Outlier Detection by Sampling with Accuracy Guarantees

Mingxi Wu and Christopher Jermaine
University of Florida

Motivation

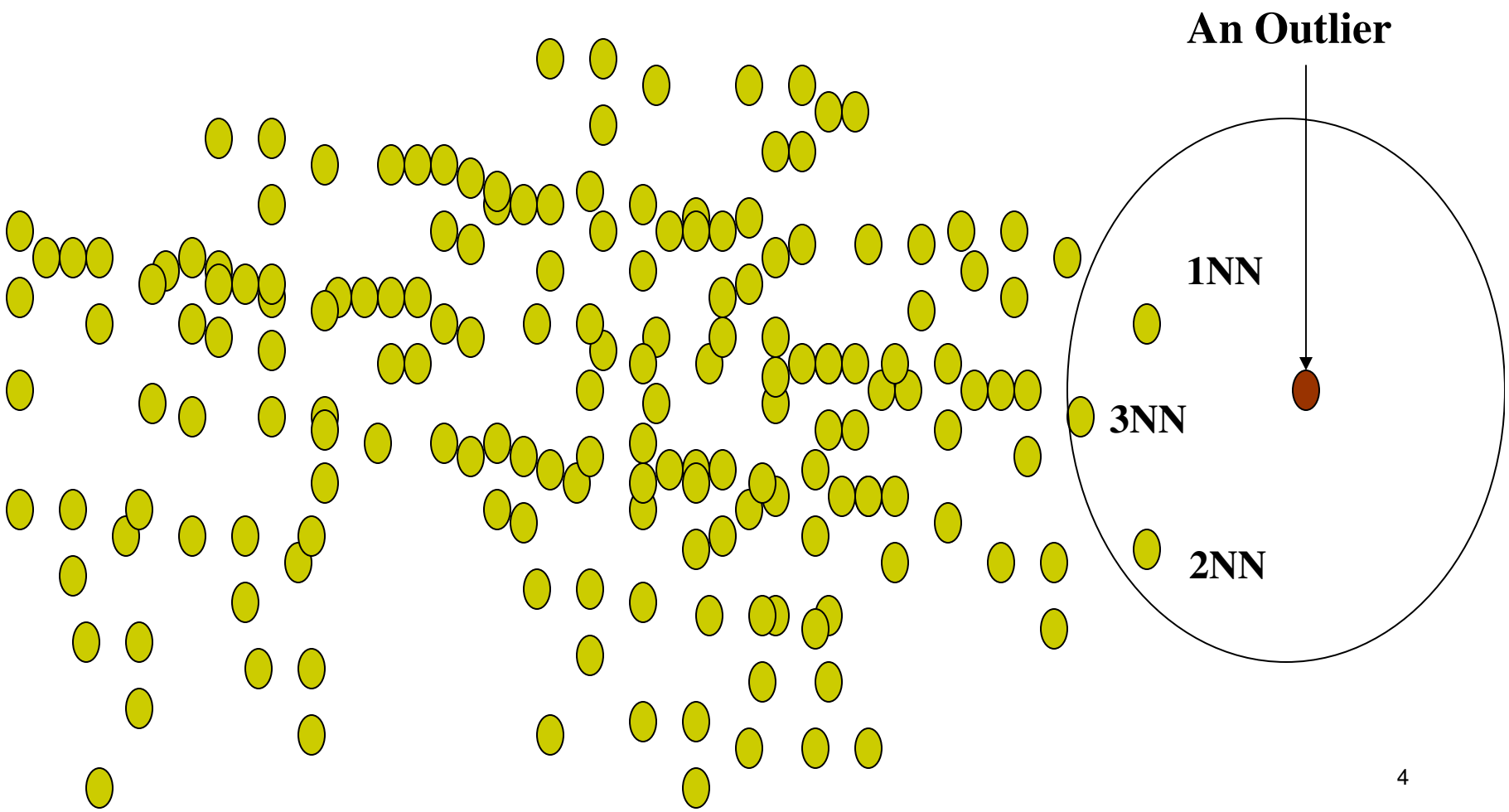
- Existing Distance-Based outlier detection requires $10^2 \sim 10^3$ distance computations per point
- Expensive distance functions are pervasive
  - Edit distance for strings
  - ERP distance for time series
  - Quadratic distance for color histograms
  - Scoring matrices for aligning biosequences
- A couple of thousand points require hours/days

Outlier Definition

- Distance-Based outliers are those points "far" from others
- Outliers are the top n examples whose distance to its kth nearest neighbor (kth-NN) is greatest

  –*Ramaswamy, Rastogi, & Shim* Sigmod 2000

Definition-cont.



Simple Sampling Algorithm (New)

1. For each point **p** in the data set
   - Randomly Sample α points for it
   - Calculate point **p**'s kth-NN distance in its sample
2. Return the top n points whose kth-NN distance in its sample is greatest

Merits: (1) Simple to implement. (2) Requires a fixed number of distance computations. (3) Applicable to any (metric or non-metric) space

Accuracy Guarantees (step 1)

- Let A be the set of true top n kth-NN outliers
- Let A' be the return set of the sampling algorithm
- Define quality measure $N = A \bigcap A'$

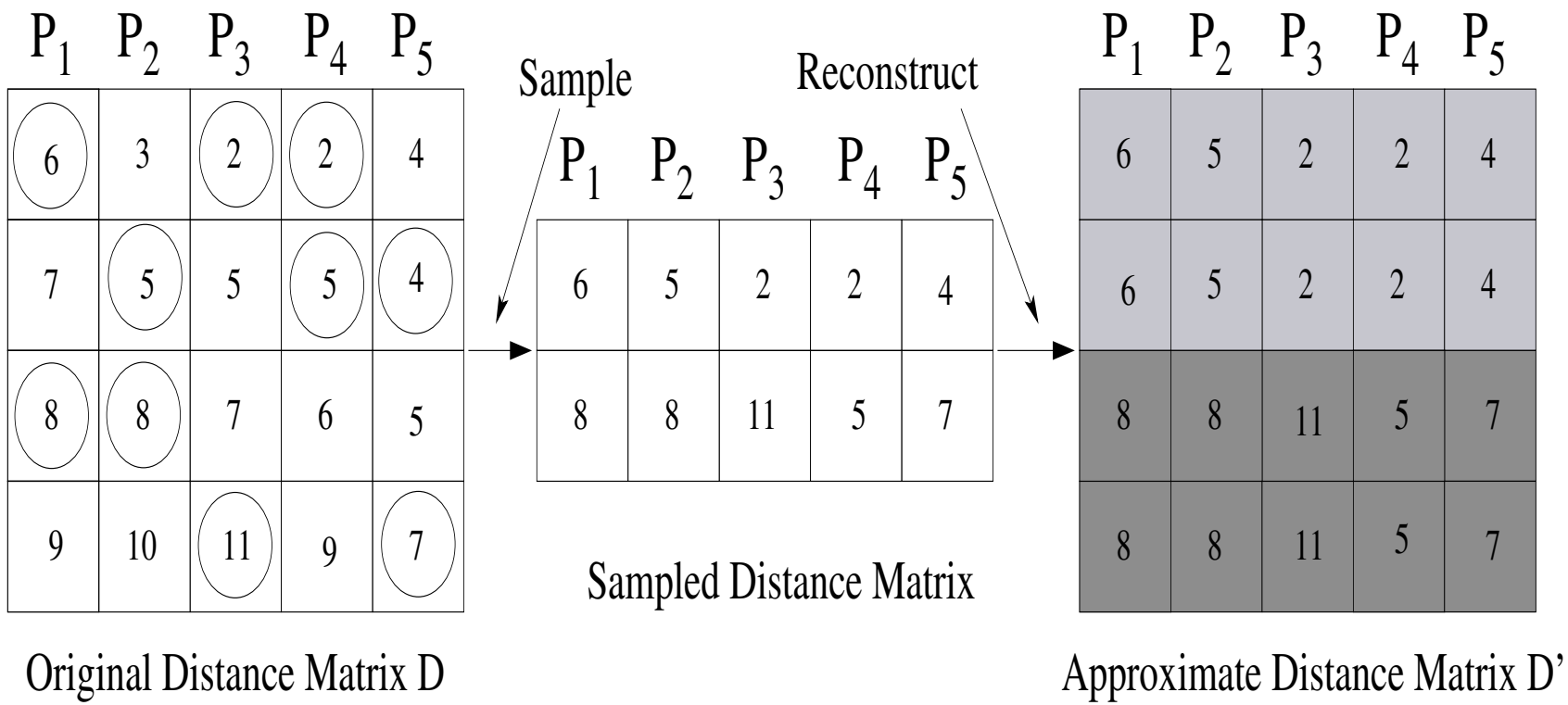
Contribution: we derived the formulas for the expectation and variance of N.

How to use step 1's theory?

- Step 1's analysis is based on the complete distance database
  - By *distance database,* we refer to a matrix storing the pairwise distance from point i to point j for every i and j.
- In reality, we want to avoid additional distance computations beyond those required by the sampling algorithm.
- Thus, we use the idea of bootstrap without replacement.

Accuracy Guarantees (step 2)

- Obtain the bootstrap distance matrix D'



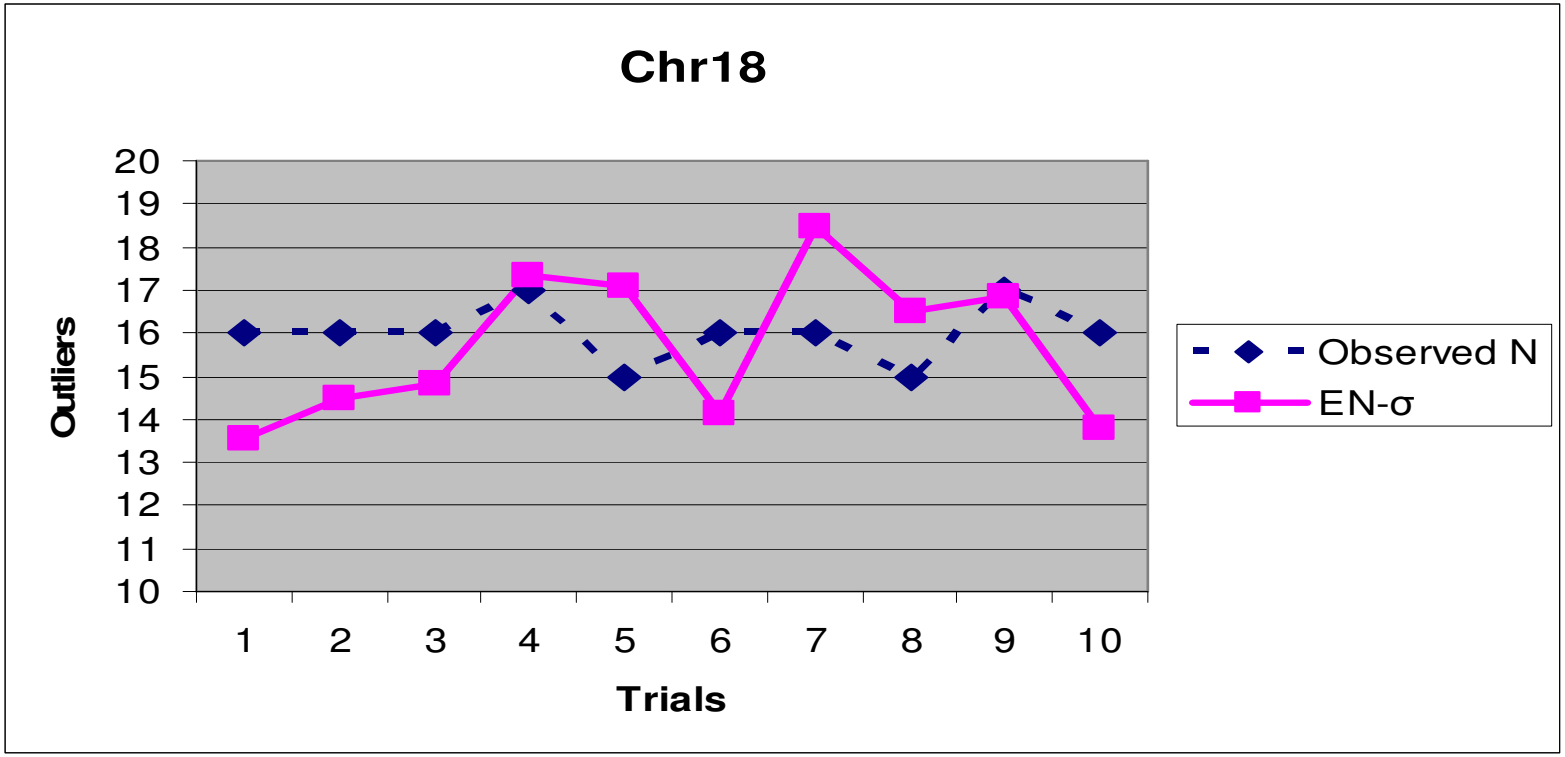
Accuracy Guarantees (step 3)

- Apply the theoretical analysis of Step 1 to the approximate distance matrix obtained in step 2
- Return the above result as the accuracy guarantees for A'

Contribution: our algorithms accomplish step 3 in sorting time of the sampled distances.

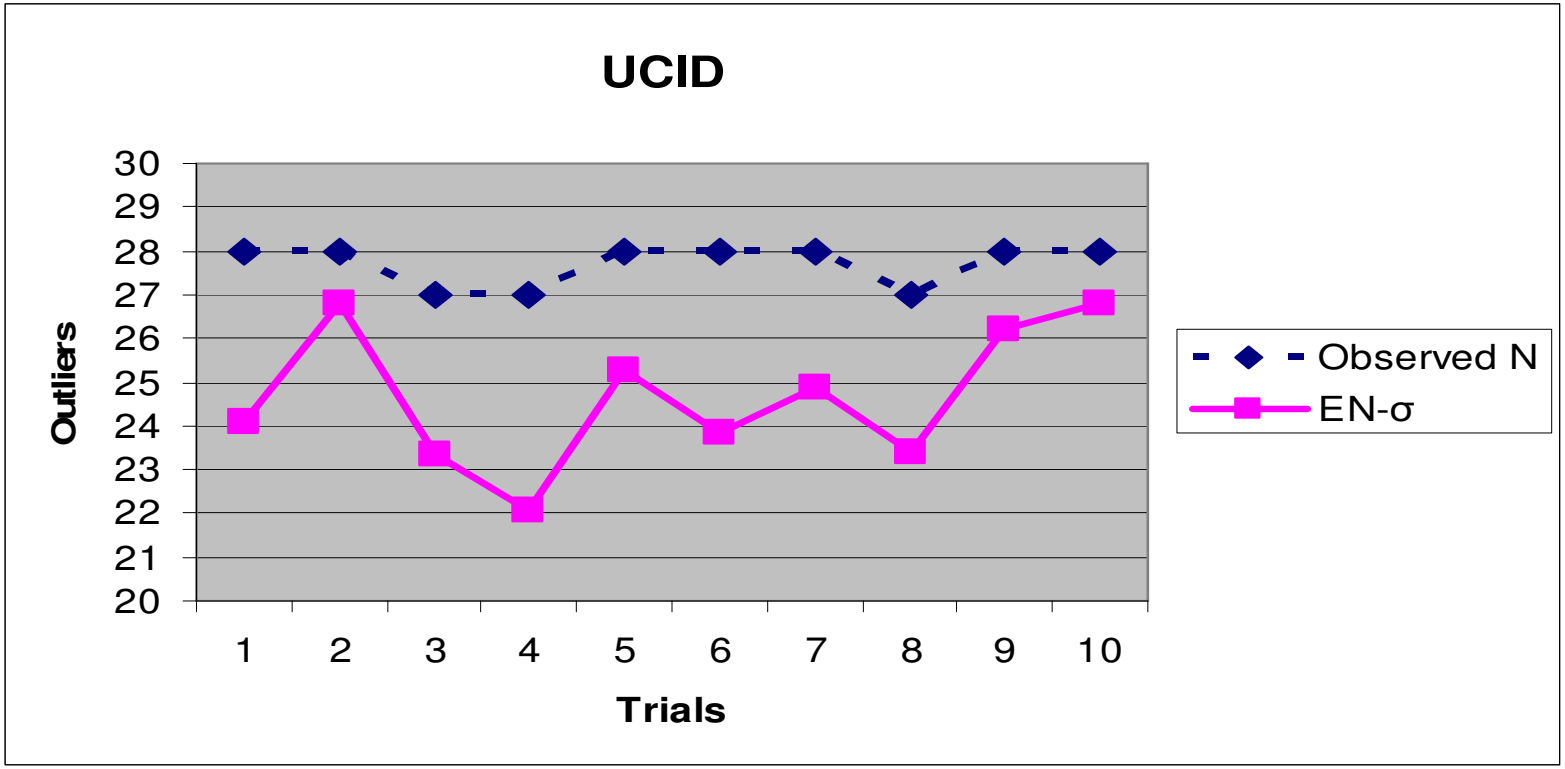
Experimental Results (Accuracy)

- Human Chromosome 18
  - 4k points, 10 samples/point, Edit distance, 5th-NN, top 30 outliers



Experimental Results (Accuracy cont.)

- UCID image database
  - 1.3k points,10 samples/point, quadratic distance, 5th-NN, top 30 outliers



Experimental Results (Efficiency)

Human Chromosome 18 data set

Algorithm	**Distance Computations/point**	**Time**
Bay's Alg. (KDD 2003)	331	24h:9m:31s
Sampling Alg.	10	47m:24s
Ratios	33.1	30.6

Experimental Results (Efficiency Cont.)

UCID image data set

Algorithm	**Distance Computations/point**	**Time**
Bay's Alg. (KDD 2003)	180	3h:48m:26s
Sampling Alg.	10	14m:07s
Ratios	18	16.2

Derivation of E[N](1)

- y_i , a constant, evaluates to one if point i is a true kth-NN outlier, zero otherwise
- M_i , a random variable, evaluates to one if point i is flagged as an outlier by the sampling alg., zero otherwise.
- Then

$$E[N] = \sum_i y_i E[M_i]$$

Derivation of E[N](2)

- T_i , a random variable, denotes the number of points rank before point i w.r.t the sampled kth-NN distance
- Given n outliers to be returned, we have:

$$\begin{aligned} E[M_i] &= \Pr[M_i = 1] \\ &= \Pr[T_i \le n-1] \end{aligned}$$

Derivation of E[N](3)

- T_i is the sum of |datasize-1| independent Bernoulli random variables
- Lyapounov Central Limit Theorem says T_i asymptotically follows a Normal distribution
- We have:

$$\Pr[T_i \le n-1] \quad = \quad \Phi\left(\frac{n-1-E[T_i]}{\sqrt{Var(T_i)}}\right)$$