Redes neurais artificiais

Redes neurais são dispositivos que habilitam o computador a reconhecer padrões. Redes são formadas por neurônios distribuídos em camadas. A primeira camada corresponde à entrada e a última à saída. Pode ou não haver camadas intermediárias, conhecidas como escondidas. Neste exercício por simplicidade, não haverá camadas escondias.

Redes que tem camadas escondidas permitem uma reconstituição mais rica, pela possibilidade de reconhecerem padrões não lineares. Entretanto, precisam usar o algoritmo de *BACKPROPAGATION*, que prevê uma maneira de distribuir os créditos de cada neurônio, de trás para a frente.

As redes precisam ser treinadas, antes de entrarem em produção normal. Os dados de treinamento são idênticos (em forma) aos dados de produção. Veja-se na figura **??** um neurônio artificial, com suas entradas e saídas.

Figura 1: Um neurônio artificial

Exercício

Este exercício vai ser feito em uma rede de 8 entradas e 4 saídas, sem camadas escondidas. Com isto, fica facilitado o algoritmo de atribuição de crédito. Também por simplicidade, as entradas e as saídas serão binárias.

Suponha uma rede neural de 8 entradas e 4 saídas, que deve ser treinada para exibir o seguinte comportamento:

Entradas	Saídas
1 0 1 0 1 0 1 0	1 0 0 0
0 1 0 1 0 1 0 1	0 1 0 0
1 1 1 1 0 0 0 0	0 0 1 0
0 0 0 0 1 1 1 1	0 0 0 1

Deve-se considerar valores tanto na entrada como na saída como {-1, 1}, ao invés de {0, 1} a fim de acelerar o treinamento, já que o fator de aprendizado é dobrado.

Inicializar todos os pesos com 0.0 e o limiar de cada neurônio em 2.0

Desta maneira, eis como fica a matriz de pesos: (que é a RNA)

-2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
-2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
-2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
-2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Para este exemplo, usaremos um fator de aprendizado $c = 0.2$

Entrando 1 0 1 0 1 0 1 0 (entrada 1), que deve ser entendida como 1 -1 1 -1 1 -1 1 -1 e operada como **1**,1 -1 1 -1 1 -1 1 -1 e somada dá como resposta -2, que não é maior ou igual a zero. Para todas as linhas. Logo a resposta da rede será: -1 -1 -1 -1.

Informando ao algoritmo que o resultado correto é 1 0 0 0 (1 -1 -1 -1) e usando a fórmula: $P_n = P_v + C \times ((R_e - R_a) \times E)$, onde P_n = Nova linha de pesos
P_v = Linha antiga de pesos
C = fator de aprendizagem (parâmetro)
R_e = Resultado esperado (fornecido durante o treinamento)
R_a = Resposta achada pela rede
E = Entrada fornecida à rede,

E eis como ficará a nova matriz de pesos

-2.0	0.4	-0.4	0.4	-0.4	0.4	-0.4	0.4	-0.4
-2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
-2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
-2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

A matriz acima pode ser assim explicada:

O resultado esperado (vide dados de treinamento) era 1, -1, -1, -1 e o resultado achado pela rede foi -1, -1, -1, -1. Logo, o primeiro neurônio (que corresponde à primeira linha da matriz) errou. Os outros 3 neurônios de saída responderam certo. Eis as correções a fazer:

linha	cálculo da correção dos pesos	observação
primeira	$P_v = P_n + (0.2 \times (1 - (-1)) \times E$	E=1, -1, 1, -1, 1, -1, 1, -1
segunda	$P_v = P_n + (0.2 \times ((-1) - (-1)) \times E$	A correção vale zero
terceira	$P_v = P_n + (0.2 \times ((-1) - (-1)) \times E$	A correção vale zero
quarta	$P_v = P_n + (0.2 \times ((-1) - (-1)) \times E$	A correção vale zero

Um novo exemplo

Vamos refazer um exercicio desde o inicio, considerando: $c = 0.1$ *e* $B = -4.0$ $Entrada_1 = 00001111$, Resultado correto: 0001 Com isto,

-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	-.2	-.2	-.2	-.2	.2	.2	.2	.2

$Entrada_2 = 11110000$, Resultado correto: 0010 Com isto,

-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	.2	.2	.2	.2	-.2	-.2	-.2	-.2
-4.0	-.2	-.2	-.2	-.2	.2	.2	.2	.2

$Entrada_3 = 11110000$, Resultado correto: 0010

-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	.4	.4	.4	.4	-.4	-.4	-.4	-.4
-4.0	-.2	-.2	-.2	-.2	.2	.2	.2	.2

$Entrada_4 = 10101010$, Resultado correto: 1000

-4.0	.2	-.2	.2	-.2	.2	-.2	.2	-.2
-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	.4	.4	.4	.4	-.4	-.4	-.4	-.4
-4.0	-.2	-.2	-.2	-.2	.2	.2	.2	.2

$Entrada_5 = 10101010$, Resultado correto: 1000

-4.0	.4	-.4	.4	-.4	.4	-.4	.4	-.4
-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	.4	.4	.4	.4	-.4	-.4	-.4	-.4
-4.0	-.2	-.2	-.2	-.2	.2	.2	.2	.2

$Entrada_6 = 10101010$, Resultado correto: 1000

-4.0	.6	-.6	.6	-.6	.6	-.6	.6	-.6
-4.0	.0	.0	.0	.0	.0	.0	.0	.0
-4.0	.4	.4	.4	.4	-.4	-.4	-.4	-.4
-4.0	-.2	-.2	-.2	-.2	.2	.2	.2	.2

Até aqui, estivemos na fase de treinamento da rede (esta fase ainda não está terminada...) Depois disso, é que a rede é considerada treinada e pode passar a responder a casos novos iguais ou diferentes aos que ela resolveu no treinamento.

☞ Para você fazer

Fase de treinamento

$B = -2.0$ *e* $c = 0.3$
Entrada 1 = 10101010 , Resultado correto = 1000
Entrada 2 = 00001111 , Resultado correto = 0001
Entrada 3 = 00001111 , Resultado correto = 0001
Entrada 4 = 00001111 , Resultado correto = 0001
Entrada 5 = 01010101 , Resultado correto = 0100
Entrada 6 = 11110000 , Resultado correto = 0010

Aplique as correções nos pesos do aprendizado e responda: Obs: o elemento coluna 1 é o bias. A matriz de pesos tem 9 colunas... Quais os pesos achados

1. [1;3]_____ 2. [2;4] _____

3. [3;5]_____ 4. [4;6] _____

Fase de Produção

5. O que seria reconhecido na entrada:
01100101 _____·_____·_____ ·_____