Estimating Dominance Norms of Multiple Data Streams

Graham Cormode
graham@dimacs.rutgers.edu

Joint work with S. Muthukrishnan

Data Stream Phenomenon

- Data is being produced faster than our ability to process it
- Leads to the data stream paradigm: process the data as it arrives, don't store or communicate the full data
- Motivated by networks (Gb per hour per router), also applied to databases, scientific data feeds, sensor networks and so on
- Theoretically leads to search for one pass, online algorithms with poly-log space and time per item in the stream

Multiple Signals

Previous work considers only a single signal at a time

Many data streams consist of multiple signals from several distributions, from which we want to extract some global information

Examples:

- financial transactions from many different individuals
- web clickstreams from many users registered on different machines
- multiple readings from multiple sensors in atmospheric monitoring

Prior Work

- Growing body of work on data stream processing in algorithms, database and network fields
- Many computations possible on streams – notably, finding frequency moments, Lp norms, quantiles, wavelet representation and so on
- Babcock Babu Datar Motwani Widom 02, Garofalakis, Gehrke, Rastogi 02, Muthukrishnan 03 give surveys from different perspectives
- But almost exclusively focus is on single massive streams, not many massive streams!

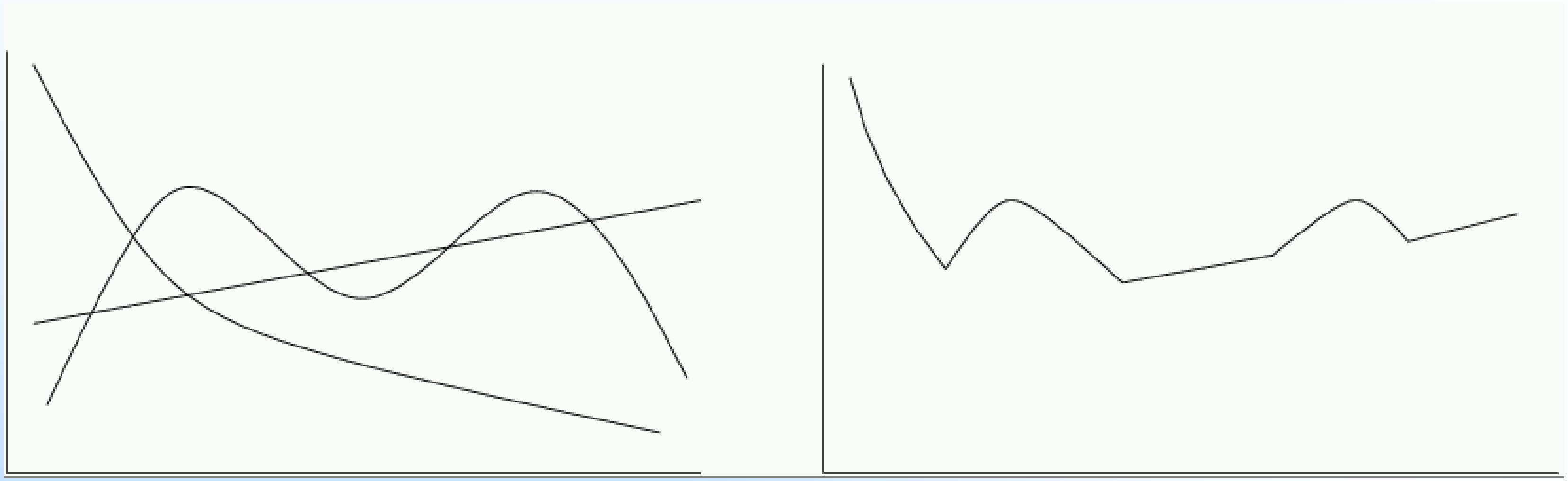
Data Stream Model

- Model data streams as simply structured series of items
- n items in the stream S=(i, a[i,j]) means a[i,j] is the value of distribution j at location i
- Assume: a[i,j] is bounded by polynomial in n
- Don’t assume that j is made explicit in stream or that we see updates for every [i,j] pair

Dominance Norm

- The dominance norm measures the “worst case influence” of the different signals
- Defined as $Dom(S) = \Sigma_i \max_j \{a[i,j]\}$
- Can think of this as the L_1 norm of the upper-envelope of the signals,
- Alternatively, as a function of the marginals of a matrix of the signal values

Dominance Norm



- Maximum possible utilization of a resource
- Applied in financial applications, electrical grid
- Treat as an indicator of actionable events

Dominance Norm

- Suppose each a[i,j] is 0 or 1
- Consider each signal to be a set X_j, then

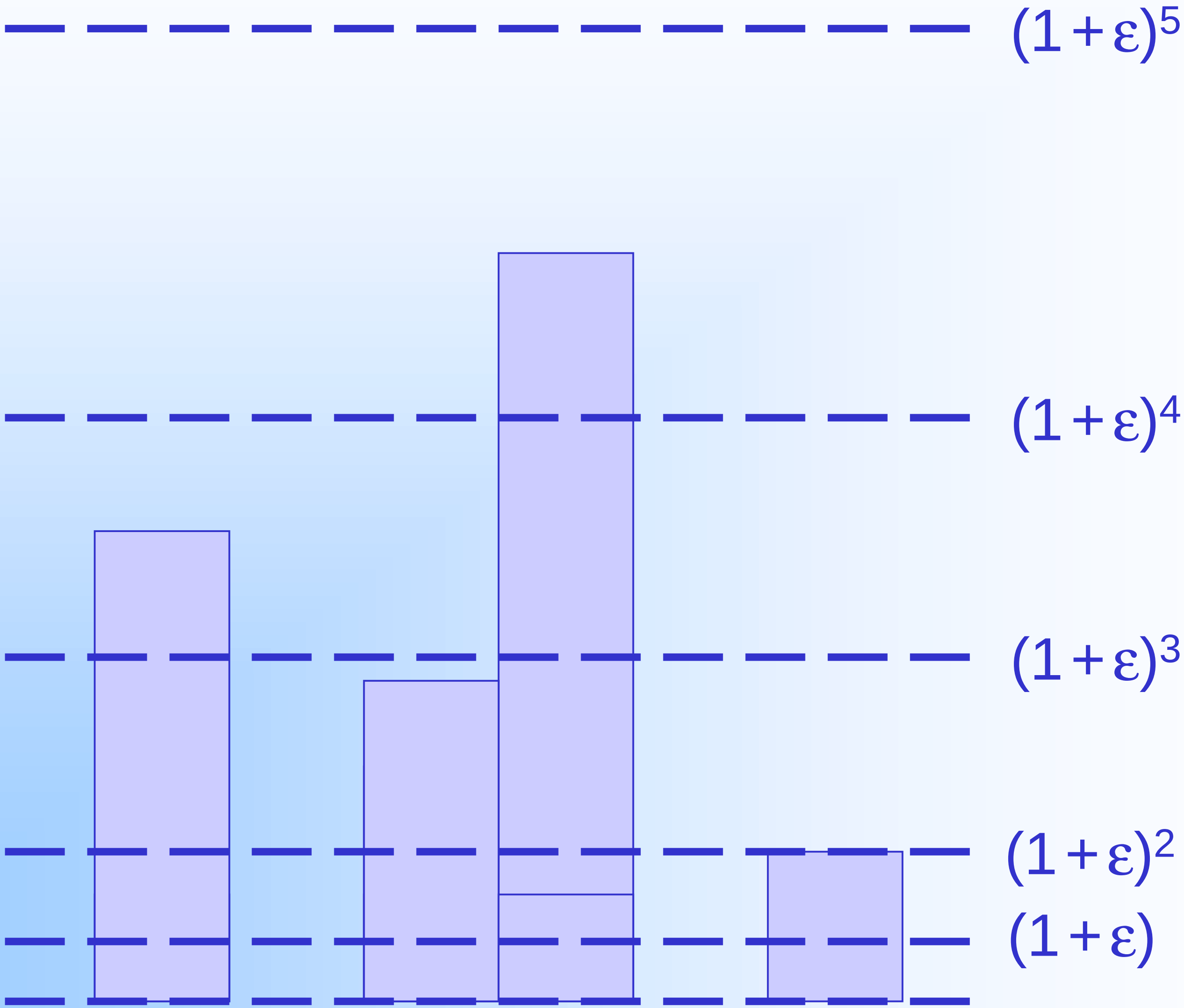
$$\mathrm{Dom}(S) = |\cup_j X_j|$$

This can be solved using existing stream algorithms for finding unions of multiple sets

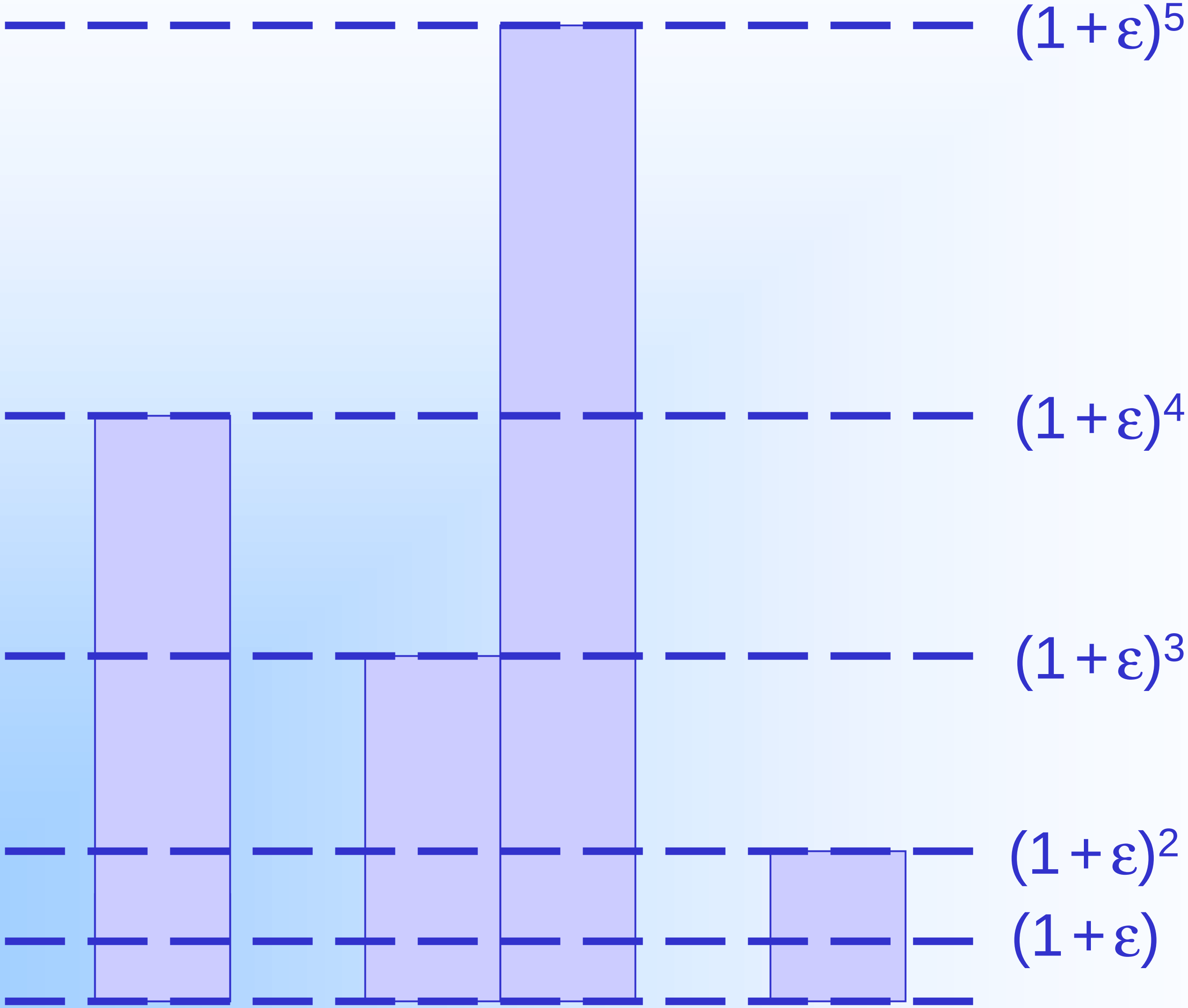
Can also be thought of as counting the number of distinct items i in the stream

Can this be generalized for arbitrary a[i,j]?

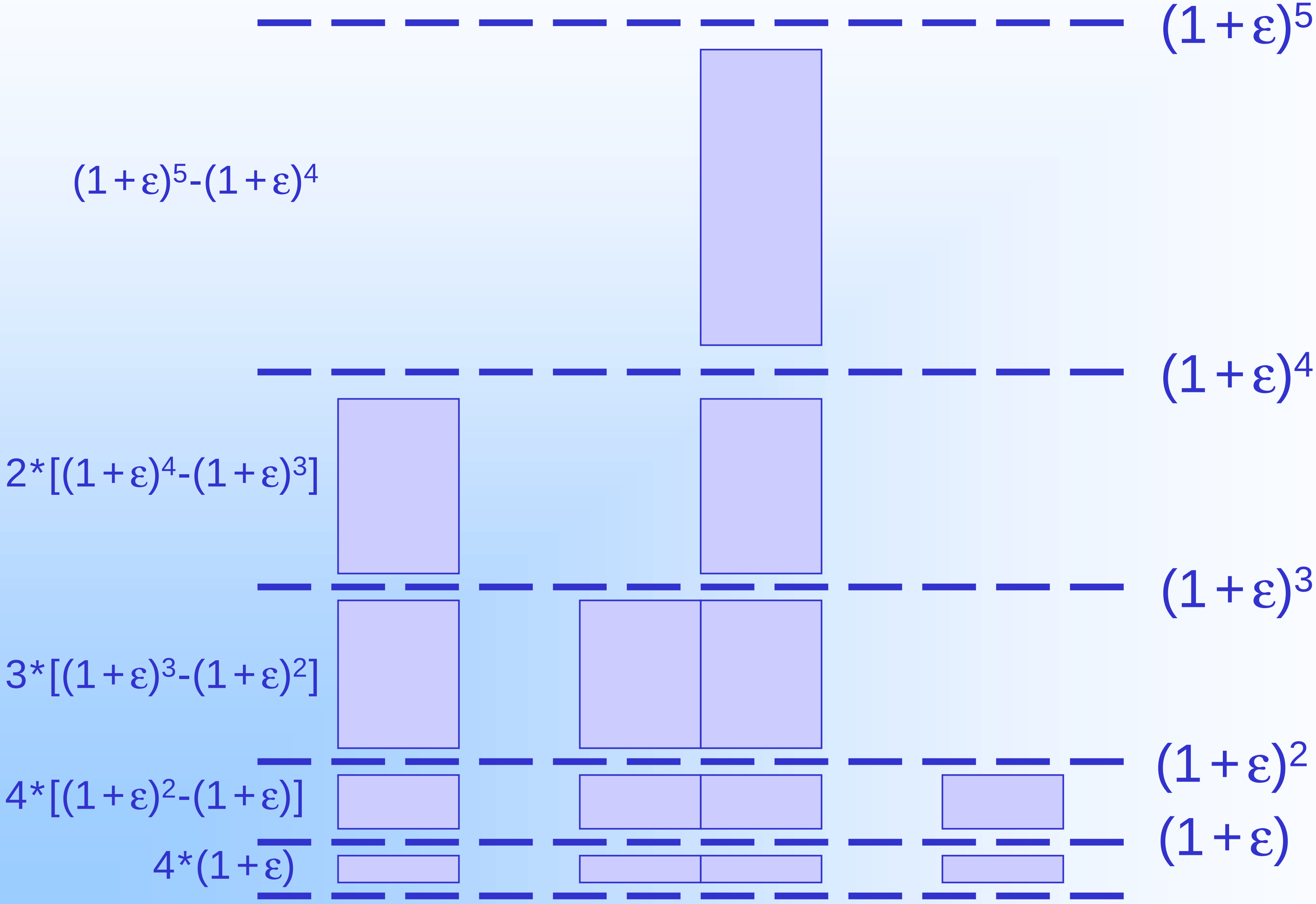
Approximation



Approximation



Approximation



Space Cost

- $\log_{1+\varepsilon}$ (max val / min val) distinct element algorithm instances = $O(\log(n) / \varepsilon)$
- Space required is $O(\text{poly-log}(n) / \varepsilon^2)$ per instance using prior work
- Total space is $O(\text{poly-log}(n)/\varepsilon^3)$
- Cubic space dependency on $1/\varepsilon$ is high – can we do better?

Reducing Space

- Try to keep just 1 distinct element count algorithm, and so reduce space cost
- Need a more flexible algorithm and new analysis
- Make a new use of Stable Distributions, used before in stream processing
- See Indyk'00, CIKM'02, CDIM'03

Idealized Algorithm

Suppose there were a distribution X such that $E(cX) = 1$ (an impossible property

- Let $x_{i,k}$ be values drawn from X.
- Set $z = 0$ initially
- For every $(i,a[i,j])$ in the stream,

$$z = z + \Sigma_{k=1}^{a[i,j]} x_{i,k}$$

- Then $E(z) = \Sigma_i \max_j \{a[i,j]\}$, and can be used to estimate $Dom(S)$

Reduction to Norms

Fix the idealized algorithm and make it practical.

Replace impossible dbn X with stable distributions by turning problem into one of norm approximation.

Let b be the matrix with $b[i,k] = |\{j \mid k \leq a[i,j]\}|$

- Define $\|b\|_p^p = \Sigma_{i,k} b^p$
- $\mathrm{Dom}(S) = |\{i,k \mid b[i,k] > 0\}| = \|b\|_0^0$

Approximate the value of $\|b\|_0^0$ with $\|b\|_p^p$ for suitably chosen small value of p.

Choosing the p-value

Absolute value of any entry in the matrix $< n$

$$\|\mathbf{b}\|_0 = \Sigma\, |b_i|^0 \leq \Sigma\, |b_i|^p \leq \Sigma\, B^p\, |b_i|^0 \leq n^p\, \|\mathbf{b}\|_0$$

Setting $n^p = (1+\varepsilon)$ means

$$\|\mathbf{b}\|_0 \leq \|b_i\|_p^{\,p} \leq (1+\varepsilon)\, \|\mathbf{b}\|_0$$

So setting $p = \varepsilon / \log n$, allows approximation of L_0 by L_p – reducing p zeros in on L_0

Stable Distributions

Use *stable distributions* to approximate $||b||_p^p$

Stable distributions have property that

$$a_1X_1 + a_2X_2 + \ldots a_nX_n \overset{\text{in dbn.}}{=} ||(a_1, a_2, \ldots , a_n)||_p X$$

if $X_1 \ldots X_n$ are stable with stability parameter p

Stable distributions exist and can be simulated for all parameters $0 < p \leq 2$.

Approximation Algorithm

- Let $x_{i,k}$ be values drawn from Stable Distribution with parameter $p = \varepsilon/\log n$.
- Set $z = 0$ initially
- For every $(i,a[i,j])$ in the stream,

$$z = z + \Sigma_{k=1}^{a[i,j]} x_{i,k}$$

- Repeat independently in parallel $O(1/\varepsilon^2 \log 1/\delta)$ times, take the median of $|z|$s as the answer

Approximation Result

- Each z distributed as $||b||_p X$
- $\text{median}(|z|^p) = \text{median}(||b||_p^p |X|^p)$
  $= ||b||_p^p \, \text{median}(|X|^p)$

Result (with rescaling of ε):

With probability at least $1-\delta$,

$$(1-\varepsilon)\mathrm{Dom}(S) \le \frac{\text{median}(|z|^p)}{\text{median}(|X|^p)} \le (1+\varepsilon)\mathrm{Dom}(S)$$

Issues to Resolve

- What is the scale factor, $\text{median}(|X|^p)$?
- How to compute efficiently (faster than $O(a[i,j])$ per update?
- How to avoid storing $x_{i,k}$ explicitly?
  - Use appropriate pseudo-random number generator to find $x_{i,k}$ when needed
  - use standard transforms to draw from stable distributions via uniform distribution

Scale Factor

- Use result from stats: in the limit as $p \to 0$, $|X|^p$ is distributed as E^{-1}, inverse exponential distribution
- Cumulative density function of E^{-1}

$$F(x) = \exp(-1/x)$$

- Median: $F(x) = \frac{1}{2} = \exp(-1/\text{median}(|X|^0)$
- So $\text{median}(|X|^0) = 1/\ln 2$

Efficient Computation

- Direct implementation means adding a[i,j] values to the counters for every update
- But, each value is drawn from a stable distribution, and we know sum of stables is a stable
- Use same trick as before, round to nearest power of $(1+\varepsilon)$ and just add the $O(\log(n)/\varepsilon)$ values to the counters
- So update time is $O(\log(n)/\varepsilon^3)$

Full results

- Approximate the Dominance norm within $1 \pm \varepsilon$ with probability at least $1-\delta$ using $O(1/\varepsilon^2 \log (1/\delta))$ counters

- Time per update is $O(1/\varepsilon^3 \log (1/\delta))$

- Possible to 'subtract off' the effect of earlier insertions – not possible with most distinct element algorithms

- A few other aspects not mentioned, full details in the paper

Other Dominances

- Natural questions: are other notions of dominance on multiple streams tractable?
- Take Min-Dominance:

$$\mathrm{MinDom}(S) = \Sigma_i \min_j \{a[i,j]\}$$

- Let X_1, X_2 be subsets of $\{1\ldots n/2\}$.
  Set $a[i,j]=1 \Leftrightarrow i \in X_j$
- Then $\mathrm{MinDom}(S) = |X_1 \cap X_2|$
- Requires $\Omega(n)$ space to approximate, even allowing probability, several passes etc.

Extensions

- Other reasonable definitions of dominances – eg Median Dominance, Relative Dominance between two streams, also require linear space
- Are there other natural quantities which are computable over streams of multiple signals?
- What quantities are good indicators for actionable events?