智能之门

神经网络和深度学习入门

（基于Python的实现）

STEP 1 基本概念

第 2 章

神经网络中的三个基本概念

本部分通过讲解神经网络的三个基本概念，简要介绍神经网络基本的训练和工作原理，并着重介绍反向传播和梯度下降。我们先从简单的线性方式说起（只有加法和乘法），而且用代入数值的方式来消除对公式的恐惧心理。然后会说到分层的复杂（非线性）函数的反向传播，同样用数值代入方式手推反向过程。

2.1 通俗地理解三大概念

- **神经网络中的三大概念是：反向传播，梯度下降，损失函数。**

神经网络训练的最基本的思想是：先“猜”（初始化）一个结果（预测结果 a），观察它和训练集中含有的真实结果 y 之间的差距，然后调整策略，有依据地向正确的方向靠近。如此反复多次，直到预测结果和真实结果接近时，就结束训练。

在神经网络训练中，我们把“猜”叫做初始化，可以随机，也可以根据以前的经验给定初始值。即使是“猜”，也是有技术含量的。

2.1 通俗地理解三大概念

- **例1（猜数）：甲乙两人玩猜数的游戏，乙提前确定好一个数，由甲来猜。**
  - 目的：猜到乙确定的数字。
  - 初始化：甲猜5。
  - 前向计算：甲每次猜的新数字。
  - 损失函数：乙在根据甲猜的数来和自己心中想的数做比较，得出“大了”或“小了”的结论。
  - 反向传播：乙告诉甲“小了”、“大了”。
  - 梯度下降：甲根据乙的反馈中的含义自行调整下一轮的猜测值。

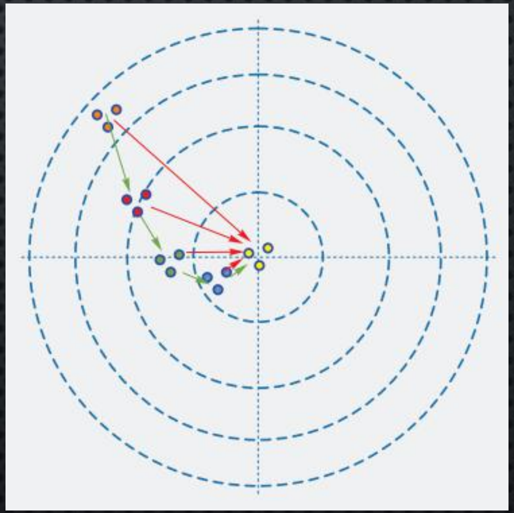
2.1 通俗地理解三大概念

- **例2（黑盒子）：假设有一个黑盒子如下图所示。我们只能看到输入和输出的数值，看不到里面的计算过程，同时黑盒子有个信息显示：我需要输出值是4。**
  - 目的：猜一个输入值，使黑盒子的输出是4。
  - 初始化：输入1。
  - 前向计算：黑盒子内部的数学逻辑。
  - 损失函数：在输出端，用输出值减4。
  - 反向传播：告诉猜数的人差值，包括正负号和值。
  - 梯度下降：在输入端，根据正负号和值，确定下一次的猜测值。

◆ **进阶玩法（破解黑盒子）：搭建神经网络进行训练。**

2.1 通俗地理解三大概念

- **例3（打靶）：小明拿了一支没有准星的步枪，射击100米外的靶子（如下图），打靶时会遇到各种干扰因素。**
  - 目的：打中靶心。
  - 初始化：随便打一枪，能上靶就行，但是要记住当时的步枪的姿态。
  - 前向计算：让子弹飞一会儿，击中靶子。
  - 损失函数：环数，偏离角度。
  - 反向传播：把靶子拉回来看。
  - 梯度下降：根据本次的偏差，调整步枪的射击角度。

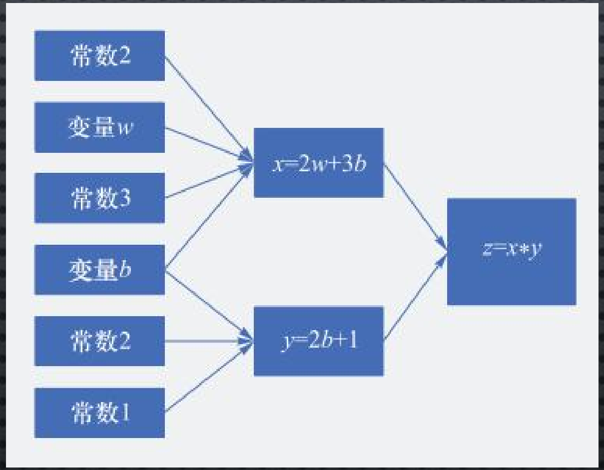


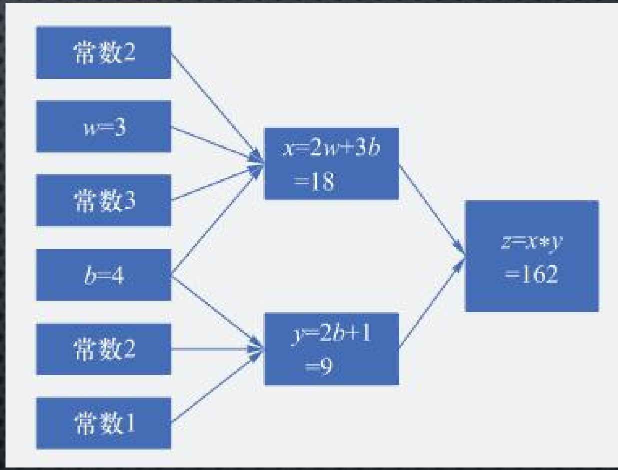
2.1 通俗地理解三大概念

- **总结反向传播与梯度下降的基本工作原理和步骤如下：**
  - 初始化。
  - 正向计算。
  - 损失函数：为我们提供了计算损失的方法。
  - 梯度下降：在损失函数基础上向着损失最小的点靠近，从而指引了网络权重调整的方向。
  - 反向传播：把损失值反向传给神经网络的各层，让各层都可以根据损失值反向调整权重。
  - 重复正向计算过程，直到精度满足要求（比如损失函数值小于 0.001）。

2.2 线性反向传播

假设有一个函数：$z = xy$，其中$\begin{cases} x = 2w + 3b \\ y = 2b + 1 \end{cases}$。注意这里$x, y, z$不是变量，只是计算结果；$w, b$才是变量，如左图。





- **正向计算**
  - 当$w = 3$，$b = 4$时，计算得到$x = 18$，$y = 9$，$z = 162$，如右图。

2.2 线性反向传播

➢ 反向传播：求w的偏导数

- 链式法则：因为 $z = xy$，其中$\begin{cases} x = 2w + 3b \\ y = 2b + 1 \end{cases}$，故而

$$\frac{\partial z}{\partial w} = \frac{\partial z}{\partial x} \cdot \frac{\partial x}{\partial w} = y \cdot 2 = 18$$

- 直接求导：$z = xy = (2w + 3b)(2b + 1) = 4wb + 2w + 6b^2 + 3b$，故而

$$\frac{\partial z}{\partial w} = 4b + 2 = 16 + 2 = 18$$

- 两种方法的运算结果一致。

2.2 线性反向传播

➢ **求w的近似变化值**

- 目标：$z = 150$。
- 由前述梯度计算结果有：

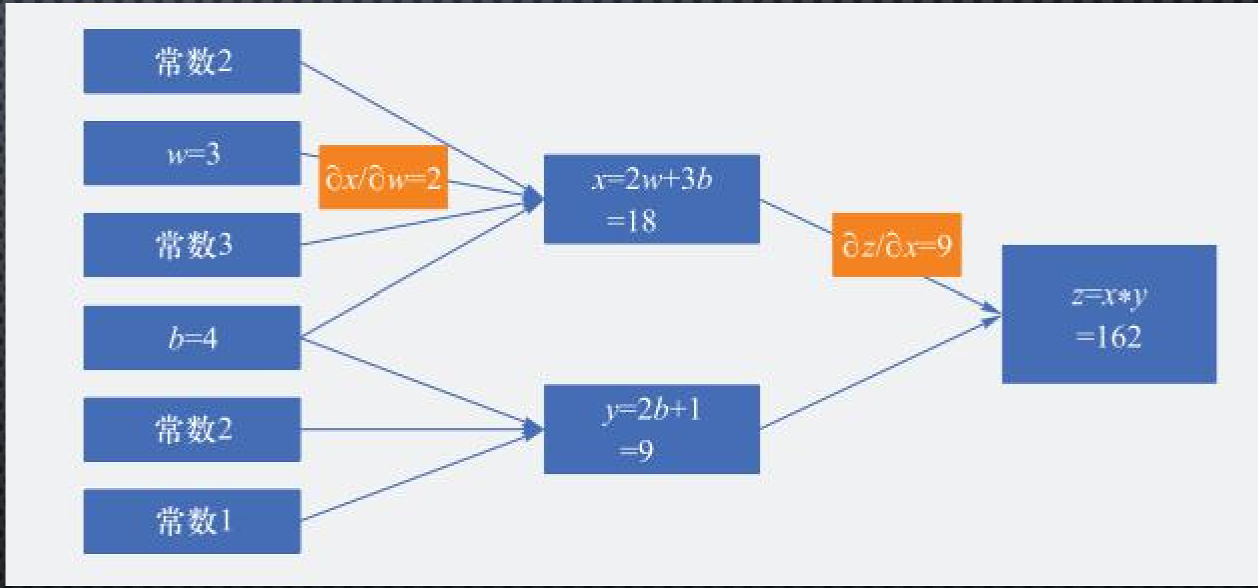
$$\Delta z = \frac{\partial z}{\partial w} \cdot \Delta w$$

$$\Delta w = \frac{\Delta z}{18} = 0.6667$$

$$w = 2.3333$$

$$z = 150.0003$$

- 计算结果与目标十分接近。



2.2 线性反向传播

➢ 反向传播：求b的偏导数

- 链式法则：因为 $z = xy$，其中$\begin{cases} x = 2w + 3b \\ y = 2b + 1 \end{cases}$，故而

$$\frac{\partial z}{\partial b} = \frac{\partial z}{\partial x} \cdot \frac{\partial x}{\partial b} + \frac{\partial z}{\partial y} \cdot \frac{\partial y}{\partial b} = y \cdot 3 + x \cdot 2 = 63$$

- 直接求导：$z = xy = (2w + 3b)(2b + 1) = 4wb + 2w + 6b^2 + 3b$，故而

$$\frac{\partial z}{\partial b} = 4w + 12b + 3 = 12 + 48 + 3 = 63$$

- 两种方法的运算结果一致。

2.2 线性反向传播

- **求b的近似变化值**
  - 目标：$z = 150$。
  - 由前述梯度计算结果有：

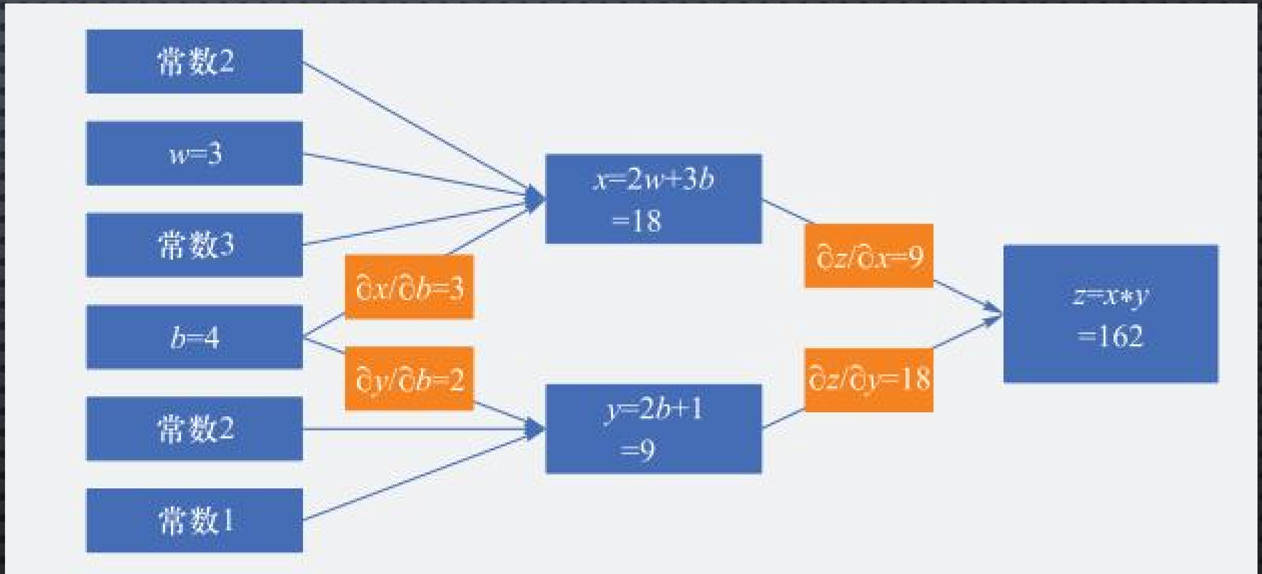
$$\Delta z = \frac{\partial z}{\partial b} \cdot \Delta b$$

$$\Delta b = \frac{\Delta z}{63} = 0.1905$$

$$w = 3.8095$$

$$z = 150.2162$$

  - 计算结果与目标十分接近。



2.2 线性反向传播

➢ 同时求w, b的近似变化值

- 目标：$z = 150$。
- 不妨设误差的一半由w产生，一半由b产生，则有：

$$\Delta w = \frac{\Delta z/2}{18} = 0.333, \qquad \Delta b = \frac{\Delta z/2}{63} = 0.095$$

$$w = 2.667, \qquad b = 3.905, \qquad z = 150.2$$

- 计算结果与目标十分接近。

2.3 非线性反向传播

设 $y=f(x)$，其中$\begin{cases} y=c \\ c=\sqrt{b} \\ b=\ln a \\ a=x^2 \end{cases}, 1<x\leq 10, 0<y<2.15$。

➢ 正向过程

- 当 $x=2$ 时，计算得到

$$a=x^2=4,\qquad b=\ln a=1.386,\qquad c=\sqrt{b}=1.177$$

➢ 反向过程

- 欲得到输出$y=2.13$，回传误差，之后即可更新输入

$$\Delta c=c-y,\qquad \Delta b=\Delta c\cdot 2\sqrt{b},\qquad \Delta a=\Delta b\cdot a,\qquad \Delta x=\frac{\Delta a}{2x}$$

2.3 非线性反向传播

- **数学解析解**

$$c = \sqrt{b} = \sqrt{\ln a} = \sqrt{\ln x^2} = 2.13,\ \ x = 9.6653$$

- **梯度迭代解**

$$\Delta c = c - y$$

$$\frac{dc}{db} = \frac{1}{2\sqrt{b}} \approx \frac{\Delta c}{\Delta b}, \qquad \Delta b = \Delta c \cdot 2\sqrt{b}$$

$$\frac{db}{da} = \frac{1}{a} \approx \frac{\Delta b}{\Delta a}, \qquad \Delta a = \Delta b \cdot a$$

$$\frac{da}{dx} = 2x \approx \frac{\Delta a}{\Delta x}, \qquad \Delta x = \frac{\Delta a}{2x}$$

- 给定初始值 $x = 2$，经过五轮迭代更新之后，得到 $c = 2.129577$，十分接近目标结果。

2.4 梯度下降

- **在自然界中，梯度下降的最好例子，就是泉水下山的过程：**
  - 水受重力影响，会在当前位置，沿着最陡峭的方向流动，有时会形成瀑布（梯度下降）；
  - 水流下山的路径不是唯一的，在同一个地点，有可能有多个位置具有同样的陡峭程度，而造成了分流（可以得到多个解）；
  - 遇到坑洼地区，有可能形成湖泊而终止下山过程（得到局部最优解而非全局最优解）。
- **梯度下降的数学公式**

$$\theta_{n+1} = \theta_n - \eta \cdot \nabla J(\theta_n)$$

  - 三要素：当前点、方向、步长。
  - 梯度：函数当前位置的最快上升点。
  - 下降：与导数相反的方向。

2.4 梯度下降

➢ 单变量函数的梯度下降

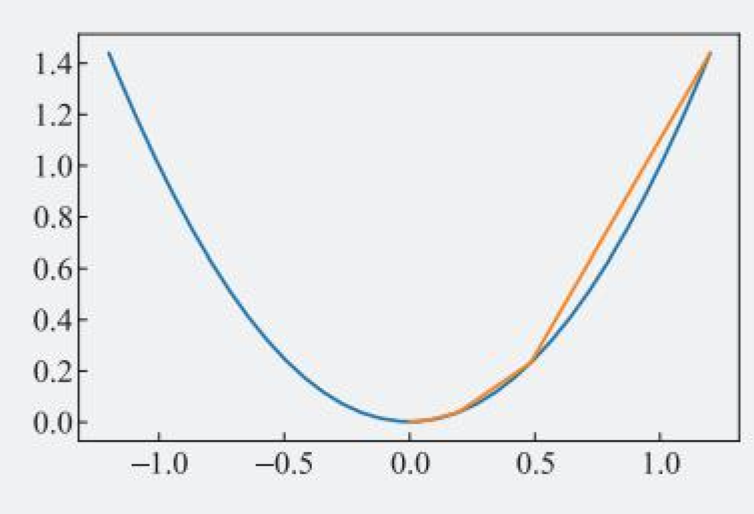
- 假设单变量函数 $J(x) = x^2$，其微分为 $J'(x) = 2x$。
- 初始位置 $x_0 = 1.2$，学习率 $\eta = 0.3$，迭代终止条件为 $J(x) < 0.01$，迭代结果如下图。

x=0.480000，y=0.230400

x=0.192000，y=0.036864

x=0.076800，y=0.005898

x=0.030720，y=0.000944



2.4 梯度下降

➢ 双变量函数的梯度下降

- 假设双变量函数 $J(x,y)=x^2+\sin^2 y$，两个一阶偏导数为

$$\frac{\partial J(x,y)}{\partial x}=2x,\qquad \frac{\partial J(x,y)}{\partial y}=2\sin y\cos y$$

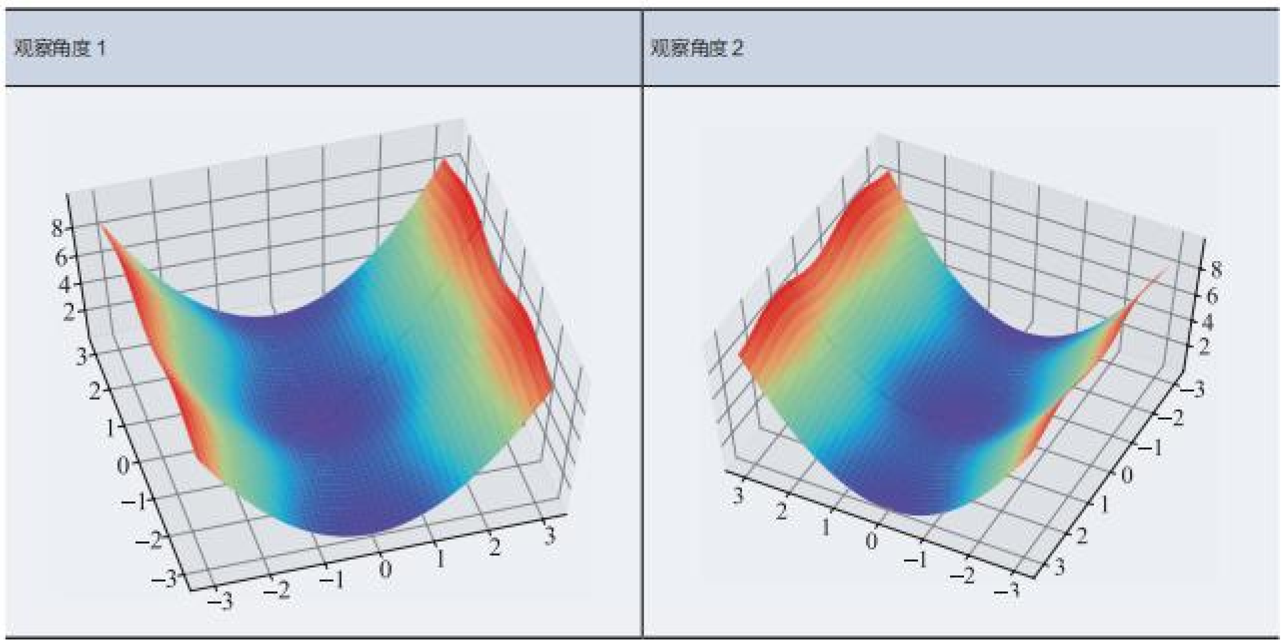
- 初始位置 $(x_0,y_0)=(3,1)$，学习率 $\eta=0.1$，迭代终止条件为 $J(x,y)<0.01$，迭代过程如下图。

迭代次数	x	y	$J(x,y)$
1	3	1	9.708 073
2	2.4	0.909 070	6.382 415
...	...	...	...
15	0.105 553	0.063 481	0.015 166
16	0.084 442	0.050 819	0.009 711

2.4 梯度下降

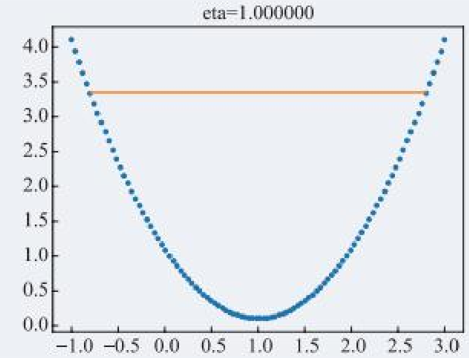
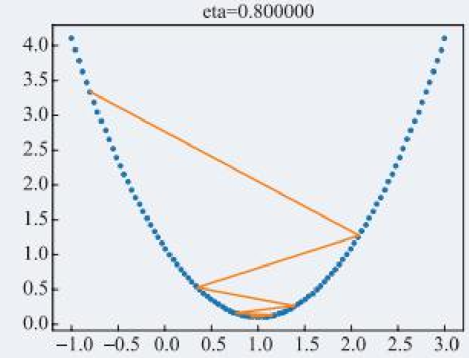
➢ **双变量函数的梯度下降**

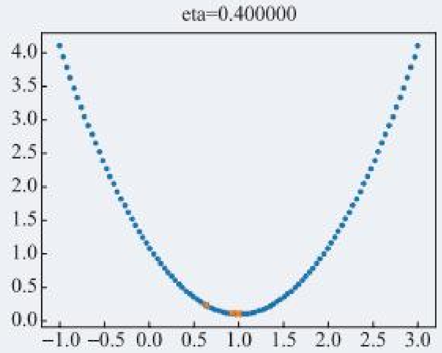
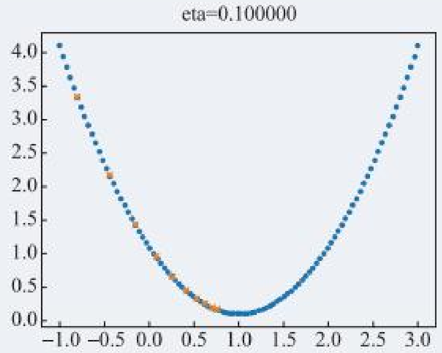
- 可视化结果：梯度下降的过程，从红色的高地一直沿着坡度向下走，直到蓝色的洼地。



2.4 梯度下降

➢ 学习率的选择

1.0		学习率太大，迭代的情况很糟糕，在一条水平线上跳来跳去，永远也不能下降
0.8		学习率大，会有这种左右跳跃的情况发生，不利于神经网络的训练

0.4		学习率合适，损失值会从单侧下降，4 步以后基本接近了理想值
0.1		学习率较小，损失值会从单侧下降，但下降速度非常慢，10 步了还没有到达理想状态

THE END

谢谢！