



链滴

2021 年 4 月份，美团 NLP 算法岗面试题 总结

作者: [julyedu](#)

原文链接: <https://ld246.com/article/1622709495742>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

添加微信: julyedufu77 回复, " 11 ", 领取最新升级
《名企AI面试100题》电子书!!

面试题

问题 1: 怎么处理数据不平衡

问题 2: 给你单链表的头节点 head, 请你反转链表, 并返回反转后的链表。

问题 3: 连续子数组的最大乘积

问题 4: 最大子数组

问题 5: 一个硬币正面概率p 那么抛到第几次抛正面期望

题目解析

问题1: 怎么处理数据不平衡

常用于解决数据不平衡的方法:

欠采样: 从样本较多的类中再抽取, 仅保留这些样本点的一部分;

过采样: 复制少数类中的一些点, 以增加其基数;

生成合成数据: 从少数类创建新的合成点, 以增加其基数。

添加额外特征: 除了重采样外, 我们还可以在数据集中添加一个或多个其他特征, 使数据集更加丰富这样我们可能获得更好的准确率结果。

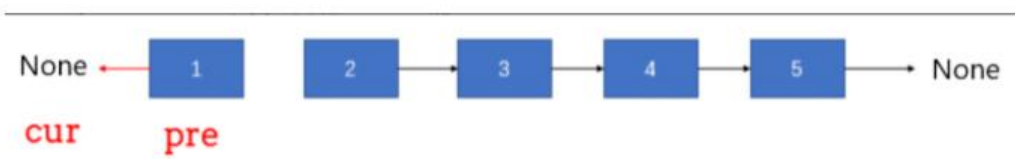
问题2: 给你单链表的头节点 head, 请你反转链表, 并返回反转后链表。

<https://leetcode-cn.com/problems/reverse-linked-list/>

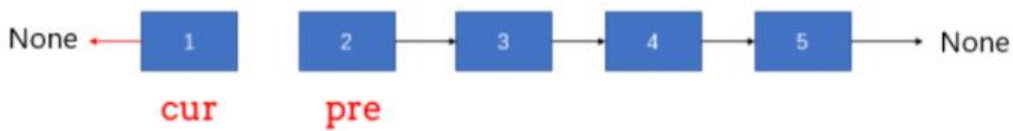
定义两个节点cur=None和pre=head



改变节点方向让pre的next指向cur, 实现一次局部反转



cur和pre向前移动一个位置



循环交换前进，直至pre为空，遍历结束，完成反转，此时cur节点为开始节head；

参考代码：

```

1 class Solution:
2     def reverseList(self, head: ListNode) -> ListNode:
3         cur, pre = None, head
4         while pre != None:
5             next_node = pre.next
6             pre.next = cur
7             cur, pre = pre, next_node
8         return cur
  
```

问题3：连续子数组的最大乘积

<https://leetcode-cn.com/problems/maximum-product-subarray/>

思路：

遍历数组时计算当前最大值、最小值，不断更新

当前最大值为 $ans_max = \max(ans_max * nums[i], nums[i])$

当前最小值为 $ans_min = \min(ans_min * nums[i], nums[i])$

由于存在负数，那么会导致最大的变最小的，最小的变最大的。

当前最大值为 $ans_max = \max(ans_min * nums[i], nums[i])$

当前最小值为 $ans_min = \min(ans_max * nums[i], nums[i])$

参考代码：

```

1 class Solution:
2     def maxProduct(self, nums: List[int]) -> int:
3         if len(nums) == 1: return nums[0]
4         ans_max = nums[0] # 当前最大值
5         ans_min = nums[0] #当前最小值
6         ans = nums[0] # 当前最大乘积
7         # nums[i]分两种情况: 大于0, 小于0
8         for i in range(1, len(nums)):
9             # 小于0时,
10            if nums[i] < 0:
11                tmp = ans_max
12                ans_max = max(ans_min*nums[i], nums[i])
13                ans_min = min(tmp*nums[i], nums[i])
14                ans = max(ans_max, ans)
15            # 大于等于0时
16            else:
17                ans_max = max(ans_max*nums[i], nums[i])
18                ans_min = min(ans_min*nums[i], nums[i])
19                ans = max(ans_max, ans)
20        return ans

```

问题4: 最大子数组

<https://leetcode-cn.com/problems/maximum-subarray/description/>

解题思路:

遍历数组, 遍历的时候记录两个值: 当前子数组的和 tmpSum, 最大值res

```

1 class Solution:
2     def maxSubArray(self, nums: List[int]) -> int :
3         tmp_sum = 0
4         res = nums[0]
5         for num in nums:
6             tmp_sum = max(tmp_sum + num, num)
7             res = max(res, tmp_sum)
8         return res

```

问题5: 一个硬币正面概率p 那么抛到第几次抛正面期望

硬币游戏, 如果在连续抛出三次正面之前不要停下来, 那么我们总计抛硬币的期望次数是多少

假设期望是x

假设第一抛是反面, 那么就浪费了一步, 平均一共需要x+1步(概率是1/2)

假设第一抛是正面, 在此基础上如果第二抛是反面, 又浪费了, 平均一共需要x+2步 (概率是1/4)

在此基础上如果第二抛是正面

假设第三抛反面，浪费，平均一共 $x+3$ 步（概率是 $1/8$ ）

假设第三抛正面，完成，只用了3步（概率是 $1/8$ ）

所以 x 的期望即 $x=(1/2)(x+1)+(1/4)(x+2)+(1/8)(x+3)+(1/8)*3$

解得 $x=14$

**添加微信：julyedufu77，回复：“11”，领取最新升级
《名企AI面试100题》电子书！！**