

20220510-书&机器学习

1.学习内容

1.1 索拉里斯星

1.2 机器学习

多层感知机的简单实现

几种常用的激活函数

ReLU函数

Sigmoid函数

tanh函数

模型泛化

2.结果描述

1.学习内容

1.1 索拉里斯星

向宇宙进发的那一刻，我们已经准备好应对一切挑战，无论是孤独、磨难、战斗还是牺牲。我们出于谦虚，没有到处张扬，但有时暗自思付，觉得自己的确非常优秀。而与此同时，与此同时，我们并没有征服宇宙的野心，而只是想把地球的边界拓展到宇宙的尽头。有些星球也许如撒哈拉沙漠般荒凉寂寥，有些也许像极地般冰封万里，有些也许像巴西雨林般葱茂神秘。但我们充满人文情怀，心地高贵，我们不想奴役其他种族，而只希望传递自己的价值观，与他们取长补短。我们自认为是沟通星际的神圣骑士。而这又是一个谎言。我们四处寻找，想要的其实仅仅是所谓的‘人’。我们不需要其他的世界。我们要的只是一面镜子。完全不同的世界会让我们不知所措。现有的世界已经让我们感到窒息。一方面，我们寄希望在其他星球找到一副完美无缺的图景，一些比我们更完善的文明；另一方面，我们又希望在宇宙深处的某个角落，找到我们原始蒙昧时代的影子。但与此同时，尽管我们无法接受，拼命辩驳，但从地球带来的绝非纯粹的美德和人类的典范！我们来的这些人，也都是有七情六欲的普通人，而当我们心底里阴暗的一面幻化成真时——那是我们竭力想逃避的部分——我们就无法再平静面对了！

1.2 机器学习

多层感知机的简单实现

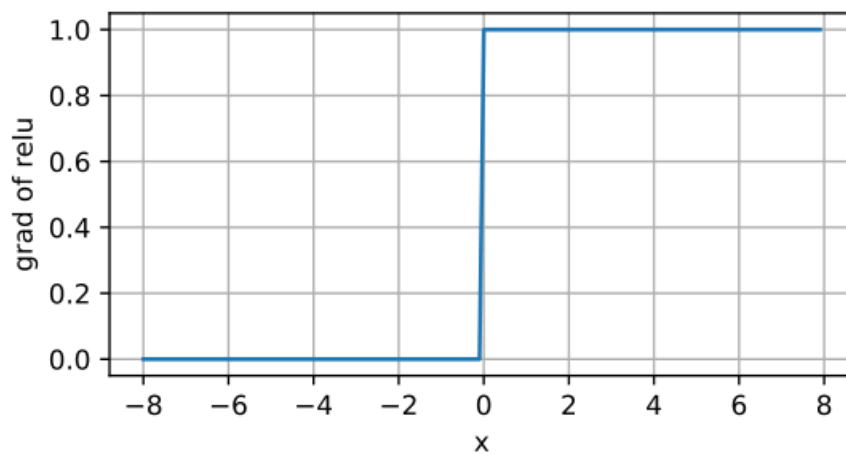
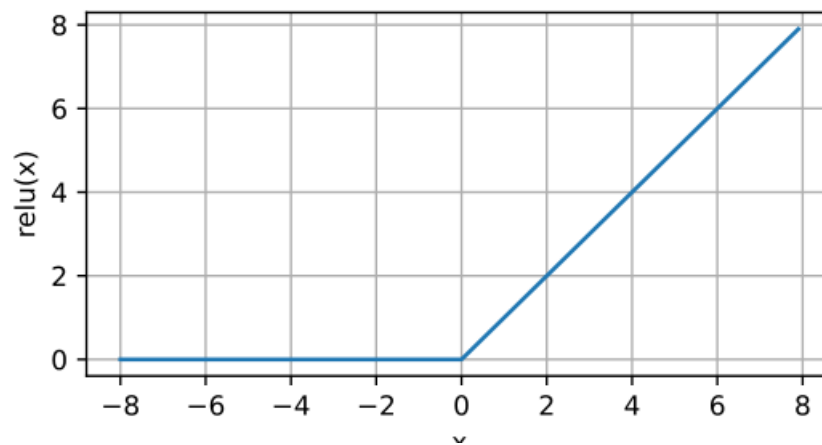
```
1  import torch
2  from torch import nn
3  from d2l import torch as d2l
4
5  #初始化模型参数
6  #网络包含256个隐藏单元
7  #通常选择2的若干次幂作为层的宽度，因为这样更适应内存在硬件中的分配和寻址方式
8  batch_size=256
9  train_iter,test_iter=d2l.load_data_fashion_mnist(batch_size)
10
11  num_inputs,num_outputs,num_hiddens=784,10,256
12  W1=nn.Parameter(torch.randn(num_inputs,num_hiddens,requires_grad=True)*0.01)
13  b1=nn.Parameter(torch.zeros(num_hiddens,requires_grad=True))
14  W2=nn.Parameter(torch.randn(num_hiddens,num_outputs,requires_grad=True)*0.01)
15  b2=nn.Parameter(torch.zeros(num_outputs,requires_grad=True))
16  params=[W1,b1,W2,b2]
17
18  #激活函数
19  def relu(X):
20      a=torch.zeros_like(X)
21      return torch.max(X,a)
22
23  #模型
24  def net(X):
25      X=X.reshape((-1,num_inputs))
26      H=relu(X@W1+b1) #这里@代表矩阵乘法
27      return (H@W2+b2)
28
29  #损失函数
30  loss=nn.CrossEntropyLoss(reduction='none')
31
32  #训练
33  num_epochs,lr=10,0.1
34  updater=torch.optim.SGD(params,lr=lr)
35  d2l.train_ch3(net,train_iter,test_iter,loss,num_epochs,updater)
36
37  d2l.predict_ch3(net,test_iter)
```

几种常用的激活函数

ReLU函数

ReLU (Rectified linear unit) 函数是一种表现良好、较受欢迎的激活函数。通过将相应的活性值设为0，仅保留正元素并丢弃所有负元素。默认当输入为0时导数为0

$$\text{ReLU}(x) = \max(x, 0)$$



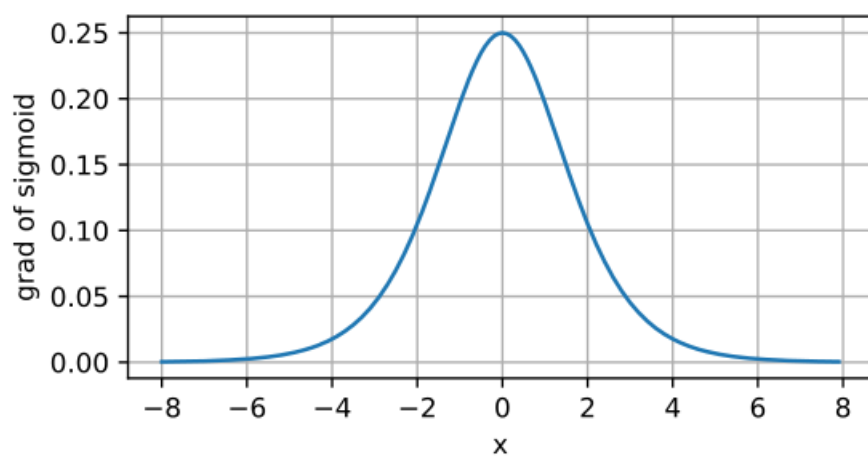
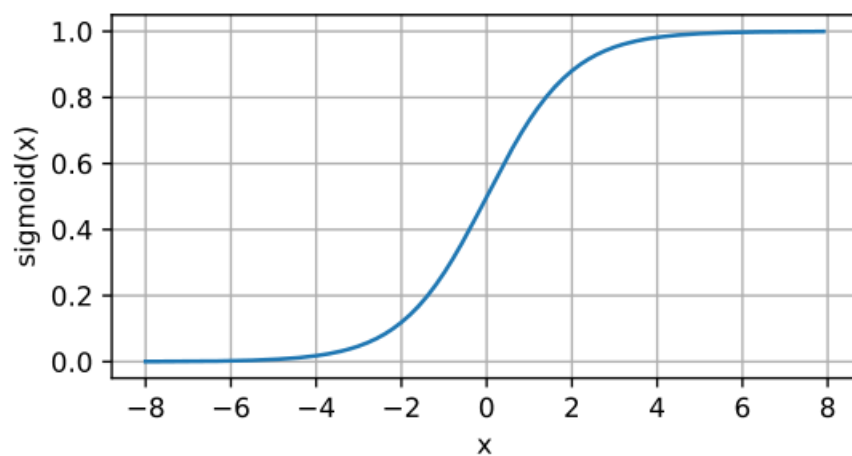
ReLU函数有许多变体，包括参数化ReLU函数。该变体添加了一个线性项，即使参数是负的，某些信息仍然可以通过：

$$\text{pReLU}(x) = \max(0, x) + \alpha \min(0, x)$$

Sigmoid函数

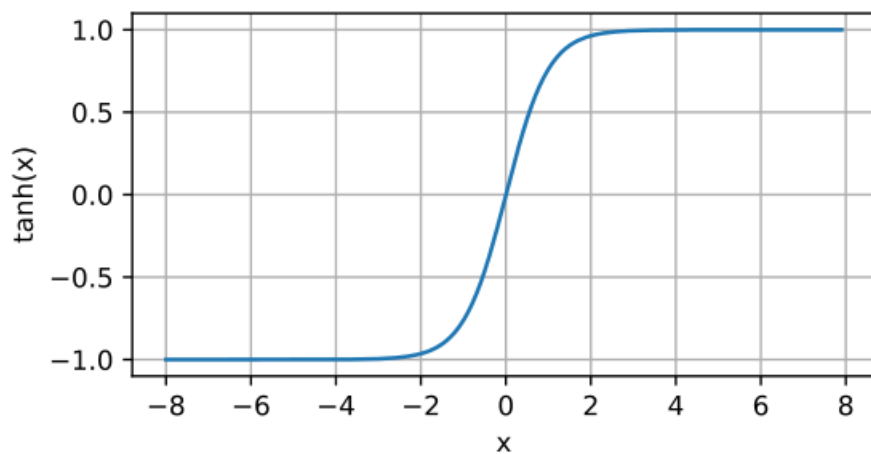
sigmoid函数将输入压缩转换到(0,1)之间。sigmoid函数在隐藏层中已经较少使用，在大部分时候被更简单、更容易训练的ReLU所取代

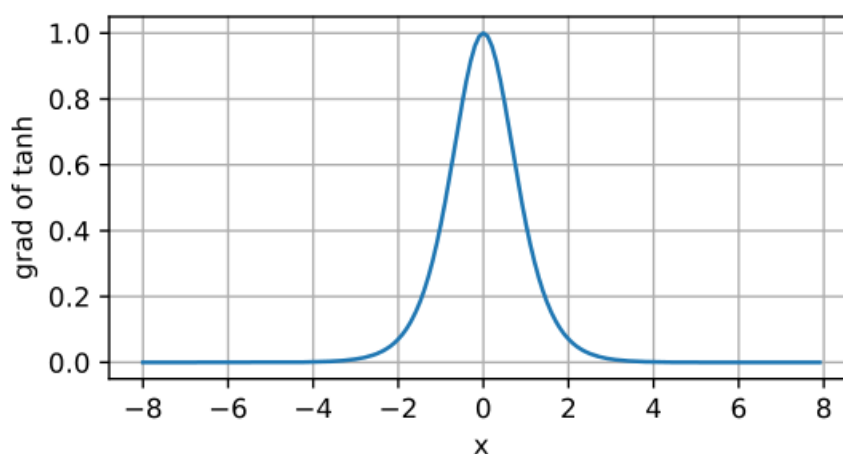
$$\text{sigmoid}(x) = 1 / (1 + \exp(-x))$$



tanh函数

tanh函数将输入压缩转换到 $(-1,1)$ 之间。 $\text{tanh}(x) = (1 - \exp(-2x)) / (1 + \exp(-2x))$





模型泛化

影响模型泛化的几个主要因素：

- 1.可调整参数的数量。当可调整参数的数量（或称自由度）很大时，模型往往更容易过拟合
- 2.参数采用的值。当权重的取值范围较大时，模型可能更容易过拟合
- 3.训练样本的数量。即使模型很简单，也很容易过拟合只包含一两个样本的数据集，而过拟合一个有几百万样本的数据集则需要一个极其灵活的模型

2.结果描述

今天大部分时间都在看一本叫《索拉里斯星》的书，是一个东欧的作家在上世纪六十年代写的。虽然年代久远，但看起来丝毫不觉得有种复古的感觉。主要讲的是几个科学家一个环绕在一颗地外行星的空间站发生的各种光怪陆离的故事。由于想象力比较贫乏，书中大部分关于那颗行星的奇幻之处的描写都没仔细看（可能是整本书最为吸引人的地方），只着重看了下人物的对话跟主线剧情。看完之后比较大的感慨便是，人类所谓的寻找外星人，很多时候可能希望找的是像人类一样的外星人，我们也常会将人类或地球生物的特点无限放大使之成为我们衡量地外生命存在与否的标尺。然而宇宙是如此浩渺，我们如何敢断底不会存在某种完全超脱我们想象的生命形态，而一旦他们以这种形式呈现在我们面前，我们可能无从建立对它的认知。索拉里斯星的大海可能就是这样一种超脱我们认知的生命形态。它不仅变幻莫测，还能提取出人类的记忆深处的某种念想并在我们面前建立一个具有自主生命和意识的映射。书中不同的科学家乃至幻想家都对索拉里斯星大海的生命形态有着各种各样的猜想，人类也实际开展了多轮考察探测，很多人甚至为此付出了生命的代价，然而直到全书结束，仍旧没有一种理论能有哪怕一丁点的把握进行断言。主人公在最后提出了一个冷冰冰的疑问，人类迄今为止所记录的种种关于索拉里斯星大海的活动模式，会不会只是这个生命在孩提时期种种无意识的随机行为，而当我们试图从这样一种随机行为中总结出某种有意义的模式或结论时，注定是徒劳无用的。我们可能压根就与这种生命存在于不同的维度。总体而言，作为一本科幻小说，《索拉里斯星》让我在某种程度上又体验了一下初读三体时的那种惊叹感，虽然读的过程多少有点囫圇吞枣。

临睡觉前还装模做样地看了会李沐的书，跟着实现了一个简单的多层感知机。代码看起来很简单，可是由于前面的章节没细读，导致对其中很多封装好的函数的调用都有点不明所以，依旧只能说学了个寂寞。明天争取把多层感知机跟深度学习计算两部分“看”完，并且好好研究一下一些常用函数的实现，之后应该就可以继续卷积神经网络的学习，加油，屏住。