Streaming algorithms, Apache DataSketches, and new results on corsets


Lee Rhodes
Distinguished Architect
Oath, Inc.


Kevin Lang
Principal Scientist
Oath, Inc


Edo Liberty
Founder
HyperCube


Alaxander Saydakov
Senior Software Engineer
Oath, Inc

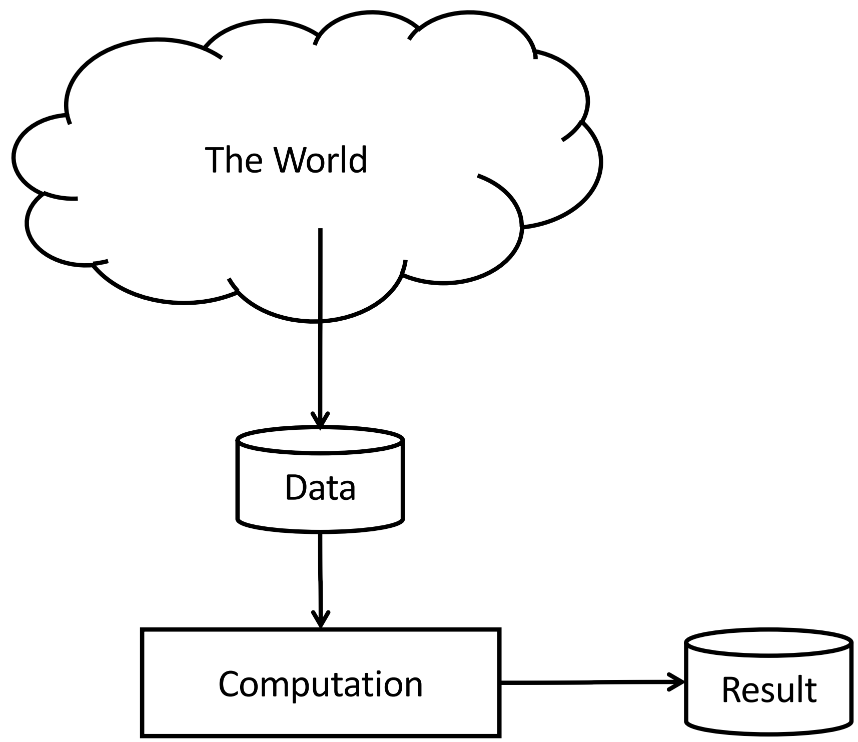

Jon Malkin
Senior Scientist
Oath, Inc


Justin Thaler
Assistant Professor
Georgetown University.

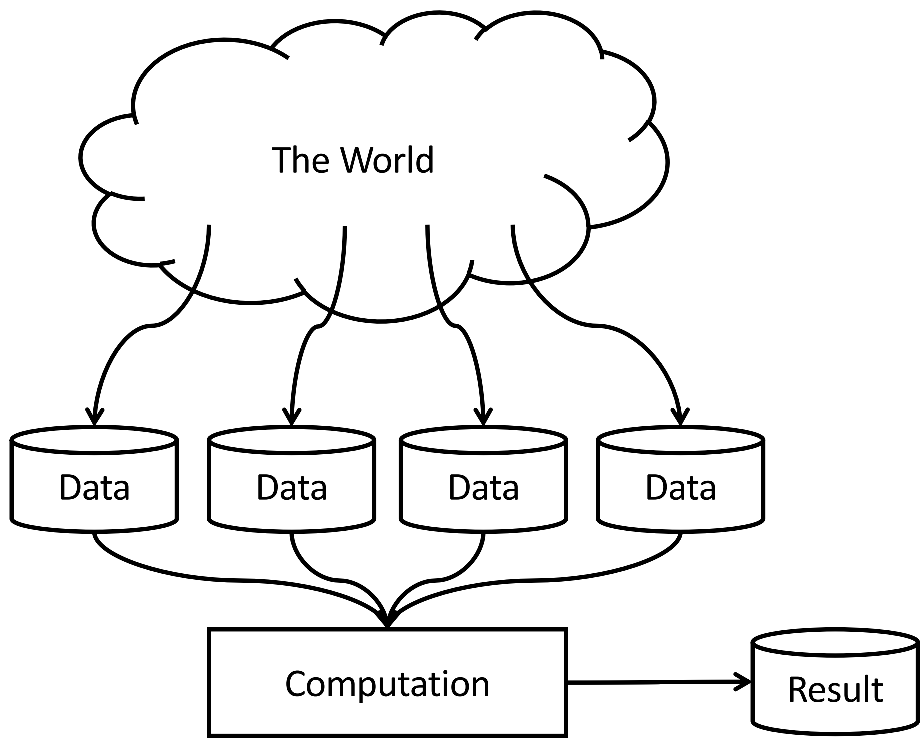

Zohar Karnin
Principal Scientist
Amazon


Nikita Ivkin
Applied Scientist
Amazon

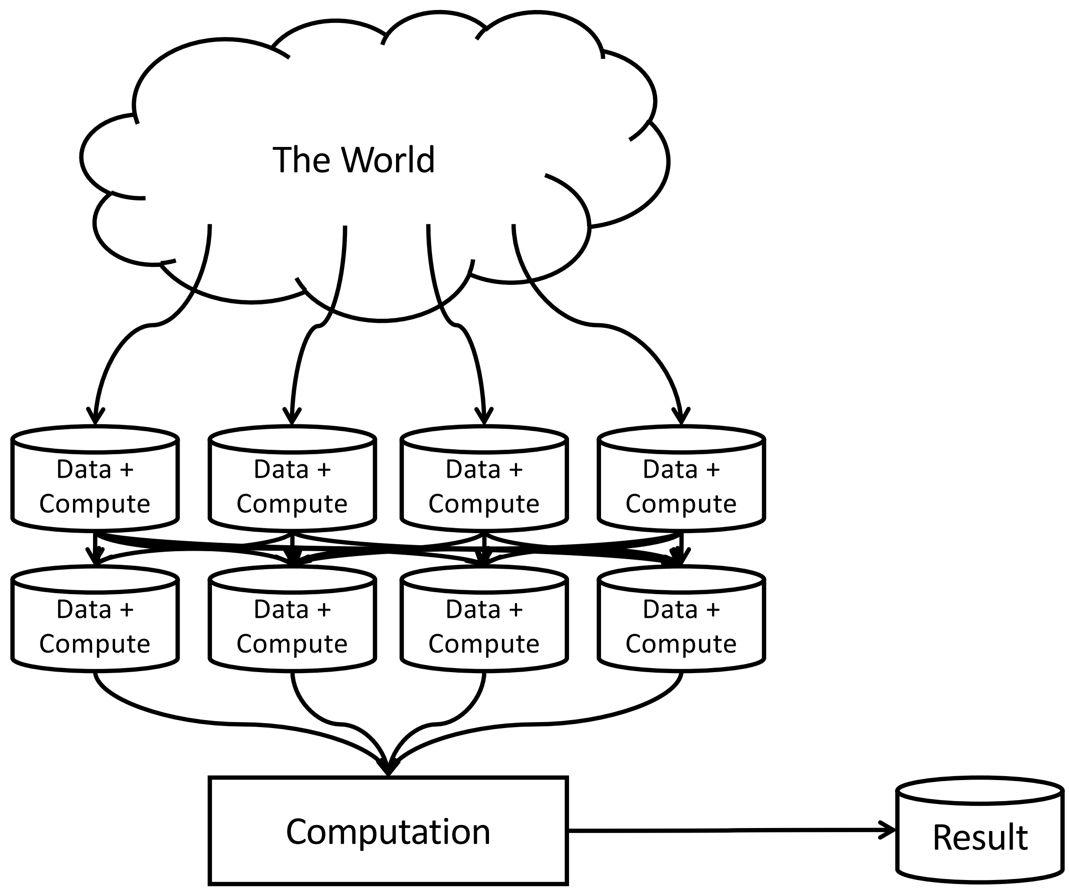
Single machine data processing



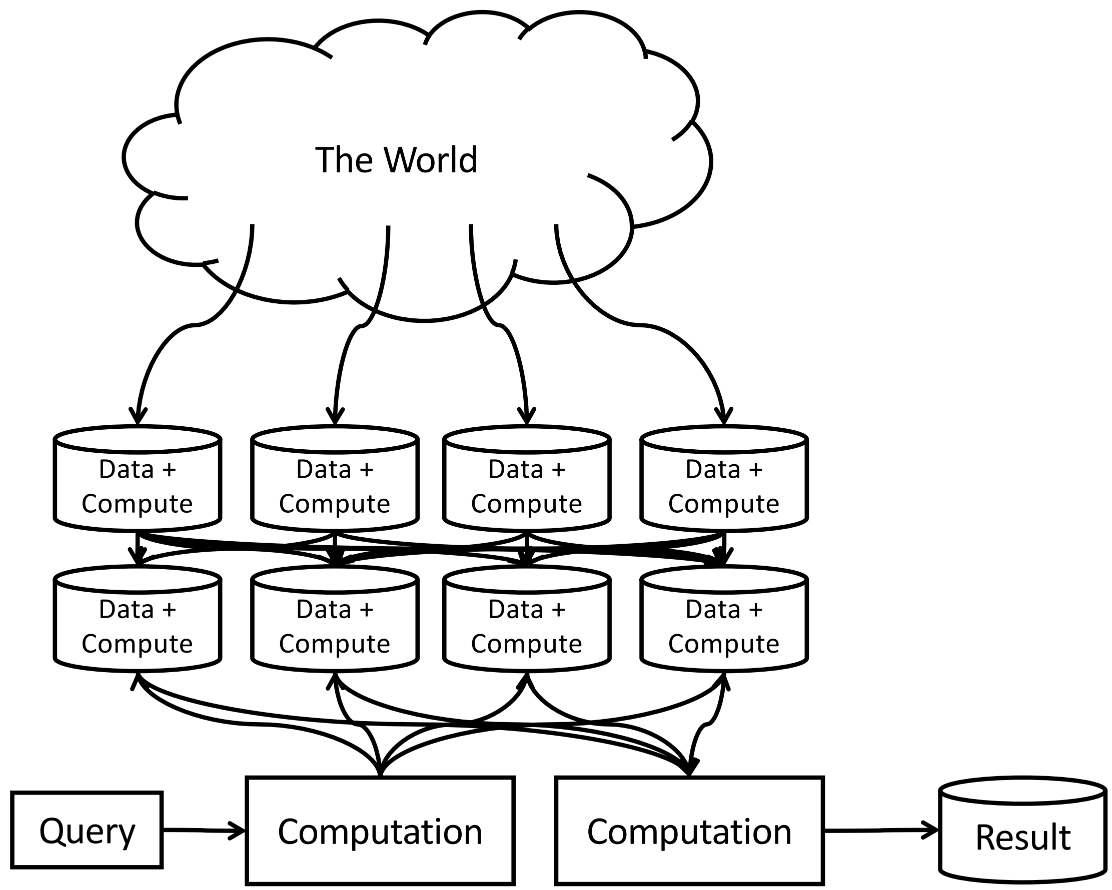
Distributed storage



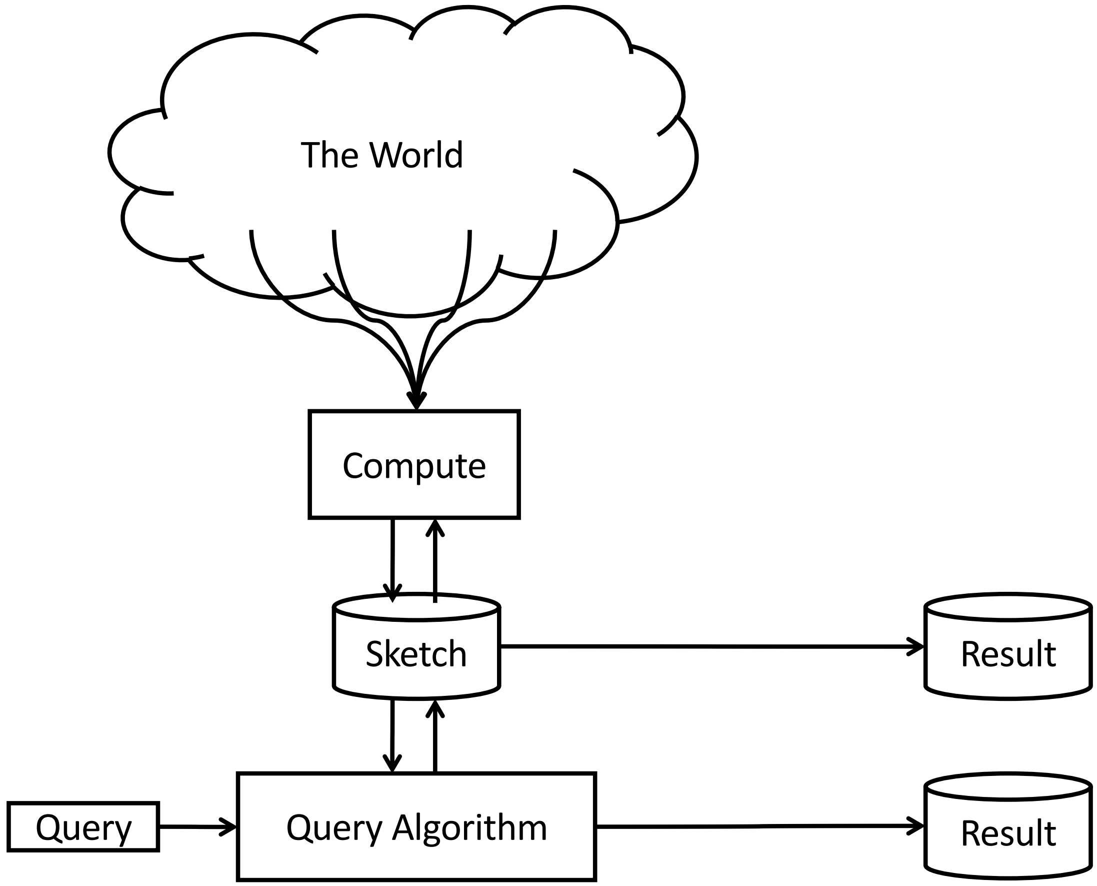
Distributed compute (map/reduce, MPI, …)



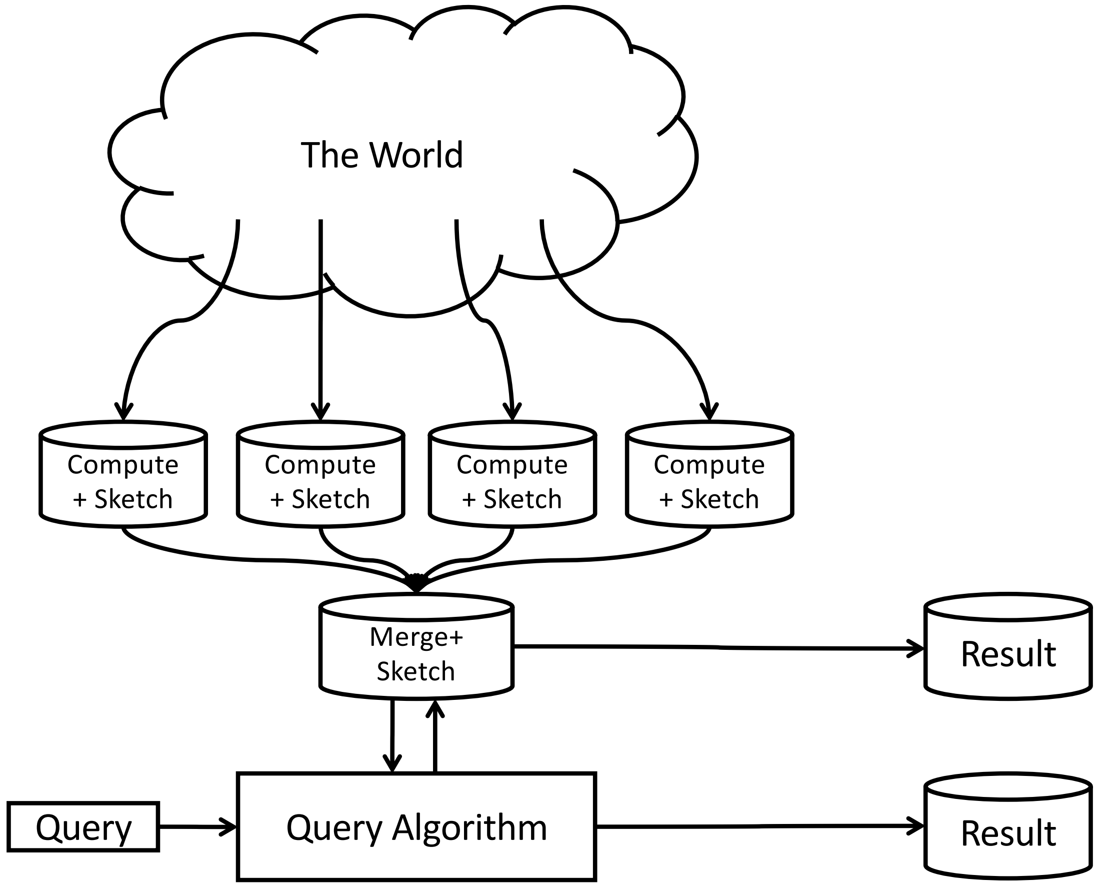
Distributed model (indexes, databases, Spark…)



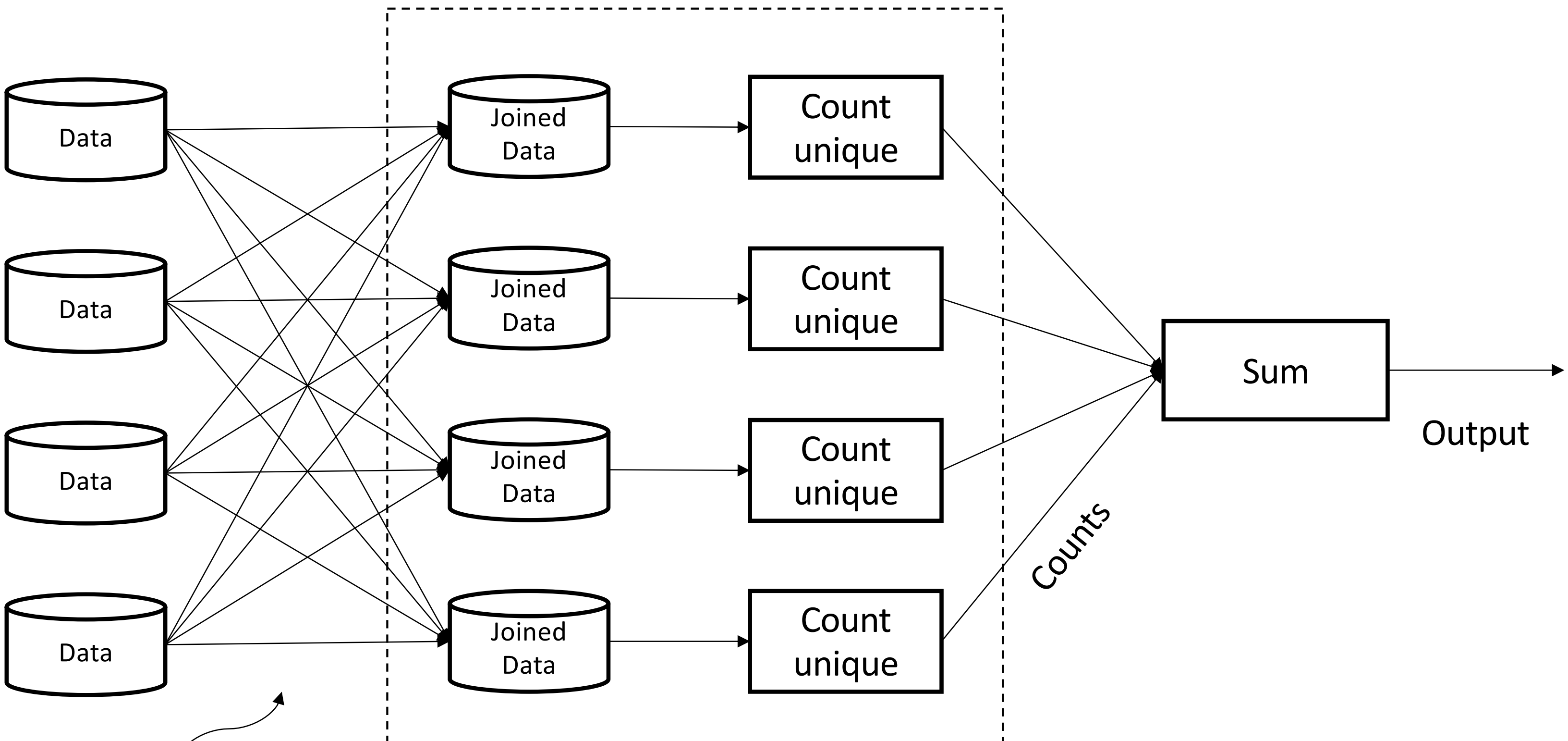
The streaming model



Mergeable Summaries

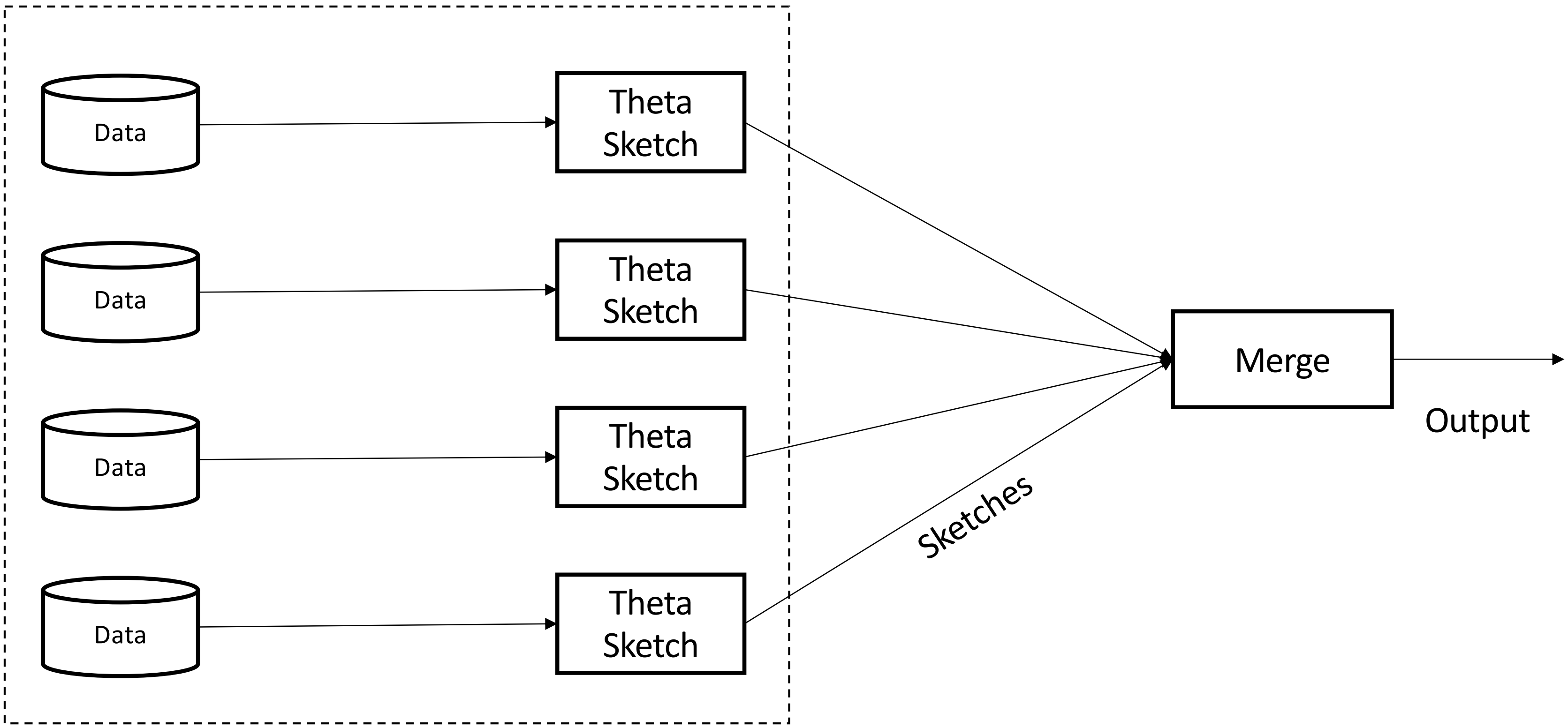


Unique Counting with Map Reduce



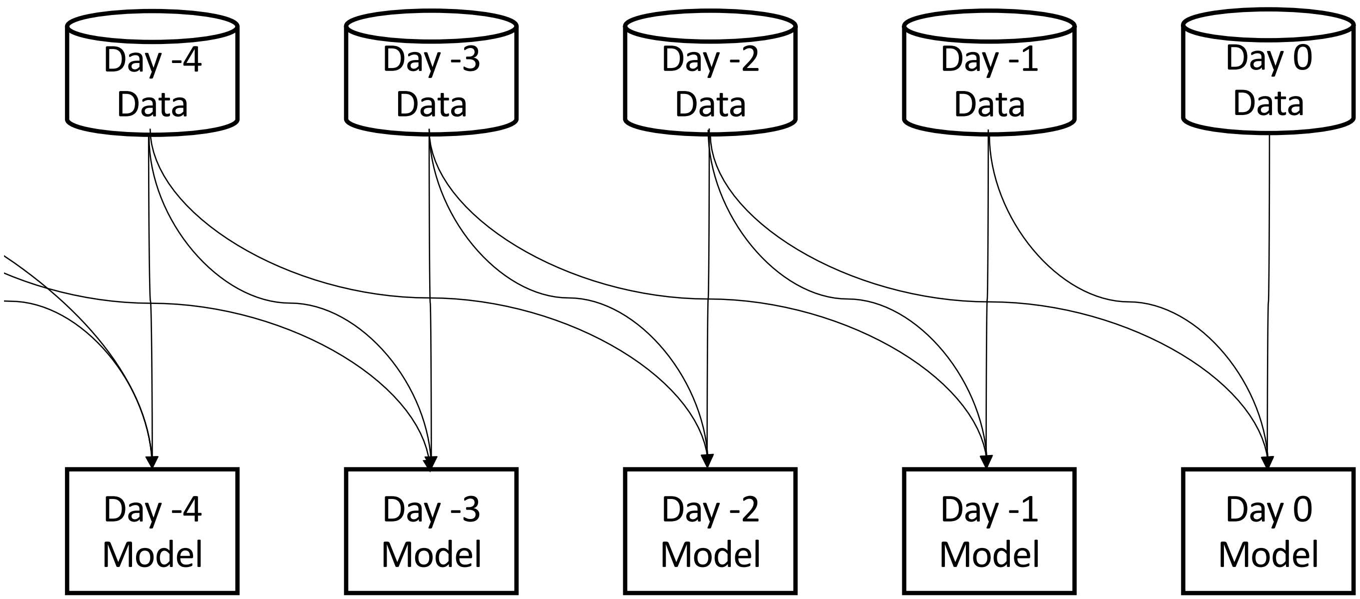
This is a shuffle operation which is very compute and network heavy!

Unique Counting with Mergeable Summaries



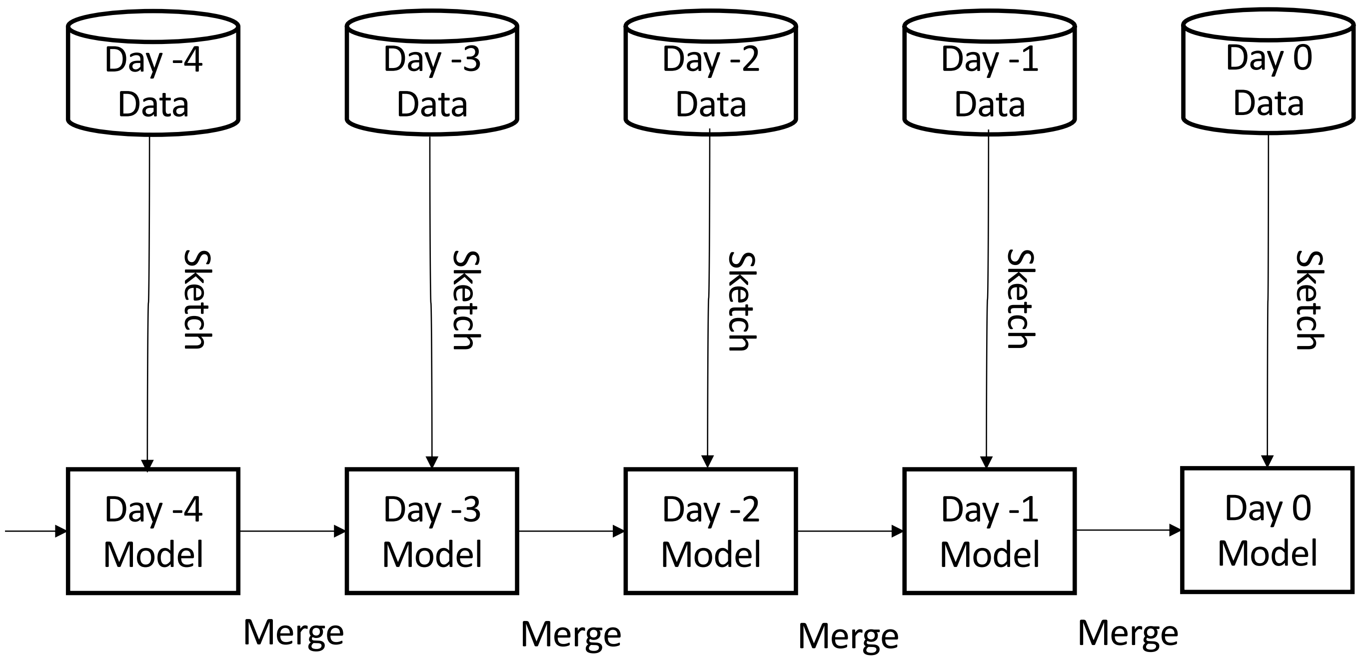
No shuffle operation is needed!

Data Mining with Traditional Windowing



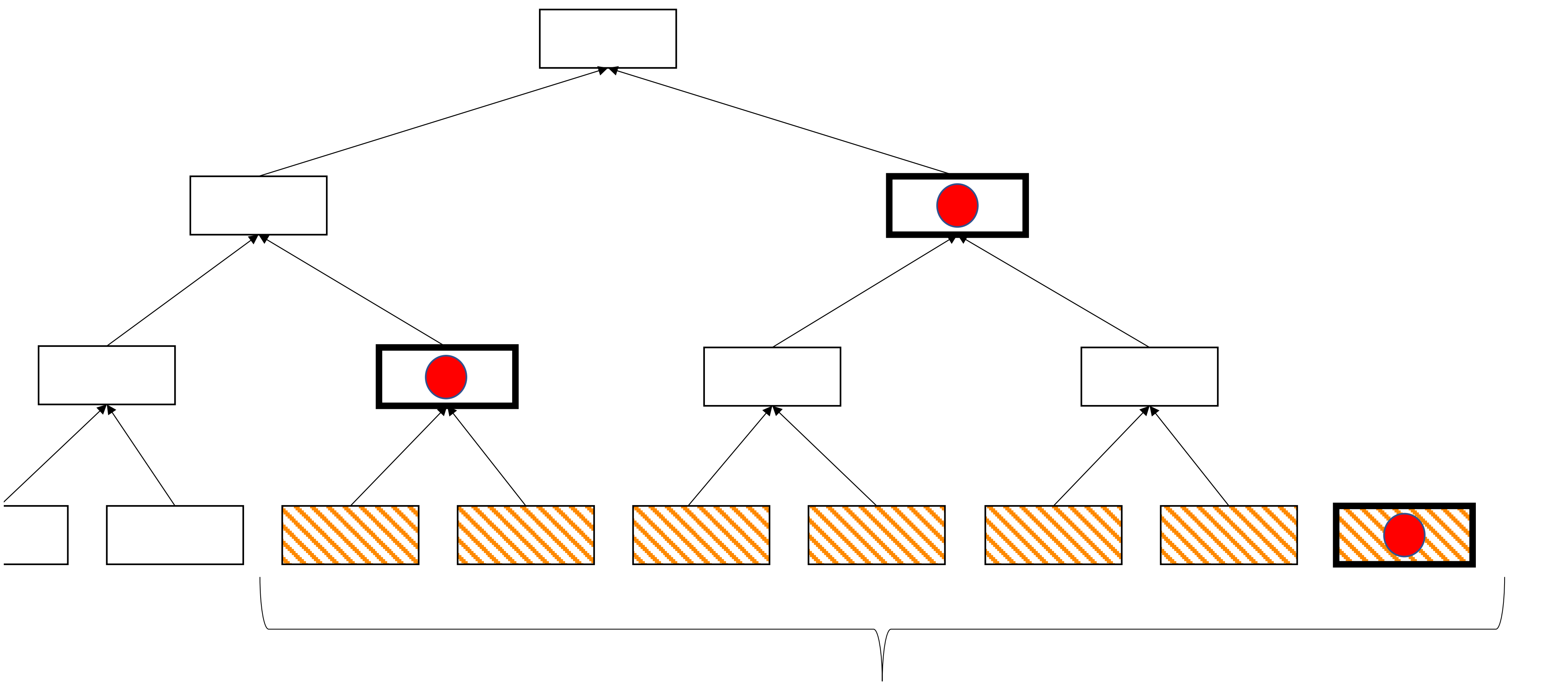
Every dataset is processed 3 times for a model consisting of 3 days

Data Mining with Mergeable Summaries



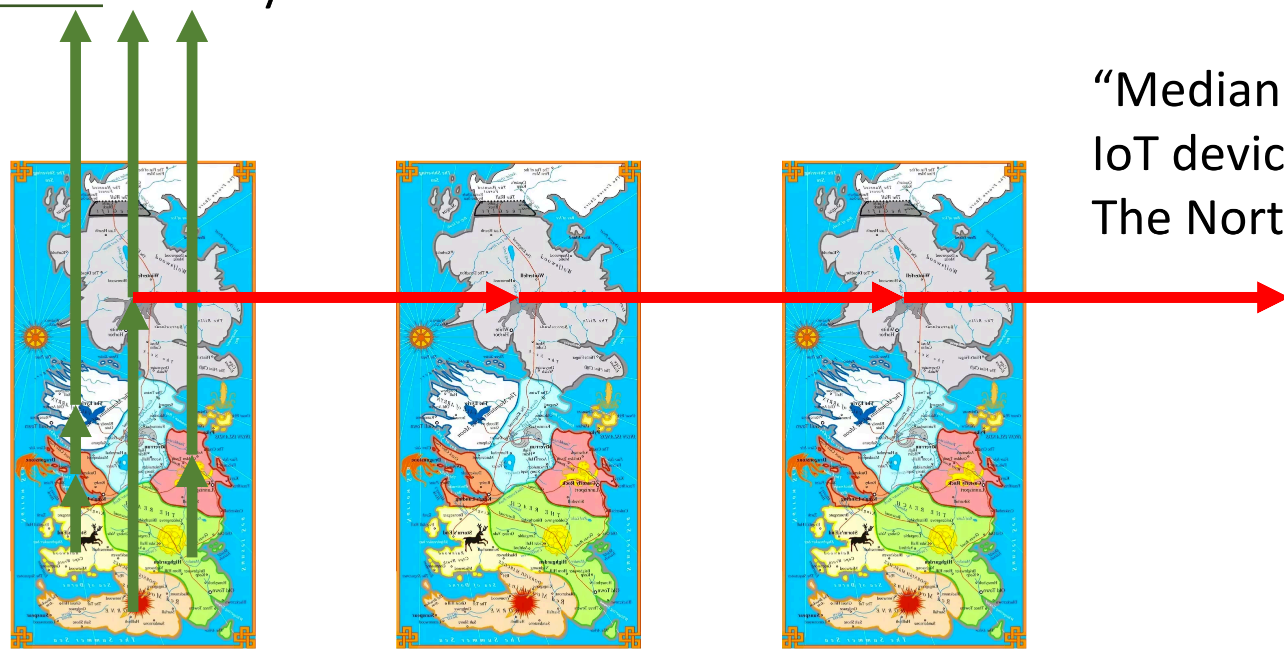
Every dataset is processed <u>once</u> for a model consisting of <u>the entire history</u>

Dynamic Windowing with Mergeable Summaries



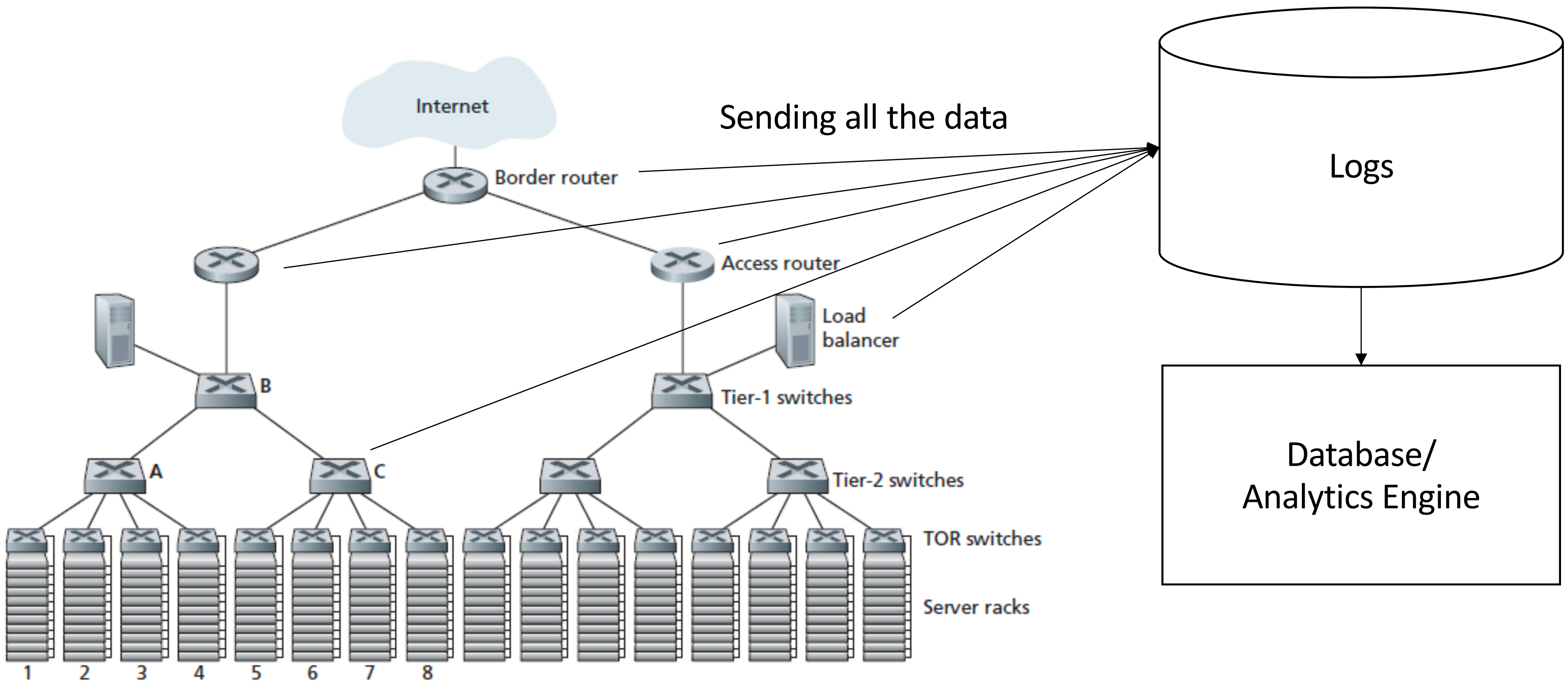
OLAP with Mergeable Summaries

"Median latency of
IoT device call in
Westeros January 2018"

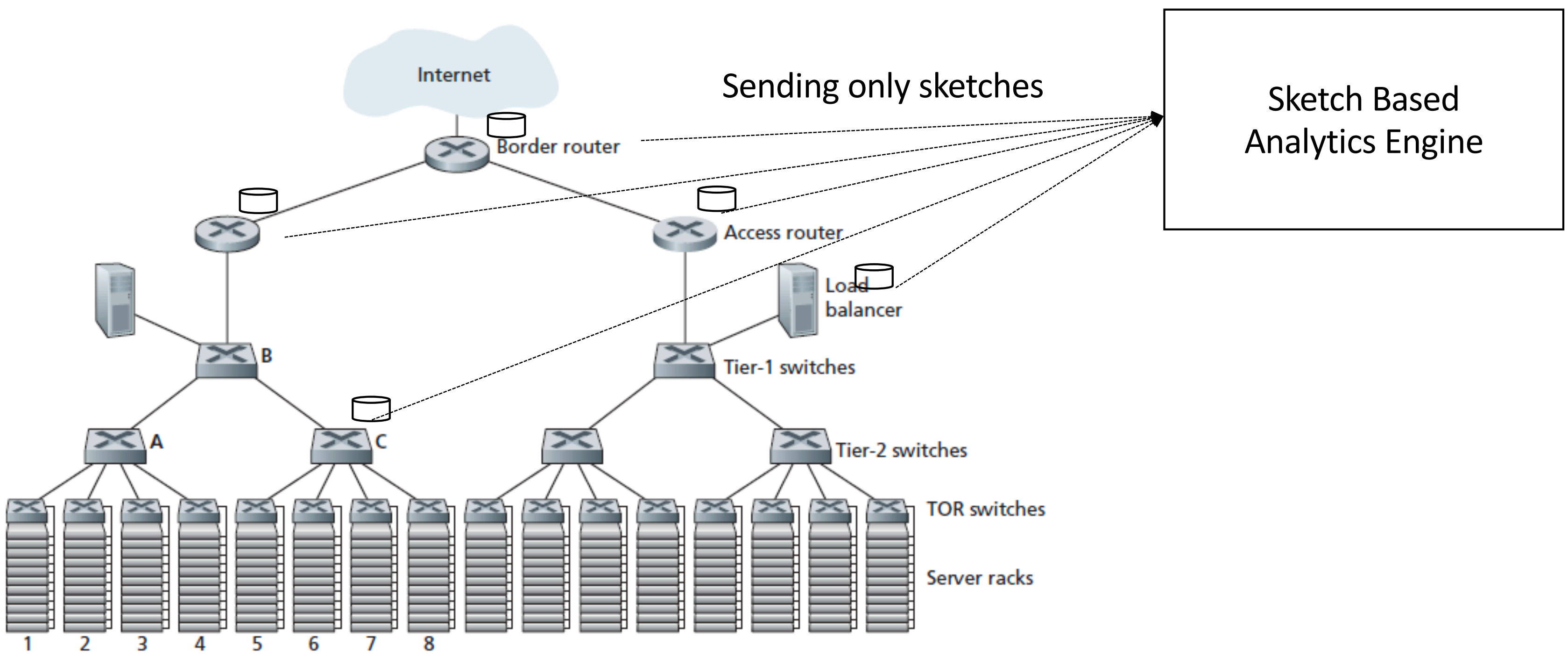


"Median latency of
IoT device call in
The North Q1 2018"

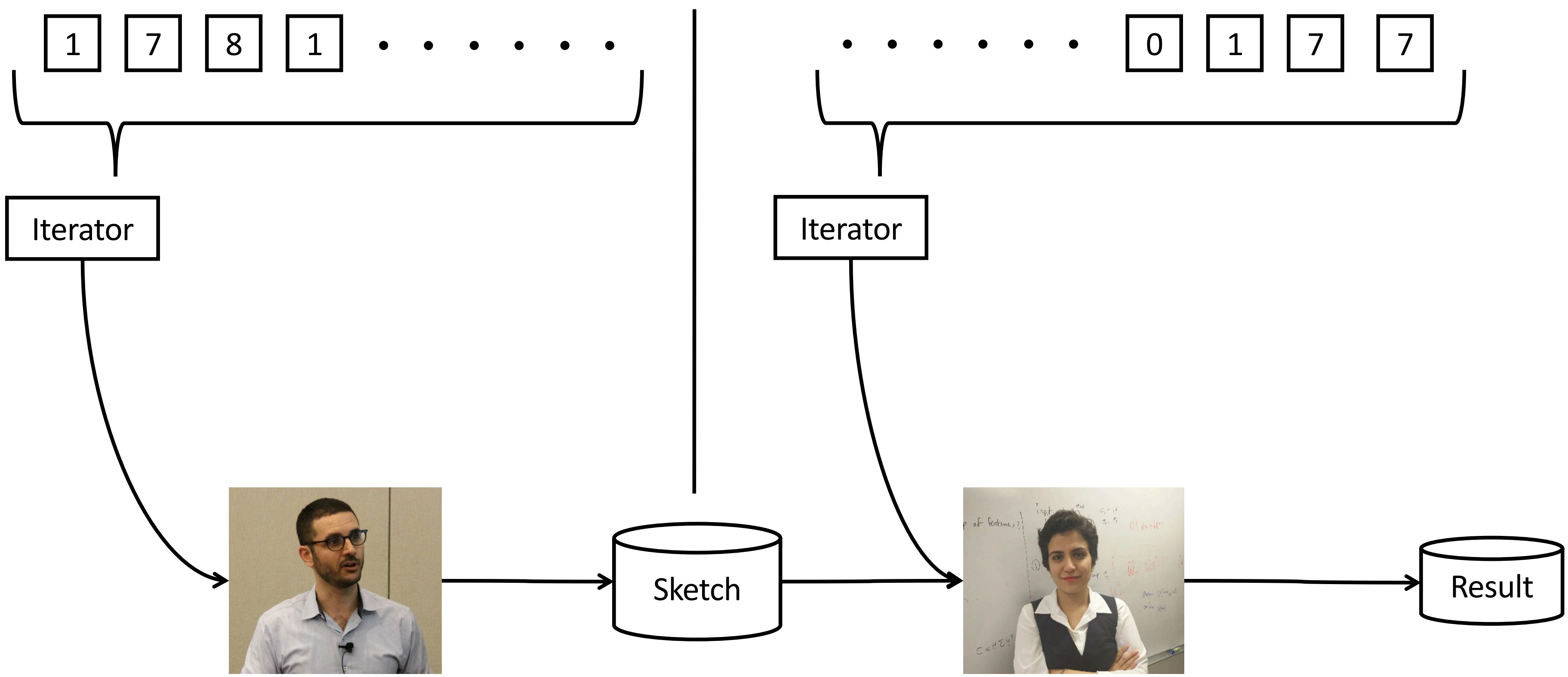
IoT and Cloud Monitoring



IoT and Cloud Monitoring



Some Basic Problems are Impossible



What can we do in this model?

Items

(words, IP-addresses, events, clicks,...)

- Counting distinct elements
- Item frequencies
- Approximate Quantiles
- Moment and entropy estimation
- Approximate set operations
- Sampling

Matrices

(text corpora, recommendations, ...)

- Covariance estimation matrix
- Low rank approximation
- Sparsification

Vectors

(text documents, images, example features,...)

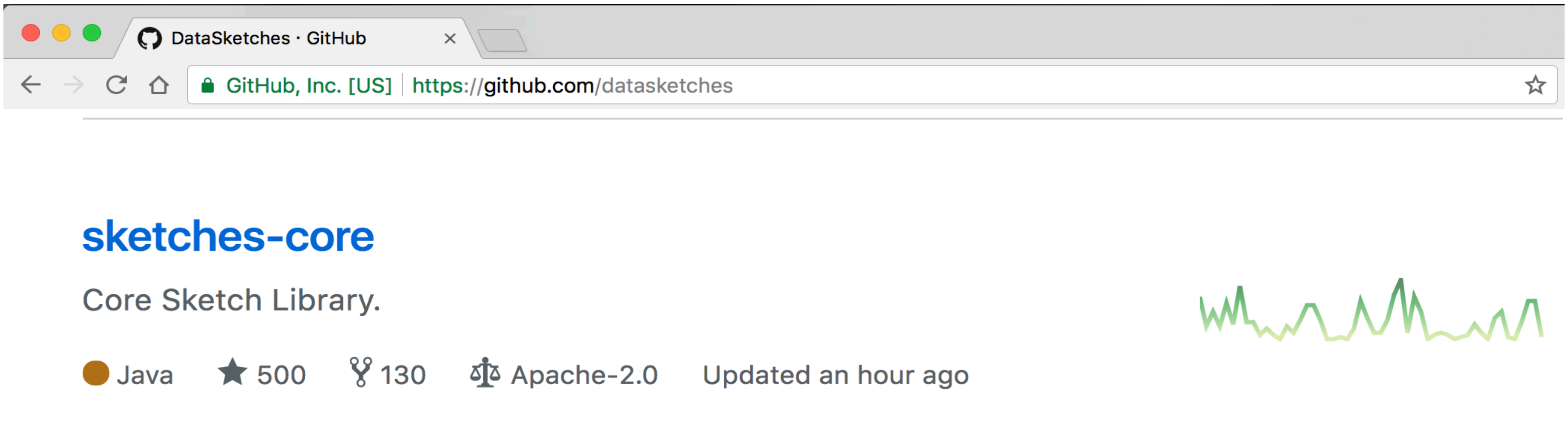
- Dimensionality reduction
- Clustering (k-means, k-median,…)
- Linear Regression
- Machine learning (some of it at least)
- Density Estimation / Anomaly detection

Graphs*

(social networks, communications, ...)

- Connectivity
- Cut Sparsification
- Weighted Matching

Apache Data Sketches




```
>> brew tap DataSketches/sketches-cmd
>> brew install data-sketches
🍺
```

L. Rhodes, K. Lang, A. Saydakov, J. Thaler, E. Liberty, and J. Malkin. DataSketches: A Java software library of stochastic streaming algorithms, 2017. https://datasketches.github.io.

Production ready

New Research

[ABL+17] Daniel Anderson, Pryce Bevan, Kevin J. Lang, Edo Liberty, Lee Rhodes, and Justin Thaler.
A high-performance algorithm for identifying frequent items in data streams.
In *ACM IMC 2017 (To Appear)*, 2017. Preliminary version available at https://arxiv.org/abs/1705.07001.

[DLRT16] Anirban Dasgupta, Kevin J. Lang, Lee Rhodes, and Justin Thaler.
A framework for estimating stream expression cardinalities. In *ACM ICDT Proceedings ‘16 *, pages 6:1–6:17, 2016.

[KLL16] Zohar S. Karnin, Kevin J. Lang, and Edo Liberty.
Optimal quantile approximation in streams. In *IEEE FOCS Proceedings ‘16*, pages 71–78, 2016.

[Lan17] Kevin J Lang. Back to the future: an even more nearly optimal cardinality estimation algorithm.
*arXiv preprint*https://arxiv.org/abs/1708.06839, 2017.

[Lib13] Edo Liberty. Simple and deterministic matrix sketching.
In *ACM KDD Proceedings ‘13*, pages 581– 588, 2013.

[LMTU16] Edo Liberty, Michael Mitzenmacher, Justin Thaler, and Jonathan Ullman.
Space lower bounds for itemset frequency sketches. In *ACM PODS Proceedings ‘16*, pages 441–454, 2016.

[MST12] Michael Mitzenmacher, Thomas Steinke, and Justin Thaler.
Hierarchical heavy hitters with the space saving algorithm. In *SIAM ALENEX Proceedings ‘12*, pages 160–174, 2012.

In this presentation

Items

(words, IP-addresses, events, clicks,...)

- Counting distinct elements
- Item frequencies
- Approximate Quantiles
- Moment and entropy estimation
- Approximate set operations
- Sampling

Vectors

(text documents, images, example features,...)

- Dimensionality reduction
- Clustering (k-means, k-median,...)
- Linear Regression
- Machine learning (some of it at least)
- Density Estimation / Anomaly detection

Matrices

(text corpora, recommendations, ...)

- Covariance estimation matrix
- Low rank approximation
- Sparsification

Graphs*

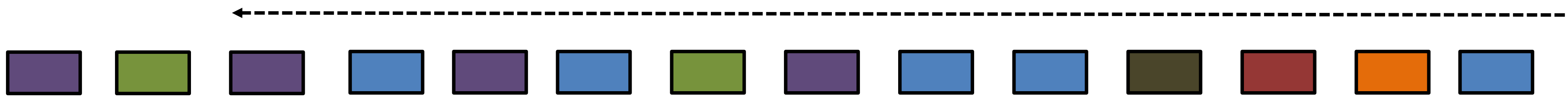
(social networks, communications, ...)

- Connectivity
- Cut Sparsification
- Weighted Matching

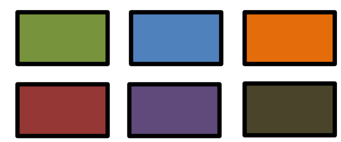
Counting Distinct Elements

N. Alon, Y. Matias, and M. Szegedy. The space complexity of approximating the frequency moments
Z. Bar-Yossef, T. Jayram, R. Kumar, D. Sivakumar, and L. Trevisan. Counting distinct elements in a data stream
E. Cohen. All-distances sketches, revisited: HIP estimators for massive graphs analysis
E. Cohen and H. Kaplan. Summarizing data using bottom-k sketches
G. Cormode. Sketch techniques for massive data
P. Flajolet and G. Nigel Martin. Probabilistic counting algorithms for data base applications
D. M. Kane, J. Nelson, and D. P. Woodruff. An optimal algorithm for the distinct elements problem
M. Thorup. Bottom-k and priority sampling, set similarity and subset sums with minimal independence
A. Dasgupta, K Lang, L. Rhodes, J. Thaler, A Framework for Estimating Stream Expression Cardinalities

Problem Definition

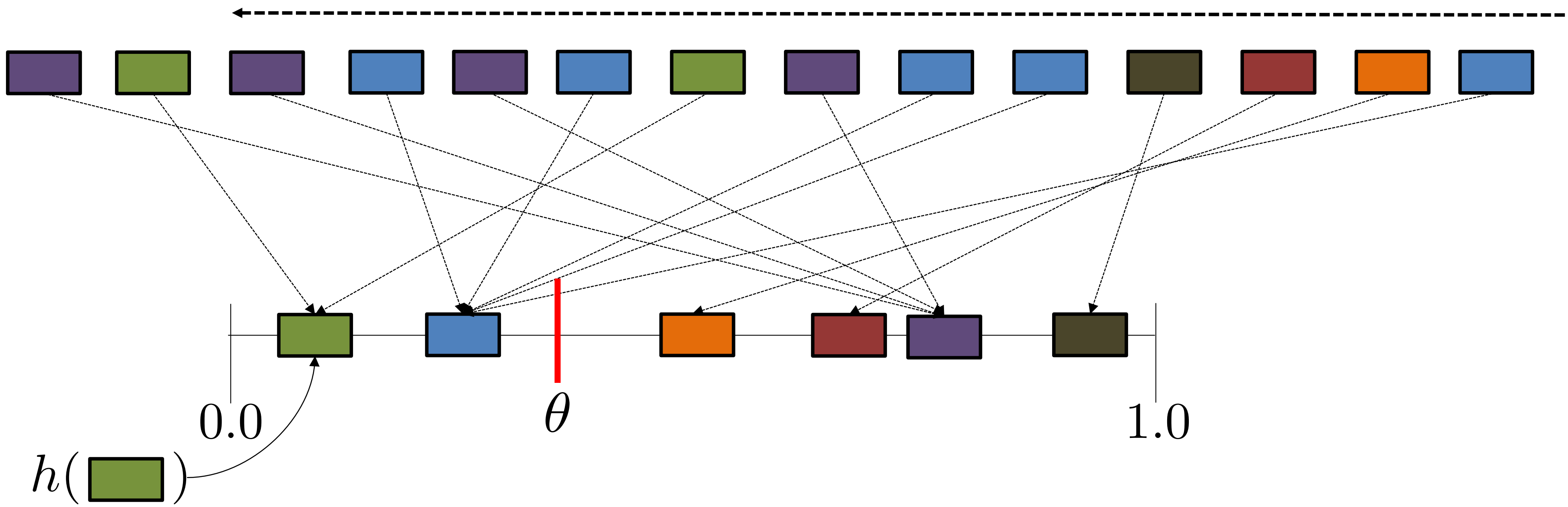


Approximate the number of *distinct* items in the stream



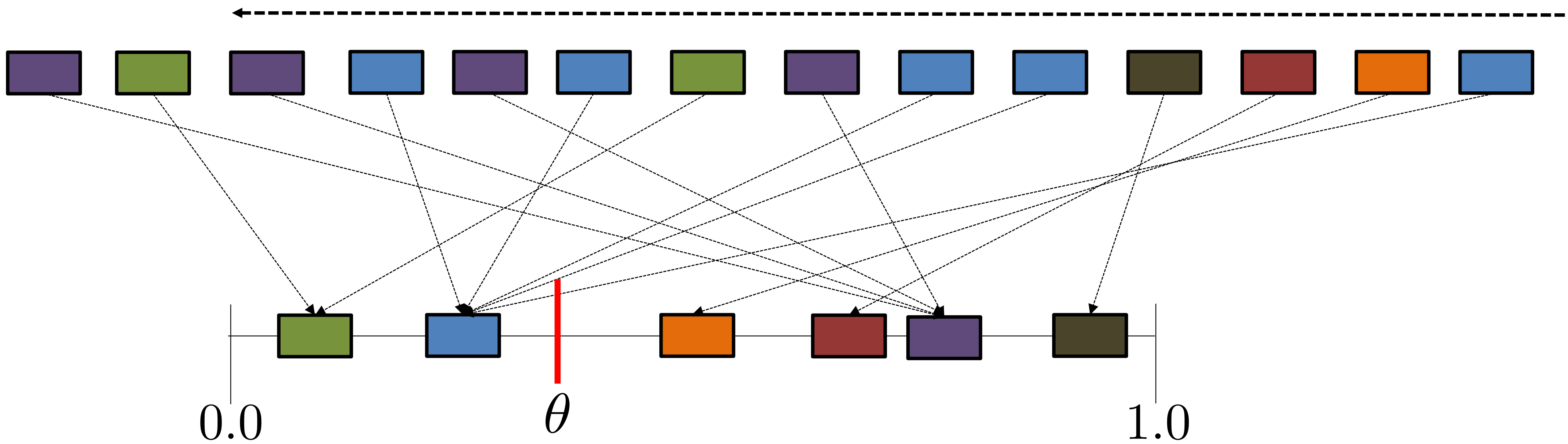
- # of unique IPs are important statistics of networks
- # of different customers using web services
- # of unique keys in a database join table
- ...

General Hashing Idea



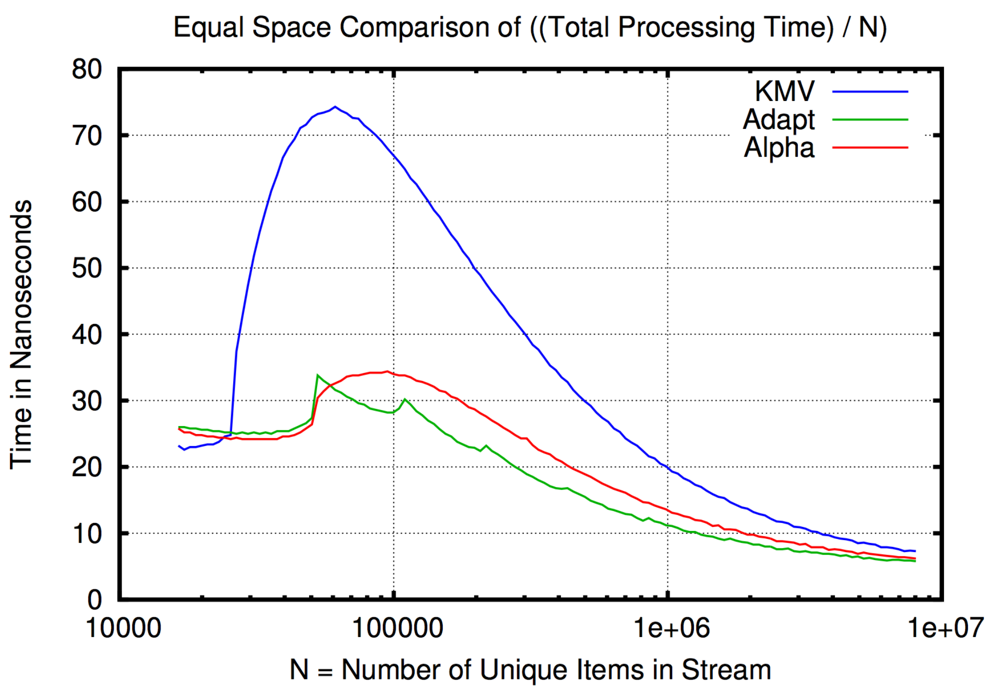
- Map all entries to the interval (0,1) using a hash function
- Keep only k values smaller than some threshold θ.
- From (k, θ) we can approximate the number of unique items.

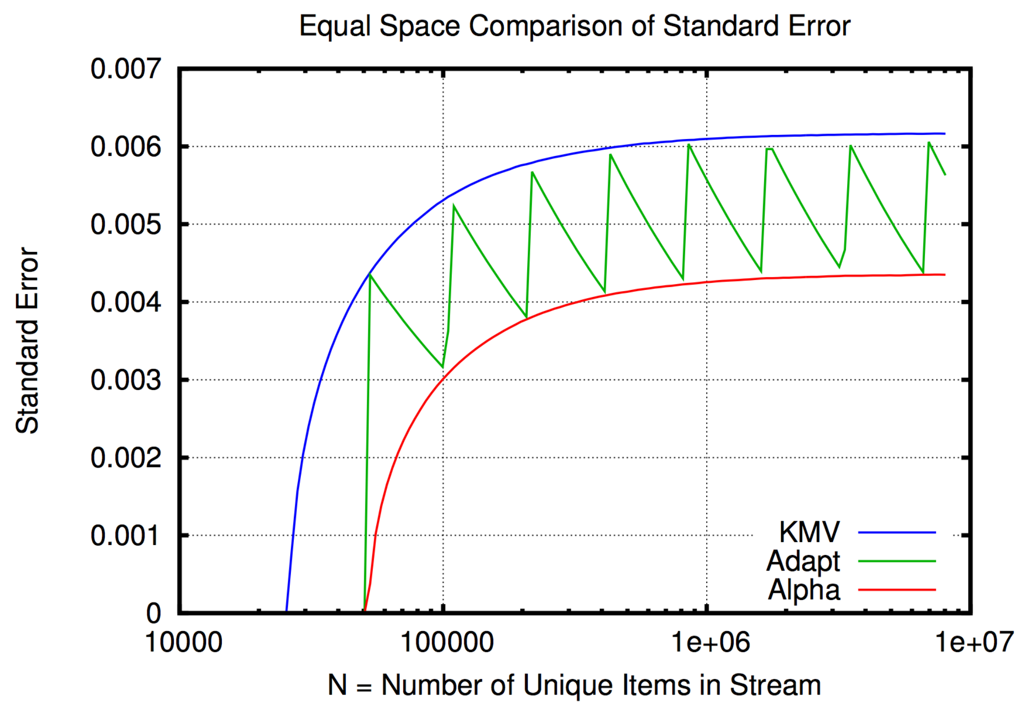
Our results



- Generalize a family of algorithms (Adaptive sampling, KMV)
- New Variance bounds for all such algorithms
- New tradeoffs between accuracy, space, and update time (alpha alg')
- Very careful implementation

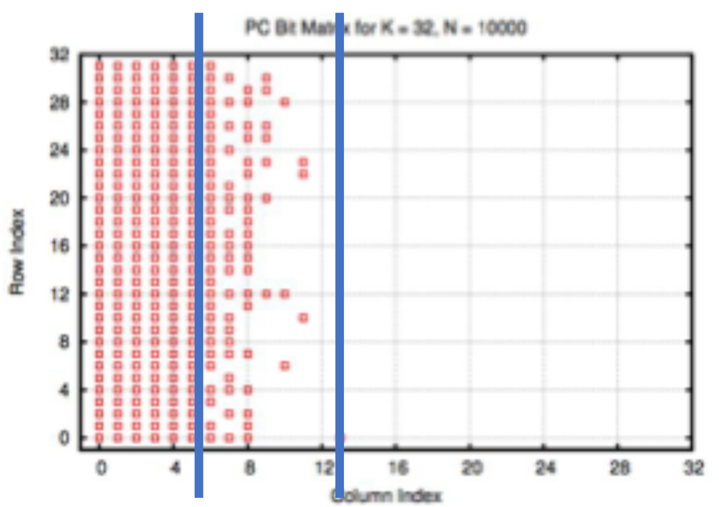
Experimental Results

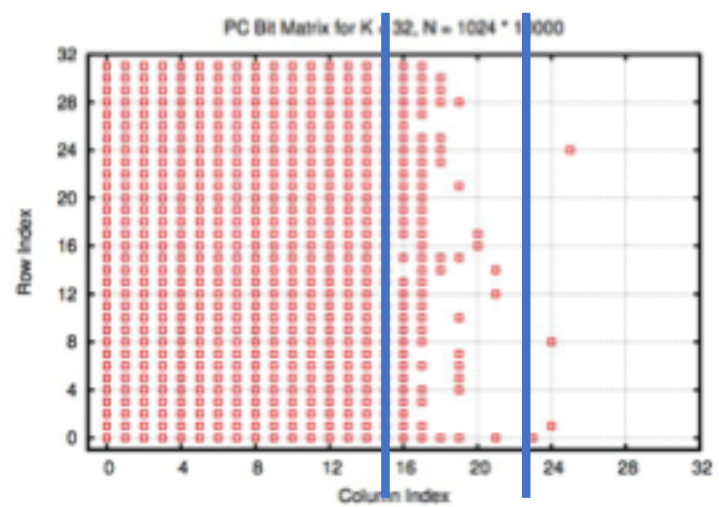




CPC - Compressed Probabilistic Counting

- A Better algorithm than **(HyperLogLog) HLL** was recently invented by **Kevin Lang**
- The result will be published by the datasketches group soon





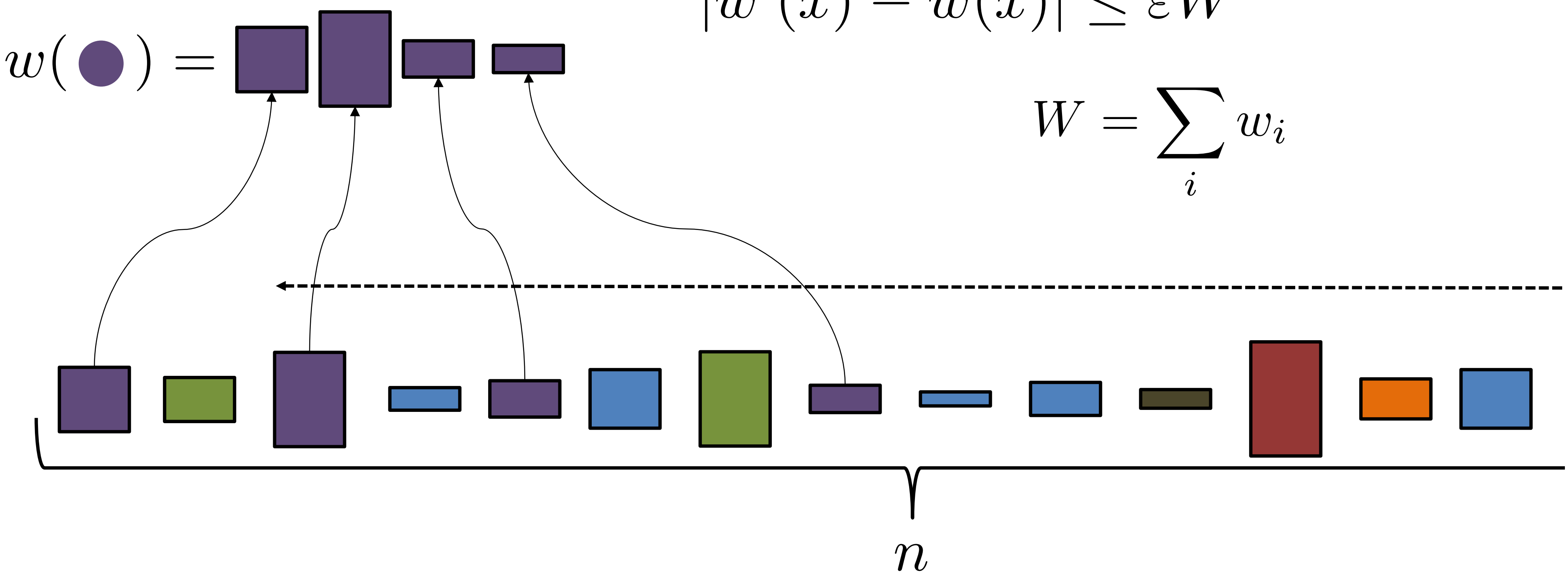
Weighted Item frequencies

Space-optimal heavy hitters with strong error bounds.] R. Berinde, P. Indyk, G. Cormode, and M. J. Strauss
An optimal algorithm for `1-heavy hitters in insertion streams and related problems A. Bhattacharyya, P. Dey, and D. P. Woodruff
Finding frequent items in data streams M. Charikar, K. Chen, and M. Farach-Colton
Methods for finding frequent items in data streams G. Cormode and M. Hadjieleftheriou
An improved data stream summary: The count-min sketch and its applications. G. Cormode and S. Muthukrishnan.
Approximate frequency counts over data streams G. S. Manku and R. Motwani.
Efficient computation of frequent and top-k elements in data streams A. Metwally, D. Agrawal, and A. El Abbadi.
Finding repeated elements J. Misra and D. Gries.
A High-Performance Algorithm for Identifying Frequent Items in Data Streams
Daniel Anderson Pryce Bevin, Kevin Lang, Edo Liberty, Lee Rhodes, Justin Thaler

Problem Definition

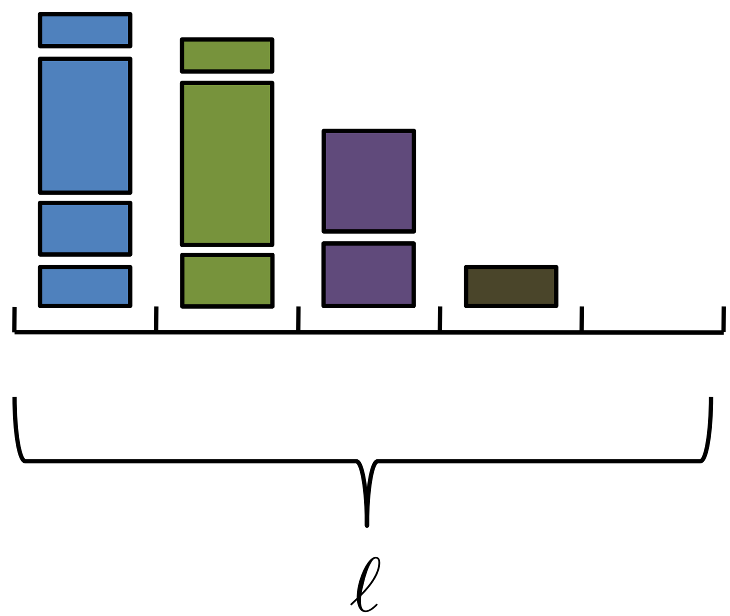
$$|w'(x) - w(x)| \leq \varepsilon W$$

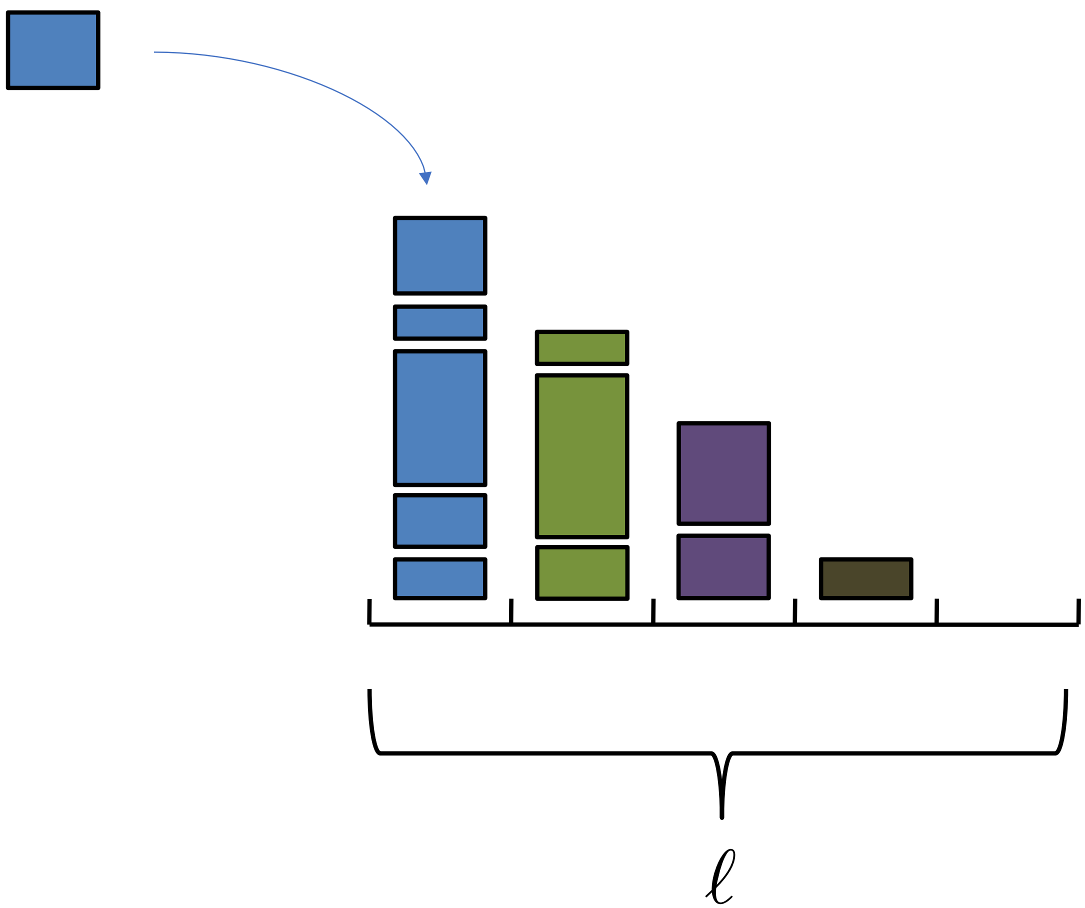
$$W = \sum_i w_i$$

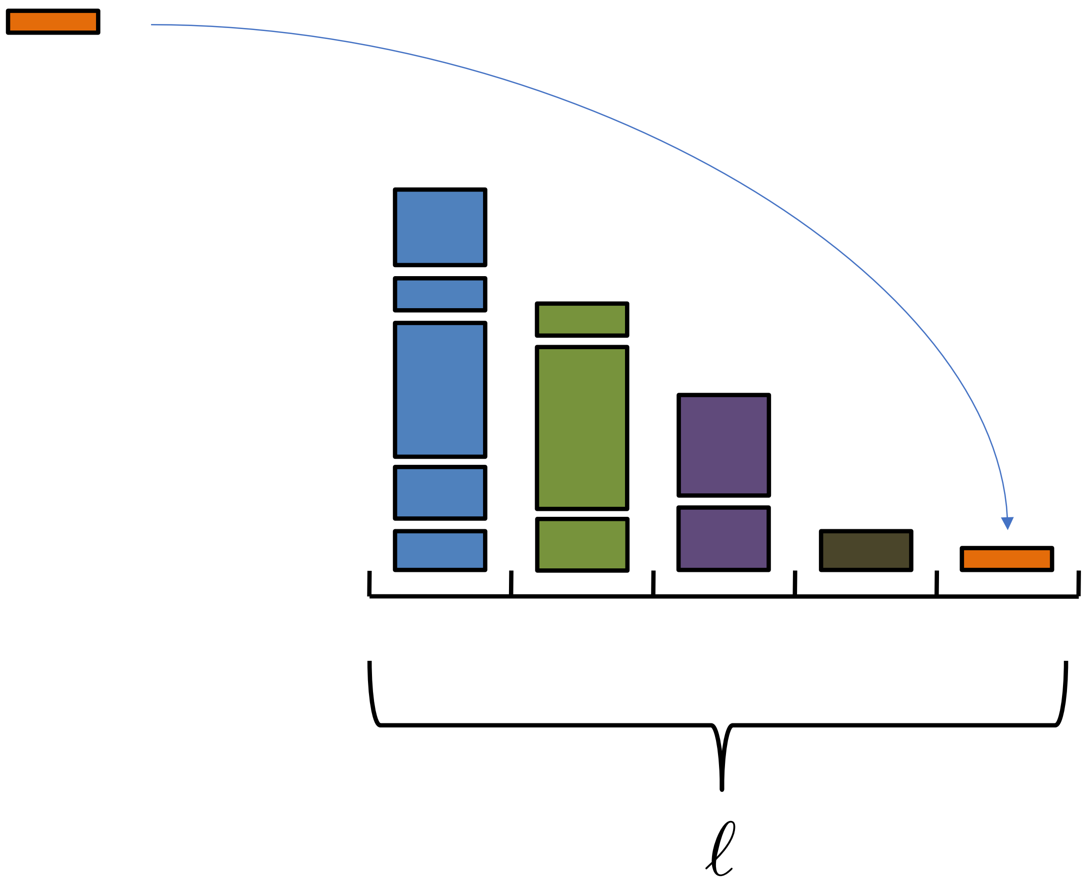


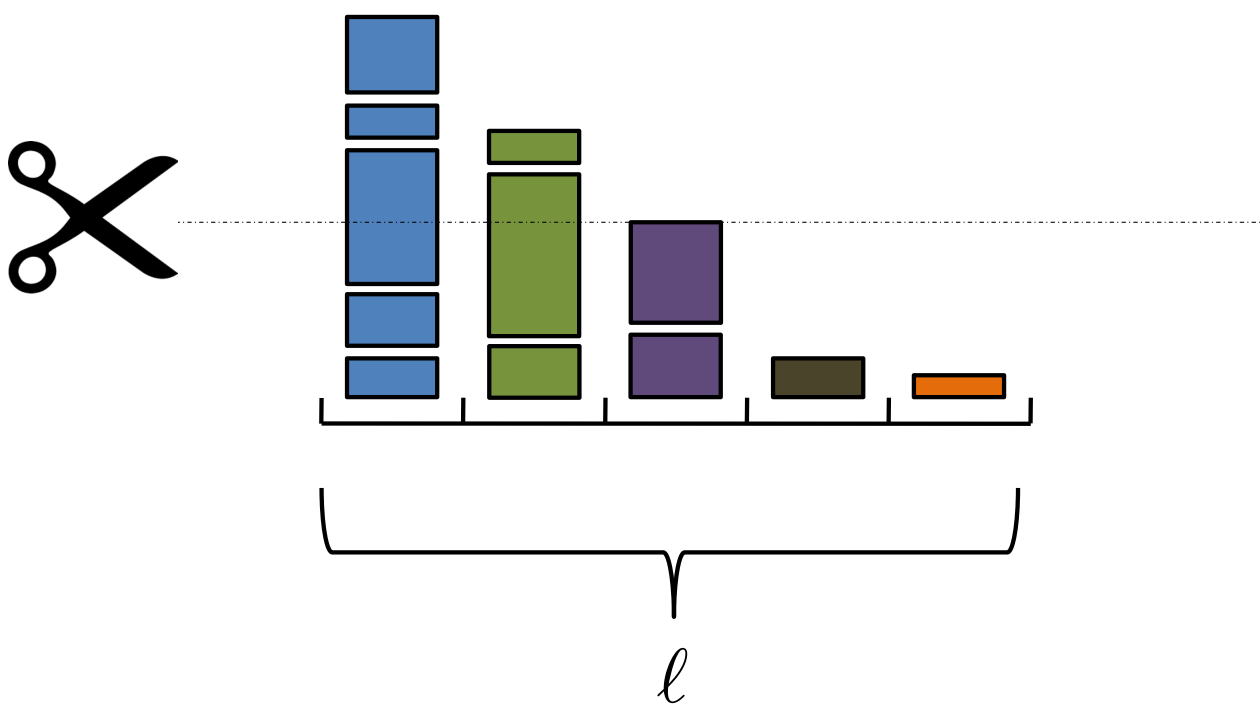
Our Contributions

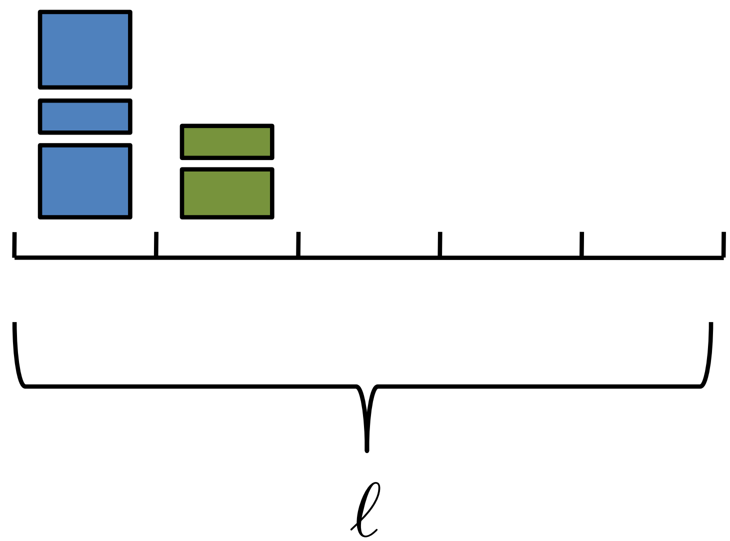
- Improved streaming algorithm for weighted updates
- Improved merging procedure
- Improved Estimator
- Careful implementation


ℓ

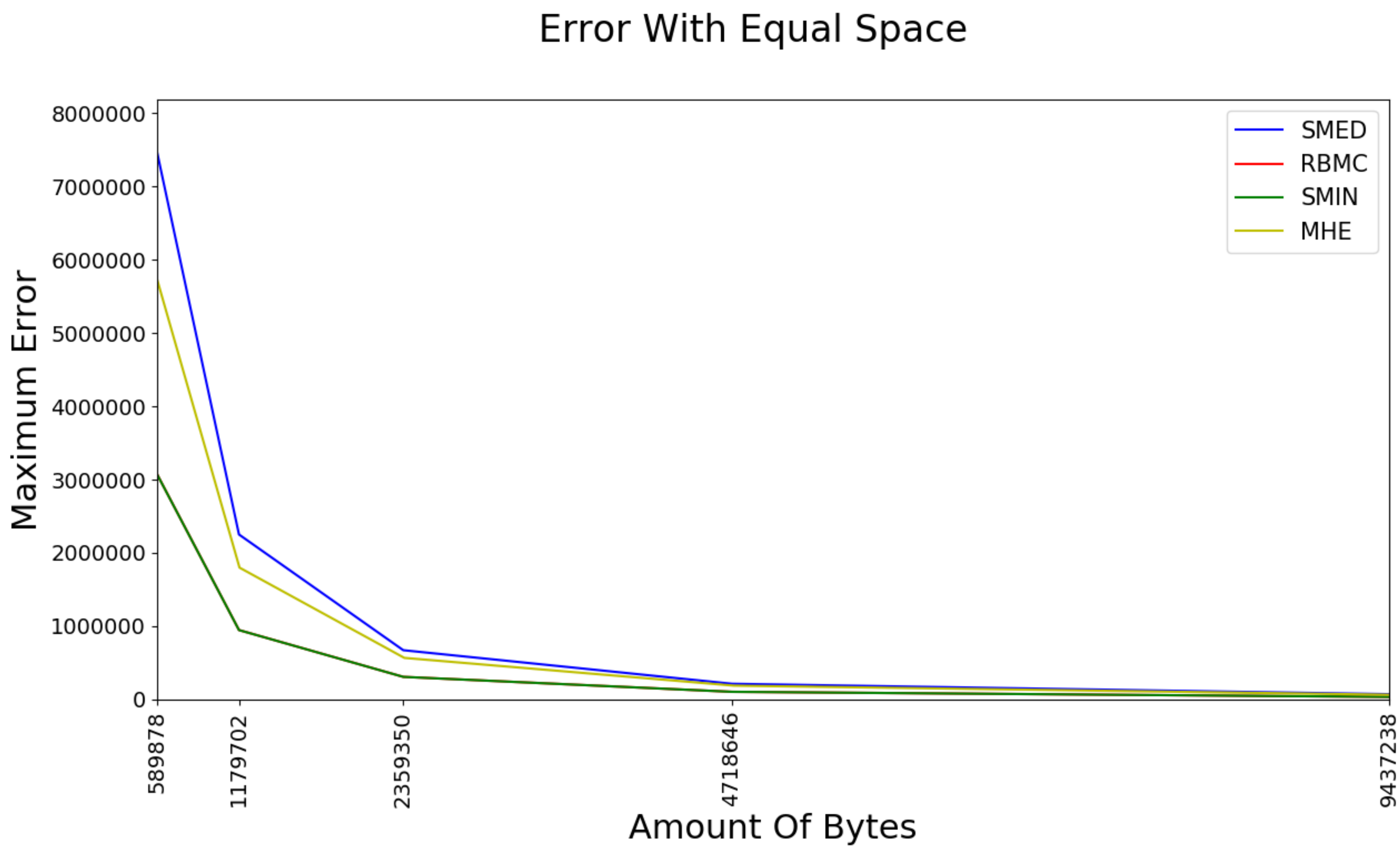

ℓ


ℓ

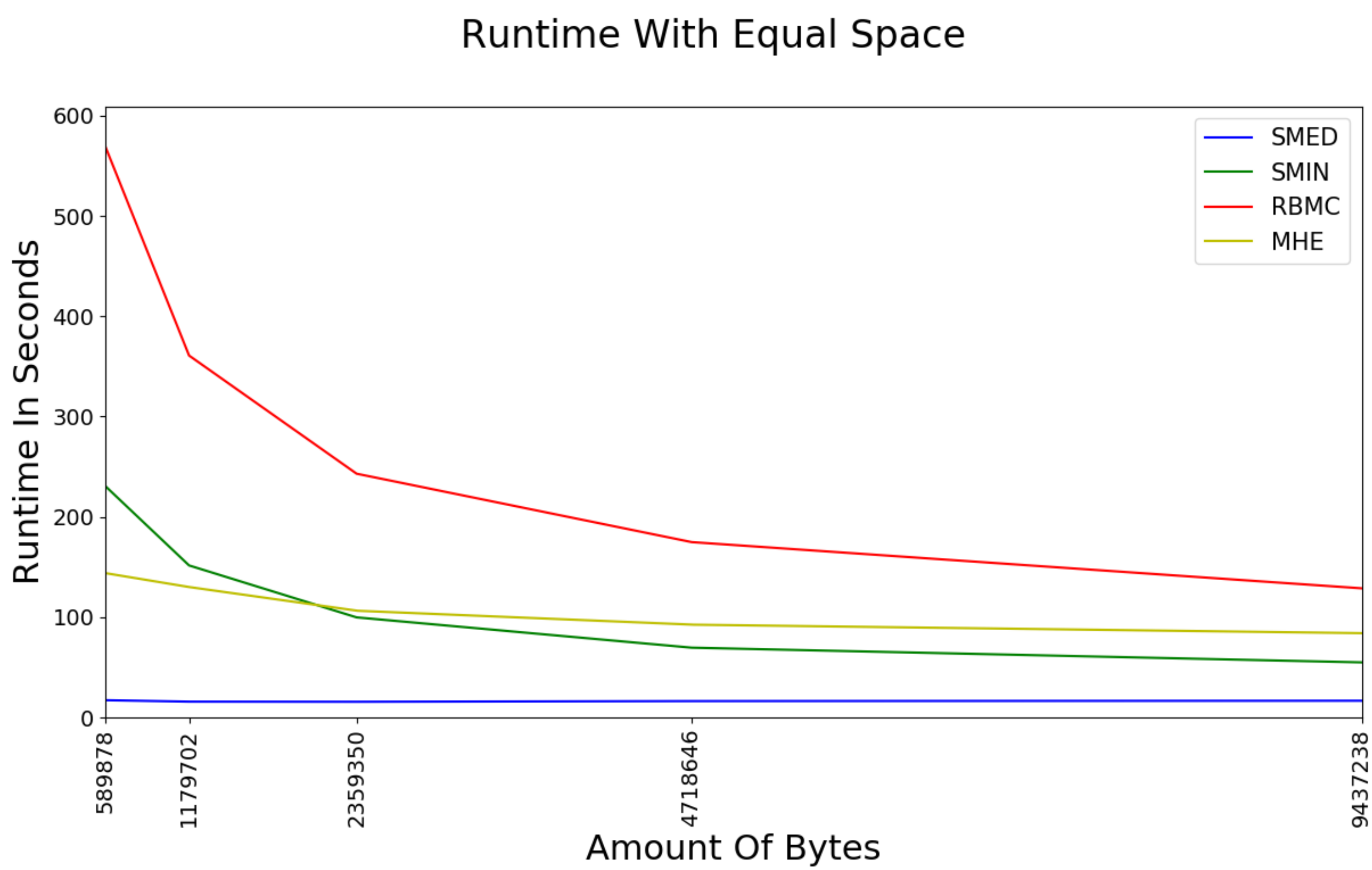

ℓ


ℓ

Comparable Error

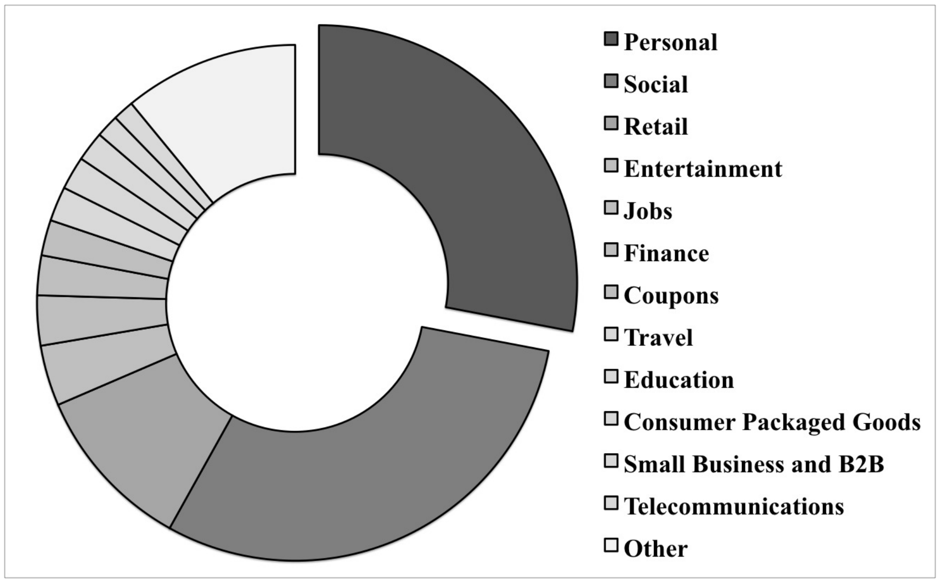


Significantly Faster Updates

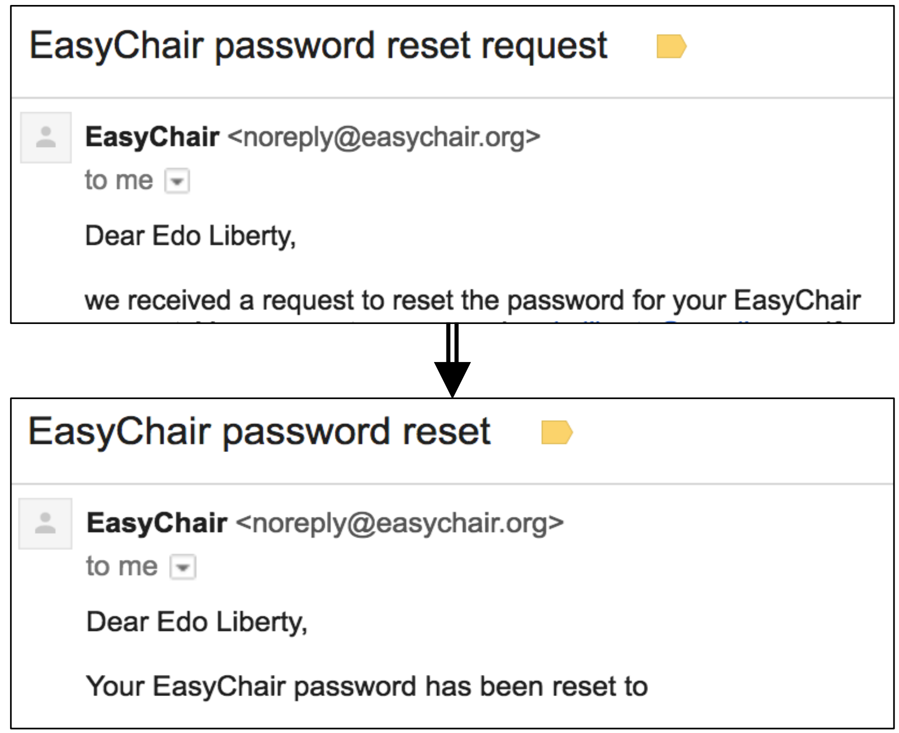


Weighted Item Frequencies Application

Threading Machine Generated Email



Ailon, Karnin, Maarek, Liberty,
Threading Machine Generated Email, WSDM 2013



Threading Machine Generated Email

If b should be threaded with a then the lift should be large

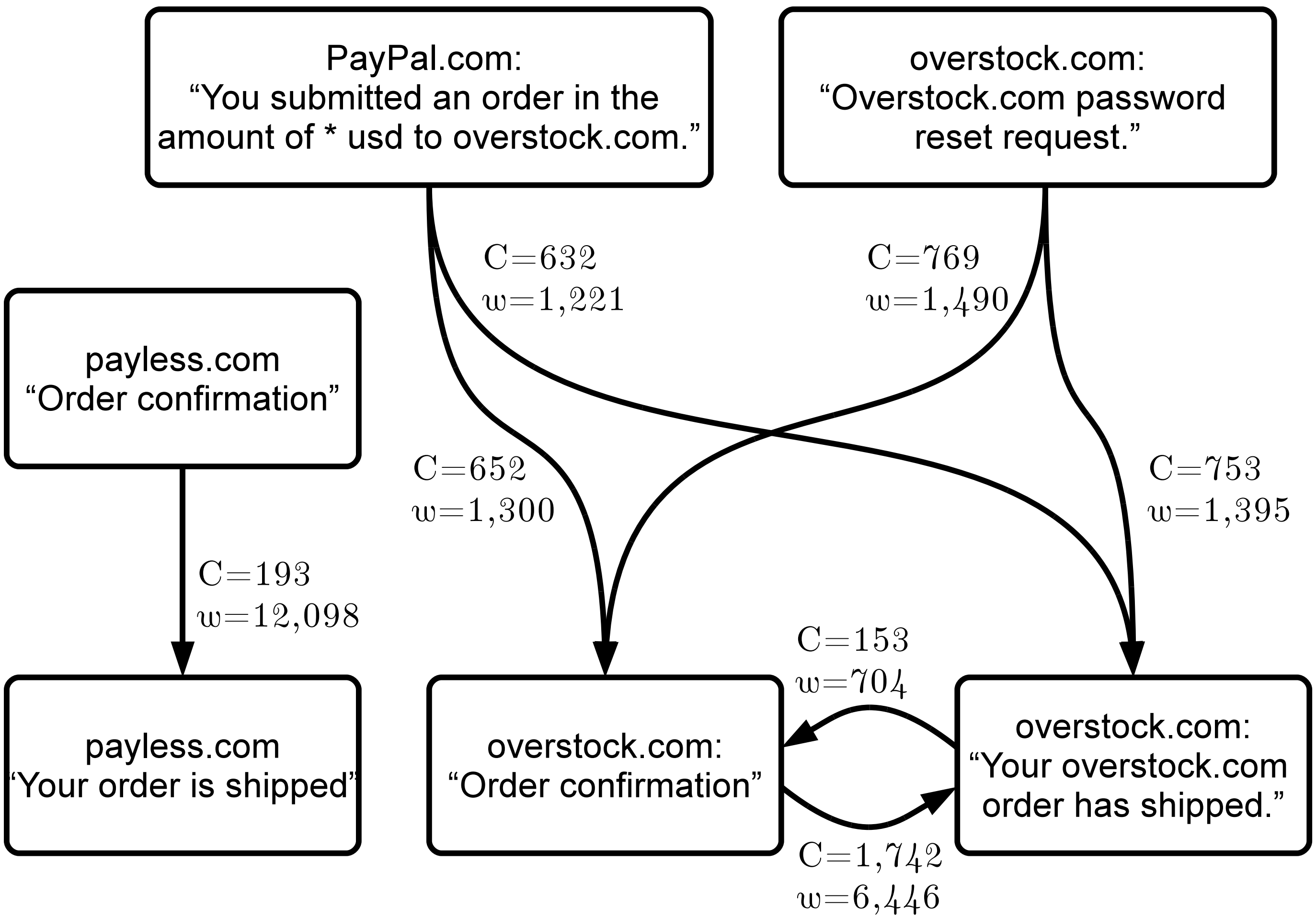
$$\mathrm{lift}(a, b) = p(b|a)/p(b) \gg 1$$

Alas, computing all pair conditional probability is impossible!

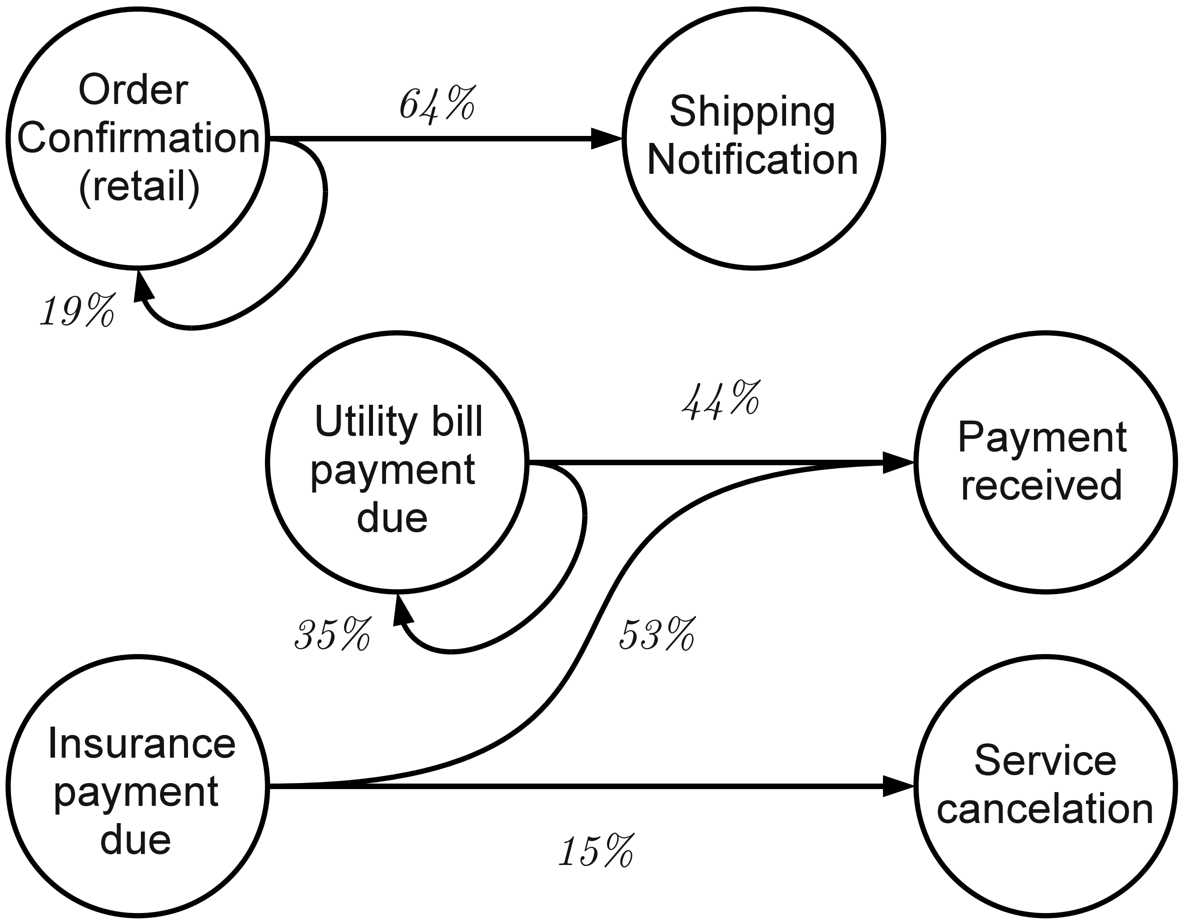
$$\mathrm{lift}(a, b) = n(b, a)\frac{n}{n(b)n(a)} = n(b, a)w(b)$$

This is possible with weighted frequency sketching!

Threading Machine Generated Email



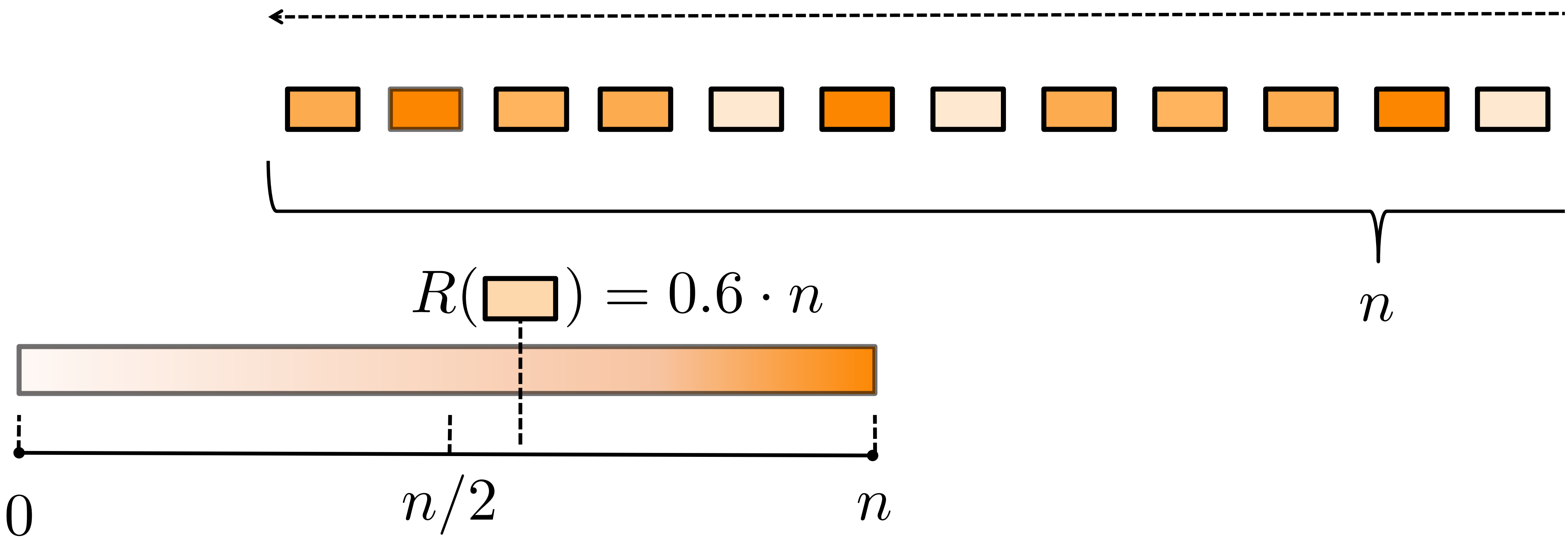
Threading Machine Generated Email



Streaming quantiles

Manku, Rajagopalan, Lindsay. Random sampling techniques for space efficient online computation of order statistics of large datasets.
Munro, Paterson. Selection and sorting with limited storage.
Greenwald, Khanna. Space-efficient online computation of quantile summaries.
Wang, Luo, Yi, Cormode. Quantiles over data streams: An experimental study.
Greenwald, Khanna. Quantiles and equidepth histograms over streams.
Agarwal, Cormode, Huang, Phillips, Wei, Yi. Mergeable summaries.
Felber, Ostrovsky. A randomized online quantile summary in $O((1/\varepsilon) \log(1/\varepsilon))$ words.
Lang, Karnin, Liberty, Optimal Quantile Approximation in Streams.
Ivking, Lang, Karnin, Liberty, Braverman, Streaming quantiles algorithms with small space and update time

Problem Definition

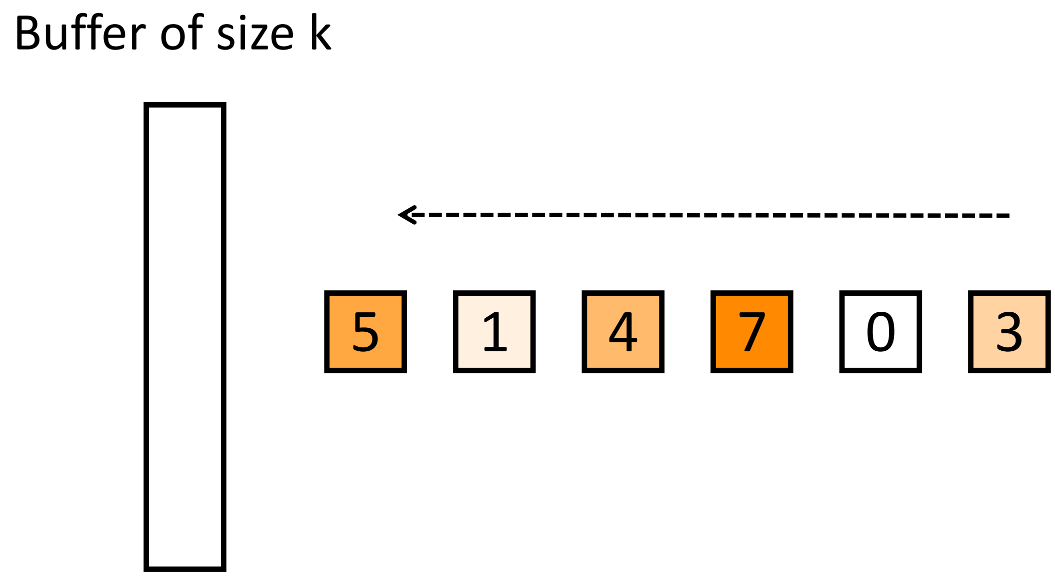


Sketch the stream to estimate $|R' - R| < \varepsilon n$

Our result

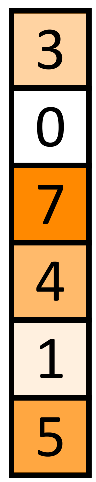
Algorithm	Simple?	Mergeable	Space Complexity	
Uniform sampling	✓	✓	$1/\varepsilon^2$	
Greenwald Khanna (GK)	✗	✗	$\log(n)/\varepsilon$	
Felber-Ostrovsky	✗	✗	$\log(1/\varepsilon)/\varepsilon$	
Manku-Rajagopalan-Lindsay (MRL)	✓	✓	$\log^2(n)/\varepsilon$	
Agarwal, Cormode, Huang, Phillips, Wei, Yi	✓✗	✓	$\log^{3/2}(1/\varepsilon)/\varepsilon$	Implemented
Karnin, Lang, Liberty	✓	✓	$\sqrt{\log(1/\varepsilon)}/\varepsilon$	Implemented
Karnin, Lang, Liberty	✓✗	✗	$\log^2\log(1/\varepsilon)/\varepsilon$	
Karnin, Lang, Liberty	✗	✗	$\log\log(1/\varepsilon)/\varepsilon$	Space Optimal
Still open...	✓	✓	$\log\log(1/\varepsilon)/\varepsilon$	

The basic buffer idea



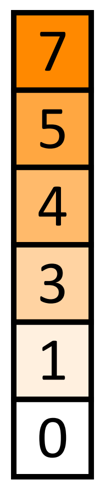
The basic buffer idea

Stores k stream entries



The basic buffer idea

The buffer sorts k stream entries

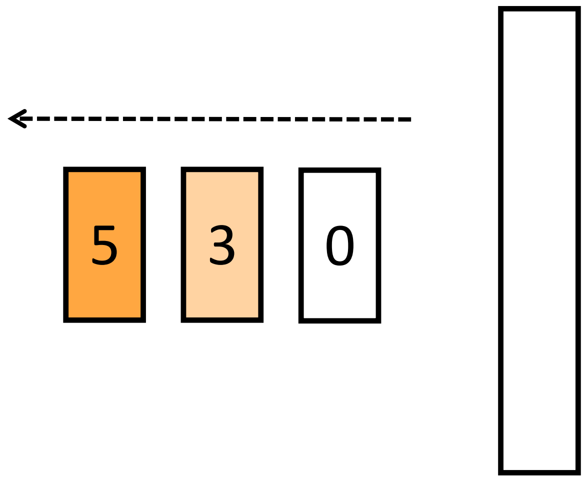


The basic buffer idea

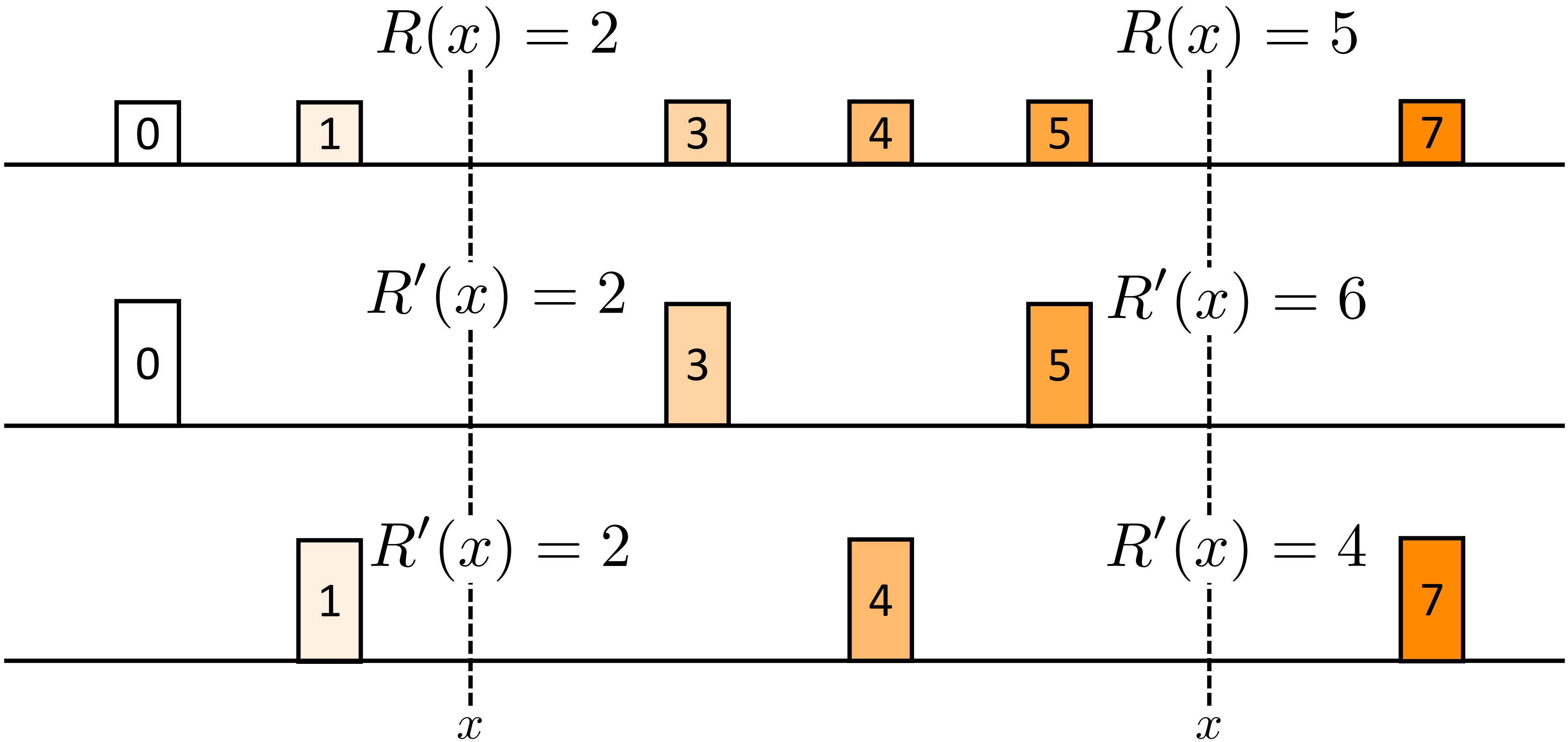
Deletes every other item

The basic buffer idea

And outputs the rest
with double the weight

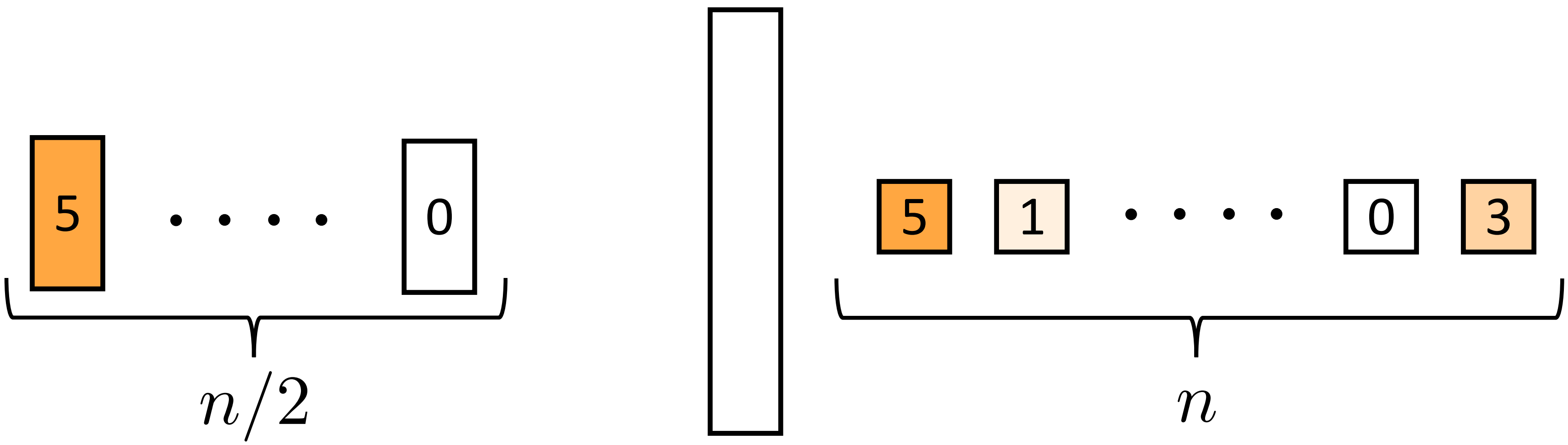


The basic buffer idea



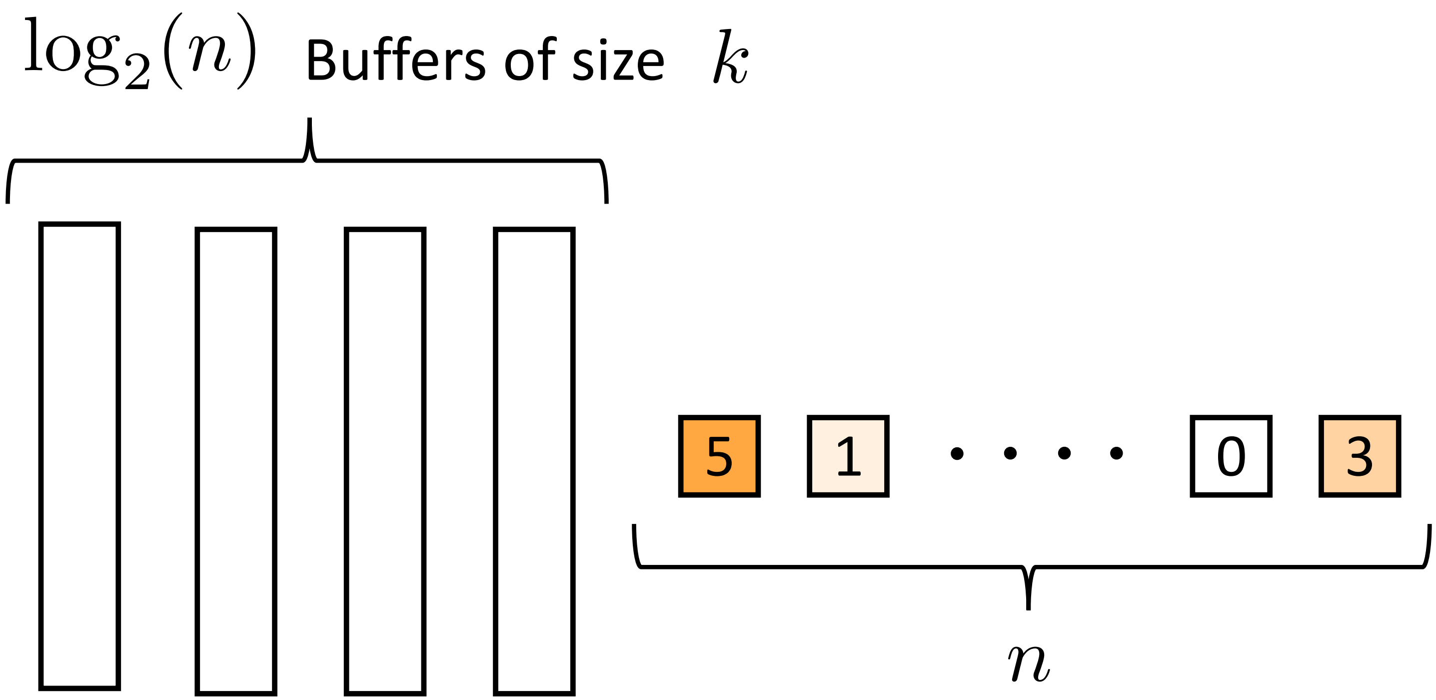
The basic buffer idea

Repeat n/k time until
the end of the stream



$$|R'(x) - R(x)| < n/k$$

Manku-Rajagopalan-Lindsay (MRL) sketch



$$|R'(x) - R(x)| \leq n \log_2(n)/k$$

Manku-Rajagopalan-Lindsay (MRL) sketch

If we set $k = \log_2(n)/\varepsilon$

We get $|R'(x) - R(x)| \leq \varepsilon n$

And we maintain only $\log_2^2(n)/\varepsilon$ items from the stream!

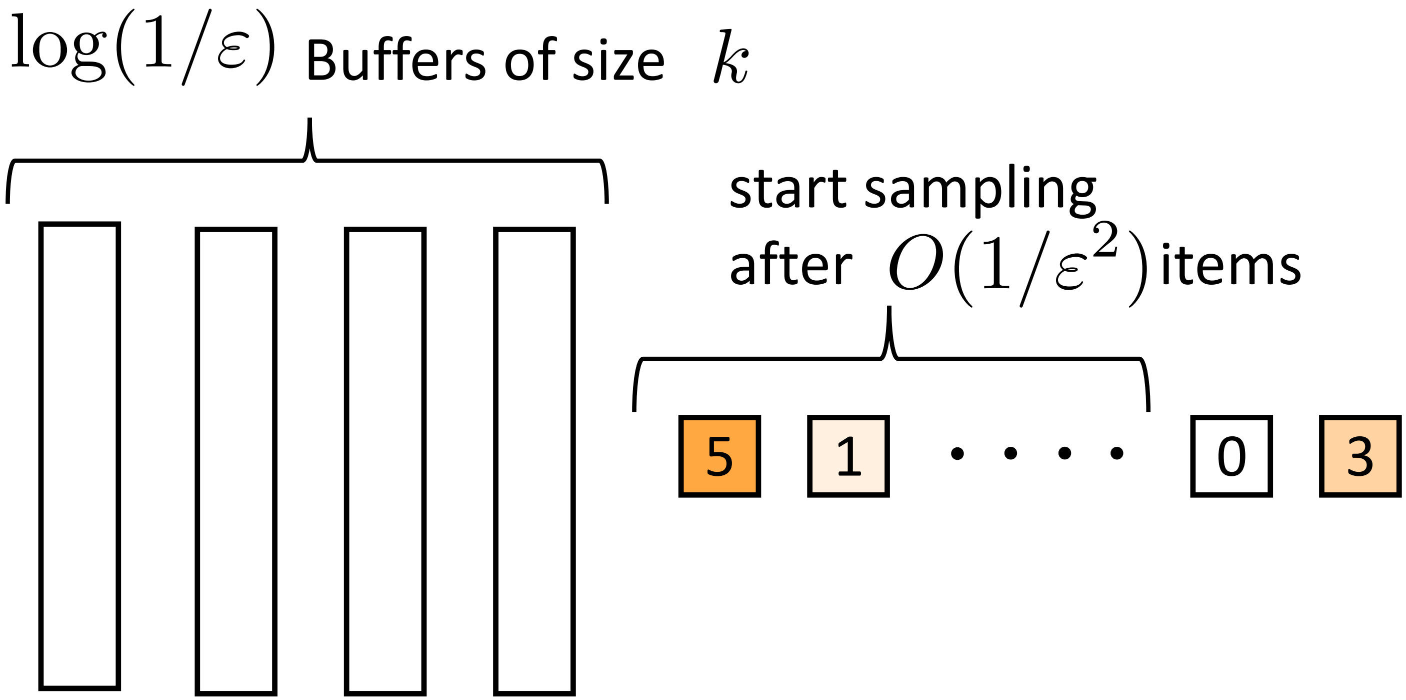
Greenwald-Khanna (GK) sketch

Uses a completely different construction

It gets $|R'(x) - R(x)| \leq \varepsilon n$

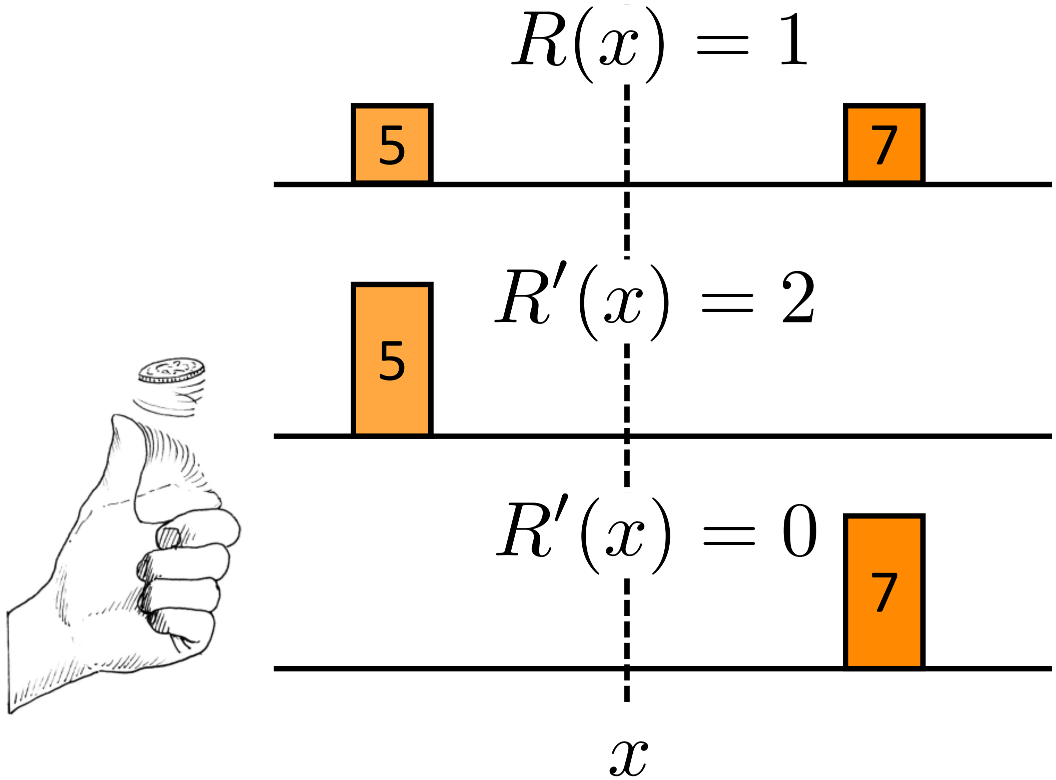
And maintains only $O(\log(n)/\varepsilon)$ items from the stream!

Agarwal, Cormode, Huang, Phillips, Wei, Yi (1)



Reduces space usage to $\log^2(1/\varepsilon)/\varepsilon$ items from the stream.

Agarwal, Cormode, Huang, Phillips, Wei, Yi (2)

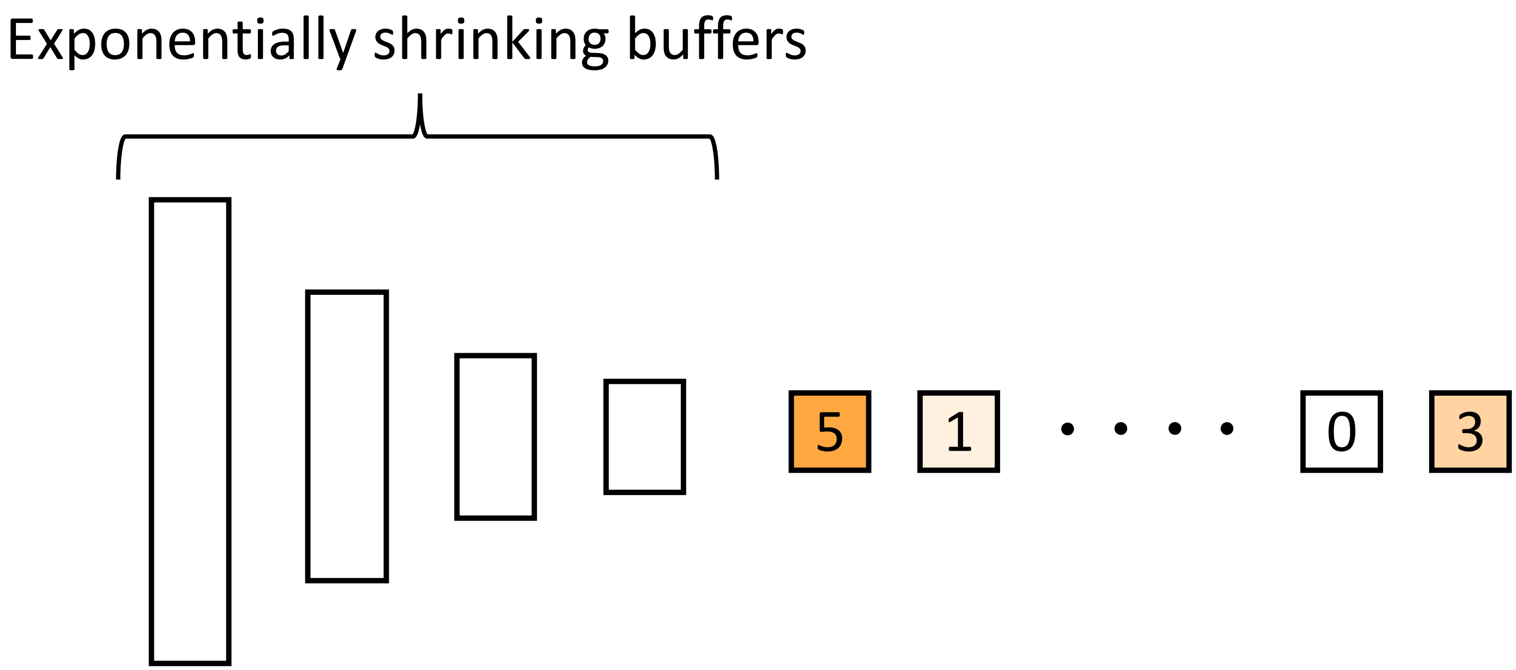


$R'(x)$ is a random variable now and

$E[R'(x)] = R(x)$

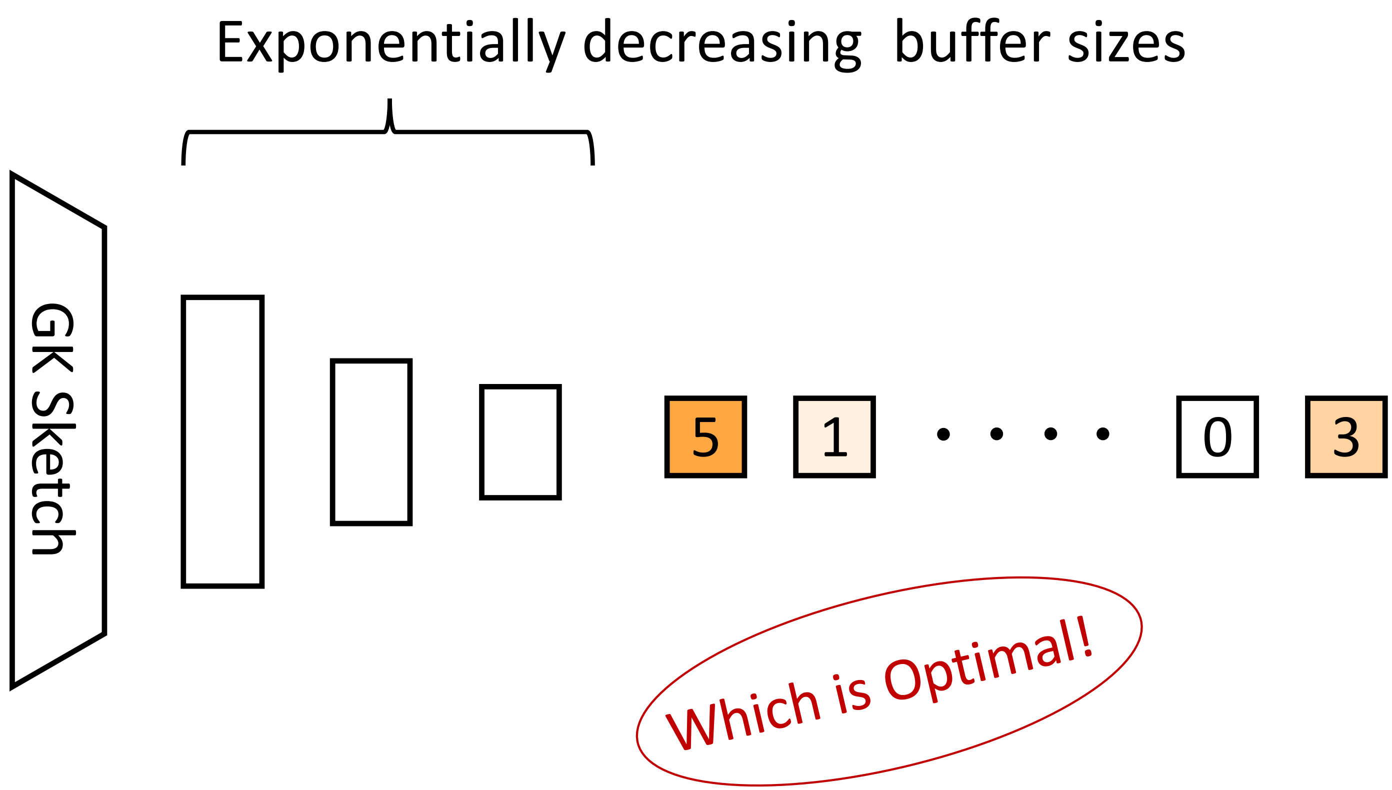
Reduces space usage to $\log^{3/2}(1/\varepsilon)/\varepsilon$ items from the stream.

Lang, Karnin, Liberty (1)



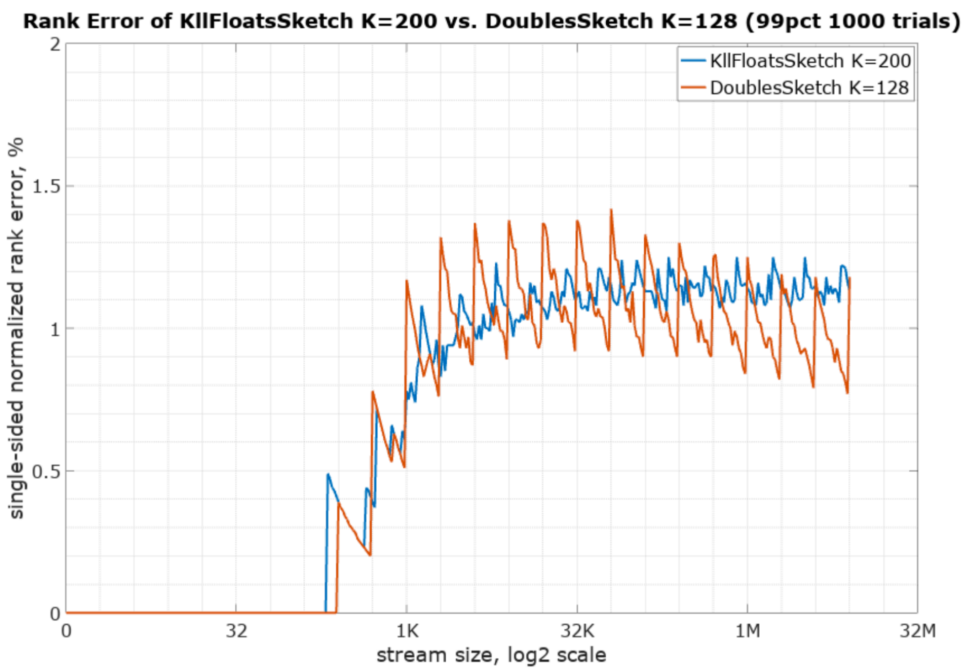
Reduces space usage to $\sqrt{\log(1/\varepsilon)}/\varepsilon$ items from the stream.

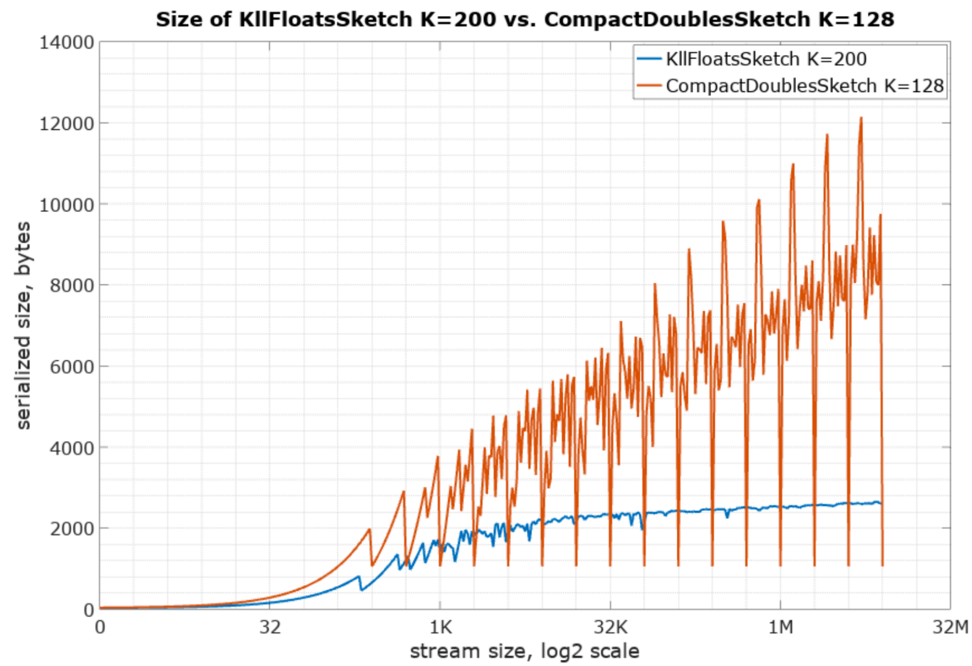
Lang, Karnin, Liberty (2)



Reduces space usage to $\log\log(1/\varepsilon)/\varepsilon$ items from the stream.

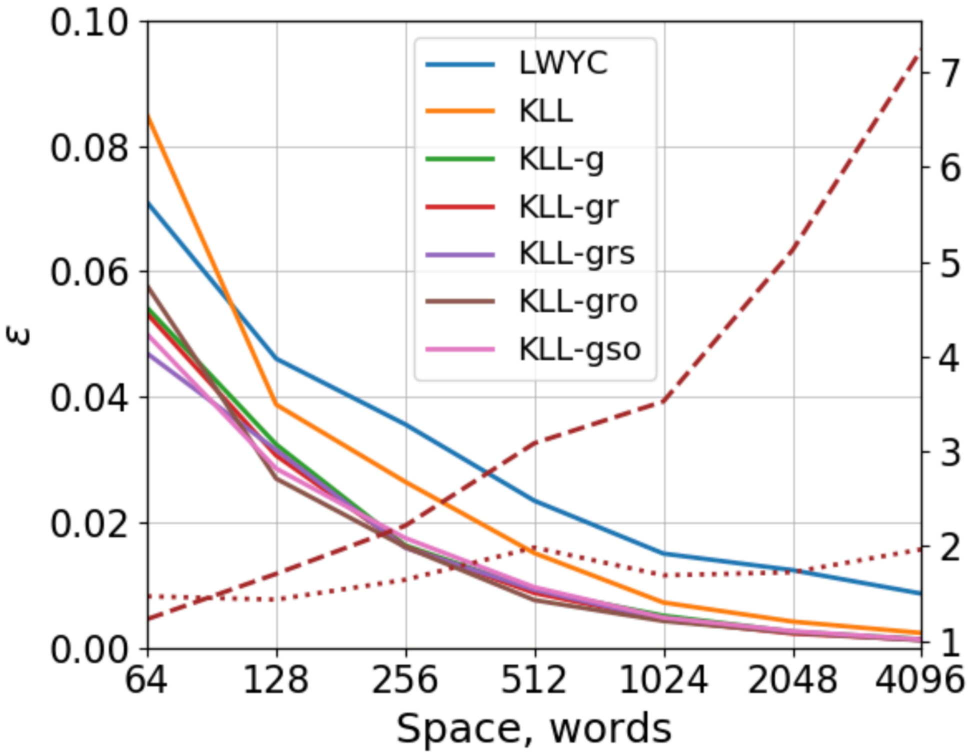
Experimental Results

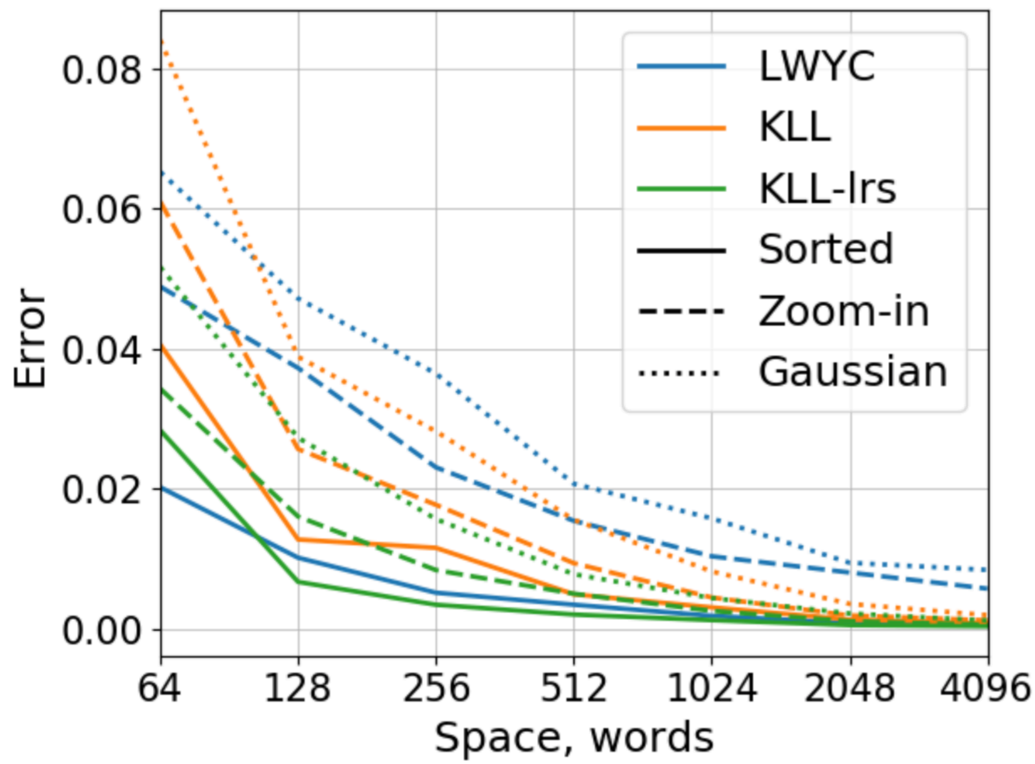




More experiments: https://datasketches.github.io/docs/Quantiles/KLLSketch.html
https://datasketches.github.io/docs/Quantiles/KllSketchVsTDigest.html

Even Newer Experimental Results





Ivking, Lang, Karnin, Liberty, Braverman, Streaming quantiles algorithms with small space and update time

What else can we do in this model?

Items

(words, IP-addresses, events, clicks,...)

- Counting distinct elements
- Item frequencies
- Approximate Quantiles
- Moment and entropy estimation
- Approximate set operations
- Sampling

Vectors

(text documents, images, example features,...)

- Dimensionality reduction
- Clustering (k-means, k-median,...)
- Linear Regression
- Machine learning (some of it at least)
- Density Estimation / Anomaly detection

Matrices

(text corpora, recommendations, ...)

- Covariance estimation matrix
- Low rank approximation
- Sparsification

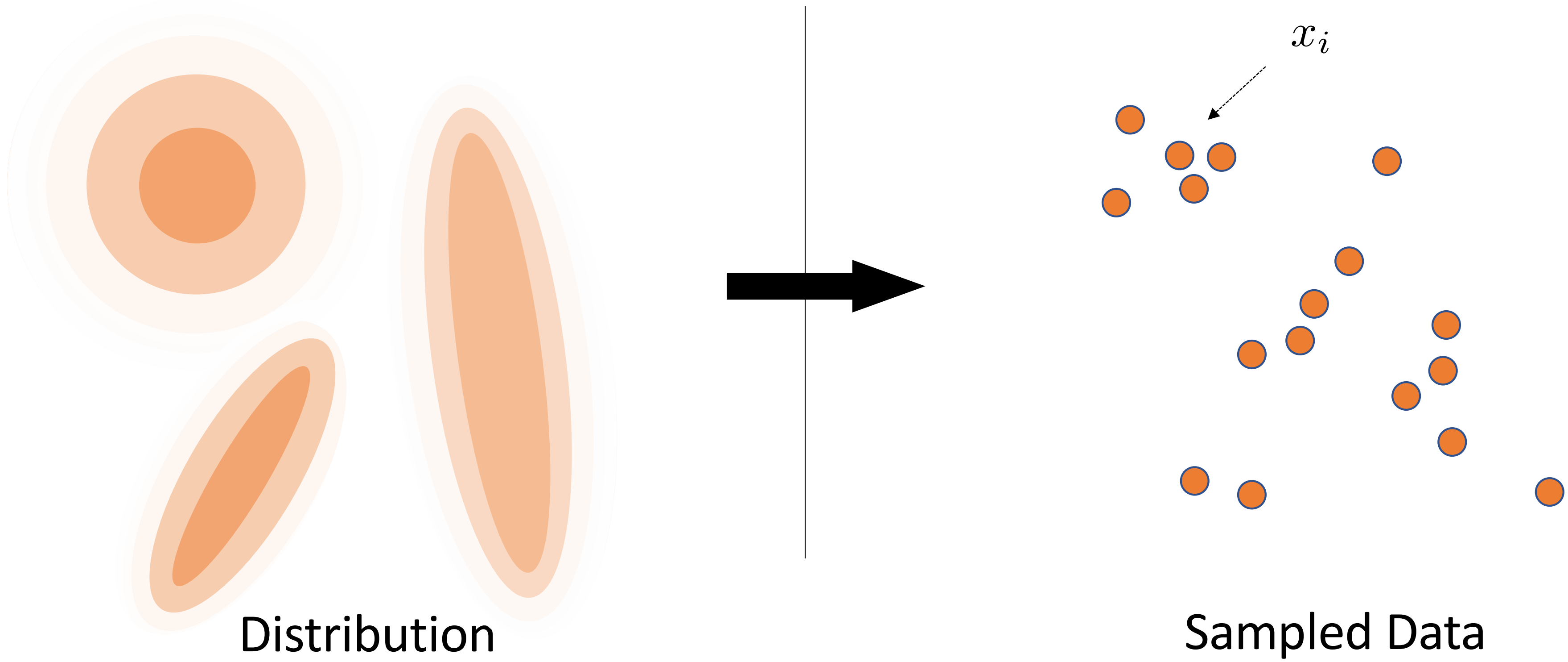
Graphs*

(social networks, communications, ...)

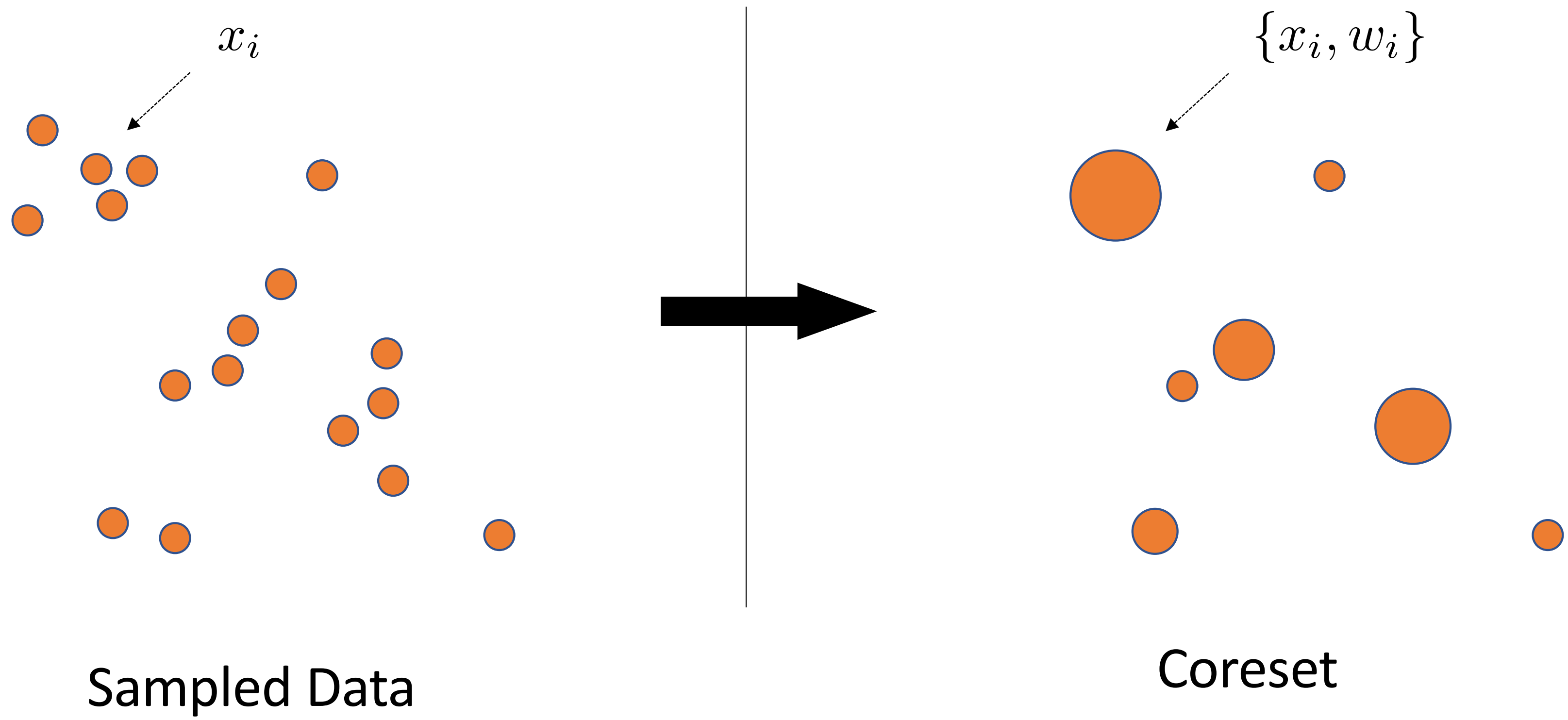
- Connectivity
- Cut Sparsification
- Weighted Matching

Very new results about
Coresets and Machine Learning

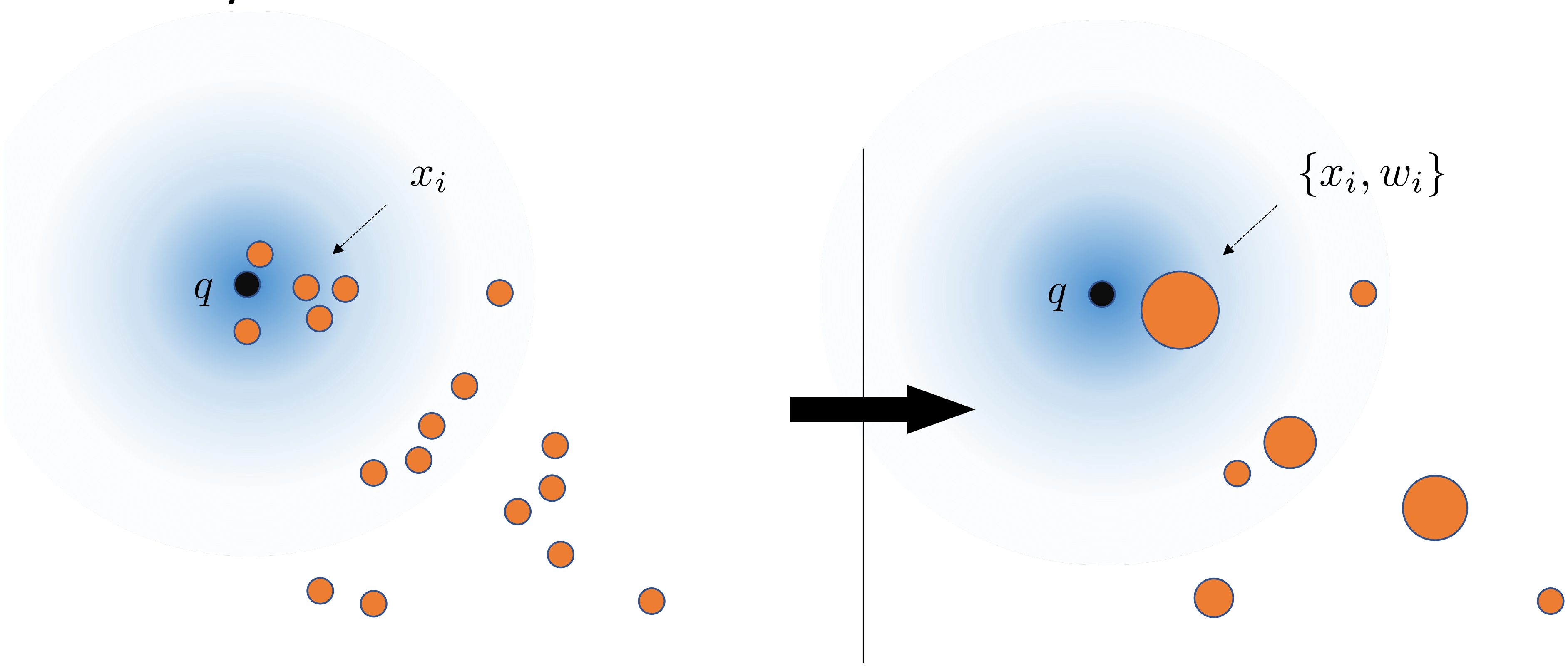
What is PAC learning?



What are Coresets?



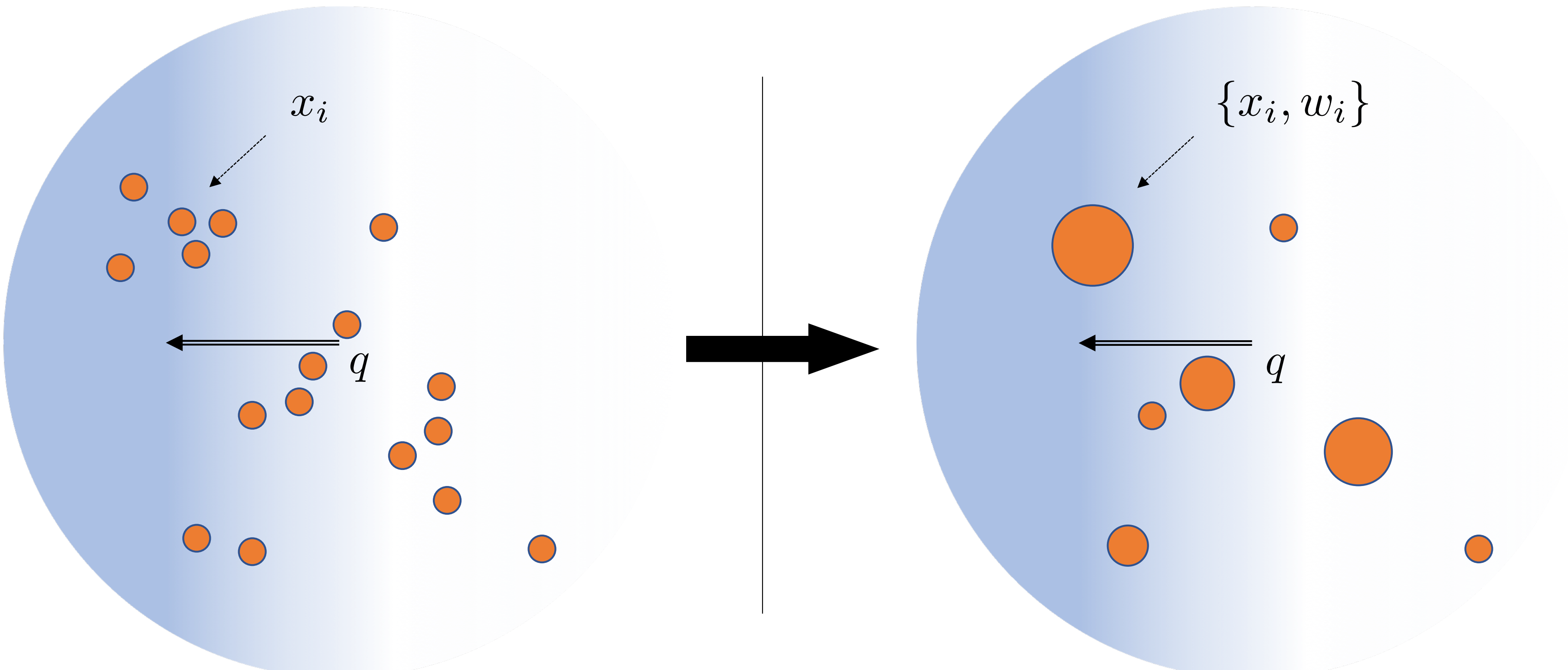
Density Estimation



$F(q) = \sum_i f(x_i, q)$

$\tilde{F}(q) = \sum_{i \in S} w_i f(x_i, q)$

Classification/Regression



$$F(q) = \sum_i f(x_i, q)$$

$$\tilde{F}(q) = \sum_{i \in S} w_i f(x_i, q)$$

ML and Coresets are intimately connected

Rademacher Complexity

$$R_m = \mathbb{E}_\sigma \max_q \frac{1}{m} | \sum_{i=1}^{m} \sigma_i f(x_i, q)|$$

$$K_m \approx O(c/\sqrt{m})$$

Sample Complexity

Model Generalization

Class Discrepancy

$$D_m = \min_\sigma \max_q \frac{1}{m} | \sum_{i=1}^{m} \sigma_i f(x_i, q)|$$

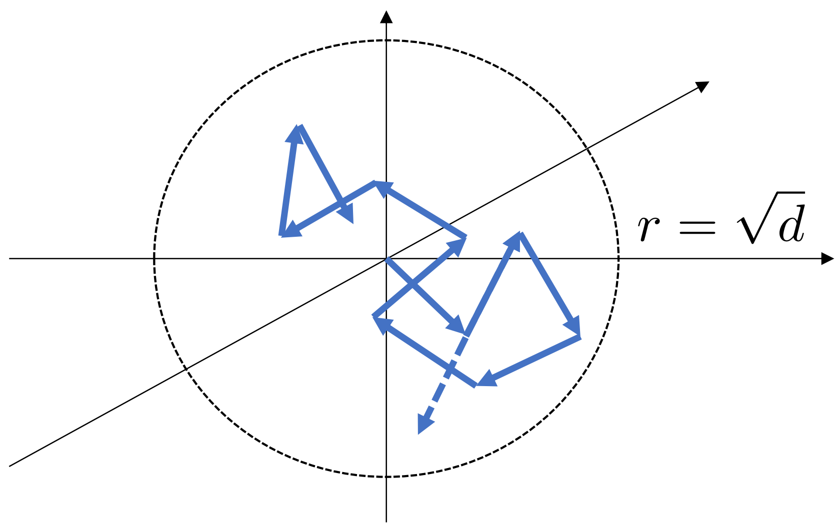
$$D_m = O(c/m)$$

Coreset Complexity

Sketch Generalization

Warmup exercise…

$$\min_{\sigma} \| \sum_{I=1}^{n} \sigma_i x_i \| \leq \underbrace{\sqrt{d}}_{\text{Does not depend on n}}$$



$$\mathbb{E}_{\sigma} \| \sum_{i=1}^{n} \sigma_i x_i \| \approx \sqrt{n}$$

That’s encouraging.....

Universal Vector Balancing Lemma

$x = x^{\otimes 1}$ $\qquad$ $xx^T = x^{\otimes 2}$ $\qquad$ $\mathrm{outer}(x, x, x) = x^{\otimes 3}$

Lemma [Karnin, Liberty, 2019]: For any set of unit vectors $x_i \in \mathbb{R}^d$ there exist signs σ such that for all k simultaneously

$$\left\| \sum_{i=1}^{n} \sigma_i x_i^{\otimes k} \right\| \leq \underbrace{\sqrt{d} \cdot \mathrm{poly}(k)}_{\text{Still does not depend on n !}}$$

Results

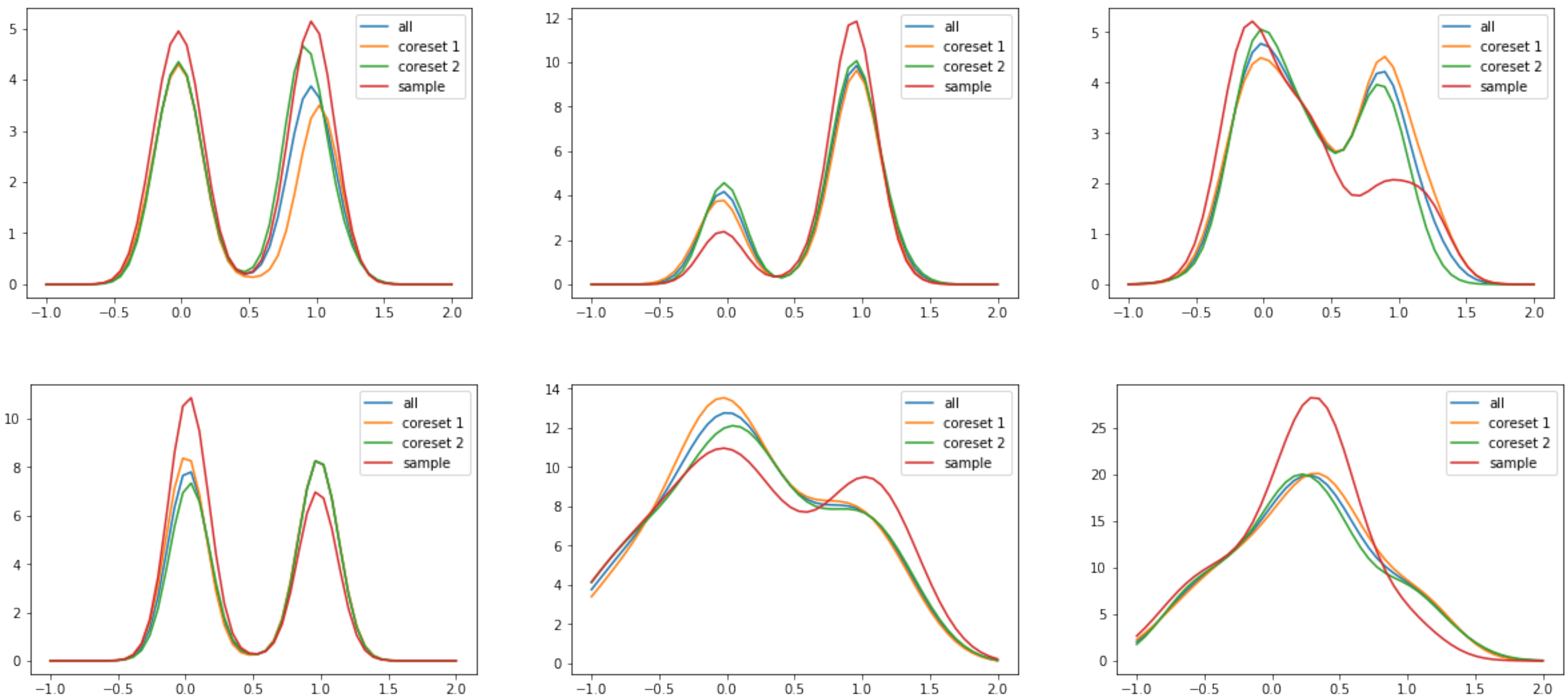
Resolves the open problem
See Philips an Tai 2018

- Sigmoid Activation Regression, Logistic Regression
- Covariance approximation, Graph Laplacians Quadratic forms
- Gaussian Kernel Density estimation

All have the above have Class Discrepancy of $D_m = O(\sqrt{d}/m)$

1) coresets of size $O(\sqrt{d}/\varepsilon)$
2) Streaming Coresets of size $O\left(\sqrt{d}/\varepsilon \cdot \log^2\left(\varepsilon n/\sqrt{d}\right)\right)$
3) Randomized Streaming Coresets of size $O\left(\sqrt{d}/\varepsilon \cdot \log^2 \log(|Q_\varepsilon|/\delta)\right)$

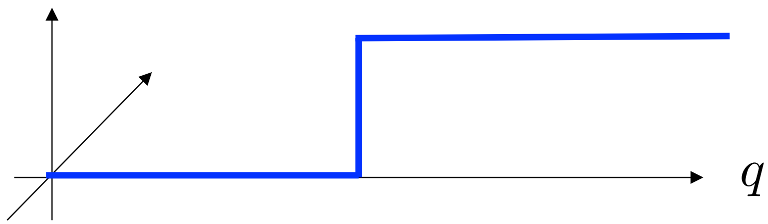
Results for density estimation



There is still a lot of work...

Classification with 0-1 loss

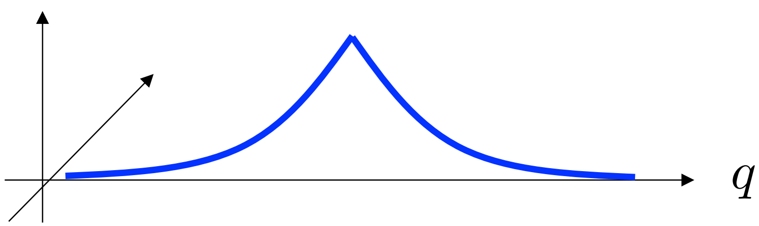
$$f(x,q) = \begin{cases} 1 & \text{if } \langle q, x \rangle > 0 \\ 0 & \text{else} \end{cases}$$



$$D_m = \ ?$$

Exponential Kernel Density

$$f(x,q) = \exp(-\|x - q\|)$$



$$D_m = \ ?$$

</slides>

Pankaj K Agarwal, Sariel Har-Peled, and Kasturi R Varadarajan. Geometric approximation via coresets. *Combinatorial and computational geometry*, 52:1–30, 2005.

Jeff M Phillips. *Small and stable descriptors of distributions for geometric statistical problems*. PhD thesis, 2009.

Jeff M. Phillips and Wai Ming Tai. Improved coresets for kernel density estimates. *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Dis- crete Algorithms, SODA 2018*

Dan Feldman and Michael Langberg. A unified framework for approximating and clustering data. 2011

Jeff M. Phillips and Wai Ming Tai. Near-optimal coresets of kernel density estimates

Elad Tolochinsky and Dan Feldman. Coresets for monotonic functions with applications to deep learning.

Sariel Har-Peled, Dan Roth, and Dav Zimak. Maximum margin coresets for active and noise tolerant learning. *IJCAI 2007*

Sariel Har-Peled and Akash Kushal. Smaller coresets for k-median and k-means clustering. T*he21st ACM Symposium on Computational Geometry, Pisa, Italy, June 6-8, 2005*

Gurmeet Singh Manku, Sridhar Rajagopalan, and Bruce G. Lindsay. Random sampling techniques for space efficient online computation of order statistics of large datasets.

Alexander Munteanu, Chris Schwiegelshohn, Christian Sohler, and David P. Woodruff. On coresets for logistic regression.

Zohar S. Karnin, Kevin J. Lang, and Edo Liberty. Optimal quantile approximation in streams. *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016*

Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *J. Mach. Learn. Res.*, 3:463–482, March 2003

Olivier Bachem, Mario Lucic, and Andreas Krause. Practical coreset constructions for machine learning

Wojciech Banaszczyk. Balancing vectors and gaussian measures of n-dimensional convex bodies. *Random Struct. Algorithms*, 12(4):351–360, July 1998