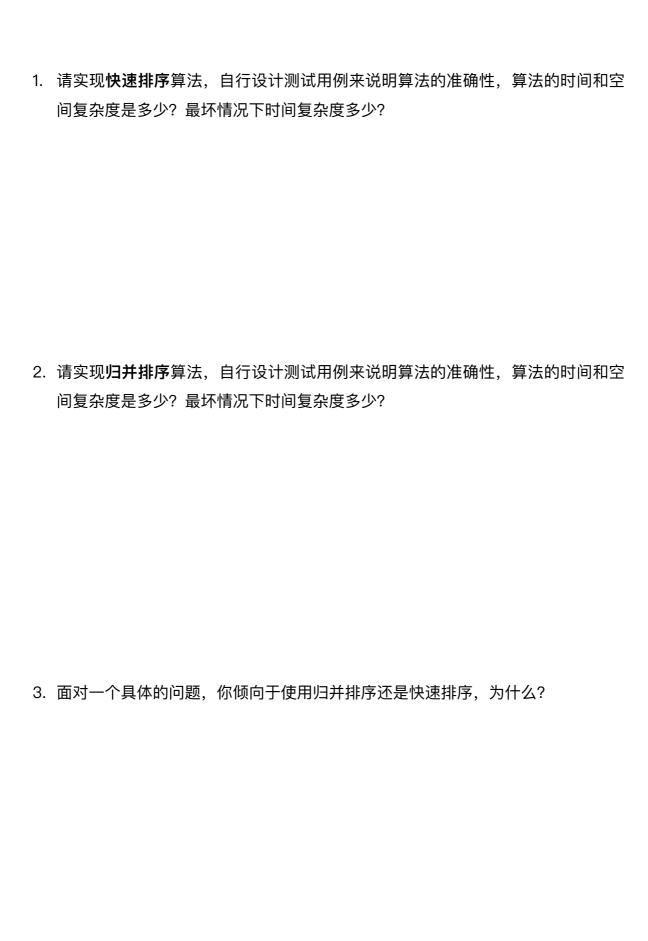
Part 1. 数据结构与算法相关



4. 对于输入大小 n, 以下每个数代表对应的时间复杂度,请按照**从小到大**的顺序排一下 $2^{\log n}$, $\sqrt{2}^{\log n}$, n^2 , n!, $(\log n)!$, n^3 , $\log(n!)$, n* 2^n , $n\log n$, e^n , $\log n$, 1, $2^{2^{n+1}}$

5. 利用主定理来回答以下几个算法的时间复杂度:

$$T(n) = 3 * T\left(\frac{n}{3}\right) + n$$

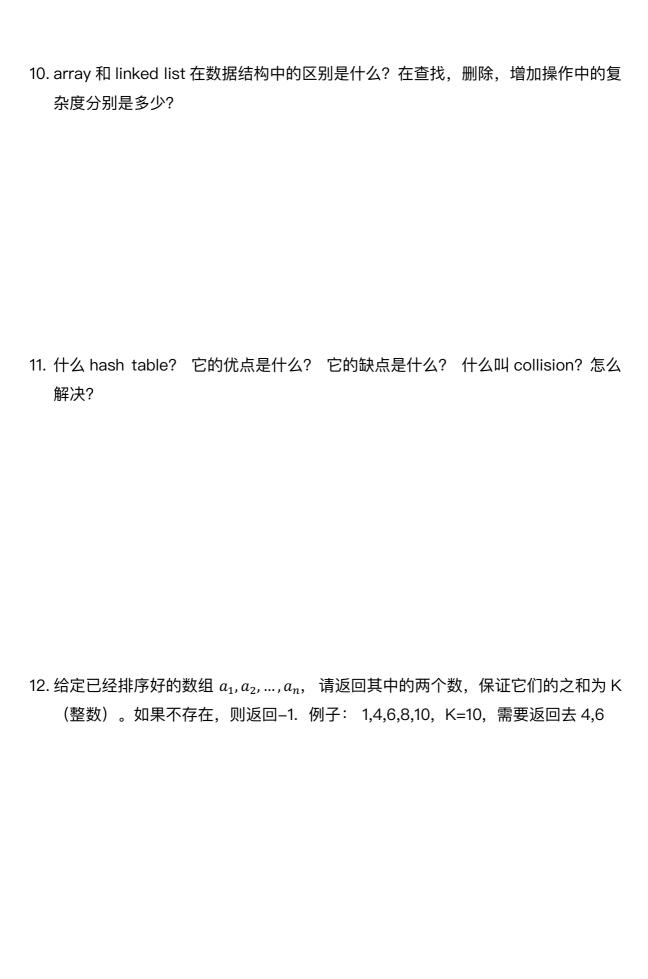
$$T(n) = 4 * T\left(\frac{n}{2}\right) + n^{2.5}$$

6. 计算递归函数的空间复杂度。 假如有一个序列是 1, 1, 1, 3, 5, 9, 17.... 利用递归来表示的话: T(n) = T(n-3) + T(n-2) + T(n-1), 如果用此递归来实现第 N 个数的求解过程,此算法的**空间复杂度**是多少?

7. 我们有 N 个没有排序的数列 $a_1, a_2, ..., a_n$ 现在需要从中选择最大的 K 个数 (K < N),请给出实现算法,确保算法的复杂度为 O(N log K). (hint: priority queue)

8. 对于给定的整数 n >0, 请求出它的平方根(sqrt(n)), 保留小数点后一位,不要使用任何 built-in function. (hint: binary search)

9. 目前有两个已经排序好的数组(从小到大) $a_1, a_2, ..., a_n$, $b_1, b_2, ..., b_m$, 请求出两个数组合并之后的 median。 尽可能想出最高效的方法



13. 我 们 平 时 都 接 触 过 short url, 比 如 对 于 https://greedy.com/154328794837567/685949281810, 我们可以存储成 https://greedy.com/afsik. 这种 URL 有很多好处(节省传输 band width, 隐私等等)。假设我们要设计一个类似的服务来接受 short url, 然后 redirect to long url, 而且我们需要处理上百亿个这种 url. 如果你是系统架构师会怎么设计? (hint: distributed hash)

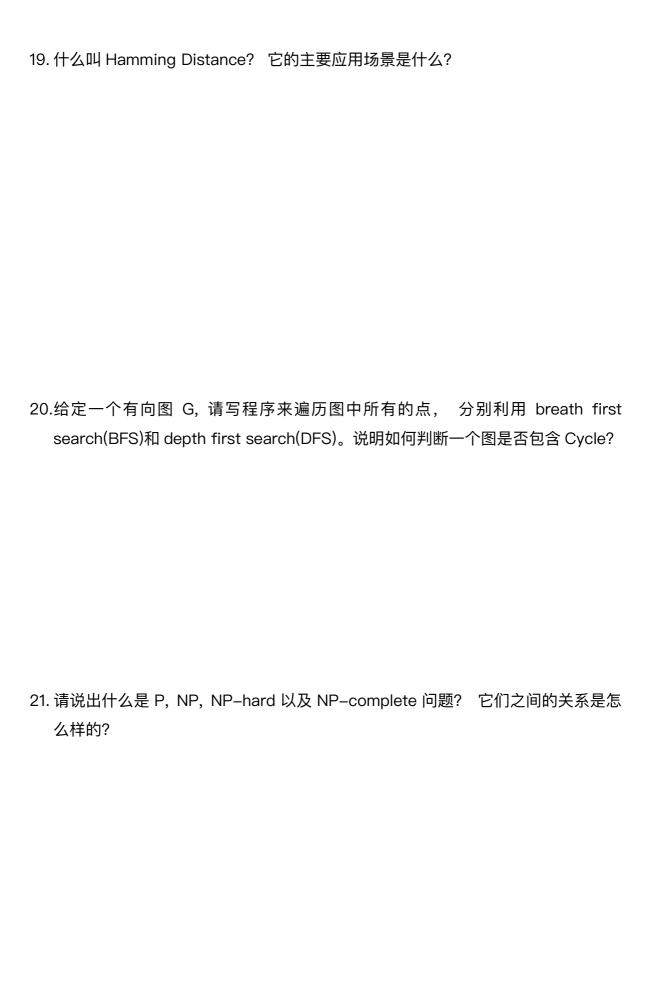
14. 树形的数据结构有很多种,包括 binary search tree, red-black tree, B-tree 等 等。是否了解在数据库系统中常用的数据结构是哪一种?为什么?

15. 给定两个字符串 s1 和 s2, 请计算它们的编辑距离。编辑距离中我们默认有 3 种不同的操作,分别是 add, delete, 以及 replace,每个操作的 cost 为 1.

16. 给定一个数组(有正,有负),找出合为最大的连续的子数组(也叫做maximum subsequence problem)。比如对于一个数组 1, 5, -4, 3, 6, 3, -3, -5, 1, 合为最大的子数组为(1, 5,-4, 3, 6, 3)

17. 给定一个字符串,请写出它所有可能的 permutation。 例子,给定 s="abca",它可能的所有字符串为: "aabc", "aacb", "abac", "abac", "acba", "acab", "baac", "baac", "baac", "baac", "caba", "caba",

18. 什么叫贪心算法,它与动态规划的主要区别是什么?



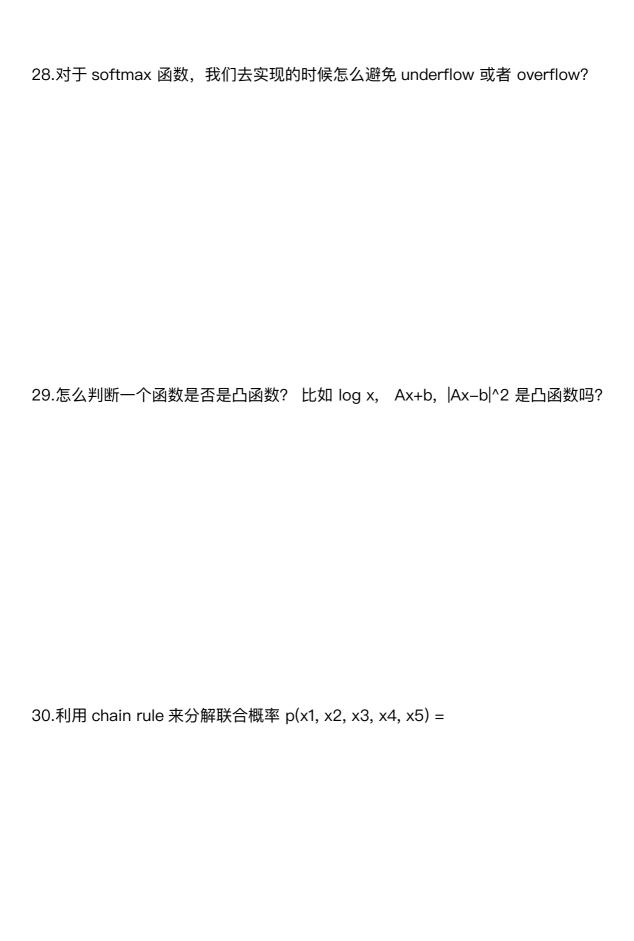
Part 2. 数学基础

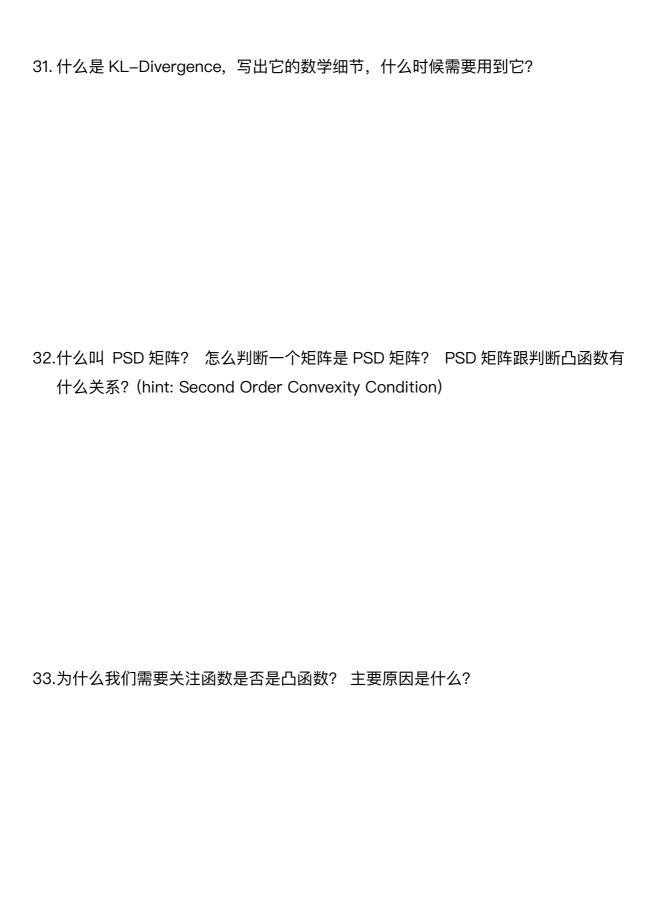
22.给定两个矩阵,怎么计算它们之间的相乘? 怎么计算一个矩阵 inverse? $A = [[1,3],[2,5]], B[[4,1],[2,4]], A*B = ? A^{-1} = ?$

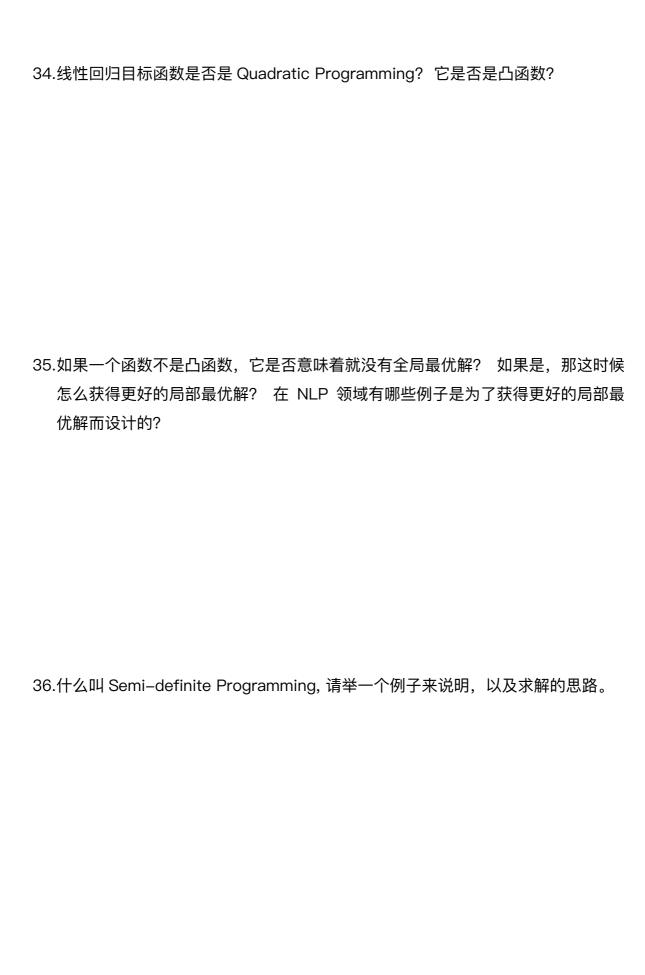
23.怎么计算一个向量的 norm? a = (3,1,5,1), |a| = ?

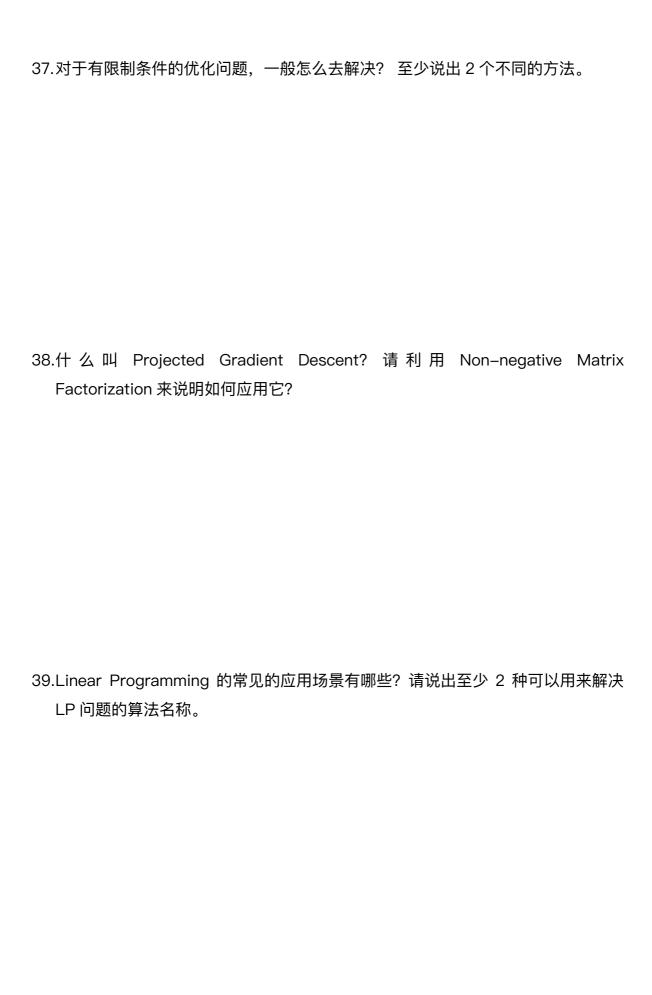
24.什么是 Frobenius norm? 给定 A = [[1,3],[2,5]], 请计算 $||A||_F^2$

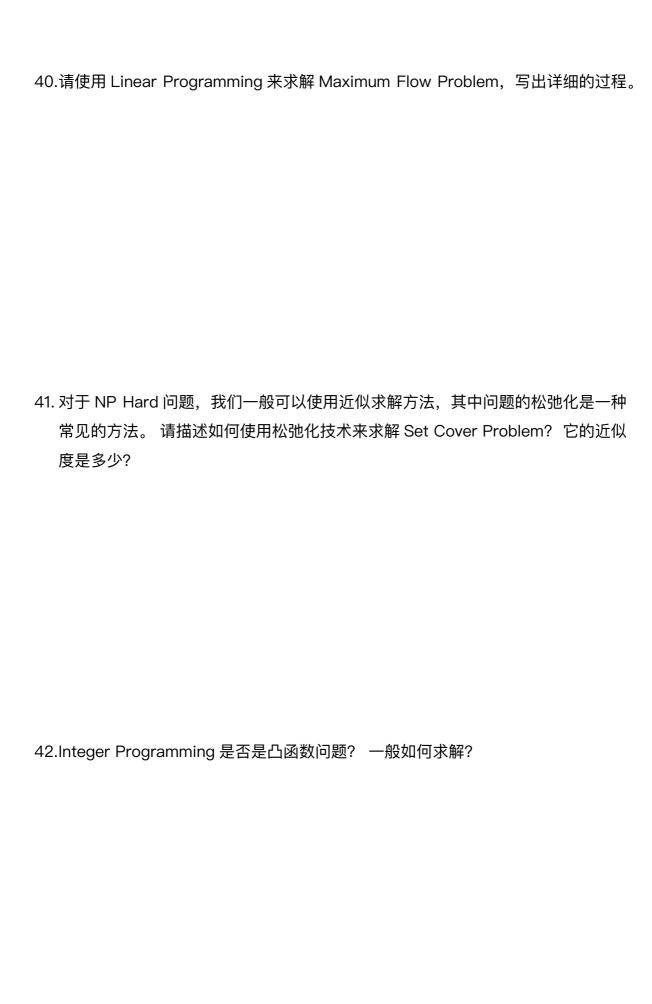




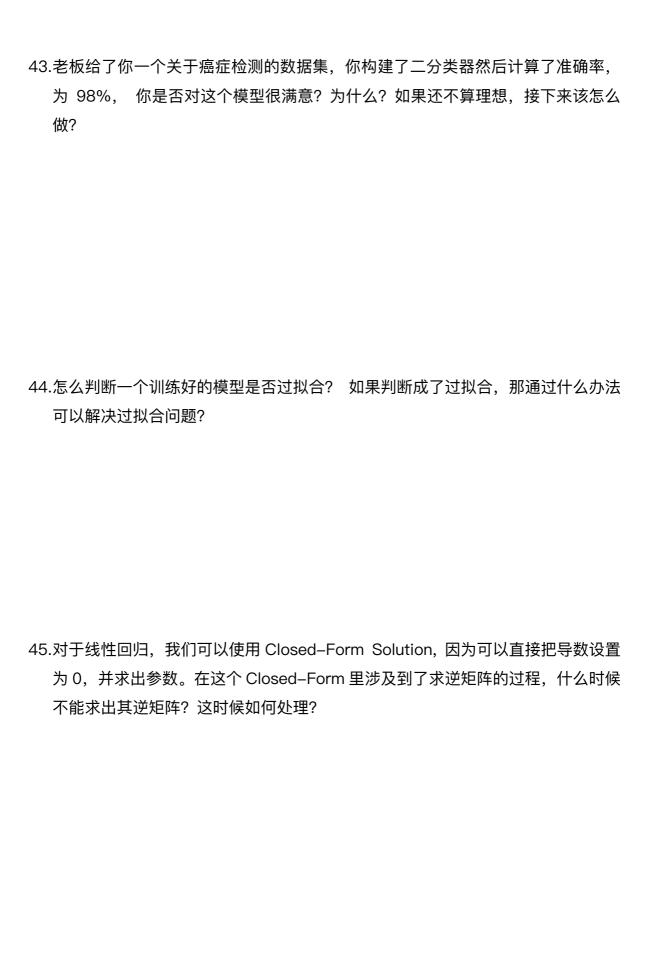


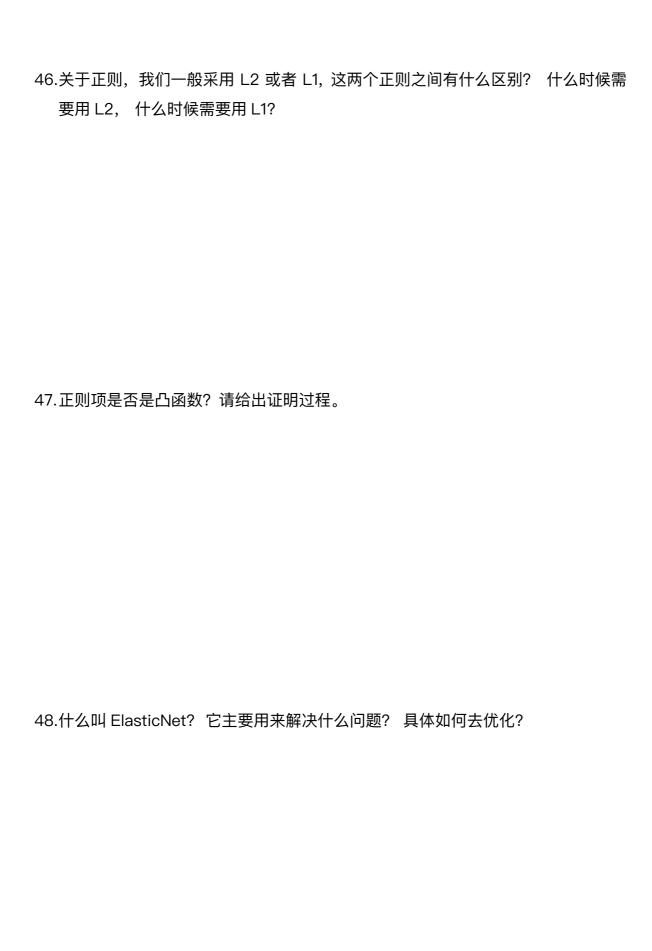


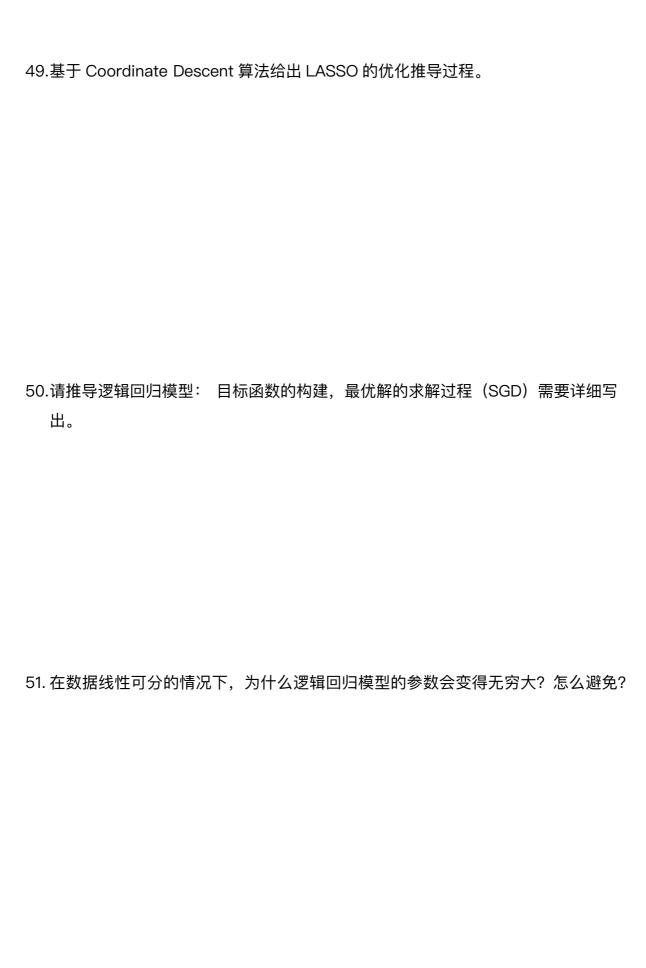


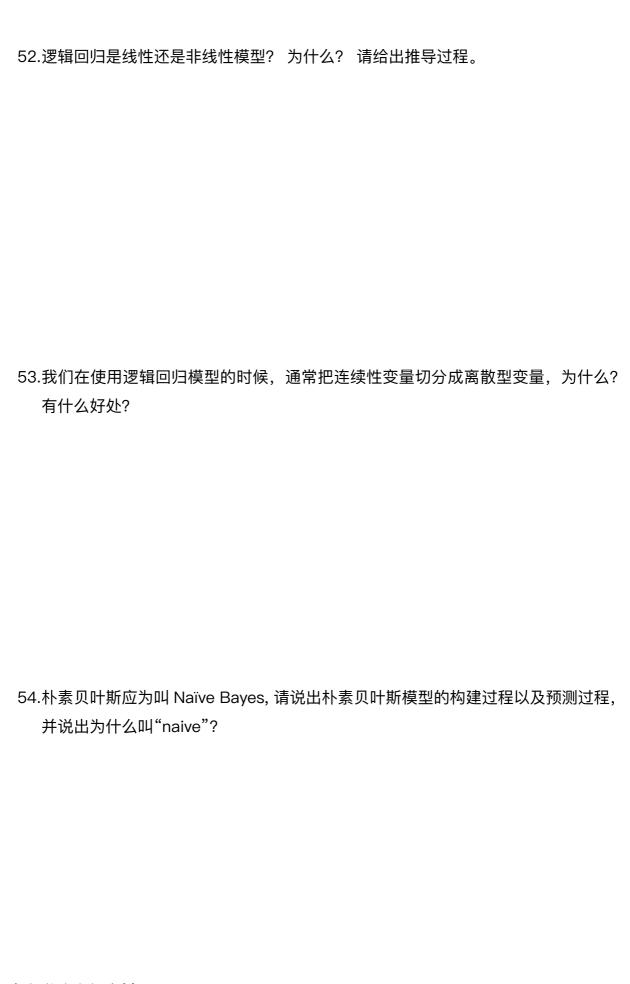


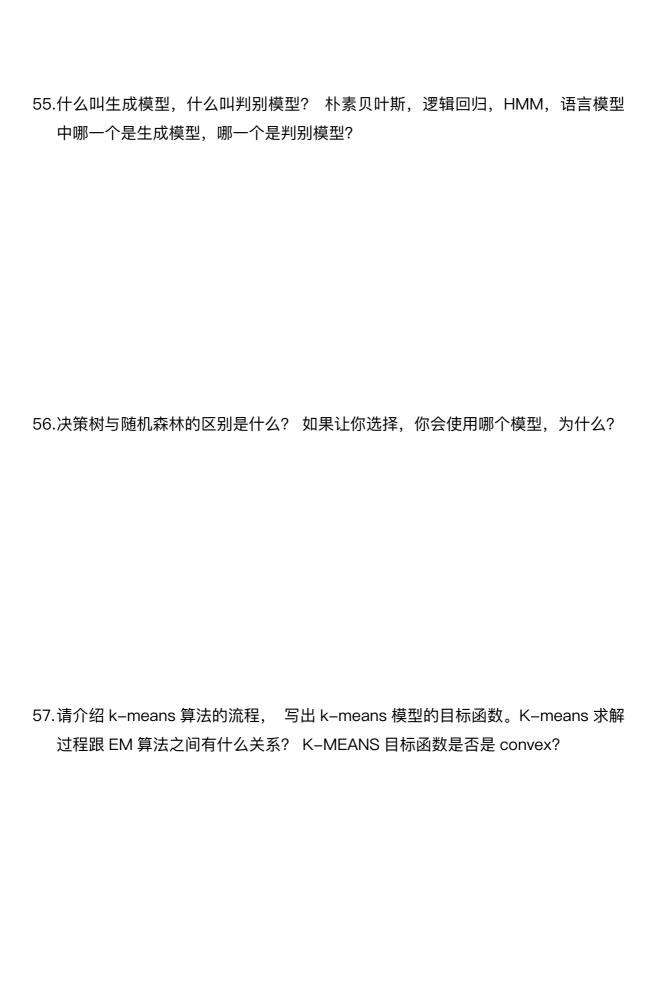
Part 3. 机器学习基础

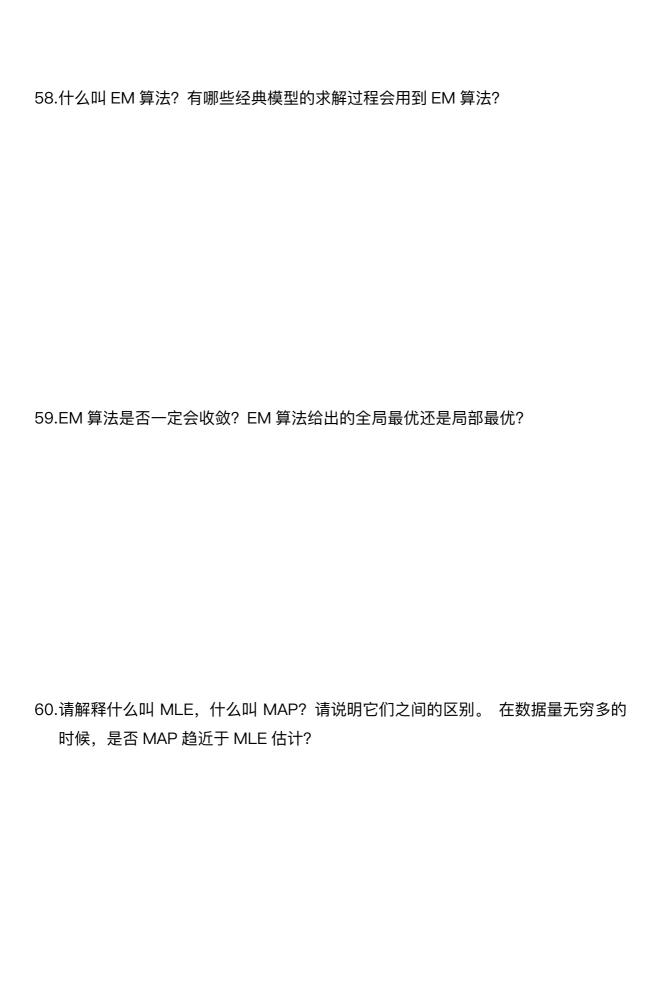


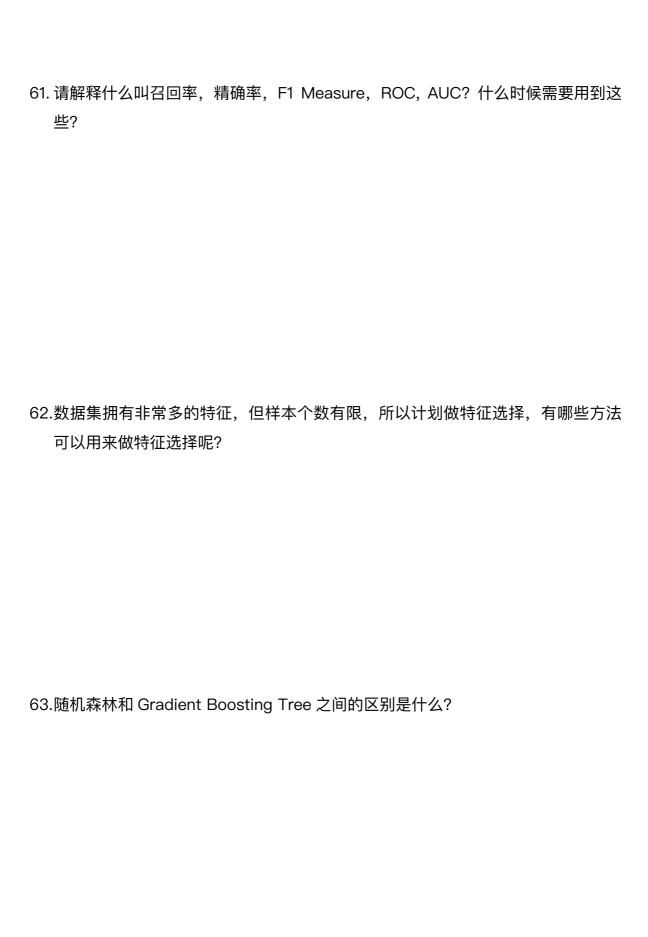


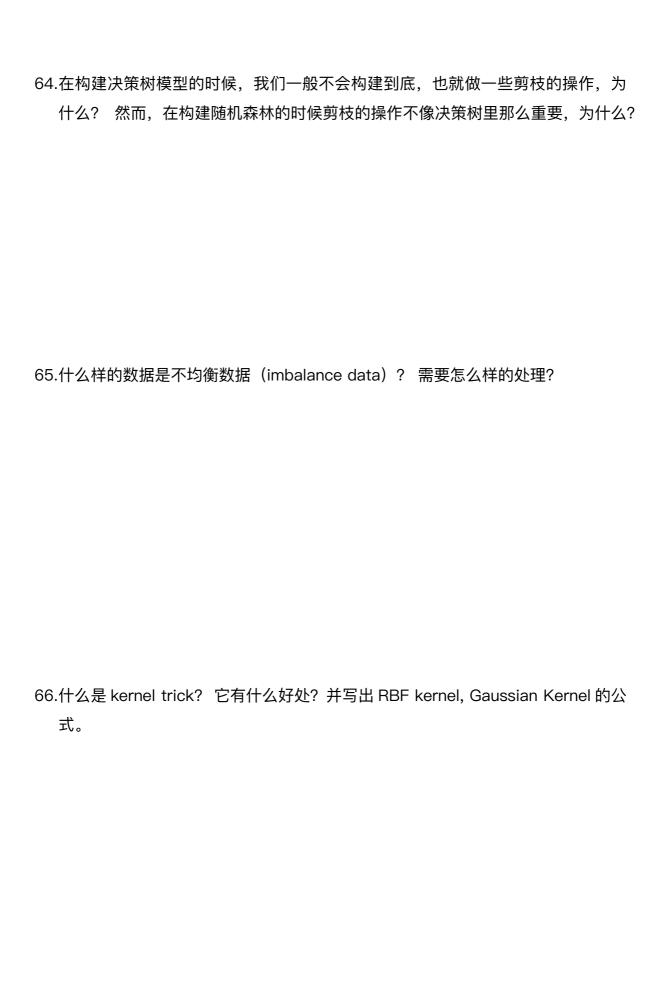


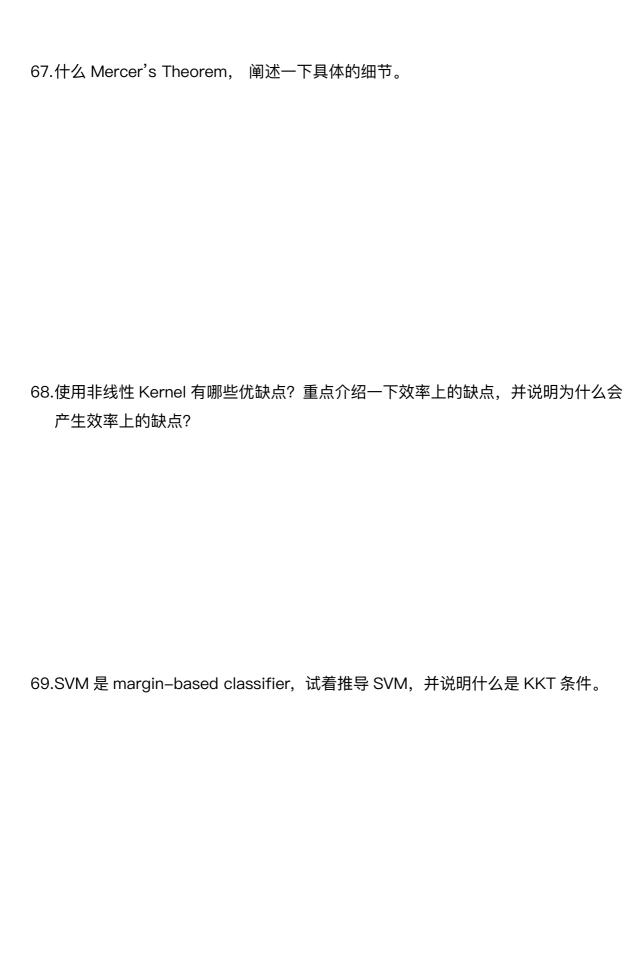


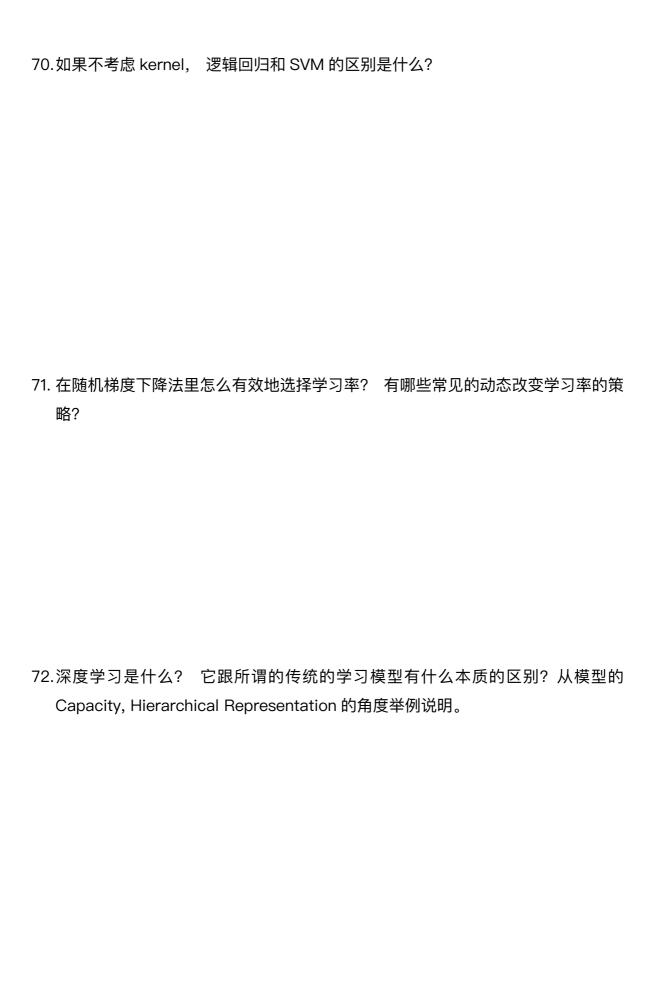


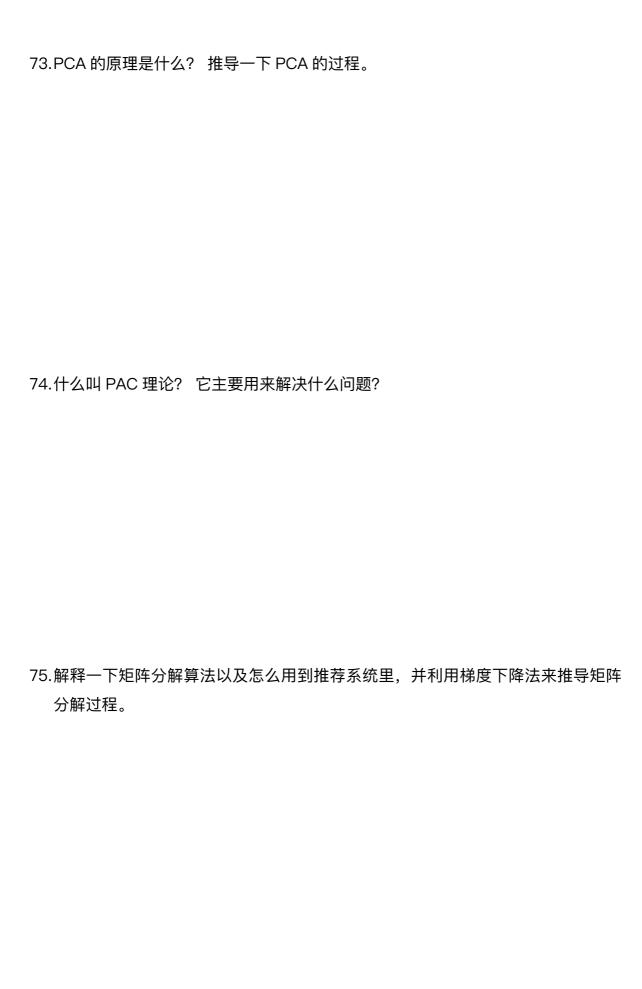


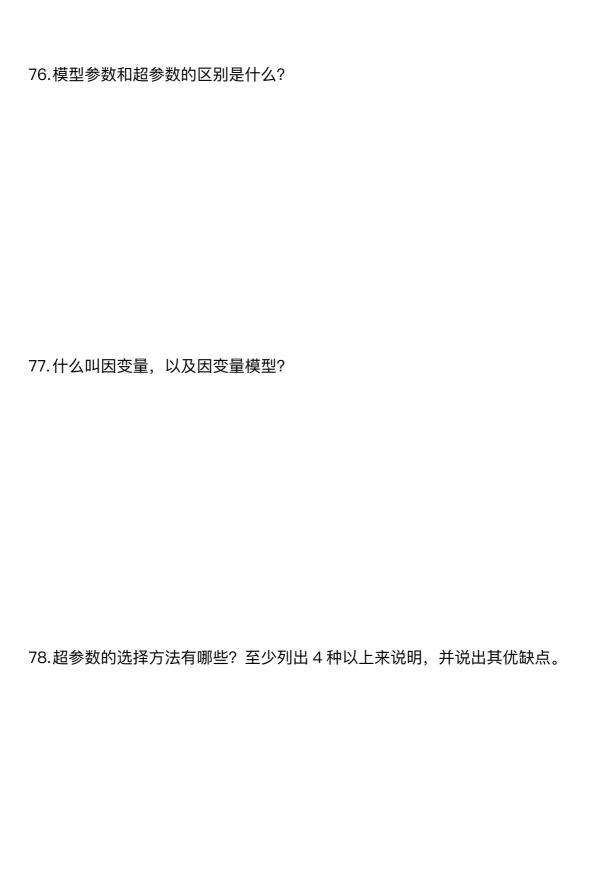


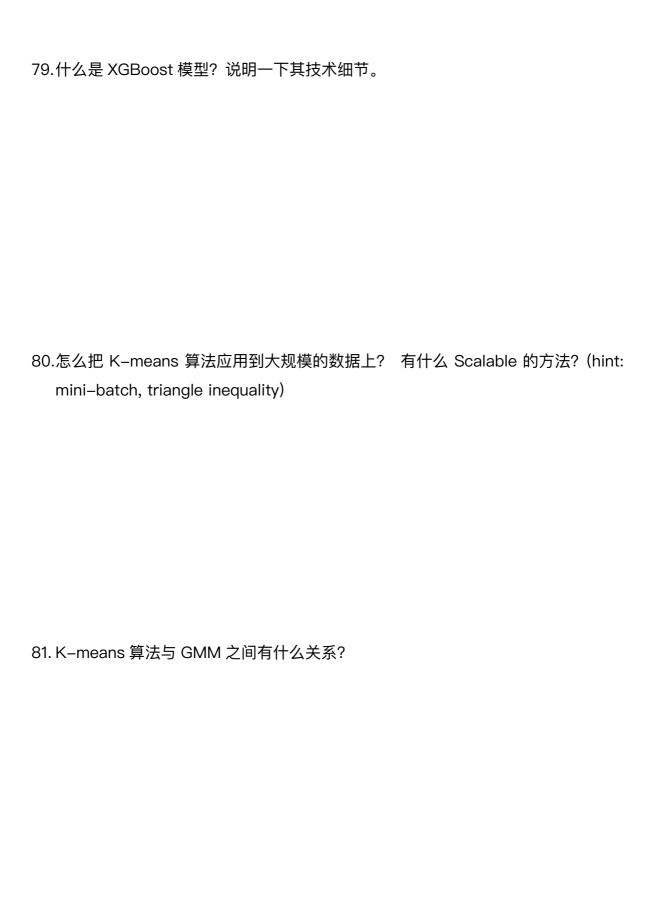


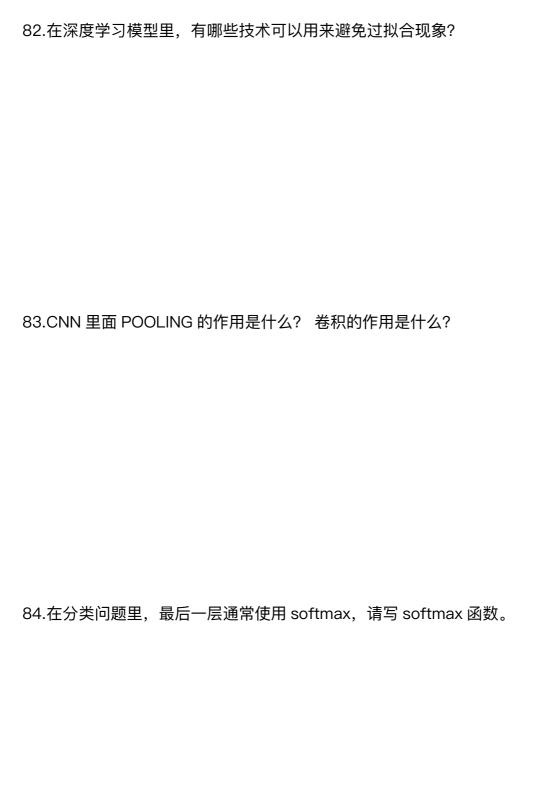


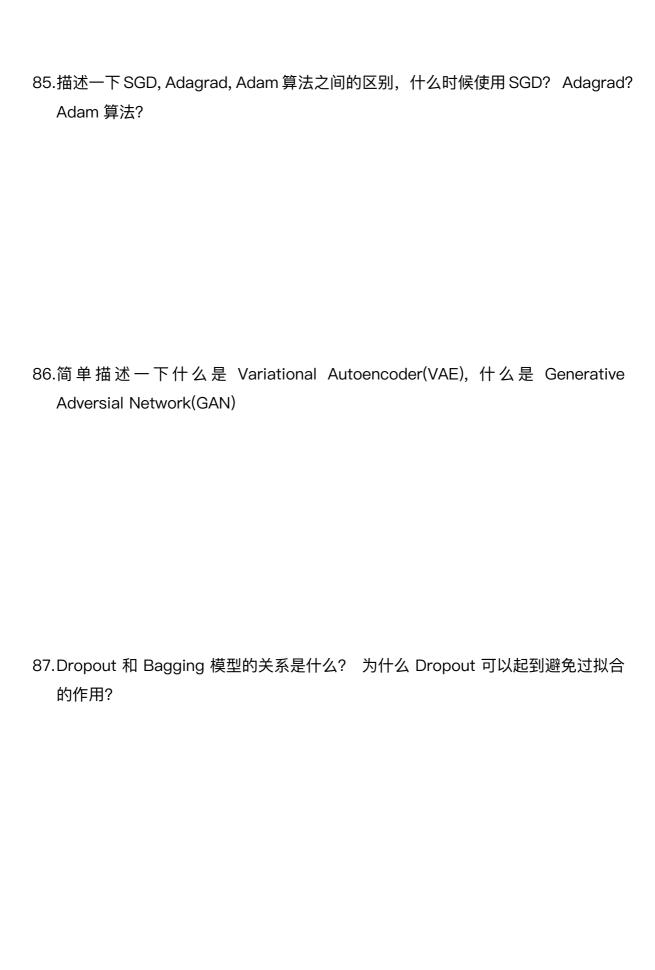




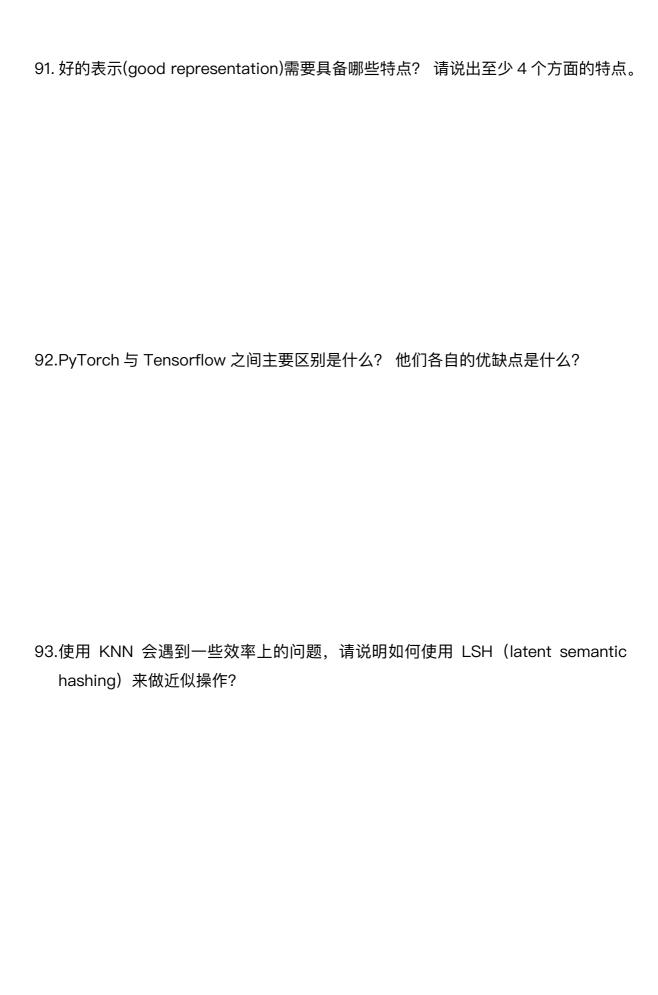












Part 4. 自然语言处理

