

USO DE REDES NEURONALES EN CLASIFICACIÓN MULTIESPECTRAL

A. TOBAR, C. HERNANDEZ, R. Balsa Y J. ARIAS
Dpto. Física Aplicada. Universidad de Santiago. Santiago de Compostela

RESUMEN

Se presenta una aplicación de redes neuronales a la clasificación de imágenes multiespectrales. Para ello se diseña una red en topología perceptrón multicapa con una capa oculta. Se procede a su entrenamiento mediante la selección de patrones y posteriormente se efectúa la clasificación sobre la imagen test. Por último se efectúa una comparación entre los resultados obtenidos sobre los patrones y sobre la imagen test.

ABSTRACT

We present an application of neural networks in the field of multispectral image classification. Our design consists of a multilayer perceptron with a hidden layer. Once the net has been trained with suitable patterns, we perform a classification of a test image and compare the obtained results with the patterns and with the test image.

1. INTRODUCCIÓN

Las redes neuronales están inspiradas en la estructura neurofisiológica del cerebro humano. Su estructura altamente conexionista, su capacidad de procesado en paralelo y su facilidad de aprendizaje deben ser, pues, inherentes a la analogía establecida (Khanna, 1990).

El espectacular desarrollo de la teoría neuronal y su uso en múltiples campos como el reconocimiento y reconstrucción de patrones, la clasificación o segmentación de imágenes (señales en general), la optimización de procesos, etc., no sería posible sin la aparición de nuevos algoritmos de aprendizaje más rápidos y potentes que los inicialmente propuestos al comienzo del estudio de las redes neuronales.

En el caso de la clasificación, la topología diseñada para la red neuronal tendrá que ser entrenada para poder discernir entre una serie de categorías que se le irán presentando. Cuando las muestras de entrenamiento sean reconocidas con un índice de aciertos lo suficientemente aceptable, la red estará en condiciones de poder clasificar cualquier señal que pueda ser asimilada a alguna de las clases conocidas por la red.

En el presente trabajo nos proponemos realizar la clasificación de una subimagen TM multiespectral de 4 bandas. Este carácter multiespectral condicionará la topología de la red, principalmente en su entrada. Nuestro propósito es evaluar la eficiencia en el entrenamiento de la red, y posteriormente el grado de aciertos alcanzado en la clasificación extendida a la subimagen.

2. TOPOLOGÍA DE LA RED

De las diversas topologías descritas en la literatura y los posibles clasificadores neuronales, nos hemos decidido por la topología del perceptrón multicapa. Esta elección está condi-

cionada por su extendido uso, por la existencia de algoritmos de aprendizaje sólidamente probados y por la simplicidad conceptual de su estructura íntima de trabajo, el perceptrón (Rosenblatt, 1961).

La topología de una red neuronal en perceptrón multicapa se muestra genéricamente en la Figura 1. En ella se aprecia que los nodos de la red se distribuyen en capas de tal modo que todas las neuronas de una capa se comunican con las de la capa siguiente, y así de modo sucesivo. A cada conexión entre neuronas se le asigna un factor de peso que varía durante el aprendizaje y quedará fijo una vez finalizado éste.

La topología requiere fijar el número de capas y el número

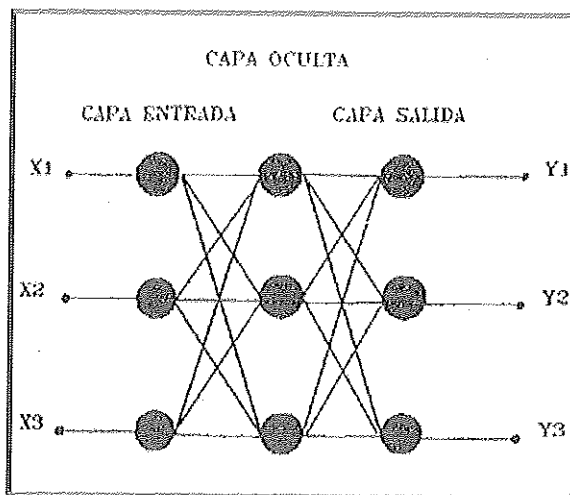


Figura 1.- Topología de una red perceptrón multicapa.

ro de nodos por capa. Dado que con tres capas en una red con alimentación positiva es posible generar regiones de decisión lo complejo que se requiera (Lippman, 1987), hemos procedido a fijar tres capas para nuestra red.

El número de nodos por capa viene influenciado por el número de entradas a la red y el número de salidas posibles. En nuestro estudio tenemos cuatro bytes de entrada (uno por banda TM) que hemos codificado en binario mediante 32 neuronas en la capa de entrada de la red. En el proceso de entrenamiento fijamos en 9 el número de clases a reconocer, por lo que la salida se puede codificar en binario mediante 4 neuronas en la última capa. Para la capa oculta, hemos elegido arbitrariamente 20 neuronas teniendo en cuenta simplemente que su número esté entre los de las capas de entrada y de salida.

3. ALGORITMO DE APRENDIZAJE

Para el entrenamiento de la red hemos usado la Regla Delta Generalizada o algoritmo de "backpropagation" de los errores (Rumelhart, 1986). En este algoritmo se busca minimizar el error cuadrático medio entre la salida obtenida del perceptrón multicapa y la salida deseada. Al estar basado en las técnicas de gradiente se requiere que las no linealidades puedan convertirse en continuas y diferenciables. Por ello, como función de transferencia de la salida de un perceptrón se asume la función sigmoide:

$$f(\alpha) = \frac{1}{1 + e^{-(\alpha)}}.$$

donde θ es el umbral de excitación para la neurona, y α la entrada a la neurona.

Los pasos de este algoritmo iterativo se explican a continuación. En la notación, los superíndices hacen relación a la capa en que está la neurona y los subíndices a la neurona en la capa especificada.

- 1.- Inicializar los pesos o coeficientes de unión entre neuronas w_{ij} y los umbrales de excitación para cada neurona.
- 2.- Presentar a la red el patrón de entrada, d_j y la salida deseada asociada y_j .
- 3.- Calcular la salida que con los coeficientes actuales obtiene la red:

$$y_{pj} = f(\text{net}^1_j), \text{ con } f \text{ la función sigmoide}$$

$$\text{net}^1_j = \sum_{i=1}^{l-1} w^1_{ji} y_i^{l-1}$$

- 4.- Adaptar los pesos si la salida obtenida no es igual a la salida deseada:

$$w^1_{ji}(t+1) = w^1_{ji}(t) + \mu \delta^1_j y_i^{l-1}, \text{ donde}$$

$$\delta^1_j = y^1_j (d_j - y^1_j) (1 - y^1_j), \text{ si se trata de la capa de salida,}$$

$$\delta^1_j = y^1_j (1 - y^1_j) \sum_k \delta^{l+1}_k w^{l+1}_{kj}, \text{ si se trata de una capa oculta.}$$

- 5.- Examinar el error cometido. Si el error cometido es superior a un valor predefinido, volver al paso 3. El umbral de error fue fijado arbitrariamente en 0,0001.

4. RESULTADOS

Sometimos la red neuronal al proceso de aprendizaje de clases espectrales relativamente bien diferenciadas en la escena de trabajo TM y en las bandas 2, 3, 4 y 5 de dicho sensor. Para asegurar una cierta homogeneidad en las muestras de cada clase, previamente se había realizado una clasificación no supervisada por técnicas clásicas. Como resultado se encontraron 9 clases lo suficientemente separables como se puede apreciar en las Tablas 1 y 2, donde se dan los centros de los "clusters" hallados y su matriz de divergencias transformadas asociadas respectivamente.

Las muestras seleccionadas para el entrenamiento, y para cada clase, se eligieron de modo que estuviesen alrededor de más-menos una desviación standard del centro de cada "cluster". Una vez seleccionadas, y acabado el proceso de aprendizaje, sobre 2754 patrones pertenecientes a todas las clases, el porcentaje de reconocimiento correcto de los patrones quedó establecido en el 98,74%. Extendida la clasificación a la subescena completa los resultados se muestran en la Tabla 3 en comparación con los de los patrones.

Como se puede apreciar en la Tabla 3, los resultados difieren ampliamente para algunas clases entre los patrones o áreas de entrenamiento (y aprendizaje) y la imagen test completa. Esto puede ser debido a las restricciones impuestas sobre los patrones: sólo se ha considerado un área de entrenamiento por clase, lo que equivale a que no haya subclases dentro de cada clase; y dentro de éstas, únicamente aquellos pixels dentro de una desviación standard al centro del "cluster" al que perteneciesen. Otro factor que sin duda influye en el resultado es el natural solape entre clases espectrales que dificulta las decisiones que debe de tomar la red en casos ciertamente dudosos.

Tabla 1.- Centros de los "clusters" de la clasificación no supervisada.

	Banda 2	Banda 3	Banda 4	Banda 5
Clase 1	26.4	21.8	14.4	13.6
Clase 2	25.0	22.0	67.9	47.9
Clase 3	27.6	24.6	85.2	65.8
Clase 4	31.7	27.6	111.2	79.9
Clase 5	37.3	37.5	96.7	103.1
Clase 6	43.7	50.5	83.4	136.0
Clase 7	40.5	44.9	74.7	100.1
Clase 8	73.2	90.9	102.0	159.2
Clase 9	84.3	97.6	91.2	122.6

Tabla 2.- Matriz de divergencias transformadas.

Clase 1	0								
Clase 2	2000	0							
Clase 3	2000	1484	0						
Clase 4	2000	1983	1580	0					
Clase 5	2000	1999	1987	1498	0				
Clase 6	2000	1999	1999	1992	1502	0			
Clase 7	2000	1986	1873	1698	1360	1327	0		
Clase 8	2000	2000	1999	1999	1956	1492	1966	0	
Clase 9	2000	2000	2000	2000	1999	1999	1999	1941	0

5. CONCLUSIONES

La simulación software de una red neuronal pone de manifiesto el potencial uso de las mismas para el problema de la clasificación multiespectral. La topología de perceptrón multicapa muestra una gran potencialidad como reconocedor de patrones aunque la complejidad computacional de la Regla Delta Generalizada es grande.

El principal inconveniente achacable a las redes es el gran tiempo computacional requerido para la fase de entrenamiento. Durante ésta se puede llegar a una situación de oscilación en la cual la red no es capaz de prosperar en su aprendizaje, por lo que se deben tomar medidas en cuanto a que la variación en los coeficientes de peso no sea demasiado brusca. Por otro lado, un factor decisorio es la presentación de los patrones. Es de suponer que con la presentación de patrones difíciles de reconocer y un índice asociado entre 0 y 1 de semejanza al patrón ideal, la red tenga mayores po-

Tabla 3.- Porcentaje de reconocimientos de patrones e imagen test.

	Patrones de entrenamiento	Imagen test
Clase 1	100.0	90.08
Clase 2	85.71	54.92
Clase 3	100.0	81.07
Clase 4	98.04	62.85
Clase 5	99.67	85.58
Clase 6	99.14	76.79
Clase 7	93.26	60.76
Clase 8	100.0	52.14
Clase 9	99.85	49.57
TOTAL	98.74	73.86

sibilidades de reconocimiento bajo circunstancias como el solape entre clases. Esto será objeto de subsiguientes experimentaciones.

Una de las ventajas frente a los clasificadores tradicionales como el gaussiano de máxima probabilidad es que la red no tiene que conocer a priori las distribuciones que siguen las distintas clases y que no siempre es posible conocer. Así, si el proceso de entrenamiento y aprendizaje converge en un tiempo razonable, las redes pueden ser más apropiadas ante el desconocimiento de las distribuciones y de la importancia relativa de una banda frente a otra.

6. BIBLIOGRAFIA

- ✓ LIPPMAN, R. (1987): An introduction to computing with neural nets. *IEEE ASSP Magazine*. Vol. 2. 4-22.
- ✓ KHANNA, T. (1990): *Foundations of neural networks*. Addison-Wesley. New York.
- ✓ ROSENBLAT, B. (1961): *Principles of neurodynamics: Perceptrons and the theory of brain. Mechanisms*. Spartan Press. Washington DC.
- ✓ RUMELHART, D.e.; HINTON, G.E. & WILLIAMS, R.J. (1986): Learning internal representations by errors propagation. In *Parallel distributed processing: Explotations in the microstructure of cognitron*. Vol. 1. D.E. Rumelhart adn J.L. McClelland, Cap. 8. MIT Press.
- ✓ KOHONEN, T. (1988): An introduction to neural computing. *Neur Networks*. Vol. 1, nº1, 3-16.