Eduardo Lobo Lustosa Cabral

AULA 5
GRADIENTE DESCENDENTE
PROBLEMA DE CLASSIFICAÇÃO BINÁRIA – RNA RASA

1. Objetivos

- Definir um problema simples de classificação binária usando uma RNA rasa (RNA de uma camada intermediária).
- Detalhar o método do Gradiente Descendente e o algoritmo de Retro-Propagação para um problema de classificação binária.
- Apresentar a vetorização do cálculo da propagação para frente e da retro-propagação para todos os exemplos.

2. Descrição do problema

- Nesse exemplo queremos determinar se um ponto no plano pertence ou não a uma determinada classe, que é definida pela sua posição no plano.
- Esse problema consiste de um problema simples de reconhecimento de padrão, que é um dos problemas usualmente resolvidos com RNAs.
- A Figura 1 apresenta os pontos do plano com as duas classes. Cada classe é definida pela cor azul ou vermelha. Note que cada pétala da flor possui pontos de uma cor predominante, ou seja, cada pétala é representante de uma das classes.

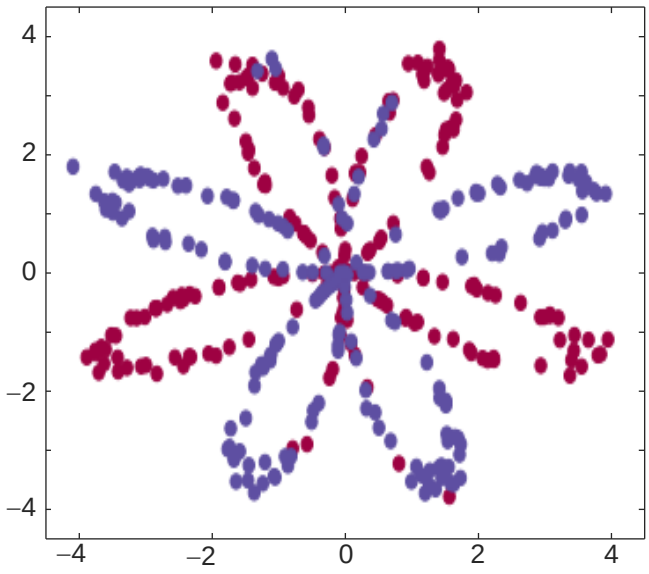

Figura 1. Problema de classificação binária de pontos no plano (Andrew Ng, deeplearning.ai).

- Esse problema de classificação consiste em: **dadas as coordenadas de um ponto no plano prever se ele é da classe azul ou vermelha**.

3. Definição do problema de classificação de padrão binária

Dados de treinamento

- Um elemento do conjunto de treinamento consiste do vetor $\mathbf{x}^{(i)}$, de dimensão (2, 1), que contém as coordenadas do ponto no plano, e a sua saída correspondente $y^{(i)}$ que classifica o ponto como pertencendo à classe azul ou vermelha, ou seja:

$$\begin{cases} y^{(i)} = 0 \Rightarrow \text{classe azul;} \\ y^{(i)} = 1 \Rightarrow \text{classe vermelha.} \end{cases}$$

- Para facilitar o treinamento da RNA, as entradas são normalizadas $\Rightarrow$ nesse caso dividimos as coordenadas dos pontos pelo módulo do seu valor máximo, ou seja:

$$\mathbf{x}^{(i)} = \frac{\mathbf{x}^{(i)}}{\max(\text{abs}(\mathbf{X}))} \tag{1}$$

onde $\mathbf{X}$ é uma matriz que contém as entradas de todos os exemplos de treinamento.

Conjunto de exemplos de treinamento

- O conjunto de exemplos de treinamento é composto por todos os dados de entrada e de saída de cada exemplo:
  - Número de exemplos de treinamento $\Rightarrow m$;
  - Cada exemplo $\Rightarrow (\mathbf{x}^{(i)}, y^{(i)})$, $\mathbf{x}^{(i)} \in \mathrm{R}^{nx}$ e $y^{(i)} \in \{0, 1\}$, ($n_x = 2$, $n_y = 1$);
  - Os m exemplos $\Rightarrow \{(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \ldots, (\mathbf{x}^{(m)}, y^{(m)})\}$.

Definição do problema

- Matematicamente podemos definir esse problema da seguinte forma:

  Dado o vetor $\mathbf{x}^{(i)} \Rightarrow$ calcular a probabilidade desse vetor pertencer à classe azul $(\hat{y}^{(i)} = 0)$, ou à classe vermelha $(\hat{y}^{(i)} = 1)$.

Configuração da RNA

- Para resolver esse problema usaremos uma RNA de duas camadas, ou seja, uma única camada escondida, conforme mostrada na Figura2.

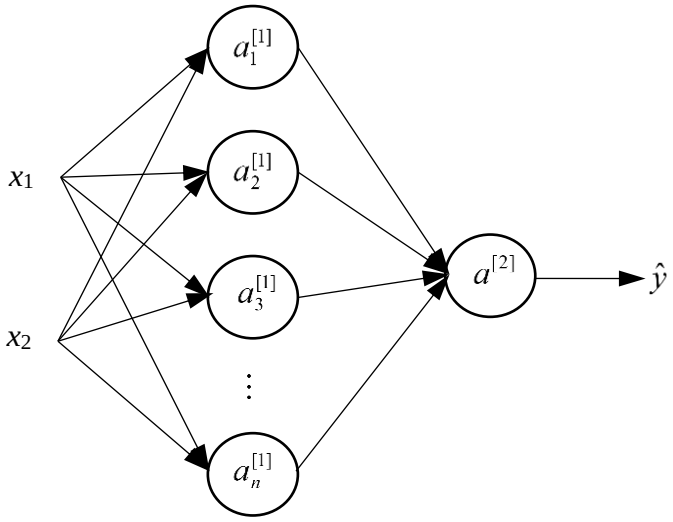

Figura 2. RNA rasa de uma saída para exemplo de classificação binária.

- A configuração da RNA é a seguinte:
  - Entrada da RNA $\Rightarrow$ vetor $\mathbf{x}^{(i)} \in \mathrm{R}^2$;
  - Saída da RNA $\Rightarrow$ $\hat{y}^{(i)} \in \{0, 1\}$;
  - Camada intermediária:
    - Números de neurônio: $n^{[1]} = n$;
    - Função de ativação da camada intermediária $\Rightarrow$ tangente hiperbólica;
  - Camada de saída:
    - Número de neurônios: $n^{[2]} = 1$;
    - Função de ativação da camada de saída $\Rightarrow$ sigmóide.

- **Como a saída da RNA deve ser uma probabilidade que varia entre 0 e 1 $\Rightarrow$ função de ativação da camada de saída deve ser a função sigmóide para garantir valores de saída entre 0 e 1.**

Função de custo

- Em problemas classificação binária utiliza-se a função de erro logística, ou seja:

$$L(\hat{y}^{(i)}, y^{(i)}) = -y^{(i)} \log \hat{y}^{(i)} - (1 - y^{(i)}) \log(1 - \hat{y}^{(i)}) \tag{2}$$

Assim, a função de custo fica sendo:

$$J(\mathbf{W},\mathbf{B}) = \frac{1}{m}\sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)}) = -\frac{1}{m}\sum_{i=1}^{m}\left[y^{(i)} \log \hat{y}^{(i)} + (1-y^{(i)})\log(1-\hat{y}^{(i)})\right] \tag{3}$$

onde **W** e **B** representam compactamente todos os pesos das ligações e vieses da RNA.

4. Propagação para frente

- **Camada intermediária** $\Rightarrow$ cálculos realizados para cada exemplo de treinamento:

$$\begin{cases} \mathbf{z}^{[1](i)} = \mathbf{W}^{[1]}\mathbf{x}^{(i)} + \mathbf{b}^{[1]} \\ \mathbf{a}^{[1](i)} = g^{[1]}(\mathbf{z}^{[1](i)}) \end{cases} \tag{4}$$

onde:

dimensão de $\mathbf{W}^{[1]} = (n, 2)$;

dimensão de $\mathbf{x}^{(i)} = 2$;

dimensão de $\mathbf{b}^{[1]}$ = dimensão de $\mathbf{a}^{[1]}$ = dimensão de $\mathbf{z}^{[1]} = (n, 1)$

- Função de ativação (tangente hiperbólica):

$$g^{[1]} = \tanh(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \tag{5}$$

- **Camada de saída** $\Rightarrow$ cálculos realizados para cada exemplo de treinamento:

$$\begin{cases} z^{[2](i)} = \mathbf{W}^{[2]}\mathbf{a}^{[1](i)} + b^{[2]} \\ \hat{y}^{(i)} = a^{[2](i)} = g^{[2]}(z^{[2](i)}) \end{cases} \tag{6}$$

onde dimensão de $\mathbf{W}^{[2]} = (1, n)$.

- Função de ativação (sigmóide):

$$g^{[2]} = \sigma(z) = \frac{1}{1+e^{-z}} \tag{7}$$

5. Retro-propagação

- **Derivadas parciais da função de custo em relação aos pesos $\mathbf{W}^{[2]}$**:

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_k^{[2]}} = \frac{1}{m}\sum_{i=1}^{m}\left\{\left[\frac{\partial L(a^{[2](i)}, y^{(i)})}{\partial a^{[2](i)}}\right]\left[\frac{dg^{[2]}(z^{[2](i)})}{dz^{[2](i)}}\right]a_k^{[1](i)}\right\}, \text{ para } k = 1,..., n \tag{8}$$

A derivada da função de erro logística é dada por:

$$\begin{aligned}\frac{dL(a^{[2](i)},y^{(i)})}{da^{[2](i)}} &= -\frac{d}{da^{[2](i)}}\left[y^{(i)}\log(a^{[2](i)})+(1-y^{(i)})\log(1-a^{[2](i)})\right]= \\ &= -\frac{y^{(i)}}{a^{[2](i)}}+\frac{1-y^{(i)}}{1-a^{[2](i)}}=\frac{a^{[2](i)}-y^{(i)}}{a^{[2](i)}(1-a^{[2](i)})}\end{aligned} \tag{9}$$

A derivada da função de ativação da camada 2 (sigmóide) é dada por:

$$\frac{dg^{[2]}(z^{[2](i)})}{dz^{[2](i)}}=\frac{d\sigma(z^{[2](i)})}{dz^{[2](i)}}=a^{[2](i)}(1-a^{[2](i)}) \tag{10}$$

Substituindo as equações (9) e (10) na equação (8), temos que:

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_k^{[2]}}=\frac{1}{m}\sum_{i=1}^{m}\left\{\left[\frac{a^{[2](i)}-y^{(i)}}{a^{[2](i)}(1-a^{[2](i)})}\right]\left[a^{[2](i)}(1-a^{[2](i)})\right]a_k^{[1](i)}\right\}, \text{ para } k = 1,..., n \tag{11}$$

Simplificando,

$$\boxed{\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_k^{[2]}}=\frac{1}{m}\sum_{i=1}^{m}\left[(a^{[2](i)}-y^{(i)})a_k^{[1](i)}\right], \text{ para } k = 1,..., n} \tag{12}$$

- **Derivada parcial da função de custo em relação aos viés $b^{[2]}$:**

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial b^{[2]}}=\frac{1}{m}\sum_{i=1}^{m}\left\{\left[\frac{\partial L(a^{[2](i)},y^{(i)})}{\partial a^{[2](i)}}\right]\left[\frac{dg^{[2]}(z^{[2](i)})}{dz^{[2](i)}}\right]\right\} \tag{13}$$

Usando os resultados das equações (9) e (10), temos:

$$\boxed{\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial b^{[2]}}=\frac{1}{m}\sum_{i=1}^{m}\left[(a^{[2](i)}-y^{(i)})\right]} \tag{14}$$

- **Derivadas parciais da função de custo em relação aos pesos $\mathbf{W}^{[1]}$:**

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_{k,j}^{[1]}}=\frac{1}{m}\sum_{i=1}^{m}\left\{\left[\frac{\partial L(a^{[2](i)},y^{(i)})}{\partial a^{[2](i)}}\right]\left[\frac{dg^{[2]}(z^{[2](i)})}{dz^{[2](i)}}\right]w_k^{[2]}\left[\frac{dg^{[1]}(z_k^{[1](i)})}{dz^{[1](i)}}\right]x_j\right\},$$

para k =1,..., n e j = 1 e 2. (15)

Usando os resultados das equações (9) e (10), temos que:

$$\left[\frac{\partial L(a^{[2](i)}, y^{(i)})}{\partial a^{[2](i)}}\right]\left[\frac{dg^{[2]}(z^{[2](i)})}{dz^{[2](i)}}\right] = a^{[2](i)} - y^{(i)} \tag{16}$$

A derivada da função de ativação da camada 1 (tangente hiperbólica) é dada por:

$$\frac{dg^{[1]}(z^{[1](i)})}{dz^{[1](i)}} = \frac{d\tanh(z^{[1](i)})}{dz^{[1](i)}} = 1 - \tanh^2(z^{[1](i)}) = 1 - \left(a^{[1](i)}\right)^2 \tag{17}$$

Note que o resultado da função de ativação da camada 1 é a própria ativação do neurônio da camada 1, ou seja, $a^{[1]}$.

Substituindo as equações (16) e (17) na equação (15), temos que:

$$\boxed{\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_{k,j}^{[1]}} = \frac{1}{m}\sum_{i=1}^{m}\left[(a^{[2](i)} - y^{(i)})w_k^{[2]}\left(1 - (a_k^{[1](i)})^2\right)x_j^{(i)}\right], \text{ para } k = 1,..., n \text{ e } j = 1 \text{ e } 2} \tag{18}$$

- **Derivadas parciais da função de custo em relação aos vieses $\mathbf{b}^{[1]}$:**

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial b_k^{[1]}} = \frac{1}{m}\sum_{i=1}^{m}\left\{\left[\frac{\partial L(a^{[2](i)}, y^{(i)})}{\partial a^{[2](i)}}\right]\left[\frac{dg^{[2]}(z^{[2](i)})}{dz^{[2](i)}}\right]w_k^{[2]}\left[\frac{dg^{[1]}(z_k^{[1](i)})}{dz^{[1](i)}}\right]\right\}, \text{ para } k = 1,..., n \tag{19}$$

Usando os resultados das equações (16) e (17), temos que:

$$\boxed{\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial b_k^{[1]}} = \frac{1}{m}\sum_{i=1}^{m}\left[(a^{[2](i)} - y^{(i)})w_k^{[2]}\left(1 - (a_k^{[1](i)})^2\right)\right], \text{ para } k = 1,..., n} \tag{20}$$

- **Atualização dos pesos da RNA em cada época:**

  - Para camada de saída:

$$\boxed{w_k^{[2]} = w_k^{[2]} - \alpha\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_k^{[2]}}, \text{ para } k = 1,..., n} \tag{21}$$

$$\boxed{b^{[2]} = b^{[2]} - \alpha\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial b^{[2]}}} \tag{22}$$

  - Para camada intermediária:

$$\boxed{w_{k,j}^{[1]} = w_{k,j}^{[1]} - \alpha\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_{k,j}^{[1]}}, \text{ para } k = 1,..., n \text{ e } j = 1 \text{ e } 2} \tag{23}$$

$$\boxed{b_k^{[1]} = b_k^{[1]} - \alpha\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial b_k^{[1]}}, \text{ para } k = 1,..., n} \tag{24}$$

- Como já mencionado, para o **completo** treinamento da RNA esse processo deve ser repetido inúmeras vezes até os parâmetros convergirem para um valor constante.

- No Quadro 1 é apresentado um resumo das equações da propagação para frente e da retro-propagação para uma RNA rasa aplicada ao problema de classificação binária.

Quadro 1. Resumo das equações para implementar a propagação para frente e a retro-propagação em uma RNA rasa para classificação binária.

Propagação para frente
$\mathbf{z}^{[1](i)} = \mathbf{W}^{[1]}\mathbf{x}^{(i)} + \mathbf{b}^{[1]}$ $\mathbf{a}^{[1](i)} = g^{[1]}(\mathbf{z}^{[1](i)})$ $z^{[2](i)} = \mathbf{W}^{[2]}\mathbf{a}^{[1](i)} + b^{[2]}$ $a^{[2](i)} = g^{[2]}(z^{[2](i)})$
Retro-propagação
$dz^{[2](i)} = a^{[2](i)} - y^{(i)}$ $d\mathbf{W}^{[2]} += \frac{1}{m} dz^{[2](i)}\mathbf{a}^{[1](i)T}$ $db^{[2]} += \frac{1}{m} dz^{[2](i)}$ $d\mathbf{z}^{[1](i)} = \mathbf{W}^{[2]T} dz^{[2](i)} * dg^{[1]}(\mathbf{z}^{[1](i)}) / dz$ $d\mathbf{W}^{[1]} += \frac{1}{m} d\mathbf{z}^{[1](i)}\mathbf{x}^{(i)T}$ $d\mathbf{b}^{[1]} += \frac{1}{m} d\mathbf{z}^{[1](i)}$

Nota: os termos $d(.)$ representam as derivadas parciais da função de custo.

6. Propagação para frente vetorizada nos exemplos

- A implementação da propagação para frente e da retro-propagação é feita de forma muito mais eficiente se for realizada vetorizadamente para todos os exemplos de treinamento sem nenhum comando de repetição.

Exemplos de treinamento para vetorização

- Para vetorização da RNA as m entradas são agrupadas em uma matriz de entradas:

$$\mathbf{X} = \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ \mathbf{x}^{(1)} & \mathbf{x}^{(2)} & \cdots & \mathbf{x}^{(m)} \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}_{(2,m)} \Rightarrow \mathbf{X} \in \mathrm{R}^{(2 \times m)}, \text{ dimensão } (2, m)$$

- As *m* saídas são agrupadas em um vetor de saídas:

$$\mathbf{y} = \begin{bmatrix} y^{(1)} & y^{(2)} & \cdots & y^{(m)} \end{bmatrix}_{(1,m)} \Rightarrow \mathbf{y} \in \mathrm{R}^{(1 x m)}, \text{ dimensão } (1, m)$$

Camada intermediária

- Os *m* vetores de estados da camada intermediária, cada um referente a um exemplo, são agrupados em uma matriz:

$$\mathbf{Z}^{[1]} = \begin{bmatrix} \mathbf{z}^{[1](1)} & \mathbf{z}^{[1](2)} & \cdots & \mathbf{z}^{[1](m)} \end{bmatrix}_{(n,m)} \Rightarrow \mathbf{Z}^{[1]} \in \mathrm{R}^{(n x m)}, \text{ dimensão } (n, m)$$

A matriz de estados $\mathbf{Z}^{[1]}$, que contém os vetores de estados para todos os exemplos, é calculada por:

$$\boxed{\mathbf{Z}^{[1]} = \mathbf{W}^{[1]}\mathbf{X} + \mathbf{b}^{[1]} = \begin{bmatrix} \leftarrow & \mathbf{w}_1^{[1]} & \rightarrow \\ \leftarrow & \mathbf{w}_2^{[1]} & \rightarrow \\ \leftarrow & \mathbf{w}_n^{[1]} & \rightarrow \end{bmatrix}_{(n,2)} \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ \mathbf{x}^{(1)} & \mathbf{x}^{(2)} & \cdots & \mathbf{x}^{(m)} \\ \downarrow & \downarrow & & \downarrow \end{bmatrix}_{(2,m)} + \begin{bmatrix} b_1^{[1]} \\ \vdots \\ b_n^{[1]} \end{bmatrix}} \tag{25}$$

onde $\mathbf{w}_k^{[1]}$ é a *k*-ésima linha da matriz de pesos $\mathbf{W}^{[1]}$.

- Os *m* vetores das ativações da camada intermediária, cada um referente a um exemplo, são agrupados em uma matriz:

$$\mathbf{A}^{[1]} = \begin{bmatrix} \mathbf{a}^{[1](1)} & \mathbf{a}^{[1](2)} & \cdots & \mathbf{a}^{[1](m)} \end{bmatrix}_{(n,m)} \Rightarrow \mathbf{A}^{[1]} \in \mathrm{R}^{(n x m)}, \text{ dimensão } (n, m)$$

A matriz de ativações $\mathbf{A}^{[1]}$, que contém os vetores de ativações para todos os exemplos, é calculada por:

$$\boxed{\mathbf{A}^{[1]} = \tanh(\mathbf{Z}^{[1]})} \tag{26}$$

Camada de saída

- Os *m* estados da camada de saída, cada um referente a um exemplo, são agrupados em um vetor:

$$\mathbf{z}^{[2]} = \begin{bmatrix} z^{[2](1)} & z^{[2](2)} & \cdots & z^{[2](m)} \end{bmatrix}_{(1,m)} \Rightarrow \mathbf{z}^{[2]} \in \mathrm{R}^{(1 x m)}, \text{ dimensão } (1, m)$$

O vetor de estados $\mathbf{z}^{[2]}$, que contém os estados para todos os exemplos, é calculado por:

$$\boxed{\mathbf{z}^{[2]} = \mathbf{W}^{[2]}\mathbf{A}^{[1]} + b^{[2]} = \mathbf{W}^{[2]} \begin{bmatrix} \uparrow & \uparrow & & \uparrow \\ \mathbf{a}^{(1)} & \mathbf{a}^{(2)} & \cdots & \mathbf{a}^{(m)} \\ \downarrow & \uparrow & & \uparrow \end{bmatrix}_{(n,m)} + b^{[2]}} \tag{27}$$

- Lembre que a dimensão do vetor de pesos da camada de saída é (1, n);
- Note que deve ser feito um acerto na dimensão de $b^{[2]}$ ao se somar um vetor linha com um escalar.

➢ As m ativações da camada de saída, cada uma referente a um exemplo, são agrupadas em um vetor:

$$\mathbf{a}^{[2]} = \left[a^{[2](1)} \quad a^{[2](2)} \quad \cdots \quad a^{[2](m)}\right]_{(1,m)} \Rightarrow \mathbf{a}^{[2]} \in \mathrm{R}^{(1x m)} \text{ (dimensão 1 x } m\text{)}$$

O vetor de ativações $\mathbf{a}^{[2]}$, que contém as ativações para todos os exemplos, é calculado por:

$$\boxed{\mathbf{a}^{[2]} = \sigma(\mathbf{z}^{[2]})} \tag{28}$$

➢ Lembre que as saídas previstas pela RNA, $\hat{\mathbf{y}}$, são as ativações da última camada, assim:

$$\boxed{\hat{\mathbf{y}} = \mathbf{a}^{[2]}} \tag{29}$$

7. Retro-propagação vetorizada nos exemplos

Camada de saída

➢ As derivadas parciais da função de custo em relação aos m estados da camada de saída, cada uma referente a um exemplo, são agrupados em um vetor denominado $d\mathbf{z}^{[2]}$, dado por:

$$d\mathbf{z}^{[2]} = \left[\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial \mathrm{z}^{[2](1)}} \quad \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial \mathrm{z}^{[2](2)}} \quad \cdots \quad \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial \mathrm{z}^{[2](m)}}\right]_{(1,m)} \Rightarrow \mathrm{d}\mathbf{z}^{[2]} \in \mathrm{R}^{(1x m)}, \text{ dimensão } (1, m)$$

Cada elemento do vetor $d\mathbf{z}^{[2]}$ é definido como sendo:

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial z^{[2](i)}} = \left[\frac{\partial L(a^{[2](i)}, y^{(i)})}{\partial a^{[2](i)}}\right]\left[\frac{dg^{[2]}(z^{[2](i)})}{dz^{[2](i)}}\right], \text{ para } i = 1,...,m \tag{30}$$

Usando os resultados das equações (9) e (10), temos que:

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial z^{[2](i)}} = a^{[2](i)} - y^{(i)}, \text{ , para } i = 1,...,m \tag{31}$$

Assim, o vetor $d\mathbf{z}^{[2]}$ pode ser escrito matricialmente por:

$$\boxed{d\mathbf{z}^{[2]} = \mathbf{a}^{[2]} - \mathbf{y}} \tag{32}$$

➢ As derivadas parciais da função de custo em relação aos pesos da camada de saída ($\mathbf{W}^{[2]}$) são agrupados em um vetor denominado $d\mathbf{W}^{[2]}$, dado por:

$$d\mathbf{W}^{[2]} = \left[\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_1^{[2]}} \quad \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_2^{[2]}} \quad \cdots \quad \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_n^{[2]}} \right]_{(1,n)} \Rightarrow \mathrm{d}\,\mathbf{W}^{[2]} \in \mathrm{R}^{(1x n)}, \text{ dimensão } (1, n)$$

Cada elemento do vetor $d\mathbf{W}^{[2]}$ é definido conforme a equação (11), repetida a seguir:

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_k^{[2]}} = \frac{1}{m}\sum_{i=1}^{m}\left[(a^{[2](i)} - y^{(i)})a_k^{[1](i)}\right], \text{ para } k = 1,...,n \tag{33}$$

Observe que todos os elementos do vetor $d\mathbf{W}^{[2]}$ podem ser calculados por uma única operação matricial, que consiste no produto entre o vetor $d\mathbf{z}^{[2]}$ e a matriz $\mathbf{A}^{[1]}$ transposta, da seguinte forma:

$$d\mathbf{W}^{[2]} = \frac{1}{m}\underbrace{\left[(a^{[2](1)} - y^{(1)}) \quad \cdots \quad (a^{[2](m)} - y^{(m)})\right]_{(1,m)}}_{d\mathbf{z}^{[2]}} \underbrace{\begin{bmatrix} \leftarrow & \mathbf{a}^{[1](i)T} & \rightarrow \\ \leftarrow & \mathbf{a}^{[1](2)T} & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{a}^{[1](m)T} & \rightarrow \end{bmatrix}_{(m,n)}}_{\mathbf{A}^{[1]T}} \tag{34}$$

ou simplesmente:

$$\boxed{d\mathbf{W}^{[2]} = \frac{1}{m} d\mathbf{z}^{[2]}\mathbf{A}^{[1]T}} \tag{35}$$

- A derivada parcial da função de custo em relação ao viés da camada de saída ($b^{[2]}$), dada pela equação (14), é simplesmente igual a soma de todos os elementos do vetor $d\mathbf{z}^{[2]}$ dividido pelo número de exemplos, ou seja:

$$\boxed{db^{[2]} = \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial b^{[2]}} = \frac{1}{m}\sum_{i=1}^{m}\left[(a^{[2](i)} - y^{(i)})\right] = \frac{1}{m}\sum_{i=1}^{m} dz^{[2](i)}} \tag{36}$$

onde $dz^{[2](i)}$ é o *i*-ésimo elemento do vetor $d\mathbf{z}^{[2]}$, referente ao *i*-ésimo exemplo.

Camada intermediária

- As derivadas parciais da função de custo em relação aos *n*x*m* estados da camada intermediária, cada um referente a um neurônio e a um exemplo, são agrupados em uma matriz denominada $d\mathbf{Z}^{[1]}$ dada por:

$$d\mathbf{Z}^{[1]} = \begin{bmatrix} \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial z_1^{[1](1)}} & \cdots & \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial z_1^{[1](m)}} \\ \vdots & \ddots & \vdots \\ \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial z_n^{[1](1)}} & \cdots & \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial z_n^{[1](m)}} \end{bmatrix}_{(n,m)} \Rightarrow \mathrm{d}\mathbf{Z}^{[1]} \in \mathrm{R}^{(n x m)}, \text{ dimensão } (n, m)$$

Cada elemento da matriz $d\mathbf{Z}^{[1]}$ é definido como sendo:

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial z_k^{[1](i)}} = \left[\frac{\partial L(a^{[2](i)}, y^{(i)})}{\partial a^{[2](i)}}\right]\left[\frac{dg^{[2]}(z^{[2](i)})}{dz^{[2](i)}}\right] w_k^{[2]} \left[\frac{dg^{[1]}(z_k^{[1](i)})}{dz^{[1](i)}}\right], \text{ para } k = 1,...,n \tag{37}$$

Usando os resultados das equações (16) e (17), temos que:

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial z_k^{[1](i)}} = (a^{[2](i)} - y^{(i)}) w_k^{[2]} \left(1 - (a_k^{[1](i)})^2\right), \text{ para } k = 1,...,n \tag{38}$$

Analisando a equação (38), conclui-se que a matriz $d\mathbf{Z}^{[1]}$ pode ser escrita por uma operação entre as matrizes $\mathbf{W}^{[2]}$, $d\mathbf{Z}^{[2]}$ e $\mathbf{A}^{[1]}$, da seguinte forma:

$$d\mathbf{Z}^{[1]} = \underbrace{\begin{bmatrix} w_1^{[2]} \\ \vdots \\ w_n^{[2]} \end{bmatrix}_{(n,1)}}_{\mathbf{W}^{[2]T}} \underbrace{\left[(a^{[2](1)} - y^{(1)}) \quad \cdots \quad (a^{[2](m)} - y^{(m)})\right]_{(1,m)}}_{d\mathbf{Z}^{[2]}} * \underbrace{\begin{bmatrix} 1 - a_1^{[1](1)} & \cdots & 1 - a_1^{[1](m)} \\ \vdots & \vdots & \vdots \\ 1 - a_n^{[1](1)} & \cdots & 1 - a_n^{[1](m)} \end{bmatrix}_{(n,m)}}_{(\mathbf{1} - \mathbf{A}^{[1]})} \tag{39}$$

onde $\mathbf{1}$ é uma matriz de uns com mesma dimensão da matriz $\mathbf{A}^{[1]}$ e o símbolo * representa multiplicação elemento por elemento.

A equação (39) pode ser escrita simplesmente como sendo:

$$\boxed{d\mathbf{Z}^{[1]} = \left[\mathbf{W}^{[2]T} d\mathbf{z}^{[2]}\right] * \left[\mathbf{1} - \mathbf{A}^{[1]}\right]} \tag{40}$$

- As derivadas parciais da função de custo em relação aos pesos da camada intermediária ($\mathbf{W}^{[1]}$) são agrupados em uma matriz denominada $d\mathbf{W}^{[1]}$ dada por:

$$d\mathbf{W}^{[1]} = \begin{bmatrix} \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_{1,1}^{[1]}} & \cdots & \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_{1,n}^{[1]}} \\ \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_{2,1}^{[1]}} & \cdots & \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_{2,n}^{[1]}} \end{bmatrix}_{(2,n)} \Rightarrow \mathrm{d}\mathbf{W}^{[1]} \in \mathrm{R}^{(2xn)}, \text{ dimensão } (2, n)$$

Cada elemento da matriz $d\mathbf{W}^{[1]}$ é definido conforme a equação (18), repetida a seguir:

$$\frac{\partial J(\mathbf{W},\mathbf{B})}{\partial w_{k,j}^{[1]}} = \frac{1}{m}\sum_{i=1}^{m}\left[(a^{[2](i)} - y^{(i)}) w_k^{[2]} \left(1 - (a_k^{[1](i)})^2\right) x_j^{(i)}\right], \text{ para } k = 1,..., n \text{ e } j = 1 \text{ e } 2 \tag{41}$$

Note que cada elemento da $\mathrm{d}\mathbf{Z}^{[1]}$ é igual ao termo entre colchetes da equação (41), ou seja;

$$dz_k^{[1](i)} = (a^{[2](i)} - y^{(i)}) w_k^{[2]} \left(1 - (a_k^{[1](i)})^2\right), \text{ para } k = 1,..., n \text{ e } i = 1,...,m \tag{42}$$

Assim, todos os elementos da matriz $d\mathbf{W}^{[1]}$ podem ser calculados por uma única operação, que consiste no produto entre a matriz $d\mathbf{Z}^{[1]}$ e a matriz de entradas $\mathbf{X}$ transposta, da seguinte forma:

$$d\mathbf{W}^{[1]} = \frac{1}{m}\begin{bmatrix} dz_1^{[1](i)} & \cdots & dz_1^{[1](m)} \\ \vdots & \ddots & \vdots \\ dz_n^{[1](i)} & \cdots & dz_n^{[1](m)} \end{bmatrix}_{(n,m)} \begin{bmatrix} \leftarrow & \mathbf{x}^{(1)} & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}^{(m)} & \rightarrow \end{bmatrix}_{(m,2)} \tag{43}$$

ou simplesmente:

$$\boxed{d\mathbf{W}^{[1]} = \frac{1}{m} d\mathbf{Z}^{[1]}\mathbf{X}^T} \tag{44}$$

- As derivadas parciais da função de custo em relação aos vieses da camada intermediária ($\boldsymbol{b}^{[1]}$), dada pela equação (20), são simplesmente a soma dos elementos de cada linha da matriz $d\mathbf{Z}^{[1]}$ dividido pelo número de exemplos, ou seja:

$$\boxed{d\mathbf{b}^{[1]} = \begin{bmatrix} \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial b_1^{[1]}} \\ \vdots \\ \frac{\partial J(\mathbf{W},\mathbf{B})}{\partial b_n^{[1]}} \end{bmatrix} = \frac{1}{m}\begin{bmatrix} \sum_{i=1}^{m}\left[(a^{[2](i)} - y^{(i)})w_k^{[2]}\left(1-(a_k^{[1](i)})^2\right)\right] \\ \vdots \\ \sum_{i=1}^{m}\left[(a^{[2](i)} - y^{(i)})w_k^{[2]}\left(1-(a_k^{[1](i)})^2\right)\right] \end{bmatrix} = \frac{1}{m}\begin{bmatrix} \sum_{i=1}^{m} dz_1^{[1](i)} \\ \vdots \\ \sum_{i=1}^{m} dz_n^{[1](i)} \end{bmatrix}} \tag{45}$$

Atualização dos pesos da RNA em cada época

- A atualização dos pesos da RNA também pode ser implementada vetorizadamente para todos os exemplos por uma única operação matricial em cada época do treinamento, ou seja:

$$\boxed{\mathbf{W}^{[2]} = \mathbf{W}^{[2]} - \alpha\, d\mathbf{W}^{[2]}} \tag{46}$$

$$\boxed{b^{[2]} = b^{[2]} - \alpha\, db^{[2]}} \tag{47}$$

$$\boxed{\mathbf{W}^{[1]} = \mathbf{W}^{[1]} - \alpha\, d\mathbf{W}^{[1]}} \tag{48}$$

$$\boxed{\mathbf{b}^{[1]} = \mathbf{b}^{[1]} - \alpha\, d\mathbf{b}^{[1]}} \tag{49}$$

onde α é a taxa de aprendizagem, como já visto.

- No Quadro 2é apresentado um resumo das equações da propagação para frente e da retro-propagação com vetorização nos exemplos para uma RNA rasa aplicada ao problema de classificação binária.

Quadro 2. Resumo das equações para implementar a propagação para frente e a retro propagação em uma RNA rasa para classificação binária com vetorização de cálculo nos exemplos.

Propagação para frente
$\mathbf{Z}^{[1]} = \mathbf{W}^{[1]}\mathbf{X} + \mathbf{b}^{[1]}$ $\mathbf{A}^{[1]} = g^{[1]}(\mathbf{Z}^{[1]})$ $\mathbf{z}^{[2]} = \mathbf{W}^{[2]}\mathbf{A}^{[1]} + b^{[2]}$ $\mathbf{a}^{[2]} = g^{[2]}(\mathbf{z}^{[2]})$
Retro propagação
$d\mathbf{z}^{[2]} = \mathbf{a}^{[2]} - \mathbf{y}$ $d\mathbf{W}^{[2]} = \frac{1}{m} d\mathbf{z}^{[2]} \mathbf{A}^{[1]T}$ $db^{[2]} = \frac{1}{m} \texttt{np.sum}(d\mathbf{z}^{[2]}, \texttt{axis=1, keepdims=True})$ $d\mathbf{Z}^{[1]} = \mathbf{W}^{[2]T} d\mathbf{z}^{[2]} * dg^{[1]}(\mathbf{Z}^{[1]}) / dz$ $d\mathbf{W}^{[1]} = \frac{1}{m} d\mathbf{Z}^{[1]} \mathbf{X}^{T}$ $d\mathbf{b}^{[1]} = \frac{1}{m} \texttt{np.sum}(d\mathbf{Z}^{[1]}, \texttt{axis=1, keepdims=True})$

- Observe que no cálculo das derivadas parciais da função de custo em relação aos vieses $\mathbf{b}^{[1]}$ e $b^{[2]}$ vetorizado nos exemplos aparece a função `sum()` da biblioteca numpy em razão dessas equações implementarem uma somatória.

8. Processo de treinamento de uma RNA

- Como já visto, o treinamento de um RNA é um processo iterativo no qual se deseja calcular os parâmetros da RNA para que ela aprenda os padrões contidos nos exemplos fornecidos.

- Dado um conjunto de exemplos de treinamento e uma estrutura de RNA, o processo de treinamento consiste nas seguintes etapas:

  1. Inicializar os parâmetros da RNA;

  2. Executar a RNA com todos os exemplos de treinamento e obter todas as saídas previstas;

  3. Calcular os erros entre as saídas desejadas e as previstas pela RNA computando a função de custo;

  4. Calcular o gradiente da função de custo em relação a todos os parâmetros da RNA;

  5. Atualizar os parâmetros da RNA;

6. Repetir os passos 2 a 5 quantas vezes for necessário para a função de custo atingir um valor mínimo desejado, ou até o número máximo de repetições for alcançado, ou até os parâmetros convergirem para valores constantes.

- No Quadro 3 é apresentado o algoritmo para implementar o processo de treinamento de um RNA sem vetorização nos exemplos para uma RNA rasa aplicada ao problema de classificação binária.

Quadro 3. Algoritmo de treinamento sem vetorização nos exemplos para um problema de classificação binária.

Loop nas épocas
for *e* =1 to *n_epocas*

Loop nos exemplos
for *i*=1 to *m*

Propagação para frente

$$\mathbf{z}^{[1](i)} = \mathbf{W}^{[1]}\mathbf{x}^{(i)} + \mathbf{b}^{[1]}$$

$$\mathbf{a}^{[1](i)} = g^{[1]}(\mathbf{z}^{[1](i)})$$

$$z^{[2](i)} = \mathbf{W}^{[2]}\mathbf{a}^{[1](i)} + b^{[2]}$$

$$a^{[2](i)} = g^{[2]}(z^{[2](i)})$$

Retro-propagação

$$dz^{[2](i)} = a^{[2](i)} - y^{(i)}$$

$$d\mathbf{W}^{[2]} += \frac{1}{m} dz^{[2](i)}\mathbf{a}^{[1](i)T}$$

$$db^{[2]} += \frac{1}{m} dz^{[2](i)}$$

$$d\mathbf{z}^{[1](i)} = \mathbf{W}^{[2]T} dz^{[2](i)} * dg^{[1]}(\mathbf{z}^{[1](i)}) / dz$$

$$d\mathbf{W}^{[1]} += \frac{1}{m} d\mathbf{z}^{[1](i)}\mathbf{x}^{(i)T}$$

$$d\mathbf{b}^{[1]} += \frac{1}{m} d\mathbf{z}^{[1](i)}$$

Atualização dos parâmetros por época

$$\mathbf{W}^{[2]} = \mathbf{W}^{[2]} - \alpha d\mathbf{W}^{[2]}$$

$$\mathbf{W}^{[1]} = \mathbf{W}^{[1]} - \alpha d\mathbf{W}^{[1]}$$

$$b^{[2]} = b^{[2]} - \alpha db^{[2]}$$

$$\mathbf{b}^{[1]} = \mathbf{b}^{[1]} - \alpha d\mathbf{b}^{[1]}$$

- No Quadro 4 é apresentado o algoritmo para implementar o processo de treinamento de um RNA com vetorização nos exemplos para uma RNA rasa aplicada ao problema de classificação binária.

Quadro 4. Algoritmo de treinamento com vetorização nos exemplos para um problema de classificação binária.

Loop nas épocas

for *e* =1 to *n_epocas*

Propagação para frente

$$\mathbf{Z}^{[1]} = \mathbf{W}^{[1]}\mathbf{X} + \mathbf{b}^{[1]}$$

$$\mathbf{A}^{[1]} = g^{[1]}(\mathbf{Z}^{[1]})$$

$$\mathbf{z}^{[2]} = \mathbf{W}^{[2]}\mathbf{A}^{[1]} + b^{[2]}$$

$$\mathbf{a}^{[2]} = g^{[2]}(\mathbf{z}^{[2]})$$

Retro-propagação

$$d\mathbf{z}^{[2]} = \mathbf{a}^{[2]} - \mathbf{y}$$

$$d\mathbf{W}^{[2]} = \frac{1}{m} d\mathbf{z}^{[2]} \mathbf{A}^{[1]T}$$

$$db^{[2]} = \frac{1}{m} \texttt{np.sum}(d\mathbf{z}^{[2]}, \texttt{axis} = 1, \texttt{keepdims} = \texttt{True})$$

$$d\mathbf{Z}^{[1]} = \mathbf{W}^{[2]T} d\mathbf{z}^{[2]} * dg^{[1]}(\mathbf{Z}^{[1]}) / dz$$

$$d\mathbf{W}^{[1]} = \frac{1}{m} d\mathbf{Z}^{[1]} \mathbf{X}^{T}$$

$$d\mathbf{b}^{[1]} = \frac{1}{m} \texttt{np.sum}(d\mathbf{Z}^{[1]}, \texttt{axis} = 1, \texttt{keepdims} = \texttt{True})$$

Atualização dos parâmetros por época

$$\mathbf{W}^{[2]} = \mathbf{W}^{[2]} - \alpha d\mathbf{W}^{[2]}$$

$$\mathbf{W}^{[1]} = \mathbf{W}^{[1]} - \alpha d\mathbf{W}^{[1]}$$

$$b^{[2]} = b^{[2]} - \alpha db^{[2]}$$

$$\mathbf{b}^{[1]} = \mathbf{b}^{[1]} - \alpha d\mathbf{b}^{[1]}$$