



Redes Neurais Feedforward e Backpropagation

André Siqueira Ruela



UFOP

Universidade Federal
de Ouro Preto



decom
departamento
de computação

Sumário

- ▶ Introdução a redes feedforward
- ▶ Algoritmo feedforward
- ▶ Algoritmo backpropagation



UFOP

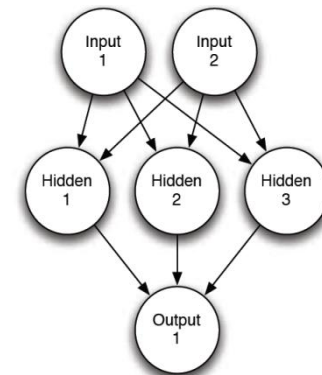
Universidade Federal
de Ouro Preto

SEVA  **Mobilis**

 **decom**
departamento
de computação

“Feedforward”

- ▶ Em uma rede feedforward, cada camada se conecta à próxima camada, porém não há caminho de volta.
- ▶ Todas as conexões portanto, têm a mesma direção, partindo da camada de entrada rumo a camada de saída.



“Backpropagation”

- ▶ O termo backpropagation define a forma com que a rede é treinada.
- ▶ O algoritmo backpropagation se trata de um treinamento supervisionado.
- ▶ Com o erro calculado, o algoritmo corrige os pesos em todas as camadas, partindo da saída até a entrada.



UFOP

Universidade Federal
de Ouro Preto

SEVA  **Mobilis**

 **decom**
departamento
de computação

Redes Feedforward

- ▶ Existem muitas formas de se estruturar uma rede feedforward.
- ▶ Deve se definir:
 - Camada de Entrada.
 - Camada(s) Intermediária(s).
 - Camada de Saída.
- ▶ Apesar de haverem técnicas básicas, alguns experimentos são necessários para se definir a estrutura ótima da rede.



UFOP
Universidade Federal
de Ouro Preto

SEVA  **Mobilis**


decom
departamento
de computação

Camada de Entrada

- ▶ A camada de entrada recebe um padrão que será apresentado a rede neural.
- ▶ Cada neurônio na camada de entrada deve representar alguma variável independente que influencia o resultado da rede neural.
- ▶ Em nossa implementação, cada entrada corresponde a um valor “double”, normalizado.

Camada de Saída

- ▶ A camada de saída retorna um padrão ao ambiente externo.
- ▶ O número de neurônios está diretamente relacionado com a tarefa que a rede neural executa.
- ▶ Em geral, o número de neurônios que um classificador deve possuir, nesta camada, é igual ao número de grupos distintos.

OCR: Entrada e Saída

- ▶ OCR: *Optimal Character Recognition*.
- ▶ Suponha uma rede neural que reconheça as 26 letras do alfabeto inglês.
- ▶ Cada padrão a ser classificado consiste em uma imagem de 35 pixels (5x7).
- ▶ A imagem consiste na entrada, sendo cada pixel representado por um respectivo neurônio. Portanto, 35 neurônios na entrada.
- ▶ A letra reconhecida é representada pelo neurônio que dispara alguma saída positiva. Portanto, 26 neurônios na saída.



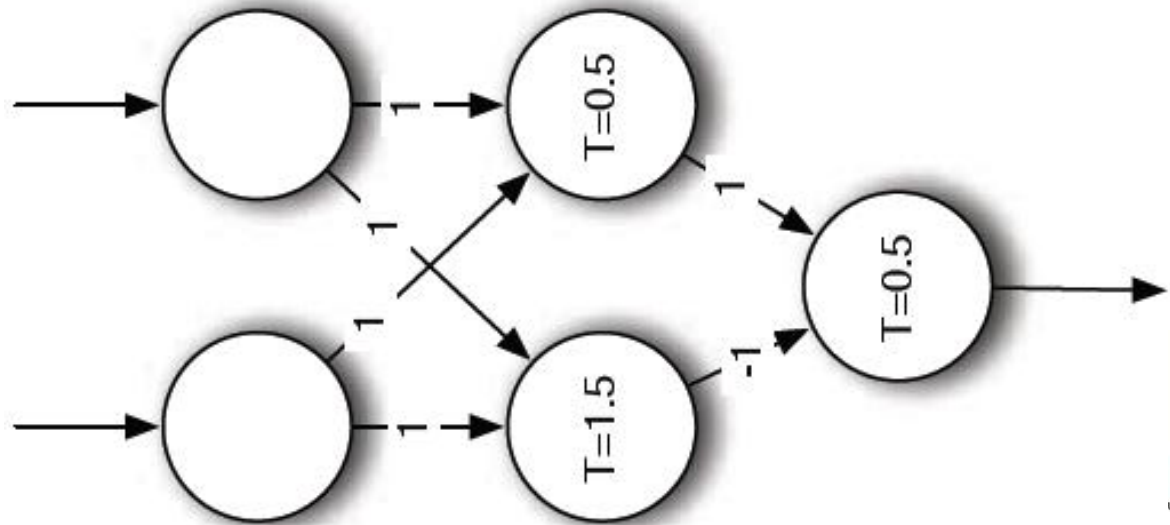
UFOP
Universidade Federal
de Ouro Preto

SEVA  **Mobilis**


decom
departamento
de computação

Exemplo: Operador XOR

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0



Funções de Ativação

- ▶ Métodos de treinamento como o backpropagation exigem que a função de ativação possua derivada.

Equation 5.1: The Sigmoid Function

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

$$f'(x) = x(1 - x)$$

Equation 5.2: The TANH Function

$$f(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$$

$$f'(x) = 1 - x^2$$



UFOP
Universidade Federal
de Ouro Preto

SEVA  **Mobilis**


decom
departamento
de computação

O número de camadas intermediárias

- ▶ Problemas que requerem duas camadas intermediárias são raros.
- ▶ Basicamente, uma rede neural com duas camadas intermediárias é capaz de representar funções de qualquer formato.
- ▶ Não existem razões teóricas para se utilizar mais que duas camadas intermediárias.
- ▶ Na prática, para muitos problemas comuns, não há razões para se utilizar mais que uma camada intermediária.



Universidade Federal
de Ouro Preto

SEVA  **Mobilis**

 **decom**
departamento
de computação

O número de neurônios nas camadas intermediárias

- ▶ Apesar de não interagirem com o ambiente externo, as camadas intermediárias exercem enorme influência no funcionamento da rede.
- ▶ *Underfitting*: poucos neurônios que não conseguem detectar adequadamente os sinais em um conjunto complicado de dados.
- ▶ *Overfitting*: muitos neurônios a serem treinados por um número limitado de informação contida no conjunto de dados.
- ▶ Muitos neurônios podem fazer com que o treinamento não termine adequadamente em tempo hábil.

O número de neurônios nas camadas intermediárias

- ▶ Regrinhas clássicas... O número de neurônios na camada intermediária deve ser:
 1. Entre o número de neurônios nas camadas de entrada e saída.
 2. $2/3$ do tamanho da camada de entrada, somado ao tamanho da camada de saída.
 3. Menor que duas vezes o tamanho da camada de entrada.



UFOP
Universidade Federal
de Ouro Preto

SEVA  **Mobilis**

 **decom**
departamento
de computação

Examinando o processo Feedforward

- ▶ Feedforward: computeOutputs(...)
 - Apresenta a rede um vetor de entrada.
 - Chama o método computeOutputs de cada camada.
 - Retorna o valor disparado pela camada de saída.

- ▶ FeedforwardLayer: computeOutputs(...)
 - Calcula o somatório dos produtos entre entradas e pesos.
 - Define os disparos de cada neurônio da próxima camada por meio da função de ativação.



UFOP
Universidade Federal
de Ouro Preto

SEVA  **Mobilis**


decom
departamento
de computação

Examinando o processo Backpropagation

- ▶ A função que calcula o erro precisa ser diferenciável, assim como a função de ativação.
- ▶ Através da derivada da função do erro, pode-se encontrar os pesos que minimizam o erro.
- ▶ Um método comum é a descida do gradiente.



UFOP
Universidade Federal
de Ouro Preto

SEVA  **Mobilis**


decom
departamento
de computação

Examinando o processo Backpropagation

- ▶ Backpropagation: iteration()
 - Percorre todos os conjuntos de treinamento.
 - Cada conjunto é apresentado a rede e o erro é calculado.
 - Após apresentar todos os conjuntos de treinamento, o método learn() é chamado.
 - Por fim, o erro global da rede é calculado.



UFOP
Universidade Federal
de Ouro Preto

SEVA  **Mobilis**


decom
departamento
de computação

Examinando o processo Backpropagation

- ▶ **Backpropagation: calcError()**
 - Limpa todos os erros anteriores nas camadas da rede.
 - A partir da saída, percorre todas as camadas da rede, até a camada de entrada.

- ▶ **BackpropagationLayer: calcError(...)**
 - Acumula uma matriz de deltas.
 - Determina a contribuição da camada para o erro observando os deltas da próxima camada e os disparees da camada atual.
 - Utiliza a derivada das funções de ativação para o cálculo do delta.



Universidade Federal
de Ouro Preto

SEVA  **Mobilis**


decom
departamento
de computação

Examinando o processo Backpropagation

- ▶ Backpropagation: learn()
 - Chamado após o cálculo de todos os erros, este método aplica os deltas aos pesos e limiares.
 - Utiliza a taxa de aprendizado e o momentum para definir como as mudanças serão aplicadas.



UFOP
Universidade Federal
de Ouro Preto

SEVA  **Mobilis**


decom
departamento
de computação