EC - 504 Project

Nearest State/Country Finder

Hitanshi Jain
Renyu Jiang
Shuyan Zhang
Yuxiang Wan

Project Mission

- Load the reference points in an efficient data structure.
- Allow users to input latitude and longitude into the data structure
- Return the K nearest reference points as output.
- Find the state and county of the input reference point by computing a majority voting among the 5 nearest points.

Github repo: https://github.com/hitanshijain/Nearest_State_Finder

Implementation

- **For distance computation between two points, we can use the “equirectangular approximation”**

$x = (\lambda2-\lambda1) * \mathrm{Cos}((\varphi1+\varphi2)/2);$

$y = (\varphi2-\varphi1);$

$\mathrm{Distance} = \mathrm{Sqrt}(x*x + y*y) * R;$

where φ is latitude, λ is longitude, R is earth’s radius (mean radius = 6371 km);

A:
Construct an AVL
O(NlogN)
+
AVL find 10 smallest
worst case: O(logN)

B:
Median Quicksort O(N + NlogN) Even in the worst case

Doubted:

```
C: Best O(2N) Worst(N + N^2)
    1. Find the largest L and smallest S O(N)
    2. Idea From Dictionary Search: Best O(N) Worst O(N^2)
        Array A in size of N
        Smallest: A[0] = S
        Biggest: A[N-1] = L
        New element: A[N*(NEW - S)/(L - S)]
        Idx = N*(NEW - S)/(L - S)
        While (A[idx] != 0 and A[idx] < New){
            Idx += 1
        }
        While(A[idx] != 0 and A[idx] >= New){
                Idx -= 1
        }
        (May have problem when the values are too closed)
        A[Idx] = New
    3. A[0:10]
```

Another Possible implementation:

C++ Map:
The key values are in increasing order by default
Searching time: O(n)
Insertion time: O(log(n) + Rebalance)
Deletion time: O(log(n) + Rebalance)

Key: Distance
Value: Index

```
idx = std::next(map, 10);
```

Thank you