Problem G
Linear Regression
Algorithm 해설

주어진 데이터 값들인 $\{(x_1, y_1), \dots, (x_n, y_n)\}$ 중에서 k개를 삭제할 수 있을 때, 가능한 최적의 선형 함수와 그때의 오차값을 계산하는 문제이다. 이 문제는 주어진 데이터 값들을 2 차원에서의 점으로 생각하고 point-to-line duality 를 이용하면, n개의 직선들 중에서 $n-k$개 이상을 교차하는 가장 짧은 수직 선분을 찾는 문제와 같다.

이를 해결하기 위해 n개의 직선들의 arrangement A 를 계산하고 각 vertex 에서 출발하는 수직 선분을 고려하는 방식을 취하면 $\Omega(n^2)$ 이상의 시간을 소요하게 되어 시간 초과가 나게 된다.

직선들의 arrangement 에서 level 을 생각해야 한다. 1 이상 n이하의 정수 m에 대해, A 의 m-level 은 위에 존재하는 직선들의 개수가 $m-1$ 개인 edge 들의 체인으로 정의된다. 따라서 A 의 upper envelope 은 1-level 이며, lower envelope 은 n-level 이다.

이 문제에서 요구하는 솔루션은 1 이상 $k+1$이하인 m에 대해, m-level 과 $(n-m+1)$-level 만을 계산하여, 각각을 linear scan 하는 방식을 취하면 된다. 이론적으로는 $O(n \log n + kn)$ 시간에 동작하는 알고리즘이 존재하나, 현실적으로 구현가능한 방법은 아래와 같으며 시간복잡도는 $O(kn \log n + k^{\frac{4}{3}}n)$이다.

1. 입력으로 주어진 점들 P 의 convex layer 를 k 단계까지 계산한다. Convex layer 의 1 단계는 P 의 convex hull, 2 단계는 P 의 convex hull 위의 점들을 삭제한 나머지들의 convex hull 로 이루어지며 이를 반복한다. 여기서 최대 $O(kn \log n)$ 시간이 사용된다.
2. 1 에서 계산한 convex layer 를 이용하여, P 의 dual 인 직선들의 arrangement 에서의 $(k+1)$-level 과 $(n-k)$-level 만을 계산한다. 이는 P 에서 rotational line sweep 을 통해 가능하다. 시작은 P 에서 왼쪽에서부터 $k+1$ 번째 점을 지나는 수직선에서 출발하여, 그 점을 중심으로 반시계방향으로 돌리다가 다른 점을 지나는 순간 회전의 중심을 바꾸는 방식으로 이루어지며, 회전 중심에 있게 되는 점의 dual 이 $(k+1)$-level 에서의 한 edge 와 상응하게 된다. 이를 효율적으로 구현하는 방법은 여러가지가 가능할 수 있으나 convex layer 를 이용하면, 다음에 회전 중심이 될 점들은 항상 convex layer 의 각 단계에서 최대 2 개 중에서 선택된다는 것, 그리고 $(k+1)$-level 의 복잡도는 $O(nk^{\frac{1}{3}})$이라는 것에 의해 총 $O(nk^{\frac{4}{3}})$ 시간에 구현이 가능하다.

3. 위에서 계산한 $(k+1)$-level 과 $(n-k)$-level 을 이용하여 dual 직선들의 arrangement 에서 $(k+1)$-level 의 윗부분과 $(n-k)$-level 의 아랫부분만을 계산한다. 이는 각 level 의 위와 아래를 지나는 선분들의 교차점을 순서대로 계산하는 plane sweep 방법을 통해 계산 가능하다. 해당 부분의 총 복잡도는 최대 $O(nk)$라는 사실에 의해, 또 sweep 의 매 순간 유지해야 하는 직선 (혹은 선분)의 개수가 항상 k개라는 점에 의해 $O(nk\log k)$ 시간에 구현 가능하다.
4. 1 이상 $k+1$ 이하인 각 i에 대해, i-level 과 $(n-k-1+i)$-level 을 동시에 linear scan 하며 각 vertex 와 반대쪽 line 과의 거리를 계산하여 최솟값을 찾고 그것의 절반을 출력한다. $O(nk)$시간.

위와 같은 방법이 아니더라도 여러가지 방법이 가능할 수 있으나, n과 k의 제한에 따라 $\Omega(nk^2)$ 시간 혹은 $\Omega(n^2)$ 시간이 필요한 솔루션은 시간 초과되도록 하였다.