UNIVERSIDAD NACIONAL DE CÓRDOBA

Facultad de Ciencias Económicas, Facultad de Ciencias Agropecuarias Y Facultad de Matemática, Astronomía y Física

TÍTULO

CLASIFICACION SUPERVISADA CON REDES NEURONALES. UN CASO DE APLICACIÓN.

Para optar al grado de :

MAGÍSTER EN ESTADÍSTICA APLICADA

Autor: María Inés Stimolo

Contadora Pública

Año 2005

COMISION ASESORA DE TESIS

Director: Dr . Sergio A. Cannas.

Miembros :

Dr . Sergio A. Cannas

Dra Margarita Díaz

.

Dra Ana Silvia Haedo

Fecha de aprobación de Tesis: 21/02/2005

RESUMEN

Uno de los objetivos del área de Marketing de una empresa es identificar los clientes que tienen alta probabilidad de abandonar el servicio prestado por la misma, los cuales son considerados clientes de riesgo.

En este trabajo se ha seleccionado una muestra con las características de alrededor de 9700 clientes extraída de una base de datos de una empresa de telefonía móvil, a partir de la cual se estima una función discriminante que permita encontrar la probabilidad de que un cliente abandone el servicio, como así también identificar factores de riesgo que permitan predecir si un cliente abandonará el mismo.

La eficacia del modelo es evaluada estimando la tasa de error, medida como la proporción de individuos mal clasificados en una muestra test, la cual es independiente al conjunto de datos considerado para la elaboración del modelo.

Entre los múltiples enfoques disponibles en la actualidad para construir reglas de clasificación se han seleccionado el Modelo de Regresión Logística y las Redes Neuronales, los que dieron como resultado un porcentaje de error de clasificación muy similar (alrededor del 12%).

Palabras claves: Decisiones empresariales. Discriminante Logístico. Redes neuronales. Percpetron multicapa. Tasas de Error.

Autor: Cra. María Inés Stimolo

Director: Dr. Sergio Alejandro Cannas

Dirección postal: Maracaibo 850 – Barrio Residencial América – Córdoba (CP 5012)

Dirección electrónica: mstimolo@eco.unc.edu.ar; stimolomarines@hotmail.com.

SUPERVISED CLASSIFICATION WITH NEURAL NETWORKS. A REAL APLICATTION.

ABSTRACT

One of the goals in many companies' Marketing departament is to identify customers with high probability to cancel the service provided by them, who are considered hazardous customers. In this work we take a random sample of the caracteristics from about 9700 customers from a mobile phone company's database. Our objective is estimate a discriminate function, to determine the probability that customers may cancel the service and identify risk factors to predict if they may leave the firm.

The model's effectiveness is evaluated by estimating the error ratio, calculated as the proportion of wrong classified individuals in a test sample, which should be independent of the complete data set, considered for the models construction.

Among the different available approaches to build classification rules, we selected the Logistic Regression Model and the Neural Networks. Both gave as a result a very similar classification error ratio about 12%.

KEY WORDS: Neural Networks, Multilayer perceptron, Logistic Discrimination, Customers decisions.

Índice General

Introducción

En un problema de decisión, se dispone de un conjunto de alternativas posibles de las cuales el sujeto decisor debe seleccionar por lo menos una. Esta selección se logra fijando criterios de valor que le permiten a quien decide, establecer si una selección es mejor que otra. Dentro de los problemas de decisión están aquellos conocidos como problemas de clasificación supervisada, en los que conocido un individuo u objeto y sus características debe decidirse su pertenencia a un conjunto o grupo previamente determinado a través de algún criterio. Estos problemas tienen múltiples aplicaciones en numerosas disciplinas, entre las que se encuentran la clasificación de organizaciones según alguna característica de las mismas como rentabilidad, dinamismo, etc; en las empresas interesa clasificar de alguna manera su clientela (activos o cancelados, adhirió a la campaña de comercialización o no, etc.), o poder clasificar grupos de alumnos (entre desertores o no), grupos de población (según el empleo, el estado de pobreza), entre muchas más que se podrían nombrar. En todos los casos interesa encontrar un modelo que permita clasificar correctamente un nuevo individuo u objeto, siendo de interés en algunas aplicaciones determinar los factores que explican porqué el individuo u objeto pertenece a un grupo determinado.

La estadística ha avanzado sobre este tipo de problemas, a través de distintas técnicas, inicialmente con la función Discriminante Lineal, luego extendido al Modelo Cuadrático, desarrollando en la última década el Discriminante Regularizado. Al mismo tiempo, se incorporaron para el abordaje de estos problemas métodos no paramétricos como el Método Kernel, el Vecino más cercano y los Arboles de Clasificación.

El objetivo de este trabajo es comparar los resultados obtenidos sobre un mismo conjunto de datos aplicando el discriminante logístico (método muy utilizado para este tipo de problemas con buenos resultados); y las redes neuronales (método innovador con el cual se está experimentando).

Entre las aplicaciones de este tipo de problemas, se destacan los estudios que realizan las empresas sobre el comportamiento de su clientela. En este sentido, les interesa determinar los motivos por los que sus clientes las eligen, o toman la decisión de abandonarlas, que permitan al área de marketing delinear estrategias para no perder clientes de su cartera y mantener su participación en el mercado.

A partir de ciertas características conocidas de los clientes de una empresa de servicios, se pretende estimar la regla que permita clasificarlos dentro del grupo de los clientes que piensan dejar de serlo, o no, y de esta manera, predecir el número de clientes que la empresa puede perder; es decir, que permita conocer la probabilidad de pertenencia de los clientes a uno de los grupos:

Grupo 1: clientes que permanecen en la empresa

Grupo 2: clientes que dejarán de comprar o utilizar los servicios de la empresa.

En otros términos, se debe estimar la probabilidad que un individuo pertenezca a un grupo, dado un conjunto de características que lo identifican.

Para construir una regla de clasificación se parte de una muestra representativa de clientes de la empresa (muestra de entrenamiento). La regla de clasificación que se selecciona será aquella que en su etapa de generalización a la población, clasifique los clientes con el mínimo error. Para testear los resultados y comparar los porcentajes de clientes mal clasificados por ambos métodos, se selecciona una muestra independiente de la que se utilizó en la etapa de estimación (muestra test).

En el capítulo 1 se desarrolla el concepto de clasificación desde el punto de vista estadístico y de la teoría general del aprendizaje. En el capítulo 2, se presenta el modelo más simple de redes neuronales, el Perceptron simple, desarrollando sus algoritmos de aprendizaje y el teorema que asegura su convergencia. Siguiendo con el perceptron simple, en el capítulo 3 se desarrolla el Discriminante logístico como un caso particular, presentando el modelo logístico binario y considerando la interpretación de los coeficientes estimados. En este capítulo se propone también un análisis previo de las variables a ingresar en el modelo y métodos para la evaluación final del modelo obtenido.

Sin embargo, el perceptron simple, y el modelo logístico como un caso particular, resultan una buena regla de clasificación sólo si los grupos son linealmente separables. La separabilidad lineal requiere que los grupos estén lo suficientemente separados uno de otro para asegurar un hiperplano que divida las regiones de decisión. Estas críticas fueron hechas por Minsky and Papert (1969). Ellos demostraron con rigurosidad matemática las limitaciones del perceptron simple para generalizar sus resultados a todo tipo de problemas. Sugieren como línea de investigación la extensión a sistemas con variables intermedias. Con el auge de la compu-

tación a mediados de la década del 80 aparece el Perceptron multicapa como una extensión del perceptron simple, el que se presenta en el capítulo 4, como un modelo más complejo de redes neuronales. Por último, se aplican los modelos a un conjunto de datos y se comparan los resultados obtenidos, los cuales están presentados y analizados en el capítulo 5.

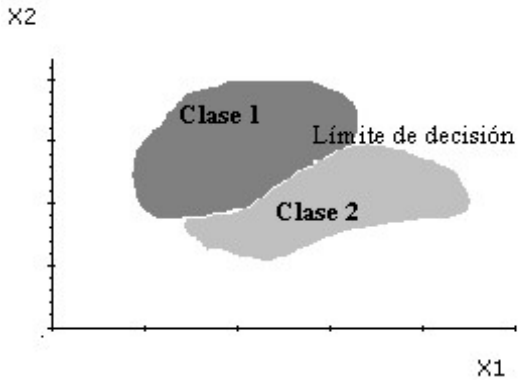
Capítulo 1

Relación de la teoría estadística con la Teoría General del Aprendizaje.

1.1 Introducción estadística a los problemas de clasificación supervisada.

En problemas de clasificación supervisada, se definen clases o grupos, y a partir de una muestra de objetos de los que se conoce el grupo de pertenencia, se construye una regla que permita asignarlos al mismo. Esta regla, puede ser utilizada en una etapa posterior, para asignar nuevos objetos -cuyo grupo de pertenencia es desconocido- a un grupo. Considerando k clases 1,2.....,K y ciertas características de cada objeto representadas en un vector $\mathbf{x}$ de dimensión p, clasificar un objeto significa tomar una de las k posibles decisiones[1] sobre la base de un valor observado $\mathbf{x} = x$. Los vectores de características de la clase k se distribuyen de acuerdo a una densidad $f(\mathbf{x}/k)$. La regla de clasificación elegida, permite obtener probabilidades $P(k/\mathbf{x})$ que constituyen los valores utilizados para tomar la decisión de asignar el objeto al grupo k. Se simboliza como π_k, a la proporción de casos de la clase k en la población de estudio, cuyo valor no siempre es conocida.

[1] Algunos autores como Ripley (1999) menciona $k + 2$ posibles decisiones $1, 2, ..., K, D,\ O.$, definiendo k clases, D significa que ´está en duda', posiblemente se pospone la clasificación hasta que se determinen nuevas características, y O representa un caso atípico, un objeto que definitivamente no pertenece a ninguna de las K clases.



Figura˜1-1: Regiones de decisión en el espacio de variables

Una regla de clasificación asigna cada punto del espacio *Rp* (correspondiente a las características del conjunto de ejemplos), a una de las k clases. Se puede considerar al espacio, dividido en k regiones de decisión $R_1, ..., R_K$, de tal forma que un punto que cae en la región R_k, es asignado a la clase k. Los límites entre estas regiones son conocidos como *límites de decisión.*

En clasificación supervisada, es importante una correcta selección del conjunto de ejemplos para los cuales el vector de características $\mathbf{x}$, y la clase k de pertenencia son conocidos. La regla o modelo de clasificación establecida para decidir la clase de pertenencia, a partir de los ejemplos, debe ser aquella que presenta el menor número de errores. Un error de clasificación ocurre, cuando se asigna un objeto a la clase k que en realidad pertenece a la clase j, o viceversa.

En el caso que la densidad $f(\mathbf{x}/k)$ y la proporción de casos de la clase k en la población π_k sean conocidas, es posible construir un procedimiento de clasificación con propiedades óptimas demostradas. Utilizando el **Teorema de Bayes,** se puede encontrar la probabilidad de que un objeto pertenezca a la clase k, una vez que se ha observado el vector de características $\mathbf{x}$ (probabilidad a posteriori) como:

$$P(k/\mathbf{x}) = \frac{f(\mathbf{x}/k)\pi_k}{f(\mathbf{x})} \tag{1.1}$$

donde $f(\mathbf{x}) = \sum_{i=1}^{K} f(\mathbf{X}/i)\pi_i$ es la función de densidad de $\mathbf{x}$, independientemente de la clase. Esto asegura que estas probabilidades sumen uno, en símbolos $\sum_{i=1}^{K} P(i/\mathbf{x}) = 1$

Para minimizar la probabilidad de objetos mal clasificados, el vector $\mathbf{x}$ es asignado a la clase

i si $P(i/\mathbf{X}) > P(j/\mathbf{X})$ para todo $i \neq j$, lo cual puede ser expresado como:

$$P(\mathbf{X}/i)\pi_i > P(\mathbf{X}/j)\pi_j \text{ para todo } i \neq j. \tag{1.2}$$

La probabilidad del error se puede calcular como:

$$P(E) = \sum P(\mathbf{x} \in i, j) \text{ para todo } i = 1, 2, 3..., K;\ j = 1, 2, 3....., K \text{ siendo } i \neq j$$

La elección del límite de decisión, debe coincidir con el valor de X que minimiza la probabilidad de mal clasificación, lo que es equivalente a asignar cada objeto a la clase que tiene mayor probabilidad a posteriori, como en (1.2). De esta manera, se sustenta la decisión de clasificación sobre el peso relativo de las probabilidades.

Es posible, reformular el proceso de clasificación en términos de un conjunto de **funciones discriminantes** $y_1(\mathbf{x}), ..., y_k(\mathbf{x})$, de tal manera que un vector $\mathbf{x}$, es asignado a la clase k si $y_i(\mathbf{x}) > y_j(\mathbf{x})$ para todo $i \neq j$. La regla de decisión que permite minimizar el error de mal clasificación, puede ser considerada en términos de funciones discriminantes, simplemente eligiendo $y_k(\mathbf{x}) = P(k/\mathbf{x}) = f(\mathbf{x}/k)\pi_k$ en forma equivalente .

En las funciones discriminantes, importan las magnitudes relativas para la determinación de las clases, esto permite reemplazar $y_k(\mathbf{x})$ por una función monótona, lo que no afecta la función de clasificación. De esta manera, se pueden escribir las funciones discriminantes de la siguiente forma:

$$y_k(\mathbf{x}) = \ln f(\mathbf{x}/k) + \ln \pi_k \tag{1.3}$$

Los límites de decisión, cuando dos regiones de decisión son contiguas, no están afectados por transformaciones monótonas de las funciones, ya que los mismos están dados al igualar las funciones discriminantes $y_i(\mathbf{x}) = y_j(\mathbf{x})$.

Si se asume una forma específica de la función de densidad $f_x(\mathbf{x})$, estamos ante un **método paramétrico**. En este caso, la distribución más utilizada por poseer un número considerado de propiedades analíticas y estadísticas, es la **distribución Normal**. Si el vector $\mathbf{x}$ es de dimensión p, la función de probabilidad para cada clase (considerando que son independientes entre sí) es:

$$f_k(\mathbf{X}) = \frac{1}{(2\pi)^{p/2}\,|\boldsymbol{\Sigma}_k|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_{k)^t}\boldsymbol{\Sigma}_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)\right\} \tag{1.4}$$

donde la media es el vector $\boldsymbol{\mu}_k$ y $\boldsymbol{\Sigma}_k$ la matriz de covarianzas. Dada esta particular función de densidad de los datos, la expresión (1.3) puede escribirse como:

$$y_k(\mathbf{X}) = -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^t\boldsymbol{\Sigma}_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_k| + \ln\pi_k \tag{1.5}$$

Si las matrices de covarianza son iguales ($\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$), la expresión (1.3) se simplifica, además si $\boldsymbol{\Sigma}$ es simétrica, su inversa también es simétrica, por lo que se verifica la siguiente igualdad:

$$\mathbf{x}^t\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k = \boldsymbol{\mu}_k^t\boldsymbol{\Sigma}^{-1}\mathbf{x} \tag{1.6}$$

y esto permite expresar $y_k(\mathbf{X})$ como:

$$y_k(\mathbf{X}) = \mathbf{w}_k^t\mathbf{x} + w_{k0} \tag{1.7}$$

donde $\mathbf{w}_k^t = \boldsymbol{\mu}_k^t\boldsymbol{\Sigma}^{-1}$ y $w_{k0} = -\frac{1}{2}\boldsymbol{\mu}_k^t\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k + \ln\pi_k$

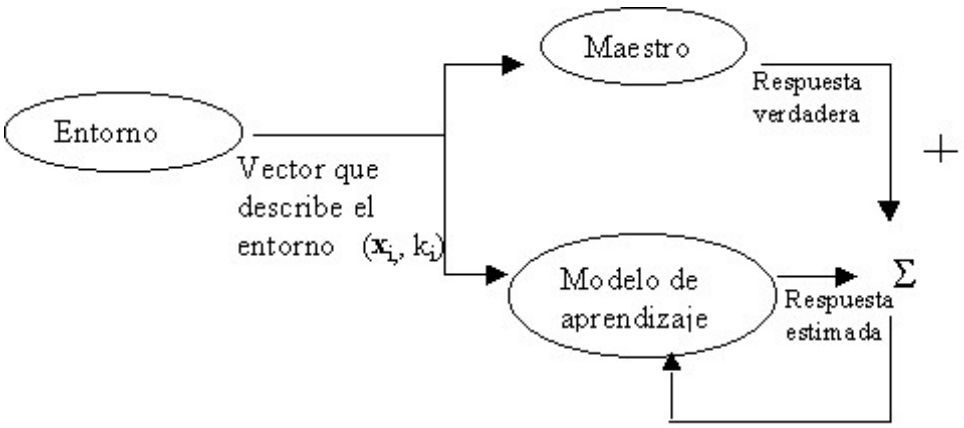
El método de máxima verosimilitud, entre otros, permite estimar los parámetros. Considerando una muestra de n datos, los mismos resultan:

$$\begin{aligned} \hat{\boldsymbol{\mu}}_k &= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i \\ \text{y } \hat{\boldsymbol{\Sigma}} &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i - \hat{\boldsymbol{\mu}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}})^t. \end{aligned}$$

1.2 Clasificación supervisada desde la Teoría General del aprendizaje.[2]

Un modelo de aprendizaje supervisado, como se muestra en la figura 1.2, consiste en tres componentes interrelacionadas:

[2] **Este punto ha sido desarrollado en base al capítulo 2 de Haykin (1994).**



Figura˜1-2: Modelo de aprendizaje supervisado

1- El **ambiente o entorno** con una desconocida distribución de probabilidad $F(\mathbf{x})$, siendo $\mathbf{x}$ el vector de características del mismo.

2- El **maestro**, representado por un conjunto de N ejemplos $(\mathbf{x}_i, k_i)$ para $i = 1, 2, ..., N$, siendo k_i la respuesta verdadera de cada ejemplo. Entre $\mathbf{x}$ y k, existe una relación dada por una función desconocida $g : \mathbf{x} \longrightarrow k$. Por lo tanto, existe la probabilidad condicionada $P(k/\mathbf{x})$, que es constante pero desconocida.

3-Un **algoritmo** que permite realizar un mapeo de funciones $\mathbf{F}$, cuyas entradas pertenecen al conjunto X, estimando un conjunto de parámetros W que permitan encontrar una respuesta $y = F(\mathbf{x}, \mathbf{w})$.

Este modelo conduce a visualizar al aprendizaje supervisado como un problema de aproximación, ya que significa encontrar la función $F(\mathbf{x}, \mathbf{w})$ que mejor se aproxime a la función deseada $g(\mathbf{x})$.

$L(k, F(\mathbf{x}, \mathbf{w}))$ representa la *función de pérdida,* comparando el valor observado k con el valor de salida obtenido a partir de un algoritmo $y = F(\mathbf{x}, \mathbf{w})$. La expresión más utilizada como función de pérdida, es la siguiente:

$$L(k, F(x, w)) = [k - F(\mathbf{x}, \mathbf{w})]^2 \tag{1.8}$$

El valor esperado de la función de pérdida se denomina *riesgo funcional,* y su expresión

analítica es:

$$R(\mathbf{w}) = \int L(k, F(\mathbf{x}, \mathbf{w}))dP(\mathbf{x}, k), \text{donde}^3 \ P(\mathbf{x}, k) = P(\mathbf{x})P(k/\mathbf{x}) \tag{1.9}$$

El objetivo que se persigue es minimizar el valor de $R(\mathbf{w})$, a partir del principio de minimización del riesgo empírico.

1.2.1 Minimización del riesgo empírico.

La idea básica del método consiste en utilizar el conjunto de N ejemplos de entrenamiento independientes y la función $F(\mathbf{x}, \mathbf{w})$ para construir la función de riesgo empírico.

$$R_{emp}(\mathbf{w}) = \frac{1}{N} \sum_{i=1}^{N} L(k_i, F(\mathbf{x}_i, \mathbf{w})) \tag{1.10}$$

la cual no depende de la distribución de probabilidad desconocida $P(\mathbf{x}, k)$, y puede minimizarse respecto al vector $\mathbf{w}$.

$\mathbf{w}_{emp}$ y $F(\mathbf{x}, \mathbf{w}_{emp})$ representan los pesos y el correspondiente mapeo que minimiza el riesgo funcional empírico; por otro lado, w_0 y $F(\mathbf{x}, \mathbf{w}_0)$ corresponden a los pesos y mapeo que minimizan el riesgo funcional real. Tanto $\mathbf{w}_{emp}$ como $\mathbf{w}_0$ pertenecen al espacio W.

El problema a considerar es la condición bajo la cual el mapeo de solución aproximada $F(\mathbf{x}, \mathbf{w}_{emp})$, se acerca al mapeo de solución deseada, medida como la desigualdad entre $R(\mathbf{w}_{emp})$ y $R(\mathbf{w}_0)$.

Fijado un valor $\mathbf{w} = \mathbf{w}^*$, el riesgo funcional $R(\mathbf{w}^*)$ determina la esperanza matemática de una variable aleatoria definida por $Z_{\mathbf{w}^*} = L(k, F(\mathbf{x}, \mathbf{w}^*))$.

En cambio, el riesgo funcional empírico para el valor fijado $R_{emp}(\mathbf{w}^*)$ es la *media empírica* de la variable $Z_{\mathbf{w}^*}$.

De acuerdo a la Ley de los Grandes Números, cuando el tamaño de la muestra N del conjunto de entrenamiento es suficientemente grande, la media empírica de Z_{w^*} converge a su valor esperado. Este teorema, provee una justificación teórica para el uso del riesgo funcional empírico $R_{emp}(\mathbf{w})$, en lugar de $R(\mathbf{w})$, pero ello no es una razón para suponer, que el vector

[3] Integral de Rieman-Stieljes

$\mathbf{w}_{emp}$ que minimiza el riesgo empírico $R_{emp}(\mathbf{w})$,también minimice el riesgo funcional $R(\mathbf{w})$. Esta condición se puede expresar en forma aproximada como sigue (Haykin1994) :

"Si el riesgo funcional empírico $R_{emp}(\mathbf{w})$ se aproxima al riesgo funcional $R(\mathbf{w})$ uniformemente en $\mathbf{w}$ con una precisión ε, entonces el mínimo de $R_{emp}(\mathbf{w})$ difiere del mínimo de $R(\mathbf{w})$ en una cantidad que no excede 2ε".

Formalmente esto significa que, para algún $\mathbf{w} \in W$ y $\varepsilon > 0$, se debe imponer la siguiente condición

$$\Pr\left\{\sup_w |R(\mathbf{w}) - R_{emp}(\mathbf{w})| > \varepsilon\right\} \longrightarrow 0 \text{ , para } N \longrightarrow \infty \text{ (Vapnik,1982)} \tag{1.11}$$

donde sup significa "supremo de". Si se satisface la condición se dice que hay convergencia uniforme en el vector de pesos $\mathbf{w}$ del riesgo empírico medio a su valor esperado.

Para cualquier valor ε y para $\alpha > 0$, podemos plantear la siguiente desigualdad:

$$\Pr\left\{\sup_w |R(\mathbf{w}) - R_{emp}(\mathbf{w})| > \varepsilon\right\} < \alpha \tag{1.12}$$

o que,

$$\Pr\left\{\sup_w |R(\mathbf{w}) - R_{emp}(\mathbf{w})| < \varepsilon\right\} > 1 - \alpha \tag{1.13}$$

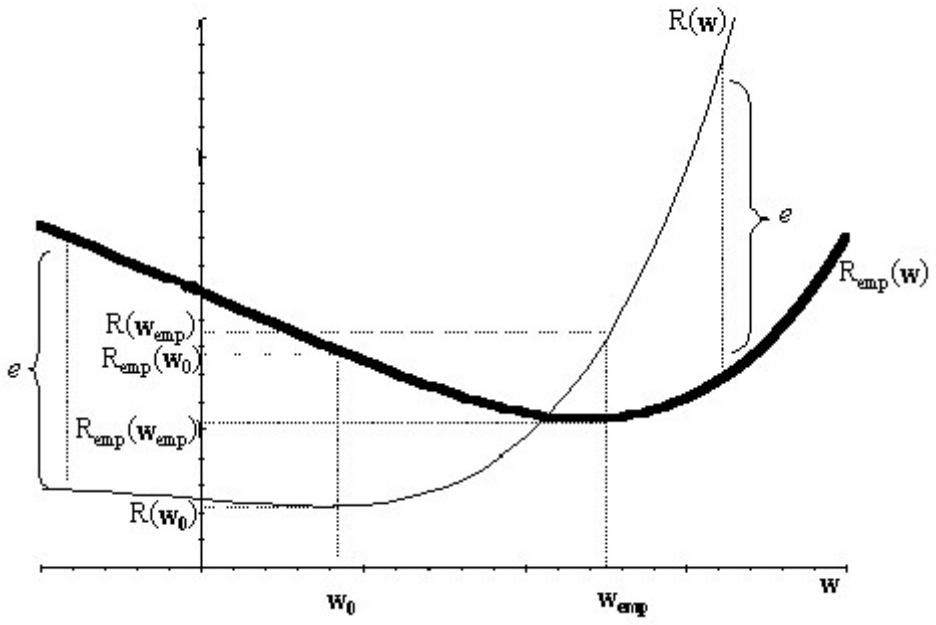
Considerando $\mathbf{w}_{emp}$ y $\mathbf{w}_0$ los puntos mínimos de $R_{emp}(\mathbf{w})$ y $R(\mathbf{w})$ respectivamente, como se puede observar en la figura 1.3, se verifica que:

$$R_{emp}(\mathbf{w}_{emp}) \leq R_{emp}(\mathbf{w}_0) \tag{1.14}$$

La condición (1.13) implica que, las siguientes desigualdades se satisfacen simultáneamente con probabilidad $(1 - \alpha)$:

$$\begin{aligned} R(\mathbf{w}_{emp}) - R_{emp}(\mathbf{w}_{emp}) &< \varepsilon \\ R_{emp}(\mathbf{w}_0) - R(\mathbf{w}_0) &< \varepsilon \end{aligned} \tag{1.15}$$

Sumando miembro a miembro las desigualdades (1.15), y teniendo en cuenta la expresión



Figura˜1-3: Relación entre el riesgo empírico y el riesgo funcional.

(1.14), se puede escribir:

$$R(\mathbf{w}_{emp}) - R(\mathbf{w}_0) < 2\varepsilon \tag{1.16}$$

condición que se satisface con probabilidad $1 - \alpha$. En forma equivalente, se establece la desigualdad $R(\mathbf{w}_{emp}) - R(\mathbf{w}_0) > 2\varepsilon$, con probabilidad α. En símbolos:

$$\Pr\{R(\mathbf{w}) - R_{emp}(\mathbf{w}) > 2\varepsilon\} < \alpha \tag{1.17}$$

Como se expresa en (1.11), el riesgo funcional empírico $R(\mathbf{w}_{emp})$ converge uniformemente al verdadero riesgo funcional $R(\mathbf{w})$, a medida que el tamaño de la muestra se hace infinitamente grande, esto permite afirmar que $R(\mathbf{w}_{emp})$ converge en probabilidad al mínimo valor posible del riesgo real $R(\mathbf{w})$, siendo condición necesaria y suficiente para la consistencia del principio de minimización del riesgo.

La teoría de convergencia define límites sobre la tasa de convergencia, los cuales están basados en un importante parámetro denominado **dimensiónVC**, en honor a quienes los des-

arrollaron Vapnik y Chervonenkis (1971). La dimensión VC es una medida de la capacidad de la familia de funciones de clasificación que pueden ser realizadas por un proceso de aprendizaje.

Para el caso de una respuesta binaria, donde $k = \{0, 1\}$, denotamos como $\mathcal{F}$ a la familia de funciones dicotómicas que pueden ser implementadas en el proceso de aprendizaje, en símbolos:

$$\mathcal{F} = \{F(\mathbf{x}, \mathbf{w}) : \mathbf{w} \in W, F : R^p \rightarrow \{0, 1\}\} \tag{1.18}$$

Se simboliza con S, al conjunto de N puntos en el espacio p dimensional de vectores iniciales: $S = \{1, 2, 3.....N\}$,y $\#S$ representa la cardinalidad de S, es decir, la medida del tamaño del conjunto. El proceso de aprendizaje, debe particionar ese conjunto en dos subespacios disjuntos S_0 y S_1, de forma tal que:

$$F(\mathbf{x}, \mathbf{w})) = \begin{cases} 0 \text{ para } \mathbf{x} \in S_0 \\ 1 \text{ para } \mathbf{x} \in S_1 \end{cases}$$

El conjunto S es separado por $\mathcal{F}$ si todas las divisiones de S ($2^{|S|}$ para el caso dicotómico) pueden ser inducidas por funciones de $\mathcal{F}$.

Designando $\Delta_{\mathcal{F}}(S)$, al número de diferentes divisiones implementadas por el proceso de aprendizaje, y $\Delta_{\mathcal{F}}(S^*)$, al máximo de divisiones que se pueden realizar sobre el conjunto S, la dimensión VC de un conjunto de funciones de clasificación ($\{F(\mathbf{x}, \mathbf{w}) : \mathbf{w} \in W\}$), es el número máximo de elementos de S que puede ser aprendido sin error. En las mayoría de las situaciones prácticas, es difícil evaluar la dimensión VC por medios analíticos, pero se relaciona con los límites de las tasas de convergencia, mostrando que una dimensión VC finita implica tasa de convergencia uniforme.

En clasificación binaria, la función de pérdida toma dos valores:

$$L(k, F(\mathbf{x}, \mathbf{w})) = \begin{cases} 0 \text{ si } F(\mathbf{x}, \mathbf{w}) = k \\ 1 \text{ en otro caso} \end{cases}$$

En este caso, el riesgo funcional $R(\mathbf{w})$ se interpreta como la probabilidad media del error de clasificación (tasa de error), y el riesgo funcional empírico $R_{emp}(\mathbf{w})$ como la frecuencia de errores realizada durante la sesión de entrenamiento (error de entrenamiento).

Dado un vector $\mathbf{w}$ y un $\varepsilon > 0$, la condición establecida por la Ley de los Grandes Números: $\Pr\{|R(\mathbf{w}) - R_{emp}(\mathbf{w})| > \varepsilon\} \longrightarrow 0$ para $N \longrightarrow \infty$, como se expresó en párrafos anteriores, no implica que la regla de clasificación (para un valor de $\mathbf{w}$) que minimiza el vector de

entrenamiento, minimice también la probabilidad media del error de clasificación. Para un conjunto N de entrenamiento suficientemente grande la proximidad entre $R(\mathbf{w})$ y $R_{emp}(\mathbf{w})$ debe cumplir una condición más fuerte dada por la expresión (1.11), donde estamos estableciendo la convergencia uniforme de la frecuencia de errores de entrenamiento hacia su probabilidad media.

Determinada una dimensión VC igual a h, se sostiene la siguiente desigualdad (Vapnik, 1982,1992):

$$\Pr\left\{\sup_w |R(\mathbf{w}) - R_{emp}(\mathbf{w})| > \varepsilon\right\} < \left(\frac{2eN}{h}\right)^h \exp(-\varepsilon^2 N) \tag{1.19}$$

donde la dimensión VC provee un límite a la tasa de convergencia uniforme. El lado derecho de la desigualdad (1.19), tiende a cero a medida que crece N. El factor $\left(\frac{2eN}{h}\right)^h$se denomina "función de crecimiento", cuyo crecimiento no es muy rápido, y al estar multiplicado por la expresión $\exp(-\varepsilon^2 N)$ (que decrece exponencialmente a medida que crece N), hace que el lado derecho de (1.19) tienda a cero.Esta condición se cumple, siempre que la dimensión VC sea finita. Una dimensión VC finita es entonces, una condición necesaria y suficiente para la convergencia uniforme.

Si $\alpha = \left(\frac{2eN}{h}\right)^h \exp(-\varepsilon^2 N\ )$, podemos despejar ε, obteniendo el siguiente resultado:

$$\varepsilon_0(N, h, \alpha) = \sqrt{\frac{h}{N}\left[\ln\left(\frac{2N}{h}\right) + 1\right] - \frac{1}{N}\ln\alpha} \tag{1.20}$$

La expresión (1.20) es un intervalo de confianza que depende de N, h, y α. El límite definido en (1.19) con $\varepsilon = \varepsilon_0(N, h, \alpha)$, es alcanzado para una probabilidad media del error igual a 0,5. Para un valor menor, que son los casos que interesan en la práctica, se considera la siguiente modificación en la desigualdad (1.19)[4]

$$\Pr\left\{\sup_w \left|\frac{R(\mathbf{w}) - R_{emp}(\mathbf{w})}{\sqrt{R(\mathbf{w})}}\right| > \varepsilon\right\} < \left(\frac{2eN}{h}\right)^h \exp(-\frac{\varepsilon^2 N}{4}) \tag{1.21}$$

Se puede encontrar un nuevo intervalo de confianza $\varepsilon_1(N, h, \alpha)$, definido en términos del anterior $\varepsilon_0(N, h, \alpha)$ como sigue:

[4]En la bibliografía se dan diferentes resultados para la expresión (1.19) depende de la desigualdad definida para su derivación, pero todos presentan una forma similar.

$$\varepsilon_1(N,h,\alpha) = 2\varepsilon_0(N,h,\alpha)\left(1+\sqrt{1+\frac{R_{emp}(\mathbf{w})}{\varepsilon_0(N,h,\alpha)}}\right) \tag{1.22}$$

El segundo término de (1.22) depende del error de entrenamiento por lo que cuando $R_{emp}(\mathbf{w}) = 0$, la expresión se reduce a :$\varepsilon_1(N,h,\alpha) = 4\varepsilon_0(N,h,\alpha)$. Con probabilidad $1-\alpha$ se puede establecer, para todos los vectores $\mathbf{w}$,la siguiente desigualdad :

$$R(\mathbf{w}) < R_{emp}(\mathbf{w}) + \varepsilon. \tag{1.23}$$

El límite general para la tasa de convergencia uniforme será:

$$R(\mathbf{w}) < R_{emp}(\mathbf{w}) + \varepsilon_1(N,h,\alpha) \tag{1.24}$$

Cuando *el error de entrenamiento* $R_{emp}(\mathbf{w})$ *es cercano a cero* se tiene :

$$R(\mathbf{w}) \lesssim R_{emp}(\mathbf{w}) + 4\varepsilon_0(N,h,\alpha) \tag{1.25}$$

el cual proporciona un límite bastante preciso para situaciones reales de aprendizaje.

Para un *valor grande del error de entrenamiento cercano a la unidad* el límite será:

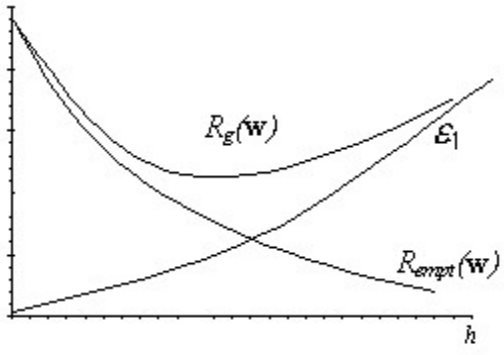
$$R(\mathbf{w}) \lesssim R_{emp}(\mathbf{w}) + \varepsilon_0(N,h,\alpha). \tag{1.26}$$

1.2.2 Minimización del riesgo estructural

El error de entrenamiento, es la frecuencia de errores realizados por el modelo de red durante la sesión de entrenamiento, para un vector de pesos $\mathbf{w}$. El error de generalización, está definido como la frecuencia de errores realizados por el modelo sobre un conjunto nuevo de ejemplos.[5] Se tienen entonces dos errores, $R_{empt}(\mathbf{w})$ y $R_{empg}(\mathbf{w})$, error de entrenamiento y error de generalización respectivamente.

En la teoría de la tasa de convergencia uniforme, se debe establecer con probabilidad $1-\alpha$ para un número de ejemplos $N > h$ (dimensiónVC) y para todas las funciones de clasificación $F(\mathbf{x},\mathbf{w})$, que el error de generalización es menor que un riesgo garantizado definido como la

[5] Aquí se asume que los ejemplos de prueba provienen de la misma población que los ejemplos de entrenamiento.



Figura˜1-4: Comportamiento del riesgo garantizado y sus componentes según la capacidad del modelo

suma de $R_{empt}(\mathbf{w})$ y $\varepsilon_1(N, h, \alpha, R_{empt}(\mathbf{w}))$:

$$R_g(\mathbf{w}) = R_{empt}(\mathbf{w}) + \varepsilon_1(N, h, \alpha, R_{empt}(\mathbf{w})) \tag{1.27}$$

Siendo $\varepsilon_1(N, h, \alpha, R_{empt}(\mathbf{w}))$ el valor definido en la expresión (1.22). Dado un número fijo de N ejemplos de entrenamiento, a medida que se incrementa la capacidad de aprendizaje (o dimensión VC), el error de entrenamiento se reduce , mientras el intervalo de confianza se incrementa en forma monótona . En consecuencia, el riesgo garantizado y el error de generalización tienden a un mínimo. Antes de alcanzar el punto mínimo (Ver figura 4.1) el problema de aprendizaje está determinado sobre la capacidad h del modelo de red, la cual es muy pequeña para una cantidad específica de ejemplos de entrenamiento. Después del punto mínimo, el problema de aprendizaje está determinado por debajo de la capacidad del modelo, por lo que esta es muy grande para el conjunto de entrenamiento.

El objetivo en la resolución de un determinado problema de aprendizaje, es alcanzar el mejor resultado de generalización, dada la capacidad del modelo y la cantidad disponible de ejemplos de entrenamiento.

El método de minimización del riesgo estructural, proporciona un procedimiento sistemático para alcanzar este objetivo controlando la dimensiónVC del modelo. Considerando una familia de funciones de clasificación binaria $\{F(\mathbf{x}, \mathbf{w}) : \mathbf{w} \in W\}$, se definen n subconjuntos con

estructuras anidadas.

$\mathcal{F}_k = \{F(\mathbf{x}, \mathbf{w}) : \mathbf{w} \in W_k\}$, $k = 1, 2, ..., n.$, de forma tal que $\mathcal{F}_1 \subset \mathcal{F}_2 \subset ..... \subset \mathcal{F}_n$, es decir cada subconjunto de funciones está contenido en un conjunto mayor. Para las dimensionesVC correspondientes a cada subconjunto, se satisface que: $h_1 < h_2 < ..... < h_n$.

El método de minimización del riesgo estructural procede de la siguiente forma:

-Se minimiza el riego empírico (el error de entrenamiento) para cada subconjunto, y se selecciona un subconjunto particular $\mathcal{F}^*$que presente el mínimo error más chico.

- Para el subconjunto $\mathcal{F}^*$ se determina el mejor compromiso entre los términos competitivos del riesgo garantizado, es decir el error de entrenamiento y el intervalo de confianza. El objetivo es, encontrar un modelo de red tal que, la dimensión VC disminuya, con el menor incremento posible del error de entrenamiento.

El principio de minimización del riesgo estructural se puede implementar de las siguientes formas:

-Variando la cantidad de parámetros.

-Agregando un parámetro regularizador en la función de riesgo funcional $R(\mathbf{w}, \lambda_k)$

-Reduciendo la dimensión del espacio de variables de entrada, lo cual tiene el efecto de reducir la cantidad de parámetros w_{ij} en el modelo.

1.3 Carácter estadístico del proceso de aprendizaje

El proceso de aprendizaje experimentado por una red neuronal es un proceso estocástico, dado que el mismo proviene de un ambiente aleatorio donde debe codificarse el conocimiento empírico del entorno, es decir las mediciones que caracterizan el fenómeno en estudio. Considerando un fenómeno descrito por un vector $\mathbf{x}$, que representa a un conjunto de variables independientes, y un escalar k para indicar la variable dependiente, con N observaciones de $\mathbf{x}$, denotadas por $\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_N$, y el correspondiente conjunto de observaciones de k, denotados por $k_1, k_2, ...k_N$. Generalmente, no se tiene conocimiento de la relación funcional exacta entre $\mathbf{x}$ y k, la cual se puede escribir como:

$$k = f(\mathbf{x}) + \varepsilon.$$

Siendo $f(\mathbf{x})$ alguna función de $\mathbf{x}$, y ε un error aleatorio esperado que representa lo que se ignora respecto a la dependencia entre k y $\mathbf{x}$.

La expresión (1.18) representa un *modelo estadístico de regresión* donde la función $f(\mathbf{x})$ se define como una esperanza condicional: $f(\mathbf{x}) = E[k/\mathbf{x}]$, la cual representa el valor de k que se obtendrá en promedio dada una particular realización del vector $\mathbf{x}$. Cuando $\varepsilon = 0$, la relación entre $\mathbf{x}$ y k es exacta.

El modelo de regresión se basa en dos propiedades:

1) $E[\varepsilon/\mathbf{x}] = 0$

2) No hay correlación entre le error y la función $f(\mathbf{x})$. Esta propiedad, conocida como el principio de ortogonalidad, expresa que la información útil del vector $\mathbf{x}$, ha sido codificada dentro de la función $f(\mathbf{x})$.En símbolos: $E[\varepsilon f(\mathbf{x})] = 0$.

El modelo de regresión es un modelo matemático cuyo propósito es utilizar el vector $\mathbf{x}$ para explicar o predecir k. Un modelo de redes neuronales provee un mecanismo para cumplir este objetivo, cuya tarea es codificar el conocimiento empírico representado por un conjunto de entrenamiento ($T = \{(\mathbf{x}_i, k_i) \mid i = 1, 2, 3...N\}$), en un conjunto de parámetros. En este contexto, $\mathbf{x}_i, k_i$ representan el vector de entrada y su correspondiente respuesta deseada. Un modelo de red encuentra una respuesta determinada $y = F(\mathbf{x}, \mathbf{w})$ a partir de un vector de parámetros $\mathbf{w}$, el cual se va ajustando en forma iterativa, en respuesta a una señal de error e, definida como la diferencia entre la respuesta deseada k, y la que se obtiene del modelo y. El criterio de optimización, es minimizar la media cuadrática de la señal de error, definiendo la siguiente función de costo:

$$\begin{aligned} L(\mathbf{w}) &= \frac{1}{2}E\left[e^2\right] \\ &= \frac{1}{2}E\left[(k - F(\mathbf{x}, \mathbf{w}))^2\right] \end{aligned} \tag{1.28}$$

La expresión (1.28) se puede reescribir como:

$$L(\mathbf{w}) = \frac{1}{2}E\left[(k - f(\mathbf{x}) + f(\mathbf{x}) - F(\mathbf{x}, \mathbf{w}))^2\right] \tag{1.29}$$

reagrupando y operando algebraicamente se llega a:

$$L(\mathbf{w}) = \frac{1}{2}E\left[(k - f(\mathbf{x}))^2\right] + \frac{1}{2}E\left[(f(\mathbf{x}) - F(\mathbf{x}, \mathbf{w}))^2\right] \tag{1.30}$$

en la fórmula anterior se elimina la esperanza del doble producto, que define la correlación entre los errores ε y las funciones $f(\mathbf{x})$ y $F(\mathbf{x}, \mathbf{w})$, por ser igual a cero.

En la expresión (1.30), el primer término es independiente de $\mathbf{w}$, por lo que el valor $\mathbf{w}_0$ que minimiza la función de costo $L(\mathbf{w})$ es la que minimiza la integral :

$$E\left[(f(\mathbf{x}) - F(\mathbf{x}, \mathbf{w}))^2\right] = \int_{R_p} P(\mathbf{x}) \left(f(\mathbf{x}) - F(\mathbf{x}, \mathbf{w})\right)^2 d\mathbf{x} \tag{1.31}$$

donde $\mathbf{x} \in R_p$ y la función de densidad del vector $\mathbf{x}$ está dada por $P(\mathbf{x})$.

De la expresión (1.30) surge la siguiente desigualdad:

$$L(\mathbf{w}) \geq \frac{1}{2}E\left[(k - E\left[k/\mathbf{x}\right])^2\right] \tag{1.32}$$

es decir, entre todas las funciones de $\mathbf{x}$, el *modelo de regresión* es el mejor estimador de la respuesta deseada k dado un vector $\mathbf{x}$, cuando la función de costo es definida como la media cuadrática.

Podemos decir, que el vector $\mathbf{w}_0$ tiene la propiedad de minimizar la media cuadrática de $F(\mathbf{x}, \mathbf{w})$, aproximándose a la esperanza condicional $f(\mathbf{x}) = E\left[k/\mathbf{x}\right]$.

La distancia al cuadrado entre el resultado de un modelo de regresión y el obtenido por un modelo de red, es decir:

$$\left(f(\mathbf{x}) - F(\mathbf{x}, \mathbf{w})\right)^2 = \left(E\left[k/\mathbf{x}\right] - F(\mathbf{x}, \mathbf{w})\right)^2 \tag{1.33}$$

representa una medida de efectividad de la función $F(\mathbf{x}, \mathbf{w})$, como predictor de k. Una red neuronal, aprende de la información contenida en el conjunto de entrenamiento, y la misma se transfiere a un vector de parámetros $\mathbf{w}$, indicada como $T \longrightarrow \mathbf{w}$

Como $\mathbf{w}$ depende del conjunto T, la función $F(\mathbf{x}, \mathbf{w})$, puede escribirse como $F(\mathbf{x}, T)$. Consideremos el error cuadrático medio de la función $F(\mathbf{x}, T)$, como un estimador de la función de regresión $E\left[k/\mathbf{x}\right]$. En símbolos:

$$E\left[\left(E\left[k/\mathbf{x}\right]-F(\mathbf{x},T)\right)^2\right] \tag{1.34}$$

Sumando y restando $E\left[F(\mathbf{x},T)\right]$,a la expresión (1.34), agrupando convenientemente y desarrollando el cuadrado se obtiene:

$$E\left[\left(E\left[k/\mathbf{x}\right]-F(\mathbf{x},T)\right)^2\right]=E\left[\left(E\left[k/\mathbf{x}\right]-E\left[F(\mathbf{x},T)\right]\right)^2\right]+E\left[\left(E\left[F(\mathbf{x},T)\right]-F(\mathbf{x},T)\right)^2\right] \tag{1.35}$$

$$E\left[\left(E\left[k/\mathbf{x}\right]-F(\mathbf{x},T)\right)^2\right]=\left(E\left[F(\mathbf{x},T)\right]-E\left[k/\mathbf{x}\right]\right)^2+E\left[\left(F(\mathbf{x},T)-E\left[F(\mathbf{x},T)\right]\right)^2\right] \tag{1.36}$$

En la expresión (1.36), el primer término $\left(E\left[F(\mathbf{x},T)\right]-E\left[k/\mathbf{x}\right]\right)^2$ representa el sesgo de la función de aproximación $F(\mathbf{x},T)$ medida respecto a la función de regresión. El segundo término $E\left[\left(F(\mathbf{x},T)-E\left[F(\mathbf{x},T)\right]\right)^2\right]$ es la varianza de la función de aproximación. El cuadrado medio del error de estimación entre la función del modelo de red y la de regresión para un conjunto de entrenamiento dado es la suma de dos componentes: sesgo y varianza.

Para alcanzar buenos resultados, ambos términos deberían ser pequeños. En el modelo de red neuronal, se parte de un determinado conjunto de entrenamiento, y el precio de alcanzar un mínimo sesgo es aumentar la varianza. El aumento del tamaño del conjunto de entrenamiento, permite reducir las dos componentes al mismo tiempo, pero la consecuencia es una lenta convergencia. A veces, dependiendo de cada problema, se puede incorporar el sesgo intencionalmente para reducir la varianza, siempre que ello no afecte al modelo.

Capítulo 2

Redes neuronales

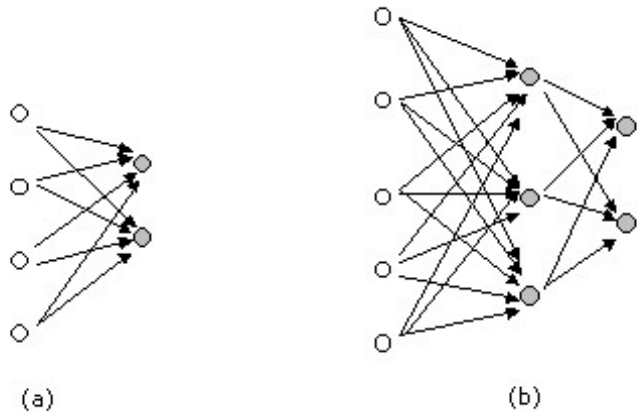
2.1 Concepto de modelo de redes neuronales

Los estudios realizados para entender y modelar las funciones orgánicas del cerebro, sirvieron de motivación para la investigación en redes neuronales. Aunque aún se está muy lejos de conocer el funcionamiento real del cerebro, su estructura básica y procesos internos, sirven como inspiración para el desarrollo de modelos de redes neuronales, los cuales han avanzado sobre otras áreas del conocimiento muy distantes de la neurobiología, en disciplinas como ingeniería, física, tecnología, estadística, etc.[1]

Los modelos de Redes Neuronales son algoritmos para realizar tareas cognitivas, tales como aprendizaje y optimización. Los sistemas de Redes Neuronales Artificiales(ANN), tienen la estructura de simples unidades de procesamientos o nodos conectados por enlaces ponderados, los cuales pueden ser representados como grafos direccionados. Las distintas conexiones entre los nodos determinan diferentes arquitecturas de redes neuronales. En problemas específicos de clasificación, se utilizan conexiones direccionadas hacia delante (feedfoward) como los de la Figura 2.1.

Un modelo de red neuronal está definido como un gráfico dirigido con las siguientes propiedades:

[1] Ver en anexo I la analogía entre estos modelos matemáticos y el modelo biológico y la historia del desarrollo de estos modelos.



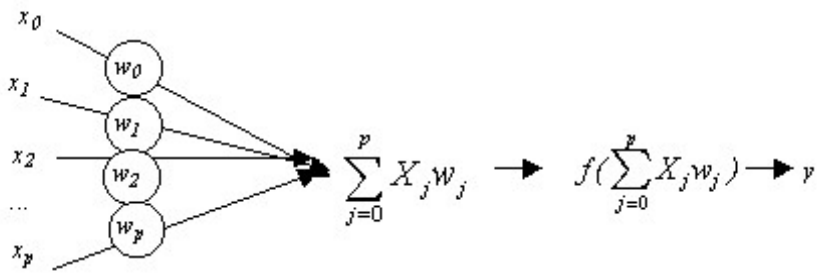
Figura˜2-1: Modelos de Redes Neuronales para clasificación (feedfoward)

1. A cada nodo i se le asocia una variable x_i, denominada *neurona.*
2. Los enlaces entre el nodo de llegada i y el nodo de salida j son denominados *sinapsis,* y están ponderados por un valor real w_{ij} o *peso sináptico.*
3. Un valor real w_0 asociado a cada nodo que recibe el nombre de *umbral de activación*[2]
4. Una *función de activación* para cada nodo, la cual determina el valor del mismo en función de su umbral de activación, la sinapsis de los enlaces entrantes, y el valor de los nodos conectados por esos enlaces. Esta función limita el rango de amplitud del valor que puede asumir la neurona x_i, el cual está normalmente definido en el intervalo [0,1] o [-1,1], y puede ser la *identidad, la escalar, la logística o la tangente hiperbólica*, entre otras.

El primer modelo de red neuronal de la figura 2.1 está formado por un conjunto de nodos de entrada de nominado capa de entrada (4 neuronas) y un conjunto de nodos de salida (3 neuronas); el segundo modelo tiene un una capa de entrada formado por 5 neuronas, un conjunto de nodos intermedio el cual se denomina capa oculta (3 neuronas) y dos neuronas de salida.

Una neurona k se define en términos matemáticos como:

[2] *En caso de tratarse de una función continua, es más adecuado denominarlo sesgo*



Figura~2-2: Representación de una neurona.

$$y_k = f(\sum_{j=1}^{p} w_{kj}x_j - w_{k0}) \tag{2.1}$$

donde $x_1, x_2, ..., x_p$ son las señales de entrada; $w_{k1}, w_{k2}, ..., w_{kp}$ son los pesos sinápticos, w_{k0} es el umbral de activación, y f la función de activación.

Agregando una señal de entrada igual a -1, (2.1) se puede expresar:

$$y_k = f(\sum_{j=0}^{p} w_{kj}x_j) \tag{2.2}$$

siendo $x_0 = -1$, siendo su representación gráfica la de la figura 2.2.

La variable y de salida, representa una función explícita de las variables de entrada. Esto se cumple para todos los modelos de redes neuronales direccionadas hacia adelante. En estos modelos las entradas son propagadas a través de la red, directamente hacia la variable de salida.

La elección de un determinado proceso de aprendizaje está influenciado por la tarea de aprendizaje que se requiera como resultado; si esta tarea es la clasificación supervisada de patrones, para resolverlo la red organiza una sesión de entrenamiento durante la cual se van presentando repetidamente a la red el conjunto de ejemplos con la clase a la cual pertenecen. Después se presenta a la red un ejemplo perteneciente a la misma población que no fue utilizado en el entrenamiento: la tarea que debe cumplir es clasificar correctamente ese patrón.

La ventaja de utilizar redes neuronales para clasificar, es que permite construir fronteras de decisión entre regiones de forma no parámetrica y por ende ofrece un método práctico para resolver problemas de clasificación muy complejos.

Los resultados obtenidos por la red se van contrastando con las respuestas correctas, y la diferencia entre ellos define una señal de error. Los parámetros de la red son ajustados bajo la influencia combinada del vector de entrenamiento y la señal de error. Este ajuste se realiza iterativamente paso a paso hasta que las respuestas obtenidas por la red sean lo más cercanas posibles a las correctas. Cuando esta condición es alcanzada debemos prescindir de los ejemplos y dejar que la red neuronal se enfrente al entorno por sí misma. Dado un adecuado conjunto de ejemplos de entradas y salidas, un algoritmo para minimizar una función de costo de interés, y el tiempo suficiente permitido para realizar el entrenamiento, el sistema de aprendizaje supervisado permite generalmente resultados satisfactorios para la clasificación.

La propiedad más interesante de una red neuronal es su habilidad de aprender desde su entorno y mejorar sus resultados a través del aprendizaje; esto tiene lugar en el tiempo acorde a una medida de error establecida.

En su proceso de aprendizaje:

1)La red neuronal es estimulada desde el entorno

2)La red neuronal sufre cambios como resultado de la estimulación anterior

3)Responde de manera diferente al entorno debido a los cambios ocurridos en su estructura interna

Si consideramos dos nodos x_j y y_k conectados por el peso sináptico w_{kj}, $w_{kj}(n)$ representa el valor del peso sináptico al momento n. Al tiempo n, un ajuste $\triangle w_{kj}(n)$ es aplicado al peso sináptico $w_{kj}(n)$ conduciendo a actualizar el valor $w_{kj}(n+1)$. Esto se puede escribir como:

$$w_{kj}(n+1) = w_{kj}(n) + \triangle w_{kj}(n) \tag{2.3}$$

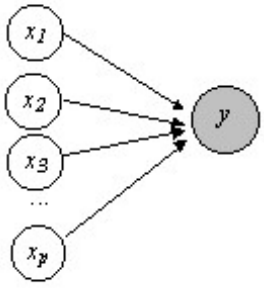
$w_{kj}(n)$ y $w_{kj}(n+1)$ son los viejos y nuevos valores de los pesos sinápticos, luego $w_{kj}(n+1)$ debe ser revaluado.

Las reglas propuestas para resolver el problema de aprendizaje son llamadas *algoritmos de aprendizaje*. Existen diferentes algoritmos de aprendizaje, diferenciándose básicamente por la

manera de realiza el ajuste de $\triangle w_{kj}(n)$.

2.2 Perceptron simple

El modelo más simple de una red neuronal es el perceptron de Rosenblatt o perceptron simple (Rosenblatt, 1962), cuya representación gráfica es la siguiente:



Perceptron simple con una neurona de salida.

Un perceptron simple representa una función de clasificación en dos clases C_1 y C_2, el valor del nodo de salida y resulta según la expresión:

$$y = f(\sum_{j=1}^{p} w_j x_j - w_0)$$

que representa una función discriminante siendo el vector X asignado[3] a C_1 si $y > 0$ y a C_2 si $y < 0$. Esta función divide el espacio R_p de variables en dos hiperplanos. Los coeficientes w_j son estimados de tal modo que minimicen la diferencia entre las salidas y la clasificación correcta, el proceso de búsqueda de estos coeficientes es los que se denomina *entrenamiento o aprendizaje de la red.*

Cuando hay más de un nodo de salida, el perceptron representa una función de clasificación en más de dos grupos, siendo la expresión de cada nodo:

$$y_k = f(\sum_{j=1}^{p} w_{kj} x_j - w_{0k}) = f(\mathbf{w}_k \mathbf{x} - w_{0k})$$

[3]El criterio de asignación puede cambiar para distintas aplicaciones.

2.2.1 Criterio del perceptron

Considerando ahora como función de activación f a la función escalón, la que se simboliza Θ

$$\Theta(a) = \begin{cases} -1 \text{ cuando } a < 0 \\ 1 \text{ cuando } a \leq 0 \end{cases} \tag{2.4}$$

siendo $a = \sum_{j=0}^{p} w_j x_j$, o en su expresión vectorial $a = \mathbf{w}^{'}\mathbf{x}$.

El objetivo que se persigue es encontrar un modelo que realice una clasificación correcta, por lo que es natural definir la función error en términos del número total de errores cometidos en el conjunto de entrenamiento. El critero del perceptron define una función error lineal por partes. Si se asocia cada vector $\mathbf{x}_i$ a su correspondiente k_i para $i = 1, 2, 3, ...N$, es decir considerando N ejemplos o patrones, de tal modo que si $k_i = 1$, el vector pertenece a C_1, y si $k_i = -1$, pertenece a C_2. A partir de la función definida en 2.4, se debe cumplir que $\mathbf{w}^{'}\mathbf{x} > \mathbf{0}$ para vectores que provienen de C_1, y $\mathbf{w}^{'}\mathbf{x} < \mathbf{0}$ para los que provienen de C_2. Una clasificación será correcta cuando $\mathbf{w}^{'}(\mathbf{x}_i k_i) > 0$. Entonces, la función error a minimizar será:

$$E(\mathbf{w}) = -\sum_{\mathbf{x}_i \in \mathbf{N}_e} \mathbf{w}^{'}(\mathbf{x}_i k_i) \tag{2.5}$$

donde $\mathbf{N}_e$ representa el conjunto de vectores $\mathbf{x}_i$ mal clasificados por el actual vector de parámetros $\mathbf{w}$.

Esta función error es conocida como *criterio del perceptron.* $E(\mathbf{w})$ es igual a cero para el caso que que todos los datos estén bien clasificados. A partir de esta función de error es posible definir un simple algoritmo de aprendizaje

$$w_j(n+1) = w_j(n) + \eta x_{j_i} k_i \tag{2.6}$$

siendo η un parámetro mayor que cero, denominado *tasa de aprendizaje*, que controla el ajuste aplicado al vector de parámetros y su valor realiza un cambio de escala sin afectar la separabilidad.

Con un simple ejemplo se puede ver gráficamente (Figura 2.4) como funciona el aprendizaje en un perceptron , considerando cinco ejemplos con sólo dos variables (x_1, x_2) y un umbral

de activación $x_0 = 1$. Los círculos blancos corresponden a los ejemplos de la clase 1 (C_1) y los círculos negros a los de la clase 2 (C_2). Dado un vector inicial de parámetros inicial $\mathbf{w}(0)$, se encuentra el primer límite de decisión. Como se observa en la Figura 2.4 (a), el punto 2 está incorrectamente clasificado. Para corregir este error de clasificación, como el punto está clasificado como C_1, cuando en realidad corresponde a C_2, se busca un nuevo vector de parámetros $\mathbf{w}(1)$, sumando a $\mathbf{w}(0)$ el opuesto del vector que corresponde al punto 2, encontrando de esta manera un nuevo límite de decisión como se muestra en la Figura 2.4 (b) . De esta manera el punto 2 queda correctamente clasificado en C_2, pero ahora están incorrectamente clasificados los puntos 3 y 4 que corresponden a C_1 y están clasificados en C_2. Considerando el vector de menor norma, el cual corresponde al punto 3, se busca un nuevo vector de parámetros $\mathbf{w}(2)$, sumando al vector $\mathbf{w}(2)$ el vector que corresponde al punto 3, el nuevo límite de decisión de la Figura 2.4(c) clasifica correctamente todos los puntos.

Si no existe un límite que separe las clases (un ejemplo clásico presentado por la bibliografía es la disyunción excluyente XOR), el problema no tiene solución, la red no puede llevar a cabo la tarea de aprendizaje independientemente de como sea entrenada. La separabilidad lineal es la condición necesaria para que un problema pueda ser resuelto por un perceptron simple, siendo un problema linealmente separable aquel para el cual puede encontrarse un plano en el espacio de la variables que separe los ejemplos de C_1, de los de C_2.

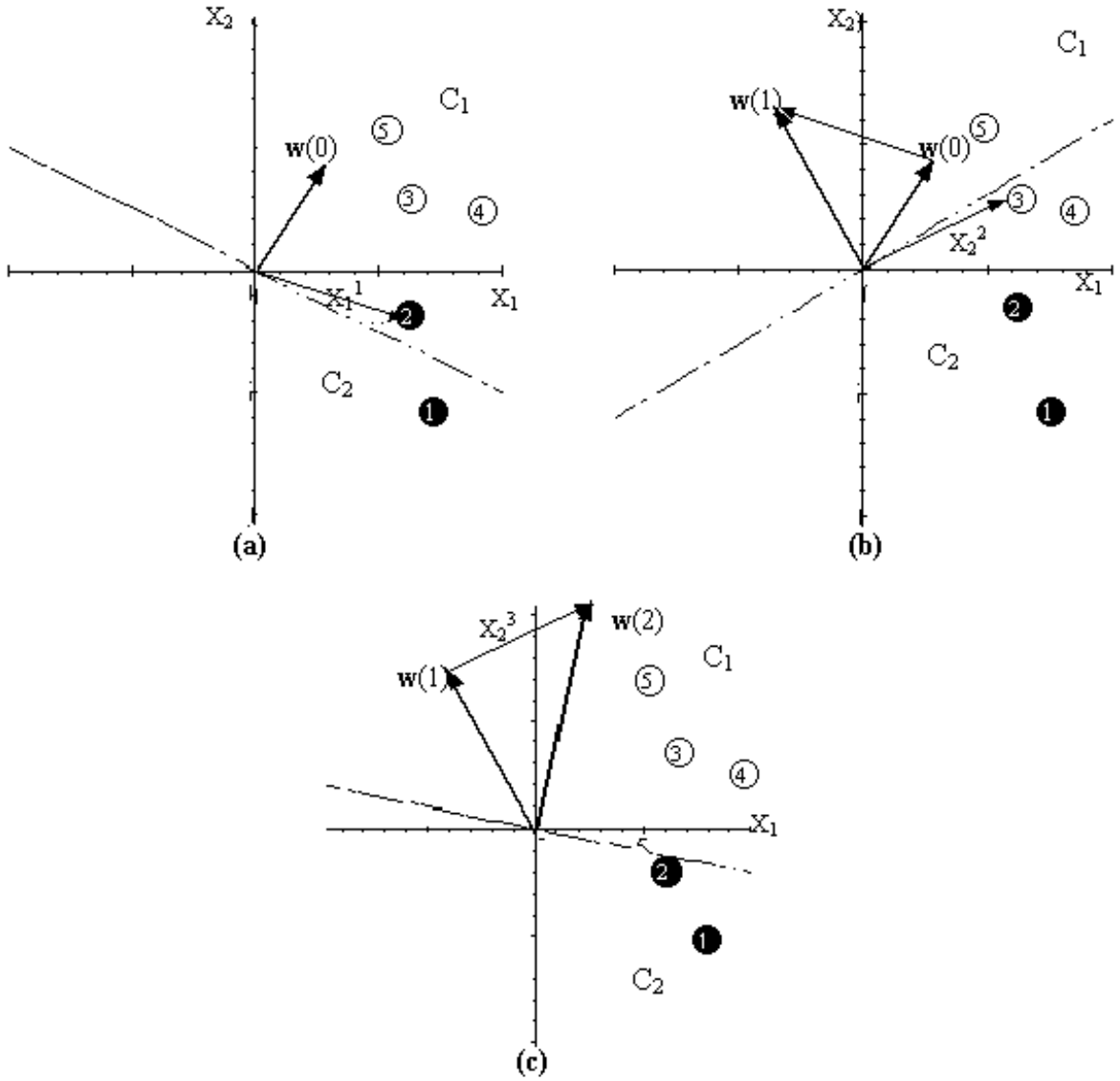
2.2.2 Teorema de convergencia del perceptron

Este teorema provee un interesante resultado: Para un conjunto de datos linealmente separable, la regla de aprendizaje establecida en 2.6, permite encontrar una solución en un número finito de pasos.

Si el conjunto es linealmente separable, esto supone que existe al menos un vector de parámetros $\mathbf{w}_0$ para el cual todos los vectores $\mathbf{x}$ están correctamente clasificados, es decir:

$$\mathbf{w}_0^{'}(\mathbf{x}_i k_i) > 0 \text{ para todo } i \tag{2.7}$$

El proceso de aprendizaje comienza con un vector $\mathbf{w}$ arbitrario, el cual puede ser el vector nulo. A cada paso del algoritmo el vector será actualizado, usando la expresión 2.6, de la



Figura~2-3: Representación gráfica del Criterio del Perceptron.

siguiente manera:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta \mathbf{x}_i k_i \tag{2.8}$$

donde $\mathbf{x}_i$, corresponde a un vector mal clasificado por el perceptron. Se simboliza con m_i, a la cantidad de veces que el vector $\mathbf{x}_i$ ha sido presentado al algoritmo y mal clasificado. En un momento cualquiera el vector $\mathbf{w}$ se calcula de la siguiente manera:

$$\mathbf{w} = \eta \sum_i m_i \mathbf{x}_i k_i \tag{2.9}$$

El objetivo es encontrar el vector $\mathbf{w}_0$, por lo que multiplicando ambos miembros de la expresión 2.9 por la traspuesta de $\mathbf{w}_0$, esta queda igual a:

$$\mathbf{w}_0' \mathbf{w} = \eta \sum_i m_i \mathbf{w}_0' \mathbf{x}_i k_i \geq \eta m \min_i \quad \mathbf{w}_0' \mathbf{x}_i k_i \tag{2.10}$$

donde $m = \sum_i m_i$ y la desigualdad resulta de reemplazar cada vector actualizado por el menor de los vectores actualizados.Comparando la expresión 2.7 y 2.10, se puede pensar que para una correcta clasificación existe un límite inferior que crece linealmente con m, pero teniendo en cuenta que $\mathbf{w}_0$ es fijo, la actualización debe cesar para algún valor finito de m. Si $\alpha = \min_i \quad \mathbf{w}_0' \mathbf{x}_i k_i$, 2.10 puede expresarse como:

$$\mathbf{w}_0' \mathbf{w} \geq \eta m \alpha \tag{2.11}$$

Haciendo uso de la desigualdad de Cauchy- Schwarz, la expresión (2.10) resulta:

$$\begin{aligned} |\mathbf{w}_0|^2 |\mathbf{w}|^2 &\geq |\mathbf{w}_0' \mathbf{w}|^2 \\ |\mathbf{w}_0|^2 |\mathbf{w}|^2 &\geq \eta^2 (m\alpha)^2 \\ |\mathbf{w}|^2 &\geq \frac{\eta^2 (m\alpha)^2}{|\mathbf{w}_0|^2} \end{aligned} \tag{2.12}$$

encontrando de esta manera un límite inferior para $|\mathbf{w}|$. Para encontrar un límite superior sobre $|\mathbf{w}|$, se tiene en cuenta el cambio en la longitud de $\mathbf{w}$ para una actualización dada por el vector

$\mathbf{x}_i$, considerando la expresión 2.8, y resolviendo el cuadrado, se tiene:

$$|\mathbf{w}(n+1)|^2 = |\mathbf{w}(n)|^2 + |\mathbf{x}_i|^2 (\eta k_i)^2 + 2\eta \mathbf{w}(n)' \mathbf{x}_i k_i \leq |\mathbf{w}(n)|^2 + |\mathbf{x}_i|^2 (\eta k_i)^2 \tag{2.13}$$

Esta desigualdad supone que el vector $\mathbf{x}_i$ ha sido mal clasificado y por lo tanto $\mathbf{w}_0'(\mathbf{x}_i k_i) < 0$. Además como $k_i = \mp 1$, entonces $k_i^2 = 1$. Siendo $|\mathbf{x}_i|^2_{\max}$, el módulo de mayor longitud, se verifica que $|\mathbf{x}_i|^2 \leq |\mathbf{x}_i|^2_{\max}$. Entonces, el cambio en el vector $\mathbf{w}$ se puede expresar como:

$$\triangle |\mathbf{w}|^2 = |\mathbf{w}(n+1)|^2 - |\mathbf{w}(n)|^2 \leq \eta |\mathbf{x}_i|^2_{\max} \tag{2.14}$$

Después de m actualizaciones del vector de parámetros, y simbolizando $\beta = |\mathbf{x}_i|^2_{\max}$,se puede expresar 2.14 como:

$$|\mathbf{w}|^2 \leq m\eta\beta \tag{2.15}$$

por lo que la longitud del vector $\mathbf{w}$ no se incrementa más que $m^{\frac{1}{2}}$,esto indica que m no puede crecer indefinidamente, y el algoritmo debe converger en un número finito de pasos.

Las ecuaciones 2.12 y 2.15 entran en conflicto para valores grandes de m, por lo que este valor debe tener un valor máximo $m_{\max}$ que surge de igualar las ecuaciones:

$$\begin{aligned} \frac{\eta^2 (m_{\max}\alpha)^2}{|\mathbf{w}_0|^2} &= m_{\max}\eta\beta \\ m_{\max} &= \frac{\beta |\mathbf{w}_0|^2}{\alpha^2 \eta} \end{aligned} \tag{2.16}$$

Estos límites no dependen del número de variables de entrada ni del número de patrones, estos últimos sólo afectan el tiempo de convergencia del algoritmo. De las expresiones 2.12, 2.15 y 2.16 se deduce que no existe una única solución para $\mathbf{w}_0$ y $m_{\max}$.

La tasa de aprendizaje puede asumir valores en el rango $0 < \eta \leq 1$. Al definir el valor que puede asumir η se debe tener en cuenta dos aspectos conflictivos: un valor pequeño provee en promedio estimaciones estables del vector de parámetros pero el tiempo de convergencia es mayor, mientras que un valor cercano a 1 permite una rápida adaptación pero el algoritmo puede volverse inestable.

En conjunto de datos que no son linealmente separables, el algoritmo nunca termina. Otro problema es que si se para el proceso de aprendizaje arbitrariamente, el vector de pesos encontrado no sirva para clasificar un nuevo conjunto de datos.

2.2.3 Funciones continuas de activación

Hasta ahora, se ha considerado una función escalón de activación. Si la función f corresponde a una función continua y derivable, la función de error resulta una función derivable respecto al vector de parámetros $\mathbf{w}$,lo que hace posible utilizar cualquier técnica de optimización para minimizar el error.

Funciones lineales

En caso que la función de activación sea la función *identidad*, considerando p variables de entrada y N ejemplos, el perceptron simple para una clasificación en dos grupos se puede definir como la siguiente función discriminante:

$$y_i = \sum_{j=0}^{p} w_j x_{i_j} \qquad i = 1, 2, ..., N$$

en forma matricial

$$y_i = \mathbf{w}'\mathbf{x}_i + \mathbf{w}_0 \qquad i = 1, 2, ..., N$$

siendo $x_0 = -1$. Cuando $y > 0$ el vector $\mathbf{x}$ es asignado a la clase 1, en caso contrario a la clase 2.

Considerando varios grupos:

$$y_i^k = \sum_{j=0}^{p} w_{kj} x_j = \mathbf{w}_i^{k\prime}\mathbf{x} + \mathbf{w}_0 \qquad i = 1, 2, ..., N$$

donde el vector $\mathbf{x}_i$ es asignado a la clase k si $y^k > y^j$, para todo $k \neq j$. Estas funciones discriminantes lineales han sido desarrolladas ampliamente en la literatura estadística (Bishop1995).

Para entrenar estas redes existen métodos que minimizan la función de costo, la cual es definida como la suma de cuadrados del error. La señal de error se define como la diferencia

entre el valor real y la respuesta del modelo.

$$e_i = k_i - y_i$$

Considerando una clasificación binaria (una red con una variable de salida), la suma del cuadrado de los errores sobre todos los ejemplos del conjunto de entrenamiento[4] está dada por:

$$E(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{N}\left[k_i - f(\mathbf{x}_i, \mathbf{w})\right]^2 \tag{2.17}$$

$$E(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{N}\left[k_i - \sum_{j=0}^{p} w_j x_{i_j}\right]^2 \tag{2.18}$$

la función de costo $E(\mathbf{w})$, es una función cuadrática. Para encontrar el vector $\mathbf{w}$ que la minimice, se debe derivar respecto a $\mathbf{w}$, y la derivada resulta una función lineal. Derivando la expresión 2.18 respecto al vector $\mathbf{w}$ e igualando la derivada a cero, resultan las siguientes ecuaciones:

$$\sum_{i=1}^{N}\left[k_i - \sum_{j=0}^{p} w_j x_{i_j}\right]^2 x_{i_j}$$

expresado en forma matricial:

$$(\mathbf{X}'\mathbf{X})\,\mathbf{W} = \mathbf{X}'\mathbf{K} \tag{2.19}$$

donde $\mathbf{X}_{(N,p+1)}$, $\mathbf{W}_{(p+1,1)}$ y $\mathbf{K}_{(N,1)}$. La matriz $\mathbf{X}'\mathbf{X}$ es una matriz cuadrada de dimensión $(p+1, p+1)$.

La solución de la ecuación 2.19, resulta:

$$\mathbf{W}' = \mathbf{X}^{-1}\mathbf{K} \tag{2.20}$$

Dado que $\mathbf{X}$ no es una matriz cuadrada, $\mathbf{X}^{-1}$ es la matriz conocida como la *pseudo-inversa* de $\mathbf{X}$, que se calcula como:

$$\mathbf{X}^{-1} = (\mathbf{X}'\mathbf{X})^{-1}\,\mathbf{X}' \tag{2.21}$$

La pseudo inversa cumple con la propiedad que $\mathbf{X}^{-1}\mathbf{X} = \mathbf{I}$, siendo $\mathbf{I}$ la matriz identidad,

[4] En caso de más de dos grupos, se deben sumar los errores sobre todos los grupos.

sin embargo $\mathbf{X}\mathbf{X}^{-1} \neq \mathbf{I}$. Si la matriz $\mathbf{X}'\mathbf{X}$ es no regular la expresión 2.21 no tienen una única solución. Sin embargo, si la pseudo inversa se define como:

$$\mathbf{X}^{-1} = \lim_{\epsilon \to 0} \left(\mathbf{X}'\mathbf{X} + \epsilon \mathbf{I}\right)^{-1} \mathbf{X}' \tag{2.22}$$

puede demostrarse que este límite existe y su valor minimiza E.

En la práctica, la solución de las ecuaciones 2.20 tienen dificultades numéricas debido a la posibilidad de $\mathbf{X}'\mathbf{X}$ sea singular o casi singular. Esto ocurre si las variables de entrada son colineales entre sí; en este caso los parámetros asumen valores grandes, en caso contrario, si son ortogonales, asumen valores pequeños. El vector $\mathbf{w}$ es el que permite encontrar la proyección ortogonal del vector $\mathbf{k}$, que pertenece al espacio N-dimensional sobre el espacio de variables.

Hasta aquí el umbral de activación w_0 ha sido considerado dentro del vector $\mathbf{w}$. Si se considera el mínimo de la expresión 2.18 respecto del vector w_0, resulta:

$$\frac{\partial E(\mathbf{w})}{\partial w_0} = \sum_{i=1}^{N} \left[\sum_{j=1}^{p} w_j x_{i_j} + w_0 - k_i \right]^2 \tag{2.23}$$

igualando a cero y despejando w_0

$$w_0 = \overline{k} - \sum_{j=1}^{p} w_j \overline{x}_j \tag{2.24}$$

donde

$$\overline{k} = \frac{1}{N} \sum_{i=1}^{N} k_i \qquad\qquad \overline{x}_j = \frac{1}{N} \sum_{i=1}^{N} x_{i_j} \tag{2.25}$$

De esta manera, el rol del umbral de activación es compensar, en el conjunto de entrenamiento, la diferencia entre la media del vector de salida y la media de los valores reales.

Si $\mathbf{X}$ es una matriz regular, la pseudoinversa se reduce a la inversa de la matriz, y en este caso el número de variables debe ser igual al número de ejemplos del conjunto de entrenamiento; además las variables (que corresponden a la columna de la matriz $\mathbf{X}$), deben ser linealmente independientes.[5] Pero es más deseable tener conjuntos de entrenamiento suficientemente grandes, cumpliendo $N > p$, para lograr que existan ejemplos suficientes en el espacio de variables que

[5] El concepto de independencia lineal de las variables, es diferente al de separabilidad lineal. Por lo expuesto hasta ahora la independencia lineal implica separabilidad lineal, pero la implicación contraria no es cierta.

permitan una buena representación del mismo. Al incrementar el número de variables surgen puntos donde los datos son escasos y no se tiene una buena representación del espacio.

Un percepton con función de activación lineal tiene una estrecha relación con el discriminante lineal desarrollado ampliamente en la literatura estadística, ambos son clasificadores lineales. El procedimiento de estimación clásico para el discriminante lineal es el método de máxima verosimilitud, y se parte del supuesto que el vector $\mathbf{x}_i$ de características de los i ejemplos, $i = 1, 2, 3, ...N$ proviene de una población normalmente distribuída, y se define en términos del vector de medias $\boldsymbol{\mu}$ y la matriz de covarianzas $\mathbf{C}$, los cuales se calculan como:

$$\boldsymbol{\mu} = \mathbf{E}(\mathbf{x}) \qquad \mathbf{C} = \mathbf{E}(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})' \tag{2.26}$$

siendo $\mathbf{E}$ el operador de la esperanza matemática.

Bajo el supuesto que el vector $\mathbf{x}$ está normalmente distribuído, y limitando el problema de clasificación a dos clases C_1 y C_2 con la misma matriz $\mathbf{C}$ de covarianzas. Cada clase tendrá su propio vector de medias $\boldsymbol{\mu}_1$ y $\boldsymbol{\mu}_2$ respectivamente,el valor del vector de medias de $\mathbf{x}$, depende de la clase de pertenencia y la matriz de covarianzas es la misma en las dos clases. Además se asume que:

-Un vector $\mathbf{x}$,tiene la misma probabilidad de pertenecer a la clase 1 ó 2.

-Los ejemplos de la clase 1 y 2 están correlacionados por lo que la matriz $\mathbf{C}$, no es diagonal.

-La matriz $\mathbf{C}$, es regular por lo que existe $\mathbf{C}^{-1}$.

Dada la función de densidad conjunta de las variables iniciales como $f(\mathbf{x}/C_i), i = 1, 2$, y definida la función de verosimilitud como $l_i(\mathbf{x}) = \ln f(\mathbf{x}/C_i)$, resulta igual a:

$$l_i(\mathbf{x}) = \mathbf{u}_i' \mathbf{C}^{-1}\mathbf{x} - \frac{1}{2}\mathbf{u}_i' \mathbf{C}^{-1}\mathbf{u}_i \qquad \text{para } i = 1, 2 \tag{2.27}$$

Restando el logaritmo de la verosimilitud entre las dos clases, se obtiene:

$$\begin{aligned} l &= l_1(\mathbf{x}) - l_2(\mathbf{x}) \\ &= (\mathbf{u}_1 - \mathbf{u}_2)' \mathbf{C}^{-1}\mathbf{x} - \frac{1}{2}\left(\mathbf{u}_1' \mathbf{C}^{-1}\mathbf{u}_1 - \mathbf{u}_2' \mathbf{C}^{-1}\mathbf{u}_2\right) \end{aligned} \tag{2.28}$$

cuya expresión está relacionada directamente con $\mathbf{x}$. La expresión xx se puede reescribir como:

$$\begin{aligned} l &= \widehat{\mathbf{w}}'\mathbf{x} - \widehat{w}_0 \\ &= \sum_{i=1}^{p} \widehat{w}_i' x_i - \widehat{w}_0 \end{aligned} \tag{2.29}$$

donde $\widehat{\mathbf{w}}$ es el estimador máximo verosímil del vector de parámetros, definido como:

$$\widehat{\mathbf{w}} = \mathbf{C}^{-1}\left(\mathbf{u}_1 - \mathbf{u}_2\right)$$

y $\widehat{w}_0$ es un umbral constante definido por:

$$\widehat{w}_0 = \frac{1}{2}\left(\mathbf{u}_1'\mathbf{C}^{-1}\mathbf{u}_1 - \mathbf{u}_2'\mathbf{C}^{-1}\mathbf{u}_2\right) \tag{2.30}$$

El discriminante lineal caracterizado por el vector de parámetros $\widehat{\mathbf{w}}$ y el umbral $\widehat{w}_0$, resuelve un problema de clasificación de acuerdo a la siguiente regla:

Si $l \geq 0$, entonces $l_1 \geq l_2$, y el vector $\mathbf{x}$ es asignado a la clase C_1

Si $l < 0$, entonces $l_1 < l_2$, y el vector $\mathbf{x}$ es asignado a la clase C_2

El discriminante lineal es similar a un perceptron simple en cuanto ambos son clasificadores lineales, sin embargo existen algunas diferencias a destacar entre ellos.

a)El perceptron simple asume como supuesto que los patrones a ser clasificados son linealmente separables. El discriminante lineal tiene como supuesto la distribución normal de las clases, las cuales pueden estas solapadas entre sí y por lo tanto no serán exactamente separables, la magnitud del solapamiento está dado por los vectores de medias $\mathbf{u}_1$ y $\mathbf{u}_2$, y las matrices de covarianzas $\mathbf{C}_1$ y $\mathbf{C}_2$. Cuando existe solapamiento el teorema de convergencia del perceptron puede oscilar continuamente en los límites de decisión entre diferentes clases.

b)El estimador máximo verosímil minimiza la probabilidad promedio del error de clasificación, independientemente del solapamiento entre las distribuciones normales subyacentes de las clases. El límite de decisión siempre se posiciona donde se superponen las distribuciones, esta posición concuerda con el supuesto que las clases tienen la misma probabilidad a priori ya que no existe información previa que permita realizar una decisión.

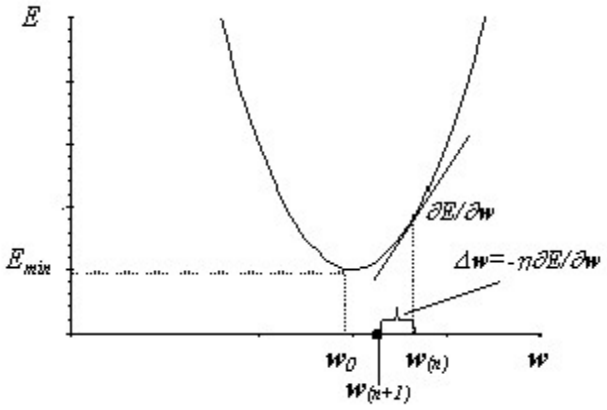
c) El algoritmo de convergencia del perceptron simple no realiza ningún supuesto respecto

a la distribución de probabilidad subyacente de los grupos, lo que lo hace más robusto que las técnicas clásicas y se obtienen buenos resultados cuando las distribuciones no son normales. En cambio el supuesto de normalidad del discriminante lineal limita mucho el área de aplicación del mismo.

Otra forma de encontrar el vector **w** que minimice la función de costo E**,** cuando la función de activación es continua y derivable, es utilizando una variedad de algoritmos de optimización basados en la derivada, entre los que se encuentra *el descenso por el gradiente*, evitando de esta manera, el uso de la matriz de inversión.

Método descenso por el gradiente.

Por el método del descenso por el gradiente el vector de parámetros va variando, ajustando sus valores en forma iterativa a lo largo de la superficie de la función de costo en el espacio de **w**, moviéndose progresivamente a la solución óptima, que es el valor mínimo de la función. Para ello, se elige aleatoriamente un vector inicial de parámetros y se va actualizando sus valores moviéndose en el espacio del vector a pequeñas distancias hacia la dirección en la que el valor de la función de costo decrece más rapidamente, esto es en dirección opuesta a la derivada respecto del vector **w**, tal como se ilustra en la figura 2.5. para un solo vector.



Figura˜2-4: Representación gráfica del Método de descenso por el gradiente.

El ajuste para actualizar el valor de un parámetro, $\triangle w_j(n)$, está dada por una cantidad

proporcional al gradiente de la función E en su actual ubicación.

$$\triangle w_j(n) = -\eta \frac{\partial E(n)}{w_j(n)} \tag{2.31}$$

$$\triangle w_j(n) = \eta \sum_{i=1}^{N} (k_i - y_i) x_i \tag{2.32}$$

Si estos cambios son realizados individualmente para cada ejemplo, la expresión 2.32 es igual a:

$$\triangle w_j(n) = \eta (k_i - y_i) x_i \tag{2.33}$$

$$= \eta \delta_i x_i \qquad \text{donde } \delta_i = k_i - y_i \tag{2.34}$$

La actualización repetida a través de los ejemplos, se utiliza en aplicaciones reales donde los ejemplos van apareciendo continuamente. En este caso, cada dato puede ser utilizado y luego descartado, mientras el sistema debe permitir que se rastreen los cambios en las características de los datos.

La regla definida y sus variantes son conocidas por diferentes nombres tales como regla LMS (least mean squares), regla adaline, regla Widrow- Hoff o regla Delta.

La expresión

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \eta \left[\mathbf{x}(n) \mathbf{w}(n) - k \right] \mathbf{x}(n) \tag{2.35}$$

describe la evolución de los parámetros en el tiempo y puede reescribirse de la siguiente forma:

$$\mathbf{w}(n+1) = \left[\mathbf{I} - \eta \mathbf{x}(n) \mathbf{x}'(n) \right] \mathbf{w}(n) + \eta \mathbf{x}(n) k \tag{2.36}$$

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

El valor de $\mathbf{w}(n)$ puede expresarse como:

$$\mathbf{w}(n) = z^{-1} \mathbf{w}(n+1) \tag{2.37}$$

donde z^{-1} es un operador de retardo. De esta manera, el algoritmo puede verse como un sistema de realimentación estocástico. Los valores que asumen el parámetro de aprendizaje y el vector

de entradas, determinan la transmisión en el proceso de retroalimentación; como consecuencia estos valores influyen en la convergencia y estabilidad del algoritmo.

El algoritmo es convergente en la media si el valor del vector $\mathbf{w}(n)$ se aproxima a $\widehat{\mathbf{w}}$ en n iteraciones

$$\mathbf{w}(n) \rightarrow \widehat{\mathbf{w}} \qquad \text{para } n \rightarrow \infty \tag{2.38}$$

Mientras que es convergente en la media cuadrática si el cuadrado del error se aproxima a un valor constante en n iteraciones, para n que tiende a infinito.

$$\mathbf{e}^2(n) \rightarrow c \qquad \text{para } n \rightarrow \infty \tag{2.39}$$

La convergencia en la media cuadrática es más amplia que la convergencia en la media, por lo que si el algoritmo converge en la media cuadrática, también es convergente en la media, pero lo contrario no es necesariamente cierto.

Para demostrar la condición de convergencia en la media, partiendo de la expresión 2.36 se puede buscar una transformación ortogonal de $\mathbf{XX'}$ como sigue:

$$\begin{aligned} \mathbf{T'}\left(\mathbf{XX'}\right)\mathbf{T} &= \mathbf{A} \\ \mathbf{XX'} &= \mathbf{T'AT} \end{aligned}$$

Siendo $\mathbf{A}$,una matriz diagonal cuyos elementos de la diagonal principal son los valores propios de $\mathbf{XX'}$ y $\mathbf{T}$ es una matriz ortogonal cuyas columnas están asociadas con los vectores propios de $\mathbf{XX'}$. Si $\mathbf{T}$ es una matriz ortogonal se cumple que $\mathbf{T}^{-1}= \mathbf{T'}$ y $\mathbf{TT'} = \mathbf{I}$.

Si $\mathbf{W}^*$ es la solución de la ecuación definida en el punto 2.19, el vector que minimiza la función de costos, es el vector de pesos en el óptimo.

$$\left(\mathbf{X'X}\right)\mathbf{w}^*= \mathbf{X'K} \tag{2.40}$$

Utilizando la transformación ortogonal y la igualdad $\mathbf{T}^{-1}= \mathbf{T'}$, la expresión 2.36 puede transformarse en:

$$\mathbf{T'w}(n+1) = \left[\mathbf{T'}-\eta\mathbf{T'TAT'}\right]\mathbf{w}(n) + \eta\mathbf{T'TAT'w}^* \tag{2.41}$$

$$= \quad [\mathbf{I}-\eta\mathbf{A}]\,\mathbf{T}'\mathbf{w}(n) + \eta\mathbf{A}\mathbf{T}'\mathbf{w}^* \tag{2.42}$$

Se define $\mathbf{V}(n)$ como una transformación del desvío entre el vector de parámetros encontrado al momento n y el óptimo $\mathbf{w}^*$, de la siguiente manera:

$$\begin{aligned} \mathbf{V}(n) &= \mathbf{T}'\,[\mathbf{w}(n) - \mathbf{w}^*] \qquad \text{siendo} \\ \mathbf{V}(n+1) &= \mathbf{T}'\,[\mathbf{w}(n+1) - \mathbf{w}^*] \end{aligned}$$

de donde surge que:

$$\mathbf{w}(n) = \mathbf{V}(n)\mathbf{T} + \mathbf{w}^*$$

reemplazando en la expresión 2.42, $\mathbf{V}(n+1)$ se puede expresar como:

$$\mathbf{V}(n+1) = [\mathbf{I}-\eta\mathbf{A}]\,\mathbf{V}(n) \tag{2.43}$$

El vector $\mathbf{V}(n+1)$ representa un sistema de ecuaciones homogéneas en diferencia de primer orden, donde cada elemento del mismo puede escribirse:

$$V_k(n+1) = [1-\eta\lambda_k]\,V_k(n) \qquad k = 1, 2, ...p. \tag{2.44}$$

donde los λ_k son los valores propios de la matriz $\mathbf{X}'\mathbf{X}$ y $V_k(n)$ es el k-ésimo elemento de vector $\mathbf{V}(n)$. Dado $V_k(0)$ un valor inicial arbitrario se puede expresar 2.44 como:

$$V_k(n+1) = [1-\eta\lambda_k]^n\,V_k(0) \qquad k = 1, 2, ...p. \tag{2.45}$$

Para que el algoritmo converja en $\sum_n \mathbf{w}(n)$, para un valor arbitrario de $V_k(0)$, se debe satisfacer la siguiente condición:

$$|1-\eta\lambda_k| < 0 \qquad k = 1, 2, ...p. \tag{2.46}$$

de donde se cumple que $V_k(n) \to 0$, a medida que $n \to \infty$.

El valor de η en 2.44, está limitado por el mayor valor propio λ_k^{max} de la matriz $\mathbf{X}'\mathbf{X}$, que corresponde al valor más empinado de la curvatura de la superficie del error. Es necesario

considerar como límite[6] $\eta < \frac{1}{\lambda_j^{max}}$, o puede ocurrir que los cambios sean de a saltos, que pasen de un lado al otro del mínimo, produciendo grandes oscilaciones en su trayectoria. La razón de proximidad en el óptimo está limitada por el menor valor propio distinto de cero $\lambda_j^{\min}$ corresponde a la dirección de menor curvatura. Si $\frac{\lambda_j^{max}}{\lambda_j^{\min}}$ es grande el progreso del algoritmo a lo largo de las curvaturas, con menos pendiente puede ser demasiado lento.

El análisis de convergencia en el error cuadrático es más complicado y demanda más supuestos sobre el comportamiento del vector $\mathbf{w}$ (Haykin, 1991); el cual establece que para asegurar la convergencia del algoritmo la tasa de aprendizaje debe satisfacer la siguiente condición

$$0 < \eta < \frac{1}{\sum_{k=1}^{p} \lambda_k}$$

siendo por álgebra matricial $\sum_{k=1}^{p} \lambda_k$ la traza de la matriz $\mathbf{XX'}$. De esta manera se cumple que

$$\sum_{k=1}^{p} \lambda_k \geq \lambda^{max} \tag{2.47}$$

si se satisface la condición 2.47, también se satisface la desigualdad $\eta < \frac{1}{\lambda_j^{max}}$.

Funciones no lineales

Todo lo analizado en el punto anterior puede generalizarse cuando las funciones de activación no son lineales, considerando generalmente funciones monótonas, siendo las más utilizadas las funciones logística y tangente hiperbólica.

Una función logística tiene la siguiente expresión:

$$f(v) = \frac{1}{1 + \exp(-av)}$$

siendo a un parámetro de pendiente. Cuando a tiende a infinito, $f(v)$ es una función escalón y si a tiende a cero, $f(v)$ tiende a una función lineal (ver figura 2.5). Los valores de la función varían entre 0 y 1

[6] Si se consideran todas las muestras posibles, y todas las expresiones se toman en términos esperados, entonces η está medida en unidades equivalentes a la inversa de la varianza.

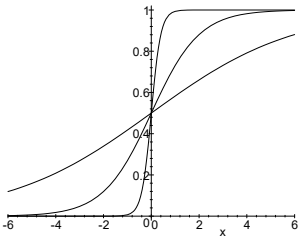


Figura 2.5: Función logística

La tangente hiperbólica tiene la siguiente expresión:

$$f(v) = \frac{1 - \exp(-ax)}{1 + \exp(-ax)}$$

es similar a la función logística sólo que varía entre -1 y1.

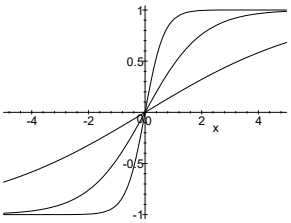


Figura 2.6:Función Tangente hiperbolica

Ambas funciones son derivables, por lo que es directa la generalización del algoritmo del descenso por le gradiente. Teniendo en cuenta que la función discriminante que definimos es:

$$y = f(\sum_{j=1}^{p} w_j x_j - w_0)$$

Igualando $v = \sum_{j=1}^{p} w_j x_j - w_0$, la función de costo se define como:

$$E(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^{N} [k_i - f\,(v)]^2$$

Siendo f, una función continua y derivable es posible encontrar la derivada de $E(\mathbf{w})$ respecto

$\mathbf{w}$, como sigue:

$$\frac{\partial E(\mathbf{w})}{\partial w_j} = -\sum_{i=1}^{N} \left[k_i - f\left(v_i\right)\right] f'\left(v\right) x_{i_j}$$

por lo que la correción de los parámetros en cada ejemplo está dada por la expresión:

$$\triangle w_j(n) = \eta \delta_i x_i \tag{2.48}$$

$$\triangle w_j(n) = \eta \left(k_i - y_i\right) f'\left(v_i\right) x_i \tag{2.49}$$

siendo $f'\left(v\right) = f\left(v\right)\left(1 - f\left(v\right)\right)$ para la función logística, y $f'\left(v\right) = 1 - \left[f\left(v\right)\right]^2$ para la tangente hiperbólica.

Las condiciones para la existencia de una solución, igual que para el caso lineal es la independencia de las variables. Esta limitación puede superarse incorporando variables intermedias, donde pueden utilizarse funciones de activación no lineales.

Capítulo 3

Modelo logístico

3.1 Presentación del modelo logístico binario

En la unidad anterior se definió un perceptron simple con una función de activación logística cuya expresión es la siguiente:

$$y = f(\sum_{j=1}^{p} w_j x_j - w_0) \tag{3.1}$$

donde

$$f(v) = \frac{1}{1+\exp(-v)} \text{ siendo } v = \sum_{j=1}^{p} w_j x_j - w_0 \tag{3.2}$$

reemplazando la expresión 3.2 en la expresión 3.1, resulta

$$y = \frac{1}{1+\exp -(\sum_{j=1}^{p} w_j x_j - w_0)} \tag{3.3}$$

Esta última expresión, se conoce en la literatura estadística como **dicriminante logístico**, la cual permite estimar las probabilidades a posteriori (la probabilidad de pertenecer a la clase k dado un vector $\mathbf{x}$ de predictores $P(k/\mathbf{x})$), sin suponer una densidad determinada para el vector de predictores $f(\mathbf{x})$ manteniendo el supuesto de la separabilidad lineal.

Para el caso de clasificación en dos grupos, las probabilidades a posteriori serán:

$$P(0/\mathbf{x}) = y(\mathbf{x}) = \frac{1}{1+\exp -(\sum_{j=1}^{p} w_j x_j - w_0)} \qquad y \qquad P(1/\mathbf{x}) = 1 - y(\mathbf{x}) \backslash = \frac{\exp -(\sum_{j=1}^{p} w_j x_j - w_0)}{1+\exp -(\sum_{j=1}^{p} w_j x_j - w_0)} \tag{3.4}$$

El logaritmo del cociente de probabilidades $P(0/\mathbf{x})$ y $P(1/\mathbf{x})$, es una función lineal:

$$Z(\mathbf{x}) = \ln\left(\frac{P(1/\mathbf{x})}{P(0/\mathbf{x})}\right) = \sum_{j=1}^{p} w_j x_j - w_0 \tag{3.5}$$

Como es conocido, el método de máxima verosimilitud permite obtener estimadores de los parámetros que maximizan la probabilidad conjunta de valores muestrales observados. En lo que sigue, se toma en gran parte el razonamiento expuesto en Hosmer (1989), donde se interpreta que para aquellos pares $(\mathbf{x}_i, k_i)$ donde $k_i = 1$, la contribución a la función de verosimilitud es $P(1/\mathbf{x}_i)$, en tanto que para aquellos pares en los que $k_i = 0$, la contribución a la función de verosimilitud es $1 - P(1/\mathbf{x}_i)$. Considerando que $P(k/\mathbf{x}_i)$ debe ser evaluada para cada individuo, la contribución a la función de verosimilitud de cada uno es expresado a través de:

$$\begin{aligned} P(k_i) &= P(k/\mathbf{x}_i)^{k_i}(1 - P(k/\mathbf{x}_i)^{1-k_i} \\ \textit{donde } k_i &= 0, 1 \textit{ es la verdadera clase de pertenencia} \end{aligned}$$

Asumiendo independencia entre las observaciones, la función de verosimilitud $l(\boldsymbol{\beta})$es el producto de $P(k_i)$ para todas las observaciones:

$$l(\boldsymbol{\beta}) = \prod_{i=1}^{N} P(k_i) = \prod_{i=1}^{N} P(k/\mathbf{x}_i)^{k_i}(1 - P(k/\mathbf{x}_i)^{1-k_i} \tag{3.6}$$

$$\ln\ l(\boldsymbol{\beta}) = L(\boldsymbol{\beta}) = \sum_{i=1}^{n} k_i \ln P(k/\mathbf{x}_i) + (1 - k_i)\ln(1 - P(k/\mathbf{x}_i)) \tag{3.7}$$

$$L(\boldsymbol{\beta}) = \sum_{i=1}^{N} k_i \ln \frac{e^{Z(\mathbf{x})}}{1+e^{Z(\mathbf{x})}} + (1 - k_i)\ln \frac{1}{1+e^{Z(\mathbf{x})}} \textit{ donde } Z(\mathbf{x}) = \sum_{j=1}^{p} w_j x_j - w_0 \tag{3.8}$$

Se debe buscar el máximo de la expresión (3.8) con respecto al vector de parámetros desconocido $\mathbf{W}'=(w_0, w_1, w_2, ..., w_p)$ de orden $p+1$, obteniendo un vector de estimadores $\widehat{\mathbf{W}}$ denominado estimador máximo verosímil. Por la condición de primer orden, la derivada de L con respecto a $\mathbf{W}$ debe ser igual al vector nulo:

$$\frac{\partial l}{\partial \mathbf{W}} = \mathbf{0} \ ,$$

arribando a las $p+1$ ecuaciones de verosimilitud:

$$\begin{aligned}
\sum_{i=1}^{N} \left[k_i - \frac{e^{\sum_{j=1}^{p} w_j x_j - w_0}}{1 + e^{\sum_{j=1}^{p} w_j x_j - w_0}} \right] &= 0 \\
\sum_{i=1}^{N} x_{i1} \left[k_i - \frac{e^{\sum_{j=1}^{p} w_j x_j - w_0}}{1 + e^{\sum_{j=1}^{p} w_j x_j - w_0}} \right] &= 0 \\
&\vdots \\
\sum_{i=1}^{N} x_{ip} \left[k_i - \frac{e^{\sum_{j=1}^{p} w_j x_j - w_0}}{1 + e^{\sum_{j=1}^{p} w_j x_j - w_0}} \right] &= 0
\end{aligned} \tag{3.9}$$

Como se advierte, se obtienen expresiones no lineales en $w_0, w_1, w_2, ..., w_p$, por lo que la estimación máximo verosímil de los coeficientes no es sencilla, ya que debe recurrirse a métodos recursivos, tales como el algoritmo de Newton-Raphson[1].

La ventaja de este enfoque es que se puede estimar *p+1* parámetros, sin tener una específica forma funcional de la densidad. Sin embargo, cuando se conoce la forma funcional, su empleo deriva en una pérdida de información. Cuando se aplica al caso normal, Hand (1999 Cap. 3) señala que "el modelo logístico estima directamente la (transformación logística) de la probabilidad condicional, mientras que el enfoque discriminante lineal deriva estas probabilidades indirectamente vía estimación de las distribuciones normales de cada clase" (Traducción del autor).

[1] Un detalle de estos algoritmos recursivos puede verse en McCullagh y Nelder(1983)

3.1.1 Pruebas de significación de los coeficientes

Después de la estimación de los coeficientes, se debe evaluar la significatividad de las variables en el modelo, es decir si la inclusión de la misma mejora o no el modelo propuesto. La diferencia entre los valores observados con los estimados, en los modelos con y sin la variable, permite determinar una medida para testear la significación de la misma. Para el discriminante logístico, esta comparación está basada en la función de verosimilitud definida en 3.6, y se denomina razón de verosimilitud o Deviance

$$\begin{aligned} D &= -2\ln\left[\frac{verosimilitud\ del\ modelo\ corriente}{verosimilitud\ del\ modelo\ saturado}\right] \\ D &= -2\sum_{i=1}^{N}\left\{k_i \ln\left(\frac{P(k/\mathbf{x}_i)}{k_i}\right) + (1-k_i)\ln\left(\frac{1-P(k/\mathbf{x}_i)}{1-k_i}\right)\right\} \end{aligned} \tag{3.10}$$

En el modelo corriente las probabilidades condicionales son estimadas a partir de los coeficientes, en tanto que en el saturado, los valores observados de la variables respuesta constituyen los valores estimados.

Para asegurar la significación de una variable independiente en el modelo se compara el valor de D que resulta excluyendo dicha variable en el modelo, con el valor de D que resulta con la variable incluida en el modelo, arribando a un estadístico designado con la letra G:

$$G = D(sin\ la\ variable) - D(con\ la\ variable) \tag{3.11}$$

Bajo la hipótesis que $w_j = 0$, para $j = 0, 1, 2, ..., p$, el estadístico resultante sigue una distribución $\chi^2(1)$ para una variable continua y $\chi^2(c-1)$ para una variable categórica con c niveles.

Otro estadístico para probar que $w_j = 0$ es el test de Wald, que se calcula como cociente entre el parámetro estimado y su desvió estándar estimado $Wd = \frac{\widehat{w}_j}{\sigma(\widehat{w}_j)}$ el que, bajo hipótesis nula, sigue una distribución normal. Numerosos estudios han evaluado la performance de este estadístico en regresión logística, concluyendo que cuando es alto el valor de su desvío estándar, esto conduce a rechazar la hipótesis nula aún cuando los coeficientes son significativos, por lo que recomiendan usar el test de razón de verosimilitud (ver Hosmer y Lemeshow, Cap. 1,1989). Por otra parte, el estadístico de Wald es la única prueba confiable para cada nivel de una variable categórica, ya que el estadístico G se calcula para todas las categorías de la variable.

Si se necesita probar que un subconjunto de coeficientes son nulos, se puede trabajar con el estadístico de razón de verosimilitud designado con la letra G, que seguirá siendo Chi cuadrado, con tantos grados de libertad como coeficientes se anulan. Específicamente, si se desea probar el modelo completo, se establece la hipótesis $\mathbf{W} = \mathbf{0}$ y G tiene p grados de libertad.

3.2 Análisis previo de las variables a ingresar al modelo logístico

Para ayudar a la selección de variables en un modelo logístico se pueden seguir los siguientes pasos:

1. Un cuidadoso análisis univariado. Para las **variables categóricas** (nominales, ordinales o continuas con pocos valores) se verifica el nivel de asociación con la variable respuesta a través de una prueba Chi Cuadrado. El estadístico Chi Cuadrado de Pearson es estadísticamente equivalente al test Chi cuadrado de razón de verosimilitud para la significación de los coeficientes. Para las **variables continuas** se debe verificar la diferencia de media entre los grupos considerados para lo cual el test T de diferencias de medias es equivalente ajustar un modelo de regresión logística con esa única variable
2. Para las variables retenidas en el punto anterior, realizar regresiones logísticas univariadas. En el caso de variables numéricas, se deben agrupar en intervalos (utilizando los cuartiles o deciles), y analizar gráficamente los coeficientes estimados que surgen de realizar una regresión logística univariada, verificando que los mismos cumplan el supuesto de linealidad en el logit. Si el supuesto no se cumple puede ser conveniente categorizar adecuadamente la variable. Para las variables categóricas, una regresión logística univariada permite determinar si todas las categorías son significativas, o si es necesario reagruparlas.
3. Todas las variables cuyo test univariado dé un nivel de significación menor[2] a 0,25 pueden ser predictoras, y se incluyen en el modelo multivariado, además de aquellas consideradas importantes científicamente. Puede ocurrir que en el análisis univa-

[2]Una justificación de este criterio puede verse en Hosmer y Lesmeshow Capítulo 4 punto 4.2

riado una variable esté debilmente asociada a la respuesta, pero ser un importante predictor cuando se incluye en análisis multivariado. Cuando del análisis univariado quedan un número grande de variables, se utiliza el método de selección paso a paso, desarrollado más adelante.En el procedimiento, se deben utilizar puntos de corte no muy restricitivos, que permitan la selección de un modelo bastante completo

4. Habiendo realizado todos los pasos anteriores, debe verificarse la importancia de cada variable, examinando la estadística de Wald y comparando cada coeficiente estimado con el coeficiente del modelo univariado conteniendo sólo esa variable. Realizando esta comparación ,las variables que que

   no contribuyen al modelo deben ser eliminadas y se debe ajustar un nuevo modelo. El test de razón de verosimilitud permite verificar si el nuevo modelo mejora, y deben compararse los coeficientes de las variables que quedaron con los coeficientes del modelo anterior. Si los coeficientes de algunas variables han cambiado marcadamente en magnitud, indica que una o más de las variables excluídas fueron importantes en el sentido de proveer un efecto necesario sobre las variables que quedan en el modelo. Este proceso termina cuando quedan en el modelo aquellas variables que son estadísticamente o científicamente importantes.

5. Una vez que hemos obtenido un modelo que contiene todas las variables esenciales, debemos analizarlas más de cerca y considerar la necesidad de incluir términos de interacción.

Procedimiento paso a paso

El procedimiento de selección o eliminación de variables del modelo está basado en un algoritmo estadístico que establece una regla de decisión, la cual chequea la importancia de las variables y determina si las incluye o no en el modelo. En cada paso, se incorpora la variable más importante en términos estadísticos, es decir, aquella que produce el mejor cambio en el valor de verosimilitud del modelo comparada con el modelo que no la contiene y se establece un criterio de remoción de las variables que se han incorporado hasta ese paso. Deben establecerse previamente dos valores, un nivel hasta el cual se juzga importante el ingreso de una variable P_E ,denominado corte de entrada. La elección del mismo es importante ya que un punto de

corte muy bajo, 0,05 por ejemplo puede ser muy restrictivo, generalmente se recomienda valores entre 0,15 y 0,25. Y para analizar la remoción de las mismas, se determina un valor de corte para remover P_R, el cual debe exceder el valor de P_E.

El procedimiento, considerando un conjunto de p variables, se puede resumir de la siguiente manera:

Paso 0: Se ajusta el modelo con la constante y se calcula la logverosimilitud del mismo L^0 (el supraíndice indica el paso considerado), a continuación se ajustan cada uno de los posibles modelos logísticos univariados comparando la logverosimilitud de cada uno L_j^0 (el subíndice indica la variable y) con el modelo de la constante calculando el estadístico G

$$G_j^o = 2(L_j^0 - L^0)$$

Se busca la probabilidad a la derecha del valor G_j^o , $P_j^0 = \Pr(\chi^2_{(g)} > G_j^o)$. A partir de estos valores se elige la variable que presente el menor nivel de significación $P^0_{x_{e_1}} = \min P_j^0$. El proceso de selección de variables continúa mientras $P^0_{x_{e_1}} < P_E$.

- **Paso1**: Seleccionada la variable que ingresa x_{e_1}, se ajusta el modelo que la contiene, y se calcula la nueva verosimilitud $L^1_{e_1}$. Para determinar la importancia de las p-1 variables que quedan, se ajustan p-1 modelos logísticos con cada una (x_j para $j = 1, 2, 3....., p$ y $j \neq e_1$) y la variable seleccionada x_{e_1}, calculando la verosimilitud para cada uno $L^1_{e_j}$, y su diferencia con el modelo seleccionado hasta ahora.

$$G_j^1 = 2(L^1_{e_j} - L^1_{e_1})$$

determinando P_j^1 (la probabilidad a la derecha de G_j^1), si estos valores son menores a P_E,se continua con el paso 2, en caso contrario finaliza el proceso de selección.

- **Paso 2**: Partiendo del modelo con las variables seleccionadas x_{e_1} y x_{e2} se analiza la remoción de las mismas teniendo en cuenta el valor P_R establecido. Para decidir la variable a remover, se calcula el estadístico G,a partir de las diferencias entre las verosimilitudes del modelo que contiene las variables x_{e_1} y x_{e2} y los modelos que resultan de eliminar

cada una de ellas:

$$G^1_{x_j} = 2(L^2_{e_1,e_2} - L^2_{e_j})$$

dado el estadístico se busca la probabilidad a la derecha $P^2_{-e_j}$, y se determina el valor máximo P^2_r =max$(P^2_{-x_{e_1}}, P^2_{-x_{e2}})$, siendo $r = x_{e_1}, x_{e2}$. Si se cumple que: $P^2_r > P_R$, entonces la variable x_r es removida del modelo.

Luego, siguiendo el criterio anterior, se continúa con la selección de las p-2 variables que no están seleccionadas, ajustando los modelos que contienen las variables x_{e_1}, x_{e_2} y x_j, para $j = 1, 2, 3.....,p$ y $j \neq e_1$ y e_2. Se evalúan la verosimilitud de cada modelo, computando los test de razón de verosimilitud con el modelo que contiene las variables x_{e_1} y x_{e_2},y se selecciona la variable con menor nivel de significación, respetando el valor P_E.

$$P^2_{e_3} = \min P^2_j.$$

- **Paso 3**: Igual al paso 2 hasta que se cumplan algunas de las dos situaciones siguientes.:

-Todas las p variables ingresaron al modelo.

-Todas las variables incorporadas al modelo tienen valores de salida menor a P_R,y las no incorporadas valores de entrada mayores a P_E.

3.3 Interpretación de los coeficientes

Para interpretar los coeficientes en una regresión logística es necesario definir la ecuación obtenida de una manera más conveniente, expresando el modelo en términos de la chance de que un evento ocurra.

La chance de que el evento ocurra (Odds ratio) se define como el cociente entre la probabilidad de que el evento ocurra: $P(1/\mathbf{x})$ y la probabilidad de que no ocurra: $P(0/\mathbf{x})$

$$\begin{aligned} Odds\ ratio &= \frac{P(1/\mathbf{x})}{1 - P(1/\mathbf{x})} = e^{Z(\mathbf{x})} \\ \ln \frac{P(1/\mathbf{x})}{1 - P(1/\mathbf{x})} &= Z(\mathbf{x}) \end{aligned}$$

De esta manera, los coeficientes estimados se interpretan como el cambio en el logaritmo del

odds por un cambio unitario en la variable independiente asociada, permaneciendo las demás variables constantes. Por ejemplo, el cociente de odds para una variable independiente x_k dicotómica es igual a:

$$\begin{aligned} \textit{cociente de odds} \quad \Psi &= \frac{\frac{P(1/\mathbf{x}_j=1)}{1-P(1/\mathbf{x}_j=1)}}{\frac{P(1/\mathbf{x}_j=0)}{1-P(1/\mathbf{x}_j=0)}} = e^{w_j} \\ \ln \Psi &= w_j \end{aligned} \tag{3.12}$$

El valor de Ψ indica en cuánto aumenta (o disminuye) la probabilidad de que se presente el evento entre los individuos que pertenecen a la categoría $x = 1$ en relación con la de los que corresponden a la categoría de referencia ($x = 0$).

Este cociente se puede extender a variables independientes politómicas, tomando una de las categorías como referencia. Para una categoría cualquiera c, si la categoría de referencia asume el valor 0, la razón de odds será e^{w_c}.

3.4 Evaluación del modelo obtenido

Una vez que se han seleccionado las variables con una escala apropiada, se procede a ajustar la regresión logística y obtener los valores estimados.

El modelo obtenido ajusta adecuadamente si:

1-la suma de la diferencia entre los valores observados y los estimados en pequeña.

2- la contribución a esta suma de cada diferencia individual no es sistemática. y es pequeña en relación a la estructura de error del modelo.

Las medidas más utilizadas son: Pearson Residual y Deviance Residual. El primero está definido por la siguiente expresión:

$$Z_i = \frac{(k_i - P(k/\mathbf{x}_i))}{\sqrt{P(k/\mathbf{x}_i)(1 - P(k/\mathbf{x}_i))}} \tag{3.13}$$

El estadístico Chi Cuadrado de Pearson es la suma de estos residuales para todos los patrones considerados.

$$X^2 = \sum_{i=1}^{N} \frac{(k_i - P(k/\mathbf{x}_i))^2}{P(k/\mathbf{x}_i)(1 - P(k/\mathbf{x}_i))} = \sum_{i=1}^{N} Z_i^2 \tag{3.14}$$

esta es una medida de bondad de ajuste similar a la Suma de Cuadrados de los Residuos en la terminología de la Regresión Lineal.

La Deviance Residual permite la comparación de los valores observados con los predichos usando el método de verosimilitud, cuya expresión es la siguiente:

$$d_i = -2\left\{k_i \ln\left(\frac{P(k/\mathbf{x}_i)}{k_i}\right) + (1-y_i)\ln\left(\frac{1-P(k/\mathbf{x}_i)}{1-k_i}\right)\right\} \tag{3.15}$$

La deviance, definida en 3.10, es la suma de la Deviance Residual para todos los individuos:

$$D = \sum_{i=1}^{N} d_i^2 \tag{3.16}$$

Bajo el supuesto que el modelo ajusta correctamente, los estadísticos X^2 y D tienen distribución chi-cuadrado con $N-(p+1)$ grados de libertad. La deviance, como ya se analizó antes, es el test estadístico de razón de verosimilitud del modelo saturado respecto al modelo con p+1 parámetros. Estos estadísticos tienen el inconveniente que a medida que se incrementa el tamaño de la muestra, se incrementan los grados de libertad, y los p_valores que se obtienen en esta distribución no son correctos. Una forma de evitar esta dificultad es agrupando los datos en una tabla cuyas filas corresponden a los valores de la variable de salida. Hosmer y Lemeshow (1980-1982) proponen un agrupamiento de las probabilidades estimadas basado en los percentiles. De esta manera se obtienen 10 columnas con N/10 observaciones donde la primera corresponde a los de menor probabilidad estimada y la última al de mayor probabilidad. Si la variable respuesta tiene dos niveles se obtiene una tabla de contingencia de $2xg$. Con el mismo criterio que el estadístico Chi Cuadrado de Pearson, se calcula el estadístico $\mathbf{C}$ de la siguiente forma:

$$\mathbf{C} = \sum_{j=1}^{g} \frac{(\sum_{i=1}^{N_j} k_i - \sum_{i=1}^{N_j} P(k/\mathbf{x}_1))}{\sum_{i=1}^{N_j} P(k/\mathbf{x}_j)\left[1-\sum_{i=1}^{N_j} P(k/\mathbf{x}_j)\right]} \tag{3.17}$$

Este estadístico tiene distribución Chi Cuadrado con g-2 grados de libertad, y tiene la ventaja de poder obtener un ajuste de manera simple y fácilmente interpretable, logrando identificar deciles de riesgo. El inconveniente que presenta es que en el agrupamiento de los datos se pierde la desviación de los datos individuales, pero una vez que se admite el ajuste del modelo,

lo anterior es subsanable realizando un diagnóstico estadístico de los residuos individuales.

En regresión logística ha sido sugerido un tipo de R^2, el cual no es ampliamente aceptado, esta medida establece la proporción de la varianza que ha sido explicada por la regresión. En el caso que la variable respuesta sea dicotómica la varianza depende de la distribución de frecuencia de la variable, siendo su valor máximo cuando cada grupo tiene el 50% de la muestra. Esto significa que esta medida no puede compararse directamente en regresiones logísticas con diferentes distribuciones marginales de sus variables dependientes. Sin embargo, han sido propuesto algunas medidas de R^2, que utilizan en su cálculo la verosimilitud como el índice de Cox and Snell (1989), que se define como:

$$R^2 = 1 - \left[\frac{l_0}{lw}\right]^{\frac{2}{N}}$$

donde l_0 es la verosimilitud del modelo que incluye solo la constante y lw es la verosimilitud del modelo con todas las variables. Este índice tiene el inconveniente que no alcanza el valor 1 siendo si valor máximo igual a $1 - (lw)^{\frac{2}{N}}$. Nagelkerke (1991) propone una corrección, estandarizando sobre el valor máximo:

$$\widetilde{R}^2 = \frac{R^2}{R^2_{\max}} = 1 - \left[\frac{l_0}{lw}\right]^{\frac{2}{N}} \text{ siendo } R^2_{\max} = 1 - [l_0]^{2/N}$$

que permite que el valor máximo del índice sea 1. Estos índices pueden utilizarse complementariamente para verificar el ajuste del modelo ya que subestiman el mismo.

Se puede recurrir a una forma más intuitiva, pero de mucha utilidad, para resumir los resultados a través de una tabla de clasificación, donde se muestra la comparación cruzada entre los valores reales y los estimados. Cuando la variable respuesta es dicotómica, se le asigna el valor 0 o 1 comparando la probabilidad estimada con un determinado punto c de corte. Si la probabilidad estimada excede al valor c, entonces se le asigna el valor 1, en caso contrario asume el valor cero. El punto de corte más utilizado es 0,5. Esto permite construir una tabla donde en las filas se muestran los valores observados en cada grupo, y en las columnas los estimados. En las celdas de cada fila se muestra como han sido clasificados por el modelo los individuos del grupo considerado.

Si el modelo predice exactamente el grupo de pertenencia acorde a algún criterio, se consi-

	Valores estimados			
		Grupo 1	Grupo 2	
Valores Observados	Grupo1	Clasificación Correcta	Error	% Clasificación correcta Grupo 1
	Grupo2	Error	Clasificación Correcta	% Clasificación correcta Grupo 2
				% Clasificación correcta Global

Figura˜3-1: Tabla de clasificación

derada como una evidencia del ajuste del modelo. Pero a veces, existen situaciones en que el ajuste del modelo es correcto, sin embargo, la tabla de clasificación resulta con un alto porcentaje de error. La precisión en la clasificación no condiciona el criterio de ajuste basado en las distancias entre lo observado y lo estimado, cuyos valores no están sistematizados y contienen la variación del modelo. Todo depende también del objetivo del estudio que se realiza, ya que si el interés está dado en explicar cuáles son las variables y en cuánto afectan el hecho de pertenecer a un grupo o no, no será tan importante el porcentaje de error en la clasificación, aspecto que será de consideración cuando el objetivo sea de clasificación y predicción de pertenencia a un grupo.

Métodos de Diagnóstico.

Después de aceptar un ajuste correcto del modelo, es necesario inspeccionar el ajuste individual de los patrones para identificar aquellos están débilmente ajustados, y la influencia que ejercen sobre los parámetros del modelo. Existen un conjunto de medidas y gráficos que permiten realizar este análisis.

Entre los estadísticos de diagnóstico, se puede considerar:

1-Los residuos individuales, que son la base de las medidas consideradas en la evaluación del modelo. A partir de la probabilidad estimada de cada patrón, se definen los residuos $\widehat{y}_i - p_i$, el residuo estandarizado de la expresión 3.13 y la deviance residual definido en 3.15. Tanto el residuo estandarizado como la deviance residual, cuando las muestras son grandes tienen

distribución normal estandarizada.

A partir de la definición de la matriz $\mathbf{H}_{NxN}$:

$$\begin{aligned} \mathbf{H} &= \mathbf{V}^{1/2}(\mathbf{X}'\mathbf{V}\mathbf{X})^{-1}\mathbf{V}^{1/2} \text{ siendo } \mathbf{V}_{NxN} \text{ una matriz diagonal} \\ \text{donde cada elemento } v_i &= P(k/\mathbf{x}_i)(1-P(k/\mathbf{x}_i) \end{aligned} \tag{3.18}$$

que permite expresar el modelo como $\mathbf{y} = \mathbf{Hk}$, se determina el valor h_i denominado "influencia" que es cada elemento de la diagonal principal de $\mathbf{H}$,el que se puede calcular como:

$$h_i = P(k/\mathbf{x}_i)(1-P(k/\mathbf{x}_i)(1,\mathbf{x}_i)(\mathbf{X}'\mathbf{V}\mathbf{X})^{-1}(1,\mathbf{x}_i)' \tag{3.19}$$

Estos valores de influencia son usados para detectar observaciones que tienen un alto impacto sobre los valores estimados. Como se puede observar en la expresión 3.19, esta medida depende de los valores estimados y de la matriz de diseño $\mathbf{X}$, para interpretar sus valores es necesario la probabilidad estimada:

- Si la probabilidad estimada es un valor entre 0,1 y 0,9, h_i puede interpretarse como una distancia, entonces un valor grande indica que ese patrón se aleja del promedio.

-Pero cuando la probabilidad estimada es menor que 0,1 o mayor que 0,9, h_i tiende a cero, por lo que su valor no tiene utilidad

2- Valores que miden el efecto de cada individuo sobre el ajuste del modelo, es decir, los cambios que se producen en el modelo al eliminar un patrón.

La distancia de Cook, mide el cambio en los residuos al eliminar un individuo, y se calcula como:

$$D_i = \frac{Z_i^2 \times h_i}{(1-h_i)^2}$$

Para medir el cambio en el estadístico Chi Cuadrado de Pearson, se calcula la siguiente medida:

$$\Delta X^2 = \frac{Z_i^2}{(1-h_i)}$$

Y para medir el cambio en la Deviance:

$$\Delta D = \frac{d_i^2}{(1 - h_i)}$$

Estas medidas dependen del valor de influencia de cada patrón h_i, por lo que este valor trasmitirá su efecto expresado en el punto anterior. En la tabla de la Figura 3.2, se resumen los valores que deberían tomar estas medidas según la probabilidad estimada y el valor real que asume cada patrón.

k= 0					
	Probabilidad estimada				
Medidas	0-0,10	0,10-0,30	0,30-0,70	0,70-0,90	0,90-1
hi	Pequeño	Grande	Moderado a pequeño	Grande	Pequeño
ΔX^2	Pequeño	Moderado	Moderado a pequeño	Moderado	Pequeño
Di	Pequeño	Grande	Moderado	Grande	Pequeño
k= 1					
	Probabilidad estimada				
Medidas	0-0,10	0,10-0,30	0,30-0,70	0,70-0,90	0,90-1
hi	Pequeño	Grande	Moderado a pequeño	Grande	Pequeño
ΔX^2	Grande	Moderado	Moderado a pequeño	Moderado	Grande
Di	Pequeño	Grande	Moderado	Grande	Pequeño

Figura˜3-2: Valores que asumen las medidas de diagnóstico.

3- Valores que miden el efecto de cada individuo sobre el valor de los parámetros estimados.

Otra medida de diagnóstico es el cambio en cada parámetro cuando el patrón es eliminado. Este cambio será la diferencia entre el parámetro que contiene el patrón y el parámetro que resulta cuando el mismo es eliminado, por ejemplo para el coeficiente $\mathbf{w}_1$y el patrón i:

$$d\mathbf{w}_1^i = \mathbf{w}_1 - \mathbf{w}_1^i \tag{3.20}$$

donde $\mathbf{w}_1$ es el coeficiente que incluye todos los casos, y $\mathbf{w}_1^i$ es el valor del coeficiente con el caso i-ésimo excluído.

Cuando esta diferencia es grande se identifican observaciones que deben ser analizadas.

La bibliografía sugiere gráficos que permiten mostrar diferentes aspectos del ajuste, entre

los cuales se pueden considerar:

- Evaluar la normalidad de la Derviance (cuando la muestra es grande)
- Z_i vs número de caso y h_i vs número de caso, permiten identificar casos con altos residuos estanzarizados o altos valores palanca respectivamente.
- ΔX^2vs $P(k/\mathbf{x}_i), D_i$vs $P(k/\mathbf{x}_i), \Delta D$ vs $P(k/\mathbf{x}_i)$, permiten detectar aquellos casos que no son ajustados correctamente por el modelo, identificando el grupo al cual pertenecen.
- $d\mathbf{w}_j^i$ vs número de caso, permite mostrar los casos que producen cambios importantes en cada parámetro.

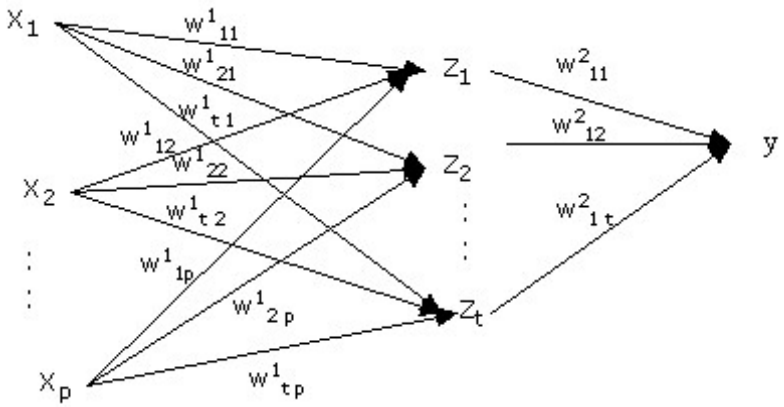
Capítulo 4

Perceptron Multicapa

Como se analizó en el capítulo 2, el perceptron simple resulta una buena regla de clasificación sólo si los grupos son linealmente separables. Estas observaciones fueron hechas por Minsky and Papert (1969), quienes demostraron con rigurosidad matemática las limitaciones del perceptron simple para generalizar sus resultados a todo tipo de problemas y sugirieron como línea de investigación la extensión a sistemas con variables intermedias o perceptron multicapa. La dificultad para encontrar un algoritmo de estimación de los parámetros fue uno de los factores que retrasó la investigación en este área. Aunque algoritmos de estimación de los parámetros fueron inventados hace tiempo atrás, ellos no fueron considerados hasta los años 80 donde los avances de la computación permitieron su implementación práctica; aparece entonces el Perceptron multicapa como una extensión del perceptron simple.

En un perceptron multicapa además de las variables de entrada y salida, se generan variables intermedias que forman las capas ocultas. Entonces, la red que representa un perceptron multicapa consiste en un conjunto de variables que constituyen la capa de entrada, una o más capas ocultas constituidas por unidades de procesamiento llamadas nodos o variables intermedias y una capa de salida, esta última está formada por las respuestas de las entradas procesadas hacia delante a través de las capas ocultas. Entre una capa de variables y la siguiente, se sigue el mismo procedimiento que un perceptron simple. La cantidad de variables en cada capa, puede ser mayor, igual o menor que la cantidad de variables de la capa anterior.

La figura siguiente corresponde a la estructura de un perceptron multicapa con una sola capa oculta, con P nodos o variables en la capa de entrada, T nodos en la capa oculta, y 1 nodo



Figura˜4-1: Estructura de un perceptron multicapa con una capa oculta

en la capa de salida.

El supraíndice de los parámetros w_{ij}, indica la capa de neuronas o variables a las que corresponde. Para cada uno de los N ejemplos del conjunto, cada variable z_t de la capa oculta se calcula como:

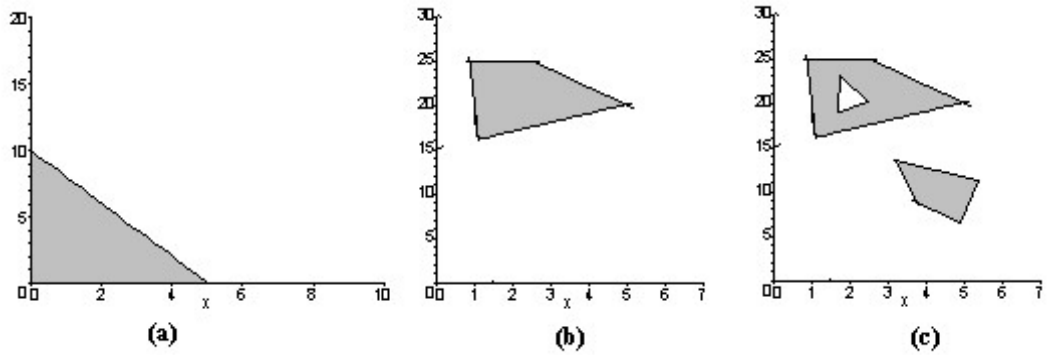
$$z_t = f\left(\sum_{p=1}^{P} w_{tp}^1 x_p\right) \tag{4.1}$$

y la variable de salida:

$$y = f\left(\sum_{t=1}^{T} w_{1t}^2 z_t\right) = f\left\{\sum_{t=1}^{T} w_{1t}^2 f_t\left(\sum_{p=1}^{P} w_{tp}^1 x_p\right)\right\} \tag{4.2}$$

donde los parámetros w_{tp}^1 conectan la p- ésima variable de entrada con la variable t-ésima de la capa oculta, y w_{1t}^2 conecta la t- ésima variable de la capa oculta con la variable de salida. f representa la función identidad, logística, tangente o la función escalón, entre otras. Puede definirse una estructura de red donde no se definan algunas conexiones entre las variables, es decir, algunos parámetros w_{ij} no son considerados (ó,equivalentemente, $w_{ij} = 0$).

Las funciones de activación pueden ser diferentes en las distintas capas intermedias y en la capa de salida. Considerando la función de activación escalón:



Figura˜4-2: Regiones de decisión según distintas estructuras de redes

$$\Theta(x) = \begin{cases} 0 \text{ cuando } x < 0 \\ 1 \text{ cuando } x \geq 0 \end{cases} \tag{4.3}$$

las variables de entrada binarias $x_i = 0, 1$ y las variables de salida definidas entre los valores 0 y 1, la red que se genera representa una función booleana.

Cuando no hay capas intermedias, como se mostró en el capítulo 2 el límite de decisión constituía un hiperplano como en la Figura 4.2 a. Pero al considerar variables intermedias, cada capa intermedia genera un hiperplano, de tal modo que la red puede formar límites de decisión alrededor de una región convexa, como en la figura 4.2 b, donde los límites corresponden a segmentos de hiperplanos. Cuando hay dos capas intermedias se pueden generar regiones de decisión arbitrarias, las cuales pueden ser no convexas o disjuntas (Figura 4.2.c)

Puede ser que variables de entrada sean continuas y las funciones de activación sigmoidales (como la logística o la tangente hiperbólica), entonces la red representa un función continua de las variables. Cuando una red neuronal utiliza en su capa intermedia la función de activación tangente, es equivalente a utilizar la función de activación logística con diferentes valores en sus parámetros, ya que ambas difieren por una transformación lineal.

El uso de las funciones sigmoidales en las variables de salidas permite limitar el rango de valores de las mismas, lo que es deseable en algunos casos como la función logística que juega un importante rol permitiendo que los valores de salida sean interpretados como probabilidades.

Esto depende del problema considerado: para los problemas de clasificación donde se estima el grupo de pertenencia dado un conjunto de características $P(k/\mathbf{x})$,la función de activación para las variables de salida es la logística, pero en otros tipos de problemas puede considerarse incluso la lineal.

En el caso especial de problemas de clasificación, las redes con una capa intermedia con función de activación sigmoidal se aproximan a la región de decisión, con bastante acierto, considerando un número de variables de la capa intermedia lo suficientemente grande. El aumento de capas intermedias, en algunos casos, permite una aproximación más eficiente en el sentido de alcanzar el mismo nivel de acierto con menor cantidad de variables en cada una.

4.1 Algoritmo de aprendizaje.

Como ya se señaló, el aprendizaje de una red consiste en minimizar una función error E respecto de los parámetros, como se definió en 2.17. La expresión de la función E (para un ejemplo) será ahora :

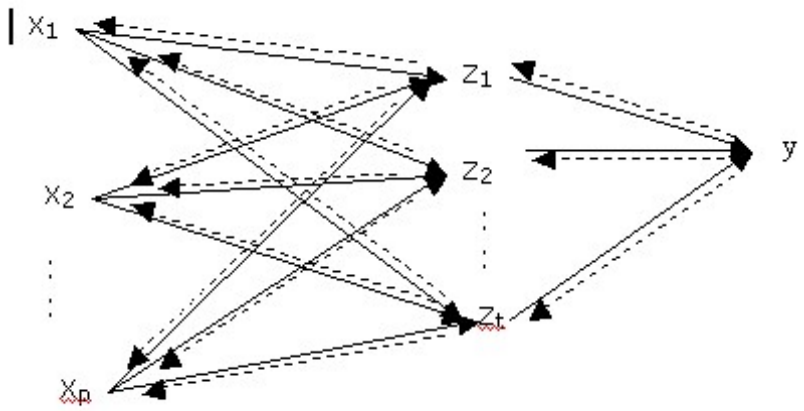
$$E = \frac{1}{2}\left[k - f\left\{\sum_{t=1}^{T} w_{1t}^{2} f_t\left(\sum_{p=1}^{P} w_{tp}^{1} x_p\right)\right\}\right]^2 \tag{4.4}$$

La matriz de parámetros $\mathbf{W}$ es estimada a partir de una muestra de entrenamiento, utilizando *el algoritmo de retropropagación de errores (error backpropagation) o regla Delta generalizada*. Este algoritmo es considerado dentro de las técnicas heurísticas, ya que está desarrollado desde un análisis de los resultados del algoritmo estándar del gradiente descendente. De manera similar a la regla del aprendizaje, el algoritmo de retropropagación realiza una corrección de los parámetros proporcional a la derivada de la función de costo respecto a los parámetros $\left(\frac{\partial E}{\partial w}\right)$, el cual puede ser expresado de la siguiente manera:

$$\triangle w = -\eta \frac{\partial E}{\partial w} \tag{4.5}$$

aplicando la regla de la cadena

$$\frac{\partial E}{\partial w} = \frac{\partial E}{\partial y}\frac{\partial y}{\partial v}\frac{\partial v}{\partial w} = -(y-k)f'(v)z$$



Figura˜4-3: Retropropagación de errores en la red

$$\text{siendo } v \quad = \quad \sum_{t=0}^{T} w_{1t}^{2} z_t \tag{4.6}$$

donde y viene dado por (4.2) y z_t por (4.1), de esta manera :

$$\triangle w_{1t}^{2} = \eta \delta_1 z_t \text{ siendo } \delta_1 = (y-k) f'(v) \tag{4.7}$$

Las variables de salida son comparadas con los valores deseados k cuyas entradas son las salidas que resultan de la última capa oculta. De esta manera, los valores de los parámetros que las conectan pueden ser calculados en forma directa por la regla del aprendizaje del perceptron definida en el capítulo 2. Cuando una neurona pertenece a la capa de salida, habiendo determinado $y-k$, se computa el valor del gradiente a partir de la expresión 4.4. Pero este procedimiento no puede ser utilizado en las capas intermedas, ya que no es posible asignarle un valor k para el cálculo de E, es decir no tienen especificada una respuesta deseada. Sin embargo, estas neuronas también son responsables de los errores cometidos en las variables de salida. El problema es determinar qué parte del error debe asignarse a cada unidad de procesamiento, y se resuelve retropropagando los errores a través de la red, como se muestra en la figura 4.3.

Para una variable oculta, la señal de error debe determinarse recursivamente en términos de las señales de error de todas las variables con la cual está directamente conectada, es en este

punto donde se complica el algoritmo.

Utilizando la expresión (4.5), la actualización de los parámetros en una capa oculta será igual a:

$$\triangle w_{tp} = \eta \delta_t x_p \tag{4.8}$$

De esta manera para la variable t de la capa intermedia, δ_t se calcula de la siguiente manera:

$$\begin{aligned} \delta_t &= -\frac{\partial E}{\partial z_t}\frac{\partial z_t}{\partial u_t} \text{ siendo } u_t = \sum_{p=0}^{P} w_{tp}^1 x_p \\ \delta_t &= -\frac{\partial E}{\partial z_t} f_t'(u_t) \end{aligned} \tag{4.9}$$

siendo $E = \frac{1}{2}(y-k)^2$ y expresando $e = y - k$, por la regla de la cadena, $\frac{\partial E}{\partial z}$ se puede expresar como:

$$\begin{aligned} \frac{\partial E}{\partial z_t} &= e\frac{\partial e}{\partial z_t} = e\frac{\partial e}{\partial v}\frac{\partial v}{\partial z_t} \\ &= ef'(v)w_{1t} \\ &= \delta_1 w_{1t} \text{ siendo } \delta_1 = ef'(v) \end{aligned} \tag{4.10}$$

Reemplazando (4.10) en (4.9), δ_t resulta:

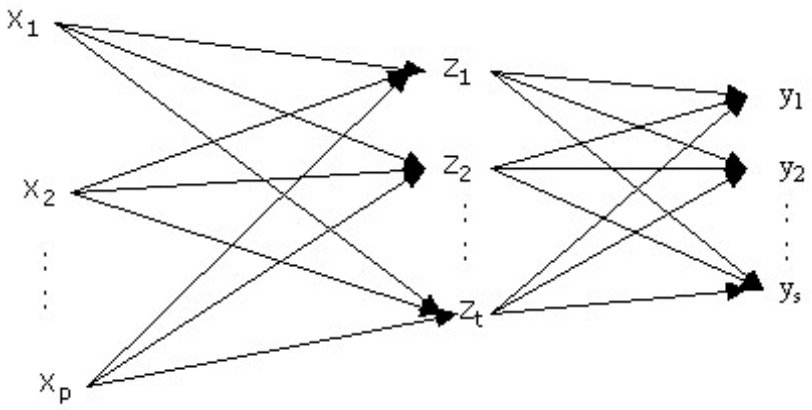
$$\delta_t = \delta_1 w_{1t} f_t'(u_t)$$

Considerando una red con más de una variable de salida, por ejemplo una red con S variables de salida $y_1, y_2, y_3, ..., y_S$, como muestra la figura 4.4.

Cada variable de salida, se puede expresar de la siguiente manera:

$$y_s = f\left(\sum_{t=1}^{T} w_{st}^2 z_t\right) \tag{4.11}$$

$$E = \frac{1}{2}\sum_{s=1}^{S}(y_s - k_s)^2 = \frac{1}{2}\sum_{s=1}^{S} e_s \tag{4.12}$$



Figura˜4-4: Perceptron multicapa con más de una variable respuesta

En este caso, δ_t tienen la siguiente expresión:

$$\delta_t = f_t'(u_t) \sum_{s=1}^{S} \delta_s w_{st}$$

En resumen, la actualización de los parámetros será igual a:

$$\triangle w_{ij} = \eta \delta_i x_j \tag{4.13}$$

donde δ_i depende si se trata de una variable de salida o una variable intermedia.

- Si la **variable** i **es de salida**, δ_i es el producto entre la derivada de la combinación lineal de las variables de que están conectadas con la varible de salida $f_i'(v_i)$ por la señal de error e_i, ambos asociados a una neurona de salida.
- Si la **variable** i **es intermedia** δ_i es igual al producto entre la derivada $f_i'(v_i)$ y la suma ponderada de los δ computados para las variables de las capas hacia la cual se conecta la variable i considerada.

En la aplicación del algoritmo backpropagation, deben distinguirse dos pasadas de computación por la arquitectura de la red:

- la **primera pasada hacia delante**, desde la capa de entrada 0 hacia la capa de salida

L(forward), considerado l el número de capas igual a $l = 0, 1, 2, 3, ..., L$, y n^l el número de nodos en cada capa. Se inicia con una matriz **W** formada por valores aleatorios uniformemente distribuidos cercanos a cero, y se calculan los valores de las variables en cada capa como:

$$x_j^l = f\left(\sum w_{ji} x_i^{l-1}\right)$$

Una vez presentados todos los ejemplos, para cada variable de la capa de salida, se computa el error como la diferencia entre la el valor observado de la variable y el que resulta de aplicar el algoritmo (o grupo de pertenencia estimado).

- la **segunda pasada hacia atrás**, desde la capa de salida a la capa de entrada (backward), la señal de error se va trasladando capa por capa y calculando recursivamente el gradiente local δ para cada variable. Este proceso permite que los parámetros w_{ij} sean sometidos a cambios acorde con la regla Delta.

En la práctica cuando se aplica el algoritmo de retropropagación, el aprendizaje es a partir de un conjunto de ejemplos que forman la muestra de entrenamiento. Una presentación completa del conjunto de ejemplos durante el proceso de entrenamiento se denomina **período.** El proceso de aprendizaje se mantiene período a período, hasta que los parámetros se estabilizan y la red converge a un valor mínimo de la función de costo.

4.1.1 Modalidades del algoritmo de aprendizaje

El proceso de aprendizaje puede proceder de dos maneras diferentes:

Modalidad ejemplo por ejemplo (on-line): en este caso los parámtros se actualizan después de la presentaciòn de cada ejemplo. Si el conjunto de entrenamiento posee N ejemplos (x_i, k_i), $i = 1, 2, 3, ..., N$, se presenta a la red el primer ejemplo (x_1, k_1) al cual se aplica el algoritmo de retropropagación resultando cierto ajuste a los parámetros de la red. Después, se presenta el segundo ejemplo (x_2, k_2), repitiendo la secuencia computacional hacia delante y retropropagando la red, resultando nuevos ajustes. Este proceso continúa hasta el último ejemplo (x_N, k_N). En este caso el ajuste computado en un paràmetro w_{ji}, está dado por el promedio de los cambios

realizados para cada ejemplo en el momento o período n

- **Modalidad en conjunto (batch):** a diferencia del proceso anterior, en este caso los parámetros son actualizados después que se presentan todos los ejemplos, calculando el error cuadrático medio como funcion de costo para el período n:

$$\overline{E} = \frac{1}{2N} \sum_{n=1}^{N} \sum_{s=1}^{S} e_s(n) \text{ siendo } S \text{ el número de variables de salida} \tag{4.14}$$

El cambio en los parámetros es diferente según el método utilizado, la modalidad ejemplo por ejemplo es preferida computacionalemente ya que requiere menos capacidad de almacenamiento, y si los ejemplos son presentados a la red de manera aleatoria es menos probable que el algoritmo encuentre un mínimo local. Por otro lado la modalidad en conjunto, asegura un estimador más exacto del vector gradiente. La efectividad del modo de entrenamiento elegido, depende del problema. (Hertz, 1991) .

4.1.2 Parámetro de momento

Como se comentó en la unidad 2, la tasa de aprendizaje η, afecta el tiempo de convergencia del algoritmo de aprendizaje, siendo mayor el tiempo a medida que η tiende a cero, pero cuando η está más cerca de uno el algoritmo se vuelve inestable. Para acelerar el tiempo de convergencia del algoritmo evitando que el sistema se torne inestable, en la actualización de los parámetros se incorpora un término de inercia o momento utilizando un **parámetro de momento** α. Este parámetro, evita que el proceso de aprendizaje termine oscilando en un mínimo local, mejorando el comportamiento del algoritmo de aprendizaje. Los cambios en los parámetros se calculan con la siguiente expresión:

$$\triangle w_{ij}(n) = \eta \delta_i(n) x_j(n) + \alpha w_{ij}(n-1) \tag{4.15}$$

Para que la serie de tiempo sea convergente se tiene que verificar $0 \leq |\alpha| < 1$, cuando $\alpha = 0$, el algoritmo trabaja sin parámetro de momento. Aunque la condición de convergencia, no establece condición de signo para α, en la práctica se utilizan valores positivos. Respecto a su valor, cuando la derivada parcial $\frac{\partial E(n-t)}{\partial w_{ij}(n-t)}$ presenta el mismo signo en iteraciones sucesivas

permite acelerar el aprendizaje , mientras que cuando la derivada presenta oscilaciones de signo, su valor disminuye y el término de momentos tiene un efecto estabilizador.

4.1.3 Parámetro de aprendizaje adaptativo

No es posible definir un parámetro de aprendizaje apropiado para cada problema en particular, y posiblemente un valor del parámetro determinado como bueno al iniciar el entrenamiento, no lo es durante el proceso de aprendizaje. Ante esto, algunos autores sugieren un ajuste automático del parámetro η a medida que avanza el proceso de aprendizaje, asignando un parámetro de aprendizaje a cada w_{ij} y variando su valor de una iteración a otra, para lo cual debe realizarse una modificación en el algoritmo de retropropagación. Denotando con $\eta_{ij}(n)$ al parámetro de aprendizaje correspondiente al parámetro $w_{ij}(n)$ en la iteración n, se puede escribir:

$$\frac{\partial E(n)}{\partial \eta_{ij}(n)} = \frac{\partial E(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial \eta_{ij}(n)} \text{ siendo } v_j(n) = \sum_i w_{ji}(n) z_i(n) \tag{4.16}$$

el parámetro $w_{ji}(n)$ se calcula como:

$$w_{ji}(n) = w_{ji}(n-1) - \eta_{ij}(n) \frac{\partial E(n-1)}{\partial w_{ji}(n-1)} \tag{4.17}$$

por lo que se puede reemplazar $w_{ji}(n)$ en la expresión de $v_j(n)$ de la siguiente manera:

$$v_j(n) = \sum_i z_i(n) \left[w_{ji}(n-1) - \eta_{ij}(n) \frac{\partial E(n-1)}{\partial w_{ji}(n-1)} \right] \tag{4.18}$$

la derivada de $v_j(n)$ respecto $\eta_{ij}(n)$ resulta:

$$\frac{\partial v_j(n)}{\partial \eta_{ij}(n)} = -z_i(n) \frac{\partial E(n-1)}{\partial w_{ji}(n-1)} \tag{4.19}$$

conociendo además que:

$$\frac{\partial y_j(n)}{\partial v_j(n)} = f_j'(v_j(n)) \text{ y } \frac{\partial E(n)}{\partial y_j(n)} = -e_j(n) \tag{4.20}$$

reemplazando 4.25 y 4.24 en 4.21:

$$\frac{\partial E(n)}{\partial \eta_{ij}(n)} = f_j'(v_j(n))e_j(n)z_i(n)\frac{\partial E(n-1)}{\partial w_{ji}(n-1)} \tag{4.21}$$

y utilizando la relación 4.6:

$$\frac{\partial E(n)}{\partial \eta_{ij}(n)} = -\frac{\partial E(n)}{\partial w_{ji}(n)}\frac{\partial E(n-1)}{\partial w_{ji}(n-1)} \tag{4.22}$$

Ahora se puede formular una actualización de la tasa de aprendizaje η_{ij}, definiendo el siguiente ajuste:

$$\Delta\eta_{ij}(n+1) = \begin{cases} a \text{ si } A_{ij}(n-1)\frac{\partial E(n)}{\partial w_{ji}(n)} > 0 \\ -b\eta_{ij}(n) \text{ si } A_{ij}(n-1)\frac{\partial E(n)}{\partial w_{ji}(n)} < 0 \\ 0 \text{ para otro caso.} \end{cases} \tag{4.23}$$

definiendo $A_{ij}(n-1)$ como:

$$A_{ij}(n-1) = (1-\xi)\frac{\partial E(n-2)}{\partial w_{ji}(n-2)} + \xi A_{ij}(n-2)$$

siendo ξ una constante positiva.

Los signos de $A_{ij}(n-1)$ y $\frac{\partial E(n)}{\partial w_{ji}(n)}$ determinan la variación de η_{ij}, ya que si son del mismo signo la tasa de aprendizaje presenta un crecimiento constante, mientras que si difieren en el signo, en una proporción b del valor corriente de $\eta_{ij}(n)$,en cualquier otro caso la tasa de aprendizaje no cambia. En caso que ambos valores a y b sean iguales a cero la tasa de aprendizaje asume un valor constante, siendo necesario ir guardando los valores $\eta_{ij}(n)$, $\Delta\eta_{ij}(n)$, y $\frac{\partial E(n-1)}{\partial w_{ji}(n-1)}$ para lo cual se necesitará memoria adicional. Esta complejidad computacional puede ser reducida utilizando la actualización completa de los ejemplos, en cuyo caso el gradiente $\frac{\partial E(n)}{\partial w_{ji}(n)}$ se calcula como promedio de las contribuciones computadas de los diferentes ejemplos presentados, es decir, se realiza una actualización por período.

4.1.4 Valores iniciales de los parámetros

Los parámetros iniciales en el proceso de entrenamiento, deben estar uniformemente distribuidos en un intervalo pequeño, para reducir el riesgo de una saturación de la red (donde el error cuadrático medio permanece constante durante un período y luego vuelve a decrecer lo que se denomina punto de silla), o que se obtengan valores del gradiente muy pequeños. Sin embargo, este valor no debe ser muy pequeño, ya que causaría derivadas del error muy pequeñas y el proceso de aprendizaje iniciaría muy lentamente.

Cuando las funciones de activación son sigmoidales con parámetro a igual a 1 (logística o tangente hiperbólica), se pueden definir valores de los w_{ij} dentro del intervalo $\left[-\frac{1}{\sqrt{n_i}}, \frac{1}{\sqrt{n_i}}\right]$ siendo n_i el número de entradas a la neurona i.

Nguyen y Widrow (1990) determinaron un algoritmo para encontrar los parámetros iniciales, que tiene como ventaja reducir el tiempo de entrenamiento. Este algoritmo selecciona valores que permitan distribuir la región afectada por cada variable oculta uniformemente sobre el espacio de las variables de entrada. Para ello se asignan valores aleatorios uniformemente distribuidos en el intervalo [-1,1] a los parámetros determinando un vector $\mathbf{w_{rn}}$, luego se ajusta la magnitud del vector $\mathbf{w}$, de modo tal que cada variable intermedia es lineal en un intervalo pequeño. Considerando que hay L variables en la capa intermedia, la magnitud de $\mathbf{w}$ se determina como:$|\mathbf{w}| = 0,7L^{\frac{1}{N}}$. De esta manera el vector de parámetros iniciales será igual a:

$$\mathbf{w}_{ini} = \mathbf{w_{rn}} \times |\mathbf{w}|$$

y el valor del sesgo $\mathbf{w}_{bi}$ se establece como un valor uniforme aleatorio entre $-|\mathbf{w}_i|$ y $|\mathbf{w}_i|$.

4.1.5 Criterios de parada del algoritmo

En puntos anteriores, fueron sugeridas variaciones del algoritmo para acelerar el aprendizaje, u otros para evitar que el algoritmo quede en un mínimo local; en este punto se presentan distintos criterios propuestos para parar el proceso de aprendizaje, es decir distintos criterios de convergencia. Se considera que el algoritmo de retropropagación converge cuando:

- la tasa de cambio del error cuadrático medio por período es lo suficientemente pequeña.

- el error cuadrático medio es menor a un valor umbral determinado para el modelo
- cuando para el vector de parámetros del último período $w_{(fin)}$, el valor absoluto de su gradiente es menor o igual a un valor fijo muy pequeño, establecido para el modelo.
- en caso de utilizar un conjunto de testeo, cuando el resultado de la generalización es adecuado, es decir, alcanza los valores estipulados para el modelo.

En algunos casos se puede establecer como criterio de parada, un número máximo de repeticiones del algoritmo o un tiempo máximo de entrenamiento.

En caso de utilizar un conjunto de validación durante el entrenamiento de la red, puede establecerse como criterio de parada un tope máximo a la cantidad de veces que el error cuadrático medio de este conjunto aumenta, desde la última vez que disminuyó durante el entrenamiento.

4.2 Consideraciones generales de la estructura de un perceptron multicapa.

Al definir una estructura de red neuronal, cabe preguntarse cuántas capas de variables intermedias se deben agregar y cuántas variables dentro de cada capa. La respuesta depende del conocimiento que se tenga de la función discriminante $f(\mathbf{x}, \mathbf{w})$, a aproximarse con el modelo de red. El algoritmo de retropropagación de errores es un método práctico que permite encontrar los parámetros, pero no proporciona ninguna ayuda respecto a la estructura de red que se debe seleccionar. El objetivo que se persigue es alcanzar una tasa de clasificación correcta que sea aceptable, siendo éste el criterio que determinará la cantidad de variables del modelo. Es posible demostrar que una red neuronal puede aproximar a una función discriminante continua, pero no la cantidad de capas ocultas que se deben proponer. Los resultados de simulaciones realizadas en problemas de clasificación muestran que dos capas son suficientes para aproximar al valor de $f(\mathbf{x}, \mathbf{w})$. Más de dos capas intermedias, permite una solución con menor cantidad de variables en total, y permite acelerar el aprendizaje, pero esto no aporta nada a la generalización de los resultados, es decir, permite una aproximación a la función pero esto no significa que ha aprendido correctamente.

Hay diferentes maneras de controlar la complejidad de la función discriminante, algunas pueden ser:

- cambiar el número de capas intermedias y de las variables dentro de cada capa intermedia
- eliminar algunos parámetros (eliminando algunas conexiones entre las variables)

Pocas capas ocultas, posiblemente no permita encontrar los parámetros necesarios para aproximar la función $f(\mathbf{x}, \mathbf{w})$, en este caso falla el proceso de aprendizaje. Si el número de capas ocultas es demasiado grande, existirán diferentes soluciones, la mayoría de las cuales no permitirán una correcta generalización de los resultados a nuevos ejemplos. Un camino que se puede seguir, es ir construyendo la red en forma incremental, comenzando con una estructura reducida, e ir incrementando las variables dentro de cada capa intermedia y las capas intermedias a medida que se van testeando los resultados.

4.2.1 Tasa de aprendizaje y tasa de momento óptima

La tasa de aprendizaje y la tasa de momento pueden definirse como óptimos cuando en promedio: permitan encontrar un mínimo local de la función error, o en el mejor de los casos el mínimo global; o cuando permite la convergencia de una configuración de red con la cual se logra la mejor generalización.

La convergencia para lograr la mejor generalización es, en la práctica, la más importante, pero la más difícil de lograr ya que el algoritmo utiliza el criterio de optimizar el error cuadrático medio de la muestra de entrenamiento, y esto no implica lograr una buena generalización de los resultados.

Haykin et al. muestra los resultados de simulaciones de multiperceptrones con dos capas ocultas y diferentes combinaciones de α y η, utilizando siempre los mismos parámetros iniciales y el mismo conjunto de entrenamiento. Obtiene como conclusión que los valores óptimos de α y η para converger a un mínimo local en la superficie de la función error son 0,1 y 0,55 respectivamente, aunque en Hertz (1991) se recomienda un valor de α cercano a 0,9.

4.2.2 Generalización

Ya se dijo que para evaluar los resultados del modelo en problemas de clasificación la probabilidad a posteriori de individuos bien clasificados se define como una estimación de $P(k/\mathbf{x})$. El algoritmo de retropropagación va calculando los parámetros minimizando el error cuadrático medio. Haykin muestra que lograr una menor tasa de error en el conjunto de entrenamiento no es una condición suficiente para lograr una correcta tasa de clasificación. Para ello es necesario medir el porcentaje de individuos correctamente clasificados con un conjunto de ejemplos de la misma población no utilizados en el entrenamiento de la red, Ya que si la red está sobre-entrenada memoriza los ejemplos del conjunto de entrenamiento, y no permite una correcta generalización de los mismos. El criterio para la selección de un modelo es que en ausencia de conocimiento previo de la función discriminante se debe elegir la función más simple, es decir la estructura de red más simple que aproxime el error de clasificación dado, demandando el menor esfuerzo computacional.

Hay tres aspectos que afectan la generalización de los resultados de un modelo de red:

1. La arquitectura de la red
2. El tamaño y eficiencia del conjunto de entrenamiento
3. La complejidad del modelo que se está tratando.

Es claro que sobre el último punto no se tiene control. En la práctica el conjunto de entrenamiento está fijado de antemano, por lo que interesa determinar la mejor arquitectura de la red que permita alcanzar una buena generalización. Esto es como se ha analizado en puntos anteriores: determinar el número de nodos, número de variables, valores de α y η,etc. Sin embargo, dada una estructura de red (acorde al tipo de problema con el cual se está trabajando), se podría determinar cuál es el tamaño de ejemplos necesarios para lograr una buena generalización de los resultados.

Baum y Haussler (1989) determinaron el tamaño de muestra mínimo que se debía considerar para una modelo de red con una capa intermedia utilizado para una clasificación binaria, para

que la red provea una buena generalización:

$$N \geq \frac{32W}{\varepsilon} \ln\left(\frac{32M}{\varepsilon}\right) \tag{4.24}$$

donde M corresponde al número total de variables en la capa intermedia, W es el total de parámetros en la red, N es el número de ejemplos que se deben presentar y ε la fracción de errores permitidos en la muestra de testeo. Esta fórmula da un tamaño de muestra muy grande, que en la práctica se puede alejar bastante del conjunto de datos disponibles, pero de todos modos puede considerarse como un criterio extremo.

Eliminando el logaritmo natural de la expresión 4.24, el tamaño del conjunto de entrenamiento es directamente proporcional a la cantidad de parámetros e inversamente proporcional a la tasa de error de la muestra de testeo. En la práctica se puede simplificar la expresión 4.24 de la siguiente manera:

$$N \geq \frac{W}{\varepsilon}$$

de esta manera el número de ejemplos de la muestra de entrenamiento debe ser aproximadamente una tasa ε del número de parámetros de la red.

4.3 Procesamiento previo de las variables iniciales

Para lograr un correcto aprendizaje, es necesario asegurarse de haber realizado un correcto preprocesamiento, en la mayoría de las aplicaciones es necesario transformar los datos de entrada antes de iniciar el entrenamiento de la red. Este preprocesamiento de la información es un factor importante en el resultado final obtenido por el modelo, e incluye la reducción de la dimensionalidad del espacio y la transformación de la variables iniciales. En todos lo casos es importante para encontrar una solución utilizar el conocimiento previo del problema a tratar, el cual puede ser incorporado en el diseño de la estructura de la red o en el preprocesamiento de las variables.

Una de las formas más comunes de preprocesamiento consiste en un reescalamiento de las variables iniciales, ya que diferentes unidades de medida hacen que las mismas difieran en magnitud, lo cual no permite reflejar su importancia relativa en determinar los valores de salida.

Para esto se sugiere estandarizar las variables:

$$\widetilde{x_i} = \frac{x_i - \overline{x_i}}{\sigma_{x_i}} \tag{4.25}$$

siendo $\widetilde{x_i}$ la variable x_i transformada, la cual tiene media cero y desviación estandar 1. Esto asegura que todas las variables asumen valores alrededor de cero. Esta transformación considera a las variables como independientes entre sí, otra trasnformación lineal más sofisticada (Fukunaga,1990) agrupa las variables en un vector $\mathbf{x}$, y calcula su vector de medias $\boldsymbol{\mu}$ y matriz de covarianzas $\boldsymbol{\Sigma}$; considerando la ecuación de valores propios de la matriz de covarianzas $\boldsymbol{\Sigma}\mathbf{u}_j = \lambda_j \mathbf{u}_j$ se define la transformación lineal como

$$\widetilde{\mathbf{x}^n} = \boldsymbol{\Lambda}^{-1/2}\mathbf{U}'(\mathbf{x}^n - \overline{\mathbf{x}}) \tag{4.26}$$

siendo $\mathbf{U} = (\mathbf{u}_1; \mathbf{u}_2, ..., \mathbf{u}_p)$ y $\boldsymbol{\Lambda} = \mathbf{diag}(\lambda_1, \lambda_2, ..., \lambda_p)$.

En otros casos, se puede realizar un reescalamiento de tal modo que las variables estén en un rango especificado, por ejemplo se pueden normalizar las variables iniciales y las respuestas deseados de tal modo que estén en el rango [-1,1], de la siguiente manera:

$$\widetilde{x_i} = 2\left(\frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)}\right) - 1 \tag{4.27}$$

siendo $\min(x_i)$ y $\max(x_i)$, el valor mínimo y máximo que puede alcanzar la variable x_i.

Para el caso de variables ordinales estas pueden ser tratadas como variables numéricas; si las variables son nominales se crean tantas variables dummy como categorías.

Para reducir la dimensionalidad del conjunto de variables de entrada puede seleccionarse un subconjunto de las mismas y eliminar las restantes; se puede empezar con aquellas variables muy correlacionadas y de esta manera eliminar información repetida. Al elegir un subconjunto se debe determinar el criterio que se utilizará para decidir si un subconjunto es mejor que otro que en principio es el mismo que se utiliza para evaluar el modelo; por otra parte, se debe determinar un procedimiento sistemático que permita ir buscando los subconjuntos candidatos a ser seleccionados, lo que implica un gran esfuerzo computacional aunque sean pocas las variables (para diez variables se pueden definir 1024 subconjuntos a considerar). Es claro que

el subconjunto de características óptimo que se obtiene a partir de un conjunto inicial depende entre otros aspectos del modelo para el cual será utilizado; en este sentido, el criterio de selección para un modelo de red neuronal sería entrenando los diferentes subconjuntos e ir evaluando sus resultados en el conjunto de testeo (considerando como criterio de selección la probabilidad de mal clasificados), lo que se hace imposible ya que el requerimiento de tiempo de procesamiento sería muy grande. Por esto es común seleccionar un subconjunto de variables utilizando un modelo lineal (discriminante lineal o logístico con sus métodos de selección hacia adelante o eliminación hacia atrás) y luego usar las variables seleccionadas en un modelo no-lineal más complejo. Cuando el conjunto de variables iniciales son todas numéricas se puede utilizar el análisis de componentes principales. Los métodos basados en clasificación no supervisada no consideran la respuesta deseada (o target), por lo que tienen la limitación de no dar buenos resultados, ya que puede ocurrir que la información que se considera importante para clasificar no lo es tanto para la representación del conjunto de datos en sí mismo.

Capítulo 5

Aplicación del Modelo Logístico.

5.1 Descripción de la Base de datos.

La lucha por mantener su clientela lleva a las empresas a estudiar el comportamiento de la misma, esto les permite tomar medidas estratégicas para permanecer en el mercado o aumentar su participación en el mismo. Entre otros aspectos, a las empresas les interesa determinar las causas por las cuales los clientes permanecen en una empresa, o toman la decisión de abandonarla, y poder predecir cuáles son los clientes que están por perder. Esto permite a la empresa tomar medidas que permitan retener al cliente y acciones más generales como políticas a seguir para evitar las causas.

Para esta aplicación, se utilizó la base de datos de los clientes de una empresa de servicios de telefonía celular, para estimar una regla que permita clasificarlos dentro de uno de estos dos grupos:

Grupo 1: clientes que permanecen en la empresa

Grupo 2: clientes que dejarán de utilizar los servicios de la empresa.

Se partió de una muestra de 9702 clientes, entre activos por una lado y cancelados o suspendidos por el otro, con la siguiente distribución en cada grupo:

Estado del cliente	Cantidad	Porcentaje
Activos	5712	58,9
Cancelado o suspendidos	3990	41,1
Total	9702	100

Tabla 5.1. Distribución de frecuencias del Estado del cliente

Se consideraron un conjunto de indicadores potencialmente explicativos de comportamiento de la clientela determinados por los expertos en Marketing de la empresa, entre los que figuran índices de consumo, índices de uso de servicios adicionales, índices de pago, tipo de aparato, zona del cliente, forma de adquisición del servicio, entre las más importantes. En total se consideraron 18 características, de las cuales 8 son numéricas, siendo las variables de análisis las siguientes:

Variables numéricas	**Descripción**	
1-Promedio consumo vida activa	Consumo promedio medido en minutos	
2-Nros de contactos a CS	Contactos con Servicios al cliente hasta 60 días después de la activación del servicio	
3-Antigüedad del cliente	Medida en días	
4-Tiempo que resta para finalizar del contrato	Medida en días	
5-Historia de pago	Índice calculado por el departamento de Marketing de la empresa (cuanto más cercano a 1 mejor será la historia de pago del cliente)	
6-Porcentaje de desalocación	Variable (expresada en porcentaje) definida por el departamento de Marketing de la empresa .	
7-Índice de relación entre consumo cuatrimestral y bimestral.	Cociente entre Promedio consumo cuatrimestral y promedio consumo bimestral	
8-Deuda total	Medida en pesos	
Variables categóricas	**Descripción**	Nro de Categorías[1]
1-Quejas en la cuenta	Cantidad de quejas en la cuenta	3
2-Duración del contrato	Depende del tipo de contrato	3
3-Región	Zona de procedencia del cliente	8
4-Forma de adquisición	Cómo fue adquirido el equipo, tipo de contrato utilizado	4
5-Tipo de cuenta	Cuenta de la empresa en la que se incluye el cliente considerado	4
6-Canal de distribución	Canal de ventas por el cual se adquirió el equipo	4
7-Estado del Agente	Estado actual del cliente	4
8-Modelo	Modelo del equipo	7
9-Débito automático	Adhesión a débito automático	2
10-Plazo de la deuda	Tipo de plazo.	2

Tabla 5.2. Variables predictoras en estudio

Se seleccionó aleatoriamente un 59,5 % de la muestra para el desarrollo de los modelos (muestra de entrenamiento), dejando el 40,5% de la muestra para la prueba u validación de los mismos (muestra test). Quedando distribuidos entre cancelados y activos como se muestra en la Tabla 5.3:

	Muestra de entrenamiento		Muestra test	
Estado del cliente	Cantidad	Porcentaje	Cantidad	Porcentaje
Activos	3421	59,3	2291	58,3
Cancelados o suspendidos	2350	40,7	1640	41,7
Total	5771	100	3931	100

Tabla 5.3. Distribución de frecuencias del estado del cliente para cada muestra.

5.2 Análisis y selección de las variables definitivas.

Dentro del análisis previo de las variables, se trató de determinar en primer lugar, si existe relación entre la variables definidas en el punto anterior y la condición de cancelado de los

[1] El detalle de las categorías se puede ver en Anexo 2 Tabla 1.

clientes. Para determinar esta relación se realizaron regresiones logísticas univariadas (lo que es equivalente a probar hipótesis de diferencias de medias para las continuas y pruebas Chi cuadrado para las variables categóricas). (Ver Tablas 2 a 5 del Anexo 2)

Dentro de las variables numéricas, la variable número de contactos promedio con CS no resultó significativa (a un nivel del 5%), y para la variable categórica canal de distribución no se rechaza la hipótesis de independencia por lo que ambas son eliminadas del conjunto de predictores.

Para identificar el mejor subconjunto de predictores de la variable respuesta se realizó una selección paso a paso (Forward Stepwise Selection), incorporando de a una las variables y comparando la verosimilitud entre los modelos resultantes. Se determinó como punto de corte máximo de probabilidad de la variable que se incorpora al modelo el valor 0,15 y como probabilidad mínima de corte para la variable que ser removida el valor 0,20. Del resultado de esta selección paso a paso, es posible eliminar las variables débito automático, quejas en la cuenta y deuda total. De todos modos, esta última variable se retiene en esta primera etapa, ya que la misma presenta un alto nivel clasificatorio en el análisis univariado. (Ver Tabla 6 del Anexo 2)

Para las variables continuas, se determinó el cumplimiento del supuesto de linealidad en el logit y para las categóricas la significatividad de cada categoría, en su defecto se propuso algún agrupamientos entre ellas analizando la diferencia de verosimilitud entre los modelos resultantes. De este análisis, resultó una primera selección de variables (Ver tabla 1 y resultados en el Anexo 3). En una etapa posterior se realizó un análisis más minucioso de las variables predictoras que permitió lograr una mejoría en la perfomance de los modelos y las que fueron consideradas definitivamente. Los cambios definitivos realizados en las variables se pueden ver en el Anexo 2 Tablas 7 y 8, siendo, en resumen, los siguientes:

- ✓ Se categorizaron las variables numéricas: Índice de relación entre consumo cuatrimestral y bimestral. y Deuda Total, ambas en tres categorías.
- ✓ Las siguientes variables categóricas fueron reagrupadas convenientemente.

Variables categóricas	**Nro de categorías definitivas**
Región	3
Forma de adquisición	3
Estado del Agente	3
Modelo	3

Tabla 5.4. Variables categóricas reagrupadas

- ✓ La variable plazo de la deuda mide lo mismo que la variable deuda total por lo que es eliminada del modelo.

Las variables predictoras definitivas para el modelo logístico propuesto son las siguientes:

Variables numéricas	
1-Promedio consumo vida activa	
2-Historia de pago	
3-Antigüedad del cliente	
4-Porcentaje de desalocación	
5-Tiempo que resta para finalizar del contrato	
Variables categóricas	Nro de categorías
6-Duración	3
7-Región	3
8 -Forma de adquisición	3
9- Tipo de cuenta	4
10-Estado del Agente	3
11-Modelo	3
12-Deuda total	3
13-Índice de relación entre consumo cuatrimestral y bimestral	3

Tabla 5.5. Variables predictoras definitivas

5.3. Resultados del Modelo Logístico

Para encontrar el modelo y sus resultados se utilizó el programa estadístico SPSS. Considerando el conjunto de variables seleccionados definitivamente, las variables categóricas fueron codificadas de acuerdo al detalle de la tabla 9 del Anexo 2.

Considerando como criterio de parada del proceso iterativo de estimación de los parámetros, un cambio en los mismos menor a 0,001, la estimación finaliza en la iteración número 6, en la Tabla 5.6 se muestran la log-verosimilitud obtenida en cada iteración.

Iteración	-2 log de la verosimilitud
Inicial	7800,388
1	4227,620
2	3679,699
3	3555,142
4	3542,941
5	3542,772
6	3542,772

Tabla 5.6 Historial de iteraciones

5.3.1 Evaluación del ajuste del modelo

El estadístico *G* para el modelo completo calculado como:

G = D (para el modelo completo) - D(modelo que incluye solo la constante)

asume el valor 4257,62 con distribución chi-cuadrado con 22 grados de libertad, siendo significativo a un nivel del 1%, con lo cual se puede concluir que al menos o quizás todos los coeficientes del modelo son diferentes de cero.

El valor de ajuste R^2 de Cox y Snell es igual a 0,52 y R^2 de Nagelkerke igual a 0,70, valores que suponen un buen ajuste bajo las consideraciones expuestas en el Capítulo 3. Para calcular el estadístico de Lemeshow y Hosmer, se construye la tabla de contingencia que surge de agrupar los valores estimados en deciles y compararlos con los valores reales dentro de cada grupo de clientes. La tabla que resulta es la siguiente:

	Activos		Cancelados		Total
	Observado	Esperado	Observado	Esperado	
1	553	565,14	24	11,86	577
2	544	548,88	33	28,12	577
3	536	532,73	41	44,27	577
4	502	512,49	75	64,51	577
5	490	481,08	87	95,92	577
6	438	420,33	139	156,67	577
7	282	276,10	295	300,90	577
8	71	72,13	506	504,87	577
9	5	11,05	572	565,95	577
10	0	1,07	578	576,93	578

Tabla 5.7. Tabla de contingencias para la prueba de Hosmer y Lemeshow

El valor del estadístico C resultante es de 24,207 el cual se distribuye X_8^2, este valor debe interpretarse con cuidado ya que el estadístico Chi Cuadrado depende del tamaño de la muestra y al tener un número grande de casos su valor puede ser significativo aún cuando el modelo ajuste correctamente. Sin embargo, del análisis de la tabla 5.7 es importante destacar que no se identifican deciles de riesgo, ya que los valores esperados son muy similares a la cantidad de clientes observados en cada grupo.

La tabla de clasificación (Tabla 5.8), considerando un punto de corte de 0,5, muestra como fueron clasificados los clientes por el modelo, tanto en la muestra de entrenamiento como en la muestra test. El porcentaje de clientes mal clasificados en la muestra de modelación fue del 11,8% el

cual aumenta al 12,20 % para la muestra test , la diferencia es del 0.4%, lo que indica un buen ajuste del modelo.

	Muestra entrenamiento			Muestra test		
	Pronosticado		% correcto			% correcto
Observado	Activo	Cancelado		Activo	Cancelado	
Activo	3240	181	94,71	2148	143	93,76
Cancelado	500	1850	78,72	335	1305	79,57
Porcentaje global			88,20	Porcentaje global		87,84

Tabla 5.8. Tabla de Clasificación

Del análisis de la Tabla 5.8, se puede destacar un mayor error de clasificación para el grupo de los clientes que están cancelados en el servicio, mientras que el grupo de clientes activos es mejor clasificado.

La empresa diseñará estrategias para evitar la cancelación del servicio sólo para aquellos clientes que el modelo clasifique como cancelados, por lo que no se tendrá en cuenta los clientes clasificados como activos. Entonces el error más riesgoso es clasificar como activos clientes que en realidad cancelarán el servicio, siendo en este modelo el error más alto. Para este conjunto de datos, es posible mejorar este error corriendo el punto de corte a un valor superior del 0,5, donde los clientes cancelados son mejor clasificados aunque aumente el error de clasificación en los activos, lo que puede observarse en el gráfico 1 del Anexo 2.

5.3.2. Análisis de los coeficientes del modelo

En la Tabla 5.9 se muestran para cada variable[2]: los coeficientes estimados, el P-value que surge del estadístico de Wald , el cociente de odds (exp(w)) y un intervalo de confianza para cada coeficiente a un nivel de significación de 0,05. Del análisis de la tabla, los coeficientes de las variables (w), resultan todos significativos. Analizando los mismos en términos de la chance que un cliente esté cancelado (exp(w) = cociente de odds), y considerando su intervalo de confianza se pueden determinar las causas por las que un cliente es considerado cancelado.

[2] Por simplicidad se utilizó la nomenclatura definida como código que figura en la Tabla 7 del Anexo2.

	w	E.T.	Wald	gl	Sig.	Exp(w)	I.C. 95.0% para EXP(w)	
							Inferior	Superior
D_EDAD	-0,01	0,00	229,60	1	0,000	0,993	0,993	0,994
D_FINCON	0,00	0,00	8,26	1	0,004	0,999	0,998	1,000
HIST_P	0,71	0,04	352,03	1	0,000	2,034	1,889	2,191
PORC_DES	0,32	0,08	17,72	1	0,000	1,377	1,186	1,598
PROM_ACT	0,00	0,00	4,68	1	0,030	1,001	1,000	1,001
D_DURAC			64,07	2	0,000			
D_DURAC(1)	-0,84	0,27	9,54	1	0,002	0,430	0,252	0,735
D_DURAC(2)	-1,47	0,19	62,03	1	0,000	0,229	0,159	0,331
REGION_3			8,01	2	0,018			
REGION_3(1)	0,19	0,10	3,58	1	0,058	1,214	0,993	1,483
REGION_3(2)	0,49	0,18	7,24	1	0,007	1,633	1,142	2,335
ADQUISI1			40,00	2	0,000			
ADQUISI1(1)	-0,22	0,13	3,09	1	0,079	0,799	0,623	1,026
ADQUISI1(2)	0,50	0,12	18,74	1	0,000	1,650	1,315	2,070
RTIP_CTA			55,17	3	0,000			
RTIP_CTA(1)	-0,81	0,42	3,72	1	0,054	0,444	0,195	1,013
RTIP_CTA(2)	-0,26	0,09	7,78	1	0,005	0,771	0,642	0,926
RTIP_CTA(3)	-2,08	0,29	51,91	1	0,000	0,125	0,071	0,220
REST_AG1			57,03	2	0,000			
REST_AG1(1)	0,78	0,10	56,78	1	0,000	2,176	1,777	2,663
REST_AG1(2)	0,34	0,14	6,43	1	0,011	1,409	1,081	1,837
MMODELO			18,60	2	0,000			
MMODELO(1)	0,39	0,10	13,96	1	0,000	1,475	1,203	1,809
MMODELO(2)	-0,11	0,12	0,72	1	0,395	0,899	0,704	1,148
TDEUDA			127,75	2	0,000			
TDEUDA(1)	-0,71	0,12	33,50	1	0,000	0,492	0,387	0,626
TDEUDA(2)	1,05	0,15	47,46	1	0,000	2,871	2,127	3,875
TPROM4_2			260,89	2	0,000			
TPROM4_2(1)	0,56	0,10	29,58	1	0,000	1,758	1,434	2,154
TPROM4_2(2)	1,85	0,11	260,73	1	0,000	6,368	5,086	7,972
Constante	0,93	0,44	4,55	1	0,033	2,539		

Tabla 5.9. Variables en la ecuación

Del análisis de la tabla anterior se puede concluir que:

- ✓ Las variables Promedio de consumo de vida activa y Tiempo que resta para finalizar el contrato, no afectan la condición de estar cancelado en el servicio.
- ✓ Las variables: Antigüedad del cliente, Tipo de Cuenta y Duración del Contrato no aumentan la chance de estar cancelado en el servicio. Por cada día de antigüedad del cliente la chance de estar cancelado disminuye en un 7%. A medida que aumenta la duración del contrato disminuye la chance de estar cancelado y las cuentas

Personales y Top tienen menos chance estar cancelados respecto a las cuentas Negocios (22,9% y 87,5% respectivamente).

- ✓ El resto de las variables del modelo, o alguna de sus categorías aumentan la posibilidad de que un cliente esté cancelado. Considerándolas desde la más riesgosa:
    - Índice de relación entre consumo cuatrimestral y bimestral (categórica) a medida que aumenta el índice que resulta del cociente del promedio de consumo de 4 meses sobre el promedio de consumo de 2 meses, aumenta la chance de estar cancelado siendo el 75,8% más para los clientes que se encuentran en el segundo intervalo y 6 veces más para los que se encuentran en el tercer intervalo(para valores del índice superiores a 500).
    - Deuda Total (categórica: Considerando como referencia a los clientes sin deuda, los que tienen deuda hasta $157, tienen un 50% menos posibilidad de estar cancelados, pero los que tienen deudas mayores a ese valor tienen un 187% más de riesgo de que se cancele su servicio.
    - Historia de pago, es una variable medida como un índice de 1 a 7 donde 1 se considera pago a término y 7 los clientes que tienen deuda de 180 días o más. Por cada aumento unitario del índice, aumenta 2,034 veces la chance de estar cancelado.
    - Estado del agente: los que están dados de baja tienen 2,176 más posibilidad de estar cancelados que los activos y para los que están tramitando la baja es de 1,409.
    - Porcentaje de desalocación, a cada incremento de un 1% de la misma, aumenta en 37% la posibilidad de cancelación del servicio.
    - Modelo: la marca Nokia tiene un 47.5% más chance de estar cancelado que la marca de referencia, considerando el resto de las marcas dentro de la categoría otros, la misma resulta no significativa.
    - Adquisición: hay un 65% más de posibilidad de cancelación en los clientes con aparatos propios que aquellos adquiridos por comodato, mientras que en los contratos de leasing y los aparatos distribuidos por la empresa disminuye la posibilidad de cancelación en un 20%.

- Considerando la región de Bs. As.,Sta Fe y La Pampa, como referencia, la región que incluye a Cuyo, el Litoral, Mediterráneo y NOAR no resulta significativa, pero la Patagónica tiene 1.6 veces más probabilidad de estar cancelado que la región de referencia.

5.3.3. Métodos de Diagnóstico del modelo

Un análisis pormenorizado de los residuos en la muestra de entrenamiento, permite detectar aquellos individuos que no son ajustados adecuadamente por el modelo, y aquellos que ejercen una influencia significativa en la estimación del mismo. Se calcularon para cada individuo las medidas definidas en el capítulo 3 tales como: residuo $y_i - k_i$, Pearson Residual (Z_i) y Deviance Residual (d_i), Valor de influencia (h_i), distancia de Cook (D_i), efecto en el Chi cuadrado de Pearson (ΔX_i^2), efecto en la Deviance (ΔD_i), y el efecto de eliminar cada individuo sobre cada parámetro $(dw_j^i$ para j = 0,1,2,.....,p).

Estas medidas fueron analizadas según los intervalos de probabilidades como se indicó en el cuadro de la figura 3.2 y a través de los gráficos 2 a 5 del anexo 2 identificando individuos con altos valores en sus residuos y que ejercen mucha influencia en la estimación del modelo. En la Tabla 5.10 se resumen las medidas calculadas para los individuos señalados.

Nro de caso	Grupo Real	Probabilidad Estimada	Grupo asignado	Distancia de CooK	Valor de influencia	Residuo	Pearson Residual	Deviance Residual
1055	0	0,962	1	0,043	0,002	-0,962	-5,04089	-2,559
1973	0	0,609	1	0,290	0,158	-0,609	-1,24907	-1,371
2306	0	0,792	1	0,171	0,043	-0,792	-1,95195	-1,772
2691	0	0,976	1	0,078	0,002	-0,976	-6,37202	-2,731
6619	0	0,966	1	0,438	0,015	-0,966	-5,31354	-2,598
7377	0	0,988	1	0,125	0,002	-0,988	-8,99365	-2,968
8686	0	0,878	1	0,325	0,043	-0,878	-2,68706	-2,053
9128	0	0,965	1	0,168	0,006	-0,965	-5,28354	-2,594
1428	1	0,077	0	0,165	0,014	0,923	3,46150	2,264
3074	1	0,002	0	0,263	0,000	0,998	24,70413	3,582
3431	1	0,011	0	0,113	0,001	0,989	9,49614	3,004
6875	1	0,016	0	0,052	0,001	0,984	7,80395	2,872
7712	1	0,010	0	0,040	0,000	0,990	10,08376	3,044
9355	1	0,231	0	0,142	0,041	0,769	1,82622	1,713

Tabla 5.10. Individuos con medidas de diagnóstico atípicos.

Se eliminaron estos casos logrando una leve mejoría en las medidas de ajuste, que no repercutió en el conjunto de datos de testeo. En la tabla de clasificación del este ajuste,

Tabla 5.11 puede observarse que el porcentaje global de clientes clasificados correctamente en la muestra test no cambió.

	Muestra entrenamiento			Muestra test		
	Estimado		% clasificación correcta	Estimado		%clasificación correcta
Observado	Activo	Cancelado		Activo	Cancelado	
Activo	3234	177	94,81	2143	148	93,54
Cancelado	492	1850	78,99	330	1309	79,87
Porcentaje global			88,37	Porcentaje global		87,84

Tabla 5.11. Tabla de Clasificación

Dado el tamaño de la muestra de modelación es insignificante la cantidad de valores atípicos que se identifican, pero estos clientes pueden ser considerados para su análisis por el departamento de marketing de la empresa.

5.3.4. Conclusiones

Un análisis más detallado del las variables permitió una mejora del 0,7% en el porcentaje de clasificación correcta de la muestra test.

El modelo definitivo permite clasificar correctamente al 87,84 % de los clientes en la etapa de generalización del modelo. Por otro lado, a partir de los parámetros es posible identificar los factores que hacen que un cliente sea considerado cancelado. Los factores que determinan una mayor probabilidad que el cliente esté cancelado son: la relación entre consumo cuatrimestral y bimestral, la historia de pago, el porcentaje de desalocación que posean, los clientes con deuda total mayor a $157, los que están dados de baja o tramitando la baja, que tenga aparato propio o que provenga de la región patagónica.

Capítulo 6

Aplicación de los modelos de Redes Neuronales.

6.1. Aspectos generales

Utilizando la misma base de datos descripta en el punto 5.1. se debe realizar una selección previa de las variables. Las redes neuronales no tienen una metodología propia de selección de variables para reducir el espacio de variables iniciales del modelo, en este caso se utiliza la selección realizada en el punto 5.2. Partiendo de las variables seleccionadas, para la construcción de un modelo de red fue necesario:

1- Determinar una configuración adecuada del modelo.
2- Entrenar el modelo de red definido.
3- Utilizar el entrenamiento anterior con la muestra de testeo para determinar una medida de desempeño, realizando varias simulaciones.

Para determinar su configuración, se modelaron perceptrones multicapa, probando diferentes topologías (número capas intermedias, y número de variables en cada capa), y diferentes funciones de activación. Como la variable respuesta asume valores 0 y 1, la función de activación de las variables de salida será la logística cuyo intervalo de variación está entre cero y uno, el cual se asume como un valor de probabilidad, determinando que si el valor de salida es menor a 0,5 le corresponde el valor 1, y 0 en el otro caso. De esta manera, comparando este valor con el verdadero grupo de pertenencia de cada cliente (k), se determina el porcentaje de error de clasificación utilizado como medida de resultado del modelo.

Para las corridas del algoritmo se utilizaron rutinas del software MATLAB 7.0, las que fueron adaptadas en caso que fuera necesario. En primer lugar se definieron los valores iniciales a los parámetros de una determinada estructura de red. Una vez que los parámetros han sino inicializados, la red está lista para ser entrenada, en este caso para

clasificar individuos. Para el entrenamiento se utilizó en mismo conjunto de ejemplos (características de los individuos y clasificación correcta) considerados en el modelo logístico, pero en este caso fue dividida en muestra de entrenamiento y muestra de validación. La muestra de entrenamiento se utilizó para calcular los gradientes y actualizar los parámetros del modelo, mientras que el conjunto de validación permite calcular una medida de error que es monitoreada durante el proceso de entrenamiento de la red. Normalmente durante la primera etapa del entrenamiento el error del conjunto de validación decrece de la misma forma que lo hace el error de la muestra de entrenamiento. Sin embargo, cuando el modelo comienza a sobre ajustar los datos, el error de validación comienza a subir, y cuando esto ocurre se determina un número especificado de iteraciones como un criterio de parada del entrenamiento. La muestra de entrenamiento considerada en esta aplicación fue dividida de la siguiente manera:

		Muestra entrenamiento	Muestra validación	Total
Activos	cantidad	2692	729	3421
	%	58,29	63,23	0,59
Cancelados	cantidad	1926	424	2350
	%	41,71	36,77	0,41
Total	cantidad	4618	1153	5771

Tabla 6.1. División de la muestra en entrenamiento y validación

La muestra de testeo no se considera para el entrenamiento, se utiliza para comparar la perfomance de los modelos. Para ir monitoreando el entrenamiento de la red, es útil utilizar un gráfico que muestre comportamiento de los errores de clasificación en la muestra test y en la muestra de entrenamiento en cada período.

El entrenamiento de red puede hacerse más eficiente realizando un procesamiento previo de las variables y el valor de clasificación real. Por un lado reduciendo la dimensión del vector de variables, considerando la reducción lograda en el punto 5.2. Para evitar problemas numéricos con las distintas escalas de las variables seleccionadas, las numéricas pueden normalizarse con media cero y desviación uno, según la expresión 4.20.

Para el proceso de aprendizaje fue utilizado el algoritmo de retropropagación de los errores. El desempeño del algoritmo puede mejorarse si se permite que la tasa de aprendizaje cambie durante el proceso de entrenamiento de la red, como se expuso en el punto 4.1.3. Un parámetro de aprendizaje adaptativo procurará mantener la tasa de

aprendizaje tan grande como sea posible mientras el aprendizaje se mantenga estable. De esta manera la tasa de aprendizaje es sensible a la complejidad de la superficie del error. Se definen una tasa de aprendizaje η, un factor de incremento de la tasa de aprendizaje η_{aum} y un factor de disminución η_{dism}, Proponiendo un valor del error cuadrático medio como objetivo y una razón máxima de incremento del mismo durante el proceso de aprendizaje. En un período cualquiera del entrenamiento, si el nuevo valor de error, excede al del período anterior en más de la razón de aumento máxima establecida, los nuevos parámetros son desechados y la nueva tasa de aprendizaje es disminuida en la tasa η_{dism}. Si el nuevo error es menor que el valor del período anterior la tasa de aprendizaje se incrementa con el factor η_{aum} . Además se consideró el uso de una tasa de momento. Los parámetros definidos en el entrenamiento para esta aplicación se muestran en la Tabla 6.2

Máximo de períodos	1300
Error cuadrático medio objetivo	0,05
Tasa de aprendizaje	0,01
Razón de incremento de la tasa de aprendizaje	1,05
Razón de disminución de la tasa de aprendizaje	0,7
Número máximo de errores en el conjunto de validación	500
Tasa de momento	0,9
Mínimo que puede asumir el gradiente	$1e^{-10}$
Incremento máximo del error cuadrático medio	1.04
Tiempo máximo de entrenamiento en segundos	infinito

Tabla 6.2. Parámetros definidos en el entrenamiento.

El proceso de aprendizaje finaliza cuando se cumplen algunas de las siguientes condiciones:

- Se alcanza un número máximo de períodos
- La función gradiente asume un valor por debajo de un mínimo establecido.
- El error cuadrático medio del conjunto de entrenamiento llega al valor del error cuadrático medio propuesto como objetivo.
- Usando un conjunto de validación, los errores de este conjunto superan a un máximo de fallas definido.

Para determinar la estabilidad en los resultados, se realizaron distintas corridas partiendo de diferentes valores iniciales de la matriz de parámetros *W*; estos valores fueron calculados utilizando el algoritmo de Nguyen-Widrow definido en el punto 4.1.4. Se

determinaron como mínimo 30 pruebas para cada modelo calculando los valores promedios de los resultados obtenidos y sus correspondientes desvíos.

Fueron considerados diferentes modelos de redes neuronales. En el Anexo 3 se muestran los resultados obtenidos con la primera selección de variables, y a continuación se detallan la estructura de cada modelo y los resultados obtenidos.

6.2 Resultados obtenidos con el conjunto de variables definitivas.

Considerando el conjunto de variables definitivas que se detallan en la tabla 7 del Anexo 2, y todos los aspectos detallados en la sección 6.1 con los parámetros establecidos en la tabla 6.2., se entrenaron distintos modelos de redes para poder determinar el modelo que clasifica mejor al conjunto de clientes de la empresa.

6.2.1 Perceptron multicapa con una capa intermedia.

La estructura del modelo se puede representar gráficamente en general como sigue:

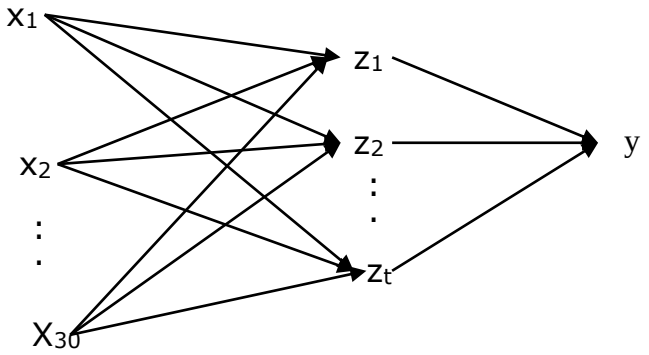


Figura 6.3 Perceptron multicapa con una capa intermedia

Como se mencionó en el capítulo 4, una de las maneras de controlar la complejidad de la función discriminante es variando la cantidad de nodos de la capa intermedia. Además se probaron distintas funciones de activación y en algunos casos se cambió el parámetro de aprendizaje. Para cada modelo se realizaron diferentes pruebas partiendo de parámetros iniciales diferentes, los resultados promedios que se obtuvieron de las pruebas realizadas se muestran en la Tabla 6.4.

Capa Intermedia				Tasa de error de entrenamiento		Tasa de error muestra test		
Función de activación	Número de variables	Parámetro de aprendizaje	Pruebas	Promedio	Desvío	Promedio	Desvío	Mínimo
Tangente	2	eta0.01	40	0,1187	0,0019	0,1277	0,0035	0,1163
Lineal	2	eta0.01	60	0,1183	0,0023	0,1226	0,0024	0,1193
Lineal	2	eta 0.00004	60	0,1188	0,0028	0,1237	0,0034	0,1185
Logística	2	eta 0.00004	30	0,1180	0,0053	0,1268	0,0078	0,1188
Tangente	3	eta0.01	60	0,1108	0,0052	0,1261	0,0031	0,1203
Lineal	3	eta0.01	90	0,1188	0,0028	0,1238	0,0028	0,1191
Lineal	3	eta 0.00004	60	0,1174	0,0023	0,1316	0,0079	0,1185
Logística	3	eta0.01	60	0,1159	0,0047	0,1299	0,0073	0,1201
Tangente	4	eta 0.01	130	0,1107	0,0057	0,1267	0,0029	0,1198
Lineal	4	eta 0.01	90	0,1184	0,0028	0,1239	0,0028	0,1188
Logística	4	eta 0.01	70	0,1144	0,0060	0,1322	0,0063	0,1198
Logística	4	eta 0.00004	30	0,1175	0,0049	0,1326	0,0074	0,1201
Tangente	5	eta 0.01	60	0,1052	0,0051	0,1265	0,0022	0,1213
Lineal	5	eta 0.01	60	0,1188	0,0027	0,1237	0,0022	0,1196
Logística	5	eta 0.01	60	0,1153	0,0054	0,1335	0,0063	0,1206
Tangente	6	eta 0.01	30	0,1043	0,0040	0,1269	0,0026	0,1201
Lineal	6	eta 0.01	30	0,1191	0,0029	0,1234	0,0024	0,1198
Logística	6	eta 0.01	30	0,1146	0,0055	0,1330	0,0052	0,1234
Logística	10	eta0.01	20	0,1150	0,0036	0,1355	0,0046	0,1254
Tangente	10	eta0.01	10	0,0986	0,0054	0,1264	0,0023	0,1239
Lineal	10	eta0.01	10	0,1203	0,0039	0,1241	0,0032	0,1208
Logística	15	eta0.01	10	0,1165	0,0027	0,1391	0,0045	0,1302
Tangente	15	eta0.01	10	0,0964	0,0033	0,1287	0,0030	0,1247
Lineal	15	eta0.01	10	0,1197	0,0029	0,1251	0,0032	0,1196

Tabla 6.4. Resultados con una capa intermedia

En los modelos con más variables intermedias se realizaron menos de 30 pruebas debido al tiempo de procesamiento, y considerando que no había mucha variabilidad en los resultados ya que en todos los modelos propuestos considerando todas las pruebas realizadas la desviación estándar del error en el porcentaje de clasificación no supera en ningún caso el 0,008.

Si los resultados se ordenan de menor a mayor según el promedio de la tasa de error en la muestra de entrenamiento, como se muestra en la Tabla 6.5 se puede concluir que el error de clasificación disminuye a medida que se incorporan variables intermedias. Pero esta mejora es muy lenta si se considera que por ejemplo entre una red con función de activación tangente con 15 variables intermedias y otra con 4 variables intermedias, la mejora en la tasa de error es del 1,55% a costa de agregar 341 parámetros al modelo.

Capa Intermedia				Tasa de error de entrenamiento		Tasa de error muestra test		
Función de activación	Número de variables	Parámetro de aprendizaje	Pruebas	Promedio	Desvío	Promedio	Desvío	Mínimo
Tangente	15	eta0.01	10	0,0964	0,0033	0,1287	0,0030	0,1247
Tangente	10	eta0.01	10	0,0986	0,0054	0,1264	0,0023	0,1239
Tangente	6	eta 0.01	30	0,1043	0,0040	0,1269	0,0026	0,1201
Tangente	5	eta 0.01	60	0,1052	0,0051	0,1265	0,0022	0,1213
Tangente	4	eta 0.01	130	0,1107	0,0057	0,1267	0,0029	0,1198
Tangente	3	eta0.01	60	0,1108	0,0052	0,1261	0,0031	0,1203
Logística	4	eta 0.01	70	0,1144	0,0060	0,1322	0,0063	0,1198
Logística	6	eta 0.01	30	0,1146	0,0055	0,1330	0,0052	0,1234
Logística	10	eta0.01	20	0,1150	0,0036	0,1355	0,0046	0,1254
Logística	5	eta 0.01	60	0,1153	0,0054	0,1335	0,0063	0,1206
Logística	3	eta0.01	60	0,1159	0,0047	0,1299	0,0073	0,1201
Logística	15	eta0.01	10	0,1165	0,0027	0,1391	0,0045	0,1302
Lineal	3	eta 0.00004	60	0,1174	0,0023	0,1316	0,0079	0,1185
Logística	4	eta 0.00004	30	0,1175	0,0049	0,1326	0,0074	0,1201
Logística	2	eta 0.00004	30	0,1180	0,0053	0,1268	0,0078	0,1188
Lineal	2	eta0.01	60	0,1183	0,0023	0,1226	0,0024	0,1193
Lineal	4	eta 0.01	90	0,1184	0,0028	0,1239	0,0028	0,1188
Tangente	2	eta0.01	40	0,1187	0,0019	0,1277	0,0035	0,1163
Lineal	5	eta 0.01	60	0,1188	0,0027	0,1237	0,0022	0,1196
Lineal	2	eta 0.00004	60	0,1188	0,0028	0,1237	0,0034	0,1185
Lineal	3	eta0.01	90	0,1188	0,0028	0,1238	0,0028	0,1191
Lineal	6	eta 0.01	30	0,1191	0,0029	0,1234	0,0024	0,1198
Lineal	15	eta0.01	10	0,1197	0,0029	0,1251	0,0032	0,1196
Lineal	10	eta0.01	10	0,1203	0,0039	0,1241	0,0032	0,1208

Tabla 6.5. Resultados con una capa intermedia (ordenados según tasa de error en la muestra de entrenamiento)

En esta aplicación el modelo interesa a los fines predictivos, lo que importa es que el mismo pueda generalizar correctamente, por lo cual interesa analizar la tasa de error de clasificación en la muestra de testeo. Ordenando los resultados según los valores promedios de la tasa de error de clasificación en la muestra de testeo (Tabla 6.6), se observa que los modelos con pocas variables en la capa intermedia y función de activación lineal o tangente son los que presentan menor tasa de error.

Capa Intermedia				Tasa de error de entrenamiento		Tasa de error muestra test		
Función de activación	Número de variables	Parámetro de aprendizaje	Pruebas	Promedio	Desvío	Promedio	Desvío	Mínimo
Lineal	2	eta0.01	60	0,1183	0,0023	0,1226	0,0024	0,1193
Lineal	6	eta 0.01	30	0,1191	0,0029	0,1234	0,0024	0,1198
Lineal	2	eta 0.00004	60	0,1188	0,0028	0,1237	0,0034	0,1185
Lineal	5	eta 0.01	60	0,1188	0,0027	0,1237	0,0022	0,1196
Lineal	3	eta0.01	90	0,1188	0,0028	0,1238	0,0028	0,1191
Lineal	4	eta 0.01	90	0,1184	0,0028	0,1239	0,0028	0,1188
Lineal	10	eta0.01	10	0,1203	0,0039	0,1241	0,0032	0,1208
Lineal	15	eta0.01	10	0,1197	0,0029	0,1251	0,0032	0,1196
Tangente	3	eta0.01	60	0,1108	0,0052	0,1261	0,0031	0,1203
Tangente	10	eta0.01	10	0,0986	0,0054	0,1264	0,0023	0,1239
Tangente	5	eta 0.01	60	0,1052	0,0051	0,1265	0,0022	0,1213
Tangente	4	eta 0.01	130	0,1107	0,0057	0,1267	0,0029	0,1198
Logística	2	eta 0.00004	30	0,1180	0,0053	0,1268	0,0078	0,1188
Tangente	6	eta 0.01	30	0,1043	0,0040	0,1269	0,0026	0,1201
Tangente	2	eta0.01	40	0,1187	0,0019	0,1277	0,0035	0,1163
Tangente	15	eta0.01	10	0,0964	0,0033	0,1287	0,0030	0,1247
Logística	3	eta0.01	60	0,1159	0,0047	0,1299	0,0073	0,1201
Lineal	3	eta 0.00004	60	0,1174	0,0023	0,1316	0,0079	0,1185
Logística	4	eta 0.01	70	0,1144	0,0060	0,1322	0,0063	0,1198
Logística	4	eta 0.00004	30	0,1175	0,0049	0,1326	0,0074	0,1201
Logística	6	eta 0.01	30	0,1146	0,0055	0,1330	0,0052	0,1234
Logística	5	eta 0.01	60	0,1153	0,0054	0,1335	0,0063	0,1206
Logística	10	eta0.01	20	0,1150	0,0036	0,1355	0,0046	0,1254
Logística	15	eta0.01	10	0,1165	0,0027	0,1391	0,0045	0,1302

Tabla 6.6. Resultados con una capa intermedia (ordenados según tasa de error en la muestra test)

6.2.2 Perceptron multicapa con dos capas intermedias.

Se probaron modelos más complejos para verificar si los mismos mejoran los resultados obtenidos con una capa intermedia, considerando entonces los parámetros indicados inicialmente en la Tabla 6.2, se agregó una capa intermedia. Siendo la siguiente la estructura de la red considerada:

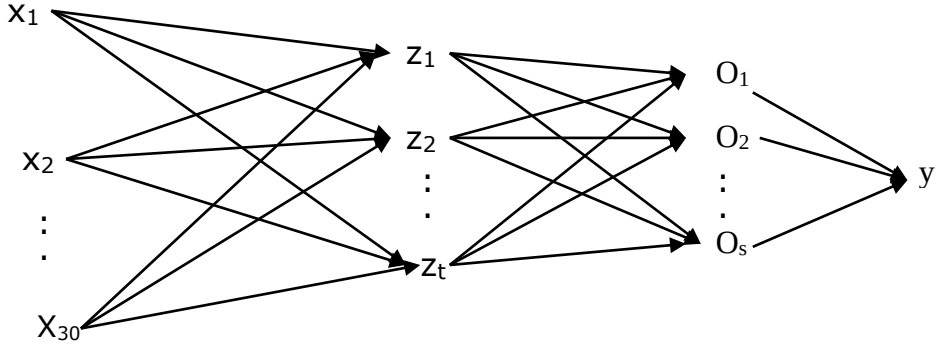


Figura 6.7. Perceptron multicapa con dos capas intermedias.

Se probaron diferentes modelos variando la cantidad de variables y las funciones de activación en cada una de las capas intermedias. Para determinar las funciones de activación se tuvieron en cuenta los resultados obtenidos con una capa intermedia; en los que con funciones de activación lineal y tangente, se obtuvieron las menores tasa de error de clasificación para la muestra de testeo. Al igual que en los modelos del punto anterior se realizaron varias pruebas cambiando los parámetros iniciales, siendo los resultados promedios obtenidos los que se muestran en la Tabla 6.8.

Primera capa intermedia		Segunda capa intermedia		Tasa de error muestra entrenamiento		Tasa de error muestra test		
Función de activación	Nro de variables	Función de activación	Nro de variables	Promedio	Desvío	Promedio	Desvío	Mínimo
Lineal	3	Tangente	2	0,1194	0,0043	0,1282	0,0048	0,1221
Tangente	3	Lineal	2	0,1212	0,0069	0,1312	0,0056	0,1241
Tangente	3	Tangente	5	0,1132	0,0053	0,1267	0,0031	0,1208
Tangente	3	Tangente	7	0,1153	0,0054	0,1270	0,0054	0,1196
Tangente	2	Tangente	5	0,1148	0,0045	0,1245	0,0061	0,1191
Tangente	2	Tangente	8	0,1153	0,0045	0,1244	0,0039	0,1198
Lineal	3	Tangente	4	0,1126	0,0071	0,1262	0,0037	0,1203
Lineal	2	Tangente	4	0,1140	0,0038	0,1241	0,0036	0,1193
Lineal	2	Lineal	3	0,1205	0,0046	0,1256	0,0047	0,1196
Lineal	2	Lineal	4	0,1180	0,0040	0,1230	0,0033	0,1188

Tabla 6.8. Resultados de diferentes multiperceptrones con dos capas intermedias

Analizando la tabla anterior se puede verificar que un modelo más complejo que resulta al incorporar una capa intermedia adicional, no logra disminuir la tasa de error en la muesta test. Esto significa que modelos con más de dos capas intermedias no mejoran la clasificación de los clientes de la empresa.

6.2.3 Perceptron multicapa con una capa intermedia considerando variables de riesgo.

Se consideró la posibilidad de lograr la misma medida de perfomance con un menor número de variables, considerando aquellas que en el discriminante logístico resultaron con más chance de que un cliente estuviera cancelado, estas son (ver Tabla 6.9):

Variables numéricas	
-Historia de pago	
-Porcentaje de desalocación	
Variables categóricas	Nro de categorías
-Estado del Agente	3
-Indice de relación entre consumo cuatrimestral y bimestral	3

Tabla 6.9. Variables predictoras consideradas en el modelo

El modelo resultante será:

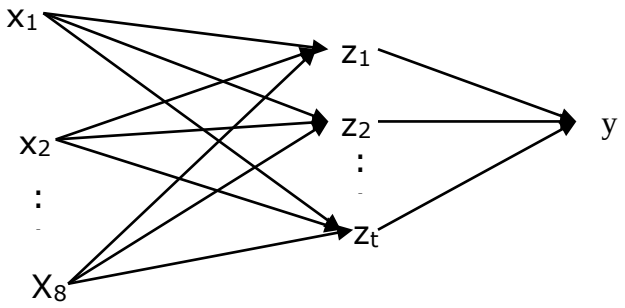


Figura 6.10. Percepton multicapa con una capa intermedia

En la tabla 6.11 se muestran los resultados obtenidos, observando que la reducción de variables no permite encontrar una función discriminante que logre la misma separación de los grupos conseguida con todas las variables consideradas inicialmente.

Capa Intermedia		Tasa de error de entrenamiento		Tasa de error muestra test	
Función de activación	Número de variables	Promedio	Desvío	Promedio	Desvío
Tangente	2	0,1595	0,0019	0,1664	0,0018
Lineal	2	0,1670	0,0068	0,1713	0,0060
Tangente	3	0,1594	0,0027	0,1666	0,0019
Lineal	3	0,1647	0,0054	0,1706	0,0053
Logística	3	0,1628	0,0044	0,1697	0,0059
Tangente	4	0,1587	0,0019	0,1664	0,0014
Lineal	4	0,1654	0,0051	0,1707	0,0043
Tangente	10	0,1584	0,0025	0,1679	0,0024
Lineal	10	0,1648	0,0055	0,1690	0,0048

Tabla 6.11. Resultados con una capa intermedia (30 pruebas para cada modelo)

La reducción de variables aumenta la tasa de error tanto en la muestra de entrenamiento como en la muestra test, lo que indica que el conjunto de variables seleccionado no permite encontrar una función discriminante que clasifique correctamente.

6.2.4 Modelo de redes neuronales seleccionado.

Analizando los modelos entrenados, se consideran aquellos con una capa intermedia, ya que los modelos más complejos (con dos capas intermedias o más) no mejoran el porcentaje de clientes bien clasificados. Dentro de los modelos de red con una capa oculta, se consideró uno dos variables intermedias y función de activación tangente. Este modelo sin dar la menor tasa de error promedio es el que permitió encontrar dentro de las 40 pruebas realizadas la menor tasa de error de clasificación en la muestra de testeo (0,1163), por lo que es el que permite una mejor generalización de los resultados.

El modelo puede representarse de la siguiente manera:

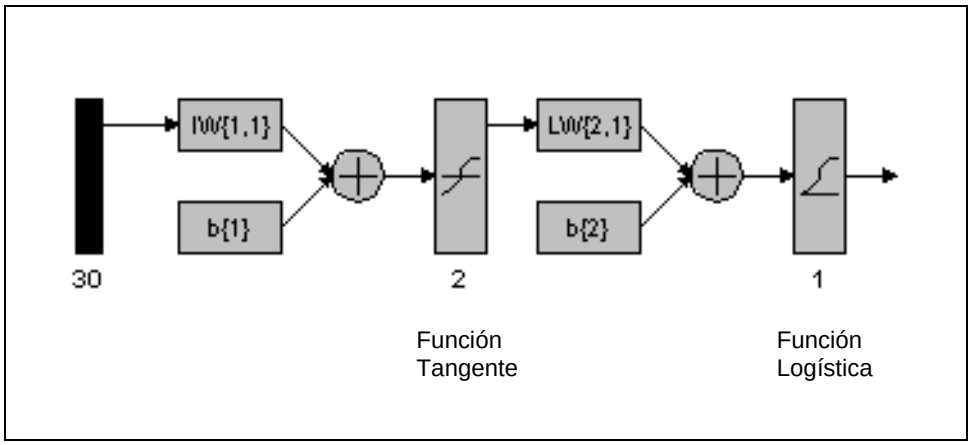
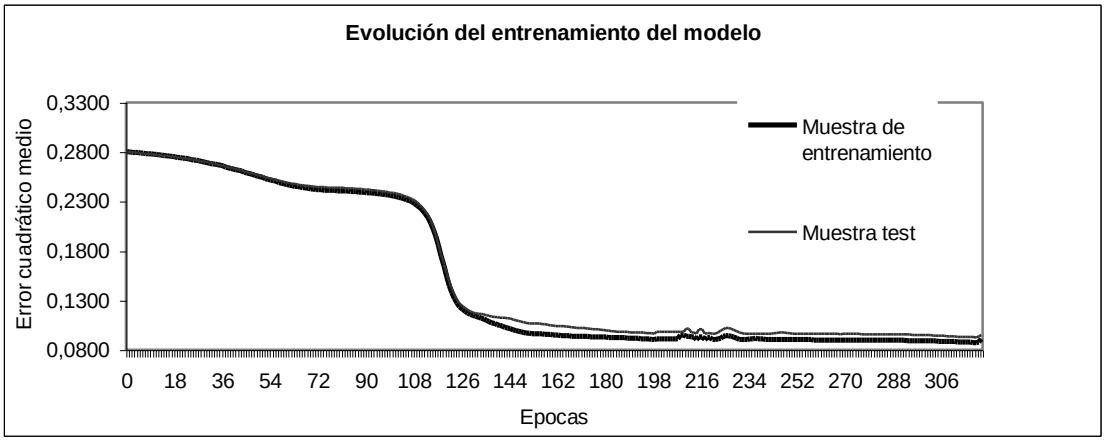


Figura 6.12. Modelo seleccionado

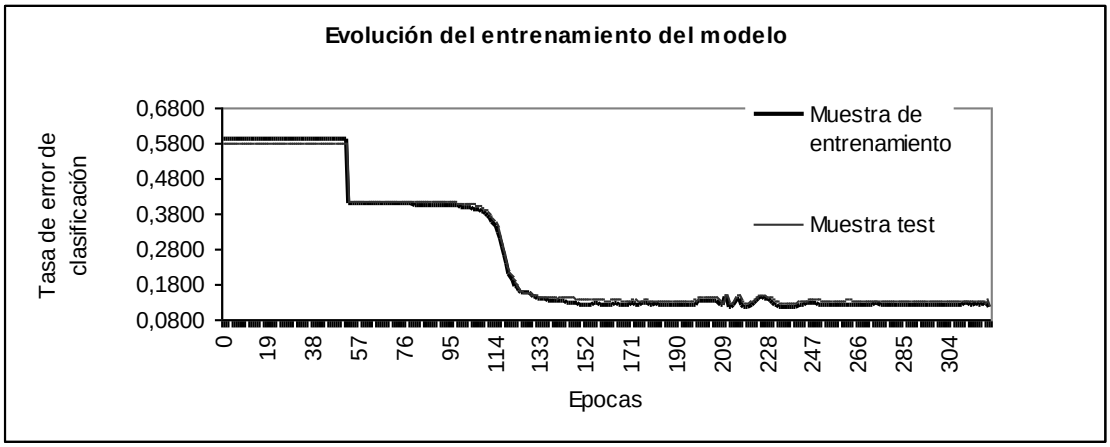
En la figura anterior *IW* representa el vector de parámetros de las conexiones entre las variables de entrada y la capa intermedia y *LW* los parámetros de las conexiones entre la capa intermedia y la capa de salida.

La red fue entrenada utilizando el algoritmo de retropropagación con tasa de aprendizaje variable y tasa de momento con los valores de la Tabla 6.2.

En la Figura 6.13 (a) se muestra el comportamiento del error cuadrático medio durante el entrenamiento de la red, mientras que en la Figura 6.13 (b) se muestra la evolución de la tasa de error de clasificación en la muestra de entrenamiento y la muestra test.



(a)



(b)

Figura 6.13. Evolución de las tasas de error durante el entrenamiento de la red

Observando en las primeras 100 corridas de la Figura 6.13 (a) y (b), el algoritmo presenta un comportamiento diferente, ya que el mismo va actualizando los parámetros de tal manera que el error cuadrático medio va disminuyendo pero esto no significa que la función permita una mejora en la clasificación de los clientes. Alrededor de las 100 corridas el error cuadrático medio y la tasa de error en la clasificación se comportan de manera similar logrando acercarse a una mejor función discriminante.

En total fueron calculados 65 parámetros, los cuales no son directamente interpretables , por lo que solo tienen interés en relación a la reconstrucción del modelo. (Tabla 11 del Anexo 2).

Para analizar como han sido clasificados los clientes tanto en la muestra de entrenamiento como en la muestra test, se elabora una tabla de clasificación como la realizada en el modelo logística (Tabla 5.8), considerando el mismo punto de corte .

	Muestra de modelación			Muestra test		
	Pronosticado		% correcto			% correcto
Observado	Activo	Cancelado		Activo	Cancelado	
Activo	3238	183	94,65%	2165	126	94,50%
Cancelado	493	1857	79,02%	331	1309	79,82%
Porcentaje global			88,29%	Porcentaje global		88,37%

Tabla 6.14. Tabla de Clasificación

De la Tabla 6.14 surge que el porcentaje de clasificación correcta global para la muestra test es de 88,37% (error de clasificación del 11,63%); pero este valor alcanza el 94,50% para el grupo de clientes activos, y el 79,82% pare el grupo de clientes cancelados.

Este modelo de red presenta, al igual que el modelo logístico un porcentaje de clasificación correcta más grande dentro del grupo delos clientes activos que para el grupo de los clientes que están cancelados en el servicio.

6.3. Conclusiones

Entrenando distintos modelos de redes neuronales, variando la cantidad de capas intermedias y la cantidad de variables en cada capa; o considerando un conjunto diferente de variables iniciales, se seleccionó el modelo menos complejo con el cual se obtuvo una menor tasa de error global de clasificación. Este modelo considera el mismo conjunto de variables seleccionados definitivamente para el modelo logístico y tiene una capa intermedia con dos variables y función de activación tangente. A partir de este modelo es posible clasificar correctamente al 88,37% de los clientes. El resultado obtenido es bueno pero no presenta una mejora importante respecto al modelo logístico. La mejora es del 0,53% al obtenido en el modelo logístico lo que significa una diferencia de 5,3 clientes más clasificados correctamente cada mil clientes considerados.

Además, se puede destacar que con esta aplicación se confirma lo dicho en el capítulo 4 que para modelos de clasificación son suficientes dos capas intermedias.

Conclusiones finales

La teoría general del aprendizaje y la teoría estadística establecen las mismas condiciones que deben cumplir los modelos que permiten resolver problemas de clasificación supervisada pero con un lenguaje diferente.

En este trabajo, se han tratado los problemas de clasificación supervisada con modelos de redes neuronales, cuyo origen proviene de otras ciencias como la neurobiología y la física, entre otras. Partiendo de un modelo de perceptron simple, y considerando el discriminante logístico (ampliamente desarrollado en la literatura estadística) como un caso particular, se realizó una aplicación práctica de este último.

El perceptron simple, por tanto el modelo logístico, tienen el inconveniente de no ser aplicables si no se verifica el supuesto de separabilidad lineal. Avanzando hacia modelos más complejos de redes se presentaron los perceptrones multicapa los cuales no necesitan verificar el supuesto de separabilidad lineal, y permiten la resolución de problemas con regiones de decisión más complicadas.

La aplicación práctica realizada tiene que ver con el campo de las decisiones empresariales, más específicamente con el área de comercialización de las empresas. El objetivo es determinar el modelo que permita clasificar a los clientes a partir de un conjunto de características definidas y poder detectar aquellos que van a abandonar el servicio de la empresa con alta probabilidad.

Hay que destacar la importancia que tiene la definición de las características de los clientes en los resultados obtenidos, donde juega un papel fundamental la colaboración de los responsables de comercialización de la empresa, además del análisis previo realizado sobre el conjunto original de características definidas.

En este problema se debe clasificar a los clientes en dos grupos, siendo la variable respuesta binaria (0:cancelado, 1: activo. Además, las características definidas para los clientes son numéricas y categóricas, las que miden características del cliente (tipo de aparato, zona del cliente, forma de adquisición del servicio,etc) y consumo y pago del servicio (índices de consumo, índices de uso de servicios adicionales, índices de pago, entre otras).

Para encontrar el modelo se utilizó una muestra de clientes (muestra de entrenamiento) y para generalizar los resultados se utilizó otra muestra obtenida de la misma población (muestra

test), esto permite evaluar la estabilidad de los modelos obtenidos en su etapa de generalización a nuevos clientes.

Los resultados obtenidos en ambos modelos fueron comparados a partir del porcentaje de clientes correctamente clasificados (o la menor tasa de error de clasificación)

Desde la teoría estadística el modelo utilizado para clasificar el conjunto de datos es el discriminante logístico, con el cual se obtuvo un porcentaje de clasificación correcta global del 87,84% de clientes bien clasificados (un error del 12,16%) en la muestra de testeo. Analizado este porcentaje dentro de cada grupo, para los activos este valor alcanza el 93,76% y para los cancelados el 79,57%, siendo más grande el error de clasificación para el grupo de los clientes que están cancelados en el servicio. Como para el empresa es más riesgoso clasificar como activos a clientes que en realidad cancelarán el servicio, se seleccionaron los clientes con alta probabilidad de cancelar el servicio los cuales presentaban poco error de clasificación.

Los parámetros obtenidos por el Discriminante logístico permiten detectar los factores de riesgo que hacen que un cliente tenga mayor probabilidad de ser cancelado, destacándose la relación entre consumo cuatrimestral y bimestral , los clientes con deuda total mayor a $157, la historia de pago, el porcentaje de desalocación, que estén dados de baja o tramitando la baja, que tenga aparato propio o que provenga de la región patagónica.

Dentro de los modelos de redes neuronales, el perceptron multicapa ha sido utilizado con muy buenos resultados para resolver diversos problemas. El mismo posee tres características distintivas (Haykin, 1994):

1. El modelo de cada neurona aporta no linealidad a la variable respuesta. Para aplicar el algoritmo de aprendizaje son necesarias funciones continuas y derivables. Una forma de no linealidad que satisface este requisito es una función sigmoidal definida por la función logística o la tangente hiperbólica.

2. Las capas ocultas no forman parte de las entradas o salidas de la red; las mismas permiten aprender tareas complejas extrayendo progresivamente las características mas importantes de los patrones de entrada (vectores iniciales).

3. La red presenta un alto grado de conectividad determinada por las relaciones entre los nodos.

La combinación de estas características en forma conjunta, junto a la habilidad de aprender

desde la experiencia adquirida en el entrenamiento, hacen que el perceptron multicapa derive en una potente herramienta informática, que permite resolver problemas que otros métodos no pueden resolver o que resuelven con una tasa de error más alta. Sin embargo, estas mismas tres características, son las responsables de las deficiencias que presenta el comportamiento de las redes neuronales. En primer lugar, la no linealidad y la alta conectividad de la red hacen muy dificultoso una análisis teórico del perceptron multicapa. En segundo lugar, el uso de capas ocultas hace que el proceso de aprendizaje tarde en visualizarse.

Se probaron diferentes modelos de redes neuronales, cambiando la cantidad de capas intermedias, la cantidad de variables dentro de cada capa, y las funciones de activación. Aumentar el número de capas intermedias o el número de variables en cada capa no mejoró el porcentaje de clientes clasificados correctamente. Con una capa intermedia se obtuvieron buenos resultados, variando el número de variables entre dos y quince, el modelo con el que se obtuvieron mejores resultados fue con dos variables y con función de activación tangente. Este modelo logra un porcentaje de clasificación correcta del 88,37% dentro de la muestra de testeo, siendo este porcentaje de 94,50% dentro del grupo de clientes activos y de 79,82% dentro del grupo de clientes cancelados.

De esta manera el Perceptron Multicapa mejora el porcentaje de clasificación respecto al Modelo Logístico, pero esa mejora no tiene un impacto importante ya que es del 0,53%, lo cual no justifica utilizar un modelo más complejo para esa diferencia. Por otra parte hay que destacar que en el modelo Logístico se pueden interpretar los parámetros obtenidos, los cuales a través de los *odds ratio* permiten identificar los factores de riesgo que hacen que un cliente sea incluido en el grupo de los cancelados, mientras que en el modelo de red no se pueden interpretar los parámetros obtenidos.

Por lo que podemos concluir que los modelos de redes neuronales artificiales no mejoran el proceso de clasificación de los clientes de la empresa considerada, ya que el discriminante logístico permite determinar los predictores significativos en el modelo y la tasa de error que alcanza no tiene una diferencia importante que la del modelo de red seleccionado .

Los modelos de redes neuronales desarrollan métodos para clasificar objetos sin estar sujetos a supuestos y una de sus ventajas es que pueden incluir cualquier tipo de variables. Como desventaja aparece la complejidad para describir el proceso que permite dicha clasificación, ya

que es prácticamente imposible reproducir la función de clasificación. A pesar de sus desventajas en este tipo de aplicaciones, las redes neuronales son una extensión de las técnicas convencionales y es importante desarrollarlas, antes que ignorar, los resultados que ofrece este campo.

Anexo 1

La teoría de redes neuronales artificiales, se ha desarrollado en base a conceptos que surgen de la neurociencia sobre la actividad del cerebro. Es importante entonces, para comprender dichas redes conocer el funcionamiento de una célula nerviosa llamada neurona, y del conjunto de neuronas entre sí. Pero, conocer una célula nerviosa no implica conocer el comportamiento colectivo de la intrincada red de células que funcionan en el cerebro.

Actualmente, la investigación en redes neuronales está ampliamente motivada por la posibilidad de realizar redes artificiales que imiten el funcionamiento real de las neuronas del cerebro. Los modelos actuales son extremadamente simplificados desde el punto de vista neurofísico.

Funcionamiento de una neurona

El cerebro posee alrededor de 10^{11}neuronas (células nerviosas) de diferentes tipos, estando todas compuestas por las mismas partes básicas independientemente de su tamaño o forma.

Una neurona (ver figura 1 y 2) está formada por una parte central llamada *soma o cuerpo de la célula,* del cual se proyectan varias extensiones de fibras nerviosas como raíces denomimadas *dendritas,* así como una fibra tubular más larga denominada *axon* la cual termina ramificándose en su extremo. Las dendritas (y por consiguiente el cuerpo de la célula) son receptores de señales desde neuronas adyacentes, mientras que el axon trasmite la actividad neuronal generada a otras células nerviosas o fibras musculares a través de uniones sinápticas o *sinapsis.* El axon de una neurona típica realiza unas miles de sinapsis con otras neuronas. La sinapsis entre dos células está separada por una pequeña hendidura que puede localizarse directamente sobre el cuerpo de la neurona o las dendritas; la energía de su influencia, generalmente disminuye al incrementarse la distancia de la sinapsis al cuerpo de la célula.

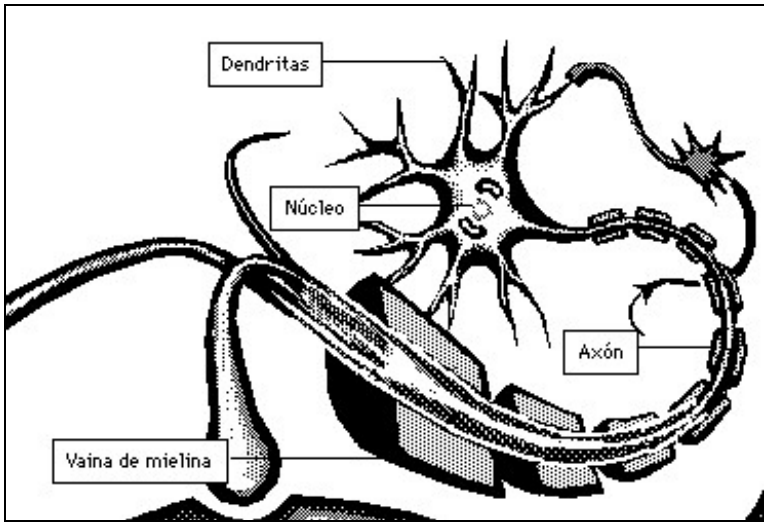


Figura 1

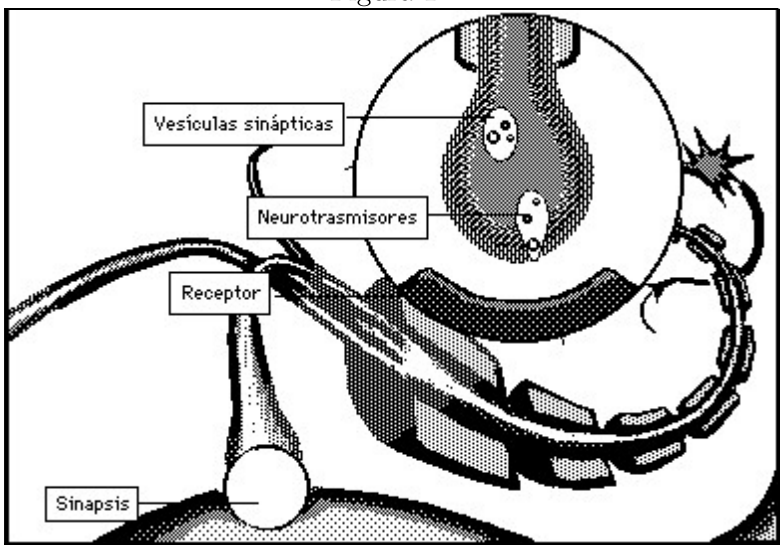


Figura 2

Las señales nerviosas son trasmitidas tanto eléctrica como químicamente (Ver Figura 3). La trasmisión eléctrica prevalece dentro de una neurona mientras que los mecanismos químicos operan entre diferentes neuronas. La trasmisión eléctrica está basada en una descarga eléctrica que comienza en el cuerpo de la célula y se trasmite por el axon hacia varias conexiones sinápticas. En estado de inactividad en el interior de la neurona, *el protoplasma* está cargado negativamente (rico en sodio) a diferencia del líquido neural que lo rodea (rico en potasio). La membrana de la célula soporta una diferencia potencial o umbral determinado. Las señales que provienen de las conexiones sinápticas realizan un debilitamiento transitorio o despolarización

que permiten el ingreso al protoplasma de iones positivos neutralizando la diferencia potencial. La membrana gradualmente recupera sus propiedades originales; durante este período de regeneración la neurona permanece inactiva ante otra exitación. Cuando la recuperación es completada, la neurona en su estado de reposo puede activarse nuevamente. La descarga que inicialmente ocurre en el cuerpo de la célula se propaga a las sinapsis a través del axon, siempre en una dirección y la intensidad de la señal trasmitida no decae en su recorrido por la fibra nerviosa (por la acción de la mielina, funda aislante que recubre el axon y la acción de los nodos Ranvier que se encuentran en el recorrido del axon, que permiten que la señal se propague como una onda guiada de un nodo a otro). La señal de trasmisión en el sistema nervioso no significa que una neurona está totalmente activa o inactiva, sino que la intensidad de una señal nerviosa está codificada en una frecuencia de sucesión de pulsos de actividad los que varían de 1 a 100 por segundo; la combinación de procesamiento de señales digitales y analógicas permite obtener una trasmisión de datos de óptima calidad, seguridad y simplicidad. La velocidad de propagación de la señal de descarga a través de la fibra nerviosa también varía ampliamente.

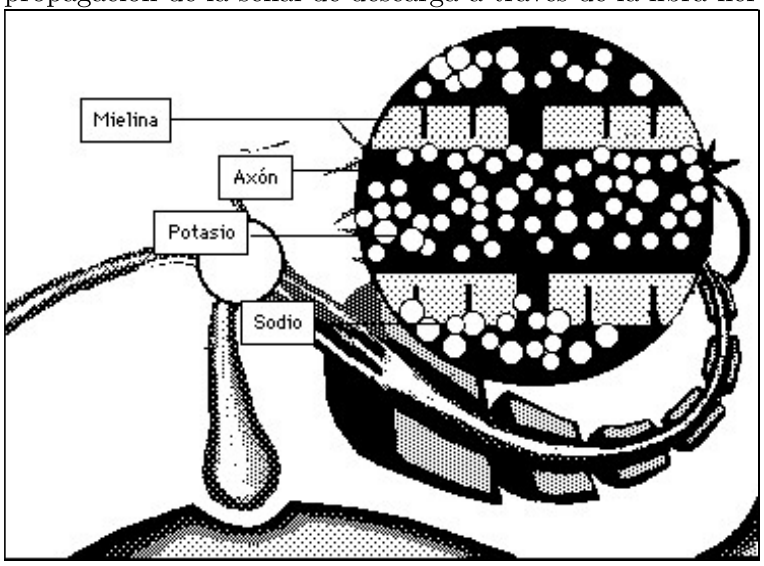


Figura 3

La señal de descarga que viaja a través del axon viene a parar en la sinapsis, porque no existe un puente de conductividad hacia la próxima neurona o músculo. La señal de trasmisión a través de la hendidura sináptica está principalmente afectada por un mecanismo químico. Cuando la señal arriba a la terminal nerviosa presináptica, son liberadas en pequeñas cantidades sustancias especiales llamadas *neurotrasmisores,* estas sustancias viajan por la hendidura sináptica

alcanzando la neurona (o fibra muscular) postsináptica. A su arribo, receptores especiales de estas sustancias modifican la conductividad de la membrana postsináptica para ciertos iones, los cuales fluyen dentro o fuera de la neurona causando la polarización o despolarización del potencial postsináptico local. Después de su acción las moléculas trasmisoras son rápidamente partidas en pequeños pedazos por enzimas. Si el potencial de polarización inducido es positivo se denomina sinapsis exitatorio, ya que la influencia de la misma tiende a activar la neurona posináptica. En caso de ser negativo, es una sinapsis inhibitoria ya que inhibe la posibilidad de enviar neurotrasmisores a través de la hendidura postsináptica.

Aunque en principio una única sinapsis puede ocasionar la activación de la neurona posináptica, esto raramente ocurre. El cuerpo de una neurona actúa como un elemento sumatorio el que va acumulando el efecto de sus varias señales de entrada; cuando el total de magnitud del potencial de depolarización en el cuerpo de la célula excede un umbral, la neurona se activa.

La influencia de una sinapsis dada depende de varios aspectos: de la fuerza de su efecto de depolarización, su ubicación con respecto al cuerpo de la célula y la tasa de repetición de las señales entrantes.

Historia de los modelos de redes neuronales

El primer modelo de procesamiento de información basados en redes con unidades binarias aparece en el trabajo de McCulloch and Pitts (1943), las que de alguna forma fueron denominadas "neuronas" aunque distaban bastante de sus análogas biológicas.

Cada elemento $i = 1, 2....n$ puede asumir valores $x_i = -1, 1$, donde $x_i = -1$ representa el estado de reposo y $x_i = 1$, el estado activo del elemento. Para simular la secuencia de estados de las neuronas reales, se supone que estos ocurren en pasos del tiempos discretos $t = 0, 1, 2, ...$ El nuevo estado de una neurona se determina por la influencia de todas las otras neuronas, lo cual se expresa por una combinación lineal de sus valores de entrada.

$$h_i(t) = \sum_j w_{ij} x_j(t) \tag{1}$$

donde la matriz w_{ij} representa la fuerza de las sinapsis (o la eficacia sináptica) entre las neuronas j e i, mientras $h_i(t)$ modela el potencial de polarización postsináptica total de la neurona i causado por la acción de todas las neuronas, representando una entrada de una

unidad neuronal, siendo x_i la salida. El supuesto más simple es que la neurona comienza a activarse si sus entradas exceden a cierto umbral b_i el cual debe ser diferente de una unidad a la siguiente. La ley que gobierna la evolución de la red es la siguiente:

$x_i(t+1) = \Theta(h_i(t) - b_i)$, donde $\Theta(h_i(t) - b_i)$ es una función escalón.

Las razones por las que este modelo difiere de una neurona real son las siguientes:

- Una neurona real responde a sus entradas en un sentido continuo como una respuesta gradual; y en este caso la función escalón definida en el modelo, no corresponde a una neurona real. La relación no lineal entre la entrada y la salida de una célula es una característica universal.

- Algunas neuronas reales resultan de la suma no lineal de sus entradas.

- La neuronas reales producen una secuencia de pulsos, no un simple nivel de respuesta, por lo que representar la tasa de activación como un simple valor x_i, aún siendo continua, ignora mucha información como la fase del pulso, que debería ser obtenida por una secuencia de pulsos.

- Las neuronas tienen diferentes demoras, no son activadas en forma sincronizadas por un reloj central , por lo que no es correcto considerar demoras $t \longrightarrow t+1$.

- La cantidad de sustancia trasmisora liberada (o disparada) por una sinapsis varía impredeciblemente. Este tipo de efecto puede ser en cierta parte modelado por una generalización estocástica de la dinámica de McCulloch and Pitts.

La siguiente expresión es una simple generalización de la ecuación 1, que incluye algunas de las características anteriores:

$$x_i = f(\sum_j w_{ij} x_j - b_i) \tag{2}$$

El valor x_i es un valor continuo denominado el estado o activación de la unidad i. La función escalón es reemplazada on una función no lineal más general, llamada función de activación, función de ganancia o función transferencia. No se han definido períodos de tiempo, simplemente se da una regla de actualización de x_i cuando ocurre. Las unidades son actualizadas asincrónicamente en orden aleatorio y en tiempos aleatorios.

La neurona de McCulloch and Pitts representa un elemento computacional muy potente. El diseño de esta red trajo como consecuencia el problema de cómo elegir los acoplamientos

w_{ij}. Esta pregunta fue retomada en 1961 por Eduardo Caianello quien desarrolló un algoritmo de aprendizaje que permitió determinar los pesos sinápticos de una red neuronal; el mismo fue denominado la ecuación de Caianello *mnemonic* e incorporada después al principio básico de la regla de aprendizaje de Hebb.

Alrededor de 1960 Frank Rosenblatt y sus colaboradores estudiaron un tipo específico de red neuronal, la que fue llamada *perceptron,* porque la consideraron como un modelo simplificado de mecanismos biológicos de procesamiento de información sensorial, como la percepción. Un perceptron consiste en dos capas de neuronas, una de entrada y otra de salida. Las neuronas de la capa de salida reciben señales sinápticas desde la capa de entrada pero no viceversa, y las neuronas dentro de cada capa no se comunican unas con otras. La información es estrictamente unidireccional, por lo que se habla de neuronas direccionadas hacia adelante (en inglés *feed-forward networks)* como se muestra en la figura 4

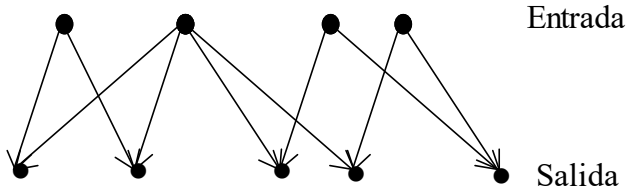


Figura 4

El grupo de Rosenblatt desarrolló un algoritmo iteractivo para calcular los pesos sinápticos, probando su convergencia. Mucha gente mostraba gran entusiasmo y pensaban que estos nuevos algoritmos eran la base de la inteligencia artificial. Sin embargo en 1969, Marvin Minsky y Seymore Papert, demostraron en su libro Perceptrones las limitaciones del perceptron para resolver problemas tan simples como una operación lógica de disyunción exclusiva . Esto llevó a Rosenblatt a estudiar estructuras con más capas de neuronas, incorporando capas intermedias, con lo cual pensaba sobreponer a las limitaciones de los perceptrones simples. Sin embargo, no existía un algoritmo de aprendizaje para determinar los pesos sinápticos, y la comunidad de la ciencia de la computación dejó en suspenso el paradigma de las redes neuronales para 20 años después.

Durante los años 70 se continuó la investigación en la teoría de redes neuronales en temas como la memoria asociativa con contenido direccionable. Otro desarrollo comenzó cuando Wi-

llian Littlem, seguido entre otros por John Hopfield, mostró la similitud entre una red neuronal del tipo propuesto por McCulloch and Pitts los sistemas de momentos magnéticos o spins introduciéndose de esta manera dentro del campo de la física, incorporando conceptos como función energía o temperatura.

En los años más recientes fue revivido el interés por las estructuras de redes direccionadas hacia delante, desarrollando un algoritmo eficiente propuesto por Werbos en 1974 para la determinación de las potencias sinápticas en redes con capas de neuronas intermedias denominadas capas ocultas, reconocido posteriormente por varios grupos de científicos alrededor de 1985 , el cual se denomina algoritmo de retropropagación del error (en inglés *error backpropagation).*

Anexo 2

Base de datos Empresa telefonía celular

Se tomó una muestra 9702 clientes, entre activos y, cancelados y suspendidos . La variable target corresponde al estado del servicio, codificada como:

0- Activos
1- Cancelados y suspendidos

Tabla 1 : Variables de análisis

Descripción	*Código*	*Unidad de medida*	*Tipo de variable*
1-Promedio consumo vida activa	prom_act	minutos	cuantitativa
2-Nro de contactos prom con CS hasta 60 dias despues de la activ	prom_cs1	Minutos	cuantitativa
3-Antiguedad del cliente en días	d_edad	días	cuantitativa
4-Tiempo que resta para el fin del contrato	d_fincon	días	cuantitativa
5-Historia de pago (cuanto más cercano a 1 mejor será la historia de pago del cliente)	hist_p	indice	cuantitativa
6-Porcentaje de desalocación	porc_des	porcentaje	cuantitativa
7-Prom Cons 4meses/2meses	prom4_2	Indice	cuantitativa
8-Deuda total	deuda	Pesos	cuantitativa
9-Quejas en la cuenta 0. Sin queja 1. Una queja 2. Dos o más quejas	queja_n1	cantidad	Categórica (3)
10-Duración del contrato	d_durac		categórica (3)
11-*Región 1.Bs As –La Pampa 2. Bs As –Sta Fer 3. Cuyo 4. Litoral Norte 5. Litoral Sur 6. Mediterráneo 7. Noroeste Argentino 8. Patagonia	rregion		categórica (8)
12-Forma de adquisición 1.Comodato 2. Stock distribuído 3.Leasing 4. Propio	Radquisi		categórica (4)
13-Tipo de cuenta 1.Negoscios 2. Otros 3.Personal 4. Top	rtip_cta		categórica (4)
14-Canal de distribución 1.Directo 2. Indirecto 3.Multinivel 4. Otros	rchannel		categórica (4)
15-Estado del Agente 1.Activo 2. Baja 3. Otros 4. Tramitando baja	rest_ag		categórica (4)
16-Modelo 1.Motorola Móvil	rmodelo		categórica (7)

2. Motorola Portátil 3. Motorola Transportable 4. Nokia Portátil 5. Otros Móvil 6. Otros Portátil 7. Otros Transportable			
17-Débito automático 0. NO 1. SI	da		categórica (2)
18-Plazo de la deuda 0. Sin deuda/Corto plazo 1. Mediano/Largo Plazo	rtd		categórica (2)

Tabla 2 : Regresiones logísticas univariadas de variables numéricas.

Variables							% Clasificación correcta		
Numéricas	B	E.T.	Wald	gl	Sig.	Exp(B)	Activos	Cancelados	Total
PROM_ACT	0.0020	0.0001	198.61	1	0.00	1.00	94.00	19.40	63.60
Constante	-0.7039	0.0349	406.74	1	0.00	0.49			
PROM_CS1	-0.0008	0.0011	0.58	1	0.44	1.00	100.00	0.00	59.30
Constante	-0.3740	0.0269	193.81	1	0.00	0.69			
D_EDAD	-0.0050	0.0002	959.03	1	0.00	1.00	76.30	63.20	71.00
Constante	1.6850	0.0695	587.67	1	0.00	5.39			
D_FINCON	0.0035	0.0002	484.75	1.00	0.00	1.00	76.80	46.90	64.70
Constante	-1.5378	0.0612	631.25	1.00	0.00	0.21			
HIST_P	0.9405	0.0243	1496.51	1.00	0.00	2.56	92.90	69.70	83.40
Constante	-2.8335	0.0661	1835.80	1.00	0.00	0.06			
PORC_DES	0.8026	0.0649	152.94	1.00	0.00	2.23	94.90	16.20	62.80
Constante	-0.5651	0.0303	347.71	1.00	0.00	0.57			
PROM4_2	0.0045	0.0001	1060.90	1.00	0.00	1.00	89.60	52.90	74.60
Constante	-1.0269	0.0352	850.58	1.00	0.00	0.36			
DEUDA	0.0011	0.0001	161.59	1.00	0.00	1.00	98.20	22.00	67.20
Constante	-0.5772	0.0301	368.72	1.00	0.00	0.56			

Tabla 3 : Prueba de muestras independientes

		Prueba de Levene para la igualdad de varianzas		Prueba T para la igualdad de medias						
	Supuesto	F	Sig.	T	gl	Sig. (bilateral)	Diferencia de medias	Error típ. de la diferencia	95% Intervalo de confianza para la diferencia	
									Inf.	Sup.
Nro de contactos con CS hasta 60 dias despues de la activ.	Varianzas iguales	2.21	.137	.772	5769	.440	.56	.72	-.86	1.98
	Varianzas no iguales			.749	4500.358	.454	.56	.74	-.90	2.03
Antiguedad del cliente en días	Varianzas iguales	57.23	.000	37.051	5769	.000	196.40	5.30	186.01	206.80
	Varianzas no iguales			38.231	5533.176	.000	196.40	5.13	186.33	206.48
Tiempo que resta para el fin del contrato	Varianzas iguales	55.87	.000	-23.777	5769	.000	-113.26	4.76	-122.60	-103.92
	Varianzas no iguales			-24.280	5391.444	.000	-113.26	4.66	-122.41	-104.12
Historia de pago	Varianzas iguales	3050.	.000	-69.989	5769	.000	-2.96	.042	-3.05	-2.88
	Varianzas no iguales			-61.824	3004.117	.000	-2.96	.04	-3.06	-2.87
Porcentaje de desalocación	Varianzas iguales	428.6	.000	-13.952	5769	.000	-.27	.01	-.30	-.23
	Varianzas no iguales			-12.187	2869.976	.000	-.27	.02	-.31	-.22
Promedio consumo vida	Varianzas iguales	547.9	.000	-15.985	5769	.000	-113.56	7.10	-127.49	-99.64
	Varianzas no iguales			-14.029	2925.355	.000	-113.56	8.09	-129.44	-97.69
Prom Cons 4mese/2meses	Varianzas iguales	3637.	.000	-39.935	5769	.000	-211.99	5.30	-222.39	-201.58
	Varianzas no iguales			-36.789	3592.379	.000	-211.99	5.76	-223.28	-200.69

Tabla 4: Variables Categóricas Regresiones logísticas univariadas

							% Clasificación correcta		
	B	E.T.	Wald	gl	Sig.	Exp(B)	Activos	Cancelados	Total
QUEJA_N1	0.00		9.55	2	0.01		100.00	0.00	59.30
QUEJA_N1(1)	-0.21	0.12	3.10	1	0.08	0.81			
QUEJA_N1(2)	-0.65	0.25	6.67	1	0.01	0.52			
Constante	-0.36	0.03	164.98	1	0.00	0.70			
D_DURAC	0.00		127.03	2	0.00		97.10	10.00	61.70
D_DURAC(1)	1.14	0.15	60.84	1	0.00	3.14			
D_DURAC(2)	-0.25	0.09	8.42	1	0.00	0.78			
Constante	-0.23	0.08	8.13	1	0.00	0.79			
REGIÓN	0.00		149.30	7	0.00		81.10	31.00	60.10
RREGION(1)	-0.28	0.11	6.10	1	0.01	0.76			
RREGION(2)	0.11	0.11	1.03	1	0.31	1.11			
RREGION(3)	-0.05	0.09	0.30	1	0.59	0.95			
RREGION(4)	0.69	0.12	35.24	1	0.00	1.99			
RREGION(5)	-0.42	0.09	20.45	1	0.00	0.66			
RREGION(6)	0.44	0.11	16.56	1	0.00	1.56			
RREGION(7)	0.46	0.12	14.26	1	0.00	1.58			
Constante	-0.40	0.07	38.00	1	0.00	0.67			
RADQUISI	0.00		76.41	3	0.00		98.30	4.00	59.90
RADQUISI(1)	-0.24	0.08	8.19	1	0.00	0.79			
RADQUISI(2)	0.99	0.18	31.75	1	0.00	2.70			
RADQUISI(3)	0.25	0.07	14.84	1	0.00	1.29			
Constante	-0.49	0.05	83.20	1	0.00	0.61			
RTIP_CTA	0.00		84.84	3	0.00		99.30	1.30	59.40
RTIP_CTA(1)	0.86	0.28	9.70	1	0.00	2.36			
RTIP_CTA(2)	0.38	0.06	41.72	1	0.00	1.46			
RTIP_CTA(3)	-1.03	0.21	23.39	1	0.00	0.36			
Constante	-0.60	0.05	156.75	1	0.00	0.55			
RCHANNEL	0.00		6.25	3	0.10		100.00	0.00	59.30
RCHANNEL(1)	0.14	0.07	4.53	1	0.03	1.16			
RCHANNEL(2)	-0.05	0.18	0.06	1	0.80	0.96			
RCHANNEL(3)	-0.07	0.21	0.12	1	0.73	0.93			
Constante	-0.48	0.06	63.21	1	0.00	0.62			
REST_AG	0.00		61.47	3	0.00		100.00	0.00	59.30
REST_AG(1)	0.46	0.06	60.72	1	0.00	1.58			
REST_AG(2)	0.21	0.12	3.05	1	0.08	1.23			
REST_AG(3)	0.33	0.10	10.19	1	0.00	1.39			
Constante	-0.64	0.04	203.30	1	0.00	0.53			
RMODELO	0.00		29.68	6	0.00		99.90	0.20	59.30
RMODELO(1)	0.04	0.31	0.02	1	0.89	1.04			
RMODELO(2)	-0.03	0.35	0.01	1	0.94	0.97			
RMODELO(3)	0.31	0.31	1.01	1	0.31	1.36			
RMODELO(4)	-0.68	0.46	2.18	1	0.14	0.51			
RMODELO(5)	0.25	0.31	0.63	1	0.43	1.28			
RMODELO(6)	0.82	0.82	1.00	1	0.32	2.27			
Constante	-0.53	0.31	3.06	1	0.08	0.59			
RTD(1)	2.48	0.07	1441.72	1	0.00	11.96	85.40	67.20	78.00
Constante	-1.33	0.04	1082.25	1	0.00	0.26			
DA(1)	1.83	0.08	506.29	1	0.00	6.22	100.00	0.00	59.30
Constante	-1.83	0.08	589.94	1	0.00	0.16			

Tabla 5: Variables categóricas: Pruebas Chi Cuadrado de Pearson

Variable	Valor	gl	Sig. asintótica (bilateral)
QUEJA_N1	9.7689	2	0.0076
D_DURAC	143.3969	2	0.0000
RREGION	152.6026	7	0.0000
RADQUISI	78.5724	3	0.0000
RTIP_CTA	90.4644	3	0.0000
RCHANNEL	6.2540	3	0.0999
REST_AG	61.7835	3	0.0000
RMODELO	30.1118	6	0.0000
RTD	595.3255	1	0.0000
DA	1670.8172	1	0.0000

Tabla 6: Resultados de la eliminación paso a paso

Variable eliminada	**-2 log de la verosimilitud**	**R cuadrado de Cox y Snell**	**R cuadrado de Nagelkerke**	**gl**	**%Clasif. Correcta**	**dif de verosim**	**dif gl**	**chi P value**
Deuda	3666,62	0,51	0,69	32	87,20			
quejas en la cuenta y deuda	3670,95	0,51	0,69	31	87,30	4,33	1	0,04
quejas en la cuenta, deuda total y da	3675,61	0,51	0,69	30	87,40	8,99	2	0,01

Tabla 7: Variable de partida (a partir de la tabla 6) **y Variables definitivas**
(verificando el cumplimiento de los supuestos del discriminante logístico)

Descripción Variables de partida	*Código*	*Descripción Variables definitivas*	*Código*
1-Promedio consumo vida activa	prom_act	1-Promedio consumo vida activa	prom_act
2-Antiguedad del cliente en días	d_edad	2-Antiguedad del cliente en días	d_edad
3-Tiempo que resta para el fin del contrato	d_fincon	3-Tiempo que resta para el fin del contrato	d_fincon
4-Historia de pago	hist_p	4-Historia de pago	hist_p
5-Porcentaje de desalocación	porc_des	5-Porcentaje de desalocación	porc_des
6-Prom Cons 4mese/2meses	prom4_2	6-Prom Cons 4mese/2meses 1. Prom consumo entre 0,02 y 1,2487 2. Prom consumo de más de 1.2487 hasta 500 3. . Prom consumo mayor a 500	tprom4_2
7-Deuda total	Deuda	7-Deuda total 1.Sin deuda 2. Deuda total hasta 157,25 3. Mayor de 157,25	tdeuda
8-Duración del contrato	d_durac	8-Duración del contrato	d_durac
9-*Región 1.Bs As –La Pampa 2. Bs As –Sta Fer 3. Cuyo 4. Litoral Norte 5. Litoral Sur 6. Mediterráneo 7. Noroeste Argentino 8. Patagonia	Región	9-*Región 1.Bs As, Sta Fey La Pampa 2. Cuyo , Litoral, Mediterráneo y NOAR 3. Patagonia	region_3
10-Forma de adquisición 1.Comodato 2. Stock distribuído 3.Leasing 4. Propio	Radquisi	10-Forma de adquisición 1.Comodato 2. Stock distribuído y Leasing 3. Propio	adquisi1
11-Tipo de cuenta 1.Negoscios 2. Otros 3.Personal 4. Top	rtip_cta	11-Tipo de cuenta 1.Negocios 2. Otros 3.Personal 4. Top	rtip_cta
12-Estado del Agente	rest_ag	12-Estado del Agente	rest_ag1

1.Activo 2. Baja 3. Otros 4. Tramitando baja		1.Activo 2. Baja 3. Tramitando baja y Otros	
13-Modelo 1.Motorola Móvil 2. Motorola Portátil 3. Motorola Transportable 4. Nokia Portátil 5. Otros Móvil 6. Otros Portátil 7. Otros Transportable	Rmodelo	13-Modelo 1.Motorola 2. Nokia 3. Otros	mmodelo
14-Plazo de la deuda 0. Sin deuda/Corto plazo 1. Mediano/Largo Plazo	Rtd		

Tabla 8: Comparación entre los modelos detalladas en la tabla 7

Variable eliminada	*-2 log de la verosimilitud*	*R cuadrado de Cox y Snell*	*R cuadrado de Nagelkerke*	*gl*	*%Clasif. Correcta*	*dif de verosim*	*dif gl*	*chi P value*
Modelo de partida	3675,02	0,51	0,69	32	87,40			
Modelo definitivo	3542,77	0,52	0,70	22	88.20	132,25	10	0,00

Tabla 9: Codificaciones de variables categóricas

Variables categóricas	Etiqueta de las categorías	Frecuencia	Codificación de parámetros		
			(1)	(2)	(3)
Tipo de cuenta	NEGOCIOS	1879	0	0	0
	OTROS	55	1	0	0
	PERSONAL	3666	0	1	0
	TOP	171	0	0	1
Región	Bs As, Sta Fe y La Pampa	1492	0	0	
	Cuyo, Medi, Litoral y NOAR	3889	1	0	
	Patagonia	390	0	1	
Forma de adquisición	COMO	1451	0	0	
	DIST y LEAS	1252	1	0	
	PROP	3068	0	1	
Prom Cons 4mese/2meses	1	2864	0	0	
	2	1308	1	0	
	3	1599	0	1	
Estado del Agente	A	2212	0	0	
	B	2742	1	0	
	O y P	817	0	1	
Deuda total	1	3275	0	0	
	2	1087	1	0	
	3	1409	0	1	
Modelo	Motorola	2890	0	0	
	Nokia	1924	1	0	
	Otros	957	0	1	
Duración del contrato	1	617	0	0	
	2	328	1	0	
	3	4826	0	1	

Tabla 10:Coeficientes de la Regresión Logística

	B	E.T.	Wald	gl	Sig.	Exp(B)	I.C. 95.0% para EXP(B)	
							Inf.	Sup.
D_EDAD	-.007	.000	229.601	1	.000	.993	.993	.994
D_FINCON	-.001	.000	8.263	1	.004	.999	.998	1.000
HIST_P	.710	.038	352.030	1	.000	2.034	1.889	2.191
PORC_DES	.320	.076	17.719	1	.000	1.377	1.186	1.598
PROM_ACT	.001	.000	4.681	1	.030	1.001	1.000	1.001
D_DURAC			64.073	2	.000			
D_DURAC(1)	-.844	.273	9.541	1	.002	.430	.252	.735
D_DURAC(2)	-1.473	.187	62.029	1	.000	.229	.159	.331
EGION_3			8.007	2	.018			
REGION_3(1)	.194	.102	3.582	1	.058	1.214	.993	1.483
REGION_3(2)	.491	.182	7.239	1	.007	1.633	1.142	2.335
ADQUISI1			40.003	2	.000			
ADQUISI1(1)	-.224	.127	3.091	1	.079	.799	.623	1.026
ADQUISI1(2)	.501	.116	18.739	1	.000	1.650	1.315	2.070
RTIP_CTA			55.169	3	.000			
RTIP_CTA(1)	-.812	.421	3.720	1	.054	.444	.195	1.013
RTIP_CTA(2)	-.260	.093	7.776	1	.005	.771	.642	.926
RTIP_CTA(3)	-2.079	.289	51.909	1	.000	.125	.071	.220
REST_AG1			57.026	2	.000			
REST_AG1(1)	.777	.103	56.783	1	.000	2.176	1.777	2.663
REST_AG1(2)	.343	.135	6.433	1	.011	1.409	1.081	1.837
MMODELO			18.604	2	.000			
MMODELO(1)	.389	.104	13.961	1	.000	1.475	1.203	1.809
MMODELO(2)	-.106	.125	.723	1	.395	.899	.704	1.148
TDEUDA			127.753	2	.000			
TDEUDA(1)	-.709	.122	33.505	1	.000	.492	.387	.626
TDEUDA(2)	1.055	.153	47.464	1	.000	2.871	2.127	3.875
TPROM4_2			260.890	2	.000			
TPROM4_2(1)	.564	.104	29.582	1	.000	1.758	1.434	2.154
TPROM4_2(2)	1.851	.115	260.730	1	.000	6.368	5.086	7.972
Constante	.932	.437	4.546	1	.033	2.539		

Gráfico 1:Grupos observados y probabilidades estimadas

```
F         ⊥                                                                  ⊥
R    1200 ⊦                                                                  ⊦
E         ⊥                                                                  ⊥
Q         ⊥                                                                  ⊥
U         ⊥                                                                 1⊥
E     800 ⊦                                                                 1⊦
N         ⊥                                                                 1⊥
C         ⊥                                                                 1⊥
Y         ⊥                                                                 1⊥
      400 ⊦   1                                                             1⊦
          ⊥ 00000                                                           1⊥
          ⊥000000000                                                       11⊥
          ⊥000000000000000000                                     1111111⊥
Predicted ┬┬┬┬┬┬┬┬┬┬┬┬┬┬┬⊦┬┬┬┬┬┬┬┬┬┬┬┬┬┬┬⊦┬┬┬┬┬┬┬┬┬┬┬┬┬┬┬⊦┬┬┬┬┬┬┬┬┬┬┬┬┬┬┬
  Prob:   0              .25             .5              .75              1
  Group:  00000000000000000000000000000011111111111111111111111111111111
          Predicted Probability is of Membership for A
          The Cut Value is .50
          Symbols: 0 - Activo
                   1 - Cancelado
          Each Symbol Represents 100 Cases.
```

Gráficos del Análisis de diagnóstico del modelo logístico

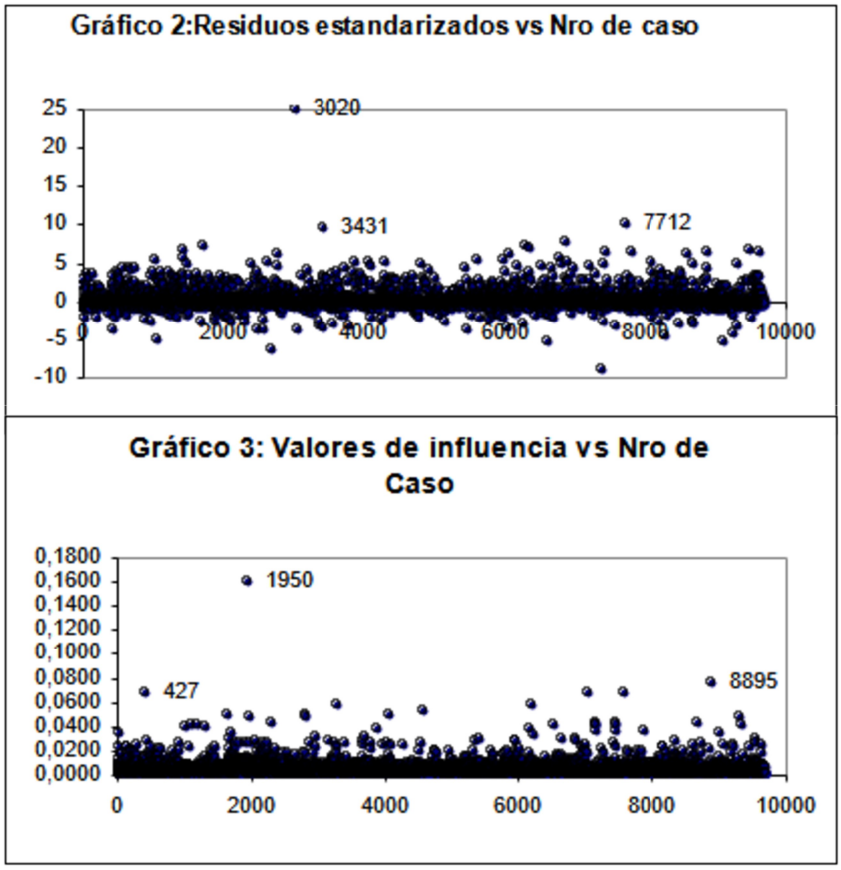


Gráfico 4

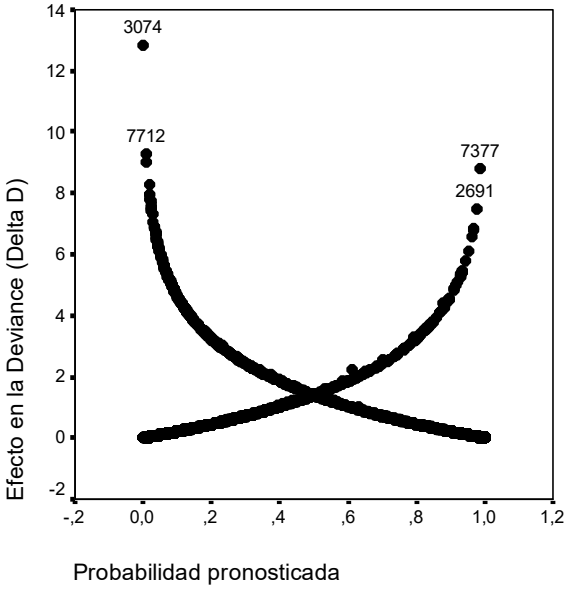


Gráfico 5

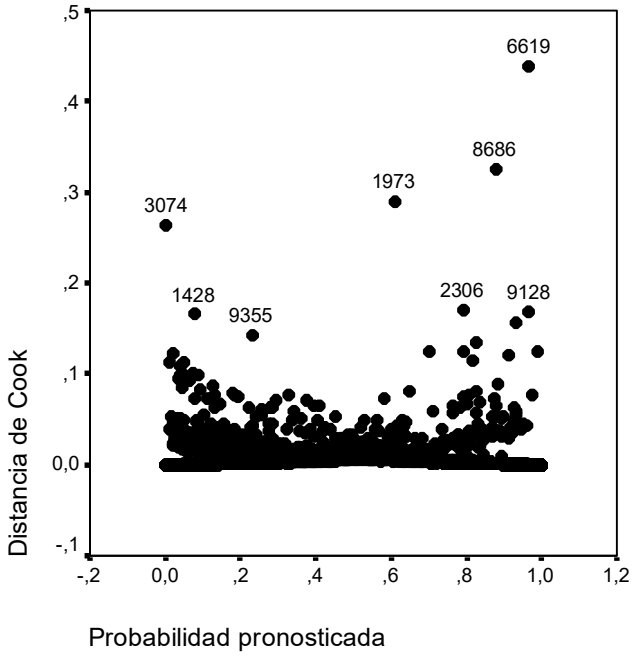


Tabla 11: Parámetros del modelo de red seleccionado (una capa intermedia con dos variables y función de activación tangente)

	Capa intermedia		
	Variable 1	Variable2	
PROM_ACT	0,0980	0,2348	
D_EDAD	-1,4971	-0,2469	
D_FINCON	-0,1123	0,1926	
PORD_ES	-0,1146	0,1858	
HIST_P	-0,5394	1,0244	
DURA1	0,3018	0,7853	
DURA2	-1,3586	0,8624	
DURA3	0,0787	-0,2739	
PROM421	-0,1242	-0,1492	
PROM422	-0,0865	0,2946	
PROM423	0,4184	0,8565	
MODEL1	-0,5375	0,1056	
MODEL2	-0,5011	0,3012	
MODEL3	-0,6347	0,1040	
TDEUDA1	-0,6444	0,3544	
TDEUDA2	-0,1820	-0,2196	
TDEUDA3	0,1931	0,5754	
REGION1	-0,7418	0,3362	
REGION2	-0,4528	0,3452	
REGION3	-0,3551	0,3731	
ADQUI1	-1,6322	0,5294	
ADQUI2	0,0728	0,2927	
ADQUI3	0,1512	0,4599	
RESTAG1	-0,4246	-0,3283	
RESTAG2	-1,0650	0,2502	
RESTAG3	-0,8348	-0,1026	
TIPCTA1	0,2952	0,6810	
TIPCTA2	-0,4777	0,8692	
TIPCTA3	-0,0397	0,6854	
TIPCTA4	0,1618	-0,2839	
Constante	3,9536	-2,3233	
Capa intermedia		Variable 1	2,0342
		Variable 2	2,9433
		Constante	-1,2968

Variable Respuesta

Anexo 3

Resultados obtenidos a partir de un primer análisis de las variables.

En la tabla siguiente se muestran los cambios resultantes en la primera etapa del análisis:

Tabla 1: Variable de partida (a partir de la tabla 6 del Anexo2) **y Variables definidas en una primera etapa** (verificando el cumplimiento de los supuestos del discriminante logístico)

Descripción Variables de partida	*Descripción Variables definitivas*
1-Promedio consumo vida activa	
2-Antiguedad del cliente en días	2-Antiguedad del cliente en días
3-Tiempo que resta para el fin del contrato	
4-Historia de pago	4-Historia de pago
5-Porcentaje de desalocación	5-Porcentaje de desalocación
6-Prom Cons 4mese/2meses	6-Prom Cons 4mese/2meses
7-Deuda total	
8-Duración del contrato	8-Duración del contrato
9-*Región 1.Bs As –La Pampa 2. Bs As –Sta Fer 3. Cuyo 4. Litoral Norte 5. Litoral Sur 6. Mediterráneo 7. Noroeste Argentino 8. Patagonia	9-*Región 1.Bs As –La Pampa 2. Bs As –Sta Fer 3. Cuyo 4. Litoral Norte 5. Litoral Sur 6. Mediterráneo 7. Noroeste Argentino 8. Patagonia
10-Forma de adquisición 1.Comodato 2. Stock distribuído 3.Leasing 4. Propio	10-Forma de adquisición 1.Comodato 2. Stock distribuído 3.Leasing 4. Propio
11-Tipo de cuenta 1.Negoscios 2. Otros 3.Personal 4. Top	11-Tipo de cuenta 1.Negocios 2. Otros y Top 3.Personal
12-Estado del Agente 1.Activo 2. Baja 3. Otros 4. Tramitando baja	12-Estado del Agente 1.Activo 2. Baja 3. Otros 4. Tramitando baja
13-Modelo 1.Motorola Móvil 2. Motorola Portátil 3. Motorola Transportable 4. Nokia Portátil 5. Otros Móvil 6. Otros Portátil 7. Otros Transportable	13-Modelo 1.Móvil 2. Portátil 3. Otros
14-Plazo de la deuda 0. Sin deuda/Corto plazo 1. Mediano/Largo Plazo	14-Plazo de la deuda 0. Sin deuda/Corto plazo 1. Mediano/Largo Plazo

Tabla 2: Tabla de clasificación

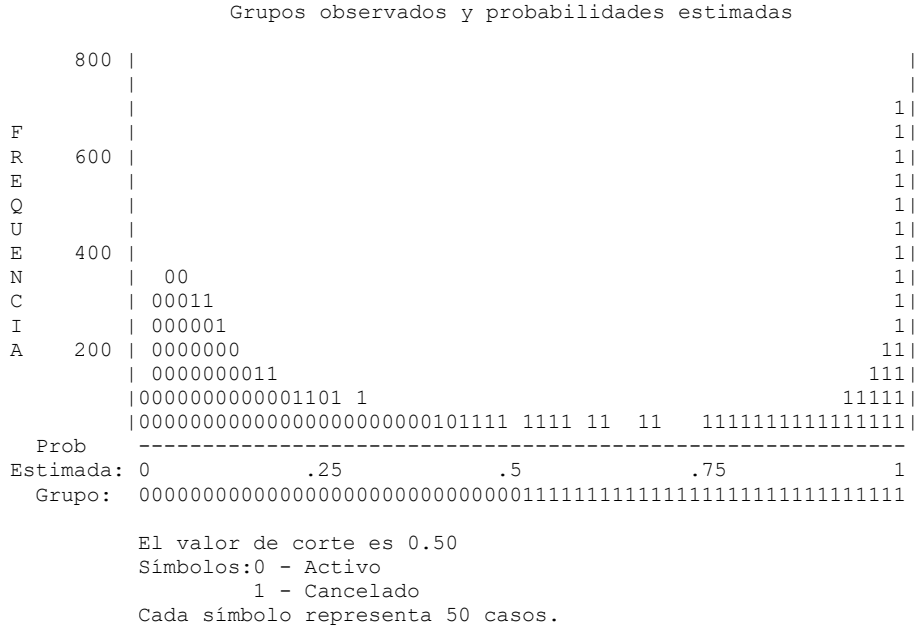
Resultados del modelo logístico

El modelo logístico estima una función, a partir de la cual los clientes de la muestra quedan clasificados como muestra la Tabla 2.

		Valores predichos					
		Muestra de modelación			Muestra test		
		Estado del cliente		Porcentaje bien clasificados	Estado del cliente		Porcentaje bien clasificados
Observados		Cancelados	Activos		Cancelados	Activos	
Estado del cliente	Activos	3176	223	93.4	2113	200	91.7
	Cancelados	528	1841	77.7	324	1297	80.0
Porcentaje sobre el total				87.0			**86.7**

En la muestra de modelación el porcentaje de clientes mal clasificados asciende al 13%, lo que puede ser visualizado en el gráfico 3. Cuando se consideran los clientes incluídos en la muestra test, el porcentaje de error se incrementa solo el 3%, lo que pone en evidencia la perfomance del modelo.

Figura 1: Clasificación de los clientes de la empresa obtenida por el discriminante logístico



Resultados Redes neuronales

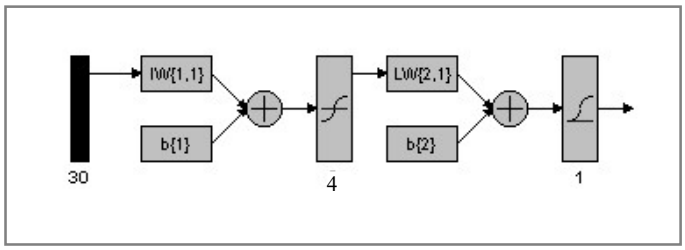
Tomando la selección de variables realizada en el modelo logístico, se estandarizaron los valores de las mismas, para eliminar problemas numéricos al utilizar escalas diferentes.

Para la construcción de un sistema de red neuronal que permita una clasificación correcta fue necesario:

1) Determinar una configuración adecuada de la red
2) Entrenar la red
3) Utilizar el entrenamiento anterior con la muestra de testeo para determinar una medida de desempeño, la que se obtiene a través de varias simulaciones.

Se probaron distintas tipologías de redes neuronales con diferentes funciones de activación, logrando mejores resultados con una capa oculta con función de activación tangente y la capa de salida con función de activación logística, como se muestra en el gráfico 4.

Figura 2: Modelo de red seleccionado



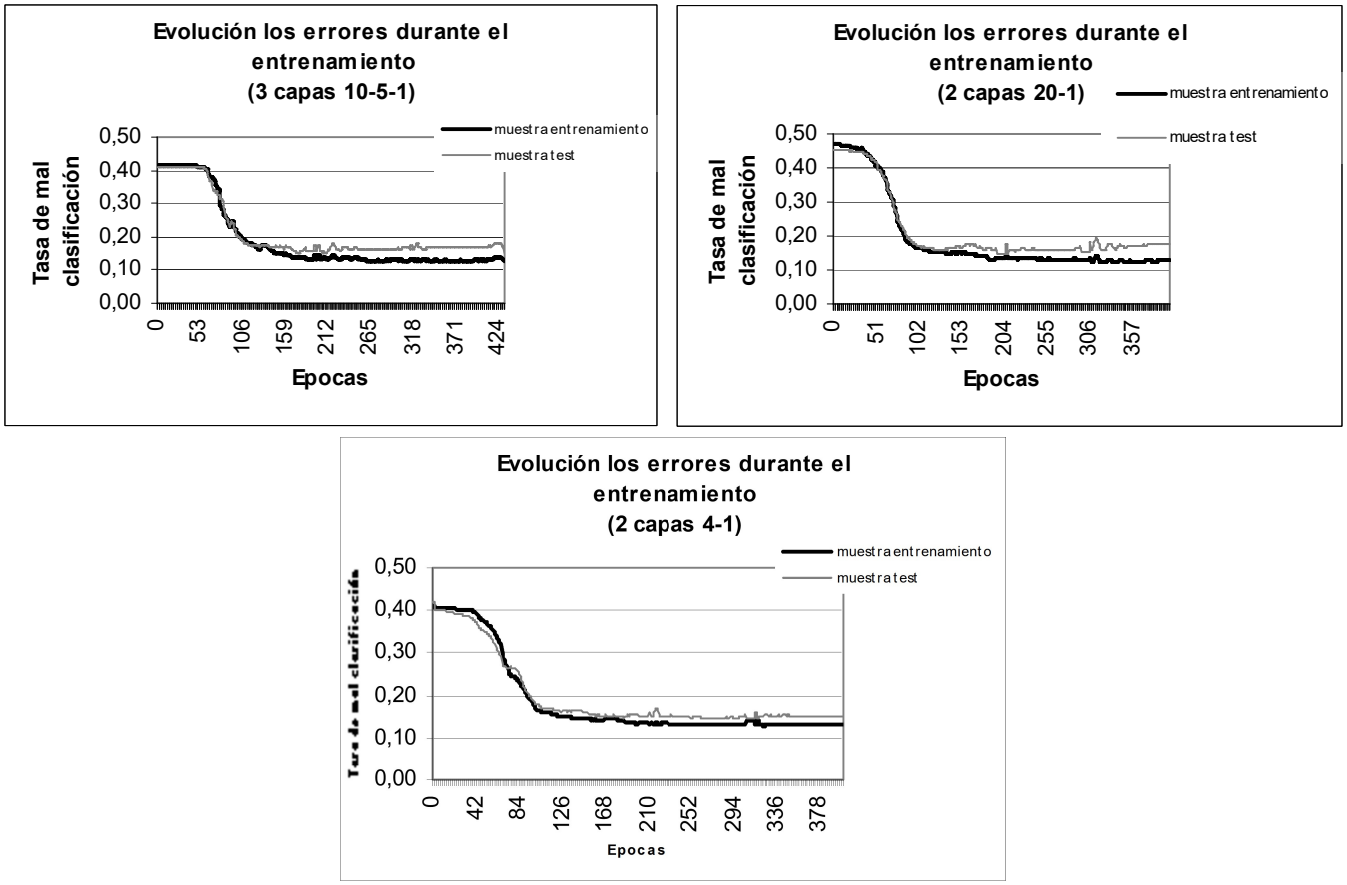
Variando el número de nodos en la capa intermedia y partiendo de diferentes valores iniciales aleatorios, se obtuvieron los siguientes resultados promedios, en la muestra test .

Tabla 3: Resultados de un Perceptron Multicapa con una capa oculta, variando la cantidad de nodos en la capa intermedia

Cantidad de nodos en la capa intermedia	Porcentaje promedio de clasificación correcta en la muestra test	Desvío estándar del porcentaje de clasificación en la muestra test
3	15,20	0,46
4	**14,96**	0,39
5	15,05	0,43
6	15,14	0,46
10	15,24	0,40

La mejor tasa de clasificación se obtuvo con 4 nodos en la capa intermedia. Se realizaron pruebas aumentado la cantidad de nodos en la capa intermedia, y agregando otra capa oculta, pero el incremento de parámetros no mejoró el resultado obtenido a partir de una red con una capa intermedia con cuatro nodos. En el gráfico 5 se muestra la evolución de los errores durante el entrenamiento de distintas tipologías de redes (la cantidad de capas indicadas en cada gráfico no considera la capa de entrada de la red), observando un comportamiento muy similar en todas.

Figura 3: Evolución de los errores durante el entrenamiento de distintas tipologías de redes



Bibliografía

[1] Anderson, T.W. (1984, 2da Edición). *An Introduction to Multivariate Statistical Analysis.* 1ra Ed. 1958. New York, Wiley.

[2] Baum, E.D. yHussler, D. (1989) *"What size net give valid generalization?"* Neural computation 1, 151-160.

[3] Bishop, Cristopher M. (1995) *Neural Networks for Pattern Recognition.* Claredon Press Oxford.

[4] Bridle, J.S, (1990) *Probabilistic interpretation of feedforward classification networks outputs, with relationships to statistical pattern recognition.* En F. Fogelman Soulié y J Hérault (Eds.), Neurocomputing: Algorithms, Architectures and Aplicattions, pp 227-236. New York: Springer-Verlag.

[5] Cox, D.R. y Snell, E.J. (1989), *The Analysis of Binary Data*, Second Edition, London: Chapman and Hall.

[6] Freeman James A. y Skapura David M.(1993). *Redes neuronales. Algoritmos, aplicaciones y técnicas de programación.*Versión en español. Copublicación de Addison Wesley Iberoamericana, S.A. y Ediciones Diaz Santos S.A.

[7] Fukunaga, K. (1990) *Statistical Pattern Recognition*, 2da Ed San Diego, CA: Academic Press.

[8] Hand, D.J.(1999)*Construccion y Assessment of Classification Rules.*Chichester, Wiley.

[9] Haykin, Simon (1994) *Neural Networks a comprehensive foundation.* Macmillan College Publihing Company

[10] Hertz Jhon, Krogh A. y Palmer Richard G.(1991) *Introduction to the theory of the neural computation.* Addison-Weslwy Publishing Company.

[11] Hosmer, D.W, Jr. and Lemeshow, S. (1989), *Applied Logistic Regression*, New York: John Wiley & Sons, Inc.

[12] Johnson, R.A. y Wichern, D.W. (1992, 3ra Ed.) *Applied Multivariate Statistical Analysis.* 1ra Ed. (1982). New York, Prentice-Hall.

[13] McCullagh y Nelder (1989).Generalized Linear Models. 2da Ed.Chapman and Hall, Londres.

[14] MüllerB. Reinhardt J.(1991), *Neural Networks.* Verlag Berlin Heidelberg (2da edición)

[15] Nagelkerke, N.J.D. (1991), "A Note on a General Definition of the Coefficient of Determination," Biometrika, 78, 691 -692.

[16] Nguyen, D.; Widrow, B.*"Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights*" Neural Networks, 1990., 1990 IJCNN International Joint Conference on 17-21, June 1990. Pages:21 - 26 vol.3.

[17] Ripley B.D. (1999) *Pattern recognition and Neural Networks* University of Cambridge.