

MBA em Ciência de Dados

Redes Neurais e Arquiteturas Profundas

Módulo I - Deep Learning e redes do tipo Perceptron

Avaliação

Moacir Antonelli Ponti

CeMEAI - ICMC/USP São Carlos

As respostas devem ser dadas no Moodle. O notebook deve ser usado apenas para codificar o necessário para responder as questões

Questão 1)

O que diferencia métodos de aprendizado profundo (*deep learning*) de métodos de aprendizado de máquina considerados rasos (*shallow*)?

- (a) Os métodos rasos comumente aprendem um mapeamento direto entre dados de entrada (atributos) e saída (alvo), enquanto os profundos aprendem uma sequência de mapeamentos (ou funções) para múltiplos espaços antes de mapear para o espaço de saída alvo
- (b) Os métodos rasos podem ser considerados aprendizado de máquina e permitem tarefas distintas como classificação, regressão, agrupamento, entre outros, enquanto os chamados profundos permitem modelar tarefas de classificação
- (c) Os métodos rasos são baseados em métodos estatísticos e árvores de decisão, enquanto os de aprendizado profundo são unicamente baseados em redes neurais
- (d) Os métodos rasos trabalham apenas com dados estruturados, enquanto que os profundos funcionam com dados estruturados e não estruturados.

Questão 2)

Seja \mathbf{z} um vetor de entrada e \mathbf{s} um vetor de saída de uma camada de rede neural baseada em Perceptron. Essa camada pode ser formulada como:

$$f(\mathbf{z}) = a(W\mathbf{z} + \mathbf{b}) = \mathbf{s}, \text{ sendo que } a() \text{ é a função de ativação.}$$

Sabendo que a entrada tem 40 dimensões e a saída tem k dimensões, Qual o

tamanho da matriz W e do vetor b e quantos parâmetros essa camada possui para serem aprendidos durante o treinamento?

- (a) W possui $40 \times k$, e b possui 40 dimensões, totalizando $40k + 40$ parâmetros
 - (b) W possui $k \times 40$, e b possui k dimensões, totalizando $41k$ parâmetros
 - (c) W possui $k \times 40$, e b possui 1 dimensão (escalar), totalizando $40k + 1$ parâmetros
 - (d) W possui $k \times k$, e b possui 40 dimensões, totalizando $2k + 40$ parâmetros
-

Questão 3)

Qual o impacto do tamanho do batch (lote) no treinamento por meio do Stochastic Gradient Descent (SGD)?

- (a) O tamanho do batch impacta na quantidade de épocas necessárias para completar o treinamento, se o tamanho do batch for grande, apenas uma época é necessária
 - (b) Quanto menor o tamanho do batch, mais rápido o treinamento, pois assim o SGD se aproxima do Gradient Descent convencional já que utiliza cada exemplo individualmente para adaptar os pesos
 - (c) Quanto menor o tamanho do batch melhor será a acurácia do modelo pois as estimativas do gradiente serão mais precisas considerando cada iteração
 - (d) Quanto menor o tamanho do batch, mais rápida cada iteração do treinamento, porém mais grosseira é a estimativa do gradiente por iteração
-

Questão 4)

Defina as sementes aleatórias do numpy para 1 e do tensorflow para 2, depois carregue a base de dados boston housing da biblioteca Keras, conforme código abaixo.

O objetivo dessa base de dados é obter a regressão do preço das casas com base em 13 características de entrada. Assim, os valores alvo (target) são escalares, tipicamente entre 10 e 50 (representando os preços em milhares de dólares).

Utilizando a biblioteca Keras, formule um modelo de rede neural sequencial, do tipo MLP, com 3 camadas ocultas contendo, respectivamente, e, na ordem, 32, 16 e 8 neurônios, todas com função de ativação do tipo `relu`.

Quantos parâmetros, no total, essa rede possui?

- (a) 4096
- (b) 1121
- (c) 53248

```

In [1]: from tensorflow import keras

        from numpy.random import seed
        seed(1)
        from tensorflow.random import set_seed
        set_seed(2)

        from tensorflow.keras.datasets import boston_housing
        (x_train, y_train), (x_target, y_target) = boston_housing.load_data()

```

In []:

Questão 5)

Utilizando a base de dados e o modelo de rede neural criado na questão anterior, compile o modelo utilizando uma função de custo `mae` (mean absolute error), o otimizador SGD e a taxa de aprendizado 0.01.

Adicione também a métrica `mse` (mean squared error) para permitir avaliá-la adicionalmente.

Normalize os dados (x) por meio da normalização z-score (calcule média e desvio no conjunto de treinamento apenas).

Utilize os dados normalizados para treinar a rede neural por 15 épocas, com batch-size 4.

Avalie o modelo treinado nos dados de teste, e reporte as posições 0 e 1 do score resultante, respectivamente relativas ao MAE e MSE calculados. Escolha a opção para a qual o intervalo se enquadre nos valores computados.

(a) MAE = (50,53), MSE = (18,22)

(b) MAE = (6,12), MSE = (25,15)

(c) MAE = (4,8), MSE = (60,80)

(d) MAE = (1,5), MSE = (15,25)

In []:

In []: