Synthetic Data Generation
Challenges and Techniques

Ju Fan (范举)
nanj@ruc.edu.cn
http://iir.ruc.edu.cn/~fanj/
2020.11.28
Talk Objectives

- Why do we need synthetic data?
- What are research challenges of synthetic data generation?
- How to solve synthetic data generation?
- What are future research directions?
Difficulties in Data Access

- Top barriers faced at work by organizations and data scientists

A recent survey from Kaggle: [https://www.kaggle.com/surveys/2017](https://www.kaggle.com/surveys/2017)
Difficulties in Data Access

- Data Scientists may not have enough resources to collect large datasets relevant to their tasks
  - AlphaGo samples 30 million data points for training
- Organizations want more data scientists to analyze their data, but they have difficulties in sharing sensitive data with them
  - A hospital wants data scientist from a university to analyze their EHR data
  - However, data sharing must be reviewed by legal departments and IRB, which takes long time without guarantee of approval
Synthetic Data Generation

- Data scientists and organizations can benefit from synthetic data generation
- Synthetic data is not the original data, but it resembles the “real thing” of original data in certain ways

VS.

Diet Coke

VS.

Coca-Cola
Synthetic Data Generation

- Data scientists and organizations can benefit from synthetic data generation
- Populating sample databases for
  - Testing systems on functionality/performance

Software test, e.g., Mockaroo

DB Benchmark, e.g., TPC-H
Synthetic Data Generation

- Data scientists and organizations can benefit from synthetic data generation
- Obtaining realistic training data for
  - Developing and evaluating ML algorithms

Same statistical properties as the original data

Same variable/label correlations as the original data
Synthetic Data Generation

- Data scientists and organizations can benefit from synthetic data generation
- Sharing internal/sensitive data for
  - Conquering difficulties collectively

Image Source: MIT News
Success in Image Generation

A. Brock, J. Donahue, K. Simonyan: Large Scale GAN Training for High Fidelity Natural Image Synthesis. ICLR 2019

Success in Text Generation

- 南陌春风早，东邻去日斜。
- 紫陌追随日，青门相见时。
- 胡风不开花，四气多作雪。

山夜有雪寒，桂里逢客时。此时人且引，酒愁一节梦。四面客归路，桂花开青竹。


The “Giant Language model Test Room” (GLTR) Project from Harvard University and MIT-IBM Watson AI Lab
Relational Data Generation

- **Challenge #1**: Relational data is more complex than image/text
- Relational data has mixed attribute types
  - Categorical, continuous, discrete etc.
- Relational data possesses semantics integrity
  - Column dependency: husband $\rightarrow$ male
- Real data has imbalanced label distribution
  - Most people have income $\leq 50K$, while only a few satisfy income $> 50K$
Relational Data Generation

- **Challenge #2**: Synthetic data should maintain the same properties of the original in certain ways.
- **Example**: Developing and evaluating ML algorithms.

![Diagram of Synthetic Data Generation](image)
Relational Data Generation

- **Challenge #3**: Synthetic data shouldn’t reveal private/sensitive information in original data
- **Example**: Anonymization vs. Re-identification attack
  - **Q**: Given the following fields, what % of the records can be traced to a specific individual
    - [Drug], [Dosage and refill information]
    - [Patient diagnosis]
    - [Patient ZIP inferred from Pharmacy ZIP]
    - [Prescription fill date]
  - **A**: 2.3% of individuals could be uniquely identified
    - 6.1% could be identifiable to a binsize of 2

Relational Data Generation

- Overview of representative generation methods

- Statistical Models
  - Copulas
  - Bayesian Network
  - Sum-Product Network

- Deep Generative Models
  - Variational Autoencoder (VAE)
  - Generative Adversarial Networks (GANs)
Outline

- Statistical Models
- Deep Generative Models
- An Empirical Study
- Summary and Future Directions
Statistical Models

- Basic Idea
  - Modeling a joint multivariate distribution for a dataset
  - Generating fake data by sampling from the distribution

1. Model
2. Sample

Original

<table>
<thead>
<tr>
<th>ID</th>
<th>Birthdate</th>
<th>Gender</th>
<th>Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>07/26/94</td>
<td>F</td>
<td>427.30</td>
</tr>
<tr>
<td>2</td>
<td>03/16/92</td>
<td></td>
<td>1001.42</td>
</tr>
<tr>
<td>3</td>
<td>01/17/87</td>
<td>M</td>
<td>33.33</td>
</tr>
<tr>
<td>4</td>
<td>08/16/94</td>
<td>M</td>
<td>54.33</td>
</tr>
<tr>
<td>5</td>
<td>09/04/82</td>
<td>M</td>
<td>5000.40</td>
</tr>
<tr>
<td>...</td>
<td>..</td>
<td>..</td>
<td>..</td>
</tr>
</tbody>
</table>

$X_1$, $X_2$, $X_3$, $X_4$

Synthetic

<table>
<thead>
<tr>
<th>ID</th>
<th>Birthdate</th>
<th>Gender</th>
<th>Balance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11/26/88</td>
<td>M</td>
<td>4002.50</td>
</tr>
<tr>
<td>2</td>
<td>06/14/91</td>
<td>F</td>
<td>352.47</td>
</tr>
<tr>
<td>3</td>
<td>04/15/90</td>
<td>F</td>
<td>129.75</td>
</tr>
<tr>
<td>4</td>
<td>02/17/95</td>
<td></td>
<td>37.20</td>
</tr>
<tr>
<td>5</td>
<td>03/29/84</td>
<td>M</td>
<td>427.92</td>
</tr>
<tr>
<td>...</td>
<td>..</td>
<td>..</td>
<td>..</td>
</tr>
</tbody>
</table>

$X_1$, $X_2$, $X_3$, $X_4$
Statistical Models

- Key Challenges
  - How to model the joint multivariate distribution?
- A simplified solution
  - Assuming that attributes are independent to each other, or the dependency can be specified by users
  - Then, we can focus single variable distribution $p(X_i)$
Copulas Model

- **Basic Idea**
  - Inferring dependency structures via copula functions, e.g., Gaussian copula function

<table>
<thead>
<tr>
<th>Age</th>
<th>Hours /week</th>
<th>Income</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>64</td>
<td>30K</td>
</tr>
<tr>
<td>31</td>
<td>82</td>
<td>60K</td>
</tr>
<tr>
<td>28</td>
<td>40</td>
<td>20K</td>
</tr>
<tr>
<td>43</td>
<td>36</td>
<td>80K</td>
</tr>
</tbody>
</table>

### Step 1: Computing DP marginal Histograms

![Histograms](image)

### Step 2: Computing DP correlation matrix through DP MLE (Maximum Likelihood Estimation)

\[
P = \begin{bmatrix}
1 & 0.053 & 0.108 \\
0.053 & 1 & 0.132 \\
0.108 & 0.132 & 1
\end{bmatrix}
\]

### Step 3: Sampling DP synthetic data

<table>
<thead>
<tr>
<th>Age</th>
<th>Hours /week</th>
<th>Income</th>
</tr>
</thead>
<tbody>
<tr>
<td>42</td>
<td>64</td>
<td>30K</td>
</tr>
<tr>
<td>31</td>
<td>82</td>
<td>60K</td>
</tr>
<tr>
<td>28</td>
<td>40</td>
<td>20K</td>
</tr>
<tr>
<td>43</td>
<td>36</td>
<td>80K</td>
</tr>
</tbody>
</table>

Figure is from [1]

Bayesian Network Model

- Basic Idea
  - Inferring dependency structures by constructing Bayesian network from the data

<table>
<thead>
<tr>
<th>age</th>
<th>gender</th>
<th>education</th>
<th>occupation</th>
<th>income</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>Male</td>
<td>HS-grad</td>
<td>Handlers-cleaners</td>
<td>&lt;=50K</td>
</tr>
<tr>
<td>53</td>
<td>Male</td>
<td>11th</td>
<td>Handlers-cleaners</td>
<td>&lt;=50K</td>
</tr>
<tr>
<td>28</td>
<td>Female</td>
<td>Bachelors</td>
<td>Prof-specialty</td>
<td>&lt;=50K</td>
</tr>
<tr>
<td>37</td>
<td>Female</td>
<td>Masters</td>
<td>Exec-managerial</td>
<td>&lt;=50K</td>
</tr>
<tr>
<td>30</td>
<td>Male</td>
<td>HS-grad</td>
<td>Transport-moving</td>
<td>&gt;50K</td>
</tr>
<tr>
<td>43</td>
<td>Female</td>
<td>Bachelors</td>
<td>Sales</td>
<td>&gt;50K</td>
</tr>
</tbody>
</table>

Bayesian Network Model

Dataset (D) → Bayesian Network (N) → Approximation of the distribution of data in D → Low dimensional marginals of D(P) → Noise

Sample → Synthetic Data
Sum-Product Network

• Basic Idea
  ◦ Inferring dependency structures by constructing a sum-product network model

Pros and Cons

• Pros:
  • The privacy protection, e.g., differential privacy, is guaranteed by theory.

• Cons:
  • It is hard to preserve the data utility
    • Real data usually has complex attribute correlation
    • Down-streaming tasks are complex
  • Not effective for mixed data types.
    • Numerical, categorical, etc.
Outline

- Statistical Models
- Deep Generative Models
- An Empirical Study
- Summary and Future Directions
Deep Generative Models

• Basic Idea
  ◦ Developing deep generative models to approximate the original dataset

• Recent Progress in Generative Models
  ◦ Variational Autoencoder (VAE)
  ◦ Generative Adversarial Networks (GANs)

The most interesting idea in the last 10 years in Machine Learning
Variational Autoencoder (VAE)

- Basic Idea
  - A VAE is an autoencoder that learns a representation (encoding) for the original dataset.

As close as possible

- Encoder Network
- Decoder Network

\[
\min_{D,E} \text{reconstruction error}(D(E(x)), x) + \frac{1}{2} \sum_i (\exp(\sigma_i) - (1 + \sigma_i) + m_i^2)
\]

Decoder can be used to generate data.

\[x' \]

Synthetic Data Generation
Vanilla GAN

- Image Generation
  - Input: a random noise $z$ from a prior $p_z(z)$
  - Output: a generated ("fake") image $x$

![Diagram of Vanilla GAN](image)

A Neural Network, e.g., MLP

Synthetic Data Generation
Vanilla GAN

- Adversarial Training Process

![Diagram showing the process of Vanilla GAN with a noise input, generator, discriminator, and output of real/fake classification.](image)
Vanilla GAN

- **Loss Function**

\[
\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]
\]

- **Training GAN by SGD**

  ◦ **Global Optimal:** \( p_g = p_{\text{data}} \)
Limitations of existing works

- Despite some attempts on GAN-based data synthesis, they have not systemically explore the design space
  - Which neural network architectures are better than others
  - How to encode record with mixed attributed types
  - How to deal with imbalanced label distribution

It calls for a *systemically experimental study*
Outline

- Statistical Models
- Deep Generative Models
- An Empirical Study
- Summary and Future Directions
The objective of data synthesis is to preserving the data utility of original data on supporting machine learning tasks, e.g., training a classification model in this paper.
Recap: Challenge of Relational Data

- **Challenge #1**: Relational data is more complex than image/text
- Relational data has mixed attribute types
  - Categorical, continuous, discrete etc.
- Relational data possesses semantics integrity
  - Column dependency: husband $\rightarrow$ male
- Real data has imbalanced label distribution
  - Most people have income $\leq 50K$, while only a few satisfy income $> 50K$
Overview of GAN Design


https://github.com/ruclty/Daisy
Overview of GAN Design

- Advantages of GAN-based data synthesis
  - No one-to-one relationship between real and fake records
  - Effectively emulating the real data for supporting downstream machine learning tasks, e.g., classification
# Design Space Exploration

<table>
<thead>
<tr>
<th>Components</th>
<th>Design Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Transformation</strong></td>
<td></td>
</tr>
<tr>
<td>Matrix as Input</td>
<td><em>Sample Form:</em> (1) Matrix (2) Vector</td>
</tr>
<tr>
<td>Vector as Input</td>
<td><em>Categorical:</em> (1) Ordinal (2) One-hot</td>
</tr>
<tr>
<td>Neural Networks</td>
<td><em>Numerical:</em> (1) Norm (2) GMM</td>
</tr>
<tr>
<td><strong>Training Algorithm</strong></td>
<td></td>
</tr>
<tr>
<td>(1) Vanilla + KL</td>
<td>(2) WGAN</td>
</tr>
<tr>
<td><strong>Conditional GAN</strong></td>
<td></td>
</tr>
<tr>
<td>(1) None</td>
<td><em>Condition:</em> (1) None (2) Label</td>
</tr>
<tr>
<td>(2) Label</td>
<td><em>Sampling:</em> (1) Random (2) Label-aware</td>
</tr>
<tr>
<td><strong>Differential Privacy</strong></td>
<td>(1) None (2) DPGAN (adding noise to gradients)</td>
</tr>
</tbody>
</table>
Data Transformation

- **Raw Records → Input (vector or matrix) of GAN**
  - **Categorical:** ordinal or one-hot encoding
  - **Numerical:** simple or mode-specific normalization

<table>
<thead>
<tr>
<th>Female</th>
<th>Bachelors</th>
<th>Sales</th>
<th>&gt;50K</th>
</tr>
</thead>
<tbody>
<tr>
<td>43</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Matrix-based

<table>
<thead>
<tr>
<th>43</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Vector-based (ordinal encoding)

| 43 | 1 | 2 | 1 | 1 |

Vector-based (one-hot encoding)

\[
\psi_{gmm} = \frac{v - \mu^{(k)}}{2\sigma^{(k)}}, \text{ where } k = \arg \max_i \pi^{(i)}
\]
Neural Network Design

• Matrix-based record representation
  ◦ Designing G and D using Convolutional Neural Networks (CNN)
  ◦ Generator G is a deconvolution process

![Deconvolution Process Diagram](image)
Neural Network Design

- Matrix-based record representation
  - Designing G and D using Convolutional Neural Networks (CNN)
  - Discriminator D is a convolution process
Neural Network Design

- Vector-based record representation
  - Designing G and D using Multilayer Perceptron (MLP)

![Diagram of Neural Network](image)

- Generator G
- Discriminator D

Synthetic Data Generation
Neural Network Design

• Vector-based record representation
  ◦ Designing $G$ using Recurrent Neural Networks, e.g., LSTM

![Diagram of Neural Network Design](image-url)

Generator $G$

Discriminator $D$
GAN Training Algorithm

**Algorithm 1: VANILLA-TRAIN** \((m, \alpha_d, \alpha_g, T)\)

**Input:** \(m\): batch size; \(\alpha_d\): learning rate of \(D\); \(\alpha_g\): learning rate of \(G\); \(T\): number of training iterations

**Output:** \(G\): Generator; \(D\): Discriminator

1. Initialize parameters \(\theta_d^{(0)}\) for \(D\) and \(\theta_g^{(0)}\) for \(G\)

2. for training iteration \(t = 1, 2, \ldots, T\) do
   
   /* Training discriminator \(D\) */
   
   3. Sample \(m\) noise samples \(\{z^{(i)}\}_{i=1}^{m}\) from noise prior \(p_z(z)\)
   
   4. Sample \(m\) samples \(\{t^{(i)}\}_{i=1}^{m}\) from real data \(p_{data}(t)\)
   
   5. \(\bar{g}_1 \leftarrow \nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[ \log D(t^{(i)}) + \log (1 - D(G(z^{(i)}))) \right]\)
   
   6. \(\theta_d^{(t)} \leftarrow \theta_d^{(t-1)} + \alpha_d \cdot \text{Adam}(\theta_d^{(t-1)}, \bar{g}_1)\)
   
   /* Training generator \(G\) */
   
   7. Sample \(m\) noise samples \(\{z^{(i)}\}_{i=1}^{m}\) from noise prior \(p_z(z)\)
   
   8. \(\bar{g}_2 \leftarrow \nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log(1 - D(G(z^{(i)})))\)
   
   9. \(\theta_g^{(t)} \leftarrow \theta_g^{(t-1)} - \alpha_g \cdot \text{Adam}(\theta_g^{(t-1)}, \bar{g}_2)\)

10. return \(G, D\)
Conditional GAN

- **Basic Idea**
  - Generate records of a given label as the condition

\[
\min_G \max_D V(G, D) = \mathbb{E}_{t \in p_{data}(t)} \left[ \log D(t|c) \right] \\
+ \mathbb{E}_{z \in p_z(z)} \left[ 1 - \log D(G(z|c)) \right]
\]

- **Random sampling**

- **Label-aware sampling**
Experimental Setup

- **Datasets**
  - Diverse domains, such as physical, social, etc.
  - Example: Adult dataset
    - 41,292 individuals extracted from the 1994 US census

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Domain</th>
<th>#Rec</th>
<th>#N</th>
<th>#C</th>
<th>#L</th>
<th>Skewness</th>
</tr>
</thead>
<tbody>
<tr>
<td>low-dimensional (#Attr ≤ 20)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTRU2</td>
<td>Physical</td>
<td>17,898</td>
<td>8</td>
<td>0</td>
<td>2</td>
<td>skew</td>
</tr>
<tr>
<td>Digits</td>
<td>Computer</td>
<td>10,992</td>
<td>16</td>
<td>0</td>
<td>10</td>
<td>balanced</td>
</tr>
<tr>
<td>Adult</td>
<td>Social</td>
<td>41,292</td>
<td>6</td>
<td>8</td>
<td>2</td>
<td>skew</td>
</tr>
<tr>
<td>CovType</td>
<td>Life</td>
<td>116,204</td>
<td>10</td>
<td>2</td>
<td>7</td>
<td>skew</td>
</tr>
<tr>
<td>high-dimensional (#Attr &gt; 20)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAT</td>
<td>Physical</td>
<td>6,435</td>
<td>36</td>
<td>0</td>
<td>6</td>
<td>balanced</td>
</tr>
<tr>
<td>Anuran</td>
<td>Life</td>
<td>7,195</td>
<td>22</td>
<td>0</td>
<td>10</td>
<td>skew</td>
</tr>
<tr>
<td>Census</td>
<td>Social</td>
<td>142,522</td>
<td>9</td>
<td>30</td>
<td>2</td>
<td>skew</td>
</tr>
<tr>
<td>Bing</td>
<td>Web</td>
<td>500,000</td>
<td>7</td>
<td>23</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
**Evaluation Process**

- **Step 1:** GAN Training
- **Step 2:** Classifier Training
- **Step 3:** Classifier Evaluation
- **Evaluation Metric**
  - We use F-Measure to evaluate the performance of classifiers $f$ and $f'$

$\text{Diff}(\mathcal{T}, \mathcal{T}') = |\text{Eval}(f|\mathcal{T}_{\text{test}}) - \text{Eval}(f'|\mathcal{T}_{\text{test}})|$

(We also evaluate the data utility by clustering and AQP.)
Neural Network Comparison

- So, which NN architecture wins?
  - CNN is not effective for relational data synthesis
  - LSTM with appropriate transformation (gmm + one-hot) achieves the best performance in most cases
  - MLP is more robust with different transformation schemes

<table>
<thead>
<tr>
<th>Classifier</th>
<th>CNN</th>
<th>MLP</th>
<th>LSTM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>nrm + ord</td>
<td>nrm + hot</td>
</tr>
<tr>
<td>DT10</td>
<td>0.475</td>
<td>0.062</td>
<td>0.062</td>
</tr>
<tr>
<td>DT30</td>
<td>0.485</td>
<td>0.071</td>
<td>0.049</td>
</tr>
<tr>
<td>RF10</td>
<td>0.417</td>
<td>0.035</td>
<td>0.038</td>
</tr>
<tr>
<td>RF20</td>
<td>0.458</td>
<td>0.060</td>
<td>0.066</td>
</tr>
<tr>
<td>AdaBoost</td>
<td>0.217</td>
<td>0.066</td>
<td>0.059</td>
</tr>
<tr>
<td>LR</td>
<td>0.047</td>
<td>0.018</td>
<td>0.088</td>
</tr>
</tbody>
</table>

Evaluating $\text{Diff}(T, T')$ on the Adult dataset
Neural Network Comparison

- Does LSTM always outperform MLP?
  - **NO!**
  - LSTM is sensitive to different hyper parameters

![Graphs showing F1-measure for different hyperparameters over epochs for LSTM and MLP on the Adult dataset.](image)

(a) Adult dataset LSTM.  (b) Adult dataset MLP.

What happens?
Mode Collapse

- LSTM with inappropriate hyper parameters would lead to **mode collapse**

<table>
<thead>
<tr>
<th>age</th>
<th>workclass</th>
<th>fnlwgt</th>
<th>education</th>
<th>...</th>
<th>income</th>
</tr>
</thead>
<tbody>
<tr>
<td>51</td>
<td>Private</td>
<td>167317</td>
<td>Prof-school</td>
<td>...</td>
<td>£50K</td>
</tr>
<tr>
<td>51</td>
<td>Private</td>
<td>167317</td>
<td>Prof-school</td>
<td>...</td>
<td>£50K</td>
</tr>
<tr>
<td>51</td>
<td>Private</td>
<td>167317</td>
<td>Prof-school</td>
<td>...</td>
<td>£50K</td>
</tr>
<tr>
<td>21</td>
<td>Private</td>
<td>167317</td>
<td>Prof-school</td>
<td>...</td>
<td>£50K</td>
</tr>
<tr>
<td>32</td>
<td>Private</td>
<td>167317</td>
<td>Prof-school</td>
<td>...</td>
<td>£50K</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>age</th>
<th>workclass</th>
<th>fnlwgt</th>
<th>education</th>
<th>...</th>
<th>income</th>
</tr>
</thead>
<tbody>
<tr>
<td>51</td>
<td>Private</td>
<td>584217</td>
<td>Prof-school</td>
<td>...</td>
<td>£50K</td>
</tr>
<tr>
<td>51</td>
<td>Private</td>
<td>15424</td>
<td>11th</td>
<td>...</td>
<td>£50K</td>
</tr>
<tr>
<td>51</td>
<td>Private</td>
<td>10245</td>
<td>HS-grad</td>
<td>...</td>
<td>£50K</td>
</tr>
<tr>
<td>32</td>
<td>Local-gov</td>
<td>167317</td>
<td>Prof-school</td>
<td>...</td>
<td>£50K</td>
</tr>
<tr>
<td>25</td>
<td>Local-gov</td>
<td>167317</td>
<td>Prof-school</td>
<td>...</td>
<td>£50K</td>
</tr>
</tbody>
</table>

**Takeaways:**
- MLP is more robust and achieves moderate results
- LSTM is more likely to result in mode collapse
- However, LSTM can achieve superior results if one pays sufficient efforts on hyper-parameter tuning
- See our paper about methods to avoid mode collapse

Dealing with imbalanced datasets

- Is Conditional GAN useful for imbalanced datasets?

Takeaway:
- Conditional GAN trained by label-aware sampling is useful to deal with imbalanced datasets
Is GAN Promising for preserving the utility?
  - Comparing with VAE and PrivBayes.

**Takeaway:**
- GAN significantly outperforms VAE and PB on synthetic data utility.
Generation Method Comparison

- Is GAN Promising for protecting the privacy?
  - Comparing with PrivBayes against the risk of re-identification.

<table>
<thead>
<tr>
<th>Method</th>
<th>Hitting Rate (%)</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Adult</td>
<td>CovType</td>
<td>Adult</td>
<td>CovType</td>
</tr>
<tr>
<td>PB-0.1</td>
<td>0.49</td>
<td>0.002</td>
<td>0.164</td>
<td>0.106</td>
</tr>
<tr>
<td>PB-0.2</td>
<td>0.88</td>
<td>0.006</td>
<td>0.147</td>
<td>0.094</td>
</tr>
<tr>
<td>PB-0.4</td>
<td>2.16</td>
<td>0.022</td>
<td>0.123</td>
<td>0.082</td>
</tr>
<tr>
<td>PB-0.8</td>
<td>4.40</td>
<td>0.056</td>
<td>0.112</td>
<td>0.073</td>
</tr>
<tr>
<td>PB-1.6</td>
<td>4.64</td>
<td>0.070</td>
<td>0.110</td>
<td>0.069</td>
</tr>
<tr>
<td>GAN</td>
<td>0.30</td>
<td>0.500</td>
<td>0.113</td>
<td>0.072</td>
</tr>
</tbody>
</table>

Takeaway:
- GAN can reduce the risk of re-identification as there is no one-to-one relationship between real and synthetic records.
Can GAN preserve DP?

- Is GAN Promising for protecting the privacy?
  - Comparing with PrivBayes with differential privacy (DP) guarantee.

- The current solution for differential privacy preserving GAN cannot beat traditional data synthesis methods with DP guarantees.
Conclusion

- GAN is very promising for relational data synthesis. It generates synthetic data with very good utility.
- GAN also achieves competitive performance on protecting privacy against the risk of re-identification.
- GAN has limitations on providing provable privacy protection.
Outline

- Statistical Models
- Deep Generative Models
- An Empirical Study
- **Summary and Future Directions**
Summary

- **Statistical Models**
  - Copulas, Bayesian Network, Sum-Product Network

- **Deep Generative Models**
  - VAE, GAN

- **An Empirical Study**
  - GAN performs well in balancing data utility and privacy
  - GAN has limitations on preserving DP

- **GAN is promising, but still not ready**
Future Directions

- On-Demand Data Generation Systems
  - Democratizing data science
  - Generating benchmarks for learnable DB
  - Supporting data marketing
Future Directions

- Deep Generative Models for Cardinality Estimation and AQP
- Our Experiments:

<table>
<thead>
<tr>
<th>Dataset</th>
<th>VAE</th>
<th>PB-0.2</th>
<th>PB-0.4</th>
<th>PB-0.8</th>
<th>PB-1.6</th>
<th>GAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>CovType</td>
<td>0.251</td>
<td>0.201</td>
<td>0.113</td>
<td>0.183</td>
<td>0.108</td>
<td>0.015</td>
</tr>
<tr>
<td>Census</td>
<td>0.469</td>
<td>2.348</td>
<td>1.262</td>
<td>0.786</td>
<td>0.767</td>
<td>0.240</td>
</tr>
<tr>
<td>Bing</td>
<td>0.632</td>
<td>0.830</td>
<td>0.805</td>
<td>0.783</td>
<td>0.761</td>
<td>0.422</td>
</tr>
</tbody>
</table>

- More works

Thanks

Q & A

Homepage: http://iir.ruc.edu.cn/~fanj/