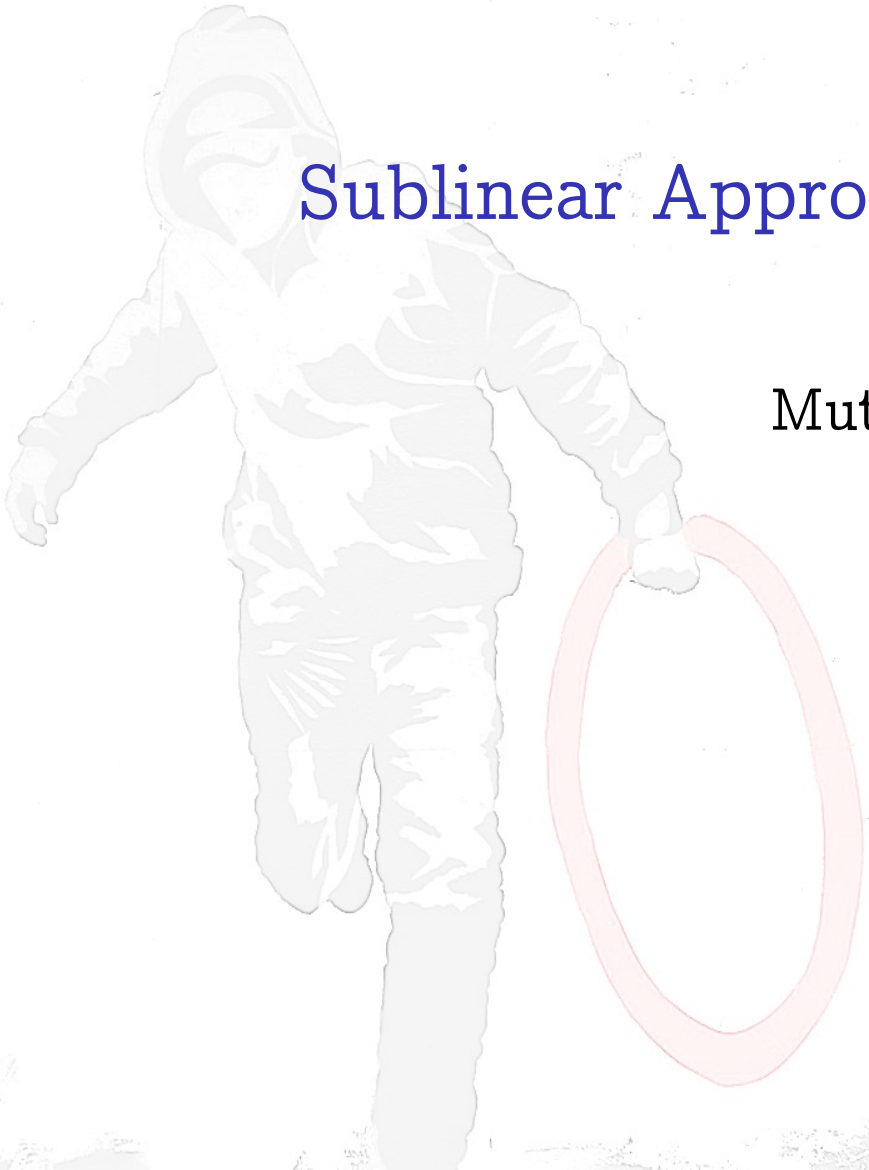
Sublinear Approach to Big Data

Muthu



Examples

- Examples of data: cellphone call logs, internet traffic, web/social/InT logs.

Cellphone traffic (cellco)	IP Traffic (ISP)	Web Traffic (Search, Ads)
TB/month weekly/monthly Reports.	TB/hour min/hours/days Detect attacks, appl.	PB/month hours/days Nearly all services.
Small team of analysts.	Small/Moderate # of researchers	Large number of engineers/analysts
File system, script language, parallel CPUs.	Optical splitters, NICs, stream mgmt engines.	1000's of m/cs, GFS,mapreduce, Bigtable, …
No publications	Alg/DB since 96. Mainly publ.	Mainly systems.

Examples

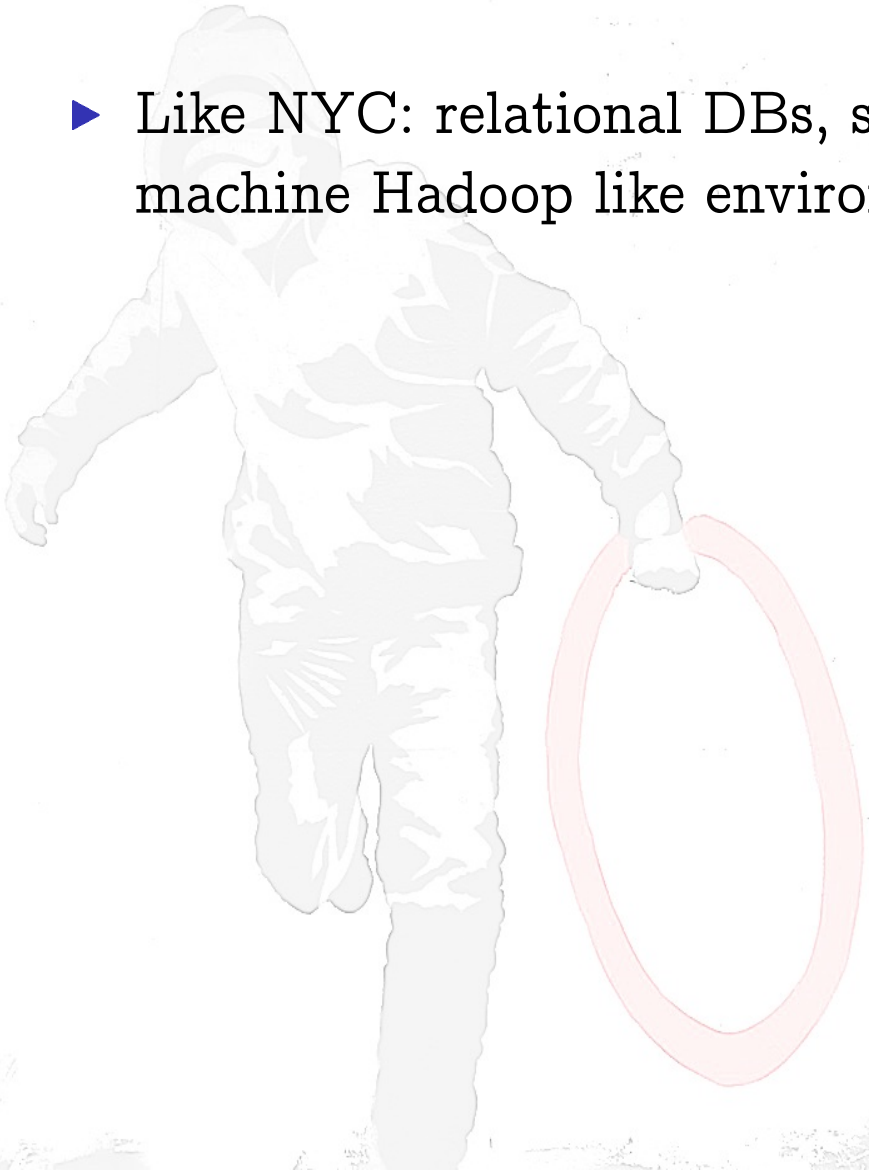
- Examples of data: cellphone call logs, internet traffic, web/social/InT logs.

Cellphone traffic (cellco)	IP Traffic (ISP)	Web Traffic (Search, Ads)
TB/month weekly/monthly Reports.	TB/hour min/hours/days Detect attacks, appl.	PB/month hours/days Nearly all services.
Small team of analysts.	Small/Moderate # of researchers	Large number of engineers/analysts
File system, script language, parallel CPUs.	Optical splitters, NICs, stream mgmt engines.	1000's of m/cs, GFS,mapreduce, Bigtable, …
No publications	Alg/DB since 96. Mainly publ.	Mainly systems.

It is not enough to stare up to the step; we must step up the stairs.
Vaclav Havel

Data in Modern Companies

- Like NYC: relational DBs, stream systems, Multiple machine Hadoop like environments.



Data in Modern Companies

- Like NYC: relational DBs, stream systems, Multiple machine Hadoop like environments.
- World got into "Big Data". From PODS13:
  - Big Data are People Too
  - Data comes from strategic agents
  - World accepts approximation and manipulation
  - Behavior. Privacy, Economics, Game theory

Data in Modern Companies

- Like NYC: relational DBs, stream systems, Multiple machine Hadoop like environments.
- World got into "Big Data". From PODS13:
  - Big Data are People Too
  - Data comes from strategic agents
  - World accepts approximation and manipulation
  - Behavior. Privacy, Economics, Game theory
- Central premise: Sublinear resources to deal with Big Data. Sublinear DB.

Sublinear Space: Indexing Streams

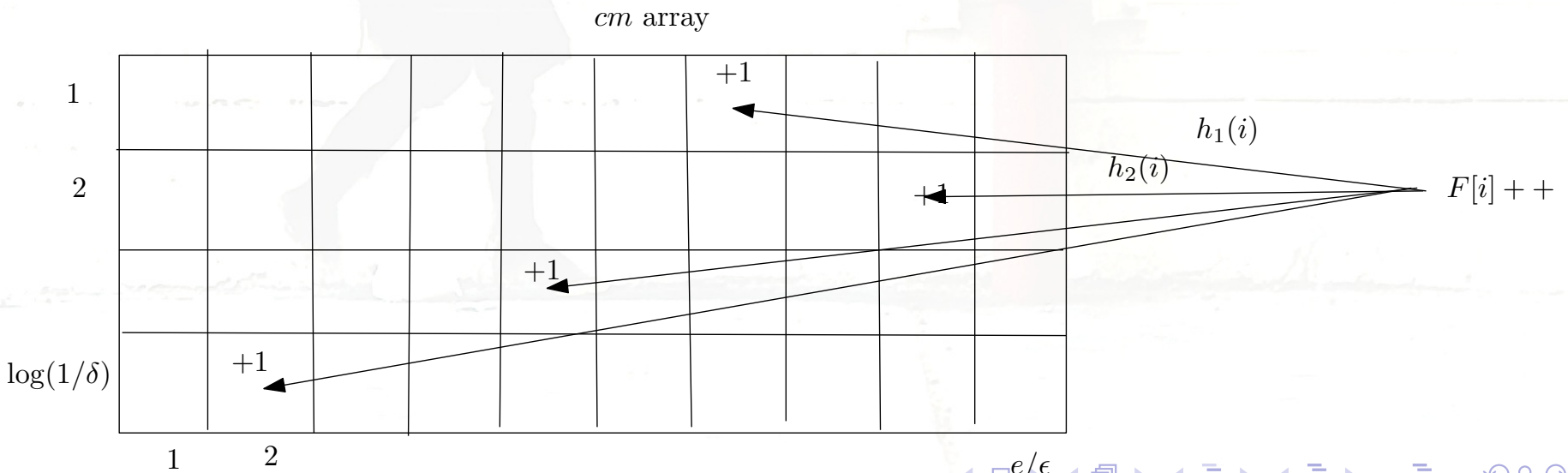
- Imagine a virtual array $F[1 \cdots n]$
- Updates: $F[i]++$, $F[i]--$
- Assume $F[i] \geq 0$ at all times
- Query: $F[i] = ?$
- Key: Use $o(n)$ space, may be $O(\log n)$ space

Count-Min Sketch

- Maintain a table of $\frac{e}{\epsilon}$ columns, and $\log \frac{1}{\delta}$ rows.
- Each row i corresponds to a hash function

$$h_i : [1, n] \rightarrow [1, \frac{e}{\epsilon}]$$

- [Update] For each update $F[i]++$,
  - for each $j = 1, \ldots, \log(1/\delta)$, update $cm[h_j(i)]++$.
- [Query] Estimate $\tilde{F}(i) = \min_{j=1,\ldots,\log(1/\delta)} cm[h_j(i)]$.



Count-Min Sketch

- Claim: With probability at least $1 - \delta$,

$$\tilde{F}[i] \leq F[i] + \varepsilon \sum_{j \neq i} F[j]$$

- Easy to see: $F[i] \leq \tilde{F}[i]$.
- Space used is $O(\frac{1}{\varepsilon} \log \frac{1}{\delta})$.
- Time per update is $O(\log \frac{1}{\delta})$. Indep of n.

G. Cormode and S. Muthukrishnan: An improved data stream summary: count-min sketch and its applications. *Journal of Algorithms, 55(1): 58-75 (2005).*

Sublinear Time: L_0 Sampling

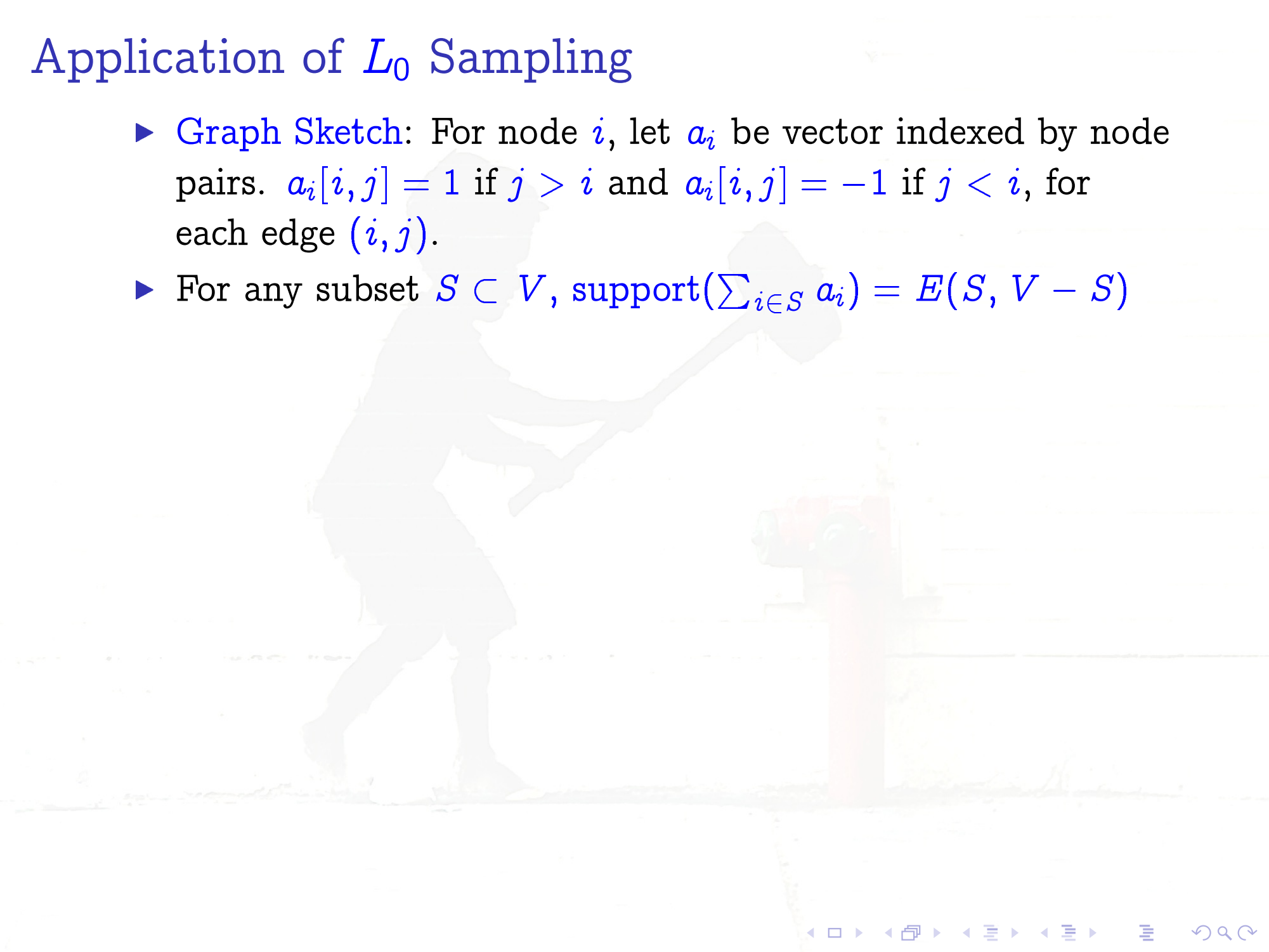
- Imagine a virtual array $F[1 \cdots n]$
- Updates: $F[i]++$, $F[i]--$
- Assume $F[i] \geq 0$ at all times

- Query: inverse sample?
  Return $i, F[i] \neq 0$ with prob $\frac{1}{|\{i|F[i] \geq 0\}|}$
- Key: Use $o(n)$ space, may be $O(\log n)$ space

Cormode, S. Muthukrishnan, Rozenbaum: Summarizing and Mining Inverse Distributions on Data Streams. VLDB05

Application of L_0 Sampling

- Graph Sketch: For node i, let a_i be vector indexed by node pairs. $a_i[i,j] = 1$ if $j > i$ and $a_i[i,j] = -1$ if $j < i$, for each edge (i,j).
- For any subset $S \subset V$, $\text{support}(\sum_{i \in S} a_i) = E(S, V - S)$



Application of L_0 Sampling

- Graph Sketch: For node i, let a_i be vector indexed by node pairs. $a_i[i,j] = 1$ if $j > i$ and $a_i[i,j] = -1$ if $j < i$, for each edge (i,j).
- For any subset $S \subset V$, $\text{support}(\sum_{i \in S} a_i) = E(S, V - S)$
- Prob: Is G connected?
  - Algorithm (Spanning Forest):
    - For each node, select an incident edge
    - Contract selected edges. Repeat until no edges
  - Data structure: L_0 sketch C for each a_j.
  - Use Ca_j to get incident edge. Then, run algorithm above. Observe:

$$\sum_{j \in S} Ca_j = C(\sum_{j \in S} a_j) \rightarrow e \in \text{support}(\sum_{j \in S} E(S, V - S))$$

Ahn, Guha, McGregor: Analyzing graph structure via linear measurements. SODA12.

Sublinear Communication: Distributed Learning

- Adversarial distribution $D_1, \ldots, D_m$ of data to m machines. Minimize communication.
- Distributed learning:
  - Alice has data D_A and Bob has D_B, and learn linear classifier g; h^* is optimal.
  - $E_D(h)$ is the number of points misclassified by h on D.
  - g has ε- error if $E_D(g) - E_D(h^*) \leq \varepsilon |D|$.
  - There is a $O(\log 1/\varepsilon)$ round two way communication protocol with $O(1)$ bits per round and ε-error.

Duame, Phillips, Saha and Venkat... Protocols for learning classifiers on distributed data. AISTATS12.

Sublinear Privacy: Pan-Privacy

- Differential Privacy. Let $A(x)$ approximate $f(x)$. $A(x)$ is DP if for all outputs t and for all neighboring data x and y, $Pr(A(x) = t) \leq e^{\varepsilon} Pr(A(y) = t)$

Sublinear Privacy: Pan-Privacy

- Differential Privacy. Let $A(x)$ approximate $f(x)$. $A(x)$ is DP if for all outputs t and for all neighboring data x and y, $Pr(A(x) = t) \leq e^{\varepsilon} Pr(A(y) = t)$
- What if internal state is breached by the adversary?
- Pan-Privacy. For every two neighboring streams, at any time, internal state and final output should be DP.
- Use sketches to get approximate pan-private algorithms.

Pan-private algorithms via statistics on sketches.
D. Mir, S. Muthukrishnan, A. Nikolov and R. Wright, PODS11.

Sublinear Machines: MapReduce Algorithms

- Basic:
  - For an input of size n
  - Sublinear number of machines $O(n^{1-\epsilon})$
  - Each with sublinear amount of memory $O(n^{1-\epsilon})$.
  - Allowed to take time polynomial time to process each map or reduce task
- Can compute Connectivity, MST in 2 rounds, provided $|E| = |V|^{1+c}$. Bunch of results of similar ilk.

MRC model [Karloff, Suri, Vassilvitskii 10], Nimble Algorithms [Kannan, Vempala]

Summary

- Sublinear machines, space, key size, rounds, running time algorithms for Big Data problems
- Recent progress on graph, social ranking problems.
- Novel perspectives like behavioral aspects of DBs have have to be understood, beyond classical issues like suitable query languages.