Trabalho 3: Redes Neurais Artificiais

Bruce Wayne
Universidade Federal do Pampa

17 de setembro de 2015

1 Definições de variáveis com vetorização

$W_{j,i}^{(\ell)}$ é a matriz de pesos da camada ℓ. Onde conecta-se os i-ésimos neurônios da camada $(\ell - 1)$ com os j-ésimos neurônios da camada ℓ.

Seja S_l o número de neurônios na camada ℓ e $S_{\ell+1}$ o número de neurônios na camada $(\ell + 1)$, então $W^{(\ell)}$ tem dimensão: $S_{\ell+1} \times (S_\ell + 1)$.

$z_i^{(\ell)}$ é a entrada ponderada do i-ésimo neurônio na camada ℓ. Basicamente, para a primeira camada temos:
$z_i^{(\ell)} = W_{i,0}^{(\ell)} \cdot X_0 + W_{i,1}^{(\ell)} \cdot X_1 + \cdots + W_{i,n}^{(\ell)} \cdot X_n$
$z^{(\ell)} = W^{(\ell)} \cdot X$

Aplicando a função de ativação g para cada neurônio na camada ℓ:
$a^{(\ell)} = g(z^{(\ell)})$

E portanto:
$z^{(\ell+1)} = W^{(\ell)} \cdot a^{(\ell)}$
$a^{(\ell+1)} = g(z^{(\ell+1)})$

Com isso, podemos atribuir $a^{(0)} = X$ e apenas usar as variáveis W, z, a até a última camada (*forward propagation*).

2 Arquitetura da rede neural

A arquitetura da nossa rede é praticamente uma *feedforward*, com a exceção que há conexões com a primeira e a última camada. Aqui vou considerar que a primeira camada (camada de entrada) tem N_1 neurônios, a segunda camada (camada oculta) tem N_2 neurônios e a última camada (camada de saída) tem 1 neurônio.

Desse modo, há $N_1 * N_2$ conexões entre a primeira e a segunda camada. Tem $N_2 * 1$ conexões entre a segunda e a última camada. E por fim, tem $N_1 * 1$ entre a primeira e a última camada.

3 Forward propagation

Para fazer o *forward propagation*, tudo o que precisamos fazer é multiplicar os vetores/matrizes até a última camada. Mas primeiro vamos definir as dimensões de cada matriz de pesos para facilitar o entendimento:

$$
\begin{aligned}
a^{(0)} &= X(N_1 \times 1) \\
&W^{(1)}(N_2 \times N_1) \\
&W^{(2)}(1 \times N_2) \\
&W^{(3)}(1 \times N_1)
\end{aligned}
$$

Agora sim, os passos até a hipótese resultante:

$$
\begin{aligned}
z^{(1)} &= W^{(1)} \cdot a^{(0)} & (N_2 \times N_1) \times (N_1 \times 1) \rightarrow (N_2 \times 1) \\
a^{(1)} &= g(z^{(1)}) & (N_2 \times 1) \\
\\
z^{(2)} &= W^{(2)} \cdot a^{(1)} & (1 \times N_2) \times (N_2 \times 1) \rightarrow (1 \times 1) \\
a^{(2)} &= g(z^{(2)}) & (1 \times 1) \\
\\
z^{(3)} &= W^{(3)} \cdot a^{(0)} & (1 \times N_1) \times (N_1 \times 1) \rightarrow (1 \times 1) \\
a^{(3)} &= g(z^{(3)}) & (1 \times 1) \\
\\
z^{(4)} &= z^{(2)} + z^{(3)} & (1 \times 1) \times (1 \times 1) \rightarrow (1 \times 1) \\
a^{(4)} &= g(z^{(4)}) & (1 \times 1)
\end{aligned}
$$

4 Backpropagation

Temos três matrizes de pesos para ajustar, e queremos minimizar a função de custo $J(W)$, definida como:

$$J(W) = \frac{1}{2}(y - a^{(4)})^2$$

Onde y é o resultado esperado e $a^{(4)}$ é o resultado previsto pela rede neural.

Então agora podemos começar a derivar:

$$\frac{\partial J(W)}{\partial W^{(3)}} = \frac{\partial J(W)}{\partial a^{(4)}} \cdot \frac{\partial a^{(4)}}{\partial z^{(4)}} \cdot \frac{\partial z^{(4)}}{\partial W^{(3)}}$$

$$\frac{\partial J(W)}{\partial W^{(2)}} = \frac{\partial J(W)}{\partial a^{(4)}} \cdot \frac{\partial a^{(4)}}{\partial z^{(4)}} \cdot \frac{\partial z^{(4)}}{\partial W^{(2)}}$$

$$\frac{\partial J(W)}{\partial W^{(1)}} = \frac{\partial J(W)}{\partial a^{(4)}} \cdot \frac{\partial a^{(4)}}{\partial z^{(4)}} \cdot \frac{\partial z^{(4)}}{\partial z^{(2)}} \cdot \frac{\partial z^{(2)}}{\partial a^{(1)}} \cdot \frac{\partial a^{(1)}}{\partial z^{(1)}} \cdot \frac{\partial z^{(1)}}{\partial W^{(1)}}$$

Agora vem a parte entediante... Já dá para ver que as três derivadas compartilham um pedaço em comum. Isso nos dá uma pista de que talvez possamos reusar algumas variáveis. Vamos primeira derivar a primeira equação:

$$\frac{\partial J(W)}{\partial W^{(3)}} = \frac{\partial J(W)}{\partial a^{(4)}} \cdot \frac{\partial a^{(4)}}{\partial z^{(4)}} \cdot \frac{\partial z^{(4)}}{\partial W^{(3)}}$$

$$= -(y - a^{(4)}) \cdot g'(z^{(4)}) \cdot \frac{\partial z^{(4)}}{\partial W^{(3)}}$$

$$= (a^{(4)} - y) \cdot g'(z^{(4)}) \cdot \frac{\partial z^{(4)}}{\partial W^{(3)}}$$

Definimos então uma nova variável $\delta^{(\ell)}$, que é um vetor de erros para cada neurônio na camada ℓ. Portanto, como estamos avaliando os neurônios da camada 4 (perceba que essa camada não existe na arquitetura original, foi apenas um modo que encontrei para realizar a soma dos pesos da primeira e da segunda camada em relação a última).

$$\delta^{(4)} = \frac{\partial J(W)}{\partial a^{(4)}} \cdot \frac{\partial a^{(4)}}{\partial z^{(4)}}$$

$$= (a^{(4)} - y) \cdot g'(z^{(4)})$$

Desse modo, temos:

$$\frac{\partial J(W)}{\partial W^{(3)}} = \delta^{(4)} \cdot \frac{\partial z^{(4)}}{\partial W^{(3)}}$$

$$= (a^{(4)} - y) \cdot g'(z^{(4)}) \cdot \frac{\partial z^{(4)}}{\partial W^{(3)}}$$

$$= (a^{(4)} - y) \cdot g'(z^{(4)}) \cdot a^{(0)}$$

Definimos essa derivada como o gradiente de $W^{(3)}$:

$$\nabla W^{(3)} = (a^{(4)} - y) \cdot g'(z^{(4)}) \cdot a^{(0)}$$

E agora fizemos o mesmo com as outras matrizes:

$$\frac{\partial J(W)}{\partial W^{(2)}} = \frac{\partial J(W)}{\partial a^{(4)}} \cdot \frac{\partial a^{(4)}}{\partial z^{(4)}} \cdot \frac{\partial z^{(4)}}{\partial W^{(2)}}$$

$$= \delta^{(4)} \cdot \frac{\partial z^{(4)}}{\partial W^{(2)}}$$

$$\nabla W^{(2)} = (a^{(4)} - y) \cdot g'(z^{(4)}) \cdot a^{(1)}$$

E para a $W^{(1)}$ definimos mais um vetor $\delta^{(2)}$:

$$\delta^{(2)} = \delta^{(4)} \cdot \frac{\partial z^{(4)}}{\partial z^{(2)}} \cdot \frac{\partial z^{(2)}}{\partial a^{(1)}} \cdot \frac{\partial a^{(1)}}{\partial z^{(1)}}$$

$$= \delta^{(4)} \cdot 1 \cdot W^{(2)} \cdot g'(z^{(1)})$$

Com isso, podemos calcular facilmente o gradiente da matriz de pesos $W^{(1)}$:

$$\frac{\partial J(W)}{\partial W^{(1)}} = \frac{\partial J(W)}{\partial a^{(4)}} \cdot \frac{\partial a^{(4)}}{\partial z^{(4)}} \cdot \frac{\partial z^{(4)}}{\partial z^{(2)}} \cdot \frac{\partial z^{(2)}}{\partial a^{(1)}} \cdot \frac{\partial a^{(1)}}{\partial z^{(1)}} \cdot \frac{\partial z^{(1)}}{\partial W^{(1)}}$$

$$= \delta^{(2)} \cdot \frac{\partial z^{(1)}}{\partial W^{(1)}}$$

$$= \delta^{(2)} \cdot a^{(0)}$$

$$\nabla W^{(1)} = (a^{(4)} - y) \cdot g'(z^{(4)}) \cdot W^{(2)} \cdot g'(z^{(1)}) \cdot a^{(0)}$$

5 Atualização dos pesos

Para atualizar os pesos, precisamos seguir a direção oposto do vetor gradiente, então subtraímos o gradiente de cada matriz do que temos temos atualmente nela, lembrando de multiplicar pela taxa de aprendizagem η para o gradiente descendente não dar passos largos e eventualmente divergir:

$$W^{(3)} = W^{(3)} - \eta \cdot \nabla W^{(3)}$$
$$W^{(2)} = W^{(2)} - \eta \cdot \nabla W^{(2)}$$
$$W^{(1)} = W^{(1)} - \eta \cdot \nabla W^{(1)}$$

E et voilà!