

Questionário - Deep Learning

1. [3pt] Considere uma rede neural de 1 camada oculta com 2 neurônios e 1 camada de saída com 2 neurônios, e função de ativação $f(\cdot)$ do tipo sigmóide para todo neurônio da rede. A rede recebe como entrada dois valores numéricos (Figura 1). A função de custo E é o erro quadrático para um único exemplo de treinamento. Use obrigatoriamente a notação y_i para a ativação de um neurônio i , e z_i para a soma ponderada correspondente. Responda:

- (a) Usando a regra da cadeia, calcule

$$\frac{\partial E}{\partial w_{kl}}$$

(referente à última camada de pesos) para um exemplo de treinamento $\langle x, t \rangle$, em função das ativações neuronais e saída desejada t .

- (b) Dado o exemplo $\langle x = (0, 1), t = (0.8, 1) \rangle$:

- i. calcule a propagação (*forward*) de sinais na rede neural.
- ii. Para esse mesmo exemplo, calcule o valor numérico da derivada parcial $\frac{\partial E}{\partial w_{kl}}$, onde $k = l = 2$ (calcule o valor final com 4 casas decimais após a vírgula).
- iii. Use o método do gradiente para ajustar o peso w_{kl} , $k = l = 2$, com taxa $\alpha = 1$.
- iv. Com o novo peso w_{kl} , calcule mais uma vez a ativação do neurônio $l = 2$ da camada de saída. Explique o novo valor com relação ao valor passado.
- v. Adicione regularização $\lambda = 0.1$ no ajuste de w_{kl} . Explique o valor resultante de w_{kl} .

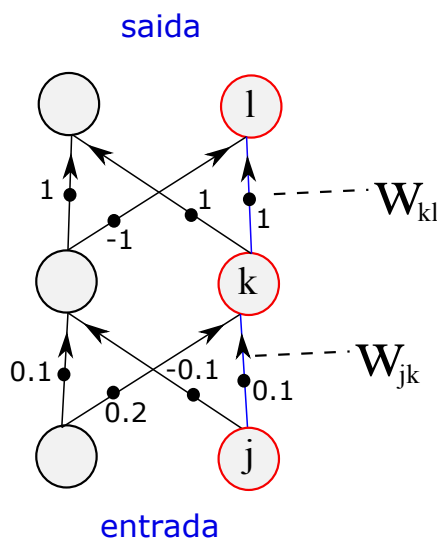


Figure 1: Rede neural: Neurônios à direita em vermelho correspondem aos índices $j=2$, $k=2$, e $l=2$ em suas respectivas camadas. Os pesos w_{kl} e w_{jk} conectam neurônios entre camadas adjacentes e tem valores inicializados conforme mostra a figura (ex.: $w_{jk} = 0.1$ para $j = 2$ e $k = 2$, aresta em azul).

2. [2pt] Como as redes neurais convolucionais conseguem ser mais eficientes do que redes multicamadas totalmente conectadas (*fully connected multilayer networks*) para vários problemas, em especial processamento de imagens e visão computacional, mas também outros problemas envolvendo dados em grade (*grid*), como séries temporais?
3. [2pt] Um programador obteve de um amigo o código do algoritmo de retropropagação de erros para redes neurais de múltiplas camadas. Ao realizar o treinamento de uma rede neural, observou que os pesos da camada $n - 1$ são atualizados conjuntamente com os pesos da camada n . É possível que o algoritmo esteja errado, devido a essa observação? Justifique.
4. [1.5pt] No treinamento de uma hipótese, a saída desejada é 0.7 e a saída da hipótese é 0.6 para um certo exemplo de treinamento. Após executar o código que ajusta os parâmetros da hipótese aplicando o descenso do gradiente para esse exemplo, a saída da hipótese muda para 0.6001 ao realizar uma nova propagação de sinais para o mesmo exemplo. Isso é esperado ou ocorreu um erro de implementação? Explique.
5. [1.5pt] Um estudante observou que ao retirar um peso de um neurônio, o desempenho da rede neural melhora para novos sinais de dados. Qual é a provável explicação disso? Discorra.
6. [opcional - 3pt] Para a mesma rede neural acima, desenvolva $\frac{\partial E}{\partial w_{jk}}$ pela regra da cadeia, expressando-a em função de ativações neuronais, pesos sinápticos, entradas da rede x_j , e possivelmente o erro delta δ_l ou δ_k . Essa derivada parcial é feita com relação à primeira camada de pesos sinápticos. Após isso, faça os mesmos passos do exercício anterior: calcule numericamente $\frac{\partial E}{\partial w_{jk}}$ para $k = 2$, $j = 2$, para o mesmo exemplo $\langle x = (0, 1), t = (0.8, 1) \rangle$; e ajuste w_{jk} pelo método do gradiente com $\alpha = 1$, $j = 2$, $k = 2$.