriz de información de Fisher y mediante métodos de aproximación más sencillos; como ejemplo hemos elegido el desarrollado por Heskes [Heskes, 2000].

4.3.1. Matriz de Información de Fisher

Considerando una red neural multicapa con una arquitectura definida por el conjunto de pesos $\mathbf{W}$, denotaremos el conjunto $\mathcal{Z} = \{\mathcal{X}, \mathcal{Y}\} = \{\mathbf{x} \in \Re^d, \mathbf{y} \in \Re^m\}$, donde $\mathcal{X}$ es el conjunto de patrones d–dimensional que obedece a una distribución de probabilidad desconocida $q(\mathbf{x})$ e $\mathcal{Y}$ es el conjunto de respuestas asociadas al conjunto $\mathcal{X}$, donde cada vector $\mathbf{y}$ es un vector de dimensión m que sigue la distribución de probabilidad condicional $q(\mathbf{y}|\mathbf{x})$. Con todo, el conjunto $\mathcal{Z}$ se comporta según la distribución de densidad conjunta $q(\mathbf{z}) = q(\mathbf{x}, \mathbf{y}) = q(\mathbf{x})\ q(\mathbf{y}|\mathbf{x})$.

Como punto de partida al aprendizaje de la red consideramos un problema de estimación semiparamétrica. La red intenta encontrar los pesos óptimos $\mathbf{W}^*$ dentro de una familia de funciones densidades $\mathcal{P}_{\mathbf{W}} = p(\mathbf{x}, \mathbf{y}; \mathbf{W})$. La distribución correspondiente al conjunto de pesos óptimos se aproxima a la distribución de densidad conjunta, o lo que es lo mismo $p(\mathbf{x}, \mathbf{y}; \mathbf{W}^*) \approx q(\mathbf{x}, \mathbf{y})$.

Una medida de lo diferente que son dos distribuciones cualesquiera es la divergencia de Kullback–Leibler, que aplicada a las distribuciones $\mathcal{P}_{\mathbf{W}}$ y $q(\mathbf{x}, \mathbf{y})$ nos da

$$\begin{aligned} d_{KL}(q, \mathcal{P}_{\mathbf{W}}) &= E_q\left[\log \frac{q(\mathbf{z})}{p(\mathbf{z}; \mathbf{W})}\right] = \int \log \frac{q(\mathbf{z})}{p(\mathbf{z}; \mathbf{W})}\ q(\mathbf{z})\ d\mathbf{z} \\ &= \int \log q(\mathbf{z})\ \ q(\mathbf{z})\ d\mathbf{z} - \int \log p(\mathbf{z}; \mathbf{W})\ \ q(\mathbf{z})\ d\mathbf{z}. \end{aligned} \tag{4.14}$$

Observando la expresión (4.14), se puede deducir que minimizar respecto a $\mathbf{W}$ la discrepancia entre las dos distribuciones es totalmente equivalente a maximizar la verosimilitud logarítmica de $p(\mathbf{z}; \mathbf{W})$ definida como

$$L(\mathbf{W}) = \int \log p(\mathbf{z}; \mathbf{W})\ \ q(\mathbf{z})\ d\mathbf{z}.$$

Para tratar en los mismos términos de minimización, consideraremos la verosimilitud logarítmica negativa $Ln(\mathbf{W}) = -L(\mathbf{W})$.

Durante el entrenamiento de un PMC, lo que realmente se hace es buscar el conjunto de pesos óptimos $\mathbf{W}^*$ minimizando la discrepancia entre la etiqueta $\mathbf{y}$ del patrón $\mathbf{x}$ y la correspondiente salida de la red $\mathbf{f}(\mathbf{x};\mathbf{W})$. La función de error más típica a minimizar es el error cuadrático medio

$$E(\mathbf{W}) = E\left[\mathcal{E}(\mathbf{x},\mathbf{y},\mathbf{W})\right] = \frac{1}{2}\, E_q\left[\parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2\right], \tag{4.15}$$

que si lo comparamos con el escenario de minimizar $Ln(\mathbf{W})$, se reduce a un problema de estimación paramétrica siempre que sea posible definir la función de densidad $p(\mathbf{z};\mathbf{W}) = c \;\; q(\mathbf{x}) \;\; \exp\left\{-\frac{1}{2} \parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2\right\}$. Precisamente, en ésto se basa Amari [Amari, 1998] para introducir la definición de su matriz de información de Fisher. Amari considera un modelo estadístico de redes neuronales de tal forma que $\mathbf{y}$ es representada por una versión con ruido de la salida de la red $\mathbf{f}(\mathbf{x},\mathbf{W})$

$$\mathbf{y} = \mathbf{f}(\mathbf{x},\mathbf{W}) + \mathbf{n},$$

donde $\mathbf{n}$ es una función de distribución normal centrada en cero y con matriz de covarianza la matriz identidad $\mathbf{I}$. De este modo, la variable $\mathcal{Z}$ se distribuye según la densidad de probabilidad, $q(\mathbf{z})$, definida por el producto de las densidades de probabilidad $q(\mathbf{x})$ y $q(\mathbf{y}|\mathbf{x};\mathbf{W}) \approx N(\mathbf{f}(\mathbf{x},\mathbf{W}),\mathbf{I})$, esto es

$$p(\mathbf{z};\mathbf{W}) = c \;\; q(\mathbf{x}) \;\; \exp\left(-\frac{1}{2} \parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2\right), \tag{4.16}$$

donde c es la constante de normalización. La función verosimilitud logarítmica negativa de $p(\mathbf{z};\mathbf{W})$, expresada por (4.16), viene dada por

$$Ln(\mathbf{W}) = \; C - \int \log q(\mathbf{x}) \; q(\mathbf{z}) \; d\mathbf{z} + \frac{1}{2} \int \parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2 \;\; q(\mathbf{z}) \; d\mathbf{z}, \tag{4.17}$$

por lo que minimizar $Ln(\mathbf{W})$ respecto de $\mathbf{W}$ es equivalente a minimizar la función de error de un PMC, expresión (4.15). En efecto, el único término en $Ln(\mathbf{W})$ dependiente de $\mathbf{W}$ es el último y éste coincide con la función de error del PMC

$$\frac{1}{2} \int \parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2 \;\; q(\mathbf{z}) \; d\mathbf{z} = \frac{1}{2}\, E_q\left[\parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2\right].$$

Como consecuencia de la deducción anterior podemos modificar el punto de vista del aprendizaje de un PMC: buscamos aprender distribuciones de probabilidad $p(\mathbf{z},\mathbf{W})$ definidas en un espacio de estructura paramétrica. Este espacio corresponde a un espacio de Riemann en el que la distancia local está definida por la divergencia de Kullback–Leibler (4.14) y la matriz de información de Fisher nos da la métrica apropiada a dicho espacio. Las componentes de la matriz de información de Fisher definida por Amari [Amari, 1990] son expresadas de la siguiente forma

$$g_{ij} = E\left[\frac{\partial \log(p(\mathbf{z};\mathbf{W}))}{\partial w_i} \; \frac{\partial \log(p(\mathbf{z};\mathbf{W}))}{\partial w_j}\right],$$

y su notación matricial completa es de la forma

$$\mathbf{G} = E\left[\nabla \log(p(\mathbf{z};\mathbf{W}))\ \ (\nabla \log(p(\mathbf{z};\mathbf{W})))^T\right]. \tag{4.18}$$

De todos los algoritmos de minimización que hemos visto hasta el momento, el único que es válido en el contexto de espacios de Riemann es el algoritmo de descenso por gradiente natural (4.13), que recordaremos aquí

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \eta_t \mathbf{G}^{-1} \nabla e(\mathbf{x}_t, \mathbf{y}_t, \mathbf{W}_t),$$

donde el tensor $\mathbf{G}$ en el espacio de parámetros de la red multicapa es la matriz de información de Fisher definida por (4.18). Se puede apreciar que el algoritmo de descenso por gradiente natural realiza un tratamiento individualizado de las componentes de $\mathbf{W}$, o de dicho de otro modo, es como si se generasen constantes de aprendizaje individualizadas.

4.3.2. Eficiencia Fisher del Gradiente Natural

En esta sección vamos a ver que los pesos óptimos obtenidos mediante el método de descenso por gradiente natural (4.13) responden a estimadores con eficiencia Fisher. Se dice que un estimador estadístico tiene eficiencia Fisher cuando de una forma asintótica se obtiene el mejor resultado posible, es decir aquel cuya varianza sea mínima. Para entender que es un estimador Fisher eficiente, antes revisaremos algunos términos de estadística general [Brandt, 1976]:

Estimador no Sesgado. Diremos que $\hat{\mathbf{W}}_N$ es un estimador no sesgado si para cualquier tamaño de muestra N, su valor esperado es idéntico al valor del parámetro a estimar $\mathbf{W}^*$. En notación matemática podemos expresarlo como $E\left[\hat{\mathbf{W}}_N\right] = \mathbf{W}^*$, o lo que es lo mismo $E\left[\left(\hat{\mathbf{W}}_N - \mathbf{W}^*\right)\right] = 0$.

Estimador Consistente. Diremos que $\hat{\mathbf{W}}_N$ es un estimador consistente si la exactitud del estimador aumenta a medida que aumenta el tamaño de la muestra; expresado en modo matemático resulta la siguiente expresión

$$\lim_{N\to\infty} E\left[\left(\hat{\mathbf{W}}_N - \mathbf{W}^*\right)^T \left(\hat{\mathbf{W}}_N - \mathbf{W}^*\right)\right] = 0.$$

Hemos visto en la sección anterior que un PMC puede asociarse a una familia de funciones distribuidas según $\mathcal{P}_{\mathbf{W}} = p(\mathbf{x}, \mathbf{y}; \mathbf{W})$. La distribución $q(\mathbf{z}) = p(\mathbf{z}, \mathbf{W}^*)$ para el conjunto pesos óptimos $\mathbf{W}^*$ pertenece a dicha familia y además la variable aleatoria $\{\mathbf{Z}_N\}$ es independiente e idénticamente distribuida (i.i.d.) según la distribución q. Sea L la verosimilitud logarítmica muestral

$$\hat{L}_N(\mathbf{W}) = \frac{1}{N} \sum_{n=1}^{N} \log p(\mathbf{z}_n, \mathbf{W})$$

y $\tilde{\mathbf{W}}_N$ el estimador de máxima verosimilitud asociado a $\hat{L}_N$. Por la ley de los grandes números, a medida que el tamaño de la muestra se hace mayor se cumple con mayor certeza que la verosimilitud logarítmica negativa, $\hat{Ln}_N(\mathbf{W}) = -\hat{L}_N(\mathbf{W})$, coincide con la divergencia de Kullback–Leibler $\hat{Ln}_N(\mathbf{W}) \propto d_{KL}(q(\mathbf{x},\mathbf{y}), p_{\mathbf{W}})$ o expresado en términos de probabilidades

$$\lim_{N\to\infty} P(\ \hat{Ln}_N(\mathbf{W};\mathbf{z}_1,\cdots,\mathbf{z}_N) \propto d_{KL}(q(\mathbf{x},\mathbf{y}), p_{\mathbf{W}})\ ) = 1.$$

A partir de este resultado y dado el vector de pesos $\mathbf{W}^*$ procedente de minimizar la divergencia de Kullback–Leibler, podremos afirmar casi con total certeza que el estimador de máxima verosimilitud muestral $\tilde{\mathbf{W}}_N$ converge a $\mathbf{W}^*$; lo que indica que el estimador de máxima verosimilitud $\tilde{\mathbf{W}}_N$ es consistente.

Según el teorema de Cramér–Rao, la varianza de un estimador no sesgado consistente $\hat{\mathbf{W}}_N$ cumple

$$E\left[\left(\hat{\mathbf{W}}_N - \mathbf{W}^*\right)\left(\hat{\mathbf{W}}_N - \mathbf{W}^*\right)^T\right] \geq \frac{1}{N}\,(\mathbf{G}^*)^{-1}, \tag{4.19}$$

donde la matriz $\mathbf{G}^*$ es la matriz de información de Fisher asociada al conjunto de parámetros $\mathbf{W}^*$ y definida según la expresión (4.18).

Un estimador se dice que tiene eficiencia Fisher cuando para tamaños de muestra grande se cumple la igualdad en la expresión (4.19)

$$\lim_{N\to\infty} N\ E\left[\left(\hat{\mathbf{W}}_N - \mathbf{W}^*\right)\left(\hat{\mathbf{W}}_N - \mathbf{W}^*\right)^T\right] = (\mathbf{G}^*)^{-1}. \tag{4.20}$$

Por tanto, se puede afirmar que la inversa de la matriz de información de Fisher (4.18) es la cota que indica si un estimador es Fisher eficiente. Un estimador Fisher eficiente $\hat{\mathbf{W}}_N$ es aquel que asintóticamente tiene la menor varianza posible. La varianza mínima en una estimación es importante si pensamos en la generalización: cuando la varianza es pequeña, existe una menor sensibilidad a variaciones en la muestra y por consiguiente mayor estabilidad en la generalización.

Si el estimador *batch* de máxima verosimilitud $\tilde{\mathbf{W}}_N$, equivalente al estimador que minimiza el error cuadrático a la salida de la red $\sum_{i=1}^{N} \parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2$, es el estimador en el aprendizaje del PMC por el método de descenso por gradiente natural, se demuestra de un modo muy sencillo en [Amari, 1998] que la covarianza del estimador $\mathbf{\Sigma}_N = E\left[\left(\tilde{\mathbf{W}}_N - \mathbf{W}^*\right)\left(\tilde{\mathbf{W}}_N - \mathbf{W}^*\right)^T\right]$ es de la forma

$$\mathbf{\Sigma}_N = \frac{1}{N}(\mathbf{G}^*)^{-1} + \mathcal{O}\left(\frac{1}{N^2}\right).$$

De aquí, se deduce que los pesos $\tilde{\mathbf{W}}_N$ obtenidos por el método de descenso por gradiente natural en entrenamientos de PMCs en modo *batch* son Fisher eficientes, ya que

$$\lim_{N\to\infty} N\ \mathbf{\Sigma}_N = (\mathbf{G}^*)^{-1}.$$

4.3.3. Gradiente Natural y Método de Gauss–Newton

La pregunta de esta sección es: si entrenamos un PMC en modo *batch* utilizando el método de descenso por gradiente natural, ¿realmente estamos utilizando un método nuevo?. En la respuesta veremos la conexión que existe entre el descenso por gradiente natural y el más sencillo de los métodos de segundo orden, el método Gauss–Newton, aplicados ambos al aprendizaje de un PMC en modo *batch*.

Como ya hemos visto para PMCs la densidad de probabilidad $p(\mathbf{z};\mathbf{W})$ está definida por la expresión (4.16), que por comodidad reproduciremos a continuación

$$p(\mathbf{z};\mathbf{W}) = c \;\; q(\mathbf{x}) \;\; \exp\left(-\frac{1}{2} \parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2\right),$$

tomando el logaritmo de la expresión anterior tenemos que

$$\log p(\mathbf{z};\mathbf{W}) = \log c \;\; + \log q(\mathbf{x}) \;\; - \frac{1}{2} \parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2$$

y su gradiente viene dado por

$$\nabla_{\mathbf{W}} \log p(\mathbf{z};\mathbf{W}) = (\mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W})) \;\; \nabla_{\mathbf{W}} \mathbf{f}(\mathbf{x},\mathbf{W}).$$

La matriz de información de Fisher $\mathbf{G}$ definida según la expresión (4.18) será

$$\begin{aligned}
\mathbf{G}(\mathbf{W}) &= E\left[\parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2 \;\; \nabla \mathbf{f}(\mathbf{x},\mathbf{W}) \;\; (\nabla \mathbf{f}(\mathbf{x},\mathbf{W}))^T\right] \\
&= \int \nabla \mathbf{f}(\mathbf{x},\mathbf{W}) \;\; (\nabla \mathbf{f}(\mathbf{x},\mathbf{W}))^T \left\{ \int \parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2 \; p(\mathbf{y}|\mathbf{x}) \; d\mathbf{y} \right\} q(\mathbf{x}) \; d\mathbf{x}.
\end{aligned}$$

El término $\int \parallel \mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}) \parallel^2 \;\; p(\mathbf{y}|\mathbf{x}) \; d\mathbf{y}$, es la varianza del error a la salida de la red. Este término es una constante, pues hemos supuesto que la diferencia $(\mathbf{y} - \mathbf{f}(\mathbf{x},\mathbf{W}))$ es una normal $N(\mathbf{0},\mathbf{I})$; de este modo podemos aproximar $\mathbf{G}$ a la siguiente expresión

$$\begin{aligned}
\mathbf{G}(\mathbf{W}) &\approx \int \nabla \mathbf{f}(\mathbf{x},\mathbf{W}) \;\; (\nabla \mathbf{f}(\mathbf{x},\mathbf{W}))^T \; q(\mathbf{x}) d\mathbf{x} \\
&= E_q\left[\nabla \mathbf{f}(\mathbf{x},\mathbf{W}) \;\; (\nabla \mathbf{f}(\mathbf{x},\mathbf{W}))^T\right] = \mathbf{H}_{Gauss},
\end{aligned}$$

donde $\mathbf{H}_{Gauss}$ es la aproximación de Gauss al hessiano de $\mathbf{f}(\mathbf{x},\mathbf{W})$.

Como conclusión, el entrenamiento en batch de un PMC por el método de descenso por gradiente natural es equivalente al entrenamiento por descenso de Gauss–Newton.

No obstante, debemos decir a favor del método de descenso por gradiente natural que se trata de un método bien fundamentado y es potencialmente aplicable a cualquier contexto; mientras que el método de Gauss–Newton sólo es factible en funciones de error cuadráticas. Por otro lado, el método de Gauss–Newton sólo es viable en entrenamiento en modo *batch*, mientras que el descenso por gradiente natural permite entrenar en modo *on–line*.

4.3.4. Otra aproximación a la matriz de Fisher para PMCs

La matriz de información de Fisher $\mathbf{G}$ es generalmente difícil de calcular; la gran mayoría de las veces no es posible conocer la densidad de probabilidad $p(\mathbf{z};\mathbf{W})$ para completar la ecuación (4.18). Heskes [Heskes, 2000] intenta resolver esta dificultad aproximando la matriz de información de Fisher sin basarse en la densidad de probabilidad. Utiliza para ello una función distancia que actúa entre la salida de la red en la última época efectuada y la salida en la época anterior: $d(\mathbf{f}(\mathbf{x},\mathbf{W}'),\mathbf{f}(\mathbf{x},\mathbf{W}))$. Como esta distancia depende del patrón de entrada $\mathbf{x}$, será necesario promediar sobre la distribución de los patrones de entrada, esto es, $D(\mathbf{W}',\mathbf{W}) = E_{\mathbf{x}}\left[d(\mathbf{f}(\mathbf{x},\mathbf{W}'),\mathbf{f}(\mathbf{x},\mathbf{W}))\right]$. La ventaja de esta función distancia es su independencia respecto de la distribución de las etiquetas $\mathbf{t}$.

Incorporando la aproximación de Heskes para el cálculo de la matriz de Fisher, la variación de pesos en una iteración del entrenamiento en modo *batch* de un PMC corresponde a la ecuación

$$\Delta\mathbf{W} = -\eta \;\; \mathbf{G}_{ap}^{-1}(\mathbf{W}) \;\; E_{\mathbf{x},\mathbf{t}}\left[\frac{\partial d(\mathbf{f}(\mathbf{x},\mathbf{W}),\mathbf{t})}{\partial \mathbf{W}}\right] \tag{4.21}$$

donde

$$\mathbf{G}_{ap}(\mathbf{W}) = \left.\frac{\partial^2 D(\mathbf{W}',\mathbf{W})}{\partial \mathbf{W}' \, \partial \mathbf{W}'^T}\right|_{\mathbf{W}'=\mathbf{W}},$$

es la matriz de Fisher aproximada. Es fácil deducir que se trata del hessiano de la función distancia entre dos iteraciones consecutivas, $D(\mathbf{W}',\mathbf{W}) = d(\mathbf{f}(\mathbf{x},\mathbf{W}'),\mathbf{f}(\mathbf{x},\mathbf{W}))$. La expresión (4.21) es por tanto, una variación del descenso por gradiente natural, en la cual la métrica que se utiliza deja de ser necesariamente la métrica de Riemann. Sólo para funciones distancia del tipo verosimilitud o divergencia de Kullback–Leibler, la matriz $\mathbf{G}_{ap}(\mathbf{W})$ coincide con la matriz de información de Fisher.

Al igual que ocurre con el gradiente natural original, con esta variación el proceso de aprendizaje de la red también se ve acelerado, llegando incluso a obtenerse en algunos casos tiempos de convergencia en el entrenamiento del PMC inferiores a los que se obtienen con la matriz verdadera de Fisher [Heskes, 2000].

En el cálculo de la nueva matriz aproximada es conveniente que las derivadas de la salida $\mathbf{y} = \mathbf{f}(\mathbf{W},\mathbf{x})$ respecto de los pesos desaparezcan. Si realizamos tal aproximación obtenemos que

$$\left.\frac{\partial^2 \, d(\mathbf{y}',\mathbf{y})}{\partial \mathbf{W}' \, \partial \mathbf{W}'^T}\right|_{\mathbf{W}'=\mathbf{W}} \approx \;\; \frac{\partial \, \mathbf{y}}{\partial \mathbf{W}} \;\; \mathbf{\Phi}(\mathbf{y}) \;\; \left(\frac{\partial \, \mathbf{y}}{\partial \mathbf{W}}\right)^T \tag{4.22}$$

donde la matriz $\mathbf{\Phi}$ es la segunda derivada de la función de error con respecto a las salidas, esto es

$$\mathbf{\Phi}(\mathbf{y}) = \left.\frac{\partial^2 d(\mathbf{y}',\mathbf{y})}{\partial \mathbf{y}' \, \partial \mathbf{y}'^T}\right|_{\mathbf{y}'=\mathbf{y}}. \tag{4.23}$$

De esto modo, la matriz de Fisher aproximada nos queda como

$$\mathbf{G}_{ap} = E_x \left[\frac{\partial^2 \, d(\mathbf{y}', \mathbf{y})}{\partial \mathbf{W}' \, \partial \mathbf{W}'^T} \bigg|_{\mathbf{W}'=\mathbf{W}} \right] \; = \; E_x \left[\frac{\partial \, \mathbf{y}}{\partial \mathbf{W}^T} \;\; \mathbf{\Phi}(\mathbf{y}) \;\; \left(\frac{\partial \, \mathbf{y}}{\partial \mathbf{W}} \right)^T \right] \;\; ,$$

En un PMC lo normal es considerar la función distancia como el error cuadrático $d(\mathbf{y}', \mathbf{y}) = \| \, \mathbf{y}' - \mathbf{y} \, \|^2 \, /2$, con lo cual $\mathbf{\Phi}$ coincide con la matriz identidad $\mathbf{I}$, y así la matriz de Fisher para el PMC vendrá dada por

$$\mathbf{G}_{ap} = E_x \left[\frac{\partial \, \mathbf{y}}{\partial \mathbf{W}^T} \;\; \left(\frac{\partial \, \mathbf{y}}{\partial \mathbf{W}} \right)^T \right] ,$$

que si nos fijamos coincide con la aproximación de Gauss (4.2) al hessiano de la salida de la red, $\mathbf{y} = \mathbf{f}(\mathbf{W}, \mathbf{x})$.

4.4. Gradiente Natural en ADnL

En la sección anterior vimos que para un PMC y dentro del espacio de parámetros $\mathbf{W}$ existe equivalencia en la búsqueda del conjunto de parámetros óptimos tanto si ésta se realiza obteniendo los estimadores de máxima verosimilitud como si se realiza minimizando la función criterio del PMC, ver expresión (4.17). En este caso, la matriz de información de Fisher (4.18), $\mathbf{G} = E \left[(\nabla \log(p(\mathbf{z}; \mathbf{W})))^T \;\; \nabla \log(p(\mathbf{z}; \mathbf{W})) \right]$, puede ser vista en términos de la función criterio $\mathbf{G} = E \left[(\nabla_{\mathbf{W}} \, \mathbf{e}(\mathbf{z}; \mathbf{W}))^T \;\; \nabla_{\mathbf{W}} \, \mathbf{e}(\mathbf{z}; \mathbf{W}) \right]$, donde además se cumple que la esperanza del vector aleatorio $\nabla_{\mathbf{W}} \mathbf{e}(\mathbf{z}; \mathbf{W})$ es el gradiente de la función criterio del PMC.

Siguiendo las pautas anteriores, la matriz de información que vamos a definir para la red ADnL parte de la representación del gradiente de la función de error de la red ADnL como la esperanza de una cierta variable aleatoria definida tanto por los términos de la entrada a la red como por los correspondientes pesos asociados, $\mathbf{\Psi} = \mathbf{\Psi}(\mathbf{x}, \mathbf{W})$; lo que viene representado por $\nabla_{\mathbf{W}} J = E \left[\mathbf{\Psi}(\mathbf{x}, \mathbf{W}) \right]$. En analogía a lo ocurrido con el PMC, la forma más natural de definir la matriz de información para ADnL será

$$\boldsymbol{\mathcal{I}}_{ADnL} = E \left[\; \mathbf{\Psi}(\mathbf{x}, \mathbf{W}) \; (\mathbf{\Psi}(\mathbf{x}, \mathbf{W}))^T \right] .$$

Esta definición de la matriz de información de Fisher tiene la ventaja de que no necesita que la función de error a minimizar dependa del etiquetado individual de los patrones de entrenamiento. Sin embargo, el entrenamiento debe realizarse en modo *batch*, pues sí que es necesario conocer el conjunto de los patrones de entrenamiento para poder calcular $\boldsymbol{\mathcal{I}}_{ADnL}$. Con todo esto, la fórmula del descenso por gradiente natural para la red ADnL vendría dada por

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \eta_t \;\; \boldsymbol{\mathcal{I}}_{ADnL}^{-1} \nabla J(\mathbf{W}_t). \tag{4.24}$$

En páginas sucesivas estudiaremos la utilidad de la métrica que J define en el espacio de parámetros. De un modo intuitivo, podemos decir que la matriz de información $\boldsymbol{\mathcal{I}}_{ADnL}$ interviene en el algoritmo de descenso por gradiente natural mejorando el paso adaptativo. Si $\boldsymbol{\mathcal{I}}_{ADnL}$ es una matriz con información relevante, su inversa hará que el paso adaptativo disminuya aminorando la velocidad en el descenso y por consiguiente centrándose en las zonas difíciles del aprendizaje. Si por el contrario, la matriz $\boldsymbol{\mathcal{I}}_{ADnL}$ no aporta demasiada información, su inversa obliga a aumentar los pasos acelerando el transito por las zonas de estancamiento.

La propuesta de esta versión del descenso por gradiente natural en el ADnL fue realizada por primera vez en [Dorronsoro et al., 2001b] aplicándolo al caso sencillo de problemas con sólo dos clases; posteriormente en [Dorronsoro y González, 2002] se amplió para problemas con N clases.

Veremos a continuación que el gradiente de cualquiera de las funciones criterio elegidas en el capítulo 3 para una red ADnL puede ser expresado como el valor esperado de un vector aleatorio $\boldsymbol{\Psi}$ separable en dos partes: una aleatoria $\mathbf{Z}$ y un término que consideraremos constante $\boldsymbol{\mu}$:

$$\frac{\partial J}{\partial W_{kl}} = E\left[\Psi_{kl}\right] = E\left[Z_{kl} - \mu_{kl}\right].$$

Dependiendo de cuál sea la función criterio J, los valores de la variable aleatoria Z_{kl} y el término μ_{kl} serán distintos. A continuación, presentamos sus valores para los tres criterios seleccionados en el capítulo 3.

Función criterio J_1: Razón de determinantes. El gradiente de la función criterio $J_1 = |\mathbf{S}_T|/|\mathbf{S}_B|$ viene dado por la ecuación (3.8)

$$\frac{\partial J_1}{\partial W_{kl}} = J_1 \;\; Tr\left(\mathbf{S}_T^{-1} E\left[\mathbf{D}_{kl} + \mathbf{D}_{kl}^T\right] - \mathbf{S}_B^{-1} \sum_{c=1}^{C} \pi_c \left(\mathbf{D}_{kl}^{\Omega_c} + (\mathbf{D}_{kl}^{\Omega_c})^T\right)\right),$$

donde

$$\begin{aligned} \mathbf{D}_{kl} &= \frac{\partial \mathbf{Y}}{\partial W_{kl}} \left(\mathbf{Y} - \overline{\mathbf{Y}}\right)^T, \\ \mathbf{D}_{kl}^{\Omega_c} &= \frac{\partial \overline{\mathbf{Y}}_c}{\partial W_{kl}} \left(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}}\right)^T. \end{aligned}$$

Así, los valores de Z_{kl} y μ_{kl} para esta función criterio son

$$\begin{aligned} Z_{kl} &= J_1 \;\; Tr\left(\mathbf{S}_T^{-1}(\mathbf{D}_{kl} + \mathbf{D}_{kl}^T)\right), \\ \mu_{kl} &= J_1 \;\; Tr\left(\mathbf{S}_B^{-1} \sum_{c=1}^{C} \pi_c \left(\mathbf{D}_{kl}^{\Omega_c} + (\mathbf{D}_{kl}^{\Omega_c})^T\right)\right). \end{aligned}$$

Simplificaremos la expresión de μ_{kl} definiendo la matriz $\boldsymbol{\Sigma}_{\mathbf{D}_{kl}}$ como $\sum_{c=1}^{C} \pi_c \left(\mathbf{D}_{kl}^{\Omega_c} + (\mathbf{D}_{kl}^{\Omega_c})^T\right)$; de este modo $\mu_{kl} = J_1 \;\; Tr\left(\mathbf{S}_B^{-1} \;\boldsymbol{\Sigma}_{\mathbf{D}_{kl}}\right)$.

Función criterio J_2: Razón de trazas. Para la función criterio $J_2 = Tr(\tilde{\mathbf{\Lambda}}\,\mathbf{S}_T)/Tr(\mathbf{S}_B)$, el gradiente viene dado por la expresión (3.12)

$$\frac{\partial J_2}{\partial W_{kl}} = \frac{2}{Tr(\mathbf{S}_B)} \sum_{p=1}^{C-1} \left(\tilde{\lambda}_p E\left[A_{kl;p}\right] - \tilde{\Lambda}_p\; E\left[B_{kl;p}\right] - J_2 \sum_{c=1}^{C} \pi_c\; \overline{y}_p^{\Omega_c}\; E_c\left[B_{kl;p}\right] \right)$$

donde

$$\begin{aligned} \tilde{\Lambda}_p &= \overline{y}_p\,(\tilde{\lambda}_p - J_2) \\ A_{kl;p} &= y_p\,\frac{\partial y_p}{\partial o_l}\,f'(a_l)\,o_k \\ B_{kl;p} &= \frac{\partial y_p}{\partial o_l}\,f'(a_l)\,o_k \end{aligned}$$

y los valores Z_{kl} y μ_{kl} son los siguientes

$$\begin{aligned} Z_{kl} &= \frac{2}{Tr(\mathbf{S}_B)} \sum_{p=1}^{C-1} \left(\tilde{\lambda}_p\; A_{kl;p} - \tilde{\Lambda}_p\; B_{kl;p} \right), \\ \mu_{kl} &= \frac{2J_2}{Tr(\mathbf{S}_B)} \sum_{p=1}^{C-1} \left(\sum_{c=1}^{C} \pi_c\; \overline{y}_p^{\Omega_c}\; E_c\left[B_{kl;p}\right] \right). \end{aligned}$$

Función criterio $J_3 = 1/Tr(\mathbf{S}_T^{-1}\mathbf{S}_B)$. El gradiente del criterio J_3 dado por la ecuación (3.14)

$$\frac{\partial J_3}{\partial W_{kl}} = {J_3}^2\;\; Tr\left(\mathbf{S}_T^{-1} E\left[\mathbf{D}_{kl} + \mathbf{D}_{kl}^T\right] \mathbf{S}_T^{-1}\mathbf{S}_B \;-\; \mathbf{S}_T^{-1}\mathbf{\Sigma}_{\mathbf{D}_{kl}}\right)$$

se descompone en $E\left[Z_{kl}\right] - \mu_{kl}$ según los términos

$$\begin{aligned} Z_{kl} &= {J_3}^2\;\; Tr\left(\mathbf{S}_T^{-1}(\mathbf{D}_{kl} + \mathbf{D}_{kl}^T)\mathbf{S}_T^{-1}\mathbf{S}_B\right), \\ \mu_{kl} &= {J_3}^2\;\; Tr\left(\mathbf{S}_T^{-1}\mathbf{\Sigma}_{\mathbf{D}_{kl}}\right). \end{aligned}$$

Para cualquiera de las funciones criterio anteriores, la definición de la nueva matriz $\boldsymbol{\mathcal{I}}_{ADnL}$ será la siguiente

$$\boldsymbol{\mathcal{I}}_{ADnL} = E\left[\mathbf{\Psi}\;\mathbf{\Psi}^T\right] \simeq E\left[(\mathbf{Z} - \boldsymbol{\mu})\;(\mathbf{Z} - \boldsymbol{\mu})^T\right] \tag{4.25}$$

En la sección 4.3.3 vimos que en un PMC el descenso por gradiente natural podría considerarse como un procedimiento Gauss–Newton, cosa que no ocurre en la red ADnL, debido a que la función criterio de Fisher J no tiene parecido con la función error cuadrático medio del PMC que es la que produce la similitud entre el gradiente natural para el PMC y el descenso según el método Gauss–Newton. Es evidente que con una red ADnL no es válida la aproximación de Gauss–Newton al hessiano, pues la función a minimizar no es cuadrática.

4.4.1. Complejidad y simplificaciones de la matriz $\mathcal{I}_{ADnL}$

La matriz de Fisher deducida según la expresión (4.25) no parece demasiado complicada, no obstante se trata de una matriz de dimensión $\mathcal{W} \times \mathcal{W}$, siendo $\mathcal{W}$ el número total de pesos en la red. El problema radica en que esta matriz $\mathcal{I}_{ADnL}$ necesita ser calculada e invertida en cada iteración, lo que hace que el coste del entrenamiento se vea encarecido. Si desarrollamos la expresión de $\mathcal{I}_{ADnL}$ (4.25)

$$\mathcal{I}_{ADnL} = E\left[\mathbf{\Psi}\ \mathbf{\Psi}^T\right] \simeq E\left[\mathbf{Z}\ \mathbf{Z}^T\right] - \boldsymbol{\mu}\ E\left[\mathbf{Z}^T\right] - E\left[\mathbf{Z}\right]\ \boldsymbol{\mu}^T + \boldsymbol{\mu}\ \boldsymbol{\mu}^T,$$

nos damos cuenta que el término dominante es el primero, $E\left[\mathbf{Z}\,\mathbf{Z}^T\right]$, con lo cual todas las deducciones que haremos para calcular la complejidad de la matriz $\mathcal{I}_{ADnL}$ se basarán en el cálculo de este término. El coste del cálculo de $\mathcal{I}_{ADnL}$ sin aproximaciones es $\mathcal{O}\left(N\ \mathcal{W}^2\right)$, donde $\mathcal{W}$ es el número total de pesos no lineales $\mathcal{W} = \sum_{h=0}^{H-1}(n_h + 1)\ n_{h+1}$.

En esta sección, realizaremos varias simplificaciones a la matriz de Fisher con el fin de intentar disminuir el coste computacional del entrenamiento de una red ADnL utilizando el gradiente natural como método de minimización para la búsqueda de los pesos óptimos. Las simplificaciones que vamos a realizar se van a basar en tres pautas:

1. La primera simplificación es la independencia por capas de los pesos de la red; con ello la matriz de Fisher se convierte en una matriz diagonal a bloques por capas y la expresión (4.25) se reduce a

   $$\mathcal{I}_{ADnL(kl)(mn)} \approx \left\{E\left[Z^h_{kl}\ Z^{h'}_{mn}\right] - \mu^h_{kl}\ E\left[Z^{h'}_{mn}\right] - E\left[Z^h_{kl}\right]\mu^{h'}_{mn} + \mu^h_{kl}\ \mu^{h'}_{mn}\right\}\delta_{h\,h'},$$

   en donde los índices de la delta de Kronecker $\delta_{h\,h'}$ están referidos a los identificadores de las capas ocultas de la red ($h = 1 \cdots H; h' = 1 \cdots H$). Si sólo tenemos una capa oculta, esta aproximación es intrascendente, puesto que $\delta_{1\,1} = 1$ y la matriz de información es exactamente $\mathcal{I}_{ADnL}$ de la expresión (4.25). En el caso de tener más de una capa la simplificación supone pasar de un coste en el cálculo de $\mathcal{I}_{ADnL}$ de $\mathcal{O}\left(N\ \mathcal{W}^2\right)$ a un coste inferior de

   $$\mathcal{O}\left\{N\left(\sum_{h=0}^{H-1}((n_h+1)\ n_{h+1})^2\right)\right\} \approx \mathcal{O}\left\{N\left(\sum_{h=0}^{H-1}(n_h\ n_{h+1})^2\right)\right\}.$$

   El cálculo de la matriz $\mathcal{I}_{ADnL}$, en cualquiera de los casos, es sencillo debido a que se aprovechan los cálculos efectuados para hallar el gradiente de J necesario en cada iteración del gradiente natural, ecuación (4.24).

2. La segunda simplificación consiste en despreciar correlaciones entre capas contiguas a la hora de calcular las derivadas de $\partial\mathbf{Y}\,/\,\partial\mathbf{W}$ necesarias para

hallar las matrices $\mathbf{D}_{kl}$ correspondientes al peso W_{kl}. En una red ADnL de H capas ocultas, el vector activación de la capa genérica h viene dado por la expresión (2.1), que la recordaremos aquí

$$\mathbf{a}^h = (\mathbf{W}^{h-1})^T \, \mathbf{o}^{h-1}.$$

Usando la regla de la cadena en derivadas parciales, $\partial \mathbf{Y} \,/\, \partial \mathbf{W}$ puede descomponerse en

$$\frac{\partial \mathbf{Y}}{\partial \mathbf{W}^h} = \frac{\partial \mathbf{Y}}{\partial \mathbf{o}^{h+1}} \; \frac{\partial \mathbf{o}^{h+1}}{\partial \mathbf{a}^{h+1}} \; \frac{\partial \mathbf{a}^{h+1}}{\partial \mathbf{W}^h} = \mathbf{\Gamma}^{h+1} \, (\mathbf{o}^h)^T.$$

La simplificación consiste en despreciar correlaciones entre las derivadas

$$\mathbf{\Gamma}^{h+1} = \frac{\partial \mathbf{Y}}{\partial \mathbf{o}^{h+1}} \; \frac{\partial \mathbf{o}^{h+1}}{\partial \mathbf{a}^{h+1}}$$

y las salidas de las neuronas $\mathbf{o}^h$, que como puede apreciarse involucran capas contiguas. Las matrices $\mathbf{\Gamma}^h$ pueden computarse eficientemente usando la regla de retropropagación $\mathbf{\Gamma}^h = \mathbf{\Gamma}^{h+1} \; (\mathbf{W}^h)^T \; f'(\mathbf{a}^h)$, donde f' es la derivada de la función de activación de las neuronas.

Al considerar que no hay correlación entre dos capas contiguas, parece razonable separar términos que realizan medidas en zonas distintas de la red. El término $\mathbf{\Gamma}^h$ mide el impacto en el error de los cambios en las actividades de las unidades ocultas de capas por encima de h; mientras que el término $\mathbf{o}^h$ se ve afectado por las unidades de capas ocultas inferiores a la capa h. La expresión final de $E\left[\mathbf{Z}\,\mathbf{Z}^T\right]$ con esta simplificación podría resumirse como

$$E\left[\mathbf{Z}\,\mathbf{Z}^T\right] \approx \left[\mathbf{O}^h \otimes \mathbf{\Upsilon}^{h+1}\right] \delta_{h\,h'} \tag{4.26}$$

donde el símbolo $\otimes$ indica el producto tensorial, además

$$\mathbf{\Upsilon}^h = E_{\mathbf{x}}\left[\mathbf{\Gamma}^h \, (\mathbf{\Gamma}^h)^T\right] \qquad \text{y} \qquad \mathbf{O}^h = E_{\mathbf{x}}\left[\mathbf{o}^h \, (\mathbf{o}^h)^T\right].$$

Como ejemplo, consideraremos el caso correspondiente al criterio $J_2 = Tr(\tilde{\Lambda}\mathbf{S}_T)/Tr(\mathbf{S}_B)$, al introducir esta aproximación las componentes de la matriz $E\left[\mathbf{Z}\,\mathbf{Z}^T\right]$ tienen la siguiente expresión

$$\begin{aligned} E\left[Z^h_{kl}\, Z^{h'}_{mn}\right] \;\; = \;\; & \delta_{hh'} \, \frac{4}{(Tr(\mathbf{S}_B))^2} \, E\left[o_k \, o_m\right] \\ & E\left[\sum_{p=1}^{C-1} \sum_{q=1}^{C-1} (\lambda_p y_p - \Lambda_p) \, (\lambda_q y_q - \Lambda_q) \, \frac{\partial y_p}{\partial o_l} \, \frac{\partial y_q}{\partial o_n} \, f'(a_l) f'(a_n)\right]. \end{aligned}$$

Como puede apreciarse, para J_2 la matriz $E\left[\mathbf{Z}\,\mathbf{Z}^T\right]$ obtenida con esta simplificación se adapta perfectamente a la expresión (4.26).

En términos de complejidad, al incorporar esta aproximación se produce una disminución notable del coste en el cálculo de $\boldsymbol{\mathcal{I}}_{ADnL}$. Generalizando, la complejidad en el cálculo de $E\left[o_k\ o_m\right]$ y $E\left[\Gamma_l\ \Gamma_n\right]$ es de orden

$$\mathcal{O}\left\{N\ \left(\sum_{h=0}^{H-1}(n_h+1)^2\ +\ n_{h+1}^2\right)\right\} \approx \mathcal{O}\left\{N\ \left(\sum_{h=0}^{H-1} n_h^2\ +\ n_{h+1}^2\right)\right\},$$

cantidad inferior a la necesitada por los anteriores cálculos de $\boldsymbol{\mathcal{I}}_{ADnL}$.

3. Podemos llegar aún más lejos en la simplificación si consideramos que no hay correlación alguna entre los pesos, en ese caso la matriz de información es diagonal total, sólo tenemos los elementos correspondientes a

$$\mathcal{I}_{ADnL(kl)(mn)} = E\left[(Z_{kl}^h - \mu_{kl}^h)\ (Z_{mn}^{h'} - \mu_{mn}^{h'})\right]\ \delta_{h\,h'}\ \delta_{k\,m}\ \delta_{l\,n},$$

donde el coste en el cálculo de $E\left[(Z_{kl}^h)^2\right] \approx E\left[o_k^2\right]\ E\left[\gamma_l^2\right]$ pasa a ser $\mathcal{O}\left(N\ \mathcal{W}\right)$. Como éste es el término dominante, el coste de esta última aproximación para $\boldsymbol{\mathcal{I}}_{ADnL}$ será $\mathcal{O}\left(N\ \mathcal{W}\right)$. Sin duda alguna, esta simplificación disminuye cuantiosamente el cálculo del gradiente natural en cada época, ecuación (4.24) y añade la ventaja de que la inversa de $\boldsymbol{\mathcal{I}}_{ADnL}$ es inmediata

$$\mathcal{I}_{ADnL}{}^{-1}_{(kl)} = \frac{1}{\mathcal{I}_{ADnL(kl)}}\ .$$

4.4.2. Variante Levenberg–Marquardt del descenso por gradiente natural

En la búsqueda de los pesos de la red, si estamos lejos del mínimo de la función criterio es posible que la matriz $\boldsymbol{\mathcal{I}}_{ADnL}$ se convierta en una matriz singular produciéndose un desbordamiento en el método del gradiente natural. Para evitar este problema, podemos implementar una aproximación del algoritmo de Levenberg–Marquardt, de forma que cuando estemos lejos del mínimo sea el descenso por gradiente tradicional el que domine la situación, asegurando un descenso al mínimo aunque éste pueda ser lento. La expresión del gradiente natural introduciendo la versión de Levenberg–Marquardt pasa a ser de la forma

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \eta_t\ \ (\boldsymbol{\mathcal{I}}_{ADnL} + \lambda\mathbf{I})^{-1}\ \nabla J(\mathbf{W}_t),$$

donde la matriz $\mathbf{I}$ es la matriz identidad y λ es el parámetro que controla en cuál de los dos métodos es conveniente trabajar para evitar la situación de explosión del algoritmo.

En cualquier caso, el comportamiento es idéntico al de Levenberg–Marquardt. Para valores grandes de λ estamos frente a un descenso por gradiente standard con la constante de variación pequeña λ^{-1} y con el convencimiento de que no estamos

perjudicando el descenso al mínimo. Para valores pequeños de λ la actualización de pesos es dominada por la variante del descenso por gradiente natural casi puro, nos encontramos en una zona cerca del mínimo y el gradiente natural acelera el proceso de localización del mínimo. De esta forma, obtenemos las ventajas del método de minimización de Levenberg–Marquardt: siempre nos movemos hacia el mínimo sin peligro de que la matriz de información de Fisher sea singular y desestabilice el sistema.

4.5. Problemas con la alternancia de pesos durante el aprendizaje de la red ADnL

En el capítulo anterior, sección 3.2, se explicó cómo cada época del aprendizaje de una red ADnL se efectúa en dos etapas: el cálculo de los pesos no lineales $\boldsymbol{\mathcal{W}}$ manteniendo los pesos lineales del discriminante de Fisher $\mathbf{W}^O$ fijos y posteriormente con estos nuevos pesos $\boldsymbol{\mathcal{W}}$ se recalculan los correspondientes al discriminante. En principio, ésto no parece tener problema alguno, pero con el criterio J_2 sí que existe contrariedades para alguno de los métodos de minimización que se han expuesto en este capítulo.

El criterio $J_2 = Tr(\tilde{\mathbf{\Lambda}}\ \mathbf{S}_T)/Tr(\mathbf{S}_B)$ tiene asociado a los pesos $\mathbf{W}^O$ la matriz diagonal $\tilde{\mathbf{\Lambda}}$, donde el vínculo entre $\tilde{\mathbf{\Lambda}} = \mathbf{\Lambda}/\lambda_1$ y $\mathbf{W}^O$ es que los elementos de $\mathbf{\Lambda}$ son los autovalores de los autovectores $\mathbf{W}^O$ y λ_1 es el mayor de los autovalores.

Para el criterio J_2 y con los métodos de minimización en los que la búsqueda de los pesos no lineales en una época se efectúa minimizando iterativamente hasta alcanzar un mínimo local o global dentro de la época en proceso (métodos tales como Quasi–Newton, gradiente conjugado, Levenberg–Marquardt, minimización en línea, etc.), la presunción de mantener los pesos $\mathbf{W}^O$ fijos deja de ser válida. En las iteraciones internas, la ligadura del conjunto de pesos $\mathbf{W}^O$ fijos y como consecuencia también la matriz $\tilde{\mathbf{\Lambda}}$ hace que se rompa el lazo existente entre ambos: $\mathbf{W}^O$ ya no son los autovectores que minimizan J_2 y la matriz $\mathbf{\Lambda}$ no es su matriz asociada; de lo que se deduce que el nuevo mínimo encontrado no tiene porqué ir en la dirección de minimización global de $J_2(\boldsymbol{\mathcal{W}}, \mathbf{W}^O)$. Para que ésto fuera así habría que recalcular en cada iteración interna el vector $\mathbf{W}^O$ y su matriz de autovalores asociada $\mathbf{\Lambda}$, rompiéndose la conjetura de que los pesos $\mathbf{W}^O$ son fijos mientras se calculan los pesos no lineales $\boldsymbol{\mathcal{W}}$.

Experimentalmente con cualquiera de los métodos anteriormente citados, para el criterio J_2 y manteniendo en cada época los pesos $\mathbf{W}^O$ fijos mientras se calcula los pesos no lineales $\boldsymbol{\mathcal{W}}$, se observa en la evolución de J_2 con el número de épocas que se producen saltos que nos indican un cambio de dirección que se aleja del mínimo. Aunque también se percibe que, a pesar de estos saltos nada afortunados, la tendencia de J_2 en función del número de épocas evoluciona hacia la minimización del criterio J_2. En problemas complicados donde la dimensionalidad del

espacio de entrada es alta, se observa que el número de saltos disminuye, llegando incluso a no darse tales saltos.

En resumen, J_2 no tiene problemas cuando se minimiza paso a paso, como ocurre con el descenso por gradiente y es válido tanto si se elige el gradiente ordinario como el natural. El secreto está en que en cada paso se recalcula $\mathbf{W}^O$ y su matriz asociada $\mathbf{\Lambda}$.

Capítulo 5

Selección de Arquitectura en ADnL

5.1. Introducción

La selección de arquitecturas es una cuestión ampliamente estudiada en PMCs y son varias las técnicas que se han propuesto para intentar resolver este problema; algunas de las más nombradas son *bootstraping*, algoritmos de poda, algoritmos de regularización, criterios de información. Una revisión de las mismas puede encontrarse en [Golden, 1996], [Ripley, 1996] y [White, 1989].

Del mismo modo, en redes del tipo ADnL es igualmente importante definir una arquitectura de red sólida, decidir cuál es su complejidad o lo que es lo mismo, cuántas unidades y capas ocultas se utilizarán en la red.

A grandes rasgos, si el número de unidades es muy bajo, la red resultante no se ajustará excesivamente bien al conjunto de entrenamiento, presentando un marcado sesgo; si, por el contrario, dicho número es muy alto, se tendrá una muy estrecha aproximación al conjunto de entrenamiento y por tanto una mala generalización señalada por una alta varianza en conjuntos de prueba. Esta dualidad entre redes "pequeñas" y "grandes" resulta pues en la conocida dualidad sesgo–varianza [Geman et al., 1992].

En este capítulo, señalaremos métodos para escudriñar redes consistentes con el menor número de unidades posibles. En la búsqueda de la arquitectura óptima, nos vamos a centrar en el término *relevancia* de las unidades de la red. El rastreo en la distinción de unidades como relevantes lo vamos a hacer de dos formas distintas y ello dependerá de que la unidad involucrada pertenezca a capas anteriores a la última capa oculta o bien a ésta misma capa. Recordaremos que los pesos que van a la capa salida son los que se deducen de aplicar un discriminante lineal de Fisher a las salidas de la última capa oculta; en todo momento han sido unos pesos "especiales", dado que han tenido un proceso de cálculo distinto al resto de

los pesos de la red y por consiguiente el cálculo de la relevancia de sus unidades también va a ser distinto.

En el capítulo comenzaremos inicialmente con el estudio de la relevancia en los pesos lineales y continuaremos con los pesos no lineales. Más adelante, en el capítulo 7, donde se muestran los resultados empíricos de la selección de arquitectura, trataremos de fusionar los dos casos de un modo práctico y adaptado a la realidad.

5.2. Relevancia en pesos lineales

En esta sección nos planteamos resolver cómo eliminar unidades no relevantes de la última capa oculta, aquella en la que existe una conexión lineal con la salida de la red. El método que emplearemos se basa en técnicas estadísticas conocidas como MANOVA (Multivariate ANalysis Of VAriance). Para entender cómo se aplica MANOVA es necesario desarrollar un complejo estadístico que iremos introduciendo a lo largo de esta sección. Desglosaremos la sección en las siguientes partes:

1. Mediante distribuciones multidimensionales de Wishart, definiremos el estadístico de Wilks y se verá la relación que existe entre éste y la distribución $\mathcal{F}$ de Fisher.

2. Desarrollaremos del test de hipótesis *razón de máxima verosimilitud* para el caso de funciones de densidad normal. Este test nos va a permitir establecer una relación entre cantidades conocidas o bien estimables de la función de densidad y el estadístico de Wilks.

3. Por último fusionaremos los dos apartados anteriores para detectar características irrelevantes dentro de un conjunto de datos. Como método se utiliza dividir el conjunto de datos en dos subconjuntos distintos: un conjunto representado por las características que son consideradas como relevantes y el otro por las consideradas como irrelevantes. Una vez definido el modelo, se aplica el test estadístico que desarrollaremos en esta sección partiendo de los dos puntos anteriores y se verifica si el modelo es fiable a un cierto nivel de significancia; en el caso de que no lo fuera, éste debe ser rechazado.

5.2.1. Conexión: Wishart–Wilks– Fisher

Basándonos en técnicas de estadística multivariante [Mardia et al., 1989], [Peña, 2002], vamos a buscar la conexión que existe entre variables aleatorias distribuidas como distribuciones multivariantes de Wishart y el conocido test de Fisher.

En el análisis de distribuciones de variables aleatorias unidimensionales muchos de los test estadísticos se basan en distribuciones independientes Chi–cuadrado, χ^2. Una hipótesis ampliamente utilizada en estadística dice que si dos estadísticos muestrales independientes a y b se distribuyen según funciones Chi–cuadrado de la forma $a \sim \sigma^2\chi^2_\alpha$ y $b \sim \sigma^2\chi^2_\beta$, donde σ^2 es la varianza de la distribución y los parámetros α y β son los grados de libertad correspondientes a la distribución Chi-cuadrado, entonces se tiene que la cantidad a/b se distribuye como α/β veces una distribución de Fisher con α y β grados de libertad $\mathcal{F}_{\alpha,\beta}$

$$\mathcal{F}_{\alpha,\beta}(X) = \left(\frac{\alpha}{\beta}\right)^{\frac{1}{2}\alpha} \frac{\Gamma\left(\frac{1}{2}(\alpha+\beta)\right)}{\Gamma\left(\frac{\alpha}{2}\right)\Gamma\left(\frac{\beta}{2}\right)} X^{\left(\frac{\alpha}{2}-1\right)} \left(1+\frac{\alpha}{\beta}X\right)^{-\frac{1}{2}(\alpha+\beta)},$$

donde Γ es la función gamma, definida como

$$\Gamma(\alpha) = \int_0^\infty t^{(\alpha-1)}\, e^{-t}\, dt.$$

A su vez, la cantidad $a/(a+b)$ se distribuye como una función beta con $\alpha/2$ y $\beta/2$ como parámetros. La relación que existe entre una función beta y la función gamma es la siguiente:

$$\mathcal{B}(\alpha,\beta) = \frac{\Gamma(\alpha)\,\Gamma(\beta)}{\Gamma(\alpha+\beta)}.$$

Obsérvese que ninguna de las dos funciones de distribuciones anteriores, la función de Fisher o la función beta, dependen del parámetro desconocido σ, desviación típica de las distribuciones de los estadísticos a y b.

En el caso de distribuciones aleatorias multidimensionales, gran parte de los estadísticos están basados en distribuciones independientes de Wishart, cuya definición es la siguiente:

Definición – Distribución de Wishart ⇒ *Sea* $\mathbf{X} = (\mathbf{x}_1, \cdots, \mathbf{x}_n)^T$ *una matriz normal* $N_d(\mathbf{0}, \mathbf{\Sigma})$*; con ello queremos decir que los* n *vectores que forman la matriz* $\mathbf{X}$ *son independientes e idénticamente distribuidos según una normal* $N_d(\mathbf{0}, \mathbf{\Sigma})$*, y sea la matriz* $\mathbf{M} = \mathbf{X}^T\mathbf{X}$*. La distribución de la matriz simétrica* $\mathbf{M}$ *se conoce como distribución de Wishart* d*–variante, con matriz de escala no singular* $\mathbf{\Sigma}$ *y* n *grados de libertad, donde* $n \geq d$*. Esta distribución se representa como* $\mathcal{W}_d(\mathbf{\Sigma}, n)$*, y su función densidad viene dada por*

$$\mathcal{F}(\mathbf{W}) = \begin{cases} c\, \left|\mathbf{M}^{\frac{n-d-1}{2}}\right| \exp\left(-\frac{1}{2}\, Tr(\mathbf{\Sigma}^{-1}\mathbf{M})\right) & \textit{si } \mathbf{M} \textit{ es definida positiva} \\ \mathbf{0} & \textit{en cualquier otro caso.} \end{cases}$$

La constante c *cumple que:*

$$c^{-1} = 2^{\frac{d\,n}{2}}\ \pi^{\frac{d(d-1)}{4}}\ |\mathbf{\Sigma}|^{\frac{n}{2}}\ \prod_{i=1}^{d} \Gamma\left(\frac{n-i+1}{2}\right).$$

Propiedades singulares de la distribución de Wishart son:

(a) *La esperanza de* $\mathbf{M}$ *es* $n\mathbf{\Sigma}$.

(b) *Si* $d = 1$ *y* $\mathbf{\Sigma} = \sigma^2$, *entonces* $\sigma^{-2}\mathbf{M}$ *se distribuye según una distribución Chi–cuadrado con* n *grados de libertad.*

Según lo visto, la distribución de Wishart se utiliza para representar la distribución muestral de las matrices de covarianza en muestras de variables normales multivariantes. En el caso escalar, la distribución que representa esta incertidumbre es la Chi–cuadrado de Pearson, χ^2. Por tanto, la distribución de Wishart puede considerarse como una generalización multivariante de la Chi-cuadrado. Luego, en concordancia con el caso anterior unidimensional, partimos de las distribuciones de dos estadísticos muestrales independientes $\mathbf{A}$ y $\mathbf{B}$ que se distribuyen como distribuciones de Wishart de dimensión d y varianza $\mathbf{\Sigma}$:

$$\mathbf{A} \sim \mathcal{W}_d(\mathbf{\Sigma}, m)$$
$$\mathbf{B} \sim \mathcal{W}_d(\mathbf{\Sigma}, n),$$

donde m y n son los grados de libertad de cada distribución y se cumple que tanto m como n son $\geq d$.

Si la matriz $\mathbf{A}^{-1}$ existe, serán los autovalores distintos de cero del producto $\mathbf{A}^{-1}\mathbf{B}$ los que van a cobrar interés. Al poder expresarse el producto $\mathbf{A}^{-1}\mathbf{B}$ de la forma cuadrática, $\mathbf{A}^{-1/2}\mathbf{B}\,\mathbf{A}^{-1/2}$, sus autovalores tienen la propiedad de ser positivos y el número de éstos vendrá dado por mín(n, d). Al igual que ocurre con el caso unidimensional, aquí la matriz de varianza $\mathbf{\Sigma}$, en principio desconocida, no tiene efecto sobre la distribución de los autovalores.

Los siguientes teoremas serán necesarios para relacionar las distribuciones de Wishart con test estadísticos de Fisher, del mismo modo que existe relación entre las distribuciones unidimensionales Chi-cuadrado y el test estadístico de Fisher.

Teorema 5.1 *Sea* $\mathbf{A}$ *una matriz aleatoria distribuida como* $\mathbf{A} \sim \mathcal{W}_d(\mathbf{\Sigma}, m)$ *independiente de la matriz aleatoria* $\mathbf{B}$ *distribuida como* $\mathbf{B} \sim \mathcal{W}_d(\mathbf{\Sigma}, n)$, *siendo* $\mathbf{\Sigma}$ *no singular y* m *y* n *no menores a* d; *entonces la cantidad*

$$\phi = \left|\mathbf{A}^{-1}\mathbf{B}\right| = \frac{|\mathbf{B}|}{|\mathbf{A}|}$$

es proporcional al producto de d *variables independientes F, de las cuales la variable* i*–ésima tiene* $(n-i+1)$ *y* $(m-i+1)$ *grados de libertad, con* $(i = 1, \cdots, d)$.

Teorema 5.2 *Sea* $\mathbf{A}$ *una matriz aleatoria distribuida como* $\mathbf{A} \sim \mathcal{W}_d(\mathbf{\Sigma}, m)$ *independiente de la matriz aleatoria* $\mathbf{B}$ *distribuida como* $\mathbf{B} \sim \mathcal{W}_d(\mathbf{\Sigma}, n)$, *siendo* $\mathbf{\Sigma}$ *no singular y* $m \geq d$; *entonces la cantidad*

$$\Lambda = \frac{|\mathbf{A}|}{|\mathbf{A}+\mathbf{B}|} = \left|\mathbf{I}+\mathbf{A}^{-1}\mathbf{B}\right|^{-1} \sim \Lambda(d, m, n)$$

se comporta como una distribución lambda de Wilks con parámetros d, m y n, y se representa como $\Lambda(d, m, n)$.

El valor de Λ *vendrá dado por los* $K = \min(n, d)$ *autovalores de* $\mathbf{A}^{-1}\mathbf{B}$ *distintos de cero*

$$\Lambda = \left|\mathbf{I} + \mathbf{A}^{-1}\mathbf{B}\right|^{-1} = \prod_{k=1}^{K}(1 + \lambda_k)^{-1}.$$

Para el caso particular en el que $d = 1$, existe una relación directa entre la distribución $\Lambda(1, m, n)$ y la $\mathcal{F}$ de Fisher:

$$\frac{1 - \Lambda(1, m, n)}{\Lambda(1, m, n)} \sim \frac{n}{m}\mathcal{F}_{n,m}, \tag{5.1}$$

lo que indica que siempre que se tenga que $d = 1$ y se conozca el valor del estadístico Λ, es posible verificar la fiabilidad del modelo contrastando para un cierto nivel de significancia la cantidad dada por $m\,(1 - \Lambda)/(n\,\Lambda)$, con el valor de la función de distribución de Fisher definida para n y m grados de libertad.

5.2.2. Test de hipótesis razón de máxima verosimilitud

La técnica de MANOVA se basa en el test de hipótesis de la razón de máxima verosimilitud. La estrategia de este test es maximizar la verosimilitud L de la muestra formada por la variable aleatoria $\mathcal{X}$ bajo la influencia tanto de la hipótesis nula H_0 como de la hipótesis alternativa H_1, esto es

$$\rho = \frac{\max\limits_{H_0}\ (L)}{\max\limits_{H_1}\ (L)} = \frac{L_0^*}{L_1^*}. \tag{5.2}$$

Para el caso de una muestra aleatoria $(\mathbf{x}_1, \cdots, \mathbf{x}_n)$ distribuida como una normal $N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, su verosimilitud viene dada por

$$L = \prod_{i=1}^{n} |\boldsymbol{\Sigma}|^{-\frac{1}{2}}\ \ (2\,\pi)^{-\frac{d}{2}}\ \ \exp\left(-\frac{1}{2}\,(\mathbf{x}_i - \boldsymbol{\mu})^T\ \boldsymbol{\Sigma}^{-1}\ (\mathbf{x}_i - \boldsymbol{\mu})\right).$$

En este caso, por sencillez en el manejo de la verosimilitud tomaremos su logaritmo; así la verosimilitud logarítmica para una distribución normal es la siguiente

$$\log L = -\frac{n}{2}\log\left|(2\pi)^d\ \boldsymbol{\Sigma}\right| - \frac{1}{2}\sum_{i=1}^{n}(\mathbf{x}_i - \boldsymbol{\mu})^T\ \boldsymbol{\Sigma}^{-1}\ (\mathbf{x}_i - \boldsymbol{\mu}).$$

La expresión anterior puede ser fácilmente recombinada para obtener el siguiente resultado

$$\log L = -\frac{1}{2}\left[n\ \log\left|(2\pi)^d\ \boldsymbol{\Sigma}\right| + n\ Tr\left(\boldsymbol{\Sigma}^{-1}(\mathbf{S} + \mathbf{d}\mathbf{d}^T)\right)\right], \tag{5.3}$$

donde $\mathbf{d} = (\bar{\mathbf{x}} - \boldsymbol{\mu})$ es la diferencia entre la media muestral y el valor de la media real y $\mathbf{S}$ es la covarianza muestral, expresada como

$$\mathbf{S} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T.$$

Obsérvese que existen dos tipos de parámetros:

1. Los muestrales, como $\bar{\mathbf{x}}$ y $\mathbf{S}$ que son la media y covarianza muestral, respectivamente.

2. Los parámetros de la función de densidad de probabilidad que por lo general son desconocidos. En las formulas anteriores, nos referimos a $\boldsymbol{\mu}$ y $\boldsymbol{\Sigma}$ que representan la media y varianza total de la función de densidad de probabilidad.

Vamos a dar un paso más hacia adelante y buscamos desarrollar el test de hipótesis de máxima verosimilitud para el caso de muestras pertenecientes a C clases, distribuidas según funciones de densidad normales e independientes con varianzas homogéneas, o lo que es lo mismo, se cumple que $\boldsymbol{\Sigma}_1 = \cdots = \boldsymbol{\Sigma}_C$. Continuando con el planteamiento de la verosimilitud logarítmica, tomamos logaritmos a cada lado de la ecuación (5.2) y se obtiene el siguiente resultado

$$\log \rho = \log L_0^* - \log L_1^* = l_0^* - l_1^*, \tag{5.4}$$

donde l_0^* y l_1^* son las verosimilitudes logarítmicas calculadas con el conjunto de estimadores de máxima verosimilitud bajo la influencia de las hipótesis H_0 y H_1, respectivamente.

La función verosimilitud logarítmica para la muestra indicada con C clases, puede deducirse a partir de la ecuación (5.3), resultando ser la siguiente:

$$l = \log L = -\frac{1}{2}\sum_{k=1}^{C}\left[n_k \log\left|(2\pi)^d \boldsymbol{\Sigma}\right| + n_k Tr(\boldsymbol{\Sigma}^{-1}(\mathbf{S}_k + \mathbf{d}_k\mathbf{d}_k^T))\right], \tag{5.5}$$

donde ahora $\mathbf{d}_k = (\bar{\mathbf{x}}_k - \boldsymbol{\mu}_k)$ es la diferencia entre la media muestral y el valor de la media real para la clase k y n_k es el número de patrones de dicha clase. Igual que para el caso de una única normal multivariante, cuando tenemos varias normales se mantienen los dos tipos de parámetros, pero definidos por clases, es decir:

1. Los muestrales, como $\bar{\mathbf{x}}_k$ y $\mathbf{S_k}$ que son la media y covarianza muestral de la clase k-ésima, respectivamente.

2. Los parámetros de la función de densidad de probabilidad. Aquí nos referimos a $\boldsymbol{\Sigma}$ que representa la varianza total de la función de densidad de probabilidad; recordamos que todas las clases tienen la misma varianza, y $\boldsymbol{\mu}_k$ que representa la media para la clase k–ésima.

En estas condiciones, la hipótesis nula a considerar es $H_0 : \boldsymbol{\mu}_1 = \cdots = \boldsymbol{\mu}_C$ con $\boldsymbol{\Sigma}_1 = \cdots = \boldsymbol{\Sigma}_C$, es decir, estamos considerando que todas las clases tienen la misma media y la misma varianza, o lo que es lo mismo, sólo tenemos una clase cuyos estimadores de máxima verosimilitud para la media y la varianza son $\hat{\boldsymbol{\mu}} = \bar{\mathbf{x}}$ y $\hat{\boldsymbol{\Sigma}} = \mathbf{S}$, respectivamente. Así, la máxima verosimilitud logarítmica l_0^* vendría dada por

$$l_0^* = -\frac{n}{2}\,\log\left|(2\,\pi)^d\,\mathbf{S}\right| - \frac{d\,n}{2}, \tag{5.6}$$

donde d y n son respectivamente la dimensión y el tamaño de la muestra. Obsérvese, que bajo la hipótesis nula, el término $\sum_{k=1}^{C} n_k\ Tr(\boldsymbol{\Sigma}^{-1}(\mathbf{S}_k + \mathbf{d}_k\mathbf{d}_k^T))$ de la expresión (5.5) se convierte en

$$\sum_{k=1}^{C} n_k\ Tr(\boldsymbol{\Sigma}^{-1}(\mathbf{S}_k + \mathbf{d}_k\mathbf{d}_k^T)) = \sum_{k=1}^{C} n_k\ Tr(\mathbf{I}) = d\,n.$$

El término $\mathbf{d}_k$ se anula por la condición de máxima verosimilitud, lo que se traduce en que no existe diferencia entre el estimador y el valor real.

La hipótesis alternativa H_1 sería considerar que las medias de las clases son distintas entre sí. En este caso el estimador de máxima verosimilitud de las medias de las clases es $\hat{\boldsymbol{\mu}}_k = \bar{\mathbf{x}}_k$ y el estimador de máxima verosimilitud de la varianza común $\hat{\boldsymbol{\Sigma}} = n^{-1}\,\mathbf{W}$, donde $\mathbf{W} = \sum n_k\mathbf{S}_k$ w es la matriz compuesta por la suma de los productos cuadrados muestrales intra–clases. Considerando estos estimadores, la máxima verosimilitud logarítmica bajo la hipótesis alternativa H_1 es

$$l_1^* = -\frac{n}{2}\,\log\left|\frac{(2\,\pi)^d}{n}\,\mathbf{W}\right| - \frac{d\,n}{2}. \tag{5.7}$$

Bajo la hipótesis H_1, el término $\sum_{k=1}^{C} n_k\ Tr(\boldsymbol{\Sigma}^{-1}(\mathbf{S}_k+\mathbf{d}_k\mathbf{d}_k^T))$ de la expresión (5.5) se convierte en

$$Tr\left(n\,\mathbf{W}^{-1}\sum_{k=1}^{C} n_k\,\mathbf{S}_k\right) = Tr(n\,\mathbf{W}^{-1}\,\mathbf{W}) = Tr(n\,\mathbf{I}) = d\,n.$$

De este modo, la ecuación (5.4) nos queda de la forma

$$\log\rho = \frac{n}{2}\left[\log\left|(2\,\pi)^d\,n^{-1}\,\mathbf{W}\right| - \log\left|(2\,\pi)^d\mathbf{S}\right|\right] = \frac{n}{2}\,\log\frac{|\mathbf{W}|}{|\mathbf{T}|},$$

donde $\mathbf{T} = n\,\mathbf{S} = \sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$ es la matriz muestral de la suma de productos cuadrados respecto de la media global. El valor final de la razón de máximas verosimilitudes vendrá dado por

$$\rho = \left[\frac{|\mathbf{W}|}{|\mathbf{T}|}\right]^{n/2} = \left|\mathbf{T}^{-1}\,\mathbf{W}\right|^{n/2}$$

Como la matriz suma de productos cuadrados inter–clases, $\mathbf{B} = \sum n_k\,(\bar{\mathbf{x}}_k - \bar{\mathbf{x}})(\bar{\mathbf{x}}_\mathbf{k} - \bar{\mathbf{x}})^\mathbf{T}$, está relacionada con $\mathbf{W}$ y $\mathbf{T}$ por $\mathbf{B} = \mathbf{T} - \mathbf{W}$, entonces es posible expresar la razón de máximas verosimilitudes de la siguiente forma

$$\rho^{2/n} = \frac{|\mathbf{W}|}{|\mathbf{B}+\mathbf{W}|} = \left|\mathbf{I} + \mathbf{W}^{-1}\mathbf{B}\right|^{-1}. \tag{5.8}$$

Bajo la hipótesis H_0, cuando la variable $\mathcal{X}$ sigue una distribución normal $N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, según los teoremas de Cochrans y Craig [Mardia et al., 1989] las matrices $\mathbf{T}$, $\mathbf{B}$ y $\mathbf{W}$ son matrices que se distribuyen como distribuciones de Wishart independientes entre sí; la correspondencia entre estas tres matrices y las distribuciones de Wishart es la siguiente

$$\begin{aligned} \mathbf{T} &\sim \mathcal{W}_d(\boldsymbol{\Sigma},\ n-d) \\ \mathbf{W} &\sim \mathcal{W}_d(\boldsymbol{\Sigma},\ n-C-d+1) \\ \mathbf{B} &\sim \mathcal{W}_d(\boldsymbol{\Sigma},\ C-1). \end{aligned}$$

Siempre que se cumpla que $n \geq (d + C)$, estamos en condiciones de aplicar el teorema (5.2) a la expresión (5.8), obteniéndose como resultado

$$\rho^{2/n} \;=\; \left|\mathbf{I} + \mathbf{W}^{-1}\mathbf{B}\right|^{-1} \sim \Lambda(d, n-C-d+1, C-1).$$

El valor de $\rho^{2/n}$ vendrá dado por los $K = min(d, C-1)$ autovalores distintos de cero de la matriz $\mathbf{W}^{-1}\mathbf{B}$ y su expresión final es la siguiente:

$$\rho^{2/n} = \left|\mathbf{I} + \mathbf{W}^{-1}\mathbf{B}\right|^{-1} = \prod_{k=1}^{K}(1+\lambda_k)^{-1}.$$

Al igual que en el apartado anterior, cuando $d = 1$ existe una relación directa entre el valor de $\Lambda(1, n-C-d+1, C-1)$ y la distribución de Fisher $\mathcal{F}_{(C-1)(n-C-d+1)}$, lo que permite verificar al nivel de significancia elegido si el modelo presentado es correcto o no. Este resultado, es el que vamos a utilizar en la siguiente sección con el propósito de discernir cuáles de las unidades de la última capa oculta son irrelevantes.

5.2.3. Test de relevancia de características

A continuación vamos a ver cómo se puede aplicar el test de hipótesis de la razón de máxima verosimilitud para intentar buscar la existencia de características irrelevantes dentro un conjunto de datos $\mathbf{X}$. Para ello, lo primero que supondremos es que la muestra formada por los datos sigue una distribución normal $N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ y dividiremos dicho conjunto en dos subconjuntos $\mathbf{X} = (\mathbf{X}_1^{(r)}, \mathbf{X}_2^{(s)})$. En el primer conjunto de dimensión r están presentes las características relevantes del experimento, mientras que en el segundo conjunto s–dimensional se encuentran aquellas

características que en un principio se suponen irrelevantes. En este apartado vamos a verificar si esta división es realmente correcta o bien debemos contemplar que en el segundo conjunto hay características relevantes.

Al considerar la división de las características en dos grupos, la media y varianza de la distribución también serán repartidas en dos subconjuntos, quedando de la siguiente forma

$$\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \boldsymbol{\mu}_2), \qquad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix},$$

donde $\boldsymbol{\mu}_1$ y $\boldsymbol{\Sigma}_{11}$ son la media y la matriz de covarianza correspondiente a las r características relevantes; del mismo modo que $\boldsymbol{\mu}_2$ y $\boldsymbol{\Sigma}_{22}$ son para las s restantes características. Las matrices $\boldsymbol{\Sigma}_{12}$ y $\boldsymbol{\Sigma}_{21}$ son, en general, matrices no cuadradas de dimensión $(r \times s)$ y $(s \times r)$, respectivamente.

Teorema 5.3 *Si* $\mathbf{X} = (\mathbf{X}_1^{(r)}, \mathbf{X}_2^{(s)}) \sim N_d(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, *entonces la variable* $\mathbf{X}_1$ *y la nueva variable definida como* $\mathbf{X}_{2\cdot1} = \mathbf{X}_2 - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\mathbf{X}_1$ *son estadísticamente independientes y están representadas por las distribuciones*

$$\mathbf{X}_1 \sim N_r(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_{11}) \qquad y \qquad \mathbf{X}_{2\cdot1} \sim N_s(\boldsymbol{\mu}_{2\cdot1}, \boldsymbol{\Sigma}_{22\cdot1})$$

donde los momentos de la distribución de $\mathbf{X}_{2\cdot1}$ *están definidos como*

$$\boldsymbol{\mu}_{2\cdot1} = \boldsymbol{\mu}_2 - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\mu}_1 \qquad y \qquad \boldsymbol{\Sigma}_{22\cdot1} = \boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}.$$

Supongamos un problema de discriminación lineal de dos clases dentro de una muestra formada por dos poblaciones normales d–dimensionales de medias $\boldsymbol{\mu}^{\Omega_1}$ y $\boldsymbol{\mu}^{\Omega_2}$ y la varianza común a las dos $\boldsymbol{\Sigma}$; además se presume que no todas las características de la muestra son relevantes: del total sólo r de ellas son relevantes y las restantes s características son irrelevantes, donde $d = r + s$.

El vector de proyección de máxima verosimilitud viene dado por $\boldsymbol{\alpha} = \boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}$, donde $\boldsymbol{\delta} = \boldsymbol{\mu}^{\Omega_1} - \boldsymbol{\mu}^{\Omega_2}$; de esta forma los nuevos puntos tras la proyección están representados por $\mathbf{x}' = \boldsymbol{\alpha}^T\mathbf{x}$. En la práctica, se trabaja con los estimadores muestrales de máxima verosimilitud: $\boldsymbol{\mu}^{\Omega_1} = \bar{\mathbf{x}}^{\Omega_1}$, $\boldsymbol{\mu}^{\Omega_2} = \bar{\mathbf{x}}^{\Omega_2}$ y $\mathbf{S} = m^{-1}(n_1\,\mathbf{S}^{\Omega_1} + n_2\,\mathbf{S}^{\Omega_2})$, donde $m = n_1 + n_2 - 2$ y las matrices $\mathbf{S}^{\Omega_1}$ y $\mathbf{S}^{\Omega_2}$ son los estimadores de máxima verosimilitud para las varianzas de cada clase

$$\mathbf{S}^{\Omega_1} = \frac{1}{(n_1 - 1)} \sum_{i=1}^{n_1} (\mathbf{x}_i^{\Omega_1} - \boldsymbol{\mu}^{\Omega_1})^T (\mathbf{x}_i^{\Omega_1} - \boldsymbol{\mu}^{\Omega_1})$$

$$\mathbf{S}^{\Omega_2} = \frac{1}{(n_2 - 1)} \sum_{i=1}^{n_2} (\mathbf{x}_i^{\Omega_2} - \boldsymbol{\mu}^{\Omega_2})^T (\mathbf{x}_i^{\Omega_2} - \boldsymbol{\mu}^{\Omega_2}).$$

La diferencia de las medias muestrales será el vector $\mathbf{d} = \bar{\mathbf{x}}^{\Omega_1} - \bar{\mathbf{x}}^{\Omega_2}$ y el vector muestral que define la línea óptima de proyección vendrá dado por $\mathbf{a} = m\mathbf{W}^{-1}\mathbf{d}$. A pesar de ser conscientes de que vamos a trabajar con estimadores muestrales de máxima verosimilitud, continuaremos el desarrollo en la búsqueda de la veracidad del modelo impuesto utilizando los parámetros no muestrales.

Como estamos tratando con dos tipos de características, las relevantes e irrelevantes, se tiene que el vector de proyección puede ser dividido en dos grupos $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1^{(r)}, \boldsymbol{\alpha}_2^{(s)} = \mathbf{0})$, en donde el vector de proyección correspondiente a las características irrelevantes es nulo; desarrollando la expresión del vector $\boldsymbol{\alpha}$ se tiene que

$$\begin{pmatrix} \boldsymbol{\alpha}_1 \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{\delta}_1 \\ \boldsymbol{\delta}_2 \end{pmatrix}.$$

Contemplando la siguiente nomenclatura para la inversa de la varianza

$$\begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}^{-1} = \begin{pmatrix} \boldsymbol{\Sigma}^{11} & \boldsymbol{\Sigma}^{12} \\ \\ \boldsymbol{\Sigma}^{21} & \boldsymbol{\Sigma}^{22} \end{pmatrix}$$

y tomando el resultado citado en [Mardia et al., 1989], las inversas de las matrices no simétricas $\boldsymbol{\Sigma}_{12}$ y $\boldsymbol{\Sigma}_{21}$ se pueden expresar como producto de matrices invertibles

$$\begin{aligned} \boldsymbol{\Sigma}^{12} &= -\boldsymbol{\Sigma}_{11}^{-1}\,\boldsymbol{\Sigma}_{12}\,\boldsymbol{\Sigma}_{22}^{-1} \\ \boldsymbol{\Sigma}^{21} &= -\boldsymbol{\Sigma}_{22}^{-1}\,\boldsymbol{\Sigma}_{21}\,\boldsymbol{\Sigma}_{11}^{-1}. \end{aligned}$$

Al sustituir $\boldsymbol{\Sigma}^{21}$ en la expresión $\boldsymbol{\alpha}_2 = \boldsymbol{\Sigma}^{21}\,\boldsymbol{\delta}_1 + \boldsymbol{\Sigma}^{22}\,\boldsymbol{\delta}_2 = \mathbf{0}$ se llega al resultado

$$\boldsymbol{\delta}_{2\cdot 1} = \boldsymbol{\delta}_2 - \boldsymbol{\Sigma}_{21}\,\boldsymbol{\Sigma}_{11}^{-1}\,\boldsymbol{\delta}_1 = \mathbf{0},$$

donde $\boldsymbol{\delta}_{2\cdot 1}$ representa la diferencia de medias escalada de la clase formada por las variables irrelevantes condicionada a las variables relevantes; lo destacable de esta expresión es que una vez que están bien definidas las características relevantes, ya no importa cuál sea el valor de las características irrelevantes, pues éstas no están aportando información.

Ahora, estamos en condiciones de poder definir la hipótesis nula de un modelo formado por C clases, donde cada población está definida por una distribución normal de igual varianza $\boldsymbol{\Sigma}$; como en el desarrollo anterior también se prevé que la muestra contiene características relevantes e irrelevantes. La hipótesis nula establece que para cada pareja poblacional la diferencia de medias condicionada $\boldsymbol{\delta}_{2\cdot 1}$ es nula, lo que está reflejado en la siguiente expresión matemática

$$H_0 :\ \boldsymbol{\delta}_{2\cdot 1}^{(\Omega_j,\Omega_k)} = (\boldsymbol{\mu}_2^{\Omega_j} - \boldsymbol{\mu}_2^{\Omega_k}) - \boldsymbol{\Sigma}_{21}\,\boldsymbol{\Sigma}_{11}^{-1}\,(\boldsymbol{\mu}_1^{\Omega_j} - \boldsymbol{\mu}_1^{\Omega_k}) = \mathbf{0} \qquad (j \neq k = 1, \cdots, C). \tag{5.9}$$

Como hemos supuesto que todas las clases están formadas por distribuciones normales de la misma varianzas $\mathbf{\Sigma}$, entonces se deduce que las varianzas condicionales son idénticas para todas las clases, es decir, $\mathbf{\Sigma}_{22\cdot 1}^{\Omega_1} = \cdots = \mathbf{\Sigma}_{22\cdot 1}^{\Omega_C}$. Por otro lado, si se observa la ecuación (5.9) se puede apreciar que para que se cumpla la hipótesis nula H_0, sólo es necesario que las medias condicionadas de las clases sean idénticas entre sí; de este modo la hipótesis nula (5.9) puede reescribirse como

$$H_0 : \boldsymbol{\mu}_{2\cdot 1}^{\Omega_1} = \cdots = \boldsymbol{\mu}_{2\cdot 1}^{\Omega_C}, \qquad \text{con} \qquad \mathbf{\Sigma}_{22\cdot 1}^{\Omega_1} = \cdots = \mathbf{\Sigma}_{22\cdot 1}^{\Omega_C}.$$

Según el teorema (5.3), la hipótesis H_0 se enfrenta al caso de una muestra de una única clase formada por las variables condicionadas $\mathbf{x}_{2\cdot 1}$, distribuidas como una normal multidimensional de media $\boldsymbol{\mu}_{2\cdot 1}$ y varianza $\mathbf{\Sigma}_{22\cdot 1}$, esto es $N_s(\boldsymbol{\mu}_{2\cdot 1}, \mathbf{\Sigma}_{22\cdot 1})$.

De este modo, podemos aplicar el estadístico de la razón de máxima verosimilitud de la hipótesis nula frente a la hipótesis alternativa H_1; ésta última implica que las medias condicionales de las clases son distintas entre sí. El valor de la razón vendrá dado por la expresión (5.8)

$$\Lambda s.r = \frac{|\mathbf{W}_{22\cdot 1}|}{|\mathbf{B}_{22\cdot 1} + \mathbf{W}_{22\cdot 1}|} = \left|\mathbf{I} + \mathbf{W}_{22\cdot 1}^{-1}\mathbf{B}_{22\cdot 1}\right|^{-1}.$$

Según hemos visto anteriormente en este capítulo, las matrices de los productos de los cuadrados condicionales $\mathbf{T}_{22\cdot 1}$, $\mathbf{W}_{22\cdot 1}$ y $\mathbf{B}_{22\cdot 1}$ bajo la hipótesis nula H_0 se comportarán como distribuciones de Wishart

$$\begin{aligned}
\mathbf{T}_{22\cdot 1} &= \mathbf{T}_{22} - \mathbf{T}_{21}\,\mathbf{T}_{11}^{-1}\,\mathbf{T}_{12} \sim \mathcal{W}_s(\mathbf{\Sigma}_{22\cdot 1},\ n-d) \\
\mathbf{W}_{22\cdot 1} &= \mathbf{W}_{22} - \mathbf{W}_{21}\,\mathbf{W}_{11}^{-1}\,\mathbf{W}_{12} \sim \mathcal{W}_s(\mathbf{\Sigma}_{22\cdot 1},\ n-C-d+1) \\
\mathbf{B}_{22\cdot 1} &= \mathbf{T}_{22\cdot 1} - \mathbf{W}_{22\cdot 1} \sim \mathcal{W}_s(\mathbf{\Sigma}_{22\cdot 1},\ C-1).
\end{aligned}$$

De todo ello, se deduce que el estadístico $\Lambda s.r$ se comporta como una distribución Lambda de Wilks: $\Lambda s.r \sim \Lambda(s, n-C-d+1, C-1)$. El valor de $\Lambda s.r$ puede expresarse de una forma más simple a partir de la dependencia explícita de los determinantes de las matrices $\mathbf{T}$ y $\mathbf{W}$ y sus submatrices $\mathbf{T}_{11}$ y $\mathbf{W}_{11}$; el resultado es el siguiente

$$\Lambda = \frac{|\mathbf{W}|}{|\mathbf{T}|}, \qquad \Lambda_r = \frac{|\mathbf{W}_{11}|}{|\mathbf{T}_{11}|}, \qquad \Lambda_{s,r} = \frac{\Lambda}{\Lambda_r},$$

donde además se cumple que

$$\begin{aligned}
|\mathbf{W}| &= |\mathbf{W}_{11}|\ |\mathbf{W}_{22\cdot 1}| \\
|\mathbf{T}| &= |\mathbf{T}_{11}|\ |\mathbf{T}_{22\cdot 1}|
\end{aligned}$$

Cuando todas las variables excepto una son relevantes, $s = 1$, el estadístico anterior se puede expresar en función de una distribución $\mathcal{F}$ de Fisher, tal y como vimos en la expresión (5.1). En este caso, se tiene además que $\mathbf{B}_{22\cdot 1}$, $\mathbf{W}_{22\cdot 1}$ y $\mathbf{T}_{22\cdot 1}$

son escalares. En resumen, bajo la hipótesis H_0 de esta sección, con el número de variables irrelevantes igual a 1 y siguiendo la expresión (5.1) se tiene que

$$\mathcal{R} = \frac{n-C-d+1}{C-1} \ \frac{1-\Lambda_{s.r}}{\Lambda_{s.r}} \sim \mathcal{F}_{(C-1),(n-C-d+1)}.$$

Es la cantidad anterior, $\mathcal{R}$, la que nos va a servir para decidir si al eliminar una unidad de la última capa oculta, aquella cuya salida se corresponde con la entrada del discriminante de Fisher, el modelo sigue siendo válido dentro de un determinado nivel de significancia. Eliminar una unidad u de esta última capa oculta implica lo siguiente:

$$\begin{cases} si \quad \mathcal{R} \ \geq \ \mathcal{F}_{(C-1),(n-C-d+1)} \Rightarrow \text{Rechazo la hipótesis} \rightleftharpoons \text{ Unidad Relevante} \\ si \quad \mathcal{R} \ < \ \mathcal{F}_{(C-1),(n-C-d+1)} \Rightarrow \text{Acepto la hipótesis} \rightleftharpoons \text{ Unidad Irrelevante.} \end{cases}$$

5.3. Relevancia en pesos no lineales

Ahora, le toca el turno al estudio de la relevancia en los pesos no lineales. Dado que los primeros estudios en la elección de arquitecturas de redes se hicieron en PMCs, vamos a partir de estas primeras técnicas para introducir posteriormente versiones mejoradas y adaptadas a la red ADnL.

5.3.1. Métodos elementales

La técnica más sencillas en PMCs consiste en la eliminación selectiva de los pesos de la red; métodos tales como OBD (Optimal Brain Damage) [Le Cun et al., 1990] y OBS (Optimal Brain Surgeon) [Hassibi y Stork, 1993] se basan en esta técnica. En concreto, la estrategia que adopta el método OBD es eliminar aquellos parámetros de *saliencia* pequeña; en otras palabras se elimina de forma paulatina los parámetros que afectan en menor grado al error de entrenamiento. La saliencia se calcula a partir de la derivada segunda de la función de error y su definición para un peso genérico W_{kl} es la siguiente

$$Sal_{W_{kl}} = \frac{\partial^2 E}{\partial {W_{kl}}^2} \, W_{kl}^2. \tag{5.10}$$

El método OBS es una mejora al OBD; en OBS se considera la información de la matriz del hessiano al completo para reajustar los pesos que no van a ser eliminados, garantizando que con la nueva modificación de pesos se minimiza el error. La saliencia de los pesos en OBS se mide exactamente igual que en OBD, ecuación (5.10) y una vez más se elimina el peso de menor saliencia; la diferencia reside en que los pesos no eliminados son modificados según la expresión

$$\Delta \mathbf{W}_{kl} = - \ \frac{W_{kl}}{\mathbf{H}^{-1}_{(W_{kl})(W_{kl})}} \ \mathbf{H}^{-1} \, \mathbf{e}_{W_{kl}},$$

donde $\mathbf{e}_{W_{kl}}$ es el vector canónico donde todos los elementos son cero excepto el correspondiente al peso a eliminar W_{kl} cuyo valor es la unidad.

La cuestión de mejora del método OBS frente al OBD es discutible si la red tiene muchos parámetros, pues el cálculo de $\mathbf{H}^{-1}$ se enrevesa a medida que aumenta la complejidad de la red. Para resolver este problema, el cálculo de la inversa de la matriz del hessiano se realiza mediante aproximaciones iterativas al finalizar el entrenamiento de la red. Con ello, se pretende evitar construir la matriz $\mathbf{H}$ real y realizar su inversión esquivando la posibilidad de obtener una matriz singular. Un seguimiento detallado del cálculo de $\mathbf{H}^{-1}$ para un PMC puede seguirse en [Hassibi y Stork, 1993].

Cada vez que se elimina el peso de menor saliencia y en el caso del método OBS después de reajustar el resto de los pesos, se vuelve a entrenar la red con la unidad correspondiente eliminada. La eliminación secuencial del peso con menor influencia en la función de error de la red se reitera hasta llegar a una red tal que con un número mínimo de parámetros es capaz de generalizar eficientemente.

Este tratamiento lo implementamos inicialmente en [Dorronsoro et al., 2000] para la red ADnL y considerando sólo dos clases. Llegamos a la conclusión de que se trata de un método bastante limitado y enfocamos la investigación hacia la búsqueda de métodos que sean más eficaces.

5.3.2. Test de Wald

Otros métodos posibles y con un mayor fundamento para eliminar pesos no relevantes en la red se basan en la realización de test estadísticos de significancia para cada parámetro dentro del modelo de red. Dentro de estos métodos se encuentra el que vamos a analizar a continuación: el test de Wald.

La conexión directa entre el test estadístico de Wald y su aprovechamiento como medida de la relevancia de los pesos de la red parte del siguiente teorema [Manoukian, 1986]:

Teorema 5.4 *Sea* $\mathbf{\Phi}(\mathbf{X}, \mathbf{W})$ *un vector evaluado a partir de una función dependiente de una variable aleatoria* $\mathbf{X}$*, definida según una distribución* $\mathcal{X}$*, y un vector de parámetros* $\mathbf{W}$*, donde además se cumple que tanto el gradiente* $\nabla_{\mathbf{W}}\mathbf{\Phi}(\mathbf{X}, \mathbf{W})$ *como el hessiano* $\nabla^2_{\mathbf{W}}\mathbf{\Phi}(\mathbf{X}, \mathbf{W})$ *existen para cualquier valor* $\mathbf{X}$ *y están acotadas por funciones integrables.*

Si en un punto aislado del espacio de parámetros $\mathbf{W}^*$*, se cumple que* $E\left[\mathbf{\Phi}(\mathbf{X}, \mathbf{W}^*)\right] = 0$ *y las matrices* $\mathbf{G}^* = \mathbf{G}(\mathbf{W}^*) = E\left[\nabla_{\mathbf{W}}\mathbf{\Phi}(\mathbf{X}, \mathbf{W}^*)\right]$ *e* $\mathcal{I}^* = \mathcal{I}(\mathbf{W}^*) = E\left[\mathbf{\Phi}(\mathbf{X}, \mathbf{W}^*)\,\mathbf{\Phi}(\mathbf{X}, \mathbf{W}^*)^T\right]$ *son no singulares y definidas positivas, entonces para la secuencia de vectores aleatorios,* $\mathbf{X}_N$*, independientes e idénticamente distribuidos según* $\mathcal{X}$*, la ecuación* $\sum_1^N \mathbf{\Phi}(\mathbf{X}_N, \mathbf{W}) = 0$ *tiene una secuencia de soluciones* $\widetilde{\mathbf{W}}_N$ *que convergen en probabilidad a* $\mathbf{W}^*$*, de forma que*

$\sqrt{N}(\widetilde{\mathbf{W}}_N - \mathbf{W}^*)$ *converge en distribución a una normal* $N(\mathbf{0}, \mathbf{\Sigma}^*)$ *de media* $\mathbf{0}$ *y varianza* $\mathbf{\Sigma}^* = \mathbf{\Sigma}(\mathbf{W}^*) = (\mathbf{G}^*)^{-1}\, \boldsymbol{\mathcal{I}}^*\, (\mathbf{G}^*)^{-1}$.

Vamos a definir $\mathbf{P}$ como una matriz de dimensiones $q \otimes \mathcal{W}$ con q el número de pesos seleccionados ($q < \mathcal{W}$); a la matriz $\mathbf{P}$ la llamaremos matriz de selección cuya característica es que sus elementos son binarios, 0 ó 1 según la siguiente regla:

$$P_{ij} = \left\{ \begin{array}{ll} 1 & \text{selección del peso a estudio} \\ 0 & \text{ausencia del peso en estudio} \end{array} \right\} \text{ con } \left\{ \begin{array}{l} i = 1, \cdots, q \\ j = 1, \cdots, \mathcal{W}. \end{array} \right.$$

Bajo la hipótesis nula en donde todos los pesos seleccionados por $\mathbf{P}$ tienen el valor de cero, $H_0 : \mathbf{PW}^* = \mathbf{0}$, la cantidad $\sqrt{N}\,\mathbf{P}\,\widetilde{\mathbf{W}}_N$ converge a una distribución de probabilidad normal definida como $N(\mathbf{0},\ \mathbf{P^T}\,\mathbf{\Sigma}^*\,\mathbf{P})$ y por tanto, la variable aleatoria normalizada definida como $\varsigma_N = \sqrt{N}\,(\mathbf{P}^T\,\mathbf{\Sigma}^*\,\mathbf{P})^{-1/2}\,\mathbf{P}\,\widetilde{\mathbf{W}}_N$, converge a la distribución normal $N(\mathbf{0}, \mathbf{I})$. Si tomamos el módulo de ς_N

$$\| \varsigma_N \|^2 = N\,(\mathbf{P}\,\widetilde{\mathbf{W}}_N)^T\,(\mathbf{P}^T\,\mathbf{\Sigma}^*\,\mathbf{P})^{-1}\,(\mathbf{P}\,\widetilde{\mathbf{W}}_N), \tag{5.11}$$

ésta nueva variable aleatoria converge en distribución a una función Chi–cuadrado con q grados de libertad χ^2_q, donde q es el número de vectores seleccionados en la matriz $\mathbf{P}$.

Aplicaremos el resultado anterior para la selección de la arquitectura de la red ADnL. En concreto enfocaremos el estudio a medir la relevancia de unidades, donde cada unidad involucra tantos pesos como unidades tenga la capa siguiente a la de la unidad en estudio. Esta medición de la relevancia de unidades se puede realizar tanto en unidades de la capa de entrada como en unidades de las capas ocultas no lineales.

Relevancia tipo Wald en ADnL

La fusión del test de Wald con la relevancia de unidades no lineales para la red ADnL fue presentada anteriormente en el trabajo [Dorronsoro et al., 2001a] para el caso concreto de dos clases. Hoy en día, ya estamos en condiciones de presentarlo para C clases.

Partimos de que tenemos localizado el conjunto de pesos óptimos de una red ADnL, $\mathbf{W}^*$, y como estamos en un mínimo, el gradiente de la función criterio para ese conjunto de pesos, $\nabla J(\mathbf{W}^*)$, es nulo. Según vimos en la sección 4.4, el gradiente de J podemos expresarlo como una esperanza, de la forma $\nabla J(\mathbf{W}) = E\,[\mathbf{\Psi}] = E\,[\mathbf{Z} - \boldsymbol{\mu}]$. Luego tenemos delante de nosotros, la primera premisa para aplicar un test de Wald: $\nabla J(\mathbf{X}, \mathbf{W}^*) = E\,[\mathbf{\Psi}(\mathbf{X}, \mathbf{W}^*)] = 0$.

Volviendo al teorema de Wald (teorema 5.4), la matriz $\mathbf{G}^*$ allí expuesta no es más que el hessiano de J en el mínimo, $\mathbf{G}^* = E\,[\nabla_{\mathbf{W}}\,\mathbf{\Psi}(\mathbf{X}, \mathbf{W}^*)] =$

$\nabla_{\mathbf{W}} E\left[\mathbf{\Psi}(\mathbf{X}, \mathbf{W}^*)\right] = \nabla^2_{\mathbf{W}} J(\mathbf{W}^*)$, y la matriz $\boldsymbol{\mathcal{I}}^*$ es la matriz de información dada por

$$\boldsymbol{\mathcal{I}}^* = E\left[\mathbf{\Psi}(\mathbf{X}, \mathbf{W}^*)\ (\mathbf{\Psi}(\mathbf{X}, \mathbf{W}^*))^T\right] = E\left[(\mathbf{Z} - \boldsymbol{\mu})(\mathbf{Z} - \boldsymbol{\mu})^{\mathbf{T}}\right].$$

Definiremos la relevancia de la unidad u como el valor de $\| \varsigma_N \|^2$, ecuación (5.11), donde se sustituye la matriz $\mathbf{\Sigma}^*$ por

$$\mathbf{\Sigma}^* = (\nabla^2_{\mathbf{W}} J(\mathbf{W}^*))^{-1}\ E\left[(\mathbf{Z} - \boldsymbol{\mu})(\mathbf{Z} - \boldsymbol{\mu})^{\mathbf{T}}\right]\ (\nabla^2_{\mathbf{W}} J(\mathbf{W}^*))^{-1}$$

y la matriz $\mathbf{P}$ tiene tantas componentes cuyo valor es la unidad como conexiones tenga la unidad u con unidades de la capa siguiente a la suya, en ADnL el número de conexiones es exactamente el número de unidades $\mathcal{K}$ de la capa receptora, así la dimensión de $\mathbf{P}$ será $\mathcal{K} \otimes \mathcal{W}$; el resto de las componentes de la matriz $\mathbf{P}$ son idénticas a cero.

De este modo, decidiremos mantener o eliminar la unidad u en función del valor de su relevancia, ecuación (5.11), cotejándolo con la distribución $\chi^2_{k,(1-\alpha)}$ de k grados de libertad y del nivel de significancia elegido α. Si el valor del estadístico es superior a la cantidad tabulada para la distribución Chi–cuadrado con k grados de libertad y nivel de significancia α, entonces se rechazaría la hipótesis. En nuestro caso, como la hipótesis es que la unidad u sea irrelevante, sólo rechazaríamos dicha hipótesis cuando la unidad en estudio sea relevante. Así pues, el valor del estadístico es tanto mayor cuanto mayor sea la importancia de la unidad que representa. En resumen, se tiene que para cada unidad u :

$$\begin{cases} si & \| \varsigma_N \|^2 \geq \chi^2_{k,(1-\alpha)} \Rightarrow \text{Rechazo la hipótesis} \rightleftharpoons \text{Unidad Relevante} \\ si & \| \varsigma_N \|^2 < \chi^2_{k,(1-\alpha)} \Rightarrow \text{Acepto la hipótesis} \rightleftharpoons \text{Unidad Irrelevante.} \end{cases}$$

5.4. Hessiano de J en ADnL

Para conocer la relevancia de una unidad no lineal es necesario conocer el hessiano; es, sin duda, el momento de desarrollar los hessianos de cada uno de los tres criterios estudiados en este trabajo. Como vamos a mirar la relevancia de los pesos por capas, entonces lo que necesitamos es el hessiano diagonal por bloques, donde cada uno de estos bloques involucra una capa con todos sus pesos. Por tanto, no desarrollaremos el cálculo del hessiano total si no el del diagonal por bloques.

5.4.1. Hessiano del criterio razón de determinantes

Para el cálculo del hessiano de $J_1 = |\mathbf{S}_T| \,/\, |\mathbf{S}_B|$ partimos de su gradiente, expresión (3.5)

$$\frac{\partial J_1}{\partial W_{kl}} = J_1\ Tr\left(\mathbf{S}_T^{-1} \frac{\partial \mathbf{S}_T}{\partial W_{kl}} - \mathbf{S}_B^{-1} \frac{\partial \mathbf{S}_B}{\partial W_{kl}}\right)$$

y recordaremos que la derivada de un matriz simétrica viene dada por la ecuación (3.13)

$$\frac{\partial \mathbf{S}(\mathbf{W})^{-1}}{\partial W_{kl}} = -\mathbf{S}(\mathbf{W})^{-1}\frac{\partial \mathbf{S}(\mathbf{W})}{\partial W_{kl}}\mathbf{S}(\mathbf{W})^{-1}.$$

A partir de las dos expresiones anteriores se obtiene que el hessiano de J_1 para dos pesos genéricos W_{kl} y W_{rs} es el siguiente:

$$\begin{aligned}\frac{\partial^2 J_1}{\partial W_{kl}\,\partial W_{rs}} &= \frac{1}{J_1}\frac{\partial J_1}{\partial W_{kl}}\frac{\partial J_1}{\partial W_{rs}} + \\ &\quad J_1\, Tr\left(-\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W_{rs}}\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W_{kl}} + \mathbf{S}_T^{-1}\frac{\partial^2 \mathbf{S}_T}{\partial W_{kl}\,\partial W_{rs}}\right. \\ &\qquad \left. + \mathbf{S}_B^{-1}\frac{\partial \mathbf{S}_B}{\partial W_{rs}}\mathbf{S}_B^{-1}\frac{\partial \mathbf{S}_B}{\partial W_{kl}} - \mathbf{S}_B^{-1}\frac{\partial^2 \mathbf{S}_B}{\partial W_{kl}\,\partial W_{rs}}\right).\end{aligned}$$

En el caso de evaluar el hessiano en el mínimo, es decir para el conjunto de pesos $\mathbf{W}^*$ donde $\nabla J(\mathbf{W}^*) = \mathbf{0}$, se tiene que el hessiano se simplifica de la siguiente forma:

$$\begin{aligned}\frac{\partial^2 J_1}{\partial W^*_{kl}\,\partial W^*_{rs}} &= J_1\, Tr\left(-\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W^*_{rs}}\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W^*_{kl}} + \mathbf{S}_T^{-1}\frac{\partial^2 \mathbf{S}_T}{\partial W^*_{kl}\,\partial W^*_{rs}}\right. \\ &\qquad \left. + \mathbf{S}_B^{-1}\frac{\partial \mathbf{S}_B}{\partial W^*_{rs}}\mathbf{S}_B^{-1}\frac{\partial \mathbf{S}_B}{\partial W^*_{kl}} - \mathbf{S}_B^{-1}\frac{\partial^2 \mathbf{S}_B}{\partial W^*_{kl}\,\partial W^*_{rs}}\right).\end{aligned}$$

Hacemos hincapié en el hessiano en el mínimo porque es donde la medición de las unidades relevantes se lleva a cabo y según la ecuación (5.11) el hessiano de la función criterio interviene en la medida de la relevancia.

De todos los términos que aparecen en el hessiano, son las derivadas segundas de $\mathbf{S}_B$ y $\mathbf{S}_T$ los términos desconocidos hasta el momento. A partir de las derivadas primeras, ecuaciones (3.6) y (3.7) respectivamente, es fácil deducir las segundas derivadas. Comenzaremos recordando la primera derivada de $\mathbf{S}_T$ para pasar a deducir su segunda derivada:

$$\frac{\partial \mathbf{S}_T}{\partial W_{kl}} = E\left[\frac{\partial \mathbf{Y}}{\partial W_{kl}}\left(\mathbf{Y}-\overline{\mathbf{Y}}\right)^T + \left(\mathbf{Y}-\overline{\mathbf{Y}}\right)\left(\frac{\partial \mathbf{Y}}{\partial W_{kl}}\right)^T\right] = E\left[\mathbf{D}_{kl} + \mathbf{D}_{kl}^T\right],$$

$$\begin{aligned}\frac{\partial^2 \mathbf{S}_T}{\partial W_{kl}\,\partial W_{rs}} &= E\left[\frac{\partial^2 \mathbf{Y}}{\partial W_{kl}\,\partial W_{rs}}\left(\mathbf{Y}-\overline{\mathbf{Y}}\right)^T + \frac{\partial \mathbf{Y}}{\partial W_{kl}}\left(\frac{\partial \mathbf{Y}}{\partial W_{rs}} - \frac{\partial \overline{\mathbf{Y}}}{\partial W_{rs}}\right)^T\right. \\ &\qquad \left. + \left(\frac{\partial \mathbf{Y}}{\partial W_{rs}} - \frac{\partial \overline{\mathbf{Y}}}{\partial W_{rs}}\right)\left(\frac{\partial \mathbf{Y}}{\partial W_{kl}}\right)^T + \left(\mathbf{Y}-\overline{\mathbf{Y}}\right)\left(\frac{\partial^2 \mathbf{Y}}{\partial W_{kl}\,\partial W_{rs}}\right)^T\right] \\ &= E\left[\mathbf{G}_{(kl)(rs)} + \left(\mathbf{G}_{(kl)(rs)}\right)^T\right],\end{aligned}$$

donde definimos la nueva matriz $\mathbf{G}_{(kl)(rs)}$ como

$$\mathbf{G}_{(kl)(rs)} = \frac{\partial^2 \mathbf{Y}}{\partial W_{kl}\, \partial W_{rs}} \left(\mathbf{Y} - \overline{\mathbf{Y}}\right)^T + \frac{\partial \mathbf{Y}}{\partial W_{kl}} \left(\frac{\partial \mathbf{Y}}{\partial W_{rs}} - \frac{\partial \overline{\mathbf{Y}}}{\partial W_{rs}}\right)^T .$$

Realizando lo mismo con $\mathbf{S}_B$ se tiene que :

$$\begin{aligned} \frac{\partial \mathbf{S}_B}{\partial W_{kl}} &= \sum_{c=1}^{C} \pi_c \left(\frac{\partial \overline{\mathbf{Y}}_c}{\partial W_{kl}} \left(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}}\right)^T + \left(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}}\right) \left(\frac{\partial \overline{\mathbf{Y}}_c}{\partial W_{kl}}\right)^T \right) \\ &= \sum_{c=1}^{C} \pi_c \left(\mathbf{D}_{kl}^{\Omega_c} + (\mathbf{D}_{kl}^{\Omega_c})^T \right), \end{aligned}$$

$$\begin{aligned} \frac{\partial^2 \mathbf{S}_B}{\partial W_{kl}\, \partial W_{rs}} &= \sum_{c=1}^{C} \pi_c \left(\frac{\partial^2 \overline{\mathbf{Y}}_c}{\partial W_{kl}\, \partial W_{rs}} \left(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}}\right)^T + \frac{\partial \overline{\mathbf{Y}}_c}{\partial W_{kl}} \left(\frac{\partial \overline{\mathbf{Y}}_c}{\partial W_{rs}} - \frac{\partial \overline{\mathbf{Y}}}{\partial W_{rs}}\right)^T \right. \\ &\qquad \left. + \left(\frac{\partial \overline{\mathbf{Y}}_c}{\partial W_{rs}} - \frac{\partial \overline{\mathbf{Y}}}{\partial W_{rs}}\right) \left(\frac{\partial \overline{\mathbf{Y}}_c}{\partial W_{kl}}\right)^T + \left(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}}\right) \left(\frac{\partial^2 \overline{\mathbf{Y}}_c}{\partial W_{kl}\, \partial W_{rs}}\right)^T \right) \\ &= \sum_{c=1}^{C} \pi_c \left(\mathbf{G}_{(kl)(rs)}^{\Omega_c} + \left(\mathbf{G}_{(kl)(rs)}^{\Omega_c}\right)^T \right), \end{aligned}$$

$$\mathbf{G}_{(kl)(rs)}^{\Omega_c} = \frac{\partial^2 \overline{\mathbf{Y}}_c}{\partial W_{kl}\, \partial W_{rs}} \left(\overline{\mathbf{Y}}_c - \overline{\mathbf{Y}}\right)^T + \frac{\partial \overline{\mathbf{Y}}_c}{\partial W_{kl}} \left(\frac{\partial \overline{\mathbf{Y}}_c}{\partial W_{rs}} - \frac{\partial \overline{\mathbf{Y}}}{\partial W_{rs}}\right)^T .$$

En el desarrollo anterior, tropezamos con que el término derivada segunda nos es desconocido, es decir, no sabemos cuánto vale la derivada segunda de $\mathbf{Y}$ ni la de $\mathbf{Y}_c$. En la sección 3.4.1, referente al cálculo del gradiente de J_1, se obtuvieron las primeras derivadas a partir del cálculo de las derivadas individuales $\partial y_p / \partial W_{kl}$ y $\partial \overline{y}_p^{\Omega_c} / \partial W_{kl}$, donde $p = (1, \cdots, C-1)$ son las unidades de la capa de salida y $\Omega_c = (\Omega_1, \cdots, \Omega_C)$ representa la clase a la que pertenece el patrón de entrada. En dicha sección se llega a la conclusión de que

$$\frac{\partial y_p}{\partial W_{kl}} = \frac{\partial y_p}{\partial a_l} \frac{\partial a_l}{\partial W_{kl}} = \frac{\partial y_p}{\partial a_l}\, o_k,$$

Con esto, la segunda derivada de y_p respecto a dos pesos genéricos queda de la

forma

$$\frac{\partial^2 y_p}{\partial W_{kl}\ \partial W_{rs}} = \underbrace{\left(\frac{\partial}{\partial W_{rs}}\ \frac{\partial y_p}{\partial a_l}\right) o_k}_{\text{Término I}} + \underbrace{\frac{\partial y_p}{\partial a_l}\ \frac{\partial o_k}{\partial W_{rs}}}_{\text{Término II}}.$$

Hasta el momento actual hemos desarrollado el hessiano considerando dos pesos cualesquiera; a partir de este instante enfocaremos el cálculo hacia un hessiano diagonal por bloques, lo que se traduce en que los pesos W_{kl} y W_{rs} pertenecen a la misma capa. Iremos desglosando cada uno de los términos de la expresión anterior con el fin de obtener el hessiano diagonal por bloques lo más ampliado posible. El orden a seguir en el desarrollo va a ser el inverso a cómo están situados los términos.

Término II: Este término no interviene cuando consideramos el hessiano diagonal por bloques: si los pesos W_{kl} y W_{rs} pertenecen a la misma capa, la derivada $\frac{\partial o_k}{\partial W_{rs}}$ es nula. La salida de la unidad k, $o_k = f(a_k)$, se corresponde con la unidad de partida de la conexión W_{kl} y cualquier peso que pertenece a la misma capa que W_{kl} no tiene conexión alguna entre o_k y el peso en cuestión, como consecuencia dicha derivada es nula.

Término I: Hemos llegado a la conclusión de que si el hessiano es diagonal por bloques, este término es el único que interviene en $\frac{\partial^2 y_p}{\partial W_{kl}\ \partial W_{rs}}$, y es el que de forma recursiva involucra las capas que están por encima a la de los pesos W_{kl} y W_{rs}. Desarrollaremos paso a paso la derivada $\frac{\partial}{\partial W_{rs}}\ \frac{\partial y_p}{\partial a_l}$:

$$\frac{\partial}{\partial W_{rs}}\ \frac{\partial y_p}{\partial a_l} = \frac{\partial^2 y_p}{\partial a_l\ \partial a_s}\ \frac{\partial a_s}{\partial W_{rs}} = \frac{\partial^2 y_p}{\partial a_l\ \partial a_s}\ o_r.$$

Si la unidad s pertenece a la última capa oculta, estamos en el caso más sencillo de derivar; dado que $y_p = \sum_{q=1}^{n^H} W_{qp}^O\ o_q$, entonces se tiene que:

$$\frac{\partial y_p}{\partial a_l} = \sum_{q=1}^{n^H} W_{qp}^O\ f'(a_q)\ \delta_{ql} = W_{lp}^O\ f'(a_l) \tag{5.12}$$

$$\frac{\partial^2 y_p}{\partial a_l\ \partial a_s} = W_{lp}^O\ f''(a_l)\ \delta_{ls} \tag{5.13}$$

Como resultado tenemos que la derivada segunda de y_p respecto a los pesos cuando las conexiones no lineales son con la última capa oculta es la siguiente expresión:

$$\frac{\partial^2 y_p}{\partial W_{kl}\ \partial W_{rs}} = \left(\frac{\partial}{\partial W_{rs}}\ \frac{\partial y_p}{\partial a_l}\right) o_k = W_{lp}^O\ f''(a_l)\ o_r\ o_k\ \delta_{ls}.$$

En el caso de que la unidad s pertenezca a cualquier otra capa oculta, entonces tenemos que tener en cuenta la retropropagación procedente de capas superiores a la citada unidad s; de este modo

$$\begin{aligned}
\frac{\partial^2 y_p}{\partial a_l\ \partial a_s} &= \frac{\partial}{a_s}\ \frac{\partial y_p}{\partial a_l} = \frac{\partial}{a_s}\ \left(\sum_m \frac{\partial y_p}{\partial a_m}\ W_{lm}\ f'(a_l)\right) \\
&= \sum_m \frac{\partial}{a_s}\ \left(\frac{\partial y_p}{\partial a_m}\right)\ W_{lm}\ f'(a_l) + \sum_m \frac{\partial y_p}{a_m}\ W_{lm}\ f''(a_l)\ \delta_{ls} \\
&= \sum_{mm'} \frac{\partial^2 y_p}{\partial a_m\ \partial a_{m'}}\ \frac{\partial a_{m'}}{\partial a_s}\ W_{lm}\ f'(a_l) + \sum_m \frac{\partial y_p}{a_m}\ W_{lm}\ f''(a_l)\ \delta_{ls} \\
&= \sum_{mm'} \frac{\partial^2 y_p}{\partial a_m\ \partial a_{m'}}\ W_{sm'}\ W_{lm}\ f'(a_s)\ f'(a_l) + \sum_m \frac{\partial y_p}{a_m}\ W_{lm}\ f''(a_l)\ \delta_{ls},
\end{aligned}$$

donde m y m' pertenecen a la misma capa y ésta es la capa siguiente a la de la unidad s; además el doble sumatorio se realiza sobre todas las unidades de dicha capa.

Finalmente, la expresión general de la derivada segunda de y_p respecto de los pesos viene dada por:

1. Cuando los pesos W_{kl} y W_{rs} conectan con la última capa oculta

$$\frac{\partial^2 y_p}{\partial W_{kl}\ \partial W_{rs}} = W^O_{lp}\ f''(a_l)\ o_r\ o_k\ \delta_{ls}.$$

Obsérvese que este término es distinto de cero sólo cuando los pesos W_{rs} y W_{kl} conectan la misma unidad receptora.

2. Para cualquier otro par de pesos procedentes de conexiones no lineales distintas a las de la última capa oculta se recurre a la siguiente fórmula de retropropagación

$$\begin{aligned}
\frac{\partial^2 y_p}{\partial W_{kl}\ \partial W_{rs}} &= \sum_{mm'} \frac{\partial^2 y_p}{\partial a_m\ \partial a_{m'}}\ W_{sm'}\ W_{lm}\ f'(a_s)\ f'(a_l)\ \ o_r\ o_k \\
&+ \sum_m \frac{\partial y_p}{a_m}\ W_{lm}\ f''(a_l)\ \ o_r\ o_k\ \delta_{ls},
\end{aligned}$$

donde el caso base de $\frac{\partial^2 y_p}{\partial a_m\ \partial a_{m'}}$ y $\frac{\partial y_p}{\partial a_m}$ son las ecuaciones (5.13) y (5.12), respectivamente.

Hasta aquí, ya tenemos cómo se construye la matriz $\frac{\partial^2 \mathbf{Y}}{\partial W_{kl}\ \partial W_{rs}}$ enfocada al caso diagonal por bloques; esta matriz es necesaria para el cálculo de las distintas matrices $\mathbf{G}_{(kl)(rs)}$, que a su vez se necesitan para el cálculo del hessiano.

Las matrices condicionadas a las clases $\mathbf{G}^{\Omega_c}_{(kl)(rs)}$, que también forman parte del cálculo del hessiano, se obtienen a través de la siguiente esperanza

$$\frac{\partial^2 \overline{y}_p^{\Omega_c}}{\partial W_{kl}\, \partial W_{rs}} = E\left[\frac{\partial^2 y_p}{\partial W_{kl}\, \partial W_{rs}} \,|\mathbf{Y} \in \Omega_c\right].$$

Con todo esto, ya estamos en condiciones de calcular el hessiano del criterio J_1 y además tenemos preparado el camino para el cálculo de los otros dos criterios, que será lo que haremos en los próximos apartados.

5.4.2. Hessiano del criterio razón de trazas

A partir de la derivada de J_2, ecuación (3.9), es fácil deducir la expresión del hessiano de J_2

$$\begin{aligned} J_2 &= \frac{Tr(\tilde{\mathbf{\Lambda}}\, \mathbf{S}_T)}{Tr(\mathbf{S}_B)}, \\ \frac{\partial J_2}{\partial W_{kl}} &= \frac{1}{Tr(\mathbf{S}_B)}\left\{Tr\left(\tilde{\mathbf{\Lambda}}\frac{\partial \mathbf{S}_T}{\partial W_{kl}}\right) - J_2\, Tr\left(\frac{\partial \mathbf{S}_B}{\partial W_{kl}}\right)\right\}, \end{aligned}$$

$$\begin{aligned} \frac{\partial^2 J_2}{\partial W_{kl}\, \partial W_{rs}} &= \frac{1}{Tr(\mathbf{S}_B)}\left\{-\frac{\partial J_2}{\partial W_{kl}}\, Tr\left(\frac{\partial \mathbf{S}_B}{\partial W_{rs}}\right) - \frac{\partial J_2}{\partial W_{rs}}\, Tr\left(\frac{\partial \mathbf{S}_B}{\partial W_{kl}}\right)\right. \\ &\qquad \left. + Tr\left(\tilde{\mathbf{\Lambda}}\, \frac{\partial^2 \mathbf{S}_T}{\partial W_{kl}\, \partial W_{rs}}\right) - J_2\, Tr\left(\frac{\partial^2 \mathbf{S}_B}{\partial W_{kl}\, \partial W_{rs}}\right)\right\}. \end{aligned}$$

En el mínimo, el hessiano de J_2 se reduce a:

$$\frac{\partial^2 J_2}{\partial W^*_{kl}\, \partial W^*_{rs}} = \frac{1}{Tr(\mathbf{S}_B)}\left\{Tr\left(\tilde{\mathbf{\Lambda}}\, \frac{\partial^2 \mathbf{S}_T}{\partial W^*_{kl}\, \partial W^*_{rs}}\right) - J_2\, Tr\left(\frac{\partial^2 \mathbf{S}_B}{\partial W^*_{kl}\, \partial W^*_{rs}}\right)\right\}.$$

Todos los términos que aparecen en el hessiano de J_2, han aparecido antes en el hessiano de J_1, luego ya nos son conocidos.

Como cabría esperar, el hessiano de J_2 es más sencillo que el correspondiente a J_1. Los términos nuevos en los dos hessianos son los mismos, sin embargo en el hessiano de J_1 se incrementa la cantidad de cálculo debido a que el número de productos de matrices es superior.

5.4.3. Hessiano del criterio J_3

Seguiremos las mismas pautas que con los dos criterios anteriores para hallar el hessiano del criterio $J_3 = 1/Tr(\mathbf{S}_T^{-1}\mathbf{S}_B)$. Partiendo de la expresión del gradien-

te (3.14)

$$\frac{\partial J_3}{\partial W_{kl}} = {J_3}^2 \; Tr\left(\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W_{kl}}\mathbf{S}_T^{-1}\mathbf{S}_B \; - \; \mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_B}{\partial W_{kl}}\right),$$

el valor del hessiano es el siguiente

$$\begin{aligned}\frac{\partial^2 J_3}{\partial W_{kl}\,\partial W_{rs}} \;=\; & \frac{2}{J_3}\,\frac{\partial J_3}{\partial W_{kl}}\,\frac{\partial J_3}{\partial W_{rs}} \\ & + \; {J_3}^2 \left\{ Tr\left(-\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W_{rs}}\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W_{kl}}\mathbf{S}_T^{-1}\mathbf{S}_B \; + \; \mathbf{S}_T^{-1}\frac{\partial^2 \mathbf{S}_T}{\partial W_{kl}\,\partial W_{rs}}\mathbf{S}_T^{-1}\mathbf{S}_B \right.\right. \\ & \qquad - \; \mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W_{kl}}\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W_{rs}}\mathbf{S}_T^{-1}\mathbf{S}_B \; + \; \mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W_{kl}}\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_B}{\partial W_{rs}} \\ & \qquad \left.\left. + \; \mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_T}{\partial W_{rs}}\,\mathbf{S}_T^{-1}\frac{\partial \mathbf{S}_B}{\partial W_{kl}} \; - \; \mathbf{S}_T^{-1}\frac{\partial^2 \mathbf{S}_B}{\partial W_{kl}\,\partial W_{rs}} \right) \right\}.\end{aligned}$$

En este caso, en el mínimo de J_3 también se produce una simplificación en el cálculo del hessiano, pero ésta es menor que en los dos casos anteriores, se tiene que sólo el primer término del hessiano es nulo, $\frac{\partial J_3}{\partial W^*_{kl}}\,\frac{\partial J_3}{\partial W^*_{rs}} = 0$.

Como puede apreciarse el hessiano del criterio J_3 es el más costoso de los tres vistos, aunque en sí el material necesario es el mismo para los tres criterios.

Parte II

Resultados Empíricos

Capítulo 6

Convergencia y Clasificación: Resultados Empíricos

6.1. Planteamiento General

En este capítulo es donde vamos a presentar las conclusiones experimentales que hemos obtenido sobre la aceleración de convergencia probando con diversos conjuntos de datos procedentes de la base de datos *UCI Machine Learning*, accesible vía web en la dirección **http://www.ics.uci.edu/∼mlearn/**.

Con cada uno de los tres criterios estudiados en el capítulo 3 veremos su comportamiento ante distintos métodos de entrenamiento, aquellos que han sido presentado en el capítulo 4. Los experimentos que vamos a realizar constan de un número X de pruebas, cada una de ellas inicializada con pesos aleatorios distintos y se obtiene el comportamiento medio de estas X repeticiones junto con su desviación típica. El número de repeticiones X va a depender de la complejidad del conjunto de entrenamiento.

Los experimentos que hemos realizado tratan de responder a la pregunta: *¿es la métrica natural más eficiente que la métrica euclídea?*. Buscamos la respuesta analizando la evolución del criterio en función del número de épocas, usando como método de minimización el descenso por gradiente tanto ordinario como natural. Además recurrimos a dos posibles modelos: el simple o época a época y el que usa minimización en línea dentro de cada época. En una segunda parte, nos centraremos en el rendimiento efectivo de cada una de las métricas.

Por último, haremos un resumen que engloba la evaluación de la combinación: criterio y método de minimización, desde el punto de vista del error cometido al clasificar los patrones de test (si procede) o de entrenamiento para el caso de contar con un único conjunto.

En todas las gráficas que se van a presentar en este capítulo, se parte de la primera época ya finalizada, es decir, no presentamos las condiciones iniciales; que

por otro lado son las mismas para todos los experimentos del mismo criterio. Esta decisión se ha tomado con el fin de poder apreciar con mayor claridad la zona de convergencia, que consideramos más importante que el salto cuantitativo que se produce en la primera iteración.

6.2. Conjunto Iris

Comenzaremos la presentación con el conocido conjunto de datos *Iris* de Fisher. Estos datos ya nos son familiares, pues hemos trabajado con ellos en el capítulo 1.

6.2.1. Minimización del criterio durante el entrenamiento

Descenso por gradiente simple

Cada uno de los experimentos que se van a presentar aquí están realizados con 20 repeticiones y lo que se representa es un promedio de esas 20 repeticiones. Las figuras (6.1), (6.2) y (6.3) corresponden a las pruebas con cada uno de los tres criterios, J_1, J_2 y J_3, respectivamente, tomando como método de entrenamiento el descenso por gradiente con los dos tipos de métrica: euclídea y natural. Los pesos iniciales en cada prueba son idénticos tanto para el gradiente ordinario como para el gradiente natural.

Las tres figuras representan para cada criterio el promedio de su evolución en función del número de épocas junto con la desviación típica; es decir, cada línea de evolución está representada por tres trayectoria: la interna es el promedio y las dos externas las que delimitan la desviación típica ($\bar{J} \pm \sigma$). La figura (6.1) consta de dos gráficas, la superior representa la evolución general del criterio J_1 con el gradiente ordinario y el natural; en el caso del gradiente natural se representa también el desglose producido por la desviación típica. La inferior es un detalle de la anterior donde el desglose de las desviaciones se hace para los dos gradientes.

En general, se puede apreciar que en la evolución de los criterios la variación existente entre las 20 pruebas es relativamente grande al principio y a medida que aumenta el número de épocas esta variación disminuye considerablemente; ésto hace que se aprecie el efecto de la convergencia.

Comparando descenso por gradiente ordinario con natural, observamos que con el último de los gradientes es con el que se obtienen mejores resultados. A pesar de ser más costoso que el gradiente ordinario, es cierto que se necesitan muchas menos iteraciones para alcanzar la asíntota óptima. En las ilustraciones aquí presentadas hemos llegado, en cualquiera de los casos, hasta 2000 épocas; pero sólo con vistas a la comparación entre los dos métodos de descenso. Observando las figuras (6.1), (6.2) y (6.3) se puede apreciar que aproximadamente con 300 iteraciones el descenso por gradiente natural ya ha alcanzada la asíntota; mientras

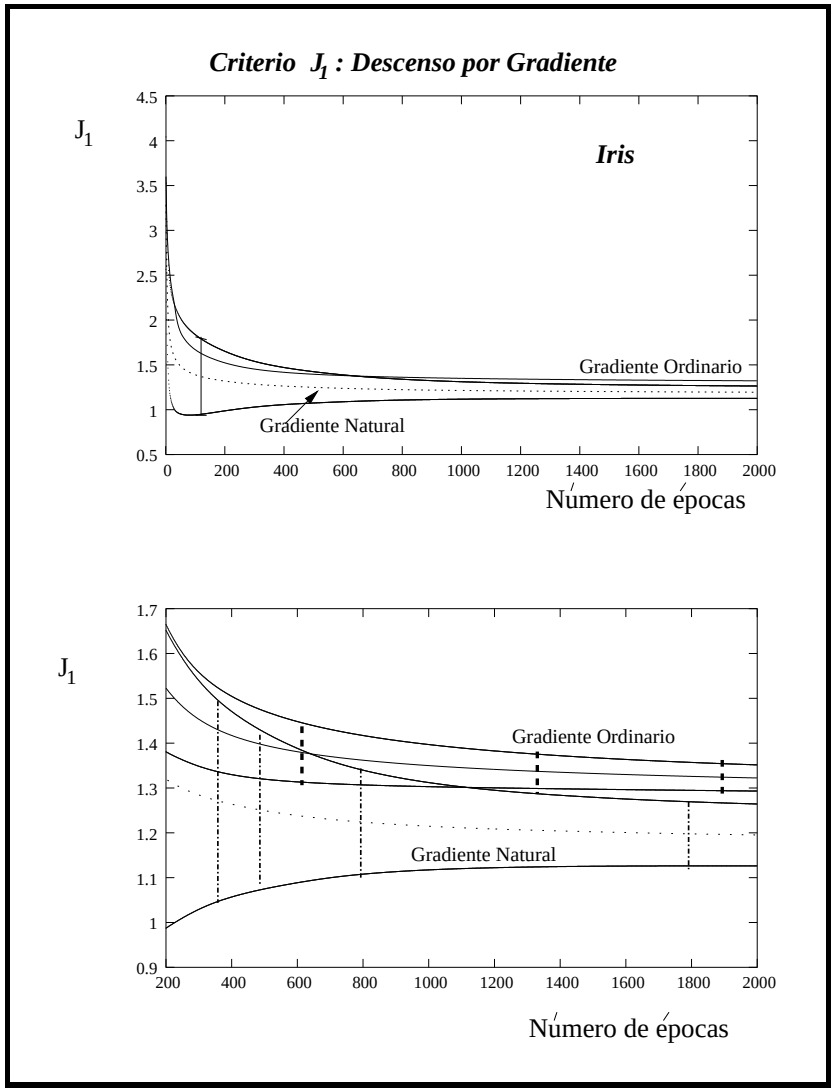


Figura 6.1: Iris $\Rightarrow$ Descenso por gradiente con el criterio J_1

que el descenso por gradiente ordinario precisa un número de iteraciones bastante más alto y se queda en un mínimo local superior al alcanzado por el gradiente natural.

Ahora, haremos un análisis cuantitativo del coste en el cálculo de los dos tipos de gradiente para una arquitectura como la que hemos utilizado aquí. En una red de una sola capa oculta el coste del gradiente es del orden $\mathcal{O}(N\,C^2\,D\,K)$ para los criterios J_1 y J_3 y $\mathcal{O}(N\,C\,D\,K)$ para el criterio J_2, ver sección 3.5, donde N es el número de patrones de entrenamiento, C el número de clases, D el número de atributos y K el número de unidades en la capa oculta. El cálculo de la matriz de información $\boldsymbol{\mathcal{I}}_{ADnL}$ para el gradiente natural supone un coste de $\mathcal{O}(N\,\mathcal{W}^2) = \mathcal{O}(N\,D^2\,K^2)$, ver sección 4.4.1. Luego, el descenso por gradiente natural para

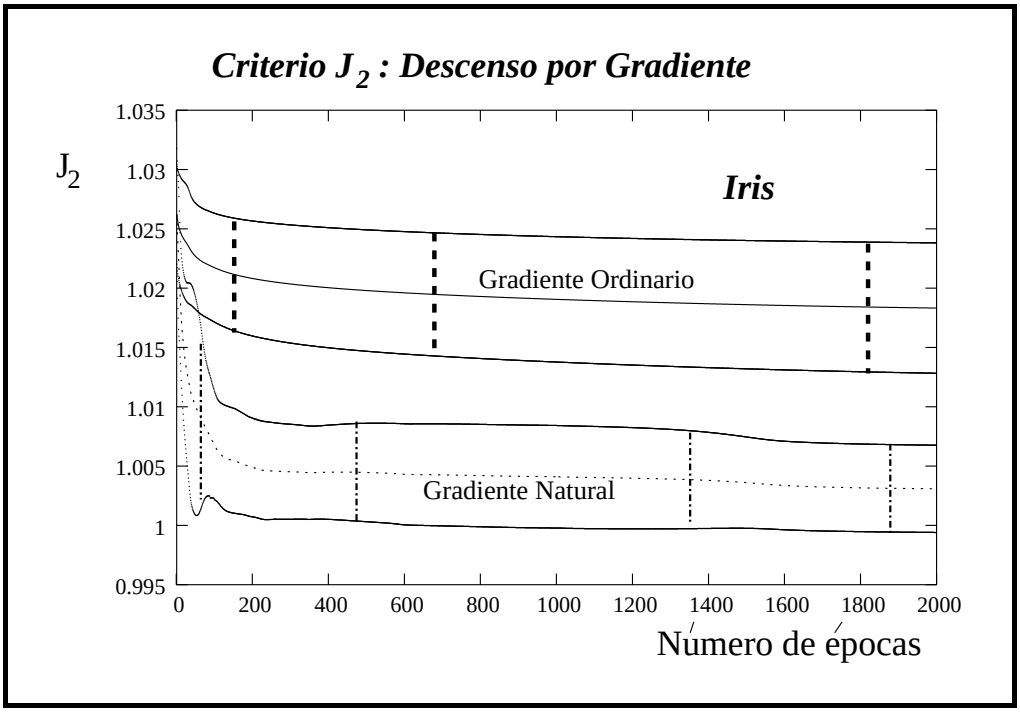


Figura 6.2: Iris ⇒ Descenso por gradiente con el criterio J_2

cada iteración implica un coste $(D\ K)/C^2$ veces mayor que el descenso ordinario para los criterios J_1 y J_3 y $(D\ K)/C$ para el criterio J_2. En el caso que nos ocupa $K = 5$, $D = 4$ y $C = 3$, ésto supone un coste por iteración 20/9 veces mayor en el gradiente natural que en el ordinario para J_1 y J_3 y 20/3 para J_2. En el cuadro (6.1) se presentan los valores J promediados junto con su desviación para el descenso por gradiente ordinario con 2000 épocas y el equivalente en el coste con el gradiente natural con 900 épocas para J_1 y J_3 y 300 épocas para J_2.

	$J \pm \sigma$	
Criterio	Gradiente Ordinario (2000 épocas)	Gradiente Natural
J_1	1,322 ± 0,029	1,219 ± 0.105 (900 épocas)
J_2	1,018 ± 0,005	1,004 ± 0.004 (300 épocas)
J_3	0,574 ± 0,008	0,535 ± 0,011 (900 épocas)

Cuadro 6.1: Iris ⇒ Gradiente natural y ordinario a igual carga computacional

A la vista del cuadro (6.1), que muestra los valores del criterio con los dos tipos de descenso por gradiente a igual coste computacional, el descenso por gradiente natural es más eficiente que el gradiente simple.

Un factor importante de cara al entrenamiento tipo descenso por gradiente

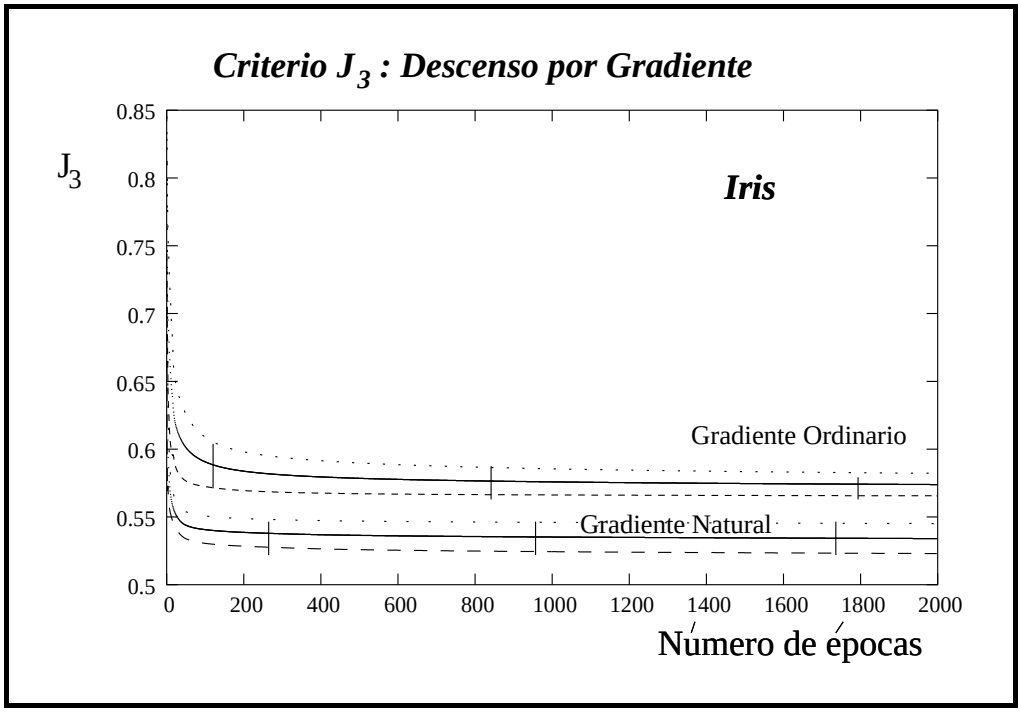


Figura 6.3: Iris $\Rightarrow$ Descenso por gradiente con el criterio J_3

es la constante de aprendizaje η, esta constante es un parámetro que ajustamos mediante el método de *prueba y error*, además esta constante η es sensible tanto al método de descenso como al criterio que se esté minimizando. Hemos realizado experimentos con distintas constantes y son las expuestas en el cuadro (6.2) las correspondientes a los mejores resultados. Se observó en las distintas simulaciones que mantener constante η no era una buena opción y decidimos comenzar por un valor inicial, que es el presentado en el cuadro (6.2), y disminuir éste ligeramente en cada iteración según la siguiente proporción $\eta_{t+1} = \eta_t(1 - \eta_t)$, de tal modo que al principio del entrenamiento permitimos mayor variación de un paso al siguiente, esto quiere decir que al principio aprendemos más deprisa, mientras que al final como en todo aprendizaje pequeños avances implican un gran esfuerzo que en nuestro caso el aumento del esfuerzo se traduce en realizar pasos más pequeños pero que nos aseguran la convergencia al mínimo. Todos los resultados que hemos expuesto y expondremos a lo largo de este trabajo corresponden a esta forma de operar con la constante de aprendizaje.

Descenso por gradiente con minimización en línea

Una variante del descenso por gradiente es utilizar en cada época un método de minimización en línea, esta versión es la que vamos a tratar en esta sección. En cada época, se realiza una búsqueda del mínimo que conlleva a su vez a realizar re-

Criterio	η_0 = *Constante de aprendizaje inicial*	
	Descenso Ordinario	Descenso Natural
J_1	10^{-3}	10^{-1}
J_2	10^{-1}	10^{-2}
J_3	10^{-1}	10^{-1}

Cuadro 6.2: Iris ⇒ Constante de aprendizaje en el descenso por gradiente simple

peticiones del proceso de minimización hasta llegar a la condición de convergencia intrínseca de la época en curso. Mencionaremos aquí que con el criterio J_2, éstos métodos de minimización no son estrictamente aplicables, revisar sección 4.5.

El comportamiento en conjunto de esta versión de descenso por gradiente es una repetición del análisis con el descenso por gradiente paso a paso: el gradiente ordinario de nuevo se queda en un mínimo por encima del gradiente natural. Una vez más se observa que la métrica de Riemann es más eficiente que la métrica euclídea. Las figuras (6.4) y (6.5) corresponden a la evolución de los dos criterios J_1 y J_3 respectivamente, para los dos tipos de métricas: euclídea y de Riemann. Como método de minimización en línea hemos usado uno de los presentes en el libro [Press et al., 1992], en concreto el que utiliza el gradiente para buscar la línea óptima de descenso. En la figura (6.4) se ha representado el gradiente ordinario como el promedio, mientras que para el natural se representa el promedio y las cotas producida por la desviación. Obsérvese que el promedio del gradiente ordinario está por encima de la cota superior del gradiente natural. En el caso del criterio J_3, figura (6.5), se puede distinguir las trayectorias acotadas de los dos descensos por gradiente y se observa que las cotas de los dos gradientes no interfieren entre sí, el gradiente ordinario siempre está por encima del gradiente natural.

Como cabe esperar, con el descenso por gradiente con minimización en línea se obtienen mínimos más bajos que los obtenidos con el descenso por gradiente paso a paso; es decir mejoramos en la búsqueda de la solución óptima. Con minimización en línea se han empleado sólo 500 épocas; pero no nos engañemos, el tiempo consumido en cada minimización parcial es elevado.

Otros métodos de minimización de segundo orden: Quasi–Newton y Gradiente Conjugado

Sin duda alguna, estos dos métodos de minimización vistos en el capítulo 4 son los más rápidos y eficaces para el ejemplo que nos ocupa: necesitan muy pocas iteraciones para llegar a la convergencia y alcanzan mínimos más bajos. Es el gradiente conjugado el que optimiza mejor; aunque es difícil decantarse por uno de los dos, pues los dos métodos van muy a la par, ver figura (6.6). Ciertamente,

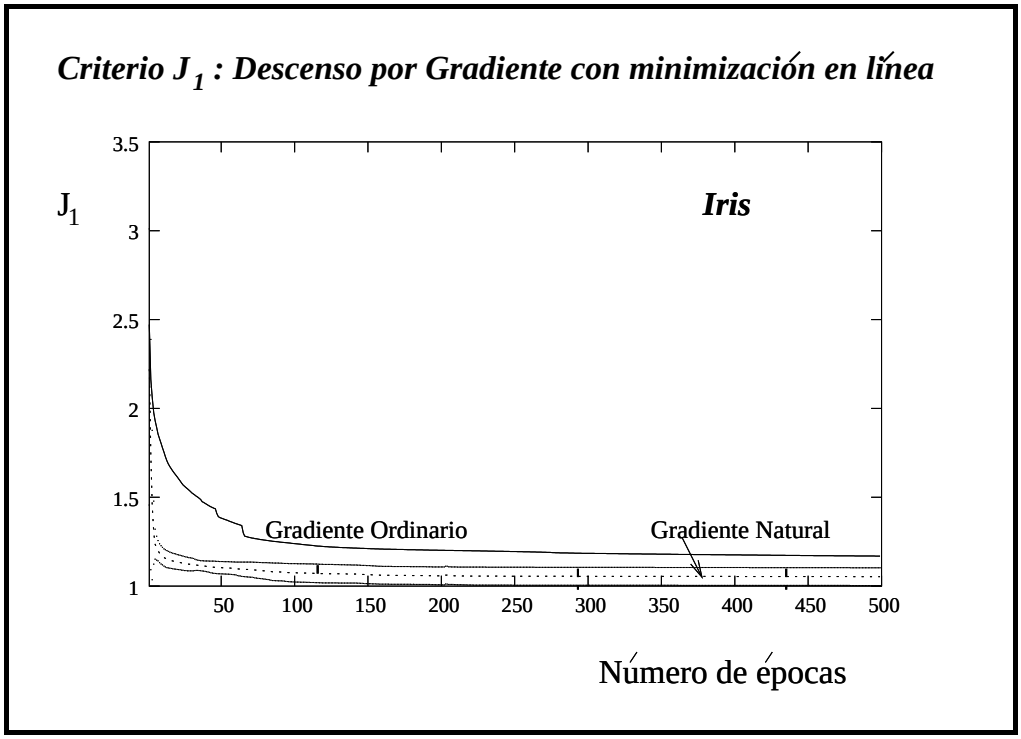


Figura 6.4: Iris ⇒ Descenso por gradiente ordinario y natural para J_1 utilizando minimización en línea

en cuanto al computo, tuvimos más problemas con el gradiente conjugado que con el Quasi–Newton; en concreto, con el criterio J_1 nos fue imposible realizar los 20 experimentos con un número de épocas superior a 30. Por encima de esa cantidad, teníamos problemas de singularidades en las matrices; pero si se observan la gráficas de la figura (6.6) se aprecia que no son necesarias muchas épocas para alcanzar la convergencia, por tanto no es un problema tan grave. Una vez más, las épocas de estos dos métodos no son simples; por cada época se realiza una búsqueda del mínimo local con un número variable de iteraciones.

6.2.2. Resultados en la clasificación

Hasta ahora nos hemos fijado en la minimización, pero también tenemos que tener presente cuál es la clasificación final para todas las posibles combinaciones de método de minimización y criterio vistas hasta el momento. En el cuadro (6.3), se recopila la clasificación para el mismo conjunto de entrenamiento con todas las posibles combinaciones. La columna tercera y cuarta del cuadro representan el promedio del tanto por ciento de acierto al clasificar los mismos datos de entrenamiento cuando se da por finalizado el entrenamiento y su desviación típica producida por las 20 pruebas distintas. Las dos últimas columnas son los promedios de los mínimos obtenidos para el criterio elegido y su desviación típica.

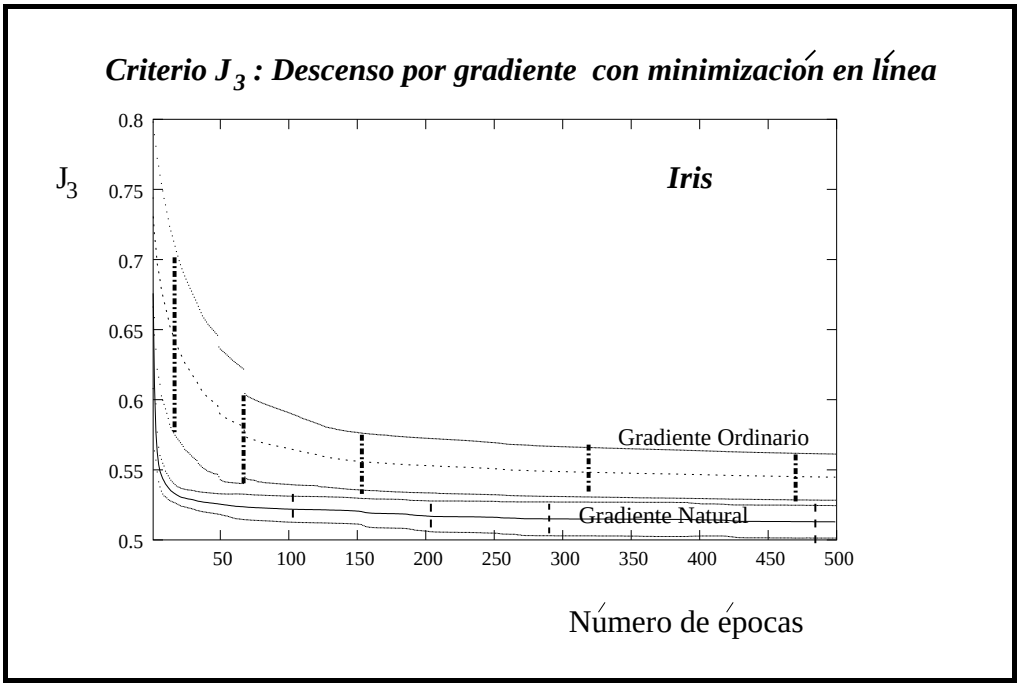


Figura 6.5: Iris ⇒ Descenso por gradiente ordinario y natural para J_3 utilizando minimización en línea

A la vista de los resultados del cuadro (6.3), vemos que se mantiene la coherencia con el entorno de minimización. Resumiendo, tenemos que para el conjunto de datos de los *Iris*, un conjunto sencillo que con el discriminante lineal de Fisher obtuvimos un acierto del 98 %, al introducir la no linealidad hemos disminuido la diferencia con respecto al 100 % de acierto, obteniéndose una mejora del $1,7$ % lograda con el entrenamiento efectuado con el criterio J_1 y el método de optimización del gradiente conjugado.

Las mejores combinaciones corresponden a los entrenamientos efectuados con los criterios J_1 y J_3, y los métodos de optimización gradiente conjugado y Quasi-Newton; mientras que las combinaciones correspondientes al criterio J_2 han sido las que peor parte se han llevado.

6.3. Conjunto Pima

Seguiremos con otra base de datos ya utilizada en el capítulo 1, la correspondiente a los indios Pima. En la sección 1.4.3 de dicho capítulo calculamos la matriz de clasificación a partir de un discriminante lineal de Fisher y el error cometido al clasificar los mismo patrones de entrenamiento ascendía al $23,18$ %.

En este apartado, usaremos la base de datos utilizada por Ripley en [Ripley, 1996]. Se trata de dos conjuntos de datos uno de entrenamiento formado

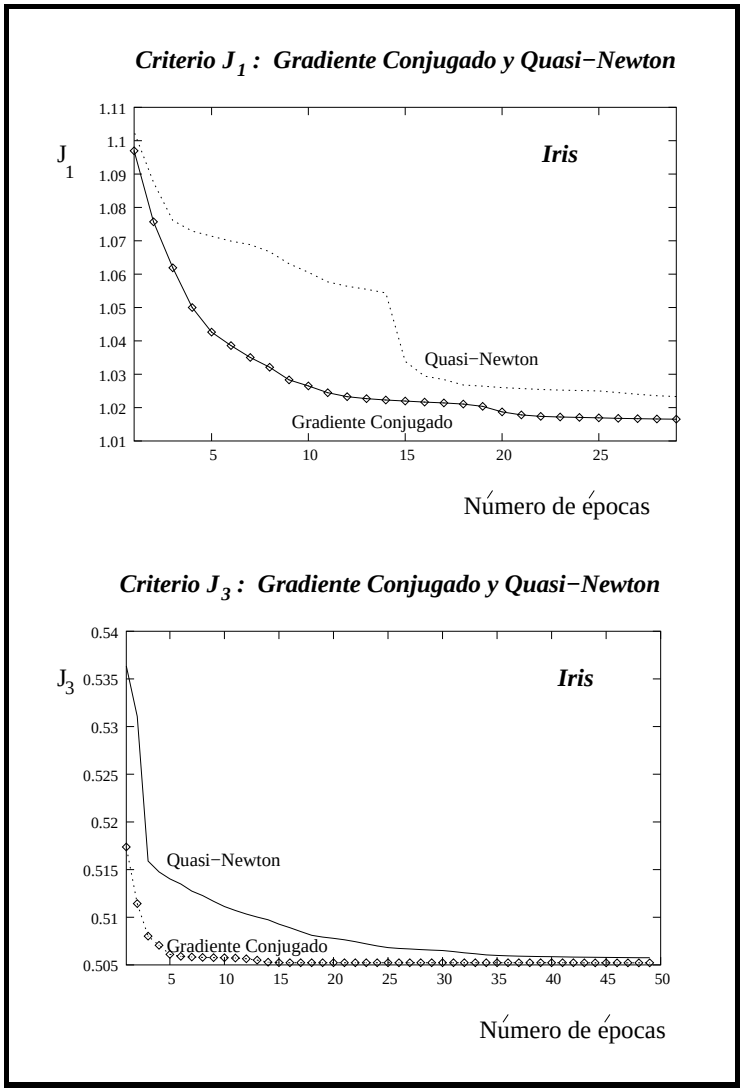


Figura 6.6: Iris ⇒ Gradiente Conjugado y Quasi–Newton con J_1 y J_3

por 200 patrones (132 No Diabetes/68 Si) y el otro de validación con 330 patrones (223 No/ 107 Si). Los datos de ambos conjuntos están basados en un estudio estadístico realizado sobre una población de mujeres de edades superiores a los 21 años. Cada medida consta de los siguientes atributos:

1. Número de embarazos
2. Concentración de glucosa en sangre (mg/dl)
3. Presión sanguínea diastólica (mmHg)
4. Proporción de grasa corporal estimada a partir de la medición de un pliegue en el tríceps (mm)

Criterio	Método de minimización	% Acierto	% σ_A	J	σ_J
J_1	Gradiente Ordinario	98,2	0,4	1,322	0,029
	Gradiente Natural	98,4	0,4	1,195	0,069
	Gradiente Ordinario (*)	98,4	0,7	1,169	0,084
	Gradiente Natural (*)	99,1	1,1	1,053	0,050
	Quasi–Newton	99,5	0,4	1,022	0,017
	Gradiente Conjugado	99,7	0,4	1,016	0,028
J_2	Gradiente Ordinario	97,9	0,6	1,018	0,005
	Gradiente Natural	98,1	0,4	1,003	0,004
J_3	Gradiente Ordinario	98,2	0,4	0,574	0,008
	Gradiente Natural	98,4	0,6	0,534	0,011
	Gradiente Ordinario (*)	98,4	0,5	0,549	0,015
	Gradiente Natural (*)	99,1	0,8	0,516	0,014
	Quasi–Newton	99,5	0,5	0,506	0,005
	Gradiente Conjugado	99,6	0,3	0,505	0,004

(*) Usando minimización en línea

Cuadro 6.3: Iris ⇒ Resumen en la clasificación del conjunto Iris

5. Indice de masa corporal ⇒ IMC ≡ Peso / altura^2, en unidades de Kg/m^2
6. Función de pedigrí en diabetes
7. Edad

Los datos de Ripley varían respecto a la base de datos de los indios Pima de UCI en que ignora el atributo correspondiente a la cantidad de insulina inyectada (μU/ml) en sangre (en el caso de ser necesaria), dado que es un atributo incompleto por no disponer de esta medida en gran parte de los patrones.

La evaluación empírica se ha realizado exactamente igual que con el conjunto de los *Iris*, con la excepción de que la medida del error de clasificación la realizamos para el conjunto de test. Cada una de las pruebas que se van a presentar están promediadas con 20 experimentos.

La arquitectura de red utilizada tiene una única capa oculta, pero hemos realizado los experimentos tanto para 5 unidades ocultas como para 10 y se verá que con 10 unidades en la capa oculta se llega a producir *overfitting*.

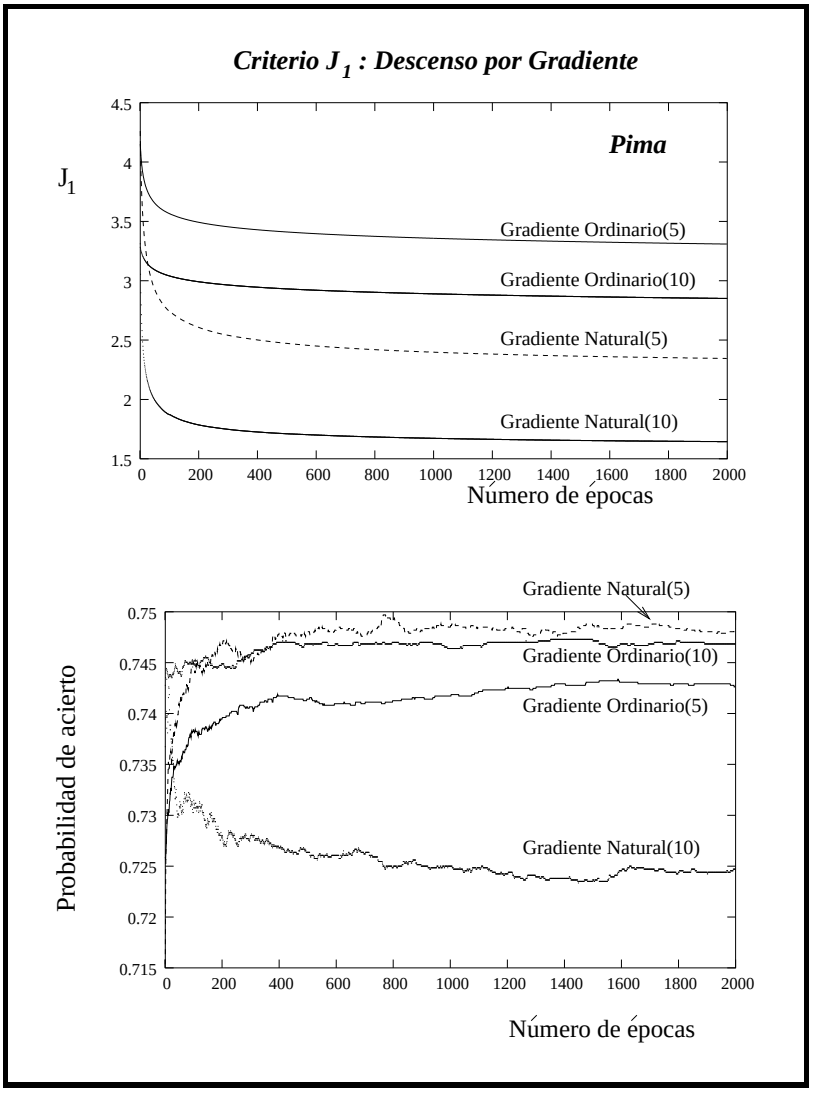


Figura 6.7: Pima $\Rightarrow$ Descenso por Gradiente para J_1

6.3.1. Minimización del criterio durante el entrenamiento

Descenso por gradiente simple

Las figuras (6.7), (6.8) y (6.9) representan el comportamiento de los criterios J_1, J_2 y J_3 respectivamente con entrenamientos realizados con redes de 5 y 10 unidades en la única capa oculta y usando como métodos de minimización los correspondientes al descenso por gradiente ordinario y natural.

Observando las tres figuras, se aprecia que la evolución de los tres criterios con el descenso por gradiente ordinario es exactamente la misma, la explicación radica en que para dos clases los tres criterios son idénticos, puesto que $\mathbf{S_T}$ y $\mathbf{S_B}$ son escalares, entonces los determinantes y trazas de dichas matrices son el

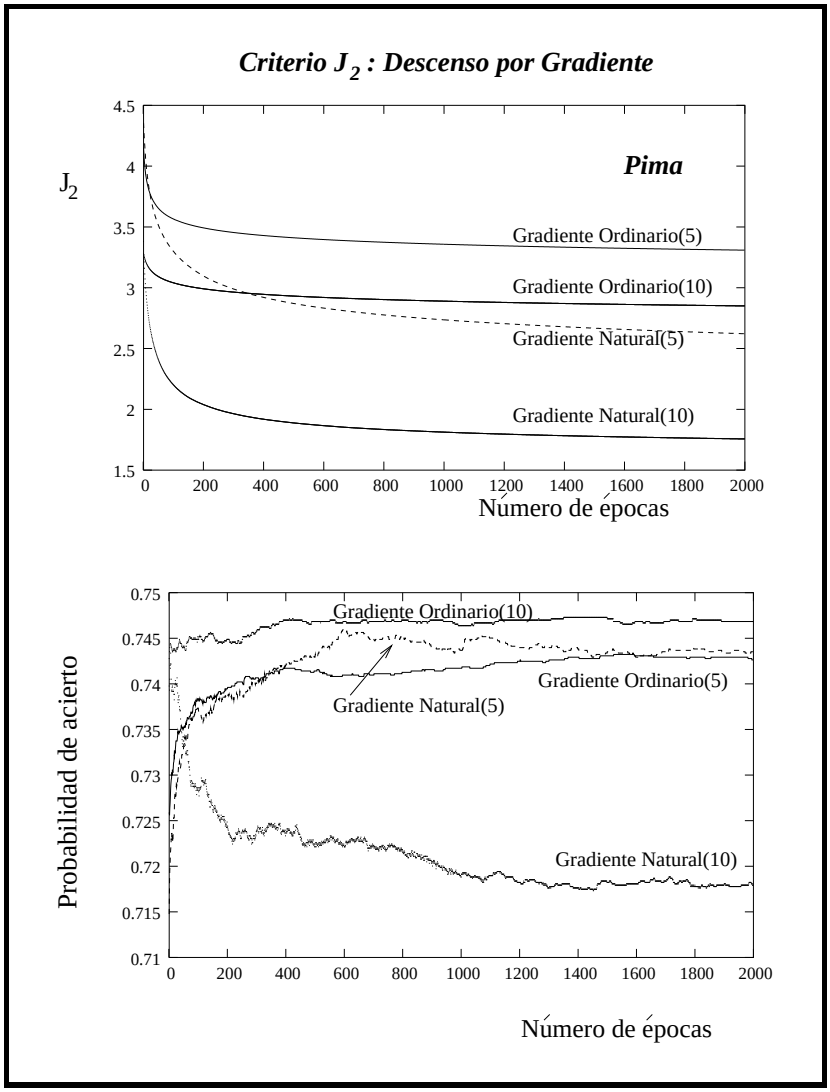


Figura 6.8: Pima $\Rightarrow$ Descenso por Gradiente para J_2

propio valor del escalar, así para todos ellos $J = S_T/S_B$. Para el resto de los métodos de minimización los resultados de los tres criterios difieren entre sí por las aproximaciones que se realizan en cada uno de los métodos de minimización.

Si nos centramos en el coste computacional, con 5 unidades en la capa oculta y tomando para el descenso por gradiente ordinario 2000 épocas, el equivalente con el gradiente natural sería $D\,K/C^2 = 5$ veces más costoso el gradiente natural que el simple para los criterios J_1 y J_3, lo que equivale para el mismo coste computacional a 400 épocas. Para J_2 el coste en el gradiente natural es $D\,K/C = 10$ veces superior, con lo cual para las 2000 épocas del gradiente ordinario tenemos que tomar 200 del gradiente natural. Cuando tomamos 10 unidades y 2000 épocas para el descenso por gradiente ordinario, el equivalente en el gradiente natural

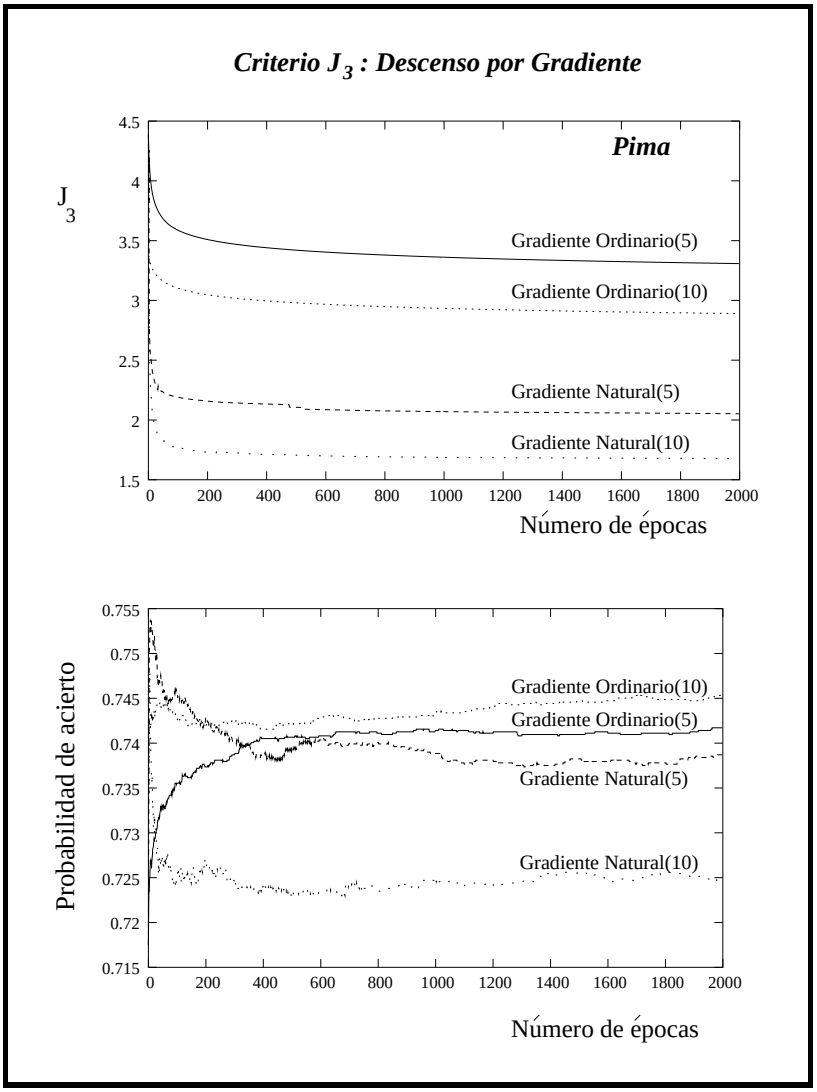


Figura 6.9: Pima ⇒ Descenso por Gradiente para J_3

para J_1 y J_3 sería $D\,K/C^2 = 10$ veces mayor éste último, en total 200 épocas y para J_1 tan sólo 100 épocas.

En el cuadro (6.4), se muestra para la simulación con 2000 épocas del gradiente ordinario su equivalente en épocas con el gradiente natural. De dicho cuadro, se puede concluir que a igual carga computacional el descenso por gradiente natural obtiene mejores resultados en la minimización que el descenso por gradiente simple.

Si comparamos las gráficas (6.7), (6.8) y (6.9) se puede ver que son muy parecidas (como ya hemos indicado deberían ser idénticas), de ahí que deduzcamos que las aproximaciones que estamos realizando para hallar la matriz de información de Fisher en el cálculo del gradiente natural son viables, remitimos a la sección 4.3.1.

	$J \pm \sigma$			
	Gradiente Ordinario (2000 épocas)		Gradiente Natural	
Criterio	5 Unidades	10 Unidades	5 Unidades	10 Unidades
J_1	3,309 ± 0,233	2,850 ± 0,249	2,501 ± 0,241	1,785 ± 0,096
J_2	3,309 ± 0,233	2,850 ± 0,249	3,097 ± 0,814	2,202 ± 0,185
J_3	3,309 ± 0,233	2,850 ± 0,249	2,133 ± 0,448	1,728 ± 0,209

Cuadro 6.4: Pima ⇒ Gradiente natural y ordinario a igual carga computacional

Un hecho que queremos ir adelantando aquí es el efecto producido por el exceso de unidades en la red. Si nos fijamos en las gráficas inferiores de las figuras (6.7), (6.8) y (6.9), donde se representa el promedio del tanto por ciento del acierto en la clasificación del conjunto de validación en función del número de épocas, se observa que el método que mejor minimiza no es precisamente con el que mejor clasificación se obtiene. Lo que ha ocurrido es que durante el entrenamiento, se ha buscado el conjunto de pesos que mejor se adapta a los datos de entrenamiento, llegando hasta el punto de producirse el defecto de memorización; con lo cual, al intentar clasificar con esos pesos un nuevo conjunto ya no es capaz de generalizar eficientemente y se obtiene una clasificación bastante peor que la obtenida cuando no se ha optimizado tanto el mínimo del criterio para el conjunto de entrenamiento, es decir, no se ha llegado al *sobreentrenamiento*.

Descenso por gradiente con minimización en línea

Cuando usamos minimización en línea en el descenso por gradiente para la base de datos *Pima*, figura (6.10), se observa que el gradiente natural no supone una mejora frente al gradiente ordinario. Las dos gráficas de la figura (6.10) corresponden a los dos criterios que admiten minimización en línea: J_1 y J_3. Las líneas de evolución para los dos tipos de entrenamiento tienen comportamientos similares (como cabría esperar), y viendo los resultados que se obtienen, podemos concluir en este caso que la mejora producida por introducir la minimización local en línea es superior a la de la elección de una métrica mejor adaptada; en este caso en concreto, no supone ninguna ventaja el uso del gradiente natural.

Compendio empírico de la minimización del criterio

A continuación, en el cuadro (6.5) presentamos un resumen con las combinaciones de entrenamientos y los valores mínimos alcanzados para el criterio elegido, así como su desviación típica. Se representa tanto para cinco unidades en la capa oculta como para diez.

Es fácil ver que al aumentar el número de unidades, se minimiza el valor del

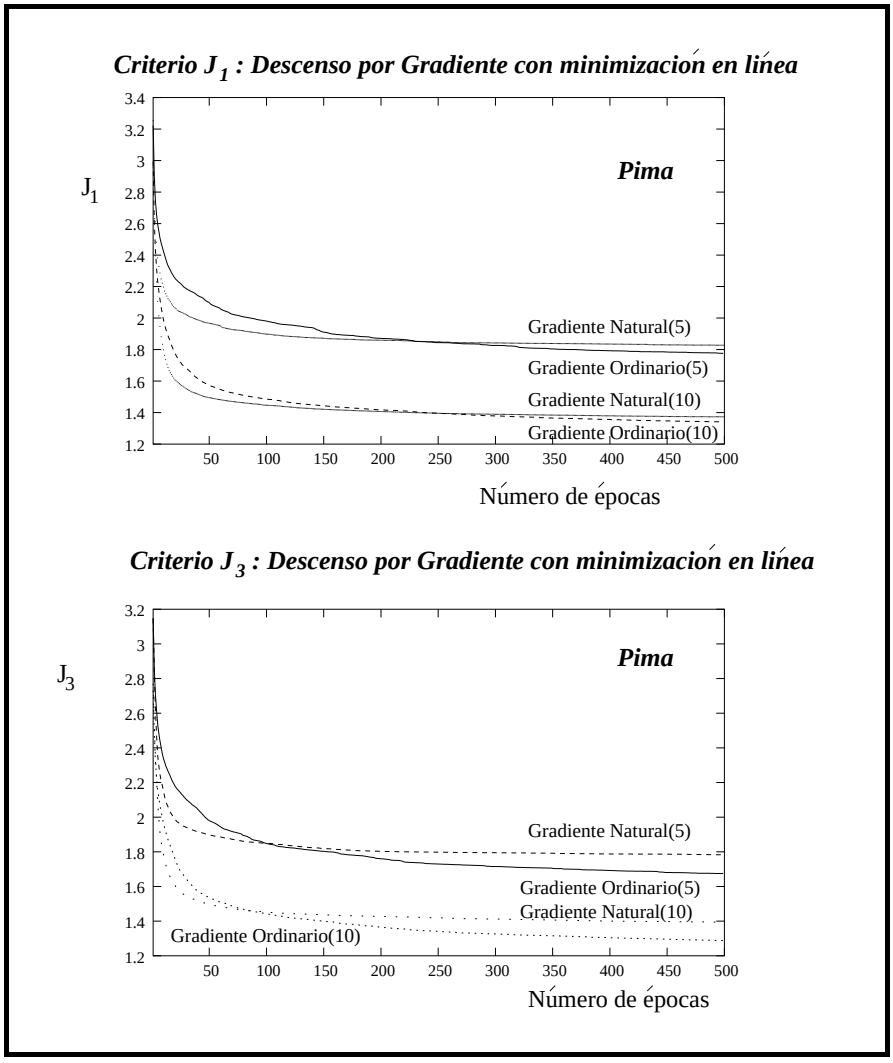


Figura 6.10: Pima ⇒ Descenso por Gradiente para J_1 y J_3 con minimización en línea

criterio para el conjunto de entrenamiento presentado y en general, también se hace más estrecha la desviación provocada por las 20 pruebas; con esto queremos decir que el mínimo encontrado está más cerca de ser el óptimo para el conjunto de entrenamiento presentado.

6.3.2. Resultados en la clasificación

Similar al cuadro (6.5), en el (6.6) presentamos un resumen con las combinaciones de los posible entrenamientos y el promedio en tanto por cierto de la asignación correcta de los patrones del conjunto de validación, realizada con los pesos obtenidos al final del entrenamiento. En la columna contigua se representa

Criterio	Método de minimización	*5 Unidades*		*10 Unidades*	
		J	σ_J	J	σ_J
J_1	Gradiente Ordinario	3,309	0,233	2,850	0,250
	Gradiente Natural	2,345	0,181	1,644	0,110
	Gradiente Ordinario (*)	1,776	0,227	1,341	0,100
	Gradiente Natural (*)	1,828	0,146	1,373	0,083
	Quasi–Newton	1,727	0,123	1,286	0,066
	Gradiente Conjugado	1,795	0,153	1,347	0,092
J_2	Gradiente Ordinario	3,309	0,233	2,850	0,250
	Gradiente Natural	2,622	0,538	1,756	0,117
J_3	Gradiente Ordinario	3,309	0,233	2,850	0,250
	Gradiente Natural	2,052	0,321	1,678	0,202
	Gradiente Ordinario (*)	1,675	0,130	1,288	0,083
	Gradiente Natural (*)	1,783	0,111	1,395	0,111
	Quasi–Newton	1,748	0,112	1,263	0,032
	Gradiente Conjugado	1,795	0,102	1,310	0,071

(*) Usando minimización en línea

Cuadro 6.5: Pima ⇒ Mínimos de J en función del método de minimización

Criterio	Método de minimización	*5 Unidades*		*10 Unidades*	
		% Acierto	% σ_A	% Acierto	% σ_A
J_1	Gradiente Ordinario	74,3	2,1	74,7	1,6
	Gradiente Natural	74,8	1,7	72,5	3,0
	Gradiente Ordinario (*)	74,1	3,4	70,4	2,0
	Gradiente Natural (*)	73,1	1,9	68,8	2,3
	Quasi–Newton	74,4	2,8	69,5	2,7
	Gradiente Conjugado	74,9	2,8	71,7	2,3
J_2	Gradiente Ordinario	74,3	2,1	74,7	1,6
	Gradiente Natural	74,4	2,2	71,8	2,4
J_3	Gradiente Ordinario	74,3	2,1	74,7	1,6
	Gradiente Natural	73,9	2,3	72,5	3,1
	Gradiente Ordinario (*)	73,2	2,2	68,7	2,1
	Gradiente Natural (*)	72,7	2,4	70,9	3,3
	Quasi–Newton	75,2	2,2	69,9	1,8
	Gradiente Conjugado	75,2	2,0	71,6	2,7

(*) Usando minimización en línea

Cuadro 6.6: Pima ⇒ Clasificación del conjunto de test en función del método de minimización

la desviación típica (σ_A) sobre las 20 pruebas. Este resumen se realiza tanto para cinco unidades ocultas como para diez.

De los resultado obtenidos, vemos que estamos ante un problema complejo. Según Ripley en [Ripley, 1996], el error de clasificación es aproximadamente del 24 %. Con ADnL obtenemos resultados de ese orden. Una vez más, es posible ver en el cuadro (6.6) que el excesivo aumento de unidades en la capa oculta provoca un aumento del error de clasificación.

6.4. Conjunto XOR_3D

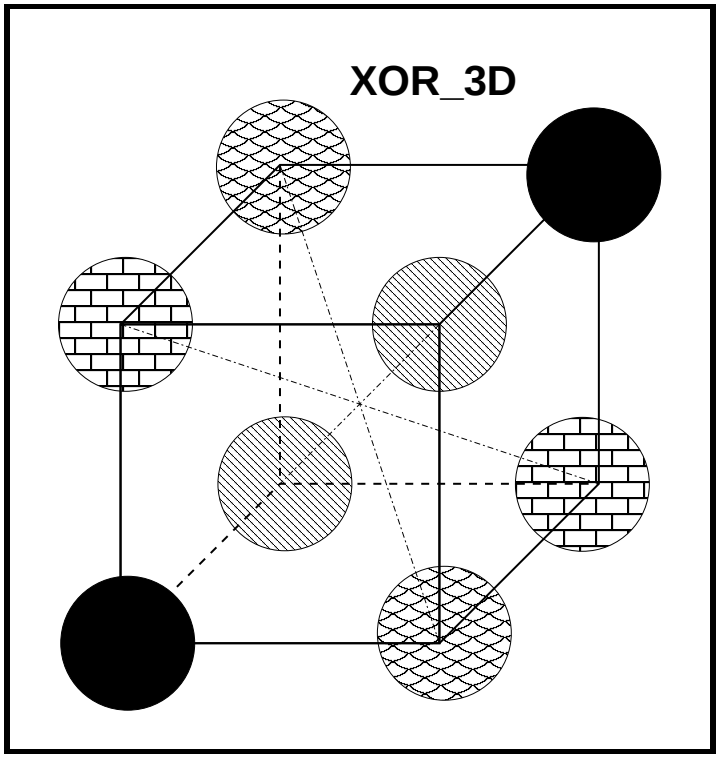


Figura 6.11: XOR_3D

En esta sección trataremos con un conjunto sintético de datos que se corresponde con un XOR tridimensional. Este conjunto está formado por cuatro clases, donde los elementos de cada clase están situados en los vértices opuestos de un cubo; en la figura (6.11) hemos representado esquemáticamente dicho conjunto. Los datos del problema han sido generados por una normal centrada en cada vértice del cubo. Para el conjunto de entrenamiento se han generado 125 patrones para cada una de las clases y para el conjunto de test 375 patrones por clase.

Como las gráficas que podríamos presentar para este conjunto de datos son muy similares a las presentadas con las dos bases de datos anteriores, omitiremos

Criterio	Método de minimización	% Acierto	% σ_A	J	σ_J
J_1	Gradiente Ordinario	96,0	5,6	1,888	1,400
	Gradiente Natural	98,8	0,6	1,300	0,096
	Gradiente Ordinario (*)	97,8	0,6	1,098	0,047
	Gradiente Natural (*)	97,2	0,9	1,075	0,021
	Quasi–Newton	97,2	0,6	1,097	0,042
	Gradiente Conjugado	96,8	0,7	1,055	0,022
J_2	Gradiente Ordinario	94,9	1,8	1,138	0,041
	Gradiente Natural	98,3	0,8	1,027	0,014
J_3	Gradiente Ordinario	93,5	2,7	0,459	0,032
	Gradiente Natural	98,9	0,5	0,356	0,005
	Gradiente Ordinario (*)	97,3	1,0	0,344	0,003
	Gradiente Natural (*)	96,8	0,9	0,341	0,002
	Quasi–Newton	97,3	0,6	0,346	0,004
	Gradiente Conjugado	96,9	0,8	0,342	0,004

(*) Usando minimización en línea

Cuadro 6.7: XOR_3D ⇒ Resumen en la clasificación del conjunto XOR_3D

las gráficas y expondremos directamente los resultados en un cuadro resumen (6.7). Los valores promediados son el resultado de 20 pruebas.

Analizaremos a continuación los resultados del cuadro (6.7). Tenemos que separar dos enfoques distintos: el primero, qué métodos minimizan mejor el conjunto de entrenamiento y el segundo, con cuáles obtengo mejores resultados al clasificar el conjunto de test. Comenzamos por el criterio J_1 y $J3$, que tienen un comportamiento similar y vemos que existe una gran diferencia en la minimización entre el gradiente simple y el resto de los métodos. En particular, el gradiente ordinario se queda muy lejos del mínimo y su desviación típica, evaluada sobre las 20 pruebas, es bastante elevada. Si nos fijamos en el porcentaje de acierto también se percibe en este caso una desviación alta, muy superior a cualquiera de los casos estudiados, de esto se deduce que el gradiente ordinario simple para las 2000 épocas de entrenamiento, no llega a alcanzar el mínimo deseado.

Ahora bien, en contraposición también se percibe el efecto de sobreentrenamiento: los mínimos más bajos no se corresponden con las mejores tasas de aciertos sobre el conjunto de test. No siempre la búsqueda del mejor mínimo conlleva a mejores resultados en la clasificación para un conjunto desconocido y ésto es lo que nos ocurre con los métodos de minimización en línea: Quasi–Newton y gradiente conjugado. La cota más elevada en cuanto al porcentaje de acierto se corresponde con el descenso por gradiente natural.

Para el criterio J_2, sólo tenemos que discutir entre el gradiente ordinario o

el gradiente natural con el método de descenso por gradiente; en este caso los resultados son similares a los de los criterios homólogos J_1 y $J3$: el gradiente ordinario una vez más pierde frente a su competidor el gradiente natural.

Si consideramos igual carga computacional, la equivalencia en el número de épocas para los dos gradientes es de 1067 épocas del gradiente natural frente a las 2000 del ordinario para los criterios J_1 y J_3 y 267 para el criterio J_2 (donde $D = 3$, $K = 10$ y $C = 4$). En el cuadro (6.8) se muestran los valores del criterio y el porcentaje de acierto a idéntica carga computacional

	$\bar{J} \pm \sigma$	
Criterio	Gradiente Ordinario (2000 épocas)	Gradiente Natural
J_1	1,888 ± 1,400	1,329 ± 0.120 (1067 épocas)
J_2	1,018 ± 0,005	1,034 ± 0.014 (267 épocas)
J_3	0,459 ± 0,032	0,356 ± 0,004 (1067 épocas)
	% (Acierto ± σ_A)	
	Gradiente Ordinario (2000 épocas)	Gradiente Natural
J_1	96,0 ± 5,6	98,6 ± 0,7 (1067 épocas)
J_2	94,9 ± 1,8	98,6 ± 0,6 (267 épocas)
J_3	93,5 ± 2,7	98,9 ± 0,4 (1067 épocas)

Cuadro 6.8: XOR_3D ⇒ Gradiente natural y ordinario a igual carga computacional

A la vista del cuadro (6.8) podemos concluir que a igual carga computacional y para el conjunto XOR_3D, el descenso por gradiente natural es mejor que el descenso por gradiente simple, no sólo en los valores del criterio y del porcentaje de acierto, sino también en la dispersión de ambos.

6.5. Conjunto Tiroides

La base de datos con la que vamos a trabajar en esta sección es una de las bases más importantes en clasificación con redes neuronales y su fama se basa en la complejidad del conjunto: una fuerte predominancia de una de las clases, que por si sola se lleva el 92 % de los patrones, junto con un gran solapamiento de las clases. Se tienen dos conjuntos uno de entrenamiento formado por 3772 patrones y otro de test con 3428 patrones. En el informe [Schiffmann et al., 1994] se presenta una revisión de varias técnicas de retropropagación en perceptrones multicapa usando esta base de datos.

El problema del tiroides trata de determinar si un paciente en estudio presenta disfunción de la glándula tiroides, de gran importancia para el funcionamiento del cuerpo humano. El desorden más común de la tiroides está causado por su baja de actividad, es decir, no produce suficiente hormona; a esta enfermedad se la conoce como *hipotiroidismo*. Con menor frecuencia, se da el caso de que la glándula tiroides es demasiado activa, segregando más hormona de la necesaria; enfermedad también conocida como *hipertiroidismo*.

Así pues, el conjunto de datos está dividido en tres clases: paciente sano (normal), paciente con hipertiroidismo y paciente con hipotiroidismo. La proporción de cada una de las clases es la siguiente:

Conjunto	# Hipertiroidismo	# Hipotiroidismo	# Normal
Entrenamiento	93	191	3488
Test	73	177	3178

Cuadro 6.9: Tiroides ⇒ Composición de la base de datos

Los pacientes sanos suponen un 92 % del conjunto de patrones. Luego un clasificador que sea significativo debe superar el 92 % de acierto. El número de atributos por patrón asciende a 21, de los cuales 15 son binarios y el resto son continuos.

Como un primer escaner, comprobamos la complejidad del problema con el discriminante lineal de Fisher, de donde obtuvimos que el error cometido sobre el conjunto de test ascendía al 25,3 %, un error muy superior al límite del 8 % impuesto por el número de pacientes sanos y enfermos que contiene la base de datos. La matriz de clasificación que se obtiene con el discriminante lineal de Fisher aplicado al conjunto de test es la siguiente:

Conjunto Test	Hipertiroidismo	Hipotiroidismo	Normal
Hipertiroidismo	44	27	2
Hipotiroidismo	1	118	58
Normal	7	773	2398

Cuadro 6.10: Tiroides ⇒ Matriz de Clasificación con el discriminante lineal de Fisher

Como se ve, el problema no puede resolverse mediante un análisis lineal. De hecho, durante bastante tiempo se ha considerado este problema como particularmente difícil de resolver mediante PMCs. Sin embargo, revisando bibliografía reciente, encontramos que hay estudios en los que se demuestra que sólo dos de los atributos binarios los referentes a tiroxina y operación del tiroides son relevantes [Duch, 2004]. De este modo se reduce considerablemente la dimensionalidad del problema al pasar de los 21 atributos originales a los 8 seleccionados. No obstante, vamos a comenzar nuestro estudio con la base del tiroides original, al ser la

más estudiada y por las dificultades que supone. Dada su gran dimensionalidad, lo que supone una gran carga computacional, decidimos que los experimentos a realizar para hallar los promedios constaran de cinco pruebas.

Criterio	Método de minimización	% Acierto	% σ_A	J	σ_J
J_1	Gradiente Ordinario	97,2	0,6	2,95	0,12
	Gradiente Natural	93,4	1,6	3,99	0,82
	Gradiente Ordinario (*)	97,5	0,3	1,821	0,055
	Gradiente Natural (*)	97,8	0,3	1,313	0,051
	Quasi–Newton	97,9	0,1	1,278	0,029
	Gradiente Conjugado	97,7	0,3	1,195	0,003
J_2	Gradiente Ordinario	91,9	1,5	1,554	0,034
	Gradiente Natural	95,5	1,6	1,428	0,206
J_3	Gradiente Ordinario	96,4	0,3	1,984	0,031
	Gradiente Natural	97,2	0,3	0,648	0,020
	Gradiente Ordinario (*)	97,7	0,3	0,601	0,006
	Gradiente Natural (*)	97,5	0,4	0,578	0,008
	Quasi–Newton	97,5	0,2	0,567	0,004
	Gradiente Conjugado	97,6	0,1	0,552	0,006

(*) Usando minimización en línea

Cuadro 6.11: Tiroides ⇒ Resumen en la clasificación del conjunto tiroides

En el cuadro (6.11) se presenta un resumen en función del criterio y tipo de minimización de la media de la clasificación para el conjunto de test y el valor medio alcanzado por el criterio con el que se ha efectuado el entrenamiento junto con las desviaciones típicas de los dos valores, evaluado todo ello sobre 5 pruebas. En todos los casos presentados en el cuadro (6.11), los valores que se obtienen del porcentaje de acierto para el conjunto de test están dentro del margen de resultados admitidos como buenos por la bibliografía existente sobre este conjunto de datos, remitimos a [Schiffmann et al., 1994]. A la vista de los resultados del cuadro (6.11) es difícil decantarse por cuál de los métodos es mejor. Al menos para este caso, parece que el criterio J_2 se encuentra en inferioridad con respecto a los otros dos.

Es un hecho, ya constatado en [Santa Cruz y Dorronsoro, 1998], que la red ADnL mejora los resultados con respecto a un PMC cuando el tamaño entre las clases está desequilibrado. En el cuadro (6.12), se recogen los promedios de las matrices de clasificación para el conjunto de test de los datos de la diabetes original con los siguientes ensayos:

1. Seis ejecuciones con un PMC de arquitectura similar a la usada para las redes ADnL.

PMC	Hipertiroidismo	Hipotiroidismo	Normal	% Error Parcial
Hipertiroidismo	56,5	8,67	7,83	22,6
Hipotiroidismo	33,5	60	83,5	66,1
Normal	19,66	14,67	3143,67	1,1
Error Total: 4,9 %				

Criterio J_1 con descenso por gradiente natural y minimización en línea

ADnL	Hipertiroidismo	Hipotiroidismo	Normal	% Error Parcial
Hipertiroidismo	57,8	12,0	3,2	20,8
Hipotiroidismo	0,4	165,8	10,8	6,3
Normal	10,8	34,8	3132,4	1,4
Error Total: 2,1 %				

Criterio J_1 con descenso por gradiente conjugado

ADnL	Hipertiroidismo	Hipotiroidismo	Normal	% Error Parcial
Hipertiroidismo	61,4	8,4	3,2	15,9
Hipotiroidismo	0,4	165,0	11,6	6,8
Normal	12,4	32,4	3133,2	1,4
Error Total: 2,0 %				

Criterio J_3 con descenso por gradiente natural y minimización en línea

ADnL	Hipertiroidismo	Hipotiroidismo	Normal	% Error Parcial
Hipertiroidismo	58,2	12,0	2,8	20,3
Hipotiroidismo	0,2	162,6	14,2	8,1
Normal	12,8	33,6	3131,6	1,5
Error Total: 2,2 %				

Criterio J_3 con Quasi–Newton como método de minimización

ADnL	Hipertiroidismo	Hipotiroidismo	Normal	% Error Parcial
Hipertiroidismo	60,2	10,6	2,2	17,5
Hipotiroidismo	0,8	163,8	12,4	7,5
Normal	16,4	36,0	3125,6	1,6
Error Total: 2,3 %				

Cuadro 6.12: Tiroides ⇒ Promedio de la clasificación para el conjunto de test

2. Cinco ejecuciones con redes ADnL entrenadas con el criterio J_1 y descenso por gradiente natural usando minimización en línea.

3. Cinco ejecuciones con redes ADnL entrenadas con el criterio J_1 y descenso por gradiente conjugado.

4. Cinco ejecuciones con redes ADnL entrenadas con el criterio J_3 y descenso por gradiente natural usando minimización en línea.

5. Cinco ejecuciones con redes ADnL entrenadas con el criterio J_3 y usando como método de minimización Quasi–Newton.

Estas cuatro formas de entrenamiento para las redes ADnL se han seleccionado al azar y sólo con la intención de no sesgar los resultados de la clasificación.

Se verifica que la red ADnL, para este conjunto de datos, mejora el resultado de la clasificación con respecto al PMC en aproximadamente un 60 %. Esta mejora se debe sobre todo a que distingue mejor los pacientes enfermos, en especial los que tienen hipotiroidismo, que dentro de los enfermos es el grueso. El error parcial al clasificar enfermos de hipotiroidismo con el PMC es de un 66 % mientras que con ADnL este error desciende hasta situarse alrededor del 6,5 % para las redes entrenadas con el criterio J_1, lo que supone una mejora del 90 %. Las redes entrenadas con el criterio J_1 resuelven mejor la situación de los enfermos con hipotiroidismo que las entrenadas con el criterio J_3, donde la mejora respecto al PMC es aproximadamente del 86 %.

El dato negativo de la clasificación con la red ADnL es que prácticamente la totalidad de los enfermos de hipotiroidismo mal clasificados son reconocidos como personas sanas; ésto conlleva un riesgo puesto que la revisión medica puede pararse en este punto afirmando que una persona está sana cuando en realidad no es así. Los enfermos de hipertiroidismo mal clasificados con ADnL, en su mayoría son clasificados como pacientes con hipotiroidismo, esto puede ser problemático si se decide aplicarles inmediatamente el tratamiento típico de enfermos de hipotiroidismo. Es cierto que existe un empeoramiento en la clasificación de los individuos sanos con respecto al PMC. Ahora bien, este agravante no es tan significativo, pues lo que lleva consigo es que el paciente mal clasificado sea sometido a un mayor número de pruebas médicas hasta dar con la mejor de las noticias: ¡está sano!.

Vamos a acabar esta sección rehaciendo los cálculos anteriores con los 8 atributos de entrada citados anteriormente, con los dos criterios más sólidos J_1 y J_3 y como métodos de minimización nos centramos en la zona inferior de la tabla, pues es donde se obtienen mejores resultados tanto en minimización del criterio como en clasificación de los patrones de test. Como la carga computacional es considerablemente menor que con los 21 atributos se han realizado 20 experimentos para calcular los promedios y los resultados obtenidos están resumidos en el cuadro (6.13).

Como puede observarse a partir de los cuadros (6.11) y (6.13), con el problema del tiroides eligiendo 8 atributos se obtienen tasas de error que en global mejoran a las obtenidas con los 21 atributos originales, con la ventaja añadida de trabajar con un problema más sencillo.

Luego concluimos que en éste, como en otros casos, demasiada información deja de ser una ayuda efectiva, pudiendo incluso llegar a corromper el aprendizaje.

Criterio	Método de minimización	% Acierto	% σ_A	J	σ_J
J_1	Gradiente Ordinario	96,3	0,8	2,332	0,459
	Gradiente Natural	97,0	1,0	1,802	0,263
	Gradiente Ordinario (*)	97,3	1,0	1,645	0,223
	Gradiente Natural (*)	97,6	0,4	1,432	0,074
	Quasi–Newton	97,7	0,6	1,724	1,117
	Gradiente Conjugado	98,2	0,2	1,320	0,073
J_2	Gradiente Ordinario	95,8	1,0	1,384	0,033
	Gradiente Natural	97,9	0,3	1,083	0,223
J_3	Gradiente Ordinario	95,8	1,0	1,041	0,205
	Gradiente Natural	96,6	1,6	0,863	0,160
	Gradiente Ordinario (*)	96,1	0,4	0,854	0,002
	Gradiente Natural (*)	97,8	0,2	0,694	0,043
	Quasi–Newton	98,2	0,2	0,571	0,152
	Gradiente Conjugado	98,1	0,3	0,563	0,016

(*) Usando minimización en línea

Cuadro 6.13: Tiroides ⇒ Resumen en la clasificación del conjunto tiroides usando 8 atributos de entrada

En el próximo capítulo, volveremos a tratar de nuevo el tema de la elección de los atributos relevantes como entradas de redes del tipo PMC.

Capítulo 7

Selección Empírica de Arquitecturas Optimas en ADnL

7.1. Introducción

En el capítulo 5 se estudió, desde el punto de vista teórico, cómo eliminar unidades irrelevantes en una red ADnL. En el capítulo actual veremos los resultados empíricos de los desarrollos teóricos anteriores. La división del capítulo está centrada en dos puntos:

1. Eliminar unidades no lineales irrelevantes, para lo cual tomaremos alguna de las bases de datos ya utilizadas anteriormente y con ayuda de los fundamentos del test de Wald, sección 5.3.2, eliminamos de la red ADnL las unidades no lineales irrelevantes.

2. En una segunda sección, nos centramos en la búsqueda del número óptimo de unidades en la última capa oculta de una red ADnL. Usaremos para este estudio el test de Wilks desarrollado en la sección 5.2 aplicándolo a alguna de las bases de datos ya mencionadas.

Vamos a trabajar con una red ADnL de una sola capa oculta, con lo que en el primer punto cuando hablamos de unidades no lineales, en realidad se trata de las unidades de entrada y en el segundo punto las unidades de la última capa oculta se corresponden con las unidades de la única capa oculta.

7.2. Estudio de Entradas Relevantes

En esta sección estudiaremos con ayuda del test de Wald, sección 5.3.2, qué unidades de entrada son relevantes en una red ADnL de una única capa oculta. Mostraremos dos tratamientos en la selección de las entradas relevantes:

1. Aplicación directa del test de Wald en donde las distribuciones subyacentes cumplen la hipótesis de Wald, sección 5.3.2. Mostraremos este primer enfoque para el conjunto de datos de las iridáceas.

2. Cuando ya no es posible aplicar directamente el test, puesto que los estadísticos de Wald que obtenemos no siguen las distribuciones de hipótesis nula, y sin embargo si que es posible entresacar información de los valores de éstos. Mostraremos este segundo enfoque con el conjunto de diabetes de los indios Pima.

7.2.1. Aplicación Directa del Test de Wald

Conjunto Iris

Con el conjunto de los datos Iris entrenados con una red ADnL de una sola capa oculta que contiene tres unidades, realizamos el estudio del test de Wald para ver la relevancia de las unidades de entrada: longitud y anchura de los sépalos y longitud y anchura de los pétalos de las tres especies de iridáceas, setosa, versicolor y virgínica.

Atributo	Test de Wald	Aceptación Hipótesis al 95 %	% Acierto
Longitud Sépalos	2,3309	Acepto $\rightleftharpoons$ Atributo Irrelevante	
Anchura Sépalos	9,7207	Rechazo $\rightleftharpoons$ Atributo Relevante	98
Longitud Pétalos	10,3706	Rechazo $\rightleftharpoons$ Atributo Relevante	
Anchura Pétalos	43,4115	Rechazo $\rightleftharpoons$ Atributo Relevante	
Anchura Sépalos	6,3402	Acepto $\rightleftharpoons$ Atributo Irrelevante	98
Longitud Pétalos	59,7876	Rechazo $\rightleftharpoons$ Atributo Relevante	
Anchura Pétalos	28,5402	Rechazo $\rightleftharpoons$ Atributo Relevante	
Longitud Pétalos	16,1927	Rechazo $\rightleftharpoons$ Atributo Relevante	96
Anchura Pétalos	38,7439	Rechazo $\rightleftharpoons$ Atributo Relevante	
Anchura Pétalos	607,4097	Rechazo $\rightleftharpoons$ Atributo Relevante	96

Cuadro 7.1: Iris $\Rightarrow$ Test de Wald en las unidades de entrada

Realizamos varias pruebas con redes ADnL de arquitectura inicial $4 \otimes 3 \otimes 2$ y fuimos eliminando atributos de entrada usando el test de Wald como método orientativo sobre qué atributo debería eliminar. En el cuadro (7.1) representamos el valor del estadístico de Wald para un red entrenada con el conjunto de los iris, de arquitectura inicial $4 \otimes 3 \otimes 2$ y usando como función criterio J_1. A medida que

comprobábamos que alguno de los atributos de entrada era irrelevante íbamos disminuyendo paulatinamente la arquitectura de la red $3 \otimes 3 \otimes 2$, para después pasar a $2 \otimes 3 \otimes 2$ e incluso a $1 \otimes 3 \otimes 2$, aunque ésta última, el test de Wald no nos la permitiría. En el cuadro está también representado el porcentaje de acierto con cada una de las arquitecturas.

Para tres unidades ocultas, esto es, $k = 3$ y tomando un nivel se significancia $\alpha = 0,05$, se tiene que $\chi^2_{k,(1-\alpha)} = \chi^2_{3,0,95} = 7{,}8147$. Valores del test de Wald superiores supondrían rechazar la hipótesis nula: todas las conexiones entre el atributo en estudio y las unidades de la capa siguiente son nulas; con lo cual si rechazamos la hipótesis tenemos que ese atributo es relevante.

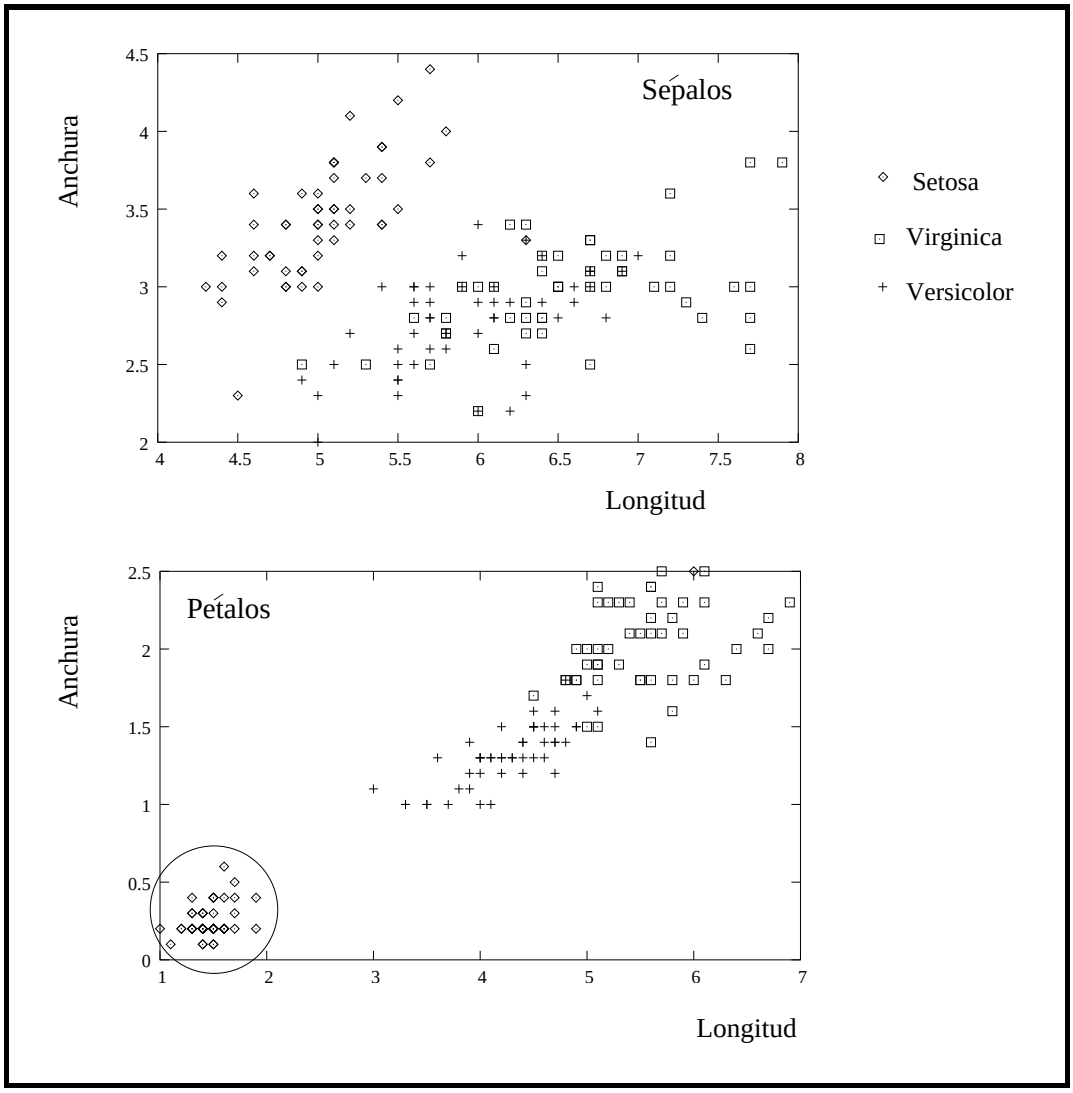


Figura 7.1: Iris ⇒ Atributos de los iris separados en Sépalos y Pétalos

A la vista de los resultados del test de Wald, pudimos comprobar que dentro

de los atributos del conjunto iris, los que realmente tienen información son los atributos correspondiente a las medidas de los pétalos. En la figura (7.1) representamos los atributos correspondientes a los sépalos y a los pétalos por separado y a simple vista se puede deducir que:

1. La clase correspondiente a los iris de la especie setosa es fácilmente separable, especialmente si tenemos en cuenta la medición de los pétalos; en este caso no tenemos dudas en una clasificación binaria: el iris en estudio es setosa o alguna de las otras dos especies.

2. En cuanto a las otras dos especies, si tomamos como medida la correspondiente al tamaño de los sépalos, se tiene que en muchos casos sería prácticamente imposible distinguir entre virgínica o versicolor. Por el contrario, si tomamos las medidas de los pétalos, sí que es posible distinguir entre las dos especies, aunque sigue existiendo una pequeña zona en la que solapan las dos especies.

Observando el porcentaje de acierto presentado en el cuadro (7.1), podríamos pensar en quedarnos con un único atributo y forzando el test de Wald, obtenemos que este atributo sería la anchura de los pétalos. Como complemento para verificar este resultado hemos realizado la clasificación de los iris tomando un sólo atributo y como método de clasificación el más sencillo: la distancia euclídea a las medias. Las matrices de clasificación resultantes están representadas en el cuadro (7.2); observándolas se verifica todo lo anteriormente dicho y si pretendemos quedarnos con un sólo atributo obtenemos que éste sería la anchura de los pétalos. Así pues hemos llegado a la misma conclusión que con ADnL y el test de Wald.

Con este ejemplo, podemos decir que con la combinación ADnL y test de Wald es posible vislumbrar unidades irrelevantes dentro de conexiones no lineales. Ahora bien, no todos los casos se ajustan tan apropiadamente a los valores tabulados de la función χ^2. Esta desavenencia se debe a que al tratar casos reales, el valor de la variable aleatoria $\| \varsigma_N \|^2$, ecuación (5.11), no converge necesariamente a la distribución Chi-cuadrado esperada. Pero en cualquier caso, sí que existe diferencia en el valor del estadístico asociado a una unidad relevante y el asociado a una unidad irrelevante: los valores de los estadísticos de unidades relevantes son de varios ordenes de magnitud superiores a los estadísticos de unidades irrelevantes.

7.2.2. Aplicación Indirecta del Test de Wald

En la mayoría de los problemas reales no es posible aplicar el test de Wald directamente, pues las premisas iniciales no se cumplen, sección 5.3.2. Por tanto tenemos que buscar otros medios para poder decidir cuáles de los atributos son relevantes. Proponemos el siguiente estudio: a partir de una red entrenada eficientemente se hallan los estadísticos de Wald para los atributos de entrada y miramos la posibilidad de eliminar aquel atributo cuyo test es menor, dado que es

Longitud Sépalos			
Especies	Setosa	Versicolor	Virgínica
Setosa	45	5	0
Versicolor	6	30	14
Virgínica	2	12	36

Error Clasificación: 26.00 %

Anchura Sépalos			
Especies	Setosa	Versicolor	Virgínica
Setosa	33	1	16
Versicolor	2	27	21
Virgínica	8	19	23

Error Clasificación: 44.67 %

Longitud Pétalos			
Especies	Setosa	Versicolor	Virgínica
Setosa	50	0	0
Versicolor	0	48	2
Virgínica	0	6	44

Error Clasificación: 5.33 %

Anchura Pétalos			
Especies	Setosa	Versicolor	Virgínica
Setosa	50	0	0
Versicolor	0	48	2
Virgínica	0	4	46

Error Clasificación: 4.00 %

Cuadro 7.2: Iris $\Rightarrow$ Matrices de clasificación considerando distancias a medias con un único atributo

el menos relevante; además para eliminar un atributo se debe cumplir que existe una diferencia significativa entre el estadístico asociado a él y los estadísticos de mayor valor.

Conjunto Pima

En la tabla (7.3) representamos los valores del estadístico de Wald durante una simulación secuencial de retirada de atributos para el conjunto de diabetes de los indios Pima. Si observamos dicha tabla se aprecia que:

- Desde el inicio, cuando consideramos todos los atributos, parece posible eli-

Atributo	Test Wald	Atributo eliminado
Número de embarazos	7.399 e-01	Eliminado
Glucosa en sangre	8.307 e+01	
Presión sanguínea	4.849 e+01	
Medición en el tríceps	4.192 e+01	
IMC	1.886 e+01	
Pedigrí	9.013 e+00	
Edad	5.145 e+01	
Glucosa en sangre	5.799 e+02	
Presión sanguínea	1.159 e+01	
Medición en el tríceps	7.408 e+00	Eliminado
IMC	1.885 e+01	
Pedigrí	1.611 e+02	
Edad	1.414 e+02	
Glucosa en sangre	1.451 e+02	
Presión sanguínea	3.322 e+01	Eliminado
IMC	4.425 e+01	
Pedigrí	6.912 e+01	
Edad	1.568 e+02	
Glucosa en sangre	8.341 e+02	
IMC	3.237 e+02	
Pedigrí	2.612 e+02	Eliminado
Edad	1.480 e+04	
Glucosa en sangre	2.658 e+03	
IMC	9.109 e+01	Eliminado
Edad	3.594 e+04	
Glucosa en sangre	8.646 e+01	
Edad	4.955 e+01	

Cuadro 7.3: Pima ⇒ Estadístico de Wald en función de los atributos

minar el correspondiente al número de embarazos, pues el valor del test de Wald es significativamente menor que el resto de los valores de los estadísticos. Si comparamos su valor con el estadístico más alto, que se corresponde con la concentración de glucosa en sangre, es más de 100 veces menor.

- El siguiente en eliminar es la medición de la proporción de grasa en el músculo del tríceps. En este caso también se observa que existe diferencia entre el valor de su estadístico asociado que es 7.4 y el correspondiente al mayor de los valores que es 580, asociado éste con la concentración de glucosa en sangre.

- En el siguiente paso, tenemos más complicada la elección del atributo a eliminar; tres de los atributos (presión sanguínea, índice de masa corporal y la función de pedigrí) están muy próximos en cuanto a su valor del estadístico de Wald, mientras que en otro margen superior están los atributos correspondientes a la concentración de glucosa en sangre y la edad de la fémina. Optamos por eliminar secuencialmente el atributo con menor valor del estadístico de Wald.

- Finalmente, llegamos a un punto en el que los estadísticos asociados a los atributos que permanecen son muy similares entre sí; éste es el momento de parar el proceso de eliminación de atributos. En nuestro caso se corresponde con los atributos concentración de glucosa en sangre y la edad de la fémina. Este resultado, en realidad se va vislumbrando a lo largo del proceso de eliminación de atributos aquí presentado, pues son estos dos atributos los que alcanzan los valores más altos del estadístico asociado durante todas las evaluaciones.

En la tabla (7.3) hemos presentado una simulación, pero sería conveniente llevar a cabo una estadística de varias simulaciones para verificar qué atributos pueden considerarse irrelevantes. Proponemos el siguiente mecanismo de evaluación de los atributos de entrada: en cada simulación eliminamos de atributo en atributo aquél con menor valor del estadístico de Wald hasta quedarnos con un único atributo y anotamos en cada experimento el orden en el que es eliminado cada uno de los atributos. Obteniéndose, al final, una tabla que representa la frecuencia con que es eliminado cada atributo en función del orden cronológico de eliminación. En el análisis de la tabla de frecuencias, se apreciará que determinados atributos son eliminados rápidamente, lo que quiere decir que se podrían considerar como irrelevantes, aquellos atributos que en gran medida son relevantes serán eliminados siempre de los últimos y en la zona intermedia quedarán los atributos que siendo importantes no son los más destacados.

Para el conjunto de diabetes de los indios Pima realizamos el ensayo anterior con 20 simulaciones y en cada caso anotamos la secuencia cronológica de atributos eliminados. Así, en el cuadro (7.4) se representa la frecuencia con que es eliminado cada atributo en función del orden cronológico que ocupa en la eliminación y en el siguiente cuadro (7.5) se representa la frecuencia acumulada en la eliminación secuencial de los atributos. Observando las dos tablas se puede deducir que hay tres tipos de atributos:

1. Los que parece que se puede prescindir de ellos, pues siempre son los prime-

Orden en la eliminación →	1º	2º	3º	4º	5º	6º	Unico atributo
Nº Embarazos	9	3	3	4	1	0	0
Glucosa en sangre	0	0	0	0	1	10	9
Presión sanguínea	3	4	8	4	0	1	0
Medición tríceps	5	7	3	1	4	0	0
IMC	1	4	1	5	5	3	1
Pedigrí	1	2	4	4	4	2	3
Edad	1	0	1	2	5	4	7

Cuadro 7.4: Pima ⇒ Frecuencia en la eliminación secuencial de atributos

Orden en la eliminación →	1º	2º	3º	4º	5º	6º
Nº Embarazos	9	12	15	19	*20*	–
Glucosa en sangre	0	0	0	0	1	11
Presión sanguínea	3	7	15	19	19	*20*
Medición tríceps	5	12	15	16	*20*	–
IMC	1	5	6	11	16	19
Pedigrí	1	3	7	11	15	17
Edad	1	1	2	4	9	13

Cuadro 7.5: Pima ⇒ Frecuencia acumulada en la eliminación secuencial de atributos

ros en ser eliminados; éstos se corresponden con el número de embarazos, medición de la relación grasa-músculo en el tríceps y la presión sanguínea.

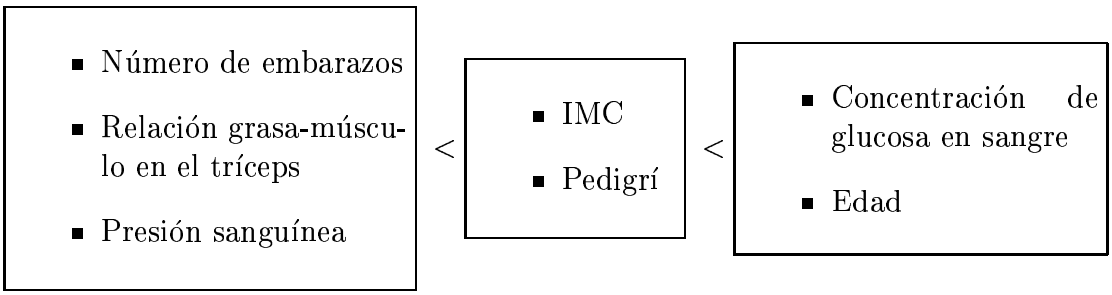
2. Los atributos que llamaremos difusos son aquellos que contienen información en cualquier estadio y ayudan para afinar en la clasificación; a éste tipo corresponden los atributos índice de masa corporal y el valor de la función de pedigrí.

3. Por último tenemos los atributos realmente relevantes, éstos son la pareja de atributos formada por la concentración de glucosa en sangre y la edad. Aunque dentro de estos dos atributos el primero es el más significativo, si bien la combinación de los dos mejora la predicción, respecto a considerar sólo la glucosa en sangre, en la clasificación de una nueva fémina como diabética o no.

Considerando sólo los dos últimos atributos que hemos clasificado como relevantes, hemos alcanzado valores medios en la probabilidad de acierto del

$(78,06 \pm 0,67)\,\%$ y considerando solamente la glucosa en sangre la probabilidad de acierto es también alta, el $(74,96 \pm 1,67)\,\%$; estos datos han sido recogidos con la ponderación de cinco entrenamientos distintos. Comparando estos resultados con los de la sección 1.4.3, donde se entrenaba con todos los atributos, se observa que se obtienen mejores resultados considerando sólo los atributos referentes a la glucosa en sangre y la edad que considerando todos en conjunto.

Como conclusión para el conjunto estudiado aquí, podríamos afirmar que la secuencia en importancia de los atributos es la siguiente:



7.3. Unidades Optimas en la Ultima Capa Oculta de una Red ADnL

Hemos visto en la sección anterior que existen ocasiones en las que tenemos más información de la que precisamos y puede ocurrir que seleccionando mejor dicha información se derive hacia un conocimiento del problema mayor que considerando todos los atributos de entrada.

Bien, pues ha llegado el momento de ver que ocurre en las unidades de la última capa oculta, y como ya hemos indicado anteriormente vamos a trabajar con redes de una sola capa oculta; entonces en este caso la pregunta es: ¿cuántos son los parámetros libres que necesito en la red ADnL?. Pues, la respuesta la tenemos en manos del test de Wilks y sus estadísticos, sección 5.2.

Como en la sección anterior, donde seleccionamos las unidades de entrada mediante el estadístico de Wald, desarrollaremos ésta considerando los dos mismos enfoques: aplicación directa del test de Wilks expuesta a través del conjunto de las iridáceas y la aplicación indirecta, donde es preciso manipular los estadísticos de Wilks para obtener información del margen de arquitecturas adecuadas. Este segundo enfoque lo veremos a través de dos conjuntos: diabetes de los indios Pima y XOR tridimensional.

Unidades	Test de Wilks	Aceptación Hipótesis al 95 % de confianza
5	4.035576 e+00	Rechazo $\rightleftharpoons$ Unidad Relevante
	3.778942 e+00	Rechazo $\rightleftharpoons$ Unidad Relevante
	7.553388 e+02	Rechazo $\rightleftharpoons$ Unidad Relevante
	1.064540 e+07	Rechazo $\rightleftharpoons$ Unidad Relevante
	1.447173 e+00	Acepto $\rightleftharpoons$ Unidad Irrelevante
4	6.363997 e+02	Rechazo $\rightleftharpoons$ Unidad Relevante
	2.331023 e+02	Rechazo $\rightleftharpoons$ Unidad Relevante
	3.531880 e+01	Rechazo $\rightleftharpoons$ Unidad Relevante
	1.678513 e+00	Acepto $\rightleftharpoons$ Unidad Irrelevante
3	1.740784 e+03	Rechazo $\rightleftharpoons$ Unidad Relevante
	7.414581 e-02	Acepto $\rightleftharpoons$ Unidad Irrelevante
	8.570281 e+07	Rechazo $\rightleftharpoons$ Unidad Relevante
2	1.724039 e+03	Rechazo $\rightleftharpoons$ Unidad Relevante
	1.802017 e+03	Rechazo $\rightleftharpoons$ Unidad Relevante

Cuadro 7.6: Iris $\Rightarrow$ Test de Wilks en las unidades de la última capa oculta

7.3.1. Aplicación Directa del Test de Wilks

Conjunto Iris

Recurrimos una vez más a los datos del conjunto de las iridáceas para buscar el número de unidades óptimas que debería tener la última capa oculta. Como vamos a utilizar una red de una sola capa oculta, entonces lo que buscamos es conocer cuántas son las unidades óptimas de la capa oculta. Para ello comenzamos expandiendo las unidades ocultas sobre las de entrada, los cuatro atributos de los iris; de este modo comenzamos con un red de 5 unidades ocultas y al finalizar el entrenamiento se midió la relevancia de las unidades ocultas con un test de Wilks, sección (5.2); aquella unidad cuyo valor de la relevancia sea menor que el correspondiente al valor del estadístico $\mathcal{F}_{(1-\alpha),(C-1,\ n-C-d+1)}$, donde d es el número de unidades ocultas. Una vez eliminada la unidad irrelevante, repetimos el mismo procedimiento con una unidad menos, hasta que llegamos al caso de que todas las unidades son relevantes. Como d podría valer entre cinco y una unidad, el valor de $\mathcal{F}$ con $n-C-d+1 = 148-d$, no se va a ver muy modificado; así para todos los entrenamientos tomamos como medida para discernir entre unidad relevante o no el valor correspondiente a las tablas de la distribución de Fisher $\mathcal{F}_{0,95\ (2,\infty)} = 3{,}00$.

En el cuadro (7.6) se expresan los valores del estadístico de Wilks asociado

a cada unidad oculta durante una de las pruebas realizadas. En dicho cuadro se señala también qué unidad es eliminada en cada entrenamiento, aquella cuyo valor del estadístico de Wilks es inferior a la cantidad $\mathcal{F}_{0,95\,(2,\infty)} = 3{,}00$, lo que indicaría que tengo que aceptar la hipótesis nula que en nuestro modelo significa que esa unidad es irrelevante. Finalmente, se llega al caso de no poder eliminar más unidades y éste se corresponde con dos unidades en la capa oculta.

Con una red extrema compuesta por los dos atributos de entrada, los correspondientes a la medida de los pétalos de los iris, y con dos unidades ocultas, realizamos 20 entrenamientos y los resultados promediados son los que presentamos a continuación: $\hat{J}_1 = 1{,}155 \pm 0{,}010$, $\%\hat{Acierto} = 96{,}13 \pm 0{,}45$ y la matriz de clasificación promediada es la siguiente

Especies	Setosa	Versicolor	Virgínica
Setosa	50	0	0
Versicolor	0	47,1	2,9
Virgínica	0	2,8	47,2

Error Clasificación: 3,8 %

7.3.2. Aplicación Indirecta del Test de Wilks

Saliéndonos del mundo idílico de los iris, nos encontramos con que en la mayoría de las bases de datos no es posible aplicar directamente el test de Wilks para obtener el número de unidades óptimo de la última capa oculta de una red ADnL; luego es preciso desarrollar un nuevo enfoque. Proponemos observar los estadísticos procedentes de una arquitectura como un todo y no considerar los estadísticos como entes separados por unidades ocultas. Así pues, para cada arquitectura hallamos la media del logaritmo decimal de los estadísticos de las unidades de la última capa oculta y la dispersión de los mismos; hemos elegido el logaritmo decimal porque nos da una medida de la magnitud de los estadísticos.

Observamos que a medida que aumentaba el número de unidades de la última capa oculta, la media de los logaritmos de los estadístico disminuía y el caso contrario ocurría con su dispersión. La explicación a este fenómeno radica en que cuando las unidades son relevantes, los valores del estadístico de Wilks son grandes y además muy próximos entre ellos, cuando comienza a haber unidades no relevantes es donde aparece diferencia en magnitud entre los estadísticos, aquellos que representan a unidades irrelevantes son muy inferiores respecto a los de unidades relevantes y ésto hace que la dispersión entre los valores de los estadísticos aumente. Tenemos pues un compromiso a la hora de elegir la arquitectura óptima; tomaremos como óptima aquella que realiza un balance de selección entre la media

de los logaritmos de los estadísticos suficientemente alta y a la vez la dispersión de éstos baja.

Estudiaremos la elección de unidades de la capa oculta para dos conjuntos de datos distintos: el conjunto Pima y el XOR tridimensional.

Conjunto Pima

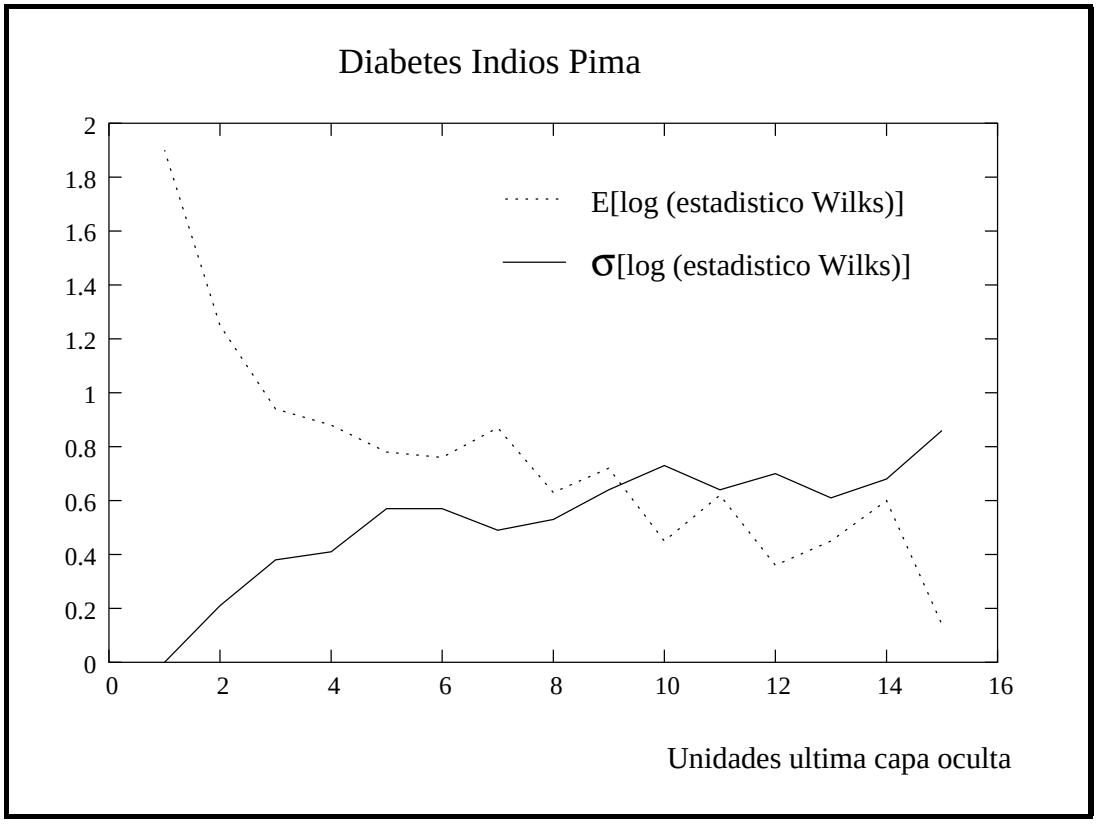


Figura 7.2: Pima ⇒ Evolución de la media y dispersión del logaritmo decimal de los estadísticos de Wilks en unidades de la última capa oculta de redes ADnL

En la figura (7.2) se representa la evolución de la media de los logaritmos decimales de los estadísticos de Wilks y su dispersión frente al número de unidades de la última capa oculta aplicado al conjunto de datos de la diabetes de los indios Pima. Los datos que se representan son la ponderación de 20 experimentos distintos. Cuando la última capa oculta tiene una sola unidad el valor del logaritmo del único estadístico promediado sobre los 20 experimentos es 1,9 y como sólo tenemos una unidad la dispersión es cero; éste es sin duda, el caso de máxima media y mínima dispersión, pero ello no quiere decir que el número de unidades óptimo en la última capa oculta sea la unidad. Hay que buscar un equilibrio entre

las dos magnitudes: media alta y dispersión baja; con lo cual recurrimos a la razón

$$R = \frac{E\,[\log(\text{Estadístico Wilks})]}{\sigma\,[\log(\text{Estadístico Wilks})]}$$

como una nueva medida indicativa del número de unidades óptimo.

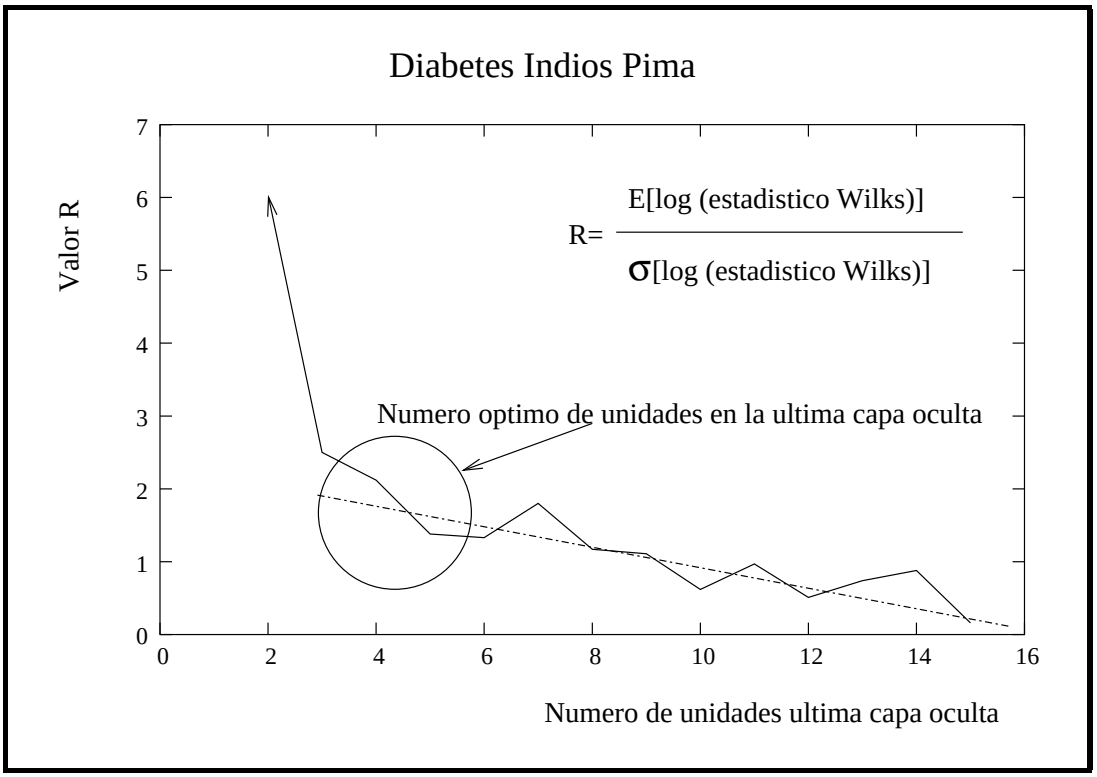


Figura 7.3: Pima $\Rightarrow$ Evolución de la razón R en función de las unidades de la última capa oculta

La gráfica (7.3) representa la evolución de dicha razón R comenzando con dos unidades en la última capa oculta, dado que con una unidad $R = \infty$, puesto que $\sigma = 0$. Observando los cambios significativos de pendiente en la tendencia de la gráfica de R, podemos asignar una zona de redes óptimas en torno a la colisión de las dos pendientes, esta zona es la ocupada por redes de 4–5 unidades en la última capa oculta y si nos queremos decantar a favor de una arquitectura diríamos que la de 4 unidades es la óptima, pues entre las dos es la que tiene mayor media y menor dispersión, dato que se puede observar en la figura (7.2), aunque la de 5 unidades también sería adecuada.

A continuación, vamos a comparar estos resultados con el que se obtiene de representar el porcentaje de acierto para el conjunto de test en función del número de unidades de la última capa oculta, figura (7.4). Según se puede ver en dicha

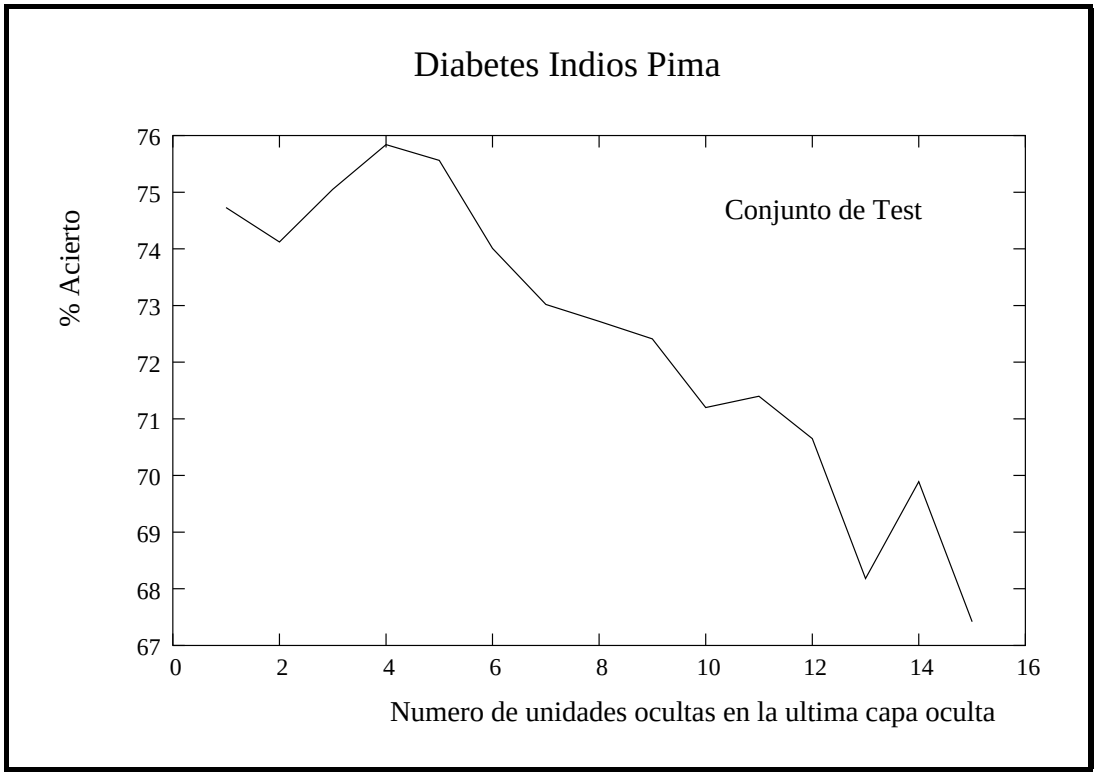


Figura 7.4: Pima ⇒ Evolución del porcentaje de acierto en función de las unidades de la última capa oculta

figura el máximo de acierto corresponde con 4 unidades en la capa oculta y a partir de 5 unidades se produce un sobreajuste cada vez más acentuado. En la sección 6.3, intuimos con datos experimentales este fenómeno de sobreajuste. Según la figura (7.3) a partir de 9 unidades es mayor la dispersión que la media de los logaritmos decimales de los estadísticos, con lo cuál podríamos concluir que redes con más de 9 unidades en la capa oculta no tienen sentido.

Como conclusión, para el conjunto de datos de la diabetes de los indios Pima, la red ADnL de una sola capa oculta que consideramos óptima es aquella formada por tan sólo dos entradas: concentración de glucosa en sangre y edad y cuatro unidades en la capa oculta. Además con este método también se ha podido vislumbrar el sobreentrenamiento al aumentar considerablemente el número de unidades.

Conjunto XOR_3D

Para el conjunto XOR tridimensional introducido en la sección 6.4 realizamos el mismo experimento anterior, sólo que el número de repeticiones es menor, utilizamos 10 repeticiones. Comprobamos durante la ejecución del experimento que el número mínimo de unidades en la capa oculta es cinco; por debajo de él no se

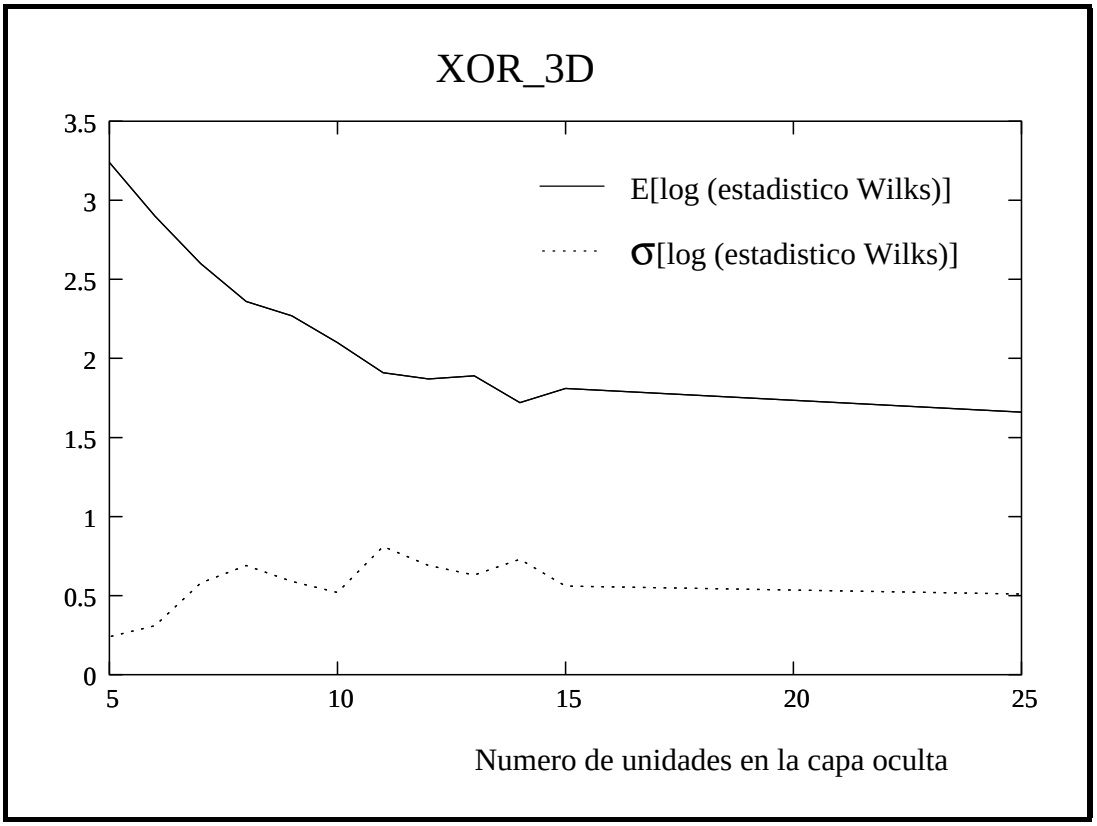


Figura 7.5: XOR_3D ⇒ Evolución de la media y dispersión del logaritmo decimal de los estadísticos de Wilks en unidades de la última capa oculta de redes ADnL

consigue entrenar eficientemente la red, luego los resultados que vamos a presentar proceden de 5 unidades en adelante.

La figura (7.5) representa la evolución de la media y dispersión del logaritmo decimal del estadístico de Wilks en función del número de unidades de la capa oculta. Obsérvese que entre 15 y 25 no se han realizado pruebas, consideramos que eran innecesarias después de estudiar un caso alejado, el correspondiente a 25 unidades, cuyo objetivo era comprobar el comportamiento asintótico.

Con XOR_3D la media y desviación mantienen trayectorias asintóticas paralelas o por lo menos, así ocurre dentro de un número razonable de unidades ocultas.

Del mismo modo que con el conjunto de diabetes de los indios Pima, aquí también calculamos la razón R, representada en la figura (7.6), donde buscamos el corte de las dos trayectorias asintóticas correspondientes al número de unidades mínimo y a considerar un número de unidades elevado ($\approx \infty$). Una vez localizada la zona de corte, elegimos el punto correspondiente a mayor media y menor dispersión que para el conjunto XOR_3D se corresponde con que el número de unidades óptimo es diez. Si recordamos en la sección 6.4, elegimos el número de

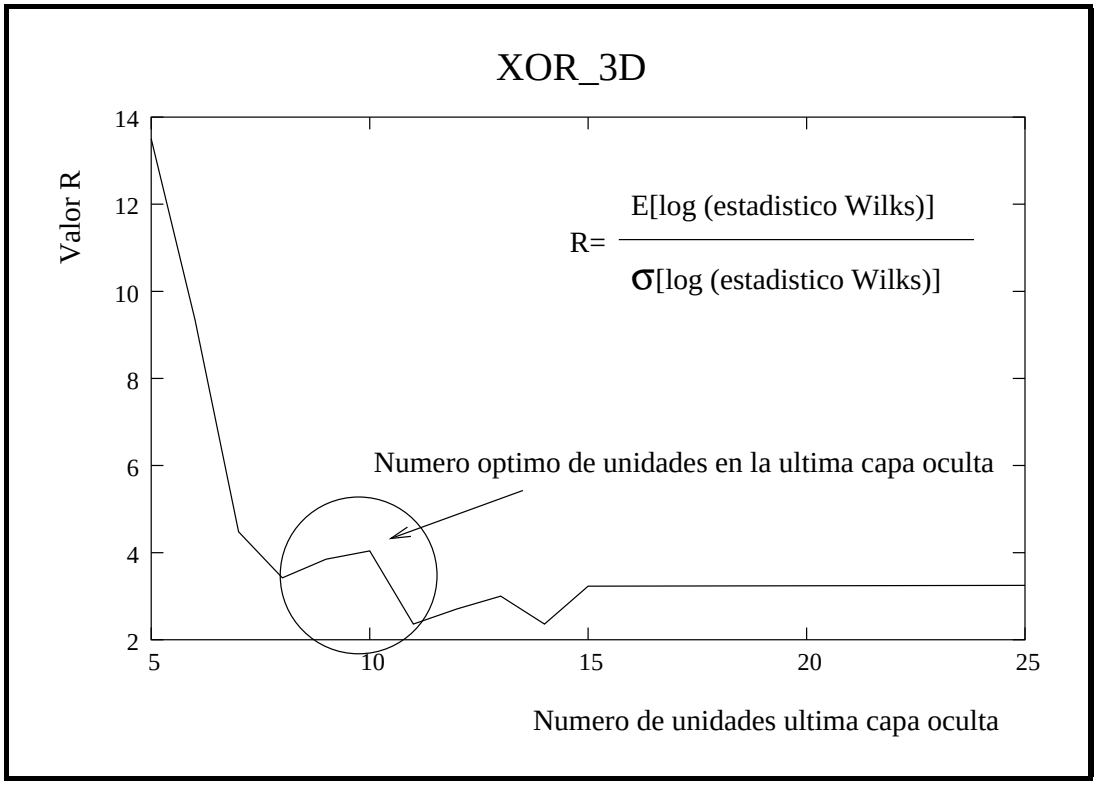


Figura 7.6: XOR_3D $\Rightarrow$ Evolución de la razón R en función de las unidades de la última capa oculta

unidades de la capa oculta precisamente como diez y esta elección se hizo en base a los mejores resultados del porcentaje de acierto para el conjunto de test, evaluados sobre diversas pruebas previas a la elección final del número de unidades. Luego, una vez más existe solapamiento entre la arquitectura óptima predicha con el test de Wilks y la que podríamos obtener con la clasificación de un conjunto de validación; pero con la ventaja de no necesitar este último conjunto, que por otro lado, no siempre es posible disponer del conjunto de validación.

Conclusiones y Futuras Líneas de Investigación

Conclusiones

Las conclusiones de este trabajo se pueden resumir en los siguientes puntos:

- La extensión a C clases de la técnica de clasificación, Análisis Discriminante no Lineal (ADnL), que siendo supervisada y similar al PMC, tiene la ventaja sobre éste de estar libre de cargas de identificación; lo que en parte, permite mejorar la clasificación de patrones en muestras con probabilidades a priori muy divergentes.

- Se han estudiado, exhaustivamente, tres funciones criterio a optimizar con ADnL y los resultados obtenidos indican que cuanto mayor información intrínseca de la muestra a analizar contenga el criterio elegido mejor será la clasificación final. En este sentido, el premio al mejor criterio se lo lleva J_1: razón de determinantes.

- Se ha introducido una matriz de información y la métrica de Riemann asociada como mejora para la convergencia en técnicas del tipo descenso por gradiente; con ello se ha definido el gradiente natural para ADnL haciendo uso de aproximaciones a la matriz de información de Fisher. Los resultados al incluir las aproximaciones propuestas para el descenso por gradiente natural muestran la deseada aceleración de la convergencia y añaden la ventaja de obtener mínimos de la función criterio a evaluar en ADnL ciertamente más significativos que los que se obtienen con el mismo método de minimización y la métrica clásica Euclídea.

- Hemos introducido procedimientos de selección de arquitecturas de ADnL, mediante dos técnicas estadísticas distintas: la combinación matriz de información y *test de Wald*, aplicable en unidades de conexiones no lineales de lo que podemos deducir que sería común y podría aplicarse a redes del tipo PMC y por otro lado el *test de Wilks* aplicado sólo en la parte lineal de ADnL, es decir, donde se realiza el discriminante lineal de Fisher.

Mediante el uso de las dos técnicas anteriores, somos capaces de ajustar, dentro de un rango, el número idóneo de unidades que debe tener la red: sin excedernos para que no se produzca sobreentrenamiento y sin quedarnos cortos de modo que no se obtengan los resultados deseados por insuficiencia de parámetros libres.

Futuras Líneas de Investigación

En cuanto a las futuras líneas de investigación, creemos que no van a ir en la línea de ADnL, pues es un tema que profesamos zanjado; pero sí podemos afirmar que los conocimientos adquiridos durante la elaboración de este trabajo nos van a permitir derivar hacia líneas de investigación similares que comprenden problemas de clasificación difíciles de abordar y que pueden agruparse bajo el epígrafe de *muestras extremas*, bien por la presencia de desequilibrios en las muestras, bien por el alto grado de solapamiento entre patrones de distintas clases que hace difícil el establecimiento de fronteras de separación o bien, por la necesidad de trabajar con patrones de muy alta dimensión. Es cierto que ADnL presenta una buena eficacia en problemas de muestra extrema de dimensionalidad baja–media, sin embargo, la complejidad y el coste computacional del entrenamiento de las redes ADnL y de los PMCs en general hace problemático su uso en problemas de alta dimensión.

En los últimos años se está proponiendo bajo el epígrafe de *clasificadores sobre márgenes* diversos clasificadores de gran potencia, que combinan una proyección no lineal multidimensional de los patrones originales con un clasificador lineal que busca obtener un margen, esto es, una distancia entre las proyecciones de datos de diferentes clases tan grande como sea posible y sin usar objetivos para los patrones. El punto de arranque está en las Máquinas de Vectores Soporte (Support Vector Machines, SVM) [Cortes y Vapnik, 1995], [Cristianini y Shawe-Taylor, 2000] y las simplificaciones a éstos surgidas recientemente como el caso de los perceptrones pararelos [Auer et al., 2001] que ofrecen aproximación universal bajo una arquitectura simple y un entrenamiento de una complejidad menor que la del algoritmo de retropropagación con la ventaja adicional de que no requiere objetivos para los patrones.

En cualquier caso, el objetivo global con vistas a un futuro se centra en la construcción de clasificadores sobre margen y su aplicación a problemas de muestra extrema que ciertamente son la mayoría de las muestras del mundo real.

Apéndices

Apéndice A

Optimización de Criterios de Fisher

A.1. Introducción

En este apéndice, vamos a ver cómo criterios para el discriminante de Fisher aparentemente distintos tienen sin embargo, una solución común. Demostraremos que resolver el análisis discriminante de Fisher para estos criterios se reduce a resolver un problema de autovalores y sus autovectores asociados.

A.2. Optimización de criterios de Fisher

Denominaremos J_1 al criterio propuesto en el capítulo 1, expresión (1.12) y J_2 a una modificación del criterio anterior, donde se elige como operador de matrices la traza en vez del determinante. Con estos dos criterios, trataremos de obtener el conjunto de vectores, $\mathbf{W}$, que definirán el subespacio óptimo de proyección de dimensión menor a la del espacio muestral, ver capítulo 1. Los dos criterios a los que nos referimos son por tanto

$$J_1(\mathbf{W}) = \frac{\left|\tilde{\mathbf{S}}_B\right|}{\left|\tilde{\mathbf{S}}_W\right|} = \frac{\left|\mathbf{W}^T\mathbf{S}_B\mathbf{W}\right|}{\left|\mathbf{W}^T\mathbf{S}_W\mathbf{W}\right|} \qquad \text{y} \qquad J_2(\mathbf{W}) = \frac{Tr\{\tilde{\mathbf{S}}_B\}}{Tr\{\tilde{\mathbf{S}}_W\}} = \frac{Tr\{\mathbf{W}^T\mathbf{S}_B\mathbf{W}\}}{Tr\{\mathbf{W}^T\mathbf{S}_W\mathbf{W}\}}$$

donde $\tilde{\mathbf{S}}_B$ y $\tilde{\mathbf{S}}_W$ son respectivamente las matrices de dispersión inter e intra–clases en la proyección; por lo tanto su dimensión es $(\tilde{d} \times \tilde{d})$. $\mathbf{S}_B$ y $\mathbf{S}_W$ son las mismas matrices en el espacio original de características, siendo su dimensión $(d \times d)$, donde $d > \tilde{d}$. La matriz de la transformación $\mathbf{W}$ que pasa de un espacio $\Re^d$ a otro $\Re^{\tilde{d}}$ es pues, de dimensión $(d \times \tilde{d})$.

A.2.1. Optimización del criterio J_1

Las referencias para optimizar el criterio J_1 están propuestas en [Fukunaga, 1990], aunque el criterio que allí se optimiza es ligeramente diferente, Fukunaga toma como criterio a optimizar $J = \ln|\tilde{\mathbf{S}}_W^{-1}\tilde{\mathbf{S}}_B| = \ln|\tilde{\mathbf{S}}_B| - \ln|\tilde{\mathbf{S}}_W|$, con el que se obtiene resultados equivalentes a los obtenidos con el criterio J_1. Una discusión similar a la propuesta en este apéndice para optimizar el criterio J_1 la realiza [Wilks, 1962].

Comenzaremos por definir la derivada de la expresión $|\mathbf{W}^T\mathbf{S}\mathbf{W}|$ respecto a $\mathbf{W}$, donde $\mathbf{S}$ es una matriz simétrica; en nuestro caso, $\mathbf{S}$ corresponderá a las matrices de dispersión $\mathbf{S}_B$ o $\mathbf{S}_W$. El conjunto de vectores $\mathbf{W}$ define el subespacio de proyección de Fisher y $|\cdot|$ representa el operador determinante de matrices. Remitiéndonos al apéndice A del libro [Fukunaga, 1990] para revisar las derivadas con matrices, se tiene que

$$\frac{\partial\left|\mathbf{W}^T\mathbf{S}\mathbf{W}\right|}{\partial\mathbf{W}} = 2\left|\mathbf{W}^T\mathbf{S}\mathbf{W}\right|\mathbf{S}\mathbf{W}\left(\mathbf{W}^T\mathbf{S}\mathbf{W}\right)^{-1} = 2\left|\tilde{\mathbf{S}}\right|\mathbf{S}\mathbf{W}(\tilde{\mathbf{S}})^{-\mathbf{1}},$$

donde $\tilde{\mathbf{S}} = \mathbf{W^T S W}$, es la transformación de la matriz $\mathbf{S}$ en el nuevo espacio definido por $\mathbf{W}$.

Luego, la derivada del criterio J_1 respecto a $\mathbf{W}$ vendrá dada por

$$\frac{\partial J_1}{\partial\mathbf{W}} = \frac{2}{\left|\tilde{\mathbf{S}}_W\right|^2}\left(\left|\tilde{\mathbf{S}}_B\right|\mathbf{S}_B\mathbf{W}\left(\mathbf{S}_B\right)^{-1}\left|\tilde{\mathbf{S}}_W\right| - \left|\tilde{\mathbf{S}}_W\right|\mathbf{S}_W\mathbf{W}\left(\mathbf{S}_W\right)^{-1}\left|\tilde{\mathbf{S}}_B\right|\right),$$

que sacando factor común $\left|\tilde{\mathbf{S}}_B\right|\left|\tilde{\mathbf{S}}_W\right|$, la expresión simplificada será

$$\frac{\partial J_1}{\partial\mathbf{W}} = 2J_1\left(\mathbf{S}_B\mathbf{W}(\tilde{\mathbf{S}}_B)^{-1} - \mathbf{S}_W\mathbf{W}(\tilde{\mathbf{S}}_W)^{-1}\right).$$

Para hallar el conjunto de vectores $\mathbf{W}$ que optimiza el criterio J_1, se debe cumplir que $\partial J_1/\partial\mathbf{W} = 0$, o lo que es lo mismo

$$\mathbf{S}_B\mathbf{W}(\tilde{\mathbf{S}}_B)^{-1} = \mathbf{S}_W\mathbf{W}(\tilde{\mathbf{S}}_W)^{-1},$$

que reordenándola de forma que a un lado de la igualdad sólo intervengan las matrices de dispersión dependientes explícitamente de $\mathbf{X}$, y al otro lado las correspondientes a la proyección, $\mathbf{Y} = \mathbf{W}^T\mathbf{X}$, nos queda

$$\mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{W} = \mathbf{W}(\tilde{\mathbf{S}}_W)^{-1}\tilde{\mathbf{S}}_B. \tag{A.1}$$

El teorema que se expone a continuación, pertenece al álgebra de matrices y nos servirá para deducir cuál es el vector $\mathbf{W}$ que optimiza la expresión (A.1).

Teorema A.1 *Dos matrices simétricas definidas positivas* $\mathbf{S}_1$ *y* $\mathbf{S}_2$ *pueden ser diagonalizadas simultáneamente de tal forma que se cumpla*

$$\mathbf{Z}^T\mathbf{S}_1\mathbf{Z} = I \qquad y \qquad \mathbf{Z}^T\mathbf{S}_2\mathbf{Z} = \mathbf{\Lambda},$$

donde $\mathbf{Z}$ *es una matriz de autovectores comunes tanto para* $\mathbf{S}_1$ *como para* $\mathbf{S}_2$*, quedando así definidas las matrices diagonales de autovalores para dichos autovectores como la matriz identidad* I *para* $\mathbf{S}_1$ *y la matriz* $\mathbf{\Lambda}$ *para* $\mathbf{S}_2$*. Como consecuencia de lo anterior, las matrices* $\mathbf{\Lambda}$ *y* $\mathbf{Z}$ *serán también la matriz diagonal de autovalores y autovectores de* $\mathbf{S}_1^{-1}\mathbf{S}_2$*, es decir*

$$\mathbf{S}_1^{-1}\mathbf{S}_2\mathbf{Z} = \mathbf{Z}\mathbf{\Lambda}.$$

Aplicando el teorema anterior, las matrices $\tilde{\mathbf{S}}_B$ y $\tilde{\mathbf{S}}_W$ pueden ser diagonalizadas simultáneamente, obteniéndose las matrices diagonales $\mathbf{\Lambda}$ e I que cumplen

$$\mathbf{Z}^T\tilde{\mathbf{S}}_B\mathbf{Z} = \mathbf{\Lambda} \qquad \text{y} \qquad \mathbf{Z}^T\tilde{\mathbf{S}}_W\mathbf{Z} = I, \tag{A.2}$$

donde $\mathbf{Z}$ es una matriz de transformación no singular de dimensión $(\tilde{d} \times \tilde{d})$, con lo cual $\mathbf{Z}^{-1}$ existe. Además considerando el teorema A.1, se cumple que las matrices $\mathbf{\Lambda}$ y $\mathbf{Z}$ contienen los $\tilde{d}$ autovalores y autovectores de $(\tilde{\mathbf{S}}_W)^{-1}\tilde{\mathbf{S}}_B$ o lo que es lo mismo

$$(\tilde{\mathbf{S}}_W)^{-1}\tilde{\mathbf{S}}_B\mathbf{Z} = \mathbf{Z}\mathbf{\Lambda}. \tag{A.3}$$

A pesar de que tanto $\tilde{\mathbf{S}}_W$ como $\tilde{\mathbf{S}}_B$ son matrices simétricas, el producto $(\tilde{\mathbf{S}}_W)^{-1}\tilde{\mathbf{S}}_B$ no tiene que ser necesariamente simétrico. No obstante, como los autovalores $\mathbf{\Lambda}$ y autovectores $\mathbf{Z}$ se han obtenido a partir de la diagonalización simultánea de las dos matrices involucradas $\tilde{\mathbf{S}}_W$ y $\tilde{\mathbf{S}}_B$, entonces se cumple que todos los autovalores son reales y positivos y los autovectores son igualmente reales y ortonormales respecto $\tilde{\mathbf{S}}_W$.

A partir de la expresión (A.3), (A.1) puede escribirse como

$$\mathbf{S}_W^{-1}\mathbf{S}_B\mathbf{W} = \mathbf{W}(\mathbf{Z}\ \mathbf{\Lambda}\ \mathbf{Z}^{-1})$$

o bien

$$\mathbf{S}_W^{-1}\mathbf{S}_B(\mathbf{W}\mathbf{Z}) = (\mathbf{W}\mathbf{Z})\ \mathbf{\Lambda}. \tag{A.4}$$

La ecuación (A.4) muestra que las componentes diagonales de $\mathbf{\Lambda}$ y sus vectores columna asociados, $(\mathbf{W}\mathbf{Z})$ son los que darán lugar a los $\tilde{d}$ autovalores y autovectores de $\mathbf{S}_W^{-1}\mathbf{S}_B$.

Es importante, hacer hincapié en que $\mathbf{S}_W^{-1}\mathbf{S}_B$ tiene d autovalores $(d > \tilde{d})$. Pero, como en realidad, la transformación lineal de Fisher va de un espacio d dimensional a otro $\tilde{d}$, obtener la matriz $\mathbf{W}^*$ que maximiza el criterio J_1 en el espacio $\Re^{\tilde{d}}$ es tan sencillo como tomar los $\tilde{d}$ vectores correspondientes a los $\tilde{d}$ autovectores de mayor

autovalor de $\mathbf{S}_W^{-1}\mathbf{S}_B$. Esto es precisamente lo que se presentó en la sección 1.3 con la expresión (1.13). De hecho, reconfigurando las expresiones en (A.2), éstas se convierten en

$$\tilde{\mathbf{S}}_B = (\mathbf{Z}^T)^{-1}\mathbf{\Lambda}\mathbf{Z}^{-1} \qquad \text{y} \qquad \tilde{\mathbf{S}}_W = (\mathbf{Z}^T)^{-1}I\mathbf{Z}^{-1},$$

y dado que el determinante de una matriz es el producto de sus autovalores, el valor que toma J_1 será

$$J_1(\mathbf{W}) = \frac{\left|\tilde{\mathbf{S}}_B\right|}{\left|\tilde{\mathbf{S}}_W\right|} = \frac{\left|(\mathbf{Z}^T)^{-1}\mathbf{\Lambda}\mathbf{Z}^{-1}\right|}{\left|(\mathbf{Z}^T)^{-1}I\mathbf{Z}^{-1}\right|} = \frac{|\mathbf{\Lambda}|}{|I|} = \prod_{i=1}^{\tilde{d}} \mathbf{\Lambda}_i, \tag{A.5}$$

siendo $\mathbf{\Lambda}_i$ los elementos de la matriz $\mathbf{\Lambda}$. El valor de J_1 será siempre positivo, puesto que es el producto de cantidades positivas. A su vez, será máximo porque hemos tomado los mayores autovalores. Para el caso de resolución de problemas de separación de C clases, vimos en el capítulo 1, sección 1.3 que serían los $\tilde{d}$ autovalores distintos de cero los que maximizan el criterio J_1.

Si se considera el criterio propuesto en [Fukunaga, 1990], $J = \ln|\tilde{\mathbf{S}}_W^{-1}\tilde{\mathbf{S}}_B| = \ln\frac{|\tilde{\mathbf{S}}_B|}{|\tilde{\mathbf{S}}_W|}$, se llega al mismo resultado. En el espacio $\Re^{\tilde{d}}$, serán de nuevo los $\tilde{d}$ mayores autovalores de $\mathbf{S}_W^{-1}\mathbf{S}_B$ los que maximizan el criterio J y su valor en dicho espacio vendrá dado por

$$J(\mathbf{W}) = \sum_{i=1}^{\tilde{d}} \ln \mathbf{\Lambda}_i.$$

Los autovectores correspondientes a esos $\tilde{d}$ autovalores elegidos serán los que definan el subespacio óptimo de proyección.

Los criterios J_1 y J han sido definidos de tal forma que buscar maximizar dichos criterios significa encontrar el conjunto de vectores $\mathbf{W}$ que definen el subespacio óptimo de proyección. Igualmente, podríamos enfocar el problema de tal forma que se elijan criterios con el objetivo de minimizarlos para obtener el vector $\mathbf{W}$ óptimo.

El criterio equivalente a J_1 buscando minimizar sería

$$J_1'(\mathbf{W}) = \frac{\left|\tilde{\mathbf{S}}_W\right|}{\left|\tilde{\mathbf{S}}_B\right|} = \frac{|\mathbf{W}^T\mathbf{S}_W\mathbf{W}|}{|\mathbf{W}^T\mathbf{S}_B\mathbf{W}|}.$$

Al minimizar J_1' se llega una vez más a una expresión idéntica a (A.4) y el valor mínimo de J_1' en el espacio $\Re^{\tilde{d}}$ vendrá dado por los $\tilde{d}$ mayores autovalores de $\mathbf{S}_W^{-1}\mathbf{S}_B$

$$J_1' = \frac{1}{\prod_{i=1}^{\tilde{d}} \mathbf{\Lambda}_i},$$

que como se puede apreciar es mínimo si $\mathbf{\Lambda}_1, \cdots, \mathbf{\Lambda}_{\tilde{d}}$ son los $\tilde{d}$ mayores autovalores de $\mathbf{S}_W^{-1}\mathbf{S}_B$. Sus correspondientes autovectores serán los que definen el subespacio óptimo de proyección.

Como conclusión, tanto maximizar los criterios J_1 o J, como minimizar el criterio J_1' desemboca exactamente al mismo subespacio definido por el conjunto de vectores $\mathbf{W}$ que minimizan J_1' o bien maximizan J_1 o J. El conjunto de vectores $\mathbf{W}$ está definido, en cualquiera de los casos, por los autovectores de $\mathbf{S}_W^{-1}\mathbf{S}_B$ correspondientes a los $\tilde{d}$ mayores autovalores.

A.2.2. Optimización del criterio J_2

En esta sección vamos a seguir el comportamiento del criterio J_2 y veremos cómo es necesario realizar una pequeña modificación a dicho criterio para poder obtener una equivalencia entre J_1 y la nueva versión modificada de J_2

Para obtener el vector $\mathbf{W}$ que maximiza el criterio,

$$J_2(\mathbf{W}) = \frac{Tr(\tilde{\mathbf{S}}_B)}{Tr(\tilde{\mathbf{S}}_W)} = \frac{Tr(\mathbf{W}^T\mathbf{S}_B\mathbf{W})}{Tr(\mathbf{W}^T\mathbf{S}_W\mathbf{W})},$$

será necesario calcular su gradiente. Se tiene que la derivada de $Tr(\mathbf{W}^T\mathbf{S}\mathbf{W})$ respecto a $\mathbf{W}$, cuando $\mathbf{S}$ es una matriz simétrica, viene dada por

$$\frac{\partial Tr(\mathbf{W}^T\mathbf{S}\mathbf{W})}{\partial \mathbf{W}} = 2Tr(\mathbf{S}\mathbf{W}),$$

de aquí se deriva que el gradiente de J_2 respecto $\mathbf{W}$ es

$$\frac{\partial J_2(\mathbf{W})}{\partial \mathbf{W}} = \frac{2}{Tr(\tilde{\mathbf{S}}_W)} Tr\left(\mathbf{S}_B\mathbf{W} - J_2\,\mathbf{S}_W\mathbf{W}\right).$$

Como buscamos optimizar J_2, haremos pues $\nabla J_2 = 0$, o lo que es lo mismo

$$\mathbf{S}_B\mathbf{W} - J_2\,\mathbf{S}_W\mathbf{W} = 0 \qquad \text{ó} \qquad \mathbf{S}_B\mathbf{W} = J_2\,\mathbf{S}_W\mathbf{W},$$

si multiplicamos por $\mathbf{S}_W^{-1}$ en ambos lados de la igualdad se llega a un problema de autovectores y autovalores

$$(\mathbf{S}_W^{-1}\mathbf{S}_B)\mathbf{W} = J_2\,\mathbf{W}.$$

Observando la expresión anterior, vemos que nos enfrentamos a un sistema degenerado, en el vector $\mathbf{W}$ que maximiza J_2 es el autovector de $\mathbf{S}_W^{-1}\mathbf{S}_B$ correspondiente al único autovalor $\lambda = J_2(\mathbf{W})$. Como $\mathbf{S}_B$ es de rango $\tilde{d}$, el producto $\mathbf{S}_W^{-1}\mathbf{S}_B$ también es de rango $\tilde{d}$, lo que implica que la matriz $\mathbf{W}$ estará formada por $\tilde{d}$ vectores independientes que son los autovectores de $\mathbf{S}_W^{-1}\mathbf{S}_B$. La corrección para romper la degeneración la encontramos en [Devijver y Kittler, 1982], donde se propone

que para obtener la solución completa se sustituya el criterio J_2 anterior por el siguiente

$$J_2(\mathbf{W}) = \frac{Tr(\mathbf{W}^T\mathbf{S}_B\mathbf{W})}{Tr(\tilde{\mathbf{\Lambda}}\mathbf{W}^T\mathbf{S}_W\mathbf{W})}. \tag{A.6}$$

La matriz $\tilde{\mathbf{\Lambda}}$ es una matriz diagonal de dimensión $\tilde{d} \times \tilde{d}$, cuyos elementos son proporcionales a los autovalores de $(\mathbf{S}_W^{-1}\mathbf{S}_B)$, su expresión es: $\tilde{\mathbf{\Lambda}} = \mathbf{\Lambda}/\lambda_1$, donde λ_1 es el autovalor mayor de $(\mathbf{S}_W^{-1}\mathbf{S}_B)$. De este modo, el elemento $\tilde{\lambda}_1$ es la unidad y el resto de los elementos cumplen que

$$\tilde{\lambda}_1 = 1 > \tilde{\lambda}_2 > \cdots > \tilde{\lambda}_{\tilde{d}}$$

De aquí se deduce que la matriz $\mathbf{W}$ está formada por los $\tilde{d}$ autovectores de $(\mathbf{S}_W^{-1}\mathbf{S}_B)$ cuyo autovalor es distinto de cero $\lambda_1 > \lambda_2 > \cdots > \lambda_{\tilde{d}} > \lambda_{\tilde{d}+1} = \cdots = \lambda_d = 0$

$$(\mathbf{S}_W^{-1}\mathbf{S}_B)\mathbf{W} = \mathbf{\Lambda}\mathbf{W}.$$

Para demostrarlo, comenzaremos por definir la derivada de $Tr(\tilde{\mathbf{\Lambda}}\mathbf{W}^T\mathbf{S}\mathbf{W})$ respecto a $\mathbf{W}$ cuando $\mathbf{S}$ es una matriz simétrica

$$\frac{\partial Tr(\tilde{\mathbf{\Lambda}}\mathbf{W}^T\mathbf{S}\mathbf{W})}{\partial \mathbf{W}} = 2Tr(\mathbf{S}\mathbf{W}\tilde{\mathbf{\Lambda}}).$$

El gradiente del criterio J_2 será pues

$$\frac{\partial J_2(\mathbf{W})}{\partial \mathbf{W}} = \frac{2}{Tr(\tilde{\mathbf{\Lambda}}\mathbf{W}^T\mathbf{S}_W\mathbf{W})} Tr\left(\mathbf{S}_B\mathbf{W} - J_2\, \mathbf{S}_W\mathbf{W}\tilde{\mathbf{\Lambda}}\right).$$

El máximo de J_2 se obtiene para $\nabla J_2 = 0$ ó lo que es lo mismo:

$$\mathbf{S}_B\mathbf{W} - J_2\, \mathbf{S}_W\mathbf{W}\tilde{\mathbf{\Lambda}} = 0.$$

Una vez más, volvemos al problema de cálculo de autovalores y autovectores de una matriz

$$(\mathbf{S}_W^{-1}\mathbf{S}_B)\mathbf{W} = J_2\, \mathbf{W}\, \tilde{\mathbf{\Lambda}} \qquad \text{ó} \qquad (\mathbf{S}_W^{-1}\mathbf{S}_B)\mathbf{W} = J_2\, \frac{\mathbf{\Lambda}}{\lambda_1}\, \mathbf{W}.$$

Como J_2 es el mayor autovalor, esto es λ_1, queda demostrado que los vectores $\mathbf{W}$ que maximizan el criterio J_2 son los autovectores de $\mathbf{S}_W^{-1}\mathbf{S}_B$ cuyos autovalores son distintos de cero.

Con esto hemos demostrado que dentro del contexto de discriminantes de Fisher, tanto si elegimos el criterio J_1 como la versión modificada del criterio J_2, expresión (A.6), el subespacio de proyección es exactamente el mismo.

Apéndice B

Diferenciación de Operadores de Matrices

En este apéndice, vamos a deducir la derivada del determinante de una matriz cuadrada de dimensión r y dependiente de un conjunto de parámetros $\mathbf{W}$

$$J(\mathbf{W}) = |\mathbf{F}(\mathbf{W})|.$$

Comenzaremos por recordar que el determinante puede ser expresado en términos de los elementos de cualquier fila o columna y sus correspondientes cofactores, por ejemplo si consideramos la fila i–ésima tendríamos

$$J(\mathbf{W}) = \sum_{j=1}^{r} F_{ij}(\mathbf{W}) F_{ij}^{*}(\mathbf{W}), \tag{B.1}$$

donde $F_{ij}(\mathbf{W})$ es el elemento (i,j)–ésimo de la matriz $\mathbf{F}(\mathbf{W})$ y $F_{ij}^{*}(\mathbf{W})$ es el cofactor de $F_{ij}(\mathbf{W})$. Además, $F_{ij}^{*}(\mathbf{W})$ es el elemento (i,j)–ésimo de la matriz adjunta de $\mathbf{F}(\mathbf{W})$, que la denominaremos por $\mathbf{F}^{*}(\mathbf{W})$

$$\mathbf{F}^{*}(\mathbf{W}) = \begin{pmatrix} F_{11}^{*}(\mathbf{W}) & \dots & F_{1r}^{*}(\mathbf{W}) \\ F_{21}^{*}(\mathbf{W}) & \dots & F_{2r}^{*}(\mathbf{W}) \\ \vdots & \ddots & \vdots \\ F_{r1}^{*}(\mathbf{W}) & \dots & F_{rr}^{*}(\mathbf{W}) \end{pmatrix},$$

que cumple la relación $\mathbf{F}^{*}(\mathbf{W}) = |\mathbf{F}(\mathbf{W})|\ \mathbf{F}^{-1}(\mathbf{W})$.

Aplicando la regla de la cadena en derivadas parciales, la derivada de $J(\mathbf{W})$ respecto al elemento w_{kl} de $\mathbf{W}$ viene dada por

$$\frac{\partial J(\mathbf{W})}{\partial w_{kl}} = \sum_{i=1}^{r} \sum_{j=1}^{r} \frac{\partial J(\mathbf{W})}{\partial F_{ij}(\mathbf{W})} \frac{\partial F_{ij}(\mathbf{W})}{\partial w_{kl}} \tag{B.2}$$

A partir de (B.1) se tiene que

$$\frac{\partial J(\mathbf{W})}{\partial F_{ij}(\mathbf{W})} = F^*_{ij}(\mathbf{W}),$$

de este modo, la expresión (B.2) puede reescribirse como

$$\frac{\partial J(\mathbf{W})}{\partial w_{kl}} = \sum_{i=1}^{r}\sum_{j=1}^{r} F^*_{ij}(\mathbf{W})\ \frac{\partial F_{ij}(\mathbf{W})}{\partial w_{kl}}. \tag{B.3}$$

En el caso de que la matriz $\mathbf{F}(\mathbf{W})$ sea una matriz simétrica ($\mathbf{F}(\mathbf{W}) \equiv \mathbf{S}(\mathbf{W})$), la expresión anterior puede simplificarse considerablemente. Para demostrarlo vamos a considerar el producto de dos matrices, $\mathbf{C} = \mathbf{AB}$, donde el elemento c_{ij} viene dado por $c_{ij} = \sum_{k=1}^{r} a_{ik}b_{kj}$ y la traza de la matriz $\mathbf{C}$ será

$$Tr(\mathbf{C}) = \sum_{i=1}^{r} c_{ii} = \sum_{i=1}^{r}\sum_{k=1}^{r} a_{ik}b_{ki}.$$

Observando la expresión (B.3), tenemos que los subíndices se comportan de la siguiente forma $\sum_{i=1}^{r}\sum_{k=1}^{r} a_{ik}b_{ik}$, y sólo son idénticos a la traza del producto de dos matrices cuando la matriz $\mathbf{B}$ es simétrica ($b_{ik} = b_{ki}$). En el caso que nos ocupa, cuando derivamos el determinante de una matriz simétrica se cumple que la derivada de la matriz también es simétrica, entonces la expresión (B.3) puede expresarse en forma matricial como

$$\frac{\partial J(\mathbf{W})}{\partial w_{kl}} = Tr\left(\mathbf{S}^*(\mathbf{W})\ \frac{\partial \mathbf{S}(\mathbf{W})}{\partial w_{kl}}\right), \tag{B.4}$$

o lo que lo mismo, sabiendo que $\mathbf{S}^*(\mathbf{W}) = |\mathbf{S}(\mathbf{W})|\ \mathbf{S}^{-1}(\mathbf{W})$, se tiene que

$$\frac{\partial J(\mathbf{W})}{\partial w_{kl}} = \frac{\partial |\mathbf{S}(\mathbf{W})|}{\partial w_{kl}} = |\mathbf{S}(\mathbf{W})|\ \ Tr\left(\mathbf{S}^{-1}(\mathbf{W})\ \frac{\partial \mathbf{S}(\mathbf{W})}{\partial w_{kl}}\right). \tag{B.5}$$

Hemos sustituido la matriz $\mathbf{F}$ por $\mathbf{S}$ para que quede más claro que las expresiones (B.4) y (B.5) son sólo válidas cuando se trata de la derivada del determinante de una matriz simétrica.

Referencias

Bibliografía

[Amari, 1990] Amari, S. (1990). *Differential-Geometrical Methods in Statistics*, volumen 28 de *Lecture Notes in Statistics*. Springer-Verlag, 2 edición.

[Amari, 1998] Amari, S. (1998). Natural gradient works efficiently in learning. *Neural Computation*, 10:251–276.

[Auer et al., 2001] Auer, P., Burgsteiner, H., y Maass, W. (2001). The p–delta rule for parallel perceptrons. *Manuscript submitted for publication*.

[Blake y Merz, 1998] Blake, C. y Merz, C. (1998). UCI repository of machine learning databases.

[Brandt, 1976] Brandt, S. (1976). *Statistical and Computational Methods in Data Analysis*. American Elsevier, second revised edición.

[Cortes y Vapnik, 1995] Cortes, C. y Vapnik, V. (1995). Support–vector networks. *Machine Learning*, 20:273–297.

[Cristianini y Shawe-Taylor, 2000] Cristianini, N. y Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines and other kernel–based learning methods*. Cambridge University Press.

[Devijver y Kittler, 1982] Devijver, P. y Kittler, J. (1982). *Pattern Recognition, A Statistical Approach*. Prentice Hall International.

[Dorronsoro et al., 1997] Dorronsoro, J. R., Ginel, F., Sánchez, C., y Santa Cruz, C. (1997). Neural fraud detection in credit card operations. *IEEE Transaction on Neural Networks*, 8(4):827–834.

[Dorronsoro y González, 2002] Dorronsoro, J. R. y González, A. (2002). Natural gradiente and multiclass NLDA networks. En *Artificial Neural Networks, ICANN 2002*, volumen 2415, págs. 673–678.

[Dorronsoro et al., 2000] Dorronsoro, J. R., González, A., y Santa Cruz, C. (2000). Weight saliency in nlda networks. En *Conference Proceedings Learning'00*, número art21.pdf.

[Dorronsoro et al., 2001a] Dorronsoro, J. R., González, A., y Santa Cruz, C. (2001a). Architecture selection in nlda networks. En *Artificial Neural Networks, ICANN 2001*, volumen 2130, págs. 27–32.

[Dorronsoro et al., 2001b] Dorronsoro, J. R., González, A., y Santa Cruz, C. (2001b). Natural gradiente learning in NLDA networks. En *Connectionist Models of Neurons, Learning Processes and Artificial Intelligence, IWANN 2001*, volumen 2084, págs. 427–434.

[Duch, 2004] Duch, W. (2004). Support vector neural training. *Enviado a IEEE Transactions on Neural Networks.*

[Duda et al., 2001] Duda, R. O., Hart, E., y Stork, D. G. (2001). *Pattern Classification.* Wiley–Interscience.

[Fisher, 1936] Fisher, R. A. (1936). The use of multiple measurement en taxonomic problems. *Ann. Eugenics*, 7:179–188.

[Fukunaga, 1990] Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition.* Academic Press.

[Gallinari et al., 1991] Gallinari, P., Thiria, S., Badran, F., y Fogelman-Soulie, F. (1991). On the relations between discriminant analysis and multilayer perceptrons. *Neural Networks*, 4(3):349–360.

[Geman et al., 1992] Geman, S., Bienenstock, E., y Doursat, R. (1992). Neural network and the bias–variance dilemma. *Neural Computation*, 4(1):1–58.

[Girosi y Poggio, 1989] Girosi, F. y Poggio, T. (1989). Representation of networks: Kolmogorov's theorem is irrelevant. *Neural Computation*, 1:169–176.

[Golden, 1996] Golden, R. (1996). *Matematical Models for Neural Networks Analysis and Design.* MIT Press.

[Hassibi y Stork, 1993] Hassibi, B. y Stork, D. G. (1993). Second order derivatives for network pruning: Optimal brain surgeon. En *Advances in Neural Information Processing Systems*, volumen 5, págs. 164–171. Morgan Kaufmann, San Mateo, C. A.

[Heskes, 2000] Heskes, T. (2000). On "natural"learning and pruning in multilayered perceptrons. *Neural Computation*, 12:1037–1057.

[Kandel et al., 2000] Kandel, E. R., Schwartz, J. H., y Jessell, T. M. (2000). *Principles of neural science.* McGraw-Hill, 4th edición.

[Kolmogorov, 1957] Kolmogorov, A. (1957). On the representation of continuos functions of several variables by superposition of continuos functions of one variable and addition. *Doklady Akademia Nauk SSRR*, 114(5):953–956.

[Kürková, 1991] Kürková, V. (1991). Kolmogorov's theorem is relevant. *Neural Computation*, 3:617–622.

[Kürková, 1992] Kürková, V. (1992). Kolmogorov's theorem and multilayer neural networks. *Neural Networks*, 5:501–506.

[Le Cun et al., 1990] Le Cun, Y., Denker, J. S., y Solla, S. A. (1990). Optimal brain damage. En Lippmann, M. y Touretzky, S., editores, *Advances in Neural Information Processing Systems*, volumen 2, págs. 598–603. Morgan Kaufmann, San Mateo, C. A.

[Manoukian, 1986] Manoukian, E. (1986). *Modern Concepts and Theorems of Mathematical Statistics*. Springer.

[Mao y Jain, 1995] Mao, J. y Jain, A. K. (1995). Artificial neural networks for feature extraction and multivariate data projection. *IEEE Transaction Neural Networks*, 6:296–317.

[Mardia et al., 1989] Mardia, K., Kent, J. T., y Bibby, J. M. (1989). *Multivariate Analysis*. Academic Press.

[Peña, 2002] Peña, D. (2002). *Análisis de datos multivariantes*. McGraw–Hill.

[Polak, 1971] Polak, E. (1971). *Computational Method in Optimization*. New York: Academic Press.

[Press et al., 1992] Press, W. H., Teukolsky, S. A., Vetterling, W. T., y Flannery, B. P. (1992). *Numerical Recipes in C. The Art of Scientific Computing*. Cambridge University Press, segunda edición.

[Rao, 1948] Rao, C. R. (1948). The utilization of multiple measurements in probles of biological classification (with discussion). *Journal Royal Statistical Society Serie B*, 10:159–203.

[Richard y Lippmann, 1991] Richard, M. y Lippmann, R. P. (1991). Neural network classifiers estimate bayesian *a posteriori* probabilities. *Neural Computation*, 3(4):461–483.

[Ripley, 1996] Ripley, B. (1996). *Pattern Recognition and Neural Networks*. Cambrigde University Press.

[Rosenblatt, 1962] Rosenblatt, F. (1962). *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mecanisms*. Washington DC: Spartan.

[Rumelhart et al., 1986] Rumelhart, D. E., Hinton, G. E., y Williams, R. J. (1986). Learning internal representations by error propagation. En D. E. Rumelhart, J. L. M. y the PDP Research Group, editores, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volumen 1, págs. 318–362. Cambridge, MA: MIT Press.

[Santa Cruz y Dorronsoro, 1998] Santa Cruz, C. y Dorronsoro, J. R. (1998). A nonlinear discriminant algorithm for feature extraction and data classification. *IEEE Transaction on Neural Networks*, 9(6):1370–1376.

[Scarselly y Tsoi, 1998] Scarselly, F. y Tsoi, A. C. (1998). Universal approximation using feedforward neural networks: A survey of some existing methods, and some new results. *Neural Networks*, 11(1):15–37.

[Schiffmann et al., 1994] Schiffmann, W., Joost, M., y Werner, R. (1994). *Optimization of the Backpropagation Algorithm for Training Multilayer Perceptrons.* Report, University of Koblenz.

[Stevens, 1987] Stevens, C. (1987). La neurona. En *El cerebro*, Investigación y Ciencia. Scientific American.

[Webb y Lowe, 1990] Webb, A. R. y Lowe, D. (1990). The optimized internal representation of multilayer classifier networks performs nonlinear discriminant analysis. *Neural Networks*, 3:367–375.

[Webb y Lowe, 1991] Webb, A. R. y Lowe, D. (1991). Optimized feature extraction and the bayes decision in feedforward classifier networks. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 13(4):355–364.

[White, 1989] White, H. (1989). Learning in artificial neural networks: a statistical perspective. *Neural Computation*, 1(4):425–464.

[Wilks, 1962] Wilks, S. S. (1962). *Mathematical Statistics.* Probability and Mathematical Statistics–Applied. Wiley, New York.