

算术编码

编码方法

假设有一段数据需要编码：

①统计数据内所有字符出现的概率。

②将区间 $[0, 1)$ 连续分割成多个子区间，每个字符占一个子区间。

如： $P(A)=0.2$ ， $Z(A)=[0, 0.2)$ ， $P(B)=0.3$ ， $Z(B)=[0.2, 0.5)$ 以此类推。

③依次读入每个字符，看字符在“目前区间”哪个子区间里，然后将这个子区间变成“目前区间”，继续重复这一步。

如：第一个字符为A，其所在子区间为 $[0, 0.2)$ ，则读入第二个字符时，将目前区间变为 $[0, 0.2)$ 。如果第二个字符依然为A，那其所在的子区间应当是 $[0, 0.04)$ 。

其中 $0.04=0.2*0.2$ 。因为目前区间由 $[0, 1)$ 变为 $[0, 0.2)$ 了。

难以理解的话，可以看下方的例子。

④重复第三步，直到读取完最后一个字符，然后在此时处理完的区间里，任选一个小数作为编码。

编码例子

数据为【ABRER】。

数据内有四个字母，分别为A/B/E/R，其出现概率与划分的区间如下。

字符	出现概率	划分区间
A	0.2	$[0, 0.2)$
B	0.2	$[0.2, 0.4)$
E	0.2	$[0.4, 0.6)$
R	0.4	$[0.6, 1)$

更新区间公式：

左边界=母区间左边界+区间长度*字符所在区间左边界。

右边界=母区间左边界+区间长度*字符所在区间右边界。

第一个字符：A。当前区间： $[0, 1)$

更新区间为 $[0 + 1 * 0, 0 + 1 * 0.2) = [0, 0.2)$ 。

第二个字符：B。当前区间：[0, 0.2)

更新区间为 $[0 + 0.2 * 0.2, 0 + 0.2 * 0.4) = [0.04, 0.08)$ 。

第三个字符：R。当前区间：[0.04, 0.08)

更新区间为 $[0.04 + 0.04 * 0.6, 0.04 + 0.04 * 1) = [0.064, 0.08)$ 。

第四个字符：E。当前区间：[0.064, 0.08)

更新区间为 $[0.064 + 0.016 * 0.4, 0.064 + 0.016 * 0.6) = [0.0704, 0.0736)$ 。

第五个字符：R。当前区间：[0.0704, 0.0736)

更新区间为 $[0.0704 + 0.0032 * 0.6, 0.0704 + 0.0032 * 1) = [0.07232, 0.0736)$ 。

在最后区间[0.07232, 0.0736) 中任取一个小数，即为编码。

通过编码反推数据

譬如编码例子中，假设我们最后得到的编码为0.07233。

0.07233在A的区间[0, 0.2) 中，故第一个字符为A，更新母区间为[0, 0.2)。

0.07233在[0.04, 0.08]，对应字符为B，故第二个字符为B.....

很快就能发现，这实际上就是编码过程的反推，以此类推，就能推出所有的数据。

比如0.07233在[0,0.2)、[0.04,0.08)、[0.064,0.08)、[0.0704,0.0736)、[0.07232,0.0736)

所以数据为ABRER。