神经网络习题

浙江大学

赵洲

题目

- **1. 以下关于神经网络的描述，哪一项是正确的?**

A. 神经网络只能用于监督学习
B. 激活函数的主要作用是引入非线性
C. 多层感知机（MLP）不包含隐藏层
D. 反向传播算法不适用于深层神经网络

解答

- **A. 神经网络只能用于监督学习**

- **错误**：
  - 神经网络不仅可以用于监督学习，还可以用于其他类型的学习任务。例如：
    - **无监督学习**：如自编码器（Autoencoders）和生成对抗网络（GANs）等。
    - **强化学习**：如深度强化学习中的深度Q网络（DQN）。
  - 因此，神经网络并不限于监督学习，也能处理无监督学习和强化学习问题。

解答

- **A. 神经网络只能用于监督学习**

- **错误**：
  - 神经网络不仅可以用于监督学习，还可以用于其他类型的学习任务。例如：
    - **无监督学习**：如自编码器（Autoencoders）和生成对抗网络（GANs）等。
    - **强化学习**：如深度强化学习中的深度Q网络（DQN）。
  - 因此，神经网络并不限于监督学习，也能处理无监督学习和强化学习问题。

解答

- **A. 神经网络只能用于监督学习**

- **错误**：
  - 神经网络不仅可以用于监督学习，还可以用于其他类型的学习任务。例如：
    - **无监督学习**：如自编码器（Autoencoders）和生成对抗网络（GANs）等。
    - **强化学习**：如深度强化学习中的深度Q网络（DQN）。
  - 因此，神经网络并不限于监督学习，也能处理无监督学习和强化学习问题。

解答

- **B. 激活函数的主要作用是引入非线性**

- **正确**:
  - 激活函数的主要作用就是引入非线性。没有激活函数，神经网络的每一层将仅仅是线性变换，整个网络的输出将是输入的线性组合，不论网络有多少层。因此，激活函数（如ReLU、Sigmoid、Tanh等）能够使神经网络学习并建模复杂的非线性关系。
  - 非线性的引入使得神经网络能够逼近任意复杂的函数，赋予其强大的表示能力。

题目

- **C. 多层感知机（MLP）不包含隐藏层**

- **错误**：
  - **多层感知机（MLP，Multilayer Perceptron）**是一个包含至少一个隐藏层的神经网络架构。MLP通常由输入层、一个或多个隐藏层、和输出层组成。隐藏层是多层感知机中的关键部分，能够帮助网络学习复杂的特征。
  - 因此，MLP必然包含隐藏层，而不是不包含。

解答

- **D. 反向传播算法不适用于深层神经网络**

- **错误**:
  - 反向传播算法（Backpropagation）是训练神经网络的一种常用算法，它适用于任何深度的神经网络，包括深层神经网络（DNN）。反向传播通过计算误差并将其通过网络反向传播来更新网络权重。在现代深度学习中，反向传播是训练深层神经网络的核心算法。
  - 事实上，反向传播是深度神经网络训练中不可或缺的一部分，因此它适用于深层神经网络。

题目

- **3.在神经网络中，梯度消失问题通常发生在以下哪种情况下?**

A. 网络的权重初始化得太小，导致输出的梯度几乎为零。

B. 激活函数使用了ReLU（Rectified Linear Unit），因为ReLU在负数区域的梯度为零。

C. 激活函数使用了Sigmoid或Tanh，尤其是在深层网络中。

D. 网络的训练数据量非常小，导致过拟合。

解答

- **A. 网络的权重初始化得太小，导致输出的梯度几乎为零。**

- **部分正确**：
  - 初始化权重过小确实可能导致梯度消失问题的加剧，但这主要影响的是网络训练的开始阶段。过小的权重使得输出值过于接近激活函数的饱和区域（例如Sigmoid或Tanh的极值区），导致梯度变得很小，进而影响梯度的传播。
  - 然而，单纯的权重初始化问题并非梯度消失的根本原因。权重初始化方法（如Xavier或He初始化）能有效缓解这个问题。
  - **结论**：这个选项部分成立，但它并不是梯度消失的主要原因。

解答

- **B. 激活函数使用了ReLU（Rectified Linear Unit），因为ReLU在负数区域的梯度为零。**

- **错误**：
  - **ReLU激活函数**的确在负数区域的梯度为零，但这并不属于**梯度消失问题**。ReLU函数在正数区域的梯度为常数（1），因此通常不会导致梯度消失。ReLU的主要问题是**"神经元死亡"**（即在训练过程中，部分神经元的输出永远为零），而不是梯度消失。
  - **结论**：ReLU并不引发梯度消失问题，虽然在负区间可能导致一些神经元"死亡"。

解答

- **C. 激活函数使用了Sigmoid或Tanh，尤其是在深层网络中。**
- **正确**：
  - Sigmoid和Tanh激活函数的**饱和性**是导致梯度消失的根本原因。它们在输入非常大或非常小时，其导数值非常接近于零，导致反向传播时梯度几乎为零，尤其在深层网络中，梯度会在层层传播时逐渐消失，从而无法有效更新网络参数。
  - Sigmoid函数的输出范围在(0, 1)之间，Tanh的输出范围在(-1, 1)之间，均容易导致梯度消失问题，特别是当输入远离零时。
  - **结论**：这是梯度消失问题的主要原因，尤其在深度神经网络中更为明显。

解答

- **D. 网络的训练数据量非常小，导致过拟合。**

- **错误**：
  - 训练数据量小会导致**过拟合**，即模型在训练集上表现很好，但在测试集上表现差。然而，**过拟合与梯度消失**是不同的概念。过拟合是指模型过度拟合训练数据，而梯度消失是指网络在训练过程中，梯度传播时逐渐变小，导致无法有效更新权重。
  - **结论**：训练数据量小导致的是过拟合问题，而不是梯度消失问题。

题目

- 判断1.
- 激活函数的引入使得神经网络能够处理非线性问题。

解答

- 正确

- 激活函数（如ReLU、Sigmoid、Tanh等）在神经网络中引入非线性，使得网络能够学习和表示复杂的非线性关系。如果没有激活函数，无论网络有多少层，整个网络仍然相当于一个线性变换，无法处理非线性问题。

题目

- 判断2.
- 在神经网络中，梯度消失问题只会在深层网络中出现。

解答

- 错误

- 虽然梯度消失问题在深层网络中更为常见，但它也可能在浅层网络中出现，尤其是在使用某些激活函数（如Sigmoid或Tanh）时。这些激活函数在输入值较大或较小时会导致梯度趋近于零，从而影响梯度传播。

题目

- 判断3.
- 在神经网络中，使用批量归一化（Batch Normalization）不仅可以加速训练过程，还能在一定程度上起到正则化的作用，减少过拟合的风险。

解答

- 正确

- 批量归一化（Batch Normalization, BN）是一种在神经网络训练过程中对每一层的输入进行标准化的技术。具体来说，BN通过将每一层的输入标准化为均值为0、方差为1的分布，并引入可学习的缩放和偏移参数.
- 不仅提高了训练效率，还通过引入噪声和稳定特征分布的方式，对模型起到了正则化的作用，降低了过拟合的可能性。

题目

1. 考虑一个简单的前馈神经网络，有一个输入层、一个隐藏层和一个输出层。输入层有2个神经元，隐藏层有2个神经元，输出层有1个神经元。激活函数均为Sigmoid函数。给定以下权重和偏置：

- 输入层到隐藏层的权重矩阵 $W_1 = \begin{bmatrix} 0.5 & -0.6 \\ 0.8 & 0.2 \end{bmatrix}$
- 隐藏层的偏置 $b_1 = \begin{bmatrix} 0.1 \\ -0.1 \end{bmatrix}$
- 隐藏层到输出层的权重矩阵 $W_2 = \begin{bmatrix} 1.0 \\ -1.0 \end{bmatrix}$
- 输出层的偏置 $b_2 = 0.0$

给定一个输入样本 $x = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$**，计算输出层的激活值。**

解答

解答：

步骤1：计算隐藏层的线性组合 z_1

$$z_1 = W_1 \cdot x + b_1$$

$$W_1 = \begin{bmatrix} 0.5 & -0.6 \\ 0.8 & 0.2 \end{bmatrix}, \quad x = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 0.1 \\ -0.1 \end{bmatrix}$$

$$W_1 \cdot x = \begin{bmatrix} 0.5 \times 1 + (-0.6) \times 2 \\ 0.8 \times 1 + 0.2 \times 2 \end{bmatrix} = \begin{bmatrix} 0.5 - 1.2 \\ 0.8 + 0.4 \end{bmatrix} = \begin{bmatrix} -0.7 \\ 1.2 \end{bmatrix}$$

$$z_1 = \begin{bmatrix} -0.7 \\ 1.2 \end{bmatrix} + \begin{bmatrix} 0.1 \\ -0.1 \end{bmatrix} = \begin{bmatrix} -0.6 \\ 1.1 \end{bmatrix}$$

解答

步骤2：应用Sigmoid激活函数计算隐藏层的输出 a_1

$$a_1 = \sigma(z_1) = \begin{bmatrix} \sigma(-0.6) \\ \sigma(1.1) \end{bmatrix}$$

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

$$\sigma(-0.6) = \frac{1}{1 + e^{0.6}} \approx \frac{1}{1 + 1.8221} \approx 0.3775$$

$$\sigma(1.1) = \frac{1}{1 + e^{-1.1}} \approx \frac{1}{1 + 0.3333} \approx 0.7503$$

$$a_1 = \begin{bmatrix} 0.3775 \\ 0.7503 \end{bmatrix}$$

解答

步骤3：计算输出层的线性组合 z_2

$$z_2 = W_2^T \cdot a_1 + b_2$$

$$W_2 = \begin{bmatrix} 1.0 \\ -1.0 \end{bmatrix}, \quad b_2 = 0.0$$

$$W_2^T \cdot a_1 = 1.0 \times 0.3775 + (-1.0) \times 0.7503 = 0.3775 - 0.7503 = -0.3728$$

$$z_2 = -0.3728 + 0.0 = -0.3728$$

解答

步骤4：应用Sigmoid激活函数计算输出层的激活值 a_2

$$a_2 = \sigma(z_2) = \frac{1}{1+e^{0.3728}} \approx \frac{1}{1+1.4516} \approx 0.4082$$

答案： 输出层的激活值约为 **0.4082**。

题目

- 2.考虑一个具有单隐藏层的神经网络，用于二分类问题。隐藏层包含3个神经元，输出层包含1个神经元。使用交叉熵损失函数和Sigmoid激活函数。假设在某一次迭代中，网络的预测值为0.8，真实标签为1。计算该样本的交叉熵损失。

解答

- **题目分析**

- 我们需要计算一个神经网络在一次迭代中的交叉熵损失。已知条件如下：

• 网络是一个二分类问题。

• 隐藏层有3个神经元，输出层有1个神经元。

• 使用Sigmoid激活函数和交叉熵损失函数。

• 网络的预测值为 $\hat{y}=0.8$，真实标签为 $y=1$。

解答

交叉熵损失函数（Cross-Entropy Loss）公式如下：

$$L = -\left[y\log(\hat{y}) + (1-y)\log(1-\hat{y})\right]$$

其中：

- y 是真实标签。
- $\hat{y}$ 是网络的预测值。

解答

解题步骤

1. 代入已知值:
    - 真实标签 $y = 1$
    - 预测值 $\hat{y} = 0.8$

解答

2. **计算交叉熵损失：**

$$L = -[1 \cdot \log(0.8) + (1-1) \cdot \log(1-0.8)]$$

由于 $(1-1)=0$，第二项的值为 0。所以损失函数简化为：

$$L = -\log(0.8)$$

解答

3. **计算对数值：**

$$\log(0.8) \approx -0.2231$$

因此，损失函数为：

$$L \approx -(-0.2231) = 0.2231$$

最终答案

该样本的交叉熵损失为 **0.2231**。

题目

3. 在一个简单的神经网络中，输入层有2个神经元，隐藏层有2个神经元，输出层有1个神经元。使用线性激活函数（即没有激活函数）。给定以下权重和偏置：

- 输入层到隐藏层的权重矩阵 $W_1 = \begin{bmatrix} 1 & -1 \\ 2 & 3 \end{bmatrix}$
- 隐藏层的偏置 $b_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$
- 隐藏层到输出层的权重矩阵 $W_2 = \begin{bmatrix} 4 \\ -2 \end{bmatrix}$
- 输出层的偏置 $b_2 = 0$

给定一个输入样本 $x = \begin{bmatrix} 3 \\ 1 \end{bmatrix}$，计算输出层的值。

解答

解答:

步骤1：计算隐藏层的线性组合 z_1

$$z_1 = W_1 \cdot x + b_1$$

$$W_1 = \begin{bmatrix} 1 & -1 \\ 2 & 3 \end{bmatrix}, \quad x = \begin{bmatrix} 3 \\ 1 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$W_1 \cdot x = \begin{bmatrix} 1 \times 3 + (-1) \times 1 \\ 2 \times 3 + 3 \times 1 \end{bmatrix} = \begin{bmatrix} 3 - 1 \\ 6 + 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 9 \end{bmatrix}$$

$$z_1 = \begin{bmatrix} 2 \\ 9 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 10 \end{bmatrix}$$

解答

步骤2：应用线性激活函数计算隐藏层的输出 a_1

由于使用线性激活函数，输出即为 z_1：

$$a_1 = z_1 = \begin{bmatrix} 2 \\ 10 \end{bmatrix}$$

解答

步骤3：计算输出层的线性组合 z_2

$$z_2 = W_2^T \cdot a_1 + b_2$$

$$W_2 = \begin{bmatrix} 4 \\ -2 \end{bmatrix}, \quad b_2 = 0$$

$$W_2^T \cdot a_1 = 4 \times 2 + (-2) \times 10 = 8 - 20 = -12$$

$$z_2 = -12 + 0 = -12$$

解答

步骤4：应用线性激活函数计算输出层的值 a_2

$$a_2 = z_2 = -12$$

答案： 输出层的值为 **-12**。

题目

一个简单的神经网络具有以下结构：

- **输入层**：包含两个输入节点 x_1 和 x_2。
- **隐藏层**：包含两个节点 h_1 和 h_2，激活函数为 ReLU。
- **输出层**：包含一个节点 y，激活函数为 Sigmoid。

网络的权重和偏置如下：

- 输入层到隐藏层：

  $$w_{11} = 0.4, \quad w_{12} = -0.7, \quad w_{21} = 0.2, \quad w_{22} = 0.5$$

  偏置：

  $$b_{h1} = 0.1, \quad b_{h2} = -0.2$$

- 隐藏层到输出层：

  $$w_{h1} = 0.6, \quad w_{h2} = -0.3$$

  偏置：

  $$b_y = 0.2$$

输入数据为 $x_1 = 1.0$，$x_2 = 2.0$。

要求：

1. 计算隐藏层节点 h_1 和 h_2 的激活值。
2. 计算输出层节点 y 的激活值。
3. 判断输出 y 是否超过阈值 0.5，并给出分类结果。

解答

1. 计算隐藏层节点的激活值

隐藏层激活函数为 ReLU:

$$ReLU(x) = \max(0, x)$$

计算 h_1:

$$h_1 = ReLU(w_{11} \cdot x_1 + w_{21} \cdot x_2 + b_{h1})$$

$$h_1 = ReLU(0.4 \cdot 1.0 + 0.2 \cdot 2.0 + 0.1)$$

$$h_1 = ReLU(0.4 + 0.4 + 0.1) = ReLU(0.9)$$

$$h_1 = 0.9$$

计算 h_2:

$$h_2 = ReLU(w_{12} \cdot x_1 + w_{22} \cdot x_2 + b_{h2})$$

$$h_2 = ReLU(-0.7 \cdot 1.0 + 0.5 \cdot 2.0 - 0.2)$$

$$h_2 = ReLU(-0.7 + 1.0 - 0.2) = ReLU(0.1)$$

$$h_2 = 0.1$$

解答

2. 计算输出层节点的激活值

输出层激活函数为 Sigmoid：

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

输出层的激活值：

$$y = \sigma(w_{h1} \cdot h_1 + w_{h2} \cdot h_2 + b_y)$$

$$y = \sigma(0.6 \cdot 0.9 + (-0.3) \cdot 0.1 + 0.2)$$

$$y = \sigma(0.54 - 0.03 + 0.2) = \sigma(0.71)$$

计算 Sigmoid：

$$y = \frac{1}{1+e^{-0.71}}$$

近似计算：

$$e^{-0.71} \approx 0.491$$

$$y \approx \frac{1}{1+0.491} \approx \frac{1}{1.491} \approx 0.671$$

解答

3. 判断输出 y 的分类结果

阈值为 0.5：

- 如果 $y > 0.5$，分类为正类；
- 如果 $y \leq 0.5$，分类为负类。

由于 $y = 0.671 > 0.5$，分类为**正类**。

解答

最终答案

1. 隐藏层激活值：
   $h_1 = 0.9$，$h_2 = 0.1$。
2. 输出层激活值：
   $y \approx 0.671$。
3. 分类结果：
   正类（$y > 0.5$）。

End