The Matrix Mechanism: Optimizing Linear Counting Queries Under Differential Privacy

Gerome Miklau
Univ. of Massachusetts, Amherst

Joint work with:

Chao Li *Univ. of Massachusetts, Amherst*
Andrew McGregor *Univ. of Massachusetts, Amherst*
Michael Hay *Cornell University*
Vibhor Rastogi *Yahoo! Research*
Dan Suciu *University of Washington*

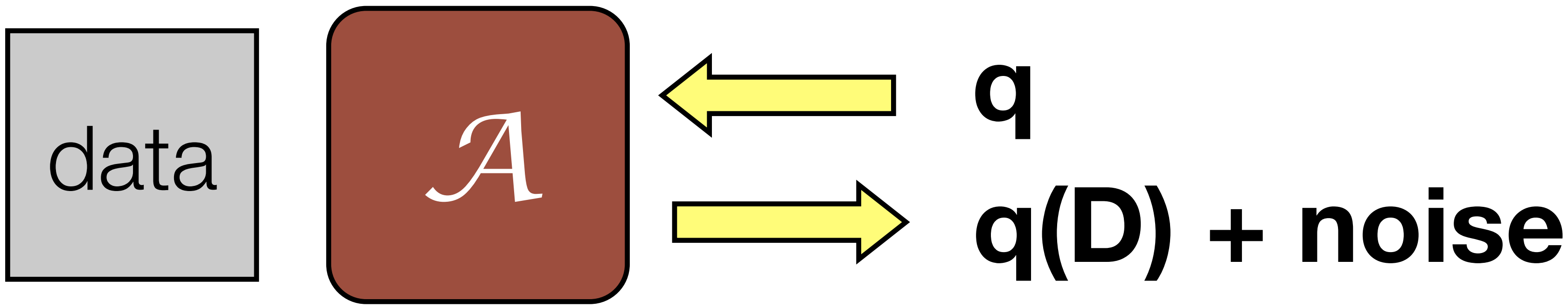
June 30, 2011

Data analysis under differential privacy

- The **differential guarantee** for participants in a data set:
  - Information released about a private data set is virtually indistinguishable whether or not a participant's data is included.
- Resistant to informed adversaries.
- Precise (public) error bounds on private output.

A central open question: what are **utility-optimal** mechanisms satisfying differential privacy?

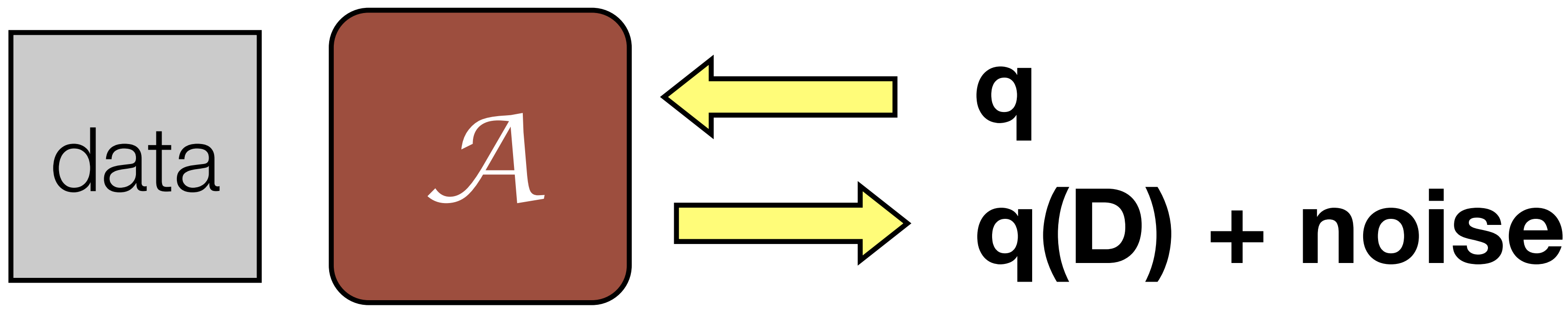
Differentially private mechanisms



An optimal mechanism is known for answering a single query

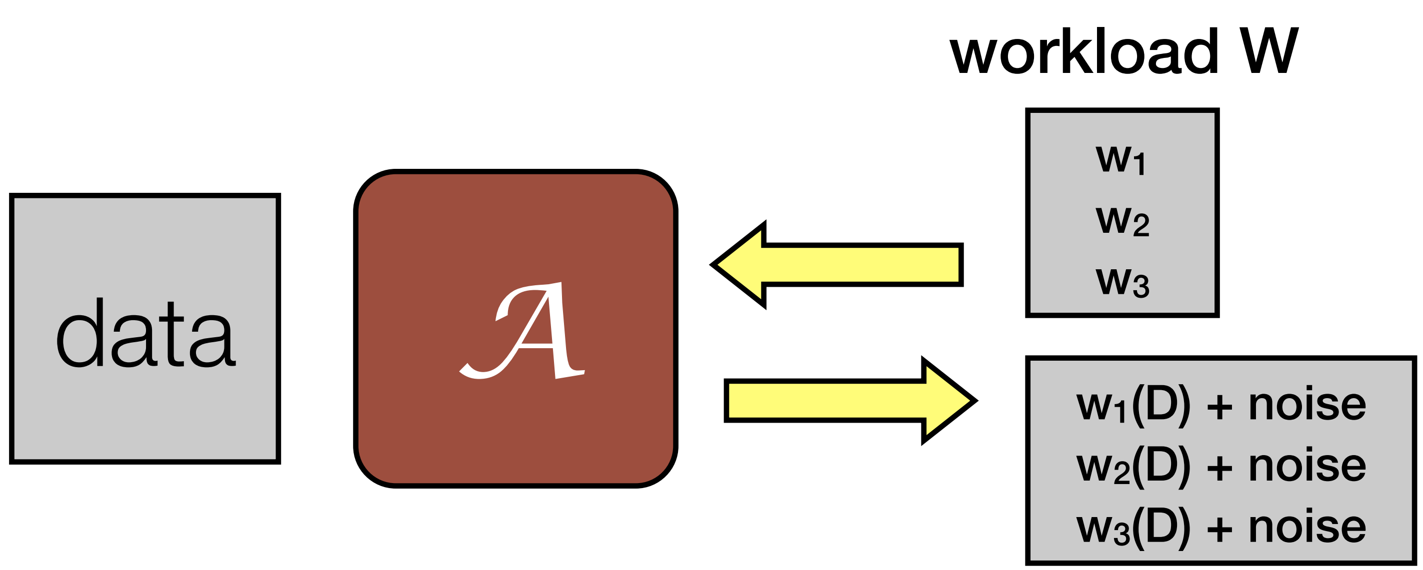
[Ghosh, 2009]

Differentially private mechanisms



An optimal mechanism is known for answering a single query

[Ghosh, 2009]



This mechanism is often sub-optimal for multiple queries

Query workloads

Query workloads

- A query workload is a set of **linear counting queries**, known ahead of time.
    - may include predicate counting queries, range queries, data cubes, sets of marginals, CDFs, etc...

Query workloads

- A query workload is a set of **linear counting queries**, known ahead of time.
  - may include predicate counting queries, range queries, data cubes, sets of marginals, CDFs, etc...
- How do query workloads arise ?
  - ... from decomposing a more complex data analysis task into a set of queries.
  - ... from multiple users accessing sensitive data, each issuing one or more queries.
  - ... from uncertainty about the eventual query answers needed--design workload to include all queries possibly of interest.

Query workloads

- A query workload is a set of **linear counting queries**, known ahead of time.
    - may include predicate counting queries, range queries, data cubes, sets of marginals, CDFs, etc...
- How do query workloads arise ?
    - ... from decomposing a more complex data analysis task into a set of queries.
    - ... from multiple users accessing sensitive data, each issuing one or more queries.
    - ... from uncertainty about the eventual query answers needed--design workload to include all queries possibly of interest.
- Our output can be treated as a **synthetic data set;** one which is designed to provide particularly accurate answers for the given workload queries.

Privacy definitions & mechanisms

Privacy definitions & mechanisms

- Differential privacy

Privacy definitions & mechanisms

- Differential privacy

A randomized algorithm A provides (ε,δ)-**differential privacy** if:

for all neighboring databases D and D', and

for any set of outputs *S*:

$$Pr[\mathcal{A}(D) \in S] \leq e^{\epsilon} Pr[\mathcal{A}(D') \in S] + \delta$$

Privacy definitions & mechanisms

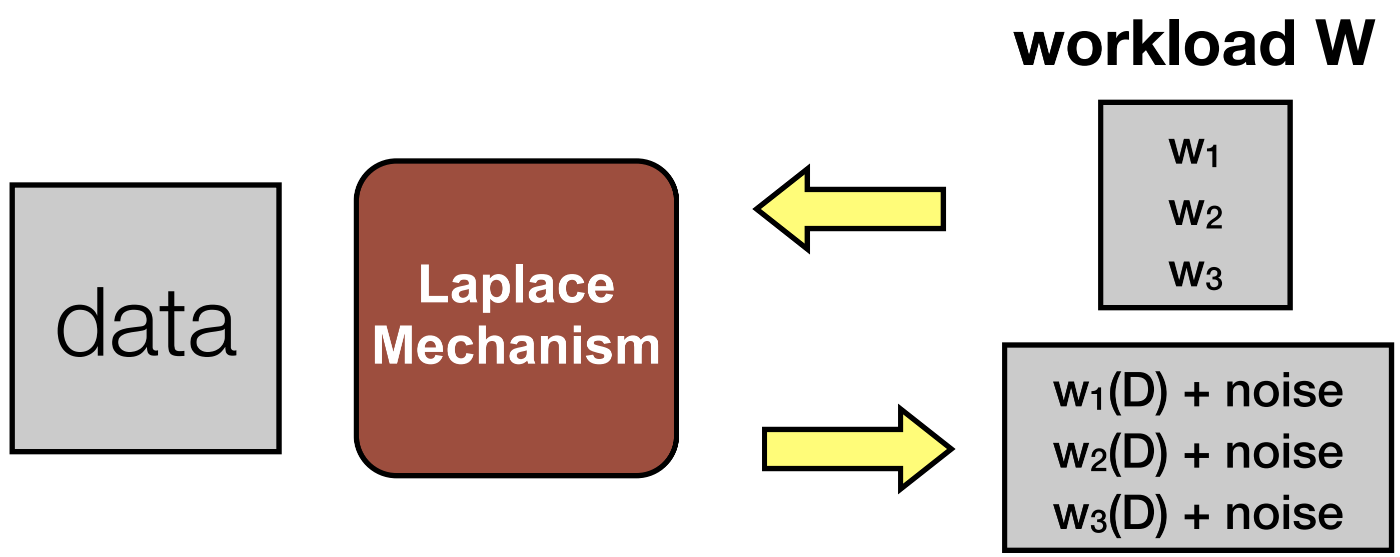
- Differential privacy

A randomized algorithm A provides (ε,δ)-**differential privacy** if:
for all neighboring databases D and D', and
for any set of outputs *S*:

$$Pr[\mathcal{A}(D) \in S] \leq e^{\epsilon} Pr[\mathcal{A}(D') \in S] + \delta$$

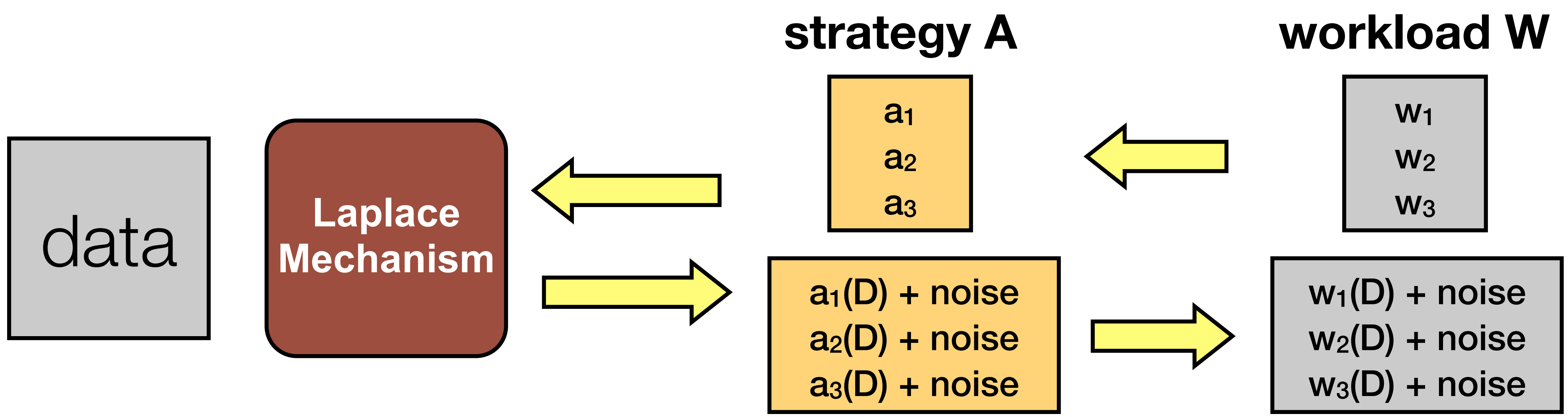
- if δ=0, standard ε-differential privacy
  - **Laplace(0,b) noise where $b=||q||_1/\varepsilon$**
- if δ>0, approximate (ε,δ)-differential privacy:
  - **Gaussian(0,σ) noise where $\sigma= ||q||_2\ f(\delta)/\varepsilon$**

Main approach

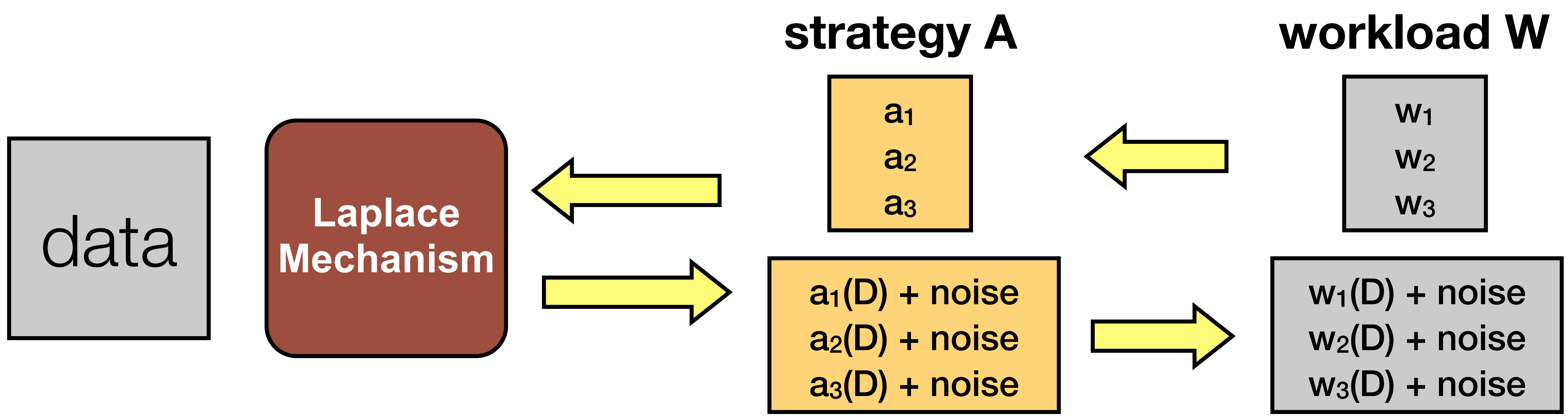


Main approach

Main approach

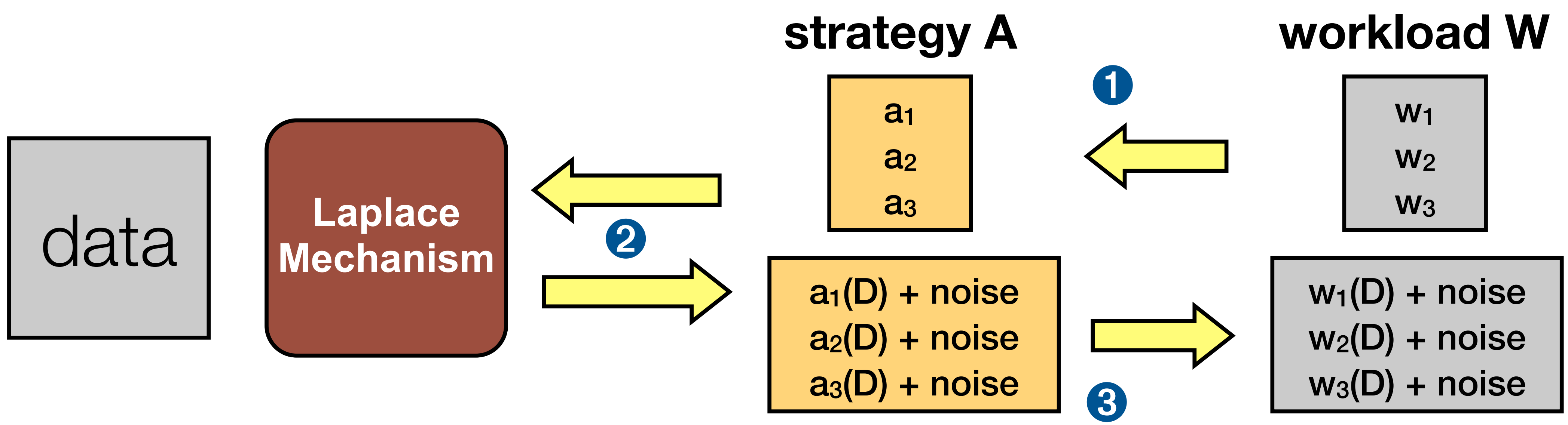


Main approach



1. (**Design**) Choose a set of queries **A** (the strategy)
2. (**Apply Laplace**) Use the Laplace mechanism to answer **A**
3. (**Derivation**) Compute each query in **W** using answers to **A**

Main approach

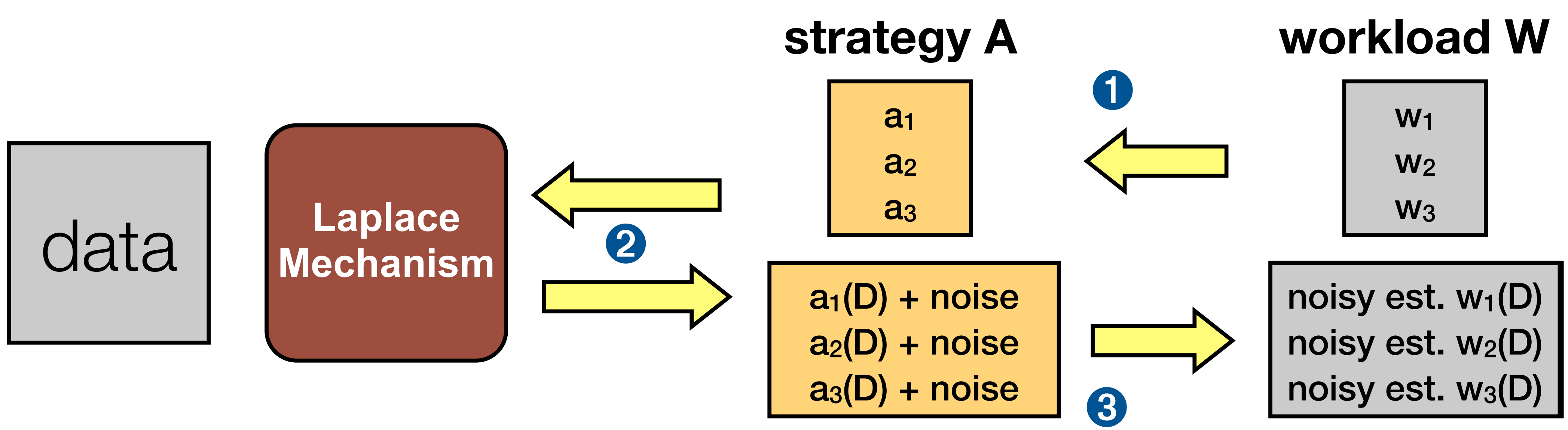


❶ (**Design**) Choose a set of queries **A** (the strategy)

❷ (**Apply Laplace**) Use the Laplace mechanism to answer **A**

❸ (**Derivation**) Compute each query in **W** using answers to **A**

Main approach

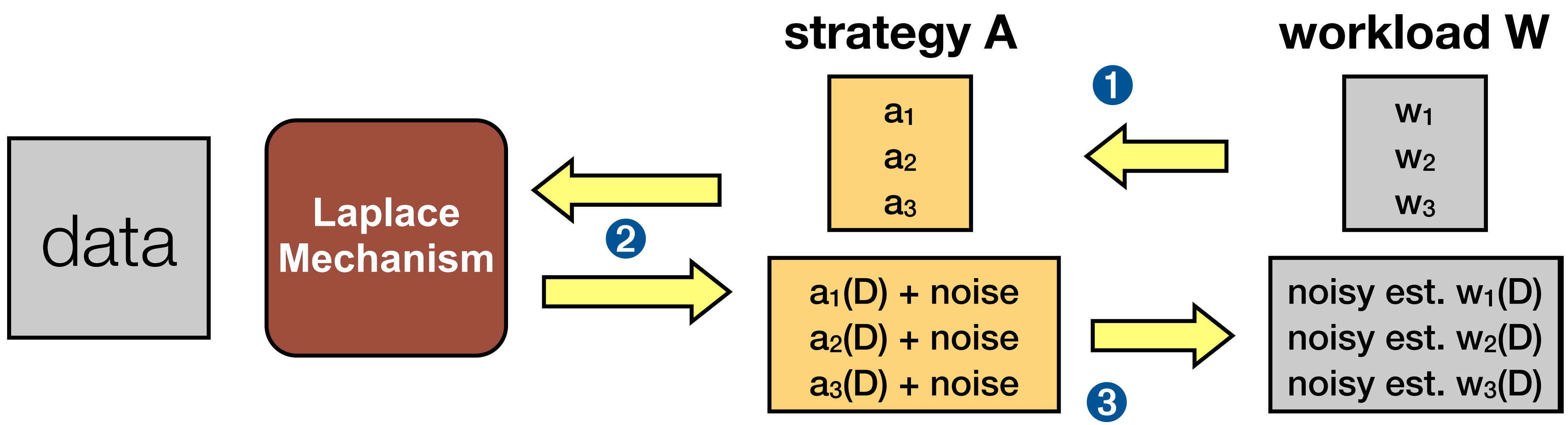


❶ (**Design**) Choose a set of queries **A** (the strategy)

❷ (**Apply Laplace**) Use the Laplace mechanism to answer **A**

❸ (**Derivation**) Compute each query in **W** using answers to **A**

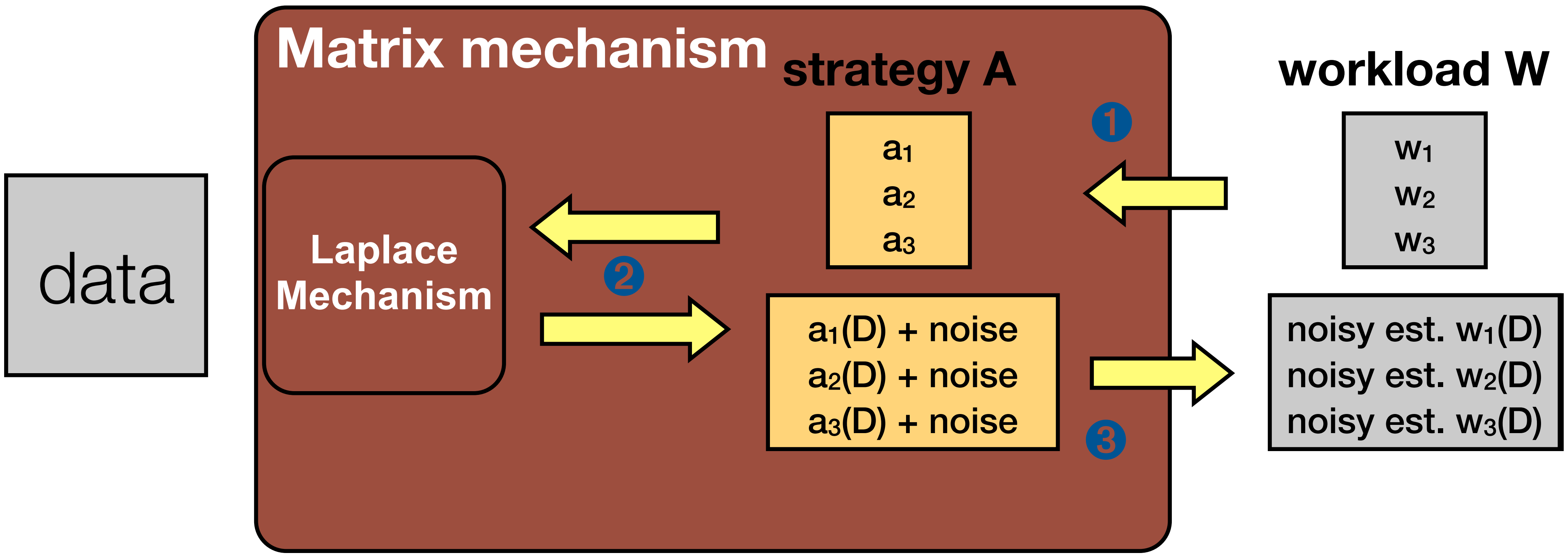
Main approach



1. (**Design**) Choose a set of queries **A** (the strategy)
2. (**Apply Laplace**) Use the Laplace mechanism to answer **A**
3. (**Derivation**) Compute each query in **W** using answers to **A**

Opportunity: choose **A** to minimize the error for queries in **W**.

Main approach



1. (**Design**) Choose a set of queries **A** (the strategy)
2. (**Apply Laplace**) Use the Laplace mechanism to answer **A**
3. (**Derivation**) Compute each query in **W** using answers to **A**

Opportunity: choose **A** to minimize the error for queries in **W**.

Outline

1. Case study: answering 1-dim range queries
2. The matrix mechanism -- formal description.
3. The matrix mechanism -- new tools & techniques.
4. Conclusion

Frequency representation of the database

Adding/re
tuple chan
component
frequency
exactly 1

Frequency v
length n, wh
"domain size
important p
the discussio

name	gender	grade
Alice	Female	91
Bob	Male	84
Carl	Male	82
Dave	Male	97
Edwina	Female	88
Faith	Female	78
Ghita	Female	85
...	...	...

Relational database

{gender, grade}

gender	grade	count
Male	100	10
Male	99	13
Male	98	5
Male	97	7
...	...	...
Female	100	15
Female	99	21
Female	98	4
Female	97	14
Female	96	9

Frequency vector

x_1
x_2
x_3
x_4
x_5
x_6
x_7
x_8
...
x_n

$\mathbf{x}$

Frequency representation of the database

name	gender	grade
Alice	Female	91
Bob	Male	84
Carl	Male	82
Dave	Male	97
Edwina	Female	88
Faith	Female	78
Ghita	Female	85
...	...	...

Relational database

Frequency vector $\mathbf{x}$

Adding/re
tuple chan
componen
frequency
exactly 1

Frequency v
length n, wh
"domain size
important p
the discussi

Frequency representation of the database

Adding/re
tuple chan
componen
frequency
exactly 1

Frequency v
length n, wh
"domain size
important p
the discussi

name	gender	grade
Alice	Female	91
Bob	Male	84
Carl	Male	82
Dave	Male	97
Edwina	Female	88
Faith	Female	78
Ghita	Female	85
...	...	...

Relational database

{grade}

grade	count
90-100	10
80-90	23
70-80	16
60-70	3

Frequency vector

x_1
x_2
x_3
x_4

$\mathbf{x}$

Answering all range queries

Goal: answer all range-count queries over x

$$\text{AllRange} = \{\ \mathbf{w} \mid \mathbf{w} = \mathbf{x_i} + \ldots + \mathbf{x_j} \quad \text{for } 1 \le i \le j \le n\ \}$$

workload W

w_1	range(x_1,x_4)	$\mathbf{x_1}$	+	$\mathbf{x_2}$	+	$\mathbf{x_3}$	+	$\mathbf{x_4}$
w_2	range(x_1,x_3)	$\mathbf{x_1}$	+	$\mathbf{x_2}$	+	$\mathbf{x_3}$		
w_3	range(x_2,x_4)			$\mathbf{x_2}$	+	$\mathbf{x_3}$	+	$\mathbf{x_4}$
w_4	range(x_1,x_2)	$\mathbf{x_1}$	+	$\mathbf{x_2}$				
w_5	range(x_2,x_3)			$\mathbf{x_2}$	+	$\mathbf{x_3}$		
w_6	range(x_3,x_4)					$\mathbf{x_3}$	+	$\mathbf{x_4}$
w_7	range(x_1,x_1)	$\mathbf{x_1}$						
w_8	range(x_2,x_2)			$\mathbf{x_2}$				
w_9	range(x_3,x_3)					$\mathbf{x_3}$		
w_{10}	range(x_4,x_4)							$\mathbf{x_4}$

Answering all range queries

Goal: answer all range-count queries over x

AllRange = { $\mathbf{w} \mid \mathbf{w} = \mathbf{x_i} + \ldots + \mathbf{x_j}$ for $1 \leq i \leq j \leq n$ }

workload W

w_1	range(x_1,x_4)	x_1	+	x_2	+	x_3	+	x_4
w_2	range(x_1,x_3)	x_1	+	x_2	+	x_3		
w_3	range(x_2,x_4)			x_2	+	x_3	+	x_4
w_4	range(x_1,x_2)	x_1	+	x_2				
w_5	range(x_2,x_3)			x_2	+	x_3		
w_6	range(x_3,x_4)					x_3	+	x_4
w_7	range(x_1,x_1)	x_1						
w_8	range(x_2,x_2)			x_2				
w_9	range(x_3,x_3)					x_3		
w_{10}	range(x_4,x_4)							x_4

x=	10		23		16		3

Answering all range queries

Goal: answer all range-count queries over x

$$\text{AllRange} = \{\ \mathbf{w} \mid \mathbf{w} = \mathbf{x_i} + \ldots + \mathbf{x_j} \quad \text{for } 1 \leq i \leq j \leq n\ \}$$

workload W

w_1	range(x_1,x_4)	x_1	+	x_2	+	x_3	+	x_4	w_1	52
w_2	range(x_1,x_3)	x_1	+	x_2	+	x_3			w_2	49
w_3	range(x_2,x_4)			x_2	+	x_3	+	x_4	w_3	42
w_4	range(x_1,x_2)	x_1	+	x_2					w_4	33
w_5	range(x_2,x_3)			x_2	+	x_3			w_5	39
w_6	range(x_3,x_4)					x_3	+	x_4	w_6	19
w_7	range(x_1,x_1)	x_1							w_7	10
w_8	range(x_2,x_2)			x_2					w_8	23
w_9	range(x_3,x_3)					x_3			w_9	16
w_{10}	range(x_4,x_4)							x_4	w_{10}	3

x=	10		23		16		3

Approach 1: basic Laplace mechanism

Two problems:

- high error
- inconsistency

Explain sensi

Workload queries

W

w_1	x_1	+	x_2	+	x_3	+	x_4
w_2	x_1	+	x_2	+	x_3		
w_3			x_2	+	x_3	+	x_4
w_4	x_1	+	x_2				
w_5			x_2	+	x_3		
w_6					x_3	+	x_4
w_7	x_1						
w_8			x_2				
w_9					x_3		
w_{10}							x_4

$$\|\mathbf{W}\|_1 = 6$$

Laplace noise

$$+ \ (6/\varepsilon) \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \\ b_8 \\ b_9 \\ b_{10} \end{bmatrix} \Rightarrow \begin{bmatrix} w'_1 \\ w'_2 \\ w'_3 \\ w'_4 \\ w'_5 \\ w'_6 \\ w'_7 \\ w'_8 \\ w'_9 \\ w'_{10} \end{bmatrix}$$

private output

Error is measured as variance

Approach 1: basic Laplace mechanism

Two problems:

- high error
- inconsistency

Explain sensi

Workload queries

w_1	x_1	+	x_2	+	x_3	+	x_4	**52**
w_2	x_1	+	x_2	+	x_3			**49**
w_3			x_2	+	x_3	+	x_4	**42**
w_4	x_1	+	x_2					**33**
w_5			x_2	+	x_3			**39**
w_6					x_3	+	x_4	**19**
w_7	x_1							**10**
w_8			x_2					**23**
w_9					x_3			**16**
w_{10}							x_4	**3**

W

$$\|W\|_1 = 6$$

$$+\ (6/\varepsilon)$$

Laplace noise

b_1	8.2
b_2	-5.4
b_3	-3.1
b_4	6.6
b_5	-7.9
b_6	2.4
b_7	-3.0
b_8	-4.9
b_9	6.7
b_{10}	4.6

private output

w'_1	60.2
w'_2	44.6
w'_3	38.9
w'_4	39.6
w'_5	31.1
w'_6	21.4
w'_7	7.0
w'_8	18.1
w'_9	22.7
w'_{10}	7.6

Error is measured as variance

Approach 1: basic Laplace mechanism

Two problems:

- high error
- inconsistency

Workload queries

$\mathbf{W}$

w_1	x_1	+	x_2	+	x_3	+	x_4	52
w_2	x_1	+	x_2	+	x_3			49
w_3			x_2	+	x_3	+	x_4	42
w_4	x_1	+	x_2					33
w_5			x_2	+	x_3			39
w_6					x_3	+	x_4	19
w_7	x_1							10
w_8			x_2					23
w_9					x_3			16
w_{10}							x_4	3

$$\|\mathbf{W}\|_1 = 6$$

$$+ \ (6/\varepsilon)$$

Laplace noise

b_1	8.2
b_2	-5.4
b_3	-3.1
b_4	6.6
b_5	-7.9
b_6	2.4
b_7	-3.0
b_8	-4.9
b_9	6.7
b_{10}	4.6

private output

w'_1	60.2
w'_2	44.6
w'_3	38.9
w'_4	39.6
w'_5	31.1
w'_6	21.4
w'_7	7.0
w'_8	18.1
w'_9	22.7
w'_{10}	7.6

$\Sigma = 55.4$

Explain sensi

Error is measured as variance

Approach 1: basic Laplace mechanism

Two problems:

- high error
- inconsistency

Explain sensi

Workload queries | **Laplace noise** | **private output**

w_1	$x_1 + x_2 + x_3 + x_4$	52	b_1	8.2	w'_1	60.2
w_2	$x_1 + x_2 + x_3$	49	b_2	-5.4	w'_2	44.6
w_3	$x_2 + x_3 + x_4$	42	b_3	-3.1	w'_3	38.9
w_4	$x_1 + x_2$	33	b_4	6.6	w'_4	39.6
w_5	$x_2 + x_3$	39	b_5	-7.9	w'_5	31.1
w_6	$x_3 + x_4$	19	b_6	2.4	w'_6	21.4
w_7	x_1	10	b_7	-3.0	w'_7	7.0
w_8	x_2	23	b_8	-4.9	w'_8	18.1
w_9	x_3	16	b_9	6.7	w'_9	22.7
w_{10}	x_4	3	b_{10}	4.6	w'_{10}	7.6

$\mathbf{W}$ $+ (6/\varepsilon)$

$\Sigma=55.4$

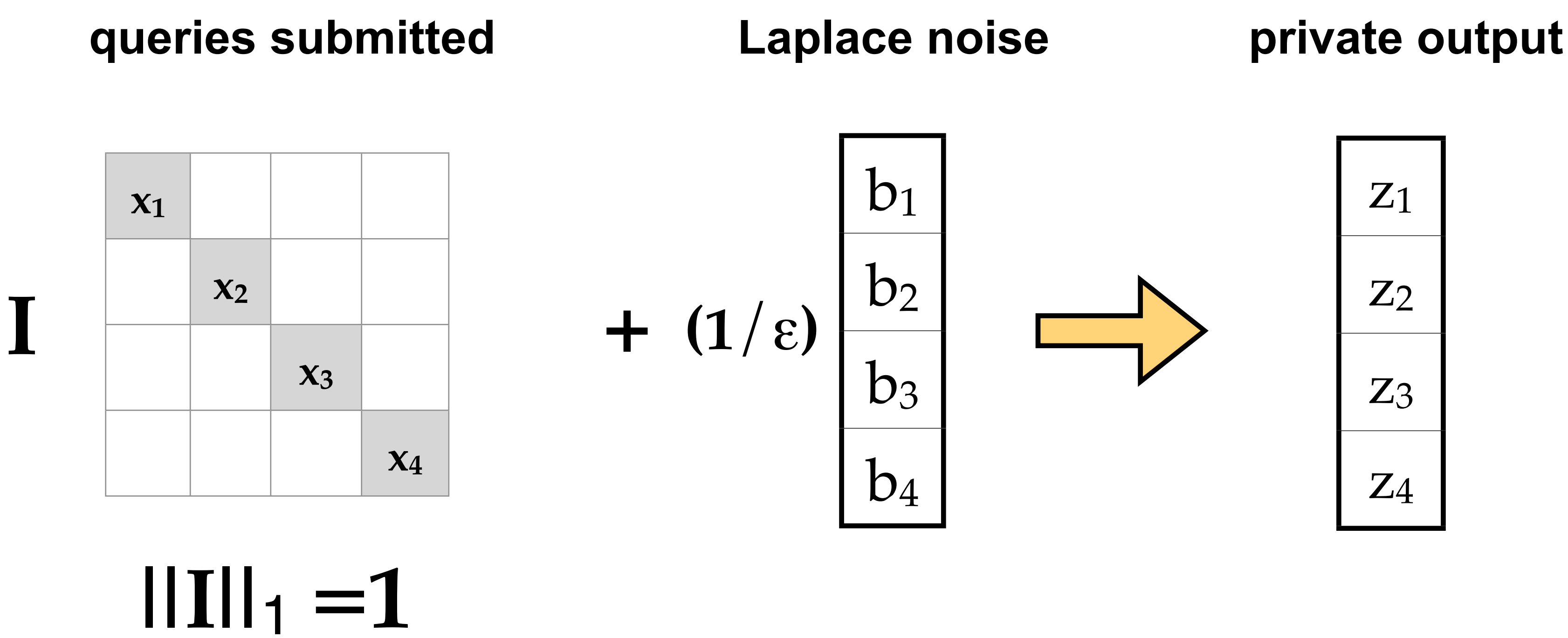
$||W||_1 = 6$

	$n=4$	n
Sensitivity $\|\|W\|\|_1$	6	$O(n^2)$
Error per query	$2(\|\|W\|\|_1/\varepsilon)^2 = 72/\varepsilon^2$	$2(\|\|W\|\|_1/\varepsilon)^2 = O(n^4)/\varepsilon^2$

Error is measured as variance

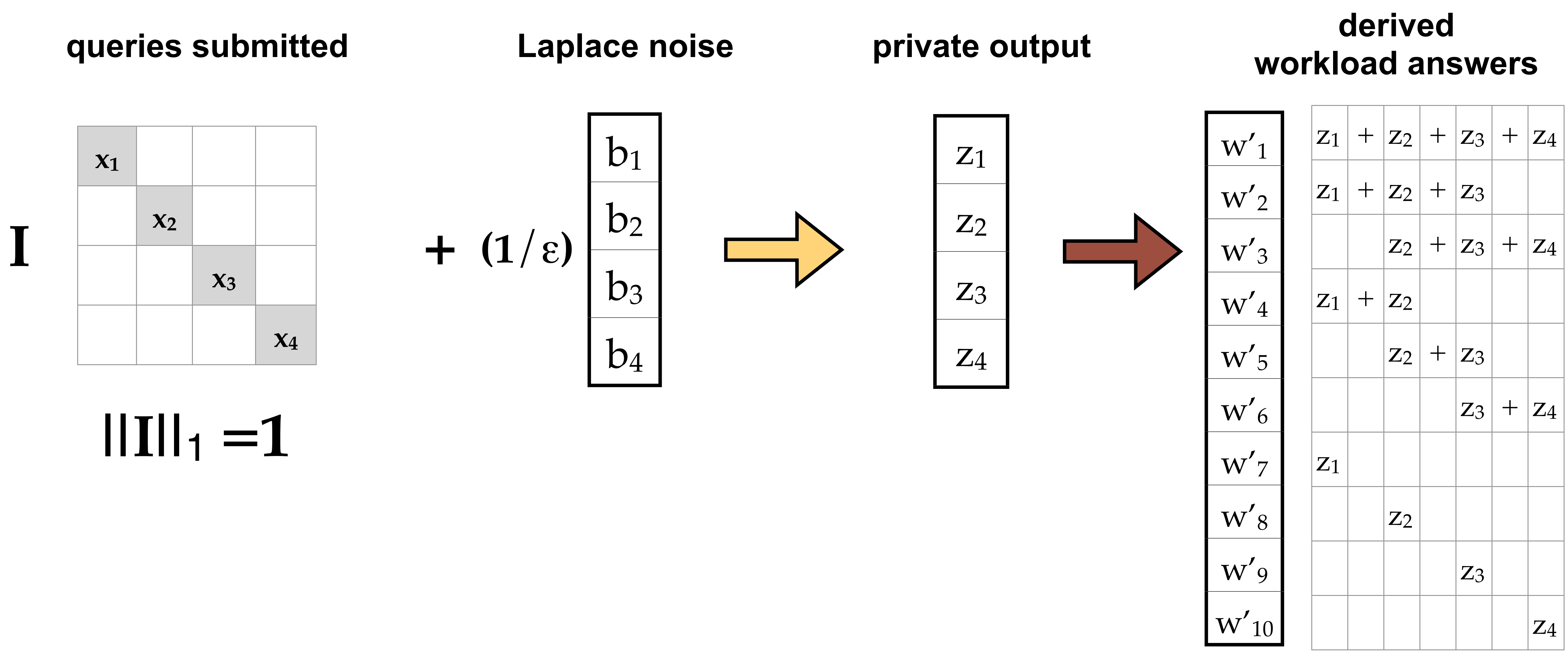
Approach 2: noisy frequency counts

Use Laplace mechanism to get noisy estimates for each x_i.



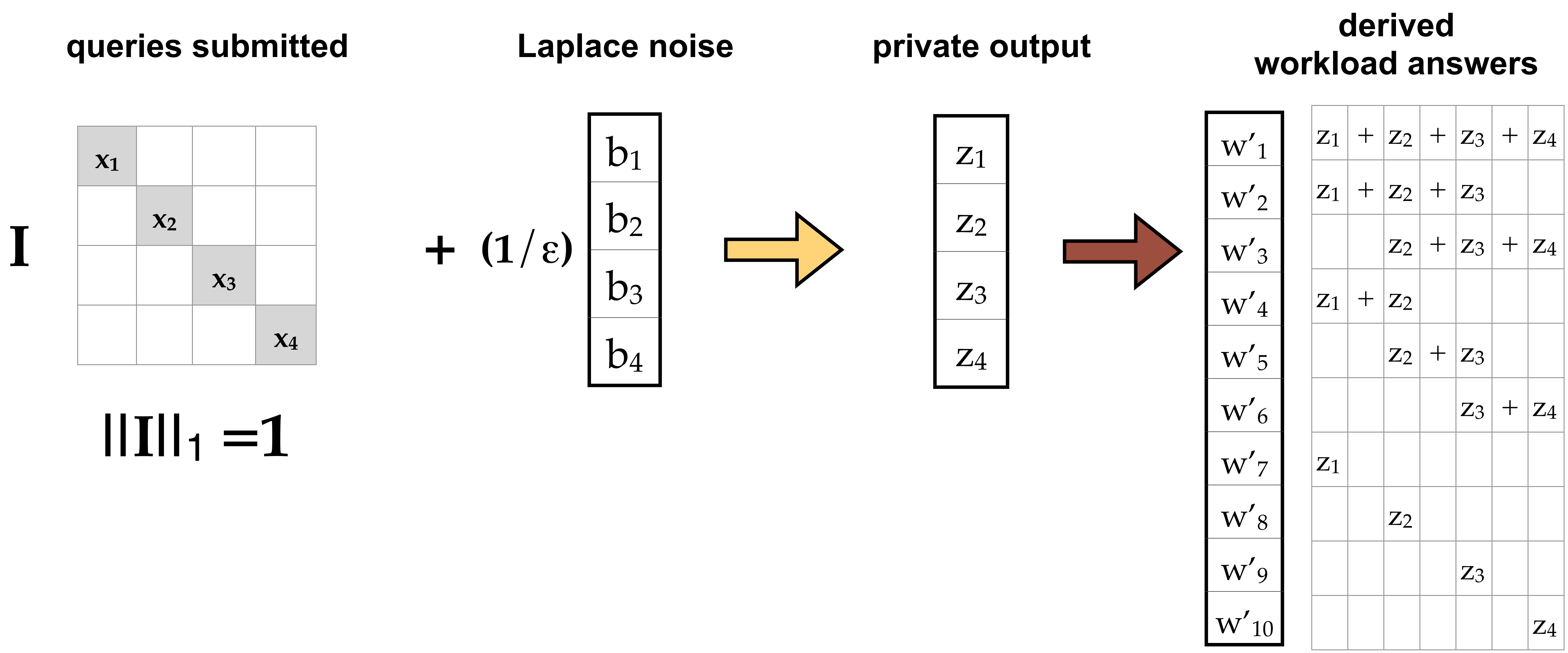
Approach 2: noisy frequency counts

Use Laplace mechanism to get noisy estimates for each x_i.



Approach 2: noisy frequency counts

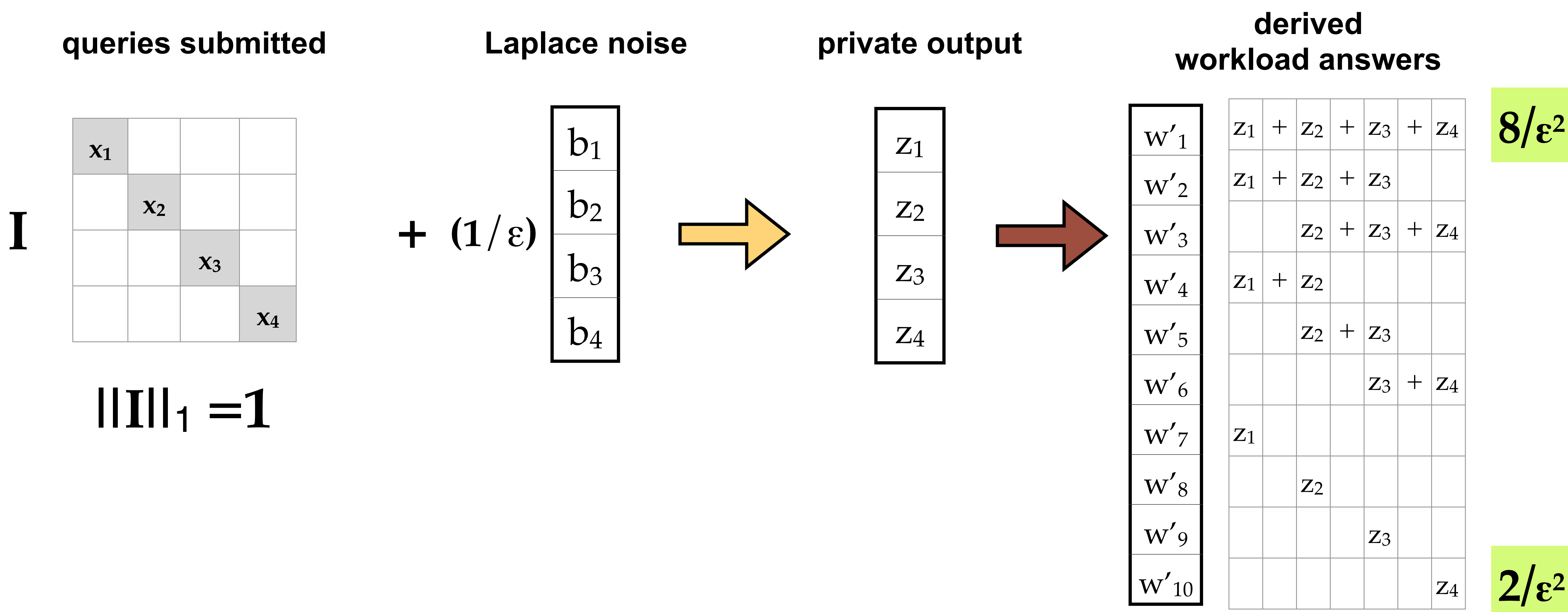
Use Laplace mechanism to get noisy estimates for each x_i.



For **w=range(x_i,x_j)** Error(w)= $2(j-i+1) / \varepsilon^2$

Approach 2: noisy frequency counts

Use Laplace mechanism to get noisy estimates for each $\mathbf{x_i}$.



For $\mathbf{w=range(x_i,x_j)}$ $\mathrm{Error}(w) = 2(j-i+1)/\varepsilon^2$

Approach 3: hierarchical queries

Hierarchical queries: recursively partition the domain, computing sums of each interval.

Queries submitted **Laplace noise** **private output**

H

x_1	+	x_2	+	x_3	+	x_4
x_1	+	x_2				
				x_3	+	x_4
x_1						
		x_2				
				x_3		
						x_4

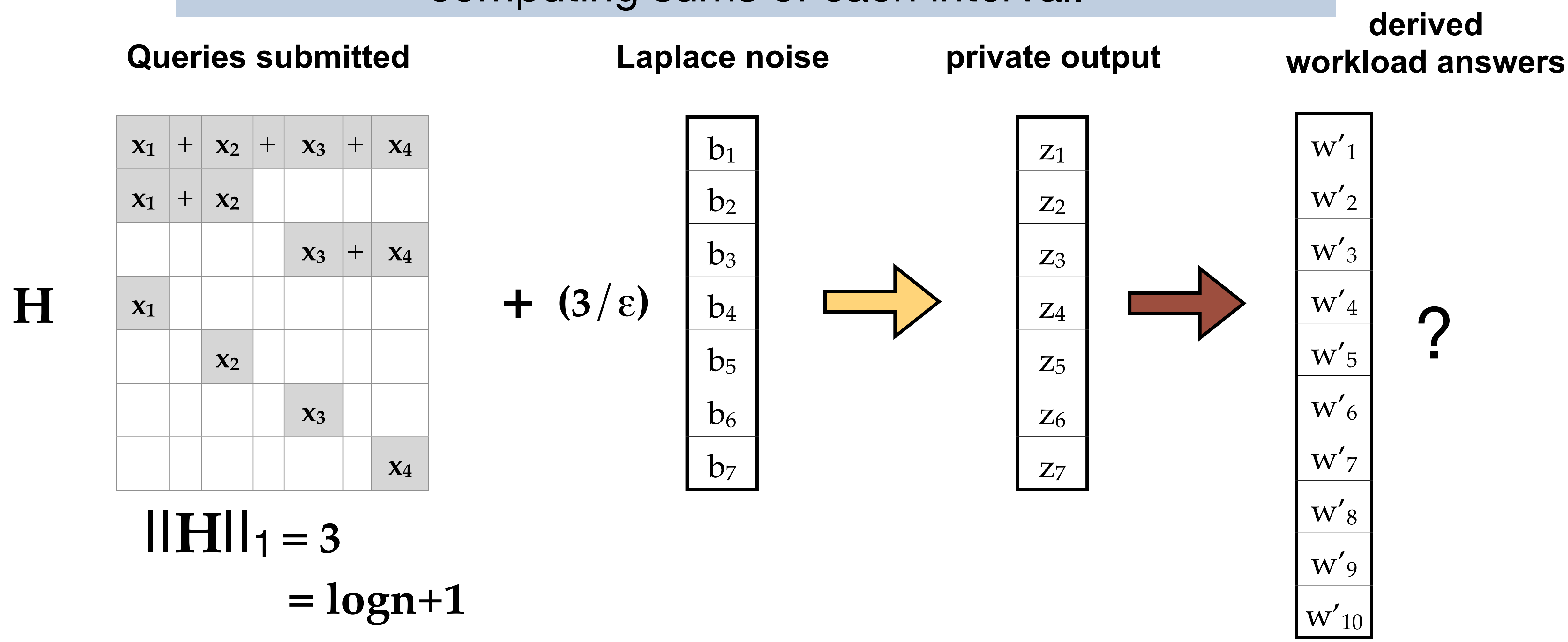
$$+\ (3/\varepsilon)\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} \Rightarrow \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \end{bmatrix}$$

$$||\mathbf{H}||_1 = 3$$

$$= \log n + 1$$

Approach 3: hierarchical queries

Hierarchical queries: recursively partition the domain, computing sums of each interval.



Approach 3: hierarchical queries

Hierarchical queries: recursively partition the domain, computing sums of each interval.

Queries submitted | **Laplace noise** | **private output** | **derived workload answers**

$$\mathbf{H} \quad \begin{bmatrix} x_1 + x_2 + x_3 + x_4 \\ x_1 + x_2 \\ x_3 + x_4 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + (3/\varepsilon) \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} \Rightarrow \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \end{bmatrix} \Rightarrow \begin{bmatrix} w'_1 \\ w'_2 \\ w'_3 \\ w'_4 \\ w'_5 \\ w'_6 \\ w'_7 \\ w'_8 \\ w'_9 \\ w'_{10} \end{bmatrix} \; ?$$

$$||\mathbf{H}||_1 = 3$$
$$= \log n + 1$$

Possible estimates for query $\text{range}(x_2, x_3) = x_2 + x_3$

Approach 3: hierarchical queries

Hierarchical queries: recursively partition the domain, computing sums of each interval.

Queries submitted | **Laplace noise** | **private output** | **derived workload answers**

H

x_1	+	x_2	+	x_3	+	x_4
x_1	+	x_2				
				x_3	+	x_4
x_1						
		x_2				
				x_3		
						x_4

$$+\ (3/\varepsilon)\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} \Rightarrow \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \end{bmatrix} \Rightarrow \begin{bmatrix} w'_1 \\ w'_2 \\ w'_3 \\ w'_4 \\ w'_5 \\ w'_6 \\ w'_7 \\ w'_8 \\ w'_9 \\ w'_{10} \end{bmatrix} \ ?$$

$$||\mathbf{H}||_1 = 3$$
$$= \log n + 1$$

Possible estimates for query $\mathrm{range}(x_2, x_3) = x_2 + x_3$

$z_5 + z_6$

Approach 3: hierarchical queries

Hierarchical queries: recursively partition the domain, computing sums of each interval.

Queries submitted | **Laplace noise** | **private output** | **derived workload answers**

$$\mathbf{H} = \begin{bmatrix} x_1 + x_2 + x_3 + x_4 \\ x_1 + x_2 \\ x_3 + x_4 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + (3/\varepsilon) \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} \Rightarrow \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \end{bmatrix} \Rightarrow \begin{bmatrix} w'_1 \\ w'_2 \\ w'_3 \\ w'_4 \\ w'_5 \\ w'_6 \\ w'_7 \\ w'_8 \\ w'_9 \\ w'_{10} \end{bmatrix} \; ?$$

$$||\mathbf{H}||_1 = 3$$
$$= \log n + 1$$

Possible estimates for query $\mathrm{range}(x_2, x_3) = x_2 + x_3$

$z_5 + z_6$

$z_2 - z_4 + z_6$

[Hay, PVLDB 10]

Approach 3: hierarchical queries

Hierarchical queries: recursively partition the domain, computing sums of each interval.

Queries submitted

H

x_1	+	x_2	+	x_3	+	x_4
x_1	+	x_2				
				x_3	+	x_4
x_1						
		x_2				
				x_3		
						x_4

$$||\mathbf{H}||_1 = 3$$
$$= \log n + 1$$

Laplace noise

$+ \ (3/\varepsilon)$ $[b_1, b_2, b_3, b_4, b_5, b_6, b_7]^T$

private output

$[z_1, z_2, z_3, z_4, z_5, z_6, z_7]^T$

derived workload answers

$[w'_1, w'_2, w'_3, w'_4, w'_5, w'_6, w'_7, w'_8, w'_9, w'_{10}]^T$?

Possible estimates for query $\text{range}(x_2, x_3) = x_2 + x_3$

$z_5 + z_6$

$z_2 - z_4 + z_6$

$z_1 - z_4 - z_7$

Approach 3: hierarchical queries

Hierarchical queries: recursively partition the domain, computing sums of each interval.

Queries submitted

H

x_1	+	x_2	+	x_3	+	x_4
x_1	+	x_2				
				x_3	+	x_4
x_1						
		x_2				
				x_3		
						x_4

$||\mathbf{H}||_1 = 3$

$= \log n + 1$

Laplace noise

$+ \ (3/\varepsilon)$ $[b_1, b_2, b_3, b_4, b_5, b_6, b_7]$

private output

$[z_1, z_2, z_3, z_4, z_5, z_6, z_7]$

derived workload answers

$[w'_1, w'_2, w'_3, w'_4, w'_5, w'_6, w'_7, w'_8, w'_9, w'_{10}]$

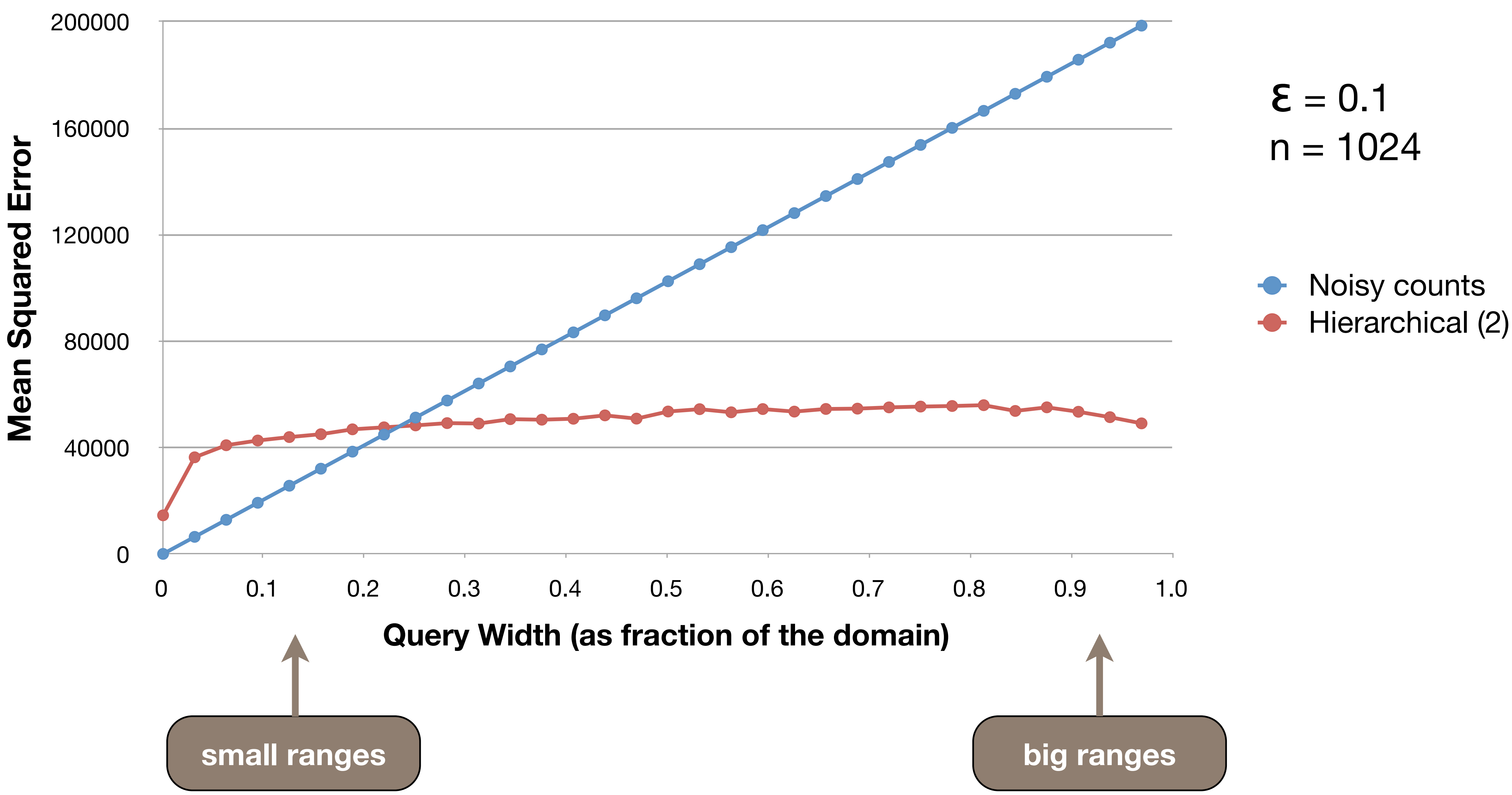
?

Possible estimates for query $\text{range}(x_2, x_3) = x_2 + x_3$

Least-squares estimate $(6z_1 + 3z_2 + 3z_3 - 9z_4 + 12z_5 + 12z_6 - 9z_7)/21$

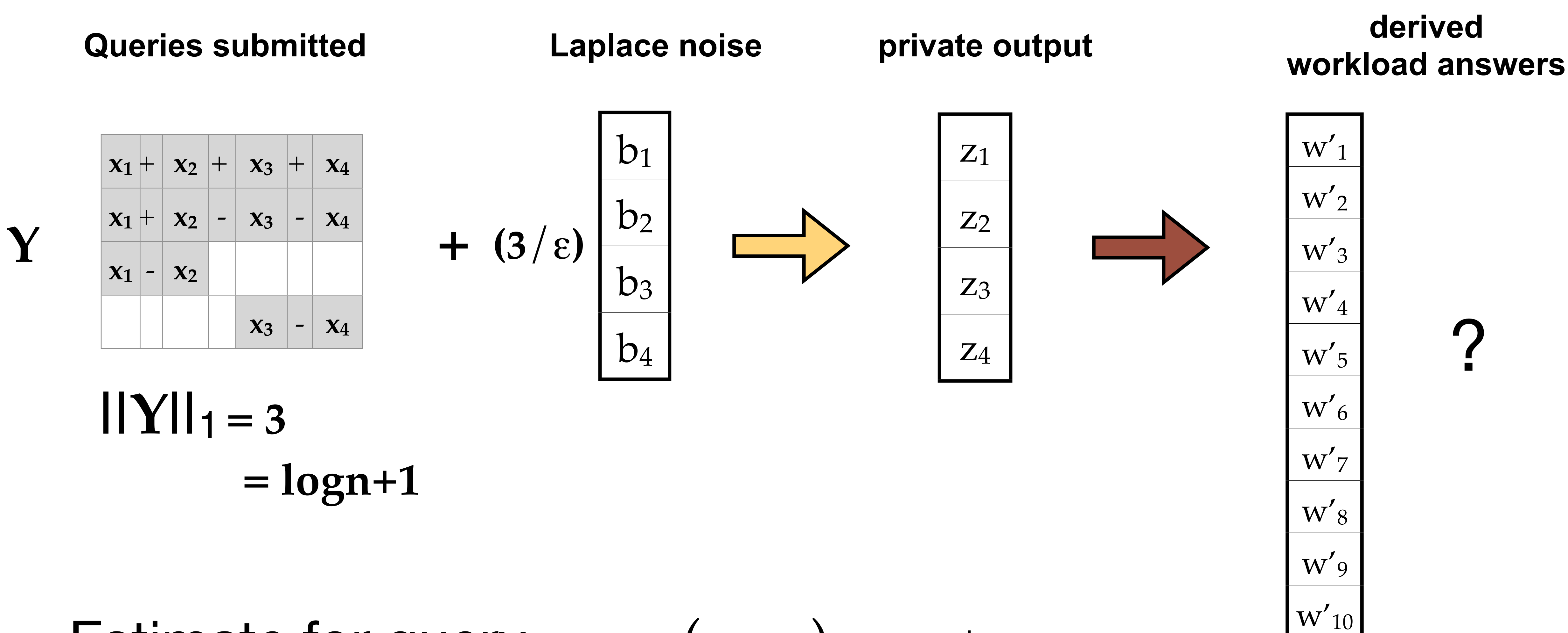
Error rates: workload of all range queries

ε-differential privacy



Approach 4: wavelet queries

Wavelet queries: use Haar wavelet to get noisy summary of data.



Estimate for query $\text{range}(x_2, x_3) = x_2 + x_3$

$$.5z_1 + 0z_2 - .5z_3 + .5z_4$$

Approach 4: wavelet queries

Wavelet queries: use Haar wavelet to get noisy summary of data.

Queries submitted | **Laplace noise** | **private output** | **derived workload answers**

$$Y \quad \begin{bmatrix} x_1 + x_2 + x_3 + x_4 \\ x_1 + x_2 - x_3 - x_4 \\ x_1 - x_2 \\ x_3 - x_4 \end{bmatrix} + (3/\varepsilon) \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} \Rightarrow \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} \Rightarrow \begin{bmatrix} w'_1 \\ w'_2 \\ w'_3 \\ w'_4 \\ w'_5 \\ w'_6 \\ w'_7 \\ w'_8 \\ w'_9 \\ w'_{10} \end{bmatrix} \quad ?$$

$$||Y||_1 = 3$$
$$= \log n + 1$$

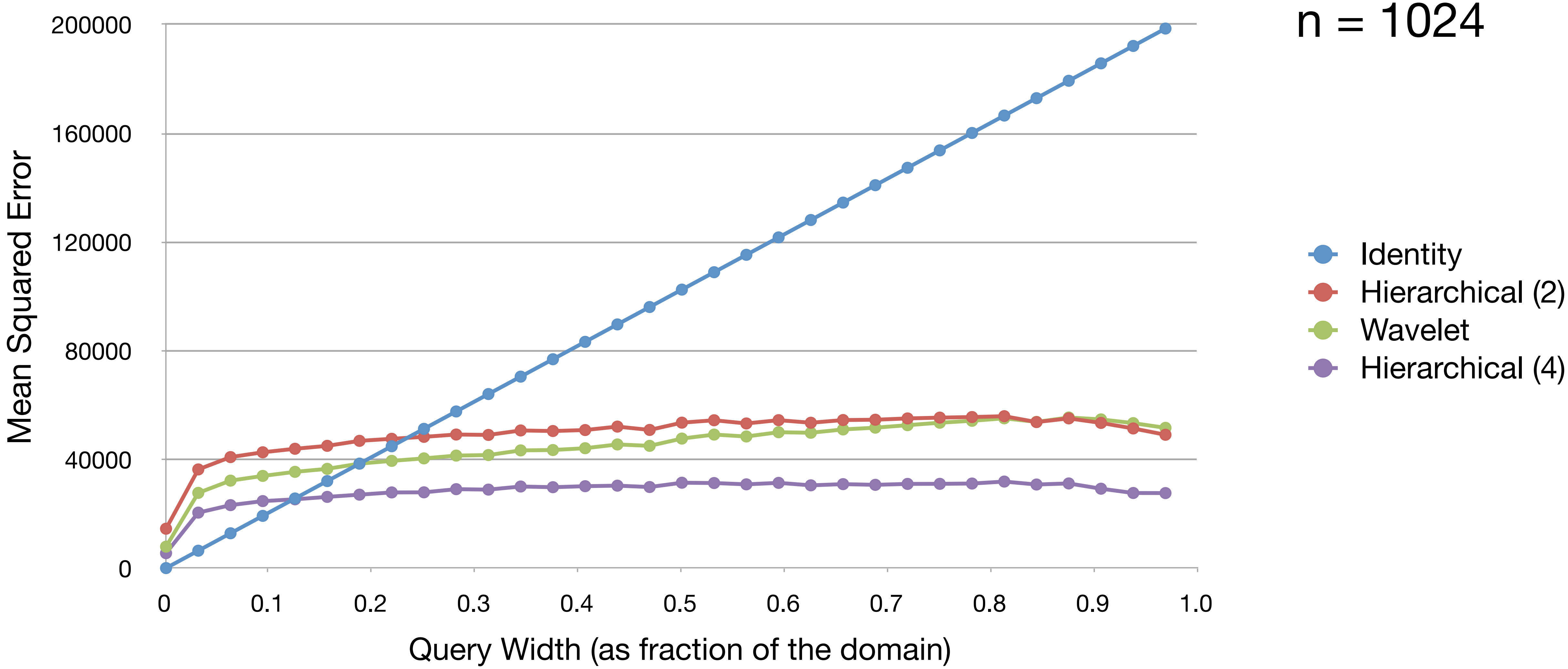
Estimate for query $\text{range}(x_2,x_3) = x_2 + x_3$

$$\boxed{.5z_1 + 0z_2 - .5z_3 + .5z_4}$$

Error: workload of all range queries

ε-differential privacy

ε = 0.1
n = 1024



Strategies for workload of all range queries

Noisy counts

x_1						
		x_2				
				x_3		
						x_4

I

Hierarchical

x_1	+	x_2	+	x_3	+	x_4
x_1	+	x_2				
				x_3	+	x_4
x_1						
		x_2				
				x_3		
						x_4

H

Wavelet

x_1	+	x_2	+	x_3	+	x_4
x_1	+	x_2	-	x_3	-	x_4
x_1	-	x_2				
				x_3	-	x_4

Y

Very low sensitivity, but large ranges estimated badly.

Low sensitivity, and all range queries can be estimated using no more than logn output entries.

Max/Avg error	$O(n/\varepsilon^2)$	$O(\log^3 n/\varepsilon^2)$	$O(\log^3 n/\varepsilon^2)$

Questions raised

Questions raised

- Are these approaches optimal for all range queries?

Questions raised

- Are these approaches optimal for all range queries?
- What about other workloads?

Questions raised

- Are these approaches optimal for all range queries?
- What about other workloads?
- Big picture:
  - x values we cannot observe directly.
  - we can request noisy estimates of any linear function of the x values, at some cost.
  - what should we request to perform our task (i.e. answer workload queries) ?

Questions raised

- Are these approaches optimal for all range queries?
- What about other workloads?
- Big picture:
  - x values we cannot observe directly.
  - we can request noisy estimates of any linear function of the x values, at some cost.
  - what should we request to perform our task (i.e. answer workload queries) ?

Optimal Experimental Design

Outline

1. Case study: answering 1-dim range queries
2. The matrix mechanism -- formal description.
3. The matrix mechanism -- new tools & techniques.
4. Conclusion

Outline

1. Case study: answering 1-dim range queries
2. The matrix mechanism -- formal description.
3. The matrix mechanism -- new tools & techniques.
4. Conclusion

Linear counting queries

A **linear counting query** w computes a linear combination of the frequency vector counts:

$$\mathbf{w(D)} = w_1x_1 + w_2x_2 + \ldots + w_nx_n \qquad \text{each } w_i \in R$$

1) Expressiveness o
queries

2) Need to list ALL
-- don't omit those
be derived

3) Can scale rows t
error rates of each

Linear counting queries

A **linear counting query** $\mathbf{w}$ computes a linear combination of the frequency vector counts:

$$\mathbf{w(D)} = w_1x_1 + w_2x_2 + \ldots + w_nx_n$$

each $w_i \in R$

... as a length n row vector:

$$\mathbf{w} = [w_1, w_2, w_3 \ldots w_n]$$

The query result is:

$$\mathbf{wx}$$

1) Expressiveness o
queries

2) Need to list ALL
-- don't omit those
be derived

3) Can scale rows t
error rates of each

Linear counting queries

1) Expressiveness o
queries

2) Need to list ALL
-- don't omit those
be derived

3) Can scale rows t
error rates of each

A **linear counting query** $\mathbf{w}$ computes a linear combination of the frequency vector counts:

$$\mathbf{w(D)} = w_1x_1 + w_2x_2 + \ldots + w_nx_n$$

each $w_i \in R$

... as a length n row vector:

$$\mathbf{w} = [w_1, w_2, w_3 \ldots w_n]$$

The query result is:

$$\mathbf{wx}$$

a set of linear counting queries is a matrix:

$$\mathbf{W} \quad \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix}$$

The query result is:

$$\mathbf{Wx}$$

The sensitivity of a query matrix

- For two neighboring databases D and D', their frequency vectors x and x' will differ in one position, by exactly 1.

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}}_{\text{answers}} = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix}}_{\text{query matrix } \mathbf{W}} \text{ x } \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \quad +1 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix}}_{\mathbf{x'}}$$

The sensitivity of a query matrix

- For two neighboring databases D and D', their frequency vectors x and x' will differ in one position, by exactly 1.

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}}_{\text{answers}} = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix}}_{\text{query matrix } \mathbf{W}} \text{ x } \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ \boxed{x_7 \quad +1} \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix}}_{\mathbf{x'}}$$

The sensitivity of a query matrix

- For two neighboring databases D and D’, their frequency vectors x and x’ will differ in one position, by exactly 1.

$$\underbrace{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}}_{\text{answers}} = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix}}_{\text{query matrix } \mathbf{W}} \times \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \quad +1 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix}}_{\mathbf{x'}}$$

$$\boxed{||\mathbf{W}||_1 = 4}$$

The sensitivity of a query matrix

- For two neighboring databases D and D’, their frequency vectors x and x’ will differ in one position, by exactly 1.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix} \text{ x } \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \quad +1 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix}$$

answers query matrix $\mathbf{W}$ $\mathbf{x'}$

$$||\mathbf{W}||_1 = 4$$

The **L_1 sensitivity** of a query matrix is:
the maximum L1 norm of the columns.

The sensitivity of a query matrix

- For two neighboring databases D and D', their frequency vectors x and x' will differ in one position, by exactly 1.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 & \boxed{1} & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \;\; +1 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix}$$

answers query matrix $\mathbf{W}$ $\mathbf{x'}$

$\|\mathbf{W}\|_1 = 4$

The $\mathbf{L_1}$ **sensitivity** of a query matrix is:
the maximum L1 norm of the columns.

(The $\mathbf{L_2}$ **sensitivity** of a query matrix is:
the maximum L2 norm of the columns.)

Laplace mechanism (matrix notation)

$$\mathbf{Laplace(W,x) = Wx + (||W||_1 / \varepsilon)b}$$

Laplace mechanism (matrix notation)

$$\mathbf{Laplace(W,x) = Wx + (||W||_1 / \varepsilon)b}$$

$$\underbrace{\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix}}_{\substack{\textbf{private} \\ \textbf{output}}} = \begin{array}{c} \\ \textbf{m} \\ \textbf{queries} \end{array} \overbrace{\underbrace{\begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & & & \vdots \\ w_{m1} & w_{m2} & \dots & w_{mn} \end{bmatrix}}_{\substack{\textbf{workload of queries} \\ \mathbf{W}}}}^{\textbf{n columns}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{\substack{\textbf{data} \\ \mathbf{x}}} + (||\mathbf{W}||_1/\varepsilon) \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}}_{\substack{\textbf{noise} \\ \mathbf{b}}}$$

m independent samples from Laplace(1)

Laplace mechanism (matrix notation)

$$\mathbf{Laplace(W,x) = Wx + (||W||_1 / \varepsilon)b}$$

$$
\underbrace{\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix}}_{\text{private output}}
=
\begin{array}{c} \text{m queries} \end{array}
\underbrace{\overbrace{\begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \vdots & & & \vdots \\ w_{m1} & w_{m2} & \dots & w_{mn} \end{bmatrix}}^{\text{n columns}}}_{\substack{\text{workload of queries} \\ \mathbf{W}}}
\underbrace{\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}}_{\substack{\text{data} \\ \mathbf{x}}}
+ (||W||_1/\varepsilon)
\underbrace{\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}}_{\substack{\text{noise} \\ \mathbf{b}}}
$$

$$\mathbf{Error(w)} = 2\,(||W||_1 / \varepsilon)^2$$

m independent samples from Laplace(1)

The matrix mechanism

est-$\underline{x}$ can be viewed as a synthetic database.

Workload query answers consistent

The matrix mechanism

est-x can be viewed as a synthetic database.

Workload query answers consistent

1. (**Design**) Choose a **full rank** query strategy $\mathbf{A}$

The matrix mechanism

est-x can be viewed as a synthetic database.

Workload query answers consistent

1. (**Design**) Choose a **full rank** query strategy $\mathbf{A}$
2. (**Apply Laplace**) Use the Laplace mechanism to answer $\mathbf{A}$

The matrix mechanism

est-x can be viewed as a synthetic database.

Workload query answers consistent

1. (**Design**) Choose a **full rank** query strategy $\mathbf{A}$
2. (**Apply Laplace**) Use the Laplace mechanism to answer $\mathbf{A}$

$$\mathbf{z} = \mathbf{A}\mathbf{x} + (||\mathbf{A}||_1 / \varepsilon)\mathbf{b}$$

The matrix mechanism

est-$\underline{\mathbf{x}}$ can be viewed as a synthetic database.

Workload query answers consistent

❶ (**Design**) Choose a **full rank** query strategy $\mathbf{A}$

❷ (**Apply Laplace**) Use the Laplace mechanism to answer $\mathbf{A}$

$$\mathbf{z} = \mathbf{A}\mathbf{x} + (\|\mathbf{A}\|_1 / \varepsilon)\mathbf{b}$$

❸ (**Derivation**) Compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ using answers $\mathbf{z}$.

The matrix mechanism

est-$\underline{\mathbf{x}}$ can be viewed as a synthetic database.

Workload query answers consistent

❶ (**Design**) Choose a **full rank** query strategy $\mathbf{A}$

❷ (**Apply Laplace**) Use the Laplace mechanism to answer $\mathbf{A}$

$$\mathbf{z} = \mathbf{Ax} + (||\mathbf{A}||_1 / \varepsilon)\mathbf{b}$$

❸ (**Derivation**) Compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ using answers $\mathbf{z}$.

- compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ that minimizes squared error: $\| \mathbf{A}\underline{\mathbf{x}} - \mathbf{z} \|_2^2$

The matrix mechanism

est-$\underline{x}$ can be viewed as a synthetic database.

Workload query answers consistent

1. (**Design**) Choose a **full rank** query strategy $\mathbf{A}$
2. (**Apply Laplace**) Use the Laplace mechanism to answer $\mathbf{A}$

$$\mathbf{z} = \mathbf{A}\mathbf{x} + (||\mathbf{A}||_1 / \varepsilon)\mathbf{b}$$

3. (**Derivation**) Compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ using answers $\mathbf{z}$.
   - compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ that minimizes squared error: $\| \mathbf{A}\underline{\mathbf{x}} - \mathbf{z} \|_2^2$

$$\underline{\mathbf{x}} = \mathbf{A}^{+}\mathbf{z} \qquad \textbf{where } \mathbf{A}^{+} = (\mathbf{A}^{T}\mathbf{A})^{-1}\mathbf{A}^{T}$$

The matrix mechanism

est-$\underline{x}$ can be viewed as a synthetic database.

Workload query answers consistent

1. (**Design**) Choose a **full rank** query strategy $\mathbf{A}$
2. (**Apply Laplace**) Use the Laplace mechanism to answer $\mathbf{A}$

$$\mathbf{z} = \mathbf{A}\mathbf{x} + (||\mathbf{A}||_1 / \varepsilon)\mathbf{b}$$

3. (**Derivation**) Compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ using answers $\mathbf{z}$.
   - compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ that minimizes squared error: $\| \mathbf{A}\underline{\mathbf{x}} - \mathbf{z} \|_2^2$

Thm: $\underline{x}$ is unbiased and has the least variance among all linear unbiased estimators.

$$\underline{\mathbf{x}} = \mathbf{A}^{+}\mathbf{z}$$

where $\mathbf{A}^{+} = (\mathbf{A}^{T}\mathbf{A})^{-1}\mathbf{A}^{T}$

Strategy matrices for the range queries

Strategy matrices for the range queries

Identity

1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

$\mathbf{I}$

Hierarchical

1	1	1	1
1	1	0	0
0	0	1	1
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

$\mathbf{H}$

Wavelet

1	1	1	1
1	1	-1	-1
1	-1	0	0
0	0	1	-1

$\mathbf{Y}$

The matrix mechanism

Given a workload $\mathbf{W}$, and any full-rank strategy matrix $\mathbf{A}$, the following randomized algorithm is ε-differentially private:

$$\mathbf{Matrix_A(W,x) = Wx + (||A||_1 / \varepsilon)\, WA^+ b}$$

$\mathbf{b=Lap(1)}$

The matrix mechanism

Given a workload $\mathbf{W}$, and any full-rank strategy matrix $\mathbf{A}$, the following randomized algorithm is ε-differentially private:

$$\mathbf{Matrix_A(W,x) = Wx + (||A||_1 / \varepsilon)\, WA^+ b}$$

$\mathbf{b = Lap(1)}$

instantiated with strategy A

The matrix mechanism

Given a workload $\mathbf{W}$, and any full-rank strategy matrix $\mathbf{A}$, the following randomized algorithm is ε-differentially private:

$$\mathbf{Matrix_A(W,x) = Wx + (||A||_1 / \varepsilon)\, WA^+ b}$$

$\mathbf{b = Lap(1)}$

instantiated with strategy A

true answer

The matrix mechanism

Given a workload $\mathbf{W}$, and any full-rank strategy matrix $\mathbf{A}$, the following randomized algorithm is ε-differentially private:

$$\mathbf{Matrix_A(W,x) = Wx + (||A||_1 / \varepsilon)\ WA^+ b} \qquad \mathbf{b = Lap(1)}$$

instantiated with strategy A

true answer

scaling by $\|\mathbf{A}\|_1$

The matrix mechanism

Given a workload $\mathbf{W}$, and any full-rank strategy matrix $\mathbf{A}$, the following randomized algorithm is ε-differentially private:

$$\mathbf{Matrix}_{\mathbf{A}}(\mathbf{W},\mathbf{x}) = \mathbf{Wx} + (||\mathbf{A}||_1 / \varepsilon)\, \mathbf{WA}^+\, \mathbf{b}$$

$\mathbf{b}$=Lap(1)

instantiated with strategy A

true answer

scaling by $||\mathbf{A}||_1$

transformation by $\mathbf{WA}^+$

The matrix mechanism

Given a workload $\mathbf{W}$, and any full-rank strategy matrix $\mathbf{A}$, the following randomized algorithm is ε-differentially private:

$$\mathbf{Matrix_A(W,x) = Wx + (||A||_1 / \varepsilon)\ WA^+ b}$$

$\mathbf{b=Lap(1)}$

- instantiated with strategy A
- true answer
- scaling by $||\mathbf{A}||_1$
- transformation by $\mathbf{WA}^+$

Compare with the Laplace mechanism:

$$\mathbf{Laplace(W,x) = Wx + (||W||_1 / \varepsilon) b}$$

Error of the matrix mechanism

NOTE: This error is completely independent of the input data!!

Given any full rank strategy $\mathbf{A}$, and any linear workload query $\mathbf{w}$, the error of the mechanism $\mathbf{Matrix}_{\mathbf{A}}$ on query $\mathbf{w}$ is:

$$\mathbf{Error}_{\mathbf{A}}(\mathbf{w}) = (2/\varepsilon^2)\ (||\mathbf{A}||_1)^2\ \ \mathbf{w}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{w}^T$$

NOTE: This error is completely independent of the input data!!

Error of the matrix mechanism

Given any full rank strategy $\mathbf{A}$, and any linear workload query $\mathbf{w}$, the error of the mechanism $\mathbf{Matrix_A}$ on query $\mathbf{w}$ is:

$$\mathbf{Error_A(w)} = (2/\varepsilon^2)\ (||\mathbf{A}||_1)^2\ \ \mathbf{w(A^TA)^{-1}w^T}$$

sensitivity term

Error of the matrix mechanism

NOTE: This error is completely independent of the input data!!

Given any full rank strategy **A**, and any linear workload query **w**, the error of the mechanism $\mathbf{Matrix_A}$ on query **w** is:

$$\mathbf{Error_A(w)} = (2/\varepsilon^2)\,(||\mathbf{A}||_1)^2\;\; \mathbf{w}(\mathbf{A}^T\mathbf{A})^{-1}\mathbf{w}^T$$

sensitivity term

"error profile" term

Profile equivalence

Definition: strategy matrices $\mathbf{A}$ and $\mathbf{B}$ are **profile equivalent** if $(\mathbf{A}^{\mathrm{T}}\mathbf{A}) = (\mathbf{B}^{\mathrm{T}}\mathbf{B})$

$\mathbf{Matrix_A}$ $\quad \mathbf{Error_A(w)} = (2/\varepsilon^2)\ (||\mathbf{A}||_1)^2\ \mathbf{w}(\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{w}^{\mathrm{T}}$

$\mathbf{Matrix_B}$ $\quad \mathbf{Error_B(w)} = (2/\varepsilon^2)\ (||\mathbf{B}||_1)^2\ \mathbf{w}(\mathbf{B}^{\mathrm{T}}\mathbf{B})^{-1}\mathbf{w}^{\mathrm{T}}$

If $(\mathbf{A}^{\mathrm{T}}\mathbf{A}) = (\mathbf{B}^{\mathrm{T}}\mathbf{B})$ and $||\mathbf{A}||_1 \leq ||\mathbf{B}||_1$ then $\mathbf{Matrix_A}$ has lower error than $\mathbf{Matrix_B}$ for **every** query.

Strategies equivalent to wavelet

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \equiv \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} > \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ \sqrt{2} & 0 & 0 & 0 \\ 0 & \sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & 0 \\ 0 & 0 & 0 & \sqrt{2} \end{bmatrix}$$

Wavelet Y — $||Y||_1 = 3$

Y′ — $||Y'||_1 = 3$

Y″ — $||Y''||_1 = 2.414$

Strategies equivalent to wavelet

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix} \equiv \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} > \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ \sqrt{2} & 0 & 0 & 0 \\ 0 & \sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & 0 \\ 0 & 0 & 0 & \sqrt{2} \end{bmatrix}$$

Wavelet Y — $||Y||_1 = 3$

Y′ — $||Y'||_1 = 3$

Y″ — $||Y''||_1 = 2.414$

Neither the hierarchical nor the wavelet strategy is **efficient**, i.e. there exist uniformly better strategies with matching error profiles.

Finding the optimal strategy

Objective: given workload $\mathbf{W}$, find the query strategy $\mathbf{A}$ that minimizes the total error.

Error for a single query:

$$\mathbf{Error_A(w)} = (2/\varepsilon^2)(||\mathbf{A}||_1)^2\ \mathbf{w(A^TA)^{-1}w^T}$$

Total error for a workload of queries:

$$\begin{aligned}\mathbf{TotalError_A(w)} &= (2/\varepsilon^2)(||\mathbf{A}||_1)^2\ \mathbf{trace(\ W(A^TA)^{-1}W^T\ )} \\ &= (2/\varepsilon^2)(||\mathbf{A}||_1)^2\ \mathbf{trace(\ W^TW(A^TA)^{-1}\ )}\end{aligned}$$

Overview of problem solutions

Overview of problem solutions

	Objective	Problem Type

Overview of problem solutions

	Objective	Problem Type
1	Given $\mathbf{W}$, choose $\mathbf{A}$ to minimize $\mathbf{TotalError_A(W)}$	SDP with rank constraints

Overview of problem solutions

	Objective	Problem Type
1	Given $\mathbf{W}$, choose $\mathbf{A}$ to minimize $\mathbf{TotalError_A(W)}$	SDP with rank constraints
2	Given $\mathbf{A^T A}$, choose $\mathbf{Q}$ to minimize $\|\|\mathbf{A}\|\|_1$	SDP with rank constraints

Overview of problem solutions

	Objective	Problem Type
1	Given $\mathbf{W}$, choose $\mathbf{A}$ to minimize $\mathbf{TotalError_A(W)}$	SDP with rank constraints
2	Given $\mathbf{A^TA}$, choose $\mathbf{Q}$ to minimize $\|\|\mathbf{A}\|\|_1$	SDP with rank constraints
3	Given $\mathbf{W}$, choose $\mathbf{A}$ to minimize $\mathbf{TotalError_A(W)}$ under (ε,δ)-differential privacy	SDP

Outline

1. Case study: answering 1-dim range queries
2. The matrix mechanism -- formal description.
3. The matrix mechanism -- new tools & techniques.
4. Conclusion

Outline

1. Case study: answering 1-dim range queries
2. The matrix mechanism -- formal description.
3. The matrix mechanism -- new tools & techniques.
4. Conclusion

New techniques

- **Optimal error**: A lower bound on the error of the optimal strategy allows us to assess the quality of existing strategies and explore “workload error complexity”.

- **Efficient strategy selection**: the strategy selection problem can be approximately solved, resulting in strategy matrices customized to arbitrary workloads.

- **Inference for sparse datasets:** by imposing non-negativity constraints during inference, accuracy can be significantly improved. (But analysis of error is harder.)

Singular value bound

- Given workload W, the optimal total error for W is greater than or equal to the SVD bound.

> THEOREM 3.3. (SINGULAR VALUE BOUND) *Given an $m \times n$ workload $\mathbf{W}$, let $\lambda_1, \lambda_2, \ldots, \lambda_n$ be the singular values of $\mathbf{W}$.*
>
> $$\min_{\mathbf{A}} \mathrm{ERROR}_{\mathbf{A}}(\mathbf{W}) \geq P(\epsilon, \delta) \frac{1}{n} (\sum_{i=1}^{n} \lambda_i)^2,$$
>
> *where* $P(\epsilon, \delta) = \frac{2 \log(2/\delta)}{\epsilon^2}$.

- Tight: bound is achievable for a certain class of workloads.

- Easy to compute.

Algorithm for efficient strategy selection

- Inspired by **optimal experimental design**
  - Given $\mathbf{W}$, choose a set of **basis queries** for the strategy:
    - $\mathbf{v_1}, \mathbf{v_2}, \ldots \mathbf{v_n}$ (the eigenvectors of $\mathbf{W}$)
    - compute optimal scalars to minimize error

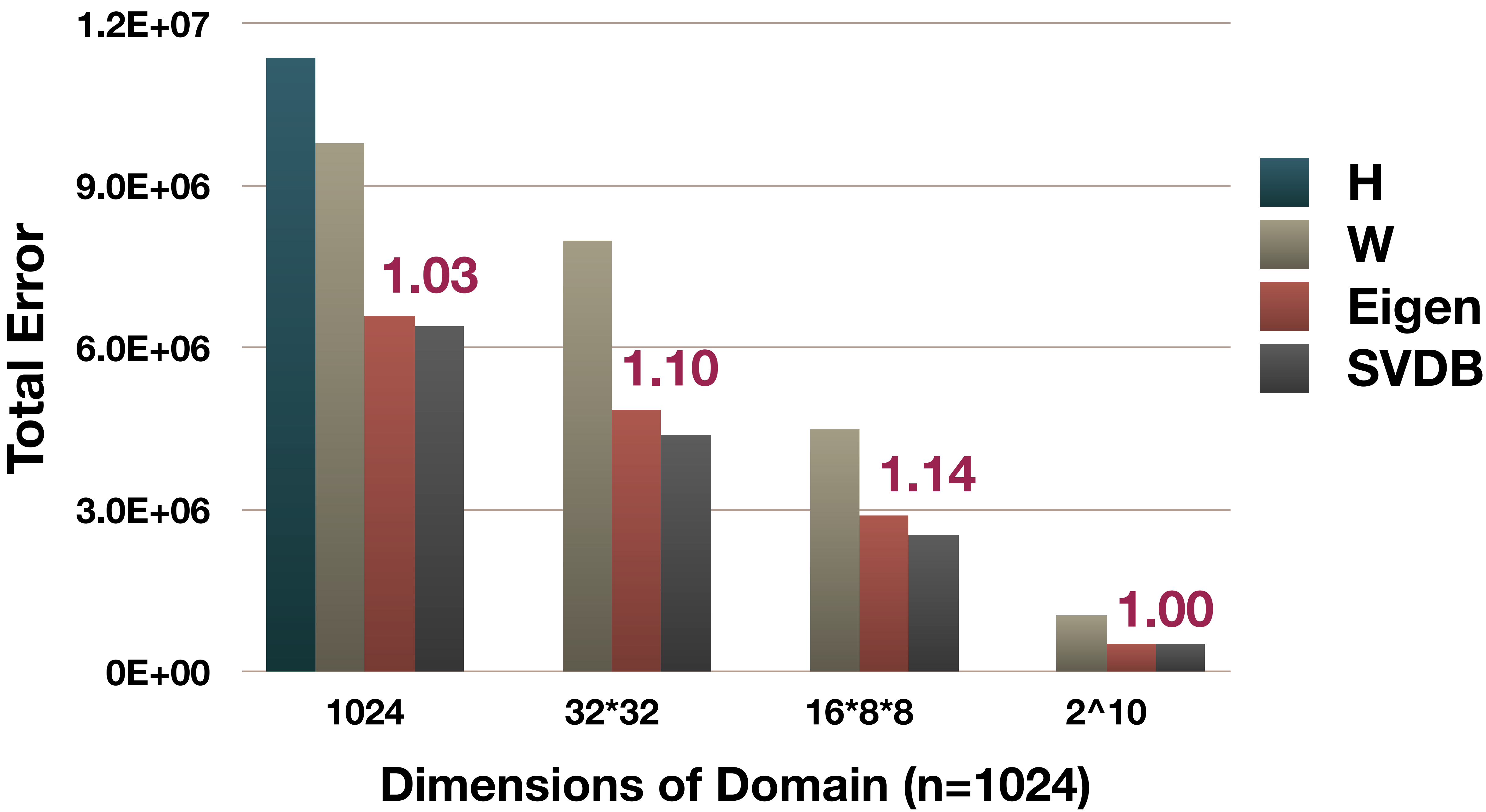
      $c_1, c_2, \ldots c_n$
- Resulting strategy matrix is:

$$\mathbf{A} = \begin{bmatrix} c_1\mathbf{v_1} \\ c_2\mathbf{v_2} \\ \ldots \\ c_n\mathbf{v_n} \end{bmatrix}$$

Approximately optimal error rates

All range queries

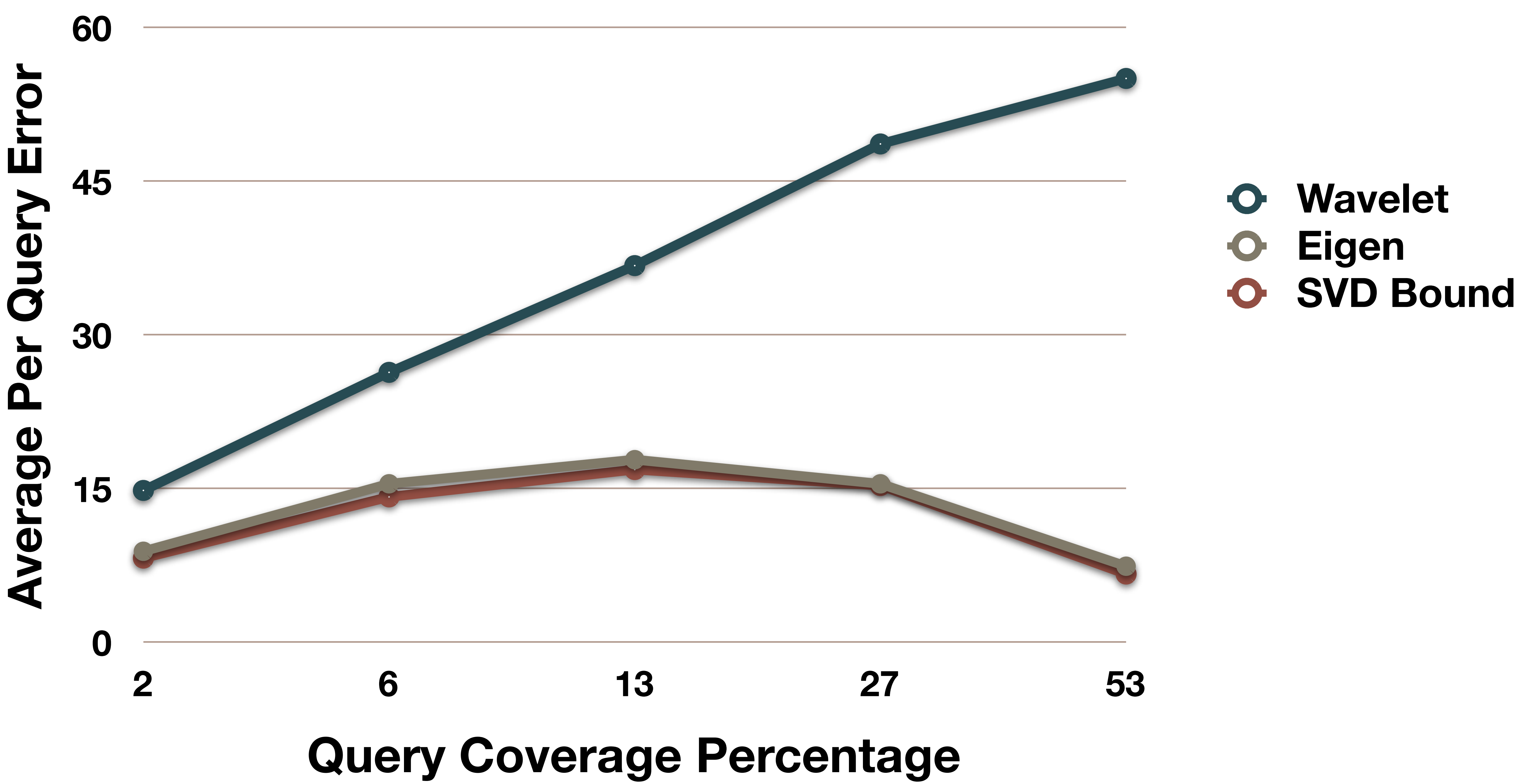
(ε,δ)-differential privacy



Customizing the strategy to the workload

Subsets of the range queries

(ε,δ)-differential privacy



Non-negative least squares

Non-negative least squares

❶ (**Design**) Choose a **full rank** query strategy $\mathbf{A}$

❷ (**Apply Laplace**) Use the Laplace mechanism to answer $\mathbf{A}$

$$\mathbf{z} = \mathbf{A}\mathbf{x} + (\|\mathbf{A}\|_1 / \varepsilon)\mathbf{b}$$

❸ (**Inference**) Compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ using answers $\mathbf{z}$.

Non-negative least squares

❶ (**Design**) Choose a **full rank** query strategy $\mathbf{A}$

❷ (**Apply Laplace**) Use the Laplace mechanism to answer $\mathbf{A}$

$$\mathbf{z} = \mathbf{A}\mathbf{x} + (||\mathbf{A}||_1 / \varepsilon)\mathbf{b}$$

❸ (**Inference**) Compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ using answers $\mathbf{z}$.

- Non-negative least squares: compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ that minimizes squared error: $\|\mathbf{A}\underline{\mathbf{x}} - \mathbf{z}\|_2^2$

where each $\underline{\mathbf{x}}_{\mathbf{i}} > 0$

Non-negative least squares

1. (**Design**) Choose a **full rank** query strategy $\mathbf{A}$
2. (**Apply Laplace**) Use the Laplace mechanism to answer $\mathbf{A}$

$$\mathbf{z} = \mathbf{Ax} + (||\mathbf{A}||_1 / \varepsilon)\mathbf{b}$$

3. (**Inference**) Compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ using answers $\mathbf{z}$.
   - Non-negative least squares: compute estimate $\underline{\mathbf{x}}$ of $\mathbf{x}$ that minimizes squared error: $\|\mathbf{A}\underline{\mathbf{x}} - \mathbf{z}\|_2^2$

where each $\underline{x}_i > 0$

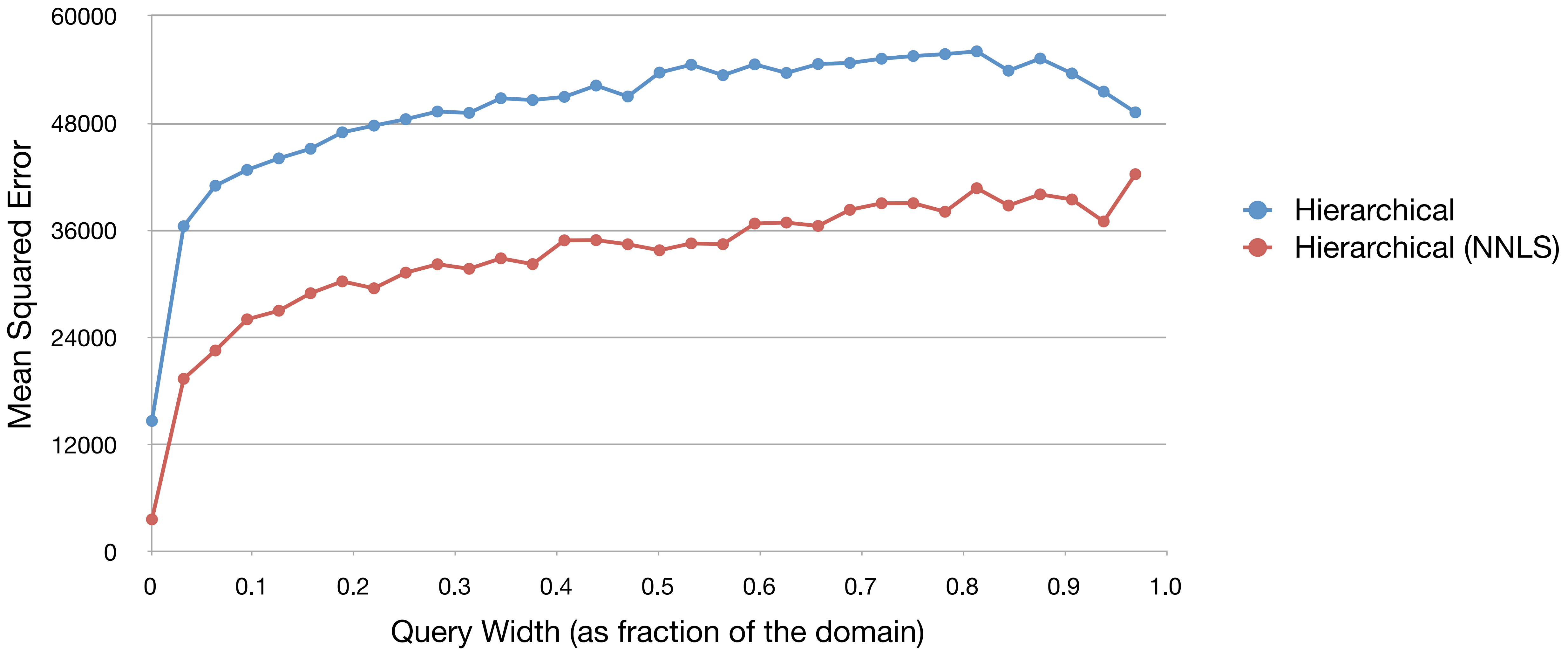
Effectiveness of non-negative least squares depends of properties of the data, epsilon, and $\mathbf{A}$.

Active set method [Lawson, 1987]

Error: all range queries, non-negative least squares

Synthetic sparse dataset

Epsilon = 0.1
n = 1024



Outline

1. Case study: answering 1-dim range queries
2. The matrix mechanism -- formal description.
3. The matrix mechanism -- new tools & techniques.
4. Conclusion

Outline

1. Case study: answering 1-dim range queries
2. The matrix mechanism -- formal description.
3. The matrix mechanism -- new tools & techniques.
4. Conclusion

Summary and conclusions

- The Matrix mechanism generalizes the Laplace & Gaussian mechanisms, improving accuracy by exploiting correlation in the workload and reducing sensitivity.

- Two recent techniques for range queries are instances of the matrix mechanism; neither is optimal, but they are close.

- One strategy does not fit all workloads: adapting the strategy to the workload is essential to achieving low error.

- It is possible to compute the optimal strategy in $O(n^8)$ time, and approximately optimal strategies in $O(n^4)$.

Open questions

- The matrix mechanism is data-independent. What are the trade-offs for data-dependent approaches?
- What makes one workload “harder” to answer than another? How can we reliably measure workload error complexity?
- How do our results compare with lower bounds for differentially private output. (Our optimal strategies result in the least error for this particular mechanism, not necessarily the lowest error possible.)
- Can we avoid the computational dependence on the domain size n, without sacrificing accuracy?
- How do we analyze the error resulting from non-negative least squares?

Questions?

Project page and implementation of the Matrix Mechanism:

http://bit.ly/ituyOt

Additional details on our work may be found here:

- **[Li, ArXiv 2011]** C. Li and G. Miklau. Efficient Batch Query Answering Under Differential Privacy. CoRR abs/1103.1367, 2011.
- **[Li, PODS 2010]** C. Li, M. Hay, V. Rastogi, G. Miklau, and A. McGregor. Optimizing Linear Counting Queries Under Differential Privacy. Principles of Database Systems (PODS) 2010.
- **[Hay, PVLDB 10]** M. Hay, V. Rastogi, G. Miklau, and D. Suciu. Boosting the accuracy of differentially-private queries through consistency. Proceedings of the VLDB Endowment (PVLDB), 2010.

References

[Ghosh, 2009] A. Ghosh, T. Roughgarden, and M. Sundararajan. Universally utility-maximizing privacy mechanisms. In Symposium on Theory of computing (STOC), pages 351–360, 2009.

[Xiao, 2010] X. Xiao, G. Wang, and J. Gehrke. Differential privacy via wavelet transforms. In International Conference on Data Engineering, 2010.

[Barak, 2007] B. Barak, K. Chaudhuri, C. Dwork, S. Kale, F. McSherry, and K. Talwar. Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In Principles of Database Systems (PODS) 2007.

[Xiao, 2011] Xiaokui Xiao, Gabriel Bender, Michael Hay, and Johannes Gehrke. iReduct: Differential privacy with reduced relative errors. SIGMOD, 2011.

[Lawson, 1987] C. L. Lawson and R. J. Hanson, Solving least squares Problems, Prentice Hall, 1987.