ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

CÁC MÔ HÌNH NHIỄU TRONG BẢO VỆ TÍNH RIÊNG TƯ

Môn học: Toán ứng dụng và thống kê

Danh sách thành viên:

Nguyễn Minh Ký - 18120048

Đoàn Văn Thanh An - 18120109

Nguyễn Hoàng Thái Dương - 18120336

Nguyễn Lê Trọng Đạt – 18120307

Mục lục

I. Giới thiệu

Các kỹ thuật **bảo mật dữ liệu (data privacy)** có mục tiêu cho phép các nhà phân tích tìm hiểu về các *xu hướng (trends)* trong dữ liệu nhạy cảm, mà không tiết lộ thông tin cụ thể cho *từng cá nhân (individuals).*

Differential privacy là kỹ thuật cung cấp khả năng bảo vệ quyền riêng tư mạnh mẽ. Các phương pháp tiếp cận thường được sử dụng trong nhiều thập kỷ gần đây như de-identification (khử nhận dạng) và aggregation (tổng hợp) đã được chứng minh là bị phá vỡ bởi các cuộc tấn công một cách tinh vi và thậm chí là các kỹ thuật hiện đại hơn như k-Anonymity (k - ẩn danh) cũng dễ bị tấn công. Vì vậy Differential privacy đang nhanh chóng trở thành "tiêu chuẩn vàng" trong việc bảo vệ quyền riêng tư.

II. Một số phương pháp bảo mật thông tin và cách tấn công

1. De-identification (Khử nhận dạng)

Là quá trình xóa *thông tin nhận dạng* (*identifying information*) khỏi tập dữ liệu

Thông tin nhận dạng không có định nghĩa chính thức. Thường được hiểu là thông tin sẽ được sử dụng xác định danh tính của chúng ta trong cuộc sáng thường ngày (tên, địa chỉ, số điện thoại, email, …

Để khử nhận dạng chỉ cần xóa các cột chứa thông tin nhận dạng.

2. Linking attacks (Tấn công liên kết)

Để thực hiện một cuộc tấn công liên kết đơn giản, chúng ta xem xét các cột khớp nhau giữa tập dữ liệu mà chúng ta đang cố gắng tấn công và dữ liệu phụ trợ mà chúng ta biết.

Các cuộc tấn công kiểu này có hiệu quả đáng ngạc nhiên:

- Chỉ một điểm dữ liệu duy nhất là đủ để thu hẹp mọi thứ xuống một vài bản ghi
- Tập hợp các bản ghi được thu hẹp giúp đề xuất thêm dữ liệu bổ trợ có thể hữu ích.
- Hai điểm dữ liệu thường đủ tốt để xác định lại một phần lớn dân số trong một tập dữ liệu cụ thể.

- Ba điểm dữ liệu (giới tính, mã ZIP, ngày sinh) xác định duy nhất 87% người dân ở Hoa Kỳ.

3. Aggregate (Tổng hợp)

Một cách khác để ngăn chặn tiết lộ thông tin cá nhân là chỉ công bố thông tin tổng hợp (*aggregate).* Ví dụ: số tuổi trung bình, mức lương trung bình, ...

Trong nhiều trường hợp, thống kê tổng hợp được chia thành các nhóm nhỏ hơn. Ví dụ: ta muốn biết mức lương trung bình của những người có trình độ học vấn cụ thể, …

Tổng hợp được cho là để cải thiện quyền riêng tư vì thật khó để xác định đóng góp của một cá nhân cụ thể vào thống kê tổng hợp. Nhưng dễ dàng bị lộ thông tin nếu nhóm quá nhỏ (ví dụ chỉ có 1 người).

4. Differencing Attacks

Ta có thể suy ra số tuổi trong thống kê tổng hợp của 1 người nào đó bằng việc trừ tổng số tuổi của nhóm cho tổng số tuổi của nhóm mà loại trừ người đó.

Các vấn đề:

- Việc tiết lộ thông tin hữu ích khiến việc đảm bảo quyền riêng tư trở nên rất khó khăn.
- Không thể phân biệt giữa truy vấn độc hại và không độc hại.

5. k-Anonymity (k-ẩn danh)

k-Anonymity là một thuộc tính của dữ liệu, đảm bảo rằng mỗi cá nhân “hòa nhập" với một nhóm ít nhất k người.

Đạt được sự ẩn danh bằng cách sửa đổi tập dữ liệu bằng cách tổng quát hóa nó, để các giá trị cụ thể trở nên phổ biến hơn và các nhóm dễ hình thành hơn.

Để đạt được k-ẩn danh, ta cần loại bỏ khá nhiều thông tin khỏi dữ liệu.

Tổng quát hóa tối ưu là cực kì khó và các ngoại lệ (outliers) có thể khiến nó thậm chí còn khó khăn hơn. Tổng quát hóa tối ưu cho k-Anonymity là NP-hard.

Để giải quyết vấn đề các yếu tố ngoại lệ, ta có thể loại bỏ hoàn toàn các trường hợp ngoại lệ. Điều này có thể ảnh hưởng đến độ chính xác, vì nó thay đổi thông tin từ thật thành giả nhưng giúp khái quát hóa nhiều dữ liệu hơn.

III. Differential Privacy

1. Định nghĩa

Không giống như k-Anonymity, Differential Privacy là một thuộc tính của thuật toán chứ không phải thuộc tính của dữ liệu.

Chúng ta có thể chứng minh dữ liệu đáp ứng differential privacy bằng cách chỉ ra rằng thuật toán tạo ra nó đáp ứng differential privacy.

Một hàm đáp ứng differential privacy thường được gọi là mechanism cơ chế. Chúng ta nói rằng một cơ chế F thỏa mãn differential privacy nếu tất cả các tập dữ liệu lân cận x và x' và tất cả các kết quả đầu ra có thể có S thỏa:

$$\frac{Pr[F(x) = S]}{Pr[F(x') = S]} \leq e^{\epsilon}$$

Hai tập dữ liệu được coi là lân cận nếu chúng khác nhau về dữ liệu của 1 cá nhân. F thường là 1 hàm ngẫu nhiên, do đó phân phối xác suất mô tả các kết quả đầu ra của nó không chỉ là một phân phối điểm.

Tính ngẫu nhiên được tích hợp trong F phải là "đủ" để một output quan sát được từ F sẽ không tiết lộ x hay x' là đầu vào.

Tham số ε được gọi là tham số bảo mật. Giá trịεcàng nhỏ thì cung cấp mức độ riêng tư càng cao. Thông thường εnằm trong khoảng $[1,10]$(quy ước chung).

2. Các đặc tính của Differential Privacy

- Tính chất nối tiếp: Nếu F1 thỏa $\epsilon 1$-Differential Privacy và F2 thỏa $\epsilon 1$-Differential Privacy và G = (F1, F2) thì G sẽ thỏa $(\epsilon 1 + \epsilon 2)$-Differential Privacy. Tính chất này rất quan trọng vì từ đó ta có thể đảm bảo được độ bảo mật khi khi thực hiện nhiều hàm thống kê trên 1 dataset.

- Tính chất song song: Nếu ta chia một dataset ra làm nhiều phần nhỏ và thực hiện 1 hàm f thỏa ϵ-Differential Privacy thì toàn bộ quá trình này cũng sẽ thỏa ϵ-Differential Privacy. Tính chất này quan trọng khi ta phân tích dữ liệu thành 1 histogram.

IV. Cơ chế Laplace

1. Định nghĩa

Cơ chế Laplace, như tên gọi của nó, sử dụng phân phối Laplace để cộng một lượng noise vào kết quả của mỗi query.

Ta có hàm mật độ xác suất của phân phối Laplace(μ = 0, b) như sau:

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

Gọi $f: \mathrm{X}^n \rightarrow \mathbb{R}^k$

Δ là độ nhạy cảm của f.

Cơ chế Laplace được định nghĩa như sau:

$$M(X) = f(X) + Y(Y_1, Y_2, \dots, Y_k)$$

với Y_i là các biến ngẫu nhiên độc lập tuân theo phân phối Laplace $\left(\frac{\Delta}{\varepsilon}\right)$

Cơ chế này thỏa ε-differential privacy.

Cách hoạt động của cơ chế Laplace là cộng vào kết quả một giá trị noise theo phân phối Laplace. Giá trị này phụ thuộc vào độ nhạy cảm của từng query.

Cơ chế Laplace là cơ chế cơ bản nhất và rất quan trọng trong diffenrential privacy.

2. Chứng minh cơ chế Laplace thỏa ε-differential privacy

Gọi X và Y là hai database hàng xóm (neighboring databases), khác nhau chỉ một đơn vị.

Ta gọi $p_x(z)$ và $p_y(z)$ lần lượt là hàm mật độ xác xuất của M(X) và M(Y) tại một điểm $z \in \mathbb{R}^k$

Ta có:

$$\frac{p_X(z)}{p_Y(\mathrm{z})} = \frac{\exp\left(-\frac{\varepsilon|f(X)-z|}{\Delta}\right)}{\exp\left(-\frac{\varepsilon|f(Y)-z|}{\Delta}\right)}$$

$$= \frac{\prod_{i=1}^{k}\exp\left(-\frac{\varepsilon|f(X)_i-z_i|}{\Delta}\right)}{\prod_{i=1}^{k}\exp\left(-\frac{\varepsilon|f(Y)_i-z_i|}{\Delta}\right)}$$

$$= \prod_{i=1}^{k}\exp\left(\frac{\varepsilon}{\Delta}(|f(Y)_i-z_i|-|f(X)_i-z_i|)\right)$$

$$\leq \prod_{i=1}^{k}\exp\left(\frac{\varepsilon}{\Delta}|f(X)_i-f(Y)_i|\right)$$

$$= \exp\left(\frac{\varepsilon}{\Delta}\prod_{i=1}^{k}|f(X)_i-f(Y)_i|\right)$$

$$= \exp\left(\frac{\varepsilon}{\Delta}\|f(X)-f(Y)\|_1\right)$$

$$\leq \exp(\varepsilon)\ \blacksquare$$

Vậy cơ chế Laplace ε-differentially private.

3. Độ chính xác của cơ chế Laplace

Độ lỗi ở đây được định nghĩa là sự chênh lệnh của kết quả lúc chỉ áp dụng query thường và khi áp dụng query với cơ chế Laplace.

Để tính được độ lỗi thì trước tiên ta tính, tail bound của phân bổ Laplace.

Ta có, với biến ngẫu nhiên $Y \sim Laplace(b)$, thì:

$$\Pr(|Y| \geq tb) = \exp(-t)$$

Gọi $f: \mathrm{X}^n \rightarrow \mathbb{R}^k$ có độ nhạy cảm ε và $p = Lap(X, f, \varepsilon) = f(x) + \mathrm{Y}\left(\frac{\Delta}{\varepsilon}\right)$ với X là một database bất kỳ, Lap là cơ chế Laplace thỏa ε-differential privacy, thì:

$$\Pr\left(|f(X) - p| \geq \left(\frac{\Delta}{\varepsilon}\right)\ln\left(\frac{1}{\beta}\right)\right) = \Pr\left(|Y| \geq \left(\frac{\Delta}{\varepsilon}\right)\ln\left(\frac{1}{\beta}\right)\right)$$

Áp dụng công thức tail bound phía trên vào:

$$\Pr\left(|Y| \geq \left(\frac{\Delta}{\varepsilon}\right)\ln\left(\frac{1}{\beta}\right)\right) = \exp\left(-ln\left(\frac{1}{\beta}\right)\right) = \beta$$

Vậy:

$$\Pr\left(|Y| \geq \left(\frac{\Delta}{\varepsilon}\right)\ln\left(\frac{1}{\beta}\right)\right) = \beta$$

hay:

$$\Pr\left(|f(X) - p| \leq \left(\frac{\Delta}{\varepsilon}\right)\ln\left(\frac{1}{\beta}\right)\right) = 1 - \beta$$

Thử xét với ví dụ sau: Ta có dataset X khảo sát 10000 người xem họ có hút thuốt hay không ($f(X_i)=0$ hoặc $f(X_i)=1$). Query hỏi có bao nhiêu người hút thuốc. Và $p = Lap(X, f, \varepsilon)$

Như vậy:

- Độ nhạy cảm $\Delta = 1$
- Ta cho $\varepsilon = 1$
- Chọn $\beta=0.05$

Suy ra, với xác xuất 95% thì:

$$p - 20 \leq f(X) \leq p + 20$$

Ta thấy rằng giá trị $20 = \left(\frac{\Delta}{\varepsilon}\right)\ln\left(\frac{1}{\beta}\right)$ ở đây có Δ phụ thuộc vào bản chất của query, còn ε thì ta sẽ tự chọn. Chọn ε nhỏ thì độ bảo mật sẽ cao hơn nhưng bù lại kết quả của cơ chế Laplace sẽ sai lệch nhiều hơn.

Thông thường ta nên chọn ε có giá trị từ 0.1 đến 5.

V. Sensitivity

1. Định nghĩa

Như chúng ta đã biết trong cơ chế Laplace, lượng nhiễu cần thiết để đảm bảo diffirential privacy cho một query phụ thuộc vào độ nhạy cảm (sensitivity) của query đó. Độ nhảy cảm sẽ phản ánh: sự ảnh hưởng đến output khi input thay đổi. Ta có cơ chế Laplace được định nghĩa như sau:

$$F(x) = f(x) + Lap(\frac{S}{\epsilon})$$

trong đó f(x) là hàm xác định (query), ϵ là tham số privacy, s là sensitivity của f.

Cho một hàm $f: D \rightarrow R$, trong đó ánh xạ tập dữ liệu (D) vào số thực, độ nhạy cảm toàn bộ (global sensitivity) của f được định nghĩa là:

$$GS(f) = \max_{x,x':d(x,x')\leq 1} |f(x) - f(x')|$$

Ở đây, $d(x, x')$ thể hiện khoảng cách giữa 2 tập dữ liệu x và x', và chúng ta nói 2 tập dữ liệu là họ hang (neighbors) nếu khoảng cách giữa chúng bé hơn hoặc bằng 1.

Định nghĩa này có thể hiểu là: Global sensitivity của cặp tập dữ liệu họ hang x và x', khoảng cách của $f(x)$ và $f(x')$ tối đa bằng $GS(f)$.

2. Tính Sensitivity

Đối với các hàm f trên tập số thực, độ nhạy cảm là dễ dàng tính được.
Đối với những hàm ánh xạ tập dữ liệu vào số thực, ta có thể dùng một số phương pháp phân tích. Cụ thể ta sẽ xét đến các query tổng hợp dữ liệu: counts, sum, averages.

a) Counting Queries:

Counting queries luôn có độ nhạy cảm bằng 1.

b) Summation Queries:

Độ nhạy cảm của các query này không đơn giản như counting query. Ví dụ: một hàm query: Tổng số tuổi của một những người có educational-num lớn hơn 10.

```
adult[adult['educational-num'] > 10].shape[0]
15772
```

Điều này đồng nghĩa với việc, khi ta thêm/bớt một dòng dữ liệu, sẽ ảnh hưởng kết quả bằng dữ liệu tuổi của người đó. Độ nhạy cảm sẽ phụ thuộc vào nội dung mà chúng ta thêm/bớt.

Chúng ta có thể chọn một số cố định để biểu diễn sensitivity. Ví dụ ta chọn sensivity bằng 125, vì không ai lớn hơn 125 tuổi. Tuy nhiên, không điều gì có thể đảm bảo được điều này. Ta gọi summation queries có unbound sensitivity, vì ta không thể xác định được chặn trên (upper bound) và chặn dưới (lower bound) của dữ liệu này. Để giải quyết vấn đề này, người ta sử dụng kỹ thuật clipping.

c) Average Queries (Trung bình/ kỳ vọng):

Cách đơn giản để trả lời average query là chia việc tính trung bình thành 2 giai đoạn: summation query và counting query. Với mỗi query ta sẽ tính toán như mô tả ở trên. Kết quả sau 2 query sẽ chia nhau để được kết quả của average query.

```
adult[adult['educational-num'] > 10]['age'].sum() / adult[adult['educational-num'] >
10]['age'].shape[0]
40.267562769464874
```

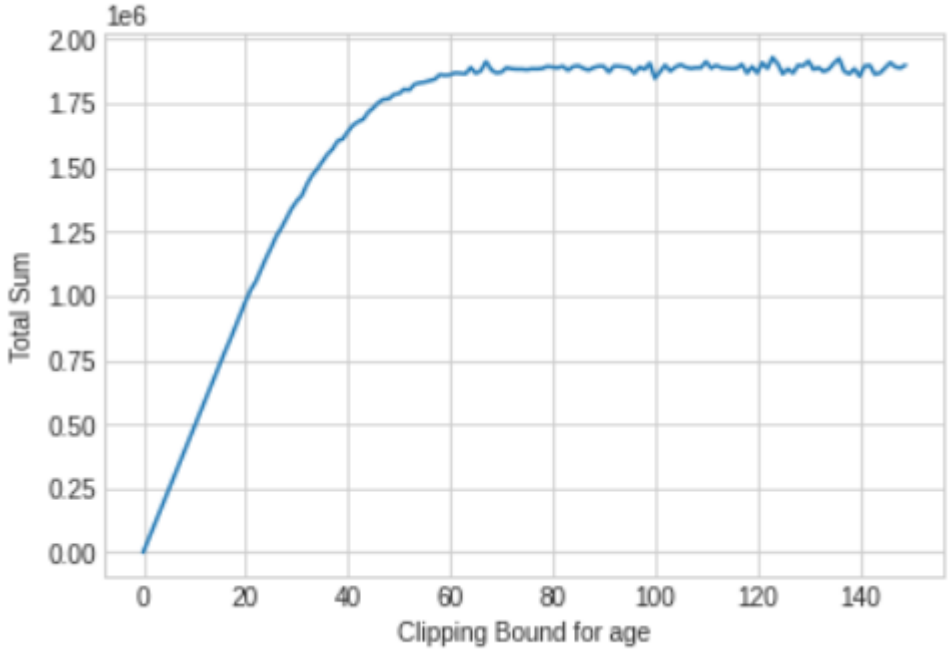
3. Clipping:

Ý tưởng của clipping là "chặn" cho upperbound và lowerbound vào giá trị xác định. Ví dụ những người nào có tuổi trên 125 sẽ được cắt thành đúng 125. Và độ nhạy cảm của hàm sum lúc này sẽ bằng upperbound – lowerbound.

Ta có thể chọn cách nhìn vào dữ liệu và chọn ra upperbounds. Ví dụ trong tập dữ liệu adult.csv. Không có dữ liệu nào có tuổi trên 90. Tuy nhiên, ta không thể chọn upperbound

bằng 90. Vì như vậy, ta đã vi phạm vào nguyên tắc của differential privacy, đó là làm lộ phần thông tin của tập dữ liệu.

Để chọn được upperbound và lowerbound, ví dụ trong trường hợp cho dữ liệu tuổi. Ta chọn lowerbound bằng 0, và từ từ tăng dần upperbound cho đến khi kết quả của query ngừng thay đổi. Ví dụ, ta tính tổng số tuổi của clipping bound từ 0 đến 150, sử dụng cơ chế



Laplace.

Độ mất mát bảo mật (privacy cost) cho để xây dựng đồ thị này là $\epsilon = 1.5$ với tính chất nối tiếp (sequential composition), vì ta thực hiện 150 query với $\epsilon = 0.01$. Chúng ta có thể chọn upper = 90.

Dựa vào đồ thị trên, ta sẽ chọn upperbound như thế nào? Một cách chọn khá tốt là tìm vùng của đồ thị đủ "trơn" (độ nhiễu thấp) và không tăng lên (clipping bound đủ tốt).

VI. Tài liệu tham khảo

1. Additive noise mechanisms, Wikipedia

2. Intro to Differential Privacy, Part 2, Lecture Note, Gautam Kamath

3. Programming Differential Privacy, Joseph P. Near and Chiké Abuah