CS208: Applied Privacy for Data Science Implementing Differential Privacy: Programming Interfaces for DP

James Honaker & Salil Vadhan

School of Engineering & Applied Sciences
Harvard University

March 4, 2019

Synthetic Data via DP Histograms

- Use singleton bins $B_y = \{y\}$ for each $y \in \mathcal{Y}$.
- Construct a DP histogram $\left(a_1, \dots, a_{|\mathcal{X}|}\right) \leftarrow M_{\text{hist}}(x)$, where $a_y \approx \#\{i : x_i = y\}$.
- Output synthetic dataset $\hat{x}$ with a_y copies of each element y.

Difficulties?

- a_y's may not be nonnegative integers.
  - Soln 1: use Geometric Mechanism and clamp at 0.
  - Soln 2: use Exponential Mechanism with range $\{0, \dots, n\}$.
- Poor utility & efficiency when $\mathcal{X}$ is large.
- Enforcing nonnegativity introduces problematic bias!

Census Bureau's Use of DP

Excerpts from:

- Simson Garfinkel "Challenges and Experiences Adapting Differentially Private Mechanisms to the 2020 Census," FCSM 2018.

See also:

- John Abowd. "The U.S. Census Bureau Adopts Differential Privacy," KDD 2018.
  [how to decide on privacy vs. accuracy]
- Dan Kifer. "Consistency with External Knowledge: The Top-Down Algorithm," Simons Privacy workshop TODAY.
  [many algorithmic issues and choices]
- Aref Danjani. "The modernization of statistical disclosure limitation at the U.S. Census Bureau," UNECE/EUROSTAT 2017.
  [challenges for other Census products]

Consistency & Optimization

- Structural Zeroes: Enforced by edit and imputation, DP can't reintroduce it
  - Householder and spouse/partner must be at least 15 yrsol
  - Every household must have exactly one householder
  - At least one of the binary race flags must be 1
  - Etc.

- Invariants: public statistics with exact values
  - State population totals
  - Linear constraints: sum of county populations equals state population
  - Single-gender group quarters (dorms, prisons)

- Optimizing accuracy: for a set Q of queries
  - Use "matrix mechanism" to determine related set Q' of queries, apply Laplace mechanism to Q' , then reconstruct synthetic data.
  - With constraints, NP-hard: use integer programming heuristics.

[LeClerc-Clark-Sexton `17, Kifer `19]

The Matrix Mechanism

[Li-Miklau-Hay-Rastogi `14]

- Common Approach: given "workload" $Q = (q_1, \ldots, q_k)$, find
  - A "query strategy" $Q' = (q'_1, \ldots, q'_\ell)$ that can be answered accurately with DP
  - A transformation B that maps accurate answers to Q' to accurate answers for Q.

- Example:
  - $Q =$ simple linear regression coefficient β
  - Q' = (S_{xy}, S_{xx})
  - $B\left(a_{xy}, a_{xx}\right) = a_{xy}/a_{xx}$.

The Matrix Mechanism

[Li-Miklau-Hay-Rastogi `14]

- Matrix Mechanism: restrict attention to the "linear" case:
  - Q, Q' are of form $q_j(x) = \sum_{i=1}^{n} f_j(x_i)$ for $f_j : \mathcal{X} \to \mathbb{R}$
  - B is a $\ell \times k$ matrix s.t.
    - for all $x \in \mathcal{X}^n$, $(q_1(x), \ldots, q_k(x)) = B \cdot \left(q'_1(x), \ldots, q'_\ell(x)\right)$

    (or equivalently for f_j's).

- When answering Q' via Laplace or Gaussian mechanism, finding the Q' and B minimizing the MSE is a rank-constrained semidefinite program of size $k \times |\mathcal{X}|$.

Example: Range Queries

$\mathcal{X} = \{0, \ldots, D-1\}$, $Q = \left(q_{[a,b]}\right)_{0 \le a \le b \le D-1}$, where

$$q_{[a,b]}(x) = \#\{i : a \le x_i \le b\}$$

1st attempt: $Q' = Q$. $B = I$.

Changing one row of x can change $\Omega(D^2)$ answers by ± 1.

Laplace mechanism has std. dev. $\Theta(D^2/\varepsilon)$ per query.

2nd attempt: Q' = histogram queries $\left(q'_y\right)_{1 \le y \le D}$.

B: $q_{[a,b]}(x) = \sum_{a \le y \le b} q'_y(x)$

Laplace mechanism has std. dev $\Theta(1/\varepsilon)$ per query in Q'.

Std devation for $q_{[a,b]} =$

The Hierarchical Strategy

[Hay-Rastogi-Miklau-Suciu `10]

- Assume $D = 2^d$, arrange elements of $\mathcal{X}$ in a binary tree,
  let $Q' = \left(q_{[a,b]}\right)_{a,b}$ smallest and largest descendants of their common ancestor

- Changing one row changes at most $2d$ queries in Q', each by ± 1.

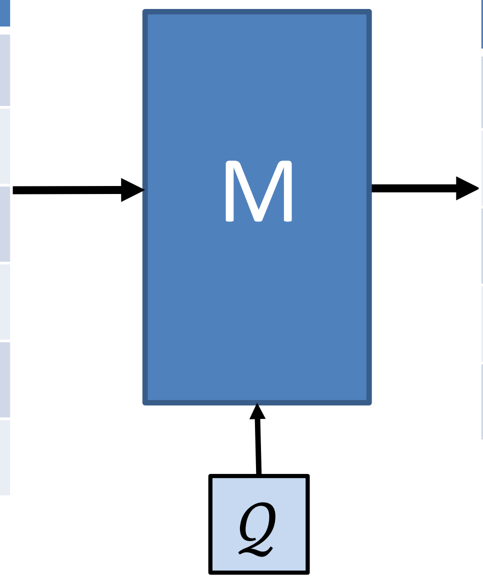
- Every range query in Q is a ± 1 linear combination of at most $2d$ queries in Q'.

- Standard deviation of error for queries in Q' is $O\left(d^{3/2}/\varepsilon\right)$.

Private Multiplicative Weights

[Blum-Ligett-Roth `08,…,Hardt-Rothblum `10]

Sex	Blood	⋯	HIV?
F	B	⋯	Y
M	A	⋯	N
M	O	⋯	N
M	O	⋯	Y
F	A	⋯	N
M	B	⋯	Y



Sex	Blood	⋯	HIV?
M	B	⋯	N
F	B	⋯	Y
M	O	⋯	Y
F	A	⋯	N
F	O	⋯	N

(ε, δ)-DP M: $\mathcal{X}^n \to \mathcal{X}^m$ such that $\forall q \in \mathcal{Q}$, $q: \mathcal{X} \to [0,1]$

$$\left|\frac{1}{n}\sum_{i=1}^{n} q(x_i) - \frac{1}{m}\sum_{i=1}^{m} q(M(x)_i)\right| \leq O\left(\frac{\sqrt{\log|\mathcal{X}| \cdot \log(1/\delta) \cdot \log|\mathcal{Q}|}}{\varepsilon n}\right)^{1/2}$$

Private Multiplicative Weights

[Hardt-Rothblum `10]

(ε, δ)-DP M: $\mathcal{X}^n \to \mathcal{X}^m$ such that $\forall q \in \mathcal{Q}$, $q: \mathcal{X} \to [0,1]$

$$\left|\frac{1}{n}\sum_{i=1}^{n} q(x_i) - \frac{1}{m}\sum_{i=1}^{m} q(M(x)_i)\right| \leq O\left(\frac{\sqrt{\log|\mathcal{X}| \cdot \log(1/\delta)} \cdot \log|\mathcal{Q}|}{\varepsilon n}\right)^{1/2}$$

Problem: computation time $\mathrm{poly}(n, |\mathcal{X}|, |\mathcal{Q}|)$.

- Exponential in dimensionality of data and query family.
- Inherent in the worst case (cf. “Complexity of DP”).

Private Mult. Weights & Dual Query

[Hardt-Rothblum `10, Gaboardi-Gallego Arias-Hsu-Roth-Wu `14]

(ε,δ)-DP $M\colon \mathcal{X}^n \to \mathcal{X}^m$ such that $\forall q \in \mathcal{Q},\ q\colon \mathcal{X} \to [0,1]$

$$\left|\frac{1}{n}\sum_{i=1}^{n} q(x_i) - \frac{1}{m}\sum_{i=1}^{m} q(M(x)_i)\right| \le O\left(\frac{\sqrt{\log|\mathcal{X}|\cdot\log(1/\delta)}\cdot\log|\mathcal{Q}|}{\varepsilon n}\right)^{1/2}$$

Problem: computation time $\mathrm{poly}(n, |\mathcal{X}|, |\mathcal{Q}|)$.

- Exponential in dimensionality of data and query family.
- Inherent in the worst case (cf. "Complexity of DP").

DualQuery:

- Use integer programming get heuristic runtime $\mathrm{poly}(n, \log|\mathcal{X}|, |\mathcal{Q}|)$.
- Privacy doesn't depend on success of heuristic.
- Proven accuracy a bit worse (exponent 1/3 instead of ½).

DualQuery Experiments I

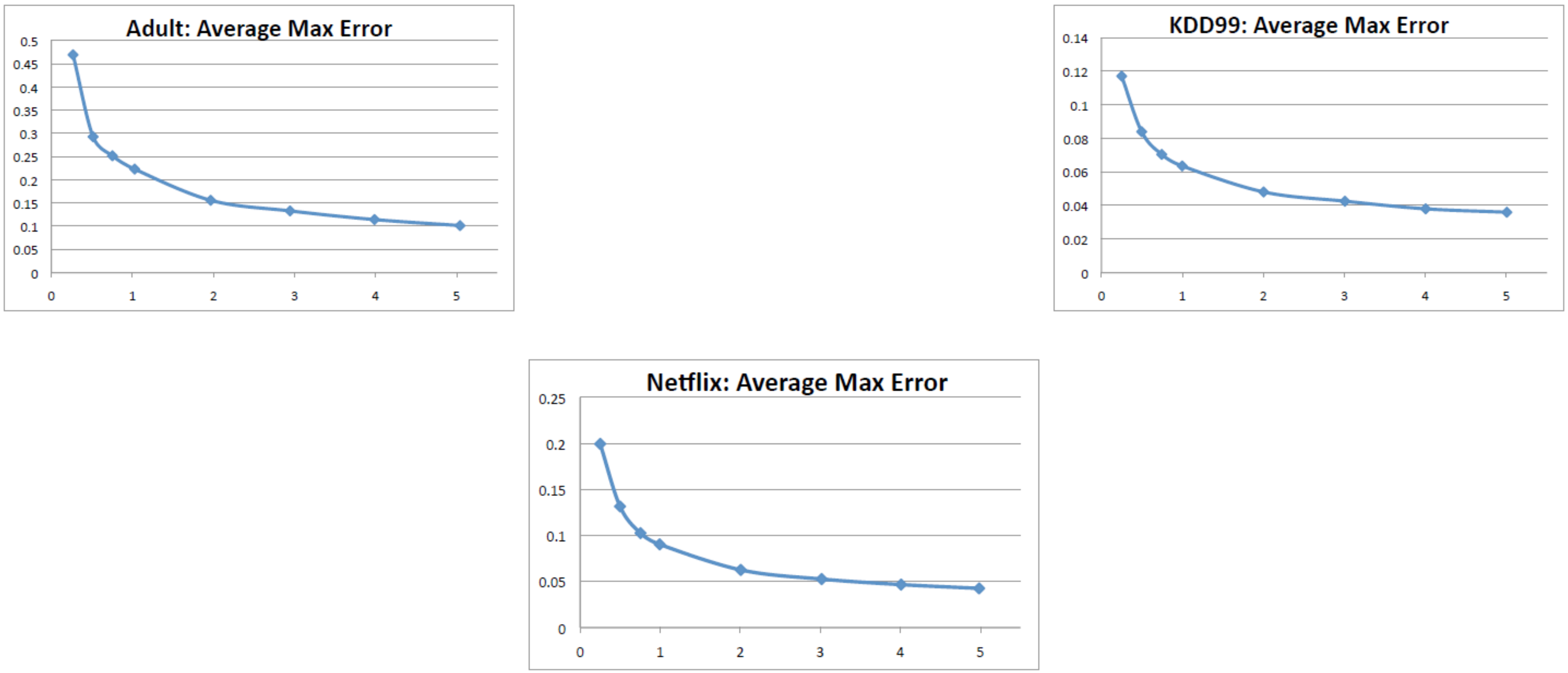


Figure 2: Average max error of $(\varepsilon, 0.001)$-private DualQuery on 500,000 3-way marginals versus ε.

DualQuery Experiments II

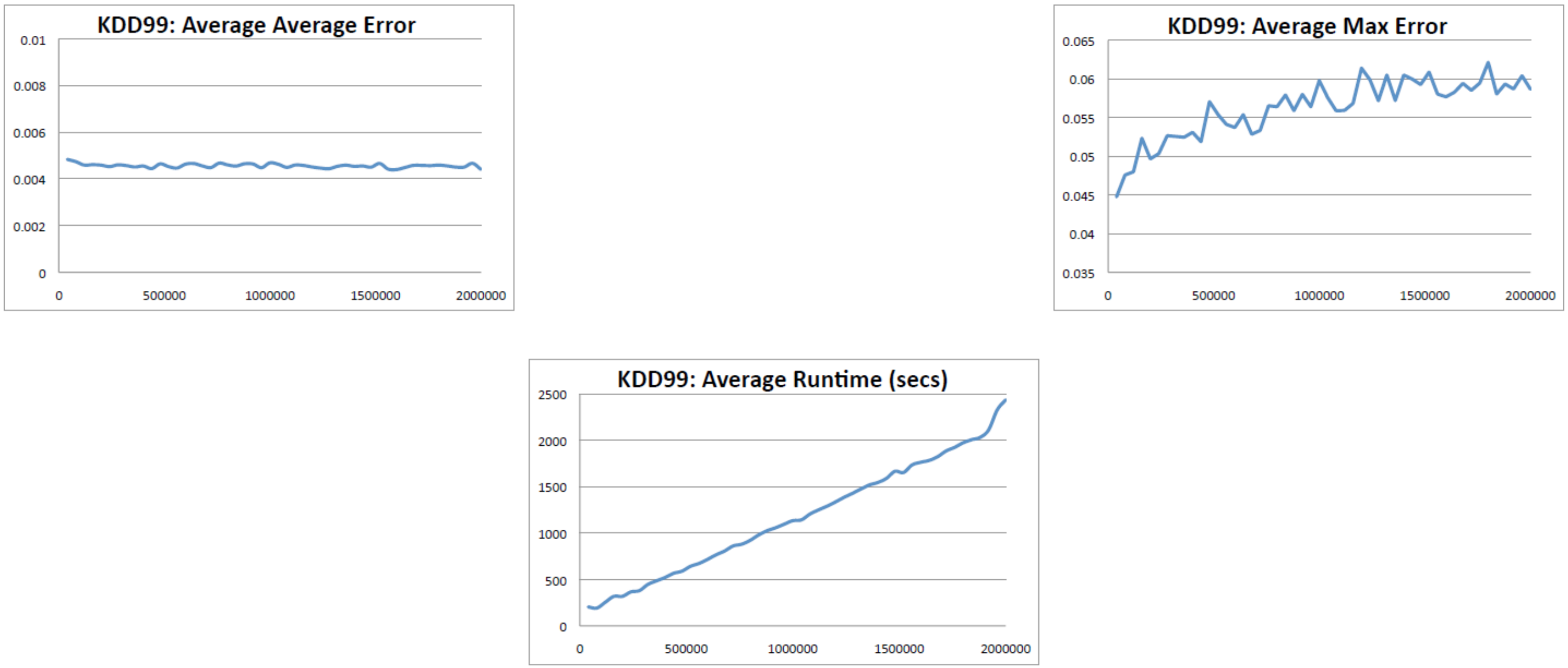


Figure 3: Error and runtime of $(1, 0.001)$-private DualQuery on KDD99 versus number of queries.

DualQuery Experiments III

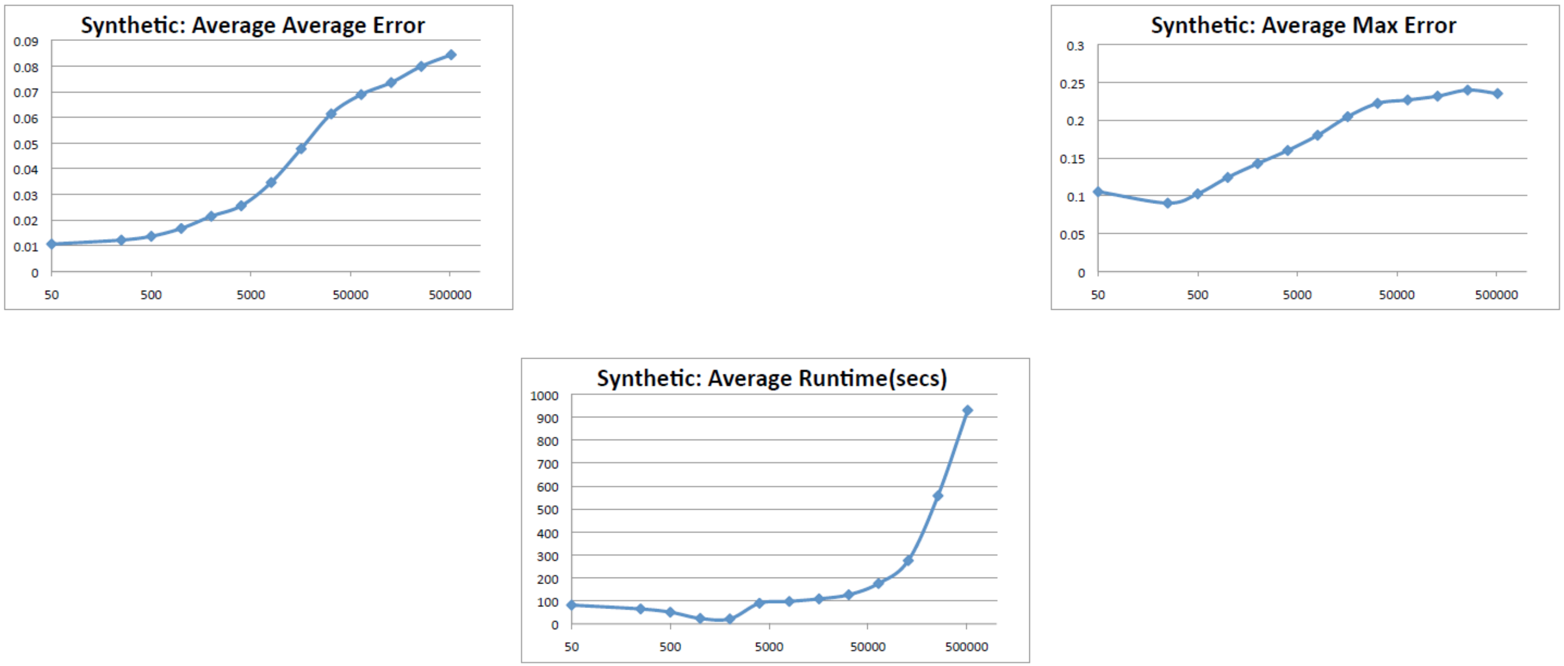


Figure 4: Error and runtime of $(1, 0.001)$-private DualQuery on 100,000 3-way marginal queries versus number of attributes.

See also: Hay et al. “Principled Evaluation of DP Algorithms using DPBench” and https://www.dpcomp.org/

Programming Frameworks for DP

Goal: make it easier for a data custodian or analyst to write programs that are DP, and be confident that they actually are DP.

Common approach (starting with PinQ [McSherry `09]):

- (Small) set of trusted DP subroutines: (Lap, Geo, ExpMech, …) only channel for info to flow from dataset to rest of program.
- Track privacy budget consumption: using composition of DP, with either a runtime monitor or static analysis.
- Allow "Lipschitz" data transformations: (recursively) track impact on privacy consumption.

Dataset Transformations

- Let $d(x, x')$ denote distance between datasets x, x'.
  - Number of rows on which they differ for public n model.
  - $|x \Delta x'|$ for unknown n model.

- Def: A mapping from datasets to datasets is c-Lipschitz (aka c-stable or c-sensitive) iff

$$\forall x, x' \ d\big(T(x), T(x')\big) \leq c \cdot d(x, x').$$

- Lemmas:
  - If M is ε-DP and T is c-Lipschitz, then $M \circ T$ is $c\varepsilon$-DP.
  - If T_1 is c_1-Lipschitz and T_2 is c_2-Lipschitz, then $T_2 \circ T_1$ is $c_1 c_2$-Lipschitz.

Calculate the Lipschitz Constants

- Per-row transforms (SELECT):
$$T\big((x_1, \dots, x_n)\big) = \big(f(x_1), \dots, f(x_n)\big).$$

- Winsorization: $T(x)$ = remove the bottom and top 20 elts (viewing x and $T(x)$ as unordered)

- Subsetting (WHERE): $T(x) = \{r \in x : \pi(r) = \text{true}\}$ (multiset) (use unknown n model)

Partitioning

- "Parallel Composition" Lemma: Let $S_1, \ldots, S_k$ be disjoint subsets of $\mathcal{X}$ and let $M_1, \ldots, M_k$ be ε-DP algorithms (for the unknown n model). Then $M(x) = \left(M_1\left(x|_{S_1}\right), \ldots, M_k\left(x|_{S_k}\right)\right)$ is ε-DP.

- A "1-Lipschitz" 1-to-k transformation $T(x) = \left(x|_{S_1}, \ldots, x|_{S_k}\right)$.

- Also have 2-to-1 transformations (Union, Intersection, Join).

Tracking Sensitivity

Transformation	Stability
Select(T, *maper*)	(1)
Where(T, *predicate*)	(1)
GroupBy(T_1, *keyselector*)	(2)
Join*(T_1,T_2, n, m, $keyselector_1$, $keyselector_2$)	(n,m)
Intersect(T_1,T_2)	(1,1)
Union(T_1,T_2)	(1,1)
Partition(T, *keyselector*, *keysList*)	(1)

Table 1. Transformation stability

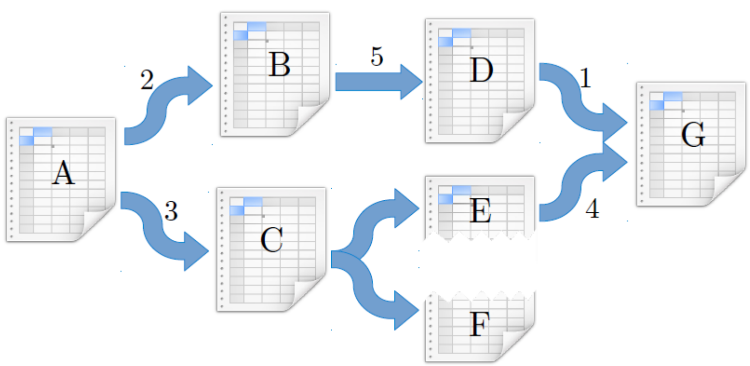


Fig. 2. Transformations

	s	Calculation
A	1	Input table
B	2	$s(A) \times 2$
C	1	$s(A) \times 3$
D	10	$s(B) \times 5$
E	3	$s(C)$
F	3	$s(C)$
G	22	$s(D) \times 1 + s(E) \times 4$

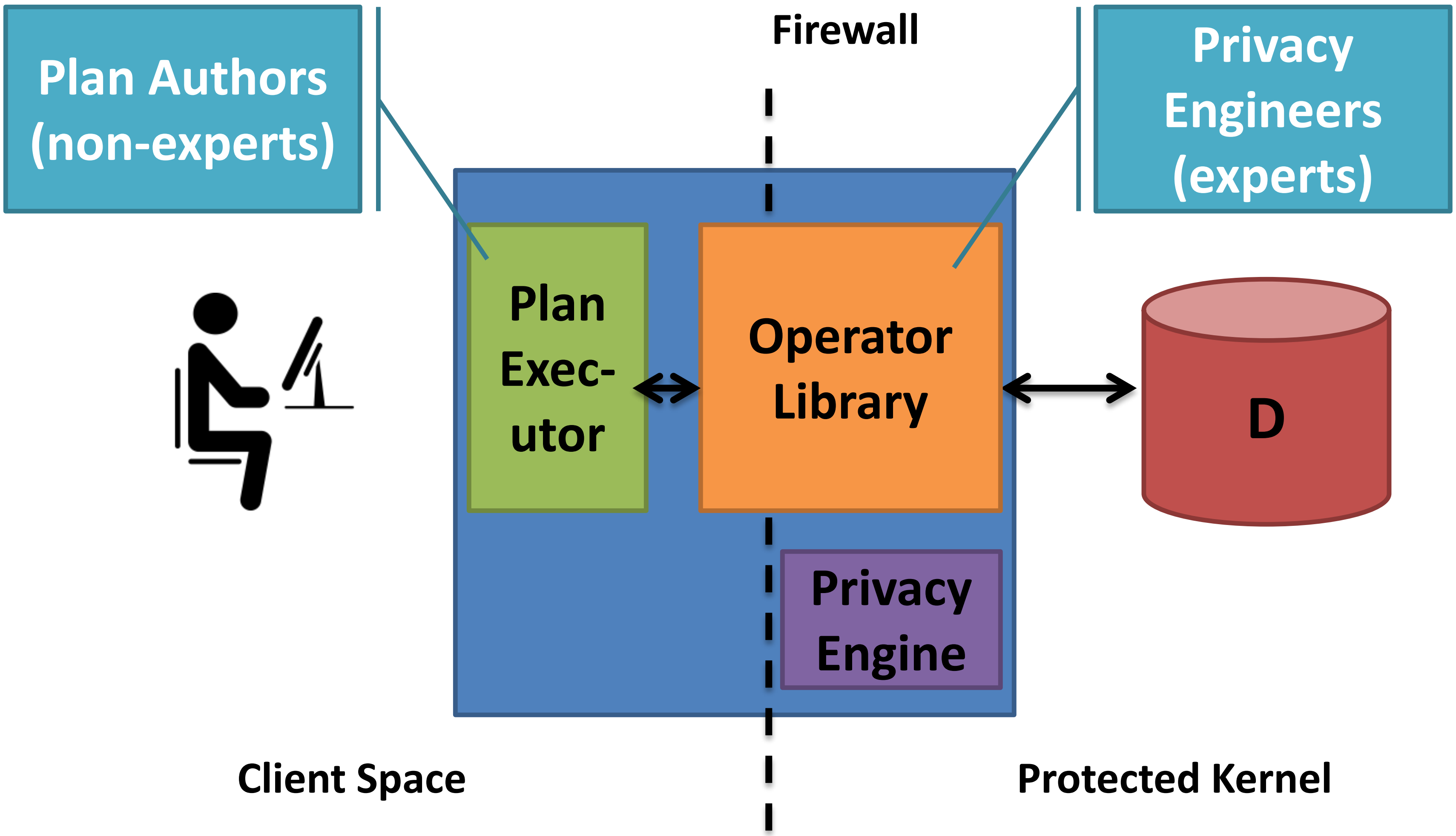
Fig. 3. Scaling factors (s)

[from Ebadi & Sands, "Featherweight PinQ", 2017]

εktelo

- Excerpts of Michael Hay’s talk “Making Privacy Technology Accessible: Benchmarks and Platforms” at Simons Institute workshop “Data Privacy: from Foundations to Applications,” March 8, 2019.

Ektelo Implementation



[slide from Ashwin Machanavajjhala]

Other Issues in Programming DP

- Multi-relational databases
  - Standard joins have unbounded Lipschitz constant, so need to truncate results or use "local sensitivity" approximations.
- Side-channel attacks
  - Info can be leaked through timing, approx. of real numbers, global state, exceptions, etc.
  - Constrain language & implementation to match model better.
- Verifying DP building blocks or more complex DP algs
  - Specialized programming languages.
  - Annotate programs with types to assist automated verification of DP.
  - Tradeoff between usability and expressiveness.
  - Now can even synthesize DP algorithms from examples!
- Guidance on Privacy Budgeting
  - Next time!
- Choice of Programming Model (e.g. SQL vs. MapReduce vs. R)