Publicly Released Large Datasets

- Useful for improving recommendation systems, collaborative research

- Contain personal information

- Mechanisms to protect privacy, e.g. anonymization by deleting name identifiers, adding noise
Roadmap

- Motivation
- Privacy definitions
  - K anonymity
- Netflix-IMDb attack
  - Weighted scoring algorithm
  - Theoretical analysis
- Conclusion
Sanitization of Databases

Real Database (RDB)

Sanitized Database (SDB)

Add noise, delete names, etc.

Health records
Census data

Protect privacy
Provide useful information (utility)
Database Privacy

- Releasing sanitized databases
  1. k-anonymity [Samarati 2001; Sweeney 2002]
  2. Differential privacy [Dwork et al. 2006] (future lecture)
Re-identification by linking

Linking two sets of data on shared attributes may uniquely identify some individuals:

Example [Sweeney] : De-identified medical data was released, purchased Voter Registration List of MA, re-identified Governor 87 % of US population uniquely identifiable by 5-digit ZIP, sex, dob
1. K-anonymity

- Quasi-identifier: Set of attributes (e.g. ZIP, sex, dob) that can be linked with external data to uniquely identify individuals in the population
  - Issue: How do we know what attributes are quasi-identifiers?

- Make every record in the table indistinguishable from at least k-1 other records with respect to quasi-identifiers

- Linking on quasi-identifiers yields at least k records for each possible value of the quasi-identifier
K-anonymity and beyond

Provides some protection: linking on ZIP, age, nationality yields 4 records
Limitations: lack of diversity in sensitive attributes, background knowledge, subsequent releases on the same data set, syntactic definition
Utility: less suppression implies better utility

l-diversity, m-invariance, t-closeness, …
Re-identification Attacks in Practice

Examples:
- Netflix-IMDB
- Movielens attack
- Twitter-Flicker
- Recommendation systems – Amazon, Hunch,..

Goal of De-anonymization: To find as much information about an individual whose record is in the released dataset
Roadmap

- Motivation
- Privacy definitions
  - $k$ anonymity
- Netflix-IMDb attack
  - Weighted scoring algorithm
  - Theoretical analysis
- Conclusion
De-anonymization Attacks Still Possible

- **Isolation Attacks**
  - Recover individual’s record from anonymized database
  - E.g., find user’s record in anonymized Netflix movie database

- **Information Amplification Attacks**
  - Find more information about individual in anonymized database
  - E.g., find ratings for specific movie for user in Netflix database
Netlix-IMDb Empirical Attack

Anonymized Netflix DB

<table>
<thead>
<tr>
<th></th>
<th>Gladiator</th>
<th>Titanic</th>
<th>Heidi</th>
</tr>
</thead>
<tbody>
<tr>
<td>r₂</td>
<td>2</td>
<td>1.5</td>
<td>1</td>
</tr>
<tr>
<td>r₃</td>
<td>0.5</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Publicly available IMDb ratings (noisy)

<table>
<thead>
<tr>
<th></th>
<th>Titanic</th>
<th>Heidi</th>
</tr>
</thead>
</table>

Used as auxiliary information

Each row corresponds to an individual

Each column corresponds to an attribute, e.g. movie

Weighted Scoring Algorithm

Isolation Attack!

Problem Statement

Attacker uses algorithm to find record

**Attacker’s goal:** Find $r_1$ or record similar to Bob’s record

Explain why empirical de-anonymization attacks work
Our Contributions

- Formalize assumptions under which de-anonymization attacks work about database and auxiliary information.

<table>
<thead>
<tr>
<th></th>
<th>Gladiator</th>
<th>Titanic</th>
<th>Heidi</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r_1$</td>
<td>4</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$r_2$</td>
<td>2</td>
<td>1.5</td>
<td>1</td>
</tr>
</tbody>
</table>

- Prove theorems to quantify effectiveness of the Weighted Scoring Algorithm under these assumptions.

- Explain an empirical attack by testing assumptions over database for Netflix IMDb Empirical Attack.

Datta et al 2012, Provable De-anonymization of Large Datasets with Sparse Dimensions
How do you measure similarity of this record with Bob’s record? (Similarity Metric)

What does auxiliary information about a record mean?
Definition: Asymmetric Similarity Metric

**Individual Attribute Similarity**

\[
T(y(i), r(i)) = 1 - \frac{|y(i) - r(i)|}{p(i)}
\]

\[
T(y(v_1), r(v_1)) = 1 - \frac{|5 - 0|}{5} = 0
\]

\[p(i): \text{ range of attribute } i\]

**Similarity Metric**

\[
S(y, r) = \sum_{i \in \text{supp}(y)} \frac{T(y(i), r(i))}{|\text{supp}(y)|}
\]

\[\text{supp}(y): \text{ non null attributes in } y\]

<table>
<thead>
<tr>
<th>Movie (i)</th>
<th>T(y(i), r(i))</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gladiator</td>
<td>0</td>
</tr>
<tr>
<td>Titanic</td>
<td>0.6</td>
</tr>
<tr>
<td>Heidi</td>
<td>0</td>
</tr>
</tbody>
</table>

Intuition: Measures how closely two people’s ratings match on one movie

Intuition: Measures how closely two people’s ratings match overall

\[S(y, r) = \frac{0.6}{2} = 3\]
Definition: Auxiliary Information

Intuition:
- **aux** about **y** should be a subset of record **y**
- **aux** can be noisy

**aux** captures information available outside normal data release process

Bound level of perturbation in **aux**

\[ \gamma \in [0,1] \]

\((m, \gamma)\)-perturbed auxiliary information

\[ \forall i \in \text{supp}(aux). T(y(i), aux(i)) \geq 1 - \gamma \]

\[ |\text{supp}(aux)| = m = \text{no. of non null attributes in } aux \]

Intuition: The fewer the number of people who watched a movie, the rarer it is.

**Weight of an attribute** \( i \)

\[
|\text{supp}(i)| = \text{no. of non null entries in column } i
\]

Use weight as an indicator of rarity

Score gives a weighted average of how closely two people match on every movie, giving higher weight to rare movies.

**Scoring Methodology**

\[
\text{Score}(\text{aux}, r_j) = \sum_{i \in \text{supp(aux)}} \frac{T(\text{aux}(i), r_j(i))}{|\text{supp(aux)}|}
\]

\[
|\text{supp(aux)}| = m = \text{no. of non null attributes in } aux
\]

Compute \text{Score} for every record \( r \) in anonymized DB to find out which one is closest to target record \( y \).
Weighted Scoring Algorithm [Narayanan et al 2008]

Compute Score for every $r$ in $D$

\[
\begin{array}{ccc}
w_i & 0.63 & 0.5 & 0.63 \\
\text{Score}(aux, r_j) & 0.52 & 0.40 & 0.23 \\
r_2 & 3 & 1 & 4 \\
r_3 & - & 2 & 4
\end{array}
\]

Eccentricity measure $> \text{threshold}$

\[
e(aux, D) = \max_{r \in D} \left( \text{Score}(aux, r) \right) - \max_{2, r \in D} \left( \text{Score}(aux, r) \right)
\]

Output record with max Score

Score($aux, r$) used to predict $S(y, r)$
Theorem 1: When Isolation Attacks work?

Intuition: If eccentricity is high, algorithm always finds the record corresponding to auxiliary information!

If

\[ \text{aux is (} m, \gamma \text{)-perturbed} \]

\[ \text{Eccentricity threshold} > \gamma \mathcal{M} \]

then

\[ \text{Score}(\text{aux}, \mathcal{O}) = \text{Score}(\text{aux}, y) \]

\[ \gamma : \text{Indicator of perturbation in aux} \]
\[ \mathcal{M} : \text{Average of weights in aux} \]
\[ \mathcal{O} : \text{Record output by algorithm} \]
\[ y : \text{Target record} \]

If \( \mathcal{O} \) is the only record with the highest score then \( \mathcal{O} = y \)
Do Assumptions hold over Netflix Database?

% Records for which Theorem 1 assumptions hold

<table>
<thead>
<tr>
<th>Perturbation measure, gamma (γ)</th>
<th>% Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.07</td>
<td>95%</td>
</tr>
<tr>
<td>0.1</td>
<td>80%</td>
</tr>
<tr>
<td>0.15</td>
<td>50%</td>
</tr>
<tr>
<td>0.2</td>
<td>20%</td>
</tr>
</tbody>
</table>

- m = 10
- m = 20

m : no. of attributes in aux

Averaged over sample of 10000 records chosen with replacement
Conclusion

- We obtain **provable** bounds about, and **verify empirically**, why some de-anonymization attacks work in practice.

- We find that even perturbed auxiliary information can be used to launch de-anonymization attacks if:
  - *Database* has many **rare dimensions** and
  - *Auxiliary information* has information about these rare dimensions.
Questions?