动态规划选讲

重庆南开中学信息学竞赛教练组

斜率优化DP

例1 nkoj 3974 最大平均值

给定一个长度为 n 数列 $\{a_n\}$，从中选一段长度至少为 L 的区间，可以计算区间内数字的平均值。求所有方案中的最大平均值。

$1 \le L \le n \le 3 * 10^5$

$1 \le a_i \le 2000$

例1 nkoj 3974 最大平均值

- 二分答案 x，将数列中每个元素减去 x，问题转化为判断数列 $\{a_n - x\}$ 中是否存在一段长度至少为 L 的区间，使得这段区间的和 ≥ 0
- 最大连续区间和，入门DP

例1 nkoj 3974 最大平均值

- 二分答案 x ，将数列中每个元素减去 x ，问题转化为判断数列 $\{a_n - x\}$ 中是否存在一段长度至少为 L 的区间，使得这段区间的和 ≥ 0
- 最大连续区间和，入门DP
- 当固定右端点 i 时:

$$maxsum_i = \max_{0\leq j\leq i-k}(S_i - S_j) = S_i - \min_{0\leq j\leq i-k} S_j$$

- 维护一下前缀最小值即可

- 复杂度 $O(n)$
- 总复杂度 $O(nlog(eps^{-1}))$

例1 nkoj 3974 最大平均值

- 是否存在 $O(n)$ 做法?
- 假设去掉二分，当固定右端点时，问题变为：

$$maxmean_i = \max_{0 \le j \le i-k} \frac{S_i - S_j}{i - j}$$

- 此时，这个值与 i 和 j 均有关，无法简单维护只与 j 有关的信息

例1 nkoj 3974 最大平均值

- 是否存在 $O(n)$ 做法?
- 假设去掉二分，当固定右端点时，问题变为：

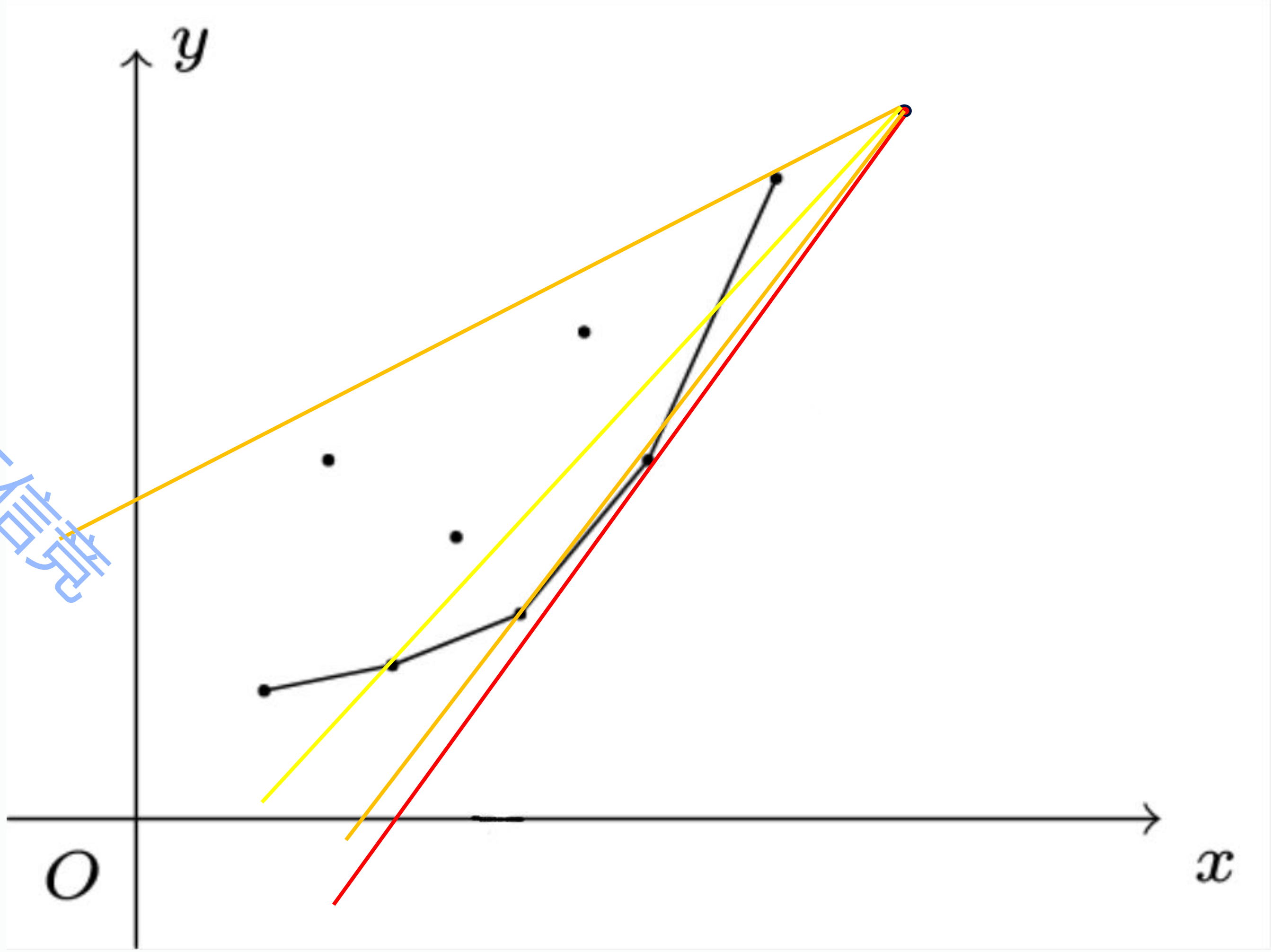
$$maxmean_i = \max_{0 \le j \le i-k} \frac{S_i - S_j}{i - j}$$

- 此时，这个值与 i 和 j 均有关，无法简单维护只与 j 有关的信息

- 数形结合
- 将 (i, S_i) 与 (j, S_j) 看作平面直角坐标系中两点
- 平均值 = 过这两点的直线的斜率

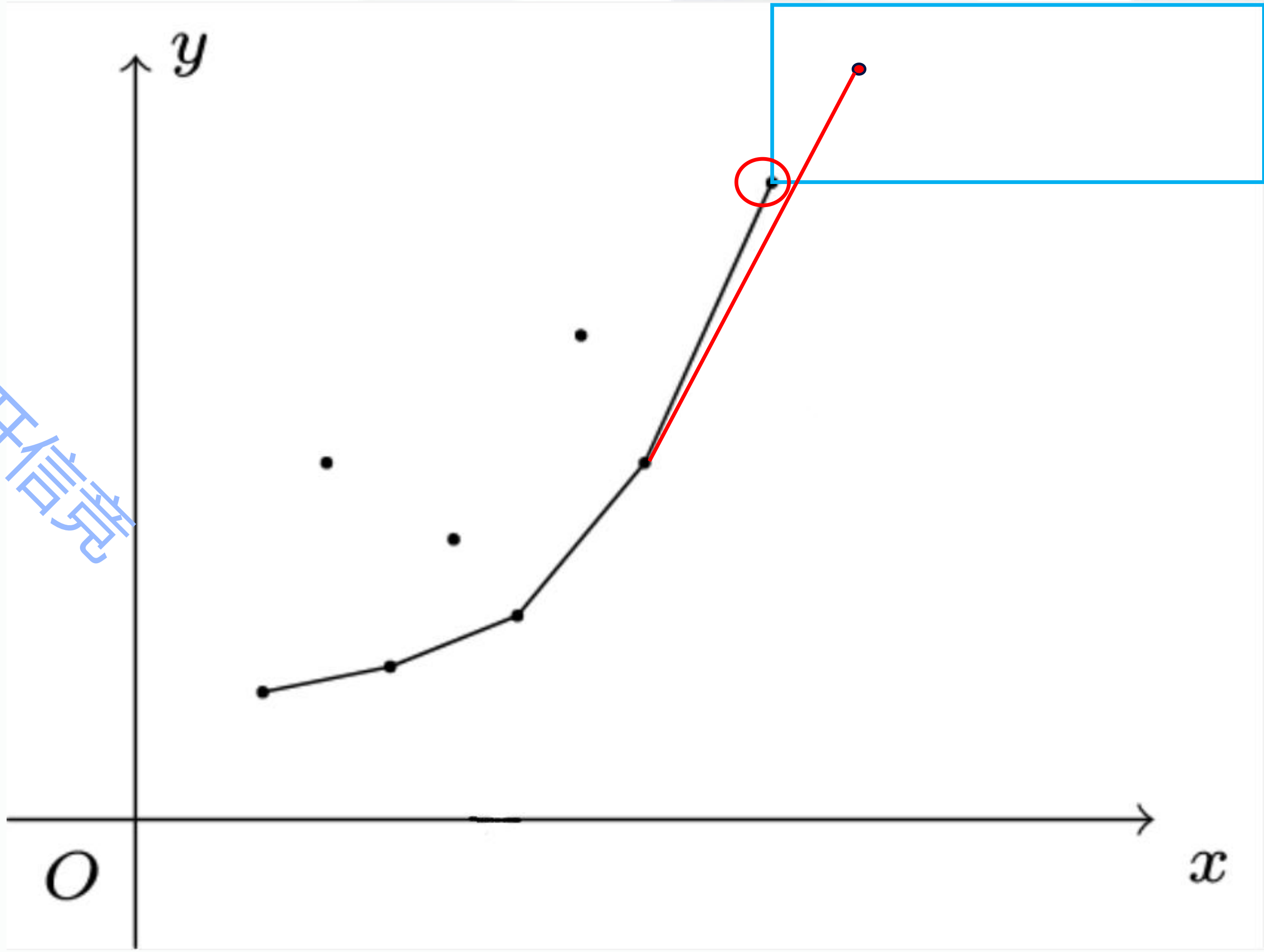
例1 nkoj 3974 最大平均值

- 固定右端点时，相当于固定一个端点 (i, S_i)，寻找使斜率最大的另一点 (j, S_j)
- 可能有效的点集构成了一个四分之一**凸包**，凸包中线段**斜率递增**
- 在凸包上找答案，即过定点作凸包的切线，比较通用的方法是**二分**找到切点



例1 nkoj 3974 最大平均值

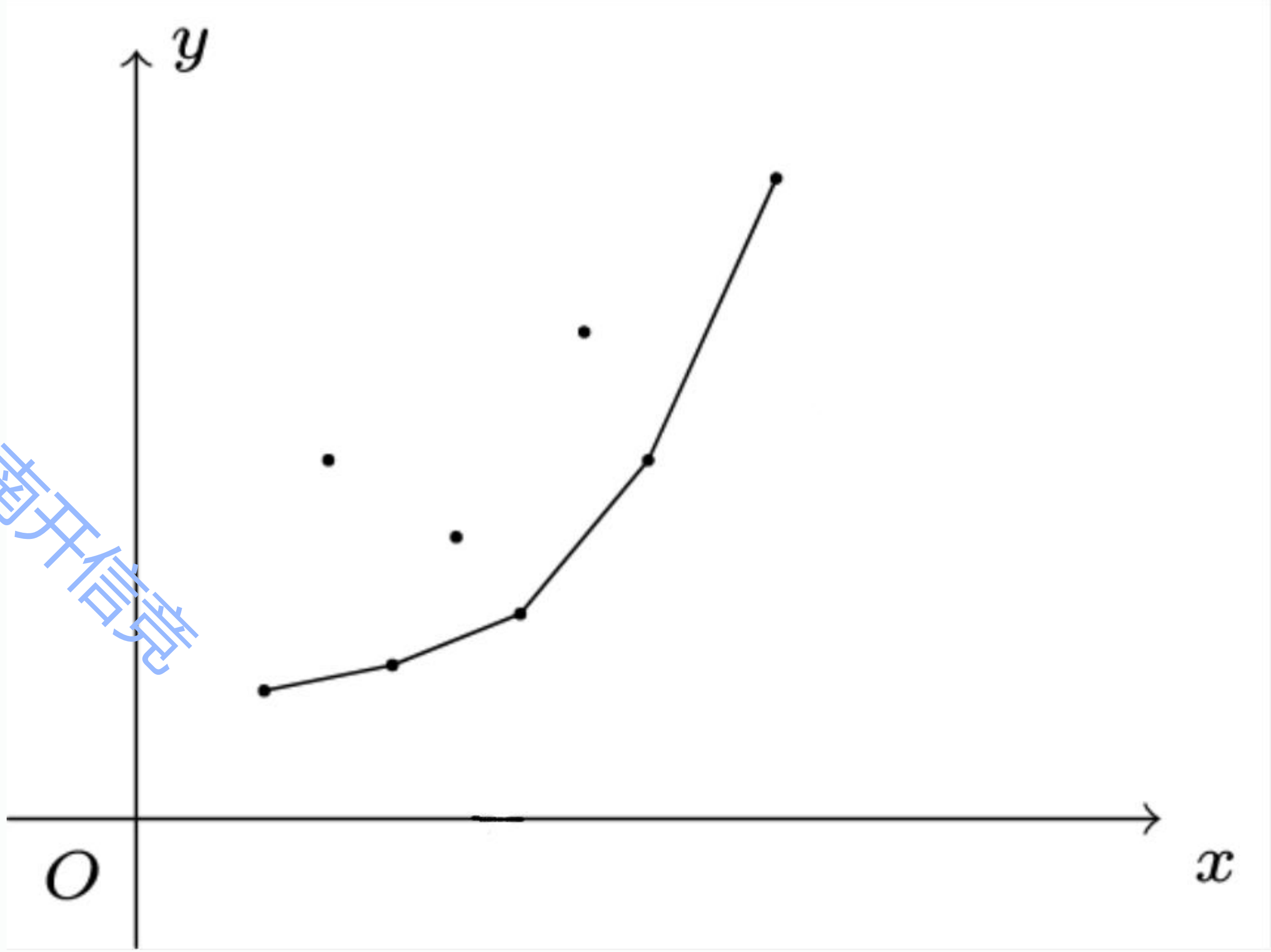
- 怎么维护凸包?
- 平衡树（毒瘤）
- 本题中，新加入的点 (i,S_i) 横纵坐标均单调不减，即位于当前凸包的右上方
- 加入该点后，只会使得凸包顶部的一些点失效，可用单调栈维护



事实上，只要横坐标 x 单调不减，就可以用单调栈维护

例1 nkoj 3974 最大平均值

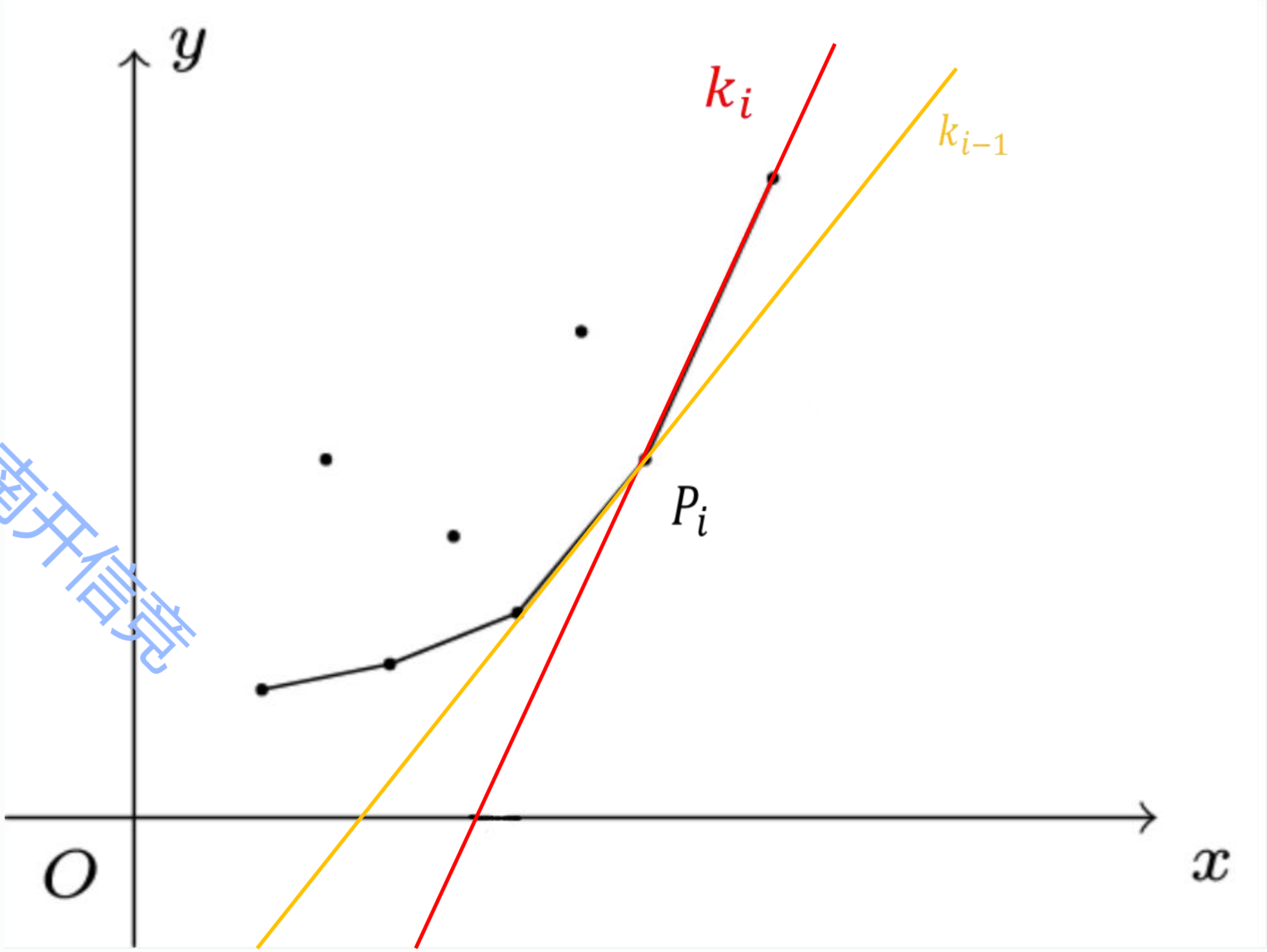
- 在凸包上找答案需要二分，总复杂度仍然是 $O(nlogn)$
- 本题还可以继续优化为 $O(n)$



例1 nkoj 3974 最大平均值

- 在凸包上找答案需要二分，总复杂度仍然是 $O(nlogn)$
- 本题还可以继续优化为 $O(n)$

- 考虑凸包上的一点 P_i ，当它成为询问的答案（即切点）时，答案（即斜率）的范围是 $[k_{i-1},k_i]$，此时，凸包上在 P_i 之前的点已经不可能贡献出比 k_{i-1}更大的答案了，可以删除

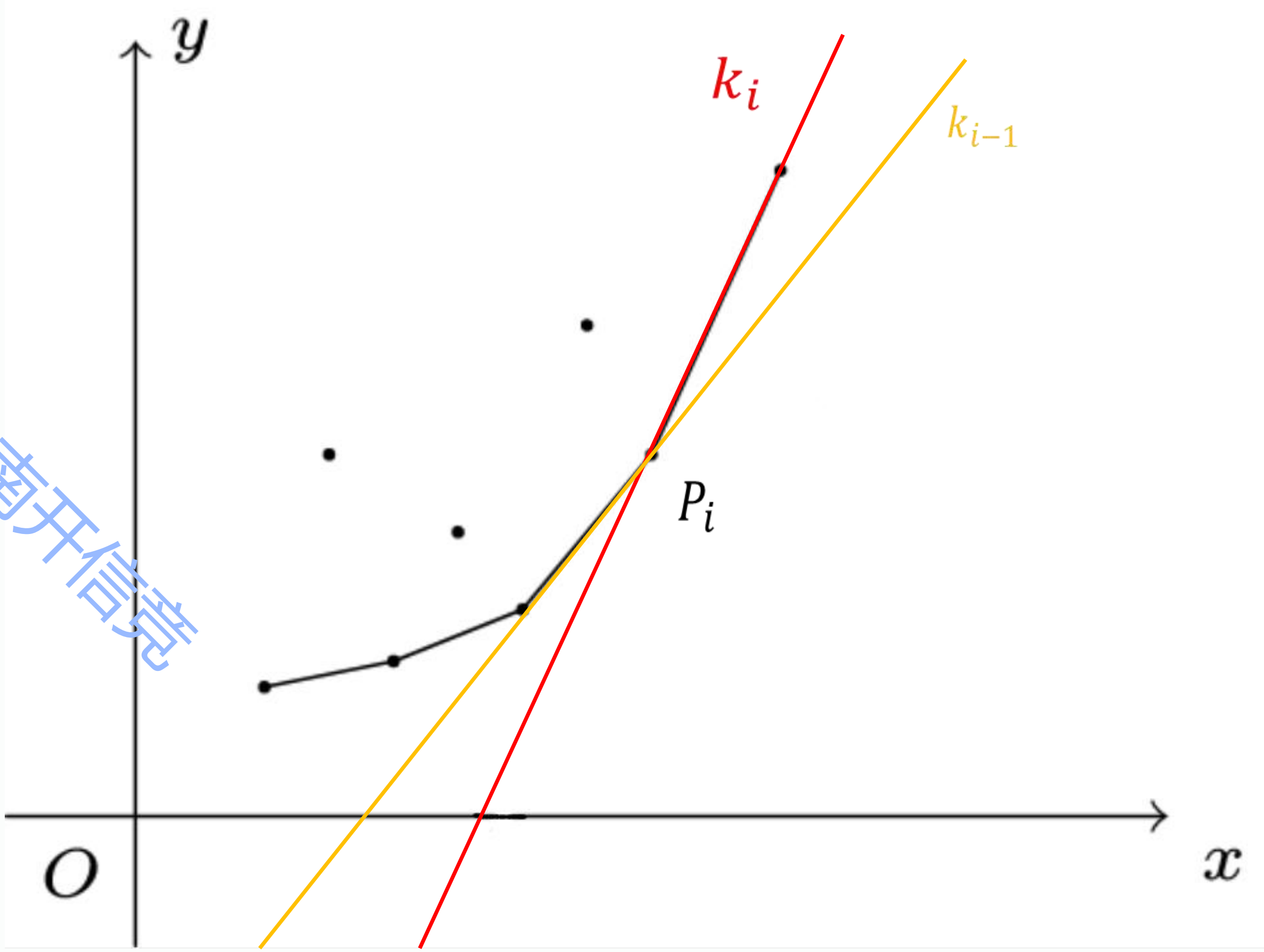


例1 nkoj 3974 最大平均值

- 要支持删除操作，把单调栈改为单调队列即可

- 注意，这个优化仅对求全局最优解生效。若要求每个右端点的答案，则必须在凸包上二分

- 每个点只会入队一次，出队一次，总复杂度 $O(n)$



例1 nkoj 3974 最大平均值

- 在计算过程中，为了避免精度误差，尽量避免使用 double 计算和比较斜率

- 可以通分后化为整数相乘进行比较（小心溢出）

- $\frac{dy_1}{dx_1} > \frac{dy_2}{dx_2} \Rightarrow dy_1 * dx_2 > dx_1 * dy_2$

- 若存在斜率相等（三点共线），一般可删除中间点

例1 nkoj 3974 最大平均值

- 参考代码

```
int h = 0, t = 0;
q[0] = 0;
for (int i = k; i <= n; ++i) {
    while (t < h && dy(i, q[t])*dx(i, q[t+1]) <= dy(i, q[t+1])*dx(i, q[t])) ++t;
    ans = max(ans, 1.0*dy(i, q[t])/dx(i, q[t]));
    int j = i - k + 1;
    while (t < h && dy(j, q[h])*dx(q[h], q[h-1]) <= dy(q[h], q[h-1])*dx(j, q[h])) --h;
    q[++h] = j;
}
```


- 先在凸包上（从队尾开始）找答案，并删去对答案无贡献的点
- 再往队首加入点，维护凸包上斜率的单调性

例2 nkoj 1919 [HNOI2008] 玩具装箱

P教授有 n 件玩具，第 i 件玩具经过压缩后长度为C_i 。P教授要将这些压缩后的玩具用若干个容器分装，为了方便整理，一个容器中的玩具编号必须是连续的。

如果将第 i 件玩具到第 j 个玩具放到一个容器中，那么容器的长度将为 $x = j - i + \sum_{k=i}^{j} C_k$

制作容器的费用与容器的长度有关，根据教授研究，如果容器长度为 x ，其制作费用为 $(x - L)^2$，其中 L 是一个常量。

P教授不关心容器的数目，他可以制作出任意长度的容器，甚至超过 L 。他只希望制造容器的总费用最小。

$1 \le n \le 5 * 10^4$

$1 \le C_i$、$L \le 10^7$

例2 nkoj 1919 [HNOI2008] 玩具装箱

- 不太能贪心，考虑DP

- 容易写出 $O(n^2)$ 的DP方程
- 设 f_i 为只考虑前 i 个玩具的最小费用，S_i 为 C_i 的前缀和

$$f_i = \min_{0\le j<i}\{f_j + \left(S_i - S_j + i - j - 1 - L\right)^2\}$$

- 上式没有明显的斜率表达式

例2 nkoj 1919 [HNOI2008] 玩具装箱

- 设 j 是 f_i 的最优转移点，则可将 min 去掉，转为等式

$$f_i = f_j + \left(S_i - S_j + i - j - 1 - L\right)^2$$

- 为了方便，设 $a_i = S_i + i$，$b_j = S_j + j + 1 + L$，则

$$f_i = f_j + \left(a_i - b_j\right)^2$$

$$f_i = f_j + a_i^2 + b_j^2 - 2a_ib_j$$

- 仍然没有明确的斜率式
- f_i 不好作为斜率，不妨考虑作为截距

例2 nkoj 1919 [HNOI2008] 玩具装箱

$$f_i = f_j + a_i^2 + b_j^2 - 2a_ib_j$$

- 设 $x = b_j$，$y = f_j + b_j^2$，$k = 2a_i$

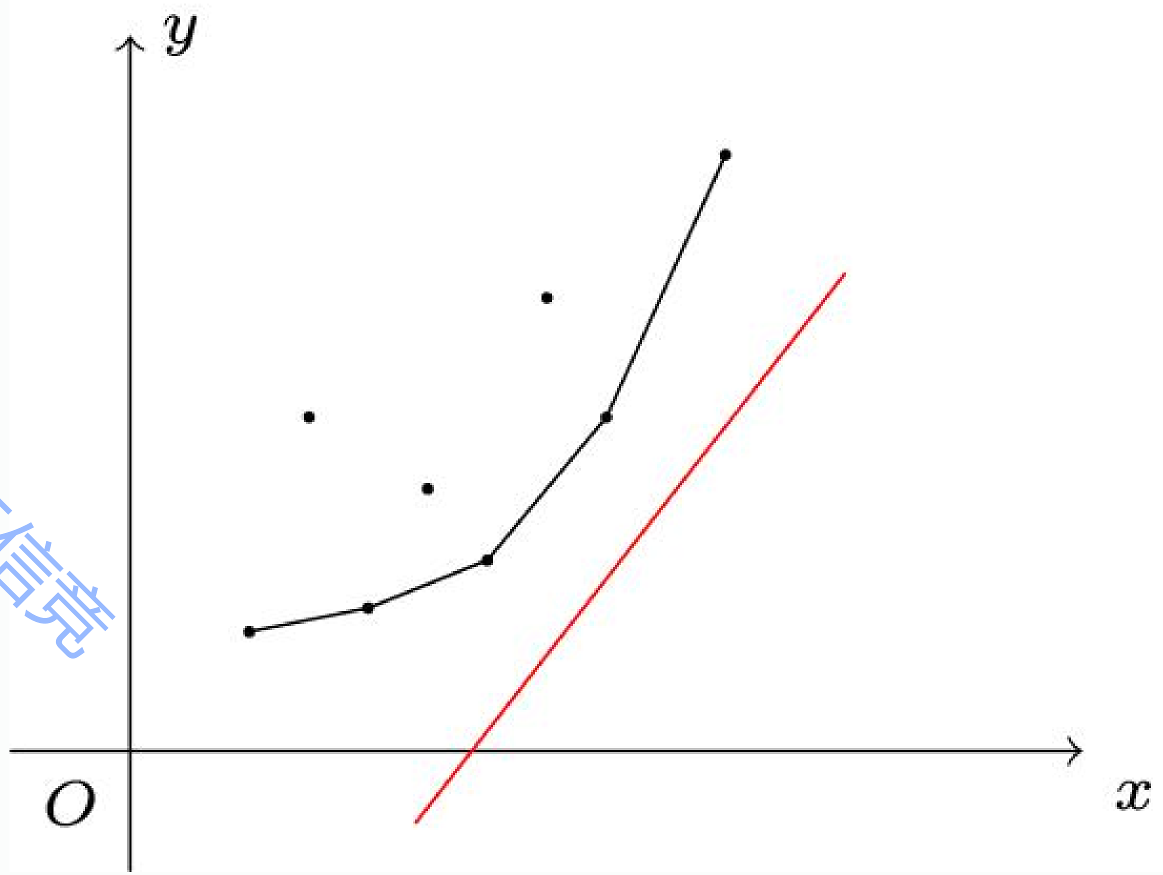
$$y = kx + f_i - a_i^2$$

- x 和 y 都只与 j 有关，而且是计算 i 之前就确定的，斜率 k 也是确定的
- 每一个决策 j 都可以当成点 (x, y)
- 问题转化为：给定平面上一些点，一条斜率固定为 k 的直线经过哪个点能获得**最小**的截距?
- 线性规划问题

例2 nkoj 1919 [HNOI2008] 玩具装箱

$$y = kx + f_i - a_i^2$$

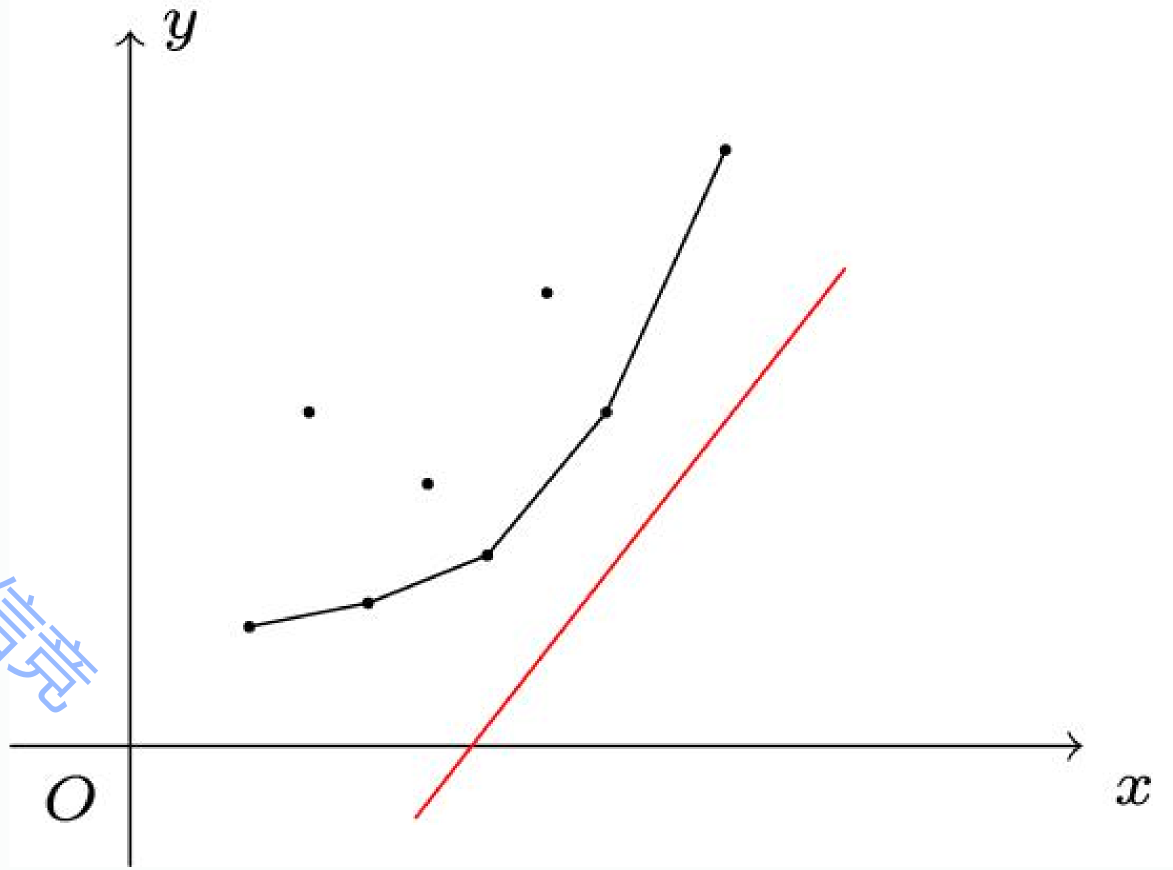
- 数形结合
- 固定斜率，最小化截距，最优解一定位于下凸壳上
- 另一种直线切凸包问题
- 横坐标 $x = b_j = S_j + j + 1 +$ 单调，可以用单调栈维护凸包
- 寻找答案依然可以在凸包上二分



例2 nkoj 1919 [HNOI2008] 玩具装箱

$$y = kx + f_i - a_i^2$$

- 注意到，本题中，斜率 $k = 2a_i$ 也是单调的，这意味着，决策点 j 也是单调的，这个性质也被成为决策单调性



- 因此，我们可以像之前一样，使用单调队列维护凸包，找到决策点 j 后，将之前的点全部删除
- 时间复杂度 $O(n)$

斜率优化DP

一般的，对于形如

$$f_i = \min/\max\{a_i * b_j + c_i + d_j\}$$

的1D/1D动态规划（其中a_i、b_j、c_i、d_j要求是仅与对应下标相关的常量，且能快速计算），可以使用例2中介绍的数形结合方法优化，这种方法叫做斜率优化。

通用套路：将乘积式$a_i * b_j$中的a_i当作斜率k，b_j当作横坐标x，和式中的d_j当作纵坐标y，而所求的f_i为截距的一部分。

例3 nkoj 1539 Land Acquisition

Farmer John 准备扩大他的农场，眼前他正在考虑购买 n 块长方形的土地。

如果单买一块土地，价格就是土地的面积。但他可以选择并购一组土地，并购的价格为这些土地中最大的长乘以最大的宽。比如并购一块 3×5 和一块 5×3 的土地，他只需要支付 $5 \times 5 = 25$ 元，比单买合算。

Farmer John 希望买下所有的土地。他发现，将这些土地分成不同的小组来并购可以节省经费。 给定每份土地的尺寸，请你帮助他计算购买所有土地所需的最小费用。

$1 \leq n \leq 5 * 10^4$

土地的长和宽不超过 10^6

例3 nkoj 1539 Land Acquisition

- 当 $w_i \le w_j$、$h_i \le h_j$ 时，第 i 块土地与第 j 块一起购买也不会产生额外费用，可以将第 i 块土地删除
- 剩余的土地构成了“阶梯”形状，按 w_i 升序排序，则 h_i 降序

例3 nkoj 1539 Land Acquisition

- 当 $w_i \le w_j$、$h_i \le h_j$ 时，第 i 块土地与第 j 块一起购买也不会产生额外费用，可以将第 i 块土地删除
- 剩余的土地构成了“阶梯”形状，按 w_i 升序排序，则 h_i 降序
- 若将 i 和 j 一起购买，则 $i \sim j$ 之间的所有土地都可以合并到一起购买，不会产生额外费用
- 至此，可以轻松写出DP方程

$$f_i = \min_{1 \le j \le i} \{f_{j-1} + w_i h_j\}$$

例3 nkoj 1539 Land Acquisition

$$f_i = \min_{1 \le j \le i} \{f_{j-1} + w_i h_j\}$$

该方程明显满足斜率优化的形式

$$f_{j-1} = -w_i h_j + f_i$$

- 横坐标 $x = -h_j$，单调递增，可单调栈维护

- 斜率 $k = w_i$，单调递增，决策单调，换为单调队列

例4 luogu P5785 [SDOI2012] 任务安排

机器上有 n 个需要处理的任务，它们构成了一个序列。这些任务被标号为 1 到 n。这 n 个任务被分成若干批，每批包含相邻的若干任务。从时刻 0 开始，这些任务被分批加工，第 i 个任务单独完成所需的时间是 T_i 。在每批任务开始前，机器需要启动时间 s ，而完成这批任务所需的时间是各个任务需要时间的总和。

注意，同一批任务将在同一时刻完成。每个任务的费用是它的完成时刻乘以一个费用系数 C_i 。

请确定一个分组方案，使得总费用最小。

$1 \le n \le 3 * 10^5$

$1 \le s \le 2^8$， $0 \le C_i \le 2^8$ ， $|T_i| \le 2^8$

例4 luogu P5785 [SDOI2012] 任务安排

- 本题是nkoj 1047的数据强化版
- 先思考（回顾）原题的 $O(n^2)$ 做法

例4 luogu P5785 [SDOI2012] 任务安排

- 本题是nkoj 1047的数据强化版
- 先思考（回顾）原题的 $O(n^2)$ 做法

- 要计算一批任务的代价，需要知道之前的任务被分为几段
- $f_{i,j}$ 表示前 i 个任务分为 j 段的最小费用，ST_i 、SC_i 为 T_i 和 C_i 的前缀和

$$f_{i,j} = \min_{0 \le k < i} \{f_{k,j-1} + (ST_i + s * j) * (SC_i - SC_k)\}$$

- 这是 $O(n^3)$ 的
- 考虑如何去掉表示段数的 j 这一维

例4 luogu P5785 [SDOI2012] 任务安排

- 当前这一批次任务的执行，会使得 i 之后的任务完成时间时刻增加 s ，这部分所产生的费用为 $(SC_n - SC_i) * s$，这仅与 i 相关，不妨并入 f_i 中，得到

$$f_i = \min_{0 \le j < i}\{f_j + (ST_i + s) * \left(SC_i - SC_j\right) + (SC_n - SC_i) * s\}$$

$$= \min_{0 \le j < i}\{f_j + ST_i * \left(SC_i - SC_j\right) + \left(SC_n - SC_j\right) * s\}$$

- 该式看起来很斜率优化，化一化

例4 luogu P5785 [SDOI2012] 任务安排

- 当前这一批次任务的执行，会使得 i 之后的任务完成时间时刻增加 s ，这部分所产生的费用为 $(SC_n - SC_i) * s$，这仅与 i 相关，不妨并入 f_i 中，得到

$$f_i = \min_{0 \le j < i}\{f_j + (ST_i + s) * \left(SC_i - SC_j\right) + (SC_n - SC_i) * s\}$$

$$= \min_{0 \le j < i}\{f_j + ST_i * \left(SC_i - SC_j\right) + \left(SC_n - SC_j\right) * s\}$$

- 该式看起来很斜率优化，化一化

$$f_j = (ST_i + s) * SC_j + f_i - SC_n * s - ST_i * SC_i$$

- 斜率 $k = ST_i + s$，不单调！注意到数据范围 $|T_i| \le 2^8$
- 此时，需要在凸包上二分找答案

例5「CEOI2017」Building Bridges

有 n 根柱子依次排列，每根柱子都有一个高度 h_i 。

现在想要建造若干座桥，在第 i 根柱子和第 j 根柱子之间建一座桥需要 $(h_i - h_j)^2$ 的代价。

在造桥前，中间所有用不到的柱子都会被拆除，因为他们会干扰造桥进程。第 i 根柱子被拆除的代价为 w_i ，注意 w_i 不一定非负，因为可能政府希望拆除某些柱子。

现在政府想要知道，通过桥梁把第 1 根柱子和第 n 根柱子连接的最小代价。注意桥梁不能在端点以外的任何地方相交。

$2 \le n \le 10^5$

$0 \le h_i$、$|w_i| \le 10^6$

例5「CEOI2017」Building Bridges

- $O(n^2)$ 的 DP 很好列

$$f_i = \min_{1 \le j < i}\{f_j + (h_i - h_j)^2 + S_{i-1} - S_j\}$$

- 一眼斜率优化

$$f_j + h_j^2 - S_j = 2h_ih_j + f_i - h_i^2 - S_{i-1}$$

例5「CEOI2017」Building Bridges

- $O(n^2)$ 的 DP 很好列

$$f_i = \min_{1\le j<i}\{f_j + \left(h_i - h_j\right)^2 + S_{i-1} - S_j\}$$

- 一眼斜率优化

$$f_j + h_j^2 - S_j = 2h_ih_j + f_i - h_i^2 - S_{i-1}$$

- 横坐标 $x = h_j$，不单调
- 斜率 $k = 2h_i$，不单调
- 平衡树上二分?
- 解决偏序问题，可用cdq分治

例5「CEOI2017」Building Bridges

- 先递归 (l, mid) ，完成它们 f 值的计算
- 然后考虑对 $(mid+1, r)$ 的贡献，即处理(l, mid) 的点到 $(mid+1, r)$ 点的转移
- 将 (l, mid) 的点构造凸包，由于是静态构造，可按 x 排序后单调栈维护
- 对 $(mid+1, r)$ 的点，依次在凸包上找答案

例5「CEOI2017」Building Bridges

- 先递归 (l, mid)，完成它们 f 值的计算
- 然后考虑对 $(mid+1, r)$ 的贡献，即处理(l, mid) 的点到 $(mid+1, r)$ 点的转移
- 将 (l, mid) 的点构造凸包，由于是静态构造，可按 x 排序后单调栈维护
- 对 $(mid+1, r)$ 的点，依次在凸包上找答案
- 二分?
- 可对 $(mid+1, r)$ 的点按斜率排序，再依次询问，这样便有了决策单调性
- 最后再递归处理 $(mid+1, r)$，它们完成内部的转移
- 复杂度 $O(nlog^2n)$

斜率优化总结

形如

$$f_i = \min/\max\{a_i * b_j + c_i + d_j\}$$

乘积式中与i相关的a_i当斜率，与j相关的(b_j, d_j)当点，答案在截距中

$$x\begin{cases}\text{单调：单调数据结构}\\ \text{不单调：平衡树/cdq分治}\end{cases}$$

$$k\begin{cases}\text{单调：双指针}\\ \text{不单调：凸包上二分}\end{cases}$$

练习题

快乐暑期作业5

A~J

密码：nkaker5