

扫码签到



人工神经网络 与 梯度下降

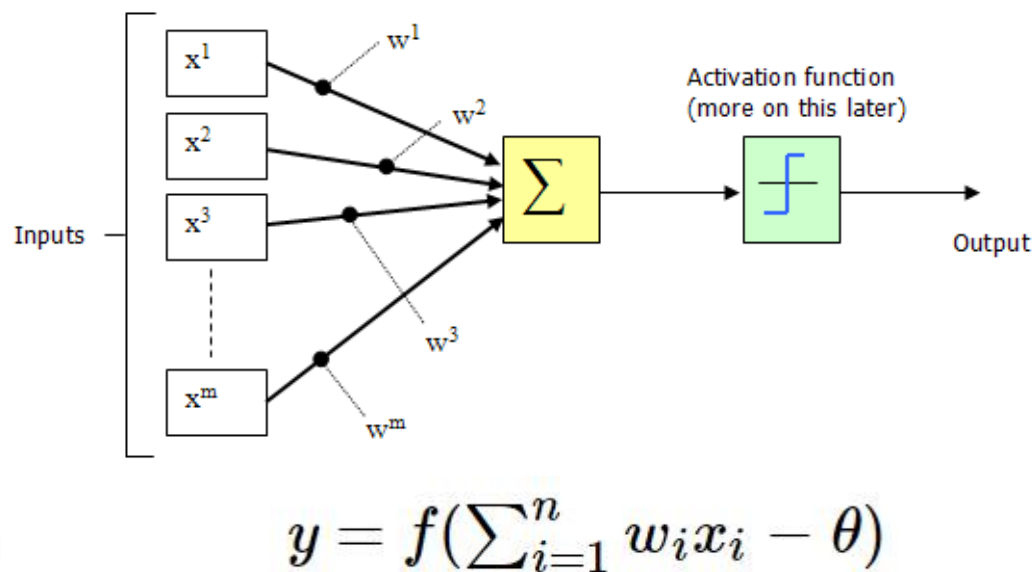
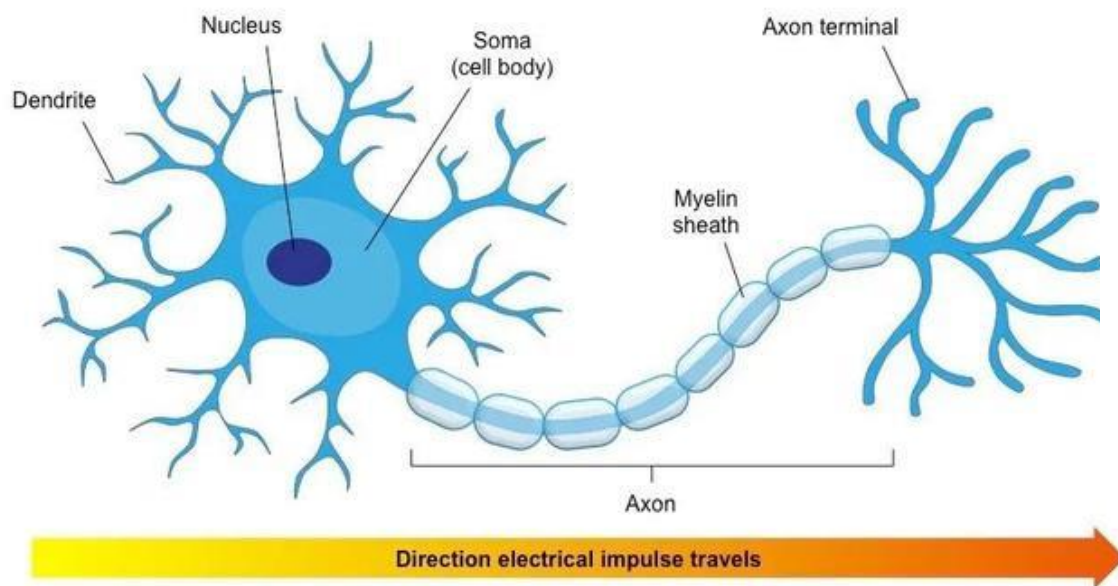


目录

- 1 人工神经网络介绍
 - 神经网络概述
 - 梯度下降
 - 利用梯度下降优化神经网络
- 2 实验任务

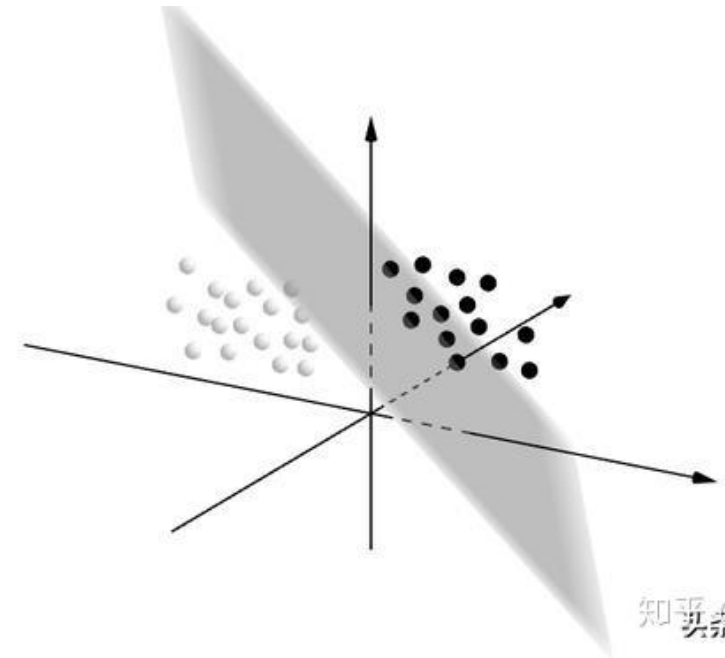
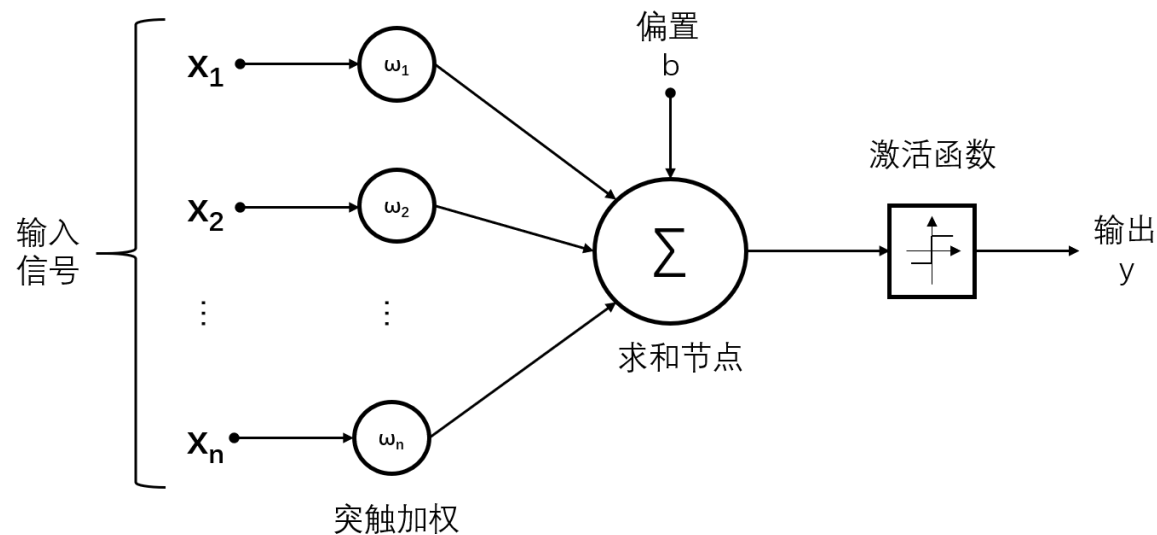
神经网络概述

- 神经网络采用了仿生学的思想，通过模拟生物神经网络的结构和功能来实现建模



单层感知机

- 由于M-P神经元模型参数需要事先设定好，为了能够自适应学习出所需要的参数，有研究人员就提出了单层感知机(Perceptron)
- 感知机的基本公式为： $y(x) = \text{sign}(w x + b)$





损失函数

- 作用：为了衡量网络表现是否良好，并为之后的网络参数优化提供指导。
- 常见的用在分类任务上的损失函数：

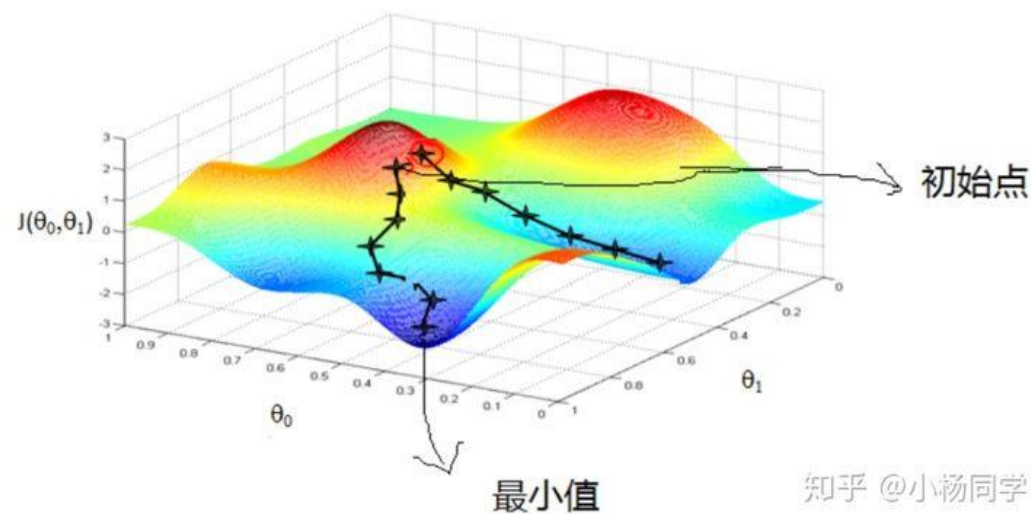
- 均方误差(MSE):
$$L_{MSE} = \frac{1}{2n} \sum_{i=1}^n (y^{(i)} - \hat{y}^{(i)})^2$$

- 交叉熵:
$$L_{CE} = - \sum_i^n y^{(i)} \log \hat{y}^{(i)} + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)})$$



梯度下降

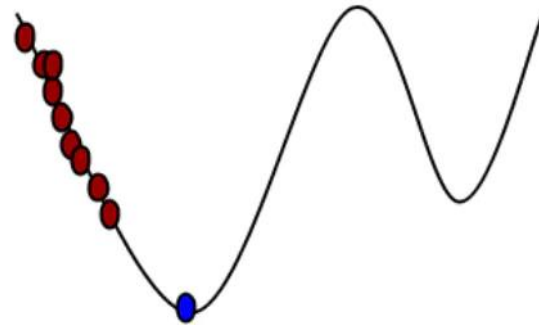
- 梯度定义：梯度是一个向量，表示某一函数在该点出的方向导数沿着该方向取得最大值。
- 也就是说该点处沿着梯度的方向变化最快，变化率最大
 - 沿着梯度方向容易找到函数最大值
 - 沿着梯度方向的反方向，容易找到函数最小值
- 梯度下降的一般公式为：
- $\theta = \theta - \eta \Delta L$



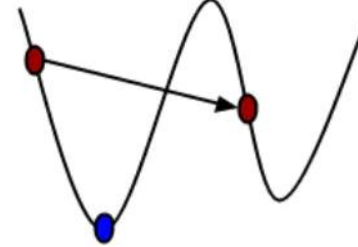


为什么要设置学习率

- 学习率限制了下一步能到达的地方
- 如果学习率太小，可能很难达到最小值
- 如果学习率太大，则会错过最小值，无法收敛



very small learning
rate needs lots of
steps



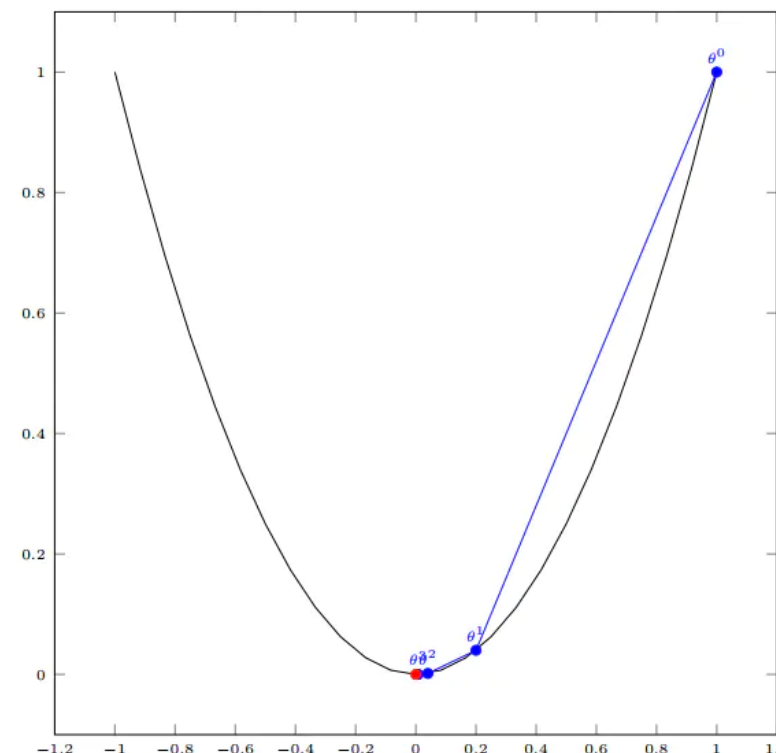
too big learning rate:
missed the minimum



单变量的梯度下降

- 假设有一个单变量的函数 $J(\theta) = \theta^2$ ，我们向找该函数的最小值
- 该函数的微分计算公式为： $J'(\theta) = 2\theta$
- 我们可以初始化起点 $\theta^0 = 1$
- 迭代过程为：

$$\begin{aligned}\theta^0 &= 1 \\ \theta^1 &= \theta^0 - \alpha * J'(\theta^0) \\ &= 1 - 0.4 * 2 \\ &= 0.2 \\ \theta^2 &= \theta^1 - \alpha * J'(\theta^1) \\ &= 0.04 \\ \theta^3 &= 0.008 \\ \theta^4 &= 0.0016\end{aligned}$$





梯度下降优化神经网络

- 假设网络的参数为 W 与 b ，采取的损失函数为 L
- 可以计算损失函数对 W 和 b 的偏导分别为 $\frac{\partial L}{\partial W}$, $\frac{\partial L}{\partial b}$
- 更新网络参数，公式为： $W = W - \eta \frac{\partial L}{\partial W}$, $b = b - \eta \frac{\partial L}{\partial b}$
- 其中 η 为学习率



举例

1. 假定网络为单层感知机，且没有激活层，没有偏置，此时，网络输出为 $y = XW$
2. 设置损失函数为 L_{MSE} ，并随机初始化网络参数 W
3. 当满足终止条件时，终止优化，否则继续
4. 计算网络输出 $y = XW$ ，以及损失 $L_{MSE} = \frac{1}{N} (XW - Y)^T (XW - Y)$
5. 求导可得 $\frac{\partial L_{MSE}}{\partial W} = \frac{1}{N} X^T (XW - Y)$
6. 根据 $W = W - \eta \frac{\partial L_{MSE}}{\partial W}$ 更新参数 W
7. 跳转到3



实验任务

- 在给定文本数据集完成文本情感分类训练，在测试集完成测试，计算准确率。
- 要求
 - 文本的特征可以使用TF或TF-IDF（可以使用sklearn库提取特征）
 - 设计合适的网络结构，选择合适的损失函数利用训练集完成网络训练，并在测试集上计算准确率
 - 需提交实验报告+代码
 - 实验报告应包含损失的可视化展示，以及学习率对准确率影响的可视化展示



参考

- 矩阵求导: <https://zhuanlan.zhihu.com/p/137702347>
- 矩阵运算库Numpy教程:
<https://www.runoob.com/numpy/numpy-tutorial.html>
- Matplotlib可视化教程:
<https://www.runoob.com/matplotlib/matplotlib-tutorial.html>