AI面试百题训练营

序号	难度	题目	注意事项
1	简单	为什么要对特征做归一化	理解清楚特征归一化所适用的模型场景
2	中等	什么是组合特征？如何处理高维组合特征？	这里的特征组合主要指的是类别特征(Categorical Feature)之间的组合
3	中等	请比较欧式距离与曼哈顿距离？	比较曼哈顿距离和欧式距离的数值特点，并结合一两个具体例子做分析
4	中等	为什么一些场景中使用余弦相似度而不是欧式距离	比较余弦相似度和欧式距离的数值特点，并结合一两个具体例子做分析
5	中等	One-hot的作用是什么？为什么不直接使用数字作为表示	理解清楚并比较One-hot编码和数字编码的特点

为什么要对特征做归一化?

特征归一化是将所有特征都统一到一个大致相同的数值区间内，通常为[0, 1]。常用的特征归一化方法有：

1. Min-Max Scaling

对原始数据进行线性变换，使结果映射到[0, 1]的范围，实现对数据的等比例缩放。

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

其中 X_{min}，X_{max} 分别为数据的最小值和最大值

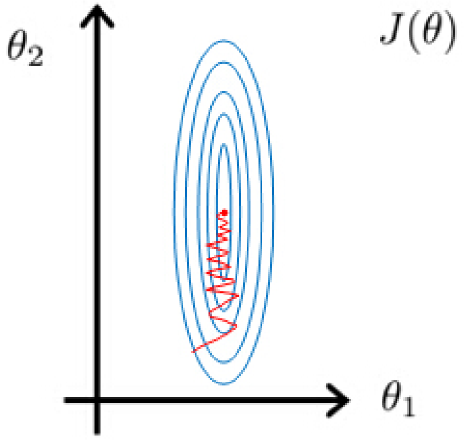
2. Z-Score Normalization

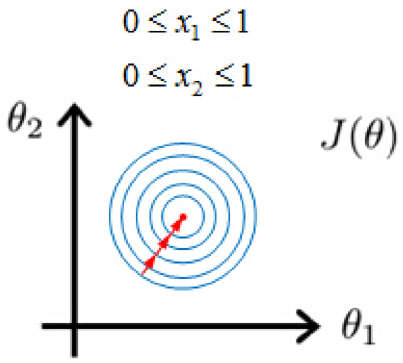
将原始数据映射到均值为 0，标准差为 1 的分布上。

$$X_{norm} = \frac{X - \mu}{\sigma}$$

其中μ为原始特征的均值，而σ为原始特征的标准差。

在采用基于梯度更新的学习方法（包括线性回归，逻辑回归，支持向量机，神经网络等）对模型求解的过程中，未归一化的数值特征在学习时，梯度下降较为抖动，模型难以收敛，通常需要较长的时间模型才能收敛；而归一化之后的数值特征则可以使得梯度下降较为稳定，进而减少梯度下降的次数，也更容易收敛。下图中，左边为特征未归一化时，模型的收敛过程；而右边是经过特征归一化之后模型的收敛过程。





2、什么是组合特征？ 如何处理高维组合特征？

狭义的组合特征即将类别特征（Categorical feature）两个或者多个特征组合（数学里面的组合概念）起来，构成高阶组合特征。

比如：假设Mac笔记本电脑的CPU型号和SSD大小对是否购买行为的影响用下面的表格表示

是否购买	CPU 型号		SSD 大小	
	Intel i5	Intel i7	256 GB	512GB
1	1	0	1	0
0	0	1	1	0
0	1	0	0	1
1	0	1	0	1

那么CPU型号和SSD大小的组合特征对是否购买行为的影响为

是否购买	CPU 型号和 SSD 大小 组合特征			
	CPU = Intel i5 SSD = 256 GB	CPU = Intel i7 SSD = 256 GB	CPU = Intel i5 SSD = 512 GB	CPU = Intel i7 SSD = 512 GB
1	1	0	0	0
0	0	1	0	0
0	0	0	1	0
1	0	0	0	1

组合特征的不同取值的个数（number of unique values）为单个特征的不同取值的个数的乘积。假设数据的特征向量为 $X = (x_1, x_2, \ldots, x_k)$ 则，$|< x_i, x_j >| = |x_i| * |x_j|$ 其中$<x_i, x_j>$ 为特征x_i和特征x_j的组合特征，$|x_i|$ 表示特征x_i 不同取值的个数，$|x_j|$表示特征x_j不同取值的个数。

假设采用以线性模型为基础的模型来拟合特征时，比如以逻辑回归为例：

$$Y = sigmoid(\sum_i \sum_j w_{ij} < x_i, x_j >)$$

需要学习的参数w_{ij}的长度为$|< x_i, x_j >|$； 如果$|x_i| = m, |x_j| = n$， 则参数规模为 m*n。当 m 和 n 非常大时，经过特征组合后的模型就会变得非常复杂。一个可行的方法就是，做特征的 embedding，即将x_i，x_j分别用长度为 k 的低维向量表示（k <<m, k <<n）； 那么学习参数的规模则变为 m * k + n * k + k * k.

请比较欧式距离与曼哈顿距离？

欧式距离，即欧几里得距离，表示两个空间点之间的直线距离。

$$d = (\sum_{k=1}^{n} |a_k - b_k|^2)^{\frac{1}{2}}$$

曼哈顿距离，所有维度距离绝对值之和。

$$d = \sum_{k=1}^{n} |a_k - b_k|$$

在基于地图，导航等应用中，欧式距离表现得理想化和现实上的距离相差较大；而曼哈顿距离就较为合适；另外欧式距离根据各个维度上的距离自动地给每个维度计算了一个"贡献权重"，这个权重会因为各个维度上距离的变化而动态的发生变化；而曼哈顿距离的每个维度对最终的距离都有同样的贡献权重。

为什么一些场景中使用余弦相似度而不是欧式距离？

假设有 A 和 B 两个向量，其余弦相似度定义为 $\cos(A, B) = \frac{A*B}{\|A\|_2 \|B\|_2}$, 即两个向量夹角的余弦。

1. 它关注的是向量之间的角度关系，相对差异，而不关心它们的绝对大小；
2. 其取值范围在[-1, 1]之间；
3. 两个向量相同时为 1，正交时为 0，相反时为-1. 即在取值范围内，余弦距离值越大，两个向量越接近；

余弦距离为向量之间的相似度量提供了一个稳定的指标，无论向量维度多与少；特征的取值范围大与小。余弦距离的取值范围始终都能保持在[-1, 1]。余弦相似度广泛应用在文本，图像和视频领域。相比之下欧氏距离则受到维度多少，取值范围大小以及可解释性的限制。当特征的取值以及特征向量经过模长归一化之后，余弦距离和欧氏距离又存在以下的单调关系。

$$\|A - B\|_2 = \sqrt{2(1 - \cos(A, B)}$$

One-hot的作用是什么？为什么不直接使用数字作为表示？

One-hot 主要用来编码类别特征，即采用哑变量(dummy variables) 对类别进行编码。它的作用是避免因将类别用数字作为表示而给函数带来抖动。直接使用数字会给将人工误差而导致的假设引入到类别特征中，比如类别之间的大小关系，以及差异关系等等。