Distance Queries from Sampled Data: Accurate and Efficient

Edith Cohen

Microsoft Research

Microsoft® Research

Sampling in Data Analysis

Data sets can be too large to store, transfer, or quickly and scalably process queries over, even when stored.
But… we are often happier with fast approximate results.

The sample is a small and flexible summary of the data

Scalable but approximate processing of queries posed over the original data.

The **same** sample can be used for many queries, that are not prespecified.

To get the most out of our sampled data we need good estimators.

Data matrix: keys × instances

Sample

Keys: (users, flow keys, origin-dest pairs…)

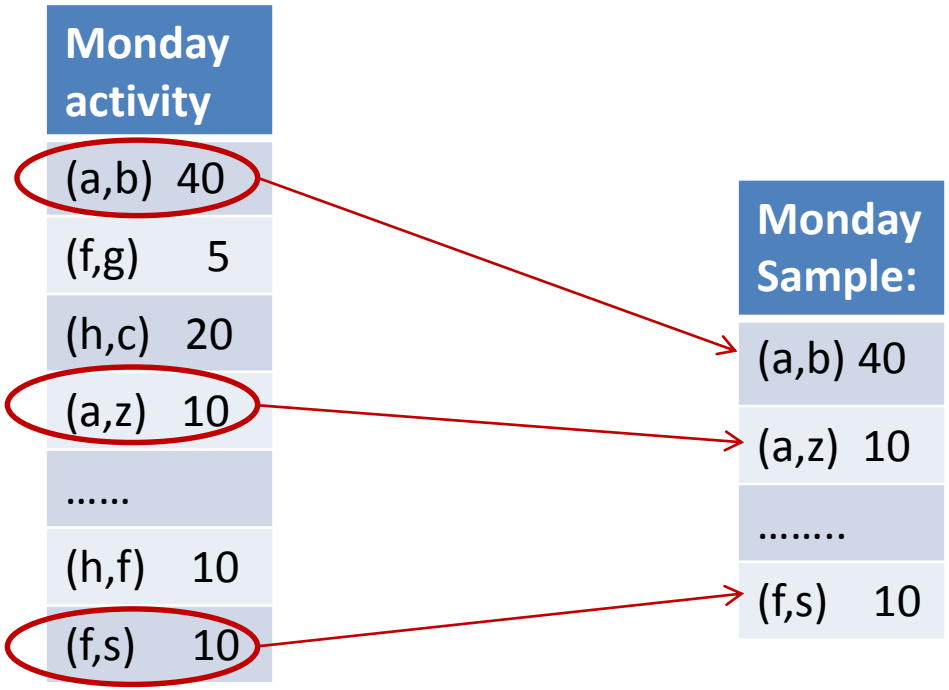
Instances (days, servers, locations, movies)

	Su	Mo	Tu	We	Th	Fr	Sa
a	40	30	10	43	55	30	20
g	0	5	0	0	4	0	10
c	5	0	0	60	3	0	2
z	20	10	5	24	15	7	4
f	0	7	6	3	8	5	20
s	0	0	0	20	100	70	50
d	13	10	8	0	0	5	6

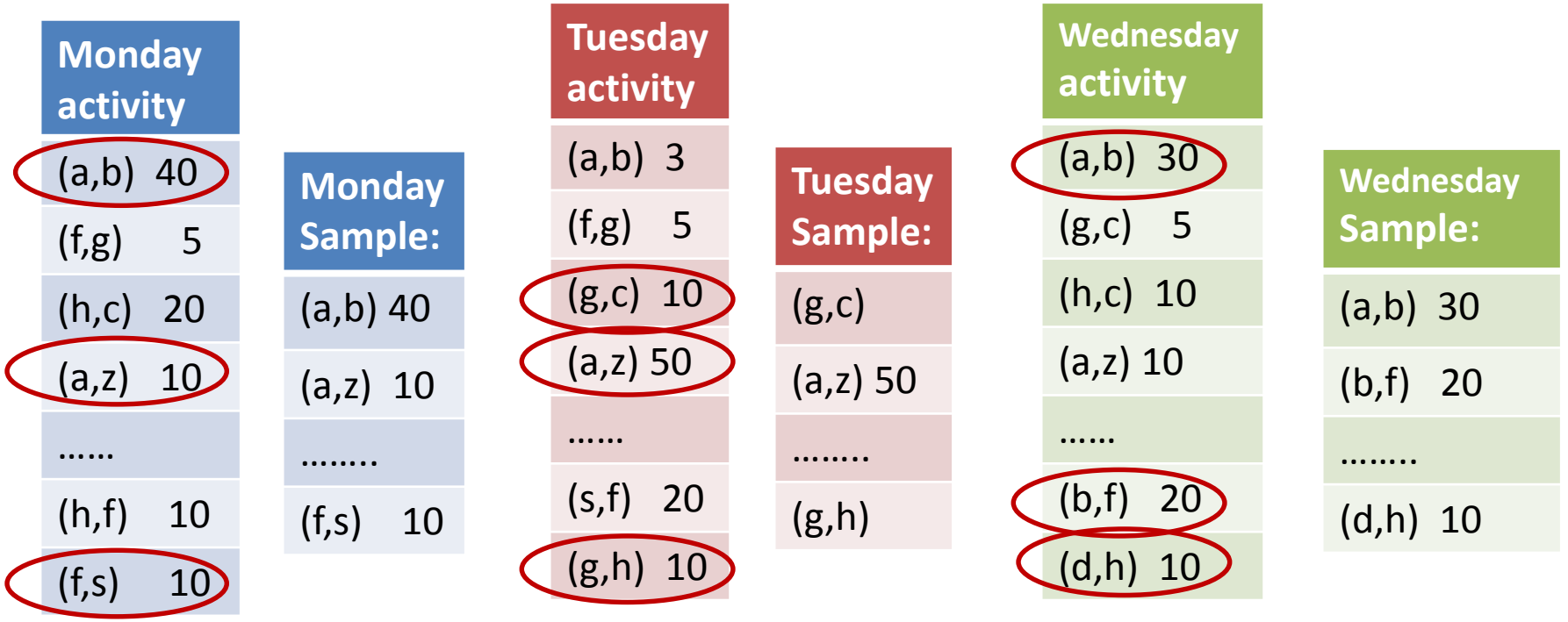
Example: Social/Communication data

keys *are node pairs* $h = (b, c)$

value $v(h)$*: activity (#messages)*



Data from multiple days



Matrix view keys × instances

In our example: keys (a,b) are user-user pairs.
Instances are days.

	Su	Mo	Tu	We	Th	Fr	Sa
(a,b)	40	30	10	43	55	30	20
(g,c)	0	5	0	0	4	0	10
(h,c)	5	0	0	60	3	0	2
(a,z)	20	10	5	24	15	7	4
(h,f)	0	7	6	3	8	5	20
(f,s)	0	0	0	20	100	70	50
(d,h)	13	10	8	0	0	5	6

Common Queries

- **Domain (subset) queries**: sum over (a function $f(x) \geq 0$ of) *selected* entries A_{hj} of $f(A_{hj})$

  In example: "Total activity from users in California to users in New York on Mon-Tue."

  Applications: monitoring, planning, billing.

- **Complex queries,** such as **distance**: use relations between entries.

  In example: change in communication patterns to detect and localize anomalies, events, new trends

Domain (subset) Queries

- Select (key, instance) pairs (domain) $\boldsymbol{D}$

$$\sum_{(h,i)\in D} f(A_{hi})$$

Domain (subset) Queries

- Select (key, instance) pairs (domain) $\boldsymbol{D}$

keys

	Su	Mo	Tu	We	Th	Fr	Sa
a	40	30	10	43	55	30	20
g	0	5	0	0	4	0	10
c	5	0	0	60	3	0	2
z	20	10	5	24	15	7	4
f	0	7	6	3	8	5	20
s	0	0	0	20	100	70	50
d	13	10	8	0	0	5	6

- L_1 (sum) answer: $0 + 60 + 3 + 5 + 24 + 15 = 107$

Horvitz Thompson Estimator (1952) for Domain queries

Select key, instance pairs H, $f(x) \geq 0$

$$Q = \sum_{(h,i)\in H} f(A_{hi})$$

$$\hat{Q} = \sum_{(h,i)\in H \cap \boldsymbol{S}} f(A_{hi})/p_{hi}$$

Unbiased if each $(h,i) \in H$ with $f(A_{hi}) > 0$ has a positive probability p_{hi} to be sampled, and p_{hi} can be computed for selected sampled entries

Performance (variance, concentration) depends on sampling scheme, but "optimal" given the scheme.

HT estimator for Domain Queries

- **Unbiased**: important because bias adds up and we are estimating sums
- **Nonnegative**: important when $f \geq 0$
- **Monotone:** more information $\Rightarrow$ higher estimate
- **Sum form**: Applied entry by entry

- **Optimal:** UMVU The unique minimum variance (unbiased, nonnegative, sum) estimator

What about distance queries?

Distance Queries

- Select keys H
- Select two instances (days) j, k

	Su	Mo	Tu	We	Th	Fr	Sa
a	40	30	10	43	55	30	20
g	0	5	0	0	4	0	10
c	5	0	0	60	3	0	2
z	20	10	5	24	15	7	4
f	0	7	6	3	8	5	20
s	0	0	0	20	100	70	50
d	13	10	8	0	0	5	6

Distance Queries

- Select keys H
- Select two instances (days) j, k

L_p^p distance $\sum_{h \in H} \left| A_{hj} - A_{hk} \right|^p$

Distance Queries

L_p^p distance $\sum_{h \in H} |A_{hj} - A_{hk}|^p$

p = 2: $(4-0)^2 + (24-15)^2 + (8-3)^2 = 122$

	Su	Mo	Tu	We	Th	Fr	Sa
a	40	30	10	43	55	30	20
g	0	5	0	0	4	0	10
c	5	0	0	60	3	0	2
z	20	10	5	24	15	7	4
f	0	7	6	3	8	5	20
s	0	0	0	20	100	70	50
d	13	10	8	0	0	5	6

Distance Queries

- Select keys H
- Select two instances (days) j, k

? L_p^p distance $\sum_{h \in H} \left| A_{hj} - A_{hk} \right|^p$

? Breakdown: increase decrease

$$\sum_{h \in H} \max\{A_{hj} - A_{hk}, 0\}^p \qquad \sum_{h \in H} \max\{A_{hk} - A_{hj}, 0\}^p$$

We would like to **estimate** the query result from the **sample**

Distance Estimators

Bad news: HT may not work at all and may not be optimal even when it does.

Good news: We derive estimators with the same nice properties (unbiased, nonnegative, sum, admissible) and easy to apply

- Derivation is specific to sampling scheme (its projection on a single key). In particular, different derivations for independent and coordinated samples
- Optimum is not unique

Sampling schemes

We usually want weighted sampling:

- Only sample positive entries.
- Sample larger values with a higher probability.

Most common weighted sampling scheme is **Probability Proportional to Size (*PPS*):**

For $\boldsymbol{\tau} > 0$, each key h gets iid $u(h) \sim U[0,1]$:

$h \in S \leftrightarrow v(h) \geq \boldsymbol{\tau} \cdot u(h)$

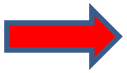
Samples of multiple instances (days)

Coordinated (shared seed): Each key $h = (a, b)$ is sampled with *same seed* $u(h)$ *in different days*

Independent: *independent seed* $u(h, d)$ *in day* d

Independent vs. Coordinated:

- Coordination is better for distance queries; Has LSH property: similar instances have similar samples
- Independent is better for domain queries when selection spans multiple instances



We derive estimators for both schemes

Coordinated Sampling of Instances

PPS sample $\tau = 100$

u		Su	Mo	Tu	We	Th	Fr	Sa
0.33	a	40	30	35	43	55	30	20
0.22	g	0	5	0	0	4	0	10
0.82	c	5	0	70	60	3	0	2
0.16	z	20	10	17	24	15	7	4
0.92	f	0	7	6	3	8	5	20
0.16	s	0	0	0	10	100	70	50
0.77	d	13	10	8	0	0	5	6

Similar instances have similar samples

Independent Sampling of Instances

PPS sample $\tau = 100$

u: Su	u: M	u: Tu
0.33	0.48	0.63
0.22	0.10	0.47
0.82	0.02	0.33
0.16	0.76	0.10
0.92	0.63	0.50
0.16	0.10	0.56
0.77	0.04	0.85

…

	Su	Mo	Tu	We	Th	Fr	Sa
a	40	30	35	43	55	30	20
g	0	5	0	0	4	0	10
c	5	0	70	60	3	0	2
z	20	10	17	24	15	7	4
f	0	7	6	3	8	5	20
s	0	0	0	10	100	70	50
d	13	10	8	0	0	5	6

Our Estimators (coordinated samples)

The L* estimator:

- The unique admissible **monotone** sum estimator
- Optimized for large distances
- 4-variance competitive (2, 2.5 for L_1 and L_2^2)

The U* estimator:

- Optimized for small distances

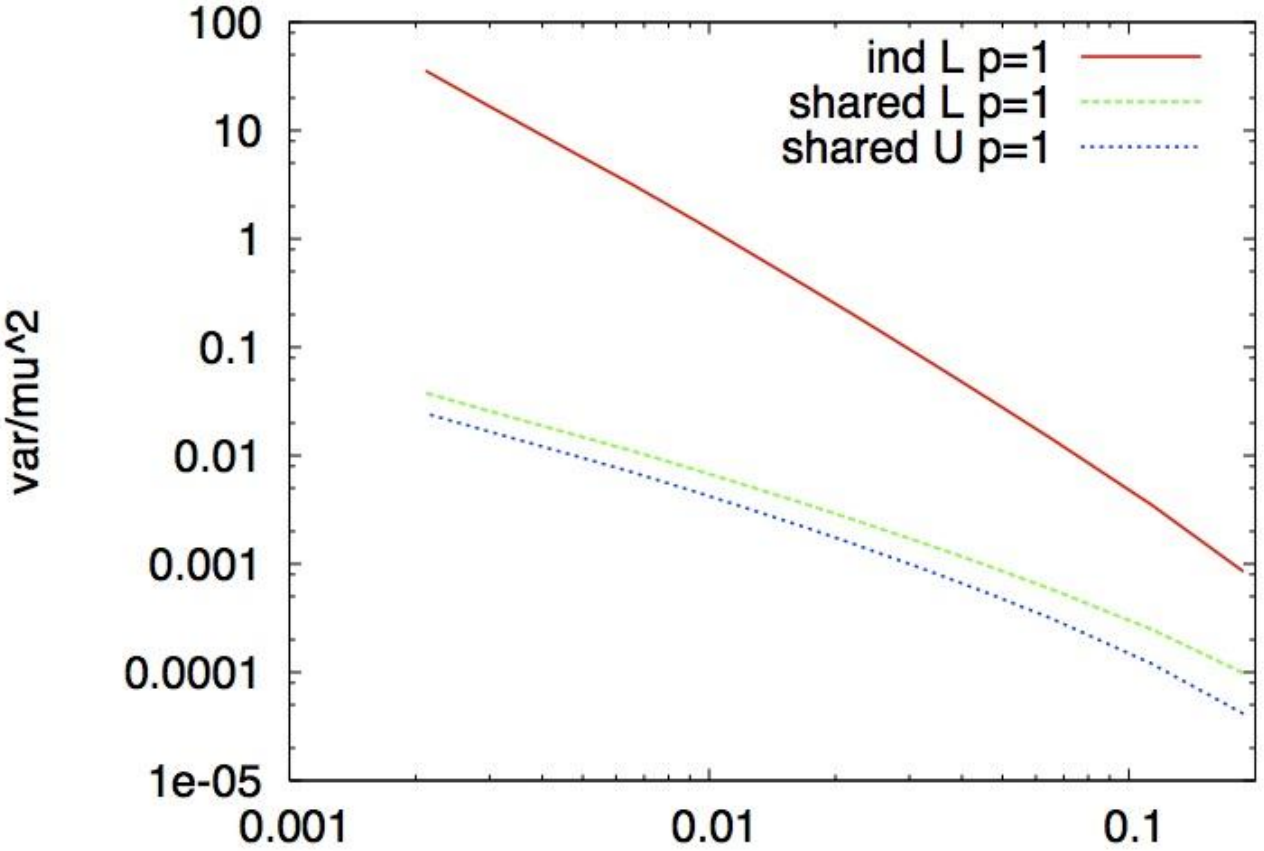
Our Estimators (Independent samples)

The L* estimator: The unique admissible **monotone** symmetric sum estimator

L_1 distance from PPS samples

Keys: destIP Instances: Time periods Value: #IP flows to key

Query Selection: IPprefix (3.8×10^4 active keys total, 65% active in each period)



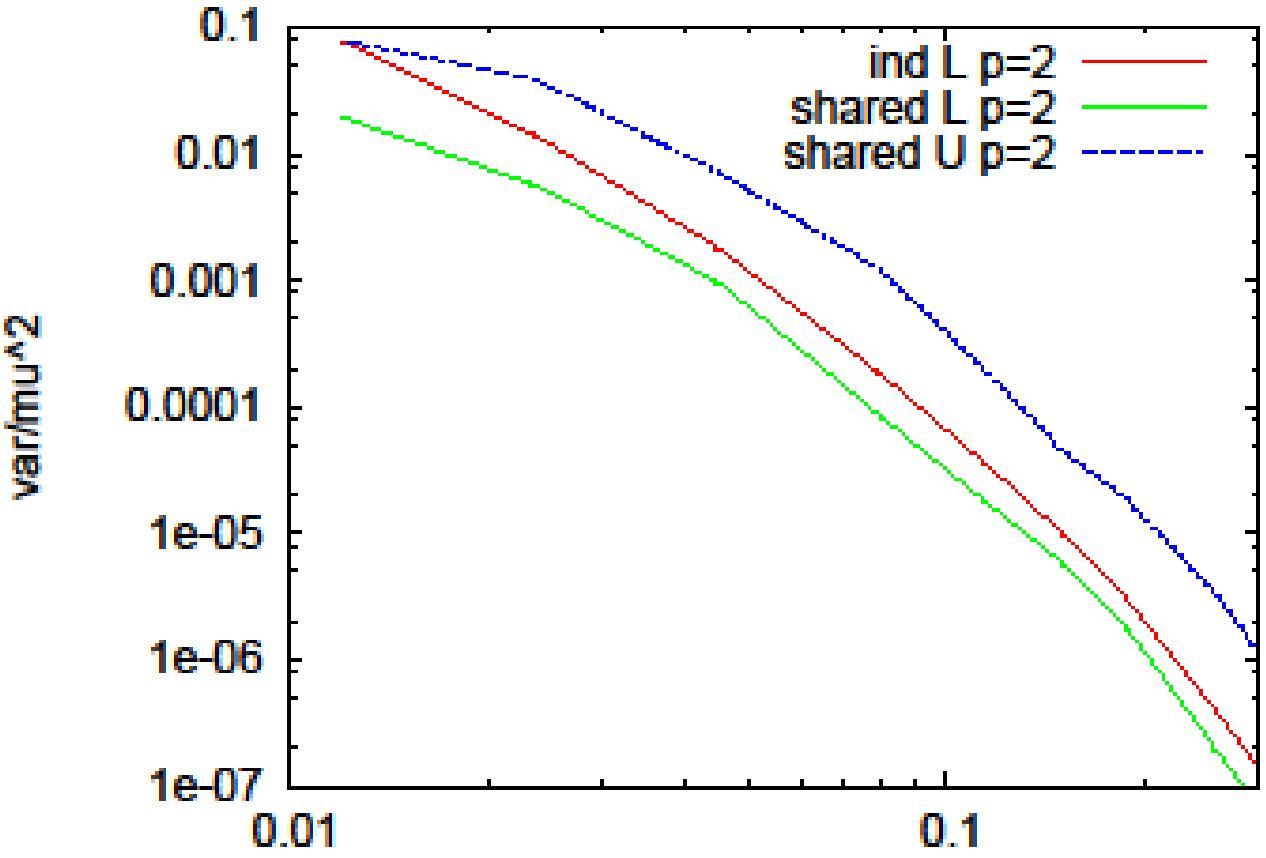
from active keys (nonzero entries)

L_2^2 distance from PPS samples

Keys: terms Instances: years

Value: #occ of term in books that year (Google ngrams data)

Query Selection: 20K most common surnames, years: 2007 2008



from active keys (nonzero entries)

Summary

- Data sets in matrix form instances × keys
- Data is sampled
  - independent or coordinated sampling of instances.
  - weighted (inclusion probability depends on value)
- We derive admissible (Pareto optimal) unbiased and nonnegative sum estimators for L_p^p

Extension: Flexible and general estimation techniques apply to other sampling schemes and other queries.

Thank you!

Distance Queries from Sampled Data: Accurate and Efficient

	Su	Mo	Tu	We	Th	Fr	Sa
a	40	30	10	43	55	30	20
z	0	5	0	0	4	0	10
h	5	0	0	60	3	0	2
b	20	10	5	24	15	7	4
f	0	7	6	3	8	5	20
g	0	0	0	20	84	70	50
d	13	10	8	0	0	5	6

DATA

↓ sampling

DATA

$$\sum_{h \in D} |v_{hi} - v_{hj}|^p \quad ?$$

Query(DATA)

↓ Estimation

$\widehat{\text{Query}}$(DATA)

Microsoft Research

Edith Cohen