CS 521
Data Mining Techniques
Instructor: Abdullah Mueen

LECTURE 8: LOCALITY SENSITIVE HASHING

Locality Preserving Dimensionality Reduction

- If two points are close ➔ they are close in low-dim space
  - True for most dim-reduction techniques
- If two points are far ➔ they are far in low-dim space
  - Only true in some dim-reduction technique

ε-NNS problem

Given a set P of points in a metric space, preprocess P so as to efficiently return a point $p \in P$ for any given query point q, such that $d(q,p) \leq (1 + \varepsilon) d(q,P)$, where $d(q,P)$ is the distance of q to the its closest point in P.

Hashing

For each object p

- For j = 1 to L
  - Project p on k random directions $r_{j1}, r_{j2}, \ldots r_{jk}$
  - Form a hash key $g_j(p)$ for the projected point
  - Insert p in the j^{th} hash table
- Set k =log (M) and L = M^a where M is the number of buckets in each hash function and **0<a<1** is a parameter to specify sensitivity. The success of the method depends on the value of **a.**
- M depends on the memory available.

Searching

For the query q

- Calculate L hash keys
- Find the colliding objects in the buckets and search exactly to find NN
- The NNs will be within a small approximation away from the true NN

```
data = E.ECG;

Q = reshape(data(1:128*700), 128 , [] )';

r = randn(10,128);

Q1 = sum(Q*r‘,2);

hist(Q1)

min(Q1)

max(Q1)

table = norminv(0.000001:1/100:1-0.00000001,mean(Q1),std(Q1,1))

hist(Q1, table)
```

Papers

http://www.mit.edu/~andoni/LSH/