Differential Privacy

[and Analysis of Social Networks]

Kobbi Nissim

BGU/Harvard/BU

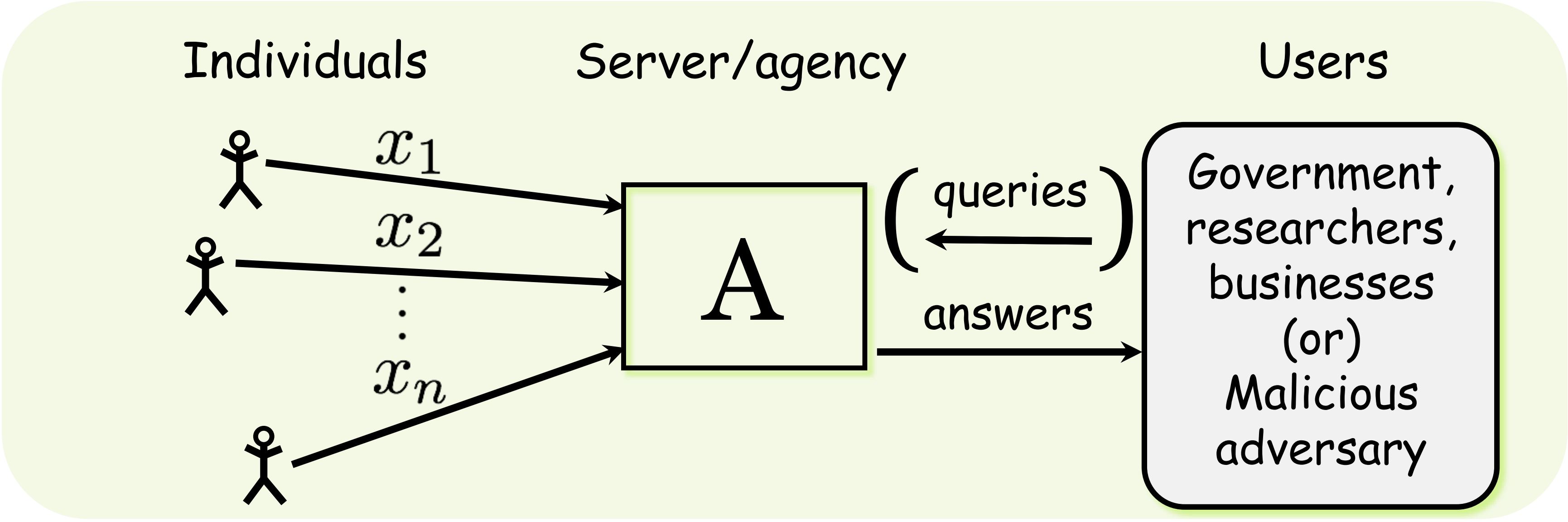
Charles River Workshop on Private Analysis of Social Networks
May 2014

Data Privacy – The Problem

- Given:
  - A dataset with sensitive information
- How to:
  - Compute and release functions of the dataset without compromising individual privacy

Data Privacy – The Problem

- Given:
  - A dataset with sensitive information
- How to:
  - Compute and release functions of the dataset without compromising individual privacy



Age of Miss America
compared with
Murders by steam, hot vapours and hot objects

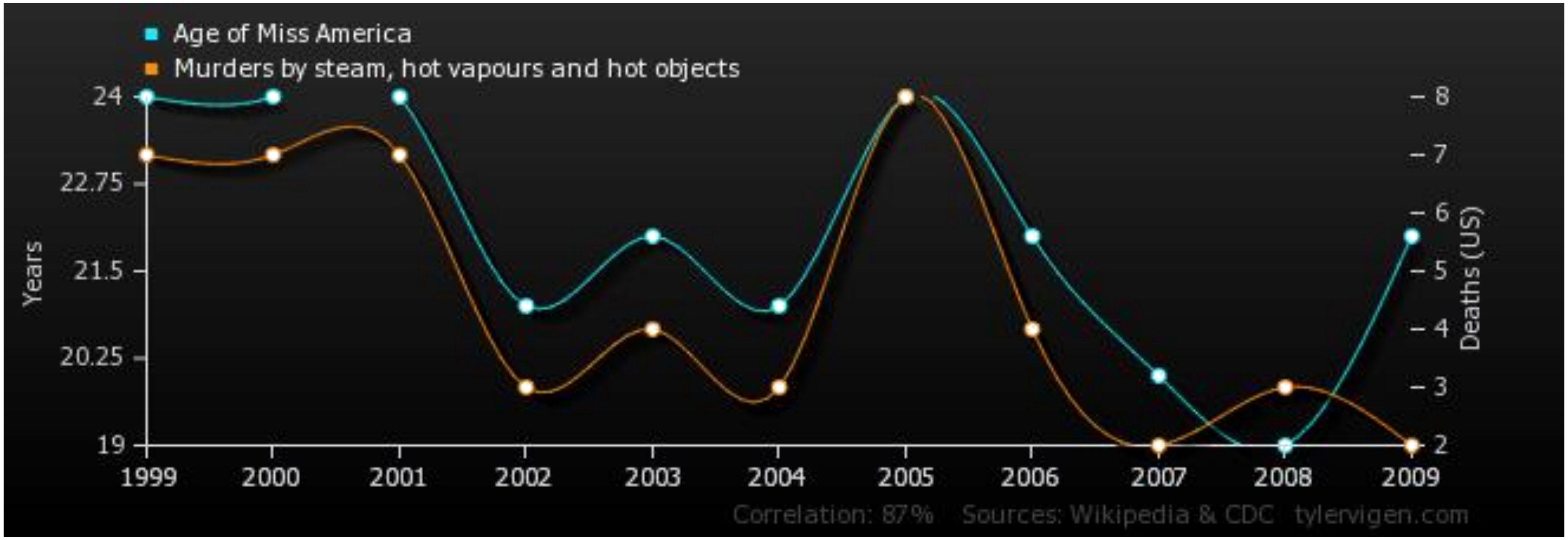
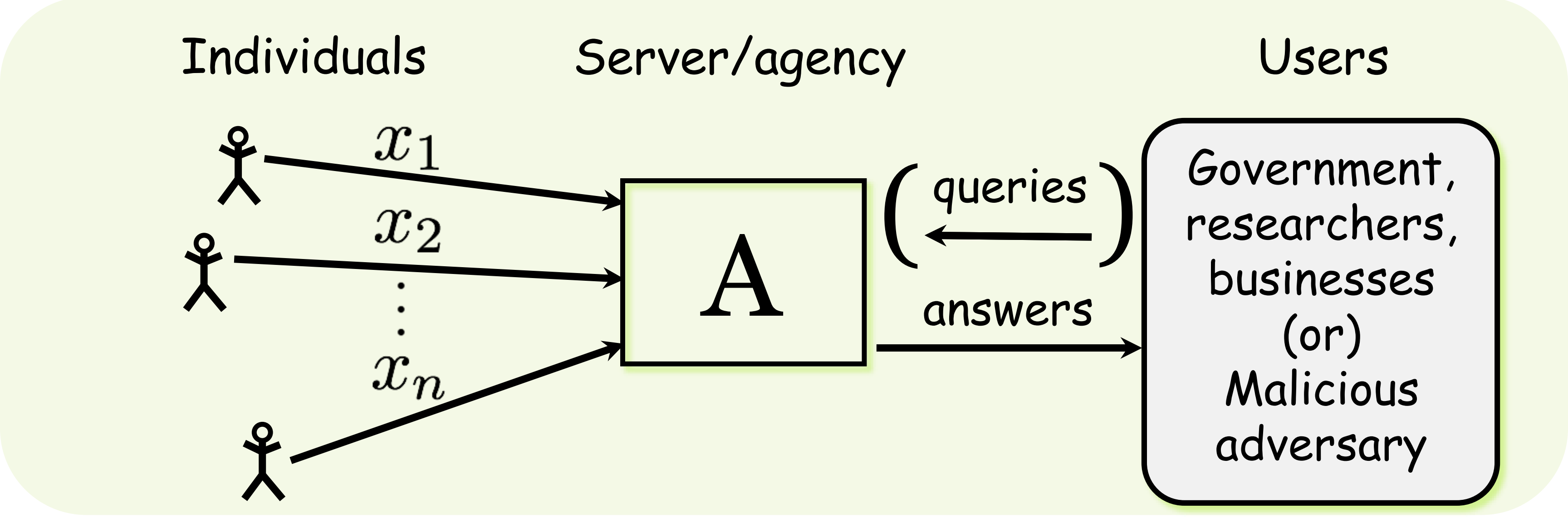


Image credit: www.tylervigen.com

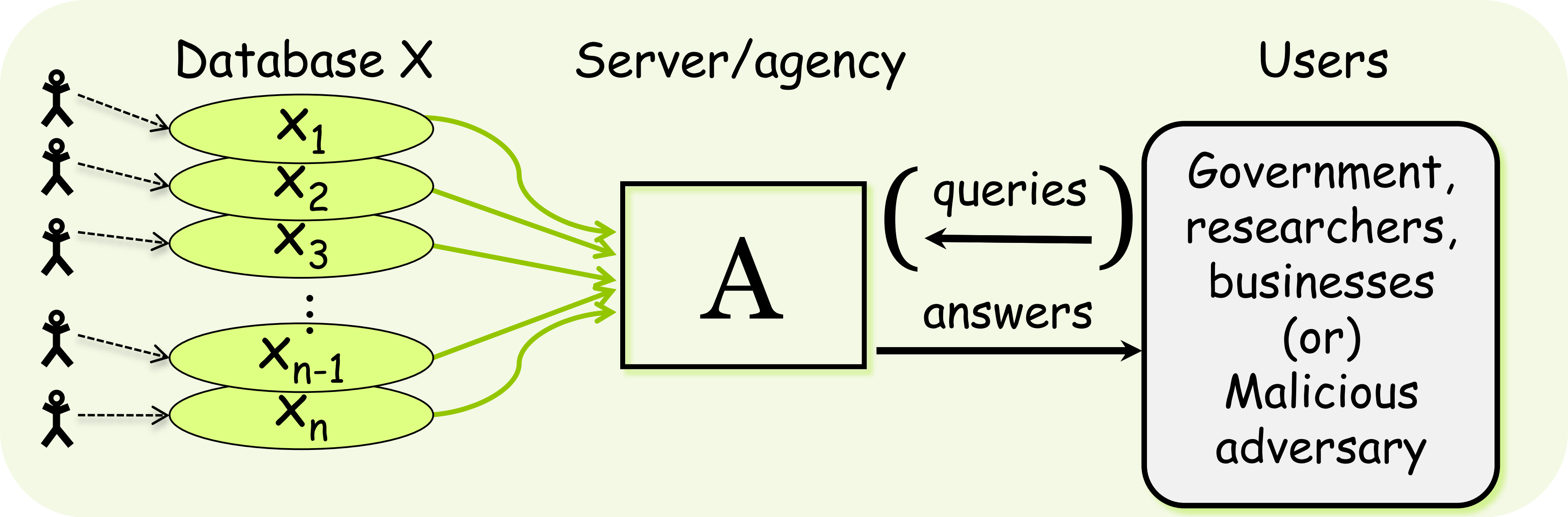
Data Privacy - The Problem

- Given:
  - A dataset with sensitive information
- How to:
  - Compute and release functions of the dataset without compromising individual privacy



Data Privacy – The Problem

- Given:
  - A dataset with sensitive information
- How to:
  - Compute and release functions of the dataset without compromising individual privacy



Data Privacy – The Problem

- Given:
  - A dataset with sensitive information
- How to:
  - Compute and release functions of the dataset without compromising individual privacy

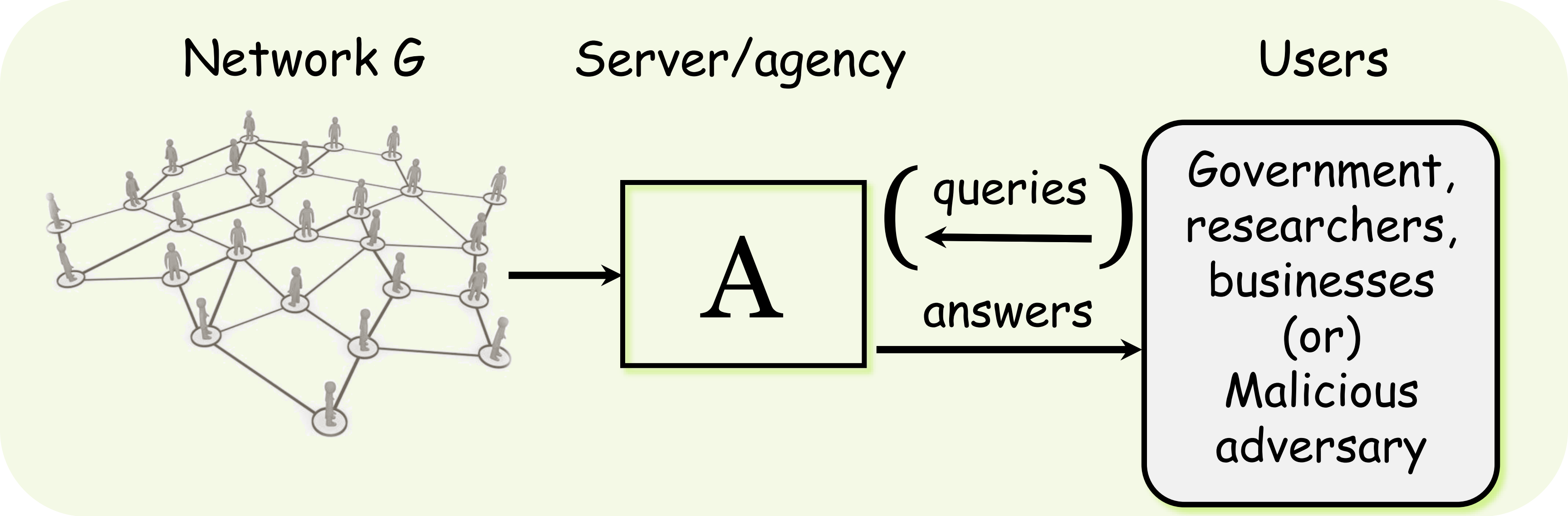


Image credit: www.perey.com

Yes, This Has Been Asked Before

Yes, This Has Been Asked Before

- Traditional approaches:
  - Anonymization, redaction, auditing, noise addition, synthetic data, …
    - Still in use
    - Accumulating litany of attacks and failures

Yes, This Has Been Asked Before

- Traditional approaches:
  - Anonymization, redaction, auditing, noise addition, synthetic data, …
    - Still in use
    - Accumulating litany of attacks and failures
- Lack of rigor leads to unforeseen breaks

Yes, This Has Been Asked Before

- Traditional approaches:
  - Anonymization, redaction, auditing, noise addition, synthetic data, …
    - Still in use
    - Accumulating litany of attacks and failures

- Lack of rigor leads to unforeseen breaks
- Privacy protection is unlike other 'incremental' algorithmic endeavors
  - Information cannot be "de-leaked", breaks are forever

Aggregate Computations and Privacy

- Aren't releases of "global" information safe?
  - Statistics, machine learning, …
  - Don't I "hide in the crowd"?

Aggregate Computations and Privacy

- Aren't releases of "global" information safe?
  - Statistics, machine learning, …
  - Don't I "hide in the crowd"?
- "Global" goal can depend on a few specific values
  - Not uncommon, e.g., Support Vector Machines

Aggregate Computations and Privacy

- Aren't releases of "global" information safe?
  - Statistics, machine learning, …
  - Don't I "hide in the crowd"?
- "Global" goal can depend on a few specific values
  - Not uncommon, e.g., Support Vector Machines
- Composition
  - Compute average salary before/after professor resigns

Aggregate Computations and Privacy

- Aren't releases of "global" information safe?
  - Statistics, machine learning, …
  - Don't I "hide in the crowd"?
- "Global" goal can depend on a few specific values
  - Not uncommon, e.g., Support Vector Machines
- Composition
  - Compute average salary before/after professor resigns
- Statistics may together encode sensitive info
  - Too many, "too accurate" stats $\Rightarrow$ reconstruct the data
  - Robust even to fairly significant noise

Data Privacy – The Problem

[Reformulated for Today's Purposes]

How to compute aggregates …

Data Privacy – The Problem

[Reformulated for Today's Purposes]

How to compute aggregates …

… while controlling the leakage of individual information

This Talk: Intro to Differential Privacy in Analysis of Graphs

- What is differential privacy
  - Differential privacy for graph data – edge/node privacy
- Interpretations of the definition
- Basic properties
- Basic techniques

Differential Privacy

- Changes to my data (almost) unnoticeable in outcome
  - I can claim that my data is different from what it really is (deniability)

Differential Privacy

- Changes to my data (almost) unnoticeable in outcome
  - I can claim that my data is different from what it really is (deniability)
- Omission/inclusion of my data (almost) unnoticeable in outcome
  - As if I chose to opt out

Differential Privacy

- Changes to my data (almost) unnoticeable in outcome
  - I can claim that my data is different from what it really is (deniability)
- Omission/inclusion of my data (almost) unnoticeable in outcome
  - As if I chose to opt out
- My data?
  - Record containing my information in a database

Differential Privacy

- Changes to my data (almost) unnoticeable in outcome
  - I can claim that my data is different from what it really is (deniability)

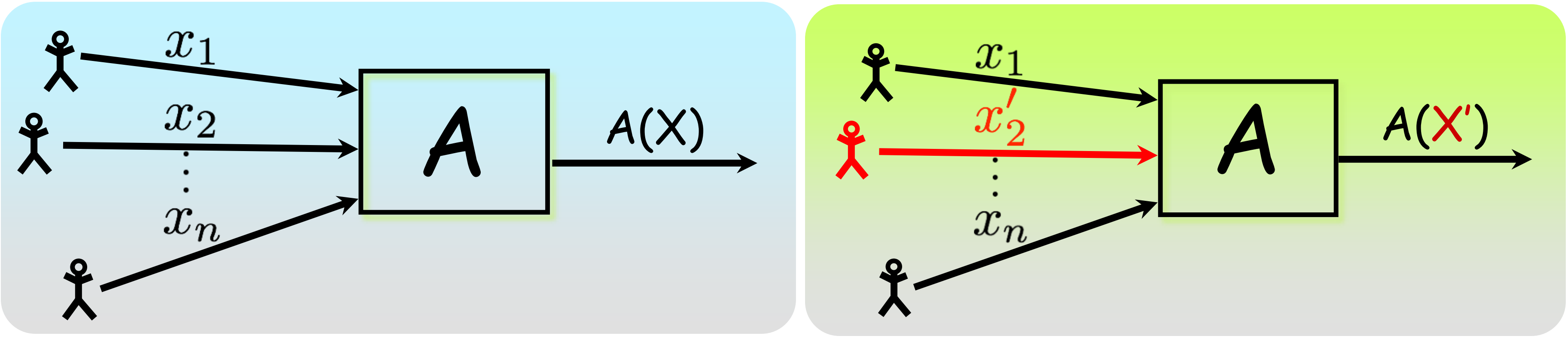
- Omission/inclusion of my data (almost) unnoticeable in outcome
  - As if I chose to opt out

- My data?
  - Record containing my information in a database
  - Graph data: edge/node

Neighboring Inputs

[What Should Be Protected?]

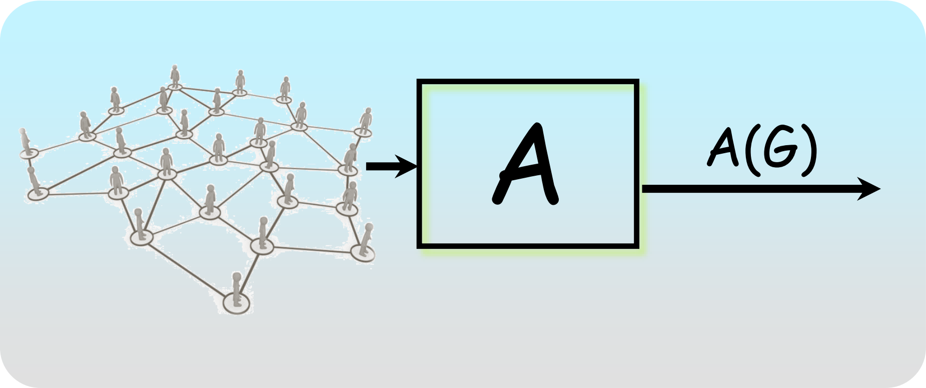
- Inputs are neighboring if they differ on the data of a single individual
  - Record privacy: Databases X, X' neighboring if differ on one record

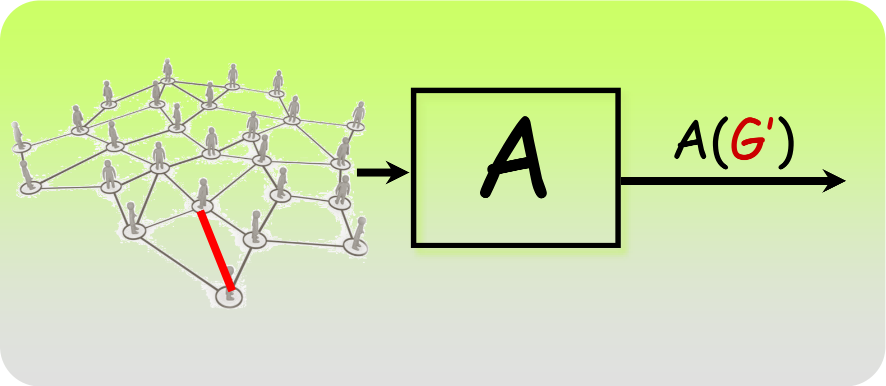


NEIGHBORING INPUTS

[WHAT SHOULD BE PROTECTED?]

- Inputs are neighboring if they differ on the data of a single individual
  - Record privacy: Databases X, X' neighboring if differ on one record
  - Edge privacy: graphs G, G' neighboring if differ on one edge

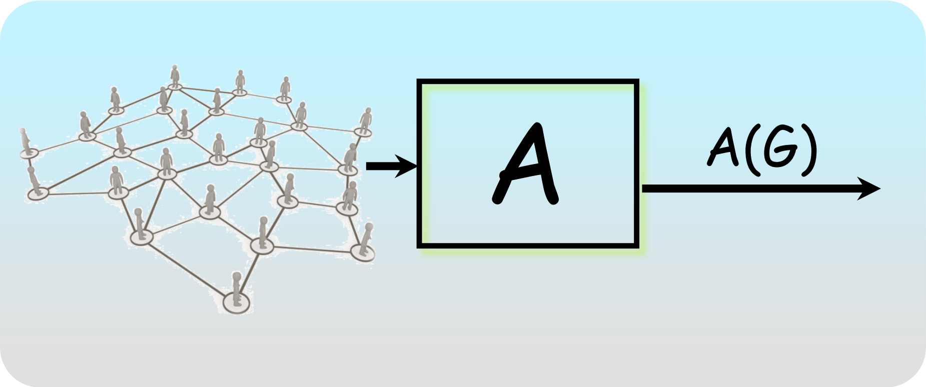




Neighboring Inputs

[What Should Be Protected?]

- Inputs are neighboring if they differ on the data of a single individual
  - Record privacy: Databases X, X′ neighboring if differ on one record
  - Edge privacy: graphs G, G′ neighboring if differ on one edge
  - Node privacy: graphs G, G′ neighboring if differ on one node and its adjacent edges



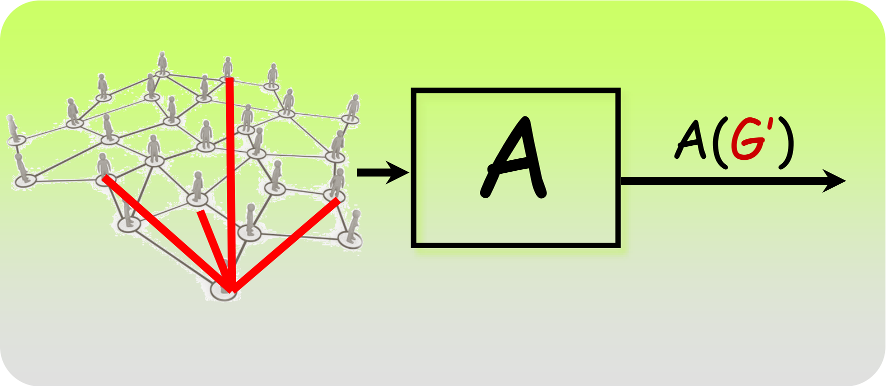
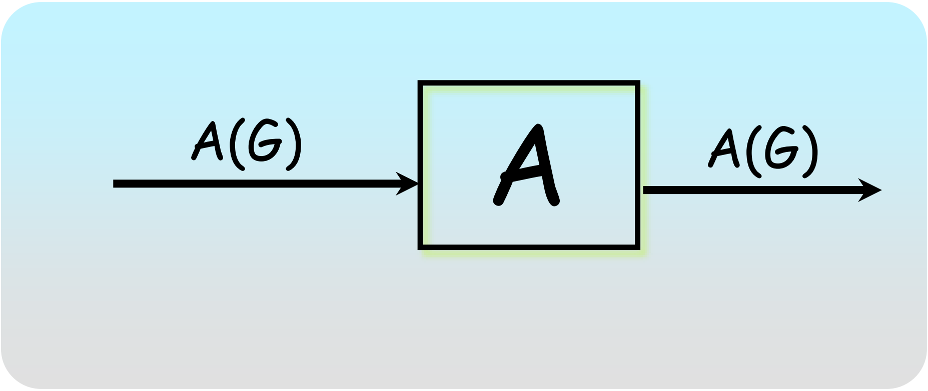
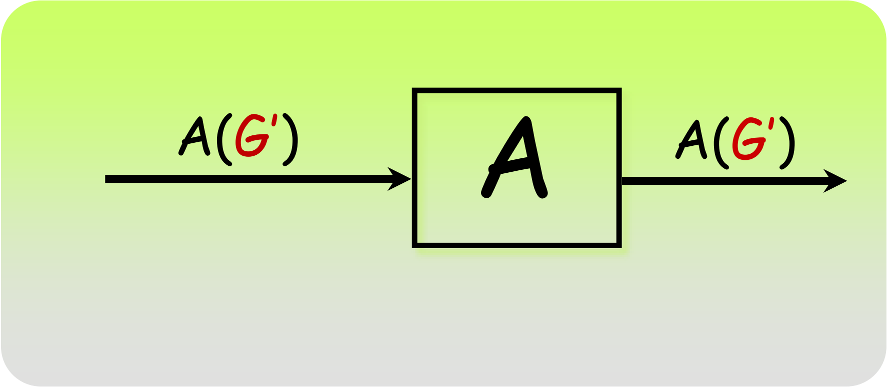


Image credit: www.perey.com

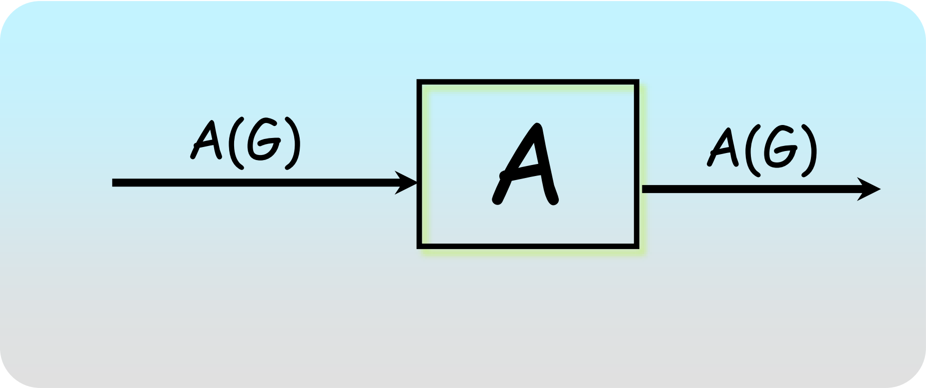
Differential Privacy
[DMNS 06]

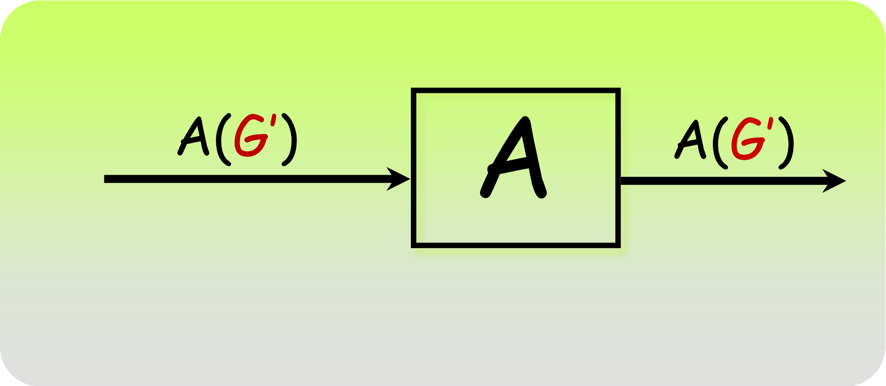




Differential Privacy
[DMNS 06]

- A is differentially private if
  - for all neighboring G, G'
  - given A's outcome, privacy attacker cannot guess whether input was G or G'

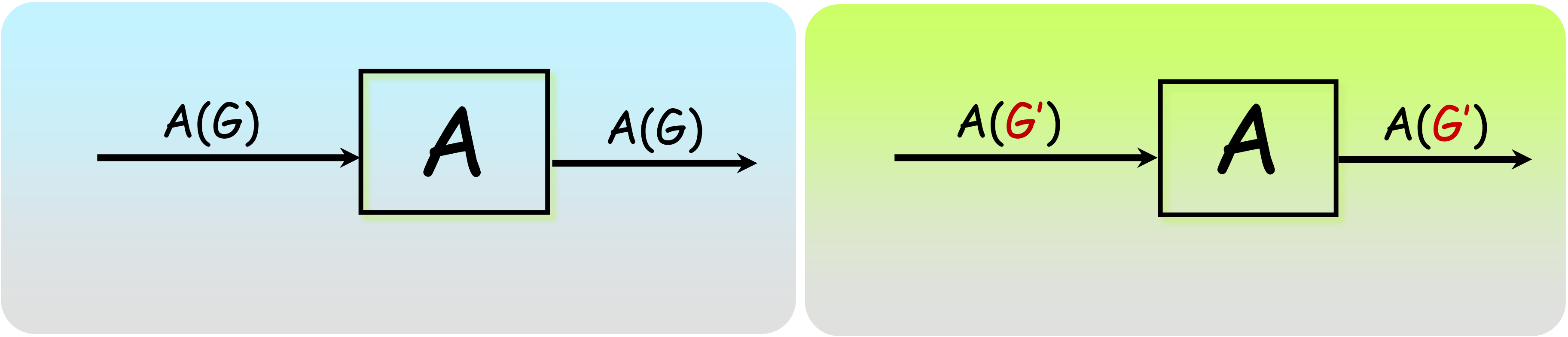




Differential Privacy
[DMNS 06]

- A is differentially private if
  - for all neighboring G, G'
  - for all subsets S of outputs

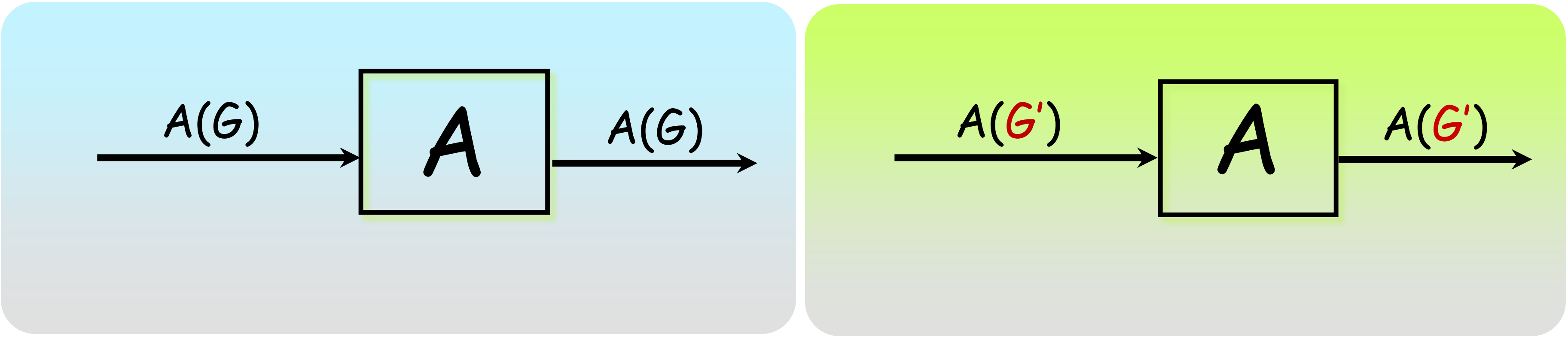
$$\Pr[A(G) \in S] \approx \Pr[A(G') \in S]$$



Differential Privacy
[DMNS 06]

- A is ε-differentially private if
  - for all neighboring G, G'
  - for all subsets S of outputs

$$\Pr[A(G) \in S] \leq e^{\epsilon} \cdot \Pr[A(G') \in S]$$



Differential Privacy
[DMNS 06]

- A is ε-differentially private if
  - for all neighboring G, G'
  - for all subsets S of outputs

$$\Pr[A(G) \in S] \leq e^{\epsilon} \cdot \Pr[A(G') \in S]$$

Notes:

- DP is a property of the algorithm A
  - No sense in saying that a particular output preserves privacy - relationship between input and output is what matters

Differential Privacy
[DMNS 06]

- A is ε-differentially private if
  - for all neighboring G, G'
  - for all subsets S of outputs

$$\Pr[A(G) \in S] \leq e^{\epsilon} \cdot \Pr[A(G') \in S]$$

Notes:

- DP is a property of the algorithm A
  - No sense in saying that a particular output preserves privacy - relationship between input and output is what matters
- The parameter ϵ measures 'leakage' or 'harm' (more later).
  - Not negligible. Think $\epsilon \approx \frac{1}{100}$ or $\epsilon \approx \frac{1}{10}$ not $\epsilon \approx 2^{-80}$
-

Differential Privacy
[DMNS 06]

- A is **ε-differentially private** if
  - for all neighboring G, G'
  - for all subsets S of outputs

$$\Pr[A(G) \in S] \leq e^{\epsilon} \cdot \Pr[A(G') \in S]$$

Notes:

- DP is a property of the algorithm A
  - No sense in saying that a particular output preserves privacy - relationship between input and output is what matters
- The parameter ϵ measures 'leakage' or 'harm' (more later).
  - Not negligible. Think $\epsilon \approx \frac{1}{100}$ or $\epsilon \approx \frac{1}{10}$ not $\epsilon \approx 2^{-80}$
- Choice of distance measure (max log ratio) not accidental

Basic Properties Of Differential Privacy

- Post processing:
  - If A is ε-dp then $B \circ A$ is ε-dp for all B

Basic Properties Of Differential Privacy

- Post processing:
  - If A is ε-dp then $B \circ A$ is ε-dp for all B
- Composition:
  - A_1, A_2: ε-dp then (A_1, A_2) is 2ε-dp.
    - More efficient composition theorems exist w.r.t. a relaxation of differential privacy

Basic Properties Of Differential Privacy

- Post processing:
  - If A is ε-dp then $B \circ A$ is ε-dp for all B

- Composition:
  - A_1, A_2: ε-dp then (A_1, A_2) is 2ε-dp.
    - More efficient composition theorems exist w.r.t. a relaxation of differential privacy
    - t executions of ϵ-dp private mechanisms are $\approx \sqrt{t}\epsilon$-dp

Interpreting Differential Privacy

- A naïve hope: Your beliefs about me are the same after you see the output as they were before.

Interpreting Differential Privacy

- ~~A naïve hope: Your beliefs about me are the same after you see the output as they were before.~~

- Suppose I smoke in public
  - A public health study could teach that I am at risk for cancer.
  - But it didn't matter whether or not my data was part of it.

Interpreting Differential Privacy

- ~~A naïve hope: Your beliefs about me are the same after you see the output as they were before.~~

- Suppose I smoke in public
  - A public health study could teach that I am at risk for cancer.
  - But it didn't matter whether or not my data was part of it.

- **Theorem** [Dwork Naor 06]: Learning things about individuals is unavoidable in the presence of arbitrary external information.

Interpreting Differential Privacy

- Compare $x \;\; = (x_1, x_2, \ldots, x_i, \ldots, x_n)$
  to $x_{-i} = (x_1, x_2, \ldots, \perp, \ldots, x_n)$

- A is ε-differentially private if for all vectors x and for all i: $A(x) \approx_\varepsilon A(x_{-i})$.

Interpreting Differential Privacy

- Compare $x \;\; = \; (x_1, x_2, \ldots, x_i, \ldots, x_n)$
  to $x_{-i} = \; (x_1, x_2, \ldots, \perp, \ldots, x_n)$

- A is ε-differentially private if for all vectors x and for all i: $A(x) \; \approx_{\varepsilon} \; A(x_{-i})$.

- No matter what you know ahead of time, you learn (almost) the same things about me whether or not my data are used.

Interpreting Differential Privacy

- Compare $x \quad = \; (x_1, x_2, \dots, x_i, \dots, x_n)$
  to $x_{-i} = \; (x_1, x_2, \dots, \perp, \dots, x_n)$

- A is ε-differentially private if for all vectors x and for all i: $A(x) \; \approx_\varepsilon \; A(x_{-i})$.

= =

- For any non-negative function p of the outcome,

$$E[p(A(x)] \leq e^{\epsilon} \cdot E[p(A(x')]$$

Interpreting Differential Privacy

- Compare $x \;\; = (x_1, x_2, \ldots, x_i, \ldots, x_n)$
  to $x_{-i} = (x_1, x_2, \ldots, \perp, \ldots, x_n)$

- A is ε-differentially private if for all vectors x and for all i: $A(x) \approx_\varepsilon A(x_{-i})$.

- For any non-negative function p of the outcome,

$$E[p(A(x)] \leq e^{\epsilon} \cdot E[p(A(x')]$$

  - Let p = my insurance premium
  - My expected premium almost does not change whether I participate in A or not!

Things to Note About Differential Privacy

- May not protect sensitive global information, *e.g.*
  - Clinical data: Smoking and cancer
  - Financial transactions: firm-level trading strategies
  - Genomic data: information about me may be revealed if enough of my family members participate
  - Social data: what if my presence affects everyone else?

Things to Note About Differential Privacy

- May not protect sensitive global information, *e.g.*
  - Clinical data: Smoking and cancer
  - Financial transactions: firm-level trading strategies
  - Genomic data: information about me may be revealed if enough of my family members participate
  - Social data: what if my presence affects everyone else?
  - Bug of feature?

Things to Note About Differential Privacy

- May not protect sensitive global information, *e.g.*
  - Clinical data: Smoking and cancer
  - Financial transactions: firm-level trading strategies
  - Genomic data: information about me may be revealed if enough of my family members participate
  - Social data: what if my presence affects everyone else?
  - Bug of feature?

- Leakage accumulates with composition
  - ε adds up with many releases
    - Very unlike what is usual in crypto
    - Inevitable in some form (reconstruction attacks)

Things to Note About Differential Privacy

- May not protect sensitive global information, *e.g.*
  - Clinical data: Smoking and cancer
  - Financial transactions: firm-level trading strategies
  - Genomic data: information about me may be revealed if enough of my family members participate
  - Social data: what if my presence affects everyone else?
  - Bug of feature?
- Leakage accumulates with composition
  - ε adds up with many releases
    - Very unlike what is usual in crypto
    - Inevitable in some form (reconstruction attacks)
  - How to set ε?

Variations on Differential Privacy

- Predecessors [DDN'03,EGS'03,DN'04,BDMN'05]
- (ε,δ)- differential privacy [DKMMN'05]
  - Require $\Pr[A(x) \in S] \leq e^{\epsilon} \Pr[A(x') \in S] + \delta$
  - Similar semantics to (ε,0)- differential privacy when $\delta \ll 1/\text{poly}(n)$
  - Allows for improved utility
- Computational variants [MPRV09,MMPRTV'10].
- Distributional variants [RHMS'09,BBGLT'11,BGKS'13].
  - Assume something about adversary's prior distribution.
  - Deterministic releases.
  - Poor composition guarantees.
- Generalizations.
  - [BLR'08, GLP'11] simulation-based definitions.
  - [KM'12, BGKS'13] General language for specifying privacy concerns, tricky to instantiate.
- Crowd-blending privacy [GHLP'12].

Example: Counting Edges

[THE BASIC TECHNIQUE]

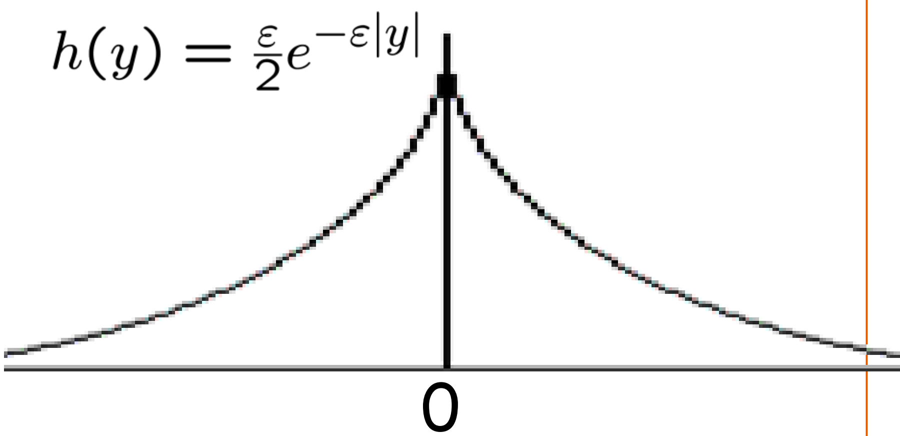
- $f(G) = \Sigma e_{ij}$ where $e_{ij} \in \{0,1\}$

Example: Counting Edges

[The basic technique]

- $f(G) = \Sigma e_{ij}$ where $e_{ij} \in \{0,1\}$
- Algorithm: On input G return $f(G) + Y$, where $Y \sim Lap(\frac{1}{\epsilon})$

- Laplace Distribution:
  - $E[Y] = 0; \sigma[Y] = \sqrt{2}/\varepsilon$



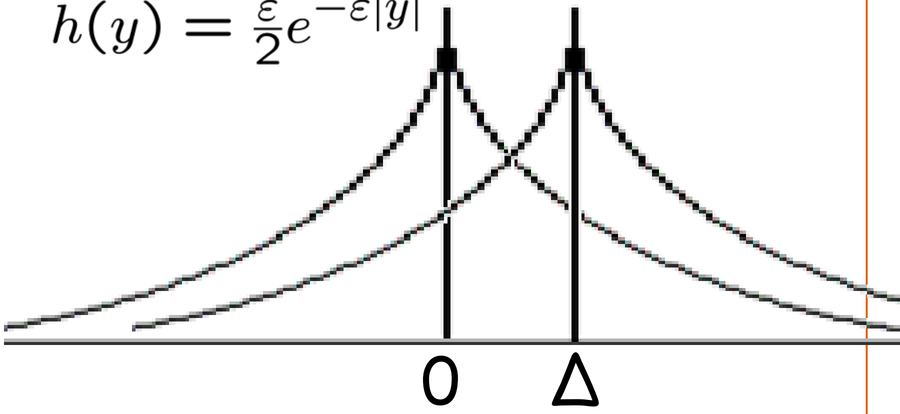
Example: Counting Edges

[THE BASIC TECHNIQUE]

- $f(G) = \Sigma e_{ij}$ where $e_{ij} \in \{0,1\}$
- Algorithm: On input G return $f(G) + Y$, where $Y \sim Lap(\frac{1}{\epsilon})$

- Laplace Distribution:
  - $E[Y] = 0; \sigma[Y] = \sqrt{2}/\varepsilon$
  - Sliding property: $\frac{h(y)}{h(y+1)} \leq e^{\epsilon}$

$h(y) = \frac{\varepsilon}{2} e^{-\varepsilon|y|}$

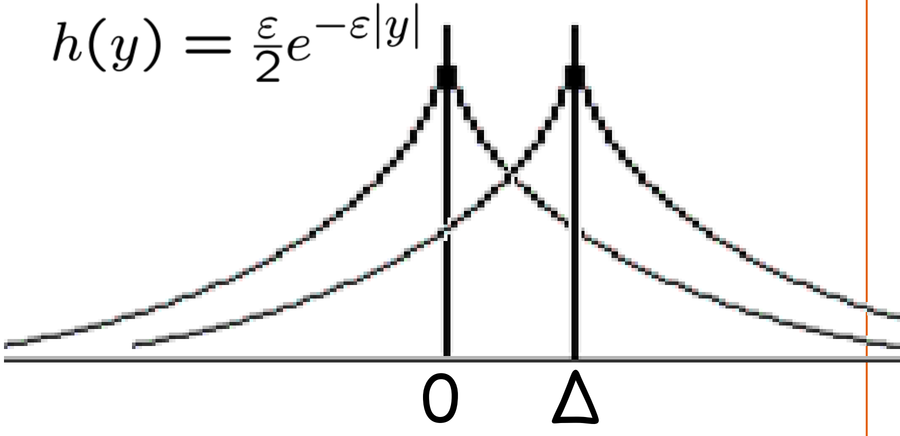


Example: Counting Edges

[THE BASIC TECHNIQUE]

- $f(G) = \Sigma e_{ij}$ where $e_{ij} \in \{0,1\}$
- Algorithm: On input G return $f(G) + Y$, where $Y \sim Lap(\frac{1}{\epsilon})$

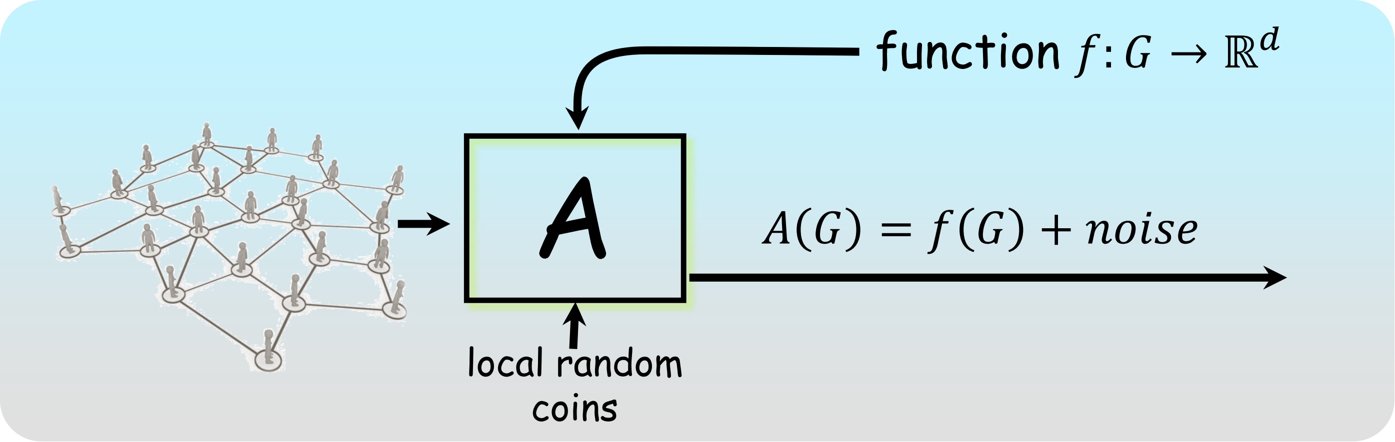
- Laplace Distribution:
  - $E[Y] = 0; \sigma[Y] = \sqrt{2}/\varepsilon$
  - Sliding property: $\frac{h(y)}{h(y+1)} \leq e^{\epsilon}$



- For G, G' edge neighboring:

$$|f(G) - f(G')| = \left|\sum_{ij} e_{ij} - \sum_{ij} e'_{ij}\right| \leq 1$$

Framework of Global Sensitivity [DMNS06]



- $GS_f = \max |f(G) - f(G')|_1$ taken over neighboring G, G'
- Theorem [DMNS06]:
  - $A(G) = f(G) + Lap^d(\frac{GS_f}{\epsilon})$ is ε-differentially private

FRAMEWORK OF GLOBAL SENSITIVITY

$GS_f = \max |f(G) - f(G')|_1$ taken over neighboring G, G'

$$A(G) = f(G) + Lap^d(\frac{GS_f}{\epsilon})$$

- Many natural functions have low global sensitivity
  - e.g., histogram, mean, covariance matrix, distance to a function, estimators with bounded "sensitivity curve", strongly convex optimization problems.

Framework of Global Sensitivity

$GS_f = \max |f(G) - f(G')|_1$ taken over neighboring G, G'

$$A(G) = f(G) + Lap^d(\frac{GS_f}{\epsilon})$$

- Many natural functions have low global sensitivity
  - e.g., histogram, mean, covariance matrix, distance to a function, estimators with bounded "sensitivity curve", strongly convex optimization problems.
- Laplace mechanism can be a programming interface [BDMN '05].
  - Implemented in several systems [McSherry '09, Roy et al. '10, Haeberlen et al. '11, Moharan et al. '12].

Edge vs. Node Privacy - Counting Edges

$GS_f = \max |f(G) - f(G')|_1$ taken over neighboring G, G'

$$A(G) = f(G) + Lap^d(\frac{GS_f}{\epsilon})$$

- Counting edges: $f(G) = \Sigma e_{ij}$ where $e_{ij} \in \{0,1\}$
- Edge privacy: $GS_f = 1$, noise $\sim\frac{1}{\epsilon}$
- Node privacy: $GS_f = n$, noise $\sim\frac{n}{\epsilon}$

Edge vs. Node Privacy - Counting Edges

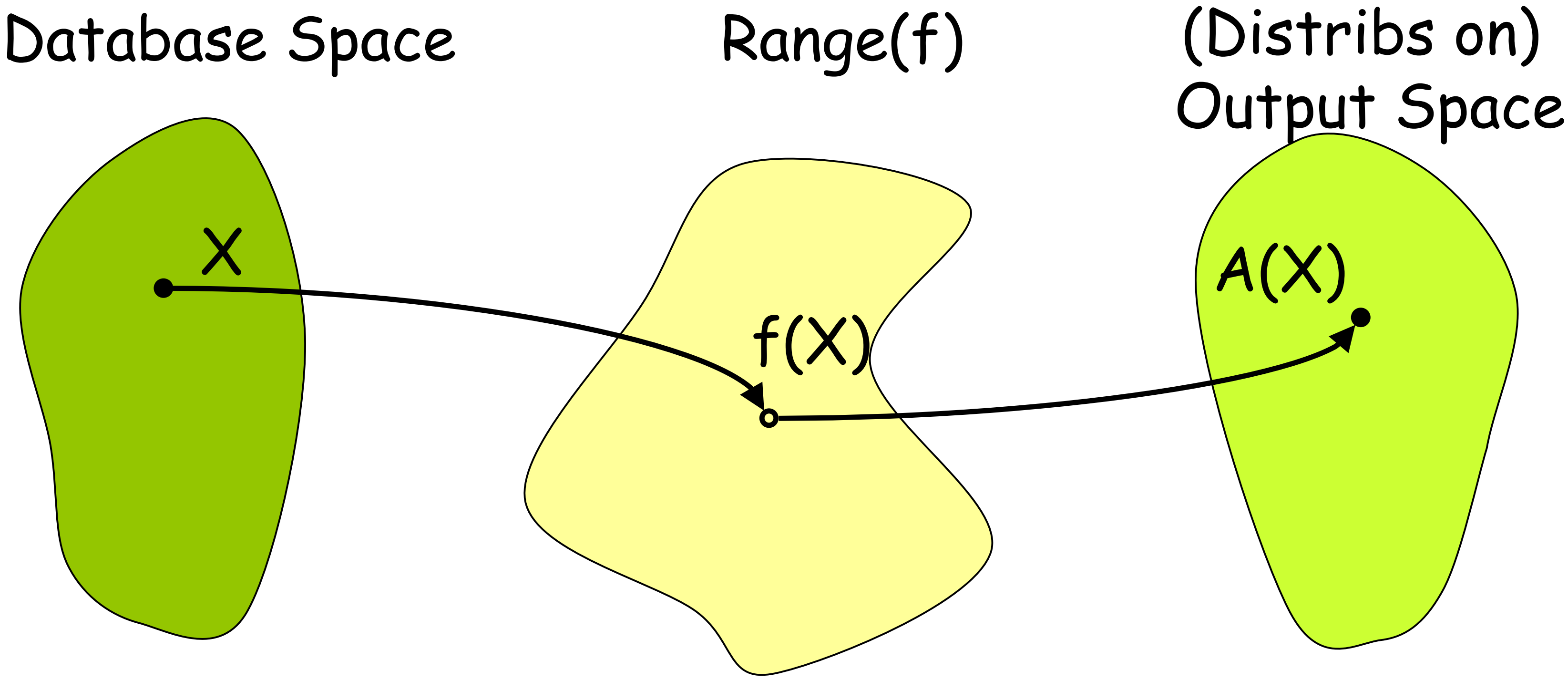
$GS_f = \max |f(G) - f(G')|_1$ taken over neighboring G, G'

$$A(G) = f(G) + Lap^d(\frac{GS_f}{\epsilon})$$

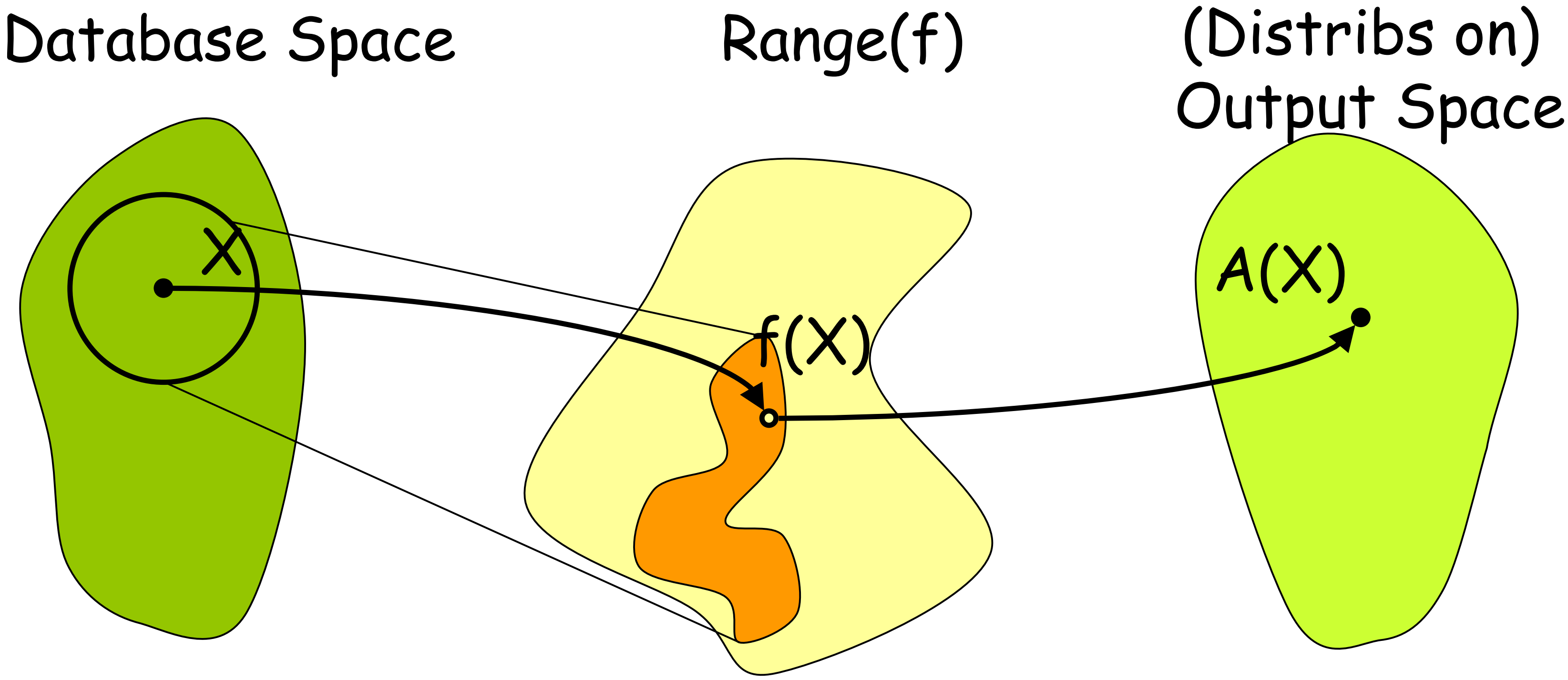
- Counting edges: $f(G) = \Sigma e_{ij}$ where $e_{ij} \in \{0,1\}$
- Edge privacy: $GS_f = 1$, noise $\sim\frac{1}{\epsilon}$
- Node privacy: $GS_f =$ n, noise $\sim\frac{n}{\epsilon}$

- Degree distribution??

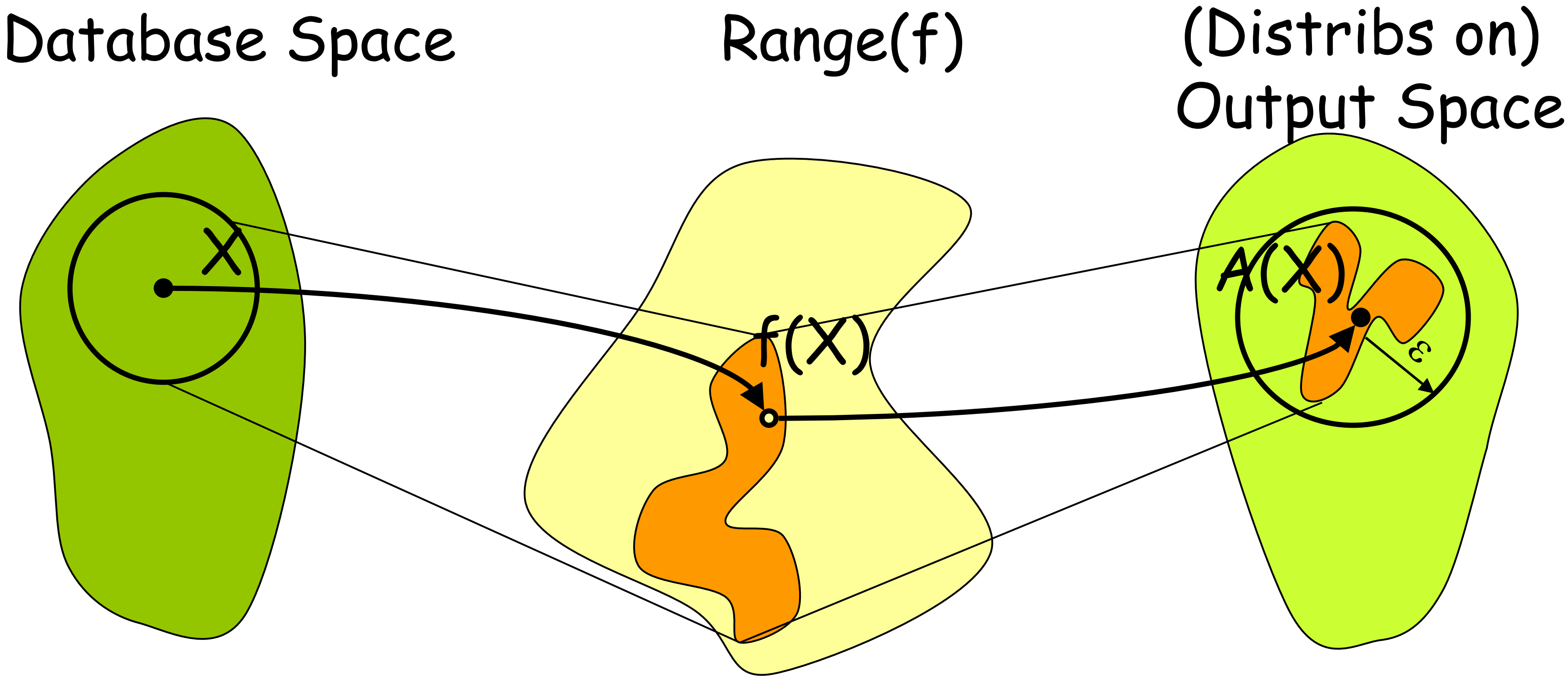
Global vs. Local Sensitivity



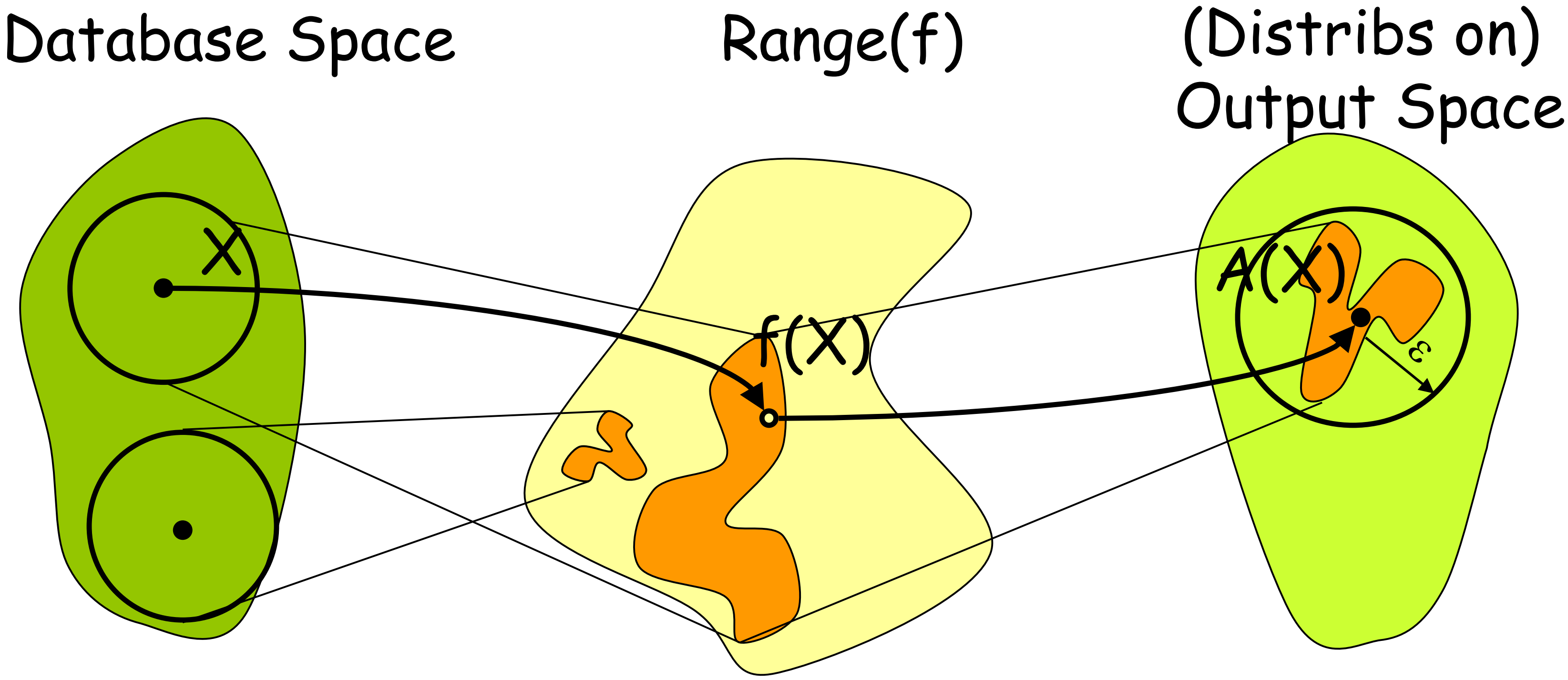
Global vs. Local Sensitivity



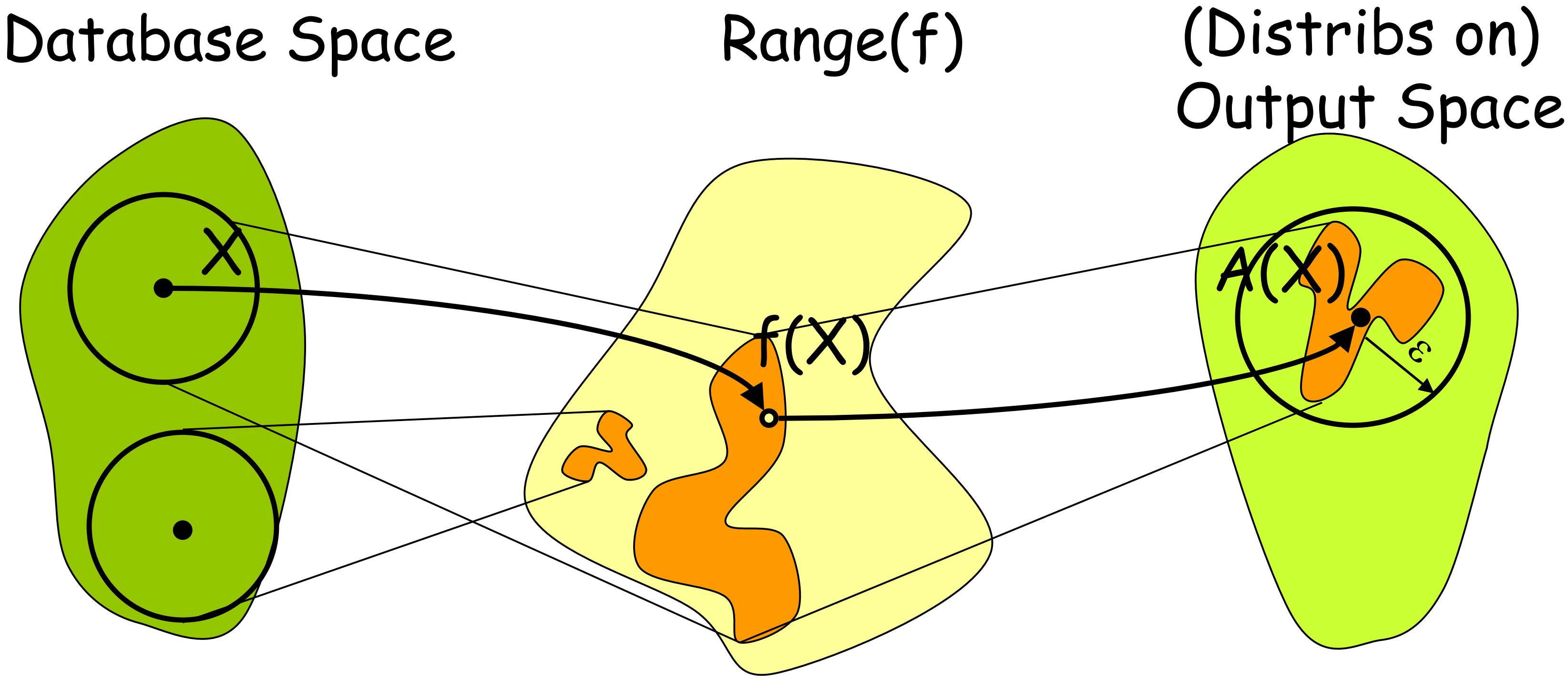
Global vs. Local Sensitivity



Global vs. Local Sensitivity

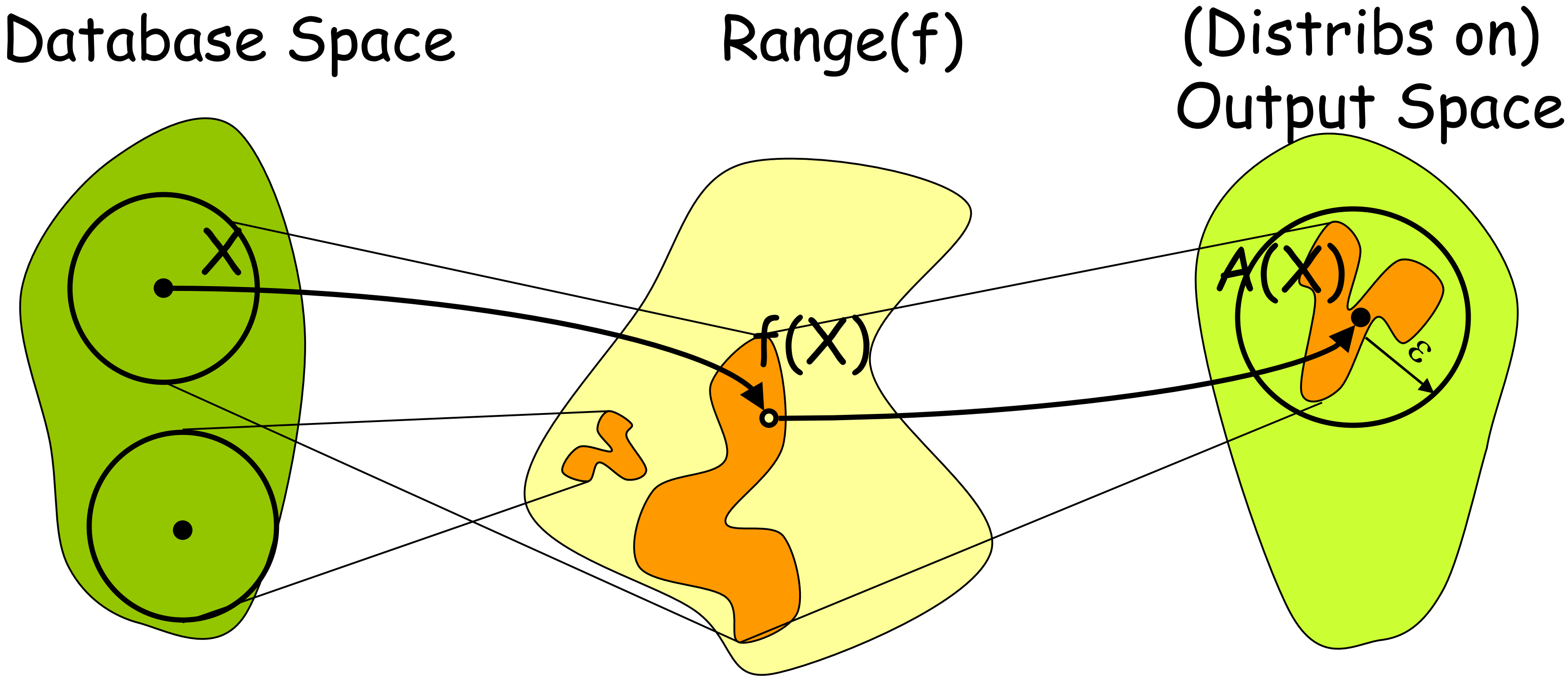


Global vs. Local Sensitivity



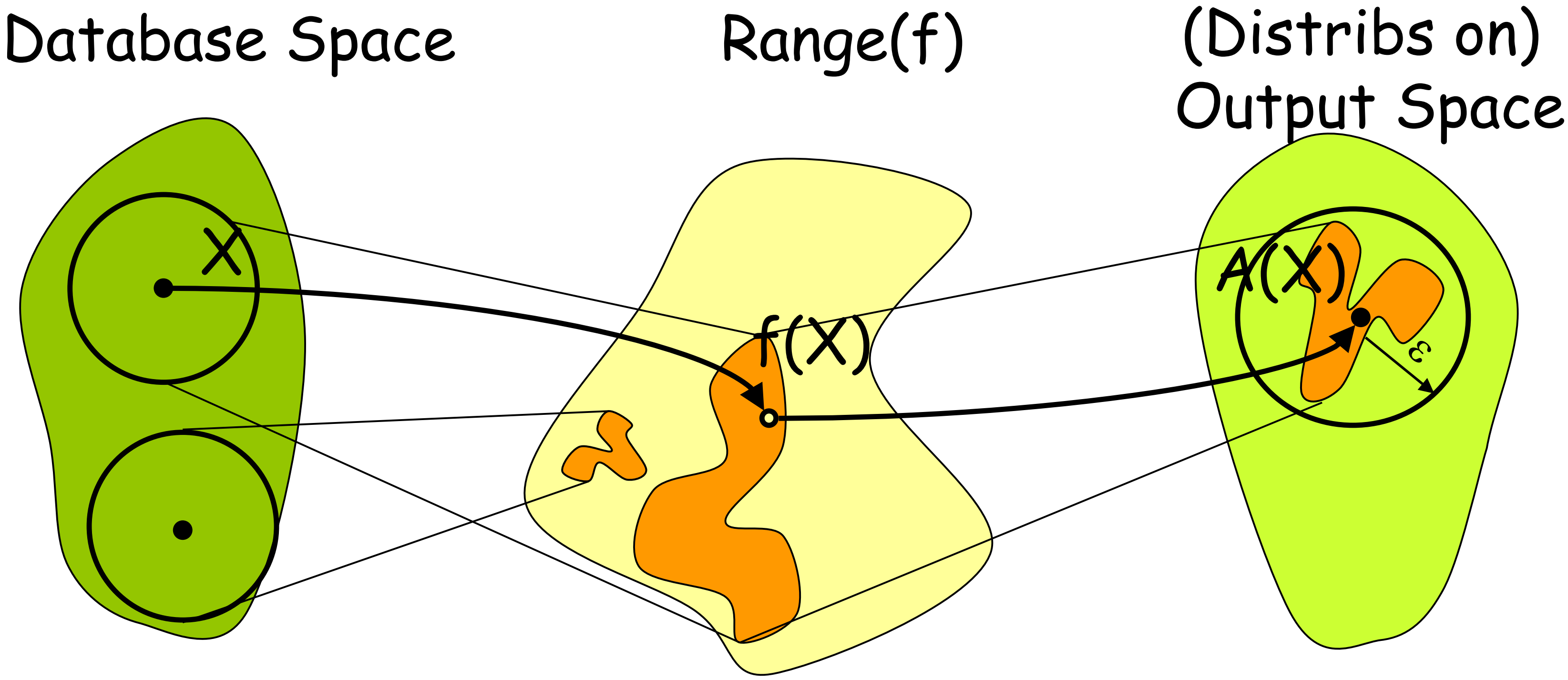
- $LS_f(X) = \max_{X' neighbor\ of\ X} |f(X) - f(X')|_1$

GLOBAL VS. LOCAL SENSITIVITY



- $LS_f(X) = \max_{X' neighbor\ of\ X} |f(X) - f(X')|_1$
  - $GS_f = \max_X LS_f(X)$

Global vs. Local Sensitivity



- $LS_f(X) = \max_{X' \text{neighbor of } X} |f(X) - f(X')|_1$
  - $GS_f = \max_X LS_f(X)$
- [NRS'07,DL'09] Techniques with error ≈ local sensitivity

Exponential Sampling [MT07]

- x_i= {books read by i this year}, Y = {book names}
- "Score" of $y \in Y$: $q(y, x) = \#\{i : y \in x_i\}$
- Goal: output book read by most

Exponential Sampling [MT07]

- x_i= {books read by i this year}, Y = {book names}
- "Score" of $y \in Y$: $q(y, x) = \#\{i: y \in x_i\}$
- Goal: output book read by most

- Mechanism: given x, output book name y with probability prop to $\exp(\frac{\epsilon}{2} \cdot q(y, x))$

Exponential Sampling [MT07]

- x_i= {books read by i this year}, Y = {book names}
- "Score" of $y \in Y$: $q(y, x) = \#\{i: y \in x_i\}$
- Goal: output book read by most

- Mechanism: given x, output book name y with probability prop to $\exp(\frac{\epsilon}{2} \cdot q(y, x))$

- Claim: Mechanism is ε-differentially private

Exponential Sampling [MT07]

- x_i= {books read by i this year}, Y = {book names}
- "Score" of $y \in Y$: $q(y, x) = \#\{i: y \in x_i\}$
- Goal: output book read by most

- Mechanism: given x, output book name y with probability prop to $\exp(\frac{\epsilon}{2} \cdot q(y, x))$

- Claim: Mechanism is ε-differentially private

- Claim: If most popular website has score $T = \max_{y \in Y} q(y, x)$, then $E[q(y_0, x)] \geq t - O(\frac{\log|Y|}{\epsilon})$

Applications of Exponential Sampling

- Very general and widely used
  - Often a 'first attempt' at a differentially private task.
- Used explicitly for
  - Learning discrete classifiers, Synthetic data generation, Convex Optimization, Genome-wide association studies, High-dimensional sparse regression, …
- But, generally inefficient [DNRRV,…]

Differential Privacy in "Practice"

- Currently, differential private algorithms hard to use.
  - Noise.
  - No off-the-shelf software.
  - Each application requires fresh thinking.
- Several systems to make use easier.
  - [McSherry'09] PINQ: variation on LINQ with differential privacy enforced by query mechanism.
  - [Haeberlen et al. '11] Programming language with privacy enforced by type system.
  - [Roy et al. '10, Moharan et al. '12] Systems for restricted classes of queries, focus on usability with legacy code.
- Hard to get right!
  - [Haeberlen et al. '11] Timing attacks.
  - [Mironov '12] Leakage via numerical errors.

Settings where Differential Privacy was Applied [partial list]

- Machine learning
- Statistics
- Continual observation and pan privacy
  - When input is supplied gradually
  - When the state of the algorithm can be subpoenaed
- Distributed settings
  - Surprising relationships with computational differential privacy
- Mechanism design
- Privacy for the analyzer
- Graph data

Conclusions

- Heuristic treatment of privacy leads to failures
  - Weaknesses: Auxiliary information, (self) composition, leakage in decisions, …
- Differential Privacy: privacy defined in terms of my effect on output
  - Meaningful despite arbitrary external information.
  - I should participate if I get benefit.
- Computations with rigorous privacy guarantees.
  - Basic Tools.
  - More advanced examples.
- Connections to many areas: Security and crypto, Machine learning, Statistics, Economics.