CS341

Lecture 6

Yaro Gorban

September 30, 2015

Non dominant points problem:

- Has a worst case of O(n^2) by comparing each set of points
- not-dominated points for a "staircase"
- all other points are under the stair case
- split into S_1 and S_2
- solve both halves
- to combine we may have to eliminate some of the non-dominated points in S_1
- this runs in O(n) time

Complexity

1. pre-sort $\Theta(n \log n)$
2. $T(n) = T(\lfloor \frac{n}{2} \rfloor) + T(\rceil \frac{n}{2} \lceil) + O(n)$
3. sloppy form $T(n) = 2T(\frac{n}{2} + O(n)$ $T(n) \in O(n \log n)$
4. since we have 2 halves of $n \log n$ time we have a time of O(nlogn)

Closest Pair:

- Find a Euclidean distance that is minimized
- Trivial Algorithm is comparing each point which is O(n^2
- Presort with respect to x-coordinates $\Theta(n \log n)$
- Split the points down the middle
- Find the δ_L and the δ_R

- Combine: determine the smallest distance between a point on the left and the the right. is this distance $<\min\delta_L, \delta_R$
- Checking every point is not what we want to do
- We create something to create the critical strip
- But what if all the points fall in the critical strip? We didn't cut the run time at all.
- So we sort the points in R(list of candidates) with respect to the y-coordinates
- **LEMMA:** Suppose the points in the critical strip R are sorted WRT y-coordinates. Suppose that R[j] and R[k] have a distance $<\delta$ where j $<$k. Then k $\leq$ j+7

Complexity

1. presort O(nlogn)
2. $T(n) = 2T(\frac{n}{2}) + O(n)(selectcandidates) + O(n \log n)(sort) + O(n)(checkstrip)$
3. T(n) is $O(n(\log n)^2)$