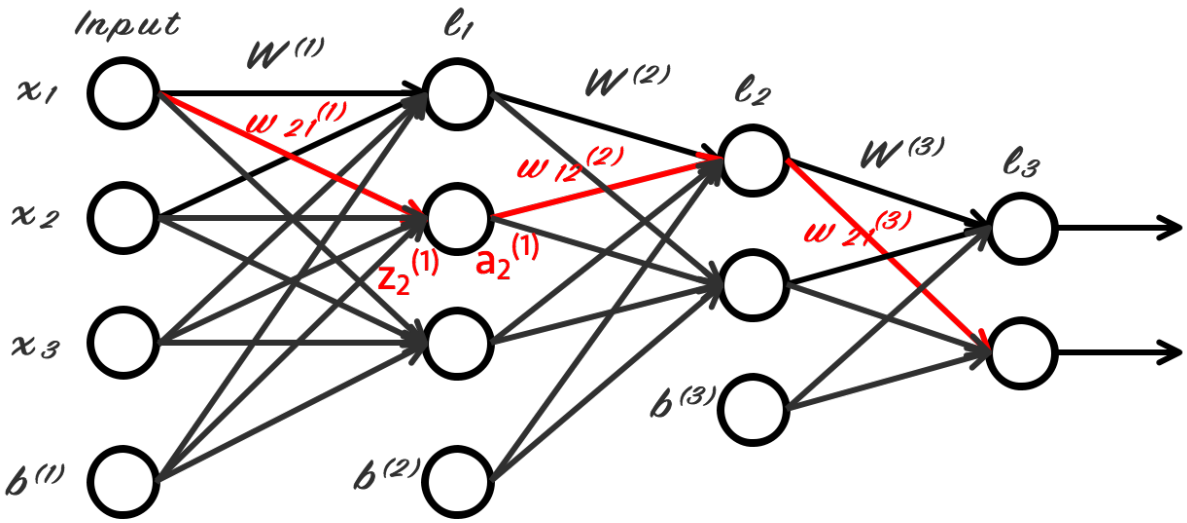
Deep Learning Technology and Application

Ge Li

Peking University

关于训练方法

前向传播计算



$$z^{(1)} = \begin{bmatrix} z_1^{(1)} \\ z_2^{(1)} \\ z_3^{(1)} \end{bmatrix} = \begin{bmatrix} w_{11}^{(1)} & w_{12}^{(1)} & w_{13}^{(1)} \\ w_{21}^{(1)} & w_{21}^{(1)} & w_{23}^{(1)} \\ w_{31}^{(1)} & w_{32}^{(1)} & w_{33}^{(1)} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} b_1^{(1)} \\ b_2^{(1)} \\ b_3^{(1)} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{3} w_{1i}^{(1)} x_i + b_1^{(1)} \\ \sum_{i=1}^{3} w_{2i}^{(1)} x_i + b_2^{(1)} \\ \sum_{i=1}^{3} w_{3i}^{(1)} x_i + b_3^{(1)} \end{bmatrix}$$

批量前向计算

$$z^{(1)} = \begin{bmatrix} z_1^{(1)} \\ z_2^{(1)} \\ z_3^{(1)} \end{bmatrix} = \begin{bmatrix} w_{11}^{(1)} & w_{12}^{(1)} & w_{13}^{(1)} \\ w_{21}^{(1)} & w_{21}^{(1)} & w_{23}^{(1)} \\ w_{31}^{(1)} & w_{32}^{(1)} & w_{33}^{(1)} \end{bmatrix} \begin{bmatrix} x_{11} & x_{21} \\ x_{12} & x_{22} \\ x_{13} & x_{23} \end{bmatrix} + \begin{bmatrix} b_1^{(1)} & b_1^{(1)} \\ b_2^{(1)} & b_2^{(1)} \\ b_3^{(1)} & b_3^{(1)} \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{3} w_{1i}^{(1)} x_{1i} + b_1^{(1)} & \sum_{i=1}^{3} w_{1i}^{(1)} x_{2i} + b_1^{(1)} \\ \sum_{i=1}^{3} w_{2i}^{(1)} x_{1i} + b_2^{(1)} & \sum_{i=1}^{3} w_{2i}^{(1)} x_{2i} + b_2^{(1)} \\ \sum_{i=1}^{3} w_{3i}^{(1)} x_{1i} + b_3^{(1)} & \sum_{i=1}^{3} w_{3i}^{(1)} x_{2i} + b_3^{(1)} \end{bmatrix}$$

用 A^{l+1} 表示与一个 Batch 对应的 $l+1$ 层所有神经元的输出值（其他变量含义相应可知），则：

$$Z^l = W^{(l)} A^{(l-1)} + B^{(l)} \qquad A^{(l)} = f(Z^{(l)})$$

批量权重更新

设一个 Batch 所对应的输入为：$X =$

$$\begin{bmatrix} x_{11} & x_{21} & \cdots & x_{m1} \\ x_{12} & x_{22} & \cdots & x_{m1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1n} & x_{2n} & \cdots & x_{mn} \end{bmatrix}$$

1. 设 $\Delta W_r^{(l)}$ 和 $\Delta b_r^{(l)}$ 分别为“当输入第 r 列数据时，第 l 层上的权重和偏置所对应的更新量”；则：
2. $\sum_{r=1}^{m} \Delta W_r^{(l)}$ 和 $\sum_{r=1}^{m} \Delta b_r^{(l)}$ 分别为“当输入完一个 Batch 的数据（X）时，第 l 层上的权重和偏置所对应的更新量的和”；
3. 这时，我们取上述更新量和的平均 $\frac{1}{m}\sum_{r=1}^{m} \Delta W_r^{(l)}$ 和 $\frac{1}{m}\sum_{r=1}^{m} \Delta W_r^{(l)}$，分别作为对第 l 层上的权重和偏置所应进行的更新。

批量梯度下降训练算法

设 X 为 M 列输入向量（一个 Batch），设 $\Delta W^{(l)}$ 和 $\Delta b^{(l)} = 0$ 分别为输入一个 Batch 后，第 l 层进行调整的权重和偏置的更新量；

1. 初始化：$\Delta W^{(l)} = 0$，$\Delta b^{(l)} = 0$
2. 计算权重和偏置的更新量矩阵（共 M 列），其中第 r 列分别为：$\Delta W_r^{(l)}$ 和 $\Delta b_r^{(l)}$；
3. 对上述两个矩阵，分别计算：$\frac{1}{m}\sum_{r=1}^{m}\Delta W_r^{(l)}$ 和 $\frac{1}{m}\sum_{r=1}^{m}\Delta W_r^{(l)}$
4. 利用上述结果进行权重更新：

$$W^{(l)} = W^{(l)} - \alpha(\frac{1}{m}\sum_{r=1}^{m}\Delta W_r^{(l)})$$
$$b^{(l)} = b^{(l)} - \alpha(\frac{1}{m}\sum_{r=1}^{m}\Delta b_r^{(l)})$$

循环执行上述过程，直到 Loss Function 输出值达到要求。

批量梯度下降的训练流程

设 X 为 M 列输入向量（一个 Batch），设 $\Delta W^{(l)}$ 和 $\Delta b^{(l)} = 0$ 分别为输入一个 Batch 后，第 l 层进行调整的权重和偏置的更新量；

1. 初始化：$\Delta W^{(l)} = 0$，$\Delta b^{(l)} = 0$
2. 计算权重和偏置的更新量矩阵（共 M 列），其中第 r 列分别为：$\Delta W_r^{(l)}$ 和 $\Delta b_r^{(l)}$；
3. 对上述两个矩阵，分别计算：$\frac{1}{m}\sum_{r=1}^{m}\Delta W_r^{(l)}$ 和 $\frac{1}{m}\sum_{r=1}^{m}\Delta W_r^{(l)}$
4. 利用上述结果进行权重更新：

$$W^{(l)} = W^{(l)} - \alpha(\frac{1}{m}\sum_{r=1}^{m}\Delta W_r^{(l)})$$
$$b^{(l)} = b^{(l)} - \alpha(\frac{1}{m}\sum_{r=1}^{m}\Delta b_r^{(l)})$$

循环执行上述过程，直到达到收敛条件。

各种训练方法

1. 批量梯度下降（Batch GD）
   1. 每轮权重更新所有样本都参与训练；
   2. 迭代多轮，直到达到收敛条件；
2. 随机梯度下降（SGD）
   1. 每轮权重更新只随机选取一个样本参与训练；
   2. 迭代多轮，达到收敛条件便可终止；
3. 小批量梯度下降（Mini-Batch SGD）
   1. 每轮权重更新（随机）取一部分样本参与训练；
   2. 迭代多轮，直到满足收敛条件；

Thanks.