DEPARTMENT OF MATHEMATICAL INFORMATICS
THE UNIVERSITY OF TOKYO

16章: 出力摂動によるデータプライバシーの確保
Chapter 16: Private Data Analysis via Output Perturbation

in *Aggarwal & Yu* (*Eds.*): *Privacy-preserving Data Mining*

担当: 鹿島 久嗣
東京大学
情報理工学研究科
数理情報学専攻

Some figures are borrowed from the book chapter, and
Nissim et al. "Smooth Sensitivity and Sampling in Private Data Analysis" in STOC'07

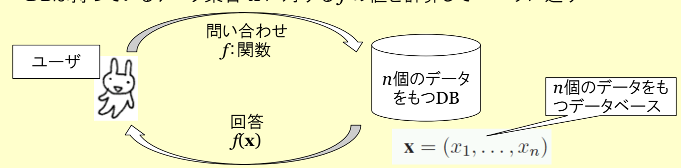
この章では、問い合わせに対するデータベースからの回答(出力)に摂動(ノイズ)を加えることで、データのプライバシーを守る方法を学びます

- 出力摂動によるデータプライバシー確保の枠組みは最近のPPDM業界で結構注目されているらしい
- とくに「差分プライバシー」という概念の導入によって、攻撃者の事前知識などに左右されずにプライバシー強度を(数学的に)議論できるところが関係者にウケているようだ
  - ただし、実用的にはまだ未知数
- そこで、この章では、その「差分プライバシー」の概念と、その枠組みにおけるプライバシーの実現方法を紹介する

背景
データベース問い合わせモデルにおけるデータのプライバシー確保

この章で考えるモデル: ユーザ(敵)がデータベースに問い合わせを行う状況

- ユーザ(ここでは敵)が、信頼のおけるデータベース(DB)に対して問い合わせを行う状況を考える
  - ユーザがDBに計算してほしい関数 f を渡す
  - DBは持っているデータ集合 $\mathbf{x}$ に対する f の値を計算してユーザに返す

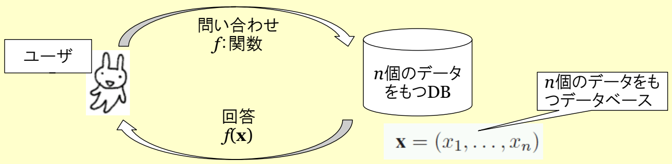


  - 関数 f の具体例
    - 和の問い合わせ(Sum query):ある性質gを満たすデータの数を数える

$$\mathrm{sum}_g(\mathbf{x}) = \sum_{i=1}^{n} g(i, x_i) \ \ where\ \ g : \mathbb{N} \times \mathcal{D} \to [0,1]$$

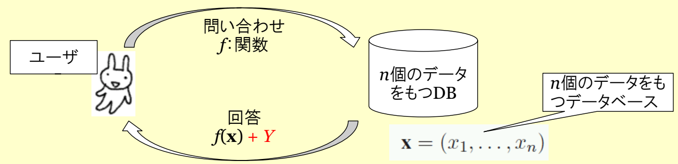
達成したいこと: DBは、ユーザの期待には応えつつも、データのプライバシーは守りたい

- 2つの相対する目標:
  - ユーザの期待には応えたい（$f(\mathbf{x})$ は計算してあげたい）
  - データのプライバシーを守りたい（$\mathbf{x}$ についての情報は漏らしたくない）
- たとえば、f として「i番目のデータのj番目の次元を返す」などとすれば、データのプライバシーは完全に破られてしまう



実現のためのアイディア: DBからの回答に、乱数による摂動を入れて返せばよいのではないか？

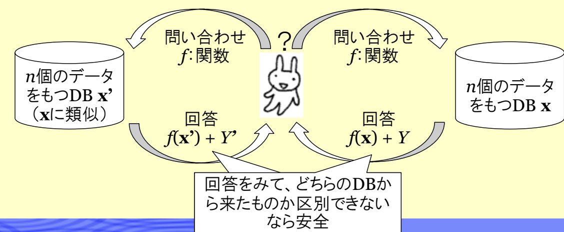
- 関数の値 $f(\mathbf{x})$ をそのまま返すのではなく、ランダム性のある摂動 Y を加えて返す（サニタイゼーション; 浄化）ことで、ユーザの要求とデータのプライバシーとを両立する
  - $f(\mathbf{x}) \rightarrow f(\mathbf{x}) + Y$ とする
  - たとえば、Y は平均0、分散1の正規分布に従うなど
- 考えるべきこと:
  - どのような分布に従って Y を発生すればデータのプライバシーが守れるか？
  - ユーザの要求（ f の精度）には、どの程度応えられるか？



新しいプライバシーの強度指標 差分プライバシー

何をもって“安全”とみなすか？: 良く似た別のDBの回答と区別できないならば“相対的に安全”であるとする

- ユーザの事前知識によって、プライバシーの強度はまちまちであるので、ある種の“相対的な”プライバシー強度を考える
- 「良く似た別のDBの回答と区別できないならば、（相対的に）安全」であるとする
  - いいかえれば「ユーザが回答をみたときに（自分のDBと別のDBの）どちらのDBからの回答か区別できないならば（相対的に）安全」
  - このDBが<u>他のDBよりも</u>多くプライベート情報を漏らすことはない、という気持ち
- 「良く似た」と「区別できない」はどのように定義できるだろうか？



「良く似た別のDB」の定義：要素がひとつだけ異なるDB

- あるDBに対する「隣のDB」を定義するために、2つのDB間の距離を定義する
- 2つのDB $\mathbf{x}$ と $\mathbf{x}'$ の距離 $\mathbf{dist}_H(\mathbf{x}, \mathbf{x}')$ は、異なる要素の数によって定義する

DEFINITION 16.4 (HAMMING DISTANCE, NEIGHBOR DATABASES)

$$\mathbf{dist}_H(\mathbf{x}, \mathbf{x}') = |\{i : x_i \neq x'_i\}|$$

  - $\mathbf{x}=\{1,2,3\}$ と $\mathbf{x}'=\{1,2\}$ の距離は1、$\mathbf{x}=\{1,2,3\}$ と $\mathbf{x}'=\{1,2,4\}$ の距離も1
- 距離が1のDB同士を「隣接する」ということにする

差分プライバシー（differential privacy）：ある答えが返ってくる確率が、良く似たDBとほぼ同じなら安全とする

- 「良く似た別のDBの回答と区別できないならば、（相対的に）安全」もしくは「ユーザが回答をみたときに（自分のDBと別のDBの）どちらのDBからの回答か区別できないならば（相対的に）安全」を数学的に表す
- 定義：ε-差分プライベート（εはある正の小さい定数）であるとは、
  - あるDB $\mathbf{x}$ から回答 $t = f(\mathbf{x})+Y$ が返ってくる確率を $h_{\mathbf{x}}(t)$
  - 隣のDB $\mathbf{x}'$ から回答 $t = f(\mathbf{x}')+Y'$ が返ってくる確率を $h_{\mathbf{x}'}(t)$

  としたときに、2つのDBから同じ答え t が返ってくる確率の比が

$$\frac{h_{\mathbf{x}}(t)}{h_{\mathbf{x}'}(t)} \leq e^{\varepsilon}$$ （つまり、大体等しい）

  が、隣接する全てのDBペア（$\mathbf{x}$, $\mathbf{x}'$）について成立すること
  - 書き換えると、$\log h_{\mathbf{x}}(t) - \log h_{\mathbf{x}'}(t) \leqq \varepsilon$ ともいえる
  - ε は小さいほど安全ということになる

差分プライバシーの重要な性質：複数のε-差分プライベートな関数を組合わせても、その性質は（ある程度）保たれる

- それぞれが ε-差分プライベートな、摂動操作（サニタイザー）が、q個あるとき、全体としては、εq-差分プライベートになる
  - ただし、摂動に使う確率変数は互いに独立であるとする

LEMMA 16.7 *Let* $\mathbf{San}_i$ *be* ε_i*-private for* $i = 1, \ldots, q$. *The sanitizer that answers according to* $\mathbf{San}_1, \ldots, \mathbf{San}_q$ *(where the randomness of each sanitizer is chosen independently of the other sanitizers) is* ε'*-private for* $\varepsilon' = \sum_i \varepsilon_i$.

proof:

$$\frac{\bar{h}_{\mathbf{x}}(t_1, \ldots, t_q)}{\bar{h}_{\mathbf{x}'}(t_1, \ldots, t_q)} = \frac{\prod_{i=1}^q h_{\mathbf{x}}^{(i)}(t_i)}{\prod_{i=1}^q h_{\mathbf{x}'}^{(i)}(t_i)} = \prod_{i=1}^q \frac{h_{\mathbf{x}}^{(i)}(t_i)}{h_{\mathbf{x}'}^{(i)}(t_i)} \leq e^{\sum_i \varepsilon_i} = e^{\varepsilon'}$$

差分プライバシーの実現
差分プライバシーを保つ摂動の設計

差分プライバシーを保つ摂動の設計：
ラプラス分布によって摂動を与えれば、差分プライバシーを実現できる

- 問い：ε-差分プライバシーを実現するには、どのような摂動分布を考えればよいだろうか？
  - つまり、$t = f(\mathbf{x})+Y$ の Y の従う確率分布 $h(Y)$をどのように定義すればよいか？
- こたえ：ラプラス分布を使えばよい
  - ラプラス分布は平均ゼロ、分散$2\lambda^2$ の、正規分布よりも裾野の厚い分布

$$\mathbf{Lap}(\lambda): \quad h(t) = \frac{1}{2\lambda} e^{-\frac{|t|}{\lambda}}$$

  - たとえば、和の問い合わせ（ある性質gを満たすデータの数を数える）の場合、パラメータを $\lambda=1/\varepsilon$ とすればよい
    - εを小さくしたいなら、摂動の分散を大きくしなければならない

差分プライバシーを保つ摂動の例：
ラプラス分布の摂動は、和の問い合わせに対して差分プライバシーを保証する

- 和の問い合わせ（Sum query）：ある性質gを満たすデータの数を数える

$$\mathrm{sum}_g(\mathbf{x}) = \sum_{i=1}^{n} g(i, x_i) \ \textit{where}\ g : \mathbb{N} \times \mathcal{D} \rightarrow [0,1]$$

- 摂動に使う確率分布hをラプラス分布 $\mathbf{Lap}(1/\varepsilon)$ にすれば、ε-差分プライベート
  - 証明：ラプラス分布であることが本質的に効いてくる（正規分布ではダメ）

$h_{\mathbf{x}}$: DB $\mathbf{x}$ から 回答 tが返ってくる確率

$$\frac{h_{\mathbf{x}}(t)}{h_{\mathbf{x}'}(t)} = \frac{h(t - f(\mathbf{x}))}{h(t - f(\mathbf{x}'))} \leq e^{\varepsilon|f(\mathbf{x}) - f(\mathbf{x}')|} \leq e^{\varepsilon}$$

h: 摂動に使う確率分布（ラプラス）

$$\frac{h(t)}{h(t')} = \frac{e^{\frac{-|t|}{\lambda}}}{e^{\frac{-|t'|}{\lambda}}} = e^{\frac{|t'|-|t|}{\lambda}} \leq e^{\frac{|t-t'|}{\lambda}}$$

三角不等式より

ラプラス分布の定義より

$$\mathbf{Lap}(\lambda): \quad h(t) = \frac{1}{2\lambda} e^{-\frac{|t|}{\lambda}}$$

摂動に使うラプラス分布の幅の決定：
ラプラス分布の幅は（小さいほど良い）大域的感度（Global Sensitivity）によって決まる

- 前頁の結果を、和問い合わせ以外の場合に一般化する
- 大域的感度（Global sensitivity）：隣接する全てのDBペア（$\mathbf{x}$, $\mathbf{x}'$）に対する、関数 f の値の差の最大値

$$\mathbf{GS}_f = \max_{\mathbf{x},\mathbf{x}' : \mathbf{dist}_H(\mathbf{x},\mathbf{x}')=1} \left\| f(\mathbf{x}) - f(\mathbf{x}') \right\|_1$$

  - たとえば、和問い合わせの場合、1 になる
- 定理：$t = f(\mathbf{x})+Y$ の Y の従う確率分布 $h(Y)$を$\mathbf{Lap}(\mathbf{GS}_f/\varepsilon)$ にすれば ε-差分プライベート になる

差分プライバシーの実現例
差分プライバシーを保つことのできる関数の具体例

差分プライバシーを保つための条件：大域的感度が小さければよい

- これまでの結果によれば、ある関数fに対し、その大域的感度

$$\mathbf{GS}_f = \max_{\mathbf{x},\mathbf{x}':\,\mathbf{dist}_H(\mathbf{x},\mathbf{x}')=1} \left\| f(\mathbf{x}) - f(\mathbf{x}') \right\|_1$$

  を計算できれば、その値を使ったラプラス分布 $\mathbf{Lap}(\mathbf{GS}_f/\varepsilon)$ を使ってプライベート化できる
- ただし、大域的感度はある程度小さくないと、もともとのf の値を凌駕するほど大きな摂動を加えなければいけなくなる（＝ユーザーの計算精度の要求を満たせない）
- では、どんな関数に対しては、小さな大域的感度が具体的に計算可能だろうか？
  - 和問い合わせ(sum query)
  - 平均・共分散
  - ヒストグラム
  - 部分集合和（subset sum）
  - …

小さな大域的感度が得られる関数の例：和問い合わせ、平均、分散、ヒストグラム、部分集合和、…

- では、どんな関数に対して大域的感度を具体的に計算可能だろうか？
  - 和問い合わせ(sum query)：$\mathbf{GS}_{\texttt{sum}} \le 1$
  - ベクトルの平均： $\mathbf{GS}_{\texttt{mean}} \le 2\gamma/n$
    - n個あるベクトル（データ）のそれぞれの長さがγ以下とすると、一個のデータの影響は大体 γ/n くらい
  - 共分散：$\mathbf{GS}_{\texttt{cov}} \le 8\gamma^2/n$
    - ベクトルの長さがγなら、分散のn倍は、γ^2 くらいの影響を受ける
  - ヒストグラム：$\mathbf{GS}_{\texttt{hist}} = 2$
    - どれかの箱に入っているデータを、別のところに移せば、2
  - 部分集合和（subset sum）： $\mathbf{GS}_{\texttt{subsets}} = 4\gamma$
    - qはデータをk個のグループに振り分ける関数

$$\texttt{subsets}_{q,g}(\mathbf{x}) = \left(\sum_{q(x_i)=1} g(x_i), \ldots, \sum_{q(x_i)=k} g(x_i) \right)$$

  - …

小さな大域的感度が得られる学習アルゴリズムの例：クラスタリング（k-平均クラスタリング法）

- k-平均クラスタリングのアルゴリズムの各ステップは

Modified k-Means Iteration:

Input: points $p_1, \ldots, p_n \in \mathbb{R}^d$, and centers $c_1, \ldots, c_k \in \mathbb{R}^d$.

1 [Compute the number of points in each of the sets S_j]

$$(\overline{s}_1, \ldots, \overline{s}_k) \leftarrow \texttt{hist}_q(p_1, \ldots, p_n)\,.$$

（DBが）データが各クラスタに何個づつ入るかを計算する（ヒストグラムで実装できる）

2 [Compute the sum of points in each of the sets S_j]

$$(\overline{m}_1, \ldots, \overline{m}_k) \leftarrow \texttt{subsets}_{q,g}(p_1, \ldots, p_n)\,.$$

（DBが）各クラスタに入ったデータの和を計算する（部分集合和で実装できる）

3 [Update each mean]
for $1 \le j \le k$:

$$\overline{c}_j' \leftarrow \frac{\overline{m}_j}{\overline{s}_j}\,.$$

（ユーザが）各クラスタに入ったデータの和を、個数で割って、平均（クラスタ中心）を計算する

- ヒストグラム、部分集合和、ともに大域的感度はすでに計算してあるので、これらに応じたラプラス分布を使って（ステップ1,2に）摂動を加えることで、ε-差分プライベートにできる
  - ステップ1: $\overline{s}_j \;=\; s_j + \hat{s}_j,$ where $\hat{s}_j \sim \left(\mathbf{Lap}(2/\varepsilon')\right)^d$
  - ステップ2: $\overline{m}_j \;=\; m_j + \hat{m}_j,$ where $\hat{m}_j \sim \left(\mathbf{Lap}(4\gamma/\varepsilon')\right)^d$

  where $\varepsilon' = \varepsilon/2$

ユーザーの計算精度に対する要求は満たされるか？：計算すべき関数 f に対して小さい大域的感度が保たれる必要がある

- ユーザの欲しいf の値を十分な精度で与えるためには、加える摂動の大きさはf のスケールと比較して十分に小さくなければいけない
  - 摂動の大きさは、大域的感度**GS**に従って決まる（概ね$\mathbf{GS}/\varepsilon$ くらいのスケールであると思ってよい）
- たとえば、
  - ヒストグラムでは、各箱に入るデータ数は、$\mathbf{GS}/\varepsilon = 2/\varepsilon$ よりも十分に大きい必要がある
  - 部分集合和では、各箱に入るデータ数とgの積は、$\mathbf{GS}/\varepsilon = 4\gamma/\varepsilon$（$\gamma$ は g の大きさ）よりも十分に大きい必要がある
- クラスタリング（ヒストグラムと部分集合和）の場合、各クラスター（各箱）に入るデータの数が十分に大きければよい

小さな大域的感度が得られる学習アルゴリズムの例：SVD（特異値分解）

- 特異値分解は、データの共分散行列の固有値問題を解く

SVD:

Input: The matrix $A \in \mathbb{R}^{d\times n}$ and a parameter $0 < k \leq n$.

1 [Approximate AA^T]

（DBが）共分散行列の各要素に摂動を加える

$$B \leftarrow \sum_i p_i p_i^T + Y \text{ where } Y \sim (\mathbf{Lap}(4\gamma^2))^{d\times d} .$$

2 Compute the top k eigenvectors of B.

（ユーザが）摂動の加わった行列の固有値問題を解く

- やはり、n が大きければ、共分散は十分に正しく与えられる

一般的な結果：統計的問い合わせ(statistical query)モデルは、差分プライベート化可能

不誠実

- 計算論的学習理論で用いられる統計的問い合わせ学習(statistical query learning)の枠組み
  - 統計的問い合わせ：ある性質pを問い合わせると、pを満たすデータの占める割合が、誤差τ以内で得られる
  - この問い合わせを使いながら、学習アルゴリズムはある関数を特定（＝学習）する
    - 問い合わせの答えに誤差があっても学習できるような関数が、統計的問い合わせ学習可能ということになる
- ここで、統計的問い合わせは、摂動を入れた和問い合わせに似ていることに気付く
  - DBは、誤差（摂動）を入れることで、データのプライバシーを守る
  - ユーザは、誤差（摂動）のある問い合わせ結果から、所望の結果を得る
- 統計的問い合わせ学習可能な関数であれば、差分プライベート化可能ということになる

より多くの関数を安全にするために 局所的感度と平滑化感度

これまでの大域的感度を用いた議論が破綻する場合：中央値の計算は大域的感度が大きくなりするので、これまでの議論が使えない

- 0と1からなる集合の中央値の問い合わせを考えてみる
- この2つは、隣り合っているので、大域的感度は1になる
  - { 0, 0, 0, 1, 1, 1, 1 }: 中央値 1
  - { 0, 0, 0, 0, 1, 1, 1 }: 中央値 0

$$\mathbf{GS}_f = \max_{\mathbf{x},\mathbf{x}':\mathbf{dist}_H(\mathbf{x},\mathbf{x}')=1} \left\| f(\mathbf{x}) - f(\mathbf{x}') \right\|_1$$

- しかし、データの値域の幅も1なので、Lap(1/ε) で摂動を入れると、大きすぎる
- そこで、データベースごとに異なる大きさの摂動を入れたらよいのでは？という気がしてくる

大域的感度が大きいときの対処法（失敗案）：
データごとの摂動を入れるために「局所的感度」を定義する → やっぱりダメ

- これまでは、全てのデータベースに対して、同じ確率分布を使って出力摂動を加えてきた
- しかし、一部の $(\mathbf{x}, \mathbf{x}')$ ペアが大域的感度を大きくしている

$$\mathbf{GS}_f = \max_{\mathbf{x},\mathbf{x}':\mathbf{dist}_H(\mathbf{x},\mathbf{x}')=1} \left\| f(\mathbf{x}) - f(\mathbf{x}') \right\|_1$$

- ならば、データベースごとに大きさの異なる摂動を入れたら良い気がしてくる
- そこで、大域的感度から類推して、「局所的感度」を考えてみることにする

$$\mathbf{LS}_f(\mathbf{x}) = \max_{\mathbf{x}':\mathbf{dist}_H(\mathbf{x},\mathbf{x}')=1} \| f(\mathbf{x}) - f(\mathbf{x}') \|_1$$

  - 各 $\mathbf{x}$ について定義されていることに注意
- 局所的感度を使って、$\mathbf{x}$ に対して摂動 $\mathrm{Lap}(\mathbf{LS}(\mathbf{x})/\varepsilon)$ を入れれば、差分プライベート化可能だろうか？ → 残念ながらNO
  - $\mathbf{LS}_{\mathrm{med}}(\mathbf{x}) = \max\,(x_{m+1} - x_m,\, x_m - x_{m-1})$：$m$ はちょうど真ん中のインデックス
  - 中央値の例では、$\mathbf{LS}_{\mathrm{med}}(\mathbf{x})=0$（つまり分散0）となる $\mathbf{x}$ がいくらでも作れる

大域的感度が大きいときの対処法（成功版）：
データごとの摂動を入れるために「平滑化感度」を定義する → うまくいく

- 局所的感度ではうまくいかないので、次の「平滑化感度」を考えてみる（遠くまで見る）

$$\mathbf{S}^*_f(\mathbf{x}) = \max_{\mathbf{x}'\in\mathcal{D}^n} \left(\underbrace{\mathbf{LS}_f(\mathbf{x}')}_{\text{局所的感度}} \cdot \underbrace{e^{-\varepsilon\cdot\mathbf{dist}_H(\mathbf{x},\mathbf{x}')}}_{\text{距離とともに減衰}} \right)$$

- 定理：$\mathrm{Lap}(\mathbf{S}^*_f(\mathbf{x})/\varepsilon)$ による摂動は、f を ε-差分プライベート化する
- 平滑化感度は、局所感度と大局的感度の間くらいのイメージ
  - $\mathbf{LS}(\mathbf{x}) \leqq \mathbf{S}^*_f(\mathbf{x}) \leqq \mathbf{GS}(\mathbf{x})$
- いかにも計算がしにくそう…

$$\mathbf{LS}_f(\mathbf{x}) = \max_{\mathbf{x}':\mathbf{dist}_H(\mathbf{x},\mathbf{x}')=1} \| f(\mathbf{x}) - f(\mathbf{x}') \|_1$$

平滑化感度の具体例：
平滑化感度の計算は自明ではないが、いくつかの場合でうまく計算できる

- 平滑化感度の計算

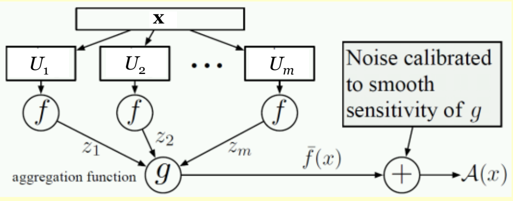
$$\mathbf{S}^*_f(\mathbf{x}) = \max_{\mathbf{x}'\in\mathcal{D}^n} \left(\mathbf{LS}_f(\mathbf{x}') \cdot e^{-\varepsilon\cdot\mathbf{dist}_H(\mathbf{x},\mathbf{x}')} \right)$$

- 平滑化感度の計算は簡単ではないが、例えば、中央値の場合、比較的簡単に計算できる
  - $\mathbf{x}$からハミング距離 k 離れた$\mathbf{x}'$の$\mathbf{LS}(\mathbf{x}')$ のなかで最大のもの$\max_{\mathbf{x}'} \mathbf{LS}(\mathbf{x}')$を計算してみる
    - k個の要素を変えることで、中央値をk個まで右か左にずらせる
      - 中央値は、x_{m-k}から x_{m+k} の間まで持っていける
    - $\mathbf{LS}_{\mathrm{med}}(\mathbf{x}) = \max\,(x_{m+1} - x_m,\, x_m - x_{m-1})$ を思い出すと、$\max_{\mathbf{x}'} \mathbf{LS}_{\mathrm{med}}(\mathbf{x})$ は、$\max\,(x_{m+1+k} - x_m, \ldots, x_m - x_{m-1-k})$
      - 僕はこれをなんとなく理解するのに3日かかりました
  - $\max_{\mathbf{x}'} \mathrm{LS}(\mathbf{x}')$ を全てのk（$\leqq n$）について計算すればOK
- ほか、最小全域木（Minimum Spanning Tree）のコストなども計算できる

平滑化感度が計算できない時の方法：
標本抽出と集約による平滑化感度の近似

不誠実

- 一般の f では必ずしも平滑化感度が計算できるとは限らない
- 「標本中抽出と集約」の枠組み
  - $\mathbf{x}$ から抽出したランダムな部分集合 $\{U_1, U_2, \ldots\}$ に対し f の値を計算し、これらから関数 g（平滑化感度が計算できる、たとえば平均）によって f の近似値 $\bar{f}(x)$ を求める
- 何がよいか？ → 話をgの平滑化感度にすり替える（イメージ）



Nissim et al. "Smooth Sensitivity and Sampling in Private Data Analysis" in STOC'07, modified by HK

サマリー：この章では、問い合わせに対するデータベースからの回答（出力）に外乱を加えることで、データのプライバシーを守る方法を学びました

- 良く似た別のDBの回答と区別できないならば“相対的に安全”であるとする「差分プライバシー」の概念を考えた
  - ユーザ（攻撃者）の知識に前提を置かないところがポイント
- 出力にどのような摂動を加えれば差分プライバシーが達成されるかを考えた
  - 答え：関数の取りうる値の幅に応じたラプラス分布を使えばよい
  - 大域的感度：隣り合ったDBに対する関数の差の最大値に応じた摂動を入れる
  - 平滑化感度：DBごとに異なった大きさの摂動を入れる

- 研究としては何ができるだろうか？
  - いろいろな関数に対する平滑化感度の計算：分散アルゴリズムをプライベート化するのと同じ路線