Getting the Most out of Your Sample

Edith Cohen

Haim Kaplan

Tel Aviv University

Why data is sampled

- Lots of Data: measurements, Web searches, tweets, downloads, locations, content, social networks, and all this both historic and current…
- To get value from data we need to be able to process queries over it
- But resources are constrained: Data too large to: transmit, store in full, to process even if stored in full…

Random samples

- A compact synopsis/summary of the data. Easier to store, transmit, update, and manipulate
- Aggregate queries over the data can be approximately (and efficiently) answered from the sample
- Flexible: Same sample supports many types of queries (which do not have to be known in advance)

Queries and estimators

The value of our sampled data hinges on the quality of our estimators

- Estimation of some basic (sub)population statistics is well understood (sample mean, variance,…)
- We need to better understand how to estimate other basic queries:
  - *difference norms* (anomaly/change detection)
  - *distinct counts*

Our Aim

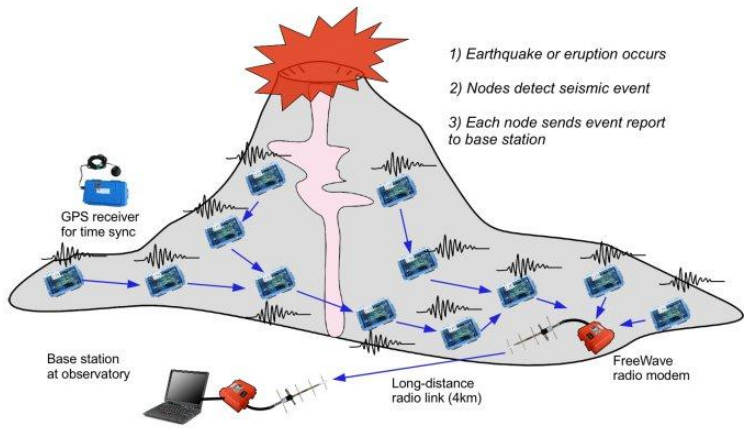
We want to estimate sum aggregates *over sampled data*

$$\sum_h f(v_1(h), v_2(h), ...)$$

- Classic sampling schemes
- Seek "variance optimal" estimators
- Understand (and meet) limits of what can be done

Example: sensor measurements

time	sensor1	sensor2
7:00	3	4
7:01	13	5
7:02	8	12
7:03	4	13
7:04	2	9
7:05	1	3
7:06	10	5
7:07	6	14
7:08	13	11
7:09	12	13
7:10	6	7
7:11	20	21
7:12	10	10



Could be

- radiation/pollution levels
- sugar levels
- temperature readings

Sensor measurements

time	sensor1	sensor2	max
7:00	3	4	**4**
7:01	13	5	**13**
7:02	8	12	**12**
7:03	4	13	**13**
7:04	2	9	**9**
7:05	1	3	**3**
7:06	10	5	**10**
7:07	6	14	**14**
7:08	13	11	**13**
7:09	12	13	**13**
7:10	6	7	**7**
7:11	20	21	**21**
7:12	10	10	**10**

We are interested in the maximum reading in each timestamp

Sensor measurements: Sum of maximums over selected subset

time	sensor1	sensor2	max
7:00	3	4	4
7:01	13	5	13
7:02	8	12	12
7:03	4	13	13
7:04	2	9	9
7:05	1	3	**3**
7:06	10	5	**10**
7:07	6	14	**14**
7:08	13	11	**13**
7:09	12	13	**13**
7:10	6	7	**7**
7:11	20	21	**21**
7:12	10	10	**10**

$$\sum_{t1}^{t2} \max(sensor1, sensor2)$$

But we only have sampled data

time	sensor1	sensor2	max
7:00	3	4	4
7:01		5	?
7:02	8	12	12
7:03	4	13	13
7:04			?
7:05	1		?
7:06		5	?
7:07	6		?
7:08		11	?
7:09	12	13	13
7:10			?
7:11		21	?
7:12	10		?

Example: distinct count of IP addresses

Interface1:

Active IP addresses
132.169.1.1
216.115.108.245
……
132.67.192.131
170.149.173.130
74.125.39.105

Interface2:

Active IP addresses
132.169.1.1
216.115.108.245
74.125.39.105
……
87.248.112.181
74.125.39.105

Distinct Count Queries

Interface1

Interface2

Query: How many distinct IP addresses from a certain AS accessed the router through one of the interfaces during this time period ?

I1 Active IP addresses
132.169.1.1 ★
216.115.108.245
……
132.67.192.131 ★
170.149.173.130
74.125.39.105

I2 Active IP addresses
132.169.1.1 ★
216.115.108.245
74.125.39.105
……
87.248.112.181
74.125.39.105

Distinct in AS
132.169.1.1
132.67.192.131

This is also a sum aggregate

$$\sum_{IP\ addresses \in AS} OR(\in \text{interface1}, \in \text{interface2})$$

We do not have all IP addresses going through each interface but just a sample

Example: Difference queries
IP traffic between time periods

day1

IP prefix	kBytes 03/05/2012
132.169.1.0/24	2534875
216.115.0.0/16	4326
132.67.192.0/20	6783467
170.149.173.130	784632
74.125.39.105/10	2573580

day2

IP prefix	kBytes 03/06/2012
132.169.1.0/24	4679235
216.115.0.0/16	1243
74.125.39.105/32	462534
87.248.112.181/20	29865
74.125.39.105/10	4572456

Difference Queries

Query: Difference between day1 and day2

$$\left(\sum_{i \in IP\ prefixes\ in\ AS} |kB_i(day1) - kB_i(day2)|^p \right)^{1/p}$$

Query: Upward change from day1 to day2

$$\left(\sum_{i \in IP\ prefixes\ in\ AS} |kB_i(day1) - kB_i(day2)|_+^p \right)^{1/p}$$

Sum Aggregates

We want to estimate sum aggregates *from samples*

$$\sum_h f(v_1(h), v_2(h), \ldots)$$

Single-tuple "reduction"

To estimate the sum $\sum_h f(v_1(h), v_2(h), \ldots)$
we sum single-tuple estimates

$$\sum_h \hat{f}(v_1(h), v_2(h), \ldots)$$

Almost WLOG since tuples are sampled (nearly) independently

Each tuple estimate $\hat{f}(v_1(h), v_2(h), \ldots)$ has high variance, but relative error of sum of unbiased (pairwise) independent estimates decreases with aggregation

Estimator properties

We seek estimators $\hat{f}(v_1(h), v_2(h), \ldots)$ that are:

must have
- Unbiased
- Nonnegative

either or both
- Pareto-optimal: No estimator has smaller variance on all data vectors (Tailor to data?)
- "Variance competitive": Not far from minimum for all data

Sometimes:
- Monotone: increases with information

Sampling schemes

- Weighted vs. Random Access (weight-oblivious)
  - Sampling probability does/does not depend on value
- Independent vs. Coordinated sampling
  - In both cases sampling of an entry is independent of values of other entries.
  - Independent: joint inclusion probability is product of individual inclusion probabilities.
  - Coordinated: sharing random bits so that joint probability is higher
- Poisson / bottom-k sampling

Start with a warm-up: Random Access Sampling

Random Access: Suppose each entry is sampled with probability p

time	sensor1	sensor2	max
7:00	**3**	**4**	**4**
7:01		5	?
7:02	**8**	**12**	**12**
7:03	**4**	**13**	**13**
7:04			?
7:05	1		?
7:06		5	?
7:07	6		?
7:08		11	?
7:09	**12**	**13**	**13**
7:10			?
7:11		21	?
7:12	10		?

$$\sum_{7:00}^{7:12} \max(sensor1, sensor2)$$

We know the maximum only when both entries are sampled = probability p^2

Tuple estimate: $\max/p^2$ if both sampled. 0 otherwise.

$$\text{estimate} = \frac{4+12+13+13}{p^2}$$

It's a sum of unbiased estimators

time	sensor1	sensor2	e(max)
7:00	**3**	**4**	**$4/p^2$**
7:01		5	0
7:02	**8**	**12**	**$12/p^2$**
7:03	**4**	**13**	**$13/p^2$**
7:04			0
7:05	1		0
7:06		5	0
7:07	6		0
7:08		11	0
7:09	**12**	**13**	**$13/p^2$**
7:10			0
7:11		21	0
7:12	10		0

This is the **Horvitz-Thompson** estimator:

$$\sum \frac{\max(v_1, v_2)}{p^2}$$

- Nonnegative
- Unbiased:

$$p^2 \frac{\max(v_1, v_2)}{p^2} + (1 - p^2)0 = \max(v_1, v_2)$$

The weakness which we address:

time	sensor1	sensor2	max
7:00	3	4	4
7:01		5	?
7:02	8	12	12
7:03	4	13	13
7:04			?
7:05	1		?
7:06		5	?
7:07	6		?
7:08		11	?
7:09	12	13	13
7:10			?
7:11		21	?
7:12	10		?

Not optimal

Ignores a lot of information in the sample

Doing the best for equal entries

time	sensor1	sensor2	max
7:00	4	4	4

If none is sampled $(1-p)^2$:

time	sensor1	sensor2	e(max)
7:00	?	?	0

If we sample at least one value $2p-p^2$:

time	sensor1	sensor2	e(max)
7:00	4	?	$4/(2p-p^2)$

time	sensor1	sensor2	e(max)
7:00	?	4	$4/(2p-p^2)$

time	sensor1	sensor2	e(max)
7:00	4	4	$4/(2p-p^2)$

What if entry values are different ?

time	sensor1	sensor2	max
7:00	**3**	4	4

If none is sampled:

time	sensor1	sensor2	e(max)
7:00	?	?	0

If we sample one value:

time	sensor1	sensor2	e(max)
7:00	3	?	$3/(2p-p^2)$

time	sensor1	sensor2	e(max)
7:00	?	4	$4/(2p-p^2)$

If we sample both

time	sensor1	sensor2	e(max)
7:00	3	4	?

Unbiasedness determines value

What if entry values are different ?

time	sensor1	sensor2	e(max)
7:00	?	?	0

time	sensor1	sensor2	e(max)
7:00	3	?	$3/(2p-p^2)$

time	sensor1	sensor2	e(max)
7:00	?	4	$4/(2p-p^2)$

time	sensor1	sensor2	e(max)
7:00	3	4	X

Unbiased: $p(1-p)\frac{3}{2p-p^2} + p(1-p)\frac{4}{2p-p^2} + p^2 X = 4$

$$X = \frac{1}{p^2}\left(4 - \frac{1-p}{2-p}(4+3)\right)$$

X>0 → nonnegative

X> $4/(2p-p^2)$ → monotone

… The L estimator

time	sensor1	sensor2	e(max)
7:00	?	?	0

time	sensor1	sensor2	e(max)
7:00	v_1	?	$v_1/(2p-p^2)$

time	sensor1	sensor2	e(max)
7:00	?	v_2	$v_2/(2p-p^2)$

time	sensor1	sensor2	e(max)
7:00	v_1	v_2	x

$$X = \frac{1}{p^2}\left(v_1 - \frac{1-p}{2-p}(v_1+v_2)\right)$$

- Nonnegative
- Unbiased
- Pareto optimal
  - Min possible variance for two equal entries
- Monotone (unique) and symmetric

Back to our sum aggregate

time	sensor1	sensor2	e(max)
7:00	3	4	$\frac{1}{p^2}\left(4-\frac{1-p}{2-p}(4+3)\right)$
7:01		5	$5/(2p-p^2)$
7:02	8	12	$\frac{1}{p^2}\left(12-\frac{1-p}{2-p}(12+8)\right)$
7:03	4	13	$\frac{1}{p^2}\left(13-\frac{1-p}{2-p}(13+4)\right)$
7:04			0
7:05	1		$1/(2p-p^2)$
7:06		5	$5/(2p-p^2)$
7:07	6		$6/(2p-p^2)$
7:08		11	$11/(2p-p^2)$
7:09	12	13	$\frac{1}{p^2}\left(13-\frac{1-p}{2-p}(13+12)\right)$
7:10			0
7:11		21	$21/(2p-p^2)$
7:12	10		$10/(2p-p^2)$

$$\sum_{7:00}^{7:12} \max(sensor1, sensor2)$$

The L estimator

time	sensor1	sensor2	e(max)
7:00	?	?	0

time	sensor1	sensor2	e(max)
7:00	v_1	?	$v_1/(2p-p^2)$

time	sensor1	sensor2	e(max)
7:00	?	v_2	$v_2/(2p-p^2)$

time	sensor1	sensor2	e(max)
7:00	v_1	v_2	x

$$X = \frac{1}{p^2}\left(v_1 - \frac{1-p}{2-p}(v_1+v_2)\right)$$

- Nonnegative
- Unbiased
- Pareto optimal
  - Min possible variance for two equal entries
- Monotone (unique) and symmetric

The U estimator

Q: What if we want to optimize the estimator for *sparse* vectors ?

U estimator: Nonnegative, Unbiased, symmetric, Pareto optimal, not monotone

time	sensor1	sensor2	e(max)
7:00	?	?	0

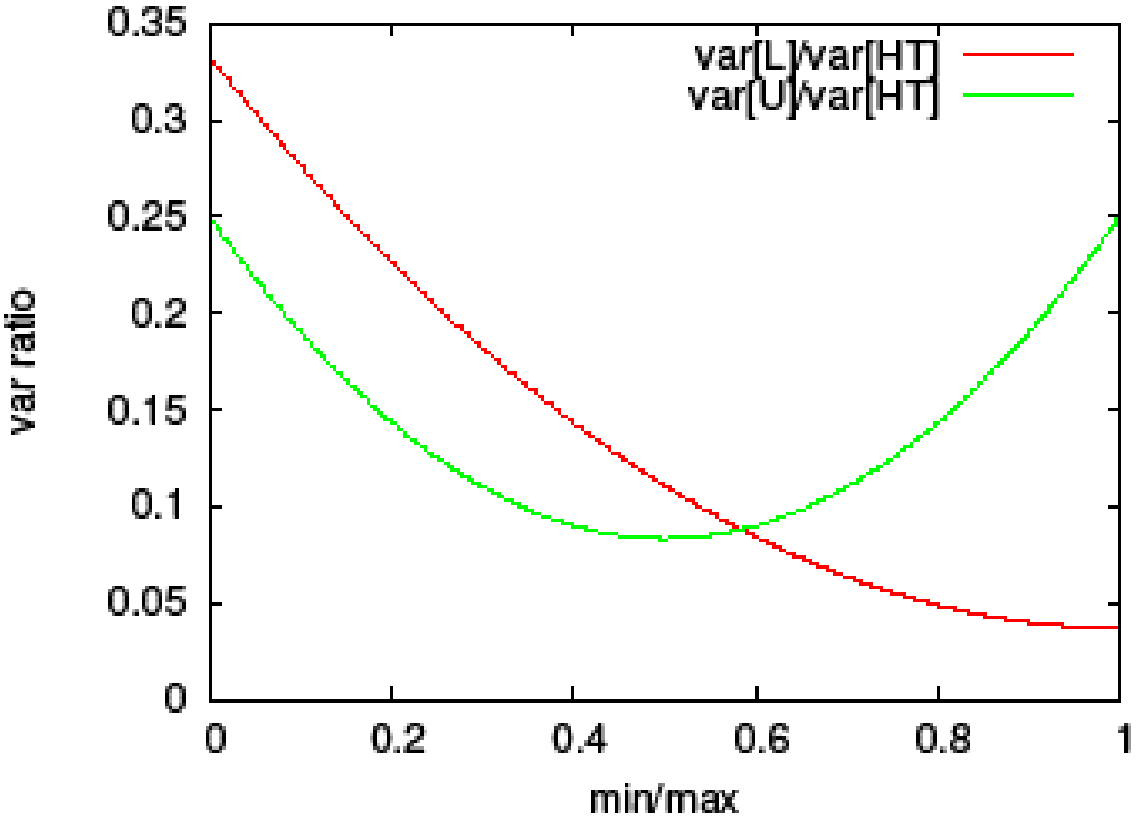
time	sensor1	sensor2	e(max)
7:00	?	v_2	$v_2/(p(1+[1-2p]_+)$

time	sensor1	sensor2	e(max)
7:00	v_1	?	$v_1/(p(1+[1-2p]_+)$

time	sensor1	sensor2	e(max)
7:00	v_1	v_2	$(\max(v_1,v_2)-(v_1+v_2)(1-p)/(1+[1-2p]_+))/p^2$

Variance ratio U/L vs. HT

$p=½$



Order-based variance optimality

Fine-grained tailoring of estimator to data patterns

An estimator is $<$-optimal if any estimator with lower variance on v has strictly higher variance on some $z<v$.

The L max estimator is $<$-optimal for $(v,v) < (v,v-x)$
The U max estimator is $<$-optimal for $(v,0) < (v,x)$

We can construct unbiased $<$-optimal estimators for any other order $<$

Weighted sampling: Estimating distinct count

Interface1:

IP addresses
132.169.1.1
216.115.108.245
132.67.192.131
170.149.173.130
74.125.39.105

Interface2:

IP addresses
132.169.1.1
216.115.108.245
74.125.39.105
87.248.112.181
74.125.39.105

Q: How many distinct IP addresses accessed the router through one of the interfaces ?

$$\sum_{IP\,addresses \in AS} OR(\in \text{interface1}, \in \text{interface2})$$

Sampling is "weighted"

Interface1:

IP addresses
132.169.1.1
216.115.108.245
132.67.192.131
170.149.173.130
74.125.39.105

Interface2:

IP addresses
132.169.1.1
216.115.108.245
74.125.39.105
87.248.112.181
74.125.39.105

If the IP address did not access the interface then we sample it with probability 0, otherwise with probability p

OR estimation

Independent "weighted" sampling with probability p

IP add	Interface1	Interface2	e(OR)
132.169.1.1	?	?	0

To be nonnegative for (0,0)

IP add	Interface1	Interface2	e(OR)
132.169.1.1	1	?	1/p

To be unbiased for (1,0)

IP add	Interface1	Interface2	e(OR)
132.169.1.1	?	1	1/p

To be unbiased for (0,1)

IP add	Interface1	Interface2	e(OR)
132.169.1.1	1	1	x

To be unbiased for (1,1)

$p^2 x + 2p(1-p)/p=1 \rightarrow x=(2p-1)/ p^2 < 0$

→ No unbiased nonnegative estimator when $p<\frac{1}{2}$

Negative result (??)

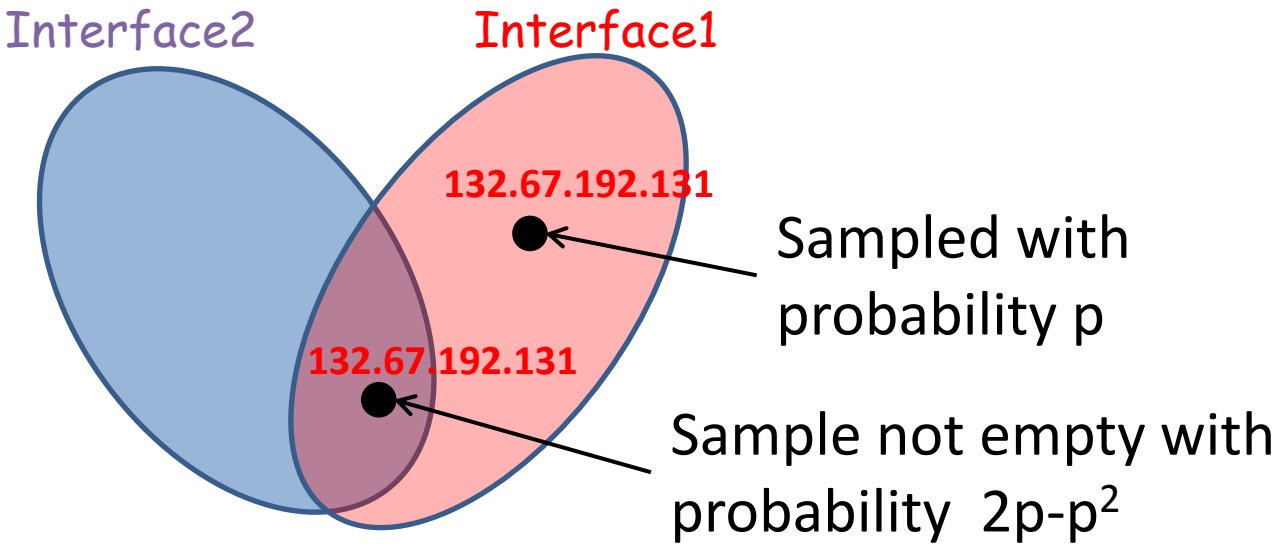
Independent "weighted" sampling: There is no non-negative unbiased estimator for OR

(related to M. Charikar, S. Chaudhuri, R. Motwani, and V. Narasayya (PODS 2000), negative results for distinct element counts)

Same holds for other functions like the *ℓ-th largest value* ($\ell < d$), and *range (Lp difference)*

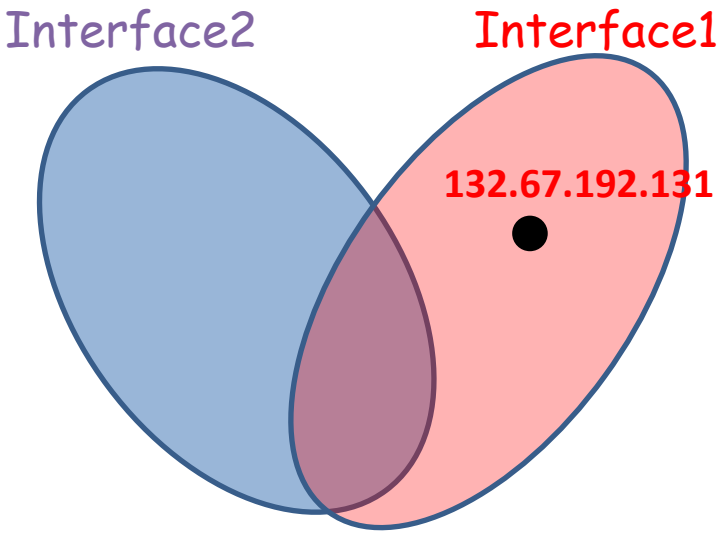
Known seeds: Same sampling scheme but we make random bits "public:" can sometimes get information on values also when entry is not sampled (and get good estimators!)

Estimate OR



How do we know if **132.67.192.131** **did not occur** in interface2 or **was not sampled** by interface2 ?

Known seeds



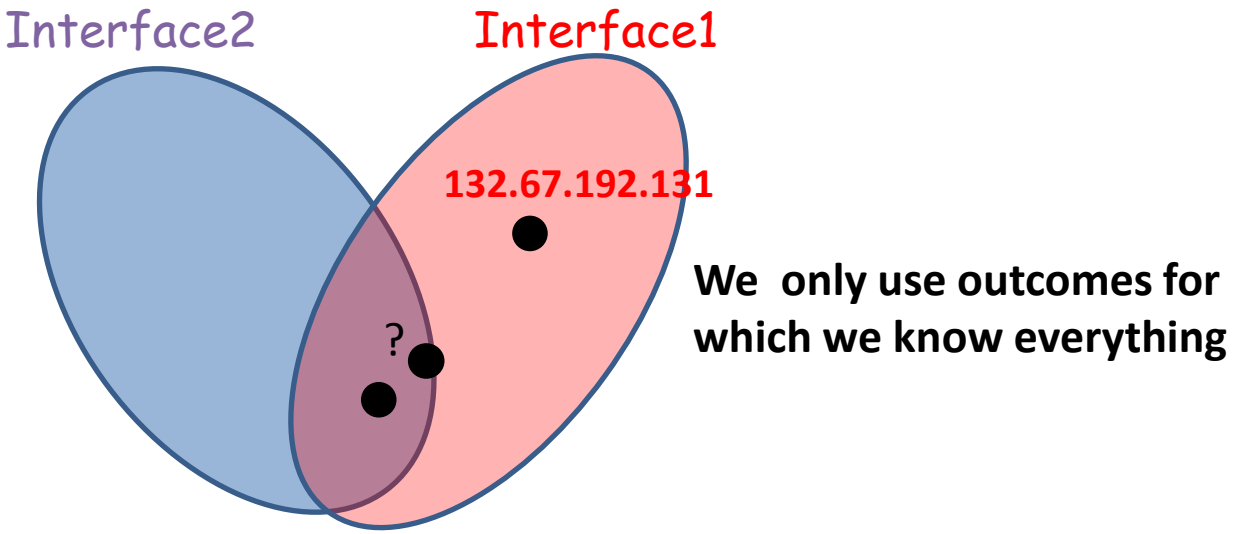
Interface1 samples active IP iff H_1(132.67.192.131) < p

Interface2 samples active IP iff H_2(132.67.192.131) < p

H_2(132.67.192.131) < p ➔ 132.67.192.131 ∉ interface2

H_2(132.67.192.131) > p ➔ We do not know

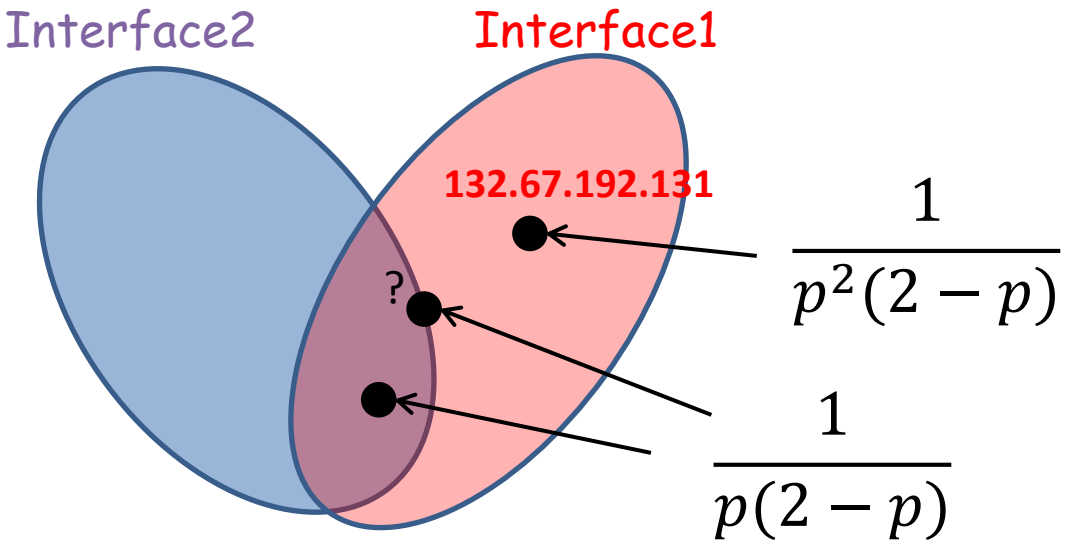
HT estimator+known seeds



With probability p^2, $H_1(132.67.192.131) < p$ and $H_2(132.67.192.131) < p$.
We know if 132.67.192.131 occurred in both interfaces. If sampled in either interface then HT estimate is $1/p^2$.

$H_2(132.67.192.131) > p$ or $H_1(132.67.192.131) > p$, the HT estimate is 0

Our L estimator:



- Nonnegative
- Unbiased
- Monotone (unique) and symmetric
- Pareto optimal
  - Min possible variance for (1,1) (both interfaces are accessed)

Unbiased

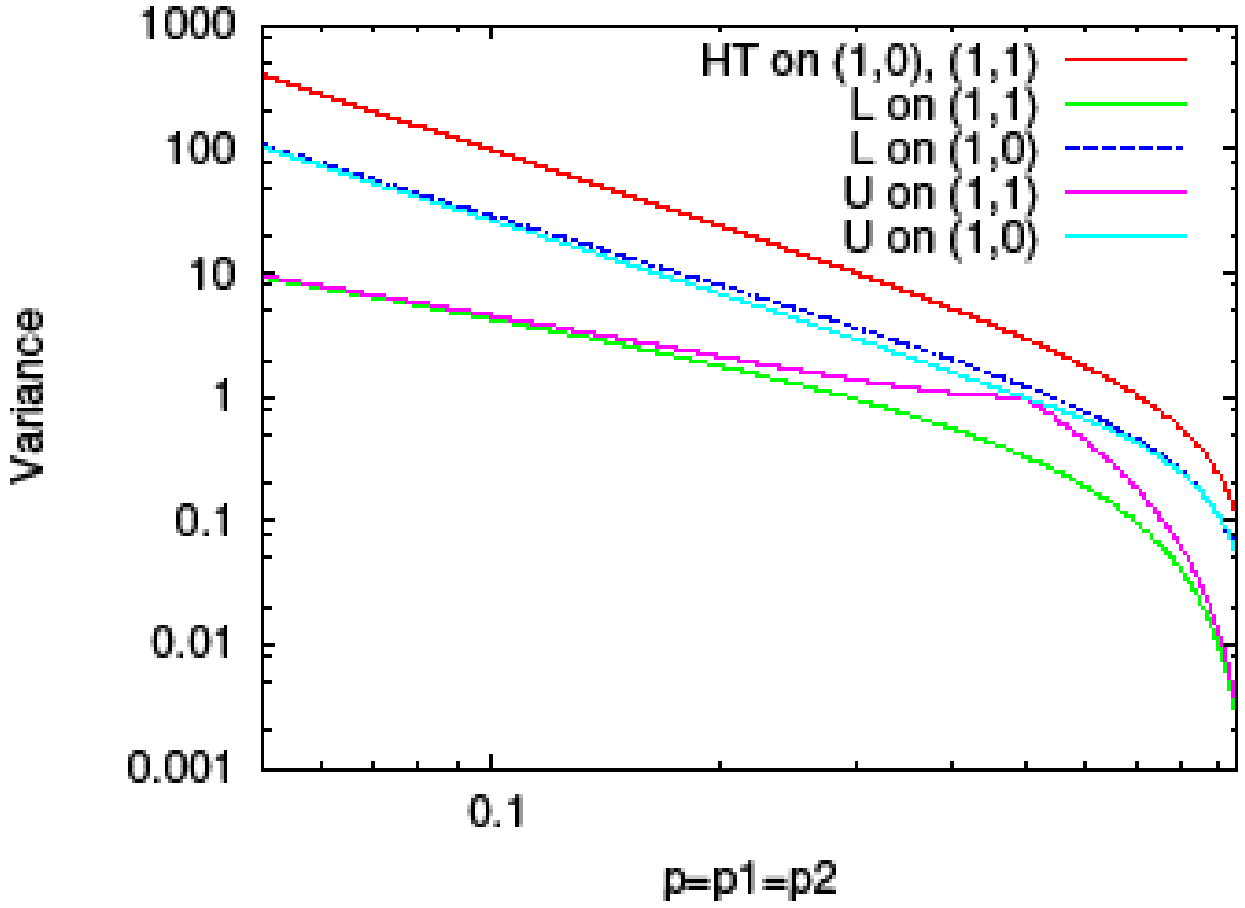
For items in the intersection (minimum possible variance):

$$(2p - p^2)\frac{1}{2p - p^2} = 1$$

For items in one interface:

$$p(1-p)\frac{1}{2p - p^2} + p^2\frac{1}{p(2p - p^2)} = 1$$

OR (ind weighted+known seeds) variance of L, U, HT



Independent Sampling + Known Seeds

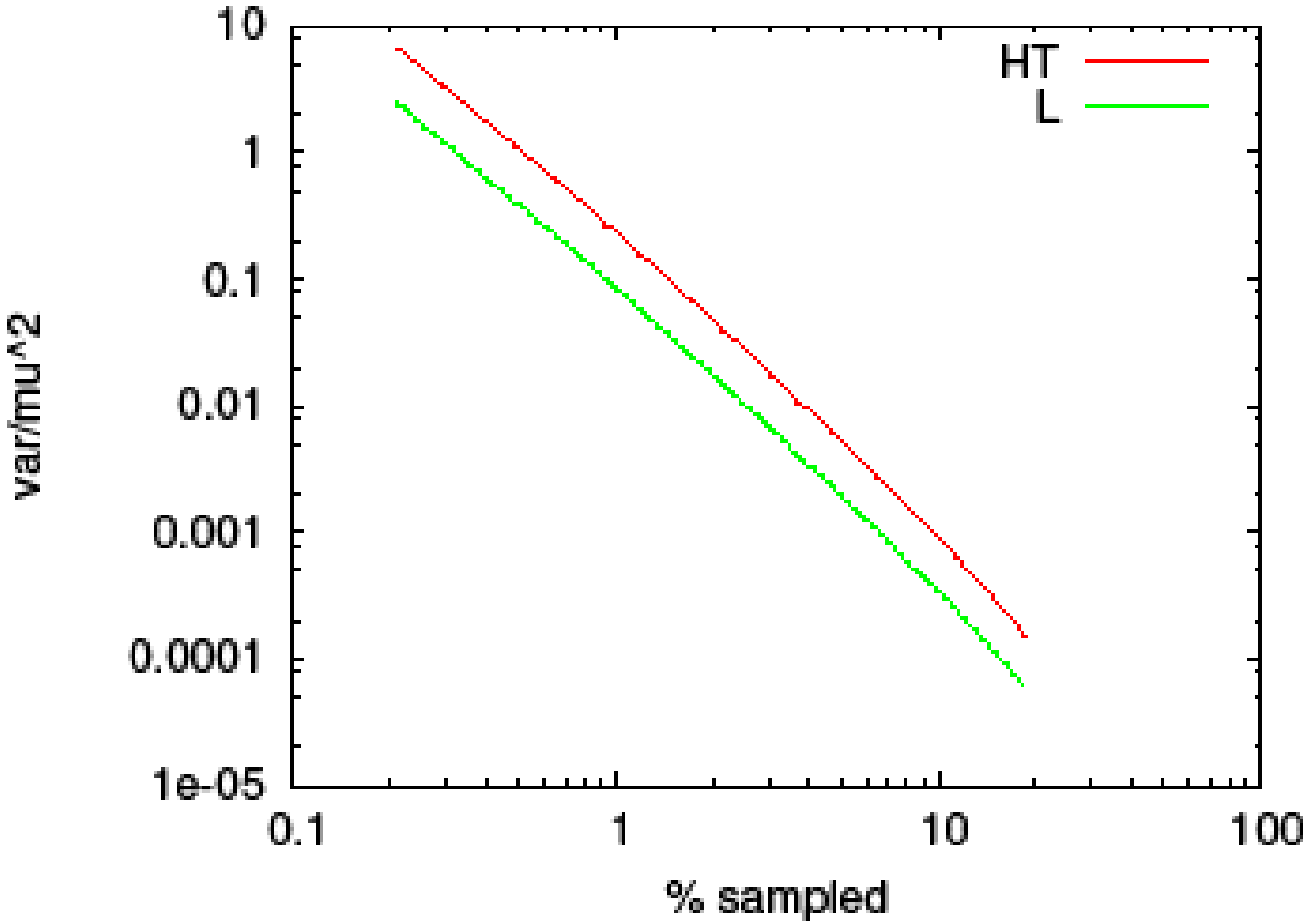
Results:

- Method to construct unbiased <-optimal estimators.
  - Nonnegative either through explicit constraints or smart selection of "<" on vectors.
- L estimator: use an order < such that <-optimal estimator is nonnegative.
  - Maximum $\max(v_1, v_2)$
  - Exponentiated range $|v_1 - v_2|^p$ (sum aggregate is L_p^p)

Take home: use known seeds with your classic weighted sampling scheme

Estimating max sum: Independent, pps, known seeds

IP flows to dest IP address in two one-hour periods:



Coordinated Sampling

Shared seeds coordinated sampling: Seeds are known and *shared across instances*:

- More similar instances have more similar samples.
- Allows for tighter estimates of some functions (difference norms, distinct counts)

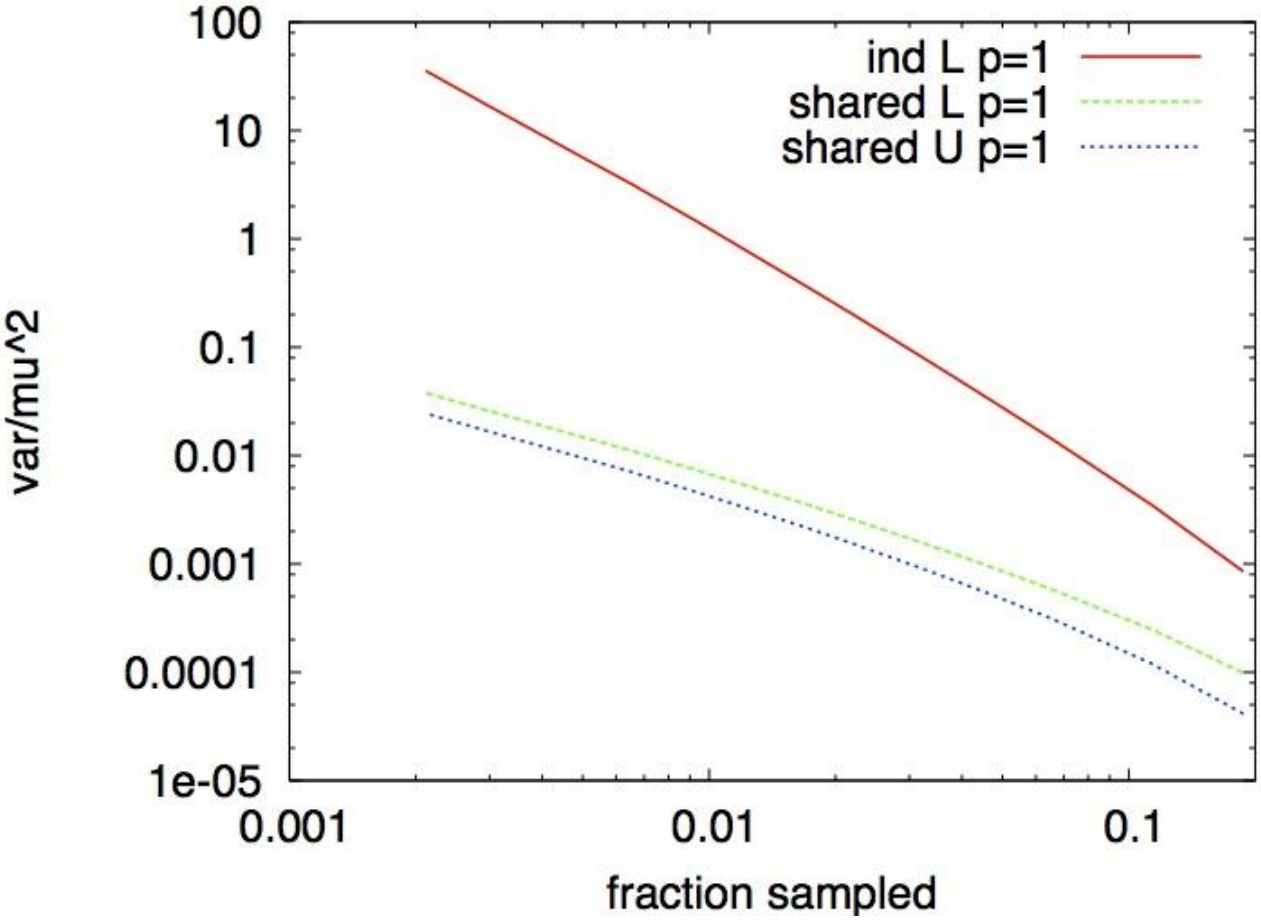
Results:

- Precise characterization of functions for which nonnegative and unbiased estimators exist. (Also, bounded, bounded variances)
- L estimator: nonnegative, unbiased, monotone both variance+ optimal and 4-competitive (on all data, all functions $E[\hat{f}^2]$ at most 4 times minimum possible)
- Construction of nonnegative order-optimal estimators

Estimating L_1 difference

Independent / Coordinated, pps, known seeds

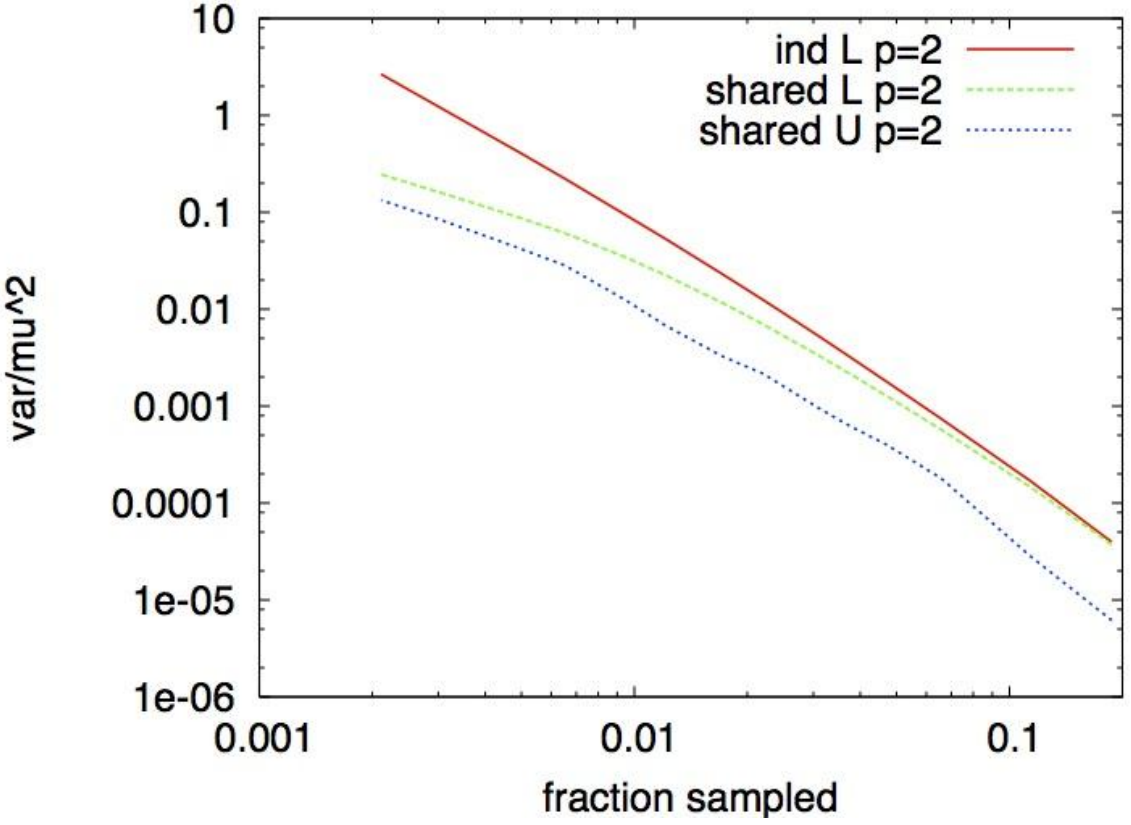
destination IP addresses: #IP flows in two time periods



Estimating L_2^2 difference

Independent / Coordinated, pps, Known seeds

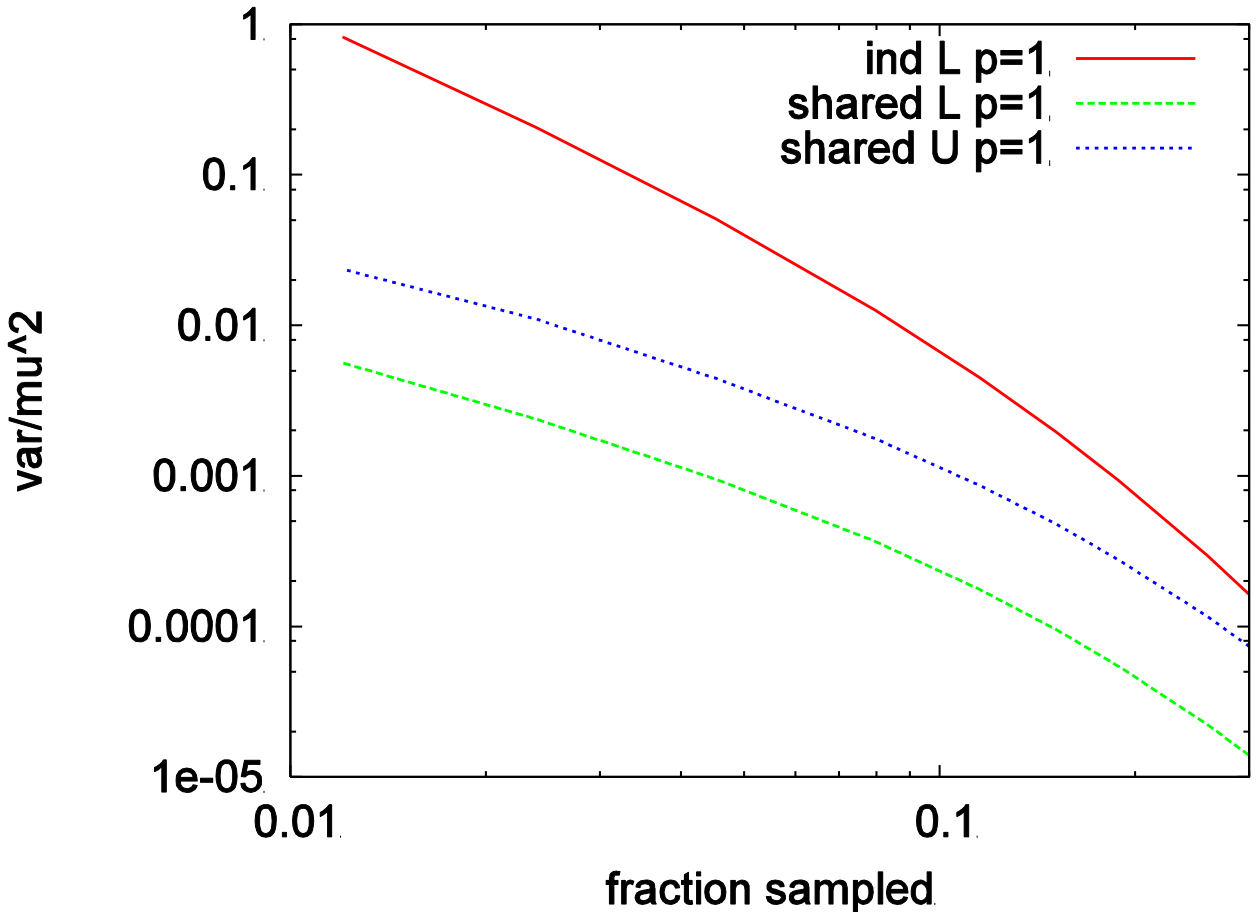
destination IP addresses: #IP flows in two time periods



Estimating L_1 difference

Independent / Coordinated, pps, Known seeds

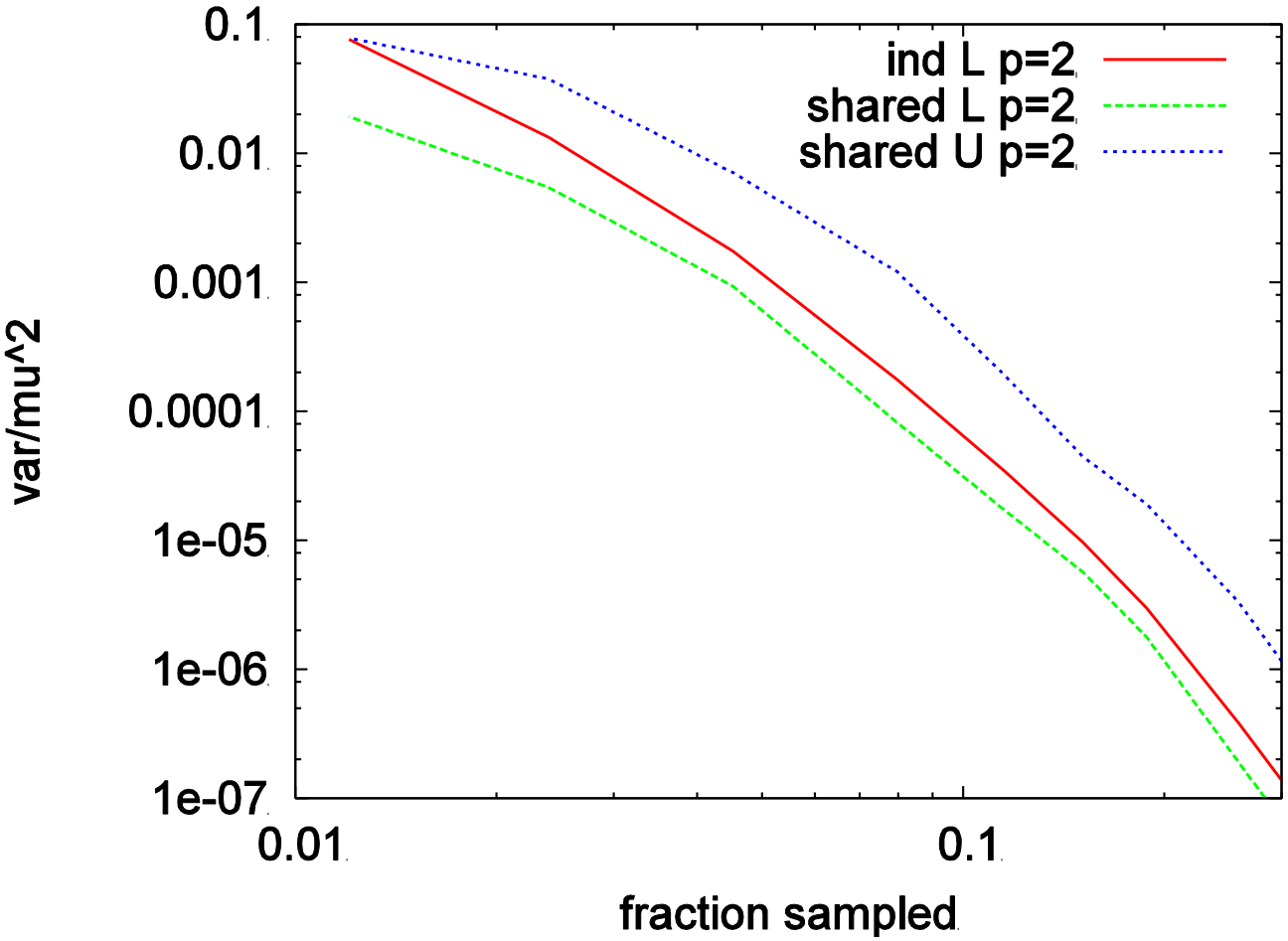
Surname occurrences in 2007, 2008 books (Google ngrams)



Estimating L_2^2 difference

Independent / Coordinated, pps, Known seeds

Surname occurrences in 2007, 2008 books (Google ngrams)



Conclusion

- We present estimators for sum aggregates over samples: unbiased, nonnegative, variance optimal
  - Tailoring estimator to data (<-optimality)
  - Classic sampling schemes: independent/coordinated weighted/weight-oblivious
- Perform well: tight estimates with small fraction sampled
  - considerable improvement over HT
  - can estimate functions for which there were no known unbiased nonnegative estimators, like L_2^2

Open/future: independent sampling (weighted+known seeds): precise characterization, derivation for >2 instances