链滴

深度学习面试 79 题：涵盖深度学习所有考点（51-65）| 文末彩蛋

作者：julyedu

原文链接：https://ld246.com/article/1623231460732

来源网站：链滴

51、什么是fine-tuning?

在实践中，由于数据集不够大，很少有人从头开始训练网络。常见的做法是使用预训练的网络（例如在mageNet上训练的分类1000类的网络）来重新fine-tuning（也叫微调），或者当做特征提取器。

以下是常见的两类迁移学习场景：

1 卷积网络当做特征提取器。使用在ImageNet上预训练的网络，去掉最后的全连接层，剩余部分当特征提取器（例如AlexNet在最后分类器前，是4096维的特征向量）。这样提取的特征叫做CNN cods。得到这样的特征后，可以使用线性分类器（Liner SVM、Softmax等）来分类图像。

2 Fine-tuning卷积网络。替换掉网络的输入层（数据），使用新的数据继续训练。Fine-tune时可以择fine-tune全部层或部分层。通常，前面的层提取的是图像的通用特征（generic features）（例如缘检测，色彩检测），这些特征对许多任务都有用。后面的层提取的是与特定类别有关的特征，因此fine-tune时常常只需要Fine-tuning后面的层。

52、什么是边框回归Bounding-Box regression，以及什么要做、怎么做?

这个问题可以牵扯出不少问题，比如

为什么要边框回归?

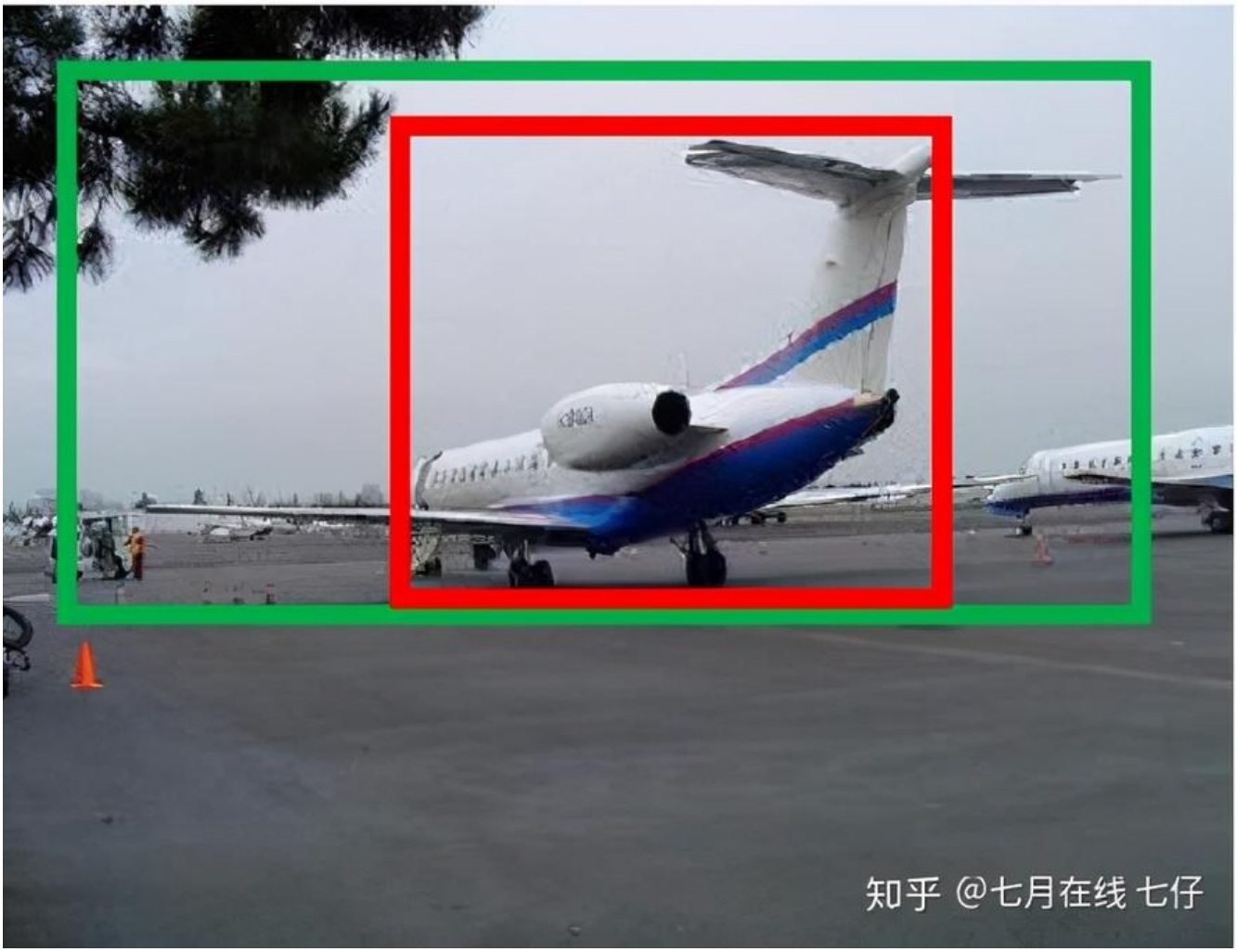
什么是边框回归?

边框回归怎么做的?

边框回归为什么宽高，坐标会设计这种形式?

为什么边框回归只能微调，在离真实值Ground Truth近的时候才能生效?

如图1所示，绿色的框表示真实值Ground Truth, 红色的框为Selective Search提取的候选区域/框Regon Proposal。那么即便红色的框被分类器识别为飞机，但是由于红色的框定位不准(IoU<0.5)， 这图也相当于没有正确的检测出飞机。



如果我们能对红色的框进行微调fine-tuning，使得经过微调后的窗口跟Ground Truth 更接近， 这岂不是定位会更准确。 而Bounding-box regression 就是用来微调这个窗口的。

53、请阐述下Selective Search的主要思想

1 使用一种过分割手段，将图像分割成小区域 (1k~2k 个)

2 查看现有小区域，按照合并规则合并可能性最高的相邻两个区域。重复直到整张图像合并成一个区位置

3 输出所有曾经存在过的区域，所谓候选区域

其中合并规则如下： 优先合并以下四种区域：

①颜色（颜色直方图）相近的

②纹理（梯度直方图）相近的

③合并后总面积小的： 保证合并操作的尺度较为均匀，避免一个大区域陆续“吃掉”其他小区域 （ ：设有区域a-b-④c-d-e-f-g-h。较好的合并方式是：ab-cd-ef-gh -> abcd-efgh -> abcdefgh。好的合并方法是：ab-c-d-e-f-g-h ->abcd-e-f-g-h ->abcdef-gh -> abcdefgh）

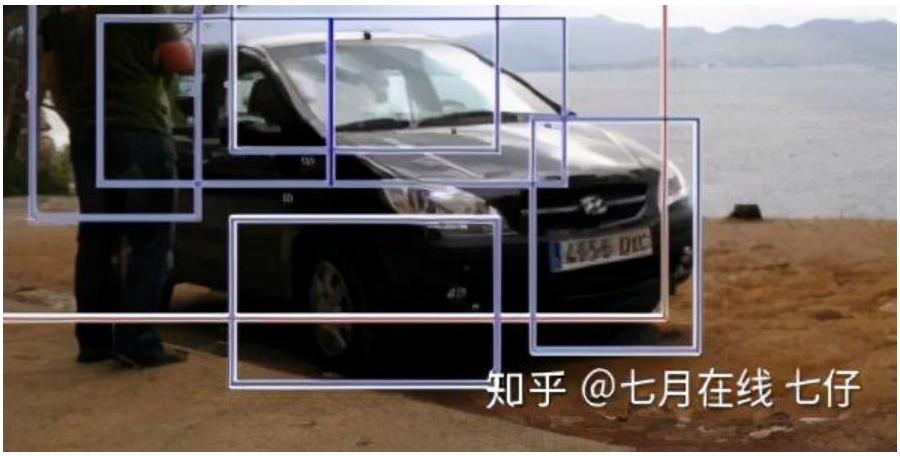
合并后，总面积在其BBOX中所占比例大的： 保证合并后形状规则。

上述四条规则只涉及区域的颜色直方图、梯度直方图、面积和位置。合并后的区域特征可以直接由子域特征计算而来，速度较快。

本题解析来源：lemon：RCNN- 将CNN引入目标检测的开山之作

54、什么是非极大值抑制（NMS）？

R-CNN会从一张图片中找出n个可能是物体的矩形框，然后为每个矩形框为做类别分类概率：



就像上面的图片一样，定位一个车辆，最后算法就找出了一堆的方框，我们需要判别哪些矩形框是没的。非极大值抑制的方法是：先假设有6个矩形框，根据分类器的类别分类概率做排序，假设从小到属于车辆的概率 分别为A、B、C、D、E、F。

(1)从最大概率矩形框F开始，分别判断A~E与F的重叠度IOU是否大于某个设定的阈值;

(2)假设B、D与F的重叠度超过阈值，那么就扔掉B、D；并标记第一个矩形框F，是我们保留下来的。

(3)从剩下的矩形框A、C、E中，选择概率最大的E，然后判断E与A、C的重叠度，重叠度大于一定的值，那么就扔掉；并标记E是我们保留下来的第二个矩形框。

就这样一直重复，找到所有被保留下来的矩形框。

55、什么是深度学习中的anchor?

解析一

提到RPN网络，就不能不说anchors。所谓anchors，实际上就是一组由rpn/generate_anchors.py成的矩形。直接运行Faster RCNN的作者在其论文中给的demo中的generate_anchors.py可以得到

下输出：

[[-84. -40. 99. 55.]

[-176. -88. 191. 103.]

[-360. -184. 375. 199.]

[-56. -56. 71. 71.]

[-120. -120. 135. 135.]

[-248. -248. 263. 263.]

[-36. -80. 51. 95.]

[-80. -168. 95. 183.]

[-168. -344. 183. 359.]]

56、CNN的特点以及优势

CNN使用范围是具有局部空间相关性的数据，比如图像，自然语言，语音

局部连接：可以提取局部特征。

权值共享：减少参数数量，因此降低训练难度（空间、时间消耗都少了）。可以完全共享，也可以局共享（比如对人脸，眼睛鼻子嘴由于位置和样式相对固定，可以用和脸部不一样的卷积核）

降维：通过池化或卷积stride实现。

多层次结构：将低层次的局部特征组合成为较高层次的特征。不同层级的特征可以对应不同任务。

57、深度学习中有什么加快收敛/降低训练难度的方法?

瓶颈结构

残差

学习率、步长、动量

优化方法

预训练

58、请写出链式法则并证明

链式法则或链锁定则（英语：chain rule），是求复合函数导数的一个法则。设f和g为两个关于x的可导函数，则复合函数$(f\circ g)(x)$的导数$(f\circ g)'(x)$为$(f\circ g)'(x)=f'(g(x))g'(x)$

以下是一个简单的例子

求函数$f(x)=(x^2+1)^3$的导数。设$g(x)=x^2+1$，$h(g)=g^3 \Rightarrow h(g(x))=g(x)^3$.

$f(x)=h(g(x))$ {|- $|f'(x)$ $|=h'(g(x))g'(x)$ $|=3(g(x))^2(2x)$ $|=3(x^2+1)^2(2x)$ |- ||$=6x(x^2+1)^2$. |}

求函数 arctan sin x的导数。

$$\frac{d}{dx}\arctan x = \frac{1}{1+x^2}$$

$$\frac{d}{dx}\arctan f(x) = \frac{f'(x)}{1+f^2(x)}$$

$$\frac{d}{dx}\arctan \sin x = \frac{\cos x}{1+\sin^2 x}$$

以下的简单的一个证明

设f和g为函数，x为常数，使得f在$g(x)$可导，且g在x可导。根据可导的定义，

$g(x+\delta)-g(x)=\delta g'(x)+\epsilon(\delta)\delta$ ，其中当$\delta\to 0$时，$\epsilon(\delta)\to 0$。

同理，

$f(g(x)+\alpha)-f(g(x))=\alpha f'(g(x))+\eta(\alpha)\alpha$ ，其中当$\alpha\to 0.$时，$\eta(\alpha)\to 0$。

现在

$$\begin{aligned} f(g(x+\delta))-f(g(x)) &= f(g(x)+\delta g'(x)+\epsilon(\delta)\delta)-f(g(x)) \\ &= \alpha_\delta f'(g(x))+\eta(\alpha_\delta)\alpha_\delta \end{aligned}$$

其中$\alpha_\delta=\delta g'(x)+\epsilon(\delta)\delta$.注意到当$\delta\to 0$时，$\frac{\alpha_\delta}{\delta}\to g'(x)$及$\alpha_\delta\to 0$，因此$\eta(\alpha_\delta)\to 0$。因此

$$\frac{f(g(x+\delta))-f(g(x))}{\delta}\to g'(x)f'(g(x)).$$

本题解析来源于这条Wikipedia：https://zh.wikipedia.org/wiki/é□¾å¼□æ³□□□□ 知乎 @七月在线 七仔

59、请写出Batch Normalization的计算方法及其应用

机器学习流程简介：

1）一次性设置（One time setup）

- 激活函数（Activation functions）
- 数据预处理（Data Preprocessing）
- 权重初始化（Weight Initialization）
- 正则化（Regularization：避免过拟合的一种技术）
- 梯度检查（Gradient checking）

2）动态训练（Training dynamics）

- 跟踪学习过程 （Babysitting the learning process）
- 参数更新 （Parameter updates）

● 超级参数优化（Hyperparameter optimization）

● 批量归一化（Batch Normalization简称BN，其中，Normalization是数据标准化或归一化、规范，Batch可以理解为批量，加起来就是批量标准化。解决在训练过程中中间层数据分布发生改变的问，以防止梯度消失或爆炸、加快训练速度）

3）评估（Evaluation）

● 模型组合（Model ensembles）

(训练多个独立的模型，测试时，取这些模型结果的平均值)

60、神经网络中会用到批量梯度下降（BGD）吗？为什用随机梯度下降（SGD）？

1）一般不用BGD

2）a. BGD每次需要用到全量数据，计算量太大

b. 引入随机因素，即便陷入局部极小，梯度也可能不为0，这样就有机会跳出局部极小继续搜索（可作为跳出局部极小的一种方式，但也可能跳出全局最小。还有解决局部极小的方式：多组参数初始化使用模拟退火技术）

61、当神经网络的调参效果不好时，从哪些角度思考？（要首先归结于overfiting）

1）是否找到合适的损失函数？（不同问题适合不同的损失函数）（理解不同损失函数的适用场景）

2）batch size是否合适？batch size太大 -> loss很快平稳，batch size太小 -> loss会震荡（理解min-batch）

3）是否选择了合适的激活函数？（各个激活函数的来源和差异）

4）学习率，学习率小收敛慢，学习率大loss震荡（怎么选取合适的学习率）

5）是否选择了合适的优化算法？（比如adam）（理解不同优化算法的适用场景）

6）是否过拟合？(深度学习拟合能力强，容易过拟合)（理解过拟合的各个解决方案）

a. Early Stopping

b. Regularization（正则化）

c. Weight Decay（收缩权重）

d. Dropout（随机失活）

e. 调整网络结构

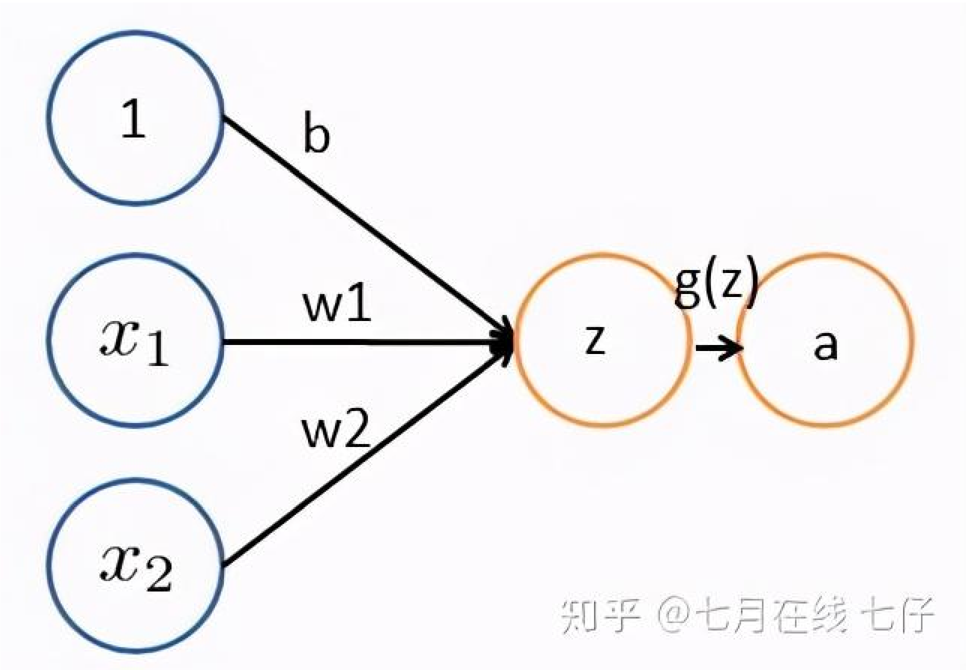
62、请阐述下卷积神经网络CNN的基本原理(全网最通俗)

1 神经元

神经网络由大量的神经元相互连接而成。每个神经元接受线性组合的输入后，最开始只是简单的线性权，后来给每个神经元加上了非线性的激活函数，从而进行非线性变换后输出。每两个神经元之间的接代表加权值，称之为权重（weight）。不同的权重和激活函数，则会导致神经网络不同的输出。

举个手写识别的例子，给定一个未知数字，让神经网络识别是什么数字。此时的神经网络的输入由一被输入图像的像素所激活的输入神经元所定义。在通过非线性激活函数进行非线性变换后，神经元被活然后被传递到其他神经元。重复这一过程，直到最后一个输出神经元被激活。从而识别当前数字是么字。

神经网络的每个神经元如下



基本wx + b的形式，其中

*x_1、x_2表示输入向量

*w_1、w_2为权重，几个输入则意味着有几个权重，即每个输入都被赋予一个权重

*b为偏置bias

*g(z) 为激活函数

*a 为输出

如果只是上面这样一说，估计以前没接触过的十有八九又必定迷糊了。事实上，上述简单模型可以追溯到20世纪50/60年代的感知器，可以把感知器理解为一个根据不同因素、以及各个因素的重要性程度而做决策的模型。

举个例子，这周末北京有一草莓音乐节，那去不去呢？决定你是否去有二个因素，这二个因素可以对应二个输入，分别用x1、x2表示。此外，这二个因素对做决策的影响程度不一样，各自的影响程度用权重w1、w2表示。一般来说，音乐节的演唱嘉宾会非常影响你去不去，唱得好的前提下 即便没人陪同都可忍受，但如果唱得不好还不如你上台唱呢。所以，我们可以如下表示：

*x_1：是否有喜欢的演唱嘉宾。x_1= 1 你喜欢这些嘉宾，x_1 = 0 你不喜欢这些嘉宾。嘉宾因素的权重w_1 = 7

*x_2：是否有人陪你同去。x_2 = 1 有人陪你同去，x_2 = 0 没人陪你同去。是否有人陪同的权重w_2 = 3。

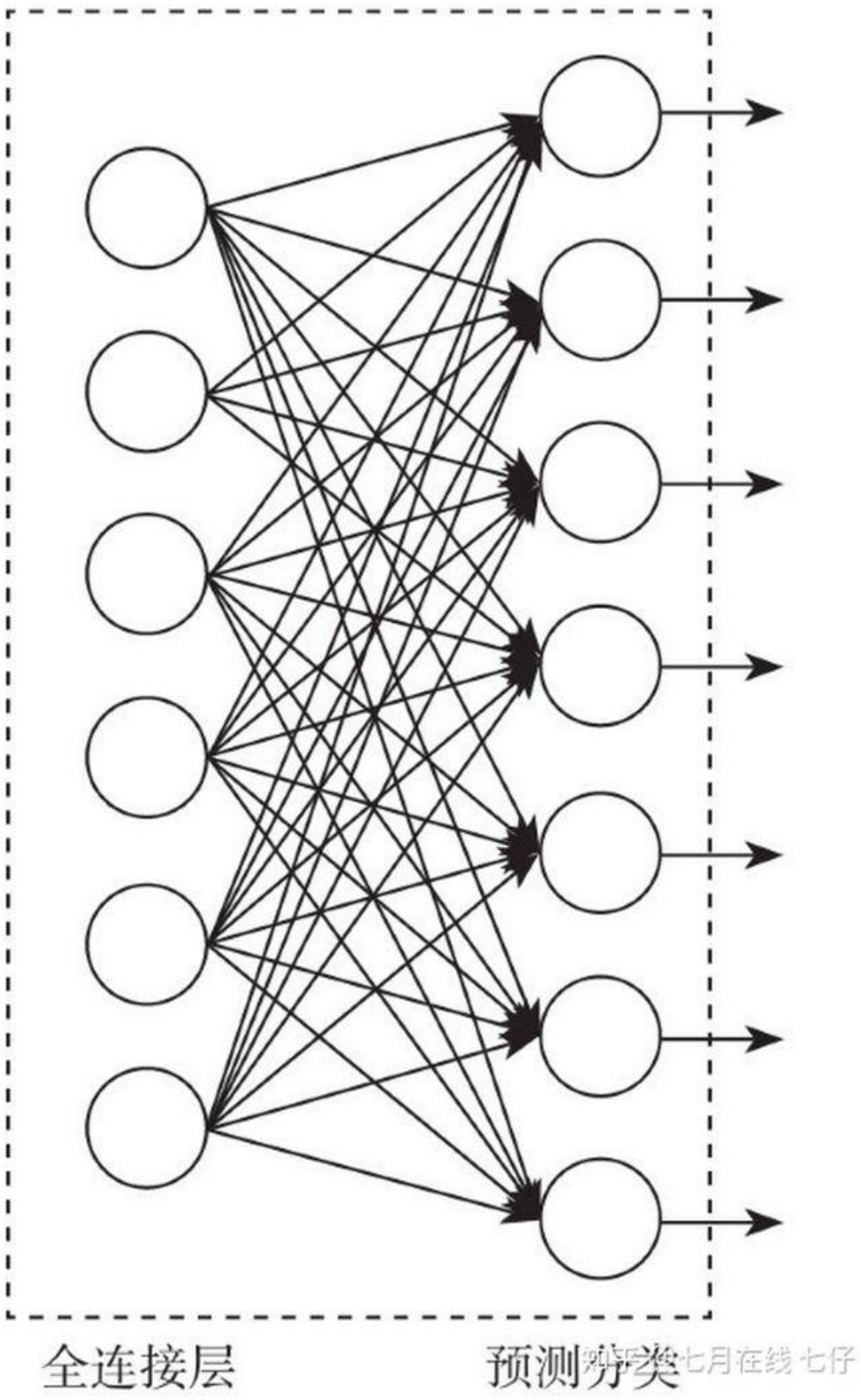
这样，咱们的决策模型便建立起来了：$g(z) = g(w_1 * x_1 + w_2 * x_2 + b)$，g表示激活函数，这里的b可以理解成 为更好达到目标而做调整的偏置项。

一开始为了简单，人们把激活函数定义成一个线性函数，即对于结果做一个线性变化，比如一个简单的线性激活函数是g(z) = z，输出都是输入的线性变换。后来实际应用中发现，线性激活函数太过局限，于是引入了非线性激活函数。

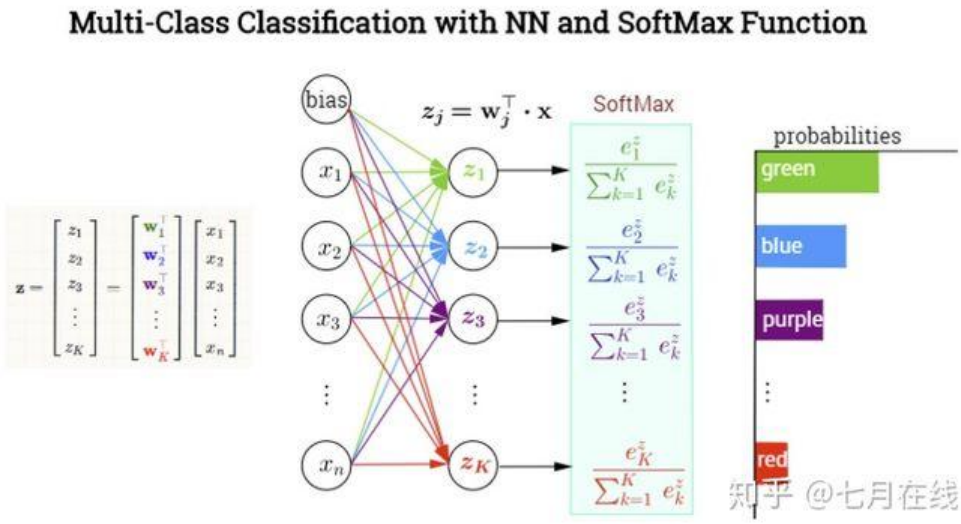
63、神经网络输出层为什么通常使用softmax?

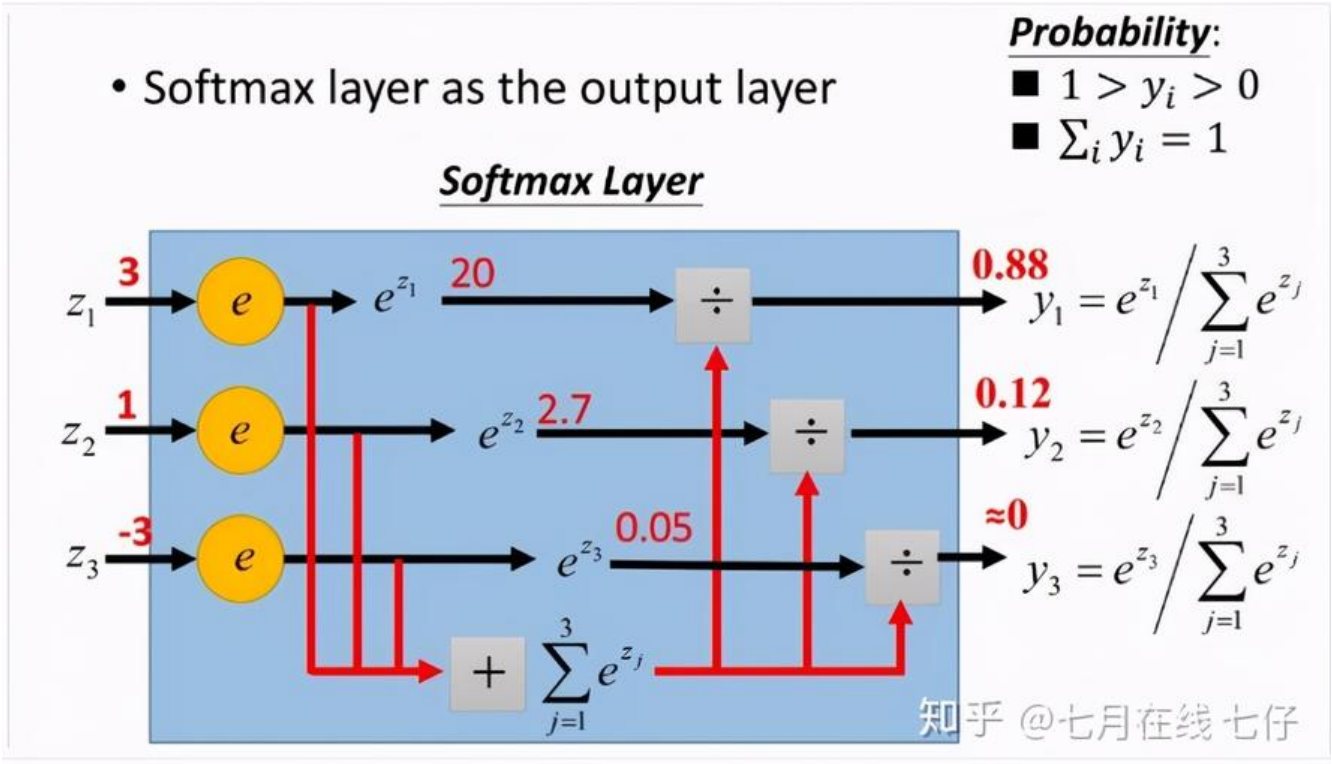
1 什么是softmax

常用于神经网络输出层的激励函数SOFTMAX长什么样子呢？如下图所示



从图的样子上看，和普通的全连接方式并无差异，但激励函数的形式却大不一样。





首先后面一层作为预测分类的输出节点，每一个节点就代表一个分类，如图所示，那么这7个节点就代表着7个分类的模型，任何一个节点的激励函数都是：

$$\sigma_i(z) = \frac{e^{z_i}}{\sum_{j=1}^{m} e^{z_i}}$$

其中i 就是节点的下标次序，而 $z_i = w_i + b_i$，也就说这是一个线性分类器的输出作为自然常数e的指数。最有趣的是最后一层有这样的特性：

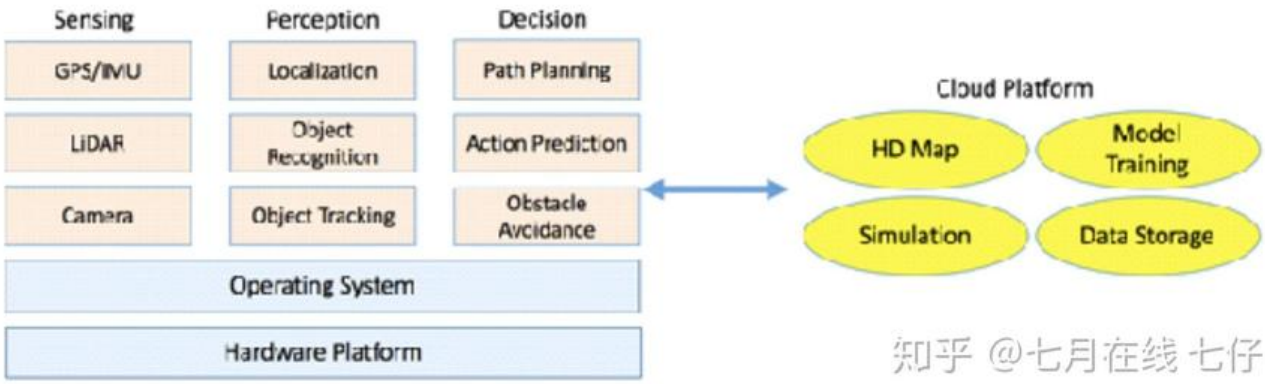
$$\sum_{i=1}^{j} \sigma_i(z) = 1$$

也就是说最后一层的每个节点的输出值的加和是1。这种激励函数从物理意义上可以解释为一个样本属于两者进行分类的时候在每个节点上输出的值都是小于等于1的，是它从属于这个分类的概率。

64、了解无人驾驶的核心技术么？

总的说来，无人驾驶系统主要由三部分组成：算法端、车端和云端。其中算法端包括传感器、感知和策等智能关键步骤的算法；车端包括机器人操作系统、各种计算硬件和车辆底盘硬件等；云端包括数挖掘、仿真模拟、高精地图以及深度学习训练等等。

相当于无人驾驶要解决4 个关键问题：我在哪？我周围有什么？接下来会发生什么？我应该怎么做？



65、如何形象的理解LSTM的三个门

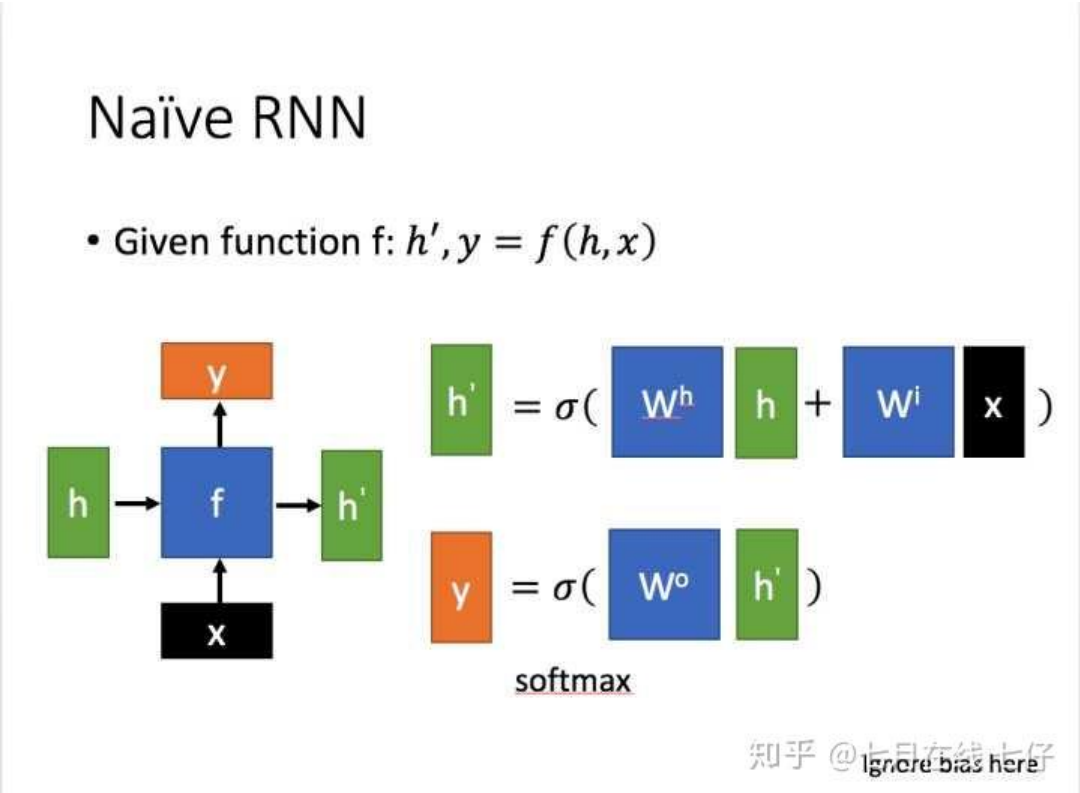
0. 从RNN说起

循环神经网络（Recurrent Neural Network，RNN）是一种用于处理序列数据的神经网络。相比一的神经网络来说，他能够处理序列变化的数据。比如某个单词的意思会因为上文提到的内容不同而有同的含义，RNN就能够很好地解决这类问题。

1. 普通RNN

先简单介绍一下一般的RNN。

其主要形式如下图所示（图片均来自台大李宏毅教授的PPT）：



这里：

x 为当前状态下数据的输入， h 表示接收到的上一个节点的输入。

y 为当前节点状态下的输出，而 h' 为传递到下一个节点的输出。

通过上图的公式可以看到，输出 h' 与 x 和 h 的值都相关。

而 y 则常常使用 h' 投入到一个线性层（主要是进行维度映射）然后使用softmax进行分类得到需要数据。

对这里的y如何通过 h' 计算得到往往看具体模型的使用方式。

通过序列形式的输入，我们能够得到如下形式的RNN。