
dpmm: Differentially Private Marginal Models, a Library for Synthetic Tabular Data Generation

Sofiane Mahiou¹ Amir Dizche¹ Reza Nazari¹ Xinmin Wu¹
Ralph Abbey¹ Jorge Silva¹ Georgi Ganev^{1,2}
¹SAS ²UCL
sofiane.mahiou@sas.com

Abstract

We propose *dpmm*, an open-source library for synthetic data generation with Differentially Private (DP) guarantees. It includes three popular marginal models – PrivBayes, MST, and AIM – that achieve superior utility and offer richer functionality compared to alternative implementations. Additionally, we adopt best practices to provide end-to-end DP guarantees and address well-known DP-related vulnerabilities. Our goal is to accommodate a wide audience with easy-to-install, highly customizable, and robust model implementations.

Our codebase is available from: <https://github.com/sassoftware/dpmm>.

1 Introduction

Privacy-preserving synthetic data, leveraging rigorous privacy definitions like Differential Privacy (DP), offers a promising solution for sharing sensitive data within and across organizations while safeguarding individual privacy. Interest in synthetic data has been growing not only in the research community [8, 21, 23], but it has also been adopted by government agencies for sharing census data [20, 36, 40], as well as in finance and healthcare sectors [15, 35, 45]. The underlying idea is to train generative machine learning models on sensitive data while satisfying the definition of DP through ingesting well-calibrated randomness and noise [12, 14].

While there are numerous proposed DP generative modeling approaches in the literature [1, 2, 4, 5, 7, 22, 24, 25, 27, 32, 33, 46–48, 50–52], marginal models, particularly MST [32] and AIM [33] (and to an extent PrivBayes [50]), have consistently demonstrated strong privacy-utility tradeoffs [16, 44]. This is especially true for relatively small datasets with fewer than 32 features. These models broadly rely on the *select-measure-generate* paradigm [30, 32], i.e., they select a collection of marginals of interest, measure them while adding noise, and then generate synthetic data that preserves them.

As a result, these three marginal models – PrivBayes, MST, and AIM – have been reimplemented by several popular open-source libraries for DP synthetic data generation [28, 29, 41–43]. However, most libraries are cumbersome to install, have outdated dependencies, do not implement all models, and the ones they do implement often have limited functionality. Moreover, many of the implementations lack actual end-to-end DP pipelines, as the data domain might be directly extracted from the input data, or data preprocessing might be done in a non-DP manner [3, 17]. Additionally, they may contain DP-related bugs such as fixed random states and floating-point vulnerabilities [6, 19, 26].

To address these limitations, we propose *Differentially Private marginal models*, or *dpmm*, a lightweight open-source library (pip installable; Apache-2.0 license) for synthetic data generation. We incorporate best practices from various scientific papers and DP libraries to provide robust implementations of the three most widely used marginal models with end-to-end DP guarantees and rich functionality (see Sec. 2). Additionally, we achieve higher utility compared to previous implementations and conduct DP auditing experiments (see Sec. 3).

Library	Marginal Model			DP Guarantee		
	PrivBayes [50]	MST [32]	AIM [33]	Data Domain	Data Preprocessing	Floating-Point Precision
<i>dpmm</i> (ours)	✓	✓	✓	✓	✓	✓
private-pgm [29]	✗	✓	✓	~ ¹	✗	✗
OpenDP [41]	✗	✓	✓	~ ²	~ ³	✓
synthcity [43]	✓	✗	✓	✗	✗	✗

Table 1: Comparison between *dpmm* and other libraries over marginal models and end-to-end DP support. ¹accepts data domain only for discrete data; ²data domain cannot be passed as input, extracts data domain in a DP way only for continuous data; ³only supports uniform discretization, which is incidentally DP.

Library	Mixed Data Support	Conditional Generation	Public Data Pretraining	Structural Zeros	Max Model Size	Serialization
<i>dpmm</i> (ours)	✓	✓	✓	✓	✓	✓
private-pgm [29]	✗	✗	✗	MST	AIM	✗
OpenDP [41]	✓	✗	✗	MST	AIM	✗
synthcity [43]	✓	✗	✗	✗	AIM	✓

Table 2: Comparison between *dpmm* and other libraries over different functionality support.

Main Contributions:

- We implement and open source *dpmm*, a lightweight library for end-to-end DP synthetic data generation, containing three popular marginal models with extended functionality – PrivBayes, MST, and AIM (see left half of Table 1 as well as Table 2).
- *dpmm* achieves higher utility than previous implementations – on average 1.5% higher than private-pgm and 147% than OpenDP/synthcity (see Fig. 1a) – due to improved preprocessing.
- *dpmm* contains state-of-the-art DP auditing procedures and effectively addresses known DP-related vulnerabilities (see right half of Table 1).

2 Overview of *dpmm*

In this section, we provide an overview of *dpmm* in terms of the implemented models, the building blocks we use to guarantee end-to-end DP pipeline, and the supported functionality across all models.

Marginal Models. As mentioned, we implement three marginal models in *dpmm* – PrivBayes [50], MST [32], and AIM [33]. We do so because they: i) are known to perform well in the research community [16, 44], ii) ranked among the best solutions in the NIST DP synthetic data challenge [39], iii) have been used to release census data by government agencies [20, 40], and iv) have been implemented in popular open-source libraries but have known DP-related vulnerabilities [3, 6, 19, 26].

The three models broadly rely on the select-measure-generate paradigm [30, 32]. They primarily differ in the first step, i.e., selecting a collection of marginals of interest. In the following step they measure these marginals noisily with the Gaussian mechanism [34] and then, one could generate synthetic datasets that are consistent with these noisy marginals through Private-PGM [31]. Next, we discuss how the three models select marginals:

- **PrivBayes** constructs an optimal Bayesian network of degree k by optimizing the mutual information scores between the nodes in the network using the Exponential mechanism [13]. While the original PrivBayes uses the Laplace mechanism [14] for marginals measurement and directly samples from them, we adjust our implementation to incorporate the Gaussian mechanism and Private-PGM, similar to [31] to unify it with the MST/AIM implementations. We refer to this model as PrivBayes+PGM (note: it satisfies (ϵ, δ) -DP instead of pure DP).
- **MST** first chooses all one-way marginals (corresponding to the columns in the dataset) and then selects a collection of two-way marginals that form a maximum spanning tree (an undirected graph) of the underlying correlation graph.
- **AIM** also starts by selecting all one-way marginals but then chooses higher-order ones iteratively based on their overall importance and comparative error to already chosen ones.

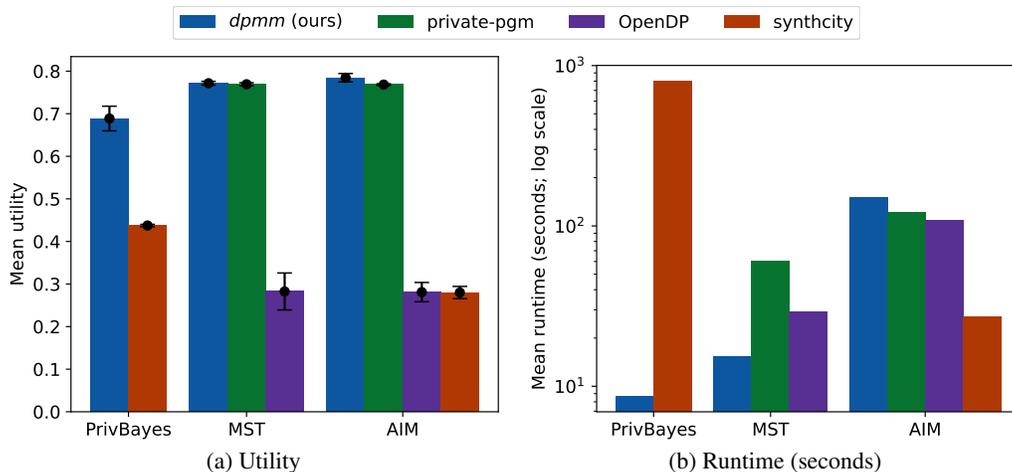


Figure 1: Comparison between *dpmm* and other libraries in terms of utility and runtime, all models are trained with default hyperparameters, $\epsilon = 1$, and $\delta = 10^{-5}$ on Wine.

DP Guarantees. Numerous popular DP libraries, including synthetic data generation ones, contain well/long-known DP vulnerabilities such as direct data domain extraction from the input data and floating-point (im)precision [3, 6, 19, 26] (see Table 1). To ensure end-to-end DP, we rely on scientific papers and best practices addressing these issues.

- **Data Domain.** *dpmm* allows the data domain (for both numerical and categorical data) to be provided as input. Alternatively, we follow Desfontaines [9]’s method to extract min/max for numerical data using half the preprocessing budget $0.5\epsilon_{proc}$ (also implemented in OpenDP).
- **Data Preprocessing.** Since all models operate on discrete data, *dpmm* has two discretization strategies – PrivTree [49] and uniform. The former is a DP tree-based method using $0.5\epsilon_{proc}$ (Laplace mechanism) and is the default since it has shown superior performance [18, 44].
- **Floating-Point Precision.** To defend against floating-point vulnerabilities when sampling from the Gaussian mechanism, *dpmm* follows Casacuberta et al. [6]’s method, whose effectiveness has been verified by researchers [10, 26] (originally implemented in OpenDP).

Functionality. Next, we discuss a set of features, which are available to all three models in *dpmm*:

- **Mixed Data Support:** the training data can contain both numerical and categorical data.
- **Conditional Generation:** generated synthetic data can satisfy any conditions at generation time, without the need to retrain the model or use negative sampling, due to Private-PGM.
- **Public Pretraining:** the model can be pretrained on public data, i.e., extract the data domain, fit data preprocessing, and select the marginals of interest without spending privacy budget.
- **Structural Zeros:** categories or numerical intervals that always have zero counts in the training data can be configured to be preserved in the generated synthetic data. Additionally, the accurate modeling of very rare categories or numerical intervals can be suppressed.
- **Max Model Size:** the size of the trained model (in terms of MBs) can be controlled.
- **Serialization:** trained models can be saved and reloaded for later use.

3 Empirical Evaluation

In this section, we evaluate the three marginal models in *dpmm* (PrivBayes+PGM, MST, and AIM) both in terms of utility and privacy, and compare them to the implementations in other libraries.

Utility. We train models with $\epsilon = 1$ (and $\epsilon_{proc} = 0.1$ for processing when applicable) using all available implementations on the Wine dataset [11]. We use the default hyperparameters for all implementations (with $\delta = 10^{-5}$), while the data domain is assumed to be known and provided as input. Utility is measured as the average score between real and synthetic datasets across three metrics: marginal (1-way) and pairwise (2-way) similarity, as well as the ability to distinguish real from synthetic records. All reported scores are averaged across ten trained model instances and ten generated synthetic datasets per trained model. We visualize the utility and runtime results in Fig. 1.

Compared to the original private-pgm, *dpmm*’s MST and AIM perform on par and even achieve slightly better results, with 1.5% improvement. We attribute this to the improved preprocessing

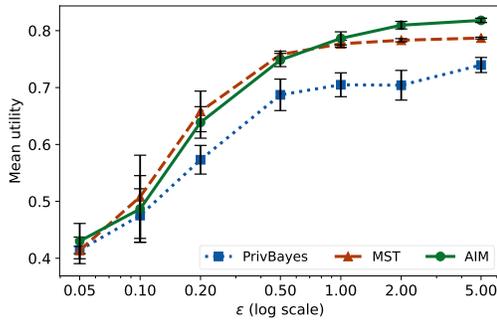


Figure 2: Utility-privacy tradeoffs of *dpmm* on Wine.

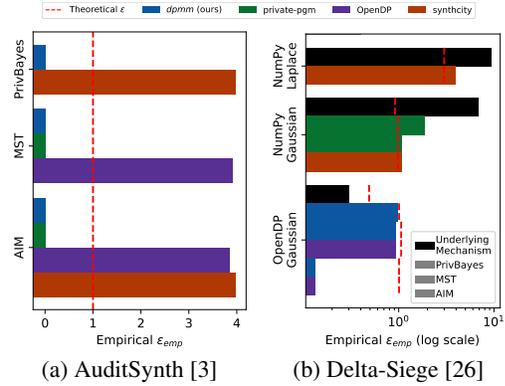


Figure 3: DP auditing of *dpmm* and other libraries.

strategy, PrivTree, consistent with [18]. Moreover, *dpmm*'s MST runtime is significantly faster due to parallelization when measuring all 1-way marginals. Unfortunately, AIM takes longer, as smaller marginal estimation errors allow the model to run for more iterations before the budget is spent. Also, *dpmm*'s implementations dramatically outperform OpenDP and synthcity by 147% on average.

Additionally, in Fig. 2, we see that all models in *dpmm* exhibit desirable behaviors, i.e., their utility increases with more available privacy budget. Consistent with previous research [16, 33, 44], MST and AIM perform better than PrivBayes+PGM, while AIM edges out MST for $\epsilon > 0.5$.

DP Auditing. We also include state-of-the-art DP auditing techniques in *dpmm*, alongside extensive unit and integration tests, to ensure that our implementations are free from (known) privacy vulnerabilities. Specifically, we implement two powerful attacks, AuditSynth [3] and Delta-Siege [26], which have been used to: i) detect general privacy violations in PrivBayes and MST [3], and ii) identify floating-point precision violations in MST and AIM [26]. To achieve a tight audit (i.e., the empirical privacy loss ϵ_{emp} approaches the theoretical guarantee ϵ) or detect a violation (i.e., $\epsilon_{emp} > \epsilon$), both attacks assume white-box access to the model (the fitted model's parameters and generated synthetic data), a worst-case target record (lying outside the domain of the rest of the data), and worst-case neighboring datasets (constructed from a couple low-dimensional records).

AuditSynth [3] uses membership inference attacks via a repeated distinguishing game [37, 38], in which an adversary attempts to determine whether the target record was part of the model's training data. We train 1,000 models per implementation ($\epsilon = 1, \delta = 10^{-3}$) and report the resulting ϵ_{emp} s in Fig. 3a. Both synthcity and OpenDP are vulnerable to this class of general attacks as $\epsilon_{emp} \gg \epsilon$ (indicating detected privacy violations), primarily because the data domain cannot be passed as input. In contrast, private-pgm and *dpmm* exhibit minimal privacy leakage that remains well within expected bounds, assuming the data domain is specified, which is consistent with prior findings [3, 18].

Delta-Siege [26] relies on the insight that many (ϵ, δ) pairs can be grouped and efficiently audited together, as they result in the same algorithm. First, we audit the underlying DP mechanisms used by each library (see black bars in Fig. 3b): NumPy Laplace (synthcity), NumPy Gaussian (private-pgm, synthcity), and OpenDP Gaussian [6] (*dpmm*, OpenDP). Mechanisms relying on NumPy exhibit floating-point precision violations ($\epsilon_{emp} \gg \epsilon$), while OpenDP Gaussian does not, as verified by previous work [10, 26]. Then, we confirm that these trends hold when auditing the full implementations of the three models (see colored bars in Fig. 3b). Specifically, we train numerous models ($\epsilon = 1, \delta = 10^{-3}$) and confirm that only *dpmm* and OpenDP avoid excessive privacy leakage.

Overall, to the best of our knowledge, *dpmm* is the only library containing state-of-the-art DP synthetic data models and does not exhibit (known) privacy-related vulnerabilities. While these audits cannot guarantee the absence of all bugs, they provide strong empirical evidence of *dpmm*'s robustness.

4 Conclusion

In this paper, we introduced *dpmm*, a library for DP synthetic data generation that includes robust implementations of PrivBayes+PGM, MST, and AIM, offering rich functionality. We hope that *dpmm* will serve both practitioners and researchers in their adoption and development of DP synthetic data. We look forward to continuously improving the functionality, trustworthiness, and performance of the library with the support of the community.

References

- [1] Nazmiye Ceren Abay, Yan Zhou, Murat Kantarcioglu, Bhavani Thuraisingham, and Latanya Sweeney. Privacy preserving synthetic data release using deep learning. In *ECML PKDD*, 2018.
- [2] Gergely Acs, Luca Melis, Claude Castelluccia, and Emiliano De Cristofaro. Differentially private mixture of generative neural networks. *IEEE TKDE*, 2018.
- [3] Meenatchi Sundaram Muthu Selva Annamalai, Georgi Ganey, and Emiliano De Cristofaro. “What do you want from theory alone?” experimenting with tight auditing of differentially private synthetic data generation. In *USENIX Security*, 2024.
- [4] Sergul Aydore, William Brown, Michael Kearns, Krishnaram Kenthapadi, Luca Melis, Aaron Roth, and Ankit A Siva. Differentially private query release through adaptive projection. In *ICML*, 2021.
- [5] Kuntai Cai, Xiaoyu Lei, Jianxin Wei, and Xiaokui Xiao. Data synthesis via differentially private markov random fields. *PVLDB*, 2021.
- [6] Sílvia Casacuberta, Michael Shoemate, Salil Vadhan, and Connor Wagaman. Widespread underestimation of sensitivity in differentially private libraries and how to fix it. In *ACM CCS*, 2022.
- [7] Thee Chanyaswad, Changchang Liu, and Prateek Mittal. Ron-gauss: Enhancing utility in non-interactive private data release. *PoPETs*, 2019.
- [8] Emiliano De Cristofaro. Synthetic Data: Methods, Use Cases, and Risks. *IEEE S&P Magazine*, 2024.
- [9] Damien Desfontaines. *Lowering the cost of anonymization*. PhD thesis, ETH Zurich, 2020.
- [10] Damien Desfontaines and Samuel Haney. How to Break, Then Fix, Differential Privacy on Finite Computers. In *USENIX PEPR*, 2023.
- [11] Dheeru Dua and Casey Graff. UCI Machine Learning Repository. <https://archive.ics.uci.edu/datasets>, 2017.
- [12] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 2014.
- [13] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *EuroCrypt*, 2006.
- [14] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *TCC*, 2006.
- [15] FCA. Using Synthetic Data in Financial Services. <https://www.fca.org.uk/publication/corporate/report-using-synthetic-data-in-financial-services.pdf>, 2024.
- [16] Georgi Ganey, Kai Xu, and Emiliano De Cristofaro. Graphical vs. Deep Generative Models: Measuring the Impact of Differentially Private Mechanisms and Budgets on Utility. In *ACM CCS*, 2024.
- [17] Georgi Ganey, Meenatchi Sundaram Muthu Selva Annamalai, and Emiliano De Cristofaro. The Elusive Pursuit of Reproducing PATE-GAN: Benchmarking, Auditing, Debugging. *TMLR*, 2025.
- [18] Georgi Ganey, Meenatchi Sundaram Muthu Selva Annamalai, Sofiane Mahiou, and Emiliano De Cristofaro. The Importance of Being Discrete: Measuring the Impact of Discretization in End-to-End Differentially Private Synthetic Data. *arXiv:2504.06923*, 2025.
- [19] Samuel Haney, Damien Desfontaines, Luke Hartman, Ruchit Shrestha, and Michael Hay. Precision-based attacks and interval refining: how to break, then fix, differential privacy on finite computers. In *TPDP*, 2022.
- [20] Shlomi Hod and Ran Canetti. Differentially Private Release of Israel’s National Registry of Live Births. In *IEEE S&P*, 2025.
- [21] Yuzheng Hu, Fan Wu, Qinbin Li, Yunhui Long, Gonzalo Munilla Garrido, Chang Ge, Bolin Ding, David Forsyth, Bo Li, and Dawn Song. SoK: Privacy-Preserving Data Synthesis. In *IEEE S&P*, 2024.
- [22] James Jordon, Jinsung Yoon, and Mihaela Van Der Schaar. PATE-GAN: generating synthetic data with differential privacy guarantees. In *ICLR*, 2018.
- [23] James Jordon, Lukasz Szpruch, Florimond Houssiau, Mirko Bottarelli, Giovanni Cherubin, Carsten Maple, Samuel N Cohen, and Adrian Weller. Synthetic Data—what, why and how? *arXiv:2205.03257*, 2022.
- [24] Haoran Li, Li Xiong, and Xiaoqian Jiang. Differentially private synthesization of multi-dimensional data using copula functions. In *EDBT*, 2014.
- [25] Terrance Liu, Giuseppe Vietri, and Steven Z Wu. Iterative methods for private synthetic data: Unifying framework and new methods. *NeurIPS*, 2021.
- [26] Johan Lokna, Anouk Paradis, Dimitar I. Dimitrov, and Martin Vechev. Group and Attack: Auditing Differential Privacy. In *ACM CCS*, 2023.
- [27] Yunhui Long, Boxin Wang, Zhuolin Yang, Bhavya Kailkhura, Aston Zhang, Carl A. Gunter, and Bo Li. G-PATE: Scalable differentially private data generator via private aggregation of teacher discriminators. In *NeurIPS*, 2021.

- [28] Sofiane Mahiou, Kai Xu, and Georgi Ganev. dpart: Differentially private autoregressive tabular, a general framework for synthetic data generation. In *TPDP*, 2022.
- [29] Ryan McKenna. private-pgm. <https://github.com/ryan112358/private-pgm>, 2019.
- [30] Ryan McKenna and Terrance Liu. A simple recipe for private synthetic data generation. *Differential Privacy*.org, 2022. <https://differentialprivacy.org/synth-data-1/>.
- [31] Ryan McKenna, Daniel Sheldon, and Gerome Miklau. Graphical-model based estimation and inference for differential privacy. In *ICML*, 2019.
- [32] Ryan McKenna, Gerome Miklau, and Daniel Sheldon. Winning the NIST Contest: A scalable and general approach to differentially private synthetic data. *JPC*, 2021.
- [33] Ryan McKenna, Brett Mullins, Daniel Sheldon, and Gerome Miklau. AIM: an adaptive and iterative mechanism for differentially private synthetic data. *PVLDB*, 2022.
- [34] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *FOCS*, 2007.
- [35] Microsoft. IOM and Microsoft release first-ever differentially private synthetic dataset to counter human trafficking. <https://www.microsoft.com/en-us/research/blog/iom-and-microsoft-release-first-ever-differentially-private-synthetic-dataset-to-counter-human-trafficking/>, 2022.
- [36] NASEM. *2020 Census Data Products: Data Needs and Privacy Considerations: Proceedings of a Workshop*. The National Academies Press, 2020.
- [37] Milad Nasr, Shuang Songi, Abhradeep Thakurta, Nicolas Papernot, and Nicholas Carlin. Adversary instantiation: Lower bounds for differentially private machine learning. In *IEEE S&P*, 2021.
- [38] Milad Nasr, Jamie Hayes, Thomas Steinke, Borja Balle, Florian Tramèr, Matthew Jagielski, Nicholas Carlini, and Andreas Terzis. Tight Auditing of Differentially Private Machine Learning. In *USENIX Security*, 2023.
- [39] NIST. 2018 Differential privacy synthetic data challenge. <https://www.nist.gov/ctl/pscr/open-innovation-prize-challenges/past-prize-challenges/2018-differential-privacy-synthetic>, 2018.
- [40] ONS. Synthesising the linked 2011 Census and deaths dataset while preserving its confidentiality. <https://datasciencecampus.ons.gov.uk/synthesising-the-linked-2011-census-and-deaths-dataset-while-preserving-its-confidentiality/>, 2023.
- [41] OpenDP. SmartNoise SDK: Tools for Differential Privacy on Tabular Data. <https://github.com/opendp/smartnoise-sdk>, 2021.
- [42] Haoyue Ping, Julia Stoyanovich, and Bill Howe. DataSynthesizer: Privacy-Preserving Synthetic Datasets. In *SSDBM*, 2017.
- [43] Zhaozhi Qian, Rob Davis, and Mihaela van der Schaar. Synthcity: a benchmark framework for diverse use cases of tabular synthetic data. In *NeurIPS Datasets and Benchmarks Track*, 2023. <https://github.com/vanderschaarlab/synthcity>.
- [44] Yuchao Tao, Ryan McKenna, Michael Hay, Ashwin Machanavajjhala, and Gerome Miklau. Benchmarking differentially private synthetic data generation algorithms. In *PPAI*, 2022.
- [45] UK ICO. Synthetic data to test the effectiveness of a vulnerable person’s detection system in financial services. <https://ico.org.uk/for-organisations/uk-gdpr-guidance-and-resources/data-sharing/privacy-enhancing-technologies/case-studies/synthetic-data-to-test-the-effectiveness-of-a-vulnerable-persons-detection-system-in-financial-services/>, 2023.
- [46] Giuseppe Vietri, Grace Tian, Mark Bun, Thomas Steinke, and Steven Wu. New oracle-efficient algorithms for private synthetic data release. In *ICML*, 2020.
- [47] Giuseppe Vietri, Cedric Archambeau, Sergul Aydore, William Brown, Michael Kearns, Aaron Roth, Ankit Siva, Shuai Tang, and Steven Z Wu. Private synthetic data for multitask learning and marginal queries. *NeurIPS*, 2022.
- [48] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. Differentially private generative adversarial network. *arXiv:1802.06739*, 2018.
- [49] Jun Zhang, Xiaokui Xiao, and Xing Xie. Privtree: A differentially private algorithm for hierarchical decompositions. In *SIGMOD*, 2016.
- [50] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao. PrivBayes: Private Data Release via Bayesian Networks. *ACM TDS*, 2017.
- [51] Xinyang Zhang, Shouling Ji, and Ting Wang. Differentially private releasing via deep generative model (technical report). *arXiv:1801.01594*, 2018.
- [52] Zhikun Zhang, Tianhao Wang, Jean Honorio, Ninghui Li, Michael Backes, Shibo He, Jiming Chen, and Yang Zhang. PrivSyn: Differentially Private Data Synthesis. In *USENIX Security*, 2021.