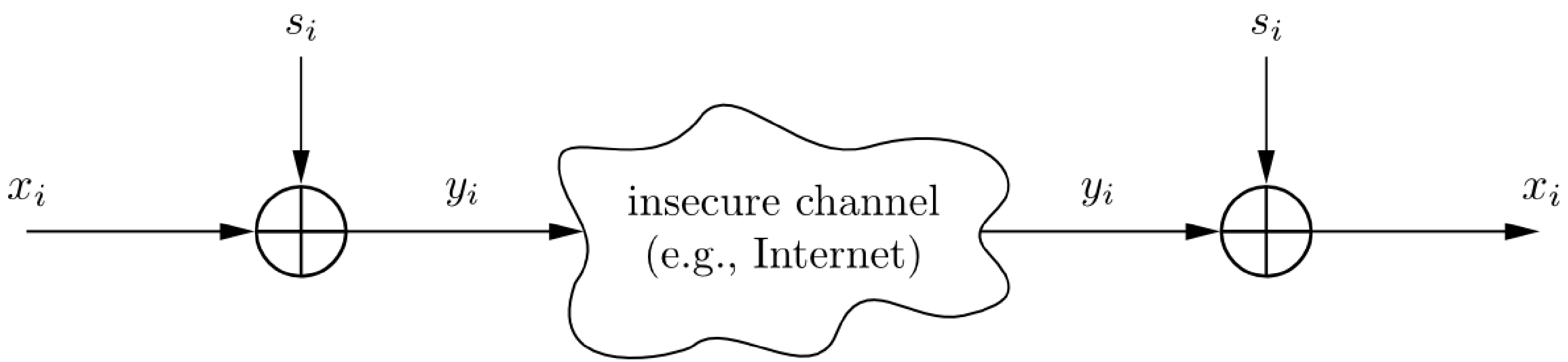
Privacy Final Review

Encryption & Decryption

Plaintext x_i, ciphertext y_i and key stream s_i consist of individual bits



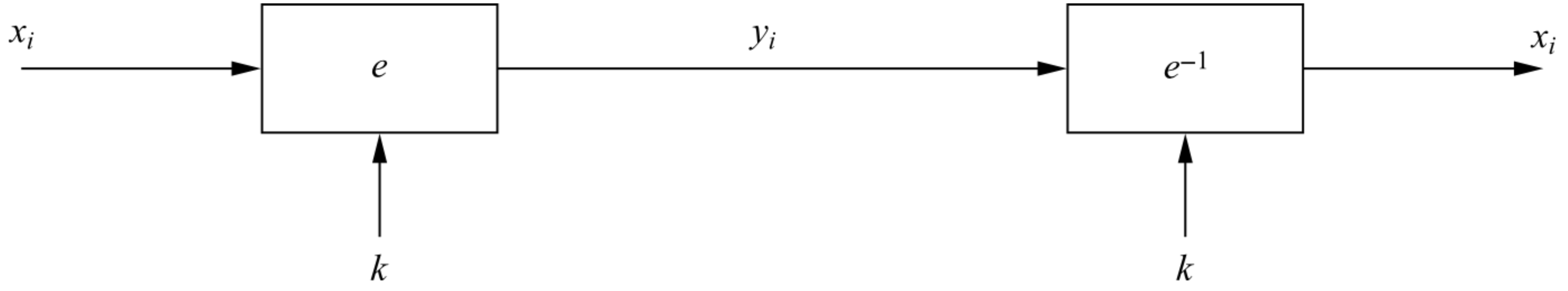
- Encryption and decryption are simple additions modulo 2 (aka XOR)
- Encryption and decryption are the same functions

Encryption: $y_i = e_{s_i}(x_i) = x_i + s_i \bmod 2, \; x_i, y_i, s_i \in \{0,1\}$

Decryption: $x_i = e_{s_i}(y_i) = y_i + s_i \bmod 2$

Electronic Code Book Mode (ECB)

- $e_k(x_i)$: the encryption of a b-bit plaintext block x_i with key k
- $e_k^{-1}(y_i)$: the decryption of b-bit ciphertext block y_i with key k
- Messages which exceed b bits are partitioned into b-bit blocks
- **Each Block is encrypted separately**



Encryption: $y_i = e_k(x_i),\ i \geq 1$

Decryption: $x_i = e_k^{-1}(y_i) = e_k^{-1}(e_k(x_i)),\ i \geq 1$

Extended Euclidean Algorithm (1)

- Extend the Euclidean algorithm to **find modular inverse** of r_1 mod r_0
- EEA computes *s*, *t*, and the gcd :

$$\gcd(r_0, r_1) = s \cdot r_0 + t \cdot r_1$$

- Take the relation **mod r_0**

$$\boxed{\begin{aligned} s \cdot r_0 + t \cdot r_1 &= 1 \\ s \cdot 0 + t \cdot r_1 &\equiv 1 \bmod r_0 \\ r_1 \cdot t &\equiv 1 \bmod r_0 \end{aligned}}$$

→ Compare with the definition of modular inverse: ***t* is the inverse of r_1 mod r_0**

- Note that $gcd\,(r_0, r_1) = 1$ in order for the inverse to exist

Extended Euclidean Algorithm (2)

Extended Euclidean Algorithm (EEA)
Input: positive integers r_0 and r_1 with $r_0 > r_1$
Output: $\gcd(r_0, r_1)$, as well as s and t such that $\gcd(r_0, r_1) = s \cdot r_0 + t \cdot r_1$.
Initialization:

$$
\begin{aligned}
s_0 &= 1 \qquad t_0 = 0 \\
s_1 &= 0 \qquad t_1 = 1 \\
i &= 1
\end{aligned}
$$

Algorithm:

1 DO
1.1 $i = i + 1$
1.2 $r_i = r_{i-2} \bmod r_{i-1}$
1.3 $q_{i-1} = (r_{i-2} - r_i)/r_{i-1}$
1.4 $s_i = s_{i-2} - q_{i-1} \cdot s_{i-1}$
1.5 $t_i = t_{i-2} - q_{i-1} \cdot t_{i-1}$
WHILE $r_i \neq 0$
2 RETURN
$\gcd(r_0, r_1) = r_{i-1}$
$s = s_{i-1}$
$t = t_{i-1}$

Remark of WHILE loop:

$\gcd(r_0, r_1) = \gcd(r_0 \bmod r_1, r_1)$

$\rightarrow r_2 = r_0 \bmod r_1,\ r_0 = q_1 r_1 + r_2$

$\rightarrow r_{i-2} = q_{i-1} r_{i-1} + r_i$

$\rightarrow r_i = r_{i-2} - q_{i-1} r_{i-1} = [s_i] r_0 + [t_i] r_1$

Example: EEA

i	q_{i-1}	r_i	s_i	t_i
2	5	7	1	-5
3	1	5	-1	6
4	1	2	2	-11
5	2	1	-5	**28**

- Calculate the modular Inverse of 12 mod 67:
- From magic table follows
- Hence **28 is the inverse** of 12 mod 67.

✔

- Check: 28 ・ 12 = 336 ≡ 1 mod 67

Euler's Phi Function (1)

- *New problem, important for public-key systems, e.g., RSA:*
  Given the set of the *m* integers {0, 1, 2, …, *m* -1},
  How many numbers in the set are **relatively prime to *m*** ?
- Answer: **Euler's Phi function *Φ(m)***
- **Example** for the sets {0,1,2,3,4,5} (*m*=6) and {0,1,2,3,4} (*m*=5)

$$\begin{aligned}\gcd(0,6) &= 6\\ \gcd(1,6) &= 1 \Leftarrow\\ \gcd(2,6) &= 2\\ \gcd(3,6) &= 3\\ \gcd(4,6) &= 2\\ \gcd(5,6) &= 1 \Leftarrow\end{aligned}$$

$$\begin{aligned}\gcd(0,5) &= 5\\ \gcd(1,5) &= 1 \Leftarrow\\ \gcd(2,5) &= 1 \Leftarrow\\ \gcd(3,5) &= 1 \Leftarrow\\ \gcd(4,5) &= 1 \Leftarrow\end{aligned}$$

→ 1 and 5 relatively prime to *m*=6,
hence ***Φ*(6) = 2**

→ ***Φ*(5) = 4**

- Testing one gcd per number in the set is **extremely slow for large *m*.**

Euler‘s Phi Function (2)

- **If** canonical factorization of *m* known**:** $$m = p_1^{e_1} \cdot p_2^{e_2} \cdot \ldots \cdot p_n^{e_n}$$
  (where p_i primes and e_i positive integers)
- **then** calculate Phi according to the relation: $$\Phi(m) = \prod_{i=1}^{n}(p_i^{e_i} - p_i^{e_i-1})$$

- Phi especially easy for $e_i = 1$, e.g., $m = p \cdot q$ → $\Phi(m) = (p-1) \cdot (q-1)$
- **Example** $m = 899 = 29 \cdot 31$:
  Φ(899) = (29-1) · (31-1) = 28 · 30 **= 840**
- **Note:** Finding $\Phi(m)$ is computationally easy **if factorization of *m* is known**
  (otherwise the calculation of $\Phi(m)$ becomes computationally infeasible for large numbers)

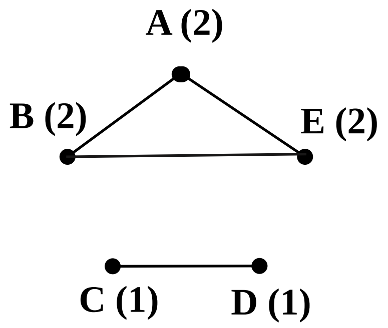
k-degree Anonymity

- Assume that adversary **A** knows that **B** has *327* connections in a social network! (background knowledge)

- If the graph is released by removing the identity of the nodes
  - **A** can find all nodes that have degree *327*
  - If there is only one node with degree *327*, **A** can identify this node as being **B**.

k-degree Anonymity

k-degree anonymity A graph ***G(V, E)*** is ***k-degree anonymous*** if every node in ***V*** has the same degree as ***k-1*** other nodes in ***V***.



2-degree anonymous

Prop 1: If G is k1-degree anonymous, then it is also k2-degree anonymous, for every $k2 \leq k1$

[Properties] It prevents the re-identification of individuals by adversaries with *a priori* knowledge of the degree of certain nodes.

K-Anonymity: Intuition

- The information for each person contained in the released table **cannot be distinguished from at least k-1 individuals** whose information also appears in the release
  - Example: you try to identify a man in the released table, but the only information you have is his birth date and gender. There are k men in the table with the same birth date and gender.
- Any quasi-identifier present in the released table must appear in at least k records

K-Anonymity Protection Model

- Private table: T
- Released table: RT
- Attributes: $A_1, A_2, \ldots, A_n$
- Quasi-identifier subset: $A_i, \ldots, A_j$

> Let RT($A_1,\ldots,A_n$) be a table, $QI_{RT} = (A_i, \ldots, A_j)$ be the quasi-identifier associated with RT, $A_i,\ldots,A_j \subseteq A_1,\ldots,A_n$, and RT satisfy *k*-anonymity. Then, each sequence of values in RT[A_x] appears with at least *k* occurrences in RT[QI_{RT}] for $x=i,\ldots,j$.

Example of a k-Anonymous Table

	Race	Birth	Gender	ZIP	Problem
t1	Black	1965	m	0214*	short breath
t2	Black	1965	m	0214*	chest pain
t3	Black	1965	f	0213*	hypertension
t4	Black	1965	f	0213*	hypertension
t5	Black	1964	f	0213*	obesity
t6	Black	1964	f	0213*	chest pain
t7	White	1964	m	0213*	chest pain
t8	White	1964	m	0213*	obesity
t9	White	1964	m	0213*	short breath
t10	White	1967	m	0213*	chest pain
t11	White	1967	m	0213*	chest pain

Figure 2 Example of *k*-anonymity, where *k*=2 and QI={*Race*, *Birth*, *Gender*, *ZIP*}

l-Diversity

Caucas	787XX	Flu
Caucas	787XX	Shingles
Caucas	787XX	Acne
Caucas	787XX	Flu
Caucas	787XX	Acne
Caucas	787XX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Shingles
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Flu

Sensitive attributes must be "diverse" within each quasi-identifier equivalence class

L-Diversity

- T*: the Anonymized Table
- q*: the generalized value of q in the published table T*
- s: a possible value of the sensitive attribute
- n(q*,s’): number of tuples with sensitive attribute s’ and non-sensitive attribute q*
- q*-block: the set of tuples in T* whose non-sensitive attribute values generalize to q*

L-Diversity

- Lack diversity: lack of diversity in the sensitive attribute manifests itself as follows:

$$\forall s' \neq s, \quad n_{(q^\star, s')} \ll n_{(q^\star, s)}$$

L-Diversity

- Then, **L-Diversity Principle** can be defined as:
  - A q*-block is L-diverse if contains at least L "well-represented" values for the sensitive attribute S.
  - A table is L-diverse if every q*-block is L-diverse.

An example

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	130**	< 30	$*$	Heart Disease
2	130**	< 30	$*$	Heart Disease
3	130**	< 30	$*$	Viral Infection
4	130**	< 30	$*$	Viral Infection
5	1485*	≥ 40	$*$	Cancer
6	1485*	≥ 40	$*$	Heart Disease
7	1485*	≥ 40	$*$	Viral Infection
8	1485*	≥ 40	$*$	Viral Infection
9	130**	3*	$*$	Cancer
10	130**	3*	$*$	Cancer
11	130**	3*	$*$	Cancer
12	130**	3*	$*$	Cancer

4-anonymous table

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	1305*	≤ 40	$*$	Heart Disease
4	1305*	≤ 40	$*$	Viral Infection
9	1305*	≤ 40	$*$	Cancer
10	1305*	≤ 40	$*$	Cancer
5	1485*	> 40	$*$	Cancer
6	1485*	> 40	$*$	Heart Disease
7	1485*	> 40	$*$	Viral Infection
8	1485*	> 40	$*$	Viral Infection
2	1306*	≤ 40	$*$	Heart Disease
3	1306*	≤ 40	$*$	Viral Infection
11	1306*	≤ 40	$*$	Cancer
12	1306*	≤ 40	$*$	Cancer

3 diverse table

- Using a 3-diverse table, we no longer are able to tell if Bob (a 31 year old American from zip code 13053) has cancer.
- We also cannot tell if Umeko(a 21 year old Japanese from zip code 13068) has a viral infection or cancer.

Probabilistic inference attacks over l-Diversity

- Each equivalence class has at least **l well-represented** sensitive values

...	**Disease**
	...
	HIV
	HIV
	...
	HIV
	pneumonia
	bronchitis
	...

10 records

8 records have HIV

2 records have other values

- Doesn’t prevent probabilistic inference attacks
  - Infer: the patient has HIV with large possibility

t-Closeness overview

- Privacy is measured by the information gain of an observer.
- We assume:
  - B0: Alice believes that Bob has the virus because he has been acting sick.
  - B1: Alice gets a summary report of the table and learns that only 1% of the population has the virus. This distribution is Q, the distribution of the sensitive attribute in the whole table. She believes that Bob is in that one percent.
  - B2: Alice takes a look at the table, and finds that Bob is in equivalence class 3 because he is 32 and lives in zip code 47623. She learns P, the distribution of the sensitive attribute values in this class. Based on P she decides that it is actually quite likely that Bob has the virus.

t-Closeness overview

- l-diversity limits the gain between B0 (belief before any knowledge of the table) and B2 (belief after examining the table and the relevant equivalence class) by requiring that P (distribution in the equivalence class) has diversity.

- Q (global distribution in the table) should be treated as public information.

- If the change from B0 to B1 is large, means that the Q contains lots of new information. But we can't control people's access to Q, so we shouldn't worry about it.

- Therefore should focusing on limiting the gain between B1 and B2. We can do so by limiting the difference between P and Q. The closer P and Q are, the closer B1 and B2 are.

t-Closeness definition

- An equivalence class is said to have **t-closeness**
  - if the distance between the distribution of a sensitive attribute (P) in this class and the distribution of the attribute in the whole table(Q) is no more than a threshold t.
  - A table is said to have t-closeness if all equivalence classes have t-closeness.

t-Closeness

Caucas	787XX	Flu
Caucas	787XX	Shingles
Caucas	787XX	Acne
Caucas	787XX	Flu
Caucas	787XX	Acne
Caucas	787XX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Flu
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Shingles
Asian/AfrAm	78XXX	Acne
Asian/AfrAm	78XXX	Flu

Distribution of sensitive attributes within each quasi-identifier group should be “close” to their distribution in the entire original database

Distance measurement

- Now that we've confirmed that limiting the difference between *P* and *Q* is the key to privacy, we need a way to measure the distance.
  - m: the number of sensitive values in an equivalence class
  - $P=(p_1,p_2,\ldots,p_m)$, $Q=(q_1,q_2,\ldots,q_m)$
- Here are some naive measurements:
  - Method 1: variational distance

$$\mathsf{D}[\mathbf{P},\mathbf{Q}] = \sum_{i=1}^{m} \frac{1}{2}|p_i - q_i|.$$

Distance measurement

- Example

	ZIP Code	Age	Salary	Disease
1	47677	29	3K	gastric ulcer
2	47602	22	4K	gastritis
3	47678	27	5K	stomach cancer
4	47905	43	6K	gastritis
5	47909	52	11K	flu
6	47906	47	8K	bronchitis
7	47605	30	7K	bronchitis
8	47673	36	9K	pneumonia
9	47607	32	10K	stomach cancer

Table 3. Original Salary/Disease Table

	ZIP Code	Age	Salary	Disease
1	476**	2*	3K	gastric ulcer
2	476**	2*	4K	gastritis
3	476**	2*	5K	stomach cancer
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
7	476**	3*	7K	bronchitis
8	476**	3*	9K	pneumonia
9	476**	3*	10K	stomach cancer

Table 4. A 3-diverse version of Table 3

- Overall distribution of the Income attribute:

Q = {3k, 4k, 5k, 6k, 7k, 8k, 9k, 10k, 11k}

- The first equivalence class in Table 4 has distribution:

P1 = {3k, 4k, 5k}

- The second equivalence class has distribution:

P2 = {6k, 8k, 11k}

D(P1,Q)=0.5*(|1/3-1/9|+ |1/3-1/9| + |1/3-1/9| + |0-1/9|+ |0-1/9|+ |0-1/9|+ |0-1/9|+ |0-1/9|+ |0-1/9|)=1/2

D(P2,Q)=0.5*(|1/3-1/9|+ |1/3-1/9| + |1/3-1/9|+|0-1/9|+ |0-1/9|+ |0-1/9|+ |0-1/9|+ |0-1/9|+ |0-1/9|)=1/2

We have D(P1,Q)= D(P2,Q)

Distance measurement

- Here are some naive measurements:
  - Method 2: Kullback-Leibler (KL) distance

$$\mathrm{D}[\mathbf{P},\mathbf{Q}] = \sum_{i=1}^{m} p_i \log \frac{p_i}{q_i} = H(\mathbf{P}) - H(\mathbf{P},\mathbf{Q})$$

	0	1
D1	0.1	0.9
D2	0.2	0.8
D3	0.9	0.1

- H(P) is the entropy of P

$$H(\mathbf{P}) = \sum_{i=1}^{m} p_i \log p_i$$

- H (P, Q) is the cross-entropy of P and Q

$$H(\mathbf{P},\mathbf{Q}) = \sum_{i=1}^{m} p_i \log q_i$$

Definition

- Differential Privacy
  - A mechanism $\mathcal{A}$ satisfies (ε, δ)-differential privacy if for any neighboring databases D, D' differing in only one tuple and any output $S \in O(\mathcal{A})$ which represents the possible output set of $\mathcal{A}$,

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

  - If $\delta = 0$, $\mathcal{A}$ satisfies ε-differential privacy

We mainly focus on ε-differential privacy, as most studies do …

Sensitivity

- Global sensitivity
  - For any query function $f: D \rightarrow R^d$,where D is a dataset and R^d is a *d*-dimension real-valued vector, the global sensitivity of *f* is defined as

$$\Delta f = \max_{D,\, D'} ||f(D) - f(D')||_1$$

  where D and D' denote neighboring databases differing in only one tuple and $|| \cdot ||_1$ denotes l_1 norm.

l_1 norm: $||v||_1 = \sum_{1 \leq i \leq d} |v_i|$

Sensitivity

- Tips
  - The global sensitivity means the maximal change of query result when changing a tuple (extreme case).
  - The global sensitivity is only related to query function, and has nothing to do with database itself.

Name	Salary
Hunter	50000
Alice	50000
Bob	20000
Eric	100000
Frank	60000

f: Compute the total salary
Valid salary: [10000, 100000]
Δf=90000 for both databases

Name	Salary
Pedro	80000
Alice	50000
Mata	10000
Eric	100000
Frank	60000

Sensitivity

- Example: Count function: $\Delta f = 1$

Name	Flu
Hunter	1
Alice	0
Eric	0
Frank	1

Count(1)=2

Neighboring

Name	Flu
Hunter	1
Alice	0
Bob	1
Eric	0
Frank	1

Count(1)=3

Neighboring

Name	Flu
Hunter	1
Alice	0
Bob	0
Eric	0
Frank	1

Count(1)=2

Sensitivity

- Example: Histogram Query $\Delta f = 2$

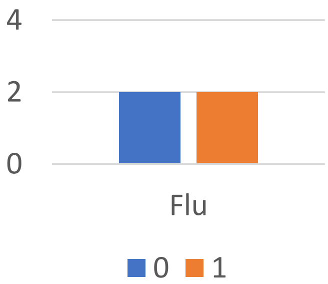
Name	Flu
Hunter	1
Alice	0
Eric	0
Frank	1

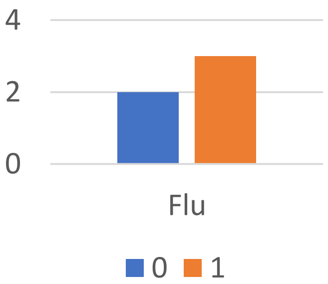
Neighboring

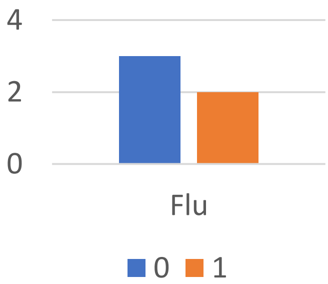
Name	Flu
Hunter	1
Alice	0
Bob	1
Eric	0
Frank	1

Neighboring

Name	Flu
Hunter	1
Alice	0
Bob	0
Eric	0
Frank	1







$Hist =< 2,2 >$

$Hist =< 2,3 >$

$Hist =< 3,2 >$

$|| < 2,2 > - < 2,3 > ||_1 = 1$

$|| < 2,3 > - < 3,2 > ||_1 = 2$

Sensitivity

- Example: Median
  - Suppose extreme case D: $(0, 0, 0, n, n)$
  - A neighboring database D': $(0, 0, n, n, n)$
  - $Med(D) = 0$
  - $Med(D') = n$
  - $\Delta f = n$ (the maximal possible element)

Sensitivity

- Local sensitivity
  - For any query function $f: D \rightarrow R^d$, the local sensitivity of f is defined as

$$LS_f(D) = \max_{D'} ||f(D) - f(D')||_1$$

where D and D' denote neighboring databases differing in only one tuple and $||\cdot||_1$ denotes l_1 norm.

Sensitivity

- Local sensitivity
  - f: Compute the maximal salary difference
  - Valid salary: [10000, 100000]

Bounded neighboring is considered

Name	Salary
Hunter	50000
Alice	50000
Bob	20000
Eric	10000
Frank	60000 $\rightarrow$ 100000

$$LS_f(D_1) = 90000 - 50000 = 40000$$

Name	Salary
Pedro	80000
Alice	50000
Mata	70000
Eric	15000 $\rightarrow$ 50000
Frank	60000

$$LS_f(D_2) = 65000 - 30000 = 35000$$

Name	Salary
Pedro	80000
Alice	60000
Mata	75000
Eric	100000
Frank	60000 $\rightarrow$ 10000

$$LS_f(D_3) = 90000 - 40000 = 50000$$

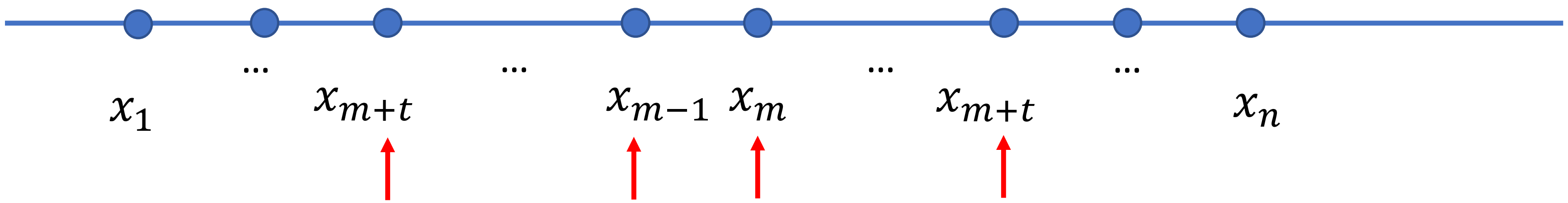
$LS_f(D)$ is much smaller than Δf which is 90000

$f(D)$ $f(D')$

Sensitivity

- Example
  - Median:
    - Suppose D: $(x_1, x_2, \dots, x_{n-1}, x_n)$, n is odd
    - $Med(D) = x_m$, $m = (n+1)/2$
    - $LS_f(D) = \max(x_m - x_{m-1}, x_{m+1} - x_m)$

$LS_f(D)$ is usually much smaller than Δf which is the maximal possible element



Sensitivity

- Smooth Sensitivity
  - Motivation
    - Avoid to employ global sensitivity
    - Databases with smaller local sensitivity could be calibrated with smaller noise
    - Add instance-specified noise while differential privacy is preserved at the same time

Sensitivity

- Smooth Sensitivity
  - Requirement
    - The difference of smooth sensitivity for neighboring databases should be bounded
    - No smaller than local sensitivity
    - No larger than global sensitivity

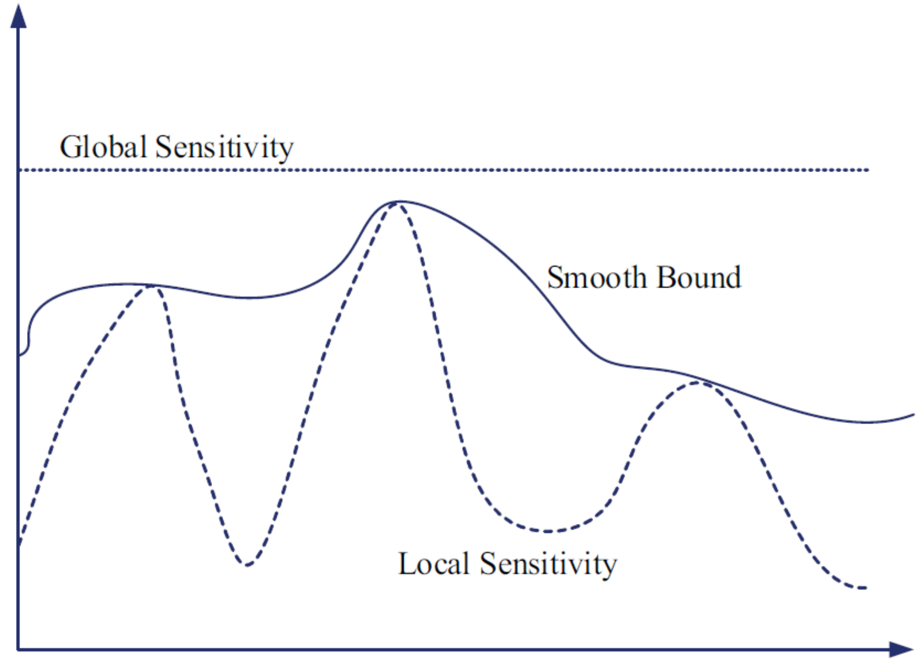
Sensitivity

- Smooth Bound
  - For β > 0, a smooth function $S: D \rightarrow R^{+}$ is a β -smooth upper bound on the local sensitivity of f if it satisfies the following requirements:
    - $S(D) \geq LS_f(D)$
    - $S(D) \leq e^{\beta} LS_f(D)$

A function S that is an upper bound on LS_f at all points and such that $\ln(S(\cdot))$ has low sensitivity

Sensitivity

- Smooth Bound



Note that the constant function $S(x) = \Delta f$ meets the requirements with $\beta = 0$.

Sensitivity

- Smooth sensitivity
  - For any query function $f: D \to R^d$, the smooth sensitivity of f is defined as

$$S^*_{f,\beta}(D) = \max_{D'}(LS_f(D') \cdot e^{-\beta d(D,D')})$$

where $d(D, D')$ denotes the Hamming distance between neighboring databases D and D'.

Sensitivity

- Property of Smooth Sensitivity
  - $S^*_{f,\beta}$ is a β-smooth upper bound on LS_f. In addition, $S^*_{f,\beta}(D) \leq S(D)$ for all database D for every β-smooth upper bound S on LS_f.
  - Key Points
    - $S^*_{f,\beta}(D) \geq LS_f(D)$
    - $S^*_{f,\beta}(D) \leq e^{\beta} LS_f(D)$
    - $S^*_{f,\beta}$ is the smallest β-smooth upper bound on LS_f

Sensitivity

- Smooth Sensitivity Brings Differential Privacy
  - 1-Dimensional Case
    - Let $f: \mathrm{D} \rightarrow \mathbb{R}$ be any real-valued function and let $S: \mathbb{D} \rightarrow \mathbb{R}$ be a β-smooth upper bound on the local sensitivity of f then
      - If $\beta \leq \frac{\varepsilon}{2(\gamma+1)}$ and $\gamma > 1$, the algorithm $x \mapsto f(x) + \frac{2(\gamma+1)S(x)}{\varepsilon}\eta$, where η is sampled from distribution with density $h(z) \propto \frac{1}{1+|z|^{\gamma}}$, is ε-differentially private

Added noise

α and β are parameters of the noise distribution

Sensitivity

- Smooth Sensitivity Brings Differential Privacy
  - 1-Dimensional Case
    - Let $f\colon \mathrm{D} \to \mathbb{R}$ be any real-valued function and let $S\colon \mathbb{D} \to \mathbb{R}$ be a β-smooth upper bound on the local sensitivity of f then
      - If $\beta \le \frac{\varepsilon}{2\ln(\frac{2}{\delta})}$ and $\delta \in (0,1)$, the algorithm $x \mapsto f(x) + \frac{2S(x)}{\varepsilon}\eta$, where $\eta \sim Lap(1)$ (ε, δ)-differentially private

Added noise

α and β are parameters of the noise distribution

$$S^*_{f,\beta}(D) = \max_{D'}(LS_f(D') \cdot e^{-\beta d(D,D')})$$

Sensitivity

- Example of Calculating Smooth Sensitivity
  - Median:
    - Suppose D: $(x_1, x_2, \dots, x_{n-1}, x_n)$, n is an odd
    - $Med(D) = x_{m,}\, m = (n+1)/2$
    - $LS_f(D) = \max(x_m - x_{m-1,}\, x_{m+1} - x_m)$
    - Let k denotes up to k tuples changed
  - The smooth sensitivity of the median is

    $$S^*_{f_{med,\varepsilon}}(D) = \max_{k=0,\dots,n}(e^{-k\beta} \cdot \max_{t=0,\dots,k+1} max(x_{m+t} - x_{m+t-k-1}, x_{m+t+1} - x_{m+t}))$$

    It can be computed in $O(n^2)$

$$S^*_{f,\beta}(D) = \max_{D'}(LS_f(D') \cdot e^{-\beta d(D,D')})$$

Sensitivity

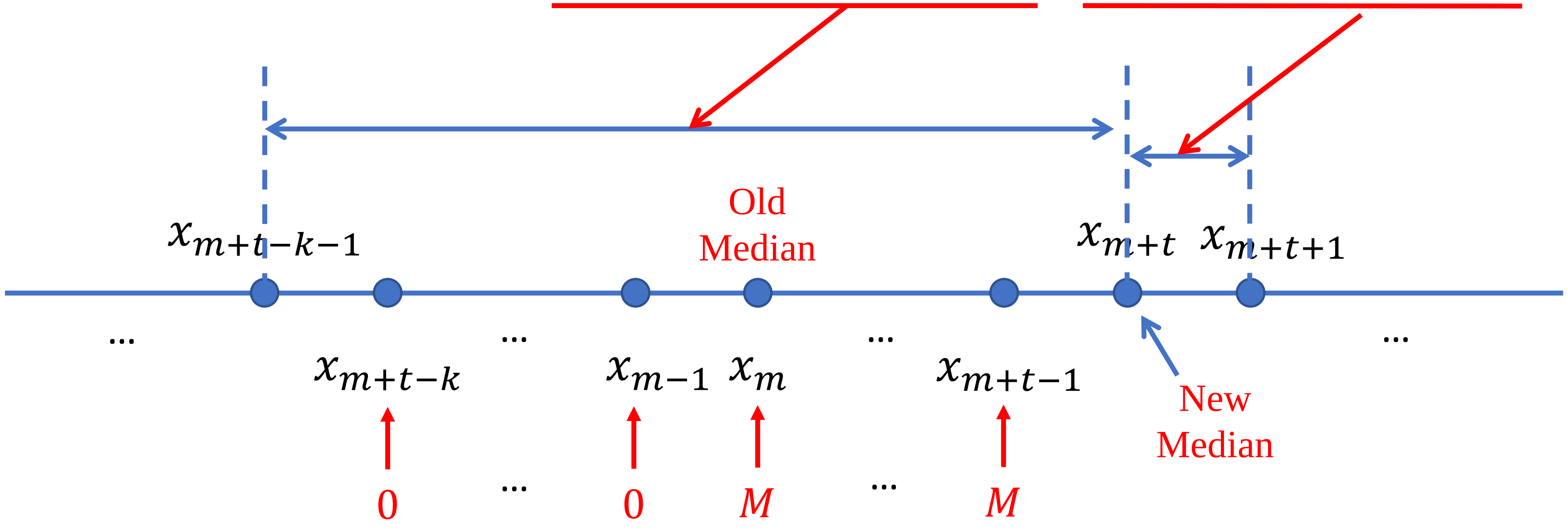
- An Idea of Computing $S^*_{f,\beta}(D)$
  - Suppose we change up to k tuples
    $$A^{(k)}(D) = \max_{D' \in \mathbb{D}: d(D,D') \leq k} LS_f(D')$$
  - Smooth sensitivity could be expressed using $A^k(D)$
    $$S^*_{f,\beta}(D) = \max_{k=0,\ldots,n} e^{-k\beta}\left(\max_{D' \in \mathbb{D}: d(D,D') \leq k} LS_f(D')\right)$$
    $$= \max_{k=0,\ldots,n} e^{-k\beta} A^k(D)$$

Sensitivity

- Computing $S^*_{f_{med,\varepsilon}}(D)$
  - For f $= Median$

$$A^{(k)}(D) = \max_{D' \in \mathbb{D}: d(D,D') \le k} LS_f(D')$$

$$= \max_{t=0,\dots,k} max(x_{m+t} - x_{m+t-k-1}, x_{m+t+1} - x_{m+t})$$



Sensitivity

- Computing $S^*_{f_{med},\varepsilon}(D)$
  - For f $= Median$

$$A^{(k)}(D) = \max_{D' \in \mathbb{D}: d(D,D') \leq k} LS_f(D')$$

Data range: [0, 10], $Med(D) = x_5 = 5$

$$D = (1,2,3,4,5,6,7,8,9)$$

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	2	3	4	5	6	7	8	9

What is th D to $D'(d(D,D'$ $\leq k)$?

Sensitivity

- To Compute the Maximum $LS_f(D')$
  - Solution to get maximum candidates
    - Let $t = 0, \dots, k$
    - Change t tuples to 10, starting from x_5 to the right
    - Change $\mathrm{k} - t$ tuples to 0, starting from x_4 to the left
  - Change 0 tuple

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	2	3	4	5	6	7	8	9

    - No tuples are change … (D)
    - $\max_{D' \in \mathbb{D}: d(D,D') \le k} LS_f(D') = LS_f(D) = \max\{x_5 - x_4, x_6 - x_5\}$

Sensitivity

- Change 1 tuple
  - Case 1: $k = 1, t = 0$

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	2	3	0	5	6	7	8	9

x_4	x_1	x_2	x_3	x_5	x_6	x_7	x_8	x_9
0	1	2	3	5	6	7	8	9

- $LS_f(D') = \max\{\boldsymbol{x_5} - \boldsymbol{x_3}, \boldsymbol{x_6} - \boldsymbol{x_5}\}$
  - Case 2: $k = 1, t = 1$

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	2	3	4	10	6	7	8	9

x_1	x_2	x_3	x_4	x_6	x_7	x_8	x_9	x_5
1	2	3	4	6	7	8	9	10

- L

$$S^*_{f_{med,\varepsilon}}(D) = \max_{k=0,\dots,n}\left(e^{-k\beta} \cdot \max_{t=0,\dots,k+1} max(x_{m+t} - x_{m+t-k-1}, x_{m+t+1} - x_{m+t})\right)$$

Sensitivity

- Change 2 tuple
  - Case 1: $k = 2, t = 0$

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	2	0	0	5	6	7	8	9

x_4	x_3	x_1	x_2	x_5	x_6	x_7	x_8	x_9
0	0	1	2	5	6	7	8	9

  - Case 2: $k = 2, t = 1$

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	2	3	0	10	6	7	8	9

x_4	x_1	x_2	x_3	x_6	x_7	x_8	x_9	x_5
0	1	2	3	6	7	8	9	10

  - Case 3: $k = 2, t = 2$

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
1	2	3	4	10	10	7	8	9

x_1	x_2	x_3	x_4	x_7	x_8	x_9	x_5	x_6
1	2	3	4	7	8	9	10	10

$$S^*_{f_{med,\varepsilon}}(D) = \max_{k=0,\ldots,n}\left(e^{-k\beta} \cdot \max_{t=0,\ldots,k+1} max(x_{m+t} - x_{m+t-k-1}, x_{m+t+1} - x_{m+t})\right)$$

Laplace Mechanism

- Mechanism
  - Definition of Laplace Mechanism
    - Given any function $f: \mathbb{N}^{|\mathcal{X}|} \rightarrow \mathbb{R}^k$, the **Laplace Mechanism** is defined as:

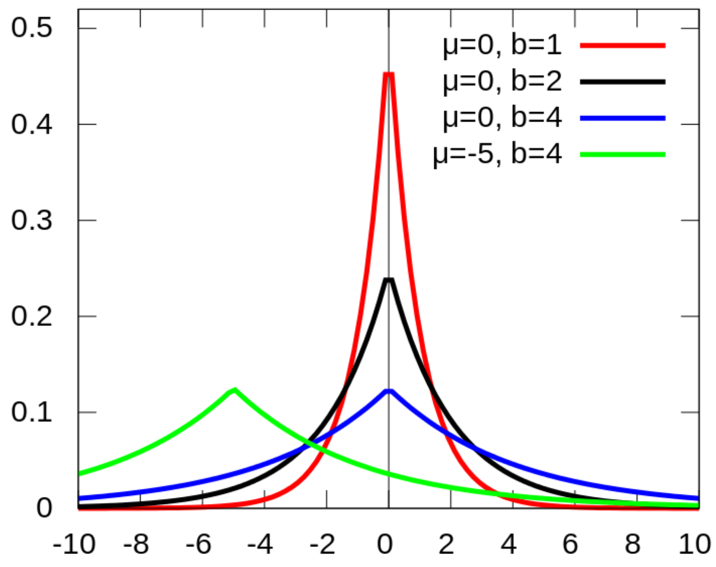
$$\mathcal{M}(D, f(.), \varepsilon) = f(D) + (Y_1, \dots, Y_k)$$

where Y_i is independent and identically distributed random variables drawn from $Lap(\Delta f/\varepsilon)$.

Laplace Mechanism works for real valued functions

Laplace Mechanism

- Mechanism
  - $Lap(\Delta f/\varepsilon)$: noise in Laplace Mechanism
    - Larger Δf brings larger noise
    - Smaller ε brings larger noise



Question:
Which Laplace Distribution brings the smallest noise?

$$Lap(x) = \frac{1}{2b}\exp(-\frac{|x|}{b})$$

$$b = \Delta f/\varepsilon$$

Laplace Mechanism

- Example
  - Among 10000 family names, which is the most common?
    - Utilization of histogram queries
    - Set $\varepsilon = 1$
    - To count the number of each family name, add independent noise $Y_i \sim Lap(1)$ ($\Delta f = 1, \varepsilon = 1$)
      - $\Pr[|Y_i| < ?] \geq 95\%$
      - Is it a small error for large population, say 300000 persons
    - Report the family name with the largest count

Laplace Mechanism

- Example
  - $\Delta f = 1, \varepsilon = 1, k = 10000$, set $\delta = 0.05$
  - Recall the property of Laplace Distribution

$$\Pr[||f(D) - y||_\infty \geq \ln(\frac{k}{\delta}) \times (\frac{\Delta f}{\varepsilon})] \leq \delta$$

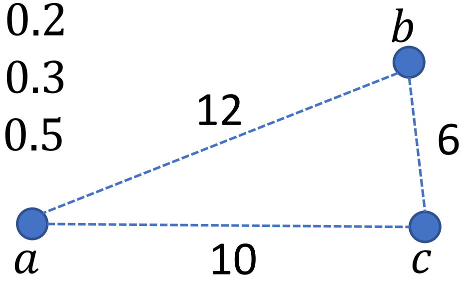
  - We can get $\Pr[Y_i \geq \ln(\frac{10000}{0.05}) \times \frac{1}{1}] \leq 0.05$, that is $\Pr\left[Y_i < \ln\left(\frac{10000}{0.05}\right)\right] \geq 95\%$
  - $\ln\left(\frac{10000}{0.05}\right) \approx 12.2$

It is a small error for large population, say 300000 persons

Location Privacy

- Existing Notions of Privacy
  - Expected Distance Error
    - A natural way to quantify the accuracy by which an adversary can guess the real location

$\Pr(a) = 0.2$
$\Pr(b) = 0.3$
$\Pr(c) = 0.5$



Three locations and their prior probability

Location\Guess	a	b	c
a	0.8	0.1	0.1
b	0.1	0.6	0.3
c	0.2	0.3	0.5

Adversary's guessing strategy

$\mathcal{K}$

Location Preserving Mechanism

Mechanism design: Maximizing the expected distance error

Expected Distance Error:

At a: $0.8 \times 0 + 0.1 \times 12 + 0.1 \times 10 = 2.2$

At b: $0.1 \times 12 + 0.6 \times 0 + 0.3 \times 6 = 3$

At c: $0.2 \times 10 + 0.3 \times 6 + 0.5 \times 0 = 3.8$

Overall: $0.2 \times 2.2 + 0.3 \times 3 + 0.5 \times 3.8 = 3.24$

Location Privacy

- Existing Notions of Privacy
  - Expected Distance Error
    - Inaccuracy estimation of adversary's side information leads to poorly designed mechanism

$\Pr(a) = 0.2$
$\Pr(b) = 0.3$
$\Pr(c) = 0.5$

Inaccurate prior probability

Location\Guess	a	b	c
a	0.8	0.1	0.1
b	0.1	0.6	0.3
c	0.2	0.3	0.5

Inaccurate adversary's guessing strategy

Poor M esign

$\mathcal{K}$

➢ Aim zing the inaccurate expected distance error

Location Preserving Mechanism

Inaccurate expected distance error