CS 506, HW 1

Prof. Jared Saia, University of New Mexico

You are encouraged to work on this omework in groups of 2 or 3. You may turn in one writeup per group, but please certify that all members worked on each problem. Several of these problems are from the book "Computational Geometry (third edition)" by Berg, et al. Mount's notes and homework problems are available in the link off the course web page.

1. In the online convex hull problem, we are given a set of n points one at a time. After receiving each point, we compute the convex hull of all points seen so far. Consider this problem in the 2D plane. Give an efficient online algorithm to update the convex hull when a new point is given. Analyze your algorithm.

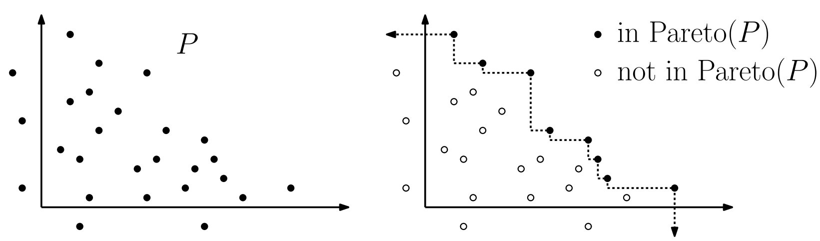


Figure 1. Example Pareto optimal figure (from David Mount hw).

2. Problem 2, HW 1 from David Mount's class (Pareto Optimal/Convex Hull problem) quoted below.
Consider a set of points $P = \{p_1, \dots p_n\}$ in the plane where $p_i = (x_i, y_i)$. A *Pareto set* for P, denoted $Pareto(P)$ is a subset of points P' such that for each $p_i \in P'$, there is no $p_j \in P$ such that $x_j \geq x_i$ and $y_j \geq y_i$. That is, each point of $Pareto(P)$ has the property that there is no point of P that is both to the right and above it. Pareto sets are important whenever you want to optimize *two* criteria (e.g. accuracy and precision of a machine learning algorithm, cheapness and flight "shortness" for airline tickets, etc.), since they represent the optimal

"envelope" of possible solutions.

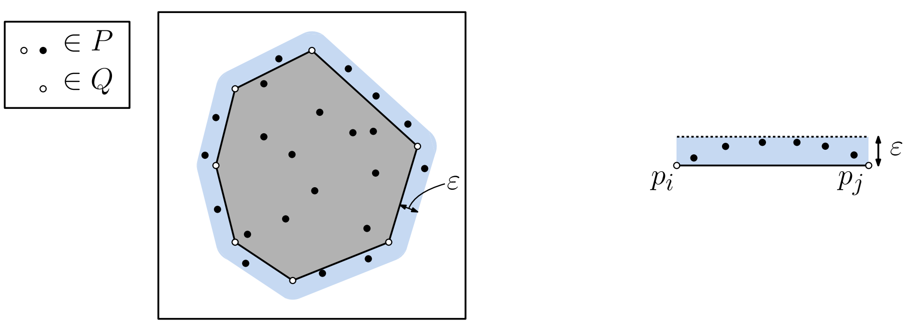
This problem explores the many similarities between Pareto sets and convex hulls. Whenever a problem asks for an algorithm, briefly justify correctness of your algorithm, explain any non-standard data structures, and derive the runtime.

(a) A point p lies on the convex hull of a set P if and only if there is a line passing through p such that all the points of P lie on one side of this line. Provide an analogous assertion for the points of $Pareto(P)$ in terms of a different shape.

(b) Devise an analogue of Graham's convex-hull algorithm for computing $Pareto(P)$ in $O(n \log n)$ time. Briefly justify your algorithm's correctness and derive its runtime. (You don't need to explain the algorithm "from scratch"; you can explain what modifications need to be made to Graham's algorithm.)

(c) Devise an analogue of Jarvis march algorithm for computing $Paret(P)$ in $O(hn)$ time where h is the cardinality of $Pareto(P)$. (As in the last part, you can just explain the differences with Jarvis's algorithm.)

(d) Devise an algorithm for computing $Pareto(P)$ in $O(n \log h)$ time. Hint: Chan!



(a) (b)

Figure 2. Computing an ϵ-sketch (from David Mount hw)

3. Based on (Problem 4, HW 3 from David Mount's class (ϵ sketch of convex hull))

You are given a set P of n points lying in the unit square in the plane.

Given any subset $Q \subseteq P$, clearly we have $\text{conv}(Q) \subseteq \text{conv}(P)$. For $\epsilon > 0$, we say that Q is an ϵ-sketch of P if every point of P lies within distance at most ϵ of $\text{conv}(Q)$ (See Figure 2 (a)).

(a) Consider the following simple greedy algorithm for computing an ϵ-sketch of a planar point set P. First let $\langle p_0, \ldots, p_{k-1} \rangle$ denote the vertices of $\text{conv}(P)$ listed in counterclockwise order. Put p_0 in Q and set $i \leftarrow 0$. Find the largest index j, $i < j \leq k$ such that all the points $\{p_{i+1}, \ldots p_{j-1}\}$ lie within distance ϵ of the line segment p_i, p_j (See Fig 2(b)). (Indices are taken modulo k so $p_k = p_0$.) If $j = k$ then stop. Otherwise, add p_j to Q, set $i \leftarrow j$ and repeat. Show that this procedure correctly produces an ϵ-sketch of P.

(b) Show that the perimeter of the convex hull P has total length $O(1)$. Hint: Show that the perimeter of a convex polytope inside another convex polytope must be smaller than the outer polytope. To show this, think about what happens to the perimeter when you repeatedly "slice out" half-planes in order to carve out the interior polytope from the surrounding polytope.

(c) Next, show that the amount of perimeter of P traversed by the greedy algorithm between any two points added to Q is always $\Omega(\epsilon)$. To show this, let v_1 be the initial hull vertex in some round of the greedy algorithm and let v_x be the first hull vertex that has distance greater than ϵ from v_1. So $d(v_1, v_j) \leq \epsilon$ for all $j \in [1, x-1]$ where d(x,y) is the the Euclidean distance between points x and y. Now what can you say about the distance between v_j and the line between v_1 and v_x??? Hint: draw a picture and find right triangles.

(d) Now, prove that $|Q| = O(1/\epsilon)$. I.e., that the size of the sketch is completely independent of the number of vertices of the original convex hull (P). What implication does this have on the amount of space needed to store an ϵ-approximation of all crepe recipes?

4. Hard: Can you adapt the ϵ-sketch convex hull problem to come up with a similar type of sketch of the upper envelope in an arrangement? What can you say formally about the number of lines in your sketch of the upper envelope and how well the sketch approximates the true upper envelope? (Challenge: Any connections to sketching a Voronoi diagram?) Hint: Duality preserves vertical distances between a point and a line, but not the normal distances. Are there constraints on the

line slopes that will allow you to get a correspondence between these two distances?