Data Anonymization

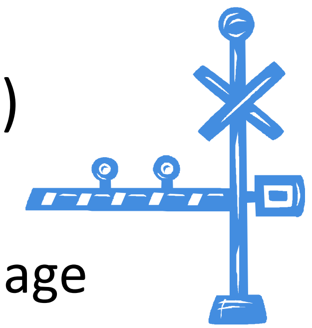
Graham Cormode

graham@research.att.com

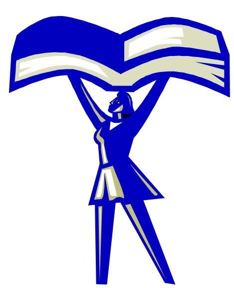
Why Anonymize?

- For Data Sharing
  - Give real(istic) data to others to study without compromising privacy of individuals in the data
  - Allows third-parties to try new analysis and mining techniques not thought of by the data owner
- For Data Retention and Usage
  - Various requirements prevent companies from retaining customer information indefinitely
  - E.g. Google progressively anonymizes IP addresses in search logs
  - Internal sharing across departments (e.g. billing $\rightarrow$ marketing)

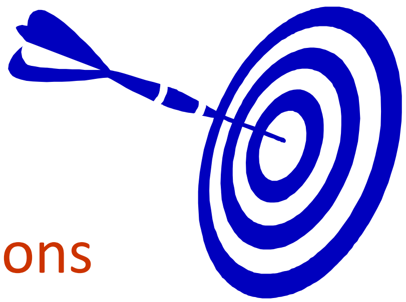
Models of Anonymization



- Interactive Model (akin to statistical databases)
  - Data owner acts as "gatekeeper" to data
  - Researchers pose queries in some agreed language
  - Gatekeeper gives an (anonymized) answer, or refuses to answer
- "Send me your code" model
  - Data owner executes code on their system and reports result
  - Cannot be sure that the code is not malicious, compiles…
- Offline, aka "publish and be damned" model
  - Data owner somehow anonymizes data set
  - Publishes the results, and retires
  - Seems to best model many real releases

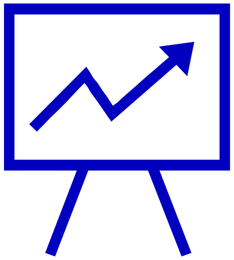


Objectives for Anonymization



- Prevent (high confidence) inference of associations
  - Prevent inference of salary for an individual in census data
  - Prevent inference of individual’s video viewing history
  - Prevent inference of individual’s search history in search logs
  - All aim to prevent linking sensitive information to an individual
- Have to model what knowledge might be known to attacker
  - Background knowledge: facts about the data set (X has salary Y)
  - Domain knowledge: broad properties of data (illness Z rare in men)

Utility



- Anonymization is meaningless if utility of data not considered
  - The empty data set has perfect privacy, but no utility
  - The original data has full utility, but no privacy
- What is “utility”? Depends what the application is…
  - For fixed query set, can look at max, average distortion
  - Problem for publishing: want to support unknown applications!
  - Need some way to quantify utility of alternate anonymizations

Part I: Syntactic Anonymizations

- "Syntactic anonymization" modifies the input data set
  - To achieve some 'syntactic property' intended to make reidentification difficult
  - Many variations have been proposed:
    - k-anonymity
    - l-diversity
    - t-closeness
    - … and many many more

Tabular Data Example

- Census data recording incomes and demographics

SSN	DOB	Sex	ZIP	Salary
11-1-111	1/21/76	M	53715	50,000
22-2-222	4/13/86	F	53715	55,000
33-3-333	2/28/76	M	53703	60,000
44-4-444	1/21/76	M	53703	65,000
55-5-555	4/13/86	F	53706	70,000
66-6-666	2/28/76	F	53706	75,000

- Releasing SSN $\rightarrow$ Salary association violates individual's privacy
  - SSN is an identifier, Salary is a sensitive attribute (SA)

Tabular Data Example: De-Identification

- Census data: remove SSN to create de-identified table

DOB	Sex	ZIP	Salary
1/21/76	M	53715	50,000
4/13/86	F	53715	55,000
2/28/76	M	53703	60,000
1/21/76	M	53703	65,000
4/13/86	F	53706	70,000
2/28/76	F	53706	75,000

- Does the de-identified table preserve an individual's privacy?
  - Depends on what other information an attacker knows

Tabular Data Example: Linking Attack

- De-identified private data + publicly available data

DOB	Sex	ZIP	Salary
1/21/76	M	53715	50,000
4/13/86	F	53715	55,000
2/28/76	M	53703	60,000
1/21/76	M	53703	65,000
4/13/86	F	53706	70,000
2/28/76	F	53706	75,000

SSN	DOB
11-1-111	1/21/76
33-3-333	2/28/76

- Cannot uniquely identify either individual's salary
  - DOB is a quasi-identifier (QI)

Tabular Data Example: Linking Attack

- De-identified private data + publicly available data

DOB	Sex	ZIP	Salary
1/21/76	M	53715	50,000
4/13/86	F	53715	55,000
2/28/76	M	53703	60,000
1/21/76	M	53703	65,000
4/13/86	F	53706	70,000
2/28/76	F	53706	75,000

SSN	DOB	Sex	ZIP
11-1-111	1/21/76	M	53715
33-3-333	2/28/76	M	53703

- Uniquely identified both individuals' salaries
  - [DOB, Sex, ZIP] is unique for majority of US residents [Sweeney 02]

Tabular Data Example: Anonymization

- Anonymization through QI attribute generalization

DOB	Sex	ZIP	Salary
1/21/76	M	537**	50,000
4/13/86	F	537**	55,000
2/28/76	*	537**	60,000
1/21/76	M	537**	65,000
4/13/86	F	537**	70,000
2/28/76	*	537**	75,000

SSN	DOB	Sex	ZIP
11-1-111	1/21/76	M	53715
33-3-333	2/28/76	M	53703

- Cannot uniquely identify tuple with knowledge of QI values
  - E.g., ZIP = 537** → ZIP $\in$ {53700, ..., 53799}

Tabular Data Example: Anonymization

- Anonymization through sensitive attribute (SA) permutation

DOB	Sex	ZIP	Salary
1/21/76	M	53715	55,000
4/13/86	F	53715	50,000
2/28/76	M	53703	60,000
1/21/76	M	53703	65,000
4/13/86	F	53706	75,000
2/28/76	F	53706	70,000

SSN	DOB	Sex	ZIP
11-1-111	1/21/76	M	53715
33-3-333	2/28/76	M	53703

- Can uniquely identify tuple, but uncertainty about SA value
  - Much more precise form of uncertainty than generalization

k-Anonymization [Samarati, Sweeney 98]

- k-anonymity: Table T satisfies k-anonymity wrt quasi-identifiers QI iff each tuple in (the multiset) T[QI] appears at least k times
  - Protects against “linking attack”
- k-anonymization: Table T’ is a k-anonymization of T if T’ is generated from T, and T’ satisfies k-anonymity

DOB	Sex	ZIP	Salary
1/21/76	M	53715	50,000
4/13/86	F	53715	55,000
2/28/76	M	53703	60,000
1/21/76	M	53703	65,000
4/13/86	F	53706	70,000
2/28/76	F	53706	75,000

→

DOB	Sex	ZIP	Salary
1/21/76	M	537**	50,000
4/13/86	F	537**	55,000
2/28/76	*	537**	60,000
1/21/76	M	537**	65,000
4/13/86	F	537**	70,000
2/28/76	*	537**	75,000

Homogeneity Attack [Machanavajjhala+ 06]

- Issue: k-anonymity requires each tuple in (the multiset) T[QI] to appear $\geq$ k times, but does not say anything about the SA values
  - If (almost) all SA values in a QI group are equal, loss of privacy!
  - The problem is with the choice of grouping, not the data
  - For some groupings, no loss of privacy

DOB	Sex	ZIP	Salary
1/21/76	M	53715	50,000
4/13/86	F	53715	55,000
2/28/76	M	53703	60,000
1/21/76	M	53703	50,000
4/13/86	F	53706	55,000
2/28/76	F	53706	60,000

Ok! $\longrightarrow$

DOB	Sex	ZIP	Salary
76-86	*	53715	50,000
76-86	*	53715	55,000
76-86	*	53703	60,000
76-86	*	53703	50,000
76-86	*	53706	55,000
76-86	*	53706	60,000

l-Diversity [Machanavajjhala+ 06]

- Intuition: Most frequent value does not appear too often compared to the less frequent values in a QI group
- Simplified *l*-diversity defn: for each group, max frequency $\leq 1/l$
  - *l*-diversity((1/21/76, *, 537**)) = 1

DOB	Sex	ZIP	Salary
1/21/76	*	537**	50,000
4/13/86	*	537**	55,000
2/28/76	*	537**	60,000
1/21/76	*	537**	50,000
4/13/86	*	537**	55,000
2/28/76	*	537**	60,000

Simple Algorithm for *l*-diversity

- A simple greedy algorithm provides *l*-diversity”
  - Sort tuples based on attributes so similar tuples are close
  - Start with group containing just first tuple
  - Keeping adding tuples to group in order until l-diversity met
  - Output the group, and repeat on remaining tuples

DOB	Sex	ZIP	Salary
1/21/76	M	53715	50,000
4/13/86	F	53715	50,000
2/28/76	M	53703	60,000
1/21/76	M	53703	65,000
4/13/86	F	53706	50,000
2/28/76	F	53706	60,000

2-diversity →

DOB	Sex	ZIP	Salary
1/21/76	M	53715	50,000
4/13/86	F	53715	50,000
2/28/76	M	53703	60,000
1/21/76	M	53703	65,000
4/13/86	F	53706	50,000
2/28/76	F	53706	60,000

  - Knowledge of the algorithm used can reveal associations!

Syntactic Anonymization Summary

- Pros:
  - Provide natural definitions (e.g. k-anonymity)
  - Keeps data in similar form to input (e.g. as tuples)
  - Give privacy beyond simply removing identifiers
- Cons:
  - No strong guarantees known against arbitrary adversaries
  - Resulting data not always convenient to work with
  - Attack and patching has led to a glut of definitions

Part 2: Differential Privacy

A randomized algorithm K satisfies ε-differential privacy if:

Given any pair of "neighboring" data sets, D and D', and any property S:

$$\Pr[\, K(D) \in S] \leq e^{\varepsilon} \Pr[\, K(D') \in S]$$

Introduced by Cynthia Dwork, Frank McSherry, Kobbi Nissim, Adam Smith in 2006

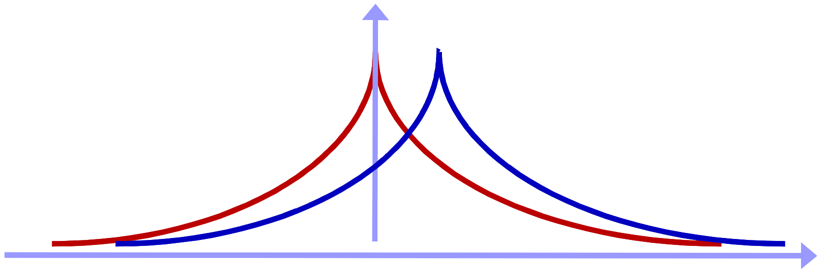
Differential Privacy for numeric functions

- Sensitivity of publishing for a numeric function f:

  $s = \max_{X,X'} |f(X) - f(X')|$, X, X' differ by 1 individual

To give ε-differential privacy for a function with sensitivity s:

- Add Laplace noise, Lap(ε/s) to the true output answer



Laplace Distribution

- Laplace with parameter λ is exponential, symmetric about 0:
  - Density at x is $f(x) \propto \exp(-|x|/\lambda)$
- Hence, $f(x)/f(x+\delta) = \exp(-|x|/\lambda)/\exp(-|x+\delta|/\lambda) \leq \exp(\delta/\lambda)$
- Differential privacy for numeric values:
  - Sensitivity = s
  - Hence, $\delta=s$
  - Set $\lambda = \varepsilon/s$
  - Ratio of probability at any point x is at most $\exp(\varepsilon)$

Sensitivity of some functions

- "Count" has sensitivity 1
  - E.g. count how many students are left-handed
- Sum and median have sensitivity Δ
  - Δ = maximum range of possible values
- Histograms / contingency tables have sensitivity 2
  - E.g. Count how many people in salary range 0-50K; 50-100K; 100-150K; 150-200K; 200K+

Dealing with sensitivity

- Sometimes sensitivity (and hence noise) can be very high:
  - Sensitivity of (sum of salaries) ~ $1BN (some people make this much)
  - Replace with clipped value (e.g. cut off at $1M)
  - Work with histograms/contingency tables instead

Contingency Tables

Zip	0-50K	50-100K	100-150K	150K+
53703	200	11	10	5
53706	18	5	65	200
53715	60	100	100	40

Noisy Contingency Tables

Zip	0-50K	50-100K	100-150K	150K+
53703	205	8	9	7
53706	19	8	66	201
53715	59	97	98	40

Does this provide sufficient privacy?

Exponential Mechanism

- Exponential mechanism gives more general way to release functions
- Given input x, define a "quality" function $q_x(y)$ over possible outputs that captures desirability of outputting y
  - $q(y) = 0$ means perfect match; larger q values less desirable
- Define s = sensitivity of function q
- Output y with probability proportional to $\exp(-\varepsilon\, q_x(y))$
  - Claim (without proof): process has (εs)–differential privacy
  - Note: must range over all possible outputs for correctness
    - May be very slow to compute if many possible outputs

Exponential Mechanism for Median

- Given input X = set of n elements in range $\{0 \ldots U\}$
- Define $rank(x)$ = number of elements less than x
  - Median: x s.t. $rank(x) = n/2$
- Set $q(y) = |rank(y) - n/2|$
  - Sensitivity of rank = 2
- Use exponential mechanism with q:
  - Elements in range $[x_j \ldots x_{j+1}]$ have same rank, so same q value
  - Compute probability of $[x_j \ldots x_{j+1}]$ as $(x_{j+1} - x_j) \cdot \exp(-\varepsilon|rank(x_j) - n/2|)$
  - Then pick element uniformly from range $x_j \ldots x_{j+1}$
  - Median now takes time $O(n)$, not $O(U)$

at&t

State of Anonymization

- Data privacy and anonymization is a subject of ongoing research in 2011
- Many unresolved challenges:
  - How can a social network release a substantial data set without revealing private connections between users?
  - How can a video website release information on viewing patterns without disclosing who watched what?
  - How can a search engine release information on search queries without revealing who searched for what?
  - How to release private information efficiently over large scale data?

Concluding Remarks

- Like crypto, anonymization proceeds by proposing anonymization methods and attacks upon them
  - Difference: Successful attacks on crypto reveal messages
  - Attacks on anonymization increase probability of inference
- Long-term goal: propose anonymization methods which resist feasible attacks
  - Anonymization should not be the weakest link