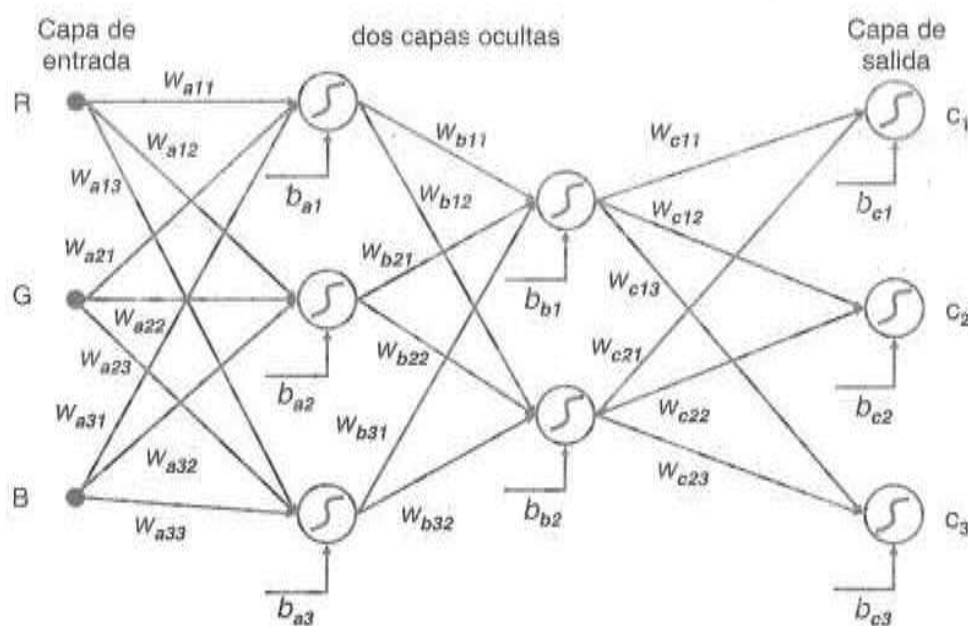


***Ejercicio 9.6** Un sistema basado en visión por computador trata de clasificar tres tipos de texturas naturales basadas en las tres componentes de color (R,G,B), que se corresponden con regiones de cielo, bosque y suelo. Se han extraído los datos de la imagen dados a continuación. Diseñar un modelo de red neuronal multicapa con entrenamiento por retropropagación para el aprendizaje. A continuación determinar a qué clase pertenecen los píxeles con los valores dados: $A = (209, 169, 131)$, $B = (89, 133, 60)$ y $C = (152, 140, 111)$.

		c_1			c_2			c_3	
R	208	210	212	90	92	87	160	154	148
G	165	170	172	130	138	128	143	146	137
B	128	130	133	60	59	62	113	115	110

Solución:

La red debe tener 3 neuronas en la capa de entrada puesto que los vectores de datos poseen 3 componentes. Diseñamos una arquitectura de red con 2 capas ocultas y 3 y 2 neuronas respectivamente en cada capa. Finalmente asumimos que la salida posee 3 neuronas para hacerlas corresponderse con las tres clases. Esta arquitectura se muestra en la figura siguiente,



suponiendo una razón de aprendizaje $\alpha = 0.5$. En todas las neuronas se utilizan las funciones logísticas de activación sigmoidal. Los valores de los vectores de entrada (R, G, B) se normalizan al rango $[0,1]$ dividiendo los valores originales por 255. La capa de salida posee tres neuronas con valores 0 ó 1 asociándose con los siguientes patrones de entrada: $(1,0,0)$ con patrones de la clase c_1 , $(0,1,0)$ con patrones de la clase c_2 y $(0,0,1)$ con patrones de la clase c_3 . Los pesos y bias entre la capa de entrada y la primera capa oculta se identifican como W_a , b_a , entre las dos capas ocultas W_b , b_b y finalmente entre la última capa oculta y la de salida W_c , b_c . Los pesos y bias iniciales con los que se comienza el proceso de aprendizaje son los que se muestran a continuación. Según esta nomenclatura,

las filas con valores w_{ij} indican el valor del peso de la conexión que une la neurona i con la j .

$$W_a = \begin{bmatrix} w_{a11} & w_{a21} & w_{a31} \\ w_{a12} & w_{a22} & w_{a32} \\ w_{a13} & w_{a23} & w_{a33} \end{bmatrix} \equiv \begin{bmatrix} -2.89 & 22.02 & -24.07 \\ 12.34 & -13.82 & -16.50 \\ 9.31 & -37.61 & -5.19 \end{bmatrix}$$

$$W_b = \begin{bmatrix} w_{b11} & w_{b21} & w_{b31} \\ w_{b12} & w_{b22} & w_{b32} \end{bmatrix} \equiv \begin{bmatrix} 5.26 & 4.66 & -0.63 \\ -1.08 & -4.34 & -5.46 \end{bmatrix}$$

$$W_c = \begin{bmatrix} w_{c11} & w_{c21} \\ w_{c12} & w_{c22} \\ w_{c13} & w_{c23} \end{bmatrix} \equiv \begin{bmatrix} -3.86 & -8.90 \\ 6.45 & -7.24 \\ -9.03 & 3.53 \end{bmatrix}$$

$$b_a(1) = 1.84; \quad b_a(2) = 7.11; \quad b_a(3) = 22.66; \quad b_b(1) = -8.17; \quad b_b(2) = 1.91$$

$$b_c(1) = 11.23; \quad b_c(2) = 0.3954; \quad b_c(3) = -2.0975$$

Con estos valores se inicia el proceso de entrenamiento, fijando como número máximo de iteraciones $1.2e^6$. El límite del error se fija en 0.1. Tras el entrenamiento se obtienen los siguientes valores para los pesos (parámetros aprendidos),

$$W_a = \begin{bmatrix} -3.01 & 21.37 & -25.9 \\ 15.04 & -11.71 & -17.03 \\ 7.67 & -38.84 & -6.04 \end{bmatrix} \quad W_b = \begin{bmatrix} 11.87 & 7.03 & -2.40 \\ -0.65 & -6.54 & -4.46 \end{bmatrix}$$

$$W_c = \begin{bmatrix} -7.80 & -8.30 \\ 14.11 & -7.85 \\ -25.09 & 2.61 \end{bmatrix}$$

Suministrando ahora los patrones A , B y C normalizados también en el rango $[0,1]$ (dividiendo por 255) se consiguen los resultados mostrados en la tabla adjunta. En dicha tabla se muestran los resultados antes y después del entrenamiento.

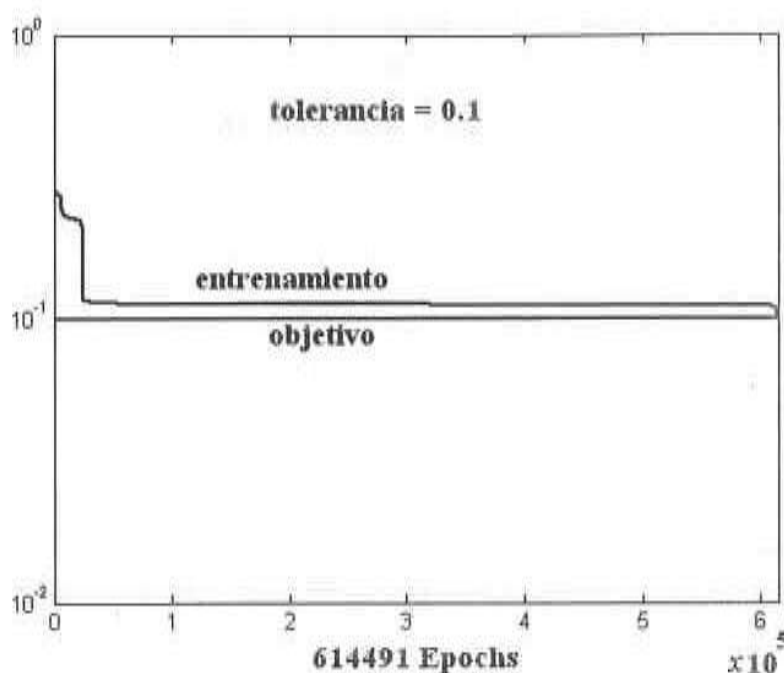
patrón	salida (c_1, c_2, c_3)	
	antes del entrenamiento	después del entrenamiento
A	(1.00, 0.71, 0.06) clase 1	(0.97, 0.01, 0.04) clase 1
B	(1.00, 0.91, 0.01) clase 1	(0.02, 0.99, 0.00) clase 2
C	(1.00, 0.72, 0.05) clase 1	(0.99, 0.00, 0.97) clase 1

En la figura siguiente se muestra la gráfica correspondiente al proceso de entrenamiento. Se observa que la convergencia para los valores de tolerancia al error no se consigue hasta la iteración 614491. Inicialmente se produce un descenso brusco para luego progresar lentamente hasta alcanzar dicha iteración. El proceso de aprendizaje finaliza con la siguiente información tras su ejecución en MATLAB.

TRAINGD, Epoch 600000/1.5e+006, MSE 0.110741/0.1, Gradient 0.000437152/1e-010

TRAINGD, Epoch 614491/1.5e+006, MSE 0.099995/0.1, Gradient 0.00668569/1e-010

TRAINGD, Performance goal met.



Los resultados del proceso de entrenamiento así como las salidas finales de la tabla pueden variar según el proceso de entrenamiento. No obstante, a la vista de los resultados de la tabla anterior se deduce que los valores de salida para los tres patrones *A*, *B* y *C* se identificarían en los tres casos con la clase 1, lo cual es incorrecto ya que realmente pertenecen a las clases 1, 2 y 3 respectivamente. Tras el entrenamiento se observa que esta anomalía se ha corregido en parte. Ahora *A*, *B* son correctamente clasificados, ya que sus valores se aproximan al valor ideal, mientras que *C* sigue sin clasificarse correctamente. Esto proviene del hecho de que el número de muestras para el entrenamiento es insuficiente y se necesitaría incrementar dicho número. Esto constituye un problema típico en los procesos de aprendizaje en general.

9.9.2 Algoritmo de Lloyd

***Ejercicio 9.7** Dado el conjunto de datos *X* de la tabla adjunta en el espacio bidimensional \mathcal{R}^2 .

x_1	1	1	1	2	2	6	6	7	7	7
x_2	1	3	5	2	3	3	4	1	3	5