04/27/2016

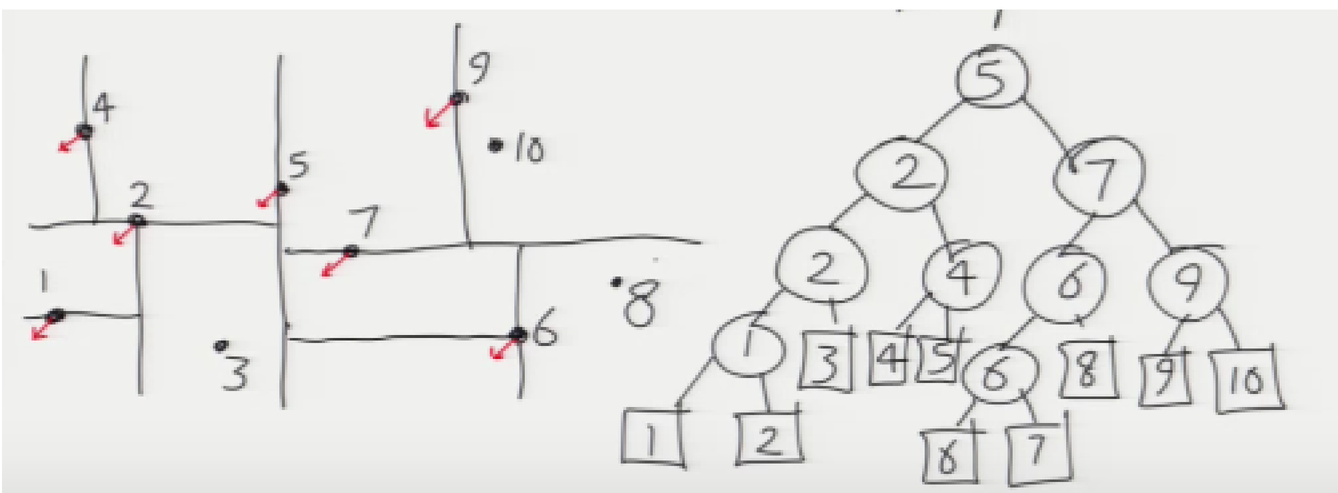
Speeding up NN

Voronoi diagrams

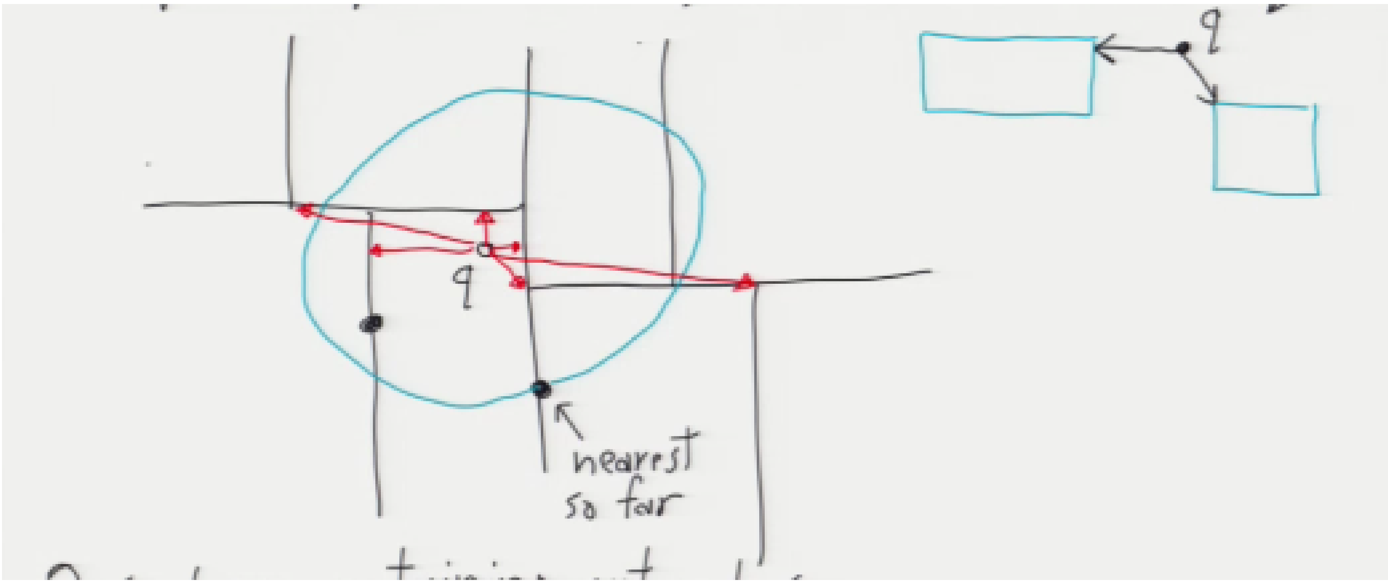
- Let P be a point set. The voronoi cell of $w \in P$ is $Vorw = \{p \in \mathbb{R}^d : |pw| \leq |pv| \ \forall v \in P\}$.
- The Voronoi diagram of P is the set of all $P's$ voronoi cells.
- Size (e.g. # of vertices) $\in \mathcal{O}(n^{\lceil \frac{d}{2} \rceil})$ .. but often in practice is in $\mathcal{O}(n)$.
- Point location: Given query point v, find the point w for which $v \in Vorw$.
- 2D: $\mathcal{O}(n \log n)$ time to compute V.d. and a trapezoidal map for point location. $\mathcal{O}(\log n)$ query time.
- dD: Use binary space partition tree (BSP tree) for point location.
- 1-NN only! What about k-NN?
- order-k voronoi diagram has a cell for each possible combination of k-NN.
- In 2D, size $\in \mathcal{O}(k^2 n)$.

k-d Trees

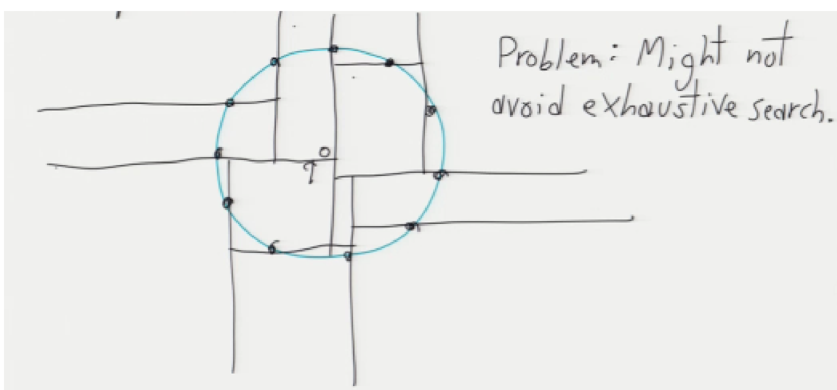
- Decision trees for NN search. Differences:
  - No entropy. Split dimension w/greatest variance or width (max-min).
  - Each internal node stores a sample point.



- Given query point q, find a sample point p such that $|qp| \leq (1 + \epsilon)|qs|$ where s is the closest point.
- The algorithm maintains:
  - Nearest neighbor found so far (or k nearest)
  - Heap of unexplored subtrees, keyed by distance from q.



$Q \leftarrow$ heap containing root node of tree
$r \leftarrow \infty$
while Q has a cell closer to q than $\frac{r}{1+\epsilon}$:
    $C \leftarrow removeMin(Q)$
    $p \leftarrow C's$ sample point
    $r \leftarrow \min\{r, |qp|\}$
    $C', C'' \leftarrow$ child cells of C
    $insert(Q, C')$
    $insert(Q, C'')$
return point that determined r



- For k-NN, replacing "r" w/a max-heap holding the k nearest neighbors.
- Works w/any L_p−norm for $p \in [1, \infty]$.
- Software:
  - ANN (David Mount, and Sunil Aryan, U. Maryland)
  - FLANN (Marius Muja and David Lowe, U. British Columbia)
  - GeRaF (Georgios Samaras, U. Athens)