Privacy Utility Tradeoff Between PETs: Differential Privacy and Synthetic Data

Qaiser Razi, Sujoya Datta, Vikas Hassija, GSS Chalapathi, *Senior Member, IEEE*, and Biplab Sikdar, *Senior Member, IEEE*

***Abstract*—Data privacy is a critical concern in the digital age. This problem has compounded with the evolution and increased adoption of machine learning (ML), which has necessitated balancing the security of sensitive information with model utility. Traditional data privacy techniques, such as differential privacy and anonymization, focus on protecting data at rest and in transit but often fail to maintain high utility for machine learning models due to their impact on data accuracy. In this article, we explore the use of synthetic data as a privacy-preserving method that can effectively balance data privacy and utility. Synthetic data is generated to replicate the statistical properties of the original dataset while obscuring identifying details, offering enhanced privacy guarantees. We evaluate the performance of synthetic data against differentially private and anonymized data in terms of prediction accuracy across various settings—different learning rates, network architectures, and datasets from various domains. Our findings demonstrate that synthetic data maintains higher utility (prediction accuracy) than differentially private and anonymized data. The study underscores the potential of synthetic data as a robust privacy-enhancing technology (PET) capable of preserving both privacy and data utility in machine learning environments.**

***Index Terms*—Accuracy, anonymization, differential privacy, machine learning, PETs, synthetic data.**

I. Introduction

DATA privacy is essential in the digital age to defend individual freedom, stop fraud and identity theft, build confidence in digital services, encourage ethical data usage, and protect national security. It preserves confidence in online interactions, stops cybercrime, and gives individuals control over their personal information.

Data privacy is the foundation of individual liberty in the digital age. It is important to secure the use of data to prevent unauthorized surveillance of individuals. Data privacy measures have evolved through various stages, reflecting changes in regulations, technological breakthroughs, and public awareness. Before the digital era, physical protection and restricted access were key components of data privacy. Only authorized staff had access to the paper records that contained personal information. Data storage changed from paper records to electronic databases with the introduction of computers. Optimization in privacy was still largely undeveloped despite the introduction of basic security features such as passwords and access limits. Implementations of database management systems (DBMS) enhanced the retrieval and organization of data. Sensitive data might be sent and stored securely thanks to advancements in encryption technology. However, because of centralized data storage and lax encryption standards, intrusions continued to happen. Cryptography and other methods effectively protect data when it is in transit and at rest. However, there has been a need for methodologies to protect data in use. Extensive work has brought forth a plethora of mechanisms that attempt to secure data when in use.

However, when one speaks of data privacy in machine learning, the accuracy of the model must be taken into consideration, as a secure yet inefficient model is not useful. Section III of this article has listed some of the key works in this regard. It is important to consider privacy from a machine-learning perspective. If data is not secure, adversaries can identify whether a specific data point is part of a dataset or make inferences about sensitive information, such as classification labels. Such possibilities are inherent in technology, and as a result, there are well-defined methods to effectively address these vulnerabilities. One may argue that algorithms to do such tasks already prevail and are in use. However, it must be noted that they secure data when it is either at rest or in motion, but not when it is in use. Data protection in processing is the main topic of concern [1]. In [2], the author proposes a new method for picture encryption and decryption based on self-adaptive chaotic substitution and a computational genetic approach. This technique aims to ensure individual and smart device authentication, protect data privacy both at rest and in transit, and address various user requirements. The authors in [3] unveil a privacy-preserving communication (PCSS) scheme for software-defined networking (SDN)-enabled smart homes. Additionally, PCSS offers privacy-preserving user queries for smart homes and prevents an intruder from learning or altering data during transmission.

Received 12 February 2024; revised 30 August 2024; accepted 7 October 2024. Date of publication 14 November 2024; date of current version 3 April 2025. *(Corresponding author: GSS Chalapathi.)*

Qaiser Razi and GSS Chalapathi are with the Department of Electrical and Electronics Engineering, Birla Institute of Technology and Science (BITS-Pilani), Rajasthan 333031, India (e-mail: p20210070@pilani.bits-pilani.ac.in; gssc@pilani.bits-pilani.ac.in).

Sujoya Datta and Vikas Hassija are with the School of Computer Engineering, KIIT University, Bhubaneshwar, Odisha 751024, India (e-mail: 2255003@kiit.ac.in; vikas.hassijafcs@kiit.ac.in).

Biplab Sikdar is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576 (e-mail: bsikdar@nus.edu.sg).

In [4], the author briefs about several technologies available that try to protect data even when it is in use, and such methods are known as computing on encrypted data (COED), which are among the so-called privacy-enhancing technologies (PETs). PETs include statistical methods like homomorphic encryption [5], multiparty computation (MPC) [6], data anonymization [7], differential privacy [8], synthetic data [9], and federated learning [10]. To safeguard the privacy of sensitive information and support vectors during computation and transmission, the authors in [11] have developed a secure and effective classification technique utilizing support vector machines (SVM). At the same time, these techniques do their part in ensuring the privacy of data while in use. The objective of this article is to compare the performance of a few selected techniques. We focus on three key techniques: differential privacy, anonymization, and synthetic data. In the context of this article, our primary concern is safeguarding data privacy in machine learning. This study deals with establishing which approach works better in ensuring privacy without compromising on accuracy.

Motivation: Our primary concern is safeguarding data privacy in machine learning when data is in use. Our aim is to evaluate the performance of synthetic data compared to differentially private and anonymized data in terms of prediction accuracy. By doing these evaluations, we emphasize the use of synthetic data, which is a replica of the original data, as it provides a balance between privacy and utility.

II. Organization

The rest of this article is organized as follows. The recent works related to three essential PETs, i.e., differential privacy, anonymization, and synthetic data, are presented in Section III. Preliminary concepts related to differential privacy and synthetic data and the problem statement for our article have been discussed in Section IV. Section V illustrates the evaluation methodology we have used in our experiments. Section VI presents the experiments that were conducted with results that support our findings. In Section VII, the conclusion to our study is presented. Finally, Section VIII presents the future scope of our research.

III. Related Works

This section presents most of the works aiming toward the privacy preservation of data while in use. It is not unknown that introducing noise to a dataset compromises accuracy while releasing the data to outside parties, as it heightens the risk of data or label leakage when extensively queried by adversaries, which will be proved in the experiments further into this article. We have divided this section into three subsections summarizing works related to differential privacy, anonymization, and synthetic data about their abilities to secure original data.

A. Works Related to Differential Privacy

Differential privacy is a widely used privacy model. Due to its high privacy guarantee, i.e., an individual's existence or absence in a dataset does not materially affect the outcomes of studies conducted on the dataset. Enforcing this stringent assurance, however, severely distorts data, restricting its applications and reducing the analytical value of the differentially private results. In [12], the authors contend that the conventional formalization of differential privacy is more stringent than the privacy guarantee that it aims to provide. This results in a considerable loss of accuracy since it restricts the data controller's capacity to modify the degree of safeguarding given actual data. In [13], the authors propose a differentially private strategy that requires only one parameter to be tuned to withstand both membership inference and model inversion attacks efficiently. Sei et al. [14] proposes true-value-based differential privacy (TDP), a unique privacy paradigm. This paradigm does not apply standard differential privacy to the "measured value" that contains mistakes but rather to the "true value" that the information owner or anonymizer does not know. The optimized differential privacy (ODP) technique is put forth by the authors in [15] as a means of protecting the confidentiality of each data point while permitting the extraction of important information. The assessment findings of [16] produced after each iteration of the K-means clustering algorithm serve as the foundation for the methodology. After each cycle, the tightness and separation between the clustered sets are first measured to assess the impact of the clustered sets dynamically. Next, by giving the typical privacy budget allocation to some weight, the assessment results are included in the process. Last, to adaptively introduce perturbation noise to each group, distinct privacy budgets are allocated to various sets of clusters during the iteration. Though there are many works on differential privacy. Differential privacy introduces noise in the data to protect individual privacy. However, the prediction accuracy significantly degrades, limiting its use in ML models.

B. Works Related to Anonymization

Anonymization is a practical solution that allows data owners, such as hospitals, banks, social network service providers, and insurance companies, to protect their user's privacy when publishing data. By anonymizing the data, it remains useful for legitimate information consumers while preventing the identification of specific individuals. Many models, algorithms, frameworks, and prototypes were developed to achieve privacy-preserving data publishing (PPDP). The goal is to maintain the overall utility and integrity of the data while ensuring that individuals cannot be identified for analysis or other purposes. Several widely used methods for maintaining privacy while anonymizing data include k-anonymity [17], l-diversity [18], t-closeness [19], and amplified randomization [20]. Healthcare data that contain sensitive information are at risk of being attacked when stored on public clouds. To address this, Wang et al. [21] introduced a framework designed to facilitate the outsourcing of healthcare data with high dimensionality to a cloud infrastructure. The framework comprises first classifying the data into segments that are sensitive and nonsensitive. The sensitive data is then stored in a private cloud, while the nonsensitive data is kept in a public cloud. To protect the sensitive data, differential privacy noise is injected into it. K

anonymity guarantees that within a dataset table, a minimum of k records exhibit similarity. However, despite this safeguard, such datasets remain susceptible to identity and attribute disclosure attacks. Kim and Chung [22] proposed a protocol that uses k-anonymity to prevent identity disclosure attacks, which can be categorized as internal or external. Internal identity disclosure occurs when the data collector can identify the data holder, on the other hand, external identity exposure occurs when a person's identity is revealed via network headers. To prevent such attacks, they proposed k-anonymity models. These models guarantee that a minimum of k records possess identical quasi-identifiers. Additionally, on the data collector's end, each group comprises at least k data holders whose quasi-identifiers are the same. Zhang et al. [23] introduces MRMondrian, a scalable MapReduce-based approach for the multidimensional anonymization of big data. The approach addresses the challenges of privacy preservation in cloud computing by leveraging the MapReduce paradigm to achieve scalability and efficiency in anonymizing large datasets. The article [24] describes the development of an unmanned aerial vehicle (UAV) system with a deep learning-based face anonymizer to protect people's privacy in videos captured by UAVs. The system aims to maintain the semantic information of the footage while anonymizing individual faces. The face anonymizer uses generative adversarial networks to modify facial features to ensure anonymity while preserving a human-like appearance. The system is implemented on a UAV platform and is shown to effectively anonymize individuals in first-person videos without resembling anyone in the dataset used. Moreover, the anonymized videos do not degrade the system's perception performance for essential functions such as simultaneous localization and mapping. Anonymization often involves removing or masking identifying information using techniques such as generalization, suppression, or noise addition, which can lead to a loss of valuable features that might be important for the model's performance, thereby reducing the utility of the anonymized data.

C. Works Related to Synthetic Data

High-quality data is crucial for artificial intelligence and machine learning techniques. However, obtaining real data can be expensive and challenging. It may contain personal and confidential information that no one wants to get published. The kind of data can be utilized for different types of problems, such as predicting machine life span and analyzing changes in climate, and health. Unfortunately, such data is not always available due to privacy concerns, its high cost, and the need for experts to collect it. This scarcity of data can be a hindrance to processing and analyzing research data in general. Real data can be substituted with synthetic data for processing and analysis. It can be used to generate rare data that are difficult to collect, such as equipment breakdowns, unusual weather, and unusual disease symptoms, to augment or increase the amount of data for training machine learning models. One promising approach to generating synthetic data is through generative adversarial networks (GANs). The author in [25] first introduced GAN as a model to generate synthetic data for various applications. Some of the popular GAN models are conditional generative adversarial network (CGAN), deep generative adversarial network (DGAN), information maximizing generative adversarial network (InfoGAN), coupled generative adversarial networks (CoGAN), auxiliary classifier generative adversarial network (AC-GAN) along with numerous other variations of GANs [26]. Antonio et al. [27] explore the utility of synthetic data generation algorithms using CTGANs in medical tabular databases for disease prediction. The study demonstrates the effectiveness of synthetic data generation techniques in small and imbalanced datasets, maintaining classification performance comparable to real data and even improving it in some cases. Using residual GCB-Net [28], a CNN-based network, author demonstrated the excellent classification performance. Utilizing deep learning techniques based on autoregressive convolutional recurrent neural networks (CRNNs), the authors in [29] have produced synthetic data for ultrawideband (UWB) and ultrahigh-frequency radio frequency identification (UHF-RFID) sensors through multivariate time series prediction. The contents of free electronic medical records (EMRs) are scarce, include a wealth of personal data, and lack uniform standards. Furthermore, it might be challenging to have an adequate quantity of specimens for the various illness types being studied. These issues have hampered the advancement of ML and NLP techniques for EMR data processing. The authors in [30] have created a model known as medical text generative adversarial network, or mtGAN, to produce synthetic EMR text to address these issues. It creates synthetic texts as electronic medical records (EMRs) for the associated diseases using disease tags as inputs. Chougule et al. [31] proposed a GAN for synthetic data generation technique for controller area networks (CANs) to address the lack of availability of large amounts of data required for developing security mechanisms in CANs. The authors propose a GAN-based technique to generate synthetic CAN datasets, which can be tailored to specific needs or conditions not available in real data. The author in [32] proposes user-driven synthetic dataset hunting methods to generate synthetic datasets with specified privacy objectives, allowing data owners to release datasets with confirmed privacy levels. It is essential to evaluate the prediction accuracy of synthetic data compared to differentially private and anonymized data under different conditions. This will prove the utility of synthetic data in comparison to other PETs. To the best of our knowledge, there is very little work done on thorough performance analysis of synthetic data, and the aim of this work is to do the same.

IV. Preliminaries

The task is to compare the performance of the machine learning model for synthetic data with that of differential privacy and anonymized data. Hence, another component is to introduce the element of differential privacy, which is done by adding noise (for instance, Laplace noise) to the gradients and then using these gradients to update the weights to train with other trainable variables, the procedure to update the gradient is shown in Fig. 1. The previous section lists work related to such

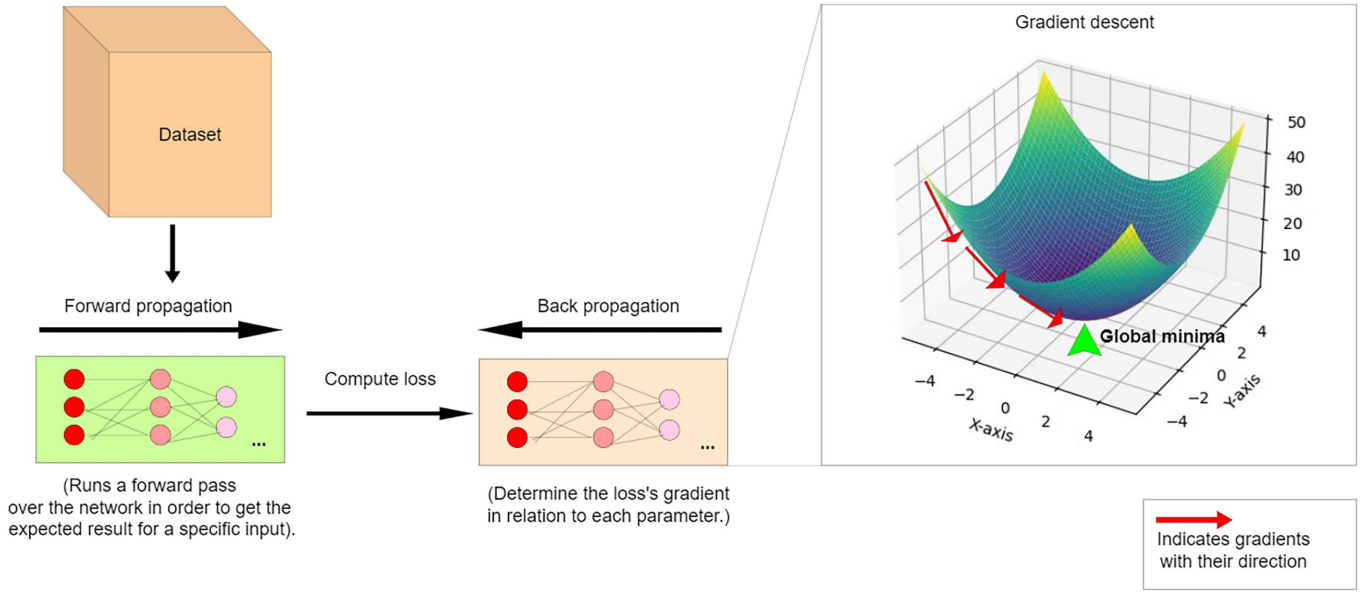

Fig. 1. Differential privacy in machine learning.

techniques concerning machine learning applications. It can be argued that there is no need for such a comparison between these techniques. However, the nature of their functioning lays the ground for such a study. Either of these meets our requirements, but the objective is to determine which one emphasizes both accuracy and security in the model. The following are the key points of differential privacy in machine learning:

1) Here, privacy is instilled within the model by adding noise to the gradients. Subsequently, the weights are updated while training on the trainable variables.
2) One could argue that adding noise directly to the data points might be sufficient. However, this would simply move the values from one point to another by a specific distance, making it easier for adversaries to trace back to the original data points.
3) Adding noise to the gradients brings arbitrary changes into the model as the accuracy and the final result would be known only when the training and testing have occurred through the specified number of iterations.

The previous section has already listed an abundance of the techniques concerned. However, they have been unable to establish which is more suitable considering data privacy across machine learning. In this work, we use datasets with some numerical features and one categorical feature of values, either 0 or 1. To begin with, we employ generative adversarial networks (GANs), a method that generates synthetic data that mimics the original dataset, on which a machine learning model can be trained. The main objective is to compare the accuracy of the ML model when trained and tested on the original and synthetic one. This method, in the interest of our study, serves the following intentions:

1) Since the idea is to establish the security of data while in use, generating synthetic data distills the knowledge contained by the original data into data points that are not members of the source. In other words, the synthesized data points together capture the distribution without exposing the identity of any particular point. Given the capabilities of an adversary, attempts may include measures such as randomly querying the dataset for data leakage or testing the model with random mini-batches of data points. However, if the members targeted are not a part of the training data itself, despite achieving expected answers, the rival will not gain any insight.
2) A synthetic dataset that is smaller in size than the original reduces the need for managing extensive storage, making it more cost-effective.
3) One might argue that using training data other than the distribution of the testing data leaves the possibility of an inaccurate model unattended. However, the synthetic dataset imitates the original distribution but might not be included within the source. Since the task of an ML model is to identify the pattern in which a distribution behaves, the generated data produces results similar to what the original produces.

V. Evaluation Methodology

A. Differential Privacy in Machine Learning

Here, gradients are manipulated by adding any noise. These updated gradients subsequently update the weights and biases, which ultimately train the model. The approach to differential privacy in machine learning proceeds as shown in Fig. 2:

1) *Initialize parameters:* Start by setting initial values for the neural network's weights and biases. These values should be modest and randomized.
2) *Forward pass:* Conduct a forward pass through the network to predict the output for a given input. This involves using the current set of parameters to propagate the input through each layer.

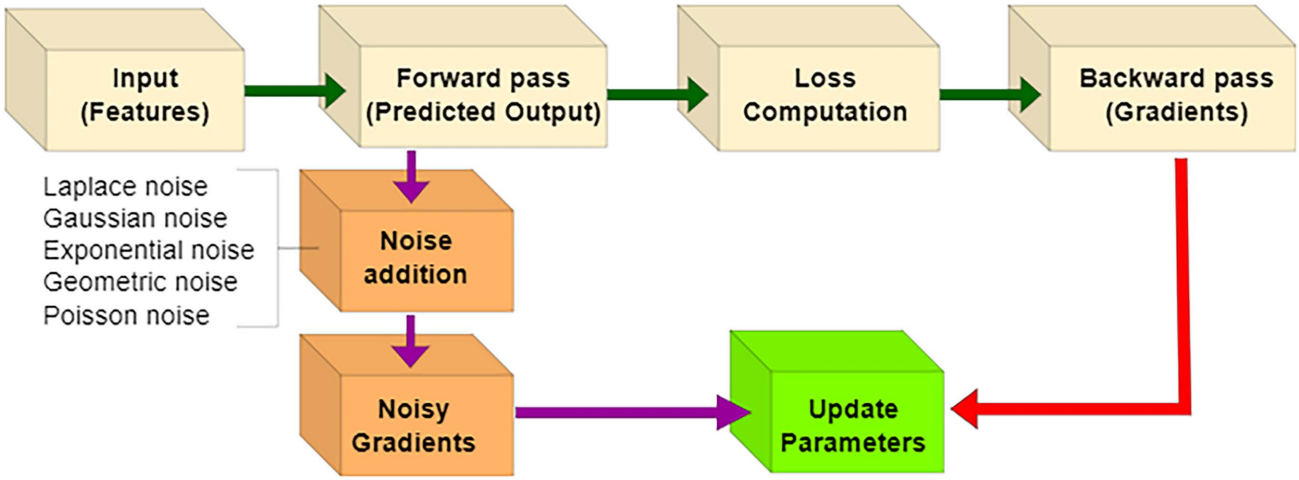

Fig. 2. Parameter update by adding different noise.

TABLE I
NOTATION SUMMARY

Notations	Meaning
$\nabla J(\theta)$	Loss function's gradient
θ	Model's parameters
b	Scale of Laplace distribution
ν	Mean of the Laplace distribution
α	Learning rate
p_n	New parameter
p_o	Old parameter
m	Batch size
x_i	Real or generated samples
y_i	Corresponding labels
z_i	Random noise vectors
$D(G(z_i))$	Discriminator's confidence in classifying the generated samples as real.

3) *Compute loss:* Determine the amount of inaccuracy or loss by contrasting the actual (ground truth) result with the output that was expected.
4) *Backward pass (Backpropagation):* Determine the loss gradient for each component, including weights and biases. To achieve this, propagate the error backward through the network. The gradients indicate both the magnitude and direction of adjustments needed to minimize the loss. Various notation used is listed in Table I.

Let $\nabla J(\theta)$ represent the loss function's gradient concerning the model's parameters θ. Now, we must add Laplace noise to every gradient vector member. Two parameters define the Laplace distribution: scale (b) and location (ν).

$$\text{Noisy gradient} = \nabla J(\theta) + \text{Laplace}(\nu, b) \tag{1}$$

where Laplace(ν, b) is a random variable drawn from Laplace distribution with mean ν and scale b. Laplace distribution is represented as follows:

$$f(x|\nu, b) = \frac{1}{2b} \exp\left(-\frac{|x-\nu|}{b}\right) \tag{2}$$

5) *Update the parameters:* Adjust the parameters to minimize the loss. This involves subtracting a fraction of the gradient from the current parameter values. The learning rate, a hyperparameter governing the size of each optimization step, determines the fraction in the update equation

$$p_n = p_o - \alpha \times \text{Noisy gradient} \tag{3}$$

where p_n is the new parameter after updation and α is the learning rate

By adding randomness to the gradient developments, this procedure makes it harder for an opponent to deduce details about specific training samples. Selecting the learning rate and laplace noise parameters (ν and b) is an important decision that frequently necessitates balancing model accuracy with privacy. The above sequence can also be done for other noises, and the distribution for that is shown in Section VI.

B. Anonymization

Anonymization is the process of transforming or modifying personal or sensitive information so that it can no longer be attributed to an individual. The goal is to protect the privacy of individuals while allowing for the use of the data for analysis, research, or other purposes. Anonymization techniques are commonly employed in various fields, including healthcare, finance, and research, to mitigate the risk of reidentification and unauthorized disclosure of sensitive information. We have used k-anonymity to anonymize the data. Sweeney [17] first proposed the k-anonymity anonymization concept for exchanging personal data. According to this notion, personal records are similar to at least k-1 records in terms of certain quasi-identifiers. There are two methods to achieve k-anonymity, i.e., suppression and generalization. Suppression replaces some entries with an asterisk "*" while generalization groups entries into categories. The steps for k-anonymization are

1) *Identify quasi-Identifiers (QI):* Identify the attributes in the dataset that, when combined, could potentially identify individuals. These are quasi-identifiers.

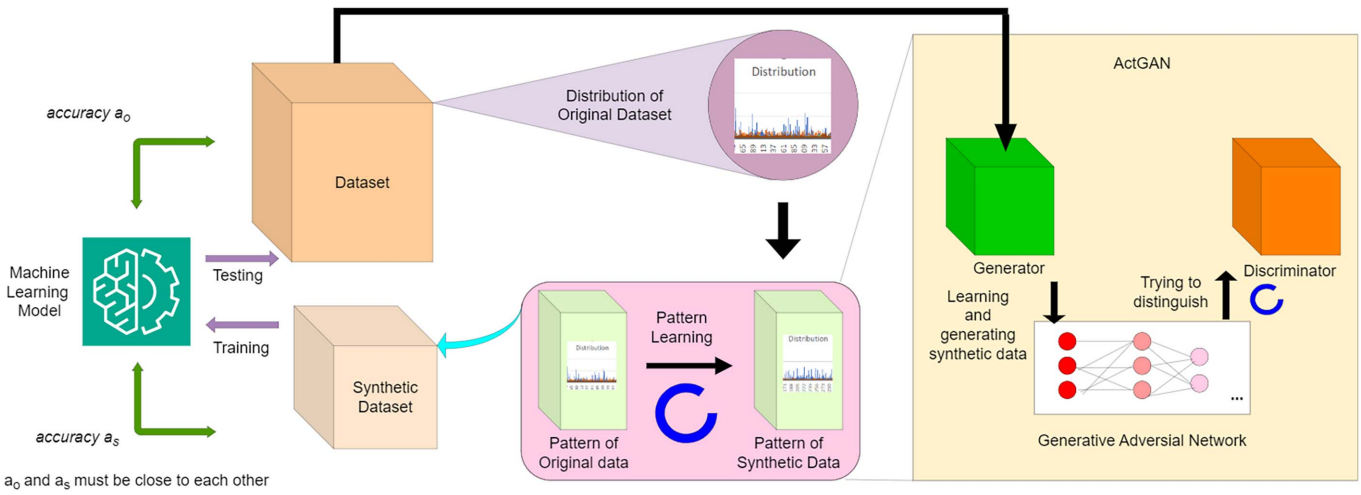

Fig. 3. System model for synthetic data: The objective is to make sure that accuracy with synthetic data a_s and original data a_o are close to each other.

2) *Group records:* Group the records based on the quasi-identifiers to form equivalence classes. All records in an equivalence class are considered indistinguishable with respect to the quasi-identifiers.
3) *Apply generalization:* Generalize the quasi-identifiers to create more generic or abstract representations. This involves making the values less specific.
4) *Ensure equivalence class size (k):* Ensure that each equivalence class has a size of at least k. This is the core requirement for k-anonymity.
5) *Suppress identifiers:* Suppress or remove specific values to achieve indistinguishability. This involves hiding or replacing certain quasi-identifier values.

C. Synthetic Data

As mentioned in the previous section, generating synthetic data out of the original data is one of the techniques we have studied concerning privacy when data is in use. Over the past few decades, evolution in methods to generate synthetic data has brought forth a plethora of algorithms. However, we never have a single solution that fits all, especially concerning data. The reason is that real-life data often suffer from class imbalance. precisely when one of the classes remains underestimated. A common method for balancing the proportion between the classes in unbalanced data is minority oversampling [33]. We have employed anyway conditional tabular generative adversarial network (ACTGAN) [34] to generate synthetic data. ACTGAN was chosen because its active learning mechanism helps ensure that the generated data covers a diverse range of patterns and variations present in the real data. This can result in synthetic datasets that are more representative and comprehensive, enhancing the generalization performance of models trained on them. In generative adversarial networks, two networks, a discriminator (D) and a generator (G), learn in a competitive environment. G's goal is to generate fictitious observations that deceive D, as it gets trained to link random noise created through a spectrum to data values in the training dataset. D has been trained to distinguish between true and fraudulent input. This indicates if it is or is not from the desired distribution [33]. The operation of ACTGAN is as shown in Fig. 3:

1) *Initialization:* First, random weights are used to initialize the discriminator and generator networks.
2) *Training the discriminator:* This stage uses a labeled dataset to train the discriminator network. The discriminator's goal is to accurately classify produced and actual samples. This training phase usually uses the binary cross-entropy loss as the loss function. The discriminator minimizes the binary cross-entropy, and the loss is given by

$$\mathrm{L}D = -\frac{1}{m}\sum i = 1^{m} \left(y_i \log(D(x_i)) + (1 - y_i)\log(1 - D(x_i))\right) \quad (4)$$

where m is the batch size, x_i are the real or generated samples, and y_i are the corresponding labels (1 for real samples, 0 for generated samples).
3) *Active learning:* An active learning stage is implemented once the discriminator has been trained. In this stage, a subset of the produced items that the discriminator has highly confidently categorized as real are chosen. The discriminator is then retrained using these chosen samples that have been introduced to the training set.

   During the active learning step, a subset of generated samples, denoted as G_{selected}, is selected based on the discriminator's confidence in classifying them as real. These selected samples are then added to the training set.
4) *Training the generator:* Training the generator in a generative adversarial network (GAN) involves iteratively optimizing the parameters of the generator neural network to produce realistic synthetic data. This process is part of a dynamic interplay with the discriminator, which attempts to distinguish between real and generated data. The generator is designed with an architecture that takes random noise as input and produces synthetic data

while the discriminator assesses the authenticity of the input. The training loop includes generating synthetic data, training the discriminator on a mix of real and generated samples, and updating the generator to improve its ability to deceive the discriminator. Fine-tuning involves adjusting hyperparameters and monitoring metrics such as loss and accuracy. GAN training is an iterative process that requires careful balancing to achieve optimal results and generate high-quality synthetic data. The generator minimizes the negative binary cross-entropy loss given by:

$$\text{LG} = -\frac{1}{m}\sum i = 1^m \log(D(G(z_i))) \tag{5}$$

where m is the batch size, z_i are random noise vectors, and $D(G(z_i))$ represents the discriminator's confidence in classifying the generated samples as real.

VI. Experimentation and Results

Initially, it was essential to evaluate the performance of a machine learning model without integrating either differential privacy, anonymization, or synthetic data. This initial assessment provided insights into the baseline accuracy achievable by a conventional model. While adjusting hyperparameters can enhance model accuracy, it is crucial to underscore our primary objective, i.e., determining which approach strikes the right balance between data privacy (in use) and utility. The focus is on comparing methods and understanding their impact on the model's effectiveness while considering privacy concerns. For this purpose, we chose the PIMA Indian diabetes dataset [35]. There are 768 instances with the following independent variables/features:

1) Number of pregnancies (numerical value)
2) Plasma glucose concentration in an oral glucose tolerance test (numerical value)
3) Blood pressure in units of mm Hg (numerical value)
4) Triceps skin fold thickness in units of mm (numerical value)
5) Insulin content in units of μ U/mL (numerical value)
6) Body mass index (numerical value)
7) Diabetes pedigree function (numerical value)
8) Age (numerical value)

The target variable is "Outcome," which is a categorical variable that denotes whether the patient has diabetes or not. We defined a funnel-shaped neural network architecture to train as it aims to capture a hierarchical representation of the features besides reducing the dimensionality of the architecture. The neural network architecture used is

1) An input layer of eight neurons
2) A hidden layer of five neurons and activation function of "ReLU"
3) A single neuron output of "sigmoid" activation function
4) Loss function: binary cross-entropy
5) Optimizer: Adam

We refer to the above network architecture as *Network architecture-1*. This also enables parameter efficiency as more neurons in the first layers enable the collection of a wide variety of features; fewer neurons in the later layers enable the consolidation and more condensed representation of these features. This may result in a more frugal model. This also helps mitigate overfitting, as reducing the number of neurons imposes a form of simplicity on the model. The model compiled is tested for a set of learning rates, which is {0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5}. Table II records the respective accuracies of a neural network for the different learning rates mentioned.

TABLE II
Performance of a Neural Network Without Implementing PETs

Learning rate	Accuracy
0.0001	0.7229
0.0005	0.7229
0.001	0.7532
0.005	0.7619
0.01	0.7229
0.05	0.7575
0.1	0.7445
0.5	0.7229

A. Differential Privacy

As elaborated in the preceding section, our approach is to instill differential privacy in our experiments, which involves introducing noise to the gradients. Notably, Laplace noise is not the sole choice available for this purpose. We have identified numerous alternative distributions from which noise can be generated and incorporated into the calculations with the model. The distributions we have examined include:

1) *Laplace noise:* A Laplace distribution with mean ν and scale b is represented as follows:

$$f(x|\nu, b) = \frac{1}{2b}\exp\left(-\frac{|x-\nu|}{b}\right). \tag{6}$$

2) *Gaussian noise:* The probability density function (PDF) of a Gaussian distribution with mean μ and standard deviation σ is given by

$$\text{f}(x) = \frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \tag{7}$$

3) *Exponential noise:* The probability density function (PDF) of an exponential distribution with rate parameter λ is given by

$$\text{f}(x) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & x < 0 \end{cases}. \tag{8}$$

4) *Geometric noise:* The probability mass function (PMF) of a geometric distribution with success probability p is given by

$$\text{P}(X = k) = (1-p)^{k-1}p \tag{9}$$

where X is the random variable representing the number of trials until the first success.

TABLE III
ACCURACY OF THE MODEL WITH DIFFERENT LEARNING RATES

Learning Rate	Accuracy							
	Original	Laplace	Gaussian	Exponential	Geometric	Poisson	Anonymized	Synthetic
0.0001	0.7446	0.5486	0.5632	0.5352	0.5778	0.5365	0.7229	0.7878
0.0005	0.7099	0.5113	0.5938	0.5392	0.6177	0.5751	0.7099	0.7922
0.001	0.7186	0.5911	0.4660	0.6097	0.6536	0.5685	0.7402	0.7965
0.005	0.7575	0.5192	0.5831	0.6217	0.6536	0.6031	0.7186	0.7922
0.01	0.7272	0.6017	0.6536	0.5964	0.6536	0.6470	0.7316	0.7835
0.05	0.7359	0.6536	0.6536	0.6536	0.6536	0.6536	0.7012	0.7662
0.1	0.7359	0.6536	0.6536	0.6536	0.6536	0.6536	0.7489	0.8268
0.5	0.7575	0.6536	0.6536	0.6536	0.6536	0.6536	0.7186	0.8008

5) *Poisson noise:* The probability mass function (PMF) of a Poisson distribution with rate parameter λ is given by

$$\mathrm{P}(X = k) = \frac{e^{-\lambda}\lambda^{k}}{k!}. \tag{10}$$

For each type of noise with a noise scale of 1 mentioned above, a neural network with a funnel architecture was trained with eight different learning rates to observe the convergence of these models. Table III records the accuracies with the addition of Laplace, Gaussian, Exponential, Geometric, and Poisson noise, respectively. When compared with the original data in Table II, the accuracies are lesser than the values for the respective learning rates, hence proving to be of lesser utility than a neural network without adding any noise to the gradients. However, utility is not our only purpose privacy, being the other side of the coin, needs discussion as well.

Any algorithm is termed as differentially private if given an output, an observer cannot be certain that a particular data point is used or not. Section V gives the mathematical approach to introduce privacy through the addition of noise to the gradients and not the data points themselves. Adding noise to the data points can also introduce privacy however, upon aggregating the values, the noise is subdued, obtaining results closer to the original answer. Hence, this is the desirable approach. Since the gradients are manipulated, the training data remains unchanged, and if an adversary attempts to query the model with data points similar to those in the original dataset, the accuracy gets closer to the original value, revealing whether or not a particular data point is present. Table III record the accuracies upon the addition of Laplace, Gaussian, Exponential, Geometric, and Poisson noise to the gradients, respectively. When compared with the values of Table II, which stores the accuracies of a neural network without adding any noise, the latter's utility is way better. Fig. 4 infer that the addition of perturbations does not assist in maintaining a balance between privacy and utility.

B. Anonymized Data

This privacy-enhancing technology (PET) is designed to alter or manipulate data with the goal of safeguarding individual privacy while maintaining the utility of the information. The approach involves applying k-anonymity to achieve data anonymization, with a specified parameter $k = 5$, ensuring that the anonymized data comprises at least five rows with identical values. The generalization technique is employed to generalize the quasi-identifier, enhancing privacy. The resulting anonymized data is then utilized for training and testing neural networks, with the corresponding accuracy recorded in a Table III. Furthermore, a comparative analysis depicting the accuracy of the anonymized data in contrast to the original data is presented in Fig. 4.

C. Synthetic Data

This pertains to another PET approach on which experiments were conducted. The strategy involves generating synthetic data derived from the original dataset, serving as input for training the neural network. The newly created dataset must emulate the original information's behavioral patterns. To qualify as appropriate synthetic data, it should mirror the behavior of the data from which it was generated. For instance, if the original set exhibits the second-highest correlation between features A and B, the synthetic data should similarly exhibit this correlation. To achieve this, we employ ACTGAN, as previously outlined in Section V. The approach adheres to a specific methodology, as detailed below:

1) Generate synthetic data using ACTGAN as mentioned in Section V.
2) Use the same neural network as the one used for original and differential privacy to train and test the generated dataset, the results of which are recorded in Table III.
3) The accuracies of the synthetic dataset and the original dataset are compared and shown in Fig. 4.

A comparison between the accuracies of these models for different learning rates and different noise vectors is recorded in Table III and also shown in graphical form in Fig. 4. It is evident from these figures that when trained using a synthetic dataset, the accuracies are even better than that of the original data, unlike what was observed regarding differential privacy and anonymized data. From Fig. 4, it is evident that the model trained on synthetic data achieves better accuracy compared to the model trained on original data. However, one must remember that privacy is another key aspect to be considered. Given that the data utilized for training the network is synthetic, representing a replication of the original data, the specific data point sought by potential adversaries is nonexistent. Consequently, despite obtaining reasonably accurate outputs, these entities are unable to discern the original data point. This underscores the

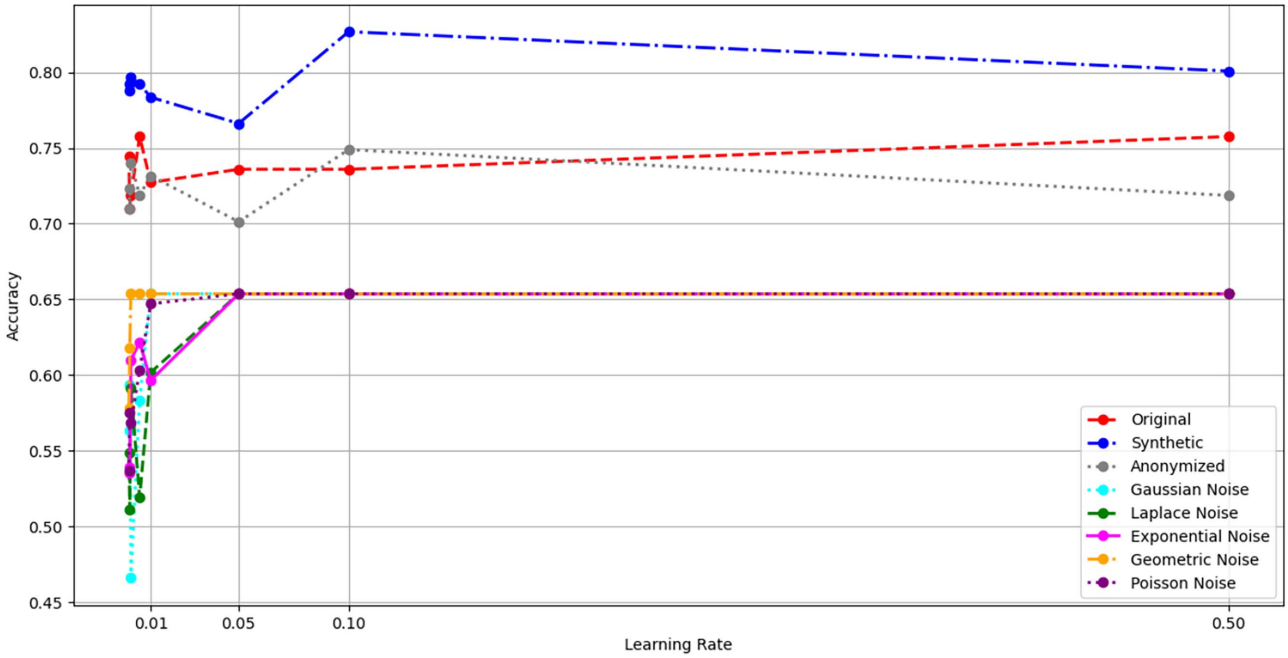

Fig. 4. Accuracy of the models with different learning rates *(Network architecture-1)*.

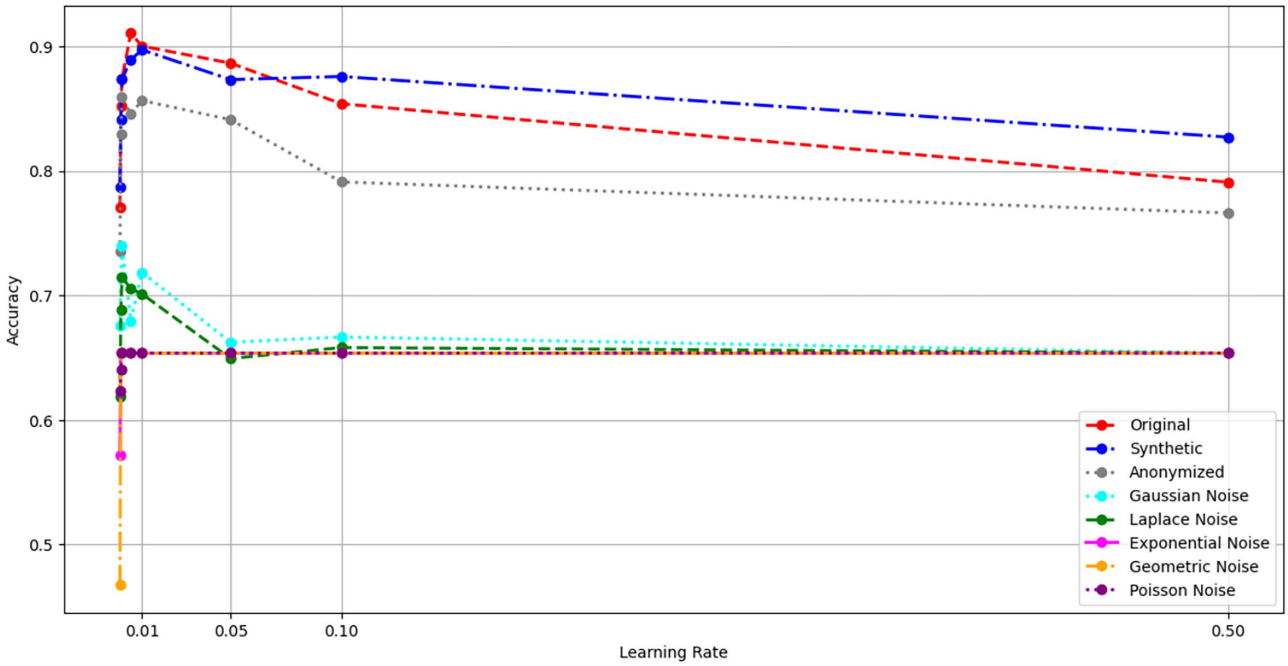

Fig. 5. Accuracy of the models with different learning rates *(Network architecture-2)*.

effectiveness of synthetic data in striking a balance between utility and privacy, which has been our objective since the beginning.

D. Comparative Analysis

1) Accuracy Versus Learning Rate (Different Network Architecture): We have performed the experiments mentioned in Sections VI-A, VI-B, and VI-C on neural networks with an architecture different from that of *Network architecture-1*. We refer to the architecture of this new neural network as *Network architecture-2*. The architecture of this new neural network is

a) An input layer of 8 neurons
b) First dense layer with 64 neurons and an activation function of "ReLU."

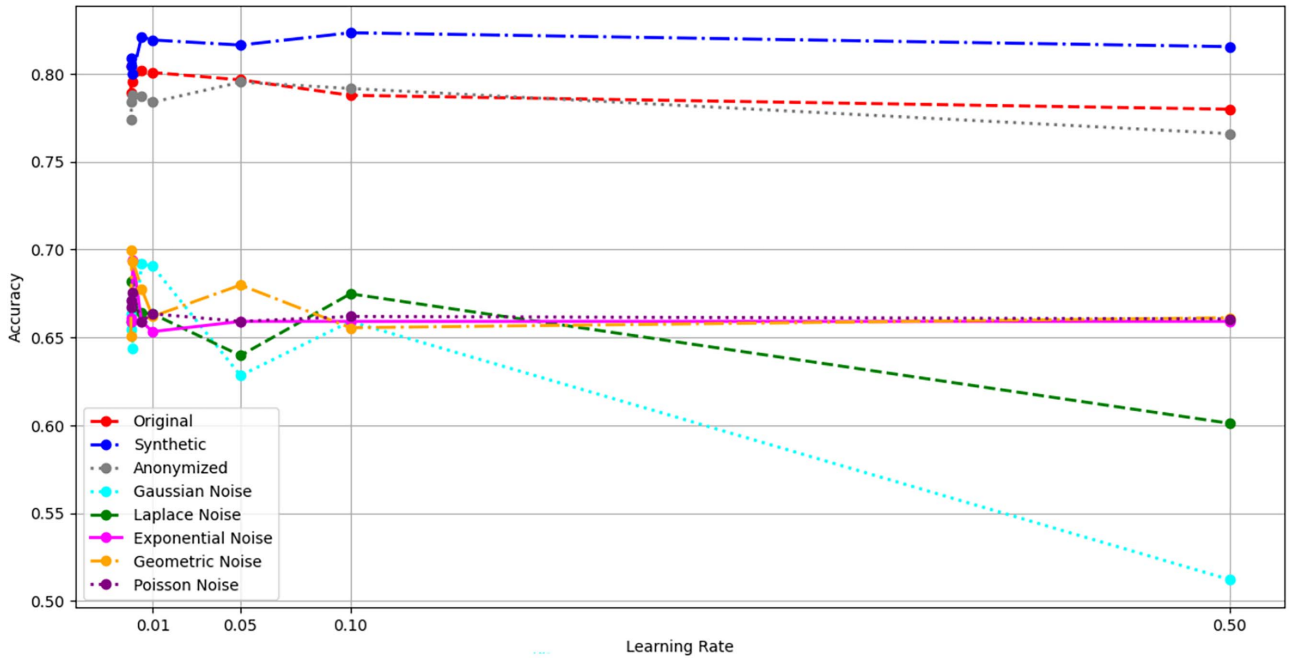

Fig. 6. Accuracy of the models with different learning rates (Employee data).

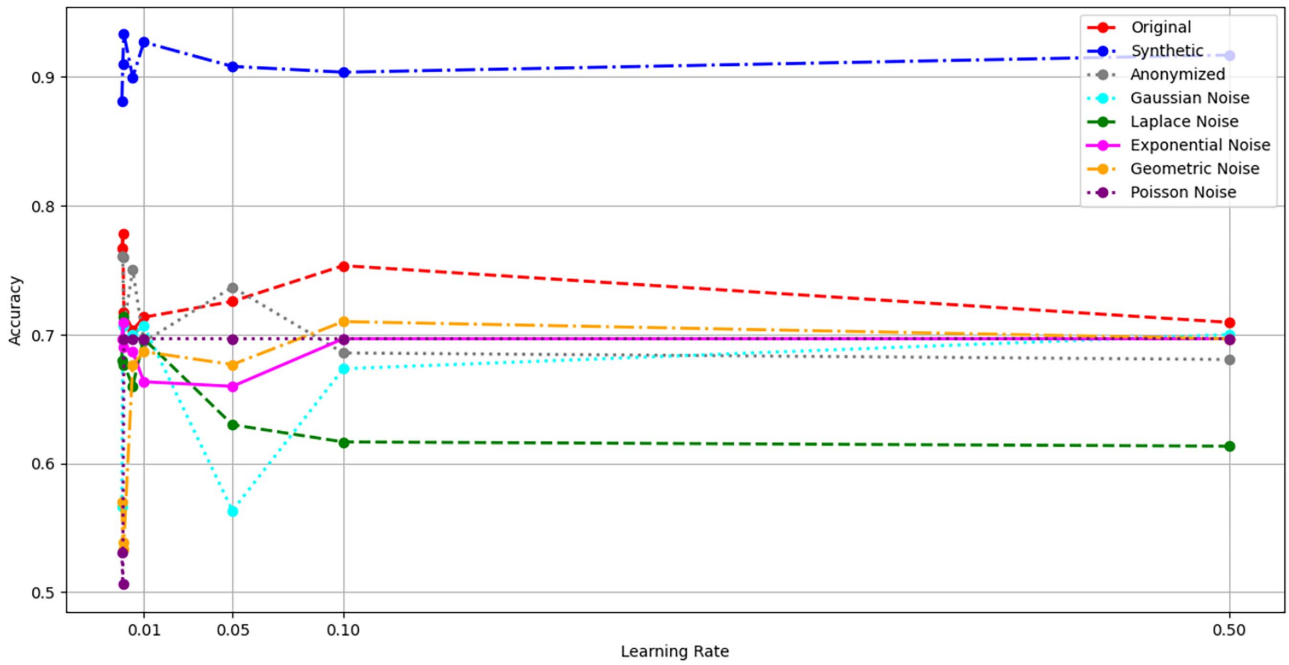

Fig. 7. Accuracy of the models with different learning rates (Credit data).

c) Batch normalization layer after the first dense layer to normalize the activations.
d) Dropout layer with a dropout rate of 0.3 to prevent overfitting.
e) Second dense layer with 32 neurons and an activation function of "ReLU."
f) Batch normalization layer after the second dense layer to normalize the activations.
g) Dropout layer with a dropout rate of 0.3 to prevent overfitting.
h) Third dense layer with 16 neurons and an activation function of "ReLU."
i) Batch normalization layer after the third dense layer to normalize the activations.
j) Loss function: Binary cross-entropy.
k) Optimizer: Adam

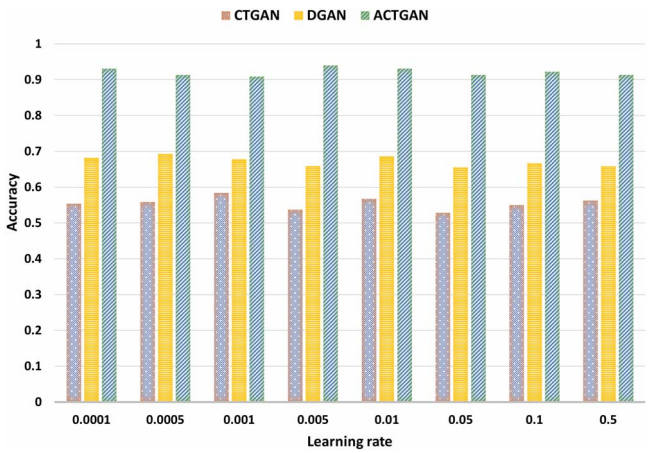

Fig. 8. Accuracy of the models with different learning rates.

The accuracy of the original, differentially private, anonymized, and synthetic data is compared. We have found that the accuracy of synthetic data is better than that of differentially private data and anonymized data. The accuracy of synthetic data is better or comparable to that of original data, which is shown in Fig. 5.

2) Accuracy Versus Learning Rate (Different Datasets): We have performed the same experiments on two different datasets: Employee datasets [36] and German credit datasets [37]. The employee dataset contains 4653 instances with the independent variables- Education, Joining year, City, Payment tier, Age, Gender, Ever benched, Experience in Current Domain, and Leave or Not (will leave the company or not), which is a target column. The German credit data dataset contains 1000 instances with the features- Checking status, Duration, Credit history, Purpose, Credit amount, Savings status, Employment, Installment commitment, Personal status, Other parties, Residence since, Property magnitude, Age, Other payment plans, Housing, Existing credits, Job, Num dependents, Own telephone, Foreign worker and class (good or bad credit risks) a classification column. We first performed differential privacy by adding different types of noise to the original data at the gradient level. We then anonymized the original data using k-anonymization. Finally, we have created the synthetic version of the original data. Then, we compared the accuracy of original data, differential privacy, anonymized data, and synthetic data. we have found that synthetic data has better accuracy than original data, differentially private data, and anonymized data. This shows that synthetic data can be used instead of original data, ensuring privacy without compromising accuracy. A comparison of the accuracy of the employee data is shown in Fig. 6, and the accuracy comparison of the credit data is shown in Fig. 7.

3) Accuracy Versus Learning Rate (Different GAN Model): We have generated synthetic data from three different GAN models: CTGAN, DGAN, and ACTGAN. We have observed that ACTGAN gives better accuracy than other GANs and even better than the original data, as shown in Fig. 8.

VII. Conclusion

Differential privacy and anonymization ensure data privacy. However, the data generated by these techniques have low prediction accuracy, limiting its utility for various applications. Synthetic data, when generated properly, can retain the essential characteristics of the original data while omitting or modifying identifying details, providing more flexibility, better privacy guarantees, and maintaining more utility for machine learning models. However, very little work has been done to evaluate the utility of synthetic data for different model parameters. We have addressed this important issue in our work by generating synthetic data through ACTGAN, which is a synthetic dataset that is more representative and comprehensive. We have then performed a thorough performance analysis of the utility of synthetic data in comparison to differentially private and anonymized data for different learning rates, network architectures, and datasets from varied domains. We have found that synthetic data has higher utility (prediction accuracy) than differentially private and anonymized data and comparable accuracy to that of original data. This shows synthetic data is promising PETs for various applications.

VIII. Future Scope

This study measures the precision of models under different learning rates when disturbances are introduced. However, it does not evaluate the extent of data leakage or privacy infringement in the models, except for a qualitative analysis of the situation. Therefore, further research could focus on determining the degree of privacy infringement.

References

[1] T. D. Dang, D. Hoang, and D. N. Nguyen, "Trust-based scheduling framework for big data processing with mapreduce," *IEEE Trans. Services Comput.*, vol. 15, no. 1, pp. 279–293, 2022.

[2] N. Shaltout, A. A. El-Latif, W. M. El-Latif, and S. Elmougy, "Applicable image security based on computational genetic approach and self-adaptive substitution," *IEEE Access*, vol. 11, pp. 2303–2317, 2023.

[3] W. Iqbal, H. Abbas, B. Rauf, Y. A. Bangash, M. F. Amjad, and A. Hemani, "PCSS: Privacy preserving communication scheme for SDN enabled smart homes," *IEEE Sensors J.*, vol. 22, no. 18, pp. 17677–17690, 2022.

[4] N. Smart, "Computing on encrypted data," *IEEE Secur. Privacy*, vol. 21, no. 4, pp. 94–98, 2023.

[5] M. Naehrig, K. Lauter, and V. Vaikuntanathan, "Can homomorphic encryption be practical?" in *Proc. 3rd ACM Workshop Cloud Comput. Secur. Workshop*, 2011, pp. 113–124.

[6] D. Evans et al., "A pragmatic introduction to secure multi-party computation," *Found. Trends Privacy Secur.*, vol. 2, no. 2–3, pp. 70–246, 2018.

[7] A. Majeed and S. Lee, "Anonymization techniques for privacy preserving data publishing: A comprehensive survey," *IEEE Access*, vol. 9, pp. 8512–8545, 2020.

[8] C. Dwork, "Differential privacy," in *International Colloquium on Automata, Languages, and Programming*. Springer, 2006, pp. 1–12.

[9] S. I. Nikolenko, "Synthetic data for deep learning," 2019, *arXiv:1909.11512*.

[10] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li, and Y. Gao, "A survey on federated learning," *Knowl. Based Syst.*, vol. 216, p. 106775, 2021.

[11] Q. Mao, Y. Chen, P. Duan, B. Zhang, Z. Hong, and B. Wang, "Privacy-preserving classification scheme based on support vector machine," *IEEE Syst. J.*, vol. 16, no. 4, pp. 5906–5916, 2022.

[12] J. Soria-Comas, J. Domingo-Ferrer, D. Sánchez, and D. Megías, "Individual differential privacy: A utility-preserving formulation of differential privacy guarantees," *IEEE Trans. Inf. Forensics Secur.*, vol. 12, no. 6, pp. 1418–1429, 2017.

[13] D. Ye, S. Shen, T. Zhu, B. Liu, and W. Zhou, "One parameter defense—Defending against data inference attacks via differential privacy," *IEEE Trans. Inf. Forensics Secur.*, vol. 17, pp. 1466–1480, 2022.

[14] Y. Sei and A. Ohsuga, "Private true data mining: Differential privacy featuring errors to manage internet-of-things data," *IEEE Access*, vol. 10, pp. 8738–8757, 2022.

[15] M. Iqbal, A. Tariq, M. Adnan, I. Ud Din, and T. Qayyum, "Fl-ODP: An optimized differential privacy enabled privacy preserving federated learning," *IEEE Access*, vol. 11, pp. 116674–116683, 2023.

[16] L. Han, Y. Xie, D. Fan, and J. Liu, "Improved differential privacy k-means clustering algorithm for privacy budget allocation," in *Proc. Int. Conf. Comput. Eng. Artif. Intell. (ICCEAI)*, 2022, pp. 221–225.

[17] L. Sweeney, "k-anonymity: A model for protecting privacy," *Int. J. Uncertainty, Fuzziness Knowl. Based Syst.*, vol. 10, no. 5, pp. 557–570, May 2002.

[18] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Trans. Knowl. Discovery Data (TKDD)*, vol. 1, no. 1, pp. 3–es, 2007.

[19] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *Proc. IEEE 23rd Int. Conf. Data Eng.*, 2006, pp. 106–115.

[20] A. Evfimievski, J. Gehrke, and R. Srikant, "Limiting privacy breaches in privacy preserving data mining," in *Proc. 22nd ACM SIGMOD-SIGACT-SIGART Symp. Princ. Database Syst.*, 2003, pp. 211–222.

[21] W. Wang, L. Chen, and Q. Zhang, "Outsourcing high-dimensional healthcare data to cloud with personalized privacy preservation," *Comput. Networks*, vol. 88, pp. 136–148, 2015.

[22] S. Kim and Y. D. Chung, "An anonymization protocol for continuous and dynamic privacy-preserving data collection," *Future Gener. Comput. Syst.*, vol. 93, pp. 1065–1073, 2019.

[23] X. Zhang et al., "Mrmondrian: Scalable multidimensional anonymisation for big data privacy preservation," *IEEE Trans. Big Data*, vol. 8, no. 1, pp. 125–139, 2017.

[24] H. Lee, M. U. Kim, Y. Kim, H. Lyu, and H. J. Yang, "Development of a privacy-preserving uav system with deep learning-based face anonymization," *IEEE Access*, vol. 9, pp. 132652–132662, 2021.

[25] I. Goodfellow et al., "Generative adversarial nets," *Adv. Neural Inf. Process. Syst.*, vol. 27, 2014.

[26] A. Figueira and B. Vaz, "Survey on synthetic data generation, evaluation methods and gans," *Mathematics*, vol. 10, no. 15, p. 2733, 2022.

[27] A. J. Rodriguez-Almeida et al., "Synthetic patient data generation and evaluation in disease prediction using small and imbalanced datasets," *IEEE J. Biomed. Health Inform.*, vol. 27, no. 6, pp. 2670–2680, Jun. 2023.

[28] Q. Li, T. Zhang, C. L. P. Chen, K. Yi, and L. Chen, "Residual GCB-net: Residual graph convolutional broad network on emotion recognition," *IEEE Trans. Cogn. Dev. Syst.*, vol. 15, no. 4, pp. 1673–1685, Apr. 2023.

[29] F. Romanelli and F. Martinelli, "Synthetic sensor data generation exploiting deep learning techniques and multimodal information," *IEEE Sensors Lett.*, vol. 7, no. 7, pp. 1–4, Jul. 2023.

[30] J. Guan, R. Li, S. Yu, and X. Zhang, "A method for generating synthetic electronic medical record text," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 18, no. 1, pp. 173–182, Jan. 2021.

[31] A. Chougule, K. Agrawal, and V. Chamola, "Scan-gan: Generative adversarial network based synthetic data generation technique for controller area network," *IEEE Internet Things Mag.*, vol. 6, no. 3, pp. 126–130, Mar. 2023.

[32] B.-C. Tai, Y.-T. Tsou, S.-C. Li, Y. Huang, P.-Y. Tsai, and Y.-C. Tsai, "User-driven synthetic dataset generation with quantifiable differential privacy," *IEEE Trans. Serv. Comput.*, vol. 16, no. 5, pp. 3812–3826, May 2023.

[33] A. Koivu, M. Sairanen, A. Airola, and T. Pahikkala, "Synthetic minority oversampling of vital statistics data with generative adversarial networks," *J. Amer. Med. Inform. Assoc.*, vol. 27, no. 11, pp. 1667–1674, 2020.

[34] C. Qi, Y. Jingjing, H. Ming, and Z. Qiang, "ACT-Gan: Radio map construction based on generative adversarial networks with act blocks," 2024, *arXiv:2401.08976*.

[35] J. W. Smith, J. E. Everhart, W. Dickson, W. C. Knowler, and R. S. Johannes, "Using the ADAP learning algorithm to forecast the onset of diabetes mellitus," in *Proc. Annu. Symp. Comput. Appl. Med. Care*. American Medical Informatics Association, 1988, p. 261.

[36] "Employee dataset," [Online] Available: https://www.kaggle.com/datasets/tawfikelmetwally/employee-dataset/discussion/481790 (Accessed: Aug 10, 2024).

[37] H. Hofmann, "Statlog (German Credit Data)," UCI Machine Learning Repository, 1994, doi: 10.24432/C5NC77.