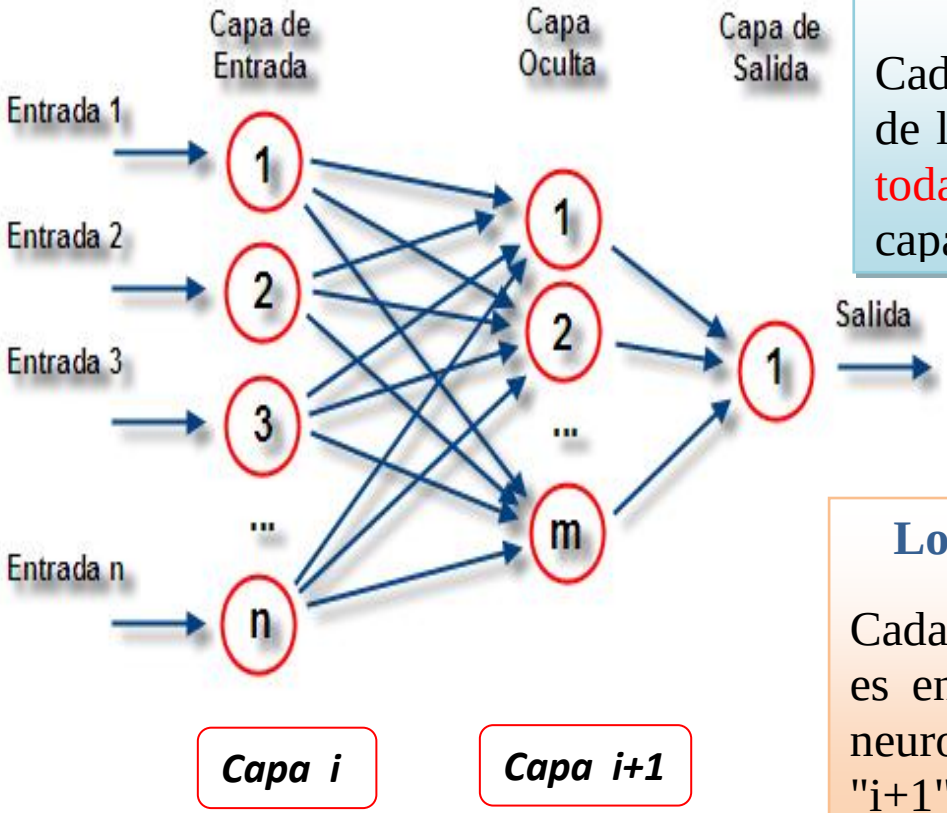

Redes Neuronales
Parte 2

En la sesión anterior vimos el comportamiento de un Perceptron simple y cómo es que podríamos re calcular los pesos de las entradas respectivas, para el aprendizaje de las mismas, bajo un cierto factor.

Pero existen situaciones de (RNAs) formadas por múltiples capas, que permite resolver problemas que no son linealmente separables, lo cual es la principal limitación del perceptrón simple).



Totalmente Conectado

Cada salida de una neurona de la capa "i" es entrada de todas las neuronas de la capa "i+1"

Localmente conectado

Cada neurona de la capa "i" es entrada de una serie de neuronas (región) de la capa "i+1".

Clasificación de Capas

Capa de entrada: Son las neuronas que introducen los patrones de entrada en la red. En estas neuronas no se produce procesamiento.

Capas ocultas: Aquellas neuronas cuyas entradas provienen de capas anteriores y cuyas salidas pasan a neuronas de capas posteriores.

Capa de salida: Neuronas cuyos valores de salida se corresponden con las salidas de toda la red.

RETROPROPAGACION

La propagación hacia atrás (también conocido como retropropagación del error o regla delta generalizada), es un algoritmo utilizado en el entrenamiento de estas redes, por ello, el perceptrón multicapa también es conocido como **red de retropropagación.**.

Limitaciones

El Perceptron Multicapa no extrapola bien, es decir, *si la red se entrena mal o de manera insuficiente*, las salidas pueden ser imprecisas.

La existencia de mínimos locales en la función de error dificulta considerablemente el entrenamiento, pues una vez alcanzado un mínimo el entrenamiento se detiene aunque no se haya alcanzado la tasa de convergencia fijada.

Cuando caemos en un mínimo local sin satisfacer el porcentaje de error permitido se puede considerar:

> *Cambiar la topología de la red (número de capas y número de neuronas), comenzar el entrenamiento con unos pesos iniciales diferentes, modificar los parámetros de aprendizaje, modificar el conjunto de entrenamiento o presentar los patrones en otro orden.*

Los MLP, MultiLayer Perceptron) se utiliza para resolver problemas de asociación de patrones, segmentación de imágenes, compresión de datos, etc.

¿Las Rnas hacen lo que les da la gana?

En los 80 los militares norteamericanos se gastaron un buen montón de millones de dólares en un proyecto para incorporar en sus tanques un sistema de detección de tanques enemigos. El sistema consistía en una cámara conectada a un ordenador que iba analizando constantemente las imágenes y avisaba en el caso de que detectase un tanque.

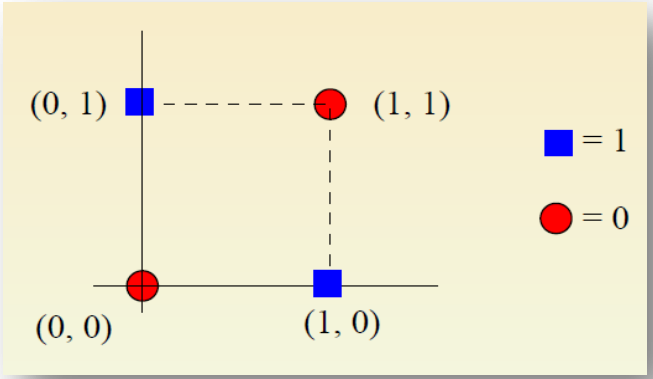
Para este problema decidieron usar una red neuronal. La entrenaron con 200 fotografías; 100 fotos de tanques ocultos tras arboledas y 100 de árboles sin tanques.

Hasta aquí todo bien, el entrenamiento funcionó perfectamente y el sistema era capaz de clasificar las fotos del entrenamiento.

Llegó el momento de probarlo y… oh! sorpresa! El sistema fallaba por todos lados y no daba ni una!.

En primer lugar pensaron que había memorizado las fotos de entrenamiento (y por lo tanto estaría intentando recordar en lugar de generalizar), pero después de mucho investigar, comprobar, verificar, analizar y gastar dinero, se dieron cuenta de que las 100 fotos con tanques las habían sacado en días nublados y las otras 100 sin tanques en días soleados, la red había aprendido a reconocer lo que era más evidente;

> El cielo, millones de dólares para que un tanque te diga si hace sol…



INVESTIGUE UD.

¿CUANTAS SOLUCIONES EXISTEN PARA RESOLVER EL PROBLEMA XOR?

Algoritmos Genéticos

Son métodos sistemáticos para la resolución de problemas de búsqueda y optimización que aplican a estos los mismos métodos de la evolución biológica: selección basada en la población, reproducción sexual y mutación.

Hay que tener en cuenta que un algoritmo genético es independiente del problema, lo cual lo hace un algoritmo *robusto*, por ser útil para cualquier problema, pero a la vez *débil*, pues no está especializado en un problema específico.

Codificación de las variables

Los algoritmos genéticos requieren que el conjunto se codifique en un *cromosoma*. Cada cromosoma tiene varios genes, que corresponden a sendos parámetros del problema.

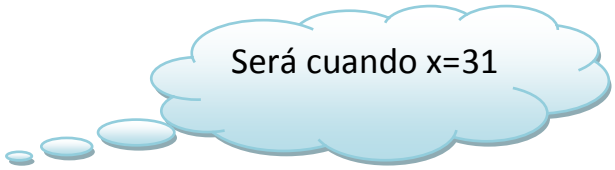
Para poder trabajar con estos genes en el ordenador, es necesario codificarlos en una *cadena*, es decir, una serie de símbolos (números o letras) que generalmente va a estar compuesta de 0s y 1s.

Ejm 1.

Tenemos la función $f(x) = X^2$

En un intervalo de x de [0; 31] Z+

¿Cuál es el valor máximo de f(x)?



Pero veamos otra manera, codificando un posible valor de X:

(0, 0, 0, 0, 0)……. 0 Es un individuo

(0, 1, 0, 1, 1)……11 Es otro individuo

(1, 1, 1, 1, 1)…… **¿? Es otro individuo**

El número de individuos **componen la población**

Ahora coja Ud. un tamaño de la población:

Por decir coja Ud. 6 individuos.

Algoritmo
Inicio
Generar la población()
Generar Individuos()
{//f=25% es la muestra
Individuos=f * población //
Componentes?
 Desde i=1 hasta el número de individuos.
       fil=i
      Desde j=1 hasta el número de componentes
                 Col=j
                componente(fil,col)= ramdom(0 ó 1)
               Fin Repertir
        Fin Repetir
  }

//Determinar los valores de X y de F(x)

GENERACION				
Nro	Individuo Binario	X	F(x)	
1	(0,1,1,0,0)	12	144	
2	(1,0,0,1,0)	18	324	
3	(0,1,1,1,1)	15	225	
4	(1,1,0,0,0)	24	576	
5	(1,1,0,1,0)	26	676	
6	(0,0,0,0,1)	1	1	

¿El mejor individuo quién es?

El individuo 5 con f = 676, es el mejor individuo de la muestra, cuya media (f) = 324.3.

//Seguimos con el algoritmo

Luego asignamos_parejas_de_individuos(), *de manera aleatoria*, con otro individuo (no consigo mismo). Este otro individuo no debe ser pareja de otro.

SELECCIÓN				
Nro	Individuo Binario	X	F(x)	Pareja
1	(0,1,1,0,0)	12	144	6
2	(1,0,0,1,0)	18	324	3
3	(0,1,1,1,1)	15	225	2
4	(1,1,0,0,0)	24	576	5
5	(1,1,0,1,0)	26	676	4
6	(0,0,0,0,1)	1	1	1

TORNEO					
Nro	Individuo Binario	X	F(x)	Pareja	Torneo
1	(0,1,1,0,0)	12	144	1-6	Gana 1
2	(1,0,0,1,0)	18	324	2-3	Gana 2
3	(0,1,1,1,1)	15	225	3-2	Gana 2
4	(1,1,0,0,0)	24	576	4-5	Gana 5
5	(1,1,0,1,0)	26	676	5-4	Gana 5
6	(0,0,0,0,1)	1	1	6-1	Gana 1

En el torneo el individuo ganador gana UN CLON, y se duplica.

Luego asignamos_parejas_de_individuos() de manera similar al proceso de selección

nro	Individuo Binario	Pareja
1	(0,1,1,0,0)	**1-5**
2	(0,1,1,0,0)	**2-3**
3	(1,0,0,1,0)	**3-2**
4	(1,0,0,1,0)	**4-6**
5	(1,1,0,1,0)	**5-1**
6	(1,1,0,1,0)	**6-4**

Y luego realizamos un el CRUCE respectivo:

Para dos individuos (pareja), establecer un punto de cruce aleatorio,.

Cruce (pareja)= int [ramdom(1; nro_componentes -1)]

nro	Individuo Binario	Pareja	Cruce
1	(0,1,1,0,0)	**1-5**	**1**
2	(0,1,1,0,0)	**2-3**	**3**
3	(1,0,0,1,0)	**3-2**	**3**
4	(1,0,0,1,0)	**4-6**	**1**
5	(1,1,0,1,0)	**5-1**	**1**
6	(1,1,0,1,0)	**6-4**	**1**

Ahora procedemos al cruce genético

De acuerdo a los que nos indique “cruce”

nro	Individuo Binario	Pareja	Cruce	Hijos
1	(0,1,1,0,0)	1-5	1	(0,1,0,1,0)
2	(0,1,1,0,0)	2-3	3	(0,1,1,1,0)
3	(1,0,0,1,0)	3-2	3	(1,0,0,0,0)
4	(1,0,0,1,0)	4-6	1	(1,1,0,1,0)
5	(1,1,0,1,0)	5-1	1	(1,1,1,0,0)
6	(1,1,0,1,0)	6-4	1	(1,0,0,1,0

De manera que nuestra población después de haber hecho los cruces respectivos será:

Nro	Individuos	X	F(x)
1	**(0,1,0,1,0)**	**10**	**100**
2	**(0,1,1,1,0)**	**14**	**196**
3	**(1,0,0,0,0)**	**16**	**256**
4	**(1,1,0,1,0)**	**26**	**676**
5	**(1,1,1,0,0)**	**28**	**784**
6	**(1,0,0,1,0**	**18**	**324**

¿Ahora quién será el mejor individuo?

El mejor individuo es 28 cuyo valor de la función es 784, con una media de 389.33333333.

CONCLUSIONES:

Respecto a la media anterior, la media actual ha subido.

El valor de la función ha mejorado, para el mejor individuo.

Genéticamente después de la selección y el cruce, los individuos son mejores, que antes de las transformaciones.

En base a la población final, iterar tantas veces como sea necesario a fin de obtener resultados óptimos.

INVESTIGUE UD.

I-¿EN QUE CONSISTE LA HOMOGENIZACION DE LA POBLACION?

II-¿CÓMO PODEMOS INTRODUCIR MUTACIONES TRAS LA SELECCIÓN Y EL CRUCE?

III-Desarrollar Max[f(x)], en un intervalo de enteros de 0 a 12

1. $F(x) = 3X^2 + 4X - 6$

2. $F(x) = X^3 - 8$

Desarrollar el Min[f(x)], en intervalos de enteros de -20 a 20

3. $F(x) = ((X-3)^2 + 16)/8)$