Generating Private Synthetic Data:
Presentation 2

Zexi Song

Mentor: Dr. Anand Sarwate

DIMACS-REU

July 17, 2015
Overview

1. Project Overview (Revisited)
2. Sensitivity of Mutual Information
3. Simulation
4. Results with Real Data
5. Future Work
Recall the process

Our goal is to publish synthetic data: we learn patterns and correlations in the population in a privacy-preserving way and then generate fake individuals/records so that the population statistics are similar.

- Given a dataset, study the patterns and correlations in the population.
- Build a probabilistic graphical model.
- Generate synthetic data that yields similar population stats.

Preserve Privacy!
Chow-Liu Algorithm

- Variables of interest $X_1, X_2, \ldots, X_m$ with dataset of size $n$.
- Create a complete graph using $X_1, X_2, \ldots, X_m$ as vertices.
- For each pair of $X_i, X_j$ calculate the mutual information $I(X_i : X_j)$, then use mutual information as edge weights.
- Find the maximum spanning tree.
CLNode Python Class

\[ I\hat{p}(X_i; X_j) = \sum_{a,b} \hat{P}(X_i = a, X_j = b) \cdot \log \frac{\hat{P}(X_i = a, X_j = b)}{\hat{P}(X_i = a)\hat{P}(X_j = b)} \]

- CLNode.py
- Input: Range, Data
- Output: Empirical Distribution, Joint Empirical Distribution, Mutual Information
Differential Privacy

Let $M$ be a randomized data release algorithm. We say that $M$ is $\epsilon$—differentially private if for all $D_1, D_2$ that differ in only one element and any event $S$,

$$Pr(M(D_1) \in S) \leq e^{\epsilon} \times Pr(M(D_2) \in S)$$
Differential Privacy

Let $M$ be a randomized data release algorithm. We say that $M$ is $\epsilon$—differentially private if for all $D_1, D_2$ that differ in only one element and any event $S$,

$$\Pr(M(D_1) \in S) \leq e^{\epsilon} \times \Pr(M(D_2) \in S)$$

“Differential privacy ensures that any sequence of outputs (responses to queries) is essentially equally likely to occur, independent of the presence or absence of any individual.”
Sensitivity

Let $f : \mathcal{D} \rightarrow \mathbb{R}$ be a function. The sensitivity of $f$ is defined as,

$$S(f) = \max_{D, D' \in \mathcal{D}} |f(D) - f(D')|$$

where $D$ and $D'$ only differ by 1 element.
Let $X$, $Y$ be discrete random variables with dataset of size $n$ then

$$S_n(I(X; Y)) \leq 2\left(\frac{1+1/n}{2} \log\left(\frac{1+1/n}{2}\right) - \frac{1-1/n}{2} \log\left(\frac{1-1/n}{2}\right) - \frac{1}{n} \log\left(\frac{1}{n}\right)\right)$$
Sensitivity of Mutual Information

Let $X$, $Y$ be discrete random variables with dataset of size $n$ then

$$S_n(I(X; Y)) \leq 2\left(\frac{1+1/n}{2} \log\left(\frac{1+1/n}{2}\right) - \frac{1-1/n}{2} \log\left(\frac{1-1/n}{2}\right) - \frac{1}{n} \log\left(\frac{1}{n}\right)\right)$$

"worst case"
Sensitivity of Mutual Information

Upper Bound of the Sensitivity of Mutual Information
The Laplace Mechanism

Given any function $f : \mathcal{D} \to \mathbb{R}$ the Laplace mechanism is defined as:

$$\mathcal{M}_L(D, f(\cdot), \epsilon) = f(D) + Z$$

where $Z \sim \text{Lap}(S(f)/\epsilon)$

**Theorem:** The Laplace mechanism preserves $\epsilon-$differential privacy.
Doubly Symmetric Binary Source

Let $X_1, X_2$ be binary variables, if $p_{X_1,X_2}(x_1, x_2)$ is given by,

$$
\begin{pmatrix}
\frac{1-a}{2} & \frac{a}{2} \\
\frac{2-a}{2} & \frac{1-a}{2}
\end{pmatrix}
$$

then $X_1, X_2$ form a doubly symmetric binary source.

$X_1 \sim \text{Bernoulli}(0.5)$

$X_2 = (X_1 + Z_{12}) \mod 2$, where

$Z_{12} \sim \text{Bernoulli}(a)$
Simulation Results

No noise added, non-private
Simulation Results

No noise added, non-private

Noise added, $\epsilon = 0.3$
Simulation Results

Fraction of correctly recovering the tree
Simulation Results

Average number of edges missed

Simulation Results 2

Average number of edges missed with different values of epsilon and dataset sizes.
Adult Data Set

Sample Size: 32878 (after deleting entries with missing values)

9 Variables Selected:
Age, Workclass, Education,
Marital Status, Occupation, Race, Gender
Annual Income >50K, Working hours per week.
Results

No noise added, non-private
Results

No noise added, non-private

Noise added, $\epsilon = 1$, 100%
Results

No noise added, non-private

Noise added, $\epsilon = 0.75, 90\%$
Results

No noise added, non-private

Noise added, $\epsilon = 0.5$, 60%
Results

No noise added, non-private

Noise added $\epsilon = 0.25, 25\%$
Results

No noise added, non-private

Noise added, $\epsilon = 0.1$, 30%
## Retrospect

<table>
<thead>
<tr>
<th>Race</th>
<th>Age</th>
<th>Workclass</th>
<th>Education</th>
<th>Marital-Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.006694</td>
<td>0.007101</td>
<td>0.010186</td>
<td>0.012826</td>
</tr>
<tr>
<td>Occupation</td>
<td>Gender</td>
<td>Income$\geq$50K</td>
<td>Hours per Week</td>
<td></td>
</tr>
<tr>
<td>0.013167</td>
<td>0.006604</td>
<td>0.005839</td>
<td>0.014553</td>
<td></td>
</tr>
</tbody>
</table>

Mutual information btw Race and others
## Retrospect

<table>
<thead>
<tr>
<th>Race</th>
<th>Age</th>
<th>Workclass</th>
<th>Education</th>
<th>Marital-Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.006694</td>
<td>0.007101</td>
<td>0.010186</td>
<td>0.012826</td>
</tr>
<tr>
<td>Occupation</td>
<td>0.013167</td>
<td>0.006604</td>
<td>0.005839</td>
<td>0.014553</td>
</tr>
</tbody>
</table>

**Mutual information btw Race and others**

<table>
<thead>
<tr>
<th>Education</th>
<th>Age</th>
<th>Workclass</th>
<th>Marital-Status</th>
<th>Occupation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.106528</td>
<td>0.029758</td>
<td>0.021323</td>
<td>0.232357</td>
</tr>
<tr>
<td>Race</td>
<td>0.010186</td>
<td>0.004389</td>
<td>0.064521</td>
<td>0.059309</td>
</tr>
</tbody>
</table>

**Mutual information btw Education and others**
The Next Step

Generate Synthetic Data from the tree structure
- Use empirical conditional distributions
- Logistic regression

Measure the performance of data generated

Extend the scope into continuous data
Thank You